AN ABSTRACT OF THE THESIS OF

Karen Jensen Butler for the Master of Science in
(Name) (Degree)

Mathematics
(Major)

Date thesis is presented June 7, 1963

Title An Algorithm for Determining the Convex Hull of N points in 3-Space

Abstract approved [redacted]
(Major professor)

An algorithm is described for determining the vertices and supporting planes (or lines) of the convex hull of a given set of N distinct points in 3-space. The method involves determining a finite sequence of convex hulls, each successive convex hull being a better approximation to the convex hull of the given N points. The final convex hull being the one desired.

The algorithm is programmed in Fortran I for an IBM 1620 with 40,000 positions of numeric storage. These programs and several examples are included, along with proofs for a few of the theorems necessary in the development of the algorithm.

AN ALGORITHM FOR DETERMINING THE CONVEX HULL OF N POINTS IN 3-SPACE

by

KAREN JENSEN BUTLER

A THESIS

submitted to

OREGON STATE UNIVERSITY

in partial fulfillment of
the requirements for the
degree of

MASTER OF SCIENCE

August 1963

APPROVED:

[redacted]

Associate Professor of Mathematics

In Charge of Major

[redacted]

Chairman of Department of Mathematics

[redacted]

Dean of Graduate School

Date thesis is presented June 7, 1963

Typed by Ola Gara

ACKNOWLEDGMENTS

The author wishes to thank Dr. Robert D. Stalley and the Statistics Computing Laboratory for assisting in the completion of this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

AN ALGORITHM FOR DETERMINING THE CONVEX HULL OF N POINTS IN 3-SPACE

Chapter 1

INTRODUCTION

Given a set of N distinct points in 3-space there exists a unique convex hull determined by these N points. The purpose of this paper is (1) to determine which of the given N points are vertices of the convex hull and (2) to obtain an equation for each supporting plane of the convex hull, or if the given N points are coplanar, an equation of the plane containing the given N points and equations for each supporting line. In particular if the given N points are collinear, we determine equations of the line containing the given N points and in this case the vertices are the two endpoints of the line segment containing the given N points.

The following discussion assumes the reader is familiar with the basic fundamentals of analytic geometry and the concepts of elementary set theory. A familiarity with the 1962 version of 1620 Fortran as described in the IBM 1620 Reference Manual (1962 edition, C26-5619-0) is assumed in Chapter 5. However a knowledge of Fortran is not essential for any other chapter.

Balinski (3) describes a method for finding all vertices of a convex polyhedral set defined by a system of linear inequalities based on a variation of the simplex method. Another procedure, the double description method proposed by Motzkin, Raiffa, Thompson and Thrall (10), builds up the convex hull by introducing the half spaces given by the linear inequalities one at a time.

Both of these methods for determining the vertices of a convex hull begin with a given set of linear inequalities. The method we propose begins with a set of N distinct points and obtains the vertices and supporting planes (or lines) of the convex hull determined by these N points.

In Chapter 2 we suggest several methods for determining the convex hull from a given set of N distinct points and describe in general the method adopted, explaining the basic idea behind it. Chapter 3 contains a detailed description of the steps the computer follows in determining a convex hull. Theorems supporting the algorithm are contained in Chapter 4. No attempt has been made to prove each statement in the algorithm. Since the main purpose of the paper is to develop a computationally feasible procedure for determining the convex hull of a given set of N distinct points the only proofs included are for those statements that are not inituitively obvious and hence require some justification. In the 3-dimensional case many proofs are omitted since they closely parallel the corresponding proofs in 2-space. Chapter 5 contains three 1620 Fortran programs for obtaining the vertices and supporting planes (or lines) of the convex hull determined by a set of N distinct points. For the 1620 with 40,000 digits of storage the maximum value for N is 51. Also included in Chapter 5 is a general flow chart for each Fortran program. These programs have been tested on approximately 70 examples. For all but two of these examples the convex hull was successfully determined. The two unsuccessful attempts to determine the convex hull were not due to an error, or errors, in the programs but rather to the limitations of the computer. These limitations and possible modifications to the computer procedure, as described in Chapter 3, are discussed in Chapter 7. Six examples are included in Chapter 6 to illustrate the possible output from the programs in Chapter 5.

Chapter 2

PRELIMINARY REMARKS

Before we can give a formal definition of a convex hull we must first establish what we mean by a convex combination. A convex combination of a finite number of points $P_1,..,P_N$ is defined as a point P with coordinates x, y and z such that

$$x = \sum_{i=1}^{N} \mu_i x_i, \qquad y = \sum_{i=1}^{N} \mu_i y_i, \qquad z = \sum_{i=1}^{N} \mu_i z_i, \text{ where}$$

$\mu_i \geq 0$, i = 1,..,N, and $\sum_{i=1}^{N} \mu_i = 1$.

We define the convex hull of a finite number of points $P_1,..,P_N$ to be the set of all convex combinations of $P_1,..,P_N$ (5, p. 208).

We will refer to a given point in two ways; as the point P_i, $1 \leq i \leq N$, or by specifying its coordinates (x_i,y_i,z_i), $1 \leq i \leq N$. It will be clear from the context whether we mean the i-th point determined by the coordinates (x_i,y_i,z_i), $1 \leq i \leq N$, or whether we are referring to the coordinates themselves.

We now define what we mean by a supporting plane of the convex hull of a given set of N points in 3-space. The equation $A_k x + B_k y + C_k z + D_k = 0$ is the equation of a supporting plane if there exist at least three noncollinear points P_1, P_2 and P_3 among the given N such that

$$A_k x_i + B_k y_i + C_k z_i + D_k = 0, \quad i = 1,2,3$$

and if either

$$A_k x_a + B_k y_a + C_k z_a + D_k \geq 0, \text{ for all } a \text{ where } 1 \leq a \leq N,$$

or

$$A_k x_a + B_k y_a + C_k z_a + D_k \leq 0, \text{ for all } a \text{ where } 1 \leq a \leq N.$$

Similarly $A_k x + B_k y + C_k = 0$ is the equation of a <u>supporting line</u> of the convex hull of a given set of N points in 2-space if there exist at least two distinct points P_1 and P_2 among the given N points such that

$$A_k x_i + B_k y_i + C_k = 0, \quad i = 1,2,$$

and if either

$$A_k x_a + B_k y_a + C_k \geq 0, \text{ for all } a \text{ where } 1 \leq a \leq N,$$

or

$$A_k x_a + B_k y_a + C_k \leq 0, \text{ for all } a \text{ where } 1 \leq a \leq N.$$

In 3-space a point P_i, $1 \leq i \leq N$, is a <u>vertex</u> of the convex hull determined by the given N points iff. there do not exist points P_1 and P_2 in the set such that

$$x_i = (1 - \lambda)x_1 + \lambda x_2,$$

$$y_i = (1 - \lambda)y_1 - \lambda y_2,$$

$$z_i = (1 - \lambda)z_1 - \lambda z_2, \quad 0 < \lambda < 1.$$

Similarly in 2-space the point P_i, $1 \leq i \leq N$, is a <u>vertex</u> of the convex hull determined by the given N points iff. there do not exist points P_1 and P_2 in the set such

$$x_i = (1 - \lambda)x_1 + \lambda x_2,$$

$$y_i = (1 - \lambda)y_1 + \lambda y_2, \quad 0 < \lambda < 1.$$

If N is relatively small there are a number of apparent ways to determine the convex hull, that is the vertices and supporting planes (or lines) of the convex hull. One possibility, if the given N points are not coplanar, would be to determine an equation for each plane that contains three distinct noncollinear points from the given set of N points. Then by substituting the coordinates of each of the N points into the left member of each of the above equations we can easily see which of these equations are equations of supporting planes.

If the given N points are coplanar we could obtain an equation for each line containing two distinct points from the given set of N points and then determine by substitution which of the equations are equations of supporting lines. These methods, while quite useful for small N, become increasingly impractical as N increases.

Another possibility for N noncoplanar points would be to first determine an equation $A_k x + B_k y + C_k z + D_k = 0$ containing at least one given point and such that

$$A_k x_i + B_k y_i + C_k z_i + D_k \leq 0, \text{ for all } i \text{ where } 1 \leq i \leq N,$$

or

$$A_k x_i + B_k y_i + C_k z_i + D_k \geq 0, \text{ for all } i \text{ where } 1 \leq i \leq N.$$

We then, if necessary, rotate the plane determined by the above equation about a line containing a given point until it contains a second given point. Then, if necessary, rotate the plane about the line determined by these two given points until it contains a third given point not collinear with the first two. By systematic rotations of this type the convex hull could be determined. If the given N points are coplanar we would use lines instead of planes and proceed in the same manner.

It seems however that it would be advantageous, if N is large, to eliminate as soon as possible any point that is not a vertex, if this could be readily determined. The algorithm described in the following pages was developed with this idea in mind.

The method involves obtaining a sequence of convex hulls determined by certain subsets of the original set of N points. We refer to these convex hulls as $H_1,..,H_f$, where H_f, which we will refer to as the <u>final convex hull</u>, is the convex hull of the given N points. Each convex hull is an approximation to the final convex hull. The convex hull, H_{j+1}, where $1 \leq j+1 \leq f$, being a better approximation to the final convex hull than H_j.

If P is a point of H_j, $j = 1,..,f-1$, then P is a point of H_{j+1}, i.e., H_{j+1}, where $1 \leq j + 1 \leq f$, contains $H_1,..,H_j$. Hence if a given point is contained in H_j but is not a vertex of H_j we may eliminate it as a possible vertex of the final convex hull, H_f.

Generally speaking we determine H_1 and then check to see if any of the given N points other than the vertices of H_1 are contained in H_1. If there are any we eliminate them and let N_1 be the number of points remaining. We then check H_1 to see if it is the final convex hull. If not we determine a new convex hull H_2 and then check to see if any of the given N points other than the vertices of H_2 are contained in H_2. We continue in this manner obtaining successive hulls and eliminating points if possible until after a finite number of approximations we obtain the final convex hull.

Chapter 3

COMPUTER PROCEDURE

The following is a general description of the computer procedure for determining the convex hull of a given set of N distinct points. If the convex hull is determined by hand the same procedure can be followed, but in many cases on inspection of the data shortcuts in the procedure will become evident. The procedure is applicable for all positive values of N. However since the convex hull is trivial if $N < 3$ we will assume in the following discussion that $N \geq 3$. The maximum value for N depends of course on the computer. For the IBM 1620 with 40,000 storage positions the maximum value for N is 51.

We begin by determining the following six points from the given set of N points:

$$(x_{\ell x}, y_{\ell x}, z_{\ell x}), \text{ where } x_{\ell x} \geq x_i,\ i = 1,..,N;$$
$$(x_{sx}, y_{sx}, z_{sx}), \text{ where } x_{sx} \leq x_i,\ i = 1,..,N;$$
$$(x_{\ell y}, y_{\ell y}, z_{\ell y}), \text{ where } y_{\ell y} \geq y_i,\ i = 1,..,N;$$
$$(x_{sy}, y_{sy}, z_{sy}), \text{ where } y_{sy} \leq y_i,\ i = 1,..,N;$$
$$(x_{\ell z}, y_{\ell z}, z_{\ell z}), \text{ where } z_{\ell z} \geq z_i,\ i = 1,..,N;$$
$$(x_{sz}, y_{sz}, z_{sz}), \text{ where } z_{sz} \leq z_i,\ i = 1,..,N.$$

We will on occasion refer to these points as $P_{\ell x}$, P_{sx}, $P_{\ell y}$, P_{sy}, $P_{\ell z}$, P_{sz} respectively, rather than by specifying their coordinates. Let $\mathcal{P}$ be the set of points $P_{\ell x}$, P_{sx}, $P_{\ell y}$, P_{sy}, $P_{\ell z}$, P_{sz}. We could let H_1 be the convex hull determined by the points in $\mathcal{P}$. It appears at first glance that this would be a good choice for our first approximation. On further investigation however we find that many times it would not be. For instance the points in $\mathcal{P}$ may be coplanar even though the given N points are not. For a proof of this statement see Theorem 4.1. In cases like this the convex hull of the six points in $\mathcal{P}$ is not a good approximation to the final convex hull.

Even if the points in $\mathcal{P}$ are not coplanar a polyhedron with these six points as vertices is not necessarily a convex polyhedron. For a proof of this statement see Theorem 4.2. A primary disadvantage of using a polyhedron that is not convex is that it is not easily determined which, if any, of the given points are inside.

While we are interested in making a good choice for H_1, the first approximating convex hull, we must keep in mind the amount of effort expended in obtaining it. We therefore satisfy ourselves with a first approximation that is not as good as the convex hull determined by all of the points in $\mathcal{Q}$ but one that requires less effort to obtain. We obtain our first approximation in the following manner.

First we check to see if $x_{\ell x} = x_{sx}$. If $x_{\ell x} \neq x_{sx}$ we reorder the given points, if necessary, so that $P_{\ell x}$ is the first point and P_{sx} is the second point. In general the k-th point will be the point P_k with coordinates x_k, y_k and z_k. Using the equation

$$(3.1)\qquad (y_2 - y_1)x + (x_1 - x_2)y + (x_2 - x_1)y_1 + (y_1 - y_2)x_1 = 0,$$

and the equation

$$(3.2)\qquad (z_2 - z_1)x + (x_1 - x_2)z + (x_2 - x_1)z_1 + (z_1 - z_2)x_1 = 0,$$

we determine equations of the line containing P_1 and P_2 (5, p. 82). If $x_{\ell x}=x_{sx}$ we check to see if $y_{\ell y}=y_{sy}$. If $y_{\ell y}\neq y_{sy}$ we reorder the given points, if necessary, so that $P_{\ell y}$ is the first point and P_{sy} is the second point. Using the equation (3.1) and the equation

$$(3.3)\qquad (z_2 - z_1)y + (y_1 - y_2)z + (y_2 - y_1)z_1 + (z_1 - z_2)y_1 = 0,$$

we determine equations of the line containing P_1 and P_2.

If $x_{\ell x}=x_{sx}$ and $y_{\ell y}=y_{sy}$ we know that $z_{\ell z}\neq z_{sz}$ since the given N points are distinct and we are assuming $N \geq 3$. Hence if $x_{\ell x}=x_{sx}$ and $y_{\ell y}=y_{sy}$ we reorder the given points, if necessary, so that $P_{\ell z}$ is the first point and P_{sz} is the second point. Using equations (3.2) and (3.3) we determine equations of the line containing P_1 and P_2.

We now check to see if the given N points are collinear. We do this by substituting the coordinates of the points $P_1,..,P_N$ into the left member of each of the two equations determined above noting the value obtained in each case. If all the values obtained are zero the given N points are collinear (5, p. 82).

If the given N points are collinear the two points P_1 and P_2 are the endpoints of the line segment containing the given N points and the convex hull has been determined. The proof of this statement is not included but is straightforward using the definitions for a convex combination and a convex hull.

If the given N points are not collinear we need a third point to use with P_1 and P_2. For our third point we choose a point P_k, $1 \leq k \leq N$, such that the perpendicular distance between the point P_k and the line containing the two points P_1 and P_2 is greater than or equal to the perpendicular distance between the line containing the two points P_1 and P_2 and any other given point P_i, $1 \leq i \leq N$.

We do this in the following manner. For each point P_i, $i = 1,..,N$, we determine the value V_i given by the equation

$$\begin{aligned} V_i = & \left[(y_2 - y_1)(z_2 - z_1) - (z_2 - z_i)(y_2 - y_1)\right]^2 \\ & + \left[(z_2 - z_1)(x_2 - x_1) - (x_2 - x_i)(z_2 - z_1)\right]^2 \\ & + \left[(x_2 - x_1)(y_2 - y_1) - (y_2 - y_i)(x_2 - x_1)\right]^2, \end{aligned}$$

where (x_1,y_1,z_1) and (x_2,y_2,z_2) are the points P_1 and P_2. From the values $V_1,..,V_N$ we pick a value V_k satisfying the condition $V_k \geq V_j$, $j = 1,..,N$. We use the point P_k as our third point. See Theorem 4.3 for proof of the statement that the perpendicular distance between the point P_k and the line containing the points P_1 and P_2 is greater than or equal to the perpendicular distance between the line containing P_1 and P_2 and any other given point.

We now reorder the given N points, if necessary, so that the k-th point becomes the third point and the third point becomes the k-th point.

In general to determine the coefficients of an equation containing three noncollinear points P_i, P_j and P_k

we use the equations

$$
(3.4)\quad
\begin{aligned}
A &= y_i z_j + y_j z_k + z_i y_k - y_k z_j - y_i z_k - y_j z_i,\\
B &= x_k z_j + x_i z_k + x_j z_i - x_i z_j - x_j z_k - x_k z_i,\\
C &= x_i y_j + x_j y_k + y_i x_k - y_j x_k - x_i y_k - x_j y_i,\\
D &= x_k y_j z_i + y_k z_j x_i + x_j y_i z_k - x_j y_k z_i - y_i z_j x_k,\\
&\quad - x_i y_j z_k,
\end{aligned}
$$

where the equation of the plane is in the form $Ax + By + Cz + D = 0$ (5, p. 111).

Using the above equations determine the coefficients of the equation $A_o x + B_o y + C_o z + D_o = 0$ containing the points P_1, P_2 and P_3. The given N points are coplanar if and only if $A_o x_i + B_o y_i + C_o z_i + D_o = 0$, $i = 1,..,N$ (8, p.262).

The procedure for determining the convex hull has so far been the same for the coplanar case and the noncoplanar case. Hereafter however it varies slightly, the procedure for the noncoplanar case being more involved. We consider first the coplanar case.

To simplify the procedure we project the given N points onto a coordinate plane $A_c x + B_c y + C_c z + D_c = 0$ satisfying the condition

$$A_o A_c + B_o B_c + C_o C_c + \neq 0$$

The point P_k, $1 \leq k \leq N$, is a vertex of the convex hull determined by the points $P_1, \ldots, P_N$ if and only if the projection point corresponding to P_k is a vertex of the convex hull determined by the projection points corresponding to $P_1, \ldots, P_N$. For proof of this statement see Theorem 4.4. Therefore we need not determine the convex hull of the points $P_1, \ldots, P_N$ but may instead determine the convex hull of the projection of these points onto an appropriate coordinate plane.

The choice of which coordinate plane to use is arbitrary as long as we satisfy the condition $A_oA_c + B_oB_c + C_oC_c \neq 0$. If this condition is not satisfied the plane containing the given N points is perpendicular to the coordinate plane chosen and the resulting projection is a line segment (11, p. 234). If we project the given N points onto a coordinate plane we are in effect setting one of the x, y or z coordinates equal to zero. We then have only two coordinates to work with which simplifies the procedure.

To satisfy the condition $A_oA_c + B_oB_c + C_oC_c \neq 0$ we first check to see if $A_o = 0$. If $A_o \neq 0$ we project the given N points onto the y-z plane. If $A_o = 0$ we check to see if $B_o = 0$. If $A_o = 0$ and $B_o \neq 0$ we project the given N points onto the x-z plane and if $A_o = 0$ and $B_o = 0$ we project the given N points onto the x-y plane.

For convenience we will refer to the coordinate plane onto which we have projected the given N points as the x-y plane, where the point (x_i, y_i) is the projection of the point (x_i, y_i, z_i), $i = 1, \ldots, N$, onto the x-y plane. Hereafter, in discussing the coplanar case, when we refer to one of the given N points we will be referring to the projection of that point onto the x-y plane unless otherwise specified.

Suppose that we have a convex hull H with supporting lines $A_i x + B_i y + C_i = 0$, $i = 1, \ldots, k$, and vertices $P_1, \ldots, P_j$. Suppose further that the equations have been normalized so that if P_m is a vertex of H and

$$A_i x_m + B_i y_m + C_i \neq 0, \quad 1 \leq i \leq k,$$

then

$$A_i x_m + B_i y_m + C_i < 0.$$

We define the <u>inside</u> of H to be the set of all point P_u satisfying the condition

$$A_i x_u + B_i y_u + C_i < 0, \quad i = 1, \ldots, k.$$

We define the <u>outside</u> of H to be the set of all points P_u satisfying the condition

$$A_i x_u + B_i y_u + C_i > 0,$$

for at least one value of i where $1 \leq i \leq k$.

We define the <u>boundary</u> of H to be the set of all points P_u satisfying the condition

$$A_i x_u + B_i y_u + C_i \leq 0, \quad i = 1,\ldots,k,$$

with equality holding for at least one value of i where $1 \leq i \leq k$.

