MR- 7034
Project 7052

CG: A COMBINATORIAL GEOMETRY MODULE FOR THE MICOM LASER TERMINAL HOMING SYSTEMS PROGRAM

Final Report

July 1973

Henry Lichtenstein
Hyo Won Bang

for

U. S. Army Missile Command
Huntsville, Alabama

MICOM Contract No. DAAH01-73-C-0761

Mathematical Applications Group, Inc.

3 Westchester Plaza

Elmsford, New York

MATHEMATICAL APPLICATIONS GROUP, INC.
3 WESTCHESTER PLAZA
ELMSFORD, N.Y. 10523
TEL. (914) 592 • 4646

July 18, 1974
M-4431

Dr. Ely M. Gelbard
Applied Physics Division
Building 208
Argonne National Laboratory
9700 South Cass Avenue
Argonne, Illinois 60439

Dear Dr. Gelbard:

Enclosed is a report describing the MAGI Combinatorial Geometry System. Some improvements and corrections have been made since this report was written, the most significant being the Checker described by Marty Cohen at the June 1974 ANS Meeting. The latest version (without listings) is described in the SAM-CE (Rev. C) Manual which will be released shortly. If you are interested in SAM-CE itself, notify Dr. Cohen or myself and we will send you a copy when it is published. The code (SAM-CE or the geometry package) will be distributed through RSIC.

Very truly yours,

Dr. Herbert A. Steinberg
Director, Scientific Services

HAS/br
Enc.

ABSTRACT

CG is the designation for a code module, which utilizes Combinatorial Geometry techniques to provide efficient modeling of complex three-dimensional targets. Its modular structure alleviates implementation in a parent program, such as the MICOM LTHS program. Interfacing with the parent code is limited to one "input" and one "output" labeled common block. Given a source ray and detector position, the CG module represents a versatile means of accounting for the purely geometric effects of a target, such as the T-54 tank.

ACKNOWLEDGEMENTS

Mr. Walter Guber is primarily responsible for the Combinatorial Geometry techniques. The T-54 tank model is a modification of the original description by Mr. Robert Goldstein.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Program Objective

CG is a FORTRAN code module comprising an executive routine (SUBROUTINE CGPACK), and eight subordinate routines, which comprise the standard MAGI Combinatorial Geometry (CG) package[1,2]. This module was designed for implementation in the MICOM LTHS program, with minimal code interfacing, to provide efficient ray tracing for complex target geometries.

1.2 Scope of Report

This report is designed to be used as an operational manual, and to serve as a descriptive summary of the Combinatorial Geometry technique in general, and the CG code module in particular. Consequently, the main body includes sections describing: (a) the CG technique; (b) preparation of CG input; and (c) the routines comprising the CG module. In addition, listings of the CG module and the CG description of a T-54 tank model are included in the appendices.

2. THE CG TECHNIQUE FOR THE DESCRIPTION AND COMPUTER PROCESSING OF COMPLEX THREE-DIMENSIONAL OBJECTS

2.1 Introduction

Combinatorial Geometry (CG) is essentially a technique for representing, in a computer, a mathematical model of a three-dimensional geometric configuration. Once in the computer, the configuration can be analyzed in many different ways by ray tracing techniques. For example, quantities such as volumes, surface areas, object boundaries, line of sight distances, etc. are readily determined. Regardless of the application, however, the basic concepts employed are the same. A discussion of these concepts can logically be broken down into two topics. That is, geometry description and ray tracing, which are discussed separately below.

2.2 Description of the Combinatorial Geometry Technique

In order to perform computer studies concerning a complex three-dimensional object one must first be able to prepare a mathematical model of the object, and its environment. The CG technique has been developed to permit a model to be produced, which is both accurate and suitable for a ray-tracing analysis program.

In effect the geometric description subdivides the problem space into unique regions. This is achieved through the use of ten specific geometric bodies (closed surfaces) and the orderly identification of the combination of those bodies, which define a region (space volume). The bodies, which will be discussed further in PREPARATION OF CG INPUT (Sec. 3), are as follows:

1. Rectangular Parallelepiped (RPP)
2. Box
3. Sphere
4. Right Circular Cylinder
5. Right Elliptical Cylinder
6. Truncated Right Angle Cone
7. Ellipsoid of Revolution
8. Right Angle Wedge
9. Arbitrary Convex Polyhedron of four, five or six sides (each side having three or four vertices).
10. Truncated Elliptic Cone

Except for the RPP's, all bodies may be arbitrarily oriented with respect to the x, y, z coordinate axes used to determine the space. It should be noted that the sides of an RPP must be parellel to the coordinate axes.

2.2.1 Region Description Technique

The basic technique for the description of the geometry consists of defining the location and shape of the various physical regions (wall, equipment, etc.) in terms of the intersections and unions of the volumes contained in a set of simple bodies. A special operator notation involving the symbols (+), (-), and (OR) is used to describe the intersections and unions. These symbols are used by the program to construct tables used in the ray-tracing portion of the problem.

If a body appears in a region description with a (+) operator, it means that the region being described is wholly contained in the body.

If a body appears in a region description with a (-) operator, it means that the region being described is wholly outside the body.

The (OR) operator is used to form regions as unions of subregions, where each subregion is defined in terms of one or more bodies, using (+) or (-) as described above. Then a point is in the region if it is in any subregion.

The technique of describing a physical region is best illustrated by an example. Consider an object composed of a sphere into which is inserted a cylinder. This is shown in cross section in Fig. 1(a).

To describe the object, we take a spherical body penetrated by a cylindrical body {Fig.1(b)}. Each body is numbered. Consider the sphere as body No. 1 and cylinder as body No. 2. If the materials in the sphere and cylinder are the same, then they can be considered as one physical region, say region 100 {Fig. 1(c)}.

The description of region 100 would be:

100 = (OR 1) (OR 2).

This means that a point is in region 100 if it is <u>either</u> inside body 1 <u>or</u> inside body 2.

If different materials are used in the sphere and cylinder, then the sphere with a cylindrical hole in it would be given a different region number (say 200) from that of the cylinder (300).

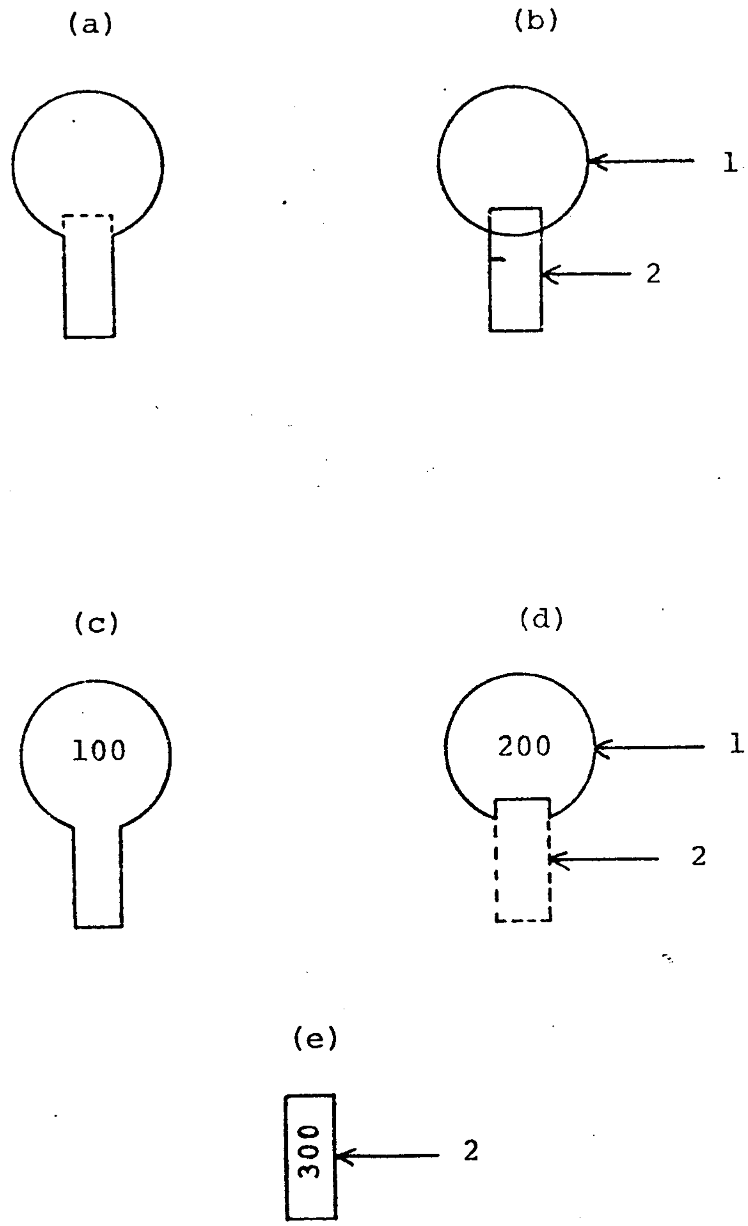


Figure 1 - Regions Produced by Intersections and Unions of Sphere and Cylinder

The description of region 200 would be {Fig. 1(d)}:

200 = (+1) (-2)

This means that points in region 200 are all those points inside body 1 which are not inside body 2.

The description of region 300 is simple {Fig. 1(e)}:

300 = (+2)

That is, all points in region 300 lie inside body 2.

This technique, of course, can be applied to combinations of more than two bodies and such region descriptions could conceivably contain a long string of (+), (-) and (OR) operators. The important thing to remember is that every spatial point in the geometry must be located in one and only one region. Further examples are given in Section 3.3.

3. PREPARATION OF CG INPUT

3.1 Introduction

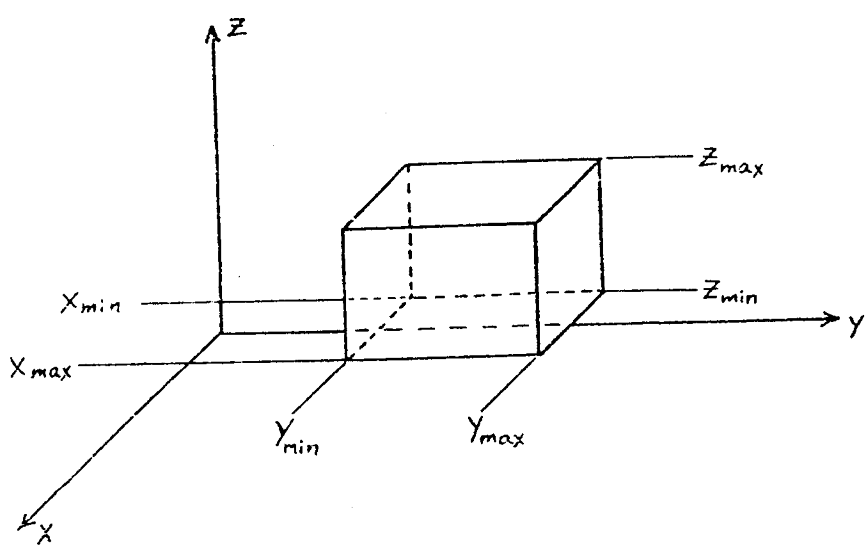
The user of the program will specify the geometry by establishing two tables. The first table will describe the type and location of the set of bodies used in the geometrical description. The second table will identify the physical region in terms of these bodies. The computer program processes these tables to put the data in the form required for ray tracing. All of the space must be divided into regions, and once again no point may be in more than one region.

3.2 Description of Body Parameters

The information required to specify each type of body is as follows.

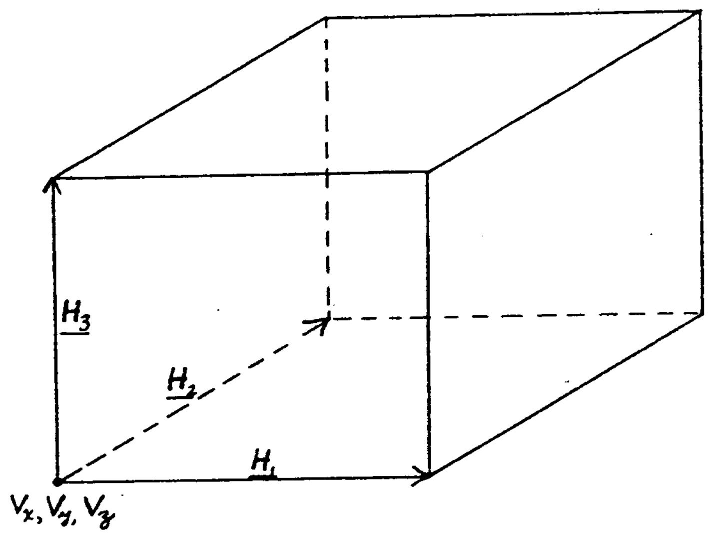
1. Rectangular Parallelepiped (RPP)

These bodies are used for gross subdivisions of the geometry and must have bounding surfaces parallel to the coordinate axes. Specify the maximum and minimum values of the x, y, and z coordinates which bound the parallelepiped.



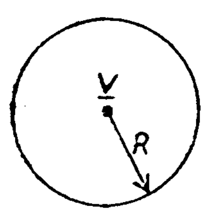
2. Box (BOX)

Specify the vertex V at one of the corners by giving its (x,y,z) coordinates. Specify a set of three mutually perpendicular vectors, H_i, representing the height, width, and length of the box, respectively. That is, the x,y, and z components of the height, width, and length vectors are given.



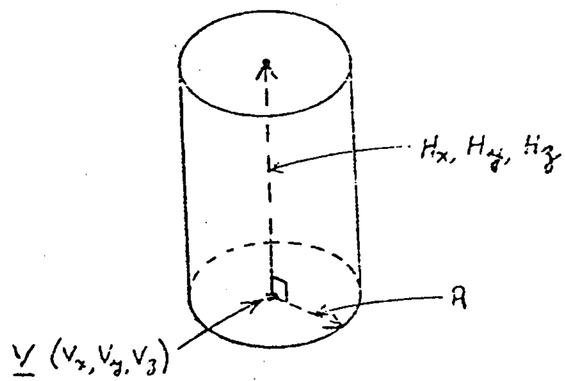
3. Sphere (SPH)

Specify the vertex V at the center and the scalar, R, denoting the radius.



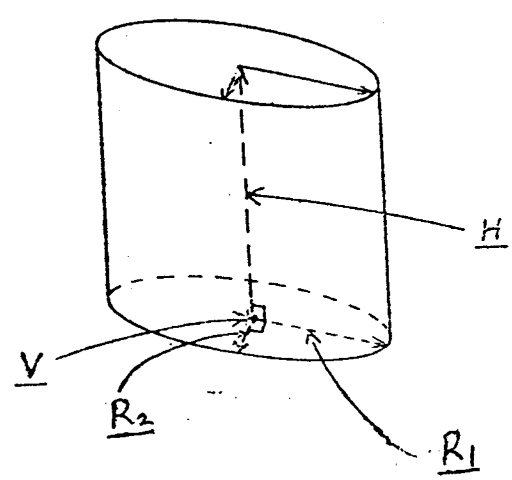
4. Right Circular Cylinder (RCC)

Specify the vertex $\underline{V}$ at the center of one base, a height vector, $\underline{H}$, expressed in terms of its x, y, and z components, and a scalar, R, denoting the base radius.



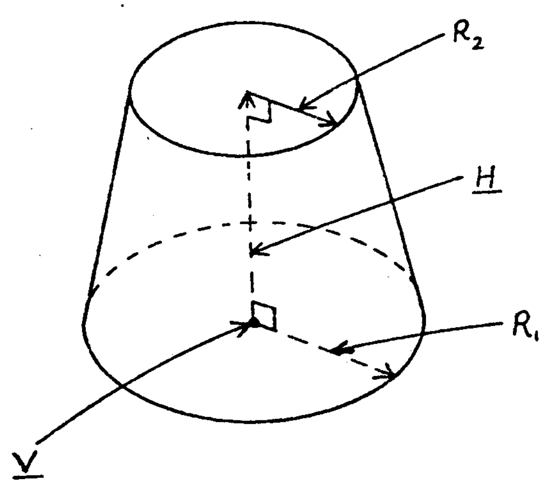
5. Right Elliptical Cylinder (REC)

Specify coordinates of the center of the base ellipse, a height vector, and two vectors in the plane of the base defining the semi-major and semi-minor axes, respectively.



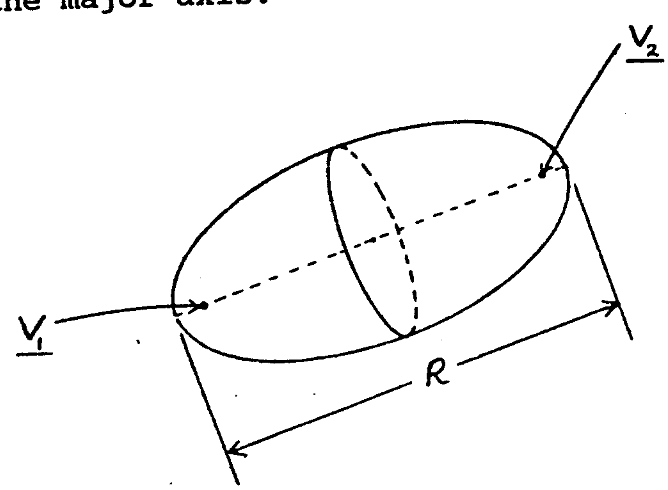
6. Truncated Right Angle Cone (TRC)

Specify a vertex **V** at the center of the lower base, the height vector, **H**, expressed in terms of its x, y, z components, and two scalars, R_1 and R_2, denoting the radii of the lower and upper bases.



