

K
PLAI

Massachusetts Institute of Technology
Department of Ocean Engineering

INPUT-OUTPUT MANUALS FOR LINEAR
DYNAMIC PROGRAMS FOR COMPLIANT RISERS

by

G. A. Kriezis
N. M. Patrikalakis
C. Chryssostomidis

Design Lab. Report No. 85-2

November, 1985

ABSTRACT

The objective of this report is to present input-output manuals, sample runs and outputs of programs RCNATOUT, RCLINDYN, RCNATIN, RCLINDY1, RCFORC1, RCLINDY3, RCFORCE, RCLINDY2, ORTHOG and ORTHOG1 which are used in the

- o estimation of the out-of-plane natural frequencies and corresponding mode numbers
- o solution of the out-of-plane linear dynamic eigenproblem
- o estimation of the in-plane natural frequencies and corresponding mode numbers
- o solution of the in-plane linear dynamic eigenproblem
- o solution of the out-of-plane linear dynamic problem for a given frequency and amplitude of excitation at the top
- o alternate solution of the out-of-plane linear dynamic eigenproblem
- o solution of the in-plane linear dynamic problem for a given frequency and amplitude of excitation at the top
- o alternate solution of the in-plane linear dynamic eigenproblem
- o evaluation of the orthogonality condition of out-of-plane natural modes, and
- o evaluation of the orthogonality condition of in-plane natural modes

of a compliant riser with a planar static configuration without torsion, respectively.

The initial approximations and the theory implemented in these programs can be found in M.I.T. Sea Grant Report 85-19.

ACKNOWLEDGMENTS

Funding for this research was obtained from the MIT Sea Grant College Program, Chevron, Conoco and Minerals Management Service. P. R. McSweeney prepared the typed manuscript.

RELATED SEA GRANT AND DESIGN LABORATORY REPORTS

1. A Mathematical Model for Compliant Risers, by N. M. Patrikalakis and C. Chryssostomidis, MIT Sea Grant Report, No. 85-17, 1985.
2. Non-Linear Statics of Non-Rotationally Uniform Rods with Torsion, by N. M. Patrikalakis and C. Chryssostomidis, MIT Sea Grant Report, No. 85-18, 1985.
3. Input-Output Manuals for Non-Linear 2-D and 3-D Static Programs for Compliant Risers, by N. M. Patrikalakis and C. Chryssostomidis, MIT Design Laboratory Report, No. 85-1, September, 1985.
4. Linear Dynamics of Compliant Risers, by N. M. Patrikalakis and C. Chryssostomidis, MIT Sea Grant Report, No. 85-19, 1985.

TABLE OF CONTENTS

	<u>Page Number</u>
Abstract	2
Acknowledgments	3
Related Sea Grant and Design Laboratory Reports	3
Table of Contents	4
Chapter I	7
Introduction	7
Chapter II	10
II.1 Input-Output Manual for Out-of-Plane Natural Frequency and Mode Number Estimation Program, RCNATOUT	10
II.2 Sample Run, Input and Output	12
Chapter III	25
III.1 Input-Output Manual for the Out-of-Plane Linear Eigen- problem Solution Program, RCLINDYN	25
III.2 Sample Run, Input and Output	31
Chapter IV	44
IV.1 Input-Output Manual for In-Plane Natural Frequency and Mode Number Estimation Program, RCNATIN	44
IV.2 Sample Run, Input and Output	46
Chapter V	55
V.1 Input-Output Manual for the In-Plane Linear Eigenproblem Solution Program, RCLINDY1	55
V.2 Sample Run, Input and Output	61
Chapter VI	75
VI.1 Input-Output Manual for the Out-of-Plane Forced Linear Dynamic Problem Solution Program, RCFORC1	75
VI.2 Sample Run, Input and Output	77

	<u>Page Number</u>
Chapter VII	86
VII.1 Input-Output Manual for the Out-of-Plane Linear Eigen-problem Solution Program, RCLINDY3	86
VII.2 Sample Run, Input and Output	90
Chapter VIII	101
VIII.1 Input-Output Manual for the In-Plane Forced Linear Dynamic Problem Solution Program, RCFORCE	101
VIII.2 Sample Run, Input and Output	104
Chapter IX	109
IX.1 Input-Output Manual for the In-Plane Linear Eigen-problem Solution Program, RCLINDY2	109
IX.2 Sample Run, Input and Output	113
Chapter X	127
Input-Output Manuals for NAG Subroutines D02RAF, D02GBF, C05NBF, C05PBF, F04ATF, F03AAF	127
Chapter XI	161
Listing of Program RCNATOUT FORTRAN A	161
Chapter XII	167
Listing of Program RCLINDYN FORTRAN A	167
Chapter XIII	190
Listing of Program RCNATIN FORTRAN A	190
Chapter XIV	200
Listing of Program RCLINDY1 FORTRAN A	200
Chapter XV	227
Listing of Program RCFORC1 FORTRAN A	227
Chapter XVI	238
Listing of Program RCLINDY3 FORTRAN A	238

	<u>Page Number</u>
Chapter XVII	256
Listing of Program RCFORCE FORTRAN A	256
Chapter XVIII	267
Listing of Program RCLINDY2 FORTRAN A	267
Appendix A. Programs ORTHOG and ORTHOG1	285
1. Input-Output Manual for Program ORTHOG	286
2. Sample Run, Input and Output	287
3. Input-Output Manual for Program ORTHOG1	292
4. Sample Run, Input and Output	292
5. Listing of Program ORTHOG FORTRAN A	298
6. Listing of Program ORTHOG1 FORTRAN A	301

Chapter I

INTRODUCTION

The programs presented in this report solve the three-dimensional linear dynamic problem of a compliant riser with a planar static configuration without torsion. The sequence of steps the prospective user is supposed to follow are described next.

- o Initially the user should use program RCINPUT for the preparation of geometrical, structural and hydrodynamic characteristics of compliant risers. This program is described in [3].
- o Next the 2-D static compliant riser problem should be solved for the required excitation. Program RCSTAT2D is used for this step, as described in [3].
- o Next it is suggested that the user employs programs RCNATOUT and RCNATIN to obtain an estimate of the natural frequencies and mode numbers for out-of-plane and in-plane motion of a compliant riser with a planar static configuration without torsion. The above step is not appropriate in a case of zero current velocity since the asymptotic theory used in RCNATOUT and RCNATIN is no longer valid [4]. This special case is treated below.
- o Using the natural frequency estimates, the out-of-plane linear dynamic eigenproblem is solved using RCLINDYN and the in-plane linear dynamic eigenproblem is solved using RCLINDY1.
- o For the case of zero current velocity the out-of-plane linear dynamic problem is solved as follows:

The program RCFORC1 is used to obtain the out-of-plane linear forced dynamic response for a certain frequency and amplitude of excitation at the top without damping. The output from this program is then used as an input to program RCLINDY3, which solves the out-of-plane linear dynamic eigenproblem. The above method for zero current can be also used for arbitrary currents.

o For the case of zero current velocity the in-plane linear dynamic problem is solved as follows:

The program RCFORCE is used to obtain the in-plane linear forced dynamic response for a certain frequency and amplitude of excitation at the top without damping. The output from this program is then used as an input to program RCLINDY2, which solves the in-plane linear dynamic eigenproblem. The above method for zero current can be also used for arbitrary currents.

o Finally, Programs ORTHOG and ORTHOG1 can be used to evaluate the orthogonality of out-of-plane and in-plane natural modes. This provides a check of the accuracy of numerical results.

A flowchart of the above procedures can be found in Table 1 below.

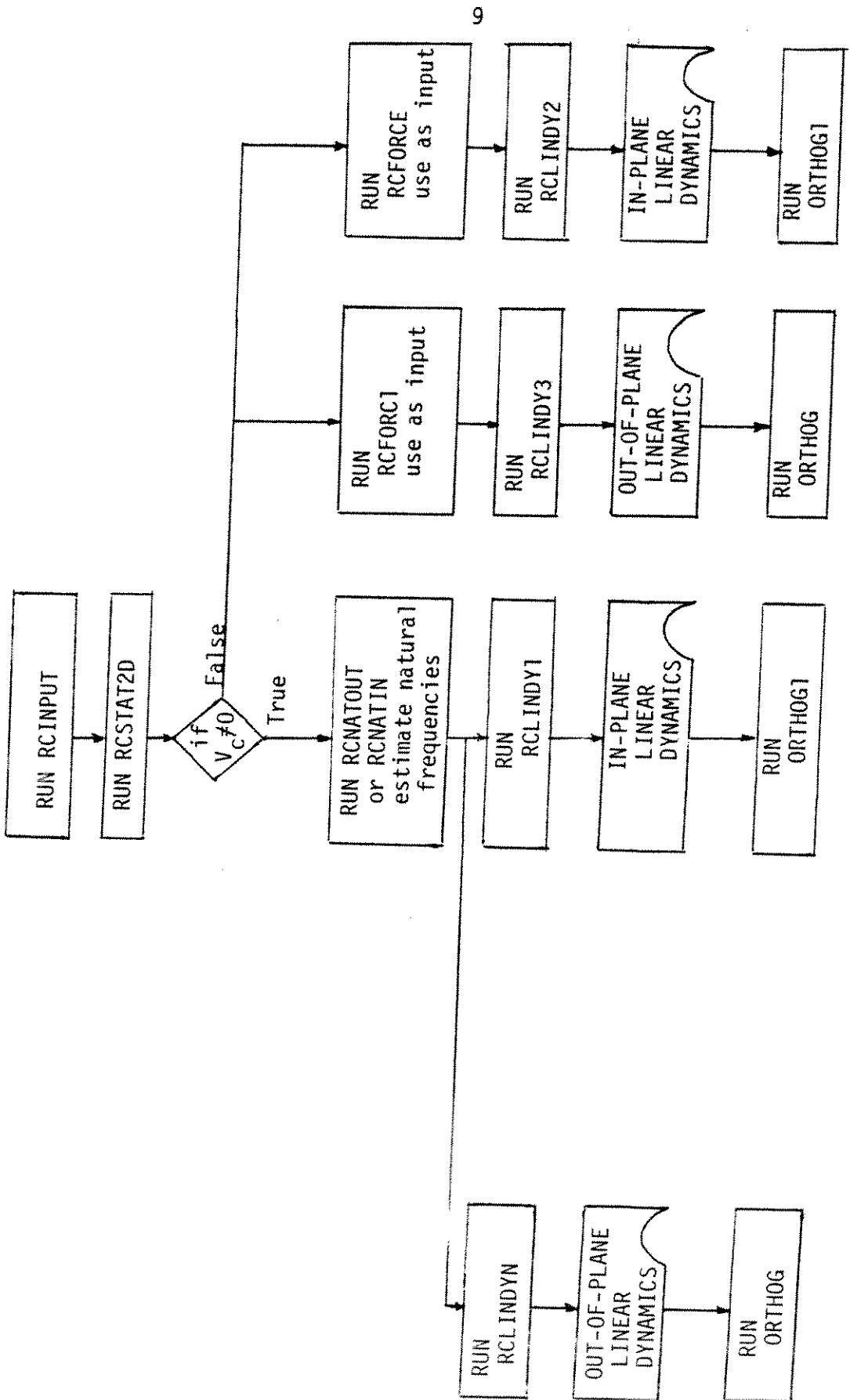


Table 1. Flowchart for the Solution of the Three-dimensional Linear Eigenproblem of a Compliant Riser with a Planar Static Configuration

Chapter II

II.1 Input-Output Manual for Out-of-Plane Natural Frequency and Mode Number Estimation Program, RCNATOUT

This program is written in FORTRAN 77 and allows the interactive estimation of the out-of-plane natural frequencies and the corresponding mode numbers of compliant risers using an asymptotic method described in [4].

Before executing the program the user must make sure that

- o Devices five and six correspond to input/output from the terminal.
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data preprocessing program RCINPUT. This file has a logical record length of 80 characters.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.
- o Device eleven must be associated with an output file with logical record length of 80 characters. This file contains the converged natural frequencies and mode numbers as well as the characteristic equation values at a series of frequency points. These natural frequency estimates can be used as inputs for a run of RCLINDYN.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The user is informed about the mean current velocity in the x direction and the maximum non-dimensional static effective tension. If the number of static division points is less than four

or greater than the value of MNP (Maximum Number of Allowable Points) the program halts execution. If the mean velocity is zero, the program halts execution because the asymptotic theory is no longer valid.

Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop (see [3]).

The user is then asked to input the upper limit of the frequency range and frequency spacing (in rad/s) for which he wishes to evaluate the characteristic equation for the eigenfrequencies. The default lower limit of the frequency range is 0.01 rad/s. The maximum permissible number of frequency points is also displayed. The frequency spacing should be such, so that the frequency points are less than the maximum allowable points. Then the user is asked if he wants to change the value of

$$\nu = (EI^{nn}(0)/TL^2)^{1/2}$$

(for definition of variables see [4]) to be used to determine the approximate natural frequencies for the out-of-plane problem. If he decides to change ν , he is asked the new value for ν .

Next the program evaluates the characteristic equation at all points in the frequency range. Then the natural frequencies in the specified range are determined. This is done by checking the change of sign in the characteristic equation. Finally, the program prints the following results in the file connected with device 11:

- o Number of natural frequencies in the specified frequency range.
- o Mode number and corresponding natural frequency estimate.
- o Two columns of real numbers, the first of which is the frequency (in rad/s) and the second, the corresponding value of the characteristic determinant at this frequency.

II.2 Sample Run, Input and Output

II.2.1 Linear Current Excitation

II.2.1.1 Interactive Session

INPUT

Device 5 = TERMINAL

Device 8 = RCJOA01 DATA A

Device 10 = RCCL2DRR DATA A

OUTPUT

Device 6 = TERMINAL

Device 11 = FREQOUT DATA A

```

R; T=0.01/0.01 20:53:35
rcrato1 rcjao1 rcc12dr freqout
FI 8 DISK RCJOAO1 DATA A ( RECFM FB LRECL 80 BLKSIZE 800
FI 10 DISK RCCL2DR DATA A ( RECFM FB LRECL 132 BLKSIZE 1320
FI 11 DISK FREQUET DATA A ( RECFM F LRECL 80
GLOBAL TXTLIB VFORTLIB CMSLIB NAG1 NAG2
LOAD RGNATOUT ( START
EXECUTION BEGINS...
MNP=151
2-D STATIC SOLUTION FROM DEVICE 10
NP = 76
MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S , VM =0.129000D+01
2-D STATIC SOLUTION SUCCESSFULLY READ
MAXIMUM STATIC EFFECTIVE TENSION/WA*L = 0.3089D+02

INPUT
THE MAXIMUM FREQUENCY AND FREQUENCY SPACING (IN RAD/S)
FOR WHICH YOU WISH TO EVALUATE THE FREQUENCY FUNCTION
MAXIMUM PERMISSIBLE NUMBER OF FREQUENCY POINTS IS = 300
?
2...01
THE VALUE OF NU TO BE USED IS = 0.1979D-01
IF YOU WANT TO CHANGE NU INPUT !
?
2
R; T=0.47/0.60 20:54:14
cp spool console stop close

```

II.2.1.2 RCJ0A01 DATA A

CHINESE LANTERN FROM JOAO, APRIL 1985

9 0 883920D+02 0.292000D+01
 0.820000D+03 0.115434D-01 0.000000D+00 0.345000D+07
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.660000D+04 0.237400D-01 - .330000D+03
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.577500D+04 0.237400D-01 - .330000D+03
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.495000D+04 0.237400D-01 - .330000D+03
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.412500D+04 0.237400D-01 - .330000D+03
 0.683920D+02 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.320000D+04 0.237400D-01 0.000000D+00
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.412500D+04 0.237400D-01 0.330000D+03
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.495000D+04 0.237400D-01 0.330000D+03
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.577500D+04 0.237400D-01 0.330000D+03
 0.250000D+01 0.499300D+02 0.404700D+02 0.824400D+02 0.292000D+01
 0.310000D+00 0.930000D+00 0.267000D+09 0.660000D+04 0.237400D-01 0.330000D+03
 0.503200D+02 0.200000D+00 0.244000D+05 - .122000D+04 0.116400D+07 - .582000D+05
 0.000000D+00 0.493200D+00 0.780600D-01
 0.503200D+02 0.200000D+00 0.152500D+05 - .122000D+04 0.727500D+06 - .582000D+05
 0.000000D+00 0.493200D+00 0.213500D+05 0.122000D+04 0.873000D+06 0.582000D+05
 0.503200D+02 0.200000D+00 0.780600D-01
 0.000000D+00 0.493200D+00 0.152500D+05 0.122000D+04 0.727500D+06 0.582000D+05
 0.503200D+02 0.200000D+00 0.780600D-01
 0.000000D+00 0.493200D+00 0.183000D+05 - .122000D+04 0.873000D+06 0.000000D+00
 0.503200D+02 0.200000D+00 0.780600D-01
 0.000000D+00 0.493200D+00 0.122000D+04 0.101850D+07 - .582000D+05
 0.503200D+02 0.200000D+00 0.780600D-01
 0.000000D+00 0.493200D+00 0.213500D+05 0.122000D+04 0.101850D+07 0.582000D+05
 0.000000D+00 0.493200D+00 0.780600D-01
 0.503200D+02 0.200000D+00 0.244000D+05 0.122000D+04 0.116400D+07 0.582000D+05

II.2.1.3 RCCL2DRR DATA A

76 0.129000D+01
 0.000000D+00 0.170039D+02 - .236894D+02 0. 157080D+01 0. 0000000D+00 0. 0000000D+00 0. 0000000D+00 0. 129964D+03 0. 1030000D+01
 0.986636D-03 0.188299D+02 - .765616D+02 0. 149202D+01 0. 393819D-04 0. 985601D-03 0. 170885D-07 0. 131639D-03 0. 103062D+01
 0.246464D-02 0.210989D+02 - .224204D+02 - .674058D+02 0. 138574D+01 0. 235333D-03 0. 244990D-02 0. 456129D-07 0. 134181D+03 0. 103154D+01
 0.394265D-02 0.228898D+02 - .180743D+02 - .591772D+02 0. 129231D+01 0. 576161D-03 0. 388758D-02 0. 770377D-07 0. 136044D+03 0. 103244D+01
 0.558056D-02 0.244206D+02 - .158539D+02 - .510930D+02 0. 120214D+01 0. 109848D-02 0. 543945D-02 0. 114492D-06 0. 137652D+03 0. 103342D+01
 0.721847D-02 0.255745D+02 - .138288D+02 - .440177D+02 0. 112438D+01 0. 174897D-02 0. 694227D-02 0. 154072D-06 0. 138881D+03 0. 103436D+01
 0.885637D-02 0.264380D+02 - .120108D+01 - .378565D+02 0. 105454D+01 0. 250525D-02 0. 839433D-02 0. 195248D-06 0. 139817D+03 0. 103527D+01
 0.111404D-01 0.272851D+02 - .981897D+01 - .306508D+02 0. 9379541D+00 0. 370698D-02 0. 103367D-01 0. 254557D-06 0. 140761D+03 0. 103650D+01
 0.134244D-01 0.278406D+02 - .799468D+01 - .246888D+02 0. 9166621D+00 0. 504106D-02 0. 121903D-01 0. 315413D-01 0. 141093D+03 0. 103766D+01
 0.157084D-01 0.282016D+02 - .649122D+01 - .198747D+02 0. 865528D+00 0. 647792D-02 0. 139655D-01 0. 377282D-06 0. 141859D+03 0. 103878D+01
 0.178993D-01 0.284264D+02 - .530522D+01 - .161128D+02 0. 826647D+00 0. 793103D-02 0. 156050D-01 0. 437246D-06 0. 142166D+03 0. 103981D+01
 0.200910D-01 0.285712D+02 - .432907D+01 - .130396D+02 0. 794827D+00 0. 944131D-02 0. 171920D-01 0. 497603D-06 0. 142391D+03 0. 104080D+01
 0.222809D-01 0.286649D+02 - .352706D+01 - .105327D+02 0. 769100D+00 0. 109965D-01 0. 187352D-01 0. 558212D-06 0. 142561D+03 0. 104177D+01
 0.244717D-01 0.287225D+02 - .286846D+01 - .849489D+01 0. 748339D+00 0. 125870D-01 0. 202419D-01 0. 6189881D-06 0. 142694D+03 0. 104272D+01
 0.280184D-01 0.287699D+02 - .204078D+01 - .198747D+02 0. 865528D+00 0. 647792D-02 0. 139655D-01 0. 377282D-06 0. 141859D+03 0. 103878D+01
 0.315651D-01 0.287848D+02 - .143952D+01 - .407588D+01 0. 705264D+00 0. 249412D-01 0. 816201D-06 0. 144286D+03 0. 104421D+01
 0.351118D-01 0.287829D+02 - .101126D+01 - .261779D+01 0. 693520D+00 0. 206150D-01 0. 272233D-01 0. 816201D-06 0. 144286D+03 0. 104421D+01
 0.413491D-01 0.287618D+02 - .543656D+00 - .103372D+01 0. 682654D+00 0. 254366D-01 0. 311804D-01 0. 1088837D-05 0. 143219D+03 0. 104959D+01
 0.475865D-01 0.287331D+02 - .2948851D+00 - .173341D+00 0. 679189D+00 0. 302851D-01 0. 351046D-01 0. 126171D-05 0. 143447D+03 0. 105206D+01
 0.547166D-01 0.286983D+02 - .139066D+00 0. 341387D+00 0. 680014D+00 0. 358327D-01 0. 395840D-01 0. 145963D-05 0. 143636D+03 0. 105487D+01
 0.618467D-01 0.2866333D+02 - .624267D-01 0. 611680D+00 0. 683352D+00 0. 413699D-01 0. 440763D-01 0. 165731D-05 0. 143825D+03 0. 105770D+01
 0.778366D-01 0.285857D+02 - .512844D-02 0. 864156D+00 0. 6957191D+00 0. 537081D-01 0. 542456D-01 0. 209976D-05 0. 144255D+03 0. 106409D+01
 0.954915D-01 0.285016D+02 0. 188175D-02 - .284039D-02 0. 961707D+00 0. 7112081D+00 0. 671701D-01 0. 656742D-01 0. 258692D-05 0. 144742D+03 0. 107126D+01
 0.114743D+00 0.284120D+02 - .137457D+01 0. 103436D+01 0. 731328D+00 0. 8162236D-01 0. 783888D-01 0. 3111651D-05 0. 145287D+03 0. 107926D+01
 0.135516D+00 0.2833178D+02 - .6252201D-02 0. 108843D+01 0. 7533368D+00 0. 969321D-01 0. 924290D-01 0. 3686608D-05 0. 145895D+03 0. 108808D+01
 0.157726D+00 0.282205D+02 - .624267D-01 0. 117302D+01 0. 778454D+00 0. 112943D+00 0. 107823D+00 0. 4292305D-06 0. 146566D+03 0. 109775D+01
 0.181288D+00 0.281202D+02 - .6932341D-02 0. 126509D+01 0. 807162D+00 0. 129482D+00 0. 124603D+00 0. 493469D-05 0. 147306D+03 0. 110829D+01
 0.206107D+00 0.280237D+02 - .751215D-02 0. 137457D+01 0. 8398986D+00 0. 146354D+00 0. 142806D+00 0. 566282D-05 0. 148117D+03 0. 111973D+01
 0.2320287D+00 0.279282D+02 - .834319D-02 0. 1500035D+01 0. 877213D+00 0. 1633338D+00 0. 162464D+00 0. 631080D-05 0. 149003D+03 0. 113208D+01
 0.2591123D+00 0.278379D+02 - .919832D-02 0. 164522D+01 0. 919704D+00 0. 180182D+00 0. 183610D+00 0. 703955D-05 0. 149969D+03 0. 114537D+01
 0.287110D+00 0.277556D+02 - .101068D-01 0. 181034D+01 0. 968024D+00 0. 196609D+00 0. 206267D+00 0. 779158D-05 0. 151019D+03 0. 115961D+01
 0.315938D+00 0.276841D+02 - .109844D-01 0. 199626D+01 0. 102285D+01 0. 212304D+00 0. 230444D+00 0. 856405D-05 0. 152155D+03 0. 117480D+01
 0.345492D+00 0.276265D+02 - .117289D-01 0. 220170D+01 0. 108486D+01 0. 226922D+00 0. 256125D+00 0. 935414D-05 0. 153380D+03 0. 119094D+01
 0.375655D+00 0.275853D+02 - .121762D-01 0. 242265D+01 0. 115458D+01 0. 240084D+00 0. 282626D+00 0. 101591D-04 0. 154694D+03 0. 120799D+01
 0.406309D+00 0.275624D+02 - .121133D-01 0. 265113D+01 0. 123235D+01 0. 251392D+00 0. 311745D+00 0. 109762D-04 0. 156094D+03 0. 122589D+01
 0.437333D+00 0.275580D+02 - .113052D-01 0. 287445D+01 0. 131811D+01 0. 260442D+00 0. 341410D+00 0. 118027D-04 0. 159751D-03 0. 1244453D+01
 0.468605D+00 0.275705D+02 - .955897D-02 0. 3075333D+01 0. 141122D+01 0. 266856D+00 0. 372006D+00 0. 126366D-04 0. 159112D+03 0. 126375D+01
 0.500000D+00 0.275954D+02 - .6830061D-02 0. 323401D+01 0. 151040D+01 0. 270313D+00 0. 403198D+00 0. 134731D-04 0. 160695D+03 0. 128335D+01
 0.531395D+00 0.276259D+02 - .327400D-02 0. 333203D+01 0. 161365D+01 0. 270596D+00 0. 434578D+00 0. 143111D-04 0. 162293D-03 0. 130307D+01
 0.562667D+00 0.276614D+02 - .775562D-02 0. 335631D+01 0. 175052D-01 0. 267626D+00 0. 465694D+00 0. 151467D-04 0. 163683D+03 0. 132262D+01
 0.593691D+00 0.276841D+02 - .129901D-01 0. 256775D+01 0. 131811D+01 0. 261480D+00 0. 496089D+00 0. 159770D-04 0. 165450D+03 0. 134172D+01
 0.624345D+00 0.277807D+02 0. 826120D-02 0. 317961D+01 0. 192143D+01 0. 252391D+00 0. 525352D+00 0. 167992D-04 0. 166982D+03 0. 136011D+01
 0.654508D+00 0.2787879D+02 0. 108104D-01 0. 300241D+01 0. 194257D+01 0. 235997D+00 0. 553151D+00 0. 176107D-04 0. 168469D-03 0. 137757D+01
 0.684062D+00 0.2800787D+02 0. 123485D-01 0. 279189D+01 0. 210048D+01 0. 226876D+00 0. 579257D+00 0. 184090D-04 0. 169902D+03 0. 139398D+01
 0.712899D+00 0.281673D+02 0. 129804D+02 0. 217762D+01 0. 211351D+00 0. 603539D+00 0. 221143D-04 0. 171917D-04 0. 140923D+01
 0.740877D+00 0.283549D+02 0. 129430D+02 0. 234500D+01 0. 182191D+01 0. 261480D+00 0. 625953D+00 0. 199563D-04 0. 172581D+03 0. 142332D+01
 0.767913D+00 0.285664D+02 0. 124729D-01 0. 213454D+01 0. 230704D+01 0. 177051D+00 0. 207002D-04 0. 173820D+03 0. 143624D+01
 0.793893D+00 0.287966D+02 0. 116809D-01 0. 201479D+01 0. 240711D+00 0. 159096D+00 0. 665290D+00 0. 214205D-04 0. 174988D+03 0. 144804D+01
 0.842274D+00 0.292904D+02 0. 963498D-02 0. 162204D+01 0. 244597D+01 0. 132391D+00 0. 697808D+00 0. 227786D-04 0. 177106D+03 0. 146847D+01
 0.864484D+00 0.295431D+02 0. 1002123D-01 0. 148606D+01 0. 248044D+01 0. 105995D+00 0. 711742D+00 0. 234102D-04 0. 178055D+03 0. 147722D+01
 0.885257D+00 0.297930D+02 0. 5373268D-02 0. 139297D+01 0. 894087D-01 0. 251033D+01 0. 140059D+00 0. 724247D+00 0. 178929D-04 0. 148508D+01
 0.904508D+00 0.3003545D+02 0. 191615D-02 0. 128280D+01 0. 253612D+01 0. 737199D-01 0. 7355405D+00 0. 245626D-04 0. 179729D+03 0. 149209D+01
 0.922164D+00 0.302664D+02 0. 366355D-02 0. 115457D+01 0. 255779D+01 0. 590927D-01 0. 745292D+00 0. 250772D-04 0. 180454D+03 0. 1498330D+01

0. 938153D+00 0. 304825D+02 0. 598605D-01 0. 889706D+00 0. 257459D+01 0. 456745D-01 0. 753988D+00 0. 255467D-04 0. 181104D+03 0. 150377D+01
 0. 945283D+00 0. 305810D+02 0. 137684D+00 0. 614876D+01 0. 258007D+00 0. 258281D+01 0. 396488D-01 0. 757800D+00 0. 257571D-04 0. 181393D+03 0. 150616D+01
 0. 952414D+00 0. 306803D+02 0. 296225D+00 0. 877526D-01 0. 258281D+01 0. 336069D-01 0. 761587D+00 0. 259683D-04 0. 181682D+03 0. 150854D+01
 0. 958651D+00 0. 307662D+02 0. 567726D+00 -808400D+00 0. 258089D+01 0. 283196D-01 0. 764896D+00 0. 261535D-04 0. 181933D+03 0. 151062D+01
 0. 964888D+00 0. 308447D+02 0. 108411D+01 -249525D+01 0. 257116D+01 0. 230502D-01 0. 7682334D+00 0. 263393D-04 0. 182178D+03 0. 151272D+01
 0. 968435D+00 0. 308780D+02 0. 156743D+01 -407742D+01 0. 255965D+01 0. 200752D-01 0. 770165D+00 0. 264451D-04 0. 182308D+03 0. 151393D+01
 0. 971982D+00 0. 308915D+02 0. 225704D+01 -616800D+01 0. 254167D+01 0. 171287D-01 0. 772139D+00 0. 265510D-04 0. 182420D+03 0. 151517D+01
 0. 975528D+00 0. 308653D+02 0. 322160D+01 -898149D+01 0. 251509D+01 0. 142264D-01 0. 774178D+00 0. 266568D-04 0. 182496D+03 0. 151645D+01
 0. 977719D+00 0. 308134D+02 0. 399900D+01 -112915D+02 0. 249297D+01 0. 124654D-01 0. 775481D+00 0. 267221D-04 0. 182509D+03 0. 151722D+01
 0. 979910D+00 0. 307161D+02 0. 495518D+01 -141546D+02 0. 246521D+01 0. 107373D-01 0. 776828D+00 0. 267873D-04 0. 182479D+03 0. 151812D+01
 0. 982101D+00 0. 305493D+02 0. 613047D+01 -176978D+02 0. 243046D+01 0. 905203D-02 0. 778228D+00 0. 268522D-04 0. 182382D+03 0. 151900D+01
 0. 984292D+00 0. 302762D+02 0. 757202D+01 -220760D+02 0. 238706D+01 0. 742251D-02 0. 779692D+00 0. 269166D-04 0. 182182D+03 0. 151992D+01
 0. 9856005D+00 0. 299538D+02 0. 891875D+01 -262003D+02 0. 234581D+01 0. 619823D-02 0. 780890D+00 0. 269664D-04 0. 181919D+03 0. 152067D+01
 0. 987718D+00 0. 294953D+02 0. 104874D+02 -310524D+02 0. 229688D+01 0. 502909D-02 0. 782142D+00 0. 270157D-04 0. 181523D+03 0. 152146D+01
 0. 989431D+00 0. 288495D+02 0. 123040D+02 -367497D+02 0. 223894D+01 0. 392837D-02 0. 783454D+00 0. 270640D-04 0. 180943D+03 0. 152228D+01
 0. 991144D+00 0. 279474D+02 0. 143899D+02 -434209D+02 0. 217042D+01 0. 291264D-02 0. 784833D+00 0. 271110D-04 0. 180110D+03 0. 152315D+01
 0. 992782D+00 0. 267614D+02 0. 166460D+02 -508361D+02 0. 209337D+01 0. 203985D-02 0. 786219D+00 0. 271543D-04 0. 178993D+03 0. 152402D+01
 0. 994419D+00 0. 251556D+02 0. 191493D+02 -593855D+02 0. 200327D+01 0. 128564D-02 0. 787672D+00 0. 271954D-04 0. 177460D+03 0. 152493D+01
 0. 996057D+00 0. 230037D+02 0. 218611D+02 -6911762D+02 0. 189816D+01 0. 676474D-03 0. 789192D+00 0. 272335D-04 0. 175384D+03 0. 152589D+01
 0. 997535D+00 0. 204712D+02 0. 244166D+02 -791404D+02 0. 178868D+01 0. 276975D-03 0. 790614D+00 0. 272646D-04 0. 172922D+03 0. 152678D+01
 0. 999013D+00 0. 172574D+02 0. 269606D+02 -901925D+02 0. 166367D+01 0. 464375D-04 0. 792073D+00 0. 272915D-04 0. 169782D+03 0. 152770D+01
 0. 100000D+01 0. 146780D+02 0. 285696D+02 -981546D+02 0. 157080D+01 0. 214481D-18 0. 793058D+00 0. 273068D-04 0. 167251D+03 0. 152832D+01

II.2.1.4 FREQOUT DATA A

CHINESE LANTERN FROM JOAO, APRIL 1985

ESTIMATES OF OUT-OF PLANE NATURAL FREQUENCIES FOR A 2-D STATIC CONFIGURATION

NUMBER OF NATURAL FREQUENCIES IN SPECIFIED RANGE : 6

MODE = 1 NATURAL FREQUENCY BETWEEN
0.330000D+00 AND 0.340000D+00 RAD/SEC

MODE = 2 NATURAL FREQUENCY BETWEEN
0.650000D+00 AND 0.660000D+00 RAD/SEC

MODE = 3 NATURAL FREQUENCY BETWEEN
0.980000D+00 AND 0.990000D+00 RAD/SEC

MODE = 4 NATURAL FREQUENCY BETWEEN
0.131000D+01 AND 0.132000D+01 RAD/SEC

MODE = 5 NATURAL FREQUENCY BETWEEN
0.164000D+01 AND 0.165000D+01 RAD/SEC

MODE = 6 NATURAL FREQUENCY BETWEEN
0.197000D+01 AND 0.198000D+01 RAD/SEC

FREQUENCY (RAD/SEC)	EQUATION	VALUE
0.100000D-01	0.950482D-01	
0.200000D-01	0.189238D+00	
0.300000D-01	0.281719D+00	
0.400000D-01	0.371655D+00	
0.500000D-01	0.458234D+00	
0.600000D-01	0.540675D+00	
0.700000D-01	0.618231D+00	
0.800000D-01	0.690203D+00	
0.900000D-01	0.755939D+00	
0.100000D+00	0.814844D+00	
0.110000D+00	0.866338D+00	
0.120000D+00	0.910098D+00	
0.130000D+00	0.945584D+00	
0.140000D+00	0.972520D+00	
0.150000D+00	0.990663D+00	
0.160000D+00	0.999846D+00	
0.170000D+00	0.999984D+00	
0.180000D+00	0.991072D+00	
0.190000D+00	0.973190D+00	
0.200000D+00	0.946496D+00	
0.210000D+00	0.911228D+00	
0.220000D+00	0.867703D+00	
0.230000D+00	0.816309D+00	
0.240000D+00	0.757509D+00	
0.250000D+00	0.691832D+00	

0.260000D+00
 0.270000D+00
 0.280000D+00
 0.290000D+00
 0.300000D+00
 0.310000D+00
 0.320000D+00
 0.330000D+00
 0.340000D+00
 0.350000D+00
 0.360000D+00
 0.370000D+00
 0.380000D+00
 0.390000D+00
 0.400000D+00
 0.410000D+00
 0.420000D+00
 0.430000D+00
 0.440000D+00
 0.450000D+00
 0.460000D+00
 0.470000D+00
 0.480000D+00
 0.490000D+00
 0.500000D+00
 0.510000D+00
 0.520000D+00
 0.530000D+00
 0.540000D+00
 0.550000D+00
 0.560000D+00
 0.570000D+00
 0.580000D+00
 0.590000D+00
 0.600000D+00
 0.610000D+00
 0.620000D+00
 0.630000D+00
 0.640000D+00
 0.650000D+00
 0.660000D+00
 0.670000D+00
 0.680000D+00
 0.690000D+00
 0.700000D+00
 0.710000D+00
 0.720000D+00
 0.730000D+00
 0.740000D+00
 0.750000D+00
 0.760000D+00
 0.770000D+00
 0.780000D+00
 0.790000D+00
 0.800000D+00
 0.810000D+00

VM/SP CONVERSATIONAL MONITOR SYSTEM

0.820000D+00	0.102481D+01
0.830000D+00	0.102524D+01
0.840000D+00	0.101636D+01
0.850000D+00	0.998259D+00
0.860000D+00	0.971072D+00
0.870000D+00	0.935036D+00
0.880000D+00	0.890468D+00
0.890000D+00	0.837751D+00
0.900000D+00	0.777372D+00
0.910000D+00	0.709848D+00
0.920000D+00	0.635786D+00
0.930000D+00	0.555850D+00
0.940000D+00	0.470754D+00
0.950000D+00	0.381262D+00
0.960000D+00	0.288180D+00
0.970000D+00	0.192345D+00
0.980000D+00	0.946224D-01
0.990000D+00	-4.10434D-02
0.100000D+01	-1029442D+00
0.101000D+01	-2009940D+00
0.102000D+01	-297370D+00
0.103000D+01	-391192D+00
0.104000D+01	-481607D+00
0.105000D+01	-567789D+00
0.106000D+01	-648947D+00
0.107000D+01	-724338D+00
0.108000D+01	-793269D+00
0.109000D+01	-855102D+00
0.110000D+01	-909264D+00
0.111000D+01	-955252D+00
0.112000D+01	-992633D+00
0.113000D+01	-102105D+01
0.114000D+01	-1040240D+01
0.115000D+01	-1050000D+01
0.116000D+01	-105023D+00
0.117000D+01	-104091D+01
0.118000D+01	-102210D+01
0.119000D+01	-993973D+00
0.120000D+01	-956751D+00
0.121000D+01	-910760D+00
0.122000D+01	-856401D+00
0.123000D+01	-794152D+00
0.124000D+01	-724565D+00
0.125000D+01	-648256D+00
0.126000D+01	-565907D+00
0.127000D+01	-478255D+00
0.128000D+01	-386085D+00
0.129000D+01	-290227D+00
0.130000D+01	-191544D+00
0.131000D+01	-909289D-01
0.132000D+01	0.107084D-01
0.133000D+01	0.112446D+00
0.134000D+01	0.213358D+00
0.135000D+01	0.312526D+00
0.136000D+01	0.409044D+00
0.137000D+01	0.502029D+00

0. 138000D+01
 0. 139000D+01
 0. 140000D+01
 0. 141000D+01
 0. 142000D+01
 0. 143000D+01
 0. 144000D+01
 0. 145000D+01
 0. 146000D+01
 0. 147000D+01
 0. 148000D+01
 0. 149000D+01
 0. 150000D+01
 0. 151000D+01
 0. 152000D+01
 0. 153000D+01
 0. 154000D+01
 0. 155000D+01
 0. 156000D+01
 0. 157000D+01
 0. 158000D+01
 0. 159000D+01
 0. 160000D+01
 0. 161000D+01
 0. 162000D+01
 0. 163000D+01
 0. 164000D+01
 0. 165000D+01
 0. 166000D+01
 0. 167000D+01
 0. 168000D+01
 0. 169000D+01
 0. 170000D+01
 0. 175000D+01
 0. 176000D+01
 0. 177000D+01
 0. 178000D+01
 0. 179000D+01
 0. 180000D+01
 0. 181000D+01
 0. 182000D+01
 0. 183000D+01
 0. 184000D+01
 0. 185000D+01
 0. 186000D+01
 0. 187000D+01
 0. 188000D+01
 0. 189000D+01
 0. 190000D+01
 0. 191000D+01
 0. 192000D+01
 0. 193000D+01

0.194000D+01 -.391007D+00
0.195000D+01 -.286790D+00
0.196000D+01 -.179668D+00
0.197000D+01 -.706141D-01
0.198000D+01 0.393802D-01
0.199000D+01 0.149311D+00
0.200000D+01 0.258171D+00
0.201000D+01 0.364961D+00

Chapter III

III.1 Input-Output Manual for the Out-of-Plane Linear Eigenproblem Solution Program, RCLINDYN

This program is written in FORTRAN 77 and allows the interactive solution of the out-of-plane linear eigenproblem for a compliant riser with a planar static configuration without torsion, as described in [4].

Before executing the program the user must make sure that

- o Devices five and six correspond to input/output from the terminal
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data preprocessing program RCINPUT. This file has a logical record length of 80 characters.
- o Device nine must be associated with an output data file with logical record length of 132 characters which upon completion of the program contains a complete list of the inputs, initial approximation and the converged solution of the problem.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.
- o Device eleven must be associated with an output file with logical record length of 117 characters. This file contains the converged solution of the problem in a concise form appropriate for use in the preparation of illustrations and/or as an input to RCLINDYN from device twelve in a subsequent execution of this program.

- o Device twelve may be associated with an input file containing an initial approximate solution of an out-of-plane linear dynamic problem, resulting from a previous execution of RCLINDYN. This option can be used if greater accuracy is needed for the solution of the problem. Device twelve may be associated with a dummy file name if this option is not used.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The user is informed about the mean current velocity in the X direction and the maximum non-dimensional static effective tension. If the number of static discretization points is less than four or greater than the value of MNP (maximum number of allowable points) the program halts execution. If the mean velocity is zero, the program halts execution because the asymptotic theory is no longer valid.

Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop, see [3].

Next the user is asked whether he wants the initial approximation to be calculated using asymptotics (input IEXIST = 0) or whether he wants to use the option to read an initial approximation from device twelve created by a previous run of the same program (input IEXIST = 1).

Case 1 (IEXIST = 0)

The user is asked whether he wants to change the value of $v = (EI^{nn}(0)/T \cdot L^2)^{1/2}$ to be used to determine the natural frequency for the out-of-plane solution. If he wants to change it, he is asked the new value for v .

Next the user is asked to supply an initial guess for the natural frequency in rad/s. The information for the approximate natural frequencies is obtained by running RCNATOUT program which provides an estimate of the natural frequencies and their corresponding mode-number.

The program calculates the approximate natural frequency using the asymptotic theory and displays its value. Then the user is given the alternatives:

- o Input 1 if he wants to redo the calculation. This can be used in case the converged frequency doesn't correspond to the mode which the user wants to evaluate.
- o Input 2 if he chooses to stop the program.
- o Input any other integer to continue with the program.

Next the user is asked to input the mode number corresponding to the approximate natural frequency, as determined from RCNATOUT program.

The initial asymptotic approximation is evaluated and the orthonormalizing constant used for r and ψ , as well as the fourth boundary condition at $s = 0$ are printed. The user is then asked if he wants to review the initial approximation by displaying it in his terminal. If he chooses to do so, he should input 1. Subsequently the approximate solution is printed in the file associated with the device nine. The following quantities are printed:

- o Number of discretization points NP
- o Natural frequency (rad/s)
- o Order of point (starting from the lower end)
- o Non-dimensional arc length
- o Non-dimensional shear force in the n direction, Q_1^n

- o Non-dimensional component of rate of rotation about ξ , Ω_1^ξ
- o Non-dimensional component of rate of rotation about ζ , Ω_1^ζ
- o Euler angle θ_1
- o Euler angle β_1 or ψ_1
- o Non-dimensional displacement r
- o Non-dimensional natural frequency

Subsequently the program prints the maximum of the absolute value of the initial ($\epsilon=0$) non-dimensional component of $\dot{\Omega}_1$ in the ξ direction. This number is used to estimate a reasonable tolerance for convergence of iterations. The user is requested to input a tolerance as a fraction of this number; e.g., 0.01 for 1% tolerance or better for all quantities of interest. Typically Ω_1^ξ will be determined with an accuracy a little better than the above fraction and all other variables will be even more accurate.

The user is given the opportunity to stop the program at this point if he wishes. He should input 0 if he wants to stop.

If he chooses to continue he is asked to input the initial increment of the continuation parameter $\delta\epsilon$ (condition $0 < \delta\epsilon \leq 1$). If no continuation is required, then use $\delta\epsilon=1$. If continuation is required, then $\delta\epsilon=0.1$ will usually suffice. For large changes between initial and final problem a smaller value of $\delta\epsilon$, e.g., 0.05 or 0.025 might be necessary.

Case 2 (IEXIST = 1)

In this case an initial approximation to the solution for $\epsilon=1$ is available from a previous run of RCLINDYN. The program reads the information from the previous run concerning the number of division points, NP, the mode number and the corresponding natural frequency. If the number of division points is greater than MNP, the program stops execution.

Then the user has the option to review the initial approximation by displaying it in his terminal. The assumed boundary condition for the curvature at the lower end of the riser is also printed.

Subsequently, the program prints the maximum of the absolute values of the initial component of $\vec{\Omega}_1$ in the $\vec{\xi}$ direction. This number is used to estimate a reasonable tolerance for the convergence of iterations.

The user is requested to input a tolerance as a fraction of this number; e.g., 0.01 for 1% tolerance or better for all quantities of interest.

Typically, Ω_1^{ξ} will be determined with an accuracy a little better than the above fraction and all other variables will be even more accurate. Subsequently the user is given the opportunity to halt the execution of the program.

After the initial approximation is fully defined the program enters NAG subroutine D02RAF which provides the iterative numerical solution to the problem. The manual of this subroutine is included in Chapter X. The input value of IFAIL is 111, indicating that the soft failure option with error message and monitoring information printing has been selected. Error messages are directed to device six.

Once the execution of D02RAF is completed, the current value of IFAIL is printed on the terminal. For an explanation of this parameter, see NAG manual for D02RAF, Chapter X. Only if IFAIL=0 or IFAIL=4, the execution of the program continues with printout of the riser characteristics and the final solution of the problem. Otherwise, the program stops. When IFAIL \neq 0, the user should consult the manual quoted above for interpretation of the results and a possible course of action.

If IFAIL=0 or 4, the riser characteristics and the final solution of the problem are printed in the file associated with device nine. For the riser characteristics the same format is used as in program RCSTAT2D. This is described in [3] (pp. 23-25).

Following, the non-dimensional out-of-plane solution is written in the file associated with device nine. The mode number, natural frequency and final number of points NP at which the solution is available is printed first. Then the following data is printed for I=1 to NP:

$s, Q_1^n, \Omega_1^\zeta, \Omega_1^\xi, \theta_1, \beta, r, \Sigma$

with

FORMAT (8(1X, D10.4)).

Next, the maximum (non-dimensional) estimated errors of $Q_1^n, \Omega_1^\zeta, \Omega_1^\xi, \theta_1, \beta, r, \Sigma$ provided by the NAG subroutine D02RAF are printed. Subsequently, the following data is written in the file associated with device eleven:

MODE, NP, SIGMAD: Mode number, number of points at which converged solution is available and dimensional natural frequency (in rad/s) with
FORMAT (1X, I2, 1X, I3, 1X, D10.4) and

For I=1 to NP the non-dimensional

$s, Q_1^n, \Omega_1^\zeta, \Omega_1^\xi, \theta_1, \beta, r, \Sigma$, and dimensional current velocity $V_c(s)$

with

FORMAT (9(1X, D12.6)).

The data written in the file associated with device eleven, due to their simple form, can be used very easily as input to plotting programs. In addition, this data is useful in subsequent runs of RCLINDYN, if such an option is selected.

III.2 Sample Run, Input and Output

III.2.1 Linear Current Excitation

III.2.1.1 Interactive Session

INPUT

```
Device 5 = TERMINAL  
Device 8 = RCJOA01 DATA A*  
Device 10 = RCCL2DRR DATA A*  
Device 12 = DUMMY DATA A
```

OUTPUT

```
Device 6 = TERMINAL  
Device 9 = JOA01 DATA A  
Device 11 = JOA01A DATA A
```

* Files included in Chapter II.

```

R: T=0 01/0.01 15:48:42
RC1 INDYD RCJOAO1 JOAO1 RCL12DRP JOAO1A DUMMY
FI 8 DISK RCJOAO1 DATA A ( RECFM FB LRECL 80 BLKSIZE 800
FI 9 DISK JOAO1 DATA A ( RECFM FB LRECL 132 BLKSIZE 1320
FI 10 DISK RCCL2DRP DATA A ( RECFM FB LRECL 132 BLKSIZE 1320
FI 11 DISK JOAO1A DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
FI 12 DISK DUMMY DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
GLOBAL TXLIB VFORTLIB CMSLIB NAG1 NAG2
LOAD RCLINDYN ( START
EXECUTION BEGINS...
NRP=151

2-D STATIC SOLUTION FROM DEVICE 10
NP = 76
MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S , VM =0.129000D+01
2-D STATIC SOLUTION SUCCESSFULLY READ
MAXIMUM STATIC TENSION/WALL = 0.3089D+02
DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS
IF YES INPUT 1 , IF NO INPUT 0
?
O
INPUT 1 IF YOU WISH TO READ INITIAL APPROXIMATION.
CREATED BY A PREVIOUS RUN OF THIS PROGRAM.
ELSE INPUT 0 (AN INITIAL APPROXIMATION WILL BE CALCULATED LOCALLY)
?
O
THE VALUE OF NU TO BE USED IS = 0.1979D-01
IF YOU WANT TO CHANGE NU INPUT 1
?
2
INPUT INITIAL GUESS FOR SIGMA IN RAD/SEC
?
.
33
.
1 FAIL FOR APPROXIMATE EVALUATION OF SIGMA IS
THE FUNCTION IS F = 0.9714D-16
THE CONVERGED SIGMA IS 0.3300D+00
INPUT 1 IF YOU WANT TO REDO THE CALCULATION
INPUT 2 IF YOU WANT TO STOP
?
3
INPUT MODE NUMBER CORRESPONDING TO ABOVE APPROXIMATE SIGMA
?
1
THE ORTHONORMALIZING CONSTANT FOR THE APPROXIMATE
SOLUTION IS SQRT(A) = 0.9334D+01
THE BOUNDARY CONDITION AT S=0 OMEGAXI(0) = 0.2747D+03
DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION
IF YES INPUT 1
?
2
MAXIMUM ABSOLUTE VALUE OF N-O OMEGA_X1 IS =0.274676D+03
THIS NUMBER CAN BE USED TO ESTIMATE
A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS
INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE
E.G. INPUT 0.01 FOR 1% ACCURACY
?
.
01
IF YOU WANT TO STOP INPUT 0
?
1
INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS
?
```

IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.0D
 IF CONTINUATION IS REQUIRED THEN 0.0D < DELEPS < 1.0D
 RECOMMENDATION :
 USUALLY DELEPS = 0.10D WILL SUFFICE
 FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM
 A SMALLER VALUE OF DELEPS, E.G. 0.05D0 OR 0.025D0 MIGHT BE
 NECESSARY
 ?

DO2RAF MONITORING INFORMATION

MONITORING NEWTON ITERATION

NUMBER OF POINTS IN CURRENT MESH = 76

CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00D+00

ITERATION NUMBER 0 RESIDUAL = 7.32D-01

SQUARED NORM OF CORRECTION = 8.48D+02

SQUARED NORM OF GRADIENT = 1.16D+03

SCALAR PRODUCT OF CORRECTION AND GRADIENT = 5.36D-01

ITERATION NUMBER 1 RESIDUAL = 3.74D-04

CONTINUATION PARAMETER EPSILON = 2.00D-01 DELEPS = 2.00D-01

MONITORING NEWTON ITERATION

NUMBER OF POINTS IN CURRENT MESH = 76

CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00D+00

ITERATION NUMBER 0 RESIDUAL = 2.37D-01

CONTINUATION PARAMETER EPSILON = 6.00D-01 DELEPS = 4.00D-01

MONITORING NEWTON ITERATION

NUMBER OF POINTS IN CURRENT MESH = 76

CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00D+00

ITERATION NUMBER 0 RESIDUAL = 1.71D+00

SQUARED NORM OF CORRECTION = 1.17D+02

SQUARED NORM OF GRADIENT = 1.53D+04

SCALAR PRODUCT OF CORRECTION AND GRADIENT = 2.94D+00

ITERATION NUMBER 1 RESIDUAL = 9.87D-05

CONTINUATION PARAMETER EPSILON = 1.00D+00 DELEPS = 8.00D-01

MONITORING NEWTON ITERATION

NUMBER OF POINTS IN CURRENT MESH = 76

CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 2.75D-04

ITERATION NUMBER 0 RESIDUAL = 1.19D+01

SQUARED NORM OF CORRECTION = 5.90D+03

SQUARED NORM OF GRADIENT = 4.60D+05

SCALAR PRODUCT OF CORRECTION AND GRADIENT = 1.42D+02

ITERATION NUMBER 1 RESIDUAL = 3.55D-03

SQUARED NORM OF CORRECTION = 2.06D-03

SQUARED NORM OF GRADIENT = 2.56D-06

SCALAR PRODUCT OF CORRECTION AND GRADIENT = 1.26D-05

ITERATION NUMBER 2 RESIDUAL = 1.19D-06

MESH SELECTION

NUMBER OF NEW POINTS 8

MONITORING NEWTON ITERATION

NUMBER OF POINTS IN CURRENT MESH = 84

CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 2.75D-04

```

ITERATION NUMBER      0      RESIDUAL = 2.68D-01
SQUARED NORM OF CORRECTION = 4.31D-01
SQUARED NORM OF GRADIENT  = 4.36D-01
SCALAR PRODUCT OF CORRECTION AND GRADIENT = 7.20D-02
ITERATION NUMBER      1      RESIDUAL = 2.78D-08

MESH SELECTION
NUMBER OF NEW POINTS 1

CORRECTION NUMBER      0      ESTIMATED MAXIMUM ERROR = 5.10D-01
ESTIMATED ERROR BY COMPONENTS
1.20D-02 1.54D-02 5.10D-01 1.86D-02 1.28D-02 2.01D-03 2.71D-03
I FAIL = 0
R : T=3.24/3.60 15:50:22
cp spool console stop close

```

III.2.1.2 JOAOI DATA A

INITIAL CONDITION FOR EPS=0, AND NP = 76 POINTS, NATURAL FREQUENCY = 0.3300D+00 RAD/SEC

	ARC	SHEAR ETA	OMEGA ZETA	OMEGA XI	THE TAU	BETA	R	SIGMA
1	0.000000D+00	-5911456D+01	0.290842D+01	0.274676D+03	0.00000D+00	0.00000D+00	-120808D-22	0.375852D+01
2	0.986636D-03	562518D+01	0.258510D+01	0.261317D+03	0.264361D+00	-727465D-02	-131498D-03	0.375852D+01
3	0.246464D-02	522202D+01	0.217428D+01	0.242505D+03	0.636515D+00	.504215D-01	.800767D-03	0.375852D+01
4	0.394265D-02	484785D+01	0.183905D+01	0.225041D+03	0.981874D+00	-122102D+00	-199985D-02	0.375852D+01
5	0.558056D-02	446456D+01	0.153973D+01	0.207144D+03	0.133561D+01	-222806D+00	-390176D-02	0.375852D+01
6	0.721847D-02	-411117D+01	0.130208D+01	0.190661D+03	0.166121D+01	-336249D+00	-635972D-02	0.375852D+01
7	0.8855637D-02	-378686D+01	0.111378D+01	0.175480D+03	0.196089D+01	-454947D+00	-932943D-02	0.375852D+01
8	0.111404D-01	-337648D+01	0.914234D+01	0.156290D+00	0.233936D+01	-6191172D+00	-142487D-01	0.375852D+01
9	0.134244D-01	-301080D+01	0.770695D+00	0.1391177D+03	0.267641D+01	-774472D+00	-199872D-01	0.375852D+01
10	0.157084D-01	-268495D+01	0.667681D+00	0.123915D+03	0.297653D+01	-915844D+00	-264465D-01	0.375852D+01
11	0.172993D-01	-240579D+01	0.596512D+00	0.110826D+03	0.3233342D+01	-103634D+01	-332544D-01	0.375852D+01
12	0.200901D-01	-215587D+01	0.544954D+00	0.990971D+02	0.346313D+01	-114162D+01	-405945D-01	0.375852D+01
13	0.222809D-01	-193211D+01	0.507679D+00	0.885835D+02	0.366851D+01	-123226D+01	-484107D-01	0.375852D+01
14	0.244717D-01	-173178D+01	0.480791D+00	0.791584D+02	0.487520D+01	-130933D+01	-566525D-01	0.375852D+01
15	0.280184D-01	-145094D+01	0.451883D+00	0.659203D+02	0.410865D+01	-140921D+01	-707835D-01	0.375852D+01
16	0.315651D-01	-107596D+01	0.498117D+00	0.548205D+02	0.432219D+01	-148230D+01	-857459D-01	0.375852D+01
17	0.351118D-01	-902127D+00	0.488881D+00	0.455082D+02	0.449962D+01	-153268D+01	-101400D+00	0.375852D+01
18	0.413491D-01	-662560D+00	0.481592D+00	0.326040D+02	0.474118D+01	-158168D+01	-130260D+00	0.375852D+01
19	0.475865D-01	-487569D+00	0.479794D+00	0.230909D+02	0.491338D+01	-159662D+01	-160401D+00	0.375852D+01
20	0.547166D-01	-344461D+00	0.498585D+00	0.152061D+02	0.504831D+01	-159009D+01	-195948D+00	0.375852D+01
21	0.618467D-01	-211860D+00	0.560485D+00	0.958321D+01	0.513557D+01	-156904D+01	-232278D+00	0.375852D+01
22	0.778360D-01	-997381D-01	0.561413D+00	0.200728D+00	0.522113D+01	-149889D+01	-315236D+00	0.375852D+01
23	0.954915D-01	-371997D-01	0.673697D+00	-215482D+01	0.521517D+01	-140387D+01	-407477D+00	0.375852D+01
24	0.114743D+00	-190348D-01	0.841041D+00	-459415D+01	0.514862D+01	-128975D+01	-507316D+00	0.375852D+01
25	0.135516D+00	-134361D-01	0.839202D+00	-626807D+01	0.503479D+01	-116006D+01	-613146D+00	0.375852D+01
26	0.157726D+00	-111642D-01	0.8365521D+00	-764339D+01	0.487983D+01	-101703D+01	-723304D+00	0.375852D+01
27	0.181288D+00	-101008D-01	0.832874D+00	-891562D+01	0.468450D+01	-860113D+00	-836040D+00	0.375852D+01
28	0.206107D+00	-940049D-02	0.828086D+00	-101494D+02	0.444773D+01	-690090D+00	-949430D+00	0.375852D+01
29	0.232087D+00	-875526D-02	0.821931D+00	-113535D+02	0.416822D+01	-508002D+00	-106142D+01	0.375852D+01
30	0.259230D+00	-806244D-02	0.814153D+00	-125149D+02	0.384536D+01	-315826D+00	-116982D+01	0.375852D+01
31	0.287110D+00	-729162D-02	0.804474D+00	-136127D+02	0.347948D+01	-116501D+00	-127239D+01	0.375852D+01
32	0.315938D+00	-643718D-02	0.792607D+00	-146239D+02	0.307218D+01	0.857134D-01	-136689D+01	0.375852D+01
33	0.345492D+00	-550276D-02	0.775286D+00	-155255D+02	0.262631D+01	0.284998D+00	-145117D+01	0.375852D+01
34	0.375655D+00	-449635D-02	0.761298D+00	-162962D+02	0.214660D+01	0.473832D+00	-152320D+01	0.375852D+01
35	0.406409D+00	-342886D-02	0.741532D+00	-161916D+02	0.163652D+01	0.643050D+00	-116501D+00	0.375852D+01
36	0.437333D+00	-231347D-02	0.719048D+00	-173722D+02	0.110417D+01	0.782240D+00	-162377D+01	0.375852D+01
37	0.468605D+00	-116511D-02	0.694130D+00	-176502D+02	0.556083D+00	0.880703D+00	-164976D+01	0.375852D+01
38	0.500000D+00	-240888D-12	0.667325D+00	-177436D+02	-481011D-12	0.928920D+00	-152320D+01	0.375852D+01
39	0.532395D+00	0.116511D-02	0.639433D+00	-176502D+02	-556083D+00	0.920347D+00	-106142D+01	0.375852D+01
40	0.562667D+00	0.231347D-02	0.611430D+00	-173722D+02	-110417D+01	0.852988D+00	-164976D+01	0.375852D+01
41	0.5936910D+00	0.342886D-02	0.584344D+00	-161916D+02	-163652D+01	0.730200D+00	-162377D+01	0.375852D+01
42	0.624345D+00	0.449635D+00	-559103D+00	-176502D+02	-1.214660D+01	0.560293D+00	-152320D+01	0.375852D+01
43	0.654508D+00	0.550275D-02	0.536402D+00	-155255D+02	-2.626231D+01	0.354985D+00	-145117D+01	0.375852D+01
44	0.684062D+00	0.643718D-02	0.516643D+00	-146239D+02	-307218D+01	0.127280D+00	-136689D+01	0.375852D+01
45	0.712890D+00	0.729161D-02	0.499939D+00	-136127D+02	-110417D+01	0.836040D+00	-136689D+01	0.375852D+01
46	0.740877D+00	0.806239D-02	0.486180D+00	-125149D+02	-3.47948D+01	-1.10471D+00	-127239D+01	0.375852D+01
47	0.767913D+00	0.875508D-02	0.475104D+00	-113536D+02	-4.16823D+01	-1.347871D+00	-116982D+01	0.375852D+01
48	0.793893D+00	0.939982D-02	0.466373D+00	-10494D+02	-4.44773D+01	-4.44773D+01	-106142D+01	0.375852D+01
49	0.818712D+00	0.100984D-02	0.459636D+00	-891586D+01	-4.68451D+01	-991161D+00	-949430D+00	0.375852D+01
50	0.842274D+00	0.111564D-01	0.454555D+00	-764418D+01	-4.79895D+01	-1.17144D+01	-723304D+00	0.375852D+01
51	0.864484D+00	0.134121D-01	0.450871D+00	-627049D+01	-5.03484D+01	-1.33237D+01	-613145D+00	0.375852D+01
52	0.885257D+00	0.189562D-01	0.448373D+00	-4.60108D+01	-5.14876D+01	-1.47524D+01	-507313D+00	0.375852D+01
53	0.904508D+00	0.369405D-01	0.357533D+00	-2.17322D+01	-5.2161607D+01	-1.60092D+01	-407472D+00	0.375852D+01

54 0.922164D+00 0.989970D-01 0.297937D+00 0.196257D+00 0.522201D+01 - .170743D+01 -.315219D+00 0.375852D+01
 55 0.938153D+00 0.210195D+00 0.299245D+00 0.948271D+01 -.513756D+01 -.232240D+00 0.375852D+01
 56 0.945283D+00 0.341715D+00 0.257395D+00 0.150620D+00 0.150620D+01 -.505116D+01 -.178740D+01 -.195894D+00 0.375852D+01
 57 0.952414D+00 0.483654D+00 0.258010D+00 0.228854D+02 -.491744D+01 -.182355D+01 -.160318D+00 0.375852D+01
 58 0.958651D+00 0.657181D+00 0.256859D+00 0.323216D+02 -.474677D+01 -.181133D+01 -.130149D+00 0.375852D+01
 59 0.964888D+00 0.894742D+00 0.250244D+00 0.451205D+02 -.450730D+01 -.176271D+01 -.101249D+00 0.375852D+01
 60 0.968435D+00 0.106714D+01 0.241180D+00 0.543576D+02 -.433135D+01 -.171057D+01 -.855641D-01 0.375852D+01
 61 0.971982D+00 0.143904D+01 0.149838D+00 0.653675D+02 -.411959D+01 -.163341D+01 -.705648D-01 0.375852D+01
 62 0.975528D+00 0.171748D+01 0.164792D+00 0.784942D+02 -.386521D+01 -.152617D+01 -.563940D-01 0.375852D+01
 63 0.977719D+00 0.191616D+01 0.136365D+00 0.878426D+02 -.368317D+01 -.144225D+01 -.48110D-01 0.375852D+01
 64 0.979910D+00 0.213807D+01 0.964465D-01 0.982704D+02 -.347949D+01 -.134248D+01 -.402701D-01 0.375852D+01
 65 0.982101D+00 0.238594D+01 0.405596D-01 0.109904D+03 -.325167D+01 -.122533D+01 -.328915D-01 0.375852D+01
 66 0.984292D+00 0.266279D+01 -.374713D-01 0.122885D+03 -.299691D+01 -.108981D+01 -.260410D-01 0.375852D+01
 67 0.986005D+00 0.290162D+01 -.119012D+00 0.134073D+03 -.277696D+01 -.970959D+00 -.210929D-01 0.375852D+01
 68 0.987718D+00 0.316202D+01 -.224509D+00 0.146264D+03 -.253700D+01 -.841461D+00 -.165385D-01 0.375852D+01
 69 0.989431D+00 0.344595D+01 -.360793D+00 0.159548D+03 -.227524D+01 -.702872D+00 -.124136D-01 0.375852D+01
 70 0.991144D+00 0.375553D+01 -.536559D+00 0.174024D+03 -.198971D+01 -.558027D+00 -.875714D-02 0.375852D+01
 71 0.992782D+00 0.407769D+01 -.751580D+00 0.189081D+03 -.169250D+01 -.417858D+00 -.573805D-02 0.375852D+01
 72 0.994419D+00 0.442741D+01 -.102458D+01 0.205418D+03 -.136978D+01 -.2822307D+00 -.322792D-02 0.375852D+01
 73 0.996057D+00 0.480515D+01 -.137071D+01 0.223167D+03 -.101897D+01 -.159917D+00 -.126757D-02 0.375852D+01
 74 0.997535D+00 0.517855D+01 -.176025D+01 0.240486D+03 -.676490D+00 -.705756D-01 -.114690D-04 0.375852D+01
 75 0.999013D+00 0.557834D+01 -.223946D+01 0.259141D+03 -.307438D+00 -.134755D-01 0.719049D-03 0.375852D+01
 76 0.100000D+01 0.586243D+01 -.261761D+01 0.272394D+03 -.451795D-01 -.761245D-15 0.894142D-03 0.375852D+01

9

= NUMBER OF RISER SEGMENTS

0.883920D+02 = UNSTRETCHED RISER LENGTH IN M

0.2920000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.8200000D+03 = INNER FLUID DENSITY IN KG/M³0.102500D+04 = SALT WATER DENSITY IN KG/M³0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.0000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.797324D+04 = MAXIMUM STATIC TENSION IN N

0.129000D+01 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

DATA PER RISER SEGMENTS FOR NSEG = 9 SEGMENTS

DIMENSIONAL QUANTITIES IN THE SYSTEM
 RLENG DXI PIETA AO WEIGHT MASS AMA1 AMAET1 AMAZ1 TMAX1 TMAX2
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03
 0.6839D+02 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 1003D+03
 EA E1ETA E1ETAS EIXI GIP S DE TA JZI AJZ1 TJZ1
 0 . 2670D+09 0 . 6600D+04 - . 3300D+03 0 . 2440D+05 - . 1220D+04 0 . 1164D+07 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 5775D+04 - . 3300D+03 0 . 2135D+05 - . 1220D+04 0 . 1019D+07 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4950D+04 - . 3300D+03 0 . 1830D+05 - . 1220D+04 0 . 8730D+06 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4125D+04 - . 3300D+03 0 . 1525D+05 - . 1220D+04 0 . 7275D+06 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 3300D+04 0 . 0000D+00 0 . 1220D+05 0 . 0000D+00 0 . 5820D+06 0 . 0000D+00 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4125D+04 0 . 3300D+03 0 . 1525D+05 0 . 1220D+04 0 . 7275D+06 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4950D+04 0 . 3300D+03 0 . 1830D+05 0 . 1220D+04 0 . 8730D+06 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 5775D+04 0 . 3300D+03 0 . 2135D+05 0 . 1220D+04 0 . 1019D+07 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 6600D+04 0 . 3300D+03 0 . 2440D+05 0 . 1220D+04 0 . 1164D+07 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00

 M O D E N U M B E R = 1
 N A T U R A L F R E Q U E N C Y = 0 . 3200D+00 RAD/S

ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = 84 POINTS

S	QETA	OMEGA ZETA	OMEGA X	THETA	BETA	R	SIGMA
0 . 0000D+00	- . 5055D+01	0 . 2797D+01	0 . 2205D+03	0 . 0000D+00	0 . 0000D+00	0 . 0000D+00	0 . 3645D+01
0 . 9866D-03	- . 4928D+01	0 . 2544D+01	0 . 2188D+03	0 . 2165D+00	- . 5542D-02	- . 1068D-03	0 . 3645D+01
0 . 1726D-02	- . 4825D+01	0 . 2369D+01	0 . 2168D+03	0 . 3768D+00	- . 1987D-01	- . 3260D-03	0 . 3645D+01
0 . 2465D-02	- . 4718D+01	0 . 2207D+01	0 . 2140D+03	0 . 5344D+00	- . 4151D-01	- . 6627D-03	0 . 3645D+01
0 . 3204D-02	- . 4607D+01	0 . 2057D+01	0 . 2108D+03	0 . 6887D+00	- . 6935D-01	- . 1115D-02	0 . 3645D+01
0 . 3943D-02	- . 4494D+01	0 . 1919D+01	0 . 2072D+03	0 . 8393D+00	- . 1023D+00	- . 1679D-02	0 . 3645D+01
0 . 4762D-02	- . 4366D+01	0 . 1779D+01	0 . 2028D+03	0 . 1001D+01	- . 1438D+00	- . 2433D-02	0 . 3645D+01
0 . 5584D-02	- . 4237D+01	0 . 1651D+01	0 . 1981D+03	0 . 1158D+01	- . 1892D+00	- . 3317D-02	0 . 3645D+01
0 . 7218D-02	- . 3979D+01	0 . 1432D+01	0 . 1880D+03	0 . 1456D+01	- . 2877D+00	- . 5459D-02	0 . 3645D+01
0 . 8856D-02	- . 3725D+01	0 . 1254D+01	0 . 1775D+03	0 . 1733D+01	- . 3917D+00	- . 8071D-02	0 . 3645D+01
0 . 1114D-01	- . 3383D+01	0 . 1061D+01	0 . 1628D+03	0 . 2086D+01	- . 5369D+00	- . 1243D-01	0 . 3645D+01
0 . 1342D-01	- . 3062D+01	0 . 924D+00	0 . 1486D+03	0 . 2404D+01	- . 6753D+00	- . 1756D-01	0 . 3645D+01
0 . 1571D-01	- . 2765D+01	0 . 8182D+00	0 . 1352D+03	0 . 1580D+01	- . 8022D+00	- . 2338D-01	0 . 3645D+01
0 . 1790D-01	- . 2503D+01	0 . 7475D+00	0 . 1232D+03	0 . 2940D+01	- . 9110D+00	- . 2955D-01	0 . 3645D+01
0 . 2009D-01	- . 2262D+01	0 . 6968D+00	0 . 1121D+03	0 . 3167D+01	- . 1007D+01	- . 3624D-01	0 . 3645D+01
0 . 2228D-01	- . 2042D+01	0 . 6609D+00	0 . 1019D+03	0 . 3375D+01	- . 1089D+01	- . 4340D-01	0 . 3645D+01
0 . 2447D-01	- . 1841D+01	0 . 6366D+00	0 . 9262D+02	0 . 3565D+01	- . 1160D+01	- . 5101D-01	0 . 3645D+01
0 . 2802D-01	- . 1552D+01	0 . 6116D+00	0 . 7937D+02	0 . 3839D+01	- . 1252D+01	- . 6414D-01	0 . 3645D+01
0 . 3157D-01	- . 1305D+01	0 . 6015D+00	0 . 6733D+02	0 . 4076D+01	- . 1320D+01	- . 7817D-01	0 . 3645D+01
0 . 3511D-01	- . 1095D+01	0 . 60008D+00	0 . 5636D+02	0 . 4280D+01	- . 1367D+01	- . 9299D-01	0 . 3645D+01
0 . 4135D-01	- . 8012D+00	0 . 6109D+00	0 . 4097D+02	0 . 4568D+01	- . 1413D+01	- . 1206D+00	0 . 3645D+01
0 . 4759D-01	- . 5845D+00	0 . 6281D+00	0 . 2958D+02	0 . 4782D+01	- . 1427D+01	- . 1497D+00	0 . 3645D+01
0 . 5472D-01	- . 4045D+00	0 . 6513D+00	0 . 2017D+02	0 . 4960D+01	- . 1419D+01	- . 1845D+00	0 . 3645D+01
0 . 6185D-01	- . 2769D+00	0 . 6784D+00	0 . 1313D+02	0 . 5084D+01	- . 1397D+01	- . 2203D+00	0 . 3645D+01
0 . 7784D-01	- . 1098D+00	0 . 7465D+00	0 . 2960D+01	0 . 5228D+01	- . 1325D+01	- . 3027D+00	0 . 3645D+01
0 . 9549D-01	- . 3647D-01	0 . 8377D+00	- . 2250D+01	0 . 5255D+01	- . 1226D+01	- . 3953D+00	0 . 3645D+01

0. 1147D+00	- 1003D-01	0 8936D+00	- 51200D+01	0 55207D+01	- 1109D+01	- 4960D+00	0 3645D+01	
0. 1355D+00	- 2992D-02	0 8916D+00	- 6744D+01	0 5106D+01	- 9768D+00	- 6031D+00	0 3645D+01	
0. 1577D+00	- 9404D-03	0 8888D+00	- 8027D+01	0 4965D+01	- 8307D+00	- 7149D+00	0 3645D+01	
0. 1813D+00	0 4724D-03	0 8850D+00	- 9264D+01	0 4783D+01	- 6699D+00	- 8298D+00	0 3645D+01	
0. 2061D+00	0 1999D-02	0 8800D+00	- 1048D+02	0 4557D+01	- 4951D+00	- 9457D+00	0 3645D+01	
0. 2321D+00	0 3807D-02	0 8737D+00	- 1165D+02	0 4284D+01	- 3075D+00	- 1061D+01	0 3645D+01	
0. 2591D+00	0 5920D-02	0 8658D+00	- 1273D+02	0 3963D+01	- 1089D+00	- 1172D+01	0 3645D+01	
0. 2871D+00	0 8339D-02	0 8560D+00	- 1368D+02	0 3594D+01	0 9743D-01	- 1278D+01	0 3645D+01	
0. 3159D+00	0 1102D-01	0 8442D+00	- 1447D+02	0 3177D+01	0 3071D+00	- 1375D+01	0 3645D+01	
0. 3455D+00	0 1389D-01	0 8301D+00	- 1505D+02	0 2715D+01	0 5139D+00	- 1462D+01	0 3645D+01	
0. 3757D+00	0 1678D-01	0 8139D+00	- 1541D+02	0 2212D+01	0 7096D+00	- 1537D+01	0 3645D+01	
0. 4063D+00	0 1945D-01	0 7954D+00	- 1554D+02	0 1676D+01	0 8845D+00	- 1596D+01	0 3645D+01	
0. 4373D+00	0 2163D-01	0 7751D+00	- 1548D+02	0 1112D+01	0 1027D+01	- 1640D+01	0 3645D+01	
0. 4686D+00	0 2301D-01	0 7532D+00	- 1528D+02	0 5309D+00	0 1127D+01	- 1665D+01	0 3645D+01	
0. 5000D+00	0 2344D-01	0 7302D+00	- 1504D+02	0 5915D-01	0 1173D+01	- 1677D+01	0 3645D+01	
0. 5314D+00	0 2300D-01	0 7066D+00	- 1483D+02	0 6483D+00	0 1158D+01	- 1662D+01	0 3645D+01	
0. 5627D+00	0 2205D-01	0 6830D+00	- 1468D+02	0 1227D+01	0 1082D+01	- 1632D+01	0 3645D+01	
0. 5937D+00	0 2101D-01	0 6599D+00	- 1456D+02	0 1785D+01	0 9474D+00	- 1585D+01	0 3645D+01	
0. 6243D+00	0 2017D-01	0 6378D+00	- 1440D+02	0 2315D+01	0 7641D+00	- 1523D+01	0 3645D+01	
0. 6545D+00	0 1966D-01	0 6175D+00	- 1411D+02	0 2806D+01	0 5450D+00	- 1445D+01	0 3645D+01	
0. 6841D+00	0 1947D-01	0 5993D+00	- 1363D+02	0 3252D+01	0 5544D+01	- 1356D+01	0 3645D+01	
0. 7129D+00	0 1948D-01	0 5836D+00	- 1293D+02	0 3649D+01	0 5542D-01	- 1256D+01	0 3645D+01	
0. 7409D+00	0 1954D-01	0 5705D+00	- 1202D+02	0 3994D+01	0 1906D+00	- 1149D+01	0 3645D+01	
0. 7679D+00	0 1957D-01	0 5598D+00	- 1095D+02	0 4287D+01	0 4257D+00	- 1038D+01	0 3645D+01	
0. 7939D+00	0 1949D-01	0 5514D+00	- 9754D+01	0 4528D+01	0 6443D+00	- 9230D+00	0 3645D+01	
0. 8187D+00	0 1936D-01	0 5450D+00	- 8494D+01	0 4720D+01	0 8436D+00	- 8083D+00	0 3645D+01	
0. 8423D+00	0 1913D-01	0 5402D+00	- 7211D+01	0 4668D+01	0 1022D+01	- 6953D+00	0 3645D+01	
0. 8645D+00	0 1957D-01	0 5367D+00	- 5898D+01	0 4975D+01	0 1180D+01	- 5866D+00	0 3645D+01	
0. 8853D+00	0 2449D-01	0 5344D+00	- 4321D+01	0 5044D+01	0 1319D+01	- 4819D+00	0 3645D+01	
0. 9045D+00	0 4574D-01	0 4399D+00	- 1827D+01	0 5068D+01	0 1439D+01	- 3846D+00	0 3645D+01	
0. 9222D+00	0 1067D+00	0 4450D+00	- 0 2527D+01	0 5030D+01	0 1539D+01	- 2954D+00	0 3645D+01	
0. 9382D+00	0 2517D+00	0 4065D+00	- 0 1130D+02	- 4894D+01	- 1614D+01	- 2161D+00	0 3645D+01	
0. 9453D+00	0 3655D+00	0 3920D+00	- 0 1750D+02	- 4782D+01	- 1637D+01	- 1816D+00	0 3645D+01	
0. 9524D+00	0 5297D+00	0 3789D+00	- 0 2599D+02	- 0 8758D+02	- 3530D+01	- 1464D+01	- 1481D+00	0 3645D+01
0. 9587D+00	0 7318D+00	0 3658D+00	- 0 3652D+02	- 0 4432D+01	- 1634D+01	- 1634D+00	0 3645D+01	
0. 9649D+00	0 1012D+01	0 4450D+00	- 0 2517D+00	- 0 5118D+02	- 4175D+01	- 1588D+01	- 1198D+00	0 3645D+01
0. 9684D+00	0 1217D+01	0 3307D+00	- 0 6189D+02	- 0 3993D+01	- 1539D+01	- 9306D+01	- 9306D+01	0 3645D+01
0. 9720D+00	0 1461D+01	0 3062D+00	- 0 7394D+02	- 0 3779D+01	- 2956D+01	- 1095D+01	- 7851D+01	0 3645D+01
0. 9755D+00	0 1752D+01	0 2680D+00	- 0 8758D+02	- 0 3530D+01	- 1468D+01	- 9715D+00	- 6473D+01	0 3645D+01
0. 9777D+00	0 1957D+01	0 2340D+00	- 0 2430D+00	- 0 2522D+01	- 1370D+01	- 1370D+01	- 5177D+01	0 3645D+01
0. 9799D+00	0 2184D+01	0 1879D+00	- 0 1082D+00	- 0 3167D+01	- 1293D+01	- 1293D+01	- 4423D+01	0 3645D+01
0. 9849D+00	0 3465D+01	0 3055D+00	- 0 1696D+03	- 0 2063D+01	- 4175D+00	- 1552D+01	- 1552D+01	0 3645D+01
0. 9882D+00	0 2435D+01	0 1250D+00	- 0 1202D+03	- 0 2956D+01	- 1539D+01	- 3708D+01	- 3708D+01	0 3645D+01
0. 9843D+00	0 4023D+01	0 7074D+00	- 0 1954D+03	- 0 2723D+01	- 1468D+01	- 1095D+01	- 3037D+01	0 3645D+01
0. 9936D+00	0 4163D+01	0 3921D-01	- 0 1336D+03	- 0 2522D+01	- 1468D+01	- 9715D+00	- 5758D+02	0 3645D+01
0. 9944D+00	0 4303D+01	0 4909D-01	- 0 1449D+03	- 0 2522D+01	- 1373D+01	- 3005D+00	- 1966D+01	0 3645D+01
0. 9984D+00	0 3198D+01	0 1618D+00	- 0 15659D+03	- 0 2303D+01	- 1219D+01	- 2406D+00	- 1552D+01	0 3645D+01
0. 9982D+00	0 4442D+01	0 1139D+01	- 0 2135D+03	- 0 2077D+03	- 1219D+01	- 2406D+00	- 1552D+01	0 3645D+01
0. 9928D+00	0 3746D+01	0 4880D+00	- 0 1827D+03	- 0 1799D+01	- 1799D+01	- 4888D+00	- 8477D+02	0 3645D+01
0. 9961D+00	0 4578D+01	0 1315D+01	- 0 2190D+03	- 0 1954D+03	- 1954D+03	- 1095D+01	- 3037D+01	0 3645D+01
0. 9968D+00	0 4698D+01	0 1490D+01	- 0 2235D+03	- 0 1730D+03	- 1730D+03	- 9036D+01	- 1186D+02	0 3645D+01
0. 9975D+00	0 4813D+01	0 1680D+01	- 0 2274D+03	- 0 1568D+03	- 1568D+03	- 5490D+01	- 7059D+03	0 3645D+01
0. 9983D+00	0 4922D+01	0 1887D+01	- 0 2307D+03	- 0 1913D+03	- 1913D+03	- 2705D+01	- 3477D+03	0 3645D+01

FILE: J0A01

VM/SP CONVERSATIONAL MONITOR SYSTEM

PAGE 005

DATA A1

0.9990D+00 0.5023D+01 - .2109D+01 0.2332D+03 - .2309D+00 -.8281D-02 -.1141D-03 0.3645D+01
0.9995D+00 0.5086D+01 -.2267D+01 0.2344D+03 -.1158D+00 -.1531D-02 -.2857D-04 0.3645D+01
0.1000D+01 0.5145D+01 -.2432D+01 0.2350D+03 0.0000D+00 0.0000D+00 -.4753D-19 0.3645D+01
MAXIMUM ESTIMATED ERROR BY COMPONENTS
0.9617D-02 0.1233D-01 0.4091D+00 0.1491D-01 0.1025D-01 0.1616D-02 0.2714D-02

III.2.1.3 JOAOTA DATA A

1 84 0.32000D+00
 0.00000D+00 - .505530D+01 0.279705D+01 0.220451D+03 0.000000D+00 0.000000D+00 0.000000D+00 0.364493D+01 0.1030000D+01
 0.98636D-03 - .492762D+01 0.254395D+01 0.218838D+03 0.216500D+00 0.554221D-02 - .106803D-03 0.364493D+01 0.103062D+01
 0.172564D-02 - .482520D+01 0.236914D+01 0.216751D+03 0.376765D+00 - .198727D-01 0.326015D-03 0.364493D+01 0.103108D+01
 0.246464D-02 - .471814D+01 0.220693D+01 0.214048D+03 0.534383D+00 - .415127D-01 0.662685D-03 0.364493D+01 0.103154D+01
 0.320364D-02 - .460742D+01 0.205696D+01 0.210834D+03 0.688722D+00 - .693535D-01 0.111463D-02 0.364493D+01 0.103199D+01
 0.394265D-02 - .449403D+01 0.191877D+01 0.207202D+03 0.839328D+00 - .102344D+00 - .167924D-02 0.364493D+01 0.103244D+01
 0.476160D-02 - .436287D+01 0.177867D+01 0.2027791D+03 0.100148D+01 - .143778D+00 - .243301D-02 0.364493D+01 0.103293D+01
 0.558056D-02 - .423732D+01 0.165150D+01 0.198066D+03 0.115842D+01 - .189220D+00 - .331745D-02 0.364493D+01 0.103342D+01
 0.721847D-02 - .397915D+01 0.143228D+01 0.188011D+03 0.145431D+01 - .802191D+00 - .233787D-01 0.364493D+01 0.103436D+01
 0.885637D-02 - .372477D+01 0.125421D+01 0.177519D+03 0.173315D+01 - .287664D+00 - .547668D-02 0.364493D+01 0.103436D+01
 0.111404D-01 - .338333D+01 0.106149D+01 0.162844D+03 0.20865D+01 - .391694D+00 - .807081D-02 0.364493D+01 0.103527D+01
 0.134244D-01 - .306241D+01 0.920385D+00 0.148613D+03 0.240403D+01 - .675324D+00 - .124325D-01 0.364493D+01 0.103650D+01
 0.157084D-01 - .276508D+01 0.818174D+00 0.135166D+03 0.269086D+01 - .802191D+00 - .175603D-01 0.364493D+01 0.103766D+01
 0.178993D-01 - .250265D+01 0.747510D+00 0.1226199D+01 0.294029D+01 - .910959D+00 - .233787D-01 0.364493D+01 0.103878D+01
 0.200901D-01 - .226199D+01 0.6976786D+00 0.112066D+03 0.316752D+01 - .100652D+01 - .295473D-01 0.364493D+01 0.103981D+01
 0.222809D-01 - .204197D+01 0.660904D+00 0.101899D+03 0.337493D+01 - .159888D+01 - .362378D-01 0.364493D+01 0.104080D+01
 0.244717D-01 - .184110D+01 0.636031D+00 0.926234D+02 0.356466D+01 - .108921D+01 - .434044D-01 0.364493D+01 0.104177D+01
 0.281848D-01 - .155224D+01 0.611623D+00 0.793695D+02 0.383893D+01 - .115988D+01 - .510060D-01 0.364493D+01 0.104272D+01
 0.315651D-01 - .130470D+01 0.601504D+00 0.673294D+02 0.407627D+01 - .125202D+01 - .641352D-01 0.364493D+01 0.104421D+01
 0.351118D-01 - .109479D+01 0.600832D+00 0.563600D+02 0.427973D+01 - .132001D+01 - .781716D-01 0.364493D+01 0.104567D+01
 0.413491D-01 - .801262D+00 0.610867D+00 0.409659D+02 0.456754D+01 - .136712D+01 - .929897D-01 0.364493D+01 0.104711D+01
 0.475865D-01 - .584529D+00 0.628124D+00 0.295764D+02 0.478221D+01 - .142654D+01 - .120581D+00 0.364493D+01 0.104959D+01
 0.547166D-01 - .404646D+00 0.651262D+00 0.201651D+02 0.496039D+01 - .148904D+01 - .149740D+00 0.364493D+01 0.105206D+01
 0.618467D-01 - .276906D+00 0.678052D+00 0.131285D+02 0.508385D+01 - .139704D+01 - .184473D+00 0.364493D+01 0.105487D+01
 0.728360D-01 - .109827D+00 0.746468D+00 0.295953D+01 0.522845D+01 - .132467D+01 - .220281D+00 0.364493D+01 0.105775D+01
 0.954915D-01 - .364650D-01 0.837656D+00 - .224979D+01 0.525523D+01 - .122618D+01 - .395272D+00 0.364493D+01 0.107126D+01
 0.114743D+00 - .100300D-01 0.893590D+00 - .511960D+01 0.520669D+01 - .110903D+01 - .495976D+00 0.364493D+01 0.107926D+01
 0.135516D+00 - .299230D-02 0.891600D+00 - .674374D+01 0.510643D+01 - .976822D+00 - .603093D+00 0.364493D+01 0.108808D+01
 0.157726D+00 - .940403D-03 0.888771D+00 - .802711D+01 0.496503D+01 - .1307502D+00 - .10652D+01 0.364493D+01 0.113208D+01
 0.181288D+00 0.472374D-03 0.884977D+00 - .833876D-02 0.3963338D+01 - .108950D+00 - .117201D+01 0.364493D+01 0.114537D+01
 0.206107D+00 0.199939D-02 0.880031D+00 - .104780D+02 0.455677D+01 - .926351D+01 - .669861D+00 - .829776D+00 0.364493D+01 0.110829D+01
 0.232087D+00 0.380747D-02 0.873707D+00 - .116459D+02 0.428422D+01 - .307502D+00 - .10652D+01 0.364493D+01 0.111973D+01
 0.259123D+00 0.591990D-02 0.865771D+00 - .127283D+02 0.3963338D+01 - .108950D+00 - .117201D+01 0.364493D+01 0.113208D+01
 0.287110D+00 0.833876D-02 0.855989D+00 - .136836D+02 0.3593383D+01 0.974325D-01 - .127776D+01 0.364493D+01 0.114537D+01
 0.315938D+00 0.216252D-01 0.775089D+00 - .154778D+02 0.111221D+01 0.102740D+01 - .127776D+01 0.364493D+01 0.115961D+01
 0.468605D+00 0.230142D-01 0.844157D+00 - .144701D+02 0.3171664D+01 0.307122D+00 - .137535D+01 0.364493D+01 0.117480D+01
 0.545492D+00 0.234398D-01 0.138891D-01 0.830131D+00 - .150526D+02 0.271461D+01 0.513889D+00 - .146241D+01 0.364493D+01 0.119094D+01
 0.375655D+00 0.167762D-01 0.2229972D-01 0.706623D+00 - .154091D+02 0.221221D+01 0.709648D+00 - .167264D+01 0.364493D+01 0.120799D+01
 0.406309D+00 0.194515D-01 0.795439D+00 - .155404D+02 0.167555D+01 0.884542D+00 - .15963D+01 0.364493D+01 0.122589D+01
 0.437333D+00 0.220505D-01 0.682994D+00 - .146830D+02 0.129297D+02 - .325240D+01 0.304319D+00 - .13559D+01 0.364493D+01 0.130307D+01
 0.593691D+00 0.210102D-01 0.659870D+00 - .145627D+02 0.570476D+00 - .120239D+02 0.554220D-01 0.125642D+01 0.364493D+01 0.132262D+01
 0.624345D+00 0.201692D-01 0.637844D+00 - .1573185D+00 0.152843D+02 0.530855D+00 0.947405D+00 - .158548D+01 0.364493D+01 0.134172D+01
 0.531395D+00 0.196641D-01 0.617500D+00 - .141086D+02 0.1403439D+00 - .159151D-01 0.513889D+00 - .166523D+01 0.364493D+01 0.136011D+01
 0.684062D+00 0.194727D-01 0.599321D+00 - .148332D+00 0.154500D+02 0.428664D+01 - .142566D+01 0.364493D+01 0.143624D+01
 0.712890D+00 0.194781D-01 0.583613D+00 - .146830D+02 0.129297D+02 - .325240D+01 0.304319D+00 - .13559D+01 0.364493D+01 0.137757D+01
 0.936391D+00 0.210102D-01 0.540194D+00 - .154778D+02 0.111221D+01 0.102740D+01 - .127776D+01 0.364493D+01 0.140923D+01
 0.767913D+00 0.195654D-01 0.559823D+00 - .143990D+02 0.121454D+01 0.764109D+00 - .166523D+01 0.364493D+01 0.142332D+01
 0.531395D+00 0.194903D-01 0.551428D+00 - .141086D+02 0.1403439D+00 - .159151D-01 0.644331D+00 - .167264D+01 0.364493D+01 0.144804D+01
 0.818712D+00 0.193019D-01 0.544993D+00 - .148332D+00 0.154500D+02 0.428664D+01 - .142566D+01 0.364493D+01 0.145876D+01
 0.842274D+00 0.191324D-01 0.540194D+00 - .152111D+01 0.486780D+01 - .102222D+01 - .695304D+00 0.364493D+01 0.146847D+01
 0.864484D+00 0.1913728D-01 0.536739D+00 - .1589813D+01 0.497549D+00 - .1118005D+01 0.585994D+00 0.364493D+01 0.147722D+01
 0.885257D+00 0.244860D-01 0.534411D+00 - .1432116D+01 0.504434D+01 - .131871D+01 - .481923D+00 0.364493D+01 0.148508D+01

0. 904508D+00	0. 457403D-01	0. 499219D+00	- 182708D+01	- . 506807D+01	- . 143897D+01	- . 384586D+00	0. 364493D+01	0. 149209D+01
0. 922164D+00	0. 106669D+00	0. 444998D+00	0. 252729D+01	- . 502990D+01	- . 153930D+01	- . 295441D+00	0. 364493D+01	0. 149830D+01
0. 938153D+00	0. 251748D+00	0. 406483D+00	0. 112981D+02	- . 489369D+01	- . 161372D+01	- . 216107D+00	0. 364493D+01	0. 150377D+01
0. 945283D+00	0. 365527D+00	0. 392025D+00	0. 175012D+02	- . 478231D+01	- . 163688D+01	- . 181612D+00	0. 364493D+01	0. 150616D+01
0. 952414D+00	0. 529722D+00	0. 378859D+00	0. 259854D+02	- . 462316D+01	- . 164606D+01	- . 148077D+00	0. 364493D+01	0. 150854D+01
0. 958651D+00	0. 731793D+00	0. 365834D+00	0. 365186D+02	- . 443191D+01	- . 163383D+01	- . 119839D+00	0. 364493D+01	0. 151062D+01
0. 964888D+00	0. 101222D+01	0. 347111D+00	0. 511785D+02	- . 417490D+01	- . 158795D+01	- . 929984D-01	0. 364493D+01	0. 151272D+01
0. 968435D+00	0. 121653D+01	0. 330735D+00	0. 618853D+02	- . 399254D+01	- . 153940D+01	- . 785134D-01	0. 364493D+01	0. 151393D+01
0. 971982D+00	0. 146132D+01	0. 306163D+00	0. 739359D+02	- . 377885D+01	- . 146806D+01	- . 647309D-01	0. 364493D+01	0. 151517D+01
0. 975528D+00	0. 175180D+01	0. 267976D+00	0. 875763D+02	- . 353035D+01	- . 1369500D+01	- . 517717D-01	0. 364493D+01	0. 151645D+01
0. 977719D+00	0. 195665D+01	0. 233961D+00	0. 973157D+02	- . 335727D+01	- . 129269D+01	- . 442263D-01	0. 364493D+01	0. 151727D+01
0. 979910D+00	0. 218361D+01	0. 187860D+00	0. 108176D+03	- . 316677D+01	- . 120159D+01	- . 370792D-01	0. 364493D+01	0. 151812D+01
0. 982101D+00	0. 243483D+01	0. 125046D+00	0. 120241D+03	- . 295640D+01	- . 109483D+01	- . 303712D-01	0. 364493D+01	0. 151900D+01
0. 984292D+00	0. 271187D+01	0. 392051D-01	0. 133567D+03	- . 272308D+01	- . 971472D+00	- . 241494D-01	0. 364493D+01	0. 151992D+01
0. 986005D+00	0. 294701D+01	0. 490914D-01	0. 144871D+03	- . 252234D+01	- . 863390D+00	- . 196567D-01	0. 364493D+01	0. 152067D+01
0. 987718D+00	0. 319847D+01	- . 161824D+00	0. 156911D+03	- . 230307D+01	- . 745715D+00	- . 155237D-01	0. 364493D+01	0. 152146D+01
0. 989431D+00	0. 346544D+01	- . 305501D+00	0. 169587D+03	- . 206277D+01	- . 619934D+00	- . 117844D-00	0. 364493D+01	0. 152228D+01
0. 991114D+00	0. 374577D+01	- . 488032D+00	0. 182705D+03	- . 179872D+01	- . 488791D+00	- . 847701D-02	0. 364493D+01	0. 152315D+01
0. 992782D+00	0. 402276D+01	- . 707444D+00	0. 195360D+03	- . 152156D+01	- . 362454D+00	- . 575769D-02	0. 364493D+01	0. 152402D+01
0. 993600D+00	0. 416285D+01	- . 8366337D+00	0. 201602D+03	- . 137342D+01	- . 300454D+00	- . 457292D-02	0. 364493D+01	0. 152447D+01
0. 994419D+00	0. 430288D+01	- . 979771D+00	0. 207683D+03	- . 121855D+01	- . 240606D+00	- . 351215D-02	0. 364493D+01	0. 152493D+01
0. 995238D+00	0. 444175D+01	- . 113911D+01	0. 213511D+03	- . 105676D+01	- . 184023D+00	- . 258042D-02	0. 364493D+01	0. 152541D+01
0. 996057D+00	0. 457810D+01	- . 131547D+01	0. 218969D+03	- . 888247D+00	- . 132049D+00	- . 178393D-02	0. 364493D+01	0. 152589D+01
0. 996796D+00	0. 469772D+01	- . 149012D+01	0. 223465D+03	- . 730619D+00	- . 903619D-01	- . 118576D-02	0. 364493D+01	0. 152633D+01
0. 997355D+00	0. 481278D+01	- . 168028D+01	0. 227433D+03	- . 568094D+00	- . 549009D-01	- . 705889D-03	0. 364493D+01	0. 152678D+01
0. 998274D+00	0. 492183D+01	- . 188654D+01	0. 230738D+03	- . 401251D+00	- . 270536D-01	- . 347715D-03	0. 364493D+01	0. 152724D+01
0. 999013D+00	0. 502341D+01	- . 210926D+01	0. 233229D+03	- . 230938D+00	- . 828089D-02	- . 114121D-03	0. 364493D+01	0. 152777D+01
0. 999506D+00	0. 508634D+01	- . 226714D+01	0. 234362D+03	- . 115780D+00	- . 153082D-02	- . 285687D-04	0. 364493D+01	0. 152801D+01
0. 100000D+01	0. 514476D+01	- . 243234D+01	0. 235002D+03	0. 000000D+00	0. 000000D+00	- . 475305D-19	0. 364493D+01	0. 152832D+01

Chapter IV

**IV.1 Input-Output Manual for In-Plane Natural Frequency
and Mode Number Estimation Program, RCNATIN**

This program is written in FORTRAN 77 and allows the interactive estimation of the in-plane natural frequencies and corresponding mode numbers of compliant risers with a planar static configuration without torsion using an asymptotic method described in [4].

Before executing the program the user must make sure that:

- o Devices five and six correspond to input/output from the terminal.
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data preprocessing program, RCINPUT. This file has a logical record length of 80 characters.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.
- o Device eleven must be associated with an output file with logical record length of 80 characters. This file contains the converged natural frequencies and mode numbers as well as the determinant of the characteristic matrix at a series of frequency points. These natural frequency estimates can be used as inputs for a run of RCLINDY1.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The user is informed about the mean current velocity in the x direction and the maximum non-dimensional static effective tension. If the number of static division points is less than four or greater than the value of MNP (maximum number of allowable points), the program halts execution. If the mean velocity is zero, the program halts execution because the asymptotic theory is no longer valid. Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop (see [3]).

The user is then asked to input the upper limit of the frequency range and frequency spacing (in rad/s) for which he wishes to evaluate the characteristic determinant. The default lower limit of the frequency range is 0.01 rad/s. The maximum permissible number of frequency points is also displayed. The frequency spacing should be such so that the frequency points are less than the maximum allowable points.

The user is asked to input the normal drag coefficient used in the static solution. Its value together with the mean current speed is then displayed. The program then asks the user to input estimates for λ and c , see [2,3]. These values will be used for the cable approximation of the planar static solution. λ must have the sign of the mean current speed which is also displayed and for small top x (static) displacements, a good estimate for c is $-1/2\lambda$, [3].

The program calculates λ and c using NAG subroutine C05NBF and displays the solution as well as the values of the functions used to determine the solution. These functions should, ideally, be zero. The user is given the opportunity to redo the calculation of λ and c by trying a different initial estimate (input IPP=1), [3]. Then the program evaluates the characteristic frequency determinant at all points in the interval requested. The determinant includes contributions from the fast and slow in-plane dynamic solutions, see [4].

Then the natural frequencies in the specified range are determined. This is done by checking the change of sign in the characteristic frequency determinant. The program then outputs the results.

In the file connected with device eleven the following are printed:

- o Number of natural frequencies in the specified frequency range.
- o Mode numbers and corresponding natural frequency estimate.
- o Two columns of real numbers, the first of which is the frequency (in rad/s) and the second, the corresponding value of the characteristic determinant at this frequency.

IV.2 Sample Run, Input and Output

IV.2.1 Linear Current Excitation

IV.2.1.1 Interactive Session

INPUT

Device 5 = TERMINAL

Device 8 = RCJ0A01 DATA A*

Device 10 = RCCL2DRR DATA A*

*Files included in Chapter II.