Points that are inside H we shall call <u>inside points</u>, points that are outside H we shall call <u>outside points</u> and points on the boundary of H we shall call <u>boundary points</u> (6, p. 201), (2, p. 38) and (11, p. 110). From the definition of a boundary point and the definition of a vertex it follows that all vertices are boundary points but not all boundary points are vertices.

The <u>points of H</u> will be the set of all inside points of H and the set of all boundary points of H, i.e., the point P_m is a point of H if and only if

$$A_i x_m + B_i y_m + C_i \leq 0, \; i = 1,\ldots,k.$$

We call the convex hull determined by the points P_1, P_2 and P_3, H_1. None of the vertices of the final convex hull are inside H_1. The proof of this statement is not included but it follows immediately from the fact that a point inside H_1 can be expressed as a convex combination of the points P_1, P_2 and P_3 (6, p. 218-19). Thus if any of the given N points are inside H_1 we may eliminate these points as potential vertices of the final convex hull.

To determine if any of the points $P_4,\ldots,P_N$ are inside H_1 we first determine the point C with coordinates c_1 and c_2 where

$$c_1 = (x_1 + x_2 + x_3)/3,$$
$$c_2 = (y_1 + y_2 + y_3)/3.$$

The point C is inside H_1. The proof of this statement is not included but it is easily obtained on substituting $(x_1 + x_2 + x_3)/3$ for x and $(y_1 + y_2 + y_3)/3$ for y in each of the equations of the supporting lines of H_1.

In general to determine an equation of the line containing the points P_k and P_m we will use the equation

$$(3.5) \quad (y_k - y_m)x + (x_m - x_k)y + (x_k - x_m)y_m + (y_m - y_k)x_m = 0 \quad (5,p.22).$$

Using the above equation determine an equation of the line containing the points P_1 and P_2, an equation of the line containing the points P_1 and P_3 and an equation of the line containing the points P_2 and P_3. Each of these equations is an equation of a supporting line of H_1. Now substitute the coordinates of the point C into the left member of each of the equations obtained above and in each case if the value obtained is positive multiply the equation through by (-1). The point C is not contained in any of the lines determined by the above equations and hence the value obtained will be either positive or negative. We will refer to the three equations obtained above as the equations $A_i x + B_i y + C_i = 0$, $i = 1,2,3$. In general we will use the symbol NE to denote the number of equations of supporting lines that we have. For H_1, NE = 3.

It is perhaps worthwhile to say a word about the notation we are introducing. In most cases this is the same notation used in the Fortran programs in Chapter 5. The use of descriptive symbols in programming is quite justifiable, and we introduce these symbols here to simplify the study of Chapter 5.

After we have so to speak normalized the equations $A_i x + B_i y + C_i = 0$, $i = 1,2,3$, so that $A_i c_1 + B_i c_2 + C_i < 0$, $i = 1,2,3$, we will refer to a point P_k as an inside point with respect to the line if $A_i x_k + B_i y_k + C_i < 0$ and an outside point with respect to the line if $A_i x_k + B_i y_k + C > 0$.

We eliminate the inside points of H_1 if any, as possible vertices of the final convex hull and let N_1 be the number of given N points still to be considered as possible vertices of the final convex hull ($N_1 \leq N$).

We now substitute the coordinates of the points P_k, $k = 1,..,N_1$, into the left member of the equation $A_1x + B_1y + C_1 = 0$, letting U_k be the value obtained in each case. If $U_k \leq 0$, $k = 1,..,N_1$, then, by definition, the line determined by $A_1x + B_1y + C_1 = 0$ is the equation of a supporting line of the final convex hull. If the line determined by $A_1x + B_1y + C_1 = 0$ is a supporting line we proceed in the same manner to determine if the equations $A_ix + B_iy + C_i = 0$, $i = 2,3$, are equations of supporting lines. If each of the equations $A_ix + B_iy + C_i = 0$, $i = 1,2,3$, is an equation of a supporting line of the given N points the final convex hull has been determined. For proof of this statement see Theorem 4.5.

Suppose now that one of the above equations is not an equation of a supporting line of the given N points. We will refer to this equation as equation NETC, $1 \leq \text{NETC} \leq \text{NE}$. (Number of Equation To be Checked) From the values U_k, $k = 1,..,N_1$, obtained by substituting the coordinates of the points $P_1,..,P_{N_1}$ into the left member of the equation NETC, we determine a value U_m such that $U_m \geq U_k$, $k = 1,..,N_1$. The point P_m corresponding to the value U_m is an outside point with respect to the plane determined by equation NETC. The perpendicular distance between the line determined by the equation NETC and the point P_m is greater than or equal to the perpendicular distance between the line determined by the equation NETC and any other given point P_k, $1 \leq k \leq N_1$, that is an outside point with respect to the line determined by equation NETC. For proof of this statement see Theorem 4.6. We use the point P_m as a vertex for our second approximation to the final convex hull. We call this second approximation H_2.

We reorder the given N_1 points, if necessary, so that the m-th point becomes the fourth point and the fourth point becomes the m-th point. The points P_1, P_2 and P_3 were the vertices of H_1. We will use the symbol NV to denote the number of given points used as vertices thus far, i.e., for H_1, NV = 3 and for H_2, NV = 4. Thus after we reorder the points we may refer to the point P_4 as the point P_{NV}.

Let MM_{NETC} and NN_{NETC} be the vertices of H_1 used to determine the equation NETC. Using equation (3.5) determine an equation of the line containing the point MM_{NETC} and the point P_{NV} and an equation of the line containing the point NN_{NETC} and the point P_{NV}. Now normalize each of the above equations and call them equations 4 and 5 respectively. H_2 will have the same supporting lines as H_1 with the exception that the supporting line determined by the equation NETC will be replaced by the supporting lines determined by equations 4 and 5. We now reorder the equations eliminating equation NETC, i.e., the (i + 1)-st equation replaces the i-th equation, i = NETC,..,4. In general when we say we reorder the equations 1,..,NE removing equation k we mean that the (i + 1)-st equation replaces the i-th equation i = k,..,NE-1. We now have NE = 4. Hull H_2 is the convex hull of the points $P_1,..,P_4$. For proof of this statement see Theorem 4.5. We continue with H_2 as we did with H_1 checking to see if any of the points P_k, k = NV+1,..,N_1, are inside H_2.

In general let H_j, j = 2,..,f, be the j-th approximation to the final convex hull which we call H_f, where 1 $\leq j \leq f$. We first check H_j to eliminate inside points, if any, in the same manner as for H_1 and let N_j be the number of given N points still under consideration.

$(N_j \leq N_{j-1}, \ldots, \leq N)$

We now substitute the coordinates of the points P_k, $k = 1, \ldots, N_j$, into the left member of the equation $A_{NETC}x + B_{NETC}y + C_{NETC} = 0$, $1 \leq NETC \leq NE$, letting U_k be the value obtained in each case.

The index NETC takes on the values 1,..,NE where the initial value of NETC for H_j, $1 < j \leq f$, is the last value of NETC for H_{j-1}. This follows from the fact that the equations 1,..,NETC-1 have already been determined to be equations of supporting lines of the final convex hull and need not be checked for each successive hull determined.

If $U_k \leq 0$, $k = 1, \ldots, N_j$, the line determined by equation NETC is, by definition, an equation of a supporting line of the final convex hull. If the line determined by the equation NETC is a supporting line we check to see if NE > NETC. If it is we increase NETC by one and check equation NETC to see if it is the equation of a supporting line in the same manner as for equation NETC-1. We continue checking for supporting lines until either (1) NE = NETC or (2) for some value of NETC the equation NETC is not the equation of a supporting line, i.e., there is at least one point P_k for which the value $U_k > 0$. In this case we determine a new vertex P_m corresponding to the value U_m where $U_m \geq U_k$, $k = 1,\ldots,N_j$. We then add one to NV and reorder the points, if necessary, so that the m-th point becomes the NV-th point, and the NV-th point becomes the m-th point. Now using equation (3.5) determine the equation NE+1 containing the points MM_{NETC} and P_{NV} and the equation NE+2 containing the points NN_{NETC} and P_{NV}. We then normalize each of the above equations, and reorder the equations 1,..,NE+2 removing equation NETC. We have added two equations and eliminated one, thus we add one to NE. We now increase j by one and start over again checking H_j for inside points in the same manner as for H_{j-1}. H_j is the convex hull determined by the points $P_1,\ldots,P_{NV}$. For proof of this statement see Theorem 4.5. We continue in the above manner obtaining successive approximations to the final convex hull until after a finite number of steps we have NE = NETC. Clearly this process comes to an end in a finite number of steps since we can add another vertex at most N-3 times. If NE = NETC all of the supporting lines of the final convex hull have been determined and this hull, H_f, is the convex hull of the given N points. For proof of this statement see Theorem 4.5.

Once we have NE = NETC we check to see if $N_f > NV$.

If it is the points $P_{NV+1},\ldots,P_{N_f}$ are boundary points of the final convex hull. This follows from the fact that the inside points have been eliminated and the vertices are among the points $P_1,\ldots,P_{NV}$. Not all of the points $P_1,\ldots,$ P_{NV} are necessarily vertices of the final convex hull. Each point was a vertex for one or more of the approximations to the final convex hull but a point $P_1,\ldots,P_{NV}$ may be contained in only one supporting line of H_f.

To determine which of the points $P_1,\ldots,P_{NV}$ are vertices we substitute the coordinates of the points $P_1,\ldots,$ P_{NV} into the left member of each of the equations $A_kx +$ $B_ky + C_k = 0$, $k = 1,\ldots,NE$. For each equation $A_kx + B_ky +$ $C_k = 0$, $1 \leq k \leq NE$, that P_i, $1 \leq i \leq NV$, satisfies we check to see if the point P_i is either of the points NN_k or MM_k, if it is we are not interested in it since we are only interested in determining if there is a point P_i, $1 \leq i \leq NV$ satisfying the equation $A_kx + B_ky + C_k = 0$ other than the two points NN_k and MM_k. Hence if P_i is either of the points NN_k or MM_k we check to see if $NV > i$. If $NV > i$ we increase i by one and continue checking. If the point P_i is neither of the points NN_k or MM_k we check to see which of the three points P_i, NN_k or MM_k is contained in the line segment determined by the other two.

We do this in the following way. Using the formula

$$D_j = \sqrt{(x_e - x_d)^2 + (y_e - y_d)^2}$$

for the distance D_j between the points P_d and P_e we determine the distance D_1 between the points NN_k and MM_k, the distance D_2 between the points MM_k and P_i and the distance D_3 between the points NN_k and P_i (9, p. 33). We then determine the value of $D_1 + D_2 - D_3$. If the value obtained is zero the point MM_k is not a vertex of the final convex hull since it is contained in the line segment determined by P_i and NN_k (9, p. 29). If the value obtained is not zero we determine the value of $D_1 + D_3 - D_2$. If this value is zero the point NN_k is not a vertex of the final convex hull since it is contained in the line segment determined by MM_k and P_i. If $D_1 + D_2 - D_3 \neq 0$ and $D_1 + D_3 - D_2 \neq 0$ the point P_i is not a vertex of the final convex hull since it is contained in the line segment determined by MM_k and NN_k.

Once we determine which of the points P_i, MM_k or NN_k is not a vertex we reorder the points $P_1, \ldots, P_{NV}$ eliminating that point and at the same time we subtract one from NV.

We want to have only one equation of each supporting line of the final convex hull, and since we have at least two equations of the supporting line containing the points P_i, NN_k and MM_k we reorder the equations of the supporting lines removing the equation k, and at the same time we subtract one from NE.

We continue in the above manner checking each point P_i, $i = 1,\ldots,NV$, until we have determined which of the NV points are vertices of the final convex hull and have removed all but one equation for each supporting line. We have now determined which of the given N points are inside the convex hull, which of the given N points are boundary points of the final convex hull, which of the given N points are vertices of the final convex hull and we have determined an equation for each supporting line. This completes the procedure for the coplanar case.

Since much of the procedure for the noncoplanar case parallels the coplanar case, making the extension for the third dimension, it would be repetitious to describe the noncoplanar case in as much detail as we have the coplanar case. Thus we will give a general description of the noncoplanar case where it parallels the coplanar case and go into a more detailed description at the point where the two procedures differ.

For the noncoplanar case we begin with the values U_k, $k = 1,..,N$, obtained by substituting the coordinates of each of the given N points into the left member of the equation $A_o x + B_o y + C_o z + D_o = 0$ containing the points P_1, P_2 and P_3. For the coplanar case H_1 is a triangle determined by the points P_1, P_2 and P_3. For the noncoplanar case we define H_1 to be a tetrahedron determined by the points P_1, P_2, P_3 and the point P_m corresponding to a value U_m determined above where $U_m \geq U_k$, $k = 1,..,N$. After we have determined the point P_m we reorder the points so that the m-th point becomes the fourth point and the fourth point becomes the m-th point.

After we have determined the coefficients of the equations of the supporting planes of H_1 using the equations (3.4) and the points P_1, P_2, P_3 and P_4 we determine the point C and proceed to determine if any of the given N points are inside H_1. We eliminate these points, if any, and then check to see if H_1 is the convex hull of the given N points.

If the tetrahedron H_1 is not the convex hull of the given N points we proceed to determine a new convex hull H_2. In the coplanar case to determine H_j from the convex hull H_{j-1}, $j = 2,..,f$, we essentially added a triangle to

H_{j-1} by replacing the supporting line determined by the equation NETC by two new supporting lines; one containing the points P_{NV} and MM_{NETC} and the other containing the points P_{NV} and NN_{NETC}. For the noncoplanar case we add a tetrahedron to H_{j-1} replacing the supporting plane determined by equation NETC by three new supporting planes; one containing the points P_{NV}, MM_{NETC} and NN_{NETC}, one containing the points P_{NV}, MM_{NETC} and KK_{NETC} and one containing the points P_{NV}, NN_{NETC} and KK_{NETC}.

We will make use of the same symbols for the noncoplanar case that we used for the coplanar case and we adopt the same definitions for inside, outside, inside points, outside points, and boundary points, assuming the extension for the third dimension when necessary. We shall refer to the given point that is a vertex of H_j but not a vertex of H_{j-1} as the point P_{NV}. We will refer to the equations of the supporting planes of H_j as the equations $A_i x + B_i y + C_i z + D_i = 0$, $i = 1,\ldots,NE$. For each equation i we shall refer to the given points used to determine the equation as the points MM_i, NN_i and KK_i. As in the coplanar case we will use the symbol NE to denote the number of equations of supporting planes for H_j, the symbol NV to denote the number of given points used as vertices so far and we will use the symbol NETC to denote the number of the equation that we are checking to determine if it is the equation of a supporting plane of the final convex hull. We refer to the equations that have already been determined to be equations of supporting planes of the final convex hull as the equations 1,.., NETC-1.

In the coplanar case the point P_{NV} could not be on the outside of more than one of the equations $A_k x + B_k y + C_k = 0$, $k = 1,..,NE$. For proof of this statement see Theorem 4.7. In the noncoplanar case however the point P_{NV} can be on the outside of more than one of the equations $A_k x + B_k y + C_k z + D_k = 0$, $k = 1,..,NE$, and thus the polyhedron determined by the equations $A_k x + B_k y + C_k z + D_k = 0$, $k = 1,..,NE$, is not necessarily convex. For an example of such a case see Theorem 4.8.

We check to see if the polyhedron is convex by substituting the coordinates of the point P_{NV} into the left member of each of the equations $A_k x + B_k y + C_k z + D_k = 0$, $k = NETC,..,NE$, checking the value obtained in each case. If all the values obtained are negative or zero the polyhedron determined by the above equations is convex. The proof is similar to the proof given for Theorem 4.5 and is not included. Suppose now that not all of the values obtained are negative or zero. Let $A_g x + B_g y + C_g z + D_g = 0$ be one of the NE equations determined above for which $A_g x_{NV} + B_g y_{NV} + C_g z_{NV} + D_g > 0$. Using equations (3.4) determine the coefficients of the plane containing the points MM_g, NN_g and P_{NV}, the coefficients of an equation of the plane containing the points MM_g, KK_g and P_{NV} and the coefficients of an equation of the plane containing the points KK_g, NN_g and P_{NV}. Normalize each of these equations and call them NE+1, NE+2 and NE+3 respectively. Now increase NE by two and reorder the equations eliminating equation g.

We continue checking each equation g where g takes on the values NETC+1,..,NE. For each value of g for which $A_g x_{NV} + B_g y_{NV} + C_g z_{NV} > 0$, $1 \leq g \leq NE$, we repeat the above procedure until we have $A_k x_{NV} + B_k y_{NV} + C_k z_{NV} + D_k$ 0, $k = 1,..,NE$.

Suppose that we eliminated two equations i and m for which $MM_i = MM_j$, and $NN_i = NN_j$ and such that $A_i x_{NV} + B_i y_{NV} + C_i z_{NV} + D_i > 0$ and $A_m x_{NV} + B_m y_{NV} + C_m z_{NV} + D_m > 0$. In expanding the convex hull we replaced equation i by three new equations, one containing MM_i, NN_i and P_{NV}, one containing MM_i, KK_i and P_{NV} and one containing NN_i, KK_i and P_{NV}. For convenience we will call these equations i_1, i_2 and i_3 respectively. Similarly we replaced equation m by three new equations, one containing MM_m, NN_m and P_{NV}, one containing MM_m, KK_m and P_{NV} and one containing NN_m, KK_m and P_{NV}. We call these m_1, m_2 and m_3 respectively. Now since $MM_i = MM_m$ and $NN_i = NN_m$ it follows that equation i_1 and equation m_1 are equations of the same plane and further this plane is not a supporting plane since KK_i is contained in one section determined by the plane and KK_m in another, neither KK_i nor KK_m being contained in the plane.

We therefore reorder the equations 1,..,NE eliminating equations i_1 and m_1 and at the same time we subtract two from NE. We repeat the above procedure eliminating equations j and k for which $MM_j = MM_k$, $NN_j = NN_k$ and $KK_j = KK_k$, $k \neq j$, until the equations $A_ix + B_iy + C_iz + D_i = 0$, $i = 1,..,NE$, are equations of the supporting planes of the convex hull determined by the points $P_1,..,P_{NV}$. We now increase j by one and start over again checking H_j for inside points proceeding in the same manner as for H_1.

We continue in the above manner obtaining successive approximations to the final convex hull until after a finite number of steps we have NETC = NE. As in the coplanar case this process comes to an end after a finite number of steps since we can add another vertex at most N-4 times. If NETC = NE all the supporting planes of the final convex hull have been determined and this hull, H_f, is the convex hull of the given N points. The proof of this last statement is similar to the proof given for Theorem 4.5 and is not included.

Once we have NETC = NE we check to see if $N_f > NV$. If it is the points $P_{NV+1},\ldots,P_{N_f}$ are boundary points of the final convex hull. This follows from the fact that the inside points have been eliminated and the vertices are contained among the points $P_1,\ldots,P_{NV}$. We now want to determine if we have more than one equation for each supporting plane. We do this by checking to see if there is an equation j containing the points NN_i, MM_i and KK_i where $i \neq j$. If there exists such an equation j we reorder the equations eliminating equation j, and at the same time subtract one from NE. We do this for each equation j where j = 1,..,NE, and for each value of j, i takes on the values j+1,..,NE.

Not all of the points $P_1,\ldots,P_{NV}$ are necessarily vertices of the final convex hull. A necessary condition for a point to be a vertex of the final convex hull is that it be contained in at least three distinct supporting planes of the convex hull. This follows from the fact that if P_k is contained in at most two supporting planes of the convex hull then P_k can be expressed in terms of the endpoints of the line segment determined by these two supporting planes and thus is not a vertex (6, p. 195). Each of the points $P_1,\ldots,P_{NV}$ was a vertex for at least one approximation to the final convex hull but it is possible for a point P_i, $1 \le i \le NV$, to be contained in only one or two supporting planes of the final convex hull.

To eliminate these points, if any, that are not vertices we substitute the coordinates of each of the points into the left member of the equations $A_k x + B_k y + C_k z + D_k = 0$, $k = 1,\ldots,NE$, and if the point is not a vertex it will satisfy at most two of the above equations.

If we determine that one of the points $P_1,\ldots,P_{NV}$ is not a vertex we reorder the points eliminating that point and at the same time we subtract one from NV.

We have thus determined which of the given N points are inside the convex hull, which of the given N points are boundary points of the final convex hull, which of the given N points are vertices of the final convex hull and we have determined an equation for each of the supporting planes. This completes the procedure for determining the convex hull for the noncoplanar case.