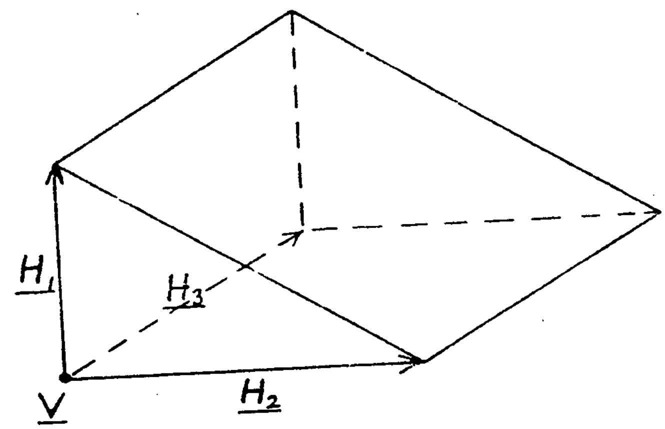
7. Ellipsoid (ELL)

Specify two vertices, $\underline{V_i}$, denoting the coordinates of the foci and a scalar, R, denoting the length of the major axis.



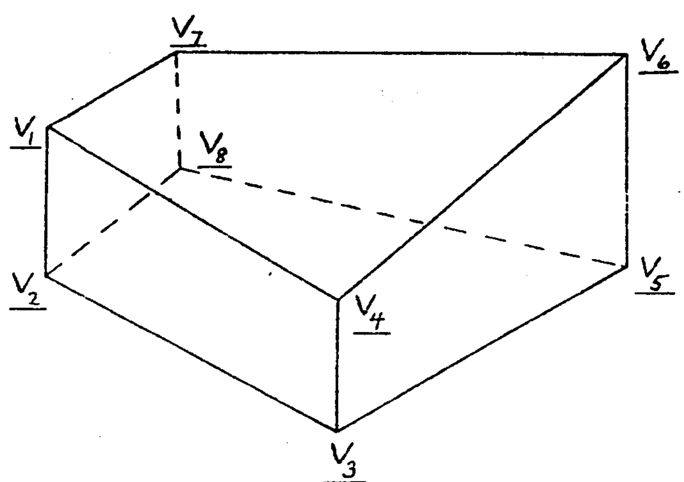
8. Right Angle Wedge (RAW)

Same input as for the boxes. However, $\underline{H_1}$ and $\underline{H_2}$ describe the two legs of the right triangle of the wedge.



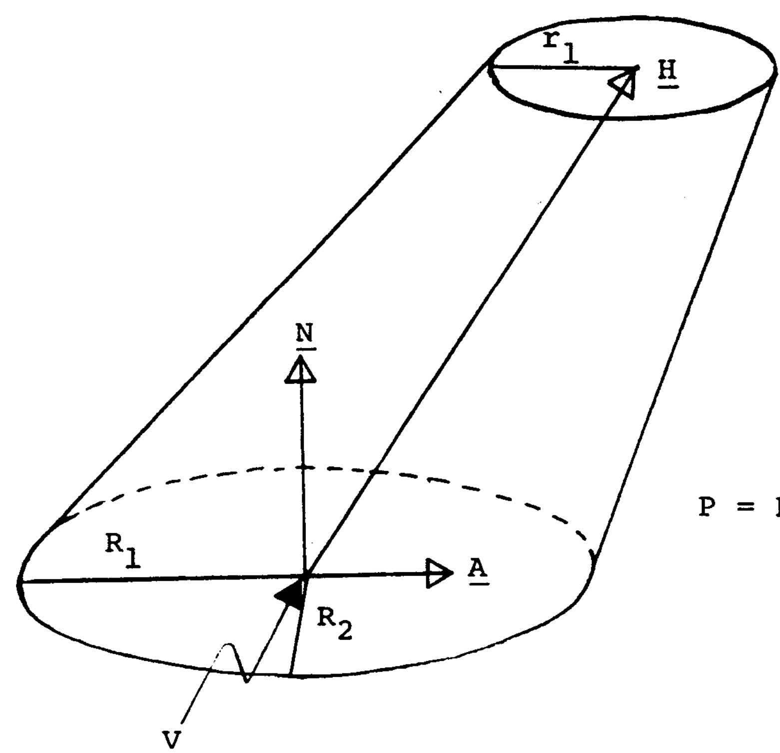
9. Arbitrary Polyhedron (ARB)

Assign an ordinal number (1 to 8) to each vertex. For each vertex, give the x, y, z coordinates. For each side of the figure list the ordinal vertex numbers. The vertices and side descriptions may be given in any order. An example is given later.



10. Truncated Elliptic Cone (TEC)

Specify the coordinates of the vertex V at the center of the larger ellipse; the x, y, and z components of height vector H; the components of normal vector, N, directed inward at V; the components of direction vector, A, along major axis; the semi-major and semi-minor axes of larger ellipse, R_1 and R_2, respectively; the ratio, P, of the larger to the smaller ellipse axis. Note that direction vectors N and A are normalized internally (after input printout).



$P = R_1/r_1$

3.3 Examples of Region Descriptions

Some representative geometries and their input descriptions are shown below.

Example 1 - Two Spheres Within an RPP (See Fig. 2)

The body input table is shown below.

TABLE I - BODY INPUT DESCRIPTION

Body	Type of Data Required
1	List the six bounding coordinate values (X_{min}, X_{max}, Y_{min}, Y_{max}, Z_{min}, Z_{max})
2	List the vertex and radius of sphere 2
3	List the vertex and radius of sphere 3

One possible region input table is shown below.

TABLE II - REGION DESCRIPTION

Region	Input
100	(+1) (-2) (-3) (Region 100 is composed of all points interior to RPP No. 1 and exterior to spheres 2 and 3)
200	(+3) (-2) (Region 200 is composed of all points interior to sphere 3 and exterior to sphere 2)
300	(+2) (+3) (Region 300 is composed of all points which are in sphere 2 and are also in sphere 3)
400	(+2) (-3) (Region 400 is composed of all points interior to sphere 2 and exterior to sphere 3)
500	(OR 2) (OR 3) (If desired, one region, the total of regions 200, 300, and 400, can be defined as region 500).

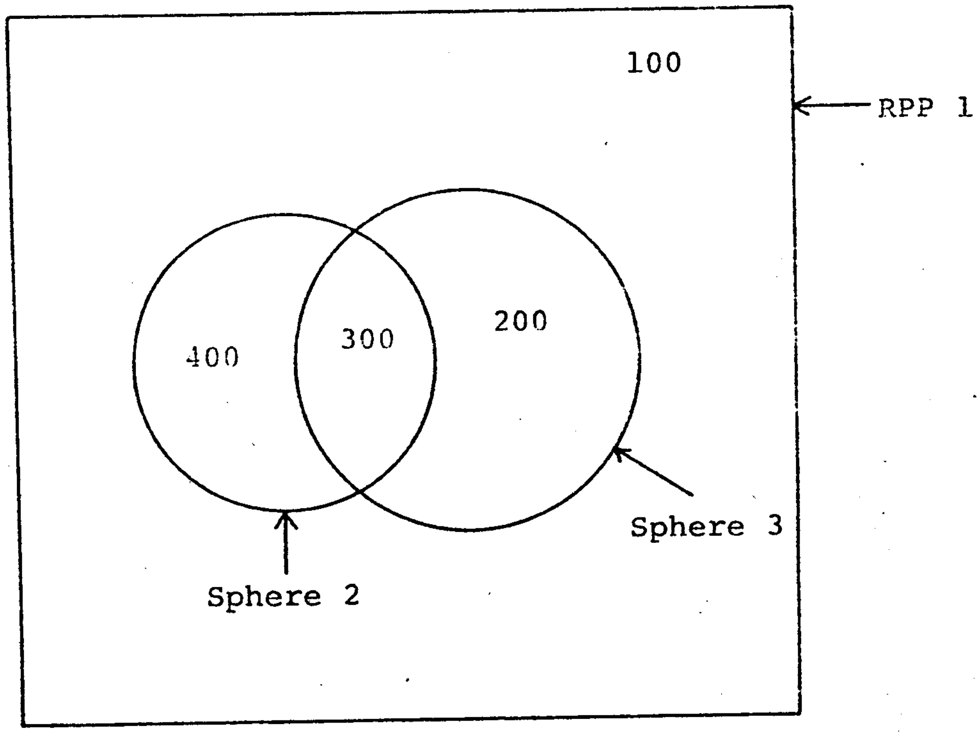


Figure 2 - Regions Produced by Intersections and Unions of Two Spheres

Example 2 - Cylinder Divided into Two Regions by a Box and with a Sphere at One End (See Fig. 3)

TABLE I - BODY INPUT DESCRIPTION

Body	Type of Data Required
1	List the six bounding coordinates of the RPP
2	List the vertex, radius, and height vector of cylinder
3	List center and radius of sphere
4	List coordinates of one corner and components of three vectors representing sides of box.

The region input is as follows.

TABLE II - REGION DESCRIPTION

Region	Input
100	(+1) (-2) (-3) (All points interior to the RPP and exterior to the cylinder and sphere. Note that region 100 includes all of the space contained inside body 4, except that portion inside cylinder 2. This space can be assigned a special region number, if desired. If, as in this example, it is not desired, it is not necessary).
200	(+2) (-4) (All points interior to the cylinder, and outside the box).
300	(+3) (-2) (All points interior to the sphere and external to the cylinder).
400	(+2) (+4) (All points interior to the cylinder and also inside the box).

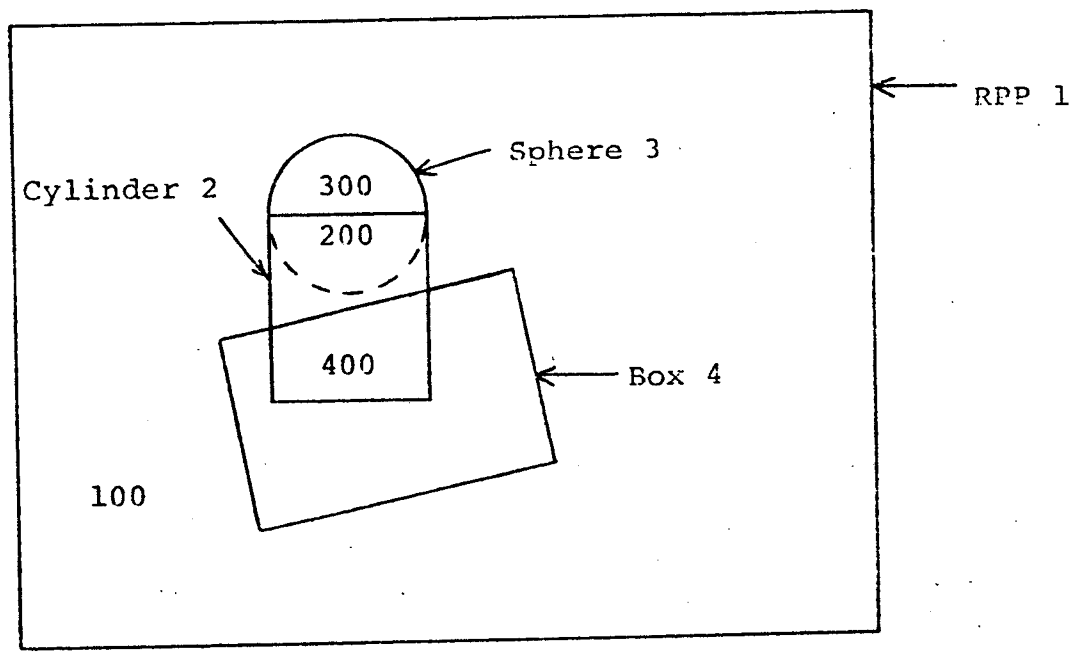


Figure 3 - Regions Produced by Intersections and Unions of Sphere, Circular Cylinder, and Box

Example 3 - Multiple Region Capability - Cylinder Containing Two Spheres, All Inside an RPP (See Fig. 4)

TABLE I - BODY INPUT DESCRIPTION

Body	Type of Data Required
1	List RPP data
2	List cylinder input
3	List sphere input
4	List sphere input

TABLE II - REGION DESCRIPTION

Region	Input
100	(+1) (-2)
200	(OR 3) (OR 4) (All points interior to 3 or 4)
300	(+2) (-3) (-4) (All points in the cylinder but not in the spheres).

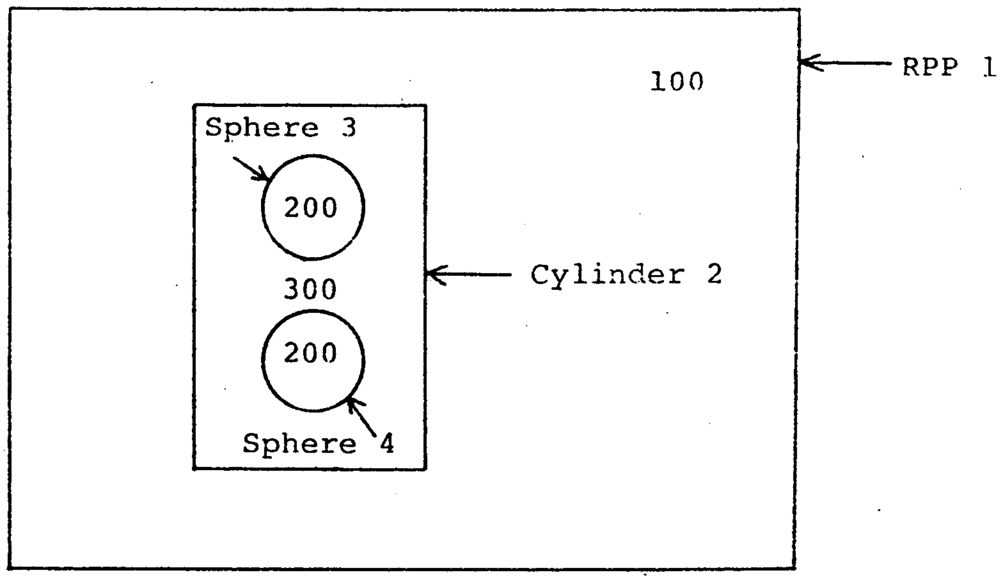


Figure 4 - Example of Physical Region Produced from Unconnected Regions Using "OR" Statement

3.4 Card Input Formats

The first card of a CG input deck is a header card:

FORMAT(I10,E15.4,A3,13A4)

Columns	Item	Description
1-10	IPRINT	= 0, print out body and region data which follow;
		= 1, print out body and region data, as well as the internal arrays in which they are stored;
		= 2, suppress geometry printout;
11-25	STRETCH	Scale factor for linear CG dimensions; a blank effects default (unity).
26-80	TITLE	55 arbitrary Hollerith characters.

The remainder of the CG input deck comprises body cards and region cards, which must appear in the order described below.

1. Body Cards

The computer assigns to each body an ordinal number which depends on the order in which the body cards are read in. Therefore, it is most important that the card sequence match the numbering sequence used in the region descriptions. Note that no gaps may be left in the body numbering sequence.

Ten different body types may be employed. The standard format for each body is as follows.

Columns	Input
1-2	Blank
3-5	Three-letter body identifier.
6	Blank

Columns	Input
7-10	Four characters or arbitrary integer data
11-70	Divided into six floating point fields of 10 columns each. Body dimensions are given here.

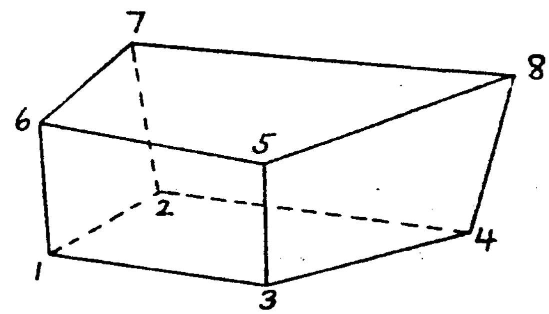
Table 1 describes the input required for each body. The quantities V, H, etc., were defined in Section 3.2.

Note that the last card of the body data must be END punched in columns 3, 4, 5. This is the signal that all body data have been treated.

TABLE I INPUT REQUIRED FOR EACH BODY TYPE

Body Type	3-letter ID	Card Columns 11-20	21-30	31-40	41-50	51-60	61-70	Number of Cards Needed
Rectangular Parallepiped	RPP	X_{min}	X_{max}	Y_{min}	Y_{max}	Z_{min}	Z_{max}	1
Box	BOX	Vx	Vy	Vz	H1x	H1y	H1z	1 of 2
		H2x	H2y	H2z	H3x	H3y	H3z	2 of 2
Sphere	SPH	Vx	Vy	Vz	R	-	-	1
Right Circular Cylinder	RCC	Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
		R	-	-	-	-	-	2 of 2
Right Elliptic Cylinder	REC	Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
		R1x	R1y	R1z	R2x	R2y	R2z	2 of 2
Ellipsoid of Revolution	ELL	V1x	V1y	V1z	V2x	V2y	V2z	1 of 2
		R	-	-	-	-	-	2 of 2
Truncated Cone	TRC	Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
		R_1	R_2	-	-	-	-	2 of 2
Right Angle Wedge	RAW	Vx	Vy	Vz	H1x	H1y	H1z	1 of 2
		H2x	H2y	H2z	H3x	H3y	H3z	2 of 2
Arbitrary Polyhedron	ARB	V1x	V1y	V1z	V2x	V2y	V2z	1 of 5
		V3x	V3y	V3z	V4x	V4y	V4z	2 of 5
		V5x	V5y	V5z	V6x	V6y	V6z	3 of 5
		V7x	V7y	V7z	V8x	V8y	V8z	4 of 5
		Face Descriptions (See Note following page)						5 of 5
Truncated Elliptic Cone	TEC	Vx	Vy	Vz	Hx	Hy	Hz	1 of 3
		Nx	Ny	Nz	Ax	Ay	Az	2 of 3
		R_1	R_2	P	-	-	-	3 of 3

Note: Each of the six faces of an ARB are described by a four-digit number giving the number of the four vertex points at the corners. The point numbers for each face must be entered in either clockwise or counterclockwise order. The format is 6(F10.0) starting in Column 11. An example is shown below.



FACE	1	2	3	4	5	6
PTS	1653.	3584.	4278.	1672.	1342.	5678.

