AD-A055 767 AEROSPACE SYSTEMS INC BURLINGTON MASS F/G 20/4

ROTOR WAKE EFFECTS ON HUB/PYLON FLOW. VOLUME II. PROGRAM SHAPES--ETC(U)

MAY 78 P SOOHOO, R B NOLL, L MORINO, N D HAMM DAAJ02-75-C-0041

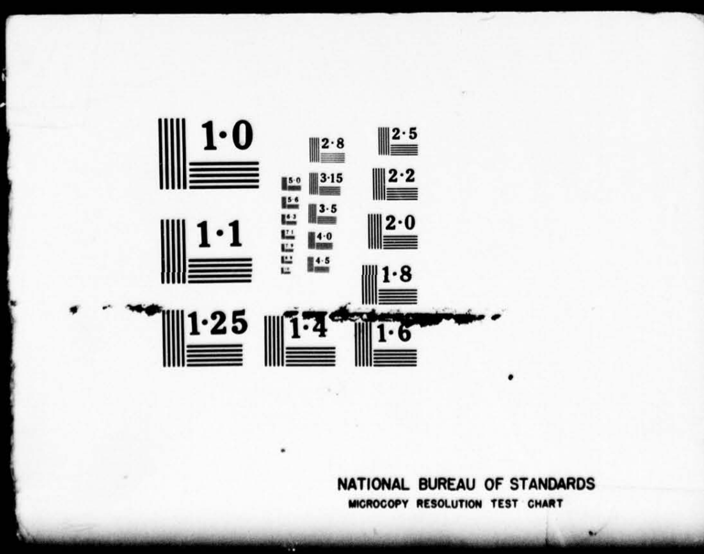
UNCLASSIFIED ASI-TR-76-38-VOL-2 USARTL-TR-78-1B NL

1 OF 2

ADA 055767

1 OF 2

ADA
055767



USARTL-TR-78-1B

ROTOR WAKE EFFECTS ON HUB/PYLON FLOW

Volume II, Program SHAPES User's Manual

Paul Soohoo
Richard B. Noll
Luigi Morino
Norman D. Hamm

Aerospace Systems, Inc.
Burlington, Mass. 01803

May 1978

Final Report for Period June 1975 - September 1977

Prepared for

APPLIED TECHNOLOGY LABORATORY
U. S. ARMY RESEARCH AND TECHNOLOGY LABORATORIES (AVRADCOM)
Fort Eustis, Va. 23604

APPLIED TECHNOLOGY LABORATORY POSITION STATEMENT

This report presents a method for determining aerodynamic characteristics of helicopter shapes under the influence of the main rotor wake. The method is condidered to be worthy of publication for dissemination of information and the stimulation of further related research. The reader is cautioned that this method does not predict flow separation as the title would imply, nor does the rotor wake fully impinge upon the body of the helicopter. The method is useful, however, as a design tool in determining rotor-fuselage aerodynamic interference.

Mr. F. A. Raitch of the Aeromechanics Technical Area served as project engineer for this effort.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: USARTL-TR-78-1B

2. GOVT ACCESSION NO.:

3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (and Subtitle): ROTOR WAKE EFFECTS ON HUB/PYLON FLOW. Volume II. Program SHAPES User's Manual.

5. TYPE OF REPORT & PERIOD COVERED: Final Report. June 1975 - September 1977

6. PERFORMING ORG. REPORT NUMBER: AST-TR-76-38, Volume II

7. AUTHOR(s):

Paul Soohoo
Richard B. Noll
Luigi Morino
Norman D. Ham

8. CONTRACT OR GRANT NUMBER(s): DAAJ02-75-C-0041

9. PERFORMING ORGANIZATION NAME AND ADDRESS:

Aerospace Systems, Inc.
Burlington, Massachusetts 01803

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62209A 1F262209AH76 00 087 EK

11. CONTROLLING OFFICE NAME AND ADDRESS: Applied Technology Laboratory, U.S. Army Research & Technology Labs (AVRADCOM), Fort Eustis, Virginia 23604

12. REPORT DATE: May 1978

13. NUMBER OF PAGES: 190

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Volume II of a two-volume report.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Helicopter
Hub/Pylon Flow
Induced Drag
Potential Aerodynamics
Rotor Wake
Rotors
Separation
Finite Element Aerodynamics
Aerodynamic Computer Program

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An investigation has been conducted to demonstrate the use of the Green's function method to study rotor wake effects on helicopter hub/pylon flow. This report consists of two volumes which document the theoretical formulation and the use of the digital computer program SHAPES (Subsonic Helicopter Aerodynamics Program with Effects of Separation).

Volume I presents the theoretical formulation and corresponding numerical procedure for the study of incompressible potential aerodynamics with separated flow. While the

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

389704 78 06 26 026

20. Abstract (Continued)

formulation is valid for fully unsteady aerodynamics, this report is mainly concerned with rotor aerodynamics. A potential flow aerodynamic program, SHAPES, with suitable rotor wake representation was developed to predict the separation characteristics of arbitrary three-dimensional helicopter configurations. In particular, the effect of the rotor blade wake, blade shank wake, and hub wake in the separation of the flow over a lifting helicopter in forward flight is analyzed.

The present method has potential application in the design of helicopters because it provides an analytical capability which can be used to develop low-drag profile as well as to explore problem areas. Extensive numerical results obtained from Program SHAPES demonstrated the flexibility and accuracy of the method. These results are in excellent agreement with existing data.

Volume II, The User's Manual, describes the structure and use of Program SHAPES. SHAPES is written in FORTRAN IV for operation on the CDC 6600 digital computer system at the Applied Technology Laboratory, U.S. Army Research and Technology Laboratories (AVRADCOM), Fort Eustis, VA. The User's Manual (Volume II) contains detailed information for preparing an input data deck and interpreting the computed data; a discussion of various subroutines, flow charts, common storage and definition of FORTRAN variables; sample cases to illustrate the program output; and a FORTRAN listing of the program.

PREFACE

This report, prepared by Aerospace Systems, Inc. (ASI), Burlington, Massachusetts, for the U.S. Army under Contract DAAJ02-75-C-0041, documents the results of research performed during the period June 1975 to September 1977. The study was sponsored by the Eustis Directorate, U.S. Army Air Mobility Research and Development Laboratory,* Fort Eustis, Virginia, with Mr. Frederick A. Raitch serving as Technical Monitor.

The effort was directed by Mr. John Zvara, President and Technical Director of ASI. Mr. Paul Soohoo served as Principal Technical Staff Member under the supervision of Mr. Richard B. Noll, Vice President of ASI. Dr. Luigi Morino, Director of Computational Continuum-Mechanics Program at Boston University, and Dr. Norman D. Ham, Director of the V/STOL Technology Laboratory at MIT, contributed to the study as Principal Consultants.

*Redesignated Applied Technology Laboratory, U.S. Army Research and Technology Laboratories (AVRADCOM), effective September 1, 1977.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1

INTRODUCTION

This volume describes the structure and use of a digital computer program, SHAPES, Subsonic Helicopter Aerodynamic Program with Effects of Separation, which was developed by Aerospace Systems, Inc. (ASI), to analyze the incompressible potential aerodynamics of helicopter configuration with rotor wakes and separation effects. Volume I of this report contains a complete description of the theoretical formulation and corresponding numerical procedure for the aerodynamic method for use in the computer program. It also includes extensive numerical results showing the flexibility and accuracy of the method as well as comparison with several existing results. Proper use of SHAPES is predicated on the assumption that the user is familiar with the techniques and limitations set forth in Volume I.

SHAPES is written almost entirely in FORTRAN IV for operation on the CDC 6600 digital computer system at the Applied Technology Laboratory, U.S. Army Research and Technology Laboratories (AVRADCOM), Fort Eustis, Virginia. The program was developed with a highly modular structure for ease of program checkout, to simplify the user's understanding of the program, and to facilitate any modifications that might be required for future applications.

Sections 2 and 3 contain programming details of SHAPES: functions of the various routines, flow charts, common storage, and definition of FORTRAN variables. The use of the program is presented in Section 4. Sample cases are presented in Section 5 to illustrate the output of Program SHAPES. Section 6 contains the FORTRAN listing of the program.

SECTION 2

COMPUTER PROGRAM SHAPES

Computer program SOUSSA (Steady Oscillatory Unsteady Subsonic and Supersonic Aerodynamics) discussed in Subsection 2.2 of Volume I has been modified to include rotor dynamics and separation from hub/pylon components. The revised program is called SHAPES, for Subsonic Helicopter Aerodynamic Program with Effects of Separation. Geometry preprocessor for single-rotor helicopter configurations is included in Program SHAPES.

In Program SHAPES the user need not be familiar with the aerodynamic portion of the program, unlike other sophisticated aerodynamic programs in which the choice for the combination of various aerodynamic functions (sources, doublets, vortices) is an art which requires considerable understanding of the method. Another advantage of SHAPES is that the paneling used for the aerodynamics is completely arbitrary and, therefore, may coincide with the one used for structural analysis. A typical finite element representation of a helicopter fuselage configuration is shown in Figure 1. The aerodynamic paneling can be accomplished manually by inputting the coordinates

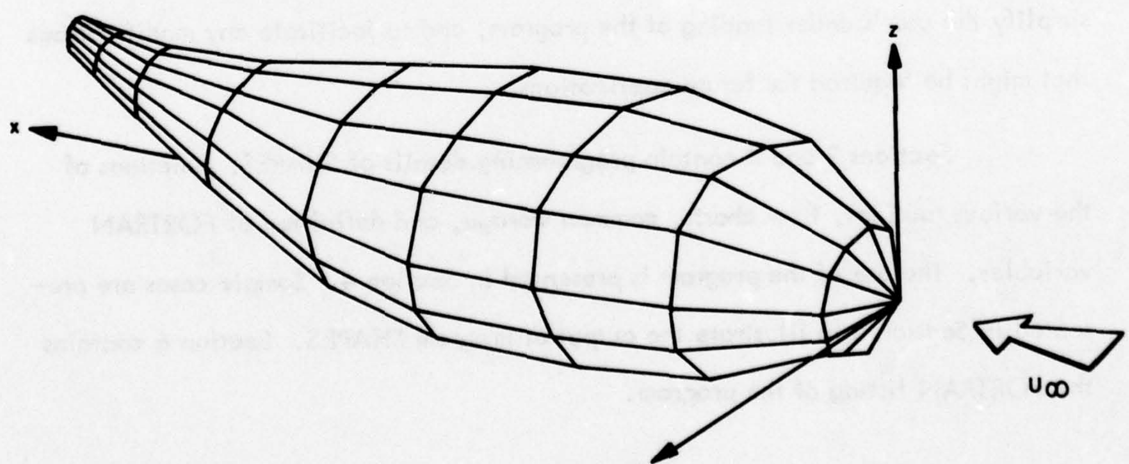


Figure 1. Finite Element Representation of Arbitrary Helicopter Fuselage.

of the corner points of all selected aerodynamic panels or by using a geometry preprocessor. A geometry preprocessor for conventional helicopter configurations (see Figure 2) is incorporated in SHAPES. The fuselage is assumed to have an elliptical center section with the nose and tail approximated by elliptical paraboloids. An ellipsoidal element is used to approximate the rotor hub and pylon sections. In addition, the rotor shaft and blade shank are represented by cylindrical sections. Given the helicopter configuration geometry, the preprocessor computes the corner point locations of required aerodynamic panels for the complete helicopter configuration. Required inputs include:

- Shape and dimensions of fuselage, pylon, shaft, and hub
- Number of blades
- Rotor diameter
- Extent of root cutout or shank
- Cross-section of shank
- Rotor blade airfoil designation
- Rotor blade chord distribution
- Rotor blade thickness distribution
- Rotor blade twist distribution
- Rotor tip speed
- Freestream velocity
- Rotor wake geometry

This capability eliminates the requirement for the selection of the aerodynamic paneling by the user.

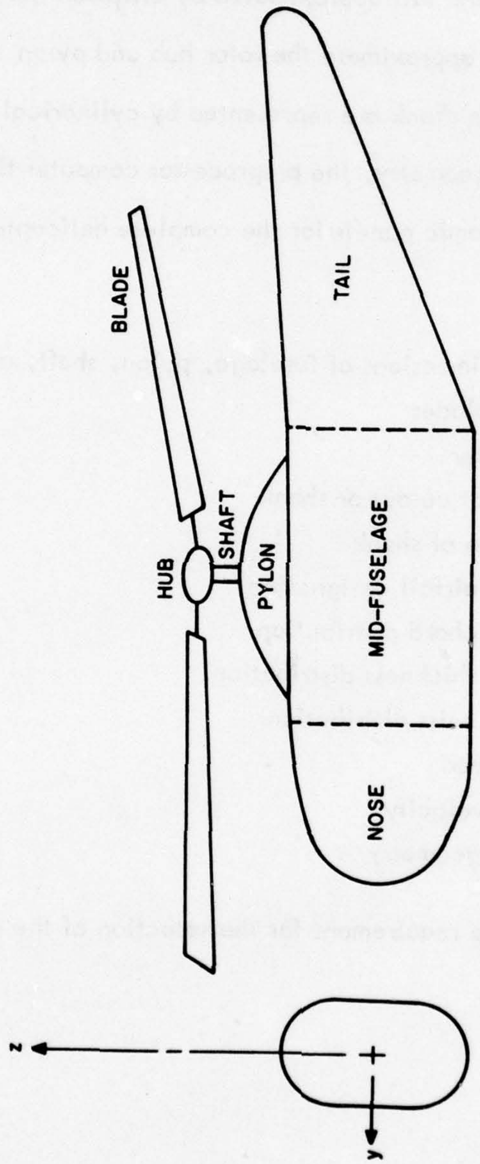


Figure 2. Typical Helicopter Configuration.

SECTION 3

PROGRAM SHAPES DESCRIPTION

This section contains a brief description of the organization of Program SHAPES and its subroutines, a flow diagram, a table of common block storage, and a list of FORTRAN variables.

3.1 PROGRAM SHAPES ORGANIZATION

The present computer program for incompressible potential aerodynamics with separated flow for a complete helicopter, i.e., fuselage, pylon, and rotor, contains three main parts: (1) geometric preprocessor, (2) coefficient matrix, and (3) pressure distribution and force. Based on data provided, the geometric preprocessor automatically generates the aerodynamic panels for the helicopter as well as the numbering and location of the panel corner points in the reference Cartesian (global) coordinate system. The matrix coefficients are determined by evaluating the doublet and source integrals over the surface of the aerodynamic panels. This yields N simultaneous linear equations with N unknown velocity potentials at the centroids of N elements. A standard IBM subroutine GELG (Gaussian elimination method) is used to solve for the unknown potentials. To obtain a continuous distribution for the velocity potential, an averaging scheme is introduced. Hence, the perturbation velocities and the pressure at the centroid of each element can be evaluated. Finally, the aerodynamic coefficients, i.e., lift and induced drag, are computed.

3.1.1 GEOMETRIC PREPROCESSOR

Presently, five main subroutines form the geometric preprocessor:

- DATA
- PREPRO

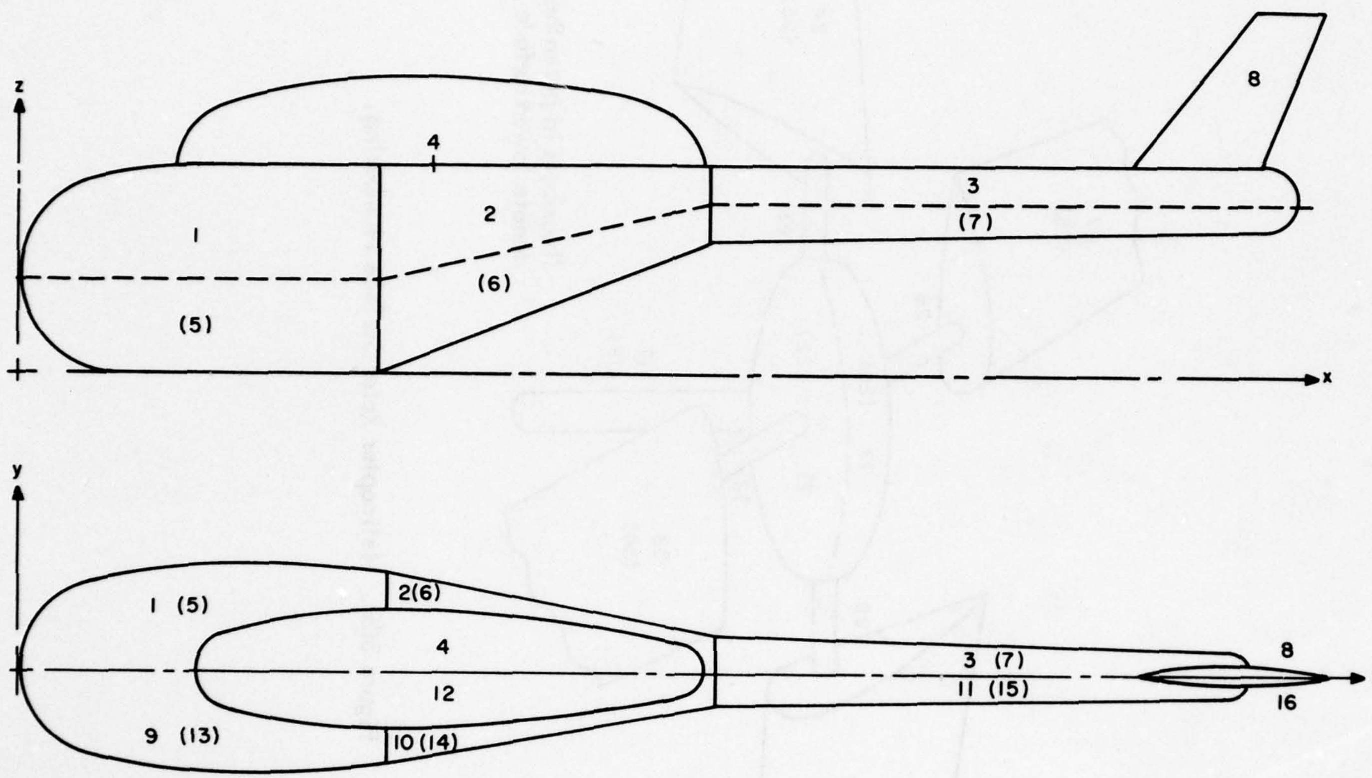
- COODPT
- GEOMET
- VEC123

Basic geometric inputs of the problem are defined by the user in DATA. This subroutine automatically generates the global subsurface numbering of the complete helicopter geometry; i.e., fuselage, pylon, shaft, hub, shank, blade, and vertical tail sections. Figure 3 shows the subsurface global numbering for the helicopter fuselage and rotor configurations. One important input for DATA is the number of the elements in the x and y directions for each subsurface, IS. This input controls the total number of aerodynamic panels used to model the geometry of the problem.

The variation of the element sizes (uniform or nonuniform finite element breakup) is another basic input parameter to the program. In the present investigation, particular attention must be paid to the flow field in the vicinity of the rotor shaft, pylon, hub, and upper fuselage elements. Hence, small elements are prescribed for the hub/pylon region, whereas larger elements can be used on the lower fuselage section. Moreover, the local curvilinear coordinates ξ and η are automatically defined so that the normal to each surface panel is always directed outward.

Finally, DATA automatically defines the parameter KWAKES, which has the value 1 if the subsurface, IS, generates a wake and a value of 0 otherwise. For example, KWAKES = 1 on the pylon. A vortex-layer wake is assumed to emanate from the separation point and this is represented as a doublet layer.

Subroutine PREPRO generates a nodal function NOFCT (IXY, IS), which relates the local nodal numbering, IXY, with the required global nodal numbering. Next, the Cartesian coordinates of these nodes is obtained by COODPT. Basic inputs required for COODPT are the shape and overall dimensions of the fuselage, pylon, and



(Numbers in parentheses denote lower surface.)

Figure 3(a). Helicopter Fuselage Surface Numbering.

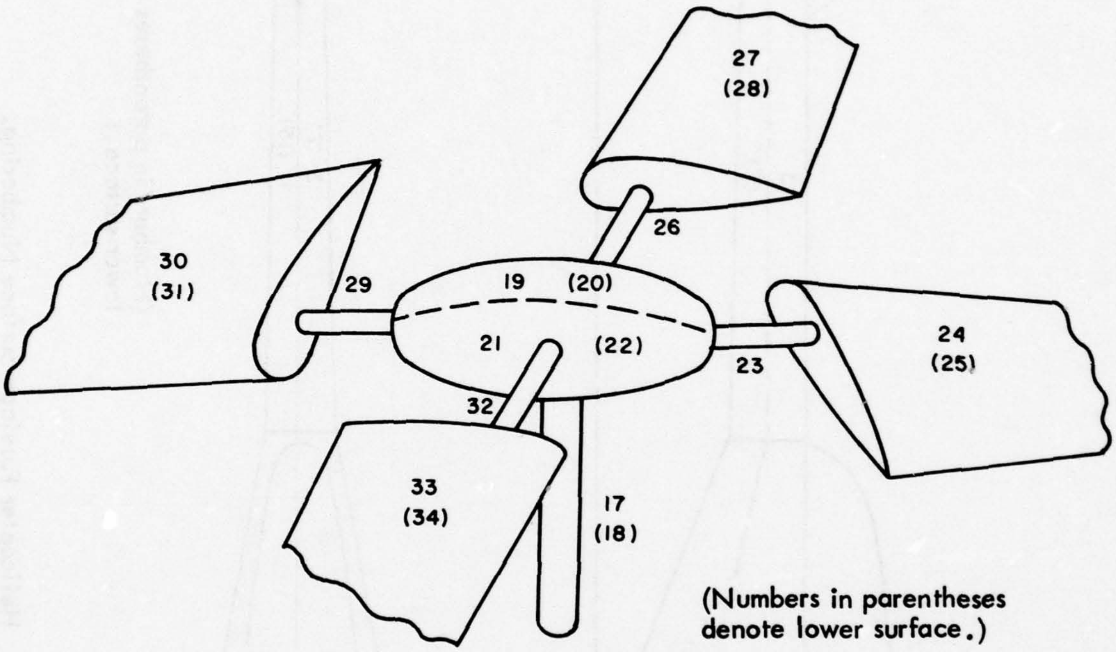


Figure 3(b). Helicopter Rotor Surface Numbering.

hub sections, blade-twist distribution, blade thickness, chord distribution, number of rotor blades, rotor diameter, and extent of root cutout (shank). Subroutine GEOMET automatically generates a parameter NODE as a function of the corner point ICORNR for each aerodynamic panel of the helicopter. Finally, VEC123 completes the geometry of the aerodynamic panel (hyperboloidal element) that contains the four corner points of the element. This results in continuity of the complete helicopter geometry. Note that the equation that represents a hyperboloidal element is given by

$$\bar{P} = \bar{P}_0 + \bar{P}_1 \xi + \bar{P}_2 \eta + \bar{P}_3 \xi\eta \tag{1}$$

where

$$\begin{Bmatrix} \bar{P}_0 \\ \bar{P}_1 \\ \bar{P}_2 \\ \bar{P}_3 \end{Bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{Bmatrix} \bar{P}_{pp} \\ \bar{P}_{pm} \\ \bar{P}_{mp} \\ \bar{P}_{mm} \end{Bmatrix} \tag{2}$$

$\bar{P}_{pp}$, $\bar{P}_{pm}$, $\bar{P}_{mp}$, and $\bar{P}_{mm}$ denote the Cartesian location of the four corner points and $\bar{P}_0$ gives the centroid of the element. This subroutine also performs the Prandtl-Glauert transformation and the rotation of the axis due to angle of attack and angle of sideslip. PRINTA writes the specifications of the problem, node coordinates, the centroid of each element, nodal numbering for each surface, and the nodal numbering of the corner points of the element.

Subroutine CHECK verifies if the maximum permissible number of elements along the x and y directions on each subsurface and the total number of elements are exceeded. Several compatibility conditions exist between the data. If an incompatible relationship is present, an error code will be printed and the execution of the program terminated.

3.1.2 COEFFICIENT MATRIX

Subroutine COEFF forms the matrix coefficients by evaluating the source and doublet integrals over the surface of the aerodynamic panels. The effect of the rotor wake and the presence of separation are automatically included. COEFF yields a system of N linear equations with N unknown velocity potentials at the centroid of N elements. SOLUTN calls GELG to solve N simultaneous linear equations. To obtain the velocity potential at the node of each element, an averaging scheme is employed by AVERAG. Using the value of the velocity potential at the nodes, PHI provides a continuous distribution of the velocity potential. Note that the equation that represents this continuous distribution is given by

$$\varphi = \varphi_0 + \varphi_1 \xi + \varphi_2 \eta + \varphi_3 \xi \eta \tag{3}$$

where

$$\begin{Bmatrix} \varphi_0 \\ \varphi_1 \\ \varphi_2 \\ \varphi_3 \end{Bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{Bmatrix} \varphi_{pp} \\ \varphi_{pm} \\ \varphi_{mp} \\ \varphi_{mm} \end{Bmatrix} \tag{4}$$

φ_{pp}, φ_{pm}, φ_{mp}, and φ_{mm} denote the velocity potential at the nodal points and φ_0 gives the value of φ at the centroid of the element.

3.1.3 PRESSURE DISTRIBUTION AND FORCE

VELXYZ evaluates the perturbation velocity (Cartesian coordinate system) at the centroid of each aerodynamic panel. Subroutine CPLINR uses Bernoulli's equation to compute the pressure distribution at the centroid. PRINTB writes the distribution of sources, doublets, velocity potential, velocity, and pressure distribution. Finally, FORCE evaluates the total lift coefficient and induced drag on the helicopter configuration.

A brief description of the main program and subroutines is given in Table 1.

3.2 FLOW CHART

In order to provide a general understanding of the overall program, the flow chart for SHAPES is shown in Figure 4.

3.3 COMMON BLOCK STORAGE

Most FORTRAN-related variables in Program SHAPES used by more than one subroutine are organized into a number of common blocks as shown in Table 2.

3.4 FORTRAN VARIABLES

Table 3 presents definitions of all principal FORTRAN variables used in Program SHAPES. Where appropriate, mathematical definitions are also indicated. The units of each variable are those used internally by SHAPES, and occasionally differ from the input units.

Table 1. Description of Main Program and Subroutines.

Preprocessor	
MAIN	Controls the logical flow of information supplied by the subroutines.
DATA	Reads input data.
PREPRO	Defines nodal numbering for the helicopter. (The surface of the helicopter configuration is divided into 34 subsurfaces with a maximum of four rotor blades permitted.)
COODPT	Defines and/or reads in the Cartesian coordinates of the nodes on the surface of the helicopter configuration.
GEOMET	Defines the four nodal corners for each aerodynamic panel.
VEC123	Defines the equation of each hyperboloidal surface (i.e., aerodynamic panel).
PRINTA	Writes specification of the problem, nodal numbering for the helicopter configuration, nodal coordinates, and the centroid of the elements.
CHECK	Verifies if the maximum defined in the main program is exceeded.
DEBUG	Writes error message.
Coefficient Matrix	
COEFF	Forms the coefficient matrix.
SOLUTN	Calls GELG to obtain the perturbation aerodynamic potential.
GELG	Solves system of general equations by Gauss elimination.
AVERAG	Obtains potential distribution at the nodes by using an averaging scheme.
PHI	Defines a continuous distribution of velocity potential.
Pressure Distribution and Force	
VELXYZ	Evaluates the velocity (perturbation) distribution at the centroid of each aerodynamic panel.

Table 1. Description of Main Program and Subroutines (Concluded).

CPLINR	Evaluates the pressure distribution at the centroid of each aerodynamic panel (Bernoulli's equation).
FORCE	Evaluates the lift and induced drag coefficients on the helicopter configuration.
PRINTB	Writes the distribution of source, doublet, wake, velocity potential, perturbation velocities, pressure, lift, and induced drag.
<u>Miscellaneous</u>	
ASINH(X)	Evaluates $\sinh^{-1}(X)$.
ATANP(X)	Evaluates principal part of $\tan^{-1}(X)$.
LOG(X)	Evaluates $\ln(X)$.

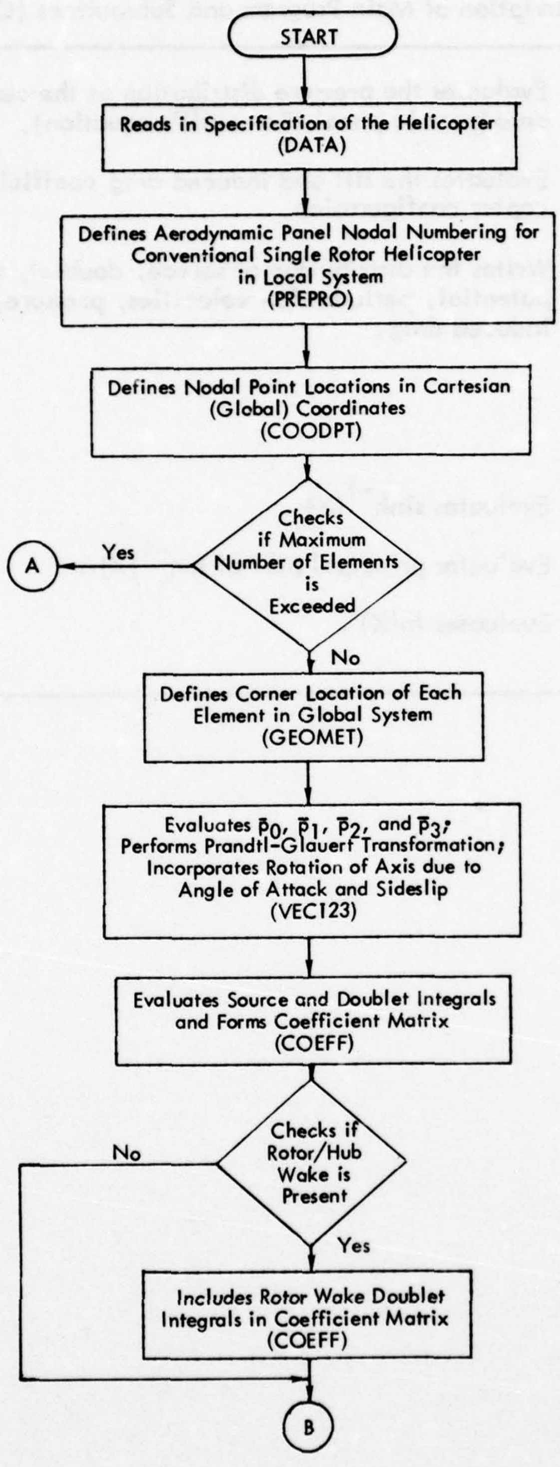


Figure 4. Flow Chart for SHAPES Program.

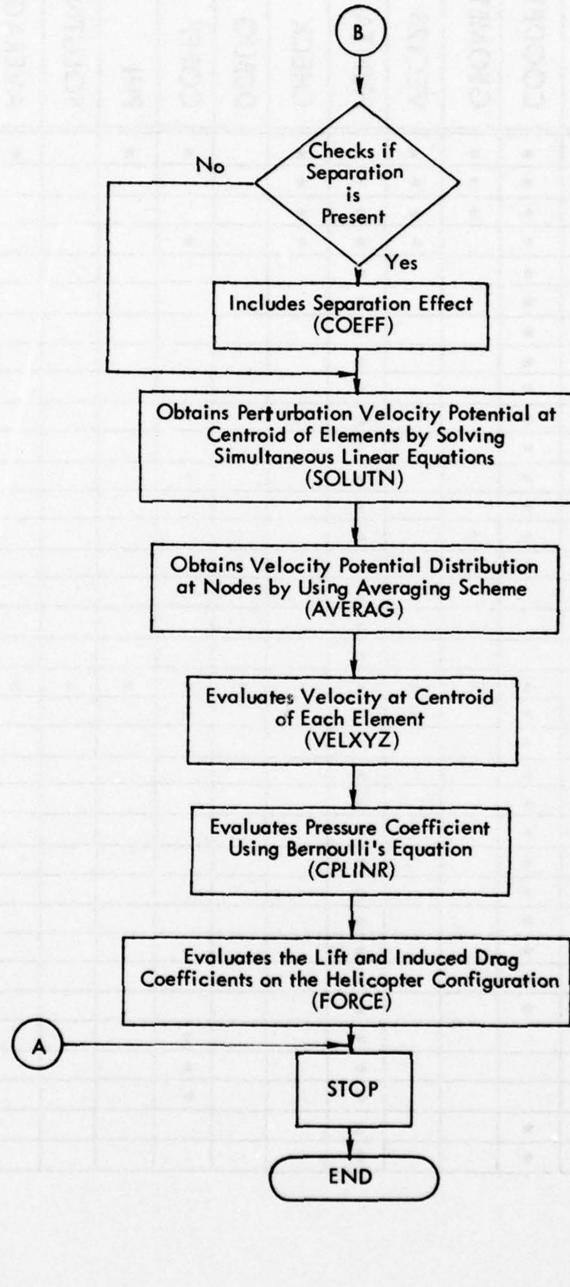


Figure 4. Flow Chart for SHAPES Program (Concluded).

Table 2. Program SHAPES Common Block Organization.

	MAIN	DATA	PREPRO	COODPT	GEOMET	VEC123	PRINTA	CHECK	DEBUG	COEFF	PHI	SOLUTN	AVERAG	VELXYZ	CPLINR	FORCE	PRINTB
ZZZ1	*	*	*	*	*	*	*	*		*	*		*	*	*	*	*
ZZZ2	*	*	*	*	*	*	*	*									*
ZZZ3	*	*	*	*	*	*	*	*									*
ZZZ4	*	*		*		*	*	*		*						*	*
ZZZ5	*	*		*			*										*
ZZZ6	*	*		*			*										*
ZZZ7	*	*		*			*										*
ZZZ8	*	*		*			*										*
ZZZ9	*	*		*			*										*
ZZZ10	*	*		*			*										*
ZZZ11	*	*		*			*										*
ZZZ12	*	*		*			*			*							*
ZZZ13	*	*		*			*										*
ZZZ14	*	*		*	*		*										*
ZZZ15	*	*		*			*										*
ZZZ16	*	*					*										*
ZZZ17	*	*					*										*
ZZZ18	*	*					*										*
ZZZ19	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
ZZZ20	*	*		*		*	*			*							*
ZZZ21	*	*	*	*			*										*
ZZZ22	*	*		*			*										*
ZZZ23	*	*		*			*										*
ZZZ24	*	*		*			*										*
ZZZ25	*	*		*			*										*
ZZZ26	*	*		*			*										*
ZZZ27	*	*		*			*										*
ZZZ28	*	*		*			*										*
ZZZ29	*	*		*			*										*
ZZZ30	*	*		*			*										*
ZZZ31	*	*					*			*							*
ZZZ32	*	*					*			*							*
ZZZ33	*	*					*			*							*
ZZZ34	*	*		*			*										*
ZZZ35	*	*		*			*										*

Table 3. FORTRAN Variables.

Variable Name	Maximum Dimension	Definition
AA	(NESQ)	A coefficient matrix (stored columnwise).
AVG	(NNODE)	The number of elements surrounding the node INODE.
BC	(NELEM)	Downwash on the centroid of IELEMth element.
CP	(NELEM)	Pressure coefficient at the centroid of the IELEMth element.
HCSI	(11)	Defines the normalized nodal line locations along the x direction for surface IS.
HETA	(11)	Defines the normalized nodal line locations along the y direction for surface IS.
ISFACE	(NS)	Defines the global numbering of the surface IS.
KNORML	(NS)	Indicates the direction of the unit normal on surface IS. = 1 outward =-1 inward
KROTOR	(NELEM)	= 1 if the element is on the rotor = 0 otherwise
KROTORS	(NS)	= 1 if the subsurface is part of the rotor assembly = 0 otherwise
KWAKE	(NELEM)	= 1 if the element is in contact with a wake, i.e., rotor blade, hub, pylon, etc. = 0 otherwise
KWAKES	(NS)	= 1 if the surface IS generates a wake = 0 if otherwise
NELEM		Total number of elements.
NNODE		Total number of nodes on the surfaces considered.
NODE	(4,NELEM)	Defines the nodal numbering at the four corners of the element IELEM.
NOFCT	(NXYMP,34)	Defines the nodal numbering of the nodes on the surface IS.
NS		Total number of surfaces considered.

Table 3. FORTRAN Variables (Continued).

Variable Name	Maximum Dimension	Definition
NT	(34)	Subtotal of the number of elements up to, but not including, the surface IS.
NTMAX		Maximum number of elements permissible on the helicopter.
NX	(34)	Number of elements along the x direction of the surface IS.
NXMAX		Maximum number of elements permissible along the x direction on any subsurface.
NXY	(34)	Total number of elements on the surface IS [NXY(IS) = NX(IS) * NY(IS)]
NXYMP		Defines the maximum number of elements permitted on a surface = NXMAX*NYMAX
NY	(34)	Number of elements along the y-direction of the surface IS.
NYMAX		Maximum number of elements permissible along the y direction on any subsurface.
PC	(3,NELEM)	x,y,z coordinates of centroid of elements in the reference Cartesian coordinate system.
P1	(3,NELEM)	A vector along the ξ direction.
P2	(3,NELEM)	A vector along the η direction.
P3	(3,NELEM)	A vector normal to the element IELEM.
PHIC	(NELEM)	Velocity potential at the center of the element.
PHI1	(NELEM)	Interpolation coefficient – the local variation of velocity potential along the ξ direction.
PHI2	(NELEM)	Interpolation coefficient – the local variation of velocity potential along the η direction.

Table 3. FORTRAN Variables (Concluded).

Variable Name	Maximum Dimension	Definition
PH13	(NELEM)	Interpolation coefficient - the local variation of downwash.
SOR	(NNODE)	Velocity potential at the nodes.
SOURCE	(NELEM)	Source distributions at the centroids of elements. After calling subroutine GELG, this variable stores the velocity potential distributions at the centroids of the elements.
VELX	(NELEM)	x component of the velocity at the centroids of the elements.
VELY	(NELEM)	y component of the velocity at the centroids of the elements.
VELZ	(NELEM)	z component of the velocity at the centroids of the elements.
XK	(3, NNODE)	Coordinates of the nodes in the reference Cartesian coordinate system.

SECTION 4
PROGRAM SHAPES USAGE

This section contains a description of the Program SHAPES data input including program options and a description of error codes.

4.1 INPUT DESCRIPTION

All SHAPES data input is accomplished via punched cards. The input variables, required formats, and options are presented in Table 4. Some inputs may require more than one card. The number of input data cards required varies from case to case depending on the type of problem considered. Several compatibility conditions exist among some of the data. If an incompatible relationship is encountered, an error code will be printed and the execution terminated. A typical data setup for running multiple cases is shown in Figure 5.

Table 4. Input Description.

Data 1:	FORMAT (I5)		(Main Program)
	NCASE	Number of cases to be run	
Data 2, 3, and 4:	FORMAT (20A4)		(Subroutine DATA)
	GG(20)	Name, location, date, description of job, and remarks	
Data 5:	FORMAT (2A4, 10I5)		(Subroutine DATA)
	GG(2)	Remarks	
	KREAD	Control code for the definition of nodal coordinates = 0 the coordinates will be defined automatically = 1 the coordinates will be read in	
Data 6:	FORMAT (2A4,10I5)		(Subroutine DATA)
	GG(2)	Remarks	
	NS	Number of surfaces considered	
	KSYMMY	Symmetry condition between left- and right-hand side of helicopter = -1 if geometry is antisymmetric = 0 if geometry has no symmetry = 1 if geometry is symmetric	
	KSYMMZ	Symmetry condition between upper and lower surfaces of helicopter = -1 if geometry is antisymmetric = 0 if geometry has no symmetry = 1 if geometry is symmetric	
Data 7:	FORMAT (2A4,10I5)		(Subroutine DATA)
	GG(2)	Remarks	
	NPYLON	Pylon is considered for analysis = 1 yes = 0 no	
	NBODY1	Fuselage nose section is considered for analysis = 1 yes = 0 no	

Table 4. Input Description (Continued).