Chapter 4

SUPPORTING THEOREMS

THEOREM 4.1 Given a set of N distinct noncoplanar points $P_1, \ldots, P_N$, let $P_{\ell x}$, P_{sx}, $P_{\ell y}$, P_{sy}, $P_{\ell z}$, P_{sz} be points among the given N such that

$$
\begin{aligned}
x_{\ell x} &\geq x_i, \quad 1 \leq i \leq N, \\
x_{sx} &\leq x_i, \quad 1 \leq i \leq N, \\
y_{\ell y} &\geq y_i, \quad 1 \leq i \leq N, \\
y_{sy} &\leq y_i, \quad 1 \leq i \leq N, \\
z_{\ell z} &\geq z_i, \quad 1 \leq i \leq N, \\
z_{sz} &\leq z_i, \quad 1 \leq i \leq N.
\end{aligned}
$$

The points $P_{\ell x}$, P_{sx}, $P_{\ell y}$, P_{sy}, $P_{\ell z}$, P_{sz} may be coplanar whether or not they are distinct.

PROOF: The above statement is clearly true if there are at most three distinct points among $P_{\ell x}$, P_{sx}, $P_{\ell y}$, P_{sy}, $P_{\ell z}$, P_{sz}. We will show by an example however that the above statement is also true in cases where there are six distinct points among $P_{\ell x}$, P_{sx}, $P_{\ell y}$, P_{sy}, $P_{\ell z}$, P_{sz}.

Consider the following set of nine points:

$P_1 = (9,6,-8)$ $\quad$ $P_4 = (6,4,5)$ $\quad$ $P_7 = (-11,0,6)$

$P_2 = (10,0,-8)$ $\quad$ $P_5 = (8,-2,-10)$ $\quad$ $P_8 = (-8,-3,-1)$

$P_3 = (-1,7,11)$ $\quad$ $P_6 = (4,5,8)$ $\quad$ $P_9 = (-10,4,12)$

For this set of points we have

$$P_{\ell x} = P_2 \qquad P_{\ell y} = P_3 \qquad P_{\ell z} = P_9$$

$$P_{sx} = P_7 \qquad P_{sy} = P_8 \qquad P_{sz} = P_5.$$

The plane determined by the equation $2x - 5y + 3z + 4 = 0$ contains the points $P_{\ell x}$, P_{sx}, $P_{\ell y}$, P_{sy}, $P_{\ell z}$, P_{sz} but it does not contain the points P_1, P_4 and P_6. This completes the proof.

THEOREM 4.2 A polyhedron with vertices $P_1, \ldots, P_6$ where

$$x_1 \geq x_i,\ 1 \leq i \leq 6,$$

$$x_2 \leq x_i,\ 1 \leq i \leq 6,$$

$$y_3 \geq y_i,\ 1 \leq i \leq 6,$$

$$y_4 \leq y_i,\ 1 \leq i \leq 6,$$

$$z_5 \geq z_i,\ 1 \leq i \leq 6,$$

$$z_6 \leq z_i,\ 1 \leq i \leq 6,$$

is not always a convex polyhedron.

PROOF: The proof will be by example. Consider the following set of six points:

$P_1 = (5,0,0)$ $\quad$ $P_3 = (3,5,7/4)$ $\quad$ $P_5 = (0,0,4)$

$P_2 = (-2,0,0)$ $\quad$ $P_4 = (3,-5,7/4)$ $\quad$ $P_6 = (0,0,-2)$

These six points satisfy the initial conditions of THEOREM 4.2 and thus it remains to show that there exists a polyhedron with these six points as vertices that is not convex. Figure 1 is an illustration of such a polyhedron. Equations of the faces of the polyhedron as illustrated in Figure 1, and a corresponding list of the vertices contained in each face can be found in Table 1.

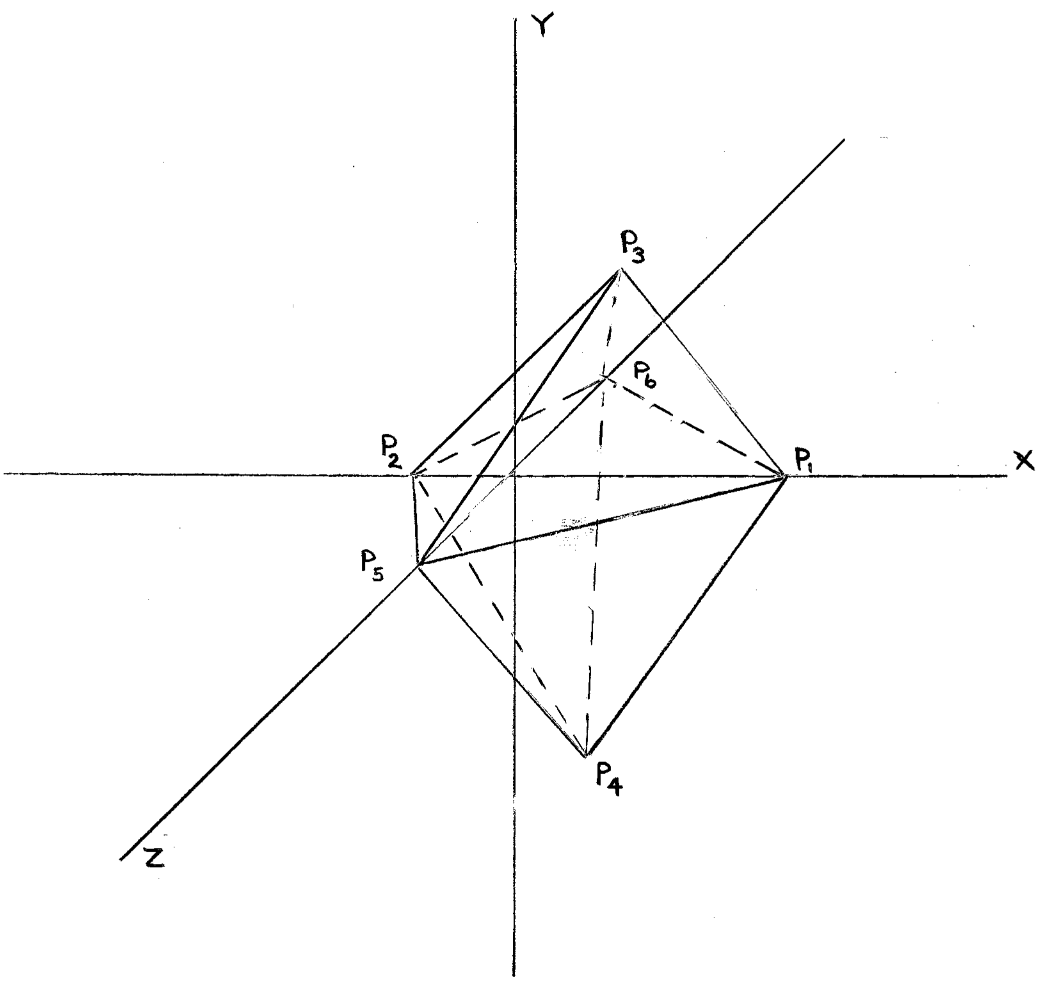


Figure 1

Table 1.

	EQUATION OF FACE	VERTICES IN FACE
1.	$-40x + 33y + 20z - 80 = 0$	P_5, P_2, P_3
2.	$-40x - 33y + 20z - 80 = 0$	P_5, P_2, P_4
3.	$-20x + 27y - 20z - 40 = 0$	P_2, P_6, P_3
4.	$-20x - 27y - 20z - 40 = 0$	P_2, P_6, P_4
5.	$40x + 51y - 100z - 200 = 0$	P_6, P_1, P_3
6.	$40x - 51y - 100z - 200 = 0$	P_6, P_1, P_4
7.	$80x - 3y + 100z - 400 = 0$	P_5, P_1, P_3
8.	$80x + 3y + 100z - 400 = 0$	P_5, P_1, P_4

Substituting the coordinates of each of the six points $P_1,..,P_6$ into the left member of equations 7 and 8 we see that they are not equations of supporting planes for the convex hull determined by the points $P_1,..,P_6$.

For equation 7 we have

$$A_7x_4 + B_7y_4 + C_7 > 0,$$

and

$$A_7x_i + B_7y_i + C_7 < 0, \ i = 2,6.$$

For equation 8 we have

$$A_8x_3 + B_8y_3 + C_8 > 0,$$

and

$$A_8x_i + B_8y_i + C_8 < 0, \ i = 2,6.$$

This completes the proof.

THEOREM 4.3 The perpendicular distance between the point P_k and the line containing the distinct points P_1 and P_2 is greater than or equal to the perpendicular distance between the point P_j and the line containing P_1 and P_2 if $V_k \geq V_j$ where, in general, the value V_i for the point P_i is obtained from the equation

$$\begin{aligned} V_i = & \left[(y_2 - y_1)(z_2 - z_1) - (z_2 - z_i)(y_2 - y_1)\right]^2 \\ & + \left[(z_2 - z_1)(x_2 - x_1) - (x_2 - x_i)(z_2 - z_1)\right]^2 \\ & + \left[(x_2 - x_1)(y_2 - y_1) - (y_2 - y_i)(x_2 - x_1)\right]^2. \end{aligned}$$

PROOF: The perpendicular distance d_i from the line containing the points P_1 and P_2 to the point P_i is given by the equation

$$\begin{aligned} d_i^2 = & \begin{vmatrix} y_2 - y_i & z_2 - z_i \\ \mu & \nu \end{vmatrix}^2 + \begin{vmatrix} z_2 - z_i & x_2 - x_i \\ \nu & \lambda \end{vmatrix}^2 \\ & + \begin{vmatrix} x_2 - x_i & y_2 - y_i \\ \lambda & \mu \end{vmatrix}^2 \end{aligned}$$

where λ, μ and ν are direction cosines of the line containing P_1 and P_2 (5, p. 96). We define the direction cosines of the line containing the points P_1 and P_2 as follows:

$$\lambda = \frac{x_2 - x_1}{(e)(D)}, \qquad \mu = \frac{y_2 - y_1}{(e)(D)}, \qquad \nu = \frac{z_2 - z_1}{(e)(D)},$$

where $D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$ and e is +1 or −1, the sign being chosen so that the first of the numbers $e(z_2 - z_1)$, $e(y_2 - y_1)$ and $e(x_2 - x_1)$ which is not zero shall be positive (5, p. 85). On substituting the above values for λ, μ and ν into the right member of the equation for V_i and simplifying we have

$$d_i^2 = \left[(y_2 - y_i)\frac{(z_2 - z_1)}{(e)(D)} - (z_2 - z_i)\frac{(y_2 - y_1)}{(e)(D)} \right]^2$$
$$+ \left[(z_2 - z_i)\frac{(x_2 - x_1)}{(e)(D)} - (x_2 - x_i)\frac{(z_2 - z_1)}{(e)(D)} \right]^2$$
$$+ \left[(x_2 - x_i)\frac{(y_2 - y_1)}{(e)(D)} - (y_2 - y_i)\frac{(x_2 - x_1)}{(e)(D)} \right]^2$$

and on multiplying both sides by e^2D^2 we obtain

$$\begin{aligned} d_i^2e^2D^2 = V_i = {} & \left[(y_2 - y_i)(z_2 - z_1) - (z_2 - z_i)(y_2 - y_1)\right]^2 \\ & + \left[(z_2 - z_i)(x_2 - x_1) - (x_2 - x_i)(z_2 - z_1)\right]^2 \\ & + \left[(x_2 - x_i)(y_2 - y_1) - (y_2 - y_i)(x_2 - x_1)\right]^2 . \end{aligned}$$

Since $e^2 = 1$ it follows that if $V_k \geq V_j$ then

$$d_k^2 D^2 \geq d_j^2 D^2.$$

Since P_1 and P_2 are distinct we know that $D^2 > 0$ and hence

$$d_k^2 \geq d_j^2.$$

Thus the perpendicular distance between the point P_k and the line containing the points P_1 and P_2 is greater than or equal to the perpendicular distance between the point P_j and the line containing the points P_1 and P_2. This completes the proof.

THEOREM 4.4 Let $\mathcal{H}$ be the unique convex hull determined by the distinct coplanar points $P_1, \ldots, P_N$. Let an equation of the plane containing $\mathcal{H}$ be $A_o x + B_o y + C_o z + D_o = 0$. Let $\mathcal{H}'$ be the projection of $\mathcal{H}$ in the coordinate plane $A_c x + B_c y + C_c z + D_c = 0$. If P_i is a point of $\mathcal{H}$ let P_i' be the projection point corresponding to the point P_i. If the plane containing $\mathcal{H}$ and the plane containing $\mathcal{H}'$ are not perpendicular, i.e., if $A_o A_c + B_o B_c + C_o C_c + D_o D_c \neq 0$ then P_k, $1 \leq k \leq N$, is a vertex of $\mathcal{H}$ if and only if P_k' is a vertex of $\mathcal{H}'$.

PROOF: In projecting a point onto a coordinate plane we are in effect setting one of the coordinates of the point equal to zero, the other two coordinates remaining unchanged. We consider first the projection of $\mathcal{H}$ onto the x-y plane assuming the plane determined by the equation $A_o x + B_o y + C_o z + D_o = 0$ is not perpendicular to the x-y plane.

If we project the points (x,y,z) of $\mathcal{H}$ onto the x-y plane we have the points (x,y) as the corresponding projection points.

We first want to show that the point (x_k,y_k,z_k) is a vertex of the convex hull determined by the points (x_i,y_i,z_i), $i = 1,..,N$, if the point (x_k,y_k) is a vertex of the convex hull determined by the points (x_i,y_i), $i = 1,..,N$.

By definition the point (x_k,y_k) is a vertex of the convex hull determined by (x_i,y_i), $i=1,..,N$, if and only if there do not exist points (x_a,y_a) and (x_b,y_b) of $\mathcal{H}'$ such that

$$x_k = (1 - \lambda)\, x_a + \lambda x_b$$
$$y_k = (1 - \lambda)\, y_a + \lambda y_b, \quad 0 < \lambda < 1.$$

Now if (x_k,y_k) cannot be written as a convex combination of (x_a, y_a) and (x_b, y_b) it follows that there do not exist points (x_a, y_a, z_a) and (x_b, y_b, z_b) of $\mathcal{H}$ such that

$$x_k = (1 - \lambda)\, x_a + \lambda x_b,$$
$$y_k = (1 - \lambda)\, y_a + \lambda y_b,$$
$$z_k = (1 - \lambda)\, z_a + \lambda z_b,\ 0 < \lambda < 1.$$

Thus the point P_k is a vertex of the convex hull determined by the points $P_1, \ldots, P_N$.

Suppose now that the point (x_k, y_k, z_k) is a vertex of the convex hull determined by the points (x_i, y_i, z_i), $i = 1, \ldots, N$, but the point (x_k, y_k) is not a vertex of the convex hull determined by the points (x_i, y_i), $i = 1, \ldots, N$.

Since (x_k, y_k) is not a vertex it follows that there exist two points (x_a, y_a) and (x_b, y_b) of $\mathcal{H}'$ such that

$$x_k = (1 - \lambda)\, x_a + \lambda x_b,$$
$$y_k = (1 - \lambda)\, y_a + \lambda y_b,\ 0 < \lambda < 1.$$

Now since (x_k, y_k, z_k) is a vertex it follows that

$$z_k \neq (1 - \lambda)\, z_a + \lambda z_b,\ 0 < \lambda < 1.$$

To prove that this does not occur we show that if it does the plane containing the points (x_k, y_k, z_k), (x_a, y_a, z_a) and (x_b, y_b, z_b), and hence all the given points, is perpendicular to the x-y plane contrary to our original assumption. Since equations of the x-y plane are in the form $C_c z = 0$, to show the perpendicularity condition,

$$A_oA_c + B_oB_c + C_oC_c + \neq 0,$$

is not satisfied all we need to show is that $C_o = 0$.

From (3.4) we have

$$C_o = x_a y_b + x_b y_k + y_a x_k - y_b x_k - x_a y_k - x_b y_a.$$

Now we substitute the value $(1-\lambda)x_a + \lambda x_b$ for x_k and the value $(1-\lambda)y_a + \lambda y_b$ for y_k in the equation given in (3.4) for C_o. Simplifying we have

$$\begin{aligned} C_o = {} & x_a y_b - x_a y_b \lambda - x_a y_b (1 - \lambda) + x_a y_b (1 - \lambda) + x_a y_b \lambda - \\ & x_a y_b + x_a y_a \lambda - x_a y_a \lambda + x_b y_b (1 - \lambda) - x_b y_b (1 - \lambda) \\ & = 0. \end{aligned}$$

Thus it follows that the plane containing the points (x_i, y_i, z_i), $i = 1,..,N$, is perpendicular to the coordinate plane containing the points (x_i, y_i), $i = 1,..,N$, contrary to our original assumption. Hence the point (x_k, y_k, z_k) is a vertex of the convex hull determined by the points (x_i, y_i, z_i), $i = 1,..,N$, only if (x_k, y_k) is a vertex of the convex hull determined by the points (x_i, y_i), $i = 1,..,N$.

This completes the proof for the case where the points are projected onto the x-y plane. The proof is the same for the x-z plane if we replace y by z and for the y-z plane if we replace x by z.

THEOREM 4.5 The polyhedron H_j, $1 \leq j \leq f$, obtained by the procedure described in Chapter 3, with supporting lines determined by the equations $A_i x + B_i y + C_i = 0$, $i = 1,..,NE$, and vertices $P_1,..,P_{NV}$, is the convex hull determined by the points $P_1,..,P_{NV}$.

PROOF: To prove the above statement we need to show that P_m is a point of H_j if and only if P_m can be expressed as a convex combination of the points $P_1,..,P_{NV}$. We begin by assuming that P_m is a convex combination of the points $P_1,..,P_{NV}$. Thus we have

$$x_m = \mu_1 x_1 + .. + \mu_{NV} x_{NV},$$

$$y_m = \mu_1 y_1 + .. + \mu_{NV} y_{NV},$$

where $\mu_i \geq 0$, $i = 1,..,NV$, and $\sum_{i=1}^{NV} \mu_i = 1$.

By definition P_m is a point of H_j if $A_i x_m + B_i y_m + C_i \leq 0$, $i = 1,..,NE$. Making the substitution $\sum_{i=1}^{NV} \mu_i x_i$ for x_m and $\sum_{i=1}^{NV} \mu_i y_i$ for y_m into the left member of the equations $A_i x + B_i y + C_i = 0$, $i = 1,..,NE$, we have

$$E = A_i\left(\sum_{j=1}^{NV} \mu_j x_j\right) + B_i\left(\sum_{j=1}^{NV} \mu_j y_j\right) + C_i, \quad i = 1,..,NE.$$

Rearranging the above terms we have

$$E = \mu_1(A_i x_1 + B_i y_1) + \ldots + \mu_{NV}(A_i x_{NV} + B_i y_{NV}) + C_i,$$

$$i = 1,..,NE$$

By construction we know that $A_i x_\ell + B_i y_\ell + C_i \leq 0$, for $i = 1,..,NE$ and $\ell = 1,..,NV$. Thus we have

$$E \leq \mu_1(-C_i) +,..,+ \mu_{NV}(-C_i) + C_i = -C_i \sum_{j=1}^{NV} \mu_j + C_i = 0.$$

Thus if P_m is a convex combination of the points $P_1,..,P_{NV}$ then P_m is a point of H_j.

We now want to prove that if P_m is a point of H_j then P_m can be written as a convex combination of the points $P_1,..,P_{NV}$. The proof will be by mathematical induction on j. We consider first the case for $j = 1$. If $j = 1$ we have $NV = 3$ and H_1 is a triangle with vertices P_1, P_2 and P_3. For proof of the statement that if P_m is a point of a triangle then P_m can be expressed as a convex combination of the vertices of the triangle see (6, p. 218-219). Using this proof we consider the statement to be true for $j = 1$. We now assume the statement to be true for $j = k$ and consider the case for $j = k + 1$.

To obtain H_{k+1} from H_k we in effect add a triangle to H . (Fig. 2)

Figure 2

By our induction hypothesis for j = k we know that if P_m is a point of H_k then P_m can be written as a convex combination of the points $P_1,..,P_{NV}$. For the case j = 1 we know that P_m can be written as a convex combination of the points P_1, P_2 and P_3. Thus for H_{k+1} if P_m is not a point of H_k but is a point of H_{k+1} we can express P_m as a convex combination of the points P_a, P_b and P_{NV+1} where $\mu_i = 0$, $i = 1,..,NV$, $i \neq a$, $i \neq b$. Thus H_{k+1} is the convex hull of the points $P_1,..,P_{NV+1}$. Hence if P_m is a point of H_j then P_m can be written as a convex combination of the points $P_1,..,P_{NV}$. This completes the proof.

