NASA Technical Memorandum 78681

COMPUTER PROGRAM FOR DETERMINING MASS PROPERTIES OF A RIGID STRUCTURE

(NASA-TM-78681) COMPUTER PROGRAM FOR DETERMINING MASS PROPERTIES OF A RIGID STRUCTURE (NASA) 47 p HC A03/MF A01 CSCL 20K

N78-20535

Unclas

G3/39 11902

REID A. HULL, JOHN L. GILBERT AND
PHILLIP J. KLICH

MARCH 1978

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

COMPUTER PROGRAM FOR DETERMINING MASS PROPERTIES OF A RIGID STRUCTURE

BY Reid A. Hull, Dr. John L. Gilbert and Phillip J. Klich

SUMMARY

A computer program (ND0702) has been developed for the rapid computation of the mass properties of complex structural systems. The program uses rigid body analyses and permits differences in structural material throughout the total system. It is based on the premise that complex systems can be adequately described by a combination of basic elemental shapes. The following thirteen widely used structural shapes were selected for inclusion in the program:

1. Discrete mass
2. Cylinder
3. Truncated cone
4. Torus
5. Beam (arbitrary cross section)
6. Circular rod (arbitrary cross section)
7. Spherical segment
8. Sphere
9. Hemisphere
10. Parallelepiped
11. Swept Trapezoidal Panel
12. Symmetric Trapezoidal Panels
13. Curved Rectangular Panel

Simple geometric data describing size and location of each element and the respective material density or weight of each element are the only required input data. From this minimum input, the program yields system weight, center of gravity, moments of inertia and products of inertia with respect to mutually perpendicular axes through the system center of gravity. The program also yields mass properties of the individual shapes relative to component axes.

Permanent configuration records and the use of iterative calculations to investigate design systems or to determine optimums contribute to the cost-effectiveness of the programs use.

INTRODUCTION

Determining the mass properties of any rigid structure is a problem that at times becomes complex, but one which can easily be dealt with utilizing computer solutions.

For rigid structures the solution of the mass properties requires transformation to an axis parallel to the system axis and becomes laborious almost to the point of being impractical.

Any complex structure must be broken down into elements in order to exact a solution. The approach selected for the program presented in this paper was to automate the input to the point where an element's shape, geometry, density or weight, and three grid points are the only requirements. This approach was influenced by the simplicity of computing the direction cosines (Euler angle relationship) from the given three grid points. The program as outlined in this paper performs essentially the same process as calculations "by hand" and is extremely useful for rigid structures skewed in space. This program also provides improved accuracy, time savings, and complete permanent records for a mass properties analysis. (This TMX is a verified expansion of the LaRC working paper "Computer Program for Determining Mass Properties of a Composite Body", by Phillip J. Klich and John L. Gilbert dated Oct. 22, 1968.)

SYMBOLS

I	moments and products of inertia
$\vec{r}$	displacement vector for differential mass
T	kinetic energy
$\vec{V}$	linear velocity
x', y', z'	rectangular coordinate component displacement vectors
x, y, z	rectangular coordinate system displacement vectors
$\vec{\omega}$	angular velocity
$l_{x'x}$	cosine of angle between x' and x axes
$l_{y'x}$	cosine of angle between y' and x axes
$l_{z'x}$	cosine of angle between z' and x axes
$l_{x'y}$	cosine of angle between x' and y axes

$l_{y'y}$	cosine of angle between y' and y axes
$l_{z'y}$	cosine of angle between z' and y axes
$l_{x'z}$	cosine of angle between x' and z axes
$l_{y'z}$	cosine of angle between y' and z axes
$l_{z'z}$	cosine of angle between z' and z axes

Subscripts

co	system coordinates to component center of mass
xxco, yyco, zzco	refer to component axis to which moments of inertia are calculated
XXCO, YYCO, ZZCO	refer to system axis to which the moments of inertia are rotated parallel to the system coordinates

Superscripts

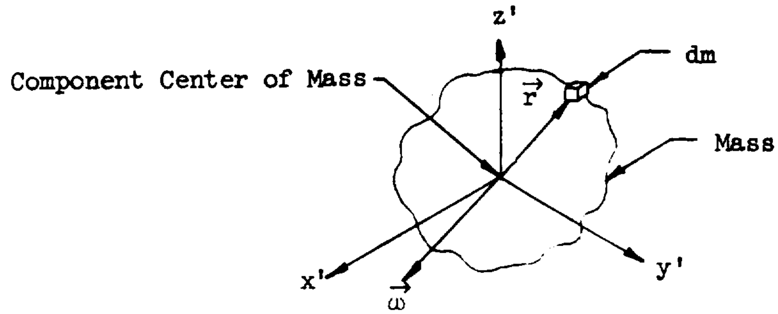
prime	denotes component coordinate system

INERTIA EQUATIONS

The mass properties of shapes such as a cylinder, sphere, etc., are easily calculated and therefore were selected as the basic component shapes for handling a system such as a spacecraft structure. Since the component shape mass properties are measured with respect to their respective center of mass, these properties have to be transferred to the system center of mass. The transformation can be made in two steps: first, the component properties are transferred to a system parallel to the system axis, and then transferred by the usual parallel axis theorem. The rotational transformation is derived by using the principle of kinetic energy. An introductory derivation of the moment of inertia, and product of inertia expressions are derived first and then transformation from the component to system coordinates is presented.

Derivation of Inertia Equations

Using the expressions of kinetic energy of a rigid body, the equations of moments of inertia and products of inertia are derived. Consider a component spinning with an angular velocity $\vec{\omega}$ as shown next



The angular kinetic energy can be written as

$$T = \frac{1}{2}\int \vec{V} \cdot \vec{V}\, dm$$

where the velocity is expressed as

$$\vec{V} = \vec{\omega} \times \vec{r}$$

Substituting in the kinetic energy expression gives

$$T = \frac{1}{2}\int (\vec{\omega} \times \vec{r}) \cdot (\vec{\omega} \times \vec{r}) dm$$

where

$$\vec{\omega} = \vec{i}\omega_x' + \vec{j}\omega_y' + \vec{k}\omega_z'$$

and

$$\vec{r} = \vec{i}x' + \vec{j}y' + \vec{k}z'$$

Taking the cross product

$$\begin{vmatrix} i & j & k \\ \omega_x' & \omega_y' & \omega_z' \\ x' & y' & z' \end{vmatrix} = \vec{i}(\omega_y' z' - \omega_z' y') + \vec{j}(\omega_z' x' - \omega_x' z') + \vec{k}(\omega_x' y' - \omega_y' x')$$

and performing the dot product results in

$$(\omega_y' z')^2 - 2(\omega_y' z')(\omega_z' y') + (\omega_z' y')^2 + (\omega_z' x')^2 - 2(\omega_z' x')(\omega_x' z')$$

$$+ (\omega_x' z')^2 + (\omega_x' y')^2 - 2(\omega_x' y')(\omega_y' x') + (\omega_y' x')^2$$

Which upon substituting into kinetic energy equation gives

$$T = \frac{1}{2} \int \Big[\omega_x'^2 (y'^2 + z'^2) + \omega_y'^2 (x'^2 + z'^2) + \omega_z'^2 (x'^2 + y'^2) - 2\omega_z'\omega_x' x'z' - 2\omega_y'\omega_z' y'z' - 2\omega_x'\omega_y' x'y' \Big] \, dm$$

This represents the rotational kinetic energy of one component of a system. Recognizing the definitions of moments and products of inertias and selecting the component coordinate system as the principal axes, we can write

$$T_{comp} = \frac{1}{2} (I_x' \omega_x'^2 + I_y' \omega_y'^2 + I_z' \omega_z'^2)$$

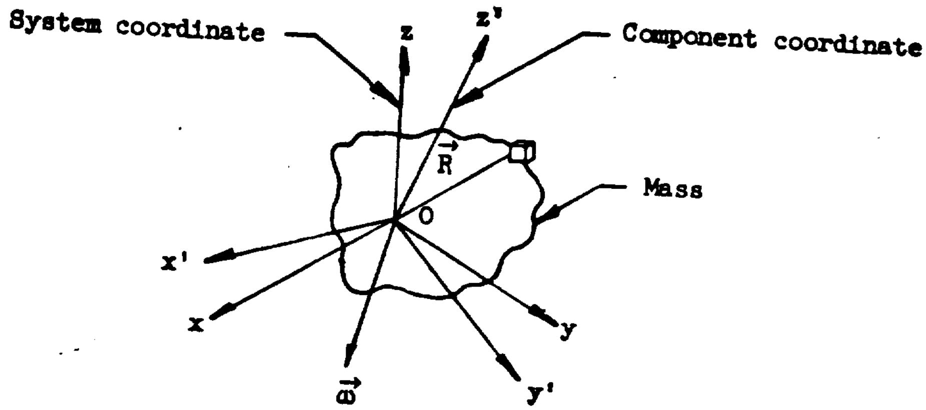
This particular selection of coordinates does not affect the final answers because kinetic energy is constant with regards to the coordinate orientation; however, it does simplify the input data and also reduces computer time.

Rotating From Component Coordinates to System Coordinates

At this point we have the kinetic energy about the component axis system. In the computer program it is at this level that mass properties are computed for the preselected shapes such as the cylinder, sphere, etc.

It is necessary to resolve the component mass properties to an axis system parallel to the system coordinates. Once parallel to this system axis then we can translate to the system center of gravity by the usual parallel axis theorem. The derivation is similar to that presented in reference 5.

In deriving the rotational transformation from the component to the system coordinates the expression for kinetic energy is again used. Given a body rotating with an angular velocity $\vec{\omega}$ we know that its kinetic energy is invariant with regard to the coordinate orientation.



The kinetic energy in the system coordinates is

$$T_{sys} = \frac{1}{2} \int (\vec{\omega} \times \vec{R}) \cdot (\vec{\omega} \times \vec{R})\, dm$$

Defining the products of inertia as

$$I_{xz} = \int yz\, dm, \quad I_{yz} = \int yz\, dm, \quad I_{xy} = \int xy\, dm$$

the energy equation becomes

$$T_{sys} = \frac{1}{2}\left[I_x\omega_x^2 + I_y\omega_y^2 + I_z\omega_z^2 + 2I_{xz}\omega_x\omega_z + 2I_{yz}\omega_y\omega_z + 2I_{xy}\omega_x\omega_y\right]$$

The products of inertia will not necessarily be zero in the system coordinates due to the coordinate rotations and, therefore, the products must be included.

It is necessary to write the angular velocity of the component system coordinates in terms of the system coordinates. For an arbitrary vector it can be written

$$\vec{\omega} = \vec{\omega}'_{comp} = \vec{\omega}_{system}$$

$$= i'\omega_x' + j'\omega_y' + k'\omega_z' = i\omega_x + j\omega_y + k\omega_z$$

Performing the dot product gives

$$\omega_x' = i' \cdot i\omega_x + i' \cdot j\omega_y + i' \cdot k\omega_z$$

$$\omega_y' = j' \cdot i\omega_x + j' \cdot j\omega_y + j' \cdot k\omega_z$$

$$\omega_z' = k' \cdot i\omega_x + k' \cdot j\omega_y + k' \cdot k\omega_z$$

Recognizing the direction cosines results in

$$\begin{Bmatrix} \omega_x' \\ \omega_y' \\ \omega_z' \end{Bmatrix} = \begin{bmatrix} l_{x'x} & l_{x'y} & l_{x'z} \\ l_{y'x} & l_{y'y} & l_{y'z} \\ l_{z'x} & l_{z'y} & l_{z'z} \end{bmatrix} \begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix}$$

Writing the component kinetic energy in matrix algebra

$$T_{comp} = \frac{1}{2}\{\omega'\}^T [I'][\omega'] = \frac{1}{2}\Big\{[DC]\{\omega\}\Big\}^T [I'] \Big\{[DC]\{\omega\}\Big\}$$

where [DC] is the direction cosine matrix shown above. Equating the system and component energy results in

$$T_{system} = T_{comp}$$

$$\frac{1}{2}\{\omega\}^T [I] \{\omega\} = \frac{1}{2}\{\omega\}^T [DC]^T [I'] [DC] \{\omega\}$$

and the resulting system inertial matrix is found to be

$$[I] = [DC]^T [I'] [DC]$$

Expanding the matrix equation we get

$$\begin{bmatrix} I_{XXCO} & I_{XYCO} & I_{XZCO} \\ I_{YXCO} & I_{YYCO} & I_{YZCO} \\ I_{ZXCO} & I_{ZYCO} & I_{ZZCO} \end{bmatrix} =$$

$$\begin{bmatrix} l_{x'x} & l_{y'x} & l_{z'x} \\ l_{x'y} & l_{y'y} & l_{z'y} \\ l_{x'z} & l_{y'z} & l_{z'z} \end{bmatrix} \begin{bmatrix} I'_x & 0 & 0 \\ 0 & I'_y & 0 \\ 0 & 0 & I'_z \end{bmatrix} \begin{bmatrix} l_{x'x} & l_{x'y} & l_{x'z} \\ l_{y'x} & l_{y'y} & l_{y'z} \\ l_{z'x} & l_{z'y} & l_{z'z} \end{bmatrix}$$

The direction cosines are determined by the method presented in appendix A.

Transferring by Parallel Axis Theorem

Now that the moments of inertia are in a system parallel to the system coordinates, we now translate by the parallel axis theorem

$$I_{XX} = I_{XXCO} + m\left[(Y_{co} - \bar{Y})^2 + (Z_{co} - \bar{Z})^2\right]$$

$$I_{YY} = I_{YYCO} + m\left[(X_{co} - \bar{X})^2 + (Z_{co} - \bar{Z})^2\right]$$

$$I_{ZZ} = I_{ZZCO} + m\left[(X_{co} - \bar{X})^2 + (Y_{co} - \bar{Y})^2\right]$$

$$I_{XY} = I_{XYCO} + m\left[(X_{co} - \bar{X})(Y_{co} - \bar{Y})\right]$$

$$I_{XZ} = I_{XZCO} + m\left[(X_{co} - \bar{X})(Z_{co} - \bar{Z})\right]$$

$$I_{YZ} = I_{YZCO} + m\left[(Y_{co} - \bar{Y})(Z_{co} - \bar{Z})\right]$$

Where I_{XXCO}, I_{YYCO}, and I_{ZZCO} are the component moments of inertia rotated parallel to the system coordinates, and $\overline{X}$, $\overline{Y}$, $\overline{Z}$ are the system center of mass coordinates and X_{co}, Y_{co}, Z_{co} are the coordinates of the component center of mass.

COMPUTER PROGRAM

This computer program is written in Fortran IV computer language. All names and descriptions are assigned in the first part of the program. Thirteen sections have been written using 13 common shapes usually found in spacecraft. The program is directed by the input data which singles out the section or shape factor desired to be used through the "go to" statement. The operation of the program is illustrated in figure 1 with a computer flow diagram.

The input data for each tiem is listed on two data cards. The basic input for each item will vary depending on the shape factor used. Each shape factor with the necessary data is discussed in the input data instructions.

After the data cards are supplied to the program the following operations are performed. The component mass properties are first printed with the moments of inertia about the component axis rotated parellel to the system coordinates. These mass properties are transferred to the system center of gravity and the following are computed: System weight, inertias about the system center of gravity, inertias about the origin, center of gravity of the system and products of inertias of the system. Based on this generated information, inertias about the system principal axes and their location is subsequently computed.