	NBODY2	Fuselage midsection is considered for analysis = 1 yes = 0 no	
	NBODY3	Fuselage tail section is considered for analysis = 1 yes = 0 no	
	NVTAIL	Vertical tail is considered for analysis = 1 yes = 0 no	
	NSHAFT	Rotor shaft is considered for analysis = 1 yes = 0 no	
	NHUB	Rotor hub is considered for analysis = 1 yes = 0 no	
	NSHANK	Blade shank is considered for analysis = 1 yes = 0 no	
	NBLADE	Rotor blade is considered for analysis = 1 yes = 0 no	
Data 8:	FORMAT (2A4,10I5)		(Subroutine DATA)
	GG(2)	Remarks	
	KPYL1	Number of elements in the x direction of the mid-fuselage before the pylon	
	KPYL2	Number of elements in the x direction of the mid-fuselage after the pylon	
Data 9:	FORMAT (2A4,7F8.3)		(Subroutine DATA)
	GG(2)	Remarks	
	UMACH	Freestream Mach number	
	OMEGA	Rotor rotational speed (rpm)	
	AREA	Reference area for evaluating aerodynamic coefficient	

Table 4. Input Description (Continued).

Data 10: FORMAT (2A4,7F8.3) (Subroutine DATA)

GG(2)	Remarks
ALFA	Angle of attack (degrees)
BETA	Angle of sideslip (degrees)

If NPYLON = 1, use Data 11; otherwise, skip.

Data 11: FORMAT (2A4,7F8.3) (Subroutine DATA)

GG(2)	Remarks
XPYCTR, YPYCTR, ZPYCTR	Coordinates of pylon's center
RXPYL, RYPYL, RZPYL	Radii of pylon section

If NVTAIL = 1, use Data 12; otherwise, skip.

Data 12: FORMAT (2A4,7F8.3) (Subroutine DATA)

GG(2)	Remarks
VSPAN	Span length of vertical tail
XLEZV	x coordinate of vertical tail leading edge
XTEZV	x coordinate of vertical tail trailing edge
TANLEV	Tangent of leading edge sweep angle
TANTEV	Tangent of trailing edge sweep angle
TAUV	Thickness ratio of vertical tail
ZPV	Height of root chord of vertical tail with respect to global Cartesian coordinate system

Table 4. Input Description (Continued).

If (NBODY1 + NBODY2 + NBODY3) ≠ 0, use DATA 13-16; otherwise, skip.

Data 13: FORMAT (2A4, 7F8.3) (Subroutine DATA)

GG(2)	Remarks
XNOSE	See Figure 6
XBD1	
XBD2	
XTAIL	

Data 14: FORMAT (2A4,7F8.3) (Subroutine DATA)

GG(2)	Remarks
YNOSE	See Figure 6
YBD1	
YBD2	
YTAIL	

Data 15: FORMAT (2A4,7F8.3) (Subroutine DATA)

GG(2)	Remarks
ZNOSE	See Figure 6
ZBD1	
ZBD2	
ZTAIL	

Data 16: FORMAT (2A4,7F8.3) (Subroutine DATA)

GG(2)	Remarks
RYBD1	Radii of fuselage section See Figure 6
RZBD1	
RYBD2	
RZBD2	

Table 4. Input Description (Continued).

If NSHAFT = 1, use Data 17; otherwise, skip.

Data 17:	FORMAT (2A4,7F8.3)		(Subroutine DATA)
	GG(2)	Remarks	
	RSHAFT	Radius of rotor shaft	
	LSHAFT	Length of rotor shaft	

If NHUB = 1, use Data 18; otherwise, skip.

Data 18:	FORMAT (2A4,7F8.3)		(Subroutine DATA)
	GG(2)	Remarks	
	XHUBCR, YHUBCR, ZHUBCR	Coordinates of hub center	
	RXHUB, RYHUB, RZHUB	Radii of hub section	

If NSHANK = 1, use Data 19; otherwise, skip.

Data 19:	FORMAT (2A4,7F8.3)		(Subroutine DATA)
	GG(2)	Remarks	
	RSHANK	Radius of blade shank	
	LSHANK	Length of blade shank	

If (NSHAFT + NHUB + NSHANK + NBLADE) $\neq 0$, use Data 20-23; otherwise, skip.

Data 20:	FORMAT (2A4,7F8.3)		(Subroutine DATA)
	GG(2)	Remarks	
	RROTOR	Blade tip radius	
	YCUT	Blade root cutout	
	XBLE	x coordinate of root chord leading edge	Reference blade at azimuth angle of 90°
	XBTE	x coordinate of root chord trailing edge	Reference blade at azimuth angle of 90°

Table 4. Input Description (Continued).

	TANBLE	Tangent of leading edge sweep angle	
	TANBTE	Tangent of trailing edge sweep angle	
	TAUBL	Thickness ratio of blade airfoil	
Data 21:	FORMAT (2A4,7F8.3)		(Subroutine DATA)
	GG(2)	Remarks	
	THET75	Blade collective pitch at three-quarter radius	
	THET1C	Blade lateral cyclic pitch	
	THET1S	Blade longitudinal cyclic pitch	
	CONING	Blade coning angle	
	AZIMUTH	Blade azimuth angle (degrees)	
Data 22:	FORMAT (2A4,7F8.3)		(Subroutine DATA)
	GG(2)	Remarks	
	RPITCH	Collective pitch angle at blade root (degrees)	
	TWIST	Blade twist (degrees)	
Data 23:	FORMAT (2A4,10I5)		(Subroutine DATA)
	GG(2)	Remarks	
	KBLADE	Number of blades	
	ITWIST	Defines blade twist distribution	
Data 24:	FORMAT (2A4,10I5)		(Subroutine DATA)
	GG(2)	Remarks	
	NWAKPY	Flow separation considered for analysis = 1 yes = 0 no	

Table 4. Input Description (Continued).

NWAKHB	Rotor hub wake is generated for analysis = 1 yes = 0 no
NWAKSK	Blade shank wake is generated for analysis = 1 yes = 0 no
NWAKBL	Rotor blade wake is generated for analysis = 1 yes = 0 no

If (NWAKHB + NWAKSK + NWAKBL) $\neq 0$, use Data 25; otherwise, skip.

Data 25: FORMAT (2A4,15,2F8.3) (Subroutine DATA)

GG(2)	Remarks
NSPIRAL	Number of elements along one rotor wake spiral
SPIRAL	Number of rotor wake spirals
UWAKE	Induced rotor velocity = $\sqrt{1/2\ C_T}\ \Omega R$

If NWAKPY = 1, use Data 26-27; otherwise, skip.

Data 26: FORMAT (2A4,1015) (Subroutine DATA)

GG(2)	Remarks
NSTAG	Vortex layer from separation line is considered for analysis = 1 yes = 0 no
NVORT	Isolated vortex (branch wake) is considered for analysis = 1 yes = 0 no

Data 27: FORMAT (2A4,7F8.3) (Subroutine DATA)

GG(2)	Remarks
CSTAG	Intensity of the vortex layer from separation line
CVORT	Intensity of the isolated vortex (branch wake)

Table 4. Input Description (Continued).

Data 28: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
KPRINT(I)	Output control code = 1 yes = 0 no I = 1 specification of the problem I = 2 nodal numbering of surfaces and corner nodal numbering of elements I = 3 Cartesian coordinates of the nodes I = 4 Cartesian coordinates of the centroids of the element I = 5 coefficient matrix AA I = 6 source integrals I = 7 velocity potential distributions I = 8 perturbation velocity distributions I = 9 pressure coefficients I = 10 lift and induced drag coefficients

If NBODY1 = 1, use Data 29; otherwise, skip.

Data 29: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in x direction
NY(KS)	Number of elements in y direction
KNSELE	Define the variation of element size = 0 define by input = 1 uniform along x direction = 2 quadratic along x direction
KNSSHP	Defines nose shape = 1; $r = R[\xi]^{1/2}$ = 2; $r = R[\xi]^{1/3}$ = 3; $r = R[\xi]^{1/4}$
KNSTYP	Defines nose type = 1 circular cross section = 2 elliptical cross section

Table 4. Input Description (Continued).

If NBODY2 = 1, use Data 30; otherwise, skip.

Data 30: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in x direction
NY(KS)	Number of elements in y direction
KBDELE	Defines the variation of element size = 0 define by input = 1 uniform along x direction = 2 quadratic along x direction
KBDSHP	Defines the fuselage shape = 1 cylindrical
KBDTYP	Defines the fuselage type = 1 circular cross section = 2 elliptical cross section

If NBODY3 = 1, use Data 31; otherwise, skip.

Data 31: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in x direction
NY(KS)	Number of elements in y direction
KTNELE	Defines the variation of element size = 0 define by input = 1 uniform along x direction = 2 quadratic along x direction
KTNSHP	Defines aft-body shape = 1; $r = R\sqrt{\xi}$ = 2; $r = R\,[\xi]^{1/3}$ = 3; $r = R\,[\xi]^{1/4}$
KTNTYP	Defines aft-body type = 1 circular cross section = 2 elliptical cross section

Table 4. Input Description (Continued).

If NPYLON = 1, use Data 32; otherwise, skip.

Data 32: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in radial direction
NY(KS)	Number of elements in circumferential direction
KPYELE	Defines the variation of element size = 0 define by input = 1 uniform element distribution along radial and circumferential direction = 2 quadratic along radial direction with uniform element distribution in circumferential direction
KPYSHP	Defines pylon shape = 1 elliptical
KPYTYP	Defines pylon type = 1 elliptical

If NVTAIL = 1, use Data 33; otherwise, skip.

Data 33: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in x direction
NY(KS)	Number of elements in y direction
KVTELE	Defines the variation of element size = 0 define by input = 1 uniform element distribution along axial and spanwise direction = 2 nonuniform (quadratic) element distribution along axial and spanwise direction = 3 quadratic along spanwise direction with uniform element distribution in axial direction = 4 quadratic along axial direction with uniform element distribution in spanwise direction

Table 4. Input Description (Continued).

KVTSHP

Defines vertical tail shape

= 1 circular biconvex airfoil

$$Z = \pm \left\{ \left[\frac{\left(\frac{c}{2}\right)^2 + \tau_{max}^2}{2\,\tau_{max}} \right]^2 - \left[X - \frac{X_{LE} + X_{TE}}{2} \right]^2 \right\}^{1/2} - \left[\frac{\left(\frac{c}{2}\right)^2 - \tau_{max}^2}{2\,\tau_{max}} \right]$$

= 2 define by following equation

$$Z = \pm \bar{\tau} \left\{ \sqrt{\left(\frac{X}{X_{TE} - X_{LE}} \right) \left(1 - \frac{X}{X_{TE} - X_{LE}} \right)} \sqrt{1 - \frac{y}{s}} \right\}$$

= 3 define by following equation

$$Z = \pm (4\,\tau)\,(2\,C_o) \left\{ \left(\frac{X}{X_{TE} - X_{LE}} \right) \left(1 - \frac{X}{X_{TE} - X_{LE}} \right) \sqrt{1 - \frac{y}{s}} \right\}$$

Table 4. Input Description (Continued).

KVTSHP (Cont.)	where
	c = chord length
	X_{LE} = x components of sectional leading edge
	X_{TE} = x components of sectional trailing edge
	y = spanwise location of the section
	S = half span
	$\bar{\tau} = \tau_{max} \left(\frac{3}{4} \sqrt{3} \right) (2\,C_o)$
	C_o = root chord
KVTTYP	Define vertical tail type = 1

If NSHAFT = 1, use Data 34; otherwise, skip.

Data 34: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in shaft direction
NY(KS)	Number of elements in circumferential direction
KSHELE	Defines the variation of element size = 0 define by input = 1 uniform element distribution
KSHSHP	Defines rotor shaft shape = 1 cylindrical
KSHTYP	Defines rotor shaft type = 1 circular cross section

Table 4. Input Description (Continued).

If NHUB = 1, use Data 35; otherwise, skip.

Data 35: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in x direction
NY(KS)	Number of elements in circumferential direction
KHBELE	Defines the variation of element size = 0 define by input = 1 uniform element distribution
KHBSHP	Defines rotor hub shape = 1 elliptical
KHBTYP	Defines rotor hub type = 1 elliptical

If NSHANK = 1, use Data 36; otherwise, skip.

Data 36: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in shank direction
NY(KS)	Number of elements in circumferential direction
KSKELE	Defines the variation of element size = 0 define by input = 1 uniform element distribution
KSKSHP	Defines blade shank shape = 1 cylindrical
KSKTYP	Defines blade shank type = 1 circular cross section

If NBLADE = 1, use Data 37; otherwise, skip.

Data 37: FORMAT (2A4,10I5) (Subroutine DATA)

GG(2)	Remarks
NX(KS)	Number of elements in x direction

Table 4. Input Description (Continued).

NY(KS) — Number of elements in y direction

KBLELE — Defines the variation of element size
= 0 define by input
= 1 uniform element distribution
= 2 nonuniform (quadratic) element distribution

KBLSHP — Defines rotor blade shape
= 1 circular biconvex airfoil

$$Z = \pm \left\{ \left[\frac{\left(\frac{c}{2}\right)^2 + \tau_{max}^2}{2\,\tau_{max}} \right]^2 - \left[X - \frac{X_{LE} + X_{TE}}{2} \right]^2 - \frac{\left(\frac{c}{2}\right)^2 - \tau_{max}^2}{2\,\tau_{max}} \right\}^{1/2}$$

= 2 define by following equation

$$Z = \pm \bar{\tau} \left\{ \sqrt{\left(\frac{X}{X_{TE} - X_{LE}} \right) \left(1 - \frac{X}{X_{TE} - X_{LE}} \right)} \sqrt{1 - \frac{r}{R}} \right\}$$

= 3 define by following equation

$$Z = \pm (4\tau)(2\,C_o) \left\{ \left(\frac{X}{X_{TE} - X_{LE}} \right) \left(1 - \frac{X}{X_{TE} - X_{LE}} \right) \sqrt{1 - \frac{r}{R}} \right\}$$

Table 4. Input Description (Continued).

KBLSHP (Cont.) where

c = blade chord

X_{LE} = x components of blade leading edge

X_{TE} = x components of blade trailing edge

r = radial location

R = rotor radius

$$\bar{\tau} = \tau_{max} \left(\frac{3}{4} \sqrt{3} \right) (2\, C_o)$$

C_o = blade root chord

KBLTYP — Defines rotor blade type = 1

If NPYLON = 1, use Data 38-39; otherwise, skip.

Data 38: FORMAT (10F8.3) (Subroutine COODPT)

HCSI(IX) — Normalized x direction nodal line coordinates [Note that HCSI(IX), IX = 1, NXP is implied with NXP denoting the number of nodal lines along the x direction]

Data 39: FORMAT (10F8.3) (Subroutine COODPT)

HETA(IY) — Normalized y direction nodal line coordinates [Note that HETA(IY), = 1, NYP is implied with NYP denoting the number of nodal lines along the y direction]

If NVTAIL = 1, use Data 40-41; otherwise, skip.

Data 40: FORMAT (10F8.3) (Subroutine COODPT)

HCSI(IX) — Normalized x direction nodal line coordinates [Note that HCSI(IX), IX = 1, NXP is implied with NXP denoting the number of nodal lines along the x direction]

Table 4. Input Description (Continued).

Data 41: FORMAT (10F8.3) (Subroutine COODPT)

HETA(IY) Normalized y direction nodal line coordinates [Note that HETA(IY), IY = 1, NYP is implied with NYP denoting the number of nodal lines along the y direction]

If NSHAFT = 1, use Data 42–43; otherwise, skip.

Data 42: FORMAT (10F8.3) (Subroutine COODPT)

HCSI(IX) Normalized x direction nodal line coordinates [Note that HCSI(IX), IX = 1, NXP is implied with NXP denoting the number of nodal lines along the x direction]

Data 43: FORMAT (10F8.3) (Subroutine COODPT)

HETA(IY) Normalized y direction nodal line coordinates [Note that HETA(IY), IY = 1, NYP is implied with NYP denoting the number of nodal lines along the y direction]

If NHUB = 1, use Data 44–45; otherwise, skip.

Data 44: FORMAT (10F8.3) (Subroutine COODPT)

HCSI(IX) Normalized x direction nodal line coordinates [Note that HCSI(IX), IX = 1, NXP is implied with NXP denoting the number of nodal lines along the x direction]

Data 45: FORMAT (10F8.3) (Subroutine COODPT)

HETA(IY) Normalized y direction nodal line coordinates [Note that HETA(IY), IY = 1, NYP is implied with NYP denoting the number of nodal lines along the y direction]

If NSHANK = 1, use Data 46–47; otherwise, skip.

Data 46: FORMAT (10F8.3) (Subroutine COODPT)

HCSI(IX) Normalized x direction nodal line coordinates [Note that HCSI(IX), IX = 1, NXP is implied with NXP denoting the number of nodal lines along the x direction]

Table 4. Input Description (Continued).

Data 47: FORMAT (10F8.3) (Subroutine COODPT)

HETA(IY) — Normalized y direction nodal line coordinates [Note that HETA(IY), IY = 1, NYP is implied with NYP denoting the number of nodal lines along the y direction]

If NBLADE = 1, use Data 48–49; otherwise, skip.

Data 48: FORMAT (10F8.3) (Subroutine COODPT)

HCSI(IX) — Normalized x direction nodal line coordinates [Note that HCSI(IX), IX = 1, NXP is implied with NXP denoting the number of nodal lines along the x direction]

Data 49: FORMAT (10F8.3) (Subroutine COODPT)

HETA(IY) — Normalized y direction nodal line coordinates [Note that HETA(IY), IY = 1, NYP is implied with NYP denoting the number of nodal lines along the y direction]

If KREAD = 1, use Data 50; otherwise, skip.

Data 50: FORMAT (3E12.4) (Subroutine COODPT)

XK(K,INODE) — The nodal coordinates in global Cartesian system [Note that (XK(K,INODE), K = 1,3), INODE = 1, NNODE) is implied]

If NWAKPY = 1 and NSTAG = 1, use Data 51-54; otherwise, skip.

Data 51: FORMAT (3E12.4) (Subroutine COODPT)

YPP(K)

Data 52: FORMAT (3E12.4)

YPM(K)

Data 53: FORMAT (3E12.4)

YMP(K)

Data 54: FORMAT (3E12.4)

YMM(K)

(Data 51–54:) The separation wake nodal coordinates in global Cartesian system [Note that K=1,3 is implied]

Table 4. Input Description (Concluded).

If NWAKPY = 1 and NVORT = 1, use Data 55-58; otherwise, skip.

Data 55:	FORMAT (3E12.4)	(Subroutine COEFF)
	YPP(K)	The isolated vortex branch wake nodal coordinates in global Cartesian system [Note that K=1,3 is implied]
Data 56:	FORMAT (3E12.4)	
	YPM(K)	
Data 57:	FORMAT (3E12.4)	
	YMP(K)	
Data 58:	FORMAT (3E12.4)	
	YMM(K)	

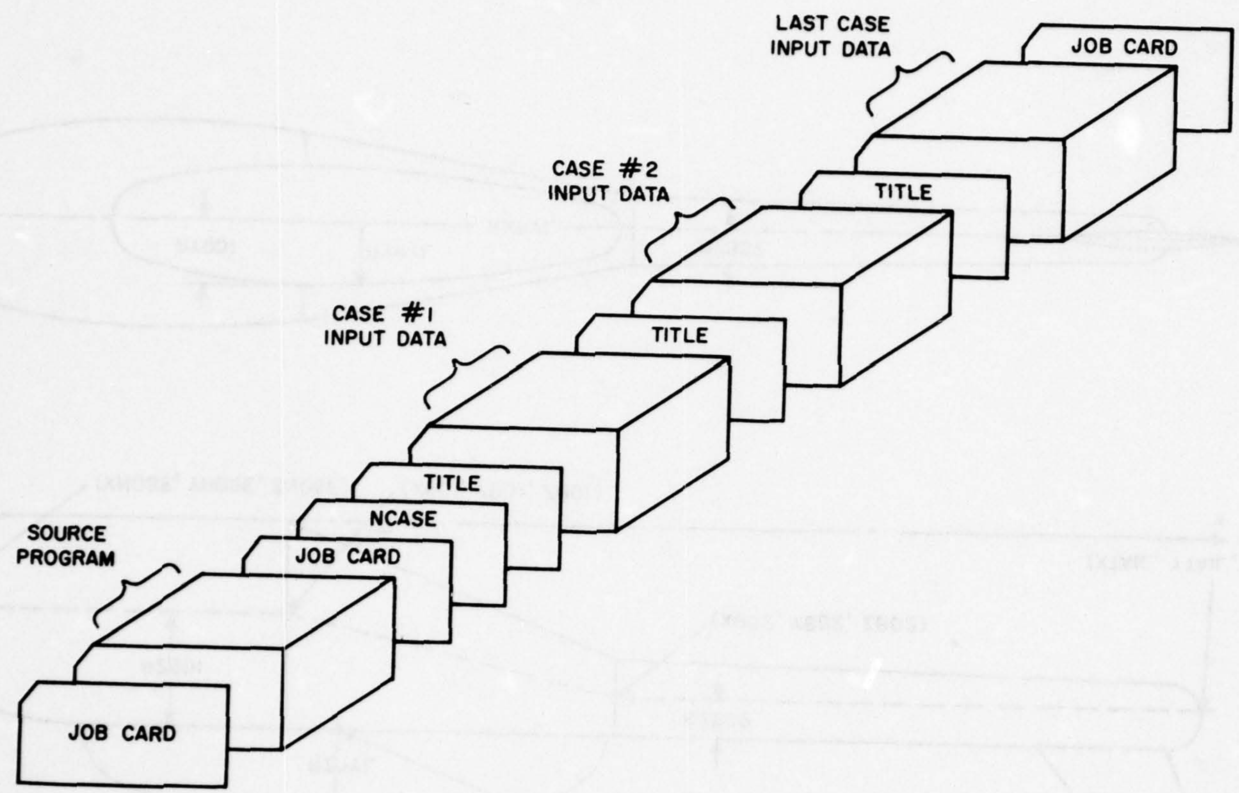


Figure 5. Program SHAPES Data Setup.

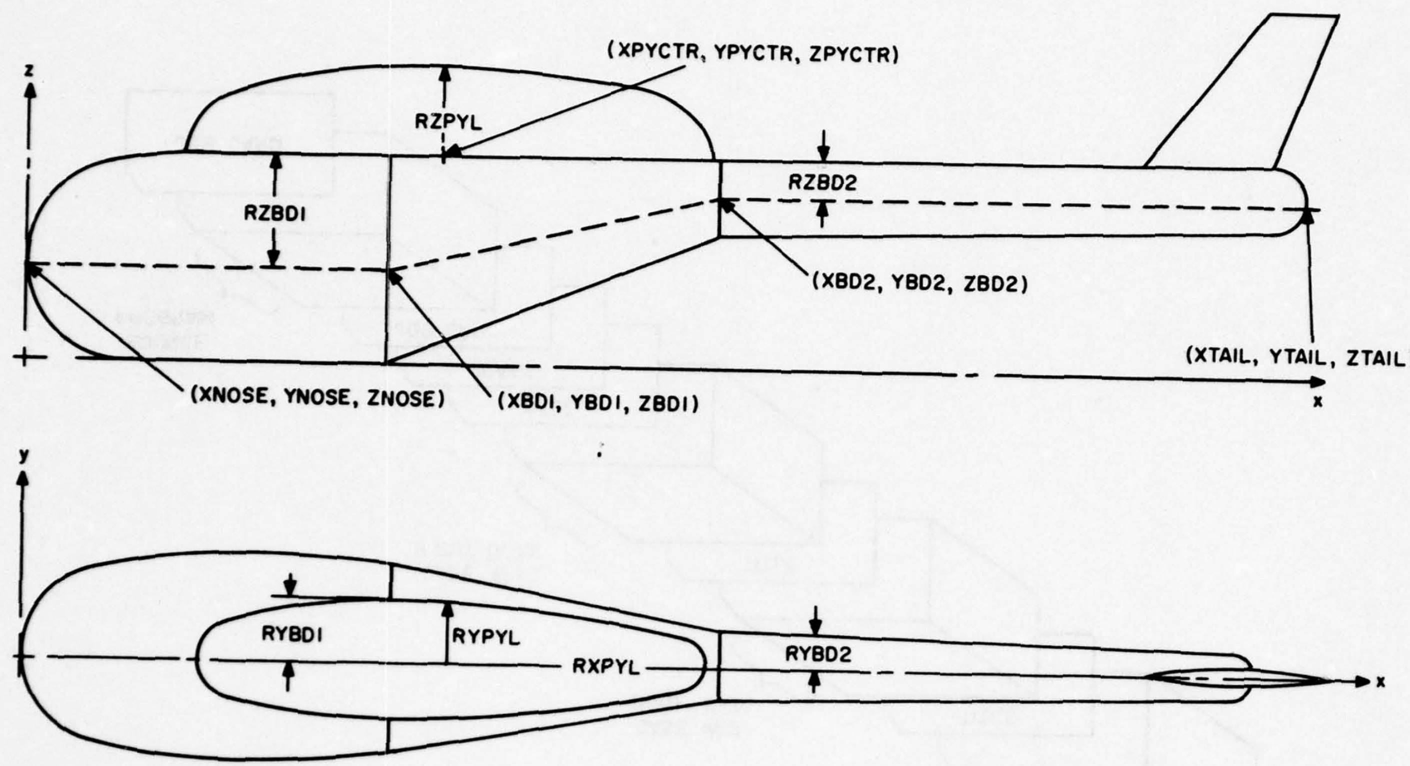


Figure 6. Basic Geometric Inputs for Helicopter Fuselage.

4.2 ERROR CODES

To insure proper data are read in and to minimize unnecessary computer time, SHAPES contains several debugging statements at specified checkpoints. If an inconsistency among input data or violation of the input specification is detected, an error code will be printed and the computer run terminated. Table 5 summarizes these error codes.

Table 5. Error Codes Summary.

Code Number	Description
100	Mach number is greater than 1 (Present SOUSSA is a subsonic program) (DATA)
200	NS ≠ (NPYLON + NVTAIL)*MULTY + (NBODY1 + NBODY2 + (NBODY3)*MULT*MULTY + (NSHAFT*MULTY + (NHUB*MULT*MULTY* + KBLADE*(NSHANK + NBLADE*MULT) (DATA)
300	Number of elements in the x direction of a sub-surface exceeds the maximum permissible value. This is limited by the storage capacity of the computer. The user has the option of changing the value of NXMAX and NYMAX defined in the main program. (DATA)
400	Same as code number 300 except for the y direction. (DATA)
500	Total number of elements required for analysis exceeds the maximum limit specified in the main program. (DATA)
IER	IBM subroutine GELG provides error code = 0 no error = 1 no result is obtained because the number of equations is less than 1 or the pivoting element at any elimination step equals 0 = N warning is indicated because of a possible loss of significant figures at elimination step N+1 where the pivoting element is less than equal to the specified tolerance times the magnitude of the greatest element of matrix A (DATA)

SECTION 5

PROGRAM OUTPUT

In Program SHAPES several output options are available and these are briefly summarized here. The amount of output required to be printed out is controlled by Data 28, described in Subsection 4.1. Among the information available is the specification of the problem defined by the user and the basic geometric inputs to the problem, i.e., overall dimensions and shapes used to model a helicopter configuration. In addition, the nodal numbering of the aerodynamic breakup as well as the Cartesian coordinate location of the nodal points and the centroid of each aerodynamic panel can be requested. Furthermore, the coefficient matrix and source distribution can be output. Also, the perturbation potential, velocity, and the pressure coefficient are available as outputs. Finally, the lift and induced drag can be requested.

SECTION 6

PROGRAM LISTING

SAMPLE CASE 1:

SINGLE-BLADE ROTOR

Data Input

```
  3
PROGRAM SHAPES
HUB PYLON FLOW SEPARATION
SINGLE ROTOR BLADE PROBLEM
KREAD      0
9          2    1    0
99         0    0    0    0    0    0    0    0    1
999        0    C
UMACH        0.000    355.030
ALFA         0.000      0.000
ROTOR       17.500      2.330     0.000     1.093     0.000     0.300     .050
THETAS       0.023      0.030     0.000     1.350    90.600
RPITCH      10.610      0.070
KBLADE     1    1
NWAKE      0    0    0    1
SPIRAL    10       3.000     20.000
KPRINT     1    1    1    1    0    1    1    1    1    1
NROTOR     3    7    2    1    1

      DATA

   NELEM =      42

   IS=    1 NX=    3 NY=    7 NXY=   21 ISFACE=    1 NT=    0 KNORML=    1 KWAKES=    1 KROTORS=    1
   IS=    2 NX=    3 NY=    7 NXY=   21 ISFACE=    2 NT=   21 KNORML=   -1 KWAKES=    1 KROTORS=    1

      PREPRO
   NNCDE=   46
```

Specifications of the Problem

```
FOR PART   24
NX= 3
NY= 7

FOR PART   25
NX= 3
NY= 7

NELEM= 42

KSYMMY= 1
KSYMMZ= 0
```

X, Y, and Z Coordinates of the Nodes

NODE=	X	Y	Z
1	0.00000	2.33000	0.00000
2	.12025	2.33000	-.01163
3	.47803	2.33000	-.06234
4	1.06448	2.33000	-.19941
5	0.00000	6.35469	0.00000
6	.12025	6.35469	-.01163
7	.47803	6.35469	-.06234
8	1.06448	6.35469	-.19941
9	0.00000	9.76020	0.00000
10	.12025	9.76020	-.01163
11	.47803	9.76020	-.06234
12	1.06448	9.76020	-.19941
13	0.00000	12.54553	0.00000
14	.12025	12.54553	-.01163
15	.47803	12.54553	-.06234
16	1.06446	12.54553	-.19941
17	0.00000	14.71367	0.00000
18	.12025	14.71367	-.01163
19	.47803	14.71367	-.06234
20	1.06446	14.71367	-.19941
21	0.00000	16.26163	0.00000
22	.12025	16.26163	-.01163
23	.47803	16.26163	-.06234
24	1.06446	16.26163	-.19941
25	0.00000	17.19041	0.00000
26	.12025	17.19041	-.01163
27	.47803	17.19041	-.06234
28	1.06448	17.19041	-.19941
29	0.00000	17.50000	0.00000
30	.11628	17.50000	-.02216
31	.47319	17.50000	-.06862
32	1.06446	17.50000	-.19941
33	.11630	2.33000	-.03269
34	.46818	2.33000	-.11491
35	.11630	6.35469	-.03269
36	.46818	6.35469	-.11491
37	.11630	9.76020	-.03269
38	.46818	9.76020	-.11491
39	.11630	12.54553	-.03269
40	.46818	12.54553	-.11491
41	.11630	14.71367	-.03269
42	.46818	14.71367	-.11491
43	.11630	16.26163	-.03269
44	.46818	16.26163	-.11491
45	.11630	17.19041	-.03269
46	.46818	17.19041	-.11491

X, Y, and Z Coordinates of the Centroid of Aerodynamic Panels

ELEM=	XPC	YPC	ZPC
1	.06012	4.34235	-.00561
2	.29914	4.34235	-.03698
3	.77126	4.34235	-.13087
4	.06012	8.05745	-.00561
5	.29914	8.05745	-.03698
6	.77126	8.05745	-.13087
7	.06012	11.15337	-.00561
8	.29914	11.15337	-.03698
9	.77126	11.15337	-.13087
10	.06012	13.63010	-.00561
11	.29914	13.63010	-.03698
12	.77126	13.63010	-.13087
13	.06012	15.48765	-.00561
14	.29914	15.48765	-.03698
15	.77126	15.48765	-.13087
16	.06012	16.72602	-.00561
17	.29914	16.72602	-.03698
18	.77126	16.72602	-.13087
19	.05963	17.34520	-.00845
20	.29741	17.34520	-.04619
21	.77003	17.34520	-.13744
22	.05815	4.34235	-.01634
23	.29224	4.34235	-.07380
24	.76633	4.34235	-.15716
25	.05815	8.05745	-.01634
26	.29224	8.05745	-.07380
27	.76633	8.05745	-.15716
28	.05815	11.15337	-.01634
29	.29224	11.15337	-.07380
30	.76633	11.15337	-.15716
31	.05815	13.63010	-.01634
32	.29224	13.63010	-.07380
33	.76633	13.63010	-.15716
34	.05815	15.48765	-.01634
35	.29224	15.48765	-.07380
36	.76633	15.48765	-.15716
37	.05815	16.72602	-.01634
38	.29224	16.72602	-.07380
39	.76633	16.72602	-.15716
40	.05864	17.34520	-.01371
41	.29397	17.34520	-.06459
42	.76756	17.34520	-.15059

Nodal Numbering for Surfaces

FOR SURFACE 24

1	5	9	13	17	21	25	29
2	6	10	14	18	22	26	30
3	7	11	15	19	23	27	31
4	8	12	16	20	24	28	32

FOR SURFACE 25

1	5	9	13	17	21	25	29
33	35	37	39	41	43	45	30
34	36	38	40	42	44	46	31
4	8	12	16	20	24	28	32

Nodal Numbering for Elements

ELEM	++	+-	-+	--
1	6	2	5	1
2	7	3	6	2
3	8	4	7	3
4	10	6	9	5
5	11	7	10	6
6	12	8	11	7
7	14	10	13	9
8	15	11	14	10
9	16	12	15	11
10	18	14	17	13
11	19	15	18	14
12	20	16	19	15
13	22	18	21	17
14	23	19	22	18
15	24	20	23	19
16	26	22	25	21
17	27	23	26	22
18	28	24	27	23
19	30	26	29	25
20	31	27	30	26
21	32	28	31	27
22	33	35	1	5
23	34	36	33	35
24	4	8	34	36
25	35	37	5	9
26	36	38	35	37
27	8	12	36	38
28	37	39	9	13
29	38	40	37	39
30	12	16	38	40
31	39	41	13	17
32	40	42	39	41
33	16	20	40	42
34	41	43	17	21
35	42	44	41	43
36	20	24	42	44
37	43	45	21	25
38	44	46	43	45
39	24	28	44	46
40	45	30	25	29
41	46	31	45	30
42	28	32	46	31

The Distribution of the Source

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

THE DISTRIBUTION OF SOURCE

FOR SUBSURFACE 24

-.54553E+01	-.98655E+01	-.13411E+02	-.16271E+02	-.18312E+02	-.10733E+02	-.10910E+02
-.27443E+01	-.47637E+01	-.64467E+01	-.77902E+01	-.87494E+01	-.88877E+01	-.53715E+01
.40619E+01	.91132E+01	.12643E+02	.15465E+02	.17488E+02	.17995E+02	.93205E+01

FOR SUBSURFACE 25

-.60686E+01	-.10951E+02	-.15012E+02	-.18247E+02	-.20505E+02	-.21201E+02	-.11787E+02
-.46065E+01	-.85095E+01	-.11727E+02	-.14336E+02	-.16288E+02	-.17888E+02	-.97158E+01
.33064E+01	.63446E+01	.87286E+01	.10639E+02	.11958E+02	.12013E+02	.61040E+01

The Distribution of the Velocity Potential

THE DISTRIBUTION OF THE VELOCITY POTENTIAL

FOR SUBSURFACE 24

.97743E+02	.12076E+03	.13228E+03	.13186E+03	.12223E+03	.16625E+03	.94556E+02
.11366E+03	.14857E+03	.15739E+03	.16143E+03	.15467E+03	.13982E+03	.11436E+03
.11917E+03	.15296E+03	.17375E+03	.18066E+03	.17560E+03	.15953E+03	.12816E+03

FOR SUBSURFACE 25

.66397E+02	.64161E+02	.59945E+02	.47274E+02	.29003E+02	.10263E+02	.27677E+02
.49032E+02	.48466E+02	.39531E+02	.23127E+02	.34100E+01	-.62300E+01	.10612E+02
.46693E+02	.42095E+02	.30273E+02	.12052E+02	-.76572E+01	-.13457E+02	.61021E+01

Pressure Distribution

THE DISTRIBUTION OF CP

FOR SUBSURFACE 24

-.80636E+00	-.13691E+01	-.19902E+01	-.25306E+01	-.28028E+01	-.28655E+01	-.13838E+01
-.12658E+00	-.19535E+00	-.26234E+00	-.36065E+00	-.41152E+00	-.38549E+00	-.21216E+00
.22533E+00	.35278E+00	.51106E+00	.65083E+00	.73322E+00	.69717E+00	.28366E+00

FOR SUBSURFACE 25

.82564E+00	.12756E+01	.18370E+01	.23371E+01	.26603E+01	.25463E+01	.12186E+01
.81657E-01	.13226E+00	.19753E+00	.25337E+00	.27727E+00	.24846E+00	.87001E-01
-.24629E+00	-.38651E+00	-.54698E+00	-.69930E+00	-.75603E+00	-.76448E+00	-.35874E+00

SAMPLE CASE 2:

XH-51A HELICOPTER ROTOR

Data Input

```
  3
PROGRAM SHAPES
HUB PYLON FLOW SEPARATION
FOUR BLADED HELICOPTER ROTOR
KREAD      0
*          0    0    0
**         0    0    0    0    0    0    0    0    1
***        0    0
UMACH       0.000    359.130
ALFA        0.000      0.000
ROTOR      17.500      2.330      0.000      1.003      0.000      0.000      .050
THETTS      0.000      0.000      0.000      0.000     90.000
RPITCH     10.510      3.000
KBLADE     4    1
NWAKE      0    0    0    1
SPIRAL    10       3.000      39.000
KPRINT     1    1    1    1    0    1    1    1    1    1
NROTOR     0    5    2    1    1

      DATA

   NELEM =     200

   IS=   1 NX=    5 NY=    5 NXY=   25 ISFACE=   1 NT=    0 KNORML=   1 KWAKES=   1 KROTCRS=   1
   IS=   2 NX=    5 NY=    5 NXY=   25 ISFACE=   2 NT=   25 KNORML=  -1 KWAKES=   1 KROTCRS=   1
   IS=   3 NX=    5 NY=    5 NXY=   25 ISFACE=   3 NT=   50 KNORML=   1 KWAKES=   1 KROTCRS=   1
   IS=   4 NX=    5 NY=    5 NXY=   25 ISFACE=   4 NT=   75 KNORML=  -1 KWAKES=   1 KROTCRS=   1
   IS=   5 NX=    5 NY=    5 NXY=   25 ISFACE=   5 NT=  100 KNORML=   1 KWAKES=   1 KROTCRS=   1
   IS=   6 NX=    5 NY=    5 NXY=   25 ISFACE=   6 NT=  125 KNORML=  -1 KWAKES=   1 KROTCRS=   1
   IS=   7 NX=    5 NY=    5 NXY=   25 ISFACE=   7 NT=  150 KNORML=   1 KWAKES=   1 KROTCRS=   1
   IS=   8 NX=    5 NY=    5 NXY=   25 ISFACE=   8 NT=  175 KNORML=  -1 KWAKES=   1 KROTCRS=   1

      PREPRO
   NNODE=  224
```

Specifications of the Problem

```
FOR PART   24
NX= 5
NY= 5

FOR PART   25
NX= 5
NY= 5

FOR PART   27
NX= 5
NY= 5

FOR PART   28
NX= 5
NY= 5

FOR PART   30
NX= 5
NY= 5

FOR PART   31
NX= 5
NY= 5

FOR PART   33
NX= 5
NY= 5

FOR PART   34
NX= 5
NY= 5

NELEM=200

KSYMMY= 0
KSYMMZ= 0
```