OUTPUT

Device 6 = TERMINAL

Device 11 = FREQIN DATA A

```

T=0 01/0.01 20:54:39
rcnatin rciaoat rcc12drr freqin
FILE 8 DISK RCJDAO1 DATA A ( RECFM FB LRECL 800
FILE 10 DISK RCCL2DRR DATA A ( RECFM FB LRECL 132 BLKSIZE 1320
FILE 11 DISK FREQIN DATA A ( RECFM F LRECL 80
GLOBAL XTLIB VFORLIB CMSLIB NAG1 NAG2
LOAD RCNATIN ( START
EXECUTION BEGINS...
MNP=151
2-D STATIC SOLUTION FROM DEVICE 10
NP = 76
MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S , VM = 0.129000D+01
2-D STATIC SOLUTION SUCCESSFULLY READ
MAXIMUM STATIC EFFECTIVE TENSION/WA*L = 0.3089D+02

INPUT
THE MAXIMUM FREQUENCY AND FREQUENCY SPACING (IN RAD/S) FOR
WHICH YOU WISH TO EVALUATE THE DETERMINANT OF MATRIX P
MAXIMUM PERMISSIBLE NUMBER OF FREQUENCY POINTS IS = 300
?
2.0
INPUT NORMAL DRAG COEFFICIENT USED IN STATIC SOLUTION
?
1.
1.
MEAN CURRENT SPEED =0.129000D+01
NORMAL DRAG COEFFICIENT =0.100000D+01
INPUT ESTIMATES OF LAMBDA AND C
LAMBDA MUST HAVE THE SIGN OF MEAN CURRENT SPEED
FOR SMALL XTOP A GOOD ESTIMATE OF C IS -0.5*LAMBDA
?
2.
3.,1.5
SOLUTION FOR CABLE ...
IFAIL = 0
LAMBDA =0.304294D+01
C = -152147D+01
VALUE OF FUNCTION FOR HORIZONTAL DISPLACEMENT =0.145726D-15
VALUE OF FUNCTION FOR VERTICAL DISPLACEMENT =-.6938889D-16
DO YOU WISH TO TRY A DIFFERENT INITIAL LAMBDA, OR C,
?
1 IF YES INPUT 1
2
2
CABLE APPROXIMATION
LAMBDA =0.304294D+01 C = -152147D+01 TENSION =0.767988D+04
R; T=0 95/1.13 20:55:31
CP spool console stop close

```

IV.2.1.2 FREQIN DATA A

CHINESE LANTERN FROM JOAO, APRIL 1985

ESTIMATES OF IN-PLANE NATURAL FREQUENCIES FOR A 2-D STATIC CONFIGURATION

NUMBER OF NATURAL FREQUENCIES IN SPECIFIED RANGE: 6

MODE = 1 NATURAL FREQUENCY BETWEEN

0.470000D+00 AND 0.480000D+00 RAD/SEC

MODE = 2 NATURAL FREQUENCY BETWEEN

0.780000D+00 AND 0.790000D+00 RAD/SEC

MODE = 3 NATURAL FREQUENCY BETWEEN

0.108000D+01 AND 0.109000D+01 RAD/SEC

MODE = 4 NATURAL FREQUENCY BETWEEN

0.137000D+01 AND 0.138000D+01 RAD/SEC

MODE = 5 NATURAL FREQUENCY BETWEEN

0.166000D+01 AND 0.167000D+01 RAD/SEC

MODE = 6 NATURAL FREQUENCY BETWEEN

0.194000D+01 AND 0.195000D+01 RAD/SEC

FREQUENCY (RAD/SEC) DETERMINANT

0.100000D-01 -.845239D+07

0.200000D-01 -.169912D+08

0.300000D-01 -.257024D+08

0.400000D-01 -.346708D+08

0.500000D-01 -.439779BD+08

0.600000D-01 -.537102D+08

0.700000D-01 -.639399D+08

0.800000D-01 -.747423D+08

0.900000D-01 -.861858D+08

0.100000D+00 -.983321D+08

0.110000D+00 -.111234D+09

0.120000D+00 -.124936D+09

0.130000D+00 -.139470D+09

0.140000D+00 -.154853D+09

0.150000D+00 -.171088D+09

0.160000D+00 -.188160D+09

0.170000D+00 -.206032D+09

0.180000D+00 -.224648D+09

0.190000D+00 -.243926D+09

0.200000D+00 -.263758D+09

0.210000D+00 -.284010D+09

0.220000D+00 -.304517D+09

0.230000D+00 -.325088D+09

0.240000D+00 -.345500D+09

0.250000D+00 -.365501D+09

A DATA

0.260000D+00	- .384809D+09
0.270000D+00	- .403118D+09
0.280000D+00	- .420093D+09
0.290000D+00	- .435379D+09
0.300000D+00	- .448600D+09
0.310000D+00	- .459367D+09
0.320000D+00	- .467281D+09
0.330000D+00	- .471937D+09
0.340000D+00	- .472933D+09
0.350000D+00	- .469878D+09
0.360000D+00	- .462394D+09
0.370000D+00	- .450128D+09
0.380000D+00	- .432761D+09
0.390000D+00	- .410013D+09
0.400000D+00	- .209611D+09
0.410000D+00	- .381652D+09
0.420000D+00	- .347504D+09
0.430000D+00	- .261481D+09
0.440000D+00	- .209611D+09
0.450000D+00	- .151979D+09
0.460000D+00	- .8888032D+08
0.470000D+00	- .203980D+08
0.480000D+00	0.528234D+08
0.490000D+00	0.130348D+09
0.500000D+00	0.211562D+09
0.510000D+00	0.295753D+09
0.520000D+00	0.382112D+09
0.530000D+00	0.469739D+09
0.540000D+00	0.557647D+09
0.550000D+00	0.644776D+09
0.560000D+00	0.729994D+09
0.570000D+00	0.812118D+09
0.580000D+00	0.889921D+09
0.590000D+00	0.962148D+09
0.600000D+00	0.102753D+10
0.610000D+00	0.108482D+10
0.620000D+00	0.113277D+10
0.630000D+00	0.117020D+10
0.640000D+00	0.119598D+10
0.650000D+00	0.120906D+10
0.660000D+00	0.120849D+10
0.670000D+00	0.119347D+10
0.680000D+00	0.116332D+10
0.690000D+00	0.111750D+10
0.700000D+00	0.105569D+10
0.710000D+00	0.977720D+09
0.720000D+00	0.883648D+09
0.730000D+00	0.773736D+09
0.740000D+00	0.648469D+09
0.750000D+00	0.508562D+09
0.760000D+00	0.354957D+09
0.770000D+00	0.188828D+09
0.780000D+00	0.115740D+08
0.790000D+00	- .175189D+09
0.800000D+00	- .369637D+09
0.810000D+00	- .569752D+09

0. 820000D+00 -. 773342D+09
0. 830000D+00 -. 978059D+09
0. 840000D+00 -. 118143D+10
0. 850000D+00 -. 138087D+10
0. 860000D+00 -. 157372D+10
0. 870000D+00 -. 175730D+10
0. 880000D+00 -. 192891D+10
0. 890000D+00 -. 208586D+10
0. 900000D+00 -. 222556D+10
0. 910000D+00 -. 234550D+10
0. 920000D+00 -. 244334D+10
0. 930000D+00 -. 251688D+10
0. 940000D+00 -. 25641BD+10
0. 950000D+00 -. 258352D+10
0. 960000D+00 -. 25734BD+10
0. 970000D+00 -. 253296D+10
0. 980000D+00 -. 246121D+10
0. 990000D+00 -. 235784D+10
0. 100000D+01 -. 222285D+10
0. 101000D+01 -. 205665D+10
0. 102000D+01 -. 186008D+10
0. 103000D+01 -. 163438D+10
0. 104000D+01 -. 138123D+10
0. 105000D+01 -. 110274D+10
0. 106000D+01 -. 801388D+09
0. 107000D+01 -. 480084D+09
0. 108000D+01 -. 142086D+09
0. 109000D+01 0. 209007D+09
0. 110000D+01 0. 569294D+09
0. 111000D+01 0. 934606D+09
0. 112000D+01 0. 130056D+10
0. 113000D+01 0. 166259D+10
0. 114000D+01 0. 201604D+10
0. 115000D+01 0. 235616D+10
0. 116000D+01 0. 267824D+10
0. 117000D+01 0. 297759D+10
0. 118000D+01 0. 324968D+10
0. 119000D+01 0. 349014D+10
0. 120000D+01 0. 369489D+10
0. 121000D+01 0. 386013D+10
0. 122000D+01 0. 398245D+10
0. 123000D+01 0. 405886D+10
0. 124000D+01 0. 408688D+10
0. 125000D+01 0. 406454D+10
0. 126000D+01 0. 399046D+10
0. 127000D+01 0. 386387D+10
0. 128000D+01 0. 368464D+10
0. 129000D+01 0. 345332D+10
0. 130000D+01 0. 317112D+10
0. 131000D+01 0. 283995D+10
0. 132000D+01 0. 246241D+10
0. 133000D+01 0. 204177D+10
0. 134000D+01 0. 158193D+10
0. 135000D+01 0. 108744D+10
0. 136000D+01 0. 563414D+09
0. 137000D+01 0. 154948D+08

0. 138000D+01 -. 550190D+09
 0. 139000D+01 -. 112711D+10
 0. 140000D+01 -. 170838D+10
 0. 141000D+01 -. 228688D+10
 0. 142000D+01 -. 285529D+10
 0. 143000D+01 -. 340623D+10
 0. 144000D+01 -. 393230D+10
 0. 145000D+01 -. 442622D+10
 0. 146000D+01 -. 488090D+10
 0. 147000D+01 -. 528954D+10
 0. 148000D+01 -. 564576D+10
 0. 149000D+01 -. 594362D+10
 0. 150000D+01 -. 617780D+10
 0. 151000D+01 -. 634361D+10
 0. 152000D+01 -. 643711D+10
 0. 153000D+01 -. 645517D+10
 0. 154000D+01 -. 639552D+10
 0. 155000D+01 -. 625682D+10
 0. 156000D+01 -. 603871D+10
 0. 157000D+01 -. 574180D+10
 0. 158000D+01 -. 536775D+10
 0. 159000D+01 -. 491920D+10
 0. 160000D+01 -. 439985D+10
 0. 161000D+01 -. 381435D+10
 0. 162000D+01 -. 316834D+10
 0. 163000D+01 -. 246837D+10
 0. 164000D+01 -. 172183D+10
 0. 165000D+01 -. 936886D+09
 0. 166000D+01 -. 122420D+09
 0. 167000D+01 0. 712103D+09
 0. 168000D+01 0. 155672D+10
 0. 169000D+01 0. 240110D+10
 0. 170000D+01 0. 323465D+10
 0. 171000D+01 0. 404664D+10
 0. 172000D+01 0. 482637D+10
 0. 173000D+01 0. 556328D+10
 0. 174000D+01 0. 624709D+10
 0. 175000D+01 0. 686794D+10
 0. 176000D+01 0. 741653D+10
 0. 177000D+01 0. 788426D+10
 0. 178000D+01 0. 826335D+10
 0. 179000D+01 0. 862786D+10
 0. 180000D+01 0. 837098D+10
 0. 185000D+01 0. 800246D+10
 0. 186000D+01 0. 752436D+10
 0. 187000D+01 0. 880553D+10
 0. 188000D+01 0. 877245D+10
 0. 189000D+01 0. 854693D+10
 0. 190000D+01 0. 872921D+10
 0. 191000D+01 0. 880553D+10
 0. 192000D+01 0. 694017D+10
 0. 193000D+01 0. 264754D+10
 0. 157664D+10

FILE: FREQIN DATA A
VM/SP CONVERSATIONAL MONITOR SYSTEM

PAGE 005

0.194000D+01	0.460907D+09
0.195000D+01	-686665D+09
0.196000D+01	-185240D+10
0.197000D+01	-302208D+10
0.198000D+01	-418116D+10
0.199000D+01	-531488D+10
0.200000D+01	-640852D+10
0.201000D+01	-744752D+10

Chapter V

V.1 Input-Output Manual for the In-Plane Linear Eigenproblem Solution Program, RCLINDY1

This program is written in FORTRAN 77 and allows the interactive solution of the in-plane linear eigenproblem for a compliant riser with a planar static configuration without torsion as described in [4].

Before executing the program the user must make sure that:

- o Devices five and six correspond to input/output from the terminal.
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data preprocessing program RCINPUT. This file has a logical record length of 80 characters.
- o Device nine must be associated with an output data file with logical record length of 132 characters which upon completion of the program contains a complete list of the input, initial approximation and the converged solution of the problem.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.
- o Device eleven must be associated with an output file with logical record length of 117 characters. This file contains the converged solution of the problem in a concise form appropriate for use in the preparation of illustrations and/or as an input to RCLINDY1 from device twelve in a subsequent execution of this program.

- o Device twelve may be associated with an input file containing an initial approximate solution of an in-plane linear eigenproblem, resulting from a previous execution of RCLINDY1. This option can be used if greater accuracy is needed for the solution of the problem. Device twelve may be associated with a dummy file name if this option is not used.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The user is informed about the mean current velocity in the x direction and the maximum non-dimensional static effective tension. If the number of static discretization points is less than four or greater than the value of MNP (maximum number of allowable points), the program halts execution. If the mean velocity is zero, the program halts execution because the asymptotic theory is not valid for this case. Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop, see [3].

Next, the user is asked whether he wants the initial approximation to be calculated using asymptotics (input IEXIST=0) or whether he wants to use the option to read an initial approximation from device twelve created by a previous run of RCLINDY1 (input IEXIST=1).

Case 1 (IEXIST = 0)

The user is asked to input the initial increment of the continuation parameter $\delta\epsilon$ (condition $0 < \delta\epsilon \leq 1$). If no continuation is required, then use $\delta\epsilon=1$.

If continuation is required, then $\epsilon=0.1$ will usually suffice. For large changes between initial and final problem, a smaller value of ϵ , e.g. 0.05 or 0.025 might be necessary. The user is asked to input the normal drag coefficient used in the static solution. Its value, together with the mean current speed, is subsequently displayed. The program asks the user to input estimates for λ and c [2,3]. These values will be used for the cable approximation of the planar static solution. λ must have the sign of the mean current speed, which is also displayed and for small top x displacement, a good estimate for c is $-(1/2)\lambda$, [3]. The program calculates λ and c and displays the solution as well as the values of the functions used to determine the solution. These functions should, ideally, be zero. The user is given the opportunity to redo the calculation of λ and c by trying a different initial estimate (input IPP=1), [3]. Next, the user is asked to supply an initial guess for the natural frequency in rad/s. The information for the approximate natural frequencies is obtained by running RCNATIN program which provides an estimate of the natural frequencies and their corresponding mode number.

During the calculation of the approximate natural frequency the user is supplied with information about the determinant of the characteristic matrix. Ideally this should become zero. The user may ignore the value of IFAIL printed in this case. Vanishing of the determinant of the characteristic matrix allows the program to calculate the approximate natural frequency and display its value. Then the user is given the following alternatives:

- o Input 1 if he wants to redo the calculation. This may be used when the converged frequency does not correspond to the mode which the user wants to evaluate.

- o Input 2 if he chooses to stop.
- o Input any other integer to continue with the program.

Subsequently, the user is asked to input the mode number corresponding to the approximate natural frequency, as determined from the RCNATIN program.

Next a linear system of five equations is solved to determine the constant coefficients multiplying the various terms composing the overall solution allowing us to obtain the initial approximation [4]. The values of the above constant coefficients are displayed on the screen. In addition, the sixth equation is also evaluated for the values of the coefficients determined above. Ideally it should be zero.

The initial approximation is then evaluated and the orthonormalizing constant used for p and q as well as the fourth boundary condition at s=0 are printed. The user is then asked if he wants to review the initial approximation by displaying it in his terminal. If he chooses to do so, he should input 1. Subsequently the approximate solution is printed in the file associated with the device nine. The following quantities are printed:

- o Number of discretization points NP
- o Natural frequency
- o Order of point (starting from the lower end)
- o Non-dimensional arc length
- o Non-dimensional dynamic effective tension, T_1
- o Non-dimensional shear force in the $\vec{\xi}$ direction, Q_1^ξ
- o Non-dimensional component of rate of rotation about \vec{n}_1 , Ω_1^n
- o Euler angle ϕ_1

- o Non-dimensional displacement p
- o Non-dimensional displacement q
- o Non-dimensional natural frequency.

Subsequently, the program prints the maximum of the absolute value of the initial ($\epsilon=0$) non-dimensional component of $\vec{\Omega}_1$ in the \vec{n} direction. This number is used to estimate a reasonable tolerance for convergence of iterations. The user is requested to input a tolerance as a fraction of this number; e.g., 0.01 for 1% tolerance or better for all quantities of interest. Typically Ω_1^n will be determined with an accuracy a little better than the above fraction and all other quantities will be even more accurate.

The user is given the opportunity to stop the program at this point if he wishes. He should input zero if he wants to stop.

Case 2 (IEXIST = 1)

In this case an initial approximation to the solution for $\epsilon=1$ is available from a previous run of RCLINDY1. The program reads the information from the previous run concerning the number of subdivision points, NP, the mode number and the corresponding natural frequency. If the number of subdivision points is greater than MNP, the program stops execution.

Then the user has the option to review the initial approximation by displaying it in his terminal. The assumed boundary condition for Ω_1^n at the lower end of the riser [4] is also printed.

Subsequently, the program prints the maximum of the absolute value of the initial component of $\vec{\Omega}_1$ in the \vec{n} direction. This number is used to

estimate a reasonable tolerance for the convergence of iterations. The user is requested to input a tolerance as a fraction of this number e.g. 0.01 for 1% tolerance or better for all quantities of interest. Typically Ω_1^2 will be determined with an accuracy a little better than the above fraction and all other quantities will be even more accurate. Subsequently the user is given the opportunity to halt the execution of the program.

After the initial approximation is fully defined the program enters NAG subroutine D02RAF, which provides the iterative numerical solution to the problem. The manual of this subroutine is included in Chapter 10. The input value of IFAIL is 111, indicating that the soft failure option with error message and monitoring information printing has been selected. Error messages are directed to device six.

Once the execution of D02RAF is completed, the current value of IFAIL is printed on the terminal. For an explanation of this parameter, see NAG manual for D02RAF, Chapter 10. Only if IFAIL=0 or IFAIL=4, the execution of the program continues with printout of the riser characteristics and the final solution of the problem. Otherwise the program stops. When IFAIL ≠ 0, the user should consult the manual quoted above for interpretation of the results and a possible course of action. If IFAIL=0 or 4, the riser characteristics and the final solution of the problem are printed in the file associated with device nine. For the riser characteristics the format used in the program RCSTAT2D is also used here. This is described in [3] (pp. 23-25).

Following, the non-dimensional in-plane solution is written in the file associated with device nine. The mode number, natural frequency and final number of points NP at which the solution is available is printed first.

Then the following data is printed for I=1 to NP: s , T_1 , Q_1^ξ , Q_1^η , ϕ_1 , p , q , σ with

FORMAT (8(1X,D10.4)).

Next the maximum (non-dimensional) estimated errors of T_1 , Q_1^ξ , Q_1^η , ϕ_1 , p , q , σ provided by the NAG subroutine D02RAF are printed. Subsequently, the following data is written in the file associated with device eleven:

MODE, NP, SIGMAD, XTOP, YTOP = Mode number, number of points at which converged solution is available, dimensional natural frequency (in rad/s) and x and y static displacements at the top with

FORMAT (1X,I2,1X,I3,3 (1X,D10.4))

and for I=1 to NP: the non-dimensional

s , T_1 , Q_1^ξ , Q_1^η , ϕ_1 , p , q , σ and the dimensional current velocity $v_c(s)$ with

FORMAT (9(1X,D12.6)).

The data written in the file associated with device eleven, due to their simple form, can be used very easily as input to plotting programs. In addition, this data is useful in subsequent runs of RCLINDY1, if such an option is selected, as explained before.

V.2 Sample Run, Input and Output

V.2.1 Linear Current Excitation

V.2.1.1 Interactive Session

INPUT

Device 5 = TERMINAL

Device 8 = RCJOA01 DATA A*

Device 10 = RCCL2DRR DATA A*

Device 12 = DUMMY DATA A

* Files included in Chapter II.

OUTPUT

Device 6 = TERMINAL
Device 9 = INPLA1 DATA A
Device 11 = INPLA1A DATA A

```

R; T=0.01/0.01 20:26:33
rc1 lindy1 rcoact1 top1 rec1/2drr inplata dummy
FI 8 DISK RCJOAO1 DATA A ( RECFM FB LRECL 80 BLKSIZE 800
FI 9 DISK INPLA1 DATA A ( RECFM FB LRECL 132 BLKSIZE 1320
FI 10 DISK RCCL2DRR DATA A ( RECFM FB LRECL 132 BLKSIZE 1320
FI 11 DISK INPLA1A DATA A ( RECFM FB LRECL 117 BLKSIZE 1320
FI 12 DISK DUMMY DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
GLOBAL TXTLIB VFORTLIB CMSLIB NAG1 NAG2
LOAD RCLINDY1 { START
EXECUTION BEGINS...
MNP=151

2-D STATIC SOLUTION FROM DEVICE 10
NP = 76

MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S . VM =0.129000D+01

2-D STATIC SOLUTION SUCCESSFULLY READ
MAXIMUM STATIC EFFECTIVE TENSION/WA*L = 0.3089D+02
DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS
IF YES INPUT 1 , IF NO INPUT 0
?
0
```

```

INPUT 1 IF YOU WISH TO READ INITIAL APPROXIMATION,
ELSE INPUT 0 (AN INITIAL APPROXIMATION WILL BE CALCULATED LOCALLY)
?
0
INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS
IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.D0
RECOMMENDATION : IF CONTINUATION IS REQUIRED THEN 0.D0 < DELEPS < 1.D0
USUALLY DELEPS = 0.1D0 WILL SUFFICE
FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM
A SMALLER VALUE OF DELEPS, E.G. 0.05D0 OR 0.025D0 MIGHT BE
NECESSARY
?
.
```

```

INPUT NORMAL DRAG COEFFICIENT USED IN STATIC SOLUTION
?
1
.
```

```

MEAN CURRENT SPEED =0.129000D+01
NORMAL DRAG COEFFICIENT =0.100000D+01
INPUT ESTIMATES OF LAMBDA AND C
LAMBDA MUST HAVE THE SIGN OF MEAN CURRENT SPEED
FOR SMALL XTOP A GOOD ESTIMATE OF C IS -0.5*LAMBDA
?
3. -1.5
SOLUTION FOR CABLE ...
FAIL = 0
LAMBDA =0.304294D+01
C = -152147D+01
VALUE OF FUNCTION FOR HORIZONTAL DISPLACEMENT =0.145726D-15
VALUE OF FUNCTION FOR VERTICAL DISPLACEMENT = -693889D-16
DO YOU WISH TO TRY A DIFFERENT INITIAL LAMDA, C, OR TOLERANCE
IF YES INPUT 1
?
```

```

2
CABLE APPROXIMATION
LAMBDA =0.304294D+01 C = -152147D+01 TENSION =0.767988D+04
INPUT INITIAL GUESS FOR SIGMA IN RAD/S
?
;
```

```

IF FAIL FOR THE DETERMINANT OF P = 0
DET = 0.130348D+09
IF FAIL FOR THE DETERMINANT OF P = 0
DET = 0.130348D+09
IF FAIL FOR THE DETERMINANT OF P = 0
DET = 0.543073D+07
IF FAIL FOR THE DETERMINANT OF P = 0
DET = 0.266538D+06
IF FAIL FOR THE DETERMINANT OF P = 0
DET = 0.645052D+03
IF FAIL FOR THE DETERMINANT OF P = 0
DET = 0.771932D-01
IF FAIL FOR THE DETERMINANT OF P = 1
DET = 0.000000D+00
IF FAIL FOR APPROXIMATE EVALUATION OF SIGMA IS
THE DETERMINANT IS = 0.0000D+00
THE CONVERGED SIGMA IS 0.4729D+00
INPUT 1 IF YOU WANT TO REDO THE CALCULATION
INPUT 2 IF YOU WANT TO STOP
?

```

3 INPUT MODE NUMBER CORRESPONDING TO ABOVE APPROXIMATE SIGMA

```

1 IF FAIL FOR THE EVALUATION OF CONSTANT COEFFICIENTS IS
CONSTANT COEFFICIENTS ARE :
A1 = 1.00 B1 = -2.076D+01
G1 = -2.251D-02 D1 = -2.251D-02
C1 = -15.16D-15 C2 = 0.5535D+00
SIXTH EQUATION = -1102D-13
THE ORTHONORMALIZING CONSTANT FOR THE APPROXIMATE
SOLUTION IS SQRT(A) = 0.3265D+01
THE BOUNDARY CONDITION AT S=0 IS OMEGAETA(0) = 0.6791D+03
DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION
IF YES INPUT 1
?

```

```

2 MAXIMUM ABSOLUTE VALUE OF N-D OMEGA ETA IS =0.679078D+03
THIS NUMBER CAN BE USED TO ESTIMATE
A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS
INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE
E.G. INPUT 0.01 FOR 1% ACCURACY
?
.O!
IF YOU WANT TO STOP INPUT 0
?
1

```

DO2RAF MONITORING INFORMATION

```

MONITORING NEWTON ITERATION
NUMBER OF POINTS IN CURRENT MESH = 76
CORRECTION NUMBER 0 RESIDUAL BE .LE. 1.00D+00
ITERATION NUMBER 0 RESIDUAL = 2.330D+01
SQUARED NORM OF CORRECTION = 1.10D+05
SQUARED NORM OF GRADIENT = 6.84D+05
SCALAR PRODUCT OF CORRECTION AND GRADIENT = 5.43D+02
ITERATION NUMBER 1 RESIDUAL = 1.28D-02

```

INITIALIZATION PARAMETER EPSILON = 2.00D-01 DEFLPS = 2.00D-01

MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 76
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00D+00
 ITERATION NUMBER 0 RESIDUAL = 2.17D-03
 CONTINUATION PARAMETER EPSILON = 6.00D-01 DELEPS = 4.00D-01
 MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 76
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00D+00
 ITERATION NUMBER 0 RESIDUAL = 3.18D-02
 CONTINUATION PARAMETER EPSILON = 1.00D+00 DELEPS = 8.00D-01
 MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 76
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 6.79D-04
 ITERATION NUMBER 0 RESIDUAL = 3.71D-01
 SQUARED NORM OF CORRECTION = 5.71D+02
 SCALAR PRODUCT OF GRADIENT = 6.85D-02
 ITERATION NUMBER 1 RESIDUAL = 1.37D-01
 MESH SELECTION
 NUMBER OF NEW POINTS 9
 MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 85
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 6.79D-04
 ITERATION NUMBER 0 RESIDUAL = 7.03D-01
 SQUARED NORM OF CORRECTION = 5.35D+00
 SCALAR PRODUCT OF GRADIENT = 1.02D+01
 ITERATION NUMBER 1 RESIDUAL = 4.95D-01
 MESH SELECTION
 NUMBER OF NEW POINTS 0
 CORRECTION NUMBER 0 ESTIMATED MAXIMUM ERROR = 1.64D+00
 ESTIMATED ERROR BY COMPONENTS
 4.87D-02 4.25D-02 1.64D+00 3.75D-02 4.72D-03 3.10D-03 1.92D-02
 IF FAIL = 0
 RI: T=2.88/3.38 20:29:45
 CP spool console stop close

V.2.1.2 INPLA1 DATA A

INITIAL CONDITION FOR EPS=0. AND NP = 76 POINTS. NATURAL FREQUENCY = 0.4729D+00 RAD/SEC

I	ARC TENSION	SHEAR XI	OMEGA ETA	PHI 1	P	Q	SIGMA
1	0.00000D+00	0.816109D+01	0.367700D+01	0.679078D+03	-256387D-14	0.425061D-17	0.63592D-16
2	0.986636D-03	0.733677D+01	0.484345D+01	0.647797D+03	0.670003D+00	0.141105D-07	0.538535D+01
3	0.246464D-02	0.614038D+01	0.615942D+01	0.559777D+03	0.155819D+01	0.101846D-05	0.538535D+01
4	0.394265D-02	0.512495D+01	0.533924D+01	0.483255D+03	0.232471D+01	0.483647D-05	0.421723D-02
5	0.558056D-02	0.427882D+01	0.454522D+01	0.410586D+03	0.305205D+01	0.143182D-04	0.538535D+01
6	0.721847D-02	0.365796D+01	0.385141D+01	0.348662D+03	0.366972D+01	0.311810D-04	0.842924D-02
7	0.885637D-02	0.320548D+01	0.326220D+01	0.296549D+03	0.419420D+01	0.570979D-04	0.200768D-01
8	0.111404D-01	0.277024D+01	0.262022D+01	0.236655D+03	0.479614D+01	0.111065D-03	0.203035D-01
9	0.134244D-01	0.249366D+01	0.210211D+01	0.188351D+03	0.527525D+01	0.188785D-03	0.415974D-01
10	0.157084D-01	0.232074D+01	0.167533D+01	0.149822D+03	0.565563D+01	0.293039D-03	0.538535D+01
11	0.178993D-01	0.221849D+01	0.134622D+01	0.120239D+03	0.594881D+01	0.420136D-03	0.666549D-01
12	0.200901D-01	0.215659D+01	0.108287D+01	0.964709D+02	0.618338D+01	0.575518D-03	0.538535D+01
13	0.222809D-01	0.212103D+01	0.865964D+00	0.73226D+02	0.637150D+01	0.760675D-03	0.936249D-01
14	0.244717D-01	0.210284D+01	0.695431D+00	0.621546D+02	0.652218D+01	0.976910D-03	0.107729D+00
15	0.280184D-01	0.209467D+01	0.502107D+00	0.434895D+02	0.670460D+01	0.139558D-02	0.131118D+00
16	0.315651D-01	0.2223106D+01	0.476954D-01	0.314968D+00	0.299003D+02	0.683067D+01	0.190264D-02
17	0.351118D-01	0.211717D+01	0.225477D+00	0.204601D+02	0.691670D+01	0.250150D-02	0.151189D+00
18	0.413491D-01	0.215181D+01	0.134957D+00	0.958401D+01	0.690001D+01	0.178239D-01	0.179578D+00
19	0.475865D-01	0.218509D+01	0.822325D-01	0.290324D+01	0.700267D+01	0.378616D-02	0.223028D+00
20	0.547166D-01	0.2223106D+01	0.476954D-01	0.2272260D+01	0.703625D+01	0.537778D-02	0.266832D+00
21	0.618467D-01	0.21172D+01	0.314968D-01	0.169761D+01	0.703674D+01	0.758883D-02	0.3117008D+00
22	0.778360D-01	0.236559D+01	0.147524D-01	0.204601D+02	0.691670D+01	0.102338D-01	0.151189D+00
23	0.954915D-01	0.246676D+01	0.752016D-02	0.124492D+02	0.671044D+01	0.290400D-01	0.538535D+01
24	0.114743D+00	0.257517D+01	0.103262D-01	0.104318D-01	0.611994D+02	0.643483D+01	0.449142D-01
25	0.135516D+00	0.268852D+01	0.280226D+01	0.106993D-01	0.202518D+02	0.605611D+01	0.724019D+00
26	0.157726D+00	0.291017D+01	0.246699D-01	0.455590D+01	0.455583D+01	0.666107D-01	0.852684D+00
27	0.181288D+00	0.291017D+01	0.147524D-01	0.8779965D+01	0.690001D+01	0.953877D-01	0.367070D+00
28	0.206107D+00	0.300242D+01	0.1108552D-01	0.295053D+02	0.492065D+01	0.132543D+00	0.478208D+00
29	0.232087D+00	0.306624D+01	0.108783D-01	0.346935D+02	0.412455D+01	0.290400D-01	0.598075D+00
30	0.259123D+00	0.308457D+01	0.100223D-01	0.530004D+02	0.315204D+01	0.236589D+00	0.1207676D+01
31	0.287110D+00	0.303583D+01	0.808126D-02	0.455718D+02	0.199087D+01	0.304879D+00	0.112023D+01
32	0.315938D+00	0.289406D+01	0.454131D-02	0.505233D+02	0.639462D+00	0.135311D+01	0.538535D+01
33	0.374592D+00	0.263045D+01	0.994640D-03	0.541976D+02	0.883430D+00	0.471432D+00	0.110024D+01
34	0.3750000D+00	0.221723D+01	0.849673D-02	0.554677D+02	0.252672D+01	0.564527D+00	0.134691D+01
35	0.406308D+00	0.163440D+01	0.171634D-01	0.530004D+02	0.419589D+01	0.657584D+00	0.126447D+01
36	0.437333D+00	0.879118D+00	0.523162D-01	0.329542D+02	0.574774D+01	0.859206D+00	0.112023D+01
37	0.468605D+00	0.252697D-01	0.308184D-01	0.473360D+02	0.700649D+01	0.813218D+00	0.743296D+00
38	0.500000D+00	0.102540D+01	0.319792D-01	0.522549D-02	0.779686D+01	0.859206D+00	0.335834D+00
39	0.5131395D+00	0.204428D+01	0.101649D-01	0.959160D-03	0.534017D+02	0.799660D+01	0.858535D+01
40	0.562267D+00	0.299771D+01	0.214453D-01	0.456243D+02	0.220777D+02	0.757630D+01	0.858355D+01
41	0.593691D+00	0.381452D+01	0.130617D-01	0.473360D+02	0.661465D+01	0.813218D+00	0.646842D+00
42	0.624345D+00	0.445157D+01	0.987896D-02	0.430250D+02	0.526319D+01	0.743296D+00	0.912648D+00
43	0.654508D+00	0.489741D+01	0.319792D-01	0.52449496D+02	0.369784D+01	0.657584D+00	0.335834D+00
44	0.684062D+00	0.204428D+01	0.516620D+01	0.5233392D-02	0.207471D+01	0.564527D+00	0.405040D-15
45	0.712889D+00	0.493935D+01	0.101529D-01	0.514282D+02	0.508355D+00	0.859206D+00	0.112023D+01
46	0.740877D+00	0.529456D+01	0.130617D-01	0.467792D+02	0.930028D+00	0.471432D+00	0.335834D+00
47	0.7677913D+00	0.459677D+01	0.929419D-02	0.430250D+02	0.220421D+01	0.8363734D+00	0.137362D+01
48	0.793893D+00	0.509516D+01	0.995628D-02	0.380425D+02	0.330207D+01	0.304879D+00	0.135311D+01
49	0.818712D+00	0.516620D+01	0.101649D-01	0.330682D+02	0.422638D+01	0.236589D+00	0.129475D+01
50	0.842274D+00	0.493935D+01	0.101529D-01	0.282956D+02	0.498760D+01	0.132543D+00	0.120767D+01
51	0.864484D+00	0.459677D+01	0.100135D-01	0.237939D+02	0.560117D+01	0.953877D+00	0.110024D+01
52	0.885257D+00	0.443013D+01	0.990896D-02	0.196380D+02	0.608260D+01	0.666107D-01	0.852684D+00
53	0.904508D+00	0.427423D+01	0.121583D+02	0.157762D+02	0.645094D+01	0.449142D-01	0.724019D+00

- 5398078D+00 0.290403D-01 - 5398078D+00 0.290403D-01 - 5398078D+00 0.290403D-01

54 0.922164D+00 - .443167D+01 0.144417D-01 0.859149D+01 0.690519D+01 0.178239D-01 - .478208D+00 0.5388535D+01
 55 0.938153D+00 - .400281D+01 0.261773D-01 0.443638D+01 0.701489D+01 0.102339D-01 - .367072D+00 0.5388535D+01
 56 0.945283D+00 - .394559D+01 0.475559D-01 0.159671D+01 0.703881D+01 0.7588897D-02 - .317011D+00 0.5388535D+01
 57 0.952414D+00 - .388782D+01 0.821929D-01 - .299471D+01 0.703766D+01 0.537764D-02 - .266829D+00 0.5388535D+01
 58 0.958651D+00 - .383625D+01 0.134972D+00 - .967204D+01 0.700353D+01 0.378614D-02 - .223027D+00 0.5388535D+01
 59 0.964880D+00 - .378642D+01 0.225410D+00 - .205471D+02 0.691701D+01 0.250153D-02 - .179579D+00 0.5388535D+01
 60 0.968435D+00 - .376244D+01 0.314744D+00 - .299819D+02 0.683066D+01 0.190263D-02 - .155189D+00 0.5388535D+01
 61 0.971982D+00 - .374671D+01 0.501690D+00 - .435587D+02 0.670432D+01 0.139553D-02 - .131178D+00 0.5388535D+01
 62 0.975528D+00 - .374682D+01 0.694851D+00 - .622029D+02 0.652172D+01 0.976923D-03 - .107731D+00 0.5388535D+01
 63 0.977719D+00 - .376105D+01 0.865252D+00 - .773590D+02 0.637093D+01 0.760684D-03 - .936256D-01 0.5388535D+01
 64 0.979910D+00 - .379438D+01 0.108201D+01 - .964922D+02 0.618273D+01 0.575510D-03 - .799014D-01 0.5388535D+01
 65 0.982101D+00 - .385689D+01 0.133751D+01 - .120243D+03 0.594811D+01 0.420117D-03 - .666531D-01 0.5388535D+01
 66 0.984292D+00 - .396487D+01 0.165223D+01 - .149484D+03 0.565583D+01 0.293018D-03 - .539982D-01 0.5388535D+01
 67 0.986005D+00 - .409586D+01 0.196344D+01 - .17198D+03 0.537822D+01 0.212246D-03 - .446103D-01 0.5388535D+01
 68 0.987718D+00 - .428705D+01 0.233600D+01 - .210294D+03 0.504875D+01 0.146756D-03 - .357576D-01 0.5388535D+01
 69 0.989431D+00 - .456233D+01 0.276922D+01 - .249554D+03 0.465775D+01 0.954559D-04 - .275398D-01 0.5388535D+01
 70 0.991144D+00 - .495466D+01 0.328231D+01 - .296064D+03 0.419377D+01 0.570909D-04 - .200753D-01 0.5388535D+01
 71 0.992782D+00 - .548057D+01 0.386595D+01 - .348616D+03 0.366934D+01 0.311749D-04 - .137708D-01 0.5388535D+01
 72 0.994419D+00 - .620745D+01 0.454591D+01 - .410517D+03 0.305211D+01 0.143216D-04 - .843054D-02 0.5388535D+01
 73 0.996057D+00 - .720526D+01 0.534022D+01 - .483205D+03 0.232483D+01 0.483784D-05 - .421800D-02 0.5388535D+01
 74 0.997535D+00 - .840856D+01 0.616029D+01 - .559740D+03 0.155829D+01 0.101897D-05 - .155352D-02 0.5388535D+01
 75 0.999013D+00 - .983077D+01 0.484415D+01 - .647771D+03 0.670238D+00 0.141589D-07 - .144511D-03 0.5388535D+01
 76 0.100000D+01 - .108123D+02 0.367725D+01 - .679066D+03 - .567637D-14 0.127518D-16 0.850122D-17 0.5388535D+01

CHINESE LANTERN FROM JOAO, LINEAR DYNAMICS PROGRAMS, MARCH 1985

9 = NUMBER OF RISER SEGMENTS

0.883920D+02 = UNSTRETCHED RISER LENGTH IN M

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M3

0.102500D+04 = SALT WATER DENSITY IN KG/M3

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M2

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M2

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.797324D+04 = AVERAGE STATIC TENSION IN N

0.129000D+01 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.189584D-16 = X COORDINATE AT TOP IN M

0.701000D+02 = Y COORDINATE AT TOP IN M

DATA PER RISER SEGMENTS FOR NSEG = 9 SEGMENTS

DIMENSIONAL QUANTITIES IN THE SYSTEM
 RLNG DXI PXIETA AO WEIGHT MASS TMASS AMAETA AMAZI TMAXI TMAZI
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.4993D+02
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.4993D+02
 0.6839D+02 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.4993D+02
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.4993D+02

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.4993D+02
 EA EIETA EIETAS EIIXI EIXIS GIP GIPS DETA JZI AJZI TJZI
 0.2670D+09 0.6600D+04 - .3300D+03 0.2440D+05 -.1220D+04 0.1164D+07 -.5820D+05 0.2000D+00 0.4932D+00 0.7806D-.01 0.5713D+00
 0.2670D+09 0.5775D+04 - .3300D+03 0.2135D+05 -.1220D+04 0.1019D+07 -.5820D+05 0.2000D+00 0.4932D+00 0.7806D-.01 0.5713D+00
 0.2670D+09 0.4950D+04 - .3300D+04 0.0000D+00 0.1220D+05 0.0000D+00 0.5820D+06 0.0000D+00 0.4932D+00 0.7806D-.01 0.5713D+00
 0.2670D+09 0.4125D+04 0.3300D+03 0.1525D+05 0.1220D+04 0.7275D+06 0.5820D+05 0.2000D+00 0.4932D+00 0.7806D-.01 0.5713D+00
 0.2670D+09 0.4950D+04 0.3300D+03 0.1830D+05 0.1220D+04 0.8730D+06 0.5820D+05 0.2000D+00 0.4932D+00 0.7806D-.01 0.5713D+00
 0.2670D+09 0.5775D+04 0.3300D+03 0.2135D+05 0.1220D+04 0.1019D+07 0.5820D+05 0.2000D+00 0.4932D+00 0.7806D-.01 0.5713D+00
 0.2670D+09 0.6600D+04 0.3300D+03 0.2440D+05 0.1220D+04 0.1164D+07 0.5820D+05 0.2000D+00 0.4932D+00 0.7806D-.01 0.5713D+00

 MODE NUMBER = 1
 NATURAL FREQUENCY = 0.4530D+00 RAD/S

ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = 85 POINTS

S	TENSION	OX1	OMEGAETA	PHI	P	Q	SIGMA
0.000D+00	0.7288D+01	0.3388D+01	0.6129D+03	0.0000D+00	0.0000D+00	0.0000D+00	0.5159D+01
0.4933D-03	0.6922D+01	0.3505D+01	0.5981D+03	0.2987D+00	-1347D-05	0.7365D-04	0.5159D+01
0.9866D-03	0.6570D+01	0.3593D+01	0.5829D+03	0.5900D+00	-8226D-05	0.2927D-03	0.5159D+01
0.1726D-02	0.6073D+01	0.3675D+01	0.5594D+03	0.1012D+01	-3986D-04	0.8834D-03	0.5159D+01
0.2465D-02	0.5612D+01	0.3707D+01	0.5355D+03	0.1417D+01	-1075D-03	0.1777D-02	0.5159D+01
0.3204D-02	0.5188D+01	0.3698D+01	0.5113D+03	0.1803D+01	-2208D-03	0.2959D-02	0.5159D+01
0.3943D-02	0.4802D+01	0.3656D+01	0.4873D+03	0.2172D+01	-3865D-03	0.4415D-02	0.5159D+01
0.4762D-02	0.4415D+01	0.3579D+01	0.4611D+03	0.2561D+01	-6362D-03	0.6329D-02	0.5159D+01
0.5581D-02	0.4070D+01	0.3477D+01	0.4354D+03	0.2928D+01	-9577D-03	0.8542D-02	0.5159D+01
0.7218D-02	0.3492D+01	0.3228D+01	0.3866D+03	0.3601D+01	-1812D-02	0.1378D-01	0.5159D+01
0.8856D-02	0.3043D+01	0.2944D+01	0.3415D+03	0.4197D+01	-2929D-02	0.2001D-01	0.5159D+01
0.1114D-01	0.2587D+01	0.2540D+01	0.2856D+03	0.4914D+01	-4847D-02	0.3012D-01	0.5159D+01
0.1342D-01	0.2280D+01	0.2159D+01	0.2375D+03	0.5511D+01	-7074D-02	0.4166D-01	0.5159D+01
0.1571D-01	0.2078D+01	0.1817D+01	0.1969D+03	0.6007D+01	-9483D-02	0.5440D-01	0.5159D+01
0.1790D-01	0.1951D+01	0.1531D+01	0.1640D+03	0.6402D+01	-1186D-01	0.6758D-01	0.5159D+01
0.2009D-01	0.1870D+01	0.1285D+01	0.1364D+03	0.6731D+01	-1422D-01	0.8155D-01	0.5159D+01
0.2228D-01	0.1821D+01	0.1075D+01	0.1132D+03	0.7005D+01	-1649D-01	0.9621D-01	0.5159D+01
0.2447D-01	0.1792D+01	0.8976D+00	0.9378D+02	0.7231D+01	-1864D-01	0.1114D+00	0.5159D+01
0.3157D-01	0.1778D+01	0.4915D+00	0.4932D+02	0.7730D+01	-2177D-01	0.1371D+00	0.5159D+01
0.3511D-01	0.1793D+01	0.3624D+00	0.3373D+02	0.7877D+01	-2440D-01	0.1637D+00	0.5159D+01
0.4135D-01	0.1832D+01	0.2121D+00	0.1517D+02	0.8029D+01	-2648D-01	0.1911D+00	0.5159D+01
0.4759D-01	0.1877D+01	0.1266D+00	0.4082D+01	0.8089D+01	-2881D-01	0.2404D+00	0.5159D+01
0.5472D-01	0.1928D+01	0.7200D-01	-0.3546D+01	0.8091D+01	-2974D-01	0.2906D+00	0.5159D+01
					-2950D-01	0.3483D+00	0.5159D+01

0. 6185D-01	0.	1978D+01	0.	42254D-01	-.	8465D+01	0.	8048D+01	-.	2819D-01	0.	4059D+00	0.	5159D+01
0. 7784D-01	0.	2092D+01	0.	1483D-01	-.	1536D+02	0.	7858D+01	-.	2252D-01	0.	5334D+00	0.	5159D+01
0. 9549D-01	0.	2214D+01	0.	9624D-02	-.	2036D+02	0.	7543D+01	-.	1276D-01	0.	6696D+00	0.	5159D+01
0. 1147D+00	0.	2345D+01	0.	1079D-01	-.	2511D+02	0.	7105D+01	0.	1509D-02	0.	8107D+00	0.	5159D+01
0. 1355D+00	0.	2480D+01	0.	1165D-01	-.	2950D+02	0.	6538D+01	0.	2098D-01	0.	9522D+00	0.	5159D+01
0. 1577D+00	0.	2615D+01	0.	1048D-01	-.	3414D+02	0.	5831D+01	0.	4668D-01	0.	1089D+01	0.	5159D+01
0. 1813D+00	0.	2741D+01	0.	9240D-02	-.	3853D+02	0.	4975D+01	0.	7982D-01	0.	1214D+01	0.	5159D+01
0. 2061D+00	0.	2849D+01	0.	7657D-02	-.	4249D+02	0.	3969D+01	0.	1214D+00	0.	1322D+01	0.	5159D+01
0. 2321D+00	0.	2925D+01	0.	5718D-02	-.	4577D+02	0.	2823D+01	0.	1724D+00	0.	1404D+01	0.	5159D+01
0. 2591D+00	0.	2954D+01	0.	3410D-02	-.	4810D+02	0.	1554D+01	0.	2333D+00	0.	1455D+01	0.	5159D+01
0. 2871D+00	0.	2915D+01	0.	7153D-03	-.	4919D+02	0.	1927D+00	0.	3039D+00	0.	1466D+01	0.	5159D+01
0. 3159D+00	0.	2789D+01	0.	2358D-02	-.	4874D+02	0.	1219D+01	0.	3834D+00	0.	1433D+01	0.	5159D+01
0. 3455D+00	0.	2555D+01	0.	5774D-02	-.	4647D+02	0.	2626D+01	0.	4695D+00	0.	1349D+01	0.	5159D+01
0. 3757D+00	0.	2192D+01	0.	9436D-02	-.	4214D+02	0.	3962D+01	0.	5587D+00	0.	1214D+01	0.	5159D+01
0. 4058D+00	0.	1681D+01	0.	1316D-01	-.	3560D+02	0.	5154D+01	0.	6402D+00	0.	1027D+01	0.	5159D+01
0. 4373D+00	0.	1041D+01	0.	1664D-01	-.	2688D+02	0.	6123D+01	0.	7231D+00	0.	7934D+00	0.	5159D+01
0. 4686D+00	0.	2650D+00	0.	1945D-01	-.	1622D+02	0.	6797D+01	0.	7839D+00	0.	5212D+00	0.	5159D+01
0. 5000D+00	0.	6060D+00	0.	2114D-01	-.	4191D+01	0.	7117D+01	0.	8203D+00	0.	2233D+00	0.	5159D+01
0. 5314D+00	0.	1521D+01	0.	2143D-01	0.	8424D+01	0.	7051D+01	0.	8273D+00	0.	8402D-01	0.	5159D+01
0. 5627D+00	0.	2419D+01	0.	2012D-01	0.	2069D+02	0.	6596D+01	0.	8028D+00	0.	3826D+00	0.	5159D+01
0. 6243D+00	0.	3236D+01	0.	1722D-01	0.	3162D+02	0.	5784D+01	0.	7494D+00	0.	6549D+00	0.	5159D+01
0. 6545D+00	0.	4455D+01	0.	1313D-01	0.	4040D+02	0.	4680D+01	0.	6730D+00	0.	8866D+00	0.	5159D+01
0. 6841D+00	0.	4822D+01	0.	3815D-02	0.	4655D+02	0.	3369D+01	0.	5823D+00	0.	1066D+01	0.	5159D+01
0. 7129D+00	0.	5036D+01	0.	3789D-03	0.	50910D+02	0.	1943D+01	0.	4858D+00	0.	1190D+01	0.	5159D+01
0. 7409D+00	0.	5121D+01	0.	3917D-02	0.	4977D+02	0.	9201D+00	0.	3041D+00	0.	1259D+01	0.	5159D+01
0. 7679D+00	0.	5106D+01	0.	6736D-02	0.	4705D+02	0.	2229D+01	0.	2275D+00	0.	1251D+01	0.	5159D+01
0. 7939D+00	0.	5019D+01	0.	8910D-02	0.	4322D+02	0.	3402D+01	0.	1629D+00	0.	1188D+01	0.	5159D+01
0. 8187D+00	0.	4885D+01	0.	1047D-01	0.	3868D+02	0.	4418D+01	0.	1101D+00	0.	1098D+01	0.	5159D+01
0. 8423D+00	0.	4723D+01	0.	1160D-01	0.	3377D+02	0.	5271D+01	0.	6833D+01	0.	9872D+00	0.	5159D+01
0. 8645D+00	0.	4559D+01	0.	1242D-01	0.	2873D+02	0.	5965D+01	0.	3629D-01	0.	8642D+00	0.	5159D+01
0. 8853D+00	0.	4377D+01	0.	1142D-01	0.	2406D+02	0.	6514D+01	0.	1231D-01	0.	7353D+00	0.	5159D+01
0. 9045D+00	0.	4210D+01	0.	1046D-01	0.	1916D+02	0.	6930D+01	0.	5039D-02	0.	6060D+00	0.	5159D+01
0. 9222D+00	0.	4055D+01	0.	1604D-01	0.	1405D+02	0.	7222D+01	0.	1681D-01	0.	4809D+00	0.	5159D+01
0. 9382D+00	0.	3914D+01	0.	4656D-01	0.	6772D+01	0.	7389D+01	0.	2567D-01	0.	3637D+00	0.	5159D+01
0. 9453D+00	0.	3851D+01	0.	7894D-01	0.	1478D+01	0.	7419D+01	0.	1231D-01	0.	7353D+00	0.	5159D+01
0. 9524D+00	0.	3788D+01	0.	1377D+00	0.	6757D+01	0.	7400D+01	0.	2643D-01	0.	2579D+00	0.	5159D+01
0. 9587D+00	0.	3733D+01	0.	2266D+00	0.	1861D+02	0.	7321D+01	0.	2597D-01	0.	2121D+00	0.	5159D+01
0. 9649D+00	0.	3683D+01	0.	3766D+00	0.	3801D+02	0.	7144D+01	0.	2413D-01	0.	1672D+00	0.	5159D+01
0. 9684D+00	0.	3662D+01	0.	5007D+00	0.	5397D+02	0.	6981D+01	0.	2236D-01	0.	1424D+00	0.	5159D+01
0. 9725D+00	0.	3655D+01	0.	6629D+00	0.	7359D+02	0.	6755D+01	0.	2004D-01	0.	1184D+00	0.	5159D+01
0. 9755D+00	0.	3669D+01	0.	8695D+01	0.	9789D+02	0.	6451D+01	0.	1722D-01	0.	9552D-01	0.	5159D+01
0. 9777D+00	0.	3697D+01	0.	1022D+01	0.	1164D+03	0.	6216D+01	0.	1527D-01	0.	820D-01	0.	5159D+01
0. 9795D+00	0.	3747D+01	0.	1196D+01	0.	1381D+03	0.	5937D+01	0.	1318D-01	0.	6908D-01	0.	5159D+01
0. 9821D+00	0.	3830D+01	0.	1391D+01	0.	1634D+03	0.	5607D+01	0.	1101D-01	0.	5685D-01	0.	5159D+01
0. 9843D+00	0.	3960D+01	0.	1605D+01	0.	1927D+03	0.	5217D+01	0.	8811D-02	0.	4542D-01	0.	5159D+01
0. 9860D+00	0.	4107D+01	0.	1781D+01	0.	2185D+03	0.	4865D+01	0.	7120D-02	0.	3711D-01	0.	5159D+01
0. 9877D+00	0.	4307D+01	0.	1958D+01	0.	2469D+03	0.	4466D+01	0.	5505D-02	0.	2942D-01	0.	5159D+01
0. 9894D+00	0.	4574D+01	0.	2125D+01	0.	2780D+03	0.	4016D+01	0.	4016D-02	0.	2243D-01	0.	5159D+01
0. 9911D+00	0.	4927D+01	0.	2262D+01	0.	3112D+03	0.	3512D+01	0.	2708D-02	0.	1621D-01	0.	5159D+01
0. 9928D+00	0.	5360D+01	0.	2340D+01	0.	3444D+03	0.	2975D+01	0.	1671D-02	0.	1106D-01	0.	5159D+01
0. 9936D+00	0.	5617D+01	0.	2349D+01	0.	3612D+03	0.	2686D+01	0.	1242D-02	0.	8810D-02	0.	5159D+01
0. 9944D+00	0.	5901D+01	0.	2333D+01	0.	3780D+03	0.	2384D+01	0.	8785D-03	0.	6784D-02	0.	5159D+01
0. 9952D+00	0.	6215D+01	0.	2275D+01	0.	3944D+03	0.	2067D+01	0.	5821D-03	0.	4998D-02	0.	5159D+01
0. 9961D+00	0.	6557D+01	0.	2117D+01	0.	4101D+03	0.	1738D+01	0.	3525D-03	0.	3465D-02	0.	5159D+01

FILE : INPLA1	DATA	A1	VM/SP CONVERSATIONAL MONITOR SYSTEM	PAGE 005
0.9968D+00	- .6890D+01	0.2044D+01	- .4235D+03	0.1430D+01
0.9975D+00	- .7241D+01	0.1861D+01	- .4357D+03	0.1112D+01
0.9983D+00	- .7608D+01	0.1619D+01	- .4464D+03	0.7865D+00
0.9990D+00	- .7985D+01	0.1310D+01	- .4551D+03	0.4534D+00
0.9995D+00	- .8238D+01	0.1062D+01	- .4597D+03	0.2276D+00
0.1000D+01	- .8489D+01	0.7774D+00	- .4629D+03	0.0000D+00
MAXIMUM ESTIMATED ERROR BY COMPONENTS				
O.4394D-01	O.3839D-01	O.1479D+01	O.3383D-01	O.4257D-02
O.4394D-01	O.3839D-01	O.1479D+01	O.3383D-01	O.4257D-02
			O.2797D-02	O.1922D-01

V.2.1.3 INPLATA DATA A

1 85 0 4530D+00 0.2145D-18 0.7931D+00
 0.000000D+00 0.728757D+01 0.338805D+01 0.612872D+03 0.000000D+00 0.000000D+00 0.515949D+01 0.103000D+01
 0.493318D-03 0.692164D+01 0.350534D+01 0.598143D+03 0.298708D+00 -1.34651D-05 0.736537D-04 0.515949D+01 0.103031D+01
 0.9866336D-03 0.657017D+01 0.3595269D+01 0.582905D+03 0.590024D+00 -822562D-05 0.292688D-03 0.515949D+01 0.103062D+01
 0.172564D-02 0.607276D+01 0.367503D+01 0.559423D+03 0.101211D+01 -398629D-04 0.883339D-03 0.515949D+01 0.103108D+01
 0.246464D-02 0.561199D+01 0.370727D+01 0.535464D+03 0.141668D+01 -107493D-03 0.177712D-02 0.515949D+01 0.103154D+01
 0.320364D-02 0.518848D+01 0.369826D+01 0.513346D+03 0.180348D+01 -220839D-03 0.295916D-02 0.515949D+01 0.103199D+01
 0.394265D-02 0.480193D+01 0.365670D+01 0.487327D+01 0.258409D+01 -386466D-03 0.441470D-02 0.515949D+01 0.103244D+01
 0.476160D-02 0.441516D+01 0.357890D+01 0.461081D+03 0.256084D+01 -636221D-03 0.632921D-02 0.515949D+01 0.103293D+01
 0.558056D-02 0.406997D+01 0.347733D+01 0.435431D+03 0.292794D+01 -957728D-03 0.854239D-02 0.515949D+01 0.103342D+01
 0.721847D-02 0.349171D+01 0.322759D+01 0.386577D+03 0.360113D+01 -181188D-02 0.137841D-01 0.515949D+01 0.103436D+01
 0.885637D-02 0.304333D+01 0.294413D+01 0.341506D+03 0.419739D+01 -292912D-02 0.200148D-01 0.515949D+01 0.103527D+01
 0.111404D-01 0.258720D+01 0.254049D+01 0.285551D+03 0.491350D+01 -484712D-02 0.301238D-01 0.515949D+01 0.103650D+01
 0.134244D-01 0.228022D+01 0.215866D+01 0.237538D+02 0.551087D+01 -707438D-02 0.416599D-01 0.515949D+01 0.103766D+01
 0.157084D-01 0.207772D+01 0.18162D+01 0.196881D+03 0.600698D+01 -186390D-01 0.543989D-01 0.515949D+01 0.103878D+01
 0.178993D-01 0.195136D+01 0.153056D+01 0.164019D+03 0.640223D+01 -118605D-01 0.675772D-01 0.515949D+01 0.103981D+01
 0.200901D-01 0.187047D+01 0.128457D+01 0.136367D+03 0.673137D+01 -142179D-01 0.815518D-01 0.515949D+01 0.104080D+01
 0.222809D-01 0.182072D+01 0.107510D+01 0.113178D+03 0.700472D+01 -164926D-01 0.962054D-01 0.515949D+01 0.104177D+01
 0.244717D-01 0.179243D+01 0.897635D+00 0.937832D+02 0.723142D+01 -186390D-01 0.111436D+00 0.515949D+01 0.104272D+01
 0.280184D-01 0.177747D+01 0.666514D+00 0.688853D+02 0.751989D+01 -217693D-01 0.137085D+00 0.515949D+01 0.104421D+01
 0.315651D-01 0.177787D+01 0.491747D+00 0.493172D+02 0.774952D+01 -294972D-01 0.348261D+00 0.515949D+01 0.104567D+01
 0.351118D-01 0.179287D+01 0.3622369D+00 0.3377278D+02 0.787677D+01 -264753D-01 0.163721D+00 0.515949D+01 0.104567D+01
 0.413491D-01 0.183230D+01 0.212072D+00 0.151730D+02 0.802928D+01 -288102D-01 0.240396D+00 0.515949D+01 0.104711D+01
 0.475865D-01 0.187684D+01 0.126611D+00 0.408233D+01 0.808933D+01 -297419D-01 0.290557D+00 0.515949D+01 0.104959D+01
 0.527166D-01 0.192780D+01 0.719961D-01 -354582D+01 0.809124D+01 -294972D-01 0.348261D+00 0.515949D+01 0.105487D+01
 0.618467D-01 0.197849D+01 0.425426D-01 -84646499D+01 0.804842D+01 -281878D-01 0.405899D+00 0.515949D+01 0.105770D+01
 0.778360D-01 0.209178D+01 0.148274D-01 -153596D+02 0.785795D+01 -225172D-01 0.533362D+00 0.515949D+01 0.106409D+01
 0.954915D-01 0.221450D+01 0.962455D+02 -203599D+02 0.754263D+01 -127626D-01 0.669598D+00 0.515949D+01 0.107126D+01
 0.114743D+00 0.234503D+01 0.107887D-01 -251053D+02 0.710500D+01 0.150905D-02 0.810699D+00 0.515949D+01 0.107926D+01
 0.135516D+00 0.248031D+01 0.116481D-01 -295045D+02 0.653779D+01 0.209842D-01 0.952151D+00 0.515949D+01 0.108808D+01
 0.157726D+00 0.261480D+01 0.104831D-01 -341440D+02 0.583097D+01 0.466757D-01 0.108865D+01 0.515949D+01 0.108775D+01
 0.181288D+00 0.274132D+01 0.923975D-02 -385303D+02 0.497480D+01 0.798173D-01 0.121412D+01 0.515949D+01 0.110829D+01
 0.206107D+00 0.284916D+01 0.765568D-02 -424885D+02 0.396939D+01 0.12142427D+00 0.132179D+01 0.515949D+01 0.111973D+01
 0.232087D+00 0.292533D+01 0.571707D-02 -457683D+02 0.282299D+01 0.172403D+00 0.140450D+01 0.515949D+01 0.113208D+01
 0.2591123D+00 0.295367D+01 0.341046D-02 -480976D+02 0.155406D+01 0.233250D+00 0.145498D+01 0.515949D+01 0.114537D+01
 0.287110D+00 0.291532D+01 0.715310D-03 -4911875D+02 0.192697D+00 0.303897D+00 0.146563D+01 0.515949D+01 0.115961D+01
 0.319538D+00 0.278938D+01 -235841D-02 -487404D+02 -121884D+01 0.3833385D+00 0.1465261D+01 0.515949D+01 0.124453D+01
 0.345492D+00 0.255450D+01 -577424D-02 -464718D+02 -262579D+01 0.469543D+00 0.1434921D+01 0.515949D+01 0.117480D+01
 0.375655D+00 0.223208D+01 0.219156D+01 -943648D-02 -4214143D+02 -396220D+01 0.558696D+00 0.1213848D+01 0.515949D+01 0.128335D+01
 0.406309D+00 0.168756D+01 -131661D-01 -356032D+02 -515379D+01 0.645507D+00 0.102714D+01 0.515949D+01 0.120799D+01
 0.437333D+00 0.104065D+01 -166429D-01 -172222D-01 0.316213D+02 -5784423D+01 0.749351D+00 0.145622D+01 0.515949D+01 0.122589D+01
 0.468605D+00 0.265030D+00 -194541D-01 -162213D+02 -679684D+01 0.468033D+01 0.673016D+00 0.515949D+01 0.126375D+01
 0.500000D+00 -605959D+00 -211442D-01 -419072D+01 -711726D+01 0.8203348D+00 0.223303D+00 0.515949D+01 0.136011D+01
 0.654508D+00 -445547D+01 -845816D-02 0.4655471D+02 -336900D+01 0.5822255D+00 0.106601D+01 0.515949D+01 0.137757D+01
 0.684062D+00 -482165D+01 -214257D-01 0.842409D+01 -705081B+01 0.827288D+00 -840288D-01 0.119041D+01 0.515949D+01 0.130307D+01
 0.5936910D+00 -241865D+01 -2011837D-01 0.206868D+02 -6595563D+01 0.802829D+00 -382637D+00 0.515949D+01 0.132262D+01
 0.624345D+00 -392492D+01 -131258D-01 0.404021D+02 -468033D+01 0.783851D+00 0.654871D+00 0.515949D+01 0.134172D+01
 0.654508D+00 -510595D+01 0.673648D-02 0.470525D+02 -422889D+01 0.886601D+00 0.515949D+01 0.142332D+01
 0.793893D+00 -501889D+01 0.890961D-02 0.340155D+01 0.162859D+00 -118841D+01 0.119041D+01 0.515949D+01 0.143624D+01
 0.818712D+00 -488453D+01 0.104710D-01 0.337676D+02 0.441778D+01 0.110093D+00 -109772D+01 0.515949D+01 0.144804D+01
 0.842274D+00 -472291D+01 0.115974D-01 0.337676D+02 0.527123D+01 0.683328D-01 0.987168D+00 0.515949D+01 0.146847D+01
 0.864484D+00 -455003D+01 0.124228D-01 0.287321D+02 0.596529D+01 0.3628662D-01 0.864213D+00 0.515949D+01 0.147722D+01

0. 885257D+00 - .437654D+01 0. 114241D-01 0. 240564D+02 0. 651358D+01 0. 123055D-01 - .735335D+00 0. 515949D+01 0. 148508D+01
 0. 904508D+00 - .421016D+01 0. 104649D-01 0. 191555D+02 0. 692951D+01 - .503949D-02 - .606038D+00 0. 515949D+01 0. 149209D+01
 0. 922164D+00 - .405543D+01 0. 160397D-01 0. 140515D+02 0. 722267D+01 - .168061D-01 - .480870D+00 0. 515949D+01 0. 149830D+01
 0. 938153D+00 - .391415D+01 0. 465641D-01 0. 677188D+01 0. 738914D+01 - .2383336D-01 - .363728D+00 0. 515949D+01 0. 150377D+01
 0. 945283D+00 - .385122D+01 0. 789381D-01 0. 147820D+01 0. 741855D+01 - .256694D-01 - .310805D+00 0. 515949D+01 0. 150616D+01
 0. 952414D+00 - .378806D+01 0. 137651D+00 - .675711D+01 0. 739973D+01 - .264323D-01 - .257905D+00 0. 515949D+01 0. 150854D+01
 0. 958651D+00 - .373282D+01 0. 226620D+00 - .186116D+02 0. 732062D+01 - .259690D-01 - .212056D+00 0. 515949D+01 0. 151062D+01
 0. 964888D+00 - .368289D+01 0. 376564D+00 - .380109D+02 0. 714404D+01 - .241340D-01 - .167200D+00 0. 515949D+01 0. 151272D+01
 0. 968435D+00 - .366240D+01 0. 500744D+00 - .539708D+02 0. 698091D+01 - .223647D-01 - .142417D+00 0. 515949D+01 0. 151393D+01
 0. 971982D+00 - .365455D+01 0. 662871D+00 - .735866D+02 0. 675469D+01 - .2003396D-01 - .118437D+00 0. 515949D+01 0. 151517D+01
 0. 975528D+00 - .366904D+01 0. 869504D+00 - .978872D+02 0. 645066D+01 - .172237D-01 - .955170D-01 0. 515949D+01 0. 151645D+01
 0. 977719D+00 - .369655D+01 0. 102223D+01 - .116410D+03 0. 621590D+01 - .152693D-01 - .819987D-01 0. 515949D+01 0. 151727D+01
 0. 979910D+00 - .374682D+01 0. 119583D+01 - .138111D+03 0. 593370D+01 - .131846D-01 - .690781D-01 0. 515949D+01 0. 151812D+01
 0. 982101D+00 - .382974D+01 0. 139093D+01 - .1633993D+03 0. 560677D+01 - .110115D-01 - .5668494D-01 0. 515949D+01 0. 151900D+01
 0. 984292D+00 - .395994D+01 0. 160488D+01 - .192653D+03 0. 521671D+01 - .881120D-02 - .454185D-02 0. 515949D+01 0. 151992D+01
 0. 986005D+00 - .410676D+01 0. 178112D+01 - .218470D+03 0. 486459D+01 - .711986D-02 - .371100D-01 0. 515949D+01 0. 152067D+01
 0. 987718D+00 - .430666D+01 0. 195814D+01 - .2469938D+03 0. 446596D+01 - .550473D-02 - .294243D-01 0. 515949D+01 0. 152146D+01
 0. 989431D+00 - .457444D+01 0. 212462D+01 - .277957D+03 0. 401639D+01 - .401631D-02 - .224318D-01 0. 515949D+01 0. 152228D+01
 0. 991144D+00 - .492222D+01 0. 226196D+01 - .3111189D+03 0. 351179D+01 - .270761D-02 - .162108D-01 0. 515949D+01 0. 152315D+01
 0. 992782D+00 - .5356044D+01 0. 233982D+01 - .344390D+03 0. 297487D+01 - .167073D-02 - .110640D-01 0. 515949D+01 0. 152402D+01
 0. 993600D+00 - .561661D+01 0. 234871D+01 - .361225D+03 0. 268610D+01 - .124198D-02 - .880992D-02 0. 515949D+01 0. 152447D+01
 0. 994419D+00 - .590121D+01 0. 232957D+01 - .377959D+03 0. 238359D+01 - .878538D-03 - .678446D-02 0. 515949D+01 0. 152493D+01
 0. 995238D+00 - .621501D+01 0. 227513D+01 - .394359D+03 0. 206732D+01 - .582125D-03 - .499846D-02 0. 515949D+01 0. 152541D+01
 0. 996057D+00 - .655749D+01 0. 217737D+01 - .410113D+03 0. 173789D+01 - .352537D-03 - .346550D-02 0. 515949D+01 0. 152589D+01
 0. 996796D+00 - .688969D+01 0. 204441D+01 - .423476D+03 0. 1429888D+01 - .2008821D-03 - .230950D-02 0. 515949D+01 0. 152633D+01
 0. 997535D+00 - .724140D+01 0. 186105D+01 - .435694D+03 0. 111242D+01 - .973845D-04 - .137845D-02 0. 515949D+01 0. 152678D+01
 0. 998274D+00 - .760847D+01 0. 161893D+01 - .446390D+03 0. 786485D+00 - .359419D-04 - .680766D-03 0. 515949D+01 0. 152724D+01
 0. 999013D+00 - .798508D+01 0. 130959D+01 - .455131D+03 0. 4533373D+00 - .735168D-05 - .224000D-03 0. 515949D+01 0. 152770D+01
 0. 999506D+00 - .823818D+01 0. 106167D+01 - .459657D+03 0. 227649D+00 - .118142D-05 - .561458D-04 0. 515949D+01 0. 152801D+01
 0. 100000D+01 - .848899D+01 0. 7777371D+00 - .462934D+03 0. 000000D+00 0. 000000D+00 0. 423855D-19 0. 515949D+01 0. 152832D+01

Chapter VI

VI.1 Input-Output Manual for the Out-of-Plane Forced Linear Dynamic Problem Solution Program, RCFORC1

This program is written in FORTRAN 77 and allows the interactive solution of the out-of-plane forced undamped linear dynamic problem for a compliant riser with a planar static configuration without torsion for given frequency and excitation at the top as described in [4]. Excitation at the top is allowed. The output of this program can be used as input to RCLINDY3 to solve the out-of-plane linear dynamic problem with homogeneous boundary conditions (out-of-plane eigenproblem).

Before executing the program the user must make sure that

- o Devices five and six correspond to input/output from the terminal
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data preprocessing program RCINPUT. This file has a logical record length of 80 characters.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.
- o Device eleven must be associated with an output file with logical record length of 117 characters. This file contains the converged solution of the problem in a concise form appropriate for use in the preparation of illustrations and/or as an input to RCLINDY3 from device twelve in a subsequent execution of this program.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The values of the mean current velocity in the x direction and the maximum non-dimensional static effective tension are printed on the terminal. If the number of static discretization points is less than four or greater than the value of MNP (maximum number of allowable points), the program halts execution.

Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop, see [3]. Then the user is asked for the excitation frequency in (rad/s). In order to get meaningful results, this frequency should be close to one of the out-of-plane natural frequencies of the compliant riser. Next he is asked for the r and β amplitudes of excitation at the top of the riser. Appropriate values for these amplitudes range from one to zero. Then the user is asked to supply the tolerance for convergence of iterations. Suggested values are the excitation amplitudes or 1/10 of these values. However, at the end of the run the results should be checked, since the maximum of the resulting variables will be evaluated within this tolerance. Then the user is asked to input the mode number corresponding to the frequency of excitation.

Subsequently, the program enters NAG subroutine D02GBF, which provides the numerical solution to the problem. The manual of this subroutine is included in Chapter X. The input value of IFAIL is 111, indicating that

the soft failure option with error message and monitoring information printing has been selected. Error messages are directed to device six. Once the execution of D02GBF is completed, the current value of IFAIL is printed on the terminal. For an explanation of this parameter, see NAG manual for D02GBF, Chapter X.

Only if IFAIL=0 or IFAIL=4, the execution of the program continues with printout of the final solution of the problem. Otherwise, the program stops. When IFAIL \neq 0, the user should consult the manual quoted above for interpretation of the results and a possible course of action. If IFAIL=0 or 4, the solution is printed in the file associated with device eleven. The following quantities are printed:

MODE, NP, SIGMAD = Mode number, number of points at which converged solution is available and dimensional natural frequency (in rad/s)

with FORMAT (1X, I2, 1X, I3, 1X, D10.4) and

for I=1, NP: The non-dimensional

s , Q_1^n , Ω_1^ζ , Ω_1^ξ , θ_1 , β , r , Σ

with FORMAT (8(1X, D12.6))

The data written in the file associated with device eleven can be used as input to program RCLINDY3.

VI.2 Sample Run, Input and Output

VI.2.1 Zero Current Excitation

VI.2.1.1 Interactive Session

INPUT

Device 5 = TERMINAL

Device 8 = RCJOA01 DATA A*

Device 10 = RC2DZERR DATA A

OUTPUT

Device 6 = TERMINAL

Device 11 = OUTZERIE

R; T=0.01 17:12:15

RCFORC1 rcjoinat rc2dzerr outzer1e
FI 8 DISK RCJOAO1 DATA A (RECFM FB LRECL 80 BLKSIZE 800
FI 10 DISK RC2DZERR DATA A (RECFM FB LRECL 132 BLKSIZE 1320
FI 11 DISK OUTZER1E DATA A (RECFM FB LRECL 117 BLKSIZE 1170
GLOBAL XTLLIB VFORTLIB CMSLIB NAG1 NAG2
LOAD RCFORC1 ! START

EXECUTION BEGINS...

MNP = 15.1

2-D STATIC SOLUTION FROM DEVICE 10

NP = 54

MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S .VM =0.000000D+00
2-D STATIC SOLUTION SUCCESSFULLY READ

MAXIMUM STATIC EFFECTIVE TENSION/WA*L = 0.5547D+00

DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS

IF YES INPUT 1 , IF NO INPUT 0

?

O INPUT EXCITATION FREQUENCY IN RAD/S

?

.049 INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR R

?

-1 INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR B

?

O. INPUT TOLERANCE FOR CONVERGENCE OF ITERATIONS
SUGGESTED VALUE IS THE EXCITATION AMPLITUDE OR 1/10 OF THE EXCITATION AMPLITUDE

?

1 INPUT MODE NUMBER CORRESPONDING TO FREQUENCY OF EXCITATION

?

DO2GBF MONITORING INFORMATION
NUMBER OF POINTS IN CURRENT MESH = 54

CORRECTION NUMBER 0 ESTIMATED MAXIMUM ERROR = 5.99D-01
ESTIMATED ERROR BY COMPONENTS
7.76D-02 1.28D-02 5.99D-01 4.95D-02 4.92D-02 1.20D-02
NUMBER OF POINTS IN CURRENT MESH = 54

CORRECTION NUMBER 1 ESTIMATED MAXIMUM ERROR = 4.72D-02
ESTIMATED ERROR BY COMPONENTS
3.84D-03 1.61D-03 4.72D-02 4.11D-03 2.75D-03 7.24D-04
I FAIL = 0
R; T=1.04/1.27 17:13:50
cp spool console stop close

VI.2.1.2 RC2DZERR DATA A

DIMENSIONAL RESULTS AT NP= 54 POINTS

S(M)	S*(M)	X(M)	Y(M)	V(M/S)	T(KN)	QX1(KN)	RETA(M)	META(KN.M)	PHI(DEG)	P(KN)
0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	0.0000D+00	-1149D+00	0.2742D-01	-9841D+01	-6707D+00	0.9000D+02	0.2904D+02
0.9081D-01	-3909D-07	0.4201D-03	0.9080D-01	0.0000D+00	-1149D+00	0.2636D-01	-9761D+01	-6762D+00	0.8947D+02	0.2905D+02
0.3628D+00	-1561D-06	0.6761D-02	0.3628D+00	0.0000D+00	-1148D+00	0.2314D-01	-9535D+01	-6922D+00	0.8785D+02	0.2909D+02
0.8150D+00	-3503D-06	0.3454D-01	0.8140D+00	0.0000D+00	-1145D+00	0.1768D-01	-9206D+01	-7174D+00	0.8509D+02	0.2916D+02
0.1445D+01	-6197D-06	0.1103D+00	0.1440D+01	0.0000D+00	-1136D+00	0.9906D-01	-8809D+01	-7492D+00	0.8107D+02	0.2925D+02
0.2252D+01	-9600D-06	0.2723D+00	0.2229D+01	0.0000D+00	-1118D+00	0.1853D-03	-8419D+01	-7840D+00	0.7570D+02	0.2937D+02
0.3230D+01	-1363D-05	0.5692D+00	0.3161D+01	0.0000D+00	-1083D+00	0.1242D-01	-8059D+01	-7166D+00	0.6889D+02	0.2951D+02
0.4377D+01	-1816D-05	0.1058D+01	0.4197D+01	0.0000D+00	-1025D+00	-2633D-01	-7777D+01	-7425D+00	0.6058D+02	0.2967D+02
0.5687D+01	-2297D-05	0.1795D+01	0.5279D+01	0.0000D+00	-9357D-01	-4036D-01	-7650D+01	-6470D+00	0.5082D+02	0.2983D+02
0.7156D+01	-2778D-05	0.2826D+01	0.6321D+01	0.0000D+00	-8135D-01	-5463D-01	-7728D+01	-6405D+00	0.3985D+02	0.3000D+02
0.7966D+01	-3014D-05	0.3474D+01	0.6807D+01	0.0000D+00	-7393D-01	-6080D-01	-7903D+01	-5219D+00	0.3390D+02	0.3007D+02
0.8777D+01	-3227D-05	0.4169D+01	0.7225D+01	0.0000D+00	-6631D-01	-6586D-01	-8182D+01	-5041D+00	0.2812D+02	0.3014D+02
0.1054D+02	-3614D-05	0.5800D+01	0.7893D+01	0.0000D+00	-5069D-01	-7263D-01	-1033D+02	-3194D+00	0.1686D+02	0.3026D+02
0.1118D+02	-3729D-05	0.6413D+01	0.8060D+01	0.0000D+00	-4604D-01	-7360D-01	-1210D+02	-2728D+00	0.1359D+02	0.3028D+02
0.1181D+02	-3833D-05	0.7034D+01	0.8194D+01	0.0000D+00	-4210D-01	-7390D-01	-1460D+02	-2260D+00	0.1084D+02	0.3031D+02
0.1245D+02	-3930D-05	0.7660D+01	0.8301D+01	0.0000D+00	-3891D-01	-7364D-01	-1843D+02	-1791D+00	0.8608D+01	0.3033D+02
0.1448D+02	-4206D-05	0.9683D+01	0.8528D+01	0.0000D+00	-3354D-01	-7007D-01	-1020D+03	-3220D+00	0.4839D+01	0.3037D+02
0.1664D+02	-4481D-05	0.1183D+02	0.8722D+01	0.0000D+00	-3468D-01	-6284D-01	-2951D+02	-1118D+00	0.6433D+01	0.3104D+02
0.1891D+02	-4801D-05	0.1407D+02	0.9098D+01	0.0000D+00	-4058D-01	-5161D-01	-1361D+02	-2425D+00	0.1350D+02	0.3044D+02
0.2127D+02	-5193D-05	0.1631D+02	0.9881D+01	0.0000D+00	-4756D-01	-3556D-01	-9519D+01	-1043D+00	0.3467D+02	0.3055D+02
0.2375D+02	-5648D-05	0.1785D+02	0.1125D+02	0.0000D+00	-5081D-01	-1541D-01	-8050D+01	-4099D+00	0.4215D+02	0.3075D+02
0.2631D+02	-6114D-05	0.1993D+02	0.1323D+02	0.0000D+00	-4663D-01	0.5200D-02	0.7811D+01	0.4225D+00	0.6077D+02	0.3104D+02
0.2893D+02	-6516D-05	0.2082D+02	0.1569D+02	0.0000D+00	-3507D-01	0.2124D-01	0.8540D+01	0.3864D+00	0.7937D+02	0.3141D+02
0.3028D+02	-6673D-05	0.2097D+02	0.1702D+02	0.0000D+00	-2755D-01	0.2641D-01	0.9317D+01	0.3542D+00	0.1350D+02	0.3044D+02
0.3162D+02	-6792D-05	0.2092D+02	0.1837D+02	0.0000D+00	-1978D-01	0.2952D-01	0.1043D+02	0.3161D+00	0.8803D+02	0.3161D+02
0.3436D+02	-6919D-05	0.2032D+02	0.2104D+02	0.0000D+00	-4991D-02	0.3066D-01	0.1419D+02	0.2325D+00	0.1089D+03	0.3181D+02
0.3714D+02	-6908D-05	0.1919D+02	0.2358D+02	0.0000D+00	0.7188D-02	0.2722D-01	0.2182D+02	0.1513D+00	0.1182D+03	0.3222D+02
0.3995D+02	-6789D-05	0.1774D+02	0.2598D+02	0.0000D+00	0.1666D-01	0.2180D-01	0.4021D+02	0.8206D-01	0.1238D+03	0.3226D+02
0.4277D+02	-6566D-05	0.1610D+02	0.2829D+02	0.0000D+00	0.2428D-01	0.1608D-01	0.1157D+03	0.2852D-01	0.1265D+03	0.3331D+02
0.4561D+02	-6272D-05	0.1440D+02	0.3056D+02	0.0000D+00	0.3101D-01	0.1093D-01	0.3446D+02	-9577D-02	0.1269D+03	0.3364D+02
0.4844D+02	-5910D-05	0.1272D+02	0.3283D+02	0.0000D+00	0.3748D-01	0.6708D-02	0.9623D-02	-3429D-01	0.1258D+03	0.3398D+02
0.5125D+02	-5481D-05	0.1112D+02	0.3514D+02	0.0000D+00	0.4404D-01	0.3493D-02	0.4840D-01	-4840D-01	0.1237D+03	0.3432D+02
0.5403D+02	-4987D-05	0.9626D+01	0.3748D+02	0.0000D+00	0.5079D-01	0.1217D-02	0.6027D+02	-5475D-01	0.1212D+03	0.3467D+02
0.5677D+02	-4430D-05	0.8261D+01	0.3986D+02	0.0000D+00	0.5577D-01	0.2596D-03	0.5903D+02	-5590D-01	0.1185D+03	0.3503D+02
0.5946D+02	-3814D-05	0.7031D+01	0.4223D+02	0.0000D+00	0.6472D-01	0.1105D-02	0.6118D+02	-5394D-01	0.1160D+03	0.3538D+02
0.6209D+02	-3143D-05	0.5931D+01	0.4464D+02	0.0000D+00	0.7174D-01	0.1478D-02	0.6540D+02	-5046D-01	0.1136D+03	0.3574D+02
0.6464D+02	-2424D-05	0.4955D+01	0.4699D+02	0.0000D+00	0.7869D-01	0.1565D-02	0.7048D+02	-4849D-01	0.1237D+03	0.3609D+02
0.6711D+02	-1665D-05	0.4093D+01	0.5778D+02	0.0000D+00	0.1022D+00	0.2089D-02	0.7696D-02	-4659D-01	0.1144D+03	0.3609D+02
0.6948D+02	-8752D-06	0.3334D+01	0.5964D+02	0.0000D+00	0.8549D-01	0.1274D-02	0.7656D-02	-4310D-01	0.1018D+03	0.3643D+02
0.7175D+02	-6508D-07	0.2671D+01	0.5373D+02	0.0000D+00	0.9209D-01	0.8224D-03	0.8136D+02	-4056D-01	0.1078D-03	0.3677D+02
0.7391D+02	-0.7548D-06	0.2096D+01	0.5581D+02	0.0000D+00	0.1045D+00	0.7960D-03	0.8375D+02	-3941D-01	0.1062D+03	0.3703D+02
0.7594D+02	0.1573D-05	0.1603D+01	0.5778D+02	0.0000D+00	0.1102D+00	0.1121D-01	0.5592D+02	-3853D-01	0.1047D+03	0.3740D+02
0.7785D+02	0.2379D-05	0.1190D+01	0.5964D+02	0.0000D+00	0.1163D+00	0.1132D-01	0.4288D-01	-4288D-01	0.1033D+03	0.3769D+02
0.7962D+02	0.3160D-05	0.8536D+00	0.6138D+02	0.0000D+00	0.1205D+00	0.6123D-02	0.6737D+02	-4898D-01	0.1018D+03	0.3777D+02
0.8124D+02	0.3905D-05	0.5879D+00	0.6298D+02	0.0000D+00	0.1250D+00	0.8595D-02	0.6234D+02	-6617D-01	0.1002D+03	0.3823D+02
0.8314D+02	0.4603D-05	0.3847D+00	0.6443D+02	0.0000D+00	0.1290D+00	0.1121D-01	0.592D+02	-8853D-01	0.9722D+02	0.3869D+02
0.8402D+02	0.5245D-05	0.2356D+00	0.6573D+02	0.0000D+00	0.1325D+00	0.1397D-01	0.5144D+02	-4288D-01	0.1033D+03	0.3769D+02
0.8516D+02	0.5820D-05	0.1324D+00	0.6687D+02	0.0000D+00	0.1355D+00	0.1678D-01	0.5982D+02	-4898D-01	0.1018D+03	0.3888D+02
0.8614D+02	0.6321D-05	0.6612D-01	0.6785D+02	0.0000D+00	0.1379D+00	0.1953D-01	0.4724D+02	-1123D+00	0.9449D+02	0.3904D+02
0.8695D+02	0.6741D-05	0.2791D-01	0.6865D+02	0.0000D+00	0.1399D+00	0.2207D-01	0.4088D+02	-1614D+00	0.9216D+02	0.3931D+02



0.8758D+02 0.7073D-05 0.9048D-02 0.6929D+02 0.0000D+00 0.1413D+00 0.2425D-01 -.3864D+02 -.1708D+00 0.9126D+02 0.3940D+02
0.8803D+02 0.7313D-05 0.1819D-02 0.6974D+02 0.0000D+00 0.1424D+00 0.2592D-01 -.3704D+02 -.1782D+00 0.9057D+02 0.3947D+02
0.8830D+02 0.7458D-05 0.1150D-03 0.7001D+02 0.0000D+00 0.1430D+00 0.2698D-01 -.3608D+02 -.1829D+00 0.9014D+02 0.3951D+02
0.8839D+02 0.7507D-05 0.0000D+00 0.7010D+02 0.0000D+00 0.1432D+00 0.2734D-01 -.3576D+02 -.1846D+00 0.9000D+02 0.3952D+02

OUTZERIE DATA A

1 54 0.4900D-01
 0.000000D+00 0.142261D+01 - 160160D+00 - 136137D+02 0.000000D+00 0.000000D+00 0.000000D+00 0.416474D+01
 0.102730D-02 0.143388D+01 - 158951D+00 - 136785D+02 - 140192D-01 - 989260D-04 0.719546D-05 0.416474D+01
 0.410499D-02 0.1467179D+01 - 155146D+00 - 138713D+02 - 156413D-01 0.412048D-03 0.115440D-03 0.416474D+01
 0.922042D-02 0.152451D+01 - 148215D+00 - 14852D+02 - 128083D+00 0.409655D-02 0.586765D-03 0.416474D+01
 0.163526D-01 0.160403D+01 - 137290D+00 - 146022D+02 - 230108D+00 0.456439D-01 0.186297D-02 0.416474D+01
 0.254721D-01 0.170548D+01 - 121216D+00 - 150843D+02 - 362907D+00 0.422806D-01 0.456594D-02 0.416474D+01
 0.365416D-01 0.182610D+01 - 990766D-01 - 155488D+02 - 524672D+00 0.939300D-01 0.948019D-02 0.416474D+01
 0.495156D-01 0.195975D+01 - 697622D-01 - 158754D+02 - 709046D+00 0.182533D+00 0.174938D-01 0.416474D+01
 0.6433406D-01 0.209552D+01 - 324159D-01 - 158693D+01 - 902589D+00 0.319317D+00 0.294693D-01 0.416474D+01
 0.809559D-01 0.221724D+01 0.122805D-01 - 153117D+02 - 108365D+01 0.510020D+00 0.460299D-01 0.416474D+01
 0.901246D-01 0.226758D+01 0.376869D-01 - 147001D+02 - 116249D+01 0.113616D+01 0.111328D+00 0.416474D+01
 0.992932D-01 0.230540D+01 0.631098D-01 - 138517D+02 - 122424D+01 0.748020D+00 0.563391D-01 0.416474D+01
 0.119277D+00 0.234194D+01 0.104680D+00 - 101547D+02 - 129677D+01 0.998210D+00 0.926107D-01 0.416474D+01
 0.126460D+00 0.233875D+01 0.112748D+00 - 831385D+01 - 130412D+01 0.107317D+01 0.101956D+00 0.416474D+01
 0.133643D+00 0.232805D+01 0.118208D+00 - 651370D+01 - 130433D+01 0.113616D+01 0.111328D+00 0.416474D+01
 0.140825D+00 0.231144D+01 0.121586D+00 - 476414D+01 - 129952D+01 0.118821D+01 0.120681D+00 0.416474D+01
 0.163850D+00 0.2222674D+01 0.124336D+00 0.414776D+00 - 126850D+01 0.127446D+01 0.150274D+00 0.416474D+01
 0.188255D+00 0.209381D+01 0.126117D+00 0.510010D+01 - 123300D+01 0.124444D+01 0.180809D+00 0.416474D+01
 0.213942D+00 0.191025D+01 0.140132D+00 0.904649D+01 - 119276D+01 0.109829D+01 0.211999D+00 0.416474D+01
 0.240804D+00 0.167423D+00 0.175048D+00 0.118668D+02 - 111668D+01 0.854944D+00 0.243118D+00 0.416474D+01
 0.268731D+00 0.139904D+01 0.230143D+00 0.130540D+02 - 965909D+00 0.562293D+00 0.272348D+00 0.416474D+01
 0.327317D+00 0.870460D+00 0.349961D+00 0.100768D+02 - 732974D+00 0.292751D+00 0.296997D+00 0.416474D+01
 0.342527D+00 0.764172D+00 0.37161618D+00 0.8622793D+01 - 460693D+00 0.107613D+00 0.314739D+00 0.416474D+01
 0.357736D+00 0.671285D+00 0.388156D+00 0.716236D+01 - 330271D+00 0.532856D+01 0.320742D+00 0.416474D+01
 0.388740D+00 0.512863D+00 0.408660D+00 0.451765D+01 - 215183D+00 0.217680D+01 0.324871D+00 0.416474D+01
 0.420200D+00 0.376396D+00 0.417498D+00 0.263385D+01 0.711936D-01 0.540248D-02 0.328794D+00 0.416474D+01
 0.451988D+00 0.249377D+00 0.420642D+00 0.158724D+01 0.133315D+00 0.213141D-01 0.327944D+00 0.416474D+01
 0.516026D+00 0.127684D+00 0.421602D+00 0.121382D+01 0.174196D+00 0.448206D-01 0.324604D+00 0.416474D+01
 0.548012D+00 - 940520D-01 0.421286D+00 0.127507D+01 0.212745D+00 0.802249D-01 0.313456D+00 0.416474D+01
 0.579800D+00 - 188488D+00 0.420350D+00 0.184583D+01 0.316411D+00 0.890200D-01 0.305931D+00 0.416474D+01
 0.611260D+00 - 269212D+00 0.419033D+00 0.205670D+01 0.382108D+00 0.920601D-01 0.296808D+00 0.416474D+01
 0.642264D+00 - 335845D+00 0.417515D+00 0.21104D+01 0.451161D+00 0.653977D-01 0.319653D+00 0.416474D+01
 0.672683D+00 - 389535D+00 0.416135D+00 0.198500D+01 0.517470D+00 0.836515D-01 0.285837D+00 0.416474D+01
 0.702392D+00 - 432774D+00 0.414957D+00 0.168349D+01 0.575278D+00 0.745852D-01 0.258181D+00 0.416474D+01
 0.731269D+00 - 469005D+00 0.4141113D+00 0.122698D+01 0.619887D+00 0.641851D-01 0.241926D+00 0.416474D+01
 0.759196D+00 - 502184D+00 0.413626D+00 0.641515D+00 0.647932D+00 0.43972D-01 0.224638D+00 0.416474D+01
 0.786058D+00 - 5363351D+00 0.413483D+00 - 473038D-01 0.5056676D+00 0.109437D-01 0.206897D+00 0.416474D+01
 0.811745D+00 - 575283D+00 0.413615D+00 - 436572D+00 0.657373D+00 0.354323D-01 0.272924D+00 0.416474D+01
 0.836150D+00 - 622176D+00 0.414143D+00 - 816762D+00 0.647373D+00 0.622988D-02 0.122276D+00 0.416474D+01
 0.856590D+00 - 981565D+00 0.316623D+00 - 419302D+01 0.359561D+00 0.252546D-02 0.115006D+00 0.416474D+01
 0.950484D+00 - 105384D+01 0.293442D+00 - 164672D+01 0.291287D+00 - 183078D-03 0.109595D-00 0.416474D+01
 0.963458D+00 - 112102D+01 0.276317D+00 - 434758D+01 0.227959D+00 - 199611D-02 0.105744D-00 0.416474D+01
 0.974528D+00 - 132494D+01 0.243230D+00 - 448478D+01 0.170601D+00 - 292964D-02 0.103156D+00 0.416474D+01
 0.983647D+00 - 127516D+01 0.247908D+00 - 467773D+00 0.779364D-01 - 259136D-02 0.101545D+00 0.416474D+01
 0.990780D+00 - 130630D+01 0.442801D-01 0.475019D+00 - 480561D+01 0.144280D+00 0.157043D+00 0.416474D+01
 0.995895D+00 - 132494D+01 0.240489D+00 - 484024D+01 0.497726D-02 - 240236D-03 0.100041D+00 0.416474D+01
 0.998973D+00 - 133130D+01 0.239586D+00 - 485199D+01 0.000000D+00 0.100003D+00 0.100000D+00 0.416474D+01

Chapter VII

VII.1 Input-Output Manual for the Out-of-Plane Linear Eigenproblem Solution Program, RCLINDY3

This program is written in FORTRAN 77 and allows the interactive solution of the out-of-plane linear dynamic eigenproblem for a compliant riser with a planar static configuration, as described in [4]. The initial ($\epsilon=0$) approximation is prepared by RCFORC1. This program is particularly useful in the case of zero current velocity where the asymptotic theory is not valid.

Before executing the program the user must make sure that:

- o Devices five and six correspond to input/output from the terminal
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data preprocessing program RCINPUT. This file has a logical record length of 80 characters.
- o Device nine must be associated with an output data file with logical record length of 132 characters which upon completion of the program contains a complete list of the input, initial approximation and the converged solution of the problem.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.

- o Device eleven must be associated with an output file with logical record length of 117 characters. This file contains the converged solution of the problem in a concise form appropriate for use in the preparation of illustrations and/or as an input to RCLINDYN from device twelve.
- o Device twelve must be associated with an input file containing the initial approximate solution of the undamped forced out-of-plane linear dynamic problem resulting from a previous execution of RCFORC1.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The values of the mean current velocity in the x direction and the maximum non-dimensional static effective tension are printed on the terminal. If the number of static discretization points is less than four or greater than the value of MNP (maximum number of allowable points) the program halts execution.

Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop, see [3].

In this case an initial ($\epsilon=0$) approximation to the solution is available from a previous run of RCFORC1. The program reads the information from the previous run concerning the number of discretization points NP, the mode number and the corresponding frequency of excitation. If the number of discretization points is greater than MNP, the program stops execution. The

same happens if NP is less than the number of static discretization points. Then the user has the option to review the initial approximation by displaying in his terminal. The assumed boundary condition for the curvature at the lower ball point of the riser is printed at the terminal. Subsequently, the initial solution provided by RCFORC1 is printed in the file associated with the device nine. The following quantities are printed:

- o Number of discretization points NP
- o Natural frequency (rad/s)
- o Order of point (starting from the lower end)
- o Non-dimensional arc length
- o Non-dimensional shear force in the \vec{n} direction, Q_1^n
- o Non-dimensional component of rate of rotation about $\vec{\xi}, \Omega_1^\xi$
- o Non-dimensional component of rate of rotation about $\vec{\zeta}, \Omega_1^\zeta$
- o Euler angle θ_1
- o Euler angle β or ψ_1
- o Non-dimensional displacement r
- o Non-dimensional natural frequency

Subsequently, the program prints the maximum of the absolute value of the initial rate of rotation about $\vec{\xi}$ direction. This number is used to estimate a reasonable tolerance for convergence of iterations. The user is requested to input a tolerance as a fraction of this number; e.g. 0.01 for 1% tolerance or better for all quantities of interest. Typically,

Ω_1^{ξ} will be determined with an accuracy a little better than the above fraction and all other variables will be even more accurate. After the run the accuracy of the results should be checked. The user is given the opportunity to stop the program at this point if he wishes.

If he chooses to continue, he is asked to input the initial increment of the continuation parameter $\delta\epsilon$ (condition $0 < \delta\epsilon \leq 1$). If no continuation is required, then use $\delta\epsilon = 1$. If continuation is required, then $\delta\epsilon = 0.1$ will usually suffice. For large changes between initial and final problem, a smaller value of $\delta\epsilon$, e.g. 0.05 or 0.25 might be necessary.

After the initial approximation is fully defined, the program enters NAG subroutine D02RAF, which provides the iterative numerical solution of the problem. The manual of this subroutine is included in Chapter X. The input value of IFAIL is 111, indicating that the soft failure option with error message and monitoring information printing has been selected. Error messages are directed to device six. Once the execution of D02RAF is completed, the current value of IFAIL is printed on the terminal. For an explanation of this parameter, see NAG manual for D02RAF, Chapter X. Only if IFAIL = 0 or IFAIL = 4, the execution of the program continues with printout of the riser characteristics and the final solution of the problem. Otherwise the program stops. When IFAIL ≠ 0, the user should consult the manual quoted above for interpretation of the results and a possible course of action. If IFAIL = 0 or 4, the riser characteristics and the final solution of the problem are printed in the file associated with device nine. For the riser characteristics the format used in the static program RCSTAT2D is also used here. This is described in [3] (pp. 23-25).

Next the non-dimensional out-of-plane solution is written in the file associated with device nine.

The mode number, natural frequency and final number of points NP at which the solution is available is printed first. Then the following data is printed for I=1 to NP:

$s, Q_1^\eta, Q_1^\zeta, Q_1^\xi, \theta_1, \beta, r, \Sigma$ with

FORMAT (8(1X,D10.4)). Next the maximum (non-dimensional) estimated errors of $Q_1^\eta, Q_1^\zeta, Q_1^\xi, \theta_1, \beta, r, \Sigma$ provided by the NAG subroutine D02RAF are printed.

Subsequently, the following data is written in the file associated with device eleven:

MODE, NP, SIGMAD: Mode number, number of points at which converged solution is available and dimensional natural frequency (in rad/s) with
FORMAT (1X,I2,1X,I3,1X,D10.4) and

for I=1 to NP the non-dimensional

$s, Q_1^\eta, Q_1^\zeta, Q_1^\xi, \theta_1, \beta, r, \Sigma$ and the dimensional current velocity $V_C(s)$ with
FORMAT (9(1X,D12.6))

The data written in the file associated with device eleven due to their simple form can be easily used as input to plotting programs. In addition, this data is useful in subsequent runs of RCLINDYN, if such an option is selected.

VII.2 Sample Run, Input and Output

VII.2.1 Zero Current Excitation

VII.2.1.1 Interactive Session

INPUT

Device 5 = TERMINAL

Device 8 = RCJOA01 DATA A*

Device 10 = RC2DZERR DATA A**

Device 12 = OUTZER1E DATA A**

OUTPUT

Device 6 = TERMINAL

Device 9 = OUTZERT1 DATA A

Device 11 = OUTZER1A DATA A

*File included in Chapter II.