A listing of the computer program is found in appendix B.

Selection of Coordinate Points

The selection of points "i" and "j" determines the length of the member as well as the first three direction cosines. Point "k" is required to calculate the other six direction cosines. Shown in figure 2 are the two coordinate systems used in this program. Point "i" locates the system coordinates (X_i, Y_i, Z_i) for the origin of the component axes and point "j determines the direction of the "x" axis of the component coordinates.

In order to determine the directions of the "y" and "z" axes, point "k is required. This point can be anywhere in the x-y plane. If it is omitted, then the program automatically positions the "y" axis parallel to the X-Y plane. For a body of revolution point "k" is not required. The following figures (2(a) and 2(b)) describe points i, j, and k.

The main deck is referred to as the computer program without the necessary data. It is always necessary to have a 789 card following the main deck and a 789 card, then a 6789 card following the data. The 6789 card separates one program from another. A description of these cards and their formats follows figure 2.

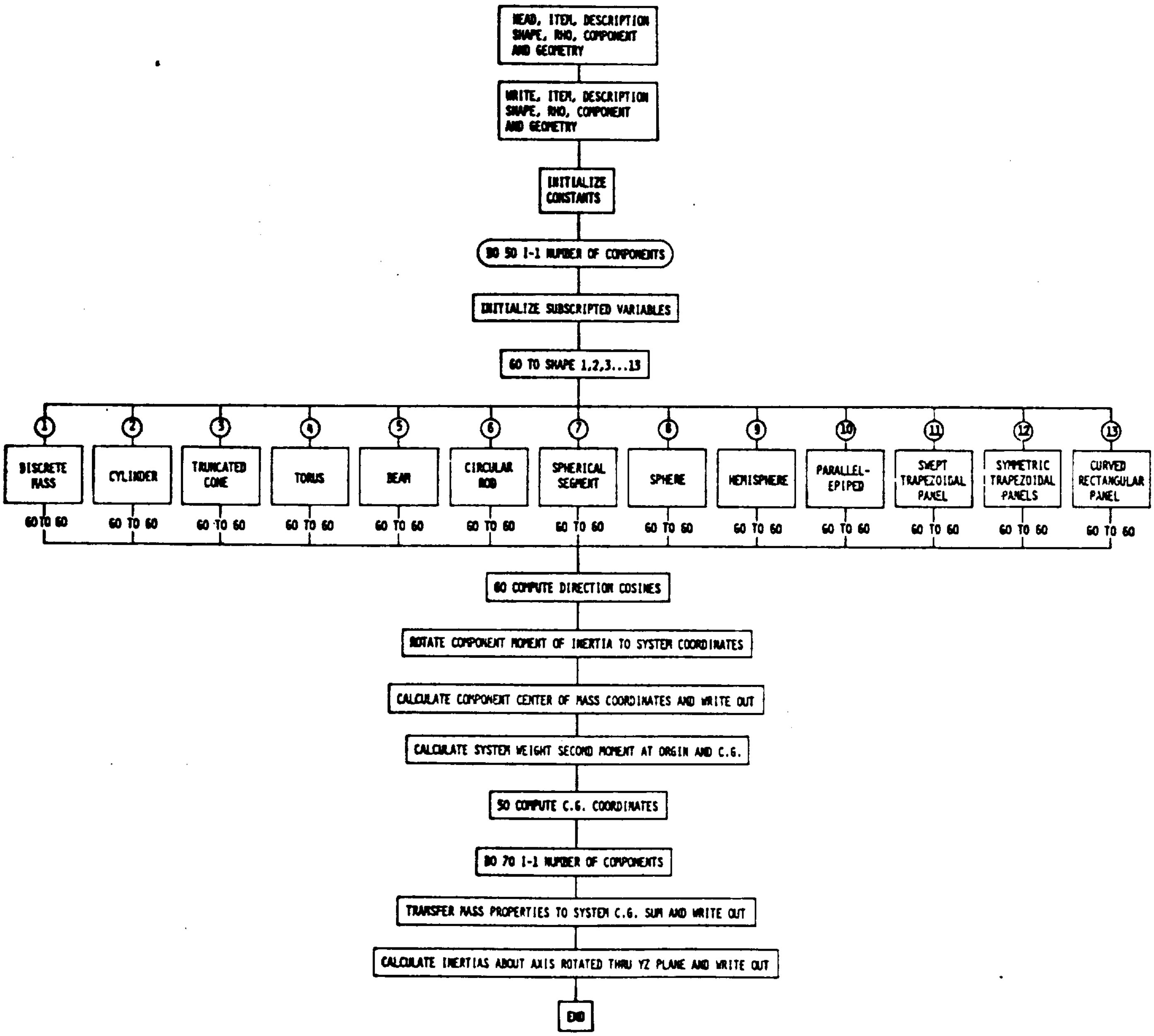
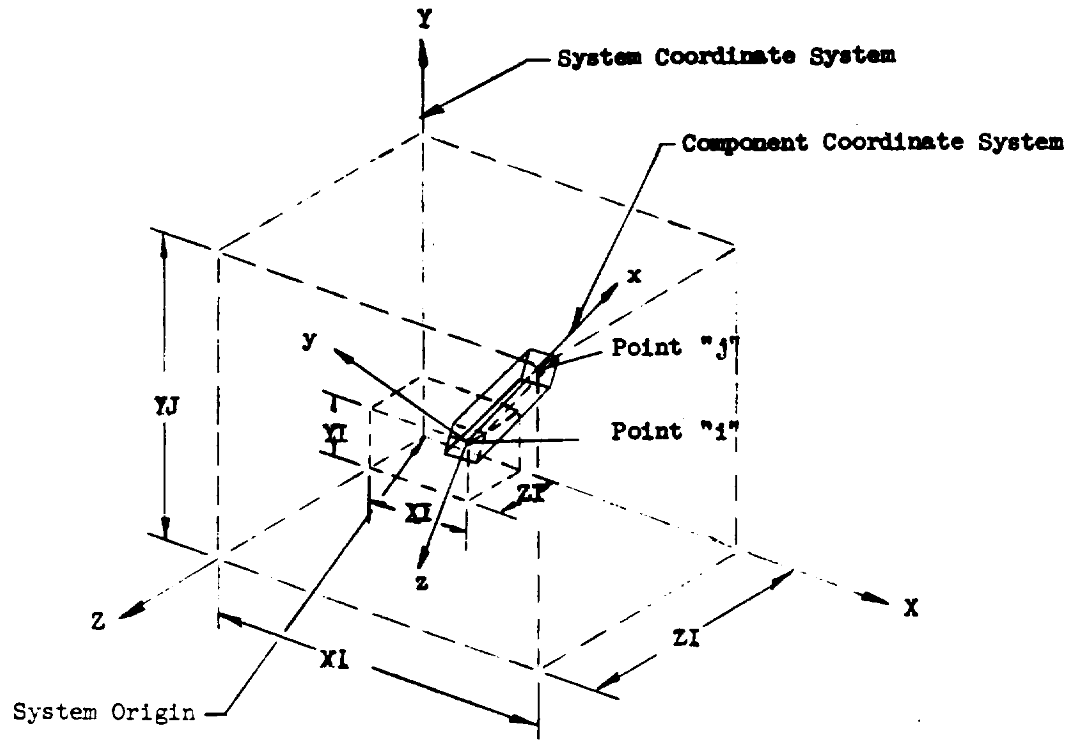
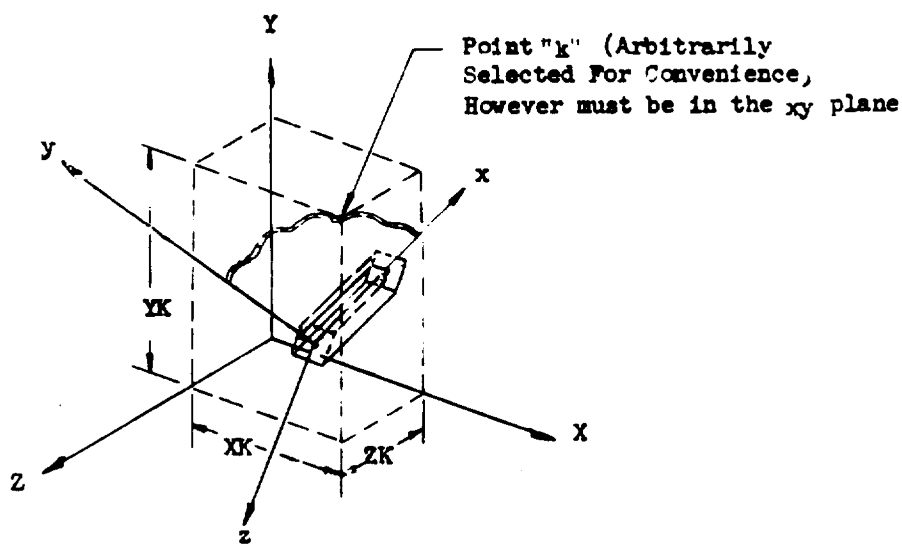


FIGURE 1 - COMPUTER FLOW DIAGRAM



LOCATION OF POINTS "i" AND 'j"

Figure 2(a)



LOCATION OF POINTS "k"

Figure 2(b)

1st Data Card

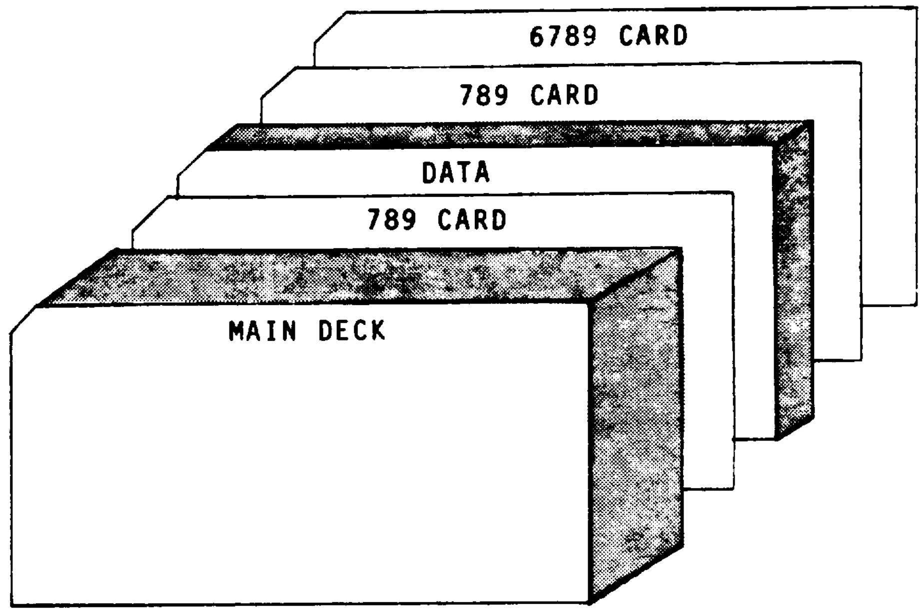
Item No.	I3 format Columns 1 through 3
Description	2A9 format Columns 4 through 21
Shape	I2 format Columns 22 and 23
Weight or density	F9.4 format Columns 24 through 32
A	F8.3 format Columns 33 through 40
B	F8.3 format Columns 41 through 48
C	F8.3 format Columns 49 through 56
D	F8.3 format Columns 57 through 64
F	F8.3 format Columns 65 through 72

It is to be noted that the input data variables A, B, C, D, and F can be geometric dimensions, cross-sectional areas, area moments of inertia, and mass moment of inertia.

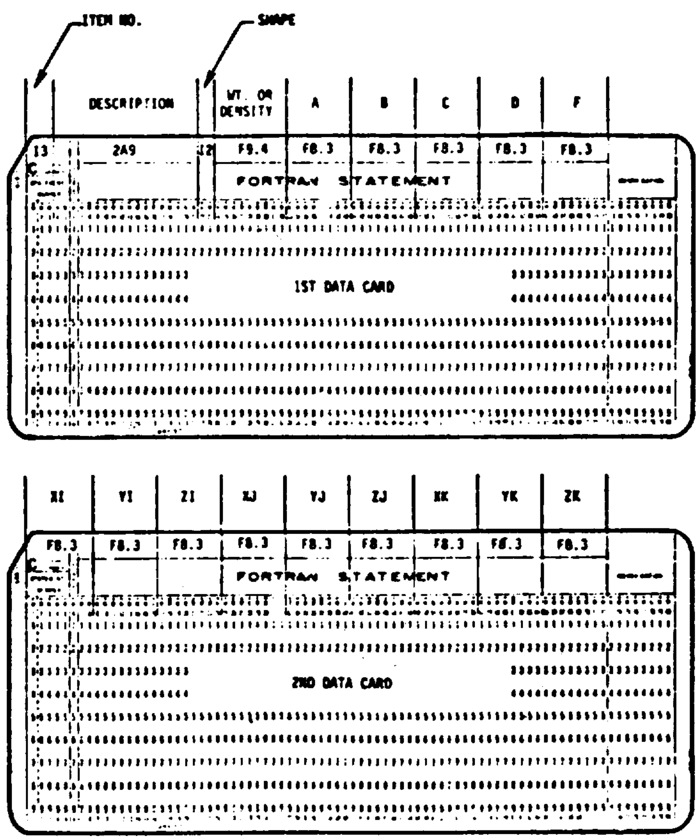
2nd Data Card

XI	F8.3 format Columns 1 through 8
YI	F8.3 format Columns 9 through 16
ZI	F8.3 format Columns 17 through 24
XJ	F8.3 format Columns 25 through 32
YJ	F8.3 format Columns 33 through 40
ZJ	F8.3 format Columns 41 through 48
XK	F8.3 format Columns 49 through 56
YK	F8.3 format Columns 57 through 64
ZK	F8.3 format Columns 65 through 72

ORDER OF CARDS FOR COMPUTER PROGRAM



DATA CARDS AND FORMATS
(INPUT DATA FOR EACH ITEM)



Component shapes in the existing programs are:

Shape 1	Discrete Mass
Shape 2	Cylinder
Shape 3	Truncated Cone
Shape 4	Torus
Shape 5	Beam (Arbitrary Cross Section)
Shape 6	Circular Road (Arbitrary Cross Section)
Shape 7	Spherical Segment
Shape 8	Sphere
Shape 9	Hemisphere
Shape 10	Parallelepiped
Shape 11	Swept Trapezoidal Panel
Shape 12	Symmetric Swept Trapezoidal Panels
Shape 13	Curved Rectangular Panel

This program can easily be modified to include additional shapes.

Frequently where precise weights of components are known it is more convenient to input this weight rather than an average density which would have to be calculated. The value of .4 has been chosen as the limiting value for inputing Rho as density (lb/cu in). A value greater than .4 is used as total weight (lbs). In some cases referred to as "thin wall" the total weight of the component must be input. The degree of flexibility for inputing various shapes can be determined from the "shape data input instructions".

