
Doctoral Dissertations

Student Theses and Dissertations

1972

Computer assisted teaching of steel design

Wendelin Henry Mueller III

Follow this and additional works at: https://scholarsmine.mst.edu/doctoral_dissertations



Part of the [Civil Engineering Commons](#)

Department: Civil, Architectural and Environmental Engineering

Recommended Citation

Mueller, Wendelin Henry III, "Computer assisted teaching of steel design" (1972). *Doctoral Dissertations*. 211.

https://scholarsmine.mst.edu/doctoral_dissertations/211

This thesis is brought to you by Scholars' Mine, a service of the Missouri S&T Library and Learning Resources. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

COMPUTER ASSISTED TEACHING

OF

STEEL DESIGN

by

WENDELIN HENRY MUELLER, III, 1941-

A DISSERTATION

Presented to the Faculty of the Graduate School of the

UNIVERSITY OF MISSOURI-ROLLA

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

CIVIL ENGINEERING

T2764
259 pages
c. 1

1972

Oktaf Ural

Advisor

W. G. Andrews

Myron S. Papp

Joseph H. Senne

John L. Best

Wei-hen ya

226884

PUBLICATION THESIS OPTION ¹⁰

This thesis has been prepared in the style utilized for publications of the American Society of Civil Engineers. Pages 1-38 will be presented for publication. Appendices II through IV have been added for a more detailed description of the thesis work.

ABSTRACT

A method for use in creating a university level course is presented. The specifics of each of the steps is given in general terms to allow its application to as many different topics as possible. This method is then specifically applied in the formation of a fundamental steel design course using the American Institute of Steel Construction Specification. The reasoning utilized in each phase of development is documented. A lecture style presentation complemented by a series of student oriented computer programs was chosen for use. The details of the course material in outline form is given along with the computer programs written for an IBM 2741 terminal.

TABLE OF CONTENTS

	Page
ABSTRACT.....	iii
LIST OF ILLUSTRATIONS.....	v
LIST OF TABLES.....	vi
INTRODUCTION.....	1
A GENERAL PROCEDURE FOR COURSE DEVELOPMENT.....	3
COURSE OBJECTIVES.....	3
COURSE CONTENT.....	4
COURSE STRATEGY.....	4
DEVELOPMENT OF THE METHOD OF PRESENTATION.....	5
APPLICATION OF THE GENERAL DEVELOPMENT PROCEDURE TO A STRUCTURAL STEEL DESIGN COURSE.....	8
OBJECTIVES OF THE STRUCTURAL STEEL DESIGN COURSE.....	8
COURSE CONTENT AND STRATEGY OF THE STRUCTURAL STEEL DESIGN COURSE.....	12
DEVELOPMENT OF THE METHOD OF PRESENTATION FOR A STRUCTURAL STEEL DESIGN COURSE.....	25
SUGGESTED METHOD OF USE.....	32
SUMMARY AND CONCLUSIONS.....	35
ACKNOWLEDGEMENTS.....	37
APPENDIX I.-REFERENCES.....	38
VITA.....	39
ADDITIONAL APPENDICES	
APPENDIX II.-LECTURE OUTLINES.....	40
APPENDIX III.-DECISION CHARTS.....	57
APPENDIX IV.-PROGRAM DOCUMENTATION AND LISTINGS.....	72

LIST OF ILLUSTRATIONS

Figures	Page
1. Organization of Course Units.....	7
2. Objectives of a Fundamental Structural Steel Design Course...	11
3. Organizational Chart of a Computer Assisted Instruction Course.....	15
4. Design Project.....	20
5. Organizational Chart of a Teacher Administered Course with Computer Support.....	22
6. Goals of the Programs in Presenting Specific Topics.....	30

LIST OF TABLES

Tables	Page
1. Minimum Topical Outline.....	18

COMPUTER ASSISTED TEACHING
OF STEEL DESIGN

By Wendelin H. Mueller, III¹

INTRODUCTION

During the past decade, the amount and complexity of the material presented in structural design courses has been steadily on the increase. This is due, primarily, to more sophisticated design techniques being related to the various industrial codes and specifications. Along with this increase in material, the universities have been decreasing the number of credit hours allotted to teaching design. This situation initiated the idea of using the computer, as a teaching aid, to improve the presentation of steel design.