**Files included in Chapter VI.

R; 1=0 01/0.01 17:14:29

FC1Indy3 reLoad1 outzer1 fc2dzer1 outzer1 outzer1
FI 8 DISK RCJOAO1 DATA A (RECFM FB LRECL 80 BLKSIZE 800
FI 9 DISK OUTZER1 DATA A (RECFM FB LRECL 132 BLKSIZE 1320
FI 10 DISK RC2DZER1 DATA A (RECFM FB LRECL 132 BLKSIZE 1320
FI 11 DISK OUTZER1A DATA A (RECFM FB LRECL 117 BLKSIZE 1170
FI 12 DISK OUTZER1E DATA A (RECFM FB LRECL 117 BLKSIZE 1170
GLOBAL XYLIB VFORTLIB CMSLIB NAG1 NAG2
LOAD RCLINDY3 ! START
EXECUTION BEGINS...
MNP=151

2-D STATIC SOLUTION FROM DEVICE 10

NP = 54 MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S , VM = 0.00000D+00

2-D STATIC SOLUTION SUCCESSFULLY READ

MAXIMUM STATIC EFFECTIVE TENSION/WA*L = 0.55470D+00

DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS

IF YES INPUT 1 , IF NO INPUT 0

?
O

MNP=151

INITIAL APPROXIMATION FROM DEVICE 12

NP = 54 MODE NUMBER = 1 SIGMAD = 0.4900D-01

DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION

IF YES INPUT 1

?
2

ASSUMED BOUNDARY CONDITION

OMEGA_X1 (0) = 136137D+02

MAXIMUM ABSOLUTE VALUE OF N-D OMEGA_X1 IS = 0.158754D+02

THIS NUMBER CAN BE USED TO ESTIMATE

A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS

INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE
E.G., INPUT 0.01 OR 0.1 IF THE INITIAL ESTIMATE FOR OMEGA_X1
IS SMALL

AFTER THE RUN CHECK THE ACCURACY OF THE SOLUTION

?
O.08

IF YOU WISH TO STOP INPUT 0

IF YOU WISH TO CONTINUE WITH ITERATIONS INPUT 1

?
1

INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS

IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.D0

IF CONTINUATION IS REQUIRED THEN 0.D0 < DELEPS < 1.D0

RECOMMENDATION :

USUALLY DELEPS = 0.100 WILL SUFFICE

FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM
A SMALLER VALUE OF DELEPS, E.G., 0.05D0 OR 0.025D0 MIGHT BE
NECESSARY

?
1

DO2RAF MONITORING INFORMATION

MONITORING NEWTON ITERATION
NUMBER OF POINTS IN CURRENT MESH = 54
CORRECTION NUMBER 0 RESIDUAL SHOULD BE I.E. 1.00D+00
ITERATION NUMBER 0 RESIDUAL = 1.52D-01
SQUARED NORM OF CORRECTION = 3.49D-01

SQUARED NORM OF GRADIENT = 1.94D+00
SCALAR PRODUCT OF CORRECTION AND GRADIENT = 2.30D-02
ITERATION NUMBER 1 RESIDUAL = 1.51D-05

CONTINUATION PARAMETER EPSILON = 2.00D-01 DELEPS = 2.00D-01

MONITORING NEWTON ITERATION
NUMBER OF POINTS IN CURRENT MESH = 54
CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00D+00
ITERATION NUMBER 0 RESIDUAL = 2.83D-02

CONTINUATION PARAMETER EPSILON = 6.00D-01 DELEPS = 4.00D-01

MONITORING NEWTON ITERATION
NUMBER OF POINTS IN CURRENT MESH = 54
CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00D+00
ITERATION NUMBER 0 RESIDUAL = 8.49D-02

CONTINUATION PARAMETER EPSILON = 1.00D+00 DELEPS = 8.00D-01

MONITORING NEWTON ITERATION
NUMBER OF POINTS IN CURRENT MESH = 54
ITERATION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.27D-04
ITERATION NUMBER 0 RESIDUAL = 1.72D-01

SQUARED NORM OF CORRECTION = 4.81D+01
SQUARED NORM OF GRADIENT = 2.96D-02
SCALAR PRODUCT OF CORRECTION AND GRADIENT = 2.96D-02
ITERATION NUMBER 1 RESIDUAL = 2.22D-02
SQUARED NORM OF CORRECTION = 1.38D+00
SQUARED NORM OF GRADIENT = 8.23D-05
SCALAR PRODUCT OF CORRECTION AND GRADIENT = 4.92D-04
ITERATION NUMBER 2 RESIDUAL = 6.56D-05

MESH SELECTION

NUMBER OF NEW POINTS 2

CORRECTION NUMBER 0 ESTIMATED MAXIMUM ERROR = 2.70D-01
ESTIMATED ERROR BY COMPONENTS
1.71D-02 4.55D-03 2.70D-01 2.60D-02 2.16D-02 5.02D-03 1.52D-02
IF FAIL = 0
R; T=2.34/2.63 17:16:31
cp spool console stop close

OUTZER1 DATA A

INITIAL CONDITION FOR EPS=0, AND NP = 54 POINTS, NATURAL FREQUENCY = 0.4900D-01 RAD/S

ARC	SHEAR ZETA	OMEGA ZETA	OMEGA XI	THETA	BETA	R	SIGMA
1	0.00000D+00	0.142261D+01	-160160D+00	-136137D+02	0.00000D+00	0.00000D+00	0.416474D+01
2	0.102730D-02	0.143388D+01	-158951D+00	-136785D+02	-140192D-01	-98926D-04	0.416474D+01
3	0.410499D-02	0.146779D+01	-155146D+00	-138713D+02	-564183D-01	0.412048D-03	0.416474D+01
4	0.922042D-02	0.152451D+01	-148215D+00	-141852D+02	-128013D+00	0.409655D-02	0.416474D+01
5	0.163526D-01	0.160430D+01	-137290D+00	-146022D+02	-230108D+00	0.409655D-02	0.416474D+01
6	0.254721D-01	0.170548D+01	-121216D+00	-150843D+02	-362907D+00	0.422806D-01	0.416474D+01
7	0.365416D-01	0.182610D+01	-990766D-01	-155488D+02	-524672D+00	0.93930D-01	0.416474D+01
8	0.495156D-01	0.195975D+01	-697622D-01	-158754D+02	-709046D+00	0.182533D+00	0.416474D+01
9	0.643406D-01	0.209552D+01	-324159D-01	-158693D+02	-902589D+00	0.319317D+00	0.416474D+01
10	0.809559D-01	0.221724D+01	0.122805D-01	-153177D+02	-108365D+01	0.510020D+00	0.460299D-01
11	0.901246D-01	0.226758D+01	0.376869D-01	-147001D+02	-116249D+01	0.626943D+00	0.456594D-02
12	0.992932D-01	0.230540D+01	0.631098D-01	-138517D+02	-122424D+01	0.748020D+00	0.948019D-02
13	0.119277D+00	0.234194D+01	0.104680D+00	-101547D+02	-129677D+01	0.998210D+00	0.174938D-01
14	0.126460D+00	0.23387875D+01	0.112748D+00	-831385D+01	-130412D+01	0.107317D+01	0.416474D+01
15	0.133643D+00	0.2232805D+01	0.118208D+00	-651370D+01	-130433D+01	0.113661D+01	0.416474D+01
16	0.140825D+00	0.231144D+01	0.121588D+00	-476414D+01	-129952D+01	0.118828D+01	0.120681D+00
17	0.163850D+00	0.222614D+01	0.124336D+00	0.414776D+02	-126850D+01	0.127468D+01	0.150274D+00
18	0.188255D+00	0.2234194D+01	0.190250D+01	0.126117D+00	0.510010D+01	0.124444D+01	0.180809D+00
19	0.213942D+00	0.190250D+01	0.175034D+00	0.175034D+00	0.904649D+01	-119276D+01	0.109829D+01
20	0.240804D+00	0.167423D+01	0.118208D+00	-651370D+01	-118668D+02	-116668D+01	0.211999D+00
21	0.268731D+00	0.139904D+01	0.230143D+00	0.130540D+02	-965909D+00	0.854944D+00	0.24318D+00
22	0.297608D+00	0.118814D+01	0.293823D+00	0.123397D+02	-732974D+00	0.272348D+00	0.416474D+01
23	0.327317D+00	0.870460D+00	0.349961D+00	0.100768D+02	-466939D+00	0.292751D+00	0.296997D+00
24	0.342527D+00	0.767472D+00	0.371618D+00	0.174504D+00	-119276D+01	0.107613D+00	0.314739D+00
25	0.357736D+00	0.671285D+00	0.388156D+00	0.121382D+01	-330271D+00	0.532876D-01	0.320742D+00
26	0.388740D+00	0.512863D+00	0.408660D+00	0.451765D+01	-215183D+00	0.217680D-01	0.324871D+00
27	0.420200D+00	0.376396D+00	0.417498D+00	0.263385D+01	-376947D-01	0.540248D-02	0.328616D+00
28	0.451988D+00	0.188488D+00	0.420642D+00	0.158724D+01	0.711936D-01	0.213141D-01	0.327944D+00
29	0.483974D+00	0.127684D+00	0.419033D+00	0.205670D+01	0.133315D+00	0.448206D-01	0.324604D+00
30	0.516026D+00	0.122720D-01	0.4211719D+00	0.716236D+01	0.174196D+00	0.653977D-01	0.320742D+00
31	0.548012D+00	0.940520D-01	0.421286D+00	0.127507D+01	0.212745D+00	0.802249D-01	0.319653D+00
32	0.579800D+00	0.188488D+00	0.420350D+00	0.154510D+01	0.259198D+00	0.890200D-01	0.305931D+00
33	0.611260D+00	0.269212D+00	0.419033D+00	0.122698D+01	0.316411D+00	0.920601D-01	0.296808D+00
34	0.642264D+00	0.335845D+00	0.417551D+00	0.211104D+01	0.382108D+00	0.899324D-01	0.285837D+00
35	0.672683D+00	0.389535D+00	0.416135D+00	0.122772D+01	0.451161D+00	0.836515D-01	0.319653D+00
36	0.702392D+00	0.432774D+00	0.414957D+00	0.198500D+01	0.517470D+00	0.745852D-01	0.272924D+00
37	0.731269D+00	0.469005D+00	0.414113D+00	0.164672D+01	0.618150D+00	0.641851D-01	0.258181D+00
38	0.759196D+00	0.502184D+00	0.413626D+00	0.149043D+00	0.641515D+00	0.536959D-01	0.241926D+00
39	0.786058D+00	0.536351D+00	0.419044D+00	-473038D-01	-0.506676D+00	0.439722D-01	0.206897D+00
40	0.811745D+00	-575283D+00	0.413659D+00	-391765D+00	-0.657372D+00	0.354323D-01	0.131678D+00
41	0.836150D+00	-622176D+00	0.414143D+00	-816762D+00	-0.647373D+00	0.1222172D-01	0.189325D+00
42	0.880723D+00	-679403D+00	0.414950D+00	-316623D+00	-0.359561D+00	0.252546D-02	0.115006D+00
43	0.900707D+00	-748880D+00	0.419044D+00	-293482D+00	-0.291287D+00	0.157043D+00	0.109595D+00
44	0.904944D+00	-776377D+00	-0.344466D+01	-0.570840D+00	-0.162046D-01	0.143316D+00	0.416474D+01
45	0.919044D+00	-827039D+00	-0.439314D+01	-0.403295D+01	-0.432950D+00	0.109437D-01	0.170610D+00
46	0.935659D+00	-905753D+00	-0.347458D+01	-0.561430D+01	-0.412048D-03	0.115440D-03	0.416474D+01
47	0.950484D+00	-981565D+00	-0.419384D+01	-0.493020D+01	-0.218123D-01	0.157043D+00	0.416474D+01
48	0.963458D+00	-105384D+01	-0.419044D+00	-0.4344758D+01	-0.227959D+00	-0.199611D-02	0.105744D+01
49	0.974280D+00	-112102D+01	-0.419044D+00	-0.448478D+01	-0.170610D+00	-0.292964D-02	0.16474D+01
50	0.983647D+00	-118138D+01	-0.426373D+00	-0.459535D+01	-0.120270D+00	-0.306826D-02	0.16474D+01
51	0.990780D+00	-123323D+01	-0.4254686D+00	-0.467773D+01	-0.779364D-01	-0.259136D-02	0.16474D+01
52	0.995895D+00	-127916D+01	-0.4247908D+00	-0.475019D+01	-0.442801D-01	-0.10641D+00	0.16474D+01
53	0.998973D+00	-132494D+01	-0.243230D+01	-0.480561D+01	-0.198262D-01	-0.892554D-03	0.100041D+00
						-0.240236D-03	0.100003D+00

54 O 100000D+01 - .133130D+01 O .239586D+00 -.485199D+01 O .000000D+00 O .000000D+00 O .100000D+00 O .416474D+01
 CHINESE LANTERN FROM JIAO, APRIL 1985

9 = NUMBER OF RISER SEGMENTS

0.883920D+02 = UNSTRETCHED RISER LENGTH IN M

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M3

0.102500D+04 = SALT WATER DENSITY IN KG/M3

0.115434D+01 = INNER CROSS SECTIONAL AREA IN M2

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M2

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

DATA PER RISER SEGMENT FOR NSEG = 9 SEGMENTS

DIMENSIONAL QUANTITIES IN THE SYSTEM

RLENG DX1 PXIETA AO MASS TMASS AMAZI TMAXI TMAETA

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.6839D+02 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

0.2500D+01 0.3100D+00 0.9300D+00 0.2374D+01 0.2920D+01 0.4993D+02 0.4047D+02 0.8244D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.1003D+03

M O D E N U M B E R = 1
 N A T U R A L F R E Q U E N C Y = 0.5357D-01 RAD/S

ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = 54 POINTS

S	QETA	OMEGA ZETA	OMEGA GAXI	THETA	BETIA	R	SIGMA
0.0000D+00	0.8581D+01	-1004D+01	-7467D+02	0.0000D+00	0.0000D+00	0.0000D+00	0.4553D+01
0.1027D-02	0.8643D+01	-9981D+00	-7505D+02	-7691D-01	-6708D-03	0.3950D-04	0.4553D+01
0.4105D-02	0.8829D+01	-9790D+00	-7616D+02	-3096D+00	0.1775D-02	0.6342D-03	0.4553D+01
0.9220D-02	0.9141D+01	-9438D+00	-7800D+02	-7033D+00	0.2148D-01	0.3225D-02	0.4553D+01
0.1635D-01	0.9578D+01	-8877D+00	-8048D+02	-1265D+01	0.8430D-01	0.1024D-01	0.4553D+01
0.2547D-01	0.1014D+02	-8045D+00	-8341D+02	-1997D+01	0.2301D+00	0.2512D-01	0.4553D+01
0.3654D-01	0.1080D+02	-6891D+00	-8637D+02	-2892D+01	0.5134D+00	0.5218D-01	0.4553D+01
0.4952D-01	0.1154D+02	-5352D+00	-8870D+02	-3918D+01	0.1000D+01	0.9635D-01	0.4553D+01
0.6434D-01	0.1230D+02	-3387D+00	-8939D+02	-5003D+01	0.1752D+01	0.1625D+00	0.4553D+01
0.8096D-01	0.1298D+02	-1024D+00	-8720D+02	-6036D+01	0.2802D+01	0.2542D+00	0.4553D+01
0.9529D-01	0.1348D+02	0.3193D-01	-8437D+02	-6499D+01	0.3452D+01	0.3117D+00	0.4553D+01
0.1193D+00	0.1369D+00	0.1653D+00	-8025D+02	-6872D+01	0.4126D+01	0.3730D+00	0.4553D+01
0.1265D+00	0.1367D+02	0.3892D+00	-6108D+02	-7368D+01	0.5503D+01	0.5152D+00	0.4553D+01
0.1336D+00	0.1361D+02	0.4384D+00	-4384D+00	-5096D+02	0.7446D+01	0.5928D+01	0.4553D+01
0.1408D+00	0.1351D+02	0.4742D+00	-4094D+02	-7484D+01	0.2802D+01	0.2542D+00	0.4553D+01
0.1638D+00	0.1302D+02	0.4943D+00	-3112D+02	-7493D+01	0.6585D+01	0.6221D+00	0.4553D+01
0.1883D+00	0.1224D+02	0.5208D+00	-15700D+01	-7435D+01	0.1426D+01	0.6759D+00	0.4553D+01
0.2139D+00	0.1114D+02	0.5356D+00	0.2602D+02	-7314D+01	0.7084D+01	0.8477D+00	0.4553D+01
0.2408D+00	0.9710D+01	0.6148D+00	0.50113D+02	-7107D+01	0.6908D+01	0.1028D+01	0.4553D+01
0.2687D+00	0.8027D+01	0.1131D+01	0.7794D+02	-6618D+01	0.4623D+01	0.1213D+01	0.4553D+01
0.2976D+00	0.6276D+01	0.1511D+01	0.7666D+01	-5622D+01	0.2930D+01	0.1397D+01	0.4553D+01
0.3273D+00	0.4682D+01	0.1863D+01	0.6595D+02	-4085D+01	0.1409D+01	0.1708D+01	0.4553D+01
0.3425D+00	0.3987D+01	0.2007D+01	0.58117D+02	-2268D+01	0.4238D+00	0.1803D+01	0.4553D+01
0.3577D+00	0.3379D+01	0.2120D+01	0.4995D+02	-1371D+01	0.1757D+00	0.1830D+01	0.4553D+01
0.3887D+00	0.2346D+01	0.2272D+01	0.3447D+02	-5656D+00	0.7181D-01	0.1845D+01	0.4553D+01
0.4202D+00	0.1494D+01	0.2345D+01	0.2237D+02	-7206D+00	0.1352D+00	0.1843D+01	0.4553D+01
0.4520D+00	0.7461D+00	0.2375D+01	0.1453D+02	-1577D+01	0.3789D+00	0.1807D+01	0.4553D+01
0.4840D+00	0.6643D-01	0.2384D+01	0.1038D+02	-2117D+01	0.6295D+00	0.1748D+01	0.4553D+01
0.5160D+00	-5500D+00	0.2386D+01	0.7664D+02	-2484D+01	0.8104D+00	0.1674D+01	0.4553D+01
0.5480D+00	-1095D+01	0.2363D+01	0.8751D+01	-2784D+01	0.9058D+00	0.1590D+01	0.4553D+01
0.5798D+00	-1559D+01	0.2360D+01	0.8464D+01	-3077D+01	0.9254D+00	0.1496D+01	0.4553D+01
0.6113D+00	-1938D+01	0.2358D+01	0.8571D+01	-3379D+01	0.8866D+00	0.1393D+01	0.4553D+01
0.6423D+00	-2233D+01	0.2368D+01	0.8426D+01	-3683D+01	0.8074D+00	0.1282D+01	0.4553D+01
0.6727D+00	-2454D+01	0.2363D+01	0.7664D+01	-3968D+01	0.7051D+00	0.1164D+01	0.4553D+01
0.7024D+00	-2617D+01	0.2364D+01	0.6150D+01	-4207D+01	0.5942D+00	0.1039D+01	0.4553D+01
0.7313D+00	-3316D+01	0.2369D+01	0.3896D+01	-4378D+01	0.4862D+00	0.9119D+00	0.4553D+01
0.7592D+00	-3565D+01	0.2358D+01	0.1000D+01	-4466D+01	0.3884D+00	0.7842D+00	0.4553D+01
0.7861D+00	-3877D+01	0.2384D+01	0.2336D+01	-2336D+02	0.3046D+00	0.6596D+00	0.4553D+01
0.8117D+00	-4230D+01	0.2236D+01	0.6226D+01	-4236D+01	0.2354D+00	0.2354D+00	0.4553D+01
0.8362D+00	-4582D+01	0.1994D+01	-1032D+02	0.4143D+01	0.2520D+01	0.5413D+01	0.4553D+01
0.9357D+00	-4917D+01	0.1817D+01	-250D+02	0.2060D+01	0.1013D+01	0.8391D-01	0.4553D+01
0.9505D+00	-5231D+01	0.1684D+01	-2510D+02	0.1643D+01	-4943D-02	0.5315D-01	0.4553D+01
0.9635D+00	-5521D+01	0.1586D+01	-2547D+02	0.1270D+01	-1442D-01	0.3156D-01	0.4553D+01
0.9907D+00	-4230D+01	0.2236D+01	-6226D+01	0.3046D+00	0.6078D-01	0.1811D+00	0.4553D+01

FILE: OUTZER1 DATA A1 VM/SP CONVERSATIONAL MONITOR SYSTEM PAGE 004

0.9745D+00 - .5780D+01 0.1513D+01 -.2562D+02 0.6569D+00 -.1871D-01 0.8381D-02 0.4553D+01
 0.9836D+00 -.6001D+01 0.1460D+01 -.2570D+02 0.4226D+00 -.1538D-01 0.3459D-02 0.4553D+01
 0.9908D+00 -.6179D+01 0.1421D+01 -.2580D+02 0.2387D+00 -.1031D-01 0.1101D-02 0.4553D+01
 0.9959D+00 -.6310D+01 0.1394D+01 -.2588D+02 0.1064D+00 -.5155D-02 0.2185D-03 0.4553D+01
 0.9990D+00 -.6389D+01 0.1378D+01 -.2593D+02 0.2665D-01 -.1379D-02 0.1368D-04 0.4553D+01
 0.1000D+01 -.6416D+01 0.1373D+01 -.2595D+02 0.1563D-16 0.0000D+00 0.4302D-21 0.4553D+01
 MAXIMUM ESTIMATED ERROR BY COMPONENTS
 0.9395D-01 0.2495D-01 0.1481D+01 0.1426D+00 0.1186D+00 0.2755D-01 0.1519D-01

OUTZERIA DATA A

1 54 0.53570+01
 0.0000000+00 0.8581200+01 - 100419D+01 - 746714D+02 0.000000D+00 0.000000D+00 0.000000D+00 0.455274D+01 0.000000D+00
 0.102730D-02 0.864306D+01 - 998126D+00 - 750457D+02 - 69053D-01 - 67076D-03 0.395024D-04 0.455274D+01 0.000000D+00
 0.410499D-02 0.882921D+01 - 978965D+00 - 761637D+02 - 309577D+00 0.177315D+00 0.214804D-01 0.322493D-03 0.455274D+01 0.000000D+00
 0.922042D-02 0.914089D+01 - 943772D+00 - 780005D+02 - 703315D+00 0.126471D+01 0.842990D-01 0.102431D-01 0.455274D+01 0.000000D+00
 0.163526D-01 0.957835D+01 - 887735D+00 - 804786D+02 - 126471D+01 0.199715D+01 0.230058D+00 0.251164D-01 0.455274D+01 0.000000D+00
 0.252214D-01 0.101375D+02 - 804464D+00 - 834136D+02 - 199715D+01 0.199715D+01 0.230058D+00 0.251164D-01 0.455274D+01 0.000000D+00
 0.365416D-01 0.108037D+02 - 689104D+00 - 863653D+02 - 887041D+02 - 289230D+01 0.513434D+00 0.521782D-01 0.455274D+01 0.000000D+00
 0.495156D-01 0.115440D+02 - 535241D+00 - 887041D+02 - 391770D+01 0.100013D+01 0.963547D-01 0.455274D+01 0.000000D+00
 0.643406D-01 0.122999D+02 - 338730D+00 - 893892D+02 - 500350D+01 0.175221D+01 0.162483D+00 0.455274D+01 0.000000D+00
 0.809559D-01 0.129792D+02 - 102413D+00 - 872021D+02 - 603610D+01 0.280235D+01 0.254196D+00 0.455274D+01 0.000000D+00
 0.901246D-01 0.132637D+02 0.319292D-01 - 843732D+02 - 649874D+02 - 689230D+01 0.345173D+01 0.311660D+00 0.455274D+01 0.000000D+00
 0.992932D-01 0.134787D+02 0.165251D+00 - 802519D+02 - 687211D+01 0.412617D+01 0.372956D+00 0.455274D+01 0.000000D+00
 0.119277D+00 0.136868D+02 - 389243D+00 - 610806D+02 - 736843D+01 0.550341D+01 0.515246D+00 0.455274D+01 0.000000D+00
 0.126460D+00 0.136703D+02 0.438396D+00 - 509592D+02 - 744612D+01 0.592822D+01 0.568453D+00 0.455274D+01 0.000000D+00
 0.1336543D+00 0.136909D+02 0.472457D+00 - 409432D+02 - 603610D+01 0.280235D+01 0.254196D+00 0.455274D+01 0.000000D+00
 0.1408825D+00 0.135143D+02 0.494263D+00 - 311218D+02 - 749253D+01 0.658481D+01 0.677585D+00 0.455274D+01 0.000000D+00
 0.1638550D+00 0.130234D+02 0.520782D+00 - 157041D+01 - 743488D+01 0.708412D+01 0.847706D+00 0.455274D+01 0.000000D+00
 0.188255D+00 0.1222356D+02 0.535573D+00 - 260207D+02 - 731442D+01 0.690796D+01 0.102768D+01 0.455274D+01 0.000000D+00
 0.2113942D+00 0.1111356D+02 0.614820D+00 - 501319D+02 - 710659D+01 0.604861D+01 0.121290D+01 0.455274D+01 0.000000D+00
 0.240804D+00 0.970987D+01 0.812237D+00 - 684526D+02 - 661798D+01 0.462253D+01 0.139724D+01 0.455274D+01 0.000000D+00
 0.268731D+00 0.802707D+01 0.113070D+01 - 779419D+02 - 5622231D+01 0.292959D+01 0.156815D+01 0.455274D+01 0.000000D+00
 0.297608D+00 0.627635D+01 0.151119D+01 - 766618D+02 - 408471D+01 0.140902D+01 0.170831D+01 0.455274D+01 0.000000D+00
 0.327317D+00 0.468168D+01 0.186331D+01 - 659464D+02 - 2626834D+01 0.423750D+00 0.180268D+01 0.455274D+01 0.000000D+00
 0.3424527D+00 0.398688D+01 0.200687D+01 - 581655D+02 - 137051D+01 0.175747D+00 0.183035D+01 0.455274D+01 0.000000D+00
 0.3575736D+00 0.337857D+01 0.212026D+01 - 499522D+02 - 565632D+00 0.718036D+01 0.184508D+01 0.455274D+01 0.000000D+00
 0.388740D+00 0.234601D+01 0.227151D+01 - 344725D+02 0.720634D+00 0.135171D+00 0.184267D+01 0.455274D+01 0.000000D+00
 0.420200D+00 0.149412D+01 0.2334496D+01 - 223651D+02 0.157730D+01 0.378913D+00 0.180653D+01 0.455274D+01 0.000000D+00
 0.451988D+00 0.746069D+00 0.237474D+01 - 145317D+02 0.211735D+01 0.629468D+00 0.174780D+01 0.455274D+01 0.000000D+00
 0.483974D+00 0.664256D+01 0.238449D+01 - 103811D+02 0.2436755D+01 0.807437D+00 0.167422D+01 0.455274D+01 0.000000D+00
 0.516026D+00 - 549957D+00 0.238589D+01 - 245360D+01 0.236286D+01 0.705097D+00 0.16375D+01 0.455274D+01 0.000000D+00
 0.548012D+00 - 109510D+01 0.238344D+01 - 149510D+01 0.23596D+01 0.278416D+01 0.905102D+00 0.158980D+01 0.455274D+01 0.000000D+00
 0.579800D+00 - 1559410D+01 0.237884D+01 - 857138D+01 0.337933D+01 0.886573D+00 0.139345D+01 0.455274D+01 0.000000D+00
 0.611266D+00 - 193787D+01 0.237320D+01 - 842557D+01 0.368339D+01 0.807437D+00 0.128235D+01 0.455274D+01 0.000000D+00
 0.642264D+00 - 223266D+01 0.236755D+01 - 7666392D+01 0.396754D+01 0.705097D+00 0.167422D+01 0.455274D+01 0.000000D+00
 0.672683D+00 - 245360D+01 0.236286D+01 - 149545D+01 0.236286D+01 0.278416D+01 0.905102D+00 0.158980D+01 0.455274D+01 0.000000D+00
 0.702392D+00 - 261694D+01 0.23596D+01 - 149545D+01 0.23596D+01 0.307679D+01 0.925434D+00 0.149506D+01 0.455274D+01 0.000000D+00
 0.731269D+00 - 274329D+01 0.235816D+01 - 149545D+01 0.235816D+01 0.384263D+01 0.484196D+00 0.911890D+00 0.455274D+01 0.000000D+00
 0.759196D+00 - 285519D+01 0.235849D+01 - 149545D+01 0.235849D+01 0.446568D+01 0.388412D+00 0.784194D+00 0.455274D+01 0.000000D+00
 0.880723D+00 - 387729D+01 0.238412D+01 - 233580D+02 0.300425D+00 0.304604D+00 0.655959D+00 0.455274D+01 0.000000D+00
 0.900707D+00 - 422988D+01 0.223626D+01 - 622567D+01 0.434971D+01 0.235391D+00 0.541302D+00 0.16375D+01 0.455274D+01 0.000000D+00
 0.919044D+00 - 312295D+01 0.236392D+01 - 103239D+02 0.414287D+01 0.179211D+00 0.103942D+01 0.455274D+01 0.000000D+00
 0.836150D+00 - 3315841D+01 0.236888D+01 - 146129D+02 0.384263D+01 0.133322D+00 0.432228D+00 0.455274D+01 0.000000D+00
 0.859175D+00 - 356542D+01 0.237548D+01 - 189900D+02 0.354857D+01 0.388412D+00 0.334785D+00 0.455274D+01 0.000000D+00
 0.963458D+00 - 5521150D+01 0.158581D+01 - 233580D+02 0.300425D+00 0.947683D-01 0.250730D+00 0.455274D+01 0.000000D+00
 0.974528D+00 - 577972D+01 0.151881D+01 - 252412D+02 0.251989D+01 0.607795D-01 0.181099D+00 0.455274D+01 0.000000D+00
 0.983647D+00 - 600074D+01 0.146046D+01 - 250126D+02 0.205969D+01 0.17205D-01 0.125902D+00 0.455274D+01 0.000000D+00
 0.990788D+00 - 617882D+01 0.142129D+01 - 257034D+02 0.422561D+00 0.101299D+01 0.839143D-01 0.455274D+01 0.000000D+00
 0.995895D+00 - 630952D+01 0.139419D+01 - 257987D+02 0.238676D+00 0.10395D-01 0.110103D-02 0.455274D+01 0.000000D+00
 0.998973D+00 - 6389341D+01 0.137828D+01 - 259347D+02 0.266463D-01 0.137928D-02 0.218463D-03 0.455274D+01 0.000000D+00
 0.1000300D+01 - 641628D+01 0.1377304D+01 - 259535D+02 0.156311D-16 0.000000D+00 0.430233D-21 0.455274D+01 0.000000D+00

Chapter VIII

VIII.1 Input-Output Manual for the In-Plane Forced
Linear Dynamic Problem Solution Program, RCFORCE

This program is written in FORTRAN 77 and allows the interactive solution of the in-plane forced undamped linear dynamic problem for a compliant riser with a planar static configuration without torsion for a given frequency and amplitude of excitation at the top, as described in [4]. Excitation at the top involving p and/or q is allowed. The output of this program can be used as input to RCLINDY2 to solve the in-plane linear dynamic problem with homogeneous boundary conditions (in-plane eigenproblem).

Before executing the program the user must make sure that

- o Devices five and six correspond to input/output from the terminal
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data pre-processing program RCINPUT. This file has a logical record length of 80 characters.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.
- o Device eleven must be associated with an output file with logical record length of 117 characters. This file contains the converged solution of the problem in a concise form appropriate for use in

the preparation of illustrations and/or as an input to RCLINDY2 from device twelve in a subsequent execution of this program.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The values of the mean current velocity in the x direction and the maximum non-dimensional static effective tension are printed on the terminal. If the number of static discretization points is less than four or greater than the value of MNP (maximum number of allowable points), the program halts execution.

Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop, see [3]. Then the user is asked for the excitation frequency in (rad/s). In order to get meaningful results, this frequency should be close to one of the in-plane natural frequencies of the compliant riser. Next he is asked for the p and q amplitudes of excitation at the top of the riser. Appropriate values for these amplitudes range from 1 to 0. Then the user is asked to supply the tolerance for convergence of iterations. Suggested values are the excitation amplitudes or 1/10 of these values. However, at the end of the run, the results should be checked, since the maximum of the resulting variables will be evaluated within this tolerance. Then the user is asked to input the mode number corresponding to the frequency of excitation.

Subsequently, the program enters NAG subroutine D02GBF, which provides the numerical solution to the linear problem. The manual of this subroutine is included in Chapter X. The input value of IFAIL is 111, indicating that the soft failure option with error message and monitoring information printing has been selected. Error messages are directed to device six. Once the execution of D02GBF is completed, the current value of IFAIL is printed on the terminal. For an explanation of this parameter, see NAG manual for D02GBF, Chapter X.

Only if IFAIL = 0 or IFAIL = 4, the execution of the program continues with printout of the final solution of the problem. Otherwise, the program stops. When IFAIL ≠ 0, the user should consult the manual quoted above for interpretation of the results and a possible course of action. If IFAIL = 0 or 4, the solution is printed in the file associated with device eleven. The following quantities are printed:

MODE, NP, SIGMAD, XTOP, YTOP = Mode number, number of points at which converged solution is available, dimensional natural frequency (in rad/s) and x and y displacements at the top with
FORMAT (1X,I2,1X,I3,3(1X,D10.4)) and
for I=1, NP: the non-dimensional
 $s, T_1, Q_1^{\xi}, Q_1^{\eta}, \phi_1, p, q, \sigma$ with
FORMAT (8(1X,D12.6))

The data written in the file associated with device eleven can be used as input to program RCLINDY2.

VIII.2 Sample Run, Input and Output

VIII.2.1 Zero Current Excitation

VIII.2.1.1 Interactive Session

INPUT

Device 5 = TERMINAL

Device 8 = RCJOA01 DATA A*

Device 10 = RC2DZERR DATA A**

OUTPUT

Device 6 = TERMINAL

Device 11 = RC2DZEE DATA A

* File included in Chapter II.

** File included in Chapter VI.

```

*DISK1: REJOAD1 RCZDERR RCZDZEE
FI 8 DISK RCJOAO1 DATA A ( RECFM FB LRECL 80 BLKSIZE 800
FI 10 DISK RC2DZERR DATA A ( RECFM FB LRECL 132 BLKSIZE 1320
FI 11 DISK RC2DZEE DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
GLOBAL TXLIB VFORLIB CMSLIB NAG1 NAG2
LOAD RCFORCE ( START
EXECUTION BEGINS...
MMPI=151

2-D STATIC SOLUTION FROM DEVICE 10
NP = 54
MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S , VM =0.000000D+00
2-D STATIC SOLUTION SUCCESSFULLY READ
MAXIMUM STATIC EFFECTIVE TENSION/WA*L = 0.5547D+00
DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS
IF YES INPUT 1 , IF NO INPUT 0
?
0
INPUT EXCITATION FREQUENCY IN RAD/S
?
.05
INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR P
?
1
INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR O
?
1
INPUT TOLERANCE FOR CONVERGENCE OF ITERATIONS
SUGGESTED VALUE IS THE EXCITATION AMPLITUDE OR 1/10 OF THE EXCITATION AMPLITUDE
?
.01
INPUT MODE NUMBER CORRESPONDING TO FREQUENCY OF EXCITATION
?
1

DO2GBF MONITORING INFORMATION
NUMBER OF POINTS IN CURRENT MESH = 57

CORRECTION NUMBER 0 ESTIMATED MAXIMUM ERROR = 5.98D-01
ESTIMATED ERROR BY COMPONENTS
3.31D-02 5.01D-02 5.98D-01 6.13D-02 5.09D-03 1.04D-02
NUMBER OF POINTS IN CURRENT MESH = 57

CORRECTION NUMBER 1 ESTIMATED MAXIMUM ERROR = 4.24D-02
ESTIMATED ERROR BY COMPONENTS
1.62D-03 3.28D-03 4.24D-02 3.68D-03 2.70D-04 6.4CD-04
NUMBER OF POINTS IN CURRENT MESH = 57

CORRECTION NUMBER 2 ESTIMATED MAXIMUM ERROR = 1.73D-02
ESTIMATED ERROR BY COMPONENTS
3.97D-04 5.08D-04 1.73D-02 8.05D-04 6.80D-05 8.33D-05
NUMBER OF POINTS IN CURRENT MESH = 57

CORRECTION NUMBER 3 ESTIMATED MAXIMUM ERROR = 1.98D-02
ESTIMATED ERROR BY COMPONENTS
5.84D-04 8.18D-04 1.98D-02 7.59D-04 6.21D-05 8.94D-05
NUMBER OF POINTS IN CURRENT MESH = 107

CORRECTION NUMBER 2 ESTIMATED MAXIMUM ERROR = 4.21D-03
ESTIMATED ERROR BY COMPONENTS
2.95D-04 2.66D-04 4.24D-03 2.69D-04 7.06D-05 7.45D-05
IFAIL = 0
R; T=3.29/3.50 14:41:44

```

VIII.2.1.2 RC2DZEE DATA A

1 107 0.5000D-01 0.0000D+00 0.7931D+00
 0.00000D+00 0.428179D+00 -102337D+00 0.119780D+02 0.000000D+00 0.000000D+00 0.000000D+00 0.424973D+01
 0.102730D-02 0.431381D+00 -.884579D-01 0.120492D+02 0.123418D-01 -.193521D-07 0.633307D-05 0.424973D+01
 0.410499D-02 0.439751D+00 -.463325D-01 0.122497D+02 0.497389D-01 -.127683D-05 0.101696D-03 0.424973D+01
 0.922042D-02 0.449351D+00 0.251938D-01 0.125393D+02 0.113166D+00 -.149823D-04 0.517418D-03 0.424973D+01
 0.163526D-01 0.453003D+00 0.126931D+00 0.128479D+02 0.203766D+00 -.8722334D-04 0.164320D-02 0.424973D+01
 0.254721D-01 0.439897D+00 0.257471D+00 0.130732D+02 0.322106D+00 -.345260D-03 0.402096D-02 0.424973D+01
 0.310068D-01 0.421901D+00 0.335017D+00 0.131165D+02 0.394617D+00 -.637251D-03 0.597600D-02 0.424973D+01
 0.365416D-01 0.396291D+00 0.410274D+00 0.130833D+02 0.467158D+00 -.106492D-02 0.831049D-02 0.424973D+01
 0.397851D-01 0.377852D+00 0.452777D+00 0.130260D+02 0.509508D+00 -.138913D-02 0.98076D-02 0.424973D+01
 0.430286D-01 0.356990D+00 0.493823D+00 0.129413D+02 0.551628D+00 -.177400D-02 0.115149D-01 0.424973D+01
 0.495156D-01 0.308235D+00 0.570986D+00 0.126931D+02 0.634831D+00 -.274850D-02 0.152003D-01 0.424973D+01
 0.532218D-01 0.276556D+00 0.611604D+00 0.125009D+02 0.681531D+00 -.343846D-02 0.175096D-01 0.424973D+01
 0.569281D-01 0.242456D+00 0.649313D+00 0.122674D+02 0.727443D+00 -.423204D-02 0.199587D-01 0.424973D+01
 0.606344D-01 0.206143D+00 0.683934D+00 0.119911D+02 0.7794621D+00 -.513593D-02 0.253395D-01 0.424973D+01
 0.643406D-01 0.167834D+00 0.715335D+00 0.116692D+02 0.816270D+00 -.177400D-02 0.115149D-01 0.424973D+01
 0.684944D-01 0.123057D+00 0.746373D+00 0.112628D+02 0.863912D+00 -.744250D-02 0.274850D-02 0.424973D+01
 0.726482D-01 0.769117D-01 0.772810D+00 0.108198D+02 0.909787D+00 -.887892D-02 0.317026D-01 0.424973D+01
 0.747252D-01 0.534342D-01 0.784297D+00 0.105854D+02 0.932017D+00 -.965547D-02 0.333943D-01 0.424973D+01
 0.768021D-01 0.297507D-01 0.794627D+00 0.103429D+02 0.953752D+00 -.104717D-01 0.3511129D-01 0.424973D+01
 0.788790D-01 0.590962D-02 0.803801D+00 0.116692D+02 0.974974D+00 -.113280D-01 0.252429D-01 0.424973D+01
 0.809559D-01 -.180480D-01 0.811824D+00 0.983600D+01 0.995671D+00 -.122252D-01 0.386232D-01 0.424973D+01
 0.832481D-01 -.444971D-01 0.819333D+00 0.953872D+01 0.101788D+01 -.132609D-01 0.405979D-01 0.424973D+01
 0.855402D-01 -.708145D-01 0.825422D+00 0.921927D+01 0.103938D+01 -.143243D-01 0.425972D-01 0.424973D+01
 0.878324D-01 -.969333D-01 0.830100D+00 0.887712D+01 0.106013D+01 -.154696D-01 0.446183D-01 0.424973D+01
 0.901246D-01 -.122781D+00 0.833369D+00 0.850910D+01 0.108006D+01 -.166431D-01 0.466585D-01 0.424973D+01
 0.924167D-01 -.148216D+00 0.835292D+00 0.812444D+01 0.109912D+01 -.178605D-01 0.487159D-01 0.424973D+01
 0.947089D-01 -.173114D+00 0.835946D+00 0.773360D+01 0.111730D+01 -.191196D-01 0.507884D-01 0.424973D+01
 0.958555D-01 -.185347D+00 0.835813D+00 0.753570D+01 0.112605D+01 -.197646D-01 0.518295D-01 0.424973D+01
 0.970010D-01 -.197427D+00 0.835380D+00 0.733627D+01 0.113457D+01 -.204199D-01 0.528735D-01 0.424973D+01
 0.981471D-01 -.209347D+00 0.834655D+00 0.7135359D+01 0.114286D+01 -.210854D-01 0.539199D-01 0.424973D+01
 0.992932D-01 -.221104D+00 0.833643D+00 0.693327D+01 0.115093D+01 -.217610D-01 0.549685D-01 0.424973D+01
 0.101791D+00 -.245960D+00 0.830486D+00 0.647682D+01 0.116768D+01 -.232572D-01 0.572646D-01 0.424973D+01
 0.103040D+00 -.257907D+00 0.828439D+00 0.623569D+01 0.117562D+01 -.240122D-01 0.584195D-01 0.424973D+01
 0.104289D+00 -.269528D+00 0.822609D+00 0.598678D+01 0.118325D+01 -.247720D-01 0.595786D-01 0.424973D+01
 0.106787D+00 -.291761D+00 0.820519D+00 0.546611D+01 0.119757D+01 -.263045D-01 0.619082D-01 0.424973D+01
 0.109285D+00 -.3121591D+00 0.8113849D+00 0.491603D+01 0.121054D+01 -.278516D-01 0.642521D-01 0.424973D+01
 0.110340D+00 -.331955D+00 0.806155D+00 0.433811D+01 0.122210D+01 -.294105D-01 0.666086D-01 0.424973D+01
 0.1114281D+00 -.349796D+00 0.797515D+00 0.373425D+01 0.123219D+01 -.309781D-01 0.689761D-01 0.424973D+01
 0.1115530D+00 -.358129D+00 0.792867D+00 0.342325D+01 0.123666D+01 -.317642D-01 0.701635D-01 0.424973D+01
 0.1116779D+00 -.365062D+00 0.788012D+00 0.310662D+01 0.124074D+01 -.325514D-01 0.713529D-01 0.424973D+01
 0.1118028D+00 -.373590D+00 0.782964D+00 0.278471D+01 0.124442D+01 -.3333391D-01 0.725442D-01 0.424973D+01
 0.1119277D+00 -.380710D+00 0.777729D+00 0.245760D+01 0.124769D+01 -.341767D-01 0.737370D-01 0.424973D+01
 0.1121073D+00 -.390150D+00 0.769917D+00 0.1985558D+01 0.125168D+01 -.352535D-01 0.745566D-01 0.424973D+01
 0.1122868D+00 -.398567D+00 0.761812B2D+00 0.151956D+01 0.125483D+01 -.363617D-01 0.771883D-01 0.424973D+01
 0.1124664D+00 -.430110D+00 0.753483D+00 0.105838D+01 0.125714D+01 -.374515D-01 0.789286D-01 0.424973D+01
 0.1126460D+00 -.412420D+00 0.744938D+00 0.602336D+00 0.125863D+01 -.385210D-01 0.806784D-01 0.424973D+01
 0.130052D+00 -.422365D+00 0.727449D+00 0.293920D+00 0.125918D+01 -.405912D-01 0.842073D-01 0.424973D+01
 0.133643D+00 -.428506D+00 0.709727D+00 -.116872D+01 0.125655D+01 -.425566D-01 0.877755D-01 0.424973D+01
 0.1372343D+00 -.431023D+00 0.691938D+00 -.202176D+01 0.125081D+01 -.44019D-01 0.913824D-01 0.424973D+01
 0.140825D+00 -.430110D+00 0.674222D+00 -.285319D+01 0.124205D+01 -.461118D-01 0.950287D-01 0.424973D+01
 0.152338D+00 -.405980D+00 0.618574D+00 0.537489D+01 0.119448D+01 -.505117D-01 0.106965D+00 0.424973D+01
 0.163850D+00 -.353701D+00 0.565111D+00 0.768383D+01 0.111911D+01 -.528892D-01 0.119213D+00 0.424973D+01
 0.176053D+00 -.274274D+00 0.508675D+00 0.990511D+01 0.101156D+01 -.526995D-01 0.132310D+00 0.424973D+01
 0.188255D+00 -.177113D+00 0.447743D+00 -.118855D+02 0.878354D+00 -.492145D-01 0.145135D+00 0.424973D+01
 0.201098D+00 -.651678D-01 0.372658D+00 -.136764D+02 0.713855D+00 -.416415D-01 0.157655D+00 0.424973D+01

0.213942D+00 0.462310D-01 0.280079D+00 - 1510422D+02 0.528604D+00 - .298064D-01 0. 168238D+00 0. 424973D+01
 0.227373D+00 0.149483D+00 0.162259D+00 - 161196D+02 0.318320D+00 - .130823D-01 0. 176033D+00 0. 424973D+01
 0.240804D+00 0.225670D+00 0.239814D-01 - 165503D+02 0.982212D-01 0.747108D-02 0. 179154D+00 0. 424973D+01
 0.268731D+00 0.256910D+00 - 290917D+00 - 152834D+02 - 353242D+00 0.577280D-01 0. 166021D+00 0. 424973D+01
 0.283169D+00 0.200215D+00 - 433148D+00 - 135001D+02 - 561876D+00 0.829434D-01 0. 148120D+00 0. 424973D+01
 0.297608D+00 0.103632D+00 - 540078D+00 - 111017D+02 - 740113D+00 0.105077D+00 0. 123318D+00 0. 424973D+01
 0.312462D+00 - 193165D-01 - 602567D+00 - 820353D+01 - 883886D+00 0.122966D+00 0. 9233383D-01 0. 424973D+01
 0.322317D+00 - 146282D+00 - 613636D+00 - 512103D+01 - 982921D+00 0.134842D+00 0. 121693D-01 0. 424973D+01
 0.342527D+00 - 263387D+00 - 579086D+00 - 202908D+01 - 103707D+01 0.140922D+00 0. 216902D-01 0. 424973D+01
 0.357736D+00 - 354192D+00 - 511427D+00 0.792840D+00 - 104603D+01 0.141497D+00 - 135646D-01 0. 424973D+01
 0.388740D+00 - 450327D+00 - 331428D+00 0.524232D+01 - 947551D-00 0.131334D+00 - 762276D-01 0. 424973D+01
 0.420200D+00 - 457438D+00 - 158265D+00 0.782262D+01 - 737166D+00 0.115318D+00 - 122906D+00 0. 424973D+01
 0.451988D+00 - 427457D+00 - 268749D-01 0.877718D+01 - 469554D+00 0.101779D+00 - 152849D+00 0. 424973D+01
 0.4839774D+00 - 400436D+00 0.612895D-01 0.855444D+01 - 189818D+00 0.943708D-01 - 167906D+00 0. 424973D+01
 0.516026D+00 - 393869D+00 0.115102D+00 0.756909D+01 0.701332D-01 0.931726D-01 - 10402D+00 0. 424973D+01
 0.548012D+00 - 409814D+00 0.142721D+00 0.615095D+01 0.290350D+00 0.964459D-01 - 162652D+00 0. 424973D+01
 0.579800D+00 - 442947D+00 0.150423D+00 0.455641D-01 0.460749D+00 0.102001D+00 - 147021D+00 0. 424973D+01
 0.611260D+00 - 486010D+00 0.143729D+00 0.297759D+01 0.579070D+00 0.108001D+00 - 125922D+00 0. 424973D+01
 0.642264D+00 - 532731D+00 0.127894D+00 0.154430D+01 0.648737D+00 0.113269D+00 - 101655D+00 0. 424973D+01
 0.6726683D+00 - 578866D+00 0.107616D+00 0.3277736D+00 0.671678D+00 0.117269D+00 - 762220D-01 0. 424973D+01
 0.7023920D+00 - 622120D+00 0.865759D-01 - 649905D+00 0.671367D+00 0.119924D+00 - 511879D-01 0. 424973D+01
 0.731269D+00 - 661587D+00 0.672384D-01 - 140108D+01 0.641298D+00 0.121406D+00 - 1276508D-01 0. 424973D+01
 0.779196D+00 - 697161D+00 0.509411D-01 - 195821D+01 0.594031D+00 0.121974D+00 - 628545D-02 0. 424973D+01
 0.786058D+00 - 729117D+00 0.3814468D-01 - 236146D+01 0.535753D+00 0.121878D+00 0.125772D-01 0. 424973D+01
 0.811745D+00 - 757855D+00 0.287846D-01 - 265097D+01 0.471201D+00 0.121309D+00 0.288546D-01 0. 424973D+01
 0.823947D+00 - 770989D+00 0.252000D-01 - 276241D+01 0.438158D+00 0.120889D+00 0.359703D-01 0. 424973D+01
 0.836150D+00 - 783833D+00 0.221604D-01 - 286019D+01 0.403840D+00 0.120377D+00 0.426786D-01 0. 424973D+01
 0.847662D+00 - 795722D+00 0.196806D-01 - 294171D+01 0.370434D+00 0.119803D+00 0.486448D-01 0. 424973D+01
 0.859175D+00 - 807409D+00 0.175372D-01 - 301425D+01 0.336142D+00 0.119136D+00 0.542603D-01 0. 424973D+01
 0.864562D+00 - 812840D+00 0.165799D-01 - 304540D+01 0.319819D+00 0.118787D+00 0.567761D-01 0. 424973D+01
 0.869949D+00 - 818264D+00 0.155964D-01 - 307478D+01 0.303333D+00 0.118408D+00 0.592276D-01 0. 424973D+01
 0.875336D+00 - 823683D+00 0.145885D-01 - 310201D+01 0.286697D+00 0.118001D+00 0.616133D-01 0. 424973D+01
 0.880723D+00 - 829095D+00 0.135627D-01 - 312986D+01 0.269907D+00 0.117564D+00 0.639326D-01 0. 424973D+01
 0.885719D+00 - 834111D+00 0.125957D-01 - 313643D+01 0.254242D+00 0.117134D+00 0.660192D-01 0. 424973D+01
 0.890715D+00 - 839124D+00 0.115276D-01 - 31310773D+01 0.238633D+00 0.116681D+00 0.680398D-01 0. 424973D+01
 0.895711D+00 - 844127D+00 0.102229D-01 - 305037D+01 0.223236D+00 0.116205D+00 0.699552D-01 0. 424973D+01
 0.900707D+00 - 849112D+00 0.856482D-02 - 296755D+01 0.208195D+00 0.115708D+00 0.718867D-01 0. 424973D+01
 0.909875D+00 - 858192D+00 0.445730D-02 - 279932D+01 0.181765D+00 0.14742D+00 0.751864D-01 0. 424973D+01
 0.914460D+00 - 862695D+00 0.191880D-02 - 272341D+01 0.169108D+00 0.114237D+00 0.767516D-01 0. 424973D+01
 0.919044D+00 - 8897007D+00 0.89200D-01 - 310731D-01 0.816270D-01 0.109579D+00 0.875244D-01 0. 424973D+01
 0.9273510D+00 - 902879D+00 - 264991D-01 0.156792D+00 0.113717D+00 0.108572D+00 0.892523D-01 0. 424973D+01
 0.936595D+00 - 908599D+00 - 400482D-01 - 201426D+01 0.681932D-01 0.108572D+00 0.898815D-01 0. 424973D+01
 0.943071D+00 - 883164D+00 - 142941D-01 - 2237676D+01 0.135327D+00 0.112728D+00 0.908376D-01 0. 424973D+01
 0.94460D+00 - 890149D+00 - 220652D-01 - 225011D+01 0.978473D-01 0.110659D+00 0.833674D-01 0. 424973D+01
 0.950484D+00 - 8997007D+00 - 310731D-01 - 212632D-01 0.816270D-01 0.103704D+00 0.854861D-01 0. 424973D+01
 0.9566971D+00 - 902879D+00 - 264991D-01 0.156792D+00 0.113717D+00 0.102168D-01 0.97342D-01 0. 424973D+01
 0.963458D+00 - 908599D+00 - 501623D-01 - 189329D+01 0.55515D-01 0.107501D+00 0.909376D-01 0. 424973D+01
 0.974528D+00 - 917900D+00 - 877924D+01 0.357342D-01 0.105513D+00 0.937334D-01 0. 424973D+01
 0.983647D+00 - 924995D+00 - 899200D-01 - 149263D+01 0.212577D-01 0.103704D+00 0.959865D-01 0. 424973D+01
 0.990780D+00 - 930066D+00 - 107425D+00 - 132901D+01 0.111845D-01 0.102168D-01 0.97342D-01 0. 424973D+01
 0.995895D+00 - 933375D+00 - 121303D+00 - 120133D+01 0.470925D-02 0.100992D+00 0.989885D-01 0. 424973D+01
 0.998973D+00 - 935209D+00 - 130235D+00 - 111995D+01 0.113587D-02 0.100252D+00 0.997464D-01 0. 424973D+01
 0.100000D+00 - 935792D+00 - 133318D+01 - 109200D+01 0.000000D+00 0.100000D+00 0.100000D+00 0.424973D+01

Chapter IX

IX.1 Input-Output Manual for the In-Plane Linear Eigenproblem Solution Program, RCLINDY2

This program is written in FORTRAN 77 and allows the interactive solution of the in-plane linear dynamic eigenproblem for a compliant riser with a planar static configuration, as described in [4]. The initial ($\epsilon=0$) approximation is prepared by RCFORCE. This program is particularly useful in the case of zero current velocity where the asymptotic theory is not valid.

Before executing the program the user must make sure that:

- o Devices five and six correspond to input/output from the terminal
- o Device eight must be associated with the data file containing geometric, structural and hydrodynamic characteristics of the compliant riser created by a previous execution of the data preprocessing program RCINPUT. This file has a logical record length of 80 characters.
- o Device nine must be associated with an output data file with logical record length of 132 characters which upon completion of the program contains a complete list of the input, initial approximation and the converged solution of the problem.
- o Device ten must be associated with the input data file containing the non-dimensional static planar solution created by RCSTAT2D. This file has a logical record length of 132 characters.

- o Device eleven must be associated with an output file with logical record length of 117 characters. This file contains the converged solution of the problem in a concise form appropriate for use in the preparation of illustrations and/or as an input to RCLINDY1 from device twelve.
- o Device twelve must be associated with an input file containing the initial approximate solution of the undamped forced in-plane linear dynamic problem, resulting from a previous execution of RCFORCE.

Input Variables [Free Format]

At the beginning of the program the static 2-D solution created from RCSTAT2D is read from device ten. The values of the mean current velocity in the x direction and the maximum non-dimensional static effective tension are printed on the terminal. If the number of static discretization points is less than four or greater than the value of MNP (maximum number of allowable points), the program halts execution.

Then the geometric, structural and hydrodynamic characteristics of the compliant riser are read from the file associated with device eight. The user has the opportunity to review this data on his terminal and decide either to continue with the program or stop, see [3].

In this case an initial ($\epsilon=0$) approximation to the solution is available from a previous run of RCFORCE. The program reads the information from the previous run concerning the number of discretization points NP, the mode number and the corresponding frequency of excitation. If the number of

discretization points is greater than MNP, the program stops execution. The same happens if NP is less than the number of static discretization points. Then the user has the option to review the initial approximation by displaying it on his terminal. The assumed boundary condition for the curvature at the lower ball point of the riser is printed at the terminal. Subsequently, the initial solution provided by RCFORCE is printed in the file associated with the device nine. The following quantities are printed:

- o Number of discretization points NP
- o Natural frequency (rad/s)
- o Order of point (starting from the lower end)
- o Non-dimensional arc length
- o Non-dimensional dynamic effective tension, T_1
- o Non-dimensional shear force in the $\vec{\xi}$ direction, Q_1^{ξ}
- o Non-dimensional component of rate of rotation about \vec{n}, Ω_1^n
- o Euler angle ϕ_1
- o Non-dimensional displacement p
- o Non-dimensional displacement q
- o Non-dimensional natural frequency

Subsequently, the program prints the maximum of the absolute value of the initial rate of rotation about \vec{n} direction. This number is used to estimate a reasonable tolerance for convergence of iterations. The user is requested to input a tolerance as a fraction of this number; e.g., 0.01 for 1% tolerance or better for all quantities of interest. Typically, Ω_1^n will be determined with an accuracy a little better than the above fraction

and all other variables will be even more accurate. After the run the accuracy of the results should be checked. The user is given the opportunity to stop the program at this point if he wishes.

If he chooses to continue, he is asked to input the initial increment of the continuation parameter $\delta\epsilon$ (condition $0 < \delta\epsilon \leq 1$). If no continuation is required, then use $\delta\epsilon = 1$. If continuation is required, then $\delta\epsilon = 0.1$ will usually suffice. For large changes between initial and final problem, a smaller value of $\delta\epsilon$, e.g., 0.05 or 0.25 might be necessary.

After the initial approximation is fully defined, the program enters NAG subroutine D02RAF, which provides the iterative numerical solution of the problem. The manual of this subroutine is included in Chapter X. The input value of IFAIL is 111, indicating that the soft failure option with error message and monitoring information printing has been selected. Error messages are directed to device six. Once the execution of D02RAF is completed, the current value of IFAIL is printed on the terminal. For an explanation of this parameter, see NAG manual for D02RAF, Chapter X. Only if IFAIL = 0 or IFAIL = 4 the execution of the program continues with printout of the riser characteristics and the final solution of the problem. Otherwise, the program stops. When IFAIL ≠ 0, the user should consult the manual quoted above for interpretation of the results and a possible course of action. If IFAIL = 0 or 4, the riser characteristics and the final solution of the problem are printed in the file associated with device nine. For the riser characteristics, the format used in the static program RCSTAT2D is also used here. This is described in [3], (pp. 23-25).

Next the non-dimensional in-plane solution is written in the file associated with device nine. The mode number, natural frequency and final number of points NP at which the solution is available is printed first. Then the following data is printed for I=1 to NP:

$s, T_1, Q_1^{\xi}, Q_1^{\eta}, \phi_1, p, q, \sigma$ with

FORMAT (8(1X,D10.4)). Next the maximum (non-dimensional) estimated errors of $T_1, Q_1^{\xi}, Q_1^{\eta}, \phi_1, p, q, \sigma$ provided by the NAG subroutine D02RAF are printed.

Subsequently, the following data is written in the file associated with device eleven:

MODE, NP, SIGMAD, XTOP, YTOP: Mode number, number of points at which converged solution is available and dimensional natural frequency (in rad/s) and x and y displacements at the top with

FORMAT (1X,I2,1X,I3,3(1X,D10.4)) and

For I=1 to NP: the non-dimensional

$s, T_1, Q_1^{\xi}, Q_1^{\eta}, \phi_1, p, q, \sigma$ and the dimensional current velocity $V_c(s)$ with

FORMAT (9(1X,D12.6)). The data written in the file associated with device eleven due to their simple form can be easily used as input to plotting programs. In addition, this data is useful in subsequent runs of RCLINDY1 if such an option is selected.

IX.2 Sample Run, Input and Output

IX.2.1 Zero Current Excitation

IX.2.1.1 Interactive Session

INPUT

Device 5 = TERMINAL
Device 8 = RCJ0A01 DATA A*
Device 10 = RC2DZERR DATA A**
Device 12 = RC2DZEE DATA A***

OUTPUT

Device 6 = TERMINAL
Device 9 = RC2DZ1 DATA A
Device 11 = RC2DZ1A DATA A

* File included in Chapter II.

** Files included in Chapter VI.

*** File included in Chapter VIII.

rc1rndy2 rcjooa1 rc2d2f1 rc2d2err rc2d2la rc2d2ee
 FI 8 DISK RCJOOA1 DATA A (RECFM FB LRECL 80 BLKSIZE 800
 FI 9 DISK RC2D21 DATA A (RECFM FB LRECL 132 BLKSIZE 1320
 FI 10 DISK RC2D2ERR DATA A (RECFM FB LRECL 132 BLKSIZE 1320
 FI 11 DISK RC2D21 DATA A (RECFM FB LRECL 117 BLKSIZE 1170
 FI 12 DISK RC2D2EE DATA A (RECFM FB LRECL 117 BLKSIZE 1170
 GLOBAL XILIB VFORTLIB CMSLIB NAG1 NAG2
 LOAD RCLINDY2 (START
 EXECUTION BEGINS...
 MNP=151
 2-D STATIC SOLUTION FROM DEVICE 10
 NP = 54
 MEAN CONVENT. VELOCITY IN THE X DIRECTION IN M/S . VM = 0.0000000D+00
 2-D STATIC SOLUTION SUCCESSFULLY READ
 MAXIMUM STATIC EFFECTIVE TENSION/WA*L = 0.5547D+00
 DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS
 IF YES INPUT 1 . IF NO INPUT 0
 ?
 0
 MNP=151
 INITIAL APPROXIMATION FROM DEVICE 12
 NP = 107 MODE NUMBER = 1 SIGMAD = 0.5000D-01
 X AT TOP = 0.0000D+00 Y AT TOP = 0.7931D+00
 DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION
 IF YES INPUT 1
 ?
 2
 ASSUMED BOUNDARY CONDITION
 OMEGA ETA (0) = 0.119780D+02
 MAXIMUM ABSOLUTE VALUE OF N-D OMEGA ETA IS = 0.1655030D+02
 THIS NUMBER CAN BE USED TO ESTIMATE
 A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS
 INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE
 E.G. INPUT 0.01 OR 0.1 IF THE INITIAL ESTIMATE FOR OMEGA ETA
 IS SMALL
 AFTER THE RUN CHECK THE ACCURACY OF THE SOLUTION
 ?
 .01
 IF YOU WISH TO STOP INPUT 0
 IF YOU WISH TO CONTINUE WITH ITERATIONS INPUT 1
 ?
 1
 INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS
 IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.0D
 IF CONTINUATION IS REQUIRED THEN 0.0D < DELEPS < 1.0D
 RECOMMENDATION :
 USUALLY DELEPS = 0.10D WILL SUFFICE
 FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM
 A SMALLER VALUE OF DELEPS, E.G. 0.05D0 OR 0.025D0 MIGHT BE
 NECESSARY
 ?
 .05

DO2RAF MONITORING INFORMATION
 MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 107

CORRECTION NUMBER 0 RESIDUAL SHOULD BE .I.E. 1.00D+00
 ITERATION NUMBER 0 RESIDUAL = 9.20D-02
 SQUARED NORM OF CORRECTION = 2.12D+01
 SQUARED NORM OF GRADIENT = 6.56D-01
 SCALAR PRODUCT OF CORRECTION AND GRADIENT * 8.47D-03
 ITERATION NUMBER 1 RESIDUAL = 2.35D-03
 CONTINUATION PARAMETER EPSILON = 1.00D-01 DELEPS = 1.00D-01

MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 107
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .I.E. 1.00D+00
 ITERATION NUMBER 0 RESIDUAL = 2.47D-02

CONTINUATION PARAMETER EPSILON = 3.00D-01 DELEPS = 2.00D-01

MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 107
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .I.E. 1.00D+00
 ITERATION NUMBER 0 RESIDUAL = 1.23D-01

CONTINUATION PARAMETER EPSILON = 7.00D-01 DELEPS = 4.00D-01

MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 107
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .I.E. 1.00D+00
 ITERATION NUMBER 0 RESIDUAL = 6.00D-01

CONTINUATION PARAMETER EPSILON = 1.00D+00 DELEPS = 8.00D-01

MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 107
 CORRECTION NUMBER 0 RESIDUAL SHOULD BE .I.E. 1.03D-04
 ITERATION NUMBER 0 RESIDUAL = 3.09D+00
 SQUARED NORM OF CORRECTION = 5.46D+03
 SQUARED NORM OF GRADIENT = 4.63D+01
 SCALAR PRODUCT OF CORRECTION AND GRADIENT = 9.55D+00
 ITERATION NUMBER 1 RESIDUAL = 4.70D-01
 SQUARED NORM OF CORRECTION = 5.81D+02
 SQUARED NORM OF GRADIENT = 7.23D-01
 SCALAR PRODUCT OF CORRECTION AND GRADIENT = 2.21D-01

ITERATION NUMBER 2 RESIDUAL = 1.14D-01
 SQUARED NORM OF CORRECTION = 7.34D+01
 SQUARED NORM OF GRADIENT = 2.61D-02
 SCALAR PRODUCT OF CORRECTION AND GRADIENT = 1.30D-02

ITERATION NUMBER 3 RESIDUAL = 3.67D-03
 SQUARED NORM OF CORRECTION = 6.09D-02
 SQUARED NORM OF GRADIENT = 2.54D-05
 SCALAR PRODUCT OF CORRECTION AND GRADIENT = 1.34D-05
 ITERATION NUMBER 4 RESIDUAL = 1.23D-05

MESH SELECTION
 NUMBER OF NEW POINTS 2

CORRECTION NUMBER 0 ESTIMATED MAXIMUM ERROR = 2.50D-01
 ESTIMATED ERROR BY COMPONENTS
 1.42D-02 1.21D-02 2.50D-01 1.85D-02 2.49D-03 2.28D-03 8.54D-02

MONITORING NEWTON ITERATION
 NUMBER OF POINTS IN CURRENT MESH = 107

CORRECTION NUMBER 1 RESIDUAL SHOULD BE .LE. 2.60D-04
ITERATION NUMBER 0 RESIDUAL = 1.33D-01
ITERATION NUMBER 1 RESIDUAL = 2.04D-04

CORRECTION NUMBER 1 ESTIMATED MAXIMUM ERROR = 2.33D-02
ESTIMATED ERROR BY COMPONENTS
 1.19D-03 1.70D-03 2.33D-02 1.53D-03 1.34D-04 2.08D-04 1.98D-05
IFAIL = 0
R: T=4.60/4.89 14:44:26
cp spool console stop close

IX.2.1.2 RC2DZ1 DATA A

INITIAL CONDITION FOR EPS=0. AND NP = 86 POINTS, NATURAL FREQUENCY = 0.36000-01 RAD/S

	ARC	TENSION	QXI	OMEGA	ETA	PHI	P	0	SIGMA	
1	0.00000D+00	- .682778D+00	0 .314342D+00	- .774100D+01	0 .000000D+00	0 .0000000D+00	0 .0000000D+00	0 .305981D+01		
2	0 .102730D-02	- .687134D+00	0 .301582D+00	- .782897D+01	- .799818D-02	0 .122874D-07	- .410039D-05	- .305981D+01		
3	0 .410499D-02	- .699326D+00	0 .262483D+00	- .808540D+01	- .324938D-01	0 .828869D-02	- .662013D-04	- .305981D+01		
4	0 .920424D-02	- .716443D+00	0 .194839D+00	- .847111D+01	- .748619D-01	0 .979655D-05	- .339739D-03	- .305981D+01		
5	0 .163526D-01	- .732786D+00	0 .959157D-01	- .892174D+01	- .136949D+00	0 .575489D-04	- .109111D-02	0 .305981D+01		
6	0 .254724D-01	- .739021D+00	- .358365D+01	- .933652D+01	- .220342D+00	0 .230081D-03	- .270509D-02	0 .305981D+01		
7	0 .365416D-01	- .722166D+00	- .198251D+00	- .959176D+01	- .325576D+00	0 .718868D-03	- .566983D-02	0 .305981D+01		
8	0 .430286D-01	- .699424D+00	- .291628D+00	- .959877D+01	- .387676D+00	0 .120444D-02	- .791453D-02	- .305981D+01		
9	0 .495156D-01	- .667314D+00	- .381809D+00	- .9488886D+01	- .449650D+00	0 .187671D-02	- .105192D-01	0 .305981D+01		
10	0 .569281D-01	- .620037D+00	- .478771D+00	- .921155D+01	- .519070D+00	0 .290713D-02	- .139106D-01	0 .305981D+01		
11	0 .643406D-01	- .562696D+00	- .567177D+00	- .874032D+01	- .585725D+00	0 .425471D-02	- .177038D-01	0 .305981D+01		
12	0 .684944D-01	- .526958D+00	- .612186D+00	- .839614D+01	- .621334D+00	0 .515767D-02	- .199861D-01	0 .305981D+01		
13	0 .726482D-01	- .489201D+00	- .653526D+00	- .8005285D+01	- .655410D+00	0 .616980D-02	- .223680D-01	0 .305981D+01		
14	0 .768021D-01	- .449771D+00	- .691085D+00	- .756087D+01	- .6877749D+00	0 .729479D-02	- .248377D-01	0 .305981D+01		
15	0 .809559D-01	- .408963D+00	- .724823D+00	- .707890D+01	- .718174D+00	0 .853741D-02	- .273820D-01	0 .305981D+01		
16	0 .832481D-01	- .386073D+00	- .741734D+00	- .678680D+01	- .734067D+00	0 .927202D-02	- .288138D-01	0 .305981D+01		
17	0 .855402D-01	- .363071D+00	- .757383D+00	- .646803D+01	- .749264D+00	0 .100399D-01	- .302630D-01	0 .305981D+01		
18	0 .901246D-01	- .317022D+00	- .784865D+00	- .5738882D+01	- .777288D+00	0 .116763D-01	- .332040D-01	0 .305981D+01		
19	0 .947089D-01	- .271659D+00	- .807335D+00	- .491859D+01	- .801741D+00	0 .134402D-01	- .361869D-01	0 .305981D+01		
20	0 .970010D-01	- .249493D+00	- .816773D+00	- .449000D+00	- .8122316D+00	0 .143668D-01	- .376885D-01	0 .305981D+01		
21	0 .992932D-01	- .227733D+00	- .825067D+00	- .404940D+01	- .8223316D+00	0 .153233D-01	- .391930D-01	0 .305981D+01		
22	0 .104289D+00	- .182692D+00	- .839085D+00	- .301144D+01	- .8400009D+00	0 .174743D-01	- .424844D-01	0 .305981D+01		
23	0 .106787D+00	- .161858D+00	- .844002D+00	- .243847D+01	- .846823D+00	0 .1856668D-01	- .441370D-01	0 .305981D+01		
24	0 .109285D+00	- .142232D+00	- .847629D+00	- .183390D+01	- .852166D+00	0 .196696D-01	- .457904D-01	0 .305981D+01		
25	0 .114281D+00	- .1062953D+00	- .851291D+00	- .537034D+00	- .858136D+00	0 .218961D-01	- .490903D-01	0 .305981D+01		
26	0 .119277D+00	- .774215D-01	- .850694D+00	- .863407D+00	- .857354D+00	0 .241378D-01	- .523615D-01	0 .305981D+01		
27	0 .126460D+00	- .471653D-01	- .843344D+00	- .293646D+01	- .843703D+00	0 .272590D-01	- .570130D-01	0 .305981D+01		
28	0 .130052D+00	- .380178D-01	- .837341D+00	- .395940D+01	- .831317D+00	0 .287194D-01	- .593192D-01	0 .305981D+01		
29	0 .133643D+00	- .327818D-01	- .830231D+00	- .497456D+01	- .815271D+00	0 .301015D-01	- .616048D-01	0 .305981D+01		
30	0 .137134D+00	- .313719D-01	- .822166D+00	- .598029D+01	- .795599D+00	0 .313940D-01	- .638646D-01	0 .305981D+01		
31	0 .140825D+00	- .336708D-01	- .813276D+00	- .697569D+01	- .772333D+00	0 .325862D-01	- .660938D-01	0 .305981D+01		
32	0 .163850D+00	- .129707D+00	- .760828D+00	- .206752D+02	- .540587D+00	0 .372029D-01	- .792239D-01	0 .305981D+01		
33	0 .188255D+00	- .347017D+00	- .719605D+00	- .187342D+02	- .150585D+00	0 .348549D-01	- .889087D-01	0 .305981D+01		
34	0 .213942D+00	- .624539D+00	- .616669D+00	- .798802D+00	- .17286D+02	0 .392717D+00	- .237022D-01	0 .305981D+01		
35	0 .240804D+00	- .841606D+00	- .261003D-01	- .0 .252573D+02	- .0 .105169D+01	0 .595840D-02	- .734984D-01	0 .305981D+01		
36	0 .268731D+00	- .852259D+00	0 .433652D+00	- .0 .233207D+02	- .0 .174017D+01	0 .976833D-02	- .335446D-01	0 .305981D+01		
37	0 .281692D+00	- .103472D+00	- .8727765D+00	- .0 .361770D+02	- .0 .205901D+01	0 .280275D+01	- .420155D-02	0 .305981D+01		
38	0 .297608D+00	- .616669D+00	- .798802D+00	- .0 .171286D+02	- .0 .233286D+01	- .108237D-01	- .0 .295655D-01	0 .305981D+01		
39	0 .312462D+00	- .439603D+00	- .893909D+00	- .0 .128362D+02	- .0 .255601D+01	- .237022D-01	- .0 .671802D-01	0 .305981D+01		
40	0 .327317D+00	- .261727D+00	0 .915401D+00	- .0 .825161D+02	- .0 .271276D+01	- .193215D+01	- .0 .105858D+00	0 .305981D+01		
41	0 .342527D+00	- .331277D-01	- .157786D+02	- .0 .143950D+01	- .0 .128696D-01	- .0 .149323D+00	- .0 .409177D+00	0 .305981D+01		
42	0 .357736D+00	- .0 .108843D-01	- .783740D+00	- .0 .667202D+00	- .0 .282461D+01	- .0 .519698D-01	- .0 .182278D+00	0 .305981D+01		
43	0 .387840D+00	- .0 .101229D+00	- .548494D+00	- .0 .768954D+01	- .0 .268851D+02	- .0 .101210D+00	- .0 .251133D+00	0 .305981D+01		
44	0 .420200D+00	- .0 .434964D-01	- .0 .323486D+00	- .0 .122972D+02	- .0 .236787D+01	- .0 .146269D+00	- .0 .311250D+00	0 .305981D+01		
45	0 .451988D+00	- .0 .858558D-01	- .0 .152831D+00	- .0 .148099D+02	- .0 .193215D+01	- .0 .179232D+00	- .0 .363857D+00	0 .305981D+01		
46	0 .483974D+00	- .0 .222449D+00	- .0 .331277D-01	- .0 .157786D+02	- .0 .280275D+01	- .0 .143950D+00	- .0 .409177D+00	0 .305981D+01		
47	0 .516026D+00	- .0 .333086D+00	- .0 .515986D-01	- .0 .156528D+02	- .0 .933331D+00	- .0 .196950D+00	- .0 .445787D+00	0 .305981D+01		
48	0 .5480112D+00	- .0 .406858D+00	- .0 .112235D+00	- .0 .147461D+02	- .0 .445410D+01	- .0 .109080D+00	- .0 .471598D+00	0 .305981D+01		
49	0 .579800D+00	- .0 .445478D+00	- .0 .153568D+00	- .0 .132982D+02	- .0 .150734D-02	- .0 .173610D+00	- .0 .311250D+00	0 .305981D+01		
50	0 .611260D+00	- .0 .456470D+00	- .0 .177840D+00	- .0 .115173D+02	- .0 .152249D+00	- .0 .1485065D+00	- .0 .305981D+01			
51	0 .642264D+00	- .0 .448792D+00	- .0 .187554D+00	- .0 .958611D+01	- .0 .129999D+00	- .0 .474994D+00	- .0 .305981D+01			
52	0 .672683D+00	- .0 .430332D+00	- .0 .186344D+00	- .0 .765093D+01	- .0 .982076D+00	- .0 .109098D+00	- .0 .454322D+00	- .0 .305981D+01		
53	0 .702392D+00	- .0 .406903D+00	- .0 .178618D+00	- .0 .580961D+01	- .0 .118183D+01	- .0 .907501D-01	- .0 .426196D+00	- .0 .305981D+01		

54 0.731269D+00 - 382227D+00 - 168924D+00 - .410868D+01 - 132481D+01 0.753517D-01 0.392999D+00 0.305981D+01
 55 0.759196D+00 - 358394D+00 - 161587D+00 - 254767D+01 - 141759D+01 0.627712D-01 0.356915D+00 0.305981D+01
 56 0.786058D+00 - 336366D+00 - 160751D+00 - 108665D+01 - 146638D+01 0.525920D-01 0.319823D+00 0.305981D+01
 57 0.811745D+00 - 316412D+00 - 170606D+00 0.346047D+00 - 147608D+01 0.443443D-01 0.283276D+00 0.305981D+01
 58 0.823947D+00 - 307238D+00 - 180880D+00 0.107145D+01 - 146747D+01 0.407920D-01 0.265858D+00 0.305981D+01
 59 0.836150D+00 - 298177D+00 - 196003D+00 0.184899D+01 - 144972D+01 0.374477D-01 0.248558D+00 0.305981D+01
 60 0.847662D+00 - 289635D+00 - 215850D+00 0.265059D+01 - 142389D+01 0.344303D-01 0.232459D+00 0.305981D+01
 61 0.853419D+00 - 285311D+00 - 228215D+00 0.308352D+01 - 140740D+01 0.329564D-01 0.224526D+00 0.305981D+01
 62 0.859175D+00 - 280931D+00 - 242439D+00 0.354235D+01 - 1388834D+01 0.315121D-01 0.216689D+00 0.305981D+01
 63 0.864562D+00 - 276709D+00 - 257639D+00 0.398862D+01 - 136804D+01 0.301717D-01 0.209458D+00 0.305981D+01
 64 0.869949D+00 - 272265D+00 - 274876D+00 0.448468D+01 - 134521D+01 0.2888295D-01 0.202341D+00 0.305981D+01
 65 0.875336D+00 - 267573D+00 - 294348D+00 0.500243D+01 - 131967D+01 0.274901D-01 0.195352D+00 0.305981D+01
 66 0.880723D+00 - 262588D+00 - 316320D+00 0.556601D+01 - 129112D+01 0.261542D-01 0.188505D+00 0.305981D+01
 67 0.885719D+00 - 257680D+00 - 3390800D+00 0.607692D+01 - 126211D+01 0.249263D-01 0.182294D+00 0.305981D+01
 68 0.890715D+00 - 252482D+00 - 364024D+00 0.652424D+01 - 123061D+01 0.237159D-01 0.176231D+00 0.305981D+01
 69 0.895711D+00 - 246964D+00 - 390963D+00 0.691670D+01 - 119701D+01 0.222523D00 0.170325D+00 0.305981D+01
 70 0.900707D+00 - 241100D+00 - 419726D+00 0.725354D+01 - 116160D+01 0.213478D-01 0.164586D+00 0.305981D+01
 71 0.909875D+00 - 229400D+00 - 476904D+00 0.783084D+01 - 109246D+01 0.192517D-01 0.154516D+00 0.305981D+01
 72 0.919044D+00 - 216414D+00 - 540127D+00 0.846722D+01 - 101779D+01 0.172569D-01 0.145081D+00 0.305981D+01
 73 0.923198D+00 - 209954D+00 - 570994D+00 0.877733D+01 - 981984D+00 0.163757D-01 0.141030D+00 0.305981D+01
 74 0.927351D+00 - 203042D+00 - 6033800D+00 0.910947D+01 - 944844D+00 0.155027D-01 0.137128D+00 0.305981D+01
 75 0.935659D+00 - 187722D+00 - 673185D+00 0.984573D+01 - 866117D+00 0.137828D-01 0.129790D+00 0.305981D+01
 76 0.943071D+00 - 172032D+00 - 741741D+00 0.1055886D+02 - 790564D+00 0.122664D-01 0.123802D+00 0.305981D+01
 77 0.950484D+00 - 154074D+00 - 816651D+00 0.112900D+02 - 709606D+00 0.107566D-01 0.118383D+00 0.305981D+01
 78 0.956971D+00 - 136143D+00 - 887743D+00 0.119763D+02 - 634179D+00 0.943293D-02 0.114137D+00 0.305981D+01
 79 0.963458D+00 - 115798D+00 - 964546D+00 0.127391D+02 - 554051D+00 0.8099925D-02 0.110384D+00 0.305981D+01
 80 0.968893D+00 - 961656D+01 - 103498D+01 0.134230D+02 - 481653D+00 0.694838D-02 0.107595D+00 0.305981D+01
 81 0.974528D+00 - 742115D+01 - 111009D+01 0.141114D-02 - 4054456D+00 0.578002D-02 0.105208D+00 0.305981D+01
 82 0.983647D+00 - 321974D+01 - 124454D+01 0.153287D+02 - 271339D+00 0.380537D-02 0.102205D+00 0.305981D+01
 83 0.990780D+00 0.675194D-02 - 135985D+01 0.164026D+02 - 158235D+00 0.219723D-02 0.100716D+00 0.305981D+01
 84 0.995895D+00 0.385894D+01 - 144856D+01 0.172392D+02 - 722201D-01 0.997480D-03 0.100144D+00 0.305981D+01
 85 0.998270D+01 0.150451D+01 0.177713D+02 - 183446D-01 0.252743D-03 0.100009D+00 0.305981D+01
 86 0.100000D+01 0.668333D+01 - 152363D+01 0.179538D+02 0.000000D+00 0.165017D-19 0.100000D+00 0.305981D+01

9 = NUMBER OF RISER SEGMENTS

0.883920D+02 = UNSTRETCHED RISER LENGTH IN M

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

DATA PER RISER SEGMENT FOR NSEG = 9 SEGMENTS
 DIMENSIONAL QUANTITIES IN THE SYSTEM
 RLNG DXI PXIETA AO WEIGHT MASS TMASS AMAETA AMAZI TMAXI
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.4993D+02
 0.2500D+01 0.3100D+00 0.9300D+00 0.2374D-01 0.4993D+02 0.4047D+02 0.5032D+02 0.0000D+00 0.1324D+03 0.4993D+02

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM STATIC TENSION IN N

0.000000D+00 = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S

0.000000D+00 = X COORDINATE AT TOP IN M

0.701037D+02 = Y COORDINATE AT TOP IN M

0.883920D+02 = NUMBER OF RISER SEGMENTS

0.292000D+01 = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH IN N/M

0.820000D+03 = INNER FLUID DENSITY IN KG/M³

0.102500D+04 = SALT WATER DENSITY IN KG/M³

0.115434D-01 = INNER CROSS SECTIONAL AREA IN M²

0.000000D+00 = INNER FLUID SPEED IN M/S

0.345000D+07 = INNER FLUID OVERPRESSURE IN N/M²

0.258105D+03 = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N

0.143164D+03 = MAXIMUM

0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 6839D+02 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2500D+01 0 . 3100D+00 0 . 9300D+00 0 . 2374D-01 0 . 2920D+01 0 . 4993D+02 0 . 4047D+02 0 . 8244D+02 0 . 5032D+02 0 . 0000D+00 0 . 1324D+03 0 . 4993D+02
 0 . 2670D+09 0 . 6600D+04 - . 3300D+03 0 . 2440D+05 - . 1220D+04 0 . 1164D+07 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 5775D+04 - . 3300D+03 0 . 2135D+05 - . 1220D+04 0 . 1019D+07 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4950D+04 - . 3300D+03 0 . 1830D+05 - . 1220D+04 0 . 8730D+06 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4125D+04 - . 3300D+03 0 . 1525D+05 - . 1220D+04 0 . 7275D+06 - . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 3300D+04 0 . 6000D+00 0 . 1220D+05 0 . 5820D+06 0 . 0000D+00 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4125D+04 0 . 3300D+03 0 . 1525D+05 0 . 1220D+04 0 . 7275D+06 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 4950D+04 0 . 3300D+03 0 . 1830D+05 0 . 1220D+04 0 . 8730D+06 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 5775D+04 0 . 3300D+03 0 . 2135D+05 0 . 1220D+04 0 . 1019D+07 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00
 0 . 2670D+09 0 . 6600D+04 0 . 3300D+03 0 . 2440D+05 0 . 1220D+04 0 . 1164D+07 0 . 5820D+05 0 . 2000D+00 0 . 4932D+00 0 . 7806D-01 0 . 5713D+00

 M O D E N U M B E R = 1
 N A T U R A L F R E Q U E N C Y = 0 . 3947D-01 RAD/S

ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = 86 POINTS

S	TENSION	QX1	OMEGAETA	PHI	P	Q	SIGMA
0 . 0000D+00	- . 2805D+01 0 . 1344D+01	- . 3675D+02 0 . 0000D+00	0 . 0000D+00 0 . 0000D+00	0 . 0000D+00 0 . 0000D+00	0 . 3355D+01		
0 . 1027D-02	- . 2824D+01 0 . 1287D+01	- . 3714D+02 - . 3795D-01	. 5847D-07 - . 1946D-04	. 0 . 3355D+01			
0 . 4105D-02	- . 2878D+01 0 . 1114D+01	- . 3828D+02 - . 1540D+00	. 3928D-05 - . 3139D-03	. 0 . 3355D+01			
0 . 9220D-02	- . 2953D+01 0 . 8154D+00	- . 4001D+02 - . 3544D+00	. 4642D-04 - . 1603D-02	. 0 . 3355D+01			
0 . 1635D-01	- . 3024D+01 0 . 3789D+00	- . 4202D+02 - . 6472D+00	. 2775D-03 - . 5162D-02	. 0 . 3355D+01			
0 . 2547D-01	- . 3049D+01 - . 2008D+00	- . 4390D+02 - . 1040D+01	. 1088D-02 - . 1278D-01	. 0 . 3355D+01			
0 . 3654D-01	- . 2969D+01 - . 9130D+00	- . 4508D+02 - . 1533D+01	. 3395D-02 - . 2676D-01	. 0 . 3355D+01			
0 . 4303D-01	- . 2865D+01 - . 1321D+01	- . 4516D+02 - . 1826D+01	. 5686D-02 - . 3733D-01	. 0 . 3355D+01			
0 . 4952D-01	- . 2718D+01 - . 1714D+01	- . 4473D+02 - . 2118D+01	. 8856D-02 - . 4960D-01	. 0 . 3355D+01			
0 . 5693D-01	- . 2504D+01 - . 2135D+01	- . 4357D+02 - . 2446D+01	. 1372D-01 - . 6557D-01	. 0 . 3355D+01			
0 . 6434D-01	- . 2246D+01 - . 2517D+01	- . 4159D+02 - . 2762D+01	. 2007D-01 - . 8345D-01	. 0 . 3355D+01			
0 . 6849D-01	- . 2085D+01 - . 2710D+01	- . 4014D+02 - . 2932D+01	. 2432D-01 - . 9422D-01	. 0 . 3355D+01			
0 . 7265D-01	- . 1915D+01 - . 2887D+01	- . 3847D+02 - . 3095D+01	. 2909D-01 - . 1055D+00	. 0 . 3355D+01			
0 . 7680D-01	- . 1737D+01 - . 3047D+01	- . 3666D+02 - . 3251D+01	. 3444D-01 - . 1171D+00	. 0 . 3355D+01			

0. 8096D-01	- 1554D+01	- 3189D+01	- 3456D+02	- 3399D+01	0. 4026D-01	- 1292D+00	0. 3355D+01	
0. 8325D-01	- 1451D+01	- 3260D+01	- 3332D+02	- 3477D+01	0. 4372D-01	- 1360D+00	0. 3355D+01	
0. 8554D-01	- 1348D+01	- 3326D+01	- 3197D+02	- 3552D+01	0. 4735D-01	- 1428D+00	0. 3355D+01	
0. 9012D-01	- 1141D+01	- 3439D+01	- 2889D+02	- 3691D+01	0. 5507D-01	- 1568D+00	0. 3355D+01	
0. 9471D-01	- 9371D+00	- 3531D+01	- 2543D+02	- 3816D+01	0. 6340D-01	- 1710D+00	0. 3355D+01	
0. 9700D-01	- 8374D+00	- 3568D+01	- 2362D+02	- 3872D+01	0. 6778D-01	- 1782D+00	0. 3355D+01	
0. 9929D-01	- 7395D+00	- 3601D+01	- 2176D+02	- 3942D+01	0. 7231D-01	- 1854D+00	0. 3355D+01	
0. 1043D+00	- 5368D+00	- 3654D+01	- 1736D+02	- 4022D+01	0. 8248D-01	- 2011D+00	0. 3355D+01	
0. 1068D+00	- 4427D+00	- 3672D+01	- 1492D+02	- 4062D+01	0. 8765D-01	- 2091D+00	0. 3355D+01	
0. 1093D+00	- 3540D+00	- 3684D+01	- 1234D+02	- 4099D+01	0. 9288D-01	- 2171D+00	0. 3355D+01	
0. 1143D+00	- 1939D+00	- 3691D+01	- 6778D+01	- 4145D+01	0. 1034D+00	- 2331D+00	0. 3355D+01	
0. 1193D+00	- 5910D-01	- 3681D+01	- 7356D+00	- 4163D+01	0. 1141D+00	- 2490D+00	0. 3355D+01	
0. 1265D+00	0. 8125D-01	- 3637D+01	0. 8222D+01	- 4137D+01	0. 1290D+00	- 2720D+00	0. 3355D+01	
0. 1301D+00	0. 1252D+00	- 3606D+01	0. 1263D+02	- 4099D+01	0. 1649D+00	- 2834D+00	0. 3355D+01	
0. 1336D+00	0. 1520D+00	- 3570D+01	0. 1700D+02	- 4096D+01	0. 1425D+00	- 2949D+00	0. 3355D+01	
0. 1372D+00	0. 1622D+00	- 3530D+01	0. 2132D+02	- 3977D+01	0. 1487D+00	- 3063D+00	0. 3355D+01	
0. 1408D+00	0. 1562D+00	- 3486D+01	0. 2555D+02	- 3893D+01	0. 1545D+00	- 3176D+00	0. 3355D+01	
0. 1638D+00	- 2327D+00	- 3150D+01	0. 5158D+02	- 2999D+01	0. 1768D+00	- 3875D+00	0. 3355D+01	
0. 1883D+00	- 1151D+01	- 2622D+01	0. 7565D+02	- 1438D+01	0. 1649D+00	- 4474D+00	0. 3355D+01	
0. 2139D+00	- 2330D+01	- 1678D+01	0. 9470D+02	- 0. 7708D+00	0. 1072D+00	- 4735D+00	0. 3355D+01	
0. 2408D+00	- 3257D+01	- 1403D+00	0. 1033D+03	0. 3446D+01	0. 1615D-02	- 4291D+00	0. 3355D+01	
0. 2687D+00	- 3338D+01	0. 1763D+01	0. 9560D+02	0. 6278D+01	0. 9367D-01	- 2795D+00	0. 3355D+01	
0. 2832D+00	- 2971D+01	0. 2631D+01	0. 8479D+02	0. 7586D+01	0. 1293D+00	- 1612D+00	0. 3355D+01	
0. 2976D+00	- 2383D+01	0. 3293D+01	0. 7019D+02	0. 8708D+01	- 1442D+00	- 2074D-01	0. 3355D+01	
0. 3125D+00	- 1654D+01	0. 3695D+01	0. 5247D+02	0. 9622D+01	- 1348D+00	0. 1391D+00	0. 3355D+01	
0. 3273D+00	- 9161D+00	0. 3792D+01	0. 3350D+02	0. 1026D+02	- 9981D-01	0. 3059D+00	0. 3355D+01	
0. 3425D+00	- 2563D+00	0. 3620D+01	0. 3620D+01	0. 1429D+02	0. 1062D+02	- 4087D-01	0. 4761D+00	
0. 3577D+00	0. 2272D+00	0. 3254D+01	0. 3500D+01	0. 1070D+02	0. 3525D-01	0. 6393D+00	0. 3355D+01	
0. 3887D+00	0. 6299D+00	0. 2267D+01	- 3269D+02	0. 1012D+02	0. 2142D+00	0. 9375D+00	0. 3355D+01	
0. 4202D+00	0. 4237D+00	0. 1315D+01	- 5169D+02	0. 8765D+01	0. 3848D+00	0. 1169D+01	0. 3355D+01	
0. 4520D+00	- 7458D-01	0. 5874D+00	- 6175D+02	0. 6940D+01	0. 5113D+00	0. 1396D+01	0. 3355D+01	
0. 4840D+00	- 5967D+00	0. 7583D-01	- 6517D+02	0. 4893D+01	0. 5793D+00	0. 1560D+01	0. 3355D+01	
0. 5160D+00	- 1005D+01	- 2837D+00	- 6391D+02	0. 2813D+01	0. 5905D+00	0. 1679D+01	0. 3355D+01	
0. 5480D+00	- 1259D+01	- 5360D+00	- 5360D+00	0. 8326D+00	0. 5567D+00	0. 1748D+01	0. 3355D+01	
0. 5798D+00	- 1368D+01	- 7018D+00	- 5263D+02	- 9530D+00	0. 4930D+00	0. 1765D+01	0. 3355D+01	
0. 6113D+00	- 1367D+01	- 7922D+00	- 4460D+02	- 2485D+01	0. 4166D+00	0. 1730D+01	0. 3355D+01	
0. 6423D+00	- 1296D+01	- 8195D+00	- 3608D+02	- 3737D+01	0. 3382D+00	0. 1650D+01	0. 3355D+01	
0. 6727D+00	- 1186D+01	- 8007D+00	- 2769D+02	- 4707D+01	0. 2666D+00	0. 1535D+01	0. 3355D+01	
0. 7024D+00	- 1063D+01	- 7554D+00	- 1984D+02	- 5412D+01	0. 2055D+00	0. 1394D+01	0. 3355D+01	
0. 7313D+00	- 9392D+00	- 7028D+00	- 1270D+02	- 5880D+01	0. 1560D+00	0. 1237D+01	0. 3355D+01	
0. 7592D+00	- 8238D+00	- 6602D+00	- 6263D+01	- 6144D+01	0. 1173D+00	0. 1073D+01	0. 3355D+01	
0. 7861D+00	- 7196D+00	- 6438D+00	- 5552D+00	- 6233D+01	0. 8746D-01	0. 9095D+00	0. 3355D+01	
0. 8534D+00	- 4853D+00	- 8688D+00	0. 1588D+02	- 5737D+01	0. 3658D-01	0. 5045D+00	0. 3355D+01	
0. 8592D+00	- 4659D+00	- 9199D+00	0. 1763D+02	- 5641D+01	0. 6469D-01	0. 7519D+00	0. 3355D+01	
0. 8646D+00	- 4472D+00	- 9747D+00	0. 1936D+02	- 5541D+02	0. 3051D-01	0. 4421D+00	0. 3355D+01	
0. 8699D+00	- 5432D+00	- 7548D+00	0. 1116D+02	- 5970D+01	0. 4711D-01	0. 6049D+00	0. 3355D+01	
0. 8477D+00	- 5047D+00	- 8247D+00	0. 1423D+02	- 5824D+01	0. 3994D-01	0. 5375D+00	0. 3355D+01	
0. 8807D+00	- 3860D+00	- 1108D+01	0. 2315D+02	- 5313D+01	0. 2504D-01	0. 3839D+00	0. 3355D+01	
0. 8857D+00	- 3652D+00	- 1188D+01	0. 2527D+02	- 5183D+01	0. 2247D-01	0. 3558D+00	0. 3355D+01	
0. 8907D+00	- 3432D+00	- 1271D+01	0. 2716D+02	- 5051D+01	0. 2020D-01	0. 3304D+00	0. 3355D+01	
0. 8957D+00	- 3201D+00	- 1362D+01	0. 2877D+02	- 4912D+01	0. 1805D-01	0. 3056D+00	0. 3355D+01	
			- 1460D+00	- 3013D+02	- 4764D+01	0. 1603D-01	0. 2816D+00	0. 3355D+01

0. 9007D+00 -.2958D+00 -.1564D+01 0. 3125D+02 -.4611D+01 0. 1414D-01 0. 2582D+00 0. 3355D+01
 0. 9099D+00 -.2478D+00 -.1770D+00 -.3314D+01 0. 3527D+02 -.4316D+01 0. 1102D-01 0. 2175D+00 0. 3355D+01
 0. 9190D+00 -.1951D+00 -.1997D+01 0. 3527D+02 -.4002D+01 0. 8373D-02 0. 1795D+00 0. 3355D+01
 0. 9232D+00 -.1692D+00 -.2107D+01 0. 3632D+02 -.3854D+01 0. 7318D-02 0. 1632D+00 0. 3355D+01
 0. 9274D+00 -.1417D+00 -.2223D+01 0. 3745D+02 -.3701D+01 0. 6344D-02 0. 1475D+00 0. 3355D+01
 0. 9357D+00 -.8122D-01 -.2472D+01 0. 3997D+02 -.3379D+01 0. 4634D-02 0. 1182D+00 0. 3355D+01
 0. 9431D+00 -.1997D-01 -.2715D+01 0. 4241D+02 -.3074D+01 0. 3366D-02 0. 9430D-01 0. 3355D+01
 0. 9505D+00 0. 4949D-01 -.2981D+01 0. 4495D+02 -.2750D+01 0. 2328D-02 0. 7274D-01 0. 3355D+01
 0. 9570D+00 0. 1183D+00 -.3233D+01 0. 4733D+02 -.2451D+01 0. 1598D-02 0. 5588D-01 0. 3355D+01
 0. 9635D+00 0. 1959D+00 -.3505D+01 0. 4998D+02 -.2136D+01 0. 1025D-02 0. 4101D-01 0. 3355D+01
 0. 9690D+00 0. 2701D+00 -.3754D+01 0. 5236D+02 -.1852D+01 0. 6516D-03 0. 2998D-01 0. 3355D+01
 0. 9745D+00 0. 3529D+00 -.4019D+01 0. 5477D+02 -.1556D+01 0. 3760D-03 0. 2054D-01 0. 3355D+01
 0. 9836D+00 0. 5102D+00 -.4494D+01 0. 5902D+02 -.1038D+01 0. 1066D-03 0. 8689D-02 0. 3355D+01
 0. 9908D+00 0. 6552D+00 -.4901D+01 0. 6278D+02 -.6033D+00 0. 2020D-04 0. 2820D-02 0. 3355D+01
 0. 9959D+00 0. 7733D+00 -.5214D+01 0. 6571D+02 -.2748D+00 0. 1844D-05 0. 5677D-03 0. 3355D+01
 0. 9990D+00 0. 8507D+00 -.5412D+01 0. 6757D+02 -.6973D-01 0. 2862D-07 0. 3586D-04 0. 3355D+01
 0. 10000D+01 0. 8777D+00 -.5479D+01 0. 6821D+02 0. 0000D+00 -.1245D-16 -.2552D-16 0. 3355D+01
 MAXIMUM ESTIMATED ERROR BY COMPONENTS
 0. 7407D-02 0. 2005D-01 0. 2382D+00 0. 1898D-01 0. 1460D-02 0. 4141D-02 0. 2697D-02

IX.2.1.3 RC2DZ1A DATA A

1 86 0.3947D-01 0.0000D+00 0.7931D+00
 0.000000D+00 - 280488D+01 0.134392D+01 - .367462D+02 0.000000D+00 0.000000D+00 0.000000D+00 0.335458D+01 0.000000D+00
 0.102730D-02 - .282420D+01 0.128741D+01 - .371430D+02 - .379533D-01 0.584282D-07 - .194577D-04 0.335458D+01 0.000000D+00
 0.410499D-02 - .287815D+01 0.111436D+01 - .382827D+02 - .154042D+00 0.392809D-05 - .313918D-03 0.335458D+01 0.000000D+00
 0.922042D-02 - .295348D+01 0.815355D+00 - .400056D+02 - .354378D+00 0.464196D-04 - .160936D-02 0.335458D+01 0.000000D+00
 0.163526D-01 - .302438D+01 0.378917D+00 - .420237D+02 - .647185D+00 0.272463D-03 - .516225D-02 0.335458D+01 0.000000D+00
 0.254721D-01 - .304900D+01 - .200839D+00 - .438958D+02 - .103958D+01 0.108839D-02 - .127814D-01 0.335458D+01 0.000000D+00
 0.365416D-01 - .294906D+01 - .910345D+00 - .450804D+02 - .153324D+01 0.339501D-02 - .267567D-01 0.335458D+01 0.000000D+00
 0.430286D-01 - .286462D+01 - .132119D+01 - .451579D+02 - .182620D+01 0.568644D-02 - .373309D-01 0.335458D+01 0.000000D+00
 0.495156D-01 - .271848D+01 - .171407D+01 - .447262D+02 - .211803D+01 0.885587D-02 - .495590D-01 0.335458D+01 0.000000D+00
 0.569281D-01 - .250426D+01 - .213501D+01 - .435691D+02 - .244577D+01 0.137155D-01 - .655734D-01 0.335458D+01 0.000000D+00
 0.643406D-01 - .224558D+01 - .251667D+01 - .415914D+02 - .276191D+01 0.200653D-01 - .834513D-01 0.335458D+01 0.000000D+00
 0.684944D-01 - .208458D+01 - .271007D+01 - .401370D+02 - .293173D+01 0.243217D-01 - .942167D-01 0.335458D+01 0.000000D+00
 0.726482D-01 - .191466D+01 - .288692D+01 - .384714D+02 - .309507D+01 0.290931D-01 - .105461D+00 0.335458D+01 0.000000D+00
 0.768021D-01 - .173730D+01 - .304676D+01 - .366041D+02 - .325107D+01 0.343993D-01 - .117133D+00 0.335458D+01 0.000000D+00
 0.809559D-01 - .155393D+01 - .318933D+01 - .345591D+02 - .339895D+01 0.402590D-01 - .129173D+00 0.335458D+01 0.000000D+00
 0.832481D-01 - .145108D+01 - .326035D+01 - .333219D+02 - .347677D+01 0.437248D-01 - .139556D+00 0.335458D+01 0.000000D+00
 0.855402D-01 - .134775D+01 - .332571D+01 - .319724D+02 - .355162D+01 0.473485D-01 - .142828D+00 0.335458D+01 0.000000D+00
 0.901246D-01 - .114090D+01 - .343936D+01 - .288889D+02 - .369131D+01 0.550732D-01 - .156798D+00 0.335458D+01 0.000000D+00
 0.947089D-01 - .937067D+00 - .353070D+01 - .254259D+02 - .381592D+01 0.634047D-01 - .171003D+00 0.335458D+01 0.000000D+00
 0.970010D-01 - .837400D+00 - .356839D+01 - .362317D+02 - .387214D+01 0.677847D-01 - .178169D+00 0.335458D+01 0.000000D+00
 0.992932D-01 - .739531D+00 - .365420D+01 - .217581D+02 - .3402212D+01 0.723068D-01 - .185363D+00 0.335458D+01 0.000000D+00
 0.104289D+00 - .536768D+00 - .365420D+01 - .1736112D+02 - .402212D+01 0.824798D-01 - .201150D+00 0.335458D+01 0.000000D+00
 0.106787D+00 - .442725D+00 - .367117D+01 - .149237D+02 - .406247D+01 0.876549D-01 - .209103D+00 0.335458D+01 0.000000D+00
 0.109285D+00 - .3539994D+00 - .368354D+01 - .123422D+02 - .409656D+01 0.928815D-01 - .217081D+00 0.335458D+01 0.000000D+00
 0.114281D+00 - .193920D+00 - .369149D+01 - .677774D+01 - .414454D+01 0.103446D+00 - .233073D+00 0.335458D+01 0.000000D+00
 0.119277D+00 - .591018D-01 - .368073D+01 - .735620D+00 - .416346D+01 0.114097D+00 - .249034D+00 0.335458D+01 0.000000D+00
 0.126460D+00 0.812466D-01 - .363720D+01 0.8221172D+01 - .413663D+01 0.128957D+00 - .271951D+00 0.335458D+01 0.000000D+00
 0.130052D+00 0.125207D+00 - .360564D+01 0.126315D+02 - .409917D+01 0.135931D+00 - .283420D+00 0.335458D+01 0.000000D+00
 0.133643D+00 0.152041D+00 - .356951D+01 0.169988D+02 - .404595D+01 0.142539D+00 - .294867D+00 0.335458D+01 0.000000D+00
 0.137234D+00 0.162191D+00 - .352957D+01 0.213197D+02 - .397171D+01 0.148731D+00 - .306277D+00 0.335458D+01 0.000000D+00
 0.140825D+00 0.1561159D+00 - .348645D+01 0.255900D+02 - .389290D+01 0.154455D+00 - .317630D+00 0.335458D+01 0.000000D+00
 0.163850D+00 - .232668D+00 - .314955D+01 0.515801D+02 - .299904D+01 0.176792D+00 - .387467D+00 0.335458D+01 0.000000D+00
 0.188255D+00 - .115107D+00 - .262194D+01 0.756509D+02 - .143626D+01 0.164920D+00 - .447380D+00 0.335458D+01 0.000000D+00
 0.213942D+00 - .232968D+01 0.329310D+01 0.701922D+02 - .870842D+01 - .144183D+00 - .207449D-01 0.335458D+01 0.000000D+00
 0.312462D+00 - .165400D+01 0.369540D+01 0.524740D+02 0.962214D+01 - .134776D+00 0.139067D+00 0.335458D+01 0.000000D+00
 0.327317D+00 - .916505D+00 0.379230D+01 0.103289D+03 0.346079D+01 0.941529D-02 - .429102D+00 0.335458D+01 0.000000D+00
 0.342527D+00 - .256349D+00 0.362038D+01 0.142878D+02 0.106216D+02 - .998120D-01 0.305900D+00 0.335458D+01 0.000000D+00
 0.357736D+00 0.227113D+00 0.325439D+01 0.847855D+02 0.758554D+01 - .408735D-01 0.476071D+00 0.335458D+01 0.000000D+00
 0.388740D+00 0.629902D+00 0.226715D+01 - .3226857D+02 0.101182D+02 0.3524553D-01 - .129263D+00 0.161154D+00 0.335458D+01 0.000000D+00
 0.548012D+00 - .125858D+01 0.536021D+00 - .593763D+02 0.832585D+00 0.590539D+00 0.167924D+01 0.335458D+01 0.000000D+00
 0.423677D+00 0.423677D+00 0.131517D+01 - .516937D+02 0.876499D+01 0.556668D+00 0.174843D+01 0.335458D+01 0.000000D+00
 0.451988D+00 - .7457765D-01 0.587430D+00 - .617493D+02 0.693978D+01 0.511324D+00 0.188797D+00 0.335458D+01 0.000000D+00
 0.483974D+00 - .596727D+00 0.758282D-01 - .6511706D+02 0.106242D+02 - .3607772D+02 0.445974D+02 0.248542D+01 0.335458D+01 0.000000D+00
 0.516026D+00 - .100525D+01 0.819463D+00 - .639147D+02 0.489331D+01 0.170741D+02 0.3500453D+01 0.579299D+00 0.155998D+01 0.335458D+01 0.000000D+00
 0.642264D+00 - .129580D+01 0.873334D+00 - .373711D+01 0.3382339D+00 0.165030D+01 0.173002D+01 0.335458D+01 0.000000D+00
 0.676833D+00 - .1186334D+01 0.800742D+00 - .2768896D+02 0.470669D+01 0.266553D+00 0.153477D+01 0.165503D+01 0.335458D+01 0.000000D+00
 0.702392D+00 - .106263D+01 - .755443D+00 - .198370D+02 0.541165D+01 0.205496D+00 0.139365D+01 0.173040D+01 0.335458D+01 0.000000D+00
 0.731269D+00 - .939236D+00 - .702763D+00 - .126999D+02 0.588025D+01 0.156026D+00 0.123684D+01 0.171762D+00 0.335458D+01 0.000000D+00
 0.759196D+00 - .823842D+00 - .660195D+00 - .626321D+01 - .614416D+01 0.107304D+01 0.17262D+00 0.135458D+01 0.000000D+00

0.786058D+00 - 7.19571D+00 - 643752D+00 - 355202D+00 - 6232B1D+01 0.874587D-01 0.909471D+00 0.335458D+01 0.000000D+00
 0.811745D+00 - 6.26689D+00 - 669387D+00 0.532086D+01 - 616957D+01 0.646932D-01 0.751910D+00 0.335458D+01 0.000000D+00
 0.823947D+00 - 584529D+00 - 702881D+00 0.815252D+01 - 608753D+01 0.554563D-01 0.677865D+00 0.335458D+01 0.000000D+00
 0.836150D+00 - 543242D+00 - 754769D+00 0.111594D+02 - 596993D+01 0.471080D-01 0.604923D+00 0.335458D+01 0.000000D+00
 0.847662D+00 - 504675D+00 - 824691D+00 0.142331D+02 - 582405D+01 0.399426D-01 0.537545D+00 0.335458D+01 0.000000D+00
 0.853419D+00 - 485335D+00 - 868801D+00 0.158843D+02 - 573740D+01 0.365836D-01 0.504507D+00 0.335458D+01 0.000000D+00
 0.859175D+00 - 465876D+00 - 919880D+00 0.176279D+02 - 564101D+01 0.338595D-01 0.471983D+00 0.335458D+01 0.000000D+00
 0.864562D+00 - 447219D+00 - 974715D+00 0.193564D+02 - 554144D+01 0.305112D-01 0.442060D+00 0.335458D+01 0.000000D+00
 0.869949D+00 - 427782D+00 - 103709D+01 0.211919D+02 - 543228D+01 0.277265D-01 0.412687D+00 0.335458D+01 0.000000D+00
 0.875336D+00 - 407446D+00 - 110779D+01 0.231487D+02 - 531291D+01 0.250436D-01 0.383917D+00 0.335458D+01 0.000000D+00
 0.880723D+00 - 386045D+00 - 118779D+01 0.252655D+02 - 518252D+01 0.224699D-01 0.355808D+00 0.335458D+01 0.000000D+00
 0.885719D+00 - 365160D+00 - 127078D+01 0.271589D+02 - 505149D+01 0.201979D-01 0.330380D+00 0.335458D+01 0.000000D+00
 0.890715D+00 - 343214D+00 - 136174D+01 0.287720D+02 - 491166D+01 0.180510D-01 0.305618D+00 0.335458D+01 0.000000D+00
 0.895711D+00 - 320111D+00 - 145979D+01 0.301338D+02 - 476440D+01 0.160301D-01 0.281561D+00 0.335458D+01 0.000000D+00
 0.9000707D+00 - 295766D+00 - 156408D+01 0.312540D+02 - 461098D+01 0.141363D-01 0.258244D+00 0.335458D+01 0.000000D+00
 0.9098875D+00 - 247778D+00 - 177029D+01 0.331442D+02 - 431577D+01 0.110159D-01 0.217473D+00 0.335458D+01 0.000000D+00
 0.919044D+00 - 195147D+00 - 199707D+01 0.352691D+02 - 400231D+01 0.837321D-02 0.179452D+00 0.335458D+01 0.000000D+00
 0.923198D+00 - 169245D+00 - 210744D+01 0.363184D+02 - 385365D+01 0.731823D-02 0.163182D+00 0.335458D+01 0.000000D+00
 0.927351D+00 - 141710D+00 - 222301D+01 0.374475D+02 - 370050D+01 0.634361D-02 0.147536D+00 0.335458D+01 0.000000D+00
 0.935659D+00 - 812238D-01 - 247165D+01 0.399668D+02 - 337914D+01 0.463410D-02 0.118182D+00 0.335458D+01 0.000000D+00
 0.943071D+00 - 199710D-01 - 271519D+01 0.424144D+02 - 307393D+01 0.336633D-02 0.943027D-01 0.335458D+01 0.000000D+00
 0.950484D+00 0.494942D-01 - 298089D+01 0.449463D+02 - 275023D+01 0.232763D-02 0.727380D-01 0.335458D+01 0.000000D+00
 0.956971D+00 0.118276D+00 - 323280D+01 0.473311D+02 - 245105D+01 0.159785D-02 0.588806D-01 0.335458D+01 0.000000D+00
 0.963458D+00 0.195871D+00 - 350467D+01 0.499798D+02 - 213554D+01 0.102467D-02 0.410095D-01 0.335458D+01 0.000000D+00
 0.9689993D+00 0.270143D+00 - 375376D+01 0.523603D+02 - 185235D+01 0.651587D-03 0.299752D-01 0.335458D+01 0.000000D+00
 0.974528D+00 0.352854D+00 - 401926D+01 0.547653D+02 - 155592D+01 0.376034D-03 0.205416D-01 0.335458D+01 0.000000D+00
 0.983647D+00 0.510206D+00 - 449423D+01 0.590215D+02 - 1037500D+01 0.106649D-03 0.868877D-02 0.335458D+01 0.000000D+00
 0.990780D+00 0.656207D+00 - 490133D+01 0.627783D+02 - 603342D+00 0.201973D-04 0.282041D-02 0.335458D+01 0.000000D+00
 0.995895D+00 0.773256D+00 - 521437D+01 0.657095D+02 - 274829D+00 0.184433D-05 0.567715D-03 0.335458D+01 0.000000D+00
 0.998973D+00 0.850695D+00 - 541178D+01 0.675743D+02 - 697264D-01 0.286221D-07 0.358650D-04 0.335458D+01 0.000000D+00
 0.100000D+01 0.877686D+00 - 547921D+01 0.682139D+02 0.000000D+00 - 124485D-16 0.255245D-16 0.335558D+01 0.000000D+00

Chapter X

INPUT-OUTPUT MANUALS FOR NAG SUBROUTINES
D02RAF, D02GBF, C05NBF, C05PBF, F04ATF, F03AAF

D02 - Ordinary Differential Equations

D02RAF - NAG FORTRAN Library Routine Document

NOTE: before using this routine, please read the appropriate implementation document to check the interpretation of *bold italicised* terms and other implementation-dependent details. The routine name may be precision-dependent.