Figure 5 - Example of Arbitrary Polyhedron

2. Region Cards

Each region must be numbered and described by a logical combination of the bodies which make up that region. Use as many cards as necessary to describe each region and begin each region on a new card. The input format, described below, is (2x, A3, 5x, 9 (A2, I5)).

Columns	Input
1-2	Blank
3-5	Arbitrary Hollerith data
6-10	Blank
11-73	Divided into nine fields, of 7 columns each. The first two columns of each field are reserved for the OR operator. The third column is for the (+) or (-) operator. The last four columns are for the body number.

Use as many cards of the above type as needed to complete a region description, but leave Columns 1-10 blank on all continuation cards.

The last region description card must be followed by a card containing END in Columns 3, 4, and 5. This informs the code that all regions have been described.

4. THE CG CODE MODULE

4.1 Introduction

The CG code module comprises an executive routine, subordinate to a calling program such as LTHS, and the MAGI package of CG routines, including one which computes surface normals. Descriptions of every routine in this module, together with calling information, are given in this section.

4.2 Subroutine CGPACK

CGPACK is the executive routine of the CG module. It is the only link between the parent (calling) program and the CG routines. All communication with the calling program is made via two labeled common blocks: one for the information specified in the calling program, /GIVE/; the other for the information returned to the calling program, /TAKE/.

The function of CGPACK is to employ the CG routines such that the following information is returned to the calling program: Given a source ray (described by position coordinates and direction cosines) and a detector point, will the ray (a) miss, or (b) hit the target, as described by CG input; if (b), then (c) determine coordinates of-, CG region of-, direction to detector from-, and unit surface normal at-, the target position hit; and, if (b), will the reflected ray (d) "see" the detector point (i.e. the reflected ray will not be intercepted by another part of the target).

Thus, CGPACK acts as the proverbial "black box", which, given a source ray and detector point, governs the CG computations necessary to account for the purely geometric effects of a complex three-dimensional target.

The logical flow of CGPACK is given in Fig. 6, below.

CALLING INSTRUCTIONS

(a) Called from: parent program

CALL CGPACK

(b) Subroutines called: GENI

GETIR

GNORM

G1

(c) Variables required:

COMMON/GIVE/INT,IPT,IREX,IRTF,IRTL,XD(3),XS(3),WS(3)

INT - logical unit number for CG input (usually 5);

IPT - logical unit number for printing (usually 6);

IREX - escape region number, the outermost region in CG input (usually defined by an RPP surrounding a smaller RPP, which in turn encloses all other CG bodies);

IRTF - first target region number (all target regions must be input contiguously);

IRTL - last target region number;

XD(3) - coordinates of detector position;

XS(3) - coordinates of source position (fixed);

WS(3) - direction cosines of source ray;

Notes: (1) For a given execution, INT and IPT are specified once.

(2) For a given CG input, IREX, IRTF, and IRTL are fixed.

(3) For fixed source and detector, only source direction cosines vary between calls to CGPACK.

(d) Variables returned:

COMMON/TAKE/MKH,IRT,XT(3),WT(3),WN(3)

MKH - (miss, kill, hit) target flag (see Notes below);

IRT - region number of target position hit ($MKH \geq 0$);

XT(3) - coordinates of target position hit ($MKH \geq 0$);

WT(3) - direction cosines of unit vector at XT(3) toward detector ($MKH \geq 0$);

WN(3) - direction cosines of outward normal at XT(3) ($MKH > 0$);

Notes: (1) MKH = -1, source ray missed the target.

(2) MKH = 0, source ray hit target, but reflected ray subsequently killed along track to detector, by intersecting another target position.

(3) MKH = 1, source ray hit target, and reflected ray reached detector.

Figure 6 - CGPACK Logical Flow

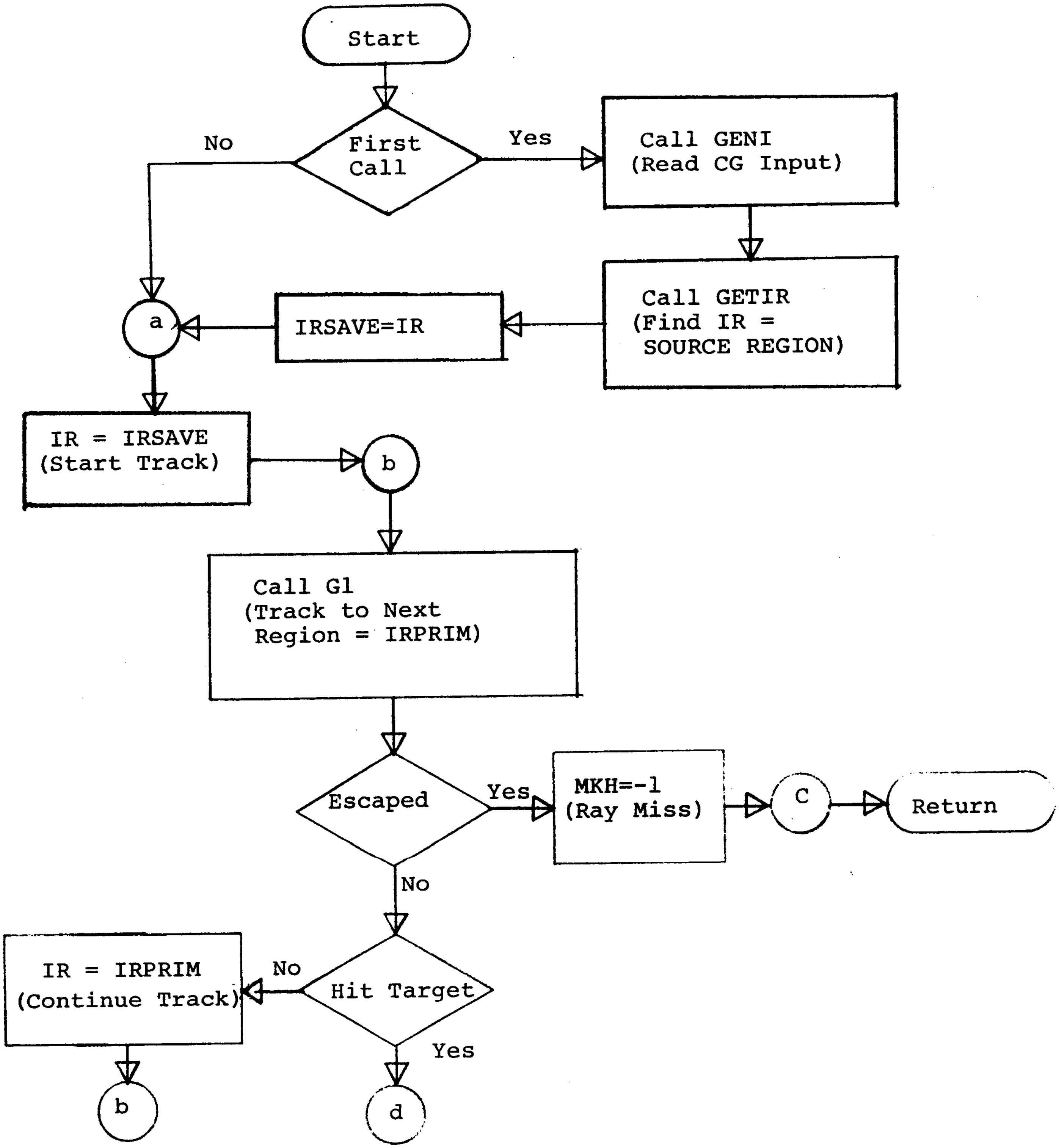
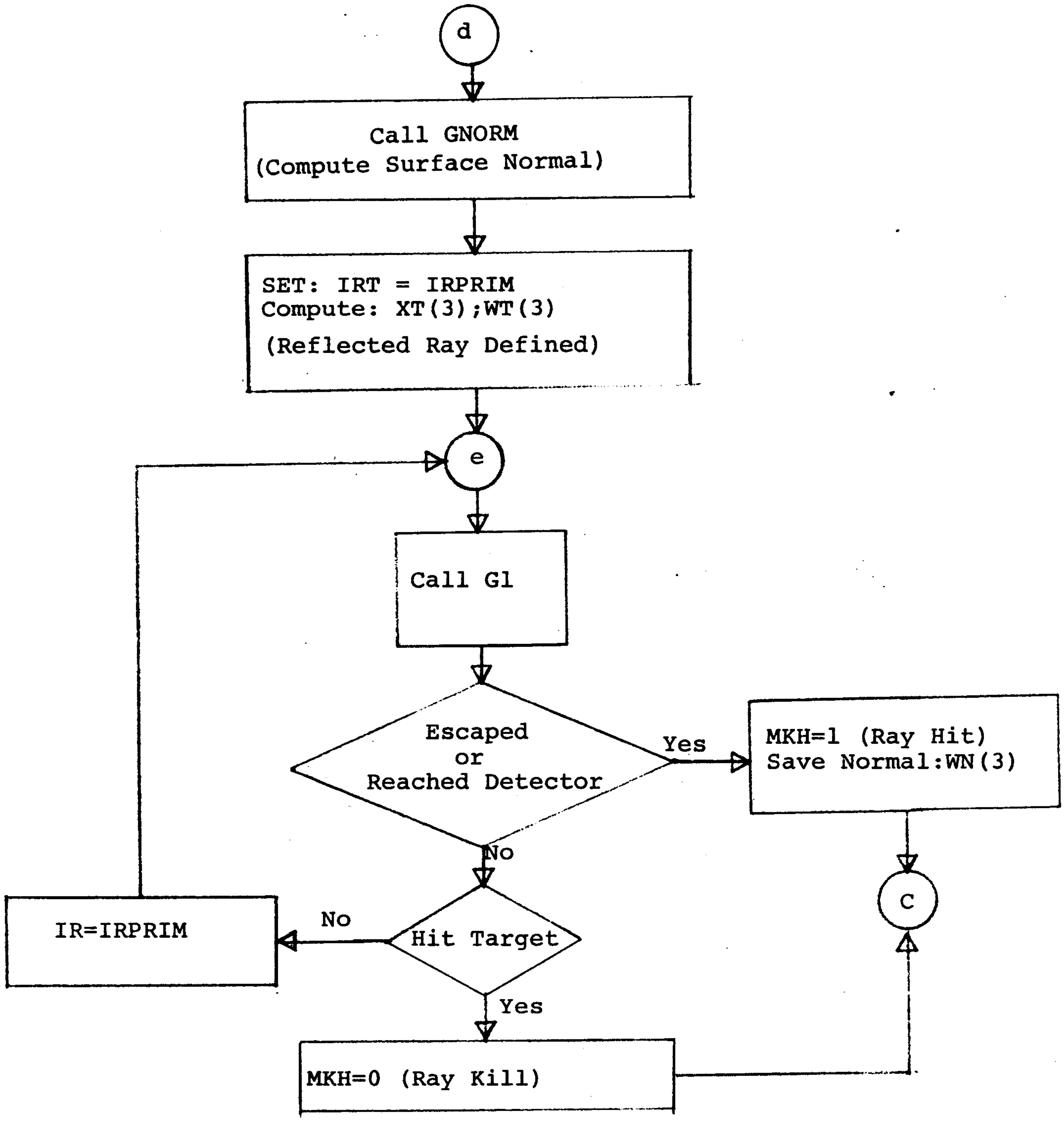


Figure 6 - Continued



4.3 Subroutine GENI

GENI is the principal CG input processing routine. It reads in the CG data, checks for errors, and stores the data into a fixed point and floating point array, MA and FPD, respectively, in the form required by the tracking routines. For an ARB body type (see Section 3.2), GENI calls GARB for additional processing. An option has been implemented which allows the unit of the linear body dimensions to be altered by a user supplied scaling factor (see description of header card in Section 3.4). The default unit is the one utilized in preparing the input deck (i.e. scaling factor equals unity).

CALLING INSTRUCTIONS

(a) Called from: CGPACK

CALL GENI

(b) Subroutine called: GARB

(c) Principal variables required:

IN - logical unit number for CG input;

IOUT- logical unit number for printing;

(d) Principal variables returned:

EPS - internally computed smallest number such that (1.+EPS)>1. (i.e. "machine epsilon");

MA(I) - CG integer array

FPD(I) -CG floating point array

4.4 Subroutine GARB

GARB is an auxiliary input processing routine, called by GENI whenever an ARB body type is encountered on input. GARB accepts the originally specified ARB input data read by GENI (see Table 1), and computes a unit normal vector and a minimum distance to the origin for each plane containing a side of the ARB. These data are then stored in the locations of the FPD array used to store the original data. The number of ARB sides (up to six), and a scaler characteristic of the ARB's minimum dimension (used for round-off tests) are also stored in the FPD array.

CALLING INSTRUCTIONS

(a) Called from: GENI

CALL GARB (FPD(LF),IERR)

(b) Subroutines called: none

(c) Principal variables required:

F - FPD array read in GENI

(d) Principal variables returned:

F - FPD array, modified as described above.

4.5 Subroutine G1

G1 is the principal CG tracking routine. Given a position and direction of a ray, and the current region number, G1 will compute the distance tracked in the current region, the cumulative distance tracked from start of trajectory, and the next region number encountered. G1 also determines the body number, whose surface defines the region (exit) boundary, and that surface number (negative if leaving and positive if entering body).

CALLING INSTRUCTIONS

(a) Called from: CGPACK

CALL G1(S)

(b) Subroutines called: GG

GGTEC

GP (for debugging only)

(c) Principal variables required:

XB(3) - position coordinates at start of track;

WB(3) - direction cosines of ray;

IR - current region number;

DIST - cumulative distance traveled from XB(3);

NASC - flags the start of a new track if negative;

KLOOP - counter of new tracks;

(d) Principal variables returned:

KLOOP - incremented for a new track;

NASC - body number whose surface defines region (exit) boundary;

LSURF - surface of body NASC crossed at region boundary;

S - distance tracked in region IR;

DIST - incremented by S;

IRPRIM - region number entered upon crossing LSURF of body NASC.

4.6 Subroutine GG

GG is the distance calculating routine for all body types except the TEC, which is handled by GGTEC. Given a position and direction of a ray, and a body number, GG computes the distance to entry, RIN, and the distance to exit, ROUT, measured from the given position coordinates of the ray.

RIN and ROUT are then stored in the FPD array, at the locations set aside for this body. GG also stores the indices for the entry and exit surfaces of the body, LRI and LRO, respectively, in the MA array. As a time saving device, the track counter KLOOP is also saved, so that any subsequent call to GG, for the same body and KLOOP value, will bypass the computations of RIN and ROUT, and retrieve the previously computed values from the FPD array.

CALLING INSTRUCTIONS

(a) Called from: G1

GETIR

CALL GG(LOCAT)

(b) Subroutines called: none

(c) Principal variables required:

LOCAT - pointer for integer data in MA array for the body;

KLOOP - track counter;

(d) Principal variables returned:

RIN - distance to body entry;

ROUT - distance to body exit;

LRI - entry surface;

LRO - exit surface;

4.7 Subroutine GGTEC

GGTEC performs the identical functions for the TEC body as GG does for all other body types.

4.8 Subroutine GETIR

GETIR determines the region number IR corresponding to a given set of position coordinates and (arbitrary) direction cosines.

This function is performed for a source ray, similar to the manner in which G1 determines the region number encountered at a boundary crossing.

CALLING INSTRUCTIONS

(a) Called from: CGPACK

CALL GETIR

(b) Subroutines called: GG

GGTEC

(c) Principal variables required:

XB(3) - position coordinates for which region number is required;

WB(3) - arbitrary direction cosines;

(d) Principal variables returned:

IR - region number for point XB(3);

4.9 Subroutine GNORM

GNORM computes the unit normal vector to the target surface struck by the source ray*. The surface point struck is computed by tracing the cumulative distance tracked by G1, DIST, along the source direction from the source point. The sign of the normal is chosen so that it is oriented opposite to the source ray direction.

CALLING INSTRUCTIONS

(a) Called from: CGPACK

(b) Subroutines called: none

(c) Principal variables required:

XB(3) - position at start of track (equal to source position coordinates);

*Several algorithms were borrowed from the MORSE Code[3].

WB(3) - source direction cosines;

DIST - distance to target point struck;

NASC - body number struck;

LSURF - surface of NASC struck;

(d) Principal variables returned:

UN(3) - direction cosines of unit normal;

4.10 Subroutine GP

GP is a debug printout routine. It can only be utilized by G1 if the variable IDBG is not set to zero. Since this option is not utilized during normal execution, IDBG has been set to zero in CGPACK.

5. SUMMARY

The CG code module, described herein, comprises an efficient software package for the geometric modeling of complex three-dimensional targets, such as a Russian T-54 tank (see CG input listing in Appendix B). The minimal required interfacing with the parent code, via two labeled common blocks, makes this module a highly adaptable "black box."

REFERENCES

1. M. O. Cohen, et al, "SAM-CE: A Three Dimensional Monte Carlo Code for the Solution of the Forward Neutron and Forward and Adjoint Gamma Ray Transport Equations, Revision A", DNA 2830F, Rev. A, MR-7021, Rev. A (July 1972).

2. S. Hui, et al, "SAMCEP: An Application of Correlated Monte Carlo to the Simultaneous Solution of Multiple, Perturbed, Time-Dependent Neutron Transport Problems in Complex Three-Dimensional Geometry", BRL CR62, MR-7020 (January 1972).