THEOREM 4.6 Let $Ax + By + C = 0$ be the equation of a line in 2-space and let P_m and P_j be two points in the x-y plane satisfying the condition

$$Ax_i + By_i + C \geq 0, \text{ for } i = m \text{ and } i = j.$$

The perpendicular distance between the point P_m and the line determined by the equation $Ax + By + C = 0$ is greater than or equal to the perpendicular distance between the point P_j and the line determined by the equation $Ax + By + C = 0$ if

$$Ax_m + By_m + C \geq Ax_j + By_j + C.$$

PROOF: In 2-space the perpendicular distance d_k between a point P_k and the line determined by the equation $Ax + By + C = 0$ can be determined by the formula

$$d_k = \frac{\left| Ax_k + By_k + C \right|}{\sqrt{A^2 + B^2}}. \qquad (11,p.43)$$

If we multiply both sides of the above equation by $\sqrt{A^2 + B^2}$ we have

$$d_k\sqrt{A^2 + B^2} = \left| Ax_k + By_k + C \right|.$$

Now if

$$Ax_m + By_m + C \geq Ax_j + By_j + C \geq 0$$

it follows that

$$d_m\sqrt{A^2 + B^2} \geq d_j\sqrt{A^2 + B^2}$$

and since $\sqrt{A^2 + B^2} > 0$ we have

$$d_m \geq d_j.$$

This completes the proof.

THEOREM 4.7 Given a set of N distinct coplanar points, let H_j, $1 \leq j \leq f$, be a convex hull obtained by the procedure described in Chapter 3 where $A_i x + B_i y + C_i = 0$, $i = 1,..,NE$, are equations of the supporting lines of H_j and $P_1,..,P_{NV}$ are the vertices. If there exists a point P_h, $NV+1 \leq h \leq N$, and a number ℓ where $1 \leq \ell \leq NE$, such that

$$A_\ell x_h + B_\ell y_h + C_\ell > 0,$$

then

$$A_m x_h + B_m y_h + C_m \leq 0, \ m = 1,..,NE, \ m \neq \ell.$$

PROOF: The proof will be by mathematical induction on j. We consider the case for j = 1 first. In determining H_1 we started with two points in $\mathcal{P}$. For convenience suppose these points were $P_{\ell x}$ and P_{sx}. Then we know for any point P_i, $1 \leq i \leq N$, $x_i \leq x_{\ell x}$ and $x_i \geq x_{sx}$. We called the first two points P_1 and P_2. For our third point P_3 we chose a point P_k such that the perpendicular distance between the line containing P_1 and P_2 and the point P_k is greater than or equal to the perpendicular distance between this line and any other given point. We considered the projection of the given N points onto the x-y plane. For convenience let $A_1x + B_1y + C_1 = 0$ be an equation of the line containing P_1 and P_2; let $A_2x + B_2y + C_2 = 0$ be an equation of the line containing P_1 and P_3 and let $A_3x + B_3y + C_3 = 0$ be an equation of the plane containing P_2 and P_3.

By a proper choice of a coordinate system we can have the point P_3 at the origin and the line containing the points P_1 and P_2 parallel to the y-axis. (see Fig. 3)

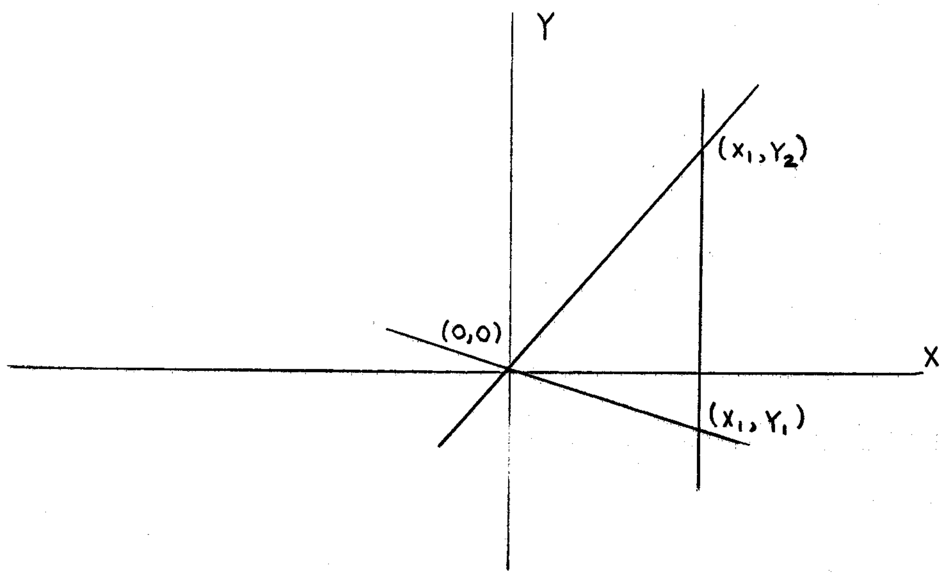


Figure 3

Using this coordinate system the equation $A_1x + B_1y + C_1 = 0$ becomes $x - x_1 = 0$, the equation $A_2x + B_2y + C_2 = 0$ becomes $y_1x - x_1y = 0$ and the equation $A_3x + B_3y + C_3 = 0$ becomes $-y_2x + x_1y = 0$. Thus the equations of the supporting lines of this triangle are

(1) $x - x_1 = 0$

(2) $y_1x - x_1y = 0$

(3) $-y_2x + x_1y = 0$

By the manner in which we chose P_3 and the choice of our present coordinate system we know that none of the given N points will have an x coordinate less than zero. Hence if we can show, using (2) and (3) above, that for

(4) $-y_2x_h + x_1y_h > 0,$

and (5) $y_1x_h - x_1y_h > 0,$

we must have $x_h < 0$, we will have proved that a point P_h cannot satisfy both of the conditions

$$A_ix_h + B_iy_h + C_i > 0, \; i = 2,3.$$

Adding equations (4) and (5) we obtain

$$(y_1 - y_2)x_h > 0.$$

Now since $y_1 - y_2 < 0$ it follows that $x_h < 0$.

By choosing a coordinate system so that the point P_1 is at the origin and the y-axis is coincident with the line $x - x_1 = 0$ we can show in a manner similar to the above that P_h cannot satisfy both of the conditions

$$A_i x_h + B_i y_h + C_i > 0, \quad i = 1,2.$$

Similarly by choosing a coordinate system so that the point P_2 is at the origin and the y-axis is coincident with the line $x - x_2 = 0$ we can show in a manner similar to the above that P_h cannot satisfy both of the conditions

$$A_i x_h + B_i y_h + C_i > 0, \quad i = 1,3.$$

Thus the statement is true for the case $j = 1$.

We now assume the statement to be true for $j = k$ and consider next the case for $j = k + 1$. To obtain H_{k+1} from H_k we essentially add one triangle to H_k. (Fig. 4)

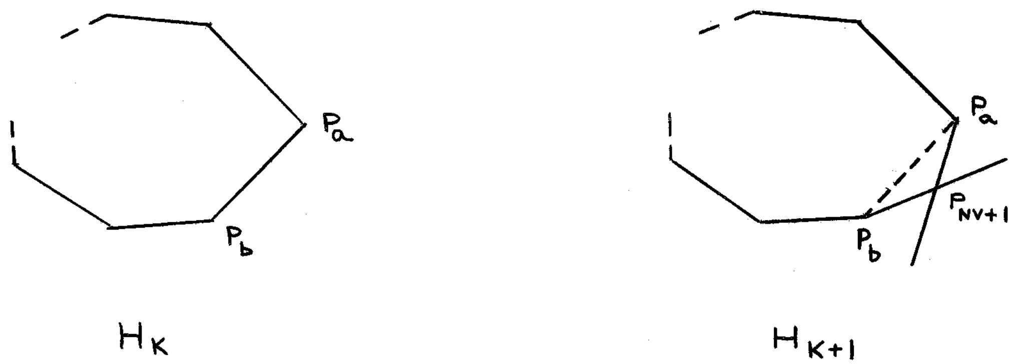


Figure 4

Considering the triangle determined by the points P_a, P_b and P_{NV+1} we can consider the point P_{NV+1} in the same way we did P_3 and thus show that the point P_h cannot satisfy the conditions

$$A_j x_h + B_j y_h + C_j > 0, \ j = NE+1, NE+2$$

where NE+1 contains the points P_a and P_{NV+1} and NE+2 contains the points P_b and P_{NV+1}. By the induction hypothesis the point P_h satisfies the condition

$$A_i x_h + B_i y_h + C_i \leq 0,$$

for NE-1 of the equations for H_k. If Q is an equation of H_k such that

$$A_q x_h + B_q y_h + C_q > 0,$$

then this equation is eliminated for H_{k+1}, and thus it follows that the original statement is also true for H_{k+1}. The statement is true therefore for all values of j.

THEOREM 4.8 Theorem 4.7 has no analogue in 3-dimensions.

PROOF: The proof is by example. Consider the following set of nine points

(-4,-4,2)	(0,0,0)	(9/4,4,1/8)

(-5/2,4,-1/3)	(1/2,4,-1/2)	(3,4,1)
(-2,4,-1)	(2,4,0)	(-2/135,1/9,-1/10)

The points (-4,4,2), (3,4,1), (-2,4,-1) and (0,0,0) are the vertices of H_1. We then add the points (2,4,0), (1/2,4,-1/2), (9/4,4,1/8) and (-5/2,4,-1/3) obtaining in turn H_2, H_3, H_4 and H_5 respectively.

Now the only point of the original nine which can still be outside H_5 is the point (-2/135,1/9,-1/10). Equations of the supporting planes of H_5 are given by the following table:

	EQUATION OF PLANE	VERTICES IN PLANE
1.	$2x - 5y + 14z = 0$	(-4,4,2), (3,4,1), (0,0,0)
2.	$y - 4 = 0$	(-4,4,2), (-2,4,-1), (3,4,-1)
3.	$4x - 3y - 20z = 0$	(-2,4,-1), (1/2,4,-1/2), (0,0,0)
4.	$2x - y - 6z = 0$	(1/2,4,-1/2), (2,4,0), (0,0,0)
5.	$2x - y - 4z = 0$	(9/4,4,1/8), (2,4,0), (0,0,0)
6.	$28x - 15y - 24z = 0$	(9/4,4,1/8), (3,4,1), (0,0,0)
7.	$-28x - 19y - 18z = 0$	(-5/2,4,-1/3), (-4,4,2), (0,0,0)
8.	$-16x - 11y - 12z = 0$	(-5/2,4,-1/3), (-2,4,-1), (0,0,0)

By substituting the coordinates of the point (-2/135,1/9,-1/10) into the left member of each of the equations (3) through (8) we find that

$$A_i(-2/135) + B_i(1/9) + C_i(-1/10) + D_i > 0, \text{ if } 3 \leq i \leq 8.$$

Chapter 5

FORTRAN PROGRAMS AND GENERAL FLOW CHARTS

Machine requirements:

IBM 1622 card input-output

IBM 1620 with at least 40,000 storage positions

indirect addressing

automatic divide

IBM 026 card punches are required for preparing data and an IBM 407 accounting machine is required to list punched output.

Restrictions:

With 40,000 storage positions the maximum number of points is 51.

Eight significant digits are retained during calculations. If more than eight are required the programs type out the message "DATA NOT ACCEPTABLE."

Data Preparation:

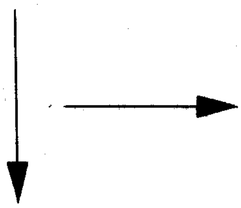
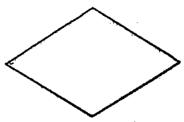
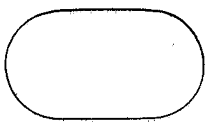
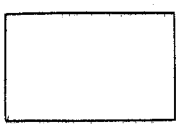
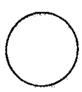
Input data for Program 1

Card No.	Field	Definition	Format	Example
1	1-2	N	I2	08

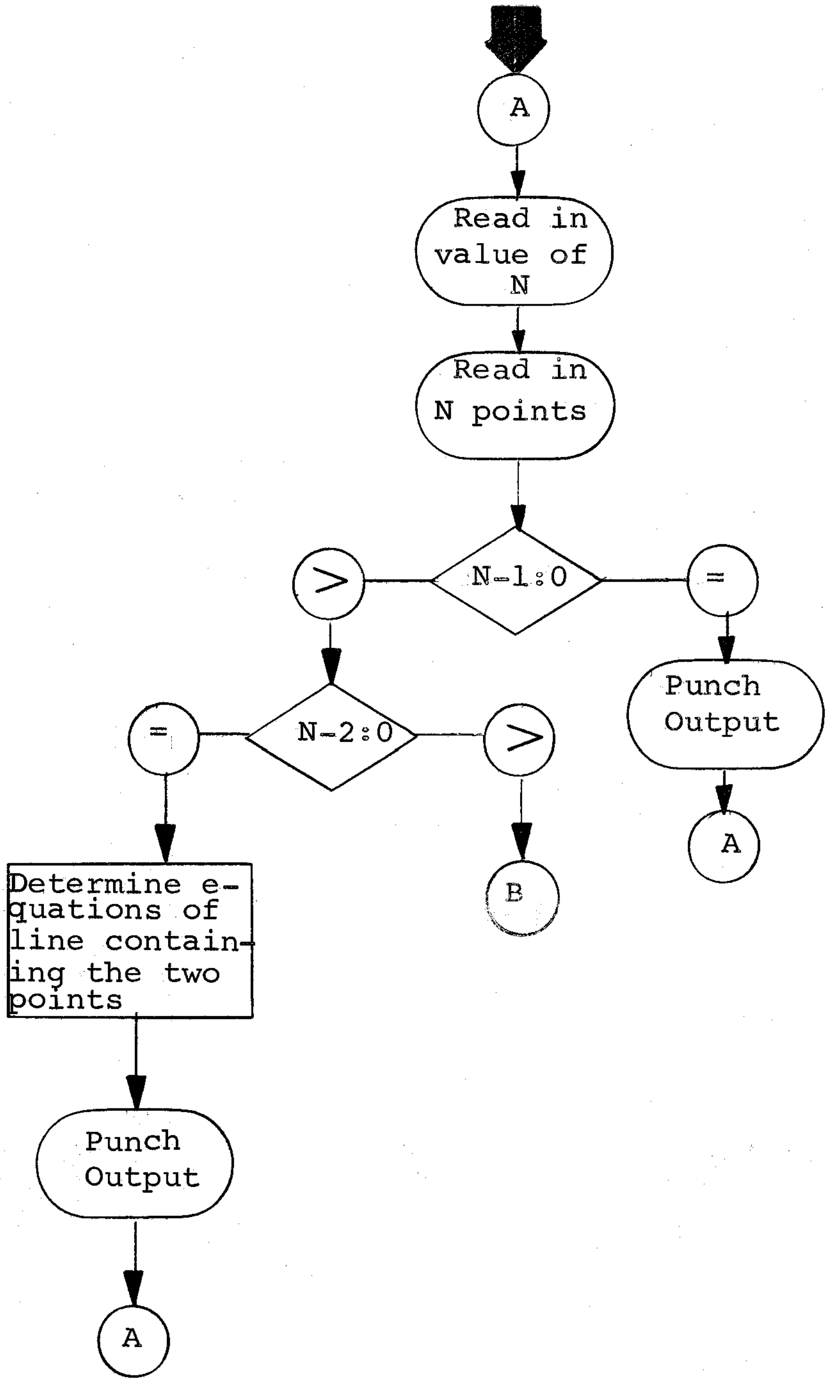
2	1-14	x_1	E14.8	+.12000000E+02
2	15-28	y_1	E14.8	+.30000000E+01
2	29-52	z_1	E14.8	+.14600000E-01
2	53-80	blank		
.				
.				
.				
N+1	1-14	x_N	E14.8	+.00000000E+02
N+1	15-28	y_N	E14.8	+.26100000E-01
N+1	29-52	z_N	E14.8	+.60000000E+01
N+1	53-80	blank		

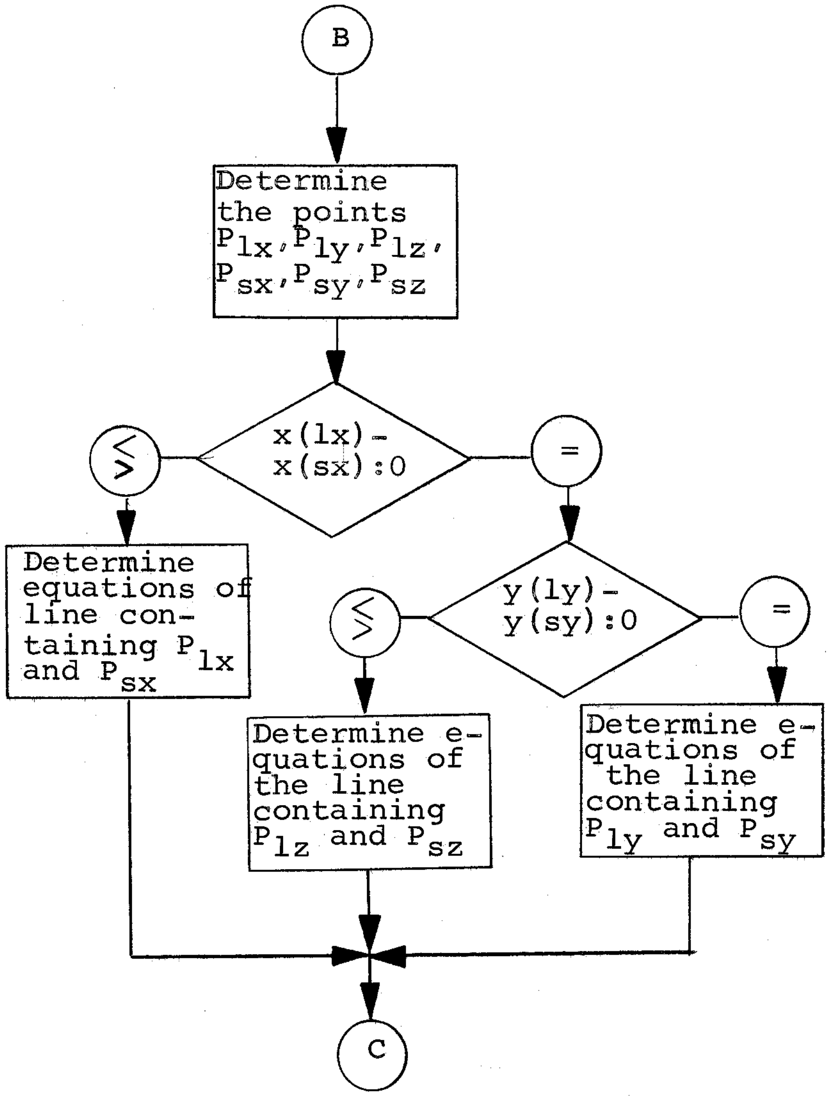
The output from Program 1 will be the final results if the message "CONVEX HULL" is typed on the typewriter. If the message "POINTS ARE COPLANAR" is typed on the typewriter the output from Program 1 is used as input for Program 2. If the message "POINTS ARE NOT COPLANAR" is typed on the console the output from Program 1 is used as input for Program 3.

