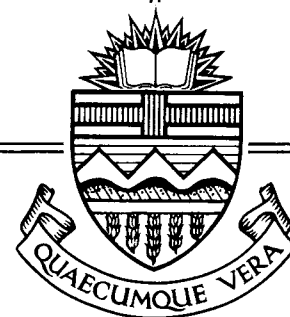


Structural Engineering Report No. 116



REINFORCED CONCRETE COLUMN DESIGN PROGRAM

by
CHI-KUN LEUNG
and
S. H. SIMMONDS

Department of Civil Engineering
The University of Alberta
Edmonton, Alberta, Canada

APRIL, 1984

REINFORCED CONCRETE COLUMN DESIGN PROGRAM

by

Chi-Kun Leung

and

S. H. Simmonds

Structural Engineering Report No. 116

Department of Civil Engineering

The University of Alberta

Edmonton, Alberta, Canada

April 1984

ABSTRACT

The development of a computer program, COLUMN1, for the design of reinforced concrete columns is presented. The columns may be either tied or spiral, long or short, braced or unbraced, and have either circular or rectangular cross-sections. Loading may be defined as either factored or unfactored with bending about one or both axes. For a given loading and material properties the column will select a satisfactory cross-section including longitudinal and lateral reinforcement. Provision is made for the designer to specify all or any portion of the cross-section, and to place limitations on bar size and bar numbers. All designs conform to either building standard CSA3-A23.3-M77 using S.I. units or ACI 318-77 using Imperial units. The program may be run in either batch or interactive mode with a hard copy output option if the latter mode is selected.

The design philosophy and engineering logic of the program are discussed and a number of examples to demonstrate most of the features of the program are included.

ACKNOWLEDGEMENT

This report describes a computer program for the design of reinforced concrete written by Mr. Chi-Kun Leung under the direction of Dr. S.H. Simmonds. The authors acknowledge the financial assistance for this study provided by the National Science and Engineering Research Council of Canada through operating grant NSERC A-1691.

TABLE OF CONTENTS

CHAPTER	PAGE
Abstract	i
Acknowledgement	ii
Table of Contents	iii
List of Tables	vi
List of Figures	vii
Notations	ix
1 INTRODUCTION	1
1.1 Column Design	1
1.2 Organization of Thesis	4
2 DESIGN PHILOSOPHY	5
2.1 Scope of Column Considered	5
2.2 Operational Modes	7
2.3 Control for Design	8
2.4 Output	9
3 ENGINEERING LOGIC	11
3.1 Introduction	11
3.2 Nomenclature and Sign Convention	11
3.3 Uniaxial and Biaxial Bending	13
3.4 Concrete Dimensions	15
3.4.1 General Consideration	15
3.4.2 Selection of Initial Concrete Section	16
3.4.3 Incrementing Concrete Section	17

CHAPTER	PAGE
3.5 Selection of Longitudinal Reinforcement	19
3.6 Evaluation of Slenderness Effects	25
3.6.1 Computation of Moment Magnifier	25
3.6.2 Evaluation of I_{se}	27
3.6.3 Minimum Moment	34
3.7 Evaluation of Strength Capacity	34
3.7.1 Adequacy Check in Strength	39
3.8 Modification of Trial Section	44
3.9 Evaluation of Longitudinal Splices	44
3.10 Selection of Lateral Reinforcement	48
4 ILLUSTRATIVE EXAMPLES	50
4.1 Example 1 - Short Tied Column (Braced)	50
4.2 Example 2 - Spiral Column (CSA)	61
4.3 Example 3 - Slender Braced Column (ACI)	65
4.4 Example 4 - Square Tied Column in Biaxial Bending	70
4.5 Example 5 - Rect. Tied Column Check	76
5 SUMMARY AND CONCLUSION	79
List of References	81
APPENDIX A USER'S MANUAL	85
A.1 Description	85
A.2 Control Commands	85
A.3 Data Input	86

CHAPTER	PAGE
APPENDIX B LOGIC FLOW OF COLUMN1	96
B.1 Flags Default	103
APPENDIX C COMPUTER CODING	104

LIST OF TABLES

TABLE	TITLE	PAGE
3.1	- Default values for maximum number of bars in section.....	20

LIST OF FIGURES

FIGURE	TITLE	PAGE
3.2.1	- Loading acting at the centroid of column section.....	12
3.2.2	- Bars distributed in x-faces only.....	12
3.2.3	- Bars distributed in y-faces only.....	14
3.2.4	- Bars distributed in both faces.....	14
3.5.1	- Bars in y-faces greater than bars in x-faces.....	22
3.5.2	- Bars in x-faces greater than bars in y-faces.....	22
3.6.1	- Bars in x-faces only w.r.t. y-axis.....	29
3.6.2	- Bars in x-faces only w.r.t. x-axis.....	29
3.6.3	- Bars in both faces w.r.t. y-axis.....	31
3.6.4	- Bars in both faces w.r.t. x-axis.....	31
3.6.5	- Bars in circular core w.r.t. both axes.....	33
3.7.1	- Linear strain distribution for bars in two opposite faces only and in both faces and in circular core.....	35
3.7.2	- Strength capacity at different location of neutral axis.....	37
3.7.3	- Strength-Capacity reduction factor.....	40
3.9.1	- Normal lap splices.....	45
3.9.2	- Tangential lap splices.....	45
4.1.1	- Batch input for example 1.....	54
4.1.2	- Batch output for example 1.....	55
4.1.3	- Interactive input for example 1.....	57
4.1.4	- Interactive output for example 1.....	58
4.1.5	- Column section for example 1.....	60

4.1.6 - Column section for example 1 specifying #30 bars.....	60
4.2.1 - Input for example 2.....	62
4.2.2 - Output for example 2.....	63
4.2.3 - Column section for example 2.....	64
4.3.1 - Input for example 3.....	67
4.3.2 - Output for example 3.....	68
4.3.3 - Column section for example 3.....	69
4.4.1 - Input for example 4.....	72
4.4.2 - Output for example 4.....	73
4.4.3 - Column section for example 4.....	75
4.5.1 - Input for example 5.....	77
4.5.2 - Output for example 5.....	78
B.1 - A general flow chart of COLUMN1.....	97

NOTATIONS

Roman letters, capital

Ast	total area of longitudinal reinforcement in a column cross section
Asx	total area of reinforcement distributed along x-faces in a column
Asy	total area of reinforcement distributed along y-faces in a column
Cm	moment gradient correction factor
Cx	size of rectangular column measured in x-direction for which moment about y-axis are being determined
Cy	size of rectangular column measured in y-direction for which moment about x-axis are being determined
Ds	distance from centroid of extreme compression reinforcement to centroid of extreme tension reinforcement in opposite half of member
Ec	modulus of elasticity of concrete
EI	flexural stiffness of a column
Es	modulus of elasticity of steel
Ig	moment of inertia of gross concrete section about the centroidal axis, neglecting the reinforcement
Ise	moment of inertia of reinforcement about the centroidal axis of the member cross-section
Mc	magnified factored moment to be used for design of compression member
Mn	nominal moment strength at section
Mu	factored moment at section
Mux	factored moment about x-axis which is primarily resisted by the reinforcement in y-faces

M_{uy} factored moment about y-axis which is primarily resisted by the reinforcement in x-faces

M₁ value of smaller factored end moment on compression member calculated by conventional elastic frame analysis, positive if member bent in single curvature, negative if bent in double curvature

M₂ value of larger factored end moment on compression member calculated by conventional elastic frame analysis, always positive

N_x number of bars in x-face only including corner bars

N_y number of bars in y-face only

P_c critical axial load

P_n nominal axial load strength at given eccentricity

P_u factored axial load strength at given eccentricity

S_b the actual clear bar spacing in x-faces or y-faces

S_{bx} the actual clear bar spacing in x-faces

S_{by} the actual clear bar spacing in y-faces

Roman letters, lower case

a depth of equivalent rectangular stress block

c distance from extreme compression fiber to neutral axis

d distance from extreme compression fiber to centroid of tension reinforcement

d_b nominal diameter of bars

d_c thickness of clear concrete cover

d_s lateral reinforcement diameter

d' distance from extreme compression fibre to the center of the bar located closest thereto

e eccentricity of design load parallel to axis
 measured from the centroid of the section
 f'_c specified compressive strength of concrete
 f_y specified yield strength of reinforcement
 h diameter of a round column or side of a
 rectangular column
 k effective length factor for compression members
 l_u unsupported length of compression member
x-face lateral faces or edges in y-direction
 with column dimension C_y
y-face end faces in x-direction with column
 dimension C_x

Greek letters

β_1 a coefficient relating the depth of the
 equivalent rectangular stress block to the
 depth from the compression face to the
 neutral axis
 β_d ratio of maximum factored dead load
 moment to maximum factored total load
 moment, always positive between zero
 and one
 ϵ_{cu} maximum usable strain at the extreme
 concrete compression fiber assumed
 equal to 0.003
 δ moment magnification factor for columns
 ρ_s maximum permissible ratio of longitudinal
 reinforcement in a column
 ρ_t total area of reinforcement divided by
 total area of cross-section in a column
 ϕ capacity reduction factor
 θ angle between longitudinal bars distributed
 along circular core

Subscripts

x about x-axis
y about y-axis

1. INTRODUCTION

1.1 Column Design

The design of a reinforced concrete column is essentially one of trial and error. Design consists of finding a cross-section that will support satisfactorily both an axial load and a bending moment. However, since the capacity of a section to carry axial load is dependent on the magnitude of the moment that is acting (in some cases the moment is a function of the section dimensions), there are no closed form solutions for determining a section uniquely. The problem is compounded when moments act about both axes of the section.

Traditionally, to reduce the required number of trials to obtain a satisfactory design, use was made of non-dimensional 'interaction diagrams'. These are plots of axial load vs. moment capacity relationships for specific material properties and section geometry. Such plots are practical for bending about one axis only, so that, in practice, it is common to consider bending only about one axis and ignore the moment in the other direction, a practice that is sometimes unrealistic and unsafe. Thus, there is a need for a computer program that will select an economical and adequate section for reinforced concrete columns with bending about both axes.

Reinforced concrete columns are classified as either tied or spiral. Both columns have longitudinal and lateral

reinforcement. With a tied column the lateral reinforcement prevents premature buckling of the longitudinal reinforcement but does not confine the concrete. With a spiral column the longitudinal reinforcement and concrete is enclosed in a spiral of reinforcement of sufficient strength and spacing that the concrete cannot fail before a significant triaxial stress condition is established in the concrete. The difference in the two types is the mode of failure. Tied columns will fail shortly after reaching maximum axial load with little ductility whereas spiral columns will demonstrate great ductility with little or no loss of load carrying capacity. The different behavior modes can be predicted and to take advantage of the enhanced characteristics of spiral columns it is necessary to meet a number of requirements that have been determined by experiment. These are specified in building codes with the result that a computer program for column design in practice must conform to many limitations imposed by the building codes.

Reinforced concrete columns must also be constructed manually. This imposes certain restrictions on dimension tolerances, number and arrangement of reinforcement bars, bar diameters, etc. For use in a design office, a computer program must permit the designer to impose on the design a number of restraints that are user dependent, that is they can vary from run to run depending on the requirements of a particular construction project.

A number of programs have been developed for design of reinforced concrete columns, such as RC.Column by Grant T. Halvorsen (15), and the column program, COLS, by Portland Cement Association (7). These programs are essentially 'checking' programs, that is the user enters concrete dimensions and reinforcement size and pattern and the program checks to see if the section will satisfy a given loading. If input section is not satisfactory, only minor alterations to the section will be made for further trials. If after these minor alterations the section is not satisfactory, the user must begin again by entering a new section which hopefully will be closer to a satisfactory section. While such programs are useful in the hands of those designers who from past experience, can predict closely the size of section that should be entered, they are of little use to the inexperienced designer. The purpose of this project is to develop a program that can be used by students to obtain quickly designs for reinforced concrete columns so that they can obtain such experience. For this reason the program must be capable of selecting a column section, including reinforcement, for any given loading, column length and bracing that conforms to specified building code requirements and any user imposed limitations, when only the loading is given. Such a program is, of course, even more useful in the hands of an experienced designer and can be incorporated into a integrated CAD system.

1.2 Organization of Thesis

The underlying philosophy as to the requirements, scope and limitations of the program are given in Chapter 2. Chapter 3 contains a discussion of the engineering decisions that are required in order to obtain complete designs. Design examples are discussed in Chapter 4. Chapter 5 contains a brief summary and conclusions. A Users' Manual for the program and a listing are given in the Appendices.

2. DESIGN PHILOSOPHY

2.1 Scope of Columns Considered

The purpose of the program, *COLUMN1* , is to provide engineers with a tool that will design most columns that will be encountered in practice in accordance with either the CSA3-A23.3-M77 (8) or the ACI318-77 (1,2) building codes. For this reason, both long and short columns subject to either uni-axial or bi-axial loading are considered but cross-sections are restricted to either rectangular or circular shapes. Furthermore, columns may be braced or unbraced. For long columns the moment magnifier method is used. For long braced columns the moment magnifier is computed from the input geometry and loading. However, for long unbraced columns the sums for the axial load and critical load for all columns in that floor are required and these values must be entered manually.

In keeping with the generality of solution but to limit the possible longitudinal bar arrangements, the following restrictions on reinforcement geometry are imposed. For circular columns the reinforcement consist of an even number of bars which must be located around the perimeter in one layer for both tied and spiral columns. The program checks for minimum number of bars and adequate bar spacing to satisfy building code requirements. Square column sections but not rectangular sections may be used for spiral columns in which case the reinforcement must be placed in a circular

pattern. For tied rectangular columns the reinforcing is restricted to being parallel to the faces, but may be located on two opposite faces only or along all faces providing that double symmetry of the cross-section is maintained. Individual bars may be specified so that the usual restriction placed on interaction diagrams in most handbooks which require reinforcement along all faces to have equal areas in each face is not required.

In keeping with the concept of designer control, the user has, for each problem, the option of specifying either 'design' or 'check'. For a check problem, the user enters both the concrete dimensions and the complete reinforcement pattern and the program determines whether the section entered is adequate for the specified loadings. For a design problem, the user specifies only as little or as much of the cross-section details as he wishes and the computer completes the design of the section to safely carry the loading.

Prior to checking a given cross-section for adequacy to resist the loading, the program checks the reinforcement for limitation of percentage of reinforcement, bar size and bar spacing. If the longitudinal reinforcement was entered by the user and was found to violate any of the imposed limitations, a message indicating the violation is printed and execution of the program terminates. Once a satisfactory section is determined the reinforcement pattern is checked to determine whether a tangential lap splice is possible and

if so an appropriate message is printed.

For tied columns the program will select the tie diameter and required tie spacing. The actual choice of tie pattern is generally standard and is left to the designer. For spiral columns the program selects the spiral diameter and pitch and the minimum number of vertical spacers to satisfy codes requirements.

2.2 Operational Modes

A single column design or multiple column designs can be solved during a single run. The user has the option of running the program in either batch or interactive mode, however, the mode selected applies to all designs that are to be solved during that run.

In batch mode operation, the user prepares a data input file, and obtains output after each run. Unlike the interactive mode, the user cannot interact with the design solution during the run.

In interactive mode, the user enters the required input line by line after being prompted by the computer. The output is displayed on the terminal screen. This output includes a summary of the column design. At this point, the user has the option of either revising the input and rerunning the design or proceeding to the next design.

In interactive mode all output during the run appears on the terminal screen. At the end of each design, the final solution is stored in a file from which a hard copy output

may be listed, if requested, at the end of the run.

2.3 Control for Design

The first line of input for each design is a user defined heading by which the design and output are identified.

When the appropriate building code is entered, the corresponding units and bar designations are assigned. In addition all data for which the code gives specific values are assigned default values. Some of these parameters established by default, such as concrete cover, maximum percentage of reinforcement, etc. can be overridden by entering desired values. Other parameters, such as strength reduction factors for tied and spiral columns or bar designations, can only be altered by editing the program and, therefore cannot be done during a program run. The entry of data that is not code dependent, such as the magnitude of loading, column length, reinforcement pattern and material strength properties cannot be assigned default values and must be entered.

The final line of input for each design is a coded number that indicates whether another design follows or whether the run is to be terminated. For batch mode, when another design follows the coded number indicates at which stage following reading of the new heading new data is to be read since input not read is taken to be the same as for the previous design. In interactive mode, the user has a number

of options depending on whether he wants to revise the design or proceed to the next design.

For each column design a number of different loading conditions may be considered. With initial loading case the user indicates whether it is a 'design' loading or a 'check' loading. However for multiple loading cases, subsequent loadings have to be a check at which case the program checks the cross-section which was either entered or designed for the initial loading for adequacy. In batch mode, the total number of load cases must be entered. In interactive mode, the total number of load cases is not required as the user has an option to revise the loading conditions as he wishes.

For detail input procedure and data assigned by default, the reader is referred to the USER'S MANUAL in Appendix A.

2.4 Output

The user has an option of either obtaining a short output or a long output. The short output provides sufficient information to evaluate the design and to construct the column and would be the usual output in design practice. The long output provides additional detail information, including an echo check of default values for all input data and results of certain intermediate calculations that permit monitoring of the design for debugging purposes or for evaluating a peculiar condition.

The output for each design consists of the heading entered by the user to identify the design, the date and time problem was run to distinguish between multiple runs of the same design if the heading was not altered. The heading is also printed at the top of all subsequent pages for the same design.

The short output also contains a summary of the control parameters, such as building code, column type and the loading conditions considered. When the loading case is a design loading condition, the output will include the loading and column section properties. When the loading case is a check loading, the output will consist of the loading and a message indicating whether or not the section was adequate. The exception to this rule is that when there is a single loading case and it is a check loading, then both the input loading and section properties are printed before the message of adequacy.

3. ENGINEERING LOGIC

3.1 Introduction

This chapter discusses the engineering decisions used in the column design program. In summary, the various steps are as follows:

- (a) enter or select initial concrete section that is minimum consistent with loading,
- (b) enter or select longitudinal reinforcement that is minimum consistent with reinforcing restraints and geometry,
- (c) consider slenderness effects such as magnification factor, and minimum moment to determine design loading,
- (d) compute axial load and moment capacity of trial section, and compare these values with applied axial load and required moment,
- (e) modify trial section if required,
- (f) evaluate longitudinal splices and select lateral reinforcement.

These steps are described in detail in the following sections.

3.2 Nomenclature and Sign Convention

The positive directions of the axial load and bending moments acting at the centroid of the column section are shown in Fig. 3.2.1. Reinforcement may be placed along only the x-face or the y-face, or both as long as double symmetry with respect to axes through the centroid is maintained. The

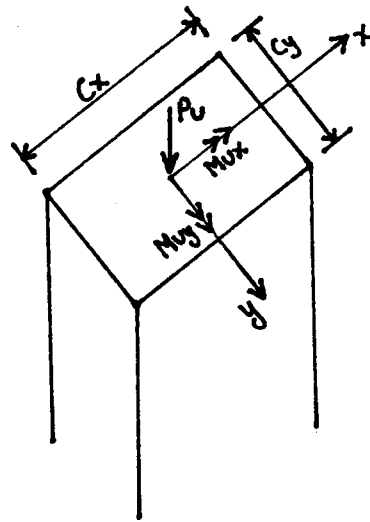


Fig. 3.2.1 Loading acting at the centroid of column section

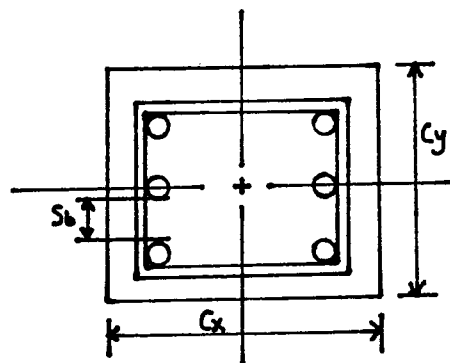


Fig. 3.2.2 Bars distributed in x-faces only

bars placed in x and y -faces are perpendicular to the respective axes as shown in Fig. 3.2.2 and 3.2.3. When bars are placed in all faces, the corner bars are considered to be located along the x-faces as shown in Fig. 3.2.4.

3.3 Uniaxial and Biaxial Bending

In uniaxial bending, user must indicate whether the compression member is braced or unbraced about the x-axis and enter the effective length factor and the unsupported length about the x-axis. The moment about the x-axis, M_{ux} , acting with P_u , is primarily resisted by the reinforcement in the y-faces.

In biaxial bending, in addition to the above data input, the user must indicate whether the compression member is braced or unbraced about the y-axis and enter the effective length factor and the unsupported length about the y-axis. Normally, the unsupported length about the x-axis is the same as that about the y-axis. The moment about the y-axis, M_{uy} , acting with P_u , is primarily resisted by the reinforcement in the x-faces.

When the entered moment value is less than the minimum moment specified in the building codes, the specified minimum moment is used.

The program determines whether column deflects in single or double curvature by examining the signs assigned to the moments at the top and bottom of the column. A negative sign to one of the entered moments indicates the

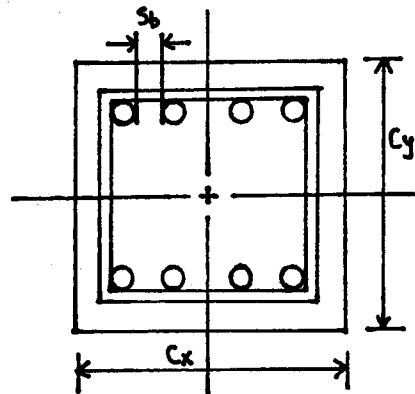


Fig. 3.2.3 Bars distributed in y-faces only

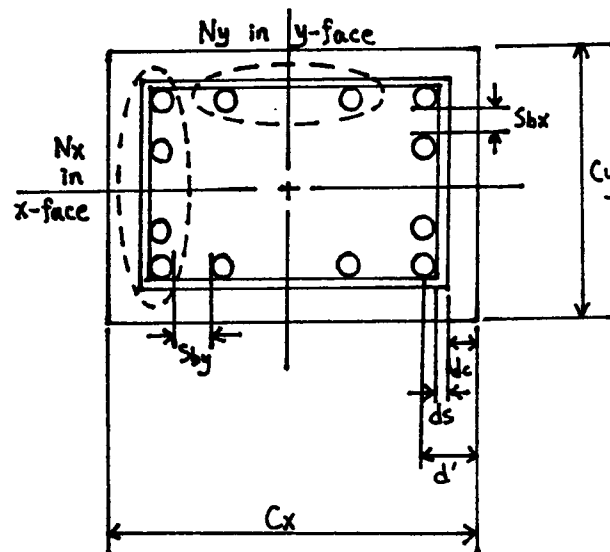


Fig. 3.2.4 Bars distributed in both faces

column deflects in double curvature. When the signs are the same, the column deflects in single curvature. The design moment is the maximum absolute value of the moment entered at either top or bottom of the column.

3.4 Concrete Dimensions

3.4.1 General Consideration

Dimensions of the column cross-section may be either user entered or determined by the program. If determined by the program each column dimension is selected to be an integer multiple of a predetermined length increment which has a default value of 2 in. for Imperial units and 50 mm for SI units or it may be user entered. To maintain a minimum initial trial so that all subsequent trials will be larger, all computed dimensions are rounded down to the next smaller value. For example, a computed value between 16.1 in. and 17.9 in. using the 2 in. default value would be selected as 16 in. Similarly, a value between 201 mm and 249 mm would be selected as 200 mm. If, however, the column cross section dimensions are entered either as specific values or, for a rectangular column as a single dimension and a column aspect ratio, the program will not alter these values, even if they are not an integer multiple of the length increment.

A minimum concrete column dimension is built into the program which is considered to be the minimum dimension that

can be physically built with the type of materials envisioned by the code. This minimum is 8 in. or 200 mm depending on units used and is the minimum dimension that will be considered in the design mode. If a column dimension less than this minimum is entered, an error message to this effect is printed and the program will go to the next problem..

3.4.2 Selecting Initial Concrete Section

When the column dimensions are not entered, an initial value of the column area, A_g^0 is computed from Eqn. 3.4.1.

$$A_g^0 = P_u / \phi \alpha_c [0.85f'_c + \rho_s (f_y - 0.85f'_c)] \quad (3.4.1)$$

where α_c is the reduction factor to account for the reduced capacity caused by the bending moments. The values of α_c used are 0.8 for tied and 0.85 for spiral columns which are values corresponding approximately for minimum moment for uniaxial bending. The value of ρ_s is the maximum reinforcement ratio to be used in this design which may be either user entered or defaults to a value of 0.03. This will result the smallest possible concrete cross-section consistent with the restraints for the initial column dimensions.

If a circular and square column is specified, the initial column dimensions are obtained directly using the values of A_g^0 . If a rectangular column is specified by a single column dimension or by the column aspect ratio, the

unknown values can also be calculated from Ag^0 only. For a rectangular column when uni-axial bending is specified, and no dimension or aspect ratio is entered, the program computes parameter Cy from Eqn. 3.4.2 which assumes minimum eccentricity based on the input loading.

$$Cy = Mux / (0.1Pu) \quad (3.4.2)$$

Cx can then be calculated from Ag^0 only. For a rectangular column when bending about both axes is specified, a column aspect ratio R is determined from Eqn. 3.4.3 which in conjunction with Ag^0 is used to obtain the initial values of Cx and Cy .

$$R = Cx / Cy = Muy / Mux \quad (3.4.3)$$

If no moments are entered, R defaults to 1.0. This same aspect ratio is maintained for all future increments of dimension, if required.

3.4.3 Incrementing Concrete Section

Using the initial concrete section, the program attempts to select longitudinal reinforcement consistent with the reinforcing constraints. If no such reinforcement can be found, the column dimensions, if not entered, are incremented in accordance with the following rules and the process of selecting reinforcement repeated. Consistent with the input philosophy, when a concrete dimension is entered it will not be altered by the program. When a dimension is

altered the increment will be the single step of the current value of the length increment.

When no column dimensions are entered but the shape is specified as square, both column dimensions are incremented equally by the length increment. If no column dimensions or aspect ratio are entered and the shape entered is rectangular, the smaller cross-section dimension is increased in increments until a square section is obtained. If a still larger section is required, the increments alterate between the directions first Cy then Cx attempting to maintain as close to a square section as possible. When a column aspect ratio is entered, a similar procedure exists except that the dimensions incremented are such as to most closely approximate the entered aspect ratio.

After a possible section of concrete with steel is selected, the slenderness ratio must be checked to determine whether the column is short or long. For uniaxial bending, when the column is long and the critical axial load is less than the nominal axial load, a larger column area is required. This is obtained by computing a new value of A_g as described in Section 3.6 and new column dimensions consistent with the rules for selecting initial trial section as outlined in Section 3.4.2 are obtained.

The trial section is then checked for adequacy in strength. When the section is not adequate, the longitudinal reinforcement is incremented first as described in Section 3.8. Should a suitable reinforcement pattern not be found

the concrete dimensions will be increased in accordance with the rules given in the second paragraph of this section.

3.5 Selection of Longitudinal Reinforcement

All reinforcing patterns whether entered or determined by the program must be symmetrical about the centroidal co-ordinate axes. For tied columns, however, bars may be placed along all sides or only on opposite sides if cross-section is square or rectangular. For spiral columns, the bars must be in a circular pattern regardless of whether column cross-section is circular or square.

When using the program in check mode, the reinforcement pattern, the bar size, and the total number of bars must be entered. Normally, the bar size is entered by assigning equal values to the minimum and maximum bar size, but if they are not equal, the program will use only the minimum bar size.

When using the program in design mode, only the reinforcement pattern must be entered. Minimum and maximum bar size may be either user specified, or defaults to #5 and #11 in *ACI* and #15 and #35 in *CSA*. The maximum reinforcement ratio, which controls the maximum area of steel that can be used before the concrete section is incremented, is either entered or defaults to a value of 0.03.

The program computes the initial steel area by multiplying the trial concrete dimensions by the minimum percentage of steel permitted by the controlling building

code. This will ensure that the most economical section is tried. When the specified number of bars is not entered, the program then computes the number of bars having a size equal to the minimum bar size and rounds this number upwards to the next even integer. This number is then compared to the maximum number of bars for that section which may be either user entered or defaults to the values given in Table 3.1. When the required total number of bars is greater than the maximum number, the bar size will be increased providing it has not reached the maximum bar size limit.

Dimension		Maximum number of bars
in.	mm	
≤16	≤400	8
18 - 22	450 - 550	12
24 - 30	600 - 750	16
32 - 36	800 - 900	20
etc.		

Table 3.1 Default values for maximum number of bars in section

The values given in Table 3.1 are considered to be representative of those used in practice and are calculated by the program when the column dimension exceeds 400 mm or 16 in. by the following expressions :

for *CSA*

$$\text{NO. OF BARS} = (C'/200 + 1)4$$

and for *ACI*

$$\text{NO. OF BARS} = (C'/8 + 1)4$$

where C' is integer equivalent of the column dimension along which the reinforcement is located, but for reinforcement pattern in both faces, the short dimension governs.

For the case when the user wishes to input the number of bars, and bars are in both faces, the number of bars in the x-face and y-face must be specified separately. The number of bars in the x-face includes the corner bars as shown in Fig. 3.2.4. When the specified number of bars is not entered, the program consistent with the pattern input computes the number of bars in the x and y -faces by considering the following conditions. If the bending moment about the x-axis is greater than the moment about the y-axis, the number of bars in the y-faces will be either greater than or equal to the number of bars in the x-faces not including the corner bars. This is accomplished by the following expressions (Fig. 3.5.1):

for bars in x-faces,

$$N_x = (N_{\max} + 4.) / 4.$$

for bars in y-faces,

$$N_y = (N_{\max} - 4.) / 4. + 0.51$$

where $N_{\max}$ is a real number equivalent of the required total

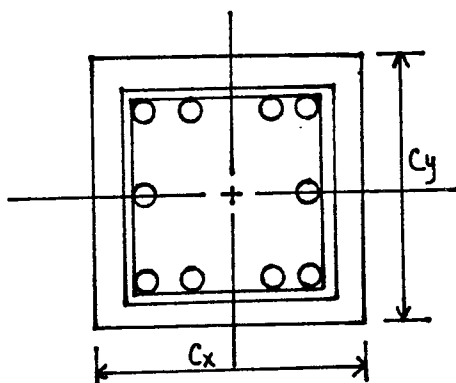


Fig. 3.5.1 Bars in y-faces greater than bars in x-faces

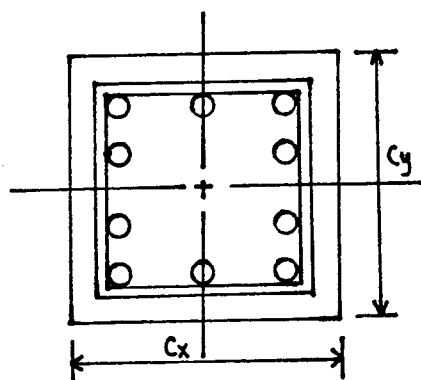


Fig. 3.5.2 Bars in x-faces greater than bars in y-faces

number of bars rounded upwards to the next even whole number, and N_x and N_y are integer equivalents of the values obtained.

Similarly, if moment about the x-axis is less than the moment about the y-axis, bars in the y-faces will be either less than or equal to bars in the x-faces not including the corner bars. The corresponding expressions are (Fig. 3.5.2): for bars in x-faces,

$$N_x = (N_{max} + 4.) / 4. + 0.51$$

for bars in y-faces,

$$N_y = (N_{max} - 4.) / 4.$$

The number of bars is also compared to the minimum number of bars permitted for that cross-section to satisfy building code requirements. When in check mode and the number of bars entered is less than the minimum number, an error message is printed and the program proceeds to the next problem. When in design mode and the number of bars either computed or entered is less than the minimum number, the number of bars is increased to this minimum value.

The required minimum clear bar spacing is determined by taking the larger value of either 1.5 times the bar diameter or a predetermined minimum clear bar spacing which has a default value of 1.5 in. for Imperial units and 40 mm for SI units. In calculating the actual bar spacing, the default values of the minimum diameter of lateral reinforcement and

the minimum value of concrete cover specified by the building codes are used unless other values have been entered or the codes control the size of lateral reinforcement diameter consistent with the reinforcing constraints. If the bar spacing is less than the specified minimum spacing, a new design using the next larger bar size will be attempted providing the bar size has not reached the maximum bar size limit or that the number of bars to be used has not been entered, otherwise the computed concrete section will be increased to satisfy the reinforcement constraints. However should the concrete dimensions be user entered an error message indicating that the design cannot be completed with the entered constraints is printed and the program proceeds to next problem.

After a trial column cross-section including longitudinal reinforcement has been found with sufficient capacity to resist the loading, the value of the reinforcement ratio is computed. When this value exceeds the maximum reinforcement ratio applicable to the design, the concrete dimensions, if not user entered, are increased and a new cross-section is obtained. This process is repeated until a cross-section meeting all constraints is determined. Should this not be possible a message indicating the constraints that cannot be satisfied is printed and the program proceeds to the next problem.

3.6 Evaluation of Slenderness Effects

3.6.1 Computation of Moment Magnifier

Before checking the capacities of the trial cross-section, it must be determined whether the column is long, that is the effects of slenderness must be considered. This decision is made in accordance with the building code standards.

For members braced against side-sway, the slenderness effects may be neglected when the slenderness ratio, $k\ell u/r$, is less than $34-12(M_1/M_2)$, in which k is the input value of effective length factor, ℓu is the unsupported length of column, r is the radius of gyration of the cross-section of the column, M_1/M_2 is the ratio of moments at the ends of the column. For members not braced against side-sway, the slenderness effects may be neglected when the slenderness ratio is less than 22. For long columns the slenderness effect is considered using moment magnifier method to increase the applied moments with a magnification factor.

For long braced columns the moment magnifier, δ , is computed from the input loading and geometry.

$$\delta = C_m / [1 - (P_u / \phi P_c)] \geq 1.0 \quad (3.6.1)$$

where $P_c = \pi^2 EI / (k\ell u)^2 \quad (3.6.2)$

$$C_m = 0.6 + 0.4(M_1/M_2) \geq 0.4 \quad (3.6.3)$$

If the equivalent moment factor, C_m , is less than 0.4, it is set to 0.4.

The critical axial load, P_c , is a function of the trial section and involves the flexural stiffness, EI . This is initially computed as

$$EI = (0.2E_c I_g + E_s I_{se}) / (1 + \beta_d) \quad (3.6.4)$$

in which E_c is modulus of elasticity of concrete, I_g is moment of inertia of gross concrete section about the centroidal axis, E_s is modulus of elasticity of steel, β_d is ratio of maximum factored dead load moment to maximum factored total load moment, I_{se} is moment of inertia of reinforcement about the centroidal axis of the member cross-section as described in Section 3.6.2. For uniaxial bending about the x-axis, the critical axial load is compared with the applied axial load, P_u . When the critical axial load is greater than the nominal axial load, compute the moment magnifier. If the moment magnifier is less than 1.0, it is set to 1.0. However when the critical axial load is less than the nominal axial load, a larger column area is required. This is obtained by using a conservative value of EI from Eqn. 3.6.5 and computing a new value of I_g from Eqn. 3.6.6.

$$EI = 0.4E_c I_g / [1 + \beta_d] \quad (3.6.5)$$

$$I_g = \frac{P_u [1 + \beta_d] [k\ell_u / \pi]^2}{\phi 0.4E_c} \quad (3.6.6)$$

in which the value of P_u / ϕ is used instead of P_c . When a new value of column area, A_g as a function of I_g , is obtained, new column dimensions consistent with the rules for selecting initial trial section as outlined in Section 3.4.2

are obtained. The process of selecting reinforcement and checking slenderness effects is repeated until the critical axial load is greater than the nominal axial load.

For long unbraced columns the sum for the axial load, ΣP_u , and critical load, ΣP_c , for all columns in that floor are required and these values must be entered manually. However when these values are not entered, a message indicating that insufficient information to compute moment magnifier is printed and the program proceeds to the next problem. The moment magnifier for long unbraced column is computed as follows :

$$\delta = C_m / [1 - (\Sigma P_u / \phi \Sigma P_c)]$$

in which $C_m = 1.0$

If the term $\phi \Sigma P_c$ is less than ΣP_u , an error message to this effect is printed and the program proceeds to the next problem.

3.6.2 Evaluation of I_{se}

The evaluation of the moment of inertia of reinforcement I_{se} for long braced columns with bars distributed in four different patterns is described in the following parts.

(a) Considering moment of inertia of reinforcement is taken about the centroidal axis of the member cross section, and using direct application of statics, the general formula for moment of inertia of reinforcement is:

$$I_{se} = \sum_i [A_{si} X_{si}^2]$$

in which A_{si} is the total area of bars at i line, X_{si} is the distance from the centroid of bars at i line to the centroidal axis of the section.

For bars in the x-face only, moment of inertia of reinforcement about y-axis (Fig. 3.6.1):

$$I_{sey} = 0.25 A_{st} \gamma_x^2 C_x^2$$

where

$$\gamma_x = [C_x - 2d_c - 2d_s - d_b] / C_x$$

Also, moment of inertia of reinforcement about x-axis (Fig. 3.6.2):

$$I_{sex} = [4 A_{st} \sum y(i)^2] / N_{max}$$

where

$$y(i) = 0.5 \gamma_y C_y - (i-1)(d_b + S_b)$$

$$\gamma_y = [C_y - 2d_c - 2d_s - d_b] / C_y$$

$$i = 1, L, \quad L = (N_{max} - 1) / 4 + 0.51$$

in which $y(i)$ is the distance from the centroid of bars at i line to the x-axis, i is the line where bars are located starting from the extreme bars, the term L here is rounded upwards to the nearer next integer.

(b) Similarly, for bars in the y-face only, moment of inertia of reinforcement about y-axis:

$$I_{sey} = [4 A_{st} \sum x(i)^2] / N_{max}$$

where

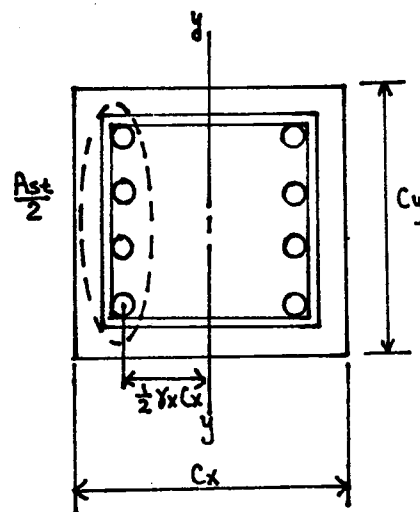


Fig. 3.6.1 Bars in x-faces only w.r.t. y-axis

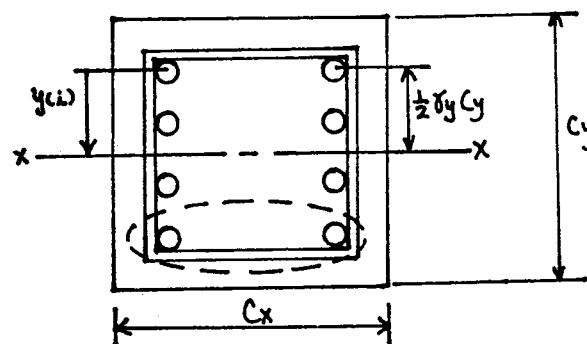


Fig. 3.6.2 Bars in x-faces only w.r.t. x-axis

$$x(i) = 0.5\gamma_x C_x - (i-1)(db + Sb)$$

$$\gamma_x = [C_x - 2dc - 2ds - db] / C_x$$

$$i = 1, L, \quad L = (N_{max} - 1.) / 4. + 0.51$$

in which $X(i)$ is the distance from the centroid of bars at i line to the y -axis, the terms i and L here are similar to those defined in part (a).

Also, moment of inertia of reinforcement about x -axis:

$$I_{sex} = 0.25 A_{st} \gamma_y^2 C_y^2$$

where

$$\gamma_y = [C_y - 2dc - 2ds - db] / C_y$$

(c) For bars in both faces, moment of inertia of reinforcement about y -axis (Fig. 3.6.3):

when no bars in y -face,

$$I_{sey} = 0.25 A_{sx} \gamma_x^2 C_x^2$$

otherwise,

$$I_{sey} = 0.25 A_{sx} \gamma_x^2 C_x^2 + (2A_{sy} / N_y) \sum x(i)^2$$

where

$$x(i) = 0.5\gamma_x C_x - db - S_{by} - (i-1)(S_{by} + db)$$

$$i = 1, L, \quad L = (2.N_y - 1.) / 4. + 0.51$$

in which i is the line where bars are located starting from the extreme bars in y -faces, the term L here is defaults to 1 when bars are less than two in each y -face.

Also, moment of inertia of reinforcement about x -axis (Fig. 3.6.4):

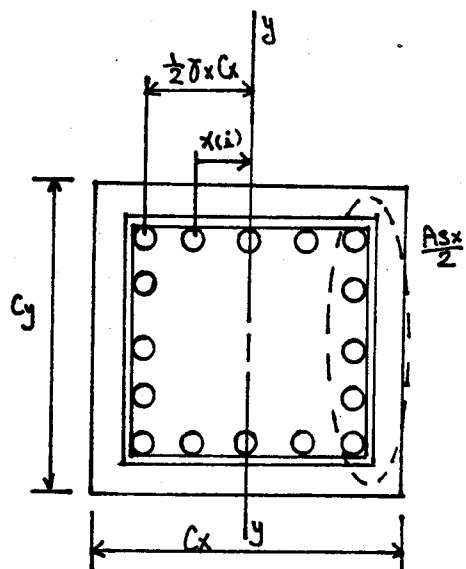


Fig. 3.6.3 Bars in both faces w.r.t. y -axis

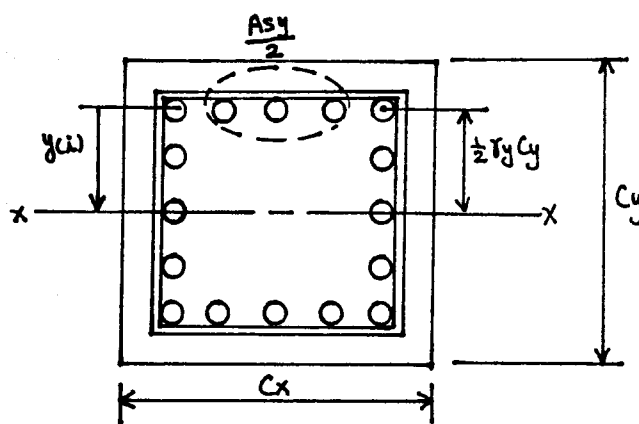


Fig. 3.6.4 Bars in both faces w.r.t. x -axis

$$I_{sex} = 0.25 A_s y y^2 C y^2 + (2 A_s x / N_x) \sum y(i)^2$$

where

$$y(i) = 0.5 y C y - (i-1)(S_b x + d_b)$$

$$i = 1, L, \quad L = [2.N_x - 1.] / 4. + 0.51$$

in which the terms i and L here are similar to those defined in part (a).

(d) For bars in round face, moment of inertia of reinforcement about y -axis (Fig. 3.6.5):

$$I_{sey} = [A_s t (0.5 D_s^2 + 4 \sum x(i)^2)] / N_{max}$$

where

$$x(i) = (0.5 D_s) \cos(i\theta)$$

$$i = 1, L, \quad L = (N_{max} - 3.) / 4. + 0.51$$

in which i is the line, where bars are located not in diameter centroidal axis, starting from the extreme bars, the term L here is similar to that defined in part (a).

Also, moment of inertia of reinforcement about x -axis:

$$I_{sex} = [A_s t (0.5 M D_s^2 + 4 \sum y(i)^2)] / N_{max}$$

where

$$y(i) = (0.5 D_s) \sin(i\theta)$$

$$M = (N_{max} - L) / N_{max}$$

$$i = 1, L, \quad L = (N_{max} - 4.) / 4. + 0.51$$

in which the term M is defaults to 1 when bars are located in the centroidal y -axis of the member cross section, otherwise, M is defaults to 0.

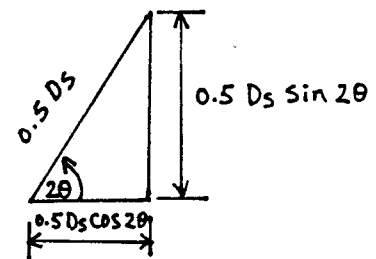
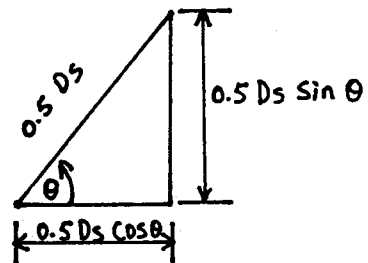
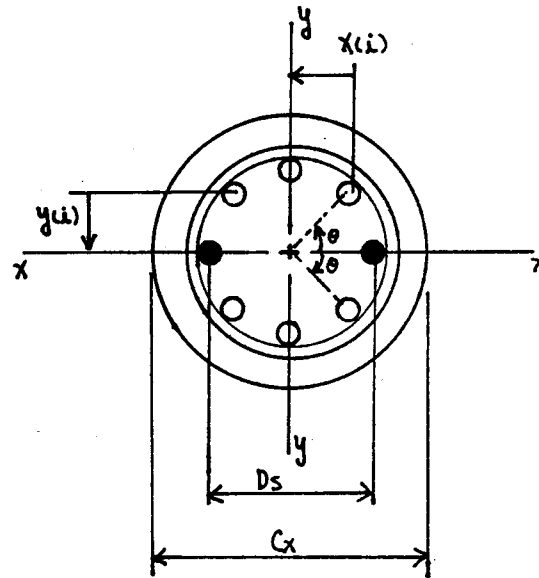


Fig. 3.6.5 Bars in circular core w.r.t. both axes

Note : The center bars in centroidal x -axis are used as relative reference line for measuring θ

3.6.3 Minimum Moment

Determine whether design of compression members is governed by either the minimum eccentricities specified in the building codes or the actual computed eccentricity. When the actual computed eccentricity is less than the specified minimum, the design moment is based on the specified minimum eccentricity rather than the entered moment.

3.7 Evaluation of Strength Capacity

When the strength capacity is computed, the trial section and loading have been modified for slenderness and minimum moment effects. For uniaxial bending, assume strains in steel and concrete are linearly distributed and the maximum usable compression strain in the extreme fibre of the concrete is limited to 0.003. When steel strains are greater than that corresponding to f_y , the stress in the reinforcement is taken equal to f_y (Fig. 3.7.1). Stresses in the concrete are assumed to follow the equivalent rectangular stress block for concrete in flexure.

To determine the section capacity it is necessary to find the location of the neutral axis. This is accomplished by considering the location of neutral axis as the unknown value and expressing the internal forces as a function of this value.

