GROUP STIFFNESSES AND GROUP FIXED END STRESSES IN STRUCTURAL ANALYSIS

By

MOHAMMED ABDUL RAWOOF

Bachelor of Engineering

Osmania University

Hyderabad, A.P., India

1970

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
MASTER OF SCIENCE
December, 1973

GROUP STIFFNESSES AND GROUP FIXED END

STRESSES IN STRUCTURAL ANALYSIS

Thesis Approved:

Thesis Adviser

Dean of the Graduate College

ACKNOWLEDGMENTS

The author wishes to express his deep and sincere appreciation to his major adviser, Dr. R. K. Munshi, for providing the guidance, inspiration and valuable encouragement throughout the graduate studies. His friendship has been invaluable.

Appreciation is expressed to Dr. W. P. Dawkins for his suggestions and for serving as a committee member.

The author wishes to express his sincere gratitude to Dr. D. S. Ellifritt, a committee member, for his careful reading of the manuscript and especially for the opportunity he provided to assist him as a graduate teaching assistant.

Also appreciated are the interest and cooperation of all his friends.

Most importantly, sincere thanks and appreciation go to his respected parents for their continued interest and constant encouragement. The author cannot thank them sufficiently.

Gratitude is also extended to Ms. Charlene Fries for her careful and accurate typing of this manuscript.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

General

The direct procedure for analysis of structural systems by the stiffness matrix method is well suited for programming the electronic digital computer to analyze structures of moderate size. Unfortunately, for large structural systems, having a high degree of kinematic indeterminacy, this direct procedure can become quite cumbersome due to insufficient addressable core storage in the computer. A segmentation of the program with temporary storage of data on auxiliary storage devices such as magnetic tapes or discs becomes necessary. The use of auxiliary storage devices generally requires a large amount of computer time for transferring data into and out of the computer memory. The utilization of auxiliary storage results in increased cost to solve a problem. To eliminate the need for auxiliary storage, a large structure may be analyzed by dividing it into parts. These parts may be referred to as substructures.

A substructure may be a single bar member or it may be a large unit consisting of a group (or subassemblage) of members. The interactions between such groups of members at connection joints play a role which is similar to the interactions of individual members framing into the joints. Equilibrium equations for the connection joints can be solved for the unknown displacements that are common to two or more substructures framing into those joints.

Historical Review

The technique of working with substructures when a structural system contains too many unknowns to be solved for has successfully been applied by Weaver (2), Beaufait et al. (3) and Wang (4) to small plane frames and plane trusses.

The concept of solving structural systems in terms of substructures and development of group stiffenesses is discussed in some detail in recent books such as Tuma and Munshi (1) and Jamal J. Azar (8).

The successful application of matrix structural analysis using substructures, by Przemieniecki (5), Rubinstein (6) and Meek (7), demonstrated the feasibility of using substructures.

Objective of the Study

Large structural systems often have repetitive geometry, i.e., they are assembled together using identical subsystems. Analysis by substructures can be of definite advantage in such cases.

The primary objective of this study is to investigate the possibility of the extension of the application of group stiffnesses to two specific problems. The first is the establishment of end stiffnesses of a planar polygonal bar system (Figure 3), and the second is the solution of a truss dome (Figure 4).

Scope of the Study

The properties of a polygonal bar are derived by starting with two bars connected at a joint. Equations are reviewed to see how the middle joint in the two bar system can be eliminated from the calculations and the properties of the two bar system expressed as if it were an equivalent single bar. This equivalent bar can in turn be combined with the next bar in the polygon and the process repeated. All interior joints in the polygonal bar can thus be eliminated from calculations. A computer program is written which accomplishes this and gives the end stiffnesses as well as fixed-end stress resultants of a planar polygonal bar.

The truss dome discussed is made up of six identical segments. The group stiffnesses of each segment are first derived by eliminating all its interior joints from the calculations. The solution of the dome is then shown synthesized using these substructure properties. Again a computer program is prepared to illustrate numerical application.

A summary and conclusion drawn from the study are recorded in the last chapter.

CHAPTER II

SINGLE BRANCH GROUP

Theory

For polygonal shape bars and frames with a large number of joints, it serves to an advantage to introduce the concept of single branch group stiffness. Group stiffnesses can be defined as the stiffnesses of a single equivalent bar to replace a given group of bars. A single branch group can best be illustrated by the polygonal frame shown in Figure 1(a).

Such polygonal bars and frames where two bars frame into each joint are called single branch systems. The development of group stiffnesses of such systems is as follows: first, two bars are taken and the joint formed by these bars is eliminated as shown later in this chapter. The group stiffnesses and group load functions obtained for this two bar system then represent the corresponding values of a single equivalent bar. Then the next bar is added to the single equivalent bar and the new joint thus formed is eliminated and the group stiffnesses and group load functions are obtained which again can represent a single equivalent bar. The procedure can be repeated for any number of bars. It may be noted that no matter how many bars are added (one at a time) the group stiffness matrix will always represent the system as a single equivalent bar and the size of the stiffness matrix will always be 6 x 6 (planar frames) and the size of the group end stress vector 6 x 1.

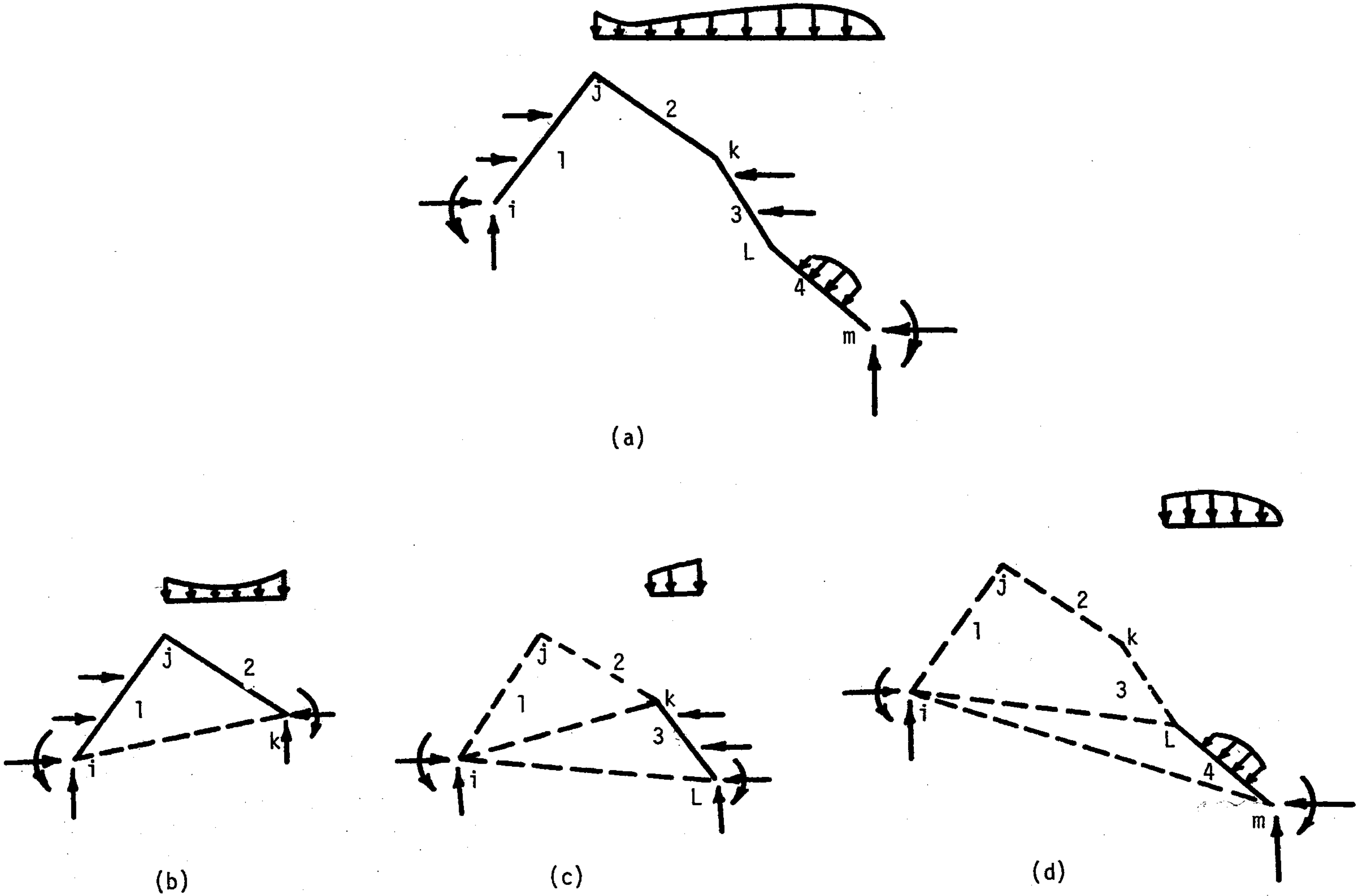

Figure 1. Single Branch Polygonal Frame

To derive the expressions for group stiffnesses, two consecutive bars, ij and jk as shown in Figure 1(b), are considered.

The joint equilibrium relations for joints i, j and k in these two bars are written as follows:

$$K_i^o \Delta_i^o + K_{ij}^o \Delta_j^o + F\sigma_i^o = W_i^o \tag{2.1a}$$

$$K_{ij}^o \Delta_i^o + K_j^o \Delta_j^o + K_{jk}^o \Delta_k^o + F\sigma_j^o = W_j^o \tag{2.1b}$$

$$K_{kj}^o \Delta_j^o + K_k^o \Delta_k^o + F\sigma_k^o = W_k^o \tag{2.1c}$$

in which K_{ij}^o is a typical segmental stiffness submatrix, the coefficients of which are the end stresses (stress resultants) induced at j due to respective unit displacements applied at i, and

$$K_j^o = K_{jj,i}^o + K_{jj,k}^o, \text{ etc.}$$

Δ_i^o, Δ_j^o, Δ_k^o are deformation vector values such that

$$\{\Delta_j^o\} = \{\delta_{jx}^o, \delta_{jy}^o, \theta_{jz}^o\}, \text{ etc.}$$

$F\sigma_i^o$, $F\sigma_j^o$, $F\sigma_k^o$ are the total fixed end stress vector values such that

$$F\sigma_j^o = F\sigma_{ji}^o + F\sigma_{jk}^o$$

and

$$\{F\sigma_{ji}^o\} = \{FN_{jix}^o, FN_{jiy}^o, FM_{jiz}^o\} \text{ etc., due to loads.}$$

W_i^o, W_j^o, W_k^o are applied joint load vector values such that

$$W_j^o = W_{ji}^o + W_{jk}^o$$

and

$$\{W_{ji}^o\} = \{P_{jix}^o, P_{jiy}^o, Q_{jiz}^o\}.$$

From Equation (2.1b), Δ_j^o is solved for in terms of the remaining matrices.

$$\Delta_j^o = -K_j^{o)-1} \{K_{ji}^o \Delta_i^o + K_{jk}^o \Delta_k^o + F\sigma_j^o - W_j^o\}. \tag{2.2a}$$

Δ_j^o is now eliminated from Equations (2.1a) and (2.1c) by substituting for it.

$$K_i^o \Delta_i^o + K_{ij}^o \{-K_j^{o)-1}(K_{ji}^o \Delta_i^o + K_{jk}^o \Delta_k^o + F\sigma_j^o - W_j^o)\} + F\sigma_i^o = W_i^o \tag{2.2b}$$

$$K_{kj}^o \{-K_j^{o)-1}(K_{ji}^o \Delta_i^o + K_{jk}^o \Delta_k^o + F\sigma_j^o - W_j^o)\} + K_k^o \Delta_k^o + F\sigma_k^o = W_k^o \tag{2.2c}$$

These equations can be rearranged and written as

$$K_i^{o)G} \Delta_i^o + K_{ik}^{o)G} \Delta_k^o + F\sigma_i^{o)G} = W_i^o \tag{2.3a}$$

$$K_{ki}^{o)G} \Delta_i^o + K_k^{o)G} \Delta_k^o + F\sigma_k^{o)G} = W_k^o \tag{2.3b}$$

in which

$$K_i^{o)G} = K_i^o - K_{ij}^o K_j^{o)-1} K_{ji}^o \tag{2.3c}$$

$$K_{ik}^{o)G} = -K_{ij}^o K_j^{o)-1} K_{jk}^o \tag{2.3d}$$

$$K_{ki}^{o)G} = -K_{kj}^o K_j^{o)-1} K_{ji}^o \tag{2.3e}$$

$$K_k^{o)G} = K_k^o - K_{kj}^o K_j^{o)-1} K_{jk}^o \tag{2.3f}$$

are the group stiffnesses (end stiffnesses of the bar ijk, acting as a single unit) and

$$F\sigma_i^{o)G} = F\sigma_i^o - K_{ij}^o K_j^{o)-1} \{F\sigma_j^o - W_j^o\} \tag{2.3g}$$

$$F\sigma_k^{o)G} = F\sigma_k^o - K_{kj}^o \, K_j^{o)-1} \, \{F\sigma_j^o - W_j^o\} \qquad (2.3h)$$

are the group fixed end stress vectors.

Equations (2.1a) through (2.1c) have thus been modified as if there is a single bar "ik". The next bar "kl" is joined at k as shown in Figure 1(c), and the equilibrium equations are set up for joint i, k and l in the equivalent frame ikl.

$$K_i^{o)G} \Delta_i^o + K_{ik}^{o)G} \Delta_k^o + F\sigma_i^{o)G} = W_i \qquad (2.4a)$$

$$K_{ki}^{o)G} \Delta_i^o + \Sigma K_k^o \Delta_k^o + K_{kl}^o \Delta_l^o + \Sigma F\sigma_k^o = W_k^o \qquad (2.4b)$$

$$K_{lk}^o \Delta_k^o + K_l^o \Delta_l^o + F\sigma_l^o = W_l^o \qquad (2.4c)$$

in which

$$\Sigma K_k^o = K_k^{o)G} + K_{kk,l}^o$$

and

$$\Sigma F\sigma_k^o = F\sigma_k^{o)G} + F\sigma_{kk,l}^o$$

The new group stiffnesses for the group ijkl obtained by eliminating joint k can be written by comparison with Equations (2.3a) and (2.3b).

$$K_i^{o)G'} \Delta_i^o + K_{il}^{o)G} \Delta_l^o + F\sigma_i^{o)G'} = W_i^o \qquad (2.5a)$$

$$K_{li}^{o)G} \Delta_i^o + K_l^{o)G} \Delta_l^o + F\sigma_l^{o)G} = W_l^o \qquad (2.5b)$$

where

$$K_i^{o)G'} = K_i^{o)G} - K_{ik}^{o)G} \, K_k^{o)G)-1} \, K_{ki}^{o)G}$$

$$K_{il}^{o)G} = -K_{ik}^{o)G} \, K_k^{o)G)-1} \, K_{kl}^o$$

$$K_{li}^{o)G} = -K_{lk}^{o)G} \, K_k^{o)G)-1} \, K_{ki}^{o)G}$$

$$K_l^{o)G} = K_l^o - K_{lk}^o K_k^{o)G)-1} K_{kl}^o$$

are the new group stiffnesses at i and l of the bar ijkl, and

$$F\sigma_i^{o)G'} = F\sigma_i^{o)G} - K_{ik}^{o)G} K_k^{o)G)-1} \{F\sigma_k^o - W_k^o\}$$

$$F\sigma_l^{o)G} = F\sigma_l^o - K_{lk}^o K_k^{o)G)-1} \{F\sigma_k^o - W_k^o\}$$

are the new group fixed end stress vectors at i and l of the bar ijkl in which

$$K_k^{o)G)-1} \equiv \{K_k^{o)G} + K_{kk,l}^o\}^{-1}$$

In the same manner the joint l can also be eliminated from the bar system ijklm, i.e., the equivalent bar system ilm, and final force deformation relation at i and m of the system ijklm (Figure 1(d)) can be written as

$$K_i^{o)G''} \Delta_i^o + K_{im}^{o)G} \Delta_m^o + F\sigma_i^{o)G''} = W_i^o \tag{2.6a}$$

$$K_{mi}^{o)G} \Delta_i^o + K_m^{o)G} \Delta_m^o + F\sigma_m^{o)G} = W_m^o \tag{2.6b}$$

where the matrices $K_i^{o)G''}$, $K_{im}^{o)G}$, $K_{mi}^{o)G}$ and $K_m^{o)G}$ are the group stiffness matrices, and $F\sigma_i^{o)G''}$, $F\sigma_m^{o)G}$ are group fixed end stress vectors, all being related to the joints i and m only, when the joints j, k and l are free to displace.