X, Y, and Z Coordinates of the Nodes

NODE=	X	Y	Z
1	0.00000	2.33000	0.00000
2	.04335	2.33000	-.00388
3	.17300	2.33000	-.01758
4	.38781	2.33000	-.04726
5	.68587	2.33000	-.10309
6	1.06448	2.33000	-.19941
7	0.00000	7.79120	0.00000
8	.04335	7.79120	-.00388
9	.17300	7.79120	-.01758
10	.38781	7.79120	-.04726
11	.68587	7.79120	-.10309
12	1.06448	7.79120	-.19941
13	0.00000	12.03880	0.00000
14	.04335	12.03880	-.00388
15	.17300	12.03880	-.01758
16	.38781	12.03880	-.04726
17	.68587	12.03880	-.10309
18	1.06448	12.03880	-.19941
19	0.00000	15.07280	0.00000
20	.04335	15.07280	-.00388
21	.17300	15.07280	-.01758
22	.38781	15.07280	-.04726
23	.68587	15.07280	-.10309
24	1.06448	15.07280	-.19941
25	0.00000	16.89320	0.00000
26	.04335	16.89320	-.00388
27	.17300	16.89320	-.01758
28	.38781	16.89320	-.04726
29	.68587	16.89320	-.10309
30	1.06448	16.89320	-.19941
31	0.00000	17.50000	0.00000
32	.04258	17.50000	-.00798
33	.17032	17.50000	-.03190
34	.38321	17.50000	-.07179
35	.68127	17.50000	-.12762
36	1.06448	17.50000	-.19941
37	.04131	2.33000	-.01207
38	.16763	2.33000	-.04623
39	.37862	2.33000	-.09632
40	.67667	2.33000	-.15215
41	.04181	7.79120	-.01207
42	.16763	7.79120	-.04623
43	.37862	7.79120	-.09632
44	.67667	7.79120	-.15215
45	.04181	12.03880	-.01207
46	.16763	12.03880	-.04623
47	.37862	12.03880	-.09632
48	.67667	12.03880	-.15215
49	.04181	15.07280	-.01207
50	.16763	15.07280	-.04623
51	.37862	15.07280	-.09632
52	.67667	15.07280	-.15215
53	.04181	16.89320	-.01207
54	.16763	16.89320	-.04623
55	.37862	16.89320	-.09632
56	.67667	16.89320	-.15215
57	-2.33000	.00000	0.00000
58	-2.33000	.04335	-.00388
59	-2.33000	.17300	-.01758
60	-2.33000	.38781	-.04726
61	-2.33000	.68587	-.10309
62	-2.33000	1.06448	-.19941
63	-7.79120	.00000	0.00000
64	-7.79120	.04335	-.00388
65	-7.79120	.17300	-.01758
66	-7.79120	.38781	-.04726

67	-7.79120	.68587	-.10309
68	-7.79120	1.06448	-.19941
69	-12.03880	.00000	0.00000
70	-12.03880	.04335	-.00388
71	-12.03880	.17300	-.01758
72	-12.03880	.38781	-.04726
73	-12.03880	.68587	-.10309
74	-12.03880	1.06448	-.19941
75	-15.07280	.00000	0.00000
76	-15.07280	.04335	-.00388
77	-15.07280	.17300	-.01758
78	-15.07280	.38781	-.04726
79	-15.07280	.68587	-.10309
80	-15.07280	1.06448	-.19941
81	-16.89320	.00000	0.00000
82	-16.89320	.04335	-.00388
83	-16.89320	.17300	-.01758
84	-16.89320	.38781	-.04726
85	-16.89320	.68587	-.10309
86	-16.89320	1.06448	-.19941
87	-17.50000	.00000	0.00000
88	-17.50000	.04258	-.00798
89	-17.50000	.17032	-.03190
90	-17.50000	.38321	-.07179
91	-17.50000	.58127	-.12762
92	-17.50000	1.06446	-.19941
93	-2.33000	.04181	-.01207
94	-2.33000	.16763	-.04623
95	-2.33000	.37862	-.09632
96	-2.33000	.67667	-.15215
97	-7.79120	.04181	-.01207
98	-7.79120	.16763	-.04623
99	-7.79120	.37862	-.09632
100	-7.79120	.67667	-.15215
101	-12.03880	.04181	-.01207
102	-12.03880	.16763	-.04623
103	-12.03880	.37862	-.09632
104	-12.03880	.67667	-.15215
105	-15.07280	.04181	-.01207
106	-15.07280	.16763	-.04623
107	-15.07280	.37862	-.09632
108	-15.07280	.67667	-.15215
109	-16.89320	.04181	-.01207
110	-16.89320	.16763	-.04623
111	-16.89320	.37862	-.09632
112	-16.89320	.67667	-.15215
113	-.00000	-2.33000	0.00000
114	-.04335	-2.33000	-.00388
115	-.17300	-2.33000	-.01758
116	-.38781	-2.33000	-.04726
117	-.68587	-2.33000	-.10309
118	-1.06448	-2.33000	-.19941
119	-.00000	-7.79120	0.00000
120	-.04335	-7.79120	-.00388
121	-.17300	-7.79120	-.01758
122	-.38781	-7.79120	-.04726
123	-.68587	-7.79120	-.10309
124	-1.06448	-7.79120	-.19941
125	-.00000	-12.03880	0.00000
126	-.04335	-12.03880	-.00388
127	-.17300	-12.03880	-.01758
128	-.38781	-12.03880	-.04726
129	-.68587	-12.03880	-.10309
130	-1.06448	-12.03880	-.19941
131	-.00000	-15.07280	0.00000
132	-.04335	-15.07280	-.00388
133	-.17300	-15.07280	-.01758
134	-.38781	-15.07280	-.04726
135	-.68587	-15.07280	-.10309
136	-1.06448	-15.07280	-.19941
137	-.00000	-16.89320	0.00000
138	-.04335	-16.89320	-.00388
139	-.17300	-16.89320	-.01758
140	-.38781	-16.89320	-.04726

141	-.68587	-16.89320	-.10309
142	-1.06448	-16.89320	-.19941
143	-.00000	-17.50000	0.00000
144	-.04258	-17.50000	-.00798
145	-.17032	-17.50000	-.03190
146	-.38321	-17.50000	-.07179
147	-.68127	-17.50000	-.12762
148	-1.06448	-17.50000	-.19941
149	-.04181	-2.33000	-.01207
150	-.16763	-2.33000	-.04623
151	-.37862	-2.33000	-.09632
152	-.67667	-2.33000	-.15215
153	-.04181	-7.79120	-.01207
154	-.16763	-7.79120	-.04623
155	-.37862	-7.79120	-.09632
156	-.67667	-7.79120	-.15215
157	-.04181	-12.03880	-.01207
158	-.16763	-12.03880	-.04623
159	-.37862	-12.03880	-.09632
160	-.67667	-12.03880	-.15215
161	-.04181	-15.07280	-.01207
162	-.16763	-15.07280	-.04623
163	-.37862	-15.07280	-.09632
164	-.67667	-15.07280	-.15215
165	-.04181	-16.89320	-.01207
166	-.16763	-16.89320	-.04623
167	-.37862	-16.89320	-.09632
168	-.67667	-16.89320	-.15215
169	2.33000	-.00000	0.00000
170	2.33000	-.04335	-.00388
171	2.33000	-.17300	-.01758
172	2.33000	-.38781	-.04726
173	2.33000	-.68587	-.10309
174	2.33000	-1.06448	-.19941
175	7.79120	-.00000	0.00000
176	7.79120	-.04335	-.00388
177	7.79120	-.17300	-.01758
178	7.79120	-.38781	-.04726
179	7.79120	-.68587	-.10309
180	7.79120	-1.06448	-.19941
181	12.03880	-.00000	0.00000
182	12.03880	-.04335	-.00388
183	12.03880	-.17300	-.01758
184	12.03880	-.38781	-.04726
185	12.03880	-.68587	-.10309
186	12.03880	-1.06448	-.19941
187	15.07280	-.00000	0.00000
188	15.07280	-.04335	-.00388
189	15.07280	-.17300	-.01758
190	15.07280	-.38781	-.04726
191	15.07280	-.68587	-.10309
192	15.07280	-1.06448	-.19941
193	16.89320	-.00000	0.00000
194	16.89320	-.04335	-.00388
195	16.89320	-.17300	-.01758
196	16.89320	-.38781	-.04726
197	16.89320	-.68587	-.10309
198	16.89320	-1.06448	-.19941
199	17.50000	-.00000	0.00000
200	17.50000	-.04258	-.00798
201	17.50000	-.17032	-.03190
202	17.50000	-.38321	-.07179
203	17.50000	-.68127	-.12762
204	17.50000	-1.06448	-.19941
205	2.33000	-.04181	-.01207
206	2.33000	-.16763	-.04623
207	2.33000	-.37862	-.09632
208	2.33000	-.67667	-.15215
209	7.79120	-.04181	-.01207
210	7.79120	-.16763	-.04623
211	7.79120	-.37862	-.09632
212	7.79120	-.67667	-.15215
213	12.03880	-.04181	-.01207
214	12.03880	-.16763	-.04623

215	12.03880	-.37862	-.09632
216	12.03860	-.67667	-.15215
217	15.07280	-.04191	-.01207
218	15.07280	-.16753	-.04623
219	15.07280	-.37862	-.09632
220	15.07230	-.67667	-.15215
221	16.89320	-.04181	-.01207
222	16.89320	-.16763	-.04623
223	16.89320	-.37862	-.09632
224	16.89320	-.67667	-.15215

X, Y, and Z Coordinates of the Centroid of Aerodynamic Panels

ELEM=	XPC	YPC	ZPC
1	.02167	5.06060	-.00194
2	.10817	5.06060	-.01073
3	.28041	5.06060	-.03242
4	.53684	5.06060	-.07517
5	.87517	5.06060	-.15125
6	.02167	9.91500	-.00194
7	.10817	9.91500	-.01073
8	.28041	9.91500	-.03242
9	.53684	9.91500	-.07517
10	.87517	9.91500	-.15125
11	.02167	13.55580	-.00194
12	.10817	13.55580	-.01073
13	.28041	13.55580	-.03242
14	.53684	13.55580	-.07517
15	.87517	13.55580	-.15125
16	.02167	15.98300	-.00194
17	.10817	15.98300	-.01073
18	.28041	15.98300	-.03242
19	.53684	15.98300	-.07517
20	.87517	15.98300	-.15125
21	.02148	17.19660	-.00296
22	.10731	17.19660	-.01534
23	.27859	17.19660	-.04213
24	.53454	17.19660	-.08744
25	.87403	17.19660	-.15738
26	.02091	5.06060	-.00604
27	.10472	5.06060	-.02915
28	.27313	5.06060	-.07127
29	.52765	5.06060	-.12423
30	.87058	5.06060	-.17578
31	.02091	9.91500	-.00604
32	.10472	9.91500	-.02915
33	.27313	9.91500	-.07127
34	.52765	9.91500	-.12423
35	.87058	9.91500	-.17578
36	.02091	13.55580	-.00604
37	.10472	13.55580	-.02915
38	.27313	13.55580	-.07127
39	.52765	13.55580	-.12423
40	.87058	13.55580	-.17578
41	.02091	15.98300	-.00604
42	.10472	15.98300	-.02915
43	.27313	15.98300	-.07127
44	.52765	15.98300	-.12423
45	.87058	15.98300	-.17578
46	.02110	17.19660	-.00501
47	.10559	17.19660	-.02455
48	.27455	17.19660	-.06156
49	.52994	17.19660	-.11197
50	.87173	17.19660	-.16964
51	-5.06060	.02167	-.00194
52	-5.06060	.10817	-.01073
53	-5.06060	.28041	-.03242
54	-5.06060	.53684	-.07517
55	-5.06060	.87517	-.15125
56	-9.91500	.02167	-.00194
57	-9.91500	.10817	-.01073
58	-9.91500	.28041	-.03242
59	-9.91500	.53684	-.07517
60	-9.91500	.87517	-.15125
61	-13.55580	.02167	-.00194
62	-13.55580	.10817	-.01073
63	-13.55580	.28041	-.03242
64	-13.55580	.53684	-.07517
65	-13.55580	.87517	-.15125
66	-15.98300	.02167	-.00194
67	-15.98300	.10817	-.01073
68	-15.98300	.28041	-.03242
69	-15.98300	.53684	-.07517
70	-15.98300	.87517	-.15125

71	-17.19660	.02148	-.00296
72	-17.19660	.10731	-.01534
73	-17.19660	.27859	-.04213
74	-17.19660	.53454	-.08744
75	-17.19660	.87403	-.15738
76	-5.06060	.02091	-.00604
77	-5.06060	.10472	-.02915
78	-5.06060	.27313	-.07127
79	-5.06060	.52765	-.12423
80	-5.06060	.87058	-.17578
81	-9.91500	.02091	-.00604
82	-9.91500	.10472	-.02915
83	-9.91500	.27313	-.07127
84	-9.91500	.52765	-.12423
85	-9.91500	.87058	-.17578
86	-13.55580	.02091	-.00604
87	-13.55580	.10472	-.02915
88	-13.55580	.27313	-.07127
89	-13.55580	.52765	-.12423
90	-13.55580	.87058	-.17578
91	-15.98300	.02091	-.00604
92	-15.98300	.10472	-.02915
93	-15.98300	.27313	-.07127
94	-15.98300	.52765	-.12423
95	-15.98300	.87058	-.17578
96	-17.19660	.02110	-.00501
97	-17.19660	.10559	-.02455
98	-17.19660	.27495	-.06156
99	-17.19660	.52994	-.11197
100	-17.19660	.87173	-.16964
101	-.02167	-5.06060	-.00194
102	-.10817	-5.06060	-.01073
103	-.28041	-5.06060	-.03242
104	-.53684	-5.06060	-.07517
105	-.87517	-5.06060	-.15125
106	-.02167	-9.91500	-.00194
107	-.10817	-9.91500	-.01073
108	-.28041	-9.91500	-.03242
109	-.53684	-9.91500	-.07517
110	-.87517	-9.91500	-.15125
111	-.02167	-13.55580	-.00194
112	-.10817	-13.55580	-.01073
113	-.28041	-13.55580	-.03242
114	-.53684	-13.55580	-.07517
115	-.87517	-13.55580	-.15125
116	-.02167	-15.98300	-.00194
117	-.10817	-15.98300	-.01073
118	-.28041	-15.98300	-.03242
119	-.53684	-15.98300	-.07517
120	-.87517	-15.98300	-.15125
121	-.02148	-17.19660	-.00296
122	-.10731	-17.19660	-.01534
123	-.27859	-17.19660	-.04213
124	-.53454	-17.19660	-.08744
125	-.87403	-17.19660	-.15738
126	-.02091	-5.06060	-.00604
127	-.10472	-5.06060	-.02915
128	-.27313	-5.06060	-.07127
129	-.52765	-5.06060	-.12423
130	-.87058	-5.06060	-.17578
131	-.02091	-9.91500	-.00604
132	-.10472	-9.91500	-.02915
133	-.27313	-9.91500	-.07127
134	-.52765	-9.91500	-.12423
135	-.87058	-9.91500	-.17578
136	-.02091	-13.55580	-.00604
137	-.10472	-13.55580	-.02915
138	-.27313	-13.55580	-.07127
139	-.52765	-13.55580	-.12423
140	-.87058	-13.55580	-.17578
141	-.02091	-15.98300	-.00604
142	-.10472	-15.98300	-.02915
143	-.27313	-15.98300	-.07127

144	-.52765	-15.98300	-.12423
145	-.87058	-15.98300	-.17578
146	-.02110	-17.19660	-.00501
147	-.10559	-17.19660	-.02455
148	-.27495	-17.19660	-.06156
149	-.52994	-17.19660	-.11197
150	-.87173	-17.19660	-.16964
151	5.06060	-.02167	-.00194
152	5.06060	-.10817	-.01073
153	5.06060	-.28041	-.03242
154	5.06060	-.53684	-.07517
155	5.06060	-.87517	-.15125
156	9.91500	-.02167	-.00194
157	9.91500	-.10817	-.01073
158	9.91500	-.28041	-.03242
159	9.91500	-.53684	-.07517
160	9.91500	-.87517	-.15125
161	13.55580	-.02167	-.00194
162	13.55580	-.10817	-.01073
163	13.55580	-.28041	-.03242
164	13.55580	-.53684	-.07517
165	13.55580	-.87517	-.15125
166	15.98300	-.02167	-.00194
167	15.98300	-.10817	-.01073
168	15.98300	-.28041	-.03242
169	15.98300	-.53684	-.07517
170	15.98300	-.87517	-.15125
171	17.19660	-.02148	-.00296
172	17.19660	-.10731	-.01534
173	17.19660	-.27859	-.04213
174	17.19660	-.53454	-.08744
175	17.19660	-.87403	-.15738
176	5.06060	-.02091	-.00604
177	5.06060	-.10472	-.02915
178	5.06060	-.27313	-.07127
179	5.06060	-.52765	-.12423
180	5.06060	-.87058	-.17578
181	9.91500	-.02091	-.00604
182	9.91500	-.10472	-.02915
183	9.91500	-.27313	-.07127
184	9.91500	-.52765	-.12423
185	9.91500	-.87058	-.17578
186	13.55580	-.02091	-.00604
187	13.55580	-.10472	-.02915
188	13.55580	-.27313	-.07127
189	13.55580	-.52765	-.12423
190	13.55580	-.87058	-.17578
191	15.98300	-.02091	-.00604
192	15.98300	-.10472	-.02915
193	15.98300	-.27313	-.07127
194	15.98300	-.52765	-.12423
195	15.98300	-.87058	-.17578
196	17.19660	-.02110	-.00501
197	17.19660	-.10559	-.02455
198	17.19660	-.27495	-.06156
199	17.19660	-.52994	-.11197
200	17.19660	-.87173	-.16964

Nodal Numbering for Surfaces

FCR SURFACE 24

1	7	13	19	25	31
2	8	14	20	26	32
3	9	15	21	27	33
4	10	16	22	28	34
5	11	17	23	29	35
6	12	18	24	30	36

FOR SURFACE 25

1	7	13	19	25	31
37	41	45	49	53	32
38	42	46	50	54	33
39	43	47	51	55	34
40	44	48	52	56	35
6	12	18	24	30	36

FCR SURFACE 27

57	63	69	75	81	87
58	64	70	76	82	88
59	65	71	77	33	89
60	66	72	78	84	90
61	67	73	79	85	91
62	68	74	80	36	92

FOR SURFACE 28

57	63	69	75	81	87
93	97	101	105	109	38
94	98	102	106	110	89
95	99	103	107	111	90
96	100	104	108	112	91
62	68	74	80	86	92

FCR SURFACE 30

113	119	125	131	137	143
114	120	126	132	138	144
115	121	127	133	139	145
116	122	128	134	140	146
117	123	129	135	141	147
118	124	130	136	142	148

FOR SURFACE 31

113	119	125	131	137	143
149	153	157	151	165	144
150	154	158	162	166	145
151	155	159	163	167	146
152	156	160	164	168	147
118	124	130	136	142	148

FOR SURFACE 33

169	175	181	187	193	199
170	176	182	188	194	200
171	177	183	189	195	201
172	178	184	190	196	202
173	179	185	191	197	203
174	180	186	192	198	204

FOR SURFACE 34

169	175	181	187	193	199
205	209	213	217	221	200
206	210	214	218	222	201
207	211	215	219	223	202
208	212	216	220	224	203
174	180	186	192	198	204

Nodal Numbering for Elements

ELEM	++	+-	-+	--
1	8	2	7	1
2	9	3	8	2
3	10	4	9	3
4	11	5	10	4
5	12	6	11	5
6	14	8	13	7
7	15	9	14	8
8	16	10	15	9
9	17	11	16	10
10	18	12	17	11
11	20	14	19	13
12	21	15	20	14
13	22	16	21	15
14	23	17	22	16
15	24	18	23	17
16	26	20	25	19
17	27	21	26	20
18	28	22	27	21
19	29	23	28	22
20	30	24	29	23
21	32	26	31	25
22	33	27	32	26
23	34	28	33	27
24	35	29	34	28
25	36	30	35	29
26	37	41	1	7
27	38	42	37	41
28	39	43	38	42
29	40	44	39	43
30	6	12	40	44
31	41	45	7	13
32	42	46	41	45
33	43	47	42	46
34	44	48	43	47
35	12	18	44	48
36	45	49	13	19
37	46	50	45	49
38	47	51	46	50
39	48	52	47	51
40	18	24	48	52
41	49	53	19	25
42	50	54	49	53
43	51	55	50	54
44	52	56	51	55
45	24	30	52	56
46	53	32	25	31
47	54	33	53	32
48	55	34	54	33
49	56	35	55	34
50	30	36	56	35
51	64	58	63	57
52	65	59	64	58
53	66	60	65	59
54	67	61	66	60
55	68	62	67	61
56	70	64	69	63
57	71	65	70	64
58	72	66	71	65
59	73	67	72	66
60	74	68	73	67
61	76	70	75	69
62	77	71	76	70
63	78	72	77	71
64	79	73	78	72
65	80	74	79	73
66	82	76	81	75
67	83	77	82	76
68	84	78	83	77

69	85	79	84	78
70	86	80	85	79
71	88	82	87	81
72	89	83	88	82
73	90	84	89	83
74	91	85	90	84
75	92	86	91	85
76	93	97	57	63
77	94	98	93	97
78	95	99	94	98
79	96	100	95	99
80	62	68	96	100
81	97	101	63	69
82	98	102	97	101
83	99	103	98	102
84	100	104	99	103
85	68	74	100	104
86	101	105	69	75
87	102	106	101	105
88	103	107	102	106
89	104	108	103	107
90	74	80	104	108
91	105	109	75	81
92	106	110	105	109
93	107	111	106	110
94	108	112	107	111
95	80	86	108	112
96	109	88	81	87
97	110	89	109	88
98	111	90	110	89
99	112	91	111	90
100	86	92	112	91
101	120	114	119	113
102	121	115	120	114
103	122	116	121	115
104	123	117	122	116
105	124	118	123	117
106	126	120	125	119
107	127	121	126	120
108	128	122	127	121
109	129	123	128	122
110	130	124	129	123
111	132	126	131	125
112	133	127	132	126
113	134	128	133	127
114	135	129	134	128
115	136	130	135	129
116	138	132	137	131
117	139	133	138	132
118	140	134	139	133
119	141	135	140	134
120	142	136	141	135
121	144	138	143	137
122	145	139	144	138
123	146	140	145	139
124	147	141	146	140
125	148	142	147	141
126	149	153	113	119
127	150	154	149	153
128	151	155	130	154
129	152	156	151	155
130	118	124	152	156
131	153	157	119	125
132	154	158	153	157
133	155	159	154	158
134	156	160	155	159
135	124	130	156	160
136	157	161	125	131
137	158	162	157	161
138	159	163	158	162
139	160	164	159	163
140	130	136	160	164
141	161	165	131	137

142	162	166	161	165
143	163	167	162	166
144	164	168	163	167
145	136	142	164	168
146	165	144	137	143
147	166	145	165	144
148	167	146	166	145
149	168	147	167	146
150	142	148	168	147
151	176	170	175	169
152	177	171	176	170
153	178	172	177	171
154	179	173	178	172
155	180	174	179	173
156	182	176	181	175
157	183	177	182	176
158	184	178	183	177
159	185	179	184	178
160	186	180	185	179
161	188	182	187	181
162	189	183	188	182
163	190	184	189	183
164	191	185	190	184
165	192	186	191	185
166	194	188	193	187
167	195	189	194	188
168	196	190	195	189
169	197	191	196	190
170	198	192	197	191
171	200	194	199	193
172	201	195	200	194
173	202	196	201	195
174	203	197	202	196
175	204	198	203	197
176	205	209	169	175
177	206	210	205	209
178	207	211	206	210
179	208	212	207	211
180	174	180	208	212
181	209	213	175	181
182	210	214	209	213
183	211	215	210	214
184	212	216	211	215
185	180	186	212	216
186	213	217	181	187
187	214	218	213	217
188	215	219	214	218
189	216	220	215	219
190	186	192	216	220
191	217	221	187	193
192	218	222	217	221
193	219	223	218	222
194	220	224	219	223
195	192	198	220	224
196	221	200	193	199
197	222	201	221	200
198	223	202	222	201
199	224	203	223	202
200	198	204	224	203

The Distribution of the Source

FOR SUBSURFACE 24

-.67898E+01	-.12965E+02	-.17601E+02	-.20237E+02	-.11351E+02
-.64984E+01	-.12450E+02	-.16917E+02	-.19516E+02	-.10964E+02
-.36705E+01	-.69826E+01	-.94592E+01	-.10879E+02	-.62031E+01
.20676E+01	.40146E+01	.54635E+01	.63976E+01	.33652E+01
.76145E+01	.14806E+02	.20137E+02	.23241E+02	.12750E+02

FOR SUBSURFACE 25

-.70504E+01	-.13521E+02	-.18387E+02	-.21175E+02	-.11844E+02
-.75670E+01	-.14784E+02	-.20220E+02	-.23463E+02	-.13102E+02
-.59701E+01	-.11792E+02	-.16203E+02	-.18950E+02	-.10664E+02
-.85895E+00	-.20238E+01	-.29820E+01	-.37337E+01	-.22639E+01
.60343E+01	.11586E+02	.15663E+02	.17933E+02	.98657E+01

FOR SUBSURFACE 27

-.67898E+01	-.12965E+02	-.17601E+02	-.20237E+02	-.11351E+02
-.64984E+01	-.12450E+02	-.16917E+02	-.19516E+02	-.10964E+02
-.36705E+01	-.69826E+01	-.94592E+01	-.10879E+02	-.62031E+01
.20676E+01	.40146E+01	.54635E+01	.63976E+01	.33652E+01
.76145E+01	.14806E+02	.20137E+02	.23241E+02	.12750E+02

FOR SUBSURFACE 28

-.70504E+01	-.13521E+02	-.18387E+02	-.21175E+02	-.11844E+02
-.75670E+01	-.14784E+02	-.20220E+02	-.23463E+02	-.13102E+02
-.59701E+01	-.11792E+02	-.16203E+02	-.18950E+02	-.10664E+02
-.85895E+00	-.20238E+01	-.29820E+01	-.37337E+01	-.22639E+01
.60343E+01	.11586E+02	.15663E+02	.17933E+02	.98657E+01

FOR SUBSURFACE 30

-.67898E+01	-.12965E+02	-.17601E+02	-.20237E+02	-.11351E+02
-.64984E+01	-.12450E+02	-.16917E+02	-.19516E+02	-.10964E+02
-.36705E+01	-.69326E+01	-.94592E+01	-.10879E+02	-.62031E+01
.20676E+01	.40148E+01	.54635E+01	.63976E+01	.33652E+01
.76145E+01	.14806E+02	.20137E+02	.23241E+02	.12750E+02

FOR SUBSURFACE 31

-.70504E+01	-.13521E+02	-.18387E+02	-.21175E+02	-.11844E+02
-.75670E+01	-.14784E+02	-.20220E+02	-.23463E+02	-.13102E+02
-.59701E+01	-.11792E+02	-.16203E+02	-.18950E+02	-.10664E+02
-.85895E+00	-.20238E+01	-.29820E+01	-.37337E+01	-.22639E+01
.60343E+01	.11586E+02	.15663E+02	.17933E+02	.98657E+01

FOR SUBSURFACE 33

-.67898E+01	-.12965E+02	-.17601E+02	-.20237E+02	-.11351E+02
-.64984E+01	-.12450E+02	-.16917E+02	-.19516E+02	-.10964E+02
-.36705E+01	-.69326E+01	-.94592E+01	-.10879E+02	-.62031E+01
.20676E+01	.40148E+01	.54635E+01	.63976E+01	.33652E+01
.76145E+01	.14806E+02	.20137E+02	.23241E+02	.12750E+02

FOR SUBSURFACE 34

-.70504E+01	-.13521E+02	-.18387E+02	-.21175E+02	-.11844E+02
-.75670E+01	-.14784E+02	-.20220E+02	-.23463E+02	-.13102E+02
-.59701E+01	-.11792E+02	-.16203E+02	-.18950E+02	-.10664E+02
-.85895E+00	-.20238E+01	-.29820E+01	-.37337E+01	-.22639E+01
.60343E+01	.11586E+02	.15663E+02	.17933E+02	.98657E+01

The Distribution of the Velocity Potential

FOR SUBSURFACE 24

.17602E+03	.20640E+03	.20732E+03	.18068E+03	.15468E+03
.17886E+03	.21339E+03	.21796E+03	.19379E+03	.16714E+03
.18380E+03	.22322E+03	.23153E+03	.20961E+03	.17963E+03
.18929E+03	.23356E+03	.24566E+03	.22568E+03	.18934E+03
.18643E+03	.23073E+03	.24566E+03	.22257E+03	.18346E+03

FOR SUBSURFACE 25

.16041E+03	.17883E+03	.17898E+03	.13619E+03	.11212E+03
.15183E+03	.16619E+03	.15586E+03	.11874E+03	.95400E+02
.14552E+03	.15728E+03	.14563E+03	.10694E+03	.82754E+02
.14380E+03	.15562E+03	.14404E+03	.10571E+03	.78885E+02
.14135E+03	.15756E+03	.14451E+03	.10988E+03	.82064E+02

FOR SUBSURFACE 27

.17602E+03	.20640E+03	.20732E+03	.18068E+03	.15468E+03
.17886E+03	.21339E+03	.21798E+03	.19379E+03	.16714E+03
.18380E+03	.22322E+03	.23153E+03	.20961E+03	.17963E+03
.18929E+03	.23356E+03	.24566E+03	.22568E+03	.18934E+03
.18643E+03	.23073E+03	.24566E+03	.22257E+03	.18346E+03

FOR SUBSURFACE 28

.16041E+03	.17883E+03	.17898E+03	.13619E+03	.11212E+03
.15183E+03	.16619E+03	.15586E+03	.11874E+03	.95400E+02
.14552E+03	.15728E+03	.14563E+03	.10694E+03	.82754E+02
.14380E+03	.15562E+03	.14404E+03	.10571E+03	.78885E+02
.14135E+03	.15756E+03	.14451E+03	.10988E+03	.82064E+02

FOR SUBSURFACE 30

.17602E+03	.20641E+03	.20732E+03	.18068E+03	.15468E+03
.17886E+03	.21339E+03	.21798E+03	.19379E+03	.16714E+03
.18380E+03	.22322E+03	.23153E+03	.20961E+03	.17963E+03
.18929E+03	.23356E+03	.24566E+03	.22568E+03	.18934E+03
.18643E+03	.23073E+03	.24566E+03	.22257E+03	.18346E+03

FOR SUBSURFACE 31

.16041E+03	.17883E+03	.17098E+03	.13619E+03	.11212E+03
.15183E+03	.16619E+03	.15586E+03	.11874E+03	.95400E+02
.14552E+03	.15728E+03	.14563E+03	.10694E+03	.82754E+02
.14360E+03	.15562E+03	.14404E+03	.10571E+03	.78685E+02
.14135E+03	.15756E+03	.14451E+03	.10988E+03	.82064E+02

FOR SUBSURFACE 33

.17602E+03	.20540E+03	.20732E+03	.18068E+03	.15468E+03
.17886E+03	.21339E+03	.21798E+03	.19379E+03	.16714E+03
.18380E+03	.22322E+03	.23193E+03	.20961E+03	.17963E+03
.18929E+03	.23356E+03	.24566E+03	.22568E+03	.18934E+03
.18643E+03	.23073E+03	.24966E+03	.22297E+03	.18346E+03

FOR SUBSURFACE 34

.16041E+03	.17883E+03	.17098E+03	.13619E+03	.11212E+03
.15183E+03	.16619E+03	.15586E+03	.11874E+03	.95400E+02
.14552E+03	.15728E+03	.14563E+03	.10694E+03	.82754E+02
.14380E+03	.15562E+03	.14404E+03	.10571E+03	.78685E+02
.14135E+03	.15756E+03	.14451E+03	.10988E+03	.82064E+02

Pressure Distribution

FOR SUBSURFACE 24

-.10193E+01	-.18540E+01	-.29055E+01	-.36553E+01	-.18860E+01
-.15173E+00	-.30130E+00	-.48997E+00	-.60060E+00	-.28944E+00
-.11633E+00	-.21550E+00	-.33475E+00	-.38244E+00	-.21574E+00
-.24355E-01	-.61409E-01	-.10785E+00	-.10601E+00	-.65347E-01
.27068E+00	.45045E+00	.66623E+00	.79997E+00	.41744E+00

FOR SUBSURFACE 25

.12332E+01	.20739E+01	.30870E+01	.38080E+01	.21446E+01
.24285E+00	.37140E+00	.51221E+00	.61583E+00	.35977E+00
.77155E-01	.11078E+00	.14419E+00	.17862E+00	.82484E-01
.19756E-01	.92368E-02	-.29301E-02	-.11323E-01	-.10274E-01
-.25524E+00	-.45014E+00	-.68576E+00	-.82448E+00	-.41975E+00

FOR SUBSURFACE 27

-.22063E+00	-.66035E+00	-.12827E+01	-.17903E+01	-.10721E+01
-.24219E-01	-.94383E-01	-.22389E+00	-.33037E+00	-.39245E+00
-.14865E-01	-.60488E-01	-.15311E+00	-.22746E+00	-.42499E+00
.69449E-02	-.19458E-02	-.51913E-01	-.99233E-01	-.39249E+00
.71613E-01	.17953E+00	.28145E+00	.32919E+00	-.22828E-01

FOR SUBSURFACE 28

.28237E+00	.75515E+00	.13274E+01	.17690E+01	.10734E+01
.58637E-01	.13832E+00	.19970E+00	.24258E+00	.29758E+00
.20845E-01	.40475E-01	.37636E-01	.31767E-01	.21774E+00
.81468E-02	.36365E-02	-.26761E-01	-.60218E-01	.19615E+00
-.50783E-01	-.15666E+00	-.31931E+00	-.44820E+00	-.14959E+00

FOR SUBSURFACE 30

.56203E+00	.51296E+00	.36899E+00	.16555E+00	.16322E-01
.03755E-01	.83474E-01	.62159E-01	.26930E-01	.99553E-03
.64417E-01	.59976E-01	.42390E-01	.16581E-01	-.31805E-02
.14005E-01	.17769E-01	.13533E-01	.32679E-02	-.10356E-01
-.14844E+00	-.12360E+00	-.85126E-01	-.38927E-01	-.14940E-01

FOR SUBSURFACE 31

-.67996E+00	-.57377E+00	-.39208E+00	-.17259E+00	-.18741E-01
-.13386E+00	-.10274E+00	-.65213E-01	-.28222E-01	-.24223E-02
-.42457E-01	-.30642E-01	-.18754E-01	-.89582E-02	.20596E-02
-.10723E-01	-.25429E-02	-.47929E-03	-.11278E-02	.61449E-02
.14121E+00	.12492E+00	.86005E-01	.34585E-01	.65353E-02

FOR SUBSURFACE 33

-.23663E+00	-.68066E+00	-.12539E+01	-.16995E+01	-.79758E+00
-.43758E-01	-.12314E+00	-.20396E+00	-.24330E+00	.10400E+00
-.37047E-01	-.95035E-01	-.13926E+00	-.13840E+00	.20607E+00
-.17294E-01	-.41695E-01	-.42407E-01	-.35046E-02	.31679E+00
.50638E-01	.14732E+00	.29965E+00	.43185E+00	.42533E+00

FOR SUBSURFACE 34

.27089E+00	.74493E+00	.13675E+01	.18664E+01	.10525E+01
.50350E-01	.13253E+00	.24729E+00	.34503E+00	.59768E-01
.13853E-01	.39664E-01	.87800E-01	.13789E+00	-.13319E+00
.88585E-03	.30078E-02	.23352E-01	.47707E-01	-.20028E+00
-.63241E-01	-.16356E+00	-.28045E+00	-.34170E+00	-.26362E+00

SAMPLE CASE 3:

BO-105 FUSELAGE (NO PYLON, NO ROTOR) WITH EFFECTS OF SEPARATION

Data Input

```
  3
PROGRAM SHAPES
HUB PYLON FLOW SEPARATION
FUSELAGE - WITH SEPARATION EFFECTS
KREAD      0
*         12      0      0
**         0      1      1      1      0      0      0      0      0
***        0      0
UAMCH          .204       0.000         -0.000
ALFA          0.000       0.000
NBODY         0.000      12.000         38.000        41.000
NBODY         0.000       0.000          0.000         0.000
NBODY         0.000       0.000          0.000         0.000
NBODY         7.250       8.250          7.250         8.250
NWAKE      1      0      0      0
NPYLON     1      1
CSTAG    -20.000        1.000
KPRINT     1      1      1      1      0      1      1      1      1      1
NBODY1     4      4      2      1      1
NBODY2     4      4      1      1      1
NBODY3     4      4      2      1      1
```

Specifications of the Problem

```
FOR PART  1
NX= 4
NY= 4

FOR PART  2
NX= 4
NY= 4

FOR PART  3
NX= 4
NY= 4

FOR PART  5
NX= 4
NY= 4

FOR PART  6
NX= 4
NY= 4

FOR PART  7
NX= 4
NY= 4

FOR PART  9
NX= 4
NY= 4

FOR PART 10
NX= 4
NY= 4

FOR PART 11
NX= 4
NY= 4

FOR PART 13
NX= 4
NY= 4

FOR PART 14
NX= 4
NY= 4

FOR PART 15
NX= 4
NY= 4

NELEM=132

KSYMMY= 0
KSYMMZ= 0
```

X, Y, and Z Coordinates of the Nodes

NODE=	X	Y	Z
1	0.00000	0.00000	0.00000
2	.75000	-.00000	2.06250
3	3.00000	-.00000	4.12500
4	6.75000	-.00000	6.18750
5	12.00000	.00000	8.25000
6	.75000	.59351	1.90550
7	3.00000	1.36723	3.61100
8	6.75000	2.08084	5.71650
9	12.00000	2.77445	7.62201
10	.75000	1.29153	1.45841
11	3.00000	2.58326	2.91682
12	6.75000	3.84469	4.37522
13	12.00000	5.12652	5.83363
14	.75000	1.57453	.78928
15	3.00000	3.34906	1.57857
16	6.75000	5.02359	2.36785
17	12.00000	6.59813	3.15714
18	.75000	1.81250	0.00000
19	3.00000	3.62500	0.00000
20	6.75000	5.43750	0.00000
21	12.00000	7.25000	0.00000
22	18.50000	-.00000	8.25000
23	25.00000	-.00000	8.25000
24	31.50000	-.00000	8.25000
25	[illegible]	[illegible]	[illegible]
26	18.50000	2.77445	7.62201
27	25.00000	2.77445	7.62201
28	31.50000	2.77445	7.62201
29	38.00000	2.77445	7.62201
30	18.50000	5.12652	5.83363
31	25.00000	5.12652	5.83363
32	31.50000	5.12652	5.83363
33	38.00000	5.12652	5.83363
34	18.50000	6.59813	3.15714
35	25.00000	6.59813	3.15714
36	31.50000	6.59813	3.15714
37	38.00000	6.59813	3.15714
38	18.50000	7.25000	0.00000
39	25.00000	7.25000	[illegible]
40	31.50000	7.25000	0.00000
41	38.00000	7.25000	0.00000
42	39.31250	-.00000	6.18750
43	40.25000	-.00000	4.12500
44	40.81250	-.00000	2.06250
45	41.00000	0.00000	0.00000
46	39.31250	2.08084	5.71650
47	40.25000	1.36723	3.81100
48	40.81250	.59351	1.90550
49	39.31250	3.84469	4.37522
50	40.25000	2.58326	2.91682
51	40.81250	1.29153	1.45841
52	39.31250	5.02359	2.36785
53	40.25000	3.34906	1.57857
54	40.81250	1.67453	.78928
55	39.31250	5.43750	0.00000
56	40.25000	3.62500	0.00000
57	40.8125	1.81250	[illegible]
58	.75000	-.00000	-2.06250
59	3.00000	-.00000	-4.12500
60	6.75000	-.00000	-6.18750
61	12.00000	.00000	-8.25000
62	.75000	.59351	-1.90550
63	3.00000	1.36723	-3.81100
64	6.75000	2.08084	-5.71650
65	12.00000	2.77445	-7.62201
66	.75[illegible]	1.29153	-1.45841
67	3.00000	2.58326	-2.91682
68	6.75000	3.84469	-4.37522