Theoretically it is possible to sum the moments of these internal forces about the point of application for the applied axial load. This could permit determining the

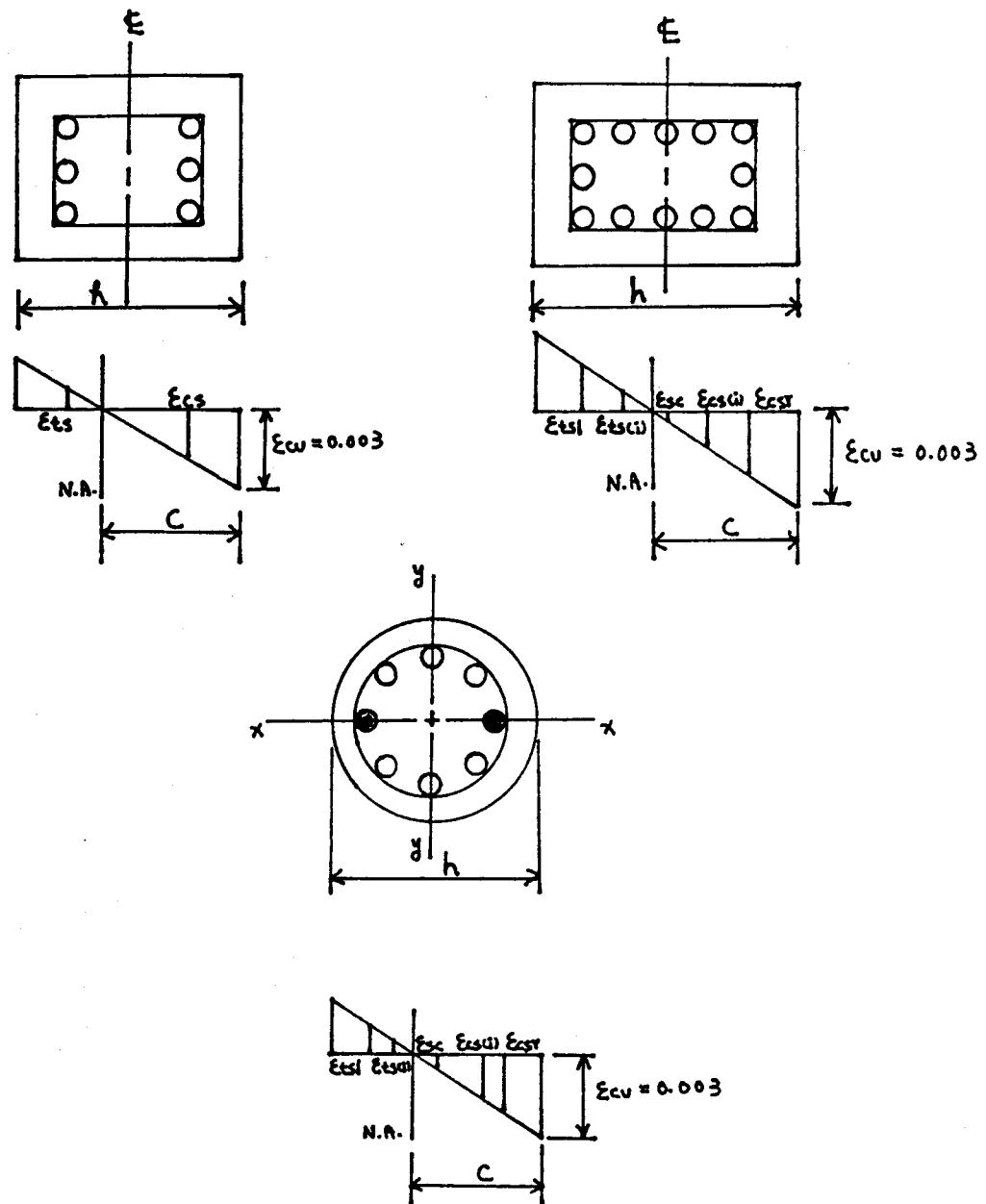


Fig. 3.7.1 Linear strain distribution for bars in two opposite faces only and in both faces and in circular core

Note : ϵ_{tsl} = strain in extreme tension bars

$\epsilon_{ts(i)}$ = strain in tension steel at i column line

ϵ_{sc} = strain in center bars

$\epsilon_{cs(i)}$ = strain in compression steel at i column line

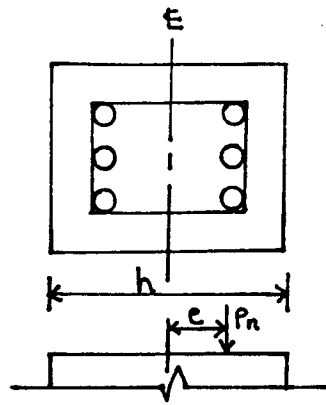
ϵ_{csr} = strain in extreme compression bars

location of the neutral axis for the minimum required moment capacity which also corresponds to the maximum axial load capacity. This axial load capacity will then be computed and compared to the applied axial load. If axial load capacity is greater section is satisfactory. If not trial section must be increased.

Although the program follows the above philosophy, the procedure was altered to overcome certain practical difficulties. For example unless a specific value of the location of neutral axis is used, it is not known whether the steel has yielded. In addition if depth of equivalent rectangular stress block is longer than the distance from extreme compression fibre to centroid of tension reinforcement, a correction for the effect of concrete displaced by compression steel will be considered. For these reasons, the location of the neutral axis was determined using the interval halving technique as described in the following paragraph of this section (Fig. 3.7.2).

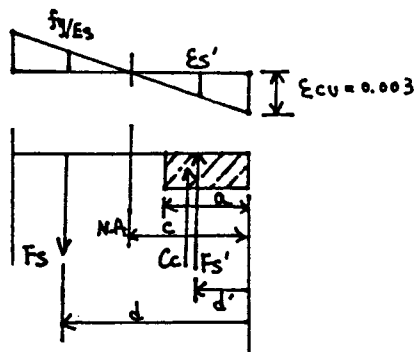
The procedure of the interval halving technique is as follows:

- (i) Initially two limits for the location of the neutral axis are determined; the upper limit as h/β_1 and the lower limit as the location of neutral axis at balanced condition
- (ii) Using a location of neutral axis as midway between the two limits established in (i) compute the moment capacity.
- (iii) If capacity is greater than applied moment, set lower limit as location of neutral axis obtained in (ii)



(i) in balanced condition

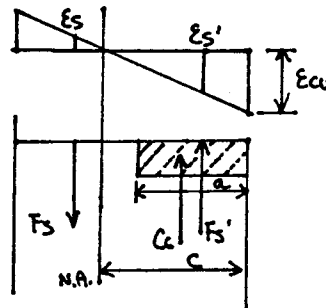
$$C_b = \left(\frac{\epsilon_{cu}}{\epsilon_{cu} + f_y/E_s} \right) d > 0.5h$$



$$\phi M_n \geq M_u = M_{bal.}$$

(ii) in compression region

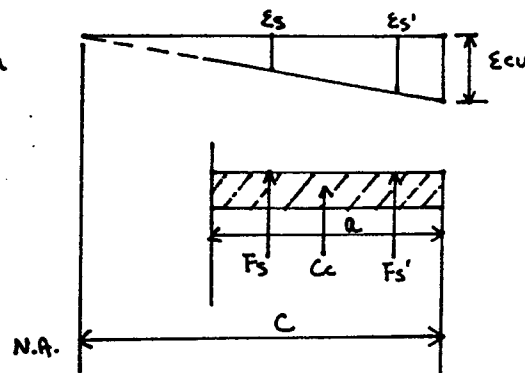
$$h \leq c > C_b$$



$$\phi M_n \geq M_u < M_{bal.}$$

(iii) in maximum load region

$$h/\beta_1 \leq c > h$$



$$\phi M_n \geq M_u = M_{min.}$$

Fig. 3.7.2 Strength capacity at different location of neutral axis

Note : C_c = force in concrete 'fibre'
 F_s' = force in compression steel
 F_s = force in tension steel

and compute new location as midway between limits.

(iv) If capacity is less than applied moment, set upper limit as location of neutral axis obtained in (ii) and compute new location as midway between limits.

(v) Procedure repeated until difference between upper and lower limits is equal to or less than 5 mm or 0.2 in..

Using the above procedure the axial capacity will always be greater than that corresponding to balanced conditions, which is generally the most economical solution. However for uniaxial bending and in check and design mode using the above procedure the axial load corresponding to the moment capacity used for selecting the position of neutral axis will be the maximum axial load and not the minimum axial load. Therefore the program should not be used in check and design mode for members approaching beams, that is for members with larger bending moments about one axis but relatively small axial loads.

The determination of the strength capacity in concrete C_c for circular columns as well as for square columns with a circular core are more complicated than rectangular columns because of the difficulty in treating the circular segments and the steel bars placed around the circular core, however the same philosophy is followed.

For biaxial bending, a similar analysis is performed for M_{uy} , the moment about y-axis acting in conjunction with P_u . The trial neutral axis and the strength capacity about y-axis are determined.

3.7.1 Adequacy Check in Strength

Before the resisting strength capacities of the trial section are compared with applied strength capacities, it must be determined whether the assumed capacity reduction factor of 0.7 for tied and 0.75 for spiral needs to be modified. This is accomplished by comparing the applied axial load P_u with P' which is the smaller of $0.1(f'_c)A_g$ or P_b at the balanced condition. When P_u is less than P' , a new capacity reduction factor is obtained from the program using the expressions (Fig. 3.7.3) :

$$\text{New } \phi = \phi + (0.9 - \phi)(P' - P_u)/P'$$

The axial load capacity is checked against the applied axial load. When the axial load capacity is less than 97% of the applied axial load, in design mode, the section is modified, but when it happens in check mode a message is printed and the program proceeds to the next problem.

The investigation of adequacy check in strength in biaxial bending is not so simple as in uniaxial bending. In biaxial bending, the applied axial load P_u is tested against $0.1(f'_c)A_g$. When the value of P_u is greater than $0.1(f'_c)A_g$, the Bresler Reciprocal Load Method is used to verify the selected section, otherwise the PCA Load Contour Method is used.

In the Bresler Reciprocal Load Method, the general equation used to calculate the provided axial load capacity, ϕP_i , for a selected column section under biaxial bending is:

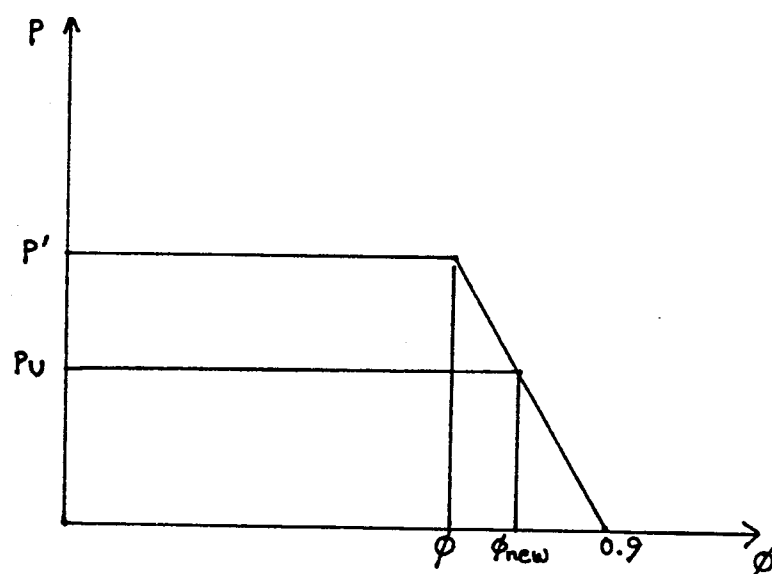


Fig. 3.7.3 Strength-Capacity reduction factor

$$1/(\phi P_i) = 1/(\phi P_{nx}) + 1/(\phi P_{ny}) - 1/(\phi P_o)$$

where

P_i =approximation of nominal axial load strength at eccentricities e_x and e_y

P_{nx} =nominal axial load strength for eccentricity e_y along the y -axis only (x -axis is axis of bending)

P_{ny} =nominal axial load strength for eccentricity e_x along the x -axis only (y -axis is axis of bending)

P_o =nominal axial load strength for zero eccentricity in which $P_o = 0.85f'_c(A_g - A_{st}) + f_y(A_{st})$

In making an adequacy check, the provided axial load capacity, ϕP_i , is compared with the applied axial load P_u , in a manner similar to that in uniaxial bending.

The equation is simple in form and the variables are easy to determine and program. The equation also gives reasonable results for loads above balanced condition load (14). Bresler (25) indicates that the experimental test results have shown the equation to be reasonably accurate when flexure does not govern design.

When the value of P_u is less than $0.1(f'_c)A_g$, the PCA Load Contour Method is used to check the selected section. This approach, developed by Parme, Nieves and Gouwens, is an extension of the Bresler Load Contour Method described in the last paragraph of this section by approximating the load contour equations as straight lines (14).

$$(M_{ox})_{equiv} = M_{ux} + M_{uy}(M_{ox}/M_{oy})[(1 - B_b)/B_b]$$

for $M_{uy}/M_{ux} \leq M_{oy}/M_{ox}$

or $(M_{oy})_{equiv} = M_{uy} + M_{ux}(M_{oy}/M_{ox})[(1-B_b)/B_b]$

for $M_{uy}/M_{ux} \geq M_{oy}/M_{ox}$

in which $B_b = B_{25} + 0.2(P_u/\phi C_c - 0.25)/(0.85 + C_s/C_c)$

for $P_u/\phi \geq 0.25C_c$

or $B_b = B_{25} + (0.25 - P_u/\phi C_c)^2 (0.85 - C_s/2C_c)$

for $P_u/\phi < 0.25C_c$

where $C_c = f'_c C_x C_y$

$C_s = A_s f_y$

B_{25} is the minimum value of B_b which occurs at $P_u/\phi = 0.25C_c$, M_{ux} and M_{uy} are the applied moment strengths of a column about its x-axis and y-axis, respectively, M_{ox} is the resisting moment strength of a column about its x-axis only without bending about the y-axis, and M_{oy} is the resisting moment strength of a column about its y-axis only without bending about the x-axis.

In the region where $C_s/C_c \geq 0.5$:

$$B_{25} = 0.485 + 0.03C_c/C_s$$

In the region where $C_s/C_c < 0.5$:

$$B_{25} = 0.545 + 0.35(0.5 - C_s/C_c)$$

Having determined which portion of the bilinear interaction curve is to be used, the selected section is verified. Design is adequate when

$$[M_{ux}/(M_{ox})_{equiv}] + [M_{uy}/(M_{oy})_{equiv}][(1-B_b)/B_b] \leq 1.0$$

for $M_{uy}/M_{ux} \leq M_{oy}/M_{ox}$

$$\text{or } [M_{uy}/(M_{oy})_{equiv}] + [M_{ux}/(M_{ox})_{equiv}][(1-B_b)/B_b] \leq 1.0$$

for $M_{uy}/M_{ux} \geq M_{oy}/M_{ox}$

When the above equation gives a value greater than 1.03, in design mode the trial section needs to be modified, but when it happens in check mode a message is printed and the program proceeds to the next problem.

Parme, et al (25) indicate that this PCA approach yields values sufficiently accurate for design, even for sections that are rectangular or have unequal reinforcement in the two adjacent faces. Gouwens (14) mentions that this procedure is a simple but accurate, conservative and complete method of design.

For the Bresler Load Contour Method, the general equation is

$$[M_{ux}/(M_{ox})_{\text{equi}}]^{\alpha} + [M_{uy}/(M_{oy})_{\text{equi}}]^{\beta} = 1.0$$

in which the values of the exponents α and β are a function of the amount, distribution and location of reinforcement, the dimension of the column, and the strength and elastic properties of the steel and concrete. Wang and Salmon (26) mentions that a value of α that is applicable to the particular column has to be investigated. Bresler (25) indicates that it is reasonably accurate to assume that $\alpha = \beta$. With α set at unity, the equation will always yield conservative results but the use of the equation becomes very conservative for high axial loads or low percentage of reinforcement and should only be used when the applied axial load P_u is less than $0.1(f'_c)A_g$.

3.8 Modification of Trial Section

When the trial section is not adequate to provide the required strength, the longitudinal reinforcement is incremented first. When the specified number of bars is not entered or the required total number of bars is less than the maximum number of bars for that section, the required total number of bars is incremented by an integer of 2, and then the process of selecting longitudinal reinforcement is repeated. If the specified number of bars is entered or the required total number of bars has reached the maximum number of bars, the bar size will be increased providing it has not reached the maximum bar size limit. Should a suitable reinforcement pattern not be found, the concrete dimensions will be increased in accordance with the rules for incrementing concrete section as outlined in Section 3.3.3.

3.9 Evaluation of Longitudinal Splices

Once a trial cross section is obtained that is consistent with the entered restraints and is adequate to satisfactorily carry the loading, this trial section is selected as the final design. The program then proceeds to examine possible splice types and to select the lateral reinforcement.

For tied columns, a check is made to determine whether the reinforcement pattern can be accommodated with normal lap splices and/or tangential lap splices (3). See Fig. 3.9.1 and Fig. 3.9.2. This is accomplished by the following

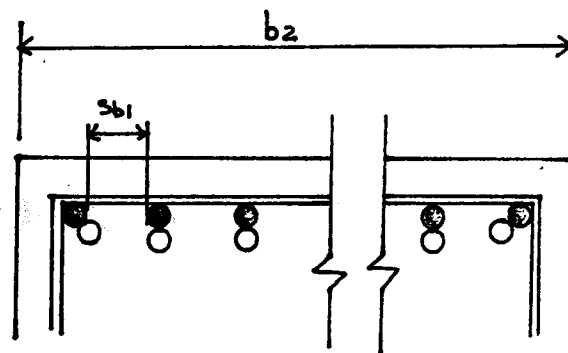


Fig. 3.9.1 Normal lap splices

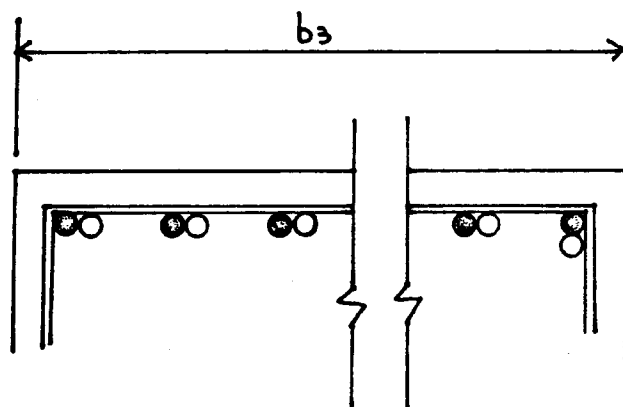


Fig. 3.9.2 Tangential lap splices

expressions :

For bar diameter, db , is less than or equal to #25 bar in CSA or #8 bar in ACI code,

$$b_2 = b_1 + [(2s_{b_1} + 2db) \cos \theta - 0.586db - 2s_{b_1}]$$

where $b_1 = 2(\text{cover} + \text{tie diam.}) + n * db + (n-1)s_{b_1}$

$$\theta = \arcsin[(1 - \sqrt{0.5})db] / (s_{b_1} + db)$$

for normal lap splices

$$b_3 = 2(\text{cover} + \text{tie diam.}) + (2n-1)db + (n-1)s_{b_1}$$

for tangential lap splices

For bar diameter, db , is greater than #25 bar in CSA or #8 bar in ACI,

$$b_2 = b_1 + 1.38db$$

where $b_1 = 2(\text{cover} + \text{tie diam.}) + n * db + (n-1)1.5db$

for normal lap splices

$$b_3 = 2(\text{cover} + \text{tie diam.}) + (2n-1)db + (n-1)1.5db$$

for tangential lap splices

in which n is number of bars per face, but for the case where bars are in both faces, n is the larger number of bars either in x or y -face, b_1, b_2 and b_3 are the minimum column dimension where bars are located, and s_{b_1} is the minimum clear bar spacing. If b_2 and/or b_3 are less than the design column dimension where the reinforcement is located, an appropriate message is printed indicating which type of splices can be accommodated.

For spiral columns, a check is made to determine whether the reinforcement can be accommodated with normal lap splices and/or tangential lap splices (3). This is accomplished by determining the number of bars n_n and n_t which are the maximum number of bars that can be accommodated using normal and tangential splices, respectively using the following expressions :

When bar diameter, db , is less than or equal to #25 bar in CSA or #8 in ACI code,

$$n_n = \frac{180}{\arcsin[(s_b + db)/(h - 3db - 2(\text{cover} + \text{spiral diam.}))]}$$

$$n_t = 180/R$$

$$\text{where } R = \arcsin[(s_b + db)/(h - db - 2(\text{cover} + \text{spiral diam.}))] + \arcsin[db/(h - db - 2(\text{cover} + \text{spiral diam.}))]$$

When bar diameter, db , is greater than #25 in CSA or #8 in ACI code,

$$n_n = \frac{180}{\arcsin[(2.5db)/(h - 3db - 2(\text{cover} + \text{spiral diam.}))]}$$

$$n_t = 180/R_1$$

$$\text{where } R_1 = \arcsin[(2.5db)/(h - db - 2(\text{cover} + \text{spiral diam.}))] + \arcsin[db/(h - db - 2(\text{cover} + \text{spiral diam.}))]$$

A message indicating the types of splices that can be accommodated is printed.

Consistent with the building code provisions, the program does not permit the use of lap splices for bar diameters greater than #35 in CSA or #11 in ACI.

3.10 Selection of Lateral Reinforcement

For tied columns, the required tie spacing per set is the least of:

- (a) 16 times the diameter of the smallest longitudinal bars;
- (b) 48 times tie diameter;
- (c) the least dimension of the compression members.

Lateral ties are required for all corner bars. If the minimum clear bar spacing between adjacent interior bars is greater than a predetermined maximum clear tied bar spacing (default values of 150 mm in *CSA* or 6 in. in *ACI*), lateral ties are required for interior bars.

For spiral columns, the ratio of spiral reinforcement, ρ_s , defined as :

$$\rho_s = \frac{\text{Volume of spiral in one loop}}{\text{Volume of core for a length } S}$$

$$= \frac{\pi d_s^2 (D_c - d_s)}{D_c^2 S}$$

Also the minimum value of spiral reinforcement to qualify as a spiral column has been determined from tests and is specified in the building codes as

$$\rho_s = 0.45(A_g/A_c - 1)(f'_c/f_{sy})$$

in which D_c is the diameter of the core, d_s is the lateral spiral diameter, and S is the maximum spacing between spirals. These two equations for ρ_s are used to solve for S .

The minimum number of vertical spacers is also determined consistent with the code requirements.

4. ILLUSTRATIVE EXAMPLES

This chapter is intended to illustrate the use of the program, COLUMN1. Detailed mode processing and control commands and data entries are discussed in Appendix A : USER'S MANUAL. Several examples demonstrating most of the features of the program are presented. Input data files and computer standard outputs of sample problems are also presented.

4.1 Example 1 - Short Tied Column (Braced)

Design a square tied column for uniaxial bending about the x-axis with bars distributed along both faces in accordance with the requirement of CSA A23.3-M77. Column is braced with unsupported length of 3000 mm. The material properties for this project are $f'_c = 30$ MPa, $f_y = 400$ MPa and $f_{sy} = 400$ MPa. All the loadings are unfactored and three load cases are considered.

Load Condition	Axial Force (KN)	Moment at Top (KN-M)	Moment at Bottom (KN-M)
LC1 :Dead Load	1880	120	-120
Live Load	2720	190	50
LC2 :Dead Load	1880	120	-120
Live Load	2160	270	70
LC3 :Dead Load	2000	150	150
Live Load	3000	200	200

This example is executed in batch mode. The computer input is presented in Fig. 4.1.1. The computer output is presented in Fig. 4.1.2 and the selected column section is shown in Fig. 4.1.5.

In the input file when the optional entries are not entered, default values are assigned. For this example, default values for the maximum reinforcement ratio of 0.03 and minimum bar size of #15 are used. The program determines whether column deflects in single or double curvature by examining the signs assigned to the moments. A negative sign to one of the entered moments indicates the column bends in double curvature.

The output consists of an echo check of the essential design parameters including the design load conditions and the required section. Additional design messages are printed. For the section selected based on the first load case, the axial load capacity is only 0.986 times the applied axial load. Although this value is 1.4% low it is within the tolerance of 3% underdesign used by the program. For the additional load cases, the section is not altered but is checked for adequacy. For the second load case, the section is satisfactory. For the last load case, the section is not satisfactory to carry the imposed loadings. This is due in part to the moment carrying bending in single curvature. At this stage, the execution of the program is terminated because the variable TPROB is given a value of 6.

In keeping with the philosophy of the program in selecting a satisfactory solution using the minimum concrete section and bar size consistent with the restraints, the design terminated when a satisfactory design using #25 bars was obtained. This will generally be the most economical

section. However in some cases the designer may want to see the effect of using a still larger bar which is accomplished by rerunning the problem with a specified minimum bar size of #30.

To illustrate the revision options using the interactive mode, Example 1 is rerun using the interactive mode and only the first load case. The selection of the interactive mode must, of course, be specified at the time of running the program. Input lines are then prompted and entered separately as requested and would essentially be the same as the first 7 lines of Fig. 4.1.1. The solution would be the same as obtained for the batch run and is shown in the first part of Fig. 4.1.4 which is the output using the hard copy option after all revisions have been made and the design terminated. Fig. 4.1.3 illustrates that portion of what is displayed on the screen at the time the revision of the minimum bar size is changed to #30. Note that the revision option permits altering only one line at a time but the entire line containing the revision must be entered. Once the revised line has been entered, the program executes the design using the revised values resulting in the display in the bottom portion of Fig. 4.1.4.

It should be noted that although the format required for an entry line is displayed the entry shown uses the option available under the MTS operating system which uses a comma to designate the end of a field specification. Note also that a blank space is required with no comma after the

entry for reinforcement pattern since this is in A2 format.

It is seen that the new design shown in Fig. 4.1.6 has the same concrete dimensions as the previous one but uses 8 #30 bars which is 6.1% less steel for the design load condition. In this example the use of the larger bars would appear more economical but as with the first design the revised section is still not adequate for the last loading case.

SHORT TIED COLUMN1 (BRACED)

C	T	3	0								
		30.	400.	400.	0.					0.	
S		0.0	0.0	0.00	0.0	0.0	0.0				
B		0.0	0	0	0	0	0	0	0		
U	1	1.000		3000.	0	0.000				0.	
U		1880.	2720.		120.	190.				-120.	50.
D		0.	0.		0.	0.					
U		1880.	2160.		120.	270.				-120.	70.
L		0.	0.		0.	0.					
U		2000.	3000.		150.	200.				150.	200.
L		0.	0.		0.	0.					
6											

Fig. 4.1.1 Batch input for example 1

SHORT TIED COLUMN1 (BRACED)

PAGE 1

00:52:23 JAN 27, 1984

TYPE = TIED

DESIGN CODE = CSA A23.3-M77

ABOUT X-AXIS : BRACED, EFF. LENGTH = 3000. MM

MATERIAL PROPERTIES :

CONCRETE : COMP. STRENGTH = 30. MPA
 DENSITY = 2400. KG/CUBIC METER
 MODULUS OF ELAS. = 27691. MPA
 LONG. STEEL : YIELD STRESS = 400. MPA
 MODULUS OF ELAS. = 200000. MPA
 TIES : YIELD STRESS = 400. MPA

DESIGN LOAD CONDITION :

PU = 7256. KN
 MUX = 491. KN-M, MUY = 0. KN-M

CONCRETE DIMENSION :

CX = 650. MM, CY = 650. MM

LONGITUDINAL REINFORCEMENT : REINFORCEMENT RATIO = 0.014

8- #25 BARS IN X-FACES
 4- #25 BARS IN Y-FACES

TIES :

#10 @ 403. MM SPACING

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN DOUBLE CURVATURE ABOUT X-AXIS

FOR SECTION SELECTED THE AXIAL LOAD CAPACITY
 UNDER APPLIED MOMENT IS 0.986 TIMES
 THE APPLIED AXIAL LOAD

TANGENTIAL LAP SPLICES CAN BE USED

LATERAL TIES REQUIRED FOR INTERIOR BARS
 IN X-FACE

LATERAL TIES REQUIRED FOR INTERIOR BARS
 IN Y-FACE

ADDITIONAL LOAD CONDITION :

PU = 6304. KN

Fig. 4.1.2 Batch output for example 1 (continued)

SHORT TIED COLUMN1 (BRACED)

PAGE 2

MUX = 627. KN-M, MUY = 0. KN-M

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN DOUBLE CURVATURE ABOUT X-AXIS

FOR SECTION ENTERED THE AXIAL LOAD CAPACITY
UNDER APPLIED MOMENT IS 1.009 TIMES
THE APPLIED AXIAL LOAD

ADDITIONAL LOAD CONDITION :

PU = 7900. KN
MUX = 550. KN-M, MUY = 0. KN-M

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

FOR SECTION ENTERED THE AXIAL LOAD CAPACITY
UNDER APPLIED MOMENT IS 0.906 TIMES
THE APPLIED AXIAL LOAD

** SECTION IS NOT ADEQUATE FOR THIS LOADING **

Fig. 4.1.2 (continued)

NEXT OPTION=? REVISE ABOVE PROBLEM,ENTER 1
 GO TO NEXT PROBLEM,ENTER 2
 TERMINATION OF PROGRAM,ENTER 3

1

REVISE OPTION=? ALTER MATERIAL ONLY,ENTER 1
 ALTER CONCRETE GEOM. ONLY,ENTER 2
 ALTER STEEL GEOM. ONLY,ENTER 3
 ALTER STABILITY GEOM. ONLY,ENTER 4
 ALTER LOADS ONLY,ENTER 5

3

RFACE=? PG=? BMAXX=? NBS=? BSMIN=? BSMAX=? DS=? NX=? NY=?
 (A2,F7.4,4I4,I3,2I2)
 REINF. PATTERN : X FOR X-FACES, Y FOR Y-FACE
 B FOR BOTH FACES, C FOR CIRCULAR
 REINF. RATIO : 0 DEFAULT TO 0.03
 MAX. PERMISSIBLE NO. BARS : 0 DEFAULT VALUE BY PROGRAM
 TOTAL NO. BARS : (NX, NY REQUIRED IF BARS IN BOTH FACES)
 MIN. BAR SIZE : 0 DEFAULT TO 15 OR 5
 MAX. BAR SIZE : 0 DEFAULT TO 35 OR 11
 MIN. LATERAL REINF. : 0 DEFAULT TO 10 OR 3
 NO. BARS IN EACH X-FACE : (NBS IS ENTERED)
 NO. BARS IN EACH Y-FACE : (NBS IS ENTERED)
 B 0.,0,0,30,

Fig. 4.1.3 Interactive input for example 1

SHORT TIED COLUMN1 (BRACED)

PAGE 1

00:57:03 JAN 27, 1984

TYPE = TIED

DESIGN CODE = CSA A23.3-M77

ABOUT X-AXIS : BRACED, EFF. LENGTH = 3000. MM

MATERIAL PROPERTIES :

CONCRETE : COMP. STRENGTH = 30. MPA
 DENSITY = 2400. KG/CUBIC METER
 MODULUS OF ELAS. = 27691. MPA
 LONG. STEEL : YIELD STRESS = 400. MPA
 MODULUS OF ELAS. = 200000. MPA
 TIES : YIELD STRESS = 400. MPA

DESIGN LOAD CONDITION :

PU = 7256. KN
 MUX = 491. KN-M, MUY = 0. KN-M

CONCRETE DIMENSION :

CX = 650. MM, CY = 650. MM

LONGITUDINAL REINFORCEMENT : REINFORCEMENT RATIO = 0.014

8- #25 BARS IN X-FACES
 4- #25 BARS IN Y-FACES

TIES :

#10 @ 403. MM SPACING

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN DOUBLE CURVATURE ABOUT X-AXIS

FOR SECTION SELECTED THE AXIAL LOAD CAPACITY
 UNDER APPLIED MOMENT IS 0.986 TIMES
 THE APPLIED AXIAL LOAD

TANGENTIAL LAP SPLICES CAN BE USED

LATERAL TIES REQUIRED FOR INTERIOR BARS
 IN X-FACE

LATERAL TIES REQUIRED FOR INTERIOR BARS
 IN Y-FACE

REVISE OPTION = ALTER STEEL OF GEOMETRY PROPERTIES ONLY

CONCRETE DIMENSION :

Fig. 4.1.4 Interactive output for example 1 (continued)

SHORT TIED COLUMN1 (BRACED)

PAGE 2

CX = 650. MM, CY = 650. MM

LONGITUDINAL REINFORCEMENT : REINFORCEMENT RATIO =0.013

6- #30 BARS IN X-FACES
2- #30 BARS IN Y-FACES

TIES :

#10 @ 478. MM SPACING

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN DOUBLE CURVATURE ABOUT X-AXIS

FOR SECTION SELECTED THE AXIAL LOAD CAPACITY
UNDER APPLIED MOMENT IS 0.975 TIMES
THE APPLIED AXIAL LOAD

TANGENTIAL LAP SPLICES CAN BE USED

Fig. 4.1.4 (continued)

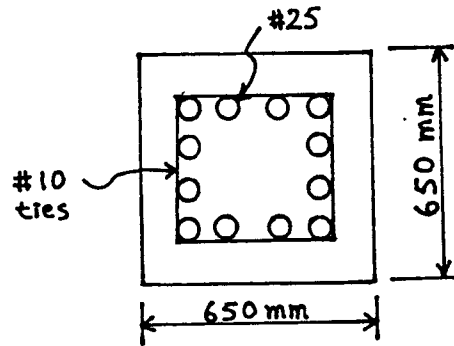


Fig. 4.1.5 Column section for example 1

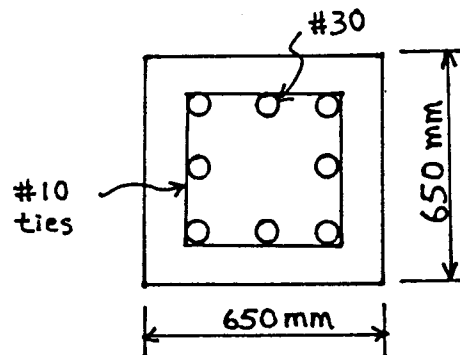


Fig. 4.1.6 Column section for example 1 specifying #30 bars

4.2 Example 2 - Spiral Column (CSA)

Design a circular spiral column for uniaxial bending about the x-axis in accordance with the requirement of CSA A23.3-M77. Column is braced with the unsupported length of 3000 mm. The material properties for this project are $f'_c = 30$ MPa, $f_y = 350$ MPa and $f_{sy} = 300$ MPa. All the loadings are factored. This column is subjected to an axial dead and live load of 3520 KN and 5280 KN respectively, with a dead load and a live load moment at both the top and bottom of column of 112 KN-M and 168 KN-M respectively.

This example is executed in batch mode. The computer input is presented in Fig. 4.2.1. The computer output is presented in Fig. 4.2.2 and the selected column section is shown in Fig. 4.2.3.

The computer input and output are similar to that in example 1. In this example, the actual computed eccentricity is less than the specified minimum, therefore the minimum moment specified in the building codes is used. The axial load capacity is only 0.974 times the applied axial load which while 2.6% low is within the tolerance of 3% underdesign used by the program.

```

SPIRAL COLUMN (CSA)
C S 1 0
    30. 350. 300. 0. 0.
C 0.0 0.0 0.00 0.0 0.0 0.0
C 0.0 0 0 0 0 0
U 1 1.000 3000. 0 0.000
F 3520. 5280. 112. 168. 112. 168.
D 0. 0. 0. 0.
6

```

Fig. 4.2.1 Input for example 2

SPIRAL COLUMN (CSA)

PAGE 1

22:19:04 FEB 1, 1984

TYPE = SPIRAL

DESIGN CODE = CSA A23.3-M77

ABOUT X-AXIS : BRACED, EFF. LENGTH = 3000. MM

MATERIAL PROPERTIES :

CONCRETE : COMP. STRENGTH = 30. MPA
 DENSITY = 2400. KG/CUBIC METER
 MODULUS OF ELAS. = 27691. MPA
 LONG. STEEL : YIELD STRESS = 350. MPA
 MODULUS OF ELAS. = 200000. MPA
 SPIRAL : YIELD STRESS = 300. MPA

DESIGN LOAD CONDITION :

PU = 8800. KN
 MUX = 280. KN-M, MUY = 0. KN-M

CONCRETE DIMENSION :

DIAMETER = 700. MM

LONGITUDINAL REINFORCEMENT : REINFORCEMENT RATIO = 0.026

14- #30 BARS IN CIRCULAR

SPIRALS :

#10 WITH 50. MM PITCH AND 3 SPACERS

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

MINIMUM ECCENTRICITY ABOUT X-AXIS GOVERNS
 DESIGN MOMENT ABOUT X-AXIS = 308.
 IN PROJECT UNITS

FOR SECTION SELECTED THE AXIAL LOAD CAPACITY
 UNDER APPLIED MOMENT IS 0.974 TIMES
 THE APPLIED AXIAL LOAD

Fig. 4.2.2 Output for example 2

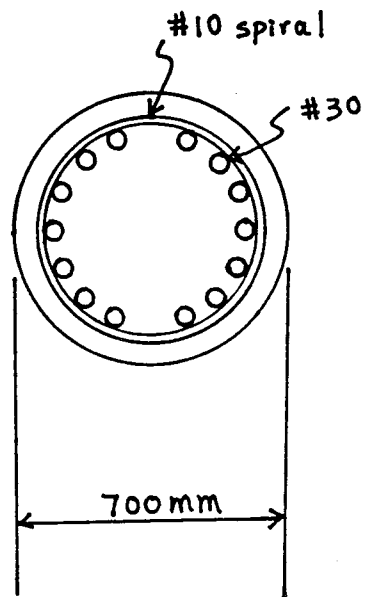


Fig. 4.2.3 Column section for example 2

4.3 Example 3 - Slender Braced Column (ACI)

Design a rectangular tied column for uniaxial bending about the x-axis with bars distributed along both faces in accordance with the requirements of ACI 318-77. Column is braced with the unsupported length of 330 in. The material properties for this project are $f'_c = 4000$ psi, $f_y = 60000$ psi and $f_{sy} = 60000$ psi. Select the concrete dimension ratio of 0.9, the minimum and the maximum bar size of #9 and #18 respectively. All the loadings are factored. This column is subjected to an axial dead and live load of 224 kips and 336 kips respectively, a dead load moment at both the top and bottom of column of 131 kip-ft, and a live load moment at the top and bottom of column of 115 kip-ft and 196 kip-ft respectively.

This example is executed in batch mode. The computer input is presented in Fig. 4.3.1. The computer output is presented in Fig. 4.3.2 and the selected column section is shown in Fig. 4.3.3.

The computer input and output are similar to that in example 1. In this example, the column was found to be "slender" and the moment magnification factor was computed as 1.41. The section selected is adequate for this loading and is 5.3% oversized.

The data for this example was taken from "COLUMN EXAMPLE 2" in 'Design Handbook In Accordance With the Strength Design Method of ACI 318-77 : Volume 2 - Columns' (3). In "COLUMN EXAMPLE 2", the moment magnification factor

was found to be 1.44 and 12 #10 bars were used due to the restriction placed on the interaction diagrams used in the example which require reinforcement along all faces to have equal areas in each face. This restriction does not exist in COLUMN1 so that different areas of steel in the different faces can be considered resulting in only 10 #10 bars being selected. Except for the reinforcement, COLUMN1 gives corresponding results to those in "DESIGN EXAMPLE 2".

SLENDER BRACED COLUMN (ACI)

A	T	1	0															
		4000.	60000.	60000.		0.												0.
R		0.0	0.0	0.9		0.0	0.0	0.0										
B		0.0		0	0	9	18											
U		1	1.000			330.	0	0.000										0.
F			224.		336.		131.		115.								131.	196.
D			0.		0.		0.		0.									
6																		

Fig. 4.3.1 Input for example 3

SLENDER BRACED COLUMN (ACI)

PAGE 1

00:49:44 JAN 27, 1984

TYPE = TIED DESIGN CODE = ACI 318-77

ABOUT X-AXIS : BRACED, EFF. LENGTH = 330. IN.

MATERIAL PROPERTIES :

CONCRETE : COMP. STRENGTH = 4000. PSI
 DENSITY = 145. PCF
 MODULUS OF ELAS. = 3644149. PSI
 LONG. STEEL : YIELD STRESS = 60000. PSI
 MODULUS OF ELAS. = 29000000. PSI
 TIES : YIELD STRESS = 60000. PSI

DESIGN LOAD CONDITION :

PU = 560. KIPS
 MUX = 327. KIP-FT, MUY = 0. KIP-FT

CONCRETE DIMENSION :

CX = 20.0 IN., CY = 22.0 IN.

LONGITUDINAL REINFORCEMENT : REINFORCEMENT RATIO = 0.029

6- #10 BARS IN X-FACES
 4- #10 BARS IN Y-FACES

TIES :

3 @ 18.0 IN. SPACING

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

MAGNIFICATION FACTOR FOR MOMENT ABOUT X-AXIS = 1.41

FOR SECTION SELECTED THE AXIAL LOAD CAPACITY
 UNDER APPLIED MOMENT IS 1.053 TIMES
 THE APPLIED AXIAL LOAD

TANGENTIAL LAP SPLICES CAN BE USED

LATERAL TIES REQUIRED FOR INTERIOR BARS
 IN Y-FACE

Fig. 4.3.2 Output for example 3

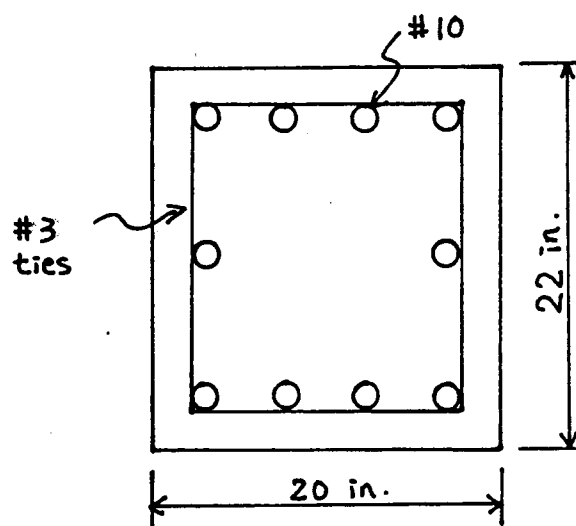


Fig. 4.3.3 Column section for example 3

4.4 Example 4 - Square Tied Column in Biaxial Bending

Design a 20 x 20 in. square tied column for biaxial bending about the x-axis and the y-axis, respectively with bars distributed along both faces in accordance with the requirement of ACI 318-77. Column is braced with the unsupported length of 100 in.. The material properties for this project are $f'_c = 6000$ psi, $f_y = 60000$ psi and $f_{sy} = 60000$ psi. Select the maximum value for reinforcement ratio of 0.04, and specify the bar size of #10. All the loadings are factored and three load cases are considered.

Load Condition	Axial Force (KIPS)	Moment at Top x-axis y-axis (KIP-FT)		Moment at Bottom x-axis y-axis (KIP-FT)	
LC1 :Dead Load	100	38	175	38	175
Live Load	100	38	175	38	175
LC2 :Dead Load	56	49	21	49	21
Live Load	84	70	32	70	32
LC3 :Dead Load	1000	400	0	400	0
Live Load	0	0	0	0	0

This example is executed in batch mode. The computer input is presented in Fig. 4.4.1. The computer output is presented in Fig. 4.4.2 and the selected column section is shown in Fig. 4.4.3.

The dead and live load for both axial and moments loadings must be entered individually whether the loadings are factored or unfactored since in considering the slenderness effects, the maximum design dead load moment is taken into account.

The computer input and output are similar to that in example 1. For the first and second load cases, the PCA Load

Contour Method is used to check the selected section for adequacy in strength because the value of P_u is less than $0.1(f'c)A_g$, and the section is found to be satisfactory to carry the imposed loadings. For the last load case, the minimum moment specified in the building codes governs, and the section is not satisfactory to carry the imposed loadings based on the axial load capacity is 0.953 times the applied axial load.

SQUARE TIED COLUMN IN BIAXIAL BENDING

A	T	3	0							
		6000.	60000.	60000.		0.			0.	
S		20.0	0.0	0.0		0.0	0.0	0.0		
B		0.04	0	0	10	10	0	0	0	
B		1	1.000		100.	1	1.000		100.	
F		100.	100.		38.		38.		38.	38.
D		175.	175.		175.		175.			
F		56.	84.		49.		70.		49.	70.
L		21.	32.		21.		32.			
F		1000.	0.		400.		0.		400.	0.
L		0.	0.		0.		0.			
6										

Fig. 4.4.1 Input for example 4

SQUARE TIED COLUMN IN BIAXIAL BENDING

PAGE 1

22:21:15 FEB 1, 1984

TYPE = TIED

DESIGN CODE = ACI 318-77

ABOUT X-AXIS : BRACED, EFF. LENGTH = 100. IN.

ABOUT Y-AXIS : BRACED, EFF. LENGTH = 100. IN.

MATERIAL PROPERTIES :

CONCRETE : COMP. STRENGTH = 6000. PSI
 DENSITY = 145. PCF
 MODULUS OF ELAS. = 4463153. PSI
 LONG. STEEL : YIELD STRESS = 60000. PSI
 MODULUS OF ELAS. = 29000000. PSI
 TIES : YIELD STRESS = 60000. PSI

DESIGN LOAD CONDITION :

PU = 200. KIPS
 MUX = 76. KIP-FT, MUY = 350. KIP-FT

CONCRETE DIMENSION :

CX = 20.0 IN., CY = 20.0 IN.

LONGITUDINAL REINFORCEMENT : REINFORCEMENT RATIO = 0.032

8- #10 BARS IN X-FACES
 2- #10 BARS IN Y-FACES

TIES :

3 @ 18.0 IN. SPACING

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT Y-AXIS

EQUIVALENT REQUIRED MOMENT ABOUT Y-AXIS
 ONLY = 446. IN PROJECT UNITS

SELECTED SECTION IS ADEQUATE BASED ON
 MOMENT CAPACITIES

TANGENTIAL LAP SPLICES CAN BE USED

LATERAL TIES REQUIRED FOR INTERIOR BARS
 IN X-FACE

Fig. 4.4.2 Output for example 4 (continued)

SQUARE TIED COLUMN IN BIAXIAL BENDING

PAGE 2

ADDITIONAL LOAD CONDITION :

PU = 140. KIPS
 MUX = 119. KIP-FT, MUY = 53. KIP-FT

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT Y-AXIS

EQUIVALENT REQUIRED MOMENT ABOUT X-AXIS
 ONLY = 183. IN PROJECT UNITS

ENTERED SECTION IS ADEQUATE BASED ON
 MOMENT CAPACITIES

ADDITIONAL LOAD CONDITION :

PU = 1000. KIPS
 MUX = 400. KIP-FT, MUY = 0. KIP-FT

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT Y-AXIS

MINIMUM ECCENTRICITY ABOUT Y-AXIS GOVERNS
 DESIGN MOMENT ABOUT Y-AXIS = 100.
 IN PROJECT UNITS

FOR SECTION ENTERED THE AXIAL LOAD CAPACITY
 UNDER APPLIED MOMENT IS 0.953 TIMES
 THE APPLIED AXIAL LOAD

** SECTION IS NOT ADEQUATE FOR THIS LOADING **

Fig. 4.4.2 (continued)

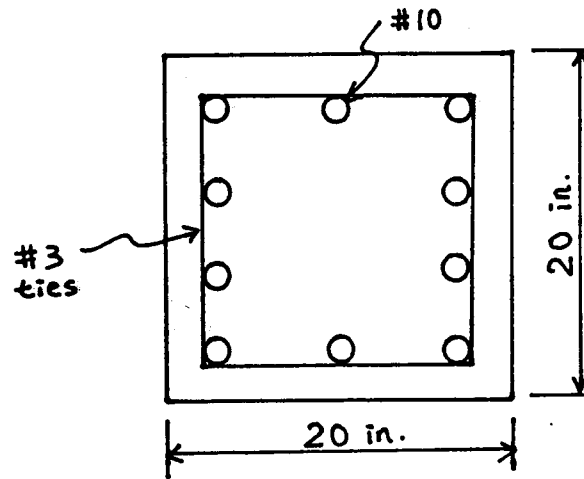


Fig. 4.4.3 Column section for example 4

4.5 Example 5 - Rect. Tied Column Check

Check a 15 x 24 in. rectangular tied column for uniaxial bending about the x-axis with 6 #8 bars distributed along the y-faces only in accordance with the requirement of ACI 318-77. Column is braced with the unsupported length of 100 in.. The material properties for this project are $f'_c = 3000$ psi, $f_y = 50000$ psi and $f_{sy} = 50000$ psi. All the loadings are factored.

Load Condition	Axial Force (KIPS)	Moment at Top (KIP-FT)	Moment at Bottom (KIP-FT)
LC1 :Dead Load	400	266	266
Live Load	0	0	0
LC2 :Dead Load	460	300	300
Live Load	0	0	0

This example is executed in batch mode. The computer input and output are presented in Fig. 4.5.1 and Fig. 4.5.2 respectively.

The computer input and output are similar to that in example 1. In this example, the section is not altered but is checked for adequacy. For the first load case, the entered section is satisfactory. For the second load case, the entered section is not satisfactory to carry the imposed loadings.

