UC Berkeley
SEMM Reports Series

Title
Computer Program for Non-Prismatic Folded Plates with Plate and Beam Elements

Permalink
https://escholarship.org/uc/item/95g748gt

Authors
Meyer, Christian
Scordelis, Alex

Publication Date
1971-12-01

eScholarship.org Powered by the California Digital Library
University of California

REPORT NO.
UC SESM 71-23

STRUCTURES AND MATERIALS RESEARCH
DEPARTMENT OF CIVIL ENGINEERING

COMPUTER PROGRAM FOR NON-PRISMATIC FOLDED PLATES WITH PLATE AND BEAM ELEMENTS

by

C. MEYER

and

A. C. SCORDELIS

Report to the Sponsors: Division of Highways, Department of Public Works, State of California, and the Bureau of Public Roads, Federal Highway Administration, United States Department of Transportation.

DECEMBER 1971

COLLEGE OF ENGINEERING
OFFICE OF RESEARCH SERVICES
UNIVERSITY OF CALIFORNIA
BERKELEY CALIFORNIA

Structures and Materials Research
Department of Civil Engineering
Division of Structural Engineering
and
Structural Mechanics

UC-SESM Report No. 71-23

COMPUTER PROGRAM FOR NON-PRISMATIC FOLDED PLATE AND BEAM ELEMENTS

by

C. Meyer
Formerly Graduate Student in Civil Engineering

and

A. C. Scordelis
Professor of Civil Engineering

to

the Division of Highways
Department of Public Works
State of California
Under Research Technical Agreement
No. 13945-14533
and
U.S. Department of Transportation
Federal Highway Administration
Bureau of Public Roads

College of Engineering
Office of Research Services
University of California
Berkeley, California

December 1971

ABSTRACT

A computer program, FINPLA2, is presented which can be utilized to analyze general highway bridge structures. This program can analyse general non-prismatic cellular structures of varying width and depth and may have an integrated three-dimensional frame. The structure is discretised by dividing it longitudinally into a certain number of structure segments by vertical sections, and by subdividing each such segment into finite elements. The structure alignment is described by a longitudinal reference line which may be a straight line, a circular curve or an arbitrary planar string polygon and cross-sections are defined with respect to this line. Orthotropic plate properties and arbitrary loadings and boundary conditions can be treated. Automatic element and coordinate generation options minimise the required input data. Several numerical examples illustrating the use of the program are also given.

KEY WORDS

Box girder bridges, multi-cell bridges, skew bridges, interchange structure, bridge bents, three-dimensional frames, non-prismatic folded plates, orthotropic folded plates, anisotropic folded plates, elastic analysis, structural analysis, structural design, finite elements, direct stiffness method.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure | Page

1. INTRODUCTION

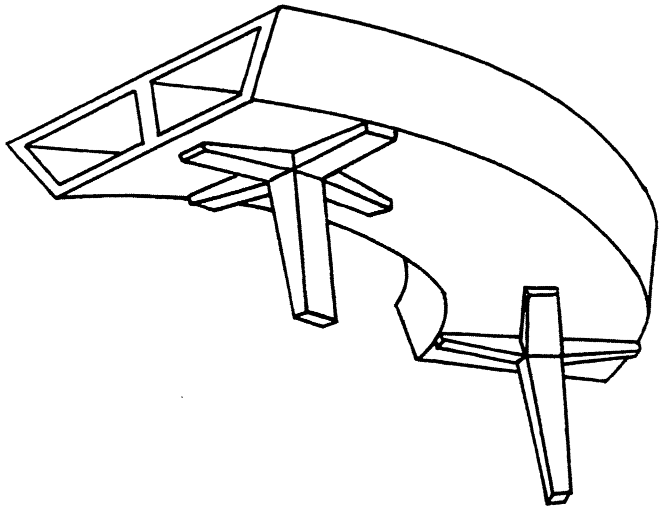
A continuing program of analytical and experimental research on box girder bridges has been conducted at the University of California since 1965. Simple and continuous spans having plan views which are straight, skew or curved have been studied. Detailed information on this research including listings of computer programs developed may be found in a series of published research reports [1 - 9].

The purpose of this report is to present a computer program for the linear elastic analysis of folded plate structures with cross sections which may vary along the span. The horizontal alignment may be arbitrary, and the plate structure can be integrated into a general three-dimensional frame.

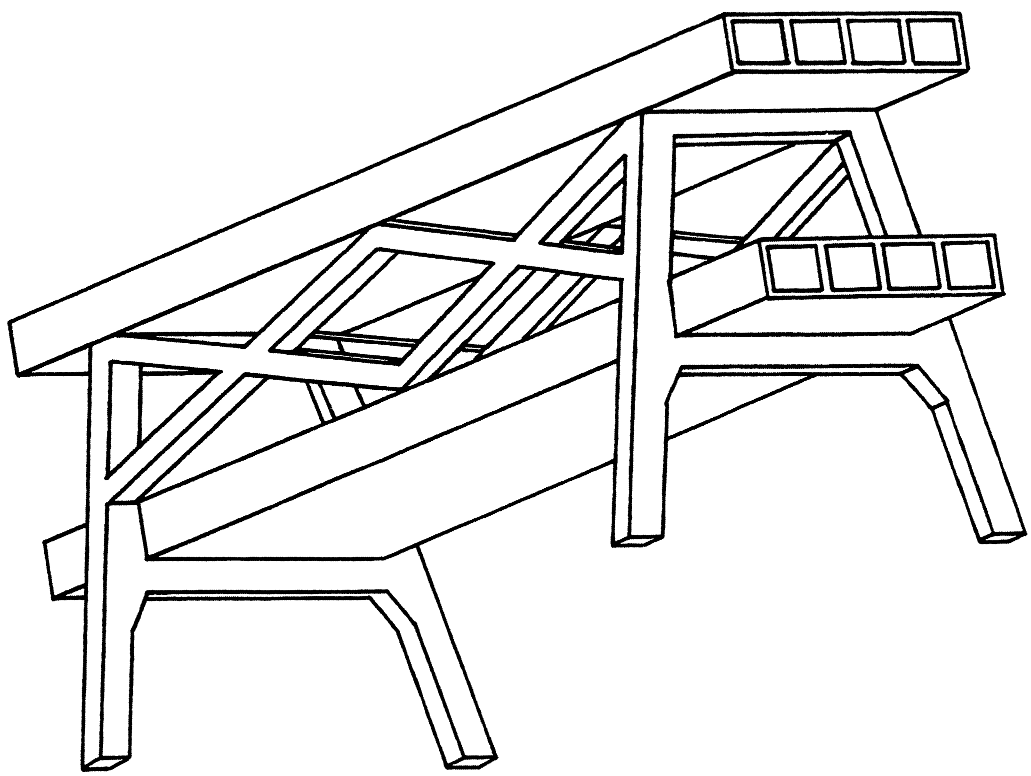
The program, called FINPLA2, is the latest in a series of programs developed at the University of California at Berkeley for the analysis of box girder bridges and is as such a generalization of a program previously reported in [5].

Although primarily written for the analysis of bridges, the program may be used for analyzing a large class of plate structures as well as general three-dimensional frames. However, the program is utilized with most advantage, if applied to plate structures which act integrally together with a three-dimensional frame, Fig. 1.1.

Bridge structures must have fairly simple geometries if closed-form analytical techniques are to be used for their analysis. For straight bridges with prismatic cross sections, efficient and accurate methods have been developed based on harmonic analysis and programmed for digital computers, both for simple span as well as continuous bridges [1, 2].



a) COLUMN-SUPPORTED CURVED BOX GIRDER BRIDGE



b) DOUBLE-DECK BOX GIRDER WITH SPACE-FRAME ACTIO

FIG. 1.1 STRUCTURES WITH FOLDED PLATE - 3D FRAME INTERACT

Also curved bridges may be accurately analyzed [6, 8] by this method as long as they are of constant curvature and cross section, with simply supported non-skew end sections.

However, bridge structures appearing in modern highway systems are very often of varying cross section. Long span box girder bridges require a variation of depth and moment of inertia, Fig. 1.2, to be economical. A first attempt of an analytical treatment of such bridges has been published recently [10]. Also, for box girder bridges of arbitrary plan geometry, but with the restriction of constant depth between top and bottom slabs, Willam and Scordelis [7] developed a finite element computer program. These bridges may be tapered in plan as they appear at on- and off-ramps in interchange structures, Fig. 1.3.

Finally, it is often difficult to predict the interaction between a bridge structure and its supporting frame or bents, although the nature of this interaction may be of prime importance for the state of stress in the bridge as well as in the bents.

For such complex structural systems as outlined above, only the finite element method of analysis seems to be versatile enough to solve these problems efficiently and accurately so as to be of practical use. Program FINPLA2 is a finite element program and may be used to analyze structures with varying depth and/or width. The structure should be of prismatic nature, i.e., the alignment should be basically one-dimensional. Thus, the examples of Fig. 1.1 may be analyzed, however, the case depicted in Fig. 1.3 b, cannot readily be treated, unless modelled as described in Chapter 4.

The theory of the elements used in FINPLA2 and a short summary of

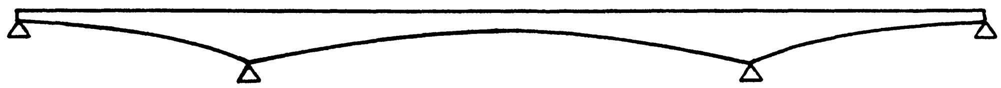
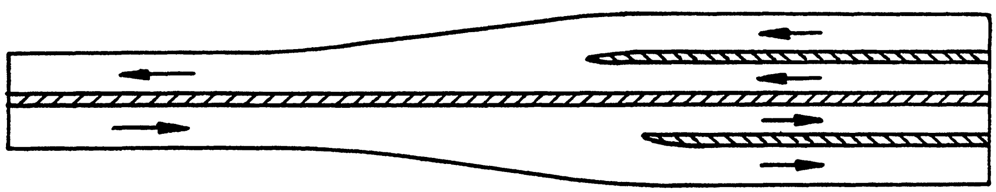
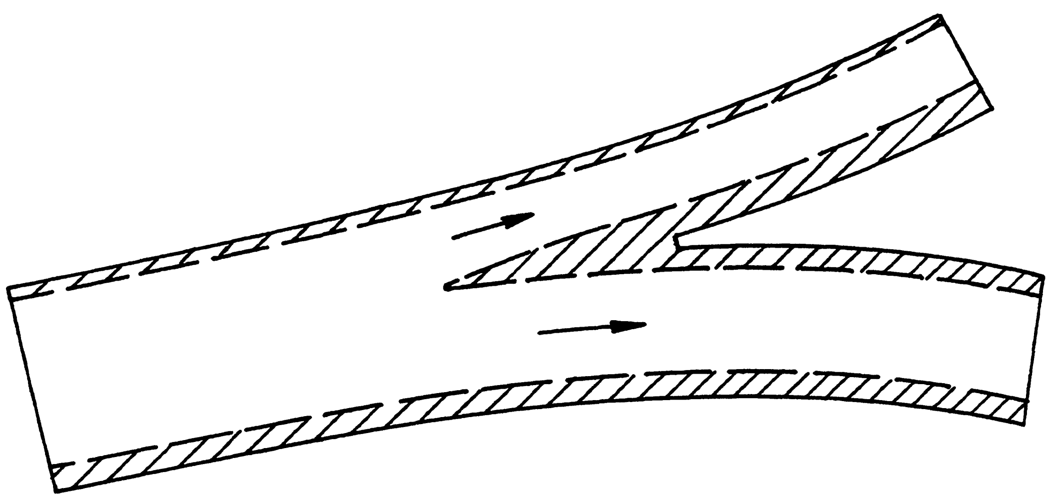


FIG. 1.2 CONTINUOUS BRIDGE WITH TAPERED DEPTH



a) HIGHWAY SECTION WITH TAPERED WIDTH



b) BRIDGE STRUCTURE WITH OFF-RAMP

FIG. 1.3 TYPICAL EXAMPLES OF BRIDGES WITH TAPERED WIDTH the general capabilities of the program have been described in [8]. In this report, more detailed programming information will be given, together with a complete description of input preparation and output generation. Some examples will illustrate the use of the program as well as its accuracy. The complete FORTRAN IV source deck listing and some supplementary program information are given in the Appendices.

2. METHOD OF ANALYSIS

2.1 Finite Element Analysis

The finite element method of analysis on which program FINPLA2 is based, has been developed in recent years and has found useful applications in numerous problems of structural engineering and structural mechanics. Its versatility lends itself to the analysis of such complex structures as described in this report. An extensive amount of literature on this powerful method has been published, and therefore only those aspects necessary for a better understanding of program FINPLA2 will be shortly described below.

Much research has already been directed toward the objective of obtaining finite element models with optimum properties for certain problems, both regarding accuracy of results and expenditure in computation. For the analysis of folded plate structures, it appears advantageous to assign six degrees of freedom to each nodal point, three translational, and three rotational. Although five degrees of freedom have often been used for general shell programs, special techniques are required to take account of the missing sixth degree of freedom, i.e., the rotation in the plane of the shell or plate. The addition of the sixth degree of freedom in FINPLA2 is particularly important since the finite element system is combined with a three-dimensional frame in which the member nodes also have six degrees of freedom. The elimination of the complications associated with the five degrees of freedom system justifies the increase in computational effort involved in solving the larger number of equations.

The actual step-by-step solution process followed in program FINPLA2 may be summarized as follows.

1. Discretize the structure into a number of quadrilateral elements by dividing it longitudinally into a number of segments, and transversely into individual elements, Fig. 2.1a.
2. Calculate all element stiffnesses and transform them to a common global coordinate frame.
3. Form the structure stiffness by assembling the element stiffnesses according to the standard direct stiffness method of structural analysis.
4. Set up the load vector containing all specified external loads. For loads distributed over the area of an element, calculate equivalent nodal loads using the tributary area principle or some other appropriate technique such as consistent load theory. If more than one load case is considered, a load matrix has to be established.
5. Apply given boundary conditions to the set of equations. If non-zero displacements are specified, a modification of the load vector (or matrix) will be necessary.
6. Solve the system of equations by a technique which takes advantage of the banded nature of the structural stiffness. If also the variation of the bandwidth is taken into account, considerable reduction in computational effort may result.
7. Once the nodal joint displacements are known from the solution of the equations of step 6, reactions may be found by multiplying the original structure stiffness (i.e., not modified due to boundary conditions) by the displacement vector. A complete matrix multiplication will also yield the residual

loads on the structure which are a measure for the accuracy with which the equations had been solved.

8. Transform nodal displacements from global to element coordinates for the computation of internal stresses and moments in plate and beam elements.

2.2 Finite Element Models

Assuming linear elastic material behavior and small deflections, bending and in-plane actions of a plate element are uncoupled, and therefore one may treat these two problems independent of each other.

The plane stress element model used in program FINPLA2, was derived in its original form by Abu Ghazaleh [11] for a rectangular shape and has also been described in detail in [2]. Willam [12] rederived the element stiffness in skew natural coordinates for a general quadrilateral shape. The special feature of this element is that in addition to the commonly used two translational degrees of freedom per node, each node is assigned a rotational degree of freedom, defined as the averaged rotation about the element z-axis, Fig. 2.1 b, i.e., for a rectangular element,

$$\theta = \frac{1}{2}\left[\left(\frac{\partial v}{\partial x}\right)_i - \left(\frac{\partial u}{\partial y}\right)_i\right] \qquad (1)$$

thus raising the total number of degrees of freedom for the element to 12. The actual element displacements u and v are assumed to vary linearly with the translational degrees of freedom u_i, v_i, and as beam functions with the rotational degrees of freedom θ_{z_i}, Fig. 2.1 d. The assumed nodal rotations introduce angular discontinuities at the nodes so that the element is incompatible.

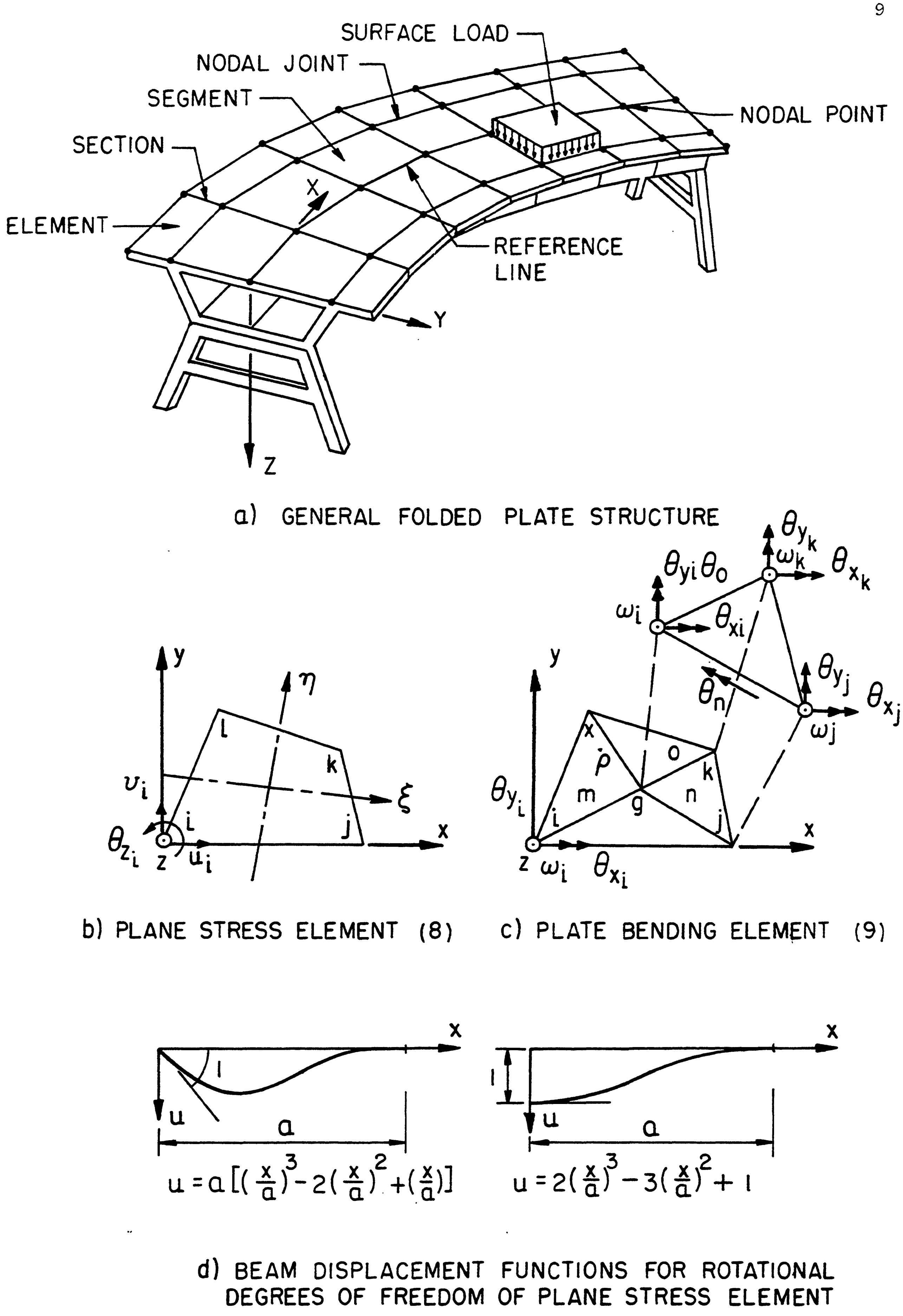


FIG. 2.1 FINITE ELEMENT MODELS

The plate bending element employed in program FINPLA2 has been derived by Clough and Felippa [13]. This compatible quadrilateral element is made up of four subtriangles, each of which has 11 degrees of freedom, Fig. 2.1 c. In combining the four subelements, a quadrilateral with 19 degrees of freedom is obtained. However, the 7 internal degrees of freedom can be eliminated from the element stiffness by static condensation, reducing the stiffness to the essential 12 degrees of freedom, for each node the rotations θ_{x_i} and θ_{y_i} and the translation w_i.

The class of structural configurations amenable to program FINPLA2 may involve quadrilateral elements whose four corner points do not lie in one plane. The resulting problem is solved by assuming that the warpage of realistic structures is small so that a plane may be sought which minimizes the sum of the distances squared from the plane to the four corner nodes. Projecting the four nodes onto this plane, a new imaginary plane quadrilateral element is found, the stiffness of which should be very close to the stiffness of the actual warped element. It can be shown that the least square fitting of the plane leads to an eigenvalue problem of third order with the three direction cosines of the desired plane normal being contained in the solution eigenvector.

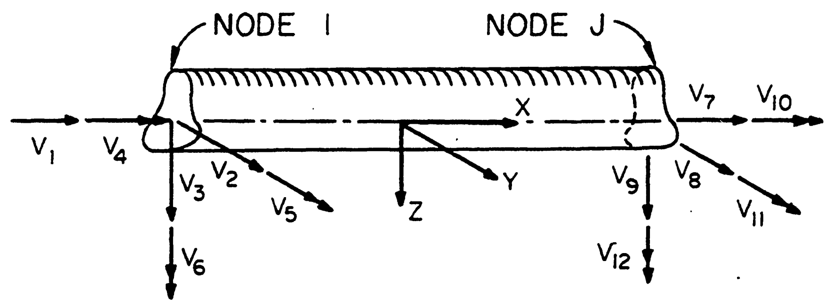
The beam elements making up the three-dimensional frame assembly have the 12 degrees of freedom shown in Fig. 2.2 a, and the complete element stiffness including shearing deformations, is given by Eq. (2) in which

$$\Phi_y = \frac{12EI_z}{GA_{sy}L^2} \quad \text{and} \quad \Phi_2 = \frac{12EI_y}{GA_{sz}L^2}$$

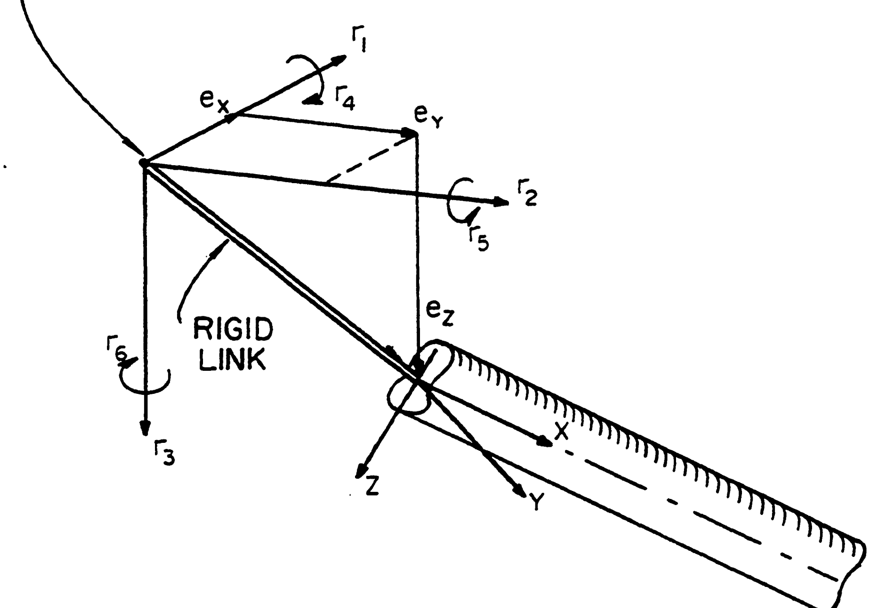
with A_{sy} and A_{sz} being the effective shear areas.

$$
[K] = \begin{bmatrix}
\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{12EI_z}{L^3(1+\Phi_y)} & 0 & 0 & 0 & \frac{6EI_z}{L^2(1+\Phi_y)} & 0 & \frac{-12EI_z}{L^3(1+\Phi_y)} & 0 & 0 & 0 & \frac{6EI_z}{L^2(1+\Phi_y)} \\
0 & 0 & \frac{12EI_y}{L^3(1+\Phi_z)} & 0 & \frac{-6EI_y}{L^2(1+\Phi_z)} & 0 & 0 & 0 & \frac{-12EI_y}{L^3(1+\Phi_z)} & 0 & \frac{-6EI_y}{L^2(1+\Phi_z)} & 0 \\
0 & 0 & 0 & \frac{GJ_x}{L} & 0 & 0 & 0 & 0 & 0 & \frac{-GJ_x}{L} & 0 & 0 \\
0 & 0 & \frac{-6EI_y}{L^2(1+\Phi_z)} & 0 & \frac{(4+\Phi_z)EI_y}{L(1+\Phi_z)} & 0 & 0 & 0 & \frac{6EI_y}{L^2(1+\Phi_z)} & 0 & \frac{(2-\Phi_z)EI_y}{L(1+\Phi_z)} & 0 \\
0 & \frac{6EI_z}{L^2(1+\Phi_y)} & 0 & 0 & 0 & \frac{(4+\Phi_y)EI_z}{L(1+\Phi_y)} & 0 & \frac{-6EI_z}{L^2(1+\Phi_y)} & 0 & 0 & 0 & \frac{(2-\Phi_y)EI_z}{L(1+\Phi_y)} \\
\frac{-EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{-12EI_z}{L^3(1+\Phi_y)} & 0 & 0 & 0 & \frac{-6EI_z}{L^2(1+\Phi_y)} & 0 & \frac{12EI_z}{L^3(1+\Phi_y)} & 0 & 0 & 0 & \frac{-6EI_z}{L^2(1+\Phi_y)} \\
0 & 0 & \frac{-12EI_y}{L^3(1+\Phi_z)} & 0 & \frac{6EI_y}{L^2(1+\Phi_z)} & 0 & 0 & 0 & \frac{12EI_y}{L^3(1+\Phi_z)} & 0 & \frac{6EI_y}{L^2(1+\Phi_z)} & 0 \\
0 & 0 & 0 & \frac{-GJ_z}{L} & 0 & 0 & 0 & 0 & 0 & \frac{GJ_x}{L} & 0 & 0 \\
0 & 0 & \frac{-6EI_y}{L^2(1+\Phi_z)} & 0 & \frac{(2-\Phi_z)EI_y}{L(1+\Phi_z)} & 0 & 0 & 0 & \frac{6EI_y}{L^2(1+\Phi_z)} & 0 & \frac{(4+\Phi_z)EI_y}{L(1+\Phi_z)} & 0 \\
0 & \frac{6EI_z}{L^2(1+\Phi_y)} & 0 & 0 & 0 & \frac{(2-\Phi_y)EI_z}{L(1+\Phi_y)} & 0 & \frac{-6EI_z}{L^2(1+\Phi_y)} & 0 & 0 & 0 & \frac{(4+\Phi_y)EI_z}{L(1+\Phi_y)}
\end{bmatrix} \tag{2}
$$

EQ. (2) STIFFNESS MATRIX FOR FRAME ELEMENT



a) DEGREES OF FREEDOM OF BEAM ELEMENT



b) ECCENTRIC BEAM CONNECTION

FIG. 2.2 BEAM ELEMENT NOTATIONS

In order to allow for arbitrary eccentricities of the beam elements, rigid links are introduced to connect the beam nodes with those nodal points predefined by plate type finite elements, Fig. 2.2 b. The transformation relating the joint displacement degrees of freedom r_i to the beam deformations v_i is given for either one of the two end points by

$$\begin{Bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{Bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & (a_{13}e_y - a_{12}e_x) & (a_{11}e_z - a_{13}e_x) & (a_{12}e_x - a_{11}e_y) \\ a_{21} & a_{22} & a_{23} & (a_{23}e_y - a_{22}e_z) & (a_{21}e_z - a_{23}e_x) & (a_{22}e_x - a_{21}e_y) \\ a_{31} & a_{32} & a_{33} & (a_{33}e_y - a_{32}e_z) & (a_{31}e_z - a_{33}e_x) & (a_{32}e_x - a_{31}e_y) \\ 0 & 0 & 0 & a_{11} & a_{12} & a_{13} \\ 0 & 0 & 0 & a_{21} & a_{22} & a_{23} \\ 0 & 0 & 0 & a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{Bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \end{Bmatrix} \tag{3}$$

where the a_{ij} quantities are the direction cosines of the element axes, and e_x, e_y, e_z constitute the components of eccentricity of the beam node under consideration. Abbreviating Eq. (3) as

$$\{v_i\} = [a_i] \; \{r_i\} \tag{4}$$

where subscript i refers to the degrees of freedom of node i, the complete element displacement transformation would follow as

$$\begin{Bmatrix} v_i \\ v_j \end{Bmatrix} = \begin{bmatrix} a_i & 0 \\ 0 & a_j \end{bmatrix} \begin{Bmatrix} r_i \\ r_j \end{Bmatrix} \tag{5}$$

so that the beam element stiffness in global coordinates becomes

$$[\bar{k}] = \begin{bmatrix} a_i & 0 \\ 0 & a_j \end{bmatrix}^T [k] \begin{bmatrix} a_i & 0 \\ 0 & a_j \end{bmatrix} \qquad (6)$$

2.3 Additional Remarks

For straight structures, it is logical to transform all element stiffness matrices to one common global coordinate system in which also the applied loads are expressed. Resulting displacements and reactions will also refer to this global frame of reference. In structures with circular curved or arbitrary horizontal alignment, however, one global Cartesian reference coordinate system may not be the most desirable. For example, circular curved structures are most appropriately analyzed in cylindrical coordinates.

In FINPLA2, a structure may be analyzed either in fixed global coordinates, XYZ, or in so-called travelling coordinates referring to only section i of the structure, $X_iY_iZ_i$, Fig. 2.3. Thus, in calculating the stiffness for all elements within segment i, those degrees of freedom associated with section i will be transformed to the coordinate frame $X_iY_iZ_i$ while the degrees of freedom associated with section i + 1 will be transformed to the coordinate frame $X_{i+1}\ Y_{i+1}\ Z_{i+1}$. In matrix notation, this process may be described as

$$[\bar{k}] = [B]^T [A]^T [k] [A] [B]$$

in which [k] is the 24 by 24 element stiffness in local coordinates

$[\bar{k}]$ is the 24 by 24 element stiffness in travelling coordinates, and

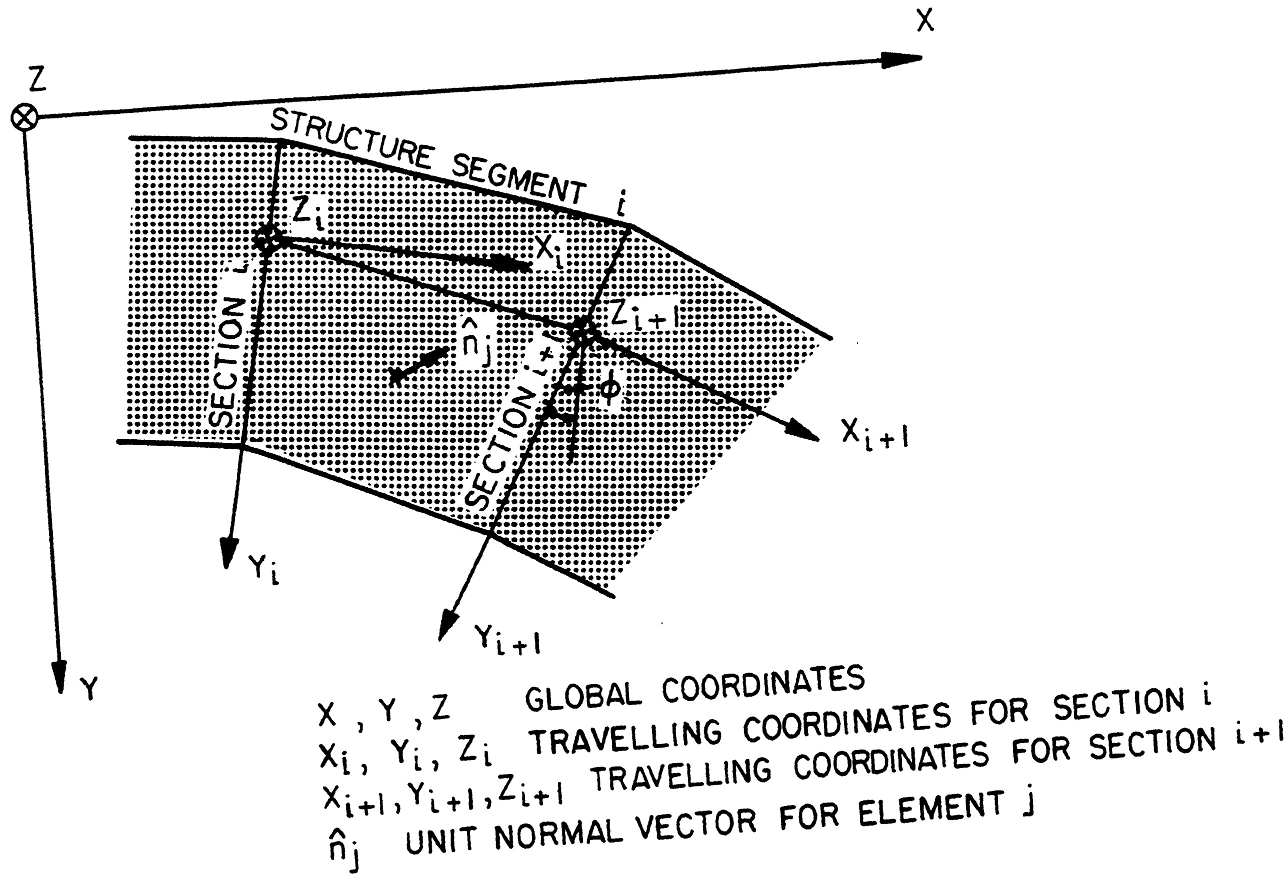


FIG. 2.3 GLOBAL AND TRAVELLING COORDINATES

$$[A] = \begin{bmatrix} a & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a \end{bmatrix} \qquad [B] = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & b & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & b & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & b & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & b \end{bmatrix}$$

with

$$[a] = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad [b] = \begin{bmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad [I] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The a_{ij} quantities are the direction cosines for the unit normal vector of the element under consideration, with respect to the travelling coordinates X_i, Y_i, Z_i, and φ is the angle between section i and i + 1.

Internal stress resultants of plate elements are usually calculated in local element coordinates which are more closely associated with the travelling coordinates $X_i Y_i Z_i$ than with X_{i+1} Y_{i+1} Z_{i+1}, Fig. 2.4.

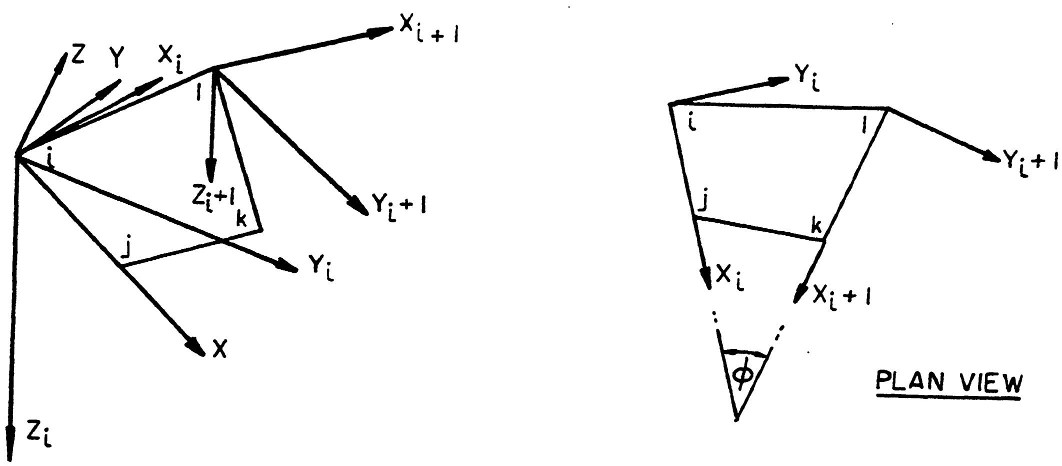


FIG. 2.4 ELEMENT COORDINATES X Y Z

Therefore, stress resultants for nodal points k and ℓ will be more informative if they are transformed as follows.

$$\begin{Bmatrix} N_x \\ N_y \\ N_{xy} \\ M_x \\ M_y \\ M_{xy} \end{Bmatrix}_{i+1} = \begin{bmatrix} \cos^2\varphi & \sin^2\varphi & -2\sin\varphi\cos\varphi & 0 & 0 & 0 \\ \sin^2\varphi & \cos^2\varphi & 2\sin\varphi\cos\varphi & 0 & 0 & 0 \\ \sin\varphi\cos\varphi & -\sin\varphi\cos\varphi & (\cos^2\varphi - \sin^2\varphi) & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos^2\varphi & \sin^2\varphi & -2\sin\varphi\cos\varphi \\ 0 & 0 & 0 & \sin^2\varphi & \cos^2\varphi & 2\sin\varphi\cos\varphi \\ 0 & 0 & 0 & \sin\varphi\cos\varphi & -\sin\varphi\cos\varphi & (\cos^2\varphi - \sin^2\varphi) \end{bmatrix} \begin{Bmatrix} N_x \\ N_y \\ N_{xy} \\ M_x \\ M_y \\ M_{xy} \end{Bmatrix}_{i} \quad (7)$$

where subscript i and i + 1 refer to the element coordinates associated with the edges i - j and k - ℓ, respectively, and φ is the angle between the two edges, Fig. 2.4.

3. PROGRAM DESCRIPTION

The program has been written in Fortran IV language for the CDC 6400 computer of the Computer Center of the University of California at Berkeley, and has been partially tested and updated on the UNIVAC 1108 of the LOGICAL Computation Center in Los Angeles. It consists of one main program called FINPLA2 and 26 subroutines. The overall logic of the program is displayed in the flow chart of Fig. 3.1, and the complete Fortran source listing is given in Appendix C, with short descriptions of each subprogram provided on comment cards. Additional detailed input specifications are provided in Appendix A, therefore only a few remarks shall be made here which are thought to be useful for a better understanding of the functioning of the program.

1. An attempt was made to keep the program as machine-independent as feasible, therefore a conversion to other comparable computing equipment should be possible with a minimum amount of effort.
2. Almost all arrays have variable dimensions so that almost no restrictions regarding the number of elements, nodal points, material types, etc., exist. The length of the blocks into which the structure stiffness is divided, is determined on the basis of how much core storage is available as working area. Thus, for practical purposes, there is no limitation on the size of problems that may be analyzed.
3. The large size of problems that are likely to be analyzed by program FINPLA2 require considerable execution times even on today's fastest computers. Therefore, optimization of

computation techniques was attempted wherever feasible. Marked reductions in execution times were achieved by

a) employment of the equation solver USOL written by Wilson [14], which utilizes the "variable block length" so that practically no limitations regarding the number of equations, bandwidth, and number of simultaneously treated load cases exist. The efficiency regarding execution times is achieved by recognizing variations of the actual bandwidth and taking advantage of it, besides other optimized programming features;

b) avoiding repetitive calculation of equal element stiffness wherever feasible, because a large computational effort is required to calculate the 24 x 24 stiffness matrices for all plate type finite elements;

c) restricting the class of permissible structures to basically prismatic geometries, i.e., chain structures, and taking advantage of the inherent simplifications regarding input, output, mesh generation, etc.

4. The length of the program suggests employment of an overlay system. Although no such linkage system is utilized in the version listed in the Appendix, the program has been arranged such that an overlay system may be incorporated with little programming effort, if it is desired to reduce the storage required by the program code.

5. Six scratch files are used for temporary storage. The file

numbers are declared at the beginning of the main program and may be easily changed in accordance with the peripheral storage equipment of a specific computer.

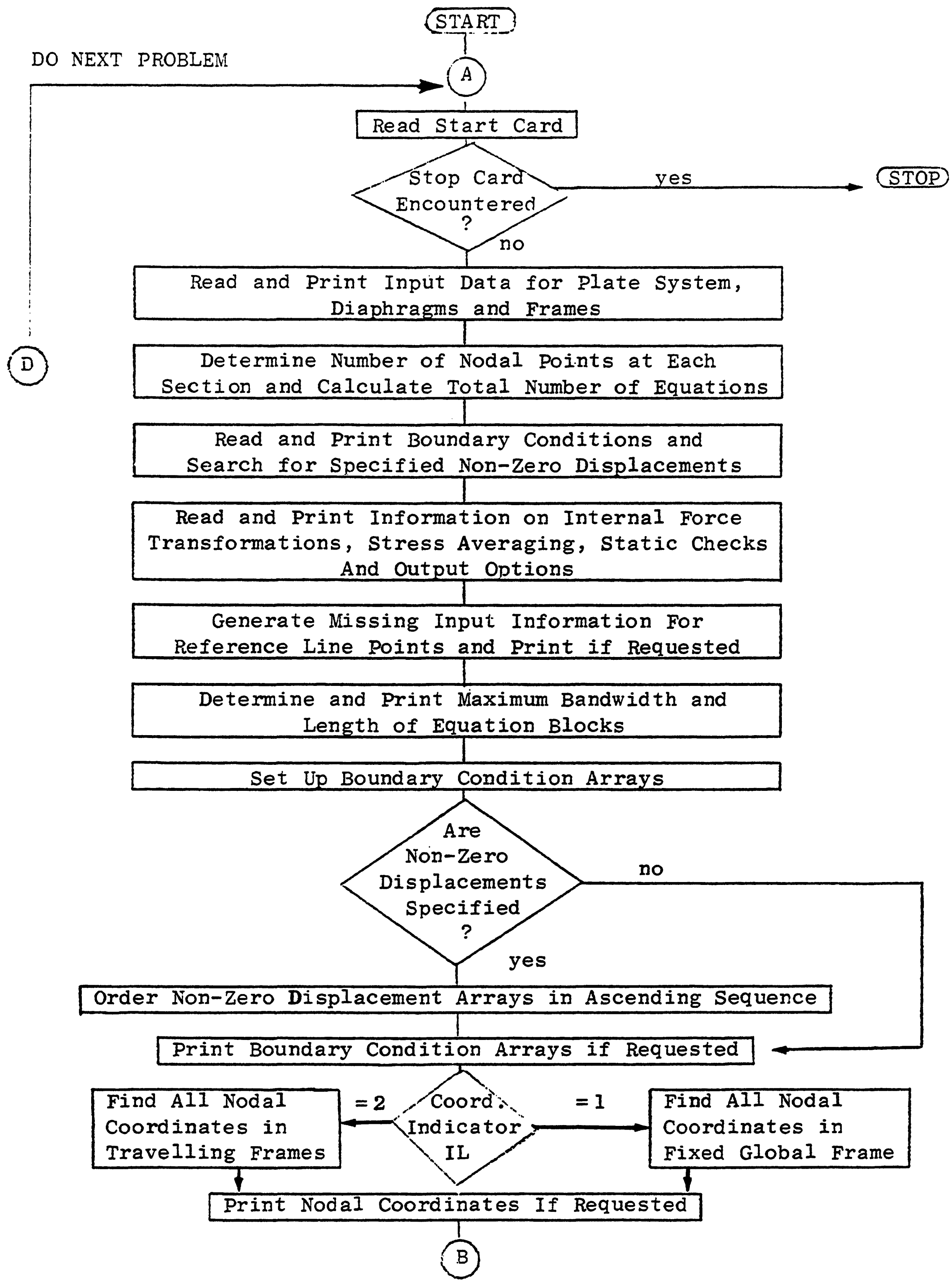


Fig. 3.1 FLOW CHART FOR OVERALL LOGIC OF PROGRAM FINPLA2

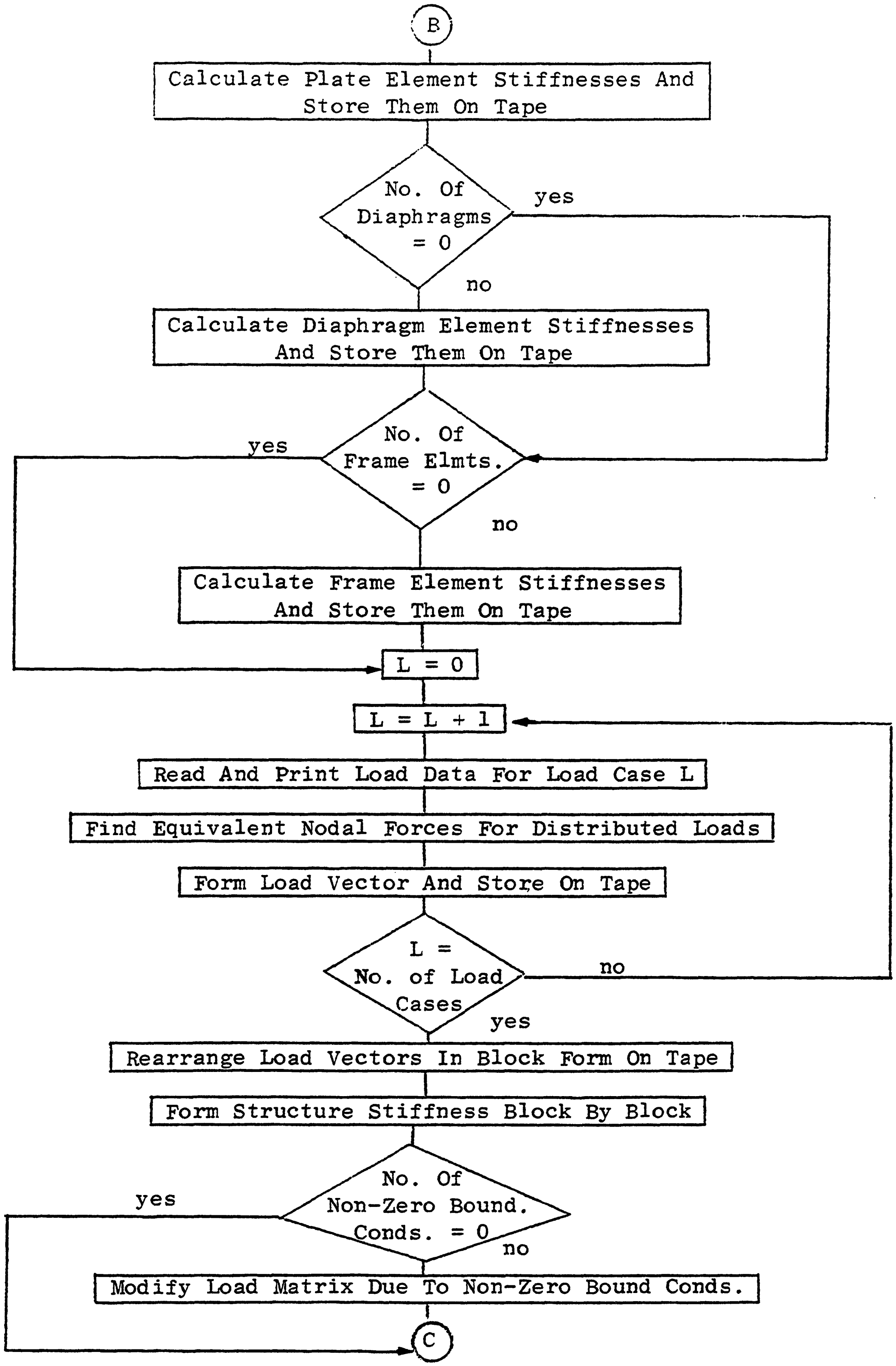


Fig. 3.1 (CONT'D) FLOW CHART FOR PROGRAM FINPLA2

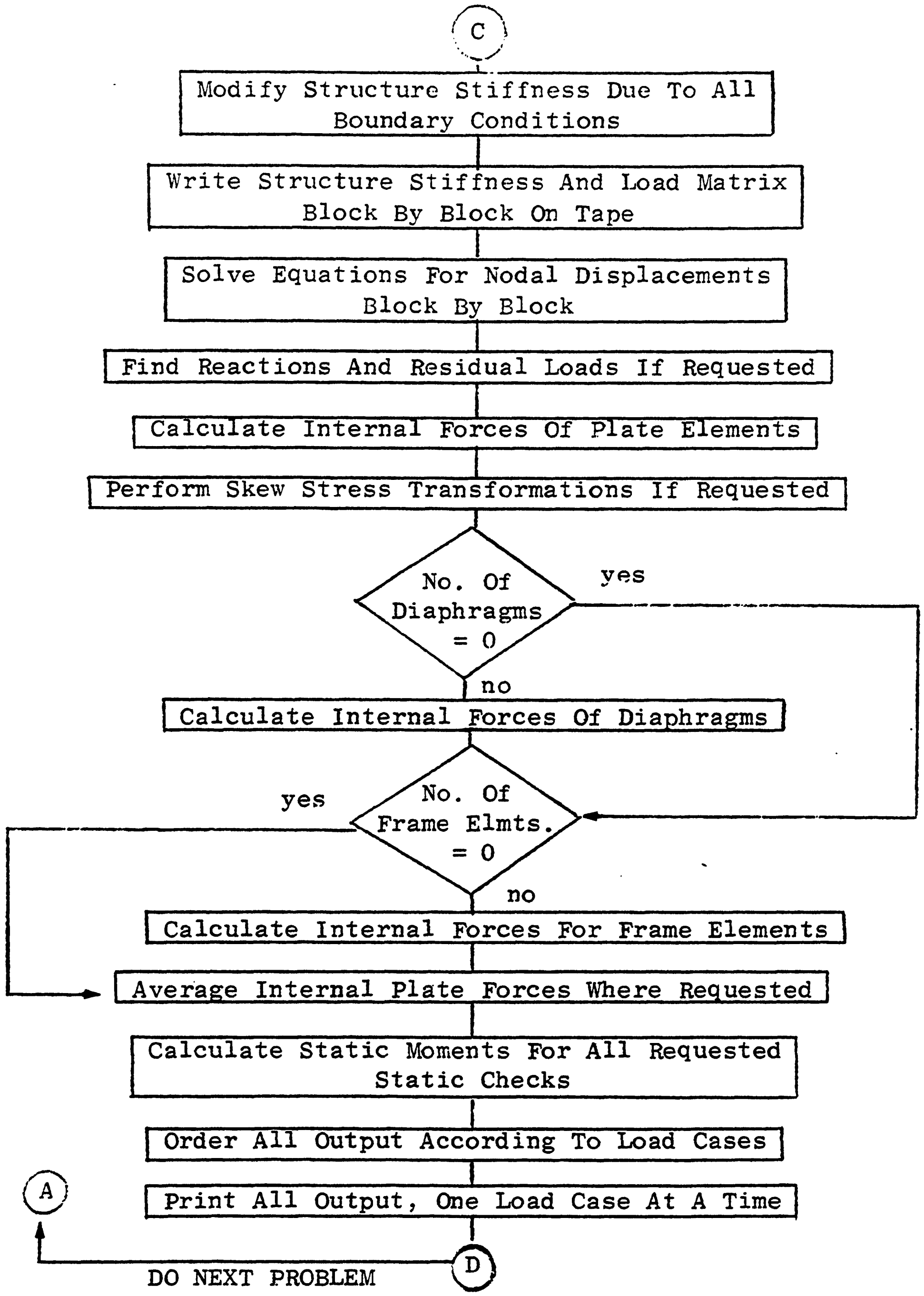


Fig. 3.1 (CONCLUDED) FLOW CHART FOR PROGRAM FINPLA2

4. PROGRAM USAGE

4.1 Capabilities and Restrictions

The program has been developed for the analysis of structures which are in general non-prismatic but basically one-dimensional in nature. Bridges are typical examples of this type of system. By describing a typical segment of the structure, only a minimum of additional input is needed to define all changes of cross sectional dimensions or plate properties, etc., occurring as one proceeds along the span.