It is proposed to develop a series of student-oriented computer programs whose purpose would be to assist in teaching steel design. The programs would be in a conversational mode in order to obtain an interaction between the student and the computer as the design process unfolds. The use of the computer is suggested, principally, because of its potential in relieving the student of time-consuming, tedious arithmetical calculations, and its ability to monitor the student's design decisions. If this potential could be realized, the course material could be presented more efficiently in less class time.

¹Graduate Student, Dept. of Civ. Engrg., Univ. of Missouri-Rolla, Rolla, MO

In the initial stage of the study, it became apparent that the successful use of the proposed computer programs would depend upon the total course being structured around their use. It is therefore decided that a total structural steel design course would be created. This course would focus on the utilization of student-oriented computer programs, operating on a time sharing computer system. Before work began on the details of the study, it was found necessary to formulate a general procedure which could be followed. This procedure outlined four basic steps necessary for the creation of any course. These are:

1. Establishment of the course objectives.
2. Decision as to the specific course content.
3. Formation of the course strategy.
4. Preparation of the method of presentation.

The details of each of these steps are made general to allow their application to other courses.

After the general procedure is established, it is then specifically applied in originating a fundamental steel design course. Each step is thoroughly documented in order to give a broad understanding of the reasoning applied in making the many required decisions. This will be of value to others who may be involved in developing similar courses. In addition, it gives teachers who use this version of the steel design course the necessary background to understand the overall philosophy which was applied. This will aid him in tailoring the course to his own individual manner of presentation.

A GENERAL PROCEDURE FOR COURSE DEVELOPMENT

The first step in the project is to define explicitly a general procedure which could be used to develop a university level course. Once defined, this procedure is then applied specifically to the development of a fundamental steel design course.

Course Objectives.-In developing a course, one must first define what goals are to be achieved by teaching the course. These goals are the course objectives. There are two basic categories of course objectives - overall objectives and performance objectives (7).²

Overall objectives are those concerned with the transfer of knowledge to the student, knowledge which will help the student in his intellectual processes of reasoning and judgement. These objectives give to the student the kind of knowledge which is applicable outside the immediate realm of the course. Their purpose is to develop in the mind of the student a more mature thought process. An example of this category of objective is training the student to think analytically by teaching mathematics. The overall objectives of a course must also integrate the new knowledge being presented to the student with that which he has previously acquired (7).

Performance objectives are concerned with the specifics of the course content. The outcome sought is the acquisition and retention of information. Performance objectives involve the intellectual capacity of memory, and the students ability to form associations. Memory allows for the retention of material. Association is involved

²Numerals in parentheses refer to corresponding items in APPENDIX I.-REFERENCES.

in the direct application of the knowledge and the extension of the knowledge by recognition of relationships between ideas and experiences.

Course Content.-The next step in creating a course is to decide on the topics to be included in the course content. The purpose of the course content is to serve as a vehicle through which the course objectives can be met. This vehicle is guided by a strategy called the course strategy. Because of their interdependence, the formation of the course content and course strategy will occur simultaneously.

A valid first step in the development of the course content is to establish the minimum number of topics to be covered in the course. These topics are then used as a foundation on which to establish the final course content. As the strategy of the course evolves, the topics are expanded as required to furnish enough material to meet the course objectives. This approach guarantees the inclusion of the set of minimum number of topics in the course content. It also guides the development towards **streamlined** courses, that is, one in which the subject matter blends together without cumbersome topics included only to teach one specific concept.

During the establishment of the set of minimum topics, care must be taken to include sufficient material to make the course useful, yet keep the number limited. This allows enough time to be spent on each, insuring adequate coverage.

Course Strategy.-The purpose of the course strategy is to guarantee that the objectives of the course are achieved. It does this by acting as a control unit of the course, forming the connection between the course content and the student. As the control unit, the course

strategy decides the order and method of presenting the course content. By proper choice of the order of presentation, the course strategy forms a smooth transition from topic to topic. In so doing, the overall objective of connecting knowledge already acquired by the student, to that being presented, can begin to be achieved. The order of presentation must also insure that the material being presented is in phase with that which the student requires. This will convey to the student a sense of organization and purpose in the material being presented (4).