Development of the Computer Program

The development of group stiffnesses and load functions for a single branch group such as a polygonal shape frame described earlier in this chapter has been programmed for numerical computation on a digital computer. The program No. 1 has been written in FORTRAN language and tested on the IBM 360-65 model.

The program generates the group stiffnesses and load functions. The results are printed in appropriate matrix form with proper headings. Two illustrative examples have been solved and the results were compared with the values given in reference (1).

The steps involved in the program are shown in Figure 2 as a flow chart.

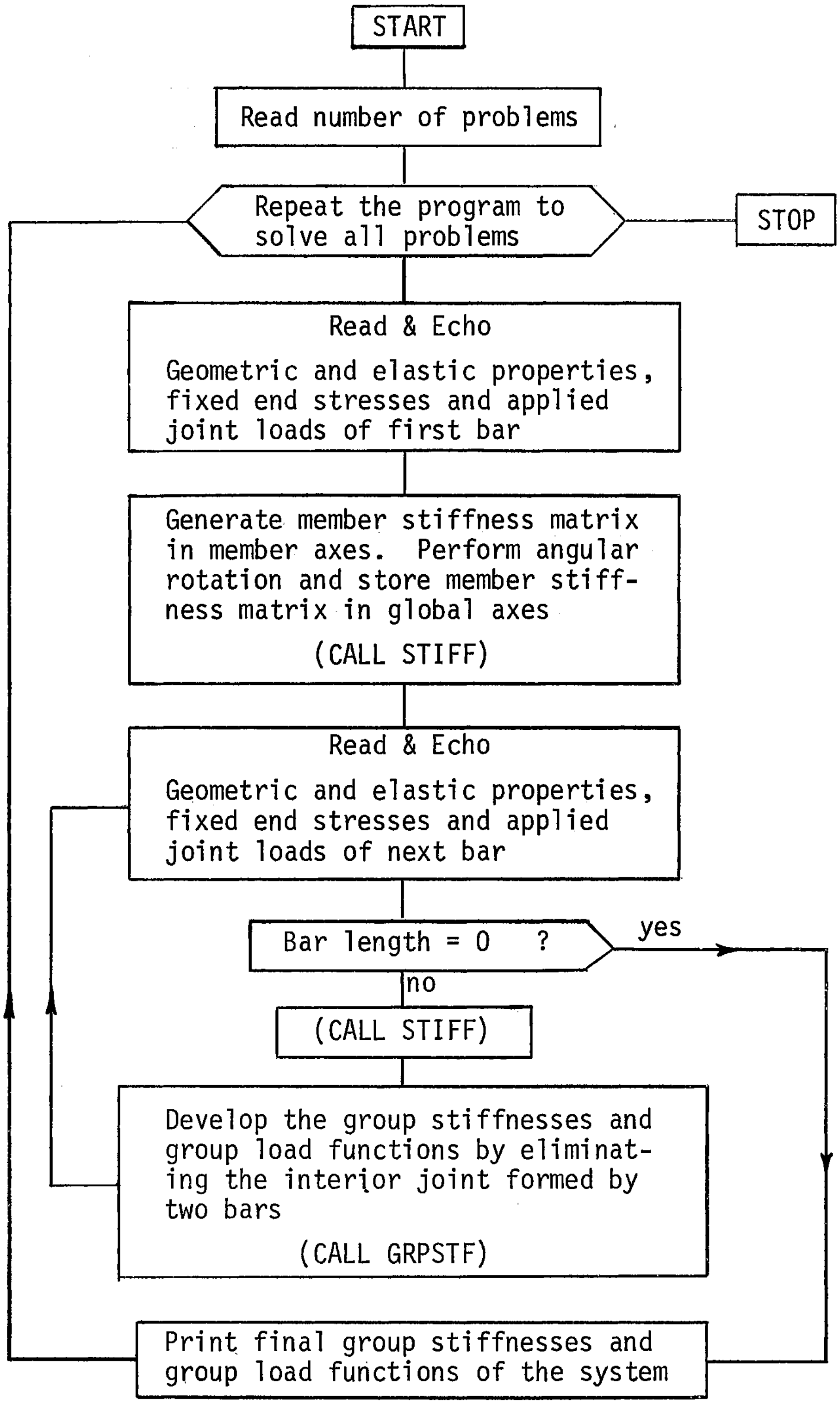

Figure 2. Flow Chart of Computer Program No. 1

CHAPTER III

MULTI-BRANCH GROUP

Theory

The technique of eliminating some unknowns and developing the stiffness matrix in terms of stresses due to certain unit displacements of the system leads to the generalized group stiffness matrix concept. It can best be illustrated by an example, such as the geodesic dome shown in Figure 3. Such complex frames and trusses are classified as multi-branch systems and their analysis can also be performed by developing group stiffnesses and group load functions. In this approach the system is conveniently partitioned into regions or substructures. A substructure can be defined as a structure restrained at the joints that are common to adjacent substructures and that connect the various substructures together. Once the substructures for a structural system have been defined, each substructure is treated independently for the loads applied within that region of the substructure and for the possible joint deformations. All the interior joints of each substructure are eliminated from calculations by using the technique discussed earlier in Chapter II. All substructures are then connected together at the boundaries by using the group stiffnesses and group load functions. The development of group stiffnesses and group load functions for a typical substructure in a multi-branch system is as follows.

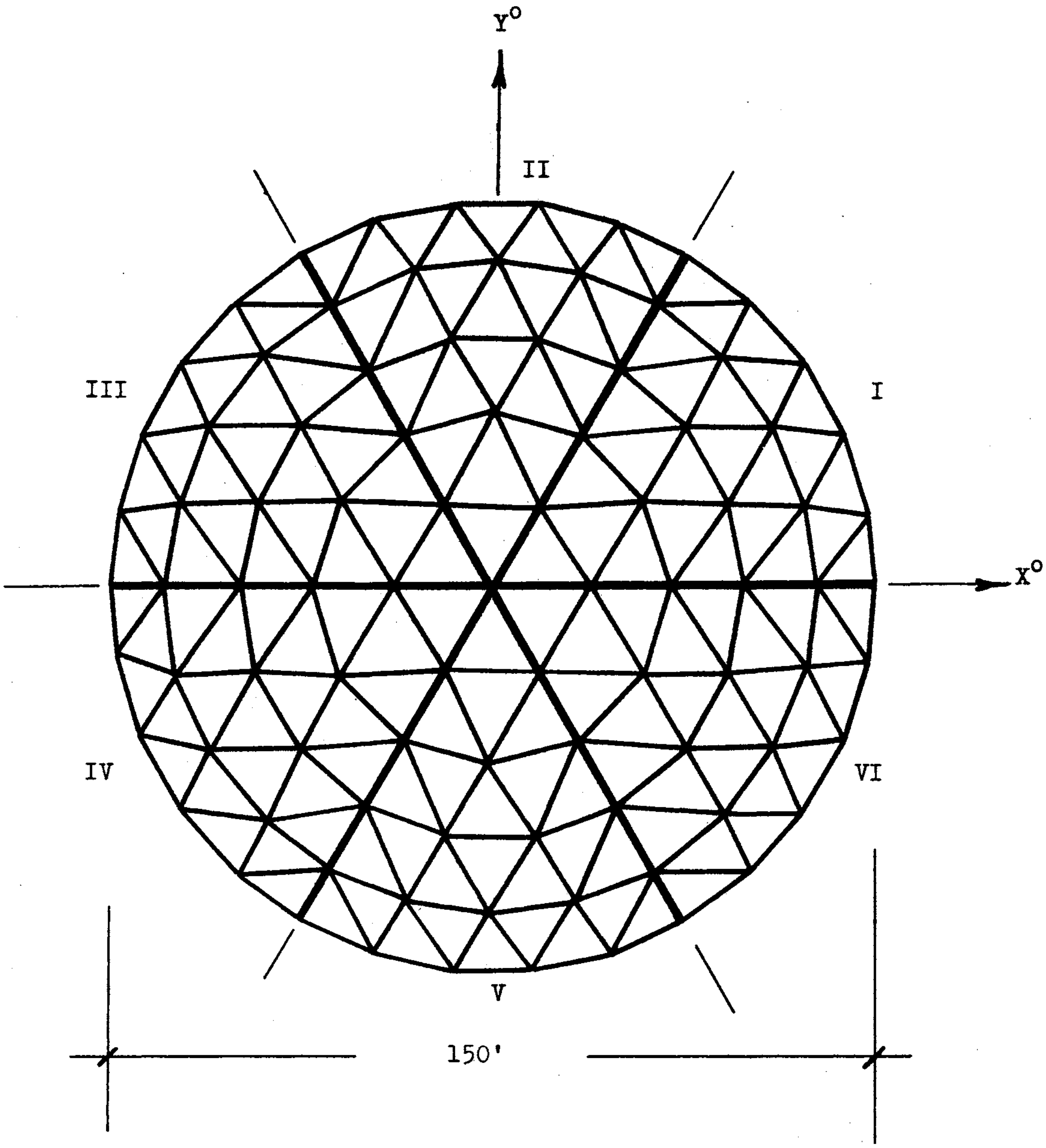

Figure 3. Geodesic Dome 150' Diameter, 45' Height

The equations of equilibrium at all joints with (any) degrees of freedom can be arranged in the following matrix form.

$$\begin{bmatrix} K_L^o & K_{LM}^o & K_{LR}^o \\ K_{ML}^o & K_M^o & K_{MR}^o \\ K_{RL}^o & K_{RM}^o & K_R^o \end{bmatrix} \begin{bmatrix} \Delta_L^o \\ \Delta_M^o \\ \Delta_R^o \end{bmatrix} + \begin{bmatrix} F\sigma_L^o \\ F\sigma_M^o \\ F\sigma_R^o \end{bmatrix} = \begin{bmatrix} W_L^o \\ W_M^o \\ W_R^o \end{bmatrix} \tag{3.1}$$

in which (see Figure 4) Δ_L^o and Δ_R^o are the deformation vector values at joints on the left and the right boundaries, respectively, such that

$$\{\Delta_L^o\} = \{\Delta_1^o, \Delta_2^o, \ldots, \Delta_5^o\};$$

Δ_M^o is the deformation vector of the interior joints such that

$$\{\Delta_M^o\} = \{\Delta_6^o, \Delta_7^o, \ldots, \Delta_{11}^o\};$$

K_{LM}^o is a typical stiffness submatrix, the coefficients of which are stress influence values at joints on the left (L) boundary due to unit deformation at interior joints (M); $F\sigma_L^o$, $F\sigma_M^o$, $F\sigma_R^o$ are the fixed end stress vector values such that

$$\{F\sigma_L^o\} = \{F\sigma_1^o, F\sigma_2^o, \ldots, F\sigma_5^o\};$$

and W_L^o, W_M^o, W_R^o are the applied joint load vector values such that

$$\{W_L^o\} = \{W_1^o, W_2^o, \ldots, W_5^o\}.$$

In Equation (3.1) Δ_M^o corresponds to interior joints to be eliminated. Following a procedure similar to the one used in Chapter II,

$$\Delta_M^o = -K_M^{o)-1} \{K_{ML}^o\Delta_L^o + K_{MR}^o\Delta_R^o + F\sigma_M^o - W_M^o\} \tag{3.2}$$

Δ_M^o can now be eliminated from the first and the third rows of matrix Equation (3.1) by substituting for it and the results rearranged as follows:

$$K_L^{o)G} \Delta_L^o + K_{LR}^{o)G} \Delta_R^o + F\sigma_L^{o)G} = W_L^o \quad (3.3)$$

$$K_{RL}^{o)G} \Delta_L^o + K_R^{o)G} \Delta_R^o + F\sigma_R^{o)G} = W_R^o \quad (3.4)$$

in which

$$K_L^{o)G} = K_L^o - K_{LM}^o K_M^{o)-1} K_{ML}^o \quad (3.5a)$$

$$K_{LR}^{o)G} = K_{LR}^o - K_{LM}^o K_M^{o)-1} K_{MR}^o \quad (3.5b)$$

$$K_{RL}^{o)G} = K_{RL}^o - K_{RM}^o K_M^{o)-1} K_{ML}^o \quad (3.5c)$$

$$K_R^{o)G} = K_R^o - K_{RM}^o K_M^{o)-1} K_{MR}^o \quad (3.5d)$$

are the multi-branch group stiffness matrices, and

$$F\sigma_L^{o)G} = F\sigma_L^o - K_{LM}^o K_M^{o)-1} (F\sigma_M^o - W_M^o) \quad (3.6a)$$

$$F\sigma_R^{o)G} = F\sigma_R^o - K_{RM}^o K_M^{o)-1} (F\sigma_M^o - W_M^o) \quad (3.6b)$$

are the multi-branch group fixed end stress vectors due to loads or other causes.

These modified functions are used to relate the interaction with adjacent substructures at the connection joints.

The given system can now be solved by setting up the system equilibrium stiffness matrix equation for just the joints at the boundaries of the substructures. Having found the boundary joint deformations, the deformations at interior joints of each substructure can be found by using Equation (3.2). Finally, all member end actions and support reactions are computed from member force-deformation relationship.