It was found to be advantageous to describe the structure plan layout by means of a reference line. This reference line may be straight, circular curved, or a general planar string ploygon. Cross sections may vary arbitrarily provided the element layout does not change its connectivity, and the number of primary nodal points does not change. Thus, a highway branch such as shown in Fig. 4.1 may only be treated by carrying a constant number of elements and nodal points from end section to end section.

Although warped surfaces are quite unusual in real structures, they may be permitted, and how this case is treated internally by the program, was briefly discussed in Section 2.2.

Diaphragms can be placed at any transverse section of the structure as long as they are made up of quadrilateral elements. The nodal points do not have to be predefined by other plate elements and therefore do not have to lie in a common vertical plane, as long as they are properly defined as secondary nodal points (for definitions see next section). Possible diaphragm configurations are depicted in Fig. 4.2.

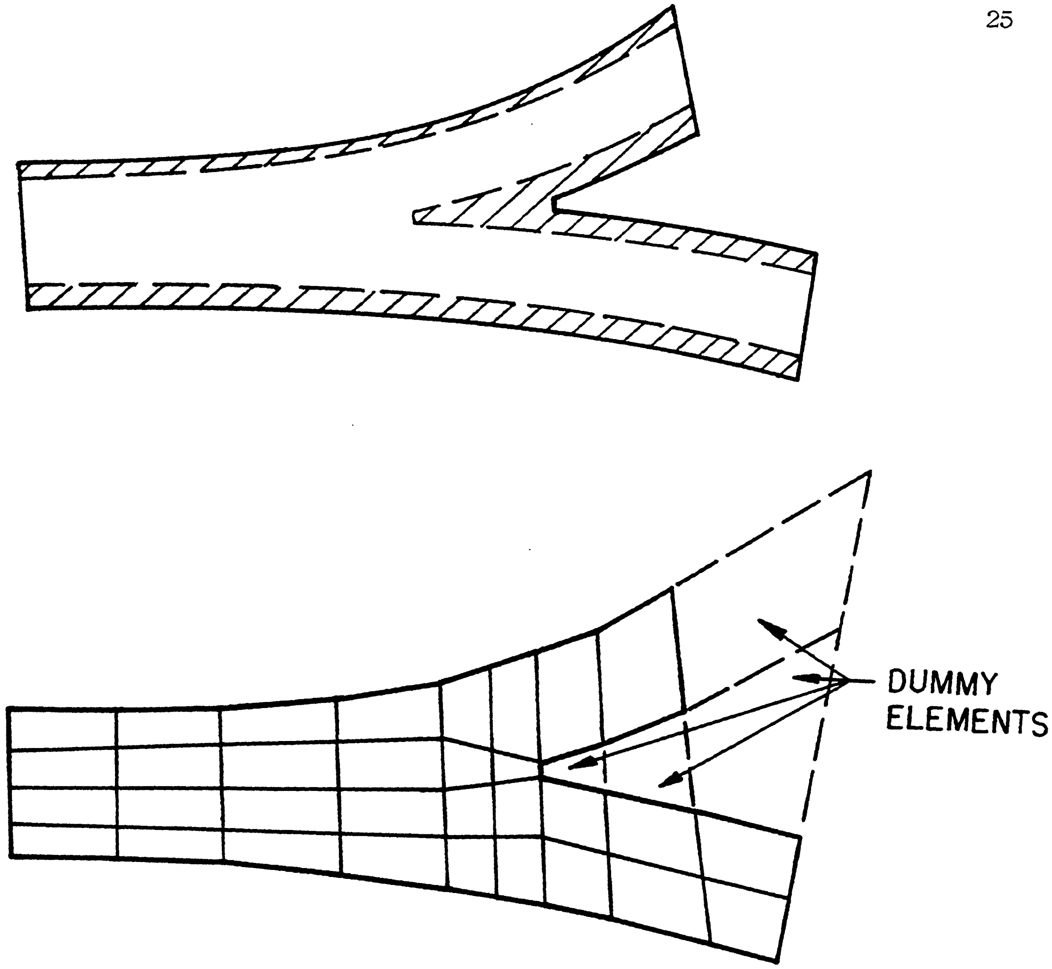


FIG. 4.1 HIGHWAY BRANCH IDEALIZATION

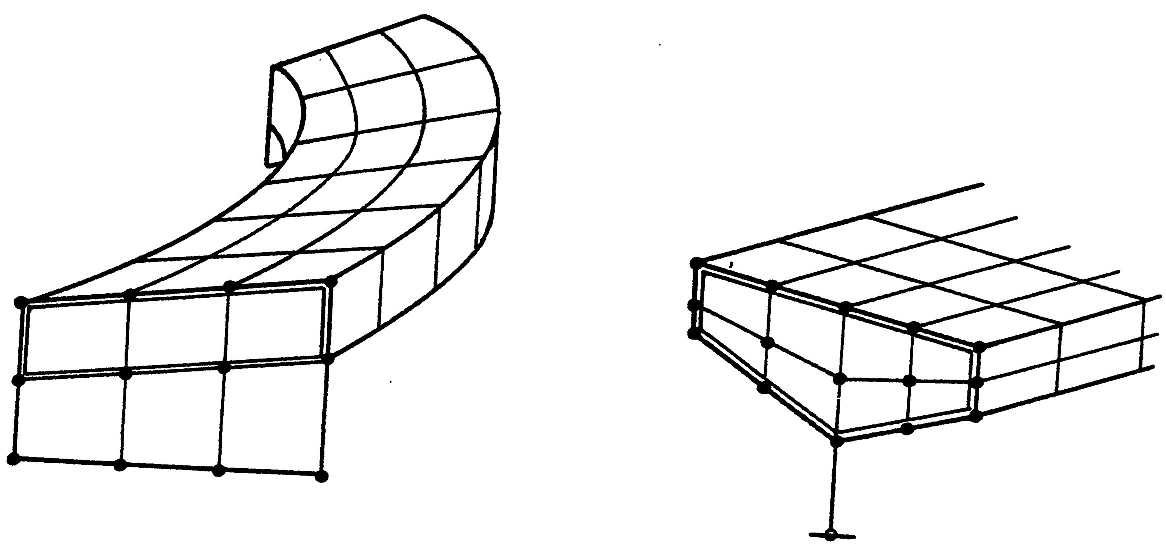


FIG. 4.2 TRANSVERSE DIAPHRAGMS

General frame elements with six degrees of freedom per end point may be connected to any nodal points of the structure provided they do not extend over more than one structure segment. Thus, frame elements which do extend over several segments have to be subdivided such as shown in Fig. 4.3. End points of the frame elements do not have to be predefined by plate elements, as long as they are declared as secondary nodal points. Fig. 4.3 shows some permissible three-dimensional frames.

Boundary conditions including specified non-zero displacements can be applied to any nodal point of the structure (primary or secondary), with any combination of the six degrees of freedom of the nodal point under consideration, and the user has the option to obtain all associated reaction forces and/or moments.

The structure may be subjected to an arbitrary number of load cases. However, the boundary conditions are assumed to be the same for all load cases. Loads may be applied either at the nodal points directly or as surface loads uniformly distributed over the entire area of any specified plate element. In the latter case, the program determines the equivalent nodal forces according to the tributary area principle.

For complex structural systems such as are likely to be analyzed by program FINPLA2, static checks are generally indispensable, for example, comparisons between the gross moment at some section due to external loads and due to internal stress resultants. For bridge structures, it is often important to know also the moment carried by each individual girder. These moments are determined by multiplying

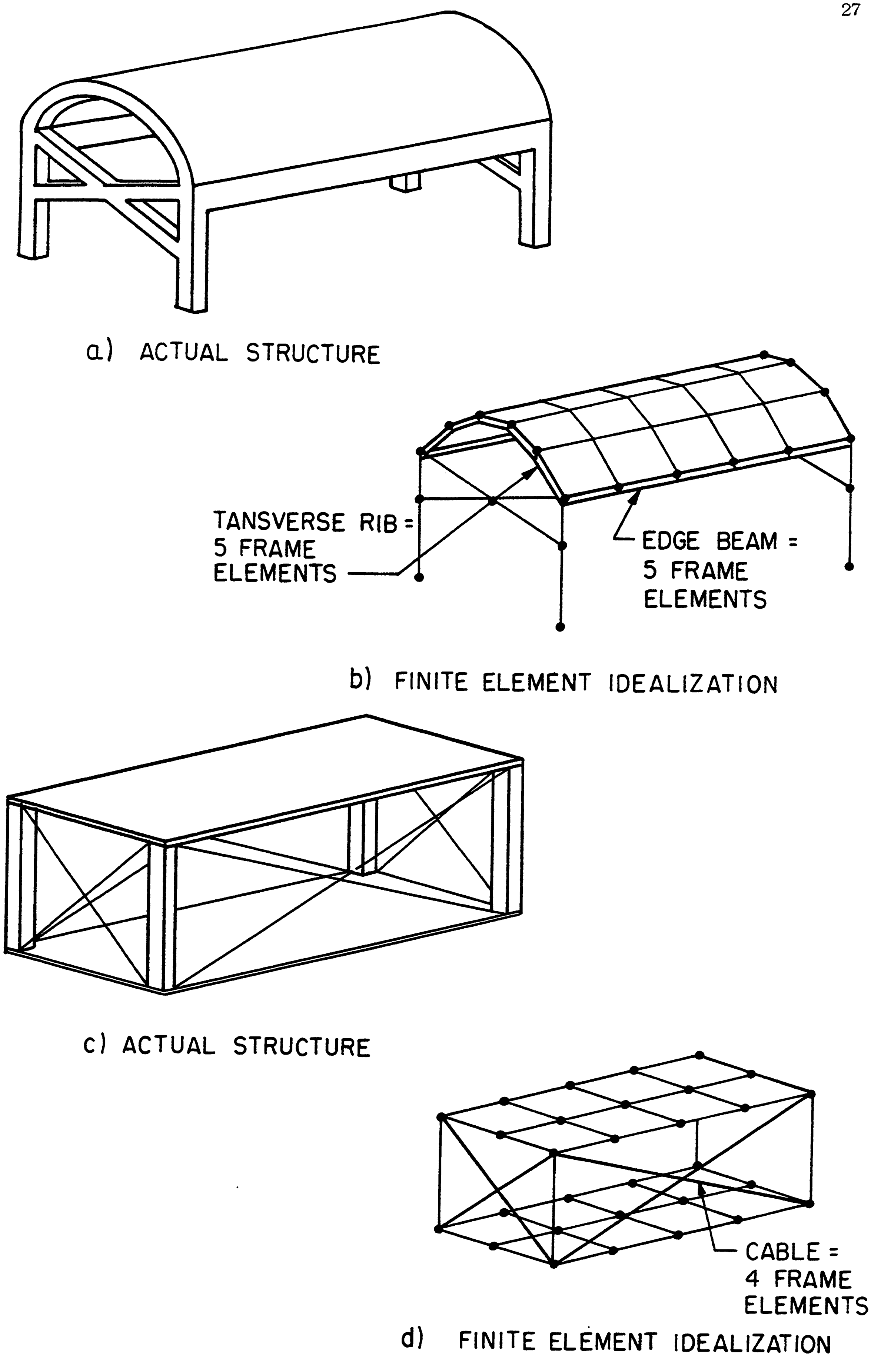


FIG. 4.3 IDEALIZATION OF STRUCTURES WITH FRAME ELEMENTS

the longitudinal stress resultants N_x with their respective lever arms to the neutral axis and integrating these differential moments over the area of a given girder. Longitudinal plate bending moments are also considered, although for cellular structures, their contribution will generally be small. In addition to girder moments, the total axial tension and compression forces in each girder are calculated and printed, the tensile forces readily indicating reinforcement requirements for reinforced concrete structures.

Care has to be taken in defining the neutral axis for the gross section, because the lever arm for stress resultant N_x is taken as the distance to the Y-axis of the travelling coordinate system. Although for a zero net axial force on the gross section, the total moment is independent of the location of the neutral axis, individual girder moments might depend on it because their net axial forces need not be zero. Therefore, if load-distribution characteristics of a structure are of interest, then the origin of the travelling coordinate system should lie on the neutral axis of the total cross section.

As was briefly discussed in Section 2.3, it might be desired in certain cases to transform internal stress output to some other coordinate system. In order to give the user complete flexibility in defining this stress output which might be essential in skew systems, an angle may be defined for any nodal point in the structure, about which the internal stress resultants should be rotated.

Because of theoretical approximations inherent in the finite element method, internal stresses are in general discontinuous between adjacent elements. In order to arrive at some meaningful interpretation of stress output, two methods are often used. In the first method nodal stresses between adjacent elements are averaged to establish sample points for the desired stress surface. The other method accepts mid-element stresses as sample points. These mid-element stress values are directly output by program FINPLA2. If also average values at the corner nodes are desired, use of an option may be made whereby the stresses of up to four elements meeting at a nodal point are averaged. It should be cautioned, however, that use of this option not be made where physical conditions such as caused by interior webs or diaphragms might render average stress values meaningless.

Certain limitations regarding core storage requirements and input parameters will be mentioned in Section 4.5.

4.2 Structure Discretization and Definitions

Before preparing actual input data, the user has to properly discretize the structure he wants to analyze. Below, the essential steps in this process will be given, followed by the definitions of various terms used in the input specifications in Appendix A.

Step 1. Choose a global coordinate system X,Y,Z. The origin will generally be placed at one end of the structure, and the X-axis will point towards the other end, Fig. 4.4. For straight structures, the X-axis will logically be chosen to be parallel to the structure centroid. Also, for other plan layouts, the user might wish to give some meaning to the global X-axis, although he is free to place it anywhere. The Z-axis points always downward, and the Y-axis follows from the right-hand rule.

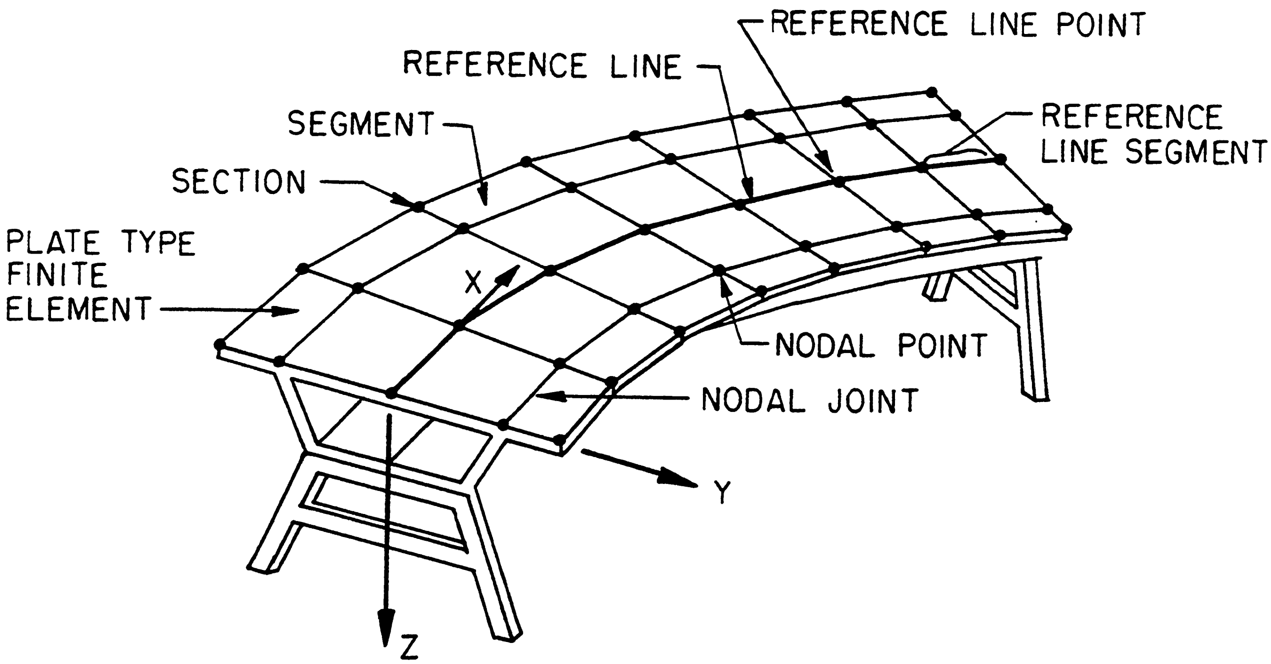


FIG. 4.4 STRUCTURAL DISCRETIZATION AND DEFINITIONS

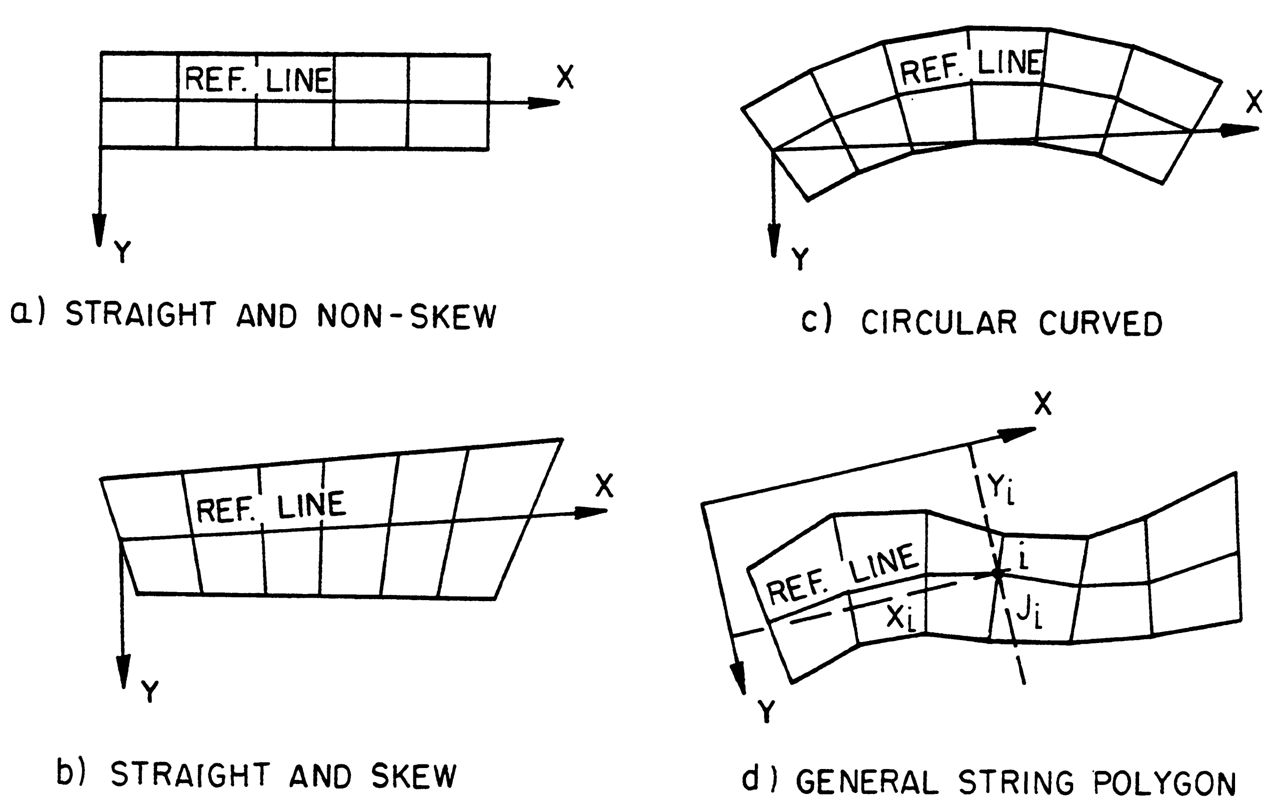


FIG. 4.5 TYPICAL REFERENCE LINE LAYOUTS

Step 2. Define a reference line. The main purpose of the reference line is to describe the general horizontal alignment of the structure. The reference line may be straight or circular curved, or it may be an arbitrary string polygon, Fig. 4.5, and is defined by the global X- and Y- coordinates of as many points as there are sections in the structure, with the first reference line point usually coinciding with the origin of the global coordinate system. For straight and circular curved layouts, one coordinate will suffice to define each reference line point. For irregular structures it will be advisable to discretize the structure first (as described below), and then to fit in the reference line such as to minimize input requirements. For structures with constant cross sections, the reference line points should be placed at some fixed position within each section so that the cross section needs to be described only once.

Step 3. Divide structure into segments. Place vertical sections through the structure at each reference line point, dividing it into n segments, Fig. 4.4. Any section i is then uniquely defined by the global X_i- and Y_i- coordinate of the corresponding reference line point, and by the angle between the section and the reference line tangent or the global Y- axis Fig. 4.5. Unless such an angle is specified, it will be assumed that the section bisects the angle between two consecutive reference line segments or is normal to the tangent, and end sections will be assumed to be normal to the first or last reference line segment. If transverse diaphragms are present, it is logical to define their planes as sections.

<u>Step 4.</u> Establish travelling coordinate systems $\bar{X}_i$, $\bar{Y}_i$, $\bar{Z}_i$. For many structures, in particular circular curved ones, it is advantageous to introduce a different coordinate system for each section i. The origin of any one such system coincides with the reference line point of that particular section i, the $\bar{Z}_i$- axis pointing downwards, and the $\bar{Y}_i$- axis lying also in the plane of the section such that the $\bar{X}_i$- axis points towards section i + 1, Fig. 4.6. Hence, all travelling coordinate systems are uniquely defined, once the reference line and the orientation of each section has been fixed. The coordinate system $\bar{X}_i$, $\bar{Y}_i$, $\bar{Z}_i$ is used to define the coordinates of all nodal points belonging to section i. External loads may also be measured in these coordinate frames, and reactions and displacements are usually most appropriately expressed in them. If this is the case, then the structure stiffness will be formed in travelling coordinates as described in Section 2.3.

<u>Step 5.</u> Divide structure segments into finite elements. The transverse division of a structure segment into a number of elements, which will in general be quadrilaterals, is assumed to be similar in all other segments, i.e., the number of elements and the joint numbering must be the same for all segments. In numbering the joints, care should be taken that the maximum nodal point difference for any one element is minimized, because this will reduce the bandwidth of the structure stiffness matrix and the required solution time.

The following definitions and remarks will help to clarify the terminology used in this input specifications, and will hopefully reduce the likelihood of improper program use.

A regular structure segment is defined to be any structure segment with a unique set of element types with the implication that many other segments are similar. Note that the geometry of the segments is irrelevant in this regard.

A section type is a cross section defined by a unique set of primary nodal point coordinates. Since nodal point coordinates are measured in travelling coordinates, two identical cross sections will be of different section types, if the reference line points are not at the same relative position, Fig. 4.6.

Primary nodal points are nodes at which plate type finite elements are interconnected, Fig. 4.6. The number of primary nodal points within a section must be the same for all other sections, and each node in a typical section is assigned a number which is taken to be the same for the corresponding points in all other sections, although their actual coordinates may vary from section to section. The string polygon connecting all primary nodal points with the same number is called a nodal joint. Primary nodal points must lie in the plane of their respective section.

Secondary nodal points are nodes either defined by end points of frame elements or created by subdividing a transverse diaphragm, without being predefined by plate type finite elements, Fig. 4.7. Thus, a node at which plate and beam elements are joined, will be a primary nodal point. Secondary nodal points may lie outside the plane of the section assigned to them.

A plate type finite element is a quadrilateral element defined within the structure by two nodal points, I and J, and by the number of

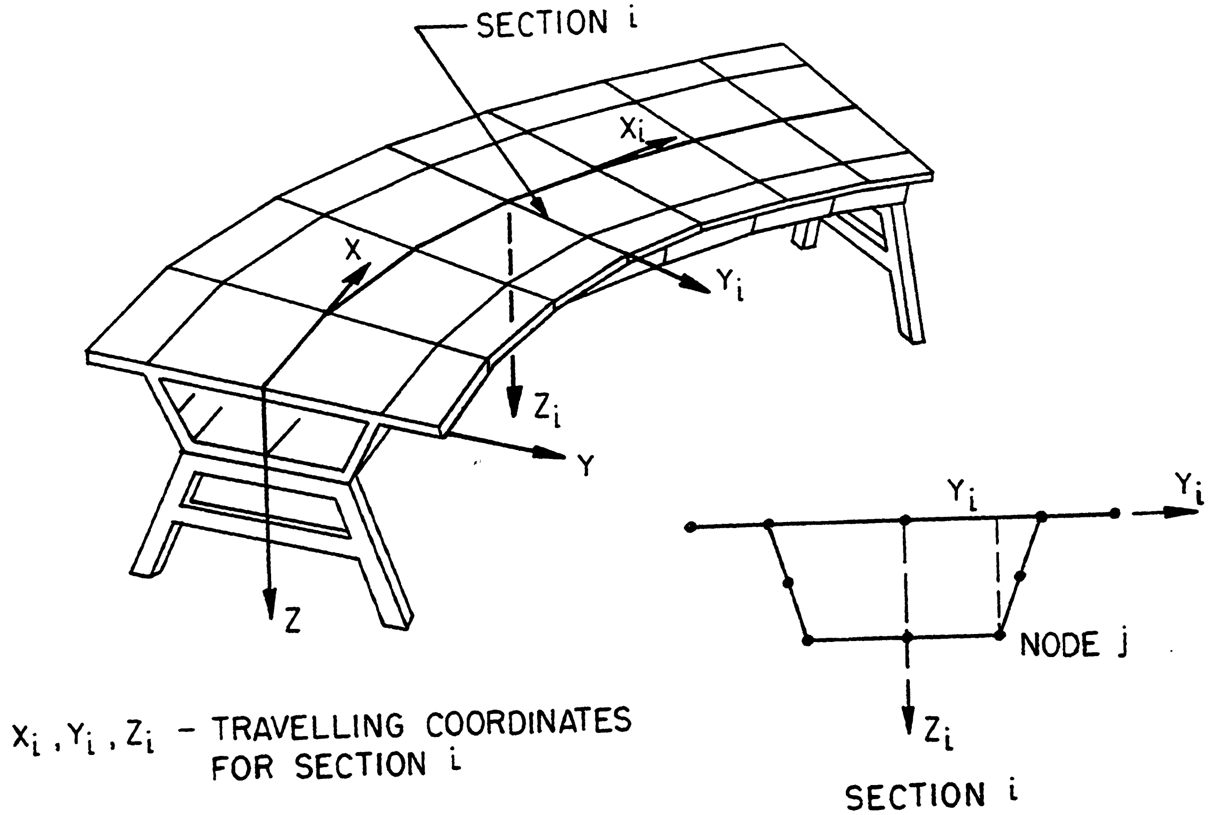


FIG. 4.6 DEFINITION OF PRIMARY NODAL POINTS

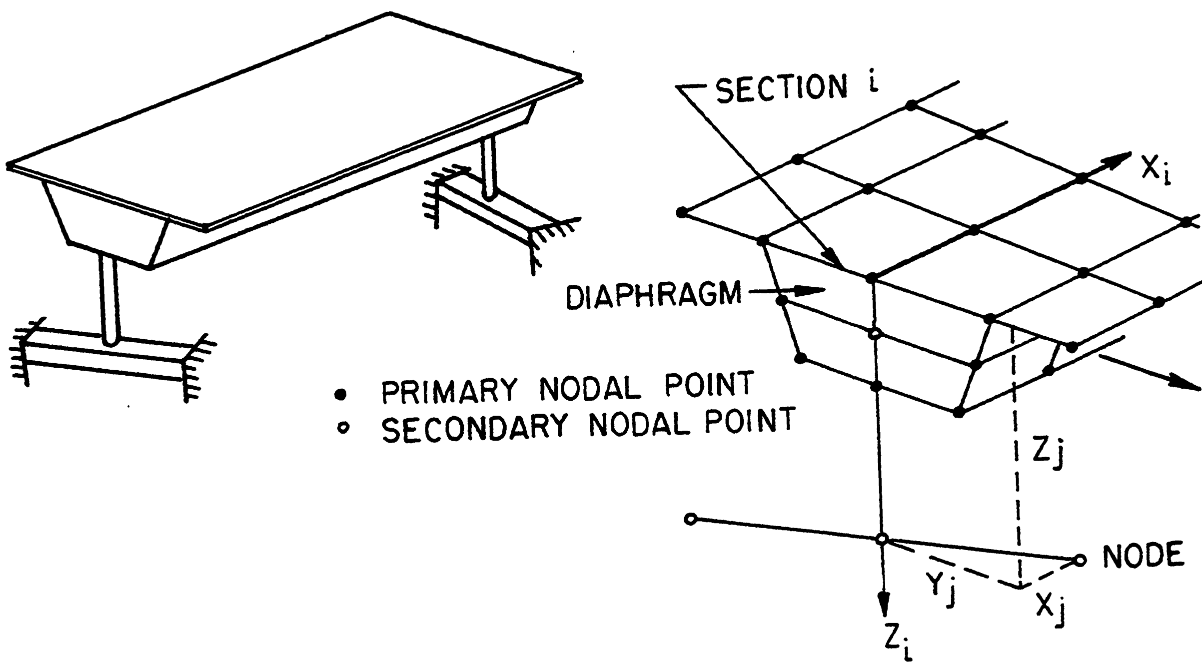


FIG. 4.7 DEFINITIONS OF SECONDARY NODAL POINTS

the segment of which it is a part. The element is assigned a number which is assumed to be the same in all other segments.

A finite element plate type is characterized by the thickness and material properties of an element, but not by its geometric properties or orientation in space.

A diaphragm is any assemblage of quadrilateral elements within any section and may be subdivided such that secondary nodal points are created. A diaphragm which is repeated often at other sections, i.e., with elements of the same plate type, may be defined as a regular diaphragm.

A special plate type finite element is a finite element with a different plate type (i.e., either different thickness or material properties) than the corresponding element in a regular structure segment.

Similarly, a special diaphragm element is characterized by a plate type different than for the corresponding element in a regular diaphragm.

One-dimensional beam or truss elements, including transverse or longitudinal ribs and stiffeners are referred to as frame elements. Either end point may be connected to any primary or secondary nodal point in the structure, with the restriction that they not extend over more than one structure segment. Eccentricities of connection are measured in the travelling coordinates for the section with which the point is associated, and are achieved by rigid links as described in Chapter 2.2.

A frame element section type is characterized by the axial stiffness and bending, torsional and shear rigidities of a frame element.

A frame element connection type is characterized by a unique set of eccentricities of connection regarding both end points, as well as by the rotation about the beam axis, which is measured by means of an auxiliary point lying in the x-z plane of the local frame element coordinate system, as described in Chapter 2.2.

An applied group displacement is a specified displacement component which is applied simultaneously to a designated group of nodal points in a given section. It may have zero or non-zero values.

Boundary conditions may also be applied as prescribed line displacements which are specified displacement components applied simultaneously to all nodal points lying on a specific nodal joint between two designated sections. These two sections may be identical if the displacement component is applied to only a single nodal point.

4.3 Input Specifications

Input data are key punched on cards as specified in Appendix A. It is very important that the sequential order given is strictly adhered to and consistent units are used throughout a problem.

4.4 Output Description and Interpretation

The output of each correctly executed job contains the following information:

1) The complete set of input data is printed out with proper headings for an easy check for input errors.

2) In addition, the user has the option to print out any combination of the following data for checking purposes:

   a) Internally generated reference point coordinates;

b) Internally generated nodal point coordinates in either global or travelling coordinates;

c) Complete boundary condition arrays;

d) Complete load vectors for all load cases;

e) All distinct element stiffnesses;

f) The complete structure stiffness before boundary conditions are applied to it.

3) The displacements of all nodal points in the structure are printed out and are positive in either the global or travelling coordinate system selected by the user.

4) The user has the option to calculate and print out all reactions in the structure, which will have non-zero values wherever the corresponding displacement components have been specified as input.

5) Similarly, another option permits the calculation and printing of all residual loads acting on the structure. These residual loads are defined by the matrix equation

$$R_r = K r - R$$

where

K = original structure stiffness matrix

r = final nodal displacement vector, containing all specified non-zero boundary displacements

R = input load vector, containing also all calculated reactions,

The residuals should theoretically be zero and practically very small quantities as compared to the applied loads and reactions.

The printout of the residual loads may be a valuable check on the accuracy of the equation solver. If they are not very small compared to the applied loads and reactions, then this will indicate either that there is an error in the input data, or that the set of equations is ill-conditioned or too large for single-precision accuracy of the computer, so that recourse to double-precision may have to be taken.

6) Internal stress resultants for all plate type finite elements are printed for all structure segments selected by the user. These stress resultants include the inplane forces N_x, N_y, N_{xy} (units force/length) and the plate-bending moments M_x, M_y, M_{xy} (units force-length/length) and are given at the four corner nodes as well as at the element center, Fig. 4.8. Unless a linear stress transformation has been requested as will be described below, internal stress resultants will be printed in element coordinates as defined in Section 4.2.

7) Similarly, internal stress resultants for all diaphragm elements are printed for all diaphragms selected by the user. These stress resultants include N_x, N_y, N_{xy}, M_x, M_y, M_{xy} and are given at the four corner nodes and at the element center, and they are always given in local element coordinates.

8) For all frame elements, the end shears V_x and V_y, the axial force S, as well as the bending moments M_y and M_z and the torque T are printed out, Fig. 4.9.

9) If use of the static check option is made, statical moments and total axial tension and compression forces are printed for

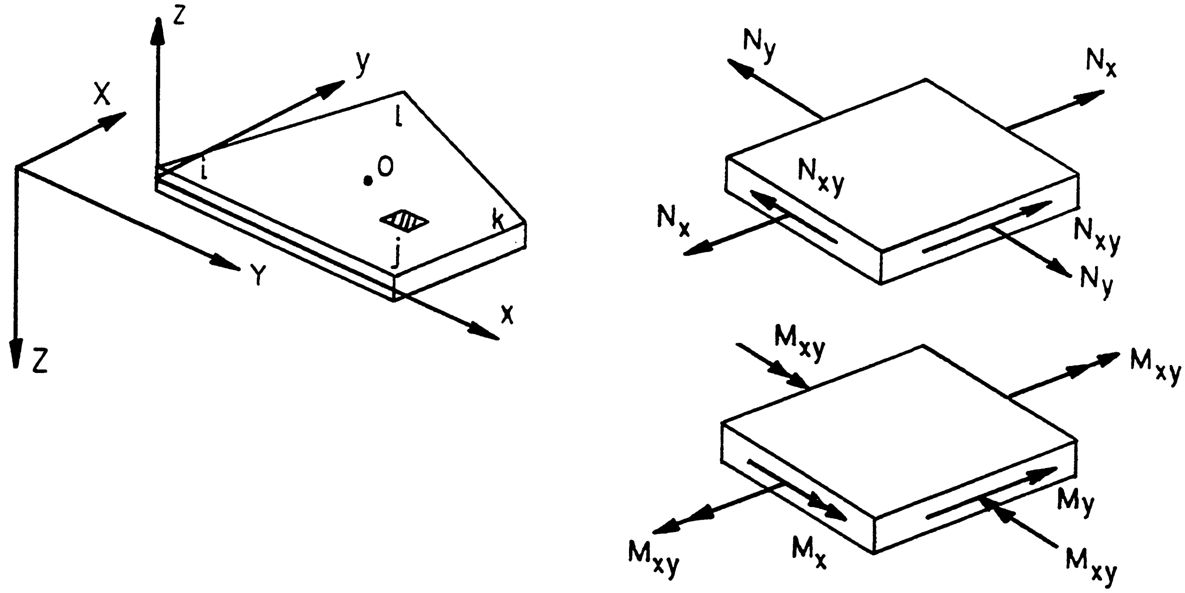


FIG. 4.8 INTERNAL STRESS RESULTANTS IN PLATE ELEMENTS

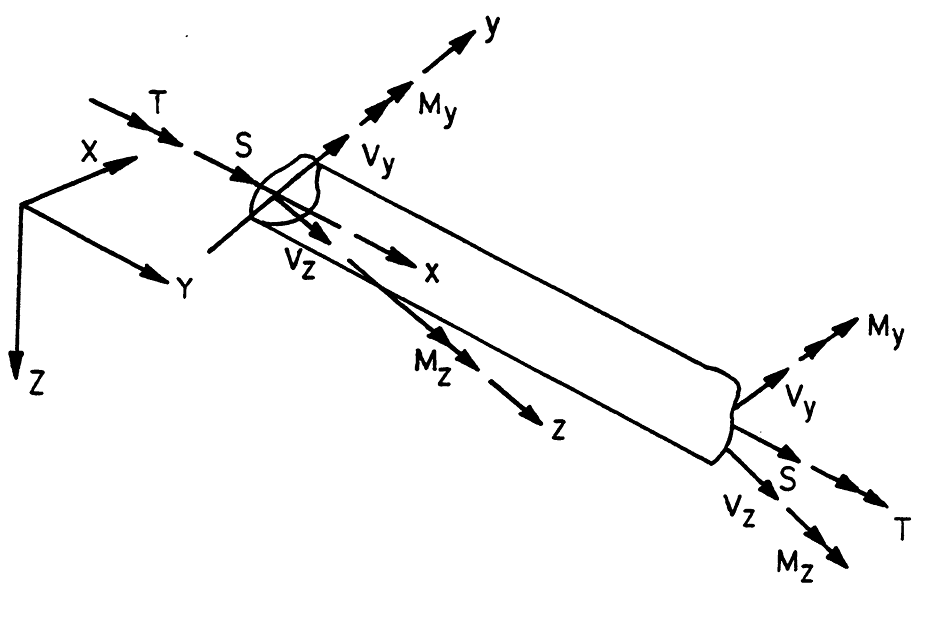


FIG. 4.9 END FORCES IN FRAME ELEMENTS

each individual girder as well as for the cross section as a whole, at any desired section of the structure. Use of this option will on the one hand provide valuable static checks, on the other hand, the user will obtain a picture of the structure's load distributing characteristics. The sum of all axial girder forces should be very small unless axial loads are applied, and the sum of all girder moments should equal the statical moment due to the external loads.

10) Finally, for each successfully executed problem, the computer execution time is printed out, broken up into the times required for

   a) Input setup and calculation of element stiffnesses
   b) Formation of all load vectors
   c) Formation of structure stiffness
   d) Solution of equations
   e) Calculation of all internal forces, reactions, residuals, and generation of all output

The output information items 3) through 9) are printed by load cases, i.e., the complete output for load case 1 is printed, then load case 2, etc.

A special program option allows the termination of execution after a check and echo printout of the input data. In this case, output information items 1) and 2)a, b, c will be printed. Numerous error printouts help in locating input errors.

Concluding, it should be remembered that the finite element theory on which the program FINPLA2 is based, is of approximate nature. Although it is true that a mesh refinement for the structure idealization will theoretically always increase the accuracy of the results, largely increased execution times might pose economical limits, and round-off within the computer might falsify the results beyond permissible bounds. Studies of numerous cases have indicated that the accuracy of solution decreases when highly skewed parallelogram elements are used. Thus it is suggested that in laying out a finite element mesh, an attempt should be made to use quadrilateral elements which approximate rectangles over as much of the structure as possible. This program permits the analysis of highly complex structures. Not only will it be difficult or even impossible to check the results by other analytical methods, but engineers might lack sufficient experience from that accumulated with more common structure types. For this reason, it is indispensable that the computer solution be thoroughly studied and interpreted by applying engineering judgement.

4.5 Storage Requirement and Execution Time Estimate

The storage requirement for program FINPLA2 and its subroutines, not counting blank common and area reserved for the systems program, amounts to approximately $71,300_8 \approx 29,400_{10}$, in the UNIVAC 1108 of the LOGICAL Computation Center, Los Angeles. The actual amount of assigned blank common area is in principle optional, although it is recommended to use as much of the available core storage as possible because this will reduce the number of equation blocks to be handled, therefore cutting down peripheral processor time. However, the amount of blank common area actually assigned shall not be less than the lower limit given below.

$$\text{Min. COMMON} = NC_1 + MAX(NC_2, NC_3, NC_4, NC_5, NC_6)$$

where

$$NC_1 = 5*NEL + 24*NSECT + (NFORST + NSAV) * (1 + NSECT + NPTS) + NSAV * NPTS * NSCK$$

$$NC_2 = NSECT * (27 + 4 * NEL) + NDIAPH + NTAD * (14 + NPTS) + 5*NPLD + 12*NFET + NEQ + 2 *NDBC + NEL + 3 * NSE + 2 * NCSTYP * NPTS + 42 * NS2PT$$

$$NC_3 = NSECT*(20 + 4 * NEL) + NEQS + 5 * (NESL + NCL) + 6 * NSSL$$

$$NC_4 = 2 * NQ * (MB + NLC) + NEQ + 2 * NDBC$$

$$NC_5 = NLC * (60 * NEL + NQ + 67)$$

$$NC_6 = NLC * (MAX(NQ,30) + 3 * NSCK * NGIR) + 180$$

and

NEL = number of elements in a structure segment

NSECT = number of sections

NPTS = number of primary nodal points in a section

NFET = number of finite element types

NSE = number of special finite elements

NDIAPH = number of diaphragms

NTAD = number of applied group displacements

NPLD = number of prescribed line displacements

NDBC = number of prescribed non-zero displacements

NCSTYP = number of different cross section types

NS2PT = number of sections with secondary nodal points

NLC = number of load cases

NESL = number of elements with surface loads

NSSL = number of elements with special surface loads

NCL = number of concentrated loads

NFORST = number of force transformation section types

NSAV = number of internal force averaging section types

NSCK = number of sections for which static checks are desired

NGIR = number of girders of structure

NQ = number of equations in one block

NEQ = number of equations for total structure

NEQS = number of blocks times number of equations per block

MB = bandwidth of structure stiffness

The quantity NQ will be calculated internally on the basis of how much blank common is actually available. For the above equation, a small number may be substituted for NQ, while NEQS should be at least equal to NEQ. However, it is not recommended to base the blank common assignment on the above equation but rather on the maximum available storage. If the minimum given above is not provided, the execution is automatically terminated with an appropriate error message.

For a complex program such as FINPLA2, it is almost impossible to make a generally valid statement regarding the expected execution times. The following list of the major factors influencing the execution time is therefore intended to permit only a qualitative time estimate.

1. Peripheral storage facilities of the computer: For large problems, tape write and read operations will constitute by far the most important single item. In fact, it has been observed, that up to 80% of the total running time may be spent on just input-output operations. Therefore it is essential

that the most efficient available hardware equipment be utilized for intended core storage, drums or disks.

2. Solution of linear equations: This item may require up to half of the central processing time. For in-core solutions, this time is approximately given by

$$T_1 = \alpha \cdot N M^2$$

where α = computer-dependent factor

N = number of non-zero equations

M = half bandwidth plus number of load vectors

For large equation systems, peripheral processing time due to equation block handling may increase this time considerably.

3. Output generation: The time required for this item is approximately given by

$$T_2 = \beta \cdot NL \cdot NEL$$

where β = computer-dependent factor

NL = number of load cases

NEL = number of finite elements for which stress output is requested.

Calculation of reactions and residuals may increase this time, but compared to stress output, this influence is generally small.

4. Calculation of element stiffnesses and formation of structure stiffness: This item is strictly dependent on the number of elements, but the time required for calculating element stiffnesses may be much reduced if many repetitive elements occur.

5. EXAMPLES

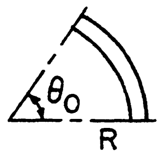
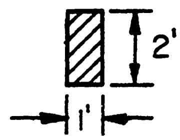
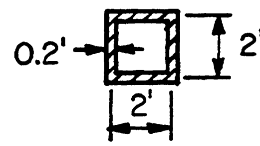
5.1 Curved Beam Problem

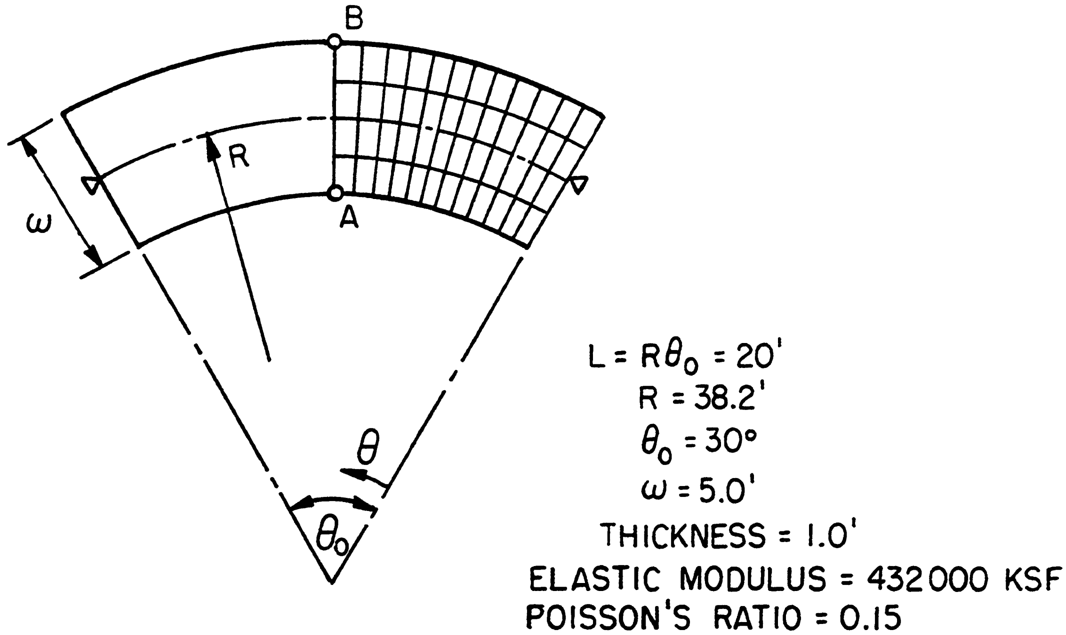
Analyzing a simply supported curved beam with torsional constraints at both ends, using FINPLA2 as well as ordinary curved beam theory [4], the comparison of Table 1 was obtained. As loading conditions, a uniformly distributed load and a concentrated midspan load were considered. The developed span length $L = R\theta_o = 20$ ft. was kept constant while the curvature was varied. In all cases the span was divided into only five 4 ft. long segments. The solid cross section was subdivided over the depth into two 1 ft. elements and for the box section, was subdivided transversely into eight 1 X 0.2 ft. elements. The results of Table 1 show satisfactory agreement, considering the small number of subdivisions employed both longitudinally and transversely in the finite element mesh.