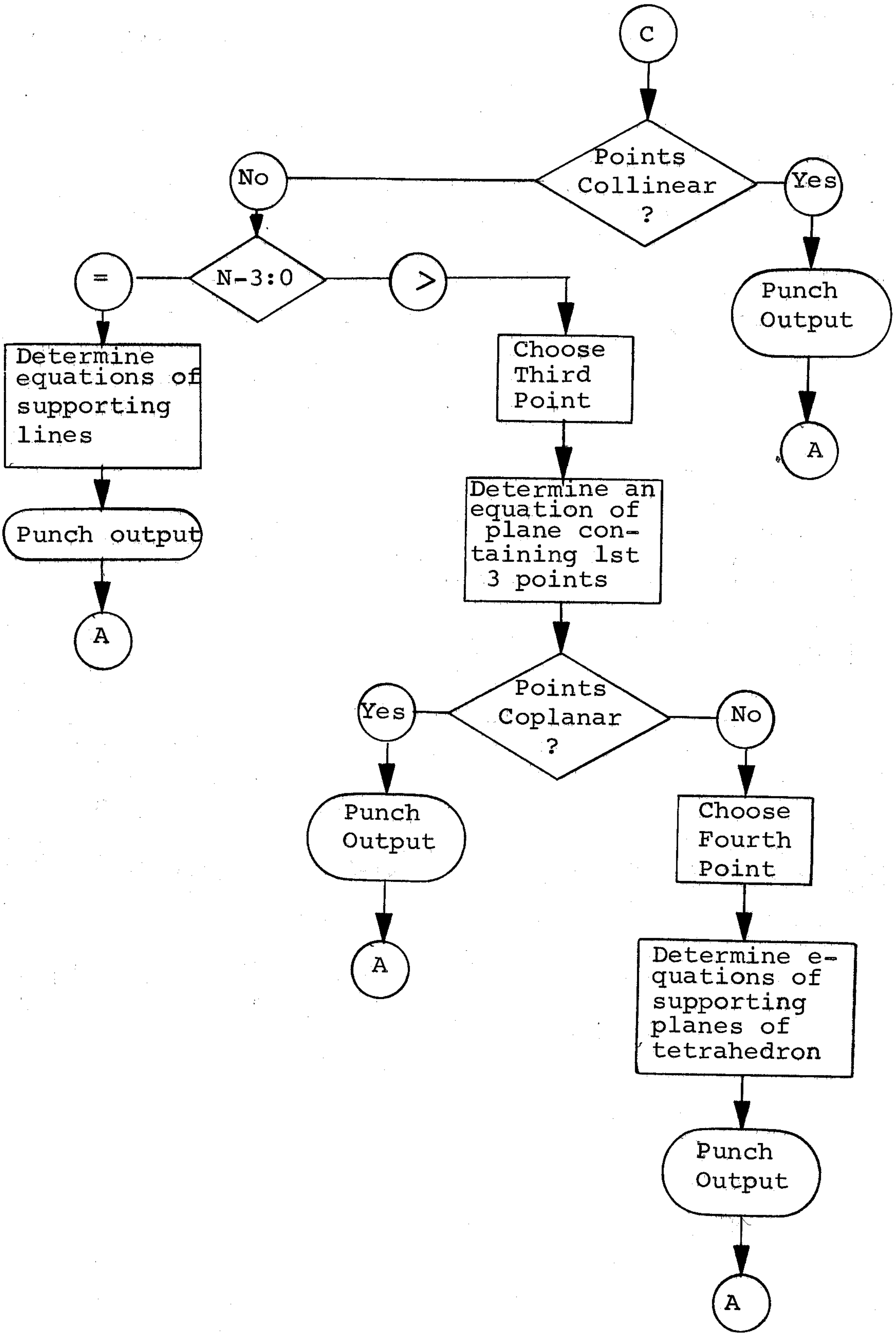
FLOW CHART SYMBOLS

SYMBOL	DESCRIPTION
	Direction of flow
	Decision function
	Input/Output function
	Processing function
	Connector or step identification

PROGRAM 1


A
Read in value of N
Read in N points
N-1:0
>
=
Punch Output
A
N-2:0
=
>
B
Determine equations of line containing the two points
Punch Output
A


B
Determine the points $P_{1x}, P_{1y}, P_{1z}, P_{sx}, P_{sy}, P_{sz}$
x(lx) - x(sx) : 0
$\lessgtr$
=
Determine equations of line containing P_{1x} and P_{sx}
y(ly) - y(sy) : 0
$\lessgtr$
=
Determine equations of the line containing P_{1z} and P_{sz}
Determine equations of the line containing P_{1y} and P_{sy}
C


C
Points Collinear ?
No
Yes
Punch Output
A
=
N-3:0
>
Determine equations of supporting lines
Punch output
A
Choose Third Point
Determine an equation of plane containing 1st 3 points
Points Coplanar ?
Yes
No
Punch Output
A
Choose Fourth Point
Determine equations of supporting planes of tetrahedron
Punch Output
A

```
C       CONVEX HULL-PROGRAM 1
        DIMENSION X(51),Y(51),Z(51),U(51),V(51),W(51),E(51)
        DIMENSION XA(4),XB(4),XC(4),XD(4)
    100 READ 1,N
        DO 101 I=1,N
    101 READ 21,X(I),Y(I),Z(I)
        NUMB1=1
        NUMB2=2
        ICOEF=1
        INN=0
        IHEAD=0
        IF(N-2)102,107,134
    102 PUNCH 3,N
        PRINT 4
        PUNCH 5
        PUNCH 6,X(N),Y(N),Z(N)
    103 PUNCH 7
        GO TO 100
    104 PRINT 8
        GO TO 100
    105 PUNCH 3,N
        I1=1
        I2=2
        PRINT 4
        PUNCH 9
        DO 106 I=1,N
    106 PUNCH 10,I,X(I),Y(I),Z(I)
    107 I1=1
        I2=2
    108 IF(X(I1)-X(I2))109,110,109
```

```
109 KPERP=1
    GO TO 113
110 IF(Y(I1)-Y(I2))111,112,111
111 KPERP=2
    GO TO 113
112 KPERP=3
    GO TO 114
113 A1=Y(I2)-Y(I1)
    B1=X(I1)-X(I2)
    C1=(X(I2)=X(I1))*Y(I1)+(Y(I1)-Y(I2))*X(I1)
    GO TO (114,115,100),KPERP
114 A2=Z(I2)-Z(I1)
    B2=X(I1)-X(I2)
    C2=(X(I2)-X(I1))*Z(I1)+(Z(I1)-Z(I2))*X(I1)
    GO TO (116,100,115),KPERP
115 A3=Z(I2)-Z(I1)
    B3=Y(I1)-Y(I2)
    C3=(Y(I2)-Y(I1))*Z(I1)+(Z(I1)-Z(I2))*Y(I1)
116 ZERO=0.
    IF(N-2)104,219,117
117 IF(INN-1)118,100,181
118 IF(IHEAD-2)119,120,120
119 PUNCH 11
    PUNCH 12
    PUNCH 13
    PUNCH 14
    IHEAD=2
120 GO TO (121,122,123),KPERP
121 PUNCH 15,NUMB1,A1,B1,ZERO,C1
    PUNCH 15,NUMB2,A2,ZERO,B2,C2
```

```
      GO TO 124
  122 PUNCH 15,NUMB1,A1,B1,ZERO,C1
      PUNCH 15,NUMB2,ZERO,A3,B3,C3
      GO TO 124
  123 PUNCH 15,NUMB1,A2,ZERO,B2,C2
      PUNCH 15,NUMB2,ZERO,A3,B3,C3
  124 GO TO (126,132,133,103),ICOEF
  126 PUNCH 16
      DO 127 I=1,2
  127 PUNCH 17,I
      IF(INN-2)128,103,128
  128 DO 129 I=3,4
  129 PUNCH 18,I
      DO 130 I=5,6
  130 PUNCH 19,I
      GO TO 103
  131 ICOEF=2
      NUMB1=1
      NUMB2=2
      GO TO 105
  132 I1=3
      ICOEF=3
      NUMB1=NUMB1+2
      NUMB2=NUMB2+2
      GO TO 108
  133 I2=1
      ICOEF=1
      NUMB1=NUMB1+2
      NUMB2=NUMB2+2
```

```
      GO TO 108
  134 A=X(1)
      B=Y(1)
      C=Z(1)
      D=X(1)
      EE=Y(1)
      F=Z(1)
      DO 146 I=1,N
      IF(A-X(I))135,135,136
  135 KL=I
      A=X(I)
  136 IF(B-Y(I))137,137,138
  137 LL=I
      B=Y(I)
  138 IF(C-Z(I))139,139,140
  139 ML=I
      C=Z(I)
  140 IF(X(I)-D)141,141,142
  141 KS=I
      D=X(I)
  142 IF(Y(I)-EE)143,143,144
  143 LS=I
      EE=Y(I)
  144 IF(Z(I)-F)145,145,146
  145 MS=I
      F=Z(I)
  146 CONTINUE
      IF(X(KL)-X(KS))147,151,147
  151 IF(Y(LL)-Y(LS))160,152,160
  147 I1=KL
      I2=KS
```

```
      KPERP=1
      GO TO 155
  160 I1=LL
      I2=LS
      KPERP=2
      GO TO 155
  152 I1=ML
      I2=MS
      KPERP=3
  155 K=I1
      J=1
  150 RA=X(K)
      RB=Y(K)
      RC=Z(K)
      X(K)=X(J)
      Y(K)=Y(J)
      Z(K)=Z(J)
      X(J)=RA
      Y(J)=RB
      Z(J)=RC
      IF(J-1)153,153,154
  153 J=2
      IF(I2-1)157,157,158
  157 K=I1
      GO TO 150
  158 K=I2
      GO TO 150
  154 INN=2
      LINE=0
      GO TO (113,113,114),KPERP
```

```
181 GO TO (182,182,189),KPERP
182 DO 183 I=1,N
183 E(I)=A1*X(I)+B1*Y(I)+C1
184 E1=E(1)
    DO 186 I=1,N
    IF(E1**2-E(I)**2)185,185,186
185 E1=E(I)
    K=I
186 CONTINUE
    IVERT=2
    L=3
    IF(E1)225,187,225
187 IF(LINE-1)188,219,104
188 LINE=1
    GO TO (189,191,191),KPERP
189 DO 190 I=1,N
190 E(I)=A2*X(I)+B2*Z(I)+C2
    GO TO 184
191 DO 192 I=1,N
192 E(I)=A3*Y(I)+B3*Z(I)+C3
    GO TO 184
225 IF(N-3)104,131,125
125 DO 226 I = 1,N
    01=((Y(2)-Y(I))*(Z(2)-Z(1))-(Z(2)-Z(I))*(Y(2)-Y(1)))**2
    02=((Z(2)-Z(I))*(X(2)-X(1))-(X(2)-X(I))*(Z(2)-Z(1)))**2
    03=((X(2)-X(I))*(Y(2)-Y(1))-(Y(2)-Y(I))*(X(2)-X(1)))**2
226 E(I)-01+02+03
    E1=E(1)
    DO 228 I=1,N
    IF(E1-E(I))227,227,228
227 E1=E(I)
    K=I
```

```
228 CONTINUE
193 RA=X(K)
    RB=Y(K)
    RC=Z(K)
    X(K)=X(L)
    Y(K)=Y(L)
    Z(K)=Z(L)
    X(L)=RA
    Y(L)=RB
    Z(L)=RC
    GO TO (199,194),IVERT
194 A=Y(1)*Z(2)+Y(2)*Z(3)+Z(1)*Y(3)-Y(3)*Z(2)-Y(1)*Z(3)-Y(2)*Z(1)
    B=X(3)*Z(2)+X(1)*Z(3)+X(2)*Z(1)-X(1)*Z(2)-X(2)*Z(3)-X(3)*Z(1)
    C=X(1)*Y(2)+X(2)*Y(3)+Y(1)*X(3)-Y(2)*X(3)-X(1)*Y(3)-X(2)*Y(1)
    DD=X(3)*Y(2)*Z(1)+Y(3)*Z(2)*X(1)+X(2)*Y(1)*Z(3)
    D=DD-X(1)*Y(2)*Z(3)-X(2)*Y(3)*Z(1)-Y(1)*Z(2)*X(3)
    CP=0.
    DO 195 I=1,N
    E(I)=A*X(I)+B*Y(I)+C*Z(I)+D
195 CP=CP+E(I)*E(I)
    IF(CP)104,209,196
196 F=E(1)
    DO 198 I=1,N
    IF(F*F-E(I)*E(I))197,197,198
197 F=E(I)
    K=I
198 CONTINUE
    L=4
    IVERT=1
    GO TO 193
199 PRINT 20
```

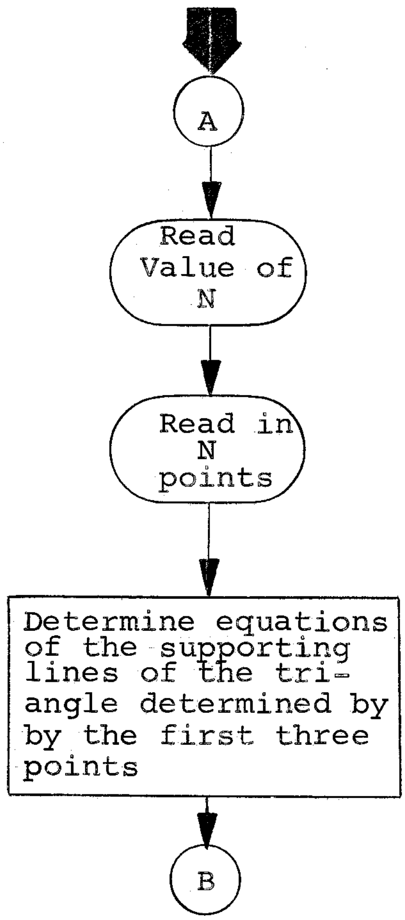
```
      C1=(X(1)+X(2)+X(3)+X(4))/4.
      C2=(Y(1)+Y(2)+Y(3)+Y(4))/4.
      C3=(Z(1)+Z(2)+Z(3)+Z(4))/4.
      II=1
      I=1
      L=1
      J=2
      K=3
      U(1)=1
      V(1)=2
      W(1)=3
  200 XA(L)=Y(I)*Z(J)+Y(J)*Z(K)+Z(I)*Y(K)-Y(K)*Z(J)-Y(I)*Z(K)-Y(J)*Z(I)
      XB(L)=X(K)*Z(J)+X(I)*Z(K)+X(J)*Z(I)-X(I)*Z(J)-X(J)*Z(K)-X(K)*Z(I)
      XC(L)=X(I)*Y(J)+X(J)*Y(K)+Y(I)*X(K)-Y(J)*X(K)-X(I)*Y(K)-X(J)*Y(I)
      DD=X(K)*Y(J)*Z(I)+Y(K)*Z(J)*X(I)+X(J)*Y(I)*Z(K)
      XD(L)=DD-X(I)*Y(J)*Z(K)-X(J)*Y(K)*Z(I)-Y(I)*Z(J)*X(K)
      GO TO (201,202,203,204),II
  201 K=4
      L=2
      II=2
      U(2)=1
      V(2)=2
      W(2)=4
      GO TO 200
  202 J=3
      L=3
      II=3
      U(3)=1
      V(3)=3
      W(3)=4
      GO TO 200
```

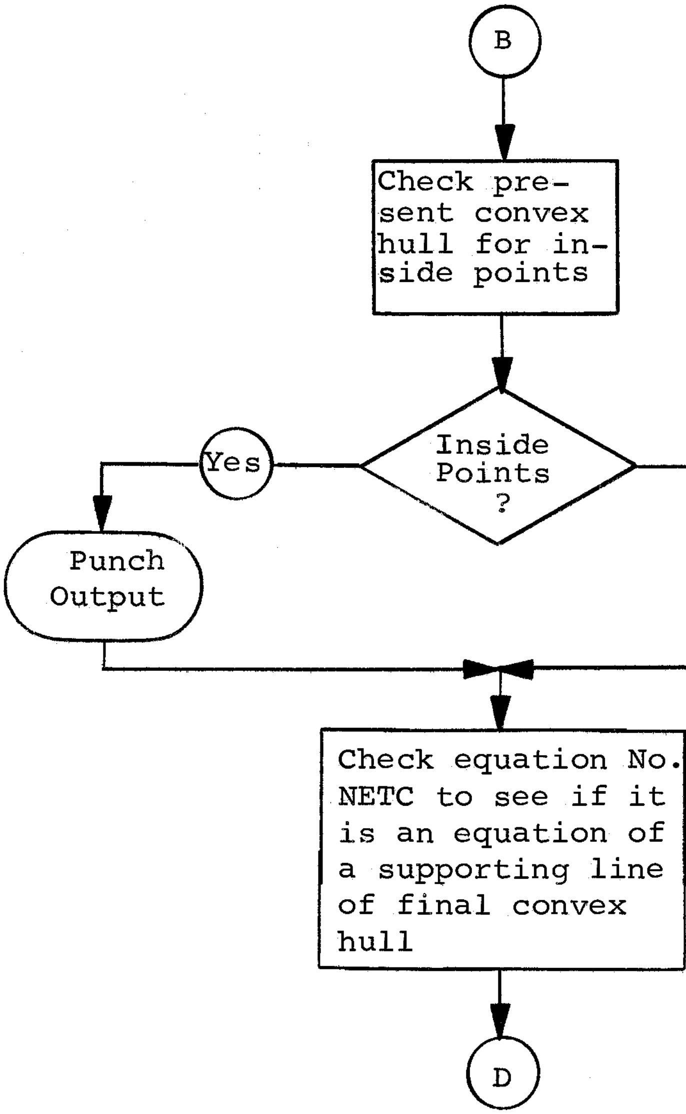
```
203 I=2
    L=4
    II=4
    U(4)=2
    V(4)=3
    W(4)=4
    GO TO 200
204 DO 206 I=1,4
    E(I)=XA(I)*C1+XB(I)*C2+XC(I)*C3+XD(I)
    IF(E(I))206,104,205
205 XA(I)=XA(I)*(-1.)
    XB(I)=XB(I)*(-1.)
    XC(I)=XC(I)*(-1.)
    XD(I)=XD(I)*(-1.)
206 CONTINUE
    PUNCH 21,C1,C2,C3
    DO 207 I=1,4
    IU=U(I)
    IV=V(I)
    IW=W(I)
207 PUNCH 21,XA(I),XB(I),XC(I),XD(I),IU,IV,IW
    PUNCH 25,N
    DO 208 I=1,N
208 PUNCH 21,X(I),Y(I),Z(I)
    GO TO 100
209 PRINT 23
    PUNCH 21,A,B,C,D
    IF(A)211,210,211
210 IF(B)213,215,213
211 DO 212 I=1,N
```

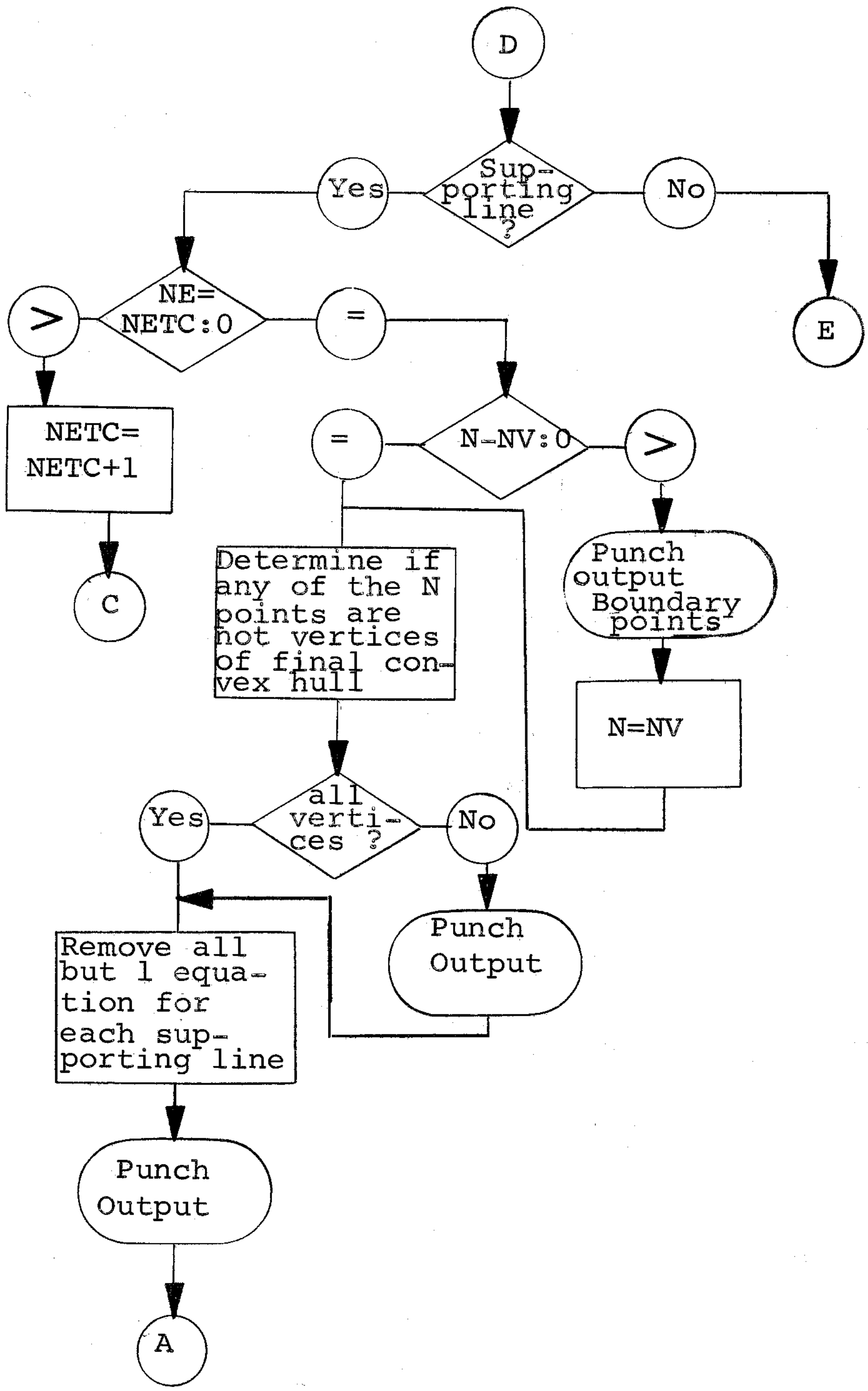
```
      U(I)=Y(I)
      V(I)=Z(I)
  212 W(I)=X(I)
      KPROJ=1
      GO TO 217
  213 DO 214 I=1,N
      U(I)=X(I)
      V(I)=Z(I)
  214 W(I)=Y(I)
      KPROJ=2
      GO TO 217
  215 DO 216 I=1,N
      U(I)=X(I)
      V(I)=Y(I)
  216 W(I)=Z(I)
      KPROJ=3
  217 PUNCH 25,N,KPROJ
      DO 218 I=1,N
  218 PUNCH 21,U(I),V(I),W(I)
      GO TO 100
  219 PUNCH 3,N
      PUNCH 26
      PRINT 4
      PUNCH 27
      DO 220 I=1,2
  220 PUNCH 10,I,X(I),Y(I),Z(I)
      ZERO=0.
      ICOEF=4
      NUMB1=1
      NUMB2=2
      PUNCH 28
```