Development of the Computer Program

The elimination procedure of interior joints and the development of multi-branch group stiffnesses and group load functions described earlier in this chapter can be programmed for solution on a digital computer. Obviously there is an infinite variety of complex structures that can be classified as multi-branch systems. Therefore, no attempt is made to write a completely general program. However, a program is written to analyze a geodesic truss dome such as the one shown in Figure 3, by using substructures. The program generates the group stiffnesses and group load functions for one of the six identical substructures of the geodesic truss dome, Figure 4. The system equilibrium matrix equation is solved for the boundary joints of all substructures. The interior joints, member forces and support reactions are then computed. Various steps involved in the program are presented in the flow chart, Figure 5.

The computer program No. 2 has been written in FORTRAN language and tested on IBM 360-65 model operated by the Oklahoma State University Computer Center.

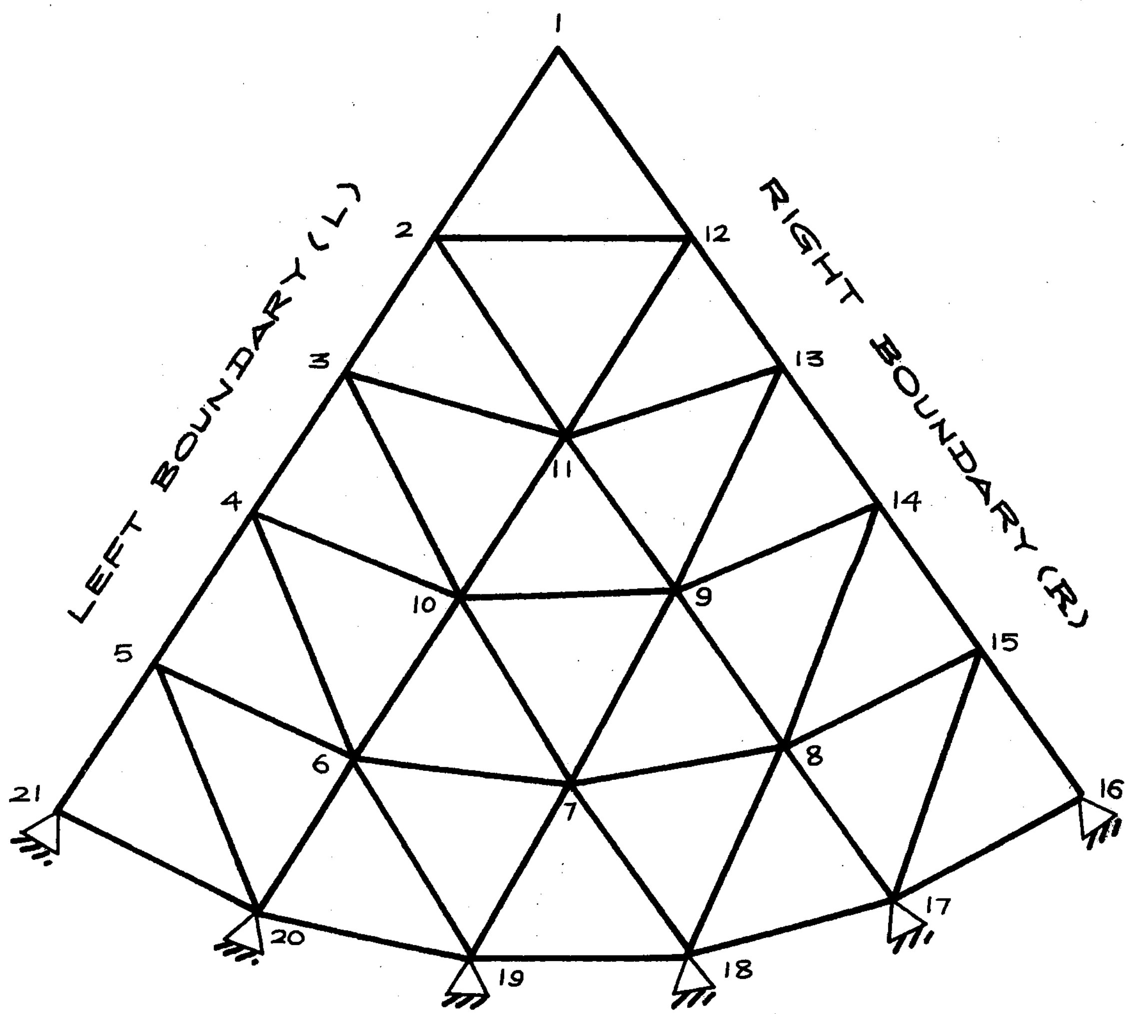

Figure 4. A Typical Substructure of the Dome

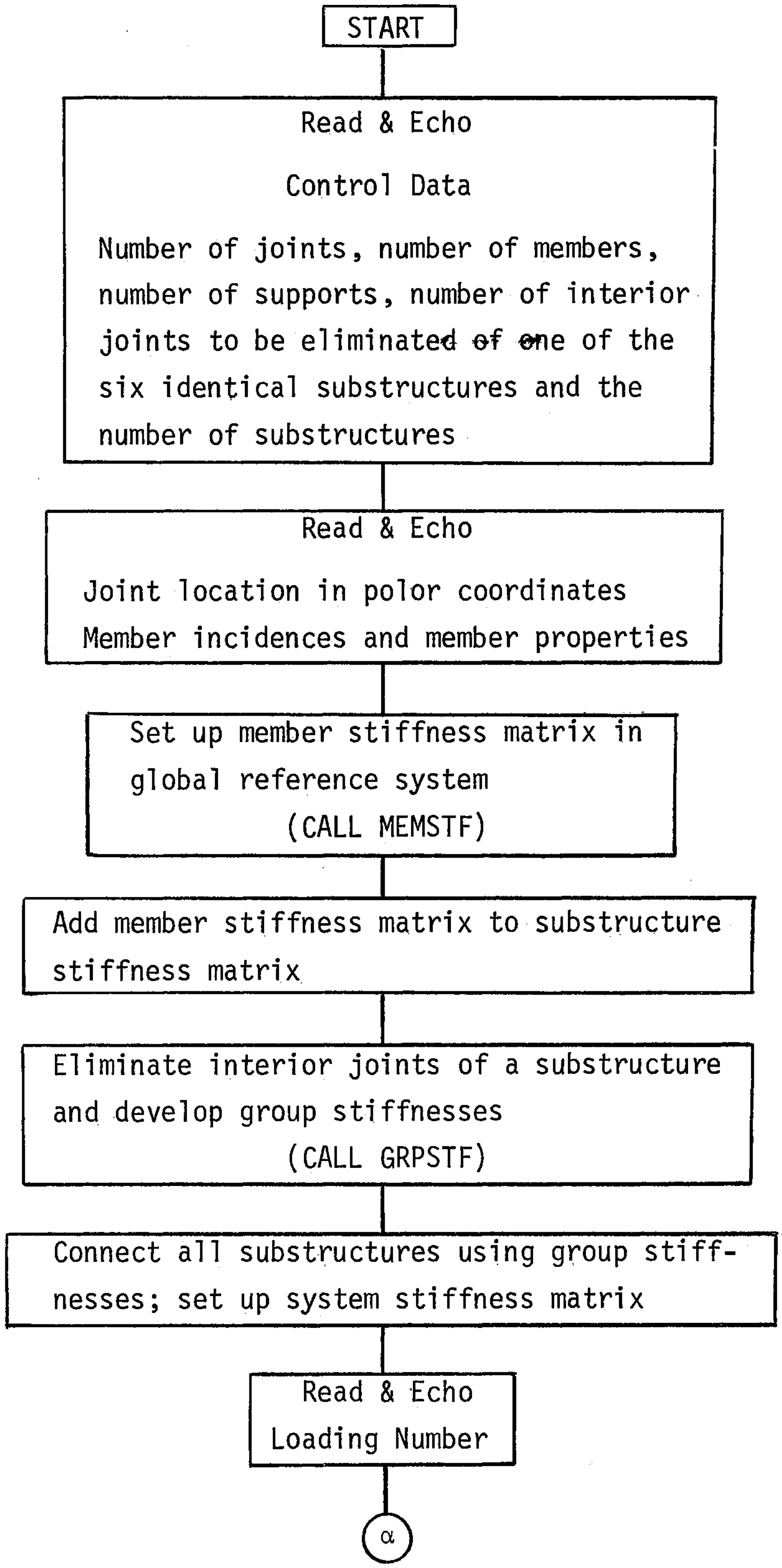

Figure 5. Flow Chart of Computer Program No. 2

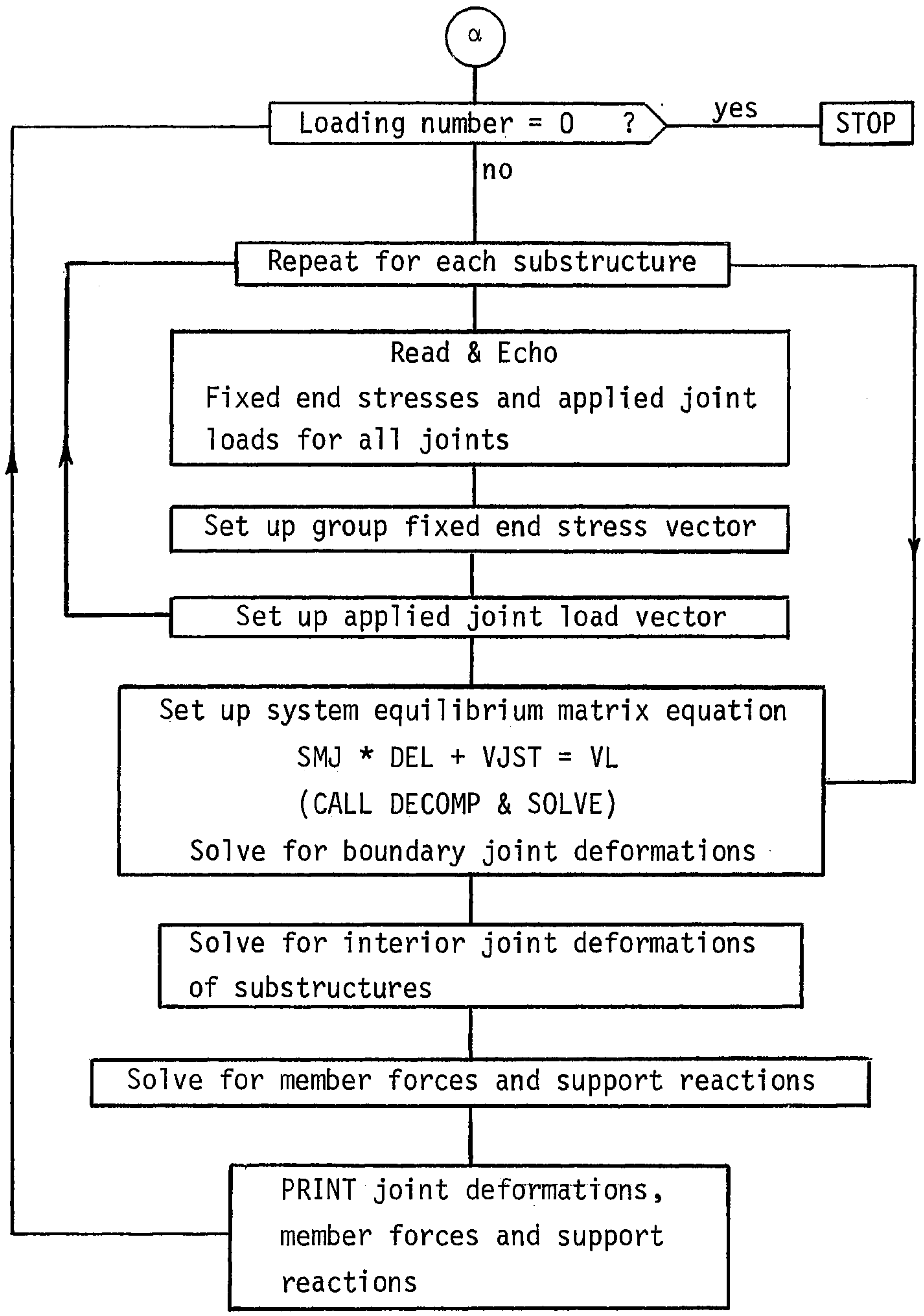

Figure 5. (Continued)

CHAPTER IV

ILLUSTRATIVE EXAMPLES

Three numerical examples are presented to illustrate the technique of working with substructures and the development of group stiffnesses and group load functions.

The first two examples demonstrate the application to single branch structures.

Example 1

A two bar system of constant cross section shown in Figure 6 is considered. It is desired to verify the group end stiffnesses and group fixed end stresses obtained by the procedure outlined in Chapter II.

It is assumed that,

$EI = 290{,}000 \text{ k-ft}^2$,

$EA = 1{,}073{,}000.00 \text{ k}$.

Member stiffness matrix for each bar in its local axes is

$$K = \begin{bmatrix} 53650 & 0 & 0 & -53650 & 0 & 0 \\ 0 & 435 & 4350 & 0 & -435 & 4350 \\ 0 & 4350 & 58000 & 0 & -4350 & 29000 \\ -53650 & 0 & 0 & 53650 & 0 & 0 \\ 0 & -435 & -4350 & 0 & 435 & -4350 \\ 0 & 4350 & 29000 & 0 & -4350 & 58000 \end{bmatrix}$$

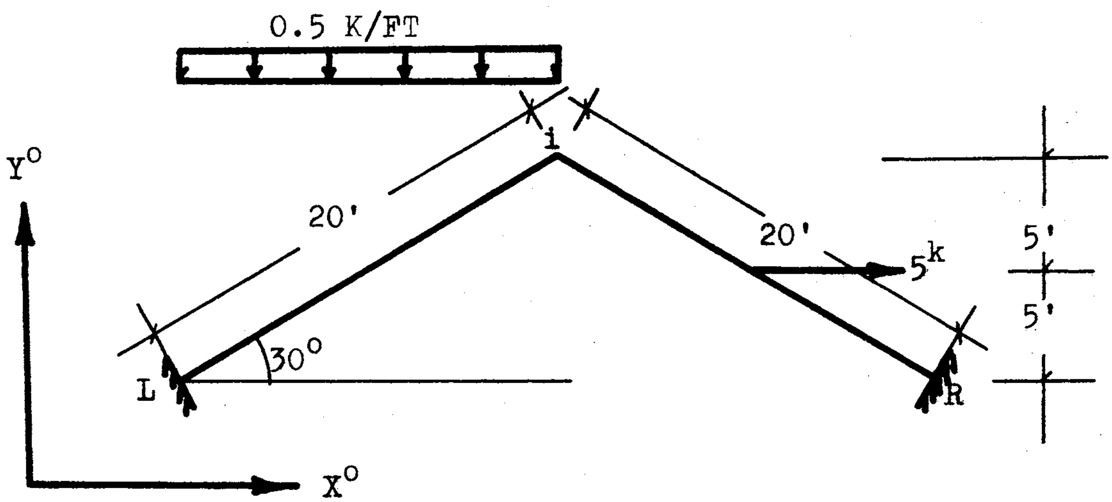

Figure 6. Two-Bar System, Symmetrical Bent Bar

Also the fixed end stress vectors in local axes are

$$F\sigma^{o}_{Li} = (0.0,\ 4.33,\ 12.52)$$

$$F\sigma^{o}_{iL} = (0.0,\ 4.33,\ -12.52)$$

$$F\sigma^{o}_{iR} = (-2.5,\ 0.0,\ -6.25)$$

$$F\sigma^{o}_{Ri} = (-2.5,\ 0.0,\ 6.25).$$

The group stiffness and group load functions obtained by using these data in Computer Program No. 1 (Appendix) are given in Table I.