RECT. TIED COLUMN CHECK

A	T	2						
			3000.	50000.	50000.			
R			15.0	24.0				
Y			0.0	0	6	8	8	
U			1	1.000		100.		
F			400.		0.	266.	0.	266.
C								
F			460.		0.	300.	0.	300.
L								
6								

Fig. 4.5.1 Input for example 5

RECT. TIED COLUMN CHECK

PAGE 1

21:49:21 FEB 2, 1984

TYPE = TIED

DESIGN CODE = ACI 318-77

ABOUT X-AXIS : BRACED, EFF. LENGTH = 100. IN.

MATERIAL PROPERTIES :

CONCRETE : COMP. STRENGTH = 3000. PSI
 DENSITY = 145. PCF
 MODULUS OF ELAS. = 3155926. PSI
 LONG. STEEL : YIELD STRESS = 50000. PSI
 MODULUS OF ELAS. = 29000000. PSI
 TIES : YIELD STRESS = 50000. PSI

CHECK LOAD CONDITION :

PU = 400. KIPS
 MUX = 266. KIP-FT, MUY = 0. KIP-FT

CONCRETE DIMENSION :

CX = 15.0 IN., CY = 24.0 IN.

LONGITUDINAL REINFORCEMENT : REINFORCEMENT RATIO = 0.013

6- # 8 BARS IN Y-FACES ONLY

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

FOR SECTION ENTERED THE AXIAL LOAD CAPACITY
 UNDER APPLIED MOMENT IS 0.985 TIMES
 THE APPLIED AXIAL LOAD

ADDITIONAL LOAD CONDITION :

PU = 460. KIPS
 MUX = 300. KIP-FT, MUY = 0. KIP-FT

ADDITIONAL COMMENTS :

COLUMN DEFLECTS IN SINGLE CURVATURE ABOUT X-AXIS

FOR SECTION ENTERED THE AXIAL LOAD CAPACITY
 UNDER APPLIED MOMENT IS 0.865 TIMES
 THE APPLIED AXIAL LOAD

** SECTION IS NOT ADEQUATE FOR THIS LOADING **

Fig. 4.5.2 Output for example 5

5. SUMMARY AND CONCLUSIONS

The reinforced concrete column design computer program, COLUMN1, run either in batch mode or in interactive mode has been developed to provide an opportunity for designers with little experience such as students to obtain more quickly a feel for reinforced concrete column design. The program is capable of performing design or analysis of most columns that will be encountered in practice in accordance with either the CSA3-A23.3-M77 or the ACI 318-77 building codes and so may be used in design offices or incorporated in a CAD system. A single column design or multiple column designs can be solved during a single run of the program. In interactive mode, a hard copy output may be obtained at the end of each run.

In design mode, the program proceeds to find the minimum feasible concrete section that may be used for the specified loads, moments, and material properties using the least possible reinforcement consistent with input constraints. This trial section is checked for slenderness effects, bar spacing and conformance to building code requirements. The user has the option of specifying all or any portion of the final cross-section. When the complete cross-section is entered (check mode) a message indicating the capacity of the column section is printed including a warning if not adequate.

A number of design examples are illustrated together with the computer outputs from the program COLUMN1. Results

compare favorably with published concrete column designs obtained using traditional methods.

REFERENCES

1. ACI Code, "Building Code Requirements for Reinforced Concrete (ACI-318-77)", American Concrete Institute, Detroit, 1977.
2. ACI Commentary, "Commentary on Building Code Requirements for Reinforced Concrete (ACI-318-77)", American Concrete Institute, Detroit, 1977.
3. ACI Committee 340, "Design Handbook In Accordance With the Strength Design Method of ACI 318-77 : Volume 2 - Columns", ACI Special Publication, SP 17A(78), American Concrete Institute, Detroit, 1978, p. 214.
4. Bock, A.H., "Slenderness Effects : Notes on ACI 318-77 Building Code Requirements for Reinforced Concrete with Design Applications", Third Edition, Portland Cement Association, Illinois, 1980, part 12.
5. Christenser, K., "Writing Easy-To-Use Computer Program", Engineering Digest, Vol. 29, Number 5, May, 1983, p. 40.
6. Clarke, D., "Computer Aided Structural Design", A Wiley-Interscience Publication, Chichester, 1978.
7. "COLS - Strength Design of Reinforced Concrete Column Section", Canadian Portland Cement Association, Ottawa.
8. CSA Code, "Code for the Design of Concrete Structures for Buildings (CAN3-A23.3-M77)", Canadian Standards Association, Ontario, 1977.

9. CSA Code, "Concrete Materials and Methods of Concrete Construction, Method of Test for Concrete (CSA3-A23.1-M77, CSA3-A23.2-M77)", Canadian Standards Association, Ontario, 1977.
10. Dobbs, J.E., "Design for Flexure and Axial Load : Notes on ACI 318-77 Building Code Requirements for Reinforced Concrete with Design Applications", Third Edition, Portland Cement Association, Illinois, 1980, part 10.
11. Engineering Bulletin, "Strength Design of Reinforced Concrete Columns", EB009.02D, Portland Cement Association, Illinois, 1978.
12. Furlong, R.W., "Design for Flexure and Axial Load : Metric Design Handbook for Reinforced Concrete Elements In Accordance With the Strength Design Methods of CSA Standard CAN3-A23.3-M77", Canadian Portland Cement Association, Ottawa.
13. Ghosh, S.K., "Design Methods and Strength Requirements : Notes on ACI 318-77 Building Code Requirements for Reinforced Concrete with Design Applications", Third Edition, Portland Cement Association, Illinois, 1980, part 4.
14. Gouwens, A.J., "Biaxial Bending Simplified", Reinforced Concrete Columns, Publication SP-50, American Concrete Institute, Detroit, 1975, p. 233.
15. Halvorsen, G.T., "Reinforced Concrete Column Design Program", ACI Seminar, Kansas City, Fall, 1983.

16. MacGregor, J.G. and Breen, J.E. and Pfrang, E.O., "Design of Slender Concrete Columns", ACI Journal, Vol. 67, No. 1, January, 1970, p. 6.
17. MacGregor, J.G., Olehafen, U.H. and Hage, S.E., "A Re-examination of the EI Value for Slender Columns", Reinforced Concrete Columns, Publication SP-50, American Concrete Institute, Detroit, 1975, p. 1.
18. Mockry, E.F. and Darwin, D., "Slender Column Interaction Diagrams", Concrete International, June, 1982, p. 44.
19. Murrill, P.W. and Smith, C.L., "Fortran IV Programming for Engineers and Scientists", Intext Educational Publishers, New York, 1973.
20. Randall, R.W., "Design for Flexure : Notes on ACI 318-77 Building Code Requirements for Reinforced Concrete with Design Applications", Third Edition, Portland Cement Association, Illinois, 1980, part 9.
21. Rice, P.F., "Design for Reinforcement : Metric Design Handbook for Reinforced Concrete Elements In Accordance With the Strength Design Methods of CSA Standard CAN3-A23.3-M77", Canadian Portland Cement Association, Ottawa.
22. Rice, P.F. and Hoffman, E.S., "Structural Design Guide to the ACI Building Code", Second Edition, Van Nostrand Reinhold Company, New York, 1979, p. 276.

23. Rogers, P. and Baltay, M.L., "Reinforced Concrete Design for Buildings", Van Nostrand Reinhold Company, New York, 1973, p. 75.
24. Saatcioglu, M., "General Design Information : Metric Design Handbook for Reinforced Concrete Elements In Accordance With the Strength Design Methods of CSA Standard CAN3-A23.3-M77", Canadian Portland Cement Association, Ottawa.
25. Schultz, D.M., "Biaxial Loading : Notes on ACI 318-77 Building Code Requirements for Reinforced Concrete with Design Applications", Third Edition, Portland Cement Association, Illinois, 1980, part 11.
26. Wang, C.K. and Salmon, C.G., "Reinforced Concrete Design", Third Edition, Harper & Row Publishers, New York, 1979, p. 388.

APPENDIX A

USER'S MANUAL

A.1 Description

(a) Operational Modes

- (i) Batch Mode
- (ii) Interactive Mode

The first entry for each run of program *COLUMN1*, is either 0 for batch mode processing or 1 for interactive mode processing. The mode selected applies to all designs to be considered in that run. For interactive mode processing, the return key or similar key must be pressed after a line of required data information is entered on the terminal keyboard.

(b) Code Specifications

- (i) Canadian Standards Association Specification
CAN3-A23.3-M77
- (ii) American Concrete Institute Specification
ACI 318-77

(c) Units

- (i) S.I. units when using *CSA* code
- (ii) Imperial or U.S. Customary units when using
ACI code.

A.2 Control Commands

For use on terminal (MTS).

For batch mode processing,

```
$RUN COLUMN1 5=DATAFILE 4=*SOURCE* 7=*SINK*  
6=UNSORT 8=MESS 9=SORT
```

For interactive mode processing,

```
$RUN COLUMN1 4=*SOURCE* 7=*SINK*  
6=UNSORT 8=MESS 9=SORT
```

Details on I/O units:

Unit 5=DATAFILE refers to input data records
stored in the file named DATAFILE.

Unit 4=*SOURCE* refers to the input from the terminal.

Unit 6=UNSORT refers to the unsorted output for
the line printer.

Unit 9=SORT refers to the sorted output for the
line printer.

Unit 8=MESS refers to the temporary storage of comments.

Unit 7=*SINK* refers to the output for the terminal.

If Device I/O units are not specified, an error message will be printed from MTS terminal informing the user that the specific device units are undefined.

A.3 Data Input

A description of data input is listed below. The second line of each description is a symbolic representation of the input line indicating the order of the entries.

Input is formatted and each of the entries in the input line is entered in according to its format field specification. When the program is executed using the

Michigan Terminal System (MTS) at the University of Alberta, free format input is permitted only for F format and I format, and individual entries are separated by a comma. All entries in an input line are required but all zero entries occurring at the end of a line may be omitted.

Input is the same for both batch and interactive mode but in interactive mode prompts are displayed for inputs.

(a) Operational Modes :

Mode FORMAT (I1)
 Mode = 0 batch mode
 = 1 interactive mode

(b) Heading Record :

HEADING FORMAT (15A4)
 HEADING = Any user defined descriptive identifier
 not exceeding 60 characters

(c) Input Control Design Record:

(1) CODE TYPE NLC OUT FORMAT (2A2, 2I2)
 CODE = C specifies CAN3-A23.3-M77
 = A specifies ACI 318-77
 TYPE = T indicates tied column
 = S indicates spiral column

NLC = total number of load cases

If NLC not entered, defaults to 1.

In interactive mode, an entry for NLC is not required because the user has the option of later entering as many load cases as desired.

OUT = 0 standard output for design practice

= 1 long output for debugging purposes

(d) Data Material Properties:

(1) FPC FY FSY WC ES FORMAT (4F7.0, F12.0)

FPC = concrete strength (MPa or psi).

If value entered is outside range 20-40 MPa or 2000-6000 psi, design is executed but warning message is printed.

FY = yield stress for longitudinal reinforcement (MPa or psi). If fy is outside range of 300-400 MPa or 40000-60000 psi, design is executed but warning message is printed.

FSY = yield stress for spiral steel (MPa or psi). If fsy is outside range of 300-400 MPa or 40000-60000 psi, design is executed but warning message is printed.

WC = mass density concrete (kg/m^3 or pcf).

If not entered default to 2400 kg/m^3 or 145 pcf.

ES = Modulus of elasticity for reinforcement
(MPa or psi). If not entered
default to 200000 MPa or 29000000 psi.

(e) Data Geometry Properties:

(1) Concrete geometry:

SHAPE Cx Cy R DC CXINCR CYINCR

FORMAT (A2, 2F6.1, F5.2, 3F5.1)

SHAPE = S indicates square cross-section
 = R indicates rectangular cross-section
 = C indicates circular cross-section

Cx = dimension of cross-section in x direction
(mm or in)

Cy = dimension of cross-section in y direction
(mm or in). If shape = S or R, value of
Cy not used.

R = ratio of Cx/Cy. This value is used with either
Cx or Cy, but when both Cx and Cy are entered,
R is not used. When shape is not rectangular,
R defaults to 1.0.

DC = minimum clear concrete cover (mm or in).
If not entered, default value is 40 mm
or 1.5 in..

CXINCR = length increments of dimension Cx value
(mm or in). If not entered, default
value is 50 mm or 2 in..

CYINCR = length increments of dimension Cy value
(mm or in). If not entered, default
value is 50 mm or 2 in..

(2) Reinforcement geometry:

RFACE PG BMAXX NBS BSMIN BSMAX DS NX NY

FORMAT (A2, F7.4, 4I4, I3, 2I2)

RFACE = X reinforcement configuration in
x-face (lateral faces) only (Fig. 3.2.2).
= Y reinforcement configuration in
y-face (end faces) only (Fig. 3.2.3).
= B reinforcement configuration in
both x and y -faces (bars in x-faces
include all corner bars) (Fig. 3.2.4).
= C means reinforcement configuration
around the core diameter (Fig. 3.6.5).

PG = maximum percentage of reinforcement

If PG not entered default to 0.03.

If PG < 0.01 value set to 0.01.

If PG > 0.08 value set to 0.08.

BMAXX = entered maximum permissible number
of bars in cross-section. If not
entered default to the value given
in Table 3.1 in Section 3.5 .

NBS = specified total number of bars

If NBS is entered and bars are in

both faces, NX and NY must be entered.

BSMIN = minimum bar size, 15 to 55 for

CSA or 5 to 18 for ACI.

If not entered default to 15 or 5.

BSMAX = maximum bar size, 15 to 55 for

CSA or 5 to 18 for ACI.

If not entered default to 35 or 11.

DS = minimum lateral reinforcement diameter,

i.e. bar size number is entered.

If not entered default value are

10 for CSA or 3 for ACI

NX = number of bars in each x-face

including corner bars

NY = number of bars in each y-face

(3) Stability information:

BENDING XBRACING KX LUX YBRACING KY LUY SPU SPC

FORMAT (A2, I4, F7.3, F10.0, I4, F7.3, F10.0,
2F8.0)

BENDING = U uni-axial bending

= B bi-axial bending

XBRACING = 0 column unbraced about x-axis

= 1 column braced about x-axis

KX = effective length factor about x-axis

LUX = unsupported length about x-axis (mm or in)

YBRACING = 0 column unbraced about y-axis

= 1 column braced about y-axis

KY = effective length factor about y-axis

LUY = unsupported length about y-axis (mm or in)

SPU = summation of axial design load for all
of the columns in the storey (KN or kips)

SPC = summation of critical load for all
of the columns in the storey (KN or kips)

(f) Data Loads Conditions:

(1) FACT PD PL MDXT MLXT MDXB MLXB

FORMAT (A2, 6F8.0)

FACT = U loading entered is unfactored

= F loading entered is factored

PD = axial dead load (must be positive since
only compression loads permitted)
(KN or kips)

PL = axial live load (must be positive)
(KN or kips)

MDXT = dead load moment about x-axis on
top of column (KN-m or kip-ft)

MLXT = live load moment about x-axis on
top of column (KN-m or kip-ft)

MDXB = dead load moment about x-axis on
bottom of column (KN-m or kip-ft)

MLXB = live load moment about x-axis on
bottom of column (KN-m or kip-ft)

Note : The applied moment about x-axis is primarily resisted by the reinforcement in the y-faces.

Note : A negative sign to one of the entered moments at the top or bottom of column indicates the column deflects in double curvature. When the signs are the same, column deflects in single curvature. Design moment is the maximum absolute value of moments entered at either top or bottom of the column. Same philosophy applies for moments about the y-axis.

(2) DESIGN MDYT MLYT MDYB MLYB

FORMAT (A2, 4F8.0)

DESIGN = D designates a design, that is
 program will select all missing values
 of cross-section to satisfy loading
 = L designates a check for adequacy in
 strength of selected section
 for multiple loading cases
 = C designates a check of entered
 section

MDYT = dead load moment about y-axis on
 top of column (KN-m or kip-ft)

MLYT = live load moment about y-axis on
 top of column (KN-m or kip-ft)

MDYB = dead load moment about y-axis on

bottom of column (KN-m or kip-ft)

MLYB = live load moment about y-axis on

bottom of column (KN-m or kip-ft)

Note : The applied moment about y-axis is primarily
resisted by the reinforcement in the x-faces.

(g) Optional Data:

(1) **OPTION**

In batch mode the option number indicates the line type at which new data will be entered. All following data lines must be repeated. A new heading may also be used. For example if a new design was to be run using same code and material properties but different concrete geometry requirements option 2 would be selected. A new heading and all data beginning with concrete geometry requirements through to new option number are repeated, but data usually entered before the concrete geometry requirement line is omitted and will default to values used for previous problem. Note, in interactive mode only one line is re-entered.

(i) in batch mode:

FORMAT (I2)

OPTION = 1 means a different column design, all
data records are to be re-entered.

= 2 means heading and new data entries
beginning with concrete geometry.

- = 3 means heading and new data entries beginning with reinforcement geometry.
- = 4 means heading and new data entries beginning with stability geometry.
- = 5 means heading and new data entries beginning with load conditions.
- = 6 means last column design has been solved, program is terminated.

(ii) in interactive mode: FORMAT (I1)

Revised above design, enter 1

Go to next design, enter 2

Termination of program, enter 3

If 2 is entered, the option is the same as that in batch, and the next data records required are prompted by the program.

If 1 is entered, the next data record is specified as follows:

REVISE

FORMAT (I1)

Alter material properties only, enter 1

Alter concrete geometry only, enter 2

Alter reinforcement geometry only, enter 3

Alter stability geometry only, enter 4

Alter loading conditions only, enter 5

APPENDIX B

LOGIC FLOW OF COLUMN1

A general flow chart showing the overall connectivity of the subroutines is given in Fig. B.1. Reference is made to specific subroutines in the program, *COLUMN1*, given in Appendix C with general comments as to the nature of the design steps being performed in that subroutine.

The sequence of events coded into the main executive program *COLUMN1* is as follows (Fig. B.1).

- (1) Enter the mode processing parameters into the terminal.
- (2) Set TPROB = 1 as initial value to indicate a new design with no previous data entered.
- (3) Read the heading.
- (4) If TPROB = 1 go to (5).
If TPROB $\neq$ 1 and
concrete geometry properties are
to be varied, go to (9).
reinforcement geometry properties
are to be varied, go to (9).
stability geometry properties are
to be varied, go to (9).
loading geometry properties are
to be varied, go to (11).
- (5) Read the design control parameters (Sub: *INPUT1*).
(If in batch mode, go to (7).)

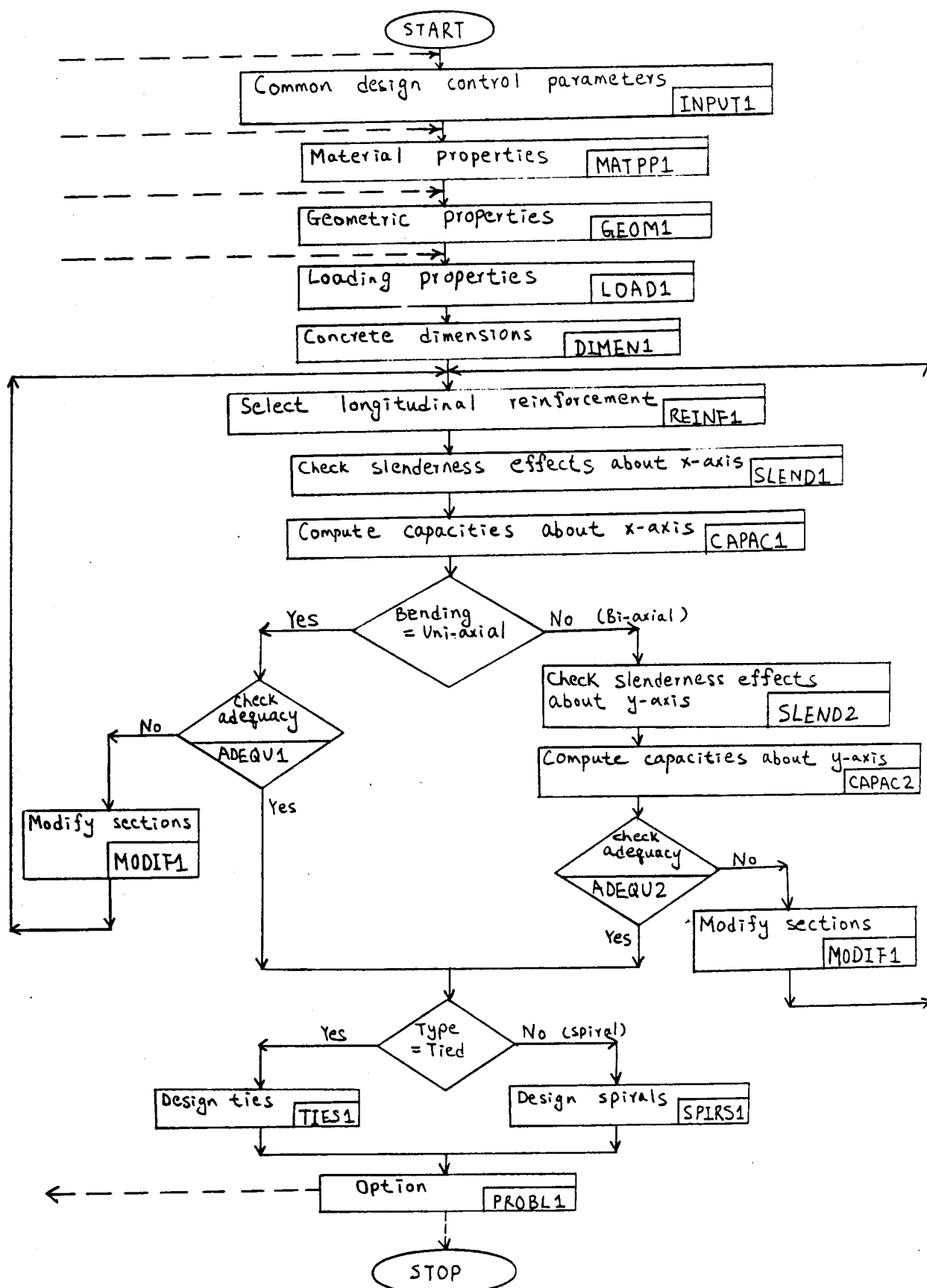


Fig. B.1 A general flow chart of COLUMN1

- (6) Used only if in interactive mode.
If altering material properties only, go to (7).
If altering concrete geometry only,
go to (9).
If altering reinforcement geometry only,
go to (9).
If altering stability geometry only,
go to (9).
If altering loading conditions only, go to (12).
- (7) Read the material properties (Sub: *MATPP1*).
(If in batch mode, go to (9).)
- (8) Used only if in interactive mode,
if the revised design alters material
properties only, go to (13).
- (9) Read the geometry properties (Sub: *GEOM1*).
(If in batch mode, go to (11).)
- (10) Used only if in interactive mode,
if the revised design alters concrete
geometry only, or alters steel geometry
only, or alters stability geometry
only, go to (13).
- (11) Set $N = NLC$ to indicate a new design with no
previous counting of iterations in which
 NLC is total number of load cases which is the
same number for every new set of problem (in
interactive mode, NLC is equal to 1);
 N is a counter do loop during a program run

of each design.

- (12) Read the loading properties (Sub: *LOAD1*).
- (13) If the condition of analysis is to check for adequacy in strength of selected column section for multiple loading cases, dimension default flag TDIMFG will be defaults to 1 for unaltered dimensions, and then go to (18).
- (14) Select the initial concrete dimension (Sub: *DIMEN1*).
- (15) If the entered dimension is less than minimum concrete column dimension, go to (33).
- (16) Select longitudinal reinforcement (Sub: *REINF1*).
- (17) If the reinforcement consistent with the entered restraints could not be obtained, a transfer of control to (33).
- (18) Check slenderness effects about x-axis (Sub: *SLEND1*).
- (19) If data input is incomplete for computing magnification factor, a transfer of control to (34). When new column dimensions are required due to stability problem, return to (16).
- (20) Compute capacities about x-axis (Sub: *CAPAC1*).
- (21) If it is bi-axial bending, go to (23).
- (22) Check capacities for adequacy, then go to (27) (Sub: *ADEQU1*).
- (23) Check slenderness effects about y-axis (Sub: *SLEND2*).
- (24) If data input is incomplete for computing

- magnification factor, a transfer of control to (35).
- (25) Compute capacities about y-axis (Sub: *CAPAC2*).
- (26) Check capacities for adequacy (Sub: *ADEQU2*).
- (27) If the resisting strength is not adequate
for a check mode, go to (33).
If design is adequate, go to (30).
- (28) Otherwise modify trial section if required
(Sub: *MODIF1*).
- (29) If design cannot be completed consistent
with the entered restraints, go to (33).
If a new section is tried, reiterate step (16).
- (30) If it is a tied column, go to (32).
- (31) Select lateral reinforcement for spiral column,
and then go to (33) (Sub: *SPIRS1*).
- (32) Select lateral reinforcement for tied (Sub: *TIES1*).
- (33) If a long output is desired, go to (35).
- (34) Otherwise a desired short output
is printed (Sub: *OUTPUT*).
- (35) If a counter of N (the number of load cases)
is less than or equal to 1, go to (37).
- (36) Otherwise sort the output with the heading
printed at the top of all subsequent pages for the
same design, and all messages information printed
afterwards, then reiterate step (12). (Sub: *SORT*).
- (37) Read the next design control parameter (Sub: *PROBL1*).
If in batch mode :
Enter 1 means a new set of all data records.

Enter 2 means heading varied with following
concrete geometry

Enter 3 means heading varied with following
reinforcement geometry

Enter 4 means heading varied with following
stability geometry

Enter 5 means heading varied with following
loading conditions

Enter 6 means the last column design is solved.

If in interactive mode :

Enter 1 means revising above design in which
a next data record is required as follows

: enter 1 means altering material properties only;

: enter 2 means altering concrete of geometry
only;

: enter 3 means altering reinforcement of
geometry only;

: enter 4 means altering stability of geometry
only;

: enter 5 means altering loading conditions only.

Enter 2 means going to next design in which
the next data record required is the same as
those in batch mode.

Enter 3 means termination of program.

(38) Then sort the output (Sub: *SORT*).

(If not revising above design in
interactive mode, go to (40))

- (39) Used only if revising above design in interactive mode, return to (6).
- (40) If it is not the last column design, return to (3).
- (41) Otherwise, end of design.

B.1 Flags Default

(a) Dimension Default Flags

DIMFLAG (TDIMFG) = 1 means Cx and Cy, or Cx and Cx/Cy,
 or Cy and Cx/Cy is entered
 (i.e. cannot alter either Cx
 or Cy)

= 2 means only Cx is entered
 (i.e. can alter Cy only)

= 3 means only Cy is entered
 (i.e. can alter Cx only)

= 4 means only Cx/Cy is entered
 (i.e. can alter Cx and Cy
 to most closely approximate
 the entered aspect ratio)

= 5 means Cx,Cy and Cx/Cy are unknown
 (i.e. can alter either Cx or Cy)

(b) Reinforcement Default Flag

MODAS (TMODAS) = 0 means column dimensions are
 incremented and process of
 selecting reinforcement repeated

= 1 means bar size is
 incremented and process of
 selecting reinforcement repeated

= 2 means number of bars is
 incremented and process of
 selecting reinforcement repeated

APPENDIX C**COMPUTER CODING**

This appendix presents a source listing of the Main routine and the subroutines of the processing program COLUMN1.

```

1 C .....
2 C COLUMN1
3 C
4 C * A PROGRAM FOR DESIGN AND CHECK OF REINFORCED CONCRETE COLUMNS*
5 C * IN ACCORDING TO CSA AND ACI SPECIFICATION BUILDING CODES
6 C * RUNNING EITHER IN BATCH MODE OR IN INTERACTIVE MODE
7 C * IS CODED BY LEUNG CHI-KUN
8 C * AS PART OF M.SC. THESIS, DEPARTMENT OF CIVIL ENGINEERING,
9 C * UNIVERSITY OF ALBERTA, EDMONTON, ALBERTA, CANADA, FALL, 1983.
10 C .....
11 C
12 C THIS IS THE MAIN ROUTINE OF THE PROGRAM IN WHICH A SEQUENCE OF
13 C EVENTS IS CODED AND ALL CORRESPONDING SUBROUTINES ARE CALLED IF
14 C REQUIRED
15 C
16 C IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y), INTEGER(T,J,N,S)
17 C COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEQ,TPROB
18 C COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TSS(8),DS(8),
19 C * PD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
20 C * TBMAX,SHAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAXX
21 C COMMON PPC,FY,FSY,EC,B1,TBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
22 C * SPC,PD,PL,FU,MX1,MX2,BDX,MV1,MV2,SDY,TOUT,TMODE
23 C COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
24 C COMMON TDMFC,AC,AS,CC,SB,OP,AST,NX,NY,SBX,SDY,DS,
25 C * V(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MXX,
26 C * MAPPL,PUDESK,MUDESK,PKX
27 C COMMON EX,MCV,HAPPY,PUDESY,MUDESY,PSY,MOVREQ,MOKREQ,
28 C * VC,YB,ECS(12),ETS(12),CS(12),FT(12),
29 C * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
30 C COMMON TESTMO
31 C COMMON NLC,JPOB,TREVIS,TDSS,SPACE,PITCH,NSPACE
32 C DIMENSION A(8)
33 C DIMENSION NED(16)
34 C DIMENSION V(6)
35 C
36 C IDENTIFIERS: DEFINE THE VARIABLES IN COMMON BLOCKS
37 C PG=MAX. PERCENTAGE OF STEEL
38 C ESMAX=MAX. STRESS = FY/ES
39 C TYPE=TYPE OF COLUMN : 1 FOR TIED, 2 FOR SPIRAL
40 C TREINF=A TRANSFER OF CONTROL IN SUBROUTINE REINF1, 1 GO TO
41 C SUBROUTINE SLEN1, 2 PROCEED TO NEXT PROBLEM
42 C TSLEN1=A TRANSFER OF CONTROL IN SUBROUTINE SLEN1, 1 RETURN
43 C TO MAIN ROUTINE AND CONTINUE EXECUTION, 2 PROCEED TO
44 C NEXT PROBLEM, 3 BACK TO SUBROUTINE REINF1
45 C TBEND=A TRANSFER OF CONTROL IN SUBROUTINE SLEN2, 1 RETURN
46 C TO MAIN ROUTINE AND CONTINUE EXECUTION, 2 PROCEED TO
47 C NEXT PROBLEM
48 C TADEQ=A TRANSFER OF CONTROL IN SUBROUTINE ADEQ1 AND ADEQ2,
49 C 1 PROCEED TO NEXT PROBLEM, 2 GO TO SUBROUTINE MODIF1,
50 C 3 SECTION IS ADEQUATE, THEN RETURN TO MAIN ROUTINE AND
51 C CONTINUE EXECUTION
52 C TPROB=A TRANSFER OF CONTROL FOR OPTIONAL DATA ENTRY, 1 ENTER
53 C A NEW SET OF ALL DATA, 2 ENTER HEADING WITH FOLLOWING
54 C CONCRETE OF GEOM. PROPERTIES, 3 ENTER HEADING WITH
55 C FOLLOWING REINFORCEMENT OF GEOM. PROPERTIES, 4 ENTER
56 C HEADING WITH FOLLOWING STABILITY OF GEOM. PROPERTIES,
57 C 5 ENTER HEADING WITH FOLLOWING LOADING PROPERTIES,
58 C 6 TERMINATE THE RUN
59 C
60 C TCODE=1 CSA CODE, 2 ACI CODE
61 C WC=MASS DENSITY CONCRETE (KG/CU.M OR PCF)
62 C ES=ELASTICITY MODULUS FOR NON-PRESTRESSED REINFORCEMENT
63 C (MPA OR PSI)
64 C DCC=MINIMUM CLEAR CONCRETE COVER (MM OR IN)
65 C DSS=MINIMUM LATERAL REINFORCEMENT DIAMETER (#)
66 C CXINCR=INCREMENT OF DIMENSION CX (MM OR IN)
67 C CYINCR=INCREMENT OF DIMENSION CY (MM OR IN)
68 C TI=DEFAULT MINIMUM BAR SIZE (#)
69 C TK=DEFAULT MAXIMUM BAR SIZE (#)
70 C TSS(8)=ARRAY DENOTATION OF BAR SIZE (#)
71 C DS(8)=ARRAY DENOTATION OF BAR DIAMETER (MM OR IN)
72 C PD=DEAD LOAD FACTOR
73 C FL=LIVE LOAD FACTOR
74 C ECU=MAX. USABLE STRAIN AT EXTREME CONCRETE COMPRESSION FIBER
75 C =0.003
76 C PMAX=MAX. ALLOWABLE REINFORCEMENT RATIO=0.08
77 C PMIN=MIN. ALLOWABLE REINFORCEMENT RATIO=0.01
78 C SB1=DEFAULT MIN. CLEAR BAR SPACING (MM OR IN)
79 C EMIN=DEFAULT MIN. ECCENTRICITY (MM OR IN)
80 C PHIV=DEFAULT CAPACITY REDUCTION FACTOR
81 C RFAC=DEFAULT STRENGTH REDUCTION FACTOR
82 C DSMIN=DEFAULT MIN. SPIRAL DIAMETER (MM OR IN)
83 C SSMIN=DEFAULT MIN. CENTER-TO-CENTER BARS SPACING OF SPIRAL
84 C (MM OR IN)
85 C SSMAX=DEFAULT MAX. CENTER-TO-CENTER BARS SPACING OF SPIRAL
86 C (MM OR IN)
87 C TBMAX=MAX. PERMISSIBLE NO. OF BARS
88 C SHAX=DEFAULT MAX. CLEAR TIED BAR SPACING (MM OR IN)
89 C J=DENOTATION OF BAR SIZE # OR BAR DIAMETER
90 C PRIME=3.1415927
91 C MINDIM=DEFAULT MIN. DIMENSION IN PRACTICE (MM OR IN)
92 C TESTD1=A TRANSFER OF CONTROL IN SUBROUTINE DIMEN1, 1 RETURN TO
93 C MAIN ROUTINE AND CONTINUE EXECUTION, 2 PROCEED TO NEXT
94 C PROBLEM
95 C LATDS(20)=ARRAY DENOTATION OF LATERAL REINFORCEMENT DIAMETER
96 C (MM OR IN)
97 C MINLAT=DEFAULT MIN. LATERAL REINFORCEMENT DIAMETER (MM OR IN)
98 C TBMAXX=ENTERED MAX. PERMISSIBLE NO. OF BARS
99 C PPC=SPECIFIED CONCRETE STRENGTH (MPA OR PSI)
100 C FV=SPECIFIED YIELD STRESS FOR LONGITUDINAL REINFORCEMENT
101 C (MPA OR PSI)
102 C FSY=SPECIFIED YIELD STRESS FOR SPIRAL STEEL (MPA OR PSI)
103 C EC=ELASTICITY MODULUS FOR CONCRETE (MPA OR PSI)
104 C B1=COEFFICIENT RELATING DEPTH OF EQUIVALENT RECT. STRESS
105 C BLOCK TO DEPTH FROM COMPRESSION FACE TO NEUTRAL AXIS
106 C TBRAC=BRACING ABOUT X-AXIS : 0 FOR UNBRACED, 1 FOR BRACED
107 C K=EFFECTIVE LENGTH FACTOR ABOUT X-AXIS
108 C LUX=UNSUPPORTED LENGTH ABOUT X-AXIS (MM OR IN)
109 C TYBRAC=BRACING ABOUT Y-AXIS
110 C KY=EFF. LENGTH ABOUT Y-AXIS
111 C LUY=UNSUPPORTED LENGTH ABOUT Y-AXIS (MM OR IN)
112 C SPU=SUM OF AXIAL DESIGN LOAD FOR COLUMNS IN STOREY (N OR LB)
113 C SPC=SUM OF CRITICAL DESIGN LOAD FOR COLUMNS IN STOREY (N OR LB)
114 C PD=AXIAL DEAD LOAD (N OR LB)
115 C PL=AXIAL LIVE LOAD (N OR LB)
116 C PU=REQUIRED AXIAL LOAD (N OR LB)
117 C MX1=SMALLER FACTORED END MOMENT ABOUT X-AXIS (N-MM OR LB-IN)
118 C MX2=LARGER FACTORED END MOMENT ABOUT X-AXIS (N-MM OR LB-IN)
119 C BDX=RATIO OF MAX. FACTORED DEAD LOAD MOMENT TO MAX. FACTORED
120 C TOTAL LOAD MOMENT ABOUT X-AXIS

```

```

121 C MY1=SMALLER FACTORED END MOMENT ABOUT Y-AXIS (N-MM OR LB-IN)
122 C MY2=LARGER FACTORED END MOMENT ABOUT Y-AXIS (N-MM OR LB-IN)
123 C SDY=RATIO OF MAX. FACTORED DEAD LOAD MOMENT TO MAX. FACTORED
124 C TOTAL LOAD MOMENT ABOUT Y-AXIS
125 C TOUT=0 FOR STANDARD SHORT OUTPUT, 1 FOR LONG OUTPUT IN DETAILS
126 C TMODE=0 FOR BATCH MODE, 1 FOR INTERACTIVE MODE
127 C TSHAPE=SHAPE OF CROSS-SECTION, 1 FOR SQUARE, 2 FOR RECTANGULAR,
128 C 3 FOR CIRCULAR
129 C CX=DIMENSION OF COLUMN SECTION IN X DIRECTION (MM OR IN)
130 C CY=DIMENSION OF COLUMN SECTION IN Y DIRECTION (MM OR IN)
131 C R=RATIO OF CX/CY
132 C TDESIG=CONDITION OF DESIGN OR ANALYSIS OF PROBLEM, 1 FOR
133 C DESIGN PROBLEM, 2 FOR ADEQUACY CHECK OF SECTIONS FOR
134 C MULTIPLE LOAD CASES, 3 FOR ADEQUACY CHECK OF GIVEN
135 C SECTIONS
136 C TRFACE=REINFORCEMENT PATTERNS, 1 FOR BARS IN X-FACES ONLY,
137 C 2 FOR BARS IN Y-FACES ONLY, 3 FOR BARS IN BOTH
138 C FACES, 4 FOR BARS IN CIRCULAR CORE
139 C NBS=SPECIFIED TOTAL NO. OF BARS
140 C NMAX=COMPUTED TOTAL NO. OF BARS
141 C TDIMFG=DIMENSION DEFAULT FLAG, 1 FOR CX AND CY ARE KNOWN,
142 C 2 FOR CX ONLY IS KNOWN, 3 FOR CY ONLY IS KNOWN,
143 C 4 FOR R IS KNOWN, 5 FOR ALL CX, CY AND R ARE UNKNOWN
144 C AC=GROSS AREA OF CROSS-SECTION (SQ. MM OR SQ. IN)
145 C ASHMIN=AREA OF LONGITUDINAL REINFORCEMENT (SQ. MM OR SQ. IN)
146 C CO=CHARACTERISTIC COEFFICIENT BASED ON SHAPE OF CROSS-SECTION
147 C SB=COMPUTED MIN. CLEAR BAR SPACING (MM OR IN)
148 C DP=DIST. FROM EXTREME COMPRESSION FIBER TO CENTROID OF
149 C TENSION REINFORCEMENT (MM OR IN)
150 C AST=TOTAL AREA OF LONGITUDINAL REINFORCEMENT (SQ. MM OR SQ. IN)
151 C NX=NO. OF BARS IN LATERAL FACES INCLUDING CORNER BARS
152 C NY=NO. OF BARS IN END FACES
153 C SBX=ACTUAL CLEAR BAR SPACING MEASURED IN LATERAL FACES
154 C (MM OR IN)
155 C SBY=ACTUAL CLEAR BAR SPACING MEASURED IN END FACES (MM OR IN)
156 C DS=DIST. FROM CENTROID OF EXTREME COMPRESSION REINFORCEMENT
157 C TO CENTROID OF EXTREME TENSION REINFORCEMENT IN OPPOSITE
158 C HALF OF MEMBER (MM OR IN)
159 C Y(12)=ARRAY DENOTATION OF BARS DISTANCE FROM X-AXIS (MM OR IN)
160 C ISEX=MOMENT OF INERTIA OF REINFORCEMENT ABOUT X-AXIS
161 C (MM4 OR IN4)
162 C X(12)=ARRAY DENOTATION OF BARS DISTANCE FROM Y-AXIS (MM OR IN)
163 C ISEY=MOMENT OF INERTIA OF REINFORCEMENT ABOUT Y-AXIS
164 C (MM4 OR IN4)
165 C THODAS=REINFORCEMENT DEFAULT FLAG, 0 MEANS NUMBER OF BARS
166 C IS INCREMENTED AND THE PROCESS OF SELECTING
167 C REINFORCEMENT REPEATED, 1 MEANS COLUMN DIMENSIONS
168 C ARE INCREMENTED AND THE PROCESS OF SELECTING
169 C REINFORCEMENT REPEATED, 2 MEANS BAR SIZE IS
170 C INCREASED TOGETHER WITH USING MAXIMUM NUMBER OF
171 C BARS
172 C PGDES=DESIGN REINFORCEMENT RATIO
173 C EY=ECCENTRICITY ABOUT X-AXIS (MM OR IN)
174 C MCX=MAGNIFIED FACTORED MOMENT ABOUT X-AXIS (N-MM OR LB-IN)
175 C MAPPX=APPLIED MOMENT ABOUT X-AXIS (N-MM OR LB-IN)
176 C PUXSX=DESIGN AXIAL LOAD ABOUT X-AXIS (N OR LB)
177 C MUXSX=DESIGN MOMENT ABOUT X-AXIS (N-MM OR LB-IN)
178 C PBX=DESIGN AXIAL LOAD AT BALANCED CONDITION ABOUT X-AXIS
179 C (N OR LB)
180 C EX=ECCENTRICITY ABOUT Y-AXIS (MM OR IN)
181 C MCY=MAGNIFIED FACTORED MOMENT ABOUT Y-AXIS (N-MM OR LB-IN)
182 C MAPPY=APPLIED MOMENT ABOUT Y-AXIS (N-MM OR LB-IN)
183 C PUYSY=DESIGN AXIAL LOAD ABOUT Y-AXIS (N OR LB)
184 C MUXSY=DESIGN MOMENT ABOUT Y-AXIS (N-MM OR LB-IN)
185 C PSY=DESIGN AXIAL LOAD AT BALANCED CONDITION ABOUT Y-AXIS
186 C (N OR LB)
187 C MDYREQ=REQUIRED EQUIVALENT UNIAXIAL MOMENT ABOUT X-AXIS
188 C (N-MM OR LB-IN)
189 C MDXREQ=REQUIRED EQUIVALENT UNIAXIAL MOMENT ABOUT Y-AXIS
190 C (N-MM OR LB-IN)
191 C YC=DISTANCE FROM EXTREME COMPRESSION FIBER TO NEUTRAL AXIS
192 C PARALLEL TO X-AXIS (MM OR IN)
193 C YB=DIST. FROM EXTREME COMPRESSION FIBER TO NEUTRAL AXIS
194 C PARALLEL TO X-AXIS AT BALANCED CONDITION (MM OR IN)
195 C ECS(12)=ARRAY DENOTATION OF STRAIN IN COMPRESSION STEEL
196 C ABOUT X-AXIS
197 C ETS(12)=ARRAY DENOTATION OF STRAIN IN TENSION STEEL
198 C ABOUT X-AXIS
199 C CS(12)=ARRAY DENOTATION OF FORCE IN COMPRESSION STEEL
200 C ABOUT X-AXIS (N OR LB)
201 C FT(12)=ARRAY DENOTATION OF FORCE IN TENSION STEEL
202 C ABOUT X-AXIS (N OR LB)
203 C XC=DIST. FROM EXTREME COMPRESSION FIBER TO NEUTRAL AXIS
204 C PARALLEL TO Y-AXIS (MM OR IN)
205 C XB=DIST. FROM EXTREME COMPRESSION FIBER TO NEUTRAL AXIS
206 C PARALLEL TO Y-AXIS AT BALANCED CONDITION (MM OR IN)
207 C ECSY(12)=ARRAY DENOTATION OF STRAIN IN COMPRESSION STEEL
208 C ABOUT Y-AXIS
209 C ETSY(12)=ARRAY DENOTATION OF STRAIN IN TENSION STEEL
210 C ABOUT Y-AXIS
211 C CSY(12)=ARRAY DENOTATION OF FORCE IN COMPRESSION STEEL
212 C ABOUT Y-AXIS (N OR LB)
213 C FTY(12)=ARRAY DENOTATION OF FORCE IN TENSION STEEL
214 C ABOUT Y-AXIS (N OR LB)
215 C TESTMO=A TRANSFER OF CONTROL IN SUBROUTINE MODIFI, 1 BACK TO
216 C SUBROUTINE REINF1, 2 PROCEED TO NEXT PROBLEM
217 C NLC=TOTAL NO. OF LOAD CASES
218 C JPOB=ONLY USED FOR INTERACTIVE MODE, 1 FOR REVISED PROBLEM,
219 C 2 FOR NEXT PROBLEM, 3 FOR TERMINATION OF PROGRAM
220 C TREVIS=ONLY USED IN INTERACTIVE MODE FOR JPOB=1, 1 ONLY FOR
221 C ALTERING MATERIAL PROPERTIES, 2 ONLY FOR ALTERING
222 C CONCRETE OF GEOM., 3 ONLY FOR ALTERING REINFORCEMENT
223 C OF GEOM., 4 ONLY FOR ALTERING STABILITY OF GEOM., 5
224 C ONLY FOR ALTERING LOADING PROPERTIES
225 C TDSST=LATERAL REINFORCEMENT USED (/)
226 C SPACE=ACTUAL SPACING REQUIRED IN TIES (MM OR IN)
227 C PITCH=ACTUAL PITCH REQUIRED IN SPIRALS (MM OR IN)
228 C NSPACR=NO. OF SPACERS REQUIRED IN SPIRALS
229 C
230 C A(9)=ARRAY DENOTATION OF AREA OF BAR SIZE (SQ. MM OR SQ. IN)
231 C HED=HEADING DESCRIBES THE AIM OF THIS RUN OF THIS PROGRAM
232 C
233 C INITIALIZE TPROB FOR THE FIRST RUN
234 C
235 C TPROB=1
236 C WRITE(7,805)
237 C 805 FORMAT('S MODE=? TYPE 0 BATCH, OR TYPE 1 INTERACTIVE.'/,
238 C ' /' THEN PRESS RETURN KEY'/)
239 C READ(4,806) TMODE
240 C 806 FORMAT(11)

