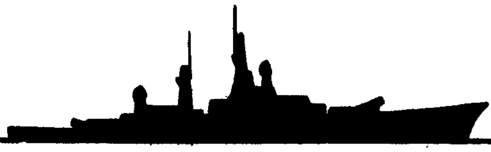


MATHEMATICAL SHIP LOFTING

PART 1. THEORY

Technical Report 1.0.0-1
Volume 2 of Final Report
Contract NObs-4427

March 1963

TODD SHIPYARDS CORPORATION
LOS ANGELES DIVISION SAN PEDRO CALIFORNIA

FORWARD

This publication is Volume 2 of six volumes of the final report on Mathematical Ship Lofting and Numerical Control of Shipyard Fabricating Equipment. The work reported on herein was performed under Bureau of Ships Contract NObs-4427, Code 770, during the period from April 1961 to March 1963.

The volumes of this final report have the following titles:

Vol. 1. Project Summary Report (Technical Report 9.0.0)

Vol. 2. Mathematical Ship Lofting -
Part 1. - Theory (Technical Report 1.0.0-1)
Part 2. - Operating Manual (Technical Report 1.0.0-2)

Vol. 3. Mathematical Ship Lofting - Summary Report (Technical Report 1.5.0)

Vol. 4. Programming System for Numerically Controlled Flame Cutting of Ships Parts - Operating Manual (Technical Report 5.0.0)

Vol. 5. Development and Testing of Programming System for Numerically Controlled Flame Cutting of Ships' Parts - Summary Report (Technical Report 5.5.0)

Vol. 6. Numerically Controlled Shipyard Fabricating Equipment - Summary Report (Technical Report 3.0.0)

The work was accomplished by the Research and Development Group of the Los Angeles Division of Todd Shipyards Corporation, San Pedro, Calif. Mr. Thomas G. Smith was Project Manager for the work, and Dr. Henry A. Schade of the University of California, Berkeley, was Principal Consultant.

This report contains the results of work accomplished by the Mathematics Team which consists of:

D. A. Atkins, Naval Architect - Mathematician (Team Supervisor)

R. A. Tapia, Mathematician

J. C. Fassino, Naval Architect - Mathematician

T. A. Dunaway, Computer Programmer

S. A. Berger, Mathematician

W. C. Webster, Naval Architect - Mathematician

Janice H. Thompson, Secretary

Others of the Todd Staff, whose contributions and participation during the Project were responsible for its successful culmination, consisted of:

Eric Chein, Data Processing Supervisor
R. W. Feeny, Numerical Control Engineer
H. S. Janssen, European Representative
K. M. Jones, Naval Architect
W. R. Lauper, Mechanical Engineer
M. R. Ward, Jr., Naval Architect

The Project is indebted to D. A. Anderson and Milton Drandell, both of IBM Corporation, and M. L. Juncosa and Philip Wolfe of Rand Corporation, who contributed to the effort.

To the following representatives of the Navy Department go our sincere thanks for their assistance and guidance during the work:

Bureau of Ships:

Captain W. S. Dawson
Captain F. E. Gorman
Comdr. William Harrison
Mr. T. H. Sarchin
Mr. G. Vidlak

David W. Taylor Model Basin:

Dr. Feodor Theilheimer
Dr. P. C. Pien
Mr. William Starkweather

For their support and continued interest, our thanks to Mr. W. Taylor Potter and Mr. Karl Fixman of the U. S. Maritime Administration.

To Admiral R. K. James, Captain D. L. Carroll Jr., Captain T. J. Banvard, and Mr. Ed Kemp of the Bureau of Ships, are due the gratitude of the industry for initiating this work. The Project is particulerly grateful for their early assistance in getting the work under way. To Mr. Ed Kemp go our sincerest thanks for his deep interest in this work and for his continued guidance to the Project.

ABSTRACT

A method of fairing ships' hulls and obtaining a surface equation representing the ships' hull through the use of linear programming is presented. Methods for producing loft data from this equation are also presented.

CONTENTS

Section I

INTRODUCTION

The mathematical tool called "Linear Programming" has been in existence for many years. Since 1951, when George Dantzig* first published the simplex method for solving linear programming problems, and through increased availability of large-scale digital computers, linear programming use has become widespread. Primarily, this use of linear programming has been confined to economic studies of product mixes, ship scheduling, plant locations, etc. In this report a new use of linear programming is described: that of curve and surface fitting.

This use of linear programming has enabled the formulation of a more sophisticated approach to the problem of defining a ship's hull mathematically. Specifically, it is possible to require that the curve or surface meet certain conditions of fairness and that the curve come as close to the offsets as possible while meeting these conditions.

Also presented are some methods for extracting the information required to construct the ship from the faired curve or surface.

---

*T. C. Koopmans, ed., <u>Activity Analysis of Production and Allocation</u>, Chap. XXI, New York, Wiley, 1951

Section II

CRITERIA AND BACKGROUND

CRITERIA

The title, "Mathematical Lofting," is meant to apply to:

1. The overall process of mathematically fairing a ship's hull
2. Providing from this hull, all of the information necessary to cut and shape parts influenced by the hull

Mathematical fairing is the basic key to this process; however, mathematical fairing becomes most efficient when integrated into a system which meets the following criteria:

(1) The system should accept as data the preliminary information on hull definition developed by the naval architect.

(2) It should include a method for eliminating gross errors in this input data.

(3) It should, from this data, obtain a fully faired hull form from which the ship can be produced.

(4) The fairing process should operate with as little human intervention as possible.

(5) The system should develop, as much as is practicable, complete loft information now provided by the manual loft.

Criterion (3) requires that the hull be fully faired. The term "fair" has been in use for a long time with many varied interpretations. To provide a basis for a mathematical fairing method, we must reduce the subjective definition of the word "fair" to a more precise mathematical definition. Curves which meet the following mathematical requirements also meet the subjective requirements of experienced loftsmen:

1-a The curve Y(X) must be continuous

1-b Its first derivative must be continuous

1-c Its second derivative must be continuous*

1-d The curve must be completely free of extraneous inflection points while possessing those inherent in the data.

1-e Deviation from scaled offsets must be as small as possible, provided above four conditions hold.

1-f Curve must look good to the eye.

Condition 1-e is added for completeness when beginning with scaled offsets; it, and 1-f are also the only conditions possibly possessing any subjectivity in their interpretation. Condition 1-d is imperative if the faired curves are to be used for construction purposes as opposed to mathematical expressions used for hydrodynamic studies only.

A system designed to meet the above criteria would start with a table of roughly faired offsets and other information from the naval architect's preliminary lines drawing. It should then provide a method for eliminating any gross errors due, for example, to misreading a scale or transcribing the wrong information.

From this data, it should provide, with a minimum of human intervention or decision, a mathematical description of a fair hull. Next, it must be able to extract from the mathematical definition of the hull sufficient information to produce the ship. A diagram of such a system is presented in Fig. 1.

---

*Continuity of first and second derivatives is equivalent to continuous slope and curvature.

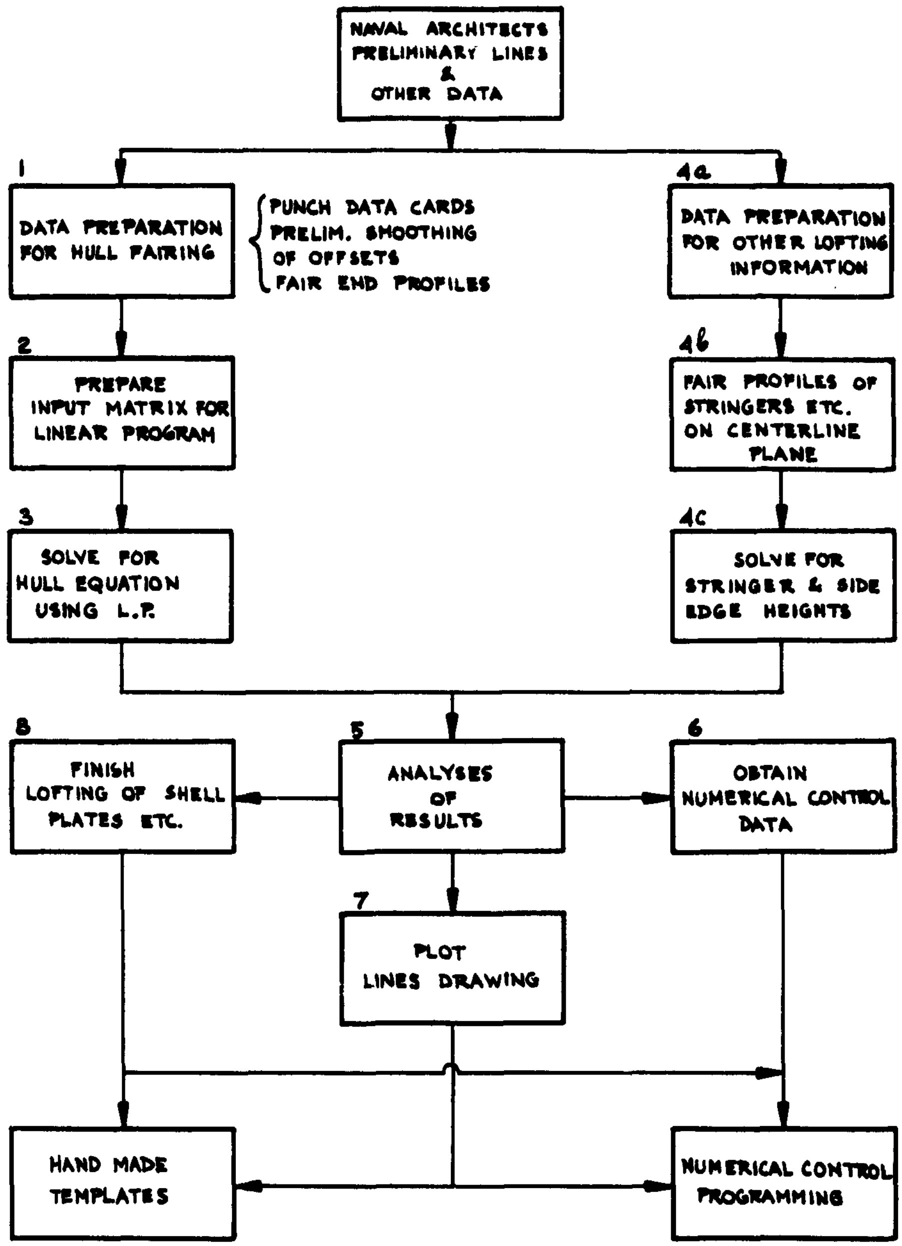


Fig. 1 Mathematical Lofting System

BACKGROUND

An extensive study of most methods proposed for mathematical lofting was undertaken prior to the initial formulation of the present system. This study revealed that all proposed systems fail, in one way or another, to meet the criteria required of a system oriented toward production. From the system which most fully met the criteria (method proposed by F. Theilheimer*), was borrowed the class of function with which to represent ships-form, as well as the notation for this function.

---

*Feodor Theilheimer and William Starkweather, The Fairing of Ship Lines on a High-Speed Computer, David W. Taylor Model Basin, Appl. Math. Lab., Report No. 1474, Jan 1961

Section III

MATHEMATICAL FAIRING

A. GENERAL DISCUSSION

The manual loft has used the batten (graphical-spline) very satisfactorily for interpolating between points as well as "fairing" points. We feel it would be desirable to mathematically-fair with the mathematical equivalent of the graphical-spline, the "analytic-spline."

Analytic Spline

Thin beam theory (based on the Bernoulli-Euler Equation) shows that a thin beam (e.g., batten) supported at a finite number of points, under small deflections, can be represented by a series of cubic equations (a single cubic in each interval defined by two successive points of support) with any two neighboring cubics having equal ordinate values, first derivatives and second derivatives at the point of support common to both curves. The third derivatives at this common point are not necessarily equal, resulting in a discontinuous third derivative for the curve representing the spline. The discontinuity of the third derivative at the juncture point is proportional to the force applied to the beam at this point.

This situation closely parallels the one existing when ships' lines are drawn from battens, and again when these curves are faired full-scale on the mold loft floor.

> Definition: An analytic-spline through a set of points is a curve through these same points consisting of a cubic in each interval defined by two successive points with any two neighboring cubics having equal ordinates, first derivatives, and second derivatives at the point common to both curves.

Returning to the original requirements for fairness of a line (Page II-1) it can clearly be seen that all an analytic-spline needs in order for it to satisfy all fairness criteria, is that it possess only those inflection points inherent in the data, while minimizing the deviation between curve and points.

Having settled on the type of curve with which to represent ships-form, we must now devise a procedure for minimizing the deviation between curve and points while requiring the occurrence of inflection points to be compatible with the information inherent in the data.

One point we would like to emphasize is that in curve-fairing, as opposed to curve-fitting, the main objective is not to pass exactly through the data points, but that the curve be pleasing to the eye and possess only those inflection points as indicated by the data. Only after these fairness conditions have been satisfied can we ask for the curve which minimizes the deviation. The problem would be less complicated were it a curve-fitting problem and not a curve-fairing problem.

Linear Programming

One computer routine which can be very satisfactorily adapted to both curve-fitting and curve-fairing is entitled "Linear Programming." The general linear programming problem can be described as follows:

> Given a set of M linear inequalities or equations in r variables, we wish to find non-negative values of these variables which will satisfy the constraints and maximize or minimize some linear function of the variables.*

The application of linear programming to curve fitting stems from the

*For a detailed definition of Linear Programming refer to G. Hadley, Linear Programming, Addison-Wesley Publishing Co., 1962 following use of linear programming:

> Linear programming offers a method of obtaining an optimal solution to a set of linear equations which has an infinite number of solutions (i.e., a procedure for determining an optimal element from a set with an infinite number of elements).

The latter interpretation of linear programming indicates that the problem of fitting a curve to a set of points can be stated as a linear programming problem by requiring the linear programming code to select from the infinite number of curves in the neighborhood of the data points, the one which comes closest to these points (i.e., select the curve which satisfies all other fairness requirements and minimizes the deviation between curve and points).

Point Smoothing and Second Derivative Requirement

The data-points which serve as input to the lofting system (mathematical of manual) are traditional offsets scaled from a preliminary design drawing prepared by a naval architect. This drawing is approximately one-fiftieth the size of the actual ship. The smallness of this drawing, although being beneficial to the naval architect, invariably causes small reading errors in the scaled offsets. There are many opportunities for errors aside from those described above to enter into the data. These errors are usually quite large and result from human frailty. These errors could be typically:

(1) An error resulting from using the wrong scale or incorrectly using the right one,

(2) An error due to transcribing incorrectly,

(3) A writing error, such as laying the point down incorrectly on the mold loft floor, or correspondingly mispunching a data card for a computer.

It is highly probable that a few points will be totally in error and will not contain any information of value. Superimposed on these and the rest of the points will be small, random errors due to reading inaccuracy.

Our first problem is then to separate the points having errors which do not at all reflect the naval architect's intention from the rest. In the mold loft this operation is done by eye. Before the loftsman proceeds with putting fair lines through the preliminary offsets, he checks them to see if they seem to fit in well with one another. Those points which seem to be in error are rechecked with the lines plan and perhaps even re-scaled. However, this is a liberty not usually afforded the computers in the modern approach; therefore, it is crucial that this process of rejection of bad points be executed if the naval architect's intentions are to be preserved in a later computation with these points (e.g., fitting a fair surface to them).

A fair curve does not have too many or too closely-spaced inflection points. One can easily verify for himself that two, closely-spaced inflection points in a curve causes it to have a "bump" in that region. Such bumps are not considered fair by a naval architect. In a similar way, three closely-spaced inflection points correspond to two "bumps." Thus, inflection points need to be relatively isolated from one another to prevent the occurrence of "bumps."

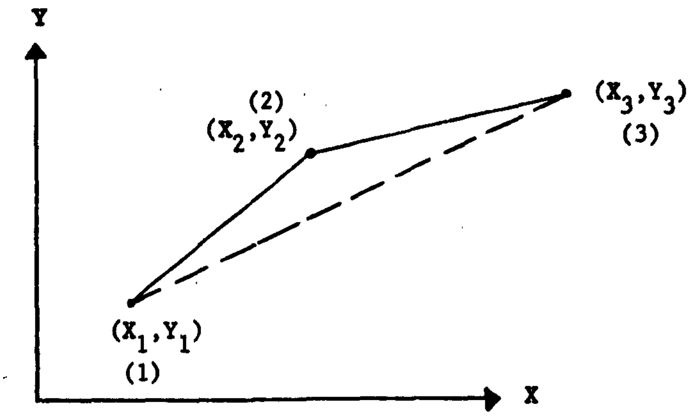


Fig. 2 Interpretation of Second Differences

Consider the set of three points shown in Fig. 2. Suppose any continuous curve which has continuous first and second derivatives passes through these points. Since, as shown, the middle point (2) lies above the line connecting the two other points, (1) and (3), there must be some portion of the curve with negative curvature. This can easily be deduced by repeated use of the Theorem of the Mean Value and the fact that the curve is required to have a continuous second derivative. One can present this result in a slightly different way. The "second difference" of the given three points is given by:

$$(1) \quad r_2 = \frac{2}{X_3-X_1} \left[\frac{Y_3-Y_2}{X_3-X_2} - \frac{Y_2-Y_1}{X_2-X_1}\right]$$

For convenience we refer to this as the second difference evaluated at point (2), even though it obviously depends on points (1) and (3) also. This second difference has the property of being negative when (2) lies above the line connecting (1) and (3), and positive when (2) lies below this line. In Fig. 2 the second difference evaluated at (2) is negative since (2) lies above the line. Finally, we arrive at the well-known result for a curve satisfying I-c.* There must be a portion of it in an interval which has curvature of the same sign as the second difference computed for that interval.

It can be seen that unlike signs of second differences at two neighboring points indicates an inflection point in the interval between these two points. We can now develop a criterion for detecting points with large errors in them. If, in any set of three consecutive points, the second difference changes signs twice we have a "bad" point. The "bad point" can now be moved an appropriate amount requiring it to be compatible with its neighboring points (i.e., not cause any "bumps").

After the points have been "smoothed" (large errors removed) a requirement that the second derivative and second difference be of the same sign at all data points will guarantee the analytic-spline to have the correct amount as well as correct spacing of inflection points.

The Coordinate Axes

The coordinate system employed is indicated in Fig. 3, below.

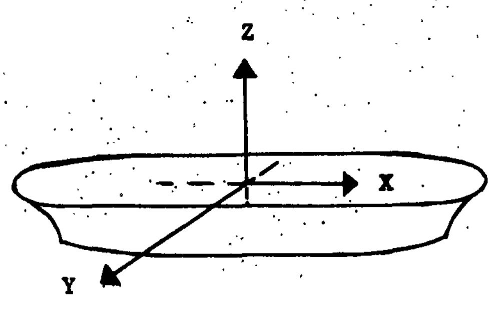


Fig. 3 The Coordinate Axes

*See Page II-2

The equation for the surface of the ship will be a function $Y = F(X,Z)$. The two-dimensional curves obtained by holding Z constant $(Y = f(X))$ will be referred to as waterlines. The two-dimensional curves obtained by holding X constant $(Y = g(Z))$ will be referred to as stations.

Notation

The notation employed offers a means of representing the analytic-spline by a single continuous equation, which has continuous first and second derivatives and a discontinuous third derivative. The discontinuities of the third derivative occur at pre-selected points. In essence, this equation consists of individual cubics between points of discontinuity of the third derivative.

The analytic spline is represented by:

$$Y(X) = A_0 + A_1X + A_2X^2 + A_3X^3 + A_4(X-a_1)^3_{\#} + \ldots + A_{N+1}(X-a_{N-1})^3_{\#} \quad * \tag{2}$$

$a_1 \ldots , a_{N-1}$ are the abscissae of the points of discontinuity of the third derivative and should be chosen to coincide with the abscissae of the original data points.

In the above representation

$$(X-a)^3_{\#} = \begin{cases} (X-a)^3 & \text{if } X > a \\ 0 & \text{if } X \leq a \end{cases}$$

Or equivalently

$$(X-a)^3_{\#} = \tfrac{1}{2}(X-a)^3 + \tfrac{1}{2}\left| (X-a)^3 \right|$$

*This notation was originally introduced by Dr. Feodor Theilheimer

From the above definition it is obvious that $(X-a)_{\#}^3$ will never be negative

The above notation is very direct.

As an example, consider the following analytic-spline

(2') $$Y(X) = -1 + 3X - 3X^2 + 2X^3 - (X-1)_{\#}^3$$

For $X \leq 1$ (2') gives the cubic

$$Y_i(X) = -1 + 3X - 3X^2 + 2X^3$$

For $X > 1$ (2') gives the cubic

$$Y_{ii}(X) = X^3$$

It can immediately be seen that Y_i and Y_{ii} have equal ordinate values, first derivatives, and second derivatives at $X = 1$, but unequal third derivatives.

B. TWO-DIMENSIONAL FAIRING

As previously stated, a requirement that the second derivative and second difference be of the same sign will guarantee compatibility of inflection points.

This constraint is expressed mathematically as:

Require that

(3) $$r_i \cdot Y''(X_i) \geqq 0 \qquad i = 1 , \ldots , N$$

Where

r_i = second difference at i^{th} data point*

$Y''(X_i)$ = second derivative at i^{th} data point

<u>Deviation Requirement</u>

The general analytic-spline $Y(X)$ is given by (2) .
The original offsets (ordinate values of original N points) are denoted by Y_i $\quad i = -1 , \ldots , N$

The maximum ordinate-deviation between curve and offset is noted by λ (we take <u>maximum</u> to mean <u>largest magnitude</u> and not maximum in an algebraic sense.)

It readily follows that

(4) $$\left| Y(X_i) - Y_i \right| \leq \lambda \qquad \text{for} \quad i = 1 , \ldots , N$$

The above expression simply says that the absolute values of the ordinate deviations at the N original points are all less than or equal to λ, (i.e., λ is the maximum deviation).

λ has a distinct value for each specific analytic-spline $Y(X)$.

We now ask the linear programming code to give us the analytic-spline which satisfies fairness constraint (3) and has the smallest value for λ (i.e., minimum-maximum deviation).

*Since the second difference is not defined for the first and last points, the second difference at the second and next to last points will be used for these two points respectively.

Formulation of Deviation and Fairness Constraints as Linear Programming Problem

The curve fitting problem can now be formulated as a linear programming problem as follows:

Minimize λ subject to the constraints

$$\left.\begin{array}{lr} (3) & r_i \cdot Y''(X_i) \geq 0 \\ (4) & |Y(X_i) - Y_i| \leq \lambda \end{array}\right\} \quad i = 1, \ldots, N$$

Since linear programming codes handle linear constraints only, constraint (4) is equivalently expressed by the following pair of linear inequalities:

$$\left.\begin{array}{l} \lambda - Y(X_i) \geq -Y_i \\ \lambda + Y(X_i) \geq Y_i \end{array}\right\} \quad i = 1, \ldots, N$$

There are now three linear constraints per point the analytic-spline must satisfy. This problem readily lends itself to linear programming routines. The problem is stated as:

Minimize λ subject to the following linear constraints:

$$\left.\begin{array}{l} \lambda - Y(X_i) \geq -Y \\ \lambda + Y(X_i) \geq Y_i \\ r_i \cdot Y''(X_i) \geq 0 \end{array}\right\} \quad i = 1, \ldots, N$$

This formulation requires N+3 non-slack variables and 3N constraints, (N is the number of original data points).

Actual Presentation

The actual presentation of the problem requires only slight modification. The inequalities are transformed to equalities by the subtraction of a positive quantity (called slack variable) from each inequality. Since present linear programming codes accept positive variables only, the variables (coefficients in equation (2)) are represented by the difference of two non-negative variables, i.e.,

$$A_i = A'_i - A''_i$$

$$A'_i,\ A''_i \geq 0$$

The formulation of the linear programming problem now becomes:

Minimize λ subject to the following constraints:

$$\lambda - (A'_0 - A''_0) - (A'_1 - A''_1)X_i - (A'_2 - A''_2)X_i^2 - (A'_3 - A''_3)X_i^3 - \ldots - (A'_{N+1} - A''_{N+1})(X_i - a_{N-1})^3_{\#} - S_i = -Y_i$$

$$\lambda + (A'_0 - A''_0) + (A'_1 - A''_1)X_i + (A'_2 - A''_2)X_i^2 + (A'_3 - A''_3)X_i^3 + \ldots + (A''_{N+1} - A''_{N+1})(X_i - a_{N-1})^3_{\#} - P_i = Y_i$$

$$r_i \cdot \left[2(A'_2 - A''_2) + 6(A'_3 - A''_3)X_i + \ldots + 6(A'_{N+1} - A''_{N+1})(X_i - a_{N-1})_{\#}\right] - Q_i = 0$$

$$i = 1, \ldots, N$$

---

*This difficulty can be eliminated, as is shown on Page III-23, Section entitled "Dual Formulation."

In this way an equation of the form:

$$Y(X) = A_0 + A_1X + A_2X^2 + A_3X^3 = A_4(X-a_1)^3_{\#} + \ldots + A_{N+1}\ (X-O_{N-1})^3_{\#}$$

will be constructed.

The maximum deviation between curve and data points will be given by the value of λ in the final solution.

C. THREE-DIMENSIONAL FAIRING

Extension of Notation to Three-Dimensions

In attempting to fit a surface to a three-dimensional set of points, the same procedure as was used in the two-dimensional case is employed.

Equation (2) is replaced by its three-dimensional extension $Y(X,Z)$:

$$Y(X,Z) = C_0(X) + C_1(X)Z + C_2(X)Z^2 + C_3(X)Z^3 + C_4(X)(Z-b_1)^3_{\#} + \dots + C_{M+1}(X)(Z-b_{M-1})^3_{\#} \tag{5}$$

Where

$$C_i(X) = c_{i,o} + c_{i,1}X + c_{i,2}X^2 + c_{i,3}X^3 + \dots + c_{i,N+1}(X-a_{N-1})^3_{\#}$$

$$i = 0 , \dots , M+1$$

a_p = X-Coordinate of P^{th} station

b_q = Z-Coordinate of Q^{th} waterline

(5) represents a surface of N stations and M waterlines.

An expansion of Equation (5) gives:

$$
\begin{aligned}
Y(X,Z) = &\left[c_{00} + c_{01}X + c_{02}X^2 + c_{03}X^3 + \ldots + c_{0,N+1}\ (X-a_{N-1})^3_{\#}\right] \\
+ &\left[c_{10} + c_{11}X + c_{12}X^2 + c_{13}X^3 + \ldots + c_{1,N+1}\ (X-a_{N-1})^3_{\#}\right] Z \\
+ &\left[c_{20} + c_{21}X + c_{22}X^2 + c_{23}X^3 + \ldots + c_{2,N+1}\ (X-a_{N-1})^3_{\#}\right] Z^2 \\
+ &\left[c_{30} + c_{31}X + c_{32}X^2 + c_{33}X^3 + \ldots + c_{3,N+1}\ (X-a_{N-1})^3_{\#}\right] Z^3 \\
+ &\left[c_{40} + c_{41}X + c_{42}X^2 + c_{43}X^3 + \ldots + c_{4,N+1}\ (X-a_{N-1})^3_{\#}\right] (Z-b_1)^3_{\#} \\
&\ \ \vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\qquad\qquad \vdots \qquad \vdots \\
+ &\left[c_{M+1,0} + c_{M+1,1}X + c_{M+1,2}X^2 + c_{M+1,3}X^3\right. \\
&\qquad\qquad\qquad \left. + \ldots + c_{M+1,N+1}(X-a_{N-1})^3_{\#}\right] (Z-b_{M-1})^3_{\#}
\end{aligned}
$$

We now have an expression that consists of individual cubics in the coordinate planes between any two stations or waterlines. This surface equation will have (N+2) (M+2) coefficients for N stations and M waterlines.

Requirements

The deviation is again required to be less than or equal to λ

$$\left| Y(X_i Z_j) - Y_{ij} \right| \leq \lambda \qquad \begin{array}{l} i = 1, \ldots, N \\ j = 1, \ldots, M \end{array}$$

Two second difference constraints are now required in order to guarantee fairness: one in the X-direction and one in the Z-direction.

FORMULATION AS LINEAR PROGRAMMING PROBLEM

The problem is now formulated as a linear programming problem as follows:

Minimize λ subject to the following constraints:

$$\left.\begin{array}{lr} \text{(b)} & \lambda - Y(X_i, Z_j) \geq -Y_{i,j} \\ \text{(bb)} & \lambda + Y(X_i, Z_j) \geq Y_{i,j} \\ \text{(bbb)} & \hat{r}_{i,j} \cdot Y''(X_i, Z_j)zz \geq 0 \\ \text{(bbbb)} & \hat{\hat{r}}_{i,j} \cdot Y''(X_i, Z_j)xx \geq 0 \end{array}\right\} \begin{array}{l} i = 1, \ldots, N \\ j = 1, \ldots, M \end{array}$$

$\hat{r}_{i,j}$ = second difference in Z-direction at j^{th} point on i^{th} station

$\hat{\hat{r}}_{i,j}$ = second difference in X-direction at i^{th} point on j^{th} waterline

$Y''(X_i, Z_j)_{zz}$ indicates the second derivative of $Y(X,Z)$ with respect to Z (X held constant) at j^{th} point on i^{th} station.

$Y''(X_i, Z_j)_{xx}$ indicates the second derivative of $Y(X,Z)$ with respect to X (Z held constant) at i^{th} point on j^{th} waterline.

This formulation requires 4R constraints
(R = number of original points.)

The maximum deviation will, as in the two-dimensional case, be given by the value of λ in the final solution.

D. DOUBLE SPLINE

On Page III-7 the analytic spline is represented by

$$\text{(i)} \qquad Y(X) = A0 + A_1X + A_2X^2 + A_3X^3 + A_4(X-a_1)^3 + \ldots + A_{N+1}(X-a_{N-1})^3$$

where $a_1, \ldots, a_{N-1}$ are the abscissae of the original input points. With the use of the dual formulation (Page III-23) the number of coefficients in Equation (1) becomes the number of deviation constraints in the dual problem. Since the size of this number is a very critical factor in the length of time it takes to solve the LP problem, we may ask the question, "What effect will the elimination of some of these coefficients have to the final solution." In other words, can we affort to reduce the computer time at the expense of possibly inducing a larger deviation between curve and offsets.

One way of systematically reducing the number of coefficients would be to add an $(X-a_i)_{\#}^3$ term only at every second data point, (i.e., if $a_1, \ldots, a_n$ are the original dat points, add $(X-a_i)_{\#}^3$ terms if, and only if i = even integer. Employing this procedure we would add terms only at $a_2, a_4, \ldots, a_{2i}$.

Motivation for this procedure can be extracted from various policies the graphical or manual loft uses. Clearly, the graphical loft does not put a weight, or "duck," at every point and generally agrees the curve with the fewest "ducks" is the most desirable, provided of course that the deviation is reasonable.

It was previously shown that each coefficient A_j for $j \geq 4$ in Equation (i) corresponds to a point where a duck would be used in the graphical loft. By eliminating every second coefficient we have in effect eliminated every second duck.

In practice, it has been shown that the elimination of these coefficients not only significantly reduces the computing time, but also does not reduce the quality of the curve. In many instances the quality of the curve was increased. A reasonable explanation of this could be that this procedure offers fewer places to have inflection points or zero second derivatives (reverse curvature or flat splots). This procedure of joining three points with a single cubic is referred to as "double splining."

If we were to "double spline," Equation (i) would be replaced with its counterpart:

$$\text{(ii)} \qquad Y(X) = A_0 + A_1X + A_2X^2 + A_3X^3 + A_4(X-a_1)_{\#}^{\,3} + A_6(X-a_4)_{\#}^{\,3} + \dots$$

Where A_1 , ... , a_n are the original input points. The only change to the original formulation, aside from the use of this different curve will be that a second derivative requirement will only be needed at the points where an $(X-a_1)_{\#}^{\,3}$ term is added. Second derivative requirements at the other points would be redundant. As before, a deviation constraint is required at all original points.

E. ALTERNATE REPRESENTATION - SUM OF THE DEVIATIONS

The use of linear programming as a curve fairing tool has suggested an alternate fit to the one just presented. This is the "sum of the deviations" problem. In this formulation, the sum of the absolute values of the deviations between the surface function and the original data are minimized (i.e., minimize $\sum_{i=1}^{n} |Y(X_i)-Y_i|$) This is made possible by that characteristic of linear programming which requires all variables to be non-negative.

Assume a relationship between the surface equation and the set of original offsets such that the value of the equation at each of the original offsets is less than, or equal to the original offset. That is,

$$Y(X_{i,j}) \leq Y_{i,j} \qquad \begin{array}{l} i = 1 , \ldots , N \\ j = 1 , \ldots , M \end{array}$$

We can change this inequality to equality by adding a dummy variable as follows (note that the dummy variable must be non-negative):

$$Y(X_{i,j}) + S_{i,j} = Y_{i,j} \qquad \begin{array}{l} i = 1 , \ldots , N \\ j = 1 , \ldots , M \end{array}$$

$$S_{i,j} \geq 0$$

In linear programming, inequalities are always changed in this manner to obtain a full set of simultaneous linear equations. The additional variables ($S_{i,j}$) are called slack variables. As previously stated, in any feasible solution in linear programming, all of the variables in the solution must be non-negative. Therefore in any optimal, feasible solution, all of the slacks which remain in the solution will either be at a zero level or positive.. By requiring a very high cost on the slacks when compared to that on the coefficients, the sum of the slacks (positive deviations) is minimized. This fairing problem is formulated as follows:

$$\left.\begin{array}{l} \text{Minimize} \sum S_{ij} \text{ subject to the following} \\ \qquad Y(X_{i,j}) + S_{i,j} = Y_{i,j} \\ \qquad \hat{r}_{i.j} \cdot y''(X_{i,j}) \geq 0 \\ \qquad \hat{\hat{r}}_{i,j} \cdot Y''(X_{i,j})_{xx} \geq 0 \end{array}\right\} \begin{array}{l} i = 1 , \ldots , N \\ j = 1 , \ldots , M \end{array}$$

The deviation can be obtained at each offset by reading the value of the slack at that point. If the slack does not enter into the final solution, or enters at a zero level, the surface passes exactly through the offset.

In practice the set of deviation slacks, plus the set of slacks for changing the curvature constraints to equalities, which together form an identity matrix, are used as the first basic feasible solution allowing the optimization phase to begin at once.

A modification of this method has been made utilizing the following formulation (this is done so the deviations will be unrestricted in sign, which in turn should give a closer fit):

$$\left.\begin{array}{l} \text{Minimize} \sum S_{i,j} + Q_{i,j} \text{ subject to the following} \\ \qquad Y(X_{i,j}) + S_{i,j} - Q_{i,j} = Y_{i,j} \\ \qquad \hat{r}_{i,j} \cdot Y''(X_{i,j})_{ff} \geq 0 \\ \qquad \hat{\hat{r}}_{i,j} \cdot Y''(X_{i,j})_{xx} \geq 0 \end{array}\right\} \begin{array}{l} i = 1 , \ldots N \\ j = 1 , \ldots M \end{array}$$

Here the slacks $S_{i,j}$ and $Q_{i,j}$ are both positive, however, only one of the pair may enter into the solution since they are linearly dependent,* (i.e., will cause a singular matrix)

This formulation allows the surface to pass over some offsets and under others. The first basic feasible solution again consists of the set of slacks $S_{i,j}$ plus those of the curvature constraints.

* $S_{i,j}$ = Positive deviation

$-Q_{i,j}$ = Negative deviation

$S_{i,j}-Q_{i,j}$ = Total deviation, as stated above, in practice, only one of $S_{i,j}$ or $Q_{i,j}$ will enter into the solution.

F. DUAL FORMULATION

It can be shown* that if we write the primal problem in the form:

$$\sum_{j=1}^{r} d_{i,j} \cdot x_j = d_i \qquad i = 1, \ldots, k$$

$$\sum_{j=1}^{r} d_{i,j} \cdot x_j \geq d_i \qquad i = k+1, \ldots, s,$$

$$x_j \geq 0 \quad j = 1, \ldots, r \qquad \text{MAX } Z = cx$$

then the dual problem is

$$D'w \geq c', \ \text{MIN } Z = d'w$$

$$w_i \ \text{unrestricted} \quad i = 1, \ldots, k$$

$$w_i \geq 0 \qquad i = k+1, \ldots, s$$

That is, wherever i^{th} constraint in the primal is an equality, the i^{th} dual variable is unrestricted with regard to sign. This property has considerable importance in our problem, because the coefficients of the surface equation are, by nature, unrestricted. The dual formulation offers a way of eliminating the representation of each coefficient as the difference of two non-negative variables, and this greatly reduces the size of the problem

*Refer to G. Hadley, Linear Programming, p 236 (Addison-Wesley Publishing Co., 1962

We therefore restate the problem as follows:

Minimize λ subject to the following constraints

$$
\left.
\begin{array}{rcl}
\lambda - Y(X_{i,j}) & = & -y_{i,j} \\
\lambda + Y(X_{i,j}) & = & y_{i,j} \\
\widehat{r}_{i,j} \cdot Y''(X_{i,j})_{ff} & \geq & 0 \\
\hat{r}_{i,j} \cdot Y''(X_{i,j})_{xx} & \geq & 0
\end{array}
\right\}
\begin{array}{l}
i = 1 , \ldots , N \\
j = 1 , \ldots , M
\end{array}
$$

If we present this formulation to the linear program in the dual form, the variables representing the deviation constraints (coefficients of the surface equation) will be unrestricted with regard to sign.

Section IV

FORMULATION OF END CONDITIONS

The previous formulation of the surface fitting problem can handle only rectangular arrays of points, that is, every waterline must have the same number of stations; and conversely, every station must have the same number of waterlines. This condition, unfortunately, does not exist when we consider the bow and stern of a ship. The lower waterlines may have fewer stations than the higher ones. The bow and stern areas also produce other problems which we shall refer to collectively as "end conditions."

If, in the equation for the surface of the ship (i.e., $Y = f(X,Z)$), we replace Y with zero we obtain the expression for the profile of the ship (i.e., $f(x,z) = 0$).

This profile is specified in the original lines drawings of the ship and at times may be very complicated. This profile must also be "fair." The present formulation offers no means for guaranteeing this. In attempting to solve this problem we can be guided by the procedure used in the manual loft.

In manual lofting the profile is "faired" as an independent unit of the ship. The ship is then required to pass through this previously "faired" profile. In this way the ship is guaranteed to have the correct prrofile, no matter how complicated it may be.

We shall attempt to construct an analogous procedure in the mathematical loft. To begin with, let us show that it is not sufficient to require that the original waterlines pass through the profile, without any additional restrictions on the surface between the original waterlines.

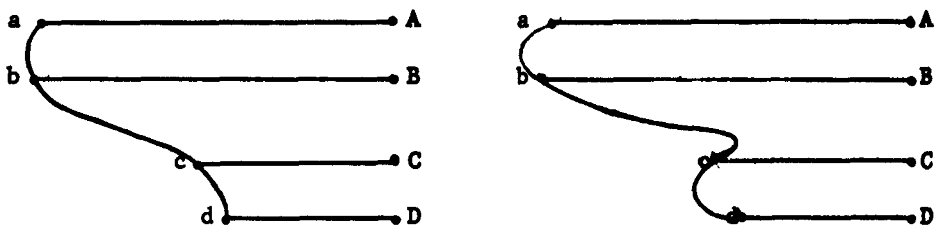


Fig. 1-a

Fig. 1-b

Fig. 1

In Fig. 1, A, B, C, and D are waterlines passing through the profile at Points a, b, c, and d, respectively. The contour of the profile desired i shown in Fig. 1-a; but in the surface fitting problem, a requirement that Waterlines A, B, C, and D pass through Points a, b, c, and d, respectively, without any additional requirements on the profile of the surface, may very possibly result in the Profile shown in Fig. 1-b. At any rate, we cannot guarantee that this will not occur.

Consider the expression for the surface of the ship:

$$Y = f(X,Z)$$

We wish to transform this expression so that Y will be equal to zero at the profile of the ship. Toward this end we construct

$$Y = f(X,Z) \cdot T(X,Z)$$

Where $T(X,Z)$ has the property that it is equal to zero if the point (X,Z) in the $Y = 0$ plane lies on the profile.

Let $$T(X,Z) = \left[1 - \left[1+f(Z)-X\right]^{3}_{\#}\right]$$

$f(Z)$ is the "faired" spline curve representation of the profile. (Clearly, since $Y = 0$ at the profile, the profile will be an expression in X and z only). Since $X = f(z)$ is the expression for the profile, for a given waterline - say $Z = Z_0$, $f(Z_0)$ will be the X value corresponding to the intersection of the profile with this waterline.

When $X = f(Z_0)$, i.e., when Point (X,Z_0) lies on the profile

$$T(X,Z_0) = \left[1- \left[1 + f(Z_0) - f(Z_0)\right]^3_\# \right]_\# = 1 - 1 = 0$$

so
$$Y = f(X,Z_0)\ T(X,Z_0) = 0$$

But, if $X \geq f(Z_0) + 1$ (i.e., X is one unit away or farther from the profile)

$$T(X,Z_0) = \left[1 - \left[1 + f(Z_0) - f(Z_0) - 1\right]^3_\#\right]_\# = 1-A_\#$$

Where $A_\# \leq 0$

here we define $A_\#$ such that

$$A_\# = \begin{cases} 0 & \text{if } A_\# \leq 0 \\ A & \text{if } A_\# > 0 \end{cases}$$

So

$T(X,Z_0) = 1 - A_\#$ with $A_\# \leq 0$, becomes

$T(X,Z_0) = 1 - 0 = 1$ for $X \geq f(Z_0) + 1$

and

$$Y(X,Z) = f(X,Z)\ T(X,Z) = f(X,Z)$$

This implies that $T(X,Z)$ is effective only between the profile and one unit away from it, also

$$0 \leq T(X,Z) \leq 1 \quad \text{for all } X,Z$$

We raise $\left[1 + f(Z) - x\right]$ to the third power so we may have continuous first and second derivatives as well as continuous curve at $X = f(Z) + 1$.

If we consider stations instead of waterlines, we see that similar results will be obtained.

The expression for the surface now becomes

$$Y(X,Z) = f(X,Z) \left[1 - \left[1 + f(Z) - X\right]^3\right]$$

$f(Z)$ = previously faired profile.

If it is desired to have infinite slope where the waterline (station) intersects this profile, we merely replace

$$T(X,Z) = \left[1 - \left[1 + f(Z) - X\right]^3_{\#}\right]_X$$

with

$$T(X,Z) = \left[1 - \left[1 + f(Z) - X\right]^3_{\#}\right]^P_{\#}$$

where $0 < P < 1$

The interval in which $T(X,Z)$ is effective does not have to be from the profile to the profile plus 1 unit, but can be from the profile to the profile plus any arbitrary distance, D , by using

$$T(X,Z) = \left[1 - \left[\frac{D + f(Z) - X}{D}\right]^3_{\#}\right]^P_{\#}$$

For $X = f(Z) + D$

$$T(X,Z) = \left[1 - \left[\frac{D + fZ - f(Z) - D}{D}\right]^3_{\#}\right]^P_{\#}$$

$$T(X,Z) = \left[1\right]^P = 1$$

It can very easily be verified that

$$T(X,Z) = 1 \quad \text{if} \quad X \geq f(Z) + D$$

or

$$Y(X,Z) = f(X,Z)$$

$f(X,Z)$ is the spline-surface representation developed in a previous section. $Y(X,Z)$ is now guaranteed to have the correct end profile and becomes a regular spline curve D units away from the profile.

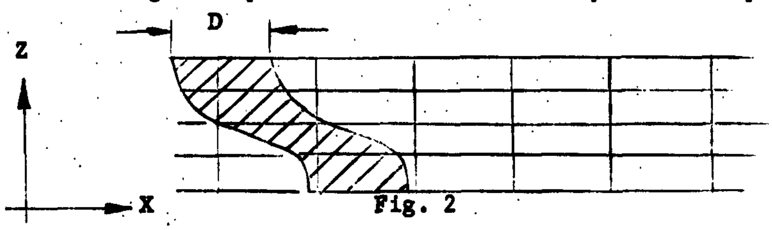


Fig. 2

In Fig. 2, the cross-hatched area represents the portion of the surface in which $T(X,Z)$ is effective.

Section V

LOFT INFORMATION

From the mathematical fairing program has come a surface equation which describes the desired hull form. It now becomes necessary to extract from this surface equation the data required for parts definition.

THE FINAL LINES DRAWING

One of the first and more important types of information required from a loft is, of course, a final body plan. The data for the preparation of such body plan is easily obtained from the mathematical ship definition. Substituting any specific frame location X into the hull surface equation $Y = f(X,Z)$, and solving this expression for various values of Z, will provide offsets Y describing the frame of interest. The values for the coordinates of these points may be obtained at any desired scale. If an automatic plotting machine is available, such as the Coordinatograph, the body plan may be scribed quickly and with great accuracy.

Other information for the lines drawing may be obtained in the same manner. Waterlines may be described by substituting a specific value of Z, into the surface equation $Y = f(X,Z)$, and solving the equation at various values of X.

The coordinates of the buttock points are not obtained in such straight-forward fashion. However, using a computer, a technique such as Newton's method may be employed to intersect the hull surface with a plane parallel to the center plane of the ship. Heights of the intersection of the plane and the surface may be obtained at many points along the length of the ship. Diagonal planes can be handled in the same way, intersecting the surface with a plane at a constant angle to the hull surface.

SHELL PLATE DEVELOPMENT

To develop the flat pattern to which a plate must be cut in order to fit properly in place on the hull when formed, requires several types of information:

a. Arc lengths along intersections of coordinate planes (station or waterline planes) and the hull surface
b. Coordinates of points along the sight edges on the hull surface
c. Arc length along the sight edge on the surface
d. True angles on the surface between coordinate planes

The mathematics for developing this information is given below.

a. Finding arc lengths on the hull surface along station or waterline planes:

The general formula for arc length along a station curve, X = constant is

$$6)\qquad S = \int_{Z_o}^{Z_1} \sqrt{1 + 1 + f_Z^2}\, dZ$$

When X is constant the hull surface equation becomes

$$7)\qquad Y = A + BZ + CZ^2 + DZ^3 + \ldots$$

Substituting 7) into 6) gives

$$8)\qquad S = \int_{Z_o}^{Z_1} \left[1 + (B + 2CZ + 3DZ^2 + \dots)^2 \right]^{\frac{1}{2}} dZ$$

which can be solved for arc length between any two heights Z .

Similarly, the equation for arc length along the surface in a waterline plane is

$$9)\qquad S = \int_{Z_o}^{X_1} \left[1 + (B + 2C + 3DX^2 + \dots)^2 \right]^{\frac{1}{2}} dX$$

b. Coordinates of any point along sight edges:

To establish the plate sight edges mathematically from the hull surface equation requires a function $Z = f(X)$ which describes the trace of the sight edge on the $X - Z$ plane. Assume this function is known for the moment.

The equation describing the hull surface is of the form

10) $Y = f(X,Z)$

Substituting $Z = f(X)$ into (5)

11) $Y = f(X)$

Given any value of X on $Z = f(X)$, Z and Y are determined by equations 6) and 8)

The method for obtaining the function $Z = f(X)$ is to use the preliminary offsets given by the naval architect. These consist of the height above the baseline for each sight edge at each station. The set of heights $(Z_1 , \ldots , Z_m)$ for each sight edge describes the trace of the sight edge on the centerline plane of the ship.

To establish $Z = f(X)$ two steps are necessary:

1. Since the heights $(Z_1 , \ldots , Z_m)$ are given only to the accuracy obtained from scaling the drawings, they must be smoothed to prevent unwanted inflections.

2. The smoothed points can then be fitted with a spline curve using the smaller two-dimensional linear programming formulation.

It may be possible to assume that in the equation $Z = f(X)$, Z is a simple function of X such as a parabola. This would require considerable analysis and perhaps revision of the naval architect's method of establishing sight edges.

c. Arc lengths along the sight edges on the surface:

The sight edge has now been defined parametrically

12)
$$X = X$$
$$Z = f(X)$$
$$Y = f(X)$$

Substitution into the general formula for arc length and integrating along the X axis gives

$$13) \qquad S = \int_{X_o}^{X_1} \left[1 + (f'_Y (X))^2 + f'_Z (X))^2 \right]^{\frac{1}{2}} dX$$

d. The true angle between coordinate planes on the hull surface:

Substitution of the directional cosines of the coordinate planes into the law of cosines gives the angle between these planes as:

$$14) \qquad \cos 0 = \frac{f_X f_Z}{\left[(1 + f_X^2)\ (1 + f_Z^2) \right]^{\frac{1}{2}}}$$

LONGITUDINALS

Many times the position of longitudinals around the hull of the ship is given by the arc length distance from the keel or other longitudinal.

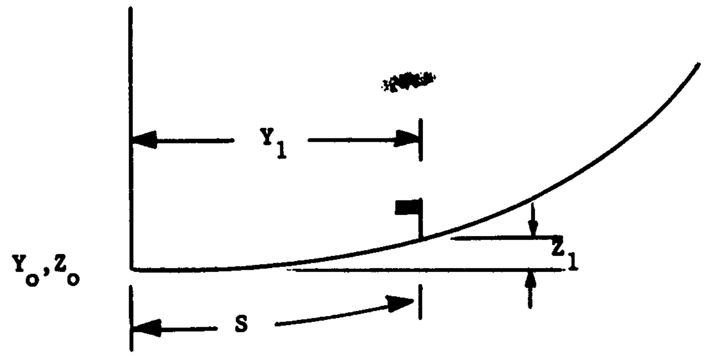


Fig. 4 Establishing the Position of Longitudinals

That is, (Fig. 4) given S and Y_o , Z_o (which may be the keel line) find Y_1 and Z_1 which are the coordinates of the position of the longitudinal.

The height Z_1 may be found by a trial and error solution of Equation 3) after substituting Z_o and S . Y_1 can then be found by substituting the station location X , and Z_1 into the surface equation.

LONGITUDINALS

Many times the position of longitudinals around the hull of the ship is given by the arc length distance from the keel or other longitudinal.

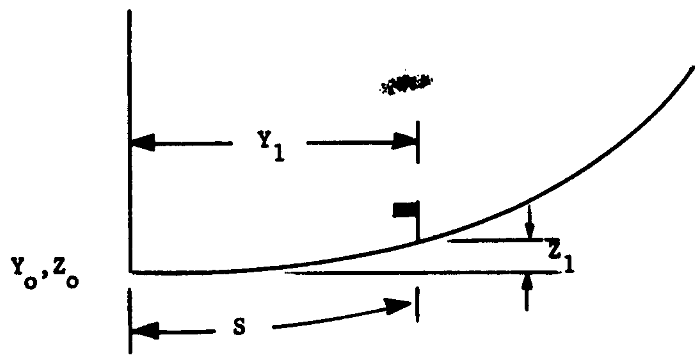


Fig. 4 Establishing the Position of Longitudinals

That is, (Fig. 4) given S and Y_o , Z_o (which may be the keel line) find Y_1 and Z_1 which are the coordinates of the position of the longitudinal.

The height Z_1 may be found by a trial and error solution of Equation 3) after substituting Z_o and S . Y_1 can then be found by substituting the station location X , and Z_1 into the surface equation.

MATHEMATICAL SHIP LOFTING

PART 2. OPERATING MANUAL

Technical Report 1.0.0-2
Volume 2 of Final Report
Contract NObs-4427

March 1963

TODD SHIPYARDS CORPORATION
LOS ANGELES DIVISION SAN PEDRO CALIFORNIA

FORWARD

This publication is Volume 2 of six volumes of the final report on Mathematical Ship Lofting and Numerical Control of Shipyard Fabricating Equipment. The work reported on herein was performed under Bureau of Ships Contract NObs-4427, Code 770, during the period from April 1961 to March 1963.

The volumes of this final report have the following titles:

Vol. 1. Project Summary Report (Technical Report 9.0.0)

Vol. 2. Mathematical Ship Lofting -
Part 1. - Theory (Technical Report 1.0.0-1)
Part 2. - Operating Manual (Technical Report 1.0.0-2)

Vol. 3. Mathematical Ship Lofting - Summary Report (Technical Report 1.5.0)

Vol. 4. Programming System for Numerically Controlled Flame Cutting of Ships Parts - Operating Manual (Technical Report 5.0.0)

Vol. 5 Development and Testing of Programming System for Numerically Controlled Flame Cutting of Ships' Parts - Summary Report (Technical Report 5.5.0)

Vol. 6. Numerically Controlled Shipyard Fabricating Equipment - Summary Report (Technical Report 3.0.0)

The work was accomplished by the Research and Development Group of the Los Angeles Division of Todd Shipyards Corporation, San Pedro, Calif. Mr. Thomas G. Smith was Project Manager for the work, and Dr. Henry A. Schade of the University of California, Berkeley, was Principal Consultant.

This report contains the results of work accomplished by the Mathematics Team which consists of:

D. A. Atkins, Naval Architect - Mathematician (Team Supervisor)

R. A. Tapia, Mathematician

J. C. Fassino, Naval Architect - Mathematician

T. A. Dunaway, Computer Programmer

S. A. Berger, Mathematician

W. C. Webster, Naval Architect - Mathematician

Janice H. Thompson, Secretary

Others of the Todd Staff, whose contributions and participation during the Project were responsible for its successful culmination, consisted of:

- Eric Chein, Data Processing Supervisor
- R. W. Feeny, Numerical Control Engineer
- H. S. Janssen, European Representative
- K. M. Jones, Naval Architect
- W. R. Lauper, Mechanical Engineer
- M. R. Ward, Jr., Naval Architect

The Project is indebted to D. A. Anderson and Milton Drandell, both of IBM Corporation, and M. L. Juncosa and Philip Wolfe of Rand Corporation, who contributed to the effort.

To the following representatives of the Navy Department go our sincere thanks for their assistance and guidance during the work:

Bureau of Ships:	David W. Taylor Model Basin:
Captain W. S. Dawson	Dr. Feodor Theilheimer
Captain F. E. Gorman	Dr. P. C. Pien
Comdr. William Harrison	Mr. William Starkweather
Mr. T. H. Sarchin	
Mr. G. Vidlak	

For their support and continued interest, our thanks to Mr. W. Taylor Potter and Mr. Karl Fixman of the U. S. Maritime Administration.

To Admiral R. K. James, Captain D. L. Carroll Jr., Captain T. J. Banvard, and Mr. Ed Kemp of the Bureau of Ships, are due the gratitude of the industry for initiating this work. The Project is particularly grateful for their early assistance in getting the work under way. To Mr. Ed Kemp go our sincerest thanks for his deep interest in this work and for his continued guidance to the Project.

CONTENTS

Section I

INTRODUCTION

The Mathematical Lofting System begins with offsets and data from a naval architect's preliminary lines plan. This data is first checked to be sure it contains no extraneous errors. It is next faired by fitting it with a surface equation which represents the molded surface of the hull and which has certain properties consistent with "fairness." Then data necessary for building the actual ship is extracted from this equation.

Sections II through IX describe the operation of this System. The instructions for the computer programs and program listings are contained in the Appendixes.

The coordinate system used throughout is shown in Fig. I-1. All dimensions used in this system are in feet unless expressly stated otherwise.

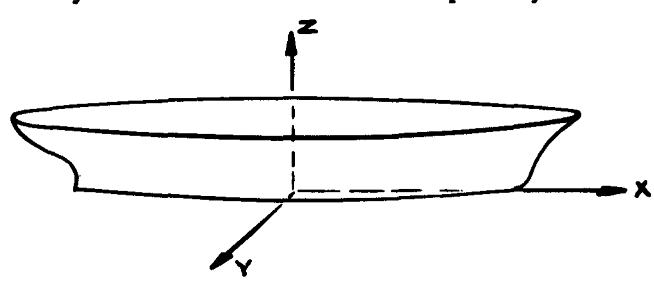


Fig. I-1 Coordinate System

A diagramatic representation of the Lofting System is shown in Fig. I-2. It is expected that anyone using this System will be familiar with the SPS and FORTRAN programming languages for the IBM-1620 computer. Although the programs included in this report represent a lofting system, it must be remembered the purpose of the programs was primarily to checkout the development work on the mathematical lofting method.

The organization of this report into Steps follows the numbering of the boxes in Fig. I-2. The flow of data through the System also follows this same order.

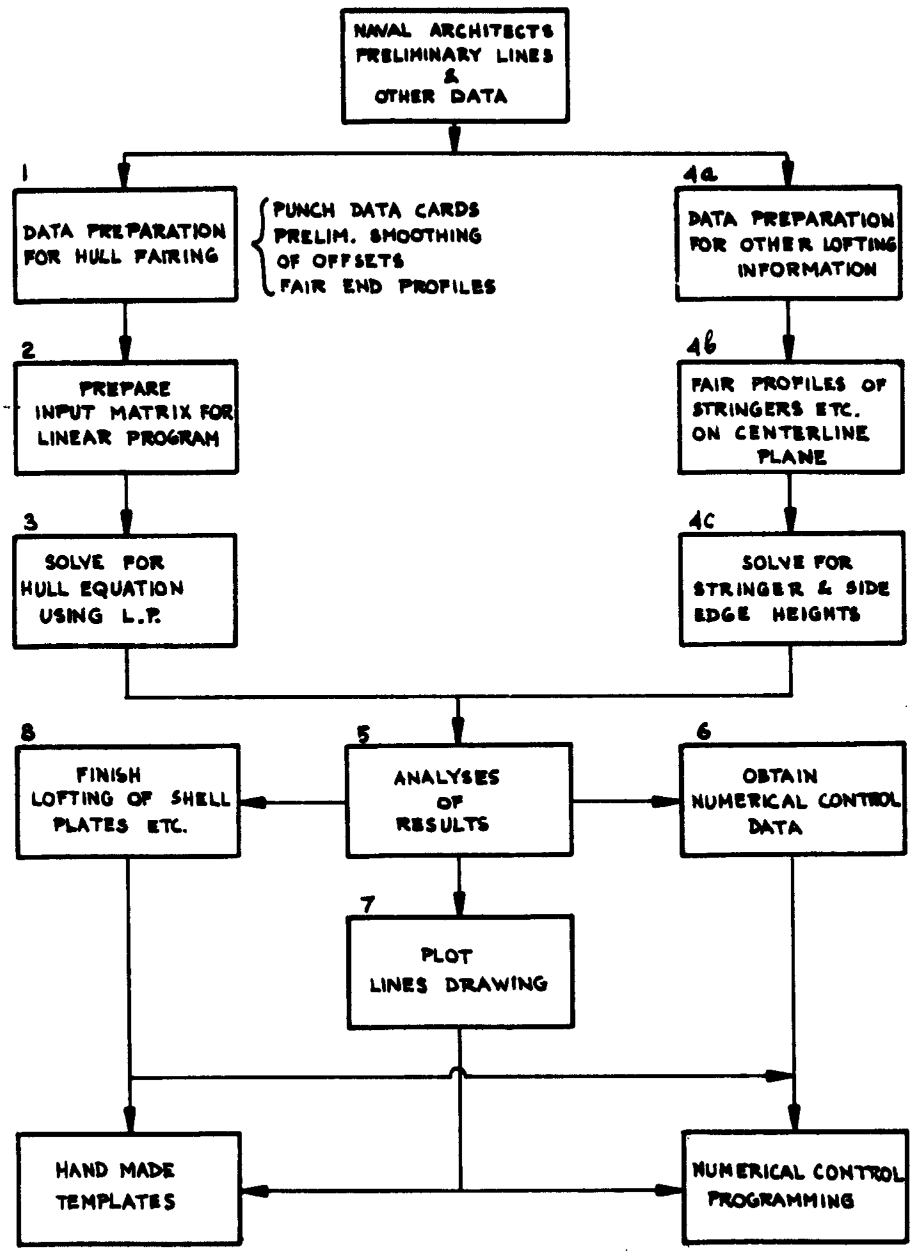


Fig. I-2 Mathematical Lofting System

Section II

PREPARATION OF INPUT DATA (STEP 1)

The flow of data through the preparation stage is shown on Fig. II-1.

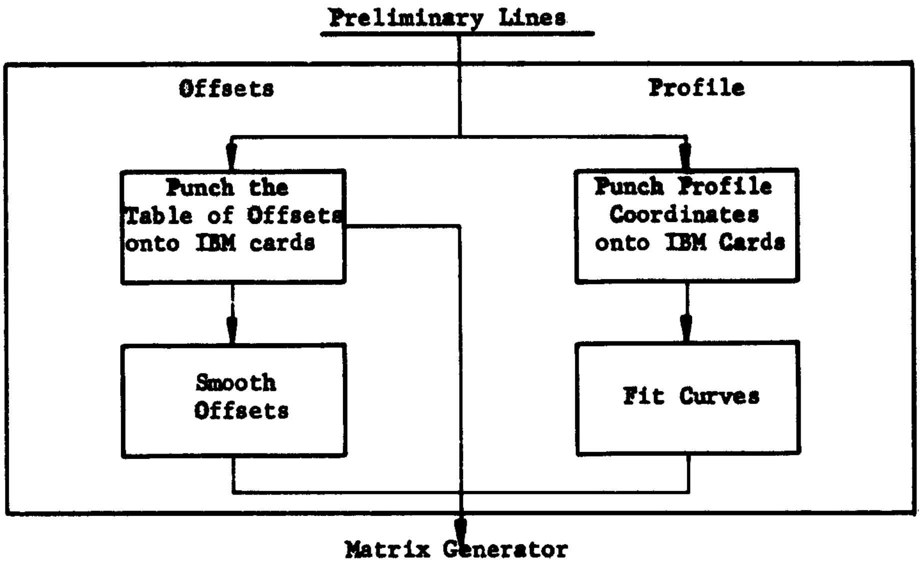


Fig. II-1 Flow Diagram for Data Preparation

A. OFFSETS

The left-hand branch of Fig. II-1 outlines the procedure for preparing the table of offsets for a ship to be lofted. This somewhat expanded table of offsets must include values for every intersection of station and waterline shown in the circumscribing rectangle of Fig. II-2. Although the values of the offsets in the shaded portions are not important and usually are made zero, they are included for

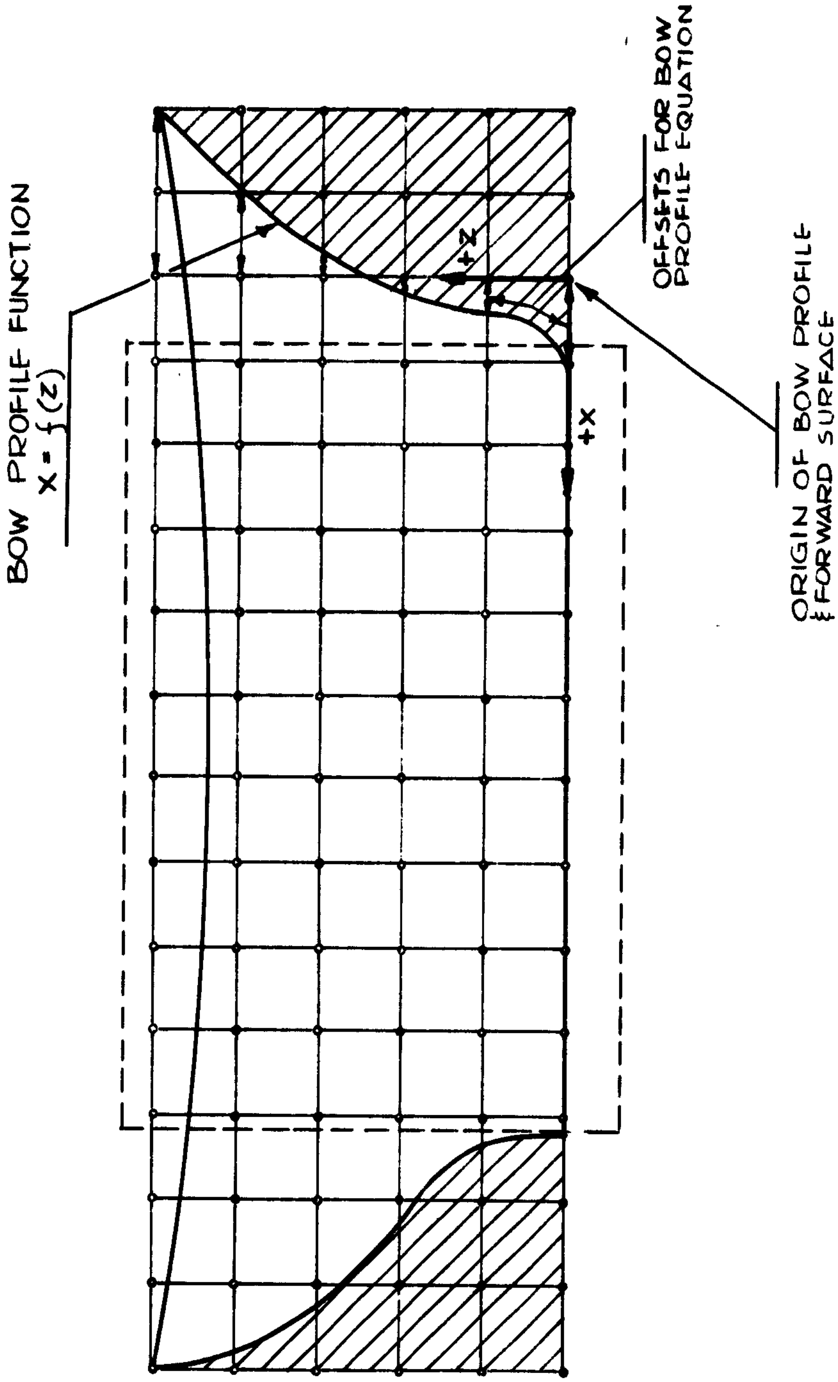

BOW PROFILE FUNCTION
X = f(z)
OFFSETS FOR BOW
PROFILE EQUATION
+Z
+X
ORIGIN OF BOW PROFILE
& FORWARD SURFACE

Fig. II-2

mechanical reasons. The offsets above the deck, or knuckle line, must be consistent with the others and faired to a degree such that the second differences calculated using these points will have the proper sign.

The offsets are punched into cards, along with their X and Z locations on the hull. These cards are read by an IBM-1620 computer program which converts them from feet and fractions to feet and decimals, and "smooths" them.* The principle is a necessary part of the lofting system. This procedure essentially points out any offsets which are obviously in error, and substitutes a value which will allow a fair curve to come near it, without causing discrepancies at other points (corrects the second differences). Presently, only the offsets within the dotted rectangle of Fig. II-2 may be smoothed by this routine, since it has not been programmed to accept the end profiles. The remaining points can be examined as noted in Step 2-A. After gaining experience in the fairing procedure, the smoothing program can often be bypassed and the data fed directly into the matrix generating routines. The listing of the signs of the second differences can then be examined for consistency. More information about this can be found in Step 2-A.

The output from this program is directly usable in the second step of the lofting system. Operating instructions, card formats, and a program listing are included in Appendix A.

*The smoothing process is described in Mathematical Ship Lofting - Part 1, Theory, Technical Report 1.0.0-1

B. PROFILES

In addition to the smoothed offsets, functions which describe the fore and aft profiles must be presented to the matrix generating routines of Step 2. This process is represented by the right-hand branch of Fig. II-1. The functions representing the end profiles are obtained from the end profile coordinates using the approach of Step 4.

The function representing the profile is $x = f(z)$ as shown in Fig. II-2. The origin of this equation must be the same as the origin of the surface equation for the hull, and must be scaled the same (see Step 2). The data points for describing the profile must be on the same waterlines as are used for the surface grid.

The coefficients of the equations for the profiles are punched on cards in the format shown in Appendix B to be entered directly into the Step 2 routines.

Section III

GENERATING THE LINEAR PROGRAMMING MATRIX (STEP 2)

The matrix for the linear program is produced by a set of 1620 programs called SMOG (Surface Matrix from Offsets Generator). These routines accept the data assembled in Step 1 and produce the specific matrix required for a particular surface, including the auxiliary information required by the LP program for bookkeeping purposes.

There are four versions of SMOG (Appendix B):

SMOG-1 produces the matrix required for a surface problem, without profile, using the dual formulation and λ fit.

SMOG-2 produces the matrix required for a surface problem, with profile, using the dual formulation and λ fit.

SMOG-3 produces the matrix required for a surface problem, with profile, using the primal formulation and sum of the deviations fit.

SMOG-4 produces a matrix for a 2-dimensional curve using the primal formulation and sum of the deviations fit.

Most surface problems will be run using SMOG-1 or 2 . The first three surface versions produce their output in the SHARE Standard LP format, although the output from SMOG-3 must be rearranged as shown in the Appendix.

The general operation of the programs is:

(1) Read the data and, if necessary, convert the offsets from feet-fractions to feet-in-decimal form

(2) Calculate and, if required, print the intersections, if any, of each station and waterline with the profile

(3) Calculate and, if desired, print values and signs of the second differences in both the waterline and station directions

(4) Calculate and punch the LP matrix, consisting of deviation constraints, curvature constraints, and the peripheral information required by the linear program.

A. MANUAL SMOOTHING

In operation (3) above, the signs of the second differences are found and can be printed. The second differences of the offsets are calculated using a general formula. Therefore, there is no requirement that the stations or waterlines chosen to describe the surface be equally spaced.

If the offsets have not previously been run through the smoothing routine, they may be visually checked at this time. That is, they may be compared against the preliminary lines drawing to see if they actually describe a curve of the same character as shown on the drawing. A second check is to investigate the consistency of the signs of the second differences. For instance, if a single positive second difference turns up in the middle of a string of negative values, the point should be carefully checked for validity and, if necessary, changed a small amount to adjust the sign.

There are also some cases when the correct values for the points will cause the wrong sign to be calculated. This is illustrated in Fig. III-1. The example is taken from Station 14 of the DLG-26 class ships. Using points from the base line, 4' and 8' waterlines, a negative sign will be calculated for the second difference at the 4' waterline, where the character of the curve suggests a positive value. Simply changing the sign within the program before punching the curvature constraints will correct this situation.

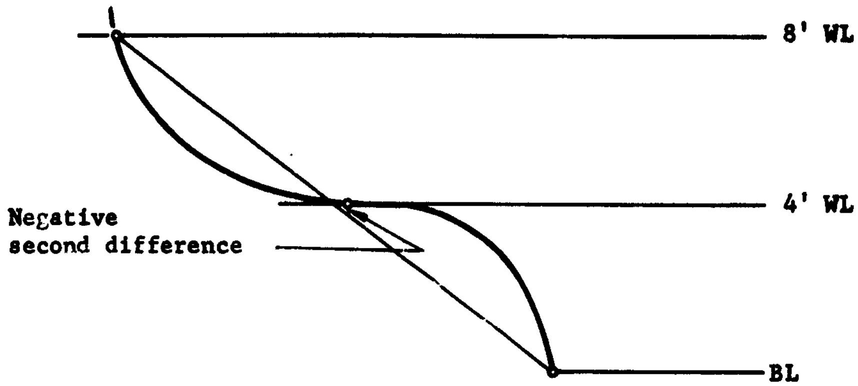


Fig. III-1

B. SCALING

All of the linear programming routines which have been examined have a restriction that the size of a matrix element be less than the absolute value of 10,000. Since some of our matrix elements are the product of two cubes $(X_i^3 \cdot Z_j^3)$ their size quickly becomes large. This has necessitated a limitation on the size of the X and Z coordinates. It can be readily seen that this limitation requires the product of the greatest X and Z coordinates to be less than the cube root of 10,000, or 21.53. In practice, the 4 x 5 combination is most common.

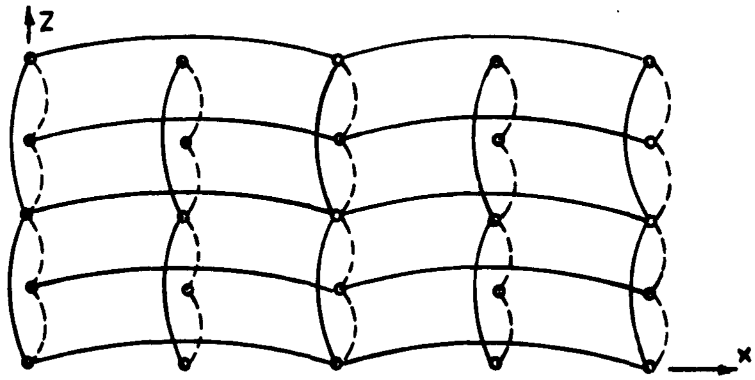
The coordinates may be scaled by dividing by some multiple of the station or waterline spacing. A surface five stations across, for example, might have a station spacing of 25' for a total length of 100' . Division by the station spacing gives a scaled value of 4 , or, the stations are one unit apart.

The most it is presently practical to scale is by four times the minimum station or waterline spacing, since this makes the minimum spacing one-fourth unit. The smallest matrix element then, becomes $(1/4)^3 \times (1/4)^3$ which, if the largest element is greater than 9000, creates an extreme range in the number size. This, of course, causes round-off and other difficulties in solving the matrix. The manner in which scaling is handled by the SMOG routines is noted in Appendix B.

Another important consideration for reducing maximum number size is to require that the first waterline and first station have zero coordinates. The SMOG programs have the ability to translate about the X or Z axis so the origin will occur at 0,0 .

C. SURFACE REPRESENTATION

All three of the surface matrix generators "double spline" the surface in the X direction. That is, they require a cubic to cover two intervals instead of one, as the "single spline" does. The cubic is required to have the proper curvature condition at every other station and is required to come as close as possible to every station. This, as can be seen in Fig. III-2, requires that every surface have an odd number of stations.



Solid lines - Double spline both ways
Combination, solid horizontal, broken vertical - double spline one way only

Fig. III-2 Span of Cubics on Surface

In the Z direction, there is a choice between double spline and single spline. The double spline representation requires a smaller matrix than the single spline representation. The double spline in the Z direction should be used whenever possible. It is, however, sometimes necessary to use single spline in order to gain the necessary flexibility for fitting surfaces with large variations in curvature and slope.

The double spline representation requires the data to contain an odd number of waterlines.

Section IV

USE OF LINEAR PROGRAM (STEP 3)

The linear programming routine which we have used most for fairing, is called LP-90.* It uses the SHARE standard format for input as punched by the SMOG programs. Some information on its use will be found in Appendix C.

Another LP system has been written by Bonner & Moore.** This program is faster than LP-90 for very dense problems, such as the fairing problem. It is, however, somewhat more difficult to use. This system also accepts SHARE standard input through the use of a special loading routine.

The matrix, which was produced by SMOG on the IBM-1620, must be converted to card/magnetic tape off-line on some machine, such as the IBM-1401, before its use with either LP.

The output from LP-90 is a magnetic tape that contains information on the problem written during its solution, and the final solution, which is the set of coefficients for the surface equation. By monitoring the output tape using the printer on the 1401, this final solution may be punched on cards. These cards may then be used as direct input to Step 5.

---

*LP-90 was written by CEIR Inc., Arlington, Virginia. It can be obtained from SHARE by sending them a magnetic tape on which they will copy the program. It is also available at any service bureau of CEIR.

**Bonner & Moore Associates, Houston, Texas. This system may be purchased from them or used at any Service Bureau Corp, Scientific Computing Center.

Section V

DEFINITION OF LONGITUDINAL STRUCTURAL INTERSECTIONS (STEP 4)

There is one type of intersection with the hull surface that requires additional information. This is the type which is shown in Fig. V-1. Examples are:

Shell plate sight edges
Longitudinal frames and stringers
Deck edges
Knuckles

The centerplane projection of these intersections is a fair spline curve.

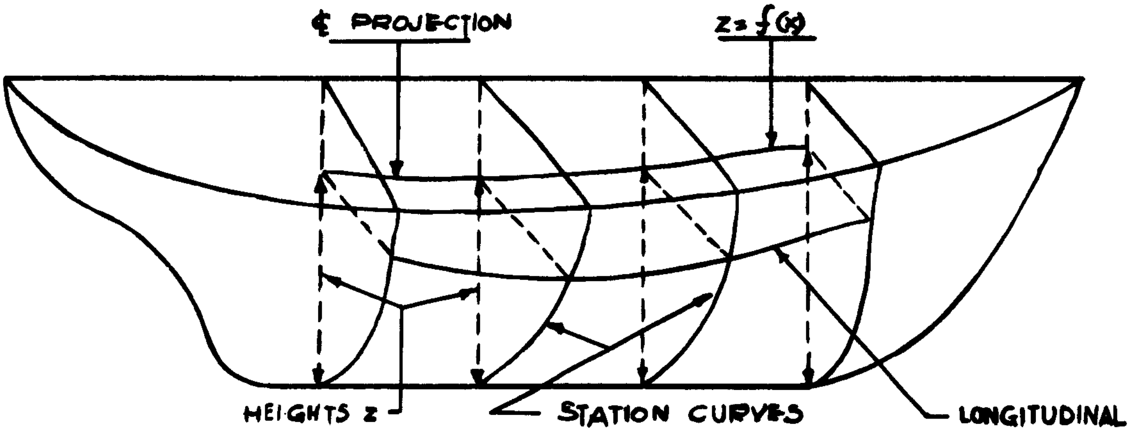


Fig. V-1 Longitudinal Intersections

In order to complete the definition of these curves, an equation for their height at any point along the length of the ship must be found ($z = f(x)$). This function is entered into the routines explained in Step 5, along with the hull equation to determine the height and half-breadth of these intersections on the fair hull. The procedure for obtaining these functions follows.

A. DATA PREPARATION

The data for each curve generally consists of the height of the curve at each station used to fair the hull surface (Fig. V-1). This data can be entered into the two-dimensional smoothing routine, Appendix A, or directly into the two-dimensional SMOG (SMOG 4) which is described below.

B. GENERATION OF MATRIX

The data from above, plus certain additional information as noted in Appendix B, are entered into SMOG 4. This program scales the data as desired, calculates and prints the second differences, and punches the matrix for the 1620 linear program given below. The matrix is punched in the sum of the deviations formulation. SMOG assumes the curve goes through the first point.

C. USE OF LINEAR PROGRAM

The linear programming code used for two-dimensional curves on the 1620 computer was obtained from the 1620 Users Group. The code* is far from ideal, since it contains many compromises in order to fit on the smallest 1620 available. It is, however, easily obtainable free of charge. There is a proprietary code written by Bonner & Moore Associates, which may be more efficient.

D. USE OF THE EQUATION

The coefficients obtained from the LP may be directly entered into the next phase after punching on cards in the proper format. These, together with the hull equation, enable the height and half-breadths of the intersections to be obtained.

*1620 Users Group Program 10.0.006

Section VI

CALCULATION OF BASIC DATA OF HULL FORM (STEP 5)

The determination of the basic data necessary to check the surface equation and to construct the ship is accomplished by a FORTRAN II program for the IBM-1620. This program is called GOBACK (the program reverses the fairing procedure and "goes back" to offsets). GOBACK is documented in Appendix D.

The program solves the surface equation for the following kinds of data:

Offsets along any waterline or station (not necessarily the original waterlines or stations) at any interval desired

Slopes and curvatures at any given interval along any waterline or station

Offsets along the intersection of the hull surface and any diagonal plane

Offsets along any buttock line

The coefficients of a Thielheimer equation for any waterline or station from the surface equation

The coefficients for standard cubic equations along any waterline or station

Offsets along any waterline or station at a variable interval, such that a straight line connecting each succeeding pair of points comes exactly at a given tolerance from the actual curve

Heights and offsets of the intersection of the hull equation with curves described by equations developed in Step 4, or with any cubic, parabola, or straight line.

The data from this program is directly usable in the programs for developing other loft information which follow it.

Section VII

PREPARATION OF NUMERICAL CONTROL DATA (STEP 6)

In Step 5, one of the output choices from GOBACK was the standard cubic equations for each interval on any waterline or station. With certain alterations, this is the data necessary to describe the hull contours for the numerical control programs.

The necessary alterations are:

> The fairing process is carried on entirely in feet and decimals thereof, while numerical control programs use inches. Therefore, the coefficients must be altered so that the independent variable can be entered in inches, and the dependent variable will be in inches.
>
> The results of the fairing process are nearly always scaled in some amount. Numerical control programs work in full scale only. Therefore, the coefficients must be re-scaled.
>
> The origin of the mathematically-lofted surface is often not at the point the numerical control programs require. Therefore, it must be possible to translate from one origin to the other.

A program has been written to make these alterations. This program accepts the output from GOBACK exactly as it is produced. In its present form, it produces data in such a format that it can be inserted directly into AUTOMAP* programs for parts definition. Program details are in Appendix E.

---

*See <u>Programming System for Numerically Controlled Flame Cutting of Ships Parts - Operating Manual</u>, Technical Report 5.0.0 (Vol. 4 of Final Report under Contract NObs-4427)

Section VIII

PLOTTING THE LINES DRAWINGS (STEP 7)

After obtaining the offsets from GOBACK, it is often desired to make a visual check on the fairness of the surface. For this purpose, a routine was written which uses the 1620 computer and a CalComp* plotter. This routine (Appendix F) accepts the output of GOBACK directly and plots stations, waterlines, or other curves as desired. While this program is written for a specific computer and plotter, the principles of operation would be the same for nearly any other computer-plotter combination.

*California Computer Products, Anaheim, California

Section IX

DETERMINATION OF OTHER LOFT DATA (STEP 8)

A. SHELL PLATE DEVELOPMENT

Using the offsets along sight edges and waterlines furnished by GOBACK, the shell plate development program furnishes the data for the flat plate patterns of the shell plate. This program is explained in Appendix G. The program will do one plate or an entire strake of plating at once. The output can be converted to numerical control input, manually plotted, or plotted by the computer using the plotting routine of Step 7. The data as punched can be entered directly into the plotting program.

When using plating developed by this program intermixed with plating developed manually, allowance must be made for a shift in seam location. This shift is fully explained in Appendix G of this report.

B. STRINGER DEVELOPMENT

Stringer plates differ from shell plates in that they have curvatures in the longitudinal direction only. That is, plates whose cross sections are straight lines but whose longitudinal edges are curves.

These plates are easily handled by the shell plate development program. It is necessary to provide a set of data describing the midline of the plate just as though a shellplate were involved. In the case of the stringer, this information is redundant, but it is needed for mechanical reasons.

C. FRAME AND PLATE BENDING OFFSETS

A program for computing the offsets for frame bending templates is presented in Appendix H. This program is equally applicable to computing offsets of plate bending templates.

The program requires an equation for the molded shape of the frame. It will furnish offsets for any portion of the frame described by the equation at any desired interval. The program accepts the equation in the form of a single standard cubic for each frame segment.

D. INNER HULL OFFSETS

Inner Hull contours of naval combatant ships, such as the DLG class, generally have the same characteristic as the outer hull contours. That is, the surface, or a large part of the surface, must be fair.

That part of the inner hull which has a fairness requirement can be faired by exactly the same methods as are used for the outer hull. Using Steps 1 through 8, the preliminary offsets are put through the same procedure as those for the hull itself.

E. INNER EDGE OF WEB FRAMES

In areas where the inner hull contours have not been faired, many web frames contain inner contours which are related to the outer hull contour. This relation is often a so-called "linear taper." The offset between the two curves is given at the upper and lower ends of the part, and the intermediate offsets are assumed to vary in a linear manner.

A program has been written to produce this taper. It is presented in Appendix I. The program uses points describing the outer hull contours and the upper and lower offset. It then finds points describing the shape of the inner hull contour.

F. DECK CAMBER OFFSETS

In Appendix J, a program for defining decks is given. This program calculates the vertical offsets from a horizontal plane at a distance equal to the molded depth above the baseline plane, to the deck for several configurations. These are:

Straight line sheer and straight line camber

Straight line sheer and parabolic camber

Parabolic sheer and straight line camber

Parabolic sheer and parabolic camber

Offsets can be obtained at any desired transverse interval at each frame.

 IX-3

Appendix A

SMOOTHING ROUTINES

CONTENTS

Appendix A

SMOOTHING ROUTINES

There are two smoothing programs. One is for two-dimensional curves and the other is for matrices of offsets describing a surface.

The three-dimensional (3-D) smoothing routine is the same as the two-dimensional (2-D) routine with an added iteration routine. The iterating routine smooths the points in one direction (waterlines) making necessary corrections, then smooths the points in the other direction (stations), iterating between the two directions until no further changes are required in the points. Several iterations may be required, of course, since a change in offset while smoothing in one direction may cause bad points in the other direction.

On any given curve, a change in offset will be made if, in the middle of a string of second differences of one sign, a single second difference of the opposite sign is found. If a single second difference of the opposite sign on either end of a curve is found, it will not be changed. Since the character of the curve cannot be examined beyond this point, the sign must be assumed to be legitimate, as indeed it may well be. The conditions just described are illustrated below:

○ Only these points will be changed

```
+ - - - - + - - - -
+ + + + - + + + + -
+ + + + - - - - + +
+ + + - - + + + + + -
+ + - + + + - +
```

If the program finds a case such as follows

```
+ + + + + - + - + + + + +
```

it will terminate execution. There are several ways to change this set to obtain a correct set of signs, and it will be possible to detect the correct one only by examining the lines drawing.

3-D SMOOTHING ROUTINE

This program is written in FORTRAN I for an IBM-1620 computer

a. Input Data

Symbols Used in the FORTRAN program:

MA = The number of waterlines contained in the data

NA = The number of stations contained in the data

WL(I) = The vertical distance Z_i from the origin of the data to waterline i

STA(J) = The horizontal distance X_j from the origin of the data to Station j

IASF = The number of feet in the offset $(Y_{i,j})$ when entered in feet, inches, and eighths form

IASI = The number of inches in the offset when entered in feet, inches, and eighths form

IASE = The number of eighth-inch intervals in the offset

ASET (I,J) = The offset $(Y_{i,j})$ when entered in decimal form after conversion from feet, inches and eighths

Header Card:

The data on this card consists of two fixed point numbers. These numbers must be right justified in the data field.

Variable	Format	Card Columns
MA	2I5	1 - 5
NA		6 - 10

Offset Data Cards:

The remaining data consists of the locations of and values for the offsets. These offsets can, by an appropriate switch setting, be read by the program in either feet and decimals of a foot form, or in feet, inches and eighths, similar to the preliminary table of offsets. The station and waterline coordinates are always in feet and decimal form.

This data consists of several subsets of data - one subset for each waterline in the offset matrix. The first card in each subset contains the height of the waterline (Z_j). The remaining cards of the subset each contain a station location (X_i) and the offset at that point ($Y_{i,j}$). The card format, if the offset is entered in feet and decimals, is:

Variable	Format	Card Columns
First card of a subset:		
WL(I)	F10.4	1 - 10
Remaining cards of a subset		
STA(J)	2F10.4	1 - 10
ASET(I,J)		11 - 20

If the offset is entered in feet, inches, and eighths from the card, formats are:

Variable	Format	Card Columns
First card of a subset:		
WL(I)	F10.4	1 - 10
Remaining cards of a subset:		
STA(J)	F10.4	1 - 10
ISAF		11 - 15
ISAE	3I5	16 - 20
ISAE		21 - 25

b. Output

1. Typewriter messages

If the program successfully completes the smoothing of the points, the message, FAIRING COMPLETE, is typed.

At any time a bad point in the data is detected, the message

BAD POINT, WATERLINE = ______ STATION = ______

LIMITS = _____ to _____, NEW POINT = ______

is typed. The limits denote the boundaries on the value of the bad offset for the proper sign of second difference. The new point is the actual offset value chosen.

If the indeterminate case described earlier is found, the following message is typed:

ADJACENT BAD POINTS

WATERLINE = _____STATION = _____

2. <u>Punched Cards</u>

There is no output from this program if the input is smooth without changes. If one or more offsets have been changed, a complete set of data is punched in exactly the same format as the decimal form of the input data.

c. <u>Sense Switch Settings</u>

SW 1 - OFF: Offsets are entered in feet and decimal form

SW 1 - ON : Offsets are entered in feet, inches, and eighths form

All other switches are ignored

SAMPIE DATA FOR 3-D SMOOTHING

Feet, Inches, and Eighths Form (Switch 1 ON)*

0 5 10 15 20 25 ← Card Columns

MA	NA	ISAF	IASI	IASE
7	5			
0.0	- WL(1)			
25.5	- STA(1)	1	0	0
51.0	- STA(2)	1	0	0
76.5		1	0	0
102.0		1	0	0
127.5		1	0	0
153.0		1	0	0
178.5		1	0	0
2.0	- WL(2)			
25.5	- STA(1)	4	0	3
51.0	- STA(2)	5	6	6
76.5		7	1	3
102.0		8	10	0
127.5		10	8	0
153.0		12	5	1
178.5		13	8	7
4.0				
25.5		5	8	0
51.0		7	10	0
76.5		10	0	3
102.0		12	4	2
127.5		14	7	6
153.0		16	7	1
178.5		18	0	2
8.0				
25.5		7	6	0
51.0		10	3	7
76.5		13	3	5
102.0		16	3	0
127.5		18	10	4
153.0		21	0	0
178.5		22	6	6
12.0				
25.5		8	5	1
51.0		11	9	3
76.5		15	2	0
102.0		18	4	1
127.5		20	0	7
153.0		23	2	7
178.5		24	9	2

*Data for 3-D smoothing in decimal form (Switch 1 **OFF**) is exactly the same as the sample output shown on **Pages A-6, A-7**.

TYPEWRITER LISTING WHEN PROCESSING SAMPLE DATA

```
260000200003RS  <------------------ Clear Memory Instruction
ENTER SUBROUTINES, PUSH START
1620 FORTRAN SUBR. AUTO DIV  9/30/61
LOAD DATA

BAD POINT, WATERLINE= 12.0000   STATION= 127.5000
LIMITS-     20.7916  TO     21.5208      NEW POINT=      21.1562


FAIRING COMPLETE
STOP
```

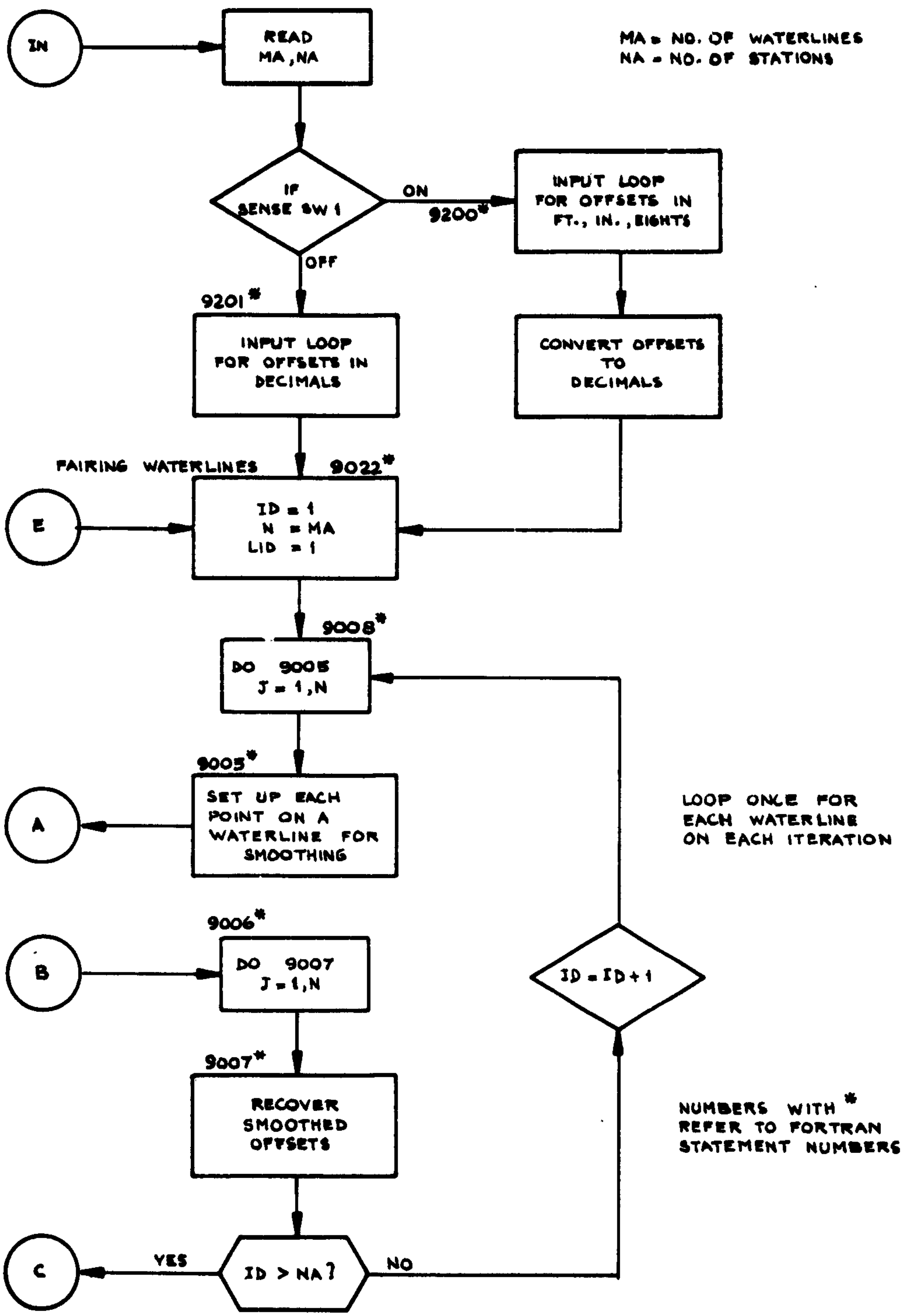
OUTPUT OBTAINED FROM PROCESSING SAMPLE DATA

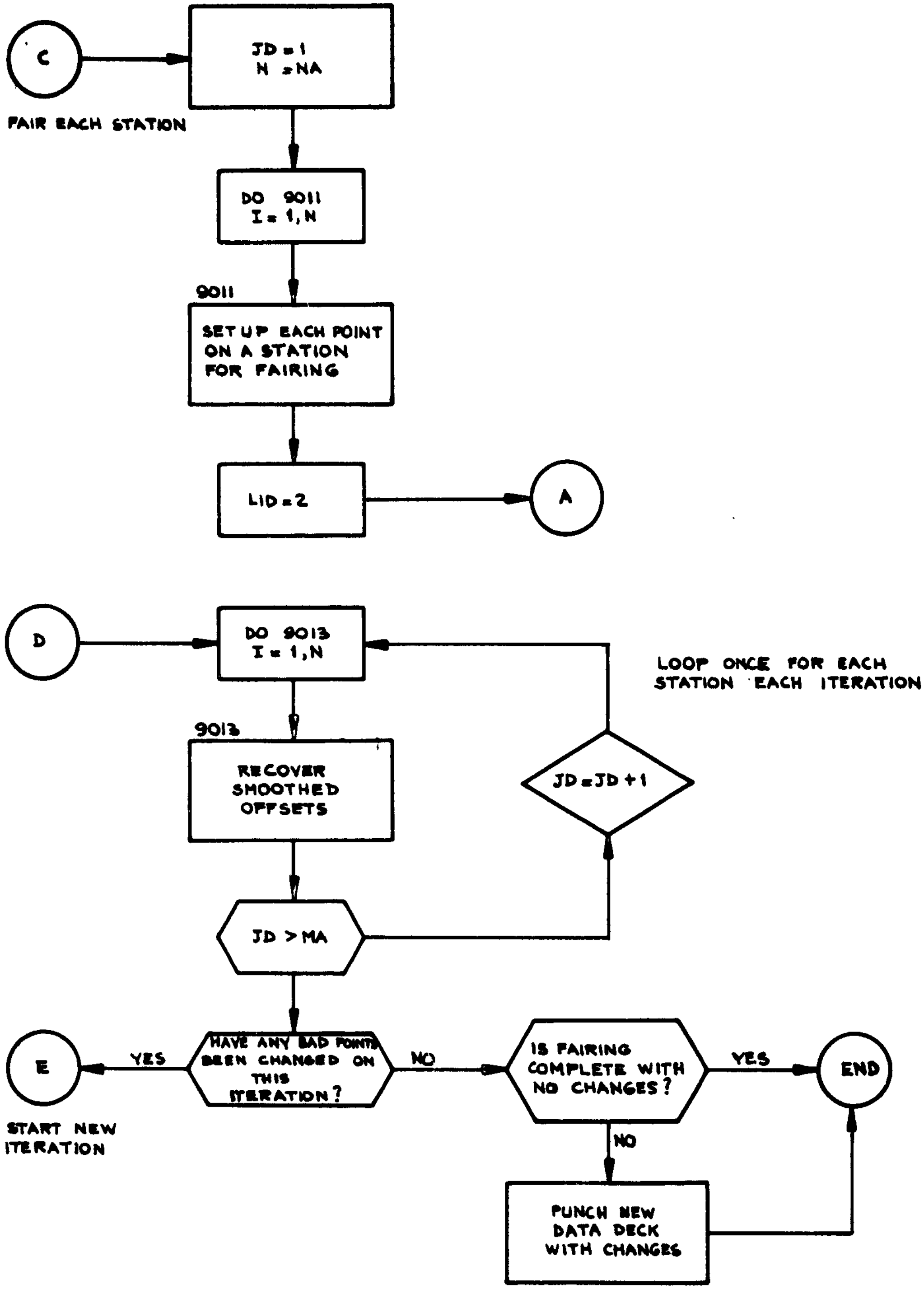
```
  7    5
   .0000
 25.5000    1.0000
 51.0000    1.0000
 76.5000    1.0000
102.0000    1.0000
127.5000    1.0000
153.0000    1.0000
178.5000    1.0000
  2.0000
 25.5000    4.0312
 51.0000    5.5625
 76.5000    7.1145
102.0000    8.8333
127.5000   10.6666
153.0000   12.4270
178.5000   13.7395
  4.0000
 25.5000    5.6666
 51.0000    7.8333
 76.5000   10.0312
102.0000   12.3541
127.5000   14.6458
153.0000   16.5937
178.5000   18.0208
  8.0000
 25.5000    7.5000
 51.0000   10.3229
 76.5000   13.3020
```

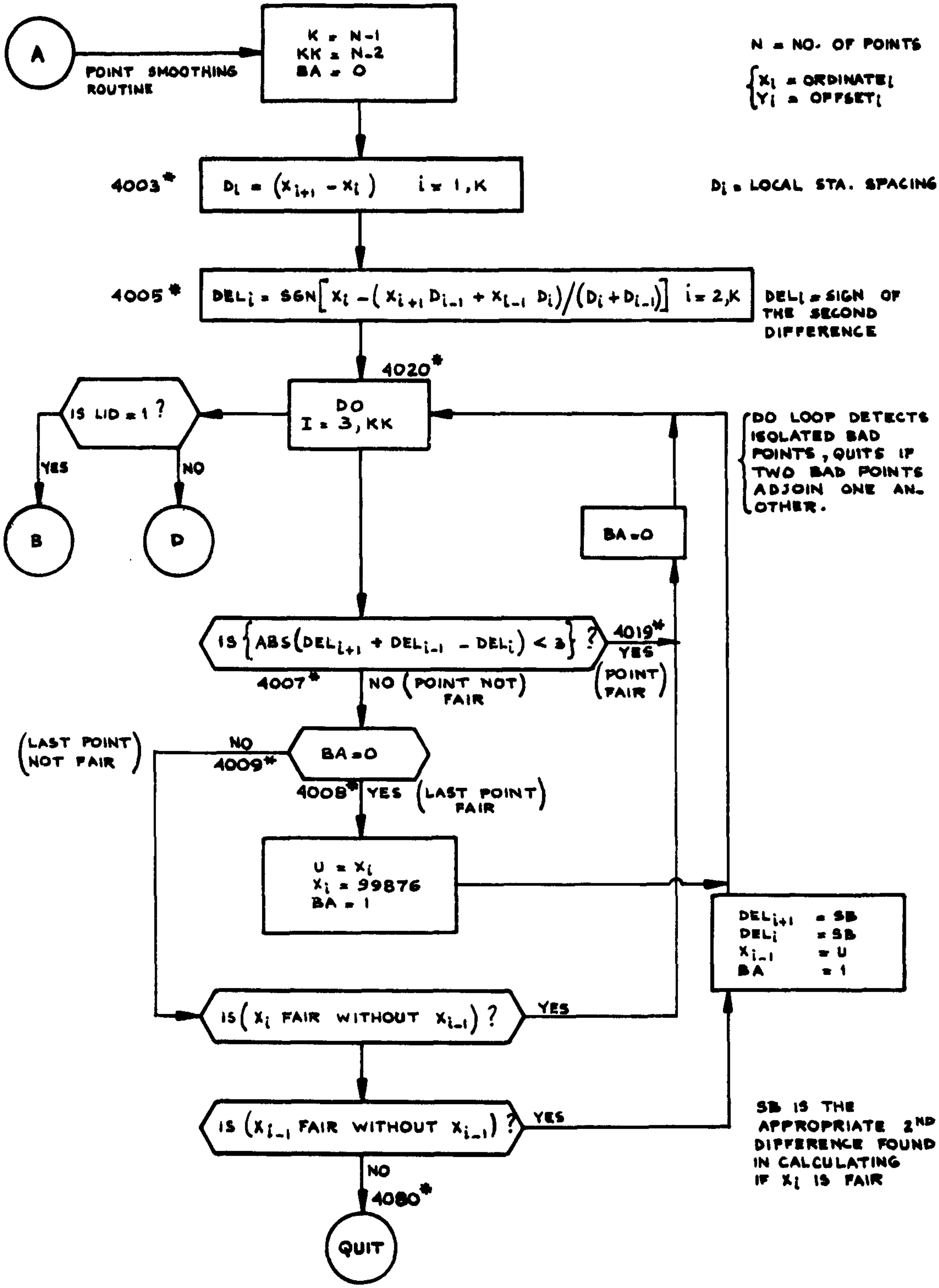
SAMPLE OUTPUT FROM 3-D SMOOTHING ROUTINE CONTINUED:

102.0000	16.2500
127.5000	18.8749
153.0000	21.0000
178.5000	22.5625
12.0000	
25.5000	8.4270
51.0000	11.7812
76.5000	15.1666
102.0000	18.3437
127.5000	21.1562
153.0000	23.2395
178.5000	24.7708

FLOW DIAGRAM FOR 3-D SMOOTHING ROUTINE




C
JD = 1
N = NA
FAIR EACH STATION
DO 9011
I = 1, N
9011
SET UP EACH POINT ON A STATION FOR FAIRING
LID = 2
A
D
DO 9013
I = 1, N
LOOP ONCE FOR EACH STATION EACH ITERATION
9013
RECOVER SMOOTHED OFFSETS
JD = JD + 1
JD > MA
HAVE ANY BAD POINTS BEEN CHANGED ON THIS ITERATION?
YES
NO
E
START NEW ITERATION
IS FAIRING COMPLETE WITH NO CHANGES?
YES
NO
END
PUNCH NEW DATA DECK WITH CHANGES


A
POINT SMOOTHING ROUTINE
K = N-1
KK = N-2
BA = 0
N = NO. OF POINTS
Xi = ORDINATEi
Yi = OFFSETi
4003*
Di = (Xi+1 - Xi) i = 1,K
Di = LOCAL STA. SPACING
4005*
DELi = SGN[Xi - (Xi+1 Di-1 + Xi-1 Di)/(Di + Di-1)] i = 2,K
DELi = SIGN OF THE SECOND DIFFERENCE
4020*
DO
I = 3, KK
IS LID = 1 ?
YES
NO
B
D
DO LOOP DETECTS ISOLATED BAD POINTS, QUITS IF TWO BAD POINTS ADJOIN ONE ANOTHER.
BA = 0
IS {ABS(DELi+1 + DELi-1 - DELi) < 3} ?
4019*
YES
(POINT FAIR)
4007*
NO (POINT NOT FAIR)
(LAST POINT NOT FAIR)
NO
4009*
BA = 0
4008*
YES
(LAST POINT FAIR)
U = Xi
Xi = 99876
BA = 1
DELi+1 = SB
DELi = SB
Xi-1 = U
BA = 1
IS (Xi FAIR WITHOUT Xi-1) ?
YES
IS (Xi-1 FAIR WITHOUT Xi-1) ?
YES
SB IS THE APPROPRIATE 2ND DIFFERENCE FOUND IN CALCULATING IF Xi IS FAIR
NO
4080*
QUIT

```
C     3-D       POINT FAIRING PROGRAM SURFACE
      DIMENSION ASET(12,25),WL(12),STA(25),AY(30)
      DIMENSION RD(25),FSET(30),DEL(25),D(25)
      LAB=1
      READ 1901,MA,NA
      IF(SENSE SWITCH 1)9200,9110
 9200 DO  9201 I=1,NA
      READ 9300,WL(I)
      DO 9201 J=1,MA
      READ 9300,STA(J),IASF,IASI,IASE
      ASF=IASF
      ASI=IASI
      ASE=IASE
 9201 ASET(I,J)=ASF+ASI/12.+ASE/96.
      GO TO 9022
 9300 FORMAT(F10.4,3I5)
 9110 DO 9001 I=1,NA
      READ 1902,WL(I)
      DO 9001 J=1,MA
 9001 READ 1902,STA(J),ASET(I,J)
 9022 Z=0.
C     FAIR EACH WATERLINE
      ID=1
      LID=1
      N=MA
 9008 DO 9005 J=1,N
      FSET(J)=ASET(ID,J)
 9005 RD(J)=STA(J)
      ADE=WL(ID)
      GO TO 4000
 9006 DO 9007 J=1,N
 9007 ASET(ID,J)=FSET(J)
      ID=ID+1
      IF (ID-NA) 9008,9008,9009
 9009 LID=2
      Z=0.
      LAB=LAB+1
C      FAIR EACH STATION
      JD=1
      N=NA
 9010 DO 9011 I=1,N
      FSET(I)=ASET(I,JD)
 9011 RD(I)=WL(I)
      ADE=STA(JD)
      GO TO 4000
 9012 DO 9013 I=1,N
 9013 ASET(I,JD)=FSET(I)
```

```
      JD=JD+1
      IF (JD-MA) 9010,9010,9014
9014  IF (Z) 9017,9017,9019
9019  LAB=LAB+1
      GO TO 9022
C     NO CHANGES FO A COMPLETE CYCLE---FAIRING COMPLETE
9017  PRINT 1906
      IF (LAB-2) 9080,9018,9018
9018  PUNCH 1901,MA,NA
      DO 9100 I=1,NA
      PUNCH 1902,WL(I)
      DO 9100 J=1,MA
9100  PUNCH 1902,STA(J),ASET(I,J)
9080  STOP
1901  FORMAT (I5,I5,F11.6)
1902  FORMAT (2F10.4)
1906  FORMAT (///16HFAIRING COMPLETE)
C     MAIN PROGRAM
4000  R=0.
4001  BA=0.
      DO 4003 I=2,N
4003  D(I)=RD(I)-RD(I-1,
      K=N-1
4004  DO 4005 I=2,K
      DEL(I)=FSET(I)-(FSET(I+1)*D(I)+FSET(I-1)*D(I+1))/(D(I)+D(I+1))
4005  DEL(I)=SGN(DEL(I))
      KK=K-1
      DO 4020 I=3,KK
      IF (ABSF(DEL(I)-DEL(I-1)-DEL(I+1))-2.) 4019,4019,4007
4007  IF (BA) 4080,4008,4009
4008  UPSET=FSET(I)
      FSET(I)=99876.
      BA=1.
      GO TO 4020
4009  SA=(FSET(I+1)*D(I-1)+FSET(I-2)*(D(I+1)+D(I)))/(RD(I+1)-RD(I-2))
      SA=SGN(UPSET-SA)
      SC=(FSET(I+1)*(D(I)+D(I-1))+FSET(I-2)*D(I+1))/(RD(I+1)-RD(I-2))
      SC=SGN(FSET(I)-SC)
      SB=(FSET(I+2)*(D(I+1)+D(I))+UPSET*D(I+2))/(RD(I+2)-RD(I-1))
      SB=SGN(FSET(I+1)-SB)
      SD=(FSET(I)*D(I-2)+FSET(I-3)*(D(I)+D(I-1)))/(RD(I)-RD(I-3))
      SD=SGN(FSET(I-2)-SD)
      TA=DEL(I-2)+SB-SA
      TC=DEL(I+1)+SD-SC
```

```
      R=R+1.
      IF (ABSF(TA)-2.) 4014,4014,4015
4014  FSET(I-1)=UPSET
      DEL(I)=SB
      DEL(I+1)=SB
      FSET(I)=99876.
      BA=-1.
      GO TO 4020
4015  IF (ABSF(TC)-2.) 4017,4017,4080
4017  BA=-1.
      GO TO 4020
4080  IDD=I-1
      PRINT 1409
      GO TO (3101,3102),LID
3101  PRINT 1408,ADE,RD(I)
      STOP
3102  PRINT 1408,RD(I),ADE
      STOP
1409  FORMAT (/////19HADJACENT BAD POINTS)
1408  FORMAT (10HWATERLINE=,F8.4,10X,8HSTATION=,F9.4)
4019  BA=0.
4020  CONTINUE
      DO 4021 I=3,KK
      IF (FSET(I)-99876.) 4021,4092,4021
4092  SL=(FSET(I+1)*D(I)+FSET(I-1)*D(I+1))/(D(I+1)+D(I))
      SA=(FSET(I-1)*(D(I)+D(I-1))-FSET(I-2)*D(I))/D(I-1)-SL
      SB=(FSET(I+1)*(D(I+1)+D(I+2))-FSET(I+2)*D(I+1))/D(I+2)-SL
      IF (ABSF(SA)-ABSF(SB)) 4120,4120,4119
4119  SA=SB
4120  FSET(I)=SL+0.5*SA
      XUP=SL+SA
      XLOW=SL
      Z=Z+1.
      GO TO (3001,3002),LID
3001  PRINT 1003,ADE,RD(I)
      GO TO 3003
3002  PRINT 1003,RD(I),ADE
3003  CONTINUE
      PRINT 1004,XLOW,XUP,FSET(I)
4021  CONTINUE
      GO TO (9006,9012),LID
1003  FORMAT (/21HBAD POINT, WATERLINE=,F8.4,3X, 8HSTATION=,F9.4)
1004  FORMAT (7HLIMITS-,F12.4,4H  TO,F12.4,5X,10HNEW POINT=,F12.4)
      END
```

2-D SMOOTHING ROUTINE

This program is written in FORTRAN I for an IBM-1620 computer. The program is exactly the same as the 3-D Smoothing, without the iterating routine. Thus it is equipped to handle just one curve at a time.

a. Input Data

Symbols Used in the FORTRAN program:

N = Number of data points to define the curve

RD(I) = Ordinate of the offset (X_i)

IFSF = The number of feet in the offset (Y_i) when entered in feet, inches, and eighths form

IFSI = The number of inches in the offset when entered in feet, inches, and eighths form

IFSE = The number of eighths inches in the offset when entered in feet, inches, and eighths form

FSET(I) = The offset (Y_i) when entered in decimal form or after conversion from feet, inches, and eighths

Header Card Format:

Variable	Format	Card Columns
N	I5	1 - 15

Offset Data Card Format:

N cards, the set of which must have one of the following formats:

Variable	Format	Card Columns
Sense Switch 1 ON:		
RD(I)	F10.4	1 - 10
IFSF		11 - 15
IFSI	3I5	16 - 20
IFSE		21 - 25
Sense Switch 1 OFF:		
RD(I)	2F10.4	1 - 10
FSET(I)		11 - 20

b. <u>Output</u>

1. If the points are smooth without change, a halt with a READER NO FEED light will occur.

2. If any point needs change its location, the limits of the possible offset values and the selected offset value will be printed.