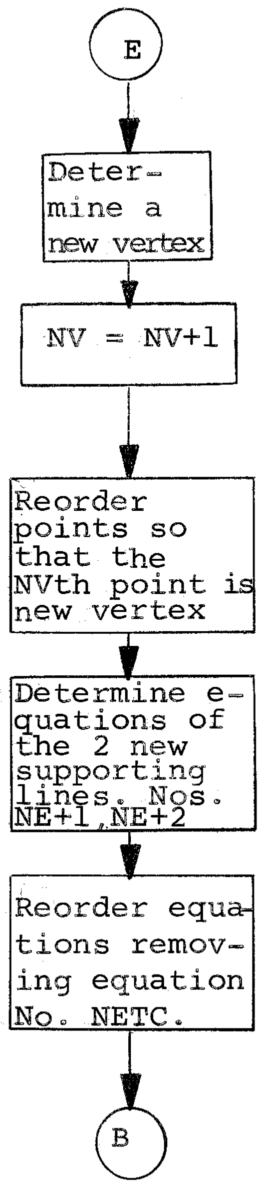
```
      PUNCH 29
      PUNCH 13
      PUNCH 14
      GO TO 120
    1 FORMAT (I2)
    3 FORMAT (/11HCONVEX HULL/2HN=,I4/)
    4 FORMAT (11HCONVEX HULL)
    5 FORMAT (13X,1HX,15X,1HY,15X,1HZ/)
    6 FORMAT (7X,E14.8,2X,E14.8,2X,E14.8)
    7 FORMAT (/31HCONVEX HULL HAS BEEN DETERMINED///)
    8 FORMAT (16HERROR IN PROGRAM)
    9 FORMAT (/8HVERTICES//2X,3HNO.,8X,1HX,15X,1HY,15X,1HZ/)
   10 FORMAT (I3,4X,E14.8,2X,E14.8,2X,E14.8)
   11 FORMAT (//29HEQUATIONS OF SUPPORTING LINES,16H ARE IN THE FORM)
   12 FORMAT (20HAX + BY + CZ + D = 0//)
   13 FORMAT (8HEQUATION,5X,1HA,15X,1HB,15X,1HC,15X,1HD)
   14 FORMAT (2X,3HNO./)
   15 FORMAT (I3,2X,E14.8,2X,E14.8,2X,E14.8,2X,E14.8)
   16 FORMAT (//8HVERTICES,13X,16HSATISFY EQUATION/)
   17 FORMAT (2X,2H1,,2X,1H2,18X,I3)
   18 FORMAT (2X,2H2,,2X,1H3,18X,I3)
   19 FORMAT (2X,2H1,,2X,1H3,18X,I3)
   20 FORMAT (23HPOINTS ARE NOT COPLANAR)
   21 FORMAT (4E14.8,3I4)
   23 FORMAT (19HPOINTS ARE COPLANAR)
   25 FORMAT (I4,I2)
   26 FORMAT (/20HPOINTS ARE COLLINEAR/)
   27 FORMAT (/9HENDPOINTS//2X,3HNO.,8X,1HX,15X,1HY,15X,1HZ/)
 28FORMAT(//21HEQUATIONS OF THE LINE,30H CONTAINING THE GIVEN N POINTS)
   29 FORMAT (15HARE IN THE FORM,21H AX + BY + CZ + D = 0//)
      END
```

PROGRAM 2 - COPLANAR




B
Check present convex hull for inside points
Inside Points ?
Yes
No
Punch Output
C
Check equation No. NETC to see if it is an equation of a supporting line of final convex hull
D


D
Sup-
porting
line
?
Yes
No
E
NE=
NETC:0
>
=
NETC=
NETC+1
C
N-NV:0
=
>
Punch
output
Boundary
points
N=NV
Determine if
any of the N
points are
not vertices
of final con-
vex hull
all
verti-
ces ?
Yes
No
Punch
Output
Remove all
but 1 equa-
tion for
each sup-
porting line
Punch
Output
A


E
Deter-
mine a
new vertex
NV = NV+1
Reorder
points so
that the
NVth point is
new vertex
Determine e-
quations of
the 2 new
supporting
lines. Nos.
NE+1,NE+2
Reorder equa-
tions remov-
ing equation
No. NETC.
B

```
C     CONVEX HULL-PROGRAM 2-COPLANAR
      DIMENSION R(51),S(51),T(51),A(51),B(51),C(51),D(51),E(51)
      DIMENSION MM(51),NN(51)
  100 READ 1,AP,BP,CP,DP
      READ 2,N,KPROJ
      DO 101 I=1,N
  101 E(I)=0.
      PUNCH 3,N
      DO 102 I=1,N
  102 READ 4,R(I),S(I),T(I)
      NETC=1
      NE=3
      NV=3
      NEWEQ=1
      INSID=1
      IBDRY=1
      C1=(R(1)+R(2)+R(3))/3.
      C2=(S(1)+S(2)+S(3))/3.
      A(1)=S(2)-S(1)
      B(1)=R(1)-R(2)
      C(1)=(R(2)-R(1))*S(1)+(S(1)-S(2))*R(1)
      MM(1)=1
      NN(1)=2
      A(2)=S(3)-S(2)
      B(2)=R(2)-R(3)
      C(2)=(R(3)-R(2))*S(2)+(S(2)-S(3))*R(2)
      MM(2)=3
      NN(2)=2
      A(3)=S(1)-S(3)
      B(3)=R(3)-R(1)
```

```
      C(3)=(R(1)-R(3))*S(1)+(S(3)-S(1))*R(1)
      MM(3)=1
      NN(3)=3
  103 DO 105 J=NEWEQ,NE
      D(J)=A(J)*C1+B(J)*C2+C(J)
      IF(D(J))105,105,104
  104 A(J)=A(J)*(-1.)
      B(J)=B(J)*(-1.)
      C(J)=C(J)*(-1.)
  105 CONTINUE
      LESSQ=2
      IF(NE-3)112,106,134
C     CHECKING FOR INSIDE POINTS
  106 IF(N-NV)112,136,186
  186 I=NV+1
  107 DO 108 J=1,NE
  108 D(J)=A(J)*R(I)+B(J)*S(I)+C(J)
      DMAX=D(1)
      DO110 J=1,NE
      IF(DMAX-D(J))109,110,110
  109 DMAX=D(J)
  110 CONTINUE
      IF(DMAX)111,120,120
  111 GO TO (113,114),INSID
  112 PRINT 5
      GO TO 100
  113 PUNCH 6
      INSID=2
  114 GO TO (115,116,117),KPROJ
  115 PUNCH 7,T(I),R(I),S(I)
      GO TO 118
```

```
116 PUNCH 7,R(I),T(I),S(I)
    GO TO 118
117 PUNCH 7,R(I),S(I),T(I)
118 N=N-1
    DO 119 J=I,N
    R(J)=R(J+1)
    S(J)=S(J+1)
119 T(J)=T(J+1)
    IF(N-I)124,107,107
120 IF(N-I)112,124,122
122 I=I+1
    GO TO 107
C   CHECKING FOR SUPPORTING LINES
124 I=NETC
    MR=MM(I)
    NR=NN(I)
    D(I)=A(I)*R(MR)+B(I)*S(MR)+C(I)
    IF(D(I))126,125,126
125 D(I)=A(I)*R(NR)+B(I)*S(NR)+C(I)
    IF(D(I))126,128,126
126 PRINT 8
    IF(SENSE SWITCH 1)128,100
128 NCK=NV+1
    IF(N-NCK)144,187,187
187 DO 129 J=NCK,N
129 D(J)=A(I)*R(J)+B(I)*S(J)+C(I)
    DMAX=D(NCK)
    DO 131 J=NCK,N
    IF(DMAX-D(J))130,130,131
130 DMAX=D(J)
```

```
      K=J
  131 CONTINUE
      IF(DMAX)132,132,133
  132 NETC=NETC+1
      IF(NE-NETC)136,124,124
  133 NV=NV+1
      RA=R(NV)
      SA=S(NV)
      TA=T(NV)
      R(NV)=R(K)
      S(NV)=S(K)
      T(NV)=T(K)
      R(K)=RA
      S(K)=SA
      T(K)=TA
      I1=MM(I)
      I2=NN(I)
      NE=NE+1
      A(NE)=S(I1)-S(NV)
      B(NE)=R(NV)-R(I1)
      C(NE)=(R(I1)-R(NV))*S(NV)+(S(NV)-S(I1))*R(NV)
      MM(NE)=MM(I)
      NN(NE)=NV
      NE(NE+1
      A(NE)=S(I2)-S(NV)
      B(NE)=R(NV)-R(I2)
      C(NE)=(R(I2)-R(NV))*S(NV)+(S(NV)-S(I2))*R(NV)
      MM(NE)=NN(I)
      NN(NE)=NV
      NEWEQ=NE-1
      GO TO 103
```

```
  134 NE=NE-1
      DO 135 J=I,NE
      A(J)=A(J+1)
      B(J)=B(J+1)
      C(J)=C(J+1)
      MM(J)=MM(J+1)
  135 NN(J)=NN(J+1)
      GO TO (166,106),LESSQ
C     CHECKING FOR BOUNDARY POINTS THAT ARE NOT VERTICES
  136 NPNCH=0
      IF(NV-N)137,144,112
  137 PUNCH 9
      IBDRY=2
      K=NV+1
      DO 143 J=K,N
  138 GO TO (139,140,141),KPROJ
  139 PUNCH 7,T(J),R(J),S(J)
      GO TO 142
  140 PUNCH 7,R(J),T(J),S(J)
      GO TO 142
  141 PUNCH 7,R(J),S(J),T(J)
  142 IF(NPNCH)112,143,160
  143 CONTINUE
      N=NV
C     CHECKING FOR BOUNDARY POINTS THAT WERE VERTICES
C     BUT ARE NOT VERTICES OF THE FINAL CONVEX HULL
  144 K=1
  145 DO 146 I=1,N
  146 D(I)=A(K)*R(I)+B(K)*S(I)+C(K)
      I=1
```

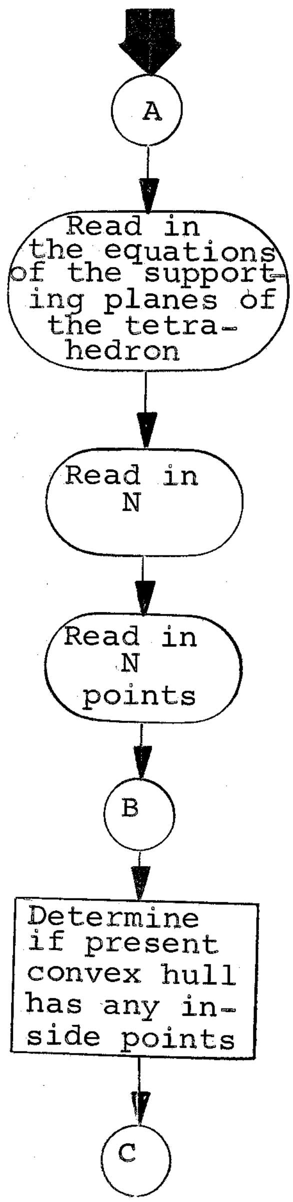
```
 147 IF(D(I))151,149,151
 149 IF(MM(K)-I)150,151,150
 150 IF(NN(K)-I)155,151,155
 151 IF(N-I)112,153,152
 152 I=I+1
     GO TO 147
 153 IF(NE-K)112,170,154
 154 K=K+1
     GO TO 145
 155 I1=MM(K)
     I2=NN(K)
     I3=I
     KK=K
     AL=SQRT((R(I1)-R(I2))**2+(S(I1)-S(I2))**2)
     BL=SQRT((R(I1)-R(I3))**2+(S(I1)-S(I3))**2)
     CL=SQRT((R(I2)-R(I3))**2+(S(I2)-S(I3))**2)
     IF(AL+BL-CL)157,156,157
 156 J=I1
     MM(K)=I3
     I1=MM(K)
     GO TO 158
 157 IF(AL+CL-BL)1157,2157,1157
1157 J=I3
     GO TO 158
2157 J=I2
     NN(K)=I3
     I2=NN(K)
 158 NPNCH=2
     I=0
     IF(IBDRY-2)159,138,112
 159 PUNCH 9
```

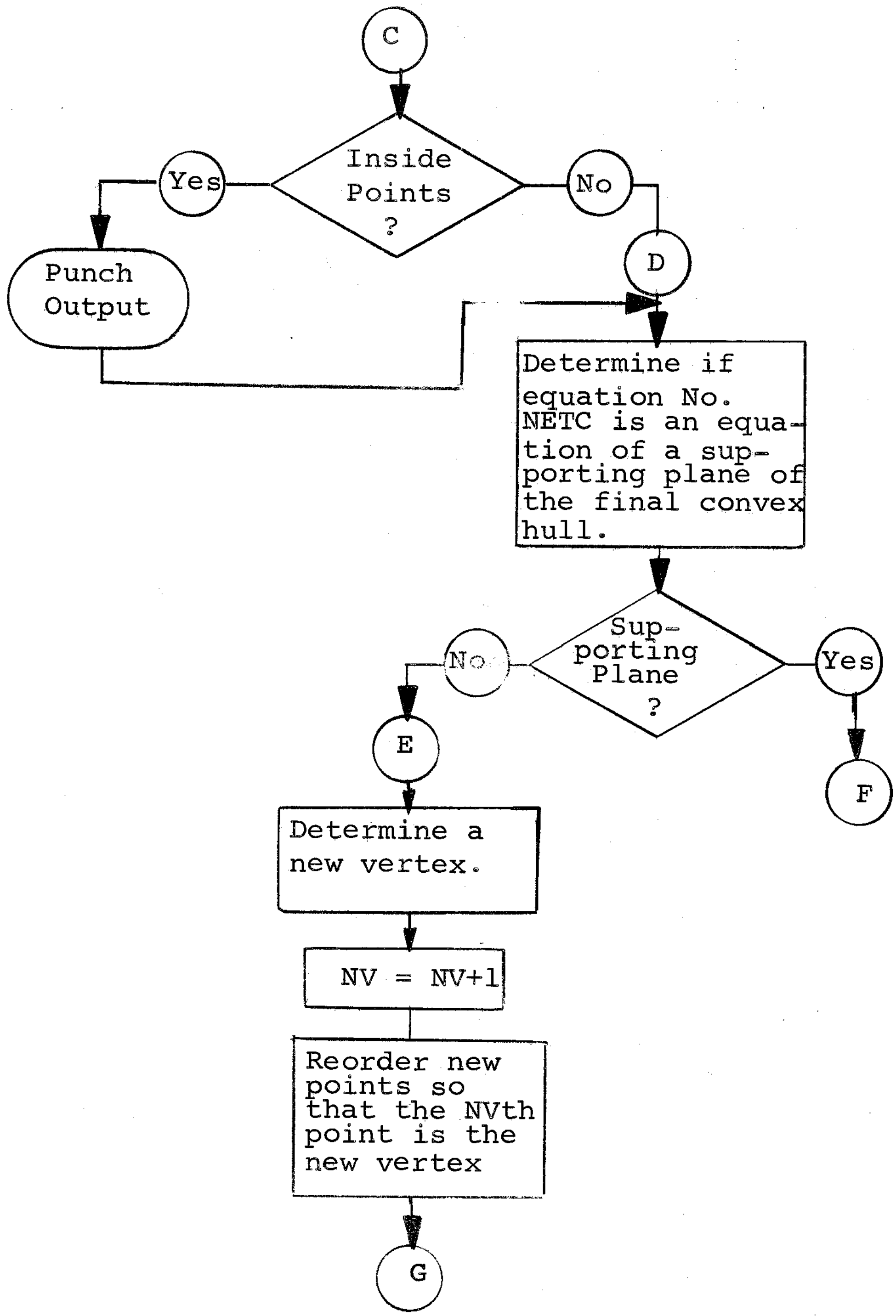
```
      GO TO 138
C     CHECKING FOR MORE THAN ONE EQUATION OF THE SAME LINE
  160 E(J)=1.
  161 I=I+1
  162 V1=A(I)*R(I1)+B(I)*S(I1)+C(I)
      IF(V1)169,163,169
  163 V2=A(I)*R(I2)+B(I)*S(I2)+C(I)
      IF(V2)169,164,169
  164 IF(I-KK)165,161,165
  165 LESSQ=1
      GO TO 134
  166 IF(NE-I)167,162,162
  167 IF(NE-KK)112,170,168
  168 K=KK+1
      GO TO 145
  169 IF(NE-I)112,167,161
  170 PUNCH 10
      NUMB=1
      DO 175 I=1,N
      IF(E(I))112,171,175
  171 GO TO (172,173,174),KPROJ
  172 PUNCH 11,NUMB,T(I),R(I),S(I)
      GO TO 190
  173 PUNCH 11,NUMB,R(I),T(I),S(I)
      GO TO 190
  174 PUNCH 11,NUMB,R(I),S(I),T(I)
  190 NUMB=NUMB+1
  175 CONTINUE
      PUNCH 12
      PUNCH 13
```

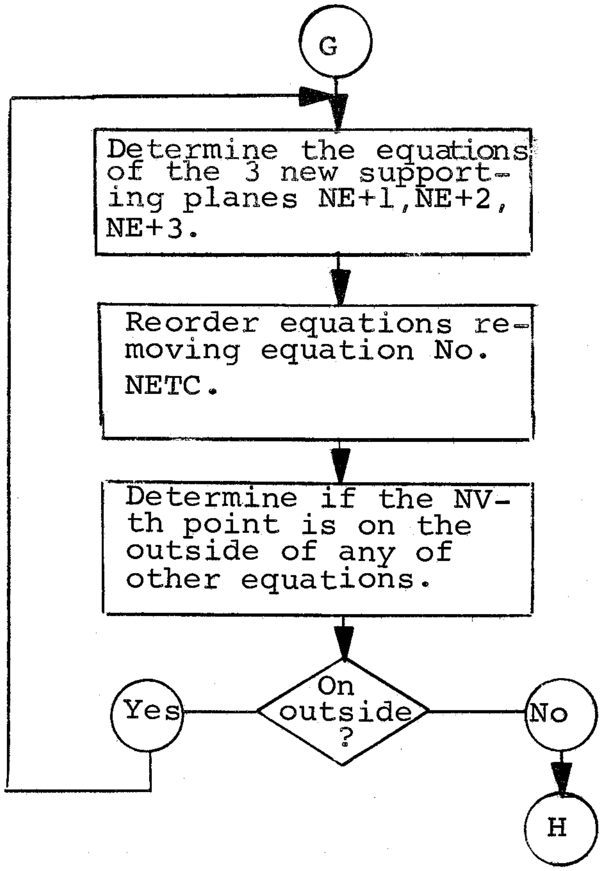
```
      PUNCH 14
      DO 179 I=1,NE
      ZERO=0.
      GO TO (176,177,178),KPROJ
  176 PUNCH 11,I,ZERO,A(I),B(I),C(I)
      GO TO 179
  177 PUNCH 11,I,A(I),ZERO,B(I),C(I)
      GO TO 179
  178 PUNCH 11,I,A(I),B(I),ZERO,C(I)
  179 CONTINUE
      PUNCH 15
      DO 191 I=1,N
      IF (E(I))112,184,180
  180 DO 184 K=1,NE
      IF(MM(K)-I)182,181,181
  181 MM(K)=MM(K)-1
  182 IF(NN(K)-I)184,183,183
  183 NN(K)=NN(K)-1
  184 CONTINUE
  191 CONTINUE
      DO 185 I=1,NE
  185 PUNCH 16,MM(I),NN(I),I
      PUNCH 17
      PUNCH 18
      PUNCH 19
      PUNCH 7,AP,BP,CP,DP
      PUNCH 20
      GO TO 100
    1 FORMAT (4E14.8)
    2 FORMAT (I4,I2)
    3 FORMAT (/11HCONVEX HULL/2HN=,I4/)
```