5.2 Curved Plate Problem

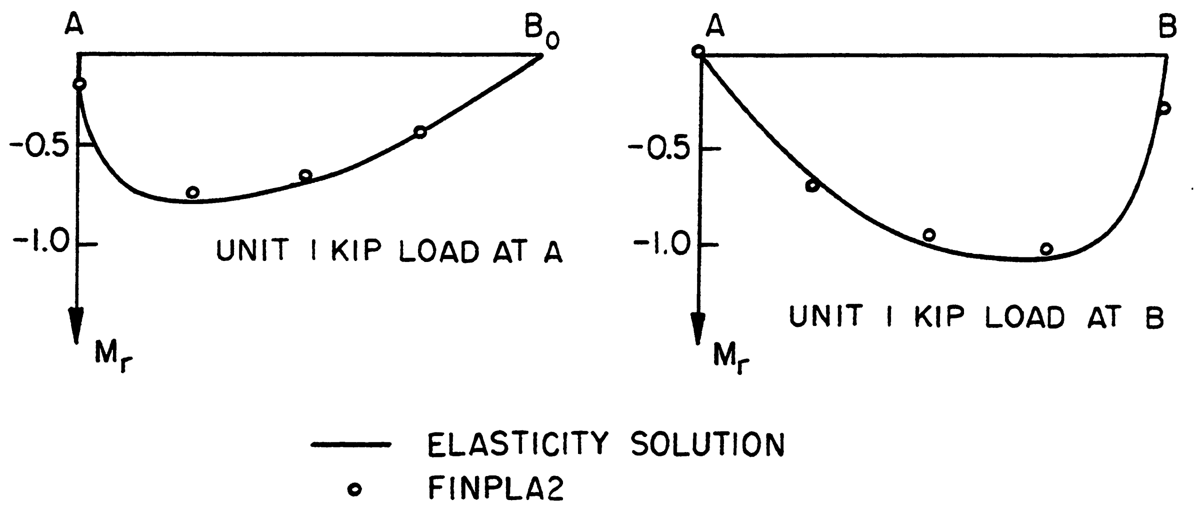
The curved plate shown in Fig. 5.1a is simply supported along the straight radial edges, while the curved edges are free. A concentrated load is placed at the midspan of either the inner or the outer edge. Figure 5.1b shows the resulting transverse plate bending moments, calculated according to curved plate theory [4], as well as results from FINPLA2, based on a mesh representation of 4 X 12 = 48 elements for half the plate. Longitudinal plate bending moments are compared in Fig. 5.2. The agreement between the finite element program and the elasticity solution, as demonstrated in both figures is excellent and fully satisfactory for practical purposes.

Table 1. Curved Beam Results

$R\theta_o$ = 20' = CONST.  E=432000 ksf, V=0.15			SECTION A 		SECTION B 	
Load Case	Output Quantity	Opening Angle θ_o	FINPLA2	Curved Beam	FINPLA2	Curved Beam
Uniform Load 1 kip/ft	Midspan Deflection (ft)	15°	.006860		.004868	
		22.50	.007127		.005075	
		30°	.007516		.005377	
	Midspan Moment (ft-k)	15°	46.75	50.36	48.79	50.36
		22.5°	48.66	50.82	49.76	50.82
		30°	51.47	51.47	51.17	51.47
Concentrated 1 kip Midspan Load	Midspan Deflection (ft)	15°	.0005547		.0003995	
		22.5°	.0005759		.0004159	
		30°	.0006067		.0004397	
	Midspan Moment (ft-k)	15°	4.251	5.029	4.500	5.029
		22.5°	4.405	5.065	4.579	5.065
		30°	4.631	5.117	4.693	5.117
	Quarterspan Moment (ft-k)	15°	2.392	2.516	2.492	2.516
		22.5°	2.508	2.541	2.548	2.541
		30°	2.679	2.581	2.631	2.581



a) PLATE DIMENSIONS AT MESH REPRESENTATION



b) TRANSVERSE BENDING MOMENTS (FT-K/FT) AT MIDSPAN

FIG. 5.1 CURVED PLATE PROBLEM

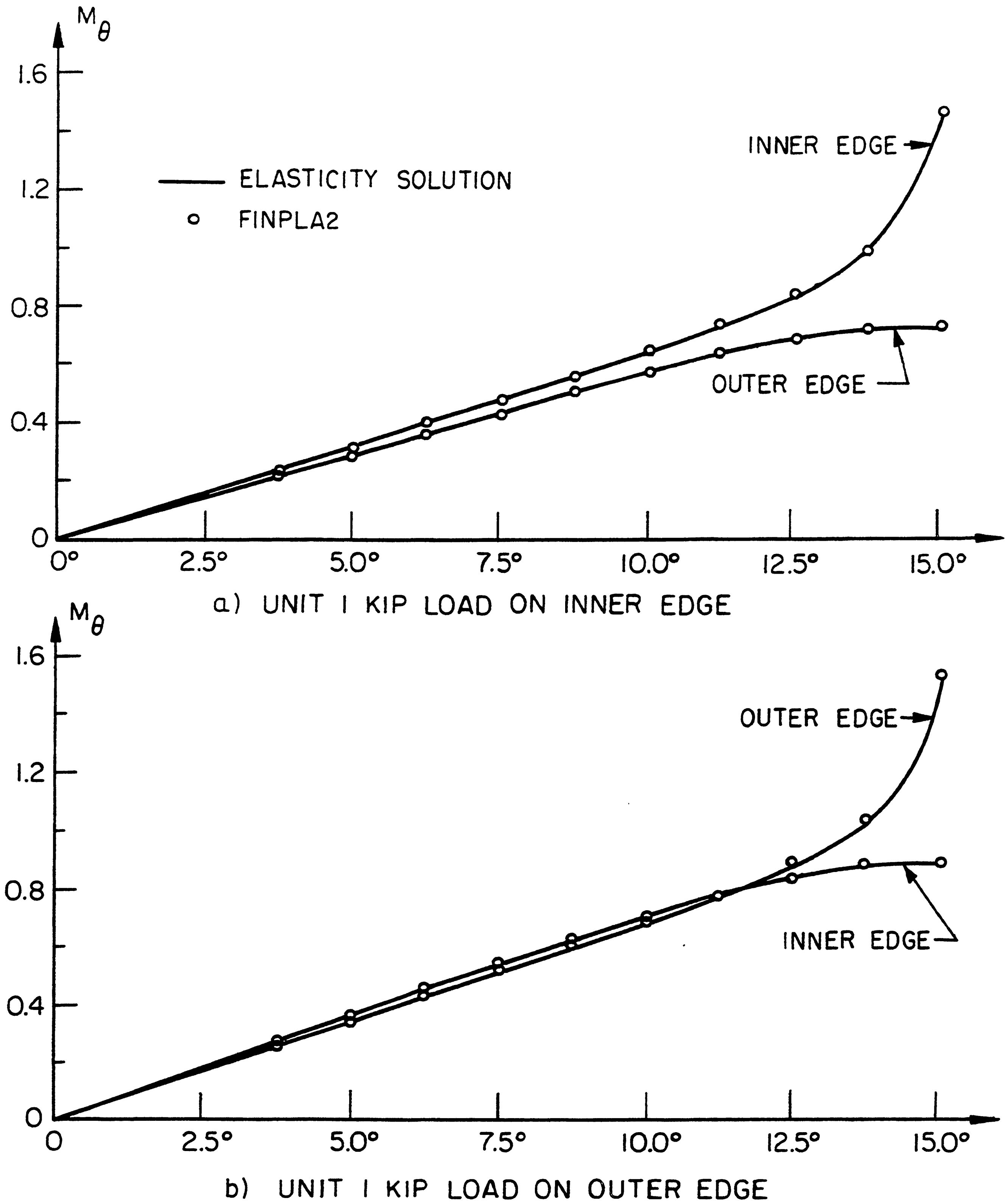


FIG. 5.2 LONGITUDINAL BENDING MOMENTS (FT-KIP/FT) IN CURVED PLATE

5.3 Simply Supported Curved Box Girder Bridge

The box girder bridge of Fig. 5.3 has a radius of curvature of 250 feet and is simply supported at both ends. A standard AASHO truck was placed either in position A or B. Figures 5.4 and 5.5 show some results for both truck positions, obtained by program FINPLA2, and, for comparison, also the results obtained from a program called CURSTR, which is based on curved folded plate theory [4]. As can be seen in these figures, agreement for midspan transverse bending moments is excellent, while longitudinal stress resultants may differ as much as 10%. But considering the coarse mesh layout utilized, and the fact that the larger stress discrepancies exist only in the webs which have very high stress gradients, the general agreement between the two theories can be considered satisfactory for practical purposes.

5.4 Tapered Continuous Box Girder Bridge

This example was selected in order to demonstrate the versatility of the program. The two-span bridge is depicted in Fig. 5.6. The first span of 60 feet is straight, but the superelevation of 12% used in the curved second span is run off in the first span, resulting in a warped deck and bottom plate. With the inclined webs also being warped, the center web remains the only plate in this span with a developable surface. The second span has a constant curvature which is different for the deck edges and the bridge centerline so that the bridge width increases towards the end of the second span.

The finite element representation for this structure is shown in Fig. 5.7. The horizontal member of the center frame support is treated as part of the diaphragm, because of its variable depth. Therefore

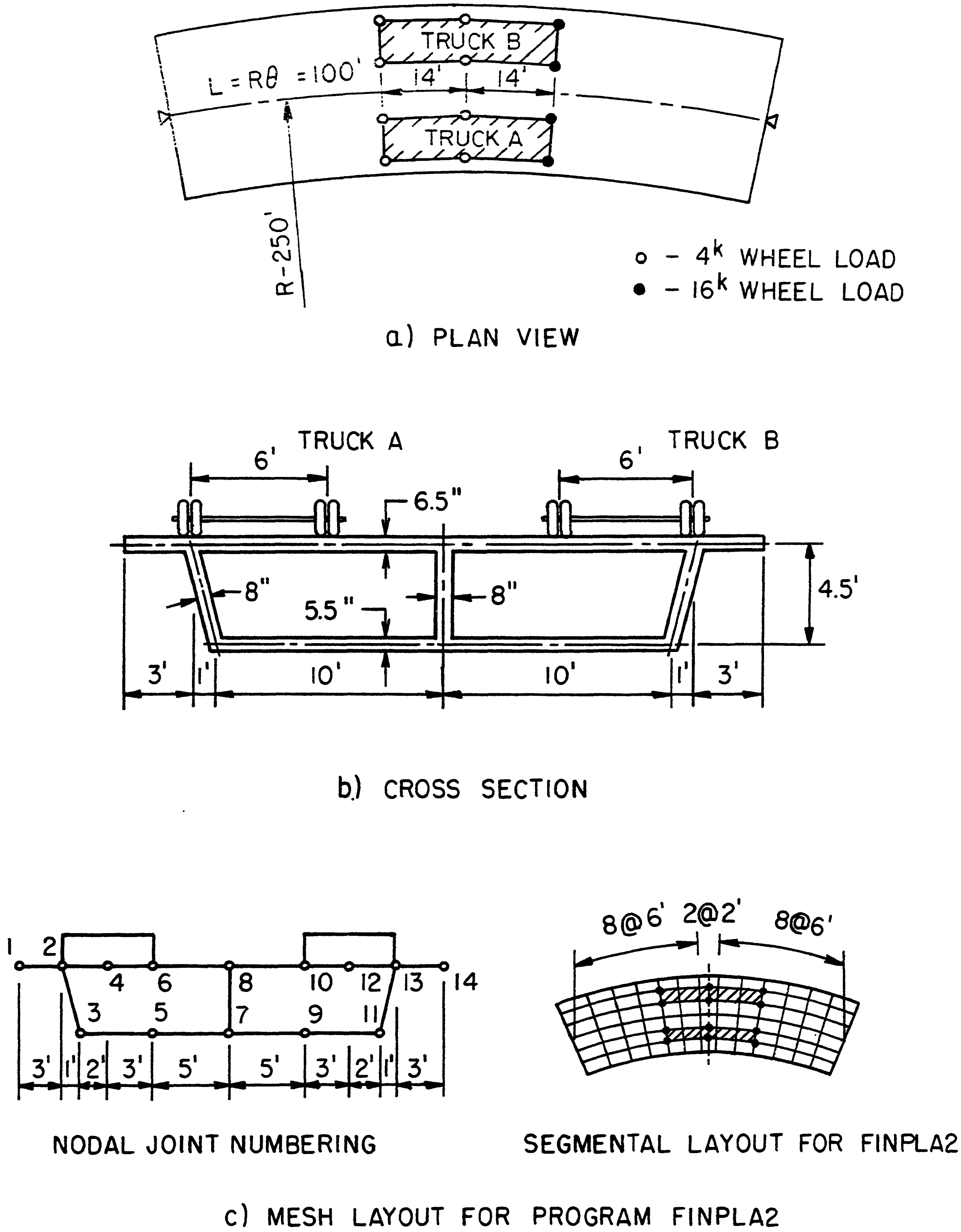


FIG. 5.3 CURVED BOX GIRDER BRIDGE

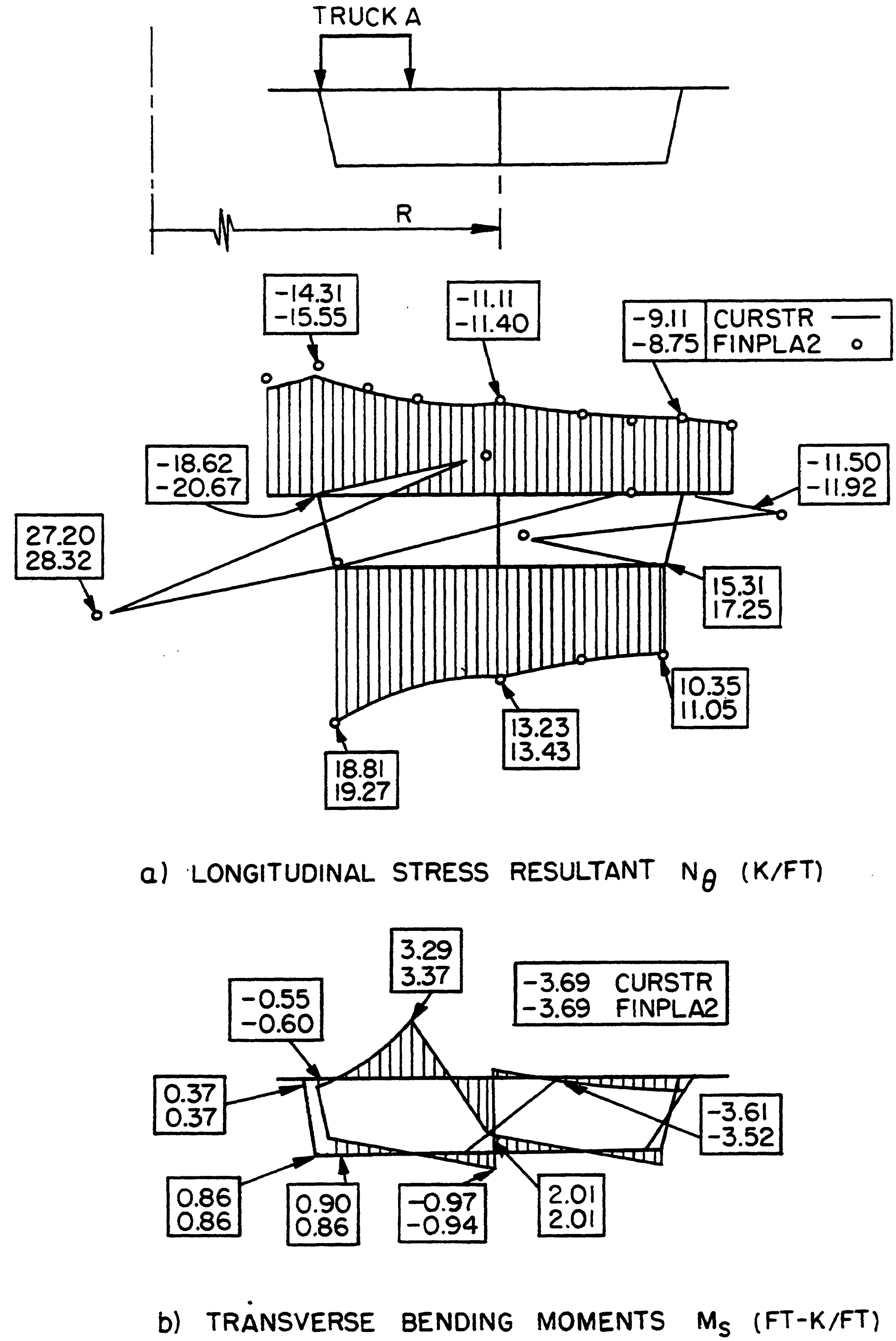
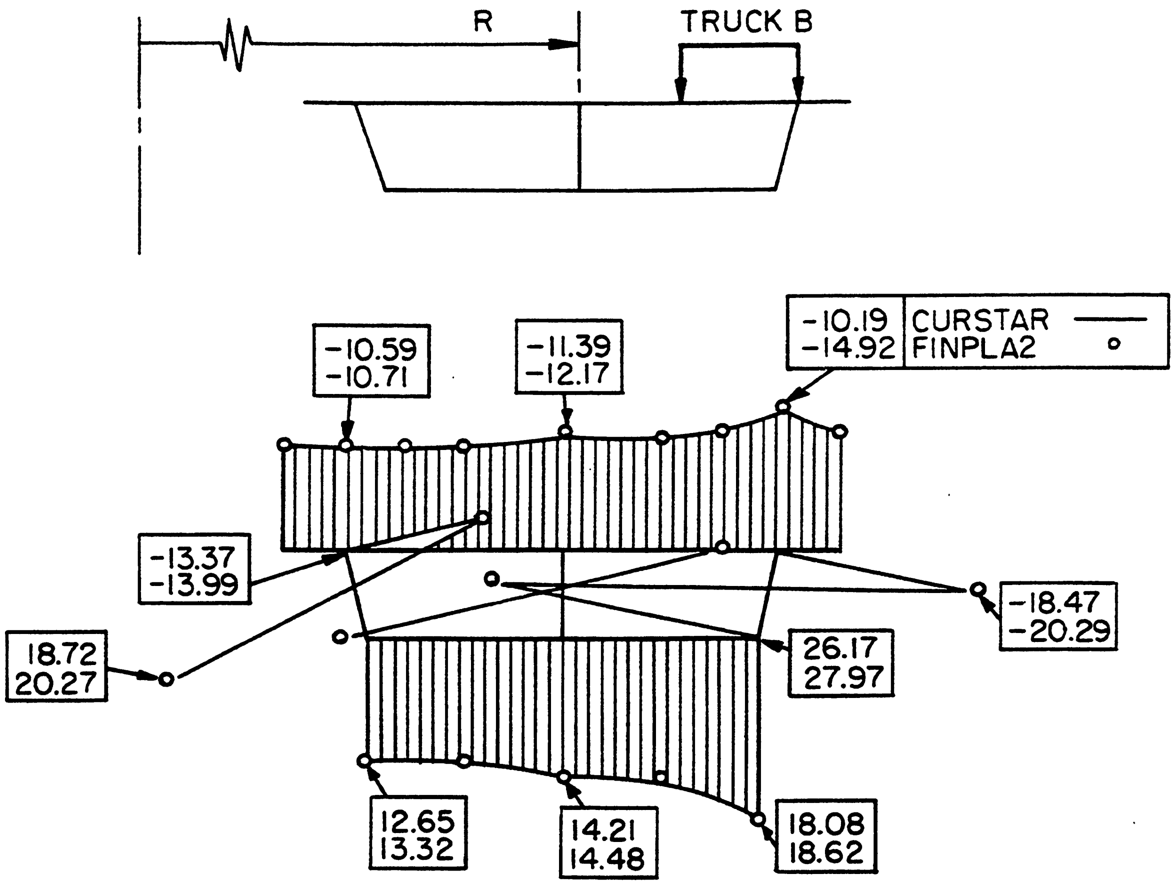
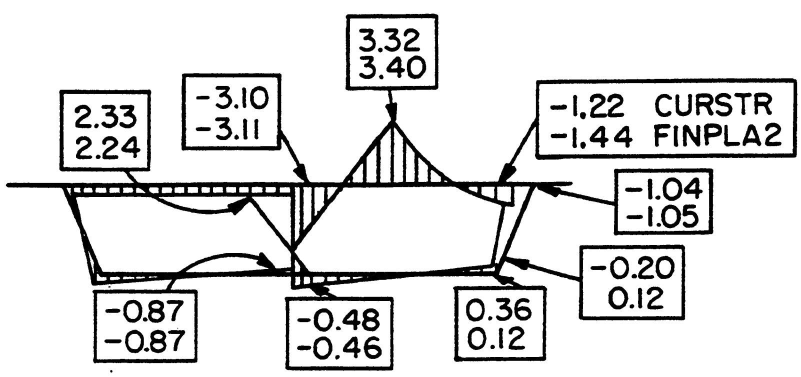


FIG. 5.4 MIDSPAN STRESSES (KIPS/FT) AND MOMENTS (FT-KIP/FT) IN CURVED BOX GIRDER BRIDGE DUE TO TRUCK A



a) LONGITUDINAL STRESS RESULTANTS N_{θ} (K/FT)



b) TRANSVERSE BENDING MOMENTS M_S (FT-K/FT)

FIG. 5.5 MIDSPAN STRESSES (KIP/FT) AND MOMENTS (FT-KIP/FT) IN CURVED BOX GIRDER BRIDGE DUE TO TRUCK B

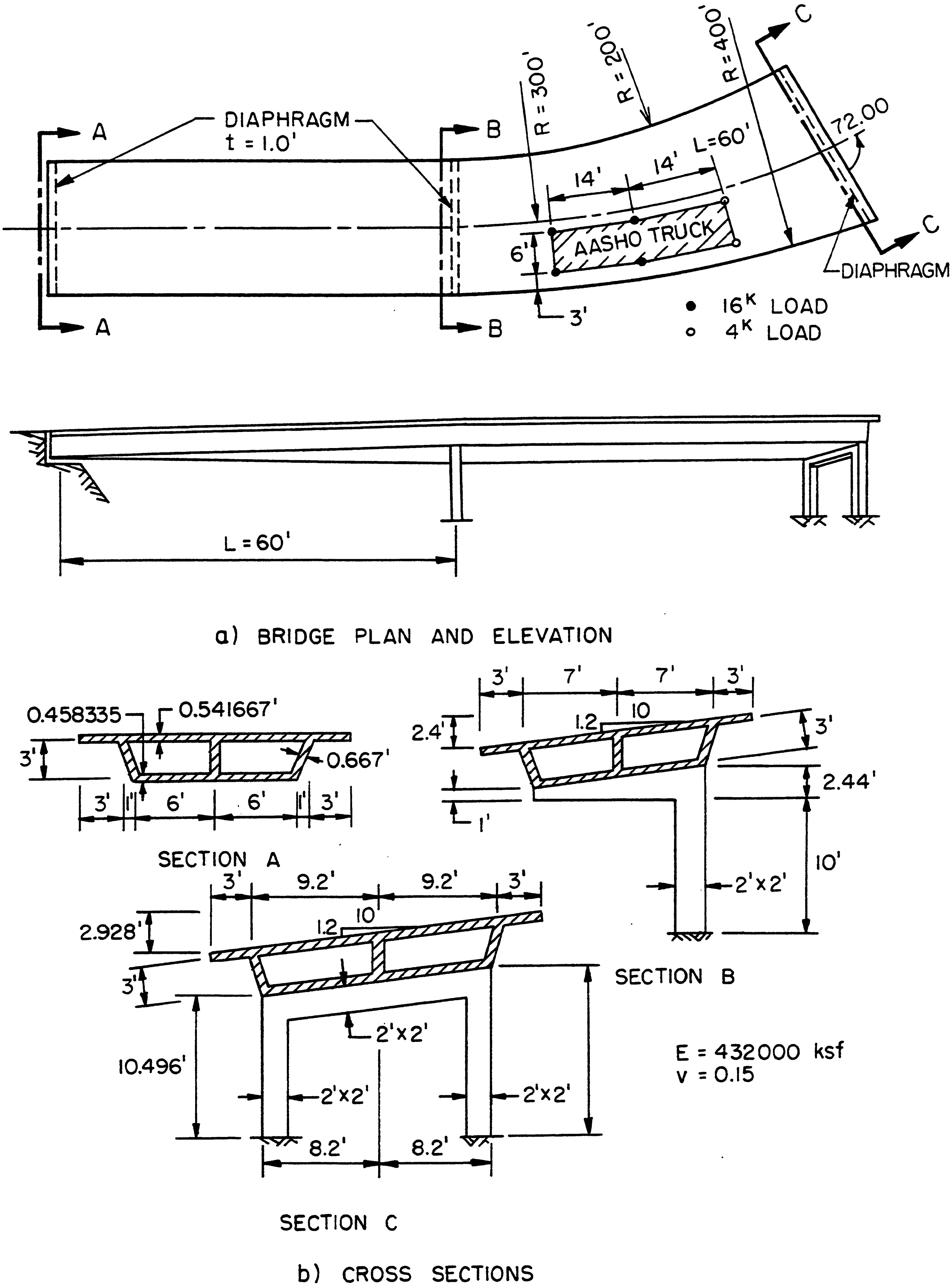
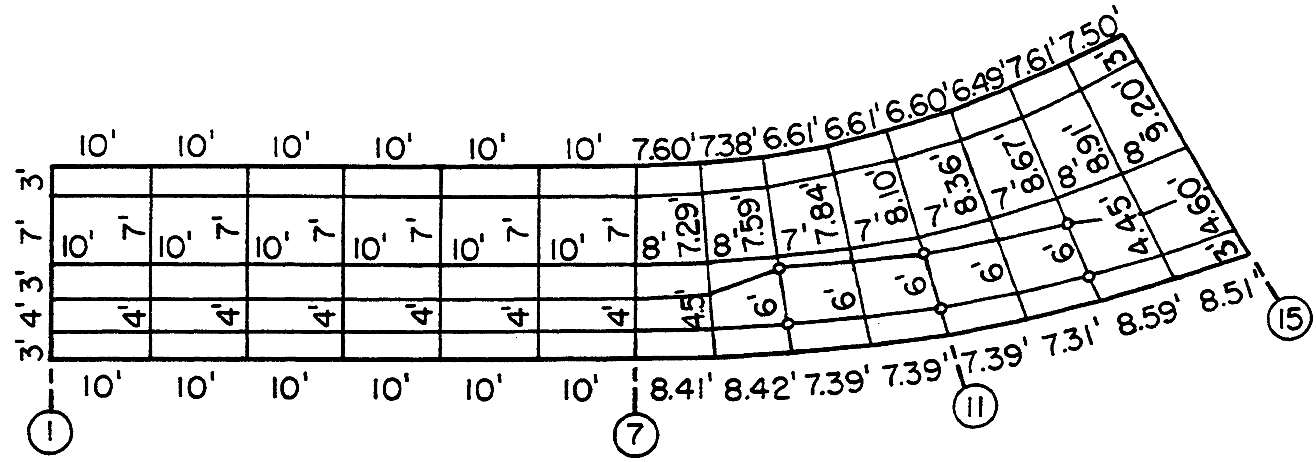
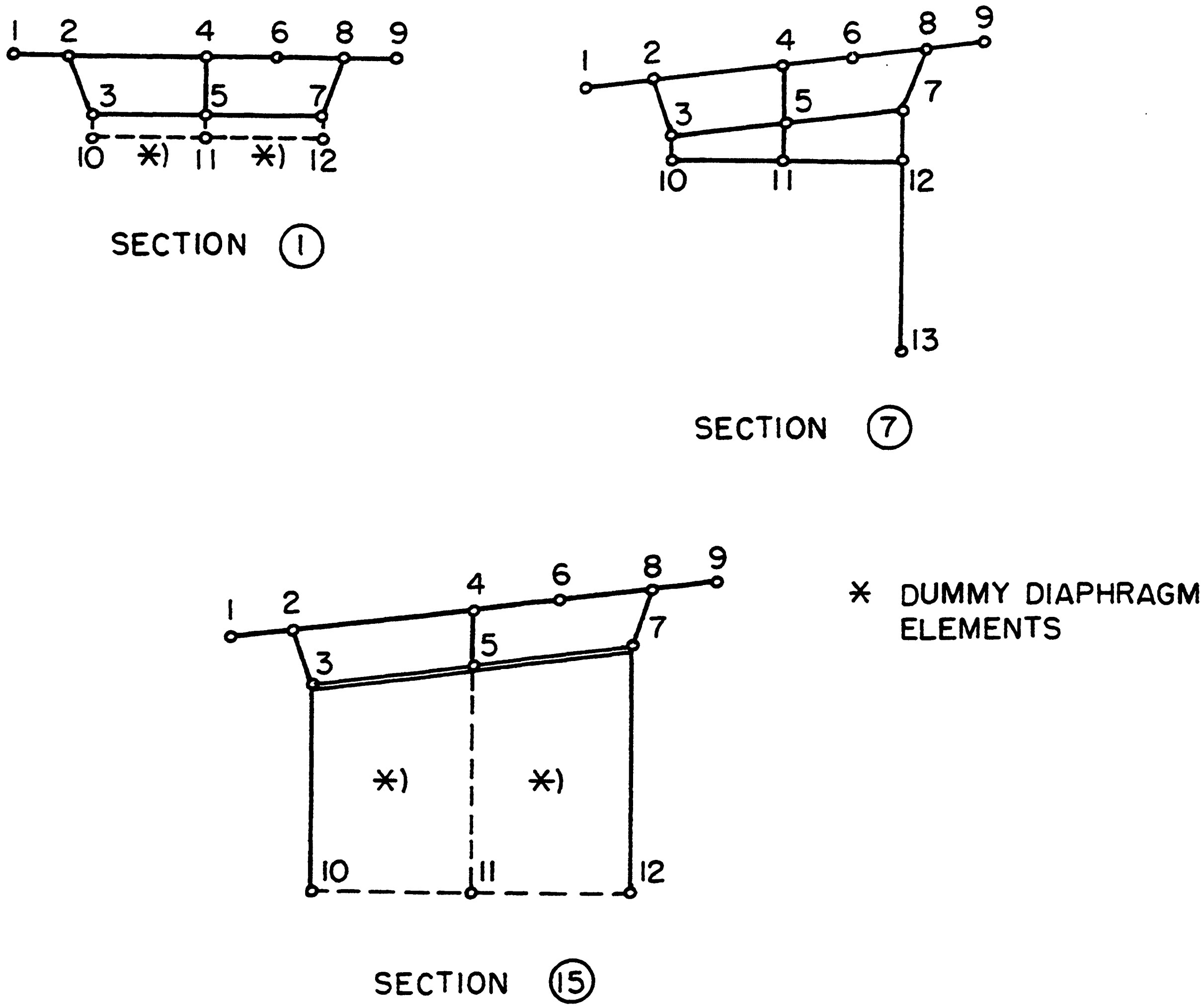


FIG. 5.6 TAPERED CONTINUOUS BOX GIRDER BRIDGE



a) PLAN VIEW



b) NODAL POINT NUMBERING

FIG. 5.7 FINITE ELEMENT MESH REPRESENTATION OF TAPERED CONTINUOUS BOX GIRDER BRIDGE

dummy elements have to be introduced for both end diaphragms as indicated, so that the number of diaphragm elements is the same at all three support sections.

Some output results for this example are summarized in Fig. 5.8. The distribution of the longitudinal stress component N_x at the center support and at the center of the curved span are shown in Fig. 5.8a. Although these distributions do look reasonable, no alternate method of analysis is readily available to verify these results. Even a statics check proves to be fairly difficult. In Fig. 5.8b, all reactions are indicated as they were output by the program. If the bridge is "bent" straight, these reactions may be summed as indicated in Fig. 5.8c, recognizing the approximations introduced by not properly transforming the reactions to the new coordinate system. From these reactions, approximate gross moments may be determined, Fig. 5.8d. In this figure, also the gross moments found by integrating internal stresses as described earlier, are indicated in parentheses. Considering all the approximations of this checking procedure and the coarseness of the finite element representation, the comparison of Fig. 5.8d appears to be acceptable. A finer finite element mesh and a more refined static check should improve the comparison.

The analysis of this example required the solution of 870 equations with a bandwidth of 96, and 148 plate type finite elements and 5 frame elements were involved. The execution times for the various program segments on the UNIVAC 1108 (including all peripheral processing time) were as follows:

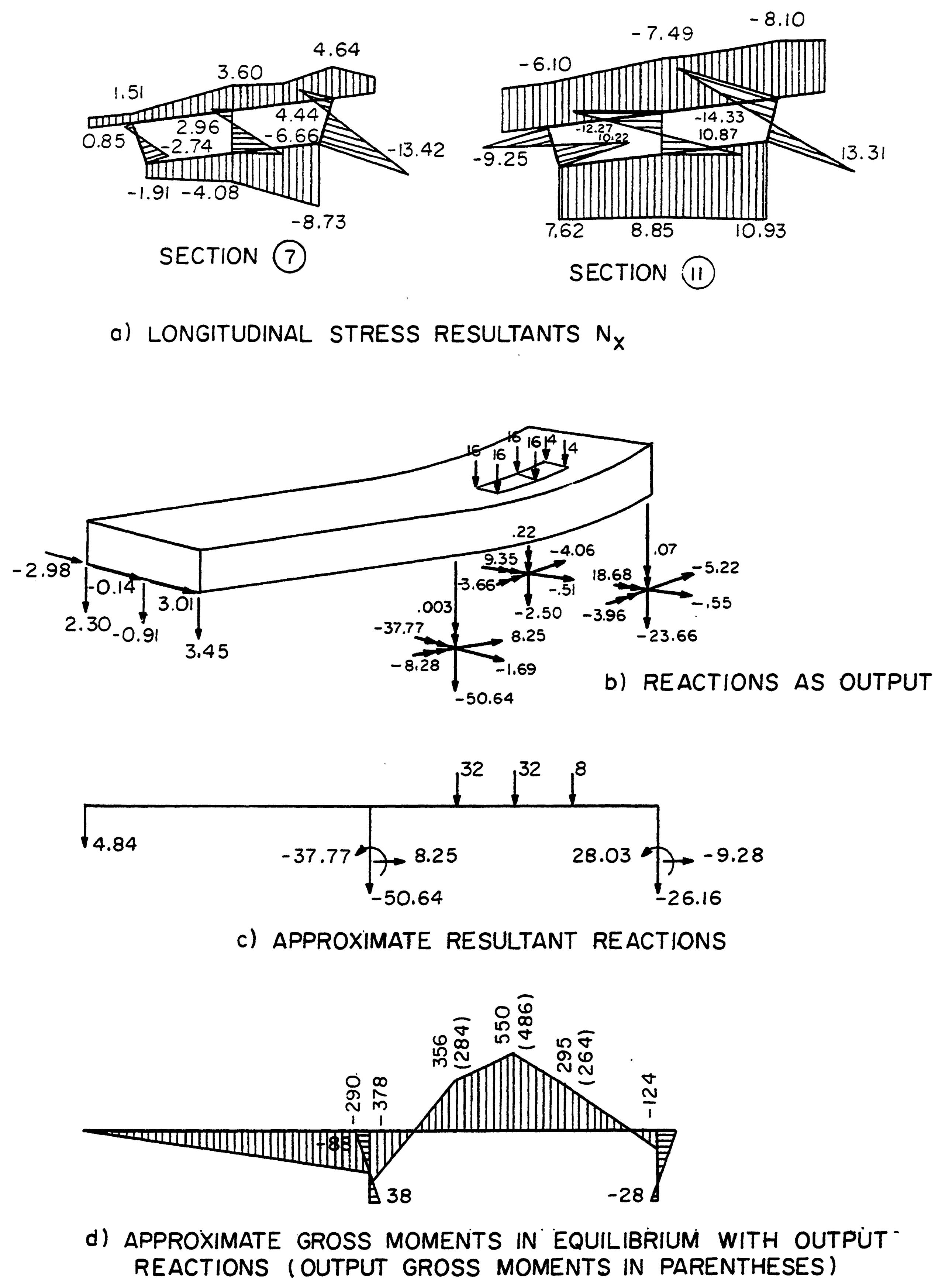


FIG. 5.8 RESULTS FOR TAPERED CONTINUOUS BOX GIRDER BRIDGE

Input setup (including calculation of element stiffnesses)	- 61.8 secs.
Formation of load vector	- .4 secs.
Formation of structure stiffness	- 57.6 secs.
Solution of equations	- 95.9 secs.
Output generation	- 47.9 secs.
Total execution time	- 263.6 secs.

6. CONCLUSIONS AND RECOMMENDATIONS FOR IMPLEMENTATION

A computer program, FINPLA2, has been presented which can be utilized to analyze general highway bridge structures. This program can analyse general non-prismatic cellular structures of varying width and depth and may have an integrated three dimensional frame. The structure is discretised by dividing it longitudinally into a certain number of structure segments by vertical sections, and by subdividing each such segment into finite elements. The structure alignment is described by a longitudinal reference line which may be a straight line, a circular curve or an arbitrary planar string polygon and cross-sections are defined with respect to this line. Orthotropic plate properties and arbitrary loadings and boundary conditions can be treated. Automatic element and coordinate generation options minimise the required input data.

A FORTRAN IV source listing is given in Appendix C for those wishing to implement the program directly onto their available computer. Information on the availability of source decks as well as a programming information supplement may be obtained from the authors. It is suggested that the input data given in Appendix B for Example 5 be used as a check case when implementing the program. Finally, it would be appreciated if any inconsistencies or errors are found in the program that they be brought to the attention of the authors.

7. ACKNOWLEDGEMENTS

This investigation was sponsored by the Division of Highways, Department of Public Works, State of California, and the Bureau of Public Roads, Federal Highway Administration, United States Department of Transportation. The opinions, findings, and conclusions expressed in this report are those of the authors and not necessarily those of the Bureau of Public Roads.

Mr. G. D. Mancarti, Assistant Bridge Engineer, and Mr. R. E. Davis, Senior Bridge Engineer, of the Research and Development Section, provided close liaison from the Bridge Department, Division of Highways, State of California.

Special thanks are due to Mr. A. F. Kabir, Research Assistant, who assisted in the preparation of the tables and drawings for the report.

The support of the Computer Center at the University of California, Berkeley, is gratefully acknowledged for providing its facilities.

8. REFERENCES

1. Scordelis, A. C., "Analysis of Simply Supported Box Girder Bridges," Structural Engineering and Structural Mechanics Report No. 66-17, University of California, Berkeley, October 1966 (PB 175 646).

2. Scordelis, A. C., "Analysis of Continuous Box Girder Bridges," Structural Engineering and Structural Mechanics Report No. 67-25, University of California, Berkeley, November 1967 (PB 178 355).

3. Scordelis, A. C., and C. Meyer, "Wheel Load Distribution in Concrete Box Girder Bridges," Structural Engineering and Structural Mechanics Report No. 69-1, University of California, Berkeley, January 1969 (PB 183 923).

4. Willam, K. J., and A. C. Scordelis, "Analysis of Orthotropic Folded Plates with Eccentric Stiffeners," Structural Engineering and Structural Mechanics Report No. 70-2, University of California, Berkeley, February 1970 (PB 191 051).

5. Meyer, C., and A. C. Scordelis, "Computer Program for Prismatic Folded Plates with Plate and Beam Elements," Structural Engineering and Structural Mechanics Report No. 70-3, February 1970 (PB 191 050).

6. Meyer, C., and A. C. Scordelis, "Analysis of Curved Folded Plate Structures," Structural Engineering and Structural Mechanics Report No. 70-8, University of California, Berkeley, June, 1970 (PB 193 535).

7. Willam, K. J., and A. C. Scordelis, "Computer Program for Cellular Structures of Arbitrary Plan Geometry," Structural Engineering and Structural Mechanics Report No. 70-10, University of California, Berkeley, September 1970 (PB 196 143).

8. Meyer, C., "Analysis and Design of Curved Box Girder Bridges," Structural Engineering and Structural Mechanics Report No. 70-22, University of California, Berkeley, December 1970 (PB 197 289).

9. Bouwkamp, J. G., Scordelis, A. C., and Wasti, S. T., "Structural Behavior of a Two Span Reinforced Concrete Box Girder Bridge Model," UCSESM Report No. 71-5, University of California, Berkeley, April, 1971 (PB 199 187).

10. Kristek, V., "Tapered Box Girders of Deformable Cross Section," Journal of the Structural Division, ASCE, Vol. 96, No. ST8, August, 1970, pp. 1761-1793.

11. Abu Ghazaleh, B. N., "Analysis of Plate Type Prismatic Structures," Ph.D. Dissertation, University of California, Berkeley, California, 1966.

12. Willam, K. J., "Finite Element Analysis of Cellular Structures," Ph.D. Dissertation, University of California, Berkeley, California, Dec. 1969.

13. Felippa, C. A., "Refined Finite Element Analysis of Linear and Nonlinear Two-Dimensional Structures," Ph.D. Dissertation, University of California, Berkeley, California, 1966.

14. Wilson, E. L., "SAP, A General Structural Analysis Program," Structural Engineering and Structural Mechanics Report No. SESM 70-20, University of California, Berkeley, California, Sept. 1970.

Copies of References 1-9 have been placed on file with the U.S. Department of Commerce and may be obtained on request at $3.00 per copy by writing to the following address:

National Technical Information Service
Operations Division
Springfield, Virginia 22151

The accession number (shown in parenthesis on the attached list) should be used when ordering a particular report.