The course strategy has the responsibility of deciding on the method of presenting course material. The method of presentation can be used to meet specific objectives of the course. Or, it may be used to save time in order to give the opportunity to cover more material or meet other course objectives. In any case, care must be taken to choose one which complements the learning process of the students being taught. Learning, being an activity of the mind, depends not only on the material being presented but also on the way in which the mind of the student responds to the extrinsic agent presenting the material (4).

Development of the Method of Presentation.-In course development, the course strategy is the planning stage; while the development of the specific method of presentation is the production stage. The specifications of the media are made decisive by the course strategy. The media must meet or exceed these specifications in order for the course to accomplish its objectives. This phase of course development also has the responsibility of documenting the various methods of presentation in a way which is acceptable by all potential teachers of the course.

This acceptability implies that the presentation be easily understood and simple to use. To accomplish this, it must utilize the technical language of the potential teacher, thus avoiding specialized terms from other technological fields which may have been applied while developing a certain medium of presentation.

The documentation must also allow flexibility in its use. It must guide the instructor during his teaching of the course, giving him the methods of presentation and their guide lines, as set forth in the course strategy. Only in this way is the teacher allowed to interject his personality into the course presentation. This pliability will enhance the final delivery of the course material to the student.

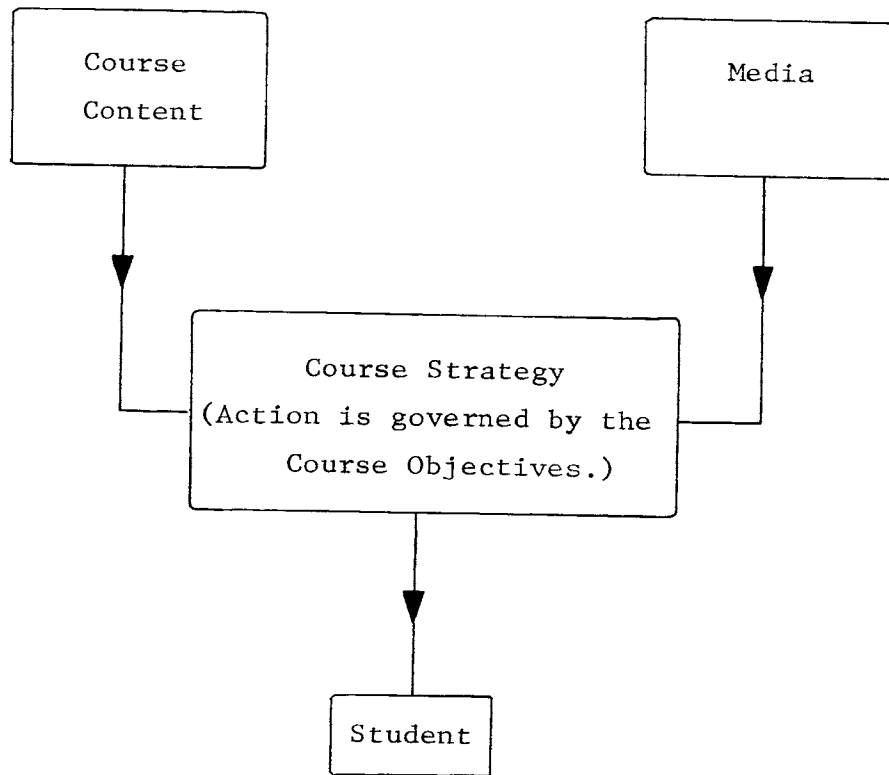


Figure 1. Organization of Course Units.

APPLICATION OF THE GENERAL DEVELOPMENT PROCEDURE
TO
A STRUCTURAL STEEL DESIGN COURSE

Using the structural steel design course for a test of the general development procedure has distinct advantages. It is a typical course taken by the undergraduate civil engineering student. Its overall purpose is to teach the fundamentals of structural steel design using the American Institute of Steel Construction's (AISC) Specification (1). The general opinion was that if the use of the procedure was a success, it would verify its usefulness in developing other courses.