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

69	12.00000	5.12552	-5.83363
70	.75000	1.57453	-.78928
71	3.00000	3.34906	-1.57857
72	6.75000	5.02359	-2.36785
73	12.00000	6.59813	-3.15714
74	18.50000	-.00000	-8.25000
75	25.00000	[illegible]	[illegible]
76	31.50000	-.00000	-8.25000
77	38.00000	.00000	-8.25000
78	18.50000	2.77445	-7.62201
79	25.00000	2.77445	-7.62201
80	31.50000	2.77445	-7.62201
81	38.00000	2.77445	-7.62201
82	18.50000	5.12552	-5.83363
83	25.00000	5.12552	-5.83363
84	31.50000	5.12552	-5.83363
85	38.00000	5.12552	-5.83363
86	18.50000	6.59813	-3.15714
87	25.00000	6.59813	-3.15714
88	31.50000	6.59813	-3.15714
89	38.00000	6.59813	-3.15714
90	39.31250	-.00000	-6.18750
91	40.25000	-.00000	-4.12500
92	40.81250	-.00000	-2.06250
93	39.31250	2.08064	-5.71650
94	40.25000	1.38723	-3.81100
95	40.81250	.69361	-1.90550
96	39.31250	3.34459	-4.37522
97	40.25000	2.58326	-2.91682
98	40.81250	1.29163	-1.45841
99	39.31250	5.02359	-2.36785
100	40.25000	3.34906	-1.57857
101	40.81250	1.67453	-.78928
102	.75000	-.69361	1.90550
103	3.00000	-1.38723	3.81100
104	6.75000	-2.08064	5.71650
105	12.00000	-2.77445	7.62201
106	.75000	-1.29163	1.45841
107	3.00000	-2.58326	2.91682
108	6.75000	-3.34459	4.37522
109	12.00000	-5.12552	5.83363
110	.75000	-1.57453	.78928
111	3.00000	-3.34906	1.57857
112	6.75000	-5.02359	2.36785
113	12.00000	-6.59813	3.15714
114	.75000	-1.81250	0.00000
115	3.00000	-3.62500	0.00000
116	6.75000	-5.43750	0.00000
117	12.00000	-7.25000	0.00000
118	18.50000	-2.77445	7.62201
119	25.00000	-2.77445	7.62201
120	31.50000	-2.77445	7.62201
121	38.00000	-2.77445	7.62201
122	18.50000	-5.12552	5.83363
123	25.00000	-5.12552	5.83363
124	31.50000	-5.12552	5.83363
125	38.00000	-5.12552	5.83363
126	18.50000	-6.59813	3.15714
127	25.00000	-6.59813	3.15714
128	31.50000	-6.59813	3.15714
129	38.00000	-6.59813	3.15714
130	18.50000	-7.25000	0.00000
131	25.00000	-7.25000	0.00000
132	31.50000	-7.25000	0.00000
133	38.00000	-7.25000	0.00000
134	39.31250	-2.08064	5.71650
135	40.25000	-1.38723	3.81100
136	40.81250	-.69361	1.90550
137	39.31250	-3.34459	4.37522
138	40.25000	-2.58326	2.91682
139	40.81250	-1.29163	1.45841
140	39.31250	-5.02359	2.36785

141	40.25000	-3.34906	1.57657
142	40.81250	-1.57453	.78928
143	39.31250	-5.43750	0.00000
144	40.25000	-3.52500	0.00000
145	40.81250	-1.31250	0.00000
146	.75000	-.59361	-1.90550
147	3.00000	-1.38723	-3.81100
148	6.75000	-2.08034	-5.71650
149	12.00000	-2.77445	-7.62201
150	.75000	-1.28163	-1.45841
151	3.00000	-2.55326	-2.91682
152	6.75000	-3.84489	-4.37522
153	12.00000	-5.12652	-5.83363
154	.75000	-1.57453	-.78928
155	3.00000	-3.34906	-1.57657
156	6.75000	-5.02359	-2.36785
157	12.00000	-6.69813	-3.15714
158	18.50000	-2.77445	-7.62201
159	25.00000	-2.77445	-7.62201
160	31.50000	-2.77445	-7.62201
161	38.00000	-2.77445	-7.62201
162	18.50000	-5.12652	-5.83363
163	25.00000	-5.12652	-5.83363
164	31.50000	-5.12652	-5.83363
165	38.00000	-5.12652	-5.83363
166	18.50000	-6.69813	-3.15714
167	25.00000	-6.69813	-3.15714
168	31.50000	-6.69813	-3.15714
169	38.00000	-6.69813	-3.15714
170	39.31250	-2.08034	-5.71650
171	40.25000	-1.38723	-3.81100
172	40.81250	-.59361	-1.90550
173	39.31250	-3.84489	-4.37522
174	40.25000	-2.55326	-2.91682
175	40.81250	-1.28163	-1.45841
176	39.31250	-5.02359	-2.36785
177	40.25000	-3.34906	-1.57657
178	40.81250	-1.57453	-.78928

X, Y, and Z Coordinates of the Centroid of Aerodynamic Panels

ELEM=	XPC	YPC	ZPC
1	.38306	.17340	.99200
2	1.91528	.52021	2.97600
3	4.97972	.35702	4.95000
4	9.57638	1.21382	5.94400
5	.38306	.49381	.84098
6	1.91528	1.48143	2.52293
7	4.97972	2.46905	4.20489
8	9.57638	3.49658	5.33664
9	.38306	.73924	.56192
10	1.91528	2.21712	1.68577
11	4.97972	3.69520	2.80962
12	9.57638	5.17325	3.93365
13	.38306	.87175	.19732
14	1.91528	2.61527	.59195
15	4.97972	4.35879	.98661
16	9.57638	6.10231	1.38125
17	15.57758	1.38723	7.93600
18	22.21721	1.38723	7.93600
19	28.85683	1.38723	7.93600
20	35.49646	1.38723	7.93600
21	15.57758	3.95049	6.72782
22	22.21721	3.95049	6.72782
23	28.85683	3.95049	6.72782
24	35.49646	3.95049	6.72782
25	15.57758	5.91233	4.49538
26	22.21721	5.91233	4.49538
27	28.85683	5.91233	4.49538
28	35.49646	5.91233	4.49538
29	15.57758	6.97406	1.57857
30	22.21721	6.97406	1.57857
31	28.85683	6.97406	1.57857
32	35.49646	6.97406	1.57857
33	39.48662	1.21382	5.94400
34	40.63578	.85702	4.96000
35	41.40190	.52021	2.97600
36	41.78495	.17340	.99200
37	39.48662	3.49658	5.33664
38	40.63578	2.46905	4.20489
39	41.40190	1.48143	2.52293
40	41.78495	.49381	.84098
41	39.48662	5.17325	3.93365
42	40.63578	3.69520	2.80962
43	41.40190	2.21712	1.68577
44	41.78495	.73924	.56192
45	39.48662	6.10231	1.38125
46	40.63578	4.35879	.98661
47	41.40190	2.61527	.59195
48	41.78495	.87175	.19732
49	.38306	.17340	-.99200
50	1.91528	.52021	-2.97600
51	4.97972	.85702	-4.95000
52	9.57638	1.21382	-5.94400
53	.38306	.49381	-.84098
54	1.91528	1.48143	-2.52293
55	4.97972	2.46905	-4.20489
56	9.57638	3.49658	-5.33664
57	.38306	.73924	-.56192
58	1.91528	2.21712	-1.68577
59	4.97972	3.69520	-2.80962
60	9.57638	5.17325	-3.93365
61	.38306	.87175	-.19732
62	1.91528	2.61527	-.59195
63	4.97972	4.35879	-.98661
64	9.57638	6.10231	-1.38125

65	15.57758	1.33723	-7.93600
66	22.21721	1.38723	-7.93600
67	28.85683	1.38723	-7.93600
68	35.49646	1.38723	-7.93600
69	15.57758	3.95049	-6.72782
70	22.21721	3.95049	-6.72782
71	28.85683	3.95049	-6.72782
72	35.49646	3.95049	-6.72782
73	15.57758	5.31233	-4.49538
74	22.21721	5.31233	-4.49538
75	28.85683	5.31233	-4.49538
76	35.49646	5.31233	-4.49538
77	15.57758	6.97406	-1.57857
78	22.21721	6.97405	-1.57857
79	28.85683	6.97406	-1.57857
80	35.49646	6.97406	-1.57857
81	39.48652	1.21352	-5.94400
82	40.63578	.34702	-4.96000
83	41.40190	.52021	-2.97500
84	41.78495	.17340	-.99200
85	39.48652	3.45650	-3.83685
86	40.63578	2.46905	-4.20489
87	41.40190	1.45143	-2.92293
88	41.78495	.49331	-.84093
89	39.48652	5.17325	-3.93345
90	40.63578	3.69520	-2.86942
91	41.40190	2.21712	-1.68577
92	41.78495	.73904	-.56192
93	39.48652	5.10231	-1.30125
94	40.63578	4.33679	-.98601
95	41.40190	2.61527	-.59195
96	41.78495	.87176	-.19732
97	.38306	-.17340	.99200
98	1.91528	-.52021	2.97600
99	4.97972	-.86702	4.96000
100	9.57638	-1.21382	5.94400
101	.38306	-.49331	.84093
102	1.91528	-1.48143	2.92293
103	4.97972	-2.46905	4.20489
104	9.57638	-3.45650	5.33684
105	.38306	-.73904	.56192
106	1.91528	-2.21712	1.68577
107	4.97972	-3.69520	2.86942
108	9.57538	-5.17323	3.93345
109	.38306	-.87176	.19732
110	1.91528	-2.61527	.59195
111	4.97972	-4.33679	.98601
112	9.57538	-5.10231	1.30125
113	15.57758	-1.38723	7.93600
114	22.21721	-1.38723	7.93600
115	28.85683	-1.38723	7.93600
116	35.49646	-1.38723	7.93600
117	15.57758	-3.95049	6.72782
118	22.21721	-3.95049	6.72782
119	28.85683	-3.95049	6.72782
120	35.49646	-3.95049	6.72782
121	15.57758	-5.31233	4.49538
122	22.21721	-5.31233	4.49538
123	28.85683	-5.31233	4.49538
124	35.49646	-5.31233	4.49535
125	15.57758	-6.97405	1.57857
126	22.21721	-6.97405	1.57857
127	28.85683	-6.97406	1.57857
128	35.49646	-6.97406	1.57857
129	39.48652	-1.21352	5.94400
130	40.63578	-.86702	4.96000
131	41.40190	-.52021	2.97600
132	41.78495	-.17340	.99200
133	39.48662	-3.45639	5.33685
134	40.63578	-2.46905	4.20489
135	41.40190	-1.45143	2.92293

136	41.78495	-.49331	.84098
137	39.48662	-5.17328	3.93346
138	40.63578	-3.69520	2.80962
139	41.40190	-2.21712	1.68577
140	41.78495	-.73904	.56192
141	39.48662	-6.10231	1.38125
142	40.63578	-4.35879	.98661
143	41.40190	-2.61527	.59196
144	41.78495	-.87175	.19732
145	.38306	-.17340	-.99200
146	1.91528	-.52021	-2.97600
147	4.97972	-.86702	-4.96000
148	9.57638	-1.21382	-6.94400
149	.38306	-.49331	-.84098
150	1.91528	-1.48143	-2.52293
151	4.97972	-2.46906	-4.20489
152	9.57638	-3.45668	-5.88684
153	.38306	-.73904	-.56192
154	1.91528	-2.21712	-1.68577
155	4.97972	-3.69520	-2.80962
156	9.57638	-5.17328	-3.93346
157	.38306	-.87175	-.19732
158	1.91528	-2.61527	-.59196
159	4.97972	-4.35879	-.98661
160	9.57638	-6.10231	-1.38125
161	15.57758	-1.38723	-7.93600
162	22.21721	-1.38723	-7.93600
163	28.85683	-1.38723	-7.93600
164	35.49646	-1.38723	-7.93600
165	15.57758	-3.95049	-6.72782
166	22.21721	-3.95049	-6.72782
167	28.85683	-3.95049	-6.72782
168	35.49646	-3.95049	-6.72782
169	15.57758	-5.91233	-4.49538
170	22.21721	-5.91233	-4.49538
171	28.85683	-5.91233	-4.49538
172	35.49646	-5.91233	-4.49538
173	15.57758	-6.97405	-1.57857
174	22.21721	-6.97405	-1.57857
175	28.85683	-6.97405	-1.57857
176	35.49646	-6.97405	-1.57857
177	39.48662	-1.21382	-6.94400
178	40.63578	-.86702	-4.96000
179	41.40190	-.52021	-2.97600
180	41.78495	-.17340	-.99200
181	39.48662	-3.45668	-5.88684
182	40.63578	-2.46906	-4.20489
183	41.40190	-1.48143	-2.52293
184	41.78495	-.49331	-.84098
185	39.48662	-5.17328	-3.93346
186	40.63578	-3.69520	-2.80962
187	41.40190	-2.21712	-1.68577
188	41.78495	-.73904	-.56192
189	39.48662	-6.10231	-1.38125
190	40.63578	-4.35879	-.98661
191	41.40190	-2.61527	-.59196
192	41.78495	-.87175	-.19732

Nodal Numbering for Surfaces

FOR SURFACE 1

1	1	1	1	1
2	6	10	14	18
3	7	11	15	19
4	8	12	16	20
5	9	13	17	21

FOR SURFACE 2

5	9	13	17	21
22	26	30	34	38
23	27	31	35	39
24	28	32	36	40
25	29	33	37	41

FOR SURFACE 3

25	29	33	37	41
42	46	49	52	55
43	47	50	53	56
44	48	51	54	57
45	45	45	45	45

FOR SURFACE 5

1	1	1	1	1
58	62	66	70	18
59	63	67	71	19
60	64	68	72	20
61	65	69	73	21

FOR SURFACE 6

61	65	69	73	21
74	78	82	86	38
75	79	83	87	39
76	80	84	88	40
77	81	85	89	41

FOR SURFACE 7

77	81	85	89	41
90	93	96	99	55
91	94	97	100	56
92	95	98	101	57
45	45	45	45	45

FOR SURFACE 9

1	1	1	1	1
2	102	106	110	114
3	103	107	111	115
4	104	108	112	116
5	105	109	113	117

FOR SURFACE 10

5	105	109	113	117
22	118	122	126	130
23	119	123	127	131
24	120	124	128	132
25	121	125	129	133

FOR SURFACE 11

25	121	125	129	133
42	134	137	140	143
43	135	138	141	144
44	136	139	142	145
45	45	45	45	45

FOR SURFACE 13

1	1	1	1	1
58	146	150	154	114
59	147	151	155	115
60	148	152	156	116
61	149	153	157	117

FOR SURFACE 14

61	149	153	157	117
74	158	162	166	130
75	159	163	167	131
76	160	164	168	132
77	161	165	169	133

FOR SURFACE 15

77	161	165	169	133
90	170	173	176	143
91	171	174	177	144
92	172	175	178	145
45	45	45	45	45

Nodal Numbering for Elements

ELEM	++	+-	-+	--
1	[illegible]	2	1	1
2	7	3	5	2
3	8	4	7	3
4	9	5	8	4
5	10	6	1	1
6	11	7	10	6
7	12	8	11	7
8	13	9	12	8
9	14	10	1	1
10	15	11	14	10
11	16	12	15	11
12	17	13	16	12
13	18	14	1	1
14	19	15	18	14
15	20	16	19	15
16	21	17	20	16
17	26	22	3	5
18	27	23	25	22
19	28	24	27	23
20	29	25	28	24
21	30	26	13	9
22	31	27	30	26
23	32	28	31	27
24	33	29	32	28
25	34	30	17	13
26	35	31	34	30
27	36	32	35	31
28	37	33	36	32
29	38	34	21	17
30	39	36	38	34
31	40	35	39	35
32	41	37	40	36
33	46	42	25	25
34	47	43	46	42
35	48	44	47	43
36	45	45	49	44
37	49	46	33	29
38	50	47	49	46
39	51	48	50	47
40	45	45	51	48
41	52	49	37	33
42	53	50	52	49
43	54	51	53	50
44	45	45	54	51
45	55	52	41	37
46	56	53	55	52
47	57	54	56	53
48	45	45	57	54
49	58	62	1	1
50	59	63	58	62
51	60	64	59	63
52	61	65	60	64
53	62	66	1	1
54	63	67	62	66
55	64	68	63	67
56	65	69	64	68
57	66	70	1	1
58	67	71	66	70
59	68	72	67	71
60	69	73	68	72
61	70	18	1	1
62	71	19	70	18
63	72	20	71	19
64	73	21	72	20

65	74	78	61	[illegible]
66	75	79	74	79
67	76	80	75	[illegible]
68	77	81	76	80
69	78	82	[illegible]	[illegible]
70	79	83	78	82
71	[illegible]	84	79	83
72	81	[illegible]	80	84
73	82	86	[illegible]	73
74	83	87	32	[illegible]
75	84	[illegible]	[illegible]	[illegible]
76	85	89	[illegible]	[illegible]
77	[illegible]	[illegible]	[illegible]	21
78	87	[illegible]	[illegible]	[illegible]
79	88	40	[illegible]	[illegible]
80	89	41	38	40
81	90	43	[illegible]	41
82	91	[illegible]	[illegible]	[illegible]
83	92	[illegible]	31	[illegible]
84	45	45	32	[illegible]
85	93	96	31	[illegible]
86	94	97	33	[illegible]
87	95	[illegible]	34	37
88	45	45	[illegible]	[illegible]
89	[illegible]	99	[illegible]	[illegible]
90	97	100	96	99
91	99	101	97	100
92	45	45	98	101
93	99	[illegible]	[illegible]	91
94	100	[illegible]	[illegible]	[illegible]
95	101	[illegible]	100	[illegible]
96	45	45	101	[illegible]
97	2	102	1	1
98	3	103	2	102
99	4	104	3	103
100	5	105	4	104
101	102	106	1	1
102	103	107	102	106
103	104	108	103	107
104	105	109	104	[illegible]
105	106	110	1	1
106	107	111	106	110
107	108	112	107	111
108	109	113	108	112
109	110	114	1	1
110	111	115	110	114
111	112	116	111	115
112	113	117	112	116
113	22	118	5	105
114	23	119	22	118
115	24	120	23	119
116	25	121	24	120
117	118	122	105	109
118	119	123	118	122
119	120	124	119	123
120	121	125	120	124
121	122	126	119	113
122	123	127	122	125
123	124	128	123	127
124	125	129	124	128
125	125	130	113	117
126	127	131	126	130
127	128	132	127	131
128	129	133	128	132

129	42	134	25	121
130	43	135	42	134
131	44	136	43	135
132	45	45	44	136
133	134	137	121	125
134	135	138	134	137
135	136	139	135	138
136	45	45	136	139
137	137	140	125	129
138	138	141	137	140
139	139	142	138	141
140	45	45	139	142
141	140	143	121	133
142	141	144	140	143
143	142	145	141	144
144	45	45	142	145
145	146	58	1	1
146	147	59	146	58
147	148	60	147	59
148	149	61	148	60
149	150	146	1	1
150	151	147	150	146
151	152	148	151	147
152	153	149	152	148
153	154	150	1	1
154	155	151	154	150
155	156	152	155	151
156	157	153	156	152
157	114	154	1	1
158	113	155	114	154
159	115	156	115	155
160	117	157	116	156
161	158	74	143	61
162	159	75	158	74
163	160	76	159	75
164	161	77	150	76
165	162	158	153	149
166	163	159	152	158
167	164	160	153	159
168	165	161	164	160
169	166	162	157	153
170	167	163	155	162
171	168	164	157	163
172	169	165	158	164
173	130	166	117	157
174	131	167	130	166
175	132	168	131	167
176	133	169	132	168
177	170	90	151	77
178	171	91	170	90
179	172	92	171	91
180	45	45	172	92
181	173	170	165	161
182	174	171	173	170
183	175	172	174	171
184	45	45	173	172
185	176	173	159	165
186	177	174	175	173
187	178	175	177	174
188	45	45	178	175
189	143	176	133	169
190	144	177	143	176
191	145	178	144	177
192	45	45	145	178

The Distribution of the Source

FOR SUBSURFACE 1

-.11944E+04	-.11943E+04	-.11941E+04	-.11940E+04
-.11322E+04	-.11317E+04	-.11309E+04	-.11304E+04
-.10005E+04	-.99957E+03	-.99827E+03	-.99738E+03
-.75270E+03	-.75175E+03	-.75051E+03	-.74969E+03

FOR SUBSURFACE 2

-.29311E+03	-.29549E+03	-.29875E+03	-.30098E+03
-.41211E+02	-.41535E+02	-.41983E+02	-.42293E+02
.18245E+03	.18456E+03	.18757E+03	.18973E+03
.50453E+03	.51193E+03	.51987E+03	.52610E+03

FOR SUBSURFACE 3

.10697E+04	.10742E+04	.10805E+04	.10848E+04
.13099E+04	.13115E+04	.13137E+04	.13153E+04
.14415E+04	.14419E+04	.14425E+04	.14431E+04
.15021E+04	.15022E+04	.15022E+04	.15023E+04

FOR SUBSURFACE 5

-.11944E+04	-.11943E+04	-.11941E+04	-.11940E+04
-.11322E+04	-.11317E+04	-.11309E+04	-.11304E+04
-.10005E+04	-.99957E+03	-.99827E+03	-.99738E+03
-.75270E+03	-.75175E+03	-.75051E+03	-.74969E+03

FOR SUBSURFACE 6

-.29311E+03	-.29549E+03	-.29875E+03	-.30098E+03
-.41211E+02	-.41535E+02	-.41983E+02	-.42293E+02
.18245E+03	.18456E+03	.18757E+03	.18973E+03
.50453E+03	.51093E+03	.51987E+03	.52610E+03

FOR SUBSURFACE 7

.10697E+04	.10742E+04	.10805E+04	.10848E+04
.13099E+04	.13115E+04	.13137E+04	.13153E+04
.14415E+04	.14419E+04	.14426E+04	.14431E+04
.15021E+04	.15022E+04	.15022E+04	.15023E+04

FOR SUBSURFACE 9

-.11944E+04	-.11943E+04	-.11941E+04	-.11940E+04
-.11322E+04	-.11317E+04	-.11309E+04	-.11304E+04
-.10005E+04	-.99957E+03	-.99827E+03	-.99738E+03
-.75270E+03	-.75175E+03	-.75051E+03	-.74909E+03

FOR SUBSURFACE 10

-.29311E+03	-.29549E+03	-.29875E+03	-.30098E+03
-.41211E+02	-.41535E+02	-.41983E+02	-.42293E+02
.18245E+03	.18456E+03	.18757E+03	.18973E+03
.50453E+03	.51093E+03	.51987E+03	.52610E+03

FOR SUBSURFACE 11

.10697E+04	.10742E+04	.10805E+04	.10846E+04
.13099E+04	.13115E+04	.13137E+04	.13153E+04
.14415E+04	.14419E+04	.14426E+04	.14431E+04
.15021E+04	.15022E+04	.15022E+04	.15023E+04

FOR SUBSURFACE 13

-.11944E+04	-.11343E+04	-.11941E+04	-.11940E+04
-.11322E+04	-.11317E+04	-.11309E+04	-.11304E+04
-.10005E+04	-.99937E+03	-.99827E+03	-.99738E+03
-.75270E+03	-.75175E+03	-.75051E+03	-.74969E+03

FOR SUBSURFACE 14

-.29311E+03	-.29549E+03	-.29875E+03	-.30098E+03
-.41211E+02	-.41335E+02	-.41983E+02	-.42293E+02
.18245E+03	.18456E+03	.18757E+03	.18973E+03
.50453E+03	.51133E+03	.51987E+03	.52510E+03

FOR SUBSURFACE 15

.10697E+04	.10742E+04	.10805E+04	.10848E+04
.13099E+04	.13115E+04	.13137E+04	.13153E+04
.14415E+04	.14419E+04	.14426E+04	.14431E+04
.15021E+04	.15022E+04	.15022E+04	.15023E+04

The Distribution of the Velocity Potential

FOR SUBSURFACE 1

-.67872E+03	-.67879E+03	-.67892E+03	-.67915E+03
-.64670E+03	-.64671E+03	-.64696E+03	-.64760E+03
-.58307E+03	-.58329E+03	-.58396E+03	-.59536E+03
-.45087E+03	-.45151E+03	-.45303E+03	-.45560E+03

FOR SUBSURFACE 2

-.12407E+03	-.12732E+03	-.13424E+03	-.14102E+03
-.11810E+02	-.15937E+02	-.22830E+02	-.31136E+02
.78425E+02	.72959E+02	.63175E+02	.51499E+02
.21893E+03	.21015E+03	.19549E+03	.17049E+03

FOR SUBSURFACE 3

.76572E+03	.73131E+03	.72782E+03	.70411E+03
.90453E+03	.89511E+03	.87721E+03	.85861E+03
.96299E+03	.95337E+03	.94101E+03	.92914E+03
.96505E+03	.96313E+03	.95990E+03	.95621E+03

FOR SUBSURFACE 5

-.68045E+03	-.68025E+03	-.67989E+03	-.67949E+03
-.65243E+03	-.65137E+03	-.65018E+03	-.64874E+03
-.59367E+03	-.59226E+03	-.58995E+03	-.58746E+03
-.46828E+03	-.45626E+03	-.46289E+03	-.45906E+03

FOR SUBSURFACE 6

-.15193E+03	-.15151E+03	-.14998E+03	-.14654E+03
-.52254E+02	-.50039E+02	-.45609E+02	-.39116E+02
.18535E+02	.22424E+02	.29785E+02	.39390E+02
.12757E+03	.13428E+03	.14625E+03	.15155E+03

FOR SUBSURFACE 7

.65948E+03	.66364E+03	.67205E+03	.68527E+03
.82571E+03	.82937E+03	.83416E+03	.84398E+03
.90704E+03	.90995E+03	.91303E+03	.91967E+03
.94695E+03	.94796E+03	.94991E+03	.95274E+03

FOR SUBSURFACE 9

-.67872E+03	-.67378E+03	-.67892E+03	-.67915E+03
-.64670E+03	-.64671E+03	-.64694E+03	-.64760E+03
-.56307E+03	-.58328E+03	-.58396E+03	-.58536E+03
-.45087E+03	-.45151E+03	-.45303E+03	-.45560E+03

FOR SUBSURFACE 10

-.12407E+03	-.12732E+03	-.13426E+03	-.14102E+03
-.11810E+02	-.15637E+02	-.22830E+02	-.31136E+02
.78425E+02	.72853E+02	.63175E+02	.51499E+02
.21893E+03	.21015E+03	.19549E+03	.17649E+03

FOR SUBSURFACE 11

.76572E+03	.75131E+03	.72782E+03	.70411E+03
.90453E+03	.89511E+03	.87721E+03	.85361E+03
.96298E+03	.95357E+03	.94101E+03	.92914E+03
.96505E+03	.95313E+03	.95950E+03	.95621E+03

FOR SUBSURFACE 13

-.68045E+03	-.68025E+03	-.67985E+03	-.67949E+03
-.65243E+03	-.65157E+03	-.65016E+03	-.64874E+03
-.59367E+03	-.53226E+03	-.58995E+03	-.59746E+03
-.46828E+03	-.46626E+03	-.46289E+03	-.45906E+03

FOR SUBSURFACE 14

-.15193E+03	-.15151E+03	-.14998E+03	-.14654E+03
-.52254E+02	-.51233E+02	-.45609E+02	-.33116E+02
.18535E+02	.22424E+02	.29785E+02	.39350E+02
.12757E+03	.13429E+03	.14625E+03	.16155E+03

FOR SUBSURFACE 15

.65948E+03	.65354E+03	.67205E+03	.69527E+03
.82571E+03	.82837E+03	.83416E+03	.84398E+03
.90764E+03	.91135E+03	.91303E+03	.91967E+03
.94696E+03	.94795E+03	.94991E+03	.95274E+03

Pressure Distribution

FOR SUBSURFACE 1			
.61577E+00	.1[illegible]273E+01	.11126E+01	.11420E+01
.66014E+00	-.10644E+00	-.10991E+00	-.11134E+00
-.17570E+00	-.17734E+00	-.17999E+00	-.18[illegible]45E+00
-.32859E+00	-.32903E+00	-.32882E+00	-.32753E+00
FOR SUBSURFACE 2			
-.28964E+00	-.28751E+00	-.28430E+00	-.28075E+00
-.13356E+00	-.13263E+00	-.13023E+00	-.12704E+00
-.15172E+00	-.14859E+00	-.14413E+00	-.13851E+00
-.45281E+00	-.44751E+00	-.43959E+00	-.43215E+00
FOR SUBSURFACE 3			
-.69056E+00	-.72871E+00	-.77607E+00	-.60[illegible]26E+00
-.16858E+00	-.15559E+00	-.21181E+00	-.23123E+00
-.36355E-01	-.45608E-01	-.58911E-01	-.7[illegible]382E-01
.28684E-02	.19030E-03	-.34147E-02	-.67867E-02
FOR SUBSURFACE 5			
.96689E+00	.10318E+01	.11214E+01	.11330E+01
-.94236E-01	-.98972E-01	-.10516E+00	-.10972E+00
-.16570E+00	-.16370E+00	-.17483E+03	-.17670E+00
-.31676E+00	-.31937E+00	-.32266E+00	-.32543E+00

FOR SUBSURFACE 6

-.27501E+00	-.27525E+00	-.27607E+00	-.27787E+00
-.11324E+00	-.11536E+00	-.11895E+00	-.12311E+00
-.11960E+00	-.12233E+00	-.12702E+00	-.13266E+00
-.42368E+00	-.42335E+00	-.42466E+00	-.42726E+00

FOR SUBSURFACE 7

-.70167E+00	-.73426E+00	-.77709E+00	-.80523E+00
-.20899E+00	-.21340E+00	-.23233E+00	-.23817E+00
-.69518E-01	-.72507E-01	-.76396E-01	-.76379E-01
-.11431E-01	-.11401E-01	-.10677E-01	-.93713E-02

FOR SUBSURFACE 9

.61577E+00	.10275E+01	.11136E+01	.11620E+01
.66014E+00	-.10544E+00	-.10991E+00	-.11134E+00
-.17570E+00	-.17794E+00	-.17959E+00	-.19045E+00
-.32859E+00	-.32903E+00	-.32882E+00	-.32753E+00

FOR SUBSURFACE 10

-.28964E+00	-.29751E+00	-.28430E+00	-.28075E+00
-.13356E+00	-.13243E+00	-.13023E+00	-.12704E+00
-.15172E+00	-.14339E+00	-.14413E+00	-.13651E+00
-.45281E+00	-.44751E+00	-.43959E+00	-.43215E+00

FOR SUBSURFACE 11

-.69056E+00	-.72371E+00	-.77507E+00	-.80526E+00
-.16858E+00	-.18558E+00	-.21161E+00	-.23123E+00
-.36355E-01	-.45509E-01	-.58911E-01	-.70362E-01
.28684E-02	.19030E-03	-.34147E-02	-.67367E-02

FOR SUBSURFACE 13

.96689E+00	.10318E+01	.11214E+01	.11630E+01
-.94236E-01	-.98472E-01	-.10516E+00	-.10972E+00
-.16570E+00	-.16970E+00	-.17483E+00	-.17670E+00
-.31676E+00	-.31937E+00	-.32266E+00	-.32543E+00

FOR SUBSURFACE 14

-.27501E+00	-.27525E+00	-.27607E+00	-.27787E+00
-.11324E+00	-.11536E+00	-.11895E+00	-.12311E+00
-.11960E+00	-.12233E+00	-.12702E+00	-.13266E+00
-.42368E+00	-.42395E+00	-.42486E+00	-.42726E+00

FOR SUBSURFACE 15

-.70167E+00	-.73426E+00	-.77709E+00	-.81523E+00
-.21899E+00	-.21942E+00	-.23233E+00	-.23517E+00
-.69518E-01	-.72807E-01	-.76396E-01	-.75379E-01
-.11431E-01	-.11401E-01	-.10977E-01	-.93713E-02

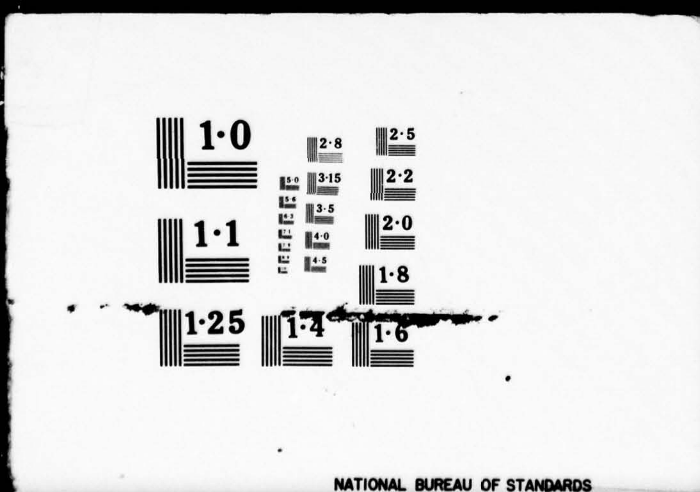
AD-A055 767 AEROSPACE SYSTEMS INC BURLINGTON MASS F/G 20/4
ROTOR WAKE EFFECTS ON HUB/PYLON FLOW. VOLUME II. PROGRAM SHAPES--ETC(U)
MAY 78 P SOOHOO, R B NOLL, L MORINO, N D HAMM DAAJ02-75-C-0041
UNCLASSIFIED ASI-TR-76-38-VOL-2 USARTL-TR-78-1B NL

2 OF 2
ADA
055767

END
DATE
FILMED
8-78
DDC

2 OF 2

ADA 055767



SAMPLE CASE 4:

MODEL 1 HELICOPTER CONFIGURATION WITH ROTOR WAKE AND SEPARATION EFFECTS

Data Input

```
  1
PROGRAM SHAPES
HUB PYLON FLOW SEPARATION
FUSELAGE - SINGLE ROTOR (TWO BLADES)
KREAD       0
*          16     0     0
**          0     1     1     1     0     0     0     0     1
***         0     0
UMACH        .047  3600.000    -0.000
ALFA        0.000      0.000
NBODY       0.000      9.000    18.000    36.000
NBODY       0.000      0.000     0.000     0.000
NBODY       0.000      0.000     0.000     0.000
NBODY       1.450      3.250     1.450     2.900
ROTOR      20 000      8.040    11.550    15.050     0.000     0.000      .120
IHEI75      0.000      0.000     0.000     0.000    90.000
RPITCH      0.000      0.000
KBLADE      2     1
NWAKE       1     0     0     1
SPIRAL     18     3.000 450.000
NPYLON      1     1
CSIAG     -60.000      1.000
KPRINT      1     1     1     1     0     1     1     1     1     1
NBODY       3     3     2     1     1
NBODY       3     3     1     1     1
NBODY       3     3     2     1     1
NROTOR      3     3     2     1     1
```

Specifications of the Problem

```
FOR PART    1
NX= 3
NY= 3

FOR PART    2
NX= 3
NY= 3

FOR PART    3
NX= 3
NY= 3

FOR PART    5
NX= 3
NY= 3

FOR PART    6
NX= 3
NY= 3

FOR PART    7
NX= 3
NY= 3

FOR PART    9
NX= 3
NY= 3

FOR PART    10
NX= 3
NY= 3

FOR PART    11
NX= 3
NY= 3

FOR PART    13
NX= 3
NY= 3

FOR PART    14
NX= 3
NY= 3

FOR PART    15
NX= 3
NY= 3

FOR PART    24
NX= 3
NY= 3

FOR PART    25
NX= 3
NY= 3

FOR PART    27
NX= 3
NY= 3

FOR PART    28
NX= 3
NY= 3

NELEM=144

KSYMMY= 0
KSYMMZ= 0
```

X, Y, and Z Coordinates of the Nodes

NODE	X	Y	Z
1	0.00000	0.00000	0.00000
2	1.00000	-.00000	1.08333
3	4.00000	-.00000	2.16667
4	9.00000	.00000	3.25000
5	1.00000	.24167	.93819
6	4.00000	.48333	1.87639
7	9.00000	.72500	2.81458
8	1.00000	.41858	.54167
9	4.00000	.83716	1.08333
10	9.00000	1.25574	1.62500
11	1.00000	.48333	-.00000
12	4.00000	.96667	-.00000
13	9.00000	1.45000	-.00000
14	12.00000	-.00000	3.13333
15	15.00000	-.00000	3.01667
16	18.00000	.00000	2.90000
17	12.00000	.72500	2.71355
18	15.00000	.72500	2.61251
19	18.00000	.72500	2.51147
20	12.00000	1.25574	1.56667
21	15.00000	1.25574	1.50833
22	18.00000	1.25574	1.45000
23	12.00000	1.45000	-.00000
24	15.00000	1.45000	-.00000
25	18.00000	1.45000	-.00000
26	28.00000	-.00000	1.93333
27	34.00000	-.00000	.96667
28	36.00000	0.00000	0.00000
29	28.00000	.48333	1.67432
30	34.00000	.24167	.83716
31	28.00000	.83716	.96667
32	34.00000	.41858	.48333
33	28.00000	.96667	-.00000
34	34.00000	.48333	-.00000
35	1.00000	-.00000	-1.08333
36	4.00000	-.00000	-2.16667
37	9.00000	.00000	-3.25000
38	1.00000	.24167	-.93819
39	4.00000	.48333	-1.87639
40	9.00000	.72500	-2.81458
41	1.00000	.41858	-.54167
42	4.00000	.83716	-1.08333
43	9.00000	1.25574	-1.62500
44	12.00000	-.00000	-3.13333
45	15.00000	-.00000	-3.01667
46	18.00000	.00000	-2.90000
47	12.00000	.72500	-2.71355
48	15.00000	.72500	-2.61251

49	18.00000	.72500	-2.51147
50	12.00000	1.25574	-1.56667
51	15.00000	1.25574	-1.50833
52	18.00000	1.25574	-1.45000
53	29.00000	-.00000	-1.93333
54	34.00000	-.00000	-.96667
55	29.00000	.48333	-1.67432
56	34.00000	.24167	-.83716
57	29.00000	.83716	-.96667
58	34.00000	.41858	-.48333
59	1.00000	-.24167	.93819
60	4.00000	-.48333	1.87639
61	9.00000	-.72500	2.61458
62	1.00000	-.41858	.54167
63	4.00000	-.83716	1.08333
64	9.00000	-1.25574	1.62500
65	1.00000	-.48333	-.00000
66	4.00000	-.96667	-.00000
67	9.00000	-1.45000	-.00000
68	12.00000	-.72500	2.71355
69	15.00000	-.72500	2.61251
70	18.00000	-.72500	2.51147
71	12.00000	-1.25574	1.56667
72	15.00000	-1.25574	1.50833
73	18.00000	-1.25574	1.45000
74	12.00000	-1.45000	-.00000
75	15.00000	-1.45000	-.00000
76	18.00000	-1.45000	-.00000
77	29.00000	-.48333	1.67432
78	34.00000	-.24167	.83716
79	29.00000	-.83716	.96667
80	34.00000	-.41858	.48333
81	29.00000	-.96667	-.00000
82	34.00000	-.48333	-.00000
83	1.00000	-.24167	-.93819
84	4.00000	-.48333	-1.87639
85	9.00000	-.72500	-2.61458
86	1.00000	-.41858	-.54167
87	4.00000	-.83716	-1.08333
88	9.00000	-1.25574	-1.62500
89	12.00000	-.72500	-2.71355
90	15.00000	-.72500	-2.61251
91	18.00000	-.72500	-2.51147
92	12.00000	-1.25574	-1.56667
93	15.00000	-1.25574	-1.50833
94	18.00000	-1.25574	-1.45000
95	29.00000	-.48333	-1.67432
96	34.00000	-.24167	-.83716
97	29.00000	-.83716	-.96667

98	34.00000	-.41858	-.48333
99	11.55000	3.75000	6.67000
100	11.93889	3.75000	6.70462
101	13.10556	3.75000	6.75642
102	15.05000	3.75000	6.67000
103	11.55000	12.77778	6.67000
104	11.93889	12.77778	6.70462
105	13.10556	12.77778	6.75642
106	15.05000	12.77778	6.67000
107	11.55000	18.19444	6.67000
108	11.93889	18.19444	6.70462
109	13.10556	18.19444	6.75642
110	15.05000	18.19444	6.67000
111	11.55000	20.00000	6.67000
112	11.93889	20.00000	6.67000
113	13.10556	20.00000	6.67000
114	15.05000	20.80000	6.67000
115	11.93889	3.75000	6.63538
116	13.10556	3.75000	6.58358
117	11.93889	12.77778	6.63538
118	13.18556	12.77778	6.58358
119	11.93889	18.19444	6.63538
120	13.10556	18.19444	6.58358
121	-11.55000	-3.75000	6.67000
122	-11.93889	-3.75000	6.70462
123	-13.10556	-3.75000	6.75642
124	-15.05000	-3.75000	6.67000
125	-11.55080	-12.77778	6.67000
126	-11.93889	-12.77778	6.70462
127	-13.10556	-12.77778	6.75642
128	-15.05000	-12.77778	6.67800
129	-11.35000	-18.19444	6.67000
130	-11.33889	-18.19444	6.70462
131	-13.10556	-18.19444	6.75642
132	-15.05000	-18.19444	6.67000
133	-11.55000	-20.00000	6.67000
134	-11.33889	-20.00000	6.67000
135	-13.10556	-20.00000	6.67000
136	-15.05000	-28.00000	6.67000
137	-11.33889	-3.75000	6.63538
138	-13.10556	-3.75000	6.58358
139	-11.93689	-12.77778	6.63538
140	-13.10556	-12.77778	6.58358
141	-11.93889	-18.19444	6.63538
142	-13.10556	-18.19444	6.58358