3. The indeterminate case described in the 3-D Writeup will cause the point numbers of the offending offsets and the word STOP to be printed on the typewriter.

c. <u>Sense Switches</u>

Same as 3-D Smoothing Program

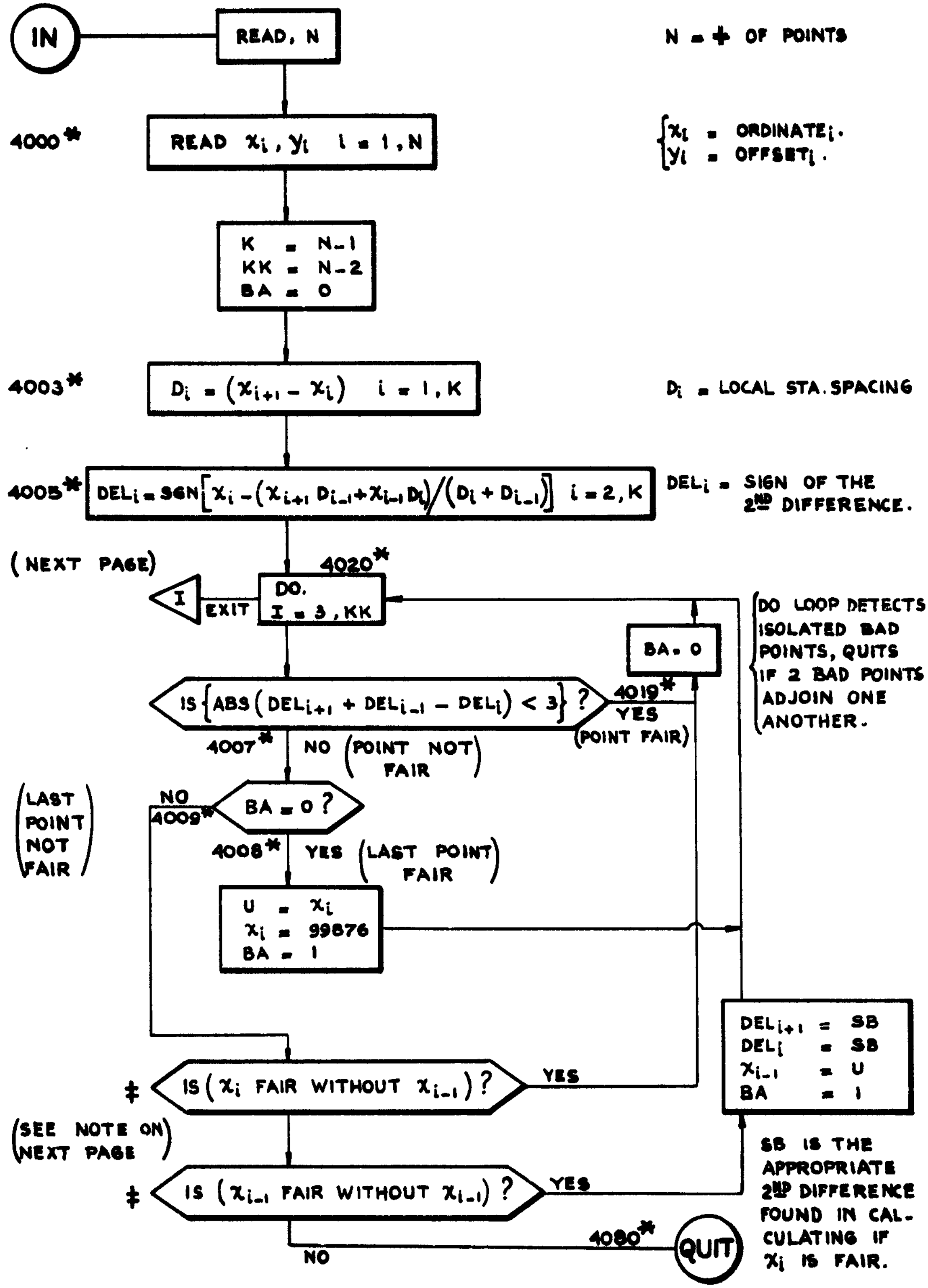
TYPEWRITER LISTING WHEN EXECUTING THE 2-D SMOOTHING PROGRAM

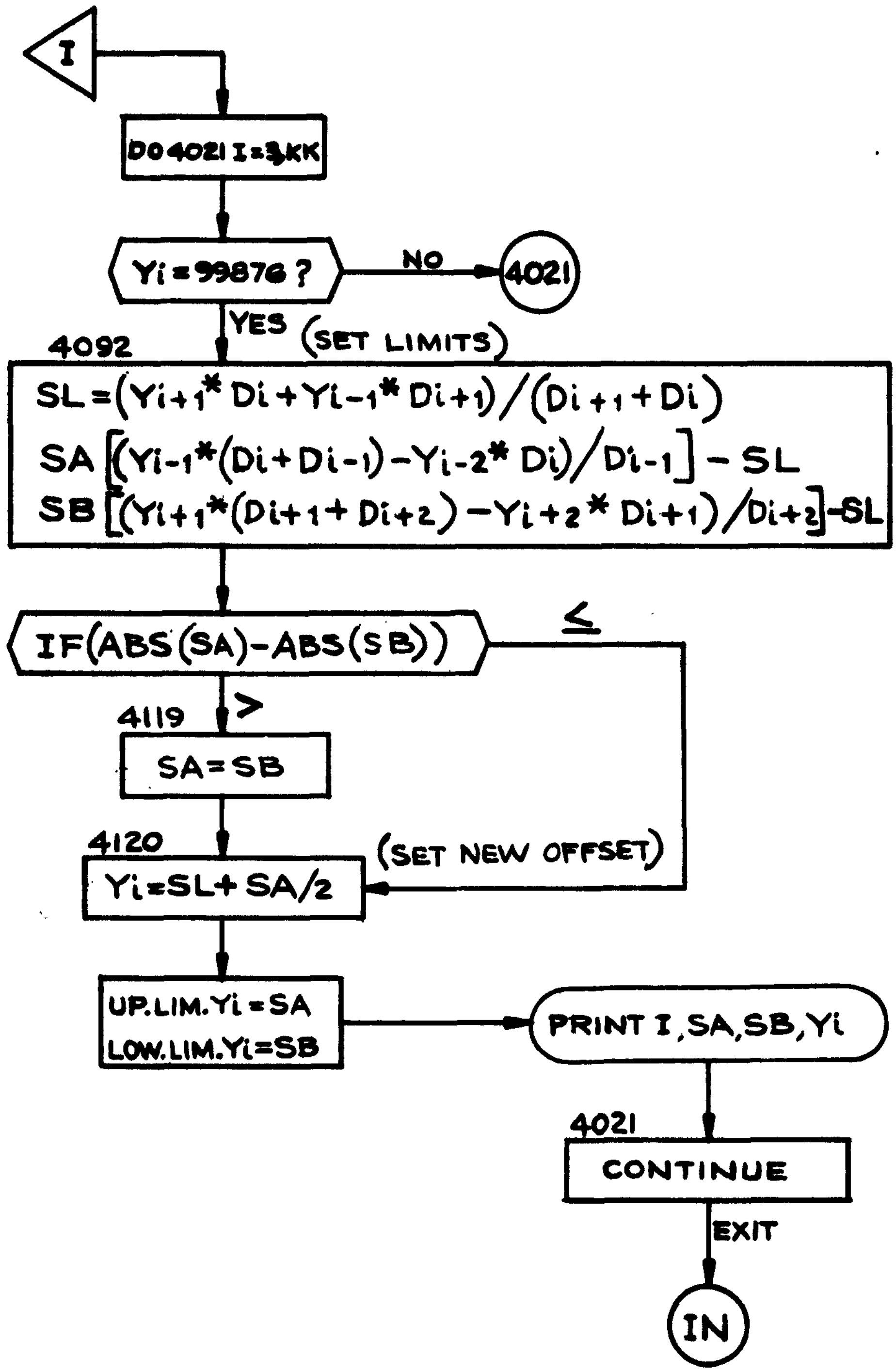
```
    260000200003     <-------------Instruction for Clearing Memory
    LOAD DATA
BAD POINT, STA= 25.5000
LIMITS-      20.7916  TO      21.5208      NEW POINT=      21.1562
STOP
```

SAMPLE INPUT DATA

```
   7
  5.1        8     5    1
 10.2       11     9    3
 15.3       15     2    0
 20.4       18     4    1
 25.5       20     0    7
 30.6       23     2    7
 35.7       24     9    2
```

FLOW DIAGRAM FOR 2-D SMOOTHING ROUTINE




I
DO 4021 I=3,KK
Yi=99876 ?
NO
4021
YES
(SET LIMITS)
4092
SL=(Yi+1* Di+Yi-1* Di+1)/(Di+1+Di)
SA [(Yi-1*(Di+Di-1)-Yi-2* Di)/Di-1] - SL
SB [(Yi+1*(Di+1+Di+2)-Yi+2* Di+1)/Di+2]-SL
IF(ABS(SA)-ABS(SB))
≤
>
4119
SA=SB
4120
(SET NEW OFFSET)
Yi=SL+SA/2
UP.LIM.Yi=SA
LOW.LIM.Yi=SB
PRINT I,SA,SB,Yi
4021
CONTINUE
EXIT
IN

```
C    2-D        POINT FAIRING PROGRAM
      DIMENSION RD(20),FSET(20),D(20),DEL(20)
     READ 1001,N
      IF(SENSE SWITCH 1)9100,9000
 9100 DO 9101 I=1,N
      READ 1005,RD(I),IFSF,IFSI,IFSE
      FSF=IFSF
      FSI=IFSI
      FSE=IFSE
 9101 FSET(I)=FSF+FSI/12.+FSE/96.
      GO TO 4000
 1005 FORMAT(F10.4,3I5)
 9000 DO 1 I=1,N
1     READ 1002,RD(I),FSET(I)
4000  R=0.
4001  BA=0.
      DO 4003 I=2,N
4003  D(I)=RD(I)-RD(I-1)
      K=N-1
4004  DO 4005 I=2,K
      DEL(I)=FSET(I)-(FSET(I+1)*D(I)+FSET(I-1)*D(I+1))/(D(I)+D(I+1))
4005  DEL(I)=SGN(DEL(I))
      KK=K-1
      DO 4020 I=3,KK
      IF (ABS(DEL(I)-DEL(I-1)-DEL(I+1))-2.) 4019,4019,4007
4007  IF (BA) 4080,4008,4009
4008  UPSET=FSET(I)
      FSET(I)=99876.
      BA=1.
      GO TO 4020
4009  SA=(FSET(I+1)*D(I-1)+FSET(I-2)*(D(I+1)+D(I)))/(RD(I+1)-RD(I-2))
     SA=SGN(UPSET-SA)
      SC=(FSET(I+1)*(D(I)+D(I-1))+FSET(I-2)*D(I+1))/(RD(I+1)-RD(I-2))
      SC=SGN(FSET(I)-SC)
      SB=(FSET(I+2)*(D(I+1)+D(I))+UPSET*D(I+2))/(RD(I+2)-RD(I-1))
      SB=SGN(FSET(I+1)-SB)
      SD=(FSET(I)*D(I-2)+FSET(I-3)*(D(I)+D(I-1)))/(RD(I)-RD(I-3))
      SD=SGN(FSET(I-2)-SD)
      TA=DEL(I-2)+SB-SA
      TC=DEL(I+1)+SD-SC
      R=R+1.
      IF (ABS(TA)-2.) 4014,4014,4015
4014  FSET(I-1)=UPSET
      DEL(I)=SB
     DEL(I+1)=SB
      FSET(I)=99876.
      BA=-1.
      CO TO 4020
4015  IF (ABS(TC)-2.) 4017,4017,4080
4017  BA=-1.
```

```
      GO TO 4020
4080  IDD=I-1
      PRINT 1408,IDD,I
1408  FORMAT (I2,I2)
      STOP
4019  BA=0.
4020  CONTINUE
      DO 4021 I=3,KK
      IF (FSET(I)-99876.) 4021,4092,4021
4092  SL=(FSET(I+1)*D(I)+FSET(I-1)*D(I+1))/(D(I+1)+D(I))
      SA=(FSET(I-1)*(D(I)+D(I-1))-FSET(I-2)*D(I))/D(I-1)-SL
      SB=(FSET(I+1)*(D(I+1)+D(I+2))-FSET(I+2)*D(I+1))/D(I+2)-SL
      IF (ABS(SA)-ABS(SB)) 4120,4120,4119
4119  SA=SB
4120  FSET(I)=SL+0.5*SA
      XUP=SL+SA
      XLOW=SL
      PRINT 1003,RD(I)
      PRINT 1004,XLOW,XUP,FSET(I)
4021  CONTINUE
1001  FORMAT (I5)
1002  FORMAT (2F10.4)
1003  FORMAT (15HBAD POINT, STA=,F10.4)
1004  FORMAT (7HLIMITS-,F12.4,4H  TO,F12.4,5X,10HNEW POINT=,F12.4)
      STOP
      END
```

Appendix B

SMOG

Contents

Section | Page

Section I

INTRODUCTION

The purpose of the SMOG programs is to generate the matrices necessary for input to the linear programming routine.

The specific use of each program presented in this Appendix is listed below.

SMOG 1 - A one pass program which generates a matrix for a surface without profile in the dual formulation and λ fit.

SMOG 2 - A two pass program which generates a matrix for a surface with profile in the dual formulation and λ fit. Pass 1 generates the deviation constraints; Pass 2 generates the curvature constraints.

SMOG 3 - A two pass program which generates a matrix for a surface with profile in the primal formulation and sum of the deviations fit. Pass 1 generates the deviation constraints; Pass 2 generates the curvature constraints.

SMOG 4 - A single pass program which generates a matrix for fairing a two-dimensional line with the primal formulation and λ fit, using the 1620 linear program

There is no requirement that the waterlines or stations used by this program be equally spaced. It is desirable because of the variation in size of the matrix elements produced, however, to have them as equally spaced as possible

SECOND DIFFERENCE EQUATIONS

The equation which is used to calculate the second differences is given below. The normalized values of the coordinates are used in finding these second differences.

For waterlines, the equation is:

$$SD_{i,j} = \frac{2}{(x_{j+1} - x_{j-1})} \left[\frac{y_{i,j+1} - y_{i,j}}{x_{j+1} - x_j} - \frac{y_{i,j} - y_{i,j-1}}{x_j - x_{j-1}} \right]$$

For stations, the equation is:

$$SD_{j,i} = \frac{2}{(z_{i+1} - z_{i-1})} \left[\frac{y_{j,i+1} - y_{j,i}}{z_{i+1} + z_i} - \frac{y_{j,i} - y_{j,i-1}}{z_i - z_{i-1}} \right]$$

NORMALIZING

The procedure for normalizing the x and z coordinates is given below. If the program establishes the normalizing and translating factors for the coordinates, these are found as follows:

$$S_x = x_1 - x_0 \qquad Tx = x_0$$

$$S_z = z_1 - z_0 \qquad Tz = z_0$$

Each x and z coordinate is then altered as follows:

$$x_j = \frac{x_j - T_x}{Sx}$$

$$z_i = \frac{z_i - Tz}{Sz}$$

If it is desired to scale and translate different amounts than this, the values Sx , Tx , Sz , and Tz may be read in.

OUTPUT DATA

The output data from each of the programs except SMOG 4 is in standard SHARE format. This is the accepted linear programming format established by the IBM 704, 7090 Users Group. Examples of each of the data cards punched by these programs is shown in Fig. B-1. Figure B-2 shows the organization of the data deck when ready for use with the linear program.

Column Number								
1–6	7	12	13	18	19		24	25–30
*			P	ROBLE	M	HEA	D	ING HERE
ROW ID							.	
		OFSET					.	
		R 1					.	
BASIS							.	
		C 100		R 1			.	
		C 101		R 2			.	
MATRIX							.	
		C 1		OFSET	-	XX	.	XXX
		C 1		R 1		XX	.	XXX
		C 1		R 2		XX	.	XXX
FIRST	B						.	
				R 1		XX	.	XX
				R 2		XX	.	XX
EOF							.	
							.	
							.	
							.	
							.	

53

Fig. B-1 LP/90 Card Formats
(SHARE Standard Format)

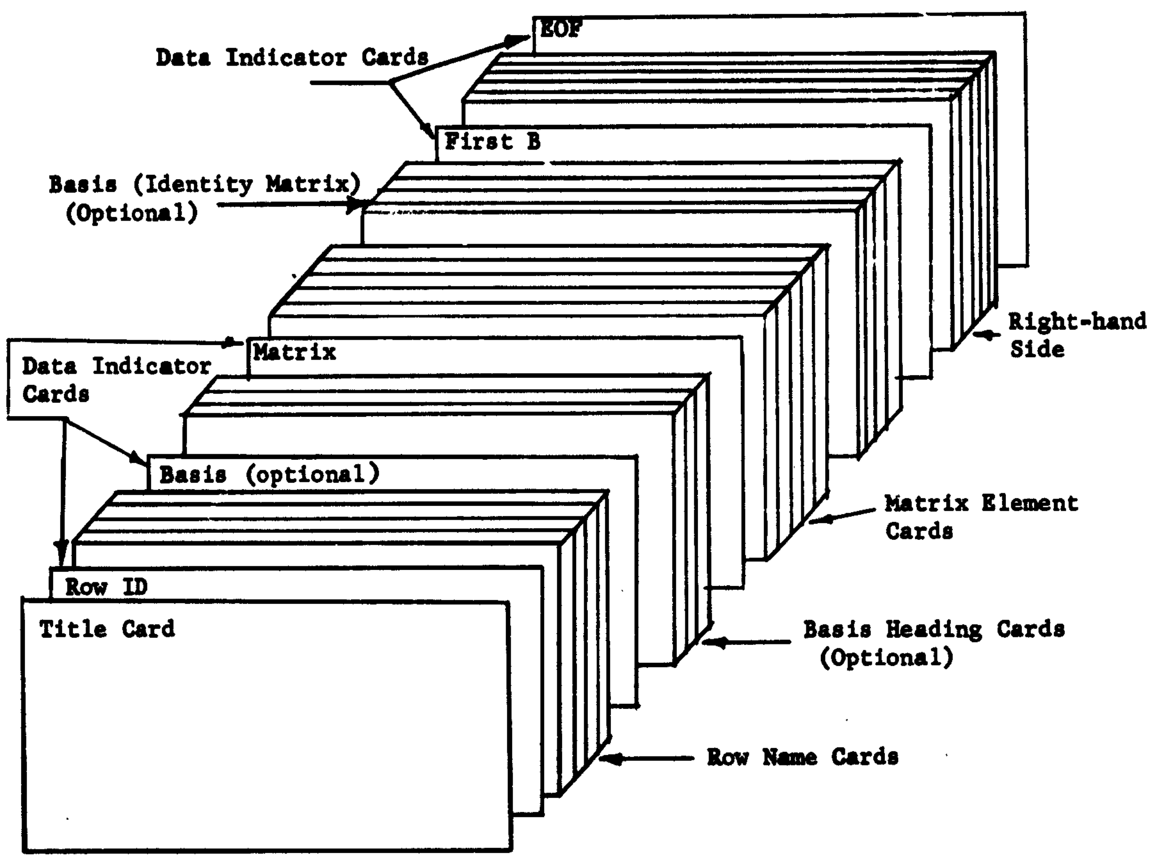


Fig. B-2 Input Card Order

Section II

SMOG 1 AND 2

The purpose of the programs, SMOG 1 and SMOG 2, is to generate the matrices required for surfaces with the following characteristics:

(1) Dual formulation

(2) λ (minimax fit)

(3) Double splined in x direction

(4) Single or double splined in the z direction

In addition, SMOG 2 produces a matrix with the modifications necessary for end profiles. The organization of each program is given in Fig. B-3.

The surface equation

Figure B-4 shows a five-station by five-waterline surface and the equation representing this surface. When producing a matrix to describe this problem to the linear program we must generate a requirement that the surface come close to each offset on the grid. The requirement is of the form

$$\left.\begin{array}{ll}\text{(a)} & \lambda - Y(x_i,z_j) = -Y_{i,j} \\ \text{(b)} & \lambda + Y(x_i,z_j) = Y_{i,j}\end{array}\right\} \begin{array}{l} i = 1\ ,\ \dots\ ,\ N \\ j = 1\ ,\ \dots\ ,\ M\end{array}$$

One (a) and one (b) column (constraints are columns in the dual) are generated for each offset. For instance, offset (1,3) Fig. B-4, would have the requirements

$$\text{(a)} \quad \lambda - A_{01}x_3 - A_{02}x_3^{\,2} - A_{03}x_3^{\,3} = -Y_{1,3}$$

$$\text{(b)} \quad \lambda + A_{01}x_3 + A_{02}x_3^{\,2} + A_{03}x_3^{\,3} = Y_{1,3}$$

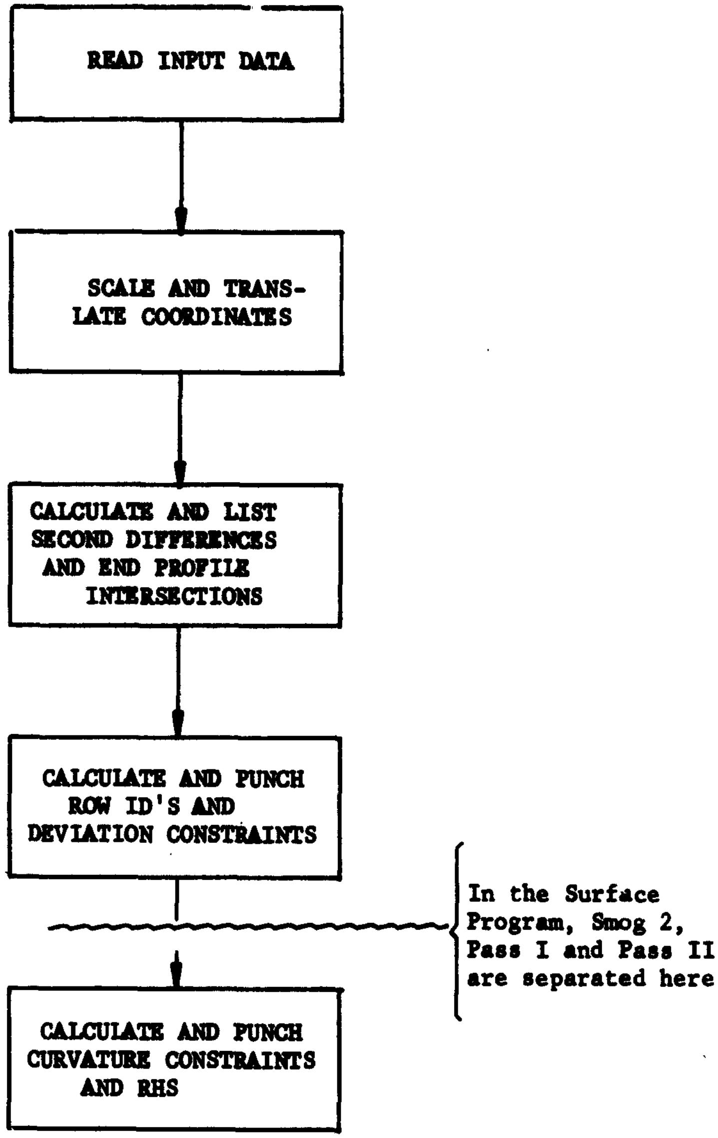
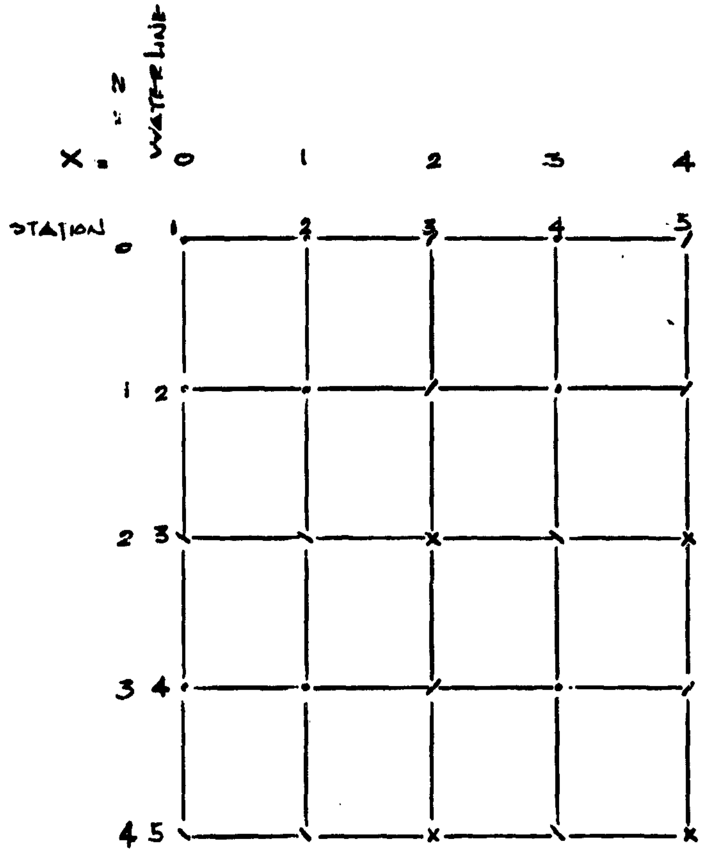


Fig. B-3 Organization of SMOG 1 and SMOG 2



$$
\begin{aligned}
Y = {} & A_{00} + A_{01}x + A_{02}x^2 + A_{03}x^3 + A_{04}(x-x_3)^3_{\#} \\
& + \left(A_{11}x + A_{12}x^2 + A_{13}x^3 + A_{14}(x-x_3)^3_{\#}\right) z \\
& + \left(A_{21}x + A_{22}x^2 + A_{23}x^3 + A_{24}(x-x_3)^3_{\#}\right) z^2 \\
& + \left(A_{31}x + A_{32}x^2 + A_{33}x^3 + A_{34}(x-x_3)^3_{\#}\right) z^3 \\
& + \left(A_{41}x + A_{42}x^2 + A_{43}x^3 + A_{44}(x-x_3)\right) (z-z_3)^3_{\#}
\end{aligned}
$$

/ Points where the sign of the x curvature is enforced

\ Points where the sign of the z curvature is enforced

Fig. B-4

The constant coefficient A_{oo} is given the value of the offset (x_1, z_1) and is not included in the matrix. (The surface always goes through this offset, therefore A_{oo} is known).

Similarly, the constraints for Point (3,3) are:

(a)
$$\lambda - \left[A_{01}x_3 + A_{02}x_3^{\ 2} + A_{03}x_3^{\ 3} \right]$$
$$- \left[A_{11}x_3 + A_{12}x_3^{\ 2} + A_{13}x_3^{\ 3} \right] z_3$$
$$- \left[A_{21}x_3 + A_{22}x_3^{\ 2} + A_{23}x_3^{\ 3} \right] z_3^{\ 2}$$
$$- \left[A_{31}x_3 + A_{32}x_3^{\ 2} + A_{33}x_3^{\ 3} \right] z_3^{\ 3} = -Y(3,3)$$

(b)
$$\lambda + \left[A_{01}x_3 + A_{02}x_3^{\ 2} + A_{03}x_3^{\ 3} \right]$$
$$+ \left[A_{11}x_3 + A_{12}x_3^{\ 2} + A_{13}x_3^{\ 3} \right] a_3$$
$$+ \left[A_{21}x_3 + A_{22}x_3^{\ 2} + A_{23}x_3^{\ 3} \right] z_3^{\ 2}$$
$$+ \left[A_{31}x_3 + A_{32}x_3^{\ 2} + A_{33}x_3^{\ 3} \right] z_3^{\ 3} = Y(3,3)$$

There are also constraints on the sign of the curvature at certain of the offsets. These constraints have the following form where $\widehat{r}_{i,j}$ and $\hat{\hat{r}}_{i,j}$ are the signs of the second differences found in Section I of this Appendix

(c) $\widehat{r}_{i,j} \cdot Y'' (x_i, z_j)_{zz} \geq 0$

(d) $\hat{\hat{r}}_{i,j} \cdot Y'' (x_i, z_j)_{xx} \geq 0$

At Offset (1,3) there is a curvature constraint in the x direction only, as shown in Fig. B-4 .

$$\text{(c)} \qquad \hat{r}_{i,j}\,(A_{02} - 3A_{03}x_3) \geq 0$$

There are x and z constraints both at offset (3,3) as follows:

$$\text{(c)} \qquad \hat{r}_{i,j}\left[-A_{02} - 3A_{03}x_3 - (A_{02} + 3A_{03}x_3)\,z_3 - (A_{02} + 3A_{03}x_3)\,z_x^{\,2} - (A_{02} + 3A_{03}x_3)\,z_3^{\,3}\right] \geq 0$$

$$\text{(d)} \qquad \hat{r}_{i,j}\left[-(A_{21}x_3 + A_{22}x_3^{\,2} + A_{23}x_3^{\,3}) - (A_{21}x_3 + A_{22}x_3^{\,2} + A_{23}x_3^{\,3})\,3z_3\right] \geq 0$$

PROVIDING A BASIC FEASIBLE SOLUTION

It is often desirable, when fairing a surface in the dual, to provide the linear program with a basic feasible solution. This consists of an identity matrix of slacks as shown in Fig. B-5.

This identity matrix is produced by the program given in Appendix K. The program is furnished with the number of the last non-basis column and the number of rows in the non-basis matrix. It will then produce a set of basis headings and the identity matrix.

The set of basis headings are inserted in the dual matrix immediately following the row identifications. The identity matrix is placed just ahead of the first B card. This matrix can be produced by SMOG 1, and the actual listing of the matrix for a surface is included in Section III of this Appendix.

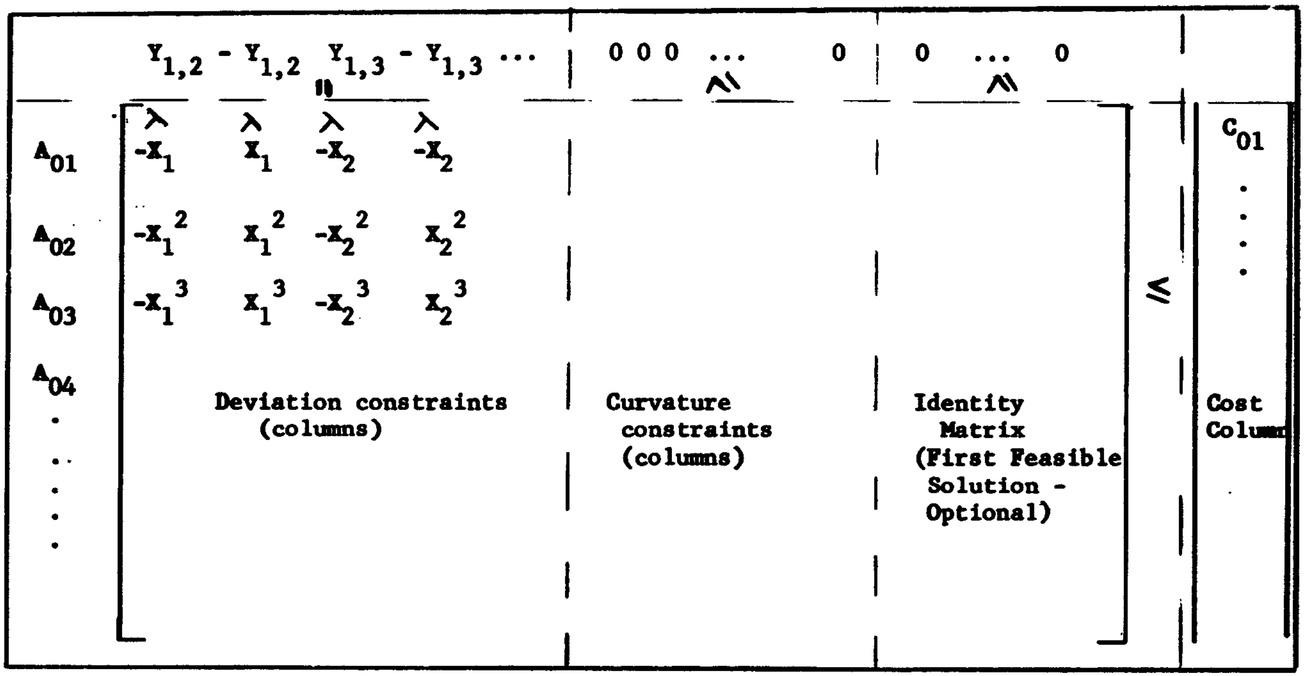


Fig. B-5 Matrix Tableau

Fig. B-5

SURFACE EQUATION WITH END PROFILE

In order to allow for end profiles on the surfaces, the surface equation $y = f(x,z)$ has been modified by multiplying it by a function T so that $y = f(x,z) \cdot T(x,z)$

where

$$\text{(e)} \qquad T(x,z) = \left[1 - \left(\frac{D + G(z) - x}{D} \right)^{r}_{\#} \right]^{P}_{\#}$$

$G(z)$ is the two-dimensional Theilheimer equation of the profile mentioned in Section II-B of the report. The equation $G(x)$ must have the same origin and be scaled the same as the surface equation. D is the fraction of the station spacing over which the function is effective, and P determines if the slope of the surface at the bow is finite or infinite. The usual value for P is 1 for finite slope, 1/3 for infinite slope. The value of r is normally 3 .

For any given value of x and z , when generating a matrix for a given surface $T(x,z)$ reduces to a number. Therefore, the deviation constraints can be modified in the following manner.

$$\text{(a)} \qquad \lambda - Y(x_i , z_j) = \frac{Y_{i,j}}{T(x_i, z_j)}$$

$$\text{(b)} \qquad \lambda + Y(x_i, z_j) = \frac{Y_{i,j}}{T(x_i, z_j)}$$

So the only modification necessary for producing deviation constraints is to divide the offset by the value of T . T is given the value one if the actual value is less than a given indicator.

The second derivative constraints modifications are more involved. The constraints are:

(1) for the x direction

$$\hat{r}_{i,j}\left[\frac{d^2}{dx^2}\left(f(x_i,z_j)\cdot T(x_i,z_j)\right)\right] \geq 0$$

(2) for the y direction

$$\hat{r}_{i,j}\left[\frac{d^2}{dz^2}\left(f(x_i,z_j)\cdot T(x_i,z_j)\right)\right] \geq 0$$

Using the rule for differentiating the product of two terms twice, these become:

(1)
$$\hat{r}_{i,j}\Big[f(x_i,z_j)\cdot T''(x_i,z_j)_{xx} + 2\left(T'(x_i,z_j)_x \cdot f'(x_i,z_j)_x\right)$$
$$+ T(x_i,z_j)\cdot f''(x_i,{}_z j)_{xx}\Big] \geq 0$$

(2)
$$\hat{r}_{i,j}\Big[f(x_i,z_j)\cdot T''(x_i,z_j)_{zz} + 2\left(T'(x_i,z_j)_z \cdot f'(x_i,z_j)_z\right)$$
$$+ T(x_i,z_j)\cdot f''(x_i,z_j)_{zz}\Big] \geq 0$$

These constraints are produced for the linear programming matrix by SMOG 2.

A. SMOG 1

OPERATING INSTRUCTIONS

This program produces a surface matrix without profile conditions in the dual formulation and λ fit. The surface will be double splined in the x direction and may be either single or double splined in the z direction. The program is written in FORTRAN II for an IBM-1620 computer.

Fortran Symbol Definitions

M	-	Total number of waterlines in the surface to be faired. This will be an odd number if the surface is double splined in the z direction.
N	-	Total number of stations in the surface. This will always be an odd number.
X(J)	-	Distance along the x axis from the origin of the surface to Station j .
Z(I)	-	Distance along the z axis from the origin of the surface to waterline i .
Y(I,J)	-	Offset of the surface at waterline i , station j , in feet and decimals
IAY	-	Number of feet in the offset if read in feet, inches and eighths.
IBY	-	Number of inches corresponding to IAY
ICY	-	Number of eighths corresponding to IAY
NAA	-	Program indicator If NAA = -1 scale factors for normalizing and translating X(J) and Z(I) are read in. If NAA = +1 scaling and translating are done by the program as given in Section I of this Appendix.
SX	-	Scale factor for X(J) when these are read in
TX	-	Translation factor for X(J) when these are read in
SZ	-	Scale factor for Z(I) when these are read in

TZ - Translation factor for Z(I) when these are read in

RHS - Right hand side - In the dual this is really the cost on the equation coefficients. The ideal value for RHS would be zero; however, this makes the solution susceptible to cycling so that a value of .0001 is usually used.

CURV - Value of the requirement vector for the curvature constraints, this value actually goes in the cost row in the dual and is always made zero.

NAB - Program indicator

If NAB = -1 , the surface will be double splined in the z direction and M must be an odd number.

If NAB = +1 , the surface will be single splined in the z direction and M is unrestricted.

Input

The input data cards are described below. There are certain input variables on which there are restrictions on the value which can be entered. These restrictions follow.

Variable	Minimum No.	Maximum No.
N	3	15
M	3	11

In describing the various data cards, the actual FORTRAN format field is used in most cases. These fields come consecutively across the card with no gaps or blank columns between, except where indicated. The field descriptions are the FORTRAN F field, which uses the FORTRAN fixed point decimal number, and the I field that uses the FORTRAN integer number, which is always right justified. The card numbers are <u>not</u> punched on the data cards.

Contents of Card						Card No.
Card Columns	1	2-12	13-53			1
Contents	*	Blank	Any alphameric problem heading (This heading will be used for identifying the problem on the LP output)			
Format	I5	I5				2
Variable	NAA	NAB				
Format	I5	I5	F11.6	F11.6		3
Variable	M	N	CURV	RHS		
Format	F10.4					Next M Groups
Variable	Z(I)					
Alternate Cards:						
Format	F10.4	F10.4			N Cards per Group	
Variable	X(J)	Y(I,J)				
or:						
Format	F10.4	I5	I5	I5		
Variable	X(J)	IAY	IBY	ICY		

This arrangement can be stated as follows:

The first card of each set gives the height z of the waterline. The following cards each give a station coordinate and the offset at the intersection of the given waterline and station. There is one set of cards for each waterline in the surface.

Format	F10.4	F10.4	F10.4	F10.4	Last card
Variable	SX	SZ	TX	TZ	

(This card only used if NAA = -1

Output

The output from this program is designed to be read directly by the LP-90 program after card to magnetic tape conversion.

The first card punched contains the Heading Read in the first input card. The next set of cards are the row identification cards. These identify each row of the matrix to be used in this solution. For this matrix all of the rows are always used and identified, including the cost row which has the name "OFSET."

After these comes a card punched "MATRIX," and then the input matrix to the linear program. The matrix is in column form with the rows in order within the column. Zero values are not punched.

The card formats are given in Fig. B- 1. The order of the output data is shown graphically in Fig. B- 2 . The sample problem output shows exactly what the output data consists of.

<u>Sense Switch Settings</u>

Sense Switch 1 - Not used	
Sense Switch 2 - ON	The values of the x and z second differences and the resulting signs are printed
Sense Switch 2 - OFF	The above is not printed
Sense Switch 3 - ON	The offsets are read in as feet, inches and eighths
Sense Switch 3 - OFF	The offsets are read in as feet and decimals

SAMPLE PROBLEM

The sample problem is a five-station by five-waterline set of offsets taken from the AG-88 class icebreaker. The stations used are Stations 13, 30, 32, 34, and 36 foot levels. The matrix is for an equation double splined in both the x and z directions.

INPUT

```
*           5X5 Z DOUBLE SPLINED AG-88 ICE BREAKER

   1   -1

   5    5 .000000     .0001

  28.0000
 137.5000    30     3     0
 150.0000    29     2     2
 162.5000    27     6     4
 175.0000    25     2     4
 187.5       21    10     6
  30.0000
 137.5000    30     7     5
 150.0000    29     7     7
 162.5000    28     2     5
 175.0000    26     2     4
 187.5000    23     3     5
  32.0000
 137.5000    30     8     5
 150.0000    29    10     4
 162.5000    28     7     5
 175.0000    26    10     4
 187.5000    24     4     3
  34.0000
 137.5000    30     8     3
 150.0000    29    11     2
 162.5000    28    10     5
 175.0000    27     3     5
 187.5000    25     1     7
  36.0000
 137.5000    30     5     5
 150.0000    29    10     4
 162.5000    28    11     5
 175.0000    27     6     6
 187.5000    25     7     3
```

TYPEWRITER LISTING OF SECOND DIFFERENCES
(Sense Switch 2 ON) **Switch 2 OUTPUT**

* 5X5 % DOUBLE SPLINED AG-88 ICE BREAKER

2ND DIFF - WLINES

WL	STA	ACT VALUE	SIGN
1	2	-.003732	1.00
1	3	-.004400	1.00
1	4	-.006266	1.00
1	5	-.006266	1.00
2	2	-.002933	1.00
2	3	-.003666	1.00
2	4	-.005733	1.00
2	5	-.005733	1.00
3	2	-.002532	1.00
3	3	-.003334	1.00
3	4	-.004799	1.00
3	5	-.004799	1.00
4	2	-.001865	1.00
4	3	-.003400	1.00
4	4	-.003599	1.00
4	5	-.003599	1.00
5	2	-.001999	1.00
5	3	-.003200	1.00
5	4	-.003467	1.00
5	5	-.003467	1.00

2ND DIFF-STATIONS

STA	WL	ACT VALUE	SIGN
1	2	-.075525	1.00
1	3	-.026025	1.00
1	4	-.052100	1.00
1	5	-.052100	1.00
2	2	-.062525	1.00
2	3	-.039050	1.00
2	4	-.031250	1.00
2	5	-.031250	1.00
3	2	-.065100	1.00
3	3	-.041675	1.00
3	4	-.041675	1.00
3	5	-.041675	1.00
4	2	-.083350	1.00
4	3	-.059875	1.00
4	4	-.041650	1.00
4	5	-.041650	1.00
5	2	-.085925	1.00
5	3	-.067700	1.00
5	4	-.083350	1.00
5	5	-.083350	1.00

1.0.0-2

```
* SMOG 1        5X5 Z DOUBLE SPLINED AG-88 ICE BREAKER
ROW ID
                 OFSET
                 R    1
                 R    2
                 R    3
                 R    4
                 R    5
                 R    6
                 R    7
                 R    8
                 R    9
                 R   10
                 R   11
                 R   12
                 R   13
                 R   14
                 R   15
                 R   16
                 R   17
                 R   18
                 R   19
                 R   20
                 R   21
                 R   22
                 R   23
                 R   24
                 R   25
MATRIX
             C    1 OFSET-      1.06260
             C    1 R    1      1.0
             C    1 R    2-     1.00000
             C    1 R    3-     1.00000
             C    1 R    4-     1.00000
             C    2 OFSET       1.06260
             C    2 R    1      1.0
             C    2 R    2      1.00000
             C    2 R    3      1.00000
             C    2 R    4      1.00000
             C    3 OFSET-      2.70840
             C    3 R    1      1.0
```

```
C    3 R     2-     2.00000
C    3 R     3-     4.00000
C    3 R     4-     8.00000
C    4 OFSET        2.70840
C    4 R     1      1.0
C    4 R     2      2.00000
C    4 R     3      4.00000
C    4 R     4      8.00000
C    5 OFSET-       5.04170
C    5 R     1      1.0
C    5 R     2-     3.00000
C    5 R     3-     9.00000
C    5 R     4-    27.00000
C    5 R     5-     1.00000
C    6 OFSET        5.04170
C    6 R     1      1.0
C    6 R     2      3.00000
C    6 R     3      9.00000
C    6 R     4     27.00000
C    6 R     5      1.00000
C    7 OFSET-       8.35420
C    7 R     1      1.0
C    7 R     2-     4.00000
C    7 R     3-    16.00000
C    7 R     4-    64.00000
C    7 R     5-     8.00000
C    8 OFSET        8.35420
C    8 R     1      1.0
C    8 R     2      4.00000
C    8 R     3     16.00000
C    8 R     4     64.00000
C    8 R     5      8.00000
C    9 OFSET         .38540
C    9 R     1      1.0
C    9 R     6-     1.00000
C    9 R    11-     1.00000
C    9 R    16-     1.0[illegible]00
C   10 OFSET-        .38[illegible]0
C   10 R     1      1.0
C   10 R     6      1.00000
C   10 R    11      1.00000
```

1.0.0-2

```
C   10 R   16      1.00000
C   11 OFSET-       .59380
C   11 R    1      1.0
C   11 R    2-     1.00000
C   11 R    3-     1.00000
C   11 R    4-     1.00000
C   11 R    6-     1.00000
C   11 R    7-     1.00000
C   11 R    8-     1.00000
C   11 R    9-     1.00000
C   11 R   11-     1.00000
C   11 R   12-     1.00000
C   11 R   13-     1.00000
C   11 R   14-     1.00000
C   11 R   16-     1.00000
C   11 R   17-     1.00000
C   11 R   18-     1.00000
C   11 R   19-     1.00000
C   12 OFSET        .59380
C   12 R    1      1.0
C   12 R    2      1.00000
C   12 R    3      1.00000
C   12 R    4      1.00000
C   12 R    6      1.00000
C   12 R    7      1.00000
C   12 R    8      1.00000
C   12 R    9      1.00000
C   12 R   11      1.00000
C   12 R   12      1.00000
C   12 R   13      1.00000
C   12 R   14      1.00000
C   12 R   16      1.00000
C   12 R   17      1.00000
C   12 R   18      1.00000
C   12 R   19      1.00000
C   13 OFSET-      2.03130
C   13 R    1      1.0
C   13 R    2-     2.00000
C   13 R    3-     4.00000
C   13 R    4-     8.00000
C   13 R    6-     1.00000
```

```
C   13 R    7-     2.00000
C   13 R    8-     4.00000
C   13 R    9-     8.00000
C   13 R   11-     1.00000
C   13 R   12-     2.00000
C   13 R   13-     4.00000
C   13 R   14-     8.00000
C   13 R   16-     1.00000
C   13 R   17-     2.00000
C   13 R   18-     4.00000
C   13 R   19-     8.00000
C   14 OFSET       2.03130
C   14 R    1      1.0
C   14 R    2      2.00000
C   14 R    3      4.00000
C   14 R    4      8.00000
C   14 R    6      1.00000
C   14 R    7      2.00000
C   14 R    8      4.00000
C   14 R    9      8.00000
C   14 R   11      1.00000
C   14 R   12      2.00000
C   14 R   13      4.00000
C   14 R   14      8.00000
C   14 R   16      1.00000
C   14 R   17      2.00000
C   14 R   18      4.00000
C   14 R   19      8.00000
C   15 OFSET-      4.04170
C   15 R    1      1.0
C   15 R    2-     3.00000
C   15 R    3-     9.00000
C   15 R    4-    27.00000
C   15 R    5-     1.00000
C   15 R    6-     1.00000
C   15 R    7-     3.00000
C   15 R    8-     9.00000
C   15 R    9-    27.00000
C   15 R   10-     1.00000
C   15 R   11-     1.00000
C   15 R   12-     3.00000
```

```
C   15 R   13-    9.00000
C   15 R   14-   27.00000
C   15 R   15-    1.00000
C   15 R   16-    1.00000
C   15 R   17-    3.00000
C   15 R   18-    9.00000
C   15 R   19-   27.00000
C   15 R   20-    1.00000
C   16 OFSET      4.04170
C   16 R    1     1.0
C   16 R    2     3.00000
C   16 R    3     9.00000
C   16 R    4    27.00000
C   16 R    5     1.00000
C   16 R    6     1.00000
C   16 R    7     3.00000
C   16 R    8     9.00000
C   16 R    9    27.00000
C   16 R   10     1.00000
C   16 R   11     1.00000
C   16 R   12     3.00000
C   16 R   13     9.00000
C   16 R   14    27.00000
C   16 R   15     1.00000
C   16 R   16     1.00000
C   16 R   17     3.00000
C   16 R   18     9.00000
C   16 R   19    27.00000
C   16 R   20     1.00000
C   17 OFSET-     6.94800
C   17 R    1     1.0
C   17 R    2-    4.00000
C   17 R    3-   16.00000
C   17 R    4-   64.00000
C   17 R    5-    8.00000
C   17 R    6-    1.00000
C   17 R    7-    4.00000
C   17 R    8-   16.00000
C   17 R    9-   64.00000
C   17 R   10-    8.00000
C   17 R   11-    1.00000
```

1.0.0-2

```
C   17 R   12-     4.00000
C   17 R   13-    16.00000
C   17 R   14-    64.00000
C   17 R   15-     8.00000
C   17 R   16-     1.00000
C   17 R   17-     4.00000
C   17 R   18-    16.00000
C   17 R   19-    64.00000
C   17 R   20-     8.00000
C   18 OFSET       6.94800
C   18 R    1      1.0
C   18 R    2      4.00000
C   18 R    3     16.00000
C   18 R    4     64.00000
C   18 R    5      8.00000
C   18 R    6      1.00000
C   18 R    7      4.00000
C   18 R    8     16.00000
C   18 R    9     64.00000
C   18 R   10      8.00000
C   18 R   11      1.00000
C   18 R   12      4.00000
C   18 R   13     16.00000
C   18 R   14     64.00000
C   18 R   15      8.00000
C   18 R   16      1.00000
C   18 R   17      4.00000
C   18 R   18     16.00000
C   18 R   19     64.00000
C   18 R   20      8.00000
C   19 OFSET        .46870
C   19 R    1      1.0
C   19 R    6-     2.00000
C   19 R   11-     4.00000
C   19 R   16-     8.00000
C   20 OFSET-       .46870
C   20 R    1      1.0
C   20 R    6      2.00000
C   20 R   11      4.00000
C   20 R   16      8.00000
C   21 OFSET-       .37510
```

```
C   21 R     1      1.0
C   21 R     2-     1.00000
C   21 R     3-     1.00000
C   21 R     4-     1.00000
C   21 R     6-     2.00000
C   21 R     7-     2.00000
C   21 R     8-     2.00000
C   21 R     9-     2.00000
C   21 R    11-     4.00000
C   21 R    12-     4.00000
C   21 R    13-     4.00000
C   21 R    14-     4.00000
C   21 R    16-     8.00000
C   21 R    17-     8.00000
C   21 R    18-     8.00000
C   21 R    19-     8.00000
C   22 OFSET         .37510
C   22 R     1      1.0
C   22 R     2      1.00000
C   22 R     3      1.00000
C   22 R     4      1.00000
C   22 R     6      2.00000
C   22 R     7      2.00000
C   22 R     8      2.00000
C   22 R     9      2.00000
C   22 R    11      4.00000
C   22 R    12      4.00000
C   22 R    13      4.00000
C   22 R    14      4.00000
C   22 R    16      8.00000
C   22 R    17      8.00000
C   22 R    18      8.00000
C   22 R    19      8.00000
C   23 OFSET-       1.61460
C   23 R     1      1.0
C   23 R     2-     2.00000
C   23 R     3-     4.00000
C   23 R     4-     8.00000
C   23 R     6-     2.00000
C   23 R     7-     4.00000
C   23 R     8-     8.00000
```

1.0.0-2

```
C  23 R    9-    16.00000
C  23 R   11-     4.00000
C  23 R   12-     8.00000
C  23 R   13-    16.00000
C  23 R   14-    32.00000
C  23 R   16-     8.00000
C  23 R   17-    16.00000
C  23 R   18-    32.00000
C  23 R   19-    64.00000
C  24 OFSET       1.61460
C  24 R    1      1.0
C  24 R    2      2.00000
C  24 R    3      4.00000
C  24 R    4      8.00000
C  24 R    6      2.00000
C  24 R    7      4.00000
C  24 R    8      8.00000
C  24 R    9     16.00000
C  24 R   11      4.00000
C  24 R   12      8.00000
C  24 R   13     16.00000
C  24 R   14     32.00000
C  24 R   16      8.00000
C  24 R   17     16.00000
C  24 R   18     32.00000
C  24 R   19     64.00000
C  25 OFSET-      3.37510
C  25 R    1      1.0
C  25 R    2-     3.00000
C  25 R    3-     9.00000
C  25 R    4-    27.00000
C  25 R    5-     1.00000
C  25 R    6-     2.00000
C  25 R    7-     6.00000
C  25 R    8-    18.00000
C  25 R    9-    54.00000
C  25 R   10-     2.00000
C  25 R   11-     4.00000
C  25 R   12-    12.00000
C  25 R   13-    36.00000
C  25 R   14-   108.00000
```

```
C   25 R   15-     4.00000
C   25 R   16-     8.00000
C   25 R   17-    24.00000
C   25 R   18-    72.00000
C   25 R   19-   216.00000
C   25 R   20-     8.00000
C   26 OFSET       3.37510
C   26 R    1      1.0
C   26 R    2      3.00000
C   26 R    3      9.00000
C   26 R    4     27.00000
C   26 R    5      1.00000
C   26 R    6      2.00000
C   26 R    7      6.00000
C   26 R    8     18.00000
C   26 R    9     54.00000
C   26 R   10      2.00000
C   26 R   11      4.00000
C   26 R   12     12.00000
C   26 R   13     36.00000
C   26 R   14    108.00000
C   26 R   15      4.00000
C   26 R   16      8.00000
C   26 R   17     24.00000
C   26 R   18     72.00000
C   26 R   19    216.00000
C   26 R   20      8.00000
C   27 OFSET--     5.88550
C   27 R    1      1.0
C   27 R    2-     4.00000
C   27 R    3-    16.00000
C   27 R    4-    64.00000
C   27 R    5-     8.00000
C   27 R    6-     2.00000
C   27 R    7-     8.00000
C   27 R    8-    32.00000
C   27 R    9-   128.00000
C   27 R   10-    16.00000
C   27 R   11-     4.00000
C   27 R   12-    16.00000
C   27 R   13-    64.00000
```

```
C  27 R   14-  256.00000
C  27 R   15-   32.00000
C  27 R   16-    8.00000
C  27 R   17-   32.00000
C  27 R   18-  128.00000
C  27 R   19-  512.00000
C  27 R   20-   64.00000
C  28 OFSET      5.88550
C  28 R    1     1.0
C  28 R    2      4.00000
C  28 R    3     16.00000
C  28 R    4     64.00000
C  28 R    5      8.00000
C  28 R    6      2.00000
C  28 R    7      8.00000
C  28 R    8     32.00000
C  28 R    9    128.00000
C  28 R   10     16.00000
C  28 R   11      4.00000
C  28 R   12     16.00000
C  28 R   13     64.00000
C  28 R   14    256.00000
C  28 R   15     32.00000
C  28 R   16      8.00000
C  28 R   17     32.00000
C  28 R   18    128.00000
C  28 R   19    512.00000
C  28 R   20     64.00000
C  29 OFSET       .44790
C  29 R    1     1.0
C  29 R    6-     3.00000
C  29 R   11-     9.00000
C  29 R   16-    27.00000
C  29 R   21-     1.00000
C  30 OFSET-      .44790
C  30 R    1     1.0
C  30 R    6      3.00000
C  30 R   11      9.00000
C  30 R   16     27.00000
C  30 R   21      1.00000
C  31 OFSET-      .31260
```

1.0.0-2

```
C  31 R    1     1.0
C  31 R    2-    1.00000
C  31 R    3-    1.00000
C  31 R    4-    1.00000
C  31 R    6-    3.00000
C  31 R    7-    3.00000
C  31 R    8-    3.00000
C  31 R    9-    3.00000
C  31 R   11-    9.00000
C  31 R   12-    9.00000
C  31 R   13-    9.00000
C  31 R   14-    9.00000
C  31 R   16-   27.00000
C  31 R   17-   27.00000
C  31 R   18-   27.00000
C  31 R   19-   27.00000
C  31 R   21-    1.00000
C  31 R   22-    1.00000
C  31 R   23-    1.00000
C  31 R   24-    1.00000
C  32 OFSET      .31260
C  32 R    1     1.0
C  32 R    2     1.00000
C  32 R    3     1.00000
C  32 R    4     1.00000
C  32 R    6     3.00000
C  32 R    7     3.00000
C  32 R    8     3.00000
C  32 R    9     3.00000
C  32 R   11     9.00000
C  32 R   12     9.00000
C  32 R   13     9.00000
C  32 R   14     9.00000
C  32 R   16    27.00000
C  32 R   17    27.00000
C  32 R   18    27.00000
C  32 R   19    27.00000
C  32 R   21     1.00000
C  32 R   22     1.00000
C  32 R   23     1.00000
C  32 R   24     1.00000
```

1.0.0-2

```
C   33 OFSET-     1.36460
C   33 R     1    1.0
C   33 R     2-   2.00000
C   33 R     3-   4.00000
C   33 R     4-   8.00000
C   33 R     6-   3.00000
C   33 R     7-   6.00000
C   33 R     8-  12.00000
C   33 R     9-  24.00000
C   33 R    11-   9.00000
C   33 R    12-  18.00000
C   33 R    13-  36.00000
C   33 R    14-  72.00000
C   33 R    16-  27.00000
C   33 R    17-  54.00000
C   33 R    18- 108.00000
C   33 R    19- 216.00000
C   33 R    21-   1.00000
C   33 R    22-   2.00000
C   33 R    23-   4.00000
C   33 R    24-   8.00000
C   34 OFSET      1.36460
C   34 R     1    1.0
C   34 R     2    2.00000
C   34 R     3    4.00000
C   34 R     4    8.00000
C   34 R     6    3.00000
C   34 R     7    6.00000
C   34 R     8   12.00000
C   34 R     9   24.00000
C   34 R    11    9.00000
C   34 R    12   18.00000
C   34 R    13   36.00000
C   34 R    14   72.00000
C   34 R    16   27.00000
C   34 R    17   54.00000
C   34 R    18  108.00000
C   34 R    19  216.00000
C   34 R    21    1.00000
C   34 R    22    2.00000
C   34 R    23    4.00000
```

```
C  34 R   24      8.00000
C  35 OFSET-      2.94800
C  35 R    1      1.0
C  35 R    2-     3.00000
C  35 R    3-     9.00000
C  35 R    4-    27.00000
C  35 R    5-     1.00000
C  35 R    6-     3.00000
C  35 R    7-     9.00000
C  35 R    8-    27.00000
C  35 R    9-    81.00000
C  35 R   10-     3.00000
C  35 R   11-     9.00000
C  35 R   12-    27.00000
C  35 R   13-    81.00000
C  35 R   14-   243.00000
C  35 R   15-     9.00000
C  35 R   16-    27.00000
C  35 R   17-    81.00000
C  35 R   18-   243.00000
C  35 R   19-   729.00000
C  35 R   20-    27.00000
C  35 R   21-     1.00000
C  35 R   22-     3.00000
C  35 R   23-     9.00000
C  35 R   24-    27.00000
C  35 R   25-     1.00000
C  36 OFSET      2.94800
C  36 R    1      1.0
C  36 R    2      3.00000
C  36 R    3      9.00000
C  36 R    4     27.00000
C  36 R    5      1.00000
C  36 R    6      3.00000
C  36 R    7      9.00000
C  36 R    8     27.00000
C  36 R    9     81.00000
C  36 R   10      3.00000
C  36 R   11      9.00000
C  36 R   12     27.00000
C  36 R   13     81.00000
```

```
C   36 R   14    243.00000
C   36 R   15      9.00000
C   36 R   16     27.00000
C   36 R   17     81.00000
C   36 R   18    243.00000
C   36 R   19    729.00000
C   36 R   20     27.00000
C   36 R   21      1.00000
C   36 R   22      3.00000
C   36 R   23      9.00000
C   36 R   24     27.00000
C   36 R   25      1.00000
C   37 OFSET-      5.09380
C   37 R    1      1.0
C   37 R    2-     4.00000
C   37 R    3-    16.00000
C   37 R    4-    64.00000
C   37 R    5-     8.00000
C   37 R    6-     3.00000
C   37 R    7-    12.00000
C   37 R    8-    48.00000
C   37 R    9-   192.00000
C   37 R   10-    24.00000
C   37 R   11-     9.00000
C   37 R   12-    36.00000
C   37 R   13-   144.00000
C   37 R   14-   576.00000
C   37 R   15-    72.00000
C   37 R   16-    27.00000
C   37 R   17-   108.00000
C   37 R   18-   432.00000
C   37 R   19-  1728.00000
C   37 R   20-   216.00000
C   37 R   21-     1.00000
C   37 R   22-     4.00000
C   37 R   23-    16.00000
C   37 R   24-    64.00000
C   37 R   25-     8.00000
C   38 OFSET       5.09380
C   38 R    1      1.0
C   38 R    2      4.00000
```

```
C   38 R    3     16.00000
C   38 R    4     64.00000
C   38 R    5      8.00000
C   38 R    6      3.00000
C   38 R    7     12.00000
C   38 R    8     48.00000
C   38 R    9    192.00000
C   38 R   10     24.00000
C   38 R   11      9.00000
C   38 R   12     36.00000
C   38 R   13    144.00000
C   38 R   14    576.00000
C   38 R   15     72.00000
C   38 R   16     27.00000
C   38 R   17    108.00000
C   38 R   18    432.00000
C   38 R   19   1728.00000
C   38 R   20    216.00000
C   38 R   21      1.00000
C   38 R   22      4.00000
C   38 R   23     16.00000
C   38 R   24     64.00000
C   38 R   25      8.00000
C   39 OFSET        .21870
C   39 R    1      1.0
C   39 R    6-     4.00000
C   39 R   11-    16.00000
C   39 R   16-    64.00000
C   39 R   21-     8.00000
C   40 OFSET-       .21870
C   40 R    1      1.0
C   40 R    6      4.00000
C   40 R   11     16.00000
C   40 R   16     64.00000
C   40 R   21      8.00000
C   41 OFSET-       .37510
C   41 R    1      1.0
C   41 R    2-     1.00000
C   41 R    3-     1.00000
C   41 R    4-     1.00000
C   41 R    6-     4.00000
```

```
C  41 R    7-     4.00000
C  41 R    8-     4.00000
C  41 R    9-     4.00000
C  41 R   11-    16.00000
C  41 R   12-    16.00000
C  41 R   13-    16.00000
C  41 R   14-    16.00000
C  41 R   16-    64.00000
C  41 R   17-    64.00000
C  41 R   18-    64.00000
C  41 R   19-    64.00000
C  41 R   21-     8.00000
C  41 R   22-     8.00000
C  41 R   23-     8.00000
C  41 R   24-     8.00000
C  42 OFSET        .37510
C  42 R    1      1.0
C  42 R    2      1.00000
C  42 R    3      1.00000
C  42 R    4      1.00000
C  42 R    6      4.00000
C  42 R    7      4.00000
C  42 R    8      4.00000
C  42 R    9      4.00000
C  42 R   11     16.00000
C  42 R   12     16.00000
C  42 R   13     16.00000
C  42 R   14     16.00000
C  42 R   16     64.00000
C  42 R   17     64.00000
C  42 R   18     64.00000
C  42 R   19     64.00000
C  42 R   21      8.00000
C  42 R   22      8.00000
C  42 R   23      8.00000
C  42 R   24      8.00000
C  43 OFSET-     1.28130
C  43 R    1      1.0
C  43 R    2-     2.00000
C  43 R    3-     4.00000
C  43 R    4-     8.00000
```

1.0.0.2

```
C   43 R    6-     4.00000
C   43 R    7-     8.00000
C   43 R    8-    16.00000
C   43 R    9-    32.00000
C   43 R   11-    16.00000
C   43 R   12-    32.00000
C   43 R   13-    64.00000
C   43 R   14-   128.00000
C   43 R   16-    64.00000
C   43 R   17-   128.00000
C   43 R   18-   256.00000
C   43 R   19-   512.00000
C   43 R   21-     8.00000
C   43 R   22-    16.00000
C   43 R   23-    32.00000
C   43 R   24-    64.00000
C   44 OFSET       1.28130
C   44 R    1      1.0
C   44 R    2      2.00000
C   44 R    3      4.00000
C   44 R    4      8.00000
C   44 R    6      4.00000
C   44 R    7      8.00000
C   44 R    8     16.00000
C   44 R    9     32.00000
C   44 R   11     16.00000
C   44 R   12     32.00000
C   44 R   13     64.00000
C   44 R   14    128.00000
C   44 R   16     64.00000
C   44 R   17    128.00000
C   44 R   18    256.00000
C   44 R   19    512.00000
C   44 R   21      8.00000
C   44 R   22     16.00000
C   44 R   23     32.00000
C   44 R   24     64.00000
C   45 OFSET-      2.68750
C   45 R    1      1.0
C   45 R    2-     3.00000
C   45 R    3-     9.00000
```

```
C  45 R   4-    27.00000
C  45 R   5-     1.00000
C  45 R   6-     4.00000
C  45 R   7-    12.00000
C  45 R   8-    36.00000
C  45 R   9-   108.00000
C  45 R  10-     4.00000
C  45 R  11-    16.00000
C  45 R  12-    48.00000
C  45 R  13-   144.00000
C  45 R  14-   432.00000
C  45 R  15-    16.00000
C  45 R  16-    64.00000
C  45 R  17-   192.00000
C  45 R  18-   576.00000
C  45 R  19-  1728.00000
C  45 R  20-    64.00000
C  45 R  21-     8.00000
C  45 R  22-    24.00000
C  45 R  23-    72.00000
C  45 R  24-   216.00000
C  45 R  25-     8.00000
C  46 OFSET       2.68750
C  46 R   1      1.0
C  46 R   2      3.00000
C  46 R   3      9.00000
C  46 R   4     27.00000
C  46 R   5      1.00000
C  46 R   6      4.00000
C  46 R   7     12.00000
C  46 R   8     36.00000
C  46 R   9    108.00000
C  46 R  10      4.00000
C  46 R  11     16.00000
C  46 R  12     48.00000
C  46 R  13    144.00000
C  46 R  14    432.00000
C  46 R  15     16.00000
C  46 R  16     64.00000
C  46 R  17    192.00000
C  46 R  18    576.00000
```

1.0.0-2

```
C  46 R   19  1728.00000
C  46 R   20    64.00000
C  46 R   21     8.00000
C  46 R   22    24.00000
C  46 R   23    72.00000
C  46 R   24   216.00000
C  46 R   25     8.00000
C  47 OFSET-     4.63550
C  47 R    1     1.0
C  47 R    2-    4.00000
C  47 R    3-   16.00000
C  47 R    4-   64.00000
C  47 R    5-    8.00000
C  47 R    6-    4.00000
C  47 R    7-   16.00000
C  47 R    8-   64.00000
C  47 R    9-  256.00000
C  47 R   10-   32.00000
C  47 R   11-   16.00000
C  47 R   12-   64.00000
C  47 R   13-  256.00000
C  47 R   14- 1024.00000
C  47 R   15-  128.00000
C  47 R   16-   64.00000
C  47 R   17-  256.00000
C  47 R   18- 1024.00000
C  47 R   19- 4096.00000
C  47 R   20-  512.00000
C  47 R   21-    8.00000
C  47 R   22-   32.00000
C  47 R   23-  128.00000
C  47 R   24-  512.00000
C  47 R   25-   64.00000
C  48 OFSET      4.63550
C  48 R    1     1.0
C  48 R    2     4.00000
C  48 R    3    16.00000
C  48 R    4    64.00000
C  48 R    5     8.00000
C  48 R    6     4.00000
C  48 R    7    16.00000
```

C	48	R	8	64.00000
C	48	R	9	256.00000
C	48	R	10	32.00000
C	48	R	11	16.00000
C	49	R	12	64.00000
C	48	R	13	256.00000
C	48	R	14	1024.00000
C	48	R	15	128.00000
C	48	R	16	64.00000
C	48	R	17	256.00000
C	48	R	18	1024.00000
C	48	R	19	4096.00000
C	48	R	20	512.00000
C	48	R	21	8.00000
C	48	R	22	32.00000
C	48	R	23	128.00000
C	48	R	24	512.00000
C	48	R	25	64.00000
C	49	OFSET		0.00000
C	49	R	3-	1.00000
C	49	R	4-	6.00000
C	50	OFSET		0.00000
C	50	R	3-	1.00000
C	50	R	4-	12.00000
C	50	R	5-	6.00000
C	51	OFSET		0.00000
C	51	R	3-	1.00000
C	51	R	4-	6.00000
C	51	R	8-	1.00000
C	51	R	9-	6.00000
C	51	R	13-	1.00000
C	51	R	14-	6.00000
C	51	R	18-	1.00000
C	51	R	19-	6.00000
C	52	OFSET		0.00000
C	52	R	3-	1.00000
C	52	R	4-	12.00000
C	52	R	5-	6.00000
C	52	R	8-	1.00000
C	52	R	9-	12.00000
C	52	R	10-	6.00000

1.0.0-2

```
C   52 R   13-     1.00000
C   52 R   14-    12.00000
C   52 R   15-     6.00000
C   52 R   18-     1.00000
C   52 R   19-    12.00000
C   52 R   20-     6.00000
C   53 OFSET       0.00000
C   53 R    3-     1.00000
C   53 R    4-     6.00000
C   53 R    8-     2.00000
C   53 R    9-    12.00000
C   53 R   13-     4.00000
C   53 R   14-    24.00000
C   53 R   18-     8.00000
C   53 R   19-    48.00000
C   54 OFSET       0.00000
C   54 R    3-     1.00000
C   54 R    4-    12.00000
C   54 R    5-     6.00000
C   54 R    8-     2.00000
C   54 R    9-    24.00000
C   54 R   10-    12.00000
C   54 R   13-     4.00000
C   54 R   14-    48.00000
C   54 R   15-    24.00000
C   54 R   18-     8.00000
C   54 R   19-    96.00000
C   54 R   20-    48.00000
C   55 OFSET       0.00000
C   55 R    3-     1.00000
C   55 R    4-     6.00000
C   55 R    8-     3.00000
C   55 R    9-    18.00000
C   55 R   13-     9.00000
C   55 R   14-    54.00000
C   55 R   18-    27.00000
C   55 R   19-   162.00000
C   55 R   23-     1.00000
C   55 R   24-     6.00000
C   56 OFSET       0.00000
C   56 R    3-     1.00000
```

```
C  56 R    4-    12.00000
C  56 R    5-     6.00000
C  56 R    8-     3.00000
C  56 R    9-    36.00000
C  56 R   10-    18.00000
C  56 R   13-     9.00000
C  56 R   14-   108.00000
C  56 R   15-    54.00000
C  56 R   18-    27.00000
C  56 R   19-   324.00000
C  56 R   20-   162.00000
C  56 R   23-     1.00000
C  56 R   24-    12.00000
C  56 R   25-     6.00000
C  57 OFSET       0.00000
C  57 R    3-     1.00000
C  57 R    4-     6.00000
C  57 R    8-     4.00000
C  57 R    9-    24.00000
C  57 R   13-    16.00000
C  57 R   14-    96.00000
C  57 R   18-    64.00000
C  57 R   19-   384.00000
C  57 R   23-     8.00000
C  57 R   24-    48.00000
C  58 OFSET       0.00000
C  58 R    3-     1.00000
C  58 R    4-    12.00000
C  58 R    5-     6.00000
C  58 R    8-     4.00000
C  58 R    9-    48.00000
C  58 R   10-    24.00000
C  58 R   13-    16.00000
C  58 R   14-   192.00000
C  58 R   15-    96.00000
C  58 R   18-    64.00000
C  58 R   19-   768.00000
C  58 R   20-   384.00000
C  58 R   23-     8.00000
C  58 R   24-    96.00000
C  58 R   25-    48.00000
```

```
C  59 OFSET     0.00000
C  59 R   11-    1.00000
C  59 R   16-    6.00000
C  60 OFSET     0.00000
C  60 R   11-    1.00000
C  60 R   12-    1.00000
C  60 R   13-    1.00000
C  60 R   14-    1.00000
C  60 R   16-    6.00000
C  60 R   17-    6.00000
C  60 R   18-    6.00000
C  60 R   19-    6.00000
C  61 OFSET     0.00000
C  61 R   11-    1.00000
C  61 R   12-    2.00000
C  61 R   13-    4.00000
C  61 R   14-    8.00000
C  61 R   16-    6.00000
C  61 R   17-   12.00000
C  61 R   18-   24.00000
C  61 R   19-   48.00000
C  62 OFSET     0.00000
C  62 R   11-    1.00000
C  62 R   12-    3.00000
C  62 R   13-    9.00000
C  62 R   14-   27.00000
C  62 R   15-    1.00000
C  62 R   16-    6.00000
C  62 R   17-   18.00000
C  62 R   18-   54.00000
C  62 R   19-  162.00000
C  62 R   20-    6.00000
C  63 OFSET     0.00000
C  63 R   11-    1.00000
C  63 R   12-    4.00000
C  63 R   13-   16.00000
C  63 R   14-   64.00000
C  63 R   15-    8.00000
C  63 R   16-    6.00000
C  63 R   17-   24.00000
C  63 R   18-   96.00000
```

```
C  63 R   19-   384.00000
C  63 R   20-    48.00000
C  64 OFSET       0.00000
C  64 R   11-     1.00000
C  64 R   16-    12.00000
C  64 R   21-     6.00000
C  65 OFSET       0.00000
C  65 R   11-     1.00000
C  65 R   12-     1.00000
C  65 R   13-     1.00000
C  65 R   14-     1.00000
C  65 R   16-    12.00000
C  65 R   17-    12.00000
C  65 R   18-    12.00000
C  65 R   19-    12.00000
C  65 R   21-     6.00000
C  65 R   22-     6.00000
C  65 R   23-     6.00000
C  65 R   24-     6.00000
C  66 OFSET       0.00000
C  66 R   11-     1.00000
C  66 R   12-     2.00000
C  66 R   13-     4.00000
C  66 R   14-     8.00000
C  66 R   16-    12.00000
C  66 R   17-    24.00000
C  66 R   18-    48.00000
C  66 R   19-    96.00000
C  66 R   21-     6.00000
C  66 R   22-    12.00000
C  66 R   23-    24.00000
C  66 R   24-    48.00000
C  67 OFSET       0.00000
C  67 R   11-     1.00000
C  67 R   12-     3.00000
C  67 R   13-     9.00000
C  67 R   14-    27.00000
C  67 R   15-     1.00000
C  67 R   16-    12.00000
C  67 R   17-    36.00000
C  67 R   18-   108.00000
```

1.0.0-2

1.0.0.2

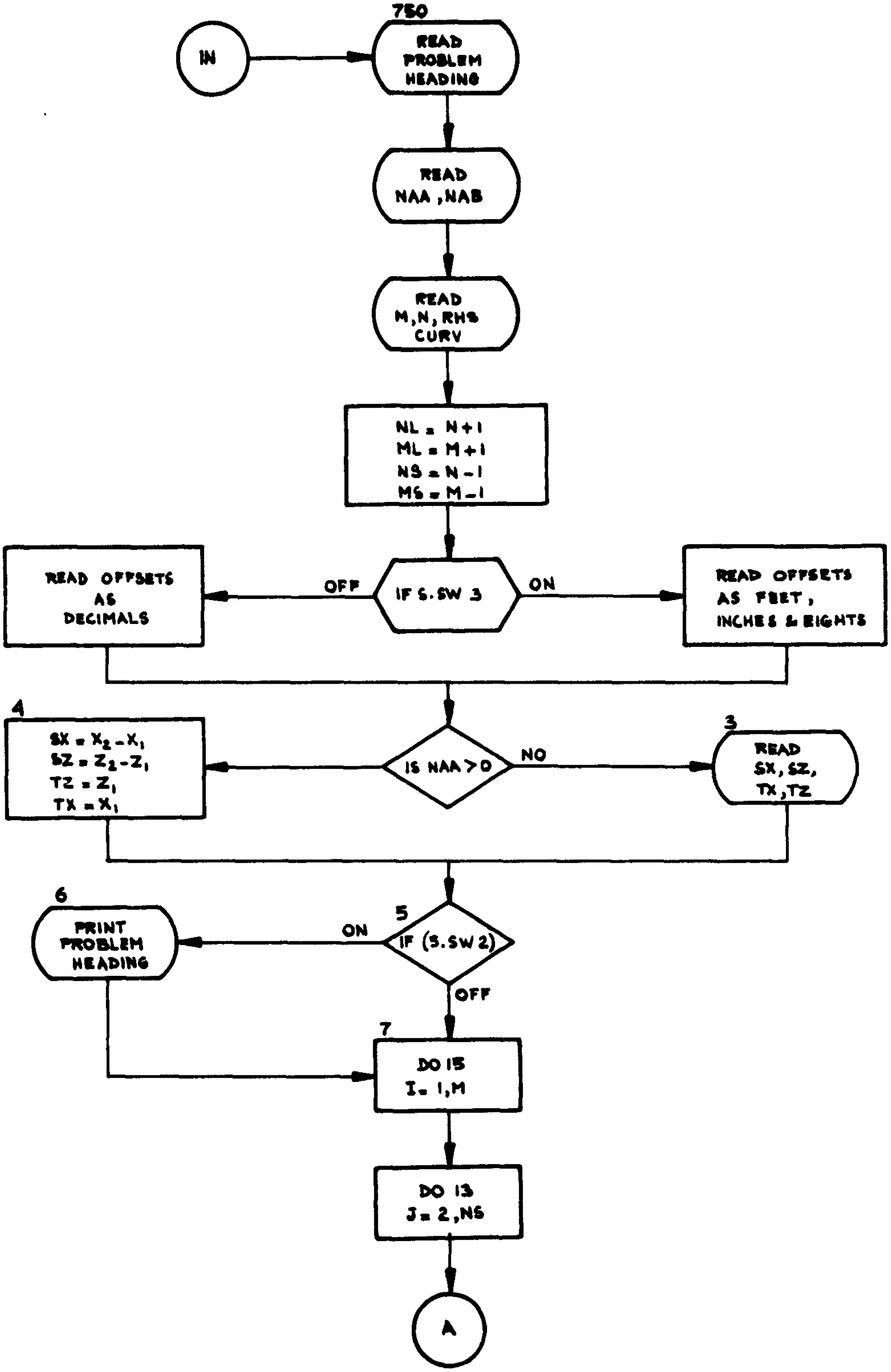
```
     C  67 R   19-   324.00000
     C  67 R   20-    12.00000
     C  67 R   21-     6.00000
     C  67 R   22-    18.00000
     C  67 R   23-    54.00000
     C  67 R   24-   162.00000
     C  67 R   25-     6.00000
     C  68 OFSET       0.00000
     C  68 R   11-     1.00000
     C  68 R   12-     4.00000
     C  68 R   13-    16.00000
     C  68 R   14-    64.00000
     C  68 R   15-     8.00000
     C  68 R   16-    12.00000
     C  68 R   17-    48.00000
     C  68 R   18-   192.00000
     C  68 R   19-   768.00000
     C  68 R   20-    96.00000
     C  68 R   21-     6.00000
     C  68 R   22-    24.00000
     C  68 R   23-    96.00000
     C  68 R   24-   384.00000
     C  68 R   25-    48.00000
FIRST B
           R    1      2.0
           R    2       .00010
           R    3       .00010
           R    4       .00010
           R    5       .00010
           R    6       .00010
           R    7       .00010
           R    8       .00010
           R    9       .00010
           R   10       .00010
           R   11       .00010
           R   12       .00010
           R   13       .00010
           R   14       .00010
           R   15       .00010
           R   16       .00010
           R   17       .00010
```

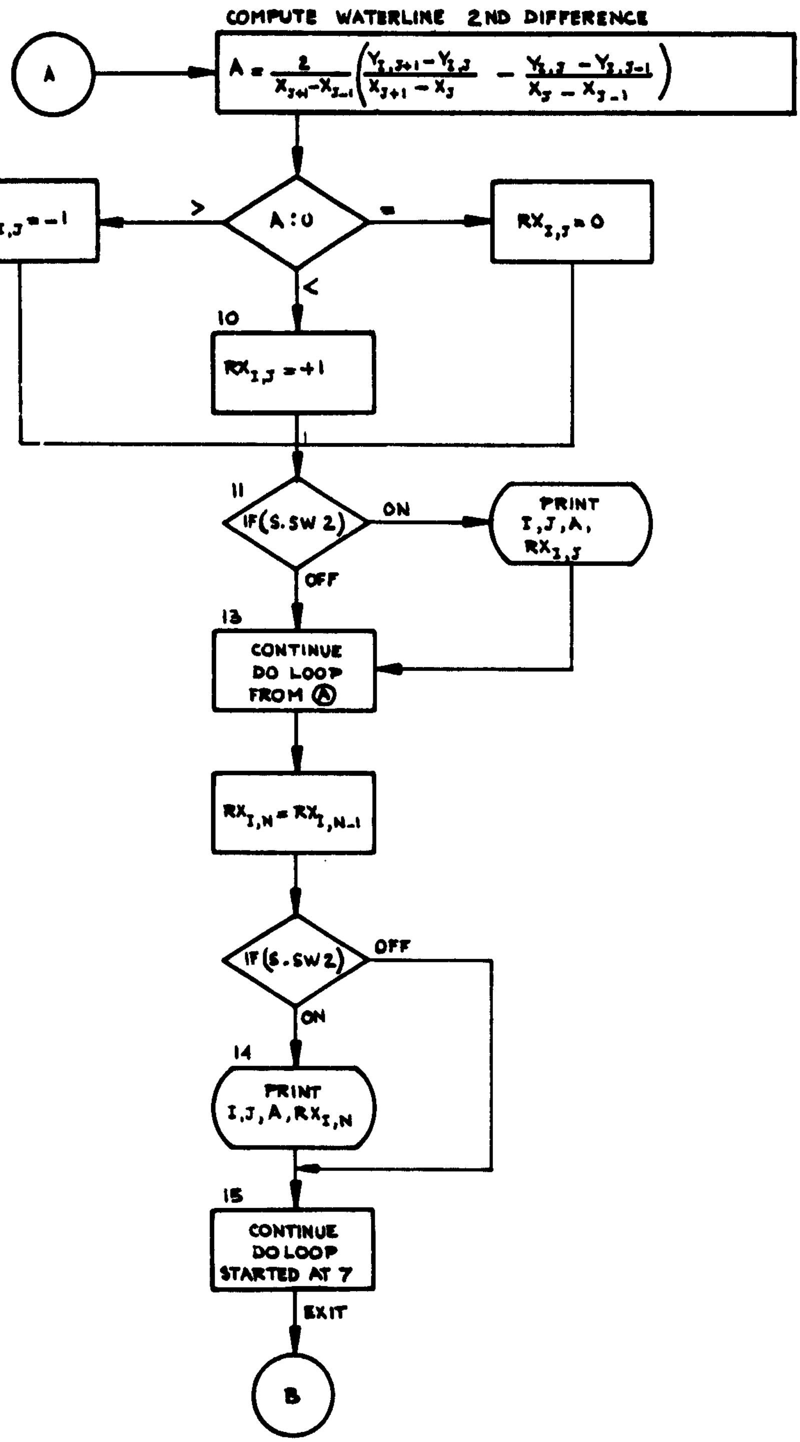
1,0.0-2

R	18	.00010
R	19	.00010
R	20	.00010
R	21	.00010
R	22	.00010
R	23	.00010
R	24	.00010
R	25	.00010

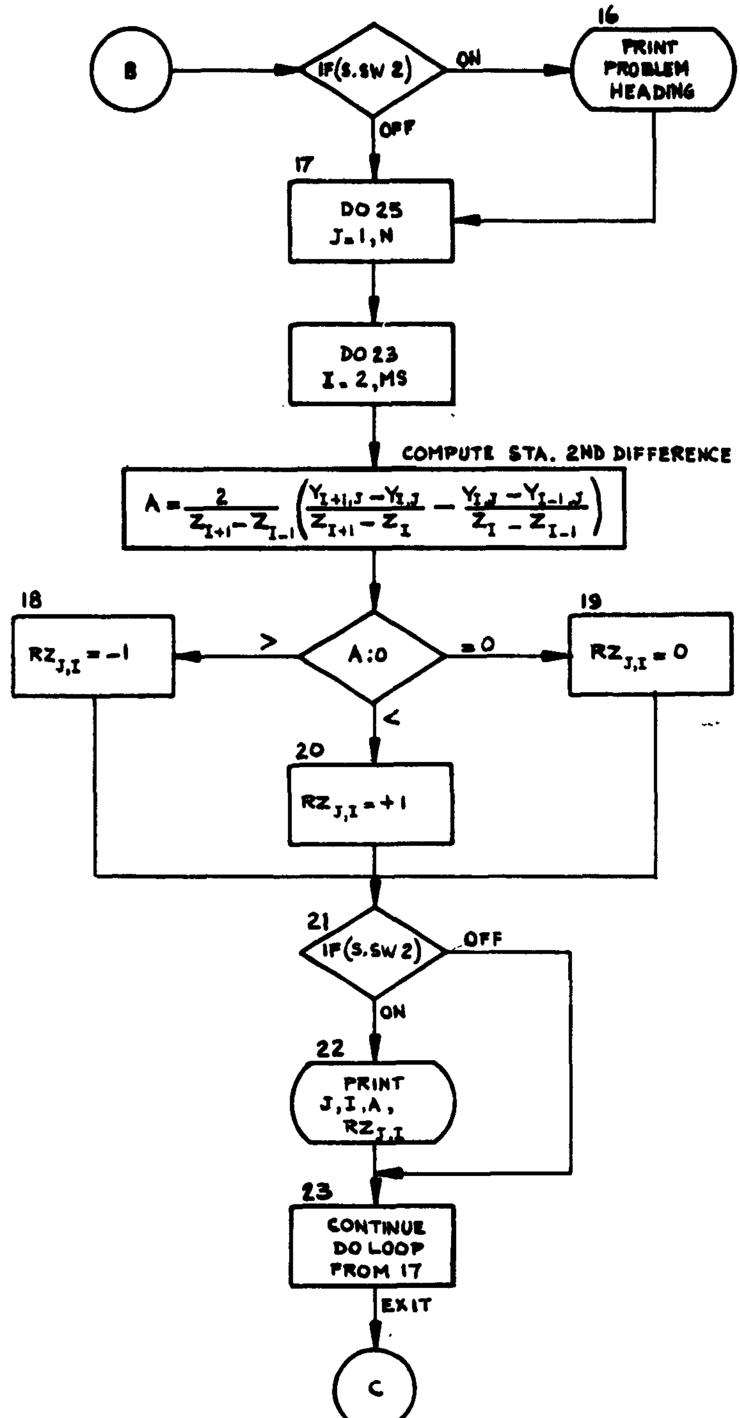
EOF

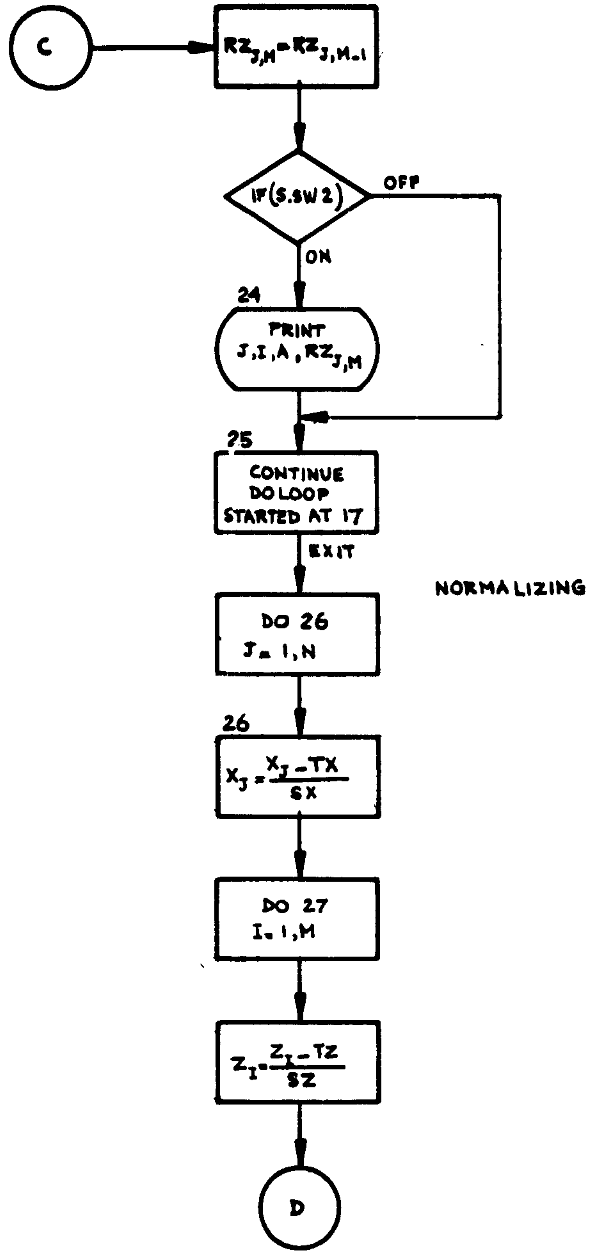
FLOW DIAGRAM FOR SMOG 1

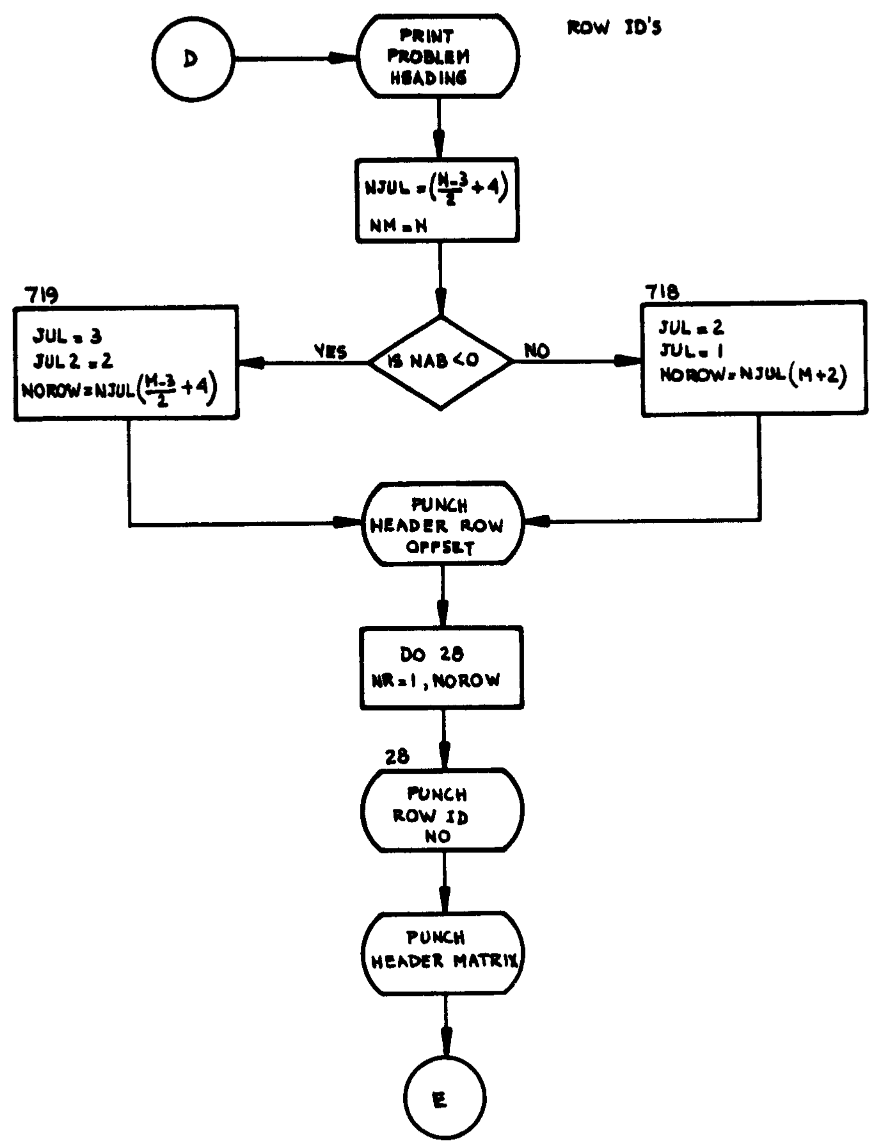




COMPUTE WATERLINE 2ND DIFFERENCE
A
$A = \frac{2}{X_{J+1}-X_{J-1}}\left(\frac{Y_{I,J+1}-Y_{I,J}}{X_{J+1}-X_{J}} - \frac{Y_{I,J}-Y_{I,J-1}}{X_{J}-X_{J-1}}\right)$
A : 0
>
=
<
$RX_{I,J} = -1$
$RX_{I,J} = 0$
10
$RX_{I,J} = +1$
11
IF(S.SW 2)
ON
OFF
PRINT I,J,A, $RX_{I,J}$
13
CONTINUE DO LOOP FROM A
$RX_{I,N} = RX_{I,N-1}$
IF(S.SW 2)
OFF
ON
14
PRINT I,J,A,$RX_{I,N}$
15
CONTINUE DO LOOP STARTED AT 7
EXIT
B

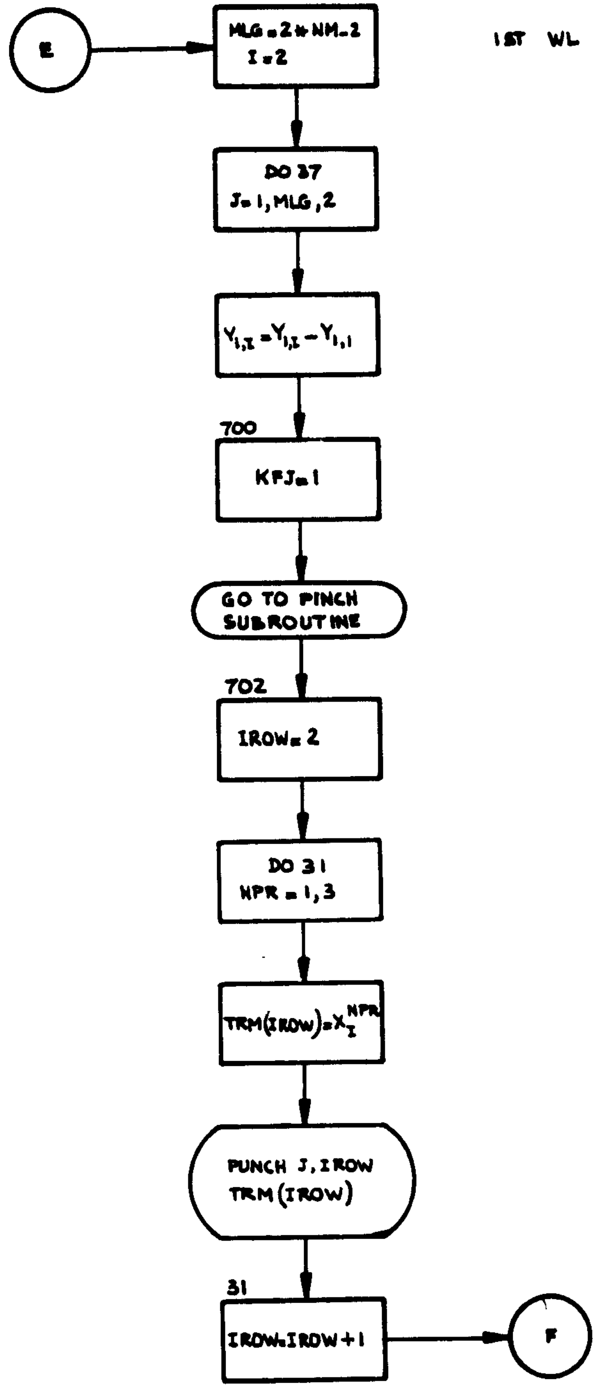

B
IF(S.SW 2)
ON
OFF
16
PRINT PROBLEM HEADING
17
DO 25 J=1,N
DO 23 I=2,MS
COMPUTE STA. 2ND DIFFERENCE
$A = \frac{2}{Z_{I+1}-Z_{I-1}}\left(\frac{Y_{I+1,J}-Y_{I,J}}{Z_{I+1}-Z_I} - \frac{Y_{I,J}-Y_{I-1,J}}{Z_I - Z_{I-1}}\right)$
18
$RZ_{J,I} = -1$
>
A:0
=0
19
$RZ_{J,I} = 0$
<
20
$RZ_{J,I} = +1$
21
IF(S.SW 2)
OFF
ON
22
PRINT J,I,A, $RZ_{J,I}$
23
CONTINUE DO LOOP FROM 17
EXIT
C

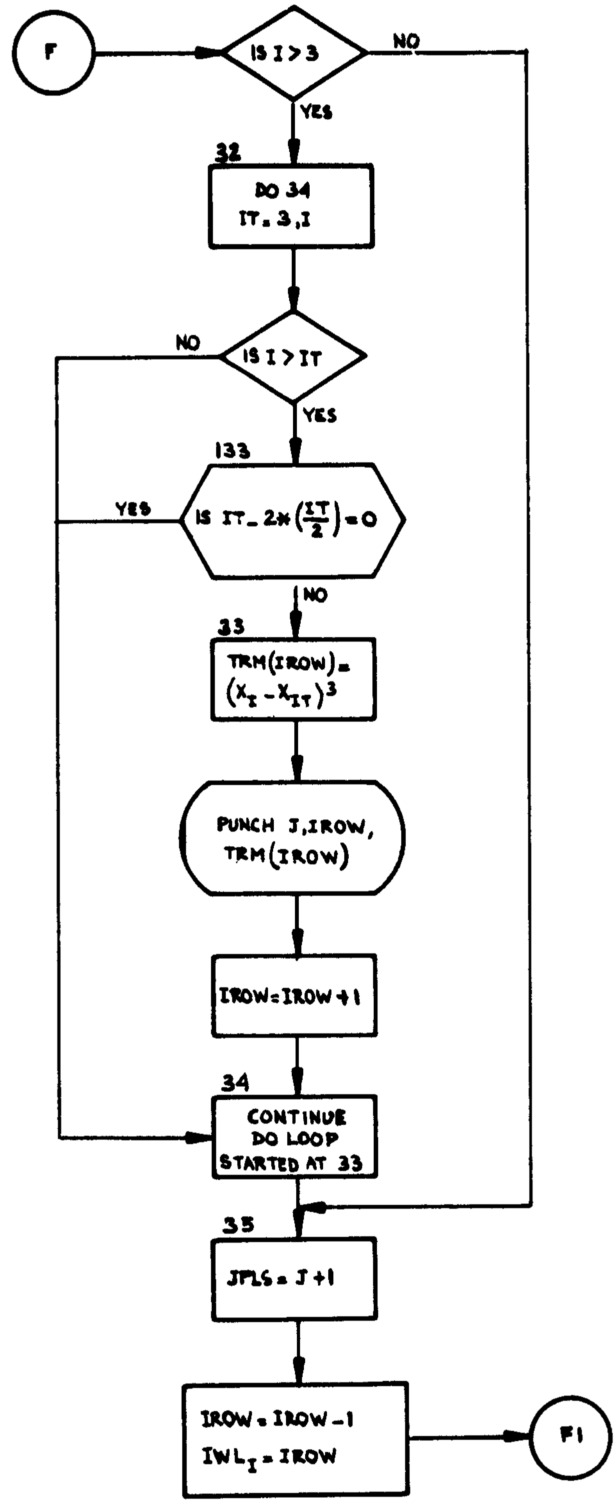

C
RZJ,M = RZJ,M-1
IF (S.SW 2)
OFF
ON
24
PRINT
J, I, A, RZJ,M
25
CONTINUE
DO LOOP
STARTED AT 17
EXIT
NORMALIZING
DO 26
J = 1, N
26
XJ = (XJ - TX) / SX
DO 27
I = 1, M
ZI = (ZI - TZ) / SZ
D

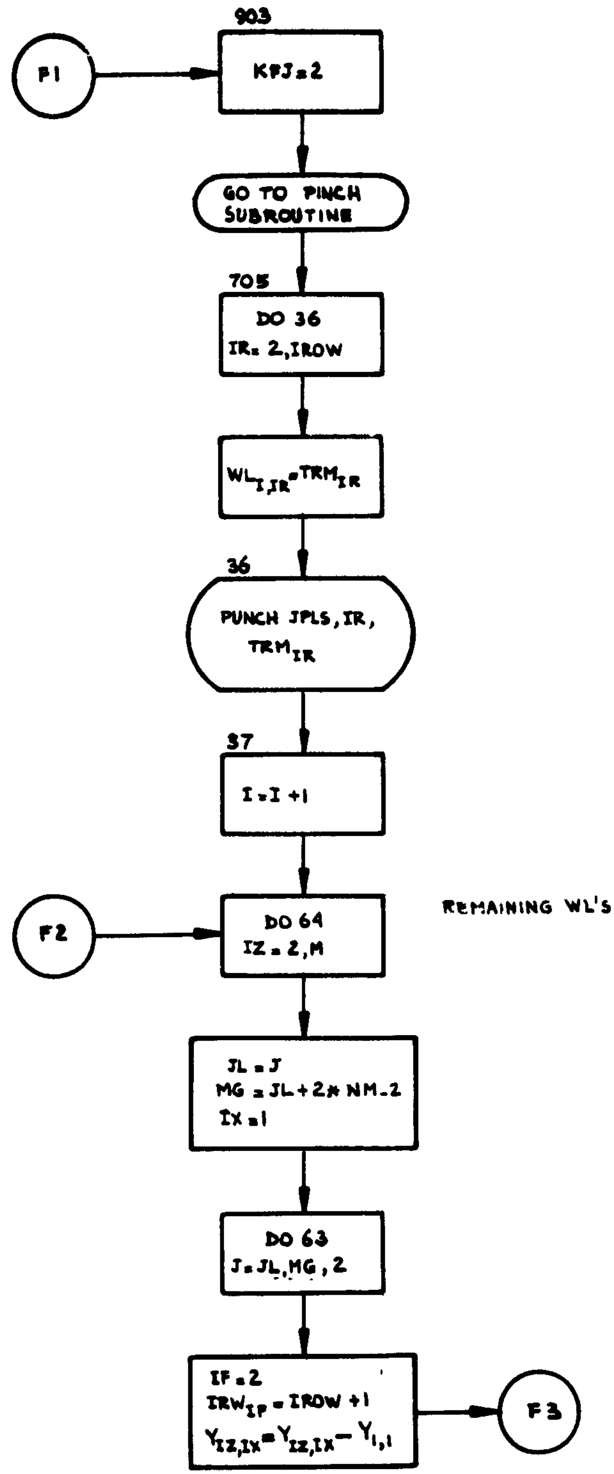


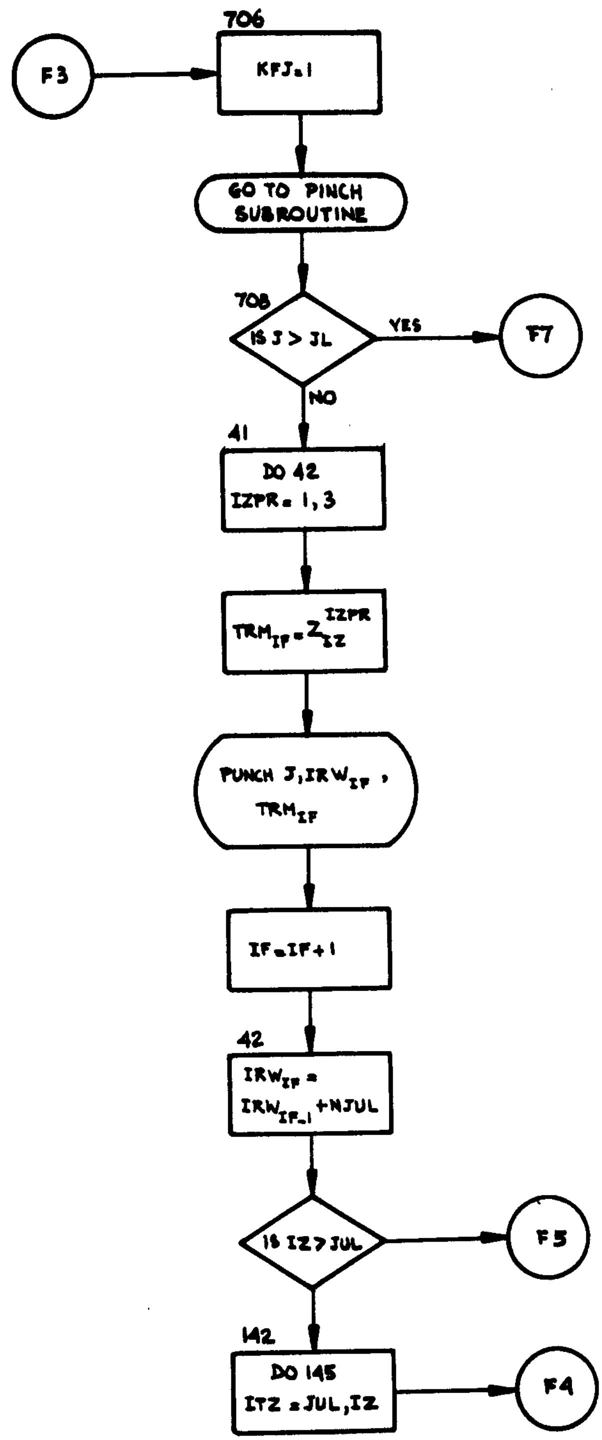
ROW ID'S
D
PRINT PROBLEM HEADING
NJUL = ((M-3)/2 + 4)
NM = N
IS NAB < 0
YES
NO
719
JUL = 3
JUL 2 = 2
NOROW = NJUL((M-3)/2 + 4)
718
JUL = 2
JUL = 1
NOROW = NJUL(M+2)
PUNCH HEADER ROW OFFSET
DO 28 NR = 1, NOROW
28
PUNCH ROW ID NO
PUNCH HEADER MATRIX
E