The results are checked out and compare very well with those computed from Tables 10-8 and 10-9 of Tuma and Munshi (1), which are presented in Table II.

Example 2

The elastic properties (end stiffnesses) and load functions (fixed end stresses) for a circular constant section bar are to be computed. To illustrate the application of the program developed in Chapter II, this bar is replaced by a polygonal bar consisting of the chord lengths connecting points located at 1/10 the total length along the curve, Figure 7.

The same values of EI and EA as used in Example 1 are also used for this 10-bar system.

The segmental stiffness matrix in member axes for a typical bar is as shown below.

TABLE I

EXAMPLE 1, RESULTS FROM COMPUTER PROGRAM NO. 1

Left End			Right End		
869.8	0.0	-4348.9	-869.8	0.0	4348.9
0.0	72.5	1255.7	0.0	-72.5	1255.7
-4348.9	1255.7	50744.6	-4348.9	-1255.7	-7244.9
-869.8	0.0	4348.9	869.8	0.0	-4348.9
0.0	-72.5	-1255.7	0.0	72.5	-1255.7
4348.9	1255.7	-7244.9	-4348.9	-1255.7	50744.6
$F\sigma_L^{o)G}$ = {2.5, 6.6, 17.2}			$F\sigma_R^{o)G}$ = {-7.5, 2.1, 10.9}		

TABLE II

EXAMPLE 1, RESULTS FROM REFERENCE (1)

Left End			Right End		
870.0	0.0	-4350.0	-870.0	0.0	4350.0
0.0	72.5	1255.8	0.0	-72.5	1255.8
-4350.0	1255.8	50750.0	4350.0	-1255.8	-7250.0
-870.0	0.0	4350.0	870.0	0.0	-4350.0
0.0	-72.5	-1255.8	0.0	72.5	-1255.8
4350.0	1255.8	-7250.0	-4350.0	-1255.8	50750.0
$F\sigma_L^{o)G}$ = {2.5, 6.6, 17.2}			$F\sigma_R^{o)G}$ = {-7.5, 2.1, 10.9}		

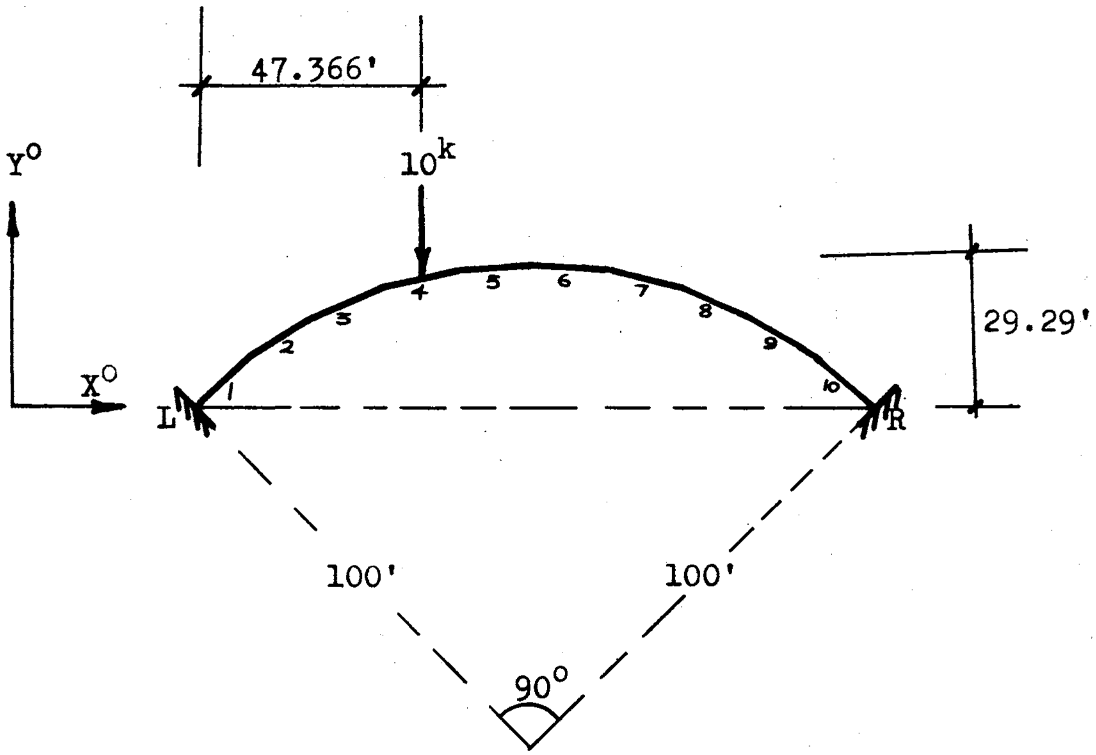

Figure 7. Ten-Bar System, Symmetrical Circular Bar

$$K = \begin{bmatrix} 6837880 & 0 & 0 & -6837880 & 0 & 0 \\ 0 & 901 & 7066 & 0 & -901 & 7066 \\ 0 & 7066 & 73923 & 0 & -7066 & 36962 \\ -6837880 & 0 & 0 & 6837880 & 0 & 0 \\ 0 & -901 & -7066 & 0 & 901 & -7066 \\ 0 & 7066 & 36962 & 0 & -7066 & 73923 \end{bmatrix}$$

Fixed end stress vectors for bar No. 4 are as follows:

$$F\sigma^{o}_{45} = (0.0,\ 5.0,\ 18.64)$$

$$F\sigma^{o}_{54} = (0.0,\ 5.0,\ -18.64).$$

The group stiffness matrix and group load functions are generated using Computer Program No. 1 (Appendix).

Table III shows these values which can be compared with the corresponding values computed from Tables 10-12 and 10-14 of Tuma and Munshi (1), shown in Table IV.

Example 3

A geodesic dome of base diameter 150 ft. and 45 ft. high (as shown in plan view in Figure 3) is analyzed by Computer Program No. 2. The dome structure is considered as a space truss and it consists of six identical substructures. All the joints at the base are assumed to be pinned end supports. Group stiffnesses and group load functions for a typical substructure are developed by using equations derived in Chapter III.

The dome is analyzed for a uniform gravity load of 1 k/sft on the actual area. Figure 8 shows a substructure with joint loads computed from respective tributary areas.

TABLE III

EXAMPLE 2, RESULTS FROM COMPUTER PROGRAM NO. 1

	Group Stiffnesses						Group Fixed End Stresses
	GLLR			GLR			Left End
	1	2	3	1	2	3	1
1	23.971302	0.000000	-458.714308	-23.971302	0.000000	458.714309	9.127295
2	0.000000	1.021301	72.217742	-0.000000	-1.021301	72.217742	7.318540
3	-458.714308	72.217742	15732.646697	458.714311	-72.217744	-5519.398024	23.042503
	GRL			GRRL			Right End
	1	2	3	1	2	3	1
1	-23.971302	-0.000000	458.714309	23.971299	0.000002	-458.714303	-9.127295
2	0.000000	-1.021301	-72.217742	0.000000	1.021301	-72.217742	2.681460
3	458.714309	72.217742	-5519.398023	-458.714294	-72.217751	15732.646664	72.123942

TABLE IV

EXAMPLE 2, RESULTS FROM REFERENCE (1)

Left End			Right End		
23.9	0	-461.09	-23.90	0	461.09
0	1.02	71.85	0	-1.02	71.85
-461.09	71.85	15835.75	461.09	-71.85	-5674.28
-23.9	0	461.09	23.90	0	-461.09
0	-1.02	-71.85	0	1.02	-71.85
461.09	71.85	-5674.28	-461.09	-71.85	15835.75
$F\sigma_L^{o)G}$ = {9.08, 7.31, 20.72}			$F\sigma_R^{o)G}$ = {-9.08, 2.69, 71.43}		

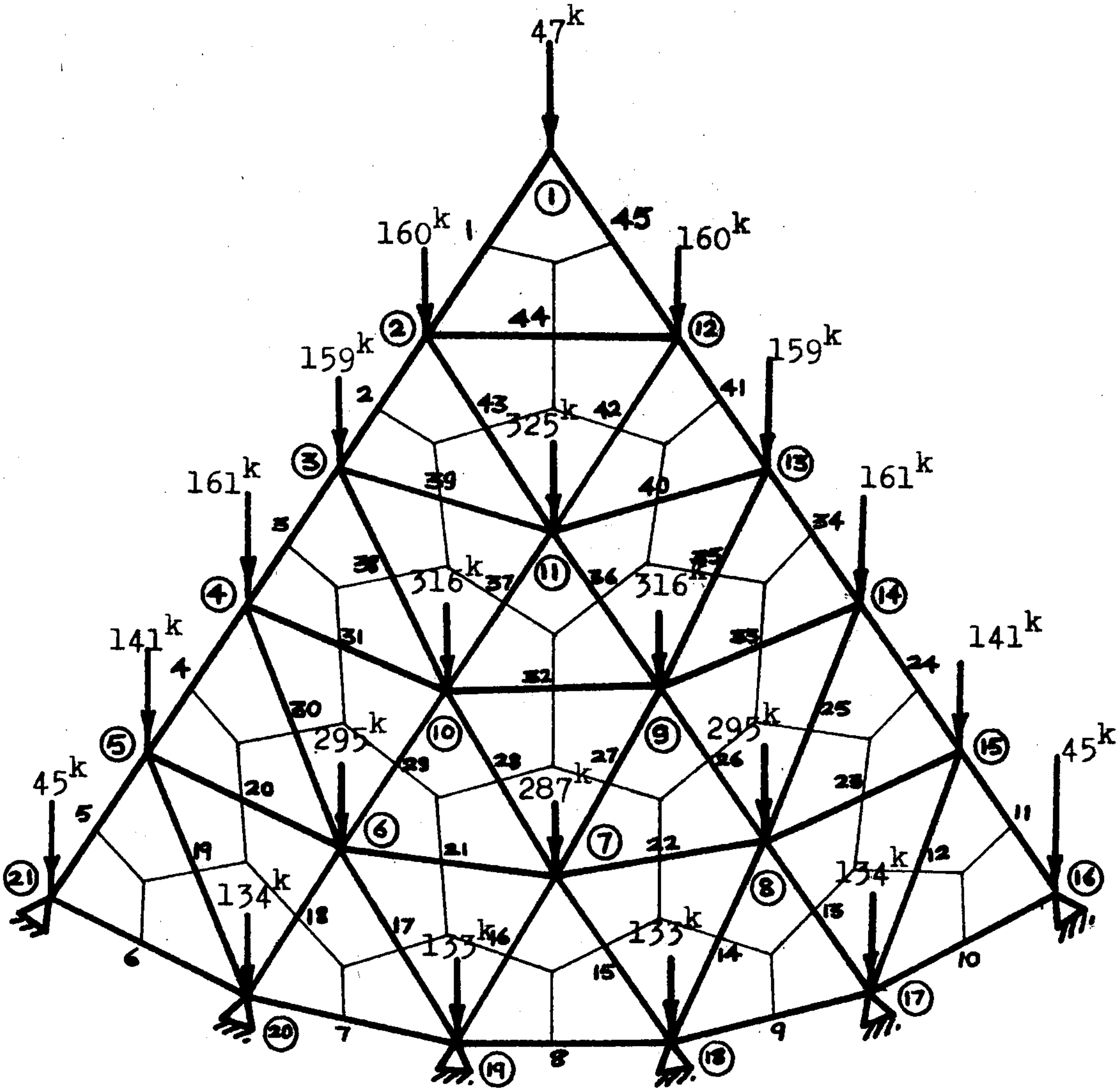

Figure 8. Substructure I of the Dome Shown in Figure 2

The system stiffness matrix is constructed by superposition of all transformed substructure group stiffness matrices. Fixed end stress vector is set up by the superposition of transformed group load functions of each substructure. The applied joint load vector is also set up by the superposition of all connecting joint loads with proper transformation. The system equilibrium matrix equation is written for those joints which are connecting adjacent substructures as shown in Figure 9. The unknown joint deformation vector is obtained by solving the system equilibrium matrix equation. The known deformations of each substructure are substituted in Equation (3.2), for solving for interior joint deformations. Finally, all member forces and reactions are computed by using the member force-deformation relationship.

All substructure boundary joint deformations are presented in Table V and the interior joint deformations of the substructure I are shown in Table VI. The final member stresses in a typical substructure are summarized in Table VII and the support reactions in Table VIII. The results are checked out and compared by solving the dome using STRUDL II.