X, Y, and Z Coordinates of the Centroid of Aerodynamic Panels

ELEM=	XPC	YPC	ZPC
1	.50055	.06042	.50538
2	2.50277	.18125	1.51615
3	6.50719	.30208	2.52691
4	.50055	.16506	.36997
5	2.50277	.49518	1.10990
6	6.50719	.82531	1.84983
7	.50055	.22548	.13542
8	2.50277	.67643	.40625
9	6.50719	1.12739	.67708
10	10.51162	.36250	2.97787
11	13.51434	.36250	2.86901
12	16.51825	.36250	2.76016
13	10.51162	.99037	2.17995
14	13.51494	.99037	2.10026
15	16.51825	.99037	2.02058
16	10.51162	1.35287	.79792
17	13.51494	1.35287	.76875
18	16.51825	1.35287	.73958
19	23.02545	.30208	2.25478
20	31.03438	.18125	1.35287
21	35.13872	.06042	.45096
22	23.02545	.82531	1.65061
23	31.03438	.49518	.99037
24	35.13872	.16506	.33012
25	23.02545	1.12739	.60417
26	31.03438	.67643	.36250
27	35.03872	.22548	.12083
28	.50055	.06042	-.50538
29	2.50277	.18125	-1.51615
30	6.50719	.30208	-2.52691
31	.50055	.16506	-.36997
32	2.50277	.49518	-1.10990
33	6.50719	.82531	-1.84983
34	.50055	.22548	-.13542
35	2.50277	.67643	-.40625
36	6.50719	1.12739	-.67708
37	10.51162	.36250	-2.97787
38	13.51494	.36250	-2.86901
39	16.51825	.36250	-2.76016
40	10.51162	.99037	-2.17995
41	13.51494	.99037	-2.10026
42	16.51825	.99037	-2.02058
43	10.51162	1.35287	-.79792
44	13.51494	1.35297	-.76875
45	16.51825	1.35287	-.73958
46	23.02545	.30208	-2.25478
47	31.13430	.18125	-1.35287
48	35.03872	.06042	-.45096
49	23.02545	.82531	-1.65061

50	31.03430	.49518	-.99037
51	35.03872	.16506	-.33012
52	23.02545	1.12739	-.60417
53	31.03430	.67643	-.36250
54	35.03872	.22548	-.12083
55	.50055	-.06042	.50538
56	2.50277	-.18125	1.51615
57	6.50719	-.30208	2.52691
58	.50055	-.16506	.36997
59	2.50277	-.49518	1.10990
60	6.50719	-.82531	1.84983
61	.50055	-.22548	.13542
62	2.50277	-.67643	.40625
63	6.50719	-1.12739	.67708
64	10.51162	-.36250	2.97787
65	13.51494	-.36250	2.86901
66	16.51825	-.36250	2.76016
67	10.51162	-.99037	2.17995
68	13.51494	-.99037	2.10026
69	16.51825	-.99037	2.02058
70	10.51162	-1.35297	.79792
71	13.51494	-1.35287	.76875
72	16.51825	-1.35297	.73958
73	23.02545	-.30208	2.25478
74	31.03430	-.18125	1.35287
75	35.03872	-.06042	.45096
76	23.02545	-.82531	1.65061
77	31.03430	-.49518	.99037
78	35.03872	-.16506	.33012
79	23.02545	-1.12739	.60417
80	31.03430	-.67643	.36250
81	35.03872	-.22548	.12083
82	.50055	-.06042	-.50538
83	2.50277	-.18125	-1.51615
84	6.50719	-.30208	-2.52691
85	.50055	-.16506	-.36997
86	2.50277	-.49518	-1.10990
87	6.50719	-.82531	-1.84983
88	.50055	-.22548	-.13542
89	2.50277	-.67643	-.40625
90	6.50719	-1.12739	-.67708
91	10.51162	-.36250	-2.97787
92	13.51494	-.36250	-2.86901
93	16.51825	-.36250	-2.76016
94	10.51162	-.99037	-2.17995
95	13.51494	-.99037	-2.10026
96	16.51825	-.99037	-2.02058
97	10.51162	-1.35287	-.79792
98	13.51494	-1.35297	-.76875

99	16.01925	-1.35287	-.73358
100	23.02545	-.30208	-2.25479
101	31.13430	-.18125	-1.35287
102	35.13872	-.06042	-.45096
103	23.02545	-.82531	-1.65061
104	31.03430	-.49518	-.99037
105	35.03872	-.16506	-.33012
106	23.02545	-1.12739	-.60417
107	31.03430	-.67643	-.36250
108	35.03872	-.22548	-.12083
109	11.75744	8.26389	6.68731
110	12.53608	8.26389	6.73052
111	14.19335	8.26389	6.71321
112	11.75744	15.48611	6.68731
113	12.53608	15.48611	6.73052
114	14.09335	15.48611	6.71321
115	11.75744	19.09722	6.67866
116	12.53608	19.09722	6.70026
117	14.19335	19.09722	6.69161
118	11.75744	8.26389	6.65269
119	12.53608	8.26389	6.60348
120	14.09335	8.26389	6.62679
121	11.75744	15.48611	6.65269
122	12.53608	15.48611	6.60948
123	14.09335	15.48611	6.62679
124	11.75744	19.09722	6.66134
125	12.53608	19.09722	6.63974
126	14.09335	19.09722	6.64839
127	-11.75744	-8.26389	6.68731
128	-12.53608	-8.26389	6.73052
129	-14.09335	-8.26389	6.71321
130	-11.75744	-15.48611	6.68731
131	-12.53608	-15.48611	6.73052
132	-14.09335	-15.48611	6.71321
133	-11.75744	-19.09722	6.67866
134	-12.53608	-19.09722	6.70026
135	-14.09335	-19.09722	6.69161
136	-11.75744	-8.26389	6.65269
137	-12.53608	-8.26389	6.60948
138	-14.09335	-8.26399	6.62679
139	-11.75744	-15.48611	6.65269
140	-12.53608	-15.48611	6.60948
141	-14.09335	-15.48611	6.62679
142	-11.75744	-19.09722	6.66134
143	-12.53608	-19.09722	6.63974
144	-14.09335	-19.09722	6.64839

Nodal Numbering for Surfaces

FOR SURFACE 1

1	1	1	1
2	5	8	11
3	6	9	12
4	7	10	13

FOR SURFACE 2

4	7	10	13
14	17	20	23
15	18	21	24
16	19	22	25

FOR SURFACE 3

16	19	22	25
26	29	31	33
27	30	32	34
23	28	28	28

FOR SURFACE 5

1	1	1	1
35	38	41	11
36	39	42	12
37	40	43	13

FOR SURFACE 6

37	40	43	13
44	47	50	23
45	48	51	24
46	49	52	25

FOR SURFACE 7

46	49	52	25
53	55	57	33
54	56	58	34
23	28	28	28

FOR SURFACE 9

1	1	1	1
2	59	62	65
3	60	63	66
4	61	64	67

FOR SURFACE 10

4	61	64	67
14	68	71	74
15	69	72	75
16	70	73	76

FOR SURFACE 11

16	70	73	76
26	77	79	81
27	78	80	82
28	28	28	28

FOR SURFACE 13

1	1	1	1
35	83	86	65
36	84	87	66
37	85	88	67

FOR SURFACE 14

37	85	88	67
44	89	92	74
45	90	93	75
46	91	94	76

FOR SURFACE 15

46	91	94	76
53	95	97	81
54	96	98	82
28	28	28	28

FOR SURFACE 24

99	103	107	111
100	104	108	112
101	105	109	113
102	106	110	114

FOR SURFACE 25

99	103	107	111
115	117	119	112
116	118	120	113
102	106	110	114

FOR SURFACE 27

121	125	129	133
122	126	130	134
123	127	131	135
124	128	132	136

FOR SURFACE 28

121	125	129	133
137	139	141	134
138	140	142	135
124	128	132	136

Nodal Numbering for Elements

ELEM	++	+-	-+	--
1	5	2	1	1
2	6	3	5	2
3	7	4	6	3
4	8	5	1	1
5	9	6	8	5
6	10	7	9	6
7	11	8	1	1
8	12	9	11	8
9	13	10	12	9
10	17	14	7	4
11	18	15	17	14
12	19	16	18	15
13	20	17	10	7
14	21	18	20	17
15	22	19	21	18
16	23	20	13	10
17	24	21	23	20
18	25	22	24	21
19	29	26	19	16
20	30	27	29	26
21	28	28	30	27
22	31	29	22	19
23	32	30	31	29
24	28	28	32	30
25	33	31	25	22
26	34	32	33	31
27	28	28	34	32
28	35	38	1	1
29	36	39	35	38
30	37	40	36	39
31	38	41	1	1
32	39	42	38	41
33	40	43	39	42
34	41	11	1	1
35	42	12	41	11
36	43	13	42	12
37	44	47	37	40
38	45	48	44	47
39	46	49	45	48
40	47	50	40	43
41	48	51	47	50
42	49	52	48	51
43	50	23	43	13
44	51	24	50	23
45	52	25	51	24
46	53	55	46	49
47	54	56	53	55
48	28	28	54	56

49	55	57	49	52
50	56	58	55	57
51	28	28	56	58
52	57	33	52	25
53	58	34	57	33
54	28	28	58	34
55	2	59	1	1
56	5	60	2	59
57	4	61	3	60
58	59	62	1	1
59	60	63	59	62
60	61	64	60	63
61	62	65	1	1
62	63	66	62	65
63	64	67	63	66
64	14	68	4	61
65	15	69	14	68
66	16	70	15	69
67	68	71	61	64
68	69	72	58	71
69	70	73	69	72
70	71	74	64	67
71	72	75	71	74
72	73	76	72	75
73	26	77	16	70
74	27	78	26	77
75	28	28	27	78
76	17	79	70	73
77	78	80	77	79
78	28	28	78	80
79	79	81	73	76
80	80	82	79	81
81	28	28	80	82
82	83	35	1	1
83	84	36	83	35
84	85	37	84	36
85	86	83	1	1
86	87	84	86	83
87	88	85	87	84
88	65	86	1	1
89	56	87	65	86
90	67	88	66	87
91	89	44	85	37
92	30	45	89	44
93	41	46	90	45
94	92	89	88	85
95	43	90	32	89
96	94	91	93	90

97	74	92	67	88
98	75	93	74	92
99	76	94	75	93
100	95	53	91	46
101	96	54	95	53
102	28	28	96	54
103	97	95	94	91
104	98	96	97	95
105	28	28	98	96
106	81	97	76	94
107	82	98	81	97
108	28	28	82	98
109	104	100	103	99
110	105	101	104	100
111	106	102	105	101
112	108	104	107	103
113	109	105	108	104
114	110	106	109	105
115	112	108	111	107
116	113	109	112	108
117	114	110	113	109
118	115	117	99	103
119	116	118	115	117
120	102	106	116	118
121	117	119	103	107
122	118	120	117	119
123	106	110	118	120
124	119	112	107	111
125	120	113	119	112
126	110	114	120	113
127	126	122	125	121
128	127	123	126	122
129	128	124	127	123
130	130	126	129	125
131	131	127	130	126
132	132	128	131	127
133	134	130	133	129
134	135	131	134	130
135	136	132	135	131
136	137	139	121	125
137	138	140	137	139
138	124	128	138	140
139	139	141	125	129
140	140	142	139	141
141	128	132	140	142
142	141	134	129	133
143	142	135	141	134
144	132	136	142	135

The Distribution of the Source

FOR SUBSURFACE 1		
-.89735E+02	-.89462E+02	-.89131E+02
-.84927E+02	-.84701E+02	-.84594E+02
-.74571E+02	-.74839E+02	-.74282E+02
FOR SUBSURFACE 2		
-.47538E+02	-.49261E+02	-.48983E+02
-.27588E+02	-.28720E+02	-.30053E+02
-.11036E+02	-.11957E+02	-.14521E+02
FOR SUBSURFACE 3		
.75639E+01	.69589E+01	.54540E+01
.13044E+02	.13474E+02	.12748E+02
.16507E+02	.16315E+02	.15939E+02
FOR SUBSURFACE 5		
-.89536E+02	-.89490E+02	-.89141E+02
-.84723E+02	-.84573E+02	-.84537E+02
-.72623E+02	-.72787E+02	-.73400E+02
FOR SUBSURFACE 6		
-.42652E+02	-.44604E+02	-.47102E+02
-.29142E+02	-.29854E+02	-.30444E+02
-.18342E+02	-.17929E+02	-.16803E+02

FOR SUBSURFACE 7

.10734E+01	.20296E+01	.35994E+01
.10713E+02	.11148E+02	.11893E+02
.15577E+02	.15562E+02	.15663E+02

FOR SUBSURFACE 9

-.89783E+02	-.89429E+02	-.89089E+02
-.84634E+02	-.84210E+02	-.84007E+02
-.73652E+02	-.72497E+02	-.71870E+02

FOR SUBSURFACE 10

-.46084E+02	-.46123E+02	-.46406E+02
-.27741E+02	-.29056E+02	-.30431E+02
-.12867E+02	-.14895E+02	-.17301E+02

FOR SUBSURFACE 11

.70612E+01	.56619E+01	.38513E+01
.13799E+02	.13237E+02	.12415E+02
.16538E+02	.16263E+02	.15865E+02

FOR SUBSURFACE 13

-.89525E+02	-.89462E+02	-.89100E+02
-.84622E+02	-.84266E+02	-.84043E+02
-.72382E+02	-.71951E+02	-.71744E+02

FOR SUBSURFACE 14

-.42542E+02	-.44103E+02	-.45755E+02
-.29117E+02	-.30195E+02	-.30847E+02
-.18708E+02	-.19042E+02	-.18739E+02

FOR SUBSURFACE 15

.74244E+00	.10670E+01	.21658E+01
.10617E+02	.10890E+02	.11551E+02
.15596E+02	.15506E+02	.15589E+02

FOR SUBSURFACE 24

-.38755E+03	-.70861E+03	-.62415E+03
-.26131E+03	-.48756E+03	-.61629E+03
.23480E+03	.48777E+03	-.42148E+02

FOR SUBSURFACE 25

-.38756E+03	-.70861E+03	-.62415E+03
-.26133E+03	-.48756E+03	-.61628E+03
.23479E+03	.48777E+03	-.42148E+02

FOR SUBSURFACE 27

-.37486E+03	-.69827E+03	-.61925E+03
-.25512E+03	-.48144E+03	-.61353E+03
.22397E+03	.39747E+03	-.48037E+02

FOR SUBSURFACE 28

-.37616E+03	-.69828E+03	-.61925E+03
-.25513E+03	-.48144E+03	-.61353E+03
.22396E+03	.39747E+03	-.48039E+02

The Distribution of the Velocity Potential

FOR SUBSURFACE 1

-.46059E+02	-.45837E+02	-.45642E+02
-.43304E+02	-.43535E+02	-.43832E+02
-.38208E+02	-.39314E+02	-.39935E+02

FOR SUBSURFACE 2

-.20751E+02	-.23836E+02	-.25505E+02
-.94652E+01	-.11712E+02	-.15051E+02
.59479E+01	.58202E+01	.19665E+01

FOR SUBSURFACE 3

-.10816E+03	-.10755E+03	-.81939E+02
-.50172E+03	-.35268E+03	-.23162E+03
-.28712E+03	-.26498E+03	-.23479E+03

FOR SUBSURFACE 5

-.46449E+02	-.46103E+02	-.45735E+02
-.44549E+02	-.44330E+02	-.44107E+02
-.40363E+02	-.40151E+02	-.40099E+02

FOR SUBSURFACE 6

-.24928E+02	-.25525E+02	-.25893E+02
-.20842E+02	-.20185E+02	-.18347E+02
-.17573E+02	-.14880E+02	-.10502E+02

FOR SUBSURFACE 7

-.46017E+02	-.51944E+02	-.60332E+02
-.11458E+03	-.12897E+03	-.16635E+03
-.17639E+03	-.18723E+03	-.20720E+03

FOR SUBSURFACE 9

-.46016E+02	-.45697E+02	-.45454E+02
- .42909E+02	-.42573E+02	-.42661E+02
-.36572E+02	-.35876E+02	-.36314E+02

FOR SUBSURFACE 10

-.18657E+02	-.19743E+02	- .22044E+02
-.96158E+01	-.12030E+02	-.15413E+02
.41598E+01	.20712E+01	-.17663E+01

FOR SUBSURFACE 11

-.10339E+03	-.10960E+03	-.84478E+02
-.50055E+03	-.35293E+03	-.23248E+03
-.28715E+03	-.26507E+03	-.23491E+03

FOR SUBSURFACE 13

-.46401E+02	-.45972E+02	-.45552E+02
-.44297E+02	-.43636E+02	-.43068E+02
-.39852E+02	-.38630E+02	-.37418E+02

FOR SUBSURFACE 14

-.24681E+02	-.24636E+02	-.23897E+02
-.21048E+02	-.20633E+02	-.18778E+02
- .18526E+02	-.16667E+02	-.13421E+02

FOR SUBSURFACE 15		
-.46536E+02	-.53579E+02	-.62687E+02
-.11372E+03	-.12969E+03	-.16475E+03
-.17643E+03	-.18733E+03	-.20732E+03
FOR SUBSURFACE 24		
-.19900E+03	-.36608E+03	-.31439E+03
-.13739E+03	-.26042E+03	-.31484E+03
.13121E+03	.22146E+03	-.16031E+02
FOR SUBSURFACE 25		
-.20091E+03	-.36731E+03	-.31523E+03
-.14058E+03	-.26242E+03	-.31620E+03
.12739E+03	.21921E+03	-.17558E+02
FOR SUBSURFACE 27		
-.19701E+03	-.36264E+03	-.31327E+03
-.13848E+03	-.25951E+03	-.31511E+03
.11989E+03	.21306E+03	-.21094E+02
FOR SUBSURFACE 28		
-.19725E+03	-.36280E+03	-.31340E+03
-.13894E+03	-.25976E+03	-.31532E+03
.11955E+03	.21280E+03	-.21302E+02