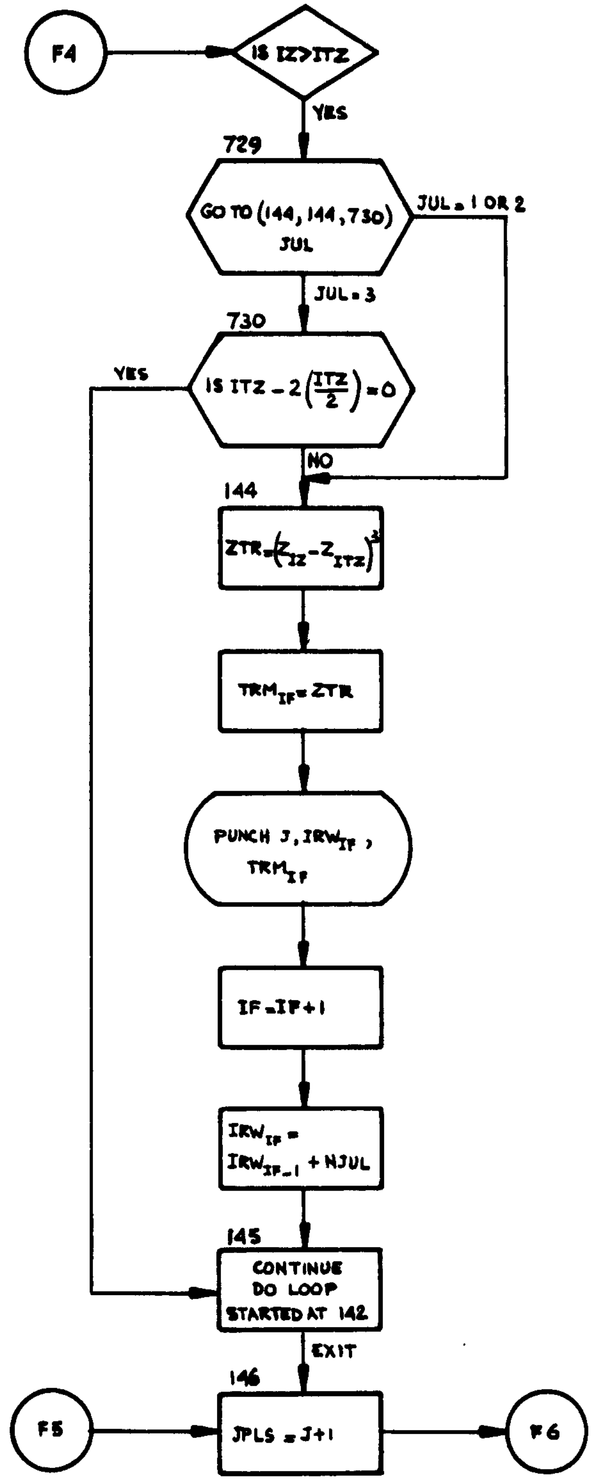
1ST WL

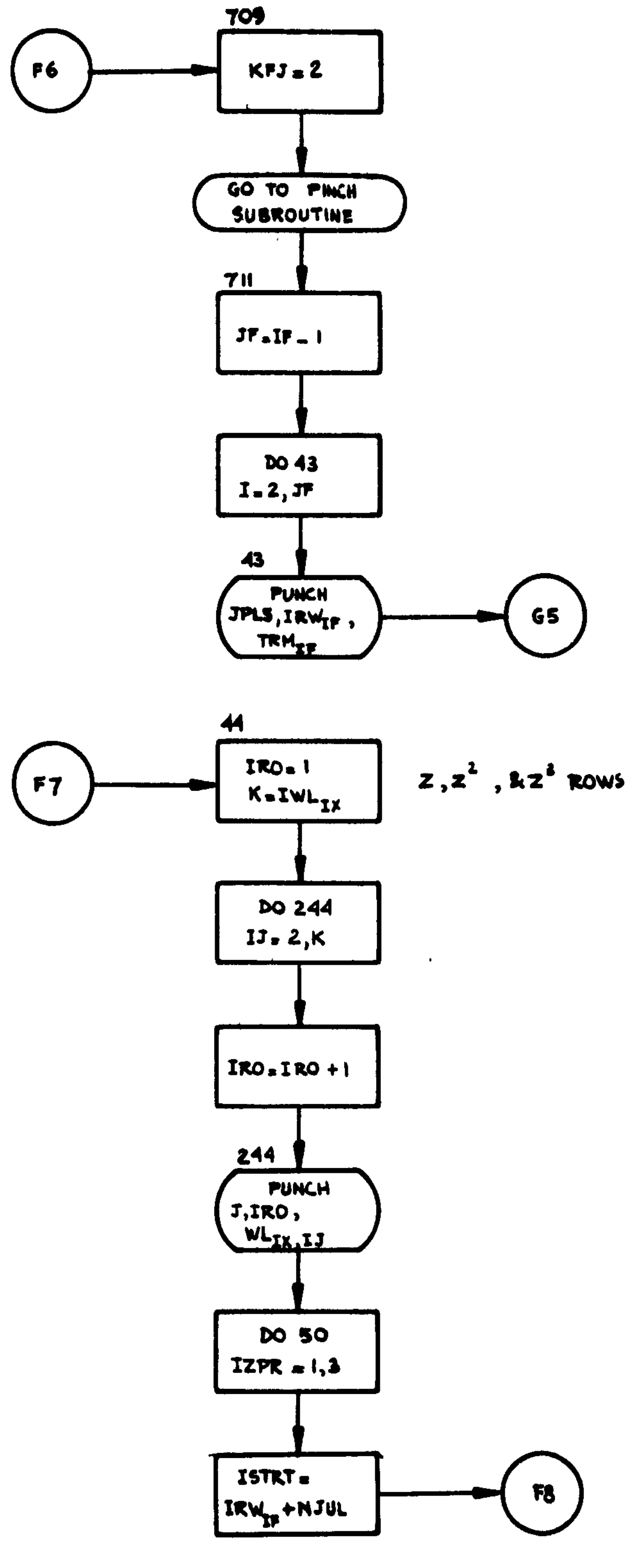

E
MLG = 2 * NM-2
I = 2
DO 37
J = 1, MLG, 2
Y1,I = Y1,I - Y1,1
700
KFJ = 1
GO TO PINCH SUBROUTINE
702
IROW = 2
DO 31
NPR = 1, 3
TRM(IROW) = X_I^NPR
PUNCH J, IROW
TRM(IROW)
31
IROW = IROW + 1
F

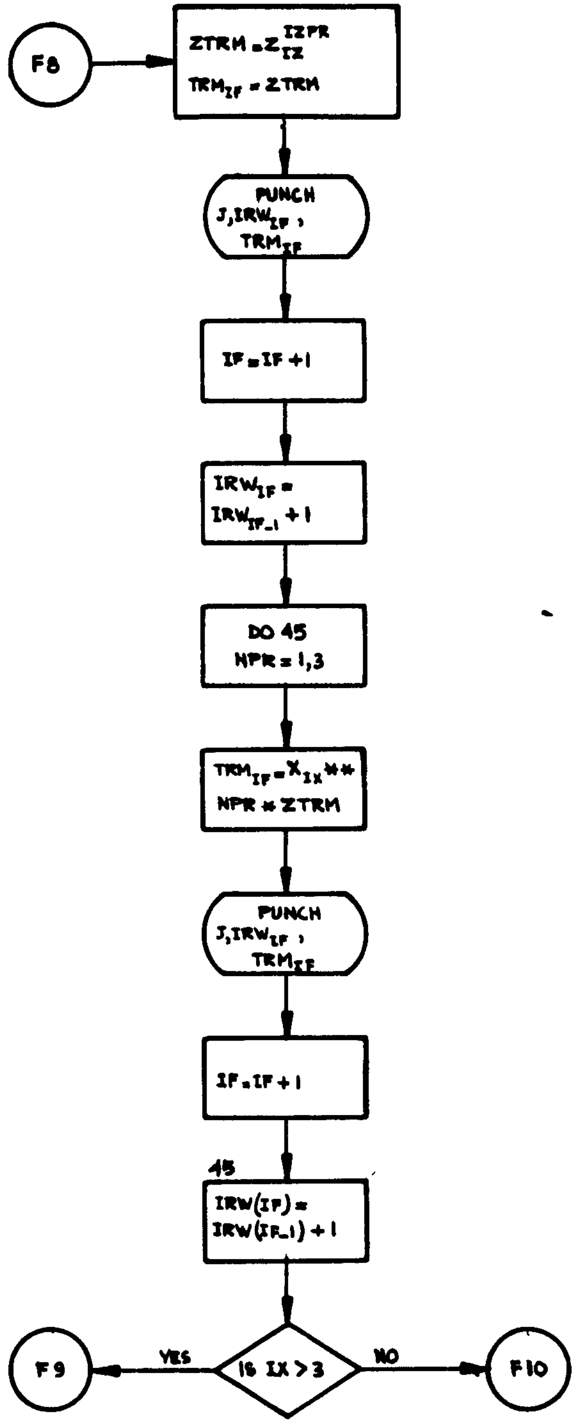

F
IS I > 3
NO
YES
32
DO 34
IT = 3,I
NO
IS I > IT
YES
133
YES
IS IT - 2*(IT/2) = 0
NO
33
TRM(IROW) = $(X_I - X_{IT})^3$
PUNCH J,IROW, TRM(IROW)
IROW = IROW + 1
34
CONTINUE DO LOOP STARTED AT 33
35
JPLS = J + 1
IROW = IROW - 1
IWL_I = IROW
F1

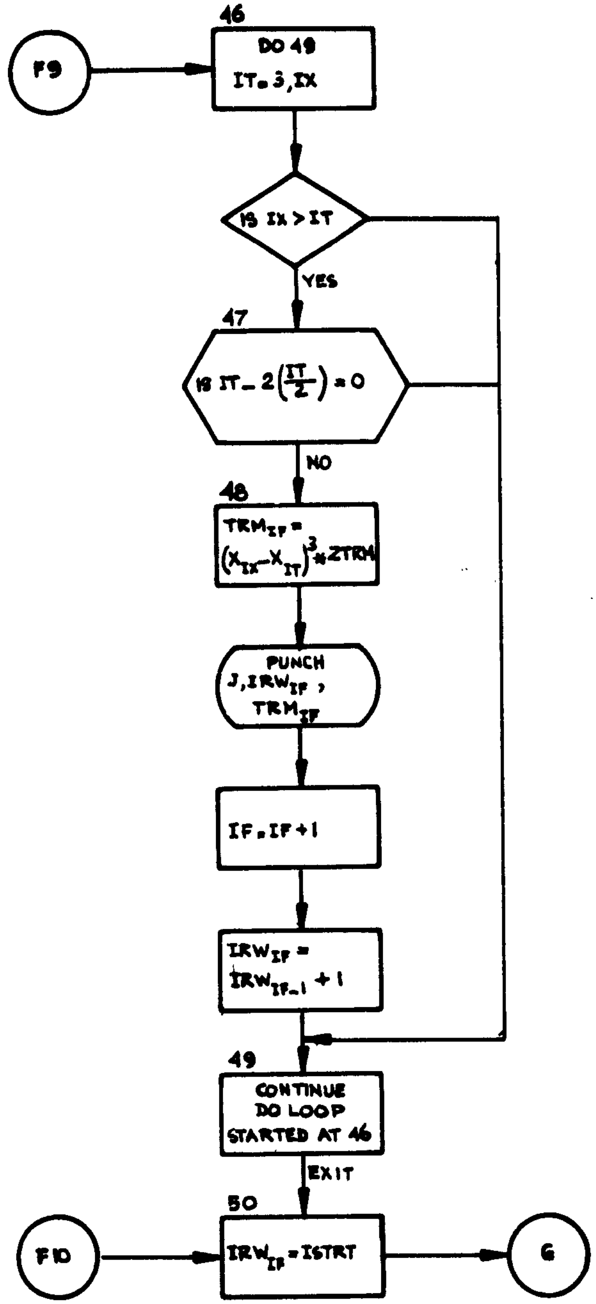

F1
903
KFJ=2
GO TO PINCH SUBROUTINE
705
DO 36
IR= 2,IROW
WL_{I,IR}=TRM_{IR}
36
PUNCH JPLS,IR, TRM_{IR}
37
I=I+1
F2
DO 64
IZ=2,M
REMAINING WL'S
JL=J
MG=JL+2*NM-2
IX=1
DO 63
J=JL,MG,2
IF=2
IRW_{IF}=IROW+1
Y_{IZ,IX}=Y_{IZ,IX}-Y_{1,1}
F3


706
F3
KFJ=1
GO TO PINCH SUBROUTINE
708
IS J > JL
YES
F7
NO
41
DO 42
IZPR=1,3
TRM$_{IF}$=Z$_{IZ}^{IZPR}$
PUNCH J, IRW$_{IF}$,
TRM$_{IF}$
IF=IF+1
42
IRW$_{IF}$=
IRW$_{IF-1}$+NJUL
IS IZ > JUL
F5
142
DO 145
ITZ=JUL,IZ
F4

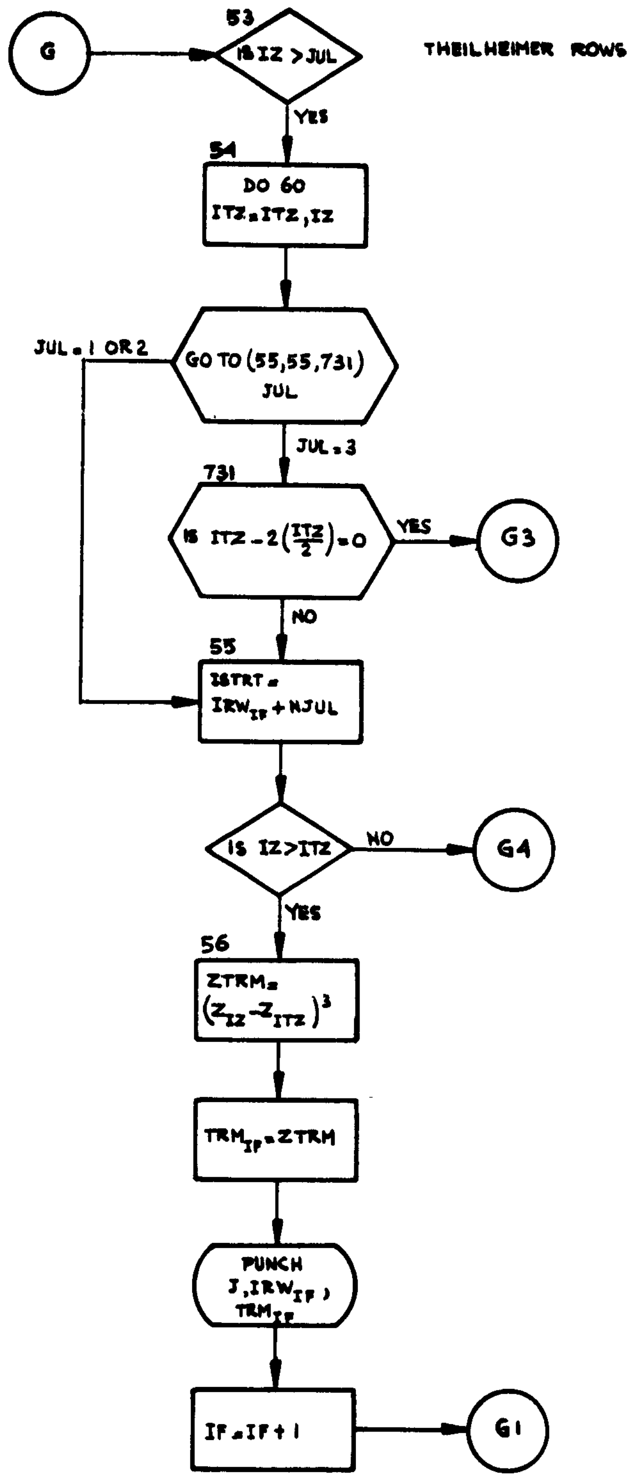

F4
IS IZ>ITZ
YES
729
GO TO (144,144,730) JUL
JUL = 1 OR 2
JUL = 3
730
IS ITZ − 2(ITZ/2) = 0
YES
NO
144
ZTR = (Z_IZ − Z_ITZ)²
TRM_IF = ZTR
PUNCH J, IRW_IF, TRM_IF
IF = IF+1
IRW_IF = IRW_IF−1 + NJUL
145
CONTINUE DO LOOP STARTED AT 142
EXIT
146
F5
JPLS = J+1
F6


709
F6
KFJ = 2
GO TO PINCH SUBROUTINE
711
JF = IF - 1
DO 43
I = 2, JF
43
PUNCH
JPLS, IRW_IF,
TRM_IF
G5
44
F7
IRO = 1
K = IWL_IX
Z, Z^2, & Z^3 ROWS
DO 244
IJ = 2, K
IRO = IRO + 1
244
PUNCH
J, IRO,
WL_IX, IJ
DO 50
IZPR = 1, 3
ISTRT =
IRW_IF + NJUL
F8

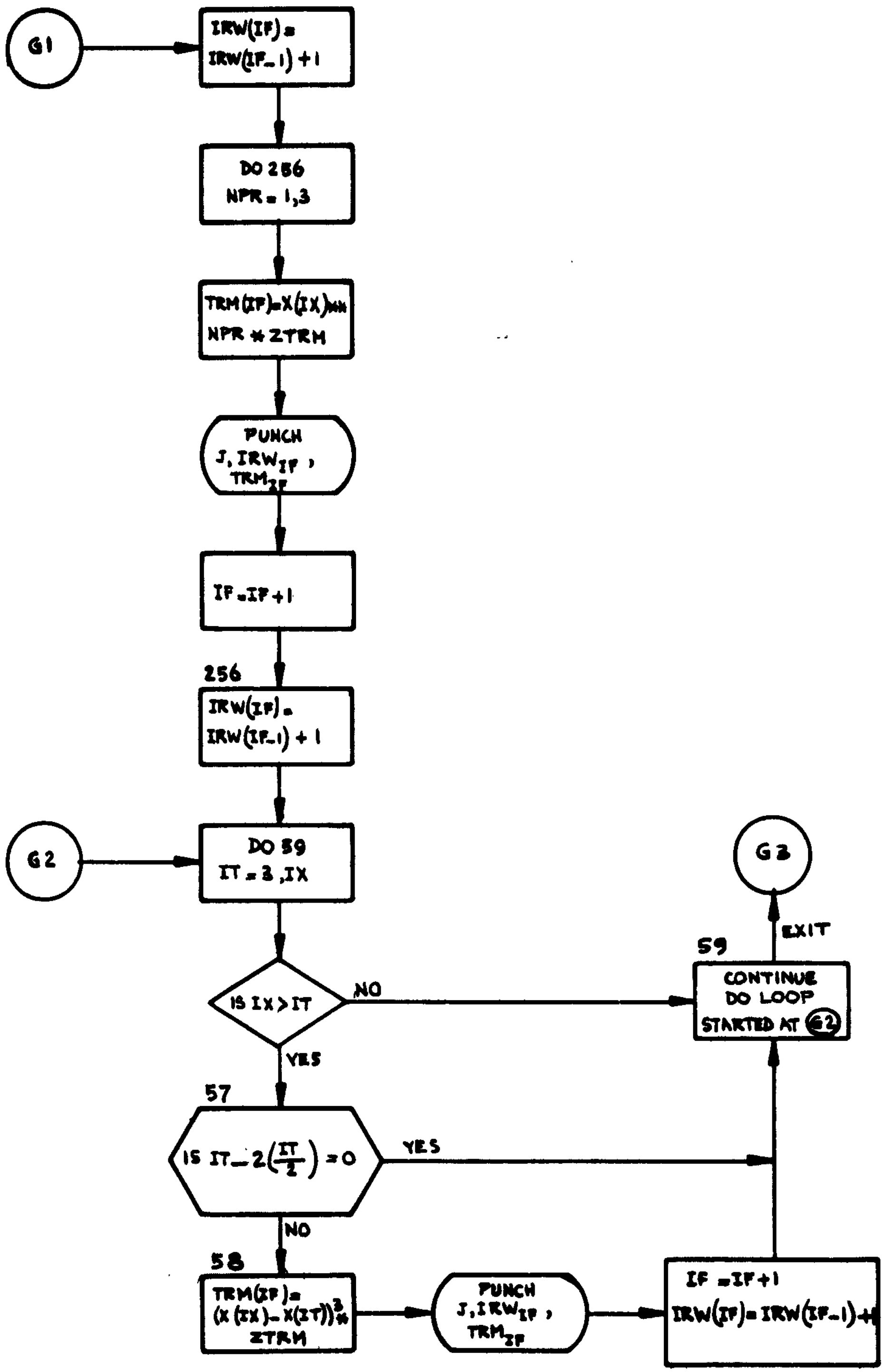

F8
ZTRM = Z IX IZPR
TRM IF = ZTRM
PUNCH J, IRW IF, TRM IF
IF = IF + 1
IRW IF = IRW IF-1 + 1
DO 45 NPR = 1,3
TRM IF = X IX ** NPR * ZTRM
PUNCH J, IRW IF, TRM IF
IF = IF + 1
45
IRW(IF) = IRW(IF-1) + 1
IS IX > 3
YES
NO
F9
F10

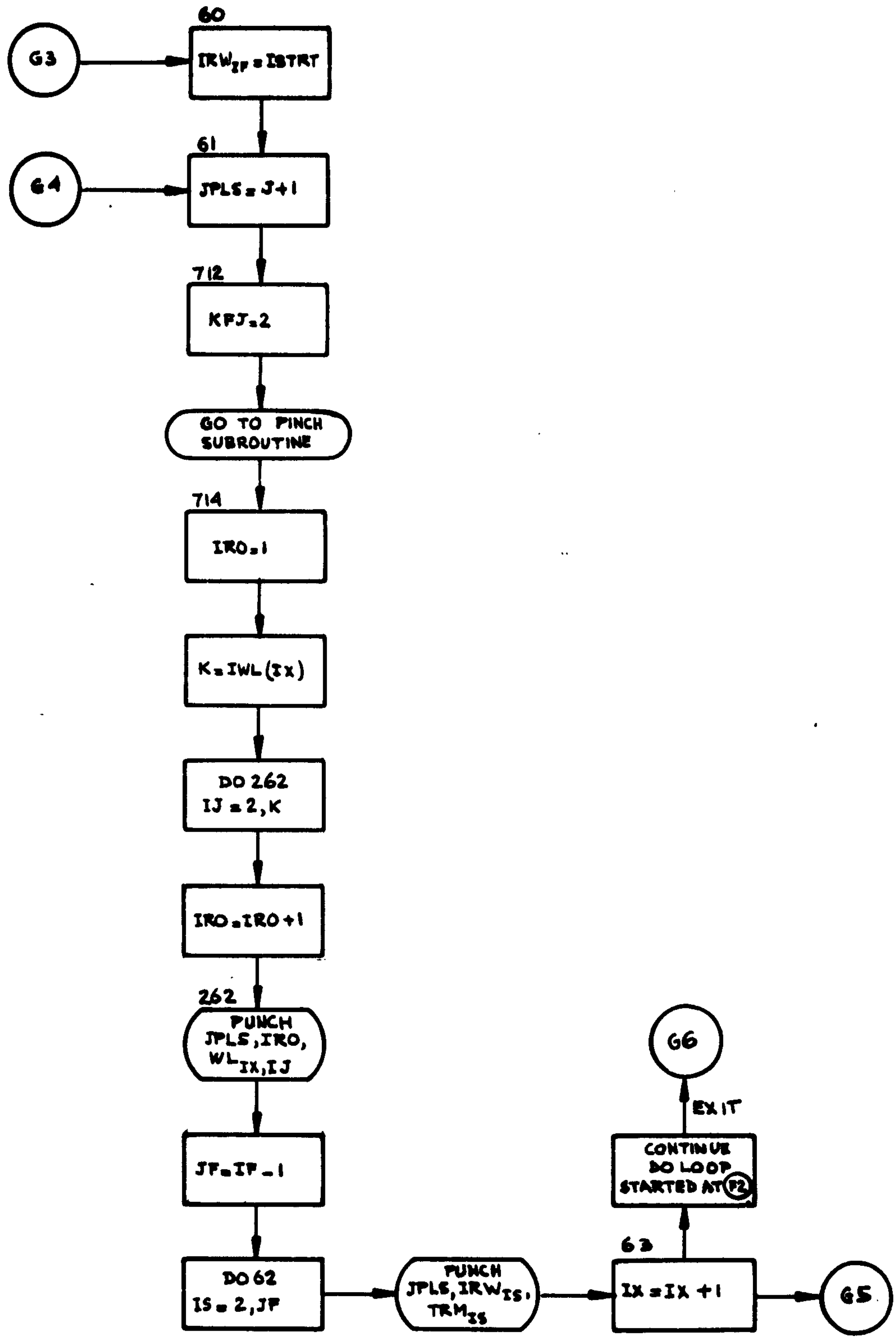


46
DO 49
IT = 3, IX
F9
IS IX > IT
YES
47
IS IT - 2(IT/2) = 0
NO
48
TRM_IF = (X_IX - X_IT)^3 * ZTRM
PUNCH J, IRW_IF, TRM_IF
IF = IF + 1
IRW_IF = IRW_IF-1 + 1
49
CONTINUE DO LOOP STARTED AT 46
EXIT
50
F10
IRW_IF = ISTRT
G

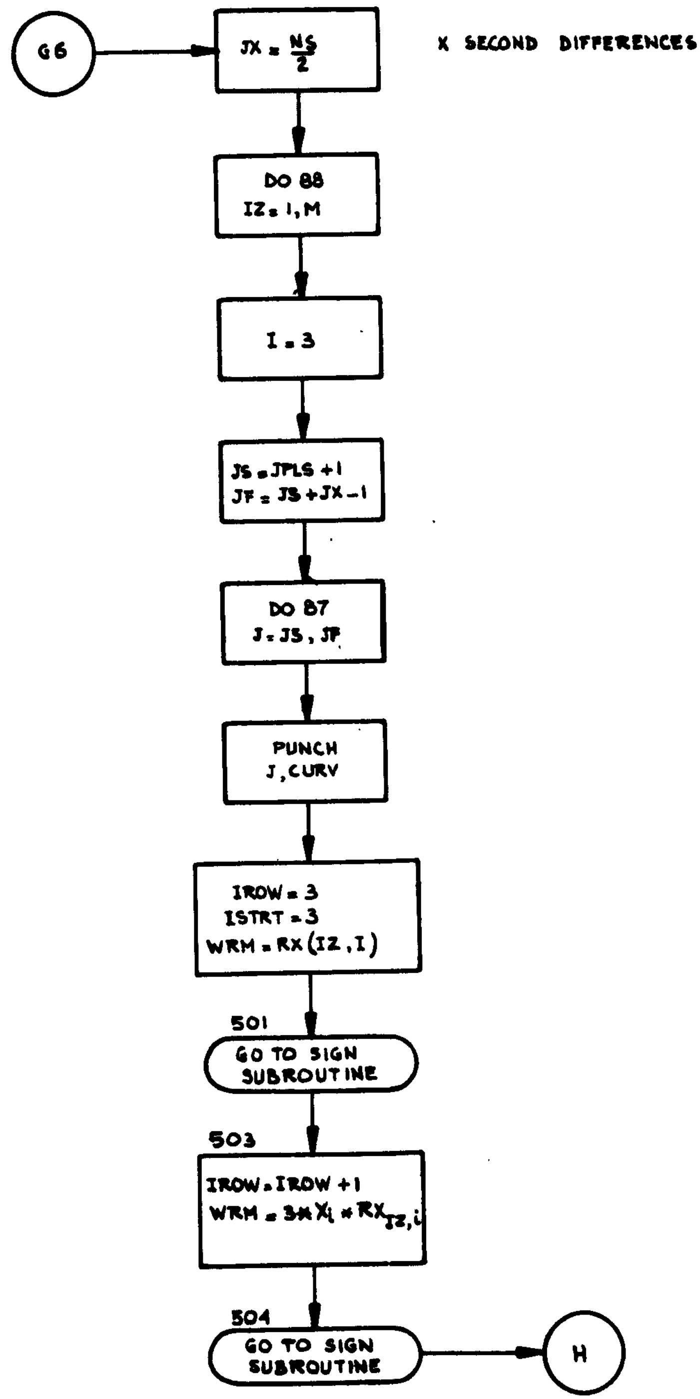


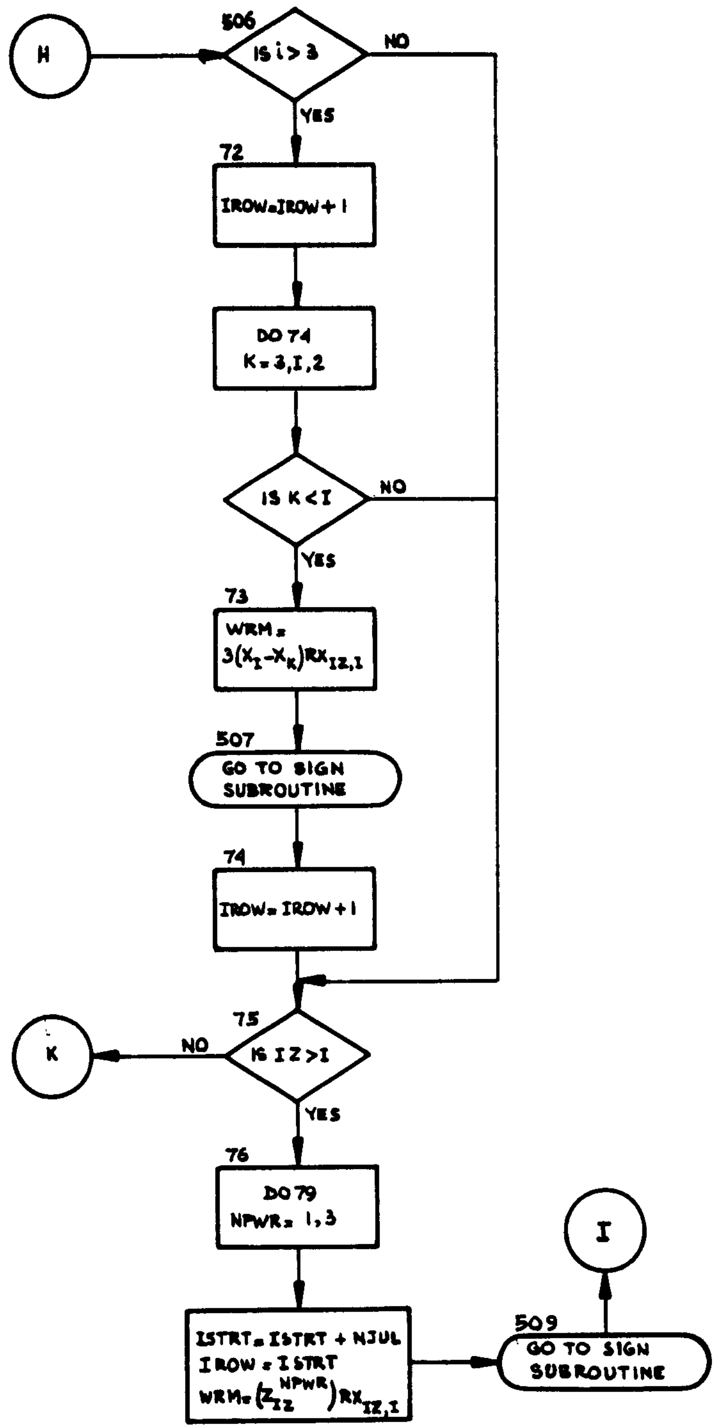
THEILHEIMER ROWS
G
53
IS IZ > JUL
YES
54
DO 60
ITZ=ITZ,IZ
GO TO (55,55,731)
JUL
JUL=1 OR 2
JUL=3
731
IS ITZ-2(ITZ/2)=0
YES
G3
NO
55
ISTRT=
IRW_IF+NJUL
IS IZ>ITZ
NO
G4
YES
56
ZTRM=
(Z_IZ-Z_ITZ)^3
TRM_IF=ZTRM
PUNCH
J,IRW_IF,
TRM_IF
IF=IF+1
G1

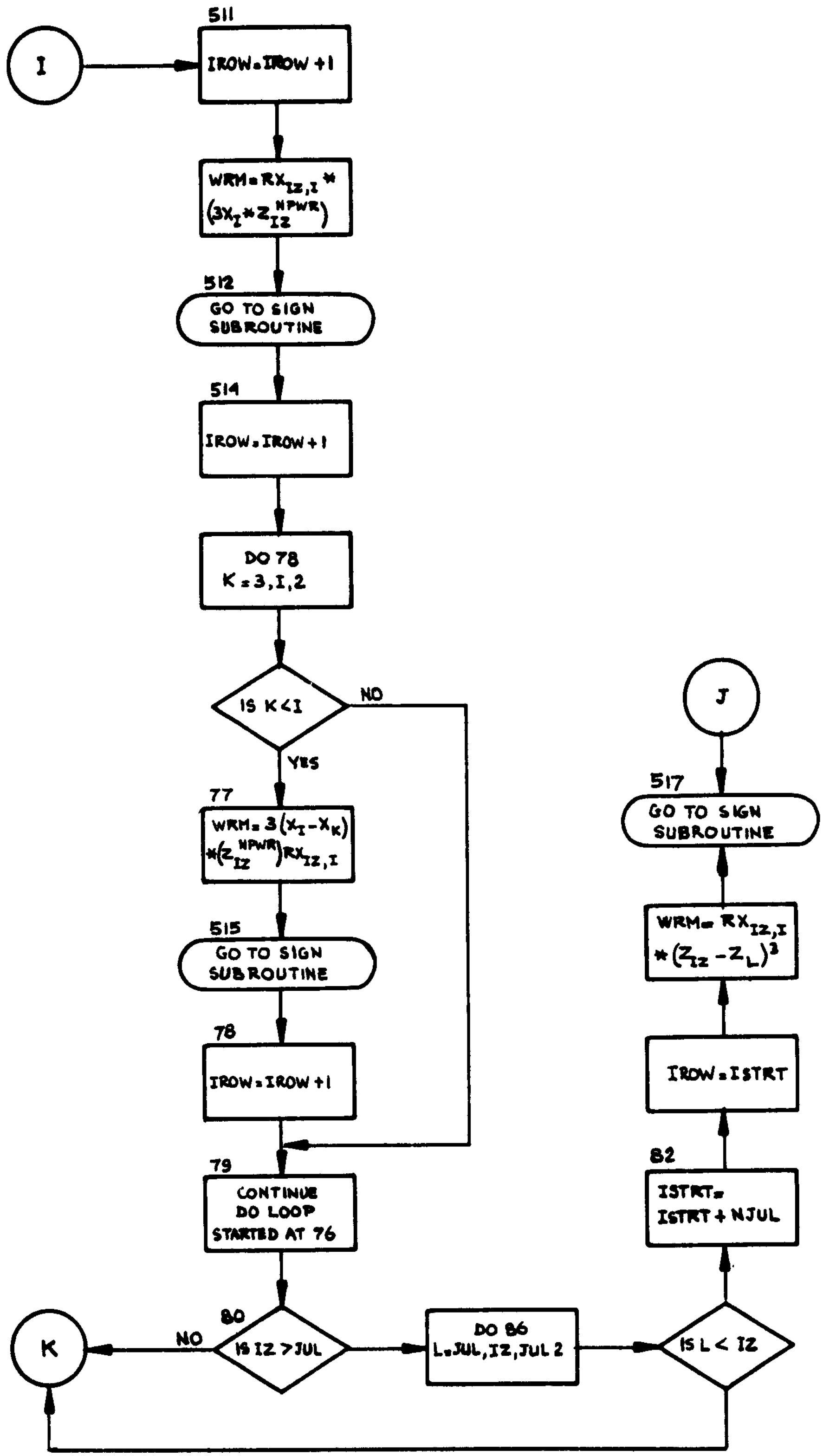

G1
IRW(IF) = IRW(IF-1) + 1
DO 256 NPR = 1,3
TRM(IF) = X(IX)** NPR * ZTRM
PUNCH J, IRW IF, TRM IF
IF = IF + 1
256
IRW(IF) = IRW(IF-1) + 1
G2
DO 59 IT = 3, IX
IS IX > IT
NO
YES
57
IS IT - 2(IT/2) = 0
YES
NO
58
TRM(IF) = (X(IX) - X(IT))**3 * ZTRM
PUNCH J, IRW IF, TRM IF
IF = IF + 1
IRW(IF) = IRW(IF-1) + 1
59
CONTINUE DO LOOP STARTED AT G2
EXIT
G3

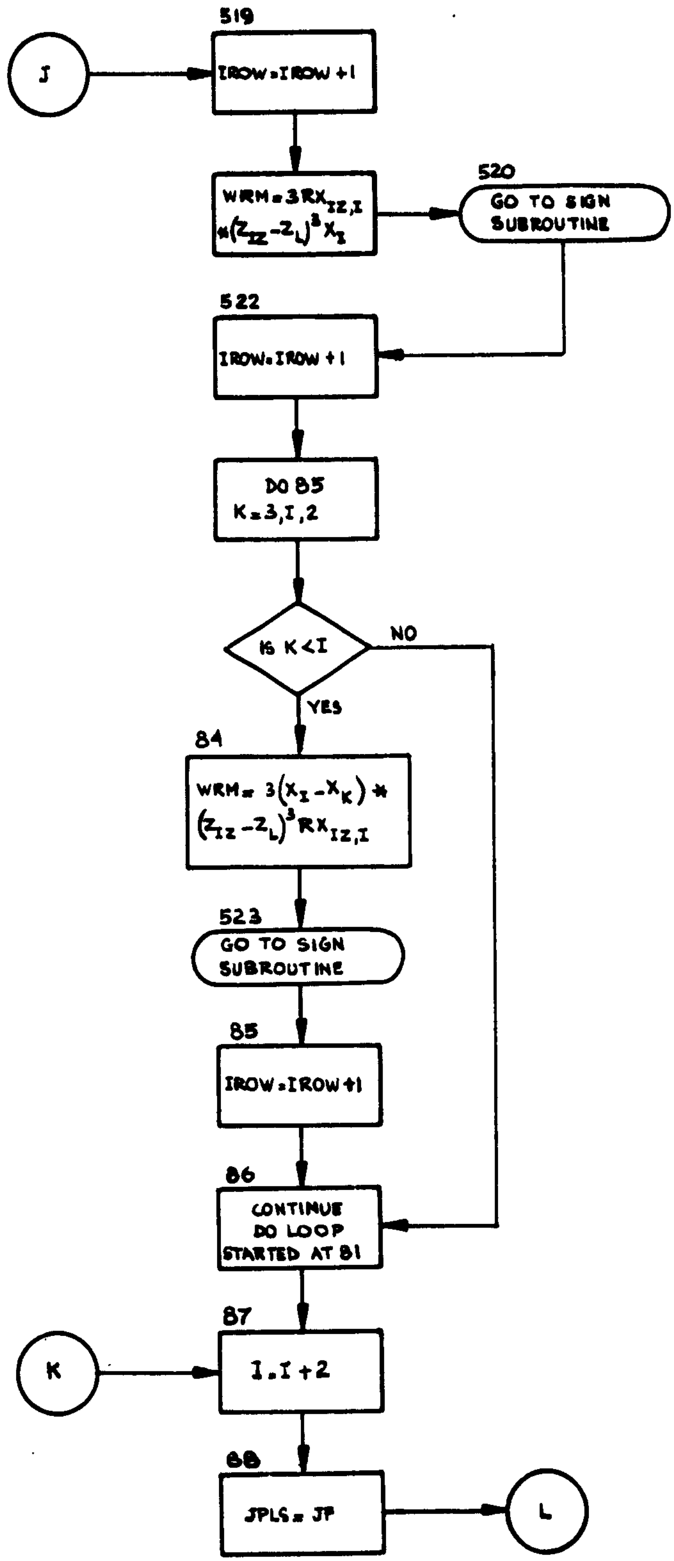

G3
60
$IRW_{IF} = ISTRT$
G4
61
JPLS = J+1
712
KFJ = 2
GO TO PINCH SUBROUTINE
714
IRO = 1
K = IWL(IX)
DO 262
IJ = 2, K
IRO = IRO + 1
262
PUNCH
JPLS, IRO,
$WL_{IX,IJ}$
JF = IF - 1
DO 62
IS = 2, JF
PUNCH
$JPLS, IRW_{IS},$
TRM_{IS}
63
IX = IX + 1
G5
CONTINUE
DO LOOP
STARTED AT F2
EXIT
G6

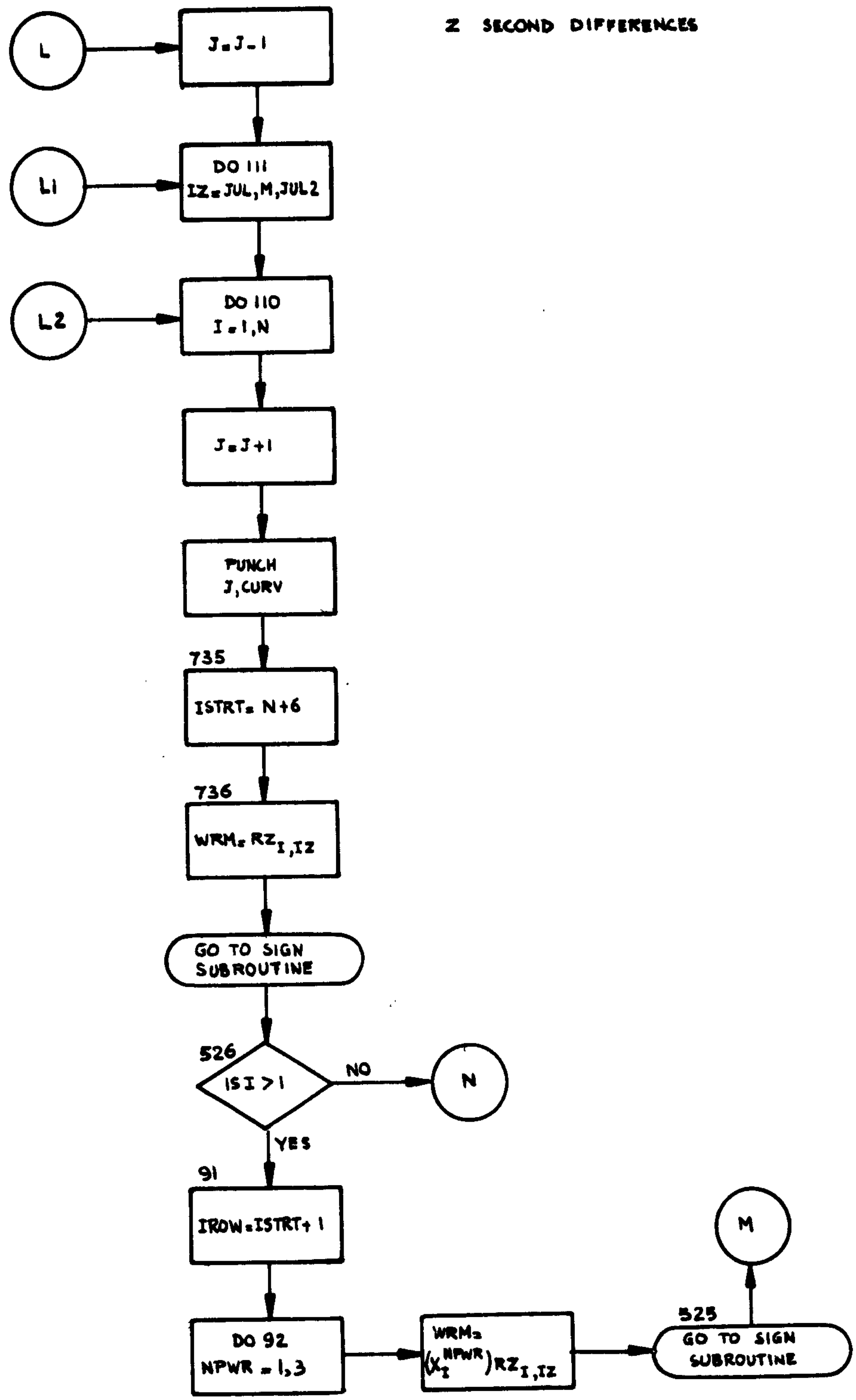
1.0.0-2

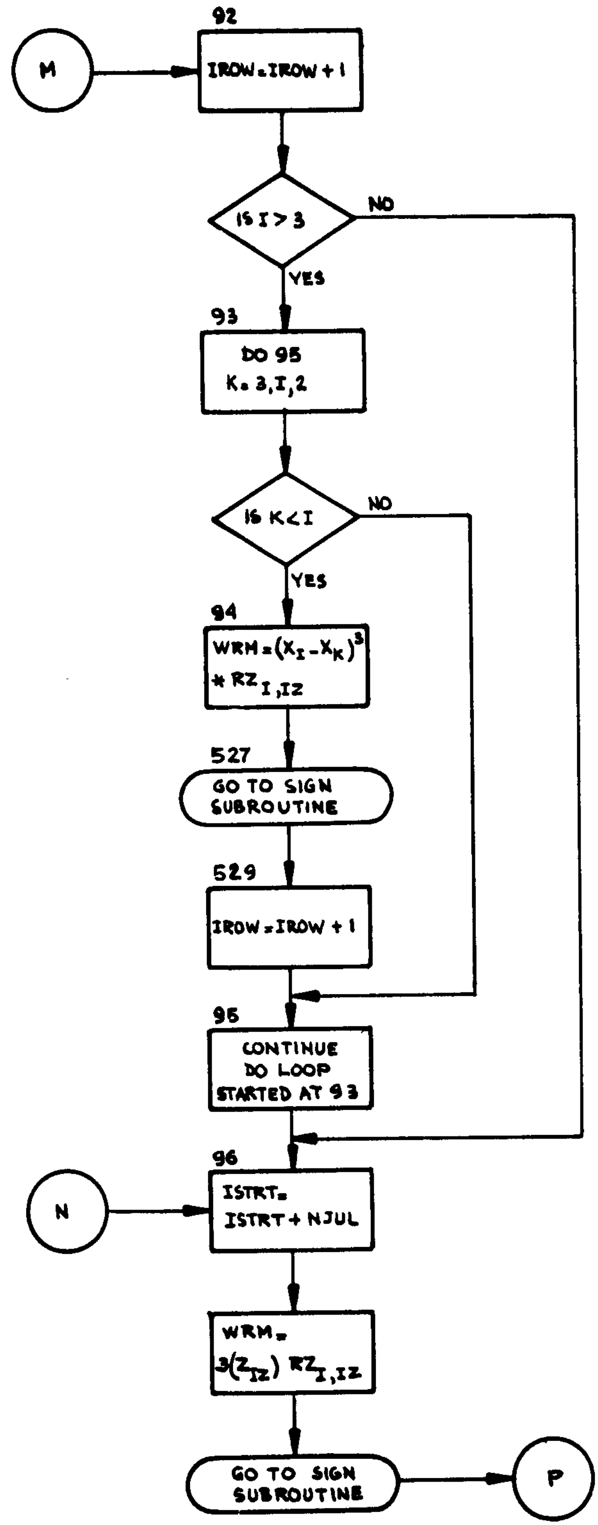

G6
JX = NS/2
X SECOND DIFFERENCES
DO 88
IZ = 1, M
I = 3
JS = JPLS + 1
JF = JS + JX - 1
DO 87
J = JS, JF
PUNCH
J, CURV
IROW = 3
ISTRT = 3
WRM = RX (IZ, I)
501
GO TO SIGN
SUBROUTINE
503
IROW = IROW + 1
WRM = 3 * Xi * RX IZ, i
504
GO TO SIGN
SUBROUTINE
H

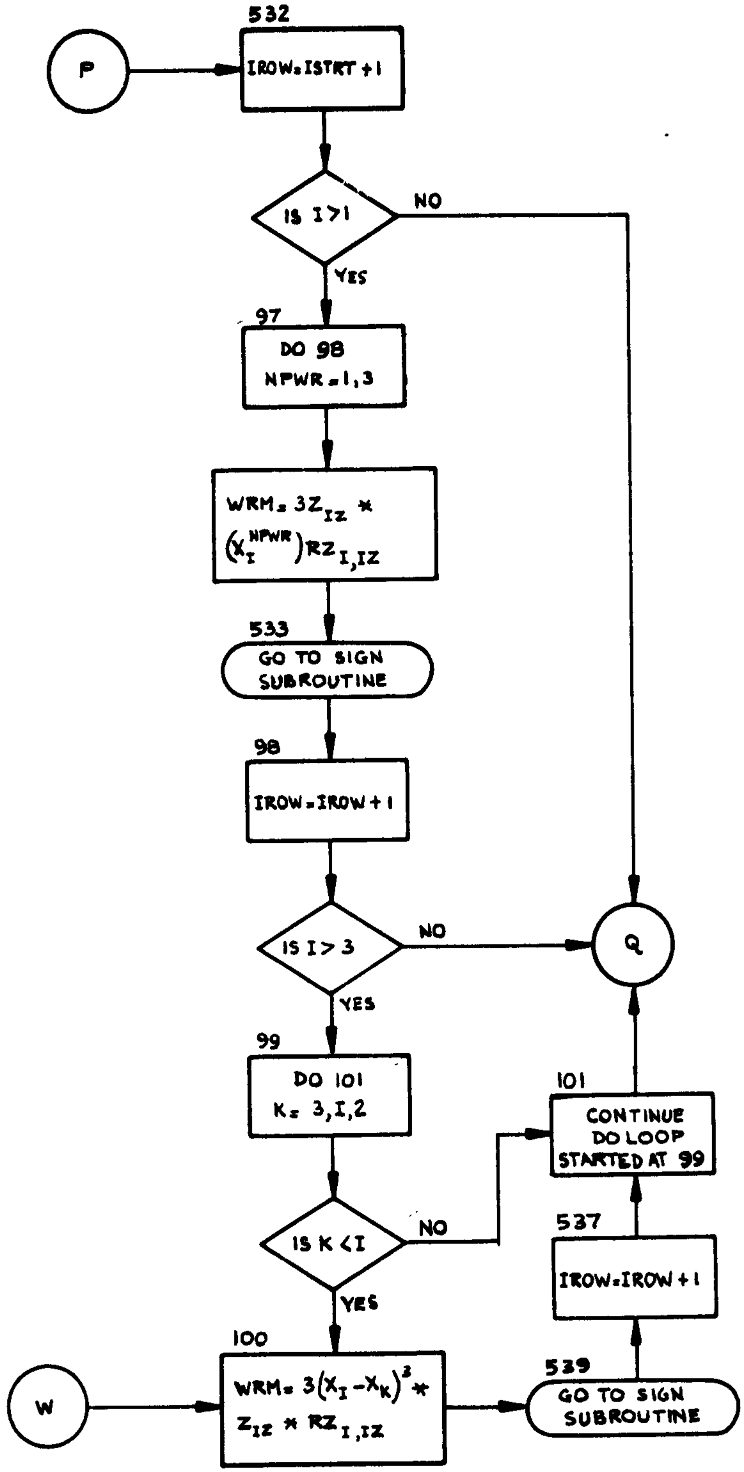

H
506
IS i > 3
NO
YES
72
IROW=IROW+1
DO 74
K=3,I,2
IS K<I
NO
YES
73
WRM=
3(X_I-X_K)RX_{IZ,I}
507
GO TO SIGN SUBROUTINE
74
IROW=IROW+1
75
IS IZ>I
NO
K
YES
76
DO 79
NPWR=1,3
ISTRT=ISTRT+NJUL
IROW=ISTRT
WRM=(Z_IZ^NPWR)RX_{IZ,I}
509
GO TO SIGN SUBROUTINE
I

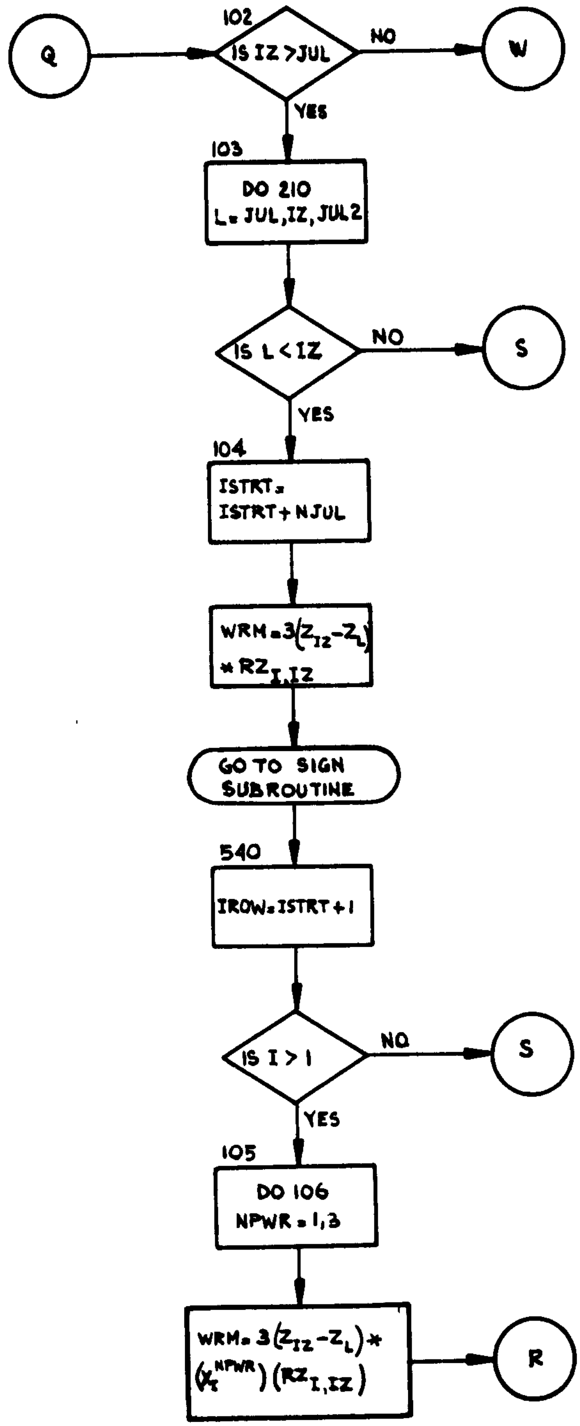

I
511
IROW=IROW+1
WRM=RX_{IZ,I}*(3X_I*Z_{IZ}^{NPWR})
512
GO TO SIGN SUBROUTINE
514
IROW=IROW+1
DO 78 K=3,I,2
IS K<I
NO
YES
77
WRM=3(X_I-X_K)*(Z_{IZ}^{NPWR})RX_{IZ,I}
515
GO TO SIGN SUBROUTINE
78
IROW=IROW+1
79
CONTINUE DO LOOP STARTED AT 76
80
IS IZ>JUL
NO
K
DO 86 L=JUL,IZ,JUL 2
IS L < IZ
J
517
GO TO SIGN SUBROUTINE
WRM=RX_{IZ,I}*(Z_{IZ}-Z_L)^3
IROW=ISTRT
82
ISTRT=ISTRT+NJUL

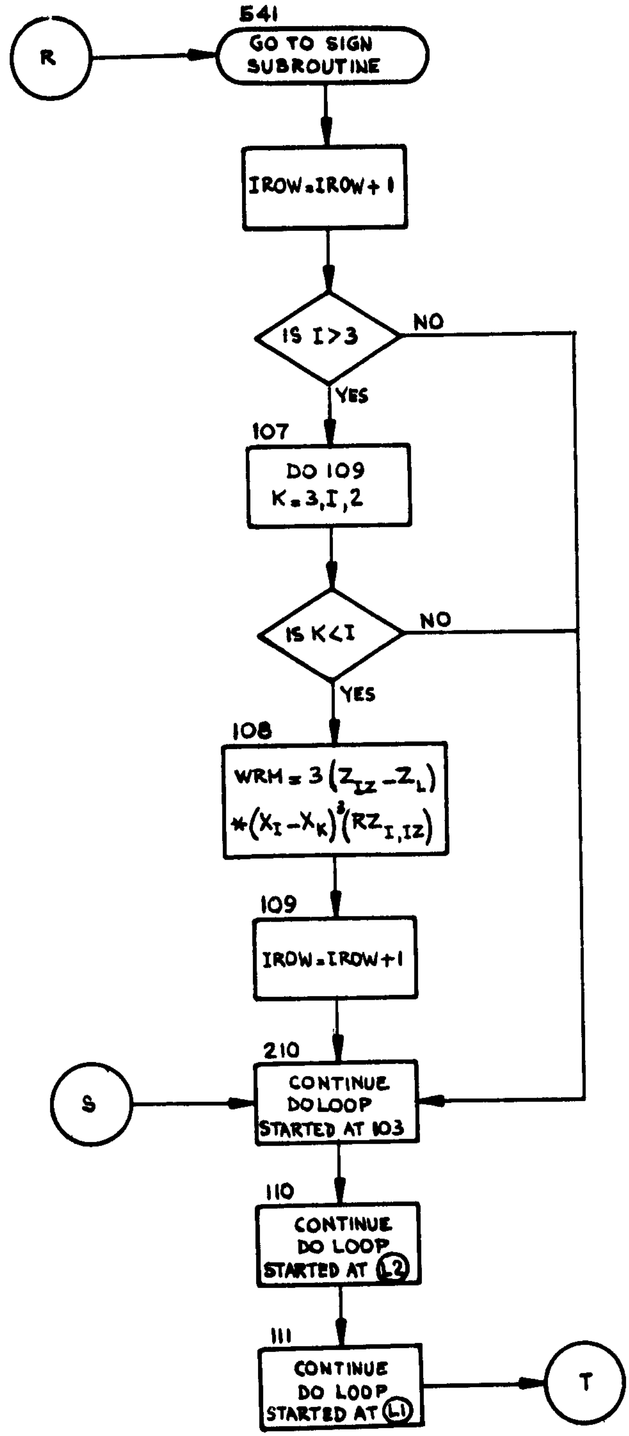

J
519
IROW=IROW+1
WRM=3RX$_{IZ,I}$ *$(Z_{IZ}-Z_L)^3X_I$
520
GO TO SIGN SUBROUTINE
522
IROW=IROW+1
DO 85
K=3,I,2
IS K<I
NO
YES
84
WRM=3(X_I-X_K)* $(Z_{IZ}-Z_L)^3$RX$_{IZ,I}$
523
GO TO SIGN SUBROUTINE
85
IROW=IROW+1
86
CONTINUE DO LOOP STARTED AT 81
87
K
I=I+2
88
JPLS=JP
L

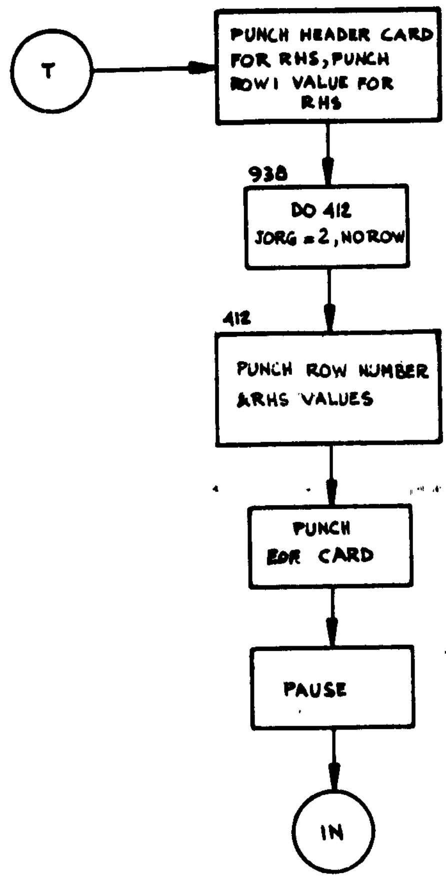

Z SECOND DIFFERENCES
L
J=J-1
L1
DO 111
IZ=JUL,M,JUL2
L2
DO 110
I=1,N
J=J+1
PUNCH
J,CURV
735
ISTRT=N+6
736
WRM=RZ I,IZ
GO TO SIGN SUBROUTINE
526
IS I > 1
NO
N
YES
91
IROW=ISTRT+1
DO 92
NPWR=1,3
WRM=(X I NPWR)RZ I,IZ
525
GO TO SIGN SUBROUTINE
M

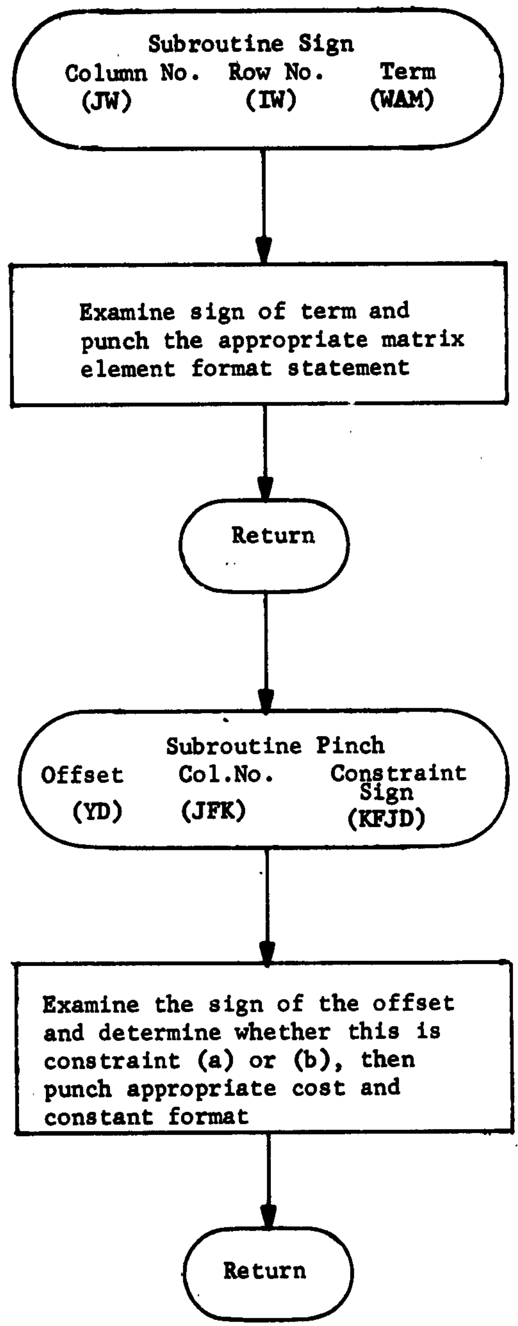

92
M
IROW=IROW+1
IS I>3
NO
YES
93
DO 95
K=3,I,2
IS K<I
NO
YES
94
WRM=(X_I-X_K)^3
*RZ_I,IZ
527
GO TO SIGN
SUBROUTINE
529
IROW=IROW+1
95
CONTINUE
DO LOOP
STARTED AT 93
96
N
ISTRT=
ISTRT+NJUL
WRM=
3(Z_IZ) RZ_I,IZ
GO TO SIGN
SUBROUTINE
P


P
532
IROW=ISTRT+1
IS I>1
NO
YES
97
DO 98
NPWR=1,3
WRM=3Z_IZ *
(X_I^NPWR)RZ_I,IZ
533
GO TO SIGN
SUBROUTINE
98
IROW=IROW+1
IS I>3
NO
YES
Q
99
DO 101
K=3,I,2
101
CONTINUE
DO LOOP
STARTED AT 99
IS K<I
NO
YES
537
IROW=IROW+1
100
W
WRM=3(X_I-X_K)^3*
Z_IZ * RZ_I,IZ
539
GO TO SIGN
SUBROUTINE


Q
102
IS IZ > JUL
NO
W
YES
103
DO 210
L = JUL, IZ, JUL2
IS L < IZ
NO
S
YES
104
ISTRT = ISTRT + NJUL
WRM = 3(Z_IZ - Z_L) * RZ_I,IZ
GO TO SIGN SUBROUTINE
540
IROW = ISTRT + 1
IS I > 1
NO
S
YES
105
DO 106
NPWR = 1,3
WRM = 3(Z_IZ - Z_L) * (X_I^NPWR)(RZ_I,IZ)
R


R
541
GO TO SIGN SUBROUTINE
IROW=IROW+1
IS I>3
NO
YES
107
DO 109
K=3,I,2
IS K<I
NO
YES
108
WRM = 3(Z_IZ - Z_L)
*(X_I - X_K)^3(RZ_I,IZ)
109
IROW=IROW+1
210
CONTINUE DOLOOP STARTED AT 103
S
110
CONTINUE DO LOOP STARTED AT L2
111
CONTINUE DO LOOP STARTED AT L1
T


T
PUNCH HEADER CARD FOR RHS, PUNCH ROW1 VALUE FOR RHS
938
DO 412 JORG = 2, NOROW
412
PUNCH ROW NUMBER & RHS VALUES
PUNCH EOR CARD
PAUSE
IN


Subroutine Sign
Column No. (JW)
Row No. (IW)
Term (WAM)
Examine sign of term and punch the appropriate matrix element format statement
Return
Subroutine Pinch
Offset (YD)
Col.No. (JFK)
Constraint Sign (KFJD)
Examine the sign of the offset and determine whether this is constraint (a) or (b), then punch appropriate cost and constant format
Return

```
C     SMOG1   DUAL  X OR X AND Z DBL SPL  NO PROFILE
      DIMENSION RX(11,15),RZ(15,11),TRM(150), IRW(250)
      DIMENSION X(15),Z(11),Y(11,15),WL(15,11),IWL(15)
C               *   *      INPUT     *    *
  750 READ 173
      READ 172,NAA,NAB
      READ 150,M,N,CURV,RHS
      NL=N+1
      ML=M+1
      NS=N-1
      MS=M-1
      DO 2 I=1,M
      READ 154,Z(I)
      DO 2 J=1,N
      IF(SENSE SWITCH 3)802,803
  802 READ 183,X(J),IAY,IBY,ICY
      AAY=IAY
      BBY=IBY
      CCY=ICY
      Y(I,J)=AAY+(BBY+(CCY/8.))/12.
      GO TO 2
  803 READ 154,X(J),Y(I,J)
    2 CONTINUE
      IF(NAA)3,3,4
    3 READ 154,SX,SZ,TX,TZ
      GO TO 5
    4 SX=X(2)-X(1)
      SZ=Z(2)-Z(1)
      TZ=Z(1)
      TX=X(1)
C       *   *     WATERLINE  2ND  DIFFERENCES     *    *
    5 IF(SENSE SWITCH 2)6,7
    6 PRINT 173
      PRINT 157
    7 DO 15 I=1,M
      DO 13 J=2,NS
      A=2./((X(J+1)-X(J-1)))
      ZA=Y(I,J+1)-Y(I,J)
      A=A*(ZA/(X(J+1)-X(J))-(Y(I,J)-Y(I,J-1))/(X(J)-X(J-1)))
      IF(A)10,9,8
    8 RX(I,J)=-1.
      GO TO 11
    9 RX(I,J)=0.
      GO TO 11
   10 RX(I,J)=1.
```

```
   11 IF(SENSE SWITCH 2)12,13
   12 PRINT 158,I,J,A,RX(I,J)
   13 CONTINUE
      RX(I,N)=RX(I,NS)
      IF(SENSE SWITCH 2)14,15
   14 PRINT 158,I,J,A,RX(I,N)
   15 CONTINUE
C        *   *   STATION  2ND  DIFFERENCES     *    *
      IF(SENSE SWITCH 2)16,17
   16 PRINT 159
   17 DO 25 J=1,N
      DO 23 I=2,MS
      A=2./(Z(I+1)-Z(I-1))
      ZAA=Y(I+1,J)-Y(I,J)
      A=A*(ZAA/(Z(I+1)-Z(I))-(Y(I,J)-Y(I-1,J))/(Z(I)-Z(I-1)))
      IF(A)20,19,18
   18 RZ(J,I)=-1.
      GO TO 21
   19 RZ(J,I)=0.
      GO TO 21
   20 RZ(J,I)=1.
   21 IF(SENSE SWITCH 2)22,23
   22 PRINT 158,J,I,A,RZ(J,I)
   23 CONTINUE
      RZ(J,M)=RZ(J,MS)
      IF(SENSE SWITCH 2)24,25
   24 PRINT 158,J,I,A,RZ(J,M)
   25 CONTINUE
C           *   *    NORMALIZING         *    *
      DO 26 J=1,N
   26 X(J)=(X(J)-TX)/SX
      DO 27 I=1,M
   27 Z(I)=(Z(I)-TZ)/SZ
C               *    *         ROW ID             *    *
      PUNCH 173
      NJUL=((N-3)/2)+4
      NM=N
      IF(NAB)719,718,718
  719 JUL=3
      JUL2=2
      NOROW=NJUL*((M-3)/2+4)
      GO TO 720
  718 JUL=2
      JUL2=1
      NOROW=(M+2)*NJUL
```

```
  720 PUNCH 160
      DO 28 NR=1,NOROW
   28 PUNCH 162,NR
C               *    *        MATRIX            *    *
      PUNCH 164
C                    *    *          FIRST  WATERLINE          *

      MLG=(2*NM)-2
      I=2
      DO 37 J=1,MLG,2
      Y(1,I)=Y(1,I)-Y(1,1)
  700 KFJ=1
      CALL PINCH (Y(1,I),J,KFJ)
  702 IROW=2
      DO 31 NPR=1,3
      TRM(IROW)=X(I)**NPR
      PUNCH 167,J,IROW,TRM(IROW)
   31 IROW=IROW+1
      IF(I-3)35,35,32
   32 DO 34 IT=3,I
      IF(I-IT)34,34,133
  133 IF(IT-2*(IT/2))33,34,33
   33 TRM(IROW)=(X(I)-X(IT))**3
      PUNCH 167,J,IROW,TRM(IROW)
      IROW=IROW+1
   34 CONTINUE
   35 JPLS=J+1
      IROW=IROW-1
      IWL(I)=IROW
  703 KFJ=2
      CALL PINCH (Y(1,I),JPLS,KFJ)
  705 DO 36 IR=2,IROW
      WL(I,IR)=TRM(IR)
   36 PUNCH 169,JPLS,IR,TRM(IR)
   37 I=I+1
C          *    *    REMAINING WL    *    *
      DO 64 IZ=2,M
      JL=J
      MG=JL+(2*NM)-2
      IX=1
      DO 63 J=JL,MG,2
      IF=2
      IRW(IF)=IROW+1
      Y(IZ,IX)=Y(IZ,IX)-Y(1,1)
  706 KFJ=1
      CALL PINCH (Y(IZ,IX),J,KFJ)
```

```
708 IF(J-JL)41,41,44
 41 DO 42 IZPR=1,3
    TRM(IF)=Z(IZ)**IZPR
    PUNCH 167,J,IRW(IF),TRM(IF)
    IF=IF+1
 42 IRW(IF)=IRW(IF-1)+NJUL
    IF(IZ-JUL)146,146,142
142 DO 145 ITZ=JUL,IZ
    IF(IZ-ITZ)146,146,729
729 GO TO (144,144,730),JUL
730 IF(ITZ-2*(ITZ/2))144,145,144
144 ZTR=(Z(IZ)-Z(ITZ))**3
    TRM(IF)=ZTR
    PUNCH 167,J,IRW(IF),TRM(IF)
    IF=IF+1
    IRW(IF)=IRW(IF-1)+NJUL
145 CONTINUE
146 JPLS=J+1
709 KFJ=2
    CALL PINCH (Y(IZ,IX),JPLS,KFJ)
711 JF=IF-1
    DO 43 IF=2,JF
 43 PUNCH 169,JPLS,IRW(IF),TRM(IF)
    GO TO 63
C      * * Z  , Z SQR , Z CUBE ROWS  * *
 44 IRO =1
    K=IWL(IX)
    DO 244 IJ=2,K
    IRO =IRO +1
244 PUNCH 167,J,IRO,WL(IX,IJ)
    DO 50 IZPR=1,3
    ISTRT=IRW(IF)+NJUL
    ZTRM=Z(IZ)**IZPR
    TRM(IF)=ZTRM
    PUNCH 167,J,IRW(IF),TRM(IF)
    IF=IF+1
    IRW(IF)=IRW(IF-1)+1
    DO 45 NPR=1,3
    TRM(IF)=(X(IX)**NPR)*ZTRM
    PUNCH 167,J,IRW(IF),TRM(IF)
    IF=IF+1
 45 IRW(IF)=IRW(IF-1)+1
    IF(IX-3)50,50,46
 46 DO 49 IT=3,IX
    IF(IX-IT)49,49,47
```

```
   47 IF(IT-2*(IT/2))48,49,48
   48 TRM(IF)=((X(IX)-X(IT))**3)*ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+1
   49 CONTINUE
   50 IRW(IF)=ISTRT
C       *   *     THEILHEIMER  ROWS       *   *
   53 IF(IZ-JUL)61,61,54
   54 DO 60 ITZ=JUL,IZ
      GO TO (55,55,731),JUL
  731 IF(ITZ-2*(ITZ/2))55,60,55
   55 ISTRT=IRW(IF)+NJUL
      IF(IZ-ITZ)61,61,56
   56 ZTRM=(Z(IZ)-Z(ITZ))**3
      TRM(IF)=ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+1
      DO 256 NPR=1,3
      TRM(IF)=(X(IX)**NPR)*ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
  256 IRW(IF)=IRW(IF-1)+1
      DO 59 IT=3,IX
      IF(IX-IT)59,59,57
   57 IF(IT-2*(IT/2))58,59,58
   58 TRM(IF)=((X(IX)-X(IT))**3)*ZTRM
      PUNCH 167 ,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+1
   59 CONTINUE
   60 IRW(IF)=ISTRT
   61 JPLS=J+1
  712 KFJ=2
      CALL PINCH (Y(IZ,IX),JPLS,KFJ)
  714 IRO =1
      K=IWL(IX)
      DO 262 IJ=2,K
      IRO =IRO +1
  262 PUNCH 169,JPLS,IRO ,WL(IX,IJ)
      JF=IF-1
      DO 62 IS=2,JF
   62 PUNCH 169,JPLS,IRW(IS),TRM(IS)
   63 IX=IX+1
```

```
   64 CONTINUE
C        *    *   X SEC  DIFFERENCES   *    *
      JX=NS/2
      DO 88 IZ=1,M
      I=3
      JS=JPLS+1
      JF=JS+JX-1
      DO 87 J=JS,JF
      PUNCH 171,J,CURV
      IROW=3
      ISTRT=3
      WRM=RX(IZ,I)
  501 CALL SIGN(J,IROW,WRM)
  503 IROW=IROW+1
      WRM=3.*X(I)*RX(IZ,I)
  504 CALL SIGN(J,IROW,WRM)
  506 IF(I-3)75,75,72
   72 IROW=IROW+1
      DO 74 K=3,I,2
      IF(K-I)73,75,75
   73 WRM=3.*(X(I)-X(K))*RX(IZ,I)
  507 CALL SIGN(J,IROW,WRM)
   74 IROW=IROW+1
   75 IF(IZ-1)87,87,76
   76 DO 79 NPWR=1,3
      ISTRT=ISTRT+NJUL
      IROW=ISTRT
      WRM=(Z(IZ)**NPWR)*RX(IZ,I)
  509 CALL SIGN(J,IROW,WRM)
  511 IROW=IROW+1
      WRM=(3.*X(I)*(Z(IZ)**NPWR))*RX(IZ,I)
  512 CALL SIGN(J,IROW,WRM)
  514 IROW=IROW+1
      DO 78 K=3,I,2
      IF(K-I)77,79,79
   77 WRM=3.*(X(I)-X(K))*(Z(IZ)**NPWR)*RX(IZ,I)
  515 CALL SIGN(J,IROW,WRM)
   78 IROW=IROW+1
   79 CONTINUE
   80 IF(IZ-JUL)87,87,81
   81 DO 86 L=JUL,IZ,JUL2
      IF(L-IZ)82,87,87
   82 ISTRT=ISTRT+NJUL
      IROW=ISTRT
      WRM=RX(IZ,I)*((Z(IZ)-Z(L))**3)
```

```
  517 CALL SIGN(J,IROW,WRM)
  519 IROW=IROW+1
      WRM=3.*X(I)*((Z(IZ)-Z(L))**3)*RX(IZ,I)
  520 CALL SIGN(J,IROW,WRM)
  522 IROW=IROW+1
      DO 85 K=3,I,2
      IF(K-I)84,86,86
   84 WRM=(3.*(X(I)-X(K))*((Z(IZ)-Z(L))**3))*RX(IZ,I)
  523 CALL SIGN(J,IROW,WRM)
   85 IROW=IROW+1
   86 CONTINUE
   87 I=I+2
   88 JPLS=JF
C         Z  SEC  DIFFERENCES    *    *
      J=J-1
      DO 111 IZ=JUL,M,JUL2
      DO 110 I=1,N
      J=J+1
      PUNCH 171,J,CURV
  735 ISTRT=N+6
  736 WRM=RZ(I,IZ)
      CALL SIGN(J,ISTRT,WRM)
  562 IF(I-1)96,96,91
   91 IROW=ISTRT+1
      DO 92 NPWR=1,3
      WRM=(X(I)**NPWR)*RZ(I,IZ)
  525 CALL SIGN(J,IROW,WRM)
   92 IROW=IROW+1
      IF(I-3)96,96,93
   93 DO 95 K=3,I,2
      IF(K-I)94,95,95
   94 WRM=((X(I)-X(K))**3)*RZ(I,IZ)
  527 CALL SIGN(J,IROW,WRM)
  529 IROW=IROW+1
   95 CONTINUE
   96 ISTRT=ISTRT+NJUL
      WRM=3.*Z(IZ)*RZ(I,IZ)
      CALL SIGN(J,ISTRT,WRM)
  532 IROW=ISTRT+1
      IF(I-1)102,102,97
   97 DO 98 NPWR=1,3
      WRM=(X(I)**NPWR)*Z(IZ)*3.*RZ(I,IZ)
  533 CALL SIGN(J,IROW,WRM)
   98 IROW=IROW+1
      IF(I-3)102,102,99
```

```
 99 DO 101 K=3,I,2
    IF(K-I)100,101,101
100 WRM=((X(I)-X(K))**3)*Z(IZ)*3.*RZ(I,IZ)
535 CALL SIGN(J,IROW,WRM)
537 IROW=IROW+1
101 CONTINUE
102 IF(IZ-JUL)110,110,103
103 DO 210 L=JUL,IZ,JUL2
    IF(L-IZ)104,210,210
104 ISTRT=ISTRT+NJUL
    WRM=(Z(IZ)-Z(L))*RZ(I,IZ)*3.
    CALL SIGN(J,ISTRT,WRM)
540 IROW=ISTRT+1
    IF(I-1)210,210,105
105 DO 106 NPWR=1,3
    WRM=(Z(IZ)-Z(L))*(X(I)**NPWR)*3.*RZ(I,IZ)
541 CALL SIGN(J,IROW,WRM)
106 IROW=IROW+1
    IF(I-3)210,210,107
107 DO 109 K=3,I,2
    IF(K-I)108,210,210
108 WRM=(Z(IZ)-Z(L))*((X(I)-X(K))**3)*3.*RZ(I,IZ)
543 CALL SIGN(J,IROW,WRM)
109 IROW=IROW+1
210 CONTINUE
110 CONTINUE
111 CONTINUE
    PUNCH 174
    PUNCH 175
738 DO 412 JORG=2,NOROW
412 PUNCH 177,JORG,RHS
737 PUNCH 176
    PRINT 178
    PAUSE 751
    GO TO 750
150 FORMAT(2I5,F11.6,F11.6)
154 FORMAT(4F10.4)
157 FORMAT(10X,17H2ND DIFF - WLINES//
   C30H   WL    STA  ACT VALUE    SIGN/)
158 FORMAT(2I5,F12.6,F8.2)
159 FORMAT(//10X,17H2ND DIFF-STATIONS//
   C30H   STA   WL   ACT VALUE    SIGN/)
160 FORMAT(6HROW ID/12X,6H OFSET)
162 FORMAT(12X,2H R,I4)
```

```
164 FORMAT(6HMATRIX)
167 FORMAT(6X,2H C,I4,2H R,I4,1H-,F11.5)
169 FORMAT(6X,2H C,I4,2H R,I4,F12.5)
171 FORMAT(6X,2H C,I4,6H OFSET,F12.5)
172 FORMAT(2I5)
173 FORMAT(1H*,5X,47H                                              )
174 FORMAT(7HFIRST B)
175 FORMAT(12X,14H R   1      2.0)
176 FORMAT(3HEOF)
177 FORMAT(12X,2H R,I4,1H ,F11.5)
178 FORMAT(28HENTER NEW DATA - PRESS START)
183 FORMAT(F10.4,3I5)
    END
```

```
      SUBROUTINE PINCH (YD,JFK,KFJD)
      IF(YD)706,707,707
  706 YD=-YD
      GO TO (752,753),KFJD
  752 PUNCH 168,JFK,YD,JFK
      GO TO 754
  753 PUNCH 166,JFK,YD,JFK
  754 YD=-YD
      RETURN
  707 GO TO (755,756),KFJD
  755 PUNCH 166,JFK,YD,JFK
      RETURN
  756 PUNCH 168,JFK,YD,JFK
      RETURN
  166 FORMAT(6X,2H C,I4,6H OFSET,F12.5/6X,2H C,I4,6H R   1,8H      1
 .0)
  168 FORMAT(6X,2H C,I4,7H OFSET-,F11.5/6X,2H C,I4,6H R   1,8H
 1.0)
  183 FORMAT(F10.5,3I5)
      END
```

```
      SUBROUTINE SIGN (JW,IW,WAM)
      IF(WAM)370,373,371
  370 WAM=-WAM
      PUNCH 169,JW,IW,WAM
      RETURN
  371 PUNCH 167,JW,IW,WAM
  373 RETURN
  169 FORMAT(6X,2H C,I4,2H R,I4,F12.5)
  167 FORMAT(6X,2H C,I4,2H R,I4,1H-,F11.5)
      END
```

B. SMOG 2

This program produces a surface matrix without profile conditions in the dual formulation and λ fit. The surface will be double splined in the x direction and may be either single or double splined in the z direction. The program is written in FORTRAN II for an IBM-1620 computer.

Organization

SMOG 2 consists of two passes as shown in Fig. B-1. The input data shown below is read into Pass 1 . At the termination of the execution of Pass 1 the data necessary for Pass 2 is punched. This data can be read directly into Pass 2 . If it becomes necessary to change the sign of a second difference as mentioned in Step 2, this may be done by altering the Pass 2 data. More information on this data is found in the description of the Pass 2 input.

B. SMOG 2

PASS 1

OPERATING INSTRUCTIONS

Pass 1 of SMOG 2 calculates the second differences of the offsets and the intersections of the profile with the waterlines and stations. It then punches the row identifications, deviation constraints and data for Pass II. The program is written in FORTRAN II for the IBM-1620 computer.

Fortran Symbol Definition

M	-	Total number of waterlines in the surface to be faired. This will be an odd number if the surface is double splined in the z direction.
N	-	Total number of stations in the surface. This will always be an odd number.
X(J)	-	Distance along the x axis from the origin of the surface to Station j .
Z(I)	-	Distance along the z axis from the origin of the surface to waterline i .
Y(I,J)	-	Offset of the surface at Waterline i , Station j , in feet and decimals
IAY	-	Number of feet in the offset if read in feet, inches and eighths.
IBY	-	Number of inches corresponding to IAY
ICY	-	Number of eighths corresponding to IAY
NAA	-	Program indicator

If NAA = -1, scale factors for normalizing and translating X(J) and Z(I) are read in.

If NAA = +1, scaling and translating are done by the program as given in Section I of this Appendix.

SX - Scale factor for X(J) when these are read in

TX - Translation factor for X(J) when these are read in

SZ - Scale factor for Z(I) when these are read in

TZ - Translation factor for Z(I) when these are read in

RHS - Right hand side - In the dual, this is really the cost on the equation coefficients. The ideal value for RHS would be zero; however, this makes the solution susceptible to cycling so that a value of .0001 is usually used.

CURV - Value of the requirement vector for the curvature constraints, this value actually goes in the cost row in the dual and is always made zero.

NAB - Program indicator

If NAB = -1 , the surface will be double splined in the z direction and M must be an odd number.

If NAB = +1 , the surface will be single splined in the z direction and M is unrestricted.

COST - The penalty assigned to λ . This will appear in Row 1 of the right-hand side. The value is normally made 20 .

EXP 1 - The value of p in the profile function, usually made .3333 .

EXP 2 - The value of r in the profile function, usually 3.0 .

POINT - The value of D in the profile function. This number is the length of the fraction of the station spacing over which the profile function is effective in terms of the full-scale surface (the value will be translated and normalized exactly like the x values are).

C(I) - Coefficients of the Theilheimer equation describing the profile. There must be (M+2) coefficients. Straight lines or other curves may be used by making all except the desired coefficients zero.

Input

The input data cards are described below. There are certain input variables on which there are restrictions on the value which can be entered. These restrictions follow.

Variable	Minimum No.	Maximum No.
N	3	15
M	3	11

In describing the various data cards, the actual FORTRAN format field is used in most cases. These fields come consecutively across the card with no gaps or blank columns between, except where indicated. The field descriptions are the FORTRAN F field, which uses the FORTRAN fixed point decimal number, and the I field that uses the FORTRAN integer number, which is always right justified. The card numbers are not punched on the data cards.

Contents of Card							Card No.
Card Columns	1	2-12	13-53				1
Contents	*	Blank	Any alphameric problem heading (This heading will be used for identifying the problem on the LP output)				
Format Variable	I5 NAA	I5 NAB	F11.6 POINT	F11.6 COST			2
Format Variable	I5 M	I5 N	F11.6 CURV	F11.6 RHS	F11.6 EXP1	F11.6 EXP2	3
Format Variable	F10.4 C(I)						Next M+2 cards

Format	F10.4
Variable	Z(I)

Alternate Cards:

Format	F10.4	F10.4
Variable	X(J)	Y(I,J)

or:

Format	F10.4	15	15	15
Variable	X(J)	IAY	IBY	ICY

N Cards per Group

Next M Groups

This arrangement can be stated as follows:

The first card of each set gives the height z of the waterline. The following cards each give a station coordinate and the offset at the intersection of the given waterline and station. There is one set of cards for each waterline in the surface.

Format	F10.4	F10.4	F10.4	F10.4	Last card
Variable	SX	SZ	TX	TZ	

(This card is entered only if NAA = -1

<u>Output</u>

The output from this program is designed to be read directly by the LP-90 program after card to magnetic tape conversion.

The first card punched contains the Loading Read in the first input card. The next set of cards are the row identification coordinates. These identify each row of the matrix to be used in this solution. For this matrix all of the rows are always used and identified, including the cost row which has the name "OFSET."

Next, a card punched "MATRIX" is produced, followed by the deviation constraint portion of the matrix. Finally the input data for Pass 2 is punched.

The output formats are shown in Fig. B-1 , and the order of the output deck in Fig B-2 . The actual output from PASS 1 is shown in the sample problem.

<u>Sense Switch Settings</u>

Sense Switch 1	-	Not used	
Sense Switch 2	-	ON	The values of the x and z second differences and the resulting signs are printed.
Sense Switch 2	-	OFF	The above is not printed
Sense Switch 3	-	ON	The offsets are read in as feet, inches, and eighths
Sense Switch 3	-	OFF	The offsets are read in as feet and decimals

SAMPLE PROBLEM

The following sample problem is a five-station by five-waterline set of offsets taken from the DLG-26 class ships. The stations are Numbers 0 , 1/2 , 1 , 1-1/2 , and 2 , and the waterlines are 12, 16, 24, and 28 .

The surface contains the bow profile of the DLG. The matrix is double splined in the z direction. Only the output from Pass 1 is listed here; the output from Pass 2 is included with the instructions from that program.

SAMPLE PROBLEM - SMOG 2, PASS 1

** SAMPLE INPUT **

```
*     TODD  DLG  5X5 END PROFI LE X-Z DBL SPL-
  1   -1   1.       20.
  5    5   0.          .001      3.          .33333333
 1.52935
- .5873
 0.
 0.
 0.
 0.
 0.
 0.
    .0000
    .0000    .000000
  12.7500   1.041666
  25.5000   2.458333
  38.2500   3.874999
  51.0000   5.333333
   1.0000
    .0000    .000000
  12.7500   1.239583
  25.5000   2.697916
  38.2500   4.197916
  51.0000   5.812500
   2.0000
    .0000    .078
  12.7500   1.479166
  25.5000   2.979166
  38.2500   4.583333
  51.0000   6.333333
   3.0000
    .0000    .343749
  12.7500   1.791666
  25.5000   3.447916
  38.2500   5.250000
  51.0000   7.166666
   4.0000
    .0000    .708333
  12.7500   2.416666
  25.5000   4.322916
  38.2500   6.416666
  51.0000   8.583333
```

End Profile Coefficients

Surface Offsets

TYPEWRITER LISTING OF INTERSECTIONS AND SECOND DIFFERENCES

(Switch 2 ON) OUTPUT

```
*     TODD  DLG  5X5 END PROFILE X-Z DBL SPL-

      END PROFILE INTERSECTIONS

              WLINES

          1    1.529350
     -.5873    0.0000
          2     .942050
     -.5873    0.0000
          3     .354750
     -.5873    0.0000
          4    -.232550
     -.5873    0.0000
          5    -.819850
     -.5873    0.0000

              STATIONS

          1    2.604003
          2     .901367
          3    0.000000
          4    0.000000
          5    0.000000


          2ND DIFF - WLINES

    WL    STA NORM VALUE    SIGN

     1     1   OUT OF SURFACE
     1     2   OUT OF SURFACE
     1     3   -5.176766     1.00
     1     4     .041668    -1.00
     1     5     .041668    -1.00
     2     1   OUT OF SURFACE
     2     2  -37.680848     1.00
     2     3     .041667    -1.00
     2     4     .114584    -1.00
     2     5     .114584    -1.00
     3     1   OUT OF SURFACE
     3     2    -.963248     1.00
     3     3     .104167    -1.00
     3     4     .145833    -1.00
     3     5     .145833    -1.00
     4     1    -.049093     1.00
     4     2     .208333    -1.00
     4     3     .145834    -1.00
     4     4     .114582    -1.00
     4     5     .114582    -1.00
```

5	1	.927938	-1.00
5	2	.197917	-1.00
5	3	.187500	-1.00
5	4	.072917	-1.00
5	5	.072917	-1.00

2ND DIFF-STATIONS

STA	WL	NORM VALUE	SIGN
1	2	OUT OF SURFACE	
1	3	OUT OF SURFACE	
1	4	-.439121	1.00
1	5	-.439121	1.00
2	2	-3.216759	1.00
2	3	.072917	-1.00
2	4	.312500	-1.00
2	5	.312500	-1.00
3	2	.041667	-1.00
3	3	.187500	-1.00
3	4	.406250	-1.00
3	5	.406250	-1.00
4	2	.062500	-1.00
4	3	.281250	-1.00
4	4	.499999	-1.00
4	5	.499999	-1.00
5	2	.041666	-1.00
5	3	.312500	-1.00
5	4	.583334	-1.00
5	5	.583334	-1.00

```
*     TODD  DLG  5X5 END PROFILE X-Z DBL SPL-12/3/62
ROW ID
              OFSET
              R    1
              R    2
              R    3
              R    4
              R    5
              R    6
              R    7
              R    8
              R    9
              R   10
              R   11
              R   12
              R   13
              R   14
              R   15
              R   16
              R   17
              R   18
              R   19
              R   20
              R   21
              R   22
              R   23
              R   24
              R   25
              R   26
MATRIX
          C    5 OFSET      2.45833
          C    5 R    1     1.0
          C    5 R    2-    1.0
          C    5 R    3-    2.00000
          C    5 R    4-    4.00000
          C    5 R    5-    8.00000
          C    6 OFSET-     2.45833
          C    6 R    1     1.0
```

1.0.0-2

```
C    6 R    2      1.0
C    6 R    3      2.00000
C    6 R    4      4.00000
C    6 R    5      8.00000
C    7 OFSET       3.87499
C    7 R    1      1.0
C    7 R    2-     1.0
C    7 R    3-     3.00000
C    7 R    4-     9.00000
C    7 R    5-    27.00000
C    7 R    6-     1.00000
C    8 OFSET-      3.87499
C    8 R    1      1.0
C    8 R    2      1.0
C    8 R    3      3.00000
C    8 R    4      9.00000
C    8 R    5     27.00000
C    8 R    6      1.00000
C    9 OFSET       5.33333
C    9 R    1      1.0
C    9 R    2-     1.0
C    9 R    3-     4.00000
C    9 R    4-    16.00000
C    9 R    5-    64.00000
C    9 R    6-     8.00000
C   10 OFSET-      5.33333
C   10 R    1      1.0
C   10 R    2      1.0
C   10 R    3      4.00000
C   10 R    4     16.00000
C   10 R    5     64.00000
C   10 R    6      8.00000
C   13 OFSET       1.24702
C   13 R    1      1.0
C   13 R    2-     1.0
C   13 R    3-     1.00000
C   13 R    4-     1.00000
C   13 R    5-     1.00000
C   13 R    7-     1.00000
C   13 R    8-     1.00000
C   13 R    9-     1.00000
```

1.0.0-2

C	13	R	10-	1.00000
C	13	R	12-	1.00000
C	13	R	13-	1.00000
C	13	R	14-	1.00000
C	13	R	15-	1.00000
C	13	R	17-	1.00000
C	13	R	18-	1.00000
C	13	R	19-	1.00000
C	13	R	20-	1.00000
C	14	OFSET-		1.24702
C	14	R	1	1.0
C	14	R	2	1.0
C	14	R	3	1.00000
C	14	R	4	1.00000
C	14	R	5	1.00000
C	14	R	7	1.00000
C	14	R	8	1.00000
C	14	R	9	1.00000
C	14	R	10	1.00000
C	14	R	12	1.00000
C	14	R	13	1.00000
C	14	R	14	1.00000
C	14	R	15	1.00000
C	14	R	17	1.00000
C	14	R	18	1.00000
C	14	R	19	1.00000
C	14	R	20	1.00000
C	15	OFSET		2.69791
C	15	R	1	1.0
C	15	R	2-	1.0
C	15	R	3-	2.00000
C	15	R	4-	4.00000
C	15	R	5-	8.00000
C	15	R	7-	1.00000
C	15	R	8-	2.00000
C	15	R	9-	4.00000
C	15	R	10-	8.00000
C	15	R	12-	1.00000
C	15	R	13-	2.00000
C	15	R	14-	4.00000
C	15	R	15-	8.00000

1.0.0-2

```
C  15 R   17-     1.00000
C  15 R   18-     2.00000
C  15 R   19-     4.00000
C  15 R   20-     8.00000
C  16 OFSET-      2.69791
C  16 R    1      1.0
C  16 R    2      1.0
C  16 R    3      2.00000
C  16 R    4      4.00000
C  16 R    5      8.00000
C  16 R    7      1.00000
C  16 R    8      2.00000
C  16 R    9      4.00000
C  16 R   10      8.00000
C  16 R   12      1.00000
C  16 R   13      2.00000
C  16 R   14      4.00000
C  16 R   15      8.00000
C  16 R   17      1.00000
C  16 R   18      2.00000
C  16 R   19      4.00000
C  16 R   20      8.00000
C  17 OFSET       4.19791
C  17 R    1      1.0
C  17 R    2-     1.0
C  17 R    3-     3.00000
C  17 R    4-     9.00000
C  17 R    5-    27.00000
C  17 R    6-     1.00000
C  17 R    7-     1.00000
C  17 R    8-     3.00000
C  17 R    9-     9.00000
C  17 R   10-    27.00000
C  17 R   11-     1.00000
C  17 R   12-     1.00000
C  17 R   13-     3.00000
C  17 R   14-     9.00000
C  17 R   15-    27.00000
C  17 R   16-     1.00000
C  17 R   17-     1.00000
C  17 R   18-     3.00000
```

```
C   17 R   19-     9.00000
C   17 R   20-    27.00000
C   17 R   21-     1.00000
C   18 OFSET-      4.19791
C   18 R    1      1.0
C   18 R    2      1.0
C   18 R    3      3.00000
C   18 R    4      9.00000
C   18 R    5     27.00000
C   18 R    6      1.00000
C   18 R    7      1.00000
C   18 R    8      3.00000
C   18 R    9      9.00000
C   18 R   10     27.00000
C   18 R   11      1.00000
C   18 R   12      1.00000
C   18 R   13      3.00000
C   18 R   14      9.00000
C   18 R   15     27.00000
C   18 R   16      1.00000
C   18 R   17      1.00000
C   18 R   18      3.00000
C   18 R   19      9.00000
C   18 R   20     27.00000
C   18 R   21      1.00000
C   19 OFSET       5.81250
C   19 R    1      1.0
C   19 R    2-     1.0
C   19 R    3-     4.00000
C   19 R    4-    16.00000
C   19 R    5-    64.00000
C   19 R    6-     8.00000
C   19 R    7-     1.00000
C   19 R    8-     4.00000
C   19 R    9-    16.00000
C   19 R   10-    64.00000
C   19 R   11-     8.00000
C   19 R   12-     1.00000
C   19 R   13-     4.00000
C   19 R   14-    16.00000
C   19 R   15-    64.00000
```

10.0-7

```
C  19 R   16-     8.00000
C  19 R   17-     1.00000
C  19 R   18-     4.00000
C  19 R   19-    16.00000
C  19 R   20-    64.00000
C  19 R   21-     8.00000
C  20 OFSET-      5.81250
C  20 R    1      1.0
C  20 R    2      1.0
C  20 R    3      4.00000
C  20 R    4     16.00000
C  20 R    5     64.00000
C  20 R    6      8.00000
C  20 R    7      1.00000
C  20 R    8      4.00000
C  20 R    9     16.00000
C  20 R   10     64.00000
C  20 R   11      8.00000
C  20 R   12      1.00000
C  20 R   13      4.00000
C  20 R   14     16.00000
C  20 R   15     64.00000
C  20 R   16      8.00000
C  20 R   17      1.00000
C  20 R   18      4.00000
C  20 R   19     16.00000
C  20 R   20     64.00000
C  20 R   21      8.00000
C  23 OFSET       1.47916
C  23 R    1      1.0
C  23 R    2-     1.0
C  23 R    3-     1.00000
C  23 R    4-     1.00000
C  23 R    5-     1.00000
C  23 R    7-     2.00000
C  23 R    8-     2.00000
C  23 R    9-     2.00000
C  23 R   10-     2.00000
C  23 R   12-     4.00000
C  23 R   13-     4.00000
C  23 R   14-     4.00000
```

```
C  23 R   15-    4.00000
C  23 R   17-    8.00000
C  23 R   18-    8.00000
C  23 R   19-    8.00000
C  23 R   20-    8.00000
C  24 OFSET-     1.47916
C  24 R    1     1.0
C  24 R    2     1.0
C  24 R    3     1.00000
C  24 R    4     1.00000
C  24 R    5     1.00000
C  24 R    7     2.00000
C  24 R    8     2.00000
C  24 R    9     2.00000
C  24 R   10     2.00000
C  24 R   12     4.00000
C  24 R   13     4.00000
C  24 R   14     4.00000
C  24 R   15     4.00000
C  24 R   17     8.00000
C  24 R   18     8.00000
C  24 R   19     8.00000
C  24 R   20     8.00000
C  25 OFSET      2.97916
C  25 R    1     1.0
C  25 R    2-    1.0
C  25 R    3-    2.00000
C  25 R    4-    4.00000
C  25 R    5-    8.00000
C  25 R    7-    2.00000
C  25 R    8-    4.00000
C  25 R    9-    8.00000
C  25 R   10-   16.00000
C  25 R   12-    4.00000
C  25 R   13-    8.00000
C  25 R   14-   16.00000
C  25 R   15-   32.00000
C  25 R   17-    8.00000
C  25 R   18-   16.00000
C  25 R   19-   32.00000
C  25 R   20-   64.00000
```

1.0.0.2

```
C   26 OFSET-     2.97916
C   26 R    1     1.0
C   26 R    2     1.0
C   26 R    3     2.00000
C   26 R    4     4.00000
C   26 R    5     8.00000
C   26 R    7     2.00000
C   26 R    8     4.00000
C   26 R    9     8.00000
C   26 R   10    16.00000
C   26 R   12     4.00000
C   26 R   13     8.00000
C   26 R   14    16.00000
C   26 R   15    32.00000
C   26 R   17     8.00000
C   26 R   18    16.00000
C   26 R   19    32.00000
C   26 R   20    64.00000
C   27 OFSET      4.58333
C   27 R    1     1.0
C   27 R    2-    1.0
C   27 R    3-    3.00000
C   27 R    4-    9.00000
C   27 R    5-   27.00000
C   27 R    6-    1.00000
C   27 R    7-    2.00000
C   27 R    8-    6.00000
C   27 R    9-   18.00000
C   27 R   10-   54.00000
C   27 R   11-    2.00000
C   27 R   12-    4.00000
C   27 R   13-   12.00000
C   27 R   14-   36.00000
C   27 R   15-  108.00000
C   27 R   16-    4.00000
C   27 R   17-    8.00000
C   27 R   18-   24.00000
C   27 R   19-   72.00000
C   27 R   20-  216.00000
C   27 R   21-    8.00000
C   28 OFSET-     4.58333
```

```
C   28 R     1      1.0
C   28 R     2      1.0
C   28 R     3      3.00000
C   28 R     4      9.00000
C   28 R     5     27.00000
C   28 R     6      1.00000
C   28 R     7      2.00000
C   28 R     8      6.00000
C   28 R     9     18.00000
C   28 R    10     54.00000
C   28 R    11      2.00000
C   28 R    12      4.00000
C   28 R    13     12.00000
C   28 R    14     36.00000
C   28 R    15    108.00000
C   28 R    16      4.00000
C   28 R    17      8.00000
C   28 R    18     24.00000
C   28 R    19     72.00000
C   28 R    20    216.00000
C   28 R    21      8.00000
C   29 OFSET        6.33333
C   29 R     1      1.0
C   29 R     2-     1.0
C   29 R     3-     4.00000
C   29 R     4-    16.00000
C   29 R     5-    64.00000
C   29 R     6-     8.00000
C   29 R     7-     2.00000
C   29 R     8-     8.00000
C   29 R     9-    32.00000
C   29 R    10-   128.00000
C   29 R    11-    16.00000
C   29 R    12-     4.00000
C   29 R    13-    16.00000
C   29 R    14-    64.00000
C   29 R    15-   256.00000
C   29 R    16-    32.00000
C   29 R    17-     8.00000
C   29 R    18-    32.00000
C   29 R    19-   128.00000
```

1.0.0.2

```
C  29 R   20-   512.00000
C  29 R   21-    64.00000
C  30 OFSET-      6.33333
C  30 R    1      1.0
C  30 R    2      1.0
C  30 R    3      4.00000
C  30 R    4     16.00000
C  30 R    5     64.00000
C  30 R    6      8.00000
C  30 R    7      2.00000
C  30 R    8      8.00000
C  30 R    9     32.00000
C  30 R   10    128.00000
C  30 R   11     16.00000
C  30 R   12      4.00000
C  30 R   13     16.00000
C  30 R   14     64.00000
C  30 R   15    256.00000
C  30 R   16     32.00000
C  30 R   17      8.00000
C  30 R   18     32.00000
C  30 R   19    128.00000
C  30 R   20    512.00000
C  30 R   21     64.00000
C  31 OFSET        .34374
C  31 R    1      1.0
C  31 R    2-     1.0
C  31 R    7-     3.00000
C  31 R   12-     9.00000
C  31 R   17-    27.00000
C  31 R   22-     1.00000
C  32 OFSET-       .34374
C  32 R    1      1.0
C  32 R    2      1.0
C  32 R    7      3.00000
C  32 R   12      9.00000
C  32 R   17     27.00000
C  32 R   22      1.00000
C  33 OFSET       1.79166
C  33 R    1      1.0
C  33 R    2-     1.0
```

```
C  33 R    3-     1.00000
C  33 R    4-     1.00000
C  33 R    5-     1.00000
C  33 R    7-     3.00000
C  33 R    8-     3.00000
C  33 R    9-     3.00000
C  33 R   10-     3.00000
C  33 R   12-     9.00000
C  33 R   13-     9.00000
C  33 R   14-     9.00000
C  33 R   15-     9.00000
C  33 R   17-    27.00000
C  33 R   18-    27.00000
C  33 R   19-    27.00000
C  33 R   20-    27.00000
C  33 R   22-     1.00000
C  33 R   23-     1.00000
C  33 R   24-     1.00000
C  33 R   25-     1.00000
C  34 OFSET-      1.79166
C  34 R    1      1.0
C  34 R    2      1.0
C  34 R    3      1.00000
C  34 R    4      1.00000
C  34 R    5      1.00000
C  34 R    7      3.00000
C  34 R    8      3.00000
C  34 R    9      3.00000
C  34 R   10      3.00000
C  34 R   12      9.00000
C  34 R   13      9.00000
C  34 R   14      9.00000
C  34 R   15      9.00000
C  34 R   17     27.00000
C  34 R   18     27.00000
C  34 R   19     27.00000
C  34 R   20     27.00000
C  34 R   22      1.00000
C  34 R   23      1.00000
C  34 R   24      1.00000
C  34 R   25      1.00000
```