1. Purpose

D02RAF solves the two-point boundary-value problem with general boundary conditions for a system of ordinary differential equations, using a deferred correction technique and Newton iteration.

2. Specification

```
SUBROUTINE D02RAF (N, MNP, NP, NUMBEG, NUMMIX, TOL, INIT, X,
1   Y, IY, ABT, FCN, G, IJAC, JACOBF, JACOBG, DELEPS,
2   JACEPS, JACGEPS, WORK, LWORK, IWORK, LIWORK, IFAIL)
C   INTEGER N, MNP, NP, NUMBEG, NUMMIX, INIT, IY, IJAC, LWORK,
C   1   IWORK(LIWORK), LIWORK, IFAIL
C   real TOL, X(MNP), Y(IY,MNP), ABT(N), DELEPS, WORK(LWORK)
C   EXTERNAL FCN, G, JACOBF, JACOBG, JACEPS, JACGEPS
```

3. Description

D02RAF solves a two-point boundary-value problem for a system of N ordinary differential equations in a range (A, B) with $B > A$. The system is written in the form

$$y'_i = f_i(x, y_1, y_2, \dots, y_N), \quad i = 1, 2, \dots, N \quad (1)$$

and the derivatives f_i are evaluated by a subroutine FCN supplied by the user. With the differential equations (1) must be given a system of N (nonlinear) boundary conditions

$$g_i(y(A), y(B)) = 0, \quad i = 1, 2, \dots, N$$

where

$$y(x) = [y_1(x), y_2(x), \dots, y_N(x)]^T. \quad (2)$$

The functions g_i are evaluated by a subroutine G supplied by the user. The solution is computed using a finite-difference technique with deferred correction allied to a Newton iteration to solve the finite-difference equations. The technique used is described fully in [1].

The user must supply an absolute error tolerance and may also supply an initial mesh for the finite-difference equations and an initial approximate solution (alternatively a default mesh and approximation are used). The approximate solution is corrected using Newton iteration and deferred correction. Then, additional points are added to the mesh and the solution is recomputed with the aim of making the error everywhere less than the user's tolerance and of approximately equidistributing the error on the final mesh. The solution is returned on this final mesh.

If the solution is required at a few specific points

then these should be included in the initial mesh. If, on the other hand, the solution is required at several specific points then the user should use the interpolation routines provided in the E01 chapter if these points do not themselves form a convenient mesh.

The Newton iteration requires Jacobian matrices

$$\left(\frac{\partial f_i}{\partial y_j} \right), \left(\frac{\partial g_i}{\partial y_j}(A) \right) \text{ and } \left(\frac{\partial g_i}{\partial y_j}(B) \right).$$

These may be supplied by the user through subroutines JACOBF for $\left(\frac{\partial f_i}{\partial y_j} \right)$ and JACOBG

for the others. Alternatively the Jacobians may be calculated by numerical differentiation using the algorithm described in [2].

For problems of the type (1) and (2) for which it is difficult to determine an initial approximation from which the Newton iteration will converge, a continuation facility is provided. The user must set up a family of problems

$$y' = f(x, y, \epsilon), \quad g(y(A), y(B), \epsilon) = 0 \quad (3)$$

where $f = [f_1, f_2, \dots, f_N]^T$ etc., and where ϵ is a continuation parameter. The choice $\epsilon = 0$ must give a problem (3) which is easy to solve and $\epsilon = 1$ must define the problem whose solution is actually required. The routine solves a sequence of problems with ϵ values

$$0 = \epsilon_1 < \epsilon_2 < \dots < \epsilon_p = 1 \quad (4)$$

The number p and the values ϵ_i are chosen by the routine so that each problem can be solved using the solution of its predecessor as a starting

approximation. Jacobians $\frac{\partial f}{\partial e}$ and $\frac{\partial g}{\partial e}$ are required and they may be supplied by the user via routines JACEPS and JACGEP respectively or may be computed by numerical differentiation.

4. References

- [1] PEREYRA, V.
PASVA3: An Adaptive Finite-Difference Fortran Program for First Order Nonlinear Ordinary Boundary Problems. In Childs, B., Scott, M., Daniel, J.W., Denman, E., and Nelson, P. (eds.) 'Codes for Boundary Value Problems in Ordinary Differential Equations', Lecture Notes in Computer Science, 76, 1979.
- [2] CURTIS, A.R., POWELL, M.J.D. and REID, J.K.
On the Estimation of Sparse Jacobian Matrices.
J. Inst. Maths. Applies., 13, pp. 117-119, 1974.

5. Parameters

N - INTEGER.

On entry, N must specify the number of differential equations.

Unchanged on exit.

MNP - INTEGER.

On entry, MNP must be set to the maximum permitted number of points in the finite-difference mesh, $MNP \geq 32$. If LWORK or LIWORK (see below) is too small then internally MNP will be replaced by the maximum permitted by these values. (A warning message will be output if on entry IFAIL is set to obtain monitoring information.)

Unchanged on exit.

NP - INTEGER.

Before entry, NP must be set to the number of points to be used in the initial mesh $4 \leq NP \leq MNP$.

On exit, NP contains the number of points in the final mesh.

NUMBEG - INTEGER.

On entry, NUMBEG must specify the number of left hand boundary conditions (that is the number involving $y(A)$ only).

$0 \leq NUMBEG < N$.

Unchanged on exit.

NUMMIX - INTEGER.

On entry, NUMMIX must specify the number of occupied boundary conditions (that is the number involving both $y(A)$ and $y(B)$).

$0 \leq NUMMIX$;
 $NUMBEG + NUMMIX \leq N$.

Unchanged on exit.

TOL - real.

On entry, TOL must specify a positive absolute error tolerance. If

$$A = x_1 < x_2 < \dots < x_{NP} = B$$

is the final mesh, $z_j(x_i)$ is the j(th) component of the approximate solution at x_i , and $y_j(x)$ is the j(th) component of the true solution of (1) and (2), then, except in extreme circumstances, it is expected that

$$|z_j(x_i) - y_j(x_i)| \leq TOL, \quad i = 1, 2, \dots, NP, \quad j = 1, 2, \dots, N. \quad (5)$$

Unchanged on exit.

INIT - INTEGER.

On entry, INIT must specify whether the user wishes to supply an initial mesh and approximate solution ($INIT \neq 0$) or whether default values are to be used, ($INIT = 0$).

Unchanged on exit.

X - real array of DIMENSION (MNP).

Before entry, the user must set $X(1) = A$ and $X(NP) = B$. If $INIT = 0$ on entry a default equispaced mesh will be used, otherwise the user must specify a mesh satisfying

$$A = X(1) < X(2) < \dots < X(NP) = B \quad (6)$$

On exit, $X(1), X(2), \dots, X(NP)$ define the final mesh (with the returned value of NP) satisfying (6).

Y - real array of DIMENSION (IY,MNP).

If $INIT = 0$ on entry, then Y need not be set.

If $INIT \neq 0$ before entry, then the array Y must contain an initial approximation to the solution such that $Y(J, I)$ contains an approximation to

$$y_j(x_i), \quad I = 1, 2, \dots, NP, \quad J = 1, 2, \dots, N.$$

On a successful exit Y contains the approximate solution $z_j(x_i)$ satisfying (5) on the final mesh, that is

$$Y(J, I) = z_j(x_i), \quad I = 1, 2, \dots, NP, \quad J = 1, 2, \dots, N,$$

where NP is the number of points in the final

D02 - Ordinary Differential Equations.

mesh. If an error has occurred then Y contains the latest approximation to the solution. The remaining columns of Y are not used.

IY - INTEGER.

On entry, IY must specify the first dimension of Y as declared in the calling (sub)program.
 $IY \geq N$.

Unchanged on exit.

ABT - *real* array of DIMENSION (N).

On successful exit, ABT(i), $i = 1, 2, \dots, N$, holds the largest estimated error (in magnitude) of the i (th) component of the the solution over all mesh points.

FCN - SUBROUTINE, supplied by the user.

FCN must evaluate the functions f_i (i.e., the derivatives y'_i) at the general point X for a given value of EPS (see equation 3).

Its specification is:

```
SUBROUTINE FCN(X,EPS,Y,F,N)
INTEGER N
real X,EPS,Y(N),F(N)
```

X - *real*.

On entry, X specifies the value of the argument X. Its value must not be changed.

EPS - *real*.

On entry, EPS specifies the value of the continuation parameter, ϵ . This is 1 if continuation is not being used. Its value must not be changed.

Y - *real* array of DIMENSION (N).

On entry, Y(i) contains the value of the argument y_i for $i = 1, 2, \dots, N$. These values must not be changed.

F - *real* array of DIMENSION (N).

On exit, F(i) must contain the values of f_i , $i = 1, 2, \dots, N$.

N - INTEGER.

On entry, N specifies the number of equations. Its value must not be changed.

→ FCN must be declared as EXTERNAL in the (sub)program from which D02RAF is called.

G - SUBROUTINE, supplied by the user.

G must evaluate the boundary conditions in equation (3) and place them in BC(i), $i = 1, 2, \dots, N$.

Its specification is:

```
SUBROUTINE G(EPS,YA,YB,BC,N)
INTEGER N
real EPS,YA(N),YB(N),BC(N)
```

EPS - *real*.

On entry, EPS specifies the value of the continuation parameter, ϵ . This is 1 if continuation is not being used. Its value must not be changed.

YA - *real* array of DIMENSION (N).

On entry, YA(i) contains the value $y_i(A)$, $i = 1, 2, \dots, N$. These values must not be changed.

YB - *real* array of DIMENSION (N).

On entry, YB(i) contains the value $y_i(B)$, $i = 1, 2, \dots, N$. These values must not be changed.

BC - *real* array of DIMENSION (N).

On exit, BC(i) must contain the value $g_i(y(A),y(B),\epsilon)$, $i = 1, 2, \dots, N$. These must be ordered as follows:

- (i) First, the NUMBEG conditions involving only $y(A)$ (that is YA).
- (ii) Next, the NUMMIX coupled conditions involving both $y(A)$ and $y(B)$ (that is YA and YB).
- (iii) Finally, the $N - NUMBEG - NUMMIX$ conditions involving $y(B)$ (that is YB).

N - INTEGER.

On entry, N specifies the number of equations. Its value must not be changed.

G must be declared as EXTERNAL in the (sub)program from which D02RAF is called.

IJAC - INTEGER.

If, on entry, IJAC = 0 then the Jacobian matrices for the Newton iterations are calculated by numerical differentiation, and the parameters JACOBF, JACOBG, JACEPS and JACGEP may be replaced by dummy actual parameters in the call to D02RAF. (The NAG routines D02GAZ, D02GAY, D02GAZ and D02GAX respectively may be used as the dummy parameters.) If IJAC ≠ 0 then the user must supply routines JACOBF and JACOBG and also when continuation is used, routines JACEPS and JACGEP.

Unchanged on exit.

D02RAF

D02 - Ordinary Differential Equations

JACOBF - SUBROUTINE, supplied by the user.

JACOBF must evaluate the Jacobian $\left[\frac{\partial f_i}{\partial y_j} \right]$ at the point (x,y) and place $\frac{\partial f_i}{\partial y_j}$ in $F(i,j)$.

$i,j = 1,2,\dots,N$.

Its specification is:

```
SUBROUTINE JACOBF(X,EPS,Y,F,N)
INTEGER N
real X,EPS,Y(N),F(N,N)
```

X - real.

On entry, X specifies the value of the argument X. Its value must not be changed.

EPS - real.

On entry, EPS specifies the value of the continuation parameter ϵ . This is 1 if continuation is not being used. Its value must not be changed.

Y - real array of DIMENSION (N).

On entry, Y(i) contains the value of the argument y_i , $i = 1,2,\dots,N$. These values must not be changed.

F - real array of DIMENSION (N,N).

On exit, F(i,j) contains the value of $\frac{\partial f_i}{\partial y_j}$, evaluated at the point (x,y) , for $i,j = 1,2,\dots,N$.

N - INTEGER.

On entry, N specifies the number of equations. Its value must not be changed.

JACOBF must be declared as EXTERNAL in the (sub)program from which D02RAF is called.

JACOBG - SUBROUTINE, supplied by the user.

JACOBG must evaluate the Jacobians $\left(\frac{\partial g_i}{\partial y_j}(A) \right)$ and $\left(\frac{\partial g_i}{\partial y_j}(B) \right)$ and place them in AJ and BJ respectively. The ordering of the rows of AJ and BJ must correspond to the ordering of the boundary conditions described in the specification of subroutine G above.

The specification of JACOBG is:

```
SUBROUTINE JACOBG(EPS,YA,YB,
1 AJ,BJ,N)
INTEGER N
real EPS,YA(N),YB(N),
1 AJ(N,N),BJ(N,N)
```

EPS - real.

On entry, EPS specifies the value of the continuation parameter, ϵ . This is 1 if continuation is not being used. Its value must not be changed.

YA - real array of DIMENSION (N).

On entry, YA(i) contains $y_i(A)$, $i = 1,2,\dots,N$. These values must not be changed.

YB - real array of DIMENSION (N).

On entry, YB(i) contains the value of $y_i(B)$, $i = 1,2,\dots,N$. These values must not be changed.

AJ - real array of DIMENSION (N,N).

On exit, AJ(i,j) contains $\frac{\partial g_i}{\partial y_j}(A)$,

 $i,j = 1,2,\dots,N$.

BJ - real array of DIMENSION (N,N).

On exit, BJ(i,j) contains $\frac{\partial g_i}{\partial y_j}(B)$,

 $i,j = 1,2,\dots,N$.

N - INTEGER.

On entry, N specifies the number of equations. Its value must not be changed.

JACOBG must be declared as EXTERNAL in the (sub)program from which D02RAF is called.

DELEPS - real.

Before entry, DELEPS must be given a value which specifies whether continuation is required. If $DELEPS \leq 0.0$ or $DELEPS \geq 1.0$ then it is assumed that continuation is not required. If $0.0 < DELEPS < 1.0$ then it is assumed that continuation is required unless $DELEPS < macheps^{\frac{1}{2}}$ (where macheps is the smallest number such that $1.0 + macheps > 1.0$) when an error exit is taken. DELEPS is used as the increment $\epsilon_2 - \epsilon_1$ (see (4)) and the choice $DELEPS = 0.1$ is recommended.

On exit, DELEPS contains an overestimate of the increment $\epsilon_p - \epsilon_{p-1}$ (in fact the value of

D02 - Ordinary Differential Equations

the increment which would have been tried if the restriction $\epsilon_p = 1$ had not been imposed). If continuation was not requested then DELEPS = 0.0 on exit.

If continuation is not requested then the parameters JACEPS and JACGEP may be replaced by dummy actual parameters in the call to D02RAF. (The NAG routines D02GAZ and D02GAX respectively may be used as the dummy parameters.)

JACEPS - SUBROUTINE, supplied by the user.

JACEPS must evaluate the derivative $\frac{\partial f_i}{\partial \epsilon}$ at the point X if continuation is being used. Otherwise the user is advised to use D02GAZ as the actual parameter JACEPS.

The specification of JACEPS is:

```
SUBROUTINE JACEPS(X,EPS,Y,F,N)
  INTEGER N
  real X,EPS,Y(N),F(N)
```

X - real.

On entry, X specifies the value of the argument X. Its value must not be changed.

EPS - real.

On entry, EPS must specify the value of the continuation parameter, ϵ . Its value must not be changed.

Y - real array of DIMENSION (N).

On entry, Y(i) contains the solution values y_i at the point X, $i = 1, 2, \dots, N$. These values must not be changed.

F - real array of DIMENSION (N).

On exit, F(i) must contain $\frac{\partial f_i}{\partial \epsilon}$ at the point (x, y) , $i = 1, 2, \dots, N$.

N - INTEGER.

On entry, N specifies the number of equations. Its value must not be changed.

JACEPS must be declared as EXTERNAL in the (sub)program from which D02RAF is called.

JACGEP - SUBROUTINE, supplied by the user.

JACGEP must evaluate the derivatives $\frac{\partial g_i}{\partial \epsilon}$ if continuation is being used. Otherwise the user is advised to use D02GAX as the actual

parameter for JACGEP.

The specification of JACGEP is:

```
SUBROUTINE JACGEP(EPS,YA,YB,
  1 BCEP,N)
  INTEGER N
  real EPS,YA(N),YB(N),BCEP(N).
```

EPS - real.

On entry, EPS must specify the value of the continuation parameter ϵ . Its value must not be changed.

YA - real array of DIMENSION (N).

On entry, YA(i) contains the value of $y_i(A)$, $i = 1, 2, \dots, N$. These values must not be changed.

YB - real array of DIMENSION (N).

On entry, YB(i) contains the value of $y_i(B)$, $i = 1, 2, \dots, N$. These values must not be changed.

BCEP - real array of DIMENSION (N).

On exit, BCEP(i) must contain $\frac{\partial g_i}{\partial \epsilon}$, $i = 1, 2, \dots, N$.

N - INTEGER.

On entry, N specifies the number of equations. Its value must not be changed.

JACGEP must be declared as EXTERNAL in the (sub)program from which D02RAF is called.

WORK - real array of DIMENSION (LWORK).

Used as working space.

LWORK - INTEGER.

On entry, LWORK must specify the dimension of the array WORK.

$LWORK \geq MNP(3N^2 + 6N + 2) + 4N^2 + 3N$

Unchanged on exit.

IWORK - INTEGER array of DIMENSION (LIWORK).

Used as working space.

LIWORK - INTEGER.

On entry, LIWORK must specify the dimension of the array IWORK:

$LIWORK \geq MNP \times (2 \times N + 1) + \sum_{i=1}^{MNP} IJAC \neq 0$

and

$$\text{LIWORK} \geq \text{MNP} \times (2 \times N + 1) + N^2 + 4 \times N + 2.$$

if $\text{IJAC} = 0$.
Unchanged on exit.

IFAIL - INTEGER.

For this routine, the normal use of IFAIL is extended to control the printing of error messages and monitoring information as well as specifying hard or soft failure (see Chapter P01).

Before entry, IFAIL must be set to a value with the decimal expansion cba, where each of the decimal digits c, b and a must have the value 0 or 1.

- a = 0 specifies hard failure, otherwise soft failure;
- b = 0 suppresses error messages, otherwise error messages will be printed (see Section 6);
- c = 0 suppresses monitoring information, otherwise monitoring information will be printed.

The recommended value for inexperienced users is 110 (i.e. hard failure with all error messages and monitoring information printed). Unless the routine detects an error (see Section 6), IFAIL contains 0 on exit.

6. Error Indicators and Warnings

Errors detected by the routine:

IFAIL = 1

One or more of the parameters N, MNP, NP, NUMBEG, NUMMIX, TOL, DELEPS, LWORK or LIWORK has been incorrectly set,
or $X(1) \geq X(NP)$
or the mesh points $X(I)$ are not ordered as in (6).

IFAIL = 2

A finer mesh is required for the accuracy requested; that is MNP is not large enough. This error exit normally occurs when the problem being solved is difficult (for example, there is a boundary layer) and high accuracy is requested. A poor initial choice of mesh points will make this error exit more likely.

IFAIL = 3

The Newton iteration has failed to converge. There are several possible causes for this error:

- (i) Faulty coding in one of the Jacobian calculation routines;
- (ii) If $\text{IJAC} = 0$ then inaccurate Jacobians may have been calculated numerically (this is a very unlikely cause);
- (iii) A poor initial mesh or initial approximate solution has been selected either by the user or by default or there are not enough points in the initial mesh. Possibly, the user should try the continuation facility.

IFAIL = 4

The Newton iteration has reached roundoff error level. It could be however that the answer returned is satisfactory. The error is likely to occur if too high an accuracy is requested.

IFAIL = 5

The Jacobian calculated by JACOBG (or the equivalent matrix calculated by numerical differentiation) is singular. This may occur due to faulty coding of JACOBG or, in some circumstances, to a zero initial choice of approximate solution (such as is chosen when INIT = 0).

IFAIL = 6

There is no dependence on ϵ when continuation is being used. This can be due to faulty coding of JACEPS or JACGEP or, in some circumstances, to a zero initial choice of approximate solution (such as is chosen when INIT = 0).

IFAIL = 7

DELEPS is required to be less than macheps for continuation to proceed. It is likely that either the problem (3) has no solution for some value near the current value of EPS (see the advisory print out from D02RAF) or that the problem is so difficult that even with continuation it is unlikely to be solved using this routine. If the latter cause is suspected then using more mesh points initially may help.

IFAIL = 8

IFAIL = 9

Indicates that a serious error has occurred in

D02 - Ordinary Differential Equations

a call to D02RAF or D02RAR respectively. Check all array subscripts and subroutine parameter lists in calls to D02RAF. Seek expert help.

7. Auxiliary Routines

Details are distributed to sites in machine-readable form.

8. Timing

There are too many factors present to quantify the timing. The time taken is negligible only on very simple problems.

9. Storage

The storage occupied by internally declared arrays is 250 *real* elements.

10. Accuracy

The solution returned by the routine will be accurate to the user's tolerance as defined by the relation (5) except in extreme circumstances. The final error estimate over the whole mesh for each component is given in the array ABT. If too many points are specified in the initial mesh, the solution may be more accurate than requested

and the error may not be approximately equidistributed.

11. Further Comments

The routine uses a labelled COMMON block AD02RA.

The user is strongly recommended to set IFAIL to obtain self-explanatory error messages, and also monitoring information about the course of the computation. The user may select the channel numbers on which this output is to appear by calls of X04AAF (for error messages) or X04ABF (for monitoring information) – see Section 13 for an example. Otherwise the default channel numbers will be used, as specified in the implementation document.

In the case where the user wishes to solve a sequence of similar problems, the use of the final mesh and solution from one case as the initial mesh is strongly recommended for the next.

12. Keywords

Boundary Value Problems,
Deferred Correction,
Differential Equations, ordinary,
Finite-Difference Method.

13. Example

We solve the differential equation

$$y''' = -yy'' - 2\epsilon(1-y^2)$$

with $\epsilon = 1$ and boundary conditions

$$y(0) = y'(0) = 0, y'(10) = 1$$

to an accuracy specified by TOL = 1.0E-4. The continuation facility is used with the continuation parameter ϵ introduced as in the differential equation above and with DELEPS = 0.1 initially. (The continuation facility is not needed for this problem and is used here for illustration.)

Note the calls to X04AAF and X04ABF prior to the call to D02RAF.

WARNING: This single precision example program may require amendment for certain implementations. The results produced may not be the same. If in doubt, please seek further advice (see *Essential Introduction to the Library Manual*).

13.1. Program Text

```
C      D02RAF EXAMPLE PROGRAM TEXT
C      MARK 8 RELEASE, NAG COPYRIGHT 1979.
C      .. LOCAL SCALARS ..
      REAL DELEPS, TOL
      INTEGER I, IFAIL, IJAC, INIT, J, LIWORK, LWORK, MNP, N, NOUT,
      * NP, NUMBEG, NUMMIX
C      .. LOCAL ARRAYS ..
      REAL ABT(3), WORK(2128), X(40), Y(3,40)
      INTEGER IWORK(303)
C      .. SUBROUTINE REFERENCES ..
```

D02RAF

```

C      D02RAF, X04AAF, X04ABF
C
C      EXTERNAL FCN, G, JACEPS, JACGEP, JACOBF, JACOBG
DATA NOUT /6/
WRITE (NOUT,99997)
WRITE (NOUT,99996)
CALL X04AAF(1, NOUT)
CALL X04ABF(1, NOUT)
TOL = 1.0E-4
LWORK = 2128
LIWORK = 303
MNP = 40
N = 3
NP = 17
NUMBEG = 2
NUMMIX = 0
X(1) = 0.0
X(NP) = 10.0
INIT = 0
DELEPS = 0.1
IJAC = 1
IFAIL = 111
CALL D02RAF(N, MNP, NP, NUMBEG, NUMMIX, TOL, INIT, X, Y, 3,
* ABT, FCN, G, IJAC, JACOBF, JACOBG, DELEPS, JACEPS, JACGEP,
* WORK, LWORK, IWORK, LIWORK, IFAIL)
IF (IFAIL.NE.0) GO TO 20
WRITE (NOUT,99999) NP
WRITE (NOUT,99998) (X(J),(Y(I,J),I=1,N),J=1,NP)
WRITE (NOUT,99995) (ABT(I),I=1,N)
20 STOP
99999 FORMAT (27H0SOLUTION ON FINAL MESH OF , I2, 7H POINTS/7X,
* 4HX(I), 5X, 5HY1(I), 8X, 5HY2(I), 8X, 5HY3(I))
99998 FORMAT (1X, 0PF10.3, 1P3E13.4)
99997 FORMAT (4(1X), 31H D02RAF EXAMPLE PROGRAM RESULTS/1X)
99996 FORMAT (1X/37HOCALCULATION USING ANALYTIC JACOBIANS)
99995 FORMAT (38H0MAXIMUM ESTIMATED ERROR BY COMPONENTS/1H , 10X,
* 1P3E13.4)
END
SUBROUTINE FCN(X, EPS, Y, F, M)
C     .. SCALAR ARGUMENTS ..
REAL EPS, X
INTEGER M
C     .. ARRAY ARGUMENTS ..
REAL F(M), Y(M)
C
F(1) = Y(2)
F(2) = Y(3)
F(3) = -Y(1)*Y(3) - 2.0*(1.0-Y(2)*Y(2))*EPS
RETURN
END
SUBROUTINE G(EPS, Y, Z, AL, M)
C     .. SCALAR ARGUMENTS ..
REAL EPS
INTEGER M
C     .. ARRAY ARGUMENTS ..

```

D02 - Ordinary Differential Equations

```

REAL AL(M), Y(M), Z(M)
C
AL(1) = Y(1)
AL(2) = Y(2)
AL(3) = Z(2) - 1.0
RETURN
END
SUBROUTINE JACEPS(X, EPS, Y, F, M)
.. SCALAR ARGUMENTS ..
REAL EPS, X
INTEGER M
.. ARRAY ARGUMENTS ..
REAL F(M), Y(M)
C
F(1) = 0.0
F(2) = 0.0
F(3) = -2.0*(1.0-Y(2)*Y(2))
RETURN
END
SUBROUTINE JACCEP(EPS, Y, Z, AL, M)
.. SCALAR ARGUMENTS ..
REAL EPS
INTEGER M
.. ARRAY ARGUMENTS ..
REAL AL(M), Y(M), Z(M)
C
.. LOCAL SCALARS ..
INTEGER I
C
DO 20 I=1,3
AL(I) = 0.0
20 CONTINUE
RETURN
END
SUBROUTINE JACOBF(X, EPS, Y, F, M)
.. SCALAR ARGUMENTS ..
REAL EPS, X
INTEGER M
.. ARRAY ARGUMENTS ..
REAL F(M,M), Y(M)
C
.. LOCAL SCALARS ..
INTEGER I, J
C
DO 40 I=1,3
DO 20 J=1,3
F(I,J) = 0.0
20 CONTINUE
40 CONTINUE
F(1,2) = 1.0
F(2,3) = 1.0
F(3,1) = -Y(3)
F(3,2) = 4.0*Y(2)*EPS
F(3,3) = -Y(1)
RETURN

```

D02RAF

```

END
SUBROUTINE JACOBG(EPS Y, Z, A, B, M)
C .. SCALAR ARGUMENTS ..
REAL EPS
INTEGER M
C .. ARRAY ARGUMENTS ..
REAL A(M,M), B(M,M), Y(M), Z(M)
C .. LOCAL SCALARS ..
INTEGER I, J
C ..
DO 40 I=1,3
  DO 20 J=1,3
    A(I,J) = 0.0
    B(I,J) = 0.0
20 CONTINUE
40 CONTINUE
  A(1,1) = 1.0
  A(2,2) = 1.0
  B(3,2) = 1.0
RETURN
END

```

13.2. Program Data

None.

13.3. Program Results

D02RAF EXAMPLE PROGRAM RESULTS

CALCULATION USING ANALYTIC JACOBIANS

D02RAF MONITORING INFORMATION

MONITORING NEWTON ITERATION

```

NUMBER OF POINTS IN CURRENT MESH = 17
CORRECTION NUMBER 0 RESIDUAL SHOULD BE .LE. 1.00E 00
ITERATION NUMBER 0 RESIDUAL = 1.00E 00
  SQUARED NORM OF CORRECTION = 9.90E 01
  SQUARED NORM OF GRADIENT = 1.00E 00
  SCALAR PRODUCT OF CORRECTION AND GRADIENT = 1.00E 00
ITERATION NUMBER 1 RESIDUAL = 5.59E-01
CONTINUATION PARAMETER EPSILON = 2.00E-01  DELEPS = 2.00E-01

```

MONITORING NEWTON ITERATION

Monitoring information omitted.

```

NUMBER OF NEW POINTS 5
MONITORING NEWTON ITERATION

```

```

NUMBER OF POINTS IN CURRENT MESH = 33
CORRECTION NUMBER 1 RESIDUAL SHOULD BE .LE. 1.22E-05
ITERATION NUMBER 0 RESIDUAL = 3.58E-04

```

D02 - Ordinary Differential Equations

SQUARED NORM OF CORRECTION = 1.70E-06
 SQUARED NORM OF GRADIENT = 2.89E-07
 SCALAR PRODUCT OF CORRECTION AND GRADIENT = 1.28E-07
 ITERATION NUMBER 1 RESIDUAL = 2.70E-08
 MESH SELECTION

NUMBER OF NEW POINTS 0
 CORRECTION NUMBER 1 ESTIMATED MAXIMUM ERROR = 6.92E-05

ESTIMATED ERROR BY COMPONENTS
 6.92E-05 1.81E-05 6.42E-05
 SOLUTION ON FINAL MESH OF 33 POINTS

X(I)	Y1(I)	Y2(I)	Y3(I)
0.000	0.0000E 00	0.0000E 00	1.6872E 00
0.062	3.2142E-03	1.0155E-01	1.5626E 00
0.125	1.2532E-02	1.9536E-01	1.4398E 00
0.188	2.7476E-02	2.8159E-01	1.3203E 00
0.250	4.7578E-02	3.6049E-01	1.2054E 00
0.312	1.0149E-01	4.9760E-01	9.9235E-01
0.375	1.7093E-01	6.0965E-01	8.0477E-01
0.438	2.5299E-01	6.9991E-01	6.4376E-01
0.500	3.0954E-01	7.4673E-01	5.5629E-01
0.562	3.6950E-01	7.8708E-01	4.7842E-01
0.625	4.9776E-01	8.5129E-01	3.4901E-01
0.688	6.3461E-01	8.9774E-01	2.5017E-01
0.750	7.7761E-01	9.3077E-01	1.7628E-01
0.812	9.7480E-01	9.5983E-01	1.0768E-01
0.875	1.1768E 00	9.7733E-01	6.3852E-02
0.938	1.3815E 00	9.8758E-01	3.6741E-02
1.000	1.5362E 00	9.9224E-01	2.3792E-02
1.062	1.6915E 00	9.9523E-01	1.5143E-02
1.125	2.0031E 00	9.9828E-01	5.8470E-03
1.188	2.1591E 00	9.9900E-01	3.5275E-03
1.250	2.3153E 00	9.9943E-01	2.0894E-03
1.312	2.6277E 00	9.9983E-01	7.0180E-04
1.375	3.2526E 00	9.9998E-01	1.1337E-04
1.438	3.8776E 00	1.0000E 00	6.5600E-06
1.500	4.5026E 00	1.0000E 00	5.7085E-06
1.562	5.1276E 00	1.0000E 00	-1.2928E-06
1.625	5.7526E 00	1.0000E 00	5.4482E-07
1.688	6.3776E 00	1.0000E 00	-2.2880E-07
1.750	7.0026E 00	1.0000E 00	8.9176E-08
1.812	7.6276E 00	1.0000E 00	-3.5784E-08
1.875	8.2526E 00	1.0000E 00	1.5339E-08
1.938	8.8776E 00	1.0000E 00	-6.7001E-09
2.000	9.5026E 00	1.0000E 00	3.5393E-09

MAXIMUM ESTIMATED ERROR BY COMPONENTS

6.9244E-05 1.8051E-05 6.4213E-05

D02GBF - NAG FORTRAN Library Routine Document

NOTE: before using this routine, please read the appropriate implementation document to check the interpretation of ***bold italicised*** terms and other implementation-dependent details. The routine name may be precision-dependent.

1. Purpose

D02GBF solves a general linear two-point boundary value problem for a system of ordinary differential equations using a deferred correction technique.

2. Specification

```
SUBROUTINE D02GBF (A, B, N, TOL, FCNF, FCNG, C, D, GAM, MNP,
1   X, Y, NP, W, LW, IW, LIW, IFAIL)
C   INTEGER N, MNP, NP, LW, IW(LIW), LIW, IFAIL
C   real A, B, TOL, C(N,N), D(N,N), GAM(N), X(N,MNP), Y(N,MNP),
C   1   W(LW)
C   EXTERNAL FCNF, FCNG
```

3. Description

D02GBF solves the linear two-point boundary value problem for a system of N ordinary differential equations in the range (A, B) . The system is written in the form

$$y' = F(x)y + G(x) \quad (1)$$

and the boundary conditions are written in the form

$$Cy(A) + Dy(B) = \gamma \quad (2)$$

Here $F(x)$, C and D are $N \times N$ matrices, and $G(x)$ and γ are N -component vectors. The approximate solution to (1) and (2) is found using a finite-difference method with deferred correction. The algorithm is a specialisation of that used in subroutine D02RAF which solves a nonlinear version of (1) and (2). The nonlinear version of the algorithm is described fully in [1]. The user supplies an absolute error tolerance and may also supply an initial mesh for the construction of the finite-difference equations (alternatively a default mesh is used). The algorithm constructs a solution on a mesh defined by adding points to the initial mesh. This solution is chosen so that the error is everywhere less than the user's tolerance and so that the error is approximately equidistributed on the final mesh. The solution is returned on this final mesh.

If the solution is required at a few specific points then these should be included in the initial mesh. If, on the other hand, the solution is required at several specific points, then the user should use the interpolation routines provided in the E01 chapter if these points do not themselves form a convenient mesh.

4. References

- [1] PEREYRA, V.
PASVA3: An Adaptive Finite-Difference Fortran Program for First Order Nonlinear, Ordinary Boundary Problems.
In Childs, B., Scott, M., Daniel, J. W., Denman, E. and Nelson, P. (eds.)
'Codes for Boundary Value Problems in Ordinary Differential Equations.'
Springer - Verlag, Lecture Notes in Computer Science, 76, 1979.

5. Parameters

A - *real*.

On entry, A must specify the left hand boundary point, A .
Unchanged on exit.

B - *real*.

On entry, B must specify the right hand boundary point, B .
 $B > A$.
Unchanged on exit.

N - INTEGER.

On entry, N must specify the number of equations; that is N is the order of system (1).
Unchanged on exit.

TOL - *real*.

On entry, TOL must specify a positive absolute error tolerance. If

$$A = x_1 < x_2 < \dots < x_{NP} = B$$

is the final mesh, $z(x)$ is the approximate

D02GBF

solution from D02GBF and $y(x)$ is the true solution of equations (1) and (2) then, except in extreme cases, it is expected that

$$\|z - y\| \leq TOL \quad (3)$$

where

$$\|u\| = \max_{1 \leq i \leq N} \max_{1 \leq j \leq NP} |u_i(x_j)|.$$

Unchanged on exit.

FCNF - SUBROUTINE, supplied by the user.

FCNF must evaluate the matrix $F(x)$ in (1) at a general point $x = X$ and place $F_{ij}(x)$ in $F(i,j)$, $i,j = 1,2,\dots,N$. Its specification is :-

SUBROUTINE FCNF(X,F)
real X, F(n,n)

where n is the numerical value of N.

X - real.

On entry, X specifies the value of the independent variable x. Its value must not be changed.

F - real array of DIMENSION (n,n).

On exit, $F(i,j)$ must contain the (i,j) (th) element of the matrix $F(x)$, $i,j = 1,2,\dots,n$. See Section 13 for an example.

FCNF must be declared as EXTERNAL in the (sub)program from which D02GBF is called.

FCNG - SUBROUTINE, supplied by the user.

FCNG must evaluate the vector $G(x)$ in (1) at a general point $x = X$ and place $G_i(x)$ in $G(I)$, $I = 1,2,\dots,N$. Its specification is:-

SUBROUTINE FCNG(X,G)
real X, G(n)

where n is the numerical value of N.

X - real.

On entry, X specifies the value of the independent variable x. Its value must not be changed.

G - real array of DIMENSION (n)

On exit, $G(i)$ must contain the i (th) element of the vector $G(x)$. See Section 13 for an example.

FCNG must be declared as EXTERNAL in the (sub)program from which D02GBF is called.

C, D - real arrays of DIMENSION (N,N)**GAM - real array of DIMENSION (N)**

Before entry the arrays C and D must be set to the matrices C and D in (2). GAM must be set to the vector γ in (2).

The routine re-orders the rows of C and D and the components of GAM so that the boundary conditions are in the order:

- (i) conditions on $y(A)$ only
- (ii) condition involving $y(A)$ and $y(B)$
- (iii) conditions on $y(B)$ only.

The routine will be slightly more efficient if the arrays C,D and GAM are ordered in this way before entry, and in this event they will be unchanged on exit.

MNP - INTEGER.

On entry, MNP must specify the maximum permitted number of mesh points.

$$MNP \geq 32.$$

Unchanged on exit.

X - real array of DIMENSION (MNP).

On entry, if $NP \geq 4$ (see NP below) the elements $X(1),X(2),\dots,X(NP)$ must define an initial mesh satisfying

$$A = X(1) < X(2) < \dots < X(NP) = B \quad (4)$$

On exit, $X(1),X(2),\dots,X(NP)$ define the final mesh (with the returned value of NP) satisfying the relation (4).

Y - real array of DIMENSION (N,MNP).

On successful exit, the array Y contains the approximate solution $z(x)$ satisfying (3), on the final mesh, that is

$$Y(J,I) = z_j(x_i), \quad I = 1,2,\dots,NP, \quad J = 1,2,\dots,N.$$

where NP is the number of points in the final mesh.

The remaining columns of Y are not used.

NP - INTEGER.

On entry, NP must satisfy $NP = 0$ or $4 \leq NP \leq MNP$. If $NP = 0$ on entry, a default value of 4 for NP and a corresponding equispaced mesh $X(1),X(2),\dots,X(NP)$ are used. If $NP \geq 4$ on entry, then the user must define an initial mesh X as in (4) above.

On exit, NP contains the number of points in the final (returned) mesh.

W - real array of DIMENSION (LW).

Used as workspace.

D02 - Ordinary Differential Equations

LW - INTEGER.

On entry, LW must specify the length of the array W.

$$LW \geq MNP \times (3N^2 + 5N + 2) + 3N^2 + 5N$$

Unchanged on exit.

IW - INTEGER array of DIMENSION (LIW).

Used as workspace.

LIW - INTEGER.

On entry, LIW must specify the length of the array IW.

$$LIW \geq MNP \times (2N + 1) + N$$

Unchanged on exit.

IFAIL - INTEGER.

For this routine, the normal use of IFAIL is extended to control the printing of error and warning messages as well as specifying hard or soft failure (see Chapter P01).

Before entry, IFAIL must be set to a value with the decimal expansion cba, where each of the decimal digits c, b and a must have the value 0 or 1.

a = 0 specifies hard failure, otherwise soft failure;

b = 0 suppresses error messages, otherwise error messages will be printed (see Section 6);

c = 0 suppresses warning messages, otherwise warning messages will be printed (see Section 6).

The recommended value for inexperienced users is 110 (i.e. hard failure with all messages printed).

Unless the routine detects an error (see Section 6), IFAIL contains 0 on exit.

6. Error Indicators and Warnings**Errors detected by the routine:-**

For some errors, the routine outputs an explanatory error message on the current error message unit (see NAG Library routine X04AAF), unless suppressed by the value of IFAIL on entry.

IFAIL = 1

One or more of the parameters N, TOL, NP, MNP, LW or LIW is incorrectly set, $B \leq A$ or the condition (4) on X is not satisfied.

IFAIL = 2

Either a row of C and the corresponding row of D are both identically zero (that is the boundary conditions are rank deficient) or one of the matrices C or D is identically zero (that is the problem is of initial value and not boundary value type).

IFAIL = 3

The routine has failed to find a solution to the specified accuracy. There are a variety of possible reasons including:

- (i) The boundary conditions are rank deficient. This will usually be indicated by a message that the Jacobian is singular (simple cases of rank deficiency are flagged with IFAIL = 2).
- (ii) Not enough mesh points are permitted in order to attain the required accuracy. This is indicated by NP = MNP on return from a call to D02GBF. This difficulty may be aggravated by a poor initial choice of mesh points.
- (iii) The accuracy requested cannot be attained on the computer being used.

IFAIL = 4

A serious error has occurred in a call to D02GBF. Check all array subscripts and subroutine parameter lists in calls to D02GBF. Seek expert help.

7. Auxiliary Routines

Details are distributed to sites in machine-readable form.

8. Timing

This depends on the difficulty of the problem, the number of mesh points (and meshes) used and the number of deferred corrections.

9. Storage

The storage occupied by internally declared arrays is 250 *real* elements.

10. Accuracy

The solution returned by the routine will be accurate to the user's tolerance as defined by the relation (3) except in extreme circumstances. If too many points are specified in the initial mesh, the solution may be more accurate than requested and the error may not be approximately equidistributed.

D02GBF

11. Further Comments

The routine uses a labelled COMMON block ADO2RA.

The user is strongly recommended to set IFAIL to obtain self-explanatory error messages, and also monitoring information about the course of the computation. The user may select the channel numbers on which this output is to appear by calls of X04AAF (for error messages) or X04ABF (for monitoring information) – see Section 13 for an example. Otherwise the default channel numbers will be used, as specified in the

implementation document.

In the case where the user wishes to solve a sequence of similar problems, the use of the final mesh from one case is strongly recommended as the initial mesh for the next.

12. Keywords

Linear Boundary Value Problems,
Deferred Correction,
Ordinary Differential Equations,
Finite-Difference Method.

13. Example

We solve the problem (written as a first order system)

$$\epsilon y'' + y' = 0$$

with boundary conditions

$$y(0) = 0, y(1) = 1$$

for the cases $\epsilon = 10^{-1}$ and $\epsilon = 10^{-2}$ using the default initial mesh in the first case, and the final mesh of the first case as initial mesh for the second (more difficult) case. We give the solution and the error at each mesh point to illustrate the accuracy of the method given the accuracy request TOL = 1.0E-3.

Note the calls to X04AAF and X04ABF prior to the call to D02GBF.

13.1. Program Text

WARNING: This single precision example program may require amendment for certain implementations. The results produced may not be the same. If in doubt, please seek further advice (see *Essential Introduction to the Library Manual*).

```

C      D02GBF EXAMPLE PROGRAM TEXT
C      MARK 8 RELEASE. NAG COPYRIGHT 1979.
C      .. SCALARS IN COMMON ..
C      REAL EPS
C      ..
C      .. LOCAL SCALARS ..
C      REAL A, B, TOL
C      INTEGER I, IFAIL, J, LIW, LW, MNP, N, NOUT, NP
C      .. LOCAL ARRAYS ..
C      REAL C(2,2), D(2,2), GAM(2), W(2052), X(70), Y(2,70)
C      INTEGER IW(352)
C      .. FUNCTION REFERENCES ..
C      REAL SOL
C      .. SUBROUTINE REFERENCES ..
C      D02GBF, X04AAF, X04ABF
C      ..
C      EXTERNAL FCNF, FCNG
C      COMMON EPS
C      DATA NOUT /6/
C      WRITE (NOUT,99999)
C      TOL = 1.0E-3
C      MNP = 70
C      NP = 0
C      N = 2

```

D02 - Ordinary Differential Equations

```

LW = 2052
LIW = 352
A = 0.0
B = 1.0
CALL X04AAF(1, NOUT)
CALL X04ABF(1, NOUT)
DO 40 I=1,2
  GAM(I) = 0.0
  DO 20 J=1,2
    C(I,J) = 0.0
    D(I,J) = 0.0
20   CONTINUE
40 CONTINUE
C(1,1) = 1.0
D(2,1) = 1.0
GAM(2) = 1.0
DO 80 I=1,2
  EPS = 10.0**(-I)
  WRITE (NOUT,99998) EPS
  IFAIL = 111
  CALL D02GBF(A, B, N, TOL, FCNF, FCNG, C, D, GAM, MNP, X,
*     Y, NP, W, LW, IW, LIW, IFAIL)
  IF (IFAIL.GT.0) GO TO 100
  WRITE (NOUT,99997) NP
  DO 60 J=1,NP
    W(J) = Y(1,J) - SOL(X(J))
60   CONTINUE
  WRITE (NOUT,99996) (X(J),Y(1,J),W(J),J=1,NP)
80 CONTINUE
100 STOP
99999 FORMAT (4(1X/), 31H D02GBF EXAMPLE PROGRAM RESULTS/1X)
99998 FORMAT (1X/24HOPROBLEM WITH EPSILON = , E10.2)
99997 FORMAT (37HOSOLUTION AND ERROR ON FINAL MESH OF , I2,
*    7H POINTS/7X, 4HX(I), 4X, 6HY(1,I), 5X, 5HERROR)
99996 FORMAT (1X, 2F10.4, E12.2)
END
SUBROUTINE FCNF(X, F)
C .. SCALAR ARGUMENTS ..
REAL X
C .. ARRAY ARGUMENTS ..
REAL F(2,2)
C .. SCALARS IN COMMON ..
REAL EPS
C
COMMON EPS
F(1,1) = 0.0
F(1,2) = 1
F(2,1) = 0.0
F(2,2) = -1.0/EPS
RETURN
END
SUBROUTINE FCNG(X, G)
C .. SCALAR ARGUMENTS ..
REAL X

```

D02GBF

D02 - Ordinary Differential Equations

```

C     .. ARRAY ARGUMENTS ..
REAL G(2)
C
C     ..
G(1) = 0.0
G(2) = 0.0
RETURN
END
REAL FUNCTION SOL(X)
C     .. SCALAR ARGUMENTS ..
REAL X
C
C     ..
C     .. SCALARS IN COMMON ..
REAL EPS
C
C     ..
C     .. FUNCTION REFERENCES ..
REAL EXP
C
C     ..
COMMON EPS
SOL = (EXP(-X/EPS)-1.0)/(EXP(-1.0/EPS)-1.0)
RETURN
END

```

13.2. Program Data

None.

13.3. Program Results

D02GBF EXAMPLE PROGRAM RESULTS

PROBLEM WITH EPSILON = 0.10E 00

D02GBF MONITORING INFORMATION

NUMBER OF POINTS IN CURRENT MESH = 15
 CORRECTION NUMBER 0 ESTIMATED MAXIMUM ERROR = 6.59E-02

ESTIMATED ERROR BY COMPONENTS
 6.57E-03 6.59E-02

NUMBER OF POINTS IN CURRENT MESH = 15
 CORRECTION NUMBER 1 ESTIMATED MAXIMUM ERROR = 3.60E-03

ESTIMATED ERROR BY COMPONENTS
 3.61E-04 3.60E-03

NUMBER OF POINTS IN CURRENT MESH = 15
 CORRECTION NUMBER 2 ESTIMATED MAXIMUM ERROR = 4.36E-04

ESTIMATED ERROR BY COMPONENTS
 4.45E-05 4.36E-04

SOLUTION AND ERROR ON FINAL MESH OF 15 POINTS

X(I)	Y(1,I)	ERROR
0.0000	0.0000	0.00E 00
0.0278	0.2425	0.14E-05
0.0556	0.4263	-0.11E-05

0.1111	0.6708	0.16E-05
0.1667	0.8112	0.12E-04
0.2222	0.8917	0.18E-04
0.2778	0.9379	0.15E-04
0.3333	0.9644	0.11E-04
0.4444	0.9883	0.24E-04
0.5556	0.9962	0.36E-04
0.6667	0.9988	0.18E-04
0.7500	0.9995	0.59E-05
0.8333	0.9998	0.12E-05
0.9167	0.9999	-0.91E-06
1.0000	1.0000	0.00E 00

PROBLEM WITH EPSILON = 0.10E-01

D02GBF MONITORING INFORMATION

NUMBER OF POINTS IN CURRENT MESH = 49
 CORRECTION NUMBER 0 ESTIMATED MAXIMUM ERROR = 5.29E-02

ESTIMATED ERROR BY COMPONENTS

5.29E-04 5.29E-02

NUMBER OF POINTS IN CURRENT MESH = 49
 CORRECTION NUMBER 1 ESTIMATED MAXIMUM ERROR = 1.26E-03

ESTIMATED ERROR BY COMPONENTS

1.27E-05 1.26E-03

NUMBER OF POINTS IN CURRENT MESH = 49
 CORRECTION NUMBER 2 ESTIMATED MAXIMUM ERROR = 2.45E-04

ESTIMATED ERROR BY COMPONENTS

1.67E-06 2.45E-04

SOLUTION AND ERROR ON FINAL MESH OF 49 POINTS

X(I)	Y(1,I)	ERROR
0.0000	0.0000	0.00E 00
0.0009	0.0884	0.69E-08
0.0019	0.1690	0.16E-07
0.0028	0.2425	0.25E-07
0.0037	0.3095	0.33E-07
0.0046	0.3706	0.41E-07
0.0056	0.4262	0.47E-07
0.0065	0.4770	0.53E-07
0.0074	0.5232	0.59E-07
0.0083	0.5654	0.64E-07
0.0093	0.6038	0.70E-07
0.0111	0.6708	0.72E-07
0.0130	0.7265	0.69E-07
0.0148	0.7727	0.92E-07
0.0167	0.8111	0.11E-06
0.0185	0.8431	0.11E-06
0.0204	0.8696	0.12E-06
0.0222	0.8916	0.12E-06
0.0241	0.9100	0.12E-06
0.0259	0.9252	0.13E-06

D02GBF

D02 - Ordinary Differential Equations

0.0278	0.9378	0.13E-06
0.0306	0.9529	0.12E-06
0.0333	0.9643	0.13E-06
0.0361	0.9730	0.15E-06
0.0389	0.9795	0.15E-06
0.0417	0.9845	0.15E-06
0.0444	0.9883	0.15E-06
0.0472	0.9911	0.14E-06
0.0500	0.9933	0.14E-06
0.0528	0.9949	0.13E-06
0.0556	0.9961	0.13E-06
0.0648	0.9985	0.33E-06
0.0741	0.9994	0.13E-05
0.0833	0.9998	0.15E-05
0.0926	0.9999	0.69E-06
0.1019	1.0000	0.36E-06
0.1111	1.0000	0.28E-06
0.1389	1.0000	0.16E-05
0.1667	1.0000	0.94E-06
0.2222	1.0000	0.50E-06
0.2778	1.0000	-0.15E-06
0.3333	1.0000	0.39E-06
0.4444	1.0000	-0.34E-06
0.5556	1.0000	0.47E-06
0.6667	1.0000	-0.13E-06
0.7500	1.0000	0.26E-06
0.8333	1.0000	0.20E-07
0.9167	1.0000	0.19E-06
1.0000	1.0000	0.00E 00

C05NBF - NAG FORTRAN Library Routine Document

NOTE: before using this routine, please read the appropriate implementation document to check the interpretation of ***bold italicised*** terms and other implementation-dependent details. The routine name may be precision-dependent.

1. Purpose

C05NBF is an easy-to-use routine to find a zero of a system of N nonlinear functions in N variables by a modification of the Powell hybrid method.

2. Specification

```
SUBROUTINE C05NBF (FCN, N, X, FVEC, XTOL, WA, LWA, IFAIL)
C   INTEGER N, LWA, IFAIL
C   real X(N), FVEC(N), XTOL, WA(LWA)
C   EXTERNAL FCN
```

3. Description

C05NBF is based upon the MINPACK routine HYBRD1. It chooses the correction at each step as a convex combination of the Newton and scaled gradient directions. Under reasonable conditions this guarantees global convergence for starting points far from the solution and a fast rate of convergence. The Jacobian is updated by the rank-1 method of Broyden. At the starting point the Jacobian is approximated by forward differences, but these are not used again until the rank-1 method fails to produce satisfactory progress.

4. References

- [1] POWELL, M.J.D.
A hybrid method for nonlinear algebraic equations. In 'Numerical Methods for Nonlinear Algebraic Equations', Ed. Rabinowitz, P., Gordon and Breach, 1970.
- [2] MORE, J.J., GARBOW, B.S. and HILLSTROM, K.E.
User Guide for MINPACK-1.
ANL-80-74, Argonne National Laboratory.

5. Parameters

FCN - SUBROUTINE, supplied by the user.

FCN must calculate the values of the functions at X and return these in the vector FVEC.

Its specification is:-

```
SUBROUTINE FCN(N,X,FVEC,IFLAG)
INTEGER N,IFLAG
real X(N),FVEC(N)
```

N - INTEGER.

On entry, N contains the number of equations. The value of N must not be changed by FCN.

X - *real* array of DIMENSION (N).

On entry, X contains the point at which the functions are to be evaluated. The values in X must not be changed by FCN.

FVEC - *real* array of DIMENSION (N).

On exit, unless IFLAG is reset to a negative number, $FVEC(i)$ must contain the value of the (i) th function evaluated at X .

IFLAG - INTEGER.

In general, IFLAG should not be reset by FCN. If, however, the user wishes to terminate execution (perhaps because some illegal point X has been reached) then IFLAG should be set to a negative integer. This value will be returned through IFAIL.

FCN must be declared as EXTERNAL in the (sub)program from which C05NBF is called.

N - INTEGER.

On entry, N must specify the number of equations. $N > 0$.

Unchanged on exit.

X - *real* array of DIMENSION at least (N).

Before entry, $X(j)$ must be set to a guess at the j (th) component of the solution ($j = 1, 2, \dots, N$).

On exit, X contains the final estimate of the solution vector.

FVEC - *real* array of DIMENSION at least (N).

On exit, FVEC contains the function values at the final point, X.

XTOL - *real*.

On entry, XTOL must specify the accuracy in X to which the solution is required. $XTOL \geq 0.0$. If XTOL is less than machine precision (see NAG FORTRAN Library routine X02AAF), then machine precision is used. See Section 10. The recommended value is the square root of machine precision.

Unchanged on exit.

WA - *real* array of DIMENSION at least (LWA).

Used as workspace.

LWA - INTEGER.

On entry, LWA must specify the dimension of the array WA. $LWA \geq \frac{1}{2}N \times (3 \times N + 13)$.

Unchanged on exit.

IFAIL - INTEGER.

Before entry, IFAIL must be assigned a value. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

Unless the routine detects an error (see next section), IFAIL contains 0 on exit.

6. Error Indicators and Warnings

Errors detected by the routine:-

IFAIL < 0

This indicates an exit from C05NBF because the user has set IFLAG negative in FCN. The value of IFAIL will be the same as the user's setting of IFLAG.

IFAIL = 1

On entry, $N \leq 0$, $XTOL < 0.0$, or $LWA < \frac{1}{2}N \times (3 \times N + 13)$.

IFAIL = 2

There have been at least $200 \times (N+1)$ evaluations of FCN. Consider restarting the calculation from the final point held in X.

IFAIL = 3

No further improvement in the approximate solution X is possible; XTOL is too small.

IFAIL = 4

The iteration is not making good progress. This failure exit may indicate that the system does not have a zero. Otherwise, rerunning C05NBF from a different starting point may avoid the region of difficulty. Alternatively consider using C05PBF or C05PCF which require an analytic Jacobian.

7. Auxiliary Routines

Details are distributed to sites in machine-readable form.

8. Timing

The time required by C05NBF to solve a given problem depends on N, the behaviour of the functions, the accuracy requested and the starting point. The number of arithmetic operations executed by C05NBF to process each call of FCN is about $11.5 \times N^2$. Unless FCN can be evaluated quickly, the timing of C05NBF will be strongly influenced by the time spent in FCN.

9. Storage

There are no internally declared arrays.

10. Accuracy

C05NBF tries to ensure that $\|X - XSOL\|_2 \leq XTOL \times \|XSOL\|_2$. If this condition is satisfied with $XTOL = 10^{-k}$ then the larger components of X have k significant decimal digits. There is a danger that the smaller components of X may have large relative errors, but the fast rate of convergence of C05NBF usually avoids this possibility. The test assumes that the functions are reasonably well behaved. If this condition is not satisfied, then C05NBF may incorrectly indicate convergence. The validity of the answer can be checked, for example, by rerunning C05NBF with a tighter tolerance.

11. Further Comments

Ideally the problem should be scaled so that at the solution the function values are of comparable magnitude.

12. Keywords

Equations, nonlinear algebraic, easy-to-use; Powell hybrid method, easy-to-use.

13. Example

To determine the values x_1, \dots, x_9 , which satisfy the tridiagonal equations:-

$$\begin{aligned}(1-2x_1)x_1 - 2x_2 &= -1 \\ -x_{i-1} + (3-2x_i)x_i - 2x_{i+1} &= -1, \quad i = 2, 3, \dots, 8 \\ -x_8 + (3-2x_9)x_9 &= -1.\end{aligned}$$

13.1. Program Text

```
C      C05NBF EXAMPLE PROGRAM TEXT
C      MARK 9 RELEASE, NAG COPYRIGHT 1981
C      .. LOCAL SCALARS ..
C      REAL FNORM, TOL
C      INTEGER IFAIL, J, NOUT
C      .. LOCAL ARRAYS ..
C      REAL FVEC(9), WA(180), X(9)
C      .. FUNCTION REFERENCES ..
C      REAL FOSABF, SQRT, X02AAF
C      .. SUBROUTINE REFERENCES ..
C      C05NBF
C
C      ..
C      EXTERNAL FCN
C      DATA NOUT /6/
C      WRITE (NOUT,99999)
C      THE FOLLOWING STARTING VALUES PROVIDE A ROUGH SOLUTION.
C      DO 20 J=1,9
C          X(J) = -1.E0
C 20 CONTINUE
C      TOL = SQRT(X02AAF(0.0))
C      IFAIL = 0
C      CALL C05NBF(FCN, 9, X, FVEC, TOL, WA, 180, IFAIL)
C      FNORM = FOSABF(FVEC,9)
C      WRITE (NOUT,99998) FNORM, IFAIL, (X(J),J=1,9)
C      STOP
99999 FORMAT (4(1X/), 31H C05NBF EXAMPLE PROGRAM RESULTS/1X)
99998 FORMAT (5X, 31H FINAL L2 NORM OF THE RESIDUALS, E12.4//5X,
* 15H EXIT PARAMETER, I10//5X, 27H FINAL APPROXIMATE SOLUTION//5X,
* (5X, 3E12.4))
END
SUBROUTINE FCN(N, X, FVEC, IFLAG)
C      .. SCALAR ARGUMENTS ..
C      INTEGER IFLAG, N
C      .. ARRAY ARGUMENTS ..
C      REAL FVEC(N), X(N)
C
C      ..
C      .. LOCAL SCALARS ..
C      REAL ONE, TEMP, TEMP1, TEMP2, THREE, TWO, ZERO
C      INTEGER K
C
C      ..
C      DATA ZERO, ONE, TWO, THREE /0.E0,1.E0,2.E0,3.E0/
C      DO 20 K=1,N
C          TEMP = (THREE-TWO*X(K))*X(K)
C          TEMP1 = ZERO
C          IF (K.NE.1) TEMP1 = X(K-1)
C          TEMP2 = ZERO
```

```

IF (K.NE.N) TEMP2 = X(K+1)
FVEC(K) = TEMP - TEMP1 - TWO*TEMP2 + JNE
20 CONTINUE
RETURN
END

```

13.2. Program Data

None.

13.3. Program Results

COSNBF EXAMPLE PROGRAM RESULTS

FINAL L2 NORM OF THE RESIDUALS 0.1193E-07

EXIT PARAMETER 0

FINAL APPROXIMATE SOLUTION

```

-0.5707E+00 -0.6816E+00 -0.7017E+00
-0.7042E+00 -0.7014E+00 -0.6919E+00
-0.6658E+00 -0.5960E+00 -0.4164E+00

```

C05PBF - NAG FORTRAN Library Routine Document

NOTE: before using this routine, please read the appropriate implementation document to check the interpretation of **bold italicised** terms and other implementation-dependent details. The routine name may be precision-dependent.

1. Purpose

C05PBF is an easy-to-use routine to find a zero of a system of N nonlinear functions in N variables by a modification of the Powell hybrid method. The user must provide the Jacobian.

2. Specification

```
SUBROUTINE C05PBF (FCN, N, X, FVEC, FJAC, LDFJAC, XTOL, WA,
1   LWA, IFAIL)
C   INTEGER N, LDFJAC, LWA, IFAIL
C   real X(N), FVEC(N), FJAC(LDFJAC,N), XTOL, WA(LWA)
C   EXTERNAL FCN
```

3. Description

C05PBF is based upon the MINPACK routine HYBRJ1. It chooses the correction at each step as a convex combination of the Newton and scaled gradient directions. Under reasonable conditions this guarantees global convergence for starting points far from the solution and a fast rate of convergence. The Jacobian is updated by the rank-1 method of Broyden. At the starting point the Jacobian is calculated, but it is not recalculated until the rank-1 method fails to produce satisfactory progress.

4. References

[1] POWELL, M.J.D.

A hybrid method for nonlinear algebraic equations. In 'Numerical methods for Nonlinear Algebraic Equations', Ed. Rabinowitz, P., Gordon and Breach, 1970.

[2] MORE, J.J., GARBOW, B.S. and HILLSTROM, K.E.

User Guide for MINPACK-1.
ANL-80-74, Argonne National Laboratory.

5. Parameters

FCN - SUBROUTINE, supplied by the user.

Depending upon the value of IFLAG, FCN must either calculate the values of the functions at X or calculate the Jacobian at X , returning these in FVEC and FJAC respectively.

Its specification is:-

```
SUBROUTINE FCN(N,X,FVEC,FJAC,
1   LDFJAC,IFLAG)
INTEGER N,LDFJAC,IFLAG
real X(N),FVEC(N),FJAC(LDFJAC,N)
```

N - INTEGER.

On entry, N contains the number of equations. The value of N must not be changed by FCN.

X - *real* array of DIMENSION (N).

On entry, X contains the point at which the functions or the Jacobian are to be evaluated. The values in X must not be changed by FCN.

FVEC - *real* array of DIMENSION (N).

On exit, unless IFLAG is reset to a negative number, or IFLAG = 2 on entry, FVEC(i) must contain the value of the i (th) function evaluated at X . FVEC must only be changed if IFLAG = 1 on entry.

FJAC - *real* array of DIMENSION (LDFJAC,N).

On exit, unless IFLAG is reset to a negative number, or IFLAG = 1 on entry, FJAC(i,j) must contain the value of $\frac{\partial F_i}{\partial X_j}$, $i,j = 1,\dots,N$

(where F_i is the i (th) function, and X_j the j (th) variable) evaluated at X . FJAC must only be changed if IFLAG = 2 on entry.

LDFJAC - INTEGER.

On entry, LDFJAC specifies the first dimension of FJAC. LDFJAC must not be changed by FCN.

IFLAG - INTEGER.

On entry, IFLAG is 1 if FVEC is to be updated and IFLAG is 2 if FJAC is to be updated.

In general, IFLAG should not be reset by FCN. If, however, the user wishes to terminate execution (perhaps because some illegal point X has been reached) then IFLAG should be set to a negative integer. This value will be returned through IFAIL.

FCN must be declared as EXTERNAL in the (sub)program from which C05PBF is called.

N - INTEGER.

On entry, N must specify the number of equations. $N > 0$.

Unchanged on exit.

X - real array of DIMENSION at least (N).

Before entry, X(j) must be set to a guess at the j(th) component of the solution ($j = 1, 2, \dots, N$).

On exit, X contains the final estimate of the solution vector.

FVEC - real array of DIMENSION at least (N).

On exit, FVEC contains the function values at the final point, X.

FJAC - real array of DIMENSION (LDFJAC,p), where $p \geq N$.

On exit, FJAC contains the orthogonal matrix Q produced by the QR factorisation of the final approximate Jacobian.

LDFJAC - INTEGER.

On entry, LDFJAC must specify the first dimension of FJAC as declared in the calling (sub)program. $LDFJAC \geq N$.

Unchanged on exit.

XTOL - real.

On entry, XTOL must specify the accuracy in X to which the solution is required. $XTOL \geq 0.0$. If XTOL is less than machine precision (see NAG FORTRAN Library routine X02AAF) then machine precision is used. See Section 10. The recommended value is the square root of machine precision.

Unchanged on exit.

WA - real array of DIMENSION at least (LWA).

Used as workspace.

LWA - INTEGER.

On entry, LWA must specify the dimension of the array WA. $LWA \geq \frac{1}{2}NX(N+13)$.

Unchanged on exit.

IFAIL - INTEGER.

Before entry, IFAIL must be assigned a value. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

Unless the routine detects an error (see next section), IFAIL contains 0 on exit.

6. Error Indicators and Warnings

Errors detected by the routine:-

IFAIL < 0

A negative value of IFAIL indicates an exit from C05PBF because the user has set IFLAG negative in FCN. The value of IFAIL will be the same as the user's setting of IFLAG.

IFAIL = 1

On entry, $N \leq 0$, $LDFJAC < N$, $XTOL < 0.0$ or $LWA < \frac{1}{2}NX(N+13)$.

IFAIL = 2

There have been $100 \times (N+1)$ evaluations of the functions. Consider restarting the calculation from the final point held in X.

IFAIL = 3

No further improvement in the approximate solution X is possible; XTOL is too small.

IFAIL = 4

The iteration is not making good progress. This failure exit may indicate that the system does not have a zero. Otherwise, rerunning C05PBF from a different starting point may avoid the region of difficulty.

7. Auxiliary Routines

Details are distributed to sites in machine-readable form.

8. Timing

The time required by C05PBF to solve a given problem depends on N, the behaviour of the functions, the accuracy requested and the starting point. The number of arithmetic operations executed by C05PBF is about $11.5 \times N^2$ to process each evaluation of the functions and about $1.3 \times N^3$ to process each evaluation of the Jacobian. Unless FCN can be

evaluated quickly, the timing of C05PBF will be strongly influenced by the time spent in FCN.

9. Storage

There are no internally declared arrays.

10. Accuracy

C05PBF tries to ensure that

$$\|X - XSOL\|_2 \leq XTOL \times \|XSOL\|_2.$$

If this condition is satisfied with $XTOL = 10^{-k}$ then the larger components of X have k significant decimal digits. There is a danger that the smaller components of X may have large relative errors, but the fast rate of convergence of C05PBF usually avoids the possibility.

The test assumes that the functions and Jacobian are coded consistently and that the functions are

reasonably well behaved. If these conditions are not satisfied then C05PBF may incorrectly indicate convergence. The coding of the Jacobian can be checked using C05ZAF. If the Jacobian is coded correctly, then the validity of the answer can be checked by rerunning C05PBF with a tighter tolerance.

11. Further Comments

Ideally the problem should be scaled so that, at the solution, the function values are of comparable magnitude.

12. Keywords

Equations, nonlinear algebraic, user-supplied Jacobian, easy-to-use; Powell hybrid method, user-supplied Jacobian, easy-to-use.

13. Example

To determine the values x_1, \dots, x_9 which satisfy the tridiagonal equations:-

$$\begin{aligned} (3 - 2x_1)x_1 - 2x_2 &= -1 \\ -x_{i-1} + (3 - 2x_i)x_i - 2x_{i+1} &= -1, \quad i = 2, 3, \dots, 8, \\ -x_8 + (3 - 2x_9)x_9 &= -1. \end{aligned}$$

13.1. Program Text

```

C      COSPBF EXAMPLE PROGRAM TEXT
C      MARK 9 RELEASE. NAG COPYRIGHT 1981
C      .. LOCAL SCALARS ..
C      REAL FNORM, TOL
C      INTEGER IFAIL, J, NOUT
C      .. LOCAL ARRAYS ..
C      REAL FJAC(9,9), FVEC(9), WA(99), X(9)
C      .. FUNCTION REFERENCES ..
C      REAL F05ABF, SQRT, X02AAF
C      .. SUBROUTINE REFERENCES ..
C      COSPBF
C      ..
C      EXTERNAL FCN
DATA NOUT /6/
WRITE (NOUT,99999)
C      THE FOLLOWING STARTING VALUES PROVIDE A ROUGH SOLUTION.
DO 20 J=1,9
  X(J) = -1.0
20 CONTINUE
TOL = SQRT(X02AAF(0.0))
IFAIL = 0
CALL COSPBF(FCN, 9, X, FVEC, FJAC, 9, TOL, WA, 99, IFAIL)
FNORM = F05ABF(FVEC,9)
WRITE (NOUT,99998) FNORM, IFAIL, (X(J),J=1,9)
STOP
99999 FORMAT (4(1X), 31H C05PBF EXAMPLE PROGRAM RESULTS/1X)
```

```

99998 FORMAT (5X, 31H FINAL L2 NORM OF THE RESIDUALS, E12.4//5X,
* 15H EXIT PARAMETER, I10//5X, 27H FINAL APPROXIMATE SOLUTION//  

* (5X, 3E12.4))  

END  

SUBROUTINE FCN(N, X, FVEC, FJAC, LDFJAC, IFLAG)  

C .. SCALAR ARGUMENTS ..  

INTEGER IFLAG, LDFJAC, N  

C .. ARRAY ARGUMENTS ..  

REAL FJAC(LDFJAC,N), FVEC(N), X(N)  

C ..  

C .. LOCAL SCALARS ..  

REAL FOUR, ONE, TEMP, TEMP1, TEMP2, THREE, TWO, ZERO  

INTEGER J, K  

C ..  

DATA ZERO, ONE, TWO, THREE, FOUR /0.E0,1.E0,2.E0,3.E0,4.E0/  

IF (IFLAG.EQ.2) GO TO 40  

DO 20 K=1,N  

    TEMP = (THREE-TWO*X(K))*X(K)  

    TEMP1 = ZERO  

    IF (K.NE.1) TEMP1 = X(K-1)  

    TEMP2 = ZERO  

    IF (K.NE.N) TEMP2 = X(K+1)  

    FVEC(K) = TEMP - TEMP1 - TWO*TEMP2 + ONE  

20 CONTINUE  

GO TO 100  

40 CONTINUE  

DO 80 K=1,N  

    DO 60 J=1,N  

        FJAC(K,J) = ZERO  

60 CONTINUE  

FJAC(K,K) = THREE - FOUR*X(K)  

IF (K.NE.1) FJAC(K,K-1) = -ONE  

IF (K.NE.N) FJAC(K,K+1) = -TWO  

80 CONTINUE  

100 CONTINUE  

RETURN  

END

```

13.2. Program Data

None.

13.3. Program Results

COSPBF EXAMPLE PROGRAM RESULTS

FINAL L2 NORM OF THE RESIDUALS 0.1193E-07

EXIT PARAMETER 0

FINAL APPROXIMATE SOLUTION

```

-0.5707E+00 -0.6816E+00 -0.7017E+00
-0.7042E+00 -0.7014E+00 -0.6919E+00
-0.6658E+00 -0.5960E+00 -0.4164E+00

```

F04ATF – NAG FORTRAN Library Routine Document

NOTE: before using this routine, please read the appropriate implementation document to check the interpretation of *bold italicised* terms and other implementation-dependent details. The routine name may be precision-dependent.

1. Purpose

F04ATF calculates the accurate solution of a set of real linear equations with a single right hand side, $Ax = b$, by Crout's factorisation method.

2. Specification

```
SUBROUTINE F04ATF (A, IA, B, N, C, AA, IAA, WKS1, WKS2,
  1  IFAIL)
C   INTEGER IA, N, IAA, IFAIL
C   real A(IA,N), B(N), C(N), AA(IAA,N), WKS1(N), WKS2(N)
```

3. Description

Given a set of linear equations, $Ax = b$, the routine first decomposes A using Crout's factorisation with partial pivoting $PA = LU$, where P is a permutation matrix, L is lower triangular and U is unit upper triangular. An approximation to x is found by forward and backward substitution in $Ly = Pb$ and $Ux = y$. The residual vector $r = b - Ax$ is then calculated and a correction, d , to x is found by the solution of $LUd = r$. x is replaced by $(x+d)$ and the process repeated until full machine accuracy is obtained. *Additional precision* accumulation of innerproducts is used throughout the calculation.

4. References

- [1] WILKINSON, J.H. and REINSCH, C.
Handbook for Automatic Computation.
Volume II, Linear Algebra.
Springer-Verlag, 1971, pp. 93–110.

5. Parameters

A – *real* array of DIMENSION (IA,p) where $p \geq N$.

Before entry, A must contain the elements of the real matrix.

Unchanged on exit.

IA – INTEGER.

On entry, IA must specify the first dimension of array A as declared in the calling (sub)program.

$IA \geq N$.

Unchanged on exit.

B – *real* array of DIMENSION at least (N).

Before entry, B must contain the elements of the right hand side. (See Section 11).

Unchanged on exit.

N – INTEGER.

On entry, N must specify the order of matrix A.

Unchanged on exit.

C – *real* array of DIMENSION at least (N).

On successful exit, C will contain the solution vector.

AA – *real* array of DIMENSION (IAA,q) where $q \geq N$.

Used as working space.

On successful exit, AA will contain the LU decomposition.

IAA – INTEGER.

On entry, IAA must specify the first dimension of array AA as declared in the calling (sub)program.

$IAA \geq N$.

Unchanged on exit.

WKS1 – *real* array of DIMENSION at least (N).

WKS2 – *real* array of DIMENSION at least (N).

Used as working space.

IFAIL – INTEGER.

On entry, IFAIL must be set to 0 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

Unless the routine detects an error (see next section), IFAIL contains 0 on exit.

6. Error Indicators and Warnings

Errors detected by the routine:-

IFAIL = 1

The matrix A is singular, possibly due to rounding errors.

IFAIL = 2

The matrix A is too ill-conditioned to produce a correctly rounded solution.

7. Auxiliary Routines

Details are distributed to sites in machine-readable form.

8. Timing

The time taken is approximately proportional to N^3 .

9. Storage

There are no internally declared arrays.

10. Accuracy

The computed solutions should be correct to full machine accuracy. For a detailed error analysis see [1], page 107.

11. Further Comments

The routine must not be called with the same name for parameters B and C.

12. Keywords

Accurate Solution of Linear Equations,
Crout Factorisation,
Real Matrix,
Single Right Hand Side.

13. Example

To solve the set of linear equations $Ax = b$ where

$$A = \begin{bmatrix} 33 & 16 & 72 \\ -24 & -10 & -57 \\ -8 & -4 & -17 \end{bmatrix} \text{ and } b = \begin{bmatrix} -359 \\ 281 \\ 85 \end{bmatrix}$$

13.1. Program Text

WARNING: This single precision example program may require amendment for certain implementations. The results produced may not be the same. If in doubt, please seek further advice (see Essential Introduction to the Library Manual).

```
C      F04ATF EXAMPLE PROGRAM TEXT
C      NAG COPYRIGHT 1975
C      MARK 4.5 REVISED
C
REAL A(5,5), B(5), C(5), AA(5,5), WKS1(18), WKS2(3)
INTEGER NIN, NOUT, I, N, J, IA, IAA, IFAIL
DATA NIN /5/, NOUT /6/
READ (NIN,99999) (WKS1(I),I=1,7)
WRITE (NOUT,99997) (WKS1(I),I=1,6)
N = 3
READ (NIN,99998) ((A(I,J),J=1,N),I=1,N), (B(I),I=1,N)
IA = 5
IAA = 5
IFAIL = 1
CALL F04ATF(A, IA, B, N, C, AA, IAA, WKS1, WKS2, IFAIL)
IF (IFAIL.EQ.0) GO TO 20
WRITE (NOUT,99996) IFAIL
STOP
20 WRITE (NOUT,99995) (C(I),I=1,N)
STOP
99999 FORMAT (6A4, 1A3)
```

```
99998 FORMAT (3F5.0)
99997 FORMAT (4(1X/), 1H , 5A4, 1A3, 7HRESULTS/1X)
99996 FORMAT (25H0ERROR IN F04ATF IFAIL = , I2)
99995 FORMAT (10HOSOLUTIONS/(1H , F4.1))
END
```

13.2. Program Data

F04ATF EXAMPLE PROGRAM DATA

```
 33   16   72
 -24  -10  -57
  -8   -4   -17
 -359  281   85
```

13.3. Program Results

F04ATF EXAMPLE PROGRAM RESULTS

SOLUTIONS

```
 1.0
 -2.0
 -5.0
```

F03AAF - NAG FORTRAN Library Routine Document

NOTE: before using this routine, please read the appropriate implementation document to check the interpretation of *bold italicised* terms and other implementation-dependent details. The routine name may be precision-dependent.

1. Purpose

F03AAF calculates the determinant of a real matrix using the factorisation method of Crout.

2. Specification

```
SUBROUTINE F03AAF (A, IA, N, DET, WKSPCE, IFAIL)
C      INTEGER IA, N, IFAIL
C      real A(IA,N), DET, WKSPCE(N)
```

3. Description

The routine calculates the determinant of A using the Crout factorisation with partial pivoting, $PA = LU$, where P is a permutation matrix, L is lower triangular and U is unit upper triangular. The determinant of A is the product of the diagonal elements of L with the correct sign determined by the row interchanges. *Additional precision* accumulation of inner-products is used throughout.

4. References

- [1] WILKINSON, J.H. and REINSCH, C.
Handbook for Automatic Computation, Volume II, Linear Algebra, pp. 93–110.
 Springer-Verlag, 1971.

5. Parameters

A – *real* array of DIMENSION (IA,p), where $p \geq N$.

Before entry, A must contain the elements of the real matrix.

On exit, it will contain the Crout factorisation, with the unit diagonal of U understood, unless an error has occurred.

IA – INTEGER.

On entry, IA must specify the first dimension of array A as declared in the calling (sub)program.

IA $\geq N$

Unchanged on exit.

N – INTEGER.

On entry, N must specify the order of the matrix.

Unchanged on exit.

DET – *real*.

On exit, DET will contain the value of the determinant, unless an error has occurred (See Section 6).

WKSPCE – *real* array of DIMENSION at least (N).

Used as working space.

IFAIL - INTEGER.

Before entry, IFAIL must be set to 0 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details. Unless the routine detects an error or gives a warning (see Section 6), IFAIL contains 0 on exit.

For this routine, because the values of output parameters may be useful even if IFAIL ≠ 0 on exit, users are recommended to set IFAIL to 1 before entry. It is then essential to test the value of IFAIL on exit.

6. Error Indicators and Warnings

Errors detected by the routine:-

IFAIL = 1

The matrix A is singular, possibly due to rounding errors. DET is set to 0.0.

IFAIL = 2

Overflow. The value of the determinant is too large to be held in the computer.

IFAIL = 3

Underflow. The value of the determinant is too small to be held in the computer.

7. Auxiliary Routines

Details are distributed to sites in machine-readable form.

8. Timing

The time taken is approximately proportional to N^3 .

9. Storage

There are no internally declared arrays.

10. Accuracy

The accuracy of the determinant depends on the conditioning of the original matrix. For a detailed error analysis see [1], p. 107.

11. Further Comments

None.

12. Keywords

Crout Factorisation,
Determinant,
Real Matrix.