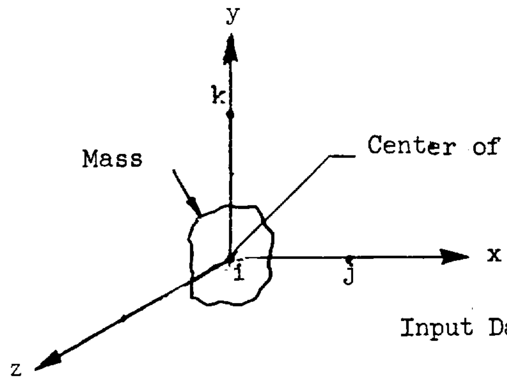
The program can be used to determine the mass properties of a component with hollows or voids. The component is treated as a standard shape with the hollows or voids included as solid material. The hollows or voids are then input as standard shapes having negative values for weight or density. The program will compute the actual weight, center of gravity, and moments and products of inertia of the component.

In the following programs for a variety of shapes the inertias about the x, y and z component axes are represented respectively by computer symbols IXXCG, IYYCG and IZZCG. For some of the shapes expressions for the moments of inertia are given; however, for the more complex shapes they are omitted but can be found by referring to the computer program in Appendix B.

Note that component axes have been located so the x axis is a principal axis and the y and z axes either coincide with, or are parellel to principal axes. Any other than the aforementioned x, y, and z axes locations will incur an error in the main computer program with shape no. 11 being the sole exception. In this instance, the x, y, and z axes locations were chosen for convenience in accordance with the "shape data input instruction" and the x axis is rotated to a principal axis by the program.

SHAPE DATA INPUT INSTRUCTION

SHAPE 1
DISCRETE MASS



IXXCG = A
IYYCG = B
IZZCG = C

Input Data: Item, Description, Shape, RHO (wt.)
A(Ixx), B(Iyy), C(Izz)

XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

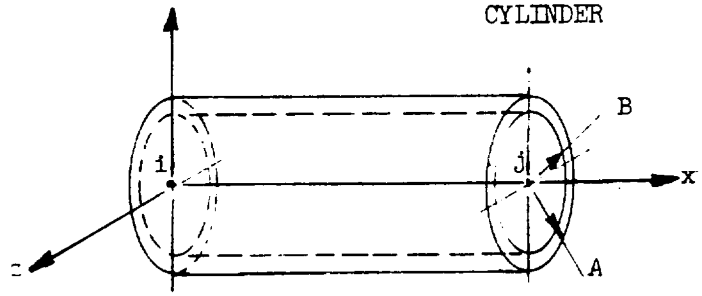
OPTIONS AVAILABLE

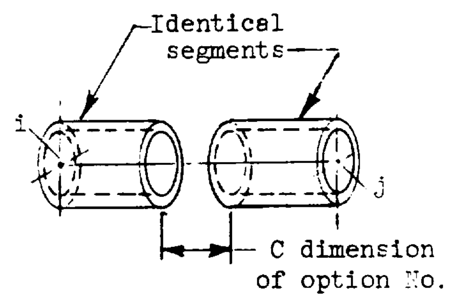
No.1 Input data just as indicated above.

No.2 If negligible, inertias A(Ixx), B(Iyy) and C(Izz) may be omitted, in which case points j and k are not required.

NOTES: Never locate point j at the system origin and if input, A(Ixx), B(Iyy) and C(Izz) must be inertias about principal axes in slug-ft.^{2}

SHAPE 2
CYLINDER





Input Data: Item, Description, Shape, RHO (density)
A, B, C (option No. 2 only)

XI, YI, ZI, XJ, YJ, ZJ

OPTIONS AVAILABLE

No.1 Input data just as indicated above.

No.2 Cylinder may be segmented requiring a C value be input.

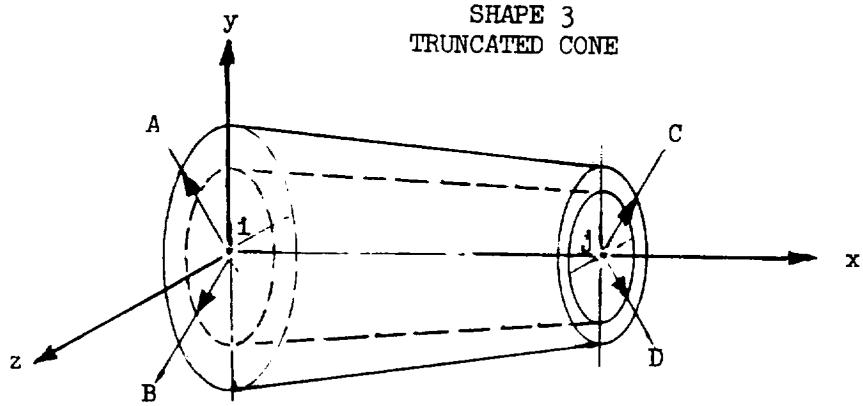
No.3 Total weight may be input for RHO in both previous options but only if it is input more than .4 pounds.

No.4 Input A and B equal and total weight for RHO and program treats shape as a thin-wall cylinder.

NOTES: Point k is not required with any option and density as such must never be input more than .4 pounds per cu. in.

SHAPE DATA INPUT INSTRUCTION

SHAPE 3
TRUNCATED CONE

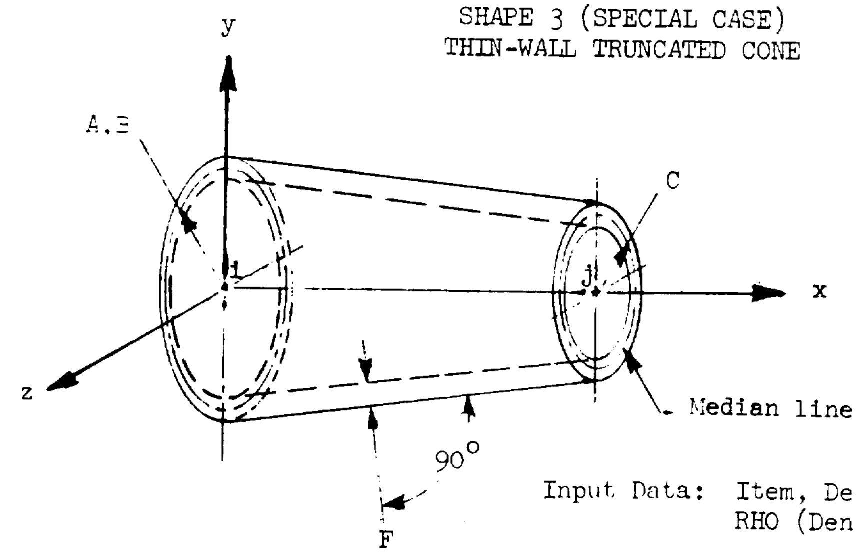


Input Data: Item, Description, Shape, RHO (density),A,B,C,D
XI, YI, ZI, XJ, YJ, ZJ

OPTIONS AVAILABLE
No.1 Input data just as indicated above
No.2 A total weight may be input for RHO but only if it is more than .4 pounds.

NOTES: The values of A minus C and or B minus D must never equal zero.
Point i is always at the cones Larger end and point k is not required.
Density as such must never be input greater than .4 lbs./cu.in.

SHAPE 3 (SPECIAL CASE)
THIN-WALL TRUNCATED CONE

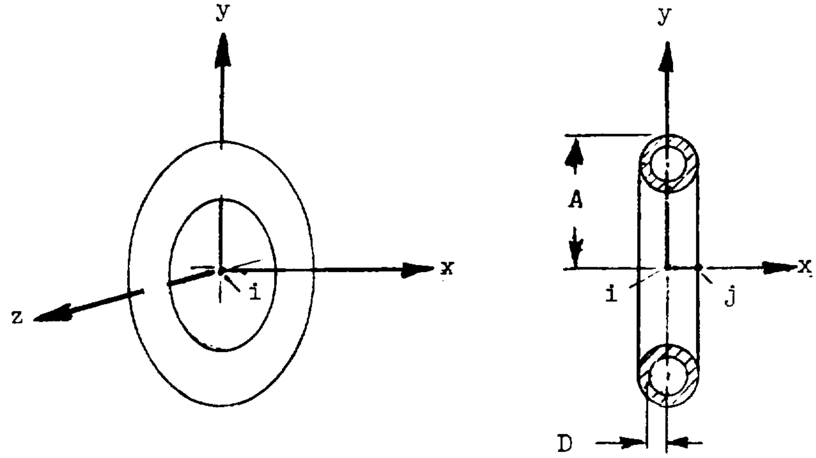


Input Data: Item, Description, Shape
RHO (Density),A,B,C,F
XI, YI, ZI, XJ, YJ, ZJ

NOTES: In this case program assumes all mass is concentrated midway between inner and outer surfaces.
Point i is always at the cone's larger end and point k is not required.

SHAPE DATA INPUT INSTRUCTION

SHAPE 4
TORUS

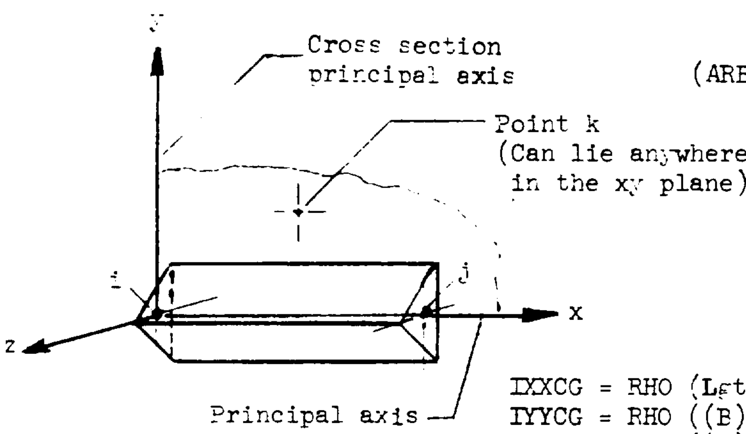


IXXCG = Y11-Y12
IYYCG = X11-X12
IZZCG = IYYCG

Input Data: Item. Description, Shape, RHO (density), A, D
XI, YI, ZI, XJ, YJ, ZJ

NOTE: Point "k" is not required.

SHAPE 5
BEAM
(ARBITRARY CROSS SECTION)



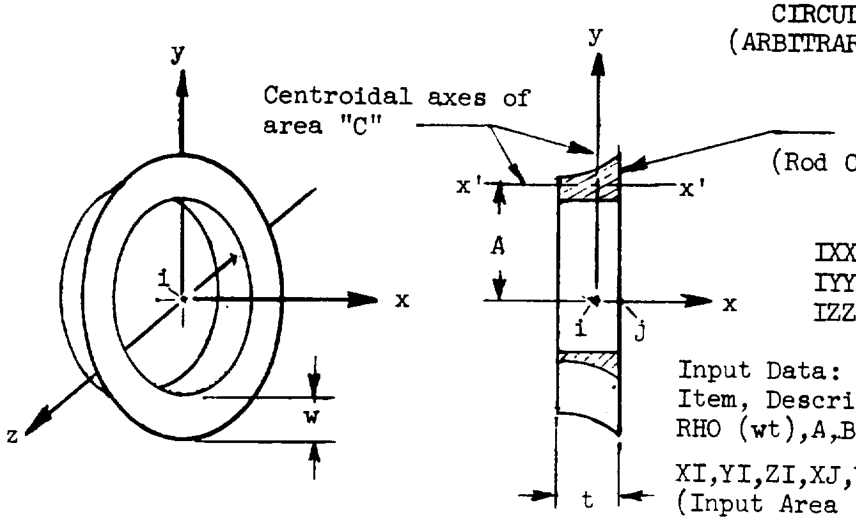
IXXCG = RHO (Lgth)(B + C)
IYYCG = RHO ((B)(Lgth) + 0.0833 (A)(Lgth)^3)
IZZCG = RHO ((C)(Lgth) + 0.0833 (A)(Lgth)^3)

Input Data: Item, Description, Shape, RHO (density), A (area),
B(Iyy), C(Izz) (Area moment of inertia in inches^4)
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

NOTES: **If beam is a body of revolution about the x (centroidal) axis, point k is not required. Iyy and Izz are area moments of inertia about principal axes of the beam cross section taken in a plane normal to x axis.**

SHAPE DATA INPUT INSTRUCTION

SHAPE 6
CIRCULAR ROD
(ARBITRARY CROSS SECTION)

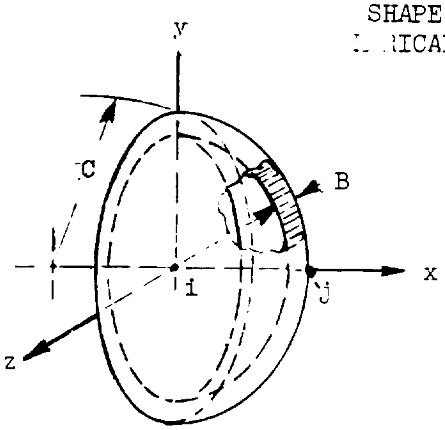


IXXCG = $(RSCG)^2(wt)$
IYYCG = $.5(RSCG)^2(wt)$
IZZCG = IYYCG

Input Data:
Item, Description, Shape
RHO (wt),A,B(Area "C" Ix'x'),C
XI,YI,ZI,XJ,YJ,ZJ
(Input Area "C" Ix'x' in inches^4)

NOTES:
Point "k" is not required and weight instead of density is the required input for RHO.
The solution for this shape is approximate in that it may be as much as one % less than correct if rod dimensions t and w are as much as 25% of dimension A. (Error incurred tends to increase as this % increases)

SHAPE -
[illegible]RICAL SEGMENT



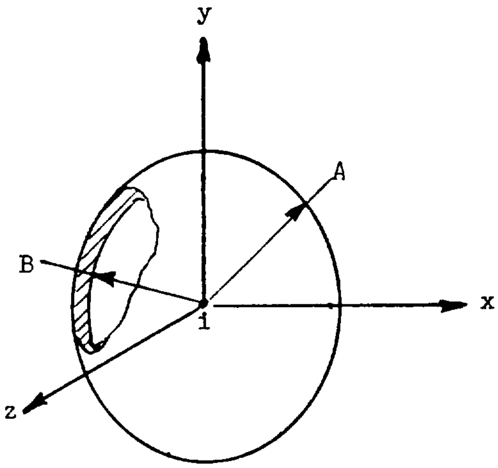
IXXCG = XI1 - XI2
IYYCG = ACTEMP
IZZCG = IYYCG

Input Data:
Item, Description, Shape, RHO (density),
B, C (outside radius)
XI,YI,ZI,YJ,YJ,ZJ

OPTIONS AVAILABLE
No.1 Input data just as indicated above
No.2 Total weight may be input for RHO if more than .4 lb.

NOTES:
Point k is not required. When computing as a solid spherical segment, B dim. will become equal the distance between points "i" and "j". Density as such must never be input greater than .4 lbs./cu. in.

SHAPE DATA INPUT INSTRUCTION



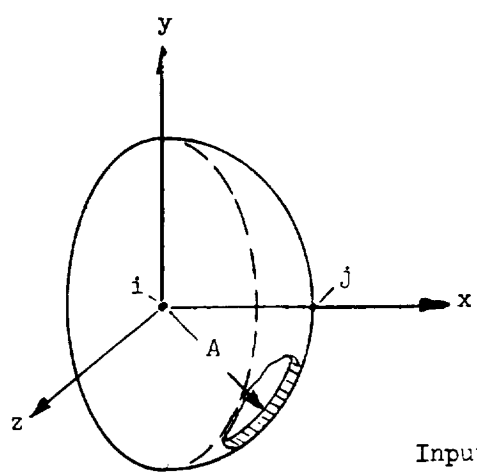
SHAPE 8
SPHERE

IXXCG = XI1-XI2
IYYCG = IXXCG
IZZCG = IXXCG

Input Data: Item, Description, Shape, RHO (density), A,B

XI, YI, ZI

NOTE: Points "j" and "k" are not required. If B = A the program selects thin-wall equations; therefore, weight instead of density should be input for RHO.