1.0.0.2

```
C   35 OFSET      3.44791
C   35 R    1     1.0
C   35 R    2-    1.0
C   35 R    3-    2.00000
C   35 R    4-    4.00000
C   35 R    5-    8.00000
C   35 R    7-    3.00000
C   35 R    8-    6.00000
C   35 R    9-   12.00000
C   35 R   10-   24.00000
C   35 R   12-    9.00000
C   35 R   13-   18.00000
C   35 R   14-   36.00000
C   35 R   15-   72.00000
C   35 R   17-   27.00000
C   35 R   18-   54.00000
C   35 R   19-  108.00000
C   35 R   20-  216.00000
C   35 R   22-    1.00000
C   35 R   23-    2.00000
C   35 R   24-    4.00000
C   35 R   25-    8.00000
C   36 OFSET-     3.44791
C   36 R    1     1.0
C   36 R    2     1.0
C   36 R    3     2.00000
C   36 R    4     4.00000
C   36 R    5     8.00000
C   36 R    7     3.00000
C   36 R    8     6.00000
C   36 R    9    12.00000
C   36 R   10    24.00000
C   36 R   12     9.00000
C   36 R   13    18.00000
C   36 R   14    36.00000
C   36 R   15    72.00000
C   36 R   17    27.00000
C   36 R   18    54.00000
C   36 R   19   108.00000
C   36 R   20   216.00000
C   36 R   22     1.00000
```

```
C  36 R   23      2.00000
C  36 R   24      4.00000
C  36 R   25      8.00000
C  37 OFSET       5.25000
C  37 R    1      1.0
C  37 R    2-     1.0
C  37 R    3-     3.00000
C  37 R    4-     9.00000
C  37 R    5-    27.00000
C  37 R    6-     1.00000
C  37 R    7-     3.00000
C  37 R    8-     9.00000
C  37 R    9-    27.00000
C  37 R   10-    81.00000
C  37 R   11-     3.00000
C  37 R   12-     9.00000
C  37 R   13-    27.00000
C  37 R   14-    81.00000
C  37 R   15-   243.00000
C  37 R   16-     9.00000
C  37 R   17-    27.00000
C  37 R   18-    81.00000
C  37 R   19-   243.00000
C  37 R   20-   729.00000
C  37 R   21-    27.00000
C  37 R   22-     1.00000
C  37 R   23-     3.00000
C  37 R   24-     9.00000
C  37 R   25-    27.00000
C  37 R   26-     1.00000
C  38 OFSET-      5.25000
C  38 R    1      1.0
C  38 R    2      1.0
C  38 R    3      3.00000
C  38 R    4      9.00000
C  38 R    5     27.00000
C  38 R    6      1.00000
C  38 R    7      3.00000
C  38 R    8      9.00000
C  38 R    9     27.00000
C  38 R   10     81.00000
```

1.0.0.2

```
C   38 R   11       3.00000
C   38 R   12       9.00000
C   38 R   13      27.00000
C   38 R   14      81.00000
C   38 R   15     243.00000
C   38 R   16       9.00000
C   38 R   17      27.00000
C   38 R   18      81.00000
C   38 R   19     243.00000
C   38 R   20     729.00000
C   38 R   21      27.00000
C   38 R   22       1.00000
C   38 R   23       3.00000
C   38 R   24       9.00000
C   38 R   25      27.00000
C   38 R   26       1.00000
C   39 OFSET        7.16666
C   39 R    1       1.0
C   39 R    2-      1.0
C   39 R    3-      4.00000
C   39 R    4-     16.00000
C   39 R    5-     64.00000
C   39 R    6-      8.00000
C   39 R    7-      3.00000
C   39 R    8-     12.00000
C   39 R    9-     48.00000
C   39 R   10-    192.00000
C   39 R   11-     24.00000
C   39 R   12-      9.00000
C   39 R   13-     36.00000
C   39 R   14-    144.00000
C   39 R   15-    576.00000
C   39 R   16-     72.00000
C   39 R   17-     27.00000
C   39 R   18-    108.00000
C   39 R   19-    432.00000
C   39 R   20-   1728.00000
C   39 R   21-    216.00000
C   39 R   22-      1.00000
C   39 R   23-      4.00000
C   39 R   24-     16.00000
```

1.0.0-2

```
C   39 R   25-     64.00000
C   39 R   26-      8.00000
C   40 OFSET-       7.16666
C   40 R    1       1.0
C   40 R    2       1.0
C   40 R    3       4.00000
C   40 R    4      16.00000
C   40 R    5      64.00000
C   40 R    6       8.00000
C   40 R    7       3.00000
C   40 R    8      12.00000
C   40 R    9      48.00000
C   40 R   10     192.00000
C   40 R   11      24.00000
C   40 R   12       9.00000
C   40 R   13      36.00000
C   40 R   14     144.00000
C   40 R   15     576.00000
C   40 R   16      72.00000
C   40 R   17      27.00000
C   40 R   18     108.00000
C   40 R   19     432.00000
C   40 R   20    1728.00000
C   40 R   21     216.00000
C   40 R   22       1.00000
C   40 R   23       4.00000
C   40 R   24      16.00000
C   40 R   25      64.00000
C   40 R   26       8.00000
C   41 OFSET         .70833
C   41 R    1       1.0
C   41 R    2-      1.0
C   41 R    7-      4.00000
C   41 R   12-     16.00000
C   41 R   17-     64.00000
C   41 R   22-      8.00000
C   42 OFSET-        .70833
C   42 R    1       1.0
C   42 R    2       1.0
C   42 R    7       4.00000
C   42 R   12      16.00000
```

1.0.0-2

C	42	R	17	64.00000
C	42	R	22	8.00000
C	43	OFSET		2.41666
C	43	R	1	1.0
C	43	R	2-	1.0
C	43	R	3-	1.00000
C	43	R	4-	1.00000
C	43	R	5-	1.00000
C	43	R	7-	4.00000
C	43	R	8-	4.00000
C	43	R	9-	4.00000
C	43	R	10-	4.00000
C	43	R	12-	16.00000
C	43	R	13-	16.00000
C	43	R	14-	16.00000
C	43	R	15-	16.00000
C	43	R	17-	64.00000
C	43	R	18-	64.00000
C	43	R	19-	64.00000
C	43	R	20-	64.00000
C	43	R	22-	8.00000
C	43	R	23-	8.00000
C	43	R	24-	8.00000
C	43	R	25-	8.00000
C	44	OFSET-		2.41666
C	44	R	1	1.0
C	44	R	2	1.0
C	44	R	3	1.00000
C	44	R	4	1.00000
C	44	R	5	1.00000
C	44	R	7	4.00000
C	44	R	8	4.00000
C	44	R	9	4.00000
C	44	R	10	4.00000
C	44	R	12	16.00000
C	44	R	13	16.00000
C	44	R	14	16.00000
C	44	R	15	16.00000
C	44	R	17	64.00000
C	44	R	18	64.00000
C	44	R	19	64.00000

C	44	R	20	64.00000
C	44	R	22	8.00000
C	44	R	23	8.00000
C	44	R	24	8.00000
C	44	R	25	8.00000
C	45	OFSET		4.32291
C	45	R	1	1.0
C	45	R	2-	1.0
C	45	R	3-	2.00000
C	45	R	4-	4.00000
C	45	R	5-	8.00000
C	45	R	7-	4.00000
C	45	R	8-	8.00000
C	45	R	9-	16.00000
C	45	R	10-	32.00000
C	45	R	12-	16.00000
C	45	R	13-	32.00000
C	45	R	14-	64.00000
C	45	R	15-	128.00000
C	45	R	17-	64.00000
C	45	R	18-	128.00000
C	45	R	19-	256.00000
C	45	R	20-	512.00000
C	45	R	22-	8.00000
C	45	R	23-	16.00000
C	45	R	24-	32.00000
C	45	R	25-	64.00000
C	46	OFSET-		4.32291
C	46	R	1	1.0
C	46	R	2	1.0
C	46	R	3	2.00000
C	46	R	4	4.00000
C	46	R	5	8.00000
C	46	R	7	4.00000
C	46	R	8	8.00000
C	46	R	9	16.00000
C	46	R	10	32.00000
C	46	R	12	16.00000
C	46	R	13	32.00000
C	46	R	14	64.00000
C	46	R	15	128.00000

1.0.0-2

```
C   46 R   17     64.00000
C   46 R   18    128.00000
C   46 R   19    256.00000
C   46 R   20    512.00000
C   46 R   22      8.00000
C   46 R   23     16.00000
C   46 R   24     32.00000
C   46 R   25     64.00000
C   47 OFSET      6.41666
C   47 R    1      1.0
C   47 R    2-     1.0
C   47 R    3-     3.00000
C   47 R    4-     9.00000
C   47 R    5-    27.00000
C   47 R    6-     1.00000
C   47 R    7-     4.00000
C   47 R    8-    12.00000
C   47 R    9-    36.00000
C   47 R   10-   108.00000
C   47 R   11-     4.00000
C   47 R   12-    16.00000
C   47 R   13-    48.00000
C   47 R   14-   144.00000
C   47 R   15-   432.00000
C   47 R   16-    16.00000
C   47 R   17-    64.00000
C   47 R   18-   192.00000
C   47 R   19-   576.00000
C   47 R   20-  1728.00000
C   47 R   21-    64.00000
C   47 R   22-     8.00000
C   47 R   23-    24.00000
C   47 R   24-    72.00000
C   47 R   25-   216.00000
C   47 R   26-     8.00000
C   48 OFSET-     6.41666
C   48 R    1      1.0
C   48 R    2      1.0
C   48 R    3      3.00000
C   48 R    4      9.00000
C   48 R    5     27.00000
```

B-112

1.0.0-2

C	48	R	6	1.00000
C	48	R	7	4.00000
C	48	R	8	12.00000
C	48	R	9	36.00000
C	48	R	10	108.00000
C	48	R	11	4.00000
C	48	R	12	16.00000
C	48	R	13	48.00000
C	48	R	14	144.00000
C	48	R	15	432.00000
C	48	R	16	16.00000
C	48	R	17	64.00000
C	48	R	18	192.00000
C	48	R	19	576.00000
C	48	R	20	1728.00000
C	48	R	21	64.00000
C	48	R	22	8.00000
C	48	R	23	24.00000
C	48	R	24	72.00000
C	48	R	25	216.00000
C	48	R	26	8.00000
C	49	OFSET		8.58333
C	49	R	1	1.0
C	49	R	2-	1.0
C	49	R	3-	4.00000
C	49	R	4-	16.00000
C	49	R	5-	64.00000
C	49	R	6-	8.00000
C	49	R	7-	4.00000
C	49	R	8-	16.00000
C	49	R	9-	64.00000
C	49	R	10-	256.00000
C	49	R	11-	32.00000
C	49	R	12-	16.00000
C	49	R	13-	64.00000
C	49	R	14-	256.00000
C	49	R	15-	1024.00000
C	49	R	16-	128.00000
C	49	R	17-	64.00000
C	49	R	18-	256.00000
C	49	R	19-	1024.00000

```
C  49 R  20-  4096.00000
C  49 R  21-   512.00000
C  49 R  22-     8.00000
C  49 R  23-    32.00000
C  49 R  24-   128.00000
C  49 R  25-   512.00000
C  49 R  26-    64.00000
C  50 OFSET-     8.58333
C  50 R    1      1.0
C  50 R    2      1.0
C  50 R    3      4.00000
C  50 R    4     16.00000
C  50 R    5     64.00000
C  50 R    6      8.00000
C  50 R    7      4.00000
C  50 R    8     16.00000
C  50 R    9     64.00000
C  50 R   10    256.00000
C  50 R   11     32.00000
C  50 R   12     16.00000
C  50 R   13     64.00000
C  50 R   14    256.00000
C  50 R   15   1024.00000
C  50 R   16    128.00000
C  50 R   17     64.00000
C  50 R   18    256.00000
C  50 R   19   1024.00000
C  50 R   20   4096.00000
C  50 R   21    512.00000
C  50 R   22      8.00000
C  50 R   23     32.00000
C  50 R   24    128.00000
C  50 R   25    512.00000
C  50 R   26     64.00000
50    3    2     5    5    5     4    4   26     1     0.00000000        .00100000
     .33333333     20.00000000       .07843137     3.00000000
     0.00000000     2.60400380
    20.00000000
    20.00000000
     1.00000000
     1.00000000
```

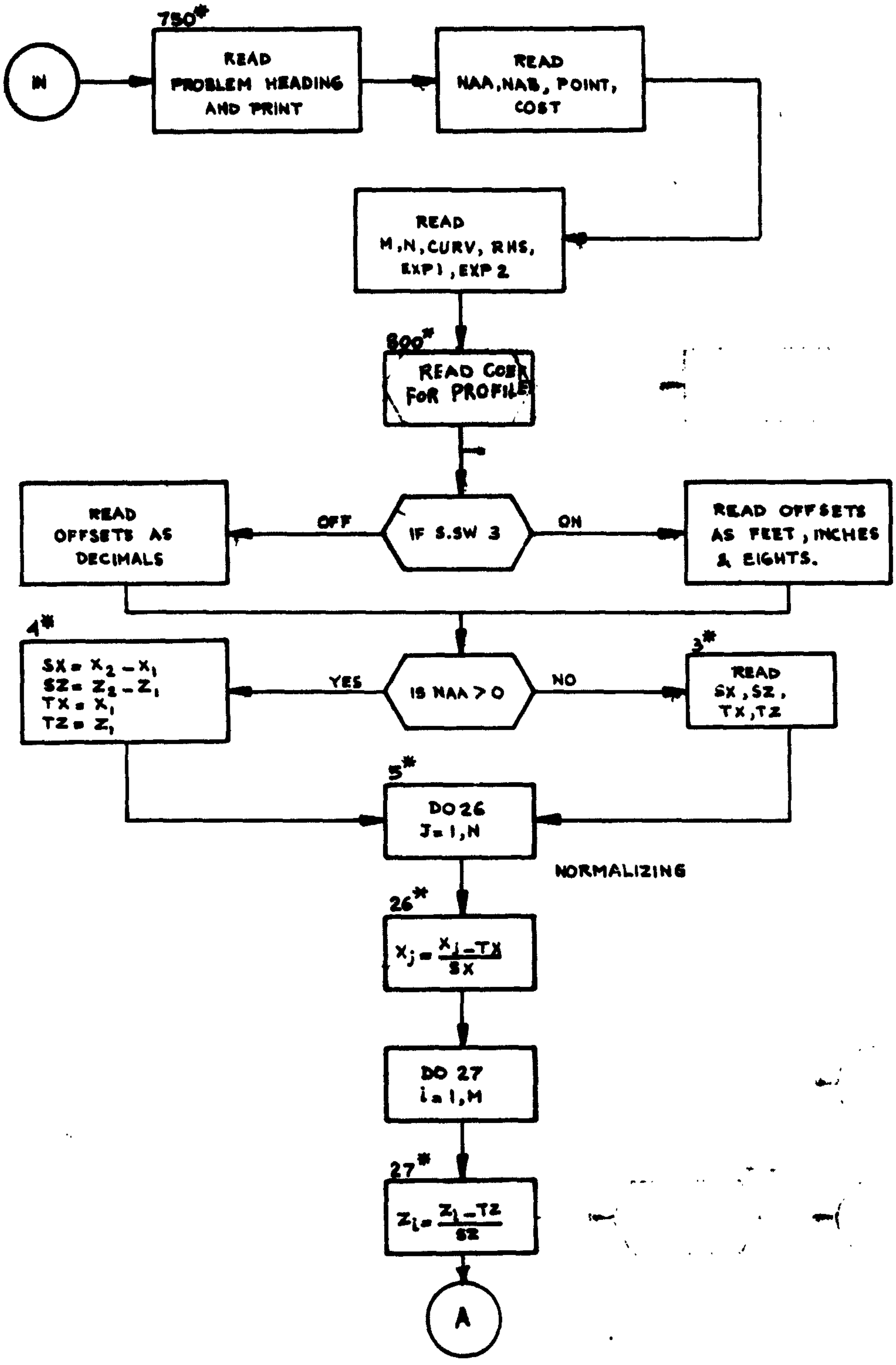
1.0.0.2

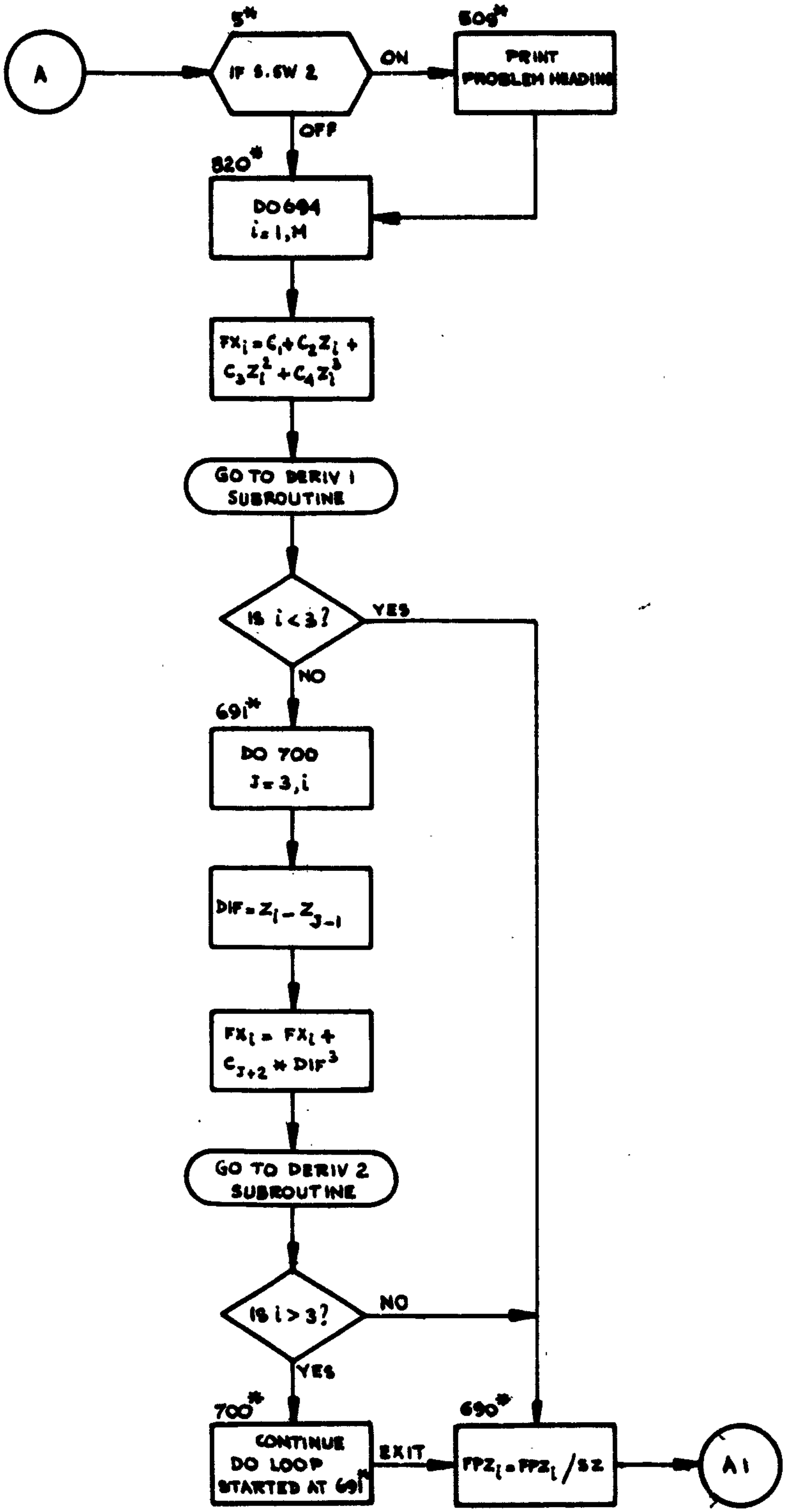
1.0.0-2

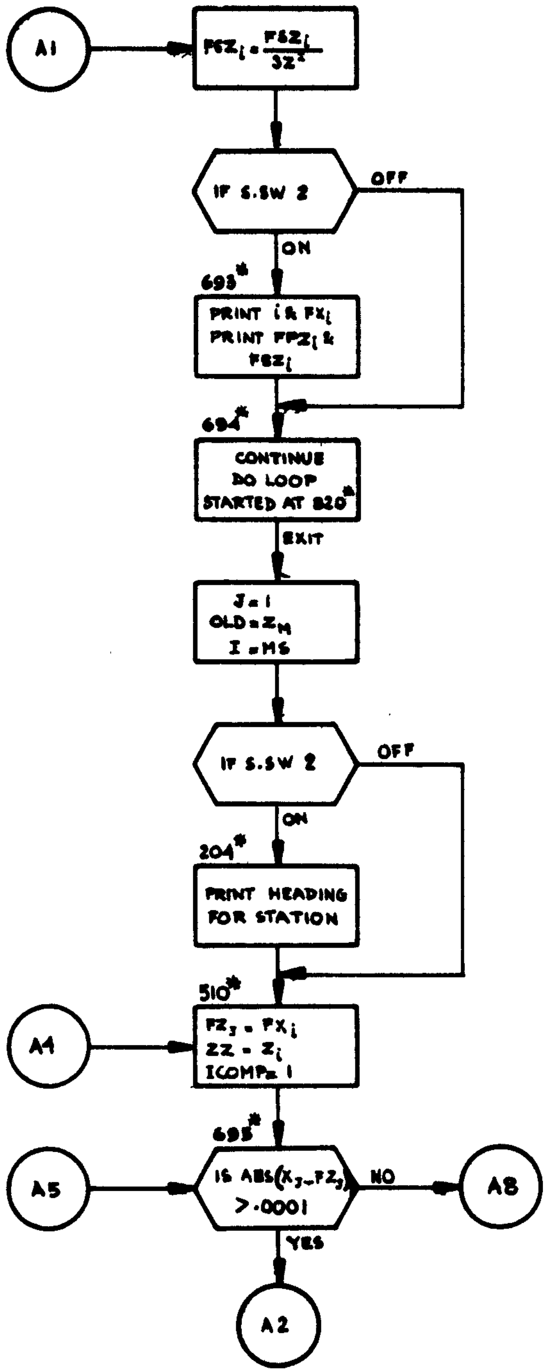
1.00000000	.90136719		
1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
2.00000000	0.00000000		
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
3.00000000	0.00000000		
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
4.00000000	0.00000000		
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
0.00000000	1.52935000	-.58730000	0.00000000
20.00000000			
20.00000000			
1.00000000			
-1.00000000			
-1.00000000			
1.00000000	.94205000	-.58730000	0.00000000
20.00000000			
1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
2.00000000	.35475000	-.58730000	0.00000000
20.00000000			
1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
3.00000000	-.23255000	-.58730000	0.00000000
1.00000000			
-1.00000000			

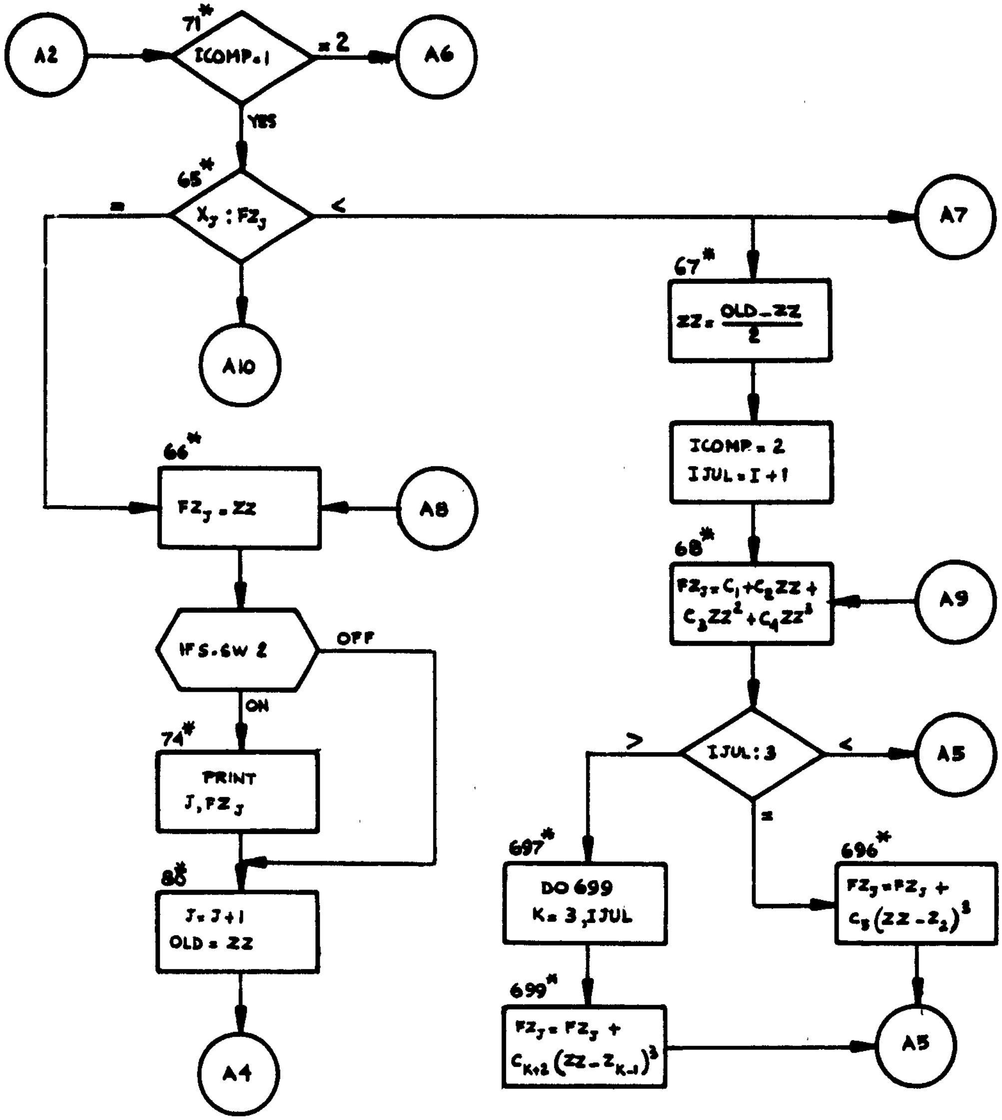
-1.00000000			
-1.00000000			
-1.00000000			
4.00000000	-.81985000	-.58730000	0.00000000
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
1.52935000			
-.58730000			
0.00000000			
0.00000000			
0.00000000			
0.00000000			
0.00000000			

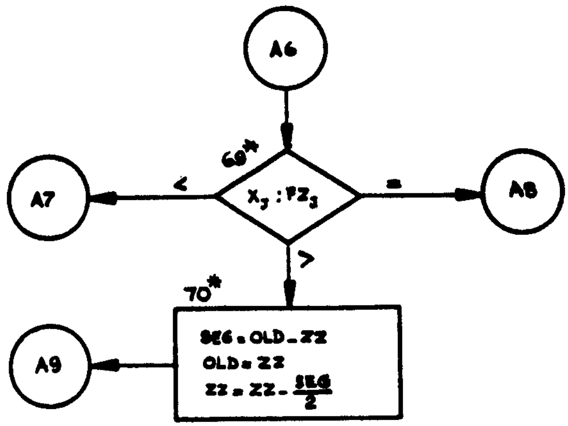
FLOW DIAGRAM - SMOG 2, PASS 1

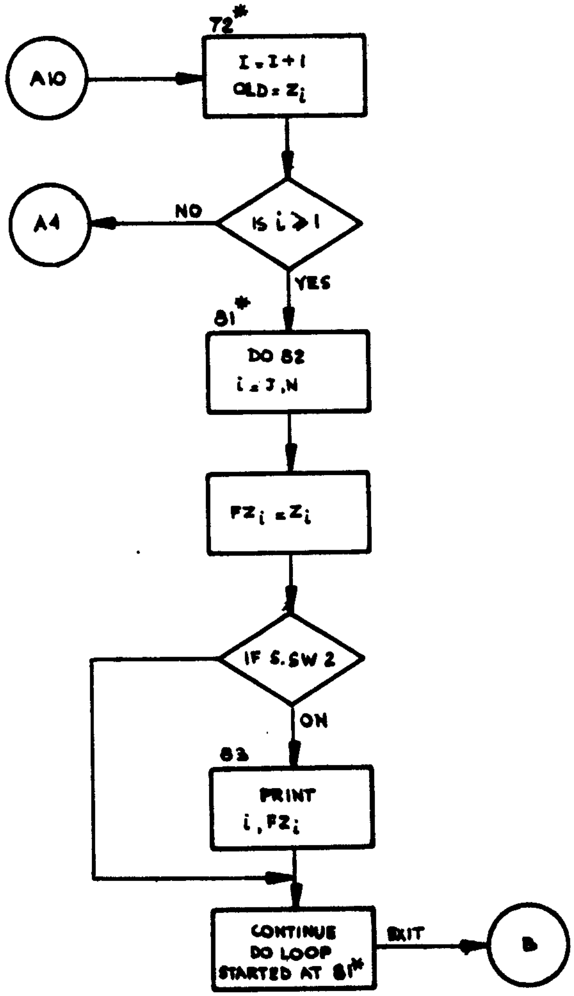


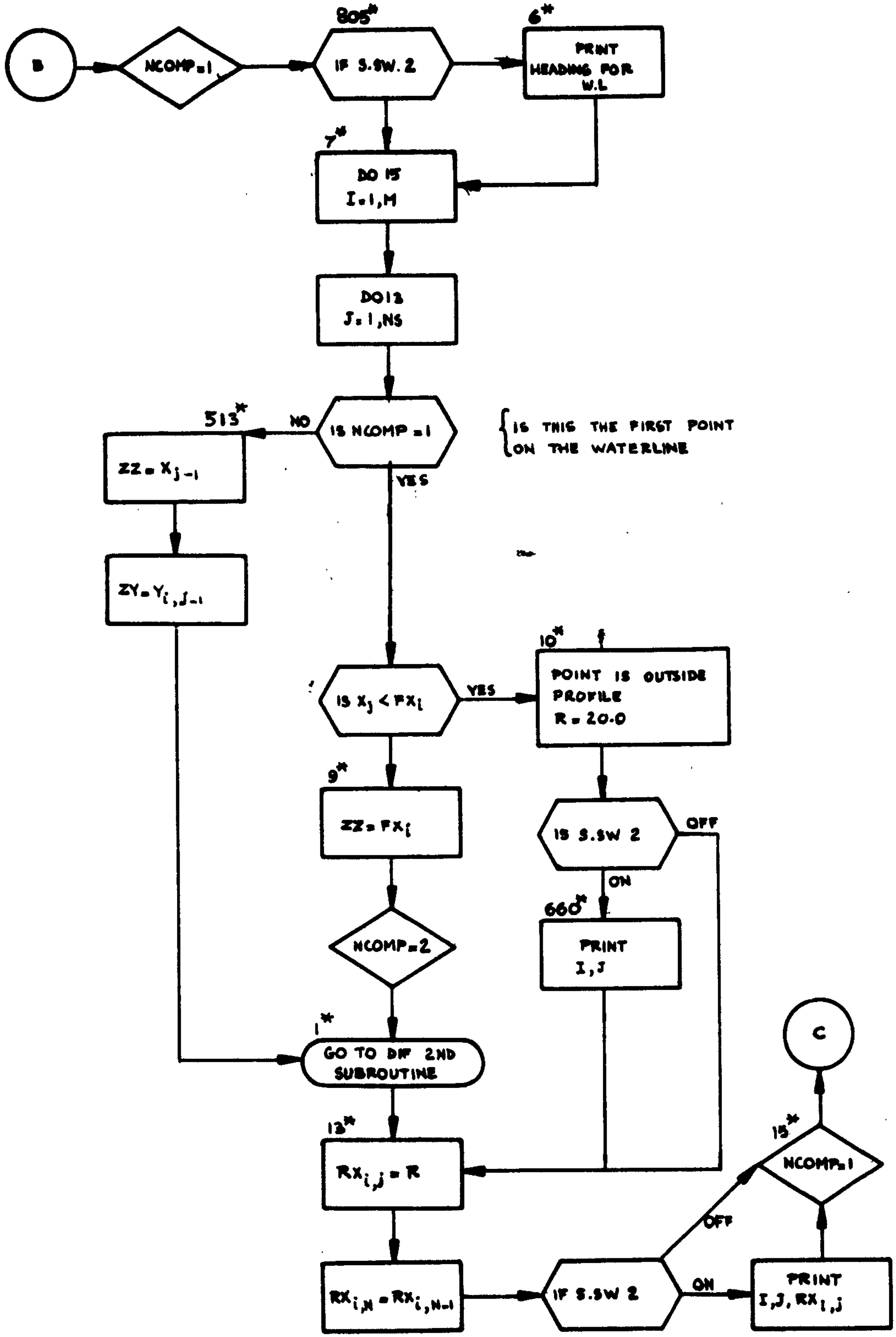

A
5*
IF S.SW 2
ON
509*
PRINT PROBLEM HEADING
OFF
820*
DO 694 i=1,M
FX_i = C_1 + C_2 Z_i + C_3 Z_i^2 + C_4 Z_i^3
GO TO DERIV 1 SUBROUTINE
IS i < 3?
YES
NO
691*
DO 700 J=3,i
DIF = Z_i - Z_J-1
FX_i = FX_i + C_J+2 * DIF^3
GO TO DERIV 2 SUBROUTINE
IS i > 3?
NO
YES
700*
CONTINUE DO LOOP STARTED AT 691*
EXIT
690*
FPZ_i = FPZ_i / SZ
A1


A1
FSZi = FSZi / 32²
IF S.SW 2
OFF
ON
693*
PRINT i & FXi
PRINT FPZi &
FSZi
694*
CONTINUE
DO LOOP
STARTED AT 820*
EXIT
J = 1
OLD = ZM
I = MS
IF S.SW 2
OFF
ON
204*
PRINT HEADING
FOR STATION
510*
A4
FZJ = FXi
ZZ = Zi
ICOMP = 1
695*
A5
IS ABS(XJ - FZJ)
> .0001
NO
A8
YES
A2

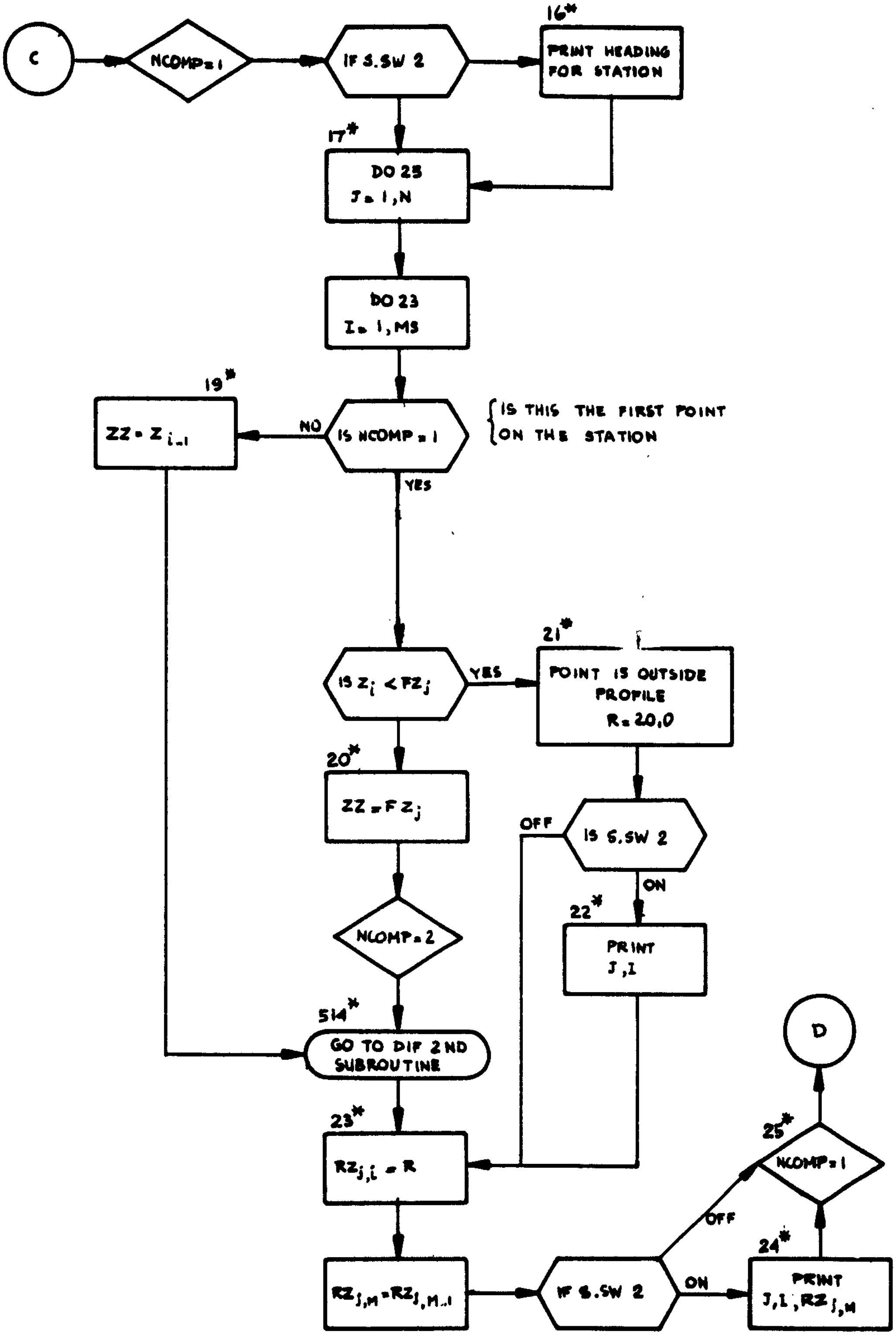

A2
71*
ICOMP=1
=2
A6
YES
65*
=
$X_J : FZ_J$
<
A7
A10
67*
$ZZ = \frac{OLD - ZZ}{2}$
ICOMP = 2
IJUL = I + 1
66*
$FZ_J = ZZ$
A8
68*
$FZ_J = C_1 + C_2 ZZ + C_3 ZZ^2 + C_4 ZZ^3$
A9
IFS.SW 2
OFF
ON
74*
PRINT
J, FZ_J
>
IJUL:3
<
A5
=
697*
696*
80*
J = J + 1
OLD = ZZ
DO 699
K = 3, IJUL
$FZ_J = FZ_J + C_5 (ZZ - Z_2)^3$
699*
$FZ_J = FZ_J + C_{K+2} (ZZ - Z_{K-1})^3$
A5
A4

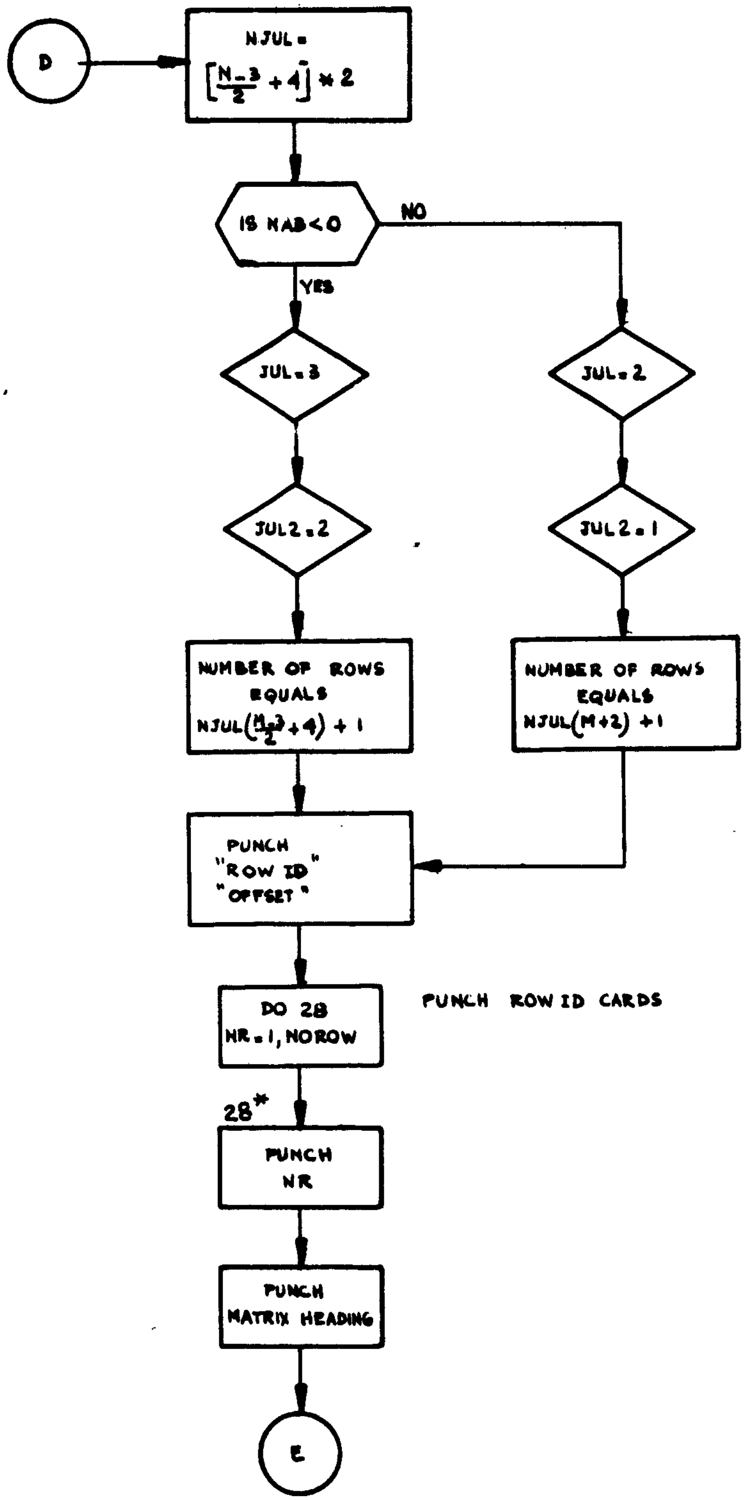

A6
69*
$X_J : FZ_J$
<
=
>
A7
A8
70*
SEG = OLD − ZZ
OLD = ZZ
ZZ = ZZ − SEG/2
A9

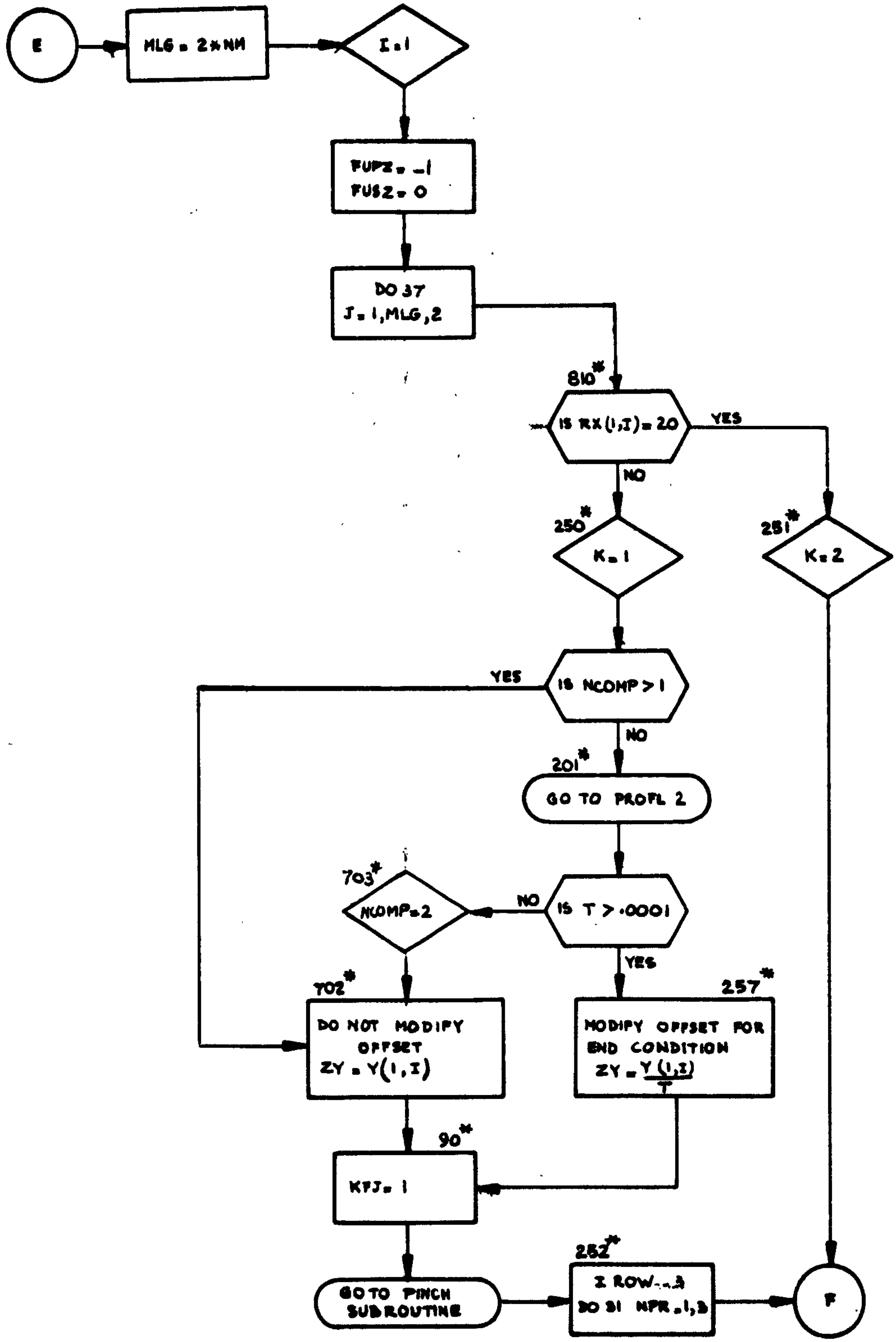

A10
72*
I = I + 1
OLD = Z_i
A4
NO
IS i > 1
YES
81*
DO 82
i = J, N
$FZ_i = Z_i$
IF S.SW 2
ON
82
PRINT
i, FZ_i
CONTINUE
DO LOOP
STARTED AT 81*
EXIT
B

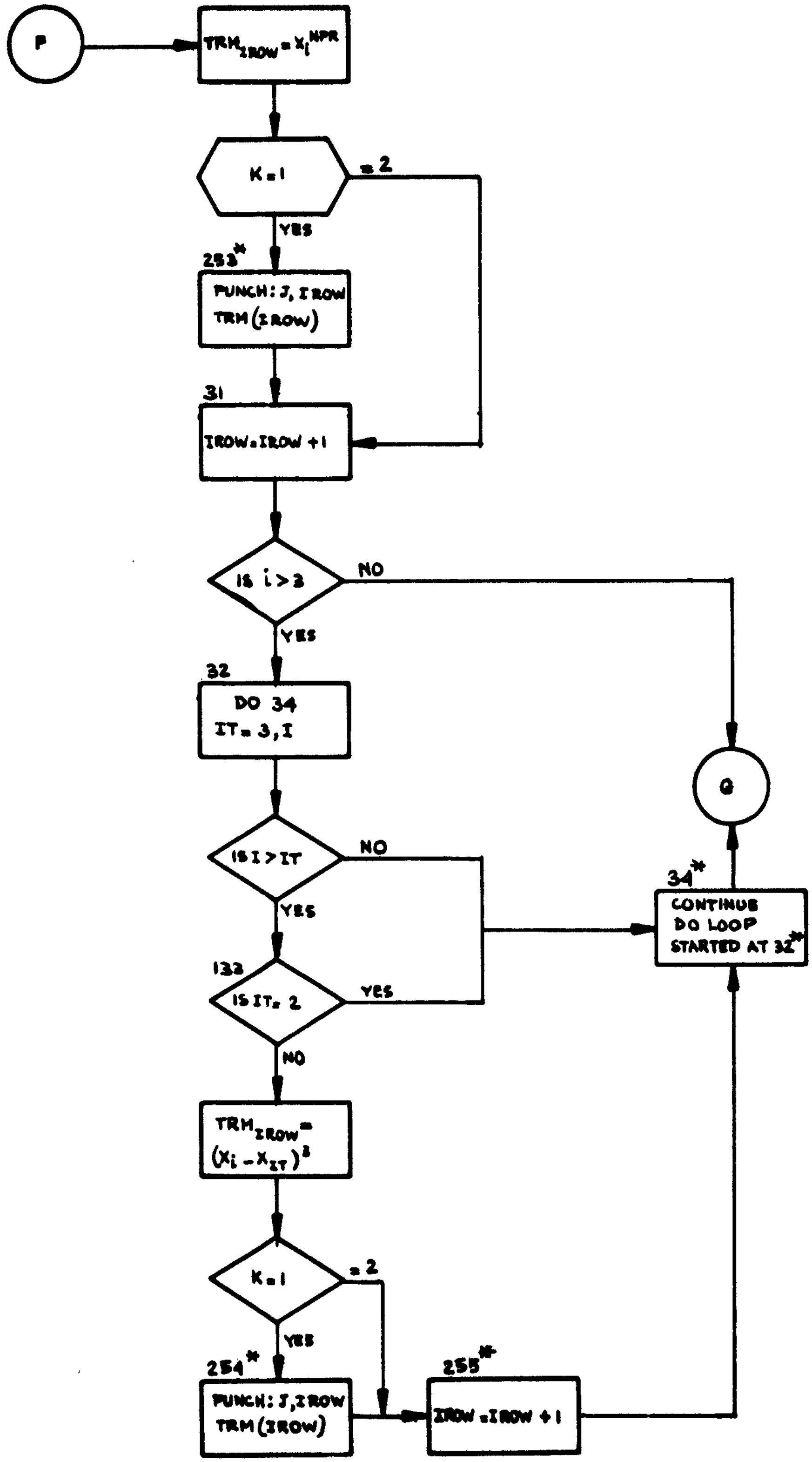


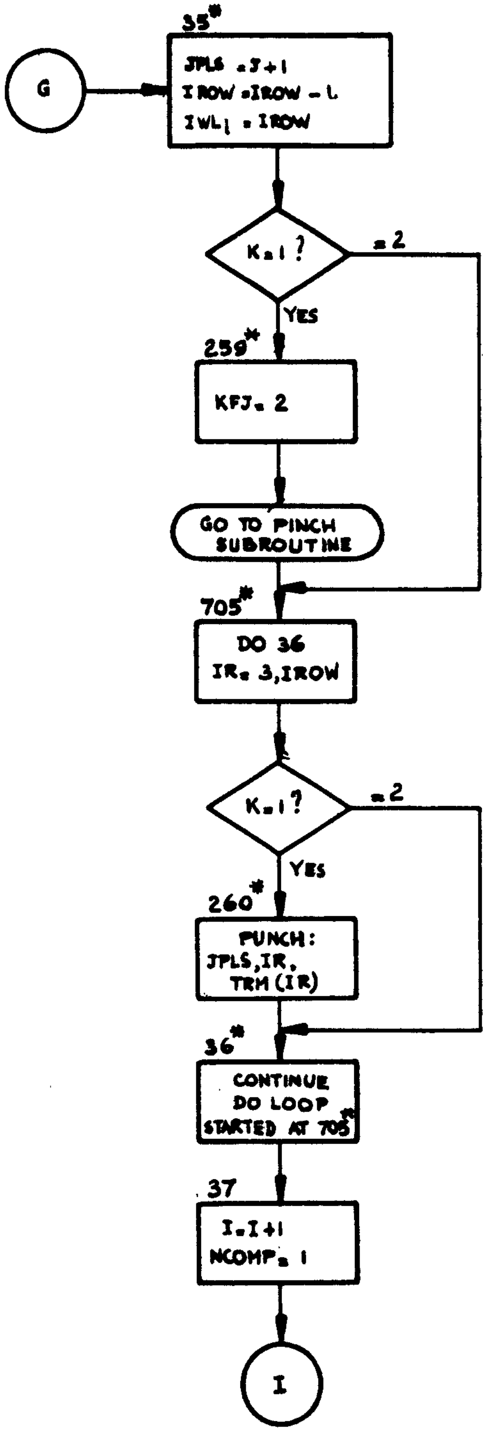
B
NCOMP=1
805*
IF S.SW. 2
6*
PRINT HEADING FOR W.L
7*
DO 15 I=1,M
DO 13 J=1,NS
513*
NO
IS NCOMP =1
IS THIS THE FIRST POINT ON THE WATERLINE
YES
ZZ = X j-1
ZY = Y i,j-1
10*
IS Xj < FXi
YES
POINT IS OUTSIDE PROFILE R = 20.0
9*
ZZ = FXi
IS S.SW 2
OFF
ON
660*
PRINT I,J
NCOMP=2
1*
GO TO DIF 2ND SUBROUTINE
C
13*
RXi,j = R
15*
NCOMP=1
OFF
RXi,N = RXi,N-1
IF S.SW 2
ON
PRINT I,J, RXi,j


C
NCOMP=1
IF S.SW 2
16*
PRINT HEADING FOR STATION
17*
DO 25 J=1,N
DO 23 I=1,MS
19*
ZZ=Z i-1
NO
IS NCOMP=1
IS THIS THE FIRST POINT ON THE STATION
YES
IS Zi < FZj
YES
21*
POINT IS OUTSIDE PROFILE R=20.0
20*
ZZ=FZj
OFF
IS S.SW 2
ON
22*
PRINT J,I
NCOMP=2
514*
GO TO DIF 2ND SUBROUTINE
D
23*
RZj,i=R
25*
NCOMP=1
OFF
24*
RZj,M=RZj,M-1
IF S.SW 2
ON
PRINT J,I,RZj,M

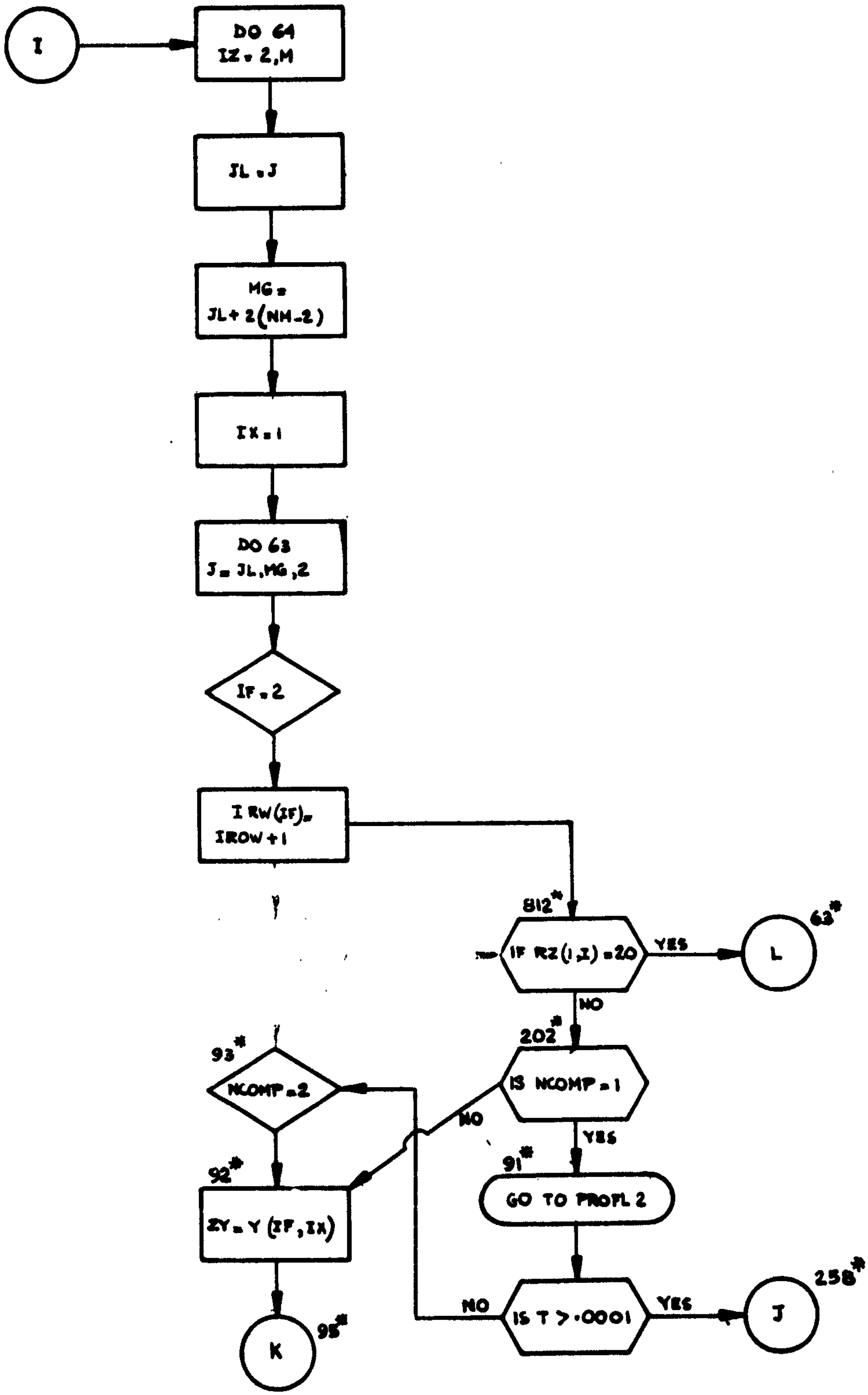

D
NJUL = [(N-3)/2 + 4] * 2
IS NAB<0
NO
YES
JUL=3
JUL=2
JUL2=2
JUL2=1
NUMBER OF ROWS EQUALS NJUL((N-3)/2+4) + 1
NUMBER OF ROWS EQUALS NJUL(M+2) +1
PUNCH "ROW ID" "OFFSET"
DO 28 NR=1,NOROW
PUNCH ROW ID CARDS
28*
PUNCH NR
PUNCH MATRIX HEADING
E

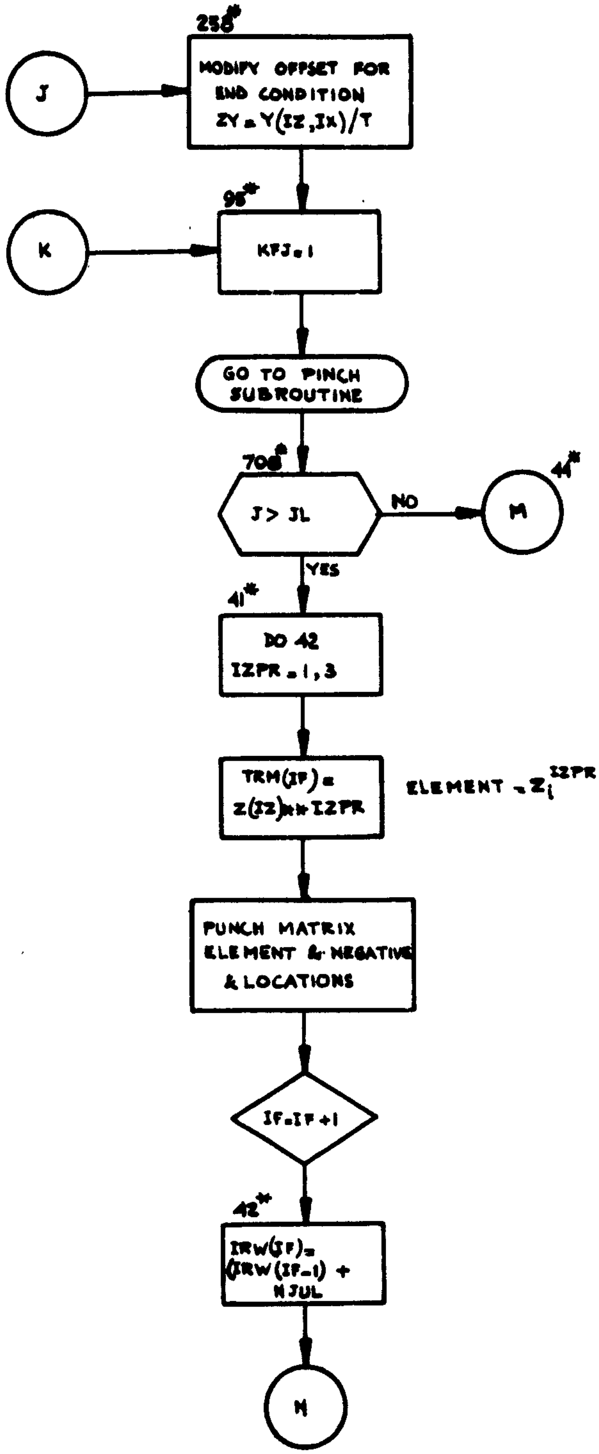

E
MLG = 2*NM
I=1
FUPZ = -1
FUSZ = 0
DO 37
J = 1,MLG,2
810*
IS RX(1,I) = 20
YES
NO
250*
K = 1
251*
K = 2
IS NCOMP > 1
YES
NO
201*
GO TO PROFL 2
703*
NCOMP = 2
NO
IS T > .0001
YES
702*
DO NOT MODIFY OFFSET
ZY = Y(1,I)
257*
MODIFY OFFSET FOR END CONDITION
ZY = Y(1,I)/T
90*
KFJ = 1
GO TO PINCH SUBROUTINE
252*
I ROW = 3
DO 31 NPR = 1,3
F

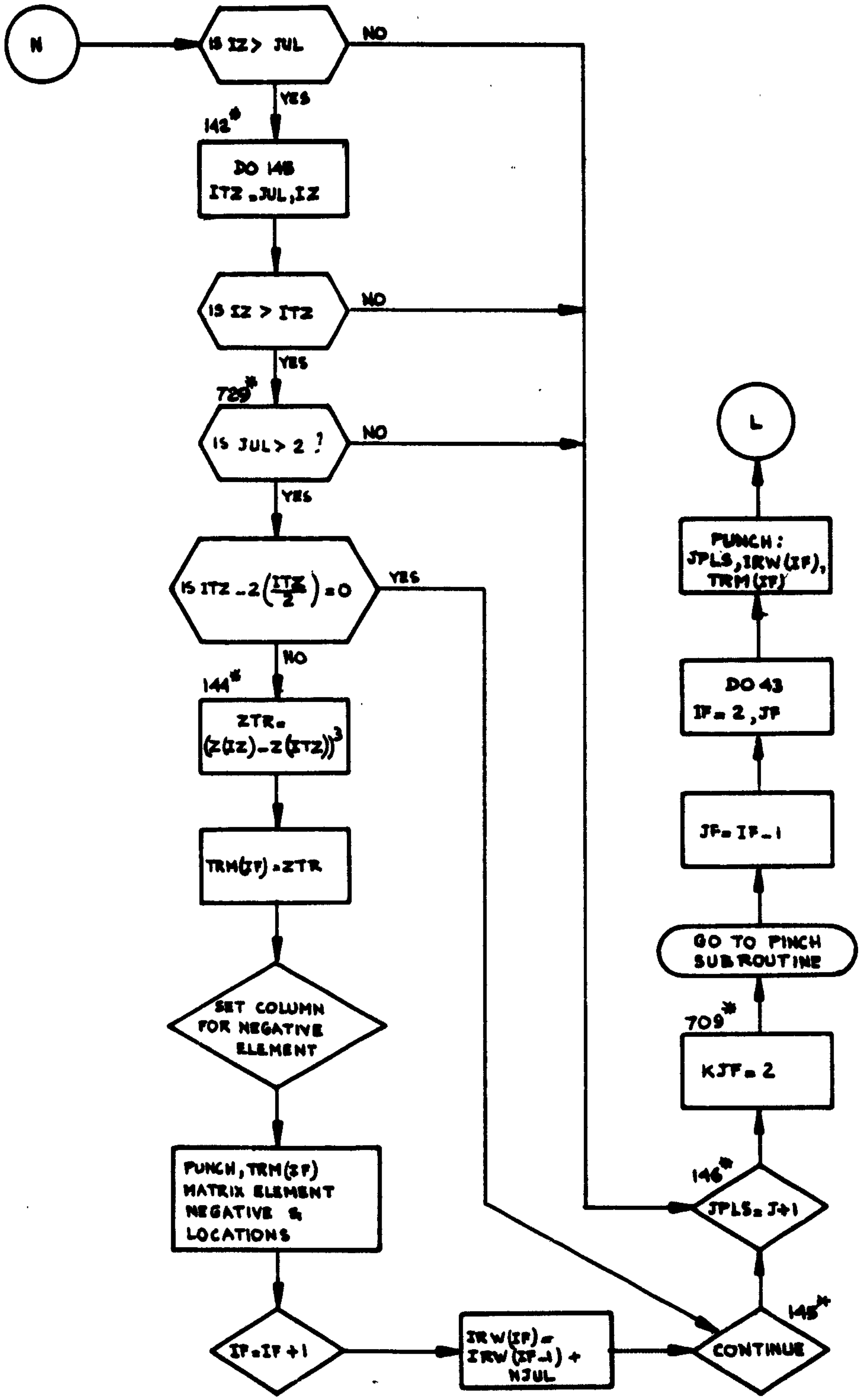

P
TRM$_{IROW}$ = X_i^{NPR}
K = 1
= 2
YES
253*
PUNCH: J, IROW
TRM (IROW)
31
IROW = IROW + 1
IS i > 3
NO
YES
32
DO 34
IT = 3, I
IS I > IT
NO
YES
133
IS IT = 2
YES
NO
Q
34*
CONTINUE
DO LOOP
STARTED AT 32*
TRM$_{IROW}$ =
$(X_i - X_{IT})^2$
K = 1
= 2
YES
254*
PUNCH: J, IROW
TRM (IROW)
255*
IROW = IROW + 1

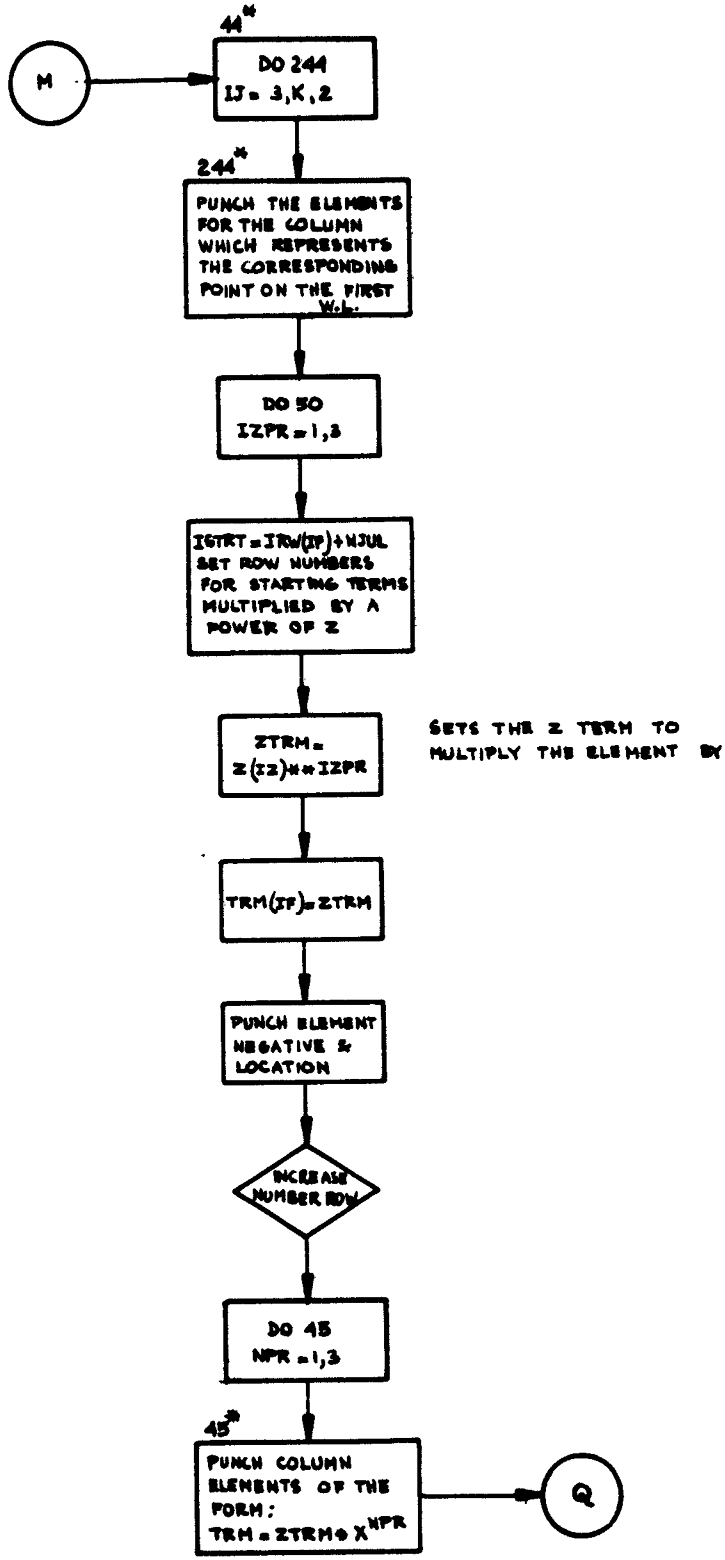


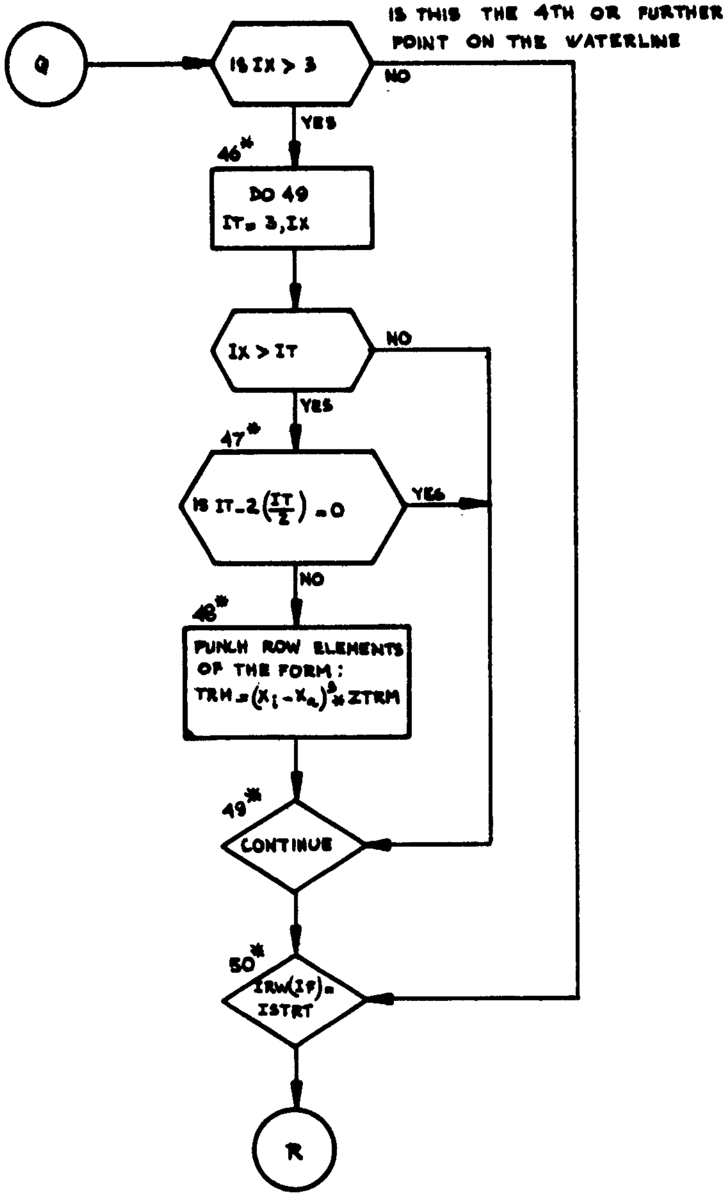
G
35*
JPLS = J+1
IROW = IROW - L
IWL1 = IROW
K = 1?
= 2
YES
259*
KFJ = 2
GO TO PINCH SUBROUTINE
705*
DO 36
IR = 3,IROW
K = 1?
= 2
YES
260*
PUNCH:
JPLS,IR,
TRM (IR)
36*
CONTINUE
DO LOOP
STARTED AT 705*
37
I = I+1
NCOMP = 1
I

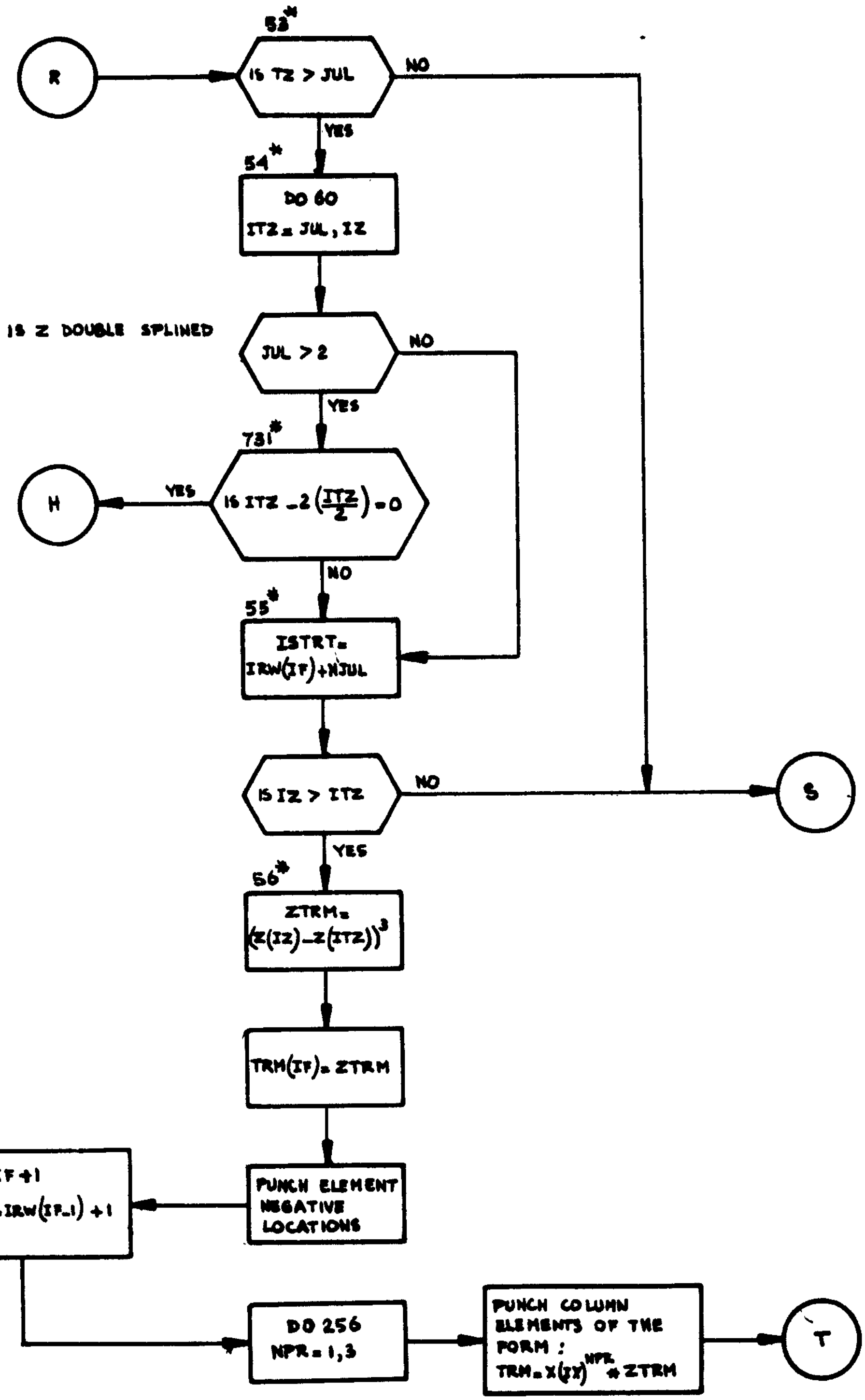

I
DO 64
IZ = 2,M
JL = J
MG =
JL + 2(NM-2)
IX = 1
DO 63
J = JL,MG,2
IF = 2
I RW(IF) =
IROW + 1
812*
IF RZ(1,I) = 20
YES
L
63*
NO
202*
IS NCOMP = 1
93*
NCOMP = 2
NO
YES
91*
GO TO PROFL 2
92*
ZY = Y (IF,IX)
NO
IS T > .0001
YES
J
258*
K
95*

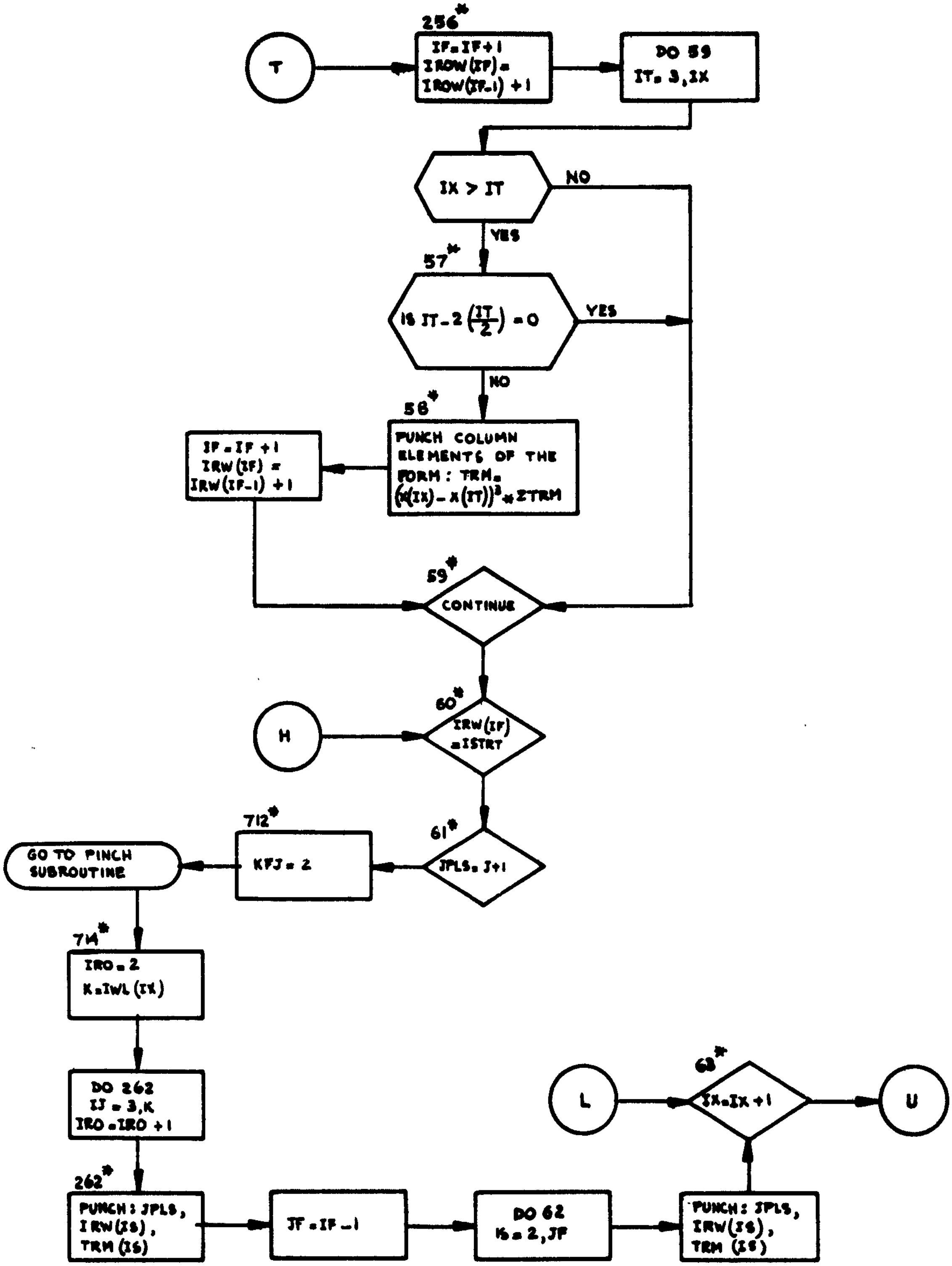

255*
MODIFY OFFSET FOR END CONDITION
ZY = Y(IZ,IX)/T
J
95*
K
KFJ = 1
GO TO PINCH SUBROUTINE
708*
J > JL
NO
M
44*
YES
41*
DO 42
IZPR = 1, 3
TRM(IF) =
Z(IZ)** IZPR
ELEMENT = Z_i^{IZPR}
PUNCH MATRIX ELEMENT & NEGATIVE & LOCATIONS
IF = IF + 1
42*
IRW(IF) = (IRW(IF-1) + NJUL
M

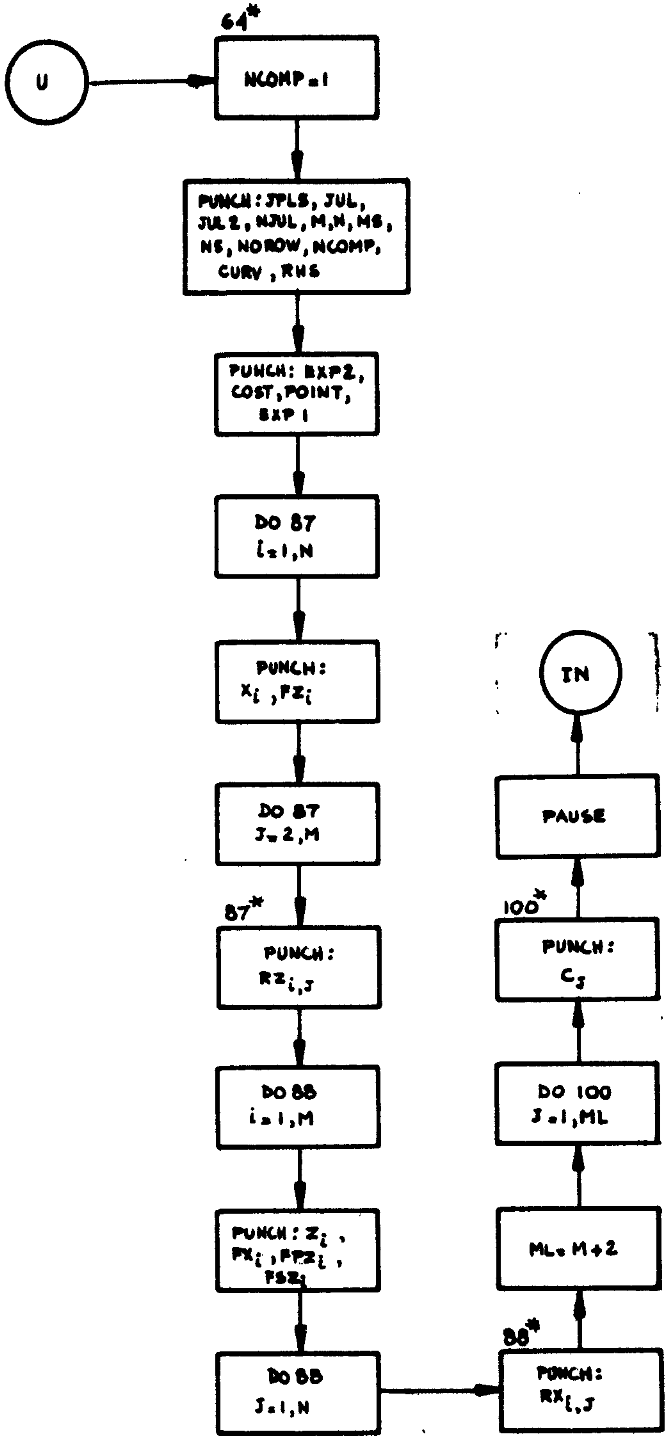

N
IS IZ > JUL
NO
YES
142*
DO 148
ITZ = JUL, IZ
IS IZ > ITZ
NO
YES
729*
IS JUL > 2 ?
NO
YES
IS ITZ - 2(ITZ/2) = 0
YES
NO
144*
ZTR = (Z(IZ) - Z(ITZ))^3
TRM(IF) = ZTR
SET COLUMN FOR NEGATIVE ELEMENT
PUNCH, TRM(IF) MATRIX ELEMENT NEGATIVE & LOCATIONS
IF = IF + 1
IRW(IF) = IRW(IF-1) + NJUL
CONTINUE
145*
146*
JPLS = J+1
709*
KJF = 2
GO TO PINCH SUBROUTINE
JF = IF - 1
DO 43
IF = 2, JF
PUNCH:
JPLS, IRW(IF), TRM(IF)
L


M
44*
DO 244
IJ = 3,K,2
244*
PUNCH THE ELEMENTS FOR THE COLUMN WHICH REPRESENTS THE CORRESPONDING POINT ON THE FIRST W.L.
DO 50
IZPR = 1,3
ISTRT = IROW(IP) + NJUL
SET ROW NUMBERS FOR STARTING TERMS MULTIPLIED BY A POWER OF Z
ZTRM = Z(IZ)** IZPR
SETS THE Z TERM TO MULTIPLY THE ELEMENT BY
TRM(IF) = ZTRM
PUNCH ELEMENT NEGATIVE & LOCATION
INCREASE NUMBER ROW
DO 45
NPR = 1,3
45*
PUNCH COLUMN ELEMENTS OF THE FORM:
TRM = ZTRM * X^NPR
Q

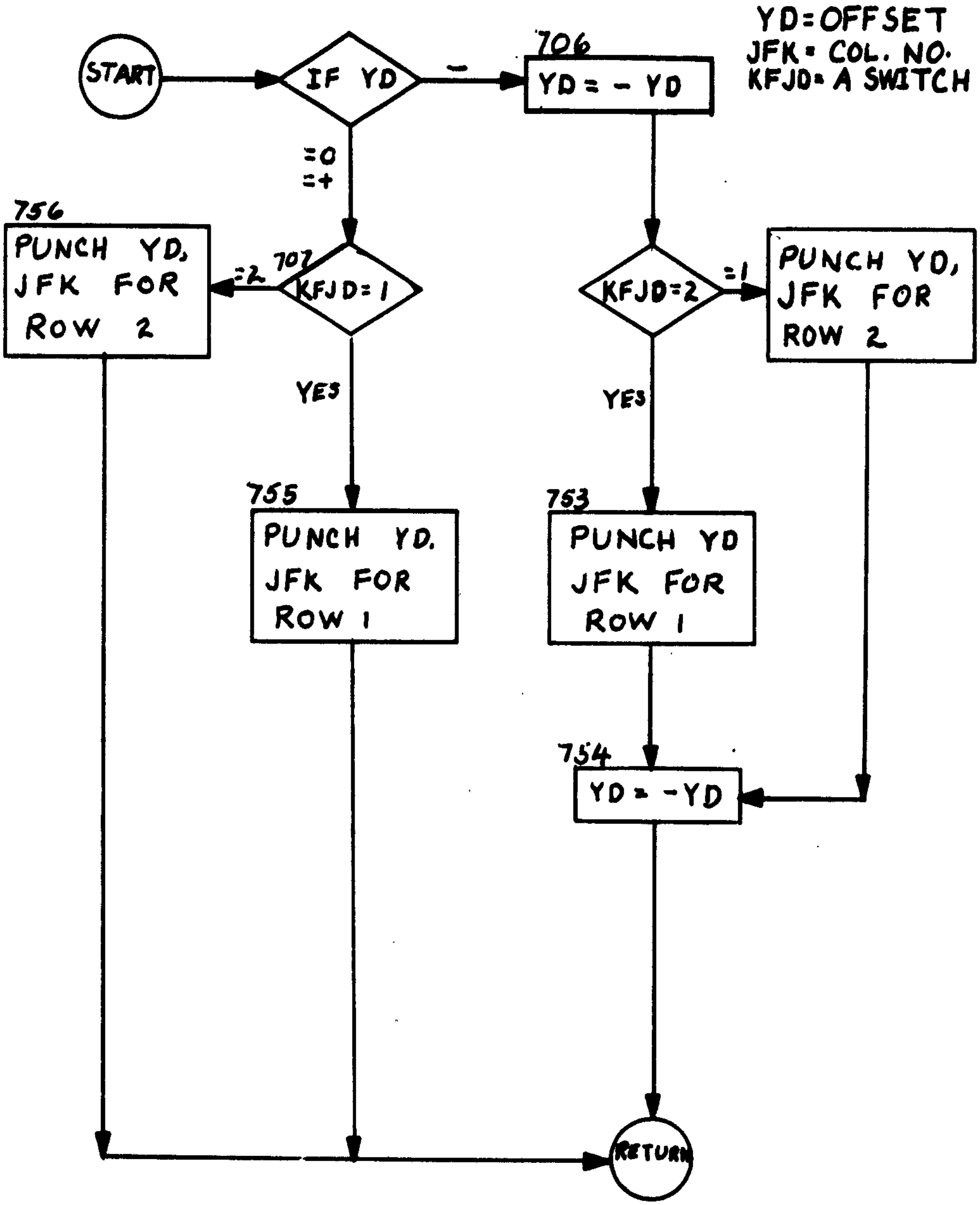

Q
IS IX > 3
IS THIS THE 4TH OR FURTHER POINT ON THE WATERLINE
NO
YES
46*
DO 49
IT = 3,IX
IX > IT
NO
YES
47*
IS IT - 2(IT/2) = 0
YES
NO
48*
PUNCH ROW ELEMENTS OF THE FORM:
TRM = (Xi - Xa)^3 * ZTRM
49*
CONTINUE
50*
IRW(IF) = ISTRT
R


R
53*
IS TZ > JUL
NO
YES
54*
DO 60
ITZ = JUL, IZ
IS Z DOUBLE SPLINED
JUL > 2
NO
YES
731*
H
YES
IS ITZ - 2 (ITZ/2) = 0
NO
55*
ISTRT = IRW(IF) + NJUL
IS IZ > ITZ
NO
S
YES
56*
ZTRM = (Z(IZ) - Z(ITZ))^3
TRM(IF) = ZTRM
PUNCH ELEMENT NEGATIVE LOCATIONS
IF = IF + 1
IRW(IF) = IRW(IF-1) + 1
DO 256
NPR = 1,3
PUNCH COLUMN ELEMENTS OF THE FORM :
TRM = X(IY)^NPR * ZTRM
T


T
256*
IF=IF+1
IROW(IF)=
IROW(IF-1) +1
DO 59
IT= 3,IX
IX > IT
NO
YES
57*
IS IT-2 (IT/2) = O
YES
NO
58*
PUNCH COLUMN
ELEMENTS OF THE
FORM: TRM=
(X(IX)-X(IT))^2*ZTRM
IF=IF+1
IRW(IF) =
IRW(IF-1) +1
59*
CONTINUE
H
60*
IRW(IF)
=ISTRT
61*
JPLS=J+1
712*
KFJ=2
GO TO PINCH
SUBROUTINE
714*
IRO=2
K=IWL(IX)
DO 262
IJ=3,K
IRO=IRO+1
262*
PUNCH: JPLS,
IRW(IS),
TRM(IS)
JF=IF-1
DO 62
IS=2,JF
PUNCH: JPLS,
IRW(IS),
TRM(IS)
68*
IX=IX+1
L
U


U
64*
NCOMP = 1
PUNCH: JPLS, JUL, JUL2, NJUL, M, N, MS, NS, NOROW, NCOMP, CURV, RHS
PUNCH: BXP2, COST, POINT, BXP1
DO 87 i=1,N
PUNCH: X_i, FZ_i
DO 87 J=2,M
87*
PUNCH: $RZ_{i,J}$
DO 88 i=1,M
PUNCH: Z_i, FX_i, FPZ_i, FSZ_i
DO 88 J=1,N
88*
PUNCH: $RX_{i,J}$
ML = M+2
DO 100 J=1,ML
100*
PUNCH: C_J
PAUSE
IN

SUBROUTINE PINCH (YD, JFK, KFJD)

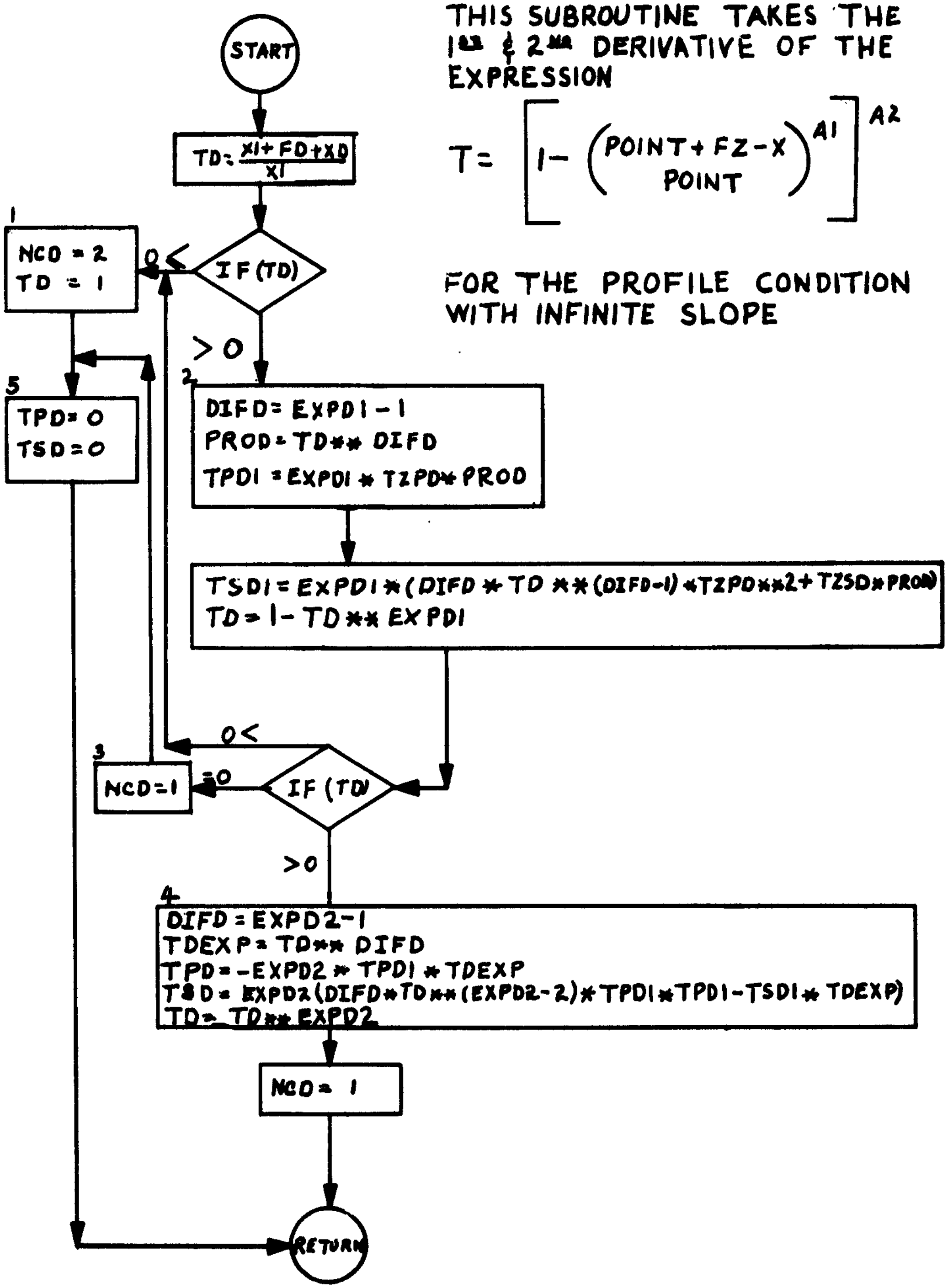


SUBROUTINE PROFL 2 XD, FD, EXPD1, TZPD, TZSD, EXPD2, X1, TD, TPD, TSD, NCD

THIS SUBROUTINE TAKES THE 1st & 2nd DERIVATIVE OF THE EXPRESSION

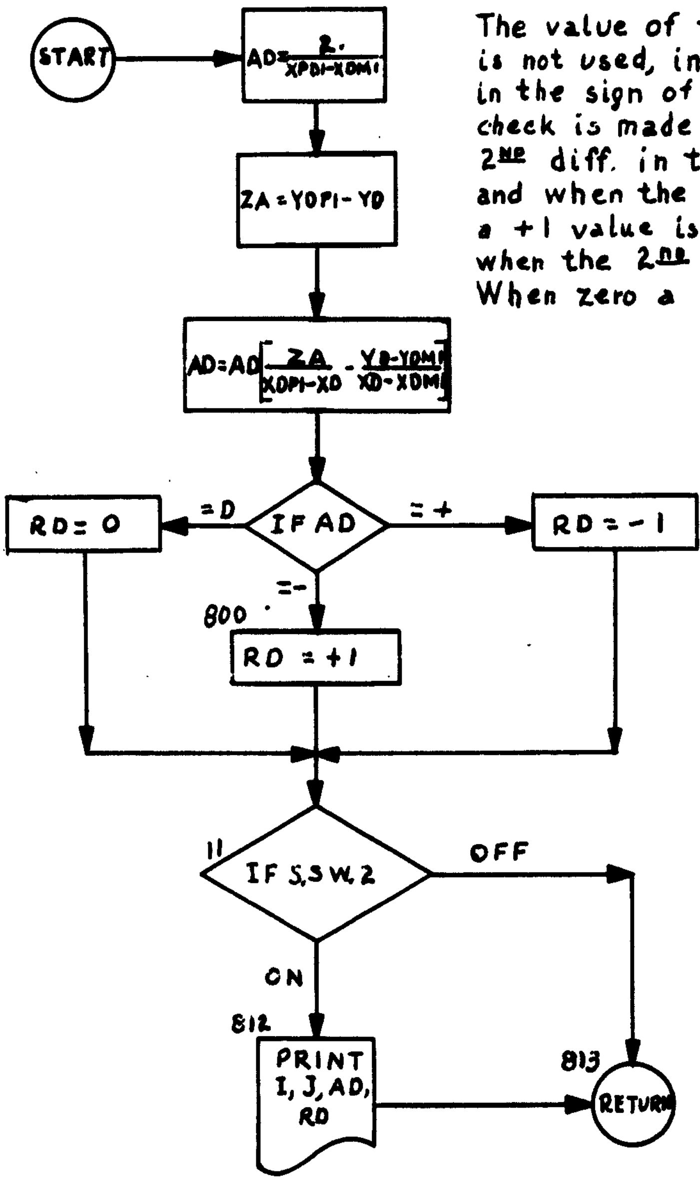
$$T = \left[1 - \left(\frac{POINT + FZ - X}{POINT} \right)^{A1} \right]^{A2}$$

FOR THE PROFILE CONDITION WITH INFINITE SLOPE



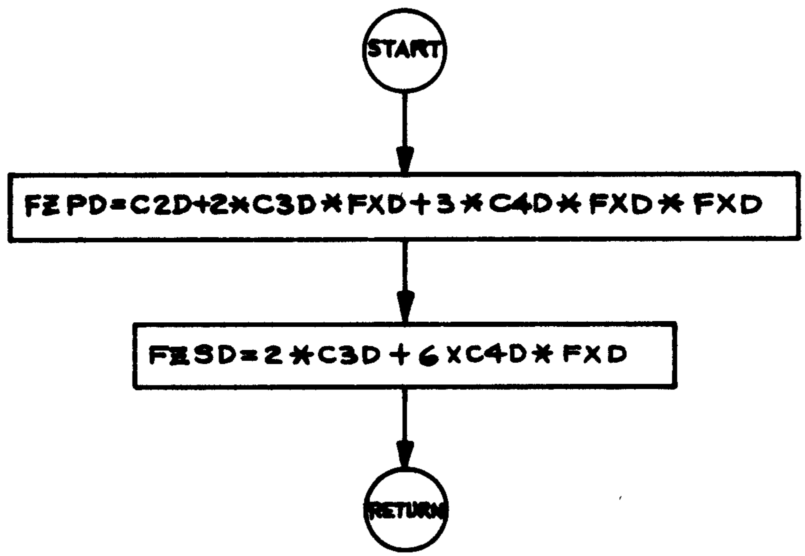
SUBROUTINE DIF2ND COMPUTES THE DIFFERENCE OF A POINT $X_{i,J}, Y_{i,J}$ BY:

$$R = \frac{2}{X_{i+1} - X_{i-1}} \left(\frac{Y_{i+1} - Y_i}{X_{i+1} - X_i} - \frac{Y_i - Y_{i-1}}{X_i - X_{i-1}} \right)$$



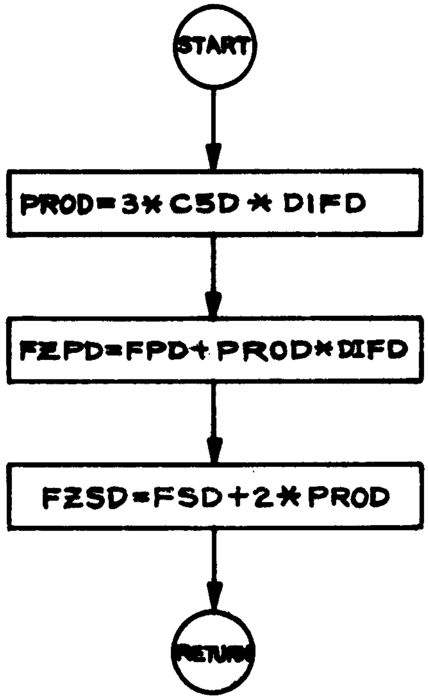
The value of the 2nd Diff. is not used, interest is only in the sign of same ∴ a check is made on the sign of 2nd diff. in this subroutine and when the sign is minus a +1 value is output and -1 when the 2nd diff. is positive. When zero a zero is output.