```

```

241      111 IF(TMODE.EQ.1) GO TO 900
242      READ(5,101) NED
243      GO TO 901
244      900 WRITE(7,800)
245      800 FORMAT('2 HEADING= DESCRIPTIVE IDENTIFIER NOT EXCEEDING',
246      * / ' 50 CHARACTERS'/)
247      READ(4,101) NED
248      901 WRITE(5,200)NED
249      200 FORMAT('1',1X,15A4,'PAGE 1'/1X,50(' ')/)
250      CALL TIME(6,-1,V)
251      WRITE(5,1000)V
252      1000 FORMAT(33X,2A4,5X,3A4/)
253      GO TO (51,52,52,52,999,55),TPROB
254
255 C
256 C READ PROBLEM CONTROL PARAMETERS
257 C
258      51 CALL INPUT1
259      IF(TCODE.EQ.1.OR.TCODE.EQ.2) GO TO 701
260      WRITE(5,700)
261      N=1
262      GO TO 72
263      701 IF(TYPE.EQ.1.OR.TYPE.EQ.2) GO TO 2
264      WRITE(5,702)
265      N=1
266      GO TO 72
267
268 C
269 C ONLY USED FOR INTERACTIVE MODE
270 C
271      2 IF(TREVIS.EQ.1) GO TO 51
272      IF(TREVIS.EQ.2) GO TO 52
273      IF(TREVIS.EQ.3) GO TO 52
274      IF(TREVIS.EQ.4) GO TO 52
275      IF(TREVIS.EQ.5) GO TO 55
276
277 C
278 C READ MATERIAL PROPERTIES OF SECTION
279 C
280      51 CALL MATPP1
281      IF(TMODE.EQ.1) GO TO 10
282      IF(PPC.LE.0.) GO TO 704
283      IF(PV.LE.0.) GO TO 704
284      IF(PSV.LE.0.) GO TO 704
285      GO TO 10
286      704 N=1
287      GO TO 72
288
289 C
290 C ONLY USED FOR INTERACTIVE MODE
291 C
292      10 IF(TREVIS.EQ.1) GO TO 52
293
294 C
295 C READ GEOMETRY PROPERTIES OF CONCRETE, REINFORCEMENT AND STABILITY
296 C
297      52 CALL GEOM1
298      IF(TSHAPE.EQ.0) GO TO 710
299      IF(TRFACE.EQ.0) GO TO 710
300      IF(TBEND.EQ.0) GO TO 710
301      GO TO 711
302      710 N=1
303      GO TO 72
304
305 C
306 C ONLY USED FOR INTERACTIVE MODE
307 C
308      711 IF(TREVIS.EQ.2)GO TO 52
309      IF(TREVIS.EQ.3) GO TO 52
310      IF(TREVIS.EQ.4) GO TO 52
311
312 C
313 C N IS A COUNTER DD LOOP. NLC IS KEEP AS AN INITIAL VALUE FOR
314 C THE STARTING OF EACH RUN
315 C
316      999 N=NLC
317
318 C
319 C READ LOADING PROPERTIES
320 C
321      55 CALL LOAD1
322      IF(TDESIG.EQ.0) GO TO 715
323      GO TO 52
324      715 N=1
325      GO TO 72
326      52 IF(TDESIG.EQ.2) GO TO 20
327
328 C
329 C SELECT INITIAL CONCRETE DIMENSION
330 C
331      CALL DIMEN1
332      GO TO(6,14),TESTDI
333
334 C
335 C SELECT REINFORCEMENT
336 C
337      6 CALL REINF1
338      GO TO (17,14),TREINF
339
340 C
341 C IF TDESIG=2, SET YDIMFG=1 FOR KNOWN DIMENSIONS
342 C
343      20 YDIMFG=1
344
345 C
346 C CHECK SLENDERNESS EFFECTS ABOUT X-AXIS
347 C
348      17 CALL SLEND1
349      GO TO (27,14,5),TSLEN1
350
351 C
352 C COMPUTE CAPACITIES ABOUT X-AXIS
353 C
354      27 CALL CAPAC1
355      IF(TBEND.EQ.2) GO TO 5
356
357 C
358 C CHECK CAPACITIES FOR ADEQUACY IN UNIAXIAL BENDING
359 C
360      CALL ADEQU1
361      GO TO 5
362
363 C
364 C CHECK SLENDERNESS EFFECT ABOUT Y-AXIS
365 C
366      5 CALL SLEND2
367      GO TO (22,14),TSLEN2
368
369 C
370 C COMPUTE CAPACITIES ABOUT Y-AXIS
371 C
372      22 CALL CAPAC2
373
374 C
375 C CHECK CAPACITIES FOR ADEQUACY IN BIAxIAL BENDING
376 C
377      CALL ADEQU2

```

```

361      9      GO TO (14,16,11),TADCO
362      C
363      C MODIFY SECTIONS IF REQUIRED
364      C
365      16      CALL MODIF1
366      GO TO (6,14),TESTMO
367      11      IF (TYPE.EQ.1) GO TO 12
368      C
369      C SELECT LATERAL REINFORCEMENT FOR SPIRAL COLUMN
370      C
371      CALL SPIRS1
372      GO TO 14
373      C
374      C SELECT LATERAL REINFORCEMENT FOR TIED
375      C
376      12      CALL TIES1
377      C
378      14      IF(TOUT.EQ.1) GO TO 71
379      C
380      C A STANDARD SHORT OUTPUT IS REQUIRED
381      C
382      CALL OUTPUT
383      C
384      C IF NOT MULTIPLE PROBLEMS OR N=1, GO TO 72
385      C
386      71      IF(N.LE.1) GO TO 72
387      C
388      C SORT THE OUTPUT IN FORMATTED TYPE
389      C
390      CALL SORT(NED,N,JPOS)
391      GO TO 55
392      C
393      C READ THE NEXT PROBLEM CONTROL PARAMETERS
394      C
395      72      CALL PROBL1
396      CALL SORT(NED,N,JPOS)
397      IF(JPOS.EQ.1) GO TO 2
398      IF(TPROB.EQ.1) GO TO 703
399      IF(TCODE.EQ.1) GO TO 1200
400      IF(TCODE.EQ.2) GO TO 1200
401      GO TO 55
402      1200 IF(TYPE.EQ.1) GO TO 1201
403      IF(TYPE.EQ.2) GO TO 1201
404      GO TO 55
405      1201 IF(FPC.LE.0.) GO TO 56
406      IF(FY.LE.0.) GO TO 56
407      IF(FSY.LE.0.) GO TO 56
408      703 IF(TPROB.EQ.1.OR.TPROB.EQ.2) GO TO 712
409      IF(TSHAPE.EQ.0) GO TO 56
410      712 IF(TPROB.LE.3) GO TO 713
411      IF(TRFACE.EQ.0) GO TO 56
412      713 IF(TPROB.LE.4) GO TO 714
413      IF(TBEND.EQ.0) GO TO 56
414      714 IF(TPROB.EQ.5) GO TO 717
415      GO TO 716
416      717 IF(TDESIG.EQ.0) GO TO 56
417      718 GO TO (111,111,111,111,111,56),TPROB
418      C
419      55      STOP
420      C
421      101      FORMAT(15A4)
422      700      FORMAT(' ',5X,'INVALID BUILDING CODE ENTERED'/)
423      702      FORMAT(' ',5X,'INVALID COLUMN TYPE ENTERED'/)
424      END
425      C
426      C *****
427      C "
428      SUBROUTINE INPUT1
429      C "
430      C "THIS SUBROUTINE READS INPUT CONTROL PARAMETERS AND
431      C "SETS CODE DEFAULT
432      C "
433      C *****
434      C
435      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
436      COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADCO,TPROB
437      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CVINCR,T1,TK,TDS(9),DS(9),
438      * PD,FL,ECU,PMAX,PMIN,SSI,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
439      * TBMX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAX,
440      * SPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BDY,TOUT,TMODE
441      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,MMAX
442      COMMON TDIMFG,AC,AS,CG,EB,DP,AST,NK,NY,SBX,SBY,DS,
443      * V(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MCX,
444      * MAPPX,PUDEX,NUDEX,PKX
445      COMMON EX,MCY,HAPPY,PUDESY,NUDESY,PBY,MOYREQ,MOXREQ,
446      * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
447      * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
448      COMMON TESTMO
449      COMMON NLC,JPOS,TREVIS,TDSIS,SPACE,PITCH,NSPACR
450      C
451      C
452      DIMENSION A(9)
453      DATA NCODE1,NCODE2/'C','A'/,NTYPE1,NTYPE2/'T','S'/
454      C
455      IF(TMODE.EQ.1) GO TO 800
456      READ (5,101) JCODE,JTYPE,NLC,TOUT
457      IF(NLC.LE.0) NLC=1
458      GO TO 801
459      800      WRITE(7,802)
460      802      FORMAT('4 CODE=? TYPE=? OUTPUT LAYOUT=? (2A2,12)'/,
461      * // CODE: TYPE C FOR CAN3-A23.3-M77',
462      * // A FOR AC1 316-77',
463      * // TYPE: TYPE T FOR TIED',
464      * // S FOR SPIRAL',
465      * // OUTPUT: TYPE O FOR SHORT',
466      * // I FOR LONG'/)
467      READ(4,803) JCODE,JTYPE,TOUT
468      C
469      C ONLY USED FOR INTERACTIVE MODE, ONE LOAD CASE IS REQUIRED
470      C
471      NLC=1
472      803      FORMAT(2A2,12)
473      C
474      C CONVERT ALPHABET TO INTEGER
475      C
476      901      IF(JCODE.EQ.NCODE1) GO TO 800
477      IF(JCODE.EQ.NCODE2) GO TO 700
478      IF(TMODE.EQ.1) GO TO 701
479      TCODE=0
480      RETURN

```

```

481 701 WRITE(7,702)
482 702 FORMAT('S INVALID BUILDING CODE ENTERED'//,
483 'S ENTER CODE: C FOR CSA',
484 'S ENTER CODE: A FOR ACI'//)
485 READ(4,703) JCODE
486 703 FORMAT(A2)
487 GO TO 801
488 700 TCODE=2
489 GO TO 801
490 800 TCODE=1
491 801 IF(JTYPE.EQ.NTYPE1) GO TO 802
492 IF(JTYPE.EQ.NTYPE2) GO TO 704
493 IF(TMODE.EQ.1) GO TO 705
494 TYPE=0
495 RETURN
496 705 WRITE(7,706)
497 706 FORMAT('S INVALID COLUMN TYPE ENTERED'//,
498 'S ENTER TYPE: T FOR TIED',
499 'S FOR SPIRAL'//)
500 READ(4,703) JTYPE
501 GO TO 801
502 704 TYPE=2
503 GO TO 803
504 802 TYPE=1
505 803 IF(TOUT.NE.1) GO TO 85
506 GO TO (87,88), TYPE
507 87 WRITE(8,312)
508 GO TO 88
509 88 WRITE(8,313)
510 GO TO(80,81),TCODE
511 80 WRITE(8,310)
512 GO TO 85
513 81 WRITE(8,311)
514 C
515 C SET CODE DEFAULTS
516 C
517 85 IF(TCODE.EQ.1) GO TO 10
518 WC=145
519 ES=2000000.
520 DCC=1.5
521 CXINCR=2.
522 CYINCR=2.
523 C
524 C BAR DESIGNATION SHOWN IN ARRAY FORM IN INCREMENT OF 1
525 C
526 TBS(1)=5
527 TBS(2)=6
528 TBS(3)=7
529 TBS(4)=8
530 TBS(5)=9
531 TBS(6)=10
532 TBS(7)=11
533 TBS(8)=14
534 TBS(9)=18
535 DB(1)=0.525
536 DB(2)=0.750
537 DB(3)=0.875
538 DB(4)=1.000
539 DB(5)=1.125
540 DB(6)=1.270
541 DB(7)=1.410
542 DB(8)=1.593
543 DB(9)=2.257
544 MINLAT=0.375
545 LATDS(3)=0.375
546 LATDS(4)=0.500
547 LATDS(5)=0.625
548 LATDS(6)=0.750
549 LATDS(7)=0.875
550 LATDS(8)=1.000
551 LATDS(9)=1.125
552 LATDS(10)=1.270
553 LATDS(11)=1.410
554 LATDS(14)=1.593
555 LATDS(18)=2.257
556 SB1=1.5
557 EMIN=1.0
558 MINDIM=8.
559 IF(TYPE.EQ.1) GO TO 11
560 PHIV=0.75
561 RFAC=0.65
562 DSMIN=0.5
563 SSMIN=1.0
564 SSMAX=3.0
565 GO TO 12
566 11 PHIV=0.70
567 RFAC=0.60
568 SMAX=5.0
569 GO TO 12
570 10 WC=2400.
571 ES=200000.
572 DCC=40.0
573 CXINCR=50.
574 CYINCR=50.
575 TBS(1)=15
576 TBS(2)=20
577 TBS(3)=25
578 TBS(4)=30
579 TBS(5)=35
580 TBS(6)=45
581 TBS(7)=55
582 DB(1)=16.0
583 DB(2)=18.5
584 DB(3)=25.2
585 DB(4)=29.9
586 DB(5)=35.7
587 DB(6)=43.7
588 DB(7)=55.4
589 MINLAT=11.3
590 LATDS(6)=6.0
591 LATDS(10)=11.3
592 LATDS(15)=16.0
593 LATDS(20)=19.5
594 LATDS(25)=25.2
595 LATDS(30)=29.9
596 LATDS(35)=35.7
597 LATDS(45)=43.7
598 LATDS(55)=55.4
599 SB1=40.
600 EMIN=25.

```

```

601      MINDIM=200.
602      IF (TYPE.EQ.1) GO TO 13
603      PHIV=0.75
604      RFAC=0.65
605      DSMIN=6.0
606      SSMIN=25.0
607      SSMAX=75.0
608      GO TO 12
609      13 PHIV=0.70
610      RFAC=0.60
611      SMAX=150.0
612      12 ECU=0.003
613      PMAX=0.08
614      PMIN=0.01
615      PRIME=3.1415927
616      C
617      RETURN
618      C
619      C *****FORMAT STATEMENTS*****
620      C
621      101 FORMAT(2A2,2I2)
622      310 FORMAT(' ','DESIGN CODE = CSA A23.3-M77'//)
623      311 FORMAT(' ','DESIGN CODE = ACI 318-77'//)
624      312 FORMAT(' ','TYPE = TIED'//)
625      313 FORMAT(' ','TYPE = SPIRAL'//)
626      C
627      END
628      C
629      C *****
630      C
631      SUBROUTINE MATPPI
632      C
633      C *THIS SUBROUTINE READS MATERIAL PROPERTIES, ECHO CHECKS
634      C *THE INPUT DATA AND EVALUATES CODE DEPENDENT VALUES
635      C
636      C *****
637      C
638      IMPLICIT REAL(A-H,I-K,M,P,R,S,W-Y),INTEGER(T,J,N,S)
639      COMMON PG,ESMAX,TYPE,TREINF,TSENI,TSEND,TSENI2,TADEO,TPROB
640      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9),
641      * PD,PL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
642      * TSMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATOS(20),MINLAT,TSMAXK
643      COMMON FPC,FY,FSY,EC,B1,TBRAC,KK,LUX,TYBRAC,KY,LUY,SPU,
644      * SPC,PD,PL,PU,NX1,MX2,BDX,MX1,MX2,BDY,TOUT,TMODE
645      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
646      COMMON TOIMFG,AG,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
647      * Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MXX,
648      * MAPPX,PUDESX,MUDESK,PBX
649      COMMON EX,MCY,MAPPY,PUDESX,MUDESK,PHY,MOYREO,MOXREO,
650      * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
651      * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
652      COMMON TESTMO
653      COMMON NLC,JPOB,TREVIS,TDESS,SPACE,PITCH,NSPACR
654      DIMENSION A(8)
655      C
656      IF (TMODE.EQ.1) GO TO 900
657      READ (6,100) FPC,FY,FSY,WCC,ESS
658      GO TO 901
659      900 WRITE(7,800)
660      800 FORMAT('8 FPC= ? FY= ? FSY= ? WCC= ? ESS= ?'//,
661      * ' ( 4F7.0,F12.0)'//,
662      * ' CONCRETE STR. : (MPA OR PSI)'//,
663      * ' LONG REINF. YIELD STRESS : (MPA OR PSI)'//,
664      * ' SPIRAL STEEL YIELD STRESS : (MPA OR PSI)'//,
665      * ' CONCRETE DENSITY : 0 DEFAULT TO 2400KG/CU.M OR 145 PCF',
666      * ' MODULUS OF ELAST. REINF. : 0 DEFAULT TO 200000 MPA OR',
667      * ' 28000000 PSI'//)
668      READ (4,100) FPC,FY,FSY,WCC,ESS
669      901 IF (TOUT.NE.1) GO TO 1
670      WRITE(6,101)
671      WRITE(6,200) FPC,WCC,FY,ESS,FSY
672      C
673      C EVALUATE CODE DEPENDENT VALUES AND THE ENTERED MATERIAL
674      C STRENGTH VALUES
675      C
676      1 F=0.65
677      S=0.65
678      C
679      C IF WC AND ESS ARE NOT ENTERED, THEIR VALUES WILL BE ASSIGNED
680      C BY DEFAULT
681      C
682      IF (WCC.GT.0.1) WC=WCC
683      IF (ESS.GT.0.1) ES=ESS
684      705 IF (FPC.LE.0.) GO TO 310
685      IF (FY.LE.0.) GO TO 311
686      IF (FSY.LE.0.) GO TO 312
687      IF (TCODE.EQ.1) GO TO 10
688      IF (FPC.GE.2000..AND.FPC.LE.8000.) GO TO 15
689      IF (TMODE.EQ.1) WRITE(7,210)
690      IF (TOUT.NE.1) GO TO 15
691      WRITE(6,210)
692      IF (FY.GE.40000..AND.FY.LE.60000.) GO TO 11
693      IF (TMODE.EQ.1) WRITE(7,211)
694      IF (TOUT.NE.1) GO TO 11
695      WRITE(6,211)
696      IF (FSY.GE.40000..AND.FSY.LE.60000.) GO TO 12
697      IF (TMODE.EQ.1) WRITE(7,212)
698      IF (TOUT.NE.1) GO TO 12
699      WRITE(6,212)
700      EC=WC*.15+.33*.SORT(FPC)
701      C=0.65-0.00005*(FPC-4000.)
702      D=AMAX1(C,F)
703      B1=AMIN1(D,S)
704      GO TO 19
705      10 IF (FPC.GE.20..AND.FPC.LE.40.) GO TO 16
706      IF (TMODE.EQ.1) WRITE(7,210)
707      IF (TOUT.NE.1) GO TO 16
708      WRITE(6,210)
709      16 IF (FY.GE.300..AND.FY.LE.400.) GO TO 13
710      IF (TMODE.EQ.1) WRITE(7,211)
711      IF (TOUT.NE.1) GO TO 13
712      WRITE(6,211)
713      13 IF (FSY.GE.300..AND.FSY.LE.400.) GO TO 14
714      IF (TMODE.EQ.1) WRITE(7,212)
715      IF (TOUT.NE.1) GO TO 14
716      WRITE(6,212)
717      14 EC=WC*.15+.043*.SORT(FPC)
718      C=0.65-0.006*(FPC-30.)
719      D=AMAX1(C,F)
720      B1=AMIN1(D,S)

```

```

721 19 ESMAX=FV/ES
722 IF(TOUT.NE.1) RETURN
723 WRITE(6,400)
724 WRITE(6,300)FPC,WC,EC,FV,ES,FSV
725 RETURN
726 310 IF(TMODE.EQ.1) GO TO 700
727 WRITE(6,220)
728 RETURN
729 311 IF(TMODE.EQ.1) GO TO 701
730 WRITE(6,221)
731 RETURN
732 312 IF(TMODE.EQ.1) GO TO 702
733 WRITE(6,222)
734 RETURN
735 700 WRITE(7,703)
736 703 FORMAT(' & INVALID CONCRETE STRENGTH ENTERED '//
737 * '// ENTER CONCRETE STR. : (MPA OR PSI) (F7.0)')
738 READ(4,704) FPC
739 704 FORMAT(F7.0)
740 GO TO 705
741 701 WRITE(7,706)
742 706 FORMAT(' & INVALID LONGITUDINAL YIELD STRESS ENTERED '//
743 * '// ENTER LONG. YIELD STRESS : (MPA OR PSI) (F7.0)')
744 READ(4,704) FV
745 GO TO 705
746 702 WRITE(7,707)
747 707 FORMAT(' & INVALID SPIRAL STEEL YIELD STRESS '//
748 * '// ENTER SPIRAL STEEL YIELD STRESS : (MPA OR PSI) (F7.0)')
749 READ(4,704) FSV
750 GO TO 705
751 C
752 C *****FORMAT STATEMENTS*****
753 C
754 100 FORMAT(4F7.0,F12.0)
755 200 FORMAT(' ',SX,'CONCRETE : COMP. STRENGTH =',F10.0,
756 * 'K/PSI/MPA',
757 * '/SX', DENSITY =',F10.0,1X,'PCF',
758 * '/KG/CUBIC METER',
759 * '/SX,'LONG. STEEL : YIELD STRESS =',F10.0,1X,'PSI/MPA',
760 * '/SX', MODULUS OF ELAS. =',F15.0,1X,'PSI/MPA',
761 * '/SX,'SPIRAL : YIELD STRESS =',F10.0,1X,'PSI/MPA')
762 300 FORMAT(' ',SX,'CONCRETE : COMP. STRENGTH =',F10.0,1X,
763 * 'PSI/MPA',
764 * '/SX', DENSITY =',F10.0,1X,'PSI/',
765 * '/KG/CUBIC METER',
766 * '/SX', MODULUS OF ELAS. =',F15.0,1X,'PSI/MPA',
767 * '/SX,'LONG. STEEL : YIELD STRESS =',F10.0,1X,'PSI/MPA',
768 * '/SX', MODULUS OF ELAS. =',F15.0,1X,'PSI/MPA',
769 * '/SX,'SPIRAL : YIELD STRESS =',F10.0,1X,'PSI/MPA')
770 101 FORMAT(' ',THE FOLLOWING MATERIAL DATA WAS ENTERED : '//)
771 400 FORMAT(' ',THE FOLLOWING MATERIAL DATA',
772 * ' WAS ASSIGNED BY DEFAULT : '//)
773 210 FORMAT(' ',SX,'THE ENTERED CONCRETE STRENGTH IS NOT',
774 * ' WITHIN THE',SX,'PREDETERMINED RANGE')
775 211 FORMAT(' ',SX,'THE ENTERED LONGITUDINAL YIELD STRESS',
776 * ' IS NOT WITHIN THE',SX,'PREDETERMINED RANGE')
777 212 FORMAT(' ',SX,'THE ENTERED SPIRAL YIELD STRESS IS NOT',
778 * ' WITHIN THE',SX,'PREDETERMINED RANGE')
779 220 FORMAT(' ',COMMENTS : '//SX,'THE ENTERED CONCRETE STRENGTH',
780 * ' IS INVALID')
781 221 FORMAT(' ',COMMENTS : '//SX,
782 * 'THE ENTERED LONGITUDINAL YIELD STRESS',
783 * ' IS INVALID')
784 222 FORMAT(' ',COMMENTS : '//SX,
785 * 'THE ENTERED SPIRAL YIELD STRESS IS',
786 * ' INVALID')
787 END
788 C
789 C *****
790 C
791 C SUBROUTINE GEOM1
792 C
793 C *THIS SUBROUTINE READS SECTION PROPERTIES, REINFORCEMENT
794 C *AND STABILITY INFORMATION, EVALUATES SOME OF THEIR DATA
795 C *PARAMETERS
796 C
797 C *****
798 C
799 C IMPLICIT REAL(A-H,I,K-M,P,R,S,W-V),INTEGER(T,J,N,S)
800 COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TSLEN2,TADEQ,TPROB
801 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9),
802 * PD,PL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RPAC,DSMIN,SSMIN,SSMAX,
803 * TBMAX,SMAX,J,PRIME,MINDIM,TESTD1,LATOS(20),MINLAT,TBMAXX
804 COMMON PPC,FV,FSV,EC,B1,TXBRAC,KK,LUX,TYBRAC,KY,LUY,SPU,
805 * SPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BDY,TOUT,TMODE
806 COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
807 COMMON TDIMFG,AG,AS,CG,SB,OP,AST,NX,NY,SBX,SBY,DS,
808 * Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MCX,
809 * MAPPX,PUDEX,MUDEX,PBX
810 COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PSY,MOYREQ,MOXREQ,
811 * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
812 * XC,XB,ECSY(12),ETSY(12),CSY(12),PTY(12)
813 COMMON TESTMD
814 COMMON NLC,JPOB,TREVIS,TDSS,SPACE,PITCH,NSPACR
815 DIMENSION A(9)
816 DATA NSHAP1,NSHAP2,NSHAP3/'S','R','C'/
817 DATA NFACE1,NFACE2,NFACE3,NFACE4/'X','Y','B','C'/
818 DATA NSEND1,NSEND2/'U','B'/
819 C
820 C USED ONLY FOR REVISING GEOM. PROPERTIES IN INTERACTIVE MODE
821 C
822 IF(TREVIS.EQ.4) GO TO 120
823 IF(TREVIS.EQ.3) GO TO 111
824 IF(TREVIS.EQ.2) GO TO 750
825 C
826 C USED FOR MULTIPLE PROBLEMS
827 C
828 IF(TPROB.EQ.3) GO TO 111
829 IF(TPROB.EQ.4) GO TO 120
830 750 IF(TMODE.EQ.1) GO TO 800
831 READ(5,100) JSHAPE,CX,CY,R,DCCC,CXINC,CYINC
832 GO TO 801
833 800 WRITE(7,800)
834 800 FORMAT(' & SHAPE=? CX=? CY=? R=? DCC=? CXINC=? CYINC=? '//
835 * ' ( A2,2F5.1,F5.2,3F5.1) ',
836 * '// SHAPE : S FOR SQUARE, R FOR RECT., C FOR CIRCULAR',
837 * '// DIM. IN X DIR. : (MM OR IN.)',
838 * '// DIM. IN Y DIR. : (MM OR IN.)',
839 * '// RATIO : (CX/CY)',
840 * '// MIN. CLEAR CONCR. COVER : 0 DEFAULT TO 40 MM OR 1.5 IN.'

```

```

841      * // LENGTH INCR. OF CX : 0 DEFAULT TO 50 MM OR 2 IN. //
842      * // LENGTH INCR. OF CY : 0 DEFAULT TO 50 MM OR 2 IN. //
843      READ(4,100) JSHAPE,CX,CY,R,CCCC,CXINC,CYINC
844
845      C CONVERT ALPHABET TO INTEGER
846      C
847      901 IF(JSHAPE.EQ.NSHAP1) GO TO 500
848      IF(JSHAPE.EQ.NSHAP2) GO TO 501
849      IF(JSHAPE.EQ.NSHAP3) GO TO 770
850      IF(TMDE.EQ.1) GO TO 771
851      WRITE(6,772)
852      772 FORMAT(' ',5X,'INVALID SHAPE OF CROSS-SECT. ENTERED//')
853      TSHAPE=0
854      RETURN
855      771 WRITE(7,773)
856      773 FORMAT(' ',4X,'INVALID SHAPE OF CROSS-SECT. ENTERED//')
857      * // ENTER SHAPE : S FOR SQUARE,
858      * // R FOR RECT.,
859      * // C FOR CIRCULAR //
860      READ(4,774) JSHAPE
861      774 FORMAT(A2)
862      GO TO 501
863      770 TSHAPE=3
864      GO TO 502
865      500 TSHAPE=1
866      GO TO 502
867      501 TSHAPE=2
868      502 IF(TOUT.NE.1) GO TO 111
869      WRITE(6,299)
870      GO TO (50,51,52),TSHAPE
871      50 WRITE(6,310)
872      GO TO 93
873      51 WRITE(6,311)
874      GO TO 93
875      52 WRITE(6,312)
876      53 IF(TSHAPE.EQ.3) GO TO 2680
877      WRITE(6,2682) CX,CY,R
878      2682 FORMAT(' ',2X,'CX = ',F6.1,1X,'IN/MM'/3X,'CY = ',F6.1,1X,
879      * // 'IN/MM'/3X,'CX/CY = ',F6.2//')
880      GO TO 2683
881      2680 WRITE(6,2681)CX
882      2681 FORMAT(' ',2X,'DIAMETER = ',F6.1,1X,'IN/MM//')
883      2683 WRITE(6,201) CCCC,CXINC,CYINC
884
885      C IF ONLY REVISING CONCRETE PROPERTIES, GO TO 2
886      C
887      111 IF(TREVIS.EQ.2) GO TO 2
888      C
889      IF(TMDE.EQ.1) GO TO 503
890      READ(5,101) JRFACE,PG,TBMAXX,NBS,TBSMI,TBSMA,TSSS,NX,NY
891      GO TO 504
892      503 WRITE(7,504)
893      504 FORMAT(' ',8F4E7 PG=7 TBMAXX=7 NBS=7 TBSMI=7 TBSMA=7 TSSS=7
894      * // ' ',A2,F7.4,F14.3,212//')
895      * // REINF. PATTERN : X FOR X-FACES, Y FOR Y-FACES,
896      * // S FOR BOTH FACES, C FOR CIRCULAR,
897      * // REINF. RATIO : 0 DEFAULT TO 0.03,
898      * // MAX. PERMISSIBLE NO. BARS : 0 DEFAULT VALUE BY PROGRAM,
899      * // TOTAL NO. BARS : (NX, NY REQUIRED IF BARS IN BOTH FACES)
900      WRITE(7,799)
901      799 FORMAT(' ',4X,'MIN. BAR SIZE : 0 DEFAULT TO 15 OR 5//',
902      * // ' ',4X,'MAX. BAR SIZE : 0 DEFAULT TO 35 OR 11//',
903      * // ' ',4X,'MIN. LATERAL REINF. : 0 DEFAULT TO 10 OR 3//',
904      * // ' ',4X,'NO. BARS IN EACH X-FACE : (NBS IS ENTERED)//',
905      * // ' ',4X,'NO. BARS IN EACH Y-FACE : (NBS IS ENTERED)//')
906      READ(4,101) JRFACE,PG,TBMAXX,NBS,TBSMI,TBSMA,TSSS,NX,NY
907
908      C
909      C IF DATA IS NOT ENTERED, ITS VALUE WOULD BE ASSIGNED BY DEFAULT
910      C
911      504 IF(TBMAXX.NE.0) TBMAXX=TBMAXX
912      C
913      C CONVERT ALPHABET TO INTEGER
914      C
915      775 IF(JRFACE.EQ.NFACE1) GO TO 503
916      IF(JRFACE.EQ.NFACE2) GO TO 504
917      IF(JRFACE.EQ.NFACE3) GO TO 505
918      IF(JRFACE.EQ.NFACE4) GO TO 776
919      IF(TMDE.EQ.1) GO TO 776
920      WRITE(6,777)
921      777 FORMAT(' ',5X,'INVALID REINFORCEMENT PATTERN ENTERED//')
922      TRFACE=0
923      RETURN
924      776 WRITE(7,778)
925      778 FORMAT(' ',4X,'INVALID REINF. PATTERN ENTERED//')
926      * // ENTER REINF. PATTERN : X FOR X-FACES, Y FOR Y-FACES,
927      * // S FOR BOTH FACES, C FOR CIRCULAR //
928      READ(4,774) JRFACE
929      GO TO 775
930      775 TRFACE=4
931      GO TO 506
932      503 TRFACE=1
933      GO TO 506
934      504 TRFACE=2
935      GO TO 506
936      505 TRFACE=3
937
938      C ASSIGN MIN. BAR SIZE # TO CORRESPONDING ARRAY DENOTATION
939      C
940      506 IF(TCODE.EQ.1) GO TO 507
941      IF(TBSMI.GT.10) GO TO 508
942      IF(TBSMI.LT.5) GO TO 508
943      TI=TBSMI-4
944      508 IF(TBSMI.LT.5) TI=1
945      IF(TBSMI.EQ.11) TI=7
946      IF(TBSMI.GE.12.AND.TBSMI.LE.14) TI=6
947      IF(TBSMI.GE.15) TI=9
948      IF(TBSMA.GT.10) GO TO 509
949      IF(TBSMA.LT.5) GO TO 509
950      TK=TBSMA-4
951      509 IF(TBSMA.LE.1) TK=7
952      IF(TBSMA.GE.2.AND.TBSMA.LE.5) TK=1
953      IF(TBSMA.EQ.11) TK=7
954      IF(TBSMA.GE.12.AND.TBSMA.LE.14) TK=6
955      IF(TBSMA.GE.15) TK=9
956      GO TO 510
957      507 IF(TBSMI.LE.15) TI=1
958      IF(TBSMI.GE.16.AND.TBSMI.LE.20) TI=2
959      IF(TBSMI.GE.21.AND.TBSMI.LE.25) TI=3
960      IF(TBSMI.GE.26.AND.TBSMI.LE.30) TI=4

```

```

961 IF(TSSMI.GE.31.AND.TSSMI.LE.35) TI=5
962 IF(TSSMI.GE.36.AND.TSSMI.LE.45) TI=6
963 IF(TSSMI.GE.46) TI=7
964 IF(TSSMA.LE.1) TK=5
965 IF(TSSMA.GE.2.AND.TSSMA.LE.15) TK=1
966 IF(TSSMA.GE.16.AND.TSSMA.LE.20) TK=2
967 IF(TSSMA.GE.21.AND.TSSMA.LE.25) TK=3
968 IF(TSSMA.GE.26.AND.TSSMA.LE.30) TK=4
969 IF(TSSMA.GE.31.AND.TSSMA.LE.35) TK=5
970 IF(TSSMA.GE.36.AND.TSSMA.LE.45) TK=6
971 IF(TSSMA.GE.46) TK=7
972 610 IF(TOUT.NE.1) GO TO 120
973 GO TO(97,98,99,99),TRFACE
974 97 WRITE(6,315)
975 GO TO 88
976 98 WRITE(6,316)
977 GO TO 88
978 99 WRITE(6,317)
979 GO TO 88
980 88 WRITE(6,318)
981 88 WRITE(6,202) TBS(TI),TBS(TK),NBS,PG,TOMAXX,TDSSS,NX,NY
982 C
983 C IF ONLY REVISING REINFORCEMENT PROPERTIES, GO TO 2
984 C
985 120 IF(TREVIS.EQ.3) GO TO 2
986 C
987 IF(TMODE.EQ.1) GO TO 1800
988 READ(5,1000) JBEND,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,SPC
989 GO TO 2999
990 1800 WRITE(7,1800)
991 1800 FORMAT('JBEND=? TXBRAC=? KX=? LUX=? TYBRAC=? KY=? LUY=?',
992 'SPU=? SPC=?')
993 C // (A2,14,F7.3,F10.0,14,F7.3,F10.0,2F8.0)
994 C // BENDING : U FOR UNIAXIAL
995 C // S FOR BIAXIAL
996 C // BRACING ABOUT X-AXIS : 0 FOR UNBRACED
997 C // 1 FOR BRACED
998 C // EFF. LENGTH FACTOR : (ABOUT X-AXIS)
999 WRITE(7,788)
1000 788 FORMAT('S UNSUP. LENGTH ABOUT X-AXIS : (MM OR IN.)')
1001 C // BRACING ABOUT Y-AXIS : 0 FOR UNBRACED
1002 C // 1 FOR BRACED
1003 C // EFF. LENGTH FACTOR : (ABOUT Y-AXIS)
1004 C // UNSUP. LENGTH ABOUT Y-AXIS : (MM OR IN.)
1005 C // SUM AXIAL DESIGN LOAD : (KN OR KIPS)
1006 C // SUM CRITICAL LOAD : (KN OR KIPS)
1007 READ(4,1000) JBEND,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,SPC
1008 SPU=SPU*1000
1009 SPC=SPC*1000
1010 C
1011 C CONVERT ALPHABET TO INTEGER
1012 C
1013 1901 IF(JBEND.EQ.NBEND1) GO TO 811
1014 IF(JBEND.EQ.NBEND2) GO TO 780
1015 IF(TMODE.EQ.1) GO TO 781
1016 WRITE(8,782)
1017 782 FORMAT('S,EX,INVALID BENDING ENTERED')
1018 TBEND=0
1019 RETURN
1020 781 WRITE(7,783)
1021 783 FORMAT('S INVALID BENDING ENTERED')
1022 C // ENTER BENDING : U FOR UNIAXIAL
1023 C // S FOR BIAXIAL
1024 READ(4,774) JBEND
1025 GO TO 1901
1026 780 TBEND=2
1027 GO TO 812
1028 811 TBEND=1
1029 812 IF(TOUT.NE.1) GO TO 2
1030 GO TO(190,181),TBEND
1031 190 WRITE(6,1310)
1032 GO TO 182
1033 181 WRITE(6,1311)
1034 182 WRITE(6,1200) TXBRAC,KX,LUX,TYBRAC,KY,LUY
1035 C
1036 C EVALUATE SOME OF OPTIONAL DATA PARAMETERS
1037 C
1038 2 IF(R.LE.0.AND.TSHAPE.NE.2) R=1.0
1039 IF(PC.LE.0.0001) PC=0.03
1040 IF(PC.LE.PMIN) PC=PMIN
1041 IF(PC.GE.PMAX) PC=PMAX
1042 IF(DCCC.GT.0.1) DCCC=DCCC
1043 IF(CXINC.GT.0.1) CXINC=CXINC
1044 IF(CYINC.GT.0.1) CYINC=CYINC
1045 IF(TDSSS.EQ.0) GO TO 20
1046 DSS=PLATOS(TDSSS)
1047 GO TO 21
1048 20 DSS=MINLAT
1049 IF(TCODE.EQ.1) TDSSS=10
1050 IF(TCODE.EQ.2) TDSSS=3
1051 21 IF(TOUT.NE.1) GO TO 3000
1052 WRITE(6,400)
1053 3000 IF(TOUT.NE.1) GO TO 3
1054 WRITE(6,300)
1055 WRITE(6,301) PC,DCC,DSS,CXINC,CYINC
1056 IF(TOUT.NE.1) GO TO 3001
1057 WRITE(6,500)
1058 3001 IF(TSSMI.LT.5) GO TO 4
1059 IF(TOUT.NE.1) GO TO 5
1060 WRITE(6,501) TBS(TI)
1061 GO TO 5
1062 4 IF(TOUT.NE.1) GO TO 5
1063 WRITE(6,502) TBS(TI)
1064 5 IF(TSSMA.LT.5) GO TO 6
1065 IF(TOUT.NE.1) GO TO 7
1066 WRITE(6,503) TBS(TK)
1067 GO TO 7
1068 6 IF(TOUT.NE.1) GO TO 7
1069 WRITE(6,504) TBS(TK)
1070 GO TO(5,8,10,11),TRFACE
1071 8 IF(TOUT.NE.1) GO TO 3002
1072 WRITE(6,505)
1073 GO TO 12
1074 9 IF(TOUT.NE.1) GO TO 3002
1075 WRITE(6,506)
1076 GO TO 12
1077 10 IF(TOUT.NE.1) GO TO 3002
1078 WRITE(6,507)
1079 GO TO 12
1080 11 IF(TOUT.NE.1) GO TO 3002

```

```

1081      WRITE(6,508)
1082      WRITE(6,509)
1083      3002 GO TO(13,14,15),TSHAPE
1084      13 IF(TOUT.NE.1) RETURN
1085      WRITE(6,510)
1086      GO TO 16
1087      14 IF(TOUT.NE.1) RETURN
1088      WRITE(6,511)
1089      GO TO 16
1090      15 IF(TOUT.NE.1) RETURN
1091      WRITE(6,512)
1092      16 IF(TSHAPE.EQ.3) GO TO 2884
1093      WRITE(6,513) CX,CY,R
1094      RETURN
1095      2884 WRITE(6,2885) CX
1096      2885 FORMAT(13X,'DIAMETER = ',F5.1,1X,'IN/MM')
1097      RETURN
1098      C
1099      C *****FORMAT STATEMENTS*****
1100      C
1101      100 FORMAT(A2,2F5.1,F5.2,3F5.1)
1102      201 FORMAT(' ',2X,
1103      = 'MINIMUM CLEAR CONCRETE COVER = ',
1104      = F5.1,1X,'IN/MM'/3X,'CXINCR = ',F5.1,1X,'IN/MM'/3X,
1105      = 'CYINCR = ',F5.1,1X,'IN/MM')
1106      101 FORMAT(A2,F7.4,4I4,13,2I2)
1107      202 FORMAT(' ',2X,'MINIMUM BAR SIZE = #',
1108      = 14/3X,'MAXIMUM BAR SIZE = #',14/3X,
1109      = 'SPECIFIED TOTAL NUMBER OF BARS USED = ',14/3X,
1110      = 'MAXIMUM PERCENTAGE OF STEEL = ',F7.4/3X,
1111      = 'ENTERED MAXIMUM PERMISSIBLE NO. OF BARS IN A FACE = ',
1112      = 14/3X,'MINIMUM LATERAL REINFORCEMENT DIAMETER = ',
1113      = 'ENTERED = #',13/3X,'NO. OF BARS IN X-FACE = ',
1114      = 'ONLY = #',12/3X,'NO. OF BARS IN Y-FACE ONLY = ',12)
1115      C
1116      300 FORMAT(2X,'PC',7X,'DCC',3X,'DSS',2X,'CXINCR',3X,
1117      = 'CYINCR')
1118      301 FORMAT(F7.4,3X,F5.1,1X,F5.3,3X,F5.1,4X,F5.1/)
1119      289 FORMAT(' ',THE FOLLOWING GEOMETRY',
1120      = ' DATA WAS ENTERED : '/')
1121      310 FORMAT(' ',2X,'SHAPE = SQUARE')
1122      311 FORMAT(' ',2X,'SHAPE = RECTANGULAR')
1123      312 FORMAT(' ',2X,'SHAPE = CIRCULAR')
1124      313 FORMAT(' ',2X,'REINFORCEMENT = X-FACE ONLY')
1125      316 FORMAT(' ',2X,'REINFORCEMENT = Y-FACE ONLY')
1126      317 FORMAT(' ',2X,'REINFORCEMENT = BOTH FACES')
1127      318 FORMAT(' ',2X,'REINFORCEMENT = AROUND THE CORE DIAMETER')
1128      400 FORMAT(' ',THE FOLLOWING GEOMETRIC DATA',
1129      = ' WAS ASSIGNED BY DEFAULT : '/')
1130      500 FORMAT(2X,'LONGITUDINAL REINFORCEMENT DETAILS:')
1131      501 FORMAT(13X,'MINIMUM SIZE= #',14)
1132      502 FORMAT(13X,'DEFAULT MINIMUM SIZE= #',14)
1133      503 FORMAT(13X,'MAXIMUM SIZE= #',14)
1134      504 FORMAT(13X,'DEFAULT MAXIMUM SIZE= #',14)
1135      505 FORMAT(13X,'REINFORCEMENT IN X-FACE ONLY')
1136      506 FORMAT(13X,'REINFORCEMENT IN Y-FACE ONLY')
1137      507 FORMAT(13X,'REINFORCEMENT IN BOTH FACES')
1138      508 FORMAT(13X,'REINFORCEMENT AROUND THE CORE DIAMETER')
1139      509 FORMAT(2X,'SPECIFIED DIMENSIONS:')
1140      510 FORMAT(13X,'SHAPE= SQUARE')
1141      511 FORMAT(13X,'SHAPE= RECTANGULAR')
1142      512 FORMAT(13X,'SHAPE= CIRCULAR')
1143      513 FORMAT(13X,'CX= ',F5.1,1X,'IN/MM',3X,'CY= ',F5.1,1X,'IN/MM',
1144      = 3X,'CX/CY= ',F5.2/)
1145      1100 FORMAT(A2,14,F7.3,F10.0,14,F7.3,F10.0,2F5.0)
1146      1200 FORMAT(' ',2X,'BRACING WHERE 0 VALUE FOR UNBRAC. 1 FOR',
1147      = ' BRAC./3X,'XBRACING = ',14/3X,'ROTATIONAL RESTRAINT',
1148      = ' ABOUT X-AXIS BENDING = ',F7.3/3X,'UNSUPPORTED LENGTH',
1149      = ' ABOUT X-AXIS = ',F10.0,1X,'IN/MM'/3X,'YBRACING = ',14/3X,
1150      = 'ROTATIONAL RESTRAINT ABOUT Y-AXIS BENDING = ',
1151      = ' F7.3/3X,'UNSUPPORTED LENGTH ABOUT Y-AXIS = ',F10.0,1X,
1152      = 'IN/MM')
1153      1310 FORMAT(' ',2X,'BENDING = UNI-AXIAL')
1154      1311 FORMAT(' ',2X,'BENDING = BI-AXIAL')
1155      END
1156      C
1157      C *****
1158      C
1159      C SUBROUTINE LOAD1
1160      C
1161      C THIS SUBROUTINE READS LOADING PROPERTIES, COMPUTES FACTORED
1162      C LOADING WITH RESPECT TO BENDING CONDITIONS
1163      C
1164      C *****
1165      C
1166      IMPLICIT REAL(A-M,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
1167      COMMON PC,ESMAX,TYPE,TREINF,TLEN1,TBEND,TLEN2,TADEO,TPROB
1168      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,YI,YK,TBS(9),DB(9),
1169      = PD,FL,ECU,PMAX,PMIN,SBI,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
1170      = TBMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAXX
1171      COMMON FPC,FY,FSY,EC,BI,YBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
1172      = SPC,PD,PL,PU,MX1,MX2,BDX,MY1,MY2,SDY,TOUT,TMODE
1173      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,MBS,MWAX
1174      COMMON TDMFG,AG,AS,CG,SB,DP,AST,NR,NY,SDX,SBY,DS,
1175      = Y(12),ISEX,K(12),ISEY,TMODAS,PCDES,EY,MCX,
1176      = MAPPX,PUDEX,MUDEX,PSY
1177      COMMON EX,MCV,MAPPY,PUDEXY,MUDEXY,PSY,MOYREG,MOKREG,
1178      = YC,YB,ECS(12),ETS(12),CS(12),FT(12),
1179      = XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
1180      COMMON TESTMO
1181      COMMON NLC,JPOB,TREVIS,TDESS,SPACE,PITCH,NSPACR
1182      DIMENSION A(8)
1183      DATA NFACT/'F'/
1184      DATA NDES1,NDES2,NDES3/'D','L','C'/
1185      C
1186      IF(TMODE.EQ.1) GO TO 902
1187      READ(5,101) JFACT,PD,PL,MDXT,MLXT,MOKB,MLKB
1188      GO TO 1990
1189      902 WRITE(7,504)
1190      504 FORMAT('S FACT=? PD=? PL=? MDXT=? MLXT=? MOKB=? MLKB=?',
1191      = /,
1192      = ( A2,6F5.0),
1193      = /,
1194      = ' FACT : U FOR UNFACTORED LOADING ENTERED',
1195      = /,
1196      = ' F FOR FACTORED LOADING ENTERED',
1197      = /,
1198      = ' AXIAL DEAD AND LIVE LOAD : (KN OR KIPS)',
1199      = /,
1200      = ' MOM. ABOUT X-AXIS - DEAD LOAD ON TOP : (KN-M OR KIP-FT)',
1201      = /,
1202      = ' LIVE LOAD ON TOP : (KN-M OR KIP-FT)',
1203      = /,
1204      = ' DEAD LOAD ON BOT : (KN-M OR KIP-FT)',
1205      = /,
1206      = ' LIVE LOAD ON BOT : (KN-M OR KIP-FT)')
1207      READ(4,405) JFACT,PD,PL,MDXT,MLXT,MOKB,MLKB
1208      505 FORMAT(A2,6F5.0)