SHAPE 9
HEMISPHERE

IXXCG = XI1-XI2
IYYCG = XI3-XI4
IZZCG = IYYCG

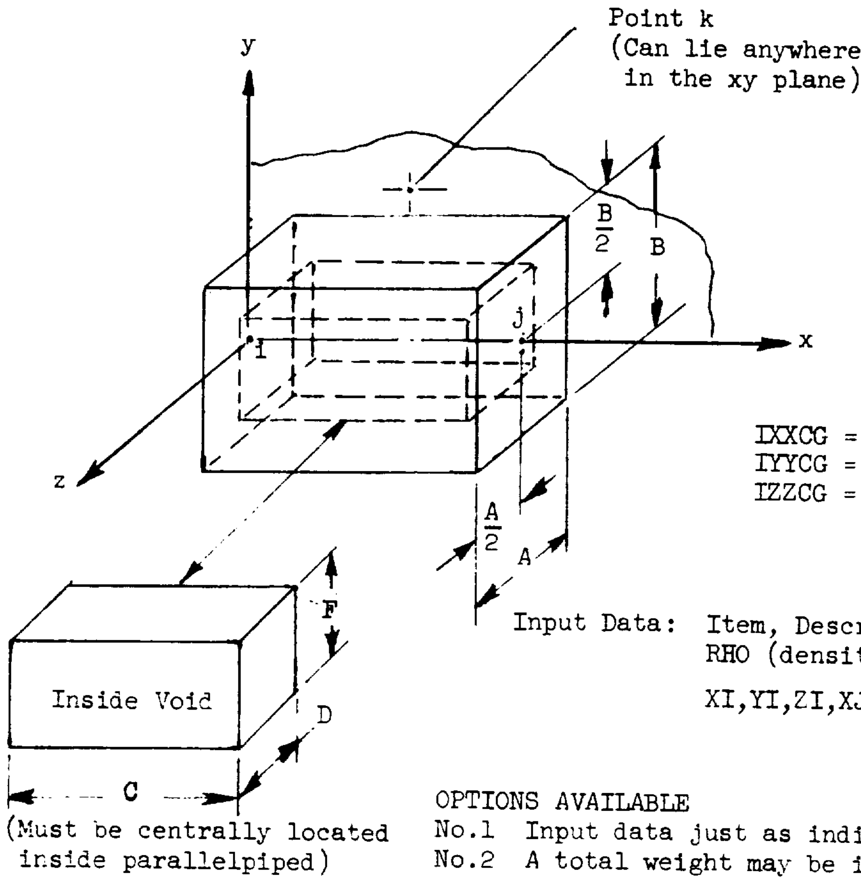
Input Data: Item, Description, Shape, RHO (density), A

XI, YI, ZI, XJ, YJ, ZJ

NOTES: Point "k" is not required. If A = Lgth (i,j) the program selects thin-wall equations; therefore, weight insted of density should be input for RHO.

SHAPE DATA INPUT INSTRUCTION

SHAPE 10
PARALLELEPIPED



IXXCG = XI1-XI2
IYYCG = XI3-XI4
IZZCG = XI5-XI6

Input Data: Item, Description, Shape,
RHO (density),A,B,C,D,F
XI,YI,ZI,XJ,YJ,ZJ,XK,YK,ZK

OPTIONS AVAILABLE

No.1 Input data just as indicated above.

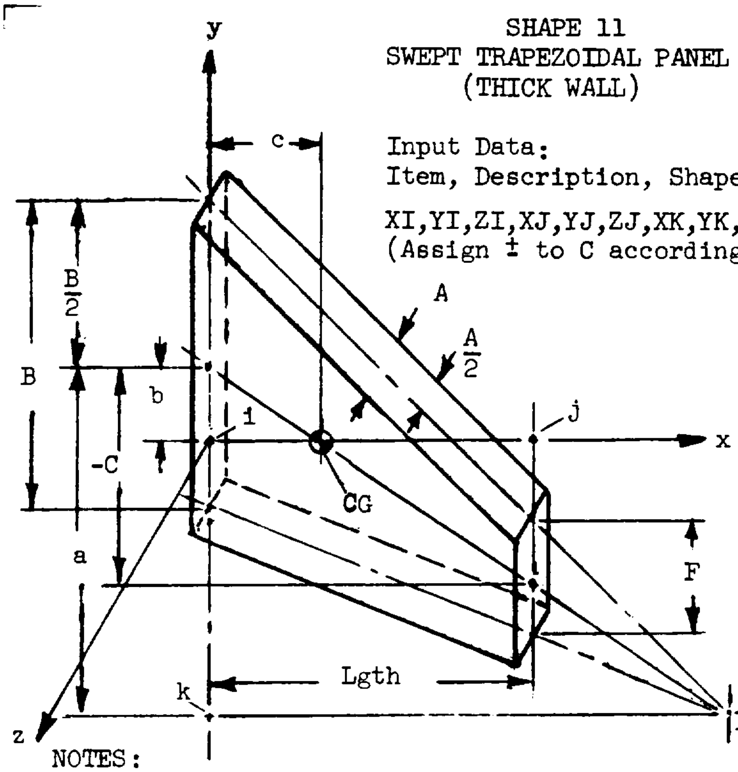
No.2 A total weight may be input for RHO but only if it is more than .4 pounds.

No.3 Input D equal A and the program selects thin-wall equations therefore total weight must be input for RHO.

NOTE: Density as such must never be input greater than .4 lbs./cu.in.

SHAPE DATA INPUT INSTRUCTION

SHAPE 11
SWEPT TRAPEZOIDAL PANEL
(THICK WALL)



Input Data:
Item, Description, Shape, RHO (density),A,B,±C,F

XI,YI,ZI,XJ,YJ,ZJ,XK,YK,ZK
(Assign ± to C according to sweep of panel)

Formulas designed to facilitate input data determination:

$$a = \frac{C(B)}{B-F}$$

$$b = \frac{C(B+2F)}{3(B+F)}$$

$$c = \frac{Lgth(B+2F)}{3(B+F)}$$

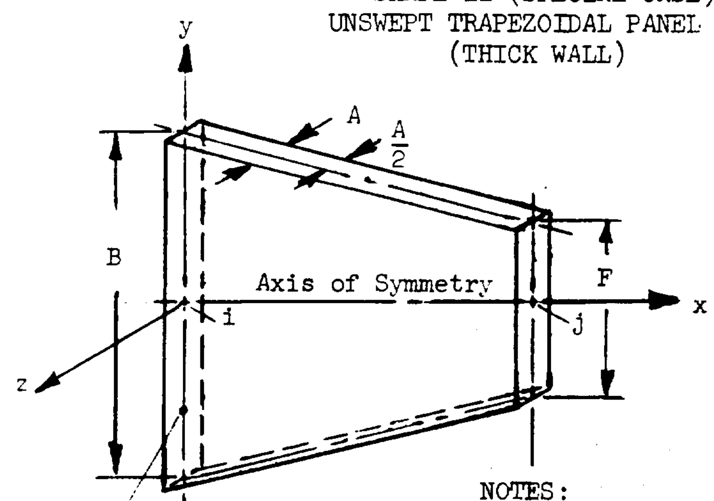
OPTIONS AVAILABLE

No.1 Input data just as indicated above

No.2 Total weight may be input for RHO if more than .4 lbs.

NOTES:
Dimension F must never be more than 98% of B. Density as such must never be input greater than .4 lb./cu.in.

SHAPE 11 (SPECIAL CASE)
UNSWEPT TRAPEZOIDAL PANEL
(THICK WALL)



Input Data:
Item, Description, Shape, RHO (density),A,B,F

XI,YI,ZI,XJ,YJ,ZJ,XK,YK,ZK

OPTIONS AVAILABLE

No.1 Input data just as indicated above.

No.2 Total weight may be input for RHO if more than .4 lbs.

Point k (Can lie anywhere on the y axis except at point i)

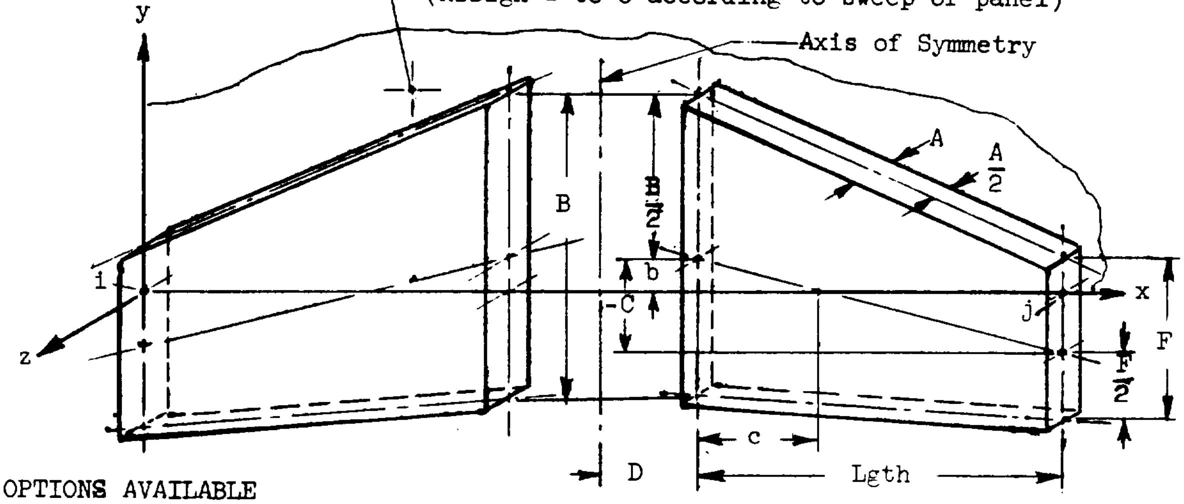
NOTES:
Dimension F must never be more than 98% of B. Density as such must never be input greater than .4 lbs./cu.in.

SHAPE DATA INPUT INSTRUCTION

SHAPE 12
SYMMETRIC SWEPT TRAPEZOIDAL PANELS (THICK WALL)

Point k
(Can lie anywhere in the xy plane)

Input Data:
Item, Description, Shape, RHO (density), A, B, ±C, D, F
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK
(Assign ± to C according to sweep of panel)



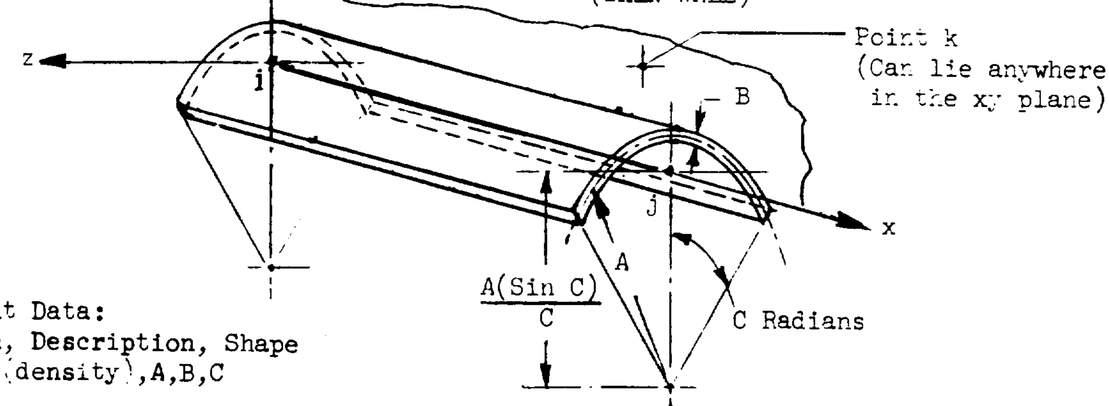
OPTIONS AVAILABLE
No.1 Input data just as indicated above.
No.2 Total weight may be input for RHO if it is more than .4 lb.

NOTES:
Dimension F must be no more than 98% of B.
Density as such must never be input greater than .4 lb./cu.in.

Formulas designed to facilitate input data determination:

$$b = \frac{C(B+2F)}{3(B+F)} \qquad c = \frac{Lgth(B+2F)}{3(B+F)}$$

SHAPE 13
CURVED RECTANGULAR PANEL (THIN WALL)



Input Data:
Item, Description, Shape
RHO (density), A, B, C

XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

NOTE: Plane xy is one about which symmentry exists.

EXAMPLE PROBLEMS

Two example problems are presented in order to show required input data. These problems were selected due to the simple calculations involved and thus could be checked by hand calculations.

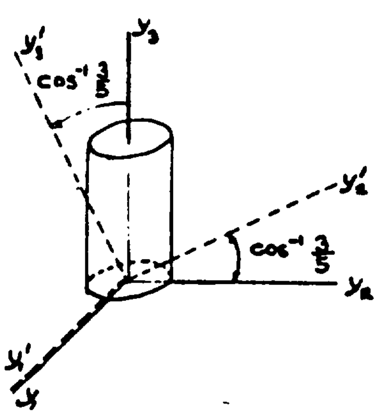
Example Problem 1

Problem 1 was taken from reference 3. It is a cylinder skewed in the y-z plane. The moments of inertia about the system origins are given in this reference and are used for comparison in this paper. It should be noted that "g" (acceleration of gravity) was taken to be 32.0 ft/sec^2 in this reference rather than 32.2 ft/sec^2 (386 in./sec^2). Depending on the value of "g" selected by the program user, the term "cons" has to be changed accordingly.

The axis through the center of the component must always be the x-axis for the computer program. Therefore, the y_3 axis of this problem corresponds with the x-axis of the computer program; the y_1 axis corresponds with the y-axis and the y_2 axis corresponds with the z-axis.

Find moments of inertia I'_{11}, $I'22$, $I'33$ which corresponds to I_{xxo}, I_{yyo}, I_{zzo} in our coordinate system.