SUBROUTINE DERIVI(C2D, C3D, C4D, FXD, FZPD, FZSD)



THIS SUBROUTINE TAKES THE 1ST & 2ND DERIVATIVE OF THE STANDARD CUBIC

SUBROUTINE DERIV2(C5D,DIFD,FPD,FSD,F2PD,F2SD)



THIS SUBROUTINE TAKES THE 1ST & 2ND DERIVATIVE OF THE THEILHEIMER TERMS AND ADDS THEM TO THE STANDARD CUBIC 1ST & 2ND DERIVATIVES

```
C     SMOG2 PASS 1  DUAL  X OR X AND Z DBL SPL  WITH PROFILE
      DIMENSION RX(11,15),RZ(15,11),TRM(150), IRW(250),FZ(11),FX(15
)
      DIMENSION X(15),Z(11),Y(11,15),WL(15,11),IWL(15),C(13),FPZ(11
),
     CFSZ(11)
C                *   *     INPUT     *    *
  750 READ 173
      PRINT 173
      READ 150,NAA,NAB,POINT,COST
      READ 150,M,N,CURV,RHS,EXP1,EXP2
      ML=M+2
      DO 751 I=1,ML
  751 READ 154,C(I)
      NCOMP=1
      NL=N+1
      ML=M+1
      NS=N-1
      MS=M-1
      DO 2 I=1,M
      READ 154,Z(I)
      DO 2 J=1,N
      IF(SENSE SWITCH 3)802,803
  802 READ 183,X(J),IAY,IBY,ICY
      AAY=IAY
      BBY=IBY
      CCY=ICY
      Y(I,J)=AAY+(BBY+(CCY/8.))/12.
      GO TO 2
  803 READ 154,X(J),Y(I,J)
    2 CONTINUE
      IF(NAA)3,3,4
    3 READ 154,SX,SZ,TX,TZ
      GO TO 5
    4 SX=X(2)-X(1)
      SZ=Z(2)-Z(1)
      TZ=Z(1)
      TX=X(1)
    5 IF(SENSE SWITCH 2)509,508
  509 PRINT 165
C               *   *     NORMALIZING          *    *
  508 DO 26 J=1,N
   26 X(J)=(X(J)-TX)/SX
      DO 27 I=1,M
   27 Z(I)=(Z(I)-TZ)/SZ
```

```
      POINT=(POINT-TX)/SX
C       *   *   END PROFILE INTERSECTIONS   *   *
      DO 694 I=1,M
      FX(I)=C(1)+C(2)*Z(I)+C(3)*Z(I)*Z(I)+C(4)*Z(I)**3
      CALL DERIV1 (C(2),C(3),C(4),Z(I),FPZ(I),FSZ(I))
      IF(I-3)690,691,691
  691 DO 700 J=3,I
      DIF=Z(I)-Z(J-1)
      FX(I)=FX(I)+C(J+2)*DIF**3
      CALL DERIV2 (C(J+2),DIF,FPZ(I),FSZ(I),FPZ(I),FSZ(I))
      IF(I-3)690,690,700
  700 CONTINUE
  690 FPZ(I)=FPZ(I)/SZ
      FSZ(I)=FSZ(I)/(SZ*SZ)
      IF(SENSE SWITCH 2)693,694
  693 PRINT 170,I,FX(I)
      PRINT 154,FPZ(I),FSZ(I)
  694 CONTINUE
      J=1
      OLD =Z(M)
      I=MS
      IF(SENSE SWITCH 2)204,510
  204 PRINT 179
  510 FZ(J)=FX(I)
      ZZ=Z(I)
      ICOMP=1
  695 IF(ABSF(X(J)-FZ(J))-.0001) 66,66,71
   71 GO TO (65,69),ICOMP
   65 IF(X(J)-FZ(J)) 67,66,72
   66 FZ(J)=ZZ
      IF(SENSE SWITCH 2) 74,80
   74 PRINT 170,J,FZ(J)
   80 J=J+1
      OLD=ZZ
      GO TO 510
   67 ZZ=ZZ+(OLD-ZZ)/2.
      ICOMP=2
      IJUL=I+1
   68 FZ(J)=C(1)+C(2)*ZZ+C(3)*ZZ*ZZ+C(4)*ZZ**3
      IF(IJUL-3) 695,696,697
  696 FZ(J)=FZ(J)+C(5)*(ZZ-Z(2))**3
      GO TO 695
  697 DO 699 K=3,IJUL
  699 FZ(J)=FZ(J)+C(K+2)*(ZZ-Z(K-1))**3
```

```
      GO TO 695
   69 IF(X(J)-FZ(J)) 67,66,70
   70 SEG=OLD-ZZ
      OLD=ZZ
      ZZ=ZZ-SEG/2.
      GO TO 68
   72 OLD=Z(I)
      I=I-1
      IF(I-1)81,510,510
   81 DO 82 I=J,N
      FZ(I)=Z(1)
      IF(SENSE SWITCH 2)83,82
   83 PRINT 170,I,FZ(I)
   82 CONTINUE
C          *   *      WATERLINE  2ND  DIFFERENCES     *    *
      IF(SENSE SWITCH 2)6,7
    6 PRINT 157
    7 DO 15 I=1,M
      DO 13 J=1,NS
      GO TO (8,513),NCOMP
    8 ZY=0.
      IF(X(J)-FX(I))10,10,9
   10 R=20.
      IF(SENSE SWITCH 2)660,13
  660 PRINT 151,I,J
      GO TO 13
    9 ZZ=FX(I)
      NCOMP=2
      GO TO 1
  513 ZZ=X(J-1)
      ZY=Y(I,J-1)
    1 CALL DIF2ND (X(J),X(J+1),ZZ,Y(I,J),Y(I,J+1),ZY,I,J,R,A)
   13 RX(I,J)=R
      RX(I,N)=RX(I,NS)
      IF(SENSE SWITCH 2)14,15
   14 PRINT 158,I,J,A,RX(I,N)
   15 NCOMP=1
C           *   *    STATION  2ND  DIFFERENCES     *    *
      IF(SENSE SWITCH 2)16,17
   16 PRINT 159
   17 DO 25 J=1,N
      DO 23 I=2,MS
      GO TO (18,19),NCOMP
   18 IF(Z(I)-FZ(J))21,21,20
   21 R=20.
```

```
      IF(SENSE SWITCH 2)22,23
   22 PRINT 151,J,I
      GO TO 23
   20 ZZ=FZ(J)
      NCOMP=2
      GO TO 514
   19 ZZ=Z(I-1)
  514 CALL DIF2ND (Z(I),Z(I+1),ZZ,Y(I,J),Y(I+1,J),Y(I-1,J),J,I,R,A)
   23 RZ(J,I)=R
      RZ(J,M)=RZ(J,MS)
      IF(SENSE SWITCH 2)24,25
   24 PRINT 158,J,I,A,RZ(J,M)
   25 NCOMP=1
C                  *    *            ROW ID               *    *
      PUNCH 173
      NJUL=((N-3)/2)+4
      NM=N
      IF(NAB)719,718,718
  719 JUL=3
      JUL2=2
      NOROW=NJUL*((M-3)/2+4)+1
      GO TO 720
  718 JUL=2
      JUL2=1
      NOROW=(M+2)*NJUL+1
  720 PUNCH 160
      DO 28 NR=1,NOROW
   28 PUNCH 162,NR
C                 *    *         MATRIX              *    *
      PUNCH 164
C                      *    *            FIRST  WATERLINE          *
      MLG=(2*NM)
      I=1
      FUPZ=-1.
      FUSZ=0.
      DO 37 J=1,MLG,2
      IF(RX(1,I)-20.)250,251,250
  251 K=2
      GO TO 252
  250 K=1
  200 GO TO (201,702),NCOMP
  201 CALL PROFL2(X(I),FX(1),EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS,
NCOMP)
      IF(T-.0001)703,703,257
  257 ZY=Y(1,I)/T
```

```
      GO TO 90
  703 NCOMP=2
  702 ZY=Y(1,I)
   90 KFJ=1
      CALL PINCH (ZY,J,KFJ)
  252 IROW=3
      DO 31 NPR=1,3
      TRM(IROW)=X(I)**NPR
      GO TO (253,31),K
  253 PUNCH 167,J,IROW,TRM(IROW)
   31 IROW=IROW+1
      IF(I-3)35,35,32
   32 DO 34 IT=3,I
      IF(I-IT)34,34,133
  133 IF(IT-2*(IT/2))33,34,33
   33 TRM(IROW)=(X(I)-X(IT))**3
      GO TO (254,255),K
  254 PUNCH 167,J,IROW,TRM(IROW)
  255 IROW=IROW+1
   34 CONTINUE
   35 JPLS=J+1
      IROW=IROW-1
      IWL(I)=IROW
      GO TO (259,705),K
  259 KFJ=2
      CALL PINCH (ZY,JPLS,KFJ)
  705 DO 36 IR=3,IROW
      WL(I,IR)=TRM(IR)
      GO TO (260,36),K
  260 PUNCH 169,JPLS,IR,TRM(IR)
   36 CONTINUE
   37 I=I+1
      NCOMP=1
C         *    *    REMAINING WL    *    *
      DO 64 IZ=2,M
      JL=J
      MG=JL+(2*NM)-2
      IX=1
      DO 63 J=JL,MG,2
      IF=2
      IRW(IF)=IROW+1
      IF(RX(IZ,IX)-20.)202,63,202
  202 GO TO (91,92),NCOMP
   91 CALL PROFL2(X(IX),FX(IZ),EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS,NCO
MP)
```

```
      IF(T-.0001)93,93,258
  258 ZY=Y(IZ,IX)/T
      GO TO 95
   93 NCOMP=2
   92 ZY=Y(IZ,IX)
   95 KFJ=1
      CALL PINCH (ZY,J,KFJ)
  708 IF(J-JL)41,41,44
   41 DO 42 IZPR=1,3
      TRM(IF)=Z(IZ)**IZPR
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
   42 IRW(IF)=IRW(IF-1)+NJUL
      IF(IZ-JUL)146,146,142
  142 DO 145 ITZ=JUL,IZ
      IF(IZ-ITZ)146,146,729
  729 GO TO (144,144,730),JUL
  730 IF(ITZ-2*(ITZ/2))144,145,144
  144 ZTR=(Z(IZ)-Z(ITZ))**3
      TRM(IF)=ZTR
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+NJUL
  145 CONTINUE
  146 JPLS=J+1
  709 KFJ=2
      CALL PINCH (ZY,JPLS,KFJ)
  711 JF=IF-1
      DO 43 IF=2,JF
   43 PUNCH 169,JPLS,IRW(IF),TRM(IF)
      GO TO 63
C     * * Z , Z SQR , Z CUBE ROWS * *
   44 IRO =2
      K=IWL(IX)
      DO 244 IJ=3,K
      IRO =IRO +1
  244 PUNCH 167,J,IRO,WL(IX,IJ)
      DO 50 IZPR=1,3
      ISTRT=IRW(IF)+NJUL
      ZTRM=Z(IZ)**IZPR
      TRM(IF)=ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+1
```

```
      DO 45 NPR=1,3
      TRM(IF)=(X(IX)**NPR)*ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
   45 IRW(IF)=IRW(IF-1)+1
      IF(IX-3)50,50,46
   46 DO 49 IT=3,IX
      IF(IX-IT)49,49,47
   47 IF(IT-2*(IT/2))48,49,48
   48 TRM(IF)=((X(IX)-X(IT))**3)*ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+1
   49 CONTINUE
   50 IRW(IF)=ISTRT
C        *    *      THEILHEIMER  ROWS          *    *
   53 IF(IZ-JUL)61,61,54
   54 DO 60 ITZ=JUL,IZ
      GO TO (55,55,731),JUL
  731 IF(ITZ-2*(ITZ/2))55,60,55
   55 ISTRT=IRW(IF)+NJUL
      IF(IZ-ITZ)61,61,56
   56 ZTRM=(Z(IZ)-Z(ITZ))**3
      TRM(IF)=ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+1
      DO 256 NPR=1,3
      TRM(IF)=(X(IX)**NPR)*ZTRM
      PUNCH 167,J,IRW(IF),TRM(IF)
      IF=IF+1
  256 IRW(IF)=IRW(IF-1)+1
      DO 59 IT=3,IX
      IF(IX-IT)59,59,57
   57 IF(IT-2*(IT/2))58,59,58
   58 TRM(IF)=((X(IX)-X(IT))**3)*ZTRM
      PUNCH 167 ,J,IRW(IF),TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+1
   59 CONTINUE
   60 IRW(IF)=ISTRT
   61 JPLS=J+1
  712 KFJ=2
      CALL PINCH (ZY,JPLS,KFJ)
```

```
714 IRO =2
    K=IWL(IX)
    DO 262 IJ=3,K
    IRO =IRO +1
262 PUNCH 169,JPLS,IRO ,WL(IX,IJ)
    JF=IF-1
    DO 62 IS=2,JF
 62 PUNCH 169,JPLS,IRW(IS),TRM(IS)
 63 IX=IX+1
 64 NCOMP=1
    PUNCH 180,JPLS,JUL,JUL2,NJUL,M,N,MS,NS,NOROW,NCOMP,CURV,RHS
    PUNCH 181,EXP2,COST,POINT,EXP1
    DO 87 I=1,N
    PUNCH 181,X(I),FZ(I)
    DO 87 J=2,M
87  PUNCH 181,RZ(I,J)
    DO 88 I=1,M
    PUNCH 181,Z(I),FX(I),FPZ(I),FSZ(I)
    DO 88 J=1,N
 88 PUNCH 181,RX(I,J)
    ML=M+2
    DO 100 J=1,ML
100 PUNCH 181,C(J)
    PRINT 178
    PAUSE 751
    GO TO 750
150 FORMAT(2I5,4F11.6)
151 FORMAT(2I5,17H   OUT OF SURFACE)
154 FORMAT(4F10.4)
157 FORMAT(//10X,17H2ND DIFF - WLINES//
   C30H   WL   STA NORM VALUE    SIGN/)
158 FORMAT(2I5,F12.6,F8.2)
159 FORMAT(//10X,17H2ND DIFF-STATIONS//
   C30H   STA   WL NORM VALUE    SIGN/)
160 FORMAT(6HROW ID/12X,6H OFSET)
162 FORMAT(12X,2H R,I4)
164 FORMAT(6HMATRIX)
165 FORMAT(//6X,25HEND PROFILE INTERSECTIONS//15X,6HWLINES/)
167 FORMAT(6X,2H C,I4,2H R,I4,1H-,F11.5)
169 FORMAT(6X,2H C,I4,2H R,I4,F12.5)
170 FORMAT(5X,I5,F12.6)
171 FORMAT(6X,2H C,I4,6H OFSET,F12.5)
172 FORMAT(2I5)
173 FORMAT(1H*,5X,47H
)
```

```
178 FORMAT(28HENTER NEW DATA - PRESS START)
179 FORMAT (/14X,8HSTATIONS/)
180 FORMAT (10I5,2F15.9)
181 FORMAT (4F15.9)
183 FORMAT(F10.5,3I5)
    END
```

```
      SUBROUTINE PINCH (YD,JFK,KFJD)
      IF(YD)706,707,707
  706 YD=-YD
      GO TO (752,753),KFJD
  752 PUNCH 168,JFK,YD,JFK,JFK
      GO TO 754
  753 PUNCH 166,JFK,YD,JFK,JFK
  754 YD=-YD
      RETURN
  707 GO TO (755,756),KFJD
  755 PUNCH 166,JFK,YD,JFK,JFK
      RETURN
  756 PUNCH 168,JFK,YD,JFK,JFK
      RETURN
  166 FORMAT(6X,2H C,I4,6H OFSET,F12.5/6X,2H C,I4,6H R    1,8H      1.0/
     C6X,2H C,I4,6H R    2,8H-     1.0)
  168 FORMAT(6X,2H C,I4,7H OFSET-,F11.5/6X,2H C,I4,6H R    1,8H      1.0/
     C6X,2H C,I4,6H R    2,8H      1.0)
      END

      SUBROUTINE PROFL2(XD,FD,EXPD1,TZPD,TZSD,EXPD2,X1,TD,TPD,TSD,NCD)
      TD=(X1+FD-XD)/X1
      IF(TD-.00001)1,1,2
    2 DIFD=EXPD1-1.
      PROD=TD**DIFD
      TPD1=EXPD1*TZPD*PROD
      TSD1=EXPD1*(DIFD*TD**(DIFD-1.)*TZPD**2+TZSD*PROD)
      TD=1.-TD**EXPD1
      IF(TD-.00001)7,7,4
    4 DIFD=EXPD2-1.
      TDEXP=TD**DIFD
      TPD=-EXPD2*TPD1*TDEXP
      TSD=EXPD2*(DIFD*TD**(EXPD2-2.)*TPD1*TPD1-TSD1*TDEXP)
      TD=TD**EXPD2
      IF(TD-.00001)7,7,6
    7 TD=0.
      GO TO 3
    6 NCD=1
      RETURN
    1 NCD=2
      TD=1.
    5 TPD=0.
      TSD=0.
      RETURN
    3 NCD=1
      GO TO 5
      END
```

```
      SUBROUTINE DIF2ND(XD,XDP1,XDM1,YD,YDP1,YDM1,ID,JD,RD,AD)
      AD=2./(XDP1-XDM1)
      ZA=YDP1-YD
      AD=AD*(ZA/(XDP1-XD)-(YD-YDM1)/(XD-XDM1))
      IF(AD)800,809,808
  808 RD=-1.
      GO TO 11
  809 RD=0.
      GO TO 11
  800 RD=1.
   11 IF(SENSE SWITCH 2)812,813
  812 PRINT 158,ID,JD,AD,RD
  158 FORMAT(2I5,F12.6,F8.2)
  813 RETURN
      END


      SUBROUTINE DERIV1 (C2D,C3D,C4D,FXD,FZPD,FZSD)
      FZPD=C2D+2.*C3D*FXD+3.*C4D*FXD*FXD
      FZSD=2.*C3D+6.*C4D*FXD
      RETURN
      END


      SUBROUTINE DERIV2 (C5D,DIFD,FPD,FSD,FZPD,FZSD)
      PROD=3.*C5D*DIFD
      FZPD=FPD+PROD*DIFD
      FZSD=FSD+2.*PROD
      RETURN
      END
```

OPERATING INSTRUCTIONS

Pass 2 of the program accepts the data punched in Pass 1. It produces the curvature constraints of the matrix and then the right-hand side , which for the dual, consists of the costs on the variables.

Fortran Input Symbols

JPLS - The number of the last column produced in Pass 1

JUL - Program indicator

JUL = 3 if the surface is double splined in the z direction

JUL = 2 if the surface is single splined in the z direction

This value is equal to the subscript of the first z value of a point of discontinuity

JUL2 - Program indicator

JUL2 = 2 if the z direction is double splined

JUL2 = 1 if the z direction is single splined

This value is equal to the number of waterline intervals between points of discontinuity

NJUL - This value $(\frac{N-3}{2} + 4)$ is equal to the maximum number of rows necessary to store the x terms, each of which is multiplied by a specific z term

M - Total number of waterlines in the surface

MS - M-1

N - Total number of stations in the surface

NS - N-1

NOROW - Total number of rows in the matrix

For z double spline $NOROW = \left[\left(\frac{M-3}{2}\right) + 4\ NJUL\right] + 1$

For z single spline $NOROW = \left[(M+2)NJUL\right]$

NCOMP - Program indicator

COST - The penalty assigned to λ . This will appear in Row 1 of the right-hand side. The value is normally made 20.

EXP1 - The value of p in the profile function usually made .3333 .

EXP2 - The value of r in the profile function, usually 3.0 .

POINT - The value of D in the profile function. This number is the length of fraction of the station spacing over which the profile function is effective in terms of the full scale surface (the value has been translated and normalized exactly like the x values are).

C(I) - Coefficients of the Theilheimer equation describing the profile. There must be (M+2) coefficients. Straight lines or other curves may be used by making all except the desired coefficients zero.

RHS - Right-hand side - In the dual this is really the cost on the equation coefficients. The ideal value for RHS would be zero; however, this makes the solution susceptible to cycling so that a value of .0001 is usually used.

CURV - Value of the requirement vector for the curvature constraints. This value actually goes in the cost row in the dual and is always made zero.

X(J) - Distance along the x axis from the origin of the surface to Station j .

Z(I) - Distance along the z axis from the origin of the surface to Waterline i .

FX(I) - The x coordinate of the intersection of Waterline i with the profile

FZ(J) - The z coordinate of the intersection of Station j with the profile

FPZ(I) - The first derivative, with respect to z , of the profile equation at Waterline i .

FSZ(I) - The second derivative with respect to z of the profile equation at Waterline i .

RZ(I,J) - The sign of the second difference of the Station Curve j at Waterline i . This will have the opposite sign from the actual second difference.

RX(I,J) - The sign of the second difference of Waterline Curve i at Station j . This will be the opposite of the actual sign.

Input Data

The following input formats are listed as a matter of record in case modifications of the data cards are to be made. Normally this data will be obtained from executing Pass 1 in a form which can be entered directly into Pass 2.

	Contents of Card									Card No.
Format	I5	I5	I5	I5	I5	I5	I5	I5		
Variable	JPLS	JUL	JUL2	NJUL	M	N	MS	NS		1
Format (continuation of Card 1)			I5	I5	F15.5	F15.5				
Variable			MOROW	NCOMP	CURV	RHS				
Format	F15.9	F15.9	F15.9	F15.9						2
Variable	EXP2	COST	POINT	EXP1						
Format	F15.9	F15.9								Next N Sets
Variable	X(I)	FZ(I)							1st card/set	
Format	F15.9									
Variable	RZ(I,J)								M Cards/set	
Format	F15.9	F15.9	F15.9	F15.9						Next M Sets
Variable	Z(I)	FX(I)	FPZ(I)	FSZ(I)					1st card/set	
Format	F15.9									
Variable	RX(I,J)								N Cards/set	
Format	F15.9									Next M+2 Cards
Variable	C(J)									

Output

The curvature constraint elements are produced first. These are punched with the formats listed in Fig. B- , as are the following cards. The elements are punched in column order with the rows ordered within the columns. Next, a card punched "FIRST B" is produced. Afterwards, the right-hand side is punched in row order. Finally, the last card in the deck is punched "EOF."

After the input data for Pass 2 is removed from the Pass 1 output, the Pass 2 output is placed directly behind the Pass 1 output. The deck is now complete and ready for input to LP-90 .

Sense Switches

No sense switches are used.

** SAMPLE INPUT **

```
  50     3     2     5     5     5     4     4    26     1     0.00000000
    .00100000
    .33333333    20.00000000      .07843137     3.00000000
   0.00000000     2.60400380
  20.00000000
  20.00000000
   1.00000000
   1.00000000
   1.00000000     .90136719
   1.00000000
  -1.00000000
  -1.00000000
  -1.00000000
   2.00000000    0.00000000
  -1.00000000
  -1.00000000
  -1.00000000
  -1.00000000
   3.00000000    0.00000000
  -1.00000000
  -1.00000000
  -1.00000000
  -1.00000000
   4.00000000    0.00000000
  -1.00000000
  -1.00000000
  -1.00000000
  -1.00000000
   0.00000000    1.52935000    -.58730000     0.00000000
  20.00000000
  20.00000000
   1.00000000
  -1.00000000
  -1.00000000
   1.00000000     .94205000    -.58730000     0.00000000
  20.00000000
   1.00000000
  -1.00000000
  -1.00000000
  -1.00000000
   2.00000000     .35475000    -.58730000     0.00000000
  20.00000000
   1.00000000
  -1.00000000
  -1.00000000
  -1.00000000
   3.00000000   -.23255000    -.58730000     0.00000000
```

1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
4.00000000	-.81985000	-.58730000	0.00000000
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
1.52935000			
-.58730000			
0.00000000			
0.00000000			
0.00000000			
0.00000000			
0.00000000			

```
C  52 OFSET      0.00000
C  52 R    4-    1.00000
C  52 R    5-    6.00000
C  53 OFSET      0.00000
C  53 R    4     1.00000
C  53 R    5    12.00000
C  53 R    6     6.00000
C  55 OFSET      0.00000
C  55 R    4     1.00000
C  55 R    5     6.00000
C  55 R    9     1.00000
C  55 R   10     6.00000
C  55 R   14     1.00000
C  55 R   15     6.00000
C  55 R   19     1.00000
C  55 R   20     6.00000
C  56 OFSET      0.00000
C  56 R    4     1.00000
C  56 R    5    12.00000
C  56 R    6     6.00000
C  56 R    9     1.00000
C  56 R   10    12.00000
C  56 R   11     6.00000
C  56 R   14     1.00000
C  56 R   15    12.00000
C  56 R   16     6.00000
C  56 R   19     1.00000
C  56 R   20    12.00000
C  56 R   21     6.00000
C  58 OFSET      0.00000
C  58 R    4     1.00000
C  58 R    5     6.00000
C  58 R    9     2.00000
C  58 R   10    12.00000
C  58 R   14     4.00000
C  58 R   15    24.00000
C  58 R   19     8.00000
C  58 R   20    48.00000
```

C	59	OFSET		0.00000
C	59	R	4	1.00000
C	59	R	5	12.00000
C	59	R	6	6.00000
C	59	R	9	2.00000
C	59	R	10	24.00000
C	59	R	11	12.00000
C	59	R	14	4.00000
C	59	R	15	48.00000
C	59	R	16	24.00000
C	59	R	19	8.00000
C	59	R	20	96.00000
C	59	R	21	48.00000
C	60	OFSET		0.00000
C	60	R	4-	1.00000
C	60	R	9-	3.00000
C	60	R	14-	9.00000
C	60	R	19-	27.00000
C	60	R	24-	1.00000
C	61	OFSET		0.00000
C	61	R	4	1.00000
C	61	R	5	6.00000
C	61	R	9	3.00000
C	61	R	10	18.00000
C	61	R	14	9.00000
C	61	R	15	54.00000
C	61	R	19	27.00000
C	61	R	20	162.00000
C	61	R	24	1.00000
C	61	R	25	6.00000
C	62	OFSET		0.00000
C	62	R	4	1.00000
C	62	R	5	12.00000
C	62	R	6	6.00000
C	62	R	9	3.00000
C	62	R	10	36.00000
C	62	R	11	18.00000
C	62	R	14	9.00000
C	62	R	15	108.00000
C	62	R	16	54.00000
C	62	R	19	27.00000

```
C   62 R    20    324.00000
C   62 R    21    162.00000
C   62 R    24      1.00000
C   62 R    25     12.00000
C   62 R    26      6.00000
C   63 OFSET        0.00000
C   63 R     4      1.00000
C   63 R     9      4.00000
C   63 R    14     16.00000
C   63 R    19     64.00000
C   63 R    24      8.00000
C   64 OFSET        0.00000
C   64 R     4      1.00000
C   64 R     5      6.00000
C   64 R     9      4.00000
C   64 R    10     24.00000
C   64 R    14     16.00000
C   64 R    15     96.00000
C   64 R    19     64.00000
C   64 R    20    384.00000
C   64 R    24      8.00000
C   64 R    25     48.00000
C   65 OFSET        0.00000
C   65 R     4      1.00000
C   65 R     5     12.00000
C   65 R     6      6.00000
C   65 R     9      4.00000
C   65 R    10     48.00000
C   65 R    11     24.00000
C   65 R    14     16.00000
C   65 R    15    192.00000
C   65 R    16     96.00000
C   65 R    19     64.00000
C   65 R    20    768.00000
C   65 R    21    384.00000
C   65 R    24      8.00000
C   65 R    25     96.00000
C   65 R    26     48.00000
C   67 OFSET        0.00000
C   67 R    12-     1.00000
C   67 R    17-    12.00000
```

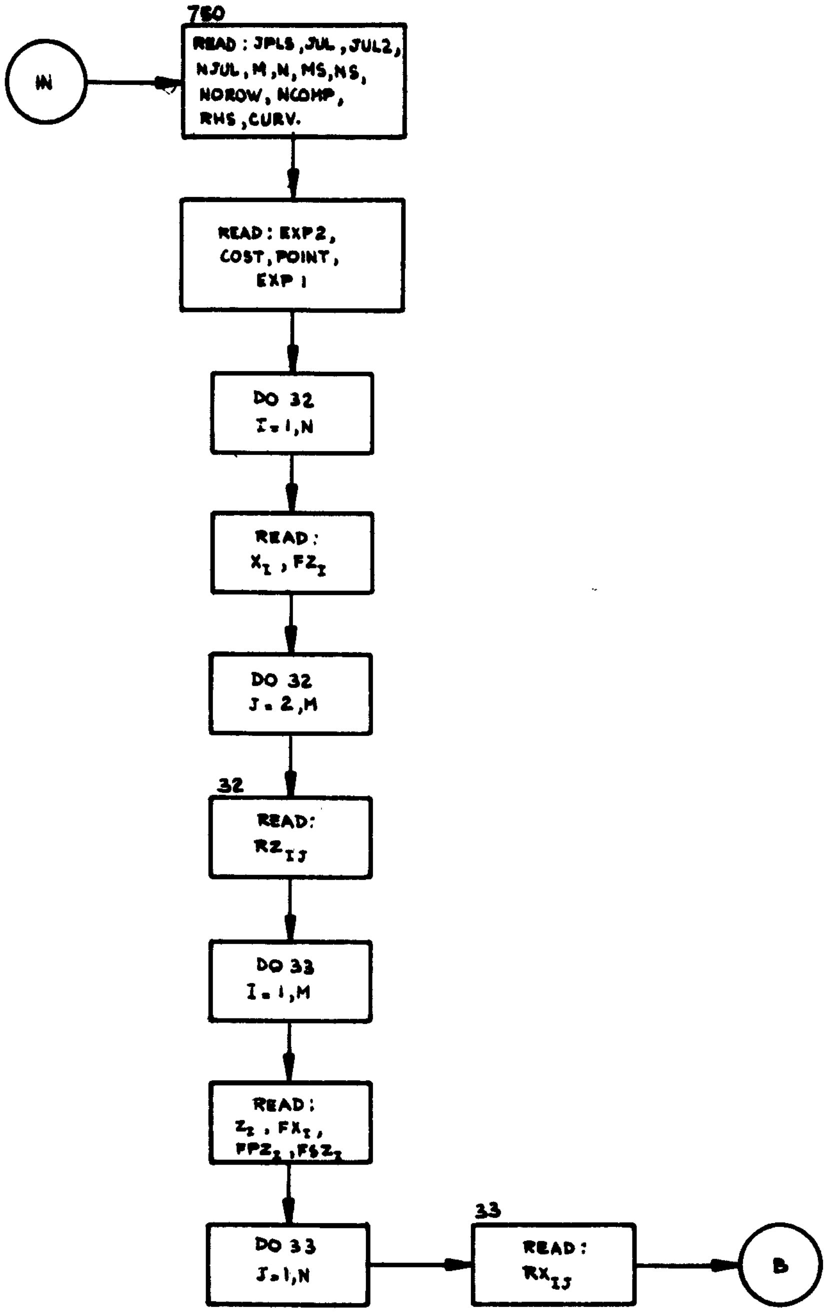
```
C   67 R   22-     6.00000
C   68 OFSET       0.00000
C   68 R    12     1.00000
C   68 R    13     1.00000
C   68 R    14     1.00000
C   68 R    15     1.00000
C   68 R    17     6.00000
C   68 R    18     6.00000
C   68 R    19     6.00000
C   68 R    20     6.00000
C   69 OFSET       0.00000
C   69 R    12     1.00000
C   69 R    13     1.00000
C   69 R    14     1.00000
C   69 R    15     1.00000
C   69 R    17    12.00000
C   69 R    18    12.00000
C   69 R    19    12.00000
C   69 R    20    12.00000
C   69 R    22     6.00000
C   69 R    23     6.00000
C   69 R    24     6.00000
C   69 R    25     6.00000
C   70 OFSET       0.00000
C   70 R    12     1.00000
C   70 R    13     2.00000
C   70 R    14     4.00000
C   70 R    15     8.00000
C   70 R    17     6.00000
C   70 R    18    12.00000
C   70 R    19    24.00000
C   70 R    20    48.00000
C   71 OFSET       0.00000
C   71 R    12     1.00000
C   71 R    13     2.00000
C   71 R    14     4.00000
C   71 R    15     8.00000
C   71 R    17    12.00000
C   71 R    18    24.00000
C   71 R    19    48.00000
C   71 R    20    96.00000
```

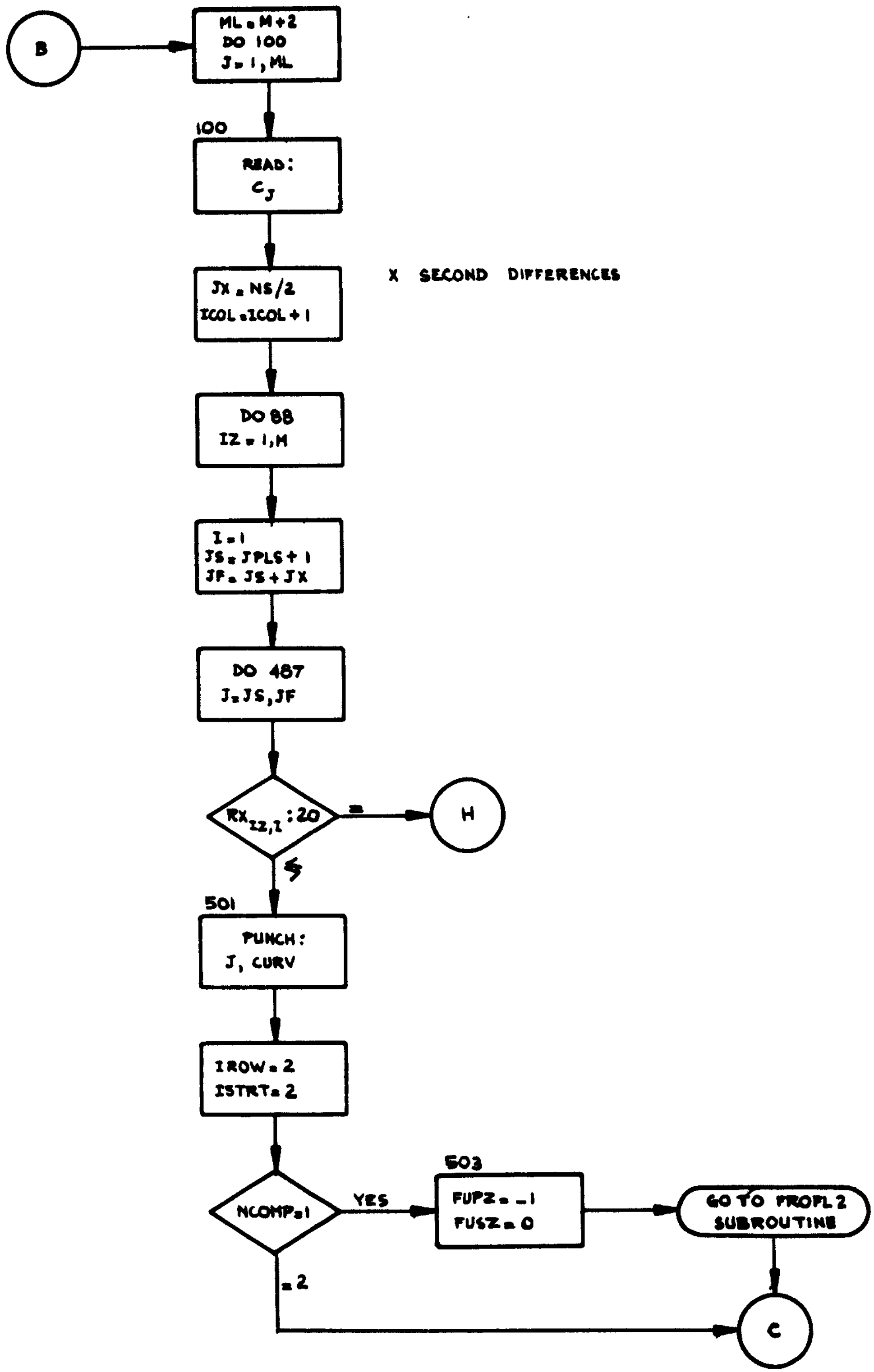
```
C  71 R   22      6.00000
C  71 R   23     12.00000
C  71 R   24     24.00000
C  71 R   25     48.00000
C  72 OFSET       0.00000
C  72 R   12      1.00000
C  72 R   13      3.00000
C  72 R   14      9.00000
C  72 R   15     27.00000
C  72 R   16      1.00000
C  72 R   17      6.00000
C  72 R   18     18.00000
C  72 R   19     54.00000
C  72 R   20    162.00000
C  72 R   21      6.00000
C  73 OFSET       0.00000
C  73 R   12      1.00000
C  73 R   13      3.00000
C  73 R   14      9.00000
C  73 R   15     27.00000
C  73 R   16      1.00000
C  73 R   17     12.00000
C  73 R   18     36.00000
C  73 R   19    108.00000
C  73 R   20    324.00000
C  73 R   21     12.00000
C  73 R   22      6.00000
C  73 R   23     18.00000
C  73 R   24     54.00000
C  73 R   25    162.00000
C  73 R   26      6.00000
C  74 OFSET       0.00000
C  74 R   12      1.00000
C  74 R   13      4.00000
C  74 R   14     16.00000
C  74 R   15     64.00000
C  74 R   16      8.00000
C  74 R   17      6.00000
C  74 R   18     24.00000
C  74 R   19     96.00000
C  74 R   20    384.00000
```

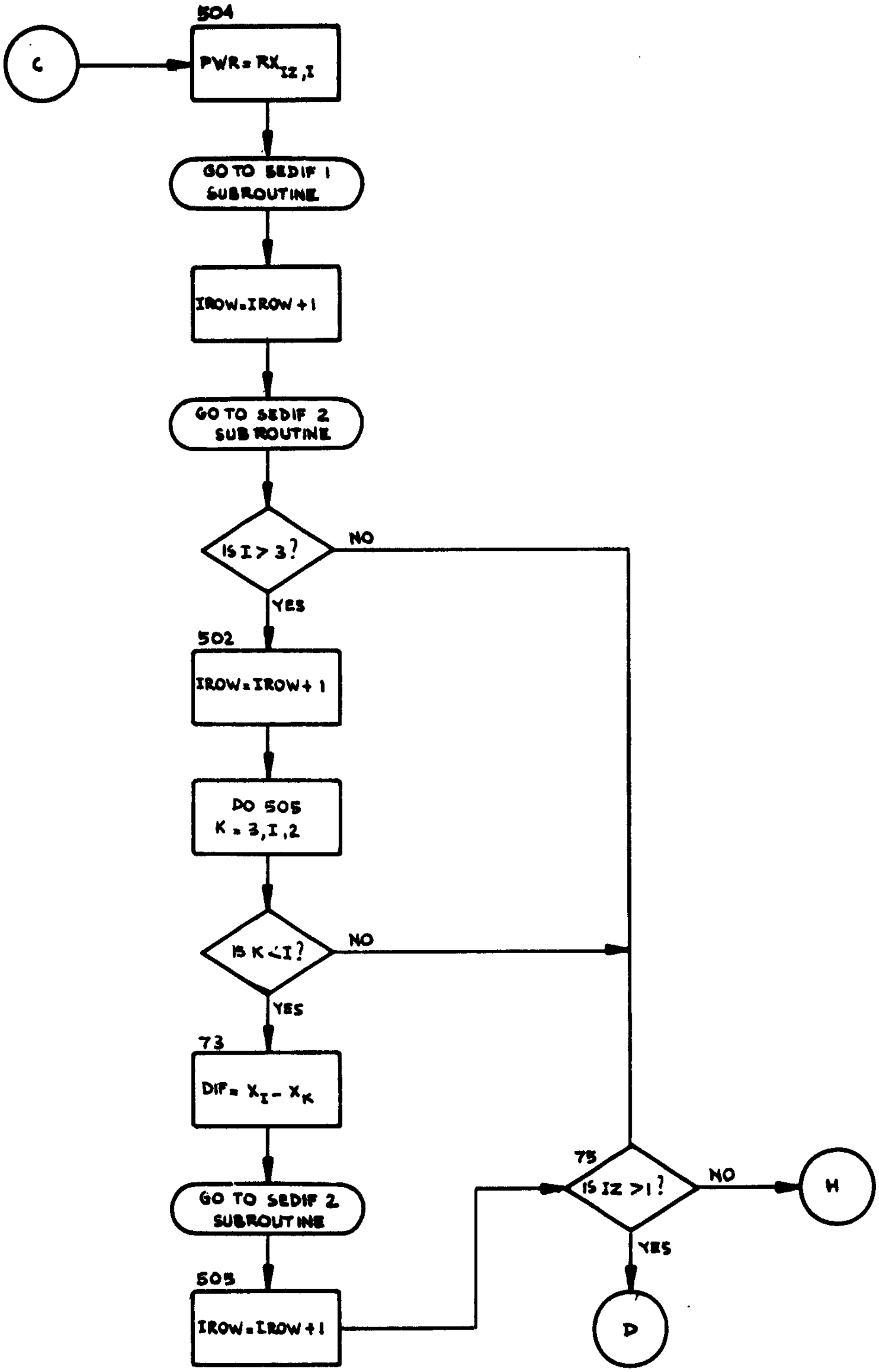
1.0.0-2

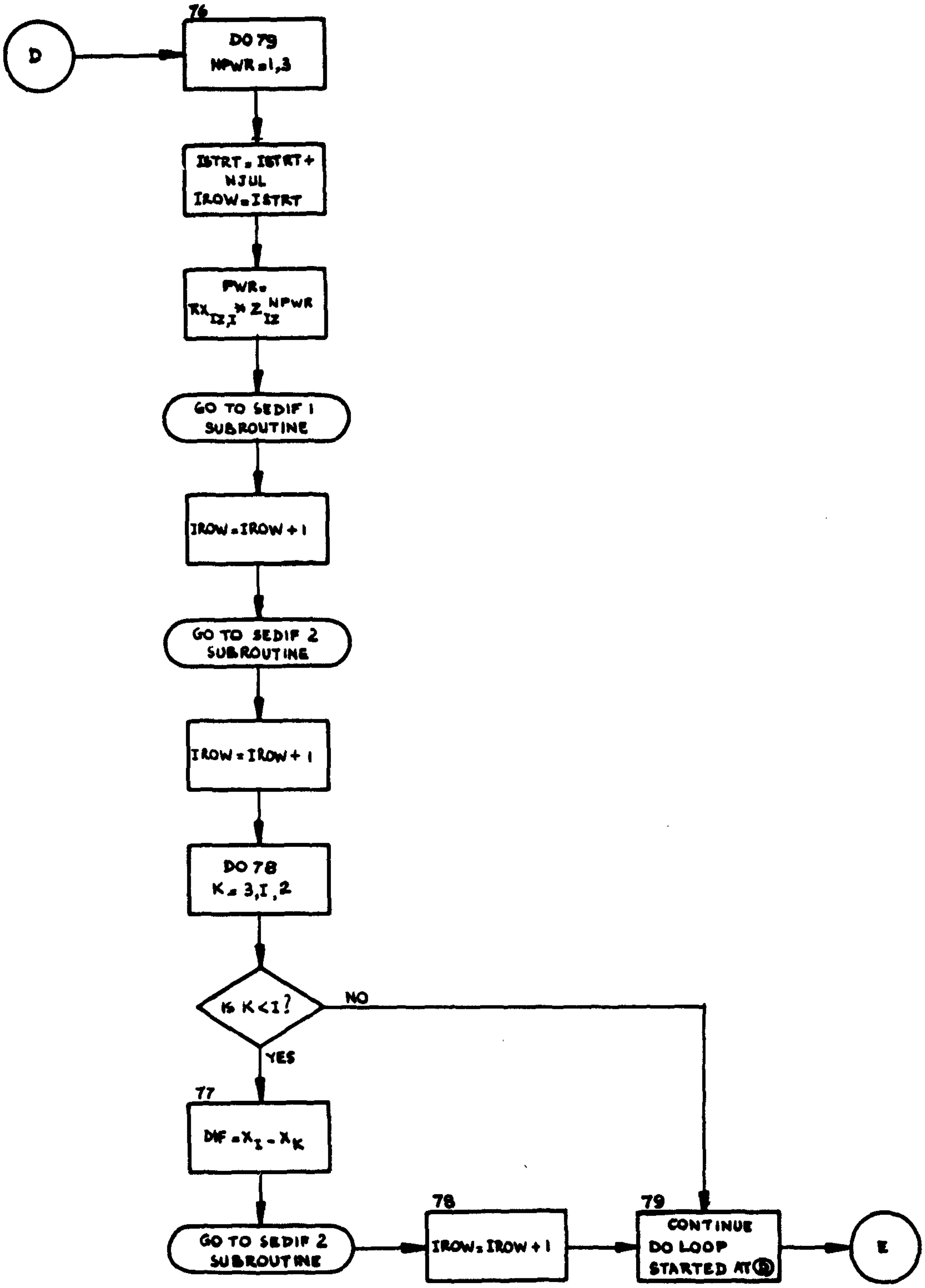
```
          C  74 R   21     48.00000
          C  75 OFSET       0.00000
          C  75 R   12      1.00000
          C  75 R   13      4.00000
          C  75 R   14     16.00000
          C  75 R   15     64.00000
          C  75 R   16      8.00000
          C  75 R   17     12.00000
          C  75 R   18     48.00000
          C  75 R   19    192.00000
          C  75 R   20    768.00000
          C  75 R   21     96.00000
          C  75 R   22      6.00000
          C  75 R   23     24.00000
          C  75 R   24     96.00000
          C  75 R   25    384.00000
          C  75 R   26     48.00000
FIRST B
                R    1     20.00000
                R    2       .00100
                R    3       .00100
                R    4       .00100
                R    5       .00100
                R    6       .00100
                R    7       .00100
                R    8       .00100
                R    9       .00100
                R   10       .00100
                R   11       .00100
                R   12       .00100
                R   13       .00100
                R   14       .00100
                R   15       .00100
                R   16       .00100
                R   17       .00100
                R   18       .00100
                R   19       .00100
                R   20       .00100
                R   21       .00100
                R   22       .00100
                R   23       .00100
```

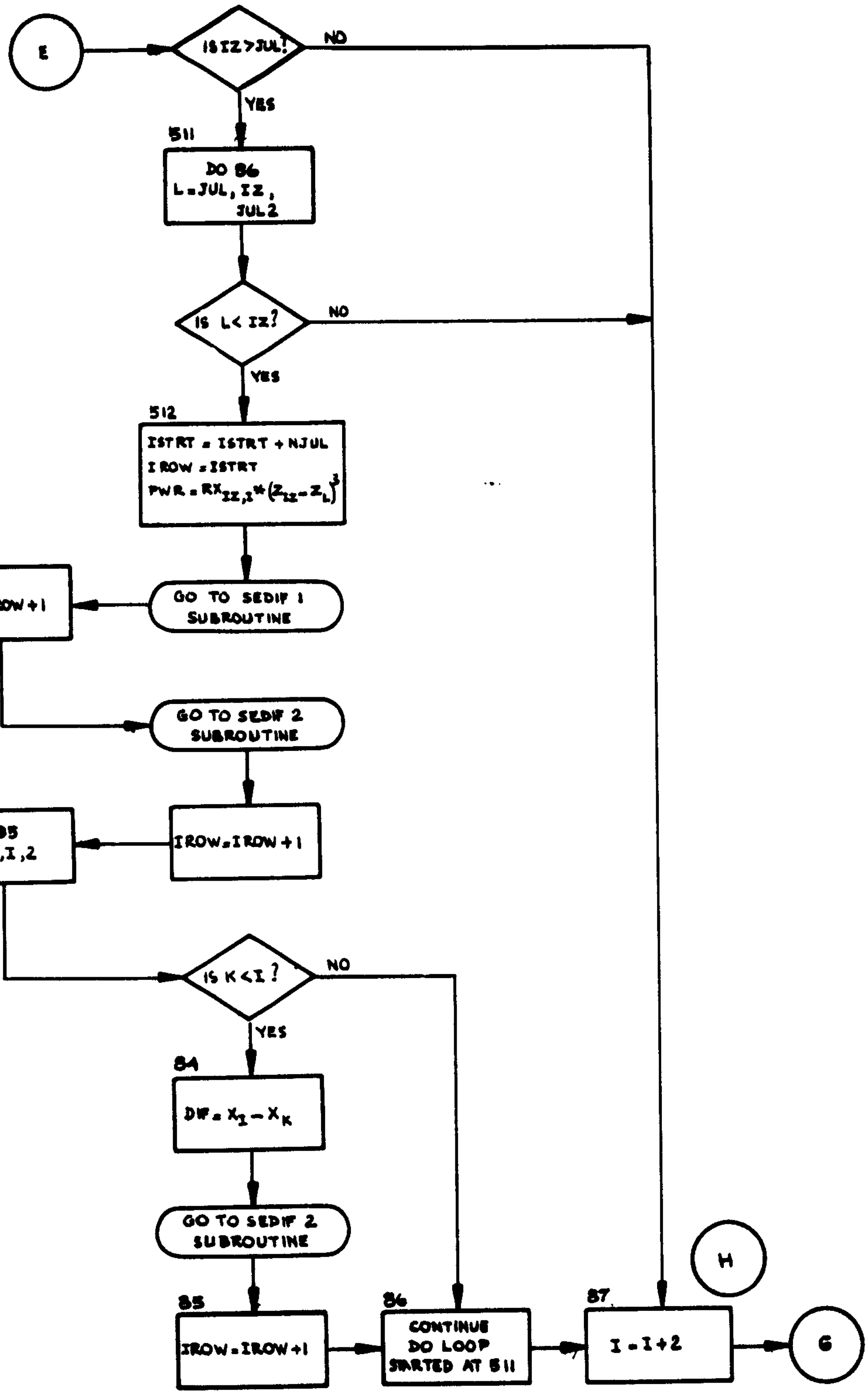
```
R  24     .00100
R  25     .00100
R  26     .00100
EOF
```

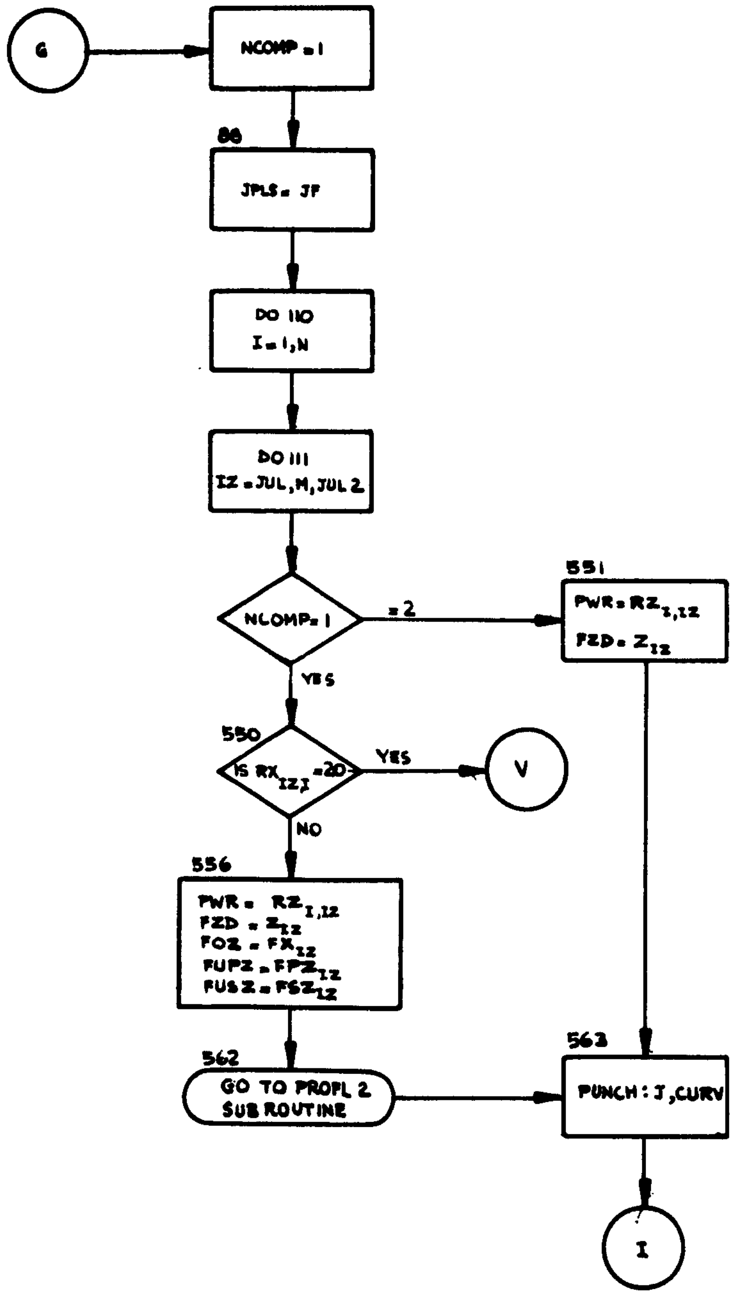

IN
750
READ: JPLS, JUL, JUL2, NJUL, M, N, MS, NS, NOROW, NCOMP, RHS, CURV.
READ: EXP2, COST, POINT, EXP 1
DO 32 I = 1, N
READ: X_I, FZ_I
DO 32 J = 2, M
32
READ: RZ_{IJ}
DO 33 I = 1, M
READ: Z_I, FX_I, FPZ_I, FSZ_I
DO 33 J = 1, N
33
READ: RX_{IJ}
B

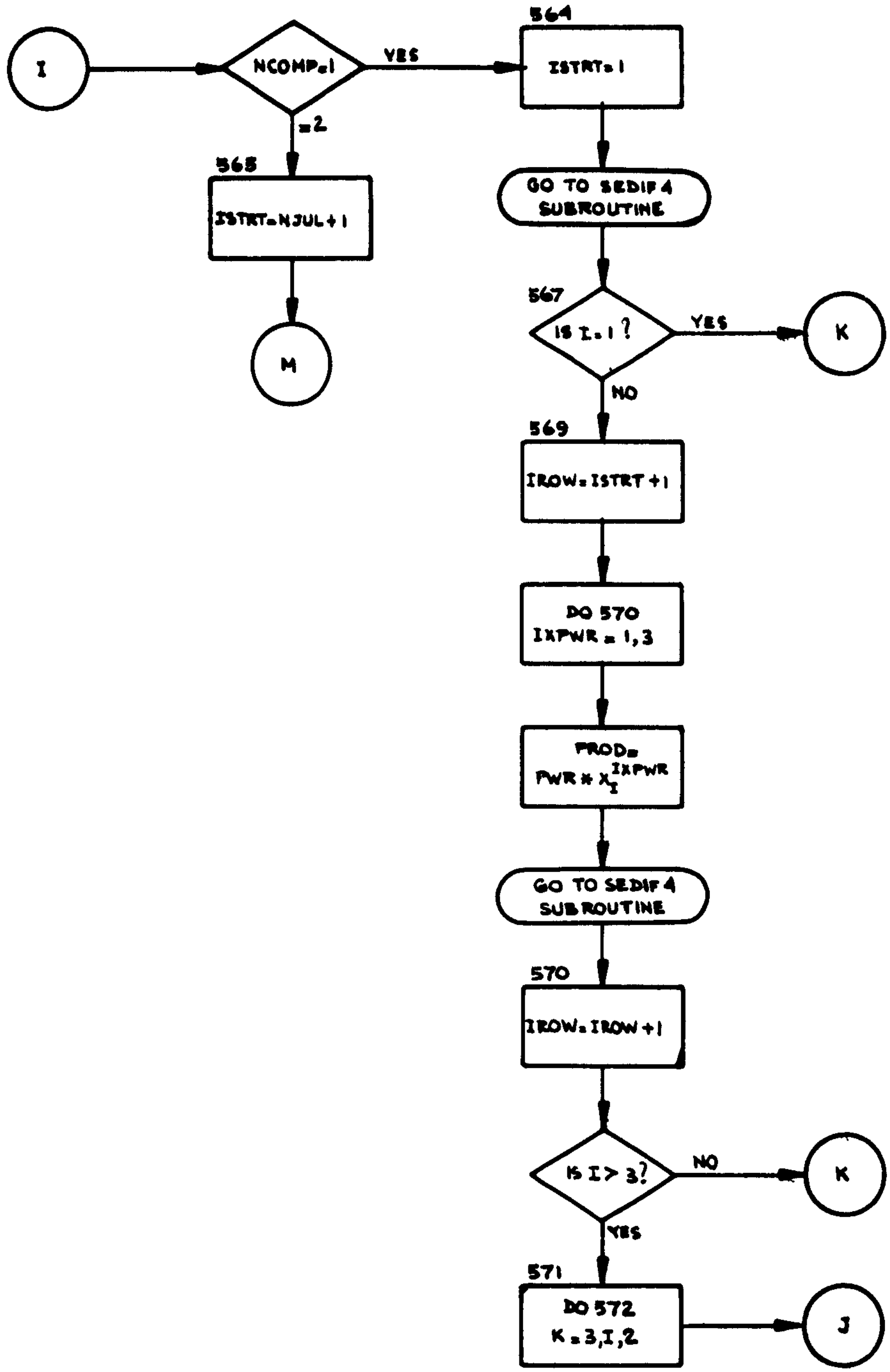

B
ML = M+2
DO 100
J = 1, ML
100
READ:
C_J
X SECOND DIFFERENCES
JX = NS/2
ICOL = ICOL + 1
DO 88
IZ = 1, M
I = 1
JS = JPLS + 1
JF = JS + JX
DO 487
J = JS, JF
$RX_{IZ,I}$: 20
=
H
≶
501
PUNCH:
J, CURV
IROW = 2
ISTRT = 2
NCOMP = 1
YES
503
FUPZ = -1
FUSZ = 0
GO TO PROFL 2
SUBROUTINE
= 2
C

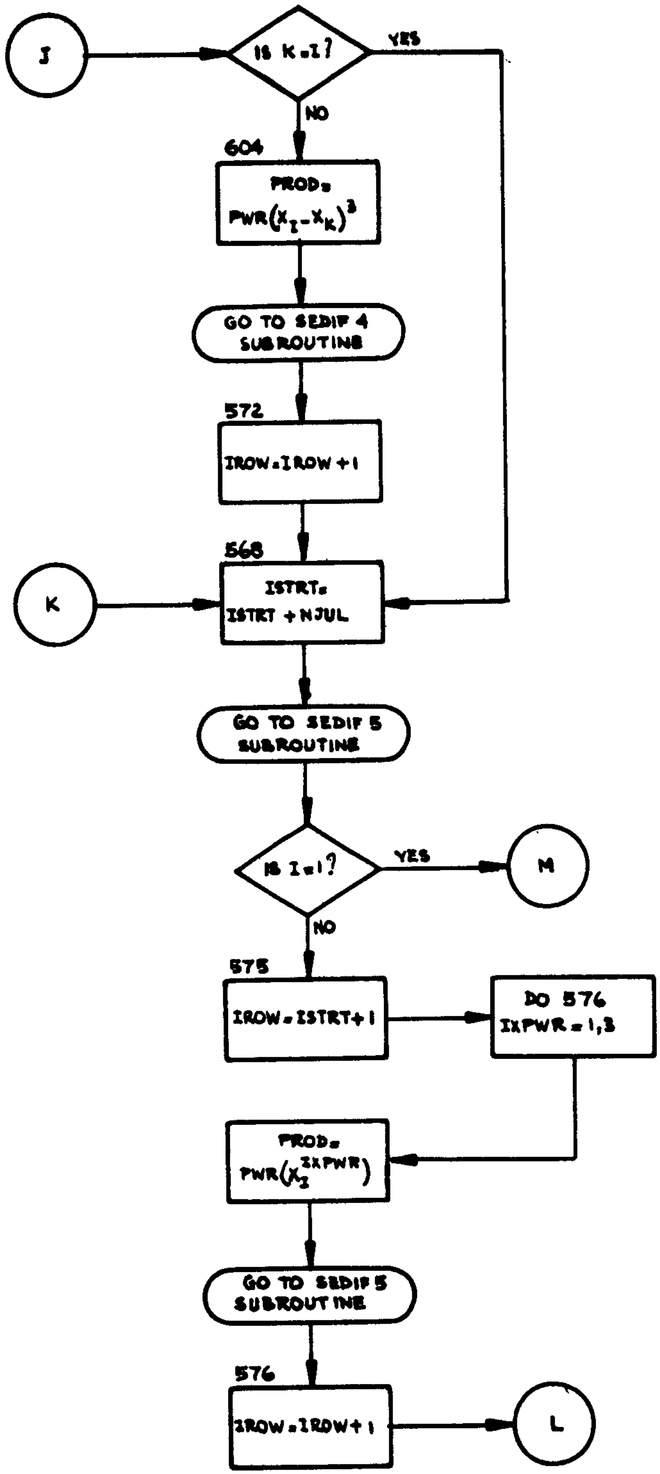

C
504
PWR = RX IZ,I
GO TO SEDIF 1 SUBROUTINE
IROW = IROW + 1
GO TO SEDIF 2 SUBROUTINE
IS I > 3?
NO
YES
502
IROW = IROW + 1
DO 505 K = 3, I, 2
IS K < I?
NO
YES
73
DIF = X I - X K
GO TO SEDIF 2 SUBROUTINE
505
IROW = IROW + 1
75
IS IZ > 1?
NO
H
YES
D

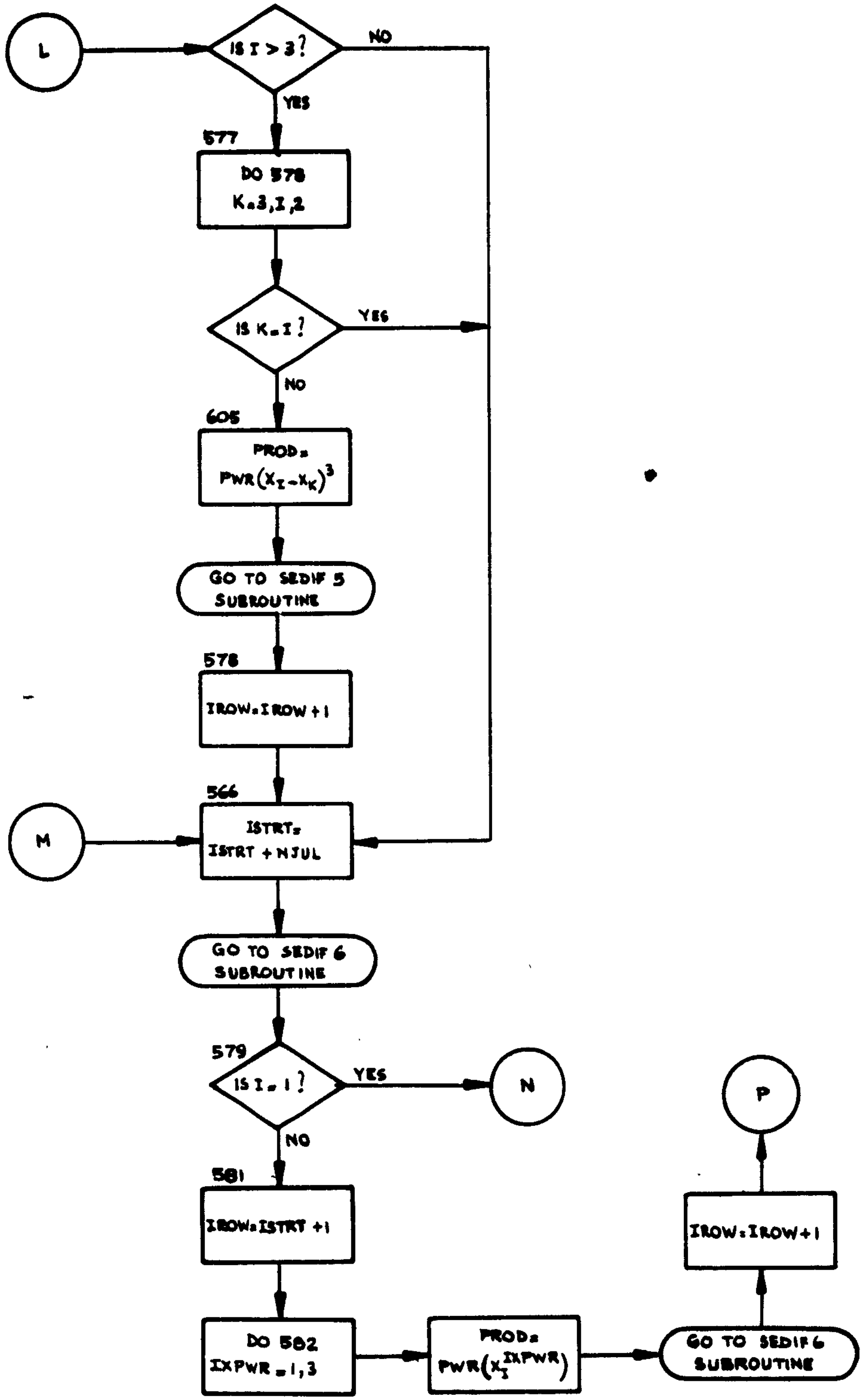

D
76
DO 79
NPWR=1,3
ISTRT=ISTRT+
NJUL
IROW=ISTRT
PWR=
TX
IZ,I
*Z
NPWR
IZ
GO TO SEDIF 1
SUBROUTINE
IROW=IROW+1
GO TO SEDIF 2
SUBROUTINE
IROW=IROW+1
DO 78
K=3,I,2
IS K<I?
NO
YES
77
DIF=X
I
-X
K
GO TO SEDIF 2
SUBROUTINE
78
IROW=IROW+1
79
CONTINUE
DO LOOP
STARTED AT (B)
E

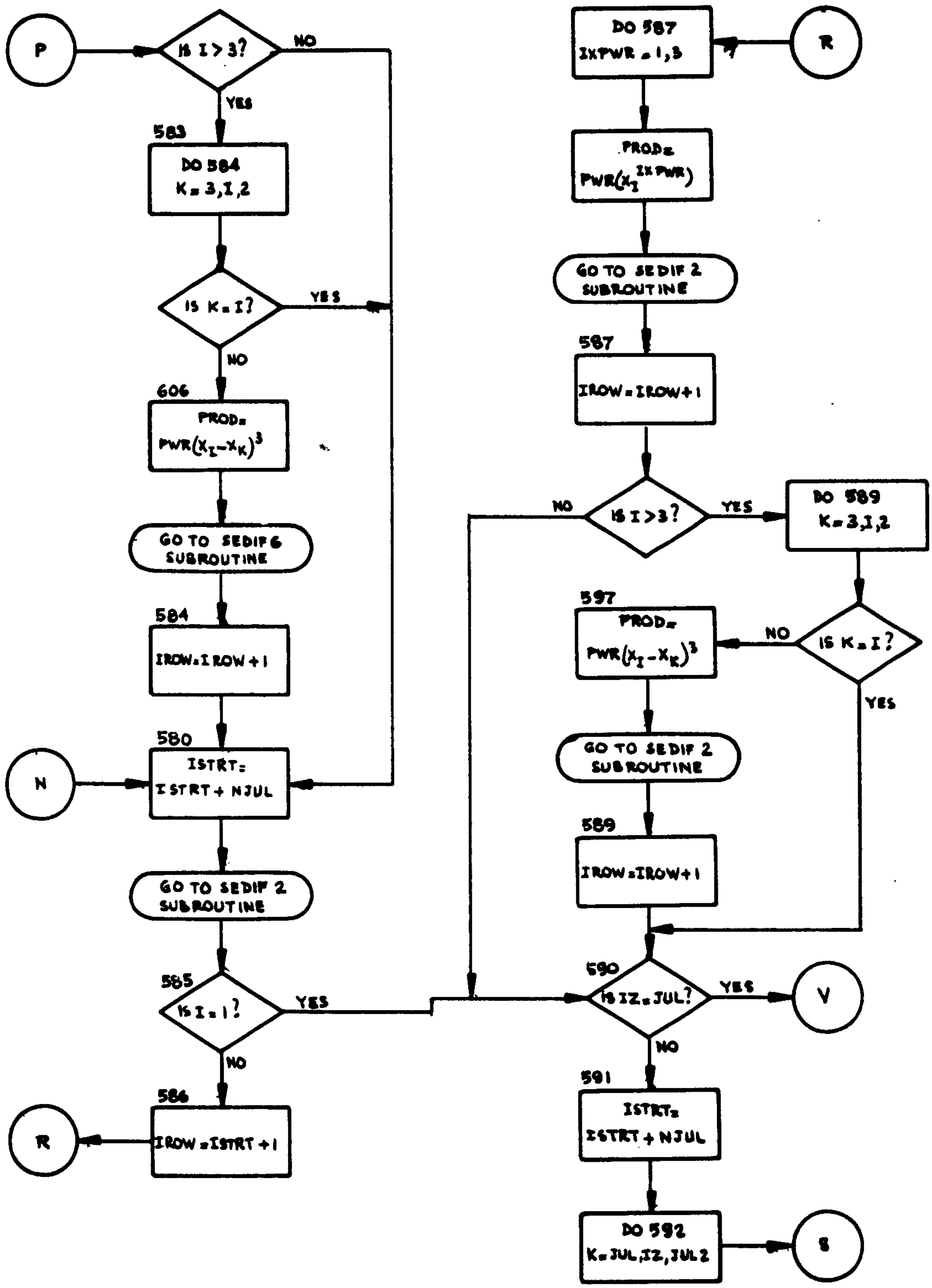

E
IS IZ > JUL?
NO
YES
511
DO 86
L = JUL, IZ,
JUL 2
IS L < IZ?
NO
YES
512
ISTRT = ISTRT + NJUL
IROW = ISTRT
PWR = RX$_{IZ,I}$ * $(Z_{IZ} - Z_L)^3$
GO TO SEDIF 1
SUBROUTINE
IROW = IROW + 1
GO TO SEDIF 2
SUBROUTINE
IROW = IROW + 1
DO 85
K = 3, I, 2
IS K < I?
NO
YES
84
DIF = $X_I - X_K$
GO TO SEDIF 2
SUBROUTINE
85
IROW = IROW + 1
86
CONTINUE
DO LOOP
STARTED AT 511
87
I = I + 2
H
G

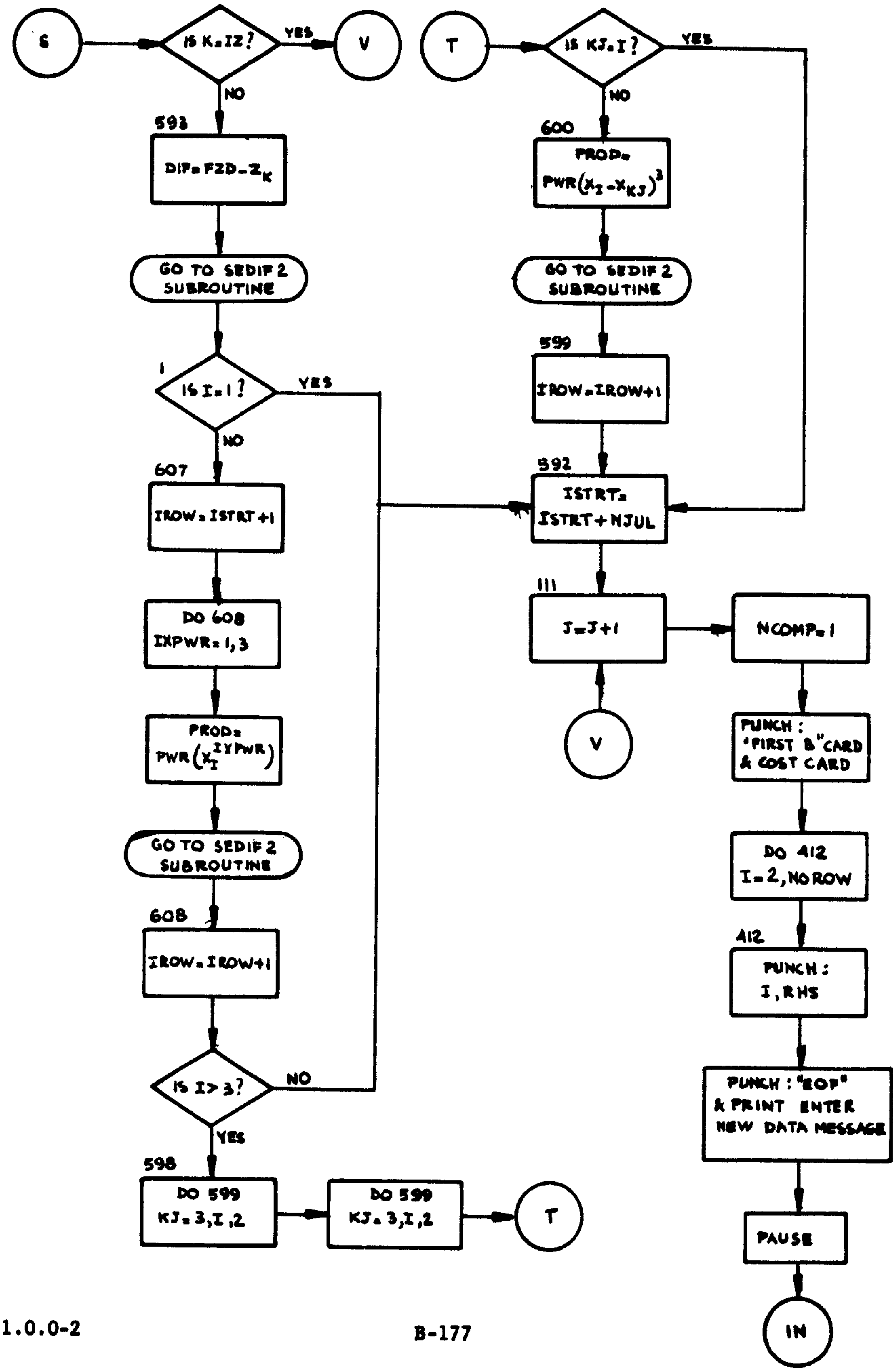

G
NCOMP = 1
86
JPLS = JF
DO 110
I = 1,N
DO 111
IZ = JUL,M,JUL 2
NCOMP = 1
= 2
YES
551
PWR = RZ I,IZ
FZD = Z IZ
550
IS RX IZ,I = 20
YES
V
NO
556
PWR = RZ I,IZ
FZD = Z IZ
FOZ = FX IZ
FUPZ = FPZ IZ
FUSZ = FSZ IZ
562
GO TO PROFL 2
SUBROUTINE
563
PUNCH : J,CURV
I


I
NCOMP=1
YES
=2
564
ISTRT=1
565
ISTRT=NJUL+1
M
GO TO SEDIF4 SUBROUTINE
567
IS I=1?
YES
K
NO
569
IROW=ISTRT+1
DO 570 IXPWR=1,3
PROD= PWR * X_I^IXPWR
GO TO SEDIF4 SUBROUTINE
570
IROW=IROW+1
IS I>3?
NO
K
YES
571
DO 572 K=3,I,2
J


J
IS K=I?
YES
NO
604
PROD=
PWR$(X_I-X_K)^3$
GO TO SEDIF 4
SUBROUTINE
572
IROW=IROW+1
568
K
ISTRT=
ISTRT+NJUL
GO TO SEDIF 5
SUBROUTINE
IS I=1?
YES
M
NO
575
IROW=ISTRT+1
DO 576
IXPWR=1,3
PROD=
PWR(X_I^{IXPWR})
GO TO SEDIF 5
SUBROUTINE
576
IROW=IROW+1
L


L
IS I > 3?
NO
YES
577
DO 578
K=3,I,2
IS K=I?
YES
NO
605
PROD=
PWR(X_I - X_K)^3
GO TO SEDIF 5
SUBROUTINE
578
IROW=IROW+1
566
M
ISTRT=
ISTRT + NJUL
GO TO SEDIF 6
SUBROUTINE
579
IS I=1?
YES
N
NO
581
IROW=ISTRT+1
DO 582
IXPWR=1,3
PROD=
PWR(X_I^IXPWR)
GO TO SEDIF6
SUBROUTINE
IROW=IROW+1
P

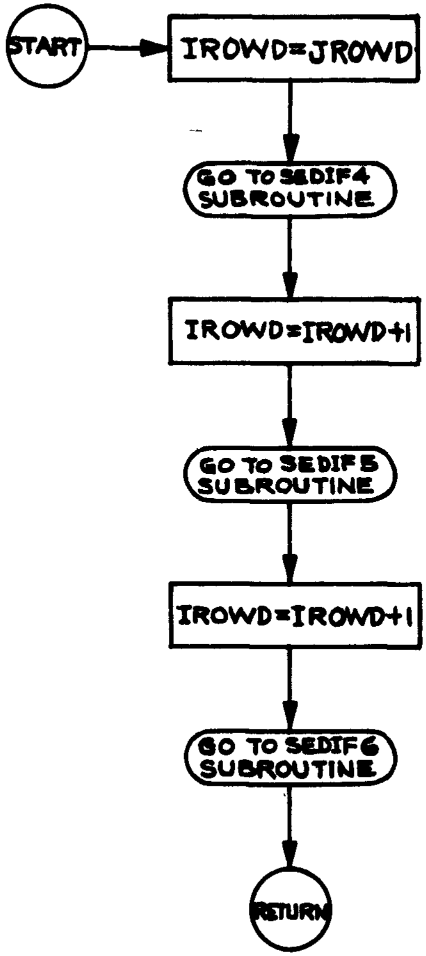

P
IS I > 3?
NO
YES
583
DO 584
K = 3,I,2
IS K = I?
YES
NO
606
PROD=
PWR(X_I-X_K)^3
GO TO SEDIF6 SUBROUTINE
584
IROW=IROW+1
580
N
ISTRT=
ISTRT+NJUL
GO TO SEDIF 2 SUBROUTINE
585
IS I = 1?
YES
NO
586
R
IROW=ISTRT+1
DO 587
IXPWR = 1,3
R
PROD=
PWR(X_I^IXPWR)
GO TO SEDIF 2 SUBROUTINE
587
IROW=IROW+1
NO
IS I>3?
YES
DO 589
K=3,I,2
597
PROD=
PWR(X_I-X_K)^3
NO
IS K=I?
YES
GO TO SEDIF 2 SUBROUTINE
589
IROW=IROW+1
590
IS IZ=JUL?
YES
V
NO
591
ISTRT=
ISTRT+NJUL
DO 592
K=JUL,IZ,JUL2
S



S
IS K=IZ?
YES
V
NO
593
DIF=FZD-Z_K
GO TO SEDIF 2 SUBROUTINE
1
IS I=1?
YES
NO
607
IROW=ISTRT+1
DO 608 IXPWR=1,3
PROD= PWR(X_I^IXPWR)
GO TO SEDIF 2 SUBROUTINE
608
IROW=IROW+1
IS I>3?
NO
YES
598
DO 599 KJ=3,I,2
DO 599 KJ=3,I,2
T
T
IS KJ=I?
YES
NO
600
PROD= PWR(X_I-X_KJ)^3
GO TO SEDIF 2 SUBROUTINE
599
IROW=IROW+1
592
ISTRT= ISTRT+NJUL
111
J=J+1
V
NCOMP=1
PUNCH: "FIRST B" CARD & COST CARD
DO 412 I=2,NOROW
412
PUNCH: I,RHS
PUNCH: "EOF" & PRINT ENTER NEW DATA MESSAGE
PAUSE
IN

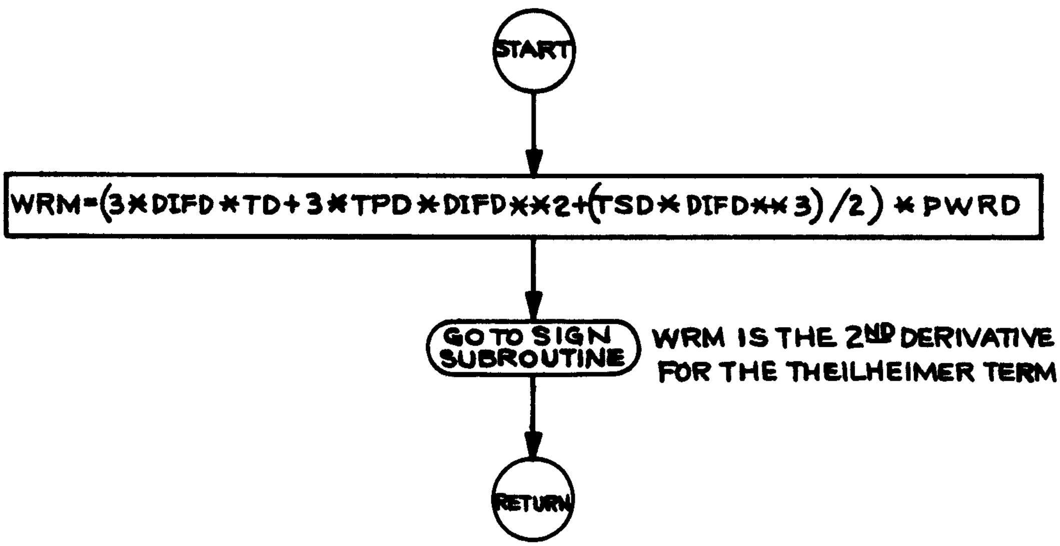
For Flow Diagram of Profile 2 Subroutine, See SMOG 2, Pass 1

SUBROUTINE SEDIF1(TD,TPD,TSD,XD,JROWD,JD,PWRD,IROWD)

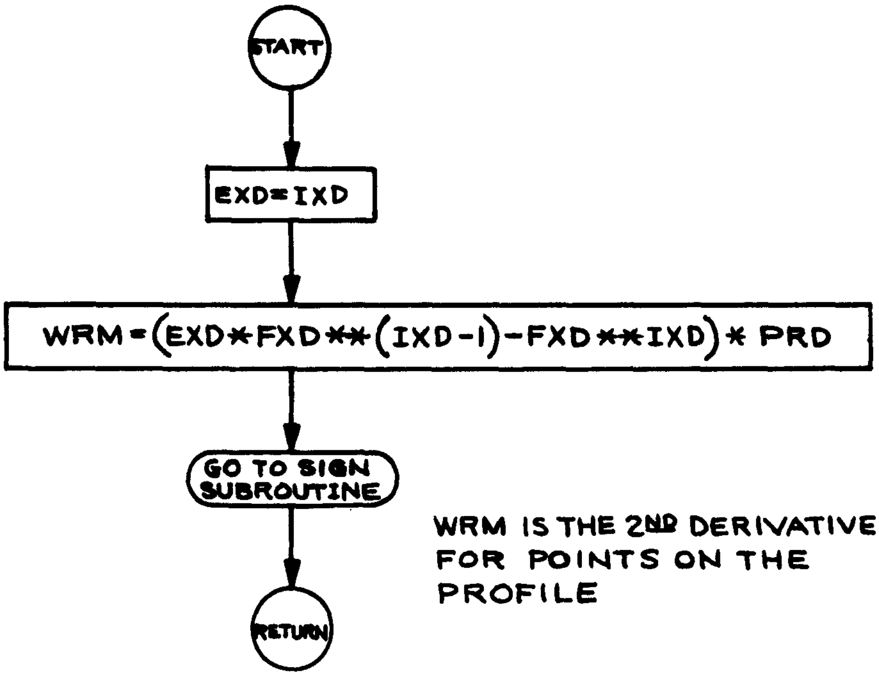


THIS SUBROUTINE COMPUTES THE 2ND DERIVATIVE OF THE STANDARD CUBIC USING SEDIF4, SEDIF5, & SEDIF6 SUBROUTINES

SUBROUTINE SEDIF2(DIFD,TD,TPD,TSD,PWRD,IROWD,JD)

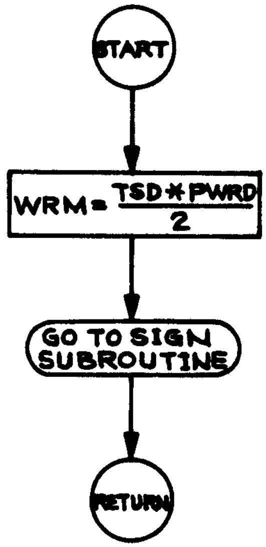


SUBROUTINE SEDIF3(IXD, FXD, PRD, IROWD, JSD)

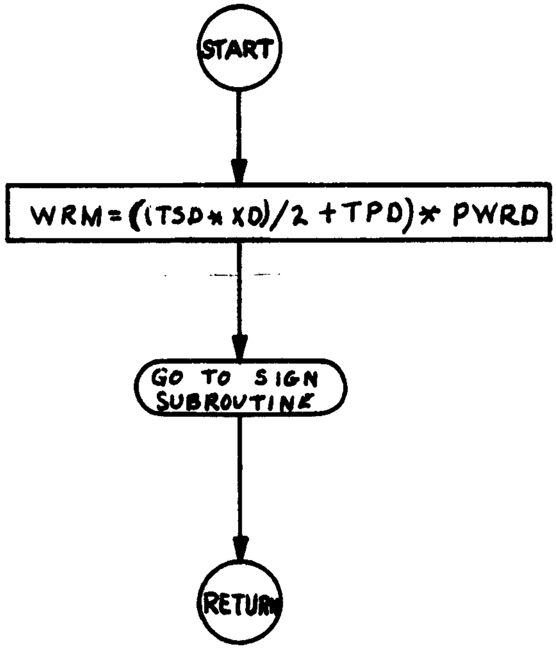


WRM IS THE 2ND DERIVATIVE FOR POINTS ON THE PROFILE

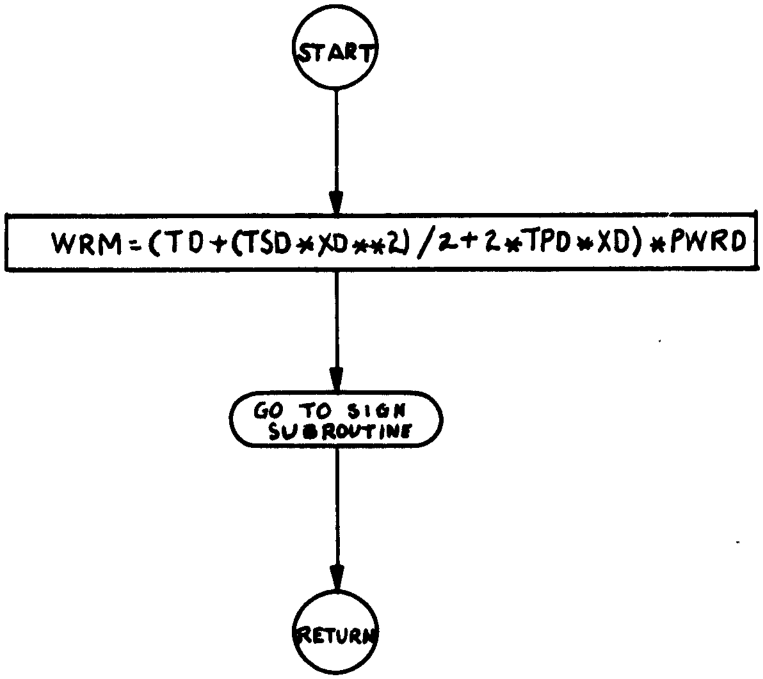
SUBROUTINE SEDIF4(TSD,PWRD,JD,IROWD)



SUBROUTINE SEDIF5 (TPD, TSD, XD, PWRD, JD, IROWD)

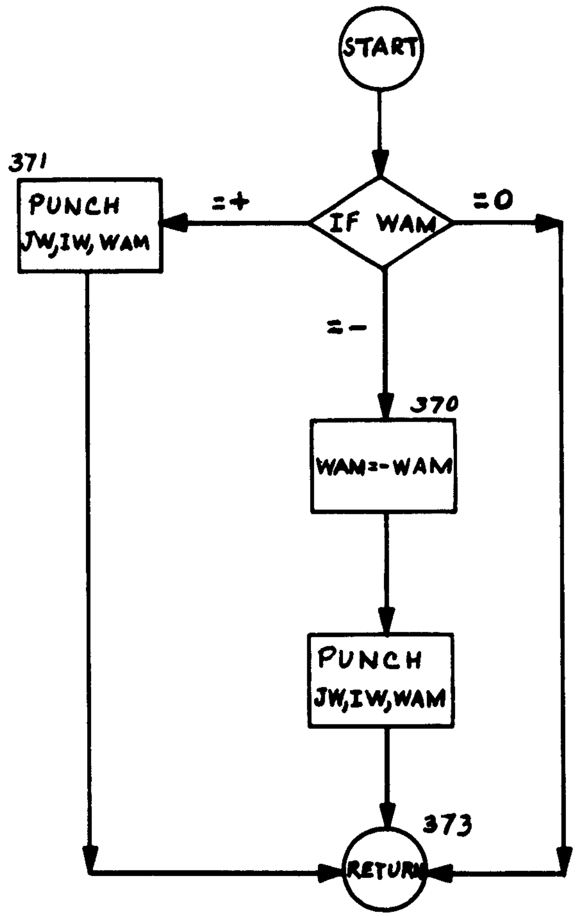


SUBROUTINE SEDIF6 (TD, TPD, TSD, XD, JD, PWRD, IROWD)



SUBROUTINE SIGN (JW, IW, WAM)

THIS SUBROUTINE PUNCHES THE 2ND DERIVATIVE FROM THE SEDIF SUBROUTINES, ALONG WITH THE IDENTIFYING SUBSCRIPT NO'S I & J, IN PROPER FORMAT



```
C     SMOG2 PASS 2  DUAL X OR X AND Z DBL SPL  WITH PROFILE
      DIMENSION RX(11,15),RZ(15,11),FZ(11),FX(15),X(15),Z(11),Y(11,
15)
      DIMENSION  FPZ(11),FSZ(11),C(13)
  750 READ 180,JPLS,JUL,JUL2,NJUL,M,N,MS,NS,NOROW,NCOMP,CURV,RHS
      READ 181,EXP2,COST,POINT,EXP1
      DO 32 I=1,N
      READ 181,X(I),FZ(I)
      DO 32 J=2,M
   32 READ 181,RZ(I,J)
      DO 33 I=1,M
      READ 181,Z(I),FX(I),FPZ(I),FSZ(I)
      DO 33 J=1,N
   33 READ 181,RX(I,J)
      ML=M+2
      DO 100 J=1,ML
  100 READ 181,C(J)
C        *    *    X SEC  DIFFERENCES   *    *
      JX=NS/2
      DO 88 IZ=1,M
      I=1
      JS=JPLS+1
      JF=JS+JX
      DO 87 J=JS,JF
      IF(RX(IZ,I)-20.)501,87,501
  501 PUNCH 171,J,CURV
      IROW=2
      ISTRT=2
      GO TO (503,504),NCOMP
  503 FUPZ=-1.
      FUSZ=0.
      CALL PROFL2 (X(I),FX(IZ),EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS
,NCOMP)
  504 PWR=RX(IZ,I)
      CALL SEDIF1 (T,TP,TS,X(I),IROW,J,PWR,IROW)
      IROW=IROW+1
      CALL SEDIF2 (X(I),T,TP,TS,PWR,IROW,J)
      IF(I-3)75,75,502
  502 IROW=IROW+1
      DO 505 K=3,I,2
      IF(K-I)73,75,75
   73 DIF=X(I)-X(K)
      CALL SEDIF2 (DIF,T,TP,TS,PWR,IROW,J)
  505 IROW=IROW+1
   75 IF(IZ-1)87,87,76
```

```
   76 DO 79 NPWR=1,3
      ISTRT=ISTRT+NJUL
      IROW=ISTRT
      PWR=RX(IZ,I)*Z(IZ)**NPWR
      CALL SEDIF1 (T,TP,TS,X(I),IROW,J,PWR,IROW)
      IROW=IROW+1
      CALL SEDIF2 (X(I),T,TP,TS,PWR,IROW,J)
      IROW=IROW+1
      DO 78 K=3,I,2
      IF(K-I)77,79,79
   77 DIF=X(I)-X(K)
      CALL SEDIF2 (DIF,T,TP,TS,PWR,IROW,J)
   78 IROW=IROW+1
   79 CONTINUE
      IF(IZ-JUL)87,87,511
  511 DO 86 L=JUL,IZ,JUL2
      IF(L-IZ)512,87,87
  512 ISTRT=ISTRT+NJUL
      IROW=ISTRT
      PWR=RX(IZ,I)*(Z(IZ)-Z(L))**3
      CALL SEDIF1 (T,TP,TS,X(I),IROW,J,PWR,IROW)
      IROW=IROW+1
      CALL SEDIF2 (X(I),T,TP,TS,PWR,IROW,J)
      IROW=IROW+1
      DO 85 K=3,I,2
      IF(K-I)84,86,86
   84 DIF=X(I)-X(K)
      CALL SEDIF2 (DIF,T,TP,TS,PWR,IROW,J)
   85 IROW=IROW+1
   86 CONTINUE
   87 I=I+2
      NCOMP=1
   88 JPLS=JF
C         Z  SEC  DIFFERENCES    *    *
      DO 110 I=1,N
      DO 111 IZ=JUL,M,JUL2
      GO TO (550,551),NCOMP
  550 IF(RX(IZ,I)-20.)556,111,556
  556 PWR=RZ(I,IZ)
      FZD=Z(IZ)
      FOZ=FX(IZ)
      FUPZ=FPZ(IZ)
      FUSZ=FSZ(IZ)
  562 CALL PROFL2(X(I),FOZ,EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS,
NCOMP)
```

```
      GO TO 563
  551 PWR=RZ(I,IZ)
      FZD=Z(IZ)
  563 PUNCH 171,J,CURV
      GO TO (564,565),NCOMP
  565 ISTRT=2+NJUL
      GO TO 566
  564 ISTRT=2
      CALL SEDIF4 (TS,PWR,J,ISTRT)
  567 IF(I-1)569,568,569
  569 IROW=ISTRT+1
      DO 570 IXPWR=1,3
      PROD=PWR*X(I)**IXPWR
      CALL SEDIF4(TS,PROD,J,IROW)
  570 IROW=IROW+1
      IF(I-3)568,568,571
  571 DO 572 K=3,I,2
      IF(K-I)604,568,604
  604 PROD=PWR*(X(I)-X(K))**3
      CALL SEDIF4(TS,PROD,J,IROW)
  572 IROW=IROW+1
  568 ISTRT=ISTRT+NJUL
      CALL SEDIF5(TP,TS,FZD,PWR,J,ISTRT)
  573 IF(I-1)575,566,575
  575 IROW=ISTRT+1
      DO 576 IXPWR=1,3
      PROD=PWR*X(I)**IXPWR
      CALL SEDIF5 (TP,TS,FZD,PROD,J,IROW)
  576 IROW=IROW+1
      IF(I-3)566,566,577
  577 DO 578 K=3,I,2
      IF(K-I)605,566,605
  605 PROD=PWR*(X(I)-X(K))**3
      CALL SEDIF5 (TP,TS,FZD,PROD,J,IROW)
  578 IROW=IROW+1
  566 ISTRT=ISTRT+NJUL
      CALL SEDIF6 (T,TP,TS,FZD,J,PWR,ISTRT)
  579 IF(I-1)581,580,581
  581 IROW=ISTRT+1
      DO 582 IXPWR=1,3
      PROD=PWR*X(I)**IXPWR
      CALL SEDIF6 (T,TP,TS,FZD,J,PROD,IROW)
  582 IROW=IROW+1
      IF(I-3)580,580,583
  583 DO 584 K=3,I,2
      IF(K-I)606,580,606
```

```
606 PROD=PWR*(X(I)-X(K))**3
    CALL SEDIF6 (T,TP,TS,FZD,J,PROD,IROW)
584 IROW=IROW+1
580 ISTRT=ISTRT+NJUL
    CALL SEDIF2 (FZD,T,TP,TS,PWR,ISTRT,J)
585 IF(I-1)586,590,586
586 IROW=ISTRT+1
    DO 587 IXPWR=1,3
    PROD=PWR*X(I)**IXPWR
    CALL SEDIF2 (FZD,T,TP,TS,PROD,IROW,J)
587 IROW=IROW+1
    IF(I-3)590,590,588
588 DO 589 K=3,I,2
    IF(K-I)597,590,597
597 PROD=PWR*(X(I)-X(K))**3
    CALL SEDIF2 (FZD,T,TP,TS,PROD,IROW,J)
589 IROW=IROW+1
590 IF(IZ-JUL)591,111,591
591 ISTRT=ISTRT+NJUL
    DO 592 K=JUL,IZ,JUL2
    IF(K-IZ)593,111,593
593 DIF=FZD-Z(K)
    CALL SEDIF2 (DIF,T,TP,TS,PWR,ISTRT,J)
  1 IF(I-1)607,592,607
607 IROW=ISTRT+1
    DO 608 IXPWR=1,3
    PROD=PWR*X(I)**IXPWR
    CALL SEDIF2 (DIF,T,TP,TS,PROD,IROW,J)
608 IROW=IROW+1
    IF(I-3)592,592,598
598 DO 599 KJ=3,I,2
    IF(KJ-I)600,592,600
600 PROD=PWR*(X(I)-X(KJ))**3
    CALL SEDIF2 (DIF,T,TP,TS,PROD,IROW,J)
599 IROW=IROW+1
592 ISTRT=ISTRT+NJUL
111 J=J+1
110 NCOMP=1
    PUNCH 174
    PUNCH 175,COST
    DO 412 I=2,NOROW
412 PUNCH 177,I,RHS
    PUNCH 176
    PRINT 178
    PAUSE 751
```

```
      GO TO 750
171   FORMAT(6X,2H C,I4,6H OFSET,F12.5)
173   FORMAT(1H*,5X,47H
     )
174   FORMAT(7HFIRST B)
175   FORMAT(12X, 6H R    1,F12.5)
176   FORMAT(3HEOF)
177   FORMAT(12X,2H R,I4,1H ,F11.5)
178   FORMAT(28HENTER NEW DATA - PRESS START)
180   FORMAT (10I5,2F15.9)
181   FORMAT (4F15.9)
      END
```

```
      SUBROUTINE PROFL2(XD,FD,EXPD1,TZPD,TZSD,EXPD2,X1,TD,TPD,TSD,NCD)
      TD=(X1+FD-XD)/X1
      IF(TD-.00001)1,1,2
    2 DIFD=EXPD1-1.
      PROD=TD**DIFD
      TPD1=EXPD1*TZPD*PROD
      TSD1=EXPD1*(DIFD*TD**(DIFD-1.)*TZPD**2+TZSD*PROD)
      TD=1.-TD**EXPD1
      IF(TD-.00001)7,7,4
    4 DIFD=EXPD2-1.
      TDEXP=TD**DIFD
      TPD=-EXPD2*TPD1*TDEXP
      TSD=EXPD2*(DIFD*TD**(EXPD2-2.)*TPD1*TPD1-TSD1*TDEXP)
      TD=TD**EXPD2
      IF(TD-.00001)7,7,6
    7 TD=0.
      GO TO 3
    6 NCD=1
      RETURN
    1 NCD=2
      TD=1.
    5 TPD=0.
      TSD=0.
      RETURN
    3 NCD=1
      GO TO 5
      END


      SUBROUTINE SEDIF1 (TD,TPD,TSD,XD,JROWD,JD,PWRD,IROWD)
      IROWD=JROWD
      CALL SEDIF4 (TSD,PWRD,JD,IROWD)
      IROWD=IROWD+1
      CALL SEDIF5 (TPD,TSD,XD,PWRD,JD,IROWD)
      IROWD=IROWD+1
      CALL SEDIF6 (TD,TPD,TSD,XD,JD,PWRD,IROWD)
      RETURN
      END


      SUBROUTINE SEDIF2 (DIFD,TD,TPD,TSD,PWRD,IROWD,JD)
      WRM=(3.*DIFD*TD+3.*TPD*DIFD**2+(TSD*DIFD**3)/2.)*PWRD
      CALL SIGN (JD,IROWD,WRM)
      RETURN
      END
```

```
      SUBROUTINE SEDIF3 (IXD,FXD,PRD,IROWD,JSD)
      EXD=IXD
      WRM=(EXD*FXD**(IXD-1)-FXD**IXD)*PRD
      CALL SIGN (JSD,IROWD,WRM)
      RETURN
      END


      SUBROUTINE SEDIF4 (TSD,PWRD,JD,IROWD)
      WRM=(TSD*PWRD)/2.
      CALL SIGN (JD,IROWD,WRM)
      RETURN
      END


      SUBROUTINE SEDIF5 (TPD,TSD,XD,PWRD,JD,IROWD)
      WRM=((TSD*XD)/2.+TPD)*PWRD
      CALL SIGN (JD,IROWD,WRM)
      RETURN
      END


      SUBROUTINE SEDIF6 (TD,TPD,TSD,XD,JD,PWRD,IROWD)
      WRM=TD
      WRM=WRM+(TSD*XD**2)/2.
      WRM=WRM+2.*TPD*XD
      WRM=WRM*PWRD
      CALL SIGN (JD,IROWD,WRM)
      RETURN
      END


      SUBROUTINE SIGN (JW,IW,WAM)
      IF(WAM)370,373,371
  370 WAM=-WAM
      PUNCH 169,JW,IW,WAM
      RETURN
  371 PUNCH 167,JW,IW,WAM
  373 RETURN
  167 FORMAT(6X,2H C,I4,2H R,I4,1H-,F11.5)
  169 FORMAT(6X,2H C,I4,2H R,I4,F12.5)
      END
```

Section III

SMOG 3

The purpose of the program named SMOG 3 is to generate a matrix with the following characteristics:

(1) Primal formulation
(2) Sum of the deviations fit
(3) Double splined in the x direction
(4) Single or double splined in the z direction
(5) Containing profile conditions

Figure B-4 (Section II of this Appendix) shows a five-station by five-waterline surface and an equation representing this surface. The matrix which represents this equation consists of two parts. The first part consists of the deviation constraints. The second part is the curvature constraints.

The deviation constraints have the form:

$$\text{(a)} \qquad Y(x_i, z_j) + S_{i,j} - Q_{i,j} = Y_{i,j}$$

thus offset (1,3) Fig. B-4 would have the constraint:

$$\text{(a)} \qquad (a_{01} - b_{01})x_3 + (a_{02} - b_{02})x_3^{\,2} + (a_{03} - b_{03})x_3^{\,3} + S_{1,3} - Q_{1,3} = Y_{1,3}$$

where

$$a_{01} - b_{01} = A_{01}$$

$$a_{02} - b_{02} = A_{02}$$

$$a_{03} - b_{03} = A_{03}$$

The curvature constraints are the same as those of SMOG 1 and SMOG 2

(b) $\hat{r}_{i,j} \cdot f''(x_i,z_j)_{xx} \geq 0$

(c) $\hat{r}_{i,j} \cdot f''(x_i,z_j)zz \geq 0$

A tableau of the matrix is given in Fig. B-6. The positive identity matrix corresponding to $S_{i,j}$ is always used to provide a basic feasible solution, and SMOG 3 provides the basis headings for this purpose.

Data Assembly Instructions

Before the data punched by Passes 1 and 2 of SMOG 3 can be used, it must be rearranged according to the instructions which follow:

Pass 1

(1) Be sure Pass 2 input data has been removed from behind the output deck

(2) Remove the first four cards from the front of the output deck. These are punched with the heading, ROW ID, OFSET, MATRIX.

(3) Sort the output deck on Column 16 with the edit off. The output deck will fall in the reject hopper and the cost row cards will fall in the Column 2 hopper.

Set the cost row elements aside.

Pass 2

(1) Remove the last two cards from the output deck. These are punched FIRST B and EOF

(2) Place the Pass 2 output deck directly behind the Pass 1 output deck.

Pass 1 and 2

(1) Sort the combined deck on Column 8 with the edit off. The right-hand side cards and row identification will fall in the reject hopper. Set these aside.

(2) Sort the combined deck on Column 25 with the edit off. The basis heading cards will fall in the reject hopper.

(3) Keypunch a card with BASIS in Columns 1-5 and place in front of the basis heading deck.

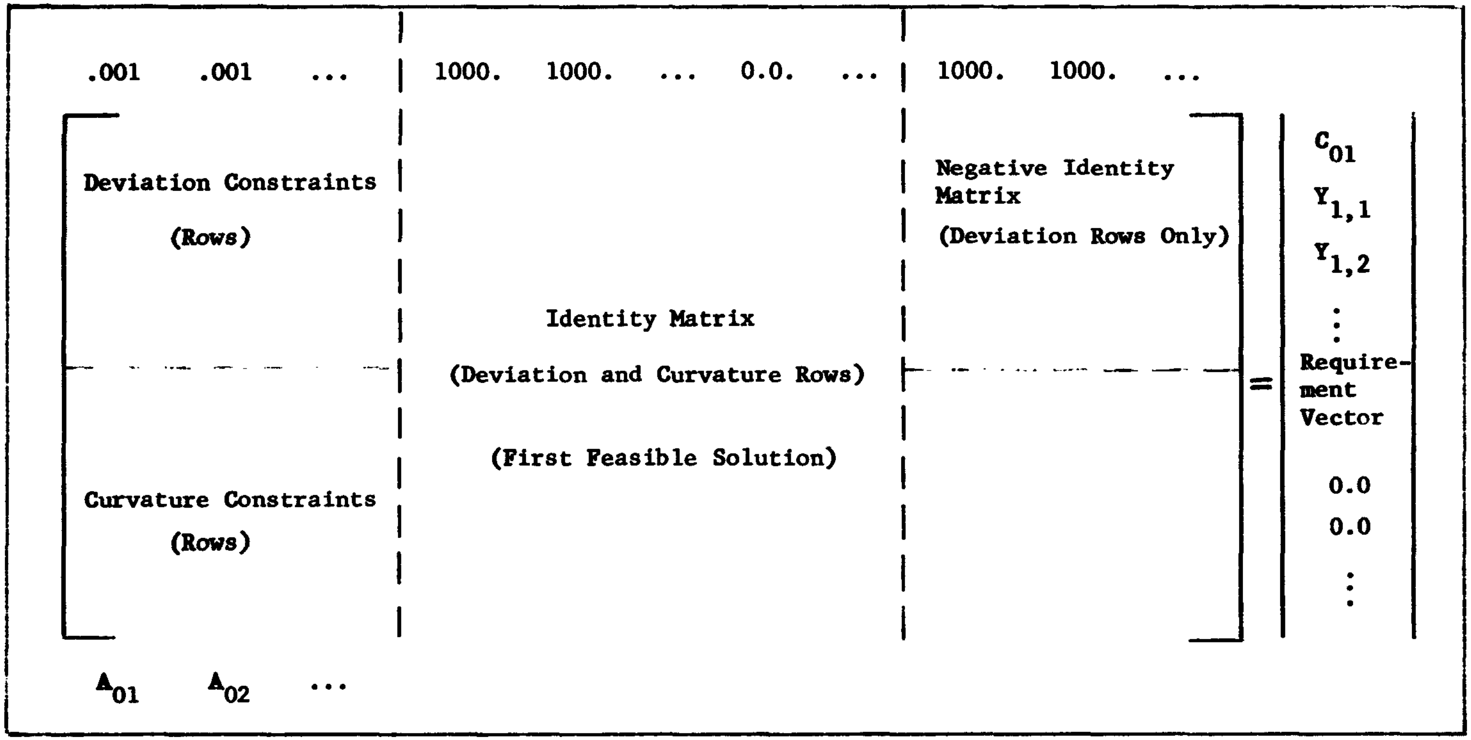


Fig. B-6 Matrix Tableau

(4) Sort the row identification - right-hand side deck on Column 25 with the edit off. The row identification cards will fall in the reject hopper.

(5) Place the following cards in the fron of a card box in the following order:

Heading card

Row ID

OFSET

The row identification deck

The basis heading deck

MATRIX

(6) Sort the combined output decks on Columns 12, 11, 10. This will place them in Column order. Put the resulting deck behind the MATRIX card.

(7) Put the First B card ahead of the right-hand side deck and the EOF card behind it. Then put this deck behind the combined output decks.

A fully assembled output deck combining the sample problem of Pass 1 and Pass 2 is given immediately following the Pass 2 sample output listing.

OPERATING INSTRUCTIONS

Pass 1 of SMOG 3 calculates the second differences of the offsets and the intersections of the profile with the waterlines and stations. It then punches the deviation constraints for the matrix. The program is written in FORTRAN II for the IBM-1620 computer.

Fortran Symbol Definitions

M	-	Total number of waterlines in the surface to be faired. This will be an odd number if the surface is double splined in the z direction.
N	-	Total number of stations in the surface. This will always be an odd number.
X(J)	-	Distance along the x axis from the origin of the surface to station j .
Z(I)	-	Distance along the z axis from the origin of the surface to Waterline i .
Y(I,J)	-	Offset of the surface at waterline i , station j , in feet and decimals
IAY	-	Number of feet in the offset if read in feet, inches and eighths.
IBY	-	Number of inches corresponding to IAY
ICY	-	Number of eighths corresponding to IAY
NAA	-	Program indicator If NAA = -1, scale factors for normalizing and translating x(J) and z(I) are read in If NAA = +1, scaling and translating are done by the program as given in Section II .
SX	-	Scale factor for X(J) when these are read in
TX	-	Translation factor for X(J) when these are read in
SZ	-	Scale factor for Z(I) when these are read in
TZ	-	Translation factor for Z(I) when these are read in

RHS - The cost on the slack variables associated with the deviation constraints. Usually made equal to 1000 .

CURV - The cost on the non-slack variables. Usually made .001 .

NAB - Program Indicator

If NAB = -1 , the surface will be double splined in the z direction, and M must be an odd number

If NAB = +1 , the surface will be single splined in the z direction and M is unrestricted

COST - Used in Pass 2. A zero value must be given

EXP1 - The value of p in the profile function, usually made .3333

EXP2 - The value of r in the profile function, usually 3.0 .

POINT - The value of D in the profile function. This number is the length of the fraction of the station spacing over which the profile function is effective in terms of the full-scale surface (the value will be translated and normalized exactly like the x values are).

C(I) - Coefficients of the Theilheimer equation describing the profile. There must be (M+2) coefficients. Straight lines or other curves may be used by making all except the desired coefficients zero.

Input

The input data cards are described below. There are certain input variables on which there are restrictions on the value which can be entered. These restrictions follow.

Variable	Minimum	Maximum
N	3	15
M	3	11

In describing the various data cards, the actual FORTRAN format field description is used in most cases. These fields come consecutively across the card with no gaps or blank columns between, except where indicated. The field descriptions are the FORTRAN F field, which uses the FORTRAN fixed point decimal number, and the I field that uses the FORTRAN integer number, which is always right justified. The card numbers are <u>not</u> punched on the data cards.

Contents of Card							Card No.
Card Columns	1	2-12	13-53				1
Contents	*	Blank	Any alphameric problem heading (This heading will be used for identifying the problem on the LP output)				
Format	I5	I5	F11.6	F11.6			2
Variable	NAA	NAB	POINT	COST			
Format	I5	I5	F11.6	F11.6	F11.6	F11.6	3
Variable	M	N	CURV	RHS	EXP1	EXP2	
Format	F10.4						Next M+2 cards
Variable	C(I)						
Format	F10.4						Next M Groups
Variable	Z(I)						
	Alternate Cards:						
Format	F10.4	F10.4					N cards per group
Variable	X(J)	Y(I,J)					
	or						
Format	F10.4	I5	I5	I5			
Variable	X(J)	IAY	IBY	ICY			

This arrangement can be stated as follows: The first card of each set gives the height z of the waterline. The following cards each give a station coordinate and the offset at the intersection of the given waterline and station. There is one set of cards for each waterline in the surface.

Format	F10.4	F10.4	F10.4	F10.4	Last Card
Variable	SX	SZ	TX	TZ	

(This card is entered only if NAA = -1)

Output

All of the data necessary for the deviation constraints in the matrix are punched by this program. The first cards punched contain the heading, "ROW ID," "OFSET" (this is the identification of the cost row) and "MATRIX."

The matrix elements are punched in row order with the columns in order within the rows. Each row also contains in order of appearance:

(1) The row identification for that row

(2) The right-hand side element for that row

(3) The costs of the two slack columns associated with the row

(4) The slack elements

(5) The basis heading card

This output must be sorted, together with the output from Pass 2, before being used with LP-90 . The sorting instructions are in the section called "Output Assembly."

Sense Switch Settings

Sense Switch 1 - Not Used	
Sense Switch 2 - ON	The values of the x and z second differences and the resulting signs are printed
Sense Switch 2 - OFF	The above is not printed
Sense Switch 3 - ON	The offsets are read in as feet, inches and eighths
Sense Switch 3 - OFF	The offsets are read in as feet and decimals
Sense Switch 4 - ON	The data for Pass 2 is punched
Sense Switch 4 - OFF	The data for Pass 2 is not punched but remains in storage for later use with Pass 2 .

SAMPLE PROBLEM

The following sample problem is a five-station by five-waterline set of offsets taken from the DLG-26 class ships. The stations are Numbers 0 , 1/2 , 1 , 1-1/2 , 2, and the waterlines are 12, 16, 20, 24, and 28. The surface contains the bow profile of the DLG. The matrix is double splined in the z direction. Only the output from Pass 1 in the order in which it was punched is listed here. The output from Pass 2 is included with the instructions from that program. The sorted, assembled, and combined outputs of Pass 1 and 2 is listed after the Pass 2 output.