APPENDIX A

Program Input Specifications

```
C                        INPUT SPECIFICATIONS
C                        ====================
C
C         1) START CARD (A5)
C            ----------
C     COL.  1 TO  5 - THE WORD 'START'
C
C         2) TITLE CARD (16A5)
C            ----------
C     COL. 1 TO 80 - TITLE OF PROBLEM TO BE PRINTED WITH OUTPUT, TITLE(I)
C
C         3) CONTROL CARD (16I4,7I1,F9,0)
C            ------------
C     COL.  1 TO  4 - NUMBER OF STRUCTURE SEGMENTS, NSEG
C     COL.  5 TO  8 - NUMBER OF PLATE ELEMENTS IN A SEGMENT, NEL
C     COL.  9 TO 12 - NUMBER OF PRIMARY NODAL POINTS IN A SECTION, NPTS
C     COL. 13 TO 16 - NUMBER OF SECTIONS WITH SECONDARY NODAL POINTS,NS2PT
C     COL. 17 TO 20 - NUMBER OF FINITE ELEMENT PLATE TYPES, NFET
C     COL. 21 TO 24 - NUMBER OF SPECIAL PLATE ELEMENTS, NSE
C     COL. 25 TO 28 - NUMBER OF DIAPHRAGMS, NDIAPH
C     COL. 29 TO 32 - NUMBER OF FRAME ELEMENTS, NFRAME, MAX,=100
C     COL. 33 TO 36 - NUMBER OF APPLIED GROUP DISPLACEMENTS, NTAD
C     COL. 37 TO 40 - NUMBER OF PRESCRIBED LINE DISPLACEMENTS, NPLD
C     COL. 41 TO 44 - NUMBER OF DIFFERENT CROSS SECTION TYPES, NCSTYP
C     COL. 45 TO 48 - NUMBER OF LOAD CASES FOR THIS PROBLEM, NLC
C     COL. 49 TO 52 - NUMBER OF FORCE TRANSFORMATION SECTION TYPES, NFORST
C     COL. 53 TO 56 - NUMBER OF INTERNAL FORCE AVERAGING SECTN, TYPES,NSAV
C     COL. 57 TO 60 - NUMBER OF SECTIONS FOR WHICH STATIC CHECKS ARE
C                     DESIRED, NSCK
C     COL. 61 TO 64 - NUMBER OF GIRDERS OF STRUCTURE (ONLY FOR NSCK,GT,0),
C                     NGIR
C     COL.       65 - ALIGNMENT INDICATOR, IA
C                        1 - ARBITRARY POLYGONAL REFERENCE LINE
C                        2 - CIRCULARLY CURVED REFERENCE LINE
C                        3 - STRAIGHT REFERENCE LINE
C     COL.       66 - SKEWNESS INDICATOR, IS
C                        1 - ALL SECTIONS NORMAL TO REFERENCE LINE TANGENT
C                        2 - ARBITRARY ANGLES BETWEEN REFERENCE LINE AND
C                            SECTIONS (IA=2 OR 3), OR BETWEEN GLOBAL Y-AXIS
C                            AND SECTIONS (IA=1)
C     COL.       67 - MATERIAL OPTION INDICATOR, IM
C                        1 - INPUT ISOTROPIC E, FNU, AND THICKNESS FOR EACH
C                            PLATE TYPE
C                        2 - INPUT COMPLETE ANISOTROPIC MATERIAL LAW FOR
C                            EACH PLATE TYPE
C     COL.       68 - REFERENCE LINE INPUT OPTION, IR (FOR IA=2 ONLY)
C                        1 - POINTS ON REFERENCE LINE DEFINED BY ANGLES (IN
C                            DEGREES)
C                        2 - POINTS ON REFERENCE LINE DEFINED BY ARC LENGTH
C                            S=R*THETA
C     COL.       69 - LOAD-DISPLACEMENT INPUT/OUTPUT OPTION, IL
C                        1 - LOADS AND DISPLACEMENTS MEASURED IN FIXED
C                            GLOBAL COORDINATES
C                        2 - LOADS AND DISPLACEMENTS MEASURED IN TRAVELLING
C                            GLOBAL COORDINATES
C     COL.       70 - INPUT CHECK TERMINATOR, ISTOP
C                        0 - EXECUTE COMPLETE PROBLEM
C                        1 - SUPPRESS PROBLEM EXECUTION AFTER PRINTOUT OF
C                            INPUT DATA
C     COL.       71 - CHECK OPTION INDICATOR, ICHK
```

```
C                      0 - NO PRINTOUT OF GENERATED INFORMATION DESIRED
C                      1 - AT LEAST ONE CHECK PRINTOUT DESIRED
C     COL. 72 TO 80 - CURVATURE RADIUS OF REFERENCE LINE,R (FOR IA=2 ONLY)
C                     ENTER NEGATIVE VALUE IF CENTER OF CURVATURE HAS
C                     NEGATIVE GLOBAL Y-COORDINATE
C
C        4) CHECK OPTION CARD (6I4)
C           -----------------
C     NOT REQUIRED IF ICHK=0
C
C     COL.  1 TO  4 - OPTION TO PRINT GENERATED REFERENCE POINT DATA, IC1
C     COL.  5 TO  8 - OPTION TO PRINT GENERATED NODAL POINT COORDS., IC2
C     COL.  9 TO 12 - OPTION TO PRINT BOUNDARY CONDITION ARRAYS, IC3
C     COL. 13 TO 16 - OPTION TO PRINT LOAD VECTORS, IC4
C     COL. 17 TO 20 - OPTION TO PRINT ELEMENT STIFFNESSES, IC5
C     COL. 21 TO 24 - OPTION TO PRINT STRUCTURE STIFFNESS, IC6
C                      1 - PRINTOUT DESIRED
C                      0 - PRINTOUT NOT DESIRED
C
C        5) REFERENCE LINE COORDINATE POINTS
C           -------------------------------
C     NOTE - EACH SECTION PLANE IS FIXED IN SPACE BY A POINT ON THE REFER-
C       ENCE LINE TO BE DEFINED BELOW AND AN ANGLE BETWEEN REFERENCE LINE
C       TANGENT OR GLOBAL X-AXIS AND SECTION.
C
C     EACH REFERENCE POINT REQUIRES ONE CARD (3F10.0)
C     COL.  1 TO 10 - GLOBAL X-COORDINATE (IA=1 OR 3), OR ARC LENGTH CO-
C                     ORDINATE (IA=2 AND IR=2), OR THETA ANGLE IN DEGREES
C                     IA=2 AND IR=1), X(I)
C     COL. 11 TO 20 - GLOBAL Y-COORDINATE (FOR IA=1 ONLY), Y(I)
C     COL. 21 TO 30 - ANGLE (IN DEGREES) BETWEEN GLOBAL Y-AXIS AND SECTION
C                     (IA=1 AND IS=2) OR BETWEEN REFER. LINE TANGENT AND
C                     SECTION (IA=2 OR 3 AND IS=2), G(I), (FOR IS.NE.1)
C                     USE RIGHT-HAND RULE FOR POSITIVE ANGLE DIRECTION.
C
C        6) NODAL POINT COORDINATE CARDS
C           ---------------------------
C     NOTE - EACH SECTION TYPE IS SPECIFIED BY DEFINING THE COORDINATES OF
C       ALL PRIMARY NODAL POINTS.
C
C     ONE CARD (I10,2F10.0) FOR EACH PRIMARY POINT IN FIRST SECTION TYPE.
C     COL.  1 TO 10 - NODAL POINT NUMBER, J
C     COL. 11 TO 20 - Y-COORDINATE, YY(I,J) (IN TRAVELLING GLOBAL FRAME)
C     COL. 21 TO 30 - Z-COORDINATE, ZZ(I,J) (IN TRAVELLING GLOBAL FRAME)
C
C     REPEAT THE CORRESPONDING DECK OF CARDS FOR THE SECOND SECTION TYPE,
C     THEN FOR THE THIRD, ETC.
C
C        7) SECONDARY NODAL POINT CARDS
C           --------------------------
C     NOTE - ONLY SECTIONS WITH SECONDARY NODAL POINTS REQUIRE ONE SET OF
C       CARDS AS FOLLOWS.
C
C     FIRST CARD (2I5)
C     COL.  1 TO  5 - SECTION NUMBER, I2SEC(I)
C     COL.  6 TO 10 - NUMBER OF SECONDARY NODAL POINTS IN THIS SECTION,
C                     I2PT(I), MAX.=10
C
C     ONE CARD (I10,3F10.0) FOR EACH SECONDARY POINT IN SECTION I2SEC(I).
C     COL.  1 TO 10 - NODAL POINT NUMBER, N2PT(J,K) (MUST BE.GT.NPTS)
```

```
C   COL. 11 TO 20 - X-COORDINATE IN TRAVELLING GLOBAL FRAME, SX(J,K)
C   COL. 21 TO 30 - Y-COORDINATE IN TRAVELLING GLOBAL FRAME, SY(J,K)
C   COL. 31 TO 40 - Z-COORDINATE IN TRAVELLING GLOBAL FRAME, SZ(J,K)
C
C        8) CROSS SECTION TYPE CARD (25I3)
C           ----------------------
C   COL.  1 TO 75 - TYPE NUMBERS FOR ALL STRUCTURE SECTIONS, STARTING
C                   AT THE ORIGIN AND PROCEEDING ALONG THE REFERENCE
C                   LINE, NSCT(I), USE MORE CARDS IF NECESSARY,
C
C        9) FINITE ELEMENT PLATE TYPE CARDS
C           -------------------------
C   IF MATERIAL OPTION IM=1, THEN EACH FINITE ELEMENT TYPE REQUIRES ONE
C   CARD (I10,3F10.0)
C   COL.  1 TO 10 - TYPE NUMBER, I
C   COL. 11 TO 20 - THICKNESS, TH(I)
C   COL. 21 TO 30 - MODULUS OF ELASTICITY, E(I)
C   COL. 31 TO 40 - POISSON-S RATIO, FNU(I)
C
C   IF MATERIAL OPTION IM=2, THEN EACH FINITE ELEMENT TYPE REQUIRES TWO
C   CARDS (I10,6F10.0/10X,6F10.0)
C
C   FIRST CARD - IN-PLANE CONSTANTS OF ANISOTROPIC MATERIAL LAW
C   COL.  1 TO 10 - TYPE NUMBER, I
C   COL. 11 TO 20 - C(1,1)
C   COL. 21 TO 30 - C(2,2)
C   COL. 31 TO 40 - C(3,3)
C   COL. 41 TO 50 - C(1,2)
C   COL. 51 TO 60 - C(1,3)
C   COL. 61 TO 70 - C(2,3)
C   SECOND CARD - PLATE BENDING CONSTANTS OF ANISOTROPIC MATERIAL LAW
C   COL. 11 TO 20 - D(1,1)
C   COL. 21 TO 30 - D(2,2)
C   COL. 31 TO 40 - D(3,3)
C   COL. 41 TO 50 - D(1,2)
C   COL. 51 TO 60 - D(1,3)
C   COL. 61 TO 70 - D(2,3)
C
C       10) PLATE ELEMENT CARDS
C           ------------------
C   NOTE - EACH PLATE TYPE FINITE ELEMENT IN A REGULAR STRUCTURE SEGMENT
C     REQUIRES ONE CARD (6I4,F10.0)
C   COL.  1 TO  4 - ELEMENT NUMBER, I
C   COL.  5 TO  8 - NODAL POINT I, NPI(I)
C   COL.  9 TO 12 - NODAL POINT J, NPJ(I)
C   COL. 13 TO 16 - FINITE ELEMENT TYPE NUMBER, KPL(I)
C   COL. 17 TO 20 - NUMBER OF GIRDER TO WHICH NODE I BELONGS, NGE(1,I)
C   COL. 21 TO 24 - NUMBER OF GIRDER TO WHICH NODE J BELONGS, NGE(2,I)
C   COL. 25 TO 34 - HORIZONTAL DISTANCE FROM NODE I TO DIVIDING LINE,
C                   IF ELEMENT BELONGS TO TWO GIRDERS, XDIV(I)
C                   THE LAST THREE ENTRIES ARE NEEDED ONLY IF STATIC
C                   CHECKS ARE DESIRED.
C
C       11) SPECIAL PLATE TYPE FINITE ELEMENT CARDS
C           ---------------------------------------
C   NOTE - EACH SPECIAL ELEMENT WHOSE ELEMENT TYPE IS DIFFERENT THAN
C     THAT OF THE CORRESPONDING ELEMENT IN A REGULAR STRUCTURE SEGMENT,
C     REQUIRES ONE CARD (4I4), NO CARDS REQUIRED IF NO SUCH ELEMENTS.
C   COL.  1 TO  4 - SPECIAL PLATE ELEMENT NUMBER, I
C   COL.  5 TO  8 - NUMBER OF CORRESP.ELEMENT IN REGULAR SEGMENT,ISEL(I)
```

```
C     COL.  9 TO 12 - SEGMENT NUMBER, ISES(I)
C     COL. 13 TO 16 - TYPE NUMBER FOR THIS SPECIAL ELEMENT, ISET(I)
C
C          12) DIAPHRAGM DECK
C              --------------
C     NO CARDS REQUIRED IF NO TRANSVERSE DIAPHRAGMS
C
C     FIRST CARD (2I5)
C     COL.  1 TO  5 - NUMBER OF ELEMENTS IN A REGULAR DIAPHRAGM,NDE,MAX=20
C     COL.  6 TO 10 - NUMBER OF SPECIAL DIAPHRAGM ELEMENTS, NDS, MAX=20
C
C     SECOND CARD (25I3)
C     COL.  1 TO 75 - NUMBERS OF SECTIONS WITH DIAPHRAGMS, NSD(I), IN
C                     CONSECUTIVE ORDER
C
C     DIAPHRAGM ELEMENT CARDS (6I4) - ONE CARD FOR EACH ELEMENT IN A
C     REGULAR DIAPHRAGM,
C     COL.  1 TO  4 - DIAPHRAGM ELEMENT NUMBER, I
C     COL.  5 TO  8 - FINITE ELEMENT TYPE NUMBER, NDTY(I)
C     COL.  9 TO 12 - NODAL POINT I, NPD(1,I)
C     COL. 13 TO 16 - NODAL POINT J, NPD(2,I)
C     COL. 17 TO 20 - NODAL POINT K, NPD(3,I)
C     COL. 21 TO 24 - NODAL POINT L, NPD(4,I)
C                     NODAL POINTS I,J,K,L MUST BE NUMBERED EITHER
C                     CLOCKWISE OR COUNTER-CLOCKWISE,
C
C     SPECIAL DIAPHRAGM ELEMENT CARDS (4I4) - ONE CARD FOR EACH SPECIAL
C     ELEMENT WHOSE TYPE IS DIFFERENT THAN THAT OF THE CORRESPONDING ELE-
C     MENT IN A REGULAR DIAPHRAGM, NO CARDS REQUIRED IF NO SUCH ELEMENTS,
C     COL.  1 TO  4 - SPECIAL DIAPHRAGM ELEMENT NUMBER, I
C     COL.  5 TO  8 - NUMBER OF CORRESP,ELE, IN REGULAR DIAPHRAGM, NDSE(I)
C     COL.  9 TO 12 - DIAPHRAGM NUMBER, NDSD(I)
C     COL. 13 TO 16 - TYPE NUMBER FOR THIS SPECIAL ELEMENT, NDST(I)
C
C          13) FRAME DECK
C              ----------
C     NO CARDS REQUIRED IF NO FRAME ELEMENTS,
C
C     FIRST CARD (2I5)
C     COL.  1 TO  5 - NUMBER OF FRAME ELEMENT SECTION TYPES, NFST, MAX=20
C     COL.  6 TO 10 - NUMBER OF FRAME ELEMENT CONNECTION TYPES,NFCT,MAX=20
C
C     FRAME ELEMENT SECTION TYPE CARDS (I10,6F10.0) - ONE CARD FOR EACH
C     TYPE
C     COL.  1 TO 10 - FRAME ELEMENT SECTION TYPE NUMBER, I
C     COL. 11 TO 20 - EA, AXIAL RIGIDITY, EA(I)
C     COL. 21 TO 30 - EI-Y, BENDING RIGIDITY ABOUT ELEMENT Y-AXIS, EIY(I)
C     COL. 31 TO 40 - EI-Z, BENDING RIGIDITY ABOUT ELEMENT Z-AXIS, EIZ(I)
C     COL. 41 TO 50 - GJ-X, TORSIONAL RIGIDITY, GJ(I)
C     COL. 51 TO 60 - GA-Y, SHEAR RIGIDITY EFFECTIVE IN Y-DIRECTION,GAY(I)
C     COL. 61 TO 70 - GA-Z, SHEAR RIGIDITY EFFECTIVE IN Z-DIRECTION,GAZ(I)
C
C     FRAME ELEMENT CONNECTION TYPE CARDS (I4,9F8.0) - ONE CARD FOR EACH
C     TYPE
C     COL.  1 TO  4 - FRAME ELEMENT CONNECTION TYPE NUMBER, I
C     COL.  5 TO 12 - X-ECCENTRICITY OF NODE I-CONNECTION, EI(1,I)
C     COL. 13 TO 20 - Y-ECCENTRICITY OF NODE I-CONNECTION, EI(2,I)
C     COL. 21 TO 28 - Z-ECCENTRICITY OF NODE I-CONNECTION, EI(3,I)
C     COL. 29 TO 36 - X-ECCENTRICITY OF NODE J-CONNECTION, EJ(1,I)
C     COL. 37 TO 44 - Y-ECCENTRICITY OF NODE J-CONNECTION, EJ(2,I)
```

```
C     COL. 45 TO 52 - Z-ECCENTRICITY OF NODE J-CONNECTION, EJ(3,I)
C     COL. 53 TO 60 - X-COORDINATE OF AUXILIARY POINT K, EK(1,I)
C     COL. 61 TO 68 - Y-COORDINATE OF AUXILIARY POINT K, EK(2,I)
C     COL. 69 TO 76 - Z-COORDINATE OF AUXILIARY POINT K, EK(3,I)
C                     NOTE - NODE I AND J ECCENTRICITIES ARE MEASURED IN
C                     TRAVELLING COORDINATES OF THE SECTION TO WHICH THEY
C                     BELONG, COORDINATES OF AUXILIARY POINT K ARE TAKEN
C                     IN TRAVELLING COORDINATES OF SECTION WITH NODE I.
C                     AUXILIARY POINT K LIES IN XZ-PLANE OF LOCAL FRAME
C                     ELEMENT AXES, ECCENTRICITIES OF NODE I AND J CONNEC-
C                     TIONS SHOULD BE SMALL COMPARED TO FRAME ELE, LENGTH,
C
C     FRAME ELEMENT CARDS (7I4) - ONE CARD FOR EACH ELEMENT
C     COL.  1 TO  4 - FRAME ELEMENT NUMBER, I
C     COL.  5 TO  8 - NODAL JOINT NUMBER OF END POINT I, NFI(I)
C     COL.  9 TO 12 - NODAL JOINT NUMBER OF END POINT J, NFJ(I)
C     COL. 13 TO 16 - SECTION NUMBER OF END POINT I, NSFI(I)
C     COL. 17 TO 20 - SECTION NUMBER OF END POINT J, NSFJ(I)
C                     THE DIFFERENCE BETWEEN NSFI AND NSFJ SHOULD NOT
C                     EXCEED 1,
C     COL. 21 TO 24 - FRAME ELEMENT SECTION TYPE NUMBER, NFTY(I)
C     COL. 25 TO 28 - FRAME ELEMENT CONNECTION TYPE NUMBER, NFSC(I)
C
C         14) BOUNDARY CONDITION DECK
C             -----------------------
C     APPLIED GROUP DISPLACEMENT CARDS
C     NOTE - EACH DISPLACEMENT COMPONENT REQUIRES A SET OF TWO CARDS, IF
C       ALL PRIMARY AND SECONDARY NODAL POINTS ARE LIKEWISE AFFECTED, ONLY
C       ONE CARD, NO CARDS REQUIRED, IF NO APPLIED GROUP DISPLACEMENTS,
C
C     FIRST CARD (4I4,F10,0)
C     COL.  1 TO  4 - DISPLACEMENT COMPONENT NUMBER, I
C     COL.  5 TO  8 - COMPONENT INDICATOR, INDT(I), EQUAL TO
C                       1 - PRESCRIBED DISPLACEMENT IN THE X-DIRECTION
C                       2 - PRESCRIBED DISPLACEMENT IN THE Y-DIRECTION
C                       3 - PRESCRIBED DISPLACEMENT IN THE Z-DIRECTION
C                       4 - PRESCRIBED ROTATION ABOUT X-AXIS
C                       5 - PRESCRIBED ROTATION ABOUT Y-AXIS
C                       6 - PRESCRIBED ROTATION ABOUT Z-AXIS
C     COL.  9 TO 12 - NUMBER OF AFFECTED NODAL POINTS, NANP(I)
C                     IF THIS EQUALS THE TOTAL NUMBER OF NODAL POINTS,
C                     OMIT THE SECOND CARD.
C     COL. 13 TO 16 - SECTION NUMBER OF APPLIED GROUP DISPLACEMENT,NSAD(I)
C     COL. 17 TO 26 - DISPLACEMENT MAGNITUDE, DTIN(I)
C                     NOTE - SPECIFIED DISPLACEMENTS ARE MEASURED IN THE
C                     COORDINATE SYSTEM SPECIFIED ON THE CONTROL CARD BY
C                     THE OPTION INDICATOR IL,
C
C     SECOND CARD (25I3)
C     COL.  1 TO 75 - AFFECTED NODAL POINTS, NAD(I,J)
C                     USE SECOND CARD IF REQUIRED
C
C     PRESCRIBED LINE DISPLACEMENT CARDS (4I4,F10,0) - ONE CARD FOR EACH
C     PRESCRIBED LINE DISPLACEMENT COMPONENT, NO CARDS REQUIRED IF NO SUCH
C     DISPLACEMENTS,
C     COL.  1 TO  4 - NODAL JOINT NUMBER, NJPD(I)
C     COL.  5 TO  8 - NUMBER OF SECTION WHERE DISPLACEMENT STARTS, NSDS(I)
C     COL.  9 TO 12 - NUMBER OF SECTION WHERE DISPLACEMENT ENDS, NSDE(I)
C                     FOR A SINGLE NODE DISPLACEMENT, THE LAST TWO ENTRIES
C                     ARE IDENTICAL,
```

```
C     COL. 13 TO 16 - COMPONENT INDICATOR, INPD(I) (VALUES AS IN PRECEDING
C                     CARDS)
C     COL. 17 TO 26 - PRESCRIBED DISPLACEMENT MAGNITUDE, PDIS(I)
C
C         15) INTERNAL FORCE TRANSFORMATION CARDS
C             -----------------------------------
C     NOTE - EACH INTERNAL FORCE TRANSFORMATION SECTION TYPE REQUIRES THE
C       FOLLOWING CARDS, NO CARDS REQUIRED IF NFORST.LE.0
C
C     FIRST CARD (25I3)
C     COL.  1 TO  3 - NUMBER  OF SECTIONS FOR WHICH INTERNAL FORCES SHOULD
C                     BE TRANSFORMED AS SPECIFIED BELOW, NFORS(I)
C     COL,  4 TO 75 - SECTION NUMBERS BELONGING TO THIS FORCE TRANSFORM-
C                     ATION TYPE, NFORSN(I,J)
C
C     SECOND CARD (9F8.0)
C     COL.  1 TO 72 - ANGLES (IN DEGREES) FOR ALL PRIMARY NODAL POINTS
C                     ABOUT WHICH OUTPUT INTERNAL FORCES SHOULD BE
C                     ROTATED, STARTING AT POINT 1, 2, ETC, ALF(I,J),
C                     USE MORE CARDS IF NECESSARY,
C
C     REPEAT THE ABOVE CARDS FOR THE SECOND FORCE TRANSFORMATION SECTION
C     TYPE, ETC.
C
C         16) INTERNAL FORCE AVERAGING SPECIFICATIONS
C             -------------------------------------
C     NOTE - EACH FORCE AVERAGING SECTION TYPE REQUIRES THE FOLLOWING SET
C       OF CARDS, NO CARDS REQUIRED IF NO INTERNAL FORCE AVERAGING DESIRED
C
C     FIRST CARD (25I3)
C     COL.  1 TO  3 - NUMBER  OF SECTIONS FOR WHICH THE INTERNAL FORCES
C                     SHOULD BE AVERAGED AS SPECIFIED BELOW, NSA(I)
C     COL.  4 TO 75 - SECTION NUMBERS BELONGING TO THIS FORCE AVERAGING
C                     TYPE, NSAVS(I,J)
C
C     SECOND CARD (12I6)
C     COL.          1 - NX -FORCE TRANSV,AVERAGING INDEX OF NODAL POINT 1
C     COL.          2 - NY -FORCE TRANSV,AVERAGING INDEX OF NODAL POINT 1
C     COL.          3 - NXY-FORCE TRANSV,AVERAGING INDEX OF NODAL POINT 1
C     COL.          4 - MX -MOMENT TRANSV,AVERAGING INDEX OF NODAL POINT 1
C     COL.          5 - MY -MOMENT TRANSV,AVERAGING INDEX OF NODAL POINT 1
C     COL.          6 - MXY-MOMENT TRANSV,AVERAGING INDEX OF NODAL POINT 1
C     COL.          7 - NX -FORCE LONGIT,AVERAGING INDEX OF NODAL POINT 1
C     COL.          8 - NY -FORCE LONGIT,AVERAGING INDEX OF NODAL POINT 1
C     COL.          9 - NXY-FORCE LONGIT,AVERAGING INDEX OF NODAL POINT 1
C     COL.         10 - MX -MOMENT LONGIT,AVERAGING INDEX OF NODAL POINT 1
C     COL.         11 - MY -MOMENT LONGIT,AVERAGING INDEX OF NODAL POINT 1
C     COL.         12 - MXY-MOMENT LONGIT,AVERAGING INDEX OF NODAL POINT 1
C     COL. 13 TO 72 - AVERAGING INDICES FOR NODAL POINTS 2 THROUGH 6,
C                     NPAV(I,1) - NPAV(I,12)
C                     ADD AS MANY CARDS (12I6) AS NECESSARY, IF MORE
C                     POINTS, ALL PRIMARY POINTS HAVE TO BE INDEXED,
C                     DO NOT LEAVE ZERO VALUES BLANK,
C                     AVERAGING INDEX IS EQUAL TO
C                       1 - FORCES/MOMENTS SHOULD BE AVERAGED
C                       0 - FORCES/MOMENTS SHOULD NOT BE AVERAGED
C
C     REPEAT THE ABOVE SET OF CARDS FOR THE NEXT FORCE AVERAGING SECTION
C     TYPE, ETC,
C
```

```
C        17) STATIC CHECK DATA (25I3)
C            ----------------
C
C    COL.  1 TO 75 - NUMBERS OF SECTIONS FOR WHICH STATIC CHECKS ARE
C                    DESIRED, NCS(I), USE SECOND CARD IF NECESSARY,
C                    NO CARDS REQUIRED IF NO STATIC CHECKS DESIRED,
C
C        18) OUTPUT SPECIFICATIONS
C            ---------------------
C    FIRST CARD (2I5)
C    COL.  1 TO  5 - REACTION OUTPUT OPTION, IREACT
C                       1 - ALL REACTIONS DESIRED
C                       0 - NO REACTIONS ARE DESIRED
C    COL.  6 TO 10 - RESIDUAL LOAD OUTPUT OPTION, IRESID
C                       1 - ALL RESIDUAL LOADS ARE DESIRED
C                       0 - NO RESIDUAL LOADS ARE DESIRED
C
C    SECOND CARD (25I3)
C    COL.  1 TO 75 - FOR EACH SEGMENT, SPECIFY ONE OPTION INDICATOR AS
C                     FOLLOWS, IOSEG(I)
C                       1 - CALCULATE AND PRINT INTERNAL PLATE FORCES
C                       2 - CALCULATE PLATE FORCES, BUT DO NOT OUTPUT
C                       3 - DO NOT CALCULATE INTERNAL PLATE FORCES
C                     NOTE THAT INTERNAL PLATE FORCES MAY BE NEEDED FOR
C                     STRESS AVERAGING AND MOMENT STATIC CHECKS,
C
C    THIRD CARD (25I3)
C    COL.  1 TO 75 - FOR EACH DIAPHRAGM, SPECIFY ONE OPTION INDICATOR AS
C                    GIVEN ABOVE, IODIA(I), OMIT IF NO DIAPHRAGMS,
C
C        19) LOAD DECKS
C            ----------
C    EACH LOAD CASE REQUIRES THE FOLLOWING DECKS OF CARDS
C
C    FIRST CARD (3I4)
C    COL.  1 TO  4 - NUMBER OF ELEMENTS IN A REGULAR SEGMENT WITH SURFACE
C                    LOADS, NESL
C    COL.  5 TO  8 - NUMBER OF ELEMENTS WITH SPECIAL SURFACE LOADS, NSSL
C    COL.  9 TO 12 - NUMBER OF CONCENTRATED OR DISTRIBUTED LINE LOADS
C                    ALONG THE NODAL JOINTS, NCL
C                    FOR TRANSVERSE LINE LOADS USE TRIBUTARY CONCEPT,
C    NOTE - ALL INPUT LOADS ARE MEASURED IN THE COORDINATE SYSTEM SPECI-
C      FIED BY THE OPTION INDICATOR IL ON THE CONTROL CARD,
C
C    ELEMENT SURFACE LOAD CARDS (7X,I3,4F10.0) - ONE CARD FOR EACH ELE-
C    MENT IN SEGMENT WITH REGULAR SURFACE LOADS, NO CARDS REQUIRED IF NO
C    REGULAR SURFACE LOADS,
C    SURFACE LOADS,
C    COL.  8 TO 10 - FINITE ELEMENT NUMBER, NELSL(I)
C    COL. 11 TO 20 - DEAD LOAD (P/PLATE AREA), DL(I)
C    COL. 21 TO 30 - LOAD IN X-DIRECTION (P/AREA PROJ.ON YZ-PLANE),XL(I)
C    COL. 31 TO 40 - LOAD IN Y-DIRECTION (P/AREA PROJ.ON XZ-PLANE),YL(I)
C    COL. 41 TO 50 - LOAD IN Z-DIRECTION (P/AREA PROJ.ON XY-PLANE),ZL(I)
C                    NOTE - LOADS ARE MEASURED IN THE COORDINATE SYSTEM
C                    SPECIFIED ON CONTROL CARD BY OPTION INDICATOR IL,
C
C    SPECIAL SURFACE LOAD CARDS (2I5,4F10.0) - ONE CARD FOR EACH ELEMENT
C    WITH LOADS IN ADDITION TO REGULAR SURFACE LOADS, NO CARDS REQUIRED
C    IF NO SUCH SURFACE LOADS,
C    COL.  1 TO  5 - SEGMENT NUMBER, NBLSL(I)
```

```
C     COL.  6 TO 10 - FINITE ELEMENT NUMBER, NSLSL(I)
C     COL. 11 TO 20 - DEAD LOAD (P/PLATE AREA), SDL(I)
C     COL. 21 TO 30 - LOAD IN X-DIRECTION (P/AREA PROJ.ON YZ-PLANE),SXL(I)
C     COL. 31 TO 40 - LOAD IN Y-DIRECTION (P/AREA PROJ.ON XZ-PLANE),SYL(I)
C     COL. 41 TO 50 - LOAD IN Z-DIRECTION (P/AREA PROJ.ON XY-PLANE),SZL(I)
C
C     JOINT LOAD CARDS (4I4,F10.0) - ONE CARD FOR EACH CONCENTRATED OR
C     DISTRIBUTED JOINT LOAD. NO CARDS REQUIRED IF NO SUCH LOADS.
C     COL.  1 TO  4 - NODAL JOINT NUMBER, NJL(I)
C     COL.  5 TO  8 - NUMBER OF SECTION WHERE LOADING STARTS, NSS(I)
C     COL.  9 TO 12 - NUMBER OF SECTION WHERE LOADING ENDS, NSO(I)
C                     FOR A CONCENTRATED JOINT LOAD THE LAST TWO ENTRIES
C                     ARE IDENTICAL.
C     COL. 13 TO 16 - COMPONENT INDICATOR, NID(I), EQUAL TO
C                        1 - APPLIED LOAD IN X-DIRECTION
C                        2 - APPLIED LOAD IN Y-DIRECTION
C                        3 - APPLIED LOAD IN Z-DIRECTION
C                        4 - APPLIED MOMENT ABOUT X-AXIS
C                        5 - APPLIED MOMENT ABOUT Y-AXIS
C                        6 - APPLIED MOMENT ABOUT Z-AXIS
C     COL. 17 TO 26 - LOAD INTENSITY, FI(I)
C                     FOR CONCENTRATED LOADS ENTER TOTAL VALUE
C
C     ALL CARDS ARE TO BE REPEATED FOR THE NEXT PROBLEM, STARTING WITH THE
C     START CARD. FOLLOWING THE LAST PROBLEM, ONE CARD IS ADDED WITH THE
C     WORD 'STOP' PUNCHED IN COLUMN 1 TO 4.
C
C*************************************************************************
```

APPENDIX B

Example Input Data

```
START     EXAMPLF 11  (REPORT EXAMPLE 4)
 EXAMPLE  11  -  WARPED,TAPERED CONTINUOUS BOX GIRDER BRIDGE WITH FRAME SUPPORT
  14  10   9   3   6   0   3   5   8   6  15   1   0   1   4   31210201
  1   1
      0.0       0.0       0.0
      10.       0.0       0.0
      20.       0.0       0.0
      30.       0.0       0.0
      40.       0.0       0.0
      50.       0.0       0.0
      60.       0.0       0.0
      68.      -.108     -2.28
    75.99      -.429     -4.56
   82.977      -.876     -6.56
   89.949     -1.500     -8.55
   96.906     -2.280    -10.54
  103.743     -3.207    -12.53
  111.741     -4.497    -14.82
  119.601     -5.979    -17.10
        1      -10.       0.0
        2       -7.       0.0
        3       -6.       3.0
        4       0.0       0.0
        5       0.0       3.0
        6       3.0       0.0
        7       6.0       3.0
        8       7.0       0.0
        9       10.       0.0
        1      -10.       .20
        2       -7.       .14
        3       -6.      3.12
        4       0.0       0.0
        5       0.0       3.0
        6       3.0      -.06
        7       6.0      2.88
        8       7.0      -.14
        9       10.      -.20
        1      -10.       .40
        2       -7.       .28
        3       -6.      3.24
        4       0.0       0.0
        5       0.0       3.0
        6       3.0      -.12
        7       6.0      2.76
        8       7.0      -.28
        9       10.      -.40
        1      -10.       .60
        2       -7.       .42
        3       -6.      3.36
        4       0.0       0.0
        5       0.0       3.0
        6       3.0      -.18
        7       6.0      2.64
        8       7.0      -.42
        9       10.      -.60
        1      -10.       .80
        2       -7.       .56
        3       -6.      3.48
        4       0.0       0.0
        5       0.0       3.0
```

6	3.0	-,24
7	6.0	2,52
8	7.0	-,56
9	10.	-,80
1	-10.	1,00
2	-7,	,70
3	-6.	3,60
4	0.0	0,0
5	0.0	3,0
6	3.0	-,30
7	6.0	2,40
8	7.0	-,70
9	10.	-1,00
1	-10.	1,20
2	-7,	,84
3	-6,	3,72
4	0.0	0.0
5	0.0	3,0
6	3.0	-,36
7	6.0	2,28
8	7.0	-,84
9	10.	-1,20
1	-10.293	1,235
2	-7,293	,875
3	-6,293	3,755
4	0.0	0,0
5	0.0	3,0
6	3,646	-,437
7	6,293	2,245
8	7,293	-,875
9	10,293	-1,235
1	-10,586	1,270
2	-7,586	,910
3	-6,586	3,790
4	0.000	0,000
5	0.000	3,000
6	1,586	-,190
7	6,586	2,210
8	7,586	-,910
9	10.586	-1,270
1	-10.842	1,301
2	-7,842	,941
3	-6,842	3,821
4	0,000	0,000
5	0.000	3,000
6	1,842	-,221
7	6.842	2,179
8	7,842	-,941
9	10,842	-1,301
1	-11,100	1,332
2	-8,100	,972
3	-7,100	3,852
4	0,000	0,000
5	0,000	3,000
6	2.100	-,252
7	7,100	2,148
8	8,100	-,972
9	11,100	-1,332
1	-11,356	1,363
2	-8,356	1,003

```
          3    -7.356      3,883
          4     0.000      0,000
          5     0.000      3,000
          6     2.356      -,283
          7     7.356      2,117
          8     8.356     -1,003
          9    11.356     -1,363
          1   -11.613      1,394
          2    -8,613      1,034
          3    -7.613      3,914
          4     0.000      0,000
          5     0,000      3,000
          6     2,613      -,314
          7     7,613      2,086
          8     8,613     -1,034
          9    11,613     -1,394
          1   -11,907      1,419
          2    -8,907      1,059
          3    -7,907      3,939
          4     0.000      0,000
          5     0.000      3,000
          6     4,453      -,534
          7     7,907      2,061
          8     8,907     -1,059
          9    11,907     -1,419
          1   -12.200      1,464
          2    -9.200      1,104
          3    -8.200      3,984
          4     0.000      0,000
          5     0.000      3,000
          6     4,600      -,552
          7     8,200      2,016
          8     9,200     -1,104
          9    12.200     -1,464
   1      3
         10          0.0       -6.0          4,0
         11          0.0        0.0          4,0
         12          0.0        6.0          4,0
   7      4
         10          0.0       -6.0          4,72
         11          0.0        0.0          4,72
         12          0.0        6,0          4,72
         13          0.0        6,0         14,72
  15      3
         10          0.0       -8,2         14,72
         11          0.0        0.0         14,72
         12          0.0        8.2         14,72
 1   2   3   4   5   6   7    8   9  10  11  12  13  14  15
          1    ,541667     432000,           ,15
          2    ,666667     432000,           ,15
          3    ,458333     432000,           ,15
          4   1,0          432000,           ,15
          5    ,0
          6   2,0          432000,           ,15
  1    1    2    1    1
  2    2    4    1    1    2        3,5
  3    4    6    1    2
  4    6    8    1    3
  5    8    9    1    3
  6    2    3    2    1
```

```
  7    4    5    2    2
  8    8    7    2    3
  9    3    5    3    1    2         3.0
 10    5    7    3    2    3         3.0
   4      2
 1   7  15
  1    4    2    3    5    4
  2    4    4    5    7    8
  3    5    3   10   11    5
  4    5    5   11   12    7
  1    3    2    6
  2    4    2    6
   1      2
          1  1728000.    576000.    576000.     42000.     75000.     75000.
  1       0.0   -.667    2.669
  2       0.0     0.0      0.0    1.229      0.0      0.0    1.229
  1   12   13    7    7    1    1
  2    3    5   15   15    1    2
  3    5    7   15   15    1    2
  4    3   10   15   15    1    1
  5    7   12   15   15    1    1
  1    2    3    1
 3   5   7
  2    3    3    1
 3   5   7
  3    1    2   15
10 12
  4    2    2   15
10 12
  5    3    2   15
10 12
  6    4    2   15
10 12
  7    5    2   15
10 12
  8    6    2   15
10 12
 13    7    7    1
 13    7    7    2
 13    7    7    3
 13    7    7    4
 13    7    7    5
 13    7    7    6
 3   9  11  13
000000111111000000111111001010111111000000111111000000111111111111111111
001010111111000000111111000000111111
 7   9  11  13
   1      1
 3   3   3   3   3   3   1   1   1   1   1   1   1   1
  0    0    6
  6    9    9    3         16.
  6   11   11    3         16.
  6   13   13    3          4.
  8    9    9    3         16.
  8   11   11    3         16.
  8   13   13    3          4.
STOP
```