Given: W = 1 slug = 32 lb
R = 24 inches
$\llcorner$ = 36 inches
$\theta = \cos^{-1} 3/5 = 53^\circ\ 8'$



Solution: (a) Direction cosines for the transformation are

$$a_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{3}{5} & \frac{4}{5} \\ 0 & -\frac{4}{5} & \frac{3}{5} \end{bmatrix}$$

And numerical values of the I_{ij} are

$$I_{ij} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$ Component Moment of Inertia

ORIGINAL PAGE IS
OF POOR QUALITY

The results of problem 1 as computed in reference 3 are now given. Hence, from the transformation equations

System Moments of Inertia

$$I'_{11} = a_{11}a_{11}I_{11} + a_{12}a_{12}I_{22} + a_{13}a_{13}I_{33} + 2a_{12}a_{13}I_{23} + 2a_{11}a_{13}I_{13} + 2a_{11}a_{12}I_{12}$$

$$I_{xx} = (1)^2\, 4 + 0 + 0 + 0 + 0 + 0 = 4 = \text{slug ft}^2 \quad (I_{xx} = I'_{11})$$

$$I'_{22} = a_{21}a_{21}I_{11} + a_{22}a_{22}I_{22} + a_{23}a_{23}I_{33} + 2a_{22}a_{23}I_{23} + 2a_{21}a_{23}I_{13} + 2a_{21}a_{22}I_{12}$$

$$I_{yy} = 0 + \left(\frac{3}{5}\right)^2 4 + \left(\frac{4}{5}\right)^2 2 + 0 + 0 + 0 = \frac{68}{25} = 2.72 \text{ slug ft}^2 \quad (I_{yy} = I'_{22})$$

$$I'_{33} = a_{31}a_{31}I_{11} + a_{32}a_{32}I_{22} + a_{33}a_{33}I_{33} + 2a_{32}a_{33}I_{23} + 2a_{31}a_{33}I_{13} + 2a_{31}a_{32}I_{12}$$

$$I_{zz} = 0 + \quad -\frac{4}{5}^{\,2}\, 4 + \quad \frac{3}{5}^{\,2}\, 2 + 0 + 0 + 0 = \frac{82}{25} = 3.28 \text{ slug ft}^2$$

This problem is now computed using Computer Program ND0702. Shown below are the coordinates of points i, j, and k for problem 1.

Point "i"

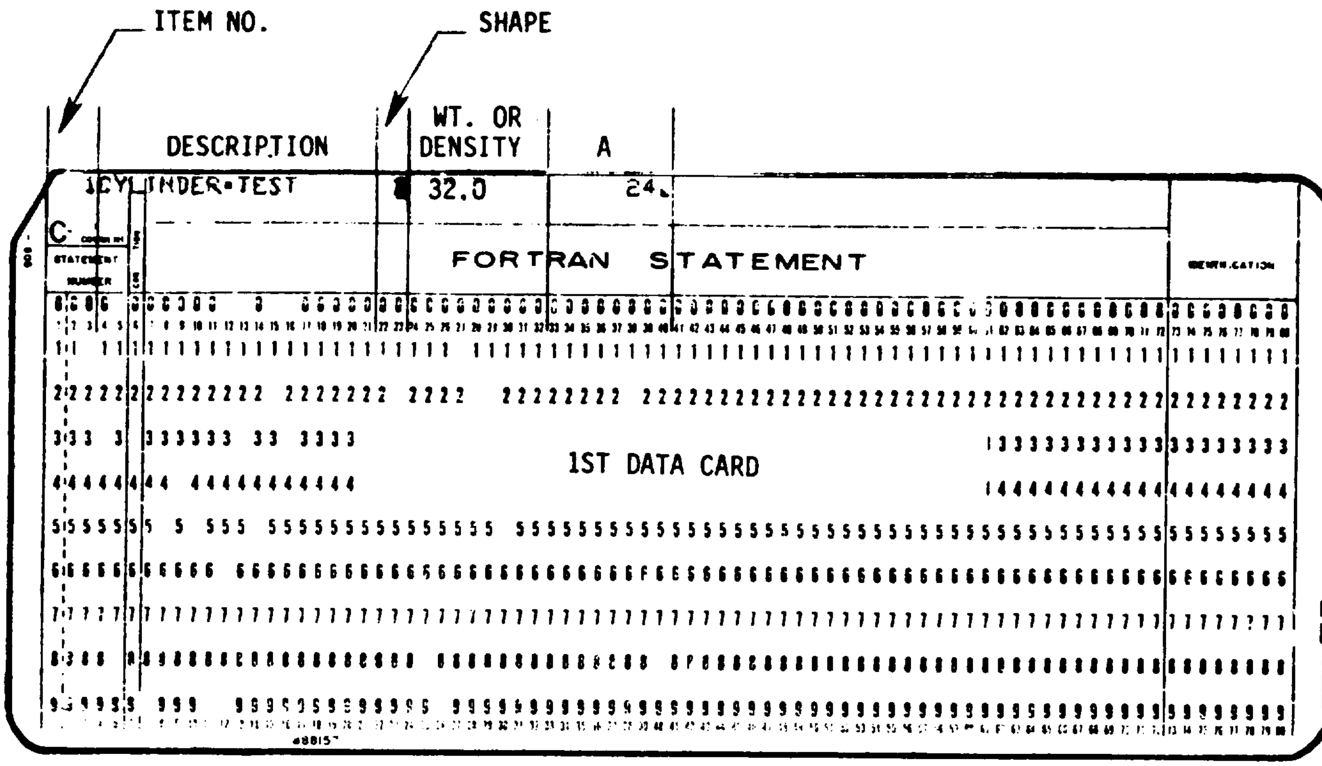
$$\begin{cases} X_i = 0 \\ Y_i = 0 \\ Z_i = 0 \end{cases}$$

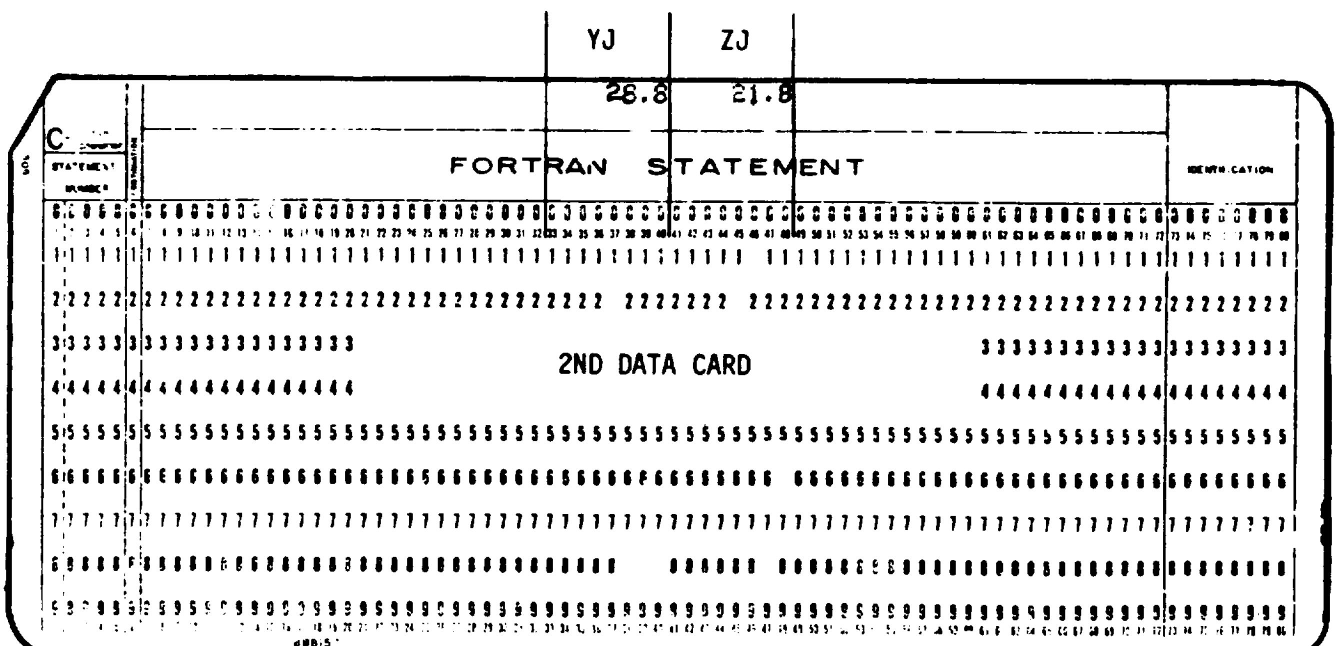
Point "j"

$$\begin{cases} X_j = 0 \\ Y_j = 36 \sin 53^\circ 8' = 28.8 \\ Z_j = 36 \cos 53^\circ 8' = 21.6 \end{cases}$$

Point "k" Omit since not required for cylinder

Reference is now made to the necessary data cards to compute this problem.


ITEM NO.
SHAPE
DESCRIPTION
WT. OR DENSITY
A
1CYLINDER-TEST
32.0
24.
FORTRAN STATEMENT
1ST DATA CARD


YJ
ZJ
26.8
21.8
FORTRAN STATEMENT
2ND DATA CARD

We now have the results of problem 1 using Computer Program

(Note that the gravitational constant used here was 32.166·· rather than 32.0 used in ref.)

```
INPUT DATA LISTED BELOW

ITEM      DESCRIPTION       SHAPE      RHO               A          B          C          D          F
  1     CYLINDER, TEST        2    .32000000E+02      20.000      0.000      0.000      0.000      0.000
       XI          YI          ZI          XJ           YJ          ZJ          XK          YK          ZK
     0.000       0.000       0.000       0.000       20.000      21.000       0.000       0.000       0.000

COMPONENT  DATA  LISTED  BELOW

ITEM      DESCRIPTION       WT        IXXCO     IYYCO     IZZCO     XCGCO     YCGCO    ZCGCO     IXYCO     IYZCO   IXZCO

  1    CYLINDER, TEST    32.00000   1.74093   1.90010   1.83047   0.00000  10.00000 10.00000  0.00000   .11938  0.00000

SYSTEM DATA LISTED BELOW (WT=LBS, INERTIAS=SLUGS FT SQUARED, C.G.=INS, SECOND MOMENT=SLUG FT SQUARED)

       SYS WT          IXXO          IYYO          IZZO          XBAR          YBAR          ZBAR
       32.000          3.970         2.706         3.263         0.000        10.000        10.000
        IXX            IYY           IZZ           IXY           IXZ           IYZ
       1.74093        1.90010       1.83047       0.00000       0.00000       .11938

INERTIAS (EIGENVALUES) ABOUT SYSTEM PRINCIPAL AXES WITH AXIS DIRECTION COSINES (EIGENVECTORS) RELATING THE PRINCIPLE AXES
TO THE X, Y, AND Z SYSTEM AXES IN THAT SEQUENCE

EIGENVALUE(1) =   .17409E+01

EIGENVECTOR(1)

  .100000E+01      0.            0.

EIGENVALUE(2) =   .17409E+01

EIGENVECTOR(2)

 0.             -.600000E+00      .800000E+00

EIGENVALUE(3) =   .19098E+01

EIGENVECTOR(3)

 0.              .800000E+00      .600000E+00
```

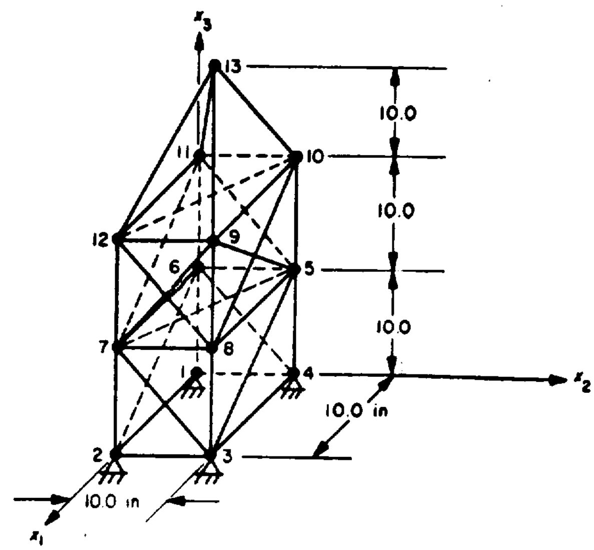
Example Problem 2

Problem 2 was taken from reference 2. In this space structure, the weight has been 'lumped' or concentrated at the joints. In the program being presented, this is not required but is used here only for illustration. To compare moment of inertia, the numbers given in reference 2 should be converted to slug-ft^2.

$$IXX = IYY - \frac{22038.464}{4.608E3} = 4.783 \text{ slug-ft}^2$$

$$IZZ = \frac{9000.00}{4.0608F3} = 1.953 \text{ slug-ft}^2$$

Example Problem 2



INPUT:
MEMBER AREAS – 0.01 in^2
WEIGHT AT EACH JOINT – 15.0 lb
MODULUS OF ELASTICITY – 10^7 psi
SEE THE FIRST TWO PAGES OF THE SAMPLE PROBLEM FOR DETAILS OF THE INPUT

JOINT COORDINATES

JOINT	X1	X2	X3
1	0.	0.	0.
2	10.00000	0.	0.
3	10.00000	10.00000	0.
4	0.	10.00000	0.
5	0.	10.00000	10.00000
6	0.	0.	10.00000
7	10.00000	0.	10.00000
8	10.00000	10.00000	10.00000
9	10.00000	10.00000	20.00000
10	0.	10.00000	20.00000
11	0.	0.	20.00000
12	10.00000	0.	20.00000
13	5.00000	5.00000	30.00000

The results of problem 2 as computed in reference 6 are now given.

```
TOTAL WEIGHTS=     195.000     195.000     195.000     0.     0.     0.
CENTER OF WEIGHT   X=   5.000,    Y=   5.000,    Z=   11.538
WEIGHT MOMENTS OF INERTIA ABOUT CENTER OF WEIGHT
    IXX= 22038.462,   IYY= 22038.462,   IZZ= 9000.000,   IXY=  -0.   IXZ=  0.000,   IYZ=  0.000
```

NOTE: Inertias are given in pound inches squared and must be converted to compare with results program yields.