Objectives of the Structural Steel Design Course - The position and the aim of the course in the educational chain of the student must be a primary consideration when choosing its objectives. The steel design course is usually attended by the student shortly after completion of his required structural analysis courses. It may be his first encounter with formal design and/or specifically steel design. After completion, he may either proceed to advanced courses or graduate with a B.S. degree.

The position of the course thus established, the choice of objectives can be made. One objective which the course must meet is the teaching of the fundamentals of structural steel design. The details of these fundamentals are defined in the course content. Because of the proximity to graduation, after which the student will begin his professional career, a certain amount of professional development should occur during the course. This professional development will help in the transition from engineering student to engineering graduate. It

must convey to the student what structural design is and its place in an engineering design project. The course must supply at least a partial answer to the question of what engineering is, its implications and ramifications.

Many students taking the steel design course will end up working in other design fields. Because of this, another course objective must be to teach steel design in such a way as to be applicable in other fields. To do this a design philosophy must be taught. This philosophy must communicate to the student that design consists of deciding the requirements of the object being designed, choosing a specific shape and size to meet these requirements and finally insuring that the object fits into and acts in harmony with the total system (2).

When this objective is specifically applied to steel design it means that the student must understand the relationship of design to structural analysis. He must acquire the knowledge and skill to choose economical members to resist loads. He must also be able to fit together the different members of a structure and make it act as an unit. The place of specifications and codes in design must also be understood. This philosophy of design is not communicated to the student in one lecture or by one problem. It is taught explicitly and implicitly over a period of time. Depending upon the approach taken by the instructor it may take as little as a few weeks or as long as the total time of the course. Any formal course must fit into the student's educational development in such a way as to form a smooth transition between the knowledge he has already attained and that which

he will acquire in the future. This starts the student from a position of knowledge about the subject and leads him into new material with a sense of confidence. Enough course material must be presented to insure that the student will begin learning advanced material from a similar position of knowledge about the subject. A fundamental steel design course must form this transition between structural analysis and advanced design. The advanced design could be done either in future courses and/or in professional practice. In every field of design the experience gained by working problems is very valuable. The experience helps him obtain the correct solution in a shorter period of time. It also gives him a "professional intuition" for the correct solution, which is useful in checking the overall correctness of a design project.

In the professional practice related to steel design, experience ranks in importance with knowledge of the fundamentals. Only with experience can an engineer acquire an instinct for accuracy and economy. Because of the importance of experience in steel design, another objective of this course is to begin the process of building a background of design experience. Figure 2 presents a summary of the objectives of the course along with its position in the educational life of the student.

It is recognized that in general the objectives of this course will overlap those of other courses. The responsibilities assumed in the development of this course concern only those objectives directly or closely associated with steel design.