** SAMPLE INPUT **

```
*     TODD  DLG  5X5 END PROFILE X-Z DBL SPL-12/3/62
   1   -1  1.         0.
   5    5  0.001    1000.      3.          .33333333
 1.52935
- .5873
 0.
 0.
 0.
 0.
 0.
     .0000
     .0000   .000000
  12.7500   1.041666
  25.5000   2.458333
  38.2500   3.874999
  51.0000   5.333333
   1.0000
     .0000   .000000
  12.7500   1.239583
  25.5000   2.697916
  38.2500   4.197916
  51.0000   5.812500
   2.0000
     .0000   .078
  12.7500   1.479166
  25.5000   2.979166
  38.2500   4.583333
  51.0000   6.333333
   3.0000
     .0000   .343749
  12.7500   1.791666
  25.5000   3.447916
  38.2500   5.250000
  51.0000   7.166666
   4.0000
     .0000   .708333
  12.7500   2.416666
  25.5000   4.322916
  38.2500   6.416666
  51.0000   8.583333
```

* TODD DLG 5X5 END PROFILE X-Z DBL SPL-12/3/62

END PROFILE INTERSECTIONS

WLINES

1	1.529350
-.5873	0.0000
2	.942050
-.5873	0.0000
3	.354750
-.5873	0.0000
4	-.232550
-.5873	0.0000
5	-.819850
-.5873	0.0000

STATIONS

1	2.604003
2	.901367
3	0.000000
4	0.000000
5	0.000000

2ND DIFF - WLINES

WL	STA	NORM VALUE	SIGN
1	1	OUT OF SURFACE	
1	2	OUT OF SURFACE	
1	3	-5.176766	1.00
1	4	.041668	-1.00
1	5	.041668	-1.00
2	1	OUT OF SURFACE	
2	2	-37.680848	1.00
2	3	.041667	-1.00
2	4	.114584	-1.00
2	5	.114584	-1.00
3	1	OUT OF SURFACE	
3	2	-.963248	1.00
3	3	.104167	-1.00
3	4	.145833	-1.00
3	5	.145833	-1.00
4	1	-.049093	1.00
4	2	.208333	-1.00
4	3	.145834	-1.00
4	4	.114582	-1.00
4	5	.114582	-1.00

5	1	.927938	-1.00
5	2	.197917	-1.00
5	3	.187500	-1.00
5	4	.072917	-1.00
5	5	.072917	-1.00

2ND DIFF-STATIONS

STA	WL	NORM VALUE	SIGN
1	2	OUT OF SURFACE	
1	3	OUT OF SURFACE	
1	4	-.439121	1.00
1	5	-.439121	1.00
2	2	-3.216759	1.00
2	3	.072917	-1.00
2	4	.312500	-1.00
2	5	.312500	-1.00
3	2	.041667	-1.00
3	3	.187500	-1.00
3	4	.406250	-1.00
3	5	.406250	-1.00
4	2	.062500	-1.00
4	3	.281250	-1.00
4	4	.499999	-1.00
4	5	.499999	-1.00
5	2	.041666	-1.00
5	3	.312500	-1.00
5	4	.583334	-1.00
5	5	.583334	-1.00

1.0.0-2

```
SMOG 3, PASS 1
    *     TODD  DLG  5X5 END PROFILE X-Z DBL SPL-12/3/62
    ROW ID
                OFSET
    MATRIX
                R    3
                R    3      2.45833
          C   1 R    3-     1.0
          C   2 R    3      1.0
          C   3 R    3-     2.00000
          C   4 R    3      2.00000
          C   5 R    3-     4.00000
          C   6 R    3      4.00000
          C   7 R    3-     8.00000
          C   8 R    3      8.00000
          C  51 OFSET    1000.00000
          C  52 OFSET    1000.00000
          C  51 R    3      1.0
          C  52 R    3-     1.0
          C  51/R    3
                R    4
                R    4      3.87499
          C   1 R    4-     1.0
          C   2 R    4      1.0
          C   3 R    4-     3.00000
          C   4 R    4      3.00000
          C   5 R    4-     9.00000
          C   6 R    4      9.00000
          C   7 R    4-    27.00000
          C   8 R    4     27.00000
          C   9 R    4-     1.00000
          C  10 R    4      1.00000
          C  53 OFSET    1000.00000
          C  54 OFSET    1000.00000
          C  53 R    4      1.0
          C  54 R    4-     1.0
          C  53 R    4
                R    5
                R    5      5.33333
          C   1 R    5-     1.0
          C   2 R    5      1.0
          C   3 R    5-     4.00000
```

```
C    4 R    5       4.00000
C    5 R    5-     16.00000
C    6 R    5      16.00000
C    7 R    5-     64.00000
C    8 R    5      64.00000
C    9 R    5-      8.00000
C   10 R    5       8.00000
C   55 OFSET     1000.00000
C   56 OFSET     1000.00000
C   55 R    5       1.0
C   56 R    5-      1.0
C   55 R    5
       R    7
       R    7       1.24702
C    1 R    7-      1.0
C    2 R    7       1.0
C    3 R    7-      1.00000
C    4 R    7       1.00000
C    5 R    7-      1.00000
C    6 R    7       1.00000
C    7 R    7-      1.00000
C    8 R    7       1.00000
C   11 R    7-      1.00000
C   12 R    7       1.00000
C   13 R    7-      1.00000
C   14 R    7       1.00000
C   15 R    7-      1.00000
C   16 R    7       1.00000
C   17 R    7-      1.00000
C   18 R    7       1.00000
C   21 R    7-      1.00000
C   22 R    7       1.00000
C   23 R    7-      1.00000
C   24 R    7       1.00000
C   25 R    7-      1.00000
C   26 R    7       1.00000
C   27 R    7-      1.00000
C   28 R    7       1.00000
C   31 R    7-      1.00000
C   32 R    7       1.00000
C   33 R    7-      1.00000
```

1.0.0-2

```
C  34 R     7        1.00000
C  35 R     7-       1.00000
C  36 R     7        1.00000
C  37 R     7-       1.00000
C  38 R     7        1.00000
C  57 OFSET    1000.00000
C  58 OFSET    1000.00000
C  57 R     7        1.0
C  58 R     7-       1.0
C  57 R     7
        R   8
        R   8        2.69791
C   1 R     8-       1.0
C   2 R     8        1.0
C   3 R     8-       2.00000
C   4 R     8        2.00000
C   5 R     8-       4.00000
C   6 R     8        4.00000
C   7 R     8-       8.00000
C   8 R     8        8.00000
C  11 R     8-       1.00000
C  12 R     8        1.00000
C  13 R     8-       2.00000
C  14 R     8        2.00000
C  15 R     8-       4.00000
C  16 R     8        4.00000
C  17 R     8-       8.00000
C  18 R     8        8.00000
C  21 R     8-       1.00000
C  22 R     8        1.00000
C  23 R     8-       2.00000
C  24 R     8        2.00000
C  25 R     8-       4.00000
C  26 R     8        4.00000
C  27 R     8-       8.00000
C  28 R     8        8.00000
C  31 R     8-       1.00000
C  32 R     8        1.00000
C  33 R     8-       2.00000
C  34 R     8        2.00000
C  35 R     8-       4.00000
```

1.0.0-2

```
C   36 R     8       4.00000
C   37 R     8-      8.00000
C   38 R     8       8.00000
C   59 OFSET      1000.00000
C   60 OFSET     -1000.00000
C   59 R     8       1.0
C   60 R     8-      1.0
C   59 R     8
       R     9
       R     9       4.19791
C    1 R     9-      1.0
C    2 R     9       1.0
C    3 R     9-      3.00000
C    4 R     9       3.00000
C    5 R     9-      9.00000
C    6 R     9       9.00000
C    7 R     9-     27.00000
C    8 R     9      27.00000
C    9 R     9-      1.00000
C   10 R     9       1.00000
C   11 R     9-      1.00000
C   12 R     9       1.00000
C   13 R     9-      3.00000
C   14 R     9       3.00000
C   15 R     9-      9.00000
C   16 R     9       9.00000
C   17 R     9-     27.00000
C   18 R     9      27.00000
C   19 R     9-      1.00000
C   20 R     9       1.00000
C   21 R     9-      1.00000
C   22 R     9       1.00000
C   23 R     9-      3.00000
C   24 R     9       3.00000
C   25 R     9-      9.00000
C   26 R     9       9.00000
C   27 R     9-     27.00000
C   28 R     9      27.00000
C   29 R     9-      1.00000
C   30 R     9       1.00000
C   31 R     9-      1.00000
```

1,0,0-2

```
C  32 R     9       1.00000
C  33 R     9-      3.00000
C  34 R     9       3.00000
C  35 R     9-      9.00000
C  36 R     9       9.00000
C  37 R     9-     27.00000
C  38 R     9      27.00000
C  39 R     9-      1.00000
C  40 R     9       1.00000
C  61 OFSET      1000.00000
C  62 OFSET      1000.00000
C  61 R     9       1.0
C  62 R     9-      1.0
C  61 R     9
      R    10
      R    10       5.81250
C   1 R    10-      1.0
C   2 R    10       1.0
C   3 R    10-      4.00000
C   4 R    10       4.00000
C   5 R    10-     16.00000
C   6 R    10      16.00000
C   7 R    10-     64.00000
C   8 R    10      64.00000
C   9 R    10-      8.00000
C  10 R    10       8.00000
C  11 R    10-      1.00000
C  12 R    10       1.00000
C  13 R    10-      4.00000
C  14 R    10       4.00000
C  15 R    10-     16.00000
C  16 R    10      16.00000
C  17 R    10-     64.00000
C  18 R    10      64.00000
C  19 R    10-      8.00000
C  20 R    10       8.00000
C  21 R    10-      1.00000
C  22 R    10       1.00000
C  23 R    10-      4.00000
C  24 R    10       4.00000
C  25 R    10-     16.00000
```

```
C  26 R   10      16.00000
C  27 R   10-     64.00000
C  28 R   10      64.00000
C  29 R   10-      8.00000
C  30 R   10       8.00000
C  31 R   10-      1.00000
C  32 R   10       1.00000
C  33 R   10-      4.00000
C  34 R   10       4.00000
C  35 R   10-     16.00000
C  36 R   10      16.00000
C  37 R   10-     64.00000
C  38 R   10      64.00000
C  39 R   10-      8.00000
C  40 R   10       8.00000
C  63 OFSET     1000.00000
C  64 OFSET     1000.00000
C  63 R   10       1.0
C  64 R   10-      1.0
C  63 R   10
      R   12
      R   12       1.47916
C   1 R   12-      1.0
C   2 R   12       1.0
C   3 R   12-      1.00000
C   4 R   12       1.00000
C   5 R   12-      1.00000
C   6 R   12       1.00000
C   7 R   12-      1.00000
C   8 R   12       1.00000
C  11 R   12-      2.00000
C  12 R   12       2.00000
C  13 R   12-      2.00000
C  14 R   12       2.00000
C  15 R   12-      2.00000
C  16 R   12       2.00000
C  17 R   12-      2.00000
C  18 R   12       2.00000
C  21 R   12-      4.00000
C  22 R   12       4.00000
C  23 R   12-      4.00000
```

```
C  24 R    12        4.00000
C  25 R    12-       4.00000
C  26 R    12        4.00000
C  27 R    12-       4.00000
C  28 R    12        4.00000
C  31 R    12-       8.00000
C  32 R    12        8.00000
C  33 R    12-       8.00000
C  34 R    12        8.00000
C  35 R    12-       8.00000
C  36 R    12        8.00000
C  37 R    12-       8.00000
C  38 R    12        8.00000
C  65 OFSET       1000.00000
C  66 OFSET       1000.00000
C  65 R    12        1.0
C  66 R    12-       1.0
C  65 R    12
      R    13
      R    13        2.97916
C   1 R    13-       1.0
C   2 R    13        1.0
C   3 R    13-       2.00000
C   4 R    13        2.00000
C   5 R    13-       4.00000
C   6 R    13        4.00000
C   7 R    13-       8.00000
C   8 R    13        8.00000
C  11 R    13-       2.00000
C  12 R    13        2.00000
C  13 R    13-       4.00000
C  14 R    13        4.00000
C  15 R    13-       8.00000
C  16 R    13        8.00000
C  17 R    13-      16.00000
C  18 R    13       16.00000
C  21 R    13-       4.00000
C  22 R    13        4.00000
C  23 R    13-       8.00000
C  24 R    13        8.00000
C  25 R    13-      16.00000
```

1.0.0-2

```
C   26 R    13      16.00000
C   27 R    13-     32.00000
C   28 R    13      32.00000
C   31 R    13-      8.00000
C   32 R    13       8.00000
C   33 R    13-     16.00000
C   34 R    13      16.00000
C   35 R    13-     32.00000
C   36 R    13      32.00000
C   37 R    13-     64.00000
C   38 R    13      64.00000
C   67 OFSET      1000.00000
C   68 OFSET      1000.00000
C   67 R    13       1.0
C   68 R    13-      1.0
C   67 R    13
       R    14
       R    14       4.58333
C    1 R    14-      1.0
C    2 R    14       1.0
C    3 R    14-      3.00000
C    4 R    14       3.00000
C    5 R    14-      9.00000
C    6 R    14       9.00000
C    7 R    14-     27.00000
C    8 R    14      27.00000
C    9 R    14-      1.00000
C   10 R    14       1.00000
C   11 R    14-      2.00000
C   12 R    14       2.00000
C   13 R    14-      6.00000
C   14 R    14       6.00000
C   15 R    14-     18.00000
C   16 R    14      18.00000
C   17 R    14-     54.00000
C   18 R    14      54.00000
C   19 R    14-      2.00000
C   20 R    14       2.00000
C   21 R    14-      4.00000
C   22 R    14       4.00000
C   23 R    14-     12.00000
```

1.0.0.2

```
C   24 R    14      12.00000
C   25 R    14-     36.00000
C   26 R    14      36.00000
C   27 R    14-    108.00000
C   28 R    14     108.00000
C   29 R    14-      4.00000
C   30 R    14       4.00000
C   31 R    14-      8.00000
C   32 R    14       8.00000
C   33 R    14-     24.00000
C   34 R    14      24.00000
C   35 R    14-     72.00000
C   36 R    14      72.00000
C   37 R    14-    216.00000
C   38 R    14     216.00000
C   39 R    14-      8.00000
C   40 R    14       8.00000
C   69 OFSET      1000.00000
C   70 OFSET      1000.00000
C   69 R    14       1.0
C   70 R    14-      1.0
C   69 R    14
       R    15
       R    15       6.33333
C    1 R    15-      1.0
C    2 R    15       1.0
C    3 R    15-      4.00000
C    4 R    15       4.00000
C    5 R    15-     16.00000
C    6 R    15      16.00000
C    7 R    15-     64.00000
C    8 R    15      64.00000
C    9 R    15-      8.00000
C   10 R    15       8.00000
C   11 R    15-      2.00000
C   12 R    15       2.00000
C   13 R    15-      8.00000
C   14 R    15       8.00000
C   15 R    15-     32.00000
C   16 R    15      32.00000
C   17 R    15-    128.00000
```

```
C   18 R   15    128.00000
C   19 R   15-    16.00000
C   20 R   15     16.00000
C   21 R   15-     4.00000
C   22 R   15      4.00000
C   23 R   15-    16.00000
C   24 R   15     16.00000
C   25 R   15-    64.00000
C   26 R   15     64.00000
C   27 R   15-   256.00000
C   28 R   15    256.00000
C   29 R   15-    32.00000
C   30 R   15     32.00000
C   31 R   15-     8.00000
C   32 R   15      8.00000
C   33 R   15-    32.00000
C   34 R   15     32.00000
C   35 R   15-   128.00000
C   36 R   15    128.00000
C   37 R   15-   512.00000
C   38 R   15    512.00000
C   39 R   15-    64.00000
C   40 R   15     64.00000
C   71 OFSET    1000.00000
C   72 OFSET    1000.00000
C   71 R   15      1.0
C   72 R   15-     1.0
C   71 R   15
       R   16
       R   16        .34374
C    1 R   16-     1.0
C    2 R   16      1.0
C   11 R   16-     3.00000
C   12 R   16      3.00000
C   21 R   16-     9.00000
C   22 R   16      9.00000
C   31 R   16-    27.00000
C   32 R   16     27.00000
C   41 R   16-     1.00000
C   42 R   16      1.00000
C   73 OFSET    1000.00000
```

1.0.0-2

1.0.0-2

```
C   74 OFSET  1000.00000
C   73 R   16     1.0
C   74 R   16-    1.0
C   73 R   16
       R   17
       R   17     1.79166
C    1 R   17-    1.0
C    2 R   17     1.0
C    3 R   17-    1.00000
C    4 R   17     1.00000
C    5 R   17-    1.00000
C    6 R   17     1.00000
C    7 R   17-    1.00000
C    8 R   17     1.00000
C   11 R   17-    3.00000
C   12 R   17     3.00000
C   13 R   17-    3.00000
C   14 R   17     3.00000
C   15 R   17-    3.00000
C   16 R   17     3.00000
C   17 R   17-    3.00000
C   18 R   17     3.00000
C   21 R   17-    9.00000
C   22 R   17     9.00000
C   23 R   17-    9.00000
C   24 R   17     9.00000
C   25 R   17-    9.00000
C   26 R   17     9.00000
C   27 R   17-    9.00000
C   28 R   17     9.00000
C   31 R   17-   27.00000
C   32 R   17    27.00000
C   33 R   17-   27.00000
C   34 R   17    27.00000
C   35 R   17-   27.00000
C   36 R   17    27.00000
C   37 R   17-   27.00000
C   38 R   17    27.00000
C   41 R   17-    1.00000
C   42 R   17     1.00000
C   43 R   17-    1.00000
```

```
C  44 R    17        1.00000
C  45 R    17-       1.00000
C  46 R    17        1.00000
C  47 R    17-       1.00000
C  48 R    17        1.00000
C  75 OFSET     1000.00000
C  76 OFSET     1000.00000
C  75 R    17        1.0
C  76 R    17-       1.0
C  75 R    17
      R    18
      R    18        3.44791
C   1 R    18-       1.0
C   2 R    18        1.0
C   3 R    18-       2.00000
C   4 R    18        2.00000
C   5 R    18-       4.00000
C   6 R    18        4.00000
C   7 R    18-       8.00000
C   8 R    18        8.00000
C  11 R    18-       3.00000
C  12 R    18        3.00000
C  13 R    18-       6.00000
C  14 R    18        6.00000
C  15 R    18-      12.00000
C  16 R    18       12.00000
C  17 R    18-      24.00000
C  18 R    18       24.00000
C  21 R    18-       9.00000
C  22 R    18        9.00000
C  23 R    18-      18.00000
C  24 R    18       18.00000
C  25 R    18-      36.00000
C  26 R    18       36.00000
C  27 R    18-      72.00000
C  28 R    18       72.00000
C  31 R    18-      27.00000
C  32 R    18       27.00000
C  33 R    18-      54.00000
C  34 R    18       54.00000
C  35 R    18-     108.00000
```

1.0.6-2

```
C   24 R   19     27.00000
C   25 R   19-    81.00000
C   26 R   19     81.00000
C   27 R   19-   243.00000
C   28 R   19    243.00000
C   29 R   19-     9.00000
C   30 R   19      9.00000
C   31 R   19-    27.00000
C   32 R   19     27.00000
C   33 R   19-    81.00000
C   34 R   19     81.00000
C   35 R   19-   243.00000
C   36 R   19    243.00000
C   37 R   19-   729.00000
C   38 R   19    729.00000
C   39 R   19-    27.00000
C   40 R   19     27.00000
C   41 R   19-     1.00000
C   42 R   19      1.00000
C   43 R   19-     3.00000
C   44 R   19      3.00000
C   45 R   19-     9.00000
C   46 R   19      9.00000
C   47 R   19-    27.00000
C   48 R   19     27.00000
C   49 R   19-     1.00000
C   50 R   19      1.00000
C   79 OFSET    1000.00000
C   80 OFSET    1000.00000
C   79 R   19      1.0
C   80 R   19-     1.0
C   79 R   19
       R   20
       R   20      7.16666
C    1 R   20-     1.0
C    2 R   20      1.0
C    3 R   20-     4.00000
C    4 R   20      4.00000
C    5 R   20-    16.00000
C    6 R   20     16.00000
C    7 R   20-    64.00000
```

1.0.0-2

```
C  36 R    18    108.00000
C  37 R    18-   216.00000
C  38 R    18    216.00000
C  41 R    18-     1.00000
C  42 R    18      1.00000
C  43 R    18-     2.00000
C  44 R    18      2.00000
C  45 R    18-     4.00000
C  46 R    18      4.00000
C  47 R    18-     8.00000
C  48 R    18      8.00000
C  77 OFSET    1000.00000
C  78 OFSET    1000.00000
C  77 R    18      1.0
C  78 R    18-     1.0
C  77 R    18
        R  19
        R  19      5.25000
C   1 R    19-     1.0
C   2 R    19      1.0
C   3 R    19-     3.00000
C   4 R    19      3.00000
C   5 R    19-     9.00000
C   6 R    19      9.00000
C   7 R    19-    27.00000
C   8 R    19     27.00000
C   9 R    19-     1.00000
C  10 R    19      1.00000
C  11 R    19-     3.00000
C  12 R    19      3.00000
C  13 R    19-     9.00000
C  14 R    19      9.00000
C  15 R    19-    27.00000
C  16 R    19     27.00000
C  17 R    19-    81.00000
C  18 R    19     81.00000
C  19 R    19-     3.00000
C  20 R    19      3.00000
C  21 R    19-     9.00000
C  22 R    19      9.00000
C  23 R    19-    27.00000
```

1.0.0.2

C	8	R	20	64.00000
C	9	R	20-	8.00000
C	10	R	20	8.00000
C	11	R	20-	3.00000
C	12	R	20	3.00000
C	13	R	20-	12.00000
C	14	R	20	12.00000
C	15	R	20-	48.00000
C	16	R	20	48.00000
C	17	R	20-	192.00000
C	18	R	20	192.00000
C	19	R	20-	24.00000
C	20	R	20	24.00000
C	21	R	20-	9.00000
C	22	R	20	9.00000
C	23	R	20-	36.00000
C	24	R	20	36.00000
C	25	R	20-	144.00000
C	26	R	20	144.00000
C	27	R	20-	576.00000
C	28	R	20	576.00000
C	29	R	20-	72.00000
C	30	R	20	72.00000
C	31	R	20-	27.00000
C	32	R	20	27.00000
C	33	R	20-	108.00000
C	34	R	20	108.00000
C	35	R	20-	432.00000
C	36	R	20	432.00000
C	37	R	20-	1728.00000
C	38	R	20	1728.00000
C	39	R	20-	216.00000
C	40	R	20	216.00000
C	41	R	20-	1.00000
C	42	R	20	1.00000
C	43	R	20-	4.00000
C	44	R	20	4.00000
C	45	R	20-	16.00000
C	46	R	20	16.00000
C	47	R	20-	64.00000
C	48	R	20	64.00000

```
C  49 R   20-      8.00000
C  50 R   20       8.00000
C  81 OFSET   1000.00000
C  82 OFSET   1000.00000
C  81 R   20       1.0
C  82 R   20-      1.0
C  81 R   20
      R   21
      R   21        .70833
C   1 R   21-      1.0
C   2 R   21       1.0
C  11 R   21-      4.00000
C  12 R   21       4.00000
C  21 R   21-     16.00000
C  22 R   21      16.00000
C  31 R   21-     64.00000
C  32 R   21      64.00000
C  41 R   21-      8.00000
C  42 R   21       8.00000
C  83 OFSET   1000.00000
C  84 OFSET   1000.00000
C  83 R   21       1.0
C  84 R   21-      1.0
C  83 R   21
      R   22
      R   22       2.41666
C   1 R   22-      1.0
C   2 R   22       1.0
C   3 R   22-      1.00000
C   4 R   22       1.00000
C   5 R   22-      1.00000
C   6 R   22       1.00000
C   7 R   22-      1.00000
C   8 R   22       1.00000
C  11 R   22-      4.00000
C  12 R   22       4.00000
C  13 R   22-      4.00000
C  14 R   22       4.00000
C  15 R   22-      4.00000
C  16 R   22       4.00000
C  17 R   22-      4.00000
```

1.0.0-2

```
C   18 R   22      4.00000
C   21 R   22-    16.00000
C   22 R   22     16.00000
C   23 R   22-    16.00000
C   24 R   22     16.00000
C   25 R   22-    16.00000
C   26 R   22     16.00000
C   27 R   22-    16.00000
C   28 R   22     16.00000
C   31 R   22-    64.00000
C   32 R   22     64.00000
C   33 R   22-    64.00000
C   34 R   22     64.00000
C   35 R   22-    64.00000
C   36 R   22     64.00000
C   37 R   22-    64.00000
C   38 R   22     64.00000
C   41 R   22-     8.00000
C   42 R   22      8.00000
C   43 R   22-     8.00000
C   44 R   22      8.00000
C   45 R   22-     8.00000
C   46 R   22      8.00000
C   47 R   22-     8.00000
C   48 R   22      8.00000
C   85 OFSET    1000.00000
C   86 OFSET    1000.00000
C   85 R   22      1.0
C   86 R   22-     1.0
C   85 R   22
       R   23
       R   23      4.32291
C    1 R   23-     1.0
C    2 R   23      1.0
C    3 R   23-     2.00000
C    4 R   23      2.00000
C    5 R   23-     4.00000
C    6 R   23      4.00000
C    7 R   23-     8.00000
C    8 R   23      8.00000
C   11 R   23-     4.00000
```

```
C   12 R   23      4.00000
C   13 R   23-     8.00000
C   14 R   23      8.00000
C   15 R   23-    16.00000
C   16 R   23     16.00000
C   17 R   23-    32.00000
C   18 R   23     32.00000
C   21 R   23-    16.00000
C   22 R   23     16.00000
C   23 R   23-    32.00000
C   24 R   23     32.00000
C   25 R   23-    64.00000
C   26 R   23     64.00000
C   27 R   23-   128.00000
C   28 R   23    128.00000
C   31 R   23-    64.00000
C   32 R   23     64.00000
C   33 R   23-   128.00000
C   34 R   23    128.00000
C   35 R   23-   256.00000
C   36 R   23    256.00000
C   37 R   23-   512.00000
C   38 R   23    512.00000
C   41 R   23-     8.00000
C   42 R   23      8.00000
C   43 R   23-    16.00000
C   44 R   23     16.00000
C   45 R   23-    32.00000
C   46 R   23     32.00000
C   47 R   23-    64.00000
C   48 R   23     64.00000
C   87 OFSET    1000.00000
C   88 OFSET    1000.00000
C   87 R   23      1.0
C   88 R   23-     1.0
C   87 R   23
       R   24
       R   24      6.41666
C    1 R   24-     1.0
C    2 R   24      1.0
C    3 R   24-     3.00000
```

1.0.0-2

1.0.0.2

C	4	R	24	3.00000
C	5	R	24-	9.00000
C	6	R	24	9.00000
C	7	R	24-	27.00000
C	8	R	24	27.00000
C	9	R	24-	1.00000
C	10	R	24	1.00000
C	11	R	24-	4.00000
C	12	R	24	4.00000
C	13	R	24-	12.00000
C	14	R	24	12.00000
C	15	R	24-	36.00000
C	16	R	24	36.00000
C	17	R	24-	108.00000
C	18	R	24	108.00000
C	19	R	24-	4.00000
C	20	R	24	4.00000
C	21	R	24-	16.00000
C	22	R	24	16.00000
C	23	R	24-	48.00000
C	24	R	24	48.00000
C	25	R	24-	144.00000
C	26	R	24	144.00000
C	27	R	24-	432.00000
C	28	R	24	432.00000
C	29	R	24-	16.00000
C	30	R	24	16.00000
C	31	R	24-	64.00000
C	32	R	24	64.00000
C	33	R	24-	192.00000
C	34	R	24	192.00000
C	35	R	24-	576.00000
C	36	R	24	576.00000
C	37	R	24-	1728.00000
C	38	R	24	1728.00000
C	39	R	24-	64.00000
C	40	R	24	64.00000
C	41	R	24-	8.00000
C	42	R	24	8.00000
C	43	R	24-	24.00000
C	44	R	24	24.00000

1.0.0-2

```
C  45 R  24-    72.00000
C  46 R  24     72.00000
C  47 R  24-   216.00000
C  48 R  24    216.00000
C  49 R  24-     8.00000
C  50 R  24      8.00000
C  89 OFSET   1000.00000
C  90 OFSET   1000.00000
C  89 R  24      1.0
C  90 R  24-     1.0
C  89 R  24
      R  25
      R  25      8.58333
C   1 R  25-     1.0
C   2 R  25      1.0
C   3 R  25-     4.00000
C   4 R  25      4.00000
C   5 R  25-    16.00000
C   6 R  25     16.00000
C   7 R  25-    64.00000
C   8 R  25     64.00000
C   9 R  25-     8.00000
C  10 R  25      8.00000
C  11 R  25-     4.00000
C  12 R  25      4.00000
C  13 R  25-    16.00000
C  14 R  25     16.00000
C  15 R  25-    64.00000
C  16 R  25     64.00000
C  17 R  25-   256.00000
C  18 R  25    256.00000
C  19 R  25-    32.00000
C  20 R  25     32.00000
C  21 R  25-    16.00000
C  22 R  25     16.00000
C  23 R  25-    64.00000
C  24 R  25     64.00000
C  25 R  25-   256.00000
C  26 R  25    256.00000
C  27 R  25-  1024.00000
C  28 R  25   1024.00000
```

```
C   29 R   25-   128.00000
C   30 R   25    128.00000
C   31 R   25-    64.00000
C   32 R   25     64.00000
C   33 R   25-   256.00000
C   34 R   25    256.00000
C   35 R   25-  1024.00000
C   36 R   25   1024.00000
C   37 R   25-  4096.00000
C   38 R   25   4096.00000
C   39 R   25-   512.00000
C   40 R   25    512.00000
C   41 R   25-     8.00000
C   42 R   25      8.00000
C   43 R   25-    32.00000
C   44 R   25     32.00000
C   45 R   25-   128.00000
C   46 R   25    128.00000
C   47 R   25-   512.00000
C   48 R   25    512.00000
C   49 R   25-    64.00000
C   50 R   25     64.00000
C   91 OFSET    1000.00000
C   92 OFSET    1000.00000
C   91 R   25      1.0
C   92 R   25-     1.0
C   91 R   25
C    1 OFSET       0.00100
C    2 OFSET       0.00100
C    3 OFSET       0.00100
C    4 OFSET       0.00100
C    5 OFSET       0.00100
C    6 OFSET       0.00100
C    7 OFSET       0.00100
C    8 OFSET       0.00100
C    9 OFSET       0.00100
C   10 OFSET       0.00100
C   11 OFSET       0.00100
C   12 OFSET       0.00100
C   13 OFSET       0.00100
C   14 OFSET       0.00100
```

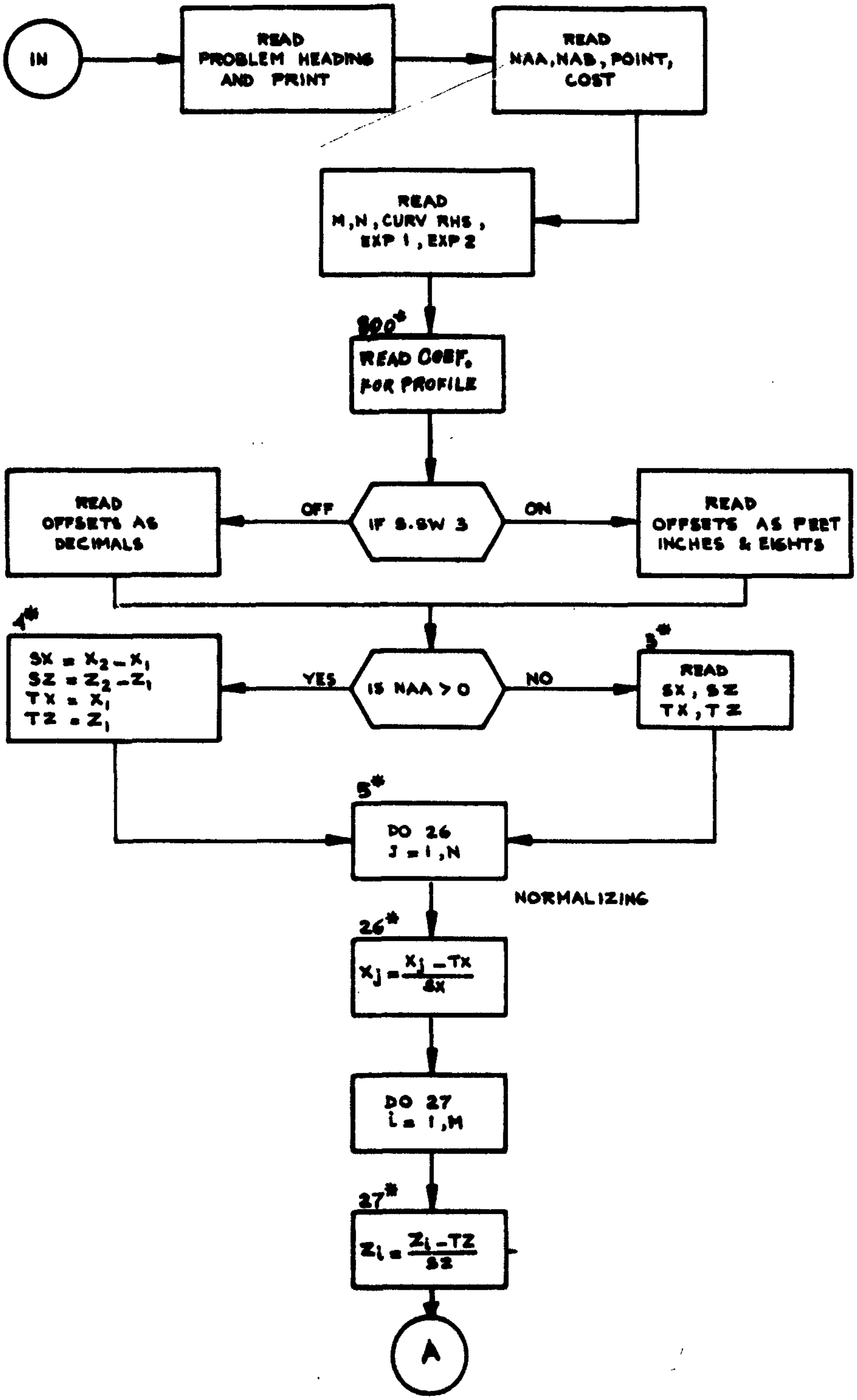
1.0.0-2

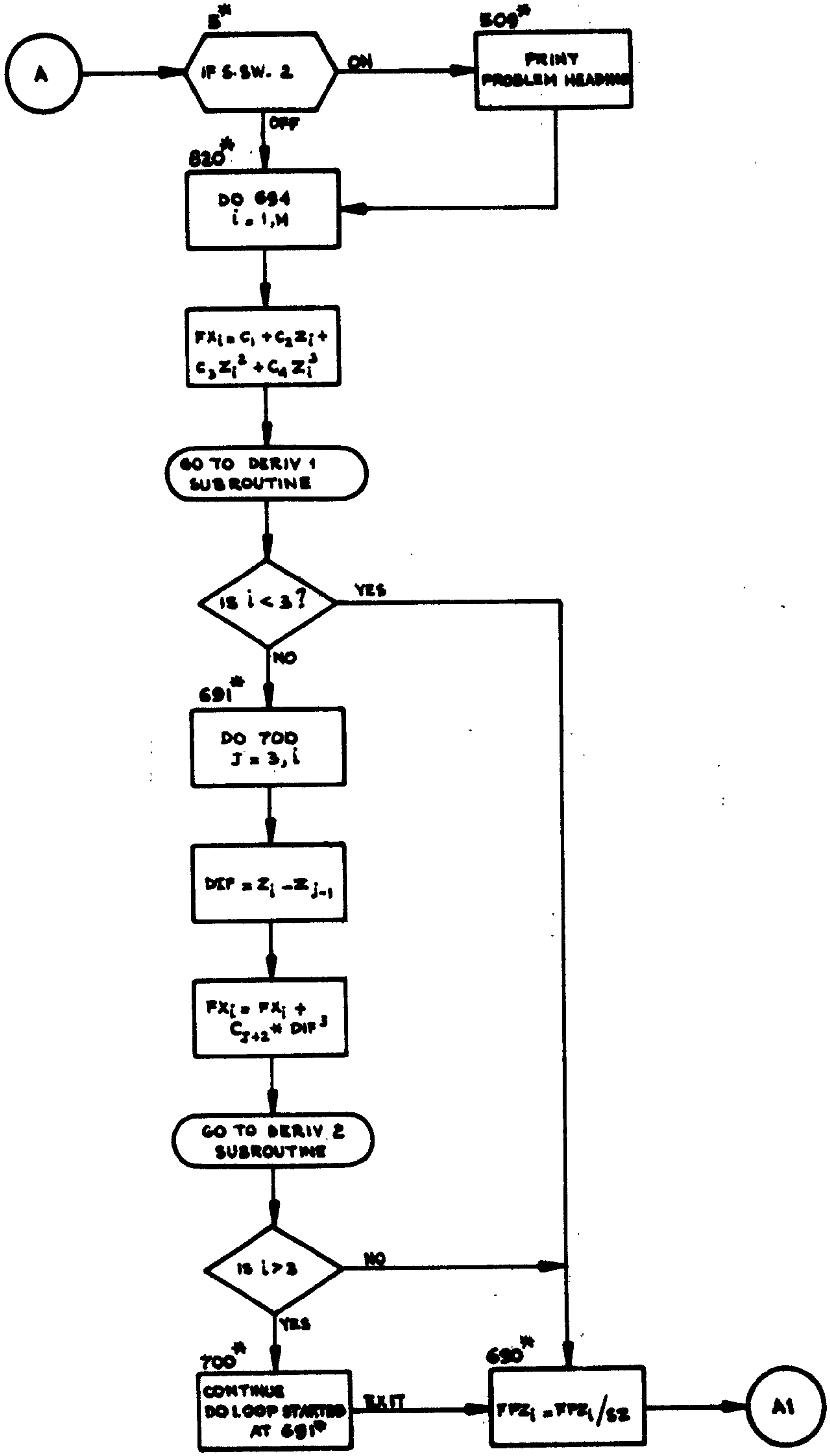
1.0.0-2

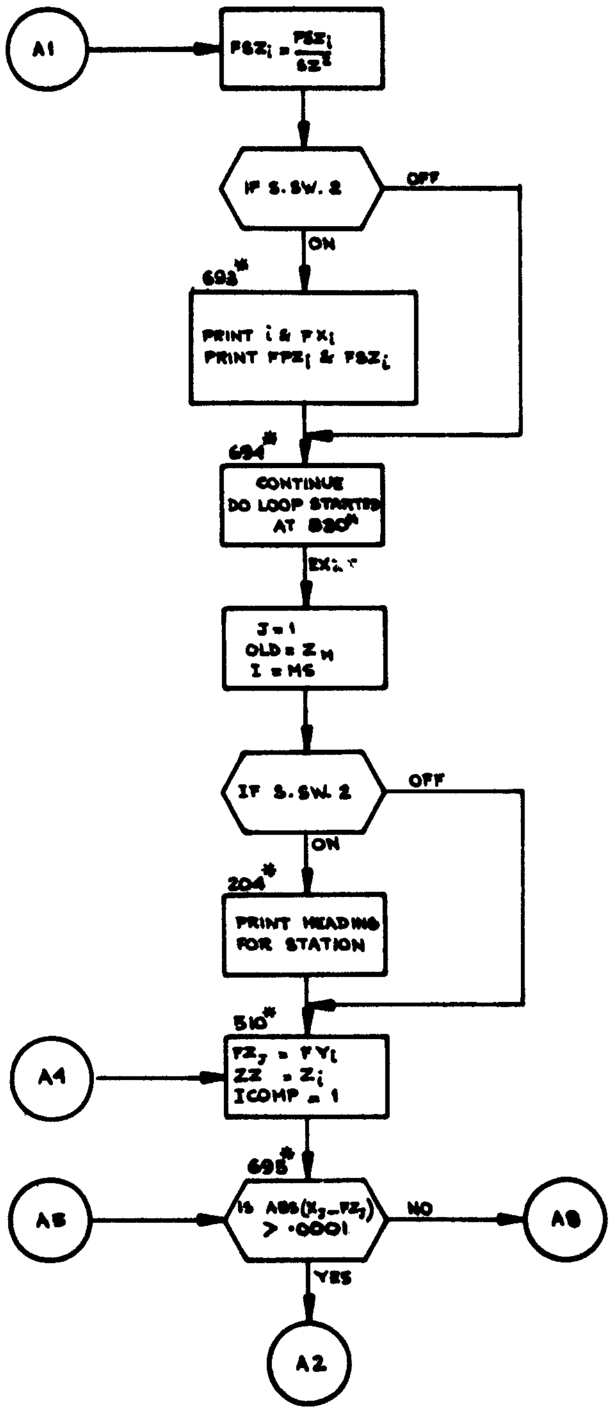
```
C   15 OFSET      0.00100
C   16 OFSET      0.00100
C   17 OFSET      0.00100
C   18 OFSET      0.00100
C   19 OFSET      0.00100
C   20 OFSET      0.00100
C   21 OFSET      0.00100
C   22 OFSET      0.00100
C   23 OFSET      0.00100
C   24 OFSET      0.00100
C   25 OFSET      0.00100
C   26 OFSET      0.00100
C   27 OFSET      0.00100
C   28 OFSET      0.00100
C   29 OFSET      0.00100
C   30 OFSET      0.00100
C   31 OFSET      0.00100
C   32 OFSET      0.00100
C   33 OFSET      0.00100
C   34 OFSET      0.00100
C   35 OFSET      0.00100
C   36 OFSET      0.00100
C   37 OFSET      0.00100
C   38 OFSET      0.00100
C   39 OFSET      0.00100
C   40 OFSET      0.00100
C   41 OFSET      0.00100
C   42 OFSET      0.00100
C   43 OFSET      0.00100
C   44 OFSET      0.00100
C   45 OFSET      0.00100
C   46 OFSET      0.00100
C   47 OFSET      0.00100
C   48 OFSET      0.00100
C   49 OFSET      0.00100
C   50 OFSET      0.00100
```

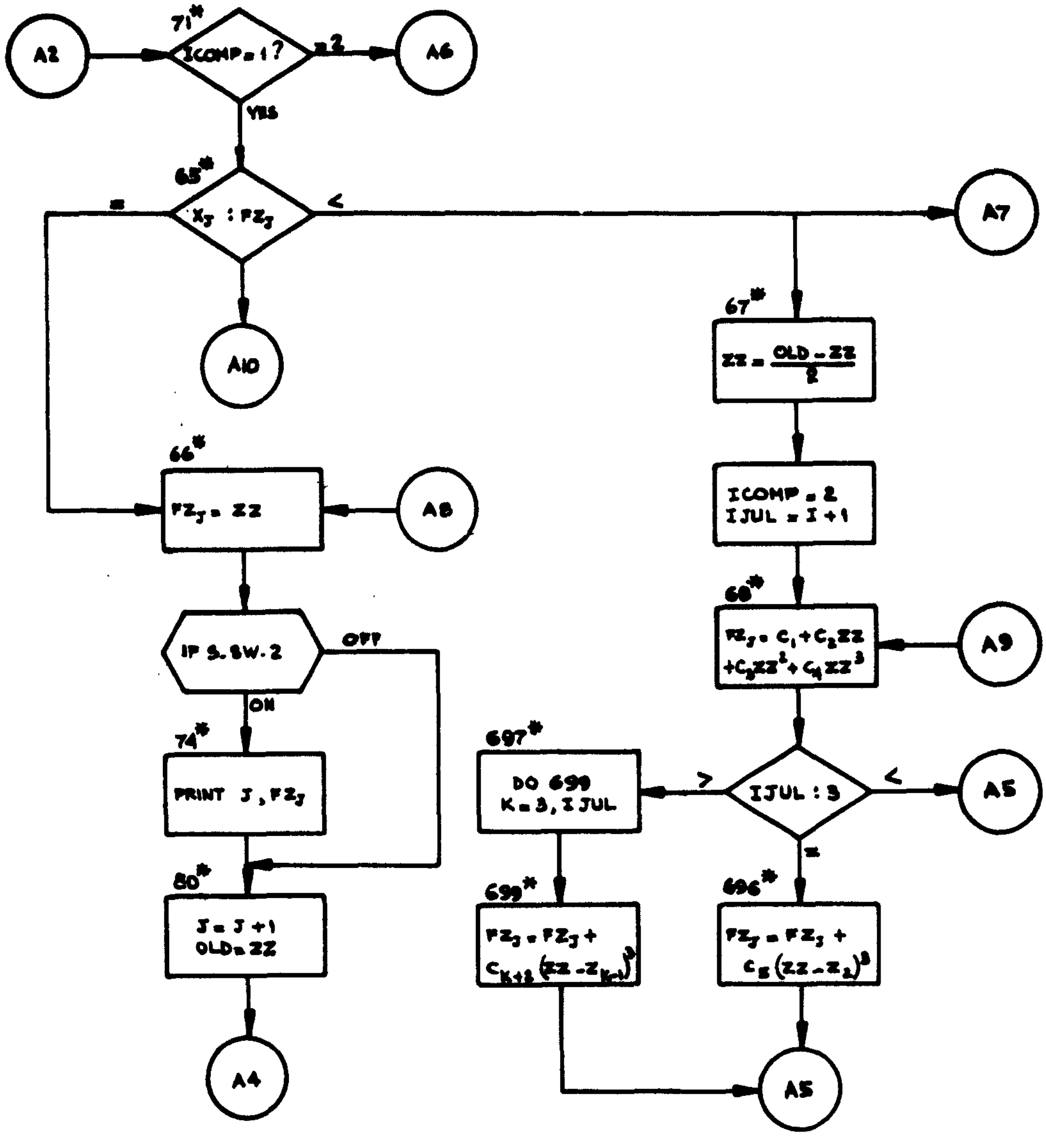
B-225

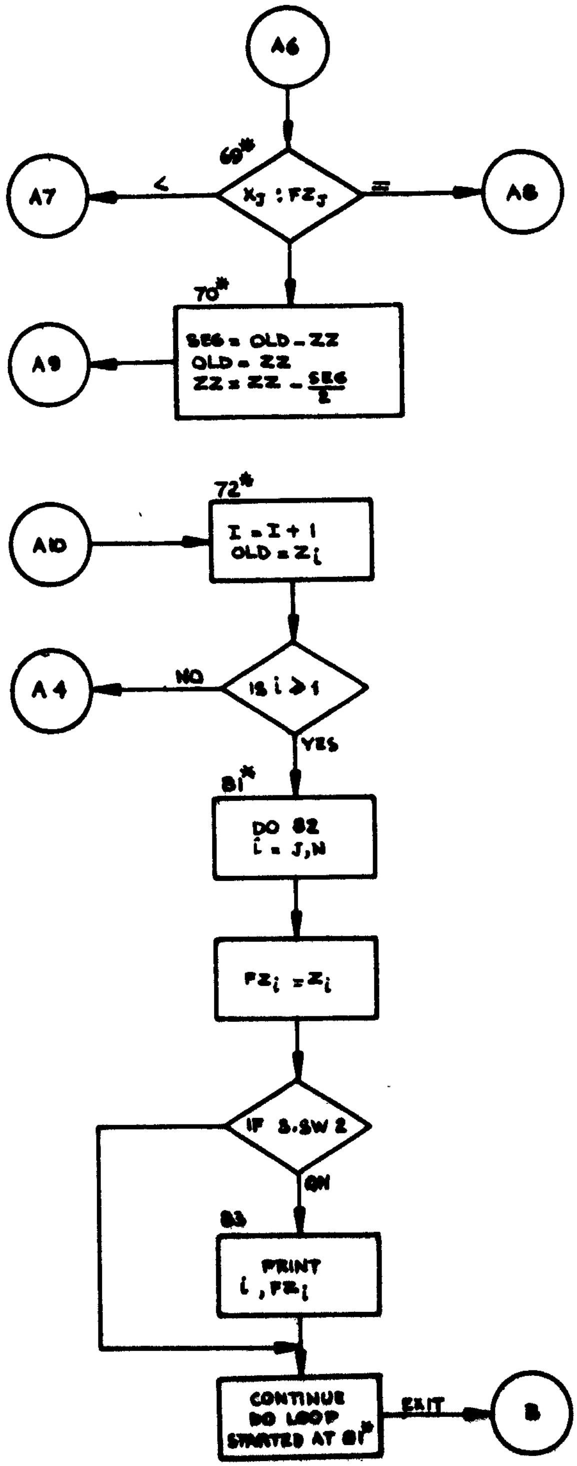
FLOW DIAGRAM - SMOG 3, PASS 1 (See SMOG 2, Pass 1 for flow diagrams of the subroutines)

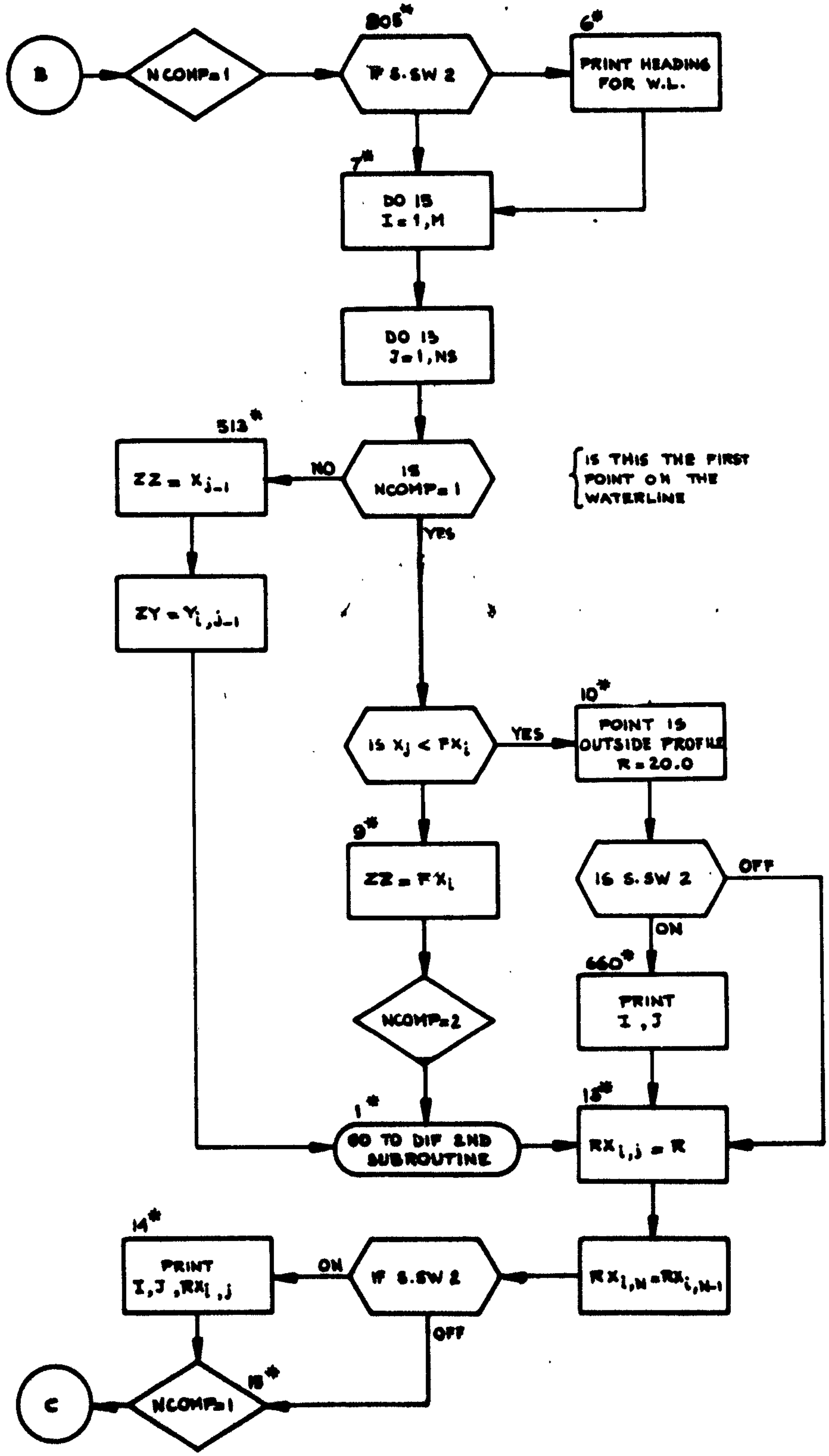


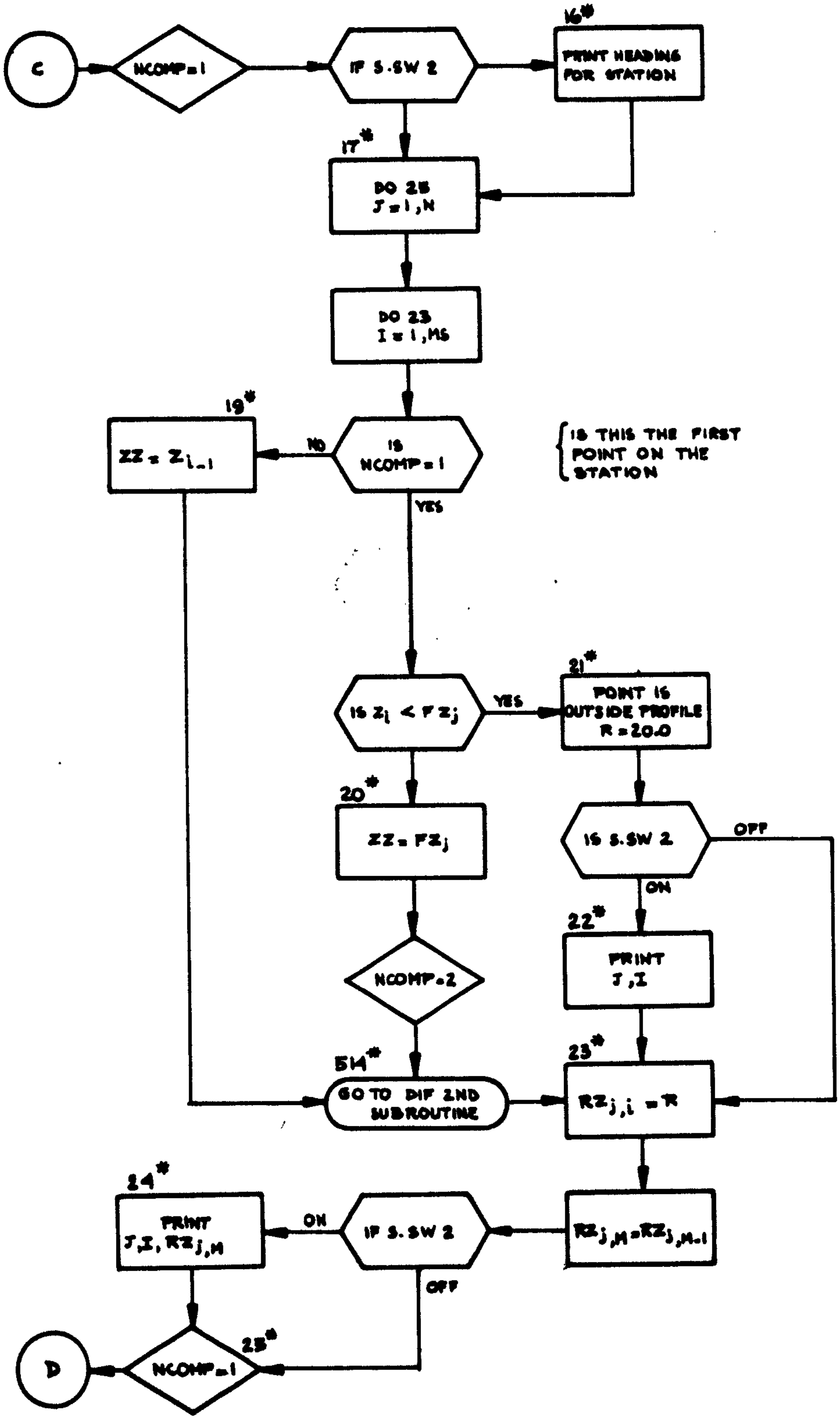

A
5*
IF S.SW. 2
ON
OFF
509*
PRINT PROBLEM HEADING
820*
DO 694
i = 1,M
FX_i = C_1 + C_2 Z_i + C_3 Z_i^2 + C_4 Z_i^3
GO TO DERIV 1 SUBROUTINE
IS i < 3 ?
YES
NO
691*
DO 700
J = 3, i
DIF = Z_i - Z_j-1
FX_i = FX_i + C_J+2 * DIF^3
GO TO DERIV 2 SUBROUTINE
IS i > 3
NO
YES
700*
CONTINUE DO LOOP STARTED AT 691*
EXIT
690*
FFZ_i = FFZ_i / S2
A1

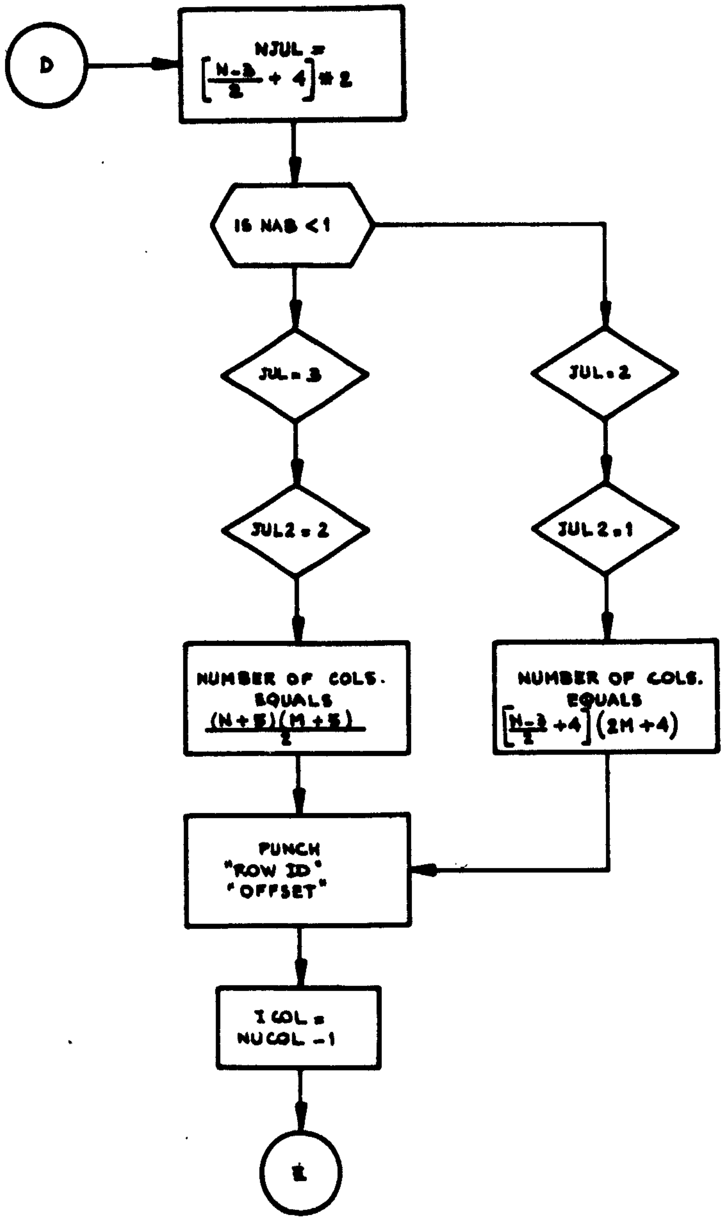

A1
FSZi = FSZi / SZ²
IF S.SW. 2
OFF
ON
693*
PRINT i & FXi
PRINT FPZi & FSZi
694*
CONTINUE
DO LOOP STARTED
AT 530*
EXIT
J = 1
OLD = ZM
I = MS
IF S.SW. 2
OFF
ON
204*
PRINT HEADING
FOR STATION
510*
A4
FZJ = FYi
ZZ = Zi
ICOMP = 1
695*
A5
IS ABS(XJ - FZJ)
> .0001
NO
A9
YES
A2

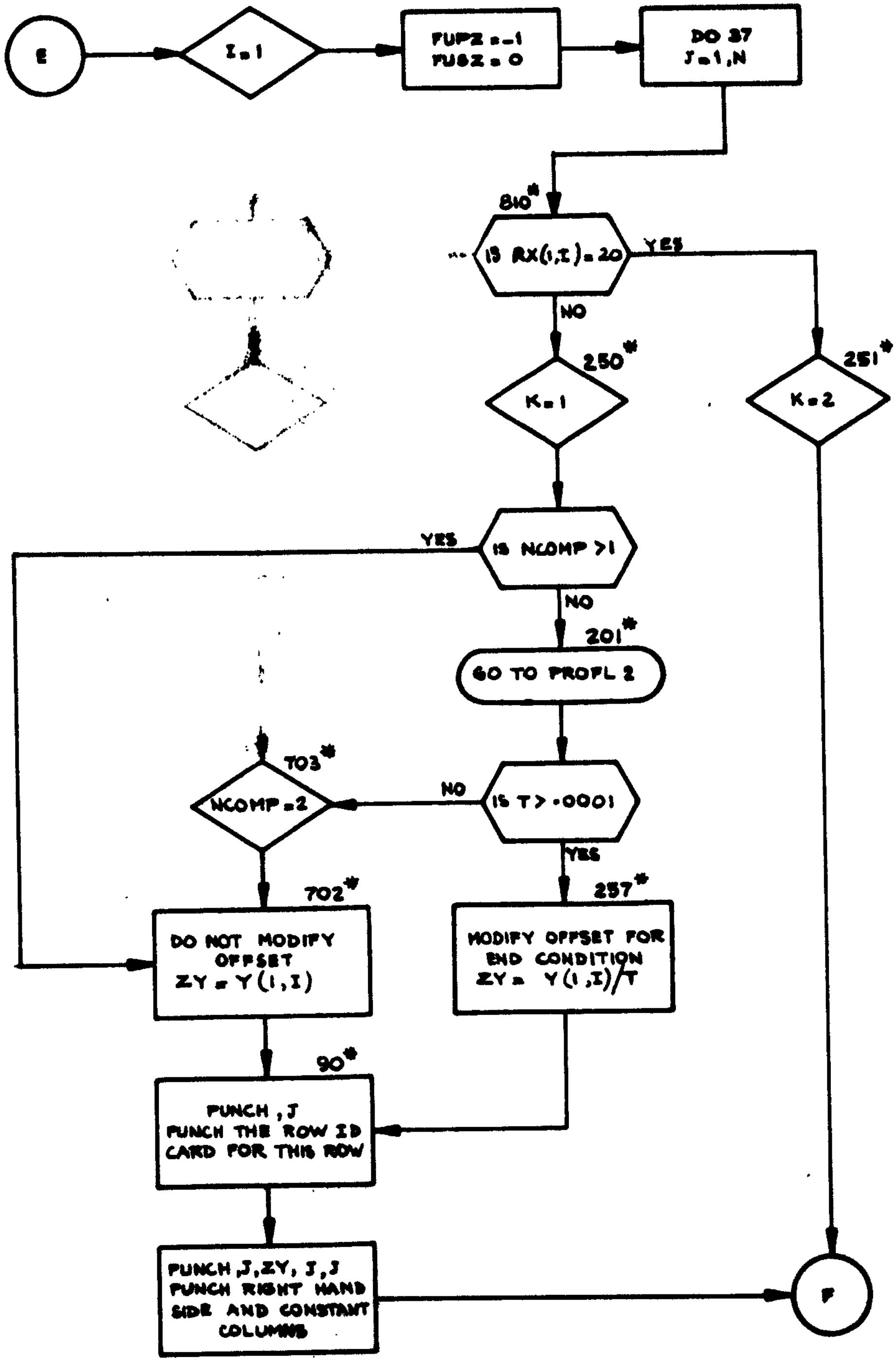

A2
71*
ICOMP = 1?
= 2
A6
YES
65*
XJ : FZJ
=
<
A7
A10
67*
ZZ = OLD - ZZ / 2
66*
FZJ = ZZ
A8
ICOMP = 2
IJUL = I + 1
68*
FZJ = C1 + C2ZZ + C3ZZ2 + C4ZZ3
A9
IF S.SW.2
OFF
ON
74*
PRINT J, FZJ
697*
DO 699
K = 3, IJUL
>
IJUL : 3
<
A5
=
80*
J = J + 1
OLD = ZZ
699*
FZJ = FZJ + CK+2 (ZZ - ZK-1)3
696*
FZJ = FZJ + C5 (ZZ - Z2)3
A4
A5


A6
69*
$X_J : FZ_J$
<
=
A7
A8
70*
SEG = OLD - ZZ
OLD = ZZ
ZZ = ZZ - SEG/2
A9
72*
I = I + 1
OLD = Z_i
A10
IS i > 1
NO
A 4
YES
81*
DO 82
i = J,N
$FZ_i = Z_i$
IF S.SW 2
ON
83
PRINT
i , FZ_i
CONTINUE
DO LOOP
STARTED AT 81*
EXIT
B

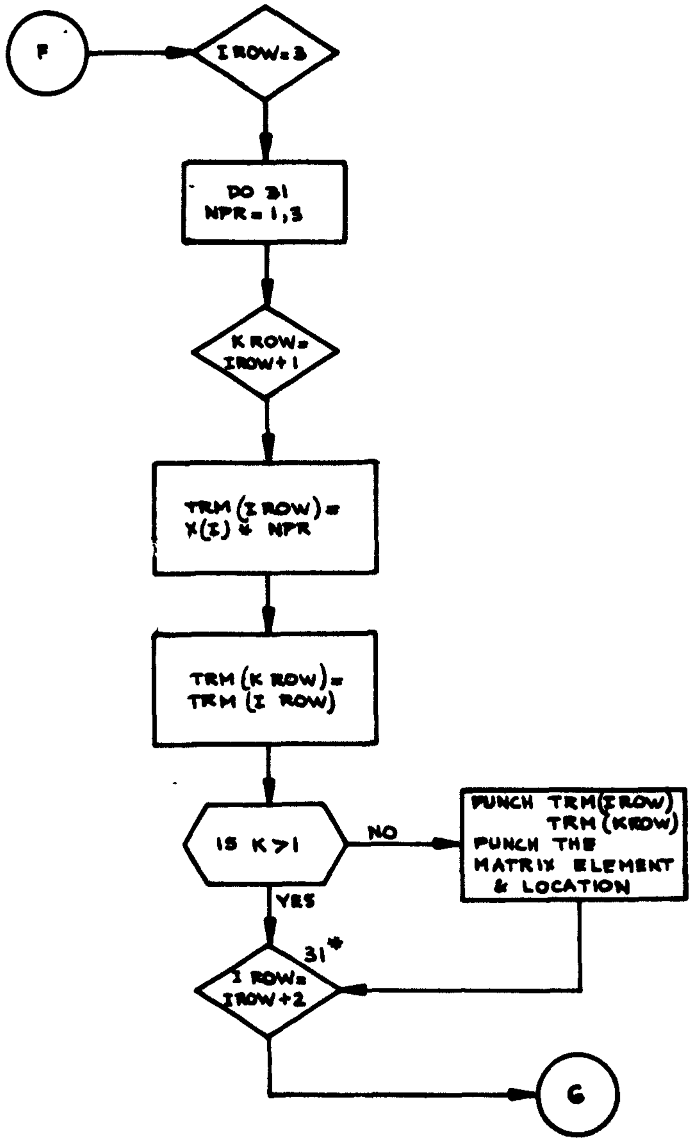

B
NCOMP=1
805*
IF S.SW 2
6*
PRINT HEADING FOR W.L.
7*
DO 15 I=1,M
DO 15 J=1,NS
513*
ZZ = X j-1
NO
IS NCOMP=1
IS THIS THE FIRST POINT ON THE WATERLINE
YES
ZY = Y i,j-1
IS Xj < FXi
YES
10*
POINT IS OUTSIDE PROFILE R=20.0
9*
ZZ = FXi
IS S.SW 2
OFF
ON
660*
PRINT I , J
NCOMP=2
1*
GO TO DIF 2ND SUBROUTINE
15*
RX i,j = R
14*
PRINT I,J ,RX i,j
ON
IF S.SW 2
RX i,N =RX i,N-1
OFF
C
NCOMP=1
18*

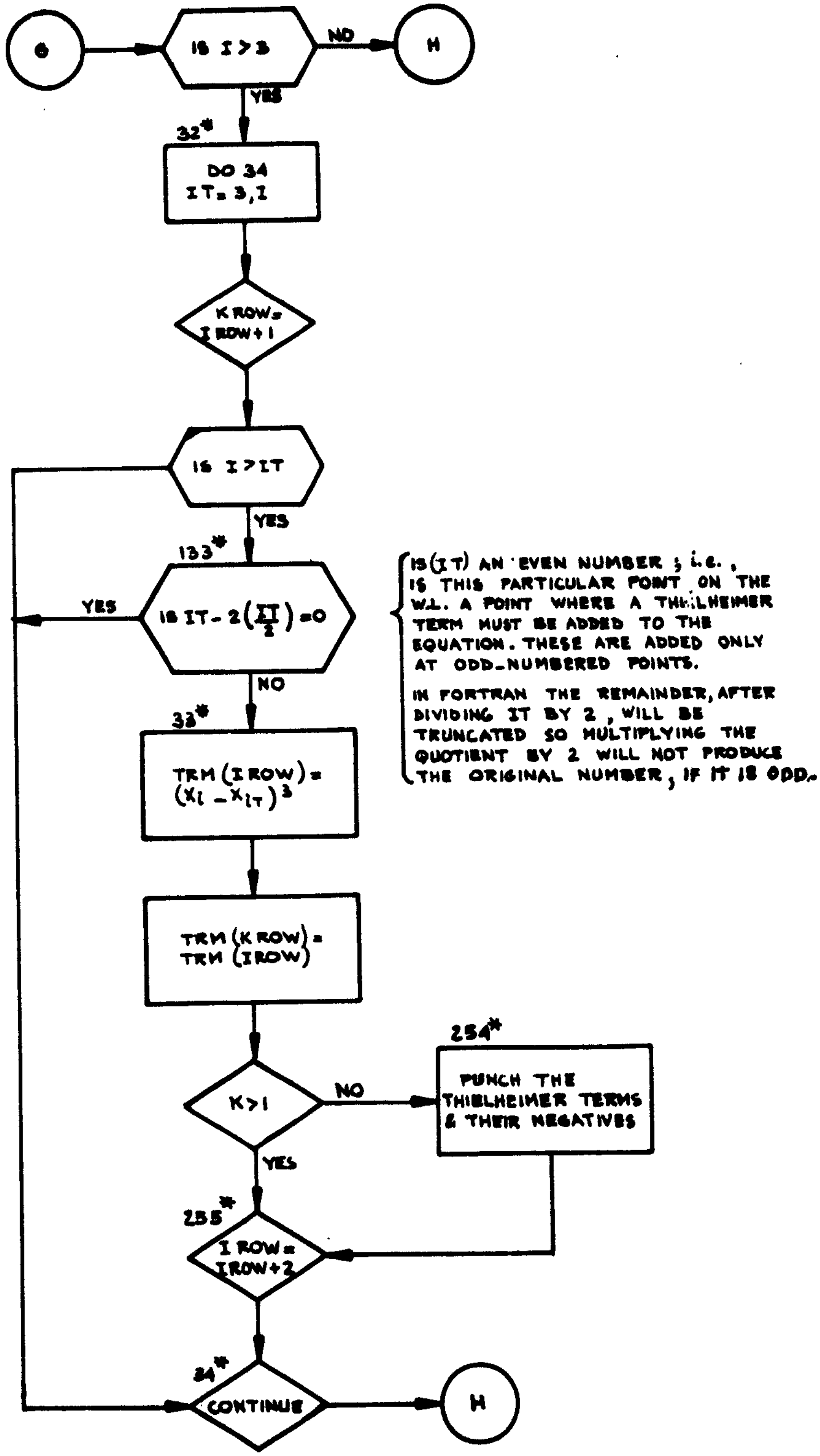

C
NCOMP=1
IF S.SW 2
16*
PRINT HEADING FOR STATION
17*
DO 25 J=1,N
DO 23 I=1,MS
19*
ZZ = Z_{i-1}
NO
IS NCOMP=1
IS THIS THE FIRST POINT ON THE STATION
YES
IS Z_i < FZ_j
YES
21*
POINT IS OUTSIDE PROFILE R=20.0
20*
ZZ = FZ_j
IS S.SW 2
OFF
ON
22*
PRINT J,I
NCOMP=2
514*
GO TO DIF 2ND SUBROUTINE
23*
RZ_{j,i} = R
24*
PRINT J,I, RZ_{j,M}
ON
IF S.SW 2
RZ_{j,M}=RZ_{j,M-1}
OFF
25*
D
NCOMP=1

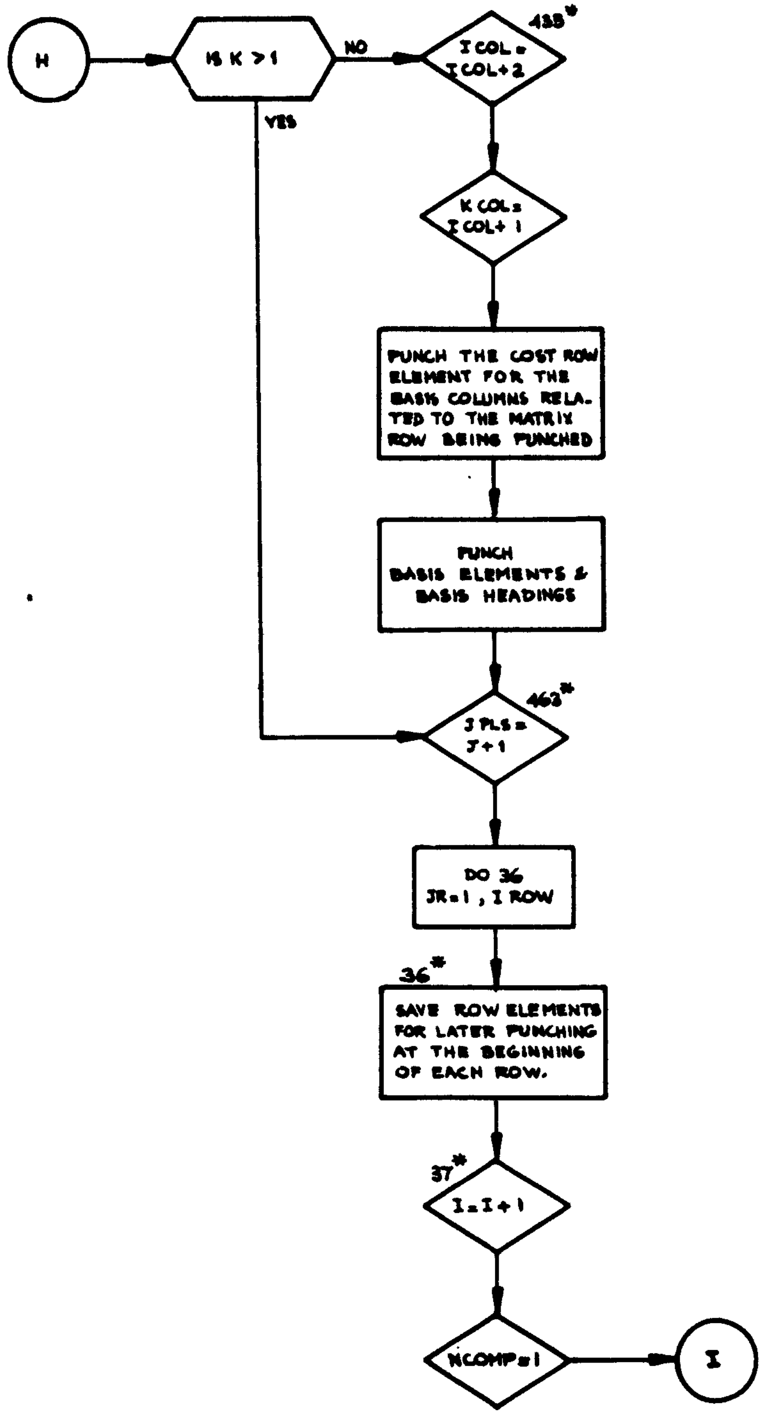

D
NJUL = [(N-3)/2 + 4] * 2
IS NAB < 1
JUL = 3
JUL = 2
JUL2 = 2
JUL2 = 1
NUMBER OF COLS. EQUALS (N+5)(M+5)/2
NUMBER OF COLS. EQUALS [(N-3)/2 + 4](2M+4)
PUNCH "ROW ID" "OFFSET"
ICOL = NUCOL -1
E

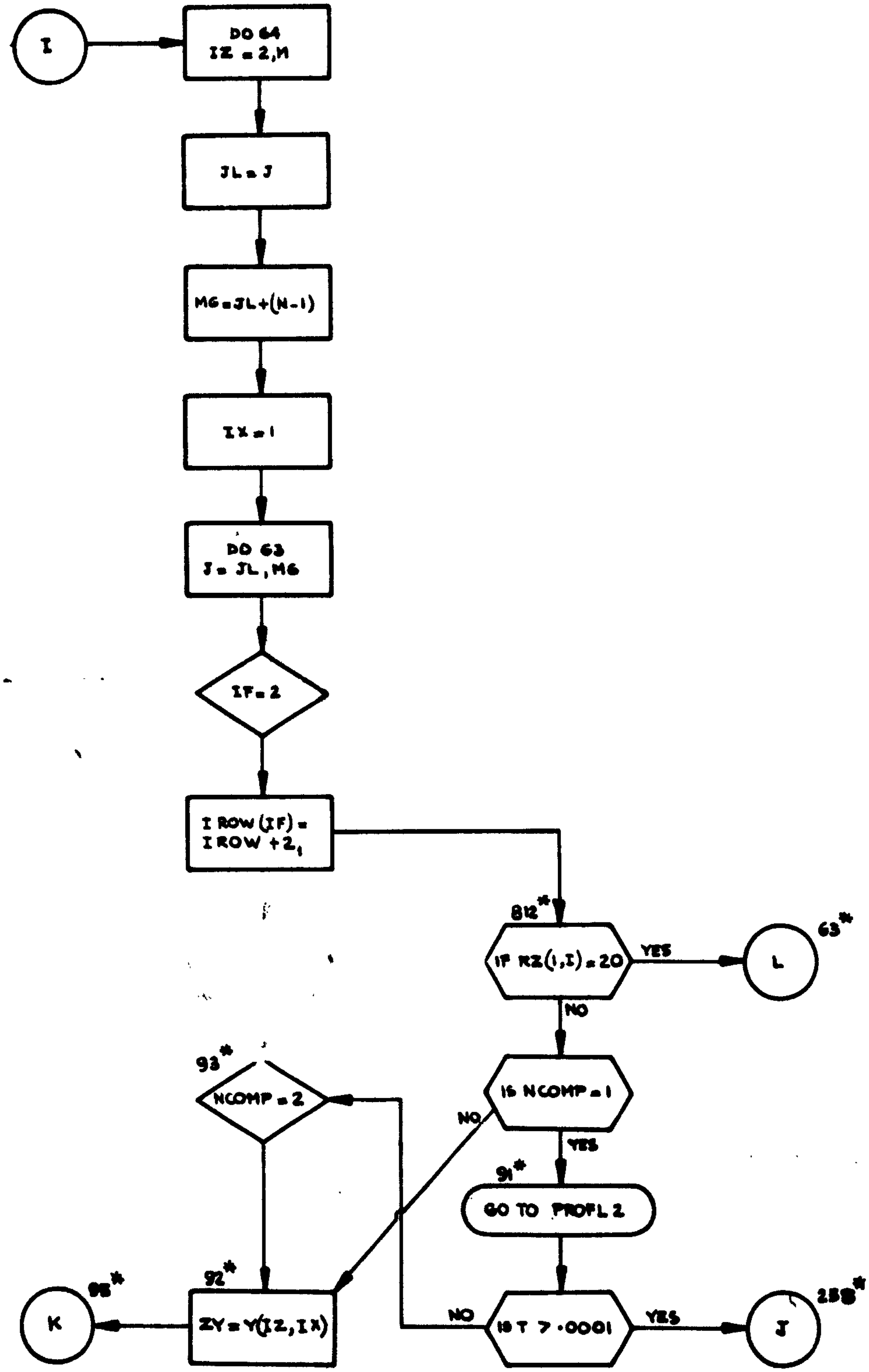

E
I = 1
FUPZ = -1
FUSZ = 0
DO 37
J = 1,N
810*
IS RX(1,I) = 20
YES
NO
250*
K = 1
251*
K = 2
IS NCOMP > 1
YES
NO
201*
GO TO PROFL 2
703*
NCOMP = 2
NO
IS T > .0001
YES
702*
DO NOT MODIFY OFFSET
ZY = Y(1,I)
257*
MODIFY OFFSET FOR END CONDITION
ZY = Y(1,I)/T
90*
PUNCH ,J
PUNCH THE ROW ID CARD FOR THIS ROW
PUNCH J,ZY, J,J
PUNCH RIGHT HAND SIDE AND CONSTANT COLUMNS
F

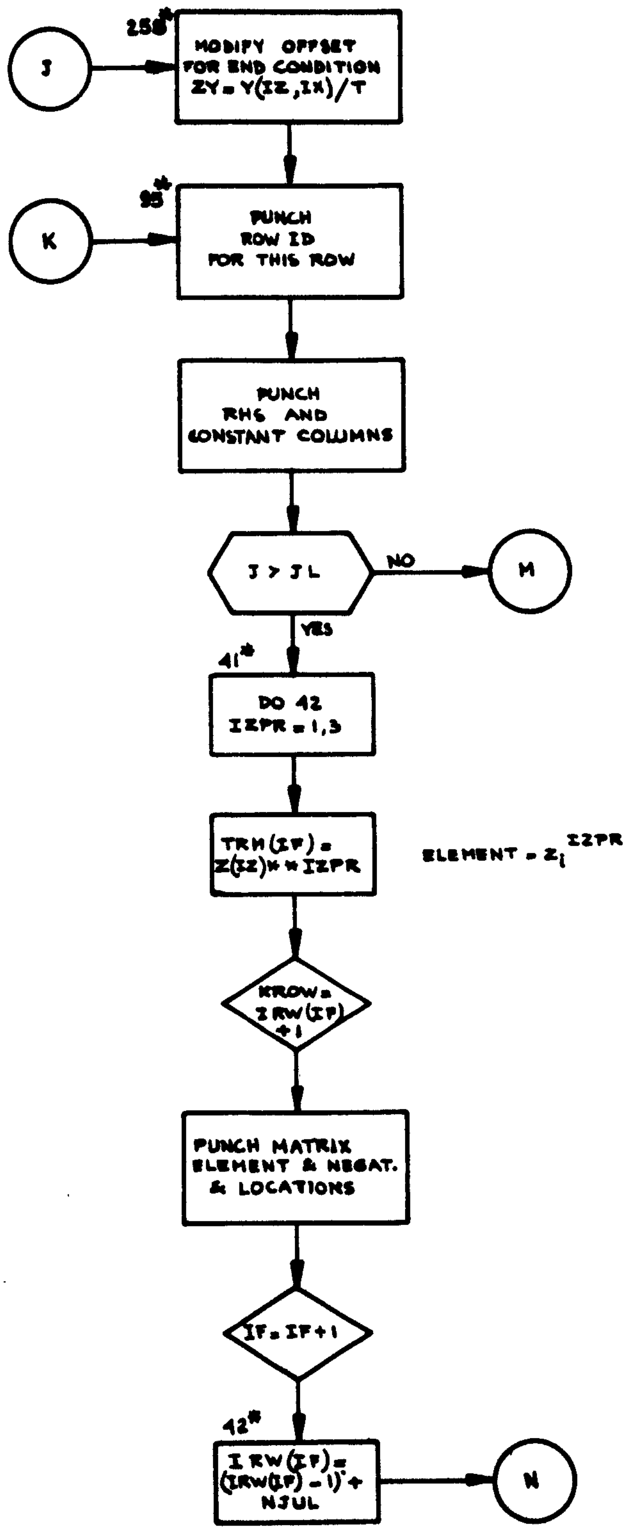
THIS LOOP CALCULATES THE FIRST THREE NON-CONSTANT TERMS OF THE EQUATION & THEIR NEGATIVE COUNTERPARTS FOR EACH POINT ON THE WATERLINE $[X_i, -X_i, X_i^2, -X_i^2, X_i^3, -X_i^3]$.



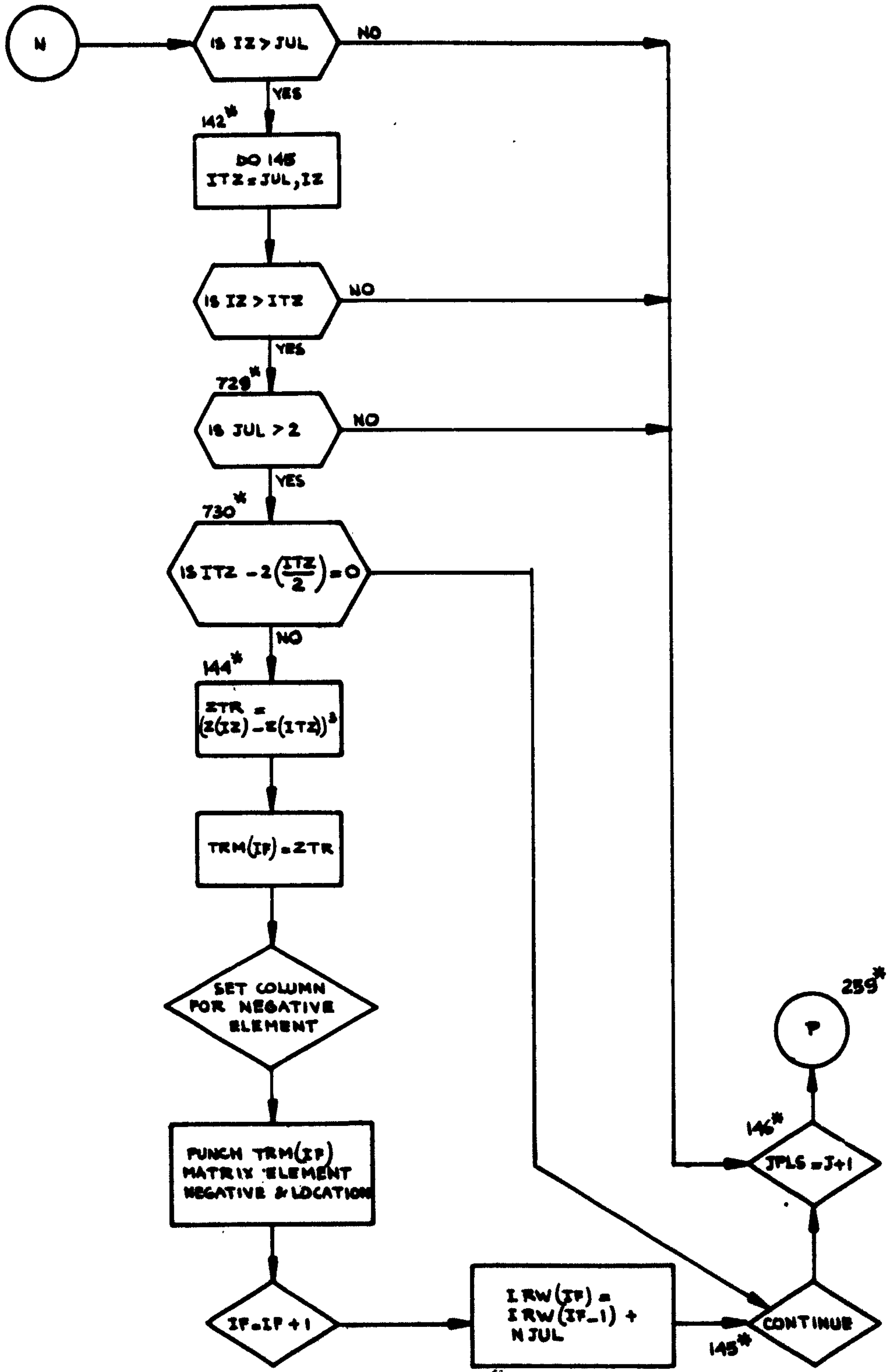

G
IS I > 3
NO
H
YES
32*
DO 34
IT = 3, I
KROW = IROW + 1
IS I > IT
YES
133*
YES
IS IT - 2(IT/2) = 0
NO
33*
TRM (IROW) = (X_I - X_IT)^3
TRM (KROW) = TRM (IROW)
K > 1
NO
254*
PUNCH THE THIELHEIMER TERMS & THEIR NEGATIVES
YES
255*
IROW = IROW + 2
34*
CONTINUE
H
IS (IT) AN EVEN NUMBER; i.e., IS THIS PARTICULAR POINT ON THE W.L. A POINT WHERE A THIELHEIMER TERM MUST BE ADDED TO THE EQUATION. THESE ARE ADDED ONLY AT ODD-NUMBERED POINTS.
IN FORTRAN THE REMAINDER, AFTER DIVIDING IT BY 2, WILL BE TRUNCATED SO MULTIPLYING THE QUOTIENT BY 2 WILL NOT PRODUCE THE ORIGINAL NUMBER, IF IT IS ODD.

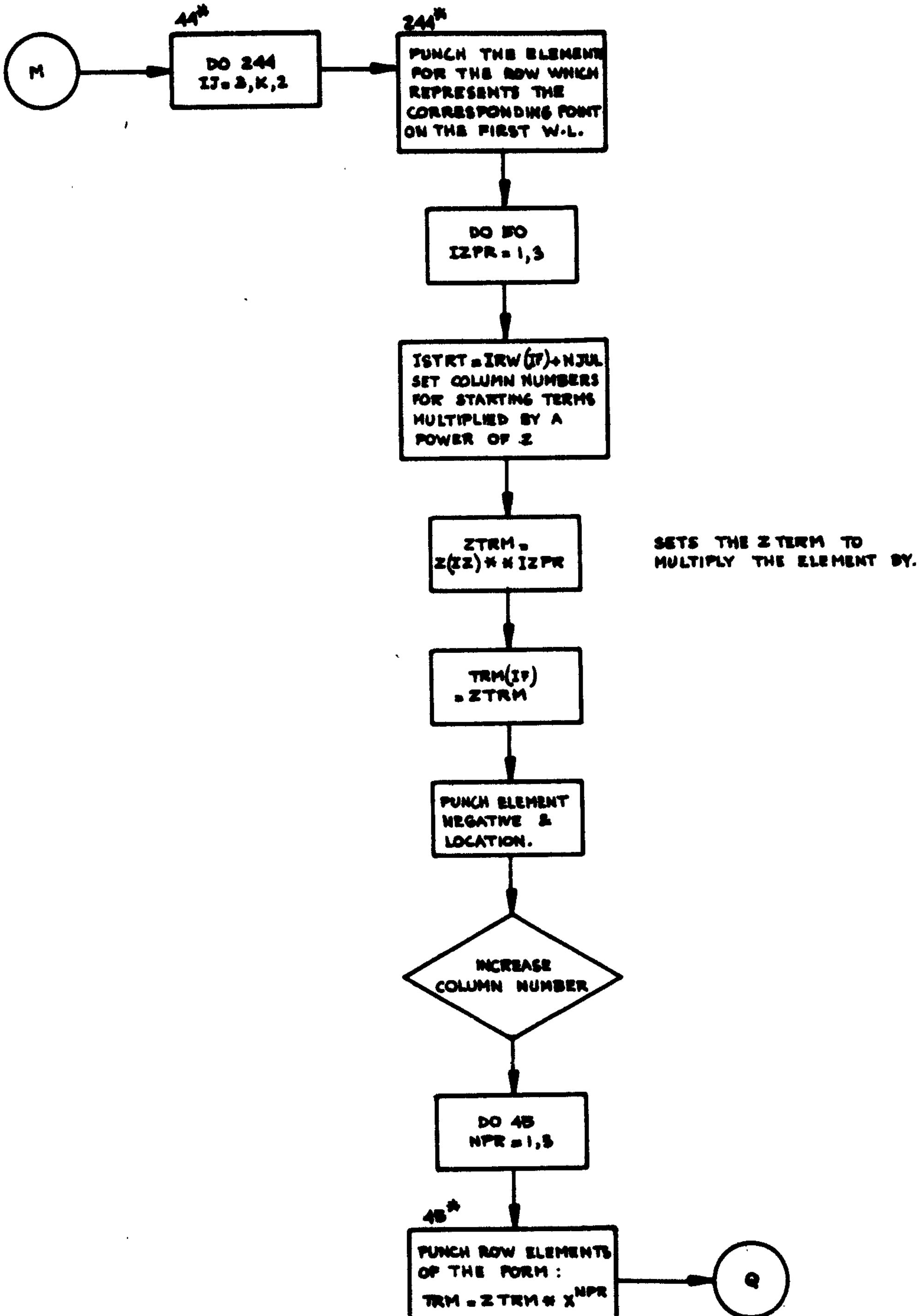

H
IS K >1
NO
YES
438*
I COL = I COL+2
K COL = I COL+1
PUNCH THE COST ROW ELEMENT FOR THE BASIS COLUMNS RELATED TO THE MATRIX ROW BEING PUNCHED
PUNCH BASIS ELEMENTS & BASIS HEADINGS
462*
J PLS = J+1
DO 36 JR=1, I ROW
36*
SAVE ROW ELEMENTS FOR LATER PUNCHING AT THE BEGINNING OF EACH ROW.
37*
I=I+1
NCOMP=1
I

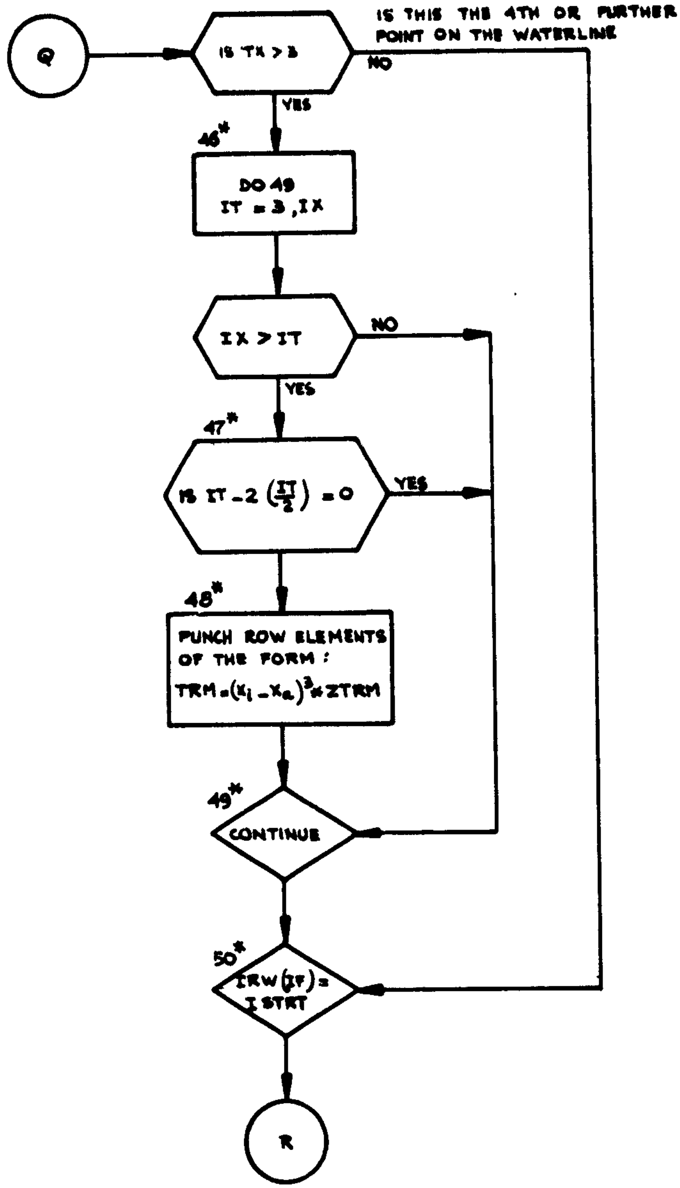

I
DO 64
IZ = 2,M
JL = J
MG = JL + (N-1)
IX = 1
DO 63
J = JL, MG
IF = 2
I ROW (IF) =
I ROW + 2,
812*
IF RZ(1,I) = 20
YES
L
63*
NO
IS NCOMP = 1
NO
YES
93*
NCOMP = 2
91*
GO TO PROFL 2
92*
ZY = Y(IZ,IX)
K
98*
NO
IS T > .0001
YES
J
258*

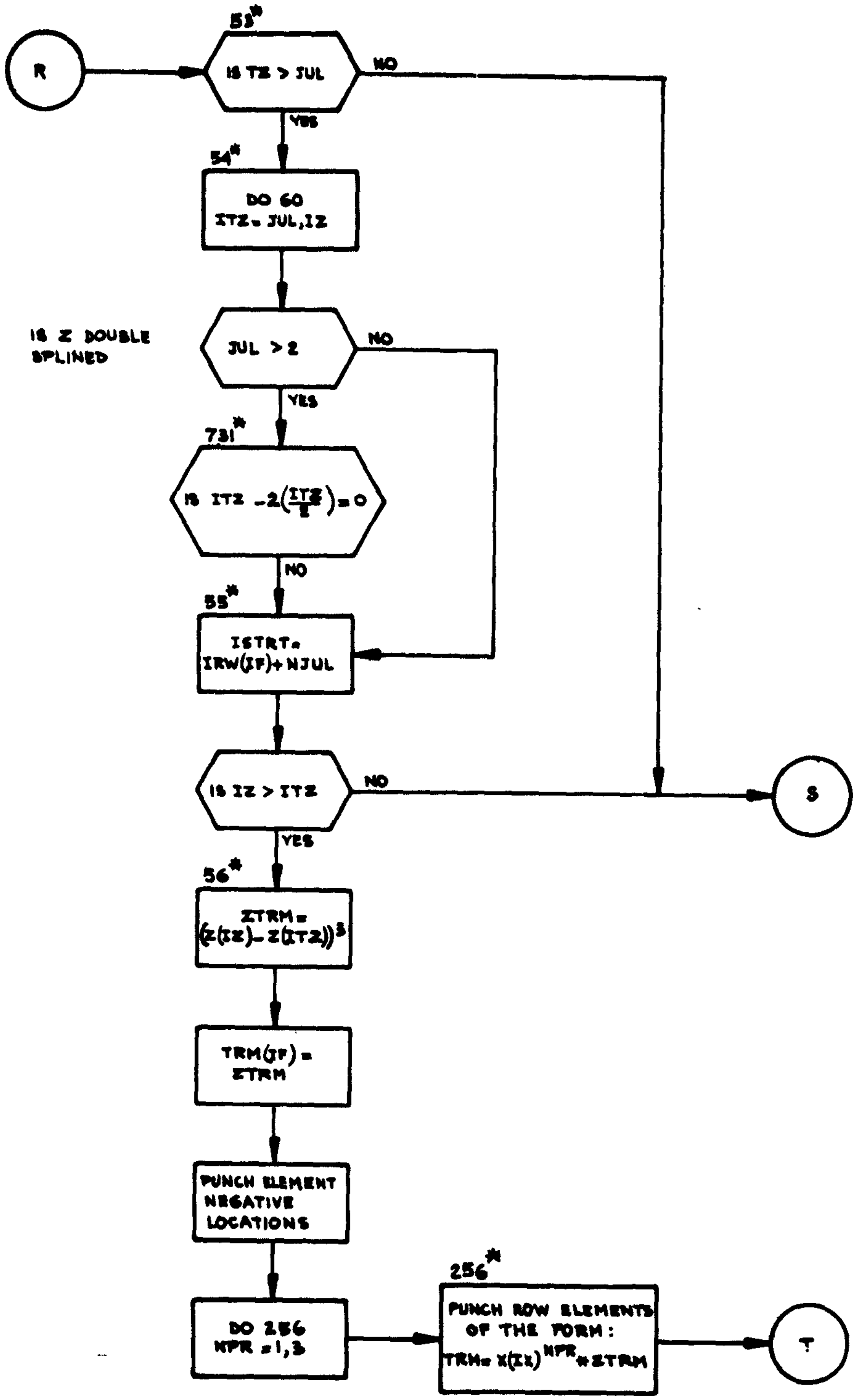


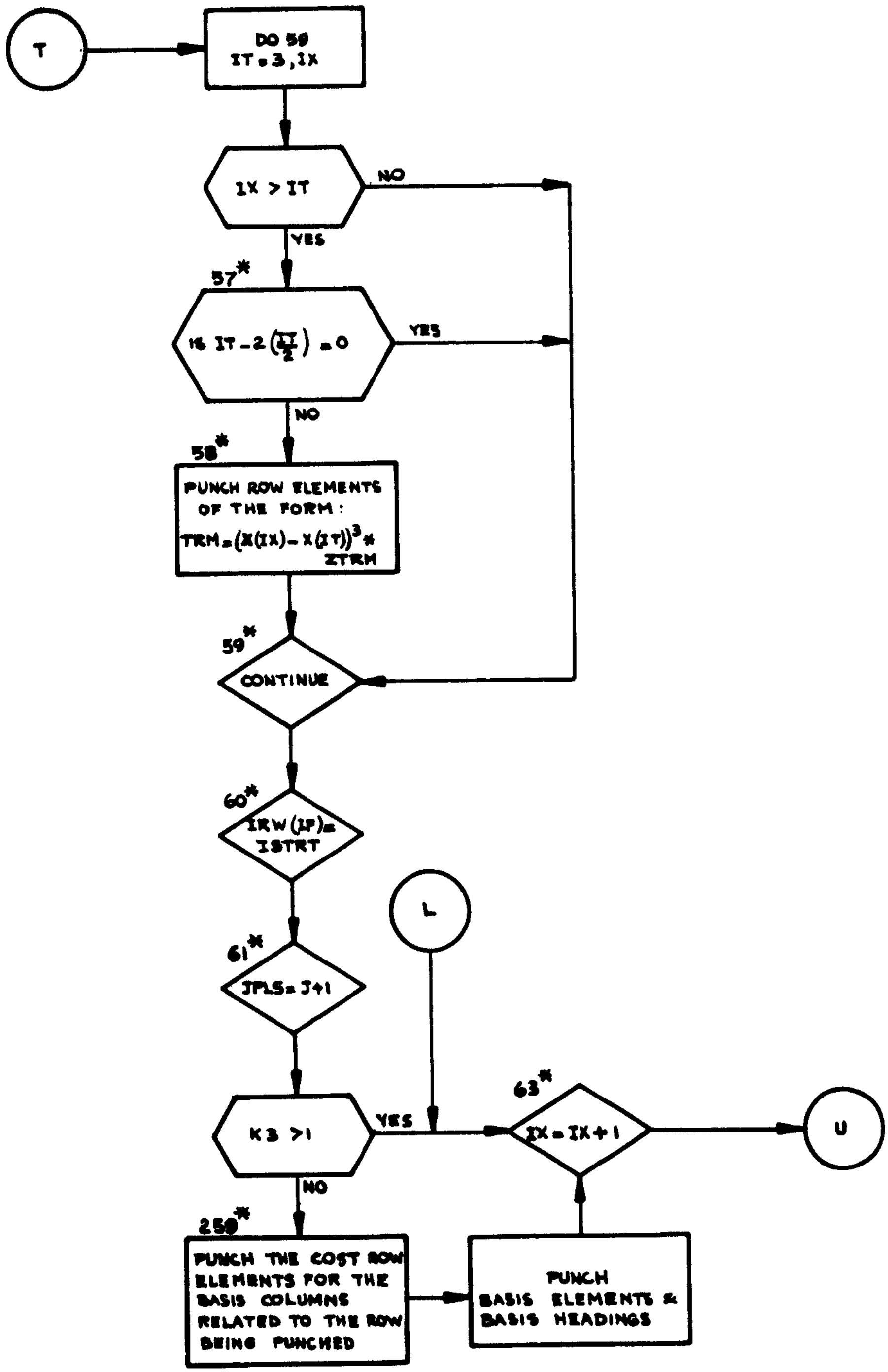
J
25B*
MODIFY OFFSET FOR END CONDITION ZY = Y(IZ,IX)/T
K
95*
PUNCH ROW ID FOR THIS ROW
PUNCH RHS AND CONSTANT COLUMNS
J > JL
NO
M
YES
41*
DO 42 IZPR = 1,3
TRM(IF) = Z(IZ)** IZPR
ELEMENT = Z_I^IZPR
KROW = IRW(IF) + 1
PUNCH MATRIX ELEMENT & NEGAT. & LOCATIONS
IF = IF + 1
42*
IRW(IF) = (IRW(IF) - 1) + NJUL
N

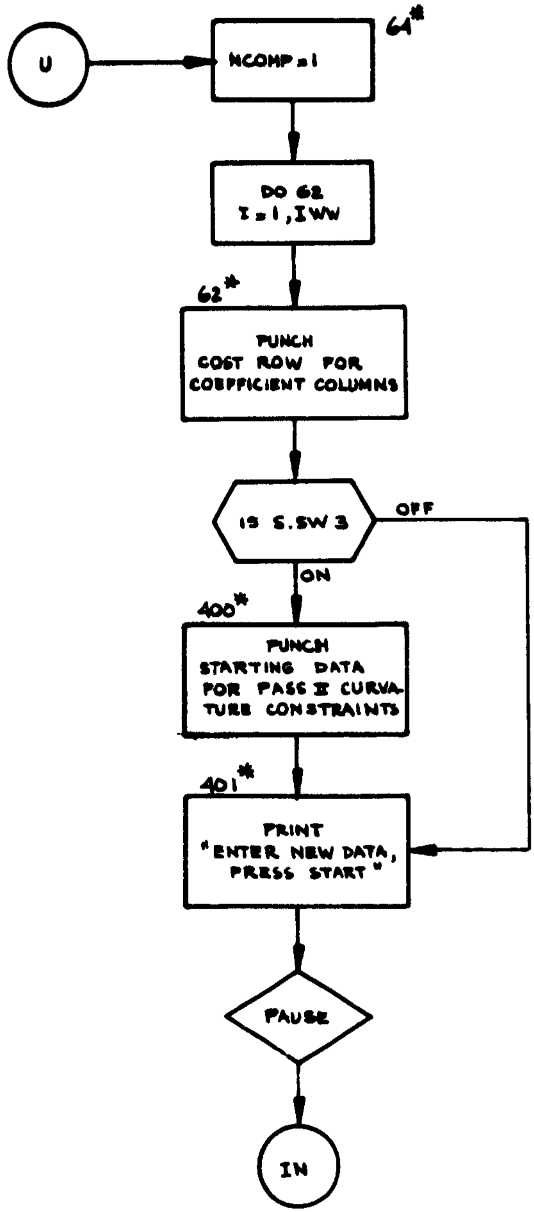

N
IS IZ > JUL
NO
YES
142*
DO 145
ITZ = JUL, IZ
IS IZ > ITZ
NO
YES
729*
IS JUL > 2
NO
YES
730*
IS ITZ - 2(ITZ/2) = 0
NO
144*
ZTR = (Z(IZ) - Z(ITZ))^3
TRM(IF) = ZTR
SET COLUMN FOR NEGATIVE ELEMENT
PUNCH TRM(IF) MATRIX ELEMENT NEGATIVE & LOCATION
IF = IF + 1
IRW(IF) = IRW(IF-1) + NJUL
CONTINUE
145*
146*
JPLS = J+1
P
259*


M
44*
DO 244
IJ=2,K,2
244*
PUNCH THE ELEMENT FOR THE ROW WHICH REPRESENTS THE CORRESPONDING POINT ON THE FIRST W.L.
DO 150
IZPR=1,3
ISTRT=IRW(IF)+NJUL
SET COLUMN NUMBERS FOR STARTING TERMS MULTIPLIED BY A POWER OF Z
ZTRM=
Z(IZ)**IZPR
SETS THE Z TERM TO MULTIPLY THE ELEMENT BY.
TRM(IF)
=ZTRM
PUNCH ELEMENT NEGATIVE & LOCATION.
INCREASE COLUMN NUMBER
DO 45
NPR=1,3
45*
PUNCH ROW ELEMENTS OF THE FORM:
TRM=ZTRM*X^NPR
Q


Q
IS TX > 3
IS THIS THE 4TH OR FURTHER POINT ON THE WATERLINE
NO
YES
46*
DO 49
IT = 3 , IX
IX > IT
NO
YES
47*
IS IT - 2 (IT/2) = 0
YES
48*
PUNCH ROW ELEMENTS OF THE FORM :
TRM = (Xi - Xa)3 * ZTRM
49*
CONTINUE
50*
IRW(IT) = I STRT
R