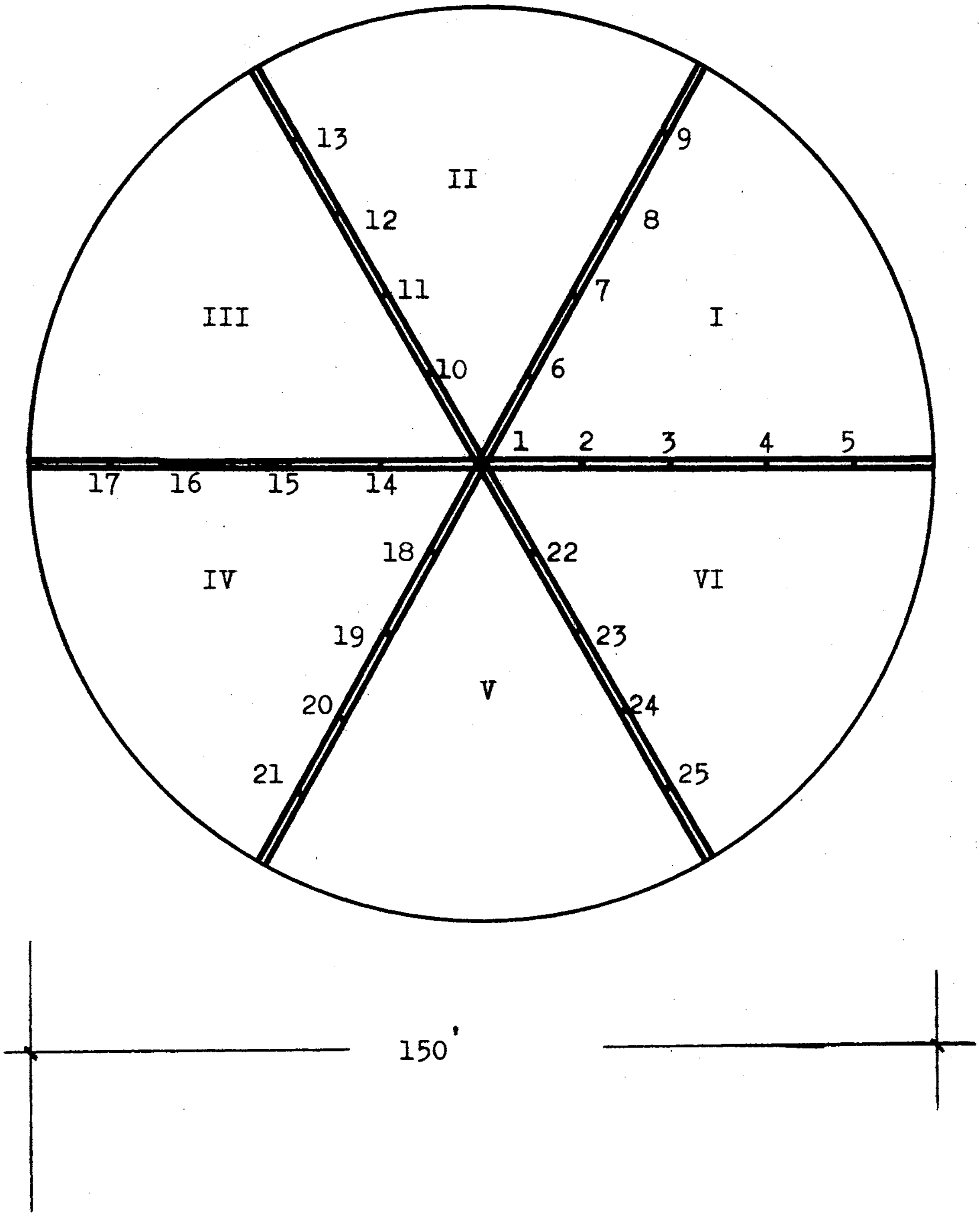

Figure 9. Substructure Boundary Joints

TABLE V

SUBSTRUCTURE BOUNDARY JOINT DEFORMATIONS, FIGURE 7

Joint Number	X-Disp.	Y-Disp.	Z-Disp.
1	0.000000	-0.000000	-0.216258
2	-0.031746	-0.000000	-0.261964
3	-0.057029	-0.000000	-0.216807
4	-0.070657	0.000000	-0.159325
5	-0.014584	0.000000	-0.042307
6	-0.015873	-0.027493	-0.261964
7	-0.028514	-0.049388	-0.216807
8	-0.035328	-0.061190	-0.159325
9	-0.007292	-0.012630	-0.042307
10	0.015873	-0.027493	-0.261964
11	0.028514	-0.049388	-0.216807
12	0.035328	-0.061190	-0.159325
13	0.007292	-0.012630	-0.042307
14	0.031746	-0.000000	-0.261964
15	0.057029	-0.000000	-0.216807
16	0.070657	-0.000000	-0.159325
17	0.014584	-0.000000	-0.042307
18	0.015873	0.027493	-0.261964
19	0.028514	0.049388	-0.216807
20	0.035328	0.061190	-0.159325
21	0.007292	0.012630	-0.042307
22	-0.015873	0.027493	-0.261964
23	-0.028514	0.049388	-0.216807
24	-0.035328	0.061190	-0.159325
25	-0.007292	0.012630	-0.042307

TABLE VI

INTERIOR JOINT DEFORMATIONS OF SUBSTRUCTURE I,
FIGURE 6

Joint Number	X-Disp.	Y-Disp.	Z-Disp.
6	0.034448	0.009763	-0.021846
7	0.046401	0.026790	-0.009544
8	0.025679	0.024951	-0.021846
9	-0.016904	-0.015073	-0.140511
10	-0.021505	-0.007103	-0.140511
11	-0.034995	-0.020205	-0.236507

TABLE VII

FINAL MEMBER STRESSES IN SUBSTRUCTURE I,
FIGURE 6

Member Number	From Joint	To Joint	Axial Force	Member Number	From Joint	To Joint	Axial Force
1	2	1	-435.5762185	23	8	15	58.2736876
2	3	2	-627.3663875	24	15	14	-647.1357546
3	4	3	-674.9366381	25	8	14	-211.9581489
4	5	4	-647.1357546	26	8	9	-590.4985033
5	21	5	-437.8402026	27	7	9	-425.8343988
6	21	20	0.0000000	28	7	10	-425.8343988
7	20	19	0.0000000	29	6	10	-590.4985033
8	19	18	0.0000000	30	6	4	-211.9581489
9	18	17	0.0000000	31	4	10	-261.2757901
10	17	16	0.0000000	32	10	9	-154.8369822
11	16	15	-437.8402026	33	9	14	-261.2757901
12	17	15	-281.5501713	34	14	13	-674.9366381
13	17	8	-596.6250395	35	9	13	-213.3368859
14	18	8	-443.5447351	36	9	11	-516.2025175
15	18	7	-520.7355366	37	10	11	-516.2025175
16	19	7	-520.7355366	38	10	3	-213.3368859
17	19	6	-443.5447351	39	3	11	-410.3527044
18	20	6	-596.6250395	40	11	13	-410.3527044
19	20	5	-281.5501713	41	13	12	-627.3663875
20	5	6	58.2736876	42	11	12	-323.0359605
21	6	7	198.2170523	43	11	2	-323.0359605
22	7	8	198.2170523	44	2	12	-522.3295297
				45	12	1	-435.5762185

TABLE VIII

SUPPORT REACTIONS FOR SUBSTRUCTURE I, FIGURE 6

Support Number	X-Force	Y-Force	Z-Force
16	-123.253118	-213.480663	361.854554
17	-356.654799	-315.889647	665.813964
18	-433.581653	-275.682385	714.758759
19	-455.538775	-237.651533	714.758759
20	-451.895859	-150.927293	665.813964
21	-246.506236	-0.000000	361.854554

CHAPTER V

SUMMARY AND CONCLUSIONS

Summary

The application of group stiffnesses for analyzing single branch systems (polygonal shape frames) and multi-branch systems (complex frames and trusses) is investigated in this study. Group end stiffnesses of a polygonal bar system is established by taking two examples of single branch systems. A two bar system of constant cross section is considered to develop and verify the group end stiffnesses and group fixed end stresses. Another example of a single branch system considered is a circular constant section bar. To illustrate the application of group stiffnesses, this bar is replaced by a polygonal bar consisting of the chord lengths connecting points located at 1/10 the total length along the curve. A computer program No. 1 (Appendix) is developed to compute the end stiffnesses as well as fixed end stress resultants of a planar polygonal bar. An attempt is also made to work with substructures in the case of multi-branch systems. A geodesic dome structure with six identical substructures is analyzed as a space truss dome. The group stiffnesses and group fixed end stresses are developed for a typical substructure and the same were used with proper axes transformation to synthesize the whole dome structure. The computer program No. 2 (Appendix) is written which accomplishes this and analyzes the dome and prints out the final joint deformations, member forces and support reactions. The same truss dome is also analyzed by using STRUDL II to verify the results of the computer program No. 2.

Conclusions

The investigation of the extension of the application of group stiffnesses to the illustrative examples showed that the concept of group stiffnesses and group fixed end stresses can be applied to plane and space structures with appreciable accuracy. Further, that it is easy and convenient to work with substructures by developing group stiffnesses and group fixed end stresses when the structural system has repetitive geometry. The first computer program can be easily modified to be suitable for a three dimensional, single branch system.

BIBLIOGRAPHY

(1) Tuma, J. J. and R. K. Munshi. "Advanced Structural Analysis." Schaum's Outline Series. New York: McGraw-Hill Book Company, 1971.

(2) Weaver, W., Jr. Computer Programs for Structural Analysis. Princeton, N. J.: D. Van Nostrand Co., Inc., 1967.

(3) Beaufait, F. W., et al. Computer Methods of Structural Analysis. Englewood Cliffs, N. J.: Prentice-Hall, Inc., 1970.

(4) Wang, C. K. Matrix Methods of Structural Analysis. Scranton, Pa.: International Text Book Co., 1970.

(5) Przemieniecki, J. S. "Matrix Structural Analysis of Substructures." AIAA Journal, Vol. 1, No. 1 (January 1963), 258-261.

(6) Rubinstein, M. F. Matrix Computer Analysis of Structures. Englewood Cliffs, N. J.: Prentice-Hall, Inc., 1966.

(7) Meek, J. L. Matrix Structural Analysis. New York: McGraw-Hill Book Company, 1971.

(8) Jamal, J. Azar. Matrix Structural Analysis. New York: Pergamon Press, Inc., 1972.