13. Example

To calculate the determinant of the real matrix:

$$\begin{pmatrix} 33 & 16 & 72 \\ -24 & -10 & -57 \\ -8 & -4 & -17 \end{pmatrix}$$

13.1. Program Text

WARNING: This single precision example program may require amendment for certain implementations. The results produced may not be the same. If in doubt, please seek further advice (see Essential Introduction to the Library Manual).

```
C      F03AAF EXAMPLE PROGRAM TEXT
C      NAG COPYRIGHT 1975
C      MARK 4.5 REVISED
C
      REAL DETERM, A(4,4), WKSPCE(18)
      INTEGER NIN, NOUT, I, N, J, IA, IFAIL
      DATA NIN /5/, NOUT /6/
      READ (NIN,99999) (WKSPCE(I),I=1,7)
      WRITE (NOUT,99997) (WKSPCE(I),I=1,6)
      N = 3
      READ (NIN,99998) ((A(I,J),J=1,N),I=1,N)
      IA = 4
      IFAIL = 1
      CALL F03AAF(A, IA, N, DETERM, WKSPCE, IFAIL)
      IF (IFAIL.EQ.0) GO TO 20
      WRITE (NOUT,99996) IFAIL
      STOP
20   WRITE (NOUT,99995) DETERM
      STOP
99999 FORMAT (6A4, 1A3)
99998 FORMAT (3F5.0)
99997 FORMAT (4(1X/), 1H , 5A4, 1A3, 7HRESULTS/1X)
99996 FORMAT (25HOERROR IN F03AAF IFAIL = , I2)
99995 FORMAT (24HOVALUE OF DETERMINANT = , F4.1)
      END
```

13.2. Program Data

```
F03AAF EXAMPLE PROGRAM DATA
      33    16    72
     -24   -10   -57
      -8    -4   -17
```

13.3. Program Results

F03AAF EXAMPLE PROGRAM RESULTS

VALUE OF DETERMINANT = 6.0

Chapter XI

Listing of Program RCNATOUT FORTRAN

```

C RCNATOUT RCN00010
C PROGRAM TO EVALUATE TRANSCENDENTAL EQUATION FOR A RANGE OF RCN00020
C FREQUENCIES IN ORDER TO DETERMINE APPROXIMATE VALUES OF THE NATURAL RCN00030
C FREQUENCIES OF COMPLIANT RISERS FOR OUT OF PLANE DYNAMICS WITH A 2-D RCN00040
C STATIC CONFIGURATION. RCN00050
C THIS PROGRAM CAN BE USED TO DETERMINE APPROXIMATE ESTIMATES OF NATURALRCN00060
C FREQUENCIES TO USE AS INPUT IN THE OUT OF PLANE DYNAMICS PROGRAM, RCN00070
C WHICH IMPROVES THESE INITIAL ESTIMATES. RCN00080
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED THROUGHOUT. RCN00090
C *****RCN00100
C*****RCN00110
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY RCN00120
C ALL RIGHTS RESERVED. *****RCN00130
C*****RCN00140
C PROGRAMMER GEORGE A. KRIEZIS JUNE 3, 1985 M.I.T. *****RCN00150
C*****RCN00160
C
C DEFINITION OF DEVICES: RCN00170
C DEVICE 5 : INPUT FROM TERMINAL RCN00180
C DEVICE 6 : OUTPUT TO TERMINAL RCN00190
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT (LRECL=80) RCN00200
C DEVICE 10 : INPUT FROM FILE CONTAINING APPROXIMATE N-D STATIC RCN00210
C SOLUTION CREATED BY RCSTAT2D (LRECL=132) RCN00220
C DEVICE 11 : OUTPUT TO FILE (LRECL=80) RCN00230
C
C DEFINITIONS OF PARAMETERS RCN00240
C MNP = MAXIMUM NUMBER OF FREQUENCIES TO EVALUATE FUNCTION. RCN00250
C
C IMPLICIT REAL*8(A-H,O-Z) RCN00260
C PARAMETER(MNP=300) RCN00270
C CHARACTER*80 NAME RCN00280
C COMMON/INPUT0/NAME RCN00290
C DIMENSION FREQ(MNP),FVALUE(MNP),RANG1(MNP),RANG2(MNP) RCN00300
C
C READ2D READS 2-D STATIC COMPLIANT RISER RCN00310
C SOLUTION CALCULATED FROM RCSTAT2D PROGRAM AND EVALUATES RCN00320
C THE MAXIMUM NONDIMENSIONAL STATIC EFFECTIVE TENSION (TOMAX) RCN00330
C
C CALL READ2D(TOMAX,VM,ICCC) RCN00340
C IF(ICCC.EQ.0) STOP RCN00350
C
C CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE RCN00360
C NONDIMENSIONAL COEFFICIENTS TO BE USED FOR THE ESTIMATION OF RCN00370
C THE NATURAL FREQUENCIES. RCN00380
C
C CALL CHARAC(TOMAX,TMAXAV,TLENG) RCN00390
C
C
C
1499 WRITE(6,1500) MNP RCN00400
1500 FORMAT(' INPUT'// THE MAXIMUM FREQUENCY AND FREQUENCY SPACING (INRCN00500
* RAD/S) // FOR WHICH YOU WISH TO EVALUATE THE FREQUENCY FUNCTION'/RCN00510
*' MAXIMUM PERMISSIBLE NUMBER OF FREQUENCY POINTS IS = ',I3) RCN00520
READ(5,*) FREQH,DF RCN00530
DIV = FREQH/DF RCN00540
XP = FLOAT(MNP-1) RCN00550
IF(DIV.GT.XP) GOTO 1499 RCN00560

```

```

C                               RCN00570
C APPROX : EVALUATES THE TRANSCENDENTAL EQUATION FOR ALL FREQUENCIESRCN00580
C IN THE RANGE REQUESTED.                                                 RCN00590
C                                                               RCN00600
C CALL APPROX(FREQ,FVALUE,TLENG,TMAXAV,TOMAX,FREQH,DF,NDIV)          RCN00610
C                                                               RCN00620
C DETERMINE THE APPROXIMATE VALUES OF EACH NATURAL FREQUENCY IN THE RCN00630
C SPECIFIED RANGE. THE EQUATION CHANGES SIGN AT EACH NATURAL           RCN00640
C FREQUENCY.                                                 RCN00650
C                                                               RCN00660
C K = 0                                         RCN00670
DO 452 I=1,NDIV-1                         RCN00680
  CHECK = FVALUE(I)*FVALUE(I+1)             RCN00690
  IF (CHECK.LT.0.D0) THEN                  RCN00700
    K = K + 1                           RCN00710
    RANG1(K) = FREQ(I)                   RCN00720
    RANG2(K) = FREQ(I+1)                 RCN00730
  END IF                                 RCN00740
452  CONTINUE                                RCN00750
C                               RCN00760
C OUTPUT THE RESULTS                      RCN00770
C                               RCN00780
      WRITE(11,8866) NAME                RCN00790
8866 FORMAT(80A)                           RCN00800
      WRITE(11,66)                         RCN00810
66   FORMAT(/' ESTIMATES OF OUT-OF PLANE NATURAL FREQUENCIES FOR A 2-D RCN00820
  *STATIC CONFIGURATION')               RCN00830
      WRITE(11,85)K                         RCN00840
85   FORMAT(' NUMBER OF NATURAL FREQUENCIES IN SPECIFIED RANGE: ',I3) RCN00850
                                              RCN00860
      DO 76 I=1,K                         RCN00870
        WRITE(11,86) I,RANG1(I),RANG2(I)   RCN00880
76   CONTINUE                                RCN00890
86   FORMAT(' MODE = ',I3/' NATURAL FREQUENCY BETWEEN'/1X,D12.6,' AND 'RCN00900
  *,D12.6,' RAD/SEC'/' **** * * * * * * * * * * * * * * * * * * * * * ') RCN00910
      WRITE(11,24)                          RCN00920
24   FORMAT(/' FREQUENCY (RAD/SEC)',2X,'EQUATION VALUE')            RCN00930
      DO 89 I=1,NDIV                     RCN00940
        WRITE(11,25)FREQ(I),FVALUE(I)     RCN00950
89   CONTINUE                                RCN00960
25   FORMAT(5X,D12.6,4X,D12.6)              RCN00970
      STOP                                  RCN00980
      END                                    RCN00990
                                              RCN01000
      SUBROUTINE CHARAC(TOMAX,TMAXAV,TLENG)          RCN01010
C                               RCN01020
C SUBROUTINE CHARAC READS THE COMPLIANT RISER CHARACTERISTICS FROM       RCN01030
C DEVICE 8 AND EVALUATES ALL THE NONDIMENSIONAL COEFFICIENTS USED       RCN01040
C IN THE EVALUATION OF THE TRANSCENDENTAL EQUATION AND THE             RCN01050
C NONDIMENSIONALIZED FREQUENCIES.
      IMPLICIT REAL*8(A-H,O-Z)            RCN01060
      PARAMETER(MNP=151)                  RCN01070
      CHARACTER*80 NAME                  RCN01080
      COMMON/INPUT0/NAME                RCN01090
      COMMON/COEF/EPSXI(2),RETAM        RCN01100
      DIMENSION EIXI(MNP),RLENG(MNP),AMAETA(MNP),TMAETA(MNP)          RCN01110
      DIMENSION RMASS(MNP),AMAXI(MNP),TMAXI(MNP)                      RCN01120

```

```

C RCN01130
READ (8,1000) NAME
1000 FORMAT(80A) RCN01140
READ (8,1008) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS RCN01150
1008 FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6)) RCN01160
TLEN = TLEN RCN01170
DO 1502 I=1,NSEG RCN01180
READ (8,1003) RLENG(I),RMASS(I),RMASST,AMAXI(I),WEIGHT,DXI,
*PXIETA,EA,EIETA,AC,EIETAS RCN01190
1502 CONTINUE RCN01200
1003 FORMAT(5(1X,D12.6)/6(1X,D12.6)) RCN01210
C RCN01220
DO 1332 I=1,NSEG RCN01230
READ(8,1333) AMAETA(I),DETA,EIXI(I),EIXIS,GIP,GIPS,AMAZI,XJ,ZI RCN01240
1332 CONTINUE RCN01250
1333 FORMAT(6(1X,D12.6)/3(1X,D12.6)) RCN01260
C RCN01270
C RCN01280
C NON - DIMENSIONALIZATIONS RCN01290
C RCN01300
C RCN01310
C RCN01320
C RCN01330
GRAV=9.81D0 RCN01340
WAM=WA/GRAV RCN01350
WT=WA*TLEN RCN01360
TOMAX = TOMAX*WT RCN01370
C RCN01380
C NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS RCN01390
C RCN01400
TOMAXL = TOMAX/TLEN RCN01410
TOML1 = TOMAX*TLEN RCN01420
TOML2 = TOML1*TLEN RCN01430
TLEN2 = TLEN**2 RCN01440
TMAXAV = 0.D0 RCN01450
DO 2000 I=1,NSEG RCN01460
    RLENG(I)=RLENG(I)/TLEN RCN01470
    TMAXI(I) = RMASS(I) + AMAXI(I) RCN01480
    TMAETA(I) = RMASS(I) + AMAETA(I) RCN01490
    TMAXAV = TMAXI(I)*RLENG(I) + TMAXAV RCN01500
2000 CONTINUE RCN01510
C RCN01520
EPSXI(1) = EIXI(1)/TOML2 RCN01530
EPSXI(2) = EIXI(NSEG)/TOML2 RCN01540
C RCN01550
HETAM = 0.DC RCN01560
DO 4321 I=1,NSEG RCN01570
    HETA = TMAETA(I)/TMAXAV RCN01580
    HETAM = HETAM + HETA*RLENG(I) RCN01590
4321 CONTINUE RCN01600
RETURN RCN01610
END RCN01620
C RCN01630
SUBROUTINE READ2D(TOMAX,VM,ICCC)
C THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM RCN01640
C DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION. RCN01650
C IMPLICIT REAL*8(A-H,O-Z) RCN01660
PARAMETER(MNP=151) RCN01670
DIMENSION STATIC(MNP),XI(MNP) RCN01680

```

```

C          RCN01690
C      STATIC(I) = STATIC EFFECTIVE TENSION TO           RCN01700
C          RCN01710
C      WRITE(6,2000) MNP           RCN01720
2000  FORMAT(' MNP=',I3)           RCN01730
C          RCN01740
C      ICCC = 1           RCN01750
      READ(10,36459) NP,VM           RCN01760
36459  FORMAT(1X,I3,1X,D12.6)           RCN01770
      WRITE(6,2311) NP,VM           RCN01780
2311  FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
*' NP =',I3/
*' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S ,VM =',D12.6)   RCN01790
C          RCN01800
C      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN           RCN01810
      ICCC=0           RCN01820
      WRITE(6,12439)           RCN01830
12439  FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS')   RCN01840
      RETURN           RCN01850
      ENDIF           RCN01860
C          RCN01870
C      IF (VM.EQ.0.0) THEN           RCN01880
      WRITE(6,124)           RCN01890
124  FORMAT(' PROGRAM IS NOT VALID FOR ZERO MEAN CURRENT SPEED'/
*' RUN STOPS')           RCN01900
      ICCC = 0           RCN01910
      RETURN           RCN01920
      END IF           RCN01930
C          RCN01940
C      READING FROM DEVICE 10           RCN01950
C          RCN01960
C      DO 1021 I=1,NP           RCN01970
      READ(10,1033) X,STATIC(I),STATI2,STATI3,STAT4,XCOOR,YCOOR,
*STRARC,TENSI,VLOCKXI           RCN01980
      XI(I) = X           RCN01990
1021  CONTINUE           RCN02000
1033  FORMAT(10(1X,D12.6))           RCN02010
C      EVALUATE MAXIMUM STATIC EFFECTIVE TENSION           RCN02020
      TOMAX=DMAX1(STATIC(1),STATIC(2))           RCN02030
      DO 9859 I=3,NP           RCN02040
      TOMAX = DMAX1(TOMAX,STATIC(I))           RCN02050
9859  CONTINUE           RCN02060
C          RCN02070
C      WRITE(6,1654) TOMAX           RCN02080
1654  FORMAT(' 2-D STATIC SOLUTION SUCCESFULLY READ'/' MAXIMUM STATIC EFRCN02130
*FECTIVE TENSION/WA*L = ',D10.4)           RCN02140
C          RCN02150
C      RETURN           RCN02160
C      END           RCN02170
C          RCN02180
C      SUBROUTINE APPROX(FREQ,FVALUE,TLENG,TMAXAV,TOMAX,FREQH,DF,NDIV)   RCN02190
C      THIS SUBROUTINE EVALUATES THE TRANSCENDENTAL EQUATION AT ALL THE RCN02200
C      SPECIFIED FREQUENCY POINTS IN ORDER TO LOCATE REGIONS OF CHANGING RCN02210
C      SIGN.           RCN02220
      IMPLICIT REAL*8(A-H,O-Z)           RCN02230
      PARAMETER(MNP=300)           RCN02240

```

```

COMMON/COEF/EPSXI(2),HETAM RCN02250
DIMENSION FVALUE(MNP),FREQ(MNP) RCN02260
C EVALUATE NU. IT IS EXACT IF THE STRAIN RELIEF UNITS ARE THE SAME RCN02270
C AT THE TWO ENDS OF THE RISER. RCN02280
XNU = DSQRT((EPSXI(1)+EPSXI(2))/2.D0) RCN02290
WRITE(6,581) XNU RCN02300
581 FORMAT(' THE VALUE OF NU TO BE USED IS = ',D10.4/' IF YOU WANT TO RCN02310
*CHANGE NU INPUT 1') RCN02320
READ(5,*) INU RCN02330
IF (INU.EQ.1) THEN RCN02340
  WRITE(6,*) ' INPUT NU' RCN02350
  READ(5,*) XNU RCN02360
END IF RCN02370
RCN02380
C EVALUATE THE TRANSCENDENTAL EQUATION FOR EACH FREQUENCY. RCN02390
C EVALUATE NUMBER OF INCREMENTS. RCN02400
C NDIV = NINT(FREQH/DF)+ 1 RCN02410
C INITIAL FREQUENCY FOR EVALUATION IS 0.01 RAD/SEC RCN02420
SIGMA = 0.01D0*TLENG*DSQRT(TMAXAV*HETAM/TOMAX) RCN02430
DS = DF*TLENG*DSQRT(TMAXAV*HETAM/TOMAX) RCN02440
DO 765 L=1,NDIV RCN02450
  FREQ(L) = .01 + (L-1)*DF RCN02460
C EVALUATE TRANSCENDENTAL EQUATION RCN02470
RCN02480
C FV = (1.D0-SIGMA**2*XNU**2)*DSIN(SIGMA)-2.D0*SIGMA*XNU*DCOS(SIGMA) RCN02490
SIGMA = SIGMA + DS RCN02500
FVALUE(L) = FV RCN02510
765 CONTINUE RCN02520
RETURN RCN02530
END RCN02540

```

Chapter XII

Listing of Program RCLINDYN FORTRAN A

```

C RCLINDYN                                     RCL00010
C THIS PROGRAM CALCULATES THE OUT OF PLANE LINEAR DYNAMIC RESPONSE OF A RCL00020
C COMPLIANT RISER WITH A 2-D STATIC CONFIGURATION BASED ON AN EMBEDDING RCL00030
C TECHNIQUE. THE PROGRAM PROVIDES AN INITIAL ANALYTIC APPROXIMATION RCL00040
C WHICH IT SUBSEQUENTLY IMPROVES USING MODIFIED NEWTON'S ITERATION AND RCL00050
C A NON-UNIFORM GRID FINITE DIFFERENCE METHOD. RCL00060
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED. RCL00070
C*****RCL00080
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY RCL00090
C ALL RIGHTS RESERVED. RCL00100
C*****RCL00110
C PROGRAMMER GEORGE A. KRIEZIS      JUNE 4, 1985      M.I.T. RCL00120
C*****RCL00130
C*****RCL00140
C
C DEFINITION OF DEVICES:
C DEVICE 5 : INPUT FROM TERMINAL RCL00150
C DEVICE 6 : OUTPUT TO TERMINAL RCL00160
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT (LRECL=80) RCL00170
C DEVICE 9 : COMPLETE OUTPUT TO FILE (LRECL=132) RCL00180
C DEVICE 10 : INPUT FROM FILE CONTAINING N-D 2-D STATIC RCL00190
C           SOLUTION CREATED BY RCSTAT2D (LRECL=132) RCL00200
C DEVICE 11 : OUTPUT TO FILE FOR PLOTS OR INPUT TO A NEW RUN RCL00210
C           OF RCLINDYN (LRECL=117) RCL00220
C DEVICE 12 : INPUT FROM FILE CONTAINING APPROXIMATE N-D SOLUTION RCL00230
C           CREATED BY A PREVIOUS RUN OF RCLINDYN (LRECL=117) RCL00240
C
C COMMON BLOCK CONTENTS (OVERALL REFERENCE):
C SOLUT = INITIAL APPROXIMATION AND SOLUTION MATRICES RCL00250
C STAT   = STATIC COMPLIANT RISER SOLUTION. (EFFECTIVE TENSION, SHEAR RCL00260
C           FORCE, OMEGA ETA, PHI0, MAXIMUM STATIC EFFECTIVE TENSION, RCL00270
C           MEAN CURRENT VELOCITY AND NUMBER OF STATIC DIVISION POINTS) RCL00280
C STAT1  = FUNCTIONS OF STATIC RESULTS. RCL00290
C INPUT0 = OUTPUT FILE HEADING RCL00300
C INPUT1 = RISER LENGTH, AVERAGE AND TOTAL WEIGHT AND NUMBER OF SEGMENTS RCL00310
C INPUT2 = RISER TUBE INTERIOR CHARACTERISTICS RCL00320
C INPUTL = RISER SEGMENTS LENGTH RCL00330
C INPUT3 = WEIGHT, STIFFNESSES AND STIFFNESS DERIVATIVES. RCL00340
C INPUT4 = CHARACTERISTIC DIAMETERS AND PERIMETER OF RISER TUBE. RCL00350
C INPUT5 = MASS AND ADDED MASS CHARACTERISTICS IN THREE DIRECTIONS RCL00360
C INPUT6 = RISER POLAR AND ADDED POLAR MOMENT OF INERTIA. RCL00370
C COEF   = NONDIMENSIONAL RISER CHARACTERISTICS USED IN THE EQUATIONS RCL00380
C COEF1  = NONDIMENSIONAL RISER CHARACTERISTICS AT A SPECIFIC POINT RCL00390
C CONST   = PROBLEM CONSTANTS (PI, GRAVITY, WATER DENSITY) RCL00400
C BOUNDA = BOUNDARY CONDITION FOR OMEGA XI AT S=0 RCL00410
C COUN   = INTEGER COUNTING VARIABLE TO DETERMINE EACH DIVISION POINT RCL00420
C DEF    = NU USED TO DETERMINE THE APPROXIMATE NATURAL FREQUENCY, RCL00430
C           NU = 1/SQRT(EPSETA) RCL00440
C
C IMPLICIT REAL*8(A-H,O-Z) RCL00450
C PARAMETER(N=7,MNP=151,NA=4,IY=7) RCL00460
C PARAMETER(LWORK=MNP*(3*N**2+6*N+2)+4*N**2+3*N) RCL00470
C PARAMETER(LIWORK=MNP*(2*N+1)+N) RCL00480
C DIMENSION WORK(LWORK), IWORK(LIWORK) RCL00490
C COMMON/SOLUT/X(MNP), Y(N,MNP), ABT(N), MODE RCL00500
C COMMON/STAT/XI(MNP), STATIC(NA,MNP), VLOCXI(MNP), TOMAX, VM, NPI RCL00510
C COMMON/COUN/ICOUNT RCL00520

```

```

C      EXTERNAL SUBROUTINES USED BY NAG LIBRARY          RCL00570
C      EXTERNAL FCN,G,JACEPS,JACGEP,JACOBF,JACOBG        RCL00580
C      DATA NOUT /6/                                     RCL00590
C                                              RCL00600
C      READ2D READS 2-D STATIC SOLUTION CALCULATED FROM RCSTAT2D PROGRAM. RCL00610
C      IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION (TOMAX) RCL00620
C                                              RCL00630
C      CALL READ2D(NPI)                                RCL00640
C                                              RCL00650
C      CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE RCL00660
C      NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE EQUATIONS. RCL00670
C                                              RCL00680
C      CALL CHARAC(TLENG,IC,NPI)                         RCL00690
C      IF(IC.EQ.0) STOP                                RCL00700
C                                              RCL00710
C      DEFINITIONS OF PARAMETERS ...
C      N = NUMBER OF EQUATIONS TO BE SOLVED BY DO2RAF    RCL00720
C      IY = NUMBER OF VARIABLES                         RCL00730
C      MNP= MAXIMUM NUMBER OF PERMITTED POINTS IN THE NON-UNIFORM RCL00740
C      F.D. MESH ( MNP >= 32)                          RCL00750
C      IF(MNP.LT.32) THEN                               RCL00760
C      WRITE(6,1257) MNP                                RCL00770
1257  FORMAT(' MNP = ',I3,' IS NOT PERMISSIBLE')      RCL00780
      STOP                                              RCL00790
      ENDIF                                             RCL00800
C      NUMBER OF BOUNDARY CONDITIONS AT S=0            RCL00810
1276  NUMBEG=4                                         RCL00820
C      NUMBER OF MIXED BOUNDARY CONDITIONS             RCL00830
      NUMMIX=0                                         RCL00840
C      PARAMETER DECLARING THAT INITIAL APPROXIMATION IS USER SPECIFIED RCL00850
      INIT=1                                           RCL00860
C      PARAMETER DECLARING THAT JACOBIANS ARE USER SPECIFIED RCL00870
      IJAC=1                                           RCL00880
C      PARAMETER CONTROLLING MONITORING OF CALCULATIONS RCL00890
      IFAIL=111                                         RCL00900
C                                              RCL00910
C                                              RCL00920
C                                              RCL00930
1499  WRITE(6,1500)                                    RCL00940
1500  FORMAT(' INPUT 1 IF YOU WISH TO READ INITIAL APPROXIMATION,/',
*'           CREATED BY A PREVIOUS RUN OF THIS PROGRAM'/
*' ELSE INPUT 0 (AN INITIAL APPROXIMATION WILL BE CALCULATED LOCALLY)') RCL00950
      RCL00960
      RCL00970
      RCL00980
      READ(5,*) IEXIST                                RCL00990
      IF((IEXIST.NE.1).AND.(IEXIST.NE.0)) GOTO 1499
      IF(IEXIST.EQ.1) THEN                            RCL01000
      RCL01010
C      READAS : READS APPROXIMATE SOLUTION FROM DEVICE 10 FROM A RCL01020
C      PREVIOUS RUN OF RCLINDYN (DELEPS = 1.0D0)          RCL01030
C      IT ALSO PROVIDES ICCC,NP, TOL= TOLERANCE OF ITERATIONS RCL01040
C      THIS ALTERNATIVE MAY, FOR EXAMPLE, BE USED IF GREATER ACCURACY RCL01050
C      IS REQUIRED FOR THE PROBLEM SOLUTION.            RCL01060
C                                              RCL01070
C                                              RCL01080
C      CALL READAS(ICCC,NP,TOL)                         RCL01090
C                                              RCL01100
C      IF THE NUMBER OF POINTS FROM THIS SOLUTION IS GREATER THAN THE RCL01110
C      STATIC SOLUTION POINTS, INTERPOLATE ALL CHARACTERISTICS TO THE RCL01120

```

```

C      NEW NUMBER OF POINTS.                               RCL01130
C      ASSUMPTION : NP ALWAYS GREATER OR EQUAL TO NPI    RCL01140
C      THIS IS VALID IF THE SAME STATIC SOLUTION WAS USED TO OBTAIN RCL01150
C      THE APPROXIMATE SOLUTION IN DEVICE 10.             RCL01160
C
C      IF (NP.GT.NPI) THEN                                RCL01170
C          CALL INTERP(NP)                                RCL01180
C          NPI=NP                                         RCL01190
C          ELSE IF (NP.LT.NPI) THEN                      RCL01200
C              WRITE(6,314)                                RCL01210
314     FORMAT(' APPROXIMATE SOLUTION IN DEVICE 10 WAS NOT OBTAINED'/
*           ' FROM THE SAME STATIC SOLUTION, PROGRAM STOPS') RCL01220
C              STOP                                         RCL01230
C              END IF                                       RCL01240
C              DELEPS = 1.D0                                RCL01250
C
C      ELSE                                              RCL01260
C
C      APPROX : PROVIDES INITIAL APPROXIMATE SOLUTION FOR EPS=0. RCL01270
C                  IT ALSO PROVIDES ICCC,NP, TOL.            RCL01280
C
C      CALL APPROX(TLENG,ICCC,NPI,TOL)                   RCL01290
C      NP = NPI                                         RCL01300
C
C      INCREMENT OF CONTINUATION PARAMETER               RCL01310
7891  WRITE(6,7890)                                     RCL01320
7890  FORMAT(' INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS') RCL01330
*/* IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.D0'/ RCL01340
*/* IF CONTINUATION IS REQUIRED THEN 0.D0 < DELEPS < 1.D0'/ RCL01350
*/* RECOMMENDATION :'/ RCL01360
*/* USUALLY DELEPS = 0.1D0 WILL SUFFICE'/ RCL01370
*/* FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM'/ RCL01380
*/* A SMALLER VALUE OF DELEPS, E.G. 0.05D0 OR 0.025D0 MIGHT BE'/ RCL01390
*/* NECESSARY')
      READ(5,*) DELEPS
      IF((DELEPS.GT.1.D0).OR.(DELEPS.LE.0.D0)) THEN
          WRITE(6,7892)
7892  FORMAT(' 0. < DELEPS <= 1.')
      GOTO 7891
      ENDIF
C
C      ENDIF
C      SET COUNTING VARIABLE
C      ICOUNT = 1
C      IF(ICCC.EQ.0) STOP
C
C      CALL X04AAF(1,NOUT)
C      CALL X04ABF(1,NOUT)
C
C      CALL NAG SUBROUTINE DG2RAF TO SOLVE THE TWO-POINT BOUNDARY VALUE RCL01590
C      PROBLEM USING A DEFERRED CORRECTION TECHNIQUE AND NEWTON'S      RCL01600
C      ITERATION.                                                 RCL01610
C
C      CALL DG2RAF(N,MNP,NP,NUMBEG,NUMMIX,TOL,INIT,X,Y,IY,ABT,FCN,G,IJAC,RCL01620
C      *JACOBF,JACOBG,DELEPS,JACEPS,JACGEP,WORK,LWORK,IWORK,LIWORK,IFAIL) RCL01630
C
C      RCL01640
C
C      RCL01650
C
C      RCL01660
C
C      RCL01670
C
C      RCL01680

```

FILE: RCLINDYN FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

      WRITE(6,9000) IFAIL
9000  FORMAT(' IFAIL =',I3)
C
C     OUTPUT THE RESULTS
C
      IF((IFAIL.EQ.0).OR.(IFAIL.EQ.4)) THEN
         CALL OUTPUT(NP)
      ENDIF
C
      STOP
END

SUBROUTINE CHARAC(TLENG,IC,NP)
C THIS SUBROUTINE READS THE COMPLIANT RISER CHARACTERISTICS AND
C EVALUATES THE NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE
C GOVERNING EQUATIONS.,
IMPLICIT REAL*8(A-H,O-Z)
PARAMETER(MNP=151,NA=4,N=7)
CHARACTER*80 NAME
COMMON/CONST/XPI,XPI2,RHOW,GRAV
COMMON/INPUT0/NAME
COMMON/INPUT1/TLEN,WA,WT,NSEG
COMMON/INPUT2/RHO0,AI,CFLUID,PRESS,AO(MNP)
COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)
COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP)
*,EIXIS(MNP),GIP(MNP),GIPS(MNP)
COMMON/INPUT4/DXI(MNP),PXEITA(MNP),DETA(MNP),DXIETA(MNP)
COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP),
TMAXI(MNP),TMAETA(MNP)
COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP)
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCXI(MNP),TOMAX,VM,NPI
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL02000
*NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV
COMMON/STAT1/CONST1(MNP),CONST2(MNP)
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
DIMENSION TENO(MNP),QXO(MNP),TOS(MNP),QXIOS(MNP)

C
C     READ RISER CHARACTERISTICS FROM DEVICE 8
READ (8,1000) NAME
1000 FORMAT(80A)
READ (8,1008) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS
1008 FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6))
      TLENG = TLEN
      DO 1502 I=1,NSEG
      READ (8,1003) RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXI(IRCL02130
*),PXEITA(I),EA(I),EIETA(I),AO(I),EIETAS(I)
1502 CONTINUE
1003 FORMAT(5(1X,D12.6)/6(1X,D12.6))
C
      DO 1332 I=1,NSEG
      READ(8,1333) AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),AMARCL02190
*ZI(I),XJZI(I),AJZI(I)
1332 CONTINUE
1333 FORMAT(6(1X,D12.6)/3(1X,D12.6))
C
      IC=1

```

```

C
7650  WRITE(6,7651)                                     RCL02250
7651  FORMAT(' DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS'/
*' IF YES INPUT 1 , IF NO INPUT 0')                   RCL02260
      READ(5,*) IPRINT                                 RCL02270
      IF((IPRINT.NE.1).AND.(IPRINT.NE.0)) GOTO 7650   RCL02280
      IF(IPRINT.EQ.1) THEN                           RCL02290
C
      WRITE(6,1000) NAME                            RCL02300
      WRITE(6,2500)                               RCL02310
2500  FORMAT(' NSEG      TLEN        WA        RHOO       AI      RCL02320
* CFLUID      PRESS')                         RCL02330
      WRITE(6,2001) NSEG,TLEN,WA,RHOO,AI,CFLUID,PRESS RCL02340
2001  FORMAT(1X,I3,6(1X,D12.6))                 RCL02350
      WRITE(6,3400)                               RCL02360
3400  FORMAT(' I      RLENG        RMASS      RMASST     AMAXI    RCL02370
* WEIGHT'   /'      DXI        PXIETA     EA        EIETA   RCL02380
*          AO        EIETAS')                     RCL02390
      DO 3002 I=1,NSEG
      WRITE(6,3003) I,RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXI,RCL02400
* (I),PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I)   RCL02410
3002  CONTINUE                                RCL02420
3003  FORMAT(1X,I3,5(1X,D12.6)/4X,6(1X,D12.6)) RCL02430
C
      WRITE(6,1334)                               RCL02440
1334  FORMAT(' I      AMAETA      DETA      EIXI      EIXIS    RCL02450
* GIP        GIPS'/'      AMAZI      JZI        AJZI' ) RCL02460
      DO 1335 I=1,NSEG
      WRITE(6,1336) I,AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),RCL02470
* AMAZI(I),XJZI(I),AJZI(I)                      RCL02480
1335  CONTINUE                                RCL02490
1336  FORMAT(1X,I3,6(1X,D12.6)/4X,3(1X,D12.6)) RCL02500
C
1701  WRITE(6,1700)                               RCL02510
1700  FORMAT(' INPUT 1 IF RISER DATA ARE CORRECT'/' INPUT 0 TO STOP') RCL02520
      READ(5,*) IC
      IF((IC.NE.0).AND.(IC.NE.1)) GOTO 1701
      IF(IC.EQ.0) RETURN
C
      ENDIF
C
      NON - DIMENSIONALIZATIONS
C
      GRAV=9.81D0
      WAM=WA/GRAV
      WT=WA*TLEN
      XPI=4.D0*DATAN(1.D0)
      XPI2=XPI/2.D0
      RHOW=1.025D3
C
      NONDIMENSIONALIZE CORRECTLY QXIO, TO FROM 2-D STATIC SOLUTION
C
      TOMAX = TOMAX*WT
      TND = WT/TOMAX
      DO 229 I=1,NP
      STATIC(1,I) = STATIC(1,I)*TND

```

```

        STATIC(2,I) = STATIC(2,I)*TND          RCL02810
229  CONTINUE                           RCL02820
C                                         RCL02830
C     NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS RCL02840
C                                         RCL02850
        TOMAXL = TOMAX/TLEN                  RCL02860
        TOML1  = TOMAX*TLEN                  RCL02870
        TOML2  = TOML1*TLEN                  RCL02880
        TLEN2   = TLEN**2                   RCL02890
        TMAXAV = 0.DO                      RCL02900
DO 2000 I=1,NSEG                         RCL02910
        RLENG(I)=RLENG(I)/TLEN             RCL02920
        WEIGHT(I)=WEIGHT(I)/TOMAXL       RCL02930
        EPSXI(I) = EIXI(I)/TOML2         RCL02940
        EPSXIS(I) = EIXIS(I)/TOML1       RCL02950
        EPSETA(I) = EIETA(I)/TOML2       RCL02960
        EPSETS(I) = EIETAS(I)/TOML1      RCL02970
        EPSPI(I) = GIP(I)/TOML2          RCL02980
        EPSPIS(I) = GIPS(I)/TOML1         RCL02990
        TMAETA(I) = RMASS(I) + AMAETA(I) RCL03000
        TMAXI(I) = RMASS(I) + AMAXI(I)    RCL03010
        TMAXAV = TMAXI(I)*RLENG(I) + TMAXAV RCL03020
        TJZI(I) = XJZI(I) + AJZI(I)       RCL03030
        DXIETA(I) = DXI(I)-DETA(I)       RCL03040
2000  CONTINUE                           RCL03050
        HETAM = 0.DO                      RCL03060
DO 4321 I=1,NSEG                         RCL03070
        HETA(I) = TMAETA(I)/TMAXAV       RCL03080
        HETAM = HETAM + HETA(I)*RLENG(I) RCL03090
        TLAMB2(I) = TLEN2*TMAXAV/TJZI(I) RCL03100
4321  CONTINUE                           RCL03110
C                                         RCL03120
C     EVALUATE FUNCTIONS OF STATIC RESULTS RCL03130
DO 521 I=1,NP                            RCL03140
        TENO(I) = STATIC(1,I)           RCL03150
        QXO(I) = STATIC(2,I)           RCL03160
521  CONTINUE                           RCL03170
C                                         RCL03180
C     EVALUATE DERIVATIVES OF STATIC QUANTITIES RCL03190
CALL DER1(TENO,XI,TOS,NP)                 RCL03200
CALL DER1(QXO,XI,QXIOS,NP)                RCL03210
DO 56 I=1,NP                            RCL03220
        CONST1(I) = TOS(I) - STATIC(2,I)*STATIC(3,I)
        CONST2(I) = QXIOS(I) + STATIC(1,I)*STATIC(3,I) RCL03230
56  CONTINUE                           RCL03240
C                                         RCL03250
C     SEG(I)=LEFT ORDINATE OF SEGMENT I RCL03260
        SEG(1)=0.DO                     RCL03270
        SEG(NSEG+1)=1.DO                 RCL03280
DO 4000 I=2,NSEG                         RCL03290
        SEG(I)=RLENG(I-1)+SEG(I-1)       RCL03300
4000  CONTINUE                           RCL03310
C                                         RCL03320
C     INTRERPOLATE STRUCTURAL DIMENSIONS TO NP POINTS RCL03330
C     ASSUMPTION: NUMBER OF RISER SEGMENTS WITH DIFFERENT RCL03340
C     CHARACTERISTICS IS SMALLER THAN THE STATIC SOLUTION POINTS RCL03350
C     NSEG < NPI                         RCL03360
C
        IF (NSEG.GE.NPI) THEN

```

```

      WRITE(6,188)                                     RCL03370
188   FORMAT(' NSEG => NPI,    PROGRAM STOPS')      RCL03380
      IC = 0                                         RCL03390
      RETURN                                         RCL03400
END IF                                           RCL03410
CALL STRUCT(EPSXI,X,NP)                         RCL03420
CALL STRUCT(EPSXIS,X,NP)                        RCL03430
CALL STRUCT(EPSETA,X,NP)                        RCL03440
CALL STRUCT(EPSETS,X,NP)                        RCL03450
CALL STRUCT(EPSPI,X,NP)                         RCL03460
CALL STRUCT(EPSPI,X,NP)                        RCL03470
CALL STRUCT(HETA,X,NP)                          RCL03480
CALL STRUCT(TLAME2,X,NP)                        RCL03490
RETURN                                         RCL03500
END                                             RCL03510
C                                               RCL03520
      SUBROUTINE DER1(ARRAY,X,DERIV,NP)            RCL03530
C THIS SUBROUTINE EVALUATES THE FIRST DERIVATIVE OF A SERIES OF DATA RCL03540
C POINTS USING A SECOND ORDER APPROXIMATION (NONUNIFORM GRID FINITE RCL03550
C DIFFERENCES)                                         RCL03560
      IMPLICIT REAL*8(A-H,O-Z)                     RCL03570
      PARAMETER (MNP=151)                           RCL03580
      DIMENSION ARRAY(MNP),DERIV(MNP),X(MNP)        RCL03590
C END POINTS DERIVATIVE FIRST ORDER             RCL03600
      DERIV(1) = (-ARRAY(1) + ARRAY(2))/(X(2)-X(1)) RCL03610
      DERIV(NP) = (-ARRAY(NP-1) + ARRAY(NP))/(X(NP)-X(NP-1)) RCL03620
C                                         RCL03630
      DO 836 I=2,NP-1                            RCL03640
      DX = X(I) - X(I-1)                         RCL03650
      DX1 = X(I+1) - X(I)                         RCL03660
      D = DX/DX1                                 RCL03670
      DERIV(I) = (D*(ARRAY(I+1)-ARRAY(I)) - (ARRAY(I-1)-ARRAY(I))/D) RCL03680
      DERIV(I) = DERIV(I)/(DX + DX1)              RCL03690
836  CONTINUE                                     RCL03700
      RETURN                                         RCL03710
      END                                             RCL03720
C                                               RCL03730
      SUBROUTINE READ2D(NP)                         RCL03740
C THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM RCL03750
C DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION. RCL03760
      IMPLICIT REAL*8(A-H,O-Z)                     RCL03770
      PARAMETER(N=7,MNP=151,NA=4)                  RCL03780
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCXI(MNP),TOMAX,VM,NPI RCL03790
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE       RCL03800
C                                         RCL03810
      STATIC(1,I) = STATIC TENSION TO             RCL03820
      STATIC(2,I) = STATIC SHEAR FORCE IN THE XI DIRECTION RCL03830
      STATIC(3,I) = STATIC OMEGA AROUND THE ETA DIRECTION RCL03840
      STATIC(4,I) = STATIC ANGLE PHI               RCL03850
C                                         RCL03860
      WRITE(6,2000) MNP                           RCL03870
2000  FORMAT(' MNP=',I3)                         RCL03880
C READ STATIC SOLUTION                         RCL03890
      READ(10,36459) NP,VM                      RCL03900
36459 FORMAT(1X,I3,1X,D12.6)                  RCL03910
      WRITE(6,2311) NP,VM                      RCL03920

```

```

2311 FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
      *' NP =',I3/
      *' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S , VM =',D12.6) RCL03930
      C
      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN RCL03940
      ICCC=0 RCL03950
      WRITE(6,12439) RCL03960
      12439 FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS') RCL03970
      RETURN RCL03980
      ENDIF RCL03990
      C
      C     READING FROM DEVICE 10
      C
      DO 1021 I=1,NP RCL04000
      READ(10,1033) X(I),STATIC(1,I),STATIC(2,I),STATIC(3,I),STATIC(4,I) RCL04010
      *,XCOOR,YCOOR,STRARC,TENSI,VLOCKXI(I) RCL04020
      XI(I) = X(I) RCL04030
      1021 CONTINUE RCL04040
      1033 FORMAT(10(1X,D12.6)) RCL04050
      C     EVALUATE MAXIMUM STATIC TENSION
      TOMAX=DMAX1(STATIC(1,1),STATIC(1,2)) RCL04060
      DO 9859 I=3,NP RCL04070
      TOMAX=DMAX1(TOMAX,STATIC(1,I)) RCL04080
      9859 CONTINUE RCL04090
      C
      WRITE(6,1654) TOMAX RCL04100
      1654 FORMAT(' 2-D STATIC SOLUTION SUCCESFULLY READ'/' MAXIMUM STATIC TENSION/WAL = ',D10.4) RCL04110
      C
      RETURN RCL04120
      END RCL04130
      RCL04140
      RCL04150
      RCL04160
      RCL04170
      RCL04180
      RCL04190
      RCL04200
      RCL04210
      RCL04220
      RCL04230
      RCL04240
      RCL04250
      RCL04260
      RCL04270
      RCL04280
      RCL04290
      RCL04300
      RCL04310
      RCL04320
      RCL04330
      RCL04340
      RCL04350
      RCL04360
      RCL04370
      RCL04380
      RCL04390
      RCL04400
      RCL04410
      RCL04420
      RCL04430
      RCL04440
      RCL04450
      RCL04460
      RCL04470
      RCL04480

      SUBROUTINE READAS(ICCC,NP,TOL)
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(N=7,MNP=151,NA=4)
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI
      COMMON/BOUNDA/BOUND

      C
      C     READAS READS INITIAL APPROXIMATION RESULTING FROM A PREVIOUS
      C     RUN OF RCLINDYN (CORRESPONDS TO EPS = 1.D0)
      C
      C     Y(1,I) = SHEAR FORCE IN THE ETA DIRECTION
      C     Y(2,I) = OMEGA ABOUT ZETA
      C     Y(3,I) = OMEGA ABOUT XI
      C     Y(4,I) = THETA
      C     Y(5,I) = ANGLE BETA
      C     Y(6,I) = OUT OF PLANE DISPLACEMENT R IN THE ETA DIRECTION
      C     Y(7,I) = NATURAL FREQUENCY
      C     X(I) = UNSTRETCHED ARC LENGTH S

      C
      WRITE(6,2000) MNP
      2000 FORMAT(' MNP=',I3)
      C     READ SOLUTION FROM DEVICE 12
      READ(12,36459) MODE,NP,SIGMAD
      36459 FORMAT(1X,I2,1X,I3,1X,D10.4)

```

```

        WRITE(6,2311) NP,MODE,SIGMAD                               RCL04490
2311  FORMAT(' INITIAL APPROXIMATION FROM DEVICE 12'/
          *' NP =',I3,' MODE NUMBER =',I2,' SIGMAD =',D10.4)   RCL04500
C
        IF((NP.LT.4).OR.(NP.GT.MNP)) THEN                      RCL04510
          ICC=0                                                 RCL04520
          WRITE(6,12439)                                       RCL04530
12439  FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS')           RCL04540
          RETURN                                              RCL04550
        ENDIF                                              RCL04560
C
C      READING DATA FROM DEVICE 12 ...
C
        DO 10011 I=1,NP                                     RCL04570
          READ(12,10012) X(I),(Y(J,I),J=1,7),VLOCXI(I)       RCL04580
10011 CONTINUE                                         RCL04590
10012 FORMAT(9(1X,D12.6))                                RCL04600
C      SET BOUNDARY CONDITION OMEGA XI(0)                  RCL04610
          BOUND = Y(3,1)                                    RCL04620
C
        WRITE(9,1052) NP,SIGMAD                             RCL04630
1052  FORMAT(' INITIAL CONDITION FOR EPS=1. AND NP = ',I3,' POINTS,  NATRCL04700
          *URAL FREQUENCY = ',D10.4,' RAD/S'/
          *' I      ARC      SHEAR ETA    OMEGA ZETA    OMEGA XI      THETARCL04720
          *'      BETA      R          SIGMA')                RCL04730
          DO 1601 I=1,NP                                     RCL04740
            WRITE(9,1603) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
          * Y(7,I)                                         RCL04750
1601  CONTINUE                                         RCL04760
1603  FORMAT(1X,I3,8(1X,D12.6))                         RCL04770
C
        WRITE(6,9561)                                         RCL04780
9561  FORMAT(' DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION'/
          *' IF YES INPUT 1')
          READ(5,*) IPRINT                                 RCL04790
          IF(IPRINT.EQ.1) THEN                           RCL04800
            WRITE(6,1052) NP                            RCL04810
            DO 9659 I=1,NP                           RCL04820
              WRITE(6,1603) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
          * Y(7,I)                                         RCL04830
9659  CONTINUE                                         RCL04840
          END IF                                           RCL04850
          WRITE(6,1722) BOUND                           RCL04860
1722  FORMAT(' ASSUMED BOUNDARY CONDITION'/
          *' OMEGA XI (0) =',D12.6)                     RCL04870
C
          TOL1 = 0.0D0                                     RCL04880
          DO 3933 I=1,NP                           RCL04890
            TOL1 = DMAX1(TOL1,DABS(Y(3,I)))             RCL04900
3933 CONTINUE                                         RCL04910
          WRITE(6,3934) TOL1                           RCL04920
3934  FORMAT(' MAXIMUM ABSOLUTE VALUE OF N-D OMEGA XI IS =',D12.6/
          *' THIS NUMBER CAN BE USED TO ESTIMATE'/
          *' A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS'/
          *' INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE'/
          *' E.G. INPUT 0.01 FOR 1% ACCURACY')            RCL04930

```

```

      READ(5,*), TOLV
      TOL = DABS(TOLV)*TOL1

C
1005  WRITE(6,1004)
1004  FORMAT(' IF YOU WISH TO STOP INPUT 0'/
     *' IF YOU WISH TO CONTINUE WITH ITERATIONS INPUT 1')
      READ(5,*), ICCC
      IF((ICCC.NE.0).AND.(ICCC.NE.1)) GOTO 1005

C
      RETURN
      END

C
      SUBROUTINE APPROX(TLENG,ICCC,NP,TOL)
C THIS SUBROUTINE EVALUATES THE APPROXIMATE ASYMPTOTIC ANALYTIC
C SOLUTION OF THE OUT OF PLANE DYNAMIC EQUATIONS.

C
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(N=7,MNP=151,NA=4)
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCXI(MNP),TOMAX,VM,NPI
      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL05250
     *NP),EPSPI(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV
      COMMON/DEF/XNU
      DIMENSION FVEC(1),XS(1),FJAC(1,1),WA(7)
      EXTERNAL FCNS

C
C APPROX DEFINES INITIAL APPROXIMATION (EPS=0.)
C
C Y(1,I) = SHEAR FORCE IN THE ETA DIRECTION
C Y(2,I) = OMEGA ABOUT ZETA
C Y(3,I) = OMEGA ABOUT XI
C Y(4,I) = THETA
C Y(5,I) = ANGLE BETA
C Y(6,I) = OUT OF PLANE DISPLACEMENT R IN THE ETA DIRECTION
C Y(7,I) = NATURAL FREQUENCY

C
C DETERMINE THE INITIAL APPROXIMATION TO THE NATURAL FREQUENCY
C SIGMA.
C
C NU IS EXACT IF THE STRAIN RELIEF UNITS ARE THE SAME AT THE TWO
C ENDS OF THE RISER.
      XNU = DSQRT((EPSXI(1)+EPSXI(NP))/2.D0)
      WRITE(6,581) XNU
581  FORMAT(' THE VALUE OF NU TO BE USED IS = ',D10.4,' IF YOU WANT TO
     *CHANGE NU INPUT 1')
      READ(5,*),INU
      IF (INU.EQ.1) THEN
        WRITE(6,*), ' INPUT NU'
        READ(5,*),XNU
      END IF
23   WRITE(6,98)
98   FORMAT(' INPUT INITIAL GUESS FOR SIGMA IN RAD/SEC ')
      READ(5,*),XS(1)
      NONDIMENSIONALIZE SIGMA
      XS(1) = XS(1)*TLENG*DSQRT(TMAXAV*HETAM/TOMAX)
      XTOL = 1.D-8
      NS = 1

```

FILE: RCLINDYN FORTRAN A

```

LDFJAC = 1 RCL05610
LWA = 7 RCL05620
IFAIL = 0 RCL05630
C CALL C05PBF TO EVALUATE SIGMA USING A MODIFICATION OF THE POWELL RCL05640
C HYBRID METHOD RCL05650
C CALL C05PBF(FCNS,NS,XS,FVEC,FJAC,LDFJAC,XTOL,WA,LWA,IFAIL) RCL05660
C WRITE(6,*) ' IFAIL FOR APPROXIMATE EVALUATION OF SIGMA IS ',IFAIL RCL05670
C SIGMAD = XS(1)*DSQRT(TOMAX/(TMAXAV*NEMAT))/TLENG RCL05680
C WRITE(6,24) FVEC(1),SIGMAD RCL05690
24 FORMAT(1X,'THE FUNCTION IS F = ',D10.4/' THE CONVERGED SIGMA IS ',RCL05700
* D10.4/' INPUT 1 IF YOU WANT TO REDO THE CALCULATION' RCL05710
* ' INPUT 2 IF YOU WANT TO STOP') RCL05720
READ(5,*) ISIGMA RCL05730
IF (ISIGMA.EQ.1) THEN RCL05740
GO TO 23 RCL05750
ELSE IF (ISIGMA.EQ.2) THEN RCL05760
ICCC = 0 RCL05770
RETURN RCL05780
END IF RCL05790
SIGMA = XS(1) RCL05800
WRITE(6,*) ' INPUT MODE NUMBER CORRESPONDING TO ABOVE APPROXIMATE RCL05810
*SIGMA' RCL05820
READ(5,*) MODE RCL05830
RCL05840
C CALL SUBROUTINES TO EVALUATE INITIAL APPROXIMATION RCL05850
C CALL INITAP(SIGMA,NP) RCL05860
C OUTPUT INITIAL APPROXIMATION TO DEVICE 9 RCL05870
C WRITE(9,1002) NP,SIGMAD RCL05880
1002 FORMAT(' INITIAL CONDITION FOR EPS=0. AND NP = ',I3,' POINTS, NATRCL05890
*URAL FREQUENCY = ',D10.4,' RAD/SEC'/ RCL05900
* I ARC SHEAR ETA OMEGA ZETA OMEGA XI THETARCL05910
* BETA R SIGMA') RCL05920
* Y(7,I) RCL05930
DO 1001 I=1,NP RCL05940
WRITE(9,1003) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I), RCL05950
* Y(7,I) RCL05960
1001 CONTINUE RCL05970
1003 FORMAT(1X,I3,8(1X,D12.6)) RCL05980
RCL05990
C WRITE(6,7561) RCL06000
7561 FORMAT(' DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION'/
*' IF YES INPUT 1') RCL06010
READ(5,*) IPRINT RCL06020
IF(IPRINT.EQ.1) THEN RCL06030
WRITE(6,1002) NP,SIGMAD RCL06040
DO 7659 I=1,NP RCL06050
WRITE(6,1003) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I), RCL06060
* Y(7,I) RCL06070
7659 CONTINUE RCL06080
END IF RCL06090
TOL1 = 0.D0 RCL06100
DO 3931 I=1,NP RCL06110
TOL1 = DMAX1(TOL1,DABS(Y(3,I))) RCL06120
3931 CONTINUE RCL06130
WRITE(6,3932) TOL1 RCL06140
3932 FORMAT(' MAXIMUM ABSOLUTE VALUE OF N-D OMEGA XI IS ',D12.6/ RCL06150
*' THIS NUMBER CAN BE USED TO ESTIMATE ' RCL06160

```

```

*' A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS'/
RCL06170
*' INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE'/
RCL06180
*' E.G. INPUT 0.01 FOR 1% ACCURACY')
RCL06190
READ(5,*) TOLV
RCL06200
TOL = DABS(TOLV)*TOL1
RCL06210
WRITE(6,*) ' IF YOU WANT TO STOP INPUT 0'
RCL06220
READ(5,*) ICCC
RCL06230
IF (ICCC.NE.0) ICCC = 1
RCL06240
C
RCL06250
RETURN
RCL06260
END
RCL06270
C
RCL06280
SUBROUTINE FCNS(N,X,FVEC,FJAC,LDFJAC,IFLAG)
RCL06290
THIS SUBROUTINE IS USED BY NAG ROUTINE C05PBF TO EVALUATE
RCL06300
THE FUNCTION WHICH DETERMINES SIGMA
RCL06310
IMPLICIT REAL*8(A-H,O-Z)
RCL06320
COMMON/DEF/XNU
RCL06330
DIMENSION X(1),FVEC(1),FJAC(1,1)
RCL06340
IF (IFLAG.EQ.1) THEN
RCL06350
  FVEC(1)=(1.D0-X(1)**2*XNU**2)*DSIN(X(1))-2.D0*X(1)*XNU*DCOS(X(1))
RCL06360
ELSE IF (IFLAG.EQ.2) THEN
RCL06370
  FJAC(1,1) = (1.D0-X(1)**2*XNU**2-2.D0*XNU)*DCOS(X(1))
RCL06380
  FJAC(1,1) = FJAC(1,1) + 2.D0*X(1)*XNU*(1.D0-XNU)*DSIN(X(1))
RCL06390
ELSE IF (IFLAG.LT.0) THEN
RCL06400
  STOP
RCL06410
END IF
RCL06420
RETURN
RCL06430
END
RCL06440
C
RCL06450
SUBROUTINE INITAP(SIGMA,NP)
RCL06460
IMPLICIT REAL*8(A-H,O-Z)
RCL06470
PARAMETER(N=7,MNP=151,NA=4)
RCL06480
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKI(MNP),TOMAX,VM,NPI
RCL06490
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
RCL06500
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL06510
*NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV
RCL06520
COMMON/BOUNDA/BOUND
RCL06530
DIMENSION R(MNP),RS(MNP),RSS(MNP),RSSS(MNP),EPSI(MNP),RSI(MNP)
RCL06540
DIMENSION RSSI(MNP),RSSEI(MNP),A1(MNP)
RCL06550
RCL06560
C
C
THIS SUBROUTINE CALCULATES THE INITIAL APPROXIMATION OF THE
RCL06570
SOLUTION ONCE SIGMA IS DETERMINED
RCL06580
C
RCL06590
FIND SIGMA
RCL06600
S2HETA = SIGMA/DSQRT(HETAM)
RCL06610
DO 444 K=1,NP
RCL06620
  Y(7,K) = S2HETA
RCL06630
444  CONTINUE
RCL06640
CONST = DCOS(SIGMA) + SIGMA*XNU*DSIN(SIGMA)
RCL06650
C
C
IBM RANGE OF A REAL*8 NUMBER SO THAT DEXP(X) IS REAL*8
RCL06660
CONDITION XMINR<X. USE A STRICTER LIMIT.
RCL06670
RCL06680
RCL06690
XMINR=-65.D0*DLOG(16.D0)
RCL06700
XMINR=XMINR/3.D0
RCL06710
C
RCL06720

```

FILE: RCLINDYN FORTRAN A

```

C      CALCULATE THE OUT OF PLANE DISPLACEMENT R AND ITS THREE          RCL06730
C      DERIVATIVES (ANALYTIC EXPRESSIONS)                                RCL06740
C
C      XNU1 = DSQRT(EPSXI(1))                                         RCL06750
C      XNU2 = DSQRT(EPSXI(NP))                                         RCL06760
C      XNU = XNU1                                                       RCL06770
C      DO 815 I=1,NP                                                 RCL06780
C          SIG = SIGMA*X(I)                                           RCL06790
C          COSS = DCOS(SIG)                                         RCL06800
C          SINS = DSIN(SIG)                                         RCL06810
C          POWER1 = -X(I)/XNU1                                       RCL06820
C          POWER2 = -(1-X(I))/XNU2                                     RCL06830
C          IF (POWER1.GT.XMINR) THEN                                 RCL06840
C              EXPO1 = DEXP(POWER1)                                    RCL06850
C          ELSE                                                       RCL06860
C              EXPO1 = 0.0D0                                         RCL06870
C          END IF                                                     RCL06880
C          IF (POWER2.GT.XMINR) THEN                                 RCL06890
C              EXPO2 = CONST*DEXP(POWER2)                           RCL06900
C          ELSE                                                       RCL06910
C              EXPO2 = 0.0D0                                         RCL06920
C          END IF                                                     RCL06930
C          CALCULATE R, RS, RSS, RSSS                                RCL06940
C          R(I) = -SINS/(XNU*SIGMA) + COSS - EXPO1 + EXPO2           RCL06950
C          RS(I) = -COSS/XNU - SIGMA*SINS + (EXPO1 + EXPO2)/XNU       RCL06960
C          RSS(I) = SIGMA*SINS/XNU - SIGMA**2*COSS + (EXPO2 - EXPO1)/XNU**2 RCL06970
C          RSSS(I)=SIGMA**2*COSS/XNU + SIGMA**3*SINS + (EXPO1+EXPO2)/XNU**3 RCL06980
C          DEFINE FIRST PART OF INITIAL APPROXIMATION               RCL06990
C          SHEAR FORCE IN THE ETA, OMEGA XI, THETA, R                RCL07000
C
C          Y(1,I) = EPSXI(I)*(-RSSS(I)) + EPSXIS(I)*(-RSS(I))        RCL07010
C          Y(3,I) = -RSS(I)                                         RCL07020
C          Y(4,I) = -RS(I)                                         RCL07030
C          Y(6,I) = R(I)                                          RCL07040
C
C          815 CONTINUE                                              RCL07050
C
C          EVALUATE THE INDEFINITE INTEGRALS TO CALCULATE ANGLE BETA USING   RCL07060
C          THE TRAPEZOIDAL INTEGRATION ALGORITHM.                         RCL07070
C
C          EPSI(I) = 0.0D0                                         RCL07080
C          RSI(I) = 0.0D0                                         RCL07090
C          RSSI(I) = 0.0D0                                         RCL07100
C          RSSEI(I) = 0.0D0                                         RCL07110
C          DO 839 I=2,NP                                         RCL07120
C              DX2 = (X(I)-X(I-1))*0.5D0                           RCL07130
C
C              EPSI(I) = EPSI(I-1) + (1.0D0/EPSPSI(I)+1.0D0/EPSPSI(I-1))*DX2 RCL07140
C
C              RSI(I) = RSI(I-1) + (STATIC(3,I)*RS(I)+STATIC(3,I-1)*RS(I-1))*DX2 RCL07150
C
C              E1 = (EPSETA(I)-EPSXI(I))*STATIC(3,I)                  RCL07160
C              E2 = (EPSETA(I-1)-EPSXI(I-1))*STATIC(3,I-1)            RCL07170
C              RSSI(I) = RSSI(I-1) + (E1*RSS(I) + E2*RSS(I-1))*DX2     RCL07180
C
C              E3 = RSSI(I)/EPSPSI(I) + RSSI(I-1)/EPSPSI(I-1)         RCL07190
C              RSSEI(I) = RSSEI(I-1) + E3*DX2                         RCL07200

```

```

839 CONTINUE                                RCL07290
C   EVALUATE THE CONSTANT OF INTEGRATION      RCL07300
C   C1 = (RSI(NP)-RSSEI(NP))/EPSI(NP)        RCL07310
C   EVALUATE THE SECOND PART OF THE APPROXIMATE SOLUTION RCL07320
C   DO 853 I=1,NP                            RCL07330
C     Y(2,I) = (RSSI(I) + C1)/EPSPI(I)        RCL07340
C     Y(5,I) = -RSI(I) + C1*EPSI(I) + RSSEI(I) RCL07350
C     A1(I) = Y(5,I)**2/TLAMB2(I) + HETA(I)*Y(6,I)**2 RCL07360
853 CONTINUE                                RCL07370
C   CALCULATE THE ORTHONORMALIZING CONSTANT FOR R AND BETA RCL07380
C   A = 0.D0                                  RCL07390
C   DO 842 I=2,NP                            RCL07400
C     DX2 = (X(I)-X(I-1))*0.5D0              RCL07410
C     A = A + (A1(I)+A1(I-1))*DX2            RCL07420
842 CONTINUE                                RCL07430
C   A = DSQRT(A)                            RCL07440
C   ORTHONORMALIZE THE SOLUTION EXCEPT SIGMA RCL07450
C   DO 843 I=1,NP                            RCL07460
C     DO 843 K=1,N-1                          RCL07470
C       Y(K,I) = Y(K,I)/A                  RCL07480
843 CONTINUE                                RCL07490
C   DETERMINE THE FOURTH BOUNDARY CONDITION AT S=0 RCL07500
C   BOUND = Y(3,1)                            RCL07510
C   WRITE(6,84) A,BOUND                      RCL07520
84 FORMAT(' THE ORTHONORMALIZING CONSTANT FOR THE APPROXIMATE'/
*' SOLUTION IS SQRT(A) = ',D10.4/' THE BOUNDARY CONDITION AT S=0
* OMEGAXI(0) = ',D10.4)
RETURN
END

C   SUBROUTINE STRUCT(ARRAY,X,NP)             RCL07580
C   THIS SUBROUTINE CHANGES A SERIES OF DATA GIVEN IN      RCL07590
C   SEGMENTS TO A SERIES OF DATA CORRESPONDING TO SPECIFIC POINTS RCL07600
C   ASSUMPTION: NSEG < NP                    RCL07610
C   IMPLICIT REAL*8(A-H,O-Z)                RCL07620
C   PARAMETER(MNP=151)                      RCL07630
C   COMMON/INPUT1/TLEN,WA,WT,NSEG            RCL07640
C   COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)      RCL07650
C   DIMENSION ARRAY(MNP),HELP(MNP),X(MNP)    RCL07660
C
C   IF(NSEG.EQ.1) THEN                      RCL07670
C     DO 83 I=1,NP                          RCL07680
C       HELP(I) = ARRAY(I)                  RCL07690
83   CONTINUE                                RCL07700
C   ELSE                                     RCL07710
C     HELP(1) = ARRAY(1)                  RCL07720
C     HELP(NP) = ARRAY(NSEG)              RCL07730
C     I=2                                    RCL07740
C     DO 84 K=2,NP-1                      RCL07750
C       IF ((X(K).GT.SEG(I-1)).AND.(X(K).LT.SEG(I))) THEN RCL07760
C         HELP(K) = ARRAY(I-1)              RCL07770
C       ELSE IF (X(K).EQ.SEG(I)) THEN      RCL07780
C         HELP(K) = 0.5D0*(ARRAY(I-1) + ARRAY(I)) RCL07790
C       ELSE IF (X(K).GT.SEG(I)) THEN      RCL07800
C         HELP(K) = ARRAY(I)              RCL07810
C       I = I + 1                         RCL07820

```

```

        END IF
84    CONTINUE
        END IF
        DO 85 K=1,NP
          ARRAY(K) = HELP(K)
85    CONTINUE
        RETURN
        END

C
C      SUBROUTINE OUTPUT(NP)
C      THIS SUBROUTINE OUTPUTS THE RESULTS IN TWO FORMATS; A COMPLETE
C      FORMAL FORM AND A FORM TO BE USED FOR PLOTTING.
C
C      IMPLICIT REAL*8(A-H,O-Z)
C      PARAMETER(MNP=151,NA=4,N=7)
C      CHARACTER*80 NAME
C      COMMON/CONST/XPI,XPI2,RHOW,GRAV
C      COMMON/INPUT0/NAME
C      COMMON/INPUT1/TLEN,WA,WT,NSEG
C      COMMON/INPUT2/RHO0,AI,CFLUID,PRESS,AO(MNP)
C      COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)
C      COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP) RCL07990
C      * ,EIXIS(MNP),GIP(MNP),GIPS(MNP) RCL08000
C      COMMON/INPUT4/DXI(MNP),PXIETA(MNP),DETA(MNP),DXIETA(MNP) RCL08010
C      COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP) RCL08020
C      * ,TMAXI(MNP),TMAETA(MNP) RCL08030
C      COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP) RCL08040
C      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCXI(MNP),TOMAX,VM,NPI RCL08050
C      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL08120
C      * NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV RCL08130
C      COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,TO,QXIO,OMEGA0,RCL08140
C      * CONS1,CONS2 RCL08150
C      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL08160
C      DIMENSION A1(MNP) RCL08170
C      CALCULATE THE ORTHONORMALIZING CONSTANT FOR R AND BETA RCL08180
C
C      DO 542 I=1,NP RCL08190
C        CALL COUNT(X(I)) RCL08200
C        A1(I) = Y(5,I)**2/TLAM2 + HH*Y(6,I)**2 RCL08210
542  CONTINUE RCL08220
        A = 0.D0 RCL08230
        DO 642 I=2,NP RCL08240
          DX2 = (X(I)-X(I-1))*0.5D0 RCL08250
          A = A + (A1(I)+A1(I-1))*DX2 RCL08260
642  CONTINUE RCL08270
        A = DSQRT(A) RCL08280
C        ORTHONORMALIZE THE SOLUTION EXCEPT SIGMA RCL08290
        DO 643 I=1,NP RCL08300
          Y(N,I) = DABS(Y(N,I)) RCL08310
          DO 643 K=1,N-1 RCL08320
            Y(K,I) = Y(K,I)/A RCL08330
643  CONTINUE RCL08340
C        OUTPUT RESULTS RCL08350
        WRITE(9,1000) NAME RCL08360
1000  FORMAT(80A) RCL08370
        WRITE(9,1001) NSEG,TLEN,WA,RHO0,RHOW,AI,CFLUID,PRESS,WT,TOMAX,VM* RCL08380
        WRITE(9,1001) RCL08390
C

```

```

1001 FORMAT(1X,I3,9X,' = NUMBER OF RISER SEGMENTS'/
*1X,D12.6,' = UNSTRETCHED RISER LENGTH IN M'/
*1X,D12.6,' = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH RCL08430
*IN N/M'/ RCL08440
*1X,D12.6,' = INNER FLUID DENSITY IN KG/M3'/
*1X,D12.6,' = SALT WATER DENSITY IN KG/M3'/
*1X,D12.6,' = INNER CROSS SECTIONAL AREA IN M2'/
*1X,D12.6,' = INNER FLUID SPEED IN M/S'/
*1X,D12.6,' = INNER FLUID OVERPRESSURE IN N/M2'/
*1X,D12.6,' = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N'/
*1X,D12.6,' = MAXIMUM STATIC TENSION IN N'/
*1X,D12.6,' = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S') RCL08450
RCL08460
RCL08470
RCL08480
RCL08490
RCL08500
RCL08510
RCL08520
C EVALUATE DIMENSIONAL NATURAL FREQUENCY RCL08530
SIGMAD = Y(N,1)*DSQRT(TOMAX/TMAXAV)/TLEN RCL08540
WRITE(9,1002) NSEG RCL08550
1002 FORMAT(//' DATA P E R R I S E R S E G M E N T F O R N S E G =RCL08560
* ',I3,' S E G M E N T S'/' D I M E N S I O N A L Q U A N T I T I RCL08570
* E S I N T H E S . I . S Y S T E M'/
* ' RLENG      DXI      PXIETA      AO      WEIGHT      MASS      RCL08580
* ' TMASS      AMAXI      AMAETA      AMAZI      TMAXI      TMAETA RCL08600
* ') RCL08610
TL = TOMAX/TLEN RCL08620
DO 1004 I=1,NSEG RCL08630
WRITE(9,1003) RLENG(I)*TLEN,DXI(I),PXIETA(I),AO(I),WEIGHT(I)*TL,RMRCL08640
*ASS(I),RMASST(I),AMAXI(I),AMAETA(I),AMAZI(I),TMAXI(I),TMAETA(I) RCL08650
1004 CONTINUE RCL08660
1003 FORMAT(12(1X,D10.4)) RCL08670
C RCL08680
        WRITE(9,10022) RCL08690
10022 FORMAT(' EA      EIETA      EIETAS      EIXI      EIXIS      RCL08700
* GIP      GIPS      DETA      JZI      AJZI      TJZI') RCL08710
DO 10023 I=1,NSEG RCL08720
WRITE(9,10024) EA(I),EIETA(I),EIETAS(I),EIXI(I),EIXIS(I),GIP(I),
*GIPS(I),DETA(I),XJZI(I),AJZI(I),TJZI(I) RCL08730
RCL08740
10023 CONTINUE RCL08750
10024 FORMAT(11(1X,D10.4)) RCL08760
C RCL08770
        WRITE(9,761) MODE,SIGMAD RCL08780
761 FORMAT(' *****' RCL08790
* /' M O D E N U M B E R = ',I2/' N A T U R A L F R E Q U E N C Y = ',D10.4,' RAD/S'/' **** RCL08800
*C Y = ',D10.4,' RAD/S'/' *****' RCL08810
*****') RCL08820
C RCL08830
        WRITE(9,1009) NP RCL08840
1009 FORMAT(/' ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = ',RCL08850
*I3,' POINTS'/
*' S      QETA      OMEGAZETA      OMEGAXI      THETA      BETA      RCL08860
* R      SIGMA') RCL08880
C NORMALIZE ERROR BY COMPONENTS. RCL08890
DO 181 I=1,N-1
    ABT(I) = ABT(I)/A RCL08900
181 CONTINUE RCL08920
DO 1010 I=1,NP RCL08930
WRITE(9,1011) X(I),(Y(J,I),J=1,7) RCL08940
1010 CONTINUE RCL08950
1011 FORMAT(8(1X,D10.4)) RCL08960

```

FILE: RCLINDYN FORTRAN A

```

      WRITE(9,10119) (ABT(I),I=1,7)                                RCL08970
10119 FORMAT(' MAXIMUM ESTIMATED ERROR BY COMPONENTS'/11X,7(1X,D10.4)) RCL08980
C
C
C   OUTPUT TO FILE CONNECTED TO DEVICE 11                         RCL08990
C   THIS CAN BE USED FOR PLOTS OR INPUT TO ANOTHER RUN OF RCLINDYN RCL09000
C
C
C   CALL STRUC(VLOCXI,NP,NPI)                                     RCL09010
      WRITE(11,36459) MODE,NP,SIGMAD                               RCL09020
36459 FORMAT(1X,I2,1X,I3,1X,D10.4)                               RCL09030
      DO 3666 I=1,NP                                              RCL09040
      WRITE(11,3667) X(I),(Y(J,I),J=1,7),VLOCXI(I)             RCL09050
3666  CONTINUE                                                 RCL09060
3667  FORMAT(9(1X,D12.6))                                         RCL09070
C
      RETURN                                                       RCL09080
      END
C
C   SUBROUTINE FCN(X,EPS,Y,F,N)
C   THIS SUBROUTINE EVALUATES THE FUNCTIONS USED BY D02RAF TO SOLVE RCL09090
C   THE PROBLEM                                                 RCL09100
      IMPLICIT REAL*8(A-H,O-Z)                                     RCL09110
      PARAMETER(MNP=151)                                         RCL09120
      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL09210
      *NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV           RCL09220
      COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,T0,QXIO,OMEGA0,RCL09230
      *CONS1,CONS2                                               RCL09240
      DIMENSION Y(N),F(N)                                         RCL09250
      LOCATE MESH POINT TO EVALUATE COEFFICIENTS                 RCL09260
      CALL COUNT(X)                                              RCL09270
C
      F(1) = -EPS*QXIO*Y(2) + (1.D0+EPS*(T0-1.D0))*Y(3)          RCL09280
      F(1) = F(1) + EPS*Y(4)*CONS1                               RCL09290
      F(1) = F(1) - EPS*Y(5)*CONS2                               RCL09300
      F(1) = F(1) - Y(7)**2*(HETAM+EPS*(HH-HETAM))*Y(6)          RCL09310
C
      F(2) = (-EPIS*Y(2) - EPS*Y(7)**2*Y(5)/TLAM2)/EPI          RCL09320
      F(2) = F(2) + Y(3)*OMEGA0*(EXI-EETA)/EPI                  RCL09330
C
      F(3) = (-EXIS*Y(3) + Y(1))/EXI                            RCL09340
      F(3) = F(3) + EPS*Y(2)*OMEGA0*(EETA-EPI)/EXI              RCL09350
C
      F(4) = Y(3) - EPS*OMEGA0*Y(5)                            RCL09360
C
      F(5) = Y(2) + OMEGA0*Y(4)                                 RCL09370
C
      F(6) = -Y(4)                                              RCL09380
C
      F(7) = 0.D0                                                RCL09390
      RETURN
      END
C
C   SUBROUTINE G(EPS,YA,YB,BC,N)
C   BOUNDARY CONDITIONS USED BY D02RAF                         RCL09400
C
      IMPLICIT REAL*8(A-H,O-Z)                                     RCL09410

```

```

      DIMENSION YA(N),YB(N),BC(N)                      RCL09530
      COMMON/BOUNDA/BOUND                                RCL09540
C
      BC(1)=YA(4)                                       RCL09550
      BC(2)=YA(5)                                       RCL09560
      BC(3)=YA(6)                                       RCL09570
      BC(4)=YA(3) - BOUND                               RCL09580
C
      BC(5)=YB(4)                                       RCL09590
      BC(6)=YB(5)                                       RCL09600
      BC(7)=YB(6)                                       RCL09610
C
      RETURN                                              RCL09620
      END                                                 RCL09630
C
      SUBROUTINE JACOBF(X,EPS,Y,F,N)                    RCL09640
C
      THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE EQUATIONS TO USE IN RCL09650
      C NEWTON'S ITERATION.                                RCL09660
C
      IMPLICIT REAL*8(A-H,O-Z)                          RCL09670
      PARAMETER(MNP=151)                                 RCL09680
      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL09730
      *NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV          RCL09740
      COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,T0,QXIO,OMEGAO,RCL09750
      *CONS1,CONS2                                         RCL09760
      DIMENSION Y(N),F(N,N)                            RCL09770
C
      LOCATE MESH POINT TO EVALUATE COEFFICIENTS        RCL09780
      CALL COUNT(X)                                     RCL09790
C
      DO 817 I=1,N                                     RCL09800
      DO 817 M=1,N                                     RCL09810
      F(I,M) = 0.D0                                     RCL09820
817  CONTINUE                                         RCL09830
C
      F(1,2) = -EPS*QXIO                                RCL09840
      F(1,3) = 1.D0 + EPS*(T0-1.D0)                   RCL09850
      F(1,4) = EPS*CONS1                                RCL09860
      F(1,5) = -EPS*CONS2                                RCL09870
      F(1,6) = -Y(7)**2*(HETAM+EPS*(HH-HETAM))       RCL09880
      F(1,7) = -2.D0*Y(7)*Y(6)*(HETAM+EPS*(HH-HETAM)) RCL09890
C
      F(2,2) = -EPIS/EPI                                RCL09900
      F(2,3) = OMEGA0*(EXI-EETA)/EPI                  RCL09910
      F(2,5) = -EPS*Y(7)**2/EPI/TLAM2                RCL09920
      F(2,7) = -EPS*2.D0*Y(7)*Y(5)/EPI/TLAM2         RCL09930
C
      F(3,1) = 1.D0/EXI                                 RCL09940
      F(3,2) = EPS*OMEGA0*(EETA-EPI)/EXI             RCL09950
      F(3,3) = -EXIS/EXI                                RCL09960
C
      F(4,3) = 1.D0                                     RCL09970
      F(4,5) = -EPS*OMEGA0                            RCL09980
C
      F(5,2) = 1.D0                                     RCL09990
      F(5,4) = OMEGA0                                    RCL10000
C
      F(6,4) = -1.D0                                     RCL10010

```

FILE: RCLINDYN FORTRAN A

```

      RETURN                               RCL10090
      END                                RCL10100
C
C      SUBROUTINE JACOBG(EPS,YA,YB,AJ,BJ,N)          RCL10110
C      THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE BOUNDARY CONDITIONS. RCL10120
C      IMPLICIT REAL*8(A-H,O-Z)                      RCL10130
C      DIMENSION YA(N),YB(N),AJ(N,N),BJ(N,N)        RCL10140
C      DO 876 K=1,N                            RCL10150
C          DO 876 I=1,N                          RCL10160
C              AJ(K,I) = 0.DO                      RCL10170
C              BJ(K,I) = 0.DO                      RCL10180
C 876 CONTINUE                           RCL10190
C
C      AJ(1,4) = 1.DO                         RCL10200
C      AJ(2,5) = 1.DO                         RCL10210
C      AJ(3,6) = 1.DO                         RCL10220
C      AJ(4,3) = 1.DO                         RCL10230
C
C      BJ(5,4) = 1.DO                         RCL10240
C      BJ(6,5) = 1.DO                         RCL10250
C      BJ(7,6) = 1.DO                         RCL10260
C      RETURN                                RCL10270
C      END                                  RCL10280
C
C      SUBROUTINE JACEPS(X,EPS,Y,F,N)           RCL10290
C      THIS SUBROUTINE EVALUATES THE DERIVATIVES OF THE FUNCTIONS WITH
C      RESPECT TO THE CONTINUATION PARAMETER EPS.          RCL10300
C      IMPLICIT REAL*8(A-H,O-Z)                  RCL10310
C      PARAMETER(NA=4,MNP=151)                   RCL10320
C      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI    RCL10330
C      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL10340
C      *NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV          RCL10350
C      COMMON/COEFL/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,T0,QXIO,OMEGA0,RCL10360
C      *CONS1,CONS2                                RCL10370
C      DIMENSION Y(N),F(N)                      RCL10380
C      LOCATE MESH POINT TO EVALUATE THE COEFFICIENTS
C      CALL COUNT(X)                            RCL10390
C
C      F(1) = Y(4)*CONS1 - Y(5)*CONS2          RCL10400
C      F(1) = F(1) - QXIO*Y(2) + (T0-1.DO)*Y(3)    RCL10410
C      F(1) = F(1) - Y(7)**2*Y(6)*(HH-HETAM)      RCL10420
C
C      F(2) = -Y(7)**2*Y(5)/EPI/TLAM2          RCL10430
C
C      F(3) = (EETA-EPI)*OMEGA0*Y(2)/EXI       RCL10440
C
C      F(4) = -OMEGA0*Y(5)                      RCL10450
C
C      F(5) = 0.DO                            RCL10460
C      F(6) = 0.DO                            RCL10470
C      F(7) = 0.DO                            RCL10480
C      RETURN                                RCL10490
C      END                                  RCL10500
C
C      SUBROUTINE JACGEP(EPS,YA,YB,BCEP,N)        RCL10510
C      THIS SUBROUTINE EVALUATES THE DERIVATIVES OF THE BOUNDARY      RCL10520
C                                         RCL10530
C                                         RCL10540
C                                         RCL10550
C                                         RCL10560
C                                         RCL10570
C                                         RCL10580
C                                         RCL10590
C                                         RCL10600
C                                         RCL10610
C                                         RCL10620
C                                         RCL10630
C                                         RCL10640

```

```

C      CONDITIONS WITH RESPECT TO THE CONTINUATION PARAMETER EPS.          RCL10650
      IMPLICIT REAL*8(A-H,O-Z)                                              RCL10660
      DIMENSION YA(N),YB(N),BCEP(N)                                         RCL10670
      DO 871 K=1,N
        BCEP(K) = 0.D0
871  CONTINUE
      RETURN
      END
C      SUBROUTINE INTERP(NP)                                                 RCL10730
C      THIS SUBROUTINE INTERPOLATES THE RISER CHARACTERISTICS AND THE      RCL10750
C      STATIC SOLUTION TO THE NEW NUMBER OF POINTS NP.                      RCL10760
C      ASSUMPTION: NP .GE. NPI                                               RCL10770
      IMPLICIT REAL*8(A-H,O-Z)                                              RCL10780
      PARAMETER(MNP=151,NA=4,N=7)                                            RCL10790
      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL10800
      *NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV                  RCL10810
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI           RCL10820
      COMMON/STAT1/CONST1(MNP),CONST2(MNP)                                     RCL10830
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE                             RCL10840
      DIMENSION HELP(MNP)                                                   RCL10850
C      INTERPOLATE STRUCTURAL DATA TO THE NEW NUMBER OF POINTS            RCL10860
      CALL STRUC(EPSXI,NP,NPI)                                              RCL10870
      CALL STRUC(EPSXIS,NP,NPI)                                             RCL10880
      CALL STRUC(EPSETA,NP,NPI)                                             RCL10890
      CALL STRUC(EPSETS,NP,NPI)                                              RCL10900
      CALL STRUC(EPSPI,NP,NPI)                                              RCL10910
      CALL STRUC(EPSPIS,NP,NPI)                                             RCL10920
      CALL STRUC(HETA,NP,NPI)                                                RCL10930
      CALL STRUC(TLAMB2,NP,NPI)                                              RCL10940
      CALL STRUC(CONST1,NP,NPI)                                             RCL10950
      CALL STRUC(CONST2,NP,NPI)                                             RCL10960
      CALL STRUC(VLOCKXI,NP,NPI)                                            RCL10970
C      INTERPOLATE STATIC SOLUTION TO THE NEW NUMBER OF POINTS            RCL10980
      DO 459 K=1,3
        DO 458 I=1,NPI
          HELP(I) = STATIC(K,I)
458   CONTINUE
        CALL STRUC(HELP,NP,NPI)                                              RCL11020
        DO 457 I=1,NP
          STATIC(K,I) = HELP(I)
457   CONTINUE
459   CONTINUE
      DO 339 I=1,NP
        XI(I) = X(I)
339   CONTINUE
      RETURN
      END
C      SUBROUTINE STRUC(ARRAY,NP,NPOLD)                                       RCL11130
C      THIS SUBROUTINE INTERPOLATES A SERIES OF DIVISION POINTS TO A NEW RCL11150
C      SERIES OF DIVISION POINTS.                                           RCL11160
      IMPLICIT REAL*8(A-H,O-Z)                                              RCL11170
      PARAMETER(MNP=151,NA=4,N=7)                                            RCL11180
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE                           RCL11190
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI          RCL11200

```

FILE: RCLINDYN FORTRAN A

```

DIMENSION ARRAY(MNP),HELP(MNP) RCL11210
C RCL11220
HELP(1) = ARRAY(1) RCL11230
HELP(NP) = ARRAY(NPOLD) RCL11240
DO 82 I=2,NPOLD RCL11250
DO 81 K=2,NP-1 RCL11260
IF ((X(K).GT.XI(I-1)).AND.(X(K).LT.XI(I))) THEN RCL11270
  CONV = (X(K)-XI(I-1))*(ARRAY(I)-ARRAY(I-1))/(XI(I)-XI(I-1)) RCL11280
  HELP(K) = ARRAY(I-1) + CONV RCL11290
ELSE IF (X(K).EQ.XI(I)) THEN RCL11300
  HELP(K) = ARRAY(I) RCL11310
END IF RCL11320
81 CONTINUE RCL11330
82 CONTINUE RCL11340
DO 85 K=1,NP RCL11350
  ARRAY(K) = HELP(K) RCL11360
85 CONTINUE RCL11370
RETURN RCL11380
END RCL11390
RCL11400
RCL11410
C SUBROUTINE COUNT(X) RCL11420
C THIS SUBROUTINE EVALUATES THE RISER CHARACTERISTICS AS WELL AS THE RCL11430
C STATIC SOLUTION AT A POINT X. RCL11440
C IT RETURNS THEIR VALUE IN COMMON BLOCK COEF1. RCL11450
C IMPLICIT REAL*8(A-H,O-Z) RCL11460
PARAMETER(MNP=151,NA=4) RCL11470
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL11470
*NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV RCL11480
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL11490
COMMON/STAT1/CONST1(MNP),CONST2(MNP) RCL11500
COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,T0,QXIO,OMEGAO,RCL11510
*CONS1,CONS2 RCL11520
COMMON/COUN/ICOUNT RCL11530
M = ICOUNT RCL11540
IF (X.EQ.XI(M)) THEN RCL11550
  EXI = EPSXI(M) RCL11560
  EXIS = EPSXIS(M) RCL11570
  EETA = EPSETA(M) RCL11580
  EETAS = EPSETS(M) RCL11590
  EPI = EPSPI(M) RCL11600
  EPIS = EPSPIS(M) RCL11610
  HH = HETA(M) RCL11620
  TLAM2 = TLAMB2(M) RCL11630
  TO = STATIC(1,M) RCL11640
  QXIO = STATIC(2,M) RCL11650
  OMEGAO = STATIC(3,M) RCL11660
  CONS1 = CONST1(M) RCL11670
  CONS2 = CONST2(M) RCL11680
  ICOUNT = ICOUNT + 1 RCL11690
  IF(X.EQ.1.D0) ICOUNT = 1 RCL11700
  ELSE IF ((X.LT.XI(M)).AND.(X.GT.XI(M-1))) THEN RCL11710
    DIFF = DABS(XI(M)-X) RCL11720
    THIS IF STATEMENT IS TO ACCOUNT FOR A COMPUTER INACCURACY IN RCL11730
    REPRESENTATING AND EQUATING REAL NUMBERS. RCL11740
    IF (DIFF.LT.1.D-7) THEN RCL11750
      ICOUNT = ICOUNT + 1 RCL11760
  
```

```

      IF (XI(M).EQ.1.00) ICOUNT = 1          RCL11770
END IF                                     RCL11780
DX = (X-XI(M-1))/(XI(M)-XI(M-1))          RCL11790
EXI = EPSXI(M-1) + (EPSXI(M)-EPSXI(M-1))*DX RCL11800
EXIS = EPSXIS(M-1) + (EPSXIS(M)-EPSXIS(M-1))*DX RCL11810
EETA = EPSETA(M-1) + (EPSETA(M)-EPSETA(M-1))*DX RCL11820
EETAS = EPSETS(M-1) + (EPSETS(M)-EPSETS(M-1))*DX RCL11830
EPI = EPSPI(M-1) + (EPSPI(M)-EPSPI(M-1))*DX     RCL11840
EPIS = EPSPIS(M-1) + (EPSPIS(M)-EPSPIS(M-1))*DX RCL11850
HH = HETA(M-1) + (HETA(M)-HETA(M-1))*DX        RCL11860
TLAM2 = TLAMB2(M-1) + (TLAMB2(M)-TLAMB2(M-1))*DX RCL11870
TO = STATIC(1,M-1) + (STATIC(1,M)-STATIC(1,M-1))*DX RCL11880
QXIO = STATIC(2,M-1) + (STATIC(2,M)-STATIC(2,M-1))*DX RCL11890
OMEGAO = STATIC(3,M-1) + (STATIC(3,M)-STATIC(3,M-1))*DX RCL11900
CONST1 = CONST1(M-1) + (CONST1(M)-CONST1(M-1))*DX   RCL11910
CONST2 = CONST2(M-1) + (CONST2(M)-CONST2(M-1))*DX   RCL11920
ELSE                                         RCL11930
C
      WRITE(6,*) ' ERROR OCCURED IN COUNTER, NP =',M RCL11940
END IF                                     RCL11950
RETURN                                     RCL11960
END                                         RCL11970
                                            RCL11980

```

Chapter XIII

Listing of Program RCNATIN FORTRAN A

```

C RCNATIN                                     RCN00010
C PROGRAM TO EVALUATE THE DETERMINANT OF MATRIX P FOR A RANGE OF      RCN00020
C FREQUENCIES IN ORDER TO DETERMINE APPROXIMATE VALUES OF THE NATURAL    RCN00030
C FREQUENCIES OF COMPLIANT RISERS FOR IN PLANE DYNAMICS WITH A 2-D      RCN00040
C STATIC CONFIGURATION.                                                 RCN00050
C THIS PROGRAM CAN BE USED TO DETERMINE APPROXIMATE ESTIMATES OF      RCN00060
C NATURAL FREQUENCIES TO USE AS INPUT IN THE IN-PLANE DYNAMICS PROGRAMS,RCN00070
C WHICH IMPROVE THESE INITIAL ESTIMATES.                               RCN00080
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED THROUGHOUT.     RCN00090
C WKB VERSION OF THE PROGRAM.                                         RCN00100
C*****RCN00110
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY                 RCN00120
C ALL RIGHTS RESERVED.                                              RCN00130
C*****RCN00140
C PROGRAMMER GEORGE A. KRIEZIS       JUNE 5, 1985      M.I.T.           RCN00150
C*****RCN00160
C                                         RCN00170
C DEFINITION OF DEVICES:                                              RCN00180
C DEVICE 5 : INPUT FROM TERMINAL                                     RCN00190
C DEVICE 5 : OUTPUT TO TERMINAL                                    RCN00200
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT   (LRECL=80)        RCN00210
C DEVICE 10 : INPUT FROM FILE CONTAINING APPROXIMATE N-D STATIC      RCN00220
C             SOLUTION CREATED BY RCSTAT2D (LRECL=132)                  RCN00230
C DEVICE 11 : OUTPUT TO FILE (LRECL=80)                                RCN00240
C                                         RCN00250
C DEFINITIONS OF PARAMETERS                                         RCN00260
C MNP = MAXIMUM NUMBER OF FREQUENCIES TO EVALUATE DETERMINANT.      RCN00270
C                                         RCN00280
C
C IMPLICIT REAL*8(A-H,O-Z)                                         RCN00290
C PARAMETER(MNP=300)                                              RCN00300
C CHARACTER*80 NAME                                              RCN00310
C COMMON/INPUTO/NAME                                             RCN00320
C DIMENSION FREQ(MNP),PVALUE(MNP),RANG1(MNP),RANG2(MNP)            RCN00330
C                                         RCN00340
C READ2D READS 2-D STATIC COMPLIANT RISER                         RCN00350
C SOLUTION CALCULATED FROM RCSTAT2D PROGRAM AND EVALUATES          RCN00360
C THE MAXIMUM NONDIMENSIONAL STATIC EFFECTIVE TENSION (TOMAX).     RCN00370
C                                         RCN00380
C CALL READ2D(TOMAX,VM,ICCC)                                         RCN00390
C IF(ICCC.EQ.0) STOP                                              RCN00400
C                                         RCN00410
C CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE  RCN00420
C NONDIMENSIONAL COEFFICIENTS TO BE USED FOR THE ESTIMATION OF      RCN00430
C THE NATURAL FREQUENCIES.                                         RCN00440
C                                         RCN00450
C CALL CHARAC(TOMAX,TMAXAV,TLENG,VM)                                RCN00460
C                                         RCN00470
C                                         RCN00480
C                                         RCN00490
1499 WRITE(6,1500) MNP                                         RCN00500
1500 FORMAT('/ INPUT'/' THE MAXIMUM FREQUENCY AND FREQUENCY SPACING (INRCN00510
* RAD/S) FOR'/' WHICH YOU WISH TO EVALUATE THE DETERMINANT OF MATRIRCN00520
*X P'/' MAXIMUM PERMISSIBLE NUMBER OF FREQUENCY POINTS IS = ',I3)  RCN00530
  READ(5,*) FREQH,DF                                              RCN00540
  DIV = FREQH/DF                                              RCN00550
  XP = FLOAT(MNP-1)                                            RCN00560

```

FILE: RCNATIN FORTRAN A

```

IF(DIV.GT.XP) GOTO 1499                               RCN00570
RCN00580
C APPROX:EVALUATES THE DETERMINANT OF P FOR ALL FREQUENCIES SELECTEDRCN00590
C IT USES THE FAST AND SLOW ASYMPTOTIC SOLUTIONS AND APPLIES THE      RCN00600
C BOUNDARY CONDITIONS FOR THE COMPLIANT RISER.                      RCN00610
RCN00620
C CALL APPROX(FREQ,PVALUE,TLENG,TMAXAV,TOMAX,FREQH,DF,NDIV)          RCN00630
RCN00640
C DETERMINE THE APPROXIMATE VALUES OF EACH NATURAL FREQUENCY IN THE   RCN00650
C SPECIFIED RANGE. THE DETERMINANT CHANGES SIGN AT EACH NATURAL        RCN00660
C FREQUENCY.                           RCN00670
RCN00680
C K = 0                                         RCN00690
DO 452 I=1,NDIV-1                                 RCN00700
  CHECK = PVALUE(I)*PVALUE(I+1)                   RCN00710
  IF (CHECK.LT.0.D0) THEN                         RCN00720
    K = K + 1                                     RCN00730
    RANG1(K) = FREQ(I)                          RCN00740
    RANG2(K) = FREQ(I+1)                        RCN00750
  END IF                                         RCN00760
452 CONTINUE                                     RCN00770
RCN00780
C OUTPUT THE RESULTS                            RCN00790
RCN00800
C WRITE(11,8666) NAME                           RCN00810
RCN00820
8666 FORMAT(80A)                                RCN00830
WRITE(11,36)
36 FORMAT(/' ESTIMATES OF IN-PLANE NATURAL FREQUENCIES FOR A 2-D STATRCN00840
*IC CONFIGURATION')                           RCN00850
*RCN00860
  WRITE(11,85)K
85 FORMAT(/' NUMBER OF NATURAL FREQUENCIES IN SPECIFIED RANGE: ',I3) RCN00870
RCN00880
  DO 76 I=1,K
    WRITE(11,86) I,RANG1(I),RANG2(I)           RCN00890
RCN00900
RCN00910
76 CONTINUE                                     RCN00920
RCN00930
86 FORMAT(' MODE = ',I3/' NATURAL FREQUENCY BETWEEN '/1X,D12.6,' AND '     RCN00940
*,D12.6,' RAD/SEC'/' *****)                  RCN00950
  WRITE(11,24)
24 FORMAT(/' FREQUENCY (RAD/SEC)',2X,'DETERMINANT')           RCN00960
RCN00970
  DO 89 I=1,NDIV
    WRITE(11,25)FREQ(I),PVALUE(I)               RCN00980
RCN00990
89 CONTINUE                                     RCN01000
RCN01010
25 FORMAT(5X,D12.6,4X,D12.6)                  RCN01020
STOP
END

SUBROUTINE CHARAC(TOMAX,TMAXAV,TLENG,VM1)          RCN01030
RCN01040
C SUBROUTINE CHARAC READS THE COMPLIANT RISER CHARACTERISTICS FROM      RCN01050
C DEVICE 8 AND EVALUATES ALL THE NONDIMENSIONAL COEFFICIENTS USED      RCN01060
C IN THE EVALUATION OF THE P MATRIX AND THE DIMENSIONAL NATURAL       RCN01070
C FREQUENCIES.                           RCN01080
  IMPLICIT REAL*8(A-H,O-Z)
  PARAMETER(MNP=151)
  CHARACTER*80 NAME
  COMMON/INPUT0/NAME
  COMMON/CONST/XPI,XPI2,RHOW,GRAV
RCN01100
RCN01110
RCN01120

```

```

COMMON/COEF/EPSETA(2),HZETAM,DXIM,VM,XL           RCN01130
DIMENSION EIETA(MNP),DXI(MNP),RLENG(MNP)          RCN01140
DIMENSION RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP) RCN01150
*,TMAXI(MNP),TMAZI(MNP),HZETA(MNP)                RCN01160
C
VM = VM1                                         RCN01170
READ (8,1000) NAME                            RCN01180
1000 FORMAT(8OA)                                RCN01190
      READ (8,1008) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS   RCN01200
1008 FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6))       RCN01210
      TLEN = TLEN                                     RCN01220
      DO 1502 I=1,NSEG                           RCN01230
      READ (8,1003) RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT,DXI(I),RCN01250
      *PXEITA,EA,EIETA(I),AO,EIETAS                 RCN01260
1502 CONTINUE                                     RCN01270
1003 FORMAT(5(1X,D12.6)/6(1X,D12.6))           RCN01280
C
      DO 1332 I=1,NSEG                           RCN01290
      READ(8,1333) AMAETA(I),DETA,EIXI,EIXIS,GIP,GIPS,AMAZI(I),XJZI,AJZIRCNO1310
1332 CONTINUE                                     RCN01320
1333 FORMAT(6(1X,D12.6)/3(1X,D12.6))           RCN01330
C
C      NON - DIMENSIONALIZATIONS
C
GRAV=9.81D0                                     RCN01340
WAM=WA/GRAV                                     RCN01350
WT=WA*TLEN                                      RCN01360
XPI=4.D0*Datan(1.D0)                           RCN01370
XPI2=XPI/2.D0                                    RCN01380
RHOW=1.025D3                                     RCN01390
TOMAX = TOMAX*WT                                 RCN01400
C
C      NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS
C
TOMAXL = TOMAX/TLEN                            RCN01410
TOML1  = TOMAX*TLEN                            RCN01420
TOML2  = TOML1*TLEN                            RCN01430
TLEN2  = TLEN**2                                RCN01440
TMAXAV = 0.D0                                    RCN01450
DXIM  = 0.D0                                    RCN01460
DO 2000 I=1,NSEG                               RCN01470
      RLENG(I)=RLENG(I)/TLEN                     RCN01480
      TMAZI(I) = RMASS(I) + AMAZI(I)             RCN01490
      TMAXI(I) = RMASS(I) + AMAXI(I)             RCN01500
      TMAXAV = TMAXI(I)*RLENG(I) + TMAXAV       RCN01510
      DXIM = DXIM + DXI(I)*RLENG(I)             RCN01520
2000 CONTINUE                                     RCN01530
      EPSETA(1) = EIETA(1)/TOML2                 RCN01540
      EPSETA(2) = EIETA(NSEG)/TOML2               RCN01550
      HZETAM = 0.D0                                RCN01560
      DO 4321 I=1,NSEG                           RCN01570
      HZETA(I) = TMAZI(I)/TMAXAV                 RCN01580
      HZETAM = HZETAM + HZETA(I)*RLENG(I)         RCN01590
4321 CONTINUE                                     RCN01600
      RETURN                                       RCN01610

```

FILE: RCNATIN FORTRAN A

```

END RCN01690
C
C SUBROUTINE READ2D(TOMAX,VM,ICCC) RCN01700
C THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM RCN01710
C DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION. RCN01720
C IT PROVIDES ALSO THE X AND Y DISPLACEMENTS AT THE TOP OF THE RCN01730
C COMPLIANT RISER. RCN01740
C IMPLICIT REAL*8(A-H,O-Z) RCN01750
C PARAMETER(MNP=151) RCN01760
C COMMON/STAT1/XTOP,YTOP RCN01770
C DIMENSION XCOOR(MNP),YCOOR(MNP),STATIC(MNP),XI(MNP) RCN01780
C
C STATIC(I) = STATIC EFFECTIVE TENSION TO RCN01790
C
C WRITE(6,2000) MNP RCN01800
2000 FORMAT(' MNP=',I3) RCN01810
C
C ICCC = 1 RCN01820
READ(10,36459) NP,VM RCN01830
36459 FORMAT(1X,I3,1X,D12.6) RCN01840
WRITE(6,2311) NP,VM RCN01850
2311 FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
*' NP =',I3/ RCN01860
*' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S ,VM =',D12.6) RCN01870
C
C IF((NP.LT.4).OR.(NP.GT.MNP)) THEN RCN01880
    ICCC=0 RCN01890
    WRITE(6,12439) RCN01900
12439 FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS') RCN01910
    RETURN RCN01920
    ENDIF RCN01930
C
C IF (VM.EQ.0.D0) THEN RCN01940
    WRITE(6,1240) RCN01950
1240 FORMAT(' PROGRAM IS NOT VALID FOR ZERO MEAN CURRENT SPEED'/
*' RUN STOPS') RCN01960
    ICCC = 0 RCN01970
    RETURN RCN01980
    END IF RCN01990
C
C READING FROM DEVICE 10 RCN02000
C
C DO 1021 I=1,NP RCN02010
    READ(10,1033) X,STATIC(I),STATI2,STATI3,STAT4,XCOOR(I),YCOOR(I),
*STRARC,TENSI,VLOCKXI RCN02020
    XI(I) = X RCN02030
1021 CONTINUE RCN02040
1033 FORMAT(10(1X,D12.6)) RCN02050
C
C EVALUATE RISER TOP X AND Y COORDINATES RCN02060
    XTOP = XCOOR(NP) RCN02070
    YTOP = YCOOR(NP) RCN02080
C
C EVALUATE MAXIMUM STATIC EFFECTIVE TENSION RCN02090
    TOMAX=DMAX1(STATIC(1),STATIC(2)) RCN02100
    DO 9859 I=3,NP*
        TOMAX = DMAX1(TOMAX,STATIC(I)) RCN02210
9859 CONTINUE RCN02220
RCN02230
RCN02240

```

FILE: RCNATIN FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

C          WRITE(6,1654) TOMAX
1654    FORMAT(' 2-D STATIC SOLUTION SUCCESFULLY READ'/' MAXIMUM STATIC EFRCN02270
          *EFFECTIVE TENSION/WA*L = ',D10.4) RCN02280
C          RETURN RCN02290
C          END RCN02300
C          SUBROUTINE APPROX(FREQ,PVALUE,TLENG,TMAXAV,TOMAX,FREQH,DF,NDIV) RCN02310
C          THIS SUBROUTINE EVALUATES THE DETERMINANT OF MATRIX P AT ALL THE RCN02320
C          SPECIFIED FREQUENCY POINTS IN ORDER TO LOCATE REGIONS OF CHANGING RCN02330
C          SIGN. RCN02340
C          IMPLICIT REAL*8(A-H,O-Z) RCN02350
C          PARAMETER(MNP=300) RCN02360
C          COMMON/CONST/XPI,XPI2,RHOW,GRAV RCN02370
C          COMMON/COEF/EPSETA(2),HZETAM,DXIM,VM,XL RCN02380
C          COMMON/OMA/OMAPP(2),PHIOO(2) RCN02390
C          DIMENSION FVEC(2),XS(2),WAS(19),WKSPCE(6) RCN02400
C          DIMENSION PVALUE(MNP),FREQ(MNP),A(6,6),P(6,6) RCN02410
C          EXTERNAL FCNS RCN02420
C          DETERMINATION OF THE CABLE APPROXIMATION OF 2-D STATIC SOLUTION. RCN02430
C          THE POWELL HYBRID METHOD IS USED (SEE PROGRAM RCSTAT2D FOR RCN02440
C          DETAILS FOR THE SLOW SOLUTION) RCN02450
C          EVALUATE K RCN02460
C          WRITE(6,1781) RCN02470
1781    FORMAT(' INPUT NORMAL DRAG COEFFICIENT USED IN STATIC SOLUTION') RCN02480
          READ(5,*)CD RCN02490
          XK = 0.5D0*RHOW*DXIM*CD*VM*DABS(VM)*TLENG RCN02500
          WRITE(6,8881) VM,CD RCN02510
8881    FORMAT(' MEAN CURRENT SPEED =',D12.6,' NORMAL DRAG COEFFICIENT =',RCN02520
          *D12.6) RCN02530
C          2601 WRITE(6,1501) RCN02540
1501    FORMAT(' INPUT ESTIMATES OF LAMBDA AND C'/
          *' LAMBDA MUST HAVE THE SIGN OF MEAN CURRENT SPEED'/
          *' FOR SMALL XTOP A GOOD ESTIMATE OF C IS -0.5*LAMBDA') RCN02550
          READ(5,*) XL,XC RCN02560
C          XS(1) = XL RCN02570
          XS(2) = XC RCN02580
          XTOL = 1.D-8 RCN02590
          NS = 2 RCN02600
          LWA = 19 RCN02610
C          CALL NAG SUBROUTINE CG5NBF TO DETERMINE THE CABLE APPROXIMATION RCN02620
          IFAIL = 0 RCN02630
          CALL CG5NBF(FCNS,NS,XS,FVEC,XTOL,WAS,LWA,IFAIL) RCN02640
C          WRITE(6,1356) IFAIL,XS(1),XS(2) RCN02650
1356    FORMAT(' SOLUTION FOR CABLE ...'/' IFAIL =',I2/
          *' LAMBDA =',D12.6/
          *' C      =',D12.6) RCN02660
C          WRITE(6,2500) FVEC(1),FVEC(2) RCN02670
2500    FORMAT(' VALUE OF FUNCTION FOR HORIZONTAL DISPLACEMENT =',D12.6/
          *' VALUE OF FUNCTION FOR VERTICAL DISPLACEMENT =',D12.6/) RCN02680

```

FILE: RCNATIN FORTRAN A

```

*' DO YOU WISH TO TRY A DIFFERENT INITIAL LAMDA, OR C.,'/          RCN02810
*' IF YES INPUT 1')          RCN02820
READ(5,*) IPP          RCN02830
IF (IPP.EQ.1) GO TO 2601          RCN02840
IF (IFAIL.NE.0) GO TO 2601          RCN02850
IF (IFAIL.NE.0) GO TO 2601          RCN02860
C          RCN02870
    XL = XS(1)          RCN02880
    XC = XS(2)          RCN02890
    TE = XK/XL          RCN02900
    WRITE(6,1654) XL,XC,TE          RCN02910
1654 FORMAT(' CABLE APPROXIMATION '/          RCN02920
*' LAMBDA =',D12.6,' C =',D12.6,' TENSION =',D12.6)          RCN02930
C          RCN02940
C DETERMINE VALUES OF PHI00 AND OMEGA ETA AT THE TOP AND BOTTOM OF      RCN02950
C THE RISER USING CABLE APPROXIMATION.          RCN02960
C          RCN02970
    S = 0.D0          RCN02980
    PHI00(1) = PHI0(S,XL,XC)          RCN02990
    OMAPP(1)=XL*DSIN(PHI00(1))**2          RCN03000
C          RCN03010
    S = 1.D0          RCN03020
    PHI00(2) = PHI0(S,XL,XC)          RCN03030
    OMAPP(2)=XL*DSIN(PHI00(2))**2          RCN03040
C          RCN03050
C CALCULATE FUNCTIONS F THAT DETERMINE THE SLOW DYNAMIC SOLUTION      RCN03060
C AT THE TOP AND BOTTOM OF THE RISER.          RCN03070
CALL FUNC(PHI00,HZETAM,XL)          RCN03080
C          RCN03090
C EVALUATE THE P MATRIX FOR EACH FREQUENCY          RCN03100
C          RCN03110
    IOS = 0          RCN03120
C EVALUATE NUMBER OF INCREMENTS          RCN03130
    NDIV = NINT(FREQH/DF)+ 1          RCN03140
C INITIAL FREQUENCY FOR EVALUATION IS 0.01 RAD/SEC          RCN03150
C          RCN03160
    SIGMA = 0.01D0*TLENG*DSQRT(TMAXAV/TOMAX)          RCN03170
C          RCN03180
    DS = DF*TLENG*DSQRT(TMAXAV/TOMAX)          RCN03190
C          RCN03200
    N = 6          RCN03210
    DO 765 L=1,NDIV          RCN03220
    FREQ(L) = .01 + (L-1)*DF          RCN03230
C          RCN03240
EVALUATE MATRIX P          RCN03250
CALL XMATR(P,SIGMA,IOS)          RCN03260
C          RCN03270
    IOS = 1          RCN03280
C FIND ITS DETERMINANT USING NAG SUBROUTINE FO3AAF (CRGUT          RCN03290
C FACTORISATION METHOD).          RCN03300
    DO 652 I=1,6          RCN03310
        DO 652 K=1,6          RCN03320
        A(I,K) = P(I,K)          RCN03330
652 CONTINUE          RCN03340
    IFAI = 1          RCN03350
    CALL FO3AAF(A,N,N,DET,WKSPCE,IFAI)
    SIGMA = SIGMA + DS
    PVALUE(L) = DET
765 CONTINUE
RETURN
END

C          RCN03360
C          SUBROUTINE FCNS(NS,XS,FVEC,IFLAG)
C THIS SUBROUTINE EVALUATES THE FUNCTIONS FOR THE HORIZONTAL AND

```

```

C      VERTICAL DISPLACEMENT OF THE CABLE SOLUTION. IT IS USED BY NAG      RCN03370
C      SUBROUTINE C05NBF.                                                 RCN03380
C      IMPLICIT REAL*8(A-H,O-Z)                                         RCN03390
C      COMMON/STAT1/XTOP,YTOP                                         RCN03400
C      DIMENSION XS(NS),FVEC(NS)                                       RCN03410
C      LAMBDA AND C ARE UNKNOWNs                                     RCN03420
C      XL = XS(1)                                                 RCN03430
C      XC = XS(2)                                                 RCN03440
C      XLINV = 1.D0/XL                                              RCN03450
C      PHIO0 = PHIO(0.D0,XL,XC)                                         RCN03460
C      PHIO1 = PHIO(1.D0,XL,XC)                                         RCN03470
C
C      F1 = XLINV*(1.D0/DSIN(PHIO0)-1.D0/DSIN(PHIO1))                RCN03480
C      F1 = F1-XTOP                                               RCN03490
C      FVEC(1) = F1                                              RCN03500
C
C      TT1 = DTAN(PHIO1/2.D0)                                         RCN03510
C      TTO = DTAN(PHIO0/2.D0)                                         RCN03520
C      F2 = XLINV*DLOG(TT1/TTO)                                         RCN03530
C      F2 = F2-YTOP                                              RCN03540
C      FVEC(2) = F2                                              RCN03550
C      RETURN                                                 RCN03560
C      END                                                       RCN03570
C
C      FUNCTION PHIO(S,XL,XC)                                         RCN03580
C      THIS FUNCTION EVALUATES PHIO0 OF THE CABLE SOLUTION AS A FUNCTION RCN03590
C      OF S, LAMBDA AND C.                                         RCN03600
C      IMPLICIT REAL*8(A-H,O-Z)                                         RCN03610
C      COMMON/CONST/XPI,XPI2,RHOW,GRAV                           RCN03620
C
C      XVAR = -(XL*S+XC)                                         RCN03630
C      PHIO = DATAN(1.D0/XVAR)                                         RCN03640
C      IF (XVAR.LT.0.D0) THEN                                         RCN03650
C          PHIO = PHIO + XPI                                         RCN03660
C      END IF                                                 RCN03670
C      RETURN                                                 RCN03680
C      END                                                       RCN03690
C
C      SUBROUTINE XMATTR(P,SIGMA,IOS)                                RCN03700
C      THIS SUBROUTINE EVALUATES THE 6X6 MATRIX P ELEMENTS           RCN03710
C      AT FREQUENCY SIGMA.                                         RCN03720
C      IMPLICIT REAL*8(A-H,O-Z)                                         RCN03730
C      COMMON/SLOW/F1(2),F2(2),F3(2),F4(2),F5(2),F6(2)             RCN03740
C      COMMON/COEF/EPSETA(2),HZETAM,DXIM,VM,XL                      RCN03750
C      COMMON/OMA/OMAPP(2),PHIO0(2)                                 RCN03760
C      DIMENSION P(6,6)                                             RCN03770
C      EVALUATE ALL CONSTANT ELEMENTS (NOT A FUNCTION OF SIGMA)     RCN03780
C      IF (IOS.EQ.0) THEN                                         RCN03790
C          DO 881 I=1,6                                         RCN03800
C          DO 881 K=1,6                                         RCN03810
C              P(I,K) = 0.D0                                         RCN03820
C
C881    CONTINUE
C          P(1,2) = DSIN(PHIO0(1))**2                            RCN03830
C          P(1,3) = 1.D0                                           RCN03840
C          P(1,5) = F1(1)                                         RCN03850
C          P(1,6) = F2(1)                                         RCN03860