This problem is now computed using Computer Program ND0702

INPUT DATA LISTED BELOW

ITEM	DESCRIPTION	SHAPE	RHO	A	B	C	D	F	XI	YI	ZI	XJ	YJ	ZJ	XK	YK	ZK
1	NODE 1	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	NODE 2	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	10.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	NODE 3	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	10.000	10.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
4	NODE 4	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	0.000	10.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
5	NODE 5	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	0.000	10.000	10.000	0.000	0.000	0.000	0.000	0.000	0.000
6	NODE 6	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	0.000	0.000	10.000	0.000	0.000	0.000	0.000	0.000	0.000
7	NODE 7	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	10.000	0.000	10.000	0.000	0.000	0.000	0.000	0.000	0.000
8	NODE 8	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	10.000	10.000	10.000	0.000	0.000	0.000	0.000	0.000	0.000
9	NODE 9	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	10.000	10.000	20.000	0.000	0.000	0.000	0.000	0.000	0.000
10	NODE 10	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	0.000	10.000	20.000	0.000	0.000	0.000	0.000	0.000	0.000
11	NODE 11	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	0.000	0.000	20.000	0.000	0.000	0.000	0.000	0.000	0.000
12	NODE 12	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	10.000	0.000	20.000	0.000	0.000	0.000	0.000	0.000	0.000
13	NODE 13	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000	5.000	5.000	30.000	0.000	0.000	0.000	0.000	0.000	0.000

COMPONENT DATA LISTED BELOW

ITEM	DESCRIPTION	WT	IXXCO	IYYCO	IZZCO	XCGCO	YCGCO	ZCGCO	IXYCO	IYZCO	IXZCO
1	NODE 1	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2	NODE 2	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3	NODE 3	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	0.00000	0.00000	0.00000	0.00000
4	NODE 4	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	0.00000	0.00000	0.00000	0.00000
5	NODE 5	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	10.00000	0.00000	0.00000	0.00000
6	NODE 6	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	10.00000	0.00000	0.00000	0.00000
7	NODE 7	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	10.00000	0.00000	0.00000	0.00000
8	NODE 8	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	10.00000	0.00000	0.00000	0.00000
9	NODE 9	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	20.00000	0.00000	0.00000	0.00000
10	NODE 10	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	20.00000	0.00000	0.00000	0.00000
11	NODE 11	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	20.00000	0.00000	0.00000	0.00000
12	NODE 12	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	20.00000	0.00000	0.00000	0.00000
13	NODE 13	15.00000	0.00000	0.00000	0.00000	5.00000	5.00000	30.00000	0.00000	0.00000	0.00000

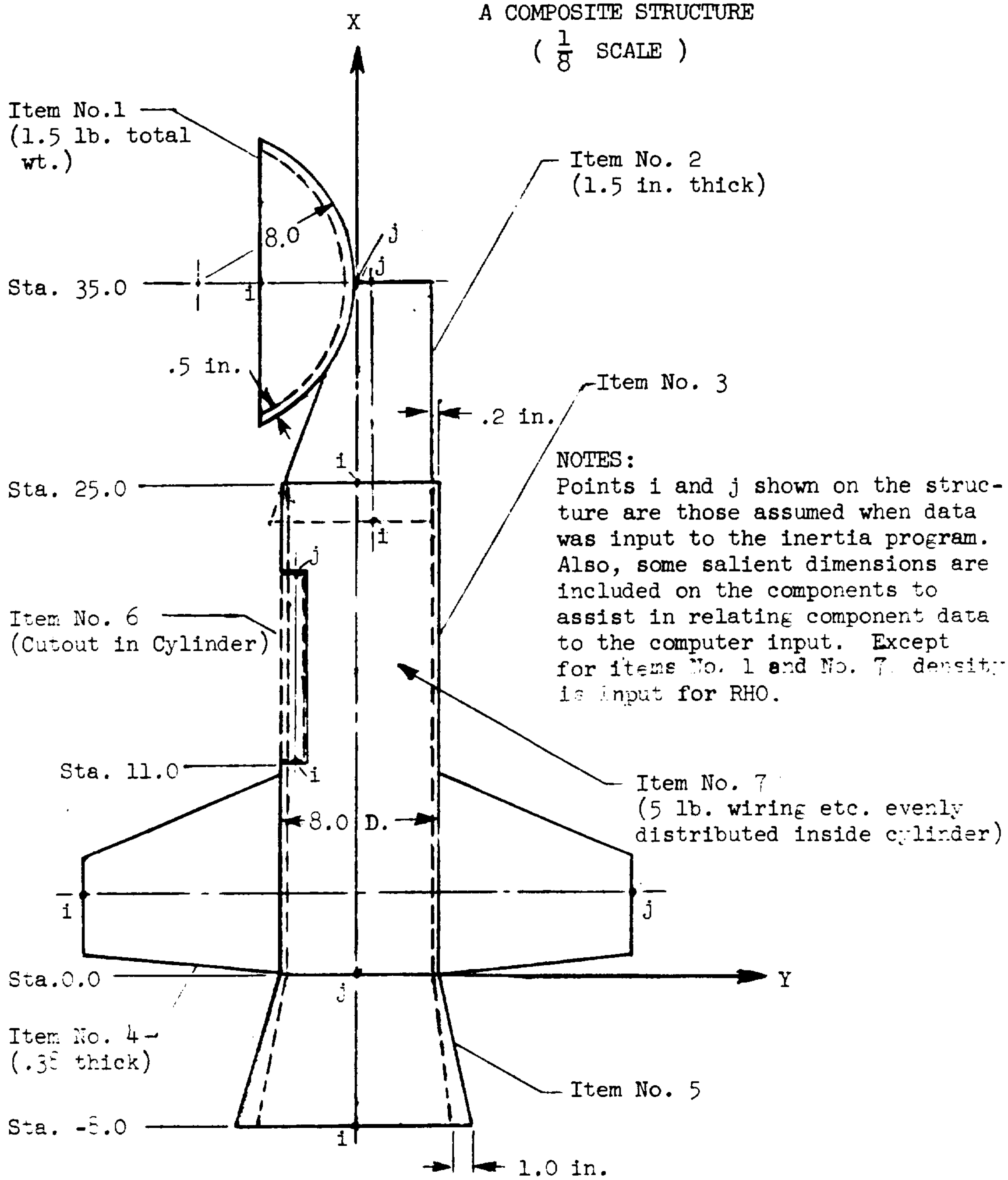
SYSTEM DATA LISTED BELOW (WT=LBS, INERTIA=SLUGS FT SQUARED, C.G.=INS, SECOND MOMENT=SLUG FT SQUARED)

SYS WT	IXXO	IYYO	IZZO	XBAR	YBAR	ZBAR
195.000	11.675	11.675	6.060	5.000	5.000	11.538

IXX	IYY	IZZ	IXY	IXZ	IYZ
6.76265	6.76265	1.95313	0.00000	0.00000	.00000

Note: No principal axes definition is included since they will lie on axes of symmetry.

EXAMPLE PROBLEM 3
A COMPOSITE STRUCTURE
($\frac{1}{8}$ SCALE)



The computer output relative to example problem 3 follows and includes, the data input. The computer calculated component data, the summed data and lastly, the inertias about the system principal axes and the individual axis locations.

INPUT DATA LISTED BELOW

ITEM	DESCRIPTION	SHAPE	RHO	A	B	C	D	F	XI	YI	ZI	XJ	YJ	ZJ	XK	YK	ZK
1	ANTENNA	7	.15000000E+01	0.000	.500	0.000	0.000	0.000	35.000	-5.000	0.000	35.000	0.000	0.000	0.000	0.000	0.000
2	DIELECTRIC BRKT	11	.15000000E+00	1.500	0.400	-2.400	0.000	3.600	21.000	.640	0.000	35.000	.640	0.000	23.000	3.000	0.000
3	BODY CYL	2	.10000000E+00	4.000	3.800	0.000	0.000	0.000	25.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
4	FINS	12	.66000000E-01	.390	10.000	-1.300	4.000	5.000	4.330	-14.000	0.000	4.330	14.000	0.000	10.000	10.000	0.000
5	FLARED ADAPTER	3	.10000000E+00	6.000	5.000	4.030	3.000	0.000	-8.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
6	CYL SEG VOID	13	-.10000000E+00	3.900	.200	1.000	0.000	0.000	11.000	-3.290	0.000	21.000	-3.290	0.000	0.000	0.000	0.000
7	WIRING ETC VOL	2	.50000000E+01	3.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	25.000	0.000	0.000	0.000	0.000	0.000

COMPONENT DATA LISTED BELOW

ITEM	DESCRIPTION	WT	IXXCO	IYYCO	IZZCO	XCGCO	YCGCO	ZCGCO	IXYCO	IYZCO	IXZCO
1	ANTENNA	1.50000	.00537	.00451	.00537	35.00000	-2.62146	0.00000	0.00000	0.00000	0.00000
2	DIELECTRIC BRKT	16.23300	.01442	.04039	.05349	28.20300	.64000	0.00000	-.00795	0.00000	0.00000
3	BODY CYL	12.25221	.04026	.15790	.15790	12.50300	0.00000	0.00000	0.00000	0.00000	0.00000
4	FINS	3.76200	.06444	.00439	.06881	4.33300	0.00000	0.00000	0.00000	0.00000	0.00000
5	FLARED ADAPTER	14.71107	.07771	.05261	.05261	-5.07517	0.00000	0.00000	0.00000	0.00000	0.00000
6	CYL SEG VOID	-1.56000	-.00150	-.00420	-.00291	16.00300	-3.29000	0.00000	0.00000	0.00000	0.00000
7	WIRING ETC VOL	5.30900	.03779	.06012	.06012	12.50300	0.00000	0.00000	0.00000	0.00000	0.00000

SYSTEM DATA LISTED BELOW (WT=LBS, INERTIAS=SLUGS FT SQUARED, C.G.=INS, SECOND MOMENT=SLUG FT SQUARED)

SYS WT	IXXO	IYYO	IZZO	XBAR	YBAR	ZBAR
51.865	.209	4.091	4.166	12.372	.223	0.000
IXX	**IYY**	**IZZ**	**IXY**	**IXZ**	**IYZ**	
.20797	2.37764	2.45180	-.02817	0.00000	0.00000	

INERTIAS (EIGENVALUES) ABOUT SYSTEM PRINCIPAL AXES WITH AXIS DIRECTION COSINES (EIGENVECTORS) RELATING THE PRINCIPAL AXES TO THE X, Y, AND Z SYSTEM AXES IN THAT SEQUENCE

EIGENVALUE(1) = .20760E+00

EIGENVECTOR(1)

.999916E+00 .129820E-01 0.

EIGENVALUE(2) = .23760E+01

EIGENVECTOR(2)

-.129820E-01 .999916E+00 0.

EIGENVALUE(3) = .24518E+01

EIGENVECTOR(3)

0. 0. .100000E+01

CONVERSION TO INTERNATIONAL SYSTEM OF UNITS

CONVERT FROM	TO	MULTIPLY BY
INCHES	METERS	.025 400
$INCHES^2$	$METERS^2$	.000 645 160
FEET	METERS	.304 800
$FEET^2$	$METERS^2$	.092 903 040
POUNDS	KILOGRAMS	.453 592 370
$POUNDS/INCHES^3$	$KILOGRAM/METER^3$	27 679.905 ....
$POUNDS/FEET^3$	$KILOGRAM/METER^3$	16.018 463 ....
SLUG	KILOGRAMS	14.593 9029 ....
SLUG $FEET^2$	KILOGRAM $METER^2$	1.355 8179 ....

CONCLUSIONS

A computer program for determining mass properties of a rigid structure is presented. The structure is broken down into preselected shapes with known properties, and input data are supplied to completely describe each shape. For complicated structures skewed in space, this program offers a practical solution to a tedious and time-consuming task. It is also practical to use this program for problems that involve repetitious or lengthy calculations.

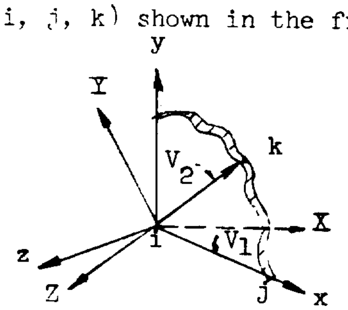
REFERENCES

1. Thomson, W. T.: Introduction to Space Dynamics, 1961.
2. Meyers, F. T.: Weight Handbook, Volume 1, 1944.
3. Pletta, D. H., and Frederick, D.: Engineering Mechanics, 1963.
4. Wada, B.: Stiffness Matrix Structural Analysis, TN-32-774 (JPL).
5. Greenwood, D. T.: Principles of Dynamics, 1965.

APPENDIX A

DIRECTION COSINE DERIVATION

The direction cosines are used to transform the properties from the component coordinate system x, y, z to system coordinates X, Y, Z. Three points (i, j, k) shown in the figure below, define the vectors $\vec{V}_1$ and $\vec{V}_2$



Vector $\vec{V}_1$ is arbitrarily selected to be coincident with the "x" axis. A unit vector on this axis can be written

$$\vec{1}_x = \frac{\vec{V}_1}{|\vec{V}_1|}$$

Taking the vector cross product of $\vec{V}_1$ with $\vec{V}_2$ and dividing by the resulting magnitude gives a unit vector on the z axis

$$\vec{1}_z = \frac{\vec{V}_1 \times \vec{V}_2}{|V_z|}$$

Similarily, a unit vector on the "y" axis is found from

$$\vec{1}_y = \vec{1}_z \times \vec{1}_x$$

The direction cosines are the X, Y, Z, components of the unit vectors on the x, y, z axes.

The direction cosine for the x axis is written as

LX = (XJ -XI)/LGTH
MX = (YJ -YI)/LGTH
NX = (ZJ -ZI)/LGTH

where the length is

$$LGTH = ((XJ - XI)^2 + (YJ - YT)^2 + (ZJ - ZI)^2)^{1/2}$$

The vector $\vec{V}_2$ can be written

T1 = XK -XI
T2 = YK -YI
T3 = ZK -ZI

A vector on the z axis is found by taking the vector cross product of $\vec{1}_x$ and $\vec{V}_2$

LZ = MX*T3-T2*NX
MZ = NX*T1-T3*LX
NZ = T2*LX-T1*MX

The length is

$$T4 = (LZ^2 + MZ^2 + NZ^2)^{1/2}$$

Normalizing to get a unit vector

LZ→LZ/T4; MZ→MZ/T4; NZ→NZ/T4

The unit vector on the y axis has a magnitude of one and is determined by the vector cross product $\vec{1}_z$ and $\vec{1}_x$

LY = MZ*NX-MX*NZ.
MY = NZ*LX-NX*LZ
NY = MX*LZ-LX*MZ

Writing the nine terms in matrix form, we get

$$[DC]=\begin{bmatrix} LX & MX & NX \\ LY & MY & NY \\ LZ & MZ & NZ \end{bmatrix}$$