R
53*
IS TZ > JUL
NO
YES
54*
DO 60
ITZ = JUL,IZ
IS Z DOUBLE SPLINED
JUL > 2
NO
YES
731*
IS ITZ - 2(ITZ/2) = 0
NO
55*
ISTRT = IRW(IF) + NJUL
IS IZ > ITZ
NO
S
YES
56*
ZTRM = (Z(IZ) - Z(ITZ))^3
TRM(IF) = ZTRM
PUNCH ELEMENT NEGATIVE LOCATIONS
DO 256
NPR = 1,3
256*
PUNCH ROW ELEMENTS OF THE FORM:
TRM = X(IX)^NPR * ZTRM
T


T
DO 59
IT = 3, IX
IX > IT
NO
YES
57*
IS IT - 2(IT/2) = 0
YES
NO
58*
PUNCH ROW ELEMENTS OF THE FORM:
TRM = (X(IX) - X(IT))^3 * ZTRM
59*
CONTINUE
60*
IRW(IF) = ISTRT
61*
JPLS = J+1
L
K3 > 1
YES
NO
63*
IX = IX + 1
U
259*
PUNCH THE COST ROW ELEMENTS FOR THE BASIS COLUMNS RELATED TO THE ROW BEING PUNCHED
PUNCH BASIS ELEMENTS & BASIS HEADINGS


U
64*
NCOMP = 1
DO 62
I = 1, IWW
62*
PUNCH
COST ROW FOR
COEFFICIENT COLUMNS
IS S.SW 3
OFF
ON
400*
PUNCH
STARTING DATA
FOR PASS II CURVA-
TURE CONSTRAINTS
401*
PRINT
"ENTER NEW DATA,
PRESS START"
PAUSE
IN

```
C     SMOG3 PASS 1  PRIMAL  SUM OF DEV  X OR X AND Z DBL SPL WI PRFL
      DIMENSION RX(11,15),RZ(15,11),TRM(150), IRW(250),FZ(11),FX(15
)
      DIMENSION X(15),Z(11),Y(11,15),WL(15,11),IWL(15),C(13),FPZ(11
),
     CFSZ(11)
      COMMON JPLS,JUL,JUL2,NJUL,M,N,MS,NS, NUCOL,NCOMP,CURV,COST
      COMMON EXP2,COST,POINT,EXP1,IWW,ICOL
      COMMON X,FZ,RZ,Z,FX,FPZ,FSZ,RX,C
C                *   *    INPUT    *    *
  750 READ 173
      PRINT 173
      READ 150,NAA,NAB,POINT,COST
      READ 150,M,N,CURV,RHS,EXP1,EXP2
      ML=M+2
      DO 751 I=1,ML
  751 READ 154,C(I)
      NCOMP=1
      NL=N+1
      ML=M+1
      NS=N-1
      MS=M-1
        DO 2 I=1,M
      READ 154,Z(I)
      DO 2 J=1,N
      IF(SENSE SWITCH 3)802,803
  802 READ 183,X(J),IAY,IBY,ICY
      AAY=IAY
      BBY=IBY
      CCY=ICY
      Y(I,J)=AAY+(BBY+(CCY/8.))/12.
      GO TO 2
  803 READ 154,X(J),Y(I,J)
    2 CONTINUE
      IF(NAA)3,3,4
    3 READ 154,SX,SZ,TX,TZ
      GO TO 508
    4 SX=X(2)-X(1)
      SZ=Z(2)-Z(1)
      TZ=Z(1)
      TX=X(1)
C               *   *    NORMALIZING         *    *
  508 DO 26 J=1,N
   26 X(J)=(X(J)-TX)/SX
      DO 27 I=1,M
   27 Z(I)=(Z(I)-TZ)/SZ
```

```
      POINT=(POINT-TX)/SX
C       *    *    END PROFILE INTERSECTIONS    *    *
    5 IF(SENSE SWITCH 2)509,820
  509 PRINT 165
  820 DO 694 I=1,M
      FX(I)=C(1)+C(2)*Z(I)+C(3)*Z(I)*Z(I)+C(4)*Z(I)**3
      CALL DERIV1 (C(2),C(3),C(4),Z(I),FPZ(I),FSZ(I))
      IF(I-3)690,691,691
  691 DO 700 J=3,I
      DIF=Z(I)-Z(J-1)
      FX(I)=FX(I)+C(J+2)*DIF**3
      CALL DERIV2 (C(J+2),DIF,FPZ(I),FSZ(I),FPZ(I),FSZ(I))
      IF(I-3)690,690,700
  700 CONTINUE
  690 FPZ(I)=FPZ(I)/SZ
      FSZ(I)=FSZ(I)/(SZ*SZ)
      IF(SENSE SWITCH 2)693,694
  693 PRINT 170,I,FX(I)
      PRINT 154,FPZ(I),FSZ(I)
  694 CONTINUE
      J=1
      OLD =Z(M)
      I=MS
      IF(SENSE SWITCH 2)204,510
  204 PRINT 179
  510 FZ(J)=FX(I)
      ZZ=Z(I)
      ICOMP=1
  695 IF(ABSF(X(J)-FZ(J))-.0001) 66,66,71
   71 GO TO (65,69),ICOMP
   65 IF(X(J)-FZ(J)) 67,66,72
   66 FZ(J)=ZZ
      IF(SENSE SWITCH 2) 74,80
   74 PRINT 170,J,FZ(J)
   80 J=J+1
      OLD=ZZ
      GO TO 510
   67 ZZ=ZZ+(OLD-ZZ)/2.
      ICOMP=2
      IJUL=I+1
   68 FZ(J)=C(1)+C(2)*ZZ+C(3)*ZZ*ZZ+C(4)*ZZ**3
      IF(IJUL-3) 695,696,697
  696 FZ(J)=FZ(J)+C(5)*(ZZ-Z(2))**3
      GO TO 695
  697 DO 699 K=3,IJUL
```

```
699 FZ(J)=FZ(J)+C(K+2)*(ZZ-Z(K-1))**3
    GO TO 695
 69 IF(X(J)-FZ(J)) 67,66,70
 70 SEG=OLD-ZZ
    OLD=ZZ
    ZZ=ZZ-SEG/2.
    GO TO 68
 72 OLD=Z(I)
    I=I-1
    IF(I-1)81,510,510
 81 DO 82 I=J,N
    FZ(I)=Z(I)
    IF(SENSE SWITCH 2)83,82
 83 PRINT 170,I,FZ(I)
 82 CONTINUE
C     *  *      WATERLINE  2ND  DIFFERENCES     *    *
805 IF(SENSE SWITCH 2)6,7
  6 PRINT 157
  7 DO 15 I=1,M
    DO 13 J=1,NS
    GO TO (8,513),NCOMP
  8 ZY=0.
    IF(X(J)-FX(I))10,10,9
 10 R=20.
    IF(SENSE SWITCH 2)660,13
660 PRINT 151,I,J
    GO TO 13
  9 ZZ=FX(I)
    NCOMP=2
    GO TO 1
513 ZZ=X(J-1)
    ZY=Y(I,J-1)
  1 CALL DIF2ND (X(J),X(J+1),ZZ,Y(I,J),Y(I,J+1),ZY,I,J,R,A)
 13 RX(I,J)=R
    RX(I,N)=RX(I,NS)
    IF(SENSE SWITCH 2)14,15
 14 PRINT 158,I,J,A,RX(I,N)
 15 NCOMP=1
C     *  *    STATION  2ND  DIFFERENCES     *    *
    IF(SENSE SWITCH 2)16,17
 16 PRINT 159
 17 DO 25 J=1,N
    DO 23 I=2,MS
    GO TO (18,19),NCOMP
 18 IF(Z(I)-FZ(J))21,21,20
```

```
   21 R=20.
      IF(SENSE SWITCH 2)22,23
   22 PRINT 151,J,I
      GO TO 23
   20 ZZ=FZ(J)
      NCOMP=2
      GO TO 514
   19 ZZ=Z(I-1)
  514 CALL DIF2ND (Z(I),Z(I+1),ZZ,Y(I,J),Y(I+1,J),Y(I-1,J),J,I,R,A)
   23 RZ(J,I)=R
      RZ(J,M)=RZ(J,MS)
      IF(SENSE SWITCH 2)24,25
   24 PRINT 158,J,I,A,RZ(J,M)
   25 NCOMP=1
C                   *    *           ROW ID               *    *
      PUNCH 173
      NJUL=(((N-3)/2)+4)*2
      NM=N
      IF(NAB)719,718,718
  719 JUL=3
      JUL2=2
      NUCOL=((N+5)*(M+5))/2
      GO TO 720
  718 JUL=2
      JUL2=1
      NUCOL=2*(M+2)*(((N-3)/2)+4)
  720 PUNCH 160
      ICOL=NUCOL-1
C                   *    *        MATRIX             *    *
      PUNCH 164
C                        *    *           FIRST  WATERLINE          *
 *
      MLG=NM
      I=1
      FUPZ=-1.
      FUSZ=0.
      DO 37 J=1,MLG
  810 IF(RX(1,I)-20.)250,251,250
  251 K=2
      GO TO 252
  250 K=1
  200 GO TO (201,702),NCOMP
  201 CALL PROFL2(X(I),FX(1),EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS
 ,NCOMP)
```

```
      IF(T-.0001)703,703,257
  257 ZY=Y(1,I)/T
      GO TO 90
  703 NCOMP=2
  702 ZY=Y(1,I)
   90 PUNCH 162,J
      PUNCH 177,J,ZY,J,J
  252 IROW=3
      DO 31 NPR=1,3
      KROW=IROW+1
      TRM(IROW)=X(I)**NPR
      TRM(KROW)=TRM(IROW)
      GO TO (253,31),K
  253 PUNCH 167,IROW,J,TRM(IROW),KROW,J,TRM(IROW)
   31 IROW=IROW+2
      IF(I-3)35,35,32
   32 DO 34 IT=3,I
      KROW=IROW+1
      IF(I-IT)34,34,133
  133 IF(IT-2*(IT/2))33,34,33
   33 TRM(IROW)=(X(I)-X(IT))**3
      TRM(KROW)=TRM(IROW)
      GO TO (254,255),K
  254 PUNCH 167,IROW,J,TRM(IROW),KROW,J,TRM(IROW)
  255 IROW=IROW+2
   34 CONTINUE
   35 GO TO(435,463),K
  435 ICOL=ICOL+2
      KCOL=ICOL+1
      PUNCH 171,ICOL,RHS,KCOL,RHS
      PUNCH 152,ICOL,J,KCOL,J,ICOL,J
  463 JPLS=J+1
      IROW=IROW-1
      IWL(I)=IROW
      DO 36 JR=3,IROW
   36 WL(I,JR)=TRM(JR)
   37 I=I+1
      NCOMP=1
      K3=1
C        *    *    REMAINING WL    *    *
      DO 64 IZ=2,M
      JL=J
      MG=JL+NM-1
      IX=1
      DO 63 J=JL,MG
```

```
      IF=2
      IRW(IF)=IROW+1
  812 IF(RX(IZ,IX)-20.)202, 63,202
  202 GO TO (91,92),NCOMP
   91 CALL PROFL2(X(IX),FX(IZ),EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS
     ,NCOMP)
      IF(T-.0001)93,93,258
  258 ZY=Y(IZ,IX)/T
      GO TO 95
   93 NCOMP=2
   92 ZY=Y(IZ,IX)
   95 PUNCH 162,J
      PUNCH 177,J,ZY,J,J
  708 IF(J-JL)41,41,44
   41 DO 42 IZPR=1,3
      TRM(IF)=Z(IZ)**IZPR
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
      IF=IF+1
   42 IRW(IF)=IRW(IF-1)+NJUL
      IF(IZ-JUL)146,146,142
  142 DO 145 ITZ=JUL,IZ
      IF(IZ-ITZ)146,146,729
  729 GO TO (144,144,730),JUL
  730 IF(ITZ-2*(ITZ/2))144,145,144
  144 ZTR=(Z(IZ)-Z(ITZ))**3
      TRM(IF)=ZTR
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+NJUL
  145 CONTINUE
  146 JPLS=J+1
      GO TO 259
C     * * Z , Z SQR , Z CUBE ROWS * *
   44 IRO=1
      K=IWL(IX)
      DO 244 IJ=3,K,2
      IRO =IRO +2
      KROW=IRO+1
  244 PUNCH 167,IRO,J,WL(IX,IJ),KROW,J,WL(IX,IJ)
      DO 50 IZPR=1,3
      ISTRT=IRW(IF)+NJUL
      ZTRM=Z(IZ)**IZPR
      TRM(IF)=ZTRM
```

```
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+2
      DO 45 NPR=1,3
      TRM(IF)=(X(IX)**NPR)*ZTRM
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
      IF=IF+1
   45 IRW(IF)=IRW(IF-1)+2
      IF(IX-3)50,50,46
   46 DO 49 IT=3,IX
      IF(IX-IT)49,49,47
   47 IF(IT-2*(IT/2))48,49,48
   48 TRM(IF)=((X(IX)-X(IT))**3)*ZTRM
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+2
   49 CONTINUE
   50 IRW(IF)=ISTRT
C       *   *    THEILHEIMER  ROWS      *   *
   53 IF(IZ-JUL)61,61,54
   54 DO 60 ITZ=JUL,IZ
      GO TO (55,55,731),JUL
  731 IF(ITZ-2*(ITZ/2))55,60,55
   55 ISTRT=IRW(IF)+NJUL
      IF(IZ-ITZ)61,61,56
   56 ZTRM=(Z(IZ)-Z(ITZ))**3
      TRM(IF)=ZTRM
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
      IF=IF+1
      IRW(IF)=IRW(IF-1)+2
      DO 256 NPR=1,3
      TRM(IF)=(X(IX)**NPR)*ZTRM
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
      IF=IF+1
  256 IRW(IF)=IRW(IF-1)+2
      DO 59 IT=3,IX
      IF(IX-IT)59,59,57
   57 IF(IT-2*(IT/2))58,59,58
   58 TRM(IF)=((X(IX)-X(IT))**3)*ZTRM
      KROW=IRW(IF)+1
      PUNCH 167,IRW(IF),J,TRM(IF),KROW,J,TRM(IF)
```

```
      IF=IF+1
      IRW(IF)=IRW(IF-1)+2
   59 CONTINUE
      IWW=KROW
   60 IRW(IF)=ISTRT
   61 JPLS=J+1
      GO TO (259,63),K3
  259 ICOL=ICOL+2
      KCOL=ICOL+1
      PUNCH 171,ICOL,RHS,KCOL,RHS
      PUNCH 152,ICOL,J,KCOL,J,ICOL,J
   63 IX=IX+1
   64 NCOMP=1
      DO 62 I=1,IWW
   62 PUNCH 171,I,CURV
      IF(SENSE SWITCH 3)400,401
  400 CONTINUE
      PUNCH 180,JPLS,JUL,JUL2,NJUL,M,N,MS,NS,NUCOL ,NCOMP, COST,COST
      PUNCH 181,EXP2,COST,POINT,EXP1,IWW,ICOL
      DO 87 I=1,N
      PUNCH 181,X(I),FZ(I)
      DO 87 J=2,M
  87  PUNCH 181,RZ(I,J)
      DO 88 I=1,M
      PUNCH 181,Z(I),FX(I),FPZ(I),FSZ(I)
      DO 88 J=1,N
   88 PUNCH 181,RX(I,J)
      ML=M+2
      DO 100 J=1,ML
  100 PUNCH 181,C(J)
  401 PRINT 178
      PAUSE 751
      GO TO 750
  150 FORMAT(2I5,4F11.6)
  151 FORMAT(2I5,17H   OUT OF SURFACE)
  152 FORMAT(7X,1HC,I4,2H R,I4,8H      1.0/7X,1HC,I4,2H R,I4,8H-
1.0/
     C  7X,1HC,I4,2H R,I4)
  154 FORMAT(4F10.4)
  157 FORMAT(//10X,17H2ND DIFF - WLINES//
     C30H    WL    STA NORM VALUE     SIGN/)
  158 FORMAT(2I5,F12.6,F8.2)
  159 FORMAT(//10X,17H2ND DIFF-STATIONS//
     C30H    STA    WL NORM VALUE     SIGN/)
  160 FORMAT(6HROW ID/12X,6H OFSET)
```

```
  162 FORMAT(12X,2H R,I4)
  164 FORMAT(6HMATRIX)
  165 FORMAT(//6X,25HEND PROFILE INTERSECTIONS//15X,6HWLINES/)
  167 FORMAT(6X,2H C,I4,2H R,I4,1H-,F11.5/6X,2H C,I4,2H R,I4,F12.5)
  169 FORMAT(6X,2H C,I4,2H R,I4,F12.5)
  170 FORMAT(5X,I5,F12.6)
  171 FORMAT(6X,2H C,I4,6H OFSET,1H ,F11.5)
  172 FORMAT(2I5)
  173 FORMAT(1H*,5X,47H
  )
  177 FORMAT(12X,2H R,I4,1H ,F11.5/7X,6HC   1 ,1HR,I4,1H-,4X,3H1.0/
     C  7X,6HC   2 ,1HR,I4,5X,3H1.0)
  178 FORMAT(28HENTER NEW DATA - PRESS START)
  179 FORMAT (/14X,8HSTATIONS/)
  180 FORMAT (10I5,2F15.9)
  181 FORMAT (4F15.9,I4,I4)
  182 FORMAT(6X,2H C,I4,6H OFSET,1H ,F11.5/6X,2H C,I4,6H OFSET,1H F
11.5)
  183 FORMAT(F10.5,3I5)
      END
```

Note: See SMOG 2, PASS 1 for listings of the subroutines

OPERATING INSTRUCTIONS

Pass 2 of the program accepts the data punched in Pass 1. It produces the curvature constraints for the matrix.

Fortran Input Symbols

JPLS	-	The number of the last column produced in Pass 1.
JUL	-	Program indicator JUL = 3 if the surface is double splined in the z direction JUL = 2 if the surface is single splined in the z direction This value is equal to the subscript of the first z value of the point of discontinuity.
JUL2	-	Program indicator JUL2 = 2 if the z direction is double splined JUL2 = 1 if the z direction is single splined. This value is equal to the number of waterline intervals between points of discontinuity.
NJUL	-	This value $2(\frac{N-3}{2}+4)$ is equal to the maximum number of rows necessary to store the x terms, each of which is multiplied by a specific z term
M	-	Total number of waterlines in the surface
MS	-	M-1
N	-	Total number of stations in the surface
NS	-	N-1
NUCOL	-	Total number of columns in the non-slack matrix For z double spline, NUCOL = $\frac{(N+5)(M+5)}{2}$ For z single spline, NUCOL = 2(M+2)NJUL
NCOMP	-	Program indicator
COST	-	Becomes equal to curvature. Not used (See RHS and CURV)

EXP1	-	The value of p in the profile functions, usually made .3333
EXP2	-	The value of r in the profile function, usually made 3.0 .
POINT	-	The value of D in the profile function. This number is the length of the fration of the station spacing over which the profile function is effective in terms of the full-scale surface (the value has been translated and normalized exactly like the x values are).
C(I)	-	Coefficients of the Theilheimer equation describing the profile. There must be (M+2) coefficients. Straight lines or other curves may be used by making all except the desired coefficients zero.
CURV	-	Is equal to the value of COST in Pass 1. This is the value of the cost on the slacks corresponding to the curvature constraints. Usually equal to zero.
RHS	-	Equal to the value of COST from Pass 1 . This is the value of the right-hand side for the curvature constraints.
X(J)	-	Distance along the x axis from the origin of the surface to Station j .
Z(I)	-	Distance along the z axis from the origin of the surface to Waterline i .
FX(I)	-	The x coordinate of the intersection of Waterline i with the profile.
FZ(J)	-	The z coordinate of the intersection of Station j with the profile
FPZ(I)	-	The first derivative with respect to z of the profile equation at Waterline i .
FSZ(I)	-	The second derivative with respect to z of the profile equation at Waterline i .
RZ(I,J)	-	The sign of the second difference of the Station Curve j at Waterline i . This will have the opposite sign from the actual second difference.
KX(I,J)	-	The sign of the second difference of Waterline Curve i at Station j . This will be the opposite of the actual sign
ICOL	-	The column number of the first curvature slack.
IWW	-	Not used in Pass 2 .

Input

The following input formats are listed as a matter of record in case modifications of the data cards are to be made. Normally, this data will be obtained from executing Pass 1 in a form which can be entered directly into Pass 2.

Contents of Card									Card NO.
Format	I5	I5	I5	I5	I5	I5	I5	I5	
Variable	JPLS	JUL	JUL2	NJUL	M	N	MS	NS	
Format	(Continuation of Card 1)			I5	I5	F15.5	F15.5		1
Variable				NUCOL	NCOMP	RHS	CURV		
Format	F15.9	F15.9	F15.9	F15.9	I4	I4			2
Variable	EXP2	COST	POINT	EXP1	IWW	ICOL			
Format	F15.9	F15.9						1st card/set	Next N Sets
Variable	X(I)	FZ(I)							
Format	F15.9							M-1 Cards/Set	
Variable	RZ(I,J)								
Format	F15.9	F15.9	F15.9	F15.9				1st Card/Set	Next M Sets
Variable	Z(I)	FX(I)	FPZ(I)	FSZ(I)					
Format	F15.9							N-1 Cards/Set	
Variable	RX(I,J)								
Format	F15.9								Next M+2 Cards
Variable	C(J)								

Output

This program produces the curvature constraints in row order with the columns ordered within the rows. It also produces the row identifications, slack cost entries, slack elements, basis headings, and right-hand side entries for these rows. The output must be sorted together with the output from Pass 1 before use.

Sense Switches:

Sense Switch 3 ON - The data punched in Pass 1 is read

Sense Switch 3 OFF- The data is picked up from the common storage locations.

* * SAMPLE INPUT * *

```
  26     3     2    10     5     5     4     4              1     0.00000000
 1000.00000000
      .33333333      0.00000000        .07843137      3.00000000   50
91
     0.00000000      2.60400380
    20.00000000
    20.00000000
     1.00000000
     1.00000000
     1.00000000       .90136719
     1.00000000
    -1.00000000
    -1.00000000
    -1.00000000
     2.00000000      0.00000000
    -1.00000000
    -1.00000000
    -1.00000000
    -1.00000000
     3.00000000      0.00000000
    -1.00000000
    -1.00000000
    -1.00000000
    -1.00000000
     4.00000000      0.00000000
    -1.00000000
    -1.00000000
    -1.00000000
    -1.00000000
     0.00000000      1.52935000       -.58730000      0.00000000
    20.00000000
    20.00000000
     1.00000000
    -1.00000000
    -1.00000000
     1.00000000       .94205000       -.58730000      0.00000000
    20.00000000
     1.00000000
    -1.00000000
    -1.00000000
    -1.00000000
     2.00000000       .35475000       -.58730000      0.00000000
    20.00000000
     1.00000000
    -1.00000000
    -1.00000000
    -1.00000000
```

3.00000000	-.23255000	-.58730000	0.00000000
1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
4.00000000	-.81985000	-.58730000	0.00000000
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
-1.00000000			
1.52935000			
-.58730000			
0.00000000			
0.00000000			
0.00000000			
0.00000000			
0.00000000			

* *   SAMPLE OUTPUT   * *

```
          R   27
          R   27      0.00000
C    5 R   27-     1.00000
C    6 R   27      1.00000
C    7 R   27-     6.00000
C    8 R   27      6.00000
C   93 OFSET     000.00000
C   93 R   27
C   93 R   27      1.0
          R   28
          R   28      0.00000
C    5 R   28      1.00000
C    6 R   28-     1.00000
C    7 R   28     12.00000
C    8 R   28-    12.00000
C    9 R   28      6.00000
C   10 R   28-     6.00000
C   94 OFSET     000.00000
C   94 R   28
C   94 R   28      1.0
          R   30
          R   30      0.00000
C    5 R   30      1.00000
C    6 R   30-     1.00000
C    7 R   30      6.00000
C    8 R   30-     6.00000
C   15 R   30      1.00000
C   16 R   30-     1.00000
C   17 R   30      6.00000
C   18 R   30-     6.00000
C   25 R   30      1.00000
C   26 R   30-     1.00000
C   27 R   30      6.00000
C   28 R   30-     6.00000
C   35 R   30      1.00000
C   36 R   30-     1.00000
C   37 R   30      6.00000
C   38 R   30-     6.00000
C   95 OFSET     000.00000
C   95 R   30
```

1.0.0-2

```
C   95 R   30      1.0
       R   31
       R   31      0.00000
C    5 R   31      1.00000
C    6 R   31-     1.00000
C    7 R   31     12.00000
C    8 R   31-    12.00000
C    9 R   31      6.00000
C   10 R   31-     6.00000
C   15 R   31      1.00000
C   16 R   31-     1.00000
C   17 R   31     12.00000
C   18 R   31-    12.00000
C   19 R   31      6.00000
C   20 R   31-     6.00000
C   25 R   31      1.00000
C   26 R   31-     1.00000
C   27 R   31     12.00000
C   28 R   31-    12.00000
C   29 R   31      6.00000
C   30 R   31-     6.00000
C   35 R   31      1.00000
C   36 R   31-     1.00000
C   37 R   31     12.00000
C   38 R   31-    12.00000
C   39 R   31      6.00000
C   40 R   31-     6.0000C
C   96 OFSET     000.00000
C   96 R   31
C   96 R   31      1.0
       R   33
       R   33      0.00000
C    5 R   33      1.00000
C    6 R   33-     1.00000
C    7 R   33      6.00000
C    8 R   33-     6.00000
C   15 R   33      2.00000
C   16 R   33-     2.00000
C   17 R   33     12.00000
C   18 R   33-    12.00000
C   25 R   33      4.00000
```

1.0.0-2

```
C   26 R   33-     4.00000
C   27 R   33     24.00000
C   28 R   33-    24.00000
C   35 R   33      8.00000
C   36 R   33-     8.00000
C   37 R   33     48.00000
C   38 R   33-    48.00000
C   97 OFSET     000.00000
C   97 R   33
C   97 R   33      1.0
       R   34
       R   34      0.00000
C    5 R   34      1.00000
C    6 R   34-     1.00000
C    7 R   34     12.00000
C    8 R   34-    12.00000
C    9 R   34      6.00000
C   10 R   34-     6.00000
C   15 R   34      2.00000
C   16 R   34-     2.00000
C   17 R   34     24.00000
C   18 R   34-    24.00000
C   19 R   34     12.00000
C   20 R   34-    12.00000
C   25 R   34      4.00000
C   26 R   34-     4.00000
C   27 R   34     48.00000
C   28 R   34-    48.00000
C   29 R   34     24.00000
C   30 R   34-    24.00000
C   35 R   34      8.00000
C   36 R   34-     8.00000
C   37 R   34     96.00000
C   38 R   34-    96.00000
C   39 R   34     48.00000
C   40 R   34-    48.00000
C   98 OFSET     000.00000
C   98 R   34
C   98 R   34      1.0
       R   35
       R   35      0.00000
```

```
C    5 R   35-      1.00000
C    6 R   35       1.00000
C   15 R   35-      3.00000
C   16 R   35       3.00000
C   25 R   35-      9.00000
C   26 R   35       9.00000
C   35 R   35-     27.00000
C   36 R   35      27.00000
C   45 R   35-      1.00000
C   46 R   35       1.00000
C   99 OFSET      000.00000
C   99 R   35
C   99 R   35       1.0
       R   36
       R   36       0.00000
C    5 R   36       1.00000
C    6 R   36-      1.00000
C    7 R   36       6.00000
C    8 R   36-      6.00000
C   15 R   36       3.00000
C   16 R   36-      3.00000
C   17 R   36      18.00000
C   18 R   36-     18.00000
C   25 R   36       9.00000
C   26 R   36-      9.00000
C   27 R   36      54.00000
C   28 R   36-     54.00000
C   35 R   36      27.00000
C   36 R   36-     27.00000
C   37 R   36     162.00000
C   38 R   36-    162.00000
C   45 R   36       1.00000
C   46 R   3(       1.00000
C   47 R   36       6.00000
C   48 R   36-      6.00000
C  100 OFSET      000.00000
C  100 R   36
C  100 R   36       1.0
       R   37
       R   37       0.00000
C    5 R   37       1.00000
```

1.0.0.2

```
C    6 R   37-      1.00000
C    7 R   37      12.00000
C    8 R   37-     12.00000
C    9 R   37       6.00000
C   10 R   37-      6.00000
C   15 R   37       3.00000
C   16 R   37-      3.00000
C   17 R   37      36.00000
C   18 R   37-     36.00000
C   19 R   37      18.00000
C   20 R   37-     18.00000
C   25 R   37       9.00000
C   26 R   37-      9.00000
C   27 R   37     108.00000
C   28 R   37-    108.00000
C   29 R   37      54.00000
C   30 R   37-     54.00000
C   35 R   37      27.00000
C   36 R   37-     27.00000
C   37 R   37     324.00000
C   38 R   37-    324.00000
C   39 R   37     162.00000
C   40 R   37-    162.00000
C   45 R   37       1.00000
C   46 R   37-      1.00000
C   47 R   37      12.00000
C   48 R   37-     12.00000
C   49 R   37       6.00000
C   50 R   37-      6.00000
C  101 OFSET      000.00000
C  101 R   37
C  101 R   37       1.0
       R   38
       R   38       0.00000
C    5 R   38       1.00000
C    6 R   38-      1.00000
C   15 R   38       4.00000
C   16 R   38-      4.00000
C   25 R   38      16.00000
C   26 R   38-     16.00000
C   35 R   38      64.00000
```

```
C  36 R   38-    64.00000
C  45 R   38      8.00000
C  46 R   38-     8.00000
C 102 OFSET     000.00000
C 102 R   38
C 102 R   38      1.0
      R   39
      R   39      0.00000
C   5 R   39      1.00000
C   6 R   39-     1.00000
C   7 R   39      6.00000
C   8 R   39-     6.00000
C  15 R   39      4.00000
C  16 R   39-     4.00000
C  17 R   39     24.00000
C  18 R   39-    24.00000
C  25 R   39     16.00000
C  26 R   39-    16.00000
C  27 R   39     96.00000
C  28 R   39-    96.00000
C  35 R   39     64.00000
C  36 R   39-    64.00000
C  37 R   39    384.00000
C  38 R   39-   384.00000
C  45 R   39      8.00000
C  46 R   39-     8.00000
C  47 R   39     48.00000
C  48 R   39-    48.00000
C 103 OFSET     000.00000
C 103 R   39
C 103 R   39      1.0
      R   40
      R   40      0.00000
C   5 R   40      1.00000
C   6 R   40-     1.00000
C   7 R   40     12.00000
C   8 R   40-    12.00000
C   9 R   40      6.00000
C  10 R   40-     6.00000
C  15 R   40      4.00000
C  16 R   40-     4.00000
```

```
C  17 R  40     48.00000
C  18 R  40-    48.00000
C  19 R  40     24.00000
C  20 R  40-    24.00000
C  25 R  40     16.00000
C  26 R  40-    16.00000
C  27 R  40    192.00000
C  28 R  40-   192.00000
C  29 R  40     96.00000
C  30 R  40-    96.00000
C  35 R  40     64.00000
C  36 R  40-    64.00000
C  37 R  40    768.00000
C  38 R  40-   768.00000
C  39 R  40    384.00000
C  40 R  40-   384.00000
C  45 R  40      8.00000
C  46 R  40-     8.00000
C  47 R  40     96.00000
C  48 R  40-    96.00000
C  49 R  40     48.00000
C  50 R  40-    48.00000
C 104 OFSET    000.00000
C 104 R  40
C 104 R  40      1.0
      R  42
      R  42    000.00000
C  21 R  42-     1.00000
C  22 R  42      1.00000
C  31 R  42-    12.00000
C  32 R  42     12.00000
C  41 R  42-     6.00000
C  42 R  42      6.00000
C 105 OFSET    000.00000
C 105 R  42
C 105 R  42      1.0
      R  43
      R  43    000.00000
C  21 R  43      1.00000
C  22 R  43-     1.00000
C  23 R  43      1.00000
```

1.0.0.2

```
C  24 R  43-     1.00000
C  25 R  43      1.00000
C  26 R  43-     1.00000
C  27 R  43      1.00000
C  28 R  43-     1.00000
C  31 R  43      6.00000
C  32 R  43-     6.00000
C  33 R  43      6.00000
C  34 R  43-     6.00000
C  35 R  43      6.00000
C  36 R  43-     6.00000
C  37 R  43      6.00000
C  38 R  43-     6.00000
C 106 OFSET    000.00000
C 106 R  43
C 106 R  43      1.0
      R  44
      R  44    000.00000
C  21 R  44      1.00000
C  22 R  44-     1.00000
C  23 R  44      1.00000
C  24 R  44-     1.00000
C  25 R  44      1.00000
C  26 R  44-     1.00000
C  27 R  44      1.00000
C  28 R  44-     1.00000
C  31 R  44     12.00000
C  32 R  44-    12.00000
C  33 R  44     12.00000
C  34 R  44-    12.00000
C  35 R  44     12.00000
C  36 R  44-    12.00000
C  37 R  44     12.00000
C  38 R  44-    12.00000
C  41 R  44      6.00000
C  42 R  44-     6.00000
C  43 R  44      6.00000
C  44 R  44-     6.00000
C  45 R  44      6.00000
C  46 R  44-     6.00000
C  47 R  44      6.00000
```

1.0.0-2

B-270

```
C  48 R  44-    6.00000
C 107 OFSET   000.00000
C 107 R  44
C 107 R  44     1.0
      R  45
      R  45   000.00000
C  21 R  45     1.00000
C  22 R  45-    1.00000
C  23 R  45     2.00000
C  24 R  45-    2.00000
C  25 R  45     4.00000
C  26 R  45-    4.00000
C  27 R  45     8.00000
C  28 R  45-    8.00000
C  31 R  45     6.00000
C  32 R  45-    6.00000
C  33 R  45    12.00000
C  34 R  45-   12.00000
C  35 R  45    24.00000
C  36 R  45-   24.00000
C  37 R  45    48.00000
C  38 R  45-   48.00000
C 108 OFSET   000.00000
C 108 R  45
C 108 R  45     1.0
      R  46
      R  46   000.00000
C  21 R  46     1.00000
C  22 R  46-    1.00000
C  23 R  46     2.00000
C  24 R  46-    2.00000
C  25 R  46     4.00000
C  26 R  46-    4.00000
C  27 R  46     8.00000
C  28 R  46-    8.00000
C  31 R  46    12.00000
C  32 R  46-   12.00000
C  33 R  46    24.00000
C  34 R  46-   24.00000
C  35 R  46    48.00000
C  36 R  46-   48.00000
```

1.0.0.~

1.0.0.2

```
C  37 R  46    96.00000
C  38 R  46-   96.00000
C  41 R  46     6.00000
C  42 R  46-    6.00000
C  43 R  46    12.00000
C  44 R  46-   12.00000
C  45 R  46    24.00000
C  46 R  46-   24.00000
C  47 R  46    48.00000
C  48 R  46-   48.00000
C 109 OFSET   000.00000
C 109 R  46
C 109 R  46     1.0
      R  47
      R  47   000.00000
C  21 R  47     1.00000
C  22 R  47-    1.00000
C  23 R  47     3.00000
C  24 R  47-    3.00000
C  25 R  47     9.00000
C  26 R  47-    9.00000
C  27 R  47    27.00000
C  28 R  47-   27.00000
C  29 R  47     1.00000
C  30 R  47-    1.00000
C  31 R  47     6.00000
C  32 R  47-    6.00000
C  33 R  47    18.00000
C  34 R  47-   18.00000
C  35 R  47    54.00000
C  36 R  47-   54.00000
C  37 R  47   162.00000
C  38 R  47-  162.00000
C  39 R  47     6.00000
C  40 R  47-    6.00000
C 110 OFSET   000.00000
C 110 R  47
C 110 R  47     1.0
      R  48
      R  48   000.00000
C  21 R  48     1.00000
```

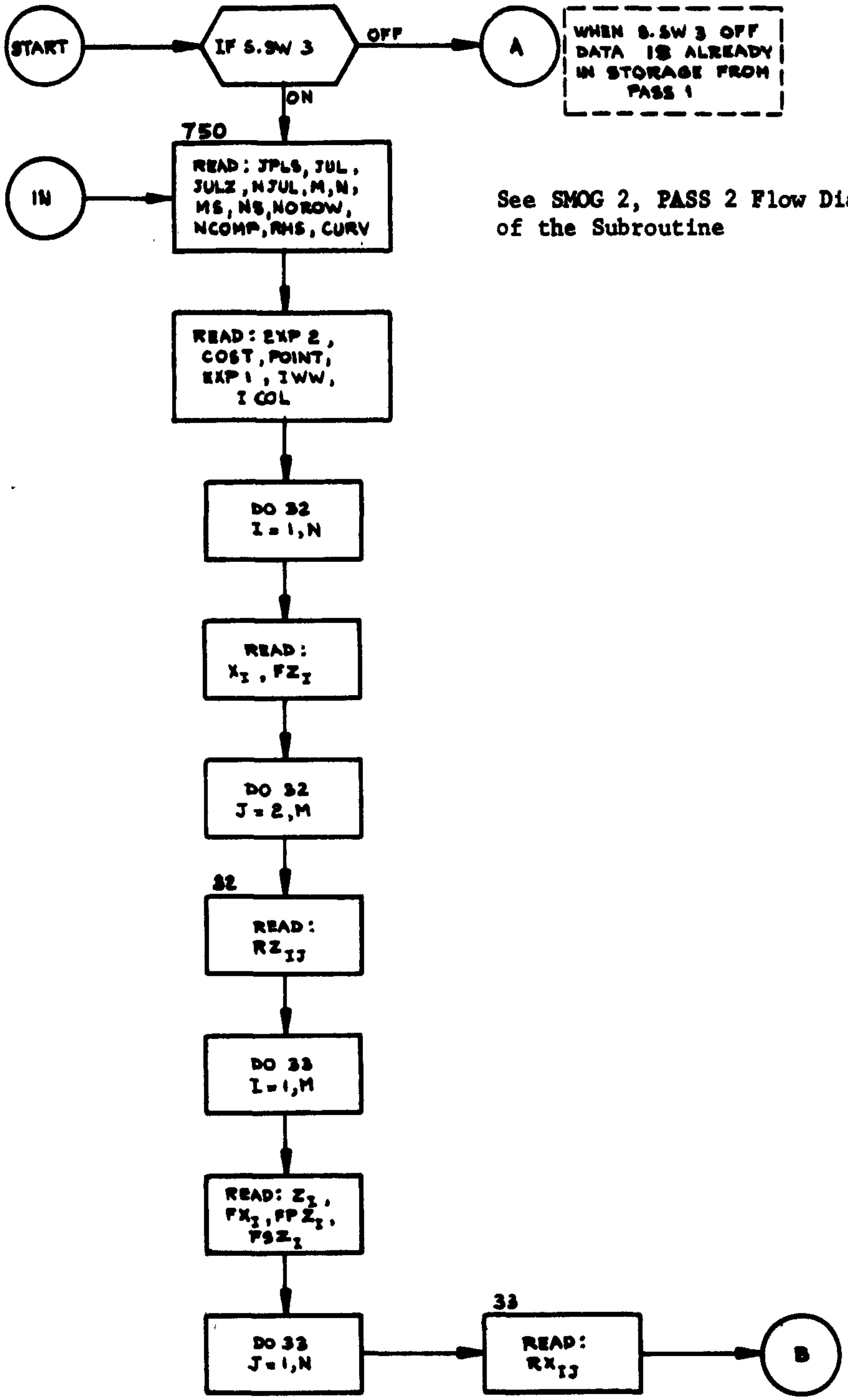
```
C  22 R  48-     1.00000
C  23 R  48      3.00000
C  24 R  48-     3.00000
C  25 R  48      9.00000
C  26 R  48-     9.00000
C  27 R  48     27.00000
C  28 R  48-    27.00000
C  29 R  48      1.00000
C  30 R  48-     1.00000
C  31 R  48     12.00000
C  32 R  48-    12.00000
C  33 R  48     36.00000
C  34 R  48-    36.00000
C  35 R  48    108.00000
C  36 R  48-   108.00000
C  37 R  48    324.00000
C  38 R  48-   324.00000
C  39 R  48     12.00000
C  40 R  48-    12.00000
C  41 R  48      6.00000
C  42 R  48-     6.00000
C  43 R  48     18.00000
C  44 R  48-    18.00000
C  45 R  48     54.00000
C  46 R  48-    54.00000
C  47 R  48    162.00000
C  48 R  48-   162.00000
C  49 R  48      6.00000
C  50 R  48-     6.00000
C 111 OFSET    000.00000
C 111 R  48
C 111 R  48      1.0
       R  49
       R  49   000.00000
C  21 R  49      1.00000
C  22 R  49-     1.00000
C  23 R  49      4.00000
C  24 R  49-     4.00000
C  25 R  49     16.00000
C  26 R  49-    16.00000
C  27 R  49     64.00000
```

1.0.0-2

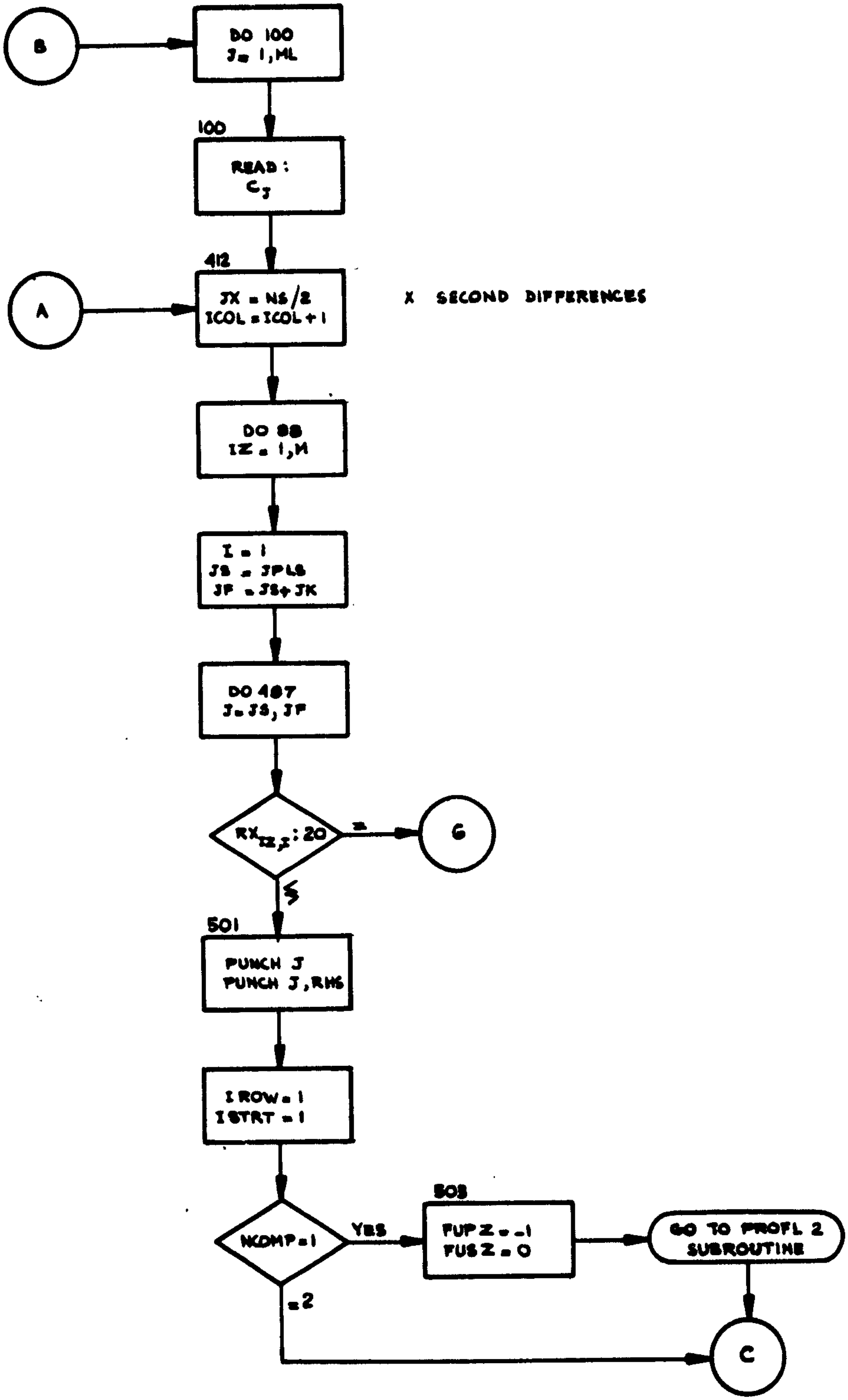
1.0.0-2

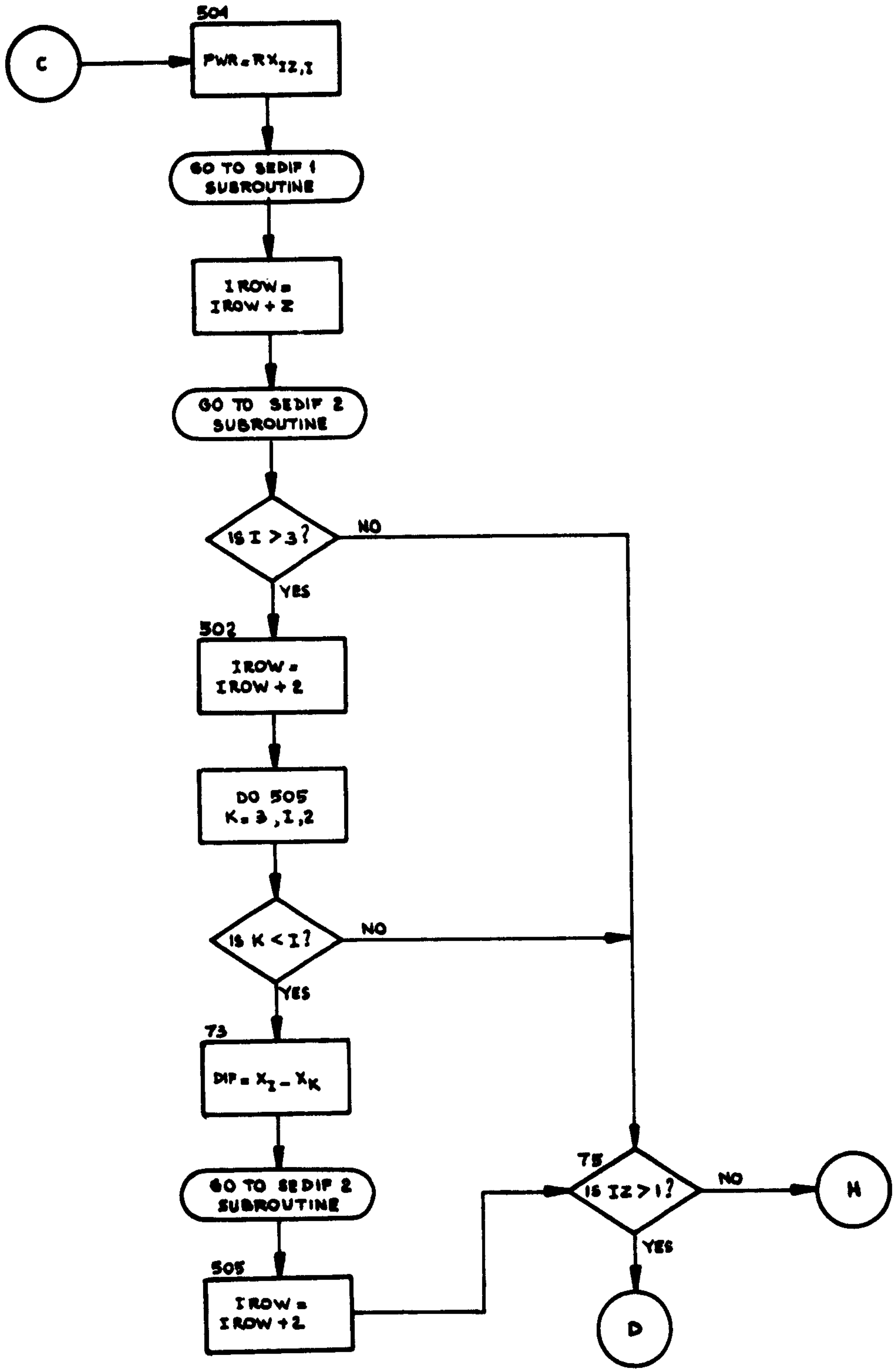
```
C  28 R  49-    64.00000
C  29 R  49      8.00000
C  30 R  49-     8.00000
C  31 R  49      6.00000
C  32 R  49-     6.00000
C  33 R  49     24.00000
C  34 R  49-    24.00000
C  35 R  49     96.00000
C  36 R  49-    96.00000
C  37 R  49    384.00000
C  38 R  49-   384.00000
C  39 R  49     48.00000
C  40 R  49-    48.00000
C 112 OFSET    000.00000
C 112 R  49
C 112 R  49      1.0
      R  50
      R  50    000.00000
C  21 R  50      1.00000
C  22 R  50-     1.00000
C  23 R  50      4.00000
C  24 R  50-     4.00000
C  25 R  50     16.00000
C  26 R  50-    16.00000
C  27 R  50     64.00000
C  28 R  50-    64.00000
C  29 R  50      8.00000
C  30 R  50-     8.00000
C  31 R  50     12.00000
C  32 R  50-    12.00000
C  33 R  50     48.00000
C  34 R  50-    48.00000
C  35 R  50    192.00000
C  36 R  50-   192.00000
C  37 R  50    768.00000
C  38 R  50-   768.00000
C  39 R  50     96.00000
C  40 R  50-    96.00000
C  41 R  50      6.00000
C  42 R  50-     6.00000
C  43 R  50     24.00000
C  44 R  50-    24.00000
C  45 R  50     96.00000
C  46 R  50-    96.00000
C  47 R  50    384.00000
C  48 R  50-   384.00000
C  49 R  50     48.00000
C  50 R  50-    48.00000
C 113 OFSET    000.00000
C 113 R  50
C 113 R  50      1.0
FIRST B
EOF
```

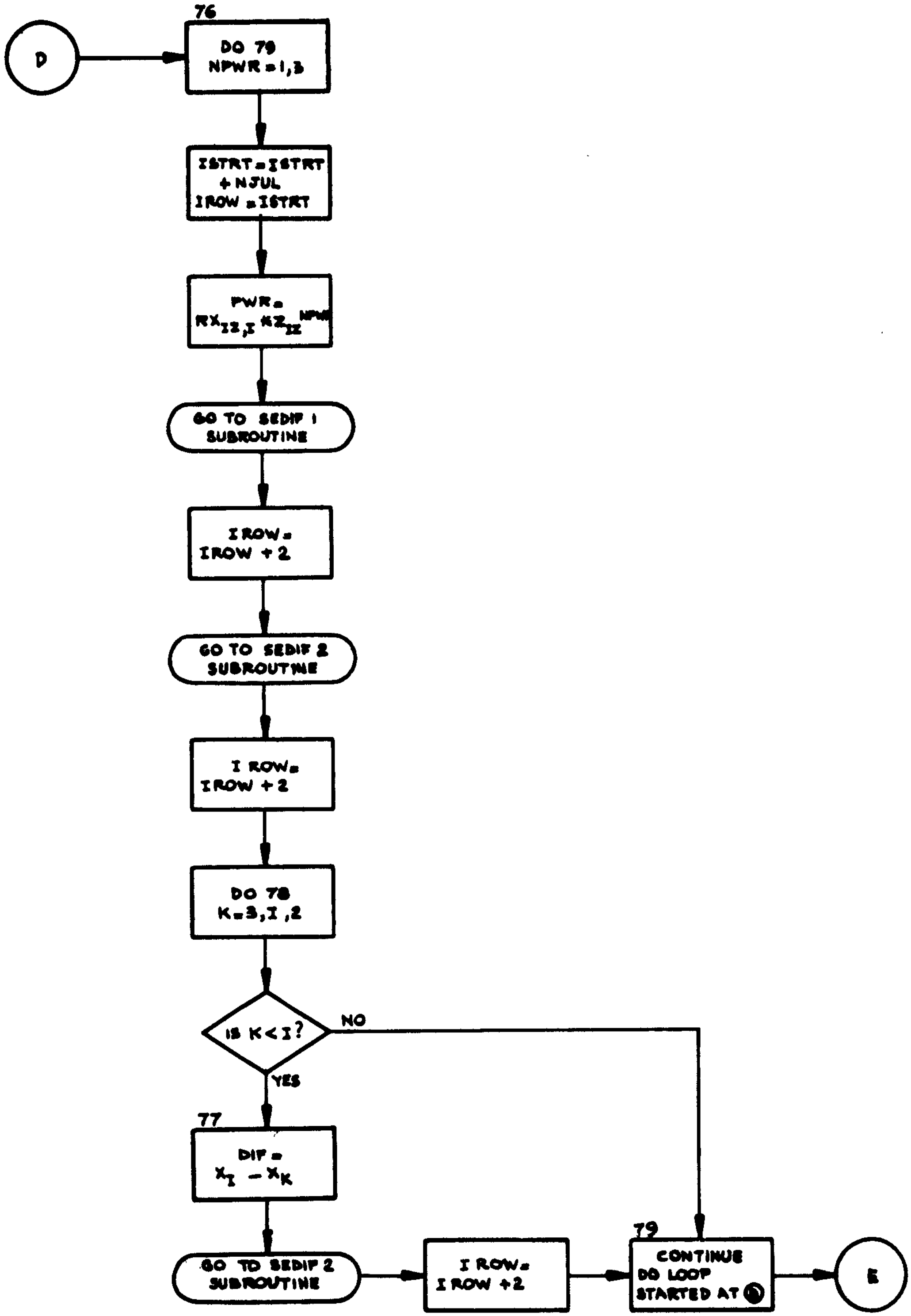
FLOW DIAGRAM - SMOG 3, PASS 2

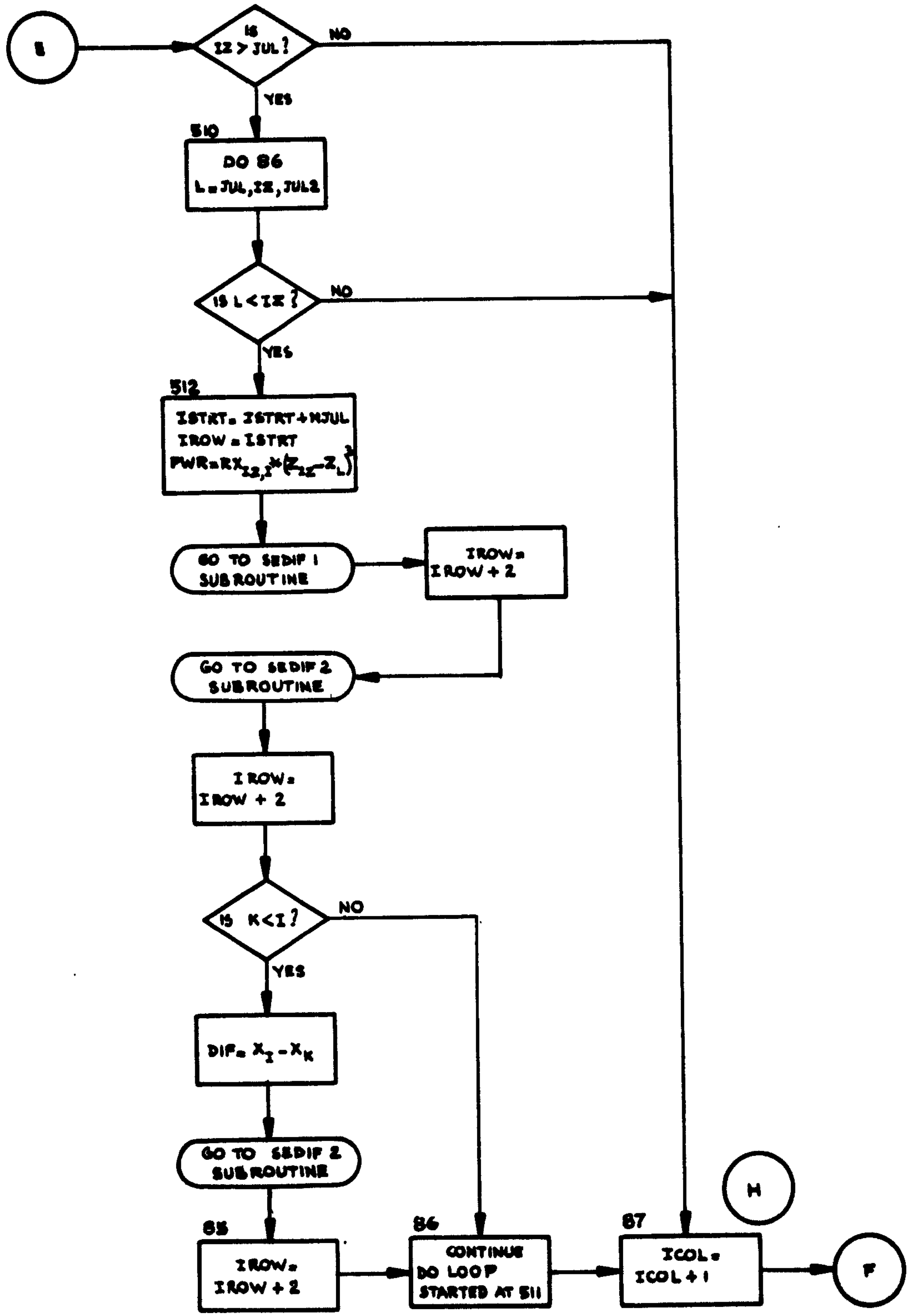


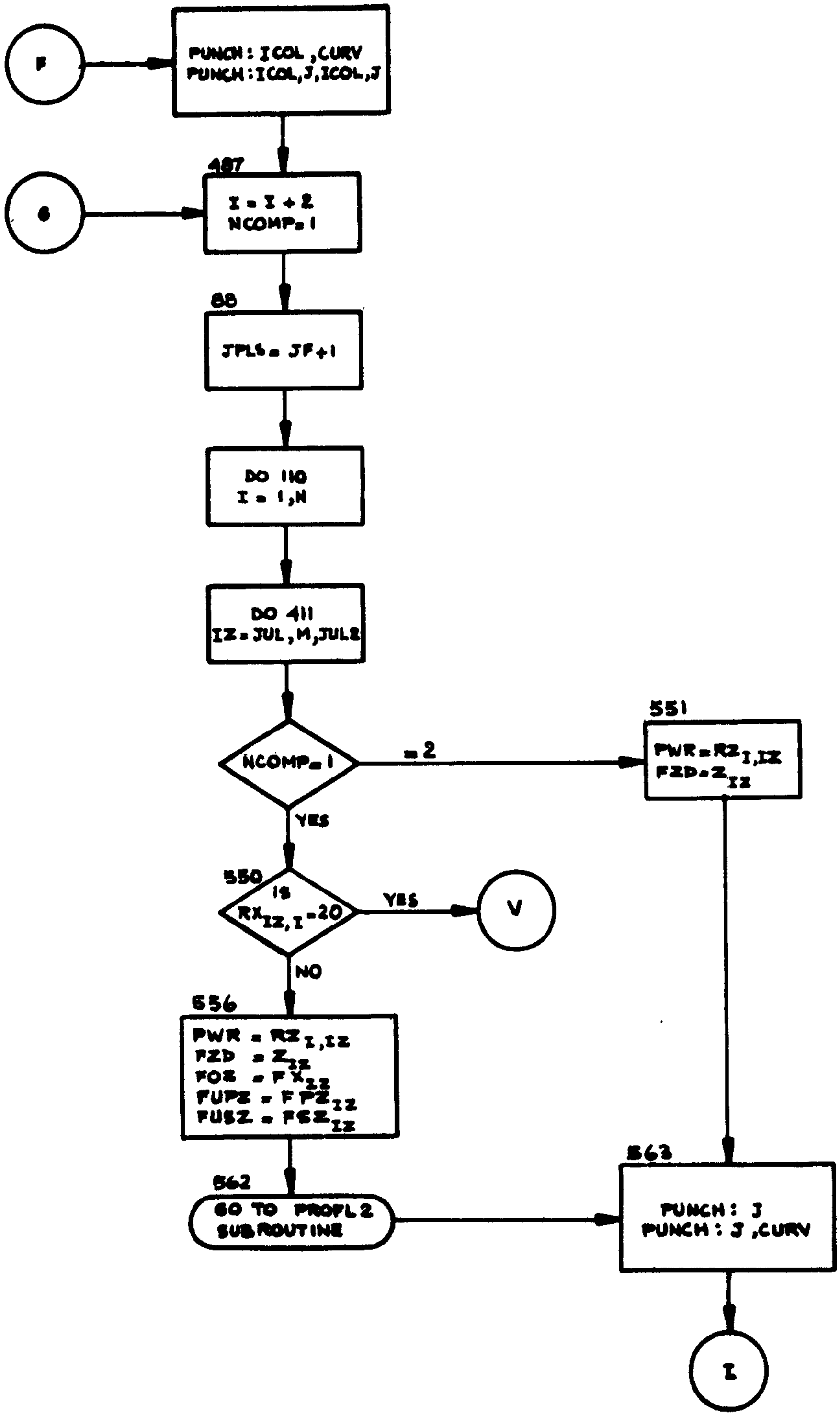
See SMOG 2, PASS 2 Flow Diagrams of the Subroutine

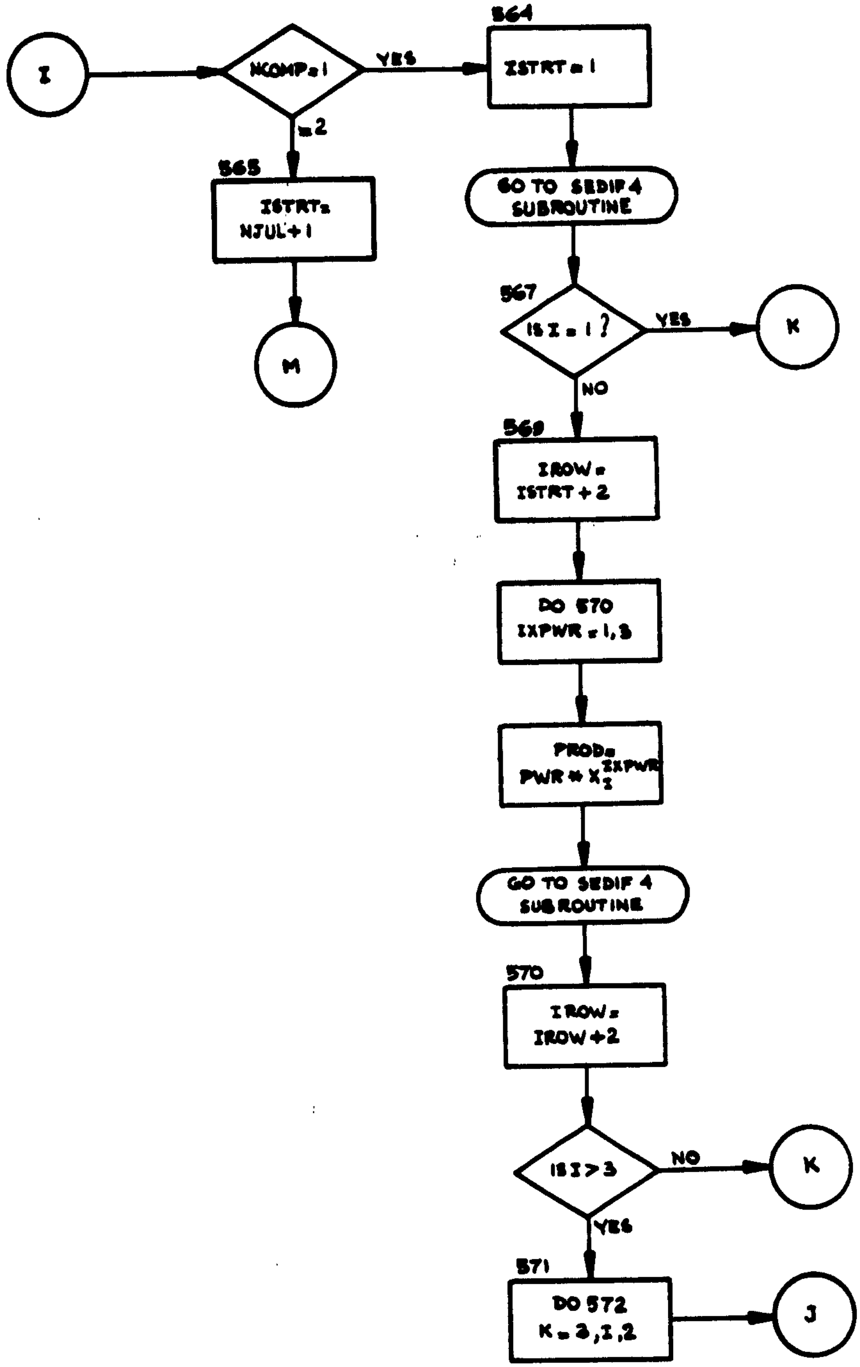

B
DO 100
J= 1,ML
100
READ:
C_J
412
A
JX = NS/2
ICOL = ICOL + 1
X SECOND DIFFERENCES
DO 88
IZ = 1,M
I = 1
JS = JPLS
JF = JS+JK
DO 487
J = JS, JF
$RX_{IZ,J}$: 20
=
G
≶
501
PUNCH J
PUNCH J, RHS
IROW = 1
ISTRT = 1
NCOMP = 1
YES
503
FUPZ = -1
FUSZ = 0
GO TO PROFL 2
SUBROUTINE
=2
C

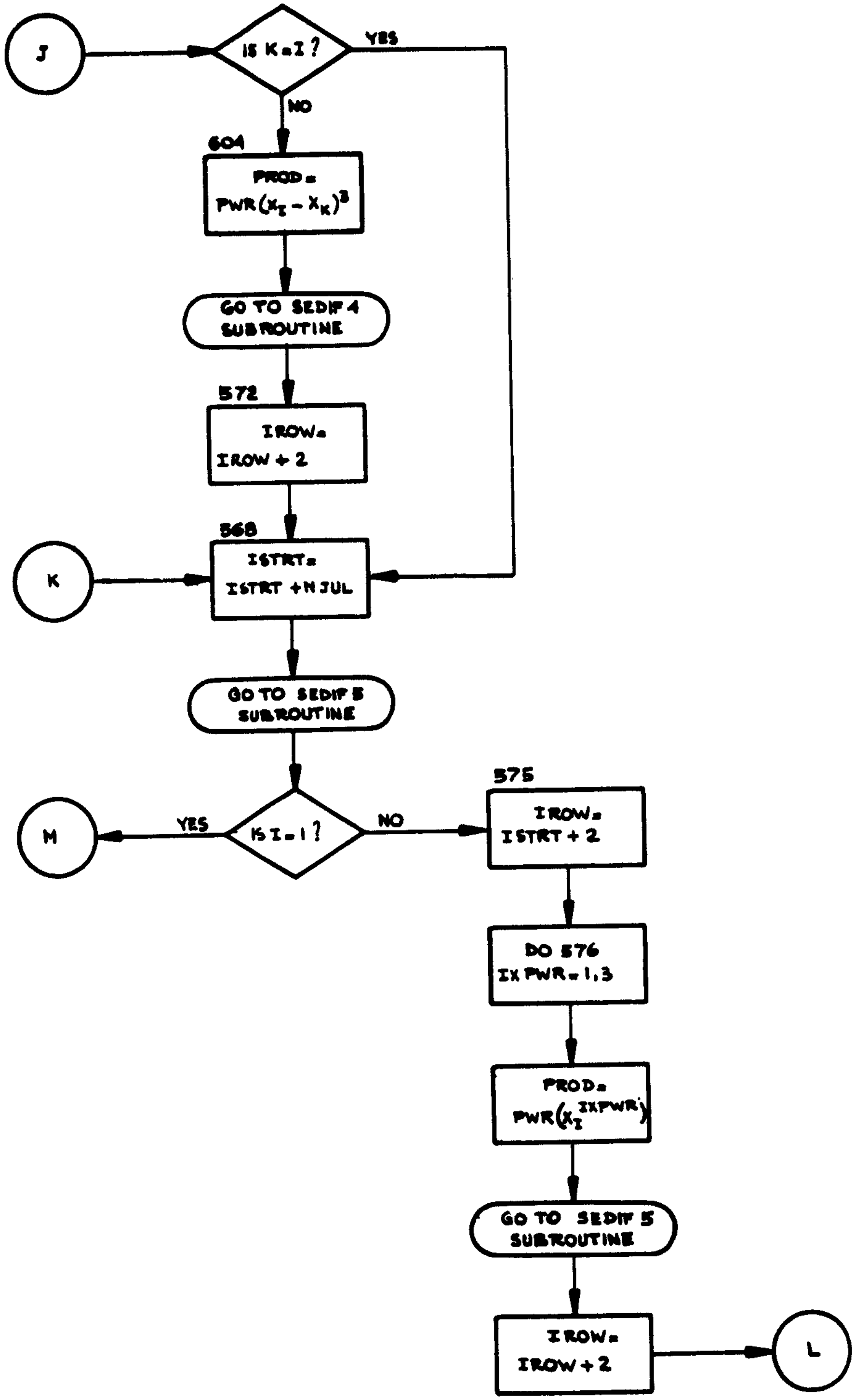

C
504
PWR = RX_IZ,I
GO TO SEDIF 1 SUBROUTINE
IROW = IROW + 2
GO TO SEDIF 2 SUBROUTINE
IS I > 3?
NO
YES
502
IROW = IROW + 2
DO 505 K = 3, I, 2
IS K < I?
NO
YES
73
DIF = X_I - X_K
GO TO SEDIF 2 SUBROUTINE
505
IROW = IROW + 2
75
IS IZ > 1?
NO
YES
H
D

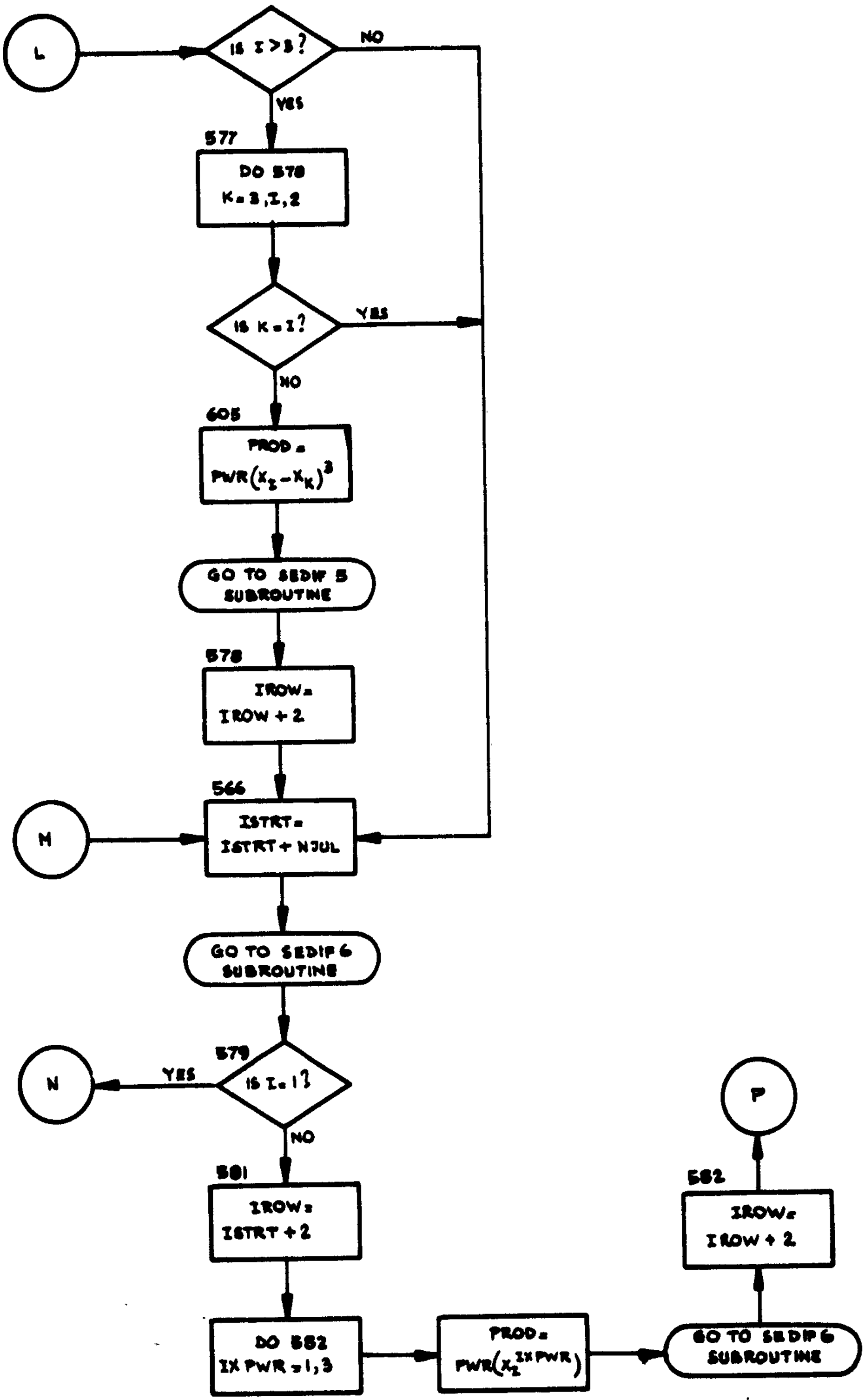

D
76
DO 79
NPWR = 1,3
ISTRT = ISTRT + NJUL
IROW = ISTRT
PWR = RX$_{II,I}$ *Z$_{II}$^{NPW}
GO TO SEDIF 1 SUBROUTINE
IROW = IROW + 2
GO TO SEDIF 2 SUBROUTINE
I ROW = IROW + 2
DO 78
K = 3, I, 2
IS K < I?
NO
YES
77
DIF = X$_I$ − X$_K$
GO TO SEDIF 2 SUBROUTINE
I ROW = I ROW + 2
79
CONTINUE DO LOOP STARTED AT
E

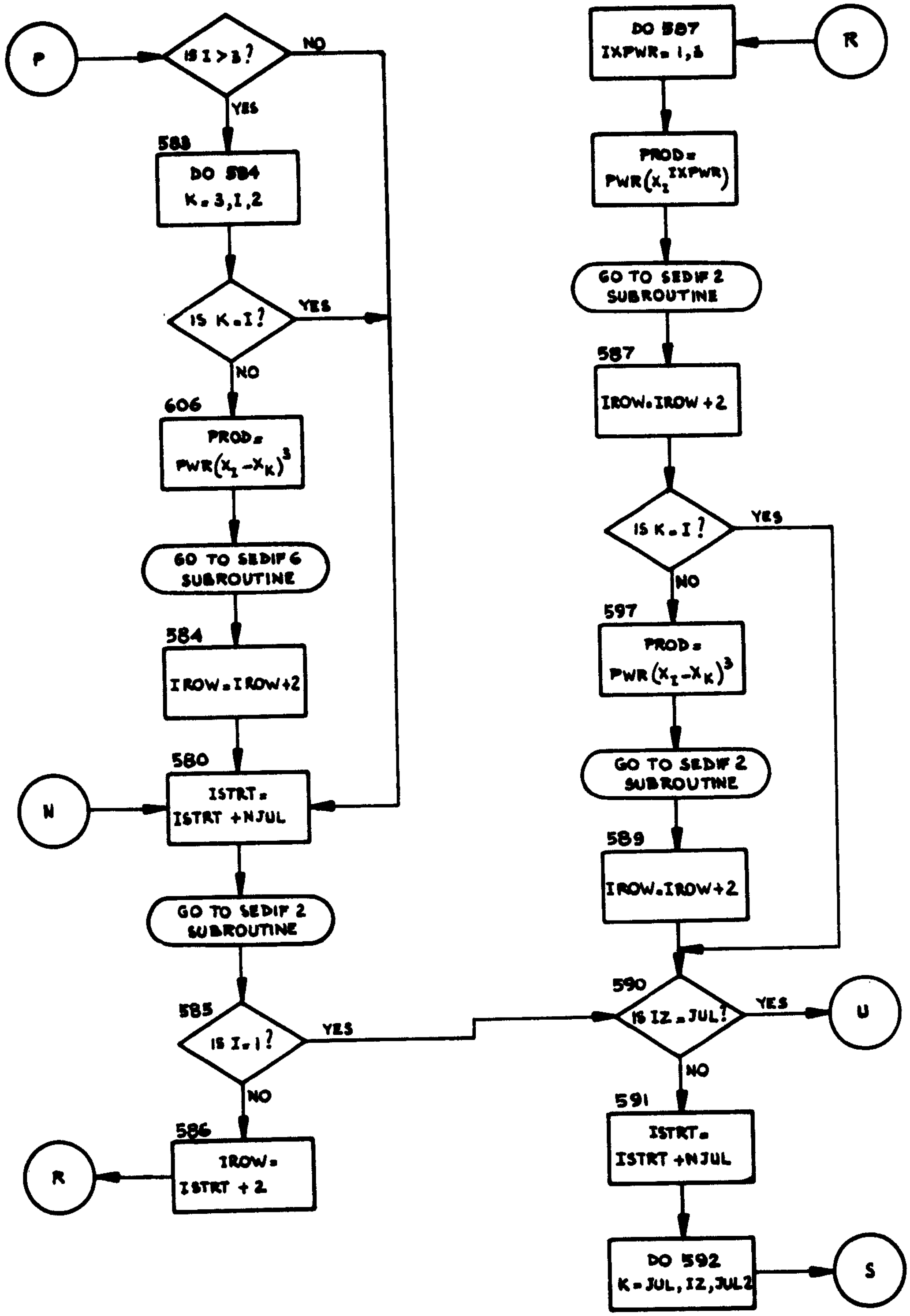

E
IS IZ > JUL ?
NO
YES
510
DO 86
L = JUL, IZ, JUL2
IS L < IZ ?
NO
YES
512
ISTRT = ISTRT + NJUL
IROW = ISTRT
PWR = RX_{IZ,I} * (Z_{IZ} - Z_L)^3
GO TO SEDIF 1 SUBROUTINE
IROW = IROW + 2
GO TO SEDIF 2 SUBROUTINE
IROW = IROW + 2
IS K < I ?
NO
YES
DIF = X_I - X_K
GO TO SEDIF 2 SUBROUTINE
85
IROW = IROW + 2
86
CONTINUE DO LOOP STARTED AT 511
87
ICOL = ICOL + 1
H
F

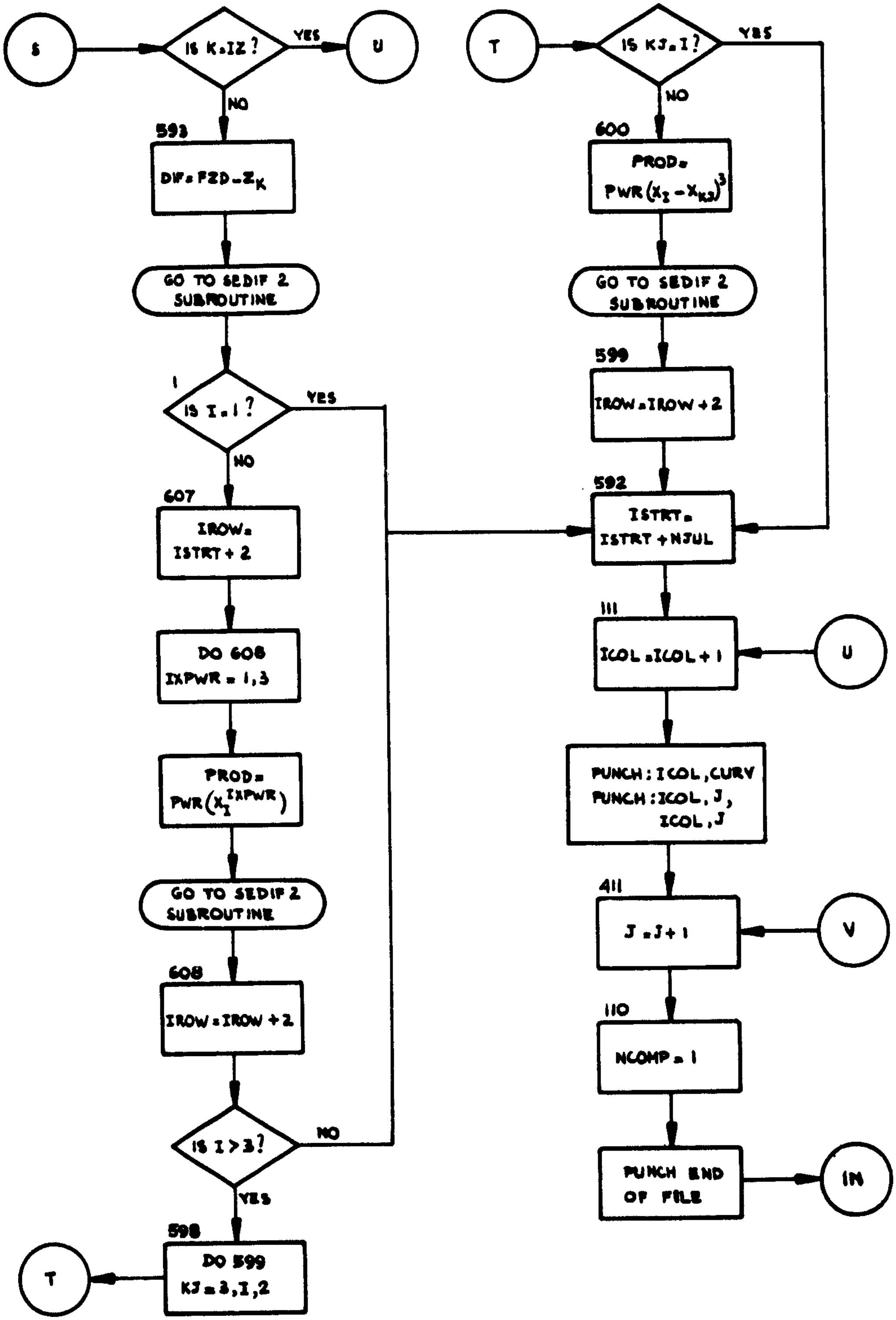

F
PUNCH: ICOL, CURV
PUNCH: ICOL, J, ICOL, J
G
487
I = I + 2
NCOMP = 1
88
JPL6 = JF+1
DO 110
I = 1,N
DO 411
IZ = JUL, M, JUL2
NCOMP = 1
= 2
551
PWR = RZ I,IZ
FZD = Z IZ
YES
550
IS
RX IZ,I = 20
YES
V
NO
556
PWR = RZ I,IZ
FZD = Z IZ
FOZ = FX IZ
FUPZ = FPZ IZ
FUSZ = FSZ IZ
562
GO TO PROFL2
SUBROUTINE
563
PUNCH: J
PUNCH: J, CURV
I


I
NCOMP=1
YES
=2
564
ISTRT=1
565
ISTRT=
NJUL+1
M
GO TO SEDIF 4
SUBROUTINE
567
IS I = 1 ?
YES
K
NO
569
IROW =
ISTRT + 2
DO 570
IXPWR = 1, 3
PROD=
PWR * X IXPWR 2
GO TO SEDIF 4
SUBROUTINE
570
IROW =
IROW + 2
IS I > 3
NO
K
YES
571
DO 572
K = 3, I, 2
J


J
IS K=I ?
YES
NO
604
PROD=
PWR(X_I - X_K)^3
GO TO SEDIF 4
SUBROUTINE
572
IROW=
IROW + 2
568
K
ISTRT=
ISTRT + NJUL
GO TO SEDIF 5
SUBROUTINE
M
YES
IS I=1 ?
NO
575
IROW=
ISTRT + 2
DO 576
IXPWR=1,3
PROD=
PWR(X_I^IXPWR)
GO TO SEDIF 5
SUBROUTINE
IROW=
IROW + 2
L


L
IS I > 3 ?
NO
YES
577
DO 578
K = 3, I, 2
IS K = I ?
YES
NO
605
PROD =
PWR $(X_I - X_K)^3$
GO TO SEDIF 5
SUBROUTINE
578
IROW =
IROW + 2
566
M
ISTRT =
ISTRT + NJUL
GO TO SEDIF 6
SUBROUTINE
579
N
YES
IS I = 1 ?
NO
581
IROW =
ISTRT + 2
DO 582
IX PWR = 1, 3
PROD =
PWR (X_I^{IXPWR})
GO TO SEDIF 6
SUBROUTINE
582
IROW =
IROW + 2
P


P
IS I > 3?
NO
YES
583
DO 584
K=3,I,2
IS K=I?
YES
NO
606
PROD=
PWR(X_I - X_K)^3
GO TO SEDIF 6
SUBROUTINE
584
IROW=IROW+2
580
N
ISTRT=
ISTRT +NJUL
GO TO SEDIF 2
SUBROUTINE
585
IS I=1?
YES
NO
586
IROW=
ISTRT + 2
R
DO 587
IXPWR= 1,2
R
PROD=
PWR(X_I^IXPWR)
GO TO SEDIF 2
SUBROUTINE
587
IROW=IROW +2
IS K=I?
YES
NO
597
PROD=
PWR(X_I - X_K)^3
GO TO SEDIF 2
SUBROUTINE
589
IROW=IROW+2
590
IS IZ=JUL?
YES
U
NO
591
ISTRT=
ISTRT +NJUL
DO 592
K=JUL,IZ,JUL2
S


S
IS K=IZ ?
YES
U
NO
593
DIF=FZD-Z_K
GO TO SEDIF 2 SUBROUTINE
1
IS I=1 ?
YES
NO
607
IROW= ISTRT+2
DO 608 IXPWR = 1,3
PROD= PWR(X_I^IXPWR)
GO TO SEDIF 2 SUBROUTINE
608
IROW=IROW+2
IS I>3?
NO
YES
598
DO 599 KJ=3,I,2
T
T
IS KJ=I?
YES
NO
600
PROD= PWR(X_I - X_KJ)^3
GO TO SEDIF 2 SUBROUTINE
599
IROW=IROW+2
592
ISTRT= ISTRT+NJUL
111
ICOL=ICOL+1
U
PUNCH: ICOL,CURV
PUNCH: ICOL, J, ICOL, J
411
J=J+1
V
110
NCOMP=1
PUNCH END OF FILE
IN

```
C     SMOG3 PASS 2 PRIMAL SUM OF DEV  X OR X AND Z DBL SPL WI PRFL
      DIMENSION RX(11,15),RZ(15,11),FZ(11),FX(15),X(15),Z(11),Y(11,
15)
      DIMENSION  FPZ(11),FSZ(11),C(13)
      COMMON JPLS,JUL,JUL2,NJUL,M,N,MS,NS,NOROW,NCOMP,RHS,COST
      COMMON EXP2,CURV ,POINT,EXP1,IWW,ICOL
      COMMON X,FZ,RZ,Z,FX,FPZ,FSZ,RX,C
      IF(SENSE SWITCH 3)750,412
  750 READ 180,JPLS,JUL,JUL2,NJUL,M,N,MS,NS,NOROW,NCOMP,RHS,CURV
      READ 181,EXP2,COST,POINT,EXP1,IWW,ICOL
      DO 32 I=1,N
      READ 181,X(I),FZ(I)
      DO 32 J=2,M
   32 READ 181,RZ(I,J)
      DO 33 I=1,M
      READ 181,Z(I),FX(I),FPZ(I),FSZ(I)
      DO 33 J=1,N
   33 READ 181,RX(I,J)
      ML=M+2
      DO 100 J=1,ML
  100 READ 181,C(J)
C        *    *    X SEC  DIFFERENCES   *    *
  412 JX=NS/2
      ICOL=ICOL+1            RHS = CURV
      DO 88 IZ=1,M
      I=1
      JS=JPLS
      JF=JS+JX
      DO 487 J=JS,JF
      IF(RX(IZ,I)-20.)501,487,501
  501 PUNCH 162,J
      PUNCH 177,J,RHS
      IROW=1
      ISTRT=1
      GO TO (503,504),NCOMP
  503 FUPZ=-1.
      FUSZ=0.
      CALL PROFL2 (X(I),FX(IZ),EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS
,NCOMP)
  504 PWR=RX(IZ,I)
      CALL SEDIF1 (T,TP,TS,X(I),J,IROW,PWR)
      IROW=IROW+2
      CALL SEDIF2 (X(I),T,TP,TS,PWR,J,IROW)
      IF(I-3)75,75,502
  502 IROW=IROW+2
```

```
      DO 505 K=3,I,2
      IF(K-I)73,75,75
   73 DIF=X(I)-X(K)
      CALL SEDIF2 (DIF,T,TP,TS,PWR,J,IROW)
  505 IROW=IROW+2
   75 IF(IZ-1)87,87,76
   76 DO 79 NPWR=1,3
      ISTRT=ISTRT+NJUL
      IROW=ISTRT
      PWR=RX(IZ,I)*Z(IZ)**NPWR
      CALL SEDIF1 (T,TP,TS,X(I),J,IROW,PWR)
      IROW=IROW+2
      CALL SEDIF2 (X(I),T,TP,TS,PWR,J,IROW)
      IROW=IROW+2
      DO 78 K=3,I,2
      IF(K-I)77,79,79
   77 DIF=X(I)-X(K)
      CALL SEDIF2 (DIF,T,TP,TS,PWR,J,IROW)
   78 IROW=IROW+2
   79 CONTINUE
      IF(IZ-JUL)87,87,511
  511 DO 86 L=JUL,IZ,JUL2
      IF(L-IZ)512,87,87
  512 ISTRT=ISTRT+NJUL
      IROW=ISTRT
      PWR=RX(IZ,I)*(Z(IZ)-Z(L))**3
      CALL SEDIF1 (T,TP,TS,X(I),J,IROW,PWR)
      IROW=IROW+2
      CALL SEDIF2 (X(I),T,TP,TS,PWR,J,IROW)
      IROW=IROW+2
      DO 85 K=3,I,2
      IF(K-I)84,86,86
   84 DIF=X(I)-X(K)
      CALL SEDIF2 (DIF,T,TP,TS,PWR,J,IROW)
   85 IROW=IROW+2
   86 CONTINUE
   87 ICOL=ICOL+1
      PUNCH 171,ICOL,CURV
      PUNCH 152,ICOL,J,ICOL,J
  487 I=I+2
      NCOMP=1
   88 JPLS=JF+1
C         Z  SEC  DIFFERENCES    *    *
      DO 110 I=1,N
      DO 411 IZ=JUL,M,JUL2
```

```
      GO TO (550,551),NCOMP
  550 IF(RX(IZ,I)-20.)556,411,556
  556 PWR=RZ(I,IZ)
      FZD=Z(IZ)
      FOZ=FX(IZ)
      FUPZ=FPZ(IZ)
      FUSZ=FSZ(IZ)
  562 CALL PROFL2(X(I),FOZ,EXP1,FUPZ,FUSZ,EXP2,POINT,T,TP,TS,
NCOMP)
      GO TO 563
  551 PWR=RZ(I,IZ)
      FZD=Z(IZ)
  563 PUNCH 162,J
      PUNCH 177,J,CURV
      GO TO (564,565),NCOMP
  565 ISTRT=1+NJUL
      GO TO 566
  564 ISTRT=1
      CALL SEDIF4 (TS,PWR,J,ISTRT)
  567 IF(I-1)569,568,569
  569 IROW=ISTRT+2
      DO 570 IXPWR=1,3
      PROD=PWR*X(I)**IXPWR
      CALL SEDIF4(TS,PROD,J,IROW)
  570 IROW=IROW+2
      IF(I-3)568,568,571
  571 DO 572 K=3,I,2
      IF(K-I)604,568,604
  604 PROD=PWR*(X(I)-X(K))**3
      CALL SEDIF4(TS,PROD,J,IROW)
  572 IROW=IROW+2
  568 ISTRT=ISTRT+NJUL
      CALL SEDIF5(TP,TS,FZD,PWR,J,ISTRT)
  573 IF(I-1)575,566,575
  575 IROW=ISTRT+2
      DO 576 IXPWR=1,3
      PROD=PWR*X(I)**IXPWR
      CALL SEDIF5 (TP,TS,FZD,PROD,J,IROW)
  576 IROW=IROW+2
      IF(I-3)566,566,577
  577 DO 578 K=3,I,2
      IF(K-I)605,566,605
  605 PROD=PWR*(X(I)-X(K))**3
      CALL SEDIF5 (TP,TS,FZD,PROD,J,IROW)
  578 IROW=IROW+2
```

```
566 ISTRT=ISTRT+NJUL
    CALL SEDIF6 (T,TP,TS,FZD,J,PWR,ISTRT)
579 IF(I-1)581,580,581
581 IROW=ISTRT+2
    DO 582 IXPWR=1,3
    PROD=PWR*X(I)**IXPWR
    CALL SEDIF6 (T,TP,TS,FZD,J,PROD,IROW)
582 IROW=IROW+2
    IF(I-3)580,580,583
583 DO 584 K=3,I,2
    IF(K-I)606,580,606
606 PROD=PWR*(X(I)-X(K))**3
    CALL SEDIF6 (T,TP,TS,FZD,J,PROD,IROW)
584 IROW=IROW+2
580 ISTRT=ISTRT+NJUL
    CALL SEDIF2 (FZD,T,TP,TS,PWR,J,ISTRT)
585 IF(I-1)586,590,586
586 IROW=ISTRT+2
    DO 587 IXPWR=1,3
    PROD=PWR*X(I)**IXPWR
    CALL SEDIF2 (FZD,T,TP,TS,PROD,J,IROW)
587 IROW=IROW+2
    IF(I-3)590,590,588
588 DO 589 K=3,I,2
    IF(K-I)597,590,597
597 PROD=PWR*(X(I)-X(K))**3
    CALL SEDIF2 (FZD,T,TP,TS,PROD,J,IROW)
589 IROW=IROW+2
590 IF(IZ-JUL)591,111,591
591 ISTRT=ISTRT+NJUL
    DO 592 K=JUL,IZ,JUL2
    IF(K-IZ)593,111,593
593 DIF=FZD-Z(K)
    CALL SEDIF2 (DIF,T,TP,TS,PWR,J,ISTRT)
  1 IF(I-1)607,592,607
607 IROW=ISTRT+2
    DO 608 IXPWR=1,3
    PROD=PWR*X(I)**IXPWR
    CALL SEDIF2 (DIF,T,TP,TS,PROD,J,IROW)
608 IROW=IROW+2
    IF(I-3)592,592,598
598 DO 599 KJ=3,I,2
    IF(KJ-I)600,592,600
600 PROD=PWR*(X(I)-X(KJ))**3
    CALL SEDIF2 (DIF,T,TP,TS,PROD,J,IROW)
```

```
599 IROW=IROW+2
592 ISTRT=ISTRT+NJUL
111 ICOL=ICOL+1
    PUNCH 171,ICOL,CURV
    PUNCH 152,ICOL,J,ICOL,J
411 J=J+1
110 NCOMP=1
    PUNCH 174
    PUNCH 176
    PRINT 178
    PAUSE 751
    GO TO 750
152 FORMAT(7X,1HC,I4,2H R,I4/7X,1HC,I4,2H R,I4,8H      1.0)
171 FORMAT(6X,2H C,I4,6H OFSET,F12.5)
173 FORMAT(1H*,5X,47H
 )
174 FORMAT(7HFIRST B)
175 FORMAT(12X, 6H R   1,F12.5)
176 FORMAT(3HEOF)
177 FORMAT(12X,2H R,I4,1H ,F11.5)
178 FORMAT(28HENTER NEW DATA - PRESS START)
180 FORMAT (10I5,2F15.9)
162 FORMAT(12X,2H R,I4)
181 FORMAT (4F15.9,I4,I4)
    END
```


B-290

```
SUBROUTINE SEDIF1 (TD,TPD,TSD,XD,JD,IROWD,PWRD)
CALL SEDIF4 (TSD,PWRD,JD,IROWD)
IROWD=IROWD+2
CALL SEDIF5 (TPD,TSD,XD,PWRD,JD,IROWD)
IROWD=IROWD+2
CALL SEDIF6 (TD,TPD,TSD,XD,JD,PWRD,IROWD)
RETURN
END


SUBROUTINE SEDIF2 (DIFD,TD,TPD,TSD,PWRD,JD,IROWD)
WRM=(3.*DIFD*TD+3.*TPD*DIFD**2+(TSD*DIFD**3)/2.)*PWRD
CALL SIGN (JD,IROWD,WRM)
RETURN
END


SUBROUTINE SEDIF3 (IXD,FXD,PRD,JSD,IROWD)
EXD=IXD
WRM=(EXD*FXD**(IXD-1)-FXD**IXD)*PRD
CALL SIGN (JSD,IROWD,WRM)
RETURN
END


SUBROUTINE SEDIF4 (TSD,PWRD,JD,IROWD)
WRM=(TSD*PWRD)/2.
CALL SIGN (JD,IROWD,WRM)
RETURN
END


SUBROUTINE SEDIF5 (TPD,TSD,XD,PWRD,JD,IROWD)
WRM=((TSD*XD)/2.+TPD)*PWRD
CALL SIGN (JD,IROWD,WRM)
RETURN
END
```

```
      SUBROUTINE SEDIF6 (TD,TPD,TSD,XD,JD,PWRD,IROWD)
      WRM=TD
      WRM=WRM+(TSD*XD**2)/2.
      WRM=WRM+2.*TPD*XD
      WRM=WRM*PWRD
      CALL SIGN (JD,IROWD,WRM)
      RETURN
      END


      SUBROUTINE SIGN (IW,JW,WAM)
      JWPL=JW+1
      IF(WAM)370,373,371
  370 WAM=-WAM
      PUNCH 169,JW,IW,WAM,JWPL,IW,WAM
      RETURN
  371 PUNCH 167,JW,IW,WAM,JWPL,IW,WAM
  373 RETURN
  167 FORMAT(6X,2H C,I4,2H R,I4,1H-,F11.5/6X,2H C,I4,2H R,I4,F12.5)
  169 FORMAT(6X,2H C,I4,2H R,I4,F12.5/6X,2H C,I4,2H R,I4,1H-,F11.5)
      END
```

* TODD DLG 5X5 FND PROFILE X-Z DBL SPL-12/3/62

ROW ID

OFSET	
R	3
R	4
R	5
R	7
R	8
R	9
R	10
R	12
R	13
R	14
R	15
R	16
R	17
R	18
R	19
R	20
R	21
R	22
R	23
R	24
R	25
R	27
R	28
R	30
R	31
R	33
R	34
R	35
R	36
R	37
R	38
R	39
R	40
R	42
R	43
R	44
R	45
R	46

1.0.0-2

```
          R  47
          R  48
          R  49
          R  50
BASIS
    C  51 R   3
    C  53 R   4
    C  55 R   5
    C  57 R   7
    C  59 R   8
    C  61 R   9
    C  63 R  10
    C  65 R  12
    C  67 R  13
    C  69 R  14
    C  71 R  15
    C  73 R  16
    C  75 R  17
    C  77 R  18
    C  79 R  19
    C  81 R  20
    C  83 R  21
    C  85 R  22
    C  87 R  23
    C  89 R  24
    C  91 R  25
    C  93 R  27
    C  94 R  28
    C  95 R  30
    C  96 R  31
    C  97 R  33
    C  98 R  34
    C  99 R  35
    C 100 R  36
    C 101 R  37
    C 102 R  38
    C 103 R  39
    C 104 R  40
    C 105 R  42
    C 106 R  43
    C 107 R  44
```

1.0.0-2

```
      C 108 R   45
      C 109 R   46
      C 110 R   47
      C 111 R   48
      C 112 R   49
      C 113 R   50
MATRIX
      C   1 OFSET       0.00100
      C   1 R    3-     1.0
      C   1 R    4-     1.0
      C   1 R    5-     1.0
      C   1 R    7-     1.0
      C   1 R    8-     1.0
      C   1 R    9-     1.0
      C   1 R   10-     1.0
      C   1 R   12-     1.0
      C   1 R   13-     1.0
      C   1 R   14-     1.0
      C   1 R   15-     1.0
      C   1 R   16-     1.0
      C   1 R   17-     1.0
      C   1 R   18-     1.0
      C   1 R   19-     1.0
      C   1 R   20-     1.0
      C   1 R   21-     1.0
      C   1 R   22-     1.0
      C   1 R   23-     1.0
      C   1 R   24-     1.0
      C   1 R   25-     1.0
      C   2 OFSET       0.00100
      C   2 R    3      1.0
      C   2 R    4      1.0
      C   2 R    5      1.0
      C   2 R    7      1.0
      C   2 R    8      1.0
      C   2 R    9      1.0
      C   2 R   10      1.0
      C   2 R   12      1.0
      C   2 R   13      1.0
      C   2 R   14      1.0
      C   2 R   15      1.0
```

1.0.0-2

```
C    2 R   16      1.0
C    2 R   17      1.0
C    2 R   18      1.0
C    2 R   19      1.0
C    2 R   20      1.0
C    2 R   21      1.0
C    2 R   22      1.0
C    2 R   23      1.0
C    2 R   24      1.0
C    2 R   25      1.0
C    3 OFSET       0.00100
C    3 R    3-     2.00000
C    3 R    4-     3.00000
C    3 R    5-     4.00000
C    3 R    7-     1.00000
C    3 R    8-     2.00000
C    3 R    9-     3.00000
C    3 R   10-     4.00000
C    3 R   12-     1.00000
C    3 R   13-     2.00000
C    3 R   14-     3.00000
C    3 R   15-     4.00000
C    3 R   17-     1.00000
C    3 R   18-     2.00000
C    3 R   19-     3.00000
C    3 R   20-     4.00000
C    3 R   22-     1.00000
C    3 R   23-     2.00000
C    3 R   24-     3.00000
C    3 R   25-     4.00000
C    4 OFSET       0.00100
C    4 R    3      2.00000
C    4 R    4      3.00000
C    4 R    5      4.00000
C    4 R    7      1.00000
C    4 R    8      2.00000
C    4 R    9      3.00000
C    4 R   10      4.00000
C    4 R   12      1.00000
C    4 R   13      2.00000
C    4 R   14      3.00000
```

```
C    4 R   15        4.00000
C    4 R   17        1.00000
C    4 R   18        2.00000
C    4 R   19        3.00000
C    4 R   20        4.00000
C    4 R   22        1.00000
C    4 R   23        2.00000
C    4 R   24        3.00000
C    4 R   25        4.00000
C    5 OFSET         0.00100
C    5 R    3-       4.00000
C    5 R    4-       9.00000
C    5 R    5-      16.00000
C    5 R    7-       1.00000
C    5 R    8-       4.00000
C    5 R    9-       9.00000
C    5 R   10-      16.00000
C    5 R   12-       1.00000
C    5 R   13-       4.00000
C    5 R   14-       9.00000
C    5 R   15-      16.00000
C    5 R   17-       1.00000
C    5 R   18-       4.00000
C    5 R   19-       9.00000
C    5 R   20-      16.00000
C    5 R   22-       1.00000
C    5 R   23-       4.00000
C    5 R   24-       9.00000
C    5 R   25-      16.00000
C    5 R   27-       1.00000
C    5 R   28        1.00000
C    5 R   30        1.00000
C    5 R   31        1.00000
C    5 R   33        1.00000
C    5 R   34        1.00000
C    5 R   35-       1.00000
C    5 R   36        1.00000
C    5 R   37        1.00000
C    5 R   38        1.00000
C    5 R   39        1.00000
C    5 R   40        1.00000
```

1.0.0-2

```
C    6 OFSET      0.00100
C    6 R    3     4.00000
C    6 R    4     9.00000
C    6 R    5    16.00000
C    6 R    7     1.00000
C    6 R    8     4.00000
C    6 R    9     9.00000
C    6 R   10    16.00000
C    6 R   12     1.00000
C    6 R   13     4.00000
C    6 R   14     9.00000
C    6 R   15    16.00000
C    6 R   17     1.00000
C    6 R   18     4.00000
C    6 R   19     9.00000
C    6 R   20    16.00000
C    6 R   22     1.00000
C    6 R   23     4.00000
C    6 R   24     9.00000
C    6 R   25    16.00000
C    6 R   27     1.00000
C    6 R   28-    1.00000
C    6 R   30-    1.00000
C    6 R   31-    1.00000
C    6 R   33-    1.00000
C    6 R   34-    1.00000
C    6 R   35     1.00000
C    6 R   36-    1.00000
C    6 R   37-    1.00000
C    6 R   38-    1.00000
C    6 R   39-    1.00000
C    6 R   40-    1.00000
C    7 OFSET      0.00100
C    7 R    3-    8.00000
C    7 R    4-   27.00000
C    7 R    5-   64.00000
C    7 R    7-    1.00000
C    7 R    8-    8.00000
C    7 R    9-   27.00000
C    7 R   10-   64.00000
C    7 R   12-    1.00000
```

1.0.0-2

```
C     7 R   13-     8.00000
C     7 R   14-    27.00000
C     7 R   15-    64.00000
C     7 R   17-     1.00000
C     7 R   18-     8.00000
C     7 R   19-    27.00000
C     7 R   20-    64.00000
C     7 R   22-     1.00000
C     7 R   23-     8.00000
C     7 R   24-    27.00000
C     7 R   25-    64.00000
C     7 R   27-     6.00000
C     7 R   28     12.00000
C     7 R   30      6.00000
C     7 R   31     12.00000
C     7 R   33      6.00000
C     7 R   34     12.00000
C     7 R   36      6.00000
C     7 R   37     12.00000
C     7 R   39      6.00000
C     7 R   40     12.00000
C     8 OFSET       0.00100
C     8 R    3      8.00000
C     8 R    4     27.00000
C     8 R    5     64.00000
C     8 R    7      1.00000
C     8 R    8      8.00000
C     8 R    9     27.00000
C     8 R   10     64.00000
C     8 R   12      1.00000
C     8 R   13      8.00000
C     8 R   14     27.00000
C     8 R   15     64.00000
C     8 R   17      1.00000
C     8 R   18      8.00000
C     8 R   19     27.00000
C     8 R   20     64.00000
C     8 R   22      1.00000
C     8 R   23      8.00000
C     8 R   24     27.00000
C     8 R   25     64.00000
```