APPENDIX C

Fortran IV Listing of Program

Considerable time, effort, and expense have gone into the development of the computer program. It is obvious that it should be used under the conditions and assumptions for which it was developed. These are described in this research report. Although the program has been extensively tested by the authors, no warranty is made regarding the accuracy and reliability of the program and no responsibility is assumed by the authors or by the sponsors of this research project.

```
      PROGRAM FINPLA2 (INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE4,TAPE5,TAPE6)
C
C                *****************************************
C                *             FINPLA-2                  *
C                * FINITE  ELEMENT  ANALYSIS OF  GENERAL *
C                * NON-PRISMATIC FOLDED PLATE STRUCTURES *
C                * WITH INCORPORATED ARBITRARY  3D-FRAME *
C                *****************************************
C
C                PROGRAMMED BY CHRISTIAN MEYER
C                UNIVERSITY OF CALIFORNIA, BERKELEY, JANUARY, 1971
C
C**********************************************************************
C
C
C        COMMON AND DIMENSION STATEMENTS
C
      COMMON A(20000)
      COMMON/SETUP/NSEG,NSECT,NEL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,
     *  NTAD,NPLD,NCSTYP,NLC,NFORST,NSAV,NSCK,IA,IS,IM,IR,IL,ISTOP,ICHK,
     *  R,NMAX,NEQ,NEQS,NQ,MBAND,NBLCK,NDBC,JFLAG,PI,NDE,NDS,NFST,NFCT,
     *  NCOM ,NBR,IC1,IC2,IC3,IC4,IC5,IC6,NGIR,      IREACT,IRESID,
     *  K1,K2,K3,K4,K5,K6
      DIMENSION TITLE(16)
      DATA SFLAG/5HSTART/, STOPF/4HSTOP/
C
C        DEFINE SCRATCH TAPES
C
      NCOMM=20000
      NS2=10
      K1=1
      K2=2
      K3=3
      K4=4
      K5=5
      K6=6
      K1=16
      K2=17
      K3=26
      K4=27
      K5=28
      K6=29
C
C        READ AND PRINT CONTROL CARDS
C
    1 READ 1001, CHECK
      IF(CHECK.EQ.SFLAG) GO TO 10
      IF(CHECK.EQ.STOPF) STOP
      GO TO 1
C
   10 CALL SECOND (T1)
      JFLAG=0
      READ 1001, (TITLE(I), I=1,16)
      READ 1002, NSEG,NEL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,NTAD,NPLD,
     *  NCSTYP,NLC,NFORST,NSAV,NSCK,NGIR,IA,IS,IM,IR,IL,ISTOP,ICHK,R
      PRINT 2000, (TITLE(I), I=1,16)
      PRINT 2001, NSEG,NEL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,NTAD,NPLD,
     *  NCSTYP,NLC,NFORST,NSAV,NSCK,NGIR
      PRINT 2002, IA,IS,IM,IR,IL,ISTOP,ICHK,R
```

```
      IC1=0
      IC2=0
      IC3=0
      IC4=0
      IC5=0
      IC6=0
      IF(ICHK.EQ.0) GO TO 15
      READ 1002,  IC1,IC2,IC3,IC4,IC5,IC6
      PRINT 2008, IC1,IC2,IC3,IC4,IC5,IC6
C
C        PREPARE CALL OF INPUT AND SETUP SUBROUTINES
C
   15 NSECT=NSEG+1
      N1=1
      N2=N1+NEL
      N3=N2+NEL
      N4=N3+NSECT
      N5=N4+NSECT+1
      N6=N5+NFORST
      N7=N6+NFORST*NSECT
      N8=N7+NFORST*NPTS
      N9=N8+NSAV
      N10=N9+NSAV*NSECT
      N11=N10+NSAV*NPTS*2
      N12=N11+NSCK
      N13=N12+2*NEL
      N14=N13+NEL
      N53=N14+NSEG
      N15=N53+NDIAPH
      N16=N15+NSECT
      K=2*NSECT*(NPTS+NS2)
      N17=N16+K
      N18=N17+K
      N19=N18+K
      N20=N19+4*NSEG*NEL
      N21=N20+NSECT
      N22=N21+NSECT
      N23=N22+NSECT
      N48=N23+6*NFET
      N49=N48+NFET
      N50=N49+NFET
      N24=N50+4*NFET
      N25=N24+NEL
      N26=N25+NSE
      N27=N26+NSE
      N28=N27+NSE
      N29=N28+NCSTYP*NPTS
      N30=N29+NCSTYP*NPTS
      N31=N30+NS2PT
      N32=N31+NS2PT
      K=NS2*NS2PT
      N33=N32+K
      N34=N33+K
      N35=N34+K
      N36=N35+K
      N37=N36+NSECT
      N38=N37+NDIAPH
      N39=N38+NTAD
      N40=N39+NTAD
      N41=N40+NTAD
```

```
      N42=N41+NTAD
      N43=N42+NTAD*(NPTS+NS2)
      N44=N43+NPLD
      N45=N44+NPLD
      N46=N45+NPLD
      N47=N46+NPLD
      NNN=N47+NPLD-1
      IF(NNN.LE.NCOMM) GO TO 16
      PH=6H INPUT
      PRINT 2004, PH, NNN
      GO TO 1
   16 CONTINUE
C
      CALL INPUT(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     *  A(N10),A(N11),A(N12),A(N13),A(N14),A(N20),A(N21),A(N22),A(N23),
     *  A(N48),A(N49),A(N50),A(N24),A(N25),A(N26),A(N27),A(N28),A(N29))
      CALL INPUT2 (A(N30),A(N31),A(N32),A(N33),A(N34),A(N35),A(N36),
     *  A(N37),A(N38),A(N39),A(N40),A(N41),A(N42),A(N43),A(N44),A(N45),
     *  A(N46),A(N47),NCSTYP,NS2PT,NFET,NTAD,NFORST,NSAV,A(N53))
      IF(JFLAG.EQ.1) GO TO 1
C
      N48=N47+NPLD
      N49=N48+NSEG
      N50=N49+NSEG
      N51=N50+NEQ
      N52=N51+NDBC
      NNN=N52+NDBC-1
      IF(NNN.LE.NCOMM) GO TO 18
      PH=6H SETUP
      PRINT 2004, PH, NNN
      GO TO 1
   18 NCOM=NCOMM-(N18+NEQ+2*NDBC)+1
C
      CALL SETUP1 (A(N1),A(N2),A(N3),A(N4),A(N15),A(N16),A(N17),A(N18),
     *  A(N19),A(N20),A(N21),A(N22),A(N23),A(N24),A(N25),A(N26),A(N27),
     *  A(N28),A(N29),A(N30),A(N31),A(N32),A(N33),A(N34),A(N35),A(N36))
      CALL SETUP2 (A(N36),A(N37),A(N38),A(N39),A(N40),A(N41),A(N42),
     *  A(N43),A(N44),A(N45),A(N46),A(N47),A(N48),A(N48),A(N49),A(N50),
     *  A(N50),A(N51),A(N52),NFET,NSECT,NMAX,NCSTYP,NS2PT,NTAD,NDBC)
      NCOM =NCOMM
      CALL SECOND (T2)
C
C         CALL SUBROUTINE LOADS FOR ONE LOAD CASE AT A TIME
C
   20 K=K4
      IF(NLC.EQ.1) K=K5
      REWIND K
      DO 50 L=1,NLC
      READ 1002, NESL,NSSL,NCL
      PRINT 2003,L,NESL,NSSL,NCL
      N21=N20+NESL
      N22=N21+NESL
      N23=N22+NESL
      N24=N23+NESL
      N25=N24+NESL
      N26=N25+NSSL
      N27=N26+NSSL
      N28=N27+NSSL
      N29=N28+NSSL
      N30=N29+NSSL
```

```
      N31=N30+NSSL
      N32=N31+NCL
      N33=N32+NCL
      N34=N33+NCL
      N35=N34+NCL
      N36=N35+NCL
      NNN=N36+NEQS-1
      IF(NNN.LE.NCOMM) GO TO 40
      PH=6H LOADS
      PRINT 2004, PH, NNN
      GO TO 1
   40 CONTINUE
C
      CALL LOADS (A(N1),A(N2),A(N4),A(N16),A(N17),A(N18),A(N19),A(N20),
     *  A(N21),A(N22),A(N23),A(N24),A(N25),A(N26),A(N27),A(N28),A(N29),
     *  A(N30),A(N31),A(N32),A(N33),A(N34),A(N35),A(N36),NESL,NSSL,NCL,
     *  NSECT,NMAX,K,L)
   50 CONTINUE
      CALL SECOND (T3)
      IF(ISTOP.EQ.0) GO TO 60
      PRINT 2005
      GO TO 1
C
C         CALL FORMK SUBROUTINE
C
   60 NL=2*NQ
      IF(NQ.EQ.NEQ) NL=NQ
      N19=N18+NQ*NLC
      N20=N19+NEQ
      N21=N20+NDBC
      N22=N21+NDBC
      N23=N22+NQ*MBAND
      IF(NQ.LT.NEQ) N23=N23+NQ*MBAND
      NNN=N23+NQ*NLC-1
      IF(NNN.LE.NCOMM) GO TO 70
      PH=6H FORMK
      PRINT 2004, PH, NNN
      GO TO 1
   70 CONTINUE
C
      CALL FORMK (A(N1),A(N2),A(N3),A(N4),A(N15),A(N18),A(N19),A(N20),
     *  A(N21),A(N22),A(N23),NL,NQ,NLC,NDBC,MBAND)
      CALL SECOND (T4)
C
C         CALL EQUATION SOLVER  USOL
C
      NSB=NQ*(MBAND+NLC)
      N19=N18+NSB
      K=MAX0(NSB,NBR*NQ*NLC)
      IF(NQ.EQ.NEQ) K=2*NQ*NLC
      N20=N19+K
      NNN=N20+NQ-1
      IF(NNN.LE.NCOMM) GO TO 80
      PH=6H  USOL
      PRINT 2004, PH, NNN
      GO TO 1
   80 CONTINUE
      CALL USOL (A(N18),A(N19),A(N20),NQ,MBAND,NLC,NBLCK,NSB,K6,K5,K2,
     *  K3,K6)
      CALL SECOND (T5)
```

```
C
C         CALL SUBROUTINE OUTPUT
C
      N19=N18+NQ*NLC
      N20=N19+NEQ
      N21=N20+NDBC
      N22=N21+NDBC
      N23=N22+NQ*MBAND
      NOUTP=N23+NQ*NLC-1
      N24=N19+67*NLC
      NC=NCOM-N24+1
      NBL=NC/(NQ*NLC)
      IF(NBL.LT.1) NBL=1
      IF(NBL.GT.NBLCK) NBL=NBLCK
      NL=NBL*NQ
      K=N24+NL*NLC-1
      IF(K.GT.NOUTP) NOUTP=K
      K=60*NEL*NLC
      NC=NCOM-N18+1
      IF(NC.LT.K) JFLAG=1
      K=K+N18-1
      IF(K.GT.NOUTP) NOUTP=K
      K=NSCK*NGIR*NLC
      N25=N18+K
      N26=N25+K
      N27=N26+K
      K=6*NMAX
      N28=N27+MAX0(30,K)
      N29=N28+K
      N30=N29+K
      K=MAX0(30,NQ)
      K=N30+K*NLC-1
      IF(K.GT.NOUTP) NOUTP=K
      IF(NOUTP.LE.NCOMM) GO TO 90
      PH=6HOUTPUT
      PRINT 2004, PH, NOUTP
      GO TO 1
   90 CONTINUE
      K=30*NEL
C
      CALL OUTPUT (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9)
     * ,A(N10),A(N11),A(N12),A(N13),A(N14),A(N15),A(N16),A(N17),A(N18),
     *  A(N19),A(N20),A(N21),A(N22),A(N23))
      CALL OUTPU2 (A(N18),A(N19),A(N24),A(N18),A(N25),A(N26),A(N27),
     *  A(N28),A(N29),A(N30),A(N30),A(N30),A(N30),NFORST,NSAV,NSECT,NMAX
     * ,NQ,MBAND,NLC,NL,NSCK,NGIR,K,NEQ,NDBC,A(N53))
      CALL SECOND (T6)
      T1=T2-T1
      T2=T3-T2
      T3=T4-T3
      T4=T5-T4
      T5=T6-T5
      T6=T1+T2+T3+T4+T5
      PRINT 2007, T1,T2,T3,T4,T5,T6
      GO TO 1
C
C***********************************************************************
C         FORMAT STATEMENTS
C***********************************************************************
C
```

```
 1001 FORMAT(16A5)                                                        FINP 300
 1002 FORMAT(16I4,7I1,F9,0)                                               FINP 301
C                                                                         FINP 302
 2000 FORMAT(1H1,20X,16A5)                                                FINP 303
 2001 FORMAT(////40H NUMBER OF STRUCTURE SEGMENTS......... =I5/           FINP 304
     *            40H NUMBER OF PLATE ELEMENTS IN A SEGMENT =I5/          FINP 305
     *            40H NUMBER OF PRIMARY NODAL POINTS........ =I5/         FINP 306
     *            40H NUMBER OF SECTIONS W, SECONDARY N,PTS =I5/          FINP 307
     *            40H NUMBER OF FINITE ELEMENT PLATE TYPES, =I5/          FINP 308
     *            40H NUMBER OF SPECIAL PLATE ELEMENTS..... =I5/          FINP 309
     *            40H NUMBER OF DIAPHRAGMS................. =I5/          FINP 310
     *            40H NUMBER OF FRAME ELEMENTS............. =I5/          FINP 311
     *            40H NUMBER OF APPLIED GROUP DISPLACEMENTS =I5/          FINP 312
     *            40H NUMBER OF PRESCRIBED LINE DISPLACEMTS =I5/          FINP 313
     *            40H NUMBER OF CROSS SECTION TYPES........ =I5/          FINP 314
     *            40H NUMBER OF LOAD CASES FOR THIS PROBLEM =I5/          FINP 315
     *            40H NUMBER OF FORCE TRANSF. SECTION TYPES =I5/          FINP 316
     *            40H NUMBER OF FORCE AVERAG. SECTION TYPES =I5/          FINP 317
     *            40H NUMBER OF STATIC CHECK SECTIONS...... =I5/          FINP 318
     *            40H NUMBER OF GIRDERS.................... =I5)          FINP 319
 2002 FORMAT(     40H ALIGNMENT INDICATOR.................. =I5/          FINP 320
     *            40H SKEWNESS INDICATOR................... =I5/          FINP 321
     *            40H MATERIAL OPTION INDICATOR............ =I5/          FINP 322
     *            40H REFERENCE LINE INPUT OPTION.......... =I5/          FINP 323
     *            40H LOAD-DISPLACEMENT INPUT/OUTPUT OPTION =I5/          FINP 324
     *            40H INPUT CHECK TERMINATOR............... =I5/          FINP 325
     *            40H CHECK OPTION INDICATOR............... =I5/          FINP 326
     *            40H CURVATURE RADIUS OF REFERENCE LINE... =F12,4)       FINP 327
 2003 FORMAT(1H1,24H INPUT FOR LOAD CASE NO.,I4////                       FINP 328
     * 48H NUMBER OF ELEMENTS WITH REGULAR SURFACE LOADS =,I5/            FINP 329
     * 48H NUMBER OF ELEMENTS WITH SPECIAL SURFACE LOADS =,I5/            FINP 330
     * 48H NUMBER OF CONCENTRATED OR LINE LOADS          =,I5)            FINP 331
 2004 FORMAT(///18H ***** FATAL ERROR//32H TOO MUCH BLANK COMMON REQUESTFINP 332
     *ED/16H FOR SUBROUTINE ,A6,8H,  NNN =,I8)                            FINP 333
 2005 FORMAT(/////47H CHECK OF INPUT DATA FOR THIS PROBLEM COMPLETED)     FINP 334
 2007 FORMAT(/////16H EXECUTION TIMES//                                   FINP 335
     *  35H INPUT SETUP                      =F10,3,10H   SECONDS/        FINP 336
     *  35H FORMATION OF LOAD MATRIX         =F10,3,10H   SECONDS/        FINP 337
     *  35H FORMATION OF STRUCTURE STIFFNESS =F10,3,10H   SECONDS/        FINP 338
     *  35H SOLUTION OF EQUATIONS            =F10,3,10H   SECONDS/        FINP 339
     *  35H OUTPUT GENERATION                =F10,3,10H   SECONDS//       FINP 340
     *  35H TOTAL EXECUTION TIME             =F10,3,10H   SECONDS)        FINP 341
 2008 FORMAT(     40H OPTION TO PRINT REFERENCE POINT DATA, =I5/          FINP 342
     *            40H OPTION TO PRINT GLOBAL NODAL COORDS,, =I5/          FINP 343
     *            40H OPTION TO PRINT BOUND.COND.ARRAYS.... =I5/          FINP 344
     *            40H OPTION TO PRINT LOAD VECTORS......... =I5/          FINP 345
     *            40H OPTION TO PRINT ELEMENT STIFFNESSES,, =I5/          FINP 346
     *            40H OPTION TO PRINT STRUCTURE STIFFNESS,. =I5)          FINP 347
      END                                                                 FINP 348
```

```
      SUBROUTINE INPUT (NPI,NPJ,NPSEC,NBLK,NFORS,NFORSN,ALF,NSA,NSAVS,
     *  NPAV,NCS,NGE,XDIV,IOSEG,X,Y,G,C,TH,E,FNU,KPL,ISEL,ISES,ISET,YY,
     *  ZZ)
      RETURN
      ENTRY INPUT2 (I2SEC,I2PT,N2PT,SX,SY,SZ,NSCT,NSD,INDT,NANP,NSAD,
     *  DTIN,NAD,NJPD,NSDS,NSDE,INPD,PDIS,NC,N2,NT,ND,NF,NS,IODIA)
C
C***********************************************************************
C     THIS SUBROUTINE READS AND PRINTS ALL INPUT EXCEPT THE LOAD DECK
C***********************************************************************
C
C         COMMON AND DIMENSION STATEMENTS
C
      DIMENSION X(1),Y(1),G(1),YY(NC,1),ZZ(NC,1),I2SEC(1),I2PT(1),
     *  N2PT(N2,1),SX(N2,1),SY(N2,1),SZ(N2,1),NSCT(1),TH(1),E(1),FNU(1),
     *  C(NT,12),NPI(1),NPJ(1),KPL(1),NGE(2,1),ISEL(1),ISES(1),ISET(1),
     *  NSD(1),NCS(1),NPSEC(1),NBLK(1),INDT(1),NANP(1),NSAD(1),DTIN(1),
     *  NAD(ND,1),NJPD(1),NSDS(1),NSDE(1),INPD(1),PDIS(1),NFORS(1),
     *  NFORSN(NF,1),ALF(NF,1),NSA(1),NSAVS(NS,1),NPAV(NS,1),XDIV(1),
     *  IOSEG(1),IODIA(1)
C
      COMMON/SETUP/NSEG,NSECT,NEL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,
     *  NTAD,NPLD,NCSTYP,NLC,NFORST,NSAV,NSCK,IA,IS,IM,IR,IL,ISTOP,ICHK,
     *  R,NMAX,NEQ,NEQS,NQ,MBAND,NBLCK,NDBC,JFLAG,PI,NDE,NDS,NFST,NFCT,
     *  NCOMM,NBR,IC1,IC2,IC3,IC4,IC5,IC6,NGIR,IREACT,IRESID,
     *  K1,K2,K3,K4,K5,K6
      COMMON/DIAPH/NDTY(20),NPD(4,20),NDSE(20),NDSD(20),NDST(20)
      COMMON/FRAME/EA(20),EIY(20),EIZ(20),GJX(20),GAY(20),GAZ(20),
     *  EI(3,20),EJ(3,20),EK(3,20),NFI(100),NFJ(100),NSFI(100),
     *  NSFJ(100),NFTY(100),NFSC(100)
C
C     EQUIVALENCE (TH,C(1,7)),(E,C(1,8)),(FNU,C(1,9))
C     BY ALLOCATING SAME FIRST WORD ADDRESS IN BLANK COMMON AREA
      NS2=10
C
C
C***********************************************************************
C         1) READ AND PRINT INPUT DATA ON STRUCTURAL CONFIGURATION
C***********************************************************************
C
      READ 1000, (X(I),Y(I),G(I), I=1,NSECT)
      PRINT 2000
      PRINT 2001, (X(I),Y(I),G(I), I=1,NSECT)
C
      DO 10 I=1,NCSTYP
      DO  5 J=1,NPTS
    5 READ 1004, K,YY(I,K),ZZ(I,K)
      PRINT 2002, I
      PRINT 2003, (J,YY(I,J),ZZ(I,J), J=1,NPTS)
   10 CONTINUE
C
      IF(NS2PT.LE.0) GO TO 30
      DO 20 I=1,NS2PT
      READ 1002, I2SEC(I),I2PT(I)
      JJ=I2PT(I)
      DO 15 J=1,JJ
   15 READ 1004, N2PT(I,J),SX(I,J),SY(I,J),SZ(I,J)
   20 CONTINUE
      PRINT 2004
```

```
      DO 25 I=1,NS2PT
      PRINT 2005, I2SEC(I)
      JJ=I2PT(I)
      IF(JJ.GT.NS2) GO TO 270
   25 PRINT 2006, (N2PT(I,J),SX(I,J),SY(I,J),SZ(I,J), J=1,JJ)
C
   30 READ 1003, (NSCT(I), I=1,NSECT)
      PRINT 2007
      PRINT 2008, (NSCT(I), I=1,NSECT)
      DO 35 I=1,NSECT
      IF(NSCT(I).GT.NCSTYP.OR.NSCT(I).LE.0) GO TO 281
   35 CONTINUE
C
C**********************************************************************
C         2) READ AND PRINT INPUT DATA ON PLATE SYSTEM
C**********************************************************************
C
      IF(IM.EQ.2) GO TO 120
      DO 105 I=1,NFET
  105 READ 1004, J,TH(J),E(J),FNU(J)
      PRINT 2013
      DO 115 I=1,NFET
      U=FNU(I)
      T=TH(I)
      PRINT 2006, I,T,E(I),U
      C(I,1)=T*E(I)/(1.-U*U)
      C(I,2)=C(I,1)
      C(I,3)=E(I)*T*0.5/(1.+U)
      C(I,4)=U*C(I,1)
      C(I,5)=0.0
      C(I,6)=0.0
      U=T*T/12.
      DO 110 J=7,12
  110 C(I,J)=C(I,J-6)*U
  115 CONTINUE
      GO TO 130
C
  120 DO 125 I=1,NFET
  125 READ 1004, J, (C(J,K), K=1,12)
      PRINT 2014
      PRINT 2015, (I,(C(I,J), J=1,12), I=1,NFET)
C
  130 DO 132 J=1,NEL
  132 READ 1010,  I,NPI(I),NPJ(I),KPL(I),NGE(1,I),NGE(2,I),XDIV(I)
      PRINT 2016
      PRINT 2044, (I,NPI(I),NPJ(I),KPL(I),NGE(1,I),NGE(2,I),
     *  XDIV(I), I=1,NEL)
      DO 133 I=1,NEL
      IF(NPI(I).GT.NPTS.OR.NPI(I).LE.0) GO TO 282
      IF(NPJ(I).GT.NPTS.OR.NPJ(I).LE.0) GO TO 282
      IF(KPL(I).GT.NFET.OR.KPL(I).LE.0) GO TO 283
  133 CONTINUE
C
      IF(NSE.LE.0) GO TO 140
      DO 135 I=1,NSE
  135 READ 1005, J,ISEL(J),ISES(J),ISET(J)
      PRINT 2018
      PRINT 2019, (I,ISEL(I),ISES(I),ISET(I), I=1,NSE)
C
C**********************************************************************
```

```
C         3) READ AND PRINT DIAPHRAGM DATA
C***********************************************************************
C
  140 IF(NDIAPH.LE.0) GO TO 150
      READ 1002, NDE,NDS
      IF(NDE.GT.20.OR.NDS.GT.20) GO TO 275
      READ 1003, (NSD(I), I=1,NDIAPH)
      PRINT 2020
      PRINT 2008, (NSD(I), I=1,NDIAPH)
      DO 141 I=1,NDIAPH
      IF(NSD(I).GT.NSECT.OR.NSD(I).LE.0) GO TO 281
  141 CONTINUE
C
      DO 142 J=1,NDE
  142 READ 1005,  I,NDTY(I),(NPD(K,I),K=1,4)
      PRINT 2021
      PRINT 2022, (I,NDTY(I),(NPD(K,I),K=1,4), I=1,NDE)
C
      IF(NDS.LE.0) GO TO 150
      DO 145 I=1,NDS
  145 READ 1005, J,NDSE(J),NDSD(J),NDST(J)
      PRINT 2023
      PRINT 2019, (J,NDSE(J),NDSD(J),NDST(J), J=1,NDS)
      DO 147 I=1,NDS
      J=NDSD(I)
  147 NDSD(I)=NSD(J)
C
C***********************************************************************
C         4) READ AND PRINT FRAME INPUT DATA
C***********************************************************************
C
  150 IF(NFRAME.LE.0) GO TO 180
      READ 1002, NFST,NFCT
      IF(NFST.GT.20.OR.NFCT.GT.20.OR.NFRAME.GT.100) GO TO 280
C
      DO 155 I=1,NFST
  155 READ 1011, J,EA(J),EIY(J),EIZ(J),GJX(J),GAY(J),GAZ(J)
      PRINT 2024
      PRINT 2025,(I,EA(I),EIY(I),EIZ(I),GJX(I),GAY(I),GAZ(I),I=1,NFST)
C
      PRINT 2026
      DO 160 I=1,NFCT
      READ 1006, J,(EI(K,J),K=1,3),(EJ(L,J),L=1,3),(EK(M,J),M=1,3)
  160 PRINT 2027,J,(EI(K,J),K=1,3),(EJ(L,J),L=1,3),(EK(M,J),M=1,3)
C
      PRINT 2028
      DO 165 I=1,NFRAME
      READ 1009, J,NFI(J),NFJ(J),NSFI(J),NSFJ(J),NFTY(J),NFSC(J)
  165 PRINT 2045,J,NFI(J),NFJ(J),NSFI(J),NSFJ(J),NFTY(J),NFSC(J)
C
C***********************************************************************
C        5) DETERMINE NUMBER OF NODAL POINTS FOR EACH SECTION
C            AND TOTAL NUMBER OF EQUATIONS
C***********************************************************************
C
  180 DO 185 I=1,NSECT
  185 NPSEC(I)=NPTS
      NMAX=NPTS
      IF(NS2PT.LE.0) GO TO 195
      DO 190 I=1,NS2PT
```

```
      II=I2SEC(I)
      JJ=NPTS+I2PT(I)
      IF(JJ.GT.NMAX) NMAX=JJ
  190 NPSEC(II)=JJ
  195 NBLK(1)=1
      DO 200 I=1,NSECT
  200 NBLK(I+1)=NBLK(I)+NPSEC(I)*6
      NEQ=NBLK(NSECT+1)-1
C
C***********************************************************************
C        6) READ AND PRINT BOUNDARY CONDITIONS
C***********************************************************************
C
      NDBC=0
      IF(NTAD.LE.0) GO TO 225
      DO 215 I=1,NTAD
      READ 1007, J,INDT(J),NANP(J),NSAD(J),DTIN(J)
      II=NANP(J)
      JJ=NSAD(J)
      IF(DTIN(J).NE.0.) NDBC=NDBC+II
      IF(II.EQ.NPSEC(JJ)) GO TO 205
      READ 1003, (NAD(J,K), K=1,II)
      GO TO 215
  205 DO 210 K=1,II
  210 NAD(J,K)=K
  215 CONTINUE
C
      PRINT 2029
      DO 220 I=1,NTAD
      PRINT 2030, I,INDT(I),NANP(I),NSAD(I),DTIN(I)
      II=NANP(I)
      PRINT 2031, (NAD(I,J), J=1,II)
  220 CONTINUE
C
  225 IF(NPLD.LE.0) GO TO 240
      DO 230 I=1,NPLD
      READ 1007, NJPD(I),NSDS(I),NSDE(I),INPD(I),PDIS(I)
      IF(PDIS(I).NE.0.) NDBC=NDBC+NSDE(I)-NSDS(I)+1
  230 CONTINUE
      PRINT 2032
      PRINT 2030, (NJPD(I),NSDS(I),NSDE(I),INPD(I),PDIS(I), I=1,NPLD)
C
C***********************************************************************
C        7) READ AND PRINT INFORMATION ON INTERNAL FORCE TRANSFORMATION
C***********************************************************************
C
  240 IF(NFORST.LE.0) GO TO 250
      PRINT 2033
      DO 245 I=1,NFORST
      READ 1003, K,(NFORSN(I,J), J=1,K)
      NFORS(I)=K
      READ 1012, (ALF(I,J), J=1,NPTS)
      PRINT 2034, I
      PRINT 2008, (NFORSN(I,J), J=1,K)
      PRINT 2035
      PRINT 2036, (J,ALF(I,J), J=1,NPTS)
  245 CONTINUE
C
C***********************************************************************
C        8) READ AND PRINT INFORMATION ON INTERNAL FORCE AVERAGING
```

```
C             AND STATIC CHECKS AND OUTPUT OPTIONS
C*********************************************************************
C
  250 IF(NSAV.LE.0) GO TO 260
      PRINT 2037
      DO 255 I=1,NSAV
      READ 1003, K,(NSAVS(I,J), J=1,K)
      NSA(I)=K
      IJ=2*NPTS
      READ 1008, (NPAV(I,J), J=1,IJ)
      PRINT 2034, I
      PRINT 2008, (NSAVS(I,J), J=1,K)
      PRINT 2038
      DO 252 J=1,NPTS
  252 PRINT 2039, J,NPAV(I,2*J-1),NPAV(I,2*J)
  255 CONTINUE
C
  260 IF(NSCK.LE.0) GO TO 265
      READ 1003, (NCS(I), I=1,NSCK)
      PRINT 2040
      PRINT 2008, (NCS(I), I=1,NSCK)
C
  265 READ 1002, IREACT,IRESID
      READ 1003, (IOSEG(I), I=1,NSEG)
      IF(IREACT.EQ.1) PRINT 2046
      IF(IREACT.EQ.0) PRINT 2047
      IF(IRESID.EQ.1) PRINT 2048
      IF(IRESID.EQ.0) PRINT 2049
      PRINT 2050
      PRINT 2008, (IOSEG(I), I=1,NSEG)
      IF(NDIAPH.LE.0) GO TO 267
      READ 1003, (IODIA(I), I=1,NDIAPH)
      PRINT 2051
      PRINT 2008, (IODIA(I), I=1,NDIAPH)
  267 PRINT 2052
      GO TO 300
C
C*********************************************************************
C         9) ERROR EXITS
C*********************************************************************
C
  270 PRINT 2041
      GO TO 295
  275 PRINT 2042
      GO TO 295
  280 PRINT 2043
      GO TO 295
  281 PRINT 2053
      GO TO 295
  282 PRINT 2054
      GO TO 295
  283 PRINT 2055
  295 JFLAG=1
C
  300 RETURN
C
C*********************************************************************
C         10) FORMAT STATEMENTS
C*********************************************************************
C
```

```
 1000 FORMAT(3F10.0)
 1002 FORMAT(2I5,4F10.0)
 1003 FORMAT(25I3)
 1004 FORMAT(I10,6F10.0/10X,6F10.0)
 1005 FORMAT(6I4)
 1006 FORMAT(I4,9F8.0)
 1007 FORMAT(4I4,F10.0)
 1008 FORMAT(12I6)
 1009 FORMAT(7I4)
 1010 FORMAT(6I4,F10.0)
 1011 FORMAT(I10,6F10.0)
 1012 FORMAT(9F8.0)
C
 2000 FORMAT(/////27H REFERENCE LINE COORDINATES//36H    X/THETA
     *Y            GAMMA)
 2001 FORMAT(3F12.4)
 2002 FORMAT(1H1,51H NODAL POINT COORDINATES FOR CROSS SECTION TYPE NO.,
     * I4//31H POINT NO.  Y-COORD      Z-COORD)
 2003 FORMAT(I7,2F12.3)
 2004 FORMAT(1H1,34H SECONDARY NODAL POINT COORDINATES//)
 2005 FORMAT(//12H SECTION NO.,I4//46H POINT NO.   X-COORD       Y-COORD
     *      Z-COORD)
 2006 FORMAT(I7,3F13.3)
 2007 FORMAT(///////44H CROSS SECTION TYPE NUMBERS FOR ALL SECTIONS//)
 2008 FORMAT(25I4)
 2013 FORMAT(1H1,35H PROPERTIES OF FINITE ELEMENT TYPES//52H  TYPE NO.
     *THICKNESS  ELASTIC MOD.  POISSON-S RATIO)
 2014 FORMAT(1H1,35H PROPERTIES OF FINITE ELEMENT TYPES//129H  TYPE NO.
     *  C(1,1)     C(2,2)     C(3,3)     C(1,2)     C(1,3)     C(2,3)     D(1,
     *1)    D(2,2)     D(3,3)     D(1,2)     D(1,3)     D(2,3))
 2015 FORMAT(I7,3X,1P12E10.3)
 2016 FORMAT(1H1,58H PROPERTIES OF PLATE ELEMENTS IN REGULAR STRUCTURE S
     *EGMENT//89H  ELEMENT NO.    NODE I       NODE J      TYPE NO.      1ST
     * GIRDER NO.  2ND GIRDER NO.    XDIV)
 2017 FORMAT(4I12,2I16)
 2018 FORMAT(1H1,33H SPECIAL PLATE ELEMENT PROPERTIES//67H  SPECIAL ELE.
     *NO.  CORRESP.REGULAR ELE.NO.  SEGMENT NO.     TYPE NO.)
 2019 FORMAT(I10,2I20,I14)
 2020 FORMAT(1H1,35H DIAPHRAGMS ARE PRESENT AT SECTIONS//)
 2021 FORMAT(////27H REGULAR DIAPHRAGM ELEMENTS//70H  ELEMENT NO.  TYPE
     *NO.      NODE I       NODE J       NODE K       NODE L)
 2022 FORMAT(I8,5I12)
 2023 FORMAT(////27H SPECIAL DIAPHRAGM ELEMENTS//67H  SPECIAL ELE.NO.  C
     *ORRESP.REGULAR ELE.NO.  DIAPHRAGM NO.  TYPE NO.)
 2024 FORMAT(1H1,42H PROPERTIES OF FRAME ELEMENT SECTION TYPES//89H  TYP
     *E NO.        EA              EI-Y             EI-Z             GJ-X
     *GA-Y          GA-Z)
 2025 FORMAT(I7,6E14.6)
 2026 FORMAT(/////45H PROPERTIES OF FRAME ELEMENT CONNECTION TYPES//20X,
     * 7H NODE I,22X,7H NODE J,22X,8H POINT K//12H  TYPE NO.  ,
     * 3(22H     X        Y         Z,7X))
 2027 FORMAT(I7,3(5X,3F8.3))
 2028 FORMAT(/////15H FRAME ELEMENTS//121H ELE.NO.          NODE I JOINT
     *      NODE J JOINT       NODE I SECT.       NODE J SECT.   SECTION TY
     *PE NO.   CONNECT.TYPE NO.)
 2029 FORMAT(1H1,28H APPLIED GROUP DISPLACEMENTS//110H  DISPL.NO.      CO
     *MPONENT NO.    NO.OF AFFECTED N.PTS   SECTION NO.       DISPL.INTENSI
     *TY   AFFECTED NODAL POINTS)
 2030 FORMAT(I7,3I18,F20.5)
 2031 FORMAT(1H+,86X,10I4/)
```

```
2032 FORMAT(1H1,30H PRESCRIBED LINE DISPLACEMENTS//85H  JOINT NO.
    * START SECTION     END SECTION        COMPONENT NO.   DISPL.INTENSI
    *TY)
2033 FORMAT(1H1,49H DATA FOR TRANSFORMATION OF INTERNAL FORCE OUTPUT//)
2034 FORMAT(///17H SECTION TYPE NO.,I4,25H   APPLICABLE TO SECTIONS/)
2035 FORMAT(///19H ANGLES OF ROTATION//28H  NODAL POINT    ALPHA(DEGR.))
2036 FORMAT(I8,F17.3)
2037 FORMAT(1H1,44H DATA FOR AVERAGING OF INTERNAL FORCE OUTPUT///)
2038 FORMAT(///51H  NODAL POINT   TRANSV.AVE.INDEX   LONGIT.AVE.INDEX)
2039 FORMAT(I8,2(13X,I6))
2040 FORMAT(///////39H STATIC CHECKS ARE DESIRED FOR SECTIONS/)
2041 FORMAT(///19H *****FATAL ERROR -/48H TOO MANY SECONDARY NODAL POIN
    *TS IN THIS SECTION)
2042 FORMAT(///19H *****FATAL ERROR -/62H TOO MANY DIAPHRAGM ELEMENTS A
    *ND/OR SPECIAL DIAPHRAGM ELEMENTS)
2043 FORMAT(///19H *****FATAL ERROR -/33H TOO MANY FRAME ELEMENTS OR TY
    *PES)
2044 FORMAT(I8,3I12,2I16,F14.3)
2045 FORMAT(I5,6I18)
2046 FORMAT(////29H REACTIONS WILL BE CALCULATED)
2047 FORMAT(////33H REACTIONS WILL NOT BE CALCULATED)
2048 FORMAT(////34H RESIDUAL LOADS WILL BE CALCULATED)
2049 FORMAT(////38H RESIDUAL LOADS WILL NOT BE CALCULATED)
2050 FORMAT(////26H SEGMENT OUTPUT INDICATORS//)
2051 FORMAT(////28H DIAPHRAGM OUTPUT INDICATORS//)
2052 FORMAT(////16H    INDICATOR KEY//,
    *  35H     1  CALCULATE AND PRINT STRESSES/,
    *  35H     2  CALCULATE STRESSES ONLY      /,
    *  35H     3  DO NOT CALCULATE STREESSES    )
2053 FORMAT(///19H *****FATAL ERROR -/30H WRONG CROSS SECTION SPECIFIED
    *  )
2054 FORMAT(///19H *****FATAL ERROR -/28H WRONG NODAL POINT SPECIFIED)
2055 FORMAT(///19H *****FATAL ERROR -/29H WRONG ELEMENT TYPE SPECIFIED)
     END
```

```
      SUBROUTINE SETUP1(NPI,NPJ,NPSEC,NBLK,IFD,TZ,TY,TX,AREA,X,Y,G,C,   SETU
     *  KPL,ISEL,ISES,ISET,YY,ZZ,I2SEC,I2PT,N2PT,SX,SY,SZ,NSCT)         SETU
      RETURN                                                            SETU
      ENTRY SETUP2 (JTYPE,NSD,INDT,NANP,NSAD,DTIN,NAD,NJPD,NSDS,NSDE,   SETU
     *  INPD,PDIS,ALPH,ICLASS,A,INDCR,ITYPE,IDISBC,DISPL,NT,NS,NM,NC,   SETU
     *  N2,ND,NA)                                                       SETU
C                                                                       SETU
C***********************************************************************SETU
C     THIS SUBROUTINE GENERATES GLOBAL NODAL COORDINATES IN THE DESIRED SETU
C     COORDINATE FRAME, ESTABLISHES THE BOUNDARY CONDITION ARRAYS, AND  SETU
C     CALCULATES ALL ELEMENT STIFFNESSES AND STORES THEM ON TAPE,       SETU
C***********************************************************************SETU
C                                                                       SETU
C          COMMON AND DIMENSION STATEMENTS                              SETU
C                                                                       SETU
       DIMENSION NPI(1),NPJ(1),NPSEC(1),NBLK(1),IFD(1),TX(NS,NM,1),     SETU
     *  TY(NS,NM,1),TZ(NS,NM,1),AREA(4,1),X(1),Y(1),G(1),C(NT,1),KPL(1),SETU
     *  ISEL(1),ISES(1),ISET(1),YY(NC,1),ZZ(NC,1),I2SEC(1),I2PT(1),     SETU
     *  N2PT(N2,1),SX(N2,1),SY(N2,1),SZ(N2,1),NSCT(1),NSD(1),INDCR(1),  SETU
     *  INDT(1),NANP(1),NSAD(1),DTIN(1),NAD(ND,1),NJPD(1),NSDS(1),      SETU
     *  NSDE(1),INPD(1),PDIS(1),ALPH(1),A(1),IDISBC(NA),DISPL(NA),      SETU
     *  ITYPE(NS,1),ICLASS(1),JTYPE(1)                                  SETU
C                                                                       SETU
      COMMON/SETUP/NSEG,NSECT,NEL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,    SETU
     *  NTAD,NPLD,NCSTYP,NLC,NFORST,NSAV,NSCK,IA,IS,IM,IR,IL,ISTOP,ICHK,SETU
     *  R,NMAX,NEQ,NEQS,NG,MBAND,NBLCK,NDBC,JFLAG,PI,NDE,NDS,NFST,NFCT, SETU
     *  NCOMM,NBR,IC1,IC2,IC3,IC4,IC5,IC6,NGIR,IREACT,IRESID,           SETU
     *  K1,K2,K3,K4,K5,K6                                               SETU
      COMMON/DIAPH/NDTY(20),NPD(4,20),NDSE(20),NDSD(20),NDST(20)        SETU
      COMMON/FRAME/EA(20),EIY(20),EIZ(20),GJX(20),GAY(20),GAZ(20),      SETU
     *  EI(3,20),EJ(3,20),EK(3,20),NFI(100),NFJ(100),NSFI(100),         SETU
     *  NSFJ(100),NFTY(100),NFSC(100)                                   SETU
      COMMON/STIFF/T(24,24),XE(4),YE(4),ZE(4),CC(6),DD(9),AA(4),        SETU
     *  IFLAG,AVE,GG,XT(167)                                            SETU
      DIMENSION ST(12,12),ECI(3),ECJ(3),ECK(3),CHK(3,3)                 SETU
      EQUIVALENCE (T,ST),(ECI,CC),(ECJ,CC(4)),(ECK,DD)                  SETU
C     EQUIVALENCE (INDCR,ITYPE),(ALPH,ICLASS),(NSCT,JTYPE)   (SAME FWA) SETU
      LOGICAL IFLAG                                                     SETU
C                                                                       SETU
C***********************************************************************SETU
C          1) GENERATE MISSING INPUT INFORMATION FOR REFERENCE LINE POINTSSETU
C***********************************************************************SETU
C                                                                       SETU
C          ARBITRARY REFERENCE LINE                                     SETU
C                                                                       SETU
      PI=3,14159265358979                                               SETU
      PI2=PI*0,5                                                        SETU
      FAC=PI/180.                                                       SETU
      GO TO (300,310,350), IA                                           SETU
  300 DO 302 I=1,NSEG                                                   SETU
  302 ALPH(I)=ALPHA(X(I),X(I+1),Y(I),Y(I+1))                            SETU
      IF(IS.EQ.2) GO TO 360                                             SETU
      G(1)=ALPH(1)                                                      SETU
      IF(NSEG.EQ.1) GO TO 307                                           SETU
      DO 305 I=2,NSEG                                                   SETU
  305 G(I)=0.5*(ALPH(I)+ALPH(I-1))                                      SETU
  307 G(NSECT)=ALPH(NSEG)                                               SETU
      GO TO 370                                                         SETU
C                                                                       SETU
```

```
C          CIRCULAR REFERENCE LINE
C
  310 AR=ABS(R)
      SR=R/AR
      XX=FAC
      IF(IR.EQ.2) XX=1./AR
      DO 325 I=1,NSECT
  325 X(I)=X(I)*XX
      TETA2=X(NSECT)/2.
      XR=AR*SIN(TETA2)
      YR= R*COS(TETA2)
      IF(IS.EQ.2) GO TO 340
      DO 335 I=1,NSECT
  335 G(I)=PI2
      GO TO 343
  340 DO 342 I=1,NSECT
  342 G(I)=G(I)*FAC
  343 DO 345 I=1,NSECT
      TT=TETA2-X(I)
      X(I)=XR-AR*SIN(TT)
      Y(I)=YR- R*COS(TT)
      IF(I.GT.1) ALPH(I-1)=ALPHA(X(I-1),X(I),Y(I-1),Y(I))
  345 G(I)=G(I)-PI2-SR*TT
      GO TO 370
C
C          STRAIGHT REFERENCE LINE
C
  350 DO 355 I=1,NSEG
      ALPH(I)=0.0
  355 A(I)=X(I+1)-X(I)
  360 DO 365 I=1,NSECT
  365 G(I)=G(I)*FAC
      IF(IA.EQ.3) GO TO 380
C
C          DISTANCE BETWEEN CONSECUTIVE REFERENCE POINTS
C
  370 DO 375 I=1,NSEG
  375 A(I)=SQRT((X(I+1)-X(I))**2+(Y(I+1)-Y(I))**2)
C
C*****************************************************************
C          2) PRINT GENERATED REFERENCE POINT DATA IF REQUESTED
C*****************************************************************
C
  380 IF(IC1.EQ.0)  GO TO 400
      PRINT 2009
      DO 385 I=1,NSEG
      TT=G(I)/FAC
      XX=ALPH(I)/FAC
  385 PRINT 2010, I,X(I),Y(I),TT,I,A(I),XX
      TT=G(NSECT)/FAC
      PRINT 2010, NSECT,X(NSECT),Y(NSECT),TT
      PRINT 2011
C
C*****************************************************************
C          3) DETERMINE MAXIMUM BANDWIDTH AND LENGTH OF EQUATION BLOCKS
C*****************************************************************
C
  400 JJ=0
      DO 405 I=1,NEL
      IJ=IABS(NPI(I)-NPJ(I))
```

```
      DO 405 J=1,NSEG
      II=IJ+NPSEC(J)
      IF(II.GT.JJ) JJ=II
  405 CONTINUE
C
      IF(NFRAME.LE.0) GO TO 430
      DO 410 I=1,NFRAME
      IF(NSFI(I).EQ.NSFJ(I)) GO TO 410
      IJ=IABS(NFI(I)-NFJ(I))
      IK=MIN0(NSFI(I),NSFJ(I))
      II=NPSEC(IK)+IJ
      IF(II.GT.JJ) JJ=II
  410 CONTINUE
C
  430 MBAND=6*JJ+6
      J=NEQ*(MBAND+NLC+1)
      IF(J.GT.NCOMM) GO TO  435
      NQ=NEQ
      NBLCK=1
      GO TO 440
  435 NQ=NCOMM/(2*(MBAND+NLC)+1)
      NBLCK=NEQ/NQ
      IF(NQ*NBLCK.LT.NEQ) NBLCK=NBLCK+1
  440 NEQS=NQ*NBLCK
      PRINT 2050, NEQ,NBLCK,NQ,MBAND
      IF(ICHK.EQ.0.AND.ISTOP.EQ.1) GO TO 850
C
C***********************************************************************
C          4) SET UP BOUNDARY CONDITION ARRAYS
C***********************************************************************
C
      DO 450 I=1,NEQ
  450 INDCR(I)=0
C
C         APPLIED GROUP DISPLACEMENTS
C
      IK=0
      IF(NTAD.LE.0) GO TO 465
      DO 460 I=1,NTAD
      II=NSAD(I)
      JJ=NBLK(II)+INDT(I)-7
      IJ=NAMP(I)
      DO 455 J=1,IJ
      JK=JJ+6*NAD(I,J)
      IF(DTIN(I).EQ.0.) GO TO 455
      IK=IK+1
      IDISBC(IK)=JK
      DISPL(IK)=DTIN(I)
  455 INDCR(JK)=1
  460 CONTINUE
C
C         PRESCRIBED LINE DISPLACEMENTS
C
  465 IF(NPLD.LE.0) GO TO 480
      DO 475 I=1,NPLD
      II=NSDS(I)
      IJ=NSDE(I)
      JJ=6*NJPD(I)+INPD(I)-7
      DO 470 J=II,IJ
      JK=JJ+NBLK(J)
```

```
      IF(PDIS(I).EQ.0.) GO TO 470
      IK=IK+1
      IDISBC(IK)=JK
      DISPL(IK)=PDIS(I)
  470 INDCR(JK)=1
  475 CONTINUE
C
C         ORDER NON-ZERO DISPLACEMENT ARRAY IN ASCENDING SEQUENCE
C
  480 IF(NDBC.LE.1) GO TO 495
      JK=NDBC-1
      DO 490 I=1,JK
      II=IDISBC(I)
      JJ=I
      IJ=I+1
      DO 485 J=IJ,NDBC
      IF(IDISBC(J).GE.II) GO TO 485
      II=IDISBC(J)
      JJ=J
  485 CONTINUE
      XX=DISPL(JJ)
      IDISBC(JJ)=IDISBC(I)
      DISPL(JJ)=DISPL(I)
      IDISBC(I)=II
  490 DISPL(I)=XX
C
C         SAVE BOUNDARY CONDITION ARRAYS ON TAPE K1 AND PRINT IF DESIRED
C
  495 IF(IC3.EQ.0) GO TO 497
      PRINT 2012
      PRINT 2013, (INDCR(J), J=1,NEQ)
      IF(NDBC.LE.0) GO TO 497
      PRINT 2014
      PRINT 2015, (IDISBC(J),DISPL(J), J=1,NDBC)
  497 REWIND K1
      WRITE (K1) (INDCR(J), J=1,NEQ)
      IF(NDBC.GT.0) WRITE (K1) IDISBC,DISPL
C
C**********************************************************************
C          5A) FIND ALL NODAL COORDINATES IN FIXED GLOBAL FRAME (IL=1)
C**********************************************************************
C
C         PRIMARY NODAL POINTS
C
  500 IF(IL.EQ.2) GO TO 530
      DO 510 I=1,NSECT
      IJ=NSCT(I)
      SB=SIN(G(I))