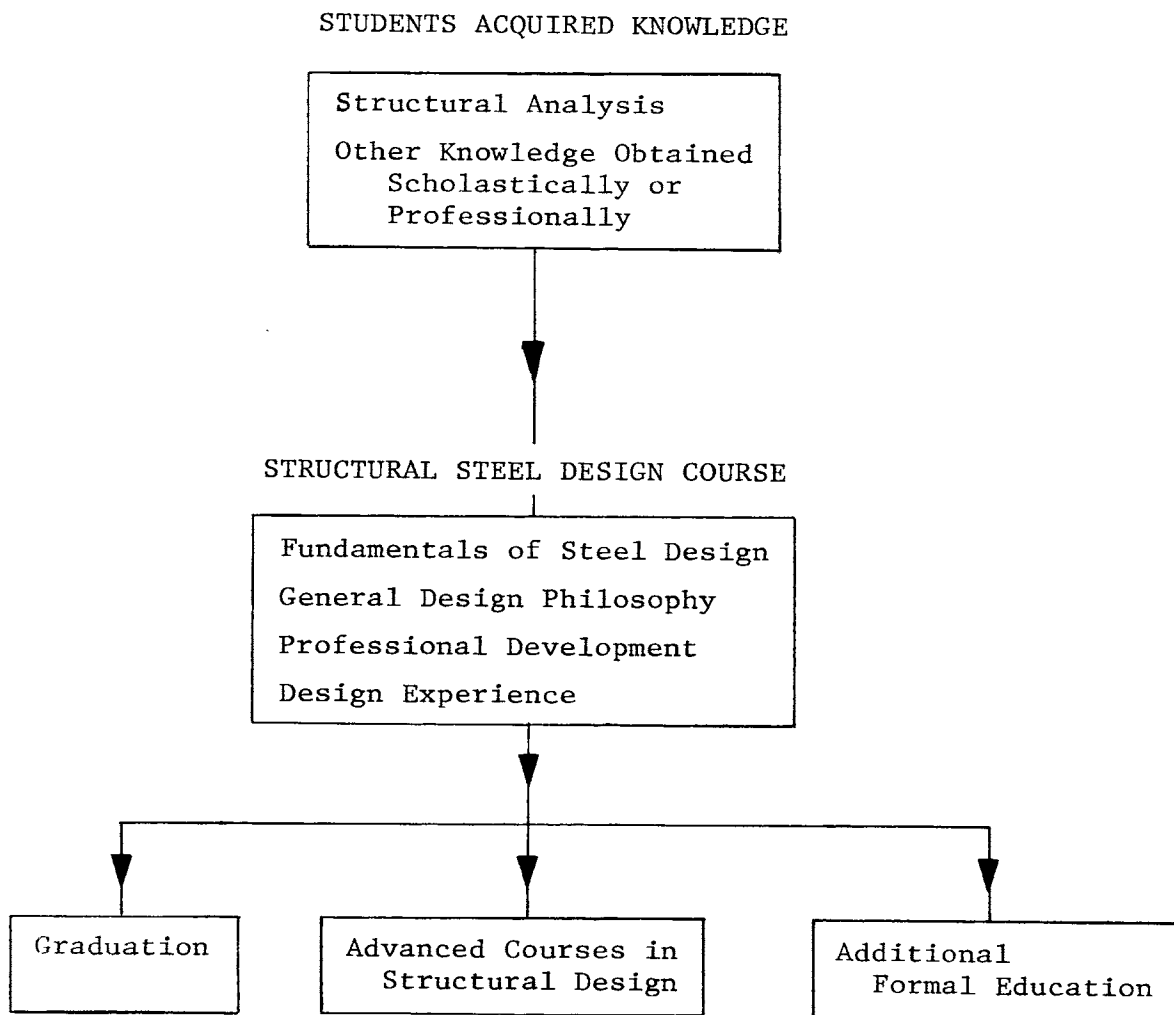


Figure 2. Objectives of a Fundamental Structural Steel Design Course.

Course Content and Strategy of the Structural Steel Design Course -

As with most courses taught at the university level, there is a limit on the amount of time allocated to teach steel design. The University of Missouri-Rolla classifies the fundamental structural steel design course as a three semester credit hour course. The classroom format is two fifty minute lecture periods and one three hour problem laboratory period per week. On an average, there would be thirty lecture and sixteen laboratory periods in a semester. To accomplish the objectives of teaching structural steel design, in the allocated time, it is necessary to use an efficient method of presenting the course material. This need for efficiency, coupled with the fact that steel design is basically a problem oriented course, suggested the use of some form of Computer Assisted Instruction (CAI)(6,8,13).

Investigation of what previously had been done in CAI revealed two facts:

First, a complete CAI course was impossible to develop under the constraints imposed on the project.

Second, it was undesirable to teach steel design using strictly the CAI technique.

To develop a complete CAI course it has been estimated that the time required would be five years (13). This implies that material for CAI presentation must be stable over a long period of time. The structural steel design course centers on the use of the American

Institute of Steel Construction's Specification³. Because of new developments in analysis techniques and material properties, this specification periodically changes. The average life of the last four adopted specifications is six years. The present AISC Specification was adopted in 1969. Assuming a six year life span of the Specification and a four year development period, the results of the project would be completed at a time when a major change to the Specification is imminent.