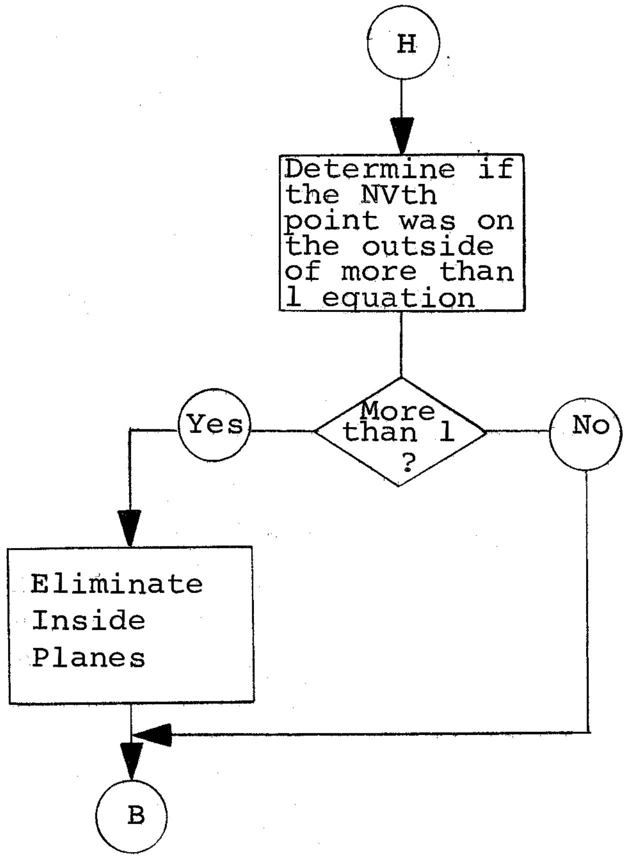
```
   4 FORMAT (3E14.8)
   5 FORMAT (16HERROR IN PROGRAM)
   6 FORMAT (/13HINSIDE POINTS//13X,1HX,15X,1HY,15X,1HZ/)
   7 FORMAT (6X,E14.8,2X,E14.8,2X,E14.8,2X,E14.8)
   8 FORMAT (19HDATA NOT ACCEPTABLE)
   9 FORMAT (/15HBOUNDARY POINTS//13X,1HX,15X,1HY,15X,1HZ/)
  10 FORMAT (/8HVERTICES//2X,3HNO.,8X,1HX,15X,1HY,15X,1HZ/)
  11 FORMAT (I4,2X,E14.8,2X,E14.8,2X,E14.8,2X,E14.8)
  12 FORMAT (/29HEQUATIONS OF SUPPORTING LINES,16H ARE IN THE FORM)
  13 FORMAT (20HAX + BY + CZ + D = 0//)
  14 FORMAT (1X,3HNO.,9X,1HA,15X,1HB,15X,1HC,15X,1HD/)
  15 FORMAT (/1X,8HVERTICES,10X,16HSATISFY EQUATION/)
  16 FORMAT (I4,1H,,I4,14X,I4)
17FORMAT (//27HCOEFFICIENTS OF AN EQUATION,24H OF THE PLANE CONTAINING)
18FORMAT (21HTHE GIVEN N POINTS IN,30H THE FORM AX + BY + CZ + D = 0//)
  19 FORMAT (13X,1HA,15X,1HB,15X,1HC,15X,1HD/)
  20 FORMAT (/31HCONVEX HULL HAS BEEN DETERMINED)
     END
```

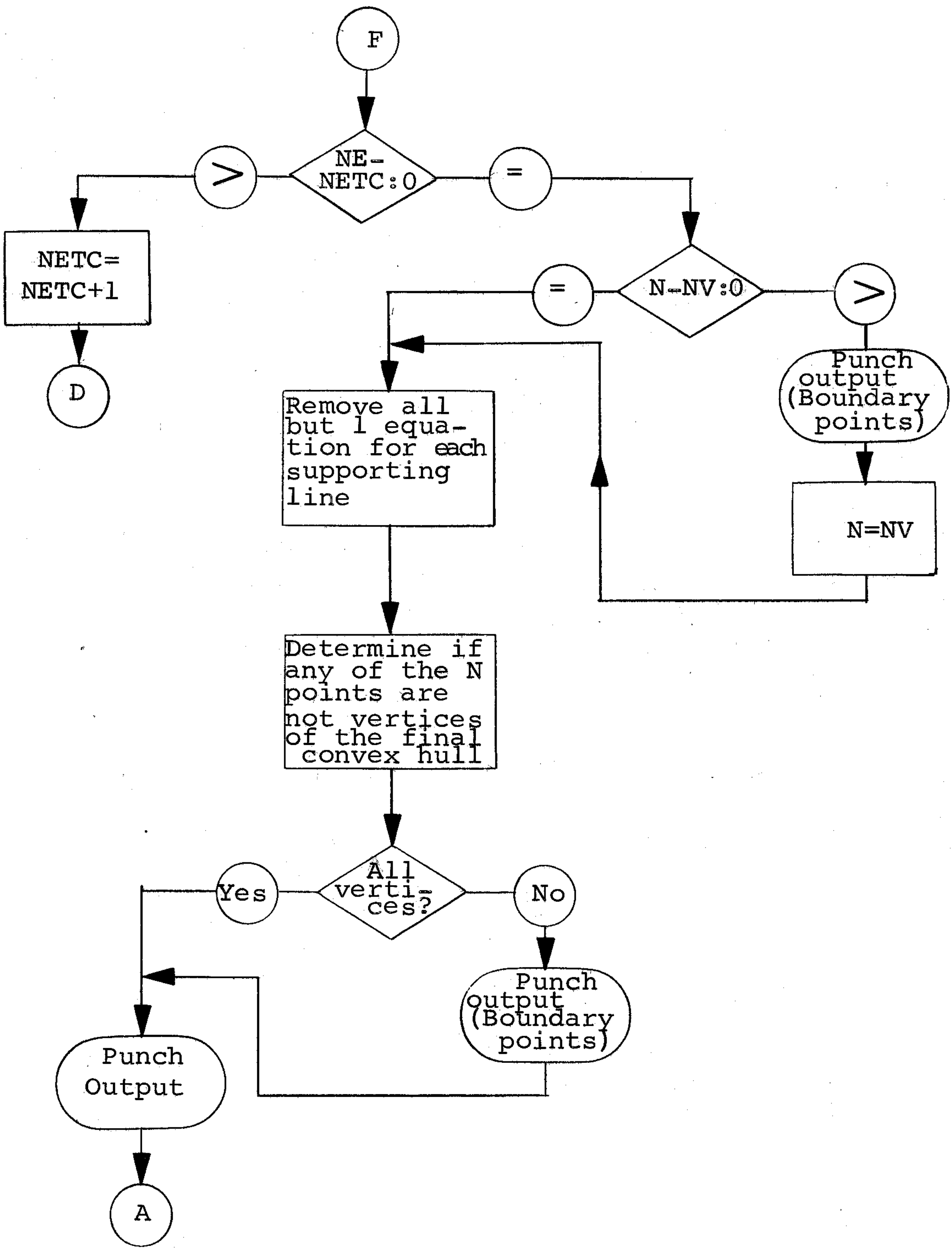
PROGRAM 3 - NOT COPLANAR




C
Inside Points ?
Yes
No
Punch Output
D
Determine if equation No. NETC is an equation of a supporting plane of the final convex hull.
Supporting Plane ?
No
Yes
E
F
Determine a new vertex.
NV = NV+1
Reorder new points so that the NVth point is the new vertex
G


G
Determine the equations of the 3 new supporting planes NE+1, NE+2, NE+3.
Reorder equations removing equation No. NETC.
Determine if the NV-th point is on the outside of any of other equations.
On outside ?
Yes
No
H


H
Determine if the NVth point was on the outside of more than 1 equation
More than 1 ?
Yes
No
Eliminate Inside Planes
B


F
NE-NETC:0
>
=
NETC=NETC+1
D
N-NV:0
=
>
Punch output (Boundary points)
N=NV
Remove all but 1 equation for each supporting line
Determine if any of the N points are not vertices of the final convex hull
All vertices?
Yes
No
Punch output (Boundary points)
Punch Output
A

```
C       CONVEX HULL-PROGRAM 3-NOT COPLANAR
        DIMENSION X(51),Y(51),Z(51),A(98),B(98),C(98),D(98),E(98)
        DIMENSION MM(98),NN(98),KK(98)
    100 KELEQ=1
C       INSID=2 IF INSIDE POINT HEADING HAS BEEN PUNCHED
        INSID=1
        NEWEQ=1
        NETC=1
        NE=4
        NV=4
        READ 1,C1,C2,C3
        DO 101 I=1,4
    101 READ 2,A(I),B(I),C(I),D(I),NN(I),MM(I),KK(I)
        READ 3,N
        PUNCH 4,N
        DO 102 I=1,N
    102 READ 1,X(I),Y(I),Z(I)
C       CHECKING FOR INSIDE POINTS
    103 IF(N-NV)133,161,213
    213 I=NV+1
    104 DO 105 J=1,NE
    105 E(J)=A(J)*X(I)+B(J)*Y(I)+C(J)*Z(I)+D(J)
        EE=E(1)
        DO 107 K=1,NE
        IF(EE-E(K))106,106,107
    106 EE=E(K)
    107 CONTINUE
        IF(EE)108,112,112
    108 GO TO (109,110),INSID
    109 PUNCH 5
        INSID=2
```

```
  110 PUNCH 6,X(I),Y(I),Z(I)
      N=N-1
      DO 111 L=I,N
      X(L)=X(L+1)
      Y(L)=Y(L+1)
  111 Z(L)=Z(L+1)
      IF(N+1-I)133,114,104
  112 IF(N-I)133,114,113
  113 I=I+1
      GO TO 104
C     CHECKING FOR SUPPORTING PLANES
  114 I123=1
      NINP=0
  115 IF(NE-NETC)161,116,116
  116 J=NETC
      MR=MM(J)
      NR=NN(J)
      KR=KK(J)
      E(J)=A(J)*X(MR)+B(J)*Y(MR)+C(J)*Z(MR)+D(J)
      IF(E(J))119,117,119
  117 E(J)=A(J)*X(NR)+B(J)*Y(NR)+C(J)*Z(NR)+D(J)
      IF(E(J))119,118,119
  118 E(J)=A(J)*X(KR)+B(J)*Y(KR)+C(J)*Z(KR)+D(J)
      IF(E(J))119,120,119
  119 PRINT 7
      IF(SENSE SWITCH 1)120,100
  120 DO 121 I=1,N
  121 E(I)=A(J)*X(I)+B(J)*Y(I)+C(J)*Z(I)+D(J)
      V=E(1)
      DO 123 I=1,N
```

```
      IF(V-E(I))122,122,123
  122 V=E(I)
      K=I
  123 CONTINUE
      IF(V)124,124,125
  124 NETC=NETC+1
      GO TO 115
  125 NV=NV+1
      XA=X(NV)
      YA=Y(NV)
      ZA=Z(NV)
      X(NV)=X(K)
      Y(NV)=Y(K)
      Z(NV)=Z(K)
      X(K)=XA
      Y(K)=YA
      Z(K)=ZA
  126 I1=MM(J)
      I2=NN(J)
      I3=KK(J)
      NE=NE+1
      GO TO (127,128,129,131),I123
  127 I1=NV
      GO TO 130
  128 I2=NV
      GO TO 130
  129 I3=NV
  130 MM(NE)=I1
      NN(NE)=I2
      KK(NE)=I3
```

```
      AA=Y(I1)*Z(I2)+Y(I2)*Z(I3)+Z(I1)*Y(I3)
      A(NE)=AA-Y(I3)*Z(I2)-Y(I1)*Z(I3)-Y(I2)*Z(I1)
      BB=X(I3)*Z(I2)+X(I1)*Z(I3)+X(I2)*Z(I1)
      B(NE)=BB-X(I1)*Z(I2)-X(I2)*Z(I3)-X(I3)*Z(I1)
      CC=X(I1)*Y(I2)+X(I2)*Y(I3)+Y(I1)*X(I3)
      C(NE)=CC-Y(I2)*X(I3)-X(I1)*Y(I3)-X(I2)*Y(I1)
      DD=X(I3)*Y(I2)*Z(I1)+Y(I3)*Z(I2)*X(I1)*X(I2)*Y(I1)*Z(I3)
      D(NE)=DD-X(I1)*Y(I2)*Z(I3)-X(I2)*Y(I3)*Z(I1)-Y(I1)-Z(I2)*X(I3)
      I123=I123+1
      GO TO 126
  131 NE=NE-1
      LJ=J
      KELEQ=3
      GO TO 157
  132 NEWEQ=J
      DO 135 I=NEWEQ,NE
      E(I)=A(I)*C1+B(I)*C2+C(I)*C3+D(I)
      IF(E(I))135,133,134
  133 PRINT 8
      GO TO 100
  134 A(I)=A(I)*(-1.)
      B(I)=B(I)*(-1.)
      C(I)=C(I)*(-1.)
      D(I)=D(I)*(-1.)
  135 CONTINUE
      DO 136 L=1,NE
  136 E(L)=A(L)*X(NV)+B(L)*Y(NV)+C(L)*Z(NV)+D(L)
      EE=E(1)
      DO 138 L=1,NE
      IF(EE-E(L))137,137,138
```

```
  137 EE=E(L)
      J=L
  138 CONTINUE
      IF(EE)140,140,139
  139 I123=1
      NINP=NINP+1
      GO TO 126
  140 IF(NINP)133,103,141
C     ELIMINATING INSIDE PLANES
  141 I=1
      KELEQ=1
  142 K=I+1
  143 KMN=MM(I)
      ID3=0
  144 IF(MM(K)-KMN)145,149,145
  145 IF(NN(K)-KMN)146,149,146
  146 IF(KK(K)-KMN)147,149,147
  147 IF(NE-K)133,154,148
  148 K=K+1
      GO TO 143
  149 ID3=ID3+1
      IF(KMN-MM(I))151,150,151
  150 KMN=NN(I)
      GO TO 144
  151 IF(KMN-NN(I))153,152,153
  152 KMN=KK(I)
      GO TO 144
  153 IF(ID3-3)154,156,133
  154 IF(NE-I-1)133,103,155
  155 I=I+1
```

```
      GO TO 142
  156 LJ=I
      IDO2=0
  157 NE=NE-1
      DO 158 LM=LJ,NE
      A(LM)=A(LM+1)
      B(LM)=B(LM+1)
      C(LM)=C(LM+1)
      D(LM)=D(LM+1)
      MM(LM)=MM(LM+1)
      NN(LM)=NN(LM+1)
  158 KK(LM)=KK(LM+1)
      GO TO (159,165,132),KELEQ
  159 IF(IDO2)133,160,142
  160 IDO2=1
      LJ=K-1
      GO TO 157
C     DETERMINING BOUNDARY POINTS THAT ARE NOT VERTICES
  161 IBDRY=1
      IF(N-NV)133,164,162
  162 IBDRY=2
      K=NV+1
      PUNCH 9
      DO 163 J=K,N
  163 PUNCH 10,X(J),Y(J),Z(J)
      N=NV
C     ELIMINATING MORE THAN ONE EQUATION OF THE SAME PLANE
  164 I=1
  165 MZ=MM(I)
      NZ=NN(I)
```

```
      KZ=KK(I)
      J=I+1
  166 EZ=A(J)*X(MZ)+B(J)*Y(MZ)+C(J)*Z(MZ)+D(J)
      IF(EZ)170,167,170
  167 EZ=A(J)*X(NZ)+B(J)*Y(NZ)+C(J)*Z(NZ)+D(J)
      IF(EZ)170,168,170
  168 EZ=A(J)*X(KZ)+B(J)*Y(KZ)*C(J)*Z(KZ)*D(J)
      IF(EZ)170,169,170
  169 LJ=J
      KELEQ=2
      GO TO 157
  170 IF(NE-J)133,172,171
  171 J=J+1
      GO TO 166
  172 IF(NE-I-1)133,174,173
  173 I=I+1
      GO TO 165
  174 I=1
      DO 175 K=1,N
  175 E(K)=0.
  176 IF(N-I)185,177,177
  177 IV=0
      DO 179 K=1,NE
      VERT=A(K)*X(I)+B(K)*Y(I)+C(K)*Z(I)+D(K)
      IF(VERT)179,178,179
  178 IV=IV+1
  179 CONTINUE
      IF(IV-3)182,180,180
  180 IF(N-I)133,185,181
  181 I=I+1
```

```
      GO TO 177
  182 IF(IBDRY-2)183,184,133
  183 PUNCH 9
  184 PUNCH 10,X(I),Y(I),Z(I)
      E(I)=1.
      I=I+1
      GO TO 176
  185 PUNCH 11
      K=1
      DO 187 I=1,N
      IF(E(I))133,186,187
  186 PUNCH 12,K,X(I),Y(I),Z(I)
      K=K+1
  187 CONTINUE
      PUNCH 13
      PUNCH 14
      PUNCH 15
      PUNCH 16
      DO 188 I=1,NE
  188 PUNCH 12,I,A(I),B(I),C(I),D(I)
      PUNCH 17
      I=1
  189 IF(E(I))133,192,190
  190 N=N-1
      DO 191 K=I,N
      E(K)=E(K+1)
      X(K)=X(K+1)
      Y(K)=Y(K+1)
  191 Z(K)=Z(K+1)
      IF(N-I)194,189,189
```

```
192 IF(N-I)194,194,193
193 I=I+1
    GO TO 189
194 K=1
195 I=1
196 DO 197 J=1,3
197 MM(J)=0
    NP=0
198 E(I)=A(K)*X(I)+B(K)*Y(I)+C(K)*Z(I)+D(K)
    IF(E(I))199,201,199
199 IF(N-I)133,205,200
200 I=I+1
    GO TO 198
201 NP=NP+1
    MM(NP)=I
    IF(NP-3)202,203,133
202 IF(N-I)133,206,200
203 PUNCH 19,MM(1),MM(2),MM(3),K
    IF(N-I)133,210,204
204 I=I+1
    GO TO 196
205 IF(NP)133,210,206
206 GO TO (207,208,209),NP
207 PUNCH 20,MM(1),K
    GO TO 210
208 PUNCH 21,MM(1),MM(2),K
    GO TO 210
209 PUNCH 19,MM(1),MM(2),MM(3),K
210 IF(NE-K)133,212,211
211 K=K+1
```

```
      GO TO 195
  212 PUNCH 18
      GO TO 100
    1 FORMAT (3E14.8)
    2 FORMAT (4E14.8,3I4)
    3 FORMAT (I4)
    4 FORMAT (/11HCONVEX HULL/2HN=,I4/)
    5 FORMAT (/13HINSIDE POINTS//13X,1HX,15X,1HY,15X,1HZ/)
    6 FORMAT (6X,E14.8,2X,E14.8,2X,E14.8)
    7 FORMAT (19HDATA NOT ACCEPTABLE)
    8 FORMAT (16HERROR IN PROGRAM)
    9 FORMAT (/15HBOUNDARY POINTS/13X,1HX,15X,1HY,15X,1HZ/)
   10 FORMAT (6X,E14.8,2X,E14.8,2X,E14.8,2X,E14.8)
   11 FORMAT (/8HVERTICES//2X,3HNO.,8X,1HX,15X,1HY,15X,1HZ/)
   12 FORMAT (I3,3X,E14.8,2,XE14.8,2X,E14.8,2X,E14.8)
   13 FORMAT (/30HEQUATIONS OF SUPPORTING PLANES,16H ARE IN THE FORM)
   14 FORMAT (20HAX + BY + CZ + D = 0/)
   15 FORMAT (8HEQUATION,5X,1HA,15X,1HB,15X,1HC,15X,1HD)
   16 FORMAT (2X,3HNO./)
   17 FORMAT (//3X,8HVERTICES,10X,16HSATISFY EQUATION/)
   18 FORMAT (//31HCONVEX HULL HAS BEEN DETERMINED///)
   19 FORMAT (I3,1H,,I3,1H,,I3,15X,I3)
   20 FORMAT (I3,23X,I3)
   21 FORMAT (I3,1H,,I3,19X,I3)
      END
```