      CB=COS(G(I))
      DO 505 J=1,NPTS
      TX(I,J,1)=X(I)-YY(IJ,J)*SB
      TY(I,J,1)=Y(I)+YY(IJ,J)*CB
  505 TZ(I,J,1)=ZZ(IJ,J)
  510 CONTINUE
C
C         SECONDARY NODAL POINTS
C
      IF(NS2PT.LE.0) GO TO 544
      DO 525 I=1,NS2PT
      II=I2SEC(I)
```

```
      IJ=I2PT(I)
      SB=SIN(G(II))
      CB=COS(G(II))
      DO 520 J=1,IJ
      JJ=N2PT(I,J)
      TX(II,JJ,1)=X(II)-SY(I,J)*SB+SX(I,J)*CB
      TY(II,JJ,1)=Y(II)+SY(I,J)*CB+SX(I,J)*SB
  520 TZ(II,JJ,1)=SZ(I,J)
  525 CONTINUE
      GO TO 544
C
C*********************************************************************
C          5B) FIND NODAL COORDINATES IN TRAVELLING FRAME (IL=2)
C*********************************************************************
C
C          PRIMARY NODAL POINTS
C
  530 DO 534 I=1,NSECT
      IJ=NSCT(I)
      DO 531 J=1,NPTS
      TX(I,J,1)=0.0
      TY(I,J,1)=YY(IJ,J)
  531 TZ(I,J,1)=ZZ(IJ,J)
      IF(I.EQ.NSECT) GO TO 534
      IK=NSCT(I+1)
      A1=ALPH(I)-G(I)
      SA=A(I)*SIN(A1)
      CA=A(I)*COS(A1)
      A2=G(I+1)-G(I)
      SB=SIN(A2)
      CB=COS(A2)
      DO 532 J=1,NPTS
      TX(I,J,2)=CA-YY(IK,J)*SB
      TY(I,J,2)=SA+YY(IK,J)*CB
  532 TZ(I,J,2)=ZZ(IK,J)
  534 CONTINUE
C
C          SECONDARY NODAL POINTS
C
      IF(NS2PT.LE.0) GO TO 544
      DO 542 I=1,NS2PT
      II=I2SEC(I)
      IJ=I2PT(I)
      DO 536 J=1,IJ
      JJ=N2PT(I,J)
      TX(II,JJ,1)=SX(I,J)
      TY(II,JJ,1)=SY(I,J)
  536 TZ(II,JJ,1)=SZ(I,J)
      IF(II.EQ.1) GO TO 542
      JI=II-1
      A1=ALPH(JI)-G(JI)
      SA=A(JI)*SIN(A1)
      CA=A(JI)*COS(A1)
      A2=G(II)-G(JI)
      SB=SIN(A2)
      CB=COS(A2)
      DO 540 J=1,IJ
      JJ=N2PT(I,J)
      TX(JI,JJ,2)=CA-SY(I,J)*SB+SX(I,J)*CB
      TY(JI,JJ,2)=SA+SY(I,J)*CB+SX(I,J)*SB
```

```
  540 TZ(JI,JJ,2)=SZ(I,J)
  542 CONTINUE
C
C*****************************************************************************
C         5C) PRINT NODAL POINT COORDINATES IF DESIRED
C*****************************************************************************
C
  544 IF(IC2.EQ.0)  GO TO 550
      IF(IL.EQ.2) GO TO 546
      PRINT 2041
      DO 545 I=1,NSECT
      JJ=NPSEC(I)
      PRINT 2042, I
  545 PRINT 2043, (J,TX(I,J,1),TY(I,J,1),TZ(I,J,1), J=1,JJ)
      GO TO 550
  546 PRINT 2044
      DO 549 I=1,NSEG
      PRINT 2045, I
      JJ=NPSEC(I)
      II=NPSEC(I+1)
      IJ=MIN0(II,JJ)
      PRINT 2046, ((J,TX(I,J,K),TY(I,J,K),TZ(I,J,K), K=1,2), J=1,IJ)
      IF(JJ-II) 547,549,548
  547 IJ=JJ+1
      PRINT 2047, (J,TX(I,J,2),TY(I,J,2),TZ(I,J,2), J=IJ,II)
      GO TO 549
  548 IJ=II+1
      PRINT 2043, (J,TX(I,J,1),TY(I,J,1),TZ(I,J,1), J=IJ,JJ)
  549 CONTINUE
  550 IF(ISTOP.EQ.1) GO TO 850
C
C*****************************************************************************
C         6) CALCULATE PLATE ELEMENT STIFFNESSES AND STORE ON TAPE K2
C*****************************************************************************
C
      DO 551 I=1,NEL
      K=KPL(I)
      DO 551 J=1,NSEG
  551 ITYPE(J,I)=K
      IF(NSE.LE.0) GO TO 553
      DO 552 I=1,NSF
      J=ISES(I)
      K=ISEL(I)
  552 ITYPE(J,K)=ISET(I)
C
C     ESTABLISH SEGMENT TYPE ARRAYS
C
  553 ICLASS(1)=1
      JTYPE(1)=1
      NCLASS=1
      IF(NSEG.LE.1) GO TO 561
      DO 560 I=2,NSEG
      GG=G(I+1)-G(I)
      II=NSCT(I)
      JJ=NSCT(I+1)
      DO 558 J=1,NCLASS
      IJ=ICLASS(J)
      GY=G(IJ+1)-G(IJ)
      IK=NSCT(IJ)
      JK=NSCT(IJ+1)
```

```
      IF(IK.NE.II.OR.JK.NE.JJ) GO TO 558
      XX=ABS(GY-GG)+ABS(A(IJ)-A(I))
      IF(XX.GT..00001) GO TO 558
      DO 557 K=1,NEL
      IF(ITYPE(I,K).NE.ITYPE(IJ,K)) GO TO 558
  557 CONTINUE
      JTYPE(I)=J
      GO TO 560
  558 CONTINUE
      NCLASS=NCLASS+1
      ICLASS(NCLASS)=I
      JTYPE(I)=NCLASS
  560 CONTINUE
C
C     CALCULATE ELEMENT STIFFNESSES FOR EACH SEGMENT TYPE
C
  561 REWIND K3
      KJ=0
      DO 562 I=1,9
  562 CHK(I,1)=0.0
      DO 570 I=1,NCLASS
      IJ=ICLASS(I)
      KK=IJ*NEL-NEL
      GG=0.0
      IF(IL.EQ.2) GG=G(IJ)-G(IJ+1)
      DO 569 J=1,NEL
      CALL ELCOOR(NSECT,NMAX,TX,TY,TZ,IJ,NPI(J),NPJ(J),0,0,IL,1,XE,YE,ZE
     *)
      JJ=ITYPE(IJ,J)
      IF(JJ.NE.KJ) GO TO 564
      DO 563 K=1,3
      XX=ABS(XE(K+1)-XE(1)-CHK(K,1))+ABS(YE(K+1)-YE(1)-CHK(K,2))
     *                              +ABS(ZE(K+1)-ZE(1)-CHK(K,3))
      IF(XX.GT..00001) GO TO 564
  563 CONTINUE
      GO TO 567
  564 KJ=JJ
      DO 565 K=1,3
      CHK(K,1)=XE(K+1)-XE(1)
      CHK(K,2)=YE(K+1)-YE(1)
  565 CHK(K,3)=ZE(K+1)-ZE(1)
      DO 566 K=1,6
      CC(K)=C(JJ,K)
  566 DD(K)=C(JJ,K+6)
      CALL TOSTIF(JFLAG)
      IF(JFLAG.EQ.1) RETURN
      IF(IC5.EQ.0) GO TO 567
      PRINT 2016, J,IJ,JJ
      CALL PRST(T,24,24)
  567 IF(IFLAG) PRINT 2048, J,IJ,AVE
      DO 568 K=1,4
  568 AREA(K,KK+J)=AA(K)
  569 WRITE (K3) T,XT
  570 CONTINUE
C
C     COPY STIFFNESSES FOR SIMILAR SEGMENTS
C
      JJ=0
      REWIND K2
      REWIND K3
```

```
      DO 580 I=1,NSEG
      K=JTYPE(I)
      KK=K*NEL-NEL
      IF(KK-JJ) 572,576,573
  572 REWIND K3
      IF(KK.EQ.0) GO TO 576
      JK=KK
      GO TO 574
  573 JK=KK-JJ
  574 DO 575 J=1,JK
  575 READ (K3) T
  576 DO 577 J=1,NEL
      READ (K3) T,XT
      WRITE (K1) XT
  577 WRITE (K2) T
      JJ=KK+NEL
  580 CONTINUE
C
      DO 590 I=1,NSEG
      IK=JTYPE(I)
      JK=ICLASS(IK)*NEL-NEL
      II=I*NEL-NEL
      DO 590 J=1,NEL
      DO 585 K=1,4
  585 AREA(K,II+J)=AREA(K,JK+J)
  590 CONTINUE
C
C***********************************************************************
C         7) CALCULATE DIAPHRAGM STIFFNESS AND STORE ON TAPE K3
C***********************************************************************
C
C        SET UP IDENTIFICATION ARRAY  IFD
C           IFD(I)=0  IF NEITHER FRAME NOR DIAPHRAGM AT SECTION I
C           IFD(I)=1  IF ONLY FRAME AT SECTION I
C           IFD(I)=2  IF ONLY DIAPHRAGM AT SECTION I
C           IFD(I)=3  IF FRAME AND DIAPHRAGM AT SECTION I
C
      DO 660 I=1,NSECT
  660 IFD(I)=0
      IF(NFRAME.LE.0) GO TO 675
      DO 665 I=1,NFRAME
      II=MIN0(NSFI(I),NSFJ(I))
  665 IFD(II)=IFD(II)+1
      DO 670 I=1,NSECT
      IF(IFD(I).GT.0) IFD(I)=1
  670 CONTINUE
  675 IF(NDIAPH.LE.0) GO TO 700
      DO 680 I=1,NDIAPH
      J=NSD(I)
  680 IFD(J)=IFD(J)+2
C
C        CALCULATE DIAPHRAGM STIFFNESS
C
  700 IF(NDIAPH+NFRAME.EQ.0) GO TO 820
      REWIND K3
      KJ=0
      DO 701 I=1,9
  701 CHK(I,1)=0.0
      DO 800 I=1,NSECT
      II=IFD(I)
```

```
      IF(II.EQ.0) GO TO 800
      IF(II.EQ.1) GO TO 720
      DO 715 J=1,NDE
      CALL ELCOOR(NSECT,NMAX,TX,TY,TZ,I,NPD(1,J),NPD(2,J),NPD(3,J),
     *  NPD(4,J),IL,2,XE,YE,ZE)
      JJ=NDTY(J)
      IF(NDS.LE.0) GO TO 703
      DO 702 K=1,NDS
      IF(NDSD(K).NE.I.OR.NDSE(K).NE.J) GO TO 702
      JJ=NDST(K)
      GO TO 703
  702 CONTINUE
  703 IF(JJ.NE.KJ) GO TO 707
      DO 705 K=1,3
      XX=ABS(XE(K+1)-XE(1)-CHK(K,1))+ABS(YE(K+1)-YE(1)-CHK(K,2))
     *                              +ABS(ZE(K+1)-ZE(1)-CHK(K,3))
      IF(XX.GT..00001) GO TO 707
  705 CONTINUE
      GO TO 712
  707 KJ=JJ
      DO 708 K=1,3
      CHK(K,1)=XE(K+1)-XE(1)
      CHK(K,2)=YE(K+1)-YE(1)
  708 CHK(K,3)=ZE(K+1)-ZE(1)
      DO 710 K=1,6
      CC(K)=C(JJ,K)
  710 DD(K)=C(JJ,K+6)
      CALL TOSTIF (JFLAG)
      IF(JFLAG.EQ.1) RETURN
  712 IF(IFLAG) PRINT 2049, J,I,AVE
      WRITE (K1) XT
  715 WRITE (K3) T
C
C**********************************************************************
C         B) CALCULATE FRAME STIFFNESS AND STORE ON TAPE K3
C**********************************************************************
C
  720 IF(II.EQ.2) GO TO 800
      DO 760 J=1,NFRAME
      KI=NSFI(J)
      KJ=NSFJ(J)
      IF(KI.NE.I.AND.KJ.NE.I) GO TO 760
      IF(KI.LT.I.OR.KJ.LT.I) GO TO 760
      JJ=NFSC(J)
      KK=NFTY(J)
      CALL ELCOOR(NSECT,NMAX,TX,TY,TZ,I,NFI(J),NFJ(J),KI,KJ,IL,3,
     *  XE,YE,ZE)
      DO 750 K=1,3
      ECI(K)=EI(K,JJ)
      ECJ(K)=EJ(K,JJ)
  750 ECK(K)=EK(K,JJ)
      IF(IL.EQ.2) GO TO 754
      S=SIN(G(KI))
      CO=COS(G(KI))
      GG=X(KI)+ECK(1)*CO-ECK(2)*S
      ECK(2)=Y(KI)+ECK(1)*S+ECK(2)*CO
      ECK(1)=GG
  754 G1=GAY(KK)
      G2=GAZ(KK)
      AVE=G(KI)
```

```
      GG=G(KJ)
      K=0
      IF(AVE,EQ.GG.OR.IL,EQ.1) GO TO 755
      K=1
      IF(KI.GT.KJ) K=2
  755 CALL ELSTIF (EA(KK),EIY(KK),EIZ(KK),GJX(KK),G1,G2,1,K,IL,XX,NLC)
      WRITE (K3) ST
      IF(IC5.EQ.0) GO TO 760
      PRINT 2017, J
      CALL PRST (ST,12,12)
  760 CONTINUE
  800 CONTINUE
  820 IF(NDIAPH.EQ.0) NDE=0
  850 RETURN
C
C***********************************************************************
C       9) FORMAT STATEMENTS
C***********************************************************************
C
 2009 FORMAT(1H1,31H GENERATED REFERENCE POINT DATA//116H  REF,POINT  GL
     *OBAL X-COORD   GLOBAL Y-COORD       GAMMA(DEGR,)          REF,LINE S
     *EGM,      LENGTH        ALPHA(DEGR,))
 2010 FORMAT(I7,3F17,4,I21,2F17,4)
 2011 FORMAT(///45H GAMMA = ANGLE BETW,SECTION AND GLOBAL Y-AXIS/
     * 54H ALPHA = ANGLE BETW,REF,LINE SEGMENT AND GLOBAL X-AXIS)
 2012 FORMAT(1H1,25H BOUNDARY CONDITION ARRAY//23H    0 - FORCE SPECIFIE
     *D/23H    1 - DISPL SPECIFIED////)
 2013 FORMAT(6I5)
 2014 FORMAT(////33H SPECIFIED NON-ZERO DISPLACEMENTS//33H  EQUATION NO,
     *    DISPL, MAGNITUDE)
 2015 FORMAT(I9,F20,4)
 2016 FORMAT(1H1,25H STIFFNESS OF ELEMENT NO,,I5,17H   IN SEGMENT NO,,
     *  I5,13H    (TYPE NO,,I3,2H )//)
 2017 FORMAT(1H2,31H STIFFNESS OF FRAME ELEMENT NO,,I5//)
 2041 FORMAT(1H1,41H FIXED GLOBAL COORDINATES OF NODAL POINTS/)
 2042 FORMAT(/12H SECTION NO,,I4,49H    POINT NO,      X-COORD     Y-COOR
     *D      Z-COORD)
 2043 FORMAT(16X,I9,4X,3F12,4)
 2044 FORMAT(1H1,39H TRAVELLING COORDINATES OF NODAL POINTS//,37X,
     * 10H SECTION I,42X,12H SECTION I+1)
 2045 FORMAT(/12H SEGMENT NO,,I4,102H    POINT NO,     X-COORD     Y-COO
     *RD      Z-COORD          POINT NO,     X-COORD     Y-COORD     Z-COOR
     *D)
 2046 FORMAT(16X,I9,4X,3F12,4,I13,4X,3F12,4)
 2047 FORMAT(65X,I13,4X,3F12,4)
 2048 FORMAT(//8H ELEMENT,I6,13H   IN SEGMENT,I6,23H   FORMS WARPED SURF
     *ACE/63H  LEAST SQUARE FIT PLANE REQUIRES AVERAGE AMOUNT OF SHIFTIN
     *G OF,F10,6)
 2049 FORMAT(//18H DIAPHRAGM ELEMENT,I6,13H   AT SECTION,I6.23H   FORMS 
     *WARPED SURFACE/63H  LEAST SQUARE FIT PLANE REQUIRES AVERAGE AMOUNT
     * OF SHIFTING OF,F10,6)
 2050 FORMAT(1H1,32H TOTAL NUMBER OF EQUATIONS      =,I6/
     *            33H  NUMBER OF BLOCKS             =,I6/
     *            33H  NUMBER OF EQUATIONS PER BLOCK =,I6/
     *            33H  BANDWIDTH                    =,I6)
C
      END


      FUNCTION ALPHA(X1,X2,Y1,Y2)
```

```
      XT=X2-X1
      YT=Y2-Y1
      IF(XT.NE.0.0) GO TO 10
      ALPHA=SIGN(YT,1,157079632679489)
      RETURN
   10 ALPHA=ATAN(YT/XT)
      RETURN
      END


      SUBROUTINE ELCOOR(NS,NM,TX,TY,TZ,I,NI,NJ,NK,NL,IL,ID,X,Y,Z)
C
C***********************************************************************
C     THIS SUBROUTINE EXTRACTS FROM THE NODAL POINT ARRAYS THE GLOBAL
C     COORDINATES FOR PLATES, DIAPHRAGM, AND FRAME ELEMENTS.
C***********************************************************************
C
      DIMENSION TX(NS,NM,1),TY(NS,NM,1),TZ(NS,NM,1),X(4),Y(4),Z(4)
C
C        PLATE AND DIAPHRAGM ELEMENTS
C
      GO TO (10,20,30), ID
   10 JI=NJ
      JJ=NI
      IF(IL.EQ.1) GO TO 15
      II=I
      IJ=2
      GO TO 25
   15 II=I+1
      IJ=1
      GO TO 25
   20 JI=NK
      JJ=NL
      II=I
      IJ=1
   25 X(1)=TX(I,NI,1)
      X(2)=TX(I,NJ,1)
      X(3)=TX(II,JI,IJ)
      X(4)=TX(II,JJ,IJ)
      Y(1)=TY(I,NI,1)
      Y(2)=TY(I,NJ,1)
      Y(3)=TY(II,JI,IJ)
      Y(4)=TY(II,JJ,IJ)
      Z(1)=TZ(I,NI,1)
      Z(2)=TZ(I,NJ,1)
      Z(3)=TZ(II,JI,IJ)
      Z(4)=TZ(II,JJ,IJ)
      RETURN
C
C        FRAME ELEMENTS
C
   30 II=NK
      IJ=NL
      JI=1
      JJ=1
      IF(IL.EQ.1) GO TO 40
      IF(NK.GT.NL) II=NL
      IJ=II
      IF(NK.LT.NL) JJ=2
      IF(NK.GT.NL) JI=2
```

```
   40 X(1)=TX(II,NI,JI)
      X(2)=TX(IJ,NJ,JJ)
      Y(1)=TY(II,NI,JI)
      Y(2)=TY(IJ,NJ,JJ)
      Z(1)=TZ(II,NI,JI)
      Z(2)=TZ(IJ,NJ,JJ)
      RETURN
      END


      SUBROUTINE PRST (T,M,NN)
      DIMENSION T(NN,1),F(3),FM(6)
      DATA  F/6H(      ,6H6E15,4,6H)      /,
     *      FM/6H(      ,6H(15X, ,6H(30X, ,6H(45X, ,6H(60X, ,6H(75X, /
      MM=M/6
      DO 650 K=1,MM
      KK=K*6-6
      IF(K,EQ,1) GO TO 615
      KI=K-1
      IA=KK+1
      IE=KK+6
      DO 610 L=1,KI
      LL=L*6-6
      DO 605 N=1,6
      NR=LL+N
  605 PRINT 1001, (T(NR,NC), NC=IA,IE)
  610 CONTINUE
  615 DO 630 L=1,6
      NR=KK+L
      IA=NR
      IE=KK+6
      F(1)=FM(L)
  630 WRITE(6,F) (T(NR,NC), NC=IA,IE)
  650 PRINT 1008
 1001 FORMAT(6E15,4)
 1008 FORMAT(//////)
      RETURN
      END
```

```
      SUBROUTINE TOSTIF (JF)
C
C***********************************************************************
C     THIS SUBROUTINE ASSEMBLES THE 24*24 GLOBAL STIFFNESS OF A GENERAL
C     QUADRILATERAL ELEMENT WITH ARBITRARY ORIENTATION IN SPACE AND  3
C     TRANSLATIONAL AND 3 ROTATIONAL DEGREES OF FREEDOM PER CORNER NODE
C     AND AN ELASTIC ANISOTROPIC MATERIAL LAW, IF THE 4 NODAL POINTS
C     FORM A WARPED SURFACE, THE *BEST* PLANE IS FITTED INBETWEEN,
C
C                              - INPUT -
C    X(4),Y(4),Z(4) - GLOBAL COORDINATES OF THE 4 CORNER POINTS
C    C(6)       - MATERIAL LAW RELATING IN-PLANE STRESSES AND STRAINS
C    D(6)       - MATERIAL LAW RELATING PLATE MOMENTS AND CURVATURES
C    G          - ANGLE BETWEEN EDGE I-J AND K-L (NEEDED ONLY IF
C                 STIFFNESS IS TO BE EXPRESSED IN TRAVELLING COORDINATES)
C
C                              - OUTPUT -
C    T(24,24) - COMPLETE ELEMENT STIFFNESS IN GLOBAL COORDINATES
C    IFLAG      - INDICATOR FLAG
C                 .FALSE. = THE 4 NODAL POINTS FORM A PLANE
C                 .TRUE.  = THE 4 NODAL POINTS FORM A WARPED SURFACE
C    AVE        - AVERAGE SHIFTING OF NODAL POINTS REQUIRED TO HAVE THEM
C                 FORM A PLANE  (FOR IFLAG=.TRUE. ONLY)
C    A(4)       - SUB-QUADRILATERAL AREAS OF ELEMENT
C    JF         - ERROR FLAG, SET EQUAL TO 1 FOR BAD JACOBIAN
C    XX(5),YY(5) - ELEMENT COORDINATES OF 4 CORNER POINTS AND MIDPOINT
C    A11,A12,...,A33 - DIRECTION COSINES OF ELEMENT COORDINATE AXES
C   ((S(I,J),J=13,19),I=1,19) - PORTION OF PLATE BENDING STIFFNESS
C                 NEEDED IN INTFOR TO FIND INTERNAL NODAL DISPLACEMENTS
C***********************************************************************
C
C     COMMON, DIMENSION, AND EQUIVALENCE STATEMENTS
C
      COMMON/STIFF/T(24,24),X(4),Y(4),Z(4),C(6),D(9),A(4),IFLAG,AVE,G,
     *  XT(167)
      DIMENSION XX(5),YY(5),S(19,19),SP(12,12),DD(3,3)
      EQUIVALENCE (C(1),D1),(C(2),D2),(C(3),D3),(C(4),D4),(C(5),D5),
     *  (C(6),D6),(S,SP),(D,DD),         (XT(149),XX),(XT(154),YY),
     *  (XT(159),A11),(XT(160),A21),(XT(161),A31),(XT(162),A12),
     *  (XT(163),A22),(XT(164),A32),(XT(165),A13),(XT(166),A23),
     *  (XT(167),A33)
C     NOTE  -  ARRAY XT(167) IS EQUIVALENCED TO THE ARRAYS/VARIABLES
C    ((S(I,J),I=1,19),J=13,19),C(6),D(9),XX(5),YY(5),A11,A12,...,A33
C    IN THAT ORDER
      LOGICAL IFLAG
C
C     INITIALIZATION
C
      DO 10 I=1,24
      DO 10 J=1,24
   10 T(I,J)=0.0
      IF(C(1).EQ.0..AND.D(1).EQ.0.) RETURN
C
C     FIND DIRECTION COSINES OF ELEMENT COORDINATE AXES
C
      XL=SQRT((X(2)-X(1))**2+(Y(2)-Y(1))**2+(Z(2)-Z(1))**2)
      A11=(X(2)-X(1))/XL
      A12=(Y(2)-Y(1))/XL
      A13=(Z(2)-Z(1))/XL
```

```
      XL=SQRT((X(4)-X(1))**2+(Y(4)-Y(1))**2+(Z(4)-Z(1))**2)
      A21=(X(4)-X(1))/XL
      A22=(Y(4)-Y(1))/XL
      A23=(Z(4)-Z(1))/XL
      A31=A12*A23-A13*A22
      A32=A13*A21-A11*A23
      A33=A11*A22-A12*A21
      XL=SQRT(A31*A31+A32*A32+A33*A33)
      A31=A31/XL
      A32=A32/XL
      A33=A33/XL
      A21=A32*A13-A33*A12
      A22=A33*A11-A31*A13
      A23=A31*A12-A32*A11
      XL=SQRT(A21*A21+A22*A22+A23*A23)
      A21=A21/XL
      A22=A22/XL
      A23=A23/XL
C
C     CHECK IF THE FOUR CORNER NODES FORM A PLANE SURFACE
C
      S1=X(3)-X(1)
      S2=Y(3)-Y(1)
      S3=Z(3)-Z(1)
      XL=SQRT(S1**2+S2**2+S3**2)
      DET=ABS(S1*A31+S2*A32+S3*A33)/XL
      IFLAG=.FALSE.
      IF(DET.LE..0000001) GO TO 20
C
C     FIND THE PLANE MINIMIZING THE SUM OF NORMAL DISTANCES SQUARED
C
      IFLAG=.TRUE.
      CALL FITPLN  (4,X,Y,Z,A3,B3,C3,AVF)
C
C     FIND MODIFIED DIRECTION COSINES OF ELEMENT COORDINATE AXES
C
      A31=-A3
      A32=-B3
      A33=-C3
   15 XL=SQRT((X(2)-X(1))**2+(Y(2)-Y(1))**2+(Z(2)-Z(1))**2)
      A11=(X(2)-X(1))/XL
      A12=(Y(2)-Y(1))/XL
      A13=(Z(2)-Z(1))/XL
      A21=A32*A13-A33*A12
      A22=A33*A11-A31*A13
      A23=A31*A12-A32*A11
      XL=SQRT(A21*A21+A22*A22+A23*A23)
      A21=A21/XL
      A22=A22/XL
      A23=A23/XL
C
C     FIND NODAL POINTS IN ELEMENT COORDINATES
```

```
   27 XL=XX(1)+XX(2)+XX(3)+XX(4)
      CALL PAREA (XL,YL,XX,YY,A)
C
C     CALCULATE PLANE STRESS STIFFNESS IN ELEMENT COORDINATES
C     AND STORE IT IN THE TOTAL ELEMENT STIFFNESS
C
      CALL Q12R12 (XX,YY,D1,D2,D3,D4,D5,D6,SP,JF)
      DO 30 I=1,4
      II=I*6-6
      DO 30 J=1,4
      JJ=J*6-6
      T(II+1,JJ+1)=SP(I   ,J   )
      T(II+1,JJ+2)=SP(I   ,J+4)
      T(II+1,JJ+6)=SP(I   ,J+8)
      T(II+2,JJ+1)=SP(I+4,J   )
      T(II+2,JJ+2)=SP(I+4,J+4)
      T(II+2,JJ+6)=SP(I+4,J+8)
      T(II+6,JJ+1)=SP(I+8,J   )
      T(II+6,JJ+2)=SP(I+8,J+4)
   30 T(II+6,JJ+6)=SP(I+8,J+8)
C
C     CALCULATE PLATE BENDING STIFFNESS IN ELEMENT COORDINATES
C     AND ADD IT INTO THE TOTAL ELEMENT STIFFNESS
C
      D(9)=D(3)
      D(8)=D(6)
      D(7)=D(5)
      D(5)=D(2)
      D(3)=D(7)
      D(2)=D(4)
      CALL SPLATE (XX,YY,D,S)
      DO 35 I=1,4
      II=I*6-4
      IK=I*3-3
      DO 35 J=1,4
      JJ=J*6-4
      JK=J*3-3
      DO 35 K=1,3
      DO 35 L=1,3
   35 T(II+K,JJ+L)=S(IK+K,JK+L)
C
      DO 37 I=1,24
      DO 37 J=1,24
   37 T(J,I)=T(I,J)
C
C     TRANSFORMATION TO GLOBAL COORDINATES
C
      DO 40 I=1,24
      DO 40 J=1,22,3
      S1=T(I,J)*A11+T(I,J+1)*A21+T(I,J+2)*A31
      S2=T(I,J)*A12+T(I,J+1)*A22+T(I,J+2)*A32
      T(I,J+2)=T(I,J)*A13+T(I,J+1)*A23+T(I,J+2)*A33
```

```
      T(I,J)=S1
   40 T(I,J+1)=S2
C
      DO 50 I=1,24
      DO 50 J=1,22,3
      S1=T(J,I)*A11+T(J+1,I)*A21+T(J+2,I)*A31
      S2=T(J,I)*A12+T(J+1,I)*A22+T(J+2,I)*A32
      T(J+2,I)=T(J,I)*A13+T(J+1,I)*A23+T(J+2,I)*A33
      T(J,I)=S1
   50 T(J+1,I)=S2
C
C     TRANSFORM FAR END STIFFNESS TO CONFORM WITH TRAVELLING COORDINATES
C
      IF(G.EQ.0.) GO TO 55
      SG=SIN(G)
      CG=COS(G)
      DO 52 I=13,22,3
      DO 51 J=1,24
      S1= T(J,I)*SG+T(J,I+1)*CG
      T(J,I)=T(J,I)*CG-T(J,I+1)*SG
   51 T(J,I+1)=S1
   52 CONTINUE
C
      DO 54 I=13,22,3
      DO 53 J=1,24
      S1= T(I,J)*SG+T(I+1,J)*CG
      T(I,J)=T(I,J)*CG-T(I+1,J)*SG
   53 T(I+1,J)=S1
   54 CONTINUE
C
   55 DO 60 I=1,6
   60 XT(133+I)=C(I)
      DO 65 I=1,9
   65 XT(139+I)=D(I)
      K=0
      DO 70 J=13,19
      DO 70 I=1,19
      K=K+1
   70 XT(K)=S(I,J)
      RETURN
      END


      SUBROUTINE SPLATE (X,Y,CM,S)
C
C*********************************************************************
C     THIS SUBROUTINE ASSEMBLES THE STIFFNESS MATRIX OF A QUADRILATERAL
C     PLATE BENDING ELEMENT FORMED OUT OF FOUR LCCT-11 TRIANGLES,
C
C                                - INPUT -
C     X(4),Y(4) - COORDINATES OF THE FOUR CORNER POINTS
C     CM(3,3)   - CONSTITUTIVE MATERIAL MATRIX
C
C                                - OUTPUT -
C     S(19,19)  - ELEMENT STIFFNESS MATRIX
C                 (ONLY THE FIRST 12*12 ELEMENTS ARE NEEDED FOR THE
C                 STRUCTURE ASSEMBLY, BUT THE 19*7 RECTANGLE HAS TO BE
C                 SAVED FOR DETERMINATION OF INTERNAL STRESSES)
C*********************************************************************
C
```

```
C      COMMON, DIMENSION, EQUIVALENCE, AND DATA STATEMENTS
C
       COMMON/TRIARG/B(3),A(3),CMT(3,3),BMT(3,3),ST(12,12)
       DIMENSION X(5),Y(5),CM(3,3),S(19,19),LOC(11,4),IPERM(4),NC(3)
       DATA IPERM /2,3,4,1/
       DATA LOC   /1, 2, 3, 4, 5, 6,13,14,15,17,16,
     1              4, 5, 6, 7, 8, 9,13,14,15,18,17,
     2              7, 8, 9,10,11,12,13,14,15,19,18,
     3             10,11,12, 1, 2, 3,13,14,15,16,19/
C
C      FIND COORDINATES OF CENTER POINT
C
       X(5)=.25*(X(1)+X(2)+X(3)+X(4))
       Y(5)=.25*(Y(1)+Y(2)+Y(3)+Y(4))
C
C      PREPARE FOR EACH TRIANGLE THE CALL OF SUBROUTINE SLCCT
C
       NTR=4
       NBF=11
       NEF=12
       NIF=7
       NDF=NBF
       NTF = NEF + NIF
       NC(3) = NBF - 6
       DO 150 I=1,361
   150 S(I,1) = 0.
       DO 260 N=1,NTR
       M = IPERM(N)
       NC(1) = N
       NC(2) = M
       L = NC(3)
       A(1) = X(L)-X(M)
       A(2) = X(N)-X(L)
       A(3) = X(M)-X(N)
       B(1) = Y(M)-Y(L)
       B(2) = Y(L)-Y(N)
       B(3) = Y(N)-Y(M)
       DO 200  I = 1,3
       DO 200  J = 1,3
   200 CMT(I,J) = CM(I,J)
       CALL SLCCT (NBF)
C
C      ASSEMBLE TRIANGLES TO OBTAIN QUADRILATERAL STIFFNESS
C
       DO 250  I = 1,NDF
       K = LOC(I,N)
       ST(I,11) = -ST(I,11)
       ST(11,I) = -ST(11,I)
       DO 250  J = 1,I
       L = LOC(J,N)
       C = S(K,L) + ST(I,J)
       S(K,L) = C
       S(L,K) = C
   250 CONTINUE
   260 CONTINUE
C
C      ELIMINATE INTERNAL DEGREES OF FREEDOM BY STATIC CONDENSATION
C
       DO 420  M = 1,NIF
       L = NTF - M
```

```
      N = L + 1
      PIVOT = S(N,N)
      IF (PIVOT.LE.0.0) GO TO 420
      DO 400  I = 1,L
      C = S(I,N)/PIVOT
      S(I,N) = C
      DO 400  J = I,L
      S(I,J) = S(I,J) - C*S(N,J)
  400 S(J,I) = S(I,J)
  420 CONTINUE
      RETURN
      END


      SUBROUTINE SLCCT  (NBF)
C
C*********************************************************************
C     STIFFNESS SUBROUTINE FOR COMPATIBLE TRIANGULAR PLATE ELEMENT
C     WITH 'NBF' BENDING D.F. (NBF=12,11,10,9) - RIGHT-HAND SYSTEM -
C     ORTHOTROPIC ELASTIC MATERIAL
C*********************************************************************
C
      COMMON/TRIARG/B(3),A(3),CMT(3,3),BMT(3,3),ST(12,12)
      DIMENSION  P(21,12), H(21), U(21), Q(3,6), HT(3), IPERM(3),
     1                             TX(3), TY(3), NKN(4,3), NSN(4,3)
      EQUIVALENCE (CM11,CMT(1)),(CM12,CMT(2)),(CM13,CMT(3)),
     1 (CM22,CMT(5)),(CM23,CMT(6)),(CM33,CMT(9))
      DATA IPERM/2,3,1/, NKN/2,5,3,6, 8,2,9,3, 5,8,6,9/,
     1 NSN/2,3,5,6, 3,1,6,4, 1,2,4,5/
C
C        INITIALIZATION
C
      NDF = NBF
      AREA = A(3)*B(2)-A(2)*B(3)
      FAC = AREA/72.
      DO 150  I = 1,3
      J = IPERM(I)
      K = IPERM(J)
      X = A(I)**2+B(I)**2
      U(I) = -(A(I)*A(J)+B(I)*B(J))/X
      X = SQRT(X)
      HT(I) =  4.0*AREA/X
      TY(I) = -0.5*B(I)/X
      TX(I) =  0.5*A(I)/X
      A1 = A(I)/AREA
      A2 = A(J)/AREA
      B1 = B(I)/AREA
      B2 = B(J)/AREA
      Q(1,I)   = B1*B1
      Q(2,I)   = A1*A1
      Q(3,I)   = 2.*A1*B1
      Q(1,I+3) = 2.*B1*B2
      Q(2,I+3) = 2.*A1*A2
  150 Q(3,I+3) = 2.*(A1*B2+A2*B1)
C
C       FORMATION OF CURVATURE MATRIX  P(21,12)
C
      DO 200  I = 1,3
      J = IPERM(I)
      K = IPERM(J)
```

```
II = 3*I
JJ = 3*J
KK = 3*K
A1 = A(I)
A2 = A(J)
A3 = A(K)
B1 = B(I)
B2 = B(J)
B3 = B(K)
U1 = U(I)
U2 = U(J)
U3 = U(K)
W1 = 1.-U1
W2 = 1.-U2
W3 = 1.-U3
B1D = 2.*B1
B2D = 2.*B2
B3D = 2.*B3
A1D = 2.*A1
A2D = 2.*A2
A3D = 2.*A3
C21 = B1-B3*U3
C22 = -B1D+B2*W2+B3*U3
C31 = A1-A3*U3
C32 = -A1D+A2*W2+A3*U3
C51 = B3*W3-B2
C52 = B2D-B3*W3-B1*U1
C61 = A3*W3-A2
C62 = A2D-A3*W3-A1*U1
C81 = B3-B2D-B2*U2
C82 = B1D-B3+B1*W1
C91 = A3-A2D-A2*U2
C92 = A1D-A3+A1*W1
DO 200  N = 1,3
L = 6*(I-1) + N
Q11 = Q(N,I)
Q22 = Q(N,J)
Q33 = Q(N,K)
Q12 = Q(N,I+3)
Q23 = Q(N,J+3)
Q31 = Q(N,K+3)
Q2333 = Q23-Q33
Q3133 = Q31-Q33
P(L   ,II-2) = 6.*(-Q11+W2*Q33+U3*Q2333)
P(L   ,II-1) = C21*Q23+C22*Q33-B3D*Q12+B2D*Q31
P(L   ,II  ) = C31*Q23+C32*Q33-A3D*Q12+A2D*Q31
P(L   ,JJ-2) = 6.*(Q22+W3*Q2333)
P(L   ,JJ-1) = C51*Q2333+B3D*Q22
P(L   ,JJ  ) = C61*Q2333+A3D*Q22
P(L   ,KK-2) = 6.*(1.+U2)*Q33
P(L   ,KK-1) = C81*Q33
P(L   ,KK  ) = C91*Q33
P(L   ,I+9 ) = 0.
P(L   ,J+9 ) = HT(J)*Q33
P(L   ,K+9 ) = HT(K)*Q2333
P(L+3 ,II-2) = 6.*(Q11+U3*Q3133)
P(L+3 ,II-1) = C21*Q3133-B3D*Q11
P(L+3 ,II  ) = C31*Q3133-A3D*Q11
P(L+3 ,JJ-2) = 6.*(-Q22+U1*Q33+W3*Q3133)
P(L+3 ,JJ-1) = C51*Q31+C52*Q33+B3D*Q12-B1D*Q23
```

```
      P(L+3 ,JJ  ) = C61*Q31+C62*Q33+A3D*Q12-A1D*Q23
      P(L+3 ,KK-2) = 6.*(1.+W1)*Q33
      P(L+3 ,KK-1) = C82*Q33
      P(L+3 ,KK  ) = C92*Q33
      P(L+3 ,I+9 ) = HT(I)*Q33
      P(L+3 ,J+9 ) = 0.
      P(L+3 ,K+9 ) = HT(K)*Q3133
      P(N+18,II-2) = 2.*(Q11+U3*Q12+W2*Q31)
      P(N+18,KK-1) = ((B1D-B2D)*Q33+C82*Q23+C81*Q31)/3.
      P(N+18,KK  ) = ((A1D-A2D)*Q33+C92*Q23+C91*Q31)/3.
  200 P(N+18,K+9 ) = HT(K)*Q12/3.
C
C        STATIC CONDENSATION OF MIDSIDE DEGREES OF FREEDOM
C
      NK = 12 - NBF
      IF (NK.LE.0)  GO TO 300
      DO 220  N = 1,NK
      K = 13 - N
      DO 220  L = 1,4
      J = NKN(L,N)
      IF (L.LE.2)  C = TX(K-9)
      IF (L.GT.2)  C = TY(K-9)
      DO 220  I = 1,21
  220 P(I,J) = P(I,J) + C*P(I,K)
C
C        FORMATION OF MOMENT VECTOR  U(21)
C
  300 DO 400  J = 1,NDF
      DO 340  L = 1,3
      KK = L + 18
      II = L - 6
      P3 = P(KK,J)
      H(KK) = 0.
      DO 340  N = 1,3
      M = IPERM(N)
      II = II + 6
      JJ = II + 3
      P1 = P(II,J)
      P2 = P(JJ,J)
      SUM = P1 + P2 + P3
      T1 = SUM + P1
      T2 = SUM + P2
      T3 = SUM + P3
      H(II) = T1
      H(JJ) = T2
  340 H(KK) = T3 + H(KK)
      DO 360  N = 1,19,3
      U(N)   = CM11*H(N) + CM12*H(N+1) + CM13*H(N+2)
      U(N+1) = CM12*H(N) + CM22*H(N+1) + CM23*H(N+2)
  360 U(N+2) = CM13*H(N) + CM23*H(N+1) + CM33*H(N+2)
C
C        FORMATION OF STIFFNESS MATRIX  ST(12,12)
C
      DO 400  I = 1,J
      X = 0.
      DO 380  N = 1,21
  380 X = X + U(N)*P(N,I)
      ST(I,J) = X*FAC
  400 ST(J,I) = ST(I,J)
      RETURN
```

```
      END

      SUBROUTINE FITPLN  (NSP,X,Y,Z,A3,B3,C3,AVE)
C
C*****************************************************************************
C     THIS SUBROUTINE COMPUTES THE 3 DIRECTION COSINES OF THE NORMAL TO
C     THE PLANE WHICH IS THE LEAST SQUARE FIT FOR 'NSP' SPACE POINTS.
C     THE 3 DIRECTION COSINES ARE THE ELEMENTS OF THE NORMALIZED
C     EIGENVECTOR CORRESPONDING TO THE SMALLEST EIGENVALUE OF THE
C     (3 X 3) NORMAL MATRIX CONSTRUCTED WITH THE COORDINATES OF THE
C     'NSP' POINTS, AND ARE OBTAINED BY INVERSE ITERATION.
C
C                                - INPUT -
C     NSP         - NUMBER OF SPACE POINTS (MUST BE AT LEAST =4)
C     X(I),Y(I),Z(I),I=1,NSP - GLOBAL COORDINATES OF NSP SPACE POINTS
C                   NOT LYING IN A PLANE
C
C                                - OUTPUT -
C     A3,B3,C3 - DIRECTION COSINES OF PLANE NORMAL, REFERRED TO GLOBAL
C                   X,Y,Z COORDINATE SYSTEM
C     X(I),Y(I),Z(I),I=1,NSP - GLOBAL COORDINATES OF SHIFTED SPACE
C                   POINTS NOW LYING IN A PLANE
C     AVE         - AVERAGE AMOUNT OF SHIFTING
C*****************************************************************************
C
      DIMENSION X(1), Y(1), Z(1)
      REAL M12, M13, M23
      PRECS = 1.0E-07
C
C     FIND CENTROID POINT (ORIGIN FOR SHIFTED GLOBAL COORDINATE SYSTEM)
C
      XC = 0.
      YC = 0.
      ZC = 0.
      XSP = NSP
      DO 100  I = 1,NSP
      XC = XC + X(I)/XSP
      YC = YC + Y(I)/XSP
  100 ZC = ZC + Z(I)/XSP
C
C     FORM MATRIX OF EIGENVALUE PROBLEM
C
      XX = 0.
      YY = 0.
      ZZ = 0.
      XY = 0.
      YZ = 0.
      ZX = 0.
      DO 110  I = 1,NSP
      X1=X(I)-XC
      Y1=Y(I)-YC
      Z1=Z(I)-ZC
      XX = XX+X1**2
      YY = YY+Y1**2
      ZZ = ZZ+Z1**2
      XY = XY+X1*Y1
      YZ = YZ+Y1*Z1
  110 ZX = ZX+Z1*X1
C
```

```
C     INITIALIZE ITERATION
C
      NIT = 0
      EPS = 1.0
      A3 = 1.0
      B3 = 1.0
      C3 = 1.0
      S=XX+YY+ZZ
      EV=-S*PRECS
C
C     INVERSE ITERATION ON EIGENVECTOR WITH DIRECTION COSINES
C
  120 U11 = XX - EV
      U12 = XY
      U13 = ZX
      M12 = XY/U11
      M13 = ZX/U11
      U22 = YY - EV - M12*U12
      U23 = YZ - M12*U13
      M23 = (YZ - M13*U12)/U22
      U33 = ZZ - EV - M13*U13 - M23*U23
      IF (U33.GT.0.)  GO TO 140
      EV = EV - 0.001/S
      GO TO 120
  140 IF (NIT.EQ.0)  GO TO 180
  160 B3 = B3 - M12*A3
      C3 = C3 - M13*A3 - M23*B3
  180 C3 = C3/U33
      B3 = (B3 - U23*C3)/U22
      A3 = (A3 - U12*B3 - U13*C3)/U11
      S = SQRT(A3**2+B3**2+C3**2)
      A3 = A3/S
      B3 = B3/S
      C3 = C3/S
      NIT = NIT + 1
      IF (NIT.LT.2)  GO TO 190
      EPS = AMAX1 (ABS(A3-AP),ABS(B3-BP),ABS(C3-CP))
      IF (EPS.LT.PRECS.OR.NIT.GT.25)   GO TO 250
  190 AP = A3
      BP = B3
      CP = C3
      IF (EPS.GT.0.005)  GO TO 160
      EV = EV + 0.999/S
      GO TO 120
C
C     FIND NEW SPACE POINT COORDINATES AND AMOUNT OF AVERAGE SHIFTING
C
  250 AVE=0.0
      D=A3*XC+B3*YC+C3*ZC
      DO 300 I=1,NSP
      EV=D-A3*X(I)-B3*Y(I)-C3*Z(I)
      X(I)=X(I)+EV*A3
      Y(I)=Y(I)+EV*B3
      Z(I)=Z(I)+EV*C3
  300 AVE=AVE+ABS(EV)
      S=NSP
      AVE=AVE/S
C
      RETURN
      END
```

```
      SUBROUTINE Q12R12 (X,Y,D1,D2,D3,D4,D5,D6,S,JFLAG)
C
C*************************************************************************
C     THIS SUBROUTINE CALCULATES THE 12*12 STIFFNESS MATRIX OF A PLANE
C     STRESS QUADRILATERAL ELEMENT WITH 3 DEGREES OF FREEDOM PER CORNER
C     POINT. AN ELASTIC ANISOTROPIC CONSTITUTIVE MATERIAL LAW RELATES
C     STRESS AND STRAIN.
C
C                                    - INPUT -
C     D1,...,D6- ELEMENTS OF THE CONSTITUTIVE MATRIX
C     X(4),Y(4)- COORDINATES OF THE FOUR CORNER POINTS
C
C                                    - OUTPUT -
C     S(12,12) - ELEMENT STIFFNESS MATRIX, ORDERED IN ACCORDANCE WITH
C                THE DOF-S  U(I),...,U(L),V(I),...,V(L),THETA(I),...,
C                THETA(L)
C     JFLAG    - ERROR FLAG, SET EQUAL TO 1 FOR BAD JACOBIAN
C*************************************************************************
C
C     DIMENSION, EQUIVALENCE, AND DATA STATEMENTS
C
      DIMENSION P(4,2),DC(4,2),A(2,2),ETA(2),Q(4,2),R(4,2),IPERM(2),
     *     XK(3),WGT(3),EXX(4),EYY(4),EXY(4),EYX(4),EXZ(4),EYZ(4),
     *     EZZ(4),X1(4),X2(4),Y1(4),Y2(4),FT6(4),FT7(4),FT3(4),PSI(4,2),
     *     PHI(4,2),X(4),Y(4),TR(12,12),IM(4),IN(4),TEMP(12),S(12,12)
      EQUIVALENCE (A11,A(1)),(A21,A(2)),(A12,A(3)),(A22,A(4))
      DATA XK/-.7745966692415, .0000000000000, .7745966692415/,
     *    WGT/ .5555555555556, .8888888888889, .5555555555556/,
     *     DC/-1., 1., 1.,-1.,-1.,-1., 1., 1./,IPERM/2,1/,
     *     IM/2,1,4,3/,IN/4,3,2,1/
C
C        INITIALIZATION
C
C
      DO 110 I=1,12
      DO 100 J=I,12
      S(I,J)=0.0
      TR(I,J)=0.0
  100 TR(J,I)=0.0
  110 TR(I,I) = 1.0
C
C        GEOMETRICAL TRANSFORMATION OF QUADRILATERAL BOUNDARIES
C
      X1(1)=0.5*(X(2)-X(1))
      X1(3)=0.5*(X(3)-X(4))
      X1(2)=X1(1)
      X1(4)=X1(3)
      Y1(1)=0.5*(Y(2)-Y(1))
      Y1(3)=0.5*(Y(3)-Y(4))
      Y1(2)=Y1(1)
      Y1(4)=Y1(3)
      X2(1)=0.5*(X(4)-X(1))
      X2(2)=0.5*(X(3)-X(2))
      X2(3)=X2(2)
      X2(4)=X2(1)
      Y2(1)=0.5*(Y(4)-Y(1))
      Y2(2)=0.5*(Y(3)-Y(2))
      Y2(3)=Y2(2)
```

```
      Y2(4)=Y2(1)
      DO 120 I=1,4
      DET =X1(I)*Y2(I)-Y1(I)*X2(I)
      FCT=DET+X2(I)*Y1(I)
      FT3(I)= Y2(I)*(1,0-Y1(I)*X2(I)/FCT)
      FT6(I)= X1(I)*(1,0-Y1(I)*X2(I)/FCT)
  120 FT7(I)= 0,5/DET
C
C        MODIFICATION OF ROTATIONS DUE TO TRANSLATIONS
C
      DO 130 I=1,4
      IA=IM(I)
      IB=IN(I)
      TR(I+8,I   )=-FT7(I)*(DC(I,1)*X2(I) - DC(I,2)*X1(I))*0,5
      TR(I+8,I+4)=-FT7(I)*(DC(I,1)*Y2(I) - DC(I,2)*Y1(I))*0,5
      TR(I+8,IA  )=+FT7(I)*DC(I,1)*X2(I)*0.5
      TR(I+8,IA+4)=+FT7(I)*DC(I,1)*Y2(I)*0,5
      TR(I+8,IB  ) =-FT7(I)*DC(I,2)*X1(I)*0,5
      TR(I+8,IB+4)=-FT7(I)*DC(I,2)*Y1(I)*0,5
  130 CONTINUE
C
      DO 300 LX=1,3
      DO 300 LY=1,3
      ETA(1)=XK(LX)
      ETA(2)=XK(LY)
      WG=WGT(LX)*WGT(LY)
C
C        FORMATION OF LOCAL DERIVATIVES
C
      DO 170 I=1,2
      J=IPERM(I)
      EI=ETA(I)
      EJ=ETA(J)
      EI2=EI*EI
      EJ2=EJ*EJ
      EI3=EI2*EI
      EJ3=EJ2*EJ
      A(I,1) = 0,
      A(I,2) = 0,
      DO 160  L = 1,4
      C2 = DC(L,J)*ETA(J)
      P(L,I)= 0,250*DC(L,I)*(1,0+C2)
      A(I,1) = A(I,1) + P(L,I)*X(L)
      A(I,2) = A(I,2) + P(L,I)*Y(L)
      Q(L,I)=(-1,+2,*DC(L,I)*EI+3,*EI2)*(2,+DC(L,J)*(3,*EJ-EJ3))/16,
  160 R(L,J)=3,*(-DC(L,I)-EI+DC(L,I)*EI2+EI3)*DC(L,J)*(1,-EJ2)/16,
  170 CONTINUE
      DET = A11*A22 - A12*A21
      FAC = WG/DET
      IF (DET.LE.0.0) GO TO 1000
C
C        FORMATION OF GLOBAL DERIVATIVES
C
      DO 200 I=1,4
      PSI(I,1)=FT3(I)*R(I,1)
      PHI(I,1)=FT3(I)*Q(I,2)
      PSI(I,2) = FT6(I)*Q(I,1)
  200 PHI(I,2) = FT6(I)*R(I,2)
      DO 220 I=1,4
      EXX(I) = A22*P(I,1) - A12*P(I,2)
```

```
      EYY(I) =-A21*P(I,1) + A11*P(I,2)
      EYX(I) = EXX(I)
      EXY(I) = EYY(I)
      EXZ(I) = -A22*PSI(I,1) + A12*PHI(I,1)
      EYZ(I) = -A21*PSI(I,2) + A11*PHI(I,2)
  220 EZZ(I) =  A21*PSI(I,1) - A11*PHI(I,1) + A22*PSI(I,2) -A12*PHI(I,2)
C
C        FORMATION OF TRIPLE PRODUCT  BT * D * B
C
      DO 250 I=1,4
      E1=EXX(I)
      E2=EXY(I)
      E3=EYX(I)
      E4=EYY(I)
      E5=EXZ(I)
      E6=EYZ(I)
      E7=EZZ(I)
      DO 240 J=I,4
      S(I,J)=S(I,J)+FAC*(E1*(D1*EXX(J)+D5*EXY(J))+E2*(D5*EXX(J)
     *  +D3*EYY(J)))
      S(I+4,J+4)=S(I+4,J+4)+FAC*(E4*(D2*EYY(J)+D6*EYX(J))+E3*(D6*EYY(J)
     *  +D3*EYX(J)))
      S(I+8,J+8)=S(I+8,J+8)+FAC*(E5*(D1*EXZ(J)+D4*EYZ(J))+D5*(E5*EZZ(J)
     *  +E7*EXZ(J))+E6*(D4*EXZ(J)+D2*EYZ(J)+D6*EZZ(J))+E7*(D6*EYZ(J)
     *  +D3*EZZ(J)))
  240 CONTINUE
      DO 250 J=1,4
      XX=D5*EXZ(J)+D6*EYZ(J)+D3*EZZ(J)
      S(I,J+4)=S(I,J+4)+FAC*(E1*(D4*EYY(J)+D5*EYX(J))+E2*(D6*EYY(J)
     *  +D3*EYX(J)))
      S(I,J+8)=S(I,J+8)+FAC*(E1*(D1*EXZ(J)+D4*EYZ(J)+D5*EZZ(J))+E2*XX)
      S(I+4,J+8)=S(I+4,J+8)+FAC*(E4*(D4*EXZ(J)+D2*EYZ(J)+D6*EZZ(J))
     *  +E3*XX)
  250 CONTINUE
  300 CONTINUE
      DO 620 I=2,12
      K=I-1
      DO 620 J=1,K
  620 S(I,J)=S(J,I)
C
C        TRANSFORMATION OF STIFFNESS DUE TO TRANSLAT, EFFECT ON ROTATION
C
      DO 700 I=1,12
      DO 640 J=1,12
      XX=0,0
      DO 630 K=I,12
  630 XX=XX+S(J,K)*TR(K,I)
  640 TEMP(J)=XX
      DO 700 J=1,12
      YY=0,0
      DO 650 K=J,12
  650 YY=YY+TR(K,J)*TEMP(K)
  700 S(J,I) = YY
      RETURN
C
C        ERROR EXIT FOR BAD JACOBIAN
C
 1000 PRINT 1200, DET, (X(I),Y(I), I=1,4)
 1200 FORMAT(// 30H DETERMINANT OF JACOBIAN IS     E15,3//
     1 42H NODAL POINT COORDINATES OF BAD ELEMENT  - /4(2F10,3/))
```

```
      JFLAG=1                                                        TOST 772
      RETURN                                                         TOST 773
      END                                                            TOST 774

      SUBROUTINE PAREA (XL,YL,X,Y,A)                                 TOST 775
      DIMENSION X(4),Y(4),A(4),IJ(4),IK(4)                           TOST 776
      DATA IJ/2,3,4,1/, IK/4,1,2,3/                                  TOST 777
      DO 10 I=1,4                                                    TOST 778
      J=IJ(I)                                                        TOST 779
      K=IK(I)                                                        TOST 780
   10 A(I)=.0625*((XL-4.*X(I))*(Y(K)-Y(J))-(YL-4.*Y(I))*(X(K)-X(J))) TOST 781
      RETURN                                                         TOST 782
      END                                                            TOST 783
```

```
      SUBROUTINE ELSTIF (EA,EIY,EIZ,GJX,GAY,GAZ,IA,II,IL,R,NLC)          ELST
C                                                                        ELST
C************************************************************************ELST
C      IF IA=1, THIS SUBROUTINE CALCULATES THE 12*12 GLOBAL STIFFNESS OF ELST