3. E. A. Straker, et al, "The MORSE Code with Combinatorial Geometry", DNA 2860T, SAI-72-511-LJ (May 1972).

APPENDIX A

CG MODULE (LISTING)

```
MEMBER NAME  CGPACK
      SUBROUTINE CGPACK
      COMMON/GIVE/INT,IPT,IREX,IRTF,IRTL,XD(3),XS(3),WS(3)
C----- INT=LOGICAL UNIT FOR GEOMETRY INPUT ( USUALLY 5).
C----- IPT=LOGICAL UNIT FOR PRINTER ( USUALLY 6).
C----- IREX=ESCAPE REGION.
C----- IRTF=FIRST TARGET REGION.
C----- IRTL=LAST TARGET REGION, WHERE ALL TARGET REGIONS INPUT CONTIGU.
C----- XD(3)=COORDINATES OF DETECTOR POSITION.
C----- XS(3)=COORDINATES OF SOURCE POSITION.
C----- WS(3)=DIRECTION COSINES OF SOURCE RAY.
      COMMON/TAKE/MKH,IRT,XT(3),WT(3),WN(3)
C----- MKH=MISS, KILL, HIT TARGET FLAG.
C----- IRT=TARGET REGION OF INITIAL HIT.
C----- XT(3)=COORDINATES OF TARGET HIT.
C----- WT(3)=DIRECTION COSINES TOWARD DETECTOR AT XT(3).
C----- WN(3)=DIRECTION COSINES OF OUTWARD NORMAL AT XT(3).
      COMMON/NORMAL/UN(3)
      COMMON/GEOM3/EPS,IRHA(200)
      COMMON/PAREM /XB(3) ,WB(3) ,E     ,IR    ,T     ,IDET  ,F     ,     PAREM
     1              NHIST ,WC    ,J12345,WP(3) ,XP(3) ,EPRIM ,ATWT  ,     PAREM
     2              NCOB  ,CSTHT ,U     ,LCHI  ,IATWT ,IERR  ,IDBG        PAREM
     3,             IRPRIM,NASC  ,LSURF ,NBO   ,LRI   ,LPO   ,KIN   ,     PAREM
     4              ROUT  ,KLOOP ,LOOP  ,ITYPE ,PINF  ,NOA ,DIST
      COMMON/TAPE  /IN    ,IOUT  ,IEDT  ,INT1  ,INT2  ,IAGG,ITRAN,INTER TAPE IN
      DATA IFIRST/0/
      IF(IFIRST.NE.0)GO TO 100
      IN=INT
      IOUT=IPT
      IDBG=0
      CALL GENI
      EP=1.0-10.0*EPS
  100 DO 110 I=1,3
      WB(I)=WS(I)
  110 XB(I)=XS(I)
      IF(IFIRST.NE.0)GO TO 113
      CALL GETIR
      IRSAVE=IR
      IFIRST=1
      GO TO 115
  113 IR=IRSAVE
  115 NASC=-1
  120 CALL G1(S)
      IF(IRPRIM.EQ.IREX)GO TO 700
      IF(IRTF.LE.IRPRIM.AND.IRPRIM.LE.IRTL)GO TO 200
C------HAVE NOT HIT TARGET YET.
      IR=IRPRIM
      GO TO 120
C------INITIAL HIT OF TARGET.
  200 CALL GNORM
      IRT=IRPRIM
      DIS=EP*DIST
      DO 210 I=1,3
      XB(I)=XB(I)&DIS*WB(I)
      XT(I)=XB(I)
  210 WB(I)=XD(I)-XB(I)
      DISD=SQRT(WB(1)**2&WB(2)**2&WB(3)**2)
      DO 220 I=1,3
      WB(I)=WB(I)/DISD
```

MEMBER NAME CGPACK

```
  220 WT(I)=WB(I)
      NASC=-1
  230 CALL G1(S)
      IF(IRPRIM.EQ.IREX.OR.DIST.GE.DISD)GO TO 900
      IF(IRTF.LE.IRPRIM.AND.IRPRIM.LE.IRTL)GO TO 800
      IR=IRPRIM
      GO TO 230
C------RAY MISSED TARGET.
  700 MKH=-1
      GO TO 1000
C------RAY HIT TARGET AFTER REFLECTION.
  800 MKH=0
      GO TO 1000
C------RAY REACHED DETECTOR.
  900 MKH=1
      DO 910 I=1,3
  910 WN(I)=UN(I)
 1000 RETURN
      END
```

```
MEMBER NAME  GARB
      SUBROUTINE GARB  ( F,IERR)
CC       NEW NAME FOR SUBROUTINE ALBERT
      COMMON/JAN263/DATEX
      DIMENSION F(9),X(3,8),IX(4,6),V(3,4)
      COMMON/ GEOM4/ IPRINT
      COMMON/TAPE  /INT   ,IOUT  ,IEDT   ,INT1   ,INT2   ,IAGG,ITRAN,INTER TAPE
      DO 10 I=1,8
      DO 10 J=1,3
      K=3*(I-1)&J
      X(J,I)=F(K)
   10 F(K)=C.
C
C
      IPMAX=0
      DO  40 I=1,6
      K=0
      NSIDE=I-1
      M=F(I&24)
      DO  20  J=1,4
      IX(J,I)=M-(M/10)*10
      IF( IX(J,I).NE.0) K=K&I
      IF( IX(J,I).GT.IPMAX) IPMAX=IX(J,I)
   20 M=M/10
      IF( K.EQ.0 ) GO TO 50
      IF( K.GE.3 ) GO TO 40
      WRITE(IOUT,22) I,J,F(I&24)
   22 FORMAT(26H ERROR IN SIDE DESCRIPTION,2I10,F10.0)
      GO TO 200
   40 CONTINUE
      NSIDE=6
C
C  FIND  MINIMUM DISTANCE BETWEEN POINTS
C
   50 DMIN=1.0E&20
      IPMAX=IPMAX-1
      DO  60 I=1,IPMAX
      DO  60 J=I,IPMAX
      D=(X(1,I)-X(1,J&1))**2&(X(2,I)-X(2,J&1))**2&(X(3,I)-X(3,J&1))**2
      IF((D.GT.0).AND.(D.LT.DMIN)) DMIN=D
   60 CONTINUE
      DMIN=0.5*SQRT(DMIN)
C
      IPMAX=IPMAX&1
      DO   100   I=1,NSIDE
      J1= IX( 3,I)
      DO   62    J=2,4,2
      J2= IX( J,I)
      DO   62    K=1,3
   62 V(K,J) = X(K,J1)- X(K,J2)
      A= V(2,2)*V(3,4) -  V(3,2)*V(2,4)
      B= V(3,2)*V(1,4) -  V(1,2)*V(3,4)
      C= V(1,2)*V(2,4) -  V(2,2)*V(1,4)
      D=-(A*X(1,J1)& B*X(2,J1)& C*X(3,J1) )
      EPS= SQRT(A*A & B*B & C*C)
      NPL=C
      NMI=0
      DO   80  J=1,IPMAX
      DS=(A*X(1,J) & B*X(2,J)& C*X(3,J) & D)/EPS
```

```
MEMBER NAME   GARB
      ADS=ABS(DS)
      IF(ADS.GT.DMIN)GO TO 70
      IF(ADS.LT.0.01*DMIN)GO TO 80
      WRITE(IOUT,65)F(I&24)
   65 FORMAT(32HOVERTICES NOT CO-PLANER FOR FACE,E15.7//)
   70 IF(DS.GT.0.0)GO TO 74
      NMI=NMI&1
      GO TO 80
   74 NPL=NPL&1
   80 CONTINUE
      IF( (NMI.EQ.0).AND.(NPL.GT.0)) GO  TO  90
      EPS=-EPS
      IF(   (NPL.EQ.0).AND.(NMI.GT.0))GO  TO  90
      IERR=IERR&1
      WRITE(IOUT,85) I,NMI,NPL,F(I&24),A,B,C,D
   85 FORMAT(26H ERROR IN FACE DESCRIPTION,3I10/5E15.7 )
      GO TO 200
   90 F( 4*I-3)=A/EPS
      F( 4*I-2)=B/EPS
      F( 4*I-1)=C/EPS
      F( 4*I  )=D/EPS
  100 CONTINUE
      F(25)=NSIDE
      F(26)=DMIN
      IF(IPRINT.NE.2) WRITE(IOUT,195) (F(I),I=1,26)
  195 FORMAT( 30X,4E15.7 )
  200 RETURN
      END
```

```
MEMBER NAME  GENI
      SUBROUTINE GENI
      DIMENSION IBIAS(10),ITY(11),IJ(21),IK(21)
      COMMON / GEOM2 /-LOCREG(200),NUMBOD(200),IROR(200),MA(4000),
     1                  FPD(4000),LDATA,NUMB,NUMR
      COMMON/GEOM3/EPS,IRHA(200)
      COMMON/ GEOM4/ IPRINT
      COMMON/PAREM /XB(3)  ,WB(3) ,E      ,IR     ,T      ,IDET  ,F     ,    PAREM
     1                NHIST ,WC    ,J12345,WP(3) ,XP(3) ,EPRIM ,ATWT  ,    PAREM
     2                NCDB  ,CSTHT ,U      ,LCHI  ,IATWT ,IERR  ,IDBG       PAREM
     3,               IRPRIM,NASC  ,LSURF  ,NBO   ,LRI   ,LRO   ,RIN   ,    PAREM
     4                ROUT  ,KLOOP ,LOOP   ,ITYPE ,PINF  ,NOA ,DIST
      COMMON/TAPE  /IN     ,IOUT  ,IEDT   ,INT1   ,INT2  ,IAGG,ITRAN,INTER TAPE IN
      EQUIVALENCE(IBIAS,IK),(ITY,IK(11))
C   1   2   3   4   5   6   7   8   9   10   11
C ARB SPH RCC REC TRC ELL BOX WED RPP TEC  END
C 24  -2  1   6   2   1   6   6   0   9
      DATA IJ    / 24,-2,1,6,2,1,6,6,0,9,
     1         3HARB,3HSPH,3HRCC,3HREC,3HTRC,3HELL,3HBOX,3HWED,3HRPP,
     2   3HTEC,3HEND/
C------RAW AND WED ARE SAME BODY.
      DATA IRAW,IWED/3HRAW,3HWED/                                            A
      DATA IOR,IBL,IEND/2HOR,3H   ,3HEND/
C------COMPUTE MACHINE EPS.
      EPS=1.0
      DO 10 I=1,100
      EPS=EPS/2.0
      IEPS=I
      IF((1.0&EPS).EQ.1.0)GO TO 11
   10 CONTINUE
   11 EPS=2.0*EPS
      IEPS=IEPS-1
      WRITE(IOUT,12)EPS,IEPS
   12 FORMAT(13HOMACHINE EPS=E13.6,5H= 2(-I2,1H)//)
C------SET WB FOR GETIR.
      WB(1)=1.0/SQRT(3.0)
      WB(2)=WB(1)
      WB(3)=WB(1)
      PINF=1.0E&20
      KLOOP=0                                                              6/73
      READ(IN,70) IPRINT,STRECH, (MA(I), I=1,14)
CC       IPRINT=0 BODY AND REGIONS ONLY, =1 ALL DATA, =2 NO DATA.          6/73
   70 FORMAT(I10,E15.4,A3,13A4)                                            6/73
      IF(STRECH.EQ.0.) STRECH=1.0                                          6/73
      WRITE(IOUT,75) STRECH, (MA(I), I=1,14)                               6/73
   75 FORMAT(16H STRETCH FACTOR=,E15.4,/,2X,A3,13A4)
C
C CLEAR GEOMETRY ARRAYS
      DO 76 I=1,21
   76 IK(I)=IJ(I)
      LFPD=4000
       LMA=4000
      DO 80 I=1,LFPD
   80 FPD(I)=0.
      DO 85 I=1,LMA
   85 MA(I)=0                                                              6/73
      IF(IPRINT.EQ.2) GO TO 87
C                                                                          6/73
      WRITE(IOUT,86)
```

```
MEMBER NAME  GENI                                                    6/73
   86 FORMAT(//50X,9HBODY DATA)                                      6/73
   87 N=1
      L2=2
      M=0
   90 L1=L2&1
      LF=L1
      L2=L1&5
      M=M&1
      READ( IN,100) ITYPE,IALP,(FPD(I),I=L1,L2)
  100 FORMAT(2X,A3,1X,A4,6E10.3)                                     6/73
      IF(STRECH.EQ.1.0) GO TO 102                                    6/73
      DO 101 I=L1,L2                                                 6/73
  101 FPD(I)=FPD(I)*STRECH                                           6/73
  102 IF(IPRINT.EQ.2) GO TO 107
      WRITE(IOUT,105) ITYPE,IALP,(FPD(I),I=L1,L2),L1,M
  105 FORMAT(2X,A3,1X,A4,6E15.7,I5,2H B,I4)                          6/73
  107 IF(ITYPE.EQ.IRAW)ITYPE=IWED
      DO 110 I=1,11
      IF( ITY(I).EQ.ITYPE ) GO TO 120
  110 CONTINUE
      STOP
  120 IF( I.EQ.11) GO TO 160
      MORE=IBIAS(I)
      MA(N)=IALP
      MA(N&2)=I
      MA(N&6)=L1-2
      N=N&7
      IF( MORE.LE.0) GO TO 150
      L1=L2&1
      L2=L1&MORE-1
      READ(IN,130) (FPD(I),I=L1,L2)
  130 FORMAT( 10X,6E10.3)                                            6/73
      IF(STRECH.EQ.1.0) GO TO 135                                    6/73
      L2X=L2                                                         6/73
      IF(ITYPE.EQ.ITY(1)) L2X=L2X-6
CC    FOR ARBS, THE LAST 6 ENTRIES ARE FACE DESCRIPTIONS, NOT DIMENSIONS
      IF(ITYPE.EQ.ITY(10))L2X=L2X-1
C------LAST TEC ENTRY IS RATIO. N AND A NORMALIZED BELOW.            6/73
      DO 132 I=L1,L2X                                                6/73
  132 FPD(I)=FPD(I)*STRECH                                           6/73
  135 IF(IPRINT.EQ.2) GO TO 150
      WRITE(IOUT,140) ( FPD(I),I=L1,L2)
  140 FORMAT(10X,6E15.7)
  150 L2=L2&2
      IF(ITYPE.EQ.ITY(1)) CALL GARB(FPD(LF),IERR)
      IF(ITYPE.NE.ITY(10))GO TO 90
C------NORMALIZE N  AND A VECTORS.
      L2=L2&9
      LF=LF-2
      Z=1./SQRT(FPD(LF&8)**2&FPD(LF&9)**2&FPD(LF&10)**2)
      FPD(LF&8)=Z*FPD(LF&8)
      FPD(LF&9)=Z*FPD(LF&9)
      FPD(LF&10)=Z*FPD(LF&10)
      Z=1./SQRT(FPD(LF&11)**2&FPD(LF&12)**2&FPD(LF&13)**2)
      FPD(LF&11)=Z*FPD(LF&11)
      FPD(LF&12)=Z*FPD(LF&12)
      FPD(LF&13)=Z*FPD(LF&13)
C------PRECOMPUTE TEC CONSTANTS AND STORE IN FPD ARRAY.
```