1.0.0.2

C	8	R	27	6.00000
C	8	R	28-	12.00000
C	8	R	30-	6.00000
C	8	R	31-	12.00000
C	8	R	33-	6.00000
C	8	R	34-	12.00000
C	8	R	36-	6.00000
C	8	R	37-	12.00000
C	8	R	39-	6.00000
C	8	R	40-	12.00000
C	9	OFSET		0.00100
C	9	R	4-	1.00000
C	9	R	5-	8.00000
C	9	R	9-	1.00000
C	9	R	10-	8.00000
C	9	R	14-	1.00000
C	9	R	15-	8.00000
C	9	R	19-	1.00000
C	9	R	20-	8.00000
C	9	R	24-	1.00000
C	9	R	25-	8.00000
C	9	R	28	6.00000
C	9	R	31	6.00000
C	9	R	34	6.00000
C	9	R	37	6.00000
C	9	R	40	6.00000
C	10	OFSET		0.00100
C	10	R	4	1.00000
C	10	R	5	8.00000
C	10	R	9	1.00000
C	10	R	10	8.00000
C	10	R	14	1.00000
C	10	R	15	8.00000
C	10	R	19	1.00000
C	10	R	20	8.00000
C	10	R	24	1.00000
C	10	R	25	8.00000
C	10	R	28-	6.00000
C	10	R	31-	6.00000
C	10	R	34-	6.00000
C	10	R	37-	6.00000

```
C   10 R   40-     6.00000
C   11 OFSET       0.00100
C   11 R    7-     1.00000
C   11 R    8-     1.00000
C   11 R    9-     1.00000
C   11 R   10-     1.00000
C   11 R   12-     2.00000
C   11 R   13-     2.00000
C   11 R   14-     2.00000
C   11 R   15-     2.00000
C   11 R   16-     3.00000
C   11 R   17-     3.00000
C   11 R   18-     3.00000
C   11 R   19-     3.00000
C   11 R   20-     3.00000
C   11 R   21-     4.00000
C   11 R   22-     4.00000
C   11 R   23-     4.00000
C   11 R   24-     4.00000
C   11 R   25-     4.00000
C   12 OFSET       0.00100
C   12 R    7      1.00000
C   12 R    8      1.00000
C   12 R    9      1.00000
C   12 R   10      1.00000
C   12 R   12      2.00000
C   12 R   13      2.00000
C   12 R   14      2.00000
C   12 R   15      2.00000
C   12 R   16      3.00000
C   12 R   17      3.00000
C   12 R   18      3.00000
C   12 R   19      3.00000
C   12 R   20      3.00000
C   12 R   21      4.00000
C   12 R   22      4.00000
C   12 R   23      4.00000
C   12 R   24      4.00000
C   12 R   25      4.00000
C   13 OFSET       0.00100
C   13 R    7-     1.00000
```

```
C   13 R    8-    2.00000
C   13 R    9-    3.00000
C   13 R   10-    4.00000
C   13 R   12-    2.00000
C   13 R   13-    4.00000
C   13 R   14-    6.00000
C   13 R   15-    8.00000
C   13 R   17-    3.00000
C   13 R   18-    6.00000
C   13 R   19-    9.00000
C   13 R   20-   12.00000
C   13 R   22-    4.00000
C   13 R   23-    8.00000
C   13 R   24-   12.00000
C   13 R   25-   16.00000
C   14 OFSET      0.00100
C   14 R    7     1.00000
C   14 R    8     2.00000
C   14 R    9     3.00000
C   14 R   10     4.00000
C   14 R   12     2.00000
C   14 R   13     4.00000
C   14 R   14     6.00000
C   14 R   15     8.00000
C   14 R   17     3.00000
C   14 R   18     6.00000
C   14 R   19     9.00000
C   14 R   20    12.00000
C   14 R   22     4.00000
C   14 R   23     8.00000
C   14 R   24    12.00000
C   14 R   25    16.00000
C   15 OFSET      0.00100
C   15 R    7-    1.00000
C   15 R    8-    4.00000
C   15 R    9-    9.00000
C   15 R   10-   16.00000
C   15 R   12-    2.00000
C   15 R   13-    8.00000
C   15 R   14-   18.00000
C   15 R   15-   32.00000
```

C	15	R	17-	3.00000
C	15	R	18-	12.00000
C	15	R	19-	27.00000
C	15	R	20-	48.00000
C	15	R	22-	4.00000
C	15	R	23-	16.00000
C	15	R	24-	36.00000
C	15	R	25-	64.00000
C	15	R	30	1.00000
C	15	R	31	1.00000
C	15	R	33	2.00000
C	15	R	34	2.00000
C	15	R	35-	3.00000
C	15	R	36	3.00000
C	15	R	37	3.00000
C	15	R	38	4.00000
C	15	R	39	4.00000
C	15	R	40	4.00000
C	16	OFSET		0.00100
C	16	R	7	1.00000
C	16	R	8	4.00000
C	16	R	9	9.00000
C	16	R	10	16.00000
C	16	R	12	2.00000
C	16	R	13	8.00000
C	16	R	14	18.00000
C	16	R	15	32.00000
C	16	R	17	3.00000
C	16	R	18	12.00000
C	16	R	19	27.00000
C	16	R	20	48.00000
C	16	R	22	4.00000
C	16	R	23	16.00000
C	16	R	24	36.00000
C	16	R	25	64.00000
C	16	R	30-	1.00000
C	16	R	31-	1.00000
C	16	R	33-	2.00000
C	16	R	34-	2.00000
C	16	R	35	3.00000
C	16	R	36-	3.00000

1.0.0-2

```
C  16 R   37-      3.00000
C  16 R   38-      4.00000
C  16 R   39-      4.00000
C  16 R   40-      4.00000
C  17 OFSET        0.00100
C  17 R    7-      1.00000
C  17 R    8-      8.00000
C  17 R    9-     27.00000
C  17 R   10-     64.00000
C  17 R   12-      2.00000
C  17 R   13-     16.00000
C  17 R   14-     54.00000
C  17 R   15-    128.00000
C  17 R   17-      3.00000
C  17 R   18-     24.00000
C  17 R   19-     81.00000
C  17 R   20-    192.00000
C  17 R   22-      4.00000
C  17 R   23-     32.00000
C  17 R   24-    108.00000
C  17 R   25-    256.00000
C  17 R   30       6.00000
C  17 R   31      12.00000
C  17 R   33      12.00000
C  17 R   34      24.00000
C  17 R   36      18.00000
C  17 R   37      36.00000
C  17 R   39      24.00000
C  17 R   40      48.00000
C  18 OFSET        0.00100
C  18 R    7       1.00000
C  18 R    8       8.00000
C  18 R    9      27.00000
C  18 R   10      64.00000
C  18 R   12       2.00000
C  18 R   13      16.00000
C  18 R   14      54.00000
C  18 R   15     128.00000
C  18 R   17       3.00000
C  18 R   18      24.00000
C  18 R   19      81.00000
```

```
C   18 R    20    192.00000
C   18 R    22      4.00000
C   18 R    23     32.00000
C   18 R    24    108.00000
C   18 R    25    256.00000
C   18 R    30-     6.00000
C   18 R    31-    12.00000
C   18 R    33-    12.00000
C   18 R    34-    24.00000
C   18 R    36-    18.00000
C   18 R    37-    36.00000
C   18 R    39-    24.00000
C   18 R    40-    48.00000
C   19 OFSET        0.00100
C   19 R     9-     1.00000
C   19 R    10-     8.00000
C   19 R    14-     2.00000
C   19 R    15-    16.00000
C   19 R    19-     3.00000
C   19 R    20-    24.00000
C   19 R    24-     4.00000
C   19 R    25-    32.00000
C   19 R    31      6.00000
C   19 R    34     12.00000
C   19 R    37     18.00000
C   19 R    40     24.00000
C   20 OFSET        0.00100
C   20 R     9      1.00000
C   20 R    10      8.00000
C   20 R    14      2.00000
C   20 R    15     16.00000
C   20 R    19      3.00000
C   20 R    20     24.00000
C   20 R    24      4.00000
C   20 R    25     32.00000
C   20 R    31-     6.00000
C   20 R    34-    12.00000
C   20 R    37-    18.00000
C   20 R    40-    24.00000
C   21 OFSET        0.00100
C   21 R     7-     1.00000
```

```
C   21 R    8-     1.00000
C   21 R    9-     1.00000
C   21 R   10-     1.00000
C   21 R   12-     4.00000
C   21 R   13-     4.00000
C   21 R   14-     4.00000
C   21 R   15-     4.00000
C   21 R   16-     9.00000
C   21 R   17-     9.00000
C   21 R   18-     9.00000
C   21 R   19-     9.00000
C   21 R   20-     9.00000
C   21 R   21-    16.00000
C   21 R   22-    16.00000
C   21 R   23-    16.00000
C   21 R   24-    16.00000
C   21 R   25-    16.00000
C   21 R   42-     1.00000
C   21 R   43      1.00000
C   21 R   44      1.00000
C   21 R   45      1.00000
C   21 R   46      1.00000
C   21 R   47      1.00000
C   21 R   48      1.00000
C   21 R   49      1.00000
C   21 R   50      1.00000
C   22 OFSET       0.00100
C   22 R    7      1.00000
C   22 R    8      1.00000
C   22 R    9      1.00000
C   22 R   10      1.00000
C   22 R   12      4.00000
C   22 R   13      4.00000
C   22 R   14      4.00000
C   22 R   15      4.00000
C   22 R   16      9.00000
C   22 R   17      9.00000
C   22 R   18      9.00000
C   22 R   19      9.00000
C   22 R   20      9.00000
C   22 R   21     16.00000
```

```
C   22 R   22     16.00000
C   22 R   23     16.00000
C   22 R   24     16.00000
C   22 R   25     16.00000
C   22 R   42      1.00000
C   22 R   43-     1.00000
C   22 R   44-     1.00000
C   22 R   45-     1.00000
C   22 R   46-     1.00000
C   22 R   47-     1.00000
C   22 R   48-     1.00000
C   22 R   49-     1.00000
C   22 R   50-     1.00000
C   23 OFSET       0.00100
C   23 R    7-     1.00000
C   23 R    8-     2.00000
C   23 R    9-     3.00000
C   23 R   10-     4.00000
C   23 R   12-     4.00000
C   23 R   13-     8.00000
C   23 R   14-    12.00000
C   23 R   15-    16.00000
C   23 R   17-     9.00000
C   23 R   18-    18.00000
C   23 R   19-    27.00000
C   23 R   20-    36.00000
C   23 R   22-    16.00000
C   23 R   23-    32.00000
C   23 R   24-    48.00000
C   23 R   25-    64.00000
C   23 R   43      1.00000
C   23 R   44      1.00000
C   23 R   45      2.00000
C   23 R   46      2.00000
C   23 R   47      3.00000
C   23 R   48      3.00000
C   23 R   49      4.00000
C   23 R   50      4.00000
C   24 OFSET       0.00100
C   24 R    7      1.00000
C   24 R    8      2.00000
```

C	24	R	9	3.00000
C	24	R	10	4.00000
C	24	R	12	4.00000
C	24	R	13	8.00000
C	24	R	14	12.00000
C	24	R	15	16.00000
C	24	R	17	9.00000
C	24	R	18	18.00000
C	24	R	19	27.00000
C	24	R	20	36.00000
C	24	R	22	16.00000
C	24	R	23	32.00000
C	24	R	24	48.00000
C	24	R	25	64.00000
C	24	R	43-	1.00000
C	24	R	44-	1.00000
C	24	R	45-	2.00000
C	24	R	46-	2.00000
C	24	R	47-	3.00000
C	24	R	48-	3.00000
C	24	R	49-	4.00000
C	24	R	50-	4.00000
C	25	OFSET		0.00100
C	25	R	7-	1.00000
C	25	R	8-	4.00000
C	25	R	9-	9.00000
C	25	R	10-	16.00000
C	25	R	12-	4.00000
C	25	R	13-	16.00000
C	25	R	14-	36.00000
C	25	R	15-	64.00000
C	25	R	17-	9.00000
C	25	R	18-	36.00000
C	25	R	19-	81.00000
C	25	R	20-	144.00000
C	25	R	22-	16.00000
C	25	R	23-	64.00000
C	25	R	24-	144.00000
C	25	R	25-	256.00000
C	25	R	30	1.00000
C	25	R	31	1.00000

1.0.0-2

```
C  25 R   33      4.00000
C  25 R   34      4.00000
C  25 R   35-     9.00000
C  25 R   36      9.00000
C  25 R   37      9.00000
C  25 R   38     16.00000
C  25 R   39     16.00000
C  25 R   40     16.00000
C  25 R   43      1.00000
C  25 R   44      1.00000
C  25 R   45      4.00000
C  25 R   46      4.00000
C  25 R   47      9.00000
C  25 R   48      9.00000
C  25 R   49     16.00000
C  25 R   50     16.00000
C  26 OFSET       0.00100
C  26 R    7      1.00000
C  26 R    8      4.00000
C  26 R    9      9.00000
C  26 R   10     16.00000
C  26 R   12      4.00000
C  26 R   13     16.00000
C  26 R   14     36.00000
C  26 R   15     64.00000
C  26 R   17      9.00000
C  26 R   18     36.00000
C  26 R   19     81.00000
C  26 R   20    144.00000
C  26 R   22     16.00000
C  26 R   23     64.00000
C  26 R   24    144.00000
C  26 R   25    256.00000
C  26 R   30-     1.00000
C  26 R   31-     1.00000
C  26 R   33-     4.00000
C  26 R   34-     4.00000
C  26 R   35      9.00000
C  26 R   36-     9.00000
C  26 R   37-     9.00000
C  26 R   38-    16.00000
```

1.0.0-2

```
C   26 R   39-    16.00000
C   26 R   40-    16.00000
C   26 R   43-     1.00000
C   26 R   44-     1.00000
C   26 R   45-     4.00000
C   26 R   46-     4.00000
C   26 R   47-     9.00000
C   26 R   48-     9.00000
C   26 R   49-    16.00000
C   26 R   50-    16.00000
C   27 OFSET       0.00100
C   27 R    7-     1.00000
C   27 R    8-     8.00000
C   27 R    9-    27.00000
C   27 R   10-    64.00000
C   27 R   12-     4.00000
C   27 R   13-    32.00000
C   27 R   14-   108.00000
C   27 R   15-   256.00000
C   27 R   17-     9.00000
C   27 R   18-    72.00000
C   27 R   19-   243.00000
C   27 R   20-   576.00000
C   27 R   22-    16.00000
C   27 R   23-   128.00000
C   27 R   24-   432.00000
C   27 R   25-  1024.00000
C   27 R   30      6.00000
C   27 R   31     12.00000
C   27 R   33     24.00000
C   27 R   34     48.00000
C   27 R   36     54.00000
C   27 R   37    108.00000
C   27 R   39     96.00000
C   27 R   40    192.00000
C   27 R   43      1.00000
C   27 R   44      1.00000
C   27 R   45      8.00000
C   27 R   46      8.00000
C   27 R   47     27.00000
C   27 R   48     27.00000
```

```
C   27 R   49     64.00000
C   27 R   50     64.00000
C   28 OFSET       0.00100
C   28 R    7      1.00000
C   28 R    8      8.00000
C   28 R    9     27.00000
C   28 R   10     64.00000
C   28 R   12      4.00000
C   28 R   13     32.00000
C   28 R   14    108.00000
C   28 R   15    256.00000
C   28 R   17      9.00000
C   28 R   18     72.00000
C   28 R   19    243.00000
C   28 R   20    576.00000
C   28 R   22     16.00000
C   28 R   23    128.00000
C   28 R   24    432.00000
C   28 R   25   1024.00000
C   28 R   30-     6.00000
C   28 R   31-    12.00000
C   28 R   33-    24.00000
C   28 R   34-    48.00000
C   28 R   36-    54.00000
C   28 R   37-   108.00000
C   28 R   39-    96.00000
C   28 R   40-   192.00000
C   28 R   43-     1.00000
C   28 R   44-     1.00000
C   28 R   45-     8.00000
C   28 R   46-     8.00000
C   28 R   47-    27.00000
C   28 R   48-    27.00000
C   28 R   49-    64.00000
C   28 R   50-    64.00000
C   29 OFSET       0.00100
C   29 R    9-     1.00000
C   29 R   10-     8.00000
C   29 R   14-     4.00000
C   29 R   15-    32.00000
C   29 R   19-     9.00000
```

```
C   29 R    20-    72.00000
C   29 R    24-    16.00000
C   29 R    25-   128.00000
C   29 R    31      6.00000
C   29 R    34     24.00000
C   29 R    37     54.00000
C   29 R    40     96.00000
C   29 R    47      1.00000
C   29 R    48      1.00000
C   29 R    49      8.00000
C   29 R    50      8.00000
C   30 OFSET        0.00100
C   30 R     9      1.00000
C   30 R    10      8.00000
C   30 R    14      4.00000
C   30 R    15     32.00000
C   30 R    19      9.00000
C   30 R    20     72.00000
C   30 R    24     16.00000
C   30 R    25    128.00000
C   30 R    31-     6.00000
C   30 R    34-    24.00000
C   30 R    37-    54.00000
C   30 R    40-    96.00000
C   30 R    47-     1.00000
C   30 R    48-     1.00000
C   30 R    49-     8.00000
C   30 R    50-     8.00000
C   31 OFSET        0.00100
C   31 R     7-     1.00000
C   31 R     8-     1.00000
C   31 R     9-     1.00000
C   31 R    10-     1.00000
C   31 R    12-     8.00000
C   31 R    13-     8.00000
C   31 R    14-     8.00000
C   31 R    15-     8.00000
C   31 R    16-    27.00000
C   31 R    17-    27.00000
C   31 R    18-    27.00000
C   31 R    19-    27.00000
```

```
C   31 R   20-    27.00000
C   31 R   21-    64.00000
C   31 R   22-    64.00000
C   31 R   23-    64.00000
C   31 R   24-    64.00000
C   31 R   25-    64.00000
C   31 R   42-    12.00000
C   31 R   43      6.00000
C   31 R   44     12.00000
C   31 R   45      6.00000
C   31 R   46     12.00000
C   31 R   47      6.00000
C   31 R   48     12.00000
C   31 R   49      6.00000
C   31 R   50     12.00000
C   32 OFSET       0.00100
C   32 R    7      1.00000
C   32 R    8      1.00000
C   32 R    9      1.00000
C   32 R   10      1.00000
C   32 R   12      8.00000
C   32 R   13      8.00000
C   32 R   14      8.00000
C   32 R   15      8.00000
C   32 R   16     27.00000
C   32 R   17     27.00000
C   32 R   18     27.00000
C   32 R   19     27.00000
C   32 R   20     27.00000
C   32 R   21     64.00000
C   32 R   22     64.00000
C   32 R   23     64.00000
C   32 R   24     64.00000
C   32 R   25     64.00000
C   32 R   42     12.00000
C   32 R   43-     6.00000
C   32 R   44-    12.00000
C   32 R   45-     6.00000
C   32 R   46-    12.00000
C   32 R   47-     6.00000
C   32 R   48-    12.00000
```

C	32	R 49-	6.00000
C	32	R 50-	12.00000
C	33	OFSET	0.00100
C	33	R 7-	1.00000
C	33	R 8-	2.00000
C	33	R 9-	3.00000
C	33	R 10-	4.00000
C	33	R 12-	8.00000
C	33	R 13-	16.00000
C	33	R 14-	24.00000
C	33	R 15-	32.00000
C	33	R 17-	27.00000
C	33	R 18-	54.00000
C	33	R 19-	81.00000
C	33	R 20-	108.00000
C	33	R 22-	64.00000
C	33	R 23-	128.00000
C	33	R 24-	192.00000
C	33	R 25-	256.00000
C	33	R 43	6.00000
C	33	R 44	12.00000
C	33	R 45	12.00000
C	33	R 46	24.00000
C	33	R 47	18.00000
C	33	R 48	36.00000
C	33	R 49	24.00000
C	33	R 50	48.00000
C	34	OFSET	0.00100
C	34	R 7	1.00000
C	34	R 8	2.00000
C	34	R 9	3.00000
C	34	R 10	4.00000
C	34	R 12	8.00000
C	34	R 13	16.00000
C	34	R 14	24.00000
C	34	R 15	32.00000
C	34	R 17	27.00000
C	34	R 18	54.00000
C	34	R 19	81.00000
C	34	R 20	108.00000
C	34	R 22	64.00000

```
C  34 R   23    128.00000
C  34 R   24    192.00000
C  34 R   25    256.00000
C  34 R   43-     6.00000
C  34 R   44-    12.00000
C  34 R   45-    12.00000
C  34 R   46-    24.00000
C  34 R   47-    18.00000
C  34 R   48-    36.00000
C  34 R   49-    24.00000
C  34 R   50-    48.00000
C  35 OFSET       0.00100
C  35 R    7-     1.00000
C  35 R    8-     4.00000
C  35 R    9-     9.00000
C  35 R   10-    16.00000
C  35 R   12-     8.00000
C  35 R   13-    32.00000
C  35 R   14-    72.00000
C  35 R   15-   128.00000
C  35 R   17-    27.00000
C  35 R   18-   108.00000
C  35 R   19-   243.00000
C  35 R   20-   432.00000
C  35 R   22-    64.00000
C  35 R   23-   256.00000
C  35 R   24-   576.00000
C  35 R   25-  1024.00000
C  35 R   30      1.00000
C  35 R   31      1.00000
C  35 R   33      8.00000
C  35 R   34      8.00000
C  35 R   35-    27.00000
C  35 R   36     27.00000
C  35 R   37     27.00000
C  35 R   38     64.00000
C  35 R   39     64.00000
C  35 R   40     64.00000
C  35 R   43      6.00000
C  35 R   44     12.00000
C  35 R   45     24.00000
```

```
C   35 R   46     48.00000
C   35 R   47     54.00000
C   35 R   48    108.00000
C   35 R   49     96.00000
C   35 R   50    192.00000
C   36 OFSET       0.00100
C   36 R    7      1.00000
C   36 R    8      4.00000
C   36 R    9      9.00000
C   36 R   10     16.00000
C   36 R   12      8.00000
C   36 R   13     32.00000
C   36 R   14     72.00000
C   36 R   15    128.00000
C   36 R   17     27.00000
C   36 R   18    108.00000
C   36 R   19    243.00000
C   36 R   20    432.00000
C   36 R   22     64.00000
C   36 R   23    256.00000
C   36 R   24    576.00000
C   36 R   25   1024.00000
C   36 R   30-     1.00000
C   36 R   31-     1.00000
C   36 R   33-     8.00000
C   36 R   34-     8.00000
C   36 R   35     27.00000
C   36 R   36-    27.00000
C   36 R   37-    27.00000
C   36 R   38-    64.00000
C   36 R   39-    64.00000
C   36 R   40-    64.00000
C   36 R   43-     6.00000
C   36 R   44-    12.00000
C   36 R   45-    24.00000
C   36 R   46-    48.00000
C   36 R   47-    54.00000
C   36 R   48-   108.00000
C   36 R   49-    96.00000
C   36 R   50-   192.00000
C   37 OFSET       0.00100
```

1.0.0-2

```
C   37 R    7-      1.00000
C   37 R    8-      8.00000
C   37 R    9-     27.00000
C   37 R   10-     64.00000
C   37 R   12-      8.00000
C   37 R   13-     64.00000
C   37 R   14-    216.00000
C   37 R   15-    512.00000
C   37 R   17-     27.00000
C   37 R   18-    216.00000
C   37 R   19-    729.00000
C   37 R   20-   1728.00000
C   37 R   22-     64.00000
C   37 R   23-    512.00000
C   37 R   24-   1728.00000
C   37 R   25-   4096.00000
C   37 R   30       6.00000
C   37 R   31      12.00000
C   37 R   33      48.00000
C   37 R   34      96.00000
C   37 R   36     162.00000
C   37 R   37     324.00000
C   37 R   39     384.00000
C   37 R   40     768.00000
C   37 R   43       6.00000
C   37 R   44      12.00000
C   37 R   45      48.00000
C   37 R   46      96.00000
C   37 R   47     162.00000
C   37 R   48     324.00000
C   37 R   49     384.00000
C   37 R   50     768.00000
C   38 OFSET        0.00100
C   38 R    7       1.00000
C   38 R    8       8.00000
C   38 R    9      27.00000
C   38 R   10      64.00000
C   38 R   12       8.00000
C   38 R   13      64.00000
C   38 R   14     216.00000
C   38 R   15     512.00000
```

1.0.0-2

```
C  38 R   17     27.00000
C  38 R   18    216.00000
C  38 R   19    729.00000
C  38 R   20   1728.00000
C  38 R   22     64.00000
C  38 R   23    512.00000
C  38 R   24   1728.00000
C  38 R   25   4096.00000
C  38 R   30-     6.00000
C  38 R   31-    12.00000
C  38 R   33-    48.00000
C  38 R   34-    96.00000
C  38 R   36-   162.00000
C  38 R   37-   324.00000
C  38 R   39-   384.00000
C  38 R   40-   768.00000
C  38 R   43-     6.00000
C  38 R   44-    12.00000
C  38 R   45-    48.00000
C  38 R   46-    96.00000
C  38 R   47-   162.00000
C  38 R   48-   324.00000
C  38 R   49-   384.00000
C  38 R   50-   768.00000
C  39 OFSET       0.00100
C  39 R    9-     1.00000
C  39 R   10-     8.00000
C  39 R   14-     8.00000
C  39 R   15-    64.00000
C  39 R   19-    27.00000
C  39 R   20-   216.00000
C  39 R   24-    64.00000
C  39 R   25-   512.00000
C  39 R   31      6.00000
C  39 R   34     48.00000
C  39 R   37    162.00000
C  39 R   40    384.00000
C  39 R   47      6.00000
C  39 R   48     12.00000
C  39 R   49     48.00000
C  39 R   50     96.00000
```

1.0.0.2

```
C  40 OFSET     0.00100
C  40 R    9     1.00000
C  40 R   10     8.00000
C  40 R   14     8.00000
C  40 R   15    64.00000
C  40 R   19    27.00000
C  40 R   20   216.00000
C  40 R   24    64.00000
C  40 R   25   512.00000
C  40 R   31-    6.00000
C  40 R   34-   48.00000
C  40 R   37-  162.00000
C  40 R   40-  384.00000
C  40 R   47-    6.00000
C  40 R   48-   12.00000
C  40 R   49-   48.00000
C  40 R   50-   96.00000
C  41 OFSET     0.00100
C  41 R   16-    1.00000
C  41 R   17-    1.00000
C  41 R   18-    1.00000
C  41 R   19-    1.00000
C  41 R   20-    1.00000
C  41 R   21-    8.00000
C  41 R   22-    8.00000
C  41 R   23-    8.00000
C  41 R   24-    8.00000
C  41 R   25-    8.00000
C  41 R   42-    6.00000
C  41 R   44     6.00000
C  41 R   46     6.00000
C  41 R   48     6.00000
C  41 R   50     6.00000
C  42 OFSET     0.00100
C  42 R   16     1.00000
C  42 R   17     1.00000
C  42 R   18     1.00000
C  42 R   19     1.00000
C  42 R   20     1.00000
C  42 R   21     8.00000
C  42 R   22     8.00000
```

```
C  42 R   23      8.00000
C  42 R   24      8.00000
C  42 R   25      8.00000
C  42 R   42      6.00000
C  42 R   44-     6.00000
C  42 R   46-     6.00000
C  42 R   48-     6.00000
C  42 R   50-     6.00000
C  43 OFSET       0.00100
C  43 R   17-     1.00000
C  43 R   18-     2.00000
C  43 R   19-     3.00000
C  43 R   20-     4.00000
C  43 R   22-     8.00000
C  43 R   23-    16.00000
C  43 R   24-    24.00000
C  43 R   25-    32.00000
C  43 R   44      6.00000
C  43 R   46     12.00000
C  43 R   48     18.00000
C  43 R   50     24.00000
C  44 OFSET       0.00100
C  44 R   17      1.00000
C  44 R   18      2.00000
C  44 R   19      3.00000
C  44 R   20      4.00000
C  44 R   22      8.00000
C  44 R   23     16.00000
C  44 R   24     24.00000
C  44 R   25     32.00000
C  44 R   44-     6.00000
C  44 R   46-    12.00000
C  44 R   48-    18.00000
C  44 R   50-    24.00000
C  45 OFSET       0.00100
C  45 R   17-     1.00000
C  45 R   18-     4.00000
C  45 R   19-     9.00000
C  45 R   20-    16.00000
C  45 R   22-     8.00000
C  45 R   23-    32.00000
```

C	45	R	24-	72.00000
C	45	R	25-	128.00000
C	45	R	35-	1.00000
C	45	R	36	1.00000
C	45	R	37	1.00000
C	45	R	38	8.00000
C	45	R	39	8.00000
C	45	R	40	8.00000
C	45	R	44	6.00000
C	45	R	46	24.00000
C	45	R	48	54.00000
C	45	R	50	96.00000
C	46	OFSET		0.00100
C	46	R	17	1.00000
C	46	R	18	4.00000
C	46	R	19	9.00000
C	46	R	20	16.00000
C	46	R	22	8.00000
C	46	R	23	32.00000
C	46	R	24	72.00000
C	46	R	25	128.00000
C	46	R	35	1.00000
C	46	R	36-	1.00000
C	46	R	37-	1.00000
C	46	R	38-	8.00000
C	46	R	39-	8.00000
C	46	R	40-	8.00000
C	46	R	44-	6.00000
C	46	R	46-	24.00000
C	46	R	48-	54.00000
C	46	R	50-	96.00000
C	47	OFSET		0.00100
C	47	R	17-	1.00000
C	47	R	18-	8.00000
C	47	R	19-	27.00000
C	47	R	20-	64.00000
C	47	R	22-	8.00000
C	47	R	23-	64.00000
C	47	R	24-	216.00000
C	47	R	25-	512.00000
C	47	R	36	6.00000

```
C  47 R   37     12.00000
C  47 R   39     48.00000
C  47 R   40     96.00000
C  47 R   44      6.00000
C  47 R   46     48.00000
C  47 R   48    162.00000
C  47 R   50    384.00000
C  48 OFSET       0.00100
C  48 R   17      1.00000
C  48 R   18      8.00000
C  48 R   19     27.00000
C  48 R   20     64.00000
C  48 R   22      8.00000
C  48 R   23     64.00000
C  48 R   24    216.00000
C  48 R   25    512.00000
C  48 R   36-     6.00000
C  48 R   37-    12.00000
C  48 R   39-    48.00000
C  48 R   40-    96.00000
C  48 R   44-     6.00000
C  48 R   46-    48.00000
C  48 R   48-   162.00000
C  48 R   50-   384.00000
C  49 OFSET       0.00100
C  49 R   19-     1.00000
C  49 R   20-     8.00000
C  49 R   24-     8.00000
C  49 R   25-    64.00000
C  49 R   37      6.00000
C  49 R   40     48.00000
C  49 R   48      6.00000
C  49 R   50     48.00000
C  50 OFSET       0.00100
C  50 R   19      1.00000
C  50 R   20      8.00000
C  50 R   24      8.00000
C  50 R   25     64.00000
C  50 R   37-     6.00000
C  50 R   40-    48.00000
C  50 R   48-     6.00000
```

```
C   50 R   50-    48.00000
C   51 OFSET    1000.00000
C   51 R    3      1.0
C   52 OFSET    1000.00000
C   52 R    3-     1.0
C   53 OFSET    1000.00000
C   53 R    4      1.0
C   54 OFSET    1000.00000
C   54 R    4-     1.0
C   55 OFSET    1000.00000
C   55 R    5      1.0
C   56 OFSET    1000.00000
C   56 R    5-     1.0
C   57 OFSET    1000.00000
C   57 R    7      1.0
C   58 OFSET    1000.00000
C   58 R    7-     1.0
C   59 OFSET    1000.00000
C   59 R    8      1.0
C   60 OFSET    1000.00000
C   60 R    8-     1.0
C   61 OFSET    1000.00000
C   61 R    9      1.0
C   62 OFSET    1000.00000
C   62 R    9-     1.0
C   63 OFSET    1000.00000
C   63 R   10      1.0
C   64 OFSET    1000.00000
C   64 R   10-     1.0
C   65 OFSET    1000.00000
C   65 R   12      1.0
C   66 OFSET    1000.00000
C   66 R   12-     1.0
C   67 OFSET    1000.00000
C   67 R   13      1.0
C   68 OFSET    1000.00000
C   68 R   13-     1.0
C   69 OFSET    1000.00000
C   69 R   14      1.0
C   70 OFSET    1000.00000
C   70 R   14-     1.0
```

1.0.0-2

```
C   71 OFSET   1000.00000
C   71 R   15     1.0
C   72 OFSET   1000.00000
C   72 R   15-    1.0
C   73 OFSET   1000.00000
C   73 R   16     1.0
C   74 OFSET   1000.00000
C   74 R   16-    1.0
C   75 OFSET   1000.00000
C   75 R   17     1.0
C   76 OFSET   1000.00000
C   76 R   17-    1.0
C   77 OFSET   1000.00000
C   77 R   18     1.0
C   78 OFSET   1000.00000
C   78 R   18-    1.0
C   79 OFSET   1000.00000
C   79 R   19     1.0
C   80 OFSET   1000.00000
C   80 R   19-    1.0
C   81 OFSET   1000.00000
C   81 R   20     1.0
C   82 OFSET   1000.00000
C   82 R   20-    1.0
C   83 OFSET   1000.00000
C   83 R   21     1.0
C   84 OFSET   1000.00000
C   84 R   21-    1.0
C   85 OFSET   1000.00000
C   85 R   22     1.0
C   86 OFSET   1000.00000
C   86 R   22-    1.0
C   87 OFSET   1000.00000
C   87 R   23     1.0
C   88 OFSET   1000.00000
C   88 R   23-    1.0
C   89 OFSET   1000.00000
C   89 R   24     1.0
C   90 OFSET   1000.00000
C   90 R   24-    1.0
C   91 OFSET   1000.00000
```

```
C   91 R   25        1.0
C   92 OFSET   1000.00000
C   92 R   25-       1.0
C   93 OFSET    000.00000
C   93 R   27        1.0
C   94 OFSET    000.00000
C   94 R   28        1.0
C   95 OFSET    000.00000
C   95 R   30        1.0
C   96 OFSET    000.00000
C   96 R   31        1.0
C   97 OFSET    000.00000
C   97 R   33        1.0
C   98 OFSET    000.00000
C   98 R   34        1.0
C   99 OFSET    000.00000
C   99 R   35        1.0
C  100 OFSET    000.00000
C  100 R   36        1.0
C  101 OFSET    000.00000
C  101 R   37        1.0
C  102 OFSET    000.00000
C  102 R   38        1.0
C  103 OFSET    000.00000
C  103 R   39        1.0
C  104 OFSET    000.00000
C  104 R   40        1.0
C  105 OFSET    000.00000
C  105 R   42        1.0
C  106 OFSET    000.00000
C  106 R   43        1.0
C  107 OFSET    000.00000
C  107 R   44        1.0
C  108 OFSET    000.00000
C  108 R   45        1.0
C  109 OFSET    000.00000
C  109 R   46        1.0
C  110 OFSET    000.00000
C  110 R   47        1.0
C  111 OFSET    000.00000
C  111 R   48        1.0
```

1.0.0.2

```
          C 112 OFSET    000.00000
          C 112 R   49      1.0
          C 113 OFSET    000.00000
          C 113 R   50      1.0
FIRST B
                R    3    2.45833
                R    4    3.87499
                R    5    5.33333
                R    7    1.24702
                R    8    2.69791
                R    9    4.19791
                R   10    5.81250
                R   12    1.47916
                R   13    2.97916
                R   14    4.58333
                R   15    6.33333
                R   16     .34374
                R   17    1.79166
                R   18    3.44791
                R   19    5.25000
                R   20    7.16666
                R   21     .70833
                R   22    2.41666
                R   23    4.32291
                R   24    6.41666
                R   25    8.58333
                R   27    0.00000
                R   28    0.00000
                R   30    0.00000
                R   31    0.00000
                R   33    0.00000
                R   34    0.00000
                R   35    0.00000
                R   36    0.00000
                R   37    0.00000
                R   38    0.00000
                R   39    0.00000
                R   40    0.00000
                R   42  000.00000
                R   43  000.00000
                R   44  000.00000
                R   45  000.00000
                R   46  000.00000
                R   47  000.00000
                R   48  000.00000
                R   49  000.00000
                R   50  000.00000
EOF
```

Section IV

SMOG 4

The SMOG 4 program produces a matrix for fairing a two-dimensional curve, either double or single splined, with the primal formulation and sum of the deviations fit. The program is written in FORTRAN II for the IBM-1620 computer.

Fortran Symbol Definitions

N	-	The number of data offsets along the curve
COSTB	-	The cost associated with slacks
COSTN	-	The cost on the non-slack variables
X(I)	-	The x coordinate of the offset
Y(I)	-	The y coordinate of the offset if read in feet and decimals.
AA	-	The number of feet in the offset if read in feet, inches, and eighths
BB	-	The number of inches in the offset if read in feet, inches, and eighths
CC	-	The number of eighths in the offset if read in feet, inches, and eighths
SX	-	Scaling factor for normalizing the x coordinates. It equals $x_1 - x_0$ if set by the program.
TX	-	Translation factor for the x coordinates. It equals x_0 if set by the program.

Input Card Formats

The card formats are arranged as follows:

Contents						Card No.
Column	1-5					1
Contains any alphameric heading						
Format	I5	F10.4	F10.4			2
Variable	N	COSTB	COSTN			
	Alternate Cards:					
Format	F10.4	F5.1	F5.1	F5.1		Next N Cards
Variable	X(I)	$\pm$ AA	$\pm$ BB	$\pm$ CC		
	or					
Format	F10.4	F10.4				
Variable	X(I)	Y(I)				
Format	F10.4	F10.4				Last Card
Variable	SX	TX				

Output

The output is in a form suitable for the 1620 Users Group, Linear Programming Routine No. 10.0.006.

Sense Switches

Switch 1 ON - The curve will be double splined

Switch 1 OFF - The curve will be single splined

Switch 2 ON - The scaling factors will be read from the card reader

Switch 2 OFF - The program will set the scaling factors

Switch 3 ON - The second differences will be printed

Switch 3 OFF - The second differences will not be printed

SMOG 4

TYPEWRITER LISTING OF SECOND DIFFERENCES OF SAMPLE DATA

2ND DIFF - WLINES

STA	ACT VALUE	SIGN
2	.000169	-1.00
3	.000238	-1.00
4	.000068	-1.00
5	.000068	-1.00

** SAMPLE INPUT **

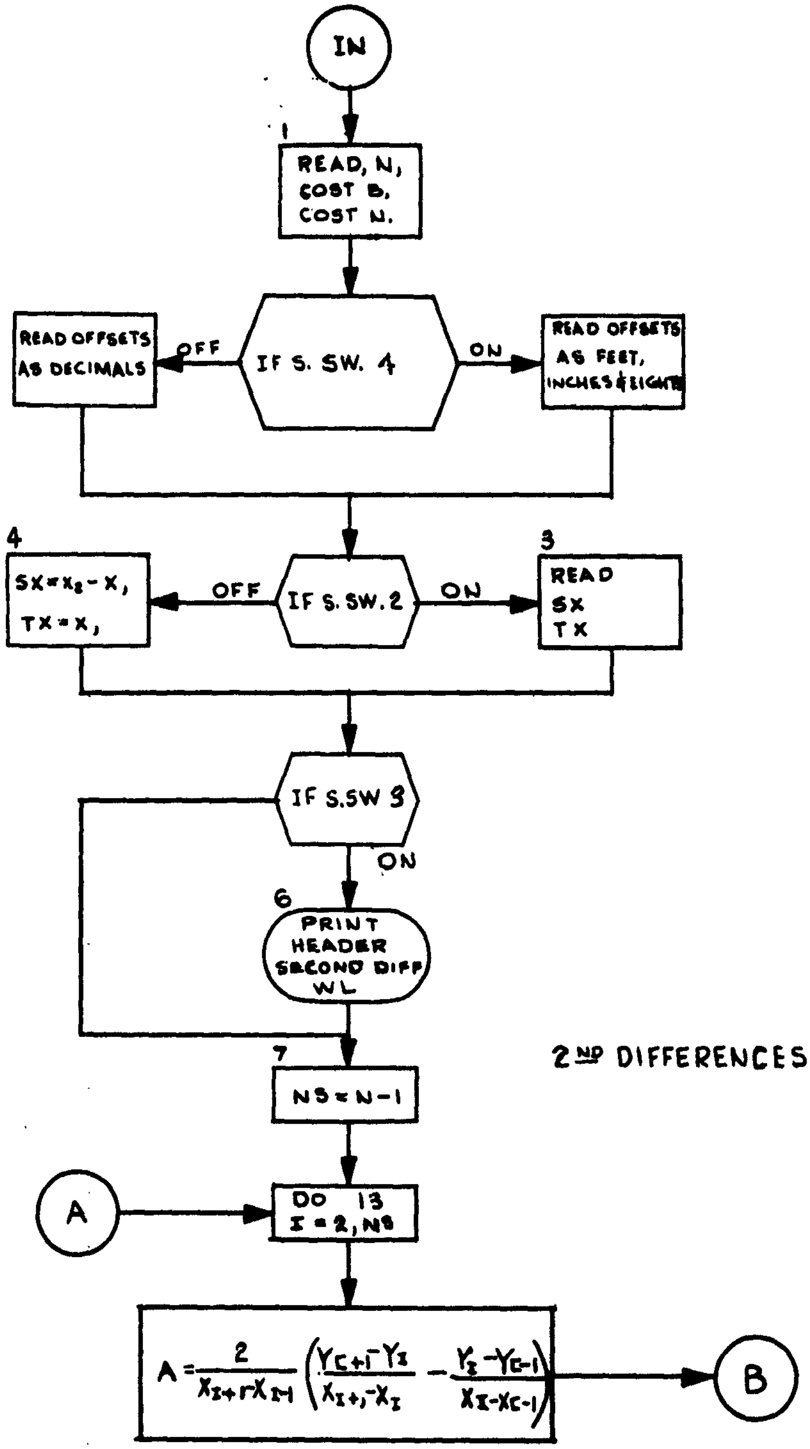
```
D1037
  5 999.9999 .001
 87.500000 30.5729
105.000000 30.307300
122.500000 30.09375
140.000000 29.953120
157.500000 29.833330
```

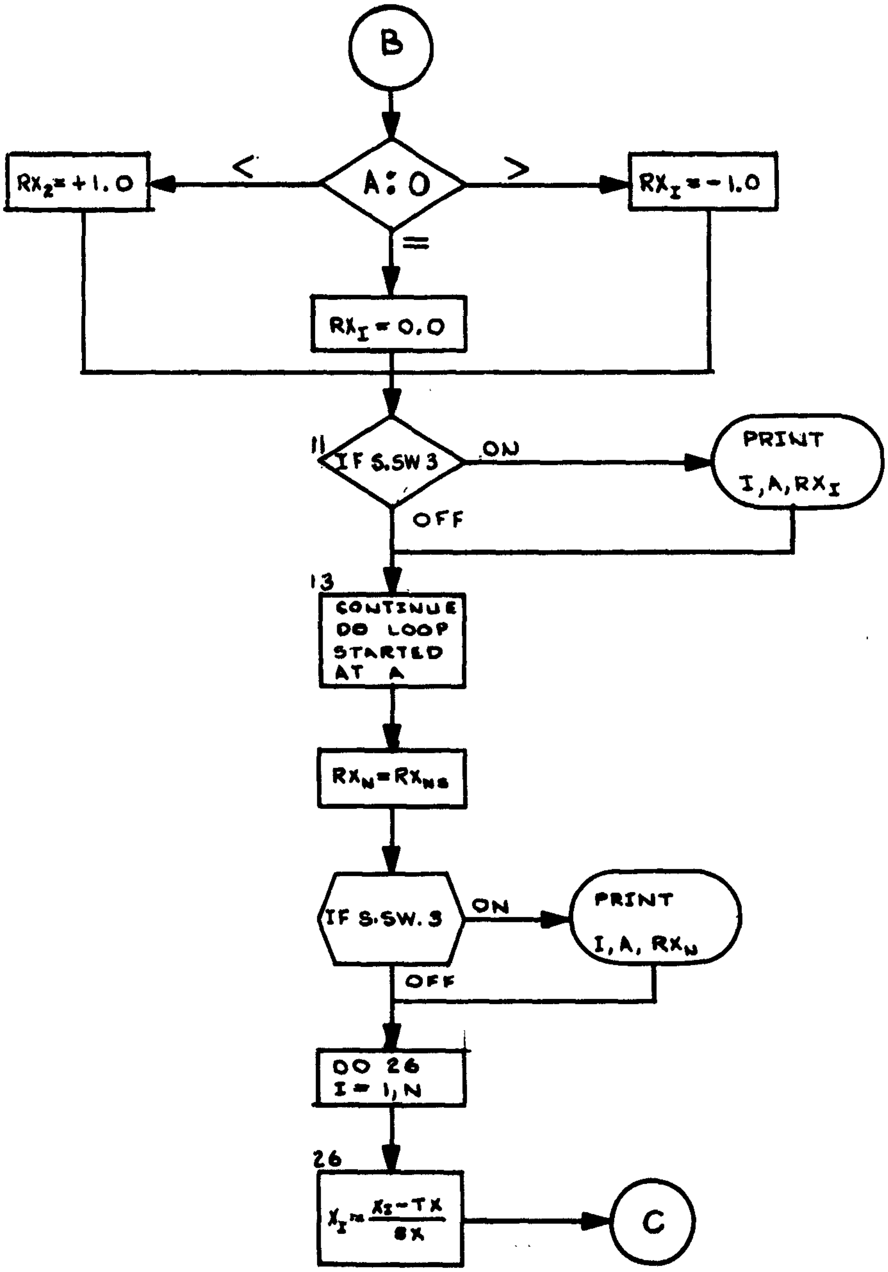
OUTPUT MATRIX FROM SAMPLE DATA

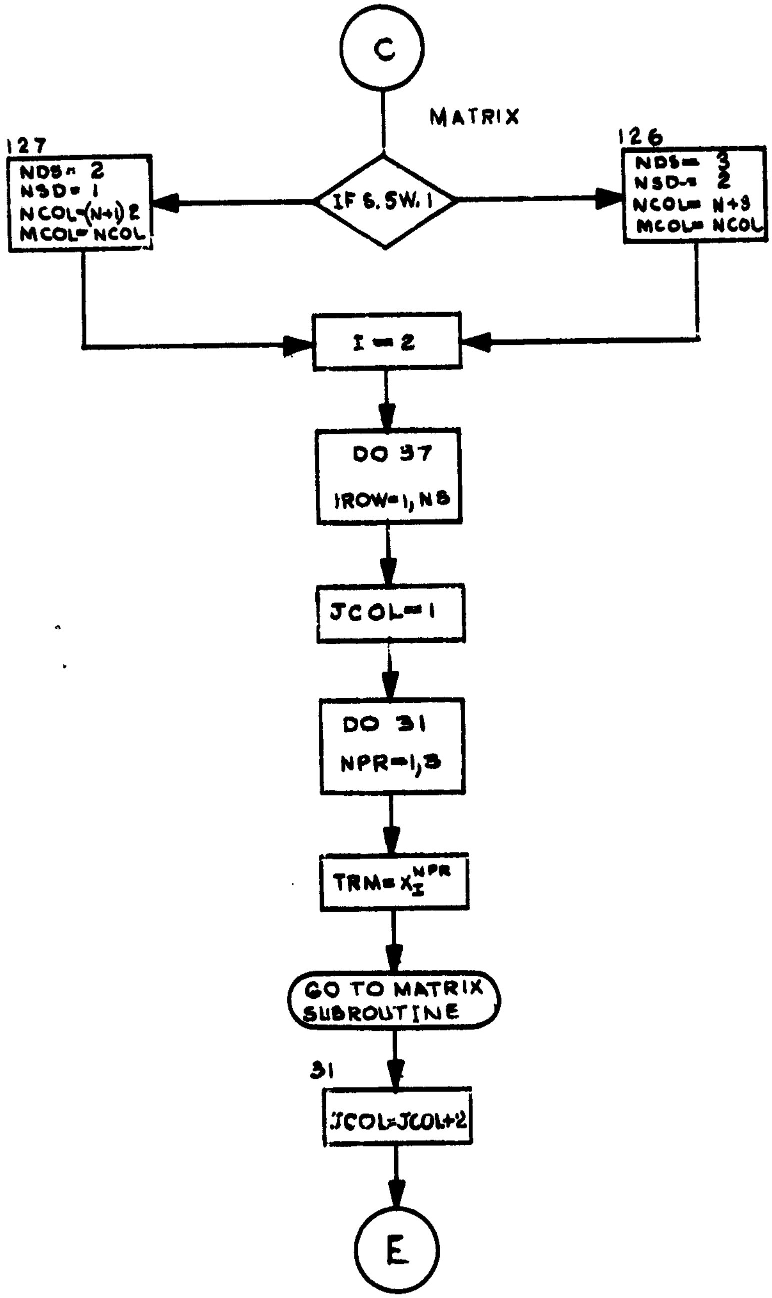
** SAMPLE OUTPUT **

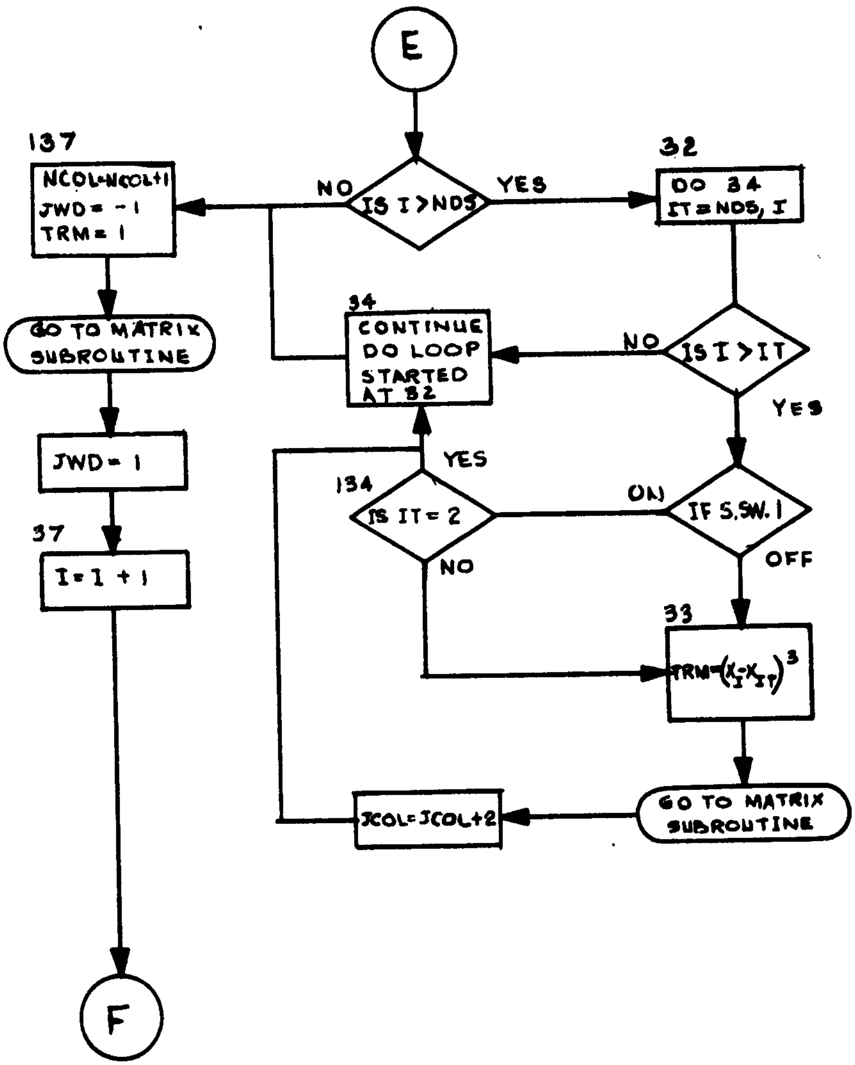
```
D1037
006 008 1
001 001      1.0000
001 002 -    1.0000
001 003      1.0000
001 004 -    1.0000
001 005      1.0000
001 006 -    1.0000
001 009      1.0000
002 001      2.0000
002 002 -    2.0000
002 003      4.0000
002 004 -    4.0000
002 005      8.0000
002 006 -    8.0000
002 010      1.0000
003 001      3.0000
003 002 -    3.0000
003 003      9.0000
003 004 -    9.0000
003 005     27.0000
003 006 -   27.0000
003 007      1.0000
003 008 -    1.0000
003 011      1.0000
004 001      4.0000
004 002 -    4.0000
004 003     16.0000
004 004 -   16.0000
004 005     64.0000
004 006 -   64.0000
004 007      8.0000
```

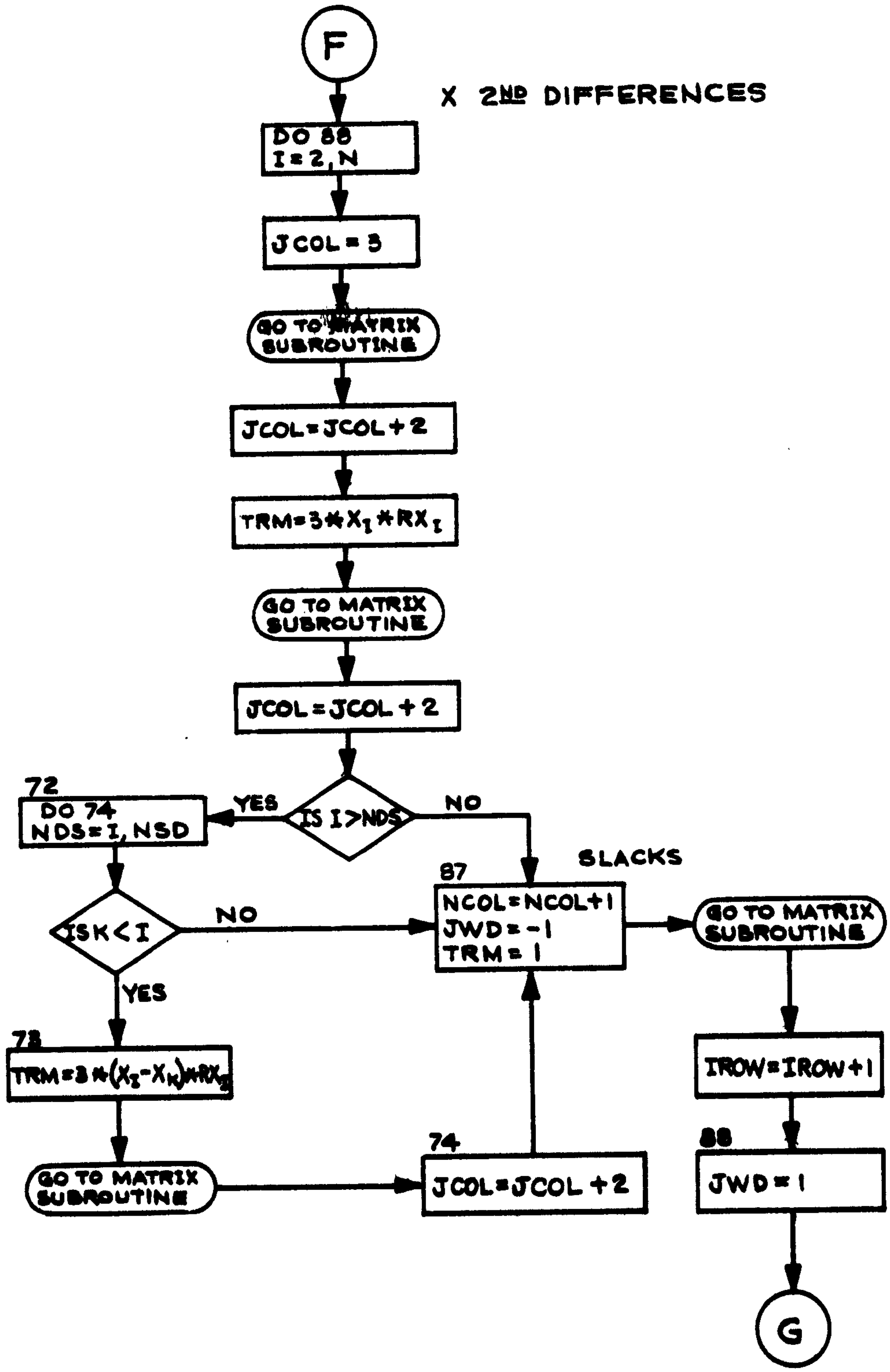
004	008	-	8.0000
004	012		1.0000
005	003	-	1.0000
005	004		1.0000
005	005	-	6.0000
005	006		6.0000
005	013		1.0000
006	003	-	1.0000
006	004		1.0000
006	005	-	12.0000
006	006		12.0000
006	007	-	6.0000
006	008		6.0000
006	014		1.0000
00J	001	101100000J	
00J	002	102100000J	
00J	003	103100000J	
00J	004	104100000J	
00J	005	105100000J	
00J	006	106100000J	
00J	007	107100000J	
00J	008	108100000J	
00J	009	109199999R	
00J	010	110199999R	
00J	011	111199999R	
00J	012	112199999R	
00J	013	113199999R	
00J	014	114199999R	
001	00J	101299999R	
002	00J	102299999R	
003	00J	103299999R	
004	00J	104299999R	
005	00J	105200000-	
006	00J	106200000-	
001	000	-	.2656
002	000	-	.4791
003	000	-	.6197
004	000	-	.7395
000			

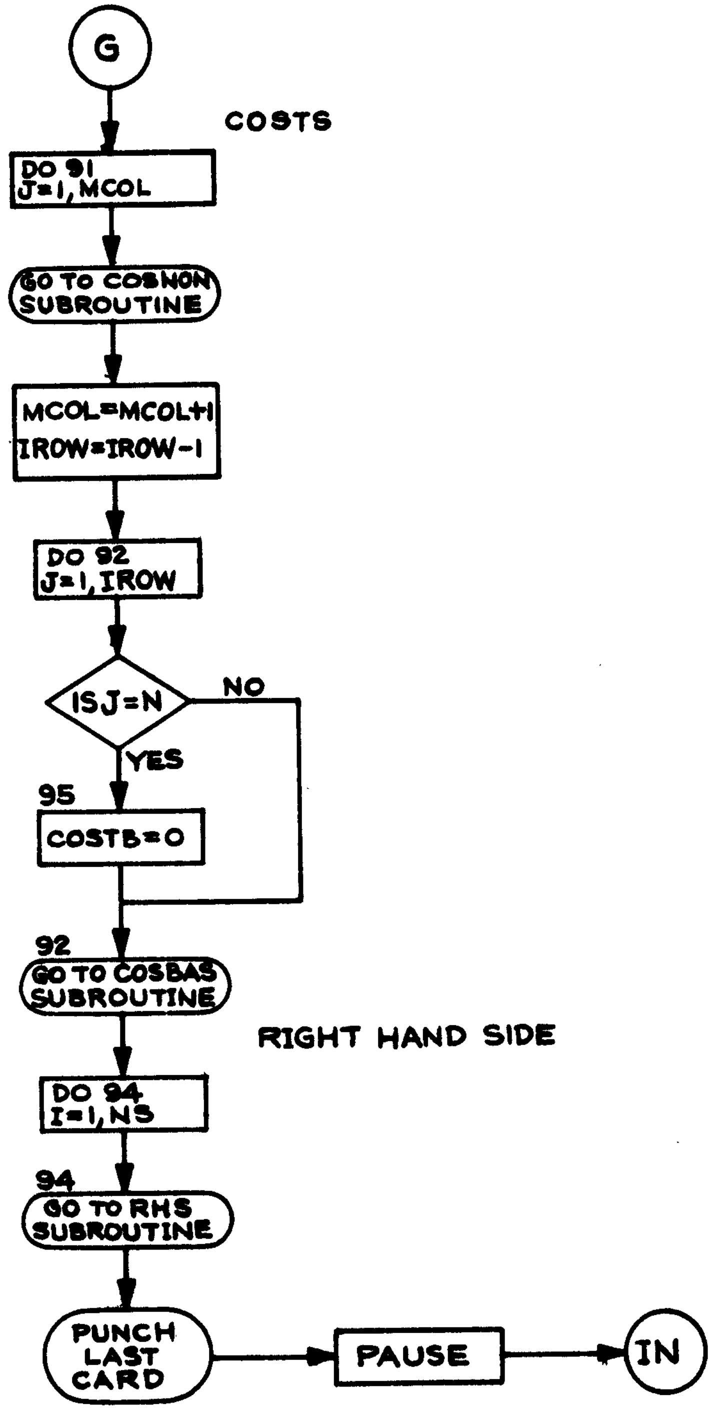

IN
1
READ, N,
COST B,
COST N.
IF S. SW. 4
OFF
READ OFFSETS AS DECIMALS
ON
READ OFFSETS AS FEET, INCHES & EIGHTS
IF S. SW. 2
OFF
4
SX = X2 - X1
TX = X1
ON
3
READ
SX
TX
IF S.SW 3
ON
6
PRINT HEADER SECOND DIFF WL
7
NS = N-1
2ND DIFFERENCES
A
DO 13
I = 2, NS
$A = \frac{2}{X_{I+1} - X_{I-1}} \left(\frac{Y_{I+1} - Y_I}{X_{I+1} - X_I} - \frac{Y_I - Y_{I-1}}{X_I - X_{I-1}} \right)$
B

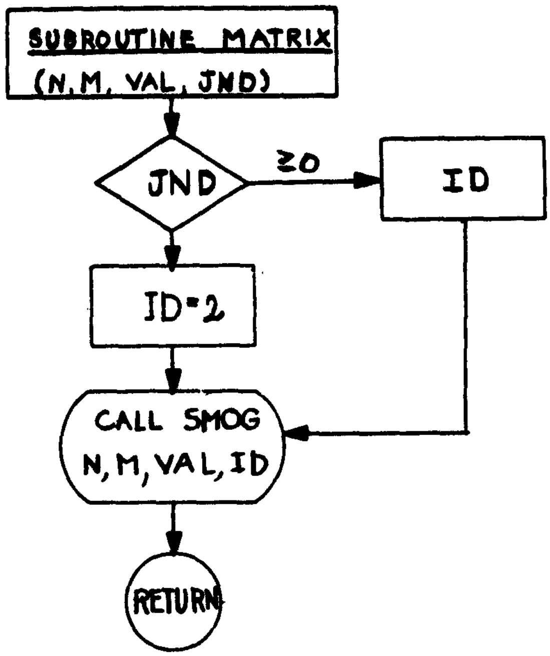

B
A : 0
<
>
=
RX2 = +1.0
RXI = -1.0
RXI = 0.0
11
IF S.SW 3
ON
OFF
PRINT
I, A, RXI
13
CONTINUE
DO LOOP
STARTED
AT A
RXN = RXNS
IF S.SW. 3
ON
OFF
PRINT
I, A, RXN
DO 26
I = 1, N
26
XI = XI - TX / SX
C

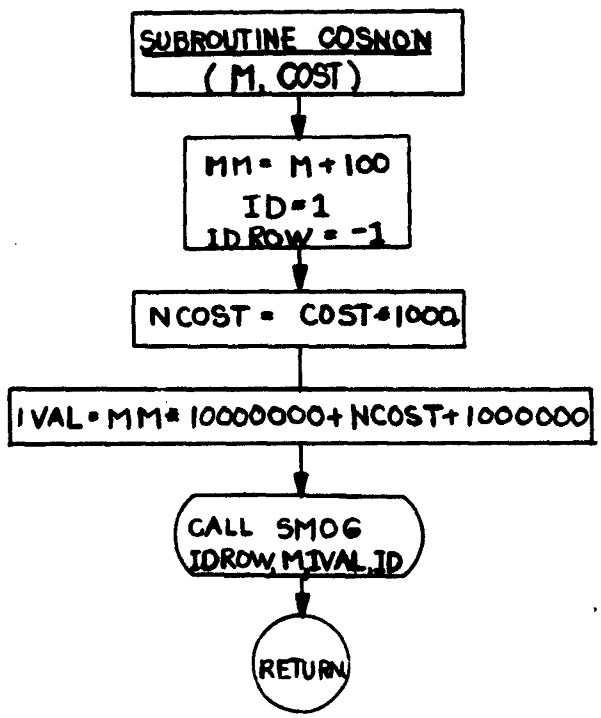

C
MATRIX
127
NDS= 2
NSD= 1
NCOL=(N+1) 2
MCOL= NCOL
IF S.SW. 1
126
NDS= 3
NSD= 2
NCOL= N+3
MCOL= NCOL
I = 2
DO 37
IROW=1, NS
JCOL=1
DO 31
NPR=1,3
TRM=X$_I^{NPR}$
GO TO MATRIX SUBROUTINE
31
JCOL=JCOL+2
E

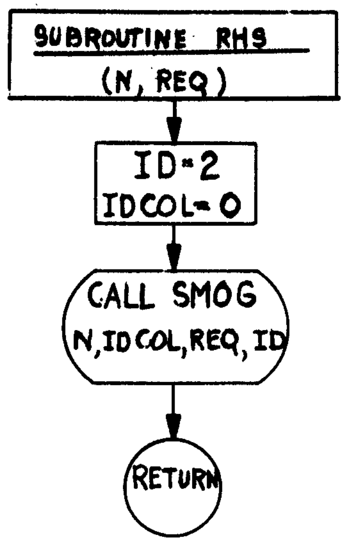

E
137
NCOL=NCOL+1
JWD = -1
TRM = 1
NO
IS I > NDS
YES
32
DO 34
IT = NDS, I
GO TO MATRIX SUBROUTINE
34
CONTINUE DO LOOP STARTED AT 32
NO
IS I > IT
YES
JWD = 1
YES
134
IS IT = 2
ON
IF S.SW. 1
37
I = I + 1
NO
OFF
33
TRM=(X_I-X_IT)^3
GO TO MATRIX SUBROUTINE
JCOL=JCOL+2
F

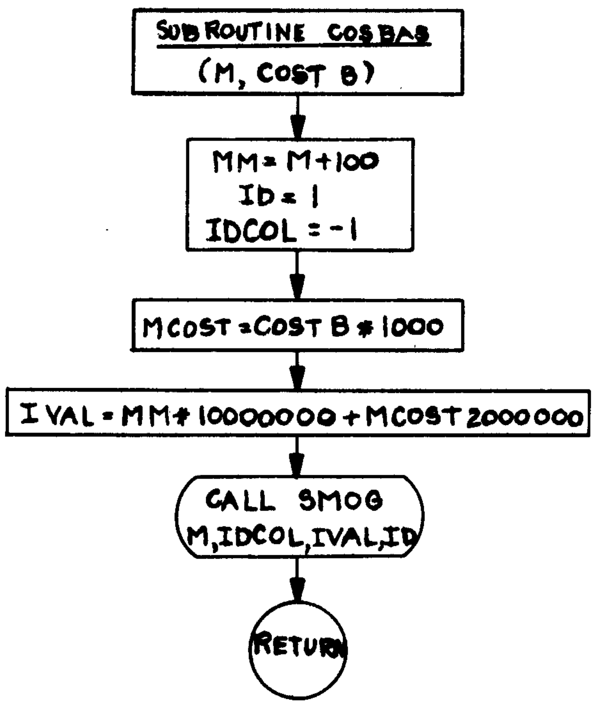

F
X 2ND DIFFERENCES
DO 88
I=2,N
JCOL=3
GO TO MATRIX SUBROUTINE
JCOL=JCOL+2
TRM=3*X_I*RX_I
GO TO MATRIX SUBROUTINE
JCOL=JCOL+2
IS I>NDS
YES
NO
72
DO 74
NDS=I,NSD
IS K<I
NO
YES
73
TRM=3*(X_I-X_K)*RX_I
GO TO MATRIX SUBROUTINE
74
JCOL=JCOL+2
SLACKS
87
NCOL=NCOL+1
JWD=-1
TRM=1
GO TO MATRIX SUBROUTINE
IROW=IROW+1
88
JWD=1
G

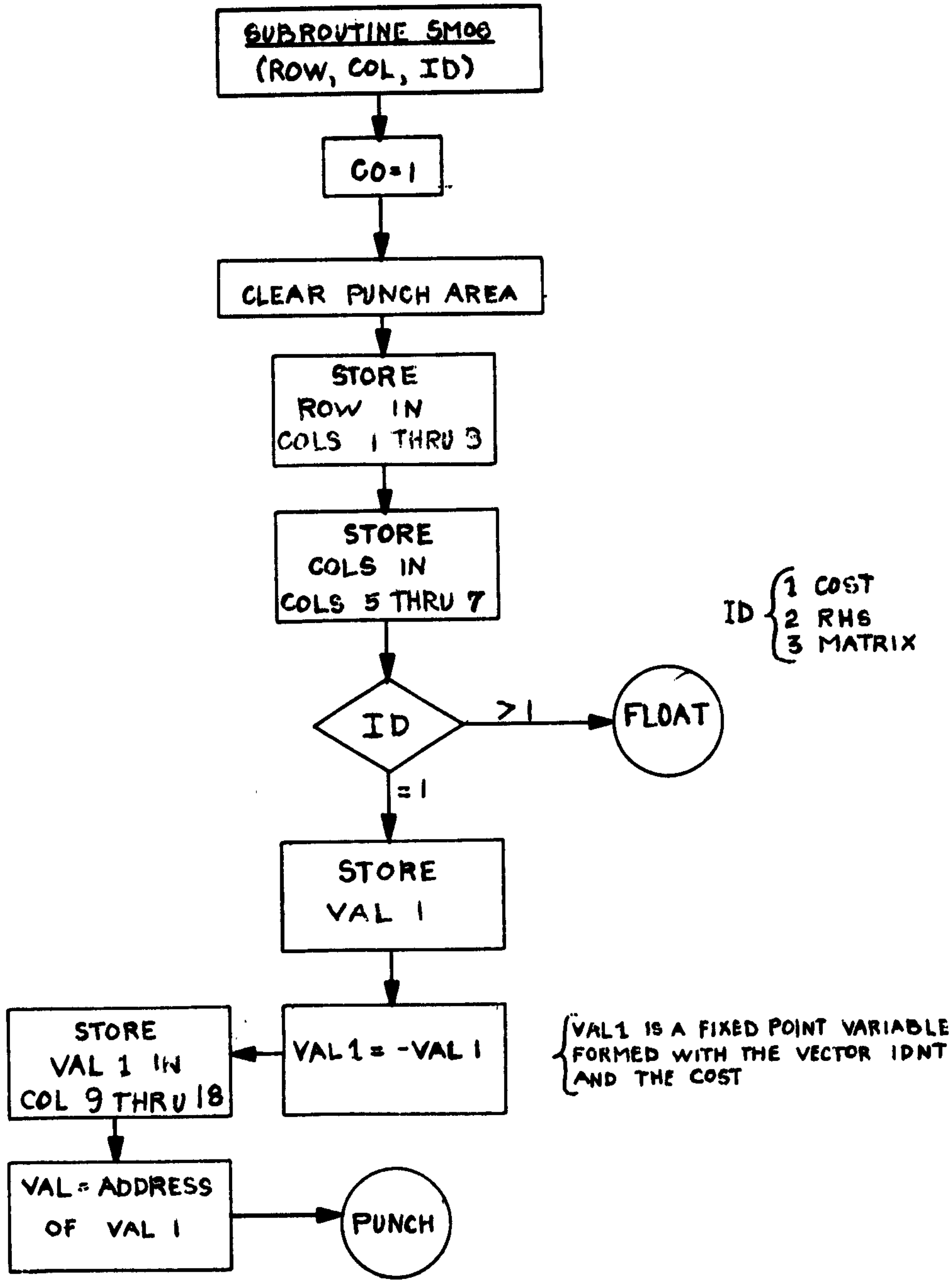

G
COSTS
DO 91
J=1, MCOL
GO TO COSNON SUBROUTINE
MCOL=MCOL+1
IROW=IROW-1
DO 92
J=1, IROW
ISJ=N
NO
YES
95
COSTB=0
92
GO TO COSBAS SUBROUTINE
RIGHT HAND SIDE
DO 94
I=1, NS
94
GO TO RHS SUBROUTINE
PUNCH LAST CARD
PAUSE
IN

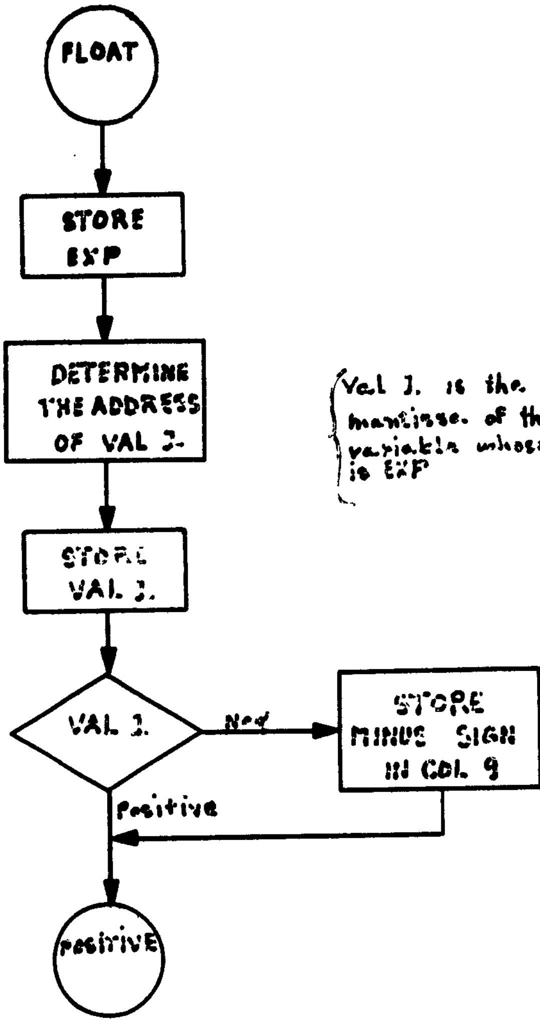

SUBROUTINE MATRIX
(N, M, VAL, JND)
JND
≥0
ID
ID=2
CALL SMOG
N, M, VAL, ID
RETURN


SUBROUTINE COSNON
(M, COST)
MM= M+100
ID=1
IDROW = -1
NCOST = COST*1000
IVAL=MM*10000000+NCOST+1000000
CALL SMOG
IDROW, M, IVAL, ID
RETURN

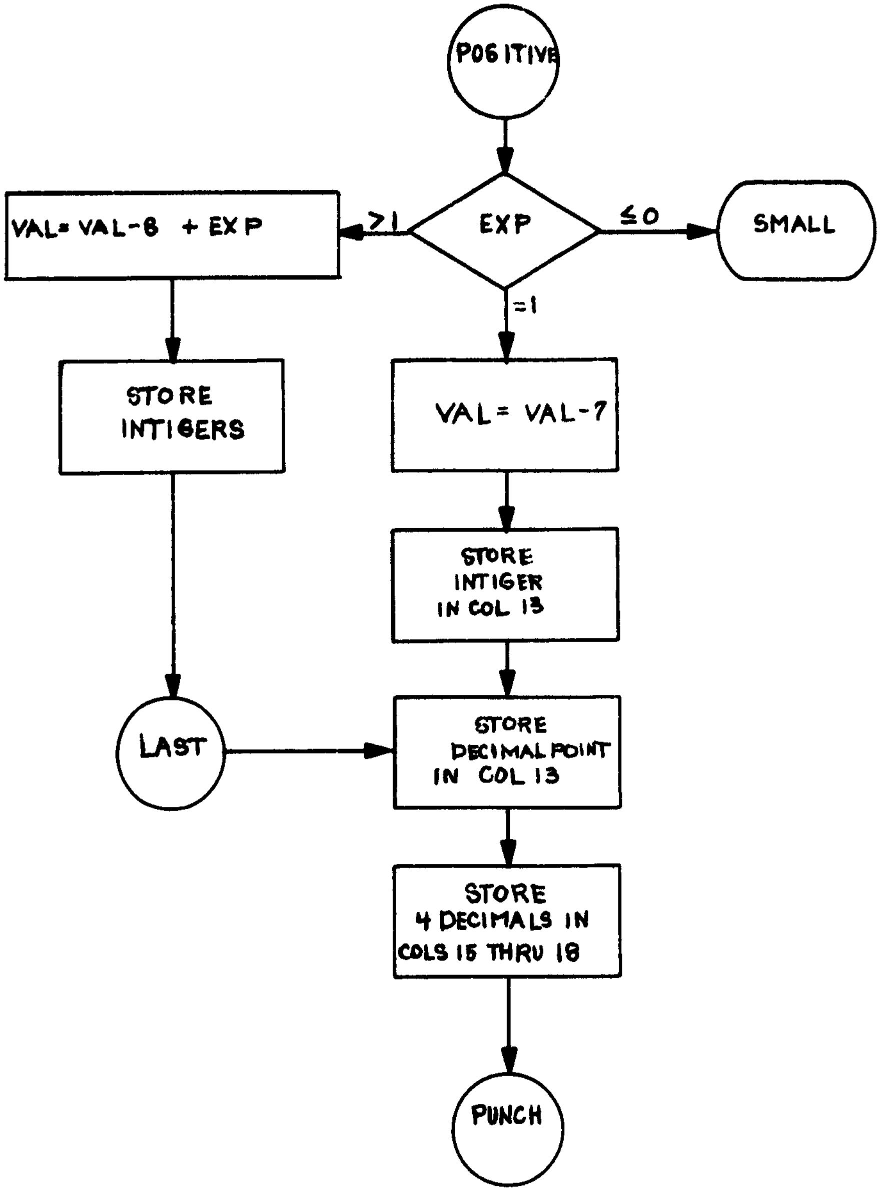

SUBROUTINE RHS
(N, REQ)
ID=2
IDCOL=0
CALL SMOG
N,IDCOL,REQ,ID
RETURN

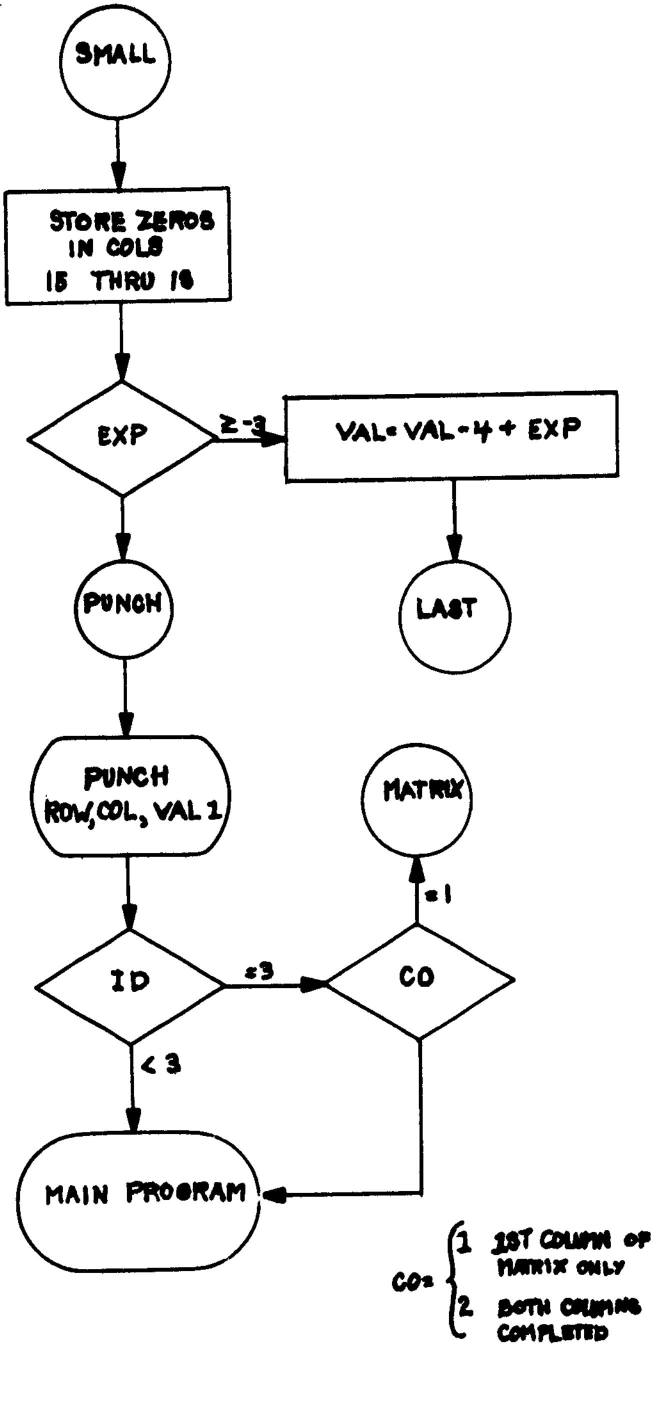

SUBROUTINE COSBAS
(M, COST B)
MM=M+100
ID=1
IDCOL=-1
MCOST=COST B*1000
IVAL=MM*10000000+MCOST 2000000
CALL SMOG
M,IDCOL,IVAL,ID
RETURN

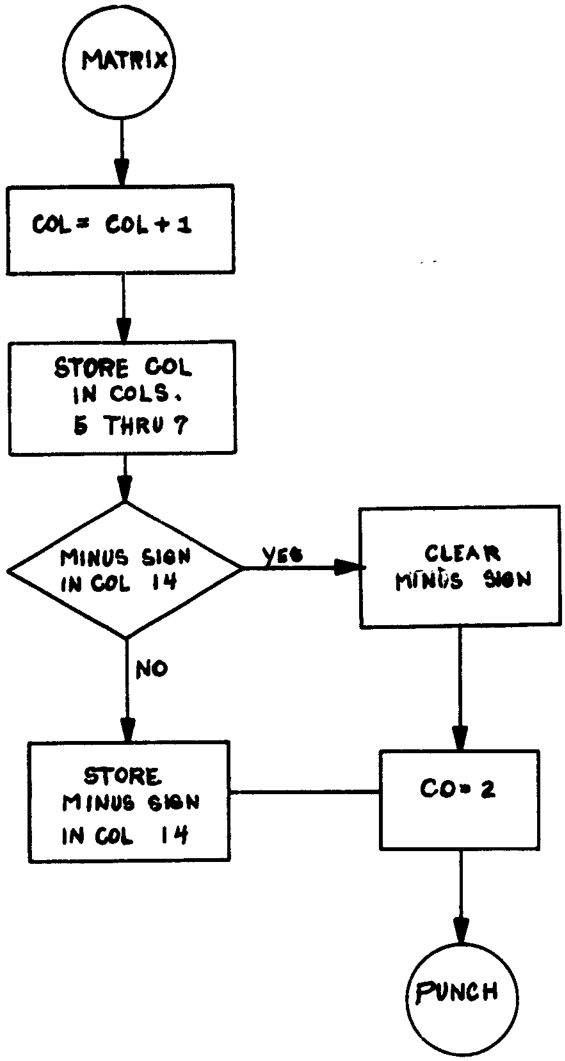

SUBROUTINE SM08
(ROW, COL, ID)
CO=1
CLEAR PUNCH AREA
STORE ROW IN COLS 1 THRU 3
STORE COLS IN COLS 5 THRU 7
ID
1 COST
2 RHS
3 MATRIX
ID
>1
FLOAT
=1
STORE VAL 1
VAL 1 = -VAL 1
VAL1 IS A FIXED POINT VARIABLE FORMED WITH THE VECTOR IDNT AND THE COST
STORE VAL 1 IN COL 9 THRU 18
VAL = ADDRESS OF VAL 1
PUNCH

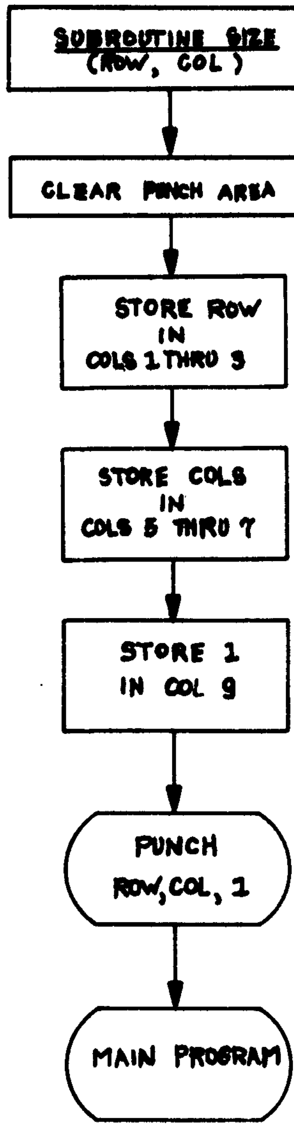


Val 1. is the mantissa of the exponent variable whose exponent is EXP


POSITIVE
EXP
>1
≤0
=1
SMALL
VAL= VAL-8 + EXP
STORE INTIGERS
VAL= VAL-7
STORE INTIGER IN COL 13
LAST
STORE DECIMAL POINT IN COL 13
STORE 4 DECIMALS IN COLS 15 THRU 18
PUNCH


SMALL
STORE ZEROS IN COLS 15 THRU 16
EXP
≥-3
VAL=VAL-4+EXP
PUNCH
LAST
PUNCH ROW,COL, VAL 1
MATRIX
=1
ID
=3
CO
<3
MAIN PROGRAM
CO=
1 1ST COLUMN OF MATRIX ONLY
2 BOTH COLUMNS COMPLETED


MATRIX
COL = COL + 1
STORE COL
IN COLS.
5 THRU 7
MINUS SIGN
IN COL 14
YES
CLEAR
MINUS SIGN
NO
STORE
MINUS SIGN
IN COL 14
CO = 2
PUNCH


SUBROUTINE SIZE (ROW, COL)
CLEAR PUNCH AREA
STORE ROW IN COLS 1 THRU 3
STORE COLS IN COLS 5 THRU 7
STORE 1 IN COL 9
PUNCH ROW, COL, 1
MAIN PROGRAM

```
*0810
C       SMOG  FOR  1620  LP   4/2/63
        DIMENSION X(30),Y(30),RX(30)
      1 READ 600
        PUNCH 600
        READ 150,N,COSTB,COSTN
        JWD=1
        NL=N+1
        DO 2 I=1,N
        IF(SENSE SWITCH 4)201,202
    201 READ 160,X(I),AA,BB,CC
        Y(I)=AA+(BB+(CC/8.))/12.
        GO TO 2
    202 READ 152,X(I),Y(I)
      2 CONTINUE
        IF(SENSE SWITCH 2)3,4
      3 READ 152,SX,TX
        GO TO 5
      4 SX=X(2)-X(1)
        TX=X(1)
      5 IF(SENSE SWITCH 3)6,7
      6 PRINT 157
C        *   *   SECOND  DIFFERENCES    *    *
      7 NS=N-1
        DO 13 I=2,NS
        A=2./((X(I+1)-X(I-1)))
        ZA=Y(I+1)-Y(I)
        A=A*(ZA/(X(I+1)-X(I))-(Y(I)-Y(I-1))/(X(I)-X(I-1)))
        IF(A)10,9,8
      8 RX(I)=-1.
        GO TO 11
      9 RX(I)=0.
        GO TO 11
     10 RX(I)=1.
     11 IF(SENSE SWITCH 3)12,13
     12 PRINT 158,I,A,RX(I)
     13 CONTINUE
        RX(N)=RX(NS)
        IF(SENSE SWITCH 3)14,15
     14 PRINT 158,I,A,RX(N)
C        *    *        NORMALIZING     *    *
     15 DO 26 I=1,N
     26 X(I)=(X(I)-TX)/SX
C        *    *          MATRIX          *    *
        IF(SENSE SWITCH 1)126,127
    126 NDS=3
        NSD=2
        NCOL=3+N
```

```
      NROW=N-1+(N-1)/2
      GO TO 128
  127 NDS=2
      NSD=1
      NCOL=(N+1)*2
      NROW=2*(N-1)
  128 MCOL=NCOL
      I=2
      CALL SIZE (NROW,MCOL)
      DO 37 IROW=1,NS
      JCOL=1
      DO 31 NPR=1,3
      TRM=X(I)**NPR
      CALL MATRIX(IROW,JCOL,TRM,JWD)
   31 JCOL=JCOL+2
      IF(I-NDS)137,137,32
   32 DO 34 IT=NDS,I
      IF(I-IT)34,34,133
  133 IF(SENSE SWITCH 1)134,33
  134 IF(IT-2*(IT/2))33,34,33
   33 TRM=(X(I)-X(IT))**3
      CALL MATRIX(IROW,JCOL,TRM,JWD)
      JCOL=JCOL+2
   34 CONTINUE
  137 NCOL=NCOL+1
      JWD=-1
      TRM=1.
      CALL MATRIX(IROW,NCOL,TRM,JWD)
      JWD=1
   37 I=I+1
C        *    *      X  SECOND DIFFERENCES      *    *
      DO 88 I=NDS,N,NSD
      JCOL=3
      CALL MATRIX(IROW,JCOL,RX(I),JWD)
      JCOL=JCOL+2
      TRM=3.*X(I)*RX(I)
      CALL MATRIX(IROW,JCOL,TRM,JWD)
      JCOL=JCOL+2
      IF(I-NDS)75,75,72
   72 DO 74 K=NDS,I,NSD
      IF(K-I)73,75,75
   73 TRM=3.*(X(I)-X(K))*RX(I)
      CALL MATRIX(IROW,JCOL,TRM,JWD)
   74 JCOL=JCOL+2
   75 CONTINUE
C        *    *      SLACKS     *    *
   87 NCOL=NCOL+1
      JWD=-1
```

```
      TRM=1.
      CALL MATRIX(IROW,NCOL,TRM,JWD)
      IROW=IROW+1
   88 JWD=1
C        *        *            COSTS              *        *
      DO 91 J=1,MCOL
   91 CALL COSNON(J,COSTN)
      IROW=IROW-1
      MCOL=MCOL+IROW
      DO 300 I=J,MCOL
  300 CALL COSNON(I,COSTB)
      DO 92 J=1,IROW
      IF(J-N)92,95,92
   95 COSTB=0.
   92 CALL COSBAS(J,COSTB)
C            *      *         R  H  S          *      *
      DO 94 I=1,NS
      Y(I+1)=Y(I+1)-Y(1)
   94 CALL RHS(I,Y(I+1))
      PUNCH 161
      PAUSE
      GO TO 1
  150 FORMAT(I5,2F10.6)
  152 FORMAT(2F10.4)
  157 FORMAT(10X,17H2ND DIFF - WLINES//
     C     8X,22HSTA  ACT VALUE    SIGN/)
  158 FORMAT(5X,I5,F12.6,F8.2)
  160 FORMAT(F10.5,3F5.1)
  161 FORMAT(3H000)
  600 FORMAT(5H     )
  601 FORMAT(19X,I4,I4,2H 1)
      END
*0810
      SUBROUTINE COSBAS(M,COSTB)
      MM=M+100
      ID=1
      IDCOL=-1
      MCOST=COSTB*1000.
      IVAL=MM*10000000+MCOST+2000000
      CALL SMOG (M,IDCOL,IVAL,ID)
      RETURN
      END
*0810
      SUBROUTINE COSNON(M,COST)
      MM=M+100
      ID=1
      IDROW=-1
      NCOST=COST *1000.
```

```
      IVAL=MM*10000000+NCOST+1000000
      CALL SMOG (IDROW,M,IVAL,ID)
      RETURN
      END
*0810
      SUBROUTINE RHS(N,REQ)
      ID=2
      IDCOL=0
      CALL SMOG (N,IDCOL,REQ,ID)
      RETURN
      END
*0810
      SUBROUTINE MATRIX(N,M,VAL,JND)
      IF(JND)509,508,508
  508 ID=3
      CALL SMOG (N,M,VAL,ID)
      RETURN
  509 ID=2
      CALL SMOG (N,M,VAL,ID)
      RETURN
      END
```

```
I-ADD OP                    P/L LABEL   OP      OPERANDS

360007200500360020100500440001200275260005900274250001100000260009000269  -
26000950026431000000200260011400274250000000001149000120000O              -
                            *                 * *  SMOG SUBROUTINE  -  4/2/63  *  **
12006                                   DORG  12006*
12006 L2 12005 00000        SMOG        SF    SMOG-1,,0*
12018 J2 12005 000-5                    SM    SMOG-1,5,C10*
12030 KO 12669 1200N                    TF    ROW,SMOG-1,0111,    STORE ROW VALUE*
12042 J2 12005 000-5                    SM    SMOG-1,5,010*
12054 KO 12659 1200N                    TF    COL,SMOG-1,0111,    STORE COLUMN VALUE*
12066 J2 12005 000J0                    SM    SMOG-1,10,010*
12078 KO 12644 1200N                    TF    ID,SMOG-1,0111,     STORE ID VALUE*
12090 J0 12649 J2677                    TFM   VAL,VAL1,017,       SAVE ADRESS OF VAL1*
12102 J6 12686 000-1                    TFM   CO,1,010,           ASSIGN CO EQUAL 1*
12114 L3 12837 00000                    CF    ZERO-9,,0*
12126 L3 12787 00000                    CF    ZERO-59,,0*
12138 L3 12737 00000                    CF    ZERO-109,,0*
12150 KO 12634 12846                    TF    STORE-ZERO,01,       CLEAR PUNCH AREA*
12162 L2 12667 00000                    SF    ROW-2,,0*
12174 PL 12479 12669                    TNF   STORE-155,ROW,01,   STORE ROW NUMBER IN COLS. 1 THRU 3*
12186 L2 12657 00000                    SF    COL-2,,0*
12198 PL 12487 12659                    TNF   STORE-147,COL,01,   STORE COLUMN NUMBER IN COLS. 5 THRU 7*
12210 J1 12005 000-5                    AM    SMOG-1,5,010*
                            *                 ID#1 FOR COST IDENTIFICATION %FIXED POINT VAR.¤*
                            *                 ID#2 FOR RHS. %FLOATING POINT VAR.¤*
                            *                 ID#3 FOR MATRIX VALUES %FLOAT.POINT VAR.¤*
12222 J4 12644 000-1                    CM    ID,1,010,            CHECK FOR FIXED VALUE*
12234 M6 12856 01100                    BP    FLOAT,,0*
12246 KO 12677 1200N        FIXED       TF    VAL1,SMOG-1,0111,   STORE COST AND VECTOR IDENTIFICATION*
12258 L2 12677 00000                    SF    VAL1,,0,            SET FLAG ON THE LAST DIGIT*
12270 PL 12509 12677                    TNF   STORE-125,VAL1,01.  STORE COST IN COLS. 9 THRU 18*
12282 L9 12475 00400        PUNCH       WACD  STORE-159,,0*
12294 J4 12644 000-3                    CM    ID,3,010,           CHECK FOR MATRIX VALUES*
12306 M7 12342 01200                    BNE   EXIT,,0*
12318 J4 12686 000-2                    CM    CO,2,010,           CHECK FOR LAST PASS OF MATRIX LOOP*
12330 M7 12362 01200                    BNE   MATRIX,,0*
12342 J2 12005 000J2        EXIT        SM    SMOG-1,12,010*
12354 M9 1200N 00000                    B     SMOG-1,,06*
12362                                   DORG  *-3*
```


10.0 2

```
I-ADD OP                 P/L LABEL   OP      OPERANDS

12362 J1 12659 000-!         MATRIX AM    COL,1,010,            NUMBER OF NEXT COLUMN‡
12374 PL 12487 12659                TNF   STORE-147,COL,01,     STORE COLUMN NUMBER IN COLS. 5 THRU 7‡
12386 L2 12490 00000                SF    STORE-144,,0‡
12398 J4 12491 000K0                CM    STORE-143,20,010,     CHECK FOR MINUS SIGN IN COL. 9‡
12410 M6 12442 01200                BE    NEG,,0‡
12422 J6 12491 000K0                TFM   STORE-143,20,010,     STORE MINUS SIGN IN COL. 9‡
12434 M9 12454 00000                B     TWO,,0‡
12442                               DORG  *-3‡
12442 J6 12491 000-0         NEG    TFM   STORE-143,,010,       CLEAR MINUS SIGN‡
12454 J6 12686 000-2         TWO    TFM   CO,2,010,             ASSIGN CO EQUAL 2‡
12466 M9 12282 00000                B     PUNCH,,0‡
12474                               DORG  *-3‡
12634       00161            STORE  DS    161‡
12644       00010            ID     DS    10‡
12649       00005            VAL    DS    5‡
12659       00010            COL    DS    10‡
12669       00010            ROW    DS    10‡
12677       00008            VAL1   DS    8‡
12679       00002            EXP    DS    2‡
12684       00005            LOCAT  DS    5‡
12686       00002            CO     DS    2‡
12736       00050                   DC    50,000000000000000000000000000000000000000000000000‡
00000-000000000000000000000000000000000000000000000‡          -6J2687J2737-
12786       00050                   DC    50,000000000000000000000000000000000000000000000000‡
00000-000000000000000000000000000000000000000000000‡          -6J2737J2787-
12836       00050                   DC    50,000000000000000000000000000000000000000000000000‡
00000-000000000000000000000000000000000000000000000‡          -6J2787J2837-
12846       00010            ZERO   DC    10,0000000000‡
000C0-000000000‡                                                -6J2837J2847-
12854       00008            ZERO4  DC    8,70707070‡
00000P0707070‡                                                  -6J2847J2855-
12856 KO 12679 1200N         FLOAT  TF    EXP,SMOG-1,0111,      STORE EXPONENT‡
12868 KO 12684 12005                TF    LOCAT,SMOG-1,01‡
                             *            LOOP TO STORE IN LOCAT THE ADRESS OF THE‡
                             *            FLOATING POINT VALUE‡
12880 MM 12924 12684         LOOP   BNF   ADRESS,LOCAT,01‡
12892 L3 12684 00000                CF    LOCAT,,0‡
12904 KO 12684 1268M                TF    LOCAT,LOCAT,0111‡
```

1.0.0-2

```
 I-ADD OP                   P/L LABEL   OP      OPERANDS

12916 M9 12880 00000                    B     LOOP,,0*
12924                                   DORG  *-3*
12924 J2 12684 000-2          ADRESS    SM    LOCAT,2,010,         ADRESS OF THE MANTISA*
12936 KO 12677 1268M                    TF    VAL1,LOCAT,0111,     STORE MANTISA IN VAL1*
12948 MM 12984 1264R                    BNF   POSIT,VAL,0111,      CHECK FOR NEGATIVE VALUE*
12960 L3 1264R 00000                    CF    VAL,,06*
12972 J6 12491 000K0                    TFM   STORE-143,20,010,    STORE MINUS SIGN IN COL. 9*
12984 J4 12679 000-1          POSIT     CM    EXP,1,010*
12996 M7 13168 01300                    BN    SMALL,,0,            CHECK FOR EXP LESS OR EQUAL THAN 0*
13008 M6 130   01100                    BP    PLUS1,,0,            CHECK FOR EXP GREATER THAN 1*
                              *               AREA FOR EXP#1*
13020 J2 12649 000-7                    SM    VAL,7,010*
13032 KN 12499 1264R                    TD    STORE-135,VAL,0111,STORE INTEGER IN COL. 13*
13044 J5 12498 00007                    TDM   STORE-136,7,0*
13056 M9 13100 00000                    B     DECIM,,0*
13064                                   DORG  *-3*
                              *               ADRESS OF THE UNITS POSITION  # VAL - 8 & EXP*
13064 J2 12649 000-8          PLUS1     SM    VAL,8,010*
13076 KJ 12649 12679                    A     VAL,EXP,01*
13088 PL 12499 1264R                    TNF   STORE-135,VAL,0111,STORE INTEGERS*
13100 J1 12649 000-1          DECIM     AM    VAL,1,010*
13112 L2 1264R 00000                    SF    VAL,,06*
13124 J1 12649 000-3                    AM    VAL,3,010*
13136 PL 12509 1264R          LAST      TNF   STORE-125,VAL,0111,STORE 4 DECIMALS IN COLS. 15 THRU 18*
13148 J6 12501 000-3                    TFM   STORE-133,3,010,     STORE DECIMAL POINT IN COL. 14*
13160 M9 12282 00000                    B     PUNCH,,0*
13168                                   DORG  *-3*
13168 KO 12509 12854          SMALL     TF    STORE-125,ZERO4,01*
13180 J4 12679 000-L                    CM    EXP,3,01011,         CHECK FOR EXP LESS THAN -3*
13192 M7 12282 01300                    BN    PUNCH,,0*
                              *               ADRESS OF THE 1/10000 POSITION # VAL - 4 & EXP*
13204 J2 12649 000-4                    SM    VAL,4,010*
13216 KJ 12649 12679                    A     VAL,EXP,01*
13228 M9 13136 00000                    B     LAST,,0*
13236                                   DORG  *-3*
12006                                   DEND  12006*
00000  L60000000500490000*                                           -8-0096-0115-
3600100005003600172005003600244005003600316005003600000000500         -
```

```
I-ADD OP                  P/L LABEL    OP        OPERANDS

00000000000010203040002040608000306090210040802161005001510200602181420 0‡ -
7041128200806142230090817263000000000005060708090012141618151811242720242‡ -
8223635203530454036324844553249465360484654627544536271801234567891234 56‡ -
789-23456789-J3456789-JK456789-JKL56789-JKLM6789-JKLMN789-JKLMN089-JKLMN‡ -
M80000000000049J20060P9-JKLMNOPQ‡     L10038800019M9000000000M9000360000 0  -
```

1.0.0 - 2

```
 1-ADD OP                   P/L LABEL   OP      OPERANDS

3600072005003600201005004400012002752600059002742500011000002600090000269  -
26000950026431000000020026001140027425000000001149000120C0000              -
                            *           * * SIZE SUBROUTINE  -  4/2/63  *  **
12006                              DORG  12006*
12006 L2 12005 00000      SIZE     SF    SIZE-1,,0*
12018 J2 12005 000-5               SM    SIZE-1,5,010*
12030 KO 12243 1200N               TF    ROW,SIZE-1,0111,   STORE ROW VALUE*
12042 J2 12005 000-5               SM    SIZE-1,5,010*
12054 KO 12253 1200N               TF    COL,SIZE-1,0111,   STORE COLUMN VALUE*
12066 L3 12565 00000               CF    ZERO-9,,0*
12078 L3 12515 00000               CF    ZERO-59,,0*
12090 L3 12465 00000               CF    ZERO-109,,0*
12102 KO 12414 12574               TF    STORE,ZERO,01,     CLEAR PUNCH AREA*
12114 L2 12241 00000               SF    ROW-2,,0*
12126 PL 12259 12243               TNF   STORE-155,ROW,01,  STORE ROW NUMBER IN COLS. 1 THRU
12138 L2 12251 00000               SF    COL-2,,0*
12150 PL 12267 12253               TNF   STORE-147,COL,01,  STORE COLUMN NUMBER IN COLS. 5 TH
12162 J5 12271 00001               TDM   STORE-143,1,0,     STORE NUMBER 1 IN COL. 9*
12174 J5 12270 00007               TDM   STORE-144,7,0*
12186 L9 12255 00400      PUNCH    WACD  STORE-159,,0*
12198 J2 12005 000-1               SM    SIZE-1,1,010*
12210 L3 12005 00000               CF    SIZE-1,,0*
12222 M9 12576 00000               B     EXIT,,0*
12243    00010            ROW      DS    10*
12253    00010            COL      DS    10*
12414    00161            STORE    DS    161*
12464    00050                     DC    50,0000000000000000000000000000000000000000000000
00000-000000000000000000000000000000000000000000000*      -6J2415J2465-
12514    00050                     DC    50,0000000000000000000000000000000000000000000000
00000-000000000000000000000000000000000000000000000*      -6J2465J2515-
12564    00050                     DC    50,0000000000000000000000000000000000000000000000
00000-000000000000000000000000000000000000000000000*      -6J2515J2565-
12574    00010            ZERO     DC    10,0000000000*
00000-000000000*                                          -6J2565J2575-
12576 M9 1200N 00000      EXIT     B     SIZE-1,,06*
12006                              DENU  12006*
00000  L6000000005004900000*                              -8-0096-0115-
3600100005003600172005003600244005003600316005003600000000500             -
```

1.0.0-2

B-365

1.0.0.2

```
I-ADD OP                P/L LABEL   OP       OPERANDS

000000000000102030400020406080003060902100408021610050015102006021814200‡ -
704112820080614223009081726300000000050607080900121416181518112427202421‡ -
822363520353045403632484455324946536048465462754453627180123456789123456‡ -
789-23456789-J3456789-JK456789-JKL56789-JKLM6789-JKLMN789-JKLMN089-JKLMN‡ -
M80000000000049J20060P9-JKLMNOPQ‡     L10038800019M9000000000M90003600000  -
```

Appendix C

OPERATING PROCEDURES FOR LINEAR PROGRAMMING

CONTENTS

Appendix C

OPERATING PROCEDURES FOR LINEAR PROGRAMMING

There are several considerations in the fairing of a surface using linear programming which may have a bearing on the final result attained. The following is a short discussion of these.

SENSE OF COORDINATE SYSTEM

The coordinate system employed in the fairing procedure is very general. The origin of each surface faired is at the intersection of the first waterline and the first station in the surface. Thus it may be at any one of the four corners of the surface, depending on which station and which waterline is chosen as "first."

For surfaces without profiles it is generally more convenient to choose the origin to be at the corner with the least value for the offset. Since this offset becomes the constant for the equation and is subtracted from the other offsets, this allows the remaining portion of the other offsets to be positive. On most ships the origin then becomes the lowest waterline on the surface and the station furthest forward if the surface is forward on the ship, or aft if the surface is aft on the ship.

For surfaces with profiles, the origin of the surface must be the same as the origin of the profile equation. Therefore, for the bow the origin of both the surface and profile is usually made the intersection of the forward perpendicular and the baseline; aft, the origin is the intersection of the aft perpendicular and the baseline.

The positive sense of the coordinates is always along the surface. If the origin is at the lower-right corner, for instance, the positive x direction is to the left, and the positive z direction is up.

BASIC FEASIBLE SOLUTION

Most linear programming codes, such as LP-90 and the code for the IBM-1620, have two modes of operation. That is, when they first start to solve a problem the Phase I Algorithm is used to find a first feasible solution, without regard to optimality; and after finding this, Phase II is used to find an optimum solution.

In larger problems the search for the first basic feasible solution is time consuming and, due to the increase in the number of calculations, encourages the buildup of round-off error. It has been found advantageous then, in these larger problems, to provide the program with a basic feasible solution so the optimization can begin at once.

This is done by producing in the matrix an identity matrix of rank equal to the number of rows in the matrix. The identity matrix is identified to the program for use as the first basic feasible solution.

Each of the SMOG programs has a routine for generating this basis and identifying it to the linear program. In SMOG 3 and SMOG 4, this routine is self-contained. For SMOG 1 and SMOG 2, the routine of Appendix K is used.

INVERSION

During the solution of a linear programming problem, difficulties may arise from three sources: first, from the increase in number of non-zero elements in the matrix, which causes the iteration times to become greater.

the origin is at the lower-right corner, for instance, the positive x direction is to the left, and the positive z direction is up.

BASIC FEASIBLE SOLUTION

Most linear programming codes, such as LP-90 and the code for the IBM-1620, have two modes of operation. That is, when they first start to solve a problem the Phase I Algorithm is used to find a first feasible solution, without regard to optimality; and after finding this, Phase II is used to find an optimum solution.

In larger problems the search for the first basic feasible solution is time consuming and, due to the increase in the number of calculations, encourages the buildup of round-off error. It has been found advantageous then, in these larger problems, to provide the program with a basic feasible solution so the optimization can begin at once.

This is done by producing in the matrix an identity matrix of rank equal to the number of rows in the matrix. The identity matrix is identified to the program for use as the first basic feasible solution.

Each of the SMOG programs has a routine for generating this basis and identifying it to the linear program. In SMOG 3 and SMOG 4, this routine is self-contained. For SMOG 1 and SMOG 2, the routine of Appendix K is used.

INVERSION

During the solution of a linear programming problem, difficulties may arise from three sources: first, from the increase in number of non-zero elements in the matrix, which causes the iteration times to become greater.

Second, because of a buildup of round-off error, some of the numbers may lose significance. Third, from the buildup of the number of transformations in the product form of the inversion, the time per iteration increases.

Most linear programming routines which are meant to handle large problems, such as LP-90, have provision for inverting the basis matrix. This inversion, carried out at intervals during the solution of the problem, will be extremely helpful in reducing all of the difficulties mentioned above.

The best method for choosing the inversion interval seems to be that of simply keeping a running check on the time per iteration. The iteration time will be found to increase slowly for several iterations, then to increase more rapidly. At this point it is time to invert. On surfaces of eighty to one-hundred offsets this usually occurs at about every thirty iterations. It is not efficient to invert in any case at intervals of less than twenty to twenty-five iterations, because the inversion time cancels any savings in iteration time.