```

```

1201 1990 IF(TCODE.EQ.1) GO TO 1991
1202 PD=PD+1000.
1203 PL=PL+1000.
1204 MDXT=MDXT+12000.
1205 MLXT=MLXT+12000.
1206 MDXB=MDXB+12000.
1207 MLXB=MLXB+12000.
1208 GO TO 903
1209 1991 PD=PD+1000.
1210 PL=PL+1000.
1211 MDXT=MDXT+1000000.
1212 MLXT=MLXT+1000000.
1213 MDXB=MDXB+1000000.
1214 MLXB=MLXB+1000000.
1215 C
1216 C CONVERT ALPHABET TO INTEGER
1217 C
1218 903 IF(JFACT.EQ.NFACT) GO TO 900
1219 TFACT=0
1220 GO TO 901
1221 900 TFACT=1
1222 901 IF(TOUT.NE.1) GO TO 2
1223 WRITE(6,150)
1224 IF(TFACT.EQ.1) GO TO 93
1225 WRITE(6,313)
1226 GO TO 94
1227 93 WRITE(6,312)
1228 94 PPD=PD/1000.
1229 PPL=PL/1000.
1230 IF(TCODE.EQ.1) GO TO 1992
1231 MMDXT=MDXT/12000.
1232 MMLXT=MLXT/12000.
1233 MMDXB=MDXB/12000.
1234 MMLXB=MLXB/12000.
1235 GO TO 1993
1236 1992 MMDXT=MDXT/1000000.
1237 MMLXT=MLXT/1000000.
1238 MMDXB=MDXB/1000000.
1239 MMLXB=MLXB/1000000.
1240 1993 WRITE(6,201) PPD,PPL,MMDXT,MMLXT,MMDXB,MMLXB
1241 2 IF(TMODE.EQ.1) GO TO 904
1242 READ(5,102) JDESIG,MDYT,MLYT,MDYS,MLYS
1243 GO TO 1994
1244 904 WRITE(7,906)
1245 906 FORMAT('A DESIGN=? MDYT=? MLYT=? MDYS=? MLYS=? (A2,4F8.0)'/,
1246 =/' DESIGN : D FOR PROGRAM DESIGNS A SECT.',
1247 =/' : L FOR ADEQ. CHECK OF SELECTED SECT.(MULT. LOAD CASES)',
1248 =/' : C FOR A CHECK OF ENTERED SECT.',
1249 =/' MOM. ABOUT Y-AXIS - DEAD LOAD ON TOP : (KN-M OR KIP-FT)',
1250 =/' LIVE LOAD ON TOP : (KN-M OR KIP-FT)',
1251 =/' DEAD LOAD ON BOT. : (KN-M OR KIP-FT)',
1252 =/' LIVE LOAD ON BOT. : (KN-M OR KIP-FT)'/)
1253 READ(4,907) JDESIG,MDYT,MLYT,MDYS,MLYS
1254 907 FORMAT(A2,4F8.0)
1255 1994 IF(TCODE.EQ.1) GO TO 1995
1256 MDYT=MDYT+12000.
1257 MLYT=MLYT+12000.
1258 MDYS=MDYS+12000.
1259 MLYS=MLYS+12000.
1260 GO TO 905
1261 1995 MDYT=MDYT+1000000.
1262 MLYT=MLYT+1000000.
1263 MDYS=MDYS+1000000.
1264 MLYS=MLYS+1000000.
1265 C
1266 C CONVERT ALPHABET TO INTEGER
1267 C
1268 905 IF(JDESIG.EQ.NDES1) GO TO 902
1269 IF(JDESIG.EQ.NDES2) GO TO 903
1270 IF(JDESIG.EQ.NDES3) GO TO 904
1271 IF(TMODE.EQ.1) GO TO 751
1272 WRITE(6,752)
1273 752 FORMAT('A,EX,INVALID DESIGN CONDITION ENTERED'/)
1274 TDESIG=0
1275 RETURN
1276 751 WRITE(7,753)
1277 753 FORMAT('A INVALID DESIGN CONDITION ENTERED'/,
1278 =/' ENTER DESIGN : D FOR PROGRAM DESIGNS A SECT.',
1279 =/' : L FOR ADEQ. CHECK OF SELECTED SECT.(MULT. LOAD CASES)',
1280 =/' : C FOR A CHECK OF ENTERED SECT.'/)
1281 READ(4,754) JDESIG
1282 754 FORMAT(A2)
1283 GO TO 905
1284 750 TDESIG=3
1285 GO TO 904
1286 902 TDESIG=1
1287 GO TO 904
1288 903 TDESIG=2
1289 904 IF(TOUT.NE.1) GO TO 3
1290 MSPU=MSPU/1000.
1291 MSPC=MSPC/1000.
1292 IF(TCODE.EQ.1) GO TO 1996
1293 MMDYT=MDYT/12000.
1294 MMLYT=MLYT/12000.
1295 MMDYS=MDYS/12000.
1296 MMLYS=MLYS/12000.
1297 GO TO 1997
1298 1996 MMDYT=MDYT/1000000.
1299 MMLYT=MLYT/1000000.
1300 MMDYS=MDYS/1000000.
1301 MMLYS=MLYS/1000000.
1302 1997 WRITE(6,202) MMDYT,MMLYT,MMDYS,MMLYS,MSPU,MSPC
1303 GO TO(130,131,132),TDESIG
1304 130 WRITE(6,135)
1305 GO TO 3
1306 131 WRITE(6,136)
1307 GO TO 3
1308 132 WRITE(6,137)
1309 C
1310 C COMPUTE FACTORED LOADINGS
1311 C
1312 3 IF(TFACT.EQ.1) GO TO 10
1313 PD=1.4
1314 PL=1.7
1315 PUF=PD+PL+PL
1316 MXT=PD+MDXT+PL+MLXT
1317 MNB=PD+MDXB+PL+MLXB
1318 MYT=PD+MDYT+PL+MLYT
1319 MYB=PD+MDYS+PL+MLYS
1320 GO TO 11

```

```

1321      10  PU=PD+PL
1322          MXT=MDXT+MLXT
1323          MXB=MDXB+MLXB
1324          MYT=MDYT+MLYT
1325          MYB=MDYB+MLYB
1326          PD=1.0
1327          PL=1.0
1328
1329      C
1330      C DETERMINE WHICH IS SMALLER AND LARGER END MOMENTS
1331      C
1332      11  IF(ABS(MXT).LE.O..AND.ABS(MXB).LE.O.) GO TO 802
1333          IF(ABS(MXT).GE.ABS(MXB)) GO TO 12
1334          MX1=MXT
1335          MX2=MXB
1336          GO TO 800
1337      12  MX1=MXB
1338          MX2=MXT
1339      800  IF(MDXB.GT.MDXT) GO TO 801
1340          SBDX=PD=MDXT/MX2
1341          SDX=ABS(SBDX)
1342          GO TO 13
1343      801  SBDX=PD=MDXB/MX2
1344          SDX=ABS(SBDX)
1345          GO TO 13
1346      802  SDX=0.
1347          MX1=0.0
1348          MX2=0.0
1349          GO TO 15
1350
1351      C
1352      C COLUMN DEFLECTS IN SINGLE CURVATURE OR DOUBLE CURVATURE
1353      C
1354      13  IF(MX1.LT.O..AND.MX2.GT.O.) GO TO 14
1355          IF(MX2.LT.O..AND.MX1.GT.O.) GO TO 14
1356          IF(TOUT.NE.1) GO TO 15
1357          IF(TMODE.EQ.1) WRITE(7,203)
1358          WRITE(6,203)
1359          GO TO 15
1360      14  IF(TOUT.NE.1) GO TO 15
1361          IF(TMODE.EQ.1) WRITE(7,204)
1362          WRITE(6,204)
1363
1364      C
1365      C ONLY USED FOR BIAxIAL BENDING
1366      C
1367      15  IF(TBEND.EQ.1) GO TO 999
1368          IF(ABS(MYT).LE.O..AND.ABS(MYB).LE.O.) GO TO 805
1369          IF(ABS(MYT).GE.ABS(MYB)) GO TO 16
1370          MY1=MYT
1371          MY2=MYB
1372          GO TO 803
1373      16  MY1=MYB
1374          MY2=MYT
1375      803  IF(MDYB.GT.MDYT) GO TO 804
1376          SBDY=PD=MDYT/MY2
1377          SDY=ABS(SBDY)
1378          GO TO 19
1379      804  SBDY=PD=MDYB/MY2
1380          SDY=ABS(SBDY)
1381          GO TO 19
1382      805  SDY=0.
1383          MY1=0.0
1384          MY2=0.0
1385          GO TO 17
1386      19  IF(MY1.LT.O..AND.MY2.GT.O.) GO TO 20
1387          IF(MY2.LT.O..AND.MY1.GT.O.) GO TO 20
1388          IF(TOUT.NE.1) RETURN
1389          IF(TMODE.EQ.1) WRITE(7,205)
1390          WRITE(6,205)
1391          GO TO 17
1392      20  IF(TOUT.NE.1) RETURN
1393          IF(TMODE.EQ.1) WRITE(7,206)
1394          WRITE(6,206)
1395          GO TO 17
1396      999  MY2=0.0
1397      17  IF(TOUT.NE.1) RETURN
1398          PPU=PU/1000.
1399          IF(TCODE.EQ.1) GO TO 1998
1400          MMX2=ABS(MX2)/12000.
1401          MMV2=ABS(MY2)/12000.
1402          GO TO 1999
1403      1998  MMX2=ABS(MX2)/1000000.
1404          MMV2=ABS(MY2)/1000000.
1405      1999  IF(TMODE.EQ.1) WRITE(7,303) PPU,MMX2,MMV2
1406          WRITE(6,303) PPU,MMX2,MMV2
1407          RETURN
1408
1409      C
1410      C *****FORMAT STATEMENTS*****
1411      C
1412      101  FORMAT(A2,8F8.0)
1413      201  FORMAT(' ',2X,'AXIAL DEAD LOAD = ',F8.0,1X,'KIPS/KN'/3X,
1414          & 'AXIAL LIVE LOAD = ',
1415          & F8.0,1X,'KIPS/KN'/3X,'MDXT= ',F8.0,1X,'KIP-FT/KN-M',
1416          & '/3X,'MLXT= ',F8.0,1X,'KIP-FT/KN-M'/3X,'MDXB= ',
1417          & F8.0,1X,'KIP-FT/KN-M'/3X,'MLXB= ',F8.0,1X,'KIP-FT/KN-M')
1418      102  FORMAT(A2,4F8.0)
1419      202  FORMAT(' ',2X,'MDYT= ',F8.0,1X,'KIP-FT/KN-M'/3X,'MLYT= ',
1420          & F8.0,1X,'KIP-FT/KN-M'/3X,'MDYB= ',F8.0,1X,'KIP-FT/KN-M',
1421          & '/3X,'MLYB= ',F8.0,1X,'KIP-FT/KN-M'/3X,'SUMMATION',
1422          & ' OF PU FOR ALL COLUMN IN A STOREY = ',F8.0,1X,'KIPS/KN',
1423          & '/3X,'SUMMATION OF CRITICAL LOAD FOR ALL COLUMN IN A STOREY= ',
1424          & F8.0,1X,'KIPS/KN'/)
1425
1426      C
1427      203  FORMAT(' ',6X,'COLUMN DEFLECTS IN SINGLE',
1428          & ' CURVATURE ABOUT X-AXIS'/)
1429      204  FORMAT(' ',6X,'COLUMN DEFLECTS IN DOUBLE',
1430          & ' CURVATURE ABOUT X-AXIS'/)
1431      205  FORMAT(' ',6X,'COLUMN DEFLECTS IN SINGLE',
1432          & ' CURVATURE ABOUT Y-AXIS'/)
1433      206  FORMAT(' ',6X,'COLUMN DEFLECTS IN DOUBLE',
1434          & ' CURVATURE ABOUT Y-AXIS'/)
1435
1436      C
1437      303  FORMAT(' ',2X,'LOAD CONDITION : '//2X,'PU = ',F8.0,1X,
1438          & 'KIPS/KN',
1439          & '/2X,'MMX2= ',F8.0,1X,'KIP-FT/KN-M',2X,'MMV2= ',F8.0,
1440          & 'KIP-FT/KN-M'/)
1441      150  FORMAT(' ',2X,'THE FOLLOWING LOADING',
1442          & ' DATA WAS ENTERED : '//)
1443      212  FORMAT(' ',2X,'LOADING = FACTORED')
1444      312  FORMAT(' ',2X,'LOADING = UNFACTORED')
1445      135  FORMAT(' ',2X,'CONDITION= DESIGN')
1446      136  FORMAT(' ',2X,'CONDITION= ADEQUACY CHECK FOR MULTIPLE LOAD')

```

```

1441      * ' CASE'//)
1442      137 FORMAT(' ',2X,'CONDITION= CHECK OF',
1443      * ' ENTERED SECTION'//)
1444      END
1445
1446      C *****
1447      C *
1448      C SUBROUTINE DIMEN1
1449
1450      C *THIS SUBROUTINE SELECTS INITIAL CONCRETE DIMENSIONS, ECHO
1451      C *CHECKS OF INPUT DATA AND ASSIGNS DIMENSION DEFAULT FLAGS
1452      C *TO CORRESPONDING DIMENSION ENTRY
1453      C *
1454      C *****
1455      C
1456      IMPLICIT REAL(A-N,I,K-M,P,R,S,W-Y), INTEGER(T,J,N,S)
1457      COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TSLEN2,TADEO,TPROB
1458      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CVINCR,TI,TK,TBS(9),DS(9),
1459      * FD,FL,ECU,PNAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
1460      * TSMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TSMAXX
1461      COMMON FPC,FY,FSY,EC,B1,TSRAC,KX,LUX,TYRAC,KY,LUY,SPU,
1462      * SPC,PD,PL,PU,MX1,MX2,BDX,MY1,MY2,BDY,TOUT,TMODE
1463      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
1464      COMMON TDIMFG,AG,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
1465      * Y(12),ISEX,X(12),ISEY,TMODAS,PDES,EY,MCX,
1466      * MAPPX,PUDEX,MUDEX,PRX
1467      COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PBY,MOYREQ,MOXREQ,
1468      * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
1469      * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
1470      COMMON TESTMO
1471      COMMON NLC,JPOS,TREVIS,TDSS,SPACE,PITCH,NSPACR
1472      DIMENSION A(9)
1473
1474      C
1475      CG=1./12.
1476
1477      C DETERMINE CROSS SECTION AREA BY USING MAX. AXIAL COMPRESSION
1478      C NOMINAL STRENGTH BASED ON MIN. ECCENTRICITY. ONLY USED FOR
1479      C SELECTING TRIAL SECTION
1480
1481      AG=AGG(PU,PHIV,RFAC,FPC,PG,FY)
1482
1483      C EVALUATE AND ECHO CHECK DATA ENTRY
1484      C
1485      IF(CX.LE.0.1) GO TO 10
1486      IF(CX.LT.MINDIM) GO TO 30
1487      IF(CY.GT.0.1) GO TO 80
1488      IF(R.LE.0.1) GO TO 12
1489      IF(TSHAPE.EQ.2) GO TO 82
1490      IF(ABS(R-1.0).LE.0.) GO TO 82
1491      IF(TOUT.NE.1) GO TO 3000
1492      IF(TMODE.EQ.1) GO TO 2000
1493      GO TO 153
1494      2000 WRITE(7,102)
1495      153 WRITE(8,102)
1496      3000 R=1.0
1497      82 CY=CX/R
1498      GO TO 11
1499      80 IF(CY.LT.MINDIM) GO TO 31
1500      XOY=CX/CY
1501      IF(TSHAPE.EQ.2) GO TO 81
1502      IF(ABS(XOY-R).LT.0.01) GO TO 11
1503      IF(TOUT.NE.1) GO TO 3001
1504      IF(TMODE.EQ.1) GO TO 2001
1505      GO TO 154
1506      2001 WRITE(7,100)
1507      154 WRITE(8,100)
1508      3001 CY=CX
1509      GO TO 11
1510      81 IF(R.LE.0.1) GO TO 11
1511      IF(ABS(XOY-R).LT.0.01) GO TO 11
1512      IF(TOUT.NE.1) GO TO 11
1513      IF(TMODE.EQ.1) GO TO 2002
1514      GO TO 155
1515      2002 WRITE(7,101)
1516      155 WRITE(8,101)
1517      GO TO 11
1518
1519      C ENTRY OF DIMENSIONS IS CLASSIFIED INTO TDIMFG=2
1520      C
1521      12 TDIMFG=2
1522      CY=AG/CX
1523
1524      C COMPUTED DIMENSION IS ROUND DOWN TO NEXT SMALLER INTEGER VALUE
1525      C
1526      CY=CVINCR=IFIX(CY/CVINCR)
1527      IF(CY.LE.MINDIM) GO TO 34
1528      GO TO 35
1529
1530      C MIN. DIMENSION IS USED IF COMPUTED ONE IS LESS THAN THAT
1531      C
1532      34 CY=MINDIM
1533      IF(TOUT.NE.1) GO TO 35
1534      IF(TMODE.EQ.1) GO TO 2004
1535      GO TO 156
1536      2004 WRITE(7,303)
1537      156 WRITE(8,303)
1538      35 AG=CX*CY
1539      GO TO 86
1540      10 IF(CY.LE.0.1) GO TO 14
1541      IF(CY.LT.MINDIM) GO TO 31
1542      IF(R.LE.0.1) GO TO 12
1543      IF(TSHAPE.EQ.2) GO TO 84
1544      IF(R.LE.1.0) GO TO 84
1545      IF(TOUT.NE.1) GO TO 3005
1546      IF(TMODE.EQ.1) GO TO 2005
1547      GO TO 157
1548      2005 WRITE(7,102)
1549      157 WRITE(8,102)
1550      3005 R=1.0
1551      84 CX=CY*R
1552      GO TO 11
1553
1554      C ENTRY OF DIMENSION IS CLASSIFIED INTO TDIMFG=3
1555      C
1556      15 TDIMFG=3
1557      CX=AG/CY
1558
1559      C COMPUTED DIMENSION IS ROUND DOWN TO NEXT INTEGER VALUE
1560      C
1561      CX=CXINCR=IFIX(CX/CXINCR)

```

```

1561       IF(CX.LE.MINDIM) GO TO 36
1562       GO TO 37
1563 C MIN. DIMENSION IS USED IF COMPUTED ONE IS LESS THAN THAT
1564 C
1565
1566 36 CX=MINDIM
1567   IF(TOUT.NE.1) GO TO 37
1568   IF(TMODE.EQ.1) GO TO 2007
1569   GO TO 158
1570 2007 WRITE(7,304)
1571 158 WRITE(6,304)
1572 37 AG=CY
1573   GO TO 88
1574 14 IF(R.LE.0.01) GO TO 16
1575   IF(TSHAPE.EQ.2) GO TO 85
1576   IF(R.LE.1.0) GO TO 85
1577   IF(TOUT.NE.1) GO TO 3006
1578   IF(TMODE.EQ.1) GO TO 2008
1579   GO TO 159
1580 2008 WRITE(7,102)
1581 159 WRITE(6,102)
1582 3006 R=1.0
1583 C
1584 C ENTRY OF DIMENSION IS CLASSIFIED INTO TDIMFG=4
1585 C
1586 85 TDIMFG=4
1587   IF(TSHAPE.EQ.3) GO TO 22
1588   F=AG/R
1589   CY=SQRT(F)
1590   CY=CYINCR+IFIX(CY/CYINCR)
1591   IF(CY.LE.MINDIM) GO TO 40
1592   GO TO 41
1593 40 CY=MINDIM
1594   IF(TOUT.NE.1) GO TO 41
1595   IF(TMODE.EQ.1) GO TO 2010
1596   GO TO 160
1597 2010 WRITE(7,303)
1598 160 WRITE(6,303)
1599 41 CX=CY/R
1600   CX=CXINCR+IFIX(CX/CXINCR)
1601   IF(CX.LE.MINDIM) GO TO 42
1602   GO TO 43
1603 42 CX=MINDIM
1604   IF(TOUT.NE.1) GO TO 43
1605   IF(TMODE.EQ.1) GO TO 2011
1606   GO TO 161
1607 2011 WRITE(7,304)
1608 161 WRITE(6,304)
1609 43 AG=CY
1610   GO TO 88
1611 22 CG=1./15.
1612   F=(4.*AG)/PRIME
1613   CX=SQRT(F)
1614   CX=CXINCR+IFIX(CX/CXINCR)
1615   IF(CX.LE.MINDIM) GO TO 38
1616   GO TO 39
1617 38 CX=MINDIM
1618   IF(TOUT.NE.1) GO TO 39
1619   IF(TMODE.EQ.1) GO TO 2013
1620   GO TO 162
1621 2013 WRITE(7,304)
1622 162 WRITE(6,304)
1623 39 CY=CX
1624   AG=(PRIME+CX**2)/4.
1625   GO TO 88
1626 C
1627 C ENTRY OF DIMENSION IS CLASSIFIED INTO TDIMFG=5
1628 C
1629 16 TDIMFG=5
1630 C
1631 C FOR UNIAXIAL BENDING, THE CONCEPT OF MIN. ECCENTRICITY (0.1M)
1632 C IS APPLIED . FOR BIAXIAL BENDING, RATIO OF CX/CY IS DETERMINED
1633 C BY CONSIDERING MY2/MX2
1634 C
1635   IF(TSEND.EQ.1) GO TO 87
1636 C
1637 C IF NO MOMENTS ARE ENTERED, R=1.0
1638 C
1639   IF(ABS(MX2).LE.0..OR.ABS(MY2).LE.0.) GO TO 240
1640   R=ABS(MY2/MX2)
1641   GO TO 241
1642 240 R=1.0
1643 241 F=R*AG
1644   CX=SQRT(F)
1645   CX=CXINCR+IFIX(CX/CXINCR)
1646   IF(CX.LE.MINDIM) GO TO 44
1647   GO TO 45
1648 44 CX=MINDIM
1649   IF(TOUT.NE.1) GO TO 45
1650   IF(TMODE.EQ.1) GO TO 2015
1651   GO TO 163
1652 2015 WRITE(7,304)
1653 163 WRITE(6,304)
1654 45 CY=CX/R
1655   CY=CYINCR+IFIX(CY/CYINCR)
1656   IF(CY.LE.MINDIM) GO TO 46
1657   GO TO 47
1658 46 CY=MINDIM
1659   IF(TOUT.NE.1) GO TO 47
1660   IF(TMODE.EQ.1) GO TO 2016
1661   GO TO 164
1662 2016 WRITE(7,303)
1663 164 WRITE(6,303)
1664 47 AG=CY
1665   GO TO 88
1666 87 CY=ABS(MX2)/(0.1*PU)
1667   CY=CYINCR+IFIX(CY/CYINCR)
1668   IF(CY.LE.MINDIM) GO TO 90
1669   GO TO 91
1670 90 CY=MINDIM
1671   IF(TOUT.NE.1) GO TO 91
1672   IF(TMODE.EQ.1) GO TO 2017
1673   GO TO 165
1674 2017 WRITE(7,303)
1675 165 WRITE(6,303)
1676 91 CX=AG/CY
1677   CX=CXINCR+IFIX(CX/CXINCR)
1678   IF(CX.LE.MINDIM) GO TO 92
1679   GO TO 47
1680 92 CX=MINDIM

```

```

1681       IF(TOUT.NE.1) GO TO 47
1682       IF(TMODE.EQ.1) GO TO 2016
1683       GO TO 168
1684       2016 WRITE(7,304)
1685       168 WRITE(6,304)
1686       GO TO 47
1687
1688 C ENTRY OF DIMENSION IS CLASSIFIED INTO TDMFC=1
1689 C
1690 11 TDMFC=1
1691 GO TO 32
1692 30 IF(TMODE.EQ.1) GO TO 2019
1693 GO TO 167
1694 2019 WRITE(7,3105)
1695 WRITE(7,301)
1696 167 IF(TOUT.NE.1) GO TO 3100
1697 WRITE(6,3105)
1698 WRITE(6,301)
1699 GO TO 33
1700 3100 WRITE(6,301)
1701 GO TO 33
1702 31 IF(TMODE.EQ.1) GO TO 2020
1703 GO TO 168
1704 2020 WRITE(7,3105)
1705 WRITE(7,302)
1706 168 IF(TOUT.NE.1) GO TO 3101
1707 WRITE(6,3105)
1708 WRITE(6,302)
1709 GO TO 33
1710 3101 WRITE(6,302)
1711
1712 C PROCEED TO NEXT PROBLEM DUE TO ENTERED DIMENSIONS LESS THAN MINDIM
1713 C
1714 33 TESTDI=2
1715 RETURN
1716 32 IF(TSHAPE.EQ.3) GO TO 20
1717 AG=CX*CY
1718 GO TO 83
1719 20 CG=1./16.
1720 AG=(PRIME*CG**2)/4.
1721 83 IF(TDESIG.EQ.1) GO TO 86
1722 IF(TOUT.NE.1) GO TO 3010
1723 IF(TMODE.EQ.1) WRITE(7,400)
1724 WRITE(6,400)
1725
1726 C RETURN TO MAIN ROUTINE TO CONTINUE EXECUTION
1727 C
1728 3010 TESTDI=1
1729 RETURN
1730 86 AS=PMIN*AG
1731 IF(TOUT.NE.1) GO TO 3012
1732 IF(TSHAPE.EQ.3) GO TO 2880
1733 IF(TMODE.EQ.1) GO TO 2024
1734 GO TO 172
1735 2024 WRITE(7,201) CX,CY
1736 172 WRITE(6,201) CX,CY
1737 201 FORMAT(' ','INITIAL TRIAL CONCRETE DIMENSION : '//2X,'CX = ',
1738 ' F6.1,1X,'IN/MM',' ,CX,CY = ',F6.1,1X,'IN/MM')
1739 GO TO 3012
1740 IF(TMODE.EQ.1) WRITE(7,2881) CX
1741 2881 FORMAT(' ','INITIAL TRIAL CONCRETE DIMENSION : '//2X,
1742 ' DIAMETER = ',F6.1,1X,'IN/MM')
1743 WRITE(6,2881) CX
1744
1745 C 3012 TESTDI=1
1746 RETURN
1747
1748 C 301 FORMAT(' ',5X,'THE GIVEN DIMENSION CX IS LESS THAN THE',
1749 ' ' MINIMUM DIMENSION : '//)
1750 302 FORMAT(' ',5X,' THE GIVEN DIMENSION CY IS LESS THAN THE',
1751 ' ' MINIMUM DIMENSION : '//)
1752 303 FORMAT(' ',5X,'MINIMUM DIMENSION CY IS USED')
1753 304 FORMAT(' ',5X,'MINIMUM DIMENSION CX IS USED')
1754 100 FORMAT(' ',5X,'GEOMETRIC INPUT DATA IS WRONG')
1755 101 FORMAT(' ',5X,'GEOMETRIC INPUT DATA IS WRONG DUE TO ASPECT',
1756 ' ' RATIO')
1757 102 FORMAT(' ',5X,'GEOMETRIC INPUT DATA IS WRONG DUE TO',
1758 ' ' ASPECT RATIO',5X,'SET IT EQUAL TO 1.0')
1759 400 FORMAT(' ',5X,'ADEQUACY CHECK OF DESIGN COLUMN SECTION')
1760 3105 FORMAT(' ',5X,'COMMENTS : '//)
1761 END
1762
1763 C
1764 C
1765 C FUNCTION AGG(PU,PHIV,RFAC,FPC,PG,FY)
1766 C
1767 C IMPLICIT REAL(A-N,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
1768 C
1769 C AGG=PU/(PHIV*RFAC*(0.65+FPC+PG*(FY-0.65*FPC)))
1770 C
1771 C RETURN
1772 C END
1773
1774 C
1775 C *****
1776 C SUBROUTINE REINF1
1777 C
1778 C *THIS SUBROUTINE SELECTS REINFORCEMENT, CHECKS BARS SPACING
1779 C *AND REINFORCEMENT RATIO, COMPUTES MOMENT OF INERTIA OF
1780 C *REINFORCEMENT
1781 C *****
1782 C
1783 C
1784 C IMPLICIT REAL(A-N,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
1785 C COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEO,TPROB
1786 C COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,T1,TK,TBS(9),DB(9),
1787 C * PD,FL,ECU,PHAX,PMIN,EB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
1788 C * TSMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAXX
1789 C COMMON FPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
1790 C * SPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BDY,TOUT,TMODE
1791 C COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
1792 C COMMON TDMFC,AG,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
1793 C * Y(12),ISEX,K(12),ISEY,TMODAS,PGDES,EY,MCX,
1794 C * MAPPX,PUDESX,MUDESX,PBX
1795 C COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PBY,MOVREQ,MOXREQ,
1796 C * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
1797 C * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
1798 C COMMON TESTMO
1799 C COMMON NLC,JPOB,TREVIS,TOSSE,SPACE,PITCH,NSPACR
1800 C DIMENSION A(9)

```

```

1801 C
1802 REWIND 8
1803 IF(TOUT.NE.1) GO TO 3000
1804 IF(TESTM.NE.1) GO TO 3100
1805 WRITE(8,3101) CX,CY
1806 3101 FORMAT(' ',EX,'NEW TRIAL CONCRETE DIMENSION :',
1807 ' 2X,CX = ',F6.1,2X,'CY = ',F6.1,2X,'IN PROJECT UNITS')
1808 3100 IF(TMODE.EQ.1) WRITE(7,400)
1809 WRITE(8,400)
1810 3000 IF(TMODAS.EQ.1) GO TO 1
1811 IF(TMODAS.EQ.2) GO TO 75
1812 IF(TDESIG.EQ.1) GO TO 15
1813 C
1814 C SPECIFIED BAR SIZE AND REQUIRED TOTAL NO. OF BARS ARE ENTERED
1815 C
1816 J=TI
1817 TK=TI
1818 A(J)=(PRIME*DB(J)**2)/4.
1819 NMAX=NBS
1820 AS=NMAX*A(J)
1821 N=IFIX(NMAX/2.+0.51)
1822 GO TO 12
1823 1 NBS=NBS
1824 GO TO 18
1825 75 NBS=NBS
1826 A(J)=(PRIME*DB(J)**2)/4.
1827 N=IFIX(NMAX/2.+0.51)
1828 GO TO 12
1829 C
1830 C FOR DESIGN PROBLEM, SELECT POSSIBLE BAR SIZE AND NO. OF BARS
1831 C
1832 15 J=TI
1833 NBS=NBS
1834 18 A(J)=(PRIME*DB(J)**2)/4.
1835 IF(NB.NE.0) GO TO 82
1836 NMAX=IFIX(AS/A(J)+0.8)
1837 N=IFIX(NMAX/2.+0.51)
1838 NMAX=2*N
1839 IF(TBMAX.NE.0) GO TO 74
1840 C MAXIMUM PERMISSIBLE NUMBER OF BARS CONTROLLED
1841 C BY THE LEAST DIMENSION IN WHICH BARS IS LOCATED AT
1842 C
1843 GO TO(70,501,502,501),TRFACE
1844 502 IF(CX.GT.CY) GO TO 70
1845 501 DIM=CX
1846 GO TO 71
1847 70 DIM=CY
1848 71 IF(TCODE.EQ.1) GO TO 72
1849 IF(DIM.LE.18.) GO TO 73
1850 TBMAX=IFIX(DIM/8.+1.0)+4
1851 GO TO 74
1852 73 TBMAX=8
1853 GO TO 74
1854 72 IF(DIM.LE.400.) GO TO 73
1855 TBMAX=IFIX(DIM/200.+1.0)+4
1856 74 IF(NMAX.LE.TBMAX) GO TO 12
1857 IF(J.GE.TK) GO TO 380
1858 J=J+1
1859 GO TO 15
1860 62 NMAX=NBS
1861 N=IFIX(NMAX/2.+0.51)
1862 SB=AMAX1(1.5*DB(J),SB1)
1863 C
1864 C CHECK MIN. LATERAL REINFORCEMENT DIAMETER, DSS
1865 C
1866 CALL REIN1(TCODE,TYPE,DSMIN,TBS,J,DSS,TSSS1)
1867 DP=DCC+DSS+DB(J)+0.5
1868 GO TO(17,18,19,20),TRFACE
1869 C
1870 C CHECK NO. OF BARS, BARS SPACING AND REINFORCEMENT RATIO
1871 C
1872 17 N=CY
1873 GO TO 21
1874 18 N=CX
1875 21 G=N-2.-DP
1876 SB=(N-1.)*(DB(J)+SB)
1877 IF(NMAX.GE.4) GO TO 22
1878 67 IF(TOUT.NE.1) GO TO 30
1879 WRITE(8,200)
1880 WRITE(8,201) NMAX,TBS(J),SB,DSS
1881 IF(TMODE.EQ.1) GO TO 670
1882 GO TO 30
1883 670 WRITE(7,200)
1884 WRITE(7,201)NMAX,TBS(J),SB,DSS
1885 30 IF(TDESIG.EQ.1) GO TO 360
1886 TREINF=2
1887 IF(TMODE.EQ.1) GO TO 2001
1888 GO TO 131
1889 2001 WRITE(7,202)
1890 131 WRITE(8,202)
1891 RETURN
1892 C
1893 C FOR DESIGN PROBLEM, ASSIGN MIN. NO. OF BARS TO NMAX IF LESS
1894 C THAN THE MIN. ONE
1895 C
1896 360 GO TO(351,351,352,353),TRFACE
1897 351 NMAX=4
1898 N=IFIX(NMAX/2.+0.51)
1899 GO TO 22
1900 352 NX=2
1901 GO TO 60
1902 353 NMAX=6
1903 N=IFIX(NMAX/2.+0.51)
1904 GO TO 354
1905 C
1906 C CHECK BAR SPACING
1907 C
1908 22 IF(B.LE.G) GO TO 24
1909 68 IF(TOUT.NE.1) GO TO 33
1910 WRITE(8,200)
1911 WRITE(8,201) NMAX,TBS(J),SB,DSS
1912 IF(TMODE.EQ.1) GO TO 671
1913 GO TO 33
1914 671 WRITE(7,200)
1915 WRITE(7,201)NMAX,TBS(J),SB,DSS
1916 33 IF(NBS.NE.0) GO TO 360
1917 C
1918 C INCREASE BAR SIZE
1919 C
1920 IF(J.GE.TK) GO TO 360

```

```

1921      J=J+1
1922      GO TO 15
1923      24  AST=NMAX+A(J)
1924      PGDES=AST/AG
1925      C
1926      C CHECK REINFORCEMENT RATIO
1927      C
1928      IF(PGDES.LE.PG) GO TO 26
1929      360 GO TO(361,362,363,364,365),TDIMFG
1930      361 IF(TMODE.EQ.1) GO TO 367
1931      GO TO 368
1932      367 WRITE(7,369)
1933      368 WRITE(8,369)
1934      C
1935      C PROCEED TO NEXT PROBLEM RESULTING FROM LIMITING CONSTRAINTS
1936      C
1937      TREINF=2
1938      RETURN
1939      362 CV=CV+CVINCR
1940      GO TO 366
1941      363 CX=CX+CXINCR
1942      GO TO 366
1943      364 IF(CX.GE.CV) GO TO 3200
1944      CX=CX+CXINCR
1945      CV=CV/R
1946      CV=CVINCR+IFIX(CV/CVINCR)
1947      GO TO 3201
1948      3200 CV=CV+CVINCR
1949      CX=CV/R
1950      CX=CXINCR+IFIX(CX/CXINCR)
1951      3201 IF(TSHAPE.EQ.3) GO TO 370
1952      GO TO 366
1953      370 AG=(PRIME+CX**2)/4.
1954      GO TO 371
1955      365 IF(CX.GE.CV) GO TO 362
1956      GO TO 363
1957      366 AG=CX*CV
1958      371 AS=PMIN*AG
1959      IF(TOUT.NE.1) GO TO 15
1960      WRITE(8,3101) CX,CV
1961      GO TO 15
1962      25 SB=(G-(N-1.)*DB(J))/(N-1.)
1963      IF(TOUT.NE.1) GO TO 29
1964      IF(TMODE.EQ.1) GO TO 2004
1965      GO TO 134
1966      2004 WRITE(7,204) NMAX,TBS(J),SB,DSS
1967      134 WRITE(8,204) NMAX,TBS(J),SB,DSS
1968      GO TO 29
1969      C
1970      C FOR BARS IN BOTH FACES
1971      C
1972      19 IF(NBS.NE.0) GO TO 373
1973      C
1974      C COMPUTE NX AND NY
1975      C
1976      IF(ABS(NX2).GE.ABS(MY2)) GO TO 372
1977      NX=IFIX((NMAX+4.)/4.+0.51)
1978      NY=IFIX((NMAX-4.)/4.)
1979      GO TO 373
1980      372 NX=IFIX((NMAX+4.)/4.)
1981      NY=IFIX((NMAX-4.)/4.+0.51)
1982      373 IF(2*NX-4) 63,60,60
1983      63 IF(TOUT.NE.1) GO TO 30
1984      IF(TMODE.EQ.1) WRITE(7,300)
1985      WRITE(8,300)
1986      WRITE(8,301) NX,NY,TBS(J),SB,DSS
1987      IF(TMODE.EQ.1) WRITE(7,301)NX,NY,TBS(J),SB,DSS
1988      GO TO 30
1989      60 CX=CV-2.*DP
1990      BX=(NX-1.)*(DB(J)+SB)
1991      IF(BX.LE.CX) GO TO 32
1992      GO TO 374
1993      32 CV=CX-2.*DP
1994      BY=(NY+1.)*(DB(J)+SB)
1995      IF(BY.LE.CV) GO TO 66
1996      IF(TOUT.NE.1) GO TO 33
1997      WRITE(8,300)
1998      IF(TMODE.EQ.1) WRITE(7,300)
1999      WRITE(8,301) NX,NY,TBS(J),SB,DSS
2000      IF(TMODE.EQ.1) WRITE(7,301)NX,NY,TBS(J),SB,DSS
2001      GO TO 33
2002      66 ASX=2.*NX*A(J)
2003      ASY=2.*NY*A(J)
2004      AST=ASX*ASY
2005      PGDES=AST/AG
2006      IF(PGDES.GT.PG) GO TO 360
2007      SBX=(CX-(NX-1.)*DB(J))/(NX-1.)
2008      SBY=(CV-(NY+1.)*DB(J))/(NY+1.)
2009      NNX=2*NX
2010      NNV=2*NY
2011      IF(TOUT.NE.1) GO TO 29
2012      IF(TMODE.EQ.1) GO TO 2005
2013      GO TO 135
2014      2005 WRITE(7,206) NNX,NNV,ASX,ASY,SBX,SBY,TBS(J),DSS
2015      135 WRITE(8,206) NNX,NNV,ASX,ASY,SBX,SBY,TBS(J),DSS
2016      GO TO 29
2017      C
2018      C FOR BARS IN CIRCULAR CORE
2019      C
2020      20 DS=CX-2.*DP
2021      G=PRIME*DS
2022      B=NMAX*(DB(J)+SB)
2023      IF(NMAX.LT.G) GO TO 67
2024      264 IF(B.GT.G) GO TO 68
2025      AST=NMAX*A(J)
2026      PGDES=AST/AG
2027      IF(PGDES.GT.PG) GO TO 360
2028      SSB=C/NMAX
2029      SS=SSB-DB(J)
2030      C
2031      C COMPUTE ANGLE BETWEEN BARS
2032      C
2033      HETA=SSB/(0.5*DS)
2034      IF(TOUT.NE.1) GO TO 709
2035      IF(TMODE.EQ.1) GO TO 2007
2036      GO TO 137
2037      2007 WRITE(7,204) NMAX,TBS(J),SB,DSS
2038      WRITE(7,208) HETA,DS
2039      137 WRITE(8,204) NMAX,TBS(J),SB,DSS
2040      WRITE(8,208) HETA,DS

```

```

2041 C
2042 C DETERMINE ISEX IN THIS SUBROUTINE FOR CONVENIENT DATA NEEDED
2043 C
2044 709 YSUM=0.
2045 TL=IFIX((NMAX-4.)/4.+0.51)
2046 DO 35 NI=1,TL
2047 STA=NI-META
2048 Y(NI)=0.5+DS*SIN(STA)
2049 YSUM=YSUM+Y(NI)**2
2050 C
2051 C DETERMINE WHETHER HAVING CENTER BARS OR NOT
2052 C
2053 NO=NMAX/2
2054 NOD=N0/2
2055 IF(N0/2.-N00.LE.0.0001) GO TO 58
2056 NM=0
2057 GO TO 57
2058 58 NM=1
2059 ISEX=(AST*(0.5+NM*DS**2+4.*YSUM))/NMAX
2060 GO TO 36
2061 29 GAMMAY=(CY-2.*DP)/CY
2062 GO TO(37,38,39,38),TRFACE
2063 37 YSUM=0.
2064 TL=IFIX((NMAX-1.)/4.+0.51)
2065 DO 40 NI=1,TL
2066 Y(NI)=0.5+GAMMAY*CY-(NI-1.)*(DB(J)+SB)
2067 YSUM=YSUM+Y(NI)**2
2068 ISEX=(4.*AST+YSUM)/NMAX
2069 GO TO 36
2070 38 ISEX=0.25+AST+GAMMAY**2*CY**2
2071 GO TO 36
2072 39 YSUM=0.
2073 TL=IFIX((2.*NX-1.)/4.+0.51)
2074 DO 41 NI=1,TL
2075 Y(NI)=0.5+GAMMAY*CY-(NI-1.)*(SBX+DB(J))
2076 YSUM=YSUM+Y(NI)**2
2077 ISEX=0.25+ASY+GAMMAY**2*CY**2+(2.*ASX+YSUM)/NX
2078 IF(TSEND.EQ.1) GO TO 47
2079 C
2080 C ALSO DETERMINE ISEY, ONLY USED FOR BIAxIAL BENDING
2081 C
2082 IF(TRFACE.EQ.4) GO TO 43
2083 GAMMAX=(CX-2.*DP)/CX
2084 GO TO(44,45,46,43),TRFACE
2085 44 ISEY=0.25+AST+GAMMAX**2*CX**2
2086 GO TO 47
2087 45 XSUM=0.
2088 TL=IFIX((NMAX-1.)/4.+0.51)
2089 DO 48 NI=1,TL
2090 X(NI)=0.5+GAMMAX*CX-(NI-1.)*(DB(J)+SB)
2091 XSUM=XSUM+X(NI)**2
2092 ISEY=(4.*AST+XSUM)/NMAX
2093 GO TO 47
2094 46 XSUM=0.
2095 TL=IFIX((2.*NY-1.)/4.+0.51)
2096 IF(TL.LE.0) TL=1
2097 DO 49 NI=1,TL
2098 X(NI)=0.5+GAMMAX*CX-DB(J)-SEY*(NI-1.)*(SBY+DB(J))
2099 XSUM=XSUM+X(NI)**2
2100 IF(NY.EQ.0) GO TO 999
2101 ISEY=0.25+ASX+GAMMAX**2*CX**2+(2.*ASY+XSUM)/NY
2102 GO TO 47
2103 999 ISEY=0.25+ASX+GAMMAX**2*CX**2
2104 GO TO 47
2105 43 XSUM=0.
2106 TL=IFIX((NMAX-3.)/4.+0.51)
2107 DO 50 NI=1,TL
2108 CTA=NI-META
2109 X(NI)=0.5+DS*COS(CTA)
2110 XSUM=XSUM+X(NI)**2
2111 ISEY=(AST*(0.5+DS**2+4.*XSUM))/NMAX
2112 C
2113 47 IF(TOUT.NE.1) GO TO 52
2114 IF(TMODE.EQ.1) GO TO 2009
2115 GO TO 138
2116 2009 WRITE(7,211) PCDES
2117 138 WRITE(8,211) PCDES
2118 C
2119 C RETURN TO MAIN ROUTINE AND CONTINUE EXECUTION
2120 C
2121 52 TREINF=1
2122 RETURN
2123 C
2124 C *****FORMAT STATEMENTS*****
2125 C
2126 400 FORMAT(' ', 'SELECTED REINFORCEMENT DETAILS :')
2127 200 FORMAT(' ', 1X, ' NMAX', 2X, 'BS(J)', 6X,
2128 ' ', 5X, 'DSS')
2129 201 FORMAT(15, 2X, 15, 2F10.3/)
2130 202 FORMAT(' ', 5X, 'THE CODE',
2131 ' ', ' REQUIREMENT FOR MINIMUM NUMBER', 6X, 'OF LONGITUDINAL',
2132 ' ', ' BARS IS NOT MEET.')
```

```

2133 204 FORMAT(' ', 2X, 'NUMBER',
2134 ' ', ' OF LONGITUDINAL BARS= ', 14/3X, 'BAR SIZE= ', 14/,
2135 ' ', 3X, 'CLEAR BAR SPACING= ', F6.2, 1X, 'IN/MM'/3X,
2136 ' ', 'LATERAL TIES DIAM.= ', F6.3, 1X, 'IN/MM')
2137 205 FORMAT(' ', 2X, 'X-FACE=YES', 37X, 'Y-FACE=YES', 17X, 'NUMBER OF',
2138 ' ', ' LONGITUDINAL BARS= ', 14, 6X, 'NUMBER OF LONGITUDINAL BARS= ',
2139 ' ', 14/3X, 'STEEL AREA= ', F12.1, 1X, 'SQ. IN/SQ. MM', 17X, 'STEEL AREA= ',
2140 ' ', F12.1, 1X, 'SQ. IN/SQ. MM', 17X, 'CLEAR BAR SPACING= ', F6.2, 1X, 'IN/MM',
2141 ' ', 22X, 'CLEAR BAR SPACING= ', F6.2, 1X, 'IN/MM/10X, ' ',
2142 ' ', 'ADDITIONAL INFORMATION =====/10X, 'BAR SIZE= ', 14/10X,
2143 ' ', 'LATERAL TIES DIAMETER= ', F6.3, 1X, 'IN/MM'/)
2144 206 FORMAT(' ', 2X, 'ANGLE BETWEEN BARS= ', F7.3, 3X, 'RADIAN'/3X,
2145 ' ', 'DIAMETER OF LONGITUDINAL STEEL CIRCLE= ', F7.2, 1X, 'IN/MM'/)
2146 211 FORMAT(' ', 2X, 'PG DESIGN= ', F6.4/)
2147 C
2148 300 FORMAT(' ', 2X, 'NX', 5X, 'NY', 4X, 'BS(J)', 6X, 'SB', 6X,
2149 ' ', 'DSS')
2150 301 FORMAT(15, 2X, 15, 2X, 15, 2F10.3/)
2151 302 FORMAT(' ', 5X, 'DESIGN CANNOT BE COMPLETED COMPLETED'/6X,
2152 ' ', 'CONSISTENT WITH THE ENTERED RESTRAINTS'/)
2153 END
2154 C
2155 C =====
2156 C
2157 C SUBROUTINE REIN1(TCODE,TYPE,DSMIN,TBS,J,DSS,TOSSS)
2158 C
2159 C THIS SUBROUTINE CHECKS MINIMUM LATERAL REINFORCEMENT DIAMETER
2160 C

```

```

2161 C *****
2162 C
2163     IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
2164     DIMENSION TBS(8)
2165 C
2166     IF(TYPE.EQ.2) GO TO 10
2167     IF(TCODE.EQ.1) GO TO 11
2168     IF(TBS(J).LE.10) GO TO 13
2169     IF(DSS.GT.0.5) GO TO 13
2170     DSS=0.5
2171     TDSS=4
2172     GO TO 13
2173 11 IF(TBS(J).LE.30) GO TO 14
2174     IF(DSS.GT.11.3) GO TO 13
2175     DSS=11.3
2176     TDSS=10
2177     GO TO 13
2178 14 IF(DSS.GT.0.30+TBS(J)) GO TO 13
2179     DSS=11.3
2180     TDSS=10
2181     GO TO 13
2182 10 IF(DSS.GT.DSMIN) GO TO 13
2183     DSS=DSMIN
2184     IF(TCODE.EQ.1) TDSS=6
2185     IF(TCODE.EQ.2) TDSS=4
2186 13 RETURN
2187 END
2188 C
2189 C *****
2190 C *
2191     SUBROUTINE SLEND1
2192 C *
2193 C *THIS SUBROUTINE CHECKS SLENDERNESS EFFECTS ABOUT X-AXIS
2194 C *
2195 C *****
2196 C
2197     IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
2198     COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TSEND,TSLEN2,TADEQ,TPROB
2199     COMMON TCODE,WC,EC,DEC,DSS,CXINCR,CVINCR,TJ,TX,TBS(8),DS(8),
2200     * PD,PL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
2201     * TBMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAX,
2202     COMMON FPC,FY,FSY,EC,B1,TBRAC,KX,LUX,TBRAC,KY,LUY,SPU,
2203     * SPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BDY,TOUT,TMODE
2204     COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
2205     COMMON TDIMFG,AG,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
2206     * Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MXX,
2207     * MAPPX,PUDESX,MUDESX,PBX
2208     COMMON EX,NCY,MAPPY,PUDESX,MUDESY,PSY,MOYREQ,MOXREQ,
2209     * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
2210     * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
2211     COMMON TESTMD
2212     COMMON NLC,JPOB,TREVIS,TDSS,SPACE,PITCH,NSPACR
2213     DIMENSION A(9)
2214 C
2215     IF(TOUT.NE.1) GO TO 3000
2216     IF(TMODE.EQ.1) WRITE(7,500)
2217     WRITE(8,500)
2218 500 FORMAT(' ','CHECK SLENDERNESS EFFECTS BENDING',
2219     * ' ABOUT X-AXIS : '//)
2220 C
2221 C COMPUTE MOMENT OF INERTIA OF CROSS-SECTION AND RADIUS OF GYRATION
2222 C AND SLENDERNESS EFFECTS ABOUT X-AXIS
2223 C
2224 3000 IGX=CG*AG*CY**2
2225     RXT=IGX/AG
2226     RX=SQRT(RXT)
2227     SX=KX*LUX/RX
2228     IF(TBRAC.EQ.1) GO TO 10
2229     IF(SX.LE.22.) GO TO 11
2230     IF(SPU.LE.0. AND.SPC.LE.0.) GO TO 12
2231     IF(PHIV*SPC.LE.SPU) GO TO 350
2232     CMX=1.0
2233     DELTAX=CMX/(1.-(SPU/(PHIV*SPC)))
2234     IF(DELTAX.LE.1.0) DELTAX=1.0
2235     GO TO 13
2236 11 DELTAX=1.0
2237     GO TO 50
2238 C
2239 C PROCEED TO NEXT PROBLEM RESULTING FROM INCOMPLETE INPUT PARAMETERS
2240 C
2241 12 TSLEN1=2
2242     IF(TMODE.EQ.1) GO TO 2001
2243     GO TO 131
2244 2001 WRITE(7,201)
2245     131 WRITE(8,201)
2246     RETURN
2247 350 TSLEN1=2
2248     IF(TMODE.EQ.1) WRITE(7,351)
2249     WRITE(8,351)
2250     RETURN
2251 10 IF(ABS(MX2).LE.0.) GO TO 610
2252     MX12=ABS(MX1/MX2)
2253     GO TO 611
2254 610 MX12=1.0
2255 611 IF(SX.LT.34.-12.*MX12) GO TO 14
2256     IF(TOUT.NE.1) GO TO 3001
2257     IF(TMODE.EQ.1) GO TO 2002
2258     GO TO 132
2259 2002 WRITE(7,202)
2260     132 WRITE(8,202)
2261 C
2262 C COMPUTE EI VALUE AND CRITICAL LOAD
2263 C
2264 3001 EIX=(0.2+EC*IGX+ES*ISEX)/(1.+BDX)
2265     PCX=(PRIME**2+EIX)/(KX*LUX)**2
2266     IF((PHIV*PCX).GT.PU) GO TO 40
2267     IF(TOUT.NE.1) GO TO 3002
2268     IF(TMODE.EQ.1) GO TO 2003
2269     GO TO 133
2270 3002 WRITE(7,400)
2271     133 WRITE(8,400)
2272 C
2273 C REDESIGN CONCRETE SECTION DUE TO STABILITY PROBLEM BY USING
2274 C SEVERAL DESIGN CODE EQUATIONS AND THE FORMULA IG=CG*AG*H
2275 C
2276 3002 IF(TDIMFG.EQ.1) GO TO 41
2277 C
2278 C SET PCX=PU/PHIV AND USE THE CONSERVATIVE CODE
2279 C EQUATION OF EI FOR SLENDER COLUMN
2280 C