MEMBER NAME GENI

```
      FPD(LF&17)=FPD(LF&12)*FPD(LF&10)-FPD(LF&13)*FPD(LF&9)
      FPD(LF&18)=FPD(LF&13)*FPD(LF&8)-FPD(LF&11)*FPD(LF&10)                     K3
      FPD(LF&19)=FPD(LF&11)*FPD(LF&9)-FPD(LF&12)*FPD(LF&8)
      FPD(LF&20)=FPD(LF&5)*FPD(LF&11)&FPD(LF&6)*FPD(LF&12)                      HDA
     1           &FPD(LF&7)*FPD(LF&13)
      FPD(LF&21)=FPD(LF&5)*FPD(LF&8)&FPD(LF&6)*FPD(LF&9)                        HDN
     1           &FPD(LF&7)*FPD(LF&10)
      FPD(LF&22)=FPD(LF&5)*FPD(LF&17)&FPD(LF&6)*FPD(LF&13)&                     HDK
     1            FPD(LF&7)*FPD(LF&19)                                          TAU
      FPD(LF&23)=(FPD(LF&14)/FPD(LF&15))**2
      D=1.0/FPD(LF&16)   - 1.0
      FPD(LF&24)=FPD(LF&23)*(FPD(LF&15)*D)**2
      FPD(LF&25)=FPD(LF&2)*FPD(LF&8)&FPD(LF&3)*FPD(LF&9)                        VDN
     1           &FPD(LF&4)*FPD(LF&10)
      GO TO 90
  160 NUMB=N/7
      WRITE(IOUT,170) NUMB,N,L2
  170 FORMAT(20H NUMBER OF BODIES    ,I5,
     1      / 20H LENGTH OF MA-ARRAY  ,I5,
     2      / 20H LENGTH OF FPD-ARRAY ,I5 )
C
C END OF BODY DATA
C                                                                               6/71
      IF(IPRINT.NE.1) GO TO 176
      WRITE(IOUT,172)
  172 FORMAT( 50X,9HFPD ARRAY)
      DO 174  I=1,L2,5
      K=I&4
  174 WRITE( IOUT,175) I,(FPD(J),J=I,K),K
  175 FORMAT( I5,5E20.7,I5 )                                                    6/71
  176 IF(IPRINT.EQ.2) GO TO 182
      WRITE(IOUT,180)
  180 FORMAT(//50X,11HREGION DATA)                                              6/71
  182 IRTRU=0
      IR=0
      IR1=0
  190 K=1
      READ(IN,200) IALP,( IBIAS(I),ITY(I),I=1,9 )
      IF(IALP.NE.IBL)IR1=IR1&1                                                  6/71
      IF(IPRINT.EQ.2) GO TO 206
      WRITE(IOUT,205) IALP,( IBIAS(I),ITY(I),I=1,9),IR1
  200 FORMAT(2X,A3,5X,9(A2,I5))
  205 FORMAT(2X,A3,5X,9(A2,I5),2X,1HR,I3)                                       6/71
  206 IF( IALP.EQ.IBL ) GO TO 220
      IF(IRTRU.NE.0)IRHA(IRTRU)=IR
      IF( IALP.EQ.IEND) GO TO 300
      IRTRU=IRTRU&1
  210 IR=IR&1
      LOCREG(IR)=N
      NUMBOD(IR)=1
      IROR(IR)=IRTRU
      MA(N)=IALP
      MA(N&1)=ITY(K)
      MA(N&2)=7*IABS(ITY(K))-6
      N=N&5
      K=K&1
      IF(K.GT.9)GO TO 190
  220 KL=K
```

```
MEMBER NAME  GENI
      DO  230  I=KL,9
      K=I
      IF( IBIAS(I).EQ.IOR) GO TO 210
      IF( ITY(I).EQ.0 )  GO TO  190
      MA(N)=ITY(K)
      MA(N&1)=      7*IABS(ITY(K))-6
      N=N&4
      NUMBOD(IR)=NUMBOD(IR)&1
  230 CONTINUE
      GO TO 190
  300 CONTINUE
      LDATA=N
      NUMR=IR
      WRITE(IOUT,310) IRTPU,IR,N
  310 FORMAT(  24H NUMBER OF INPUT REGIONS  ,I5
     1        / 24H NUMBER OF  CODE REGIONS  ,I5
     2        / 24H LENGTH OF INTEGER ARRAY  ,I5)
      IF(IPRINT.NE.1) GO TO 327                                        6/71
      WRITE(IOUT,315)
  315 FORMAT(/54HOCODE REGION LOC. OF REG. DATA NO. OF BODIES INPUT REG)
      WRITE( IOUT,317) ( I,LOCREG(I),NUMBOD(I),IROR(I),I=1,NUMR)
  317 FORMAT( I12,I18,I14,I10)
      WRITE( IOUT,320 )
  320 FORMAT( 50X,10H MA -ARRAY)
      DO  325  I=1,LDATA,10
      K=I&9
  325 WRITE( IOUT,326) I,(MA(J),J=I,K),K
  326 FORMAT( I5,I10,9I5,I10)
C
C END OF REGION DATA
C
  327 RETURN                                                           6/71
      END
```

```
MEMBER NAME  GETIR
      SUBROUTINE GETIR
C     FINDS REGION NUMBER FOR POINT XB
C     NEEDS GG AND GENI
      LOGICAL LOGG
      DIMENSION IRS(10)
      COMMON / GEOM2 / LOCREG(200),NUMBOD(200),IROR(200),MA(4000),
     1                FPD(4000),LDATA,NUMB,NUMR
      COMMON/PAREM /XB(3) ,WB(3) ,E     ,IR    ,T     ,IDET  ,F     ,    PAREM
     1             NHIST ,WC    ,J12345,WP(3) ,XP(3) ,EPRIM ,ATWT  ,    PAREM
     2             NCDB  ,CSTHT ,U     ,LCHI  ,IATWT ,IERR  ,IDBG       PAREM
     3,            IRPRIM,NASC  ,LSURF ,NBO   ,LRI   ,LRO   ,RIN   ,    PAREM
     4             ROUT  ,KLOOP ,LOOP  ,ITYPE ,PINF  ,NOA ,DIST
      COMMON/TAPE  /INT   ,IOUT  ,IEDT  ,INT1  ,INT2  ,IAGG,ITRAN,INTER TAPE
      KLOOP=KLOOP&1
      L=0
      DO 700 IRP=1,NUMR
      N=LOCREG(IRP)&1
      NUM=NUMBOD(IRP)*4&N-4
      DO 650 I=N,NUM,4
      NBO=MA(I)
      LOCAT=MA(I&1)
      ITYPE=MA(LOCAT&2)
      IF(ITYPE.NE.10)CALL GG(LOCAT)
      IF(ITYPE.EQ.10)CALL GGTEC(LOCAT)
      LOGG=(ROUT.LE.0.).OR.(RIN.GT.0.)
      IF(NBO)620,650,630
  620 LOGG=.NOT.LOGG
  630 IF(LOGG)GOTO700
  650 CONTINUE
      IR=IROR(IRP)
      IF(L.NE.0.AND.IR.EQ.IRS(L))GOTO700
      L=L&1
      IRS(L)=IR
      IF(L.GE.10)GOTO300
  700 CONTINUE
      IF(L-1)200,5000,300
  300 WRITE(IOUT,101)XB,L,(IRS(K),K=1,L)
  101 FORMAT(19HOREGION OVERLAP FOR 3E20.7/I3,8H REGIONS 10I10)
      GO TO 5000
  200 IR=0
      WRITE(IOUT,100) XB
  100 FORMAT(21HOREGION NOT FOUND FOR 3E20.7)
 5000 RETURN
      END
```

```
MEMBER NAME  GG
      SUBROUTINE GG(LOCAT)
      COMMON/JAN233 / DATEX
CC        WEDGE BUG CORRECTED BY MOC
      DIMENSION ASQ(3),PV(3),G(3)
      DIMENSION VHAB(12),V(3),H(3),A(3),B(3)
      DIMENSION  IBOX(3)
      COMMON / GEOM2 / LOCREG(200),NUMBOD(200),IROR(200),MA(4000),
     1                 FPD(4000),LDATA,NUMB,NUMR
      COMMON/GEOM3/ EPS,IRHA(200)
      COMMON/PAREM /XB(3) ,WB(3) ,E      ,IR     ,T      ,IDET  ,F    ,      PAREM
     1              NHIST ,WC    ,J12345,WP(3)  ,XP(3)  ,EPRIM ,ATWT ,      PAREM
     2              NCDB  ,CSTHT ,U     ,LCHI   ,IATWT  ,IERR  ,IDBG        PAREM
     3,             IRPRIM,NASC  ,LSURF ,NBO    ,LRI    ,LRO   ,RIN  ,      PAREM
     4              ROUT  ,KLOOP ,LOOP  ,ITYPE  ,PINF   ,NOA ,DIST
      COMMON/TAPE  /INT   ,IOUT  ,IEOT  ,INT1   ,INT2   ,IAGG,ITRAN,INTERA
      EQUIVALENCE (VHAB(1),V(1)),(VHAB(4),H(1)),(VHAB(7),A(1)),(VHAB(10)
     1,B(1))
      DATA IBOX/2,1,3/
      L=LOCAT
      LOOP=MA(L&1)
      K=MA(L&6)
      IF( LOOP.NE.KLOOP ) GO TO 1000
      LRI=MA(L&3)
      LRO=MA(L&4)
      RIN=FPD(K)
      ROUT=FPD(K&1)
      GO TO 2005
 1000 RIN=PINF
      ROUT=-PINF
      IF( (ITYPE.LT.1).OR.(ITYPE.GT.9) ) GO TO 2011                     16/71
      MA(L&1)=KLOOP
      GO TO (1100,1200,1400,1300,1400,1600,1700,1800,1700 ),ITYPE
C             ARB  SPH  RCC  REC  TRC  ELL  BOX  WED  RPP
C
C ARB  ARBITRARY POLYHEDRON
C
 1100 DMIN=FPD(K&27)*1.0E-6
      NSIDE= FPD(K&26)
      I1=K&2
      I2=I1& 4*NSIDE -1
      L=0
      DO  1190  I=I1,I2,4
      DX=   FPD(I)*XB(1) & FPD(I&1)*XB(2)&FPD(I&2)*XB(3) & FPD(I&3)
      L=L&1
      DY=  -FPD(I)*WB(1)  -FPD(I&1)*WB(2)-FPD(I&2)*WB(3)
      IF(  ABS(DY ).LE.1.0E-6 )  GO TO 1190
      DZ=DX/DY
      PV(1)= XB(1) & DZ*WB(1)
      PV(2)= XB(2) & DZ*WB(2)
      PV(3)= XB(3) & DZ*WB(3)
      DO   1120  J=I1,I2,4
      IF( J.EQ.I) GO TO  1120
      DX=FPD(J)*PV(1)&FPD(J&1)*PV(2)&FPD(J&2)*PV(3)& FPD(J&3)
      IF(  DX.GE.0. ) GO TO  1120
      IF( -DX.LT.DMIN ) GO TO  1120
      GO  TO 1190
    0 CONTINUE
      IF(DZ.GT.ROUT) GO TO 1140
```

```
MEMBER NAME  GG
      RIN=DZ
      LRI=L
      GO TO 2000
 1140 RIN=ROUT
      LRI=LRO
      LRO=L
      ROUT=DZ
      IF(RIN.GT.-PINF) GO TO 2000
 1190 CONTINUE
      GO TO 2000
C
C
 1200 CONTINUE
      DX= XB(1) -FPD(K&2)
      DY= XB(2) -FPD(K&3)
      DZ= XB(3) -FPD(K&4)
      Y= DX*WB(1) & DY*WB(2) & DZ*WB(3)
      C= DX*DX & DY*DY & DZ*DZ -FPD(K&5)**2
      DX=Y*Y- C
      IF( DX.LT.0.) GO TO 2000
      DY=SQRT(DX)
      RIN= -Y-DY
      ROUT=-Y&DY
      LRI=1
      LRO=1
      GO TO 2000
C
 1300 DO 1301 I=1,12
      J=K&1&I
 1301 VHAB(I)=FPD(J)
      RIN=-PINF
      ROUT=PINF
      LRO=0
      LRI=0
C
C4    COMPUTE DOT PRODUCTS OF A.A AND B.B
C
      AA=A(1)*A(1)&A(2)*A(2)&A(3)*A(3)
      BB=B(1)*B(1)&B(2)*B(2)&B(3)*B(3)
C
C5    COMPUTE (V-XB) FOR X,Y,Z COORDINATES
C
      V1XB1=V(1)-XB(1)
      V2XB2=V(2)-XB(2)
      V3XB3=V(3)-XB(3)
C
C6    TRANSFORM XV(X,Y,Z) TO THE COORDINATES OF THE REC
C
      VPA=V1XB1*A(1)&V2XB2*A(2)&V3XB3*A(3)
      VPB=V1XB1*B(1)&V2XB2*B(2)&V3XB3*B(3)
C
C7    TRANSFORM WB(X,Y,Z) TO THE COORDINATES OF THE REC
C
      WBA=WB(1)*A(1)&WB(2)*A(2)&WB(3)*A(3)
      WBB=WB(1)*B(1)&WB(2)*B(2)&WB(3)*B(3)
      WBAWBA=WBA*WBA
      WBBWBB=WBB*WBB
      AAAA=AA*AA
```

```
MEMBER NAME  GG
      BBBB=BB*BB
      AMBD=WBA*VPA*BBBB&WBB*VPB*AAAA
      UM=BBBB*VPA*VPA&AAAA*VPB*VPB-AAAA*BBBB
      DEN=WBAWBA*BBBB&WBBWBB*AAAA
      IF(ABS(DEN).LE.1.0E-6)GOTO 10
      AMBDA=AMBD/DEN
      UMU=UM/DEN
      DISC=AMBDA**2-UMU
      IF(DISC.LE.0.)GOTO 300
C
C8    COMPUTE THE INTERSECT POINTS ON THE QUADRATIC SURFACE
C
      SD=SQRT(DISC)
      R1=AMBDA-SD
      R2=AMBDA&SD
      GOTO 20
   10 R1=-PINF
      R2=PINF
   20 HH=H(1)*H(1)&H(2)*H(2)&H(3)*H(3)
      WH=WB(1)*H(1)&WB(2)*H(2)&WB(3)*H(3)
      VPH=V1XB1*H(1)&V2XB2*H(2)&V3XB3*H(3)
C
C9    DETERMINE IF RAY PARALLEL TO PLANAR SURFACES
C
      IF(WH)40,70,50
   40 IF(VPH.GE.0.)GOTO 300
C
C10   COMPUTE THE INTERSECT POINTS ON THE PLANAR SURFACES
C
      CP=VPH/WH
      CM=(VPH&HH)/WH
      LCP=1
      LCM=2
      GOTO 100
   50 VPHHH=VPH&HH
      IF(VPHHH.LE.0.)GOTO 300
      CP=VPHHH/WH
      CM=VPH/WH
      LCM=1
      LCP=2
      GOTO 100
   70 CP=PINF
      CM=-CP
  100 IF(CM.GT.R1)GOTO 110
C
C11   RIN FOR THE QUADRATIC SURFACE
C
      RIN=R1
      LRI=3
      GOTO 120
C
C12   RIN FOR A PLANAR SURFACE
C
  110 RIN=CM
      LRI=LCM
  120 IF(CP.LE.R2)GOTO 130
C
C13   ROUT FOR THE QUADRATIC SURFACE
```

```
MEMBER NAME  GG
C
      ROUT=R2
      LRO=3
      GOTO 200
C14   ROUT FOR A PLANAR SURFACE
C
  130 ROUT=CP
      LRO=LCP
  200 IF(ROUT.LE.RIN)GOTO300
      GOTO(210,210,220),LRO
C
C15   DETERMINE IF ROUT OF PLANAR SURFACE OCCURS WITHIN ELLIPTIC
C     CROSS-SECTION
C
  210 F1=DEN*ROUT**2-2.*AMBD*ROUT&UM
      IF(F1)250,250,300
C
C16   DETERMINE IF ROUT OF QUADRATIC OCCURS BETWEEN PLANAR SURFACES
C
  220 F1=ROUT*WH-VPH
      IF(F1)300,250,230
  230 IF(F1.GT.HH)GOTO 300
  250 GOTO(260,260,270),LRI
C
C17   DETERMINE IF RIN OF PLANE WITHIN ELLIPTIC CROSS SECTION
C
  260 F1=DEN*RIN**2-2.*AMBD*RIN&UM
      IF(F1.GT.0.0)GO TO 300
      GO TO 2000
C
C18   DETERMINE IF RIN OF QUADRATIC SURFACE BETWEEN PLANAR SURFACES
C
  270 F1=RIN*WH-VPH
      IF(F1)300,2000,280
  280 IF(F1.LE.HH)GOTO 2000
C
C19   RAY MISSES BODY
C
  300 RIN=PINF
      ROUT=-PINF
      LPT=0
      LRO=0
      GO TO 2000
C
C
 1400 RB=FPD(K&8)
      RT=FPD(K&9)
      IF(ITYPE.EQ.3)RT=RB
      DX=FPD(K&2)-XB(1)
      DY=FPD(K&3)-XB(2)
      DZ=FPD(K&4)-XB(3)
      H1=FPD(K&5)
      H2=FPD(K&6)
      H3=FPD(K&7)
      INTSEC=0
      INTR1=0
      INTR2=0
```

MEMBER NAME GG

```
      PVPV=DX**2 & DY**2 & DZ**2
      VPW=DX*WB(1) & DY*WB(2) & DZ*WB(3)
      WH= H1*WB(1) & H2*WB(2) & H3*WB(3)
      VPH=H1*DX & H2*DY & H3*DZ
      HH =H1**2 & H2**2 & H3**2
      RTRB=RT-RB
      RRR= RB-RTRB/HH*VPH
      VPHHH=VPH & HH
      UM= HH*(PVPV-RRR**2) -VPH**2
      AMBD=HH*VPW -WH*(VPH-RTRB*RRR)
      DEN= HH -WH**2*(1.0&RTRB**2/HH)
      IF(ABS(DEN).GT.1.0E-6 ) GO TO 1420
      IF(RTRB.EQ.0)GO TO 1470
      R2=UM/(2.0*AMBD)
      F1=R2*WH-VPH
      IF(F1.LT.0.0)GO TO 1470
      IF((F1-HH).GT.0.0)GO TO 1470
      INTSEC=INTSEC&1
      IF(WH.LE.0.0)GO TO 1405
      IF(RTRB)1410,1410,1415
 1405 IF(RTRB.LE.0.0)GO TO 1415
 1410 LRO=3
      ROUT=R2
      GO TO 1480
 1415 LRI=3
      RIN=R2
      INTSEC=INTSEC&1
      GO TO 1472
 1420 AMBDA=AMBD/DEN
      DISC=AMBDA**2 -UM/DEN
      IF(DISC)1498,1470,1422
 1422 SD=SQRT(DISC)
      R1=AMBDA-SD
      R2=AMBDA&SD
      F1=R2*WH-VPH
      IF(F1.LT.0.0)GO TO 1424
      IF((F1-HH).GT.0.0)GO TO 1424
      INTR2=INTR2&1
 1424 F1=R1*WH-VPH
      IF(F1.LT.0.0)GO TO 1426
      IF( (F1-HH).GT.0.) GO TO 1426
      INTR1=INTR1&1
      GO TO 1430
 1426 IF(INTR2.EQ.0)GO TO 1470
      ROUT=R2
      RIN=R2
      LRO=3
      LRI=3
      INTSEC=INTSEC&1
      GO TO 1470
 1430 IF(INTR2.GT.0)GO TO 1432
      ROUT=R1
      RIN=R1
      LRO=3
      LRI=3
      INTSEC=INTSEC&1
      GO TO 1470
 1432 IF(R1-R2)1434,1498,1436
```

```
MEMBER NAME  GG
 1434 RIN=R1
      ROUT=R2
      LRO=3
      LRI=3
      GO TO 1496
 1436 RIN=R2
      ROUT=R1
      LRO=3
      LRI=3
      GO TO 1496
 1470 IF(WH)1472,1498,1480
 1472 IF(VPH.GE.0.0)GO TO 1498
      CP=VPH/WH
      F1=CP**2 -2.0*CP*VPW & PVPV -RB**2
      IF(F1.GT.0.0)GO TO 1474
      INTSEC=INTSEC&1
      ROUT=CP
      LRO=1
      IF(INTSEC.GE.2)GO TO 1496
 1474 CM=VPHHH/WH
      F1=CM**2-2.0*((VPW&WH)*CM-VPH)&HH&PVPV-RT**2
      IF(F1.GT.0.0)GO TO 1498
      RIN=CM
      LRI=2
      GO TO 1496
 1480 IF(VPHHH.LT.0.0)GO TO 1498
      CP=VPHHH/WH
      F1=CP**2-2.0*((VPW&WH)*CP-VPH)&HH&PVPV-RT**2
      IF(F1.GT.0.0)GO TO 1486
      INTSEC=INTSEC&1
      ROUT=CP
      LRO=2
 1486 IF(INTSEC.GT.1)GO TO 1496
      CM=VPH/WH
      F1=CM**2-2.0*CM*VPW&PVPV-RB**2
      IF(F1.GT.0.0)GO TO 1498
      RIN=CM
      LRI=1
 1496 GO TO 2000
 1498 RIN=0.0
      ROUT=-PINF
      GO TO 2000
C
 1600 CONTINUE
      A1=0.
      A2=0.
      B1=0.
      B2=0.
      JA=K&1
      DO   1610  J=1,3
      JA=JA&1
      DX=XB(J)-FPD(JA)
      A1=A1& DX*WB(J)
      B1=B1& DX*DX
      DX=XB(J)-FPD(JA&3)
      A2=A2& DX*WB(J)
 1610 B2=B2& DX*DX
      A1=2.0*A1
```

```
MEMBER NAME  CG
      A2=2.0*A2
      C=FPD(K&8)
      C2=2.0*C
      X=(A2-A1)/C2
      Y=( C**2&B2-B1)/C2
      ALAMD=X*X-1.
      ALAM1=( X*Y-A2*0.5)/ALAMD
      U=(Y*Y-B2)/ALAMD
      C=ALAM1*ALAM1-U
      IF(C.LT.0 ) GO TO 2000
      C=SQRT(C)
      RIN= -ALAM1-C
      ROUT=-ALAM1&C
      LRI=1
      LRO=1
      GO TO 2000
 1700 CONTINUE
      RIN =-PINF
      ROUT=&PINF
      DO 1798  I=1,3
      IF(ITYPE.LT.9)GO TO 1705
      JV=K&2*I&1
      X =FPD(JV)-FPD(JV-1)
      VP= FPD(JV-1)-XB(I)
      W =WB(I)
      GO TO 1711
 1705 JV=K&1
      X=0.
      VP=0.
      W=0.
      JA=JV&3*I
      DO  1710  J=1,3
      JV=JV&1
      JA=JA&1
      DX=FPD(JA)
      VP=VP& (FPD(JV)  -XB(J))*DX
      W=W& WB(J)*DX
 1710 X=X&DX*DX
 1711 IF( W ) 1720,1712,1740
 1712 IF(-VP.LT.0) GO TO 1799
      IF(-VP-X )1798,1798,1799
 1720 DY=VP/W
      LO=2*IBOX(I)-1
      IF(DY.LE.0 ) GO TO 1799
      DZ=(VP&X)/W
      LI=LO&1
      GO TO 1760
 1740 DY=(VP&X)/W
      LO=2*IBOX(I)
      IF( DY.LE.0) GO TO 1799
      DZ=VP/W
      LI=LO-1
 1760 IF(ROUT.LE.DY) GO TO 1780
      ROUT=DY
      LRO=LO
 1780 IF( RIN.GE.DZ) GO TO 1798
      RIN=DZ
      LRI=LI
```