```

FILE: RCNATIN FORTRAN A

```

P(2,2) = DSIN(2.D0*PHIOO(1))          RCN03930
P(2,3) = -1.D0/(DSQRT(EPSETA(1))*OMAPP(1))  RCN03940
P(2,5) = F3(1)                         RCN03950
P(2,6) = F4(1)                         RCN03960
P(3,3) = 1.D0/(EPSETA(1)*OMAPP(1))      RCN03970
P(3,5) = F5(1)                         RCN03980
P(3,6) = F6(1)                         RCN03990
P(4,4) = 1.D0                           RCN04000
P(4,5) = F1(2)                          RCN04010
P(4,6) = F2(2)                          RCN04020
P(5,4) = 1.D0/(DSQRT(EPSETA(2))*OMAPP(2))  RCN04030
P(5,5) = F3(2)                          RCN04040
P(5,6) = F4(2)                          RCN04050
P(6,4) = 1.D0/(EPSETA(2)*OMAPP(2))      RCN04060
P(6,5) = F5(2)                          RCN04070
P(6,6) = F6(2)                          RCN04080
END IF                                    RCN04090
C EVALUATE ELEMENTS THAT ARE A FUNCTION OF SIGMA
COSS = DCOS(SIGMA)                      RCN04100
SINS = DSIN(SIGMA)                      RCN04110
P(2,1) = SIGMA/XL                       RCN04120
P(3,1) = SIGMA*DSIN(PHIOO(1)*2.D0)       RCN04130
P(3,2) = - SIGMA**2/XL + 2.D0*OMAPP(1)*DCOS(2.D0*PHIOO(1))  RCN04140
P(4,1) = SINS*DSIN(PHIOO(2))**2          RCN04150
P(4,2) = COSS*DSIN(PHIOO(2))**2          RCN04160
P(5,1) = SINS*DSIN(PHIOO(2)*2.D0) + SIGMA*COSS/XL        RCN04170
P(5,2) = COSS*DSIN(PHIOO(2)*2.D0) - SIGMA*SINS/XL        RCN04180
C P(6,1) = (2.D0*OMAPP(2)*DCOS(PHIOO(2)*2.D0) - SIGMA**2/XL)*SINS  RCN04190
P(6,1) = P(6,1) + SIGMA*COSS*DSIN(PHIOO(2)*2.D0)          RCN04200
C P(6,2) = (2.D0*OMAPP(2)*DCOS(PHIOO(2)*2.D0) - SIGMA**2/XL)*COSS  RCN04210
P(6,2) = P(6,2) - SIGMA*SINS*DSIN(PHIOO(2)*2.D0)          RCN04220
RETURN                                     RCN04230
END                                         RCN04240
C SUBROUTINE FUNC(PHIOO,HZETAM,XL)
C THIS SUBROUTINE EVALUATES THE FUNCTIONS THAT RESULT FROM THE
C APPROXIMATE SLOW SOLUTION FOR IN-PLANE DYNAMICS OF COMPLIANT
C RISERS. THIS SUBROUTINE EVALUATES THESE FUNCTIONS AT THE TWO ENDS
C OF THE RISER.                                     RCN04250
IMPLICIT REAL*8(A-H,O-Z)                  RCN04260
COMMON/SLOW/F1(2),F2(2),F3(2),F4(2),F5(2),F6(2)    RCN04270
DIMENSION PHIOO(2)                         RCN04280
XP12 = 2.D0*DATAN(1.D0)                    RCN04290
H = HZETAM                                 RCN04300
H2 = H**2                                   RCN04310
DO 537 I=1,2
  COSPHI = DCOS(PHIOO(I))                 RCN04320
  SINPHI = DSIN(PHIOO(I))                 RCN04330
  COS2PH = DCOS(2.D0*PHIOO(I))            RCN04340
  COSPH2 = COSPHI**2                     RCN04350
  SINPH2 = SINPHI**2                     RCN04360
C F1(I) = COSPHI*(1.D0+(H-1.D0)*COSPH2/6.D0*(3.D0+H)/20.D0*COSPH2)RCN04470
F1(I) = F1(I) + COSPHI*(H+1.D0)*COSPH2/6.D0*RCN04480

```

FILE: RCNATIN FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

C
F2(I) = 1.D0 - H2 - H*(H-1.D0)/2.D0*COSPH2          RCN04490
F2(I) = F2(I) + H2*(SINPHI + COSPHI*(XPI2 - PHI00(I)))  RCN04500
C
F3(I) ==SINPHI*(H-1.D0)/2.D0*COSPH2*(1.D0+(3.D0+H)*COSPH2/12.D0)RCN04510
F3(I) = F3(I) - SINPHI                                RCN04520
C
F4(I) = SINPHI*(H*(H-1.D0)*COSPHI - H2*(XPI2 - PHI00(I)))  RCN04530
RCN04540
RCN04550
C
F5(I) = 2.D0 - H + (H-1.D0)*(6.D0-H)*COSPH2/6.D0      RCN04560
F5(I) = F5(I) + 5.D0/24.D0*(H-1.D0)*(H+3.D0)*COSPH2**2  RCN04570
F5(I) = -F5(I)*XL*SINPH2*COSPHI                      RCN04580
C
F6(I) = H*(H-1)*COS2PH + H2*SINPHI - H2*(XPI2 - PHI00(I))*COSPHI   RCN04590
F6(I) = XL*SINPH2*F6(I)                                RCN04600
RCN04610
RCN04620
RCN04630
RCN04640
RCN04650
RCN04660
537 CONTINUE
RETURN
END

```

200

Chapter XIV

Listing of Program RCLINDY1 FORTRAN A

C RCLINDY1 RCL00010
C THIS PROGRAM CALCULATES THE IN-PLANE LINEAR DYNAMIC RESPONSE OF A RCL00020
C COMPLIANT RISER WITH A 2-D STATIC CONFIGURATION BASED ON AN RCL00030
C EMBEDDING TECHNIQUE. THE PROGRAM PROVIDES AN INITIAL ANALYTIC RCL00040
C APPROXIMATION WHICH IT SUBSEQUENTLY IMPROVES USING MODIFIED NEWTON'S RCL00050
C ITERATION AND A NON-UNIFORM GRID FINITE DIFFERENCE METHOD. RCL00060
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED. RCL00070
C***** RCL00080
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY RCL00090
C ALL RIGHTS RESERVED. RCL00100
C***** RCL00110
C PROGRAMMER GEORGE A. KRIEZIS JUNE 5, 1985 M.I.T. RCL00120
C***** RCL00130
C
C DEFINITION OF DEVICES: RCL00140
C DEVICE 5 : INPUT FROM TERMINAL RCL00150
C DEVICE 6 : OUTPUT TO TERMINAL RCL00160
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT (LRECL=80) RCL00170
C DEVICE 9 : COMPLETE OUTPUT TO FILE (LRECL=132) RCL00180
C DEVICE 10 : INPUT FROM FILE CONTAINING N-D STATIC 2-D RCL00190
C SOLUTION CREATED BY RCSTAT2D (LRECL=132) RCL00200
C DEVICE 11 : OUTPUT TO FILE FOR PLOTS OR INPUT TO A NEW RUN RCL00210
C OF RCLINDY1 (LRECL=117) RCL00220
C DEVICE 12 : INPUT FROM FILE CONTAINING APPROXIMATE N-D SOLUTION RCL00230
C CREATED BY A PREVIOUS RUN OF RCLINDY1 (LRECL=117) RCL00240
C
C COMMON BLOCK CONTENTS (OVERALL REFERENCE): RCL00250
C SOLUT = INITIAL APPROXIMATION AND SOLUTION MATRICES RCL00270
C STAT = STATIC COMPLIANT RISER SOLUTION. (EFFECTIVE TENSION, SHEAR RCL00280
C FORCE, OMEGA ETA, PHI0, MAXIMUM STATIC EFFECTIVE TENSION, RCL00290
C MEAN CURRENT VELOCITY AND NUMBER OF STATIC DIVISION POINTS). RCL00300
C STAT1 = X AND Y DISPLACEMENTS AT TOP OF RISER. RCL00310
C STAT2 = FUNCTIONS OF STATIC RESULTS. RCL00320
C INPUT0 = OUTPUT FILE HEADING RCL00330
C INPUT1 = RISER LENGTH, AVERAGE AND TOTAL WEIGHT AND NUMBER OF SEGMENTS RCL00350
C INPUT2 = RISER TUBE INTERIOR CHARACTERISTICS RCL00360
C INPUTL = RISER SEGMENTS LENGTH RCL00370
C INPUT3 = WEIGHT, STIFFNESSES AND STIFFNESS DERIVATIVES. RCL00380
C INPUT4 = CHARACTERISTIC DIAMETERS AND PERIMETER OF RISER TUBE. RCL00390
C INPUT5 = MASS AND ADDED MASS CHARACTERISTICS IN THREE DIRECTIONS RCL00400
C INPUT6 = RISER POLAR AND ADDED POLAR MOMENT OF INERTIA. RCL00410
C COEF = NONDIMENSIONAL RISER CHARACTERISTICS USED IN THE EQUATIONS RCL00420
C COEF1 = NONDIMENSIONAL RISER CHARACTERISTICS AT A SPECIFIC POINT RCL00430
C CONST = PROBLEM CONSTANTS (PI, GRAVITY, WATER DENSITY) RCL00440
C BOUNDA = BOUNDARY CONDITION FOR OMEGA ETA AT S=0 RCL00450
C COUN = INTEGER COUNTING VARIABLE TO DETERMINE EACH DIVISION POINT RCL00460
C MATRIX = MATRIX P FOR ESTIMATION OF THE APPROXIMATE NATURAL RCL00470
C FREQUENCIES. RCL00480
C SLOW = APPROXIMATE SLOW SOLUTION VECTORS RCL00490
C OMA = LAMBDA, PHI0 AND OMEGA ETA FOR THE CABLE APPROXIMATION RCL00500
C
IMPLICIT REAL*8(A-H,O-Z) RCL00510
PARAMETER(N=7,MNP=151,NA=4,IY=7) RCL00520
PARAMETER(LWORK=MNP*(3*N**2+6*N+2)+4*N**2+3*N) RCL00530
PARAMETER(LIWORK=MNP*(2*N+1)+N) RCL00540
DIMENSION WORK(LWORK),IWORK(LIWORK) RCL00550
RCL00560

```

COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL00570
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKI(MNP),TOMAX,VM,NPI RCL00580
COMMON/COUN/ICOUNT RCL00590
C EXTERNAL SUBROUTINES USED BY NAG LIBRARY RCL00600
EXTERNAL FON,G,JACEPS,JACGEP,JACOBF,JACOBG RCL00610
DATA NOUT /6/ RCL00620
RCL00630
C READ2D READS 2-D STATIC SOLUTION CALCULATED FROM RCSTAT2D PROGRAM. RCL00640
C IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC EFFECTIVE TENSION RCL00650
C CALL READ2D(NPI) RCL00660
RCL00670
C CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE RCL00680
C NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE EQUATIONS. RCL00690
C RCL00700
CALL CHARAC(TLENG,IC,NPI) RCL00710
IF(IC.EQ.0) STOP RCL00720
RCL00730
C DEFINITIONS OF PARAMETERS ... RCL00740
C N = NUMBER OF EQUATIONS TO BE SOLVED BY DO2RAF RCL00750
C IY = NUMBER OF VARIABLES RCL00760
C MNP= MAXIMUM NUMBER OF PERMITTED POINTS IN THE NON-UNIFORM RCL00770
C F.D. MESH ( MNP >= 32) RCL00780
C IF(MNP.LT.32) THEN RCL00790
      WRITE(6,1257) MNP RCL00800
1257  FORMAT(' MNP = ',I3,' IS NOT PERMISSIBLE') RCL00810
      STOP RCL00820
      ENDIF RCL00830
C NUMBER OF BOUNDARY CONDITIONS AT S=0 RCL00840
NUMBEG=4 RCL00850
C NUMBER OF MIXED BOUNDARY CONDITIONS RCL00860
NUMMIX=0 RCL00870
C PARAMETER DECLARING THAT INITIAL APPROXIMATION IS USER SPECIFIED RCL00880
INIT=1 RCL00890
C PARAMETER DECLARING THAT JACOBIANS ARE USER SPECIFIED RCL00900
IJAC=1 RCL00910
C PARAMETER CONTROLLING MONITORING OF CALCULATIONS RCL00920
IFAIL=111 RCL00930
RCL00940
C RCL00950
C RCL00960
1499 WRITE(6,1500) RCL00970
1500 FORMAT(' INPUT 1 IF YOU WISH TO READ INITIAL APPROXIMATION,/' RCL00980
*'          CREATED BY A PREVIOUS RUN OF THIS PROGRAM'/' RCL00990
*' ELSE INPUT 0 (AN INITIAL APPROXIMATION WILL BE CALCULATED LOCALLY) RCL01000
*Y*)') RCL01010
      READ(5,*) IEXIST RCL01020
      IF((IEEXIST.NE.1).AND.(IEEXIST.NE.0)) GOTO 1499 RCL01030
      IF(IEEXIST.EQ.1) THEN RCL01040
RCL01050
C READAS : READS APPROXIMATE SOLUTION FROM DEVICE 10 FROM A RCL01060
C PREVIOUS RUN OF RCLINDY1 (DELEPS = 1.0) RCL01070
C IT ALSO PROVIDES ICCC,NP, TOL= TOLERANCE OF ITERATIONS RCL01080
C THIS ALTERNATIVE MAY, FOR EXAMPLE, BE USED IF GREATER ACCURACY RCL01090
C IS REQUIRED FOR THE PROBLEM SOLUTION. RCL01100
RCL01110
CALL READAS(ICCC,NP,TOL) RCL01120

```

```

C IF THE NUMBER OF POINTS FROM THIS SOLUTION IS GREATER THAN THE RCL01130
C STATIC SOLUTION POINTS, INTERPOLATE ALL CHARACTERISTICS TO THE RCL01140
C NEW NUMBER OF POINTS. RCL01150
C ASSUMPTION : NP ALWAYS GREATER OR EQUAL TO NPI RCL01160
C THIS IS VALID IF THE SAME STATIC SOLUTION WAS USED TO OBTAIN THE RCL01170
C APPROXIMATE SOLUTION IN DEVICE 10. RCL01180
C
C IF (NP.GE.NPI) THEN RCL01190
C   CALL INTERP(NP)
C   NPI=NP
C ELSE IF (NP.LT.NPI) THEN RCL01200
C   WRITE(6,314) RCL01210
C   314   FORMAT(' APPROXIMATE SOLUTION IN DEVICE 10 WAS NOT OBTAINED',/ RCL01220
C          * ' FROM THE SAME STATIC SOLUTION, PROGRAM STOPS') RCL01230
C          STOP RCL01240
C          END IF RCL01250
C          DELEPS = 1.0D0 RCL01260
C
C ELSE RCL01270
C
C   INCREMENT OF CONTINUATION PARAMETER RCL01280
7891  WRITE(6,7890) RCL01290
C   7890  FORMAT(' INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS',/ RCL01300
C          *' IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.0D0',/ RCL01310
C          *' IF CONTINUATION IS REQUIRED THEN 0.0D0 < DELEPS < 1.0D0',/ RCL01320
C          *' RECOMMENDATION :',/ RCL01330
C          *' USUALLY DELEPS = 0.1D0 WILL SUFFICE',/ RCL01340
C          *' FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM',/ RCL01350
C          *' A SMALLER VALUE OF DELEPS, E.G. 0.05D0 OR 0.025D0 MIGHT BE',/ RCL01360
C          *' NECESSARY')
C          READ(5,*) DELEPS RCL01370
C          IF((DELEPS.GT.1.0D0).OR.(DELEPS.LE.0.0D0)) THEN RCL01380
C            WRITE(6,7892) RCL01390
7892  FORMAT(' 0. < DELEPS <= 1.')
C            GOTO 7891 RCL01400
C            ENDIF RCL01410
C
C APPROX : PROVIDES INITIAL APPROXIMATE SOLUTION FOR EPS=0. RCL01420
C           IT ALSO PROVIDES ICCC,NP, TOL. RCL01430
C
C CALL APPROX(TLENG,ICCC,NPI,TOL) RCL01440
C   NP = NPI RCL01450
C
C   ENDIF RCL01460
C   SET COUNTING VARIABLE RCL01470
C   ICOUNT = 1 RCL01480
C   IF(ICCC.EQ.0) STOP RCL01490
C
C CALL X04AAF(1,NOUT) RCL01500
C CALL X04ABF(1,NOUT) RCL01510
C
C CALL NAG SUBROUTINE D02RAF TO SOLVE THE TWO-POINT BOUNDARY VALUE RCL01520
C PROBLEM USING A DEFERRED CORRECTION TECHNIQUE AND NEWTON'S RCL01530
C ITERATION. RCL01540

```

```

      CALL D02RAF(N,MNP,NP,NUMBEG,NUMMIX,TOL,INIT,X,Y,IY,ABT,FCN,G,IJAC,RCL01690
      *JACOBF,JACOBG,DELEPS,JACEPS,JACGEP,WORK,LWORK,IWORK,LIWORK,IFAIL) RCL01700
C                                         RCL01710
      WRITE(6,9000) IFAIL
9000  FORMAT(' IFAIL =',I3)                         RCL01720
C                                         RCL01730
C                                         RCL01740
C                                         RCL01750
C                                         RCL01760
C                                         RCL01770
      IF((IFAIL.EQ.0).OR.(IFAIL.EQ.4)) THEN          RCL01780
         CALL OUTPUT(NP)                            RCL01790
      ENDIF
C                                         RCL01800
      STOP
      END                                         RCL01810
                                         RCL01820
                                         RCL01830
                                         RCL01840
SUBROUTINE CHARAC(TLENG,IC,NP)
C THIS SUBROUTINE READS THE COMPLIANT RISER CHARACTERISTICS AND
C EVALUATES THE NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE
C GOVERNING EQUATIONS.
C                                         RCL01850
      IMPLICIT REAL*8(A-H,O-Z)                      RCL01860
      PARAMETER(MNP=151,NA=4,N=7)                    RCL01870
      CHARACTER*80 NAME
      COMMON/CONST/XPI,XPI2,RHOW,GRAV
      COMMON/INPUT/NAME
      COMMON/INPUT1/TLEN,WA,WT,NSEG
      COMMON/INPUT2/RHOO,AI,CFLUID,PRESS,AO(MNP)
      COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)
      COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP)RCL01960
      *,EIXIS(MNP),GIP(MNP),GIPS(MNP)               RCL01970
      COMMON/INPUT4/DXI(MNP),PXIETA(MNP),DETA(MNP),DXIETA(MNP)           RCL01980
      COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(RCL01990
      *MNP),TMAXI(MNP),TMAZI(MNP)                  RCL02000
      COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP)             RCL02010
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI       RCL02020
      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL02030
      *ZETAM,TMAXAV,DXIM
      COMMON/STAT2/CONST1(MNP),CONST2(MNP)
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
      DIMENSION TENO(MNP),QXO(MNP),TOS(MNP),QXIOS(MNP)
C                                         RCL02040
C                                         RCL02050
C                                         RCL02060
C                                         RCL02070
C                                         RCL02080
C                                         RCL02090
C                                         RCL02100
      READ RISER CHARACTERISTICS FROM DEVICE 8
      READ (8,1000) NAME
1000  FORMAT(80A)
      READ (8,1008) NSEG,TLEN,WA,RHOO,AI,CFLUID,PRESS
1008  FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6))
      TLENG = TLEN
      DO 1502 I=1,NSEG
      READ (8,1003) RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXI(IRCL02160
      *,PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I)
1502  CONTINUE
1003  FORMAT(5(1X,D12.6)/6(1X,D12.6))
C                                         RCL02170
      DO 1332 I=1,NSEG
      READ(8,1333) AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),AMARCL02220
      *ZI(I),XJZI(I),AJZI(I)
1332  CONTINUE
                                         RCL02230
                                         RCL02240

```

FILE: RCLINDY1 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

1333 FORMAT(6(1X,D12.6)/3(1X,D12.6))          RCL02250
C                                              RCL02260
    IC=1                                         RCL02270
C                                              RCL02280
7650  WRITE(6,7651)                           RCL02290
7651  FORMAT(' DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS'/
*' IF YES INPUT 1 , IF NO INPUT 0')
    READ(5,*) IPRINT
    IF((IPRINT.NE.1).AND.(IPRINT.NE.0)) GOTO 7650
    IF(IPRINT.EQ.1) THEN
C
        WRITE(6,1000) NAME
        WRITE(6,2500)
2500  FORMAT(' NSEG      TLEN           WA         RHOO       AI'      RCL02300
* CFLUID      PRESS')                         RCL02310
        WRITE(6,2001) NSEG,TLEN,WA,RHOO,AI,CFLUID,PRESS RCL02320
2001  FORMAT(1X,I3,6(1X,D12.6))               RCL02330
        WRITE(6,3400)
3400  FORMAT(' I      RLENG          RMASS        RMASST     AMAXI'      RCL02340
* WEIGHT'   /'      DKI          PXIETA      EA        EIETA RCL02350
*          AO          EIETAS')                RCL02360
        DO 3002 I=1,NSEG
        WRITE(6,3003) I,RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXIRCL02470
* (I),PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I)
3002  CONTINUE
3003  FORMAT(1X,I3,5(1X,D12.6)/4X,6(1X,D12.6))
C
        WRITE(6,1334)
1334  FORMAT(' I      AMAETA        DETA        EIXI       EIXIS'      RCL02480
*      GIP          GIPS'/'        AMAZI       JZI        AJZI' ) RCL02490
        DO 1335 I=1,NSEG
        WRITE(6,1336) I,AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),RCL02500
*AMAZI(I),XJZI(I),AJZI(I)
1335  CONTINUE
1336  FORMAT(1X,I3,6(1X,D12.6)/4X,3(1X,D12.6))
C
1701  WRITE(6,1700)
1700  FORMAT(' INPUT 1 IF RISER DATA ARE CORRECT'/' INPUT 0 TO STOP')
    READ(5,*) IC
    IF((IC.NE.0).AND.(IC.NE.1)) GOTO 1701
    IF(IC.EQ.0) RETURN .
C
        ENDIF
C
C      NON - DIMENSIONALIZATIONS
C
        GRAV=9.81DO
        WAM=WA/GRAV
        WT=WA*TLEN
        XPI=4.DO*DATAN(1.DO)
        XPI2=XPI/2.DO
        RHOW=1.025D3
C
C      NONDIMENSIONALIZE CORRECTLY QXIO, TO FROM 2-D STATIC SOLUTION
C
        TOMAX = TOMAX*WT

```

FILE: RCLINDY1 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

TND = WT/TOMAX                               RCL02810
DO 229 I=1,NP
  STATIC(1,I) = STATIC(1,I)*TND             RCL02820
  STATIC(2,I) = STATIC(2,I)*TND             RCL02830
229 CONTINUE                                  RCL02840
C
C      NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS   RCL02850
C
TOMAXL = TOMAX/TLEN                         RCL02860
TOML1  = TOMAX*TLEN                         RCL02870
TOML2  = TOML1*TLEN                         RCL02880
TLEN2   = TLEN**2                            RCL02890
TMAXAV = 0.DO                                RCL02900
DXIM   = 0.DO                                RCL02910
DO 2000 I=1,NSEG
  RLENG(I)=RLENG(I)/TLEN                   RCL02920
  WEIGHT(I)=WEIGHT(I)/TOMAXL                RCL02930
  EOM(I) = TOMAX/EA(I)                      RCL02940
  EPSETA(I) = EIETA(I)/TOML2                RCL02950
  EPSETS(I) = EIETAS(I)/TOML1               RCL02960
  TJZI(I) = XJZI(I) + AJZI(I)              RCL02970
  TMAZI(I) = RMASS(I) + AMAZI(I)            RCL02980
  TMAXI(I) = RMASS(I) + AMAXI(I)            RCL02990
  TMAXAV = TMAXI(I)*RLENG(I) + TMAXAV     RCL03000
  DXIM = DXIM + DXI(I)*RLENG(I)           RCL03010
  DXIETA(I) = DXI(I)-DETA(I)              RCL03020
2000 CONTINUE                                  RCL03030
  HZETAM = 0.DO                                RCL03040
  DO 4321 I=1,NSEG
    HZETA(I) = TMAZI(I)/TMAXAV             RCL03050
    HZETAM = HZETAM + HZETA(I)*RLENG(I)    RCL03060
    HXI(I) = TMAXI(I)/TMAXAV               RCL03070
4321 CONTINUE                                  RCL03080
C
C      CALCULATE DERIVATIVES OF STATIC QUANTITIES          RCL03090
C
DO 737 I=1,NP
  TENO(I) = STATIC(1,I)                      RCL03100
  QXO(I) = STATIC(2,I)                      RCL03110
737 CONTINUE                                  RCL03120
  CALL DER1(TENO,XI,TOS,NP)                 RCL03130
  CALL DER1(QXO,XI,QXIOS,NP)                RCL03140
C
C      EVALUATE FUNCTIONS OF STATIC RESULTS               RCL03150
C
DO 56 I=1,NP
  CONST1(I) = TOS(I) - STATIC(2,I)*STATIC(3,I)  RCL03160
  CONST2(I) = QXIOS(I) + STATIC(1,I)*STATIC(3,I)  RCL03170
56 CONTINUE                                  RCL03180
C
C      SEG(I)=LEFT ORDINATE OF SEGMENT I                  RCL03190
C
SEG(I)=0.DO                                 RCL03200
SEG(NSEG+1)=1.DO                            RCL03210
DO 4000 I=2,NSEG
  SEG(I)=RLENG(I-1)+SEG(I-1)                RCL03220
4000 CONTINUE                                  RCL03230
C
C      INTRERPOLATE STRUCTURAL DIMENSIONS TO NP POINTS   RCL03240
C
C      ASSUMPTION: NUMBER OF RISER SEGMENTS WITH DIFFERENT RCL03250
C      CHARACTERISTICS IS SMALLER THAN THE STATIC SOLUTION POINTS RCL03260

```

```

C      NSEG < NPI                                     RCL03370
C
C      IF (NSEG.GE.NPI) THEN                           RCL03380
C          WRITE(6,188)                                 RCL03390
188      FORMAT(' NSEG => NPI, PROGRAM STOPS')       RCL03400
          IC = 0                                       RCL03410
          RETURN                                      RCL03420
          END IF                                      RCL03430
          CALL STRUCT(EPSETA,X,NP)                      RCL03440
          CALL STRUCT(EPSETS,X,NP)                      RCL03450
          CALL STRUCT(EOM,X,NP)                         RCL03460
          CALL STRUCT(HZETA,X,NP)                        RCL03470
          CALL STRUCT(HXI,X,NP)                         RCL03480
          RETURN                                      RCL03490
          END                                           RCL03500
C
C      SUBROUTINE READ2D(NP)                           RCL03510
C      THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM RCL03520
C      DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION. RCL03530
C      IMPLICIT REAL*8(A-H,O-Z)                      RCL03540
C      PARAMETER(N=7,MNP=151,NA=4)                    RCL03550
C      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL03560
C      COMMON/STAT1/XTOP,YTOP                         RCL03570
C      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE      RCL03580
C      DIMENSION XCOOR(MNP),YCOOR(MNP)                RCL03590
C
C      STATIC(1,I) = STATIC EFFECTIVE TENSION TO     RCL03600
C      STATIC(2,I) = STATIC SHEAR FORCE IN THE XI DIRECTION RCL03610
C      STATIC(3,I) = STATIC OMEGA AROUND ETA DIRECTION RCL03620
C      STATIC(4,I) = STATIC ANGLE PHI                 RCL03630
C
C      WRITE(6,2000) MNP                                RCL03640
2000  FORMAT(' MNP=',I3)                            RCL03650
C      READ STATIC SOLUTION                          RCL03660
      READ(10,36459) NP,VM                           RCL03670
36459  FORMAT(1X,I3,1X,D12.6)                      RCL03680
      WRITE(6,2311) NP,VM                           RCL03690
2311  FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
      *' NP =',I3/
      *' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S ,VM =',D12.6) RCL03700
C
      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN            RCL03710
          ICCC=0                                     RCL03720
          WRITE(6,12439)                            RCL03730
12439  FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS') RCL03740
          RETURN                                     RCL03750
      ENDIF                                         RCL03760
C
C      READING FROM DEVICE 10                       RCL03770
C
      DO 1021 I=1,NP                               RCL03780
      READ(10,1033) X(I),STATIC(1,I),STATIC(2,I),STATIC(3,I),STATIC(4,I) RCL03790
      *,XCOOR(I),YCOOR(I),STRARC,TENSI,VLOCKXI(I)                  RCL03800
      XI(I) = X(I)                                    RCL03810
1021  CONTINUE                                     RCL03820
1033  FORMAT(10(1X,D12.6))                      RCL03830

```

```

C      EVALUATE RISER TOP X AND Y DISPLACEMENTS          RCL03930
      XTOP = XCOOR(NP)                                RCL03940
      YTOP = YCOOR(NP)                                RCL03950
C      EVALUATE MAXIMUM STATIC EFFECTIVE TENSION        RCL03960
      TOMAX=DMAX1(STATIC(1,1),STATIC(1,2))            RCL03970
      DO 9859 I=3,NP                                  RCL03980
         TOMAX = DMAX1(TOMAX,STATIC(1,I))             RCL03990
9859  CONTINUE                                     RCL04000
C                                              RCL04010
      WRITE(6,1654) TOMAX                            RCL04020
1654  FORMAT(' 2-D STATIC SOLUTION SUCCESFULLY READ'/' MAXIMUM STATIC EF RCL04030
*EFFECTIVE TENSION/WA*L = ',D10.4)                RCL04040
C                                              RCL04050
      RETURN                                         RCL04060
      END                                            RCL04070
RCL04080
      SUBROUTINE READAS(ICCC,NP,TOL)                 RCL04090
      IMPLICIT REAL*8(A-H,O-Z)                        RCL04100
      PARAMETER(N=7,MNP=151,NA=4)                      RCL04110
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE       RCL04120
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL04130
      COMMON/STAT1/XTOP,YTOP                          RCL04140
      COMMON/BOUNDA/BOUND                           RCL04150
C                                              RCL04160
C      READAS READS INITIAL APPROXIMATION RESULTING FROM A PREVIOUS      RCL04170
C      RUN OF RCLINDY1 (CORRESPONDS TO EPS = 1.D0)                    RCL04180
C                                              RCL04190
C      Y(1,I) = DYNAMIC TENSION                         RCL04200
C      Y(2,I) = SHEAR FORCE IN THE XI DIRECTION        RCL04210
C      Y(3,I) = OMEGA ABOUT ETA                         RCL04220
C      Y(4,I) = DYNAMIC ANGLE PHI                      RCL04230
C      Y(5,I) = DISPLACEMENT IN THE ZI DIRECTION, P (TANGENTIAL)    RCL04240
C      Y(6,I) = DISPLACEMENT IN THE XI DIRECTION, Q (NORMAL)        RCL04250
C      Y(7,I) = NATURAL FREQUENCY                       RCL04260
C      X(I) = UNSTRETCHED ARC LENGTH S                  RCL04270
C                                              RCL04280
      WRITE(6,2000) MNP                                RCL04290
2000  FORMAT(' MNP=',I3)                            RCL04300
C      READ SOLUTION FROM DEVICE 12                   RCL04310
      READ(12,36459) MODE,NP,SIGMAD,XTOP,YTOP        RCL04320
36459  FORMAT(1X,I2,1X,I3,3(1X,D10.4))            RCL04330
      WRITE(6,2311) NP,MODE,SIGMAD,XTOP,YTOP          RCL04340
2311  FORMAT(' INITIAL APPROXIMATION FROM DEVICE 12'/
     *' NP =',I3,' MODE NUMBER =',I2,' SIGMAD =',D10.4/' X AT TOP =',
     *D10.4,' Y AT TOP =',D10.4)                    RCL04350
      RCL04360
      RCL04370
C                                              RCL04380
      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN            RCL04390
         ICCC=0                                    RCL04400
         WRITE(6,12439)                            RCL04410
12439  FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS') RCL04420
      RETURN                                     RCL04430
      ENDIF                                       RCL04440
C                                              RCL04450
C      READING DATA FROM DEVICE 12 ...              RCL04460
C                                              RCL04470
      DO 10011 I=1,NP                            RCL04480

```

FILE: RCLINDY1 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

      READ(12,10012) X(I),(Y(J,I),J=1,7),VLOCXI(I)          RCL04490
10011 CONTINUE                                         RCL04500
10012 FORMAT(9(1X,D12.6))                           RCL04510
C   SET BOUNDARY CONDITION OMEGA ETA(0)           RCL04520
    BOUND = Y(3,1)                                RCL04530
C
      WRITE(9,1052) NP,SIGMAD                         RCL04540
1052 FORMAT(' INITIAL CONDITION FOR EPS=1. AND NP = ',I3,' POINTS, NATRCL04560
      *URAL FREQUENCY = ',D10.4,' RAD/S'/
      *'     I      ARC      TENSION      QXI      OMEGA ETA      PHI
      *      P          Q          SIGMA')            RCL04570
      DO 1601 I=1,NP                               RCL04590
      WRITE(9,1603) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
      * Y(7,I)                                     RCL04610
1601 CONTINUE                                         RCL04620
1603 FORMAT(1X,I3,8(1X,D12.6))                     RCL04630
C
      WRITE(6,9561)                                 RCL04640
9561 FORMAT(' DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION?'
      *' IF YES INPUT 1')
      READ(5,*) IPRINT                            RCL04660
      IF(IPRINT.EQ.1) THEN                         RCL04670
      WRITE(6,1052) NP                           RCL04680
      DO 9659 I=1,NP                           RCL04690
      WRITE(6,1603) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
      * Y(7,I)                                     RCL04720
9659 CONTINUE                                         RCL04730
      END IF                                         RCL04740
      WRITE(6,1722) BOUND                         RCL04750
1722 FORMAT(' ASSUMED BOUNDARY CONDITION '
      *' OMEGA ETA (0) =',D12.6)                  RCL04760
C
      TOL1 = 0.D0                                RCL04770
      DO 3933 I=1,NP                           RCL04780
      TOL1 = DMAX1(TOL1,DABS(Y(3,I)))           RCL04790
3933 CONTINUE                                         RCL04800
      WRITE(6,3934) TOL1                         RCL04810
3934 FORMAT(' MAXIMUM ABSOLUTE VALUE OF N-D OMEGA ETA IS =',D12.6/
      *' THIS NUMBER CAN BE USED TO ESTIMATE '
      *' A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS'
      *' INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE'
      *' E.G. INPUT 0.01 FOR 1% ACCURACY')
      READ(5,*) TOLV                            RCL04820
      TOL = DABS(TOLV)*TOL1                      RCL04830
C
1005  WRITE(6,1004)                                RCL04840
1004  FORMAT(' IF YOU WISH TO STOP INPUT 0'
      *' IF YOU WISH TO CONTINUE WITH ITERATIONS INPUT 1')
      READ(5,*) ICCC                            RCL04850
      IF((ICCC.NE.0).AND.(ICCC.NE.1)) GOTO 1005
C
      RETURN                                         RCL04930
      END                                           RCL04940
C
      SUBROUTINE APPROX(TLENG,ICCC,NP,TOL)
C      THIS SUBROUTINE EVALUATES THE APPROXIMATE ASYMPTOTIC ANALYTIC
      RCL04950
      RCL04960
      RCL04970
      RCL04980
      RCL04990
      RCL05000
      RCL05010
      RCL05020
      RCL05030
      RCL05040

```

```

C      SOLUTION OF THE IN-PLANE DYNAMIC EQUATIONS.          RCL05050
C      THE SOLUTION CONSISTS OF A SLOW AND A FAST PART.       RCL05060
C      IMPLICIT REAL*8(A-H,O-Z)                                RCL05070
C      PARAMETER(N=7,MNP=151,NA=4)                            RCL05080
C      COMMON/CONST/XPI,XPI2,RHOW,GRAV                         RCL05090
C      COMMON/SOLUT/ X(MNP),Y(N,MNP),ABT(N),MODE              RCL05100
C      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI   RCL05110
C      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL05120
*ZETAM,TMAXAV,DXIM                                         RCL05130
COMMON/MATRIX/P(6,6),IOS                                 RCL05140
COMMON/OMA/OMAPP(MNP),PHIOO(MNP),XL                      RCL05150
DIMENSION FVEC(2),XS(2),WAS(19),FVEC1(1),XSS(1),WAS1(8)    RCL05160
DIMENSION A(5,5),B(5),C(5),AA(5,5),WKS1(5),WKS2(5)        RCL05170
EXTERNAL FCNS,FCNSS                                     RCL05180
C
C      APPROX DEFINES INITIAL APPROXIMATION (EPS=0.)         RCL05190
C
C      Y(1,I) = DYNAMIC TENSION                           RCL05210
C      Y(2,I) = SHEAR FORCE IN THE XI DIRECTION           RCL05220
C      Y(3,I) = OMEGA ABOUT ETA                          RCL05230
C      Y(4,I) = DYNAMIC ANGLE PHI                         RCL05240
C      Y(5,I) = DISPLACEMENT P IN THE ZETA DIRECTION (TANGENTIAL) RCL05250
C      Y(6,I) = DISPLACEMENT Q IN THE XI DIRECTION (NORMAL) RCL05260
C      Y(7,I) = NATURAL FREQUENCY                         RCL05270
C
C      DETERMINATION OF THE CABLE APPROXIMATION OF 2-D STATIC SOLUTION RCL05280
C      THE POWELL HYBRID METHOD IS USED (SEE PROGRAM RCSTAT2D FOR RCL05290
C      DETAILS FOR THE SLOW SOLUTION)
      WRITE(6,1781)                                         RCL05300
1781 FORMAT(' INPUT NORMAL DRAG COEFFICIENT USED IN STATIC SOLUTION') RCL05310
      READ(5,*)CD                                         RCL05320
      XK = 0.5D0*RHOW*DXIM*CD*VM*DABS(VM)*TLENG          RCL05330
      WRITE(6,8881) VM,CD                               RCL05340
8881 FORMAT(' MEAN CURRENT SPEED =',D12.6,' NORMAL DRAG COEFFICIENT =',RCL05350
      *D12.6)                                         RCL05360
C
      2601 WRITE(6,1501)                                     RCL05370
1501 FORMAT(' INPUT ESTIMATES OF LAMBDA AND C'/
      *' LAMBDA MUST HAVE THE SIGN OF MEAN CURRENT SPEED'/
      *' FOR SMALL XTOP A GOOD ESTIMATE OF C IS -0.5*LAMBDA') RCL05380
      READ(5,*) XL,XC                                     RCL05390
C
      XS(1) = XL                                         RCL05400
      XS(2) = XC                                         RCL05410
      XTOL = 1.D-8                                       RCL05420
      NS = 2                                           RCL05430
      LWA = 19                                         RCL05440
      READ(5,*) XL,XC                                     RCL05450
C
      XS(1) = XL                                         RCL05460
      XS(2) = XC                                         RCL05470
      XTOL = 1.D-8                                       RCL05480
      NS = 2                                           RCL05490
      LWA = 19                                         RCL05500
      READ(5,*) XL,XC                                     RCL05510
C
      CALL NAG SUBROUTINE C05NBF TO DETERMINE THE CABLE APPROXIMATION RCL05520
      IFAIL = 0                                         RCL05530
      CALL C05NBF(FCNS,NS,XS,FVEC,XTOL,WAS,LWA,IFAIL)    RCL05540
C
      WRITE(6,1356) IFAIL,XS(1),XS(2)                  RCL05550
1356 FORMAT(' SOLUTION FOR CABLE... /' IFAIL =',I2/
      *' LAMBDA =',D12.6/
      *' C      =',D12.6)                                RCL05560

```

```

C
      WRITE(6,2500) FVEC(1),FVEC(2)
2500 FORMAT(' VALUE OF FUNCTION FOR HORIZONTAL DISPLACEMENT =',D12.6/
     *' VALUE OF FUNCTION FOR VERTICAL DISPLACEMENT =',D12.6/
     *' DO YOU WISH TO TRY A DIFFERENT INITIAL LAMDA, C, OR TOLERANCE'/
     *' IF YES INPUT 1')
      READ(5,*) IPP
      IF (IPP.EQ.1) GO TO 2601
      IF (IFAIL.NE.0) GO TO 2601

C
      XL = XS(1)
      XC = XS(2)
      TE = XK/XL
      WRITE(6,1654) XL,XC,TE
1654 FORMAT(' CABLE APPROXIMATION'/
     *' LAMBDA =',D12.6,' C =',D12.6,' TENSION =',D12.6)

C
C      DETERMINE SERIES PHIO0 AND OMEGA ETA USING THE CABLE APPROXIMATIONRCL05780
C
      DO 1239 I=1,NPI
      S = XI(I)
      PHIO0(I) = PHIO(S,XL,XC)
      OMAPP(I)=XL*DSIN(PHIO0(I))**2
1239 CONTINUE

C
C      CALCULATE FUNCTIONS F THAT DETERMINE THE SLOW DYNAMIC SOLUTION
C
      CALL FUNC(PHIO0,HZETAM,XL,NPI)

C
C      DETERMINE THE INITIAL APPROXIMATION TO THE NATURAL FREQUENCY
C      DETERMINANT OF THE 6X6 MATRIX P SHOULD BE ZERO.
      IOS = 0
      NSS = 1
      LWAI = 8
23  WRITE(6,*) ' INPUT INITIAL GUESS FOR SIGMA IN RAD/S'
      READ(5,*) XSS(1)

C      NONDIMENSIONALIZE SIGMA
      XSS(1) = XSS(1)*TLENG*DSQRT(TMAXAV/TOMAX)
      XTOL = 1.D-8
      IFAIL = 0

C
C      CALL CO5NBF TO EVALUATE SIGMA USING A MODIFICATION OF THE POWELL
C      HYBRID METHOD
      CALL CO5NBF(FCNSS,NSS,XSS,FVEC1,XTOL,WAS1,LWA1,IFAIL)
      WRITE(6,*) ' IFAIL FOR APPROXIMATE EVALUATION OF SIGMA IS ',IFAIL
      SIGMAD = XSS(1)*DSQRT(TOMAX/TMAXAV)/TLENG

C
      WRITE(6,24) FVEC1(1),SIGMAD
24  FORMAT(1X,'THE DETERMINANT IS = ',D10.4/' THE CONVERGED SIGMA IS 'RCL06090
     *,D10.4/' INPUT 1 IF YOU WANT TO REDO THE CALCULATION'/' INPUT 2 IF RCL06100
     * YOU WANT TO STOP')
      READ(5,*) ISIGMA
      IF (ISIGMA.EQ.1) THEN
          GO TO 23
      ELSE IF (ISIGMA.EQ.2) THEN
          ICCC = 0

```

```

        RETURN                               RCL06170
        END IF                               RCL06180
        SIGMA = XSS(1)                      RCL06190
        WRITE(6,*) ' INPUT MODE NUMBER CORRESPONDING TO ABOVE APPROXIMATE RCL06200
*SIGMA'                                RCL06210
        READ(5,*) MODE                        RCL06220
C                                         RCL06230
C                                         ELIMINATE ONE ROW AND ONE COLUMN FROM P MATRIX, AND DETERMINE RCL06240
C                                         THE LINEAR SYSTEM OF EQUATIONS TO SOLVE FOR THE CONSTANT RCL06250
C                                         COEFFICIENTS OF THE APPROXIMATE SOLUTION. (SET A1=1.) RCL06260
C                                         RCL06270
        CALL XMATR(P,SIGMA,IOS)             RCL06280
        DO 750 K=1,5                         RCL06290
          DO 751 I=2,6
            A(K,I-1) = P(K,I)              RCL06300
 751  CONTINUE                           RCL06310
            B(K) = -P(K,1)                  RCL06320
 750  CONTINUE                           RCL06330
                                         RCL06340
C                                         RCL06350
C                                         USE NAG SUBROUTINE F04ATF TO SOLVE THE LINEAR SYSTEM TO DETERMINE RCL06360
C                                         THE CONSTANT COEFFICIENTS USING CROUT'S FACTORISATION METHOD. RCL06370
C                                         RCL06380
        IFA = 0                             RCL06390
        IA = 5                             RCL06400
        CALL F04ATF(A,IA,B,IA,C,AA,IA,WKS1,WKS2,IFA) RCL06410
        WRITE(6,*) ' IFAIL FOR THE EVALUATION OF CONSTANT COEFFICIENTS IS' RCL06420
        *,IFA                               RCL06430
        WRITE(6,863) (C(I),I=1,IA)          RCL06440
 863  FORMAT(' CONSTANT COEFFICIENTS ARE:'// A1 = 1.00      B1 =', RCL06450
        *D10.4/' G1 =',D10.4,' D1 =',D10.4/' C1 =',D10.4,' C2 =',D10.4) RCL06460
C                                         CHECK IF SIXTH EQUATION IS SATISFIED RCL06470
        TEST = P(6,1)                      RCL06480
        DO 766 I=1,5
          TEST = TEST + P(6,I+1)*C(I)    RCL06490
 766  CONTINUE                           RCL06500
        WRITE(6,775) TEST                  RCL06510
 775  FORMAT(' SIXTH EQUATION = ',D10.4) RCL06520
C                                         RCL06530
C                                         CALL SUBROUTINES TO EVALUATE INITIAL APPROXIMATION RCL06540
        CALL INITAP(C,SIGMA,NP)           RCL06550
C                                         OUTPUT INITIAL APPROXIMATION TO DEVICE 9 RCL06570
        WRITE(9,1002) NP,SIGMAD          RCL06580
 1002 FORMAT(' INITIAL CONDITION FOR EPS=0. AND NP = ',I3,' POINTS, NATRCL06590
        *URAL FREQUENCY = ',D10.4,' RAD/SEC'/
        *' I      ARC      TENSION      SHEAR XI      OMEGA ETA      PHI1RCL06610
        *' P      Q      SIGMA')
        DO 1001 I=1,NP
          WRITE(9,1003) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
        * Y(7,I)
 1001 CONTINUE                           RCL06660
 1003 FORMAT(1X,I3,8(1X,D12.6))       RCL06670
C                                         RCL06680
        WRITE(6,7561)                   RCL06690
 7561 FORMAT(' DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION'/
        *' IF YES INPUT'1')
        READ(5,*) IPRINT               RCL06710
                                         RCL06720

```

```

IF(IPRINT.EQ.1) THEN                                RCL06730
  WRITE(6,1002) NP,SIGMAD                           RCL06740
  DO 7659 I=1,NP                                    RCL06750
    WRITE(6,1003) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I), Y(7,I) , RCL06760
  * Y(7,I)
7659 CONTINUE                                       RCL06770
  END IF                                             RCL06780
C
  TOL1 = 0.D0                                         RCL06790
  DO 3931 I=1,NP                                    RCL06800
    TOL1 = DMAX1(TOL1,DABS(Y(3,I)))                RCL06810
3931 CONTINUE                                       RCL06820
  WRITE(6,3932) TOL1                               RCL06830
3932 FORMAT(' MAXIMUM ABSOLUTE VALUE OF N-D OMEGA ETA IS =',D12.6/
*' THIS NUMBER CAN BE USED TO ESTIMATE '/          RCL06840
*' A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS'/
*' INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE'/
*' E.G. INPUT 0.01 FOR 1% ACCURACY')               RCL06850
  READ(5,*) TOLV                                     RCL06860
  TOL = DABS(TOLV)*TOL1                            RCL06870
  WRITE(6,*) ' IF YOU WANT TO STOP INPUT 0'         RCL06880
  READ(5,*) ICCC                                     RCL06890
  IF (ICCC.NE.0) ICCC = 1                           RCL06900
C
  RETURN                                              RCL06910
END                                                 RCL06920
C
SUBROUTINE PCNS(NS,XS,FVEC,IFLAG)                  RCL06930
C
THIS SUBROUTINE IS USED BY NAG ROUTINE COSNBF TO DETERMINE THE      RCL06940
CABLE APPROXIMATION.                                            RCL06950
IMPLICIT REAL*8(A-H,O-Z)                                RCL06960
COMMON/STAT1/XTOP,YTOP                                 RCL06970
DIMENSION XS(NS),FVEC(NS)                            RCL06980
C
XL = XS(1)                                           RCL06990
XC = XS(2)                                           RCL07000
XLINV = 1.D0/XL                                      RCL07010
PHIO0 = PHIO(0.D0,XL,XC)                            RCL07020
PHIO1 = PHIO(1.D0,XL,XC)                            RCL07030
C
F1 = XLINV*(1.D0/DSIN(PHIO0)-1.D0/DSIN(PHIO1))     RCL07040
F1 = F1-XTOP                                         RCL07050
FVEC(1) = F1                                         RCL07060
C
TT1 = DTAN(PHIO1/2.D0)                             RCL07070
TTO = DTAN(PHIO0/2.D0)                             RCL07080
F2 = XLINV*DLOG(TT1/TTO)                           RCL07090
F2 = F2-YTOP                                         RCL07100
FVEC(2) = F2                                         RCL07110
RETURN                                              RCL07120
END                                                 RCL07130
C
FUNCTION PHIO(S,XL,XC)                              RCL07140
C
THIS FUNCTION EVALUATES PHIO0 GIVEN S, LAMBDA AND C      RCL07150
IMPLICIT REAL*8(A-H,O-Z)                            RCL07160
COMMON/CONST/XPI,XPI2,RHOW,GRAV                     RCL07170

```

```

C
XVAR = -(XL*S+XC)                                RCL07290
PHIO = DATAN(1.D0/XVAR)                            RCL07300
IF (XVAR.LT.0.D0) THEN                           RCL07310
  PHIO = PHIO + XPI                             RCL07320
END IF                                              RCL07330
RETURN                                              RCL07340
END                                                 RCL07350
RCL07360
RCL07370
RCL07380
RCL07390
RCL07400
RCL07410
RCL07420
RCL07430
RCL07440
RCL07450
RCL07460
RCL07470
RCL07480
RCL07490
RCL07500
RCL07510
RCL07520
RCL07530
RCL07540
RCL07550
RCL07560
RCL07570
RCL07580
RCL07590
RCL07600
RCL07610
RCL07620
RCL07630
RCL07640
RCL07650
RCL07660
RCL07670
RCL07680
RCL07690
RCL07700
RCL07710
RCL07720
RCL07730
RCL07740
RCL07750
RCL07760
RCL07770
RCL07780
RCL07790
RCL07800
RCL07810
RCL07820
RCL07830
RCL07840

C
SUBROUTINE FCNSS(NSS,XSS,FVEC1,IFLAG)           SUBROUTINE FCNSS(NSS,XSS,FVEC1,IFLAG)
THIS SUBROUTINE IS USED BY NAG ROUTINE COSNBF TO EVALUATE THE
DETERMINANT OF MATRIX P AND SOLVE FOR THE APPROXIMATE NATURAL
FREQUENCY.                                         RCL07390
IMPLICIT REAL*8(A-H,O-Z)                         RCL07400
PARAMETER(N=6)                                    RCL07410
COMMON/MATRIX/ P(6,6),IOS                        RCL07420
DIMENSION XSS(NSS),FVEC1(NSS),A(N,N),WKSPCE(N)  RCL07430
CREATE THE MATRIX P                               RCL07440
SIGMA = XSS(1)                                    RCL07450
CALL XMATR(P,SIGMA,IOS)                          RCL07460
IOS = 1                                           RCL07470
FIND ITS DETERMINANT USING NAG SUBROUTINE F03AAF (CROUT
FACTORISATION METHOD).                         RCL07480
DO 652 I=1,6                                     RCL07490
  DO 652 K=1,6
    A(I,K) = P(I,K)                            RCL07500
652 CONTINUE                                     RCL07510
  IFAI = 1                                         RCL07520
  CALL F03AAF(A,N,N,DET,WKSPCE,IFAI)            RCL07530
  WRITE(6,842) IFAI,DET                         RCL07540
842 FORMAT(' IFAIL FOR THE DETERMINANT OF P =',I2/' DET =',D12.6) RCL07550
C
  FVEC1(1) = DET                                RCL07560
  RETURN                                         RCL07570
  END                                             RCL07580
RCL07590

C
SUBROUTINE XMATR(P,SIGMA,IOS)                   SUBROUTINE XMATR(P,SIGMA,IOS)
THIS SUBROUTINE EVALUATES THE MATRIX P(6,6) ELEMENTS
AT FREQUENCY SIGMA.                            RCL07600
IMPLICIT REAL*8(A-H,O-Z)                         RCL07610
PARAMETER(MNP=151,NA=4)                          RCL07620
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL07630
COMMON/CGEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL07710
*ZETAM,TMAXAV,DXIM                                RCL07720
COMMON/SLOW/F1(MNP),F2(MNP),F3(MNP),F4(MNP),F5(MNP),F6(MNP) RCL07730
COMMON/OMA/OMAPP(MNP),PHIO0(MNP),XL               RCL07740
DIMENSION P(6,6)                                  RCL07750
C
EVALUATE ALL CONSTANT ELEMENTS (NOT A FUNCTION OF SIGMA) RCL07760
IF (IOS.EQ.0) THEN
  SINP0 = DSIN(PHIO0(1))                         RCL07770
  SINP1 = DSIN(PHIO0(NPI))                       RCL07780
  SIN2P0 = DSIN(2.D0*PHIO0(1))                  RCL07790
  SIN2P1 = DSIN(2.D0*PHIO0(NPI))                RCL07800
  COSP0 = DCOS(PHIO0(1))                         RCL07810
  COSP1 = DCOS(PHIO0(NPI))                      RCL07820
  COS2P0 = DCOS(2.D0*PHIO0(1))                 RCL07830
RCL07840

```

```

COS2P1 = DCOS(2.D0*PHI00(NPI)) RCL07850
DO 881 I=1,6 RCL07860
  DO 881 K=1,6 RCL07870
    P(I,K) = 0.D0 RCL07880
881  CONTINUE RCL07890
  P(1,2) = SINPO**2 RCL07900
  P(1,3) = 1.D0 RCL07910
  P(1,5) = F1(1) RCL07920
  P(1,6) = F2(1) RCL07930
  P(2,2) = SIN2PO RCL07940
  P(2,3) = -1.D0/(DSQRT(EPSETA(1))*OMAPP(1)) RCL07950
  P(2,5) = F3(1) RCL07960
  P(2,6) = F4(1) RCL07970
  P(3,3) = 1.D0/(EPSETA(1)*OMAPP(1)) RCL07980
  P(3,5) = F5(1) RCL07990
  P(3,6) = F6(1) RCL08000
  P(4,4) = 1.D0 RCL08010
  P(4,5) = F1(NPI) RCL08020
  P(4,6) = F2(NPI) RCL08030
  P(5,4) = 1.D0/(DSQRT(EPSETA(NPI))*OMAPP(NPI)) RCL08040
  P(5,5) = F3(NPI) RCL08050
  P(5,6) = F4(NPI) RCL08060
  P(6,4) = 1.D0/(EPSETA(NPI)*OMAPP(NPI)) RCL08070
  P(6,5) = F5(NPI) RCL08080
  P(6,6) = F6(NPI) RCL08090
END IF RCL08100
C EVALUATE ELEMENTS THAT ARE A FUNCTION OF SIGMA
COSS = DCOS(SIGMA) RCL08110
SINS = DSIN(SIGMA) RCL08120
P(2,1) = SIGMA/XL RCL08130
C RCL08140
  P(3,1) = SIGMA*SIN2PO RCL08150
  P(3,2) = - SIGMA**2/XL + 2.D0*OMAPP(1)*COS2PO RCL08160
C RCL08170
  P(4,1) = SINS*SINP1**2 RCL08180
  P(4,2) = COSS*SINP1**2 RCL08190
C RCL08200
  P(5,1) = SIGMA*COSS/XL + SIN2P1*SINS RCL08210
  P(5,2) = -SIGMA*SINS/XL + SIN2P1*COSS RCL08220
C RCL08230
  P(6,1) = SINS*(2.D0*OMAPP(NPI)*COS2P1 - SIGMA**2/XL) RCL08240
  P(6,1) = P(6,1) + SIGMA*COSS*SIN2P1 RCL08250
  P(6,2) = COSS*(-SIGMA**2/XL + 2.D0*OMAPP(NPI)*COS2P1) RCL08260
  P(6,2) = P(6,2) - SIGMA*SIN2P1*SINS RCL08270
RETURN RCL08280
END RCL08290
C RCL08300
SUBROUTINE INITAP(C,SIGMA,NP) RCL08310
IMPLICIT REAL*8(A-H,O-Z) RCL08320
PARAMETER(N=7,MNP=151,NA=4) RCL08330
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL08340
COMMON/STAT2/CONST1(MNP),CONST2(MNP) RCL08350
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL08360
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL08380
*ZETAM,TMAXAV,DXIM RCL08390
COMMON/SLOW/F1(MNP),F2(MNP),F3(MNP),F4(MNP),F5(MNP),F6(MNP) RCL08400

```

FILE: RCLINDY1 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

COMMON/BOUNDA/BOUND RCL08410
COMMON/OMA/OMAPP(MNP),PHIOO(MNP),XL RCL08420
DIMENSION P(MNP),Q(MNP),QS(MNP),A1(MNP),C(5),PHII(MNP) RCL08430
DIMENSION OMETA1(MNP),OMET1S(MNP),Y2(MNP),Y2S(MNP) RCL08440
C RCL08450
C THIS SUBROUTINE CALCULATES THE INITIAL APPROXIMATION OF THE RCL08460
C SOLUTION ONCE SIGMA AND THE CONSTANT COEFFICIENTS ARE DETERMINED. RCL08470
C RCL08480
DO 444 K=1,NP RCL08490
    Y(7,K) = SIGMA
444 CONTINUE RCL08500
C RCL08510
C IBM RANGE OF A REAL*8 NUMBER SO THAT DEXP(X) IS REAL*8 RCL08520
C CONDITION XMINR<X. USE A STRICTER LIMIT. RCL08530
C RCL08540
XMINR=-65.D0*DLOG(16.D0) RCL08550
XMINR=XMINR/3.D0 RCL08560
C RCL08570
C CALCULATE THE IN PLANE DISPLACEMENTS P AND Q AS WELL AS QS RCL08580
C (ANALYTIC EXPRESSIONS) RCL08590
C RCL08600
C RCL08610
XNU1 = DSQRT(EPSETA(1)) RCL08620
XNU2 = DSQRT(EPSETA(NP)) RCL08630
DO 815 I=1,NP RCL08640
    SIG = SIGMA*X(I)
    COSS = DCOS(SIG)
    SINS = DSIN(SIG)
    SINP = DSIN(PHIOO(I))
    COSP = DCOS(PHIOO(I))
    SIN2P = DSIN(PHIOO(I)*2.D0)
    COS2P = DCOS(PHIOO(I)*2.D0)
    SINP2 = SINP**2
    POWER1 = -X(I)/XNU1
    POWER2 = -(1-X(I))/XNU2
    IF (POWER1.GT.XMINR) THEN
        EXPO1 = DEXP(POWER1)
    ELSE
        EXPO1 = 0.D0
    END IF
    IF (POWER2.GT.XMINR) THEN
        EXPO2 = DEXP(POWER2)
    ELSE
        EXPO2 = 0.D0
    END IF
C CALCULATE P
C
P(I) = SINS*SINP2 + C(1)*COSS*SINP2 + C(2)*EXPO1 + C(3)*EXPO2 RCL08870
P(I) = P(I) + C(4)*F1(I) + C(5)*F2(I) RCL08880
C CALCULATE Q
C
Q(I) = SIGMA*COSS/XL + SIN2P*SINS RCL08910
Q(I) = Q(I) + C(1)*(-SIGMA*SINS/XL + SIN2P*COSS) RCL08920
Q(I) = Q(I) - C(2)*EXPO1/(XNU1*OMAPP(I)) + C(4)*F3(I) RCL08930
Q(I) = Q(I) + C(3)*EXPO2/(XNU2*OMAPP(I)) + C(5)*F4(I) RCL08940
C CALCULATE QS*

```

```

D1 = SINS*(2.D0*OMAPP(I)*COS2P - SIGMA**2/XL) RCL08970
D1 = D1 + SIGMA*SIN2P*COSS RCL08980
C RCL08990
D2 = COSS*(- SIGMA**2/XL + 2.D0*OMAPP(I)*COS2P) RCL09000
D2 = D2 - SIN2P*SIGMA*SINS RCL09010
C RCL09020
QS(I) = D1 + C(1)*D2 + C(2)*EXP01/(XNU1**2*OMAPP(I)) + C(4)*F5(I) RCL09030
QS(I) = QS(I) + C(3)*EXP02/(XNU2**2*OMAPP(I)) + C(5)*F6(I) RCL09040
C RCL09050
A1(I) = P(I)**2*HZETAM + Q(I)**2 RCL09060
815 CONTINUE RCL09070
C CALCULATE THE ORTHONORMALIZING CONSTANT FOR P AND Q RCL09080
A = 0.D0 RCL09090
DO 842 I=2,NP RCL09100
DX2 = (X(I)-X(I-1))*0.5D0 RCL09110
A = A + (A1(I)+A1(I-1))*DX2 RCL09120
842 CONTINUE RCL09130
A = DSQRT(A) RCL09140
C ORTHONORMALIZE P, Q AND QS AND DETERMINE PHI1 RCL09150
DO 843 I=1;NP RCL09160
P(I) = P(I)/A RCL09170
Q(I) = Q(I)/A RCL09180
QS(I) = QS(I)/A RCL09190
PHI1(I) = (QS(I)+STATIC(3,I)*P(I))/(1.D0+EOM(I)) RCL09200
843 CONTINUE RCL09210
C RCL09220
C EVALUATE THE DERIVATIVES OF PHI1 AND OMETA1 NUMERICALLY. RCL09230
CALL DER1(PHI1,X,OMETA1,NP) RCL09240
CALL DER1(OMETA1,X,OMET1S,NP) RCL09250
C DETERMINE THE INITIAL APPROXIMATION RCL09260
DO 853 I=1,NP RCL09270
Y(2,I) = -EPSETS(I)*OMETA1(I) - EPSETA(I)*OMET1S(I) RCL09280
Y(3,I) = OMETA1(I) RCL09290
Y(4,I) = PHI1(I) RCL09300
Y(5,I) = P(I) RCL09310
Y2(I) = Y(2,I) RCL09320
Y(6,I) = Q(I) RCL09330
853 CONTINUE RCL09340
C EVALUATE DYNAMIC TENSION FROM THE FIRST EQUATION BY INTEGRATION RCL09350
C USING THE TRAPEZOIDAL RULE. THE INTEGRATION CONSTANT IS DETERMINED RCL09360
C USING THE SECOND EQUATION AND EVALUATING TENSION IN THE MIDDLE OF RCL09370
C THE RISER. RCL09380
CALL DER1(Y2,X,Y2S,NP) RCL09390
C EVALUATE CONSTANT OF INTEGRATION AT NI POINT RCL09400
NI = NINT(NP/2.D0) RCL09410
Y(1,NI) = -(Y2S(NI) + Y(3,NI) + Q(NI)*Y(7,1)**2)/STATIC(3,NI) RCL09420
Y(1,NI) = Y(1,NI) - CONST1(NI)*PHI1(NI)/STATIC(3,NI) RCL09430
C INTEGRATE FIRST EQUATION, FORWARDS AND BACKWARDS RCL09440
DO 871 I=NI+1,NP RCL09450
DX2 = (X(I)-X(I-1))*0.5D0 RCL09460
TERM = (STATIC(2,I)*OMETA1(I) + STATIC(2,I-1)*OMETA1(I-1))*DX2 RCL09470
TERM = TERM + (STATIC(3,I)*Y2(I) + STATIC(3,I-1)*Y2(I-1))*DX2 RCL09480
TERM = TERM + (CONST2(I)*PHI1(I) + CONST2(I-1)*PHI1(I-1))*DX2 RCL09490
Y(1,I) = Y(1,I-1) - HZETAM*Y(7,1)**2*(P(I)+P(I-1))*DX2 + TERM RCL09500
871 CONTINUE RCL09510
DO 872 I=NI,2,-1 RCL09520

```

```

DX2 = (-X(I)+X(I-1))*0.5D0 RCL09530
TERM = (STATIC(2,I)*OMETA1(I) + STATIC(2,I-1)*OMETA1(I-1))*DX2 RCL09540
TERM = TERM + (STATIC(3,I)*Y2(I) + STATIC(3,I-1)*Y2(I-1))*DX2 RCL09550
TERM = TERM + (CONST2(I)*PHI1(I) + CONST2(I-1)*PHI1(I-1))*DX2 RCL09560
Y(1,I-1) = Y(1,I) - HZETAM*Y(7,1)**2*(P(I)+P(I-1))*DX2 + TERM RCL09570
872 CONTINUE RCL09580
C DETERMINE THE FOURTH BOUNDARY CONDITION AT S=0 RCL09590
BOUND = Y(3,1) RCL09600
WRITE(6,84) A, BOUND RCL09610
84 FORMAT(' THE ORTHONORMALIZING CONSTANT FOR THE APPROXIMATE'/
*' SOLUTION IS SQRT(A) = ',D10.4/' THE BOUNDARY CONDITION AT S=0 ISRCL09630
* OMEGAETA(0) = ',D10.4) RCL09640
RETURN RCL09650
END RCL09660
RCL09670
C SUBROUTINE FUNC(PHI00,HZETAM,XL,NP) RCL09680
C THIS SUBROUTINE EVALUATES THE FUNCTIONS THAT RESULT FROM THE RCL09690
C APPROXIMATE SLOW SOLUTION FOR IN-PLANE DYNAMICS OF COMPLIANT RCL09700
C RISERS. RCL09710
IMPLICIT REAL*8(A-H,O-Z) RCL09720
PARAMETER(MNP=151) RCL09730
COMMON/SLOW/F1(MNP),F2(MNP),F3(MNP),F4(MNP),F5(MNP),F6(MNP) RCL09740
DIMENSION PHI00(MNP) RCL09750
XPI2 = 2.D0*DATAN(1.D0) RCL09760
H = HZETAM RCL09770
H2 = H**2 RCL09780
DO 537 I=1,NP RCL09790
  COSPHI = DCOS(PHI00(I)) RCL09800
  SINPHI = DSIN(PHI00(I)) RCL09810
  COS2PH = DCOS(2.D0*PHI00(I)) RCL09820
  COSPH2 = COSPHI**2 RCL09830
  SINPH2 = SINPHI**2 RCL09840
  RCL09850
C   F1(I) = COSPHI*(1.D0+(H-1.D0)*COSPH2/6.D0*(3.D0+H)/20.D0*COSPH2)RCL09860
  F1(I) = F1(I) + COSPHI*(H-1.D0)*COSPH2/6.D0 RCL09870
C   F2(I) = 1.D0 - H2 - H*(H-1.D0)/2.D0*COSPH2 RCL09880
  F2(I) = F2(I) + H2*(SINPHI + COSPHI*(XPI2 - PHI00(I))) RCL09890
C   F3(I) = -SINPHI*(H-1.D0)/2.D0*COSPH2*(1.D0+(3.D0+H)*COSPH2/12.D0)RCL09920
  F3(I) = F3(I) - SINPHI RCL09930
C   F4(I) = SINPHI*(H*(H-1.D0)*COSPHI - H2*(XPI2 - PHI00(I))) RCL09940
C   F5(I) = 2.D0 - H + (H-1.D0)*(6.D0-H)*COSPH2/6.D0 RCL09970
  F5(I) = F5(I) + 5.D0/24.D0*(H-1.D0)*(H+3.D0)*COSPH2**2 RCL09980
  F5(I) = -F5(I)*XL*SINPH2*COSPHI RCL09990
C   F6(I) = H*(H-1)*COS2PH + H2*SINPHI - H2*(XPI2 - PHI00(I))*COSPHI RCL10010
  F6(I) = XL*SINPH2*F6(I) RCL10020
537 CONTINUE RCL10030
88 FORMAT(I3,7(2X,D12.6)) RCL10040
RETURN RCL10050
END RCL10060
C SUBROUTINE STRUCT(ARRAY,X,NP) RCL10070
RCL10080

```

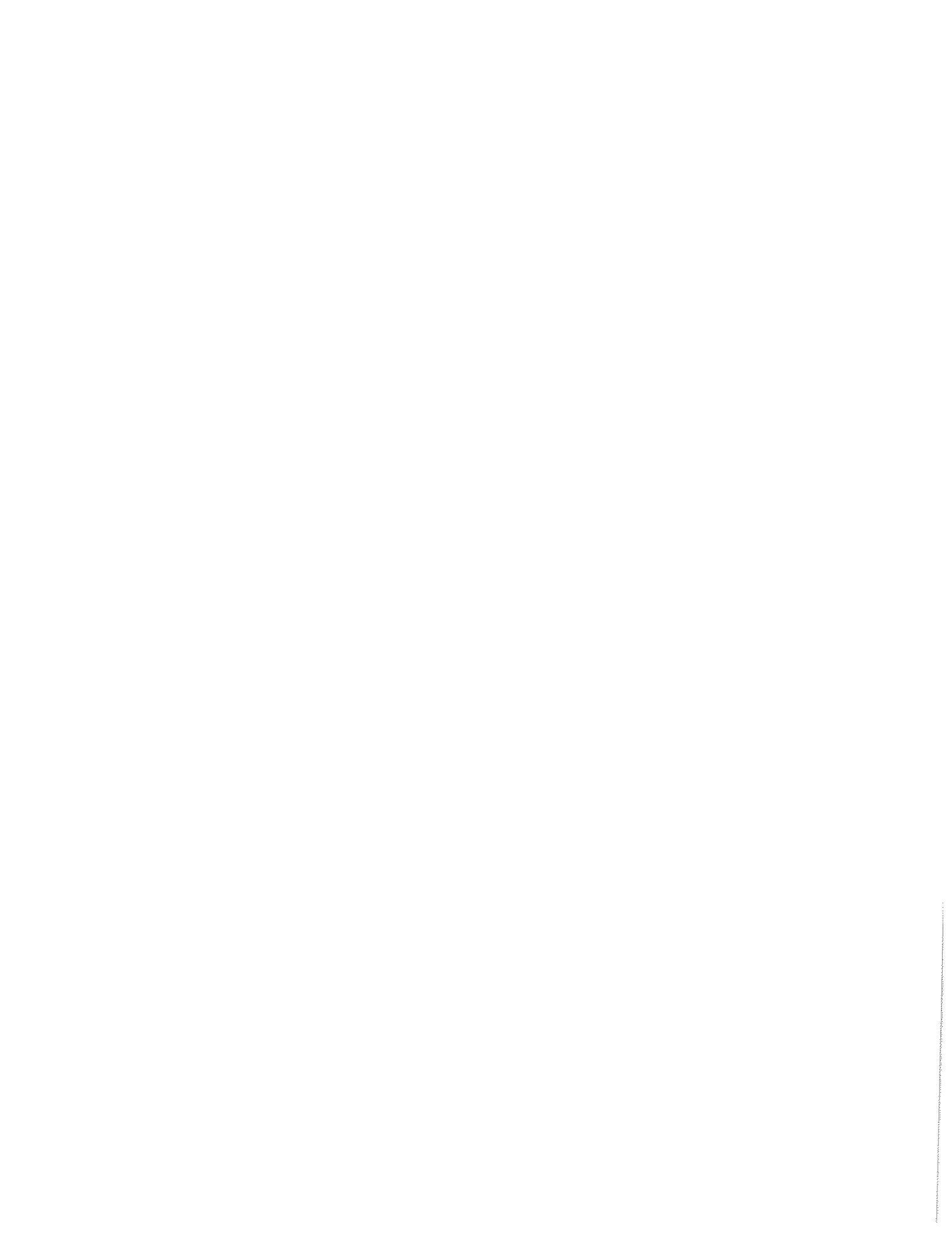
FILE: RCLINDY1 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

C   THIS SUBROUTINE CHANGES A SERIES OF DATA GIVEN IN          RCL10090
C   SEGMENTS TO A SERIES OF DATA CORRESPONDING TO SPECIFIC POINTS RCL10100
C   ASSUMPTION: NSEG < NP                                     RCL10110
C   IMPLICIT REAL*8(A-H,O-Z)                                 RCL10120
C   PARAMETER(MNP=151)                                       RCL10130
C   COMMON/INPUT1/TLEN,WA,WT,NSEG                           RCL10140
C   COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)                     RCL10150
C   DIMENSION ARRAY(MNP),HELP(MNP),X(MNP)                   RCL10160
C
C   IF(NSEG.EQ.1) THEN                                         RCL10170
C     DO 83 I=1,NP                                           RCL10180
C       HELP(I) = ARRAY(1)                                    RCL10190
83    CONTINUE                                              RCL10200
C   ELSE                                                       RCL10210
C     HELP(1) = ARRAY(1)                                    RCL10220
C     HELP(NP) = ARRAY(NSEG)                                RCL10230
C     I=2                                                    RCL10240
C     DO 84 K=2,NP-1                                       RCL10250
C       IF ((X(K).GT.SEG(I-1)).AND.(X(K).LT.SEG(I))) THEN RCL10260
C         HELP(K) = ARRAY(I-1)                                RCL10270
C       ELSE IF (X(K).EQ.SEG(I)) THEN                         RCL10280
C         HELP(K) = 0.5D0*(ARRAY(I-1) + ARRAY(I))            RCL10290
C       ELSE IF (X(K).GT.SEG(I)) THEN                         RCL10300
C         HELP(K) = ARRAY(I)                                  RCL10310
C         I = I + 1                                         RCL10320
C       END IF                                                 RCL10330
84    CONTINUE                                              RCL10340
C     END IF                                                 RCL10350
C     DO 85 K=1,NP                                           RCL10360
C       ARRAY(K) = HELP(K)                                    RCL10370
85    CONTINUE                                              RCL10380
C   RETURN                                                 RCL10390
C   END                                                   RCL10400
C
C   SUBROUTINE DER1(ARRAY,X,DERIV,NP)                         RCL10410
C
C   THIS SUBROUTINE EVALUATES THE FIRST DERIVATIVE OF A SERIES OF DATA RCL10440
C   POINTS USING A SECOND ORDER APPROXIMATION (NONUNIFORM GRID FINITE RCL10450
C   DIFFERENCES).                                            RCL10460
C   IMPLICIT REAL*8(A-H,O-Z)                                 RCL10470
C   PARAMETER (MNP=151)                                     RCL10480
C   DIMENSION ARRAY(MNP),DERIV(MNP),X(MNP)                  RCL10490
C
C   USE FIRST ORDER FOR END POINTS DERIVATIVES             RCL10500
C   DERIV(1) = (-ARRAY(1) + ARRAY(2))/(X(2)-X(1))           RCL10510
C   DERIV(NP) = (-ARRAY(NP-1) + ARRAY(NP))/(X(NP)-X(NP-1)) RCL10520
C
C   DO 836 I=2,NP-1                                         RCL10530
C     DX = X(I) - X(I-1)                                    RCL10540
C     DX1 = X(I+1) - X(I)                                   RCL10550
C     D = DX/DX1                                           RCL10560
C     DERIV(I) = (D*(ARRAY(I+1)-ARRAY(I)) - (ARRAY(I-1)-ARRAY(I))/D) RCL10570
C     DERIV(I) = DERIV(I)/(DX+DX1)                          RCL10580
836  CONTINUE                                              RCL10590
C   RETURN                                                 RCL10600
C   END                                                   RCL10610
C
C   SUBROUTINE OUTPUT(NP)                                    RCL10620

```



```

C      THIS SUBROUTINE OUTPUTS THE RESULTS IN TWO FORMATS; A COMPLETE      RCL10650
C      FORMAL FORM AND A FORM TO BE USED FOR PLOTTING.                      RCL10660
      IMPLICIT REAL*8(A-H,O-Z)                                              RCL10670
      PARAMETER(MNP=151,NA=4,N=7)                                             RCL10680
      CHARACTER*80 NAME                                                       RCL10690
      COMMON/CONST/XPI,XPI2,RHOW,GRAV                                         RCL10700
      COMMON/INPUT0/NAME                                                       RCL10710
      COMMON/INPUT1/TLEN,WA,WT,NSEG                                            RCL10720
      COMMON/INPUT2/RHO0,AI,CFLUID,PRESS,AO(MNP)                             RCL10730
      COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)                                       RCL10740
      COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP) RCL10750
      * ,EIXIS(MNP),GIP(MNP),GIPS(MNP)                                         RCL10760
      COMMON/INPUT4/DXI(MNP),PXIETA(MNP),DETA(MNP),DXIETA(MNP)                RCL10770
      COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP) RCL10780
      * ,TMAXI(MNP),TMAZI(MNP)                                               RCL10790
      COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP)                                RCL10800
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCXI(MNP),TOMAX,VM,NPI            RCL10810
      COMMON/STAT1/XTOP,YTOP                                                 RCL10820
      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL10830
      * ZETAM,TMAXAV,DXIM                                                   RCL10840
      COMMON/COEFL/EETA,EETAS,HZET,HX,EOMI,TO,QXIC,OMEGAO,CONS1,CONS2       RCL10850
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE                               RCL10860
      DIMENSION AI(MNP)                                                       RCL10870
C      CALCULATE THE ORTHONORMALIZING CONSTANT FOR P AND Q                  RCL10880
C
      DO 542 I=1,NP                                                       RCL10890
         CALL COUNT(X(I))
         A1(I) = Y(5,I)**2*HZET + HX*Y(6,I)**2
542 CONTINUE
      A = 0.D0
      DO 642 I=2,NP                                                       RCL10940
         DX2 = (X(I)-X(I-1))*0.5D0
         A = A + (A1(I)+A1(I-1))*DX2
642 CONTINUE
      A = DSQRT(A)
C      ORTHONORMALIZE THE SOLUTION EXCEPT SIGMA (POSITIVE)                 RCL11000
      DO 643 I=1,NP                                                       RCL11010
         Y(N,I) = DABS(Y(N,I))
         DO 643 K=1,N-1
            Y(K,I) = Y(K,I)/A
643 CONTINUE
C      OUTPUT RESULTS                                                       RCL11060
      WRITE(9,1000) NAME
1000 FORMAT(80A)
      WRITE(9,1001) NSEG,TLEN,WA,RHO0,RHOW,AI,CFLUID,PRESS,WT,TOMAX,VM,
      *XTOP*TLEN,YTOP*TLEN
C
      1001 FORMAT(1X,I3,9X,' = NUMBER OF RISER SEGMENTS'/
      *1X,D12.6,' = UNSTRETCHED RISER LENGTH IN M'/
      *1X,D12.6,' = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH RCL11140
      *IN N/M'/
      *1X,D12.6,' = INNER FLUID DENSITY IN KG/M3'/
      *1X,D12.6,' = SALT WATER DENSITY IN KG/M3'/
      *1X,D12.6,' = INNER CROSS SECTIONAL AREA IN M2'/
      *1X,D12.6,' = INNER FLUID SPEED IN M/S'/
      *1X,D12.6,' = INNER FLUID OVERPRESSURE IN /N/M2'/
      RCL11150
      RCL11160
      RCL11170
      RCL11180
      RCL11190
      RCL11200

```

```

*IX,D12.6,' = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N'/' RCL11210
*IX,D12.6,' = AVERAGE STATIC TENSION IN N'/' RCL11220
*IX,D12.6,' = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S'/' RCL11230
*IX,D12.6,' = X COORDINATE AT TOP IN M'/' RCL11240
*IX,D12.6,' = Y COORDINATE AT TOP IN M') RCL11250
C EVALUATE DIMENSIONAL NATURAL FREQUENCY RCL11260
SIGMAD = Y(N,1)*DSQRT(TOMAX/TMAXAV)/TLEN RCL11270
WRITE(9,1002) NSEG RCL11280
1002 FORMAT(//' DATA PER RISER SEGMENT FOR NSEG =',RCL11290
*' ',I3,' SEGMENTS/' DIMENSIONAL QUANTITY IRCL11300
*' ES IN THE S.I. SYSTEM/' RCL11310
*' RLENG      DXI      PXIETA     AO      WEIGHT      MASS   RCL11320
*' TMASS      AMAXI     AMAETA     AMAZI    TMAXI     TMAZI   RCL11330
*' )
TL = TOMAX/TLEN RCL11340
DO 1004 I=1,NSEG RCL11350
WRITE(9,1003) RLENG(I)*TLEN,DXI(I),PXIETA(I),AO(I),WEIGHT(I)*TL,RMRCL11370
*ASS(I),RMASST(I),AMAXI(I),AMAETA(I),AMAZI(I),TMAXI(I),TMAZI(I) RCL11380
1004 CONTINUE RCL11390
1003 FORMAT(12(1X,D10.4)) RCL11400
C RCL11410
WRITE(9,10022) RCL11420
10022 FORMAT(' EA      EIETA     EIETAS     EIXI      EIXIS   RCL11430
*' GIP      GIPS      DETA       JZI       AJZI     TJZI') RCL11440
DO 10023 I=1,NSEG RCL11450
WRITE(9,10024) EA(I),EIETA(I),EIETAS(I),EIXI(I),EIXIS(I),GIP(I), RCL11460
*GIPS(I),DETA(I),XJZI(I),AJZI(I),TJZI(I) RCL11470
10023 CONTINUE RCL11480
10024 FORMAT(11(1X,D10.4)) RCL11490
C RCL11500
WRITE(9,761) MODE,SIGMAD RCL11510
761 FORMAT(' *****' RCL11520
*' MODE NUMBER = ',I2,' NATURAL FREQUENCY RCL11530
*' CY = ',D10.4,' RAD/S'/' *****' RCL11540
*****') RCL11550
C RCL11560
WRITE(9,1009) NP RCL11570
1009 FORMAT(/' ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = ', RCL11580
*' I3,' POINTS'/' RCL11590
*' S      TENSION     QXI      OMEGAETA     PHI      P   RCL11600
*' Q      SIGMA') RCL11610
DO 1010 I=1,NP RCL11620
WRITE(9,1011) X(I),(Y(J,I),J=1,7) RCL11630
1010 CONTINUE RCL11640
1011 FORMAT(8(1X,D10.4)) RCL11650
DO 189 I=1,N-1 RCL11660
ABT(I) = ABT(I)/A RCL11670
189 CONTINUE RCL11680
WRITE(9,10119) (ABT(I),I=1,7) RCL11690
10119 FORMAT(' MAXIMUM ESTIMATED ERROR BY COMPONENTS'/11X,7(1X,D10.4)) RCL11700
C RCL11710
C RCL11720
C OUTPUT TO FILE CONNECTED TO DEVICE 11 RCL11730
C THIS CAN BE USED FOR PLOTS OR INPUT TO ANOTHER RUN OF RCLINDY1 RCL11740
C RCL11750
C RCL11760

```

```

CALL STRUC(VLOCKXI,NP,NPI)                                     RCL11770
  WRITE(11,36459) MCDE,NP,SIGMAD,XTOP,YTOP                   RCL11780
36459 FORMAT(1X,I2,1X,I3,3(1X,D10.4))                         RCL11790
    DO 3666 I=1,NP                                         RCL11800
      WRITE(11,3667) X(I),(Y(J,I),J=1,7),VLOCKXI(I)          RCL11810
3666  CONTINUE                                           RCL11820
3667  FORMAT(9(1X,D12.6))                                    RCL11830
C
      RETURN                                              RCL11840
      END                                                 RCL11850
C
      SUBROUTINE FCN(X,EPS,Y,F,N)                           RCL11860
C THIS SUBROUTINE EVALUATES THE FUNCTIONS USED BY D02RAF TO SOLVE RCL11890
C THE PROBLEM.                                            RCL11900
      IMPLICIT REAL*8(A-H,O-Z)                                RCL11910
      PARAMETER(MNP=151)                                     RCL11920
      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL11930
      *ZETAM,TMAXAV,DXIM                                     RCL11940
      COMMON/COEF1/EETA,EETAS,HZET,HX,EOMI,TO,QXIO,OMEGA0,CONS1,CONS2 RCL11950
      DIMENSION Y(N),F(N)                                    RCL11960
C LOCATE MESH POINT TO EVALUATE COEFFICIENTS               RCL11970
      CALL COUNT(X)                                         RCL11980
C
      F(1) = QXIO*Y(3) + OMEGA0*Y(2) + CONS2*Y(4)           RCL11990
      F(1) = F(1) - (HZETAM + EPS*(HZET-HZETAM))*Y(5)*Y(7)**2 RCL12000
C
      F(2) = -(1.D0 + EPS*(TO-1.D0))*Y(3) - OMEGA0*Y(1)       RCL12010
      F(2) = F(2) - CONS1*Y(4)                               RCL12020
      F(2) = F(2) - (1.D0 + EPS*(HX-1.D0))*Y(6)*Y(7)**2       RCL12030
C
      F(3) = - Y(2)/EETA - EETAS*Y(3)/EETA                  RCL12040
C
      F(4) = Y(3)                                         RCL12050
C
      F(5) = OMEGA0*Y(6) + EPS*EOMI*Y(1)                     RCL12060
C
      F(6) = Y(4) + Y(4)*EOMI*TO - OMEGA0*Y(5)             RCL12070
C
      F(7) = 0.D0                                         RCL12080
      RETURN                                              RCL12090
      END                                                 RCL12100
C
      SUBROUTINE G(EPS,YA,YB,BC,N)                           RCL12110
C BOUNDARY CONDITIONS USED BY D02RAF                      RCL12120
      IMPLICIT REAL*8(A-H,O-Z)                                RCL12130
      DIMENSION YA(N),YB(N),BC(N)                            RCL12140
      COMMON/BOUNDA/BOUND                                     RCL12150
C
      BC(1)=YA(4)                                         RCL12160
      BC(2)=YA(5)                                         RCL12170
      BC(3)=YA(6)                                         RCL12180
      BC(4)=YA(3) - BOUND                                  RCL12190
C
      BC(5)=YB(4)                                         RCL12200
      BC(6)=YB(5)                                         RCL12210
      BC(7)=YB(6)                                         RCL12220
C

```

```

C
      RETURN                                     RCL12330
      END                                         RCL12340
C
      SUBROUTINE JACOBF(X,EPS,Y,F,N)             RCL12350
      THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE EQUATIONS TO USE IN RCL12380
      NEWTON'S ITERATION.                         RCL12360
      IMPLICIT REAL*8(A-H,O-Z)                   RCL12370
      PARAMETER(MNP=151)                          RCL12390
      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL12420
      *ZETAM,TMAXAV,DXIM                         RCL12430
      COMMON/COEF1/EETA,EETAS,HZET,HX,EOM1,T0,QXIO,OMEGAO,CONS1,CONS2   RCL12440
      DIMENSION Y(N),F(N,N)                      RCL12450
C      LOCATE MESH POINT TO EVALUATE COEFFICIENTS
      CALL COUNT(X)                             RCL12460
C
      DO 817 I=1,N
        DO 817 M=1,N
          F(I,M) = 0.DO
 817 CONTINUE
C
      F(1,2) = OMEGA0                           RCL12530
      F(1,3) = QXIO                            RCL12540
      F(1,4) = CONS2                           RCL12550
      F(1,5) = -(HZETAM + EPS*(HZET-HZETAM))*Y(7)**2    RCL12560
      F(1,7) = -2.DO*Y(7)*(HZETAM+EPS*(HZET-HZETAM))*Y(5)  RCL12570
C
      F(2,1) = -OMEGA0                           RCL12580
      F(2,3) = -(1.DO + EPS*(T0-1.DO))         RCL12590
      F(2,4) = -CONS1                           RCL12600
      F(2,6) = -(1.DO + EPS*(HX-1.DO))*Y(7)**2    RCL12610
      F(2,7) = -2.DO*Y(7)*Y(6)*(1.DO + EPS*(HX-1.DO))  RCL12620
C
      F(3,2) = -1.DO/EETA                        RCL12630
      F(3,3) = -EETAS/EETA                       RCL12640
C
      F(4,3) = 1.DO                            RCL12650
C
      F(5,1) = EOM1*EPS                         RCL12660
      F(5,6) = OMEGA0                           RCL12670
C
      F(6,4) = 1.DO + EOM1*T0                  RCL12680
      F(6,5) = -OMEGA0                           RCL12690
      RETURN
      END
C
      SUBROUTINE JACOBG(EPS,YA,YB,AJ,BJ,N)       RCL12730
      THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE BOUNDARY CONDITIONS.RCL12800
      IMPLICIT REAL*8(A-H,O-Z)                   RCL12740
      DIMENSION YA(N),YB(N),AJ(N,N),BJ(N,N)     RCL12750
      DO 876 K=1,N
        DO 876 I=1,N
          AJ(K,I) = 0.DO
          BJ(K,I) = 0.DO
 876 CONTINUE
C

```

```

AJ(1,4) = 1.DO          RCL12890
AJ(2,5) = 1.DO          RCL12900
AJ(3,6) = 1.DO          RCL12910
AJ(4,3) = 1.DO          RCL12920
C                               RCL12930
BJ(5,4) = 1.DO          RCL12940
BJ(6,5) = 1.DO          RCL12950
BJ(7,6) = 1.DO          RCL12960
RETURN                     RCL12970
END                        RCL12980
RCL12990
C
SUBROUTINE JACEPS(X,EPS,Y,F,N)          RCL13000
THIS SUBROUTINE EVALUATES THE DERIVATIVES OF THE FUNCTIONS WITH RCL13010
RESPECT TO THE CONTINUATION PARAMETER EPS. RCL13020
IMPLICIT REAL*8(A-H,O-Z)          RCL13030
PARAMETER(NA=4,MNP=151)          RCL13040
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL13050
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL13060
*ZETAM,TMAXAV,DXIM          RCL13070
COMMON/COEF1/EETAS,HZET,HX,EOMI,TO,QXIO,OMEGAO,CONS1,CONS2 RCL13080
DIMENSION Y(N),F(N)          RCL13090
LOCATE MESH POINT TO EVALUATE THE COEFFICIENTS          RCL13100
CALL COUNT(X)          RCL13110
RCL13120
F(1) = -(HZET-HZETAM)*Y(5)*Y(7)**2          RCL13130
RCL13140
F(2) = -(HX-1.DO)*Y(6)*Y(7)**2 - (TO-1.DO)*Y(3)          RCL13150
RCL13160
F(3) = 0.DO          RCL13170
F(4) = 0.DO          RCL13180
F(5) = EOMI*Y(1)          RCL13190
RCL13200
F(6) = 0.DO          RCL13210
F(7) = 0.DO          RCL13220
RETURN                     RCL13230
END                        RCL13240
RCL13250
C
SUBROUTINE JACGEP(EPS,YA,YB,BCEP,N)          RCL13260
THIS SUBROUTINE EVALUATES THE DERIVATIVES OF THE BOUNDARY RCL13270
CONDITIONS WITH RESPECT TO THE CONTINUATION PARAMETER EPS. RCL13280
IMPLICIT REAL*8(A-H,O-Z)          RCL13290
DIMENSION YA(N),YB(N),BCEP(N)          RCL13300
DO 871 K=1,N          RCL13310
    BCEP(K) = 0.DO          RCL13320
871 CONTINUE          RCL13330
RETURN                     RCL13340
END                        RCL13350
RCL13360
C
SUBROUTINE INTERP(NP)          RCL13370
THIS SUBROUTINE INTERPOLATES THE RISER CHARACTERISTICS AND THE RCL13380
STATIC SOLUTION TO THE NEW NUMBER OF POINTS NP.          RCL13390
ASSUMPTION: NP .GE. NPI          RCL13400
IMPLICIT REAL*8(A-H,O-Z)          RCL13410
PARAMETER(MNP=151,NA=4,N=7)          RCL13420
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL13430
*ZETAM,TMAXAV,DXIM          RCL13440

```

```

COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI      RCL13450
COMMON/STAT2/CONST1(MNP),CONST2(MNP)                                RCL13460
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE                           RCL13470
DIMENSION HELP(MNP)                                                 RCL13480
C   INTERPOLATE STRUCTURAL DATA TO THE NEW NUMBER OF POINTS       RCL13490
    CALL STRUC(EPSETA,NP,NPI)                                         RCL13500
    CALL STRUC(EPSETS,NP,NPI)                                         RCL13510
    CALL STRUC(EOM,NP,NPI)                                            RCL13520
    CALL STRUC(HXI,NP,NPI)                                           RCL13530
    CALL STRUC(HZETA,NP,NPI)                                         RCL13540
    CALL STRUC(CONST1,NP,NPI)                                         RCL13550
    CALL STRUC(CONST2,NP,NPI)                                         RCL13560
    CALL STRUC(VLOCKXI,NP,NPI)                                        RCL13570
C   INTERPOLATE STATIC SOLUTION TO THE NEW NUMBER OF POINTS       RCL13580
    DO 459 K=1,3
        DO 458 I=1,NPI
            HELP(I) = STATIC(K,I)
458    CONTINUE
        CALL STRUC(HELP,NP,NPI)                                         RCL13620
        DO 457 I=1,NP
            STATIC(K,I) = HELP(I)
457    CONTINUE
459    CONTINUE
        DO 339 I=1,NP
            XI(I) = X(I)
339    CONTINUE
        RETURN
    END
C   SUBROUTINE STRUC(ARRAY,NP,NPOLD)                                 RCL13730
C   THIS SUBROUTINE INTERPOLATES A SERIES OF DIVISION POINTS TO A NEW RCL13750
C   SERIES OF DIVISION POINTS.                                       RCL13760
IMPLICIT REAL*8(A-H,O-Z)                                         RCL13770
PARAMETER(MNP=151,NA=4,N=7)                                         RCL13780
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE                           RCL13790
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI      RCL13800
DIMENSION ARRAY(MNP),HELP(MNP)                                     RCL13810
C
    HELP(1) = ARRAY(1)                                              RCL13820
    HELP(NP) = ARRAY(NPOLD)                                         RCL13830
    DO 82 I=2,NPOLD
    DO 81 K=2,NP-1
        IF ((X(K).GT.XI(I-1)).AND.(X(K).LT.XI(I))) THEN
            CONV = (X(K)-XI(I-1))*(ARRAY(I)-ARRAY(I-1))/(XI(I)-XI(I-1))
            HELP(K) = ARRAY(I-1) + CONV
        ELSE IF (X(K).EQ.XI(I)) THEN
            HELP(K) = ARRAY(I)
        END IF
81    CONTINUE
82    CONTINUE
    DO 85 K=1,NP
        ARRAY(K) = HELP(K)
85    CONTINUE
    RETURN
    END
C

```

FILE: RCLINDY1 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

SUBROUTINE COUNT(X) RCL14010
C THIS SUBROUTINE EVALUATES THE RISER CHARACTERISTICS AS WELL AS THE RCL14020
C STATIC SOLUTION AT A POINT X. RCL14030
C IT RETURNS THEIR VALUE IN COMMON BLOCK COEF1. RCL14040
C IMPLICIT REAL*8(A-H,O-Z) RCL14050
C PARAMETER(MNP=151,NA=4) RCL14060
C COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL14070
*ZETAM,TMAXAV,DXIM RCL14080
C COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL14090
C COMMON/COEF1/EETA,EETAS,HZET,HX,EOMI,TO,QXIO,OMEGAO,CONS1,CONS2 RCL14100
C COMMON/STAT2/CONST1(MNP),CONST2(MNP) RCL14110
C COMMON/COUN/ICOUNT RCL14120
M = ICOUNT RCL14130
IF (X.EQ.XI(M)) THEN RCL14140
  EETA = EPSETA(M) RCL14150
  EETAS = EPSETS(M) RCL14160
  HX = HXI(M) RCL14170
  HZET = HZETA(M) RCL14180
  EOMI = EOM(M) RCL14190
  TO = STATIC(1,M) RCL14200
  QXIO = STATIC(2,M) RCL14210
  OMEGA0 = STATIC(3,M) RCL14220
  CONS1 = CONST1(M) RCL14230
  CONS2 = CONST2(M) RCL14240
  ICOUNT = ICOUNT + 1 RCL14250
  IF(X.EQ.1.D0) ICOUNT = 1 RCL14260
ELSE IF ((X.LT.XI(M)).AND.(X.GT.XI(M-1))) THEN RCL14270
  DIFF = DABS(XI(M)-X) RCL14280
C THIS IF STATEMENT IS TO ACCOUNT FOR A COMPUTER INACCURACY IN RCL14290
C REPRESENTING AND EQUATING REAL NUMBERS. RCL14300
  IF (DIFF.LT.1.D-7) THEN RCL14310
    ICOUNT = ICOUNT + 1 RCL14320
    IF (XI(M).EQ.1.D0) ICOUNT = 1 RCL14330
  END IF RCL14340
  DX = (X-XI(M-1))/(XI(M)-XI(M-1)) RCL14350
  EETA = EPSETA(M-1) + (EPSETA(M)-EPSETA(M-1))*DX RCL14360
  EETAS = EPSETS(M-1) + (EPSETS(M)-EPSETS(M-1))*DX RCL14370
  HX = HXI(M-1) + (HXI(M)-HXI(M-1))*DX RCL14380
  HZET = HZETA(M-1) + (HZETA(M)-HZETA(M-1))*DX RCL14390
  EOMI = EOM(M-1) + (EOM(M)-EOM(M-1))*DX RCL14400
  TO = STATIC(1,M-1) + (STATIC(1,M)-STATIC(1,M-1))*DX RCL14410
  QXIO = STATIC(2,M-1) + (STATIC(2,M)-STATIC(2,M-1))*DX RCL14420
  OMEGA0 = STATIC(3,M-1) + (STATIC(3,M)-STATIC(3,M-1))*DX RCL14430
  CONS1 = CONST1(M-1) + (CONST1(M)-CONST1(M-1))*DX RCL14440
  CONS2 = CONST2(M-1) + (CONST2(M)-CONST2(M-1))*DX RCL14450
  ELSE RCL14460
C   WRITE(6,*) ' ERROR OCCURED IN COUNTER, NP =',M RCL14470
  END IF RCL14480
  RETURN RCL14490
END RCL14500

```

Chapter XV

Listing of Program RCFORC1 FORTRAN A

```

C RCFORC1 RCF00010
C THIS PROGRAM CALCULATES THE OUT OF PLANE LINEAR DYNAMIC RESPONSE OF A RCF00020
C COMPLIANT RISER WITH A 2-D STATIC CONFIGURATION FOR A GIVEN FREQUENCY RCF00030
C AND AMPLITUDE OF EXCITATION AT THE TOP (R OR B EXCITATION IS ALLOWED).RCF00040
C A NONUNIFORM GRID FINITE DIFFERENCES AND DEFERRED CORRECTION TECHNIQUE RCF00050
C IS USED.THE OUTPUT OF THIS PROGRAM CAN BE USED AS INPUT TO RCLINDY3 TORCF00060
C SOLVE THE OUT OF PLANE DYNAMICS PROBLEM WITH HOMOGENEOUS BOUNDARY RCF00070
C CONDITIONS (EIGENPROBLEM). RCF00080
C THIS PROGRAM WILL BE PARTICULARLY USEFUL IN THE CASE OF ZERO CURRENT RCF00090
C VELOCITY WHERE THE ASYMPTOTIC THEORY IS NO LONGER VALID. RCF00100
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED. RCF00110
C***** RCF00120
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY RCF00130
C ALL RIGHTS RESERVED. RCF00140
C***** RCF00150
C PROGRAMMER GEORGE A. KRIEZIS OCTOBER 5, 1985 M.I.T. RCF00160
C***** RCF00170
C RCF00180
C DEFINITION OF DEVICES: RCF00190
C DEVICE 5 : INPUT FROM TERMINAL RCF00200
C DEVICE 6 : OUTPUT TO TERMINAL RCF00210
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT (LRECL=80) RCF00220
C DEVICE 10 : INPUT FROM FILE CONTAINING APPROXIMATE N-D STATIC RCF00230
C SOLUTION CREATED BY RCSTAT2D (LRECL=132) RCF00240
C DEVICE 11 : OUTPUT TO FILE FOR PLOTS OR INPUT TO A RUN RCF00250
C OF RCLINDY3 (LRECL=117) RCF00260
C RCF00270
C COMMON BLOCK CONTENTS (OVERALL REFERENCE): RCF00280
C SOLUT = SOLUTION MATRICES RCF00290
C STAT = STATIC COMPLIANT RISER SOLUTION RCF00300
C STAT2 = FUNCTIONS OF STATIC RESULTS RCF00310
C INPUT1 = RISER LENGTH, AVERAGE AND TOTAL WEIGHT AND NUMBER OF SEGMENTS RCF00320
C INPUTL = RISER SEGMENTS LENGTH RCF00330
C COEF = NONDIMENSIONAL RISER CHARACTERISTICS USED IN THE EQUATIONS RCF00340
C COEF1 = NONDIMENSIONAL RISER CHARACTERISTICS AT A SPECIFIC POINT RCF00350
C COUN = INTEGER COUNTING VARIABLE TO DETERMINE EACH DIVISION POINT RCF00360
C FREQ = NONDIMENSIONAL FREQUENCY OF EXCITATION RCF00370
C RCF00380
C IMPLICIT REAL*8(A-H,O-Z) RCF00390
PARAMETER(N=6,MNP=151,NA=4,IY=6) RCF00400
PARAMETER(LW=MNP*(3*N**2+5*N+2)+3*N**2+5*N) RCF00410
PARAMETER(LIW=MNP*(2*N+1)+N) RCF00420
DIMENSION W(LW),IW(LIW),C(N,N),D(N,N),GAM(N) RCF00430
COMMON/SOLUT/X(MNP),Y(N,MNP),MODE RCF00440
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKI(MNP),TOMAX,VM,NPI RCF00450
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCF00460
*N),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV RCF00470
COMMON/COUN/ICOUNT RCF00480
COMMON/FREQ/XLAMDA RCF00490
C EXTERNAL SUBROUTINES USED BY NAG LIBRARY RCF00500
EXTERNAL FCNF,FCNG RCF00510
DATA NOUT /6/ RCF00520
RCF00530
C READ2D READS 2-D STATIC SOLUTION CALCULATED FROM RCSTAT2D PROGRAM.RCF00540
C IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC EFFECTIVE TENSION RCF00550
RCF00560

```