APPENDIX B

```
      PROGRAM NAME       (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C     LATEST DIRCOS IN MAIN DECK AND 13 SHAPES 12/20/72
      DIMENSION  ITEM(200),SHAPE(200),RHO(200),A(200),B(200),C(200),
     1 D(200),F(200),        XI(200),YI(200),ZI(200),XJ(200),YJ(200),ZJ(2
     200),XK(200),YK(200),ZK(200),IXXCG(200),IYYCG(200),IZZCG(200),IXYCG
     3(200),IYZCG(200),IXZCG(200),XCG(200),YCG(200),ZCG(200),XL(200),YL(
     4200),ZL(200),IXXCO(200),IYYCO(200),IZZCO(200),IXYCO(200),IXZCO(200
     5),IYZCO(200),DES(3,200),W(200),O(200),P(200),ARR(3,3),E(3),CRR(3)
      REAL IXX,IYY,IZZ,IXXCG,IYYCG,IZZCG,IXYCG,IXZCG,
     1IYZCG,LX,MX,NX,LY,MY,NY,LZ,MZ,NZ,IXY,IXZ,IYZ,LGTH,IXXCO,IYYCO,
     2IZZCO,IXYCO,IXZCO,IYZCO,IXXO,IYYO,IZZO,KI,KII,KIII,IXYK,IXYCGP
      INTEGER SHAPE
      I=0
 1010 I=I+1
  105 FORMAT(I5)
      READ(5,101) ITEM(I),DES(1,I),DES(2,I),SHAPE(I),RHO(I),A(I),B(I),
     1C(I),D(I),F(I),      XI(I),YI(I),ZI(I),XJ(I),YJ(I),ZJ(I),XK(I),
     2YK(I),ZK(I)
  101 FORMAT (I3,2A9,I2,F9.4,5F8.3/9F8.3)
      IF(EOF(5))1009,1010
 1009 IMAX=I-1
      WRITE(6,271)
  271 FORMAT(1H0* INPUT DATA LISTED BELOW*//)
      WRITE(6,103)(ITEM(I),DES(1,I),DES(2,I),SHAPE(I),RHO(I),A(I),B(I),
     1C(I),D(I),F(I),      XI(I),YI(I),ZI(I),XJ(I),YJ(I),ZJ(I),XK(I),
     2YK(I),ZK(I),I=1,IMAX)
  103 FORMAT (2X*ITEM            DESCRIPTION          SHAPE          RHO
     1        A                  B                  C                 D                 E
     2  */I5,6X,2A9,I5,1X,F16.8,5F15.3/*            XI                YI
     3       ZI                XJ                YJ                ZJ                XK
     4            YK               ZK*/9F14.3//)
      WRITE(6,261)
  261 FORMAT(3X9HCOMPONENT2X4HDATA2X6HLISTED2X5HBELOW//)
      WRITE(6,250)
  250 FORMAT(1H0* ITEM          DESCRIPTION              WT            IXXCO
     1 IYYCO        IZZCO        XCGCO         YCGCO        ZCGCO        IXYCO        IYZCO
     2  IXZCO*//)
      TW=0.
      XMOM=0.
      YMOM=0.
      ZMOM=0.
```

```
      IXX=0.
      IYY=0.
      IZZ=0.
      IXY=0.
      IXZ=0.
      IYZ=0.
      IXXO=0.
      IYYO=0.
      IZZO=0.
      PI=3.141593
      CONS=4632.
      DO50 I=1,IMAX
      XL(I)=0.
      YL(I)=0.
      ZL(I)=0.
      IXYCG(I)=0.
      IXZCG(I)=0.
      IYZCG(I)=0.
      LGTH=SQRT((XJ(I)-XI(I))**2+(YJ(I)-YI(I))**2+(ZJ(I)-ZI(I))**2)
      ITEMP=SHAPE(I)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13),ITEMP

C     DISCRETE MASS    *SHAPE 1*
C     POINT  J MUST BE SELECTED ANYWHERE ON THE X AXIS FOR DIR. COSINES

    1 W(I)=RHO(I)
      IXXCG(I)=A(I)*CONS
      IYYCG(I)=B(I)*CONS
      IZZCG(I)=C(I)*CONS
      IF((ABS(XJ(I)).NE. 0.).OR.(ABS(YJ(I)).NE.0.).OR. (ABS(ZJ(I)).NE.0.
     1)) GO TO 704
      LGTH=10.
      XJ(I)=1.1*XI(I)+10.
      IF(XJ(I).EQ.XI(I))XJ(I)=XI(I)+10.
  704 XL(I)=0.0
      GO TO 60

C     CYLINDER    *SHAPE 2*

    2 IF(B(I).EQ.A(I))GO TO 33
      IF(B(I).LT.0.)B(I)=0.
      IF(C(I).LT.0.)C(I)=0.
      VOLCYL =PI*(A(I)**2-B(I)**2)*(LGTH-C(I))
      W(I)=RHO(I)*VOLCYL
      IF(RHO(I).GT.0.4) W(I)=RHO(I)
      RHO(I)=W(I)/VOLCYL
      IXXCG(I)=(.5*W(I)*(A(I)**2+B(I)**2))
      IYYCG(I)=.25*W(I)*(A(I)**2+B(I)**2)+W(I)*(LGTH**3-C(I)**3)
     $/(12.*(LGTH-C(I)))
      IZZCG(I)=IYYCG(I)
      XL(I)=LGTH/2.
      GO TO 60
```

```
      CYLINDER (THIN WALL)    *SHAPE 2*

   33 W(I)=RHO(I)
      XL(I)=LGTH/2.
      IXXCG(I)=(RHO(I)*A(I)**2)
      IYYCG(I)=.5*RHO(I)*A(I)**2+RHO(I)*(LGTH**3-C(I)**3)/
     2((LGTH-C(I))*12.0)
      IZZCG(I)=IYYCG(I)
      GO TO 60

      TRUNCATED CONE    *SHAPE 3* (REF--R. HULL)

    3 IF(B(I).EQ.A(I))GO TO 30
      VOL =
     $1.0472*LGTH*(A(I)**2+A(I)*C(I)+C(I)**2-B(I)**2-B(I)*D(I)-D(I)**2)
      W(I) = RHO(I)*VOL
      IF(RHO(I).GT.0.4) W(I)=RHO(I)
      RHO(I) = W(I)/VOL
      XL(I)= (LGTH**2)*(A(I)**2+2.*A(I)*C(I)+3.*C(I)**2-B(I)**2-2.*B(I)*
     $D(I)-3.*D(I)**2)*.2618/VOL
      KI =.05*3.141593*RHO(I)*((C(I)**5-A(I)**5)*LGTH/(C(I)-A(I))-
     $(D(I)**5-B(I)**5)*LGTH/(D(I)-B(I)))
      KII=3.1416*RHO(I)*LGTH**3*((A(I)**2-B(I)**2)/3.0-.5*((A(I)-
     $C(I))*A(I)-(B(I)-D(I))*B(I))+.2*((A(I)-C(I))**2-(B(I)-D(I))**2))
      KIII=1.0472*(LGTH)*(A(I)**2+A(I)*C(I)+C(I)**2-B(I)**2-B(I)*D(I)-
     $D(I)**2)*RHO(I)*XL(I)**2
      IXXCG(I)=2.0*(KI)
      IYYCG(I)=KI+KII-KIII
      IZZCG(I)=IYYCG(I)
      GO TO 60

      TRUNCATED CONE (THIN WALL)    *SHAPE 3*

   30 VOL=F(I)*SQRT((A(I)-C(I))**2+LGTH**2)*3.14159*(A(I)+C(I))
      W(I) = VOL*RHO(I)
      XL(I)=LGTH/3.*((2.*C(I)+A(I))/(C(I)+A(I)))
      IYYCG(I)=(RHO(I)/4.*(A(I)**2+C(I)**2)+RHO(I)*LGTH**2/18.*(1.+2.*A(
     1I)*C(I)/(A(I)+C(I))**2))
      IZZCG(I)=IYYCG(I)
      IXXCG(I)=(RHO(I)/2.*(A(I)**2+C(I)**2))
      GO TO 60

      TORUS    *SHAPE 4*

    4 A(I)=A(I)-LGTH
      IF(D(I).LT.0.)D(I)=0.
      VOL1=2.*PI  **2*LGTH**2*A(I)
      VOL2=2.*PI  **2*D(I)**2*A(I)
      ACVOL=VOL1-VOL2
      W(I)=RHO(I)*ACVOL
      XM1=(RHO(I)*VOL1)
```

```
      XI1=.125*XM1*(4.*A(I)**2+5.*LGTH**2)
      XM2=(RHO(I)*VOL2)
      XI2=.125*XM2*(4.*A(I)**2+5.*D(I)**2)
      IYYCG(I)=(XI1-XI2)
      IZZCG(I)=IYYCG(I)
      YI1=.25*XM1*(4.*A(I)**2+3.*LGTH**2)
      YI2=.25*XM2*(4.*A(I)**2+3.*D(I)**2)
      IXXCG(I)=YI1-YI2
      GO TO 60

C     BEAM (ARBITRARY CROSS SECTION)   *SHAPE 5*

    5 XL(I)=LGTH/2.
      IXXCG(I)=(RHO(I)*LGTH*(B(I)+C(I)))
      IYYCG(I)=(RHO(I)*(B(I)*LGTH+.0833*A(I)*LGTH**3))
      IZZCG(I)=(RHO(I)*(C(I)*LGTH+.0833*A(I)*LGTH**3))
      VOL=A(I)*LGTH
      W(I)=RHO(I)*VOL
      GO TO 60

C     CIRCULAR ROD (ARBITRARY CROSS SECTION)   *SHAPE 6*

    6 W(I)=RHO(I)
      RSCG=(A(I)**2*C(I)+B(I))/(A(I)*C(I))
      IXXCG(I)=RSCG**2*W(I)
      IYYCG(I)=.5*RSCG**2*W(I)
      IZZCG(I)=IYYCG(I)
      GO TO 60

C     SPHERICAL SEGMENT   *SHAPE 7*

    7 F(I)=C(I)-B(I)
      D(I)=C(I)-LGTH
      G   =LGTH-B(I)
      VOL1=1.0472*LGTH**2*(3.*C(I)-LGTH)
      VOL2=1.0472*G   **2*(3.*F(I)-G   )
      ACVOL=VOL1-VOL2
      W(I)=ACVOL*RHO(I)
      IF(RHO(I).GT.0.4) W(I)=RHO(I)
      RHO(I) = W(I)/ACVOL
      XBAR1=.75*(2.*C(I)-LGTH)**2/(3.*C(I)-LGTH)
      XBAR2=.75*(2.*F(I)-G   )**2/(3.*F(I)-G   )
      XLT  =(XBAR1*VOL1-XBAR2*VOL2)/ACVOL
      XL(I)=XLT   -D(I)
      XM1=VOL1*RHO(I)
      XM2=VOL2*RHO(I)
      TM=XM1-XM2
      XI)=(2.*LGTH*XM1/(3.*C(I)-LGTH))*(C(I)**2-.75*C(I)*LGTH+.15*LGTH
     1**2)
```

```
      XI2=(2.*G    *XM2/(3.*F(I)-G    ))*(F(I)**2-.75*F(I)*G    +.15*G
     1**2)
      IXXCG(I)=(XI1-XI2)
      TEMP1=.05236*RHO(I)*(15.*C(I)**4*C(I)-10.*C(I)**2*C(I)**3+3.*C(I)
     1**5)+.20944*RHO(I)*(5.*C(I)**2*C(I)**3-3.*C(I)**5)
      TEMP2=.05236*RHO(I)*(15.*C(I)**4*D(I)-10.*C(I)**2*D(I)**3+3.*D(I)
     1**5)+.20944*RHO(I)*(5.*C(I)**2*D(I)**3-3.*D(I)**5)
      TEMP3=.05236*RHO(I)*(15.*F(I)**4*F(I)-10.*F(I)**2*F(I)**3+3.*F(I)
     1**5)+.20944*RHO(I)*(5.*F(I)**2*F(I)**3-3.*F(I)**5)
      TEMP4=.05236*RHO(I)*(15.*F(I)**4*D(I)-10.*F(I)**2*D(I)**3+3.*D(I)
     1**5)+.20944*RHO(I)*(5.*F(I)**2*D(I)**3-3.*D(I)**5)
      ACTEMP=((TEMP1-TEMP2)-(TEMP3-TEMP4))-(TM*XLT    **2)
      IYYCG(I)=ACTEMP
      IZZCG(I)=IYYCG(I)
      GO TO 60
C
C     SPHERE    *SHAPE 8*
C
    8 LGTH=A(I)
      XJ(I)=XI(I)+LGTH
      YJ(I)=YI(I)
      ZJ(I)=ZI(I)
      XJ(I) = XI(I)+10.
      IF(B(I).EQ.A(I))GO TO 34
      VOL1=4.188791*A(I)**3
      VOL2=4.188791*B(I)**3
      ACVOL=VOL1-VOL2
      W(I)=RHO(I)*ACVOL
      XM1=(RHO(I)*VOL1)
      XM2=(RHO(I)*VOL2)
      XI1=(.4*XM1*LGTH**2)
      XI2=(.4*XM2*B(I)**2)
      IXXCG(I)=(XI1-XI2)
      IYYCG(I)=IXXCG(I)
      IZZCG(I)=IXXCG(I)
      GO TO 60
C     SPHERE (THIN WALL)    *SHAPE 8*
C
   34 W(I)=RHO(I)
      IXXCG(I)=(.667*RHO(I)*A(I)**2)
      IYYCG(I)=IXXCG(I)
      IZZCG(I)=IXXCG(I)
      XL(I)=0.0
      GO TO 60
C     HEMISPHERE    *SHAPE 9*
C
    9 IF(LGTH.EQ.A(I))GO TO 35
      IF(A(I).LT.0.)A(I)=0.
      VOL1=2.09439*LGTH**3
      VOL2=2.09439*A(I)**3
```

```
      ACVOL=VOL1-VOL2
      W(I)=RHO(I)*ACVOL
      XBAR1=.375*LGTH
      XBAR2=.375*A(I)
      XL(I)=(XBAR1*VOL1-XBAR2*VOL2)/ACVOL
      XM1=(RHO(I)*VOL1)
      XM2=(RHO(I)*VOL2)
      XI1=(.4*XM1*LGTH**2)
      XI2=(.4*XM2*A(I)**2)
      IXXCG(I)=(XI1-XI2)
      XI3=(.26*XM1*LGTH**2)
      XI4=(.26*XM2*A(I)**2)
      IYYCG(I)=(XI3-XI4)
      IZZCG(I)=IYYCG(I)
      GO TO 60

C     HEMISPHERE (THIN WALL)   *SHAPE 9*

   35 W(I)=RHO(I)
      XL(I)=LGTH/2.
      IXXCG(I)=.666*RHO(I)*LGTH**2
      IYYCG(I)=.4166*RHO(I)*LGTH**2
      IZZCG(I)=IYYCG(I)
      GO TO 60

C     PARALLFLEPIPED  *SHAPE 10*

   10 IF(D(I).EQ.A(I))GO TO 36
      IF(D(I).LT.0.)D(I)=0.
      VOL1=LGTH*B(I)*A(I)
      VOL2=C(I)*F(I)*D(I)
      ACVOL=VOL1-VOL2
      W(I)=RHO(I)*ACVOL
      IF(RHO(I).GT.0.0) W(I)=RHO(I)
      RHO(I) = W(I)/ACVOL
      XL(I)=LGTH/2.
      XM1=VOL1*RHO(I)
      XM2=VOL2*RHO(I)
      XI1=(.083333*XM1*(B(I)**2+A(I)**2))
      XI2=(.083333*XM2*(F(I)**2+D(I)**2))
      IXXCG(I)=(XI1-XI2)
      XI3=(.083333*XM1*(LGTH**2+A(I)**2))
      XI4=(.083333*XM2*(C(I)**2+D(I)**2))
      IYYCG(I)=(XI3-XI4)
      XI5=(.083333*XM1*(LGTH**2+B(I)**2))
      XI6=(.083333*XM2*(C(I)**2+F(I)**2))
      IZZCG(I)=(XI5-XI6)
      GO TO 60
```

```
C     PARALLELEPIPED (THIN WALL)  *SHAPE 10*

   36 XL(I)=LGTH/2.
      W(I)=RHO(I)
      TEMP1=(LGTH*B(I)*A(I))
      TEMP2=(LGTH*B(I)+B(I)*A(I)+LGTH*A(I))
      IXXCG(I)=(.083333*RHO(I)*(B(I)**2+A(I)**2)+(RHO(I)/6.)*(TEMP1*
     1(B(I)+A(I))/TEMP2))