MEMBER NAME GG

```
 1798 CONTINUE
      GO TO 2000
 1799 R IN=PINF
      ROUT=-PINF
      GO TO 2000
C
 1800 R IN=-PINF
      ROUT=PINF
      CM=-P INF
      CP=PINF
      L =0
      L 1=0
      KK=0
      LRI=0
      LRO=C
      DX= XB(1)- FPD(K&2)
      DY= XB(2)- FPD(K&3)
      DZ= XB(3)- FPD(K&4)
      DO 1830  I=1,3
      JV=K&2&3*I
      ASQ(I)=FPD(JV)**2 & FPD(JV&1)**2 & FPD(JV&2)**2
      PV(I)=DX*FPD(JV) & DY*FPD(JV&1) & DZ*FPD(JV&2)
      G(I)=WB(1)*FPD(JV)&WB(2)*FPD(JV&1) & WB(3)*FPD(JV&2)
      IF( I.EQ.3 ) GO TO 1801
      IF( G(I) )  1810,1811,1860
 1810 IF(-PV(I).GE.0) GO TO 1840
      TEMP=-PV(I)/G(I)
      IF( TEMP.GE.CP) GO TO 1830
      CP=TEMP
      L =I
      IF( I.GT.1 ) GO TO 1850
      LRO=3
      GO TO 1830
 1850 LRO=1
      GO TO 1830
 1860 IF(-PV(I).LE.0 ) GO TO 1830
      TEMP=-PV(I)/G(I)
      IF(TEMP.LE.CM) GO TO 1830
      CM=TEMP
      KK=I
      LRI=3
      IF( I.EQ.1) GO TO 1830
      LRI=1
      GO TO 1830
 1811 IF( PV(I).LE.0.) GO TO    1881
      IF( PV(I).GE.ASQ(I)) GO TO 1881
 1830 L 1=L 1&I
 1801 IF( G(3) ) 1815,1821,1823
 1815 TEMP=-PV(3)&ASQ(3)
      IF(TEMP.GE.0.) GO TO 1818
      TEMP=TEMP/G(3)
      IF( TEMP.LE.CM) GO TO 1819
      CM=TEMP
      KK=3
      LRI=6
 1818 IF(-PV(3).GE.0.)  GO TO 1840
 1819 TEMP=-PV(3)/G(3)
```

```
MEMBER NAME  GG
      IF(TEMP.GE.CP) GO TO 1829
      CP=TEMP
      L=3
      LRO=5
      GO TO 1829
18    IF(PV(3).LE.0.) GO TO 1840
      IF(PV(3)-ASQ(3)) 1829,1829,1840                                    C1/73
1823  IF(-PV(3).LE.0.) GO TO 1826                                        C1/73
      TEMP=-PV(3)/G(3)                                                   C1/73
      IF(TEMP.LE.CM) GO TO 1826
      CM=TEMP
      KK=3
      LRI=5
1826  TEMP=-PV(3)&ASQ(3)
      IF(TEMP.LE.0) GO TO 1840
      TEMP=TEMP/G(3)
      IF( TEMP.GE.CP ) GO TO 1829
      CP=TEMP                                                            01/73
      L=3
      LRO=6
1829  AG=ASQ(2)*G(1) & ASQ(1)*G(2)
      PV4=PV(1)*ASQ(2) & PV(2)*ASQ(1)
      TOP=ASQ(1)*ASQ(2)-PV4
      IF(AG) 1831,1835,1833
1831  TEMP=TOP/AG
      IF(TEMP.LE.CM) GO TO 1838
      CM=TEMP
      KK=4
      LRI=2
      GO TO 1838
1833  IF(TOP.LT.0.) GO TO 1840
      TEMP=TOP/AG
      IF( TEMP-CP) 1837,1838,1838
1835  IF(PV4.LE.0.) GO TO 1840
      IF(-TOP) 1838,1840,1840
1837  CP=TEMP
      L=4
      LRO=2
1838  IF(L&KK.LE.0) GO TO 1840
      ROUT=CP
      RIN=CM
1840  CONTINUE
      IF( (ROUT.LT.PINF).AND.(ROUT.GT.0.).AND.(ROUT.GT.RIN)) GO TO 2000
1881  ROUT=-PINF
      RIN=PINF
      LRI=0
      LRO=0
      GO TO 2000

2000  MA(LOCAT&3)=LRI
      MA(LOCAT&4)=LRO
      IF(ROUT.LT.RIN) ROUT=-PINF                                         F
      IF(ROUT.EQ.RIN) ROUT=ROUT*(1.&EPS*10.0)
      EPD(K)=RIN
      EPD(K&1)=ROUT
2005  RETURN
2C    FORMAT(13H IN GG ITYPE=,I5,5X,3HIR=,I5,5X,4HNBO=,I5)               6/71
2011  WRITE(6,2010)ITYPE,IR,NBO                                          6/71
```

← for $r_{in}=r_{out}$, bump r_{out} by $(1+10\epsilon)$

```
MEMBER NAME  GG
      STOP
      END
```

```
MEMBER NAME  GGTEC
      SUBROUTINE GGTEC(LOCAT)
C------TREAT TEC BODY.
      COMMON / GEOM2 / LOCREG(200),NUMBOD(200),IROR(200),MA(4000),
     1                 FPD(4000),LDATA,NUMB,NUMR
      COMMON/GEOM3/EPS,IRHA(200)
      COMMON/PAREM /XB(3) ,WB(3) ,E      ,IR     ,T      ,IDET  ,F     ,      PAREM
     1              NHIST ,WC    ,J12345,WP(3) ,XP(3) ,EPRIM ,ATWT  ,      PAREM
     2              NODB  ,CSTHT ,U     ,LCHI  ,IATWT ,IERR  ,IDBG         PAREM
     3,             IRPRIM,NASC  ,LSURF ,NBO   ,LRI   ,LRO   ,RIN   ,      PAREM
     4              ROUT  ,KLOOP ,LOOP  ,ITYPE ,PINF  ,NOA ,DIST
      EQUIVALENCE(WB(1),WB111),(WB(2),WB222),(WB(3),WB333),(K,LF)
      EPT=10.0*EPS
      L=LOCAT
      LOOP=MA(L&1)
      K=MA(L&6)
      IF( LOOP.NE.KLOOP ) GO TO 1000
      LRI=MA(L&3)
      LRO=MA(L&4)
      RIN=FPD(K)
      ROUT=FPD(K&1)
      GO TO 2005
 1000 RIN=PINF
      ROUT=-PINF
      MA(L&1)=KLOOP
      D1=FPD(LF&2)-XB(1)
      D2=FPD(LF&3)-XB(2)
      D3=FPD(LF&4)-XB(3)
      BOC4      =D1*FPD(LF&8)&D2*FPD(LF&9)&D3*FPD(LF&10)                  DDN
       DDA      =D1*FPD(LF&11)&D2*FPD(LF&12)&D3*FPD(LF&13)                DDA
      HDA=FPD(K&20)
      BOC8=D1*FPD(K&17)&D2*FPD(K&18)&D3*FPD(K&19)                         DDK
      TAU=FPD(K&23)
      WDA=WB111*FPD(K&11) & WB222*FPD(K&12) & WB333*FPD(K&13)
      WDN=WB111*FPD(K&8 ) & WB222*FPD(K&9 ) & WB333*FPD(K&10)
      WDK=WB111*FPD(K&17)  & WB222*FPD(K&18)  & WB333*FPD(K&19)
      IF(WDN.GT.EPT.OR.WDN.LT.-EPT) GO TO 1310
      GAMMA=-BOC4/FPD(K&21)
      IF(GAMMA.LT.0.0.OR.GAMMA.GT.1.)GO TO 2000
      R1=FPD(K&14)
      R2=FPD(K&15)
      R3=R1/FPD(K&16)
      R4=R2/FPD(K&16)
      TM2=(GAMMA*(R3-R1)&R1)**2
      TMM2=(GAMMA*(R4-R2)&R2)**2
      V=GAMMA*HDA&DDA
      TT=GAMMA*FPD(K&22)&BOC8
      A=TMM2*WDA**2&TM2*WDK**2
      B=-(TMM2*WDA*V&TM2*WDK*TT)
      DISC=B**2-A*(TMM2*V**2&TM2*TT**2-TM2*TMM2)
      IF(DISC) 2000,1306,1305
 1305 DISC= SQRT(DISC)
 1306 RIN=(-B-DISC)/A
      ROUT=(-B&DISC)/A
      LRI=3
      LRO=3
      GO TO 2000
 1310 FLIPD=1.0
      IF(WDN.LT.0.0) GO TO 1315
```

MEMBER NAME GGTEC

```
      FLIPD=-1.0
      WDA=-WDA
      WDN=-WDN
      WDK=-WDK
 1315 ALPHA=FPD(K&21)/WDN
      BETA=BOC4/WDN
      A=(ALPHA*WDA-HDA)**2&TAU*(ALPHA*WDK-FPD(K&22))**2 - FPD(K&24)
      D=1.0/FPD(LF&16)   - 1.0
      BOC12      =-BOC8      *FPD(K&22)&D*FPD(LF&15)**2
      B=-ALPHA*BETA*WDA**2&WDA*(ALPHA*DDA&BETA*HDA)-DDA*HDA
     1  &TAU*(-ALPHA*BETA*WDK**2&ALPHA*WDK*BOC8
     2  &BETA*WDK*FPD(K&22)&BOC12)
      CC=(DDA-BETA*WDA)**2&TAU*((BOC8      -BETA*WDK)**2-FPD(K&15)**2)
      DISC=B**2-A*CC
      IF(DISC) 2000,1330,1320
 1320 DISC= SQRT(DISC)
 1330 IF(A.LT.EPT.AND.A.GT.-EPT) GO TO 1339
      IF(A) 1335,1339,1340
 1335 SIGMA1=(B-DISC)/A
      SIGMA2=(B&DISC)/A
      GO TO 1345
 1339 SIGMA1=CC/(B&B)
      SIGMA2=-PINF
      IF(SIGMA1) 2000,1345,1345
 1340 SIGMA1=(B&DISC)/A
      SIGMA2=(B-DISC)/A
 1345 SIGMAP=1.0E20
      IF(FPD( K&16).LT.1.00001) GO TO 1346
      SIGMAP=FPD( K&16)/(FPD( K&16)-1.0)
 1346 IF(SIGMAP.LT.SIGMA2) GO TO 2000
      IF(SIGMAP.GT.SIGMA1) GO TO 1347
      IF(SIGMA2.LE.0.0) GO TO 2000
      IF(SIGMA2-1.0) 1380,1380,1370
 1347 IF(SIGMA2.GT.1.0.OR.SIGMA1.LT.0.0) GO TO 2000
      IF(SIGMA1.LT.1.0) GO TO 1348
      IF(SIGMA2) 1370,1370,1375
 1348 IF(SIGMA2.GT.0.)GO TO 1360
      LRI=3
      LRO=1
      RIN=ALPHA*SIGMA1 & BETA
      ROUT=BETA
      GO TO 1390
 1360 LRI=3
      LRO=3
      RIN= ALPHA*SIGMA1 & BETA
      ROUT=ALPHA*SIGMA2 & BETA
      GO TO 1390
 1370 LRI=2
      LRO=1
      RIN=ALPHA & BETA
      ROUT=BETA
      GO TO 1390
 1375 LRI=2
      LRO=3
      RIN= ALPHA & BETA
      ROUT=ALPHA*SIGMA2 & BETA
      GO TO 1390
 1380 LRI=3
```