The resources available for the project precluded the development of a total CAI course. There are two major costs in the development of a CAI course: manpower and hardware.

According to the United States Civil Service Commission Bureau of Training, (1971, p. 5) "...a team of individuals is required [to develop CAI] consisting of: author, instructional programmer, audiovisual expert and behavior scientist, among others." (13)

Since these different types of individuals were not directly connected with the project, they could only be made available by recruiting them on a full or part time basis. This would have required a large expenditure of funds for salaries, which was just not available. Along with this large expenditure for manpower, developing a CAI course requires large investments in hardware.

³A steel design course could be taught around the American Association of State Highway Officials (AASHO), American Iron and Steel Institute (AISI) or any other such specification. The specification is a vehicle to unite material behavior with guarantees of reasonable safety margins to the public. It is minimal protection but not prohibitive to initiative. AISC was chosen for use at the University of Missouri-Rolla because it is the specification under which the largest tonnage of hot rolled steel is designed.

The United States Civil Service Commission Bureau of Training, states that (1971, p. 17) "...the computer, \$6500 per month rental for an IBM 1800 computer and peripherals...Terminals, \$7500 per month rental for 25 terminals." (13)

This does not include salaries for any of the required operating personnel. Hardware or money to obtain it was not available to be used in teaching structural steel design.

CAI is basically the use of the computer in a highly individualized and interactive tutorial system. The computer is programmed to present the course material to the student by means of a terminal device, such as, a teletype, cathode-ray tube or IBM 2741 terminal. After presentation, the student is questioned to determine whether or not he has acquired enough knowledge to proceed. If he has, new information is presented; if not, he is directed to material specifically designed to remedy the particular mistake made. If the student still does not learn, he is given additional information. This whole procedure is continued throughout the presentation of the course material.

The computer may be programmed to assign homework based on the student's needs. These needs being established by monitoring the trainee's response in the various drill sessions directed by the computer. These may also serve in directing the student to other instructional devices, such as audio tape, video tape, and models. Figure 3 shows the general organizational chart of a CAI course. It illustrates how the computer takes sole responsibility for the presentation of material. The effect of this responsibility is to have a course with individualization of instruction (13). This is often cited as the main advantage of CAI.

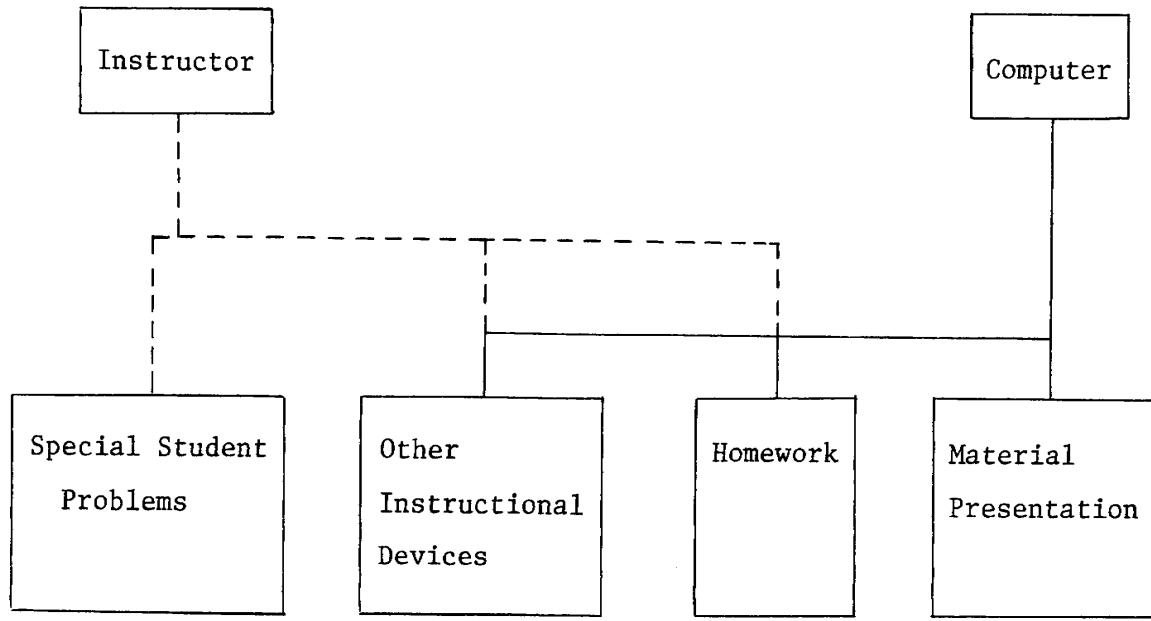


Figure 3. Organizational Chart of a Computer Assisted Instruction Course.