C               A ONE-DIMENSIONAL BEAM ELEMENT WITH ARBITRARY ORIENTATIONELST
C               IN SPACE, ACCORDING TO THE MATRIX TRANSFORMATION         ELST
C                                   T = AT * K * A                       ELST
C      IF IA=2, THIS SUBROUTINE CALCULATES INTERNAL FORCES OF A ONE-     ELST
C               DIMENSIONAL BEAM ELEMENT FOR GIVEN END DISPLACEMENTS,    ELST
C               ACCORDING TO THE MATRIX EQUATION   S = K * A * R         ELST
C                                                                        ELST
C                                 -  INPUT  -                            ELST
C    (X(I),Y(I),Z(I),I=1,2) - GLOBAL COORDINATES OF END NODAL POINTS I,J ELST
C    (ECCI(I),ECCJ(I),I=1,3)- GLOBAL ECCENTRICITIES OF CONNECTION FOR ENDELST
C                             POINTS I AND J                             ELST
C    (AUXK(I), I=1,3)       - GLOBAL COORDINATES OF AUXILIARY POINT K .  ELST
C                             LYING IN LOCAL XZ-PLANE                    ELST
C    EA,EIY,EIZ,GJX,GAY,GAZ - ELEMENT RIGIDITIES                         ELST
C    R(NLC,12)              - NODAL DISPLACEMENT MATRIX (FOR IA=2 ONLY)  ELST
C    NLC                    - NUMBER OF LOAD CASES (FOR IA=2 ONLY)       ELST
C    II                     - TRANSFORMATION INDICATOR (FOR IA=1 ONLY)   ELST
C                             0  NO TRANSFORMATION REQUIRED              ELST
C                             1  TRANSFORM NODE J STIFFNESS              ELST
C                             2  TRANSFORM NODE I STIFFNESS              ELST
C    IL                     - COORDINATE FRAME INDICATOR                 ELST
C    G1, G2                 - ANGLE BETWEEN SECTIONS I AND I+1           ELST
C                             AND GLOBAL Y-AXIS                          ELST
C************************************************************************ELST
C                                                                        ELST
      COMMON/STIFF/T(12,12),XK(12,12),A(12,12),D(144),X(4),Y(4),Z(4),    ELST
     *  ECCI(3),ECCJ(3),AUXK(3),DUMMY(11),G1,G2,FILL(167)                ELST
      COMMON/SETUP/JUNK(48),K1,K2,K3,K4,K5,K6                            ELST
      DIMENSION R(NLC,1),IPERM(3),JPERM(3)                               ELST
      DATA IPERM/3,1,2/, JPERM/2,3,1/                                    ELST
C                                                                        ELST
C     INITIALIZATION                                                     ELST
C                                                                        ELST
      IF(IA.EQ.2) GO TO 60                                               ELST
      DO 3 I=1,12                                                        ELST
      DO 3 J=1,12                                                        ELST
      A(I,J)=0.0                                                         ELST
    3 T(I,J)=0.0                                                         ELST
C                                                                        ELST
C************************************************************************ELST
C     TRANSFORMATION OF CONNECTION ECCENTRICITIES                        ELST
C************************************************************************ELST
C                                                                        ELST
      IF(IL.EQ.2) GO TO 4                                                ELST
      SI=SIN(G1)                                                         ELST
      CI=COS(G1)                                                         ELST
      SJ=SIN(G2)                                                         ELST
      CJ=COS(G2)                                                         ELST
      GO TO 6                                                            ELST
    4 S=SIN(G2-G1)                                                       ELST
      C=COS(G2-G1)                                                       ELST
      IF(II.EQ.1) GO TO 5                                                ELST
      SJ=S                                                               ELST
      CJ=C                                                               ELST
      GO TO 7                                                            ELST
```

```
    5 SI=S
      CI=C
C
    6 XL=ECCI(1)*CI-ECCI(2)*SI
      ECCI(2)=ECCI(1)*SI+ECCI(2)*CI
      ECCI(1)=XL
      XL=AUXK(1)*CI-AUXK(2)*SI
      AUXK(2)=AUXK(1)*SI+AUXK(2)*CI
      AUXK(1)=XL
      IF(II.EQ.1) GO TO 8
    7 XL=ECCJ(1)*CJ-ECCJ(2)*SJ
      ECCJ(2)=ECCJ(1)*SJ+ECCJ(2)*CJ
      ECCJ(1)=XL
C
C**********************************************************************
C     ELEMENT COORDINATE TRANSFORMATION MATRIX
C**********************************************************************
C
    8 VX=X(2)-X(1)
      VY=Y(2)-Y(1)
      VZ=Z(2)-Z(1)
      XL=SQRT(VX*VX+VY*VY+VZ*VZ)
      D=1./XL
      A(1,1)=VX*D
      A(1,2)=VY*D
      A(1,3)=VZ*D
      VX=AUXK(1)-X(1)-ECCI(1)
      VY=AUXK(2)-Y(1)-ECCI(2)
      VZ=AUXK(3)-Z(1)-ECCI(3)
      D=1./SQRT(VX*VX+VY*VY+VZ*VZ)
      A(3,1)=VX*D
      A(3,2)=VY*D
      A(3,3)=VZ*D
      A(2,1)=A(3,2)*A(1,3)-A(3,3)*A(1,2)
      A(2,2)=A(3,3)*A(1,1)-A(3,1)*A(1,3)
      A(2,3)=A(3,1)*A(1,2)-A(3,2)*A(1,1)
      D=1./SQRT(A(2,1)*A(2,1)+A(2,2)*A(2,2)+A(2,3)*A(2,3))
      A(2,1)=A(2,1)*D
      A(2,2)=A(2,2)*D
      A(2,3)=A(2,3)*D
      A(3,1)=A(1,2)*A(2,3)-A(2,2)*A(1,3)
      A(3,2)=A(1,3)*A(2,1)-A(1,1)*A(2,3)
      A(3,3)=A(1,1)*A(2,2)-A(2,1)*A(1,2)
      DO 10 I=1,3
      DO 10 J=1,3
      A(I+3,J+3)=A(I,J)
      A(I+6,J+6)=A(I,J)
   10 A(I+9,J+9)=A(I,J)
C
      DO 15 I=1,3
      DO 15 J=1,3
      K=IPERM(J)
      L=JPERM(J)
      A(I  ,J+3)=A(I,K)*ECCI(L)-A(I,L)*ECCI(K)
   15 A(I+6,J+9)=A(I,K)*ECCJ(L)-A(I,L)*ECCJ(K)
C
C**********************************************************************
C     LOCAL ELEMENT STIFFNESS  (INCLUDING SHEAR DEFORMATIONS)
C**********************************************************************
C
```

```
      XL2=XL*XL
      XL3=XL*XL2
      VX=12.*EIZ/XL2
      VY=12.*EIY/XL2
      PY=0.0
      IF(GAY.GT.0.) PY=VX/GAY
      PZ=0.0
      IF(GAZ.GT.0.) PZ=VY/GAZ
      PYL=XL*(1.+PY)
      PZL=XL*(1.+PZ)
      T(1,1)=EA/XL
      T(7,7)=T(1,1)
      T(1,7)=-T(1,1)
      T(2,2)=VX/PYL
      T(8,8)=T(2,2)
      T(2,8)=-T(2,2)
      T(3,3)=VY/PZL
      T(9,9)=T(3,3)
      T(3,9)=-T(3,3)
      T(4,4)=GJX/XL
      T(10,10)=T(4,4)
      T(4,10)=-T(4,4)
      T(5,5)=(4.+PZ)*EIY/PZL
      T(11,11)=T(5,5)
      T(5,11)=(2.-PZ)*EIY/PZL
      T(6,6)=(4.+PY)*EIZ/PYL
      T(12,12)=T(6,6)
      T(6,12)=(2.-PY)*EIZ/PYL
      T(2,6)=6.*EIZ/(XL*PYL)
      T(2,12)=T(2,6)
      T(6,8)=-T(2,12)
      T(8,12)=T(6,8)
      T(3,5)=-6.*EIY/(XL*PZL)
      T(3,11)=T(3,5)
      T(5,9)=-T(3,5)
      T(9,11)=T(5,9)
      DO 35 I=1,12
      DO 35 J=I,12
   35 T(J,I)=T(I,J)
C
C*********************************************************************
C     FIND STIFFNESS IN GLOBAL COORDINATES
C*********************************************************************
C
      DO 40 I=1,12
      DO 40 J=1,12
      XK(I,J)=0.0
      DO 40 K=1,12
   40 XK(I,J)=XK(I,J)+T(I,K)*A(K,J)
      WRITE (K1) XK
      DO 50 I=1,12
      DO 50 J=I,12
      T(I,J)=0.0
      DO 45 K=1,12
   45 T(I,J)=T(I,J)+A(K,I)*XK(K,J)
   50 T(J,I)=T(I,J)
C
C*********************************************************************
C     TRANSFORMATION OF STIFFNESS FOR THE CASE OF TRAVELLING COORDINATES
C*********************************************************************
```

```
C
      IF(II.EQ.0) GO TO 100
      K=1
      IF(II.EQ.1) K=7
      L=K+3
      DO 54 I=K,L,3
      DO 52 J=1,12
      XL=-T(J,I)*S+T(J,I+1)*C
      T(J,I)=T(J,I)*C+T(J,I+1)*S
   52 T(J,I+1)=XL
   54 CONTINUE
      DO 58 I=K,L,3
      DO 56 J=1,2
      XL=-T(I,J)*S+T(I+1,J)*C
      T(I,J)=T(I,J)*C+T(I+1,J)*S
   56 T(I+1,J)=XL
   58 CONTINUE
      GO TO 100
C
C***********************************************************************
C     FIND INTERNAL ELEMENT FORCES
C***********************************************************************
C
   60 READ (K1) XK
      DO 80 L=1,NLC
      DO 70 I=1,12
      D(I)=0.0
      DO 70 J=1,12
   70 D(I)=D(I)+XK(I,J)*R(L,J)
      DO 80 I=1,12
   80 R(L,I)=D(I)
C
  100 RETURN
      END
```

```
      SUBROUTINE LOADS (NPI,NPJ,NBLK,TZ,TY,TX,AREA,NELSL,DL,XL,YL,ZL,
     *  NBLSL,NSLSL,SDL,SXL,SYL,SZL,NJL,NSS,NSO,NID,FI,FORCE,NESL,NSSL,
     *  NCL,NS,NM,KL,L)
C
C**********************************************************************
C     THIS SUBROUTINE FORMS ONE LOAD VECTOR AT A TIME AND STORES IT
C     ON TAPE KL  (=K4 IF NLC.GT.1,  =K5 IF NLC.EQ.1)
C**********************************************************************
C
C        COMMON AND DIMENSION STATEMENTS
C
      DIMENSION NPI(1),NPJ(1),NBLK(1),TX(NS,NM,1),TY(NS,NM,1),AREA(4,1),
     *  TZ(NS,NM,1),NELSL(1),DL(1),XL(1),YL(1),ZL(1),NBLSL(1),NSLSL(1),
     *  SDL(1),SXL(1),SYL(1),SZL(1),NJL(1),NSS(1),NSO(1),NID(1),FI(1),
     *  FORCE(1)
      DIMENSION XE(4),YE(4),ZE(4),NN(4),AA(4)
      COMMON/SETUP/NSEG,NSECT,NFL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,
     *  NTAD,NPLD,NCSTYP,NLC,NFORST,NSAV,NSCK,IA,IS,IM,IR,IL,ISTOP,ICHK,
     *  R,NMAX,NEG,NEQS,NQ,MBAND,NBLCK,NDBC,JFLAG,PI,NDE,NDS,NFST,NFCT,
     *  NCOMM,NBR,IC1,IC2,IC3,IC4,IC5,IC6,NGIR,IREACT,IRESID,
     *  K1,K2,K3,K4,K5,K6
C
C**********************************************************************
C        1) READ AND PRINT INPUT DATA FOR LOAD CASE NO.  L
C**********************************************************************
C
      DO 10 I=1,NEQS
   10 FORCE(I)=0.0
C
      IF(NESL.LE.0) GO TO 20
      READ 1001, (NELSL(I),DL(I),XL(I),YL(I),ZL(I), I=1,NESL)
      PRINT 2001
      PRINT 2002, (NELSL(I),DL(I),XL(I),YL(I),ZL(I), I=1,NESL)
C
   20 IF(NSSL.LE.0) GO TO 25
      READ 1002,(NBLSL(I),NSLSL(I),SDL(I),SXL(I),SYL(I),SZL(I),I=1,NSSL)
      PRINT 2003
      PRINT2004,(NBLSL(I),NSLSL(I),SDL(I),SXL(I),SYL(I),SZL(I),I=1,NSSL)
C
   25 IF(NCL.LE.0) GO TO 30
      READ 1003, (NJL(I),NSS(I),NSO(I),NID(I),FI(I), I=1,NCL)
      PRINT 2005
      PRINT 2006, (NJL(I),NSS(I),NSO(I),NID(I),FI(I), I=1,NCL)
   30 IF(ISTOP.EQ.1) RETURN
C
C**********************************************************************
C        2) CALCULATE EQUIVALENT NODAL FORCES DUE TO SURFACE LOADS
C**********************************************************************
C
C        REGULAR SURFACE LOADS
C
      IF(NESL.LE.0) GO TO 50
      II=0
      DO 45 I=1,NSEG
      IK=NBLK(I)-7
      JK=NBLK(I+1)-7
      DO 40 J=1,NESL
      IJ=NELSL(J)
      NI=NPI(IJ)
```

```
      NJ=NPJ(IJ)
      CALL ELCOOR (NSECT,NMAX,TX,TY,TZ,I,NI,NJ,0,0,IL,1,XE,YE,ZE)
      NN(1)=IK+NI*6
      NN(2)=IK+NJ*6
      NN(3)=JK+NJ*6
      NN(4)=JK+NI*6
      KK=II+IJ
      DO 35 K=1,4
   35 AA(K)=AREA(K,KK)
      CALL ADLOAD (FORCE,XE,YE,ZE,DL(J),XL(J),YL(J),ZL(J),AA,NN)
   40 CONTINUE
   45 II=II+NEL
C
C        SPECIAL SURFACE LOADS
C
   50 IF(NSSL.LE.0) GO TO 70
      DO 60 I=1,NSSL
      II=NBLSL(I)
      JJ=NSLSL(I)
      KK=II*NEL-NEL+JJ
      NI=NPI(JJ)
      NJ=NPJ(JJ)
      NN(1)=NBLK(II)+NI*6-7
      NN(2)=NBLK(II)+NJ*6-7
      NN(3)=NBLK(II+1)+NJ*6-7
      NN(4)=NBLK(II+1)+NI*6-7
      CALL ELCOOR (NSECT,NMAX,TX,TY,TZ,II,NI,NJ,0,0,IL,1,XE,YE,ZE)
      DO 55 K=1,4
   55 AA(K)=AREA(K,KK)
      CALL ADLOAD (FORCE,XE,YE,ZE,SDL(I),SXL(I),SYL(I),SZL(I),AA,NN)
   60 CONTINUE
C
C*********************************************************************
C        3) ADD CONCENTRATED NODAL LOADS INTO FORCE VECTOR
C*********************************************************************
C
   70 IF(NCL.LE.0) GO TO 110
      DO 100 I=1,NCL
      II=NSS(I)
      IJ=NSO(I)
      N=NJL(I)
      JJ=6*N+NID(I)-7
      IF(II.LT.IJ) GO TO 75
      IK=NBLK(II)+JJ
      FORCE(IK)=FORCE(IK)+FI(I)
      GO TO 100
C
   75 IJ=IJ-1
      DO 90 J=II,IJ
      IF(IL.EQ.2) GO TO 80
      XT=TX(J+1,N,1)-TX(J,N,1)
      YT=TY(J+1,N,1)-TY(J,N,1)
      ZT=TZ(J+1,N,1)-TZ(J,N,1)
      GO TO 85
   80 XT=TX(J,N,2)-TX(J,N,1)
      YT=TY(J,N,2)-TY(J,N,1)
      ZT=TZ(J,N,2)-TZ(J,N,1)
   85 XX=SQRT(XT*XT+YT*YT+ZT*ZT)
      IK=NBLK(J)+JJ
      JK=NBLK(J+1)+JJ
```

```
      FF=FI(I)*XX/2.
      FORCE(IK)=FORCE(IK)+FF
   90 FORCE(JK)=FORCE(JK)+FF
  100 CONTINUE
C
C***********************************************************************
C         4) WRITE LOAD VECTOR ON TAPE KL AND PRINT OUT IF DESIRED
C***********************************************************************
C
  110 II=1
      DO 120 I=1,NBLCK
      IJ=II+NQ-1
      WRITE (KL) (FORCE(J), J=II,IJ)
  120 II=II+NQ
      IF(IC4.EQ.0) GO TO 130
      PRINT 2008, L
      PRINT 2007, (FORCE(I), I=1,NEQS)
  130 RETURN
C
C***********************************************************************
C         FORMAT STATEMENTS
C***********************************************************************
C
 1001 FORMAT(7X,I3,4F10.0)
 1002 FORMAT(2I5,4F10.0)
 1003 FORMAT(4I4,F10.0)
C
 2001 FORMAT(/////30H REGULAR SURFACE LOAD ELEMENTS//60H ELEMENT NO.    D
     *EAD LOAD       X-LOAD       Y-LOAD       Z-LOAD)
 2002 FORMAT(I7,5X,4F12.4)
 2003 FORMAT(/////30H SPECIAL SURFACE LOAD ELEMENTS//74H SEGMENT NO.    E
     *LEMENT NO.   DEAD LOAD       X-LOAD       Y-LOAD       Z-LOAD)
 2004 FORMAT(I7,I14,5X,4F12.4)
 2005 FORMAT(/////28H CONCENTRATED AND LINE LOADS//68H JOINT NO.    START
     * SECT.   END SECT.  COMPONENT NO.   LOAD INTENSITY)
 2006 FORMAT(I6,3I13,F20.4)
 2007 FORMAT(6E14.6)
 2008 FORMAT(1H1,30H LOAD VECTOR FOR LOAD CASE NO.,I5//)
      END


      SUBROUTINE ADLOAD (FORCE,XE,YE,ZE,DL,XL,YL,ZL,AA,NN)
C
C***********************************************************************
C     THIS SUBROUTINE ADDS NODAL LOADS EQUIVALENT TO DISTRIBUTED LOADS
C     INTO THE LOAD VECTOR  FORCE.
C***********************************************************************
C
      DIMENSION FORCE(1),XE(4),YE(4),ZE(4),AA(4),NN(4)
C
      L=1
      XX=XE(1)+XE(2)+XE(3)+XE(4)
      YY=YE(1)+YE(2)+YE(3)+YE(4)
      ZZ=ZE(1)+ZE(2)+ZE(3)+ZE(4)
      IF(DL.EQ.0.) GO TO 15
      IND=3
      X=DL
      GO TO 30
   15 L=2
      IF(XL.EQ.0.) GO TO 20
```

```
      CALL PAREA (ZZ,YY,ZE,YE,AA)
      IND=1
      X=XL
      GO TO 30
   20 L=3
      IF(YL.EQ.0.) GO TO 25
      CALL PAREA (ZZ,XX,ZE,XE,AA)
      IND=2
      X=YL
      GO TO 30
   25 L=4
      IF(ZL.EQ.0.) GO TO 50
      CALL PAREA (YY,XX,YE,XE,AA)
      IND=3
      X=ZL
C
   30 DO 40 K=1,4
      IJ=NN(K)+IND
   40 FORCE(IJ)=FORCE(IJ)+AA(K)*X
C
      GO TO (15,20,25,50), L
   50 RETURN
      END
```

```
      SUBROUTINE FORMK(NPI,NPJ,NPSEC,NBLK,IFD,FORC1,INDCR,IDISBC,DISPL, FRMK
     *  BIGK,FORC2,NL,NZ,NC,ND,MB)                                      FRMK
C                                                                       FRMK
C***********************************************************************FRMK
C     THIS SUBROUTINE ASSEMBLES THE STRUCTURE STIFFNESS AND ARRANGES    FRMK
C     THE EQUATION BLOCKS SUCH THAT THEY ARE READY FOR THE SOLUTION     FRMK
C     PROCESS BY SUBROUTINE USOL.                                       FRMK
C***********************************************************************FRMK
C                                                                       FRMK
C         DIMENSION AND COMMON STATEMENTS                               FRMK
C                                                                       FRMK
       DIMENSION NPI(1),NPJ(1),NPSEC(1),NBLK(1),IFD(1),BIGK(NL,MB),     FRMK
     *  FORC1(NZ,NC),FORC2(NZ,NC),INDCR(1),IDISBC(ND),DISPL(ND)         FRMK
      DIMENSION NN(4),ST(12,12)                                         FRMK
      COMMON/SETUP/NSEG,NSECT,NEL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,    FRMK
     *  NTAD,NPLD,NCSTYP,NLC,NFORST,NSAV,NSCK,IA,IS,IM,IR,IL,ISTOP,ICHK,FRMK
     *  R,NMAX,NEQ,NEQS,NQ,MBAND,NBLCK,NDBC,JFLAG,PI,NDE,NDS,NFST,NFCT, FRMK
     *  NCOMM,NBR,IC1,IC2,IC3,IC4,IC5,IC6,NGIR,IREACT,IRESID,           FRMK
     *  K1,K2,K3,K4,K5,K6                                               FRMK
      COMMON/DIAPH/NDTY(20),NPD(4,20),NDSE(20),NDSD(20),NDST(20)        FRMK
      COMMON/FRAME/EA(20),EIY(20),EIZ(20),GJX(20),GAY(20),GAZ(20),      FRMK
     *  EI(3,20),EJ(3,20),EK(3,20),NFI(100),NFJ(100),NSFI(100),         FRMK
     *  NSFJ(100),NFTY(100),NFSC(100)                                   FRMK
      COMMON/STIFF/T(24,24),DUMMY(201)                                  FRMK
      EQUIVALENCE (T,ST)                                                FRMK
      LOGICAL L1,L2                                                     FRMK
C                                                                       FRMK
C***********************************************************************FRMK
C         1) REARRANGE LOAD VECTORS ON TAPE K5 (IF  NLC.GT.1)           FRMK
C***********************************************************************FRMK
C                                                                       FRMK
      L2=NFRAME+NDIAPH.EQ.0                                             FRMK
      IF(NLC.EQ.1) GO TO 9                                              FRMK
      REWIND K5                                                         FRMK
      DO 8 I=1,NBLCK                                                    FRMK
      REWIND K4                                                         FRMK
      DO 2 J=1,I                                                        FRMK
    2 READ (K4) (FORC1(K,1), K=1,NQ)                                    FRMK
      WRITE(K5) (FORC1(K,1), K=1,NQ)                                    FRMK
      DO 6 J=2,NLC                                                      FRMK
      DO 4 K=1,NBLCK                                                    FRMK
    4 READ (K4) (FORC1(L,1), L=1,NQ)                                    FRMK
    6 WRITE(K5) (FORC1(L,1), L=1,NQ)                                    FRMK
    8 CONTINUE                                                          FRMK
C                                                                       FRMK
C***********************************************************************FRMK
C         2) INITIALIZE STRUCTURE STIFFNESS BIGK                        FRMK
C***********************************************************************FRMK
C                                                                       FRMK
C         INITIALIZE LEADING BLOCK                                      FRMK
C                                                                       FRMK
    9 MBL=MB+NLC                                                        FRMK
      NB=NBLCK                                                          FRMK
      REWIND K2                                                         FRMK
      REWIND K3                                                         FRMK
      REWIND K4                                                         FRMK
      REWIND K5                                                         FRMK
      DO 15 J=1,NQ                                                      FRMK
      DO 15 K=1,MB                                                      FRMK
```

```
   15 BIGK(J,K)=0.0
C
C         SHIFT UP TRAILING BLOCK
C
      NS=NQ+1
      NE=NQ+NQ
      II=1
      NFIN=0
      MFIN=0
      N=0
      MMF=0
      DO 200 I=1,NB
      IJ=II+NQ
      IK=IJ+NQ-1
      IF(I.EQ.1) GO TO 22
      DO 20 J=1,NQ
      DO 20 K=1,MB
   20 BIGK(J,K)=BIGK(J+NQ,K)
C
C         INITIALIZE TRAILING BLOCK
C
   22 IF(I.EQ.NB.OR.NB.EQ.1) GO TO 27
      DO 25 J=NS,NE
      DO 25 K=1,MB
   25 BIGK(J,K)=0.0
C
C***********************************************************************
C         3) STORE PLATE ELEMENT STIFFNESSES
C***********************************************************************
C
C         FIND POSITION OF FIRST READ RECORD ON TAPE K2
C
   27 KK=II
      KI=NBLK(NFIN+1)
      IF(KI.GE.II.AND.KI.LT.IJ) GO TO 40
      REWIND K2
      IF(NFIN.EQ.0) GO TO 35
      JJ=NFIN*NEL
      DO 30 J=1,JJ
   30 READ (K2)
   35 N=NFIN
      KK=IJ
C
C         STORE PLATE ELEMENT STIFFNESSES
C
   40 N=N+1
      IF(N.GT.NSEG) GO TO 50
      NI=NBLK(N)-7
      NJ=NPSEC(N)*6
      DO 45 J=1,NEL
      READ (K2) T
      NN(1)=6*NPI(J)+NI
      NN(2)=6*NPJ(J)+NI
      NN(3)=NN(2)+NJ
      NN(4)=NN(1)+NJ
      CALL STORE (NN,KK,IK,BIGK,NL,4,T,24,II)
   45 CONTINUE
      NI=NBLK(N+2)-1
      IF(NI.LE.IK) NFIN=N
      NI=NBLK(N+1)
```

```
      KK=II
      IF(NI.LT.II) KK=IJ
      IF(NI.LT.IJ) GO TO 40
C
C************************************************************************
C         4) ADD DIAPHRAGM AND FRAME STIFFNESSES
C************************************************************************
C
   50 IF(L2) GO TO 100
      M=MFIN
   60 M=M+1
      IF(M.GT.NSECT) GO TO 100
      IF(IFD(M).EQ.0) GO TO 60
      KI=NBLK(M)
      IF(KI.GE.IJ) GO TO 100
      KK=II
      IF(KI.GE.II) GO TO 70
      KK=IJ
      REWIND K3
      IF(MMF.EQ.0) GO TO 70
      DO 65 K=1,MMF
   65 READ (K3)
   70 IF(IFD(M).EQ.1) GO TO 80
C
C         DIAPHRAGMS
C
      NI=NBLK(M)-7
      DO 75 J=1,NDE
      READ (K3) T
      DO 72 K=1,4
   72 NN(K)=NI+6*NPD(K,J)
      CALL STORE (NN,KK,IK,BIGK,NL,4,T,24,II)
   75 CONTINUE
      L1=NBLK(M+1)-1.LE.IK
      IF(L1) MMF=MMF+NDE
C
C         FRAMES
C
   80 IF(IFD(M).EQ.2) GO TO 95
      JJ=0
      L1=.TRUE.
      DO 90 J=1,NFRAME
      KI=NSFI(J)
      KJ=NSFJ(J)
      IF(KI.NE.M.AND.KJ.NE.M) GO TO 90
      IF(KI.LT.M.OR.KJ.LT.M) GO TO 90
      READ (K3) ST
      JJ=JJ+1
      KI=NBLK(KI)+6*NFI(J)-1
      KJ=NBLK(KJ)+6*NFJ(J)-1
      KA=KK
      IF(KI.GT.IK.OR.KJ.GT.IK) L1=.FALSE.
      IF(KI.LT.KK.OR.KJ.LT.KK) KA=IJ
      NN(1)=KI-6
      NN(2)=KJ-6
      CALL STORE (NN,KA,IK,BIGK,NL,2,ST,12,II)
   90 CONTINUE
      IF(L1) MMF=MMF+JJ
   95 IF(L1) MFIN=M
      GO TO 60
```

```
C
C**********************************************************************
C        5) FORM BLOCK COMBINATION OF ORIGINAL BIGK AND FORCE MATRICES
C**********************************************************************
C
  100 DO 110 L=1,NLC
  110 READ (K5)(FORC1(K,L), K=1,NQ)
      WRITE (K4)((BIGK(J,K),J=1,NQ),K=1,MB), FORC1
  200 II=II+NQ
C
C**********************************************************************
C        6) SETUP FOR MODIFICATION DUE TO BOUNDARY CONDITIONS
C**********************************************************************
C
      REWIND K1
      REWIND K2
      REWIND K3
      REWIND K4
      REWIND K5
      REWIND K6
      N1=K2
      N2=K3
      NBR=(MB-1)/NQ+2
      II=1
      IK=1
      KC=NQ+MB-1
C
C        READ BOUNDARY CONDITION ARRAYS FROM TAPE K1
C
      READ (K1)(INDCR(I), I=1,NEQ)
      IF(NDBC.GT.0) READ (K1)IDISBC,DISPL
C
C        READ ORIGINAL STRUCTURE STIFFNESS FROM TAPE K4
C
      DO 400 I=1,NB
      IJ=II+NQ-1
      READ (K4)((BIGK(J,K),J=1,NQ),K=1,MB), FORC1
      IF(NDBC.EQ.0) GO TO 350
      IF(I.GT.1) GO TO 210
      NI=K5
      GO TO 215
  210 IF(NBR.EQ.2) GO TO 220
      REWIND N1
      REWIND N2
      NI=N2
  215 DO 217 L=1,NLC
  217 READ (NI) (FORC1(K,L), K=1,NQ)
      GO TO 230
  220 DO 225 K=1,NQ
      DO 225 L=1,NLC
  225 FORC1(K,L)=FORC2(K,L)
C
C**********************************************************************
C        7) MODIFY LEADING BLOCK OF LOAD MATRIX DUE TO NON-ZERO B.C.
C**********************************************************************
C
  230 IF(IK.GT.NDBC) GO TO 280
      DO 270 J=IK,NDBC
      JJ=IDISBC(J)-II+1
      JI=JJ-MB+1
```

```
      IF(JI.GT.NQ) GO TO 280
      IF(JI.LT.1) JI=1
      KI=MIN0(JJ,NQ)
      DO 240 K=JI,KI
      JK=JJ-K+1
      XX=BIGK(K,JK)*DISPL(J)
      DO 240 L=1,NLC
  240 FORC1(K,L)=FORC1(K,L)-XX
C
      KI=NQ-JJ+1
      IF(KI.LT.2) GO TO 270
      IF(KI.LT.MB) GO TO 245
      KI=MB
      IK=IK+1
  245 DO 250 K=2,KI
      JK=JJ+K-1
      XX=BIGK(JJ,K)*DISPL(J)
      DO 250 L=1,NLC
  250 FORC1(JK,L)=FORC1(JK,L)-XX
  270 CONTINUE
C
C*********************************************************************
C        8) MODIFY TRAILING BLOCK OF LOAD MATRIX DUE TO NON-ZERO DISPLS,
C*********************************************************************
C
  280 IF(I.EQ.NB) GO TO 350
      JA=2
      JK=NQ+2
      IF(NBR.EQ.2.OR.I.EQ.1) GO TO 285
      NI=N1
      GO TO 290
  285 IF(I+NBR-2.GT.NB) GO TO 350
      NI=K5
C
  290 DO 293 L=1,NLC
  293 READ (NI) (FORC2(K,L),  K=1,NQ)
      IF(IK.GT.NDBC) GO TO 315
      DO 310 J=IK,NDBC
      JJ=IDISBC(J)-II+1
      IF(JJ.GT.IJ) GO TO 315
      JI=JK-JJ
      KI=JI+NQ-1
      IF(KI.LT.MB) GO TO 295
      KI=MB
      IK=IK+1
  295 KK=NQ+1
      DO 305 K=JI,KI
      XX=BIGK(JJ,K)*DISPL(J)
      DO 300 L=1,NLC
  300 FORC2(KK,L)=FORC2(KK,L)-XX
  305 KK=KK+1
  310 CONTINUE
C
  315 IF(NBR.EQ.2) GO TO 320
      WRITE (N2) FORC2
  320 IF(JA.EQ.NBR.OR.I+JA-1.EQ.NB) GO TO 350
      IF(JA.EQ.NBR-1) GO TO 330
      NI=K5
  330 JA=JA+1
      JK=JK+NQ
```

```
      GO TO 290
C
C*************************************************************************
C         9) MODIFY STRUCTURE STIFFNESS DUE TO BOUNDARY CONDITIONS
C*************************************************************************
C
  350 J=II-1
  355 J=J+1
      JJ=J-II+1
      IF(JJ.GT.KC.OR.J.GT.NEQ) GO TO 390
      IF(INDCR(J).EQ.0) GO TO 355
      JI=MAX0(1,JJ-MB+1)
      KI=MIN0(JJ,NQ)
      DO 360 K=JI,KI
  360 BIGK(K,JJ-K+1)=0.0
      IF(JJ.GT.NQ) GO TO 355
      DO 365 K=2,MB
  365 BIGK(JJ,K)=0.0
      GO TO 355
C
C*************************************************************************
C        10) SAVE STRUCTURE STIFFNESS AND LOAD BLOCK ON TAPE K6
C*************************************************************************
C
  390 WRITE(K6) ((BIGK(J,K),J=1,NQ),K=1,MB), FORC1
      NI=N1
      N1=N2
      N2=NI
  400 II=II+NQ
C
C         PRINT ORIGINAL STRUCTURE STIFFNESS MATRIX IF DESIRED
C
      IF(IC6.EQ.0) GO TO 500
      REWIND K4
      PRINT 2000
      II=1
      DO 420 I=1,NB
      PRINT 2001, I
      READ (K4)((BIGK(J,K),J=1,NQ),K=1,MB)
      DO 410 J=1,NQ
      PRINT 2002, II
      PRINT 2003, (BIGK(J,K),K=1,MB)
  410 II=II+1
  420 CONTINUE
C
  500 RETURN
C
 2000 FORMAT(1H1,21H  STRUCTURE STIFFNESS)
 2001 FORMAT(///10H BLOCK NO.,I5//)
 2002 FORMAT(/17H     EQUATION NO.,I8/)
 2003 FORMAT(8E12.4)
      END


      SUBROUTINE STORE (NN,IA,IB,BIGK,NL,L,T,NR,IC)
C
C*************************************************************************
C     THIS SUBROUTINE ADDS THAT PORTION OF ONE- OR TWO-DIMENSIONAL
C     STIFFNESSES INTO THE STRUCTURE STIFFNESS BIGK WHICH FALLS
C     BETWEEN EQUATION IA AND IB.
```

```
C*********************************************************************
C
      DIMENSION BIGK(NL,1),NN(4),T(NR,1)
C
      IE=0
      DO 40 I=1,L
      IS=NN(I)
      IF(IS+6.LT.IA.OR.IS.GT.IB) GO TO 40
      JE=0
      DO 30 J=1,L
      JS=NN(J)
      IF(JS.LT.IS) GO TO 30
      DO 20 II=1,6
      IIS=IS+II
      IF(IIS.GT.IB.OR.IIS.LT.IA) GO TO 20
      IIE=IE+II
      DO 10 JJ=1,6
      JJS=JS+JJ
      IF(JJS.LT.IIS) GO TO 10
      IJS=IIS-IC+1
      JJS=JJS-IIS+1
      JJE=JE+JJ
      BIGK(IJS,JJS)=BIGK(IJS,JJS)+T(IIE,JJE)
   10 CONTINUE
   20 CONTINUE
   30 JE=JE+6
   40 IE=IE+6
      RETURN
      END
```

```
      SUBROUTINE USOL (A,B,MAXB,NEQB,MB,LL,NBLOCK,NSB,NORG,NBKS,NT1,
     *  NT2,NRST)
C
C***********************************************************************
C        GENERAL EQUATION SOLVER FOR SYMMETRIC POSITIVE-DEFINITE SYSTEMS
C
C                          - INPUT -
C     A,B     - STORAGE AREAS FOR LEADING AND TRAILING BLOCKS OF
C               EQUATIONS, INCLUDING ALL LOAD VECTORS
C     MAXB    - STORAGE AREA FOR VARIABLE BANDWIDTH INFORMATION
C     NEQB    - NUMBER OF EQUATIONS IN A BLOCK
C     MB      - MAXIMUM HALF-BAND WIDTH OF MATRIX
C     LL      - NUMBER OF LOAD VECTORS
C     NBLOCK - TOTAL NUMBER OF BLOCKS
C     NSB=(MB+LL)*NEQB - LENGTH OF STORAGE AREA FOR BLOCK A AND B
C     NORG=6 - NUMBER OF TAPE ON WHICH ORIGINAL EQUATIONS ARE STORED,
C               ONE BLOCK PER RECORD
C     NBKS=5 - NUMBER OF TAPE USED FOR TEMPORARY STORAGE OF REDUCED
C               EQUATIONS
C     NT1=2, NT2=3 - NUMBERS OF TAPES USED FOR TEMPORARY STORAGE IF
C               MB.GT.NEQB
C     NRST=6 - NUMBER OF TAPE ON WHICH UNKNOWNS ARE STORED, ONE
C               BLOCK PER RECORD, IN REVERSED ORDER
C***********************************************************************
C
      DIMENSION A(NSB),B(NSB),MAXB(NEQB)
C
      NC=MB+LL
      NBR=(MB-1)/NEQB+1
      INC=NEQB-1
      NMB=NEQB*MB
      N2=NT2
      N1=NT1
      REWIND NORG
      REWIND NBKS
C
C     REDUCE EQUATIONS BLOCK BY BLOCK
C
      DO 900 N=1,NBLOCK
      IF (N.GT.1.AND.NBR.EQ.1) GO TO 110
      IF (NBR.EQ.1) GO TO 105
      REWIND N1
      REWIND N2
  105 NI=N1
      IF(N.EQ.1) NI=NORG
      READ (NI) A
  110 DO 300 I=1,NEQB
      D=A(I)
      IF(D) 115,300,120
  115 M=NEQB*(N-1)+I
      PRINT 116, M,D
  116 FORMAT (33H0SET OF EQUATIONS MAY BE SINGULAR   /
     *  26H DIAGONAL TERM OF EQUATION   I8, 8H  EQUALS   1PE12.4)
      D=-D
  120 D=SQRT(D)
      A(I)=D
C
      II=I
      DO 125 J=2,NC
```

```
      II=II+NEQB
  125 A(II)=A(II)/D
C
      DO 130 J=I,NMB,NEQB
      IF (A(J).NE.0.) MAXB(I)=J
  130 CONTINUE
C
      JL=I+1
      IF (JL.GT.NEQB) GO TO 300
      II=I
      DO 200 J=JL,NEQB
      II=II+NEQB
      IF(II.GT.NMB) GO TO 200
      C=A(II)
      IF (C.EQ.0.0) GO TO 200
C
      KK=J
      MAX=MAXB(I)
      DO 150 JJ=II,MAX,NEQB
      A(KK)=A(KK)-C*A(JJ)
  150 KK=KK+NEQB
C
      KK=J +NMB
      JJ=I+NMB
      DO 175 L=1,LL
      A(KK)=A(KK)-C*A(JJ)
      KK=KK+NEQB
  175 JJ=JJ+NEQB
  200 CONTINUE
  300 CONTINUE
      WRITE (NBKS) A,MAXB
C
C     SUBSTITUTE INTO REMAINING EQUATIONS
C
      DO 800 NN=1,NBR
      IF(N+NN.GT.NBLOCK) GO TO 800
      NI=N1
      IF(N.EQ.1) NI=NORG
      IF(NN.EQ.NBR) NI=NORG
      READ (NI) B
      IL=1+NN*NEQB*NEQB
      DO 700 I=1,NEQB
      II=IL
      DO 690 K=1,NEQB
      IF (II.GT.NMB) GO TO 690
      C=A(II)
      IF (C.EQ.0.0) GO TO 690
      MAX=MAXB(K)
C
      KK=I
      DO 640 JJ=II,MAX,NEQB
      B(KK)=B(KK)-C*A(JJ)
  640 KK=KK+NEQB
C
      KK=I+NMB
      JJ=K+NMB
      DO 650 L=1,LL
      B(KK)=B(KK)-C*A(JJ)
      KK=KK+NEQB
  650 JJ=JJ+NEQB
```

```
C
  690 II=II-INC
  700 IL=IL+NEQB
C
      IF(NBR.NE.1) GO TO 750
      DO 740 I=1,NSB
  740 A(I)=B(I)
      GO TO 800
  750 WRITE (N2) B
  800 CONTINUE
C
      M=N1
      N1=N2
  900 N2=M
C
C     BACKSUBSTITUTION
C
      LS=LL*NEQB
      NEB=NEQB*(NBR+1)
      NUM=NBR*NEQB
      MAX=NEB*LL
      DO 905 I=1,MAX
  905 B(I)=0.
      REWIND NRST
C
      DO 1000 N=1,NBLOCK
      BACKSPACE NBKS
      READ (NBKS) A,MAXB
      BACKSPACE NBKS
      DO 910 L=1,LL
      K=L*NEB
      DO 910 J=1,NUM
      I=K-NEQB
      B(K)=B(I)
  910 K=K-1
C
      I=NMB
      DO 920 L=1,LL
      K=(L-1)*NEB
      DO 920 J=1,NEQB
      I=I+1
      K=K+1
  920 B(K)=A(I)
C
      DO 955 I=1,NEQB
      J=NEQB+1-I
      MAX=MAXB(J)
      IF (A(J).EQ.0.) GO TO 955
      DO 950 L=1,LL
      KK=J+(L-1)*NEB
      JJ=KK+1
      IL=J+NEQB
      C=B(KK)
      DO 940 II=IL,MAX,NEQB
      C=C-A(II)*B(JJ)
  940 JJ=JJ+1
  950 B(KK)=C/A(J)
  955 CONTINUE
C
      I=0
```

```
      DO 960 L=1,LL
      K=(L-1)*NEB
      DO 960 J=1,NEQB
      K=K+1
      I=I+1
  960 A(I)=B(K)
C
      WRITE (NRST) (A(I),I=1,LS)
 1000 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE OUTPUT (NPI,NPJ,NPSEC,NBLK,NFORS,NFORSN,ALF,NSA,NSAVS,
     *  NPAV,NCS,NGE,XDIV,IOSEG,IFD,TZ,TY,DIS,INDCR,IDISBC,DISPL,BIGK,
     *  FORCE)
      RETURN
      ENTRY OUTPU2 (F,DD,DISP,GIRMOM,TENS,COMP,B1,B2,B3,FF,D,E,DI,
     *  NY,NS,NT,NM,NR,MB,NC,NQB,NK,NG,NF,NX,ND,IODIA)
C
C**********************************************************************
C     THIS SUBROUTINE ARRANGES FINAL NODAL DISPLACEMENTS IN CORRECT
C     ORDER AND CALCULATES ALL UNKNOWN REACTIONS AND RESIDUAL LOADS,
C     INTERNAL FORCES FOR ALL PLATE AND BEAM TYPE ELEMENTS ARE COMPUTED,
C     AND ALL OUTPUT IS PRINTED OUT FOR ONE LOAD CASE AT A TIME,
C**********************************************************************
C
C         COMMON AND DIMENSION STATEMENTS
C
      DIMENSION NPI(1),NPJ(1),NPSEC(1),NBLK(1),NFORS(1),NFORSN(NY,1),
     *  ALF(NY,1),NSA(1),NSAVS(NS,1),NPAV(NS,1),NCS(1),NGE(2,1),XDIV(1),
     *  IFD(1),TZ(NT,NM,1),TY(NT,NM,1),DIS(NR,NC),INDCR(NX),IDISBC(ND),
     *  DISPL(ND),BIGK(NR,MB),FORCE(NR,NC),F(NF,1),DD(NC,67),B1(1),
     *  DISP(NQB,NC),GIRMOM(NK,NG,NC),TENS(NK,NG,NC),COMP(NK,NG,NC),
     *  B2(1),B3(1),FF(30,NC),D(30,NC),E(12,NC),DI(NR,NC),NN(4),MM(4),
     *  IOSEG(1),IODIA(1)
C
C     EQUIVALENCE (INDCR,DD),(DIS,F,GIRMOM),(D,E,DI,FF)
C     (THE ARRAY SEQUENCE DIS,INDCR,IDISBC,DISPL,BIGK,FORCE  STARTS AT
C     THE SAME ADDRESS AS THE ARRAY SEQUENCE  GIRMOM,TENS,COMP,B1,B2,
C     B3,D, THE ARRAY SEQUENCE  DD,DISP STARTS AT THE FIRST WORD ADDRESS
C     OF ARRAY  INDCR)
C
      COMMON/SETUP/NSEG,NSECT,NEL,NPTS,NS2PT,NFET,NSE,NDIAPH,NFRAME,
     *  NTAD,NPLD,NCSTYP,NLC,NFORST,NSAV,NSCK,IA,IS,IM,IR,IL,ISTOP,ICHK,
     *  R,NMAX,NEQ,NEQS,NQ,MBAND,NBLCK,NDBC,JFLAG,PI,NDE,NDS,NFST,NFCT,
     *  NCOMM,NBR,IC1,IC2,IC3,IC4,IC5,IC6,NGIR,IREACT,IRESID,
     *  K1,K2,K3,K4,K5,K6
      COMMON/DIAPH/NDTY(20),NPD(4,20),NDSE(20),NDSD(20),NDST(20)
      COMMON/FRAME/EA(20),EIY(20),EIZ(20),GJX(20),GAY(20),GAZ(20),
     *  EI(3,20),EJ(3,20),EK(3,20),NFI(100),NFJ(100),NSFI(100),
     *  NSFJ(100),NFTY(100),NFSC(100)
      COMMON/STIFF/T(24,24),XE(4),YE(4),ZE(4),ECI(3),ECJ(3),ECK(3),
     *  Q(180)
      LOGICAL L1,L2,L3
C
C**********************************************************************
C       1) ADD SPECIFIED DISPLACEMENTS INTO SOLUTION MATRIX
C**********************************************************************
C
      L1=IREACT.EQ.0
      L2=IRESID.EQ.0
      NB=NBLCK
      NMB=NQ+MB-1
      REWIND K1
      REWIND K5
      REWIND K6
      READ (K1) INDCR
      IF(NDBC.GE.1) READ (K1) IDISBC,DISPL
      DO 10 I=1,NB
   10 READ (K6) DIS
C
```

```
      KK=1
      II=0
      DO 30 I=1,NB
      DO 20 J=1,NQ
      JJ=II+J
      IF(JJ.GT.NEQ) GO TO 20
      IF(INDCR(JJ).EQ.0) GO TO 20
      DO 15 K=1,NLC
   15 DIS(J,K)=0.0
   20 CONTINUE
   22 IF(KK.GT.NDBC) GO TO 26
      IK=IDISBC(KK)
      IF(IK.GT.JJ) GO TO 26
      IK=IK-II
      DO 24 K=1,NLC
   24 DIS(IK,K)=DISPL(KK)
      KK=KK+1
      GO TO 22
   26 WRITE (K5) DIS
      IF(I.EQ.NB) GO TO 35
      BACKSPACE K6
      BACKSPACE K6
      READ (K6) DIS
   30 II=II+NQ
C
C**********************************************************************
C         2) FIND RESIDUAL LOAD CHECKS AND REACTIONS
C**********************************************************************
C
C         READ ORIGINAL STRUCTURE STIFFNESS AND LOAD MATRIX FROM TAPE K4
C
   35 IF(L1.AND.L2) GO TO 142
      REWIND K4
      REWIND K5
      REWIND K6
      N1=K2
      N2=K3
      JK=0
      DO 140 I=1,NB
      READ (K4) BIGK,FORCE
      IF(I.EQ.1) GO TO 47
      READ (N1) DIS
      DO 40 J=1,NQ
      DO 40 K=1,NLC
   40 FORCE(J,K)=FORCE(J,K)-DIS(J,K)
      NZ=JJ-1
      DO 45 J=1,NZ
   45 BACKSPACE K5
   47 READ (K5) DIS
C
C         DIAGONAL BLOCK MULTIPLICATION
C
      DO 65 J=1,NQ
      L3=INDCR(JK+J).EQ.1
      IF(L1.AND.L3.OR.L2.AND..NOT.L3) GO TO 65
      JJ=J+1
      KI=MAX0(1,JJ-MB)
      DO 50 K=KI,J
      XX=BIGK(K,JJ-K)
      DO 50 L=1,NLC
```

```
   50 FORCE(J,L)=FORCE(J,L)-XX*DIS(K,L)
      IF(J.EQ.NQ) GO TO 65