CHAPTER 6

EXAMPLES

Example 1

INPUT DATA FOR PROGRAM 1

```
 3
+.20000000E+01-.50000000E+01+.20000000E+02
+.20000000E+03-.40000000E+01-.50000000E+02
-.30000000E+02+.80000000E+02+.40000000E+01
```

OUTPUT FROM PROGRAM 1

CONVEX HULL
N = 3

VERTICES

NO.	X	Y	Z
1	.20000000E+01	-.50000000E+01	.20000000E+02
2	.20000000E+03	-.40000000E+01	-.50000000E+02
3	-.30000000E+02	.80000000E+02	.40000000E+01

EQUATIONS OF SUPPORTING LINES ARE IN THE FORM
AX + BY + CZ + D = 0

EQUATION NO.	A	B	C	D
1	.10000000E+01	-.19800000E+03	.00000000E-99	-.99200000E+03
2	-.70000000E+02	.00000000E-99	-.19800000E+03	.41000000E+04
3	-.84000000E+02	-.23000000E+03	.00000000E-99	.15880000E+05
4	-.54000000E+02	.00000000E-99	-.23000000E+03	-.70000000E+03
5	-.85000000E+02	-.32000000E+02	.00000000E-99	.10000000E+02
6	.16000000E+02	.00000000E-99	-.32000000E+02	.60800000E+03

VERTICES	SATISFY EQUATION
1, 2	1
1, 2	2
2, 3	3
2, 3	4
1, 3	5
1, 3	6

CONVEX HULL HAS BEEN DETERMINED

<u>Example 2</u>

INPUT DATA FOR PROGRAM 1

```
 9
+.20000000E+01+.38000000E+02-.16000000E+02
+.29000000E+02+.38000000E+02-.16000000E+02
```

```
-.60000000E+03+.38000000E+02-.16000000E+02
-.26800000E+03+.38000000E+02-.16000000E+02
-.11000000E+02+.38000000E+02-.16000000E+02
+.29800000E+03+.38000000E+02-.16000000E+02
+.36010000E+04+.38000000E+02-.16000000E+02
+.65020000E+04+.38000000E+02-.16000000E+02
+.26000000E+02+.38000000E+02-.16000000E+02
```

OUTPUT FROM PROGRAM 1

CONVEX HULL
N= 9

POINTS ARE COLLINEAR

ENDPOINTS

NO.	X	Y	Z
1	.65020000E+04	.38000000E+02	-.16000000E+02
2	-.60000000E+03	.38000000E+02	-.16000000E+02

EQUATIONS OF THE LINE CONTAINING THE GIVEN N POINTS
ARE IN THE FOR AX + BY + CZ + D = 0

EQUATION NO.	A	B	C	D
1	.00000000E-99	.71020000E+04	.00000000E-99	-.26987600E+06
2	.00000000E-99	.00000000E-99	.71020000E+04	.11363200E+06

CONVEX HULL HAS BEEN DETERMINED

Example 3

```
INPUT DATA FOR PROGRAM 1
 1
+.10000000E+01+.20000000E+01+.30000000E+01

OUTPUT FROM PROGRAM 1

CONVEX HULL
N=    1

               X                 Y                 Z

       .10000000E+01     .20000000E+01     .30000000E+01

CONVEX HULL HAS BEEN DETERMINED
```

Example 4

```
INPUT DATA FOR PROGRAM 1

 8
+.10000000E+01+.20000000E+01-.50000000E+01
+.40000000E+01+.20000000E+01-.30000000E+01
+.40000000E+01+.20000000E+01+.40000000E+01
+.40000000E+01+.20000000E+01+.00000000E+00
+.40000000E+01+.20000000E+01-.10000000E+01
+.10000000E+01+.20000000E+01+.70000000E+01
```

```
+.00000000E+00+.20000000E+01+.00000000E+00
-.20000000E+01+.20000000E+01+.00000000E+00
```

OUTPUT DATA FROM PROGRAM 1 AND INPUT DATA FOR PROGRAM 2

```
-.00000000E-99-.45000000E+02-.00000000E-99 .90000000E+02
   8 2
 .40000000E+01-.10000000E+01 .20000000E+01
 .10000000E+01 .70000000E+01 .20000000E+01
-.20000000E+01 .00000000E-99 .20000000E+01
 .10000000E+01-.50000000E+01 .20000000E+01
 .40000000E+01-.30000000E+01 .20000000E+01
 .40000000E+01 .40000000E+01 .20000000E+01
 .40000000E+01 .00000000E-99 .20000000E+01
 .00000000E-99 .00000000E-99 .20000000E+01
```

OUTPUT FROM PROGRAM 2

CONVEX HULL
N=   8

INSIDE POINTS

```
            X                Y                Z

      .00000000E-99    .20000000E+01    .00000000E-99
```

BOUNDARY POINTS

```
            X                Y                Z

      .40000000E+01    .20000000E+01    .00000000E-99
      .40000000E+01    .20000000E+01   -.10000000E+01
```

VERTICES

NO.	X	Y	Z
1	.10000000E+01	.20000000E+01	.70000000E+01
2	-.20000000E+01	.20000000E+01	.00000000E-99
3	.40000000E+01	.20000000E+01	.40000000E+01
4	.10000000E+01	.20000000E+01	-.50000000E+01
5	.40000000E+01	.20000000E+01	-.30000000E+01

EQUATIONS OF SUPPORTING LINES ARE IN THE FORM
AX + BY + CZ D = 0

NO.	A	B	C	D
1	-.70000000E+01	.00000000E-99	.30000000E+01	-.14000000E+02
2	.50000000E+01	.00000000E-99	-.00000000E-99	-.20000000E+02
3	.30000000E+01	.00000000E-99	.30000000E+01	-.24000000E+02
4	-.50000000E+01	.00000000E-99	-.30000000E+01	-.10000000E+02
5	.20000000E+01	.00000000E-99	-.30000000E+01	-.17000000E+02

VERTICES	SATISFY EQUATION
2, 1	1
5, 3	2
1, 3	3
2, 4	4
4, 5	5

COEFFICIENTS OF AN EQUATION OF THE PLANE CONTAINING
THE GIVEN N POINTS IN THE FORM AX + BY + CZ + D = 0

A	B	C	D
-.00000000E-99	-.45000000E+02	-.00000000E-99	.90000000E+02

CONVES HULL HAS BEEN DETERMINED

Example 5

INPUT DATA FOR PROGRAM 1

```
15
+.40000000E+01+.20000000E+01-.60000000E+01
+.60000000E+01+.20000000E+01-.40000000E+01
+.10000000E+02+.20000000E+01-.50000000E+01
+.19000000E+02+.40000000E+01-.70000000E+01
-.30000000E+01+.20000000E+01-.30000000E+01
+.90000000E+01+.30000000E+01-.60000000E+01
+.80000000E+01+.20000000E+01-.70000000E+01
+.90000000E+01+.20000000E+01-.40000000E+01
+.70000000E+01+.20000000E+01-.12000000E+02
+.16000000E+02+.10000000E+01+.30000000E+01
+.40000000E+01+.40000000E+01+.10000000E+01
+.30000000E+01+.30000000E+01-.20000000E+01
+.80000000E+01+.30000000E+01-.10000000E+01
+.40000000E+01+.30000000E+01-.20000000E+01
+.10000000E+02+.20000000E+01-.10000000E+01
```

OUTPUT DATA FROM PROGRAM 1 AND INPUT DATA FOR PROGRAM 3

```
 .90000000E+01 .27500000E+01-.15000000E+01
-.16000000E+02 .11600000E+03-.30000000E+02-.37000000E+03   1   2   3
 .24000000E+02 .12600000E+03 .45000000E+02-.64500000E+03   1   2   4
 .80000000E+01-.20800000E+03-.60000000E+02 .26000000E+03   1   3   4
-.16000000E+02-.34000000E+02 .45000000E+02 .15500000E+03   2   3   4
 15
 .19000000E+02 .40000000E+01-.70000000E+01
 .40000000E+01 .40000000E+01 .10000000E+01
-.30000000E+01 .20000000E+01-.30000000E+01
 .16000000E+02 .10000000E+01 .30000000E+01
 .70000000E+01 .20000000E+01-.12000000E+02
 .40000000E+01 .20000000E+01-.60000000E+01
 .60000000E+01 .20000000E+01-.40000000E+01
 .10000000E+02 .20000000E+01-.50000000E+01
 .90000000E+01 .30000000E+01-.60000000E+01
 .80000000E+01 .20000000E+01-.70000000E+01
 .90000000E+01 .20000000E+01-.40000000E+01
 .30000000E+01 .30000000E+01-.20000000E+01
 .80000000E+01 .30000000E+01-.10000000E+01
 .40000000E+01 .30000000E+01-.20000000E+01
 .10000000E+02 .20000000E+01-.10000000E+01

OUTPUT FROM PROGRAM 3

CONVEX HUL
N=   15

INSIDE POINTS
```

X	Y	Z
.30000000E+01	.30000000E+01	-.20000000E+01
.80000000E+01	.30000000E+01	-.10000000E+01
.40000000E+01	.30000000E+01	-.20000000E+01
.10000000E+02	.20000000E+01	-.10000000E+01
.40000000E+01	.20000000E+01	-.60000000E+01
.60000000E+01	.20000000E+01	-.40000000E+01
.10000000E+02	.20000000E+01	-.50000000E+01
.90000000E+01	.30000000E+01	-.60000000E+01
.80000000E+01	.20000000E+01	-.70000000E+01
.90000000E+01	.20000000E+01	-.40000000E+01

VERTICES

NO.	X	Y	Z
1	.19000000E+02	.40000000E+01	-.70000000E+01
2	.40000000E+01	.40000000E+01	.10000000E+01
3	-.30000000E+01	.20000000E+01	-.30000000E+01
4	.16000000E+02	.10000000E+01	.30000000E+01
5	.70000000E+01	.20000000E+01	-.12000000E+02

EQUATIONS OF SUPPORTING PLANES ARE IN THE FORM
AX + BY + CZ + D = 0

EQUATION NO.	A	B	C	D
1	.24000000E+02	.12600000E+03	.45000000E+02	-.64500000E+03

2	-.16000000E+02	-.34000000E+02	.45000000E+02	.15500000E+03
3	-.18000000E+02	.10300000E+03	-.20000000E+02	-.32000000E+03
4	-.16000000E+02	.17100000E+03	-.30000000E+02	-.59000000E+03
5	.35000000E+02	-.13500000E+03	-.30000000E+02	-.33500000E+03
6	-.90000000E+01	-.23100000E+03	-.10000000E+02	.40500000E+03

VERTICES	SATISFY EQUATION
1, 2, 4	1
2, 3, 4	2
2, 3, 5	3
1, 2, 5	4
1, 4, 5	5
3, 4, 5	6

CONVEX HULL HAS BEEN DETERMINED

<u>Example 6</u>

INPUT DATA FOR PROGRAM 1

```
10
+.30000000E+01+.40000000E+01+.80000000E+01
+.60000000E+01-.30000000E+01+.70000000E+01
+.10000000E+01+.20000000E+01-.50000000E+01
+.40000000E+01+.20000000E+01-.30000000E+01
+.40000000E+01+.20000000E+01+.40000000E+01
+.40000000E+01+.20000000E+01+.00000000E+00
```

```
+.40000000+01+.20000000E+01-.10000000E+01
+.10000000E+01.20000000E+01+.70000000E+01
+.00000000E+00.20000000E+01+.00000000E+00
-.20000000E+01.20000000E+01+.00000000E+00
```

OUTPUT DATA FROM PROGRAM 1 AND INPUT DATA FOR PROGRAM 3

```
 .27500000E+01 .12500000E+01 .30000000E+01
-.54000000E+02-.29000000E+02 .41000000E+02-.50000000E+02   1   2   3
-.60000000E+01 .63000000E+02-.12000000E+02-.13800000E+03   1   2   4
 .75000000E+02 .32000000E+02 .10000000E+01-.36100000E+03   1   3   4
-.15000000E+02-.66000000E+02-.30000000E+02 .10200000E+03   2   3   4
  10
 .30000000E+01 .40000000E+01 .80000000E+01
-.20000000E+01 .20000000E+01 .00000000E-99
 .60000000E+01-.30000000E+01 .70000000E+01
 .40000000E+01 .20000000E+01-.30000000E+01
 .10000000E+01 .20000000E+01-.50000000E+01
 .40000000E+01 .20000000E+01 .40000000E+01
 .40000000E+01 .20000000E+01 .00000000E-99
 .40000000E+01 .20000000E+01-.10000000E+01
 .10000000E+01 .20000000E+01 .70000000E+01
 .00000000E-99 .20000000E+01 .00000000E-99
```

OUTPUT FROM PROGRAM 3

CONVEX HULL
N= 10

INSIDE POINTS

X	Y	Z
.00000000E-99	.20000000E+01	.00000000E-99

BOUNDARY POINTS

X	Y	Z
.40000000E+01	.20000000E+01	-.10000000E+01
.40000000E+01	.20000000E+01	.00000000E-99

VERTICES

NO.	X	Y	Z
1	.30000000E+01	.40000000E+01	.80000000E+01
2	-.20000000E+01	.20000000E+01	.00000000E-99
3	.60000000E+01	-.30000000E+01	.70000000E+01
4	.40000000E+01	.20000000E+01	-.30000000E+01
5	.10000000E+01	.20000000E+01	.70000000E+01
6	.10000000E+01	.20000000E+01	-.50000000E+01
7	.40000000E+01	.20000000E+01	.40000000E+01

EQUATIONS OF SUPPORTING PLANES ARE IN THE FORM
AX + BY + CZ + D = 0

EQUATION NO.	A	B	C	D
1	-.50000000E+01	-.50000000E+01	.20000000E+02	-.12500000E+03
2	-.35000000E+02	-.35000000E+02	.15000000E+02	-.00000000E-99
3	-.14000000E+02	.11000000E+02	.60000000E+01	-.50000000E+02

4	.40000000E+01	.35000000E+02	-.60000000E+01	-.10400000E+03
5	-.10000000E+02	.49000000E+02	-.60000000E+01	-.11800000E+03
6	.10000000E+02	-.26000000E+02	-.15000000E+02	-.33000000E+02
7	-.25000000E+02	-.61000000E+02	-.15000000E+02	.72000000E+02
8	.14000000E+02	.70000000E+01	-.00000000E-99	-.70000000E+02
9	.35000000E+02	.14000000E+02	-.00000000E-99	-.16800000E+03
10	.26000000E+02	.11000000E+02	.10000000E+01	-.13000000E+03

VERTICES	SATISFY EQUATION
1, 3, 5	1
2, 3, 5	2
1, 2, 5	3
1, 4, 6	4
1, 2, 6	5
3, 4, 6	6
2, 3, 6	7
1, 4, 7	8
3, 4, 7	9
1, 3, 7	10

CONVEX HULL HAS BEEN DETERMINED

Chapter 7

MODIFICATIONS OF THE COMPUTER PROCEDURE

Errors involved in numerical calculations depend in part on the calculating device used. In determining the convex hull using the IBM 1620 with 40,000 storage positions we encounter two types of errors. These errors would be present to some degree on any computer used. The first type of errors are inherent errors due to the fact that the coordinates of a point can be specified to at most eight significant digits. Thus we may be using a point whose coordinates are specified to eight significant digits to represent a point whose coordinates require more than eight digits (perhaps infinitely many) for their exact specifications. In such cases we are in effect approximating the actual convex hull by determining the convex hull of a given set of approximation points.

The second type of error is a truncation error, since only eight significant digits can be maintained throughout the calculations and no rounding occurs. Due to truncation errors the coefficients of the equations of supporting planes (or lines) may be approximations to the actual coefficients if the actual coefficients require more than eight significant digits for their exact specification.

This presents us with a problem, since the equation of a plane (or line) that we obtain may not be the actual equation but rather an approximation to the actual equation. Hence in substituting the coordinates of a point in this plane (or line) into the equation we have determined to represent the plane (or line) the value obtained may not be zero. At present the programs are set up so that after the coefficients of an equation of the plane (or line) have been determined we substitute the coordinates of each of the three points used to determine the equation back into the equation and if any one of the three values obtained is not zero the message "DATA NOT ACCEPTABLE" is typed on the console and the processing terminates.

There still exists the possibility however that all three of the values obtained may be zero and yet on substituting the coordinates of another point in the plane (or line) into the equation representing the plane (or line) the value obtained is not zero.

A possible solution to this problem would be to define an interval say from -e to +e and agree that if the value obtained on substituting the coordinates of a point into the left member of the equation fell within this interval we would consider that the point satisfied the equation and hence was a point of the plane (or line).

In doing this however we are presented with another problem. We may be considering a point to be on the plane (or line) when actually it is not. Thus it may be that a point actually is a vertex of the convex hull but is not recognized as a vertex. By choosing e small enough however, we could be assured that a point was within a certain distance of a plane (or line) if the value obtained on substituting the coordinates of the point into the equation fell within the interval from -e to +e.

We now want to consider a modification to the procedure that would simplify the task of normalizing each of the equations of the supporting planes (or lines).

In determining the convex hull by the procedure described in Chapter 3 we first determine the vertices of H_1 and then using these points we determine the coordinates of the point C. Throughout the procedure we use the point C to normalize the equations of the supporting planes (or lines). We do this by substituting the coordinates of the point C into the left member of the equation of the plane (or line) and if the value obtained is positive we multiply the equation by (-1).

If after we determine the point C we translate the origin to this point, then to normalize an equation we could merely check the constant term and if it is positive multiply the equation by (-1). This translation would probably be worthwhile if the convex hull were being determined by hand. If a computer is being used the advisability of such a translation depends on several factors. Among the things to be considered are the amount of storage available and the number of equations to be normalized in relation to the number of points to be translated.

Since it would be desirable to obtain the vertices and supporting planes with respect to the original coordinate system, we would have to perform another translation once the vertices were determined. For the IBM 1620 with 40,000 storage positions the program for determining the convex hull must be divided into three separate programs because of the amount of storage required. Thus the translation might not be advisable since it would require more storage. However if storage is no problem such a translation would reduce the amount of time required to normalize an equation. Whether or not the total amount of time required to determine the convex hull is reduced depends on the number of equations to be normalized in relation to the number of points to be translated.

BIBLIOGRAPHY

1. Alexandroff, Paul. Elementary concepts of topology. Tr by Alan E. Farley. New York, Dover, 1961. 73 p.

2. Arnold, B. H. Intuitive concepts in elementary topology. Englewood Cliffs, N. J., Prentice-Hall, 1962. 180 p.

3. Balinski, Michel L. An algorithm for finding vertices of convex polyhedral sets. Ph.D. thesis. Princeton, Princeton University, 1959. 38 numb. leaves. (Microfilm)

4. Eggleston, H. G. Convexity. Cambridge, University Press, 1958. 136 p. (Cambridge Tracts in Mathematics and Mathematical Physics, No. 47)

5. Eisenhart, Luther Pfahler. Coordinate geometry. New York, Dover, 1939, 297 p.

6. Hadley, G. Linear algebra. Reading, Mass., Addison-Wesley, 1961. 290 p.

7. International Business Machines. Reference manual IBM 1620 Fortran. San Jose, 1962. 94 p.

8. Love, Clyde E. and Earl D. Rainville. Analytic geometry. New York, Macmillan, 1956. 302 p.

9. Mancill, Julian D. Modern analytical trigonometry. New York, Dodd, Mead, 1960. 329 p.

10. Motzkin, T. S. The double description method. In: Contributions to the theory of games, ed. by H. W. Kuhn and A. W. Tucker. vol. 2. Princeton, New Jersey, Princeton University Press, 1953. p.51-73. (Annals of Mathematics Studies no. 28)

11. Smith, Edward S., Meyer Salkover and Howard K. Justice. Analytic geometry. New York, Wiley, 1954. 306 p.

12. Yaglom, I. M. and V. G. Boltyanskii. Convex figures. Tr. by Paul E. Kelly and Lewis F. Walton. New York, Holt, Rinehart and Winston, 1961. 301 p.