In presenting the disadvantages of CAI, the United States Service Commission Bureau of Training alludes to one which becomes obvious only when considering a specific subject (13).

They state (1971, p. 5)

"...It is known by people who work in the field that CAI is not appropriate for all types of subject matter, ...certain types of strategies, and obviously that subject matter which is seemingly dependent on those strategies for effective presentation are overly ambitious."

The logical means of presenting the course material in structural steel design is to use a professional designer. He can convey much of the skill needed to be a designer indirectly through his manner of presentation. As he teaches and works problems for the class, the student can observe the design procedure in detail. A procedure in which the final choice of a member is most important; the mathematics being used only guides the designer to this final choice. Through the instructor's presentation, the student can learn the parts played by safety, economy and practicality in designing, attributes of a design as important as the numbers calculated to justify the adequacy of the member. A professional manner of presenting calculations and communicating the final results, can also be taught in this way.

This is not to say that CAI could not accomplish the same objectives. However, it is believed that development of this type of CAI course would be an ambitious project, and still would not be as effective as one which depended upon a professional designer for its presentation. Thus, the decision was made to form the structural steel design course around the presentation of course material through a series of lectures by a professional designer. Although at this stage of

development CAI had been eliminated, the use of the computer was still very much under consideration.

The overall strategy indicated that a Teacher Administered Instruction Course (TAI) would be the best approach to teaching steel design. The next step was to establish whether or not a TAI course could meet all the course objectives.

The minimum topics, as established by the University of Missouri-Rolla, are presented in Table 1. These are the topics which currently must be taught in order to give to the student the fundamentals of steel design. Teaching these topics fulfills the first objective of the course. Using the TAI approach and these fundamentals, professional development within the student can also be achieved. The instructor can accomplish this directly or indirectly through his presentation of course material. The most important thing concerning this goal is that the teacher be aware that it is a course objective. He can achieve it in his own manner, drawing on example problems, personal experiences and direct discussion. Professional development will also be instilled in the student by the direct contact with a person in the profession, which the use of TAI guarantees. Developing within the student practical experience in steel design can only be achieved by working problems. These problems may consist of single member design problems or parametric studies. In any case the fundamentals as outlined in Table 1 will suffice.

Forming a connection between previous education and future knowledge can be accomplished using the fundamentals. If the instructor is aware of this objective, he can accomplish it by simply beginning

Table 1. Minimum Topical Outline

1. Introduction on structural design in metals.
2. Mechanical properties of structural metals.
3. Design of tension members.
4. Design of compression members.
5. Design of beams--laterally supported and unsupported.
6. Design of beam—columns.
7. Design of connections (riveted, welded and bolted).
8. Design of built-up members.
9. Plastic analysis and design.
10. Introduction to the computer aided design with available programs.

each new topic from an area of knowledge familiar to the student. For example, beam design can be easily introduced by a short discussion of beam analysis. The results of the analysis are then used in the design.

Structuring the connection between the knowledge being taught and that which the student will acquire later is achieved if the fundamentals of steel design are taught. From these fundamentals the student can advance to more sophisticated design problems, either via advanced courses or self study.

The last objective which must be considered is teaching a design philosophy which has application in other fields. A part of this objective can be achieved by teaching the fundamentals as outlined in Table 1. These can be used to explain the purpose of codes and specifications; and to teach the student the part of design in which a member or size of structure is chosen to meet some specified criteria.