APPENDIX

LISTING OF COMPUTER PROGRAMS

```
C
C
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     *                                                                   *
C     *                                                                   *
C     * PROGRAM # 1                                                       *
C     *                                                                   *
C     * SINGLE BRANCH GROUP STIFFNESSES  AND GROUP LOAD FUNCTIONS         *
C     *                                                                   *
C     *                                                                   *
C     * LANGUAGE USED        : FORTRAN IV                                 *
C     * DIGITAL MACHINE      : IBM 360 - 65                               *
C     * PROGRAMMER           : MOHAMMED A. RAWOOF                         *
C     *                                                                   *
C     *                                                                   *
C     * DESCRIPTION OF PROGRAM                                            *
C     *                                                                   *
C     *       THIS PROGRAM DEVELOPS GROUP STIFFNESSES AND                 *
C     *       GROUP LOAD FUNCTION OF SINGLE BRANCH POLYGONAL              *
C     *       SHAPE FRAME MADE UP OF STRAIGHT BARS OF CONSTANT            *
C     *       CROSS SECTION.  THE ILLUSTRATIVE EXAMPLES 1 & 2             *
C     *       (CHAPTER IV) WERE SOLVED WITH THIS PROGRAM.                 *
C     *                                                                   *
C     *                                                                   *
C     *                                                                   *
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      IMPLICIT REAL * 8 ( A-H,O-Z )
C
      DIMENSION TITLE(10)
     DIMENSION  SLLR(3,3),SLR(3,3),SRRL(3,3),SRL(3,3),SRRX(3,3),
    1SRX(3,3),SXXR(3,3),SXR(3,3),WXR(3,1),
    2FSLR(3,1),WLR(3,1),FSRL(3,1),WRL(3,1),FSRX(3,1),WRX(3,1),FSXR(3,1)
      DIMENSION  T1(3,3), T2(3,3), T3(3,3), T4(3,3), T5(3,1), T6(3,1)
C
    1     FORMAT ( 10X , I5 )
    2     FORMAT (///,50X, ' FIXED END STRESSES ',//, 40X,'LEFT END
     1                            RIGHT END ' , / )
    5     FORMAT ( 10A8 )
    8     FORMAT (/,10X,'FIXED END STRESSES',2X,3(F8.4,2X),9X,3(F8.4,2X))
    9     FORMAT (/,10X,'APPLIED JOINT LOAD',2X,3(F8.4,2X),9X,3(F8.4,2X))
   10     FORMAT ( ///,20X, 10A8 , / )
   20 FORMAT ( 4F10.5 )
   25     FORMAT ( //, 10X, 'BAR #',I5,///,17X,'LENGTH            EA      ',
     1    10X,'  EI  ', 10X,'  THETA ',//,10X,4(F12.1,5X) )
   30 FORMAT (6F8.4,6F5.2)
  110 FORMAT (1H1 )
  120 FORMAT (////,40X,17HGROUP STIFFNESSES ,////)
  130 FORMAT (35X,4HGLLR,45X,3HGLR,///,
     1    22X,'1',13X,'2',15X,'3',17X,'1',14X,'2',15X,'3',// )
  140 FORMAT (10X,I1,5X,3(F13.6,2X),5X,3(F13.6,2X) )
  150 FORMAT (///,35X,4H GRL,45X,4HGRRL,//,
     1    22X,'1',13X,'2',15X,'3',17X,'1',14X,'2',15X,'3',// )
  153     FORMAT ( 1H1,////,20X,'  GROUP FIXED END STRESSES ',///,15X,
     1     ' LEFT END',//,20X,'1',//)
  155 FORMAT (10X,I1,5X,F13.6)
  160     FORMAT (////,15X,' RIGHT END ',//,20X,'1',//)
  170 FORMAT (10X,I1,5X,F13.6)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         READ NUMBER OF PROBLEMS
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
          READ 1, NPRBLM
          DO 1000 NUM = 1 , NPRBLM
          READ 5, TITLE
      PRINT 110
          PRINT 10, TITLE
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         READ PROPERTIES OF  FIRST BAR
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      READ 20, XL, EA, EI, THETA
          EA = EA * 100.0
          MN = 1
      PRINT 25, MN, XL , EA, EI , THETA
      READ 30,(FSLR(I,1),I=1,3), (FSRL(I,1),I=1,3),(WLR(I,1),I=1,3),
     1    (WRL(I,1),I=1,3)
          PRINT  2
          PRINT 8,  ( FSLR(I,1),I=1,3 ), ( FSRL(I,1),I=1,3 )
          PRINT 9, (WLR(I,1),I=1,3) , ( WRL(I,1),I=1,3 )
      CALL STIFF (XL,EA,EI,THETA,SLLR,SLR,SRRL,SRL)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C ----    READ PROPERTIES OF NEXT BAR
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
   50 READ 20, XL, EA, EI, THETA
      IF (XL) 100,100,55
   55 READ 30,(FSRX(I,1),I=1,3), (FSXR(I,1),I=1,3),(WRX(I,1),I=1,3),
     1    (WXR(I,1),I=1,3)
          MN = MN + 1
          EA = EA * 100.0
      PRINT 25, MN, XL , EA, EI , THETA
          PRINT  2
          PRINT 8,  ( FSRX(I,1),I=1,3 ), ( FSXR(I,1),I=1,3 )
          PRINT 9, (WRX(I,1),I=1,3) , ( WXR(I,1),I=1,3 )
      CALL STIFF (XL,EA,EI,THETA,SRRX,SRX,SXXR,SXR)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C ----    CALCULATE GROUP STIFFNESSES
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
        CALL GRPSTF (SLLR,SLR,SRRL,SRL,SRRX,SRX,SXXR,SXR,
     1    FSLR,FSRL,FSRX,FSXR,WLR,WRL,WRX,WXR, T1,T2,T3,T4,T5,T6 )
          CALL DUPL ( T1, SLLR, 3 , 3 )
          CALL DUPL ( T2, SLR , 3 , 3 )
```

```
         CALL DUPL ( T3 , SRL  , 3 , 3 )
         CALL DUPL ( T4 , SRRL , 3 , 3 )
         CALL DUPL ( T5 , FSLR , 3 , 1 )
         CALL DUPL ( T6 , FSRL , 3 , 1 )
         GO TO 50
 100  PRINT 110
      PRINT 120
      PRINT 130
      DO 135 I=1,3
 135  PRINT 140, I,(SLLR(I,J),J=1,3),(SLR(I,J),J=1,3)
      PRINT 150
      DO 152 I=1,3
 152  PRINT 140, I,(SRL(I,J),J=1,3),(SRRL(I,J),J=1,3)
      PRINT 153
      DO 154 I=1,3
 154  PRINT 155, I,FSLR(I,1)
      PRINT 160
      DO 165 I=1,3
 165  PRINT 170, I,FSRL(I,1)
1000     CONTINUE
      PRINT 110
      STOP
      END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C ----    SUBROUTINE TO SET UP MEMBER STIFFNESS MATRIX AND TRANSFORMATION
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE STIFF (XL,EA,EI,THETA,SAAB,SAB,SBBA,SBA)
         IMPLICIT REAL * 8 ( A-H,O-Z )
      DIMENSION SAAB(3,3),SAB(3,3),SBBA(3,3),SBA(3,3),W(3,3),WT(3,3)
      DIMENSION X(3,3)
C ----   INITIALIZE
         CALL ZERO (SAAB,3,3)
         CALL ZERO (SAB ,3,3)
         CALL ZERO (SBBA,3,3)
         CALL ZERO (SBA ,3,3)
         CALL ZERO (W,3,3)
         CALL ZERO (WT,3,3)
         CALL ZERO (X,3,3)
      SAAB(1,1) = EA/ XL
      SAAB(2,2) = (12.0 *EI) / (XL**3)
      SAAB(2,3) = (6.0 * EI) / (XL**2)
      SAAB(3,2) = SAAB(2,3)
      SAAB(3,3) = (4.0 * EI) / XL
      SAB(1,1) = - SAAB(1,1)
      SAB(2,2) = - SAAB(2,2)
      SAB(2,3) = SAAB(2,3)
      SAB(3,2) = - SAAB(2,3)
      SAB(3,3) = (2.0 * EI) /XL
      SBBA(1,1) = SAAB(1,1)
      SBBA(2,2) = SAAB(2,2)
      SBBA(2,3) = - SAAB(2,3)
      SBBA(3,2) = SBBA(2,3)
      SBBA(3,3) = SAAB(3,3)
      THETA = THETA * 3.1415926535      / 180.0
      W(1,1) =DCOS(THETA)
      W(1,2) =DSIN(THETA)
      W(2,1) = - W(1,2)
      W(2,2) = W(1,1)
      W(3,3) = 1.0
      CALL TRAN (W,WT,3,3)
      CALL MULT (SAAB,W,X,3,3)
         CALL ZERO (SAAB,3,3)
      CALL MULT (WT,X,SAAB,3,3)
         CALL ZERO (X,3,3)
      CALL MULT (SAB,W,X,3,3)
         CALL ZERO (SAB ,3,3)
      CALL MULT (WT,X,SAB,3,3)
      CALL TRAN (SAB,SBA,3,3)
         CALL ZERO (X,3,3)
      CALL MULT (SBBA,W,X,3,3)
         CALL ZERO (SBBA,3,3)
      CALL MULT (WT,X,SBBA,3,3)
      RETURN
      END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C ----    SUBROUTINE TO FORM GROUP STIFFNESSES
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE GRPSTF(SLLR,SLR,SRRL,SRL,SRRX,SRX,SXXR,SXR,
     1   FSLR,FSRL,FSRX,FSXR,WLR,WRL,WRX,WXR, D1,D2,D3,D4,D5,D6  )
C
         IMPLICIT REAL * 8 ( A-H,O-Z )
      DIMENSION SLLR(3,3),SLR(3,3),SSRR(3,3),SRL(3,3),SRX(3,3),SXR(3,3),
     1   SXXR(3,3),FSLR(3,1),SFSR(3,1),FSXR(3,1),SWR(3,1),
     2   SRRL(3,3),X(3,3),Y(3,3),P(3,1),Q(3,1)
      DIMENSION  DSLR(3,3), SSRRI(3,3),FSRL(3,1),SRRX(3,3)
         DIMENSION  FSRX(3,1),WRL(3,1),WRX(3,1),WLR(3,1),WXR(3,1)
         DIMENSION  D1(3,3), D2(3,3), D3(3,3), D4(3,3), D5(3,1), D6(3,1)
C
      CALL ADSUB (SRRL,SRRX,SSRRI,3,3,+1)
      CALL ADSUB (FSRL,FSRX,SFSR,3,1,+1)
      CALL ADSUB (WRL,WRX,SWR,3,1,+1)
      CALL INVERT (SSRRI,3)
         CALL ZERO (X,3,3)
         CALL ZERO (Y,3,3)
         CALL ZERO (P,3,1)
         CALL ZERO (Q,3,1)
         CALL DUPL (SLR,DSLR,3,3)
      CALL MULT (SLR,SSRRI,X,3,3)
      CALL MULT (X,SRL,Y,3,3)
      CALL ADSUB (SLLR,Y, D1 ,3,3,-1)
         CALL ZERO  (X,3,3)
         CALL ZERO  (Y,3,3)
      CALL MULT (SLR,SSRRI,X,3,3)
         CALL ZERO (SLR,3,3)
```

```
      CALL MULT (X,SRX,Y,3,3)
      CALL ADSUB (Y,Y, D2,3,3,0)
         CALL ZERO  (X,3,3)
         CALL ZERO (Y,3,3)
      CALL MULT(SXR,SSRRI,X,3,3)
      CALL MULT(X,SRL,Y,3,3)
         CALL ZERO (SRL,3,3)
      CALL ADSUB (Y,Y, D3,3,3,0)
         CALL ZERO  (X,3,3)
         CALL ZERO  (Y,3,3)
      CALL MULT (SXR,SSRRI,X,3,3)
      CALL MULT (X,SRX,Y,3,3)
      CALL ADSUB (SXXR,Y, D4 ,3,3,-1)
      CALL ADSUB (SFSR,SWR,P,3,1,-1)
      CALL MULT (DSLR,SSRRI,Y,3,3)
      CALL MULT (Y,P,Q,3,1)
      CALL ADSUB (FSLR,Q, D5 ,3,1,-1)
         CALL ZERO  (Y,3,3)
         CALL ZERO (P,3,1)
         CALL ZERO (Q,3,1)
      CALL ADSUB (SFSR,SWR,P,3,1,-1)
      CALL MULT (SXR,SSRRI,Y,3,3)
      CALL MULT (Y,P,Q,3,1)
      CALL ADSUB (FSXR,Q, D6 ,3,1,-1)
         RETURN
         END
C
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         SUBROUTINE PRNT  TO PRINT MATRIX X OF  M ROWS AND  N COLUMNS
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE PRNT (X,M,N)
         IMPLICIT REAL * 8 ( A-H,O-Z )
         DIMENSION X(M,N)
      DO 10 I=1,M
   10 PRINT 20, I,(X(I,J),J=1,N)
   20 FORMAT (//,10X,I1,5X,3(D13.6,2X))
         RETURN
         END
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         SUBROUTINE MULT   TO MULTIPLY TWO MATRICES X(M X M),
C       . Y(M X N) AND STORE THE PRODUCT AS  Z(M X N)
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SUBROUTINE MULT (X,Y,Z,M,N)
         IMPLICIT REAL * 8 ( A-H,O-Z )
      DIMENSION  X(M,M),Y(M,N),Z(M,N)
      DO 100 I=1,M
      DO 80 J=1,N
      TEMP =0.0
      DO 50 K=1,M
   50 TEMP = TEMP+X(I,K) * Y(K,J)
   80 Z(I,J) = TEMP
  100 CONTINUE
```

```
         RETURN
         END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         SUBROUTINE  ADSUB TO ADD(X+Y) OR SUBTRACT (X-Y)  OR
C         TO CHANGE SIGN OF MATRIX X AND STORE RESULT AS  Z ( M X N )
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE ADSUB (X,Y,Z,M,N,ISIGN)
         IMPLICIT REAL * 8 ( A-H,O-Z )
      DIMENSION X(M,N),Y(M,N),Z(M,N)
      IF (ISIGN) 10,40,70
   10 DO 30 I=1,M
      DO 20 J =1,N
   20 Z(I,J) = X(I,J)  - Y(I,J)
   30 CONTINUE
      GO TO 100
   40 DO 60 I=1,M
      DO 50 J=1,N
   50 Z(I,J)= -X(I,J)
   60 CONTINUE
      GO TO 100
   70 DO 90 I=1,M
      DO 80 J=1,N
   80 Z(I,J)=X(I,J) + Y(I,J)
   90 CONTINUE
  100    RETURN
         END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         SUBROUTINE  TRAN  TO TRANSPOSE X(M X N) AS  Y(N X M)
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE TRAN (X,Y,M,N)
         IMPLICIT REAL * 8 ( A-H,O-Z )
      DIMENSION  X(M,N),Y(N,M)
      DO 20 I=1,M
      DO 10 J=1,N
   10 Y(J,I) = X(I,J)
   20 CONTINUE
         RETURN
         END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         SUBROUTINE  ZERO  TO MAKE  ALL ELEMENTS OF
C         MATRIX  X(M X N) ZERO
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
         SUBROUTINE ZERO (X,M,N)
         IMPLICIT REAL * 8 ( A-H,O-Z )
         DIMENSION  X(M,N)
         DO 10 I=1,M
         DO 10 J=1,N
         X(I,J) =0.0
   10    CONTINUE
```

```
          RETURN
          END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         SUBROUTINE DUPL  TO DUPLICATE MATRIX Y AS X(M X N)
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
          SUBROUTINE DUPL (X,Y,M,N)
          IMPLICIT REAL * 8 ( A-H,O-Z )
          DIMENSION X(M,N),Y(M,N)