MEMBER NAME GGTEC

```
      LRO=1
      RIN=ALPHA*SIGMA2 & BETA
      ROUT=BETA
 1390 IF(FLIPD.GE.0.0) GO TO 2000
      RTP=RIN
      ITP=LRI
      RIN=-ROUT
      LRI=LRO
      ROUT=-RTP
      LRO=ITP
 2000 MA(LOCAT&3)=LRI
      MA(LOCAT&4)=LRO
      IF(ROUT.LT.RIN)ROUT=-PINF
      IF(ROUT.EQ.RIN)ROUT=ROUT*(1.&EPT)
      FPD(K)=RIN
      FPD(K&1)=ROUT
 2005 RETURN
      END
```

```
MEMBER NAME  GNORM
      SUBROUTINE GNORM                                                    NORM 100
      DIMENSION X(3),H(3),IBOX1(3),IBOX2(3)
      COMMON/GEOM2/LJ3REG(200),NUMBOD(200),IROR(200),MA(4000),
     1                FPD(4000), LDATA,NUMB,NUMR                          NORML 30
      COMMON/NORMAL/UN(3)
      COMMON/PAREM /XB(3) ,WB(3)  ,E      ,IR     ,T      ,IDET  ,F     , PAREM
     1                NHIST ,WC     ,J12345,WP(3) ,XP(3)  ,EPRIM ,ATWT  , PAREM
     2                NCDB  ,CSTHT  ,U      ,LCHI   ,IATWT  ,IERR  ,IDBG    PAREM
     3,               IRPRIM,NASC   ,LSURF  ,NBO    ,LRI    ,LRO   ,RIN   , PAREM
     4                ROUT  ,KLOOP  ,LOOP   ,ITYPE  ,PINF   ,NOA ,DIST
      COMMON/TAPE  /INT    ,IOUT   ,IEDT   ,INT1   ,INT2   ,IAGG,ITRAN,INTER TAPE
      DATA (IBOX1(I),I=1,3)/10,7,4/                                       NORM 11
      DATA (IBOX2(I),I=1,3)/4,10,7/                                       NORM 12
C                                                                         NORM 13
      LSUR=IABS(LSURF)                                                    NORM 14
      DO 25 I=1,3                                                         NORM 15
   25 XP(I)=XB(I)&DIST*WB(I)
      MASC=7*NASC
      K=MA(MASC)
      ITYPE=MA(MASC-4)
C           ARB   SPH   RCC   REC   TRC   ELL   BOX   WED   RPP   TEC
      GO TO (1100,1200,1400,1300,1400,1600,1700,1500,1900,1800),ITYPE
   99 FORMAT(31H ROUND-OFF ERROR IN NORMAL NBO=,I5,6HLSURF=,I5,3HXP=,     NORM 81
     1       3E12.5)
  999 WRITE(IOUT,99)NBO,LSURF,(XP(I),I=1,3)
      STOP 1
C  ARB                                                                    NORM 29
 1100 I=K&1&4*(LSUR -1)                                                   NORM 30
      DO 1110 J=1,3                                                       NORM 31
 1110 UN(J)=FPD(I&J)                                                      NORM 32
      GO TO 2000
C  SPH                                                                    NORM 33
 1200 DO 1210 I=1,3                                                       NORM 34
 1210 UN(I)=(XP(I)-FPD(K&1&I))/FPD(K&5)                                   NORM 35
      GO TO 2000
C------REC.
 1300 IF(LSUR.EQ.3)GO TO 1310
      K=K&4
      GO TO 1502
 1310 K=K&1
      AN=FPD(K&7)**2&FPD(K&8)**2&FPD(K&9)**2
      XMU=SQRT(AN-(FPD(K&10)**2&FPD(K&11)**2&FPD(K&12)**2))
      YMU=1./(FPD(K&4)**2&FPD(K&5)**2&FPD(K&6)**2)
      R1=0.0
      I=K&3
      DO 1320 J=1,3
 1320 R1=(XP(J)-FPD(K&J)) * FPD(I&J)*YMU&R1
      J=I&3
      DO 1330 JJ=1,3
      X(JJ)= FPD(K&JJ)&R1*FPD(I&JJ)
 1330 H(JJ)= FPD(J&JJ)*XMU
      DO 1340 JJ = 1,3
      UN(JJ) = X(JJ) &H(JJ)
 1340 H(JJ) = X(JJ)-H(JJ)
      DO 135C JJ=1,3
 1350 X(JJ) = XP (JJ) - UN(JJ)
      AN = 2.0 * SQRT ( AN / (X(1)**2& X(2)**2 & X(3)**2))
      DO 1360 JJ=1,3
```

```
MEMBER NAME   GNORM
 1360 UN(JJ) = H(JJ) - UN(JJ) - AN*X(JJ)
      GO TO 1505
C   TRC AND RCC
 1400 H2=0.0
      DO 1410 I=1,3
      H(I)=FPD(K&4&I)
 1410 H2=H2&H(I)**2
      GO TO (1415,1415,1435), LSUR
 1415 H2=SQRT(H2)
      DO 1420 I=1,3
 1420 UN(I)=H(I)/H2
      GO TO 2000
 1435 DO 1440 I=1,3
 1440 X(I)=XP(I)-FPD(K&1&I)
      XMU=0.0
      A2=0.0
      DO 1445 I=1,3
      J=MOD(I,3)&1
      A2=A2&(H(I)*X(J)-H(J)*X(I))**2
 1445 XMU=XMU&FPD(K&4&I)*X(I)
      R =FPD(K&8)-FPD(K&9)
      IF(ITYPE.EQ.3) R=0.0
      B1=R/SQRT(H2*(H2&R**2))
      B2=1.0/SQRT(A2*(H2&R**2))
      DO 1450 I=1,3
 1450 UN(I)=B2*(X(I)*H2-H(I)*XMU)&B1*H(I)
      GO TO 2000
C---- WED (= RAW).
 1500 IF(LSUR.EQ.2)GO TO 1550
      IF(LSUR.EQ.3)K=K&4
      IF(LSUR.EQ.1)K=K&7
      IF(LSUR.GT.4)K=K&10
 1502 DO 1503 J=1,3
 1503 UN(J)=FPD(K&J)
 1505 AN=1./SQRT(UN(1)**2&UN(2)**2&UN(3)**2)
      DO 1510 J=1,3
 1510 UN(J)=AN*UN(J)
      GO TO 2000
 1550 K=K&4
      I=K&3
      XMU= FPD(K&1)**2&FPD(K&2)**2&FPD(K&3)**2
      AN=  FPD(I&1)**2&FPD(I&2)**2&FPD(I&3)**2
      DO 1560 J=1,3
 1560 UN(J)=XMU*FPD(I&J)& AN*FPD(K&J)
      GO TO 1505
C   ELL
 1600 DX=0.0
      AN=0.0
      C2=FPD(K&8)**2/4.0
      DO 1620 I=1,3
      IA=I&K&1
      X(I)=XP(I)-.5*(FPD(IA)&FPD(IA&3))
      H(I)=.5*(FPD(IA&3)-FPD(IA))
 1620 DX=DX&X(I)*H(I)
      DO 1630 I=1,3
      UN(I)=C2*X(I)-DX*H(I)
 1630 AN=AN&UN(I)**2
      IF(AN.LT.1.0E-12) GO TO 999
```

```
MEMBER NAME  GNORM
      AN=SQRT(AN)                                                       NORM 750
      DO 1635 I=1,3                                                     NORM 760
 1635 UN(I)=UN(I)/AN                                                    NORM 770
      GO TO 2000                                                        NORM 780
C   BOX
 1700 L= 1&(LSUR-1)/2                                                   NORM 830
      I1=IBOX1(L)                                                       NORM 840
      I2=IBOX2(L)                                                       NORM 850
      DO 1720 I=1,3                                                     NORM 860
      X(I)=FPD(K&I1&I)                                                  NORM 870
 1720 H(I)=FPD(K&I2&I)                                                  NORM 880
      AN=0.0                                                            NORM 890
      DO 1730 I=1,3                                                     NORM 900
      J=MOD(I,3)&1                                                      NORM 910
      JJ=MOD(I&1,3)&1                                                   NORM 920
      UN(I)=X(J)*H(JJ)-X(JJ)*H(J)                                       NORM 930
 1730 AN=AN&UN(I)**2                                                    NORM 940
      IF(AN.LT.1.0E-12) GO TO 999                                       NORM 950
      AN=SQRT(AN)                                                       NORM 960
      DO 1735 I=1,3                                                     NORM 970
 1735 UN(I)=UN(I)/AN                                                    NORM 980
      GO TO 2000                                                        NORM 990
C------TEC.
 1800 IF(LSUR.EQ.3)GO TO 1820
      K=K&7
      DO 1810 J=1,3
 1810 UN(J)=FPD(K&J)
      GO TO 2000
 1820 LF=K&2
      XDN=XP(1)*FPD(LF&6)&XP(2)*FPD(LF&7)&XP(3)*FPD(LF&8)
      HDN=1.0/FPD(K&21)
      DDN=XDN-FPD(K&25)
      A1=-DDN*HDN                                                       T2-T1
      R1=XP(1)-FPD(LF)    &A1*FPD(LF&3)                                  TEMP1
      R2=XP(2)-FPD(LF&1)&A1*FPD(LF&4)                                    TEMP2
      R3=XP(3)-FPD(LF&2)&A1*FPD(LF&5)                                    TEMP3
      B1=(R1*FPD(LF&9)&R2*FPD(LF&10)&R3*FPD(LF&11))/FPD(K&23)            T3
      B2=R1*FPD(K&17)&R2*FPD(K&18)&R3*FPD(K&19)                          T4
      B3=FPD(K&20)*HDN                                                   T5
      B4=FPD(K&22)*HDN                                                   T6
      A2=FPD(LF&13)*(1.0/FPD(LF&14) -1.0)                                R4-R2
      A3=(-A1*A2&FPD(LF&13))*A2*HDN                                     EM*R*HDN
      A1=-(B1*B3&B2*B4&A3)
      I=K&16
      LF=LF&5
      K=LF&3
      DO 1850 J=1,3
 1850 UN(J)=A1*FPD(LF&J)&B1*FPD(K&J)&B2*FPD(I&J)
      GO TO 1505
C   RPP
 1900 DO 1905 I=1,3                                                     NOR 1000
 1905 UN(I)=0.0                                                         NOR 1010
      GO TO (1910,1910,1920,1920,1930,1930), LSUR                       NOR 1020
 1910 UN(2)=1.0                                                         NOR 1030
      GO TO 2000                                                        NOR 1040
 1920 UN(1)=1.0                                                         NOR 1050
      GO TO 2000                                                        NOR 1060
 1930 UN(3)=1.0                                                         NOR 1070
```

```
MEMBER NAME  GNORM
C-------OUTWARD NORMAL.
 2000 AN=UN(1)*WB(1)&UN(2)*WB(2)&UN(3)*WB(3)
      IF(AN.LE.0.0)GO TO 3000
      DO 2010 J=1,3
   10 UN(J)=-UN(J)
 3000 RETURN
      END
```

NOF 1160

```
MEMBER NAME  GP
      SUBROUTINE GP(K)
C------NEW NAME FOR SUBROUTINE PR .
      COMMON/FEB073/DATEX
      COMMON / DBG / N,NUM,LOCAT,ISAVE,INEXT,IRP,INEX,SMIN
      COMMON / GEOM2 / LOCREG(200),NUMBOD(200),IROR(200),MA(4000),
     1                 FPD(4000),LDATA,NUMB,NUMR
      COMMON/PAREM /XB(3) ,WB(3) ,E      ,IR     ,T      ,IDET   ,F     ,   PAREM
     1              NHIST ,WC     ,J12345,WP(3) ,XP(3) ,EPRIM  ,ATWT  ,   PAREM
     2              NCDB  ,CSTHT ,U      ,LCHI   ,IATWT  ,IERR   ,ICBG      PAREM
     3,             IRPRIM,NASC   ,LSURF  ,NBO    ,LRI    ,LRO    ,RIN   ,  PAREM
     4              ROUT  ,KLOOP ,LOOP   ,ITYPE  ,PINF   ,NOA ,DIST          P01
      IF(IDBG.EQ.1.AND.K.NE.8) GO TO 200
      WRITE(6,50) K,XB,WB,IR                                                   E
   50 FORMAT( 3H GP,       I5,6E15.8,I5)
      WRITE(6,100) IR,IRPRIM,NASC,LSURF,NBO,LRI,LRO,KLOOP,LOOP,ITYPE,
     1             N,NUM,LOCAT,ISAVE,INEXT,IRP,INEX,LDATA,K,K,                A
     2             RIN,ROUT,SMIN,DIST
  100 FORMAT(10I10,/10I10,/5E15.8)
  200 RETURN
      END
```

MEMBER NAME G1

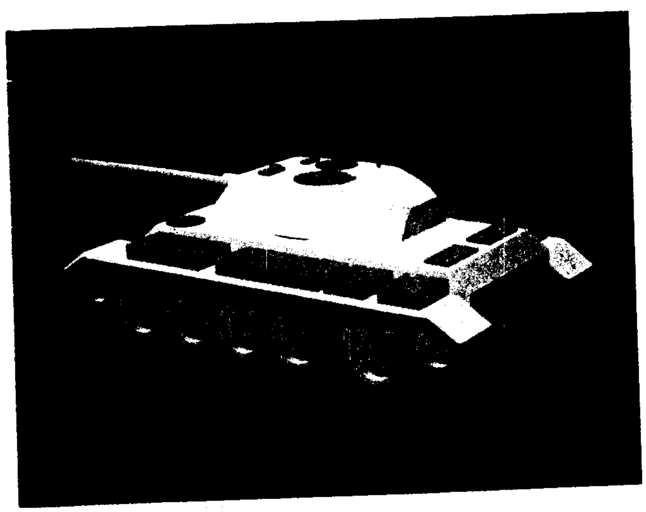
```
      SUBROUTINE G1(S)
      COMMON/JAN243 / DATEX
      LOGICAL LOGG
      COMMON / DBG / N,NUM,LOCAT,ISAVE,INEXT,IRP,INEX,SMIN
      COMMON / GEOM2 / LOCREG(200),NUMBOD(200),IROR(200),MA(4000),
     1                 FPD(4000),LDATA,NUMB,NUMR
      COMMON/GEOM3/EPS,IRHA(200)
      COMMON/PAREM /XB(3) ,WB(3) ,E      ,IRTRU ,T      ,IDET  ,F     ,    PAREM
C-------------------------------------  IR  ---(NORMALLY).
     1              NHIST ,WC     ,J12345,WP(3) ,XP(3) ,EPRIM ,ATWT  ,    PAREM
     2              NCDB  ,CSTHT ,U      ,LCHI  ,IATWT ,IERR  ,IDBG       PAREM
     3,             IRPRIM,NASC  ,LSURF ,NBO   ,LRI   ,LRO   ,RIN   ,    PAREM
     4              ROUT  ,KLOOP ,LOOP  ,ITYPE ,PINF  ,NOA ,DIST
      COMMON/TAPE  /INT    ,IOUT  ,IEDT  ,INT1  ,INT2  ,IAGG,ITRAN,INTER TAPE
      DATA LMAX/4000/,KLS/0/
      SP=0.
      IF(NASC.GT.0) GO TO 40
      KLOOP=KLOOP&1
      DIST=C.
   40 IF(KLOOP.EQ.KLS)GO TO 110
      IRH=IRHA(IRTRU)
      IRL=1
      IF(IRTRU.GT.1)IRL=IRHA(IRTRU-1)&1
      IR=IRL
      IF(IR.EQ.IRH)GO TO 110
      DIS=(1.0-EPS)*DIST
      DO 70 IRP=IRL,IRH
      N=LOCREG(IRP)&1
      NUM=NUMBOD(IRP)*4&N-4
      DO 65 I=N,NUM,4
      NBO=MA(I)
      LOCAT=MA(I&1)
      ITYPE=MA(LOCAT&2)
      IF(ITYPE.NE.10)CALL GG(LOCAT)
      IF(ITYPE.EQ.10)CALL GGTEC(LOCAT)
      LOGG=(ROUT.LE.DIS ).OR.(RIN.GT.DIS )
      IF(NBO)62,65,63
   62 LOGG=.NOT.LOGG
   63 IF(LOGG)GO TO 70
   65 CONTINUE
      IR=IRP
      GO TO 110
   70 CONTINUE
      WRITE(IOUT,80)IRL,IRH,XB,WB,DIST
   80 FORMAT(40H1REGION NOT FOUND FOR IRL,IRH/XB,WB,DIST,2I5/7E15.6)
      STOP
  110 SMIN=PINF
      N=LOCREG(IR)&1
      NUM=NUMBOD(IR)*4 &N-4
      IF(IDBG.NE.0) CALL GP(1)
C THE LOOP UPTO 300 FINDS THE NEXT BODY THAT THE RAY WILL INTERSECT
C
      ISAVE=0
      DO 300 I=N,NUM,4
      NBO=MA(I)
      LOCAT=MA(I&1)
      ITYPE=MA(LOCAT&2)
      IF(ITYPE.NE.10)CALL GG(LOCAT)
```

```
MEMBER NAME  G1
      IF(ITYPE.EQ.10)CALL GGTEC(LOCAT)
      IF(IDBG.NE.0) CALL GP(2)
      IF(ROUT.LE.0) GO TO 300
      IF( NBO ) 150,300,200
  150 IF( (DIST.GT.RIN).OR.(RIN.GT.SMIN)) GO TO 300
      SMIN=RIN
      NASC=-NBO
      LSURF=LRI
      ISAVE=I
      GO TO  300
  200 IF( (DIST.GT.ROUT).OR.(ROUT.GT.SMIN))GO TO 300
      SMIN=ROUT
      NASC=NBO
      LSURF=-LRO
      ISAVE=I
  300 CONTINUE
      IF(ISAVE.NE.0) GO TO 309
      WRITE(IOUT,305) IR,NHIST,J12345,NASC,N,NUM,NBO,LOCAT,LSURF,ISAVE,
     1  LRO,ROUT,RIN,DIST,SMIN,E,XB,WB                                    01/73
  305 FORMAT(24H NO VALID DISTANCE IN G1,/,11I5,5E13.5,/,6E13.5)
      IRPRIM=0
      GO TO 5000
C
C NOW TO FIND NEXT REGION
C
  309 S=SMIN -DIST&SP
      DIST=SMIN
      DIS=(1.0&EPS)*DIST
      INEXT=ISAVE&2
  310 IRP=MA(INEXT)
      IF(IDBG.NE.0) CALL GP(3)
      IF( IRP.EQ.0)GO TO  600
      N=LOCREG(IRP)&1
      NUM=NUMBOD(IRP)*4 & N-4
C
C THE LOOP TO 400 EXAMINES REGION IRP TO SEE IF IT IS THE NEXT REGION
C
      IF(IDBG.NE.0) CALL GP(4)
      DO 400 I=N,NUM,4
      NBO=MA(I)
      LOCAT=MA(I&1)
      ITYPE=MA(LOCAT&2)
      IF(ITYPE.NE.10)CALL GG(LOCAT)
      IF(ITYPE.EQ.10)CALL GGTEC(LOCAT)
      IF(IDBG.NE.0) CALL GP(5)
      IF(NBO) 320,400 ,330
  320 IF(                   (ROUT.LE.DIS ).OR.(RIN.GT.DIS ) ) GO TO 400
      GO TO 500
  330 IF( (RIN.LE.DIS ).AND.(DIS .LT.ROUT) )  GO TO 400
      GO TO 500
  400 CONTINUE
C
C FOUND A REGION
      GO TO 750
  500 INEX=INEXT & 1
      INEXT=MA(INEX)
      IF(INEXT.GT.0) GO TO 310
C
```

```
MEMBER NAME  G1
C SEARCH ALL REGIONS
C
      MA(INEX)=LDATA
      INEXT=LDATA
      LDATA=LDATA&2
  .  J CONTINUE
      IF(IDBG.NE.0) CALL GP(6)
      DO  700 IRP=1,NUMR
      IF(IRP.EQ.IR)GO TO 700
      N=LOCREG(IRP)&1
      NUM=NUMBOD(IRP)*4&N-4
      DO  650   I=N,NUM,4
      NBD=MA(I)
      LOCAT=MA(I&1)
      ITYPE=MA(LOCAT&2)
      IF(ITYPE.NE.10)CALL GG(LOCAT)
      IF(ITYPE.EQ.10)CALL GGTEC(LOCAT)
      IF(IDBG.NE.0) CALL GP(7)
      IF(NBD) 620,650,630
  620 IF(                    (ROUT.LE.DIS ).OR.(RIN.GT.DIS )) GO TO 650
      GO TO 700
  630 IF( (RIN.LE.DIS ).AND.(DIS .LT.ROUT)) GO TO 650
      GO TO 700
  650 CONTINUE
      MA(INEXT)=IRP
      IF(INEXT.LT.LMAX) GO TO 750
      WRITE(IOUT,655) NHIST,KLOOP
  655 FORMAT(21H GEOMETRY ARRAY FULL ,2I10)
      IRPRIM=0
      GO TO 5000
  7 ? CONTINUE
      WRITE(IOUT,703)IRTRU,IR,XB,WB,DIST
  703 FORMAT(22HONEXT REGION NOT FOUND/7H IRTRU=I5,4X,
     C 4H IR=I5,5X,3HXB=3E15.6/4H WB=3E15.6,5X,5HDIST=E15.6)
      IRPRIM=0
 5000 KLS=KLOOP
      RETURN
  750 IRPRIM=IROR(IRP)
      IF(IDBG.NE.0) CALL GP(8)
      IR=IRP
      IF(IRPRIM.NE.IRTRU)GO TO 5000
      SP=SP&S
      GO TO 110
      END
```