```

```

2281      IGX=(PU/PHIV)*(KX=LUX/PRIME)**2*(1.+SDX)/(O.4*EC)
2282      GO TO (41,42,43,44,45),TDIMFG
2283      41      IF(TMODE.EQ.1) GO TO 2004
2284      GO TO 134
2285      2004 WRITE(7,401)
2286      134 WRITE(8,401)
2287      C
2288      C PROCEED TO NEXT PPOBLEM RESULTING FROM CONSTANT DIMENSIONS
2289      C
2290      TLEN1=2
2291      RETURN
2292      C
2293      C COMPARING NEW AREA WITH THE OLD ONE
2294      C
2295      42      AGNEW=(IGX*CX**2/CG)**0.333333
2296      IF(AGNEW.GT.AG) GO TO 5000
2297      C
2298      C IF OLD AREA > NEW COMPUTED ONE, INC. DIMENSION W.R.T.
2299      C ENTERED DIMENSIONS
2300      C
2301      CY=CY+CVINCR
2302      GO TO 61
2303      5000 AG=AGNEW
2304      CY=AG/CX
2305      CY=CYINCR+IFIX(CY/CVINCR+0.999)
2306      IF(CY.LE.MINDIM) GO TO 60
2307      GO TO 61
2308      60      CY=MINDIM
2309      IF(TOUT.NE.1) GO TO 61
2310      IF(TMODE.EQ.1) GO TO 2005
2311      GO TO 135
2312      2005 WRITE(7,300)
2313      135 WRITE(8,300)
2314      61      AG=CX*CY
2315      GO TO 60
2316      43      AGNEW=IGX/(CG*CY**2)
2317      IF(AGNEW.GT.AG) GO TO 5001
2318      CX=CX+CXINCR
2319      GO TO 62
2320      5001 AG=AGNEW
2321      CX=AG/CY
2322      CX=CXINCR+IFIX(CX/CXINCR+0.999)
2323      IF(CX.LE.MINDIM) GO TO 62
2324      GO TO 63
2325      62      CX=MINDIM
2326      IF(TOUT.NE.1) GO TO 63
2327      IF(TMODE.EQ.1) GO TO 2006
2328      GO TO 136
2329      2006 WRITE(7,301)
2330      136 WRITE(8,301)
2331      63      AG=CX*CY
2332      GO TO 60
2333      44      IF(TSHAPE.EQ.3) GO TO 47
2334      A1=R*IGX/CG
2335      AGNEW=SQRT(A1)
2336      IF(AGNEW.GT.AG) GO TO 5002
2337      IF(CX.GT.CY) GO TO 5003
2338      CX=CX+CXINCR
2339      GO TO 65
2340      5003 CY=CY+CVINCR
2341      CX=CY*R
2342      CX=CXINCR+IFIX(CX/CXINCR+0.999)
2343      GO TO 67
2344      5002 AG=AGNEW
2345      A2=AG*R
2346      CX=SQRT(A2)
2347      CX=CXINCR+IFIX(CX/CXINCR+0.999)
2348      IF(CX.LE.MINDIM) GO TO 64
2349      GO TO 66
2350      64      CX=MINDIM
2351      IF(TOUT.NE.1) GO TO 66
2352      IF(TMODE.EQ.1) GO TO 2007
2353      GO TO 137
2354      2007 WRITE(7,301)
2355      137 WRITE(8,301)
2356      65      CY=CX/R
2357      CY=CYINCR+IFIX(CY/CVINCR+0.999)
2358      IF(CY.LE.MINDIM) GO TO 66
2359      GO TO 67
2360      66      CY=MINDIM
2361      IF(TOUT.NE.1) GO TO 67
2362      IF(TMODE.EQ.1) GO TO 2008
2363      GO TO 138
2364      2008 WRITE(7,300)
2365      138 WRITE(8,300)
2366      67      AG=CX*CY
2367      GO TO 60
2368      47      AG1=(PRIME*IGX)/(4.*CG)
2369      AGNEW=SQRT(AG1)
2370      IF(AGNEW.GT.AG) GO TO 5004
2371      CX=CX+CXINCR
2372      GO TO 69
2373      5004 AG=AGNEW
2374      A1=4.*AG/PRIME
2375      CX=SQRT(A1)
2376      CX=CXINCR+IFIX(CX/CXINCR+0.999)
2377      IF(CX.LE.MINDIM) GO TO 68
2378      GO TO 69
2379      68      CX=MINDIM
2380      IF(TOUT.NE.1) GO TO 69
2381      IF(TMODE.EQ.1) GO TO 2009
2382      GO TO 139
2383      2009 WRITE(7,301)
2384      139 WRITE(8,301)
2385      69      CY=CX
2386      AG=(PRIME*CX**2)/4.
2387      GO TO 60
2388      45      IF(TSEND.EQ.1) GO TO 81
2389      GO TO 82
2390      81      AGNEW=IGX/(CG*CY**2)
2391      IF(AGNEW.GT.AG) GO TO 5005
2392      CX=CX+CXINCR
2393      GO TO 73
2394      5005 AG=AGNEW
2395      CX=AG/CY
2396      CX=CXINCR+IFIX(CX/CXINCR+0.999)
2397      IF(CX.LE.MINDIM) GO TO 87
2398      GO TO 73
2399      87      CX=MINDIM
2400      IF(TOUT.NE.1) GO TO 73

```

```

2401      IF(TMODE.EQ.1) GO TO 2010
2402      GO TO 140
2403      2010 WRITE(7,301)
2404      140 WRITE(8,301)
2405      GO TO 73
2406      82 A1=R+IX/CG
2407      AGNEW=SQRT(A1)
2408      IF(AGNEW.GT.AG) GO TO 5006
2409      IF(CX.GT.CY) GO TO 5007
2410      CX=CX+CXINCR
2411      GO TO 71
2412      5007 CY=CY+CYINCR
2413      CX=CY+R
2414      CX=CXINCR+IFIX(CX/CXINCR+0.999)
2415      GO TO 73
2416      5006 AG=AGNEW
2417      A2=R+AG
2418      CX=SQRT(A2)
2419      CX=CXINCR+IFIX(CX/CXINCR+0.999)
2420      IF(CX.LE.MINDIM) GO TO 70
2421      GO TO 71
2422      70 CX=MINDIM
2423      IF(TOUT.NE.1) GO TO 71
2424      IF(TMODE.EQ.1) GO TO 2011
2425      GO TO 141
2426      2011 WRITE(7,301)
2427      141 WRITE(8,301)
2428      71 CX=CX/R
2429      CY=CYINCR+IFIX(CY/CYINCR+0.999)
2430      IF(CY.LE.MINDIM) GO TO 72
2431      GO TO 73
2432      72 CY=MINDIM
2433      IF(TOUT.NE.1) GO TO 73
2434      IF(TMODE.EQ.1) GO TO 2012
2435      GO TO 142
2436      2012 WRITE(7,300)
2437      142 WRITE(8,300)
2438      73 AG=CY
2439      80 TMODE=0
2440      AS=PMIN+AG
2441      IF(TOUT.NE.1) GO TO 3005
2442      IF(TSHAPE.EQ.3) GO TO 2880
2443      IF(TMODE.EQ.1) GO TO 2014
2444      GO TO 144
2445      2014 WRITE(7,403) CX,CY
2446      144 WRITE(8,403) CX,CY
2447      GO TO 3005
2448      2880 IF(TMODE.EQ.1) WRITE(7,2881) CX
2449      2881 FORMAT(' ','NEW TRIAL CONCRETE DIM. :'/BX,'DIAMETER =',
2450      ' FE. 1,1X,'IN/MM'/)
2451      WRITE(8,2881) CX
2452      C BACK TO SUBROUTINE REINF1 AND REITERATE THE STEPS
2453      C
2454      3005 TLEN1=3
2455      RETURN
2456      C
2457      C DETERMINE THE GOVERNING ECCENTRICITY, COMPUTE END MOMENT MUX AND
2458      C MAGNIFICATION FACTOR DELTAX FOR LONG COLUMNS
2459      C
2460      40 IF(ABS(MX2).LE.0.1) GO TO 15
2461      B=MX1/MX2
2462      GO TO 20
2463      15 B=1.0
2464      20 IF(TCODE.EQ.1) GO TO 18
2465      EMINY=0.8+0.03*CY
2466      GO TO 17
2467      18 IF(TYPE.EQ.1) GO TO 18
2468      F=0.05*CY
2469      GO TO 18
2470      18 F=0.1*CY
2471      18 EMINY=AMAX1(F,EMIN)
2472      18 EY=ABS(MX2)/PU
2473      17 IF(EY.LE.EMINY) GO TO 510
2474      GO TO 511
2475      510 EY=EMINY
2476      IF(TCODE.EQ.1) MUXX=(PU*EY)/1000000.
2477      IF(TCODE.EQ.2) MUXX=(PU*EY)/12000.
2478      IF(TMODE.EQ.1) GO TO 550
2479      GO TO 551
2480      550 WRITE(7,512) MUXX
2481      551 WRITE(8,512) MUXX
2482      511 MUX=PU*EY
2483      CMX=0.8+0.4*B
2484      IF(CMX.LT.0.4) CMX=0.4
2485      DELTAX=CMX/(1.-PU/(PHIV*PCX))
2486      C IF DELTAX IS LESS THAN 1.0, IT WOULD BE DEFAULTS TO 1.0
2487      IF(DELTA.LE.1.0) DELTAX=1.0
2488      GO TO 513
2489      C
2490      C DETERMINE THE GOVERNING ECCENTRICITY AND COMPUTE END MOMENT MUX
2491      C FOR SHORT COLUMNS
2492      14 IF(TOUT.NE.1) GO TO 3007
2493      IF(TMODE.EQ.1) GO TO 2015
2494      GO TO 145
2495      2015 WRITE(7,203)
2496      145 WRITE(8,203)
2497      3007 DELTAX=1.0
2498      GO TO 50
2499      13 IF(TCODE.EQ.1) GO TO 50
2500      EMINY=0.8+0.03*CY
2501      GO TO 51
2502      50 IF(TYPE.EQ.1) GO TO 52
2503      F=0.05*CY
2504      GO TO 53
2505      52 F=0.1*CY
2506      53 EMINY=AMAX1(F,EMIN)
2507      51 EY=ABS(MX2)/PU
2508      IF(EY.LE.EMINY) GO TO 514
2509      GO TO 515
2510      514 EY=EMINY
2511      MUX=PU*EY
2512      IF(TCODE.EQ.1) MUXX=MUX/1000000.
2513      IF(TCODE.EQ.2) MUXX=MUX/12000.
2514      IF(TMODE.EQ.1) GO TO 552
2515      GO TO 553
2516      552 WRITE(7,512) MUXX
2517      553 WRITE(8,512) MUXX
2518      515 MUX=PU*EY
2519      513 IF(DELTA.LE.1.001) GO TO 520
2520

```

```

2521      WRITE(8,204)DELTA
2522      IF(TMODE.EQ.1) WRITE(7,205)DELTA
2523
2524      C COMPUTE MAGNIFIED MOMENT THAT APPLIED MOMENT
2525      C
2526      620 MCX=DELTA*MUX
2527      EY=MCX/PU
2528      MMINX=PU*EMINY
2529      MAPPX=MCX
2530
2531      C RETURN TO MAIN ROUTINE AND CONTINUE EXECUTION
2532      C
2533      TLEN1=1
2534      RETURN
2535
2536      C
2537      C *****FORMAT STATEMENTS*****
2538
2539      201 FORMAT(' ',SX,'THIS IS A LONG UNBRACED FRAME. MORE INPUT',
2540      *'/SX,PARAMETERS WILL BE REQUIRED TO COMPUTE'/SX,
2541      *'MOMENT MAGNIFIER ABOUT X-AXIS ONLY'/)
2542      202 FORMAT(' ',SX,'SLENDERNESS EFFECT MUST BE CONSIDERED'/SX,
2543      *'BY MAGNIFYING MOMENT ABOUT X-AXIS'/)
2544      203 FORMAT(' ',SX,'SLENDERNESS EFFECT MAY BE NEGLECTED'/)
2545      400 FORMAT(' ',SX,'THE CROSS SECTION OF COLUMN IS TOO SMALL',
2546      *'/SX,TO PROVIDE STABILITY'/)
2547      401 FORMAT(' ',SX,'COLUMN DIMENSIONS ARE CONSTANT AND THE'/SX,
2548      *'SECTION IS NOT ADEQUATE TO PROVIDE STABILITY'/)
2549      402 FORMAT(' ',SX,'NEW TRIAL CONCRETE DIM. : '/SX,'CX '= ',PS.1,1X,
2550      *'IN/MM',SX,'CY '= ',PS.1,1X,'IN/MM'/)
2551      300 FORMAT(' ',SX,'MINIMUM DIMENSION CY IS USED'/)
2552      301 FORMAT(' ',SX,'MINIMUM DIMENSION CX IS USED'/)
2553      512 FORMAT(' ',SX,'MINIMUM ECCENTRICITY ABOUT X-AXIS GOVERNS',
2554      *'/SX,DESIGN MOMENT ABOUT X-AXIS = ',PS.0/SX,'IN PROJECT',
2555      *'UNIT'/)
2556      204 FORMAT(' ',SX,'MAGNIFICATION FACTOR FOR MOMENT ABOUT',
2557      *'X-AXIS = ',PS.2/)
2558      205 FORMAT(' ',SX,'COMMENTS : /// ',SX,'MAGNIFICATION FACTOR',
2559      *' FOR MOMENT ABOUT X-AXIS = ',PS.2/)
2560      351 FORMAT(' ',SX,'SUM OF CRITICAL LOAD IS LESS THAN'/SX,
2561      *'SUM OF NOMINAL AXIAL LOAD'/)
2562
2563      C
2564      C END
2565
2566      C *****
2567      C SUBROUTINE CAPAC1
2568      C
2569      C THIS SUBROUTINE INVESTIGATES STRENGTH CAPACITY IN
2570      C COMPRESSION CONTROLS REGIONS ( PNK > PSX )
2571      C
2572      C *****
2573      C
2574      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
2575      COMMON PG,ESMAX,TYPE,TREINF,TLEN1,TBEND,TLEN2,TADEQ,TPROB
2576      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9),
2577      *PD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
2578      *TBMX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAXX
2579      COMMON FPC,FY,FSY,EC,B1,TBRAC,KX,LUX,TBRAC,KY,LUY,SPU,
2580      *SPC,PD,PL,PU,MK1,MK2,BOK,MV1,MV2,BDV,TOUT,TMODE
2581      COMMON TSHAPE,CX,CY,R,TDESIS,TRFACE,MBS,MWAX
2582      COMMON TDMFPC,AG,AS,CG,SG,DS,AST,MK,MV,SBX,SBY,DS,
2583      *Y(12),ISEX,K(12),ISEY,TMODAS,PGDES,EY,MCX,
2584      *MAPPX,PUDEX,MUDEX,PSX
2585      COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PSY,MOYREQ,MOKREQ,
2586      *YC,YB,ECS(12),ETS(12),CS(12),FT(12),
2587      *XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
2588      COMMON TESTMO
2589      COMMON NLC,JPOS,TREVIS,TDSS,SPACE,PITCH,NSPACR
2590      DIMENSION A(3)
2591
2592      C
2593      IF(TOUT.NE.1) GO TO 4000
2594      IF(TMODE.EQ.1) WRITE(7,300)
2595      WRITE(8,300)
2596      300 FORMAT(' ',SX,'CAPACITY CHECK FOR BENDING ABOUT X-AXIS : '//)
2597      4000 IF(TRFACE.EQ.2) GO TO 10
2598
2599      C
2600      C COMPUTE STRENGTH CAPACITIES WITH BARS IN LATERAL FACES OR
2601      C IN BOTH FACES OR IN CIRCULAR CORE
2602      C
2603      C
2604      2600      CALL CAP12
2605      GO TO 11
2606
2607      C
2608      C COMPUTE STRENGTH CAPACITIES WITH BARS IN END FACES ONLY
2609      C
2610      C
2611      10      CALL CAP11
2612      11      IF(TOUT.NE.1) RETURN
2613      PUDX=PUDEX/1000.
2614      MPBX=PSX/1000.
2615      IF(TCODE.EQ.1) GO TO 1990
2616      MUDX=MUDEX/12000.
2617      GO TO 1991
2618      1990 MUDX=MUDEX/1000000.
2619      1991 WRITE(8,200) YC,YB,PUDX,MUDX,MPBX
2620      IF(TMODE.EQ.1) WRITE(7,200)YC,YB,PUDX,MUDX,MPBX
2621      RETURN
2622
2623      C
2624      C
2625      200 FORMAT(' ',2X,'LOCATION OF NEUTRAL AXIS // X-AXIS= ',
2626      *'PS.1,1X,'IN/MM'/3X,'NEUTRAL AXIS AT BALANCED CONDITION= ',
2627      *'PS.1,1X,'IN/MM'/3X,'DESIGN AXIAL= ',
2628      *'LOAD WITH ONLY X-AXIS BENDING= ',PS.0,1X,'KIPS/KN'/3X,
2629      *'DESIGN MOMENT ABOUT X-AXIS ONLY= ',PS.0,1X,'KIP-FT/KN-M'/3X,
2630      *'LOAD AT BALANCED STRAIN CONDITION= ',PS.0,1X,'KIPS/KN'/)
2631
2632      C
2633      C END
2634
2635      C *****
2636      C SUBROUTINE CAP11
2637      C
2638      C THIS SUBROUTINE COMPUTES LOCATION OF NEUTRAL AXIS AND
2639      C DESIGN STRENGTH AT COMPRESSION CONTROL REGION AND AT
2640      C BALANCED CONDITIONS FOR BARS DISTRIBUTED IN END FACES ONLY
2641      C
2642      C *****
2643      C
2644      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
2645      COMMON PG,ESMAX,TYPE,TREINF,TLEN1,TBEND,TLEN2,TADEQ,TPROB
2646      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9),
2647      *PD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
2648      *TBMX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAXX

```

```

2641      COMMON FPC,FY,FSY,EC,B1,TNBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
2642      * SPC,PD,PL,PU,MK1,MK2,SDX,MV1,MV2,BDY,TOUT,TMODE
2643      COMMON TSHAPE,CK,CY,R,TDESIG,TRFACE,NBS,NMAX
2644      COMMON TDIMFG,AC,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
2645      * Y(12),ISEX,K(12),ISEY,TMODAS,PGDES,EY,MCK,
2646      * MAPPX,PUDEX,MUDEX,PKX
2647      COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PSY,MOYREO,MOKREO,
2648      * VC,YB,ECS(12),ETS(12),CS(12),FT(12),
2649      * KC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
2650      COMMON TESTMO
2651      COMMON NLC,JPOB,TREVIS,TOSSE,SPACE,PITCH,NSPACR
2652      DIMENSION A(9)
2653
2654      C
2655      D=CY-DP
2656
2657      C
2658      C COMPUTE COMPRESSION FORCE IN STEEL
2659      C
2660      CSS=0.5*AST*(FY-0.55*FPC)
2661
2662      C
2663      C USING HALPING INTERVAL TECHNIQUE TO FIND LOCATION OF NEUTRAL
2664      C AXIS
2665      C
2666      VL=(ECU*D)/(ECU+ESMAX)
2667      YU=CY/B1
2668      10  YC=(VL+YU)/2.
2669
2670      C
2671      C COMPUTE FORCE IN TENSION REINFORCEMENT AND COMPRESSION FORCE
2672      C IN CONCRETE
2673      C
2674      FTY=0.5*AST*ECU*ES*((D-YC)/YC)
2675      CC=0.55*FPC*B1*YC*CK
2676      YV=YU-VL
2677      IF(TCODE.EQ.1) DIFFER=5
2678      IF(TCODE.EQ.2) DIFFER=0.2
2679      IF(ABS(YV).LT.DIFFER) GO TO 11
2680      FUN=F11(YC,FTY,CSS,CC,D,CY,EY,DP,B1)
2681      IF(FUN) 12,11,14
2682
2683      12  YU=YC
2684      GO TO 10
2685
2686      14  VL=YC
2687      GO TO 10
2688
2689      11  IF(B1*YC.GT.D) GO TO 20
2690      GO TO 21
2691
2692      C
2693      C THE EFFECT OF DISPLACED CONCRETE IS INCLUDED
2694      C
2695      20  FTY=FTY+0.5*AST*0.55*FPC
2696      21  PNX=CC+CSS*FTY
2697      MNX=FTY*(D-0.5*CY)+CSS*(0.5*CY-DP)+CC*(0.5*CY-0.5*B1*YC)
2698      PUDEX=PNX/PNX
2699      MUDEX=PNX/MNX
2700
2701      C
2702      YB=(ECU*D)/(FY/ES+ECU)
2703      PSY=PNX/0.55*FPC*(B1-YB*CK-0.5*AST)
2704      RETURN
2705      END
2706
2707      C
2708      REAL FUNCTION F11(YC,FTY,CSS,CC,D,CY,EY,DP,B1)
2709
2710      C
2711      C TAKING MOMENTS OF THE INTERNAL FORCES ABOUT APPLIED AXIAL LOAD
2712      C AND WRITING AN EQUATION IN TERMS OF FUNCTION OF UNKNOWN NEUTRAL
2713      C AXIS
2714      C
2715      F11=FTY*(D-(0.5*CY-EY))+CSS*(0.5*CY-EY-DP)+CC*(0.5*CY-EY
2716      * -0.5*B1*YC)
2717      RETURN
2718      END
2719
2720      C
2721      C *****
2722      C SUBROUTINE CAPA12
2723      C
2724      C *THIS SUBROUTINE COMPUTES LOCATION OF NEUTRAL AXIS AND
2725      C *DESIGN STRENGTH AT THE COMPRESSION CONTROL REGION AND
2726      C *AT BALANCED CONDITION FOR BARS DISTRIBUTED IN LATERAL
2727      C *FACES OR IN BOTH FACES OR IN CIRCULAR CORE
2728      C
2729      C *****
2730
2731      C
2732      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
2733      COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEO,TPROB
2734      COMMON TCODE,WC,ES,DCC,DSS,CKINCR,CVINCR,TI,TK,TSS(9),DS(9),
2735      * PD,PL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,BSMIN,SSMIN,SSMAX,
2736      * TBMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAX
2737      COMMON FPC,FY,FSY,EC,B1,TNBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
2738      * SPC,PD,PL,PU,MK1,MK2,SDX,MV1,MV2,BDY,TOUT,TMODE
2739      COMMON TSHAPE,CK,CY,R,TDESIG,TRFACE,NBS,NMAX
2740      COMMON TDIMFG,AC,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
2741      * Y(12),ISEX,K(12),ISEY,TMODAS,PGDES,EY,MCK,
2742      * MAPPX,PUDEX,MUDEX,PKX
2743      COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PSY,MOYREO,MOKREO,
2744      * VC,YB,ECS(12),ETS(12),CS(12),FT(12),
2745      * KC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
2746      COMMON TESTMO
2747      COMMON NLC,JPOB,TREVIS,TOSSE,SPACE,PITCH,NSPACR
2748      DIMENSION A(9)
2749
2750      C
2751      D=CY-DP
2752
2753      C
2754      C DETERMINE BALANCED AXIAL LOAD BY DIRECT APPLICATION OF STATICS
2755      C
2756      YB=(ECU*D)/(ECU+ESMAX)
2757      AS=B1*YB
2758
2759      C
2760      C DETERMINE CONTROL PARAMETERS IN COUNTER DO LOOP
2761      C
2762      GO TO (10,11,12,13),TRFACE
2763      10  TL=IFIX((NMAX-1.)/4.+0.51)
2764      GO TO 14
2765      12  TL=IFIX((2.*NX-1.)/4.+0.51)
2766      GO TO 14
2767      13  TL=IFIX((NMAX-4.)/4.+0.51)
2768
2769      C
2770      C DETERMINE STRAINS AT THE VARIOUS BARS AND EVALUATE THEM
2771      C WHETHER THEY YIELD OR NOT
2772      C
2773      14  DO 15 NI=1,TL
2774      ECS(NI)=((YB-0.5*CY+Y(NI))-ECU)/YB
2775      IF(ABS(ECS(NI)).GT.ESMAX) GO TO 50

```

```

2761      GO TO 81
2762      ECS(NI)=(ESMAX+ECS(NI))/ABS(ECS(NI))
2763      ETS(NI)=((Y(NI)+0.5*CY-YB)*ESMAX)/(D-YB)
2764      IF(ABS(ETS(NI)).GT.ESMAX) GO TO 82
2765      GO TO 15
2766      ETS(NI)=(ESMAX+ETS(NI))/ABS(ETS(NI))
2767      15 CONTINUE
2768      C
2769      C CHECK WHETHER SECTION HAVING EXTREME COMPRESSION BARS AND EXTREME
2770      C TENSION BARS AND CENTER BARS OR NOT HAVING THEM
2771      C
2772      GO TO (16,11,17,18),TRFACE
2773      16 NO=NMIX/2
2774      NOO=NO/2
2775      C
2776      C FOR BARS IN LATERAL FACES, CONSIDERING NO EXTREME BARS IN
2777      C COMPRESSION AND IN TENSION
2778      C
2779      ECSR=0.
2780      ETSL=0.
2781      GO TO 19
2782      17 NO=NX
2783      NOO=NO/2
2784      GO TO 20
2785      18 NO=NMIX/2
2786      NOO=NO/2
2787      IF(NO/2.-NOO.LE.0.0001) GO TO 20
2788      GO TO 21
2789      20 ECSR=((YB-DP)*ECU)/YB
2790      IF(ABS(ECSR).GT.ESMAX) GO TO 83
2791      GO TO 84
2792      83 ECSR=(ESMAX+ECSR)/ABS(ECSR)
2793      84 ETSL=ESMAX
2794      GO TO 19
2795      21 ECSR=0.
2796      ETSL=0.
2797      19 IF(TRFACE.EQ.4) GO TO 22
2798      IF(NO/2.-NOO.LE.0.0001) GO TO 23
2799      22 ESC=((YB-0.5*CY)*ECU)/YB
2800      IF(ABS(ESC).GT.ESMAX) GO TO 85
2801      GO TO 24
2802      85 ESC=(ESMAX+ESC)/ABS(ESC)
2803      GO TO 24
2804      23 ESC=0.
2805      C
2806      C CALCULATE FORCES IN SECTION AT BALANCED CONDITION
2807      C
2808      24 GO TO (25,11,26,27),TRFACE
2809      C
2810      C BASED ON ECSR=0 AND ETSL=0
2811      C
2812      25 CSR=0.
2813      FTL=0.
2814      GO TO 28
2815      C
2816      C COMPUTE FORCES IN EXTREME COMPRESSION AND IN EXTREME TENSION
2817      C
2818      26 CSR=(NY*PRIME*DB(J)**2*(ES+ECSR-0.85*FPC))/4.
2819      FTL=(NY*PRIME*DB(J)**2*(ES+ETSL)/4.
2820      C
2821      C COMPUTE FORCES IN CONCRETE AND IN CENTER BARS
2822      C
2823      28 CC=0.85*FPC*AB*CX
2824      IF(ESC.LE.0.) GO TO 29
2825      CSC=(2.*PRIME*DB(J)**2*(ES+ESC-0.85*FPC))/4.
2826      GO TO 30
2827      29 CSC=0.
2828      GO TO 30
2829      C
2830      C DETERMINE AREA OF SEGMENT FOR A CIRCULAR REINFORCEMENT PATTERN
2831      C
2832      27 RR=0.5*CY
2833      IF(AB.LE.RR) GO TO 31
2834      PP=AB-RR
2835      GO TO 32
2836      31 PP=RR-AB
2837      32 Q=RR**2-PP**2
2838      QO=SQRT(Q)
2839      PR=PP/RR
2840      RFAD=ARCOS(PR)
2841      IF(AB.GT.RR) GO TO 86
2842      GO TO 87
2843      86 RFAD=PRIME-RFAD
2844      87 AREA=(CY**2*(RFAD-SIN(RFAD)*COS(RFAD)))/4.
2845      CC=0.85*FPC*AREA
2846      IF(ECSR.LE.0.) GO TO 33
2847      CSR=(PRIME*DB(J)**2*(ES+ECSR-0.85*FPC))/4.
2848      FTL=(PRIME*DB(J)**2*(ES+ETSL)/4.
2849      GO TO 34
2850      33 CSR=0.
2851      FTL=0.
2852      34 CSC=(2.*PRIME*DB(J)**2*(ES+ESC-0.85*FPC))/4.
2853      C
2854      C COMPUTE FORCES WITH BARS IN VARIOUS COMPRESSION AND TENSION SIDE
2855      C
2856      30 CSSUM=0.
2857      FTSUM=0.
2858      DO 35 NI=1,TL
2859      CS(NI)=(2.*PRIME*DB(J)**2*(ES+ECS(NI)-0.85*FPC))/4.
2860      CSSUM=CSSUM+CS(NI)
2861      FT(NI)=(2.*PRIME*DB(J)**2*(ES+ETS(NI))/4.
2862      FTSUM=FTSUM+FT(NI)
2863      35 CONTINUE
2864      C
2865      C CHECK TO SEE ALL FT(NI) AND FTL ARE ACTUALLY A TENSION. THAT IS
2866      C WHETHER EFFECT OF THE DISPLACED CONCRETE FTP IS INCLUDED OR NOT
2867      C
2868      NII=1
2869      36 FTP=((TL-NII+1.)**2.*PRIME*DB(J)**2*0.85*FPC)/4.
2870      IF(NII.GT.TL) GO TO 37
2871      IF(FT(NII).LT.0.) GO TO 37
2872      NII=NII+1
2873      GO TO 36
2874      37 IF(FTL.GE.0.) GO TO 38
2875      IF(TRFACE.EQ.4) GO TO 39
2876      FTL=FTL+(NY*PRIME*DB(J)**2*0.85*FPC)/4.
2877      GO TO 38
2878      38 FTL=FTL+(PRIME*DB(J)**2*0.85*FPC)/4.
2879      38 PBX=PHIV*(CC+CSR+CSC+CSSUM-FTL-FTSUM-FTP)
2880      C

```

```

2881 C COMPUTE DESIGN STRENGTH AT COMPRESSION CONTROL REGION.
2882 C APPLICATION MORE OR LESS IS SAME AS ABOVE.
2883 C USING HALPING INTERVAL TECHNIQUE TO FIND LOCATION OF NEUTRAL
2884 C AXIS
2885 C
2886 VL=YB
2887 YU=CY/81
2888 VC=(VL+YU)/2.
2889 VY=YU-VL
2890 IF(ICODE.EQ.1) DIFFER=5
2891 IF(ICODE.EQ.2) DIFFER=0.2
2892 IF(ABS(VY).LT.DIFFER) GO TO 41
2893 AC=B1*YC
2894 C
2895 C COMPUTE STRAINS IN THE STEEL
2896 C
2897 DO 42 NI=1,TL
2898 ECS(NI)=((VC-0.5*CY+V(NI))*ECU)/YC
2899 IF(ABS(ECS(NI)).GT.ESMAX) GO TO 88
2900 GO TO 89
2901 88 ECS(NI)=(ESMAX+ECS(NI))/ABS(ECS(NI))
2902 89 ETS(NI)=((V(NI)+0.5*CY-VC)*ECU)/YC
2903 IF(ABS(ETS(NI)).GT.ESMAX) GO TO 90
2904 GO TO 42
2905 90 ETS(NI)=(ESMAX+ETS(NI))/ABS(ETS(NI))
2906 42 CONTINUE
2907 C
2908 C CHECK ECSR AND ETSL AND ESC EQUAL TO ZERO OR NOT
2909 C
2910 GO TO (43,11,44,45),TRFACE
2911 45 IF(INO/2.-NOO.LE.0.0001) GO TO 44
2912 GO TO 46
2913 44 ECSR=((VC-DP)*ECU)/YC
2914 IF(ABS(ECSR).GT.ESMAX) GO TO 91
2915 GO TO 92
2916 91 ECSR=(ESMAX+ECSR)/ABS(ECSR)
2917 92 ETSL=((D-YC)*ECU)/YC
2918 IF(ABS(ETSL).GT.ESMAX) GO TO 93
2919 GO TO 43
2920 93 ETSL=(ESMAX+ETSL)/ABS(ETSL)
2921 GO TO 43
2922 46 ECSR=0.
2923 ETSL=0.
2924 43 IF(TRFACE.EQ.4) GO TO 47
2925 IF(INO/2.-NOO.LE.0.0001) GO TO 46
2926 47 ESC=((VC-0.5*CY)*ECU)/YC
2927 IF(ABS(ESC).GT.ESMAX) GO TO 94
2928 GO TO 49
2929 94 ESC=(ESMAX+ESC)/ABS(ESC)
2930 GO TO 48
2931 48 ESC=0.
2932 C
2933 C CALCULATE FORCES IN SECTION
2934 C
2935 49 GO TO (50,11,51,52),TRFACE
2936 50 CSR=0.
2937 FTL=0.
2938 GO TO 53
2939 51 CSR=(NY*PRIME*DB(J)**2*(ES+ECSR-0.85*FPC))/4.
2940 FTL=(NY*PRIME*DB(J)**2*(ES+ETSL)/4.
2941 52 CC=0.85*FPC*AC*CX
2942 C
2943 C COMPUTE POINT OF APPLICATION FOR CC
2944 C
2945 VCC=CY/2.-0.5*AC
2946 IF(ESC.LE.0.) GO TO 54
2947 CSC=(2.*PRIME*DB(J)**2*(ES+ESC-0.85*FPC))/4.
2948 GO TO 55
2949 54 CSC=0.
2950 GO TO 56
2951 52 IF(AC.LE.RR) GO TO 56
2952 PP=AC-RR
2953 GO TO 57
2954 56 PP=RR-AC
2955 57 Q=RR**2-PP**2
2956 Q=SQRT(Q)
2957 PR=PP/RR
2958 RFAD=ARCCOS(PR)
2959 IF(AC.GT.RR) GO TO 55
2960 GO TO 56
2961 95 RFAD=PRIME-RFAD
2962 96 AREA=(CY**2*(RFAD-SIN(RFAD)*COS(RFAD)))/4.
2963 SEG=(CY**3*(SIN(RFAD))**3)/12.
2964 C
2965 C COMPUTE POINT OF APPLICATION FOR CC IN CIRCULAR BARS PATTERN
2966 C
2967 VCC=SEG/AREA
2968 CC=0.85*FPC*AREA
2969 IF(ECSR.LE.0.) GO TO 58
2970 CSR=(PRIME*DB(J)**2*(ES+ECSR-0.85*FPC))/4.
2971 FTL=(PRIME*DB(J)**2*(ES+ETSL)/4.
2972 GO TO 59
2973 58 CSR=0.
2974 FTL=0.
2975 59 CSC=(2.*PRIME*DB(J)**2*(ES+ESC-0.85*FPC))/4.
2976 60 DO 60 NI=1,TL
2977 CS(NI)=(2.*PRIME*DB(J)**2*(ES+ECS(NI)-0.85*FPC))/4.
2978 FT(NI)=(2.*PRIME*DB(J)**2*(ES+ETS(NI))/4.
2979 60 CONTINUE
2980 C
2981 C CHECK TO SEE ALL FT(NI) AND FTL ARE ACTUALLY A TENSION
2982 C
2983 NI=1
2984 61 IF(FT(NI).GT.0.) GO TO 62
2985 FT(NI)=FT(NI)+(2.*PRIME*DB(J)**2*0.85*FPC)/4.
2986 NI=NI+1
2987 IF(NI.LE.TL) GO TO 61
2988 62 IF(FTL.GE.0.) GO TO 63
2989 IF(TRFACE.EQ.4) GO TO 64
2990 FTL=FTL+(NY*PRIME*DB(J)**2*0.85*FPC)/4.
2991 GO TO 63
2992 64 FTL=FTL+(PRIME*DB(J)**2*0.85*FPC)/4.
2993 FUN=F12(FT,CS,V,FTL,CY,DP,EY,TL,CSR,CSC,CC,VCC)
2994 IF(FUN) 65,41,66
2995 YU=YC
2996 GO TO 40
2997 VL=YC
2998 GO TO 40
2999 41 PTSUM=0.
3000 CSSUM=0.

```

```

3001      DO 87 NI=1,TL
3002      FTSUM=FTSUM+FT(NI)
3003      CSSUM=CSSUM+CS(NI)
3004      87 CONTINUE
3005      PNK=CSR+CSSUM+CSC+CC-FTL-FTSUM
3006      FTSUM1=0.
3007      CSUM1=0.
3008      DO 100 NI=1,TL
3009      FTSUM1=FTSUM1+FT(NI)*Y(NI)
3010      CSUM1=CSUM1+CS(NI)*Y(NI)
3011      100 CONTINUE
3012      MNK=FTL*(0.5*CY-DP)+CSR*(0.5*CY-DP)+CC+YCC+FTSUM1+CSUM1
3013      PUDEX=PHIV*PNK
3014      MUDEX=PHIV*MNK
3015      11 RETURN
3016      END
3017      C
3018      C
3019      FUNCTION F12(FT,CS,Y,FTL,CY,DP,EY,TL,CSR,CSC,CC,YCC)
3020      C
3021      C TAKING MOMENTS OF THE INTERNAL FORCES ABOUT APPLIED AXIAL
3022      C LOAD AND WRITING AN EQUATION IN TERMS OF FUNCTION OF UNKNOWN
3023      C NEUTRAL AXIS
3024      C
3025      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
3026      DIMENSION FT(12),CS(12),Y(12)
3027      FTSUM=0.
3028      CSUM=0.
3029      IF(TL.LE.0) GO TO 100
3030      DO 10 NI=1,TL
3031      FTSUM=FTSUM+FT(NI)*(Y(NI)+EY)
3032      CSUM=CSUM+CS(NI)*(Y(NI)-EY)
3033      10 CONTINUE
3034      100 F12=FTSUM+CSUM+FTL*(0.5*CY-DP+EY)+CSR*(0.5*CY-DP-EY)-CSC*EY
3035      * -CC*(EY-YCC)
3036      RETURN
3037      END
3038      C
3039      C *****
3040      C *
3041      C SUBROUTINE ADEQU1
3042      C *
3043      C *THIS SUBROUTINE ADEQUACY CHECKS DESIGN STRENGTH CAPACITY
3044      C *WITH REQUIRED STRENGTH
3045      C *
3046      C *****
3047      C
3048      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
3049      COMMON PG,ESMAX,TYPE,TREINF,TLEN1,TBEND,TLEN2,TADEO,TPROB
3050      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9),
3051      * PD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
3052      * TBMAX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAXX
3053      COMMON PPC,FY,FSY,EC,B1,TBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
3054      * SPC,PD,PL,PU,MX1,MX2,BOX,MV1,MV2,BDY,TOUT,TMODE
3055      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
3056      COMMON TDIMFG,AG,AS,CG,SB,DP,AST,NK,NV,SBX,SBY,DS,
3057      * Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MCX,
3058      * MAPPX,PUDEX,MUDEX,PBX
3059      COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PBY,MDYREQ,MOXREQ,
3060      * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
3061      * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
3062      COMMON TESTMO
3063      COMMON NLC,JPOB,TREVIS,TOSSE,SPACE,PITCH,NSPACR
3064      DIMENSION A(9)
3065      C
3066      C DETERMINE WHETHER CAPACITY REDUCTION FACTOR NEEDS TO BE
3067      C MODIFIED
3068      C
3069      PDA=0.1*PPC=AG
3070      IF(PU.GE.PDA) GO TO 50
3071      PBXX=PBX/PHIV
3072      PDASH=AMIN1(PDA,PBXX)
3073      C
3074      C NEW PHIV VALUE
3075      PHIVN=PHIV+((0.9-PHIV)*(PDASH-PU))/PDASH
3076      IF(PHIVN.LT.PHIV) PHIVN=PHIV
3077      PBX=PBXX*PHIVN
3078      PUDEX=PUDEX*PHIVN/PHIV
3079      MUDEX=MUDEX*PHIVN/PHIV
3080      50 IF(TOUT.NE.1) GO TO 10
3081      IF(TMODE.EQ.1) WRITE(7,400)
3082      WRITE(8,400)
3083      400 FORMAT(' ','ADEQUACY CHECK FOR UNI-AXIAL BENDING : '/')
3084      C
3085      C ADEQUACY CHECK DESIGN AXIAL LOAD
3086      C
3087      10 RATIOF=PUDEX/PU
3088      MOREA=RATIOF-1.0
3089      IF(PUDEX.GE.PU) GO TO 20
3090      C
3091      C SET TOLERANCE LIMIT OF 0.03 FOR COMPARING PROVIDED AXIAL
3092      C LOAD AND MOMENT WITH APPLIED AXIAL LOAD AND MOMENT
3093      C
3094      IF(MOREA+0.03.GE.0.) GO TO 20
3095      IF(TDESIG.EQ.1) GO TO 19
3096      133 IF(TMODE.EQ.1) GO TO 2001
3097      GO TO 131
3098      2001 WRITE(7,208)RATIOF
3099      131 WRITE(8,208)RATIOF
3100      IF(MOREA+0.03.GE.0.) GO TO 2880
3101      IF(TMODE.EQ.1) WRITE(7,2881)
3102      WRITE(8,2881)
3103      2881 FORMAT(' ','BX','' SECTION IS NOT ADEQUATE FOR THIS',
3104      * ' ' LOADING '' '/')
3105      C
3106      C PROCEED TO NEXT PROBLEM RESULTING FROM UNSATISFACTORY SECTION
3107      C
3108      2880 TADEO=1
3109      RETURN
3110      19 IF(TMODE.EQ.1) GO TO 2002
3111      GO TO 132
3112      2002 IF(TOUT.NE.1) GO TO 132
3113      WRITE(7,207) RATIOF
3114      132 WRITE(8,207) RATIOF
3115      C
3116      C GO TO SUBROUTINE MODIF1 TO INCREASE REINFORCEMENT
3117      C
3118      TADEO=2
3119      RETURN
3120      20 IF(TDESIG.NE.1) GO TO 133

```

```

3121      IF(TMODE.EQ.1) GO TO 2008
3122      GO TO 138
3123      2008 WRITE(7,205) RATIOF
3124      138 WRITE(8,206) RATIOF
3125
3126      C
3127      C RETURN TO MAIN ROUTINE AND CONTINUE EXECUTION
3128      C
3129      TADEQ=3
3130      RETURN
3131
3132      C
3133      C ***FORMAT STATEMENTS***
3134
3135      205 FORMAT(' ',5X,'FOR SECTION SELECTED THE AXIAL LOAD CAPACITY',
3136      * /6X,'UNDER APPLIED MOMENT IS',
3137      * F7.3,' TIMES'/6X,'THE APPLIED AXIAL LOAD'/)
3138
3139      206 FORMAT(' ',5X,'FOR SECTION ENTERED THE AXIAL LOAD CAPACITY',
3140      * /6X,'UNDER APPLIED MOMENT IS',
3141      * F7.3,' TIMES'/6X,'THE APPLIED AXIAL LOAD'/)
3142
3143      207 FORMAT(' ',5X,'FOR SECTION SELECTED THE AXIAL LOAD CAPACITY',
3144      * /6X,'UNDER APPLIED MOMENT IS',
3145      * F7.3,' TIMES'/6X,'THE APPLIED AXIAL LOAD, AND IS NOT',
3146      * ' ACCEPTABLE, '/6X,'INCREASING REINFORCEMENT'/)
3147      END
3148
3149      C
3150      C *****
3151      C *
3152      C SUBROUTINE SLEND2
3153      C *
3154      C * THIS SUBROUTINE CHECKS SLENDERNESS EFFECTS ABOUT Y-AXIS.
3155      C * APPLICATION MORE OR LESS IS SIMILAR TO SLEND1
3156      C * *****
3157      C
3158      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y), INTEGER(T,J,N,S)
3159      COMMON PC,ESMAX,TYPE,TREINF,TSLEN1,TSLEN2,TADEQ,TPROB,
3160      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(8),DB(8),
3161      * FD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
3162      * TBMX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAXX,
3163      COMMON FPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
3164      * SPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BDY,TOUT,TMODE
3165      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
3166      COMMON TDIMFG,AC,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
3167      * Y(12),ISEX,X(12),ISEY,TMODAS,PDES,EY,MCX,
3168      * MAPPX,PUDEXX,MUDEXX,PBX
3169      COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PSY,MOYREQ,MOXREQ,
3170      * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
3171      * XC,XB,ECSY(12),ETSY(12),CSV(12),PTY(12)
3172      COMMON TESTMD
3173      COMMON NLC,JPOS,TREVIS,TBSS,SPACE,PITCH,NSPACR
3174      DIMENSION A(8)
3175
3176      C
3177      IF(TOUT.NE.1) GO TO 3000
3178      IF(TMODE.EQ.1) WRITE(7,300)
3179      WRITE(8,300)
3180
3181      300 FORMAT(' ',2X,'CHECK SLENDERNESS EFFECTS BENDING',
3182      * ' ABOUT Y-AXIS'/1X,70(' ')/)
3183
3184      C
3185      C COMPUTE MOMENT OF INERTIA OF CROSS-SECTION AND RADIUS OF
3186      C GYRATION AND SLENDERNESS EFFECTS ABOUT Y-AXIS
3187      C
3188      3000 IGY=CG*AG*CX**2
3189      RYT=IGY/AG
3190      RV=SQRT(RYT)
3191      SY=KY=LUY/RV
3192      IF(TYBRAC.EQ.1) GO TO 10
3193      IF(SY.LE.22.) GO TO 11
3194      IF(SPU.LE.0.1.AND.SPC.LE.0.1) GO TO 12
3195      CMY=1.0
3196
3197      C
3198      C COMPUTE MAGNIFICATION FACTOR DELTAY
3199      C
3200      DELTAY=CMY/(1.-SPU/(PHIV*SPC))
3201      IF(DELTAY.LE.1.0) DELTAY=1.0
3202      GO TO 13
3203
3204      11 DELTAY=1.0
3205      GO TO 50
3206
3207      C
3208      C PROCEED TO NEXT PROBLEM RESULTING FROM INCOMPLETE INPUT PARAMETERS
3209      C
3210      12 TSLEN2=2
3211      IF(TMODE.EQ.1) GO TO 2001
3212      GO TO 131
3213
3214      2001 WRITE(7,201)
3215      131 WRITE(8,201)
3216      RETURN
3217
3218      10 IF(ABS(MV2).LE.0.) GO TO 510
3219      MV12=ABS(MV1/MV2)
3220      GO TO 511
3221
3222      510 MV12=1.0
3223      511 IF(SY.LT.34.-12.*MV12) GO TO 14
3224      IF(TOUT.NE.1) GO TO 3001
3225      IF(TMODE.EQ.1) GO TO 2002
3226      GO TO 132
3227
3228      2002 WRITE(7,202)
3229      132 WRITE(8,202)
3230
3231      C
3232      C COMPUTE EI VALUE AND CRITICAL LOAD
3233      C
3234      3001 EIV=(0.2*EC*IGY+ES*ISEY)/(1.+BDY)
3235      PCY=(PRIME**2*EIV)/(KY=LUY)**2
3236
3237      C
3238      C DETERMINE GOVERNING ECCENTRICITY AND REQUIRED END MOMENT MU
3239      C FOR LONG COLUMNS
3240      C
3241      IF(ABS(MV2).LE.0.1) GO TO 15
3242      B=MV1/MV2
3243      GO TO 20
3244
3245      15 B=1.0
3246      20 IF(TCODE.EQ.1) GO TO 16
3247      EMINX=0.6+0.03*CX
3248      GO TO 17
3249
3250      16 IF(TYPE.EQ.1) GO TO 18
3251      F=0.05*CX
3252      GO TO 19
3253
3254      18 F=0.1*CX
3255      19 EMINX=AMAX1(F,EMIN)
3256      17 EX=ABS(MV2)/PU
3257      IF(EX.LE.EMINX) GO TO 500
3258      GO TO 501
3259
3260      500 EX=EMINX