```

CALL READ2D(NPI) RCF00570
C RCF00580
C CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE RCF00590
C NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE EQUATIONS. RCF00600
C RCF00610
C CALL CHARAC(TLENG,IC,NPI) RCF00620
C IF(IC.EQ.0) STOP RCF00630
C RCF00640
C DEFINITIONS OF PARAMETERS ... RCF00650
C N = NUMBER OF EQUATIONS TO BE SOLVED BY DO2GBF RCF00660
C MNP= MAXIMUM NUMBER OF PERMITTED POINTS IN THE NON-UNIFORM RCF00670
C F.D. MESH ( MNP >= 32) RCF00680
C IF(MNP.LT.32) THEN RCF00690
C     WRITE(6,1257) MNP RCF00700
1257    FORMAT(' MNP = ',I3,' IS NOT PERMISSIBLE') RCF00710
C     STOP RCF00720
C     ENDIF RCF00730
C PARAMETER CONTROLLING MONITORING OF CALCULATIONS RCF00740
C IFAIL=111 RCF00750
C BOUNDARY POINTS FOR TWO-POINT BOUNDARY VALUE PROBLEM RCF00760
C A=0.DO RCF00770
C B=1.DO RCF00780
C RCF00790
C     WRITE(6,*)' INPUT EXCITATION FREQUENCY IN RAD/S' RCF00800
C     READ(5,*) SIGM RCF00810
C     NONDIMENSIONALIZE SIGMA RCF00820
C     SIGMA = SIGM*TLENG*DSQRT(TMAXAV/TOMAX) RCF00830
C     XLAMDA = SIGMA**2 RCF00840
C     WRITE(6,*)' INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR R' RCF00850
C     READ(5,*) BOUNDR RCF00860
C     WRITE(6,*)' INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR B' RCF00870
C     READ(5,*) BOUNDB RCF00880
C     WRITE(6,*)' INPUT TOLERANCE FOR CONVERGENCE OF ITERATIONS' RCF00890
C     WRITE(6,*)' SUGGESTED VALUE IS THE EXCITATION AMPLITUDE OR 1/10 OFRCF00900
C     * THE EXCITATION AMPLITUDE' RCF00910
C     READ(5,*) TOL RCF00920
C     WRITE(6,*)' INPUT MODE NUMBER CORRESPONDING TO FREQUENCY OF EXCITARCF00930
C     *TION' RCF00940
C     READ(5,*) MODE RCF00950
C     NP = NPI RCF00960
C     DO 632 I=1,N RCF00970
C         GAM(I) = 0.DO RCF00980
C         DO 632 K=1,N RCF00990
C             C(I,K)=0.DO RCF01000
C             D(I,K)=0.DO RCF01010
C 632 CONTINUE RCF01020
C     DEFINE BOUNDARY CONDITIONS USED IN THE EQUATIONS. RCF01030
C     GAM(5) = BOUNDB RCF01040
C     GAM(6) = BOUNDR RCF01050
C     C(1,4) = 1.DO RCF01060
C     C(2,5) = 1.DO RCF01070
C     C(3,6) = 1.DO RCF01080
C     D(4,4) = 1.DO RCF01090
C     D(5,5) = 1.DO RCF01100
C     D(6,6) = 1.DO RCF01110
C     SET COUNTING VARIABLE RCF01120

```

```

        ICOUNT = 1                                RCF01130
C                                                 RCF01140
        CALL X04AAF(1,NOUT)                         RCF01150
        CALL X04ABF(1,NOUT)                         RCF01160
C                                                 RCF01170
C                                                 RCF01180
C                                                 RCF01190
C                                                 RCF01200
C                                                 RCF01210
C                                                 RCF01220
C                                                 RCF01230
C                                                 RCF01240
C                                                 RCF01250
9000  FORMAT(' IFAIL =',I3)                     RCF01260
C                                                 RCF01270
C                                                 RCF01280
C                                                 RCF01290
C                                                 RCF01300
C                                                 RCF01310
C                                                 RCF01320
C                                                 RCF01330
C                                                 RCF01340
C                                                 RCF01350
C                                                 RCF01360
C                                                 RCF01370
C                                                 RCF01380
C                                                 RCF01390
C                                                 RCF01400
C                                                 RCF01410
C                                                 RCF01420
CHARACTER*80 NAME                           RCF01430
COMMON/INPUT1/TLEN,WA,WT,NSEG                RCF01440
COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)          RCF01450
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCF01460
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCFO1460
*NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV      RCF01470
COMMON/STAT2/CONST1(MNP),CONST2(MNP)          RCF01480
COMMON/SOLUT/X(MNP),Y(N,MNP),MODE           RCF01490
DIMENSION RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP) RCF01500
*,TMAXI(MNP),TMAZI(MNP),TMAETA(MNP)         RCF01510
DIMENSION XJZI(MNP),AJZI(MNP),TJZI(MNP)      RCF01520
DIMENSION DXI(MNP),PXEITA(MNP),DETA(MNP),DXIETA(MNP),AO(MNP) RCF01530
DIMENSION WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP), RCF01540
*EIXIS(MNP),GIP(MNP),GIPS(MNP)               RCF01550
DIMENSION TENO(MNP),QXO(MNP),TOS(MNP),QXIOS(MNP) RCF01560
C                                                 RCF01570
C                                                 RCF01580
C                                                 RCF01590
READ (8,1000) NAME                           RCF01600
1000  FORMAT(80A)                            RCF01610
READ (8,1008) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS RCF01620
1008  FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6)) RCF01630
TLENG = TLEN                                  RCF01640
DO 1502 I=1,NSEG                            RCF01650
READ (8,1003) RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXI(*),PXEITA(I),EA(I),EIETA(I),AO(I),EIETAS(I) RCF01660
1502  CONTINUE                               RCF01670
1003  FORMAT(5(1X,D12.6)/6(1X,D12.6))       RCF01680

```

FILE: RCFORC1 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

C
      DO 1332 I=1,NSEG
      READ(8,1333) AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),AMARCF01710
      *ZI(I),XJZI(I),AJZI(I)
1332  CONTINUE
1333  FORMAT(6(1X,D12.6)/3(1X,D12.6))
C
      IC=1
C
7650  WRITE(6,7651)
7651  FORMAT(' DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS'/
      *' IF YES INPUT 1 , IF NO INPUT 0')
      READ(5,*) IPRINT
      IF((IPRINT.NE.1).AND.(IPRINT.NE.0)) GOTO 7650
      IF(IPRINT.EQ.1) THEN
C
      WRITE(6,1000) NAME
      WRITE(6,2500)
2500  FORMAT(' NSEG      TLEN          WA          RHOO        AI
      * CFLUID      PRESS')
      WRITE(6,2001) NSEG,TLEN,WA,RHOO,AI,CFLUID,PRESS
2001  FORMAT(1X,I3,6(1X,D12.6))
      WRITE(6,3400)
3400  FORMAT('   I     RLENG         RMASS        RMASST      AMAXI
      * WEIGHT'   /'    DXI          PXIETA      EA
      *           AO          EIETAS')
      DO 3002 I=1,NSEG
      WRITE(6,3003) I,RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXIRCFC01960
      *(I),PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I)
3002  CONTINUE
3003  FORMAT(1X,I3,5(1X,D12.6)/4X,6(1X,D12.6))
C
      WRITE(6,1334)
1334  FORMAT('   I     AMAETA        DETA        EIXI        EIXIS
      * GIP        GIPS'/'        AMAZI        JZI
      DO 1335 I=1,NSEG
      WRITE(6,1336) I,AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),RCF02050
      *AMAZI(I),XJZI(I),AJZI(I)
1335  CONTINUE
1336  FORMAT(1X,I3,6(1X,D12.6)/4X,3(1X,D12.6))
C
1701  WRITE(6,1700)
1700  FORMAT(' INPUT 1 IF RISER DATA ARE CORRECT'/' INPUT 0 TO STOP')
      READ(5,*) IC
      IF((IC.NE.0).AND.(IC.NE.1)) GOTO 1701
      IF(IC.EQ.0) RETURN
C
      ENDIF
C
C      NON - DIMENSIONALIZATIONS
C
      GRAV=9.81DC
      WAM=WA/GRAV
      WT=WA*TLEN
      XPI=4.D0*DATAN(1.D0)
      XPI2=XPI/2.D0

```

```

RHOW=1.025D3                               RCF02250
C                                         RCF02260
C   NONDIMENSIONALIZE CORRECTLY QXIO, TO FROM 2-D STATIC SOLUTION RCF02270
C                                         RCF02280
C                                         RCF02290
C                                         RCF02300
C                                         RCF02310
C                                         RCF02320
C                                         RCF02330
C                                         RCF02340
229  CONTINUE                               RCF02350
C                                         RCF02360
C   NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS RCF02370
C                                         RCF02380
C                                         RCF02390
C                                         RCF02400
C                                         RCF02410
C                                         RCF02420
C                                         RCF02430
C                                         RCF02440
C                                         RCF02450
C                                         RCF02460
C                                         RCF02470
C                                         RCF02480
C                                         RCF02490
C                                         RCF02500
C                                         RCF02510
C                                         RCF02520
C                                         RCF02530
C                                         RCF02540
C                                         RCF02550
C                                         RCF02560
C                                         RCF02570
C                                         RCF02580
C                                         RCF02590
C                                         RCF02600
C                                         RCF02610
C                                         RCF02620
C                                         RCF02630
C                                         RCF02640
C   CALCULATE DERIVATIVES OF STATIC QUANTITIES RCF02650
C                                         RCF02660
DO 737 I=1,NP                               RCF02670
  TENO(I) = STATIC(1,I)
  QXO(I) = STATIC(2,I)
737  CONTINUE                               RCF02680
                                         RCF02690
                                         RCF02700
                                         RCF02710
C   EVALUATE FUNCTIONS OF STATIC RESULTS. RCF02720
DO 56  I=1,NP                               RCF02730
  CONST1(I) = TOS(I) - STATIC(2,I)*STATIC(3,I)
  CONST2(I) = QXIOS(I) + STATIC(1,I)*STATIC(3,I)
56   CONTINUE                               RCF02740
                                         RCF02750
C   SEG(I)=LEFT ORDINATE OF SEGMENT I. RCF02760
C   SEG(1)=0.D0                               RCF02770
C   SEG(NSEG+1)=1.D0                           RCF02780
                                         RCF02790
                                         RCF02800

```

```

DO 4000 I=2,NSEG
SEG(I)=RLENG(I-1)+SEG(I-1) RCF02810
4000 CONTINUE RCF02820
C INTRERPULATE STRUCTURAL DIMENSIONS TO NP POINTS RCF02830
C ASSUMPTION: NUMBER OF RISER SEGMENTS WITH DIFFERENT RCF02840
C CHARACTERISTICS IS SMALLER THAN THE STATIC SOLUTION POINTS RCF02850
C NSEG < NPI RCF02860
C RCF02870
IF (NSEG.GE.NPI) THEN RCF02880
WRITE(6,188) RCF02890
188 FORMAT(' NSEG => NPI, PROGRAM STOPS')
IC = 0 RCF02910
RETURN RCF02920
END IF RCF02930
CALL STRUCT(EPSETA,X,NP) RCF02940
CALL STRUCT(EPSETS,X,NP) RCF02950
CALL STRUCT(EPSPSI,X,NP) RCF02960
CALL STRUCT(EPSPIS,X,NP) RCF02970
CALL STRUCT(EPSXI,X,NP) RCF02980
CALL STRUCT(EPSXIS,X,NP) RCF02990
CALL STRUCT(HETA,X,NP) RCF03000
CALL STRUCT(TLAMB2,X,NP)
RETURN RCF03010
END RCF03020
RCF03030
RCF03040
C SUBROUTINE READ2D(NP) RCF03050
C THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM RCF03060
C DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION.RCF03070
C IMPLICIT REAL*8(A-H,O-Z) RCF03080
C PARAMETER(N=6,MNP=151,NA=4) RCF03090
C COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCF03100
C COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCF03110
C RCF03120
C STATIC(1,I) = STATIC TENSION TO RCF03130
C STATIC(2,I) = STATIC SHEAR FORCE IN THE XI DIRECTION RCF03140
C STATIC(3,I) = STATIC OMEGA IN THE ETA DIRECTION RCF03150
C STATIC(4,I) = STATIC ANGLE PHI RCF03160
C RCF03170
WRITE(6,2000) MNP RCF03180
2000 FORMAT(' MNP=',I3) RCF03190
C READ STATIC SOLUTION RCF03200
READ(10,36459) NP,VM RCF03210
36459 FORMAT(1X,I3,1X,D12.6) RCF03220
WRITE(6,2311) NP,VM RCF03230
2311 FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
*' NP =',I3/
*' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S ,VM =',D12.6) RCF03240
C RCF03250
IF((NP.LT.4).OR.(NP.GT.MNP)) THEN RCF03260
ICCC=0 RCF03270
WRITE(6,12439) RCF03280
12439 FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS')
RETURN RCF03290
ENDIF RCF03300
C RCF03310
C RCF03320
C RCF03330
C RCF03340
C RCF03350
C RCF03360
C READING FROM DEVICE 10

```

```

C                                         RCF03370
DO 1021 I=1,NP                         RCF03380
READ(10,1033) X(I),STATIC(1,I),STATIC(2,I),STATIC(3,I),STATIC(4,I) RCF03390
*,XCOOR,YCOOR,STRARC,TENSI,VLOCKI(I)
XI(I) = X(I)                           RCF03400
1021 CONTINUE                           RCF03410
1033 FORMAT(10(1X,D12.6))               RCF03420
C EVALUATE MAXIMUM STATIC EFFECTIVE TENSION RCF03430
TOMAX=DMAX1(STATIC(1,1),STATIC(1,2))   RCF03440
DO 9859 I=3,NP                         RCF03450
TOMAX=DMAX1(TOMAX,STATIC(1,I))        RCF03460
9859 CONTINUE                           RCF03470
C                                         RCF03480
WRITE(6,1654) TOMAX                   RCF03490
1654 FORMAT(' 2-D STATIC SOLUTION SUCCESFULLY READ'/' MAXIMUM STATIC EFR RCF03510
*EFFECTIVE TENSION/WA*L = ',D10.4)    RCF03520
C                                         RCF03530
RETURN                                 RCF03540
END                                     RCF03550
C                                         RCF03560
SUBROUTINE STRUCT(ARRAY,X,NP)          RCF03570
C THIS SUBROUTINE CHANGES A SERIES OF DATA GIVEN IN SEGMENTS RCF03580
C TO A SERIES OF DATA CORRESPONDING TO SPECIFIC POINTS RCF03590
C ASSUMPTION: NSEG < NP                RCF03600
IMPLICIT REAL*8(A-H,O-Z)              RCF03610
PARAMETER(MNP=151)                     RCF03620
COMMON/INPUT1/TLEN,WA,WT,NSEG          RCF03630
COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)   RCF03640
DIMENSION ARRAY(MNP),HELP(MNP),X(MNP) RCF03650
C                                         RCF03660
IF(NSEG.EQ.1) THEN                    RCF03670
  DO 83 I=1,NP                         RCF03680
    HELP(I) = ARRAY(I)                 RCF03690
83  CONTINUE                           RCF03700
ELSE                                     RCF03710
  HELP(1) = ARRAY(1)                  RCF03720
  HELP(NP) = ARRAY(NSEG)             RCF03730
  I=2                                    RCF03740
  DO 84 K=2,NP-1                      RCF03750
    IF ((X(K).GT.SEG(I-1)).AND.(X(K).LT.SEG(I))) THEN RCF03760
      HELP(K) = ARRAY(I-1)            RCF03770
    ELSE IF (X(K).EQ.SEG(I)) THEN    RCF03780
      HELP(K) = 0.5D0*(ARRAY(I-1) + ARRAY(I)) RCF03790
    ELSE IF (X(K).GT.SEG(I)) THEN    RCF03800
      HELP(K) = ARRAY(I)            RCF03810
      I = I + 1                      RCF03820
    END IF                            RCF03830
84  CONTINUE                           RCF03840
  END IF                                RCF03850
  DO 85 K=1,NP                         RCF03860
    ARRAY(K) = HELP(K)                 RCF03870
85  CONTINUE                           RCF03880
  RETURN                                RCF03890
  END                                   RCF03900
C                                         RCF03910
SUBROUTINE DER1(ARRAY,X,DERIV,NP)      RCF03920

```

```

C      THIS SUBROUTINE EVALUATES THE FIRST DERIVATIVE OF A SERIES OF DATARCF03930
C      POINTS USING A SECOND ORDER APPROXIMATION (NONUNIFORM GRID FINITE RCF03940
C      DIFFERENCES).                                                 RCF03950
      IMPLICIT REAL*8(A-H,O-Z)                                         RCF03960
      PARAMETER (MNP=151)                                              RCF03970
      DIMENSION ARRAY(MNP),DERIV(MNP),X(MNP)                           RCF03980
      C USE FIRST ORDER FOR END POINTS DERIVATIVES
      DERIV(1) = (-ARRAY(1) + ARRAY(2))/(X(2)-X(1))                  RCF03990
      DERIV(NP) = (-ARRAY(NP-1) + ARRAY(NP))/(X(NP)-X(NP-1))          RCF04000
      C
      DO 836 I=2,NP-1                                               RCF04010
        DX = X(I) - X(I-1)                                         RCF04020
        DX1 = X(I+1) - X(I)                                         RCF04030
        D = DX/DX1                                                 RCF04040
        DERIV(I) = (D*(ARRAY(I+1)-ARRAY(I)) - (ARRAY(I-1)-ARRAY(I))/D) RCF04050
        DERIV(I) = DERIV(I)/(DX+DX1)                                     RCF04060
836  CONTINUE
      RETURN
      END
      C
      SUBROUTINE OUTPUT(NP)
      C THIS SUBROUTINE OUTPUTS THE RESULTS IN A FORM TO BE USED AS INPUT RCF04140
      C FOR A RUN OF RCLINDY3.                                         RCF04150
      IMPLICIT REAL*8(A-H,O-Z)                                         RCF04160
      PARAMETER(MNP=151,NA=4,N=6)                                       RCF04170
      COMMON/INPUT1/TLEN,WA,WT,NSEG                                     RCF04180
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI    RCF04190
      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCF04200
      *NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV             RCF04210
      COMMON/SOLUT/X(MNP),Y(N,MNP),MODE                                RCF04220
      COMMON/FREQ/XLAMDA                                             RCF04230
      C
      SIGMAD = DSQRT(XLAMDA*TOMAX/TMAXAV)/TLEN                      RCF04240
      SIGMA = DSQRT(XLAMDA)                                           RCF04250
      C
      C OUTPUT TO FILE CONNECTED TO DEVICE 11                         RCF04270
      C THIS CAN BE USED FOR PLOTS OR INPUT TO A RUN OF RCLINDY3     RCF04280
      C
      WRITE(11,36459) MODE,NP,SIGMAD                                 RCF04290
36459  FORMAT(1X,I2,1X,I3,1X,D10.4)                               RCF04300
      DO 3666 I=1,NP                                              RCF04310
        WRITE(11,3667) X(I),(Y(J,I),J=1,6),SIGMA                   RCF04320
3666  CONTINUE
3667  FORMAT(8(1X,D12.6))
      C
      RETURN
      END
      C
      SUBROUTINE FCNF(X,F)
      C THIS SUBROUTINE EVALUATES THE MATRIX F(X) AT A GENERAL POINT X RCF04430
      C THIS MATRIX IS USED BY NAG SUBROUTINE D02GBF.                 RCF04440
      IMPLICIT REAL*8(A-H,O-Z)                                         RCF04450
      PARAMETER(MNP=151,N=6)                                         RCF04460
      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCF04480
      *
```

```

*NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV           RCF04490
COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,TO,QXIO,OMEGAO,RCF04500
*CONS1,CONS2                                                 RCF04510
COMMON/FREQ/XLAMDA                                         RCF04520
DIMENSION F(N,N)                                            RCF04530
C LOCATE MESH POINT TO EVALUATE COEFFICIENTS                RCF04540
CALL COUNT(X)                                               RCF04550
C
DO 875 I=1,N
  DO 875 K=1,N
    F(I,K) = 0.D0
875 CONTINUE
C
F(1,2) = -QXIO                                         RCF04620
F(1,3) = TO                                              RCF04630
F(1,4) = CONS1                                           RCF04640
F(1,5) = -CONS2                                           RCF04650
F(1,6) = -XLAMDA*HH                                     RCF04660
C
F(2,2) = -EPIS/EPI                                       RCF04680
F(2,3) = (EXI-EETA)*OMEGAO/EPI                         RCF04690
F(2,5) = -XLAMDA/TLAM2/EPI                            RCF04700
C
F(3,1) = 1.DO/EXI                                       RCF04720
F(3,2) = (EETA-EPI)*OMEGAO/EXI                         RCF04730
F(3,3) = -EXIS/EXI                                      RCF04740
C
F(4,3) = 1.DO                                           RCF04760
F(4,5) = -OMEGAO                                         RCF04770
C
F(5,2) = 1.DO                                           RCF04790
F(5,4) = OMEGA0                                         RCF04800
C
F(6,4) = -1.DO                                           RCF04820
RETURN
END
C
SUBROUTINE FCNG(X,G)                                     RCF04850
C THIS SUBROUTINE EVALUATES VECTOR G(X) AT A GENERAL POINT X. RCF04860
C IT IS USED BY NAG SUBROUTINE DO2GBF.                   RCF04870
C IMPLICIT REAL*8(A-H,O-Z)                               RCF04880
PARAMETER(N=6)                                         RCF04890
C DIMENSION G(N)                                         RCF04900
DO 817 I=1,N
  G(I) = 0.D0
817 CONTINUE
RETURN
END
C
SUBROUTINE COUNT(X)                                     RCF04980
C THIS SUBROUTINE EVALUATES THE RISER CHARACTERISTICS AS WELL AS THE RCF04990
C STATIC SOLUTION AT A GENERAL POINT X.                  RCF05000
C IT RETURNS THEIR VALUE IN COMMON BLOCK COEF1.          RCF05010
C IMPLICIT REAL*8(A-H,O-Z);                           RCF05020
PARAMETER(MNP=151,NA=4)                                RCF05030
COMMON/COEF1/EPSKI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSP1(MNP) RCF05040

```

```

*NP),EPSPI(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV           RCF05050
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI   RCF05060
COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,TO,QXIO,OMEGAO,RCF05070
*CONS1,CONS2
COMMON/STAT2/CONST1(MNP),CONST2(MNP)                         RCF05080
COMMON/COUN/ICOUNT                                         RCF05090
M = ICOUNT                                                 RCF05100
IF (X.EQ.XI(M)) THEN                                         RCF05110
  EETA = EPSETA(M)                                         RCF05120
  EETAS = EPSETS(M)                                         RCF05130
  EXI = EPSXI(M)                                           RCF05140
  EXIS = EPSXIS(M)                                         RCF05150
  EPI = EPSPPI(M)                                         RCF05160
  EPIS = EPSPSIS(M)                                         RCF05170
  HH = HETA(M)                                              RCF05180
  TLAM2 = TLAMB2(M)                                         RCF05190
  TO = STATIC(1,M)                                         RCF05200
  QXIO = STATIC(2,M)                                         RCF05210
  OMEGAO = STATIC(3,M)                                         RCF05220
  CONS1 = CONST1(M)                                         RCF05230
  CONS2 = CONST2(M)                                         RCF05240
  ICOUNT = ICOUNT + 1                                       RCF05250
  IF(X.EQ.1.D0) ICOUNT = 1                                 RCF05260
ELSE IF ((X.LT.XI(M)).AND.(X.GT.XI(M-1))) THEN          RCF05270
  DIFF = DABS(XI(M)-X)                                     RCF05280
C THIS IF STATEMENT IS TO ACCOUNT FOR A COMPUTER INACCURACY IN      RCF05290
C REPRESENTING AND EQUATING REAL NUMBERS.                         RCF05300
  IF (DIFF.LT.1.D-7) THEN                                    RCF05310
    ICOUNT = ICOUNT + 1                                     RCF05320
    IF (XI(M).EQ.1.D0) ICOUNT = 1                         RCF05330
  END IF
  DX = (X-XI(M-1))/(XI(M)-XI(M-1))                      RCF05350
  EXI = EPSXI(M-1) + (EPSXI(M)-EPSXI(M-1))*DX           RCF05360
  EXIS = EPSXIS(M-1) + (EPSXIS(M)-EPSXIS(M-1))*DX       RCF05370
  EETA = EPSETA(M-1) + (EPSETA(M)-EPSETA(M-1))*DX       RCF05380
  EETAS = EPSETS(M-1) + (EPSETS(M)-EPSETS(M-1))*DX       RCF05390
  EPI = EPSPPI(M-1) + (EPSPPI(M)-EPSPPI(M-1))*DX        RCF05400
  EPIS = EPSPSIS(M-1) + (EPSPSIS(M)-EPSPSIS(M-1))*DX     RCF05410
  HH = HETA(M-1) + (HETA(M)-HETA(M-1))*DX                RCF05420
  TLAM2 = TLAMB2(M-1) + (TLAMB2(M)-TLAMB2(M-1))*DX      RCF05430
  TO = STATIC(1,M-1) + (STATIC(1,M)-STATIC(1,M-1))*DX     RCF05440
  QXIO = STATIC(2,M-1) + (STATIC(2,M)-STATIC(2,M-1))*DX   RCF05450
  OMEGAO = STATIC(3,M-1) + (STATIC(3,M)-STATIC(3,M-1))*DX   RCF05460
  CONS1 = CONST1(M-1) + (CONST1(M)-CONST1(M-1))*DX        RCF05470
  CONS2 = CONST2(M-1) + (CONST2(M)-CONST2(M-1))*DX        RCF05480
  ELSE
    WRITE(6,*) ' ERROR OCCURED IN COUNTER, NP =',M          RCF05500
  END IF
  RETURN
END

```

Chapter XVI

Listing of Program RCLINDY3 FORTRAN A

```

C RCLINDY3                                     RCL00010
C THIS PROGRAM CALCULATES THE OUT OF PLANE LINEAR DYNAMIC RESPONSE OF A RCL00020
C COMPLIANT RISER WITH A 2-D STATIC CONFIGURATION BASED ON AN RCL00030
C EMBEDDING TECHNIQUE. THE INITIAL EMBEDDED APPROXIMATION IS PREPARED RCL00040
C BY PROGRAM RCFORC1 AND THE FINAL SOLUTION IS OBTAINED USING MODIFIED RCL00050
C NEWTON'S ITERATION AND A NON-UNIFORM GRID FINITE DIFFERENCE METHOD. RCL00060
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED. RCL00070
C*****RCL00080
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY RCL00090
C ALL RIGHTS RESERVED. RCL00100
C*****RCL00110
C PROGRAMMER    GEORGE A. KRIEZIS      OCTOBER 5, 1985      M.I.T.      RCL00120
C*****RCL00130
C
C DEFINITION OF DEVICES: RCL00140
C DEVICE 5 : INPUT FROM TERMINAL RCL00150
C DEVICE 6 : OUTPUT TO TERMINAL RCL00160
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT (LRECL=80) RCL00180
C DEVICE 9 : COMPLETE OUTPUT TO FILE (LRECL=132) RCL00190
C DEVICE 10 : INPUT FROM FILE CONTAINING APPROXIMATE N-D STATIC RCL00200
C             SOLUTION CREATED BY RCSTAT2D (LRECL=132) RCL00210
C DEVICE 11 : OUTPUT TO FILE FOR PLOTS OR INPUT TO A NEW RUN RCL00220
C             OF RCLINDY3 (LRECL=117) RCL00230
C DEVICE 12 : INPUT FROM FILE CONTAINING APPROXIMATE EMBEDDED SOLUTION RCL00240
C             CREATED BY RCFORC1 (LRECL=117) RCL00250
C
C COMMON BLOCK CONTENTS (OVERALL REFERENCE): RCL00260
C SOLUT = INITIAL EMBEDDED APPROXIMATION AND SOLUTION MATRICES RCL00280
C STAT = STATIC COMPLIANT RISER SOLUTION RCL00290
C STAT2 = FUNCTIONS OF STATIC RESULTS RCL00300
C INPUT0 = OUTPUT FILE HEADING RCL00310
C INPUT1 = RISER LENGTH, AVERAGE AND TOTAL WEIGHT AND NUMBER OF SEGMENTS RCL00320
C INPUT2 = RISER TUBE INTERIOR CHARACTERISTICS RCL00330
C INPUTL = RISER SEGMENTS LENGTH RCL00340
C INPUT3 = WEIGHT, STIFFNESSES AND STIFFNESS DERIVATIVES RCL00350
C INPUT4 = CHARACTERISTIC DIAMETERS AND PERIMETER OF RISER TUBE. RCL00360
C INPUT5 = MASS AND ADDED MASS CHARACTERISTICS IN THREE DIRECTIONS RCL00370
C INPUT6 = RISER POLAR AND ADDED POLAR MOMENT OF INERTIA. RCL00380
C COEF = nondimensional riser characteristics used in the equations RCL00390
C COEF1 = nondimensional riser characteristics at a specific point RCL00400
C CONST = problem constants (pi, gravity, water density) RCL00410
C BOUND1 = boundary condition for omega xi at s=0 and embedded boundary RCL00420
C             conditions from program RCFORC1 RCL00430
C COUN = integer counting variable to determine each division point RCL00440
C
C IMPLICIT REAL*8(A-H,O-Z) RCL00450
C PARAMETER(N=7,MNP=151,NA=4,IY=7) RCL00460
C PARAMETER(LWORK=MNP*(3*N**2+6*N+2)+4*N**2+3*N) RCL00470
C PARAMETER(LIWORK=MNP*(2*N+1)+N) RCL00480
C DIMENSION WORK(LWORK),IWORK(LIWORK) RCL00490
C COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL00500
C COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL00510
C COMMON/COUN/ICOUNT RCL00520
C EXTERNAL SUBROUTINES USED BY NAG LIBRARY RCL00530
C EXTERNAL FCN,G,JACEPS,JACGEP,JACCBF,JACOBG RCL00540
C DATA NOUT /6/ RCL00550
C                                         RCL00560

```

```

C          RCL00570
C          READ2D READS 2-D STATIC SOLUTION CALCULATED FROM RCSTAT2D PROGRAM. RCL00580
C          IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC EFFECTIVE TENSION RCL00590
C          RCL00600
C          CALL READ2D(NPI) RCL00610
C          RCL00620
C          CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE RCL00630
C          NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE EQUATIONS. RCL00640
C          RCL00650
C          CALL CHARAC(TLENG,IC,NPI) RCL00660
C          IF(IC.EQ.0) STOP RCL00670
C          RCL00680
C          DEFINITIONS OF PARAMETERS ... RCL00690
C          N = NUMBER OF EQUATIONS TO BE SOLVED BY DO2RAF RCL00700
C          IY = NUMBER OF VARIABLES RCL00710
C          MNP= MAXIMUM NUMBER OF PERMITTED POINTS IN THE NON-UNIFORM RCL00720
C          F.D. MESH ( MNP >= 32) RCL00730
C          IF(MNP.LT.32) THEN RCL00740
C              WRITE(6,1257) MNP
1257      FORMAT(' MNP = ',I3,' IS NOT PERMISSIBLE') RCL00750
C              STOP RCL00760
C              ENDIF RCL00770
C              NUMBER OF BOUNDARY CONDITIONS AT S=0 RCL00780
1276      NUMBEG=4 RCL00790
C              NUMBER OF MIXED BOUNDARY CONDITIONS RCL00800
C              NUMMIX=0 RCL00810
C              PARAMETER DECLARING THAT INITIAL APPROXIMATION IS USER SPECIFIED RCL00820
C              INIT=1 RCL00830
C              PARAMETER DECLARING THAT JACOBIANS ARE USER SPECIFIED RCL00840
C              IJAC=1 RCL00850
C              PARAMETER CONTROLLING MONITORING OF CALCULATIONS RCL00860
C              IFAIL=111 RCL00870
C              RCL00880
C              RCL00890
C              RCL00900
C              READAS : READS APPROXIMATE EMBEDDED SOLUTION FROM DEVICE 10 RCL00910
C              IT ALSO PROVIDES ICCC,NP, TOL= TOLERANCE OF ITERATIONS RCL00920
C              RCL00930
C              CALL READAS(ICCC,NP,TOL) RCL00940
C              RCL00950
C              IF THE NUMBER OF POINTS FROM THIS SOLUTION IS GREATER THAN THE RCL00960
C              STATIC SOLUTION POINTS, INTERPOLATE ALL CHARACTERISTICS TO THE RCL00970
C              NEW NUMBER OF POINTS. RCL00980
C              ASSUMPTION : NP ALWAYS GREATER OR EQUAL TO NPI RCL00990
C              THIS IS VALID IF THE SAME STATIC SOLUTION WAS USED TO OBTAIN THE RCL01000
C              APPROXIMATE EMBEDDED SOLUTION IN DEVICE 10. RCL01010
C              RCL01020
C              RCL01030
C              IF (NP.GE.NPI) THEN RCL01040
C                  CALL INTERP(NP)
C                  NPI=NP RCL01050
C              ELSE IF (NP.LT.NPI) THEN RCL01060
C                  WRITE(6,314)
314          FORMAT(' APPROXIMATE SOLUTION IN DEVICE 10 WAS NOT OBTAINED'/
*                   ' FROM THE SAME STATIC SOLUTION, PROGRAM STOPS') RCL01080
C                  STOP RCL01090
C              END IF RCL01100
C              RCL01110
C              RCL01120

```

FILE: RCLINDY3 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

C      INCREMENT OF CONTINUATION PARAMETER
7891  WRITE(6,7890)
7890  FORMAT(' INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS' RCL01130
          *' IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.D0'/
          *' IF CONTINUATION IS REQUIRED THEN 0.D0 < DELEPS < 1.D0'/
          *' RECOMMENDATION :'/ RCL01140
          *' USUALLY DELEPS = 0.1D0 WILL SUFFICE'/
          *' FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM'/
          *' A SMALLER VALUE OF DELEPS, E.G. 0.05D0 OR 0.025D0 MIGHT BE'/
          *' NECESSARY')
          READ(5,*) DELEPS
          IF((DELEPS.GT.1.D0).OR.(DELEPS.LE.0.D0)) THEN
              WRITE(6,7892)
7892  FORMAT(' 0. < DELEPS <= 1.')
              GOTO 7891
          END IF
C      SET COUNTING VARIABLE
          ICOUNT = 1
          IF(ICCC.EQ.0) STOP
C      CALL X04AAF(1,NOUT)
          CALL X04ABF(1,NOUT)
C      CALL NAG SUBROUTINE D02RAF TO SOLVE THE TWO-POINT BOUNDARY VALUE RCL01270
C      PROBLEM USING A DEFERRED CORRECTION TECHNIQUE AND NEWTON'S RCL01280
C      ITERATION. RCL01290
          CALL D02RAF(N,MNP,NP,NUMBEG,NUMMIX,TOL,INIT,X,Y,IY,ABT,FCN,G,IJAC,RCL01300
          *JACOBF,JACOBG,DELEPS,JACEPS,JACGEP,WORK,LWORK,IWORK,LIWORK,IFAIL) RCL01310
C      WRITE(6,9000) IFAIL
9000  FORMAT(' IFAIL =',I3) RCL01320
C      OUTPUT THE RESULTS RCL01330
C      IF((IFAIL.EQ.0).OR.(IFAIL.EQ.4)) THEN RCL01340
          CALL OUTPUT(NP) RCL01350
          ENDIF RCL01360
C      STOP RCL01370
          END RCL01380
C      SUBROUTINE CHARAC(TLEN,IC,NP)
          THIS SUBROUTINE READS THE COMPLIANT RISER CHARACTERISTICS AND RCL01390
          EVALUATES THE NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE RCL01400
          GOVERNING EQUATIONS. RCL01410
          IMPLICIT REAL*8(A-H,O-Z) RCL01420
          PARAMETER(MNP=151,NA=4,N=7) RCL01430
          CHARACTER*80 NAME RCL01440
          COMMON/CONST/XPI,XPI2,RHOW,GRAV RCL01450
          COMMON/INPUT0/NAME RCL01460
          COMMON/INPUT1/TLEN,WA,WT,NSEG RCL01470
          COMMON/INPUT2/RHO0,AI,CFLUID,PRESS,AO(MNP) RCL01480
          COMMON/INPUTL/RLENG(MNP),SEG(MNP+1) RCL01490
          COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP) RCL01500
          *,EIXIS(MNP),GIP(MNP),GIPS(MNP) RCL01510
          COMMON/INPUT4/DXI(MNP),FXIETA(MNP),DETA(MNP),DXIETA(MNP) RCL01520

```

```

COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP) RCL01690
*MNP),TMAXI(MNP),TMAETA(MNP) RCL01700
COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP) RCL01710
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL01720
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCLO1730
*NPI),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV RCL01740
COMMON/STAT2/CONST1(MNP),CONST2(MNP) RCL01750
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL01760
DIMENSION TENO(MNP),QXO(MNP),TOS(MNP),QXIOS(MNP) RCL01770

C
C      READ RISER CHARACTERISTICS FROM DEVICE 8
READ (8,1000) NAME RCL01780
1000 FORMAT(8OA) RCL01790
      READ (8,1008) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS RCL01800
1008 FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6)) RCL01810
      TLENG = TLEN RCL01820
      DO 1502 I=1,NSEG RCL01830
      READ (8,1003) RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXI(IRCL01860
      *) ,PXEITA(I),EA(I),EIETA(I),AO(I),EIETAS(I) RCL01840
1502 CONTINUE RCL01850
1003 FORMAT(5(1X,D12.6)/6(1X,D12.6)) RCL01870
RCL01880
C
      DO 1332 I=1,NSEG RCL01890
      READ(8,1333) AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),AMARCL01920
      *ZI(I),XJZI(I),AJZI(I) RCL01930
1332 CONTINUE RCL01940
1333 FORMAT(6(1X,D12.6)/3(1X,D12.6)) RCL01950
C
      IC=1 RCL01960
C
7650  WRITE(6,7651) RCL01970
7651  FORMAT(' DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS'/
      *' IF YES INPUT 1 , IF NO INPUT 0') RCL01980
      READ(5,*) IPRINT RCL01990
      IF((IPRINT.NE.1).AND.(IPRINT.NE.0)) GOTO 7650 RCL02000
      IF(IPRINT.EQ.1) THEN RCL02010
C
      WRITE(6,1000) NAME RCL02020
      WRITE(6,2500) RCL02030
2500  FORMAT(' NSEG      TLEN          WA          RHO0          AI          AI RCL02040
      * CFLUID          PRESS') RCL02050
      WRITE(6,2001) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS RCL02060
2001  FORMAT(1X,I3,6(1X,D12.6)) RCL02070
      WRITE(6,3400) RCL02080
3400  FORMAT('   I     RLENG        RMASS        RMASST        AMAXI RCL02090
      * WEIGHT'    /'     DXI        PXEITA        EA        EIETA RCL02100
      *           AO        EIETAS') RCL02110
      DO 3002 I=1,NSEG RCL02120
      WRITE(6,3003) I,RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXIRCL02170
      *(I),PXEITA(I),EA(I),EIETA(I),AO(I),EIETAS(I) RCL02130
3002 CONTINUE RCL02140
3003  FORMAT(1X,I3,5(1X,D12.6)/4X,6(1X,D12.6)) RCL02150
C
      WRITE(6,1334) RCL02160
1334  FORMAT('   I     AMAETA'        DETA'        EIXI'        EIXIS' RCL02220
      *           GIP'        GIPS'/'        AMAZI'        JZI        AJZI' ) RCL02230
RCL02240

```

```

DO 1335 I=1,NSEG
WRITE(6,1336) I,AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),RCL02250
*AMAZI(I),XJZI(I),AJZI(I)
1335 CONTINUE
RCL02260
RCL02270
1336 FORMAT(1X,I3,6(1X,D12.6)/4X,3(1X,D12.6))
RCL02280
C
RCL02290
1701 WRITE(6,1700)
RCL02300
1700 FORMAT(' INPUT 1 IF RISER DATA ARE CORRECT'/' INPUT 0 TO STOP')
RCL02310
READ(5,*) IC
RCL02320
IF((IC.NE.0).AND.(IC.NE.1)) GOTO 1701
RCL02330
IF(IC.EQ.0) RETURN
RCL02340
C
RCL02350
ENDIF
RCL02360
C
NON - DIMENSIONALIZATIONS
RCL02370
C
RCL02380
GRAV=9.81D0
RCL02390
WAM=WA/GRAV
RCL02400
WT=WA*TLEN
RCL02410
XPI=4.D0*DATAN(1.D0)
RCL02420
XPI2=XPI/2.D0
RCL02430
RHOW=1.025D3
RCL02440
C
NONDIMENSIONALIZE CORRECTLY QXIO, TO FROM 2-D STATIC SOLUTION
RCL02450
TOMAX = TOMAX*WT
RCL02460
TND = WT/TOMAX
RCL02470
DO 229 I=1,NP
RCL02480
    STATIC(1,I) = STATIC(1,I)*TND
RCL02490
    STATIC(2,I) = STATIC(2,I)*TND
RCL02500
229 CONTINUE
RCL02510
C
NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS
RCL02520
C
TOMAXL = TOMAX/TLEN
RCL02530
TOML1 = TOMAX*TLEN
RCL02540
TOML2 = TOML1*TLEN
RCL02550
TLEN2 = TLEN**2
RCL02560
TMAXAV = 0.D0
RCL02570
DO 2000 I=1,NSEG
RCL02580
    RLENG(I)=RLENG(I)/TLEN
RCL02590
    WEIGHT(I)=WEIGHT(I)/TOMAXL
RCL02600
    EPSXI(I) = EIXI(I)/TOML2
RCL02610
    EPSXIS(I) = EIXIS(I)/TOML1
RCL02620
    EPSETA(I) = EIETA(I)/TOML2
RCL02630
    EPSETS(I) = EIETAS(I)/TOML1
RCL02640
    EPSPI(I) = GIP(I)/TOML2
RCL02650
    EPSPIS(I) = GIPS(I)/TOML1
RCL02660
    TUZI(I) = XJZI(I) + AJZI(I)
RCL02670
    TMAETA(I) = RMASS(I) + AMAETA(I)
RCL02680
    TMAXI(I) = RMASS(I) + AMAXI(I)
RCL02690
    TMAXAV = TMAXI(I)*RLENG(I) + TMAXAV
RCL02700
    DXIM = DXIM + DXI(I)*RLENG(I)
RCL02710
    DXIETA(I) = DXI(I)-DETA(I)
RCL02720
2000 CONTINUE
RCL02730
    NETAM = 0.D0
RCL02740

```

```

DO 4321 I=1,NSEG
    HETA(I) = TMAETA(I)/TMAXAV
    HETAM = HETAM + HETA(I)*RLENG(I)
    TLAMB2(I) = TLEN2*TMAXAV/TJZI(I)
4321 CONTINUE
C
C      CALCULATE DERIVATIVES OF STATIC QUANTITIES
DO 737 I=1,NP
    TENO(I) = STATIC(1,I)
    QXO(I) = STATIC(2,I)
737 CONTINUE
CALL DER1(TENO,XI,TOS,NP)
CALL DER1(QXO,XI,QXIOS,NP)
C      EVALUATE FUNCTIONS OF STATIC RESULTS
DO 56 I=1,NP
    CONST1(I) = TOS(I) - STATIC(2,I)*STATIC(3,I)
    CONST2(I) = QXIOS(I) + STATIC(1,I)*STATIC(3,I)
56 CONTINUE
C
C      SEG(I)=LEFT ORDINATE OF SEGMENT I
SEG(1)=0.D0
SEG(NSEG+1)=1.D0
DO 4000 I=2,NSEG
    SEG(I)=RLENG(I-1)+SEG(I-1)
4000 CONTINUE
C      INTERPOLATE STRUCTURAL DIMENSIONS TO NP POINTS
C      ASSUMPTION: NUMBER OF RISER SEGMENTS WITH DIFFERENT
C      CHARACTERISTICS IS SMALLER THAN THE STATIC SOLUTION POINTS
C      NSEG < NPI
C
IF (NSEG.GE.NPI) THEN
    WRITE(6,188)
188 FORMAT(' NSEG => NPI, PROGRAM STOPS')
    IC = 0
    RETURN
END IF
CALL STRUCT(EPSXI,X,NP)
CALL STRUCT(EPSXIS,X,NP)
CALL STRUCT(EPSETA,X,NP)
CALL STRUCT(EPSETS,X,NP)
CALL STRUCT(EPSPI,X,NP)
CALL STRUCT(EPSPI,S,X,NP)
CALL STRUCT(HETA,X,NP)
CALL STRUCT(TLAMB2,X,NP)
RETURN
END
C
SUBROUTINE READ2D(NP)
C      THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM
C      DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION.
IMPLICIT REAL*8(A-H,O-Z)
PARAMETER(N=7,MNP=151,NA=4)
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
C
STATIC(1,I) = STATIC EFFECTIVE TENSION TO

```

RCL02810
RCL02820
RCL02830
RCL02840
RCL02850
RCL02860
RCL02870
RCL02880
RCL02890
RCL02900
RCL02910
RCL02920
RCL02930
RCL02940
RCL02950
RCL02960
RCL02970
RCL02980
RCL02990
RCL03000
RCL03010
RCL03020
RCL03030
RCL03040
RCL03050
RCL03060
RCL03070
RCL03080
RCL03090
RCL03100
RCL03110
RCL03120
RCL03130
RCL03140
RCL03150
RCL03160
RCL03170
RCL03180
RCL03190
RCL03200
RCL03210
RCL03220
RCL03230
RCL03240
RCL03250
RCL03260
RCL03270
RCL03280
RCL03290
RCL03300
RCL03310
RCL03320
RCL03330
RCL03340
RCL03350
RCL03360

```

C      STATIC(2,I) = STATIC SHEAR FORCE IN THE XI DIRECTION          RCL03370
C      STATIC(3,I) = STATIC OMEGA IN THE ETA DIRECTION                RCL03380
C      STATIC(4,I) = STATIC ANGLE PHI                                RCL03390
C
C      WRITE(6,2000) MNP                                         RCL03400
2000 FORMAT(' MNP=',I3)                                         RCL03410
C      READ STATIC SOLUTION                                         RCL03420
      READ(10,36459) NP,VM                                         RCL03430
36459 FORMAT(1X,I3,1X,D12.6)                                     RCL03440
      WRITE(6,2311) NP,VM                                         RCL03450
2311 FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
*' NP =',I3/
*' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S ,VM =',D12.6) RCL03460
C
C      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN                         RCL03500
      ICCC=0                                         RCL03510
      WRITE(6,12439)                                         RCL03520
12439 FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS')               RCL03530
      RETURN                                         RCL03540
      ENDIF                                         RCL03550
C
C      READING FROM DEVICE 10                                     RCL03560
C
DO 1021 I=1,NP                                         RCL03590
READ(10,1033) X(I),STATIC(1,I),STATIC(2,I),STATIC(3,I),STATIC(4,I) RCL03600
*,XCOOR,YCOOR,STRARC,TENSI,VLOCKI(I)
XI(I) = X(I)                                         RCL03620
1021 CONTINUE                                         RCL03630
1033 FORMAT(10(1X,D12.6))                           RCL03640
C      EVALUATE MAXIMUM STATIC EFFECTIVE TENSION             RCL03650
TOMAX=DMAX1(STATIC(1,1),STATIC(1,2))                 RCL03660
DO 9859 I=3,NP                                         RCL03670
TOMAX=DMAX1(TOMAX,STATIC(1,I))                      RCL03680
9859 CONTINUE                                         RCL03690
C
      WRITE(6,1654) TOMAX                                         RCL03710
1654 FORMAT(' 2-D STATIC SOLUTION SUCCESFULLY READ'/' MAXIMUM STATIC EF RCL03720
*FECTIVE TENSION/WA*L = ',D10.4)                     RCL03730
C
      RETURN                                         RCL03740
      END                                         RCL03750
C
SUBROUTINE READAS(ICCC,NP,TOL)                         RCL03760
IMPLICIT REAL*8(A-H,O-Z)                               RCL03770
PARAMETER(N=7,MNP=151,NA=4)                            RCL03780
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE            RCL03790
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKI(MNP),TOMAX,VM,NPI RCL03800
COMMON/BOUNDA/BOUND,BOUNDR,BOUNDB                    RCL03810
C
READAS READS INITIAL EMBEDDED APPROXIMATION RESULTING FROM A RCL03820
RUN OF RCFORC1                                         RCL03830
C
Y(1,I) = SHEAR FORCE IN THE ETA DIRECTION           RCL03840
Y(2,I) = OMEGA ABOUT ZETA                           RCL03850
Y(3,I) = OMEGA ABOUT XI                            RCL03860
Y(4,I) = DYNAMIC ANGLE THETA                        RCL03870
C
RCL03880
RCL03890
RCL03900
RCL03910
RCL03920

```

```

C      Y(5,I) = DYNAMIC ANGLE BETA          RCL03930
C      Y(6,I) = OUT OF PLANE DISPLACEMENT, R  RCL03940
C      Y(7,I) = NATURAL FREQUENCY            RCL03950
C      X(I)   = UNSTRETCHED ARC LENGTH S    RCL03960
C
C      WRITE(6,2000) MNP                    RCL03970
2000  FORMAT(' MNP=',I3)                  RCL03980
C
C      READ(12,36459) MODE,NP,SIGMAD        RCL04000
36459 FORMAT(1X,I2,1X,I3,1X,D10.4)
      WRITE(6,2311) NP,MODE,SIGMAD         RCL04010
2311  FORMAT(' INITIAL APPROXIMATION FROM DEVICE 12'/
*' NP =',I3,' MODE NUMBER =',I2,' SIGMAD =',D10.4) RCL04020
C
      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN    RCL04030
        ICCC=0                            RCL04040
        WRITE(6,12439)                      RCL04050
12439  FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS') RCL04060
        RETURN                           RCL04070
      ENDIF                             RCL04080
C
C      READING DATA FROM DEVICE 12 ...
C
      DO 10011 I=1,NP                    RCL04100
      READ(12,10012) X(I),(Y(J,I),J=1,7)  RCL04110
10011 CONTINUE                         RCL04120
10012 FORMAT(8(1X,D12.6))              RCL04130
C
C      EMBEDDED BOUNDARY CONDITIONS AND OMEGA XI(0)
      BOUND = Y(3,1)                      RCL04140
      BOUNDB = Y(5,NP)                   RCL04150
      BOUNDR = Y(6,NP)                   RCL04160
C
      WRITE(9,1052) NP,SIGMAD             RCL04170
1052  FORMAT(' INITIAL CONDITION FOR EPS=0. AND NP = ',I3,' POINTS,  NATRCL04260
      *URAL FREQUENCY = ',D10.4,' RAD/S'/
      *' I      ARC      SHEAR ETA      OMEGA ZETA      OMEGA XI      THETARCL04280
      *     BETA      R           SIGMA')      RCL04270
      DO 1501 I=1,NP                    RCL04290
      WRITE(9,1503) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
      * Y(7,I)                          RCL04300
1501  CONTINUE                         RCL04310
1503  FORMAT(1X,I3,8(1X,D12.6))       RCL04320
C
      WRITE(6,9561)
9561  FORMAT(' DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION'/
      *' IF YES INPUT 1')
      READ(5,*) IPRINT
      IF(IPRINT.EQ.1) THEN               RCL04370
        WRITE(6,1052) NP                RCL04380
        DO 9659 I=1,NP                 RCL04390
        WRITE(6,1603) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
      * Y(7,I)                          RCL04400
9659  CONTINUE                         RCL04410
      END IF                           RCL04420
      WRITE(6,1722) BOUND               RCL04430
1722  FORMAT(' ASSUMED BOUNDARY CONDITION'/
      *' ')                            RCL04440

```

```

      ** OMEGA XI (0) =',D12.6)
C
      TOL1 = 0.0
      DO 3931 I=1,NP
      TOL1 = DMAX1(TOL1,DABS(Y(3,I)))
3931 CONTINUE
      WRITE(6,3932) TOL1
3932 FORMAT(' MAXIMUM ABSOLUTE VALUE OF N-D OMEGA XI IS =',D12.6/
      ** THIS NUMBER CAN BE USED TO ESTIMATE /
      ** A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS'/
      ** INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE'/
      ** E.G. INPUT 0.01 OR 0.1 IF THE INITIAL ESTIMATE FOR OMEGA XI'/
      ** IS SMALL'/' AFTER THE RUN CHECK THE ACCURACY OF THE SOLUTION')
      READ(5,*) TOLV
      TOL = DABS(TOLV)*TOL1
C
1005 WRITE(6,1004)
1004 FORMAT(' IF YOU WISH TO STOP INPUT 0'/
      ** IF YOU WISH TO CONTINUE WITH ITERATIONS INPUT 1')
      READ(5,*) ICCC
      IF((ICCC.NE.0).AND.(ICCC.NE.1)) GOTO 1005
C
      RETURN
      END
C
      SUBROUTINE STRUCT(ARRAY,X,NP)
C
      THIS SUBROUTINE CHANGES A SERIES OF DATA GIVEN IN
C
      SEGMENTS TO A SERIES OF DATA CORRESPONDING TO SPECIFIC POINTS
C
      ASSUMPTION: NSEG < NP
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(MNP=151)
      COMMON/INPUT1/TLEN,WA,WT,NSEG
      COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)
      DIMENSION ARRAY(MNP),HELP(MNP),X(MNP)
C
      IF(NSEG.EQ.1) THEN
        DO 83 I=1,NP
        HELP(I) = ARRAY(1)
83      CONTINUE
      ELSE
        HELP(1) = ARRAY(1)
        HELP(NP) = ARRAY(NSEG)
        I=2
        DO 84 K=2,NP-1
          IF ((X(K).GT.SEG(I-1)).AND.(X(K).LT.SEG(I))) THEN
            HELP(K) = ARRAY(I-1)
          ELSE IF (X(K).EQ.SEG(I)) THEN
            HELP(K) = 0.5DC*(ARRAY(I-1) + ARRAY(I))
          ELSE IF (X(K).GT.SEG(I)) THEN
            HELP(K) = ARRAY(I)
            I = I + 1
          END IF
84      CONTINUE
        END IF
        DO 85 K=1,NP
          ARRAY(K) = HELP(K)
85
      
```

RCL04490
RCL04500
RCL04510
RCL04520
RCL04530
RCL04540
RCL04550
RCL04560
RCL04570
RCL04580
RCL04590
RCL04600
RCL04610
RCL04620
RCL04630
RCL04640
RCL04650
RCL04660
RCL04670
RCL04680
RCL04690
RCL04700
RCL04710
RCL04720
RCL04730
RCL04740
RCL04750
RCL04760
RCL04770
RCL04780
RCL04790
RCL04800
RCL04810
RCL04820
RCL04830
RCL04840
RCL04850
RCL04860
RCL04870
RCL04880
RCL04890
RCL04900
RCL04910
RCL04920
RCL04930
RCL04940
RCL04950
RCL04960
RCL04970
RCL04980
RCL04990
RCL05000
RCL05010
RCL05020
RCL05030
RCL05040

```

85 CONTINUE                               RCL05050
    RETURN                                RCL05060
    END                                   RCL05070
RCL05080
C
SUBROUTINE DER1(ARRAY,X,DERIV,NP)          RCL05090
C
THIS SUBROUTINE EVALUATES THE FIRST DERIVATIVE OF A SERIES OF DATA RCL05100
C
POINTS USING A SECOND ORDER APPROXIMATION (NONUNIFORM GRID FINITE RCL05110
C
DIFFERENCES).                           RCL05120
IMPLICIT REAL*8(A-H,O-Z)                 RCL05130
PARAMETER (MNP=151)                      RCL05140
DIMENSION ARRAY(MNP),DERIV(MNP),X(MNP)   RCL05150
C
USE FIRST ORDER FOR END POINTS DERIVATIVES RCL05160
DERIV(1) = (-ARRAY(1) + ARRAY(2))/(X(2)-X(1)) RCL05170
DERIV(NP) = (-ARRAY(NP-1) + ARRAY(NP))/(X(NP)-X(NP-1)) RCL05180
C
DO 836 I=2,NP-1                         RCL05190
    DX = X(I) - X(I-1)                   RCL05200
    DX1 = X(I+1) - X(I)                  RCL05210
    D = DX/DX1                          RCL05220
    DERIV(I) = (D*(ARRAY(I+1)-ARRAY(I)) - (ARRAY(I-1)-ARRAY(I))/D) RCL05230
    DERIV(I) = DERIV(I)/(DX+DX1)        RCL05240
836 CONTINUE                               RCL05250
    RETURN                                RCL05260
    END                                   RCL05270
RCL05280
C
SUBROUTINE OUTPUT(NP)                     RCL05290
C
THIS SUBROUTINE OUTPUTS THE RESULTS IN TWO FORMATS; A COMPLETE RCL05300
C
FORMAL FORM AND A FORM TO BE USED FOR PLOTTING. RCL05310
IMPLICIT REAL*8(A-H,O-Z)                 RCL05320
PARAMETER(MNP=151,NA=4,N=7)               RCL05330
CHARACTER*80 NAME                         RCL05340
COMMON/CONST/XPI,XPI2,RHOCW,GRAV        RCL05350
COMMON/INPUTO/NAME                        RCL05360
COMMON/INPUT1/TLEN,WA,WT,NSEG             RCL05370
COMMON/INPUT2/RHOA,AI,CFLUID,PRESS,AO(MNP) RCL05380
COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)       RCL05390
COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP) RCL05410
*,EIXIS(MNP),GIP(MNP),GIPS(MNP)          RCL05420
COMMON/INPUT4/DXI(MNP),PXIETA(MNP),DETA(MNP),DXIETA(MNP) RCL05430
COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP) RCL05440
*TMAXI(MNP),TMAETA(MNP)                  RCL05450
COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP) RCL05460
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKI(MNP),TOMAX,VM,NPI RCL05470
COMMON/STAT1/XTOP,YTOP                  RCL05480
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MNP) RCL05490
*,EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV RCL05500
COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,TO,QXIO,OMEGAO,RCLO5510
*CONS1,CONS2                            RCL05520
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL05530
DIMENSION A1(MNP)                        RCL05540
C
CALCULATE THE ORTHONORMALIZING CONSTANT FOR R AND BETA RCL05550
C
DO 542 I=1,NP                           RCL05560
    CALL COUNT(X(I))                    RCL05570
    A1(I) = Y(5,I)**2/TLAM2 + HH*Y(6,I)**2 RCL05580
542 CONTINUE                               RCL05590
                                            RCL05600

```

```

A = 0.0D0                                     RCL05610
DO 642 I=2,NP                                RCL05620
  DX2 = (X(I)-X(I-1))*0.5D0                  RCL05630
  A = A + (A1(I)+A1(I-1))*DX2                RCL05640
642 CONTINUE                                    RCL05650
  A = DSQRT(A)                                 RCL05660
C   ORTHONORMALIZE THE SOLUTION EXCEPT SIGMA    RCL05670
DO 643 I=1,NP                                RCL05680
  Y(N,I) = DABS(Y(N,I))                      RCL05690
  DO 643 K=1,N-1                            RCL05700
    Y(K,I) = Y(K,I)/A                         RCL05710
643 CONTINUE                                    RCL05720
C   OUTPUT RESULTS                             RCL05730
  WRITE(9,1000) NAME                          RCL05740
1000 FORMAT(80A)                                RCL05750
  WRITE(9,1001) NSEG,TLEN,WA,RHO0,RHOW,AI,CFLUID,PRESS,WT,TOMAX,VM RCL05760
C
1001 FORMAT(1X,I3,9X,' = NUMBER OF RISER SEGMENTS'/
*1X,D12.6,' = UNSTRETCHED RISER LENGTH IN M'/
*1X,D12.6,' = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH RCL05770
*IN N/M'/
*1X,D12.6,' = INNER FLUID DENSITY IN KG/M3'/ RCL05780
*1X,D12.6,' = SALT WATER DENSITY IN KG/M3'/ RCL05790
*1X,D12.6,' = INNER CROSS SECTIONAL AREA IN M2'/ RCL05800
*1X,D12.6,' = INNER FLUID SPEED IN M/S'/ RCL05810
*1X,D12.6,' = INNER FLUID OVERPRESSURE IN N/M2'/ RCL05820
*1X,D12.6,' = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N'/ RCL05830
*1X,D12.6,' = MAXIMUM STATIC TENSION IN N'/ RCL05840
*1X,D12.6,' = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S') RCL05850
C   EVALUATE DIMENSIONAL NATURAL FREQUENCY      RCL05860
  SIGMAD = Y(N,1)*DSQRT(TOMAX/TMAXAV)/TLEN      RCL05870
  WRITE(9,1002) NSEG                           RCL05880
1002 FORMAT(//' DATA PER RISER SEGMENT FOR NSEG =RCL05930
* ,I3,' SEGMENTS'/' DIMENSIONAL QUANTITY IRCL05940
* E S I N T H E S . I . S Y S T E M '/
*   RLENG      DXI      PXIETA      AO      WEIGHT      MASS      RCL05950
*   TMASS      AMAXI     AMAETA     AMAZI     TMAXI     TMAETA RCL05960
*) )                                         RCL05970
  TL = TOMAX/TLEN                           RCL05980
  DO 1004 I=1,NSEG                           RCL05990
  WRITE(9,1003) RLENG(I)*TLEN,DXI(I),PXIETA(I),AO(I),WEIGHT(I)*TL,RMRCL06010
*ASS(I),RMASST(I),AMAXI(I),AMAETA(I),AMAZI(I),TMAXI(I),TMAETA(I) RCL06020
1004 CONTINUE                                    RCL06030
1003 FORMAT(12(1X,D10.4)/)                    RCL06040
C
  WRITE(9,10022)                               RCL06050
10022 FORMAT('      EA      EIETA      EIETAS      EIXI      EIXIS      RCL06060
*   GIP      GIPS      DETA      JZI      AJZI      TJZI') RCL06070
  DO 10023 I=1,NSEG                           RCL06080
  WRITE(9,10024) EA(I),EIETA(I),EIETAS(I),EIXI(I),EIXIS(I),GIP(I), RCL06090
*GIPS(I),DETA(I),XJZI(I),AJZI(I),TJZI(I) RCL06100
10023 CONTINUE                                    RCL06110
10024 FORMAT(11(1X,D10.4)/)                    RCL06120
C
  WRITE(9,761) MODE,SIGMAD                   RCL06130
761 FORMAT(' *****'*)' RCL06140
                                         RCL06150
                                         ' RCL06160

```

FILE: RCLINDY3 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

      /*' MODE NUMBER = ',I2/' NATURAL FREQUENCY RCL06170
      *C Y = ',D10.4,' RAD/S/' **** RCL06180
      *****')
C
      WRITE(9,1009) NP
RCL06190
1009 FORMAT(/' ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = ',
      *I3,' POINTS')
RCL06200
      '' S      QETA      OMEGAZETA   OMEGAXI      THETA
      * R      SIGMA')
RCL06210
      DO 1010 I=1,NP
RCL06220
      WRITE(9,1011) X(I),(Y(J,I),J=1,7)
RCL06230
1010 CONTINUE
RCL06240
1011 FORMAT(8(1X,D10.4))
RCL06250
C NORMALIZE ESTIMATED ERROR BY COMPONENTS.
RCL06260
      DO 189 I=1,N-1
RCL06270
      ABT(I)=ABT(I)/A
RCL06280
189 CONTINUE
RCL06290
      WRITE(9,10119) (ABT(I),I=1,7)
RCL06300
10119 FORMAT(' MAXIMUM ESTIMATED ERROR BY COMPONENTS'/11X,7(1X,D10.4))
RCL06310
C
C
C OUTPUT TO FILE CONNECTED TO DEVICE 11
RCL06320
C THIS CAN BE USED FOR PLOTS OR INPUT TO ANOTHER RUN OF RCLINDY3
RCL06330
C
C
CALL STRUC(VLOCKI,NP,NPI)
RCL06340
WRITE(11,36459) MODE,NP,SIGMAD
RCL06350
36459 FORMAT(1X,I2,1X,I3,1X,D10.4)
RCL06360
      DO 3666 I=1,NP
RCL06370
      WRITE(11,3667) X(I),(Y(J,I),J=1,7),VLOCKI(I)
RCL06380
3666 CONTINUE
RCL06390
3667 FORMAT(9(1X,D12.6))
RCL06400
C
      RETURN
RCL06410
      END
RCL06420
C
SUBROUTINE FCN(X,EPS,Y,F,N)
RCL06430
C THIS SUBROUTINE EVALUATES THE FUNCTIONS USED BY DO2RAF TO SOLVE
RCL06440
C THE PROBLEM.
RCL06450
      IMPLICIT REAL*8(A-H,O-Z)
RCL06460
      PARAMETER(MNP=151)
RCL06470
      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MNP),
      *NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV
RCL06480
      COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,TO,QXIO,OMEGAO,RCL06490
      *CONS1,CONS2
RCL06500
      DIMENSION Y(N),F(N)
RCL06510
C LOCATE MESH POINT TO EVALUATE COEFFICIENTS
RCL06520
      CALL COUNT(X)
RCL06530
C
      F(1) = TO*Y(3) - QXIO*Y(2) + CONS1*Y(4) - CONS2*Y(5)
RCL06540
      F(1) = F(1) - HH*Y(6)*Y(7)**2
RCL06550
C
      F(2) = -EPIS*Y(2) + (EXI-EETA)*OMEGAO*Y(3)
RCL06560
      F(2) = -(F(2) - Y(5)*Y(7)**2/TLAM2)/EPI
RCL06570
C
      F(3) = (-EXIS*Y(3) + Y(1) + (EETA-EPI)*Y(2)*OMEGAO)/EXI
RCL06580
      F(3) = -(F(3) - Y(4)*Y(7)**2/TLAM2)/EPI
RCL06590
      F(4) = -(F(4) - Y(5)*Y(7)**2/TLAM2)/EPI
RCL06600
      F(5) = -(F(5) - Y(6)*Y(7)**2/TLAM2)/EPI
RCL06610
      F(6) = -(F(6) - Y(7)**2/TLAM2)/EPI
RCL06620
      F(7) = -(F(7) - Y(8)**2/TLAM2)/EPI
RCL06630
      F(8) = -(F(8) - Y(9)**2/TLAM2)/EPI
RCL06640
      F(9) = -(F(9) - Y(10)**2/TLAM2)/EPI
RCL06650
      F(10) = -(F(10) - Y(11)**2/TLAM2)/EPI
RCL06660
      F(11) = -(F(11) - Y(12)**2/TLAM2)/EPI
RCL06670
      F(12) = -(F(12) - Y(13)**2/TLAM2)/EPI
RCL06680
      F(13) = -(F(13) - Y(14)**2/TLAM2)/EPI
RCL06690
      F(14) = -(F(14) - Y(15)**2/TLAM2)/EPI
RCL06700
      F(15) = -(F(15) - Y(16)**2/TLAM2)/EPI
RCL06710
      F(16) = -(F(16) - Y(17)**2/TLAM2)/EPI
RCL06720

```

```

C
C      F(4) = Y(3) - OMEGA0*Y(5)                                RCL06730
C
C      F(5) = OMEGA0*Y(4) + Y(2)                                RCL06740
C
C      F(6) = -Y(4)                                              RCL06750
C
C      F(7) = 0.D0                                                RCL06760
C      RETURN
C      END
C
C      SUBROUTINE G(EPS,YA,YB,BC,N)
C      BOUNDARY CONDITIONS USED BY DO2RAF
C      IMPLICIT REAL*8(A-H,O-Z)
C      DIMENSION YA(N),YB(N),BC(N)
C      COMMON/BOUNDA/BOUND,BOUNDR,BOUNDB
C
C      BC(1)=YA(4)                                              RCL06830
C      BC(2)=YA(5)                                              RCL06840
C      BC(3)=YA(6)                                              RCL06850
C      BC(4)=YA(3) - BOUND                                     RCL06860
C
C      BC(5)=YB(4)                                              RCL06870
C      BC(6)=YB(5) - BOUNDB + EPS*BOUNDB
C      BC(7)=YB(6) - BOUNDR + EPS*BOUNDR
C
C      RETURN
C      END
C
C      SUBROUTINE JACOBF(X,EPS,Y,F,N)
C      THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE EQUATIONS TO USE IN RCL07000
C      NEWTON'S ITERATION.
C      IMPLICIT REAL*8(A-H,O-Z)
C      PARAMETER(MNP=151)
C      COMMON/CCEEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL07070
C      *NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV
C      COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,T0,QXIO,OMEGA0,RCL07090
C      *CONS1,CONS2
C      DIMENSION Y(N),F(N,N)
C      LOCATE MESH POINT TO EVALUATE COEFFICIENTS
C      CALL COUNT(X)
C
C      DO 817 I=1,N
C          DO 817 M=1,N
C              F(I,M) = 0.D0
C 817 CONTINUE
C
C      F(1,2) = -QXIO
C      F(1,3) = T0
C      F(1,4) = CONS1
C      F(1,5) = -CONS2
C      F(1,6) = -HH*Y(7)**2
C      F(1,7) = -2.D0*Y(7)*HH*Y(6)
C
C      F(2,2) = -EPIS/EPI
C      F(2,3) = (EXI-EETA)*OMEGA0/EPI

```

FILE: RCLINDY3 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

C
C      SUBROUTINE JACGEP(EPS,YA,YB,BCEP,N)          RCL07850
C      THIS SUBROUTINE EVALUATES THE DERIVATIVES OF THE BOUNDARY    RCL07860
C      CONDITIONS WITH RESPECT TO THE CONTINUATION PARAMETER EPS.   RCL07870
C      IMPLICIT REAL*8(A-H,O-Z)                         RCL07880
C      COMMON/BOUNDA/BOUND,BOUNDR,BOUNDB                RCL07890
C      DIMENSION YA(N),YB(N),BCEP(N)                  RCL07900
C      DO 871 K=1,N                                    RCL07910
C          BCEP(K) = 0.D0                               RCL07920
871  CONTINUE                                     RCL07930
C          BCEP(5) = BOUNDB                           RCL07940
C          BCEP(6) = BOUNDR                           RCL07950
C          RETURN                                     RCL07960
C          END                                       RCL07970
C
C      SUBROUTINE INTERP(NP)                         RCL07980
C      THIS SUBROUTINE INTERPOLATES THE RISER CHARACTERISTICS AND THE RCL07990
C      STATIC SOLUTION TO THE NEW NUMBER OF POINTS NP.           RCL08000
C      ASSUMPTION: NP .GE. NPI                          RCL08010
C      IMPLICIT REAL*8(A-H,O-Z)                         RCL08020
C      PARAMETER(MNP=151,NA=4,N=7)                      RCL08030
C      COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL08060
*NP),EPSPIS(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV
C      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI      RCL08070
C      COMMON/STAT2/CONST1(MNP),CONST2(MNP)            RCL08080
C      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE       RCL08090
C      DIMENSION HELP(MNP)                           RCL08100
C
C      INTERPOLATE STRUCTURAL DATA TO THE NEW NUMBER OF POINTS     RCL08110
CALL STRUC(EPSXI,NP,NPI)                         RCL08120
CALL STRUC(EPSXIS,NP,NPI)                        RCL08130
CALL STRUC(EPSETA,NP,NPI)                        RCL08140
CALL STRUC(EPSETS,NP,NPI)                        RCL08150
CALL STRUC(EPSPI,NP,NPI)                         RCL08160
CALL STRUC(EPSPIS,NP,NPI)                        RCL08170
CALL STRUC(HETA,NP,NPI)                          RCL08180
CALL STRUC(TLAMB2,NP,NPI)                        RCL08190
CALL STRUC(CONST1,NP,NPI)                        RCL08200
CALL STRUC(CONST2,NP,NPI)                        RCL08210
CALL STRUC(VLOCKXI,NP,NPI)                       RCL08220
C
C      INTERPOLATE STATIC SOLUTION TO THE NEW NUMBER OF POINTS     RCL08230
DO 459 K=1,3                                     RCL08240
DO 458 I=1,NPI                                  RCL08250
    HELP(I) = STATIC(K,I)                         RCL08260
458  CONTINUE                                    RCL08270
        CALL STRUC(HELP,NP,NPI)                     RCL08280
        DO 457 I=1,NP                            RCL08290
            STATIC(K,I) = HELP(I)                 RCL08300
457  CONTINUE                                    RCL08310
459  CONTINUE                                    RCL08320
        DO 339 I=1,NP                            RCL08330
            XI(I) = X(I)                         RCL08340
339  CONTINUE                                    RCL08350
        RETURN                                     RCL08360
        END                                       RCL08370
C
C      SUBROUTINE STRUC(ARRAY,NP,NPOLD)             RCL08380

```

```

C THIS SUBROUTINE INTERPOLATES A SERIES OF DIVISION POINTS TO A NEW RCL08410
C SERIES OF DIVISION POINTS. RCL08420
IMPLICIT REAL*8(A-H,O-Z) RCL08430
PARAMETER(MNP=151,NA=4,N=7) RCL08440
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL08450
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL08460
DIMENSION ARRAY(MNP),HELP(MNP) RCL08470
C RCL08480
HELP(1) = ARRAY(1) RCL08490
HELP(NP) = ARRAY(NPOLD) RCL08500
DO 82 I=2,NPOLD RCL08510
DO 81 K=2,NP-1 RCL08520
  IF ((X(K).GT.XI(I-1)).AND.(X(K).LT.XI(I))) THEN RCL08530
    CONV = (X(K)-XI(I-1))*(ARRAY(I)-ARRAY(I-1))/(XI(I)-XI(I-1)) RCL08540
    HELP(K) = ARRAY(I-1) + CONV RCL08550
  ELSE IF (X(K).EQ.XI(I)) THEN RCL08560
    HELP(K) = ARRAY(I) RCL08570
  END IF RCL08580
81  CONTINUE RCL08590
82  CONTINUE RCL08600
  DO 85 K=1,NP RCL08610
    ARRAY(K) = HELP(K) RCL08620
85  CONTINUE RCL08630
  RETURN RCL08640
END RCL08650
C RCL08660
SUBROUTINE COUNT(X) RCL08670
C THIS SUBROUTINE EVALUATES THE RISER CHARACTERISTICS AS WELL AS THE RCL08680
C STATIC SOLUTION AT A POINT X. RCL08690
C IT RETURNS THEIR VALUE IN COMMON BLOCK COEF1. RCL08700
IMPLICIT REAL*8(A-H,O-Z) RCL08710
PARAMETER(MNP=151,NA=4) RCL08720
COMMON/COEF/EPSXI(MNP),EPSXIS(MNP),EPSETA(MNP),EPSETS(MNP),EPSPI(MRCL08730
*N),EPSPI(MNP),HETA(MNP),TLAMB2(MNP),HETAM,TMAXAV RCL08740
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL08750
COMMON/COEF1/EXI,EXIS,EETA,EETAS,EPI,EPIS,HH,TLAM2,TO,QXIO,OMEGAO,RCL08760
*CONS1,CONS2 RCL08770
COMMON/STAT2/CONST1(MNP),CONST2(MNP) RCL08780
COMMON/COUN/ICOUNT RCL08790
M = ICOUNT RCL08800
IF (X.EQ.XI(M)) THEN RCL08810
  EXI = EPSXI(M)
  EXIS = EPSXIS(M)
  EETA = EPSETA(M)
  EETAS = EPSETS(M)
  EPI = EPSPI(M)
  EPIS = EPSPI(M)
  HH = HETA(M)
  TLAM2 = TLAMB2(M)
  TO = STATIC(1,M)
  QXIO = STATIC(2,M)
  OMEGA0 = STATIC(3,M)
  CONS1 = CONST1(M)
  CONS2 = CONST2(M)
  ICOUNT = ICOUNT + 1
  IF(X.EQ.1.D0) ICOUNT=-1

```

```

ELSE IF ((X.LT.XI(M)).AND.(X.GT.XI(M-1))) THEN RCL08970
  DIFF = DABS(XI(M)-X) RCL08980
C THIS IF STATEMENT IS TO ACCOUNT FOR A COMPUTER INACCURACY IN RCL08990
C REPRESENTING AND EQUATING REAL NUMBERS. RCL09000
  IF (DIFF.LT.1.D-7) THEN RCL09010
    ICOUNT = ICOUNT + 1 RCL09020
    IF (XI(M).EQ.1.DO) ICOUNT = 1 RCL09030
  END IF RCL09040
  DX = (X-XI(M-1))/(XI(M)-XI(M-1)) RCL09050
  EXI = EPSXI(M-1) + (EPSXI(M)-EPSXI(M-1))*DX RCL09060
  EXIS = EPSXIS(M-1) + (EPSXIS(M)-EPSXIS(M-1))*DX RCL09070
  EETA = EPSETA(M-1) + (EPSETA(M)-EPSETA(M-1))*DX RCL09080
  EETAS = EPSETS(M-1) + (EPSETS(M)-EPSETS(M-1))*DX RCL09090
  EPI = EPSPI(M-1) + (EPSPI(M)-EPSPI(M-1))*DX RCL09100
  EPIS = EPSPIS(M-1) + (EPSPIS(M)-EPSPIS(M-1))*DX RCL09110
  HH = HETA(M-1) + (HETA(M)-HETA(M-1))*DX RCL09120
  TLAM2 = TLAMB2(M-1) + (TLAMB2(M)-TLAMB2(M-1))*DX RCL09130
  TO = STATIC(1,M-1) + (STATIC(1,M)-STATIC(1,M-1))*DX RCL09140
  QKIO = STATIC(2,M-1) + (STATIC(2,M)-STATIC(2,M-1))*DX RCL09150
  OMEGA0 = STATIC(3,M-1) + (STATIC(3,M)-STATIC(3,M-1))*DX RCL09160
  CONST1 = CONST1(M-1) + (CONST1(M)-CONST1(M-1))*DX RCL09170
  CONST2 = CONST2(M-1) + (CONST2(M)-CONST2(M-1))*DX RCL09180
  ELSE RCL09190
C
    WRITE(6,*) ' ERROR OCCURED IN COUNTER, NP =',M RCL09200
  END IF RCL09210
  RETURN RCL09220
END RCL09230
RCL09240

```

256

Chapter XVII

Listing of Program RCFORCE FORTRAN A

C RCFORCE RCF00010
C THIS PROGRAM CALCULATES THE IN-PLANE LINEAR DYNAMIC RESPONSE OF A RCF00020
C COMPLIANT RISER WITH A 2-D STATIC CONFIGURATION FOR A GIVEN FREQUENCY RCF00030
C AND AMPLITUDE OF EXCITATION AT THE TOP (P OR Q EXCITATION IS ALLOWED). RCF00040
C A NONUNIFORM GRID FINITE DIFFERENCES AND DEFERRED CORRECTION TECHNIQUE RCF00050
C IS USED. THE OUTPUT OF THIS PROGRAM CAN BE USED AS INPUT TO RCLINDY2 TORCF00060
C SOLVE THE IN-PLANE DYNAMICS PROBLEM WITH HOMOGENEOUS BOUNDARY RCF00070
C CONDITIONS (EIGENPROBLEM). RCF00080
C THIS PROGRAM WILL BE PARTICULARLY USEFUL IN THE CASE OF ZERO CURRENT RCF00090
C VELOCITY WHERE THE ASYMPTOTIC THEORY IS NO LONGER VALID. RCF00100
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED. RCF00110
C***** RCF00120
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY RCF00130
C ALL RIGHTS RESERVED. RCF00140
C***** RCF00150
C PROGRAMMER GEORGE A. KRIEZIS JUNE 6, 1985 M.I.T. RCF00160
C***** RCF00170
C
C DEFINITION OF DEVICES:
C DEVICE 5 : INPUT FROM TERMINAL RCF00180
C DEVICE 6 : OUTPUT TO TERMINAL RCF00190
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT (LRECL=80) RCF00200
C DEVICE 10 : INPUT FROM FILE CONTAINING APPROXIMATE N-D STATIC RCF00210
C SOLUTION CREATED BY RCSTAT2D (LRECL=132) RCF00220
C DEVICE 11 : OUTPUT TO FILE FOR PLOTS OR INPUT TO A RUN RCF00230
C OF RCLINDY2 (LRECL=117) RCF00240
C RCF00250
C RCF00260
C COMMON BLOCK CONTENTS (OVERALL REFERENCE):
C SOLUT = SOLUTION MATRICES RCF00270
C STAT = STATIC COMPLIANT RISER SOLUTION RCF00280
C STAT1 = X AND Y DISPLACEMENTS AT TOP OF RISER RCF00290
C STAT2 = FUNCTIONS OF STATIC RESULTS RCF00300
C INPUT1 = RISER LENGTH, AVERAGE AND TOTAL WEIGHT AND NUMBER OF SEGMENTS RCF00310
C INPUTL = RISER SEGMENTS LENGTH RCF00320
C COEF = NONDIMENSIONAL RISER CHARACTERISTICS USED IN THE EQUATIONS RCF00330
C COEF1 = NONDIMENSIONAL RISER CHARACTERISTICS AT A SPECIFIC POINT RCF00340
C COUN = INTEGER COUNTING VARIABLE TO DETERMINE EACH DIVISION POINT RCF00350
C FREQ = NONDIMENSIONAL FREQUENCY OF EXCITATION RCF00360
C RCF00370
C RCF00380
IMPLICIT REAL*8 (A-H,O-Z)
PARAMETER(N=6,MNP=151,NA=4,IY=6)
PARAMETER(LW=MNP*(3*N**2+5*N+2)+3*N**2+5*N)
PARAMETER(LIW=MNP*(2*N+1)+N)
DIMENSION W(LW),IW(LIW),C(N,N),D(N,N),GAM(N)
COMMON/SOLUT/X(MNP),Y(N,MNP),MODE
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCF00470
*ZETAM,TMAXAV,DXIM
COMMON/COUN/ICOUNT
COMMON/FREQ/XLAMDA
C EXTERNAL SUBROUTINES USED BY NAG LIBRARY
EXTERNAL FCNF,FCNG
DATA NOUT /6/
C
C READ2D READS 2-D STATIC SOLUTION CALCULATED FROM RCSTAT2D PROGRAM. RCF00550
C IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC EFFECTIVE TENSION RCF00560

```

C          CALL READ2D(NPI)                                RCF00570
C          CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE   RCF00580
C          NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE EQUATIONS.           RCF00590
C          CALL CHARAC(TLENG,IC,NPI)                                RCF00600
C          IF(IC.EQ.0) STOP                                     RCF00610
C          DEFINITIONS OF PARAMETERS ...                         RCF00620
C          N = NUMBER OF EQUATIONS TO BE SOLVED BY DC2GBF      RCF00630
C          MNP= MAXIMUM NUMBER OF PERMITTED POINTS IN THE NON-UNIFORM    RCF00640
C          F.D. MESH ( MNP >= 32)                                RCF00650
C          IF(MNP.LT.32) THEN                                 RCF00660
C              WRITE(6,1257) MNP                               RCF00670
1257    FORMAT(' MNP = ',I3,' IS NOT PERMISSIBLE')        RCF00680
          STOP                                              RCF00690
          ENDIF                                             RCF00700
C          PARAMETER CONTROLLING MONITORING OF CALCULATIONS     RCF00710
          IFAIL=111                                         RCF00720
C          BOUNDARY POINTS FOR TWO-POINT BOUNDARY VALUE PROBLEM   RCF00730
          A=0.DO                                           RCF00740
          B=1.DO                                           RCF00750
C          WRITE(6,*)' INPUT EXCITATION FREQUENCY IN RAD/S'       RCF00760
          READ(5,*) SIGM                                     RCF00770
C          NONDIMENSIONALIZE SIGMA                         RCF00780
          SIGMA = SIGM*TLENG*DSQRT(TMAXAV/TOMAX)            RCF00790
          XLAMDA = SIGMA**2                                RCF00800
          WRITE(6,*)' INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR P'   RCF00810
          READ(5,*) BOUNDP                                RCF00820
          WRITE(6,*)' INPUT AMPLITUDE OF EXCITATION AT TOP OF RISER FOR Q'   RCF00830
          READ(5,*) BOUNDQ                                RCF00840
          WRITE(6,*)' INPUT TOLERANCE FOR CONVERGENCE OF ITERATIONS'      RCF00850
          WRITE(6,*)' SUGGESTED VALUE IS THE EXCITATION AMPLITUDE OR 1/10 OF RCF00860
          * THE EXCITATION AMPLITUDE'                      RCF00870
          READ(5,*) TOL                                    RCF00880
          WRITE(6,*)' INPUT MODE NUMBER CORRESPONDING TO FREQUENCY OF EXCITARCF00890
          *TION'                                         RCF00900
          READ(5,*) MODE                                RCF00910
          NP = NPI                                         RCF00920
          DO 632 I=1,N                                  RCF00930
              GAM(I) = 0.DO                            RCF00940
          DO 632 K=1,N                                  RCF00950
              C(I,K)=0.DO                           RCF00960
              D(I,K)=0.DO                           RCF00970
632    CONTINUE                                     RCF00980
C          DEFINE BOUNDARY CONDITIONS USED IN THE EQUATIONS.      RCF00990
          GAM(5) = BOUNDP                                RCF01000
          GAM(6) = BOUNDQ                                RCF01010
          C(1,4) = 1.DO                                RCF01020
          C(2,5) = 1.DO                                RCF01030
          C(3,6) = 1.DO                                RCF01040
          D(4,4) = 1.DO                                RCF01050
          D(5,5) = -1.DO                               RCF01060
          D(6,6) = 1.DO                                RCF01070

```

```

C      SET COUNTING VARIABLE                               RCF01130
      ICOUNT = 1                                         RCF01140
C
C      CALL X04AAF(1,NOUT)                                RCF01150
C      CALL X04ABF(1,NOUT)                                RCF01160
C
C      CALL NAG SUBROUTINE D02GBF TO SOLVE THE LINEAR TWO-POINT   RCF01170
C      BOUNDARY VALUE PROBLEM USING A DEFERRED CORRECTION TECHNIQUE. RCF01180
C
C      CALL D02GBF(A,B,N,TOL,FCNF,FCNG,C,D,GAM,MNP,X,Y,NP,W,LW,IW,LIW,IFARCF01220
*IL)                                              RCF01210
C
C      WRITE(6,9000) IFAIL                                RCF01230
9000  FORMAT(' IFAIL =',I3)                           RCF01240
C
C      OUTPUT THE RESULTS                               RCF01250
C
C      IF((IFAIL.EQ.0).OR.(IFAIL.EQ.4)) THEN          RCF01260
         CALL OUTPUT(NP)                                RCF01270
      ENDIF                                              RCF01280
C
C      STOP                                               RCF01290
      END                                              RCF01300
R
SUBROUTINE CHARAC(TLENG,IC,NP)                         RCF01310
C
C      THIS SUBROUTINE READS THE COMPLIANT RISER CHARACTERISTICS AND   RCF01320
C      EVALUATES THE NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE   RCF01330
C      GOVERNING EQUATIONS.                                     RCF01340
C
C      IMPLICIT REAL*8(A-H,O-Z)                            RCF01350
C      PARAMETER(MNP=151,NA=4,N=6)                         RCF01360
C      CHARACTER*80 NAME                                 RCF01370
C      COMMON/INPUT1/TLEN,WA,WT,NSEG                      RCF01380
C      COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)                RCF01390
C      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI   RCF01400
C      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCF01410
*ZETAM,TMAXAV,DXIM                                    RCF01420
C      COMMON/STAT2/CONST1(MNP),CONST2(MNP)               RCF01430
C      COMMON/SOLUT/X(MNP),Y(N,MNP),MODE                RCF01440
C      DIMENSION RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP) RCF01450
* ,TMAXI(MNP),TMAZI(MNP)                            RCF01460
C      DIMENSION XJZI(MNP),AJZI(MNP),TJZI(MNP)           RCF01470
C      DIMENSION DXI(MNP),PXIETA(MNP),DETA(MNP),DXIETA(MNP),AO(MNP)   RCF01480
C      DIMENSION WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP),   RCF01490
*EIXIS(MNP),GIP(MNP),GIPS(MNP)                     RCF01500
C      DIMENSION TENO(MNP),QXO(MNP),TOS(MNP),QXIOS(MNP)    RCF01510
C
C      READ RISER CHARACTERISTICS FROM DEVICE 8.          RCF01520
      READ (8,1000) NAME                                RCF01530
1000  FORMAT(80A)                                     RCF01540
      READ (8,1008) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS   RCF01550
1008  FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6))        RCF01560
      TLENG = TLEN                                     RCF01570
      DO 1502 I=1,NSEG                                RCF01580
      READ (8,1003) RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXI(I) RCF01590
* ,PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I)       RCF01600
1502  CONTINUE                                     RCF01610
R

```

FILE: RCFORCE FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

1003 FORMAT(5(1X,D12.6)/6(1X,D12.6))          RCF01690
C
      DO 1332 I=1,NSEG                         RCF01700
      READ(8,1333) AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),AMARCF01720
      *ZI(I),XJZI(I),AJZI(I)                      RCF01730
1332 CONTINUE                                     RCF01740
1333 FORMAT(6(1X,D12.6)/3(1X,D12.6))          RCF01750
C
      IC=1                                         RCF01760
C
7650  WRITE(6,7651)                             RCF01770
7651  FORMAT(' DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS'/
      *' IF YES INPUT 1 , IF NO INPUT 0')        RCF01780
      READ(5,*) IPRINT                          RCF01790
      IF((IPRINT.NE.1).AND.(IPRINT.NE.0)) GOTO 7650
      IF(IPRINT.EQ.1) THEN                      RCF01800
C
      WRITE(6,1000) NAME                         RCF01810
      WRITE(6,2500)                               RCF01820
2500  FORMAT(' NSEG      TLEN           WA       RHOO      AI      RCF01830
      * CFLUID      PRESS')                     RCF01840
      WRITE(6,2001) NSEG,TLEN,WA,RHOO,AI,CFLUID,PRESS
2001  FORMAT(1X,I3,6(1X,D12.6))               RCF01850
      WRITE(6,3400)                               RCF01860
3400  FORMAT('   I     RLENG         RMASS      RMASST    AMAXI   RCF01870
      * WEIGHT'   /'   DXI        PXIETA      EA      EIETA
      *          AO        EIETAS')            RCF01880
      DO 3002 I=1,NSEG                         RCF01890
      WRITE(6,3003) I,RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXIRCF01900
      *(I),PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I) RCF01910
3002 CONTINUE                                     RCF01920
3003  FORMAT(1X,I3,5(1X,D12.6)/4X,6(1X,D12.6)) RCF01930
C
      WRITE(6,1334)                               RCF01940
1334  FORMAT('   I     AMAETA        DETA      EIXI      EIXIS   RCF01950
      *     GIP        GIPS'/'  AMAZI      JZI      AJZI' ) RCF01960
      DO 1335 I=1,NSEG                         RCF01970
      WRITE(6,1336) I,AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),RCF02060
      *AMAZI(I),XJZI(I),AJZI(I)                  RCF02070
1335 CONTINUE                                     RCF02080
1336  FORMAT(1X,I3,6(1X,D12.6)/4X,3(1X,D12.6)) RCF02090
C
1701  WRITE(6,1700)                               RCF02100
1700  FORMAT(' INPUT 1 IF RISER DATA ARE CORRECT'/' INPUT 0 TO STOP') RCF02110
      READ(5,*) IC                            RCF02120
      IF((IC.NE.0).AND.(IC.NE.1)) GOTO 1701
      IF(IC.EQ.0) RETURN                      RCF02130
C
      ENDIF                                     RCF02140
C
C      NON - DIMENSIONALIZATIONS             RCF02150
C
      GRAV=9.81DO                                RCF02160
      WAM=WA/GRAV                                 RCF02170
      WT=WA*TLEN                                 RCF02180
      XPI=4.0*DATAN(1.0)                         RCF02190

```

```

XPI2=XPI/2.DO                                RCF02250
RHOW=1.025D3                                 RCF02260
C
C   NONDIMENSIONALIZE CORRECTLY QXIO, TO FROM 2-D STATIC SOLUTION    RCF02270
C
TOMAX = TOMAX*WT                            RCF02280
TND = WT/TOMAX                             RCF02290
DO 229 I=1,NP
  STATIC(1,I) = STATIC(1,I)*TND            RCF02300
  STATIC(2,I) = STATIC(2,I)*TND            RCF02310
229  CONTINUE                                RCF02320
C
C   NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS      RCF02330
C
TOMAXL = TOMAX/TLEN                         RCF02340
TOML1 = TOMAX*TLEN                          RCF02350
TOML2 = TOML1*TLEN                          RCF02360
TLEN2 = TLEN**2                             RCF02370
TMAXAV = 0.DO                               RCF02380
DXIM = 0.DO                                RCF02390
DO 2000 I=1,NSEG
  RLENG(I)=RLENG(I)/TLEN                  RCF02400
  WEIGHT(I)=WEIGHT(I)/TOMAXL              RCF02410
  EOM(I) = TOMAX/EA(I)                   RCF02420
  EPSETA(I) = EIETA(I)/TOML2             RCF02430
  EPSETS(I) = EIETAS(I)/TOML1            RCF02440
  TJZI(I) = XJZI(I) + AJZI(I)           RCF02450
  TMAZI(I) = RMASS(I) + AMAZI(I)        RCF02460
  TMAXI(I) = RMASS(I) + AMAXI(I)        RCF02470
  TMAXAV = TMAXI(I)*RLENG(I) + TMAXAV  RCF02480
  DXIM = DXIM + DXI(I)*RLENG(I)       RCF02490
  DXIETA(I) = DXI(I)-DETA(I)          RCF02500
2000  CONTINUE                                RCF02510
  HZETAM = 0.DO                            RCF02520
  DO 4321 I=1,NSEG
    HZETA(I) = TMAZI(I)/TMAXAV          RCF02530
    HZETAM = HZETAM + HZETA(I)*RLENG(I) RCF02540
    HXI(I) = TMAXI(I)/TMAXAV          RCF02550
4321  CONTINUE                                RCF02560
C
C   CALCULATE DERIVATIVES OF STATIC QUANTITIES      RCF02570
DO 737 I=1,NP
  TENO(I) = STATIC(1,I)                  RCF02580
  QXO(I) = STATIC(2,I)                  RCF02590
737  CONTINUE                                RCF02600
  CALL DER1(TENO,XI,TOS,NP)            RCF02610
  CALL DER1(QXO,XI,QXIOS,NP)          RCF02620
C
C   EVALUATE FUNCTIONS OF STATIC RESULTS.        RCF02630
DO 56 I=1,NP
  CONST1(I) = TOS(I) - STATIC(2,I)*STATIC(3,I) RCF02640
  CONST2(I) = QXIOS(I) + STATIC(1,I)*STATIC(3,I) RCF02650
56  CONTINUE                                RCF02660
C
C   SEG(I)=LEFT ORDINATE OF SEGMENT I          RCF02670
  SEG(1)=0.DO                            RCF02680
  SEG(NSEG+1)=1.DO                        RCF02690

```

```

        DO 4000 I=2,NSEG
        SEG(I)=RLENG(I-1)+SEG(I-1)                                RCF02810
4000  CONTINUE                                              RCF02820
C   INTRERPOLATE STRUCTURAL DIMENSIONS TO NP POINTS          RCF02830
C   ASSUMPTION: NUMBER OF RISER SEGMENTS WITH DIFFERENT      RCF02840
C   CHARACTERISTICS IS SMALLER THAN THE STATIC SOLUTION POINTS RCF02850
C   NSEG < NPI                                               RCF02860
C
        IF (NSEG.GE.NPI) THEN                                     RCF02870
          WRITE(6,188)                                         RCF02880
188     FORMAT(' NSEG => NPI, PROGRAM STOPS')                  RCF02890
          IC = 0                                              RCF02900
          RETURN                                             RCF02910
        END IF                                              RCF02920
        CALL STRUCT(EPSETA,X,NP)                                 RCF02930
        CALL STRUCT(EPSETS,X,NP)                                 RCF02940
        CALL STRUCT(EOM,X,NP)                                  RCF02950
        CALL STRUCT(HZETA,X,NP)                                 RCF02960
        CALL STRUCT(HXI,X,NP)                                  RCF02970
        RETURN                                             RCF02980
        END                                                 RCF02990
C
C   SUBROUTINE READ2D(NP)
C   THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM RCF03000
C   DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION. RCF03010
C   IMPLICIT REAL*8(A-H,O-Z)                               RCF03020
C   PARAMETER(N=6,MNP=151,NA=4)                           RCF03030
C   COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCF03040
C   COMMON/STAT1/XTOP,YTOP                            RCF03050
C   COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE          RCF03060
C   DIMENSION XCOOR(MNP),YCOOR(MNP)                    RCF03070
C
C   STATIC(1,I) = STATIC TENSION TO                   RCF03080
C   STATIC(2,I) = STATIC SHEAR FORCE IN THE XI DIRECTION RCF03090
C   STATIC(3,I) = STATIC OMEGA IN THE ETA DIRECTION    RCF03100
C   STATIC(4,I) = STATIC ANGLE PHI                      RCF03110
C
        WRITE(6,2000) MNP                                         RCF03120
2000  FORMAT(' MNP=',I3)                                    RCF03130
C   READ STATIC SOLUTION                                 RCF03140
        READ(10,36459) NP,VM                                RCF03150
36459  FORMAT(1X,I3,1X,D12.6)                            RCF03160
        WRITE(6,2311) NP,VM                                RCF03170
2311  FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
          *' NP =',I3/
          *' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S, VM =',D12.6) RCF03180
C
        IF((NP.LT.4).OR.(NP.GT.MNP)) THEN                 RCF03190
          ICCC=0                                         RCF03200
          WRITE(6,12439)                                RCF03210
12439  FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS')       RCF03220
          RETURN                                         RCF03230
        ENDIF                                            RCF03240
C
C   READING FROM DEVICE 10                                RCF03250
C

```

```

DO 1021 I=1,NP                               RCF03370
READ(10,1033) X(I),STATIC(1,I),STATIC(2,I),STATIC(3,I),STATIC(4,I) RCF03380
*,XCOOR(I),YCOOR(I),STRARC,TENSI,VLOCXI(I)
XI(I) = X(I)                                RCF03390
1021 CONTINUE                                 RCF03400
1033 FORMAT(10(1X,D12.6))                     RCF03410
C   EVALUATE RISER TOP X AND Y COORDINATES    RCF03420
XTOP = XCOOR(NP)                            RCF03430
YTOP = YCOOR(NP)                            RCF03440
C   EVALUATE MAXIMUM STATIC EFFECTIVE TENSION RCF03450
TOMAX=DMAX1(STATIC(1,1),STATIC(1,2))        RCF03460
DO 9859 I=3,NP                            RCF03470
TOMAX=DMAX1(TOMAX,STATIC(1,I))            RCF03480
9859 CONTINUE                                 RCF03490
C
      WRITE(6,1654) TOMAX                      RCF03500
1654 FORMAT(' 2-D STATIC SOLUTION SUCCESSFULLY READ'// ' MAXIMUM STATIC EFRFCF03530
*EFFECTIVE TENSION/WA*L = ',D10.4)          RCF03540
C
      RETURN                                     RCF03550
      END                                         RCF03560
C
      SUBROUTINE STRUCT(ARRAY,X,NP)             RCF03570
C   THIS SUBROUTINE CHANGES A SERIES OF DATA GIVEN IN SEGMENTS    RCF03580
C   TO A SERIES OF DATA CORRESPONDING TO SPECIFIC POINTS         RCF03590
C   ASSUMPTION: NSEG < NP                                     RCF03600
IMPLICIT REAL*8(A-H,O-Z)                   RCF03610
PARAMETER(MNP=151)                         RCF03620
COMMON/INPUT1/TLEN,WA,WT,NSEG               RCF03630
COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)         RCF03640
DIMENSION ARRAY(MNP),HELP(MNP),X(MNP)       RCF03650
C
      IF(NSEG.EQ.1) THEN                         RCF03660
        DO 83 I=1,NP                           RCF03670
          HELP(I) = ARRAY(I)                  RCF03680
83      CONTINUE                                 RCF03690
      ELSE                                     RCF03700
        HELP(1) = ARRAY(1)                  RCF03710
        HELP(NP) = ARRAY(NSEG)              RCF03720
        I=2                                    RCF03730
        DO 84 K=2,NP-1                      RCF03740
          IF ((X(K).GT.SEG(I-1)).AND.(X(K).LT.SEG(I))) THEN
            HELP(K) = ARRAY(I-1)            RCF03750
          ELSE IF (X(K).EQ.SEG(I)) THEN
            HELP(K) = 0.5D0*(ARRAY(I-1) + ARRAY(I)) RCF03760
          ELSE IF (X(K).GT.SEG(I)) THEN
            HELP(K) = ARRAY(I)            RCF03770
            I = I + 1                    RCF03780
          END IF                           RCF03790
84      CONTINUE                                 RCF03800
      END IF                                     RCF03810
      DO 85 K=1,NP                           RCF03820
        ARRAY(K) = HELP(K)                  RCF03830
85      CONTINUE                                 RCF03840
      RETURN                                     RCF03850
      END                                         RCF03860

```

FILE: RCFORCE FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

C
C          SUBROUTINE DER1(ARRAY,X,DERIV,NP)                                RCF03930
C          THIS SUBROUTINE EVALUATES THE FIRST DERIVATIVE OF A SERIES OF DATA RCF03940
C          POINTS USING A SECOND ORDER APPROXIMATION (NONUNIFORM GRID FINITE RCF03950
C          DIFFERENCES).                                                 RCF03960
C          IMPLICIT REAL*8(A-H,O-Z)                                         RCF03970
C          PARAMETER (MNP=151)                                              RCF03980
C          DIMENSION ARRAY(MNP),DERIV(MNP),X(MNP)                           RCF03990
C          USE FIRST ORDER FOR END POINTS DERIVATIVES                      RCF04000
C          DERIV(1) = (-ARRAY(1) + ARRAY(2))/(X(2)-X(1))                     RCF04010
C          DERIV(NP) = (-ARRAY(NP-1) + ARRAY(NP))/(X(NP)-X(NP-1))           RCF04020
C
C          DO 836 I=2,NP-1
C              DX = X(I) - X(I-1)                                         RCF04030
C              DX1 = X(I+1) - X(I)                                         RCF04040
C              D = DX/DX1                                              RCF04050
C              DERIV(I) = (D*(ARRAY(I+1)-ARRAY(I)) - (ARRAY(I-1)-ARRAY(I))/D) RCF04060
C              DERIV(I) = DERIV(I)/(DX+DX1)                               RCF04070
836 CONTINUE
      RETURN
      END
C
C          SUBROUTINE OUTPUT(NP)                                         RCF04140
C          THIS SUBROUTINE OUTPUTS THE RESULTS IN A FORM TO BE USED AS INPUT RCF04150
C          FOR A RUN OF RCLINDY2.                                         RCF04160
C          IMPLICIT REAL*8(A-H,O-Z)                                         RCF04170
C          PARAMETER(MNP=151,NA=4,N=6)                                     RCF04180
C          COMMON/INPUT1/TLEN,WA,WT,NSEG                                 RCF04190
C          COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI   RCF04200
C          COMMON/STAT1/XTOP,YTOP                                      RCF04210
C          COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCF04230
C          *ZETAM,TMAXAV,DXIM
C          COMMON/SOLUT/X(MNP),Y(N,MNP),MODE                           RCF04240
C          COMMON/FREQ/XLAMDA                                         RCF04250
C
C          SIGMAD = DSQRT(XLAMDA*TOMAX/TMAXAV)/TLEN                    RCF04260
C          SIGMA = DSQRT(XLAMDA)                                         RCF04270
C
C          OUTPUT TO FILE CONNECTED TO DEVICE 11                         RCF04280
C          THIS CAN BE USED FOR PLOTS OR INPUT TO A RUN OF RCLINDY2       RCF04290
C
C          WRITE(11,36459) MODE,NP,SIGMAD,XTOP,YTOP                      RCF04300
36459 FORMAT(1X,I2,1X,I3,3(1X,D10.4))
DO 3666 I=1,NP
WRITE(11,3667) X(I),(Y(J,I),J=1,6),SIGMA
3666 CONTINUE
3667 FORMAT(8(1X,D12.6))
C
      RETURN
      END
C
C          SUBROUTINE FCNF(X,F)
C          THIS SUBROUTINE EVALUATES THE MATRIX F(X) AT A GENERAL POINT X RCF04450
C          THIS MATRIX IS USED BY NAG/SUBROUTINE D02GBF.                  RCF04460
RCF04470
RCF04480

```