          DO 10 I=1,M
          DO 10 J=1,N
          Y(I,J)=X(I,J)
   10     CONTINUE
          RETURN
          END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         SUBROUTINE INVERT TO  REPLACE X(M X M) AS ITS INVERT
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE INVERT (X,M)
          IMPLICIT REAL * 8 ( A-H , O-Z )
      DIMENSION  X(M,M)
      DO 60 I=1,M
      S=1.0/X(I,I)
      DO 10 J=1,M
   10 X(I,J)=X(I,J) * S
      X(I,I) = S
      DO 60 J=1,M
      IF (J .EQ. I) GO TO 60
      S=X(J,I)
      X(J,I)=0.0
      DO 50 K=1,M
   50 X(J,K)=X(J,K)-S*X(I,K)
   60 CONTINUE
          RETURN
          END
$ENTRY
```

```
C
C
C
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     *                                                                     *
C     *                                                                     *
C     *   PROGRAM  # 2                                                      *
C     *                                                                     *
C     *   MULTIBRANCH GROUP STIFFNESSES AND                                 *
C     *   GROUP LOAD FUNCTIONS OF A SPACE TRUSS                             *
C     *                                                                     *
C     *   LANGUAGE USED          :    FORTRAN IV                            *
C     *                                                                     *
C     *   DIGITAL MACHINE        :    IBM 360 - 65                          *
C     *                                                                     *
C     *   PROGRAMMER             :    MOHAMMED  A.  RAWOOF                  *
C     *                                                                     *
C     *   DESCRIPTION OF PROGRAM :                                          *
C     *                                                                     *
C     *   THIS PROGRAM DEVELOPS GROUP STIFFNESSES AND                       *
C     *   GROUP LOAD FUNCTIONS OF EACH SUBSTRUCTURE AND                     *
C     *   CONNECTS ALL SUBSTRUCTURES OF THE SPACE TRUSS.                    *
C     *   CALCULATES JOINT DEFORMATIONS OF CONNECTION NODES AND             *
C     *   ALL OTHER NODES AND COMPUTES ALL MEMBER END ACTIONS               *
C     *   AND SUPPORT REACTIONS.                                            *
C     *   THE ILLUSTRATIVE EXAMPLE # 3 (CHAPTERIV ) HAS BEEN                *
C     *   SOLVED WITH THIS PROGRAM.                                         *
C     *                                                                     *
C     *   INPUT PARAMETERS:                                                 *
C     *                                                                     *
C     *   PROBLEM TITLE                                                     *
C     *   NJT , NMEM , NSPRTS , NIJE , NSUBRS     ( CONTROL DATA )          *
C     *   JN , R , THETA , Z(JN)          ( JOINT COORDINATE DATA )         *
C     *   MN , JOINTJ(MN) , JOINTK(MN) , AE(MN)    ( MEMBER DATA )          *
C     *   LN        ( LOADING NUMBER )                                      *
C     *   LOAD TITLE                                                        *
C     *   JN , FX , FY , FZ               ( JOINT STRESSES )                *
C     *   JN , WX , WY , WZ               ( JOINT LOADS )                   *
C     *                                                                     *
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C
      IMPLICIT  REAL  * 8  (A-H , O-Z )
C
      DIMENSION  TITLE (10)
      DIMENSION  S(63,63),X(21),Y(21),Z(21),JOINTJ(45),JOINTK(45),
     1 AE(45),FS(63,1),W(63,1),SM(6,6),SMJ(75,75),GS(27,27),
     2 GTEMP(12,12),TEMP1(15,18),TEMP2(12,18),TEMP3(3,3),TEMP4(3,1),
     3 TEMP5(12,1),TEMP6(24,1),
     4 FS1(15,1),FS2(18,1),FS3(12,1),W2(18,1),GFS(27,1),YY(18,1),
     5 C(15,1),E(12,1),VJST(75,1),VL(75,1)
      DIMENSION GTEMP1(24,24), SG(27,27)
      DIMENSION  RGFS(27,1) , RR(27,27)
      DIMENSION ST21(18,15),ST23(18,12),SD2(18,18),ALPHA(18,1),
     1 BETA(18,1),GAMA(18,1),DELTAL(15,1),DELTAR(12,1),DELTAM(18,1)
      DIMENSION  U(63,1), REACTX(21), REACTY(21), REACTZ(21)
      DIMENSION   SAVE1(3,24), SAVE1T(24,3),SAVE2(3,12),SAVE2T(12,3)
C
      DATA      ONE , PI , SIXTY , H80
     1 /          1.0D00 ,  3.14159265358979D00,60.0D00,180.0D00  /
C
    1 FORMAT ( 10A8 )
    5 FORMAT ( 1H1 )
   10 FORMAT (//,10X,10A8 )
   20 FORMAT (10X, 5I5 )
   25 FORMAT ( //,10X, 'JOINT COORDINATE DATA , POLAR COORDINATES',
     1 //,10X,'  JT.  #                RADIUS                 THETA
     2 Z-COORD.',/)
   30 FORMAT (///,10X,21HDATA FOR SUBSTRUCTURE,//,10X,12HCONTROL DATA
     1 ,//,58H               NO.JTS.    NO.MEMS.    NO.SUPRTS.    NO.IJE
     2 NO.SUBSTRS,//,11X,I5,6X,I5,7X,I5,9X,I5,6X,I5,// )
   40 FORMAT (10X,I5,5X, 3F10.6 )
   50 FORMAT (//,10X,21HJOINT COORDINATE DATA//,70H            JT.NO.
     1      X-COORD.           Y-COORD.             Z-COORD.   , / )
   60 FORMAT (/,11X,I5,5X,3(1PD15.6, 5X))
   70 FORMAT (10X,3I5,5X,F20.6 )
   80 FORMAT (//,10X,39HMEMBER INCIDENCES AND MEMBER PROPERTIES//,
     1 49H           MEM.NO.      FROM   TO JT.            AE   ,/ )
   90 FORMAT (/,10X,I5,5X,I5,3X,I5,6X,1PD15.6 )
  145 FORMAT (///,10X,19HAPPLIED JOINT LOADS,/, '           JT.NO. ',
     1 7X, 'X-LOAD                 Y-LOAD                Z-LOAD '/ )
  125 FORMAT (///,10X,14HSUBSTRUCTURE #,I3 )
  205 FORMAT (10X, I5 )
  210 FORMAT (//,10X,10H LOADING #,I5,//)
  510 FORMAT ( 1H1,///,10X,6HJT. #  ,2X, 13H   X-DISP.    ,5X,
     1 'Y-DISP.          Z-DISP. ',// )
  520 FORMAT ( /, 11X,I3, 4X, 3( F10.6 , 5X ) )
  560 FORMAT ( ////,10X,'MEM.NO.',3X,'FROM       TO JOINT        AXIAL'
     1 'FORCE' , // )
  610 FORMAT ( 12X,3(I2,7X), F14.7 )
  710 FORMAT (1H1,////,10X,'SUPPORT NO.',5X,'  X-FORCE   ',3X,
     1 '  Y-FORCE   ',3X,'  Z-FORCE   ',// )
  720 FORMAT ( 13X,I2,11X,3(F12.6,3X )  )
 9000 FORMAT ( ////,10X,'***** ERROR *****',/)
C
C         READ & ECHO TITLE
      READ 1, TITLE
      PRINT 5
```

```
      PRINT 10, TITLE
C
C     READ & ECHO CONTROL DATA
      READ 20, NJT,NMEM,NSPRTS,NIJE,NSUBRS
      PRINT30, NJT,NMEM,NSPRTS,NIJE,NSUBRS
C
      ND = 3 * NJT
      NN = NJT - (NSPRTS + NIJE )
      N = (NN+1)/ 2
      L = 3 * (NN - N)
      N = 3 * N
      M = 3 * NIJE
      K =  N + L
      JMS = 3 * (NSUBRS * (L/3) + 1 )
      NP1 = N + 1
      NPMP1 = N+M+1
C
C     READ & ECHO JOINT COORDINATE DATA
      PIOH80 = PI / H80
      PRINT 25
      DO 45 I = 1 , NJT
      READ 40, JN, R , THETA ,Z(JN)
      PRINT 60, JN, R , THETA , Z(JN)
      THETAR = THETA * PIOH80
      X(JN) = R * DCOS(THETAR )
      Y(JN) = R * DSIN(THETAR )
   45 CONTINUE
      PRINT 5
      PRINT 50
      DO 55 JN = 1 , NJT
   55 PRINT 60, JN, X(JN), Y(JN), Z(JN)
C     READ & ECHO MEMBER DATA AND MEMBER PROPERTIES
      DO 65 I = 1 , NMEM
   65 READ 70, MN, JOINTJ(MN), JOINTK(MN), AE(MN)
      PRINT 5
      PRINT 80
      DO 75 MN= 1 , NMEM
   75 PRINT 90,MN, JOINTJ(MN), JOINTK(MN), AE(MN)
C
C     SET UP SUBSTRUCTURE STIFFNESS MATRIX
C
C     INITIALIZE
      CALL ZERO (S,ND,ND)
C     MEMBER  STIFFNESS MATRIX
      DO 100 MN = 1 , NMEM
      JMN = JOINTJ(MN)
      KMN = JOINTK(MN)
      DX = X(KMN) - X(JMN)
      DY = Y(KMN) - Y(JMN)
      DZ = Z(KMN) - Z(JMN)
      XL = DSQRT (DX*DX + DY*DY + DZ*DZ )
      CX = DX / XL
      CY = DY / XL
      CZ = DZ / XL
      AEOL = AE(MN) / XL
      CALL MEMSTF   (CX,CY,CZ,AEOL,SM )
C
C     ADD MEMBER STIFFNESS MATRIX TO SUBSTRUCTURE STIFF. MATRIX
C
      JSHIFT = 3 * (JMN-1)
      KSHIFT = 3 * (KMN-1)
      DO 95 JJ = 1 , 3
      DO 95 KK = 1 , 3
      J1 = JSHIFT + JJ
      J2 = JSHIFT + KK
      K1 = KSHIFT + JJ
      K2 = KSHIFT + KK
      S(J1,J2) = S(J1,J2) + SM(JJ,KK)
      S(K1,K2) = S(K1,K2) + SM(JJ+3,KK+3)
      S(K1,J2) = S(K1,J2) + SM(JJ+3,KK)
      S(J1,K2) = S(J1,K2) + SM(JJ,KK+3)
   95 CONTINUE
  100 CONTINUE
C
C     ELIMINATE INTERIOR JOINTS OF THE SUBSTRUCTURE
C     SUBSTRUCTURE STIFF. MATRIX  ( GROUP STIFFNESSES )
C
      CALL GRPSTF  ( S, ND, L, M, N, GS, TEMP1, TEMP2,ST21,ST23,SD2)
C
C
C     CONNECT ALL SUBSTRUCTURES
C     SET UP JOINT STIFFNESS MATRIX
C
      KM3 = K - 3
      CALL ZERO (SMJ,JMS,JMS)
      CALL ZERO (GTEMP,L,L)
      CALL ZERO (GTEMP1,KM3,KM3)
      CALL  ADSM (SMJ,JMS,JMS,GS,K,K,1,1 )
C
      NS = NSUBRS - 1
      JSHIFT = 4
C
C
C
C
      DO 180 I = 1 , NS
      THETA = I * SIXTY
      CALL ROTATE ( GS , K, THETA, SG, RR )
      CALL RMVSM ( SG , K , K , TEMP3 , 3 , 3 , 1 , 1 )
      CALL ADSM ( SMJ,JMS,JMS, TEMP3 , 3,3, 1 , 1 )
      JSHIFT = JSHIFT + 12
      IF ( JSHIFT .EQ. 64 ) GO TO 150
      CALL RMVSM ( SG , K,K, GTEMP1,KM3,KM3, 4 , 4)
      CALL ADSM ( SMJ,JMS,JMS,GTEMP1 ,KM3,KM3, JSHIFT , JSHIFT )
      CALL RMVSM ( SG,K,K, SAVE1,3,24, 1 , 4 )
      CALL ADSM ( SMJ,JMS,JMS, SAVE1,3,24, 1 , JSHIFT )
      CALL TRAN ( SAVE1, SAVE1T, 3, 24 )
      CALL ADSM ( SMJ,JMS,JMS, SAVE1T,24,3, JSHIFT, 1 )
```

```
      GO TO  180
C
C
  150 CONTINUE
      CALL RMVSM ( SG ,K , K , GTEMP , L , L , 4 , 4 )
      CALL ADSM ( SMJ ,JMS,JMS, GTEMP , L , L , 64 , 64 )
      CALL RMVSM ( SG , K,K, GTEMP , L,L, 16 , 16 )
      CALL ADSM ( SMJ ,JMS,JMS, GTEMP , L,L, 4 , 4 )
      CALL RMVSM ( SG,K,K, GTEMP,L,L, 4 , 16 )
      CALL ADSM ( SMJ,JMS,JMS, GTEMP,L,L, 64 , 4 )
      CALL RMVSM ( SG,K,K, GTEMP,L,L, 16 , 4 )
      CALL ADSM ( SMJ,JMS,JMS, GTEMP,L,L, 4 , 64 )
      CALL RMVSM ( SG,K,K, SAVE2,3,12, 1 , 4 )
      CALL ADSM ( SMJ,JMS,JMS, SAVE2,3,12, 1 , 64 )
      CALL TRAN ( SAVE2, SAVE2T, 3 , 12 )
      CALL ADSM ( SMJ,JMS,JMS, SAVE2T,12,3, 64 , 1 )
      CALL RMVSM ( SG,K,K, SAVE2,3,12, 1 , 16 )
      CALL ADSM ( SMJ,JMS,JMS, SAVE2,3,12, 1 , 4 )
      CALL TRAN ( SAVE2, SAVE2T , 3 , 12 )
      CALL ADSM ( SMJ,JMS,JMS, SAVE2T,12,3, 4 , 1 )
  180 CONTINUE
C
C     SYSTEM STIFFNESS MATRIX " SMJ "
C
      THETA = SIXTY
      CALL ROTATE ( SG, K , THETA, SG ,RR )
C
C     JOINT STRESSES & APPLIED JOINT LOADS
C
C     LOADING NUMBER
  200 READ 205, LN
      IF ( LN .EQ. 0 ) GO TO 9999
C     READ & ECHO LOAD TITLE
      READ 1, TITLE
      PRINT 5
      PRINT 210, LN
      PRINT 10, TITLE
C     READ JOINT STRESSES
      CALL ZERO (VJST,JMS,1)
      CALL ZERO (VL,JMS,1)
      DO 500 I = 1 , NSUBRS
C     INITIALIZE
      CALL ZERO (FS,ND,1)
      DO 215 NJ = 1 , NJT
      READ 40, JN , FX , FY , FZ
      JROW = 3 * (JN - 1 )
      FS(JROW+1,1) = FX
      FS(JROW+2,1) = FY
      FS(JROW+3,1) = FZ
  215 CONTINUE
C     ECHO JOINT STRESSES
C     INITIALIZE
C
C     READ APPLIED JOINT LOADS
C
      CALL ZERO (W,ND,1)
      DO 240 NJ = 1 , NJT
      READ 40, JN , WX , WY , WZ
      WZ = -WZ
      JROW = 3 * (JN-1)
      W(JROW+1,1) = WX
      W(JROW+2,1) = WY
      W(JROW+3,1) = WZ
  240 CONTINUE
C     ECHO APPLIED JOINT LOADS
      PRINT 125, I
      PRINT 145
      JN = 0
      DO 250 II = 1 , ND , 3
      JN = JN+1
      PRINT 60, JN,W(II,1),W(II+1,1),W(II+2,1)
  250 CONTINUE
      IF ( I .NE. 1 ) GO TO 300
C     INITIALIZE
      CALL ZERO (YY,M,1)
      CALL ZERO (C,N,1)
      CALL ZERO (E,L,1)
      CALL RMVSM (FS,ND,1,FS1,N,1,1,1)
      CALL RMVSM (FS,ND,1,FS2,M,1,NP1,1)
      CALL RMVSM (FS,ND,1,FS3,L,1,NPMP1,1)
      CALL RMVSM (W,ND,1,W2,M,1,NP1,1)
      CALL ADSUB (FS2,W2,YY,M,1,-1)
      CALL MULT (TEMP1,YY,C,N,M,1)
      CALL ADSUB (FS1,C,C,N,1,-1)
      CALL MULT (TEMP2,YY,E,L,M,1)
      CALL ADSUB (FS3,E,E,L,1,-1)
      CALL ZERO (GFS,K,1)
C
C     SET UP GROUP FIXED END STRESSES MATRIX " GFS "
C
      CALL ADSM (GFS,K,1,C,N,1,1,1)
      CALL ADSM (GFS,K,1,E,L,1,NP1,1)
C     SET UP JOINT STRESS VECTOR " VJST "
      CALL ADSM (VJST,JMS,1,GFS,K,1,1,1)
      CALL DUPL ( GFS , RGFS , K , 1 )
      GO TO 360
  300 CONTINUE
      PRINT 5
      IMONE = I - 1
      THETA = IMONE * SIXTY