```

```

3241      IF(TCODE.EQ.1) MUYY=(PU*EX)/1000000.
3242      IF(TCODE.EQ.2) MUYY=(PU*EX)/12000.
3243      IF(TMODE.EQ.1) GO TO 550
3244      GO TO 551
3245      550 WRITE(7,502) MUYY
3246      551 WRITE(8,502) MUYY
3247      MUYP=PU*EX
3248      CMY=0.8+0.4*B
3249      IF(CMY.LT.0.4) CMY=0.4
3250      DELTAY=CMY/(1.-PU/(PHIV*PCY))
3251      IF(DELTAY.LE.1.0) DELTAY=1.0
3252      GO TO 504
3253      14 IF(TOUT.NE.1) GO TO 3007
3254      IF(TMODE.EQ.1) GO TO 2003
3255      GO TO 133
3256      2003 WRITE(7,203)
3257      133 WRITE(8,203)
3258      3007 DELTAY=1.0
3259      GO TO 50
3260      C DETERMINE GOVERNING ECCENTRICITY AND REQUIRED END MOMENT
3261      C FOR SHORT COLUMNS
3262      C
3263      13 IF(TCODE.EQ.1) GO TO 50
3264      EMINX=0.6+0.03*CX
3265      GO TO 51
3266      50 IF(TYPE.EQ.1) GO TO 52
3267      F=0.06*CX
3268      GO TO 53
3269      52 F=0.1*CX
3270      53 EMINX=AMAX1(F,EMIN)
3271      51 EX=ABS(MY2)/PU
3272      IF(EX.LE.EMINX) GO TO 505
3273      GO TO 505
3274      505 EX=EMINX
3275      MUYP=PU*EX
3276      IF(TCODE.EQ.1) MUYY=MUY/1000000.
3277      IF(TCODE.EQ.2) MUYY=MUY/12000.
3278      IF(TMODE.EQ.1) GO TO 552
3279      GO TO 553
3280      552 WRITE(7,502) MUYY
3281      553 WRITE(8,502) MUYY
3282      505 MUYP=PU*EX
3283      504 IF(DELTAY.LE.1.001) GO TO 520
3284      WRITE(8,204)DELTAY
3285      IF(TMODE.EQ.1) WRITE(7,204)DELTAY
3286      C COMPUTE MAGNIFIED MOMENT THAT IS APPLIED MOMENT
3287      520 MCY=DELTAY*MUY
3288      EX=MCY/PU
3289      MMINY=PU*EMINX
3290      MAPPY=MCY
3291      C
3292      C RETURN TO MAIN ROUTINE AND CONTINUE EXECUTON
3293      C
3294      TSLEN2=1
3295      RETURN
3296      C
3297      C *****FORMAT STATEMENTS*****
3298      C
3299      201 FORMAT(' ',EX,'THIS IS A LONG UNBRACED FRAME. MORE INPUT',
3300      *'/EX, 'PARAMETERS WILL BE REQUIRED TO COMPUTE'/EX,
3301      * 'MOMENT MAGNIFIER ABOUT Y-AXIS ONLY'/)
3302      202 FORMAT(' ',EX,'SLENDERNESS EFFECT MUST BE CONSIDERED'/EX,
3303      * 'BY MAGNIFYING MOMENT ABOUT Y-AXIS'/)
3304      203 FORMAT(' ',EX,'SLENDERNESS EFFECT MAY BE NEGLECTED'/)
3305      502 FORMAT(' ',EX,'MINIMUM ECCENTRICITY ABOUT Y-AXIS GOVERNS',
3306      * '/EX, 'DESIGN MOMENT ABOUT Y-AXIS = ',F8.0/EX, 'IN PROJECT',
3307      * ' UNIT'/)
3308      204 FORMAT(' ',EX,'MAGNIFICATION FACTOR FOR MOMENT ABOUT',
3309      * ' Y-AXIS = ',F5.2/)
3310      C
3311      C END
3312      C
3313      C *****
3314      C
3315      C SUBROUTINE CAPAC2
3316      C
3317      C * THIS SUBROUTINE INVESTIGATES STRENGTH CAPACITY ABOUT Y-AXIS
3318      C * IN COMPRESSION CONTROLS REGIONS ( PHV > PHV ) AND AT BALANCED
3319      C * CONDITION WITH BARS DISTRIBUTED IN LATERAL FACES OR IN
3320      C * END FACES OR IN BOTH FACES OR IN CIRCULAR CORE
3321      C * APPLICATION MORE OR LESS IS SIMILAR TO CAPAC1
3322      C
3323      C *****
3324      C
3325      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
3326      COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEQ,TPROB
3327      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9),
3328      * PD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
3329      * TBMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAXX
3330      COMMON FPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
3331      * SPC,PD,PL,PU,MX1,MX2,BDX,MY1,MY2,BDY,TOUT,TMODE
3332      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
3333      COMMON TDIMFG,AG,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
3334      * Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MCX,
3335      * MAPPX,PUDEX,MUDESX,PSX
3336      COMMON EX,MCY,MAPPY,PUDESX,MUDESY,PSY,MOYREQ,MOXREQ,
3337      * VC,YB,ECS(12),ETS(12),CS(12),PT(12),
3338      * XC,XB,ECSY(12),ETSY(12),CSY(12),PTY(12)
3339      COMMON TESTMO
3340      COMMON NLC,JPOB,TREVIS,YDSS,SPACE,PITCH,NSPACR
3341      DIMENSION A(8)
3342      C
3343      IF(TOUT.NE.1) GO TO 4000
3344      IF(TMODE.EQ.1) WRITE(7,300)
3345      WRITE(8,300)
3346      300 FORMAT(' ',2X,'CAPACITY CHECK FOR BENDING ABOUT Y-AXIS'/1X,
3347      * ' 70(''/'')')
3348      4000 IF(TRFACE.EQ.1) GO TO 10
3349      C
3350      C COMPUTE DESIGN STRENGTH WITH BARS IN END FACES OR IN BOTH FACES
3351      C OR IN CIRCULAR CORE
3352      C
3353      CALL CAPA22
3354      GO TO 11
3355      C
3356      C COMPUTE DESIGN STRENGTH WITH BARS IN LATERAL FACES
3357      C
3358      10 CALL CAPA21
3359      11 IF(TOUT.NE.1) RETURN
3360      PUZY=PUDESX/1000.

```

```

3361      MPBY=PBV/1000.
3362      IF(TCODE.EQ.1) GO TO 1990
3363      MUDY=MUDSY/12000.
3364      GO TO 1991
3365      1990 MUDY=MUDSY/1000000.
3366      1991 WRITE(8,200) XC,XB,PUDY,MUDY,MPBY
3367      IF(TMODE.EQ.1) WRITE(7,200)XC,XB,PUDY,MUDY,MPBY
3368      RETURN
3369
3370 C
3371 200 FORMAT(' ',2X,'LOCATION OF NEUTRAL AXIS // Y-AXIS= ',
3372 * F8.3,1X,'IN/MM'/3X,'NEUTRAL AXIS AT BALANCED CONDITION= ',
3373 * F8.3,1X,'IN/MM'/3X,'DESIGN AXIAL ',
3374 * 'LOAD WITH ONLY Y-AXIS BENDING= ',F8.0,1X,'KIPS/KN'/3X,
3375 * 'DESIGN MOMENT ABOUT Y-AXIS ONLY= ',F8.0,1X,'KIP-FT/KN-M'/3X,
3376 * 'LOAD AT BALANCED STRAIN CONDITION= ',F8.0,1X,'KIPS/KN'/)
3377
3378 C
3379 C
3380 C
3381 C
3382 C
3383 C
3384 C
3385 C
3386 C
3387 C
3388 C
3389 C
3390 C
3391 C
3392 C
3393 C
3394 C
3395 C
3396 C
3397 C
3398 C
3399 C
3400 C
3401 C
3402 C
3403 C
3404 C
3405 C
3406 C
3407 C
3408 C
3409 C
3410 C
3411 C
3412 C
3413 C
3414 C
3415 C
3416 C
3417 C
3418 C
3419 C
3420 C
3421 C
3422 C
3423 C
3424 C
3425 C
3426 C
3427 C
3428 C
3429 C
3430 C
3431 C
3432 C
3433 C
3434 C
3435 C
3436 C
3437 C
3438 C
3439 C
3440 C
3441 C
3442 C
3443 C
3444 C
3445 C
3446 C
3447 C
3448 C
3449 C
3450 C
3451 C
3452 C
3453 C
3454 C
3455 C
3456 C
3457 C
3458 C
3459 C
3460 C
3461 C
3462 C
3463 C
3464 C
3465 C
3466 C
3467 C
3468 C
3469 C
3470 C
3471 C
3472 C
3473 C
3474 C
3475 C
3476 C
3477 C
3478 C
3479 C
3480 C

```

SUBROUTINE CAPA21
 *THIS SUBROUTINE DETERMINES DESIGN STRENGTH AT BALANCED
 *CONDITION AND IN COMPRESSION REGION WITH BARS IN
 *LATERAL FACES.
 *APPLICATION MORE OR LESS IS SIMILAR TO CAPA12
 *
 IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y), INTEGER(T,J,N,S)
 COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEQ,TPROB
 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DS(9),
 * FD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
 * TBMAX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAX
 COMMON FPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
 * SPC,PD,PL,PU,MK1,MK2,SDX,MV1,MV2,SDY,TOUT,TMODE
 COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,HSS,HMAX
 COMMON TDINFG,AG,AS,CG,SG,DP,AST,MK,MV,SDX,SEY,DS,
 * V(12),ISEX,K(12),ISEY,TMODAS,PDES,EV,MCK,
 * MAPPK,PUDSK,MUDESK,PRX
 COMMON EX,MCV,MAPPV,PUDSY,MUDSY,PBY,MOVREO,MOXREO,
 * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
 * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
 COMMON TESTNO
 COMMON NLC,JPOS,TREVIS,TOSSE,SPACE,PITCH,NSPACR
 DIMENSION A(9)
 D=CX-DP
 C COMPUTE COMPRESSION FORCE IN STEEL
 CSS=0.5*AST*(FY-0.55*FPC)
 C USING HALPING INTERVAL TECHNIQUE TO FIND LOCATION OF NEUTRAL AXIS
 XL=(ECU*D)/(ECU+ESMAX)
 XU=CX/B1
 10 XC=(XL+XU)/2.
 C COMPUTE FORCES IN TENSION REINFORCEMENT AND COMPRESSION CONCRETE
 FTT=0.5*AST*ECU+ES*((D-XC)/XC)
 CC=0.5*FPC*BI*XC*CY
 XU=XU-XL
 IF(TCODE.EQ.1) DIFFER=5
 IF(TCODE.EQ.2) DIFFER=0.2
 IF(ABS(XC).LT.DIFFER) GO TO 11
 FUN=F21(XC,FTT,CSS,CC,D,CX,EX,DP,B1)
 IF(FUN) 12,11,14
 12 XU=XC
 GO TO 10
 14 XL=XC
 GO TO 10
 11 IF(B1*XC-CT.D) GO TO 20
 GO TO 21
 C EFFECT OF DISPLACED CONCRETE IS INCLUDED
 20 FTT=FTT+0.5*AST*0.55*FPC
 21 PHV=CC+CSS-FTT
 MHV=FTT*(D-0.5*CX)+CSS*(0.5*CX-DP)+CC*(0.5*CX-0.5*BI*XC)
 PUDSY=PHV/PHV
 MUDESY=PHIV*MHV
 C COMPUTE BALANCED AXIAL LOAD
 XB=(ECU*D)/(FY/ES+ECU)
 PBV=PHIV*0.55*FPC*(BI*XB-CY-0.5*AST)
 RETURN
 END
 REAL FUNCTION F21(XC,FTT,CSS,CC,D,CX,EX,DP,B1)
 C TAKING MOMENT OF INTERNAL FORCES ABOUT APPLIED AXIAL LOAD
 C AND WRITING AN EQUATION IN TERM OF FUNCTION OF UNKNOWN
 C NEUTRAL AXIS
 F21=FTT*(D-(0.5*CX-EX))+CSS*(0.5*CX-EX-DP)+CC*(0.5*CX-EX
 * -0.5*BI*XC)
 RETURN
 END

SUBROUTINE CAPA22
 *THIS SUBROUTINE DETERMINES DESIGN STRENGTH AT BALANCED
 *CONDITION AND IN COMPRESSION CONTROL REGION WITH BARS
 *DISTRIBUTED IN END FACES OR IN BOTH FACES OR IN CIRCULAR CORE.
 *APPLICATION MORE OR LESS IS SIMILAR TO CAPA12
 *
 IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y), INTEGER(T,J,N,S)
 COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEQ,TPROB
 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DS(9),
 * FD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
 * TBMAX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAX
 COMMON FPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,

```

3481      * SPC,PD,PL,PU,MX1,MX2,BDX,MY1,MY2,BDY,TOUT,TMODE
3482      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
3483      COMMON TDIMFG,AG,AS,CG,SG,DP,AST,NX,NY,SBX,SBY,DS,
3484      * Y(12),ISEX,X(12),ISEY,TMODAS,PQDES,EY,MCK,
3485      * MAPPK,PUDESX,MUDESX,PSX
3486      COMMON EX,MCV,MAPPV,PUDESX,MUDESX,PSY,MOYREO,MOXREO,
3487      * YC,YB,ECS(12),ETX(12),CS(12),FT(12),
3488      * XC,YB,ECSV(12),ETSV(12),CSV(12),FTV(12)
3489      COMMON TESTMO
3490      COMMON NLC,JPOB,TREVIS,TDSS,SPACE,PITCH,NSPACR
3491      DIMENSION A(9)
3492
3493      C
3494      D=CX-DP
3495
3496      C
3497      C COMPUTE BALANCED AXIAL LOAD BY USING DIRECT APPLICATION OF STATICS
3498      C
3499      XB=(ECU-D)/(ECU+ESMAX)
3500      AB=B1-XB
3501
3502      C
3503      C COMPUTE CONTROL PARAMETERS IN COUNTER DO LOOP
3504      C
3505      GO TO (10,11,12,13),TRFACE
3506      11 TL=IFIX((NMAX-1.)/4.+0.51)
3507      GO TO 14
3508      12 TL=IFIX((2.-NY-1.)/4.+0.51)
3509      IF(TL.LE.0) GO TO 1
3510      GO TO 14
3511      13 TL=IFIX((NMAX-3.)/4.+0.51)
3512
3513      C
3514      C COMPUTE STRAINS AT VARIOUS BARS AND CHECK WHETHER THEY YIELD
3515      C OR NOT
3516      C
3517      14 DO 15 NI=1,TL
3518      ECSV(NI)=((XB-0.5+CX+X(NI))*ECU)/XB
3519      IF(ABS(ECSV(NI)).GT.ESMAX) GO TO 80
3520      GO TO 81
3521      80 ECSV(NI)=(ESMAX+ECSV(NI))/ABS(ECSV(NI))
3522      81 ETSV(NI)=((X(NI)+0.5+CX-XB)*ESMAX)/(D-XB)
3523      IF(ABS(ETSV(NI)).GT.ESMAX) GO TO 82
3524      GO TO 15
3525      82 ETSV(NI)=(ESMAX+ETSV(NI))/ABS(ETSV(NI))
3526      15 CONTINUE
3527
3528      C
3529      C CHECK WHETHER HAVING EXTREME COMPRESSION BARS AND EXTREME TENSION
3530      C BARS AND CENTER BARS OR NOT HAVING THEM
3531      C
3532      1 GO TO (10,16,17,18),TRFACE
3533      16 N0=NMAX/2
3534      N00=N0/2
3535
3536      C
3537      C CONSIDERING NO EXTREME COMPRESSION AND TENSION BARS IN END FACES
3538      C
3539      ECSR=0.
3540      ETSL=0.
3541      GO TO 19
3542      17 N0=NY
3543      N00=N0/2
3544      GO TO 20
3545      18 N0=NMAX/2+1
3546      N00=N0/2
3547      20 ECSR=((XB-DP)*ECU)/XB
3548      IF(ABS(ECSR).GT.ESMAX) GO TO 83
3549      GO TO 84
3550      83 ECSR=(ESMAX+ECSR)/ABS(ECSR)
3551      84 ETSL=ESMAX
3552      IF(N0.EQ.0) GO TO 23
3553      IF(N0/2.-N00.LE.0.0001) GO TO 23
3554      ESC=((XB-0.5+CX)*ECU)/XB
3555      IF(ABS(ESC).GT.ESMAX) GO TO 85
3556      GO TO 24
3557      85 ESC=(ESMAX+ESC)/ABS(ESC)
3558      GO TO 24
3559      23 ESC=0.
3560
3561      C
3562      C CALCULATE FORCES IN SECTION
3563      C
3564      24 GO TO (10,25,26,27),TRFACE
3565
3566      C
3567      C BASED ON ECSR=0 AND ETSL=0
3568      C
3569      25 CSR=0.
3570      FTL=0.
3571      CC=0.85*FPC*AB*CY
3572      GO TO 30
3573      26 CSR=(NX*PRIME*OB(J)**2+(ES+ECSR-0.85*FPC))/4.
3574      FTL=(NX*PRIME*OB(J)**2+ES*ETSL)/4.
3575      CC=0.85*FPC*AB*CY
3576      IF(TL.LE.0) GO TO 28
3577      GO TO 30
3578      28 CSSUM=0.
3579      FTSUM=0.
3580      GO TO 2
3581
3582      C
3583      C DETERMINE AREA OF SEGMENT FOR A CIRCULAR REINFORCEMENT PATTERN
3584      C
3585      27 RR=0.5*CX
3586      IF(AB.LE.RR) GO TO 31
3587      PP=AB-RR
3588      GO TO 32
3589      31 PP=RR-AB
3590      Q=RR**2-PP**2
3591      QQ=SQRT(Q)
3592      PR=PP/RR
3593      RFAD=ARCOS(PR)
3594      IF(AB.GT.RR) GO TO 86
3595      GO TO 87
3596      86 RFAD=PRIME-RFAD
3597      87 AREA=(CX**2*(RFAD-SIN(RFAD)*COS(RFAD)))/4.
3598      CC=0.85*FPC*AREA
3599      CSR=(PRIME*OB(J)**2+(ES+ECSR-0.85*FPC))/4.
3600      FTL=(PRIME*OB(J)**2+ES*ETSL)/4.
3601
3602      C
3603      C COMPUTE FORCES IN VARIOUS BARS
3604      C
3605      30 CSSUM=0.
3606      FTSUM=0.
3607      DO 35 NI=1,TL
3608      CSV(NI)=(2.-PRIME*OB(J)**2+(ES+ECSV(NI)-0.85*FPC))/4.
3609      CSSUM=CSSUM+CSV(NI)
3610      FTV(NI)=(2.-PRIME*OB(J)**2+ES*ETSV(NI))/4.

```

```

3601      FTSUM=FTSUM+FTY(N1)
3602      35 CONTINUE
3603      2 IF(ESC.LE.O.) GO TO 20
3604      CSC=(2.*PRIME-DB(J))*2*(ES-ESC-O.85*FPC)/4.
3605      GO TO 21
3606      20 CSC=O.
3607      C
3608      C CHECK WHETHER ALL FORCES IN TENSION REINFORCEMENT ARE ACTUALLY
3609      C A TENSION OR NOT, THAT IS THE EFFECT OF THE DISPLACED CONCRETE
3610      C FTP IS INCLUDED OR NOT
3611      C
3612      21 IF(TL.LE.O) GO TO 4
3613      N1=N1+1
3614      FTP=((TL-N1+1)*2.*PRIME-DB(J))*2*O.85*FPC/4.
3615      IF(N1.GT.TL) GO TO 37
3616      IF(FTY(N1).LT.O.) GO TO 37
3617      N1=N1+1
3618      GO TO 36
3619      4 FTP=O.
3620      37 IF(FTL.GE.O.) GO TO 38
3621      IF(TRFACE.EQ.4) GO TO 39
3622      FTL=FTL*(NX*PRIME-DB(J))*2*O.85*FPC/4.
3623      GO TO 38
3624      38 FTL=FTL*(PRIME-DB(J))*2*O.85*FPC/4.
3625      39 PBY=PHIV*(CC+CSR+CSC+CSSUM-FTL-FTSUM-FTP)
3626      C
3627      C COMPUTE DESIGN STRENGTH IN COMPRESSION CONTROL REGION
3628      C
3629      C USING HALPING INTERVAL TECHNIQUE TO FIND LOCATION OF NEUTRAL AXIS
3630      C
3631      XL=X0
3632      XU=CU/BI
3633      40 XC=(XL+XU)/2.
3634      XN=XU-XL
3635      IF(TCODE.EQ.1) DIFFER=6
3636      IF(TCODE.EQ.2) DIFFER=O.2
3637      IF(ABS(XN).LT.DIFFER) GO TO 41
3638      AC=BI*XC
3639      C
3640      C COMPUTE STRAINS AT VARIOUS BARS AND CHECK THEY YIELD OR NOT
3641      C
3642      IF(TL.LE.O) GO TO 3
3643      DO 42 NI=1,TL
3644      ECSV(NI)=((XC-O.5*CX+X(NI))-ECU)/XC
3645      IF(ABS(ECSV(NI)).GT.ESMAX) GO TO 86
3646      GO TO 88
3647      88 ECSV(NI)=(ESMAX+ECSV(NI))/ABS(ECSV(NI))
3648      89 ETSY(NI)=((X(NI)-O.5*CX-XC)-ECU)/XC
3649      IF(ABS(ETSY(NI)).GT.ESMAX) GO TO 90
3650      GO TO 42
3651      90 ETSY(NI)=(ESMAX+ETSY(NI))/ABS(ETSY(NI))
3652      42 CONTINUE
3653      C
3654      C CHECK ECSR AND ETSL AND ESC EQUAL TO ZERO OR NOT
3655      C
3656      3 GO TO (10,43,44,44),TRFACE
3657      44 ECSR=((XC-CP)-ECU)/XC
3658      IF(ABS(ECSR).GT.ESMAX) GO TO 91
3659      GO TO 92
3660      91 ECSR=(ESMAX+ECSR)/ABS(ECSR)
3661      92 ETSL=((O-XC)-ECU)/XC
3662      IF(ABS(ETSL).GT.ESMAX) GO TO 93
3663      GO TO 94
3664      93 ETSL=(ESMAX+ETSL)/ABS(ETSL)
3665      94 IF(NO.EQ.O) GO TO 48
3666      43 IF(NO/2.-MOD.LE.O.0001) GO TO 48
3667      ESC=((XC-O.5*CX)-ECU)/XC
3668      IF(ABS(ESC).GT.ESMAX) GO TO 95
3669      GO TO 48
3670      95 ESC=(ESMAX+ESC)/ABS(ESC)
3671      GO TO 48
3672      48 ESC=O.
3673      C
3674      C CALCULATE FORCES IN SECTION
3675      C
3676      49 GO TO (10,50,51,52),TRFACE
3677      C
3678      C BASED ON ECSR=O AND ETSL=O
3679      C
3680      50 CSR=O.
3681      FTL=O.
3682      CC=O.85*FPC*AC*CY
3683      C
3684      C COMPUTE POINT OF APPLICATION FOR CC
3685      C
3686      XCC=CX/2.-O.5*AC
3687      GO TO 55
3688      51 CSR=(NX*PRIME-DB(J))*2*(ES+ECSR-O.85*FPC)/4.
3689      FTL=(NX*PRIME-DB(J))*2*ES+ETSL/4.
3690      CC=O.85*FPC*AC*CY
3691      XCC=CX/2.-O.5*AC
3692      IF(TL.LE.O) GO TO 53
3693      GO TO 55
3694      C
3695      C DETERMINE AREA OF SEGMENT IN CIRCULAR REINFORCEMENT PATTERN
3696      C
3697      52 IF(AC.LE.RR) GO TO 56
3698      PP=AC-RR
3699      GO TO 57
3700      56 PP=RR-AC
3701      57 Q=RR**2-PP**2
3702      QQ=SQRT(Q)
3703      PR=PP/RR
3704      RFAD=ARCOS(PR)
3705      IF(AC.GT.RR) GO TO 56
3706      GO TO 57
3707      58 RFAD=PRIME-RFAD
3708      59 AREA=(CX**2*(RFAD-SIN(RFAD)+COS(RFAD)))/4.
3709      SEG=(CX**3*(SIN(RFAD)))/12.
3710      C
3711      C COMPUTE POINT OF APPLICATION FOR CC
3712      C
3713      XCC=SEG/AREA
3714      CC=O.85*FPC*AREA
3715      CSR=(PRIME-DB(J))*2*(ES+ECSR-O.85*FPC)/4.
3716      FTL=(PRIME-DB(J))*2*ES+ETSL/4.
3717      55 DO 60 NI=1,TL
3718      ECSV(NI)=(2.*PRIME-DB(J))*2*(ES+ECSV(NI)-O.85*FPC)/4.
3719      FTY(NI)=(2.*PRIME-DB(J))*2*ES+ETSY(NI)/4.
3720      60 CONTINUE

```

```

3721 53 IF(ESC.LE.O.) GO TO 54
3722 CSC=(2.*PRIME+DB(J))*2*(ES+ESC-O.85+FPC)/4.
3723 GO TO 46
3724 54 CSC=O.
3725 C
3726 C CHECK TO SEE ALL FTY(NI) AND FTL ARE ACTUALLY A TENSION OR NOT
3727 C
3728 46 IF(TL.LE.O) GO TO 62
3729 NI=1
3730 61 IF(FTY(NI).GT.O.) GO TO 62
3731 FTY(NI)=FTY(NI)*(2.*PRIME+DB(J))*2*O.85+FPC)/4.
3732 NI=NI+1
3733 IF(NI.LE.TL) GO TO 61
3734 62 IF(FTL.GE.O.) GO TO 63
3735 IF(TRACE.EQ.4) GO TO 64
3736 FTL=FTL+(NX*PRIME+DB(J))*2*O.85+FPC)/4.
3737 GO TO 63
3738 64 FTL=FTL+(PRIME+DB(J))*2*O.85+FPC)/4.
3739 63 IF(TL.LE.O) GO TO 6
3740 GO TO 7
3741 FTY(1)=O.
3742 CSY(1)=O.
3743 7 FUN=F22(FTY,CSY,X,FTL,CX,DP,EX,TL,CSR,CSC,CC,XCC)
3744 IF(FUN) 55,41,55
3745 55
3746 NX=XC
3747 GO TO 40
3748 XL=XC
3749 GO TO 40
3750 41 FTSUM=O.
3751 CSSUM=O.
3752 IF(TL.LE.O) GO TO 8
3753 DO 67 NI=1,TL
3754 FTSUM=FTSUM+FTY(NI)
3755 CSSUM=CSSUM+CSY(NI)
3756 67 CONTINUE
3757 8 PHV=CSR+CSSUM+CSC+CC-FTL-FTSUM
3758 FTSUM1=O.
3759 CSUM1=O.
3760 IF(TL.LE.O) GO TO 9
3761 DO 100 NI=1,TL
3762 FTSUM1=FTY(NI)*X(NI)+FTSUM1
3763 CSUM1=CSY(NI)*X(NI)+CSUM1
3764 100 CONTINUE
3765 MNV=FTL*(O.5+CX-DP)+CSR*(O.5+CX-DP)+CC+XCC+FTSUM1+CSUM1
3766 PUDESX=PHV/MNV
3767 MUDESX=PHV/MNV
3768 10 RETURN
3769 END
3770 C
3771 C FUNCTION F22(FTY,CSY,X,FTL,CX,DP,EX,TL,CSR,CSC,CC,XCC)
3772 C
3773 C TAKING MOMENT OF INTERNAL FORCES ABOUT APPLIED AXIAL LOAD
3774 C AND WRITING AN EQUATION IN TERMS OF FUNCTION OF UNKNOWN
3775 C NEUTRAL AXIS
3776 C
3777 IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
3778 DIMENSION FTY(12),CSY(12),X(12)
3779 FTSUM=O.
3780 CSUM=O.
3781 IF(TL.LE.O) GO TO 100
3782 DO 10 NI=1,TL
3783 FTSUM=FTSUM+FTY(NI)*(X(NI)-EX)
3784 CSUM=CSUM+CSY(NI)*(X(NI)-EX)
3785 10 CONTINUE
3786 100 F22=FTSUM+CSUM+FTL*(O.5+CX-DP+EX)+CSR*(O.5+CX-DP-EX)-CSC*EX
3787 -CC*(EX-XCC)
3788 RETURN
3789 END
3790 C
3791 C *****
3792 C "
3793 C SUBROUTINE ADEQU2
3794 C "
3795 C *THIS SUBROUTINE CHECKS DESIGN STRENGTH CAPACITY WITH THE
3796 C *REQUIRED STRENGTH IN BIAXIAL BENDING
3797 C "
3798 C *****
3799 C
3800 IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
3801 COMMON PG,ESMAX,TYPE,TREINF,TLEN1,TBEND,TLEN2,TADEO,TPROB
3802 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TSS(9),DS(9),
3803 * PD,PL,ECU,PMAX,PMIN,SBT,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
3804 * TBMAX,SMAX,J,PRIME,MINDIM,TESTOI,LATDS(20),MINLAT,TBMAXX,
3805 COMMON PPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
3806 * EPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BOY,TOUT,TMODE
3807 COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
3808 COMMON TDIMFC,AG,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
3809 * Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MCH,
3810 * MAPPX,PUDESX,MUDESX,PBX
3811 COMMON EX,MCY,MAPPY,PUDESX,MUDESX,PHV,MVYREO,MOKREO,
3812 * YC,YB,ECS(12),ETS(12),CS(12),FT(12),
3813 * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
3814 COMMON TESTMO
3815 COMMON NLC,JPOS,TREVIS,TDSS,SPACE,PITCH,NSPACR
3816 DIMENSION A(8)
3817 EQUIVALENCE (MOK,MUDESX),(MVY,MUDESX),(MX,MAPPX),(MY,MAPPY)
3818 C
3819 IF(TOUT.NE.1) GO TO 4000
3820 IF(TMODE.EQ.1) WRITE(7,400)
3821 WRITE(8,400)
3822 400 FORMAT(' ',2X,'ADEQUACY CHECK FOR BIAXIAL BENDING'/IX,
3823 * TO(' ')/)
3824 4000 PDASH=O.1+FPC*AG
3825 C
3826 C DETERMINE WHETHER CHECKING DESIGN AXIAL LOAD OR DESIGN MOMENT
3827 C
3828 IF(PU.GE.PDASH) GO TO 10
3829 GO TO 11
3830 C
3831 C USING BRESLER RECIPROCAL LOAD METHOD
3832 C
3833 10 PO=PHIV*(O.85+FPC*(AG-AST))+FY*AST
3834 P10=1./PUDESX+1./PUDESX-1./PO
3835 C
3836 C COMPUTE PROVIDED AXIAL LOAD
3837 C
3838 PUPRO=1./P10
3839 RATPX=PUPRO/PU
3840 MOREC=RATPX-1.O

```

```

3841      IF(PUPRO.GE.PU) GO TO 85
3842      C
3843      C SET A TOLERANCE LIMIT OF 0.03 FOR COMPARING PROVIDED
3844      C AXIAL LOAD WITH APPLIED AXIAL LOAD
3845      C
3846      IF(MOREC+0.03.GE.O.) GO TO 85
3847      IF(TDESIG.EQ.1) GO TO 34
3848      75 IF(TMODE.EQ.1) GO TO 2001
3849      GO TO 131
3850      2001 WRITE(7,215)RATPX
3851      131 WRITE(8,215)RATPX
3852      IF(MOREC+0.03.GE.O.) GO TO 2880
3853      IF(TMODE.EQ.1) WRITE(7,2881)
3854      WRITE(8,2881)
3855      2881 FORMAT(' ','BX','== SECTION IS NOT ADEQUATE FOR THIS',
3856      ' ' ' LOADING =='/)
3857      C
3858      C PROCEED TO NEXT PROBLEM RESULTING FROM UNSATISFACTORY SECTION
3859      C
3860      TADEQ=1
3861      RETURN
3862      34 IF(TMODE.EQ.1) GO TO 2002
3863      GO TO 132
3864      2002 IF(TOUT.NE.1) GO TO 132
3865      WRITE(7,215)RATPX
3866      132 WRITE(8,215)RATPX
3867      C
3868      C GO TO SUBROUTINE MODIF1 TO INCREASE REINFORCEMENT
3869      C
3870      TADEQ=2
3871      RETURN
3872      85 IF(TDESIG.NE.1) GO TO 75
3873      IF(TMODE.EQ.1) GO TO 2016
3874      GO TO 148
3875      2016 WRITE(7,217) RATPX
3876      148 WRITE(8,217) RATPX
3877      GO TO 70
3878      C
3879      C COMPUTE NEW CAPACITY REDUCTION FACTOR, PHIVN
3880      11 PBXX=PBX/PHIV
3881      PSYY=PSY/PHIV
3882      PDS=AMIN1(PBXX,PSYY)
3883      PDAA=AMIN1(PDS,PDASH)
3884      PHIVN=PHIV+((0.9-PHIV)*(PDAA-PU))/PDAA
3885      IF(PHIVN.LT.PHIV) PHIVN=PHIV
3886      MOX=MOX*PHIVN/PHIV
3887      MOY=MOY*PHIVN/PHIV
3888      MX=MX*PHIVN/PHIV
3889      MY=MY*PHIVN/PHIV
3890      PN=PU/PHIVN
3891      SM=MOX/MOY
3892      C
3893      C USING LOAD CONTOUR METHOD:PARME AND GOUWENS APPROACH
3894      C
3895      CCC=PPC*CX*CY
3896      CSC=PY*AST
3897      C
3898      C CALCULATE B25, THE MINIMUM VALUE OF BB WHICH OCCURS AT
3899      C P=0.25*CCC
3900      C
3901      IF(CSC/CCC.GE.O.5) GO TO 16
3902      B25=0.545+0.35*(0.5-CSC/CCC)**2
3903      GO TO 17
3904      16 B25=0.485+0.03*(CCC/CSC)
3905      17 IF(PN/CCC.GE.O.25) GO TO 18
3906      BB=B25+(0.25-PN/CCC)**2*(0.65-0.5*(CSC/CCC))
3907      GO TO 19
3908      18 BB=B25+(0.2*(PN/CCC-0.25))/(0.65+CSC/CCC)
3909      C
3910      C DETERMINE WHICH PORTION OF THE BILINEAR INTERACTION
3911      C CURVE IS TO BE USED
3912      C
3913      C DETERMINE EQUIVALENT REQUIRED UNIAXIAL MOMENT
3914      C
3915      19 MOXREQ=MX+(MY*SM)/((1.-BB)/BB)
3916      MOYREQ=MY+(MX/SM)/((1.-BB)/BB)
3917      C
3918      C USING EQN. B1 : A STRAIGHT-LINE APPROXIMATION TO THE LOAD CONTOUR
3919      C
3920      IF(MY/MX.GE.1./SM) GO TO 20
3921      B1=MX/MOXREQ+(MY/MOYREQ)/((1.-BB)/BB)
3922      IF(TCODE.EQ.1) GO TO 1990
3923      MOXRE=MOXREQ/12000.
3924      GO TO 1991
3925      1990 MOXRE=MOXREQ/1000000.
3926      1991 IF(TMODE.EQ.1) GO TO 2017
3927      GO TO 147
3928      2017 IF(TOUT.NE.1) GO TO 147
3929      WRITE(7,200) MOXRE
3930      147 WRITE(8,200) MOXRE
3931      GO TO 21
3932      20 B1=(MX/MOXREQ)/((1.-BB)/BB)+MY/MOYREQ
3933      IF(TCODE.EQ.1) GO TO 1992
3934      MOYRE=MOYREQ/12000.
3935      GO TO 1993
3936      1992 MOYRE=MOYREQ/1000000.
3937      1993 IF(TMODE.EQ.1) GO TO 2018
3938      GO TO 148
3939      2018 IF(TOUT.NE.1) GO TO 148
3940      WRITE(7,201) MOYRE
3941      148 WRITE(8,201) MOYRE
3942      C
3943      C SET TOLERANCE LIMIT OF 0.03
3944      C
3945      21 IF(B1.LE.1.03) GO TO 22
3946      IF(TDESIG.EQ.1) GO TO 23
3947      IF(TMODE.EQ.1) GO TO 2020
3948      GO TO 150
3949      2020 WRITE(7,203)
3950      150 WRITE(8,203)
3951      C
3952      C PROCEED TO NEXT PROBLEM RESULTING FROM B1 VALUE > 1.0
3953      C
3954      TADEQ=1
3955      RETURN
3956      23 IF(TMODE.EQ.1) GO TO 2021
3957      GO TO 151
3958      2021 IF(TOUT.NE.1) GO TO 151
3959      WRITE(7,204)
3960      151 WRITE(8,204)

```

```

3961 C
3962 C GO TO SUBROUTINE MODIFI TO INCREASE REINFORCEMENT
3963 C
3964 TADEQ=2
3965 RETURN
3966 22 IF(TDESIG.NE.1) GO TO 133
3967 IF(TMODE.EQ.1) GO TO 2022
3968 GO TO 152
3969 2022 WRITE(7,205)
3970 152 WRITE(8,205)
3971 GO TO 70
3972 133 IF(TMODE.EQ.1) WRITE(7,420)
3973 WRITE(8,420)
3974 TADEQ=1
3975 RETURN
3976 C
3977 C RETURN TO MAIN ROUTINE AND CONTINUE EXECUTION
3978 C
3979 70 TADEQ=3
3980 RETURN
3981 C
3982 C ***FORMAT STATEMENTS***
3983 C
3984 200 FORMAT(' ',SX,'EQUIVALENT REQUIRED MOMENT ABOUT X-AXIS',
3985 ' /SX, ONLY =',F8.0,' IN PROJECT UNITS')
3986 201 FORMAT(' ',SX,'EQUIVALENT REQUIRED MOMENT ABOUT Y-AXIS',
3987 ' /SX, ONLY =',F8.0,' IN PROJECT UNITS')
3988 203 FORMAT(' ',SX,'ENTERED SECTION IS NOT ADEQUATE BASED ON/SX,
3989 'MOMENT CAPACITIES')
3990 204 FORMAT(' ',SX,'SELECTED SECTION IS NOT ADEQUATE BASED ON/SX,
3991 'MOMENT CAPACITIES, INCREASING REINFORCEMENT ')
3992 205 FORMAT(' ',SX,'SELECTED SECTION IS ADEQUATE BASED ON/SX,
3993 'MOMENT CAPACITIES')
3994 215 FORMAT(' ',SX,'FOR SECTION ENTERED THE AXIAL LOAD CAPACITY',
3995 ' /SX, UNDER APPLIED MOMENT IS',
3996 ' FT.3, TIMES/SX, THE APPLIED AXIAL LOAD')
3997 216 FORMAT(' ',SX,'FOR SECTION SELECTED THE AXIAL LOAD CAPACITY',
3998 ' /SX, UNDER APPLIED MOMENT IS',
3999 ' FT.3, TIMES/SX, THE APPLIED AXIAL LOAD, AND IS NOT',
4000 ' ACCEPTABLE, /SX, INCREASING REINFORCEMENT ')
4001 217 FORMAT(' ',SX,'FOR SECTION SELECTED THE AXIAL LOAD CAPACITY',
4002 ' /SX, UNDER APPLIED MOMENT IS',
4003 ' FT.3, TIMES/SX, THE APPLIED AXIAL LOAD')
4004 420 FORMAT(' ',SX,'ENTERED SECTION IS ADEQUATE BASED ON/SX,
4005 'MOMENT CAPACITIES')
4006 END
4007 C
4008 C *****
4009 C
4010 SUBROUTINE MODIFI
4011 C
4012 C *THIS SUBROUTINE MODIFIES SECTIONS
4013 C
4014 C *****
4015 C
4016 IMPLICIT REAL(A-N,I,K-M,P,R,S,W-V), INTEGER(T,J,N,S)
4017 COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEQ,TPROB
4018 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CVINCR,T1,TK,TBS(8),DB(9),
4019 ' FD,FL,ECU,PMAX,PMIN,S81,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
4020 ' TSMAX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TSMAXX
4021 COMMON FPC,FY,FSY,EC,B1,TKBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
4022 ' SPC,PD,PL,PU,MX1,MX2,SDX,MV1,MV2,BDV,TOUT,TMODE
4023 COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
4024 COMMON TDIMFG,AG,AS,CC,SB,DP,AST,NX,NY,SBX,SBY,DS,
4025 ' Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MCK,
4026 ' MAPPX,PUDEXX,MUDEXX,PBX
4027 COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PBY,MOYREQ,MOXREQ,
4028 ' YC,YB,ECS(12),ETS(12),CS(12),FT(12),
4029 ' XC,XB,ECSY(12),ETSY(12),CSY(12),PTY(12)
4030 COMMON TESTMO
4031 COMMON NLC,JPOS,TREVIS,TDSSS,SPACE,PITCH,NSPACR
4032 DIMENSION A(8)
4033 EQUIVALENCE (MOX,MUDEXX),(MOY,MUDESY),(MX,MAPPX),(MY,MAPPY)
4034 C
4035 IF(TOUT.NE.1) GO TO 13
4036 IF(TMODE.EQ.1) WRITE(7,400)
4037 WRITE(8,400)
4038 400 FORMAT(' ',MODIFICATION INFORMATION :')
4039 C
4040 C CHECK REQUIRED TOTAL NO. OF BARS IS ENTERED OR NOT
4041 C
4042 13 IF(NBS.NE.0) GO TO 7
4043 C
4044 C CHECK MAX. PERMISSIBLE NO. OF BARS IS USED OR NOT
4045 C
4046 IF(NMAX.LT.TBMAX) GO TO 8
4047 C
4048 C INCREASE BAR SIZE
4049 C
4050 7 IF(J.GE.TK) GO TO 10
4051 J=J+1
4052 TMODAS=1
4053 GO TO 20
4054 C
4055 C INCREASE TOTAL NO. OF BARS USED
4056 C
4057 8 NMAX=NMAX+2
4058 TMODAS=2
4059 GO TO 20
4060 C
4061 C CHECK INCREASING COLUMN DIMENSIONS IS ALLOWED OR NOT
4062 C
4063 10 GO TO(14,15,16,17,18),TDIMFG
4064 14 IF(TOUT.NE.1) GO TO 130
4065 IF(TMODE.EQ.1) GO TO 2000
4066 GO TO 130
4067 2000 WRITE(7,322)
4068 130 WRITE(8,322)
4069 C
4070 C PROCEED TO NEXT PROBLEM RESULTING FROM LIMITING CONSTRAINTS
4071 C
4072 TESTMO=2
4073 RETURN
4074 C
4075 C BACK TO SUBROUTINE REINF1 TO REITERATE THE STEPS
4076 C
4077 20 TESTMO=1
4078 RETURN
4079 15 CY=CY+CVINCR
4080 GO TO 22

```

```

4081 16 CX=CX+CXINCR
4082 GO TO 22
4083 17 IF(CX.GE.CY) GO TO 3200
4084 CX=CX+CXINCR
4085 CY=CY/R
4086 CY=CYINCR+IFIX(CY/CVINCR)
4087 GO TO 3201
4088 3200 CY=CY+CVINCR
4089 CX=CY/R
4090 CX=CXINCR+IFIX(CX/CXINCR)
4091 3201 IF(TSHAPE.EQ.3) GO TO 65
4092 GO TO 22
4093 65 AG=(PRIME+CX**2)/4.
4094 GO TO 66
4095 18 IF(CX.GE.CY) GO TO 15
4096 CX=CX+CXINCR
4097 22 AG=CY
4098 C
4099 C SET FLAG STATING THAT A NEW TRIAL STEEL SECTION IS SELECTED
4100 C
4101 66 TMOAS=0
4102 AS=PMIN+AG
4103 IF(TOUT.NE.1) GO TO 4002
4104 IF(TSHAPE.EQ.3) GO TO 2880
4105 IF(TMODE.EQ.1) GO TO 2002
4106 GO TO 132
4107 2002 WRITE(7,203) CX,CY
4108 132 WRITE(8,203) CX,CY
4109 GO TO 4002
4110 2880 IF(TMODE.EQ.1) WRITE(7,2881)CX
4111 WRITE(8,2881)CX
4112 2881 FORMAT(' ',6X,'MODIFIED CONCRETE DIMENSION :'/2X,'DIAMETER ',
4113 ' F6.1,2X,'IN PROJECT UNITS'/)
4114 C
4115 C BACK TO SUBROUTINE REINF1 TO REITERATE THE STEPS
4116 C
4117 4002 TESTMO=1
4118 RETURN
4119 C
4120 C ***FORMAT STATEMENTS***
4121 C
4122 203 FORMAT(' ',6X,'MODIFIED CONCRETE DIMENSION :',
4123 '2X,'CX= ',F6.1,2X,'CY= ',F6.1,2X,'IN PROJECT UNITS'/)
4124 322 FORMAT(' ',6X,'DESIGN CANNOT BE COMPLETED'/6X,
4125 'CONSISTENT WITH ENTERED RESTRAINTS'/)
4126 C
4127 END
4128 C
4129 C *****
4130 C
4131 SUBROUTINE SPIRS1
4132 C
4133 C THIS SUBROUTINE CHECKS REINFORCEMENT WITH LAP SPLICES AND
4134 C SELECTS LATERAL REINFORCEMENT FOR SPIRAL COLUMN
4135 C
4136 C *****
4137 C
4138 IMPLICIT REAL(A-N,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
4139 COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEO,TPROB
4140 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CVINCR,TI,YK,TBS(9),DB(9),
4141 PD,PL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
4142 TBMAX,SMAX,J,PRIME,MINDIM,TESTD1,LATDS(20),MINLAT,TBMAXX
4143 COMMON PPC,FY,FSY,EC,S1,TXBRAC,KK,LUX,TYBRAC,KY,LUY,SPU,
4144 SPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BOY,TOUT,TMODE
4145 COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
4146 COMMON YDIMP,AG,AS,CG,SB,DP,AST,NX,NV,SBY,DS,
4147 V(12),ISEX,X(12),ISEY,TMOAS,PGDES,EY,MCX,
4148 HAPPH,PUDESX,MUDESX,PSX
4149 COMMON EX,MCV,HAPPY,PUDESY,MUDESY,PSY,MOVREQ,MOXREQ,
4150 VC,VB,ECS(12),ETS(12),CS(12),FT(12),
4151 KC,KB,ECSV(12),ETSV(12),CSV(12),FTV(12)
4152 COMMON TESTMO
4153 COMMON NLC,JPOB,TREVIS,TDSSE,SPACE,PITCH,NSPACR
4154 DIMENSION A(9)
4155 C
4156 C CALCULATE CENTER-TO-CENTER SPACING OF SPIRAL BARS
4157 C
4158 IF(TOUT.NE.1) GO TO 3001
4159 IF(TMODE.EQ.1) WRITE(7,300)
4160 WRITE(8,300)
4161 GO TO 3001
4162 300 FORMAT(' ',6X,'SELECT SPIRAL :'/)
4163 3000 SMIN=DSS+SSMIN
4164 SMAX=DSS+SSMAX
4165 S3=SMAX-SMIN
4166 C
4167 C DETERMINE MIN. RATIO OF SPIRAL REINFORCEMENT, PSMIN
4168 C
4169 DC=CK-2.*DCC
4170 ACORE=(PRIME+DC**2)/4.
4171 PSMIN=0.45*(AG/ACORE-1.)*(PPC/FSY)
4172 C
4173 C COMPUTE MAX. SPACING, S1
4174 C
4175 S1=(PRIME+DSS**2*(DC-DSS))/(DC**2+PSMIN)
4176 IF(TCODE.EQ.1) GO TO 10
4177 PITCH=AMIN1(S1,S3)
4178 GO TO 11
4179 C
4180 C DETERMINE DISTANCE BETWEEN TURNS OF SPIRALS
4181 C
4182 10 S2=DC/6.
4183 PITCH=AMIN1(S1,S2,S3)
4184 IF(TOUT.NE.1) GO TO 11
4185 IF(TMODE.EQ.1) GO TO 2002
4186 GO TO 132
4187 2002 WRITE(7,202)TDSSE,PITCH
4188 132 WRITE(8,202)TDSSE,PITCH
4189 202 FORMAT(' ',6X,'#',I2,1X,',', PITCH =',F6.1,2X,
4190 'IN PROJECT UNITS'/)
4191 C
4192 C DETERMINE NUMBER OF SPACERS NEEDED FOR SPIRALS
4193 C
4194 11 IF(TCODE.EQ.1) GO TO 12
4195 CCX1=20.
4196 CCX2=30.
4197 CCX3=24.
4198 CCX4=0.625
4199 GO TO 2000
4200 12 CCX1=500.