```

IMPLICIT REAL*8(A-H,O-Z) RCF04490
PARAMETER(MNP=151,N=6) RCF04500
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCF04510
*ZETAM,TMAXAV,DXIM RCF04520
COMMON/COEF1/EETA,EETAS,HZET,HX,EOM1,TO,QXIO,OMEGAO,CONS1,CONS2 RCF04530
COMMON/FREQ/XLAMDA RCF04540
DIMENSION F(N,N) RCF04550
C LOCATE MESH POINT TO EVALUATE COEFFICIENTS RCF04560
CALL COUNT(X) RCF04570
C
DO 875 I=1,N RCF04580
  DO 875 K=1,N RCF04590
    F(I,K) = 0.D0 RCF04600
875 CONTINUE RCF04610
C
  F(1,2) = OMEGA0 RCF04620
  F(1,3) = QXIO RCF04630
  F(1,4) = CONS2 RCF04640
  F(1,5) = -XLAMDA*HZET RCF04650
C
  F(2,1) = - OMEGA0 RCF04660
  F(2,3) = -TO RCF04670
  F(2,4) = -CONS1 RCF04680
  F(2,6) = -XLAMDA*HX RCF04690
C
  F(3,2) = -1.D0/EETA RCF04700
  F(3,3) = -EETAS/EETA RCF04710
C
  F(4,3) = 1.D0 RCF04720
C
  F(5,1) = EOM1 RCF04730
  F(5,6) = OMEGA0 RCF04740
C
  F(6,4) = 1.D0 + EOM1*TO RCF04750
  F(6,5) = -OMEGA0 RCF04760
RETURN RCF04770
END RCF04780
C
SUBROUTINE FCNG(X,G) RCF04790
C THIS SUBROUTINE EVALUATES VECTOR G(X) AT A GENERAL POINT X. RCF04800
C IT IS USED BY NAG SUBROUTINE D02GBF. RCF04810
IMPLICIT REAL*8(A-H,O-Z) RCF04820
PARAMETER(N=6) RCF04830
DIMENSION G(N) RCF04840
DO 817 I=1,N RCF04850
  G(I) = 0.D0 RCF04860
817 CONTINUE RCF04870
RETURN RCF04880
END RCF04890
C
SUBROUTINE COUNT(X) RCF04900
C THIS SUBROUTINE EVALUATES THE RISER CHARACTERISTICS AS WELL AS THE RCF05000
C STATIC SOLUTION AT A GENERAL POINT X. RCF05010
C IT RETURNS THEIR VALUE IN COMMON BLOCK COEF1. RCF05020
IMPLICIT REAL*8(A-H,O-Z) RCF05030
PARAMETER(MNP=151,NA=4) RCF05040

```

```

COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCF05050
*ZETAM,TMAXAV,DXIM                                         RCF05060
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI   RCF05070
COMMON/CGEF1/EETA,EETAS,HZET,HX,EOMI,TG,QXIO,OMEGAO,CONS1,CONS2 RCF05080
COMMON/STAT2/CONST1(MNP),CONST2(MNP)                         RCF05090
COMMON/COUN/ICOUNT                                         RCF05100
M = ICOUNT                                                 RCF05110
IF (X.EQ.XI(M)) THEN                                     RCF05120
  EETA = EPSETA(M)
  EETAS = EPSETS(M)
  HX = HXI(M)
  HZET = HZETA(M)
  EOMI = EOM(M)
  TO = STATIC(1,M)
  QXIO = STATIC(2,M)
  OMEGAO = STATIC(3,M)
  CONS1 = CONST1(M)
  CONS2 = CONST2(M)
  ICOUNT = ICOUNT + 1
  IF(X.EQ.1.D0) ICOUNT = 1
ELSE IF ((X.LT.XI(M)).AND.(X.GT.XI(M-1))) THEN        RCF05140
  DIFF = DABS(XI(M)-X)
C THIS IF STATEMENT IS TO ACCOUNT FOR A COMPUTER INACCURACY IN RCF05150
C REPRESENTING AND EQUATING REAL NUMBERS.                  RCF05160
  IF (DIFF.LT.1.D-7) THEN                                 RCF05170
    ICOUNT = ICOUNT + 1
    IF (XI(M).EQ.1.D0) ICOUNT = 1
  END IF                                                 RCF05180
  DX = (X-XI(M-1))/(XI(M)-XI(M-1))
  EETA = EPSETA(M-1) + (EPSETA(M)-EPSETA(M-1))*DX      RCF05190
  EETAS = EPSETS(M-1) + (EPSETS(M)-EPSETS(M-1))*DX     RCF05200
  HX = HXI(M-1) + (HXI(M)-HXI(M-1))*DX                 RCF05210
  HZET = HZETA(M-1) + (HZETA(M)-HZETA(M-1))*DX         RCF05220
  EOMI = EOM(M-1) + (EOM(M)-EOM(M-1))*DX               RCF05230
  TO = STATIC(1,M-1) + (STATIC(1,M)-STATIC(1,M-1))*DX   RCF05240
  QXIO = STATIC(2,M-1) + (STATIC(2,M)-STATIC(2,M-1))*DX RCF05250
  OMEGAO = STATIC(3,M-1) + (STATIC(3,M)-STATIC(3,M-1))*DX RCF05260
  CONS1 = CONST1(M-1) + (CONST1(M)-CONST1(M-1))*DX      RCF05270
  CONS2 = CONST2(M-1) + (CONST2(M)-CONST2(M-1))*DX      RCF05280
ELSE                                                 RCF05290
  WRITE(6,*) ' ERROR OCCURED IN COUNTER, NP=6,M          RCF05300
END IF                                               RCF05310
RETURN                                              RCF05320
END                                                 RCF05330
RCF05340
RCF05350
RCF05360
RCF05370
RCF05380
RCF05390
RCF05400
RCF05410
RCF05420
RCF05430
RCF05440
RCF05450
RCF05460
RCF05470
RCF05480
RCF05490

```

Chapter XVIII

Listing of Program RCLINDY2 FORTRAN A

```

C RCLINDY2                                     RCL00010
C THIS PROGRAM CALCULATES THE IN-PLANE LINEAR DYNAMIC RESPONSE OF A   RCL00020
C COMPLIANT RISER WITH A 2-D STATIC CONFIGURATION BASED ON AN           RCL00030
C EMBEDDING TECHNIQUE. THE INITIAL EMBEDDED APPROXIMATION IS PREPARED   RCL00040
C BY PROGRAM RCFORCE AND THE FINAL SOLUTION IS OBTAINED USING MODIFIED   RCL00050
C NEWTON'S ITERATION AND A NON-UNIFORM GRID FINITE DIFFERENCE METHOD.   RCL00060
C DOUBLE PRECISION AND THE NAG FORTRAN LIBRARY ARE USED.                 RCL00070
C*****RCL00080
C COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY                   RCL00090
C ALL RIGHTS RESERVED.                                                    RCL00100
C*****RCL00110
C PROGRAMMER    GEORGE A. KRIEZIS      JUNE 6, 1985      M.I.T.        RCL00120
C*****RCL00130
C
C DEFINITION OF DEVICES:                                              RCL00140
C DEVICE 5 : INPUT FROM TERMINAL                                     RCL00150
C DEVICE 6 : OUTPUT TO TERMINAL                                    RCL00160
C DEVICE 8 : INPUT FROM FILE PREPARED BY RCINPUT (LRECL=80)          RCL00170
C DEVICE 9 : COMPLETE OUTPUT TO FILE (LRECL=132)                    RCL00180
C DEVICE 10 : INPUT FROM FILE CONTAINING APPROXIMATE N-D STATIC     RCL00190
C             SOLUTION CREATED BY RCSTAT2D (LRECL=132)                  RCL00200
C DEVICE 11 : OUTPUT TO FILE FOR PLOTS OR INPUT TO A NEW RUN        RCL00210
C             OF RCLINDY2 (LRECL=117)                                 RCL00220
C DEVICE 12 : INPUT FROM FILE CONTAINING APPROXIMATE EMBEDDED SOLUTION RCL00230
C             CREATED BY RCFORCE (LRECL=117)                           RCL00240
C
C COMMON BLOCK CONTENTS (OVERALL REFERENCE):                         RCL00250
C SOLUT = INITIAL EMBEDDED APPROXIMATION AND SOLUTION MATRICES       RCL00260
C STAT  = STATIC COMPLIANT RISER SOLUTION                            RCL00270
C STAT1 = X AND Y DISPLACEMENTS AT TOP OF RISER                      RCL00280
C STAT2 = FUNCTIONS OF STATIC RESULTS                                RCL00290
C INPUT0 = OUTPUT FILE HEADING                                         RCL00300
C INPUT1 = RISER LENGTH, AVERAGE AND TOTAL WEIGHT AND NUMBER OF SEGMENTS RCL00310
C INPUT2 = RISER TUBE INTERIOR CHARACTERISTICS                         RCL00320
C INPUTL = RISER SEGMENTS LENGTH                                       RCL00330
C INPUT3 = WEIGHT, STIFFNESSES AND STIFFNESS DERIVATIVES              RCL00340
C INPUT4 = CHARACTERISTIC DIAMETERS AND PERIMETER OF RISER TUBE.       RCL00350
C INPUT5 = MASS AND ADDED MASS CHARACTERISTICS IN THREE DIRECTIONS    RCL00360
C INPUT6 = RISER POLAR AND ADDED POLAR MOMENT OF INERTIA.               RCL00370
C COEF  = NONDIMENSIONAL RISER CHARACTERISTICS USED IN THE EQUATIONS  RCL00380
C COEF1 = NONDIMENSIONAL RISER CHARACTERISTICS AT A SPECIFIC POINT    RCL00390
C CONST = PROBLEM CONSTANTS (PI, GRAVITY, WATER DENSITY)              RCL00400
C BOUND1 = BOUNDARY CONDITION FOR OMEGA ETA AT S=0 AND EMBEDDED BOUNDARY RCL00410
C           CONDITIONS FROM PROGRAM RCFORCE                           RCL00420
C COUN  = INTEGER COUNTING VARIABLE TO DETERMINE EACH DIVISION POINT RCL00430
C
C IMPLICIT REAL*8(A-H,O-Z)                                           RCL00440
C PARAMETER(N=7,MNP=151,NA=4,IY=7)                                     RCL00450
C PARAMETER(LWORK=MNP*(3*N**2+6*N+2)+4*N**2+3*N)                     RCL00460
C PARAMETER(LIWORK=MNP*(2*N+1)+N)                                       RCL00470
C DIMENSION WORK(LWORK),IWORK(LIWORK)                                    RCL00480
C COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE                           RCL00490
C COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCXI(MNP),TOMAX,VM,NPI          RCL00500
C COMMON/COUN/ICOUNT                                         RCL00510
C EXTERNAL SUBROUTINES USED BY NAG LIBRARY                           RCL00520
C EXTERNAL FCN,G,JACEPS,JACGEP,JACOBF,JACOFG                         RCL00530
C

```

```

      DATA NOUT /6/                                     RCL00570
C
C      READ2D READS 2-D STATIC SOLUTION CALCULATED FROM RCSTAT2D PROGRAM. RCL00580
C      IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC EFFECTIVE TENSION   RCL00590
C
C      CALL READ2D(NPI)                                         RCL00600
C
C      CHARAC READS RISER CHARACTERISTICS FROM UNIT 8 AND EVALUATES THE    RCL00610
C      NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE EQUATIONS.           RCL00620
C
C      CALL CHARAC(TLENG,IC,NPI)                                         RCL00630
C      IF(IC.EQ.0) STOP                                              RCL00640
C
C      DEFINITIONS OF PARAMETERS ...
C      N = NUMBER OF EQUATIONS TO BE SOLVED BY DO2RAF                 RCL00650
C      IY = NUMBER OF VARIABLES                                         RCL00660
C      MNP= MAXIMUM NUMBER OF PERMITTED POINTS IN THE NON-UNIFORM      RCL00670
C      F.D. MESH ( MNP >= 32)                                         RCL00680
C      IF(MNP.LT.32) THEN
C          WRITE(6,1257) MNP
1257    FORMAT(' MNP = ',I3,' IS NOT PERMISSIBLE')
          STOP
      ENDIF
C      NUMBER OF BOUNDARY CONDITIONS AT S=0                           RCL00690
1276    NUMBEG=4                                         RCL00700
C      NUMBER OF MIXED BOUNDARY CONDITIONS                         RCL00710
      NUMMIX=0                                         RCL00720
C      PARAMETER DECLARING THAT INITIAL APPROXIMATION IS USER SPECIFIED RCL00730
      INIT=1                                         RCL00740
C      PARAMETER DECLARING THAT JACOBIANS ARE USER SPECIFIED            RCL00750
      IJAC=1                                         RCL00760
C      PARAMETER CONTROLLING MONITORING OF CALCULATIONS             RCL00770
      IFAIL=111                                         RCL00780
C
C      READAS : READS APPROXIMATE EMBEDDED SOLUTION FROM DEVICE 10
C              IT ALSO PROVIDES ICCC, NP, TOL= TOLERANCE OF ITERATIONS
C
C      CALL READAS(ICCC,NP,TOL)                                     RCL00790
C
C      IF THE NUMBER OF POINTS FROM THIS SOLUTION IS GREATER THAN THE
C      STATIC SOLUTION POINTS, INTERPOLATE ALL CHARACTERISTICS TO THE
C      NEW NUMBER OF POINTS.                                         RCL00800
C
C      ASSUMPTION : NP ALWAYS GREATER OR EQUAL TO NPI                RCL00810
C      THIS IS VALID IF THE SAME STATIC SOLUTION WAS USED TO OBTAIN THE
C      APPROXIMATE EMBEDDED SOLUTION IN DEVICE 10.                  RCL00820
C
C      IF (NP.GE.NPI) THEN
C          CALL INTERP(NP)
C          NPI=NP
C      ELSE IF (NP.LT.NPI) THEN
C          WRITE(6,314)
314    FORMAT(' APPROXIMATE SOLUTION IN DEVICE 10 WAS NOT OBTAINED'/
*          ' FROM THE SAME STATIC SOLUTION, PROGRAM STOPS')
          STOP
      END IF

```

```

C
C      INCREMENT OF CONTINUATION PARAMETER          RCL01130
7891  WRITE(6,7890)                                RCL01140
7890  FORMAT(' INPUT INITIAL INCREMENT OF CONTINUATION PARAMETER DELEPS') RCL01150
      /*' IF NO CONTINUATION IS REQUIRED INPUT DELEPS = 1.D0'/
      '*' IF CONTINUATION IS REQUIRED THEN 0.D0 < DELEPS < 1.D0'/
      '*' RECOMMENDATION :'/                         RCL01170
      '*' USUALLY DELEPS = 0.1D0 WILL SUFFICE'/
      '*' FOR LARGE CHANGES BETWEEN INITIAL AND FINAL PROBLEM'/
      '*' A SMALLER VALUE OF DELEPS, E.G. 0.05D0 OR 0.025D0 MIGHT BE'/
      '*' NECESSARY)
      READ(5,*) DELEPS
      IF((DELEPS.GT.1.D0).OR.(DELEPS.LE.0.D0)) THEN
         WRITE(6,7892)
7892  FORMAT(' 0. < DELEPS <= 1.')
         GOTO 7891
      END IF
C      SET COUNTING VARIABLE                      RCL01270
      ICOUNT = 1
      IF(ICCC.EQ.0) STOP                           RCL01280
C
      CALL X04AAF(1,NOUT)                          RCL01290
      CALL X04ABF(1,NOUT)                          RCL01300
C
      CALL NAG SUBROUTINE DO2RAF TO SOLVE THE TWO-POINT BOUNDARY VALUE RCL01310
      C      PROBLEM USING A DEFERRED CORRECTION TECHNIQUE AND NEWTON'S RCL01320
      C      ITERATION.                                RCL01330
      CALL DO2RAF(N,MNP,NP,NUMBEG,NUMMIX,TOL,INIT,X,Y,IY,ABT,FCN,G,IJAC,RCL01400
      *JACOBF,JACOBG,DELEPS,JACEPS,JACGEP,WORK,LWORK,IWORK,LIWORK,IFAIL) RCL01410
C
      WRITE(6,9000) IFAIL                         RCL01420
9000  FORMAT(' IFAIL =',I3)                      RCL01430
C
C      OUTPUT THE RESULTS                        RCL01440
C
      IF((IFAIL.EQ.0).OR.(IFAIL.EQ.4)) THEN
         CALL OUTPUT(NP)
      ENDIF
C
      STOP
      END

      SUBROUTINE CHARAC(TLEN,IC,NP)
C      THIS SUBROUTINE READS THE COMPLIANT RISER CHARACTERISTICS AND RCL01550
C      EVALUATES THE NONDIMENSIONAL COEFFICIENTS TO BE USED IN THE RCL01560
C      GOVERNING EQUATIONS.                         RCL01570
      IMPLICIT REAL*8(A-H,O-Z)                    RCL01580
      PARAMETER(MNP=151,NA=4,N=7)                  RCL01590
      CHARACTER*80 NAME                           RCL01600
      COMMON/CONST/XPI,XPI2,RHOW,GRAV            RCL01610
      COMMON/INPUT0/NAME                         RCL01620
      COMMON/INPUT1/TLEN,WA,WT,NSEG               RCL01630
      COMMON/INPUT2/RHO0,AI,CFLUID,PRESS,AO(MNP) RCL01640
      COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)        RCL01650
      COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EITAS(MNP),EIXI(MNP) RCL01660
      *,EIXIS(MNP),GIP(MNP),GIPS(MNP)           RCL01670
      RCL01680

```

```

COMMON/INPUT4/DXI(MNP),PXIETA(MNP),DETA(MNP),DXIETA(MNP) RCL01690
COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP),TMAXI(MNP),TMAZI(MNP) RCL01700
*      RCL01710
COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP) RCL01720
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI RCL01730
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL01740
*      RCL01750
*ZETAM,TMAXAV,DXIM RCL01760
COMMON/STAT2/CONST1(MNP),CONST2(MNP) RCL01770
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL01780
DIMENSION TENO(MNP),QXO(MNP),TOS(MNP),QXIOS(MNP) C RCL01790
C      READ RISER CHARACTERISTICS FROM DEVICE 8
READ (8,1000) NAME RCL01800
1000 FORMAT(8OA) RCL01810
      READ (8,1008) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS RCL01820
1008 FORMAT(1X,I3,2(1X,D12.6)/4(1X,D12.6)) RCL01830
      TLENG = TLEN RCL01840
      DO 1502 I=1,NSEG RCL01850
      READ (8,1003) RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXI(IRCL01870
*      ,PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I) RCL01860
1502 CONTINUE RCL01880
1003 FORMAT(5(1X,D12.6)/6(1X,D12.6)) RCL01890
C      RCL01900
      DO 1332 I=1,NSEG RCL01910
      READ(8,1333) AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),AMARCL01930
*      ,ZI(I),XJZI(I),AJZI(I) RCL01920
1332 CONTINUE RCL01940
1333 FORMAT(6(1X,D12.6)/3(1X,D12.6)) RCL01950
C      RCL01960
      IC=1 C RCL01970
C      RCL01980
7650  WRITE(6,7651) RCL01990
7651  FORMAT(' DO YOU WISH A TERMINAL COPY OF RISER CHARACTERISTICS'/
*' IF YES INPUT 1 , IF NO INPUT 0') RCL02000
      READ(5,*) IPRINT RCL02010
      IF((IPRINT.NE.1).AND.(IPRINT.NE.0)) GOTO 7650 RCL02020
      IF(IPRINT.EQ.1) THEN RCL02030
      RCL02040
      RCL02050
C      RCL02060
      WRITE(6,1000) NAME RCL02070
      WRITE(6,2500) RCL02080
2500  FORMAT(' NSEG      TLEN          WA          RHO0          AI
*      CFLUID          PRESS') RCL02090
      WRITE(6,2001) NSEG,TLEN,WA,RHO0,AI,CFLUID,PRESS RCL02100
2001  FORMAT(1X,I3,6(1X,D12.6)) RCL02110
      WRITE(6,3400) RCL02120
3400  FORMAT('   I      RLENG          RMASS          RMASST          AMAXI
*      WEIGHT'   /'          DXI          PXIETA          EA          EIETA RCL02130
*      AO          EIETAS') RCL02140
      DO 3002 I=1,NSEG RCL02150
      WRITE(6,3003) I,RLENG(I),RMASS(I),RMASST(I),AMAXI(I),WEIGHT(I),DXIRCLO2180
*      ,(I),PXIETA(I),EA(I),EIETA(I),AO(I),EIETAS(I) RCL02160
3002 CONTINUE RCL02170
3003  FORMAT(1X,I3,5(1X,D12.6)/4X,6(1X,D12.6)) RCL02190
C      RCL02200
      WRITE(6,1334) RCL02210
1334  FORMAT('   I      AMAETA          DETA          EIXI          EIXIS') RCL02220
*      RCL02230
      RCL02240

```

FILE: RCLINDY2 FORTRAN A

VM/SP CONVERSATIONAL MONITOR SYSTEM

```

*      GIP          GIPS'/'      AMAZI        JZI        AJZI' )    RCL02250
DO 1335 I=1,NSEG
      WRITE(6,1336) I,AMAETA(I),DETA(I),EIXI(I),EIXIS(I),GIP(I),GIPS(I),RCL02270
      *AMAZI(I),XJZI(I),AJZI(I)
1335 CONTINUE
1336 FORMAT(1X,I3,6(1X,D12.6)/4X,3(1X,D12.6))
C
1701 WRITE(6,1700)
1700 FORMAT(' INPUT 1 IF RISER DATA ARE CORRECT'/' INPUT 0 TO STOP')
      READ(5,*) IC
      IF((IC.NE.0).AND.(IC.NE.1)) GOTO 1701
      IF(IC.EQ.0) RETURN
C
      ENDIF
C
C      NON - DIMENSIONALIZATIONS
C
      GRAV=9.81D0
      WAM=WA/GRAV
      WT=WA*TLEN
      XPI=4.D0*DATAN(1.D0)
      XPI2=XPI/2.D0
      RHOW=1.025D3,
C
C      NONDIMENSIONALIZE CORRECTLY QXIO, TO FROM 2-D STATIC SOLUTION
C
      TOMAX = TOMAX*WT
      TND = WT/TOMAX
      DO 229 I=1,NP
          STATIC(1,I) = STATIC(1,I)*TND
          STATIC(2,I) = STATIC(2,I)*TND
229  CONTINUE
C
C      NONDIMENSIONAL COEFFICIENTS USED IN THE EQUATIONS
C
      TOMAXL = TOMAX/TLEN
      TOML1 = TOMAX*TLEN
      TOML2 = TOML1*TLEN
      TLEN2 = TLEN**2
      TMAXAV = 0.D0
      DXIM = 0.D0
      DO 2000 I=1,NSEG
          RLENG(I)=RLENG(I)/TLEN
          WEIGHT(I)=WEIGHT(I)/TOMAXL
          EOM(I) = TOMAX/EA(I)
          EPSETA(I) = EIETA(I)/TOML2
          EPSETS(I) = EIETAS(I)/TOML1
          TJZI(I) = XJZI(I) + AJZI(I)
          TMAZI(I) = RMASS(I) + AMAZI(I)
          TMAXI(I) = RMASS(I) + AMAXI(I)
          TMAXAV = TMAXI(I)*RLENG(I) + TMAXAV
          DXIM = DXIM + DXI(I)*RLENG(I)
          DXIETA(I) = DXI(I)-DETA(I)
2000 CONTINUE
      HZETAM = 0.D0
      DO 4321 I=1,NSEG

```

```

HZETA(I) = TMAZI(I)/TMAXAV                                RCL02810
HZETAM = HZETAM + HZETA(I)*RLENG(I)                      RCL02820
HXI(I) = TMAXI(I)/TMAXAV                                  RCL02830
4321 CONTINUE                                              RCL02840
C
C      CALCULATE DERIVATIVES OF STATIC QUANTITIES          RCL02850
DO 737 I=1,NP                                            RCL02860
    TENO(I) = STATIC(1,I)                                 RCL02870
    QXO(I) = STATIC(2,I)                                 RCL02880
737 CONTINUE                                              RCL02890
CALL DER1(TENO,XI,TOS,NP)                               RCL02900
CALL DER1(QXO,XI,QXIOS,NP)                            RCL02910
C
C      EVALUATE FUNCTIONS OF STATIC RESULTS                RCL02920
DO 56 I=1,NP                                            RCL02930
    CONST1(I) = TOS(I) - STATIC(2,I)*STATIC(3,I)        RCL02940
    CONST2(I) = QXIOS(I) + STATIC(1,I)*STATIC(3,I)       RCL02950
56 CONTINUE                                              RCL02960
C
C      SEG(I)=LEFT ORDINATE OF SEGMENT I                  RCL02970
SEG(1)=0.D0                                              RCL02980
SEG(NSEG+1)=1.D0                                         RCL02990
DO 4000 I=2,NSEG                                         RCL03000
    SEG(I)=RLENG(I-1)+SEG(I-1)                          RCL03010
4000 CONTINUE                                              RCL03020
C
C      INTRERPOLATE STRUCTURAL DIMENSIONS TO NP POINTS   RCL03030
C
C      ASSUMPTION: NUMBER OF RISER SEGMENTS WITH DIFFERENT
C      CHARACTERISTICS IS SMALLER THAN THE STATIC SOLUTION POINTS
C      NSEG < NPI                                         RCL03040
C
IF (NSEG.GE.NPI) THEN                                     RCL03050
    WRITE(6,188)                                         RCL03060
188 FORMAT(' NSEG => NPI, PROGRAM STOPS')
    IC = 0
    RETURN
END IF
CALL STRUCT(EPSETA,X,NP)
CALL STRUCT(EPSETS,X,NP)
CALL STRUCT(EOM,X,NP)
CALL STRUCT(HZETA,X,NP)
CALL STRUCT(HXI,X,NP)
RETURN
END

C
SUBROUTINE READ2D(NP)
C
C      THIS SUBROUTINE READS THE STATIC COMPLIANT RISER SOLUTION FROM
C      DEVICE 10. IT EVALUATES THE MAXIMUM NONDIMENSIONAL STATIC TENSION. RCL03250
IMPLICIT REAL*8(A-H,O-Z)
PARAMETER(N=7,MNP=151,NA=4)
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI
COMMON/STAT1/XTOP,YTOP
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
DIMENSION XCOOR(MNP),YCOOR(MNP)

C
STATIC(1,I) = STATIC EFFECTIVE TENSION TO               RCL03330
STATIC(2,I) = STATIC SHEAR FORCE IN THE XI DIRECTION  RCL03340
STATIC(3,I) = STATIC OMEGA IN THE ETA DIRECTION       RCL03350
C
RCL03360

```

```

C      STATIC(4,I) = STATIC ANGLE PHI          RCL03370
C
C      WRITE(6,2000) MNP                      RCL03380
2000  FORMAT(' MNP=',I3)                   RCL03390
C      READ STATIC SOLUTION                  RCL03400
      READ(10,36459) NP,VM                 RCL03410
36459 FORMAT(1X,I3,1X,D12.6)             RCL03420
      WRITE(6,2311) NP,VM                 RCL03430
2311  FORMAT(' 2-D STATIC SOLUTION FROM DEVICE 10'/
*' NP =',I3/                         RCL03440
*' MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S ,VM =',D12.6) RCL03450
C
      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN    RCL03460
        ICCC=0                           RCL03470
        WRITE(6,12439)                   RCL03480
12439  FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS') RCL03490
      RETURN                           RCL03500
      ENDIF                            RCL03510
C
C      READING FROM DEVICE 10              RCL03520
C
      DO 1021 I=1,NP                    RCL03530
      READ(10,1033) X(I),STATIC(1,I),STATIC(2,I),STATIC(3,I),STATIC(4,I) RCL03540
*,XCOOR(I),YCOOR(I),STRARC,TENSI,VLOCKI(I)
      XI(I) = X(I)                     RCL03550
1021  CONTINUE                        RCL03560
      FORMAT(10(1X,D12.6))            RCL03570
C      EVALUATE RISER TOP X AND Y COORDINATES RCL03580
      XTOP = XCOOR(NP)                RCL03590
      YTOP = YCOOR(NP)                RCL03600
C      EVALUATE MAXIMUM STATIC EFFECTIVE TENSION RCL03610
      TOMAX=DMAX1(STATIC(1,1),STATIC(1,2)) RCL03620
      DO 9859 I=3,NP                 RCL03630
      TOMAX=DMAX1(TOMAX,STATIC(1,I))  RCL03640
9859  CONTINUE                        RCL03650
C
      WRITE(6,1654) TOMAX               RCL03660
1654  FORMAT(' 2-D STATIC SOLUTION SUCCESFULLY READ'/' MAXIMUM STATIC EFRCL03670
*,EFFECTIVE TENSION/WA*L = ',D10.4) RCL03680
C
      RETURN                           RCL03690
      END                             RCL03700
C
      SUBROUTINE READAS(ICCC,NP,TOL)     RCL03710
      IMPLICIT REAL*8(A-H,O-Z)          RCL03720
      PARAMETER(N=7,MNP=151,NA=4)       RCL03730
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE RCL03740
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKI(MNP),TOMAX,VM,NPI RCL03750
      COMMON/STAT1/XTOP,YTOP           RCL03760
      COMMON/BOUNDA/BOUND,BOUNDP,BOUNDO RCL03770
C
      READAS READS INITIAL EMBEDDED APPROXIMATION RESULTING FROM A RCL03780
      RUN OF RCFORCE                   RCL03790
C
      Y(1,I) = DYNAMIC TENSION         RCL03800
      Y(2,I) = SHEAR FORCE IN THE XI DIRECTION RCL03810
C

```

```

C      Y(3,I) = OMEGA ABOUT ETA                      RCL03930
C      Y(4,I) = DYNAMIC ANGLE PHI                     RCL03940
C      Y(5,I) = DISPLACEMENT IN THE ZI DIRECTION, P (TANGENTIAL) RCL03950
C      Y(6,I) = DISPLACEMENT IN THE XI DIRECTION, Q (NORMAL)   RCL03960
C      Y(7,I) = NATURAL FREQUENCY                      RCL03970
C      X(I)    = UNSTRETCHED ARC LENGTH S              RCL03980
C
C      WRITE(6,2000) MNP
2000 FORMAT(' MNP=',I3)                                RCL03990
C
C      READ(12,36459) MODE,NP,SIGMAD,XTOP,YTOP          RCL04000
36459 FORMAT(1X,I2,1X,I3,3(1X,D10.4))                RCL04010
C      WRITE(6,2311) NP,MODE,SIGMAD,XTOP,YTOP          RCL04020
2311 FORMAT(' INITIAL APPROXIMATION FROM DEVICE 12'/
*' NP =',I3,' MODE NUMBER =',I2,' SIGMAD =',D10.4/' X AT TOP =',
*D10.4,' Y AT TOP =',D10.4)                         RCL04030
C
C      IF((NP.LT.4).OR.(NP.GT.MNP)) THEN               RCL04040
      ICCC=0                                         RCL04050
      WRITE(6,12439)                                  RCL04060
12439 FORMAT(' NP IS INAPPROPRIATE ; RUN STOPS')     RCL04070
      RETURN                                         RCL04080
      ENDIF                                         RCL04090
C
C      READING DATA FROM DEVICE 12 ...
C
      DO 10011 I=1,NP                                RCL04100
      READ(12,10012) X(I),(Y(J,I),J=1,7)             RCL04110
10011 CONTINUE                                     RCL04120
10012 FORMAT(8(1X,D12.6))                         RCL04130
C      EMBEDDED BOUNDARY CONDITIONS AND OMEGA ETA(0) RCL04140
      BOUND = Y(3,1)                                 RCL04150
      BOUNDP = Y(5,NP)                               RCL04160
      BOUNDQ = Y(6,NP)                               RCL04170
C
      WRITE(9,1052) NP,SIGMAD                         RCL04180
1052 FORMAT(' INITIAL CONDITION FOR EPS=0. AND NP = ',I3,' POINTS, NATRCL04290
      *URAL FREQUENCY = ',D10.4,' RAD/S'/
      *'      I      ARC      TENSION      QXI      OMEGA ETA      PHI
      *      P          Q          SIGMA')            RCL04300
      DO 1601 I=1,NP                                RCL04310
      WRITE(9,1603) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
      * Y(7,I)                                         RCL04320
1601 CONTINUE                                     RCL04330
1603 FORMAT(1X,I3,8(1X,D12.6))                   RCL04340
C
      WRITE(6,9561)
9561 FORMAT(' DO YOU WISH A TERMINAL COPY OF INITIAL APPROXIMATION'/
      *' IF YES INPUT 1')
      READ(5,*) IPRINT
      IF(IPRINT.EQ.1) THEN
      WRITE(6,1052) NP
      DO 9659 I=1,NP
      WRITE(6,1603) I,X(I),Y(1,I),Y(2,I),Y(3,I),Y(4,I),Y(5,I),Y(6,I),
      * Y(7,I)
9659 CONTINUE                                     RCL04350
RCL04360
RCL04370
RCL04380
RCL04390
RCL04400
RCL04410
RCL04420
RCL04430
RCL04440
RCL04450
RCL04460
RCL04470
RCL04480

```

```

        END IF
        WRITE(6,1722) BOUND
1722 FORMAT(' ASSUMED BOUNDARY CONDITION '/
*' OMEGA ETA (0) =',D12.6)
C
        TOL1 = 0.D0
        DO 3931 I=1,NP
          TOL1 = DMAX1(TOL1,DABS(Y(3,I)))
3931 CONTINUE
        WRITE(6,3932) TOL1
3932 FORMAT(' MAXIMUM ABSOLUTE VALUE OF N-D OMEGA ETA IS =',D12.6/
*' THIS NUMBER CAN BE USED TO ESTIMATE '/
*' A REASONABLE TOLERANCE FOR CONVERGENCE OF ITERATIONS'/
*' INPUT A FRACTION OF THIS NUMBER TO DETERMINE THE TOLERANCE'/
*' E.G. INPUT 0.01 OR 0.1 IF THE INITIAL ESTIMATE FOR OMEGA ETA'/
*' IS SMALL'/' AFTER THE RUN CHECK THE ACCURACY OF THE SOLUTION')
        READ(5,*) TOLV
        TOL = DABS(TOLV)*TOL1
C
1005 WRITE(6,1004)
1004 FORMAT(' IF YOU WISH TO STOP INPUT 0'/
*' IF YOU WISH TO CONTINUE WITH ITERATIONS INPUT 1')
        READ(5,*) ICCC
        IF((ICCC.NE.0).AND.(ICCC.NE.1)) GOTO 1005
C
        RETURN
        END
C
        SUBROUTINE STRUCT(ARRAY,X,NP)
C THIS SUBROUTINE CHANGES A SERIES OF DATA GIVEN IN
C SEGMENTS TO A SERIES OF DATA CORRESPONDING TO SPECIFIC POINTS
C ASSUMPTION: NSEG < NP
C IMPLICIT REAL*8(A-H,O-Z)
C PARAMETER(MNP=151)
C COMMON/INPUT1/TLEN,WA,WT,NSEG
C COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)
C DIMENSION ARRAY(MNP),HELP(MNP),X(MNP)
C
        IF(NSEG.EQ.1) THEN
          DO 83 I=1,NP
            HELP(I) = ARRAY(I)
83      CONTINUE
        ELSE
          HELP(1) = ARRAY(1)
          HELP(NP) = ARRAY(NSEG)
          I=2
          DO 84 K=2,NP-1
            IF ((X(K).GT.SEG(I-1)).AND.(X(K).LT.SEG(I))) THEN
              HELP(K) = ARRAY(I-1)
            ELSE IF (X(K).EQ.SEG(I)) THEN
              HELP(K) = 0.5D0*(ARRAY(I-1) + ARRAY(I))
            ELSE IF (X(K).GT.SEG(I)) THEN
              HELP(K) = ARRAY(I)
              I = I + 1
            END IF
84      CONTINUE
        END IF
      
```

RCL04490
RCL04500
RCL04510
RCL04520
RCL04530
RCL04540
RCL04550
RCL04560
RCL04570
RCL04580
RCL04590
RCL04600
RCL04610
RCL04620
RCL04630
RCL04640
RCL04650
RCL04660
RCL04670
RCL04680
RCL04690
RCL04700
RCL04710
RCL04720
RCL04730
RCL04740
RCL04750
RCL04760
RCL04770
RCL04780
RCL04790
RCL04800
RCL04810
RCL04820
RCL04830
RCL04840
RCL04850
RCL04860
RCL04870
RCL04880
RCL04890
RCL04900
RCL04910
RCL04920
RCL04930
RCL04940
RCL04950
RCL04960
RCL04970
RCL04980
RCL04990
RCL05000
RCL05010
RCL05020
RCL05030
RCL05040

```

        END IF
        DO 85 K=1,NP
          ARRAY(K) = HELP(K)
85 CONTINUE
      RETURN
      END
C
      SUBROUTINE DER1(ARRAY,X,DERIV,NP)
C      THIS SUBROUTINE EVALUATES THE FIRST DERIVATIVE OF A SERIES OF DATA
C      POINTS USING A SECOND ORDER APPROXIMATION (NONUNIFORM GRID FINITE
C      DIFFERENCES).
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER (MNP=151)
      DIMENSION ARRAY(MNP),DERIV(MNP),X(MNP)
C      USE FIRST ORDER FOR END POINTS DERIVATIVES
      DERIV(1) = (-ARRAY(1) + ARRAY(2))/(X(2)-X(1))
      DERIV(NP) = (-ARRAY(NP-1) + ARRAY(NP))/(X(NP)-X(NP-1))
C
      DO 836 I=2,NP-1
        DX = X(I) - X(I-1)
        DX1 = X(I+1) - X(I)
        D = DX/DX1
        DERIV(I) = (D*(ARRAY(I+1)-ARRAY(I)) - (ARRAY(I-1)-ARRAY(I))/D)
        DERIV(I) = DERIV(I)/(DX+DX1)
836 CONTINUE
      RETURN
      END
C
      SUBROUTINE OUTPUT(NP)
C      THIS SUBROUTINE OUTPUTS THE RESULTS IN TWO FORMATS; A COMPLETE
C      FORMAL FORM AND A FORM TO BE USED FOR PLOTTING.
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(MNP=151,NA=4,N=7)
      CHARACTER*80 NAME
      COMMON/CONST/XPI,XPI2,RHOW,GRAV
      COMMON/INPUT/NAME
      COMMON/INPUT1/TLEN,WA,WT,NSEG
      COMMON/INPUT2/RHO0,AI,CFLUID,PRESS,AO(MNP)
      COMMON/INPUTL/RLENG(MNP),SEG(MNP+1)
      COMMON/INPUT3/WEIGHT(MNP),EA(MNP),EIETA(MNP),EIETAS(MNP),EIXI(MNP)
      *,EIXIS(MNP),GIP(MNP),GIPS(MNP)
      COMMON/INPUT4/DXI(MNP),PXIETA(MNP),DETA(MNP),DXIETA(MNP)
      COMMON/INPUT5/RMASS(MNP),RMASST(MNP),AMAXI(MNP),AMAETA(MNP),AMAZI(MNP),
      *TMAXI(MNP),TMAZI(MNP)
      COMMON/INPUT6/XJZI(MNP),AJZI(MNP),TJZI(MNP)
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKI(MNP),TOMAX,VM,NPI
      COMMON/STAT1/XTOP,YTOP
      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL05520
      *ZETAM,TMAXAV,DXIM
      COMMON/COEF1/EETA,EETAS,HZET,HX,EOMI,TO,QXIO,OMEGAO,CONS1,CONS2
      COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE
      DIMENSION A1(MNP)
C      CALCULATE THE ORTHONORMALIZING CONSTANT FOR P AND Q
      DO 542 I=1,NP
        CALL COUNT(X(I))

```

RCL05050
RCL05060
RCL05070
RCL05080
RCL05090
RCL05100
RCL05110
RCL05120
RCL05130
RCL05140
RCL05150
RCL05160
RCL05170
RCL05180
RCL05190
RCL05200
RCL05210
RCL05220
RCL05230
RCL05240
RCL05250
RCL05260
RCL05270
RCL05280
RCL05290
RCL05300
RCL05310
RCL05320
RCL05330
RCL05340
RCL05350
RCL05360
RCL05370
RCL05380
RCL05390
RCL05400
RCL05410
RCL05420
RCL05430
RCL05440
RCL05450
RCL05460
RCL05470
RCL05480
RCL05490
RCL05500
RCL05510
RCL05520
RCL05530
RCL05540
RCL05550
RCL05560
RCL05570
RCL05580
RCL05590
RCL05600

```

      A1(I) = Y(5,I)**2*HZET + HX*Y(6,I)**2          RCL05610
542 CONTINUE                                         RCL05620
      A = 0.0D0                                         RCL05630
      DO 642 I=2,NP                                    RCL05640
         DX2 = (X(I)-X(I-1))*0.5D0                   RCL05650
         A = A + (A1(I)+A1(I-1))*DX2                 RCL05660
642 CONTINUE                                         RCL05670
      A = DSQRT(A)                                     RCL05680
C      ORTHONORMALIZE THE SOLUTION EXCEPT SIGMA      RCL05690
      DO 643 I=1,NP                                    RCL05700
         Y(N,I) = DABS(Y(N,I))                      RCL05710
      DO 643 K=1,N-1                                 RCL05720
         Y(K,I) = Y(K,I)/A                         RCL05730
643 CONTINUE                                         RCL05740
C      OUTPUT RESULTS                                RCL05750
      WRITE(9,1000) NAME                            RCL05760
1000 FORMAT(80A)                                     RCL05770
      WRITE(9,1001) NSEG,TLEN,WA,RHO0,RHOW,AI,CFLUID,PRESS,WT,TOMAX,VM, RCL05780
      *XTOP*TLEN,YTOP*TLEN                           RCL05790
C
1001 FORMAT(1X,I3,9X,' = NUMBER OF RISER SEGMENTS'/
      *1X,D12.6,' = UNSTRETCHED RISER LENGTH IN M'/
      *1X,D12.6,' = APPROXIMATE AVERAGE EFFECTIVE WEIGHT PER UNIT LENGTH RCL05800
      *IN N/M'/
      *1X,D12.6,' = INNER FLUID DENSITY IN KG/M3'/
      *1X,D12.6,' = SALT WATER DENSITY IN KG/M3'/
      *1X,D12.6,' = INNER CROSS SECTIONAL AREA IN M2'/
      *1X,D12.6,' = INNER FLUID SPEED IN M/S'/
      *1X,D12.6,' = INNER FLUID OVERPRESSURE IN N/M2'/
      *1X,D12.6,' = APPROXIMATE TOTAL EFFECTIVE WEIGHT IN WATER IN N'/
      *1X,D12.6,' = MAXIMUM STATIC TENSION IN N'/
      *1X,D12.6,' = MEAN CURRENT VELOCITY IN THE X DIRECTION IN M/S'/
      *1X,D12.6,' = X COORDINATE AT TOP IN M'/
      *1X,D12.6,' = Y COORDINATE AT TOP IN M')       RCL05810
C      EVALUATE DIMENSIONAL NATURAL FREQUENCY        RCL05820
      SIGMAD = Y(N,1)*DSQRT(TMAXAV/TLEN)             RCL05830
      WRITE(9,1002) NSEG                             RCL05840
1002 FORMAT(//' DATA PER RISER SEGMENT FOR NSEG =RCL05850
      * ',I3,' SEGMENTS'/' DIMENSIONAL QUANTITY IRCL05990
      * E S I N T H E S . I . S Y S T E M '/        RCL06000
      *' RLENG      DXI      PXIETA     AO      WEIGHT    MASS   RCL06010
      *' TMASS      AMAXI     AMAETA     AMAZI    TMAXI    TMAZI  RCL06020
      *')
      TL = TOMAX/TLEN                               RCL06030
      DO 1004 I=1,NSEG                             RCL06040
      WRITE(9,1003) RLENG(I)*TLEN,DXI(I),PXIETA(I),AO(I),WEIGHT(I)*TL,RMRCL06060
      *ASS(I),RMASST(I),AMAXI(I),AMAETA(I),AMAZI(I),TMAXI(I),TMAZI(I) RCL06070
1004 CONTINUE                                         RCL06080
1003 FORMAT(12(1X,D10.4)/)                         RCL06090
C
      WRITE(9,10022)
10022 FORMAT(' EA      EIETA     EIETAS     EIXI      EIXIS    RCL06100
      * GIP      GIPS      DETA       JZI       AJZI      TJZI')  RCL06110
      *')
      DO 10023 I=1,NSEG                           RCL06120
      WRITE(9,10024) EA(I),EIETA(I),EIETAS(I),EIXI(I),EIXIS(I),GIP(I), RCL06130
      *GIPS(I),DETA(I),JZI(I),AJZI(I),TJZI(I)        RCL06140
      *')
      WRITE(9,10024) EA(I),EIETA(I),EIETAS(I),EIXI(I),EIXIS(I),GIP(I), RCL06150
      *GIPS(I),DETA(I),JZI(I),AJZI(I),TJZI(I)        RCL06160

```

```

10023 CONTINUE                                         RCL06170
10024 FORMAT(11(1X,D10.4)/)                           RCL06180
C
      WRITE(9,761) MODE,SIGMAD                         RCL06190
761 FORMAT(' ****'                                     RCL06200
      *' M O D E N U M B E R = ',I2/'   N A T U R A L F R E Q U E N C Y = ',D10.4,' RAD/S'/' ****'*)          RCL06210
      *C Y = ',D10.4,' RAD/S'/' ****'*)                  RCL06220
      *****)                                           RCL06230
C
      WRITE(9,1009) NP                                 RCL06240
1009  FORMAT(/' ORTHONORMALIZED NONDIMENSIONAL RESULTS AT NP = ',RCL06250
      *I3,' POINTS')                                RCL06260
      *'     S      TENSION      QXI      OMEGAETA      PHI      P
      *'     Q      SIGMA')                            RCL06270
      DO 1010 I=1,NP                               RCL06280
      WRITE(9,1011) X(I),(Y(J,I),J=1,7)             RCL06290
1010  CONTINUE                                     RCL06300
1011  FORMAT(8(1X,D10.4))                          RCL06310
C
      NORMALIZE ESTIMATED ERROR BY COMPONENTS.        RCL06320
      DO 189 I=1,N-1                               RCL06330
      ABT(I)=ABT(I)/A                             RCL06340
189   CONTINUE                                     RCL06350
      WRITE(9,10119) (ABT(I),I=1,7)                RCL06360
10119 FORMAT(' MAXIMUM ESTIMATED ERROR BY COMPONENTS'/11X,7(1X,D10.4)) RCL06370
C
C
      OUTPUT TO FILE CONNECTED TO DEVICE 11           RCL06380
C
      THIS CAN BE USED FOR PLOTS OR INPUT TO ANOTHER RUN OF RCLINDY1 RCL06390
C
C
      CALL STRUC(VLOCKXI,NP,NPI)                      RCL06400
      WRITE(11,36459) MODE,NP,SIGMAD,XTOP,YTOP         RCL06410
36459 FORMAT(1X,I2,1X,I3,3(1X,D10.4))             RCL06420
      DO 3666 I=1,NP                               RCL06430
      WRITE(11,3667) X(I),(Y(J,I),J=1,7),VLOCKXI(I) RCL06440
3666  CONTINUE                                     RCL06450
3667  FORMAT(9(1X,D12.6))                         RCL06460
C
      RETURN                                         RCL06470
      END                                            RCL06480
C
      SUBROUTINE FCN(X,EPS,Y,F,N)                    RCL06490
C
      THIS SUBROUTINE EVALUATES THE FUNCTIONS USED BY DO2RAF TO SOLVE RCL06500
      THE PROBLEM.                                    RCL06510
      IMPLICIT REAL*8(A-H,O-Z)                      RCL06520
      PARAMETER(MNP=151)                            RCL06530
      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL06540
      *ZETAM,TMAXAV,DXIM                           RCL06550
      COMMON/COEF1/EETA,EETAS,HZET,HX,EOM1,TO,QXIO,OMEGAO,CONS1,CONS2 RCL06560
      DIMENSION Y(N),F(N)                           RCL06570
C
      LOCATE MESH POINT TO EVALUATE COEFFICIENTS    RCL06580
      CALL COUNT(X)                                RCL06590
C
      F(1) = QXIO*Y(3) + OMEGAO*Y(2) + CONS2*Y(4) RCL06600
      F(1) = F(1) - HZET*Y(5)*Y(7)**2            RCL06610
C
      F(1) = QXIO*Y(3) + OMEGAO*Y(2) + CONS2*Y(4) RCL06620
      F(1) = F(1) - HZET*Y(5)*Y(7)**2            RCL06630
C
      F(1) = QXIO*Y(3) + OMEGAO*Y(2) + CONS2*Y(4) RCL06640
      F(1) = F(1) - HZET*Y(5)*Y(7)**2            RCL06650
C
      LOCATE MESH POINT TO EVALUATE COEFFICIENTS    RCL06660
      CALL COUNT(X)                                RCL06670
C
      F(1) = QXIO*Y(3) + OMEGAO*Y(2) + CONS2*Y(4) RCL06680
      F(1) = F(1) - HZET*Y(5)*Y(7)**2            RCL06690
C
      F(1) = QXIO*Y(3) + OMEGAO*Y(2) + CONS2*Y(4) RCL06700
      F(1) = F(1) - HZET*Y(5)*Y(7)**2            RCL06710
C
      LOCATE MESH POINT TO EVALUATE COEFFICIENTS    RCL06720

```

```

F(2) = -TO*Y(3) - OMEGA0*Y(1) RCL06730
F(2) = F(2) - CONS1*Y(4) RCL06740
F(2) = F(2) - HX*Y(6)*Y(7)**2 RCL06750
C RCL06760
F(3) = - Y(2)/EETA - EETAS*Y(3)/EETA RCL06770
C RCL06780
F(4) = Y(3) RCL06790
C RCL06800
F(5) = OMEGA0*Y(6) + EOMI*Y(1) RCL06810
C RCL06820
F(6) = Y(4) + Y(4)*EOMI*TO - OMEGA0*Y(5) RCL06830
C RCL06840
F(7) = 0.D0 RCL06850
RETURN RCL06860
END RCL06870
C RCL06880
SUBROUTINE G(EPS,YA,YB,BC,N) RCL06890
C BOUNDARY CONDITIONS USED BY D02RAF RCL06900
IMPLICIT REAL*8(A-H,O-Z) RCL06910
DIMENSION YA(N),YB(N),BC(N) RCL06920
COMMON/BOUNDA/BOUND,BOUNDP,BOUNDO RCL06930
C RCL06940
BC(1)=YA(4)
BC(2)=YA(5)
BC(3)=YA(6)
BC(4)=YA(3) - BOUND RCL06950
C RCL06960
BC(5)=YB(4)
BC(6)=YB(5) - BOUNDP + EPS*BOUNDP RCL06970
BC(7)=YB(6) - BOUNDQ + EPS*BOUNDQ RCL06980
C RCL06990
RETURN RCL07000
END RCL07010
C RCL07020
RCL07030
RCL07040
RCL07050
C RCL07060
SUBROUTINE JACOBF(X,EPS,Y,F,N) RCL07070
C THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE EQUATIONS TO USE IN RCL07080
C NEWTON'S ITERATION. RCL07090
IMPLICIT REAL*8(A-H,O-Z) RCL07100
PARAMETER(MNP=151) RCL07110
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL07120
*ZETAM,TMAXAV,DXIM RCL07130
COMMON/COEF1/EETA,EETAS,HZET,HX,EOMI,TO,QXIO,OMEGA0,CONS1,CONS2 RCL07140
DIMENSION Y(N),F(N,N) RCL07150
C LOCATE MESH POINT TO EVALUATE COEFFICIENTS RCL07160
CALL COUNT(X) RCL07170
C RCL07180
DO 817 I=1,N RCL07190
DO 817 M=1,N RCL07200
F(I,M) = 0.D0 RCL07210
817 CONTINUE RCL07220
C RCL07230
F(1,2) = OMEGA0
F(1,3) = QXIO
F(1,4) = CONS2
F(1,5) = -HZET*Y(7)**2
F(1,7) = -2.D0*Y(7)*HZET*Y(5) RCL07240
RCL07250
RCL07260
RCL07270
RCL07280

```

```

C
      F(2,1) = -OMEGAO          RCL07290
      F(2,3) = -TO              RCL07300
      F(2,4) = -CONS1           RCL07310
      F(2,6) = -HX*Y(7)**2       RCL07320
      F(2,7) = -2.D0*Y(7)*Y(6)*HX RCL07330
C
      F(3,2) = -1.D0/EETA        RCL07340
      F(3,3) = -EETAS/EETA       RCL07350
C
      F(4,3) = 1.D0              RCL07360
C
      F(5,1) = EOMI              RCL07370
      F(5,6) = OMEGAO            RCL07380
C
      F(6,4) = 1.D0 + EOMI*T0     RCL07390
      F(6,5) = -OMEGAO           RCL07400
      RETURN                      RCL07410
      END                         RCL07420
C
      SUBROUTINE JACOBG(EPS,YA,YB,AJ,BJ,N)      RCL07430
C
      THIS SUBROUTINE EVALUATES THE JACOBIAN OF THE BOUNDARY CONDITIONS. RCL07440
      IMPLICIT REAL*8(A-H,O-Z)                    RCL07450
      DIMENSION YA(N),YB(N),AJ(N,N),BJ(N,N)      RCL07460
      DO 876 K=1,N                                RCL07470
         DO 876 I=1,N                            RCL07480
            AJ(K,I) = 0.D0                         RCL07490
            BJ(K,I) = 0.D0                         RCL07500
876 CONTINUE
      C
         AJ(1,4) = 1.D0                         RCL07510
         AJ(2,5) = 1.D0                         RCL07520
         AJ(3,6) = 1.D0                         RCL07530
         AJ(4,3) = 1.D0                         RCL07540
      C
         BJ(5,4) = 1.D0                         RCL07550
         BJ(6,5) = 1.D0                         RCL07560
         BJ(7,6) = 1.D0                         RCL07570
         RETURN
         END
      C
      SUBROUTINE JACEPS(X,EPS,Y,F,N)             RCL07580
      C
      THIS SUBROUTINE EVALUATES THE DERIVATIVES OF THE FUNCTIONS WITH    RCL07590
      RESPECT TO THE CONTINUATION PARAMETER EPS.                         RCL07600
      IMPLICIT REAL*8(A-H,O-Z)                    RCL07610
      PARAMETER(NA=4,MNP=151)                   RCL07620
      COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI      RCL07630
      COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL07640
      *ZETAM,TMAXAV,DXIM                         RCL07650
      COMMON/COEF1/EETA,EETAS,HZET,HX,EOMI,TO,QXIO,OMEGAO,CONS1,CONS2   RCL07660
      DIMENSION Y(N),F(N)
      C
         F(1) = 0.D0                           RCL07670
      C
         F(2) = 0.D0                           RCL07680
      C

```

```

F(3) = 0.D0          RCL07850
F(4) = 0.D0          RCL07860
F(5) = 0.D0          RCL07870
C
F(6) = 0.D0          RCL07880
F(7) = 0.D0          RCL07890
RETURN              RCL07900
END                 RCL07910
RCL07920
C
SUBROUTINE JACGEP(EPS,YA,YB,BCEP,N)          RCL07930
C THIS SUBROUTINE EVALUATES THE DERIVATIVES OF THE BOUNDARY      RCL07940
C CONDITIONS WITH RESPECT TO THE CONTINUATION PARAMETER EPS.      RCL07950
IMPLICIT REAL*8(A-H,O-Z)                      RCL07960
COMMON/BOUNDA/BOUND,BOUNDP,BOUNDO             RCL07970
DIMENSION YA(N),YB(N),BCEP(N)                RCL07980
DO 871 K=1,N                                     RCL07990
    BCEP(K) = 0.D0                                RCL08000
871 CONTINUE                                     RCL08010
    BCEP(5) = BOUNDP                            RCL08020
    BCEP(6) = BOUNDO                            RCL08030
RETURN                                         RCL08040
END                                            RCL08050
RCL08060
C
SUBROUTINE INTERP(NP)                         RCL08070
C THIS SUBROUTINE INTERPOLATES THE RISER CHARACTERISTICS AND THE      RCL08080
C STATIC SOLUTION TO THE NEW NUMBER OF POINTS NP.                  RCL08090
C ASSUMPTION: NP .GE. NPI                                     RCL08100
IMPLICIT REAL*8(A-H,O-Z)                      RCL08110
PARAMETER(MNP=151,NA=4,N=7)                    RCL08120
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL08140
*ZETAM,TMAXAV,DXIM                               RCL08150
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI      RCL08160
COMMON/STAT2/CONST1(MNP),CONST2(MNP)            RCL08170
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE       RCL08180
DIMENSION HELP(MNP)                           RCL08190
C
INTERPOLATE STRUCTURAL DATA TO THE NEW NUMBER OF POINTS          RCL08200
CALL STRUC(EPSETA,NP,NPI)                      RCL08210
CALL STRUC(EPSETS,NP,NPI)                      RCL08220
CALL STRUC(EOM,NP,NPI)                        RCL08230
CALL STRUC(HXI,NP,NPI)                        RCL08240
CALL STRUC(HZETA,NP,NPI)                      RCL08250
CALL STRUC(CONST1,NP,NPI)                      RCL08260
CALL STRUC(CONST2,NP,NPI)                      RCL08270
CALL STRUC(VLOCKXI,NP,NPI)                     RCL08280
C
INTERPOLATE STATIC SOLUTION TO THE NEW NUMBER OF POINTS*        RCL08290
DO 459 K=1,3                                     RCL08300
    DO 458 I=1,NPI
        HELP(I) = STATIC(K,I)                   RCL08310
458 CONTINUE                                     RCL08320
CALL STRUC(HELP,NP,NPI)                        RCL08330
DO 457 I=1,NP                                    RCL08340
    STATIC(K,I) = HELP(I)                      RCL08350
457 CONTINUE                                     RCL08360
459 CONTINUE                                     RCL08370
DO 339 I=1,NP*                                RCL08380
    XI(I) = X(I)                             RCL08390
339 CONTINUE                                     RCL08400

```

```

339 CONTINUE                               RCL08410
    RETURN                                 RCL08420
    END                                    RCL08430
C
C      SUBROUTINE STRUC(ARRAY,NP,NPOLD)      RCL08440
C      THIS SUBROUTINE INTERPOLATES A SERIES OF DIVISION POINTS TO A NEW RCL08450
C      SERIES OF DIVISION POINTS.            RCL08460
IMPLICIT REAL*8(A-H,O-Z)                  RCL08470
PARAMETER(MNP=151,NA=4,N=7)                RCL08480
COMMON/SOLUT/X(MNP),Y(N,MNP),ABT(N),MODE   RCL08490
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI   RCL08500
DIMENSION ARRAY(MNP),HELP(MNP)             RCL08510
C
C      HELP(1) = ARRAY(1)                    RCL08520
C      HELP(NP) = ARRAY(NPOLD)              RCL08530
DO 82 I=2,NPOLD                           RCL08540
DO 81 K=2,NP-1                            RCL08550
IF ((X(K).GT.XI(I-1)).AND.(X(K).LT.XI(I))) THEN   RCL08560
    CONV = (X(K)-XI(I-1))*(ARRAY(I)-ARRAY(I-1))/(XI(I)-XI(I-1))   RCL08570
    HELP(K) = ARRAY(I-1) + CONV           RCL08580
ELSE IF (X(K).EQ.XI(I)) THEN             RCL08590
    HELP(K) = ARRAY(I)                 RCL08600
END IF                                     RCL08610
81  CONTINUE                                RCL08620
82  CONTINUE                                RCL08630
DO 85 K=1,NP                            RCL08640
    ARRAY(K) = HELP(K)                  RCL08650
85  CONTINUE                                RCL08660
RETURN                                    RCL08670
END                                      RCL08680
C
C      SUBROUTINE COUNT(X)                  RCL08690
C      THIS SUBROUTINE EVALUATES THE RISER CHARACTERISTICS AS WELL AS THE RCL0870
C      STATIC SOLUTION AT A POINT X.        RCL08710
C      IT RETURNS THEIR VALUE IN COMMON BLOCK COEF1.   RCL08720
IMPLICIT REAL*8(A-H,O-Z)                  RCL08730
PARAMETER(MNP=151,NA=4)                  RCL08740
COMMON/COEF/EPSETA(MNP),EPSETS(MNP),HZETA(MNP),HXI(MNP),EOM(MNP),HRCL08750
*ZETAM,TMAXAV,DXIM                      RCL08760
COMMON/STAT/XI(MNP),STATIC(NA,MNP),VLOCKXI(MNP),TOMAX,VM,NPI   RCL08770
COMMON/COEF1/EETA,EETAS,HZET,HX,EOMI,TO,QXIO,OMEGAO,CONS1,CONS2   RCL08780
COMMON/STAT2/CONST1(MNP),CONST2(MNP)     RCL08790
COMMON/COUN/ICOUNT                        RCL08800
M = ICOUNT                                RCL08810
IF (X.EQ.XI(M)) THEN                     RCL08820
    EETA = EPSETA(M)                   RCL08830
    EETAS = EPSETS(M)                 RCL08840
    HX = HXI(M)                      RCL08850
    HZET = HZETA(M)                  RCL08860
    EOMI = EOM(M)                    RCL08870
    TO = STATIC(1,M)                 RCL08880
    QXIO = STATIC(2,M)               RCL08890
    OMEGAO = STATIC(3,M)             RCL08900
    CONS1 = CONST1(M)                RCL08910
    CONS2 = CONST2(M)                RCL08920
    ICOUNT = ICOUNT + 1              RCL08930
RCL08940
RCL08950
RCL08960

```

```

IF(X.EQ.1.DO) ICOUNT = 1 RCL08970
ELSE IF ((X.LT.XI(M)).AND.(X.GT.XI(M-1))) THEN RCL08980
  DIFF = DABS(XI(M)-X) RCL08990
C THIS IF STATEMENT IS TO ACCOUNT FOR A COMPUTER INACCURACY IN RCL09000
C REPRESENTING AND EQUATING REAL NUMBERS. RCL09010
  IF (DIFF.LT.1.D-7) THEN RCL09020
    ICOUNT = ICOUNT + 1 RCL09030
    IF (XI(M).EQ.1.DO) ICOUNT = 1 RCL09040
  END IF RCL09050
  DX = (X-XI(M-1))/(XI(M)-XI(M-1)) RCL09060
  EETA = EPSETA(M-1) + (EPSETA(M)-EPSETA(M-1))*DX RCL09070
  EETAS = EPSETS(M-1) + (EPSETS(M)-EPSETS(M-1))*DX RCL09080
  HX = HXI(M-1) + (HXI(M)-HXI(M-1))*DX RCL09090
  HZET = HZETA(M-1) + (HZETA(M)-HZETA(M-1))*DX RCL09100
  EOMI = EOM(M-1) + (EOM(M)-EOM(M-1))*DX RCL09110
  TO = STATIC(1,M-1) + (STATIC(1,M)-STATIC(1,M-1))*DX RCL09120
  QXIO = STATIC(2,M-1) + (STATIC(2,M)-STATIC(2,M-1))*DX RCL09130
  OMEGA0 = STATIC(3,M-1) + (STATIC(3,M)-STATIC(3,M-1))*DX RCL09140
  CONST1 = CONST1(M-1) + (CONST1(M)-CONST1(M-1))*DX RCL09150
  CONST2 = CONST2(M-1) + (CONST2(M)-CONST2(M-1))*DX RCL09160
  ELSE RCL09170
C
    WRITE(6,*) ' ERROR OCCURED IN COUNTER, NP =',M RCL09180
  END IF RCL09190
  RETURN RCL09200
END RCL09210

```

RCL09220

Appendix A

Programs ORTHOG and ORTHOG1

1. Input-Output Manual for Program ORTHOG

This program is written in FORTRAN 77 and allows the user to check the orthogonality condition for the out-of-plane eigenvectors. The orthogonality condition is derived in [4].

Before executing the program the user must make sure that

- o Devices five and six correspond to input-output from the terminal
- o Device four must be associated with an input data file containing an out-of-plane eigensolution resulting from an execution of Program RCLINDYN or RCLINDY3. The input file has a logical record length of 117 characters.
- o Device three must be associated with an input data file containing another (or the same) out-of-plane eigensolution resulting from an execution of Program RCLINDYN or RCLINDY3.

This input file also has a logical record length of 117 characters.

Input Variables

At the beginning of the program the user is asked to input the total mass per unit length in the \vec{n} direction, m_T^n and the total mass per unit length in the $\vec{\xi}$ direction, m_T^ξ . These values should be given in kg/m. Then, the user is asked to input the riser length in meters and the total polar moment of mass inertia per unit length $J_T^{\xi\xi}$ in kg·m. For simplicity, the above values are assumed to be constant along the length of the riser for the purposes of this program. The program then evaluates and prints the value of the orthogonality condition.

2. Sample Run, Input and Output

Interactive Session

INPUT

Device 5 = TERMINAL

Device 4 = JOA01A DATA A*

Device 3 = JOA02A DATA A

OUTPUT

Device 6 = TERMINAL

* File included in Chapter III.

```

R; T=0,01/0,01 15.55.06
or thos Joao1a Joao2a
FI 4 DISK JOAO1A DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
FI 3 DISK JOAO2A DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
GLOBAL TEXTLIB VFORTLIB CMSLIB
LOAD ORTHOG ( START
EXECUTION BEGINS
INPUT TOTAL MASS PER UNIT LENGTH IN ETA DIRECTION AND
TOTAL MASS PER UNIT LENGTH IN XI DIRECTION IN KG/M
? 100.3, 132.4
INPUT RISER LENGTH IN M AND
TOTAL POLAR MOMENT OF MASS INERTIA PER UNIT LENGTH IN KG*M
? 88.392,0.5713
ORTHOGONAL VALUE = - .2452500 .03
R; T=0,43/0,50 15:55:43
cp spool console stop close
%
```

289

JOAO2A DATA A

2 96 0.64850+00

0.0000000+00 - 1036320+02 0.1063270+02 0.281375D+03 0.0000000+00 0.0000000+00 0.738635D+01 0.1030000+01
 0.493318D-03 - 102797D+02 0.104803D+02 0.289141D+03 0.140771D+00 0.243418D-02 - 347224D-04 0.738635D+01 0.103031D+01
 0.986636D-03 - 101901D+02 0.103305D+02 0.295937D+03 0.285122D+00 0.59029D-03 - 139773D-03 0.738635D+01 0.103062D+01
 0.172564D-02 - 100453D+02 0.101120D+02 0.304448D+03 0.506556D+00 - 145766D-01 0.432300D-03 0.738635D+01 0.103108D+01
 0.246464D-02 - 988894D+01 0.990234D+01 0.311104D+03 0.732656D+00 - 389026D-01 - 890190D-03 0.738635D+01 0.103154D+01
 0.320364D-02 - 972230D+01 0.970282D+01 0.316090D+03 0.961739D+00 - 723979D-01 - 151627D-02 0.738635D+01 0.103199D+01
 0.394265D-02 - 954692D+01 0.95430D+01 0.319582D+03 0.119244D+01 - 113863D+00 - 231225D-02 0.738635D+01 0.103244D+01
 0.476160D-02 - 934400D+01 0.931883D+01 0.321909D+03 0.144857D+01 - 167750D+00 - 339268D-02 0.738635D+01 0.103293D+01
 0.558056D-02 - 913375D+01 0.913781D+01 0.322806D+03 0.170400D+01 - 228546D+00 - 468458D-02 0.738635D+01 0.103342D+01
 0.639951D-02 - 891778D+01 0.891712D+01 0.322457D+03 0.195769D+01 - 294902D+00 - 618396D-02 0.738635D+01 0.103389D+01
 0.721847D-02 - 869764D+01 0.861898D+01 0.321032D+03 0.220886D+01 - 365552D+00 - 789007D-02 0.738635D+01 0.103436D+01
 0.803742D-02 - 847461D+01 0.868047D+01 0.318679D+03 0.245685D+01 - 439383D+00 - 980056D-02 0.738635D+01 0.103481D+01
 0.885637D-02 - 824986D+01 0.855518D+01 0.315537D+03 0.270119D+01 - 515381D+00 - 119126D-01 0.738635D+01 0.103521D+01
 0.999838D-02 - 793578D+01 0.840099D+01 0.310079D+03 0.303509D+01 - 623413D+00 - 151881D-01 0.738635D+01 0.103588D+01
 0.111404D-01 - 762280D+01 0.826924D+01 0.303590D+03 0.336052D+01 - 623413D+00 - 173194BD+00 - 188400D-01 0.738635D+01 0.103656D+01
 0.122824D-01 - 731298D+01 0.815804D+01 0.2965294D+03 0.367702D+01 - 839434D+00 - 228585D-01 0.738635D+01 0.103708D+01
 0.134244D-01 - 700788D+01 0.806528D+01 0.2888394D+03 0.398431D+01 - 944292D+00 - 272331D-01 0.738635D+01 0.103766D+01
 0.157084D-01 - 641715D+01 0.792747D+01 0.271397D+03 0.457105D+01 - 114211D-01 - 370033D-01 0.738635D+01 0.103878D+01
 0.1788993D-01 - 587910D+01 0.784483D+01 0.254539D+03 0.509886D+01 - 134135D+01 - 475962D-01 0.738635D+01 0.103981D+01
 0.2006901D-01 - 537025D+01 0.762296D+01 0.237072D+03 0.559297D+01 - 146710D+00 - 593081D-01 0.738635D+01 0.104080D+01
 0.222809D-01 - 489711D+01 0.778500D+01 0.220109D+03 0.605435D+01 - 159576D+01 - 720665D-01 0.738635D+01 0.104177D+01
 0.244717D-01 - 445522D+01 0.779261D+01 0.203702D+03 0.648421D+01 - 171286D+01 - 858013D-01 0.738635D+01 0.104272D+01
 0.280844D-01 - 380379D+01 0.783926D+01 0.178813D+03 0.711706D+01 - 185819D+01 - 1099210D+00 0.738635D+01 0.104421D+01
 0.315651D-01 - 323077D+01 0.792556D+01 0.154315D+03 0.767395D+01 - 205173D+01 - 273890D+00 0.738635D+01 0.105206D+01
 0.333384D-01 - 297334D+01 0.785179D+01 0.428143D+02 0.972776D+01 - 198367D+01 - 341846D+00 0.738635D+01 0.104567D+01
 0.351118D-01 - 273836D+01 0.798195D+01 0.142073D+03 0.792373D+01 - 199820D+01 - 1499810D+00 0.738635D+01 0.104639D+01
 0.413491D-01 - 202455D+01 0.804264D+01 0.130535D+03 0.815481D+01 - 202649D+01 - 164237D+00 0.738635D+01 0.104711D+01
 0.475865D-01 - 148635D+01 0.853846D+01 0.672562D+02 0.933421D+01 - 207665D+01 - 217224D+00 0.738635D+01 0.104959D+01
 0.547166D-01 - 102879D+01 0.835160D+01 0.4908010-01 0.920620D+00 0.234464D+02 0.9970510-01 - 198367D+01 - 341846D+00 0.738635D+01 0.105206D+01
 0.618467D-01 - 697615D+00 0.977276D+01 0.25142D+00 0.706995D+01 0.101218D+02 - 188517D+01 - 412072D+00 0.738635D+01 0.105770D+01
 0.866637D-01 - 128955D+01 0.827877D+01 0.951616D+02 0.883546D+01 - 205173D+01 - 161253D+01 0.738635D+01 0.106409D+01
 0.954915D-01 - 527751D-01 0.113366D+01 0.120046D+02 - 262252D+02 0.984330D+01 - 126235D+01 - 661629D+00 0.738635D+01 0.106767D+01
 0.114743D+00 - 235602D-01 0.1492659D+01 0.120986D+02 - 396194D+02 0.922968D+01 - 853752D+00 - 174932D+00 0.738635D+01 0.107126D+01
 0.1355160+00 0.4908010-01 0.120838D+02 - 484120D+02 0.922968D+01 - 140999D+01 - 9322918D+00 0.738635D+01 0.107126D+01
 0.157726D+00 0.695142D+01 0.101173D+02 - 726163D+02 0.832914D+01 - 409263D+00 - 111529D+01 0.738635D+01 0.107926D+01
 0.181288D+00 0.924205D+01 0.257431D+02 0.120375D+02 - 627461D+02 0.576485D+01 - 530085D-01 - 128745D+01 0.738635D+01 0.108808D+01
 0.206107D+00 0.11829659D+00 0.120046D+02 - 680285D+02 0.411107D+01 0.521929D+01 - 143987D+00 0.738635D+01 0.109775D+01
 0.232087D+00 0.3238979D+00 0.119646D+02 - 597513D-02 - 396929D+01 0.225377D+01 - 156250D+01 0.738635D+01 0.110829D+01
 0.259123D+00 0.182949D+00 0.366735D+00 0.116352D+02 - 714650D+02 0.226096D+01 - 142548D+01 0.738635D+01 0.111973D+01
 0.287110D+00 0.219343D+00 0.115938D+02 - 711425D+02 0.244089D+00 0.178410D+01 - 164534D+01 0.738635D+01 0.113208D+01
 0.315938D+00 0.417659D+00 0.115680D+02 - 113632D+02 - 104259D+02 0.136316D+01 - 164979D+01 0.738635D+01 0.122589D+01
 0.345492D+00 0.295842D+00 0.117322D+02 0.493345D+02 - 396929D+01 0.207536D+01 - 165656D+01 0.738635D+01 0.124453D+01
 0.375655D+00 0.3238979D+00 0.116877D+02 - 597513D-02 - 573709D+02 0.225377D+01 - 157252D+01 0.738635D+01 0.126375D+01
 0.406309D+00 0.182949D+00 0.366735D+00 0.116352D+02 - 719262D+01 0.216022D+01 - 121775D+01 0.738635D+01 0.120799D+01
 0.437333D+00 0.395514D+00 0.115938D+02 - 711425D+02 0.244089D+00 0.178410D+01 - 167919D+01 0.738635D+01 0.130307D+01
 0.468605D+00 0.417659D+00 0.115680D+02 - 113632D+02 - 10935D+02 0.136316D+01 - 164979D+01 0.738635D+01 0.132262D+01
 0.500909D+00 0.432237D+00 0.115615D+02 - 115615D+02 0.242767D+01 - 112677D+02 - 211887D+01 0.738635D+01 0.134172D+01
 0.531395D+00 0.438919D+00 0.115759D+02 0.156939D+02 - 109402D+02 - 109402D+02 - 131320D+00 0.738635D+01 0.134530D+01
 0.562667D+00 0.437097D+00 0.116110D+02 - 101323D+02 - 101323D+02 - 131320D+00 0.738635D+01 0.136011D+01
 0.593691D+00 0.425831D+00 0.116646D+02 - 104259D+02 0.394508D+02 - 893302D+01 0.738635D+01 0.137570D+01
 0.624345D+00 0.404903D+00 0.117322D+02 0.493345D+02 - 729545D+01 0.101103D+01 0.126106D+01 0.134172D+01
 0.654508D+00 0.375331D+00 0.118083D+02 0.573709D+02 - 543186D+01 0.125915D+01 0.128514D+01 0.137570D+01
 0.684062D+00 0.339204D+00 0.118874D+02 0.631725D+02 - 340516D+01 0.145110D+01 0.128827D+01 0.137757D+01
 0.71289QD+00 0.299266D+00 0.119641D+02 0.664702D+02 - 131964D+01 0.158168D+01 0.128827D+01 0.139398D+01
 0.744282D+00 0.164979D+01 0.164979D+01 0.724282D+01 0.164979D+01 0.164979D+01 0.140923D+01

0.740877D+00 0.258339D+00 0.120346D+02 0.671906D+02 0.727812D+00 -241253D+01 0.165807D+01 0.738635D+01 0.142332D+01
 0.767913D+00 0.218840D+00 0.120963D+02 0.654763D+02 0.265500D+01 -198665D+01 0.161234D+01 0.738635D+01 0.143624D+01
 0.793893D+00 0.182504D+00 0.121486D+02 0.616473D+01 0.439895D+01 -148709D+01 0.152071D+01 0.738635D+01 0.144804D+01
 0.818712D+00 0.150391D+00 0.121896D+02 0.561487D+02 0.591174D+01 -949040D+00 0.139269D+01 0.738635D+01 0.145876D+01
 0.842274D+00 0.122785D+00 0.122220D+02 0.494448D+02 0.7188888D+01 -400663D+00 0.123828D+01 0.738635D+01 0.146847D+01
 0.864484D+00 0.90925D+01 0.122463D+02 0.419971D+02 0.820931D+01 0.136026D+00 0.106728D+01 0.738635D+01 0.146847D+01
 0.885257D+00 0.735839D+01 0.122636D+02 0.337727D+02 0.898483D+01 0.647302D+00 0.888698D+00 0.738635D+01 0.147722D+01
 0.904508D+00 0.311112D+02 0.149232D+02 0.215164D+02 0.9494459D+01 0.111367D+01 0.710824D+00 0.738635D+01 0.148508D+01
 0.913336D+00 -678272D+01 0.108293D+02 0.135545D+02 0.963600D+01 0.131779D+01 0.626382D+00 0.738635D+01 0.149209D+01
 0.922164D+00 -183918D+00 0.102566D+02 0.394919D+01 0.969846D+01 0.151213D+01 0.541039D+00 0.738635D+01 0.149519D+01
 0.938153D+00 -6117448D+00 0.933165D+01 -245370D+02 0.950693D+01 0.182586D+01 0.381502D+00 0.738635D+01 0.149830D+01
 0.945283D+00 -9457914D+01 -140563D+01 0.865224D+01 -429205D+02 0.925640D+01 0.194156D+01 0.320611D+00 0.738635D+01 0.150377D+01
 0.952414D+00 -195362D+01 0.838715D+01 -664750D+02 0.886145D+01 0.202746D+01 0.256011D+00 0.738635D+01 0.150616D+01
 0.958651D+00 -268542D+01 0.814560D+01 -1283336D+03 0.836681D+01 0.206193D+01 0.202285D+00 0.738635D+01 0.150854D+01
 0.964888D+00 -2932910D+01 0.808347D+01 -139700D+03 0.769563D+01 0.203251D+01 0.152194D+00 0.738635D+01 0.151062D+01
 0.968435D+00 -320561D+01 0.802593D+01 -151770D+03 0.722283D+01 0.200811D+01 0.138748D+00 0.738635D+01 0.151272D+01
 0.971982D+00 -379902D+01 0.793821D+01 -175924D+03 0.667648D+01 0.197452D+01 0.125720D+00 0.738635D+01 0.151393D+01
 0.975528D+00 -448209D+01 0.789217D+01 -2003382D+03 0.605747D+01 0.187757D+01 0.101070D+00 0.738635D+01 0.151517D+01
 0.977719D+00 -494736D+01 0.7878686D+01 -216426D+03 0.563803D+01 0.173616D+01 0.784927D+01 0.738635D+01 0.151645D+01
 0.979910D+00 -544689D+01 0.790655D+01 -232888D+03 0.518903D+01 0.162410D+01 0.656803D+01 0.738635D+01 0.151727D+01
 0.982401D+00 -59846D+01 0.795888D+01 -249372D+03 0.470976D+01 0.1491200D+01 0.538193D+01 0.738635D+01 0.151812D+01
 0.984429D+00 -654571D+01 0.805342D+01 -265356D+03 0.419988D+01 0.133681D+01 0.429752D+01 0.738635D+01 0.151900D+01
 0.986605D+00 -700679D+01 0.816382D+01 -277026D+03 0.377998D+01 0.116146D+01 0.332146D+01 0.738635D+01 0.151992D+01
 0.987718D+00 -748062D+01 0.831340D+01 -287392D+03 0.334190D+01 0.101112D+01 0.263799D+01 0.738635D+01 0.152067D+01
 0.988574D+00 -772081D+01 0.840500D+01 -291884D+03 0.311631D+01 0.851523D+00 0.202800D+01 0.738635D+01 0.152315D+01
 0.989431D+00 -796196D+01 0.850897D+01 -295778D+03 0.288660D+01 0.768988D+00 0.177157D+01 0.738635D+01 0.152458D+01
 0.990288D+00 -820295D+01 0.862620D+01 -298951D+03 0.265303D+01 0.685569D+00 0.149435D+01 0.738635D+01 0.152402D+01
 0.991144D+00 -8442255D+01 0.875745D+01 -301266D+03 0.241597D+01 0.601935D+00 0.125712D+01 0.738635D+01 0.152271D+01
 0.991963D+00 -866901D+01 0.8896669D+01 -302533D+03 0.218645D+01 0.518911D+00 0.104004D+01 0.738635D+01 0.152315D+01
 0.992782D+00 -889146D+01 0.904987D+01 -302716D+03 0.195472D+01 0.440986D+00 0.851570D+02 0.738635D+01 0.152541D+01
 0.9936000+00 -910807D+01 0.921717D+01 -301638D+03 0.172161D+01 0.365447D+00 0.681990D+02 0.738635D+01 0.152589D+01
 0.994419D+00 -931172D+01 0.939870D+01 -299107D+03 0.148786D+01 0.22605D+00 0.531536D+02 0.738635D+01 0.152633D+01
 0.995238D+00 -951714D+01 0.959476D+01 -294903D+03 0.125445D+01 0.164629D+00 0.400189D+02 0.738635D+01 0.152447D+01
 0.996057D+00 -970558D+01 0.980291D+01 -288792D+03 0.102288D+01 0.110575D+00 0.287891D+02 0.738635D+01 0.152493D+01
 0.996796D+00 -986408D+01 0.100014D+02 -281437D+03 0.816918D+00 0.194634D+00 0.426653D+02 0.738635D+01 0.152589D+01
 0.997535D+00 -100100D+02 0.102080D+02 -272122D+03 0.615344D+00 0.293426D+00 0.364512D+01 0.738635D+01 0.152633D+01
 0.998274D+00 -101418D+02 0.104207D+02 -2606400+03 0.419959D+00 0.129413D+01 0.354165D+03 0.738635D+01 0.152678D+01
 0.999013D+00 -102583D+02 0.106368D+02 -246778D+03 0.232868D+00 -176602D+03 0.113546D+03 0.738635D+01 0.152724D+01
 0.999506D+00 -103270D+02 0.107813D+02 -236107D+03 0.113649D+00 -271504D+02 0.280429D+04 0.738635D+01 0.152801D+01
 0.100000D+01 -103882D+02 0.109346D+02 -2242210+03 0.452180D-18 0.257594D-18 -171372D-17 0.738635D+01 0.152832D+01

3. Input-Output Manual for Program ORTHOG1

This program is written in FORTRAN 77 and allows the user to check the orthogonality condition for the in-plane eigenvectors. The orthogonality condition is derived in [4].

Before executing the program the user must make sure that

- o Devices five and six correspond to input-output from the terminal
- o Device four must be associated with an input data file containing an in-plane eigensolution resulting from an execution of Program RCLINDY1 or RCLINDY2. The input file has a logical record length of 117 characters.
- o Device three must be associated with an input data file containing another (or the same) in-plane eigensolution resulting from an execution of Program RCLINDY1 or RCLINDY2. This input file also has a logical record length of 117 characters.

Input Variables

At the beginning of the program the user is asked to input the total mass per unit length in the $\vec{\zeta}$ direction, m_T^ζ and the total mass per unit length in the $\vec{\xi}$ direction, m_T^ξ . These values should be given in kg·m. For simplicity, these values are assumed to be constant along the length of the riser for the purposes of this program. The program then evaluates and prints the value of the orthogonality condition.

4. Sample Run, Input and Output

Interactive Session

INPUT

Device 5 = TERMINAL

Device 4 = INPLA1A DATA A*

Device 3 = INPLA2A DATA A

OUTPUT

Device 6 = TERMINAL

* File included in Chapter V.

```

R; T=0.01/0.01 16:28:29
$orthog1 top1a1 impla1a impla2a
F1>4 DISK INPLA1A DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
F1 3 DISK INPLA2A DATA A ( RECFM FB LRECL 117 BLKSIZE 1170
GLOBAL TXLIB VFORTLIB CMSLIB
LOAD *ORTHOG1 ( START
EXECUTION BEGINS
INPUT TOTAL MASS PER UNIT LENGTH IN ZETA DIRECTION AND
TOTAL MASS PER UNIT LENGTH IN XI DIRECTION IN KG/M
49 93 132 4
ORTHOGONAL VALUE = -1920750.03
R; T=0.40/0.48 16:29:03
CP spool console stop close

```

295

INPLA2A DATA A

0. 90450B0D+00	0	379384D+01	- .319958D+01	- .755089D+02	- .946514D+01	0	.124517D-01	0	.953440D+00	0	.872960D+01	0	.149209D+01	
0. 922164D+00	0	356536D+01	- .467227D-01	- .583054D+02	- .106465D+02	0	.311546D-01	0	.775431D+00	0	.872560D+01	0	.149830D+01	
0. 938153D+00	0	333393D+01	- .967660D-01	- .392719D+02	- .114265D+02	0	.425697D-01	0	.598373D+00	0	.872960D+01	0	.150377D+01	
0. 945283D+00	0	323398D+01	- .143269D+00	- .280900D+02	- .116667D+02	0	.455599D-01	0	.515808D+00	0	.872960D+01	0	.150616D+01	
0. 952414D+00	0	312611D+01	- .225724D+00	- .131551D+02	- .118137D+02	0	.468657D-01	0	.431972D+00	0	.872960D+01	0	.150854D+01	
0. 958651D+00	0	302956D+01	- .353195D+00	0	.642572D+01	- .118347D+02	0	.460811D-01	0	.358325D+00	0	.872960D+01		
0. 964888D+00	0	293852D+01	- .579221D+01	- .371064D+02	- .116990D+02	0	.429576D-01	0	.285384D+00	0	.872960D+01	0	.151062D+01	
0. 971982D+00	0	289747D+01	- .775258D+00	0	.621301D+02	- .115223D+02	0	.399257D-01	0	.244677D+00	0	.872960D+01	0	.151272D+01
0. 975528D+00	0	288523D+01	- .104251D+01	- .931264D+02	- .112476D+02	0	.359145D-01	0	.204974D+00	0	.872960D+01	0	.151393D+01	
0. 977719D+00	0	292056D+01	- .1399973D+01	0	.132231D+03	- .108481D+02	0	.310189D-01	0	.166684D+00	0	.872960D+01		
0. 9799100+00	0	299234D+01	- .167499D+01	0	.162597D+03	- .105251D+02	0	.275988D-01	0	.143915D+00	0	.872960D+01		
0. 982101D+00	0	311793D+01	- .2001127D+01	0	.198860D+03	- .101291D+02	0	.239588D-01	0	.151727D+01	0	.872960D+01		
0. 984292D+00	0	332386D+01	- .238560D+01	0	.242086D+03	- .964650D+01	0	.242001D+00	0	.872960D+01	0	.151812D+01		
0. 986005D+00	0	356367D+01	- .283237D+01	0	.293449D+03	- .905937D+01	0	.200754D-01	0	.101086D+00	0	.872960D+01		
0. 987718D+00	0	3899945D+01	- .322488D+01	0	.340019D+03	- .851680D+01	0	.161474D-01	0	.813838D-01	0	.151900D+01		
0. 989431D+00	0	436225D+01	- .365060D+01	0	.392829D+03	- .788912D+01	0	.131068D-01	0	.669288D-01	0	.151992D+01		
0. 991144D+00	0	499021D+01	- .409797D+01	0	.452272D+03	- .716529D+01	0	.101837D-01	0	.534417D-01	0	.152067D+01		
0. 992782D+00	0	578628D+01	- .454363D+01	0	.518459D+03	- .633386D+01	0	.747037D-02	0	.410532D-01	0	.152146D+01		
0. 994419D+00	0	681514D+01	- .493090D+01	0	.587697D+03	- .542792D+01	0	.506659D-02	0	.872960D+01	0	.152228D+01		
0. 995238D+00	0	742900D+01	- .522073D+01	0	.661605D+03	- .314500D-01	0	.314500D-02	0	.299144D-01	0	.872960D+01		
0. 996057D+00	0	811423D+01	- .530529D+01	0	.699826D+03	- .40536D+01	0	.166846D-02	0	.205924D-01	0	.152315D+01		
0. 996796D+00	0	879493D+01	- .532996D+01	0	.738331D+03	- .384786D+01	0	.127556D-01	0	.872960D+01	0	.152402D+01		
0. 997535D+00	0	953438D+01	- .528681D+01	0	.772890D+03	- .325893D+01	0	.110981D-02	0	.944558D-02	0	.872960D+01		
0. 998274D+00	0	103293D+02	- .495193D+01	0	.806768D+03	- .270054D+01	0	.674862D-03	0	.658361D-02	0	.872960D+01		
0. 999013D+00	0	111737D+02	- .462565D+01	0	.839368D+03	- .211685D+01	0	.187966D-03	0	.440937D-02	0	.872960D+01		
0. 999506D+00	0	117606D+02	- .433622D+01	0	.869980D+03	- .877000D+00	0	.144026D-04	0	.264538D-02	0	.872960D+01		
0. 100000D+01	0	123613D+02	- .3983336D+01	0	.906355D+03	- .442988D+00	0	.236103D-05	0	.109254D-03	0	.872960D+01		
						- .841281D-18	0	.000000D+00	0	.984623D-19	0	.872960D+01		

5. Listing of Program

ORTHOG FORTRAN A

```

C      PROGRAM ORTHOG CHECKS ORTHOGONALITY OF EIGENVECTORS FOR THE      ORT00010
C      LINEAR DYNAMICS OF COMPLIANT RISERS (OUT OF PLANE DYNAMICS).      ORT00020
C      DOUBLE PRECISION IS USED THROUGHOUT.      ORT00030
C*****
C      COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY      ORT00040
C      ALL RIGHTS RESERVED.      ORT00050
C*****      ORT00060
C      PROGRAMMER GEORGE A. KRIEZIS      JUNE 5, 1985      M.I.T.      ORT00070
C*****      ORT00080
C*****      ORT00090
C
C      DEFINITION OF DEVICES:      ORT00100
C      DEVICE 4 : INPUT FROM FILE CONTAINING AN OUT OF PLANE      ORT00110
C                  DYNAMICS SOLUTION (LRECL=117)      ORT00120
C      DEVICE 3 : INPUT FROM FILE CONTAINING AN OUT OF PLANE      ORT00130
C                  DYNAMICS SOLUTION (LRECL=117)      ORT00140
C      IMPLICIT REAL*8(A-H,O-Z)      ORT00150
C      PARAMETER(MNP=151)      ORT00160
C      DIMENSION X1(MNP),X2(MNP),BETA1(MNP),BETA2(MNP),R1(MNP),R2(MNP)      ORT00170
C      DIMENSION A1(MNP)      ORT00180
C      WRITE(6,54)      ORT00190
C
54 FORMAT(' INPUT TOTAL MASS PER UNIT LENGTH IN ETA DIRECTION AND'/
* ' TOTAL MASS PER UNIT LENGTH IN XI DIRECTION IN KG/M')      ORT00200
      READ(5,*) TMAETA,TMAXI      ORT00210
      HETA = TMAETA/TMAXI      ORT00220
      WRITE(6,72)      ORT00230
C
72 FORMAT(' INPUT RISER LENGTH IN M AND'/' TOTAL POLAR MOMENT OF MASSORT00260
* INERTIA PER UNIT LENGTH IN KG*M')      ORT00270
      READ(5,*) TLEN,TJZI      ORT00280
      TLAMB2 = TLEN**2*TMAXI/TJZI      ORT00290
      READ(4,30) MODE1,NP1,SIGMA1      ORT00300
      DO 50 I=1,NP1      ORT00310
          READ(4,28) X1(I),BETA1(I),R1(I)
50 CONTINUE      ORT00320
30 FORMAT(1X,I2,1X,I3,1X,D10.4)      ORT00330
28 FORMAT(1X,D12.6,53X,D12.6,1X,D12.6,26X)      ORT00340
      READ(3,30) MODE2,NP2,SIGMA2      ORT00350
      DO 51 I=1,NP2      ORT00360
          READ(3,28) X2(I),BETA2(I),R2(I)
51 CONTINUE      ORT00370
C
      INTERPOLATE DATA TO THE SAME NUMBER OF POINTS (HIGHER NUMBER)      ORT00380
      IF (NP2.GT.NP1) THEN      ORT00390
          NP=NP2      ORT00400
          CALL STRUC(BETA1,X1,X2,NP2,NP1)      ORT00410
          CALL STRUC(R1,X1,X2,NP2,NP1)      ORT00420
          DO 66 I=1,NP2      ORT00430
              X1(I) = X2(I)      ORT00440
66 CONTINUE      ORT00450
      ELSE IF (NP1.GE.NP2) THEN      ORT00460
          CALL STRUC(BETA2,X2,X1,NP1,NP2)      ORT00470
          CALL STRUC(R2,X2,X1,NP1,NP2)      ORT00480
          NP = NP1      ORT00490
      END IF      ORT00500
C
      CALCULATE THE ORTHONORMALIZING CONSTANT      ORT00510
      DO 99 I=1,NP      ORT00520
          E = BETA1(I)*BETA2(I)/TLAMB2      ORT00530
          A1(I) = HETA*R1(I)*R2(I) + E      ORT00540
99 CONTINUE      ORT00550
      END      ORT00560

```

```

99 CONTINUE                               ORT00570
    A = 0.D0                                ORT00580
    DO 42 I=2,NP                            ORT00590
        DX2=(XI(I)-XI(I-1))*0.5D0          ORT00600
        A = A + (A1(I) + A1(I-1))*DX2      ORT00610
42 CONTINUE                               ORT00620
    WRITE(6,14) A                          ORT00630
14 FORMAT(' ORTHOGONAL VALUE = ',D12.6)   ORT00640
    STOP                                    ORT00650
    END                                     ORT00660
C                                         ORT00670
C     SUBROUTINE STRUC(ARRAY,XI,X,NP,NPOLD)  ORT00680
C     THIS SUBROUTINE INTERPOLATES A SERIES AT NPOLD DIVISION POINTS TO ORT00690
C     A NEW SERIES AT NP DIVISION POINTS.    ORT00700
C     IMPLICIT REAL*8(A-H,O-Z)              ORT00710
C     PARAMETER(MNP=151)                   ORT00720
C     DIMENSION ARRAY(MNP),HELP(MNP),XI(MNP),X(MNP) ORT00730
C     HELP(1) = ARRAY(1)                  ORT00740
C     HELP(NP) = ARRAY(NPOLD)            ORT00750
C     DO 82 I=2,NPOLD                  ORT00760
C         DO 81 K=2,NP-1                ORT00770
C             IF((X(K).GT.XI(I-1)).AND.(X(K).LT.XI(I))) THEN ORT00780
C                 CONV = (X(K)-XI(I-1))*(ARRAY(I)-ARRAY(I-1))/(XI(I)-XI(I-1)) ORT00790
C                 HELP(K) = ARRAY(I-1) + CONV           ORT00800
C             ELSE IF (X(K).EQ.XI(I)) THEN          ORT00810
C                 HELP(K) = ARRAY(I)               ORT00820
C             END IF                           ORT00830
C81  CONTINUE                               ORT00840
C82  CONTINUE                               ORT00850
C     DO 85 K=1,NP                         ORT00860
C         ARRAY(K) = HELP(K)               ORT00870
C85  CONTINUE                               ORT00880
C     RETURN                                 ORT00890
C     END                                   ORT00900

```

6. Listing of Program

ORTHOG1 FORTRAN A

```

C      PROGRAM ORTHOG1 CHECKS ORTHOGONALITY OF EIGENVECTORS FOR THE      ORT00010
C      LINEAR DYNAMICS OF COMPLIANT RISERS (IN-PLANE DYNAMICS).      ORT00020
C      DOUBLE PRECISION IS USED THROUGHOUT.      ORT00030
C*****ORT00040
C      COPYRIGHT, 1985, MASSACHUSETTS INSTITUTE OF TECHNOLOGY      ORT00050
C      ALL RIGHTS RESERVED.      ORT00060
C*****ORT00070
C      PROGRAMMER GEORGE A. KRIEZIS      JUNE 5, 1985      ORT00080
C*****ORT00090
C      DEFINITION OF DEVICES:      ORT00100
C      DEVICE 4 : INPUT FROM FILE CONTAINING AN IN-PLANE DYNAMICS      ORT00120
C                  SOLUTION (LRECL=117)      ORT00130
C      DEVICE 5 : INPUT FROM FILE CONTAINING AN IN-PLANE DYNAMICS      ORT00140
C                  SOLUTION (LRECL=117)      ORT00150
C      IMPLICIT REAL*8(A-H,O-Z)      ORT00160
C      PARAMETER(MNP=151)      ORT00170
C      DIMENSION X1(MNP),X2(MNP),BETA1(MNP),BETA2(MNP),R1(MNP),R2(MNP)      ORT00180
C      DIMENSION A1(MNP)
C      HXI = 1.D0
C      WRITE(6,45)
45   FORMAT(' INPUT TOTAL MASS PER UNIT LENGTH IN ZETA DIRECTION AND'//      ORT00220
*' TOTAL MASS PER UNIT LENGTH IN XI DIRECTION IN KG/M')      ORT00230
      READ(5,*) TZETA,TXI
      HZETA = TZETA/TXI
      READ(4,30) MODE1,NP1,SIGMA1,XTOP,YTOP
      DO 5C I=1,NP1
          READ(4,28) X1(I),BETA1(I),R1(I)
50   CONTINUE
30   FORMAT(1X,I2,1X,I3,1X,D10.4,1X,D10.4,1X,D10.4)      ORT00300
28   FORMAT(1X,D12.6,53X,D12.6,1X,D12.6,26X)      ORT00310
      READ(3,30) MODE2,NP2,SIGMA2,XTOP2,YTOP2
      DO 51 I=1,NP2
          READ(3,28) X2(I),BETA2(I),R2(I)
51   CONTINUE
C      INTERPOLATE DATA TO THE SAME NUMBER OF POINTS.
C      IF (NP2.GT.NP1) THEN
        NP=NP2
        CALL STRUC(BETA1,X1,X2,NP2,NP1)
        CALL STRUC(R1,X1,X2,NP2,NP1)
        DO 66 I=1,NP2
            X1(I) = X2(I)
66   CONTINUE
        ELSE IF (NP1.GE.NP2) THEN
            CALL STRUC(BETA2,X2,X1,NP1,NP2)
            CALL STRUC(R2,X2,X1,NP1,NP2)
            NP = NP1
        END IF
C      CALCULATE THE ORTHONORMALIZING CONSTANT
        DO 99 I=1,NP
            E = BETA1(I)*BETA2(I)*HZETA
            A1(I) = HXI*R1(I)*R2(I) + E
99   CONTINUE
        A = 0.D0
        DO 42 I=2,NP
            DX2=(X1(I)-X1(I-1))*0.5D0

```

```

A = A + (A1(I) + A1(I-1))*DX2
42 CONTINUE
    WRITE(6,14) A
14 FORMAT(' ORTHOGONAL VALUE = ',D12.6)
    STOP
    END

C
C          SUBROUTINE STRUC(ARRAY,XI,X,NP,NPOLD)
C THIS SUBROUTINE INTERPOLATES A SERIES AT NPOLD DIVISION POINTS TO
C A NEW SERIES AT NP DIVISION POINTS.
C IMPLICIT REAL*8(A-H,O-Z)
C PARAMETER(MNP=151)
C DIMENSION ARRAY(MNP),HELP(MNP),XI(MNP),X(MNP)
C HELP(1) = ARRAY(1)
C HELP(NP) = ARRAY(NPOLD)
DO 82 I=2,NPOLD
    DO 81 K=2,NP-1
        IF((X(K).GT.XI(I-1)).AND.(X(K).LT.XI(I))) THEN
            CONV = (X(K)-XI(I-1))*(ARRAY(I)-ARRAY(I-1))/(XI(I)-XI(I-1))
            HELP(K) = ARRAY(I-1) + CONV
        ELSE IF (X(K).EQ.XI(I)) THEN
            HELP(K) = ARRAY(I)
        END IF
81    CONTINUE
82    CONTINUE
    DO 85 K=1,NP
        ARRAY(K) = HELP(K)
85    CONTINUE
    RETURN
    END

```