      CALL ROTATE ( SG , K, THETA, SG, RR )
      CALL MULT ( RR , RGFS, GFS, K,K, 1)
      CALL ZERO (TEMP4,3,1)
      CALL ZERO (TEMP5,L,1)
      CALL ZERO (TEMP6,KM3,1)
      CALL RMVSM (GFS,K,1,TEMP4,3,1,1,1)
      CALL ADSM (VJST,JMS,1,TEMP4,3,1,1,1)
      CALL RMVSM (GFS,K,1,TEMP6,KM3,1,4,1)
```

```
      IF ( I .EQ. 2 ) GO TO 310
      IF ( I .EQ. 3 ) GO TO 320
      IF ( I .EQ. 4 ) GO TO 330
      IF ( I .EQ. 5 ) GO TO 340
      IF ( I .EQ. 6 ) GO TO 350
      IF ( I .GT. 6 ) PRINT 9000
      GO TO 9999
C
  310 CALL ADSM (VJST,JMS,1,TEMP6,KM3,1,16,1)
      GO TO 360
  320 CALL ADSM (VJST,JMS,1,TEMP6,KM3,1,28,1)
      GO TO 360
  330 CALL ADSM (VJST,JMS,1,TEMP6,KM3,1,40,1)
      GO TO 360
  340 CALL ADSM (VJST,JMS,1,TEMP6,KM3,1,52,1)
      GO TO 360
  350 CALL RMVSM (GFS,K,1,TEMP5,L,1,4,1)
      CALL ADSM (VJST,JMS,1,TEMP5,L,1,64,1)
      CALL RMVSM ( GFS,K,1, TEMP5, L, 1 , 16, 1 )
      CALL ADSM (VJST,JMS,1,TEMP5,L,1,4,1 )
      GO TO  380
  360 CONTINUE
  380 CONTINUE
C
C     SET UP APPLIED JOINT LOAD VECTOR
      CALL ZERO (TEMP6,KM3,1)
      CALL RMVSM (W,ND,1,TEMP4,3,1,1,1)
      CALL ADSM (VL,JMS,1,TEMP4,3,1,1,1)
      CALL RMVSM (W,ND,1,TEMP5,12,1,4,1)
      CALL ADSM (TEMP6,KM3,1,TEMP5,L,1,1,1)
      CALL ZERO (TEMP5,L,1)
      CALL RMVSM (W,ND,1,TEMP5,L,1,34,1)
      CALL ADSM (TEMP6,KM3,1,TEMP5,L,1,13,1)
      IF ( I .NE. 1 ) GO TO 400
      CALL ADSM (VL,JMS,1,TEMP6,KM3,1,4,1)
      GO TO 500
  400 IF ( I .EQ. 2 ) GO TO 410
      IF ( I .EQ. 3 ) GO TO 420
      IF ( I .EQ. 4 ) GO TO 430
      IF ( I .EQ. 5 ) GO TO 440
      IF ( I .EQ. 6 ) GO TO 405
      IF ( I .GT. 6 )  PRINT 9000
      GO TO 9999
  405 CALL ADSM (VL,JMS,1,TEMP5,L,1,4,1)
      CALL ZERO (TEMP5,L,1)
      CALL RMVSM (TEMP6,KM3,1,TEMP5,L,1,1,1)
      CALL ADSM (VL,JMS,1,TEMP5,L,1,64,1)
      GO TO 500
  410 CALL ADSM (VL,JMS,1,TEMP6,KM3,1,16,1)
      GO TO 500
  420 CALL ADSM (VL,JMS,1,TEMP6,KM3,1,28,1)
      GO TO 500
  430 CALL ADSM (VL,JMS,1,TEMP6,KM3,1,40,1)
      GO TO 500
  440 CALL ADSM (VL,JMS,1,TEMP6,KM3,1,52,1)
  500 CONTINUE
C     SOLVE FOR DEFORMATIONS ( CONNECTING NODES )
C     SMJ * DEL + VJST = VL
C
C
C
C
C
C
C
      CALL ADSUB (VL,VJST,  VL ,JMS,1,-1)
      CALL ZERO (VJST,JMS,1)
      CALL DECOMP ( SMJ, JMS )
      CALL SOLVE ( SMJ, VL , VJST, JMS )
C
C
      PRINT 510
      JN = 0
      DO 515  I = 1 , JMS , 3
      JN = JN + 1
      PRINT 520 , JN, VJST(I,1), VJST(I+1,1), VJST(I+2,1)
  515 CONTINUE
      CALL RMVSM  (VJST,75,1,DELTAL,15,1,1,1 )
      CALL RMVSM  (VJST,75,1,DELTAR,12,1,16,1 )
      CALL MULT ( ST21,DELTAL,ALPHA,18,15,1 )
      CALL MULT ( ST23,DELTAR,BETA,18,12,1 )
      CALL MULT ( SD2, YY, GAMA, 18,18,1 )
      CALL ADSUB ( ALPHA, BETA, BETA, 18,1, +1 )
      CALL ADSUB ( BETA, GAMA, DELTAM, 18, 1, +1 )
      CALL ADSUB ( DELTAM,DELTAM,DELTAM, 18, 1, 0 )
      PRINT 5
      PRINT 510
      JN = 0
      DO  550 I=1,18,3
      JN = JN+1
      PRINT 520, JN, DELTAM(I,1), DELTAM(I+1,1), DELTAM(I+2,1)
  550 CONTINUE
C
      CALL ZERO ( U, 63, 1 )
      CALL ADSM ( U,63,1, DELTAL,15,1,1,1 )
      CALL ADSM ( U,63,1, DELTAM,18,1, 16,1 )
      CALL ADSM ( U,63,1, DELTAR,12,1, 34,1 )
      PRINT 5
      PRINT 560
C
C     SOLVE FOR MEMBER END FORCES & REACTIONS
      DO 555  I = 1,21
      REACTX(I) = 0.0
      REACTY(I) = 0.0
      REACTZ(I) = 0.0
  555 CONTINUE
      DO 700  MN = 1 , NMEM
      JMN = JOINTJ(MN)
      KMN = JOINTK(MN)
```

```
      JJ = 3 * (JMN - 1 )
      JK = 3 * (KMN - 1 )
      DX = X(KMN) - X(JMN)
      DY = Y(KMN) - Y(JMN)
      DZ = Z(KMN) - Z(JMN)
      XL = DSQRT ( DX*DX + DY*DY + DZ*DZ )
      CX = DX/XL
      CY = DY/XL
      CZ = DZ/XL
      AEOL = AE(MN) / XL
C     CALCULATE MEMBER AXIAL FORCE
      FM  =  AEOL*(-CX*U(JJ+1,1)-CY*U(JJ+2,1)-CZ*U(JJ+3,1)
     1            +CX*U(JK+1,1)+CY*U(JK+2,1)+CZ*U(JK+3,1)   )
C
      IF ( MN .LT. 6 ) FM = 2*FM
      IF ( MN .EQ.11 ) FM = 2*FM
      IF ( MN .EQ.24 ) FM = 2*FM
      IF ( MN .EQ.34 ) FM = 2*FM
      IF ( MN .EQ.41 ) FM = 2*FM
      IF ( MN .EQ.45 ) FM = 2*FM
      PRINT 610,  MN, JOINTJ(MN), JOINTK(MN), FM
C     CALCULATE REACTIONS
      IF ( JMN .LT. 16 ) GO TO 700
      I = JMN
      REACTX(I) = REACTX(I) - FM * CX
      REACTY(I) = REACTY(I) - FM * CY
      REACTZ(I) = REACTZ(I) - FM * CZ
  700 CONTINUE
      PRINT 710
      DO 725 I = 16, 21
      PRINT 720,   I, REACTX(I),REACTY(I),REACTZ(I)
  725 CONTINUE
      PRINT 5
C
C
      GO TO 200
 9999 STOP
      END
C
C
C     SUBROUTINE  MULTIPLY TWO MATRICES
      SUBROUTINE MULT (X,Y,Z,M,N,K)
      IMPLICIT REAL * 8 ( A-H , O-Z )
      DIMENSION  X(M,N), Y(N,K), Z(M,K)
      DO 100 I = 1 , M
      DO 800 J = 1 , K
      TEMP = 0.0
      DO 500 L = 1 , N
  500 TEMP = TEMP + X(I,L) * Y(L,J)
  800 Z(I,J) = TEMP
  100 CONTINUE
      RETURN
      END
C
C     SUBROUTINE ADD SUBMATRIX INTO A LARGE MATRIX
      SUBROUTINE ADSM ( X,M,N,Y,I,J,K,L )
      IMPLICIT REAL * 8 ( A-H , O-Z )
      DIMENSION X(M,N) , Y (I,J)
      KK = K
      DO 200 II = 1 , I
      LL = L
      DO 100 JJ = 1 , J
      X(KK,LL) = X(KK,LL) + Y(II,JJ)
      LL = LL + 1
  100 CONTINUE
      KK = KK + 1
  200 CONTINUE
      RETURN
      END
C
C     SUBROUTINE: REMOVE SUBMATRIX FROM A LARGE MATRIX
      SUBROUTINE  RMVSM ( X,M,N,Y,I,J,K,L )
      IMPLICIT REAL * 8 ( A-H , O-Z )
      DIMENSION  X(M,N) , Y(I,J)
      KK = K
      DO 100 II = 1 , I
      LL = L
      DO 200 JJ = 1 , J
      Y(II,JJ) = X(KK,LL)
      LL = LL + 1
  200 CONTINUE
      KK = KK + 1
  100 CONTINUE
      RETURN
      END
C
C     SUBROUTINE PRNT ( X,M,N )
      SUBROUTINE PRNT ( X,M,N )
      IMPLICIT REAL * 8 ( A-H , O-Z )
      DIMENSION  X(M,N)
      K = 1
      KK = 8
      IF ( KK .GT. N )  KK = N
      PRINT 1
  100 PRINT 50,  ( L , L=K , KK )
      DO 10 I = 1 , M
      PRINT 20, I,( X(I,J),J = K , KK )
   10 CONTINUE
      IF ( KK .EQ. N ) GO TO 200
      K = KK+1
      KK = KK+8
      IF ( KK .GT. N )  KK = N
      GO TO 100
    1 FORMAT ( 1H1 )
   20 FORMAT  (/,5X,I3,2X,5(1PD13.6,2X) )
   50 FORMAT (//,12X,8(I3,12X),/)
  200 CONTINUE
      RETURN
```

```
      END
C
C
C     SUBROUTINE: SET UP MEMBER STIFFNESS MATRIX
      SUBROUTINE  MEMSTF   (CX,CY,CZ,AEOL,S)
      IMPLICIT REAL * 8 ( A-H , O-Z )
      DIMENSION  S(6,6)
      CALL ZERO (S,6,6 )
C     SET UP SPACE TRUSS MEMBER STIFFNESS MATRIX AND ROTATION MATRIX
      Q = DSQRT (CX*CX + CZ*CZ )
      IF ( Q .LT. 1.0D-04 ) GO TO 200
      S(1,1) = AEOL * CX * CX
      S(1,2) = AEOL * CX* CY
      S(1,3) = AEOL * CX * CZ
      S(2,1) = S(1,2)
      S(2,2) = AEOL * CY * CY
      S(2,3) = AEOL * CY * CZ
      S(3,1) = S(1,3)
      S(3,2) = S(2,3)
      S(3,3) = AEOL * CZ * CZ
      DO 100 I = 1 , 3
      DO 100 J = 1 , 3
      ST = S(I,J)
      S(I,J+3) = - ST
      S(I+3,J) = - ST
      S(I+3,J+3) = ST
  100 CONTINUE
      GO TO 300
  200 ST = AEOL * CY *CY
      S(2,2) = ST
      S(5,2) = -ST
      S(2,5) = -ST
      S(5,5) = ST
  300 CONTINUE
      RETURN
      END
C
C     SUBROUTINE: TO SET UP GROUP STIFFNESS MATRIX
      SUBROUTINE GRPSTF  ( S,ND,L,M,N,GS,TEMP1,TEMP2,S2221,S2223,S22 )
      IMPLICIT REAL * 8 ( A-H , O-Z )
C
C
      DIMENSION S(63,63),S11(15,15),S12(15,18),S13(15,12),S21(18,15),
     1 S22(18,18),S23(18,12),S31(12,15),S32(12,18),S33(12,12),
     2 GS(27,27),TEMP1(15,18),TEMP2(12,18),A(15,15),B(15,12),D(12,12),
     3 BT(12,15),X(15,18),Z(12,18)
      DIMENSION  S2221(18,15), S2223(18,12)
C
C
      K = N + L
C     INITIALIZE
      CALL ZERO (A,N,N)
      CALL ZERO (B,N,L)
      CALL ZERO (D,L,L)
      CALL ZERO (BT,L,N)
      CALL ZERO (TEMP1,N,M)
      CALL ZERO (TEMP2,L,M)
      CALL ZERO (GS,K,K)
      CALL ZERO (X,N,M)
      CALL ZERO (Z,L,M)
C
      NP1 = N + 1
      NPMP1 = N+M+1
      CALL RMVSM (S,ND,ND,S11,N,N,1,1 )
      CALL RMVSM (S,ND,ND,S12,N,M,1,NP1 )
      CALL RMVSM (S,ND,ND,S13,N,L,1,NPMP1)
      CALL RMVSM (S,ND,ND,S21,M,N,NP1,1)
      CALL RMVSM (S,ND,ND,S22,M,M,NP1,NP1 )
      CALL RMVSM (S,ND,ND,S23,M,L,NP1,NPMP1)
      CALL RMVSM (S,ND,ND,S31,L,N,NPMP1,1)
      CALL RMVSM (S,ND,ND,S32,L,M,NPMP1,NP1 )
      CALL RMVSM (S,ND,ND,S33,L,L,NPMP1,NPMP1)
      CALL INVERT ( S22 , M )
      CALL MULT (S12,S22,X,N,M,M)
      CALL DUPL (X,TEMP1,N,M)
      CALL MULT (X,S21,A,N,M,N)
      CALL ADSUB (S11,A,A,N,N,-1 )
      CALL MULT (X,S23,B,N,M,L)
      CALL ADSUB (S13,B,B,N,L,-1)
      CALL MULT (S32,S22,Z,L,M,M)
      CALL DUPL (Z,TEMP2,L,M)
      CALL MULT (Z,S23,D,L,M,L)
      CALL ADSUB (S33,D,D,L,L,-1)
      CALL TRAN ( B , BT ,N,L )
      CALL ADSM (GS,K,K,A,N,N,1,1)
      CALL ADSM (GS,K,K,B,N,L,1,N+1)
      CALL ADSM (GS,K,K,BT,L,N,N+1,1)
      CALL ADSM (GS,K,K,D,L,L,N+1,N+1 )
      CALL MULT (S22,S21,S2221,18,18,15 )
      CALL MULT (S22,S23,S2223,18,18,12 )
      RETURN
      END
C
      SUBROUTINE ADSUB (X,Y,Z,M,N,ISIGN)
         IMPLICIT REAL * 8 ( A-H,O-Z )
      DIMENSION X(M,N),Y(M,N),Z(M,N)
      IF (ISIGN) 10,40,70
   10 DO 30 I=1,M
      DO 20 J =1,N
   20 Z(I,J) = X(I,J) - Y(I,J)
   30 CONTINUE
      GO TO 100
   40 DO 60 I=1,M
      DO 50 J=1,N
   50 Z(I,J)= -X(I,J)
   60 CONTINUE
      GO TO 100
   70 DO 90 I=1,M
```

```
      DO 80 J=1,N
   80 Z(I,J)=X(I,J) + Y(I,J)
   90 CONTINUE
  100    RETURN
         END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C        SUBROUTINE  TRAN   TO TRANSPOSE X(M X N) AS  Y(N X M)
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE TRAN (X,Y,M,N)
         IMPLICIT REAL * 8 ( A-H,O-Z )
      DIMENSION  X(M,N),Y(N,M)
      DO 20 I=1,M
      DO 10 J=1,N
   10 Y(J,I) = X(I,J)
   20 CONTINUE
         RETURN
         END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C        SUBROUTINE  ZERO  TO MAKE  ALL ELEMENTS OF
C        MATRIX  X(M X N) ZERO
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
         SUBROUTINE ZERO (X,M,N)
         IMPLICIT REAL * 8 ( A-H,O-Z )
         DIMENSION  X(M,N)
         DO 10 I=1,M
         DO 10 J=1,N
         X(I,J) =0.0
   10    CONTINUE
         RETURN
         END
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C        SUBROUTINE DUPL  TO DUPLICATE MATRIX Y AS X(M X N)
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
         SUBROUTINE DUPL (X,Y,M,N)
         IMPLICIT REAL * 8 ( A-H,O-Z )
         DIMENSION X(M,N),Y(M,N)
         DO 10 I=1,M
         DO 10 J=1,N
         Y(I,J)=X(I,J)
   10    CONTINUE
         RETURN
         END
C.
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C        SUBROUTINE INVERT TO  REPLACE X(M X M) AS ITS  INVERT
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      SUBROUTINE INVERT (X,M)
         IMPLICIT REAL * 8 ( A-H , O-Z )
      DIMENSION  X(M,M)
      DO 60 I=1,M
      S=1.0/X(I,I)
      DO 10 J=1,M
   10 X(I,J)=X(I,J) * S
      X(I,I) = S
      DO 60 J=1,M
      IF (J .EQ. I) GO TO 60
      S=X(J,I)
      X(J,I)=0.0
      DO 50 K=1,M
   50 X(J,K)=X(J,K)-S*X(I,K)
   60 CONTINUE
         RETURN
         END
$ENTRY
```

VITA

Mohammed Abdul Rawoof

Candidate for the Degree of

Master of Science

Thesis: GROUP STIFFNESSES AND GROUP FIXED END STRESSES IN STRUCTURAL ANALYSIS

Major Field: Civil Engineering

Biographical:

Personal Data: Born September 25, 1946, in Nizamabad, India, the son of Mr. and Mrs. Mohammed Abdul Jabbar.

Education: Graduated from Government High School, Chowk, Hyderabad, India, in 1962; received the Bachelor of Engineering degree from Osmania University in 1970; completed requirements for the Master of Science degree at Oklahoma State University in December, 1973.

Professional Experience: Worked as a Junior Engineer, PWD(R&B), Government of A.P., India, from March 1971 to August 1971; served as a graduate teaching assistant from January 1972 to May 1973, and as a graduate research assistant from August 1973 to present in the School of Civil Engineering, Oklahoma State University.