```

```

4201      CCX2=750.
4202      CCX3=600.
4203      CCX4=19.5
4204      2000 IF(DSS.LT.CCX4) GO TO 13
4205      IF(CX.LE.CCX3) GO TO 14
4206      17 NSPACR=4
4207      GO TO 15
4208      14 NSPACR=3
4209      GO TO 15
4210      13 IF(CX.LT.CCX1) GO TO 16
4211      IF(CX.LT.CCX2) GO TO 14
4212      GO TO 17
4213      16 NSPACR=2
4214      15 IF(TOUT.NE.1) RETURN
4215      IF(TMODE.EQ.1) GO TO 2001
4216      GO TO 131
4217      2001 WRITE(7,201) NSPACR
4218      131 WRITE(8,201) NSPACR
4219      201 FORMAT(' ',EX,'NUMBER OF SPACERS REQUIRED= ',13)
4220      RETURN
4221      C
4222      C DETERMINE WHETHER REINFORCEMENT CAN BE ACCOMMODATED WITH
4223      C LAP SPLICES OR NOT
4224      C
4225      3001 IF(TCODE.EQ.1) GO TO 135
4226      BAR=DB(4)
4227      BARL=DB(7)
4228      GO TO 136
4229      135 BAR=DB(3)
4230      BARL=DB(6)
4231      136 IF(DB(J).GE.BAR) GO TO 137
4232      A2=(SB+DB(J))/(CX-3.*DB(J)-2.*(DCC+DSS))
4233      NN=IFIX(180./ARSIN(A2)+0.51)
4234      A3=(SB+DB(J))/(CX-DB(J)-2.*(DCC+DSS))
4235      A4=DB(J)/(CX-DB(J)-2.*(DCC+DSS))
4236      NT=IFIX(180./(ARSIN(A3)+ARSIN(A4))+0.51)
4237      GO TO 138
4238      137 A2=(2.5*DB(J))/(CX-3.*DB(J)-2.*(DCC+DSS))
4239      NN=IFIX(180./ARSIN(A2)+0.51)
4240      A3=(2.5*DB(J))/(CX-DB(J)-2.*(DCC+DSS))
4241      A4=DB(J)/(CX-DB(J)-2.*(DCC+DSS))
4242      NT=IFIX(180./(ARSIN(A3)+ARSIN(A4))+0.51)
4243      C
4244      C CHECK LAP SPLICES CAN BE USED OR NOT
4245      C
4246      138 IF(DB(J).GT.BARL) GO TO 151
4247      GO TO 145
4248      C
4249      C CHECK NORMAL LAP SPLICES
4250      C
4251      141 IF(NMAX.GT.NN) GO TO 143
4252      GO TO 3000
4253      151 IF(TMODE.EQ.1) WRITE(7,152) BARL
4254      WRITE(8,152) BARL
4255      GO TO 3000
4256      143 IF(TMODE.EQ.1) WRITE(7,146)
4257      WRITE(8,146)
4258      GO TO 3000
4259      C
4260      C CHECK TANGENTIAL LAP SPLICES
4261      C
4262      145 IF(NMAX.GT.NT) GO TO 147
4263      GO TO 141
4264      147 IF(TMODE.EQ.1) WRITE(7,150)
4265      WRITE(8,150)
4266      GO TO 3000
4267      C
4268      C
4269      146 FORMAT(' ',EX,'NORMAL LAP SPLICES CAN BE USED')
4270      150 FORMAT(' ',EX,'TANGENTIAL LAP SPLICES CAN BE USED')
4271      152 FORMAT(' ',EX,'LAP SPLICES SHOULD NOT BE USED')
4272      * ' FOR DIAMETER//EX,'BAR LARGER THAN',FB.4,1X,
4273      * 'IN PROJECT UNITS')
4274      END
4275      C
4276      C *****
4277      C
4278      C SUBROUTINE TIES1
4279      C
4280      C *THIS SUBROUTINE SELECTS LATERAL REINFORCEMENT FOR TIED COLUMN *
4281      C
4282      C *****
4283      C
4284      IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y),INTEGER(T,J,N,S)
4285      COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEQ,TPROB
4286      COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9)
4287      * PD,PL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
4288      * TBMAX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAXX
4289      COMMON FPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
4290      * SPC,PD,PL,PU,MX1,MX2,BDX,MY1,MY2,SDY,TOUT,TMODE
4291      COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
4292      COMMON TDIMFG,AG,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
4293      * Y(12),ISEX,X(12),ISEY,TMODAS,PGDES,EY,MCK,
4294      * MAPPX,PUDESX,MUDESX,PSX
4295      COMMON EX,MCY,HAPPY,PUDESY,MUDESY,PSY,MOYREQ,MOXREQ,
4296      * YC,YB,ECSY(12),ETSY(12),CS(12),FT(12),
4297      * XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
4298      COMMON TESTMD
4299      COMMON NLC,JPOB,TREVIS,TDSSS,SPACE,PITCH,NSPACR
4300      DIMENSION A(8)
4301      C
4302      C SELECT COLUMN TIES SPACING REQUIREMENT PER SET; S
4303      C
4304      IF(TOUT.NE.1) GO TO 3002
4305      IF(TMODE.EQ.1) WRITE(7,304)
4306      WRITE(8,304)
4307      GO TO 3002
4308      3003 SD=18.*DB(J)
4309      SS=48.*DSS
4310      SPACE=AMIN1(SD,SS,CX,CY)
4311      IF(TOUT.NE.1) GO TO 3000
4312      IF(TMODE.EQ.1) GO TO 3001
4313      GO TO 3100
4314      3001 WRITE(7,206)TDSSS,SPACE
4315      3100 WRITE(8,206) TDSSS,SPACE
4316      C
4317      C DETERMINE THE LATERAL TIES REQUIRED FOR INTERIOR BARS
4318      C AND ALL CORNER BARS
4319      C
4320      3000 GO TO(10,10,11,14),TRFACE

```

```

4321 10 IF(SB.LE.SMAX) GO TO 13
4322 IF(TOUT.NE.1) GO TO 20
4323 IF(TMODE.EQ.1) GO TO 2000
4324 GO TO 130
4325 2000 WRITE(7,200)
4326 130 WRITE(8,200)
4327 GO TO 20
4328 13 IF(TOUT.NE.1) GO TO 20
4329 IF(TMODE.EQ.1) GO TO 2001
4330 GO TO 131
4331 2001 WRITE(7,201)
4332 131 WRITE(8,201)
4333 20 IF(NMAX/2.GE.4) GO TO 132
4334 GO TO 14
4335 132 IF(TMODE.EQ.1) GO TO 2002
4336 GO TO 133
4337 2002 WRITE(7,300)
4338 133 WRITE(8,300)
4339 GO TO 14
4340 C
4341 C FOR BARS IN BOTH FACES
4342 C
4343 11 IF(SBX.LE.SMAX) GO TO 15
4344 IF(TOUT.NE.1) GO TO 16
4345 IF(TMODE.EQ.1) GO TO 2003
4346 GO TO 134
4347 2003 WRITE(7,202)
4348 134 WRITE(8,202)
4349 GO TO 16
4350 15 IF(TOUT.NE.1) GO TO 16
4351 IF(TMODE.EQ.1) GO TO 2004
4352 GO TO 135
4353 2004 WRITE(7,203)
4354 135 WRITE(8,203)
4355 16 IF(SBY.LE.SMAX) GO TO 17
4356 IF(TOUT.NE.1) GO TO 21
4357 IF(TMODE.EQ.1) GO TO 2005
4358 GO TO 136
4359 2005 WRITE(7,204)
4360 136 WRITE(8,204)
4361 GO TO 21
4362 17 IF(TOUT.NE.1) GO TO 21
4363 IF(TMODE.EQ.1) GO TO 2006
4364 GO TO 137
4365 2006 WRITE(7,205)
4366 137 WRITE(8,205)
4367 21 IF(NX.GE.4) GO TO 138
4368 GO TO 143
4369 138 IF(TMODE.EQ.1) GO TO 2007
4370 GO TO 139
4371 2007 WRITE(7,301)
4372 139 WRITE(8,301)
4373 143 IF(NY.GE.2) GO TO 140
4374 GO TO 14
4375 140 IF(TMODE.EQ.1) GO TO 2008
4376 GO TO 141
4377 2008 WRITE(7,302)
4378 141 WRITE(8,302)
4379 14 RETURN
4380 C
4381 C DETERMINE WHETHER REINFORCEMENT CAN BE ACCOMMODATED WITH
4382 C LAP SPLICES
4383 C
4384 3002 GO TO(145,146,147,14),TRFACE
4385 C
4386 C COMPUTE NO. OF BARS PER FACE BETWEEN SPLICES
4387 C
4388 145 NF=NMAX/2
4389 S4=CY
4390 GO TO 148
4391 146 NF=NMAX/2
4392 S4=CX
4393 GO TO 148
4394 147 IF(NX.GT.NY) GO TO 170
4395 NF=NY
4396 S4=CX
4397 GO TO 148
4398 170 NF=NX
4399 S4=CY
4400 148 IF(TCODE.EQ.1) GO TO 149
4401 BAR=DB(4)
4402 BARL=DB(7)
4403 GO TO 150
4404 149 BAR=DB(3)
4405 BARL=DB(5)
4406 150 IF(DB(J).GE.BAR) GO TO 151
4407 S1=2.*(DCC+DSS)+NF*DB(J)+(NF-1.)*S1
4408 A1=(1.-0.5+0.5)*DB(J)/(S1+DB(J))
4409 META=ARCSIN(A1)
4410 S2=S1+(2.*S1+2.*DB(J))*COS(META)-0.546*DB(J)-2.*S1
4411 S3=2.*(DCC+DSS)+(2.*NF-1.)*DB(J)+(NF-1.)*S1
4412 GO TO 152
4413 151 S1=2.*(DCC+DSS)+NF*DB(J)+(NF-1.)*1.5*DB(J)
4414 S2=S1+1.38*DB(J)
4415 S3=2.*(DCC+DSS)+(2.*NF-1.)*DB(J)+(NF-1.)*1.5*DB(J)
4416 C
4417 C CHECK LAP SPLICES CAN BE USED OR NOT
4418 C
4419 152 IF(DB(J).GT.BARL) GO TO 155
4420 GO TO 158
4421 C
4422 C CHECK FOR NORMAL LAP SPLICES
4423 C
4424 155 IF(S4.GT.S2) GO TO 157
4425 GO TO 3003
4426 156 IF(TMODE.EQ.1) WRITE(7,156) BARL
4427 WRITE(8,156) BARL
4428 GO TO 3003
4429 157 IF(TMODE.EQ.1) WRITE(7,160)
4430 WRITE(8,160)
4431 GO TO 3003
4432 C
4433 C CHECK FOR TANGENTIAL LAP SPLICES
4434 C
4435 159 IF(S4.GT.S3) GO TO 161
4436 GO TO 155
4437 161 IF(TMODE.EQ.1) WRITE(7,163)
4438 WRITE(8,163)
4439 GO TO 3003
4440 C

```

```

4441 C      ***FORMAT STATEMENTS***
4442 C
4443 200 FORMAT(' ',SX,'TIES ARE REQUIRED ON INTERMEDIATE BARS',
4444 ' ',BECAUSE BAR SPACING'/SX,'EXCEEDS MAXIMUM CLEAR BAR',
4445 ' ',SPACING.'/')
4446 201 FORMAT(' ',SX,'OMIT TIES ON INTERMEDIATE BARS BECAUSE BAR
4447 '/SX,'SPACING DOES NOT EXCEED MAXIMUM CLEAR BAR SPACING'/)
4448 202 FORMAT(' ',SX,'TIES ARE REQUIRED ON INTERMEDIATE BARS IN',
4449 ' ',XFACE BECAUSE BAR'/SX,'SPACING EXCEEDS MAXIMUM CLEAR',
4450 ' ',BAR SPACING.'/')
4451 203 FORMAT(' ',SX,'OMIT TIES ON INTERMEDIATE BARS IN XFACE',
4452 ' ',BECAUSE BAR'/SX,'SPACING DOES NOT EXCEED MAXIMUM CLEAR BAR',
4453 ' ',SPACING.'/')
4454 204 FORMAT(' ',SX,'TIES ARE REQUIRED ON INTERMEDIATE BARS IN',
4455 ' ',YFACE BECAUSE BAR'/SX,'SPACING EXCEEDS MAXIMUM CLEAR',
4456 ' ',BAR SPACING.'/')
4457 205 FORMAT(' ',SX,'OMIT TIES ON INTERMEDIATE BARS IN YFACE',
4458 ' ',BECAUSE BAR'/SX,'SPACING DOES NOT EXCEED MAXIMUM CLEAR BAR',
4459 ' ',SPACING.'/')
4460 206 FORMAT(' ',SX,'P',12,1X,'@ ',F8.1,2X,'IN PROJECT UNITS',
4461 ' ',SPACING.'/')
4462 300 FORMAT(' ',SX,'LATERAL TIES REQUIRED FOR INTERIOR BARS',
4463 ' ',/SX,'IN ALTERNATE FACE'/)
4464 301 FORMAT(' ',SX,'LATERAL TIES REQUIRED FOR INTERIOR BARS',
4465 ' ',/SX,'IN X-FACE'/)
4466 302 FORMAT(' ',SX,'LATERAL TIES REQUIRED FOR INTERIOR BARS',
4467 ' ',/SX,'IN Y-FACE'/)
4468 304 FORMAT(' ',SELECT TIES :'/)
4469 160 FORMAT(' ',SX,'NORMAL LAP SPLICES CAN BE USED'/)
4470 163 FORMAT(' ',SX,'TANGENTIAL LAP SPLICES CAN BE USED'/)
4471 166 FORMAT(' ',SX,'LAP SPLICES SHOULD NOT BE USED',
4472 ' ',FOR DIAMETER'/SX,'BAR LARGER THAN',F8.4,1X,
4473 ' ',IN PROJECT UNITS'/)
4474 C
4475 END
4476 C
4477 C *****
4478 C
4479 SUBROUTINE OUTPUT
4480 C
4481 C THIS SUBROUTINE IS FORMATTED A SHORT OUTPUT LAYOUT
4482 C
4483 C *****
4484 C
4485 IMPLICIT REAL(A-H,I,K-M,P,R,S,W-V),INTEGER(T,J,N,S)
4486 COMMON PG,ESMAX,TYPE,TREINF,TSLEN1,TBEND,TSLEN2,TADEQ,TPROB
4487 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TBS(9),DB(9),
4488 ' FD,FL,ECU,PMAX,PMIN,SBI,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
4489 ' TBMAX,SHAX,J,PRIME,MINDIM,TESTD1,LATDS(201),MINLAT,TBMAXX
4490 COMMON PPC,FY,FSY,EC,B1,TXBRAC,KX,LUX,TYBRAC,KY,LUY,SPU,
4491 ' SPC,PD,PL,PU,MX1,MX2,BDX,MY1,MY2,BDY,TOUT,TMODE
4492 COMMON TSHAPE,CX,CY,R,TDESIG,TRFACE,NBS,NMAX
4493 COMMON TBINFO,AC,AS,CG,SB,DP,AST,NX,NY,SBX,SBY,DS,
4494 ' V(12),ISEX,X(12),ISSEY,TMODAS,PGDES,EY,MCK,
4495 ' MAPPX,PUDEXX,MUDESX,PRX
4496 COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PSY,MOYREQ,MOXREQ,
4497 ' YC,YB,ECS(12),ETS(12),CS(12),FT(12),
4498 ' XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
4499 COMMON TESTMO
4500 COMMON NLC,JPOS,TREVIS,TDSSS,SPACE,PITCH,NSPACR
4501 C
4502 C ONLY USED FOR REVISING PROBLEM IN INTERACTIVE MODE
4503 C
4504 IF(TREVIS.EQ.1) GO TO 206
4505 IF(TREVIS.EQ.2) GO TO 212
4506 IF(TREVIS.EQ.3) GO TO 212
4507 IF(TREVIS.EQ.4) GO TO 208
4508 IF(TREVIS.EQ.5) GO TO 211
4509 C
4510 C ONLY USED FOR ADEQUACY CHECK IN MULTIPLE LOAD CASES
4511 C
4512 IF(TDESIG.EQ.2) GO TO 203
4513 IF(TYPE.EQ.1.AND.TCODE.EQ.1) GO TO 227
4514 GO TO 231
4515 227 WRITE(6,100)
4516 IF(TMODE.EQ.1) WRITE(7,100)
4517 231 IF(TYPE.EQ.1.AND.TCODE.EQ.2) GO TO 228
4518 GO TO 232
4519 228 WRITE(6,101)
4520 IF(TMODE.EQ.1) WRITE(7,101)
4521 232 IF(TYPE.EQ.2.AND.TCODE.EQ.1) GO TO 229
4522 GO TO 233
4523 229 WRITE(6,102)
4524 IF(TMODE.EQ.1) WRITE(7,102)
4525 233 IF(TYPE.EQ.2.AND.TCODE.EQ.2) GO TO 230
4526 GO TO 208
4527 230 WRITE(6,103)
4528 IF(TMODE.EQ.1) WRITE(7,103)
4529 208 LX=KX*LUX
4530 IF(TXBRAC.EQ.1) GO TO 204
4531 IF(TCODE.EQ.1) GO TO 150
4532 WRITE(6,109)LX
4533 IF(TMODE.EQ.1) WRITE(7,109)LX
4534 GO TO 205
4535 150 WRITE(6,151)LX
4536 IF(TMODE.EQ.1) WRITE(7,151)LX
4537 GO TO 205
4538 204 IF(TCODE.EQ.1) GO TO 152
4539 WRITE(6,110)LX
4540 IF(TMODE.EQ.1) WRITE(7,110)LX
4541 GO TO 205
4542 152 WRITE(6,153)LX
4543 IF(TMODE.EQ.1) WRITE(7,153)LX
4544 205 IF(TBEND.EQ.1) GO TO 215
4545 LY=KY*LUY
4546 IF(TYBRAC.EQ.1) GO TO 207
4547 IF(TCODE.EQ.1) GO TO 154
4548 WRITE(6,111)LY
4549 IF(TMODE.EQ.1) WRITE(7,111)LY
4550 GO TO 215
4551 154 WRITE(6,155)LY
4552 IF(TMODE.EQ.1) WRITE(7,155)LY
4553 GO TO 215
4554 207 IF(TCODE.EQ.1) GO TO 156
4555 WRITE(6,112)LY
4556 IF(TMODE.EQ.1) WRITE(7,112)LY
4557 GO TO 215
4558 156 WRITE(6,157)LY
4559 IF(TMODE.EQ.1) WRITE(7,157)LY
4560 C

```

```

4561 C ONLY USED FOR REVISING STABILITY GEOM. PROPERTIES IN
4562 C INTERACTIVE MODE
4563 C
4564 215 IF(TREVIS.EO.4) GO TO 213
4565 206 WRITE(6,104)
4566 IF(TMDE.EO.1) WRITE(7,104)
4567 IF(TCODE.EO.1) GO TO 200
4568 WRITE(6,105)FPC,WC,EC,FY,ES
4569 IF(TMDE.EO.1) WRITE(7,105)FPC,WC,EC,FY,ES
4570 GO TO 201
4571 200 WRITE(6,106)FPC,WC,EC,FY,ES
4572 IF(TMDE.EO.1) WRITE(7,106)FPC,WC,EC,FY,ES
4573 201 IF(TYPE.EO.1) GO TO 202
4574 IF(TCODE.EO.1) GO TO 209
4575 WRITE(6,107)FSY
4576 IF(TMDE.EO.1) WRITE(7,107)FSY
4577 GO TO 203
4578 209 WRITE(6,113)FSY
4579 IF(TMDE.EO.1) WRITE(7,113)FSY
4580 GO TO 203
4581 202 IF(TCODE.EO.1) GO TO 210
4582 WRITE(6,108)FSY
4583 IF(TMDE.EO.1) WRITE(7,108)FSY
4584 GO TO 203
4585 210 WRITE(6,114)FSY
4586 IF(TMDE.EO.1) WRITE(7,114)FSY
4587 C
4588 C ONLY USED FOR REVISING MATERIAL PROPERTIES IN INTERACTIVE MODE
4589 C
4590 203 IF(TREVIS.EO.1) GO TO 213
4591 211 IF(TDESIG.EO.1) WRITE(6,2880)
4592 IF(TDESIG.EO.2) WRITE(6,2881)
4593 IF(TDESIG.EO.3) WRITE(6,2882)
4594 IF(TCODE.EO.1) GO TO 212
4595 MPU=PU/1000.
4596 MMX2=ABS(MX2)/12000.
4597 MMV2=ABS(MY2)/12000.
4598 WRITE(6,115)MPU,MMX2,MMY2
4599 IF(TMDE.EO.1) WRITE(7,115)MPU,MMX2,MMY2
4600 GO TO 214
4601 MPU=PU/1000.
4602 MMX2=ABS(MX2)/1000000.
4603 MMV2=ABS(MY2)/1000000.
4604 WRITE(6,116)MPU,MMX2,MMY2
4605 IF(TMDE.EO.1) WRITE(7,116)MPU,MMX2,MMY2
4606 214 IF(TDESIG.EO.2) GO TO 225
4607 213 WRITE(6,117)
4608 IF(TMDE.EO.1) WRITE(7,117)
4609 IF(TCODE.EO.1) GO TO 216
4610 IF(TSHAPE.EO.3) GO TO 2883
4611 WRITE(6,118)CX,CY
4612 IF(TMDE.EO.1) WRITE(7,118)CX,CY
4613 GO TO 217
4614 2883 WRITE(6,2884)CX
4615 IF(TMDE.EO.1) WRITE(7,2884)CX
4616 GO TO 217
4617 216 IF(TSHAPE.EO.3) GO TO 2885
4618 WRITE(6,119)CX,CY
4619 IF(TMDE.EO.1) WRITE(7,119)CX,CY
4620 GO TO 217
4621 2885 WRITE(6,2886)CX
4622 IF(TMDE.EO.1) WRITE(7,2886)CX
4623 C
4624 C ONLY USED FOR EXISTING DIAGNOSED ERROR IN SUBROUTINE DIMEN1
4625 C
4626 217 IF(YESDI.EO.2) GO TO 226
4627 IF(TRFACE.EO.3) NMAX=2*(NX+NY)
4628 PGDES=(NMAX*PRIME+DB(J)**2)/(AG*4.)
4629 WRITE(6,120)PGDES
4630 IF(TMDE.EO.1) WRITE(7,120)PGDES
4631 GO TO(218,219,220,221),TRFACE
4632 218 WRITE(6,121)NMAX,TBS(J)
4633 IF(TMDE.EO.1) WRITE(7,121)NMAX,TBS(J)
4634 GO TO 222
4635 219 WRITE(6,122)NMAX,TBS(J)
4636 IF(TMDE.EO.1) WRITE(7,122)NMAX,TBS(J)
4637 GO TO 222
4638 220 NNX=2*NX
4639 NNY=2*NY
4640 WRITE(6,123)NNX,TBS(J),NNY,TBS(J)
4641 IF(TMDE.EO.1) WRITE(7,123)NNX,TBS(J),NNY,TBS(J)
4642 GO TO 222
4643 221 WRITE(6,124)NMAX,TBS(J)
4644 IF(TMDE.EO.1) WRITE(7,124)NMAX,TBS(J)
4645 222 IF(SPACE.LE.0..AND.PITCH.LE.0.) GO TO 226
4646 IF(TYPE.EO.1) GO TO 223
4647 IF(TCODE.EO.1) GO TO 224
4648 WRITE(6,126)TDSSS,PITCH,NSPACR
4649 IF(TMDE.EO.1) WRITE(7,126)TDSSS,PITCH,NSPACR
4650 GO TO 226
4651 224 WRITE(6,127)TDSSS,PITCH,NSPACR
4652 IF(TMDE.EO.1) WRITE(7,127)TDSSS,PITCH,NSPACR
4653 GO TO 226
4654 223 IF(TCODE.EO.1) GO TO 226
4655 WRITE(6,128)TDSSS,SPACE
4656 IF(TMDE.EO.1) WRITE(7,128)TDSSS,SPACE
4657 GO TO 226
4658 225 WRITE(6,129)TDSSS,SPACE
4659 IF(TMDE.EO.1) WRITE(7,129)TDSSS,SPACE
4660 226 WRITE(6,130)
4661 IF(TMDE.EO.1) WRITE(7,130)
4662 C
4663 C EVALUATE THE ENTERED VALUES OF MATERIAL STRENGTH
4664 C
4665 IF(TCODE.EO.1) GO TO 545
4666 IF(FPC.GE.2000..AND.FPC.LE.6000.) GO TO 540
4667 WRITE(6,530)
4668 540 IF(FY.GE.40000..AND.FY.LE.60000.) GO TO 541
4669 WRITE(6,531)
4670 541 IF(FSY.GE.40000..AND.FSY.LE.60000.) GO TO 542
4671 WRITE(6,532)
4672 GO TO 542
4673 545 IF(FPC.GE.20..AND.FPC.LE.40.) GO TO 543
4674 WRITE(6,530)
4675 543 IF(FY.GE.300..AND.FY.LE.400.) GO TO 544
4676 WRITE(6,531)
4677 544 IF(FSY.GE.300..AND.FSY.LE.400.) GO TO 542
4678 WRITE(6,532)
4679 C
4680 C DETERMINE COLUMN DEFLECT IN SINGLE OR DOUBLE CURVATURE

```

```

4681 C
4682 542 IF(ABS(MX1).LE.O.) GO TO 328
4683 IF(MX1.LT.O..AND.MX2.GT.O.) GO TO 327
4684 IF(MX2.LT.O..AND.MX1.GT.O.) GO TO 327
4685 WRITE(6,303)
4686 IF(TMODE.EQ.1) WRITE(7,303)
4687 GO TO 328
4688 327 WRITE(6,304)
4689 IF(TMODE.EQ.1) WRITE(7,304)
4690 328 IF(TBEND.EQ.1) GO TO 500
4691 IF(ABS(MY1).LE.O.) GO TO 1000
4692 IF(MY1.LT.O..AND.MY2.GT.O.) GO TO 329
4693 IF(MY2.LT.O..AND.MY1.GT.O.) GO TO 329
4694 1000 WRITE(6,305)
4695 IF(TMODE.EQ.1) WRITE(7,305)
4696 GO TO 500
4697 329 WRITE(6,306)
4698 IF(TMODE.EQ.1) WRITE(7,306)
4699 C
4700 C EMPTY STORAGE PARAMETERS
4701 C
4702 500 PITCH=0.
4703 SPACE=0.
4704 RETURN
4705 C
4706 C FORMAT
4707 C
4708 100 FORMAT(' ',TYPE = TIED,15X,DESIGN CODE = CSA A23.3-M77')
4709 101 FORMAT(' ',TYPE = TIED,15X,DESIGN CODE = ACI 318-77')
4710 102 FORMAT(' ',TYPE = SPIRAL,15X,DESIGN CODE = CSA A23.3-M77')
4711 103 FORMAT(' ',TYPE = SPIRAL,15X,DESIGN CODE = ACI 318-77')
4712 104 FORMAT(' ',MATERIAL PROPERTIES '/')
4713 105 FORMAT(' ',SX,CONCRETE : COMP. STRENGTH =,F7.0,
4714 : 1X,PSI/6X, DENSITY =,F6.0,
4715 : 1X,PCF/6X, MODULUS OF ELAS. =,F11.0,
4716 : 1X,PSI/6X, LONG. STEEL : YIELD STRESS =,F8.0,
4717 : 1X,PSI/6X, MODULUS OF ELAS. =,F11.0,PSI')
4718 106 FORMAT(' ',SX,CONCRETE : COMP. STRENGTH =,F6.0,
4719 : 1X,MPA/6X, DENSITY =,F7.0,
4720 : 1X,KG/CUBIC METER,
4721 : 1X,MPA/6X, LONG. STEEL : YIELD STRESS =,F8.0,
4722 : 1X,MPA/6X, MODULUS OF ELAS. =,F11.0,MPA')
4723 107 FORMAT(' ',SX,SPIRAL : YIELD STRESS =,F8.0,
4724 : 1X,PSI/6X,
4725 : 1X,PSI/6X,
4726 108 FORMAT(' ',SX,TIES : YIELD STRESS =,F8.0,
4727 : 1X,PSI/6X,
4728 113 FORMAT(' ',SX,SPIRAL : YIELD STRESS =,F8.0,
4729 : 1X,MPA/6X,
4730 114 FORMAT(' ',SX,TIES : YIELD STRESS =,F8.0,
4731 : 1X,MPA/6X,
4732 109 FORMAT(SX,ABOUT X-AXIS : UNBRACED, EFF. LENGTH =,F7.0,
4733 : 1X,IN/6X,
4734 110 FORMAT(SX,ABOUT X-AXIS : BRACED, EFF. LENGTH =,F7.0,
4735 : 1X,IN/6X,
4736 111 FORMAT(SX,ABOUT X-AXIS : UNBRACED, EFF. LENGTH =,F7.0,
4737 : 1X,IN/6X,
4738 112 FORMAT(SX,ABOUT Y-AXIS : BRACED, EFF. LENGTH =,F7.0,
4739 : 1X,IN/6X,
4740 151 FORMAT(SX,ABOUT X-AXIS : UNBRACED, EFF. LENGTH =,F7.0,
4741 : 1X,MM/6X,
4742 153 FORMAT(SX,ABOUT X-AXIS : BRACED, EFF. LENGTH =,F7.0,
4743 : 1X,MM/6X,
4744 155 FORMAT(SX,ABOUT Y-AXIS : UNBRACED, EFF. LENGTH =,F7.0,
4745 : 1X,MM/6X,
4746 157 FORMAT(SX,ABOUT Y-AXIS : BRACED, EFF. LENGTH =,F7.0,
4747 : 1X,MM/6X,
4748 115 FORMAT(' ',SX,PU =,F8.0,1X,
4749 : KIPS/6X,MUX =,F8.0,1X,KIP-FT,2X,MUY =,F8.0,1X,
4750 : KIP-FT/6X,
4751 116 FORMAT(' ',SX,PU =,F8.0,1X,KN,
4752 : /6X,MUX =,F8.0,1X,KN-M,2X,MUY =,F8.0,1X,KN-M/6X,
4753 117 FORMAT(' ',CONCRETE DIMENSION '/')
4754 118 FORMAT(' ',SX,CX =,F6.1,1X,IN,6X,CY =,F6.1,1X,IN/6X,
4755 119 FORMAT(' ',SX,CX =,F6.0,1X,MM,6X,CY =,F6.0,1X,MM/6X,
4756 120 FORMAT(' ',LONGITUDINAL REINFORCEMENT : SX,
4757 : REINFORCEMENT RATIO =,F5.3/6X,
4758 121 FORMAT(' ',SX,12, : #,12,1X,BARS IN X-FACES ONLY')
4759 122 FORMAT(' ',SX,12, : #,12,1X,BARS IN Y-FACES ONLY')
4760 123 FORMAT(' ',SX,12, : #,12,1X,BARS IN X-FACES/6X,
4761 : 12,1X,BARS IN Y-FACES/6X,
4762 124 FORMAT(' ',SX,12, : #,12,1X,BARS IN CIRCULAR/6X,
4763 125 FORMAT(' ',SPIRALS : //,SX,12,1X,WITH,
4764 : F5.1,1X,IN,PITCH AND,12,1X,SPACERS/6X,
4765 127 FORMAT(' ',SPIRALS : //,SX,12,1X,WITH,
4766 : F6.0,1X,MM,PITCH AND,12,1X,SPACERS/6X,
4767 128 FORMAT(' ',TIES : //,SX,12,1X,1X,1X,IN,
4768 : SPACING/6X,
4769 129 FORMAT(' ',TIES : //,SX,12,1X,1X,1X,MM,
4770 : SPACING/6X,
4771 130 FORMAT(' ',ADDITIONAL COMMENTS '/')
4772 303 FORMAT(' ',SX,COLUMN DEFLECTS IN SINGLE,
4773 : CURVATURE ABOUT X-AXIS/6X,
4774 304 FORMAT(' ',SX,COLUMN DEFLECTS IN DOUBLE,
4775 : CURVATURE ABOUT X-AXIS/6X,
4776 305 FORMAT(' ',SX,COLUMN DEFLECTS IN SINGLE,
4777 : CURVATURE ABOUT Y-AXIS/6X,
4778 306 FORMAT(' ',SX,COLUMN DEFLECTS IN DOUBLE,
4779 : CURVATURE ABOUT Y-AXIS/6X,
4780 530 FORMAT(' ',SX,THE ENTERED CONCRETE STRENGTH IS NOT,
4781 : WITHIN THE/6X,PREDETERMINED RANGE')
4782 531 FORMAT(' ',SX,THE ENTERED LONGITUDINAL YIELD STRESS,
4783 : IS NOT WITHIN THE/6X,PREDETERMINED RANGE')
4784 532 FORMAT(' ',SX,THE ENTERED SPIRAL YIELD STRESS IS NOT,
4785 : WITHIN THE/6X,PREDETERMINED RANGE')
4786 2880 FORMAT(' ',DESIGN LOAD CONDITION '/')
4787 2881 FORMAT(' ',ADDITIONAL LOAD CONDITION '/')
4788 2882 FORMAT(' ',CHECK LOAD CONDITION '/')
4789 2884 FORMAT(' ',SX,DIAMETER =,F6.1,1X,IN/6X,
4790 2886 FORMAT(' ',SX,DIAMETER =,F6.0,1X,MM/6X,
4791 C
4792 END
4793 C
4794 C =====
4795 C =
4796 SUBROUTINE PROBL1
4797 C =
4798 C THIS SUBROUTINE READS OPTIONAL CONTROL PARAMETER
4799 C =
4800 C =====

```

```

4801 C
4802 IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y), INTEGER(T,J,N,S)
4803 COMMON PS,ESMAX,TYPE,TREINF,TSLEN1,TSLEN2,TADEQ,TPROB
4804 COMMON TCODE,WC,ES,DCC,DSS,CXINCR,CYINCR,TI,TK,TSS(9),DB(9),
4805 = PD,FL,ECU,PMAX,PMIN,SB1,EMIN,PHIV,RFAC,DSMIN,SSMIN,SSMAX,
4806 = TBMX,SMAX,J,PRIME,MINDIM,TESTDI,LATDS(20),MINLAT,TBMAXX
4807 COMMON FFC,TY,FSY,EC,B1,TXSRAC,KX,LUX,TYSRAC,KY,LUY,SPU,
4808 = SPC,PD,PL,PU,MX1,MX2,BDX,MV1,MV2,BDY,TOUT,TMODE
4809 COMMON TSHAPE,CX,CY,R,TDESIC,TRFACE,NBS,NMAX
4810 COMMON TDIMFC,AG,AS,CG,SD,DP,AST,NX,NY,SBX,SB'',DS,
4811 = V(12),ISEX,X(12),ISEY,TMODAS,PCDES,EY,MCX,
4812 = MAPPX,PUDESX,MUDESX,PBX
4813 COMMON EX,MCY,MAPPY,PUDESY,MUDESY,PBY,MOYREQ,MOXREQ,
4814 = VC,YB,ECS(12),ETS(12),CS(12),FT(12),
4815 = XC,XB,ECSY(12),ETSY(12),CSY(12),FTY(12)
4816 COMMON TESTMO
4817 COMMON NLC,JPOB,TREVIS,TDSS,SPACE,PITCH,NSPACR
4818
4819 C
4820 DIMENSION A(8)
4821
4822 C
4823 C EMPTY STORAGE PARAMETERS
4824 C
4825 TMODAS=0
4826 TREVIS=0
4827 TPROB=0
4828
4829 C
4830 IF(TMODE.EQ.1) GO TO 1
4831 GO TO 8
4832 1 WRITE(7,101)
4833 101 FORMAT('A NEXT OPTION= ? REVISE ABOVE PROBLEM,ENTER 1'//,
4834 = '/' GO TO NEXT PROBLEM,ENTER 2'//,
4835 = '/' TERMINATION OF PROGRAM,ENTER 3'//)
4836 READ(4,1000) JPOB
4837 GO TO (500,501,502),JPOB
4838 502 TPROB=5
4839 GO TO 6
4840 501 WRITE(7,102)
4841 102 FORMAT('A NEXT OPTION= ? ALL DATA REENTERED,ENTER 1'//,
4842 = '/' FOLLOWING CONCRETE GEOM. VARIED WITH HEADING,ENTER 2'//,
4843 = '/' FOLLOWING STEEL GEOM. VARIED WITH HEADING,ENTER 3'//,
4844 = '/' FOLLOWING STABILITY GEOM. VARIED WITH HEAD.,ENTER 4'//,
4845 = '/' FOLLOWING LOADING PROP. VARIED WITH HEADING,ENTER 5'//,
4846 = '/' LAST COLUMN DESIGN PROBLEM,ENTER 6'//)
4847 READ(4,1000)TPROB
4848 GO TO 6
4849 500 WRITE(7,103)
4850 103 FORMAT('A REVISE OPTION= ? ALTER MATERIAL ONLY,ENTER 1'//,
4851 = '/' ALTER CONCRETE GEOM. ONLY,ENTER 2'//,
4852 = '/' ALTER STEEL GEOM. ONLY,ENTER 3'//,
4853 = '/' ALTER STABILITY GEOM. ONLY,ENTER 4'//,
4854 = '/' ALTER LOADS ONLY,ENTER 5'//)
4855 READ(4,1000)TREVIS
4856 GO TO (503,504,505,506,507),TREVIS
4857 503 WRITE(8,401)
4858 GO TO 8
4859 504 WRITE(8,402)
4860 GO TO 8
4861 505 WRITE(8,403)
4862 GO TO 8
4863 506 WRITE(8,404)
4864 GO TO 8
4865 507 WRITE(8,405)
4866 GO TO 8
4867 8 READ(5,100) TPROB
4868
4869 C
4870 100 FORMAT(12)
4871 1000 FORMAT(11)
4872 401 FORMAT(' ','REVISE OPTION = ALTER MATERIAL',
4873 = ' PROPERTIES ONLY'//)
4874 402 FORMAT(' ','REVISE OPTION = ALTER CONCRETE',
4875 = ' OF GEOMETRY PROPERTIES ONLY'//)
4876 403 FORMAT(' ','REVISE OPTION = ALTER STEEL OF',
4877 = ' GEOMETRY PROPERTIES ONLY'//)
4878 404 FORMAT(' ','REVISE OPTION = ALTER STABILITY',
4879 = ' OF GEOMETRY PROPERTIES ONLY'//)
4880 405 FORMAT(' ','REVISE OPTION = ALTER LOADS PROPERTIES',
4881 = ' ONLY'//)
4882
4883 C
4884 8 RETURN
4885 END
4886
4887 C
4888 C *****
4889 C *
4890 SUBROUTINE SORT(HEAD,N,JPOB)
4891 C *
4892 C =
4893 C =THIS SUBROUTINE IS TO SORT THE COMPUTER RESULTS SO THAT
4894 C =A HEADING WILL BE PRINTED ON THE TOP OF EVERY NEW PAGE.
4895 C =ASSUMPTION : THE PRINTER TRUNCATES ANY EXTRA CHARACTERS
4896 C =IF THE LENGTH OF A PRINTED LINE HAS EXCEEDED DEFAULT LIMIT
4897 C =
4898 C *****
4899
4900 C
4901 IMPLICIT REAL(A-H,I,K-M,P,R,S,W-Y), INTEGER(T,J,N)
4902 DIMENSION LINE(50),HEAD(15)
4903
4904 C
4905 ENDFILE 8
4906 REWIND 8
4907 TNJ=8
4908 307 IF(TNJ.EQ.8) GO TO 407
4909 ENDFILE 8
4910 REWIND 8
4911
4912 C
4913 ENDFILE 8
4914 REWIND 8
4915
4916 C
4917 C NUMBER IS THE TOTAL NUMBER OF LINES PRINTED ON EACH PAGE
4918 C
4919 NUMBER=58
4920 LOCA=3
4921 NPAGE=2
4922 507 READ(5,987,END=707)LINE
4923 LOCA=LOCA+1
4924 IF (LOCA.GT. NUMBER) GO TO 507
4925 WRITE(8,987)LINE
4926 GO TO 507
4927 507 WRITE(8,988)HEAD,NPAGE
4928 WRITE(8,987)LINE
4929 LOCA=5
4930 NPAGE=NPAGE+1
4931 GO TO 507

```

```

4921      986 FORMAT('1',1X,15A4,'PAGE',13/1X,60(' ')/)
4922      987 FORMAT(50A4)
4923      707 REWIND 6
4924      REWIND 8
4925      RETURN
4926
4927      C
4928      407 READ(8,987,END=808)LINE
4929      WRITE(6,987)LINE
4930      GO TO 407
4931      808 TNJ=10
4932      IF(JPOS.EQ.1) GO TO 100
4933      IF(M.GT.1) GO TO 110
4934      GO TO 307
4935      110 N=N-1
4936      100 ENDFILE 8
4937      REWIND 6
4938      RETURN
4939      END
End of file

```

RECENT STRUCTURAL ENGINEERING REPORTS

Department of Civil Engineering

University of Alberta

84. *Finite Element Programs for Frame Analysis* by M. El Zanaty and D.W. Murray, April 1980.
85. *Test of a Prestressed Concrete Secondary Containment Structure* by J.G. MacGregor, S.H. Simmonds and S.H. Rizkalla, April 1980.
86. *An Inelastic Analysis of the Gentilly-2 Secondary Containment Structure* by D.W. Murray, C. Wong, S.H. Simmonds and J.G. MacGregor, April 1980.
87. *Nonlinear Analysis of Axisymmetric Reinforced Concrete Structures* by A.A. Elwi and D.W. Murray, May 1980.
88. *Behavior of Prestressed Concrete Containment Structures - A Summary of Findings* by J.G. MacGregor, D.W. Murray, S.H. Simmonds, April 1980.
89. *Deflection of Composite Beams at Service Load* by L. Samantaraya and J. Longworth, June 1980.
90. *Analysis and Design of Stub-Girders* by T.J.E. Zimmerman and R. Bjorhovde, August 1980.
91. *An Investigation of Reinforced Concrete Block Masonry Columns* by G.R. Sturgeon, J. Longworth and J. Warwaruk, September 1980.
92. *An Investigation of Concrete Masonry Wall and Concrete Slab Interaction* by R.M. Pacholok, J. Warwaruk and J. Longworth, October 1980.
93. *FEPARCS5 - A Finite Element Program for the Analysis of Axisymmetric Reinforced Concrete Structures - Users Manual* by A. Elwi and D.W. Murray, November 1980.
94. *Plastic Design of Reinforced Concrete Slabs* by D.M. Rogowsky and S.H. Simmonds, November 1980.
95. *Local Buckling of W Shapes Used as Columns, Beams, and Beam-Columns* by J.L. Dawe and G.L. Kulak, March 1981.
96. *Dynamic Response of Bridge Piers to Ice Forces* by E.W. Gordon and C.J. Montgomery, May 1981.
97. *Full-Scale Test of a Composite Truss* by R. Bjorhovde, June 1981.
98. *Design Methods for Steel Box-Girder Support Diaphragms* by R.J. Ramsay and R. Bjorhovde, July 1981.

99. *Behavior of Restrained Masonry Beams* by R. Lee, J. Longworth and J. Warwaruk, October 1981.
100. *Stiffened Plate Analysis by the Hybrid Stress Finite Element Method* by M.M. Hrabok and T.M. Hrudehy, October 1981.
101. *Hybslab - A Finite Element Program for Stiffened Plate Analysis* by M.M. Hrabok and T.M. Hrudehy, November 1981.
102. *Fatigue Strength of Trusses Made From Rectangular Hollow Sections* by R.B. Ogle and G.L. Kulak, November 1981.
103. *Local Buckling of Thin-Walled Tubular Steel Members* by M.J. Stephens, G.L. Kulak and C.J. Montgomery, February 1982.
104. *Test Methods for Evaluating Mechanical Properties of Waferboard: A Preliminary Study* by M. MacIntosh and J. Longworth, May 1982.
105. *Fatigue Strength of Two Steel Details* by K.A. Baker and G.L. Kulak, October 1982.
106. *Designing Floor Systems for Dynamic Response* by C.M. Matthews, C.J. Montgomery and D.W. Murray, October 1982.
107. *Analysis of Steel Plate Shear Walls* by L. Jane Thorburn, G.L. Kulak, and C.J. Montgomery, May 1983.
108. *Analysis of Shells of Revolution* by N. Hernandez and S.H. Simmonds, August 1983.
109. *Tests of Reinforced Concrete Deep Beams* by D.M. Rogowsky, J.G. MacGregor and S.Y. Ong, September 1983.
110. *Shear Strength of Deep Reinforced Concrete Continuous Beams* by D.M. Rogowsky and J.G. MacGregor, September 1983.
111. *Drilled-In Inserts in Masonry Construction* by M.A. Hatzinikolas, R. Lee, J. Longworth and J. Warwaruk, October 1983.
112. *Ultimate Strength of Timber Beam Columns* by T.M. Olatunji and J. Longworth, November 1983.
113. *Lateral Coal Pressures in a Mass Flow Silo* by A.B.B. Smith and S.H. Simmonds, November 1983.
114. *Experimental Study of Steel Plate Shear Walls* by P.A. Timler and G.L. Kulak, November 1983.
115. *End Connection Effects on the Strength of Concrete Filled HSS Columns* by S.J. Kennedy and J.G. MacGregor, April 1984.
116. *Reinforced Concrete Column Design Program* by C-K. Leung and S.H. Simmonds, April 1984.