      KK=2
      KJ=MIN0(NQ,J+MB-1)
      DO 60 K=JJ,KJ
      XX=BIGK(J,KK)
      IF(XX.EQ.0.) GO TO 60
      DO 55 L=1,NLC
   55 FORCE(J,L)=FORCE(J,L)-XX*DIS(K,L)
   60 KK=KK+1
   65 CONTINUE
C
C        BLOCK MULTIPLICATION FOR BLOCKS TO FOLLOW
C
      IF(I.EQ.NB) GO TO 120
      REWIND N2
      WRITE (N2) FORCE
      JJ=2
      II=1
      KJ=JK+NQ
   67 IF(I.EQ.1.OR.JJ.EQ.NBR) GO TO 70
      READ (N1) FORCE
      GO TO 80
   70 DO 75 K=1,NQ
      DO 75 L=1,NLC
   75 FORCE(K,L)=0.0
C
   80 IJ=II+NQ
      JL=MIN0(NQ,MB-II)
      DO 90 J=1,JL
      L3=INDCR(KJ+J).EQ.1
      IF(L1.AND.L3.OR.L2.AND..NOT.L3) GO TO 90
      JI=IJ+J
      KI=MAX0(1,JI-MB)
      DO 85 K=KI,NQ
      XX=BIGK(K,JI-K)
      DO 85 L=1,NLC
   85 FORCE(J,L)=FORCE(J,L)+XX*DIS(K,L)
   90 CONTINUE
      WRITE (N2) FORCE
      IF(JJ.EQ.NBR.OR.I+JJ.GT.NB) GO TO 100
      JJ=JJ+1
      II=II+NQ
      KJ=KJ+NQ
      GO TO 67
C
C        REMAINING MULTIPLICATIONS FOR BLOCK I
C
  100 REWIND N2
      READ (N2) FORCE
      JJ=2
      II=NQ+1
C
  105 READ (K5) DIS
      JL=MAX0(1,II-MB+1)
      DO 115 J=JL,NQ
      L3=INDCR(JK+J).EQ.1
      IF(L1.AND.L3.OR.L2.AND..NOT.L3) GO TO 115
      JI=II-J
      KI=MIN0(NQ,MB-JI)
```

```
      DO 110 K=1,KI
      XX=BIGK(J,JI+K)
      DO 110 L=1,NLC
  110 FORCE(J,L)=FORCE(J,L)-XX*DIS(K,L)
  115 CONTINUE
      IF(JJ.EQ.NBR.OR.JJ+I-1.EQ.NB) GO TO 120
      JJ=JJ+1
      II=II+NQ
      GO TO 105
C
C         SEPARATE RESIDUAL LOADS FROM REACTIONS
C
  120 DO 125 J=1,NQ
      DO 125 K=1,NLC
  125 DIS(J,K)=0.0
C
      DO 135 J=1,NQ
      IF(INDCR(JK+J).EQ.0) GO TO 135
      DO 130 K=1,NLC
      DIS(J,K)=-FORCE(J,K)
  130 FORCE(J,K)=0.0
  135 CONTINUE
C
C         WRITE RESIDUAL LOADS AND REACTIONS ON TAPE K6
C
      IF(.NOT.L1) WRITE (K6) DIS
      IF(.NOT.L2) WRITE (K6) FORCE
      NI=N1
      N1=N2
      N2=NI
  140 JK=JK+NQ
C
C*********************************************************************
C         3) CALCULATE INTERNAL FORCES FOR PLATE ELEMENTS
C*********************************************************************
C
  142 REWIND K2
      NBL=NQB/NQ
      LL=1
  145 REWIND K5
      JI=1
      DO 150 I=1,NBL
      JJ=JI+NQ-1
      READ (K5)((DISP(J,K), J=JI,JJ), K=1,NLC)
  150 JI=JI+NQ
      IK=0
      JJ=0
      JK=NBL
      IF(LL.EQ.2) GO TO 175
C
      DO 170 I=1,NSEG
      DO 151 J=I,NSEG
      IF(IOSEG(J).NE.3) GO TO 154
  151 CONTINUE
      GO TO 171
  154 NI=NBLK(I+2)
      CALL LOCDIS(DISP,NQ,NQB,NB,NBL,NLC,NBLK(I),NI,IK,JK,JJ,I,KL,K5)
      IF(IOSEG(I).EQ.3) GO TO 165
      NI=NPSEC(I)
      JL=0
```

```
      IJ=0
      IF(NFORST.LE.0) GO TO 153
      DO 152 J=1,NFORST
      LL=NFORS(J)
      DO 152 K=1,LL
      JK=NFORSN(J,K)
      IF(JK.EQ.I) JL=J
      IF(JK.EQ.I+1) IJ=J
  152 CONTINUE
C
  153 DO 160 J=1,NEL
      NN(1)=NPI(J)
      NN(2)=NPJ(J)
      NN(3)=NPJ(J)+NI
      NN(4)=NPI(J)+NI
      CALL ELDIS (4,KL,NQB,NLC,NQ,NN,DD,DISP,DIS,K5)
      CALL INTFOR (DD,NLC,IL,K1)
C
C        TRANSFORM INTERNAL FORCES ABOUT ANGLE ALF IF DESIRED
C
      LL=0
      KK=JL
      KKK=1
      MM(1)=NN(1)
      MM(2)=NN(2)
      IF(JL.EQ.0) GO TO 159
  155 DO 158 K=1,2
      JK=MM(K)
      XX=ALF(KK,JK)*PI/180.
      S=SIN(XX)
      C=COS(XX)
      SS=S*S
      CC=C*C
      SC=S*C
      II=LL-2
  156 II=II+3
      DO 157 L=1,NLC
      T1=CC*DD(L,II)+SS*DD(L,II+1)-2.*SC*DD(L,II+2)
      T2=SS*DD(L,II)+CC*DD(L,II+1)+2.*SC*DD(L,II+2)
      DD(L,II+2)=SC*(DD(L,II)-DD(L,II+1))+(CC-SS)*DD(L,II+2)
      DD(L,II+1)=T2
  157 DD(L,II)=T1
      IF(II.EQ.LL+1) GO TO 156
  158 LL=LL+6
C
  159 IF(IJ.EQ.0.OR.KKK.EQ.2) GO TO 160
      IF(KK.EQ.0) LL=12
      KK=IJ
      KKK=2
      MM(2)=NN(1)
      MM(1)=NN(2)
      GO TO 155
  160 WRITE (K2)((DD(L,K), K=1,30), L=1,NLC)
      GO TO 170
  165 DO 167 J=1,NEL
  167 READ (K1)
  170 CONTINUE
C
C**************************************************************************
C        4) CALCULATE INTERNAL FORCES FOR DIAPHRAGMS AND FRAMES
```

```
C*********************************************************************
C
C         DIAPHRAGM ELEMENTS
C
  171 IF(NDIAPH+NFRAME.LE.0) GO TO 201
      REWIND K3
      LL=2
      J=0
      DO 172 I=1,NSECT
      IF(IFD(I).LT.2) GO TO 172
      J=J+1
      IF(IODIA(J).NE.1) IFD(I)=IFD(I)-2
  172 CONTINUE
C
      GO TO 145
  175 DO 200 I=1,NSECT
      IF(IFD(I).EQ.0) GO TO 200
      IF(IFD(I).EQ.1) GO TO 190
      NI=NBLK(I+1)
      CALL LOCDIS(DISP,NQ,NQB,NB,NBL,NLC,NBLK(I),NI,IK,JK,JJ,I,II,K5)
      DO 185 J=1,NDE
      DO 180 K=1,4
  180 NN(K)=NPD(K,J)
      CALL ELDIS (4,II,NQB,NLC,NQ,NN,DD,DISP,DIS,K5)
      CALL INTFOR (DD,NLC,1,K1)
  185 WRITE (K3)((DD(L,K),K=1,30),L=1,NLC)
C
C         FRAME ELEMENTS
C
  190 IF(IFD(I).EQ.2) GO TO 200
      NI=NBLK(I+2)
      IF(I.EQ.NSECT) NI=NBLK(I+1)
      CALL LOCDIS(DISP,NQ,NQB,NB,NBL,NLC,NBLK(I),NI,IK,JK,JJ,I,II,K5)
      DO 195 J=1,NFRAME
      K=NSFI(J)
      L=NSFJ(J)
      IF(K.NE.I.AND.L.NE.I) GO TO 195
      IF(K.LT.I.OR.L.LT.I) GO TO 195
      NN(1)=NFI(J)
      IF(K.GT.I) NN(1)=NN(1)+NPSEC(I)
      NN(2)=NFJ(J)
      IF(L.GT.I) NN(2)=NN(2)+NPSEC(I)
      CALL ELDIS (2,II,NQB,NLC,NQ,NN,DD,DISP,DIS,K5)
      CALL ELSTIF (S1,S2,S3,S4,S5,S6,2,0,IL,DD,NLC)
      WRITE(K3) ((DD(L,K), K=1,12), L=1,NLC)
  195 CONTINUE
  200 CONTINUE
C
C*********************************************************************
C         5) AVERAGE INTERNAL PLATE FORCES WHERE REQUESTED
C*********************************************************************
C
  201 IF(NSAV.EQ.0) GO TO 224
      IF(JFLAG.EQ.1) GO TO 222
      LA=NLC+1
      LB=NLC+NLC
      REWIND K1
      REWIND K2
C
      DO 220 I=1,NSECT
```

```
      II=0
      DO 203 J=1,NSAV
      JJ=NSA(J)
      DO 202 K=1,JJ
      IF(NSAVS(J,K),NE,I) GO TO 202
      II=J
      GO TO 204
  202 CONTINUE
  203 CONTINUE
C
  204 IF(I,GT,1) GO TO 207
  205 IF(IOSEG(I).EQ,3) GO TO 220
      IJ=1
      GO TO 211
  206 IJ=2
      GO TO 213
  207 IF(IOSEG(I-1),NE,3) GO TO 208
      IF(I,EQ,NSECT) GO TO 220
      GO TO 205
  208 IF(I,EQ,NSECT) GO TO 206
      IF(IOSEG(I).EQ,3) GO TO 206
      IJ=3
C
C        TRANSVERSE AVERAGING
C
  211 JI=1
      DO 212 J=1,NEL
      JJ=JI+29
      READ (K2)((F(K,L), K=JI,JJ), L=LA,LB)
  212 JI=JI+30
C
  213 IF(II.EQ,0) GO TO 215
      CALL AVRAGE (NPAV,F,NPI,NPJ,NEL,NPTS,II,IJ,NSAV,NF,1,NLC)
      GO TO (217,215,214), IJ
C
C        LONGITUDINAL AVERAGING
C
  214 CALL AVRAGE (NPAV,F,NPI,NPJ,NEL,NPTS,II,4,NSAV,NF,0,NLC)
  215 IF(IJ,EQ,1) GO TO 217
      JI=1
      DO 216 J=1,NEL
      JJ=JI+29
      WRITE (K1)((F(K,L), K=JI,JJ), L=1,NLC)
  216 JI=JI+30
C
      IF(IJ,EQ,2) GO TO 220
  217 DO 218 K=1,NF
      DO 218 L=1,NLC
  218 F(K,L)=F(K,L+NLC)
  220 CONTINUE
      GO TO 224
  222 PRINT 2015
C
C*************************************************************************
C        6) CALCULATE MOMENTS FOR STATIC CHECKS
C*************************************************************************
C
  224 KL=K1
      IF(NSAV,EQ,0,OR,JFLAG,EQ,1) KL=K2
      IF(NSCK,EQ,0) GO TO 251
```

```
      REWIND KL
      DO 226 I=1,NSCK
      DO 226 J=1,NGIR
      DO 226 K=1,NLC
      GIRMOM(I,J,K)=0.0
      TENS(I,J,K)=0.0
  226 COMP(I,J,K)=0.0
C
      JJ=0
      IK=0
      DO 250 I=1,NSECT
      DO 228 J=1,NSCK
      IF(NCS(J).EQ.I) GO TO 230
  228 CONTINUE
      GO TO 248
C
  230 IF(I.EQ.NSECT) JJ=JJ-1
      IF(JJ.LE.0) GO TO 234
      IJ=JJ*NEL
      DO 232 J=1,IJ
  232 READ (KL) FF
      JJ=0
  234 IK=IK+1
      IF(I.LT.NSECT.OR.JJ.GE.0) GO TO 236
      DO 235 J=1,NEL
  235 BACKSPACE KL
  236 DO 246 J=1,NEL
      READ (KL) FF
      II=NPI(J)
      IJ=NPJ(J)
      IF(I.EQ.NSECT) GO TO 239
      M1=1
      M2=7
      GO TO 240
  239 M1=19
      M2=13
  240 ZI=TZ(I,II,1)
      ZJ=TZ(I,IJ,1)
      YI=TY(I,II,1)
      YJ=TY(I,IJ,1)
      N1=NGE(1,J)
      N2=NGE(2,J)
      YH=YJ-YI
      ZH=ZJ-ZI
      XX=SQRT(YH*YH+ZH*ZH)
      IF(N2.NE.0.AND.YH.NE.0.) GO TO 242
C
      DO 241 L=1,NLC
  241 CALL ADDMOM (ZI,ZJ,XX,YH,FF(M1,L),FF(M2,L),FF(M1+3,L),FF(M2+3,L),
     *  GIRMOM(IK,N1,L),TENS(IK,N1,L),COMP(IK,N1,L))
      GO TO 246
C
  242 AA=XDIV(J)/YH
      XM=AA*XX
      ZM=ZI+AA*ZH
      DO 244 L=1,NLC
      F1=FF(M1,L)+AA*(FF(M2,L)-FF(M1,L))
      F2=FF(M1+3,L)+AA*(FF(M2+3,L)-FF(M1+3,L))
      CALL ADDMOM (ZI,ZM,XM,XDIV(J),FF(M1,L),F1,FF(M1+3,L),F2,
     *  GIRMOM(IK,N1,L),TENS(IK,N1,L),COMP(IK,N1,L))
```

```
  244 CALL ADDMOM (ZM,ZJ,XX-XM,YH-XDIV(J),F1,FF(M2,L),F2,FF(M2+3,L),
     *  GIRMOM(IK,N2,L),TENS(IK,N2,L),COMP(IK,N2,L))
  246 CONTINUE
      JJ=0
      GO TO 250
  248 IF(IOSEG(I).NE.3) JJ=JJ+1
  250 CONTINUE
C
C***********************************************************************
C         7) ORDER OUTPUT ACCORDING TO LOAD CASES
C***********************************************************************
C
  251 REWIND K4
  252 DO 300 L=1,NLC
      REWIND KL
      REWIND K3
      REWIND K5
      REWIND K6
      JJ=0
C
      DO 280 I=1,NSECT
      II=NBLK(I)-JJ
      IJ=NBLK(I+1)-JJ-1
      IK=II
      K=0
  254 JJ=JJ+NQ
      JK=IJ
      IF(JK.GT.NQ) JK=NQ
      KK=K
      READ (K5) DI
      DO 256 J=IK,JK
      K=K+1
  256 B1(K)=DI(J,L)
C
      IF(L1) GO TO 259
      K=KK
      READ (K6) DI
      DO 258 J=IK,JK
      K=K+1
  258 B2(K)=DI(J,L)
C
  259 IF(L2) GO TO 261
      K=KK
      READ (K6) DI
      DO 260 J=IK,JK
      K=K+1
  260 B3(K)=DI(J,L)
  261 IF(JK.EQ.IJ) GO TO 262
      IK=1
      IJ=IJ-NQ
      GO TO 254
C
  262 WRITE (K4)(B1(J), J=1,K)
      IF(.NOT.L1) WRITE (K4)(B2(J), J=1,K)
      IF(.NOT.L2) WRITE (K4)(B3(J), J=1,K)
      IF(JK.EQ.NQ) GO TO 264
      BACKSPACE K5
      IF(.NOT.L1) BACKSPACE K6
      IF(.NOT.L2) BACKSPACE K6
      JJ=JJ-NQ
```

```
C
  264 IF(I.EQ.NSECT) GO TO 268
      DO 266 J=1,NEL
      IF(IOSEG(I).LT.3) READ (KL) D
      IF(IOSEG(I).EQ.1) WRITE (K4)(D(K,L), K=1,30)
  266 CONTINUE
C
  268 IF(IFD(I).EQ.0) GO TO 280
      IF(IFD(I).LT.2) GO TO 272
      DO 270 J=1,NDE
      READ (K3) D
  270 WRITE (K4)(D(K,L), K=1,30)
C
  272 IF(IFD(I).EQ.2) GO TO 280
      DO 275 J=1,NFRAME
      KI=NSFI(J)
      KJ=NSFJ(J)
      IF(KI.NE.I.AND.KJ.NE.I) GO TO 275
      IF(KI.LT.I.OR.KJ.LT.I) GO TO 275
      READ (K3) E
      WRITE (K4)(E(K,L), K=1,12)
  275 CONTINUE
C
  280 CONTINUE
  300 CONTINUE
C
C***********************************************************************
C          B) PRINT ALL OUTPUT FOR ONE LOAD CASE AT A TIME
C***********************************************************************
C
C         DISPLACEMENTS
C
      REWIND K4
      DO 400 L=1,NLC
      PRINT 2000,L
      DO 395 I=1,NSECT
      IF(I.GT.1) PRINT 2001
      II=NPSEC(I)
      JJ=II*6
      READ (K4)(B1(J), J=1,JJ)
      PRINT 2002, I
      IJ=1
      DO 310 J=1,II
      IK=IJ+5
      PRINT 2003, J, (B1(K), K=IJ,IK)
  310 IJ=IJ+6
C
C          REACTIONS
C
      IF(L1) GO TO 317
      READ (K4)(B1(J), J=1,JJ)
      PRINT 2004
      IJ=1
      DO 315 J=1,II
      IK=IJ+5
      PRINT 2003, J, (B1(K), K=IJ,IK)
  315 IJ=IJ+6
C
C          RESIDUAL LOADS
C
```

```
  317 IF(L2) GO TO 325
      READ (K4)(B1(J), J=1,JJ)
      PRINT 2005
      IJ=1
      DO 320 J=1,II
      IK=IJ+5
      PRINT 2003, J, (B1(K), K=IJ,IK)
  320 IJ=IJ+6
C
C         PLATE FORCES
C
  325 IF(I.EQ.NSECT) GO TO 340
      IF(IOSEG(I).NE.1) GO TO 340
      PRINT 2006, I,L
      MM(1)=I
      MM(2)=I
      MM(3)=I+1
      MM(4)=I+1
      DO 335 J=1,NEL
      NN(1)=NPI(J)
      NN(2)=NPJ(J)
      NN(3)=NPJ(J)
      NN(4)=NPI(J)
      READ (K4)(B1(K), K=1,30)
      PRINT 2007, J
      IJ=1
      DO 330 K=1,4
      JJ=IJ+5
      PRINT 2008, NN(K),MM(K),(B1(JK), JK=IJ,JJ)
  330 IJ=IJ+6
  335 PRINT 2009, (B1(JK), JK=25,30)
C
C         DIAPHRAGM FORCES
C
  340 IF(IFD(I).EQ.0) GO TO 375
      IF(IFD(I).LT.2) GO TO 360
      PRINT 2010, I,L
      DO 350 J=1,NDE
      READ (K4)(B1(K), K=1,30)
      PRINT 2011, J
      IJ=1
      DO 345 K=1,4
      JJ=IJ+5
      PRINT 2012, NPD(K,J), (B1(JK), JK=IJ,JJ)
  345 IJ=IJ+6
  350 PRINT 2009, (B1(JK), JK=25,30)
C
C         FRAME FORCES
C
  360 IF(IFD(I).EQ.2) GO TO 375
      PRINT 2013, I,L
      DO 370 J=1,NFRAME
      KI=NSFI(J)
      KJ=NSFJ(J)
      IF(KI.NE.I.AND.KJ.NE.I) GO TO 370
      IF(KI.LT.I.OR.KJ.LT.I) GO TO 370
      READ (K4)(B1(K), K=1,12)
      PRINT 2014, J
      PRINT 2008, NFI(J),KI,(B1(K), K=1,6)
      PRINT 2008, NFJ(J),KJ,(B1(K), K=7,12)
```

```
  370 CONTINUE
C
C         GIRDER MOMENTS
C
  375 IF(NSCK.LE.0) GO TO 395
      DO 380 J=1,NSCK
      IF(NCS(J).NE.I) GO TO 380
      IK=J
      GO TO 385
  380 CONTINUE
      GO TO 395
  385 P1=0.0
      P2=0.0
      P3=0.0
      P4=0.0
      DO 390 J=1,NGIR
      P2=P2+GIRMOM(IK,J,L)
      P3=P3+TENS(IK,J,L)
  390 P4=P4+COMP(IK,J,L)
      IF(P2.EQ.0.) GO TO 394
      PRINT 2016, I
      DO 392 J=1,NGIR
      P5=GIRMOM(IK,J,L)/P2*100.
      P1=P1+P5
  392 PRINT 2017, J,GIRMOM(IK,J,L),P5,TENS(IK,J,L),COMP(IK,J,L)
      PRINT 2018, P2,P1,P3,P4
      GO TO 395
  394 PRINT 2019, I
  395 CONTINUE
  400 CONTINUE
C
C***********************************************************************
C         FORMAT STATEMENTS
C***********************************************************************
C
 2000 FORMAT(1H1,27H   OUTPUT FOR LOAD CASE NO.,I6/////)
 2001 FORMAT(1H1)
 2002 FORMAT(50X,12H SECTION NO.,I5///26H NODAL POINT DISPLACEMENTS//
     * 112H NODAL POINT   X-DISPLACEMENT   Y-DISPLACEMENT   Z-DISPLACEME
     *NT     X-ROTATION        Y-ROTATION        Z-ROTATION)
 2003 FORMAT(I7,5X,6E17.6)
 2004 FORMAT(////10H REACTIONS//112H NODAL POINT         X-FORCE
     *Y-FORCE           Z-FORCE           X-MOMENT          Y-MOMENT
     *Z-MOMENT    )
 2005 FORMAT(////15H RESIDUAL LOADS//112H NODAL POINT         X-FORCE
     *     Y-FORCE           Z-FORCE           X-MOMENT          Y-MOMENT
     *      Z-MOMENT    )
 2006 FORMAT(1H1,49H INTERNAL FORCES IN PLATE ELEMENTS OF SEGMENT NO.,
     * I6,45X,14H LOAD CASE NO.,I5)
 2007 FORMAT(////10X,12H ELEMENT NO.,I5//120H N.POINT   SECTION        NX
     *-FORCE           NY-FORCE          NXY-FORCE         MX-MOMENT        M
     *Y-MOMENT        MXY-MOMENT    )
 2008 FORMAT(I5,I10,3X,6E17.6)
 2009 FORMAT(18H MIDELEMENT NODE   ,6E17.6)
 2010 FORMAT(1H1,53H INTERNAL FORCES IN DIAPHRAGM ELEMENTS AT SECTION NO
     *.,I6,41X,14H LOAD CASE NO.,I5)
 2011 FORMAT(////10X,12H ELEMENT NO.,I5//120H NODAL POINT             NX
     *-FORCE           NY-FORCE          NXY-FORCE         MX-MOMENT        M
     *Y-MOMENT        MXY-MOMENT    )
 2012 FORMAT(I7,11X,6E17.6)
```

```
 2013 FORMAT(1H1,49H INTERNAL FORCES IN FRAME ELEMENTS AT SECTION NO.,
     * I6,45X,14H LOAD CASE NO.,I5)
 2014 FORMAT(////10X,12H ELEMENT NO.,I5//120H N.POINT   SECTION     AXI
     *AL FORCE          Y-SHEAR             Z-SHEAR            TORQUE
     *  Y-MOMENT          Z-MOMENT    )
 2015 FORMAT(////61H SORRY - NOT ENOUGH CORE STORAGE FOR INTERNAL FORCE
     *AVERAGING/8X,49H (FOR THIS SIZE OF A PROBLEM IT-S QUITE A JOB...))
 2016 FORMAT(/////60H GIRDER MOMENTS AND AXIAL STRESS RESULTANTS AT SECT
     *ION   I =I4//62H GIRDER NO.   MOMENT   PERCENTAGE     TENSION
     * COMPRESSION)
 2017 FORMAT(I6,E16.6,F9.2,2E16.6)
 2018 FORMAT(//6H TOTAL,E16.6,F9.2,2E16.6)
 2019 FORMAT(////27H STATICAL MOMENT AT SECTION,I6,10H   IS ZERO)
      RETURN
      END


      SUBROUTINE ELDIS (M,KK,NQB,NLC,NQ,NN,DD,DISP,DIS,KT)
C
C*********************************************************************
C     THIS SUBROUTINE EXTRACTS THE NODAL DISPLACEMENTS FOR ANY
C     STRUCTURAL ELEMENT FROM DISPLACEMENT MATRIX DISPL
C*********************************************************************
C
      DIMENSION NN(4),DD(NLC,67),DISP(NQB,NLC),DIS(NQ,NLC)
      MM=6*M
      NB=NQB
      JJ=0
      IJ=0
      DO 30 I=1,M
      II=6*NN(I)-6+KK
      DO 20 J=1,6
      IF(II+J.GT.NQB) GO TO 30
      DO 10 K=1,NLC
   10 DD(K,IJ+J)=DISP(II+J,K)
   20 JJ=JJ+1
   30 IJ=IJ+6
      IF(JJ.EQ.MM) RETURN
C
C         SEARCH FOR REMAINING DISPLACEMENTS IN SUBSEQUENT BLOCKS
C         TO BE TEMPORARILY STORED IN BUFFER ARRAY  DIS
C
      JK=1
   35 READ (KT) DIS
      IJ=0
      DO 50 I=1,M
      II=6*NN(I)-6+KK
      DO 45 J=1,6
      IN=II+J-NB
      IF(IN.LE.0) GO TO 45
      IF(IN.GT.NQ) GO TO 50
      DO 40 K=1,NLC
   40 DD(K,IJ+J)=DIS(IN,K)
      JJ=JJ+1
   45 CONTINUE
   50 IJ=IJ+6
      IF(JJ.EQ.MM) GO TO 60
      JK=JK+1
      NB=NB+NQ
      GO TO 35
```

```
C
   60 DO 70 I=1,JK
   70 BACKSPACE KT
      RETURN
      END


      SUBROUTINE LOCDIS (DISP,NQ,NQB,NB,NBL,NLC,N1,N2,IK,JK,JJ,I,II,KT)
C
C***********************************************************************
C     THIS SUBROUTINE SHIFTS THE EQUATION BLOCK IN WHICH THE DISPLACE-
C     MENTS FOR SEGMENT/SECTION  I  START, INTO THE FIRST BLOCK OF THE
C     DISP-ARRAY, IF POSSIBLE OR NECESSARY, AND FILLS THE REST OF THE
C     DISP-ARRAY WITH THE SUCCEEDING BLOCKS. OUTPUT QUANTITY  * II *  IS
C     THE ROW NUMBER MINUS 1 IN DISP-ARRAY OF FIRST EQUATION OF SECTION/
C     SEGMENT  I.
C***********************************************************************
C
      DIMENSION DISP(NQB,NLC)
      II=N1-1
      IF(NBL.EQ.NB.OR.I.EQ.1) GO TO 50
      KI=N2-JJ-1
      IF(KI.LE.NQB) GO TO 50
      IJ=N1/NQ-IK
      IF(IJ.EQ.0) GO TO 50
      NI=(NBL-IJ)*NQ
      JI=IJ*NQ
      IF(NI) 25,20,10
   10 DO 15 J=1,NI
      DO 15 K=1,NLC
   15 DISP(J,K)=DISP(J+JI,K)
   20 IK=IK+IJ
      KI=NI+1
      GO TO 35
   25 KI=IJ-NBL+1
      IF(JK+KI.GT.NB) KI=NB-JK
      DO 30 J=1,KI
   30 READ (KT)((DISP(K,L), K=1,NQ), L=1,NLC)
      JK=JK+KI
      IF(NBL.EQ.1) GO TO 45
      KI=NQ+1
      IK=IK+IJ
      IJ=NBL-1
   35 IF(JK+IJ.GT.NB) IJ=NB-JK
      DO 40 J=1,IJ
      KJ=KI+NQ-1
      READ (KT)((DISP(K,L), K=KI,KJ), L=1,NLC)
   40 KI=KI+NQ
      JK=JK+IJ
   45 JJ=JJ+JI
   50 II=II-JJ
      RETURN
      END


      SUBROUTINE ADDMOM (X1,X2,XL,XH,XN1,XN2,XM1,XM2,G,T,C)
C
      F1=.5*XN1*XL
      F2=.5*XN2*XL
      G=G+(F1*(X2+X1+X1)+F2*(X1+X2+X2))/3.
```

```
      G=G+.5*(XM1+XM2)*XH
C
      IF(F2.EQ.0.) GO TO 10
      IF(F1/F2.GE.0.) GO TO 10
      X=ABS(XL*XN1/(XN1-XN2))
      F1=.5*XN1*X
      F2=.5*XN2*(XL-X)
   10 IF(F1.GT.0.) GO TO 20
      C=C+F1
      GO TO 30
   20 T=T+F1
   30 IF(F2.GT.0.) GO TO 40
      C=C+F2
      GO TO 50
   40 T=T+F2
   50 RETURN
      END


      SUBROUTINE AVRAGE (NPAV,F,NPI,NPJ,NEL,NPTS,II,KA,NS,NF,MM,NLC)
C
C************************************************************************
C          THIS SUBROUTINE AVERAGES INTERNAL FORCES IN PLATE ELEMENTS
C            FOR  MM=1   -   TRANSVERSE AVERAGING
C                 MM=0   -   LONGITUDINAL AVERAGING
C************************************************************************
C
      DIMENSION NPAV(NS,1),F(NF,1),NPI(1),NPJ(1),NN(5)
      LOGICAL LI(6)
C
      DO 100 J=1,NPTS
      NTEMP=NPAV(II,2*J-MM)
      IF(NTEMP.EQ.0) GO TO 100
      IJ=0
      DO 10 K=1,NEL
      IF(NPI(K).NE.J.AND.NPJ(K).NE.J) GO TO 10
      IJ=IJ+1
      IF(IJ.GT.5) GO TO 100
      IF(IJ.GT.2.AND.KA.LT.4) GO TO 100
      NN(IJ)=K
      IF(NPJ(K).EQ.J) NN(IJ)=-K
   10 CONTINUE
      IF(IJ.NE.2.AND.KA.LT.4) GO TO 100
C
      M=6
      DO 20 K=1,6
      KK=M-K
      KL=NTEMP/10**KK
      LI(K)=.TRUE.
      IF(KL.EQ.1) LI(K)=.FALSE.
   20 NTEMP=NTEMP-KL*10**KK
C
   25 KC=1
      IF(KA.EQ.4) KC=IJ
      DO 70 N=1,KC
      IF(KA.EQ.2) GO TO 60
      IA=30
      IB=24
      IC=NLC
   30 K1=NN(N)*30-IA
```

```
      IF(NN(N).LT.0) K1=-NN(N)*30-IB
      IF(KA.EQ.4) GO TO 35
      K2=NN(2)*30-IA
      IF(NN(2).LT.0) K2=-NN(2)*30-IB
      ID=IC
      GO TO 40
   35 K2=K1+18
      IF(NN(N).LT.0) K2=K1+6
      ID=0
   40 DO 50 K=1,6
      IF(LI(K)) GO TO 50
      DO 45 L=1,NLC
      F(K1+K,L+IC)=0.5*(F(K1+K,L+IC)+F(K2+K,L+ID))
   45 F(K2+K,L+ID)=F(K1+K,L+IC)
   50 CONTINUE
      IF(KA.EQ.1.OR.IA.EQ.12) GO TO 100
      IF(KA.EQ.4) GO TO 70
   60 IA=12
      IB=18
      IC=0
      GO TO 30
   70 CONTINUE
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INTFOR (DIS,NLC,IL,N)                                  INTF
C                                                                       INTF
C***********************************************************************INTF
C     THIS SUBROUTINE CALCULATES THE INTERNAL STRESS RESULTANTS AND     INTF
C     MOMENTS AT THE FOUR CORNER POINTS AND THE MIDPOINT OF A GENERAL   INTF
C     QUADRILATERAL ELEMENT WITH ARBITRARY ORIENTATION IN SPACE, FOR    INTF
C                    NLC  DIFFERENT LOAD CASES,                         INTF
C                                                                       INTF
C                               -  INPUT  -                             INTF
C     DIS(NLC,24) - GLOBAL NODAL DISPLACEMENT MATRIX, ORDERED AS FOLLOWSINTF
C                    (U(I),V(I),W(I),XROT(I),YROT(I),ZROT(I),I=I,J,K,L) INTF
C     C1,...,C6   - PLANE STRESS MATERIAL LAW                           INTF
C     D(3,3)      - PLATE BENDING MATERIAL LAW                          INTF
C     A(3,3)      - DIRECTION COSINES OF ELEMENT AXES                   INTF
C     X(5),Y(5)   - ELEMENT COORDINATES OF 4 CORNER POINTS AND MIDPOINT INTF
C     S(19,7)     - PORTION OF CONDENSED PLATE BENDING STIFFNESS        INTF
C                   NOTE - ALL INPUT EXCEPT THE DIS-MATRIX IS STORED ON INTF
C                   TAPE N AS ONE RECORD, EQUIVALENCED TO DUMMY ARRAY XTINTF
C     NLC         - NUMBER OF LOAD CASES                                INTF
C                                                                       INTF
C                               -  OUTPUT  -                            INTF
C     DIS(NLC,30) - STRESS RESULTANT MATRIX, ORDERED ROW-WISE AS        INTF
C                    (NX(I),NY(I),NXY(I),MX(I),MY(I),MXY(I),I=I,J,K,L,O)INTF
C***********************************************************************INTF
C                                                                       INTF
C         DIMENSION, EQUIVALENCE, AND DATA STATEMENTS                   INTF
C                                                                       INTF
      DIMENSION XT(167),ST(19,7),D(3,3),X(5),Y(5),A(3,3),U(24),         INTF
     *  DIS(NLC,67),IPERM(24),JPERM(3)                                  INTF
      EQUIVALENCE (XT(1),ST),(XT(134),C1),(XT(135),C2),(XT(136),C3),    INTF
     *  (XT(137),C4),(XT(138),C5),(XT(139),C6),(XT(140),D),(XT(149),X), INTF
     *  (XT(154),Y),(XT(159),A)                                         INTF
      LOGICAL M                                                         INTF
      DATA IPERM /1,7,13,19,2,8,14,20,6,12,18,24,                       INTF
     *            3,4,5,9,10,11,15,16,17,21,22,23/,JPERM/2,1,3/         INTF
C                                                                       INTF
C         TRANSFORM DISPLACEMENTS FROM GLOBAL TO ELEMENT COORDINATES    INTF
C                                                                       INTF
      READ (N) XT                                                       INTF
      T2=X(3)-X(4)                                                      INTF
      T1=Y(3)-Y(4)                                                      INTF
      M=.TRUE.                                                          INTF
      IF(IL.EQ.1) GO TO 5                                               INTF
      M=.FALSE.                                                         INTF
      SS=SQRT(T1*T1+T2*T2)                                              INTF
      S=T1/SS                                                           INTF
      C=T2/SS                                                           INTF
    5 DO 50 L=1,NLC                                                     INTF
      II=0                                                              INTF
      DO 20 I=1,8                                                       INTF
      IF(M.OR.I.LE.4) GO TO 8                                           INTF
      T2=DIS(L,II+1)*C+DIS(L,II+2)*S                                    INTF
      DIS(L,II+2)=-DIS(L,II+1)*S+DIS(L,II+2)*C                          INTF
      DIS(L,II+1)=T2                                                    INTF
    8 DO 10 J=1,3                                                       INTF
   10 U(J)=A(J,1)*DIS(L,II+1)+A(J,2)*DIS(L,II+2)+A(J,3)*DIS(L,II+3)     INTF
      DO 15 J=1,3                                                       INTF
   15 DIS(L,II+J)=U(J)                                                  INTF
   20 II=II+3                                                           INTF
```

```
C
C         SEPARATE IN-PLANE FROM BENDING DISPLACEMENT COMPONENTS
C
      DO 25 I=1,24
   25 U(I)=DIS(L,I)
      DO 30 I=1,24
      J=IPERM(I)
   30 DIS(L,I)=U(J)
   50 CONTINUE
C
C         CALCULATE BENDING MOMENTS AND IN-PLANE STRESS RESULTANTS
C
      CALL FPLATE (X,Y,D,ST,DIS(1,13),DIS(1,32),DIS(1,44),DIS(1,53),
     *  DIS(1,38),NLC)
      CALL PLAG12 (X,Y,C1,C2,C3,C4,C5,C6,DIS(1,1),DIS(1,53),NLC)
C
C         REORDER STRESSES AND MOMENTS ACCORDING TO NODAL POINTS
C
      DO 60 L=1,NLC
      DO 60 I=1,5
      II=I*6-6
      IJ=I*3+34
      DO 60 J=1,3
      K=JPERM(J)+IJ
      DIS(L,II+J)=DIS(L,K+15)
   60 DIS(L,II+J+3)=DIS(L,K)
C
C         TRANSFORMATION TO LOCAL COORDINATES OF EDGE K-L
C
      IF(T1.EQ.0.) GO TO 100
      SS=S*S
      CC=C*C
      SC=S*C
      CS=CC-SS
      DO 80 L=1,NLC
      DO 70 I=13,24,3
      T1=CC*DIS(L,I)+SS*DIS(L,I+1)-2.*SC*DIS(L,I+2)
      T2=SS*DIS(L,I)+CC*DIS(L,I+1)+2.*SC*DIS(L,I+2)
      DIS(L,I+2)=SC*(DIS(L,I)-DIS(L,I+1))+CS*DIS(L,I+2)
      DIS(L,I)=T1
   70 DIS(L,I+1)=T2
   80 CONTINUE
C
  100 RETURN
      END


      SUBROUTINE PLAG12 (X,Y,D1,D2,D3,D4,D5,D6,DIS,ST,NLC)
C
C***********************************************************************
C     THIS SUBROUTINE CALCULATES INTERNAL STRESSES OF A PLANE STRESS
C     QUADRILATERAL ELEMENT WITH 3 DEGREES OF FREEDOM PER CORNER NODE.
C
C                             - INPUT -
C     D1,...,D6 - ELEMENTS OF THE CONSTITUTIVE MATRIX
C     X(5),Y(5) - ELEMENT COORDINATES OF 4 CORNER POINTS AND MIDPOINT
C     DIS(NLC,12)-DISPLACEMENT MATRIX FOR -NLC- LOAD CASES, ORDERED
C                 ROW-WISE AS   (U(I),U(J),U(K),U(L),V(I),...,ZROT(L))
C     NLC        - NUMBER OF DIFFERENT LOAD CASES
C
```

```
C                                - OUTPUT -
C      DIS(NLC,15)-NODAL POINT STRESS RESULTANTS, ORDERED ROW-WISE AS
C                  NX(I),NY(I),NXY(I),NX(J),...,NXY(0)
C**********************************************************************
C
C      DIMENSION, EQUIVALENCE, AND DATA STATEMENTS
C
       DIMENSION   DIS(NLC,1) ,X(4),Y(4),P(4,2),DC(4,2),A(2,2),ETA(2),
      1     Q(4,2),R(4,2),EXX(4),EYY(4),EXY(4),EYX(4),EXZ(4),EYZ(4),
      2     EZZ(4),X1(4),X2(4),Y1(4),Y2(4),TR(4,8),IM(4),IN(4),FT3(4),
      3     FT6(4),FT7(4),PSI(4,2),PHI(4,2),EX(4,2),EY(4,2),AX(5),AY(5),
      4     ST(NLC,1)
       EQUIVALENCE (A11,A(1)),(A21,A(2)),(A12,A(3)),(A22,A(4)),(EX,EXX),
      1     (EX(5),EYX),(EY,EYY),(EY(5),EXY),(PHI,Y2)
       DATA AX/-1., 1., 1.,-1., 0.,/,AY/-1.,-1., 1., 1., 0.,/,
      1     DC/-1., 1., 1.,-1.,-1.,-1., 1., 1./, IM/2,1,4,3/, IN/4,3,2,1/
C
C         INITIALIZATION
C
       DO 100 I=1,8
       DO 100 J=1,4
   100 TR(J,I)=0.0
       DO 110 I=1,NLC
       DO 110 J=1,15
   110 ST(I,J)=0.0
C
C         GEOMETRICAL TRANSFORMATION OF QUADRILATERAL BOUNDARIES
C
       X1(1)=0.5*(X(2)-X(1))
       X1(3)=0.5*(X(3)-X(4))
       X1(2)=X1(1)
       X1(4)=X1(3)
       Y1(1)=0.5*(Y(2)-Y(1))
       Y1(3)=0.5*(Y(3)-Y(4))
       Y1(2)=Y1(1)
       Y1(4)=Y1(3)
       X2(1)=0.5*(X(4)-X(1))
       X2(2)=0.5*(X(3)-X(2))
       X2(3)=X2(2)
       X2(4)=X2(1)
       Y2(1)=0.5*(Y(4)-Y(1))
       Y2(2)=0.5*(Y(3)-Y(2))
       Y2(3)=Y2(2)
       Y2(4)=Y2(1)
       DO 120 I=1,4
       DET =X1(I)*Y2(I)-Y1(I)*X2(I)
       FCT=DET+X2(I)*Y1(I)
       FT3(I)= Y2(I)*(1.0-Y1(I)*X2(I)/FCT)
       FT6(I)= X1(I)*(1.0-Y1(I)*X2(I)/FCT)
   120 FT7(I)= 0.5/DET
C
C         MODIFICATION OF ROTATIONS
C
       DO 130 I=1,4
       IA=IM(I)
       IB=IN(I)
       TR(I  ,I  )=-FT7(I)*(DC(I,1)*X2(I) - DC(I,2)*X1(I))*0.5
       TR(I  ,I+4)=-FT7(I)*(DC(I,1)*Y2(I) - DC(I,2)*Y1(I))*0.5
       TR(I  ,IA )=+FT7(I)*DC(I,1)*X2(I)*0.5
       TR(I  ,IA+4)=+FT7(I)*DC(I,1)*Y2(I)*0.5
```

```
      TR(I   ,IB   )=-FT7(I)*DC(I,2)*X1(I)*0,5
  130 TR(I   ,IB+4)=-FT7(I)*DC(I,2)*Y1(I)*0,5
      DO 140 L=1,NLC
      DO 140 I=1,4
      RR=0,0
      DO 135 J=1,8
  135 RR=RR+TR(I,J)*DIS(L,J)
  140 DIS(L,I+8)=DIS(L,I+8)+RR
C
C        FORMATION OF LOCAL DERIVATIVES
C
      IA=0
      DO 350 II=1,5
      ETA(1)=AX(II)
      ETA(2)=AY(II)
      DO 150  I = 1,2
      J=3-I
      A(I,1) = 0,
      A(I,2) = 0,
      EI=ETA(I)
      EI2=EI*EI
      EI3=EI2*EI
      EJ=ETA(J)
      EJ2=EJ*EJ
      EJ3=EJ2*EJ
      DO 145 L=1,4
      XX=DC(L,J)
      YY=DC(L,I)
      C2=XX*EJ
      P(L,I)=0,25*YY*(1,0+C2)
      A(I,1) = A(I,1) + P(L,I)*X(L)
      A(I,2) = A(I,2) + P(L,I)*Y(L)
      Q(L,I)=(-1,+2,*YY*EI+3,*EI2)*(2,+XX*(3,*EJ-EJ3))*0,0625
  145 R(L,J)=0,1875*(YY*(EI2-1,)-EI+EI3)*XX*(1,-EJ2)
  150 CONTINUE
      DET=1,/(A11*A22-A12*A21)
C
C        FORMATION OF GLOBAL DERIVATIVES
C
      DO 200 I=1,4
      PSI(I,1)=FT3(I)*R(I,1)
      PHI(I,1)=FT3(I)*Q(I,2)
      PSI(I,2) = FT6(I)*Q(I,1)
  200 PHI(I,2) = FT6(I)*R(I,2)
      DO 220 I=1,4
      DO 210 J=1,2
      K=3-J
      EX(I,J) = A22*P(I,1) - A12*P(I,2)
  210 EY(I,J) =-A21*P(I,1) + A11*P(I,2)
      EXZ(I) = -A22*PSI(I,1) + A12*PHI(I,1)
      EYZ(I) = -A21*PSI(I,2) + A11*PHI(I,2)
  220 EZZ(I) =  A21*PSI(I,1) - A11*PHI(I,1) + A22*PSI(I,2) -A12*PHI(I,2)
C
C        COMPUTATION OF NODAL POINT STRESSES
C
      DO 300 L=1,NLC
      XX=0,0
      YY=0,0
      ZZ=0,0
      DO 250 I=1,4
```

```
      DX=DIS(L,I)
      VXX=EXX(I)*DX
      VXY=EXY(I)*DX
      DX=DIS(L,I+4)
      VYY=EYY(I)*DX
      VYX=EYX(I)*DX
      DX=DIS(L,I+8)
      VXZ=EXZ(I)*DX
      VYZ=EYZ(I)*DX
      VZZ=EZZ(I)*DX
      DX=VXX+VXZ
      DY=VXY+VYX+VZZ
      DZ=VYY+VYZ
      XX=XX+D1*DX+D5*DY+D4*DZ
      YY=YY+D4*DX+D6*DY+D2*DZ
  250 ZZ=ZZ+D5*DX+D3*DY+D6*DZ
      ST(L,IA+1)=XX*DET
      ST(L,IA+2)=YY*DET
  300 ST(L,IA+3)=ZZ*DET
  350 IA=IA+3
C
      RETURN
      END


      SUBROUTINE FPLATE (X,Y,CM,S,R1,R,CVT,CV,BM,NLC)
C
C*********************************************************************
C     THIS SUBROUTINE DETERMINES BENDING MOMENTS IN A QUADRILATERAL
C     PLATE BENDING ELEMENT FORMED OUT OF FOUR LCCT-11 TRIANGLES,
C
C                                - INPUT -
C     X(5),Y(5)- ELEMENT COORDINATES OF 4 CORNER POINTS AND MIDPOINT
C     CM(3,3)  - CONSTITUTIVE MATERIAL MATRIX
C     S(19,7)  - PORTION OF CONDENSED ELEMENT STIFFNESS FOR INTERIOR
C                DEGREES OF FREEDOM
C    R1(NLC,12)- DISPLACEMENT MATRIX FOR -NLC- DIFFERENT LOAD CASES,
C                ORDERED ROW-WISE AS   (W(I),XROT(I),YROT(I),I=I,J,K,L)
C     NLC      - NUMBER OF DIFFERENT LOAD VECTORS
C
C                                - OUTPUT -
C    BM(NLC,15)- BENDING MOMENTS AT THE CORNER POINTS AND ELEMENT CENTER
C                FOR -NLC- LOAD CASES, ORDERED ROW-WISE AS
C                (MX(I),MY(I),MXY(I),I=I,J,K,L,O)
C*********************************************************************
C
C     COMMON, DIMENSION, EQUIVALENCE, AND DATA STATEMENTS
C
      COMMON/TRIARG/B(3),A(3),CMT(3,3),DUMMY(153)
      DIMENSION IPERM(4),NC(3),FAC(3),X(5),Y(5),CM(3,3),S(19,7),
     *  R1(NLC,1),R(NLC,1),CVT(NLC,1),CV(NLC,1),BM(NLC,1)
      DATA IPERM /2,3,4,1/, FAC /.5,.5,.25/
C
C     INITIALIZATION
C
      NTR = 4
      NBF = 11
      DO 130 L=1,NLC
      DO 130 I=13,55
  130 R1(L,I)=0.0
```

```
C
C     BACKSUBSTITUTION FOR ELIMINATED DEGREES OF FREEDOM
C
      DO 140 L=13,19
      M=L-1
      DO 140 K=1,M
      C=S(K,L-12)
      DO 140 N=1,NLC
  140 R1(N,L)=R1(N,L)-C*R1(N,K)
      NC(3)=NBF-6
C
C     PREPARE CALL OF SUBROUTINE FLCCT FOR EACH TRIANGLE
C
      DO 300 N=1,NTR
      M = IPERM(N)
      NC(1) = N
      NC(2) = M
      L = NC(3)
      A(1) = X(L)-X(M)
      A(2) = X(N)-X(L)
      A(3) = X(M)-X(N)
      B(1) = Y(M)-Y(L)
      B(2) = Y(L)-Y(N)
      B(3) = Y(N)-Y(M)
      DO 190 L=1,NLC
      DO 180 K=1,3
      R(L,K  )= R1(L,3*N+K-3)
      R(L,K+3)= R1(L,3*M+K-3)
  180 R(L,K+6)= R1(L,K+12)
      R(L,10) = R1(L,M+15)
  190 R(L,11) =-R1(L,N+15)
      CALL FLCCT (NBF,R,CVT,NLC)
C
C     ACCUMULATE AND AVERAGE NODAL CURVATURES
C
      DO 260 J=1,3
      L=NC(J)*3-3
      C=FAC(J)
      M=J*3-3
      DO 250 K=1,NLC
      DO 250 I=1,3
  250 CV(K,L+I)=CV(K,L+I)+C*CVT(K,M+I)
  260 CONTINUE
  300 CONTINUE
C
C     CALCULATE BENDING MOMENTS
C
      DO 400 L=1,NLC
      J=1
      DO 350 K=1,5
      M=J
      DO 350 I=1,3
      BM(L,J)=CM(I,1)*CV(L,M)+CM(I,2)*CV(L,M+1)+CM(I,3)*CV(L,M+2)
  350 J=J+1
  400 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FLCCT  (NBF,R,CVT,NLC)
C
C****************************************************************************
C          THIS SUBROUTINE COMPUTES CURVATURES FOR A LCCT TRIANGULAR
C     PLATE BENDING ELEMENT WITH  'NBF' BENDING D.F. -RIGHT HAND SYSTEM
C****************************************************************************
C
      COMMON/TRIARG/B(3),A(3),CMT(3,3),DUMMY(153)
      DIMENSION U(3),HT(3),TX(3),TY(3),Q(3,6),IPERM(3),NKN(2,3),
     *  R(NLC,1),CVT(NLC,1)
      DATA IPERM/2,3,1/, NKN/2,5, 8,2, 5,8/
C
C        INITIALIZATION
C
      AREA = A(3)*B(2)-A(2)*B(3)
      DO 120  I = 1,3
      J = IPERM(I)
      X = A(I)**2+B(I)**2
      U(I) = -(A(I)*A(J)+B(I)*B(J))/X
      X = SQRT(X)
      TX(I) =  0.5*A(I)/X
      TY(I) = -0.5*B(I)/X
      HT(I) =  4.0*AREA/X
      A1 = A(I)/AREA
      B1 = B(I)/AREA
      A2 = A(J)/AREA
      B2 = B(J)/AREA
      Q(1,I)   = B1*B1
      Q(2,I)   = A1*A1
      Q(3,I)   = 2.*A1*B1
      Q(1,I+3) = 2.*B1*B2
      Q(2,I+3) = 2.*A1*A2
  120 Q(3,I+3) = 2.*(A1*B2+A2*B1)
C
C        RECOVER DISPLACEMENT DOF-S ELIMINATED BY KINEMATIC CONSTRAINTS
C
      M = 12 - NBF
      IF (M.LE.0)  GO TO 160
      DO 140 L=1,NLC
      DO 140 N=1,M
      K=13-N
      L1=NKN(1,N)
      L2=NKN(2,N)
  140 R(L,K)=(R(L,L1)+R(L,L2))*TX(K-9)+(R(L,L1+1)+R(L,L2+1))*TY(K-9)
C
C        DETERMINE CURVATURES AT THE NODES
C
  160 M=1
      DO 300 I=1,3
      J = IPERM(I)
      K = IPERM(J)
      II = 3*I
      JJ = 3*J
      KK = 3*K
      A2 = A(J)
      A3 = A(K)
      B2 = B(J)
      B3 = B(K)
      U2 = U(J)
      U3 = U(K)
```

```
      W2 = 1.-U2
      W3 = 1.-U3
      C21 = -(2.+W2)*B2-(2.+U3)*B3
      C22 = B2*W2-B3*U3
      C31 = -(2.+W2)*A2-(2.+U3)*A3
      C32 = A2*W2-A3*U3
      C51 = 4.*B3-B2+B3*W3
      C52 = B2-B3*W3
      C61 = 4.*A3-A2+A3*W3
      C62 = A2-A3*W3
      C81 = B3-4.*B2-B2*U2
      C82 = B2*U2-B3
      C91 = A3-4.*A2-A2*U2
      C92 = A2*U2-A3
      DO 200  N = 1,3
      Q11 = Q(N,I)
      Q22 = Q(N,J)
      Q33 = Q(N,K)
      Q12 = G(N,I+3)
      G23 = G(N,J+3)
      Q31 = Q(N,K+3)
      G1 = Q22-G33
      G2 = Q22-G23
      G3 = Q33-Q23
      G4 = G23+Q1
      G5 = Q23-Q1
      R1=-6.*Q11+3.*((U3-W2)*Q1+(U3+W2)*Q23)
      R2= 6.*Q22+3.*W3*Q4
      R3= 6.*Q33+3.*U2*Q5
      R4=C21*G1+C22*Q23+4.*(B2*G31-B3*G12)
      R5=C31*G1+C32*Q23+4.*(A2*G31-A3*G12)
      R6=C51*Q22+C52*Q3
      R7=C61*Q22+C62*Q3
      R8=C81*Q33+C82*Q2
      R9=C91*Q33+C92*Q2
      DO 180 L=1,NLC
  180 CVT(L,M)=R1*R(L,II-2)+R2*R(L,JJ-2)+R3*R(L,KK-2)+(R4*R(L,II-1)
     *  +R5*R(L,II)+R6*R(L,JJ-1)+R7*R(L,JJ)+R8*R(L,KK-1)+R9*R(L,KK)
     *  +HT(K)*Q4*R(L,K+9)+HT(J)*Q5*R(L,J+9))/2.
  200 M=M+1
  300 CONTINUE
C
      RETURN
      END
```