APPENDIX B

CG T-54 TANK INPUT (LISTING)



Picture generated by
MAGI SynthaVision process

0			TANK			
ARB 1	13.	5.	-3.5	19.	4.2	-3.5
	1.	4.2	-3.5	2.	5.	-3.5
	18.	5.	3.5	19.	4.2	3.5
	1.	4.2	3.5	2.	5.	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 2	19.	4.2	-3.5	20.8	2.5	-3.5
	1.	2.5	-3.5	1.	4.2	-3.5
	19.	4.2	3.5	20.8	2.5	3.5
	1.	2.5	3.5	1.	4.2	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 3	20.8	2.5	-3.5	18.	1.	-3.5
	1.2	1.	-3.5	1.	2.5	-3.5
	20.8	2.5	3.5	18.	1.	3.5
	1.2	1.	3.5	1.	2.5	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 4	20.5	4.	-6.	22.	3.	-6.
	21.8	3.	-6.	20.5	3.8	-6.
	20.5	4.	-3.5	22.	3.	-3.5
	21.8	3.	-3.5	20.5	3.8	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 5	1.5	3.8	-6.	19.	0.	0.
	0.	0.2	0.	0.	0.	2.5
ARB 6	1.5	4.	-6.	1.5	3.8	-6.
	0.2	3.	-6.	0.	3.	-6.
	1.5	4.	-3.5	1.5	3.8	-3.5
	0.2	3.	-3.5	0.	3.	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 7	20.5	4.	6.	22.	3.	6.
	21.8	3.	6.	20.5	3.8	6.
	20.5	4.	3.5	22.	3.	3.5
	21.8	3.	3.5	20.5	3.8	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 8	1.5	3.8	3.5	19.	0.	0.
	0.	0.2	0.	0.	0.	2.5
ARB 9	1.5	4.	6.	1.5	3.8	6.
	0.2	3.	6.	0.	3.	6.
	1.5	4.	3.5	1.5	3.8	3.5
	0.2	3.	3.5	0.	3.	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
RCC 10	20.2	2.3	-5.5	0.	0.	1.7
	0.9					
RCC 11	20.2	2.3	-5.6	0.	0.	0.5
	0.75					
RCC 12	20.2	2.3	-5.3	0.	0.	0.3
	0.2					
RCC 13	17.5	1.5	-5.5	0.	0.	1.7
	1.3					
RCC 14	17.5	1.5	-5.6	0.	0.	0.5
	1.1					
RCC 15	17.5	1.5	-5.3	0.	0.	0.3
	0.2					
RCC 16	14.5	1.5	-5.5	0.	0.	1.7
	1.3					
RCC 17	14.5	1.5	-5.6	0.	0.	0.5
	1.1					
RCC	14.5	1.5	-5.3	0.	0.	0.3
	0.2					

```
RCC 19   11.5   1.5   -5.5   0.     0.     1.7
         1.3
RCC 20   11.5   1.5   -5.6   0.     0.     0.5
         1.1
RCC 21   11.5   1.5   -5.3   0.     0.     0.3
         0.2
RCC 22   8.5    1.5   -5.5   0.     0.     1.7
         1.3
RCC 23   8.5    1.5   -5.6   0.     0.     0.5
         1.1
RCC 24   8.5    1.5   -5.3   0.     0.     0.3
         0.2
RCC 25   4.5    1.5   -5.5   0.     0.     1.7
         1.3
RCC 26   4.5    1.5   -5.6   0.     0.     0.5
         1.1
RCC 27   4.5    1.5   -5.3   0.     0.     0.3
         0.2
RCC 28   1.5    2.3   -5.5   0.     0.     1.7
         0.9
RCC 29   1.5    2.3   -5.6   0.     0.     0.5
          0.75
RCC 30   1.5    2.3   -5.3   0.     0.     0.3
         0.2
RCC 31   20.2   2.3   3.8    0.     0.     1.7
         0.9
RCC 32   17.5   1.5   3.8    0.     0.     1.7
         1.3
RCC 33   14.5   1.5   3.8    0.     0.     1.7
         1.3
RCC 34   11.5   1.5   3.8    0.     0.     1.7
         1.3
RCC 35   8.5    1.5   3.8    0.     0.     1.7
         1.3
RCC 36   4.5    1.5   3.8    0.     0.     1.7
         1.3
RCC 37   1.5    2.3   3.8    0.     0.     1.7
         0.9
BOX 38   4.2    2.8   -6.    13.6   0.     0.
         0.     0.2   0.     0.     0.     2.5
ARB 39   17.8   3.    -6.    20.2   3.4    -6.
         20.2   3.2   -6.    17.8   2.8    -6.
         17.8   3.    -3.5   20.2   3.4    -3.5
         20.2   3.2   -3.5   17.8   2.8    -3.5
         1234.  5678. 1562.  2673.  3784.  1584.
ARB 40   20.3   3.4   -6.    20.8   3.25   -6.
         20.7   3.1   -6.    20.2   3.2    -6.
         20.3   3.4   -3.5   20.8   3.25   -3.5
         20.7   3.1   -3.5   20.2   3.2    -3.5
         1234.  5678. 1562.  2673.  3784.  1584.
ARB 41   20.8   3.25  -6.    21.3   2.6    -6.
         21.1   2.6   -6.    20.7   3.1    -6.
         20.8   3.25  -3.5   21.3   2.6    -3.5
         21.1   2.6   -3.5   20.7   3.1    -3.5
         1234.  5678. 1562.  2673.  3784.  1584.
BOX 42   21.1   2.    -6.    0.2    0.     0.
         0.     0.6   0.     0.     0.     2.5
ARB 43   21.1   2.    -6.    21.3   2.     -6.
```

	20.93	1.45	-6.	20.8	1.6	-6.
	21.1	2.	-3.5	21.3	2.	-3.5
	20.93	1.45	-3.5	20.8	1.6	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 44	20.8	1.6	-6.	20.93	1.45	-6.
	17.8	0.	-6.	17.8	0.2	-6.
	20.8	1.6	-3.5	20.93	1.45	-3.5
	17.8	0.	-3.5	17.8	0.2	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 45	4.2	0.	-6.	13.6	0.	0.
	0.	0.2	0.	0.	0.	2.5
ARB 46	4.2	0.	-6.	0.9	1.38	-6.
	1.	1.5	-6.	4.2	0.2	-6.
	4.2	0.	-3.5	0.9	1.38	-3.5
	1.	1.5	-3.5	4.2	0.2	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 47	0.9	1.38	-6.	1.	1.5	-6.
	0.6	2.	-6.	0.4	2.	-6.
	0.9	1.38	-3.5	1.	1.5	-3.5
	0.6	2.	-3.5	0.4	2.	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 48	0.4	2.	-6.	0.2	0.	0.
	0.	0.6	0.	0.	0.	2.5
ARB 49	0.4	2.6	-6.	0.6	2.6	-6.
	1.	3.1	-6.	0.9	3.28	-6.
	0.4	2.6	-3.5	0.6	2.6	-3.5
	1.	3.1	-3.5	0.9	3.28	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 50	0.9	3.28	-6.	1.	3.1	-6.
	1.5	3.2	-6.	1.5	3.4	-6.
	0.9	3.28	-3.5	1.	3.1	-3.5
	1.5	3.2	-3.5	1.5	3.4	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 51	1.5	3.2	-6.	1.5	3.4	-6.
	4.2	3.	-6.	4.2	2.8	-6.
	1.5	3.2	-3.5	1.5	3.4	-3.5
	4.2	3.	-3.5	4.2	2.8	-3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 52	4.2	2.8	3.5	13.6	0.	0.
	0.	0.2	0.	0.	0.	2.5
ARB 53	17.8	3.	6.	20.2	3.4	6.
	20.2	3.2	6.	17.8	2.8	6.
	17.8	3.	3.5	20.2	3.4	3.5
	20.2	3.2	3.5	17.8	2.8	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 54	20.3	3.4	6.	20.8	3.25	6.
	20.7	3.1	6.	20.2	3.2	6.
	20.3	3.4	3.5	20.8	3.25	3.5
	20.7	3.1	3.5	20.2	3.2	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 55	20.8	3.25	6.	21.3	2.6	6.
	21.1	2.6	6.	20.7	3.1	6.
	20.8	3.25	3.5	21.3	2.6	3.5
	21.1	2.6	3.5	20.7	3.1	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 56	21.1	2.	3.5	0.2	0.	0.
	0.	0.6	0.	0.	0.	2.5
ARB 57	21.1	2.	6.	21.3	2.	6.

	20.93	1.45	6.	20.8	1.6	6.
	21.1	2.	3.5	21.3	2.	3.5
	20.93	1.45	3.5	20.8	1.6	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 58	20.8	1.6	6.	20.93	1.45	6.
	17.8	0.	6.	17.8	0.2	6.
	20.8	1.6	3.5	20.93	1.45	3.5
	17.8	0.	3.5	17.8	0.2	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 59	4.2	0.	3.5	13.6	0.	0.
	0.	0.2	0.	0.	0.	2.5
ARB 60	4.2	0.	6.	0.9	1.38	6.
	1.	1.5	6.	4.2	0.2	6.
	4.2	0.	3.5	0.9	1.38	3.5
	1.	1.5	3.5	4.2	0.2	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 61	0.9	1.38	6.	1.	1.5	6.
	0.6	2.	6.	0.4	2.	6.
	0.9	1.38	3.5	1.	1.5	3.5
	0.6	2.	3.5	0.4	2.	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 62	0.4	2.	3.5	0.2	0.	0.
	0.	0.6	0.	0.	0.	2.5
ARB 63	0.4	2.6	6.	0.6	2.6	6.
	1.	3.1	6.	0.9	3.28	6.
	0.4	2.6	3.5	0.6	2.6	3.5
	1.	3.1	3.5	0.9	3.28	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 64	0.9	3.28	6.	1.	3.1	6.
	1.5	3.2	6.	1.5	3.4	6.
	0.9	3.28	3.5	1.	3.1	3.5
	1.5	3.2	3.5	1.5	3.4	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
ARB 65	1.5	3.2	6.	1.5	3.4	6.
	4.2	3.	6.	4.2	2.8	6.
	1.5	3.2	3.5	1.5	3.4	3.5
	4.2	3.	3.5	4.2	2.8	3.5
	1234.	5678.	1562.	2673.	3784.	1584.
BOX 66	13.5	4.0	-5.8	4.3	0.	0.
	0.	0.7	0.	0.	0.	2.3
BOX 67	7.	4.	-5.8	5.5	0.	0.
	0.	0.7	0.	0.	0.	2.3
BOX 68	4.7	4.	-5.8	2.	0.	0.
	0.	0.7	0.	0.	0.	2.3
BOX 69	2.	4.	-5.8	2.	0.	0.
	0.	0.7	0.	0.	0.	2.3
BOX 70	14.8	4.	3.5	3.2	0.	0.
	0.	0.7	0.	0.	0.	2.3
BOX 71	11.	4.	4.8	2.	0.	0.
	0.	0.7	0.	0.	0.	1.
BOX 72	6.	4.	3.5	3.3	0.	0.
	0.	0.7	0.	0.	0.	2.3
BOX 73	2.2	4.	3.5	3.3	0.	0.
	0.	0.7	0.	0.	0.	2.3
BOX 74	19.	3.7	-3.3	0.2	0.	0.
	0.	0.6	0.	0.	0.	6.6
TEC 75	11.	5.4	0.	0.	2.1	0.
	0.	1.	0.	1.	0.	0.

```
          6.9        3.4        1.2
BOX  76   3.9        5.3       -2.5       1.4        0.         0.
          0.         1.7        0.        0.         0.         5.
ARB  77   4.         6.4       -3.4       8.5        7.6       -3.4
          8.5        7.7       -3.4       4.         7.7       -3.4
          4.         6.4        3.4       8.5        7.6        3.4
          8.5        7.7        3.4       4.         7.7        3.4
          1234.      5678.      1562.     2673.      3784.      1584.
ARB  78   13.9       7.6       -3.4       17.        7.6       -3.4
          18.        5.3       -3.4       15.8       5.3       -3.4
          13.9       7.6        3.4       17.        7.6        3.4
          18.        5.3        3.4       15.8       5.3        3.4
          1234.      5678.      1562.     2673.      3784.      1584.
RCC  79   11.        5.         0.        0.         1.         0.
          2.5
RCC  80   10.        7.4       -1.        0.         0.2        0.
          1.3
RCC  81   11.        7.4        1.5       0.         0.2        0.
          1.
RCC  82   17.        4.9       -2.5       0.         0.2        0.
          0.95
RCC  83   14.        6.5        0.        2.5        0.         0.
          0.4
RCC  84   16.5       6.5        0.        10.5       0.         0.
          0.2
BOX  85   2.3        5.        -3.        1.2        0.         0.
          0.         0.2        0.        0.         0.         2.5
BOX  86   2.3        5.         0.5       1.2        0.         0.
          0.         0.2        0.        0.         0.         2.5
BOX  87   8.6        7.5       -1.6       0.3        0.         0.
          0.         0.15       0.        0.         0.         1.2
B    88   12.5       7.5       -1.7       0.6        0.         0.
          0.         0.2        0.        0.         0.         1.3
BOX  89   12.3       7.5        1.        0.4        0.         0.
          0.         0.2        0.        0.         0.         1.
RCC  90   13.4       7.5        0.        0.         0.1        0.
          0.3
RCC  91   9.         7.5        2.        0.         0.5        0.
          0.1
RPP  92   -100000.   100000.   -100000.   100000.   -100000.    100000.
RPP  93   -1000100.  1000100.  -1000100.  1000100.  -1000100.   1000100.
RPP  94   -1.        28.        5.3       8.        -7.         7.
RPP  95   -1.        28.       -1.        5.4       -3.5        3.5
RPP  96   -1.        28.        3.5       5.4       -7.4       -3.4
RPP  97   -1.        28.        3.5       5.4        3.4        7.
RPP  98   19.        28.       -1.        3.6       -7.        -3.4
RPP  99   15.9       19.1      -1.        3.6       -7.        -3.4
RPP 100   6.9        16.       -1.        3.6       -7.        -3.4
RPP 101   2.7        7.        -1.        3.6       -7.        -3.4
RPP 102   -1.        2.8       -1.        3.6       -7.        -3.4
RPP 103   19.        28.       -1.        3.6        3.4        7.
RPP 104   6.9        19.1      -1.        3.6        3.4        7.
RPP 105   -1.        7.        -1.        3.6        3.4        7.
END
 1           1         0        0        0
 2           2
             3         0        0        0
 4           4         0        0        0
```

5	5	0	0	0
6	6	0	0	0
7	7	0	0	0
8	8	0	0	0
9	9	0	0	0
10	10	-11	-12	
11	12	0	0	0
12	13	-14	-15	0
13	15	0	0	0
14	16	-17	-18	0
15	18	0	0	0
16	19	-20	-21	0
17	21	0	0	0
18	22	-23	-24	0
19	24	0	0	0
20	25	-26	-27	0
21	27	0	0	0
22	28	-29	-30	0
23	30	0	0	0
24	31	0	0	0
25	32	0	0	0
26	33	0	0	0
27	34	0	0	0
28	35	0	0	0
29	36	0	0	0
30	37	0	0	0
31	38	0	0	0
32	39	0	0	0
33	40	0	0	0
34	41	0	0	0
35	42	0	0	0
36	43	0	0	0
37	44	0	0	0
38	45	0	0	0
39	46	0	0	0
40	47	0	0	0
41	48	0	0	0
42	49	0	0	0
43	50	0	0	0
44	51	0	0	0
45	52	0	0	0
46	53	0	0	0
47	54	0	0	0
48	55	0	0	0
49	56	0	0	0
50	57	0	0	0
51	58	0	0	0
52	59	0	0	0
53	60	0	0	0
54	61	0	0	0
55	62	0	0	0
56	63	0	0	0
57	64	0	0	0
58	65	0	0	0
59	66	0	0	0
60	67	0	0	0
61	68	0	0	0
62	69	0	0	0

63	70	0	0	0					
64	71	0	0	0					
65	72	0	0	0					
66	73	0	0	0					
67	74	0	0	0					
68	75	-76	-77	-78					
69	79	0	0	0					
70	80	0	0	0					
71	81	0	0	0					
72	82	0	0	0					
73	83	0	0	0					
74	84	0	0	0					
75	85	0	0	0					
76	86	0	0	0					
77	87	0	0	0					
78	88	0	0	0					
79	89	0	0	0					
80	90	0	0	0					
81	91	0	0	0					
82	11	-12							
83	14	-15							
84	17	-18							
85	20	-21							
86	23	-24							
87	26	-27							
88	29	-30							
89	76								
90	77								
91	78	-84	-83						
92	92	-94	-95	-96	-97	-98	-99	-100	-101
	-102	-103	-104	-105					
	93	-92							
94	94	-75	-79	-80	-81	-83	-84	-87	-88
	-89	-90	-91						
95	95	-1	-2	-3	-74	-79	-82	-85	-86
96	96	-1	-2	-4	-5	-6	-66	-67	-68
	-69	-82							
97	97	-1	-2	-7	-8	-9	-70	-71	-72
	-73								
98	98	-2	-3	-4	-10	-12	-39	-40	-41
	-42	-43	-44						
99	99	-2	-3	-13	-15	-38	-39	-44	-45
100	100	-2	-3	-16	-18	-19	-21	-22	-24
	-38	-45							
101	101	-2	-3	-25	-27	-45	-46	-51	-38
102	102	-2	-3	-6	-28	-30	-46	-47	-48
	-49	-50	-51						
103	103	-2	-3	-7	-31	-53	-54	-55	-56
	-57	-58							
104	104	-2	-3	-33	-34	-35	-52	-53	-58
	-59	-32							
105	105	-2	-3	-36	-37	-52	-59	-60	-61
	-62	-63	-64	-65	-9				

END