Pressure Distribution

FOR SUBSURFACE 1

.16033E+01	.12491E+01	.13863E+01
.47196E+00	-.37578E-01	-.35702E-01
-.75373E-01	-.71136E-01	-.69669E-01

FOR SUBSURFACE 2

-.17921E+00	-.17405E+00	-.15750E+00
-.16858E+00	-.18174E+00	-.15948E+00
.62216E+00	.56760E+00	.43089E+00

FOR SUBSURFACE 3

.91396E+00	.75819E+00	.48358E+00
.73212E+00	.86923E+00	.57373E+00
-.15581E+01	.89266E-01	.15728E+00

FOR SUBSURFACE 5

.92646E+00	.12525E+01	.13879E+01
-.35608E-01	-.38367E-01	-.36135E-01
-.70634E-01	-.70751E-01	-.69486E-01

FOR SUBSURFACE 6

-.12349E+00	-.12867E+00	-.13994E+00
-.48501E-01	-.68291E-01	-.10844E+00
.17198E+00	.20744E+00	.28982E+00

FOR SUBSURFACE 7

.18416E+00	.21073E+00	.38861E+00
.38839E+00	.34409E+00	.40358E+00
.11388E+01	.73242E+00	.39221E+00

FOR SUBSURFACE 9

.17426E+01	.12271E+01	.13663E+01
.45358E+00	-.59901E-01	-.55051E-01
-.89829E-01	-.83759E-01	-.77312E-01

FOR SUBSURFACE 10

-.16883E+00	-.15273E+00	-.13413E+00
-.14338E+00	-.13943E+00	-.11648E+00
.52453E+00	.57705E+00	.44364E+00

FOR SUBSURFACE 11

.91436E+00	.75143E+00	.47793E+00
.72772E+00	.86170E+00	.56648E+00
-.15614E+01	.81267E-01	.15053E+00

FOR SUBSURFACE 13

.91495E+00	.12334E+01	.13684E+01
-.44118E-01	-.52030E-01	-.52583E-01
-.74141E-01	-.75231E-01	-.74709E-01

FOR SUBSURFACE 14

-.11879E+00	-.11590E+00	-.12045E+00
-.43102E-01	-.50677E-01	-.77182E-01
.17265E+00	.21433E+00	.30114E+00

FOR SUBSURFACE 15

.18181E+00	.20744E+00	.29630E+00
.37629E+00	.33818E+00	.33673E+00
.11291E+01	.72403E+00	.38526E+00

FOR SUBSURFACE 24

-.24559E+03	-.41692E+03	-.15643E+03
-.40284E+03	-.87233E+03	-.90483E+03
-.17987E+03	-.47176E+03	-.66969E+03

FOR SUBSURFACE 25

-.23212E+03	-.39809E+03	-.14613E+03
-.40121E+03	-.86989E+03	-.90090E+03
-.18330E+03	-.47636E+03	-.67019E+03

FOR SUBSURFACE 27

-.22548E+03	-.39161E+03	-.14203E+03
-.37732E+03	-.83780E+03	-.87553E+03
-.16108E+03	-.45538E+03	-.65340E+03

FOR SUBSURFACE 28

-.22392E+03	-.38923E+03	-.14061E+03
-.37728E+03	-.83753E+03	-.87504E+03
-.16944E+03	-.45653E+03	-.65346E+03

SECTION 6

PROGRAM LISTING

```
      PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C
C     NS(MAX)=34
C     NELEM(MAX)=400
C     NNODE(MAX)=350
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSEX,NSBODY,NS,NI(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA,AREA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/NSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      COMMON/ZZZ32/CSTAG,CVORT
      COMMON/ZZZ33/UWAKE
      COMMON/ZZZ34/YCUT
      COMMON/ZZZ35/ITWIST,RPITCH,TWIST
      COMMON/ZZZ50/XCTR,YCTR,ZCTR,RX,RY,RZ
      COMMON/ZZZ51/NSPAN,XLEZH,XTEZH,TANLEH,TANTEH,TAUH,ZP
      DIMENSION VELY(250),VELZ(250),PHI3(250),KROTOR(250)
      DIMENSION CP(250),VELX(250),GG(2),AVG(250)
      DIMENSION PC(3,250),P1(3,250),P2(3,250),P3(3,250),XK(3,250)
      DIMENSION KWAKE(250),NODE(4,250),NOECT(40,34)
      DIMENSION PHIC(250),PHI1(250),PHI2(250),BC(250)
      DIMENSION AA(62500),SOURCE(250),SOR(250)
      EQUIVALENCE (AA(1),PHIC(1)),(AA(401),PHI1(1))
      EQUIVALENCE (AA(801),PHI2(1)),(AA(1201),PHI3(1))
      EQUIVALENCE (AA(1601),VELX(1)),(AA(2001),VELY(1))
      EQUIVALENCE (AA(2401),VELZ(1)),(AA(2801),CP(1))
      REAL LSHAFT,LSHANK
C
      NREAD=5
```

```
      NWRITE=6
      NXMAX=6
      NYMAX=6
      NTMAX=400
      NXMAXP=NXMAX+1
      NYMAXP=NYMAX+1
      NXYMP=NXMAXP*NYMAXP
      NSTAG=0
      NVORT=0
      NSPIRAL=0
      NWAKBL=0
C
      READ(NREAD,2) NCASE
      WRITE(NWRITE,2)NCASE
    2 FORMAT(10I5)
    3 FORMAT(10F8.3)
    4 FORMAT(2A4,2I5)
C
      DO 999 ICASE=1,NCASE
C
C
      CALL DATA(NELEM)
C
      CALL REFFRO(NNODE,NXYMP,NOFCT,NELEM,NTWAKE)
C
      CALL COOORT(NELEM,NXYMP,NOFCT,XK,NNODE)
C
      CALL CHECK(NELEM,NXMAX,NYMAX,NTMAX)
C
      CALL GEOMET(NELEM,NXYMP,NOFCT,NODE,XK,NNODE,KWAKE,KROTOR)
      WRITE(6,60)
   60 FORMAT(10X,*--- MAIN FROM GEOMAT ---*)
C
      IF(KPRINT(1).EQ.1)
     1 CALL FRINTA(NELEM,NXYMF,XK,NNODE,PC,NOFCT,NODE,1)
C
      IF(KPRINT(2).EQ.1)
     1CALL PRINTA(NELEM,NXYMP,XK,NNODE,PC,NOFCT,NODE,2)
C
      CALL VEC123(NELEM,NODE,PC,R1,R2,R3,XK,NNODE)
C
      IF(KPRINT(3).EQ.1)
     1CALL PRINTA(NELEM,NXYMP,XK,NNODE,PC,NOFCT,NODE,3)
C
      IF (KPRINT(4).EQ.1)
     1CALL PRINTA(NELEM,NXYMP,XK,NNODE,PC,NOFCT,NODE,4)
C
      NESQ=NELEM**2
C
      DO 6 NN=1,NESQ
      AA(NN)=0.0
    6 CONTINUE
C
      DO 8 I=1,NELEM
      NNN=I+(I-1)*NELEM
      SOURCE(I)=0.0
      AA(NNN)=1.0
```

```
      9 CONTINUE
C
        CALL COEFF(NELEM,PC,NNODE,XK,NODE,KWAKE,NESQ,AA,SOURCE,BC,
       1 KROTOR,1)
C
        IF(NSTAG.NE.0)
       1 CALL COEFF(NELEM,PC,NNODE,XK,NODE,KWAKE,NESQ,AA,SOURCE,BC,
       2 KROTOR,2)
C
        IF(NVORT.NE.0)
       1 CALL COEFF(NELEM,PC,NNODE,XK,NODE,KWAKE,NESQ,AA,SOURCE,BC,
       2 KROTOR,3)
C
        IF(NBLADE.EQ.1 .AND. NWAKBL.NE.0)
       1 CALL COEFF(NELEM,PC,NNODE,XK,NODE,KWAKE,NESQ,AA,SOURCE,BC,
       2 KROTOR,4)
C
        IF(KPRINT(5).EQ.1)CALL PRINTB(NELEM,NESQ,AA,SOURCE,1)
C
        IF(KPRINT(6).EQ.1)CALL PRINTB(NELEM,NESQ,AA,SOURCE,2)
C
        CALL SOLUTN(NELEM,NESQ,AA,SOURCE)
C
        IF(KPRINT(7).EQ.1)CALL PRINTB(NELEM,NESQ,AA,SOURCE,3)
C
        CALL AVERAG(SOURCE,SOR,[illegible],NELEM,NODE,NOECT,NXYMP,NNODE,AVG)
C
        CALL PHI(SOR,NNODE,NODE,PHIC,PHI1,PHI2,PHI3,NELEM)
C
        CALL VELXYZ(VELX,VELY,VELZ,PHI1,PHI2,PHI3,PC,P1,P2,P3,NELEM)
C
        CALL CPLINR(CP,VELX,VELY,VELZ,PHIC,NELEM,PC,KROTOR)
C
        IF(KPRINT(8).EQ.1)CALL PRINTB(NELEM,NESQ,AA,CP,4)
C
        CALL FORCE(NNODE,XK,NELEM,NODE,CP)
C
    999 CONTINUE
   1000 STOP
        END
```

```
C
      SUBROUTINE DATA(NELEM)
C
C
C     DATE CREATED :   MARCH 31, 1975
C
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSEX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      COMMON/ZZZ32/CSTAG,CVORT
      COMMON/ZZZ34/YCUT
      COMMON/ZZZ33/UWAKE
      COMMON/ZZZ50/XCTR,YCTR,ZCTR,RX,RY,RZ
      COMMON/ZZZ51/HSPAN,XLEZH,XTEZH,TANLEH,TANTEH,TAUH,ZP
      COMMON/ZZZ35/ITMIST,RPITCH,TMIST
      DIMENSION ACHORD(20),WNPP(2),WNPM(2),WNMP(2),WNMM(2)
      DIMENSION GG(20),KSFACE(20)
      DIMENSION TCHORD(10),TAXIS(10),VCHORD(10),VAXIS(10)
      REAL LSHAFT,LSHANK
C
      NSEX= 0
      DO 19 I=1,3
      READ(NREAD,10)  GG
19    WRITE(NWRITE,20)GG
      READ(NREAD,10)  GG(1),GG(2),KREAD
      WRITE(NWRITE,20)GG(1),GG(2),KREAD
10    FORMAT(20A4)
20    FORMAT(1X,20A4)
```

```
C
C     USE KSYMMY=0
C     USE KSYMMZ=0
C
      READ(NREAD,18)   GG(1),GG(2),NS,KSYMMY,KSYMMZ
      WRITE(NWRITE,28)GG(1),GG(2),NS,KSYMMY,KSYMMZ
      READ(NREAD,18)   GG(1),GG(2),NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,
     1 NSHAFT,NHUB,NSHANK,NBLADE
      WRITE(NWRITE,28)GG(1),GG(2),NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,
     1 NSHAFT,NHUB,NSHANK,NBLADE
      READ(NREAD,18)   GG(1),GG(2),KPYL1,KPYL2
      WRITE(NWRITE,28)GG(1),GG(2),KPYL1,KPYL2
18    FORMAT(2A4,10I5)
28    FORMAT(1X,2A4,10I5)
      READ(NREAD,12)   GG(1),GG(2),UMACH,OMEGA,AREA
      WRITE(NWRITE,22)GG(1),GG(2),UMACH,OMEGA,AREA
      READ(NREAD,12) GG(1),GG(2),ALFA,ABETA
      WRITE(NWRITE,22)GG(1),GG(2),ALFA,ABETA
12    FORMAT(2A4,7F8.3)
22    FORMAT(1X,2A4,7F10.3)
      IF(NPYLON.EQ.1)
     1READ(NREAD,12)   GG(1),GG(2),XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      IF(NPYLON.EQ.1)
     1WRITE(NWRITE,22)GG(1),GG(2),XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      IF(NVTAIL.EQ.1)
     1 READ(NREAD,12)   GG(1),GG(2),VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAUV
     2 ,ZPV
      IF(NVTAIL.EQ.1)
     1WRITE(NWRITE,22) GG(1),GG(2),VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAUV
     2 ,ZPV
C
      NTBD=NBODY1+NBODY2+NBODY3
      NROTOR=NSHAFT+NHUB+NSHANK+NBLADE
C
      IF(NTBD.NE.0)
     1READ(NREAD,12)GG(1),GG(2),XNOSE,XBD1,XBD2,XTAIL
      IF(NTBD.NE.0)
     1WRITE(NWRITE,22)GG(1),GG(2),XNOSE,XBD1,XBD2,XTAIL
      IF(NTBD.NE.0)
     1READ(NREAD,12)   GG(1),GG(2),YNOSE,YBD1,YBD2,YTAIL
      IF(NTBD.NE.0)
     1WRITE(NWRITE,22)GG(1),GG(2),YNOSE,YBD1,YBD2,YTAIL
      IF(NTBD.NE.0)
     1READ(NREAD,12)   GG(1),GG(2),ZNOSE,ZBD1,ZBD2,ZTAIL
      IF(NTBD.NE.0)
     1WRITE(NWRITE,22)GG(1),GG(2),ZNOSE,ZBD1,ZBD2,ZTAIL
      IF(NTBD.NE.0)
     1READ(NREAD,12)   GG(1),GG(2),RYBD1,RZBD1,RYBD2,RZBD2
      IF(NTBD.NE.0)
     1WRITE(NWRITE,22)GG(1),GG(2),RYBD1,RZBD1,RYBD2,RZBD2
C
      IF(NSHAFT.EQ.1)READ(NREAD,12)   GG(1),GG(2),RSHAFT,LSHAFT
      IF(NSHAFT.EQ.1)WRITE(NWRITE,22)GG(1),GG(2),RSHAFT,LSHAFT
      IF(NHUB.EQ.1)
     1 READ(NREAD,12)GG(1),GG(2),XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      IF(NHUB.EQ.1)
     1WRITE(NWRITE,22)GG(1),GG(2),XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
```

```
      IF(NSHANK.EQ.1)   READ(NREAD,12)GG(1),GG(2),RSHANK,LSHANK
      IF(NSHANK.EQ.1)WRITE(NWRITE,22)GG(1),GG(2),RSHANK,LSHANK
      IF(NBLADE.EQ.1) READ(NREAD,12) GG(1),GG(2),RROTOR,YCUT,XBLE,XBTE,
     1 TANBLE,TANBTE,TAUBL
      IF(NBLADE.EQ.1)WRITE(NWRITE,22)GG(1),GG(2),RROTOR,YCUT,XBLE,XBTE,
     1 TANBLE,TANBTE,TAUBL
      TANLEB=TANBLE
      TANTEB=TANBTE
      IF(NBLADE.EQ.1)   READ(NREAD,12)GG(1),GG(2),THET75,THET1C,THET1S,
     1 CONING,AZIMUTH
      IF(NBLADE.EQ.1)WRITE(NWRITE,22)GG(1),GG(2),THET75,THET1C,THET1S,
     1 CONING,AZIMUTH
      IF(NBLADE.EQ.1)   READ(NREAD,12)GG(1),GG(2),RPITCH,TWIST
      IF(NBLADE.EQ.1)WRITE(NWRITE,22)GG(1),GG(2),RPITCH,TWIST
      IF(NBLADE.EQ.1)   READ(NREAD,18)GG(1),GG(2),KBLADE,ITWIST
      IF(NBLADE.EQ.1)WRITE(NWRITE,28)GG(1),GG(2),KBLADE,ITWIST
      READ(NREAD,13)   GG(1),GG(2),NWAKPY,NWAKHB,NWAKSK,NWAKBL
      WRITE(NWRITE,23)GG(1),GG(2),NWAKPY,NWAKHB,NWAKSK,NWAKBL
      IF((NWAKHB+NWAKSK+NWAKBL).NE.0)
     1 READ(NREAD,16)   GG(1),GG(2),NSPIRAL,SPIRAL,UWAKE
      IF((NWAKHB+NWAKSK+NWAKBL).NE.0)
     1 WRITE(NWRITE,26)GG(1),GG(2),NSPIRAL,SPIRAL,UWAKE
      IF(NWAKPY.EQ.1)   READ(NREAD,18)GG(1),GG(2),NSTAG,NVORT
      IF(NWAKPY.EQ.1)WRITE(NWRITE,28)GG(1),GG(2),NSTAG,NVORT
      IF(NWAKPY.EQ.1)   READ(NREAD,12)GG(1),GG(2),CSTAG,CVORT
      IF(NWAKPY.EQ.1)WRITE(NWRITE,22)GG(1),GG(2),CSTAG,CVORT
   16 FORMAT(    2A4,I5,2F8.3)
   26 FORMAT(1X,2A4,I5,2F8.3)
      READ(NREAD,19)   GG(1),GG(2),(KPRINT(K),K=1,10)
      WRITE(NWRITE,29)GG(1),GG(2),(KPRINT(K),K=1,10)
C
      IF(UMACH.GE. 1.0) CALL DEBUG(100)
      MULTY=1
      IF(KSYMMY.EQ.0)MULTY=2
      MULT=1
      IF(KSYMMZ.EQ.0)MULT=2
      MS=0
      IF(NIBD.NE.0)MS=MS+(NBODY1+NBODY2+NBODY3)*MULT*MULTY
      IF(NPYLON.NE.0.OR.NVTAIL.NE.0)MS=MS+(NPYLON+NVTAIL)*MULTY
      IF(NWUB.NE.0) MS=MS+NSWFIT*MULTY+NWUB*MULT*MULTY
      IF (NROTOR.NE.0) MS=MS+KBLADE*(NSHANK+NBLADE*MULT)
C     IF(MS.NE.MS) CALL DEBUG(200)
C
C
      REFLEN=1.
      BETA=SQRT(ABS(UMACH**2-1.))
      KS=0
      NSBODY=0
      NSBTOT=0
C
C
C ---------------- NOSE -------------------------------------
C
      IF(NBODY1.EQ.0)GO TO 192
      KS=KS+1
      NSBODY=NSBODY+1
      READ(NREAD,18)   GG(1),GG(2),NX(KS),NY(KS),KNSELE,KNSSHP,KNSTYP
```

```
      WRITE(NWRITE,28)GG(1),GG(2),NX(KS),NY(KS),KNSELE,KNSSHP,KNSTYP
      NXY(KS)=NX(KS)*NY(KS)
      ISFACE(KS)=1
      KNORML(KS)=1
      KWAKES(KS)=0
      KROTORS(KS)=0
C
      IF(KSYMMY.NE.0)GO TO 170
      NSBODY=NSBODY+1
      JS=KS+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
      NX(JS)=NX(KS)
      NY(JS)=NY(KS)
      NXY(JS)=NXY(KS)
      ISFACE(JS)=9
      KWAKES(JS)=0
      KNORML(JS)=-1
      KROTORS(JS)=0
170   CONTINUE
      IF(KSYMMZ.NE.0)GO TO 199
      NSBODY=NSBODY+1
      KSL=KS+NBODY1+NBODY2+NBODY3+NPYLON
      NX(KSL)=NX(KS)
      NY(KSL)=NY(KS)
      NXY(KSL)=NXY(KS)
      ISFACE(KSL)=5
      KWAKES(KSL)=0
      KNORML(KSL)=-1
      KROTORS(KSL)=0
C
      IF(KSYMMY.NE.0)GO TO 199
      NSBODY=NSBODY+1
      JSL=KSL+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
      NX(JSL)=NX(KS)
      NY(JSL)=NY(KS)
      NXY(JSL)=NXY(KS)
      ISFACE(JSL)=13
      KWAKES(JSL)=0
      KNORML(JSL)=1
      KROTORS(JSL)=0
199   CONTINUE
C
C     ============== FUSLAGE ==============================
C
      IF(NBODY2.EQ.0)GO TO 299
      KS=KS+1
      NSBODY=NSBODY+1
      READ(NREAD,18)  GG(1),GG(2),NX(KS),NY(KS),KBDELE,KBDSHP,KBDTYP
      WRITE(NWRITE,28)GG(1),GG(2),NX(KS),NY(KS),KBDELE,KBDSHP,KBDTYP
      NXY(KS)=NX(KS)*NY(KS)
      ISFACE(KS)=2
      KNORML(KS)=1
      KROTORS(KS)=0
      KWAKES(KS)=0
C
      IF(KSYMMY.NE.0)GO TO 270
      NSBODY=NSBODY+1
      JS=KS+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
```

```
      NX(JS)=NX(KS)
      NY(JS)=NY(KS)
      NXY(JS)=NXY(KS)
      ISFACE(JS)=10
      KWAKES(JS)=0
      KNORML(JS)=-1
      KROTORS(JS)=0
 270  CONTINUE
      IF(KSYMMZ.NE.0)GO TO 299
      NSBODY=NSBODY+1
      KSL=KS+NBODY1+NBODY2+NBODY3+NPYLON
      NX(KSL)=NX(KS)
      NY(KSL)=NY(KS)
      NXY(KSL)=NXY(KS)
      ISFACE(KSL)=6
      KWAKES(KSL)=0
      KNORML(KSL)=-1
      KROTORS(KSL)=0
C
      IF(KSYMMY.NE.0)GO TO 299
      NSBODY=NSBODY+1
      JSL=KSL+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
      NX(JSL)=NX(KS)
      NY(JSL)=NY(KS)
      NXY(JSL)=NXY(KS)
      ISFACE(JSL)=14
      KWAKES(JSL)=0
      KNORML(JSL)=-1
      KROTORS(JSL)=0
 299  CONTINUE
C
C  --------------- AFT-BODY -------------------------------
C
      IF(NBODY3.EQ.0)GO TO 399
      KS=KS+1
      NSBODY=NSBODY+1
      READ(NREAD,19)   GG(1),GG(2),NX(KS),NY(KS),KTNELE,KTNSHP,KTNTYP
      WRITE(NWRITE,29)GG(1),GG(2),NX(KS),NY(KS),KTNELE,KTNSHP,KTNTYP
      NXY(KS)=NX(KS)*NY(KS)
      ISFACE(KS)=3
      KNORML(KS)=1
      KWAKES(KS)=0
      KROTORS(KS)=0
C
      IF(KSYMMY.NE.0)GO TO 370
      NSBODY=NSBODY+1
      JS=KS+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
      NX(JS)=NX(KS)
      NY(JS)=NY(KS)
      NXY(JS)=NXY(KS)
      ISFACE(JS)=11
      KWAKES(JS)=0
      KNORML(JS)=-1
      KROTORS(JS)=0
 370  CONTINUE
      IF(KSYMMZ.NE.0)GO TO 399
      NSBODY=NSBODY+1
```

```
      KSL=KS+NBODY1+NBODY2+NBODY3+NPYLON
      NX(KSL)=NX(KS)
      NY(KSL)=NY(KS)
      NXY(KSL)=NXY(KS)
      ISFACE(KSL)=7
      KWAKES(KSL)=0
      KNORML(KSL)=-1
      KROTORS(KSL)=0
C
      IF(KSYMMY.NE.0)GO TO 399
      NSBODY=NSBODY+1
      JSL=KSL+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
      NX(JSL)=NX(KS)
      NY(JSL)=NY(KS)
      NXY(JSL)=NXY(KS)
      ISFACE(JSL)=15
      KWAKES(JSL)=0
      KNORML(JSL)=-1
      KROTORS(JSL)=0
  399 CONTINUE
C
C ---------------- PYLON ------------------------------------
C
      IF(NPYLON.EQ.0)GO TO 499
      KS=KS+1
      NSBODY=NSBODY+1
      READ(NREAD,18)   GG(1),GG(2),NX(KS),NY(KS),KPYELE,KPYSHP,KPYTYP
      WRITE(NWRITE,28)GG(1),GG(2),NX(KS),NY(KS),KPYELE,KPYSHP,KPYTYP
      NXY(KS)=NX(KS)*NY(KS)
      ISFACE(KS)=4
      KNORML(KS)=1
      KWAKES(KS)=1
      KROTORS(KS)=0
C
      IF(KSYMMY.NE.0)GO TO 499
      NSBODY=NSBODY+1
      JS=KS+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
      NX(JS)=NX(KS)
      NY(JS)=NY(KS)
      NXY(JS)=NXY(KS)
      ISFACE(JS)=12
      KWAKES(JS)=1
      KNORML(JS)=-1
      KROTORS(JS)=0
  499 CONTINUE
C
      NBODYS=NBODY1+NBODY2+NBODY3
C
C ---------------- VERTICAL TAIL ----------------------------
C
      NSBODY=NSBODY+1
      IF(NVTAIL.EQ.0)GO TO 599
      KS=NBODYS*MULT+NPYLON+1
      READ(NREAD,18)   GG(1),GG(2),NX(KS),NY(KS),KVTELE,KVTSHP,KVTTYP
      WRITE(NWRITE,28)GG(1),GG(2),NX(KS),NY(KS),KVTELE,KVTSHP,KVTTYP
      NXY(KS)=NX(KS)*NY(KS)
       ISFACE(KS)=8
```

```
      KNORML(KS)=-1
      KWAKES(KS)=1
      KROTORS(KS)=0
C
      IF(KSYMMY.NE.0)GO TO 599
      NSBODY=NSBODY+1
      JS=KS+(NBODY1+NBODY2+NBODY3)*MULT+NPYLON+NVTAIL
      NX(JS)=NX(KS)
      NY(JS)=NY(KS)
      NXY(JS)=NXY(KS)
      ISFACE(JS)=16
      KWAKES(JS)=1
      KNORML(JS)=+1
      KROTORS(JS)=0
C
599   CONTINUE
C
C  ------------- ROTOR SHAFT -------------
C
      IF(NSHAFT.EQ.0)GO TO 699
      NSBODY=NSBODY+1
      KS=(NBODY1+NBODY2+NBODY3)*MULT*MULTY+(NPYLON+NVTAIL)*MULTY
      KS=KS+1
      READ(NREAD,13)  GG(1),GG(2),NX(KS),NY(KS),KSHELE,KSHSHP,KSHTYP
      WRITE(NWRITE,28)GG(1),GG(2),NX(KS),NY(KS),KSHELE,KSHSHP,KSHTYP
      NXY(KS) = NX(KS)*NY(KS)
      ISFACE(KS) = 17
      KNORML(KS)=1
      KWAKES(KS)=0
      KROTORS(KS)=1
C
      IF(KSYMMY.NE.0)GO TO 699
      NSBODY=NSBODY+1
      JS=KS+1
      NX(JS)=NX(KS)
      NY(JS)=NY(KS)
      NXY(JS)=NXY(KS)
      ISFACE(JS)=18
      KWAKES(JS)=0
      KNORML(JS)=-1
      KROTORS(JS)=1
C
699   CONTINUE
C
C  -------------- ROTOR HUB --------------
C
      IF(NHUB.EQ.0)GO TO 799
      KS=NBODYS*MULT*MULTY+(NPYLON+NVTAIL+NSHAFT)*MULTY
      KS=KS+1
      NSBODY=NSBODY+1
      READ(NREAD,13) GG(1),GG(2),NX(KS),NY(KS),KHBELE,KHBSHP,KHBTYP
      WRITE(NWRITE,29)GG(1),GG(2),NX(KS),NY(KS),KHBELE,KHBSHP,KHBTYP
      NXY(KS)=NX(KS)*NY(KS)
      ISFACE(KS)=13
      KNORML(KS)=1
      KWAKES(KS)=1
      KROTORS(KS)=1
```

```
C
      IF(KSYMMY.NE.0)GO TO 770
      NSBODY=NSBODY+1
      JS=KS+NHUB*MULT
      NX(JS)=NX(KS)
      NY(JS)=NY(KS)
      NXY(JS)=NXY(KS)
      ISFACE(JS)=21
      KWAKES(JS)=1
      KNORML(JS)=-1
      KROTORS(JS)=1
  770 CONTINUE
      IF(KSYMMZ.NE.0)GO TO 799
      NSBODY=NSBODY+1
      KSL=KS+1
      NX(KSL)=NX(KS)
      NY(KSL)=NY(KS)
      NXY(KSL)=NXY(KS)
      ISFACE(KSL)=20
      KWAKES(KSL)=1
      KNORML(KSL)=-1
      KROTORS(KSL)=1
C
      IF(KSYMMY.NE.0)GO TO 799
      NSBODY=NSBODY+1
      JSL=JS+NHUB*MULT
      JSL=KSL+NHUB*MULT
      NX(JSL)=NX(KS)
      NY(JSL)=NY(KS)
      NXY(JSL)=NXY(KS)
      ISFACE(JSL)=22
      KWAKES(JSL)=1
      KNORML(JSL)=1
      KROTORS(JSL)=1
  799 CONTINUE
C
C ---------------- ROTOR BLADE SHANK ----------------
C
      IF(NSHANK.EQ.0)GO TO 899
      KS=(NBODY1+NBODY2+NBODY3)*MULT*MULTY+(NRYLON+NVTAIL)*MULTY+
     1NSHAFT*MULTY+NHUB*MULT*MULTY+1
      READ(NREAD,10)  GG(1),GG(2),NX(KS),NY(KS),KSKELE,KSKSHP,KSKTYP
      WRITE(NWRITE,20)GG(1),GG(2),NX(KS),NY(KS),KSKELE,KSKSHP,KSKTYP
C
      DO 810 K=1,KBLADE
      NSBODY=NSBODY+1
      KSS=KS+(K-1)*(NSHANK+NBLADE*MULT)
      NX(KSS)=NX(KS)
      NY(KSS)=NY(KS)
      NXY(KSS)=NX(KS)*NY(KS)
      ISFACE(KSS)=23+(K-1)*3
      KNORML(KSS)=1
      KWAKES(KSS)=1
      KROTORS(KSS)=1
  810 CONTINUE
C
  899 CONTINUE
```

```
C
C  --------------- ROTOR BLADE ---------------
C
      IF(NBLADE.EQ.0)GO TO 999
      KS=   (NBODY1+NBODY2+NBODY3)*MULT*MULTY+(NPYLON+NVTAIL)*MULTY+
     1 NSHAFT*MULTY+NHUB*MULT*MULTY+NSHANK+1
      READ(NREAD,10)   GG(1),GG(2),NX(KS),NY(KS),KBLELE,KBLSHP,KBLTYP
      WRITE(NWRITE,20)GG(1),GG(2),NX(KS),NY(KS),KBLELE,KBLSHP,KBLTYP
C
      DO 910 K=1,KBLADE
      NSBODY=NSBODY+1
      KSS=KS+(K-1)*(NSHANK+NBLADE*MULT)
      NX(KSS)=NX(KS)
      NY(KSS)=NY(KS)
      NXY(KSS)=NX(KS)*NY(KS)
      ISFACE(KSS)=24+(K-1)*3
      KNORML(KSS)=1
      KWAKES(KSS)=1
      KROTORS(KSS)=1
C
      IF(KSYMMZ.NE.0)GO TO 910
      NSBODY=NSBODY+1
      KSSL=KSS+1
      NX(KSSL)=NX(KSS)
      NY(KSSL)=NY(KSS)
      NXY(KSSL)=NXY(KSS)
      ISFACE(KSSL)=25+(K-1)*3
      KWAKES(KSSL)=1
      KNORML(KSSL)=-1
      KROTORS(KSSL)=1
910   CONTINUE
C
999   CONTINUE
C
C
      NELEM=0
      WRITE(6,8002)
      DO 1000 IS=1,NS
1000  NELEM=NELEM+NXY(IS)
      WRITE(6,8001)NELEM
 8001 FORMAT(//5X,*NELEM = *,I5/)
C
8002  FORMAT(/10X,4HDATA)
C
      NSBTOT=0
      DO 1100 IS=1,NS
      NT(IS)=NSBTOT
      NSBTOT=NSBTOT+NX(IS)*NY(IS)
      WRITE(6,8003) IS,NX(IS),NY(IS),NXY(IS),ISFACE(IS),NT(IS),
     1KNORML(IS),KWAKES(IS)
8003  FORMAT (5X,*IS= *,I5,2X,*NX=*,I5,2X,*NY=*,I5,
     1   *  NXY=*,I5,*  ISFACE=*,I5,*  NT=*,I5,
     2   *  KNORML=*,I5,*  KWAKES=*,I5)
1100  CONTINUE
C
C
      RETURN
      END
```

```
C
      SUBROUTINE COORPT(NELEM,NXYMP,NOFCT,XK,NNODE)
C
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NBYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAUV,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      COMMON/ZZZ33/UWAKE
      COMMON/ZZZ34/YCUT
      COMMON/ZZZ50/XCTR,YCTR,ZCTR,RX,RY,RZ
      COMMON/ZZZ51/HSPAN,XLEZH,XTEZH,TANLEH,TANTEH,TAUH,ZP
      COMMON/ZZZ35/ITWIST,RPITCH,TWIST
      DIMENSION ACHORD(20),WNPP(2),WNPM(2),WNMP(2),WNMM(2)
      DIMENSION XK(3,NNODE),NOFCT(NXYMP,34),GG(20),KSFACE(34)
      DIMENSION DFPP(2),DFPM(2),DFMP(2),DFMM(2)
      DIMENSION HCHORD(11),HAXIS(11),VCHORD(11),VAXIS(11)
      DIMENSION HCSI(11),HETA(11)
      REAL LSHAFT,LSHANK
C
      MSFX=34
      KREAD=0
C
      IF(KREAD.EQ.0)GO TO 9
      READ(NREAD,15)  ((XK(K,INODE),K=1,3),INODE=1,NNODE)
      WRITE(NWRITE,25)((XK(K,INODE),K=1,3),INODE=1,NNODE)
    8 FORMAT(3E12.4)
      GO TO 900
    9 CONTINUE
C
```

```
      REFLEN=1.
      BETA=SQRT(ABS(UMACH**2-1.))
      MULT=1
      IF(KSYMMZ.EQ.0)MULT=2
      MULTY=1
      IF(KSYMMY.EQ.0)MULTY=2
   15 FORMAT(10F8.3)
   25 FORMAT(1X,10F8.3)
      PI=3.14159265
  300 FORMAT(2X,*IS=*,I5,2X,*IX=*,I5,2X,*IY=*,I5,2X,*IXY=*,I5,2X,
     1 *NOFCT=*,I5)
C
C
      KFACE=NBODY1+NBODY2+NBODY3+NPYLON
C
C     ========== PYLON ==========
C
      IF(NPYLON.EQ.0)GO TO 10
      KS=KFACE
      DXX=1./NX(KS)
      DYY=1./NY(KS)
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      LPYELE=KPYELE
      IF(LPYELE.NE.0) GO TO 102
      IF(LPYELE.NE.0) GO TO 102
      LPYELE=1
      READ(NREAD,15)    (HCSI(IX),IX=1,NXP)
      WRITE(NWRITE,25) (HCSI(IX),IX=1,NXP)
      READ(NREAD,15)    (HETA(IY),IY=1,NYP)
      WRITE(NWRITE,25) (HETA(IY),IY=1,NYP)
  102 CONTINUE
      DO 118 IX=1,NXP
      DO 118 IY=1,NYP
      XX=(IX-1)*DXX
      YY=(IY-1)*DYY
      IF(KPYELE.EQ.0) XX=HCSI(IX)
      IF(KPYELE.EQ.0) YY=HETA(IY)
      GO  TO(131,132,133),LPYELE
 131  CONTINUE
      CSI=XX
      ETA=YY
      GO TO 113
 132  CONTINUE
      CSI=XX*XX
      ETA=YY
 133  CONTINUE
      CSI=1.-(1.-XX)**2
      ETA=YY
 113  CONTINUE
      THETA=PI*ETA
      RX=CSI*RXPYL
      RY=CSI*RYPYL
      X0=RX*COS(THETA)
      Y0=RY*SIN(THETA)
      Z0=RZPYL*SQRT(1.-(X0/RXPYL)**2-(Y0/RYPYL)**2)
      IXY=IX+(IY-1)*NXP
```

```
      INODE=NOFCT(IXY,KS)
      XK(1,INODE)=X0+XPYCTR
      XK(2,INODE)=Y0+YPYCTR
      XK(3,INODE)=Z0+ZPYCTR
      WRITE(6,140)IX,IY,IXY,CSI,ETA,INODE
      WRITE(6,141)THETA,RX,RY
      WRITE(6,142)X0,Y0,Z0
      WRITE(6,143)(XK(L,INODE),L=1,3)
  140 FORMAT(10X,*IX=*,I5,2X,*IY=*,I5,2X,*IXY=*,I5,2X,*CSI=*,E12.6,2X,
     1*ETA=*,E12.6,2X,*INODE=*,I5)
  141 FORMAT(10X,*THETA=*,E12.5,2X,*RX=*,E12.6,2X,*RY=*,E12.6)
  142 FORMAT(10X,*X0=*,E12.6,2X,*Y0=*,E12.6,2X,*Z0=*,E12.6)
  143 FORMAT(10X,*XK=*,3(E12.6,2X))
 115  CONTINUE
 199  CONTINUE
C
C
      KS=0
C
C     ===========NOSE===========================
C
      IF(NBODY1.EQ.0)GO TO 599
      KS=KS+1
      DO 598 IBODY1=1,MULT
      IF(IBODY1.EQ.1)SIGNZ=+1.
      IF(IBODY1.EQ.2)SIGNZ=-1.
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      DXX=1./NX(KS)
      TINCR=0.5*PI/NY(KS)
      DO 550 IX=1,NXP
      XX=(IX-1)*DXX
      IF(KNSELE.EQ.1)CSI=XX
      IF(KNSELE.EQ.2)CSI=XX*XX
      X=XNOSE+(XBD1-XNOSE)*CSI
      IF(KNSSHP.EQ.1)RRY=RYBD1*SQRT(CSI)
      IF(KNSSHP.EQ.1)RRZ=RZBD1*SQRT(CSI)
      IF(KNSSHP.EQ.2)RRY=RYBD1*(CSI)**.3333333
      IF(KNSSHP.EQ.2)RRZ=RZBD1*(CSI)**.3333333
      IF(KNSSHP.EQ.3)RRY=RYBD1*(CSI)**.25
      IF(KNSSHP.EQ.3)RRZ=RZBD1*(CSI)**.25
      DO 550 IY=1,NYP
      THET=(IY-1)*TINCR
      THETA=(+0.5*PI-THET)
      ETA=COS(THETA)
      YY=RRY*ETA
      ZZ=SIGNZ*RRZ*SIN(THETA)
      DELZ=(ZBD1-ZNOSE)
      DOWNZ=DELZ-DELZ*CSI
      ZZ=ZZ-DOWNZ
      ZZ=ZZ+ZBD1
C
C
      IXY=IX+(IY-1)*NXP
      IF(IBODY1.EQ.1)IS=KS
      IF(IBODY1.EQ.2)IS=KS+KFACE
      WRITE(6,140)IX,IY,IXY,CSI,ETA,NOFCT(IXY,IS)
```

```
      WRITE(6,142)X,YY,ZZ
      XK(1,NOFCT(IXY,IS))=X
      XK(2,NOFCT(IXY,IS))=YY
      XK(3,NOFCT(IXY,IS))=ZZ
550   CONTINUE
598   CONTINUE
599   CONTINUE
C
C     ----------BODY------------------------
C
      IF(NBODY2.EQ.0)GO TO 69
      KS=KS+1
      DO 698 IBODY2=1,MULT
      IF(IBODY2.EQ.1)SIGNZ=+1.
      IF(IBODY2.EQ.2)SIGNZ=-1.
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      TINCR=0.5*PI/NY(KS)
      DXX=1./NX(KS)
      DO 631 IX=1,NXP
      XXX=(IX-1)*DXX
      IF(KBDELE.EQ.1)CSI=XXX
      IF(KBDELE.EQ.2)CSI=XXX*XXX
      R1=RYBD1+(RYBD2-RYBD1)*CSI
      R2=RZBD1+(RZBD2-RZBD1)*CSI
      DELZ=ZBD2-ZBD1
      ZZZ=DELZ*CSI
      XX=XBD1+(XBD2-XBD1)*CSI
      DO 631 IY=1,NYP
      THETA=(0.5*PI      -(IY-1)*TINCR)
      YY=R1*COS(THETA)
      ZZ=SIGNZ*R2*SIN(THETA)
      ZZ=ZZ+ZZZ+ZBD1
      IXY=IX+(IY-1)*NXP
      IF(IBODY2.EQ.1)IS=KS
      IF(IBODY2.EQ.2)IS=KS+KFACE
      XK(1,NOFCT(IXY,IS))=XX
      XK(2,NOFCT(IXY,IS))=YY
      XK(3,NOFCT(IXY,IS))=ZZ
631   CONTINUE
698   CONTINUE
699   CONTINUE
C
C
C     ----------TAIL NOSE---- ----------------
C
      IF(NBODY3.EQ.0)GO TO 899
      KS=KS+1
      DO 898 IBODY3=1,MULT
      IF(IBODY3.EQ.1)SIGNZ=+1.
      IF(IBODY3.EQ.2)SIGNZ=-1
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      DXX=1./NX(KS)
      TINCR=0.5*PI/NY(KS)
      DO 858 IX=1,NXP
      XX=(IX-1)*DXX
```

```
      XX=1.-XX
      IF(KTNELE.EQ.1)CSI=XX
      IF(KTNELE.EQ.2)CSI=XX*XX
      IF(KTNSHP.EQ.1)RRY=RYBD2*SQRT(CSI)
      IF(KTNSHP.EQ.1)RRZ=RZBD2*SQRT(CSI)
      IF(KTNSHP.EQ.2)RRY=RYBD2*CSI
      IF(KTNSHP.EQ.2)RRZ=RZBD1*CSI
      CSI=1.-CSI
      X=XBD2+(XTAIL-XBD2)*CSI
      DO 850 IY=1,NYP
      THET=(IY-1)*TINCR
      THETA=(+0.5*PI-THET)
      ETA=COS(THETA)
      YY=RRY*ETA
      ZZ=SIGNZ*RRZ*SIN(THETA)
C
C
      DELZ=ZTAIL-ZBD2
      DOWNZ=CELZ*CSI
      ZZ=ZZ+DOWNZ+ZBD2
C
C
      IXY=IX+(IY-1)*NXP
      IF(IBODY3.EQ.1)IS=KS
      IF(IBODY3.EQ.2)IS=KS+KFACE
      XK(1,NOFCT(IXY,IS))=X
      XK(2,NOFCT(IXY,IS))=YY
      XK(3,NOFCT(IXY,IS))=ZZ
850   CONTINUE
898   CONTINUE
899   CONTINUE
C
C     ---------- VERTICAL TAIL -----------
C
      IF(NVTAIL.EQ.0)GO TO 399
      KS=(NBODY1+NBODY2+NBODY3)*MULI+NPYLON+1
      DO 398 IVTL=1,1
C
      IF(KVTTYP.EQ.1)GO TO 3170
      READ(NREAD,15)  GG(1),GG(2),WNPP,WNPM,WNMP,WNMM
      WRITE(NWRITE,25)GG(1),GG(2),WNPP,WNPM,WNMP,WNMM
3170  CONTINUE
      SIGNZ=+1.
      TAILLN=VSPAN-2.*R
      TAILWD=TAILLN/2.
      RLE=0.
      RTE=0.
      DXX=1./NX(KS)
      DYY=1./NY(KS)
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      LVTELE=KVTELE
      IF(LVTELE.NE.0)GO TO 3130
      LVTELE=1
      READ(NREAD,15)  GG(1),GG(2),(HCSI(IX),IX=1,NXP)
      WRITE(NWRITE,25)GG(1),GG(2),(HCSI(IX),IX=1,NXP)
      READ(NREAD,15)  GG(1),GG(2),(HETA(IY),IY=1,NYP)
```

```
      WRITE(NWRITE,25)GG(1),GG(2),(HETA(IY),IY=1,NYP)
3100  CONTINUE
      DO 318 IX=1,NXP
      DO 318 IY=1,NYP
      XX=(IX-1)*DXX
      YY=(IY-1)*DYY
      IF(KVTELE.EQ.0)XX=HCSI(IX)
      IF(KVTELE.EQ.0)YY=HETA(IY)
      GO TO(3131,3132,3133,3134),LVTELE
3131  CONTINUE
      CSI=XX
      ETA=YY
      GO TO 313
3132  CONTINUE
      CSI=XX*XX
      ETA=1.-(1.-YY)**2
      GO TO 313
3133  CONTINUE
      CSI=XX
      ETA=1.-(1.-YY)**2
      GO TO 313
3134  CONTINUE
      CSI=XX*XX
      ETA=YY
313   CONTINUE
C     IF(KVTTYP.NE.1)GO TO 3139
      Y=TAILWD*ETA
      XLE=XLEZV+TANLEV*(Y-RLE)
      XTE=XTEZV+TANTEV*(Y-RTE)
      VCHORD(IY)=XTE-XLE
      VAXIS(IY)=XLE+VCHORD(IY)*0.5
      X=XLE+(XTE-XLE)*CSI
C
3139  CONTINUE
      IF(KVTTYP.NE.2)GO TO 31 9
      X1=(WNPP(1)-WNMP(1))*CSI+WNMP(1)
      Y1=(WNPP(2)-WNMP(2))*CS.+WNMP(2)
      X2=(WNPM(1)-WNMM(1))*CSI+WNMM(1)
      Y2=(WNPM(2)-WNMM(2))*CSI+WNMM(2)
      X3=(WNPP(1)-WNPM(1))*ETA+WNPM(1)
      Y3=(WNPP(2)-WNPM(2))*ETI+WNPM(2)
      X4=(WNMP(1)-WNMM(1))*ETI+WNMM(1)
      Y4=(WNMP(2)-WNMM(2))*ETA+WNMM(2)
      DET=(X1-X2)*(Y3-Y4)-(X3-X4)*(Y1-Y2)
      IF(DET.EQ.0.)GO TO 3147
      X=((X1-X2)*(X4*Y3-X3*Y4)-(X3-X4)*(X2*Y1-X1*Y2))/DET
      Y=((Y1-Y2)*(X4*Y3-X3*Y4)-(Y3-Y4)*(X2*Y1-X1*Y2))/DET
      GO TO 3149
3147  CONTINUE
      X=X4
      Y=Y4
3149  CONTINUE
C
      GO TO (3151,3152,3153),KWISHA
3151  CONTINUE
C
C     FOR CIRCULAR BICONVEX
```

```
C
      XC=0.5*(XLE+XTE)
      PL=0.5*(XTE-XLE)
      H=TAUV*PL
      ATAU=2.*H
      ETACR=1.-ATAU/VSPAN
      XXC=X-XC
      Z=0.
      IF(H.EQ.0..OR.IY.EQ.NYP.OR.IX.EQ.1.OR.IX.EQ.NXP)GO TO 315
      Z=SIGNZ*(SQRT(((PL*PL+H*H)/(2.*H))**2-XXC**2)-(PL*PL-H*H)/(2.*H))
      IF(ETA.GE.ETACR)Z=Z*SQRT(1.-((ETA-ETACR)*VSPAN/ATAU)**2)
      GO TO 315
3152  CONTINUE
      TAUBAR=TAUV*.75*SQRT(3.)*(XTEZV-XLEZV)
      Z=SIGNZ*TAUBAR*SQRT(CSI)*(1.-CSI)*SQRT(1.-ETA**2)
      GO TO 315
3153  CONTINUE
      Z=SIGNZ*4.*TAUV*(XTEZV-XLEZV)*CSI*(1.-CSI)*SQRT(1.-ETA**2)
315   CONTINUE
C
      IXY=IX+(IY-1)*NXP
      XK(1,NOFCT(IXY,KS))=X
      XK(2,NOFCT(IXY,KS))=Z
      XK(3,NOFCT(IXY,KS))=Y+ZVTAIL
318   CONTINUE
398   CONTINUE
399   CONTINUE
C
C     ---------  SHAFT --------
C
      IF(NSHAFT.EQ.0)GO TO 299
      KS=(NBODY1+NBODY2+NBODY3)*MULT*MULTY+(NPYLON+NVTAIL)*MULTY
      KS=KS+1
      DXX=1./NX(KS)
      DYY=1./NY(KS)
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      LSHELE=KSHELE
      IF(LSHELE.NE.0) GO TO 2:1
      IF(LSHELE.NE.0)GO TO 201
      LSHELE=1
      READ(NREAD,15)  (HCSI(IX),IX=1,NXP)
      WRITE(NWRITE,25)(HCSI(IX),IX=1,NXP)
      READ(NREAD,15)  (HETA(IY),IY=1,NYP)
      WRITE(NWRITE,25)(HETA(IY),IY=1,NYP)
  201 CONTINUE
      DO 218 IX=1,NXP
      DO 218 IY=1,NYP
      XX=(IX-1)*DXX
      YY=(IY-1)*DYY
      IF(KSHELE.EQ.0) XX=HCSI(IX)
      IF(KSHELE.EQ.0) YY=HETA(IY)
      GO TO (231,232,233),LSHELE
231   CONTINUE
      CSI=XX
      ETA=YY
      GO TO 213
```

```
232   CONTINUE
      CSI=XX*XX
      ETA=YY
      GO TO 213
233   CONTINUE
      CSI=1.-(1.-XX)**2
      ETA=YY
213   CONTINUE
      THETA=PI*ETA
      R=RSHAFT
      X0=R*COS(THETA)
      Y0=R*SIN(THETA)
      Z0=CSI*LSHAFT
      IXY=IX+(IY-1)*NXP
      INODE=NOFCT(IXY,KS)
      XK(1,INODE)=X0+XPYCTR
      XK(2,INODE)=Y0+YPYCTR
      XK(3,INODE)=ZPYCTR+LSHAFT-Z0+RZPYL
      WRITE(6,61)KS,IX,IY,IXY,INODE
      WRITE(6,62)X0,Y0,Z0