      IYYCG(I)=(.083333*RHO(I)*(LGTH**2+A(I)**2)+(RHO(I)/6.)*(TEMP1*
     1(LGTH+A(I))/TEMP2))
      IZZCG(I)=(.083333*RHO(I)*(LGTH**2+B(I)**2)+(RHO(I)/6.)*(TEMP1*
     1(LGTH+B(I))/TEMP2))
      GO TO 60

C     SWEPT TRAPEZOIDAL PANEL (THICK WALL) *SHAPE 11*  (REF--R. HULL)

   11 W(I)=A(I)*RHO(I)*(LGTH*(B(I)+F(I))/2.)
      IF(RHO(I).GT.0.4) W(I)=RHO(I)
      RHO(I)=(W(I)) /(((B(I)+F(I))/2.0)*LGTH*A(I))
      XL(I)=LGTH*(B(I)+2.0*F(I))/(3.0*(B(I)+F(I)))
      XT = F(I)*LGTH/(B(I)-F(I))
      TNTAU = C(I)/LGTH
      FETAN = (F(I)/2.- B(I)/2.+ C(I))/LGTH
      AFTAN = (C(I)-F(I)/2.0+B(I)/2.0)/LGTH
      THRB=LGTH+XT
      BCG=((FETAN+AFTAN)/2.)*(THRB-XL(I))
      XPAN=AFTAN-FETAN
      IYYCG(I)= RHO(I)*((XPAN*A(I)**3*(THRB**2-XT**2))/24.+  XPAN*A(I)*
     2(THRB**4-XT**4)/4.)-((THRB-XL(I))**2)*W(I)
      XFRXX=   XPAN*A(I)* (BCG**2*(THRB**2-XT**2)/2.-BCG*(FETAN+AFTAN)*
     1(THRB**3-XT**3)/3.+(FETAN+AFTAN)**2*(THRB**4-XT**4)/16.)
      IXXCG(I)= RHO(I)* (XPAN**3 *A(I)*(THRB**4-XT**4)/48. +XPAN*A(I)**3
     1* (THRB**2-XT**2)/24. +XFRXX)
      IZZCG(I)=A(I)*RHO(I)*(XPAN**3*(THRB**4-XT**4)/48.+XPAN*(((THRB-XL(
     3I))**2+BCG**2)*(THRB**2-XT**2)/2.-(2.*(THRB-XL(I))+(FETAN+AFTAN)*B
     4CG)*(THRB**3-XT**3)/3.+(4.+(FETAN+AFTAN)**2)*(THRB**4-XT**4)/16.))
      IF (ABS(C(I)).LT. .001) GO TO 60
      APXCG=B(I)/2.+ABS(TNTAU*XL(I))
      H=APXCG+SQRT((XK(I)-XI(I))**2+(YK(I)-YI(I))**2+(ZK(I)-ZI(I))**2)
      CGK=H-APXCG
      PRB=THRB-LGTH
      SMH=H*PRB/THRB
      PRIXYK=.5*F(I)*(2.*SMH-F(I)) *(LGTH*PRB/3.+PRB**2/12.)
      IXYK=THRB**2*(H**2-(H-B(I))**2)/24.-PRIYYK
```

```
      IXYCGP = ABS(IXYK*A(I)*RHO(I) -W(I)*CGK*XL(I))*C(I)/ABS(C(I))
      PAXRC=.7854*ABS(C(I))/C(I)
      IF(ABS(IXXCG(I)-IYYCG(I)).LT. .001) GO TO 20
      PAXRC= ATAN((2.0*IXYCGP)/(IYYCG(I)-IXXCG(I)))/2.0
       IF(IXXCG(I).GT.IYYCG(I)) PAXRC =.5*(ABS(2.*PAXRC)-3.1416)*
     1 PAXRC/ABS(PAXRC)
   20  TMPIXX =IXXCG(I)*COS(PAXRC)**2+IYYCG(I)*SIN(PAXRC)**2-
     12.0*IXYCGP*SIN(PAXRC)*COS(PAXRC)
      IYYCG(I)= IXXCG(I)*SIN(PAXRC)**2+ IYYCG(I)*COS(PAXRC)**2 +
     2(2.*IXYCGP*SIN(PAXRC)*COS(PAXRC) )
      IXXCG(I)=TMPIXX
   21 SHFTI=ABS(TAN(PAXRC))*XL(I)
      XJ(I)= (XI(I)*(LGTH-XL(I))+XJ(I)*XL(I))/LGTH
      YJ(I)= (YI(I)*(LGTH-XL(I))+YJ(I)*XL(I))/LGTH
      ZJ(I)= (ZI(I)*(LGTH-XL(I))+ZJ(I)*XL(I))/LGTH
      XI(I)=(XI(I)*(SHFTI+CGK) -XK(I)*SHFTI)/CGK
      YI(I)=(YI(I)*(SHFTI+CGK) -YK(I)*SHFTI)/CGK
      ZI(I)=(ZI(I)*(SHFTI+CGK) -ZK(I)*SHFTI)/CGK
      LGTH=SQRT((XJ(I)-XI(I))**2+(YJ(I)-YI(I))**2+(ZJ(I)-ZI(I))**2)
      XL(I) = LGTH
      GO TO 60

C     SYMMETRIC TRAPEZOIDAL PANELS (THICK WALL)*SHAPE 12* (REF--R. HULL)

   12 LGTH = (LGTH - 2.0* D(I))/2.0
      VOL=(LGTH*(B(I)+F(I))    )*A(I)
      W(I)= RHO(I)*VOL
      IF(RHO(I).GT.0.4) W(I)=RHO(I)
      RHO(I)=(W(I)) /(((B(I)+F(I))     )*LGTH*A (I))
      XL(I)=LGTH*(B(I)+2.0*F(I))/(3.0*(B(I)+F(I)))
      IYYCG(I) = (LGTH**3)*  (F(I)**2+4.*F(I)*B(I)+B(I)**2)*A(I)*RHO(I)/
     2((36.*(F(I)+B(I)))*.5)+(A(I)**2/12.+(D(I)+XL(I))**2)*W(I)
      XT = F(I)*LGTH/(B(I)-F(I))
      IF(B(I).LT.F(I)) XT=B(I)*LGTH/(B(I)-F(I))
      TNTAU = C(I)/LGTH
      FETAN = (F(I)/2.- B(I)/2.+ C(I))/LGTH
      AFTAN = (C(I)-F(I)/2.0+B(I)/2.0)/LGTH
      IF(B(I).LT.F(I)) LGTH = 0.0
      KI=ABS(RHO(I)*A(I)*(FETAN**3-AFTAN**3)/3.0)
      KII = (B(I)*A(I)*(RHO(I))/(LGTH+XT))*(TNTAU*(LGTH+XT-XL(I)))**2
      KII=ABS(KII)
      IF(B(I).LT.F(I)) LGTH = (1.0-F(I)/B(I))*XT
      XTLGTH = LGTH + ABS(XT)
      IXXCG(I)=(KI*((XTLGTH )**4-(XT)**4)/2.0)-KII*((XTLGTH )**2-XT**2)
     1+ A(I)**2*W(I)/12.
      IZZCG(I) = IXXCG(I) + IYYCG(I)-[illegible]2*[illegible]6.0
      XL(I) = LGTH + D(I)
      LGTH=SQRT((XJ(I)-XI(I))**2+(YJ(I)-YI(I))**2+(ZJ(I)-ZI(I))**2)
      GO TO [illegible]
```

```
      CURVED THIN WALL PANEL *SHAPE 13*

   13 W(I)= 2.*LGTH*A(I)*B(I) *C(I) *RHO(I)
      IF(RHO(I).GT.0.4) W(I)=RHO(I)
      RHO(I) = W(I)/(2.*LGTH*A(I)*B(I)*C(I))
      KI=2.*C(I)*A(I)*B(I)*LGTH*RHO(I)
      KII=LGTH**2/12.
      XL(I)=LGTH/2.
      IXXCG(I)=LGTH
     $          *RHO(I)*A(I)**3*B(I)*(2.*C(I)-(2.*SIN(C(I))**2)/C(I))
      IYYCG(I)=KI
     $        *(A(I)**2*(C(I)-SIN(C(I))*COS(C(I))))/(2.*C(I)) +KII)
      IZZCG(I)=KI
     $        * (A(I)**2*(C(I)+SIN(C(I))*COS(C(I))-2.*SIN(C(I))**2/
     1C(I))/(2.*C(I))+KII)
      GO TO 60

      BEGIN DIRCOS

   60 IF((ABS(XK(I)).NE. 0.).OR.(ABS(YK(I)).NE.0.).OR. (ABS(ZK(I)).NE.0.
     1)) GO TO 90
      XK(I)=XI(I)-(YJ(I)-YI(I))
      YK(I)=YI(I)+(XJ(I)-XI(I))
      ZK(I)=ZI(I)
      IF((YJ(I).NE.YI(I)).OR.(XJ(I).NE.XI(I)))GO TO 90
      YK(I)=LGTH+YI(I)
   90 LX=(XJ(I)-XI(I))/LGTH
      MX=(YJ(I)-YI(I))/LGTH
      NX=(ZJ(I)-ZI(I))/LGTH
      T1= XK(I)-XI(I)
      T2= YK(I)-YI(I)
      T3= ZK(I)-ZI(I)
      LZ= MX*T3-T2*NX
      MZ=NX*T1-T3*LX
      NZ=T2*LX-T1*MX
      T4= SQRT(LZ**2+ MZ**2 +NZ**2)
      LZ=LZ/T4$ MZ=MZ/T4$ NZ=NZ/T4
      LY= MZ*NX-MX*NZ
      MY=NZ*LX-NX*LZ
      NY=MX*LZ-LX*MZ
      END DIRCOS
```

```
C      ROTATE COMPONENT MOMENT OF INERTIA TO SYSTEM COORDINATES

       IXXCO(I)=(IXXCG(I)*(LX)**2+IYYCG(I)*(LY)**2+IZZCG(I)*(LZ)**2)/CONS
       IYYCO(I)=(IXXCG(I)*(MX)**2+IYYCG(I)*(MY)**2+IZZCG(I)*(MZ)**2)/CONS
       IZZCO(I)=(IXXCG(I)*(NX)**2+IYYCG(I)*(NY)**2+IZZCG(I)*(NZ)**2)/CONS
       IXYCO(I)= ((IXXCG(I)*(LX)*(MX)+IYYCG(I)*(LY)*(MY)+IZZCG(I)*(LZ)*(M
      1Z)))/CONS
       IXZCO(I)= ((IXXCG(I)*(LX)*(NX)+IYYCG(I)*(LY)*(NY)+IZZCG(I)*(LZ)*(N
      1Z)))/CONS
       IYZCO(I)= ((IXXCG(I)*(MX)*(NX)+IYYCG(I)*(MY)*(NY)+IZZCG(I)*(MZ)*(N
      1Z)))/CONS

C      CALCULATE COMPONENT CENTER OF MASS COORDINATES AND WRITE OUT

       XCG(I)=XI(I)+XL(I)*LX
       YCG(I)=YI(I)+XL(I)*MX
       ZCG(I)=ZI(I)+XL(I)*NX
       WRITE(6,300)ITEM(I),DES(1,I),DES(2,I),W(I),IXXCO(I),IYYCO(I),IZZCO
      1(I),XCG(I),YCG(I),ZCG(I),IXYCO(I),IYZCO(I),IXZCO(I)
  300 FORMAT(I5,5X,2A9,6F11.5,4F9.5/)

C      CALCULATE SYSTEM WEIGHT, SECOND MOMENT AT ORGIN AND C.G.

       TW=TW+W(I)
       IXXO=IXXO+IXXCO(I)+W(I)*(ZCG(I)**2+YCG(I)**2)/CONS
       IYYO=IYYO+IYYCO(I)+W(I)*(ZCG(I)**2+XCG(I)**2)/CONS
       IZZO=IZZO+IZZCO(I)+W(I)*(XCG(I)**2+YCG(I)**2)/CONS
       XMOM=XMOM+W(I)*XCG(I)
       YMOM=YMOM+W(I)*YCG(I)
   50  ZMOM=ZMOM+W(I)*ZCG(I)

C      COMPUTE SYSTEM C.G. COORDINATES

       XBAR=XMOM/TW
       YBAR=YMOM/TW
       ZBAR=ZMOM/TW
       WRITE(6,262)
  262 FORMAT(1HC* SYSTEM DATA LISTED BELOW (WT=LBS, INERTIAS=SLUGS FT SQ
      1UARED, C.G.=INS, SECOND MOMENT=SLUG FT SQUARED)*//)
       WRITE(6,100)TW,IXXO,IYYO,IZZO,XBAR,YBAR,ZBAR
  100 FORMAT(11X3HSYS1X2HWT12X4HIXXO14X4HIYYO14X4HIZZO16X4HXBAR14X4HYBAR
      114X4HZBAR/7F18.3)
```

```
      TRANSFER MASS PROPERTIES TO SYSTEM C.G., SUM AND WRITE OUT

      DO70 I=1,IMAX
      DELX=XCG(I)-XBAR
      DELY=YCG(I)-YBAR
      DELZ=ZCG(I)-ZBAR
      IXX=IXX+IXXCO(I)+W(I)*(DELY**2+DELZ**2) /CONS
      IYY=IYY+IYYCO(I)+W(I)*(DELX**2+DELZ**2) /CONS
      IZZ= IZZ+IZZCO(I)+W(I)*(DELY**2+DELX**2)/CONS
      IXY=IXY+IXYCO(I)+W(I)*DELX*DELY/CONS
      IXZ=IXZ+IXZCO(I)+W(I)*DELX*DELZ/CONS
   70 IYZ=IYZ+IYZCO(I)+W(I)*DELY*DELZ/CONS
      WRITE(6,200)IXX,IYY,IZZ,IXY,IXZ,IYZ
  200 FORMAT(12X3HIXX16X3HIYY15X3HIZZ15X3HIXY15X3HIXZ15X3HIYZ/6F18.5/)

      COMPUTE INERTIAS (EIGENVALUES) ABOUT PRINCIPAL AXES AND EACH AXIS
      DIRECTION COSINES (EIGENVECTORS) AND WRITE OUT


      PRINT 340
  340 FORMAT(/1X,*INERTIAS (EIGENVALUES) ABOUT SYSTEM PRINCIPAL AXES WIT
     1H AXIS DIRECTION COSINES (EIGENVECTORS) RELATING THE PRINCIPAL AXE
     1S*/1X,*TO THE X, Y, AND Z SYSTEM AXES IN THAT SEQUENCE*///)
      MAX=3
      N=3
      ARR(1,1)=IXX
      ARR(2,1)=ARR(1,2)=IXY
      ARR(1,3)=ARR(3,1)=IXZ
      ARR(2,2)=IYY
      ARR(3,2)=ARR(2,3)=IYZ
      ARR(3,3)=IZZ
      CALL SYMQL(MAX,N,ARR,E,CRR,IERR)
      IF(IERR .NE. 0) GO TO 332
      DO 336 J=1,3
      PRINT 337,J,E(J)
  337 FORMAT(1X,*EIGENVALUE(*I1*) = *F12.5//)
      PRINT 339,J
  339 FORMAT(1X,*EIGENVECTOR(*I1*)*/)
      PRINT 338,(ARR(I,J),I=1,3)
  338 FORMAT(1X,3(F14.6,5X)////)
  336 CONTINUE
      GO TO 334
  332 PRINT 333,IERR
  333 FORMAT(1X,*ERROR -- IERR = *I5)
  334 STOP
      END
```

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM 78681		

4. Title and Subtitle	5. Report Date
Computer Program for Determining Mass Properties of a Rigid Structure	March 1978
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Reid A. Hull, Dr. John L. Gilbert and Phillip J. Klich	
	10. Work Unit No.
9. Performing Organization Name and Address	
National Aeronautics and Space Administration Langley Research Center Hampton, Virginia 23665	11. Contract or Grant No.
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Technical Memorandum
National Aeronatutics and Space Administration Washington, DC 20546	14. Sponsoring Agency Code

15. Supplementary Notes

Tentatively will become a Reference Publication

16. Abstract

A method is presented for determining the complete mass properties of a rigid structure.

Computer technology is utilized to exercise the greatest savings possible in solving for and printing out this information including properties relative to the total structure's principal axes.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Mass Properties Computer Rigid Structure	Unclassified - Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	46	$4.50

*For sale by the National Technical Information Service, Springfield, Virginia 22161