      WRITE(6,63)(XK(L,INODE),L=1,3)
   61 FORMAT(10X,*KS=*,I5,*  IX=*,I5,*  IY=*,I5,*  IXY=*,I5,
     1 *  INODE=*,I5)
   62 FORMAT(10X,*X0=*,E12.6,*  Y0=*,E12.6,*  Z0=*,E12.6)
   63 FORMAT(10X,* XK= *,3(E12.6,2X))
210   CONTINUE
299   CONTINUE
C
      IF(NHUB.EQ.0) GO TO 499
      KS=(NBODY1+NBODY2+NBODY3)*MULT*MULTY+(NPYLON+NVTAIL)*MULTY+
     1 NSHAFT*MULTY
      KS=KS+1
      DO 498 IMULT=1,MULT
C
      IF(IMULT.EQ.1) SIGNZ= 1
      IF(IMULT.EQ.2) SIGNZ=-1.
C
      DXX=1./NX(KS)
      DYY=1./NY(KS)
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      LHBELE=KHBELE
      IF(LHBELE.NE.0)GO TO 400
      LHBELE=1
      READ(NREAD,15)   (HCSI(IX),IX=1,NXP)
      WRITE(NWRITE,25)(HCSI(IX),IX=1,NXP)
      READ(NREAD,15)   (HETA(IY),IY=1,NYP)
      WRITE(NWRITE,25)(HETA(IY),IY=1,NYP)
400   CONTINUE
      DO 410 IX=1,NXP
      DO 410 IY=1,NYP
      XX=(IX-1)*DXX
      YY=(IY-1)*DYY
      IF(KHBELE.EQ.0) XX=HCSI(IX)
      IF(KHBELE.EQ.0) YY=HETA(IY)
      GO TO (431,432,433),LHBELE
431   CONTINUE
```

```
      CSI=XX
      ETA=YY
      GO TO 413
432   CONTINUE
      CSI=XX*XX
      ETA=YY
      GO TO 413
433   CONTINUE
      CSI=1.-(1.-XX)**2
      ETA=YY
413   CONTINUE
      THETA=PI*ETA
      RX=CSI*RXHUB
      RY=CSI*RYHUB
      X0=RX*COS(THETA)
      Y0=RY*SIN(THETA)
      Z0=SIGNZ*RZHUB*SQRT(1.- X0/RXPYL)**2-(Y0/RYPYL)**2)
      IXY=IX+(IY-1)*NXP
      IS=KS+(IMULT-1)
      INODE=NOFCT(IXY,IS)
      XK(1,INODE)=X0+XHUBCR
      XK(2,INODE)=Y0+YHUBCR
      XK(3,INODE)=Z0+ZHUBCR
      WRITE(6,61)IS,IX,IY,IXY,INODE
      WRITE(6,62)XJ,Y0,Z0
      WRITE(6,63)(XK(L,INODE),L=1,3)
418   CONTINUE
498   CONTINUE
499   CONTINUE
C
C
900   CONTINUE
C
      NSS=(NBODY1+NBODY2+NBODY3)*MULT+NPYLON
     1 +NVTAIL
      DO 910 IS=1,NSS
      JS=IS+NSS
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 910 IX=1,NXP
      DO 910 IY=1,NYP
      IXY=IX+(IY-1)*NXP
      XK(1,NOFCT(IXY,JS))=XK(1,NOFCT(IXY,IS))
      XK(2,NOFCT(IXY,JS))=-XK(2,NOFCT(IXY,IS))
      XK(3,NOFCT(IXY,JS))=XK(3,NOFCT(IXY,IS))
      WRITE(6,54)IXY,IS
   54 FORMAT(10X,*IXY=*,I5,2X.*IS=*,I5)
910   CONTINUE
C
      IF(NSHAFT.EQ.0) GO TO 911
      IS=NSS*MUL TY+1
      JS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 912 IX=1,NXP
      DO 912 IY=1,NYP
      IXY=IX+(IY-1)*NXP
```

```
      WRITE(6,54)IXY,IS
      XK(1,NOFCT(IXY,JS))= XK(1,NOFCT(IXY,IS))
      XK(2,NOFCT(IXY,JS))=-XK(2,NOFCT(IXY,IS))
      XK(3,NOFCT(IXY,JS))= XK(3,NOFCT(IXY,IS))
  912 CONTINUE
  911 CONTINUE
C
      IF(NHUB.EQ.0) GO TO 913
      IS1=NSS*MULTY+NSHAFT*MULTY+1
      IS2=IS1+1
      DO 914 IS=IS1,IS2
      JS=IS+MULTY
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 915 IX=1,NXP
      DO 915 IY=1,NYP
      IXY=IX+(IY-1)*NXP
      WRITE(6,54)IXY,IS
      XK(1,NOFCT(IXY,JS))= XK(1,NOFCT(IXY,IS))
      XK(2,NOFCT(IXY,JS))=-XK(2,NOFCT(IXY,IS))
      XK(3,NOFCT(IXY,JS))= XK(3,NOFCT(IXY,IS))
  915 CONTINUE
  914 CONTINUE
  913 CONTINUE
C
C
C
C     ========= BLADE  SHANK =========
C
      IF(NSHANK.EQ.0)GO TO 799
      KS=(NBODY1+NBODY2+NBODY3)*MULT*MULTY+(NPYLON+NVTAIL)*MULTY+
     1 NHUB*MULT*MULTY+NSHAFT*MULTY+1
      DO 798 IB=1,KBLADE
C
      IF(IB.EQ.1)SIGNY=1.
      IF(IB.EQ.2)SIGNY=-1.
C
      DXX=1./NX(KS)
      DYY=1./NY(KS)
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      LSKELE=KSKELE
      IF(LSKELE.NE.0)GO TO 700
      LSKELE=1
      READ(NREAD,15)   (HCSI(IX);IX=1,NXP)
      WRITE(NWRITE,25)(HCSI(IX),IX=1,NXP)
      READ(NREAD,15)   (HETA(IY);IY=1,NYP)
      WRITE(NWRITE,25)(HETA(IY),IY=1,NYP)
  700 CONTINUE
      DO 718 IX=1,NXP
      DO 718 IY=1,NYP
      XX=(IX-1)*DXX
      YY=(IY-1)*DYY
      IF(KSKELE.EQ.0) XX=HCSI(IX)
      IF(KSKELE.EQ.0) YY=HETA(IX)
  731 CONTINUE
      CSI=XX
```

```
      ETA=YY
      GO TO 713
732   CONTINUE
      CSI=1.-(1.-XX)**2
      ETA=YY
      GO TO 713
733   CONTINUE
      CSI=XX*XX
      ETA=YY
713   CONTINUE
      THETA=2.*PI*ETA
      R=RSHANK
      X0=R*COS(THETA)
      Z0=R*SIN(THETA)
      Y0=CSI*LSHANK
C
      IXY=IX+(IY-1)*NXP
      IS=KS+(IB-1)*(NBLADE*MULT+NSHANK)
      INODE=NOFCT(IXY,IS)
      XK(1,INODE)=XHUBCR+X0
      XK(2,INODE)=SIGNY*(YHUBCR+RYHUB+LSHANK-Y0)
      XK(3,INODE)=ZHUBCR+Z0
      WRITE(6,61)IS,IX,IY,IXY,INODE
      WRITE(6,62)X0,Y0,Z0
      WRITE(6,63)(XK(L,INODE),L=1,3)
718   CONTINUE
798   CONTINUE
799   CONTINUE
C
C
C
C
C     --------- ROTOR BLADE --------
C
      IF(NBLADE.EQ.0) GO TO 1039
      KKS=(NBODY1+NBODY2+NBODY3)*MULT*MULTY+(NPYLON+NVTAIL)*MULTY
     1 +NSHAFT*MULTY+NHUB*MULT*MULTY+NSHANK
      DO 1098 IB=1,KBLADE
      KS=KKS+1+(IB-1)*(NBLADE*MULT+NSHANK)
      DO 1097 IMULT=1,MULT
      WRITE(6,55)IB,IMULT,IXY,IS,X,Y,Z
C
      SIGNY=1.
      IF(IMULT.EQ.1)SIGNZ=1.
      IF(IMULT.EQ.2)SIGNZ=-1.
C
      DXX=1./NX(KS)
      DYY=1./NY(KS)
      NXP=NX(KS)+1
      NYP=NY(KS)+1
      LBLELE=KBLELE
      IF(LBLELE.NE.0)GO TO 1088
      LBLELE=1
      READ(NREAD,15)   (HCSI(IX),IX=1,NXP)
      WRITE(NWRITE,25)(HCSI(IX),IX=1,NXP)
      READ(NREAD,15)   (HETA(IY),IY=1,NYP)
      WRITE(NWRITE,25)(HETA(IY),IY=1,NYP)
```

```
1000  CONTINUE
      DO 1010 IX=1,NXP
      DO 1010 IY=1,NYP
      XX=(IX-1)*DXX
      YY=(IY-1)*DYY
      IF(KBLELE.EQ.0) XX=HCSI(IX)
      IF(KBLELE.EQ.0) YY=HETA(IY)
      GO TO (1031,1032,1033,1034),LBLELE
1031  CONTINUE
      CSI=XX
      ETA=YY
      GO TO 1013
1032  CONTINUE
      CSI=XX*XX
      ETA=1.-(1.-YY)**2
      GO TO 1013
1033  CONTINUE
      CSI=XX
      ETA=1.-(1.-YY)**2
      GO TO 1013
1034  CONTINUE
      CSI=XX*XX
      ETA=YY
1013  CONTINUE
      BCHORD=XBTE-XBLE
      XLEZ=XBLE
      XTEZ=XBTE
      TANBLE=TANLEB
      TANBTE=TANTEB
      SPAN=2.*RROTOR
      Y=(RROTOR-YCUT)*ETA+YCUT
      XLE=XLEZ+TANBLE*(Y-YCUT)
      XTE=XTEZ+TANBTE*(Y-YCUT)
      FCHORD=XTE-XLE
      X=XLE+FCHORD*CSI
C
C
      GO TO (1051,1052,1053),K3LSHP
1051  CONTINUE
C
C      FOR CIRCULAR BICONVEX
C
      XC=0.5*(XLE+XTE)
      PL=0.5*(XTE-XLE)
      TAUBL=.050
      TAU=TAUBL
      H=TAU*PL
      ATAU=2.*H
      ETACR=1.-ATAU/SPAN
      XXC=X-XC
      Z=0.
      IF(H.EQ.0..OR.IY.EQ.NYP.OR.IX.EQ.1.OR.IX.EQ.NXP)GO TO 115
      Z=SIGNZ*(SQRT(((PL*PL+H*H)/(2.*H))**2-XXC**2)-(PL*PL-H*H)/(2.*H))
      IF(ETA.GE.ETACR)Z=Z*SQRT(1.-((ETA-ETACR)*SPAN/ATAU)**2)
      GO TO 115
1052  CONTINUE
      TAUBAR=TAU*.75*SQRT(3.)*(XTEZ-XLEZ)
```

```
      Z=SIGNZ*TAUBAR*SQRT(CSI)*(1.-CSI)*SQRT(1.-ETA**2)
      GO TO 115
 1053 CONTINUE
      Z=SIGNZ*4.*TAU*(XTEZ-XLEZ)*CSI*(1.-CSI)*SQRT(1.-ETA**2)
  115 CONTINUE
C
      IXY=IX+(IY-1)*NXP
      IS=KS
      IF(IMULT.EQ.2)IS=KS+1
      INODE=NOECT(IXY,IS)
      XK(1,INODE)=X
      XK(2,INODE)=Y*SIGNY
      XK(3,INODE)=Z+ZHUBCR
C
      AZIM=AZIMUTH*PI/180.
      GO TO (301,302),ITWIST
C
  301 CONTINUE
      PITCHR=RPITCH*PI/180.
      PITCHT=(RPITCH+TWIST)*PI/180.
      X1=YCUT
      X2=RROTOR
      XX=XK(1,INODE)
      YY=XK(2,INODE)
      ZZ=XK(3,INODE)
      C1=(YY-X2)/(X1-X2)
      C2=(YY-X1)/(X2-X1)
      DX=X2-X1
      PITCH=C1*PITCHR+C2*PITCHT
      GO TO 303
  302 CONTINUE
      PITCHR=RPITCH*PI/180.
      PIT75=(THET75+THET1C*COS(AZIM)+THET1S*SIN(AZIM))*PI/180.
      X1=YCUT
      X2=.75*RROTOR
      XX=XK(1,INODE)
      YY=XK(2,INODE)
      ZZ=XK(3,INODE)
      C1=(YY-X2)/(X1-X2)
      C2=(YY-X1)/(X2-X1)
      DX=X2-X1
      PITCH=C1*PITCHR+C2*PIT75
  303 CONTINUE
C
C BLADE TWIST
C
      COSPIT=COS(PITCH)
      SINPIT=SIN(PITCH)
C
      XK1=XK(1,INODE)
      XK3=XK(3,INODE)
      XK(1,INODE)= XK1*COSPIT+XK3*SINPIT
      XK(3,INODE)=-XK1*SINPIT+XK3*COSPIT
C
C CONING ANGLE
C
      CONE=CONING*PI/180.
```

```
      COSC=COS(CONE)
      SINC=SIN(CONE)
C
      XK2=XK(2,INODE)
      XK3=XK(3,INODE)
      XK(2,INODE)= XK2*COSC-XK3*SINC
      XK(3,INODE)= XK2*SINC+XK3*COSC
C
C AZIMUTH
C
      HZIM=(AZIMUTH-90.)*PI/180.
      COSHZ=COS(HZIM)
      SINHZ=SIN(HZIM)
C
      XK1=XK(1,INODE)
      XK2=XK(2,INODE)
      XK(1,INODE)=XK1*COSHZ-XK2*SINHZ
      XK(2,INODE)= XK1*SINHZ+XK2*COSHZ
C
      ANG=2.*PI/KBLADE
      ANGR=ANG*(IB-1)
C
      XK1=XK(1,INODE)
      XK2=XK(2,INODE)
C
      XK(1,INODE)= XK1*COS(ANGR)-XK2*SIN(ANGR)
      XK(2,INODE)= XK1*SIN(ANGR)+XK2*COS(ANGR)
C
      WRITE(6,55)IB,IMULT,IXY,IS,X,Y,Z
   55 FORMAT(10X,*IB=*,I5,2X,*IMULT=*,I5,2X,*IXY=*,I5,2X,*IS=*,I5,2X,
     1 *XYZ=*,3(E12.6,2X))
 1010 CONTINUE
 1097 CONTINUE
 1098 CONTINUE
 1099 CONTINUE
C
      WRITE(6,60)
   60 FORMAT(10X,*=== END OF COORDT ===*)
C
      RETURN
      END
```

```
C
      SUBROUTINE CHECK(NELEM,NXMAX,NYMAX,NTMAX)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSEX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(18),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      REAL LSHAFT,LSHANK
C
      DO 100 IS=1,NS
      IF(NX(IS).GT.NXMAX) CALL DEBUG(300)
      IF(NY(IS).GT.NYMAX) CALL DEBUG(300)
  100 CONTINUE
      IF(NELEM.GT.NTMAX) CALL DEBUG(500)
      WRITE(6,60)
   60 FORMAT(10X,*--- END OF CHECK --- *)
      RETURN
      END
```

```
C
      SUBROUTINE PREPRO(NNODE,NXYMP,NOFCT,NELEM,NTWAKE)
C
C     THIS SUBROUTINE FUNCTIONED AS A PRE-PROCESSER. ON RETURN,
C     THE CO-ORDINATES OF NODES IN THE GLOBAL NUMBERING SYSTEM
C     WILL BE RE-CONSTRUCTED TO GIVE CO-ORDINATES IN THE LOCAL
C     NUMBERING SYSTEM FOR INDIVIDUAL SURFACES.
C
C
      INTEGER PP,PM,MP,MM
      DIMENSION NOFCT(NXYMP,34)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAR
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAK4B,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAK 4B,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANG4B,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPYL1,KPYL2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
C
C
C     DO 24 I=1,NS
C24   WRITE(6,234)NX(I),NY(I),NT(I),KNORML(I),ISFACE(I)
CC
      MULT=1
      IF (KSYMMZ.EQ.0) MULT=2
      MULTY=1
      IF(KSYMMY.EQ.0)MULTY=2
CC
CC
CC
CC
      NTBODY=NBODY1+NBODY2+NBODY3
```

```
CC
      NODEX=0
      NODEU=0
      IF(NTBODY.EQ.0) GO TO 10
      NODEX=1
      DO 20 I=1,NTBODY
   20 NODEX=NODEX+NX(I)
   10 CONTINUE
CC
      IF(NTBODY.NE.0)
     1NODEU=NODEU+NODEX*(NY(NTBODY)+1) -(NTBODY-1)*NY(NTBODY)
CC
CC  NODEX=NUMBER OF NODES ALONG AIRCRAFT CENTERLINE
CC
CC    NODEU=NUMBER OF NODES ON UPPER RHS FUSELAGE BODY
CC
CC
CC
      IS=0
      NODE0=0
      NLAST=0
C
      IF(NBODY1.EQ.0)GO TO 99
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 80 IY=1,NYP
      DO 80 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1.AND.IY.GT.1)GO TO 85
      NODE0=NODE0+1
85    CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=1
      NOFCT(IXY,IS)=INODE
80    CONTINUE
99    CONTINUE
C
      NLAST=NODE0
      IF(NBODY2.EQ.0)GO TO 199
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 180 IY=1,NYP
      DO 180 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1)GO TO 185
      NODE0=NODE0+1
185   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=(NX(1)+1)+(IY-1)*NX(1)
      NOFCT(IXY,IS)=INODE
180   CONTINUE
199   CONTINUE
C
```

```
      NLAST=NODE0
      IF(NBODY3.EQ.0)GO TO 299
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 280 IY=1,NYP
      DO 280 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1)GO TO 285
      IF(IX.EQ.NXP.AND.IY.GT.1)GO TO 285
      NODE0=NODE0+1
285   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=
     1 NX(1)*(NY(1)+1)+1+
     2 NX(2)+(IY-1)*NX(2)
      IF(IX.EQ.NXP)INODE=
     1 NX(1)*(NY(1)+1)+1+
     2 NX(2)*(NY(2)+1)+
     3 NX(3)
      NOECT(IXY,IS)=INODE
280   CONTINUE
299   CONTINUE
C
      NLAST=NODE0
C
      IF(NPYLON.EQ.0)GO TO 399
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 380 IY=1,NYP
      DO 380 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1.AND.IY.GT.1) GO TO 385
      IF(IX.EQ.NXP)GO TO 385
      NODE0=NODE0+1
385   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=
     1 NLAST+1
      IF(IX.EQ.NXP) INODE= NX(1)*(NY(1)+1)+1+NYP-(IY-1)
      NOECT(IXY,IS)=INODE
380   CONTINUE
399   CONTINUE
      NLAST=NODE0
C
      IF(KSYMMZ.NE.0)GO TO 499
      IF(NBODY1.EQ.0) GO TO 499
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 480 IY=1,NYP
      DO 480 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
```

```
      IF(IX.EQ.1) GO TO 485
      IF(IY.EQ.NYP) GO TO 485
      NODE0=NODE0+1
485   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=1
      IF(IX.NE.1.AND.IY.EQ.NYP)INODE=
     1 NX(1)*NY(1)+IX
      NOFCT(IXY,IS)=INODE
480   CONTINUE
499   CONTINUE
C
      NLAST=NODE0
C
      IF(KSYMMZ.NE.0) GO TO 599
      IF(NBODY2.EQ.0)GO TO 599

      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 580 IY=1,NYP
      DO 580 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1)GO TO 585
      IF(IY.EQ.NYP)GO TO 585
      NODE0=NODE0+1
585   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1.AND.IY.NE.NYP)INODE=
     1 NLAST-NX(1)*(NY(1)-2)-(NX(1)-1)-1+(IY-1)*NX(1)
      IF(IX.EQ.1.AND.IY.EQ.NYP)INODE=
     1 NX(1)*(NY(1)+1)+1
      IF(IY.EQ.NYP.AND.IX.NE.1)INODE=
     1 NX(1)*(NY(1)+1)+1+
     2 NX(2)*NY(2)-1+IX
      NOFCT(IXY,IS)=INODE
580   CONTINUE
599   CONTINUE
C
      NLAST=NODE0
      IF(KSYMMZ.NE.0)GO TO 699
      IF(NBODY3.EQ.0)GO TO 699
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 680 IY=1,NYP
      DO 680 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1) GO TO 685
      IF(IY.EQ.NYP)GO TO 685
      IF(IX.EQ.NXP)GO TO 685
      NODE0=NODE0+1
685   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1.AND.IY.NE.NYP) INODE=
```

```
     1 NLAST-NX(2)*(NY(2)-2)-(NX(2)-1)-1+(IY-1)*NX(2)
      IF(IY.EQ.NYP.AND.(IX.NE.1.OR.IX.NE.NXP))INODE=
     1 NX(1)*(NY(1)+1)+1+
     2 NX(2)*(NY(2)+1)+
     3 (NX(3)-1)*NY(3)+1+(IX-1)
      IF(IX.EQ.NXP)INODE=
     1 NX(1)*(NY(1)+1)+1+
     2 NX(2)*(NY(2)+1)+
     3 NX(3)
      IF(IX.EQ.1 .AND. IY.EQ.NYP)INODE=
     1 NX(1)*(NY(1)+1)+1+NX(2)*(NY(2)+1)
      NOFCT(IXY,IS)=INODE
680   CONTINUE
690   CONTINUE
C
      NLAST=NODE0
      NODET=NLAST
      IF(NVTAIL.EQ.0)GO TO 79
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 780 IY=1,NYP
      DO 780 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      NODE0=NODE0+1
      INODE=NODE0+NLAST
      NOFCT(IXY,IS)=INODE
780   CONTINUE
790   CONTINUE
C
      NLAST=NODE0
      IF(KSYMMY.NE.0) GO TO 899
      IF(NBODY1.EQ.0)GO TO 899
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 880 IY=1,NYP
      DO 880 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1) GO TO 885
      IF(IY.EQ.1) GO TO 885
      NODE0=NODE0+1
885   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=1
      IF(IY.EQ.1)INODE=IX
      NOFCT(IXY,IS)=INODE
880   CONTINUE
899   CONTINUE
C
      NLAST=NODE0
      IF(KSYMMY.NE.0) GO TO 999
      IF(NBODY2.EQ.0)GO TO 999
      IS=IS+1
      NXP=NX(IS)+1
```

```
      NYP=NY(IS)+1
      KPYL1=1
      KPYL2=2
      K1=KPYL1+2
      K2=K1+NX(4)-2
      DO 980 IY=1,NYP
      DO 980 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(NPYLON.NE.0) GO TO 950
      IF(IX.EQ.1)GO TO 985
      IF(IY.EQ.1)GO TO 985
      NODE0=NODE0+1
985   CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=
     1 NLAST-NX(1)*(NY(1)-1)-(NX(1)-1)-1+(IY-1)*NX(1)
      IF(IY.EQ.1)INODE=
     2 NX(1)*(NY(1)+1)+1+(IX-1)
      IF(IX.EQ.1.AND.IY.EQ.1)INODE=
     1 NX(1)+1
      GO TO 949
  950 CONTINUE
      IF(IX.EQ.1) GO TO 951
      IF(IY.EQ.1.AND.(IX.LT.K1.OR.IX.GT.K2)) GO TO 951
      NODE0=NODE0+1
  951 CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=NLAST-NX(1)*(NY(1)-1)-(NX(1)-1)-1+(IY-1)*NX(1)
      IF(IY.EQ.1.AND.(IX.LT.K1.OR.IX.GT.K2))INODE=
     1 NX(1)*(NY(1)+1)+1+(IX-1)
      IF(IX.EQ.1.AND.IY.EQ.1)INODE=NX(1)+1
  949 CONTINUE
      NOFCT(IXY,IS)=INODE
980   CONTINUE
999   CONTINUE
C
      NLAST=NODE0
      IF(KSYMMY.NE.0) GO TO 1039
      IF(NBOCY3.EQ.0)GO TO 1033
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1080 IY=1,NYP
      DO 1080 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1)GO TO 1085
      IF(IY.EQ.1)GO TO 1085
      IF(IX.EQ.NXP)GO TO 1085
      NODE0=NODE0+1
1085  CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=
     1 NLAST-NX(2)*(NY(2)-1)+(IY-2)*NX(2)
      IF(IY.EQ.1)INODE=
     1 NX(1)*(NY(1)+1)+1+NX(2 *(NY(2)+1)+(IX-1)
```

```
      IF(IX.EQ.1.AND.IY.EQ.1)INODE=
     1 NX(1)*(NY(1)+1)+1+NX(2)
      IF(IX.EQ.NXP) INODE=
     1 NX(1)*(NY(1)+1)+1+
     2 NX(2)*(NY(2)+1)+NX(3)
      NOECT(IXY,IS)=INODE
1080  CONTINUE
1099  CONTINUE
C
      NLAST=NODE0
      IF(KSYMMY.NE.0) GO TO 1199
      IF(NPYLON.EQ.0)GO TO 1199
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1180 IY=1,NYP
      DO 1180 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IY.EQ.1)GO TO 1185
      IF(IX.EQ.1.OR.(IX.EQ.NXP))GO TO 1185
      IF(IY.EQ.NYP)GO TO 1185
      NODE0=NODE0+1
1185  CONTINUE
      INODE=NODE0
      IF(IY.EQ.1)INODE=NODEU+
      IF(IX.EQ.1)INODE=NODEU+1
      IF(IY.EQ.NYP.AND.IX.NE.1)INODE=NODEU+1+(NXP-2)*(NYP-1)+1+(IX-2)
      IF(IX.EQ.NXP)INODE=NLAST-(NX(3)-1)*NY(3)-NX(2)*NY(2)+1-(IY-1)
      IF(IY.EQ.1.AND.IX.EQ.NXP)INODE=NX(1)*(NY(1)+1)+1+NY(4)+1
      IF(IY.EQ.NYP.AND.IX.EQ.NXP)INODE=NX(1)*(NY(1)+1)+1+1
      NOECT(IXY,IS)=INODE
1180  CONTINUE
1199  CONTINUE
C
      NLAST=NODE0
      IF(KSYMMZ.NE.0) GO TO 1299
      IF(NBODY1.EQ.0)GO TO 1299
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1280 IY=1,NYP
      DO 1280 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IY.EQ.1)GO TO 1285
      IF(IX.EQ.1)GO TO 1285
      IF(IY.EQ.NYP)GO TO 1285
      NODE0=NODE0+1
1285  CONTINUE
      INODE=NODE0
      PSS=0.
      IF(NPYLON.EQ.0)PSS=1.
      IF(PSS.EQ.1.)GO TO 1290
      IF(IX.EQ.1)INODE=1
      IF(IX.NE.1.AND.IY.EQ.1)INODE=
     1 NODEU+
```

```
      2 (NX(4)-1)*(NY(4)+1)+IX
       IF(IX.NE.1.AND.IY.EQ.NYP) INODE=NLAST-
      1 NX(4)*(NY(4)-1)-(NX(3)-1)*NY(3)-
      2 NX(2)*NY(2)-NX(1)+(IX-1)
        GO TO 1250
 1290  CONTINUE
        IF(IX.EQ.1)INODE=1
        IF(IX.NE.1.AND.IY.EQ.1)INODE=
      1 NODEU+(IX-1)
        IF(IX.NE.1.AND.IY.EQ.NYP) INODE=
      1NODEU+NX(1)*NY(1)+NX(2)*NY(2)+(NX(3)-1)*NY(3)+
      2 NX(1)*(NY(1)-1)+(IX-1)
 1250  CONTINUE
        NOECT(IXY,IS)=INODE
 1280  CONTINUE
 1299  CONTINUE
C
        NLAST=NODE0
        IF(KSYMMZ.NE.0)GO TO 1399
        IF(NBODY2.EQ.0)GO TO 1333
        IS=IS+1
        NXP=NX(IS)+1
        NYP=NY(IS)+1
        DO 1380 IY=1,NYP
        DO 1380 IX=1,NXP
        IXY=IX+(IY-1)*NXP
C
        IF(IY.EQ.1)GO TO 1385
        IF(IX.EQ.1)GO TO 1385
        IF(IY.EQ.NYP)GO TO 1385
        NODE0=NODE0+1
 1385  CONTINUE
        INODE=NODE0
        PSS=0.
        IF(NPYLON.EQ.0)PSS=1.
        IF(PSS.EQ.1.)GO TO 1390
        IF(IX.EQ.1.AND.(IY.NE.1.OR.IY.NE.NYP)) INODE=
      1 NLAST-NX(1)*(NY(1)-2)+(IY-2)*NX(1)
        IF(IX.EQ.1.AND.IY.EQ.1)INODE=
      1 NODEU+(NX(4)-1)*(NY(4)+1)+1+NX(1)
        IF(IX.GT.1.AND.IY.EQ.1)INODE=
      1 NODEU+(NX(4)-1)*(NY(4)+1)+1+
      2 NX(1)*NY(1)+(IX-1)
        IF(IX.EQ.1.AND.IY.EQ.NYP)INODE=
      1 NLAST-NX(1)*(NY(1)-1)-NX(4)*(NY(4)-1)-
      2 (NX(3)-1)*NY(3)-NX(2)*NY(2)
        IF(IX.GT.1.AND.IY.EQ.NYP)INODE=
      1 NLAST-NX(1)*(NY(1)-1)-
      2 (NX(4)-1)*(NY(4)-1)-(NX(3)-1)*NY(3)-
      3 NX(2)+(IX-1)
        GO TO 1350
 1390  CONTINUE
        IF(IX.EQ.1.AND.(IY.NE.1.OR.IY.NE.NYP)) INODE=
      1 NLAST-NX(1)-(NX(1)-1)-1+(IY-2)*NX(1)
      2 +NX(1)
        IF(IX.EQ.1.AND.IY.EQ.1)INODE=
      1 NODEU+NX(1)
```

```
      IF(IY.EQ.1.AND.IX.NE.1)INODE=
     1 NODEU+NX(1)*NY(1)+(IX-1)
      IF(IX.EQ.1.AND.IY.EQ.NYP) INODE=
     1 NLAST-NX(1)*(NY(1)-1)-(NX(3)-1)*NY(3)-NX(2)*NY(2)
      IF(IX.NE.1.AND.IY.EQ.NYP)INODE=
     1 NLAST-NX(1)*(NY(1)-1)-(NX(3)-1)*NY(3)-NX(2)+(IX-1)
 1350 CONTINUE
      NOECT(IXY,IS)=INODE
 1360 CONTINUE
 1390 CONTINUE
      NLAST=NODE0
      IF(NBODY3.EQ.0)GO TO 1433
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1480 IY=1,NYP
      DO 1480 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1)GO TO 1485
      IF(IY.EQ.1)GO TO 1485
      IF(IY.EQ.NYP)GO TO 1485
      IF(IX.EQ.NXP)GO TO 1485
      NODE0=NODE0+1
 1485 CONTINUE
      INODE=NODE0
      PSS=0.
      IF(NPYLON.EQ.0)PSS=1.
      IF(PSS.EQ.1.)GO TO 1490
      IF(IX.EQ.1)INODE=NLAST-
     1 NX(2)+(IY-2)*NX(2)
      IF(IX.EQ.1.AND.IY.EQ.1)INODE=
     1 NODEU+(NX(4)-1)*(NY(4)+1)+1+
     2 NX(1)*NY(1)+NX(2)
      IF(IX.NE.1.AND.IY.EQ.1)INODE=
     1 NODEU+(NX(4)-1)*(NY(4)+1)+1+NX(1)*NY(1)+
     2 NX(2)*NY(2)+(IX-1)
      IF(IY.EQ.NYP.AND.IX.EQ.1)INODE=
     1 NLAST-NX(2)*(NY(2)-1)-NX(1)*(NY(1)-1)-
     2 (NX(4)-1)*(NY(4)+1)-NX(3)*NY(3)
     3 +NX(3)
      IF(IY.EQ.NYP.AND.IX.NE.1)INODE=
     1 NLAST-NX(2)*(NY(2)-1)-NX(1)*(NY(1)-1)-
     2 (NX(4)-1)*(NY(4)-1)-(NX(3)-1)+(IX-1)
      IF(IX.EQ.NXP)INODE=
     1 NX(1)*(NY(1)+1)+NX(2)*(NY(2)+1)+NX(3)+1
      GO TO 1450
 1490 CONTINUE
      IF(IY.EQ.1)INODE=
     1 NODEU+NX(1)*NY(1)+NX(2)*NY(2)+(IX-1)
      IF(IY.EQ.NYP)INODE=
     1 NLAST-NX(2)*(NY(2)-1)-NX(1)*(NY(1)-1)-NX(3)+IX
      IF(IX.EQ.NXP)INODE=
     1 NODEU-(NX(3)-1)*NY(3)
      IF(IX.EQ.1)INODE=
     1 NLAST-NX(2)-(NX(2)-1)-1+(IY-2)*NX(2)
     2  +NX(2)
```

```
      IF(IX.EQ.1.AND.IY.EQ.1)INODE=
     1 NODEU+NX(1)*NY(1)+NX(2)
      IF(IX.EQ.1.AND.IY.EQ.NYP) INODE=
     1 NLAST-NX(2)*(NY(2)-1)-NX(1)*(NY(1)-1)-(NX(3)-1)*NY(3)
 1450 CONTINUE
      NOFCT(IXY,IS)=INODE
1480  CONTINUE
1499  CONTINUE
      NLAST=NODE0
      IF(NYTAIL.EQ.0)GO TO 1593
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1580 IY=1,NYP
      DO 1580 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1.OR.IX.EQ.NXP) GO TO 1585
      IF(IY.EQ.NYP) GO TO 1585
      NODE0=NODE0+1
1585  CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=
     1 NODET+(IY-1)*(NX(8)+1)
      IF(IX.EQ.NXP) INODE=
     1 NODET+1+NX(8)+(IY-1)*(NX(8)+1)
      IF(IY.EQ.NYP) INODE=
     1 NODET+(NX(8)+1)*NY(8)+IX
      NOFCT(IXY,IS)=INODE
1580  CONTINUE
1599  CONTINUE
C
      NLAST=NODE0
      NBODY=NODE0
      IF(NSHAFT.EQ.0)GO TO 16-3
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1680 IY=1,NYP
      DO 1680 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      NODE0=NODE0+1
      INODE=NODE0
      NOFCT(IXY,IS)=INODE
1680  CONTINUE
1699  CONTINUE
C
      NLAST=NODE0
      IF(KSYMMY.NE.0) GO TO 1739
      IF(NSHAFT.EQ.0)GO TO 1733
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1780 IY=1,NYP
      DO 1780 IX=1,NXP
      IXY=IX+(IY-1)*NXP
```

```
C
      IF(IY.EQ.1.OR.IY.EQ.NYP)GO TO 1785
      NODE8=NODE8+1
 1785 CONTINUE
      INODE=NODE8
      IF(IY.EQ.1)INODE=
     1 NBODY+IX
      IF(IY.EQ.NYP)INODE=
     1 NBODY+(NX(IS)+1)*NY(IS)+IX
      NOFCT(IXY,IS)=INODE
 1780 CONTINUE
 1700 CONTINUE
C
      NLAST=NODE8
      IF(NHUB.EQ.0)GO TO 1899
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1880 IY=1,NYP
      DO 1880 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1.AND.IY.GT.1)GO TO 1885
      NODE8=NODE8+1
 1885 CONTINUE
      INODE=NODE8
      IF(IX.EQ.1)INODE=NLAST+1
      NOFCT(IXY,IS)=INODE
 1880 CONTINUE
 1899 CONTINUE
C
      NLAST=NODE8
      IF(KSYMMZ.NE.0) GO TO 1999
      IF(NHUB.EQ.0)GO TO 1999
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 1980 IY=1,NYP
      DO 1980 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1.AND.IY.GT.1)GO TO 1985
      IF(IX.EQ.NXP)GO TO 1985
      NODE8=NODE8+1
 1985 CONTINUE
      INODE=NODE8
      IF(IX.EQ.1)INODE=NBODY+
     1 (NX(15)+1)*(NY(15)+1)+
     2 (NX(16)+1)*(NY(16)-1)+
     3 NX(17)*(NY(17)+1)+1+1
      IF(IX.EQ.NXP) INODE=NBODY+
     1 (NX(15)+1)*(NY(15)+1)+
     2 (NX(16)+1)*(NY(16)-1)+
     3 (NX(17)+1)+(IY-1)*NX(17)
      NOFCT(IXY,IS)=INODE
 1980 CONTINUE
 1999 CONTINUE
```

```
C
      NLAST=NODE0
      IF(KSYMMY.NE.0) GO TO 2199
      IF(NHUB.EQ.0)GO TO 2099
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2080 IY=1,NYP
      DO 2080 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1)GO TO 2085
      IF(IY.EQ.1.OR.IY.EQ.NYP) GO TO 2085
      NODE0=NODE0+1
2085  CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=NBODY+
     1 (NX(15)+1)*(NY(15)+1)+
     2 (NX(16)+1)*(NY(16)-1)+
     3 1
      IF(IY.EQ.1.AND.IX.NE.1)INODE=
     1 NBODY+(NX(15)+1)*(NY(1 )+1)+
     2 (NX(16)+1)*(NY(16)-1)+
     3 IX
      IF(IY.EQ.NYP.AND.IX.NE.1) INODE=
     1 NBODY+(NX(15)+1)*(NY(1 )+1)+
     2 (NX(16)+1)*(NY(16)-1)+
     3 NX(17)*NY(17)+IX
      NOFCT(IXY,IS)=INODE
2080  CONTINUE
2099  CONTINUE
C
      NLAST=NODE0
      IF(KSYMMZ.NE.0) GO TO 2199
      IF(NHUB.EQ.0)GO TO 2199
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2180 IY=1,NYP
      DO 2180 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1)GO TO 2185
      IF(IY.EQ.1.OR.IY.EQ.NYP) GO TO 2185
      IF(IX.EQ.NXP)GO TO 2185
      NODE0=NODE0+1
2185  CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=NBODY+
     1 (NX(15)+1)*(NY(15)+1)+
     2 (NX(16)+1)*(NY(16)-1)+
     3 NX(17)*(NY(17)+1)+1+
     4 1
      IF(IY.EQ.1.AND.IX.NE.1)INODE=NBODY+
     1 (NX(15)+1)*(NY(15)+1)+
     2 (NX(16)+1)*(NY(16)-1)+
     3 NX(17)*(NY(17)+1)+1+
```

```
    4 IX
      IF(IY.EQ.NYP.AND.IX.NE.1)INODE=NBODY+
    1 (NX(15)+1)*(NY(15)+1)+
    2 (NX(16)+1)*(NY(16)-1)+
    3 NX(17)*(NY(17)+1)+1+
    4 (NX(18)-1)*NY(18)+IX
      IF(IX.EQ.NXP)INODE=NBODY+
    1 (NX(15)+1)*(NY(15)+1)+
    2 (NX(16)+1)*(NY(16)-1)+
    3 NX(17)*(NY(17)+1)+1+
    4 (NX(18)-1)*(NY(18)+1)+1+
    5 NX(18)+(IY-2)*NX(18)
      IF(IX.EQ.NXP.AND.IY.EQ.NYP)INODE=
    1 NBODY+
    2 (NX(15)+1)*(NY(15)+1)+
    3 (NX(16)+1)*(NY(16)-1)+
    4 NX(17)*(NY(17)+1)+1
      IF(IX.EQ.NXP.AND.IY.EQ.1)INODE=
    1 NBODY+
    2 (NX(15)+1)*(NY(15)+1)+
    3 (NX(16)+1)*(NY(16)-1)+NX(17)
    4 +1
      NOFCT(IXY,IS)=INODE
 2180 CONTINUE
 2199 CONTINUE
C
      NLAST=NODE0
      IF(NSHANK.EQ.0)GO TO 2299
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2280 IY=1,NYP
      DO 2280 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IY.EQ.NYP)GO TO 2285
      NODE0=NODE0+1
 2285 CONTINUE
      INODE=NODE0
      IF(IY.EQ.NYP)INODE=NLAST+IX
      NOFCT(IXY,IS)=INODE
 2280 CONTINUE
 2299 CONTINUE
C
      NLAST=NODE0
      IF(NBLADE.EQ.0)GO TO 2399
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2380 IY=1,NYP
      DO 2380 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      NODE0=NODE0+1
      INODE=NODE0
      NOFCT(IXY,IS)=INODE
 2380 CONTINUE
```

```
2399  CONTINUE
C
      NLAST=NODE0
      IF(KSYMMZ.NE.0) GO TO 2439
      IF(NBLADE.EQ.0)GO TO 2499
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2480 IY=1,NYP
      DO 2480 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1.OR.IX.EQ.NXP)GO TO 2485
      IF(IY.EQ.NYP)GO TO 2485
      NODE0=NODE0+1
2485  CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=
     1 NLAST-NXP*NYP+(IY-1)*NX?
     1 +1
      IF(IX.EQ.NXP)INODE=NLAST-NXP*NYP+1+(IY-1)*NXP+NX(IS)
      IF(IY.EQ.NYP)INODE=NLAST-NXP+IX
      NOFCT(IXY,IS)=INODE
2480  CONTINUE
2499  CONTINUE
C
      NLAST=NODE0
      NTEMP=NODE0
      IF(NSHANK.EQ.0)GO TO 2599
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2580 IY=1,NYP
      DO 2580 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IY.EQ.NYP)GO TO 2585
      NODE0=NODE0+1
CC
2585  CONTINUE
      INODE=NODE0
      IF(IY.EQ.NYP)INODE=NLAST+IX
      NOFCT(IXY,IS)=INODE
2580  CONTINUE
2599  CONTINUE
C
      NLAST=NODE0
      IF(NBLADE.EQ.0)GO TO 2699
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2680 IY=1,NYP
      DO 2680 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      NODE0=NODE0+1!
      INODE=NODE0
```

```
      NOFCT(IXY,IS)=INODE
 2680 CONTINUE
 2699 CONTINUE
C
      NLAST=NODE0
      IF(NBLADE.EQ.0)GO TO 2793
      IS=IS+1
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 2780 IY=1,NYP
      DO 2780 IX=1,NXP
      IXY=IX+(IY-1)*NXP
C
      IF(IX.EQ.1.OR.IX.EQ.NXP)GO TO 2785
      IF(IY.EQ.NYP)GO TO 2785
      NODE0=NODE0+1
 2785 CONTINUE
      INODE=NODE0
      IF(IX.EQ.1)INODE=NLAST-NXP*NYP+(IY-1)*NXP
     1 +1
      IF(IX.EQ.NXP) INODE=NLAST-NXP*NYP+1+(IY-1)*NXP+NX(IS)
      IF(IY.EQ.NYP) INODE=NLAST-NXP+IX
      NOFCT(IXY,IS)=INODE
 2780 CONTINUE
 2799 CONTINUE
      NLAST=NODE0
      NODE=NLAST
CC
CC
C
      IF(KSYMMY.EQ.0)NSYMMY=1
      IF(KSYMMY.NE.0)NSYMMY=2
      IF(KSYMMZ.EQ.0)NSYMMZ=1
      IF(KSYMMZ.NE.0)NSYMMZ=2
C
      NNODE=NODE0
      WRITE(6,4000)
 4000 FORMAT(//10X,6HPRERRO)
      WRITE(6,4001) NNODE
 4001 FORMAT(5X,6HNNODE=,I5)
      DO 100 IS=1,NS
      NXP=NX(IS)+1
      NYP=NY(IS)+1
      DO 100 IY=1,NYP
      DO 100 IX=1,NXP
      IXY=IX+(IY-1)*NXP
      WRITE(6,200)IS,IX,IY,IXY,NOFCT(IXY,IS)
  100 CONTINUE
  200 FORMAT(2X,*IS=*,I5,2X,*IX=*,I5,2X,*IY=*,I5,2X,*IXY=*,I5,2X,
     1 *NOFCT=*,I5)
      RETURN
      END
```

```
      SUBROUTINE GEOMET(NELEM,NXYMP,NOFCT,NODE,XK,NTNODE,KWAKE,KROTOR)
C
C
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/FMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KINELE,KINSHP,KINTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      DIMENSION KROTOR(NELEM)
      DIMENSION XK(3,NTNODE),NOFCT(NXYMP,34),KWAKE(NELEM),NODE(4,NELEM)
      REAL LSHAFT,LSHANK
C
C
      NSH=NS
      NHALF=NELEM
      DO 1999 IS=1,NSH
      ISFIX=ISFACE(IS)
      NXX=NX(IS)
      NXP=NXX+1
      NYY=NY(IS)
      DO 999 IX=1,NXX
      DO 999 IY=1,NYY
C
C
      IELEM=IX+NX(IS)*(IY-1)+NT(IS)
      IF(KNORML(IS).EQ.-1)GO TO 906
C
C     --      -+
C     +-      ++
C
```

```
      IXMM=IX
      IXPM=IX+1
      IXMP=IX
      IXPP=IX+1
      IYMM=IY
      IYPM=IY
      IYPP=IY+1
      IYMP=IY+1
C
      GO TO 907
906   CONTINUE
C
C     -+      --
C     ++      +-
C
      IXMM=IX
      IXMP=IX
      IXPP=IX+1
      IXPM=IX+1
      IYMM=IY+1
      IYMP=IY
      IYPP=IY
      IYPM=IY+1
907   CONTINUE
C
      IXYPP=IXPP+(IYPP-1)*NXP
      IXYPM=IXPM+(IYPM-1)*NXP
      IXYMP=IXMP+(IYMP-1)*NXP
      IXYMM=IXMM+(IYMM-1)*NXP
C
      NODE(1,IELEM)=NOFCT(IXY?P,IS)
      NODE(2,IELEM)=NOFCT(IXY?M,IS)
      NODE(3,IELEM)=NOFCT(IXY?P,IS)
      NODE(4,IELEM)=NOFCT(IXY??,IS)
      KWAKE(IELEM)=0
      KROTOR(IELEM)=0
      IF(KWAKES(IS).GT.0.AND..X.EQ.NXX)KWAKE(IELEM)=1
      IF(KROTORS(IS).NE.0) KROTOR(IELEM)=1
      WRITE(6,50)IS,IELEM,NODE(1,IELEM),NODE(2,IELEM),NODE(3,IELEM),
     1 NODE(4,IELEM),KWAKE(IELEM),KROTOR(IELEM)
   50 FORMAT(5X,*IS=*,I5,2X,*IELEM=*,I5,2X,*NODE1=*,I5,2X,*NODE2=*,I5,
     1 2X,*NODE3=*,I5,2X,*NODE4=*,I5,2X,*KWAKE=*,I5,2X,*KROTOR=*,I5)
C
999   CONTINUE
1999  CONTINUE
C
      WRITE(6,60)
   60 FORMAT(10X,*=== END OF GEOMET ===*)
      RETURN
      END
```

```
C
      SUBROUTINE VEC123(NELEM,NODE,PC,P1,P2,P3,XK,NNODE)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NI(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      DIMENSION XK(3,NNODE),NODE(4,NELEM)
      DIMENSION PC(3,NELEM),P1(3,NELEM),P2(3,NELEM),P3(3,NELEM)
      REAL LSHAFT,LSHANK
C
      ALFAR=ALFA*3.14159/180.
      SINALF=SIN(ALFAR)
      COSALF=COS(ALFAR)
      BETA=SQRT(ABS(1.-UMACH*UMACH))
C
      BET=ABETA*3.14159/180.
      SINBET=SIN(BET)
      COSBET=COS(BET)
C
      DO 100 INODE=1,NNODE
      XK1=XK(1,INODE)
      XK3=XK(3,INODE)
      XK(1,INODE)=(XK1*COSALF+XK3*SINALF)/BETA
      XK(3,INODE)=-XK1*SINALF+XK3*COSALF
      WRITE(6,61)INODE,XK1,XK3
   61 FORMAT(10X,*INODE=*,I5,2X,*XK1=*,E16.6,*XK3=*,E16.6)
  100 CONTINUE
      DO 101 INODE=1,NNODE
      XK1=XK(1,INODE)
```

```
      XK2=XK(2,INODE)
      XK(1,INODE)=XK1*COSBET+XK2*SINBET
      XK(2,INODE)=-XK1*SINBET+XK2*COSBET
  101 CONTINUE
      DO 200 IELEM=1,NELEM
      DO 199 K=1,3
      YPP=XK(K,NODE(1,IELEM))
      YPM=XK(K,NODE(2,IELEM))
      YMP=XK(K,NODE(3,IELEM))
      YMM=XK(K,NODE(4,IELEM))
      PC(K,IELEM)=(YPP+YPM+YMP+YMM)/4.
      P1(K,IELEM)=(YPP+YPM-YMP-YMM)/4.
      P2(K,IELEM)=(YPP-YPM+YMP-YMM)/4.
      P3(K,IELEM)=(YPP-YPM-YMP+YMM)/4.
C
199   CONTINUE
200   CONTINUE
C
      WRITE(6,60)
   60 FORMAT(10X,"--- END OF VEC.123 ---")
      RETURN
      END
```

```
C       ########## SUBROUTINE PRINTA ##############################################
C
      SUBROUTINE PRINTA(NELEM,NXYMP,XK,NNODE,PC,NOFCT,NODE,NPRINT)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAR
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      DIMENSION PC(3,NELEM),XK(3,NNODE),NOFCT(NXYMP,34),NODE(4,NELEM)
      REAL LSHAFT,LSHANK
C
      GO TO (1,2,3,4),NPRINT
    1 CONTINUE
      WRITE(NWRITE,110)
  110 FORMAT(///2X,*SPECIFICATIONS OF THE PROBLEM */)
      DO 112 II=1,NS
      IF(NXY(II).EQ.0)GO TO 112
      WRITE(NWRITE,113)ISFACE(II)
      WRITE(NWRITE,114)NX(II),NY(II)
  112 CONTINUE
  113 FORMAT(//2X,*FOR PART *,I3)
      WRITE(NWRITE,115)NELEM,KSYMMY,KSYMMZ,REFLEN,SPAN,TAU,
     1ALFA,ALFABC,UMACH,IZZZ
  114 FORMAT(2X,*NX =*,I2,/2X,*NY=*,I2/)
  115 FORMAT(//2X,*NELEM=*,I3//2X,*KSYMMY=*,I2/2X,*KSYMMZ=*,I2//
     1 2X,*REFERENCE LENGTH =*,F6.2/2X,*SPAN =*,F6.2/
     1 2X,*WING THICKNESS=*,F3.5//2X,*ALFA = *,F7.3/2X,*ALFABC =*,F7.3//
     1 2X,*MACH NUMBER =*,F7.3//2X,*IZZZ=*,I5//)
      WRITE(NWRITE,116)TANGLE,TANGTE,CHORD,R
  116 FORMAT(2X,*TANGLE=*,F6.2/2X,*TANGTE=*,F6.2//2X,*CHORD =*,F6.2
```

```
      1 //2X,*RADIUS =*,F6.2)
        RETURN
2       CONTINUE
        WRITE(NWRITE,222)
    222 FORMAT(//4X,*NODE=*,4X,*X*,10X,*Y*,10X,*Z*//)
        DO 2210 INODE=1,NNODE
2210    WRITE(NWRITE,221)INODE,(XK(K,INODE),K=1,3)
221     FORMAT(4X,I4,3(1X,F10.5))
220     CONTINUE
        RETURN
3       CONTINUE
        WRITE(NWRITE,340)
    340 FORMAT(////1X,*ELEM=*,4X,*XPC*,7X,*YPC*,7X,*ZPC*)
        DO 344 I=1,NELEM
344     WRITE(NWRITE,345)I,(PC(K,I),K=1,3)
345     FORMAT(1X,I3,12F10.5)
        RETURN
4       CONTINUE
        WRITE(NWRITE,41)
     41 FORMAT(//5X,*NODAL NUMBERING FOR SURFACES*//)
        DO 44 IS=1,NS
        NXP=NX(IS)+1
        NYP=NY(IS)+1
        NXYP=NXP*NYP
        WRITE(NWRITE,46)ISFACE(IS)
        DO 44 IX=1,NXP
        WRITE(NWRITE,45)(NOFCT(IXY,IS),IXY=IX,NXYP,NXP)
44      CONTINUE
45      FORMAT(11I6)
     46 FORMAT(/5X,*FOR SURFACE *,I5/)
        WRITE(NWRITE,47)
     47 FORMAT(//5X,*NODAL NUMBERING FOR ELEMENTS*//)
        WRITE(NWRITE,49)
        DO 48 IELEM=1,NELEM
48      WRITE(NWRITE,45)IELEM,(NODE(ICORNR,IELEM),ICORNR=1,4)
     49 FORMAT(//2X,*ELEM*,4X,*++*,4X,*+-*,4X,*-+*,4X,*--*/)
        RETURN
        END
```

```
C
C      ********** SUBROUTINE DEBUG  ****************************************
C
      SUBROUTINE DEBUG(K)
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      WRITE(NWRITE,1)K
    1 FORMAT(/2X,*   ERROR CODE = *,I6/)
      RETURN
      END
```

```
C
C       ********** SUBROUTINE PRINTB ****************************************
C
C
      SUBROUTINE PRINTB(NELEM,NESQ,AA,SOURCE,NPRINT)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,RCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THETLC,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
C
      DIMENSION SOURCE(NELEM),AA(NESQ)
      DIMENSION ABSVAL(100),EMSEAN(100)
      REAL LSHAFT,LSHANK
      NY4=4*(NY(1)-1)
      GO TO(1,2,3,4,5),NPRINT
    1 CONTINUE
      WRITE(NWRITE,110)
  110 FORMAT(///2X,*DISTRIBUTION OF AA(I,J)*/)
      DO 11 I=1,NELEM
      WRITE(NWRITE,111)I
      N1=I
      N2=NELEM*NELEM
  111 FORMAT(2X,*INDEX =*,I2)
      WRITE(NWRITE,112)(AA(K),K=N1,N2,NELEM)
  112 FORMAT(1X,8E15.6)
   11 CONTINUE
      RETURN
    2 CONTINUE
      WRITE(NWRITE,221)
```

```
  221 FORMAT(///2X,*THE DISTRIBUTION OF SOURCE*)
225   CONTINUE
      NSBTOT=0
      DO 229 IS=1,NS
      WRITE(NWRITE,223)ISFACE(IS)
  223 FORMAT(//5X,*FOR SUBSURFACE*I3)
      IND=NSBTOT
      NSBTOT=NSBTOT+NX(IS)*NY(IS)
      IFIN=NSBTOT
      NXX=NX(IS)
      DO 226 IX=1,NXX
      WRITE(NWRITE,228)
      IF(NPRINT.GE.5.AND.(IX.EQ.NXX))GO TO 226
      IND=IND+1
      WRITE(NWRITE,227)(SOURC (KK),KK=IND,IFIN,NXX)
226   CONTINUE
227   FORMAT(1X,8E15.5)
228   FORMAT(/)
229   CONTINUE
      RETURN
3     CONTINUE
      WRITE(NWRITE,330)
  330 FORMAT(///2X,*THE DISTRIBUTION OF THE VELOCITY POTENTIAL*)
      GO TO 225
4     CONTINUE
      WRITE(NWRITE,440)
  440 FORMAT(///2X,*THE DISTRIBUTION OF CP*)
      GO TO 225
5     CONTINUE
      RETURN
      END
```

```
C
C
C
      SUBROUTINE SOLUTN(NELEM,NESQ,AA,SOURCE)
C
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,RCHORD,TAUBL,ALFA2
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      DIMENSION AA(NESQ),SOURCE(NELEM)
      REAL LSHAFT,LSHANK
C
      IF(KPRINT(5).EQ.1)CALL PRINTB(NELEM,NESQ,AA,SOURCE,1)
      IF(KPRINT(6).EQ.1)CALL PRINTB(NELEM,NESQ,AA,SOURCE,2)
      TOL=0.001
      CALL CCGELG(SOURCE,AA,NELEM,1,TOL,IER)
C
      IF(IER.NE.0)WRITE(NWRITE,101)IER
C     IF(IER.NE.0)CALL DEBUG(1001)
  101 FORMAT(////2X,*   IER = *,I4///)
      IF(KPRINT(7).EQ.1)CALL PRINTB(NELEM,NESQ,AA,SOURCE,3)
      RETURN
      END
```

```
C
      SUBROUTINE GGELG(R,A,M,N,EPS,IER)
C
C
C
C
      DIMENSION A(1),R(1)
      IF(M)23,23,1
C
C     SEARCH FOR GREATEST ELEMENT IN MATRIX A
    1 IER=0
      PIV=0.
      MM=M*M
      NM=N*M
      DO 3 L=1,MM
      TBB= ABS(A(L))
      IF(TBB-PIV)3,3,2
    2 PIV=TBB
      I=L
    3 CONTINUE
      TOL=EPS*PIV
C     A(I) IS PIVOT ELEMENT. PIV CONTAINS THE ABSOLUTE VALUE OF A(I).
C
C
C     START ELIMINATION LOOP
      LST=1
      DO 17 K=1,M
C
C     TEST ON SINGULARITY
      IF(PIV)23,23,4
    4 IF(IER)7,5,7
    5 IF(PIV-TOL)6,6,7
    6 IER=K-1
    7 PIVI=1./A(I)
      J=(I-1)/M
      I=I-J*M-K
      J=J+1-K
C     I+K IS ROW-INDEX, J+K COLUMN-INDEX OF PIVOT ELEMENT
C
C     PIVOT ROW REDUCTION AND ROW INTERCHANGE IN RIGHT HAND SIDE R
      DO 8 L=K,NM,M
      LL=L+I
      TB=PIVI*R(LL)
      R(LL)=R(L)
    8 R(L)=TB
C
C     IS ELIMINATION TERMINATED
      IF(K-M)9,18,18
C
C     COLUMN INTERCHANGE IN MATRIX A
    9 LEND=LST+M-K
      IF(J)12,12,10
   10 II=J*M
      DO 11 L=LST,LEND
      TB=A(L)
      LL=L+II
      A(L)=A(LL)
```

```
   11 A(LL)=TB
C
C     ROW INTERCHANGE AND PIVOT ROW REDUCTION IN MATRIX A
   12 DO 13 L=LST,NM,M
      LL=L+I
      TB=PIVI*A(LL)
      A(LL)=A(L)
   13 A(L)=TB
C
C     SAVE COLUMN INTERCHANGE INFORMATION
      A(LST)=J
C
C     ELEMENT REDUCTION AND NEXT PIVOT SEARCH
      PIV=0.
      LST=LST+1
      J=0
      DO 16 II=LST,LEND
      PIVI=-A(II)
      IST=II+M
      J=J+1
      DO 15 L=IST,NM,M
      LL=L-J
      A(L)=A(L)+PIVI*A(LL)
      TBB= ABS(A(L))
      IF(TBB-PIV)15,15,14
   14 PIV=TBB
      I=L
   15 CONTINUE
      DO 16 L=K,NM,M
      LL=L+J
   16 R(LL)=R(LL)+PIVI*R(L)
   17 LST=LST+M
C     END OF ELIMINATION LOOP
C
C
C     BACK SUBSTITUTION AND BACK INTERCHANGE
   18 IF(M-1)23,22,19
   19 IST=MM+M
      LST=M+1
      DO 21 I=2,M
      II=LST-I
      IST=IST-LST
      L=IST-M
      L=A(L)+.5
      DO 21 J=II,NM,M
      TB=R(J)
      LL=J
      DO 20 K=IST,MM,M
      LL=LL+1
   20 TB=TB-A(K)*R(LL)
      K=J+L
      R(J)=R(K)
   21 R(K)=TB
   22 RETURN
C
C
C     ERROR RETURN
   23 IER=-1
      RETURN
      END
```

```
C       ########## SUBROUTINE COEFF ######################################
C
C       #################### DATE CREATED : APRIL 10 ,1975 ############
C
        SUBROUTINE COEFF(NELEM,PC,NNODE,XK,NODE,KWAKE,NESQ,AA,SOURCE,
      1 BC,KROTOR,ICHECK)
C
C        HIS SUBPROGRAM IS TO CALCULATE
C       AND          FOR                             FLOW
C
C
        COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
        COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
        COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
      1 NBLADE
        COMMON/ZZZ4/FMACH,OMEGA,ALFA,ABETA
        COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
        COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
        COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
        COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
        COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
        COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
        COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
        COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
        COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
        COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
        COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
        COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
        COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
        COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
        COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
        COMMON/ZZZ20/PI
        COMMON/ZZZ21/KPY1,KPY2
        COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
        COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
        COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
        COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
        COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
        COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
        COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
        COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
        COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
        COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
        COMMON/ZZZ32/CSTAG,CVORT
        COMMON/ZZZ33/UWAKE
        DIMENSION KROTOR(NELEM)
        DIMENSION QV(3)
        DIMENSION XK(3,NNODE),NODE(4,NELEM),
      1PC(3,NELEM),SOURCE(NELEM),KWAKE(NELEM),AA(NESQ)
        DIMENSION A1(3,2),A2(3,2),A1A1(2),A2A2(2),QCRA1(3),QCRA2(3)
        DIMENSION A1V(3),A2V(3),VNORM(3),UN(3),SN(3),SNUN(3),A1A2(2,2)
        DIMENSION PZ(3),Q(3,4),QMCRA1(3,2),QMCRA2(3,2),AA1(3,2),AA2(3,2)
        DIMENSION VMM(3),VOD(3),VMD(3),VTEPZ(3),VTENZ(3),SIGNPT(3)
        DIMENSION AVA1(3),AVA2(3),A1CRA2(4),QM(3,4)
        DIMENSION WPP(3),WPM(3),WMP(3),WMM(3),WPC(3)
        DIMENSION BC(NELEM          )
        DIMENSION YPP(3),YPM(3),YMP(3),YMM(3)
```

```
      DIMENSION PCJ(3)
      REAL LSHAFT,LSHANK
C
      DOTPRD(X1,Y1,Z1,X2,Y2,Z2)=X1*X2+Y1*Y2+Z1*Z2
C
      PROMIX(XX1,YY1,ZZ1,XX2,YY2,ZZ2,XX3,YY3,ZZ3)=(YY2*ZZ3-YY3*ZZ2)*XX1
     1                     -(XX2*ZZ3-XX3*ZZ2)*YY1+(XX2*YY3-XX3*YY2)*ZZ1
      XPROX(DX,DY,DZ,EX,EY,EZ)=DY*EZ-EY*DZ
      XPROY(DX,DY,DZ,EX,EY,EZ)=DZ*EX-EZ*DX
      XPROZ(DX,DY,DZ,EX,EY,EZ)=DX*EY-EX*DY
C
      IF(UMACH.GT.1.)CALL DEBUG(400)
      BETA=SQRT(1.-UMACH**2)
C
C
C
C
      SIGMPT(1)=1.0
C
      CONSI=10.5/3.14159
C
C
      KOUNT=0
      DO 100 J=1,NELEM
C
      KSPIRAL=1
      LSPIRAL=1
      IF(ICHECK.EQ.4) KSPIRAL=KSPIRAL
      IF(ICHECK.EQ.4) LSPIRAL=NSPIRAL
      DO 100 ISP=1,KSPIRAL
      DO 100 NSP=1,LSPIRAL
C
      GO TO (100,200,300,400),ICHECK
C
  100 CONTINUE
C
      DO 94 K=1,3
      YPP(K)=XK(K,NODE(1,J))
      YPM(K)=XK(K,NODE(2,J))
      YMP(K)=XK(K,NODE(3,J))
      YMM(K)=XK(K,NODE(4,J))
      PCJ(K)=PC(K,J)
94    CONTINUE
      GO TO 500
  200 CONTINUE
C     NSTAG=1
      IF(KOUNT.EQ.1) RETURN
      READ(NREAD,201)  (YPP(K),K=1,3)
      WRITE(NWRITE,202)(YPP(K),K=1,3)
      READ(NREAD,201)  (YPM(K),K=1,3)
      WRITE(NWRITE,202)(YPM(K),K=1,3)
      READ(NREAD,201)  (YMP(K),K=1,3)
      WRITE(NWRITE,202)(YMP(K),K=1,3)
      READ(NREAD,201)  (YMM(K),K=1,3)
      WRITE(NWRITE,202)(YMM(K),K=1,3)
      DO 205 K=1,3
  205 PCJ(K)=(YPP(K)+YPM(K)+YMP(K)+YMM(K))/4.
```

```
      KOUNT=1
      GO TO 500
  300 CONTINUE
C     NVORT=1
      IF (KOUNT.EQ.1) RETURN
      READ(NREAD,201)   (YPP(K),K=1,3)
      WRITE (NWRITE,202) (YPP(K),K=1,3)
      READ(NREAD,201)   (YPM(K),K=1,3)
      WRITE (NWRITE,202) (YPM(K),K=1,3)
      READ(NREAD,201)   (YMP(K),K=1,3)
      WRITE (NWRITE,202) (YMP(K),K=1,3)
      READ(NREAD,201)   (YMM(K ,K=1,3)
      WRITE (NWRITE,202) (YMM(K ,K=1,3)
      DO 305 K=1,3
  305 PCJ(K)=(YPP(K)+YPM(K)+YMP(K)+YMM(K))/4.
      KOUNT=1
  201 FORMAT(    10F8.3)
  202 FORMAT(1X,10F8.3)
      GO TO 500
  400 CONTINUE
C     NSPIRAL=1
      IF(KWAKE(J).EQ.0) GO TO 100
      IF(KROTOR(J).EQ.0)GO TO 100
      DO 494 K=1,3
      YPP(K)=XK(K,NODE(1,J))
      YPM(K)=XK(K,NODE(2,J))
      YMP(K)=XK(K,NODE(3,J))
      YMM(K)=XK(K,NODE(4,J))
  494 CONTINUE
C
      RATIO=UWAKE/OMEGA
      RATIO=RATIO*50./(2.*PI)
      RATIO=-1.*RATIO
      FACTOR=2.*PI/NSPIRAL
C     DO 100 ISP=1,ISPIRAL
C     DO 100 NSP=1,NSPIRAL
      IWAKE=NSP+(ISP-1)*NSPIRAL
C
      THETA1=(IWAKE-1)*FACTOR
      THETA2= IWAKE    *FACTOR
C
      SIN1=SIN(THETA1)
      COS1=COS(THETA1)
C
      SIN2=SIN(THETA2)
      COS2=COS(THETA2)
C
      XPP2= YPP(1)*COS2+YPP(2)*SIN2
      YPP2=-YPP(1)*SIN2+YPP(2)*COS2
C
      XMP1= YPP(1)*COS1+YPP(2)*SIN1
      YMP1=-YPP(1)*SIN1+YPP(2)*COS1
C
      XMM1= YPM(1)*COS1+YPM(2)*SIN1
      YMM1=-YPM(1)*SIN1+YPM(2)*COS1
C
      XPM2= YPM(1)*COS2+YPM(2)*SIN2
```

```
      YPM2=-YPM(1)*SIN2+YPM(2)*COS2
C
      RPP2=SQRT(XPP2*XPP2+YPP2*YPP2)
      RMP1=SQRT(XMP1*XMP1+YMP1*YMP1)
      RMM1=SQRT(XMM1*XMM1+YMM1*YMM1)
      RPM2=SQRT(XPM2*XPM2+YPM2*YPM2)
C
      IF(NSP.GT.1) GO TO 503
C
      ZP=0.0
      ZPP2=ZP+RATIO*THETA2*RPP2/RROTOR
      ZMP1=ZP+RATIO*THETA1*RMP1/RROTOR
       ZMM1=ZP+RATIO*THETA1*RMM1/RROTOR
      ZPM2=ZP+RATIO*THETA2*RPM2/RROTOR
      GO TO 504
  503 CONTINUE
      ZPP2=ZP+RATIO*(THETA2-FACTOR)+RATIO*FACTOR*RPP2/RROTOR
      ZMP1=ZP+RATIO*(THETA1-FACTOR)+RATIO*FACTOR*RMP1/RROTOR
      ZMM1=ZP+RATIO*(THETA1-FACTOR)+RATIO*FACTOR*RMM1/RROTOR
      ZPM2=ZP+RATIO*(THETA2-FACTOR)+RATIO*FACTOR*RPM2/RROTOR
  504 CONTINUE
C
      YPP(1)=XPP2
      YPP(2)=YPP2
      YPP(3)=ZPP2
C
      YPM(1)=XPM2
      YPM(2)=YPM2
      YPM(3)=ZPM2
C
      YMP(1)=XMP1
      YMP(2)=YMP1
      YMP(3)=ZMP1
C
      YMM(1)=XMM1
      YMM(2)=YMM1
      YMM(3)=ZMM1
C
      IF(ZPP2.LT.(ZBD1+RZBD1+ZPYL))GO TO 180
      IF(ZPM2.LT.(ZBD1+RZBD1+ZPYL)) GO TO 180
C
      DO 505 K=1,3
  505 PCJ(K)=(YPP(K)+YPM(K)+YMP(K)+YMM(K))/4.
C
C
  500 CONTINUE
C
      DO 1002 K=1,3
      AA1(K,1)=0.5*(YPP(K)-YMP(K))
      AA1(K,2)=0.5*(YPM(K)-YMM(K))
      AA2(K,1)=0.5*(YPP(K)-YPM(K))
      AA2(K,2)=0.5*(YMP(K)-YMM(K))
      A1(K,1)=AA1(K,1)
      A1(K,2)=AA1(K,2)
      A2(K,1)=AA2(K,1)
      A2(K,2)=AA2(K,2)
 1002 CONTINUE
```

```
      DO 1003 L=1,2
      A1A1(L)=DOTPRO(A1(1,L),A1(2,L),A1(3,L),A1(1,L),A1(2,L),A1(3,L))
 1003 A2A2(L)=DOTPRO(A2(1,L),A2(2,L),A2(3,L),A2(1,L),A2(2,L),A2(3,L))
C
      DO 1004 K=1,3
      IF(A1A1(1).EQ.0.)A1(K,1)=A1(K,2)
      IF(A1A1(2).EQ.0.)A1(K,2)=A1(K,1)
      IF(A2A2(1).EQ.0.)A2(K,1)=A2(K,2)
      IF(A2A2(2).EQ.0.)A2(K,2)=A2(K,1)
 1004 CONTINUE
C
      DO 1006 L=1,2
      A1A1(L)=DOTPRO(A1(1,L),A1(2,L),A1(3,L),A1(1,L),A1(2,L),A1(3,L))
 1006 A2A2(L)=DOTPRO(A2(1,L),A2(2,L),A2(3,L),A2(1,L),A2(2,L),A2(3,L))
C
      DO 1007 L=1,2
      DO 1007 M=1,2
 1007 A1A2(L,M)=DOTPRO(A1(1,L),A1(2,L),A1(3,L),A2(1,M),A2(2,M),A2(3,M))
C
      A1CRA2(1)=SQRT(A1A1(2)*A2A2(1)-A1A2(2,1)**2)
      A1CRA2(2)=SQRT(A1A1(1)*A2A2(1)-A1A2(1,1)**2)
      A1CRA2(3)=SQRT(A1A1(1)*A2A2(2)-A1A2(1,2)**2)
      A1CRA2(4)=SQRT(A1A1(2)*A2A2(2)-A1A2(2,2)**2)
C
      DO 1008 K=1,3
      AVA1(K)=0.5*(A1(K,1)+A1(K,2))
 1008 AVA2(K)=0.5*(A2(K,1)+A2(K,2))
C
      YNORM(1)=AVA1(2)*AVA2(3 -AVA1(3)*AVA2(2)
      YNORM(2)=AVA1(3)*AVA2(1)-AVA1(1)*AVA2(3)
      YNORM(3)=AVA1(1)*AVA2(2)-AVA1(2)*AVA2(1)
C
      SN(1)=YNORM(1)/BETA
      SN(2)=YNORM(2)
      SN(3)=YNORM(3)
      ASN=SQRT(SN(1)**2+SN(2)**2+SN(3)**2)
      AYN=SQRT(YNORM(1)**2+YNORM(2)**2+YNORM(3)**2)
      DO 1010 K=1,3
      UN(K)=YNORM(K)/AYN
      SNUN(K)=SN(K)/ASN
 1010 CONTINUE
C
C
      DO 170 I=1,NELEM
C
      DO 160 ISYMMY=1,NSYMMY
      DO 160 ISYMMZ=1,NSYMMZ
      SIGNPT(2)=3.-2*ISYMMY
      SIGNPT(3)=3.-2*ISYMMZ
C
      DO 1102 K=1,3
 1102 PZ(K)=PCJ(K)-PC(K,I)*SIGNPT(K)
      QDOTUN=DOTPRO(UN(1),UN(2),UN(3),PZ(1),PZ(2),PZ(3))
C
      DO 1110 K=1,3
      Q(K,1)=YPM(K)-PC(K,I)*SIGNPT(K)
      Q(K,2)=YPP(K)-PC(K,I)*SIGNPT(K)
```

```
      Q(K,3)=YMP(K)-PC(K,I)*SIGNPT(K)
      Q(K,4)=YMM(K)-PC(K,I)*SIGNPT(K)
1110  CONTINUE
      DO 1111 K=1,3
      QM(K,1)=0.5*(Q(K,1)+Q(K,2))
      QM(K,2)=0.5*(Q(K,2)+Q(K,3))
      QM(K,3)=0.5*(Q(K,3)+Q(K,4))
      QM(K,4)=0.5*(Q(K,4)+Q(K,1))
1111  CONTINUE
C
      DO 1112 K=1,3
      KP1=K+1
      KP2=K+2
      IF(KP1.GT.3)KP1=KP1-3
      IF(KP2.GT.3)KP2=KP2-3
      QMCRA1(K,1)=QM(KP1,2)*A1(KP2,1)-QM(KP2,2)*A1(KP1,1)
      QMCRA1(K,2)=QM(KP1,4)*A1(KP2,2)-QM(KP2,4)*A1(KP1,2)
      QMCRA2(K,1)=QM(KP1,1)*A2(KP2,1)-QM(KP2,1)*A2(KP1,1)
      QMCRA2(K,2)=QM(KP1,3)*A2(KP2,2)-QM(KP2,3)*A2(KP1,2)
1112  CONTINUE
C
      RC=SQRT(DOTPRO(PZ(1),PZ(2),PZ(3),PZ(1),PZ(2),PZ(3)))
C
      DO 155 ICORNR=1,4
      GO TO (5502,5504,5506,5508),ICORNR
5502  CONTINUE
      SIGN12=-1.
      ICSI=1
      IETA=2
      GO TO 5510
5504  CONTINUE
      SIGN12=+1.
      ICSI=1
      IETA=1
      GO TO 5510
5506  CONTINUE
      SIGN12=-1.
      ICSI=2
      IETA=1
      GO TO 5510
5508  CONTINUE
      SIGN12=+1.
      ICSI=2
      IETA=2
5510  CONTINUE
C
      DO 5520 K=1,3
      QV(K)=Q(K,ICORNR)
      A1V(K)=A1(K,IETA)
      A2V(K)=A2(K,ICSI)
      QCRA1(K)=QMCRA1(K,IETA)
      QCRA2(K)=QMCRA2(K,ICSI)
5520  CONTINUE
C
      QQ=SQRT(DOTPRO(QV(1),QV(2),QV(3),QV(1),QV(2),QV(3)))
C
      CALL LOG(ICORNR,QQ,QV,A1V,QCRA1,ALOG1,1)
```

```
C
      CALL LOG(ICORNR,QQ,QV,A2V,QCRA2,ALOG2,2)
C
      TANP=0.
      IF(QDOTUN.EQ.0.)GO TO 5550
      HNUMER=-DOTPRO(QCRA1(1),QCRA1(2),QCRA1(3),QCRA2(1),QCRA2(2),
     1                                                   QCRA2(3))
      DENOM=QQ*QDOTUN*A1CRA2(ICORNR)
      IF(DENOM.NE.0.)TANP=ATAN2(HNUMER,DENOM)
 5550 CONTINUE
C
      COEFF1=DOTPRO(UN(1),UN(2),UN(3),QCRA1(1),QCRA1(2),QCRA1(3))
      COEFF2=DOTPRO(UN(1),UN(2),UN(3),QCRA2(1),QCRA2(2),QCRA2(3))
C
      SRCINT=-CONST*SIGN12*
     1      (-COEFF1*ALOG1+COEFF2*ALOG2-QDOTUN*TANP)
C
      GO TO (801,802,803,804),ICHECK
  801 CONC=1.0
      GO TO 850
  802 CONC=CSTAG
      GO TO 850
  803 CONC=CVORT
      GO TO 850
  804 CONC=1.0
  850 CONTINUE
      DBTINT=-CONST*SIGN12*TANP*CONC
C
      SGNINT=1.
C
      NNN=I+(J-1)*NELEM
C
      AA(NNN)=AA(NNN)-SGNINT*DBTINT
C
C
      IF(ICHECK.NE.1)GO TO 600
      UFREE=UMACH*1117.
      OMEGAX=0.
      OMEGAY=0.
      OMEGAZ=OMEGA*2.*PI/60.
      RX=PC(1,J)
      RY=PC(2,J)
      RZ=PC(3,J)
      BCX=XPROX(OMEGAX,OMEGAY,OMEGAZ,RX,RY,RZ)
      BCY=XPROY(OMEGAX,OMEGAY,OMEGAZ,RX,RY,RZ)
      BCZ=XPROZ(OMEGAX,OMEGAY,OMEGAZ,RX,RY,RZ)
      BCR=DOTPRO(BCX,BCY,BCZ,UN(1),UN(2),UN(3))
      IF(KROTOR(J).EQ.0) BCR=0.
      BCB=-1.*UFREE*UN(1)
      BC(J)=BCB+BCR
      SOURCE(I)=SOURCE(I)+SGNINT*SRCINT*          BC(J   )
  600 CONTINUE
C
C
150   CONTINUE
155   CONTINUE
160   CONTINUE
```

```
170   CONTINUE
180   CONTINUE
C
C
      RETURN
      END
```

```
C
C
      SUBROUTINE LOG(ICORNR,QQ,QV,AV,QCRAV,ALOG,ICOOR)
      DIMENSION QV(3)
      DIMENSION AV(3),QCRAV(3)
C
      DOTPRO(X1,Y1,Z1,X2,Y2,Z2)=X1*X2+Y1*Y2+Z1*Z2
C
      AVAV=DOTPRO(AV(1),AV(2),AV(3),AV(1),AV(2),AV(3))
      QQAV=DOTPRO(QV(1),QV(2),QV(3),AV(1),AV(2),AV(3))
      QXA=SQRT(DOTPRO(QCRAV(1),QCRAV(2),QCRAV(3),
     1                QCRAV(1),QCRAV(2),QCRAV(3)))
      ALOG=ASINH(QQAV/QXA)/SQRT(AVAV)
      RETURN
      END
```

```
C
C
      FUNCTION ATANP(HNUMER,DENOM)
      ADENOM=ABS(DENOM)
      ATANP=ATAN2(HNUMER,ADENOM)
      IF(DENOM.LT.0.)ATANP=-ATANP
      RETURN
      END
```

```
C
C
      SUBROUTINE AVERAG(SOURCE,SOR,PC,NELEM,NODE,NOFCT,NXYMF,NNODE,
     1 AVG)
      DIMENSION SOURCE(NELEM),SOR(NNODE),AVG(NNODE),PC(3,NELEM)
      DIMENSION NODE(4,NELEM),NOFCT(NXYMP,34)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSEX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,X3D2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,Y3D2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,Z3D2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHA FT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFA9
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      INTEGER FM,PP
      REAL LSHAFT,LSHANK
C
      DO 90 INODE=1,NNODE
      AVG(INODE)=0.
   90 SOR(INODE)= 0.
      DO 100 IELEM=1,NELEM
      PP=NODE(1,IELEM)
      PM=NODE(2,IELEM)
      MP=NODE(3,IELEM)
      MM=NODE(4,IELEM)
      SOR(PP)=SOR(PP)+SOURCE(IELEM)
      SOR(PM)=SOR(PM)+SOURCE(IELEM)
      SOR(MP)=SOR(MP)+SOURCE(IELEM)
      SOR(MM)=SOR(MM)+SOURCE(IELEM)
      AVG(PP)=AVG(2P)+1.
      AVG(PM)=AVG(PM)+1.
      AVG(MP)=AVG(MP)+1.
      AVG(MM)=AVG(MM)+1.
```

```
100   CONTINUE
      DO 110 INODE=1,NNODE
110   SOR(INODE)=SOR(INODE)/AVG(INODE)
  199 CONTINUE
      SOR(1)=0.0
      SOR(2)=0.0
      RETURN
      END
```

```
C
      SUBROUTINE PHI(SOR,NNODE,NODE,PHIC,PHI1,PHI2,PHI3,NELEM)
      DIMENSION SOR(NNODE),PHIC(NELEM),PHI1(NELEM),PHI2(NELEM)
      DIMENSION PHI3(NELEM)
      DIMENSION NODE(4,NELEM)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KINELE,KINSHP,KINTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      INTEGER PM,PP
      REAL LSHAFT,LSHANK
      DO 701 IELEM=1,NELEM
      PP=NODE(1,IELEM)
      PM=NODE(2,IELEM)
      MP=NODE(3,IELEM)
      MM=NODE(4,IELEM)
      PHIC(IELEM)=(SOR(PP)+SOR(PM)+SOR(MP)+SOR(MM))/4.
      PHI1(IELEM)=(SOR(PP)+SOR(PM)-SOR(MP)-SOR(MM))/4.
      PHI2(IELEM)=(SOR(PP)-SOR(PM)+SOR(MP)-SOR(MM))/4.
      PHI3(IELEM)=(SOR(PP)-SOR(PM)-SOR(MP)+SOR(MM))/4.
701   CONTINUE
      WRITE(6,719)
  719 FORMAT(//6X,*ELEM*,9X,*PHIC*,11X,
     1*PHI1*9X,*PHI2*,9X,*PHI3*)
      DO 720 IELEM=1,NELEM
  720 WRITE(6,721)IELEM,PHIC(IELEM),PHI1(IELEM),PHI2(IELEM),PHI3(IELEM)
721   FORMAT(5X,I6,5X,6E15.5)
      RETURN
      END
```

```
C
      SUBROUTINE VELXYZ(VELX,VELY,VELZ,PHI1,PHI2,PHI3,PC,P1,P2,P3,
     1 NELEM)
      DIMENSION VELX(NELEM),PHI1(NELEM),PHI2(NELEM)
      DIMENSION PHI3(NELEM)
      DIMENSION VELY(NELEM),VELZ(NELEM)
      DIMENSION PC(3,NELEM),P1(3,NELEM),P2(3,NELEM),P3(3,NELEM)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSFX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THET75,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
C
      DOTPRO(X1,X2,X3,Y1,Y2,Y3)=X1*Y1+X2*Y2+X3*Y3
      VECPRX(D1,D2,D3,E1,E2,E3)=D2*E3-D3*E2
      VECPRY(D1,D2,D3,E1,E2,E3)=D3*E1-D1*E3
      VECPRZ(D1,D2,D3,E1,E2,E3)=D1*E2-D2*E1
C
      DO 888 JELEM=1,NELEM
C
      A1XC=P1(1,JELEM)
      A1YC=P1(2,JELEM)
      A1ZC=P1(3,JELEM)
      A2XC=P2(1,JELEM)
      A2YC=P2(2,JELEM)
      A2ZC=P2(3,JELEM)
C
      A1XA2X=VECPRX(A1XC,A1YC,A1ZC,A2XC,A2YC,A2ZC)
      A1XA2Y=VECPRY(A1XC,A1YC,A1ZC,A2XC,A2YC,A2ZC)
      A1XA2Z=VECPRZ(A1XC,A1YC,A1ZC,A2XC,A2YC,A2ZC)
```

```
C
      A1XA2S=DOTPRO(A1XA2X,A1XA2Y,A1XA2Z,A1XA2X,A1XA2Y,A1XA2Z)
      A1XA2A=SQRT(A1XA2S)
C
      A1A1=DOTPRO(A1XC,A1YC,A1ZC,A1XC,A1YC,A1ZC)
      A1A2=DOTPRO(A1XC,A1YC,A1ZC,A2XC,A2YC,A2ZC)
      A2A2=DOTPRO(A2XC,A2YC,A2ZC,A2XC,A2YC,A2ZC)
C
      A11=A2A2/A1XA2S
      A12=-A1A2/A1XA2S
      A21=A12
      A22=A1A1/A1XA2S
C
      UNORMX=A1XA2X/A1XA2A
      UNORMY=A1XA2Y/A1XA2A
      UNORMZ=A1XA2Z/A1XA2A
C
      A1XU=A11*A1XC+A12*A2XC
      A1YU=A11*A1YC+A12*A2YC
      A1ZU=A11*A1ZC+A12*A2ZC
C
      A2XU=A21*A1XC+A22*A2XC
      A2YU=A21*A1YC+A22*A2YC
      A2ZU=A21*A1ZC+A22*A2ZC
C
      A3XU=UNORMX
      A3YU=UNORMY
      A3ZU=UNORMZ
C
      VELX(JELEM)=PHI1(JELEM)*A1XU+PHI2(JELEM)*A2XU
      VELY(JELEM)=PHI1(JELEM)*A1YU+PHI2(JELEM)*A2YU
      VELZ(JELEM)=PHI1(JELEM)*A1ZU+PHI2(JELEM)*A2ZU+PHI3(JELEM)*A3ZU
800   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CPLINR(CP,VELX,VELY,VELZ,PHIC,NELEM,PC,KROTOR)
      DIMENSION CP(NELEM),VELX(NELEM),PHIC(NELEM)
      DIMENSION VELY(NELEM),VELZ(NELEM)
      DIMENSION PC(3,NELEM),KROTOR(NELEM)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSEX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/KNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THETZS,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      DOTPRO(X1,Y1,Z1,X2,Y2,Z2)=X1*X2+Y1*Y2+Z1*Z2
      XPROX(DX,DY,DZ,EX,EY,EZ)=DY*EZ-EY*DZ
      XPROY(DX,DY,DZ,EX,EY,EZ)=DZ*EX-EZ*DX
      XPROZ(DX,DY,DZ,EX,EY,EZ)=DX*EY-EX*DY
      BETA2A=ABS(1.-UMACH**2)
      BETA=SQRT(BETA2A)
      UFREE=UMACH*1117
      OMEGAX=0.
      OMEGAY=0.
      OMEGAZ=OMEGA
      OMEGAZ=OMEGAZ*2.*PI/60.
      IF(UMACH.EQ.0) UFREE=OMEGA*2.*PI*RROTOR/60.
      DO 900 IELEM=1,NELEM
      RX=PC(1,IELEM)
      RY=PC(2,IELEM)
      RZ=PC(3,IELEM)
      VX=VELX(IELEM)
      VY=VELY(IELEM)
      VZ=VELZ(IELEM)
      CPB=(-2.*UFREE*VX-(VX*VX+VY*VY+VZ*VZ))/(UFREE*UFREE)
      CPB=(-2.*UFREE*VX   )/(UFREE*UFREE)
```

```
      CRX=XPROX(OMEGAX,OMEGAY.OMEGAZ,RX,RY,RZ)
      CRY=XPROY(OMEGAX,OMEGAY,OMEGAZ,RX,RY,RZ)
      CRZ=XPROZ(OMEGAX,OMEGAY,OMEGAZ,RX,RY,RZ)
      CPR=DOTPRO(CRX,CRY,CRZ,VX,VY,VZ)
      CPR=CPR*2./(UFREE*UFREE)
      IF(KROTOR(IELEM).EQ.0)CPR=0.
      CP(IELEM)=CPB+CPR
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FORCE(NNODE,XK,NELEM,NODE,CP)
      DIMENSION A1(3),A2(3),XK(3,NNODE),NODE(4,NELEM),PC(3),CP(NELEM)
      COMMON/ZZZ1/NX(34),NY(34),NXY(34),KSYMMY,KSYMMZ,NSYMMY,NSYMMZ
      COMMON/ZZZ2/NSEX,NSBODY,NS,NT(34),ISFACE(34),KNORML(34),KWAKES(34)
      COMMON/ZZZ3/NPYLON,NBODY1,NBODY2,NBODY3,NVTAIL,NSHAFT,NHUB,NSHANK,
     1 NBLADE
      COMMON/ZZZ4/UMACH,OMEGA,ALFA,ABETA
      COMMON/ZZZ5/XPYCTR,YPYCTR,ZPYCTR,RXPYL,RYPYL,RZPYL
      COMMON/ZZZ6/XNOSE,XBD1,XBD2,XTAIL
      COMMON/ZZZ7/YNOSE,YBD1,YBD2,YTAIL
      COMMON/ZZZ8/ZNOSE,ZBD1,ZBD2,ZTAIL
      COMMON/ZZZ9/RYBD1,RZBD1,RYBD2,RZBD2
      COMMON/ZZZ10/RSHAFT,LSHAFT,RSHANK,LSHANK
      COMMON/ZZZ11/XHUBCR,YHUBCR,ZHUBCR,RXHUB,RYHUB,RZHUB
      COMMON/ZZZ12/RROTOR,BCHORD,TAUBL,ALFAB
      COMMON/ZZZ13/THETZ5,THET1C,THET1S,CONING,AZIMUTH
      COMMON/ZZZ14/KBLADE,TANLEB,TANTEB,XBLE,XBTE,KROTORS(34)
      COMMON/ZZZ15/VSPAN,XLEZV,XTEZV,TANLEV,TANTEV,TAU,ZVTAIL
      COMMON/ZZZ16/NWAKPY,NWAKHB,NWAKSK,NWAKBL
      COMMON/ZZZ17/WAKLPY,WAKLHB,WAKLSK,WAKLBL
      COMMON/ZZZ18/WANGPY,WANGHB,WANGSK,WANGBL
      COMMON/ZZZ19/KPRINT(10),NREAD,NWRITE,KREAD
      COMMON/ZZZ20/PI
      COMMON/ZZZ21/KPY1,KPY2
      COMMON/ZZZ22/KNSELE,KNSSHP,KNSTYP
      COMMON/ZZZ23/KBDELE,KBDSHP,KBDTYP
      COMMON/ZZZ24/KTNELE,KTNSHP,KTNTYP
      COMMON/ZZZ25/KPYELE,KPYSHP,KPYTYP
      COMMON/ZZZ26/KVTELE,KVTSHP,KVTTYP
      COMMON/ZZZ27/KSHELE,KSHSHP,KSHTYP
      COMMON/ZZZ28/KHBELE,KHBSHP,KHBTYP
      COMMON/ZZZ29/KSKELE,KSKSHP,KSKTYP
      COMMON/ZZZ30/KBLELE,KBLSHP,KBLTYP
      COMMON/ZZZ31/NSTAG,NVORT,NSPIRAL,SPIRAL
      INTEGER PP,PM,MP,MM
C
      DOTPRO(X1,Y1,Z1,X2,Y2,Z2)=X1*X2+Y1*Y2+Z1*Z2
      XPROX(DX,DY,DZ,EX,EY,EZ)=DY*EZ-DZ*EY
      XPROY(DX,DY,DZ,EX,EY,EZ)=DZ*EX-DX*EZ
      XPROZ(DX,DY,DZ,EX,EY,EZ)=DX*EY-DY*EX
C
      BETA=SQRT(1.-UMACH*UMACH)
      MULT=1
      MULTZ=1
      IF(KSYMMZ.NE.0) MULTZ=2
      IF(KSYMMY.NE.0) MULT=2
      CLIFT=0.0
      CDRAG=0.0
      CM=0.0
      DO 100 J=1,NELEM
      PP=NODE(1,J)
      PM=NODE(2,J)
      MP=NODE(3,J)
      MM=NODE(4,J)
      DO 200 K=1,3
      YPP=XK(K,PP)
      YPM=XK(K,PM)
```

```
      YMP=XK(K,MP)
      YMM=XK(K,MM)
      A1(K)=(YPP+YPM-YMP-YMM)/4.
      A2(K)=(YPP-YPM+YMP-YMM)/4.
      PC(K)=(YPP+YPM+YMP+YMM)/4.
  200 CONTINUE
      A1(1)=A1(1)*BETA
      A2(1)=A2(1)*BETA
      PC(1)=PC(1)*BETA
C
      A1XA2X=XPROX(A1(1),A1(2),A1(3),A2(1),A2(2),A2(3))
      A1XA2Y=XPROY(A1(1),A1(2),A1(3),A2(1),A2(2),A2(3))
      A1XA2Z=XPROZ(A1(1),A1(2),A1(3),A2(1),A2(2),A2(3))
C
      DELL=A1XA2Z
      CLIFT=CLIFT-4.*CP(J)*DELL*MULT*MULTZ
C
      DELD=A1XA2X
      CDRAG=CDRAG-4.*CP(J)*DELD*MULT*MULTZ
C
  100 CONTINUE
C
      AREA=1.
C
      CLIFT=CLIFT/AREA
      CDRAG=CDRAG/AREA
C
      WRITE(NWRITE,300)CLIFT
  300 FORMAT(//10X,*LIFT COEFFICIENT =*,E12.6)
      WRITE(NWRITE,301)CDRAG
  301 FORMAT(//10X,*DRAG (INDUCED) COEFFICIENT =*,E12.6)
      RETURN
      END
```