This is only a part of design. Total design must include the fitting together of the different pieces into a total system which behaves as a unit. Teaching the fundamentals cannot convey this to the student. They must be expanded to include projects which serve as a vehicle to communicate total design philosophy to the student. An example of such a project is given in Figure 4. The student is given a structural configuration and asked to perform a complete design. He is responsible for establishing the design criteria, i.e., loads, performing the design and insuring that the members will act as a system, i.e., design connections. These basic steps in design are applicable to fields other than steel design. The design of a

Design a roof truss, simply supported by masonry walls, to span 30 ft. center to center bearing. Paragraph and page numbers refer to the AISC Manual of Steel Construction.

Given:

Distance between trusses - 30 ft.

Loads:	Live	Snow - 30 psf horizontal projection (Ref. p. 5-224).
	Dead	Truss - 5 psf Roof corrugated steel (Ref. p. 6-16).
	Crane	12 ton capacity (Ref. par. 1.3.4).
	Wind	as required.

Specification: AISC

Members: Truss Double angles.

Purlins Channels.

Steel: A-36

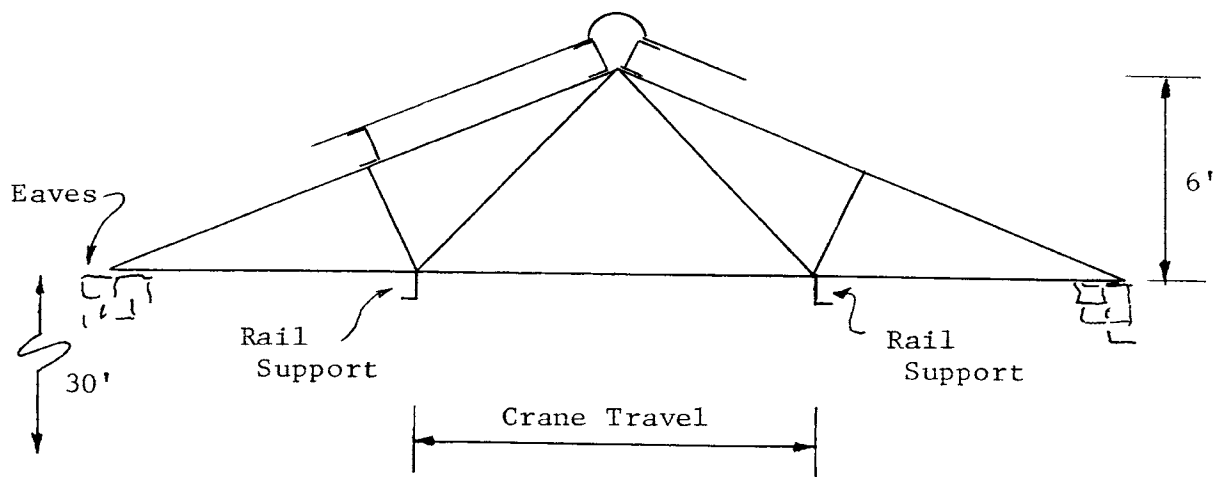


Figure 4. Design Project.

footing, earth dam all follow the same basic philosophy.

Given enough time a teacher-administered steel design course can accomplish all of the stated objectives. However, unlimited time is not available. In order to accomplish the objectives of steel design in the allotted time, a strategy must be devised which will save class time.

In teaching steel design much of the material is presented by the instructor working problems. Different problems are used to emphasize different points in the AISC Specification. A large number of problems are usually worked to teach the variations which occur in design. If another method of presenting these problems could be developed, a real savings in class time would be achieved.

A computer terminal programmed in a conversational mode (12) is an excellent medium for presenting problems. It has the capability of allowing an interaction with the student during the solution of the problem and can be programmed to present the output in the format desired. To bring into the course the use of the computer terminal and retain the presentation by the instructor the course would have to be organized as shown in Figure 5. When comparing this to the CAI's organizational chart (Figure 3), one striking difference becomes apparent. In CAI, the instructor and the computer are put on the same level, with the computer solely responsible for the presentation of the material. In Figure 5, the computer has been made subject to the instructor and the amount of responsibility the computer receives is controlled by the instructor. Making this change has virtually eliminated the major advantage of CAI, i.e., individualization

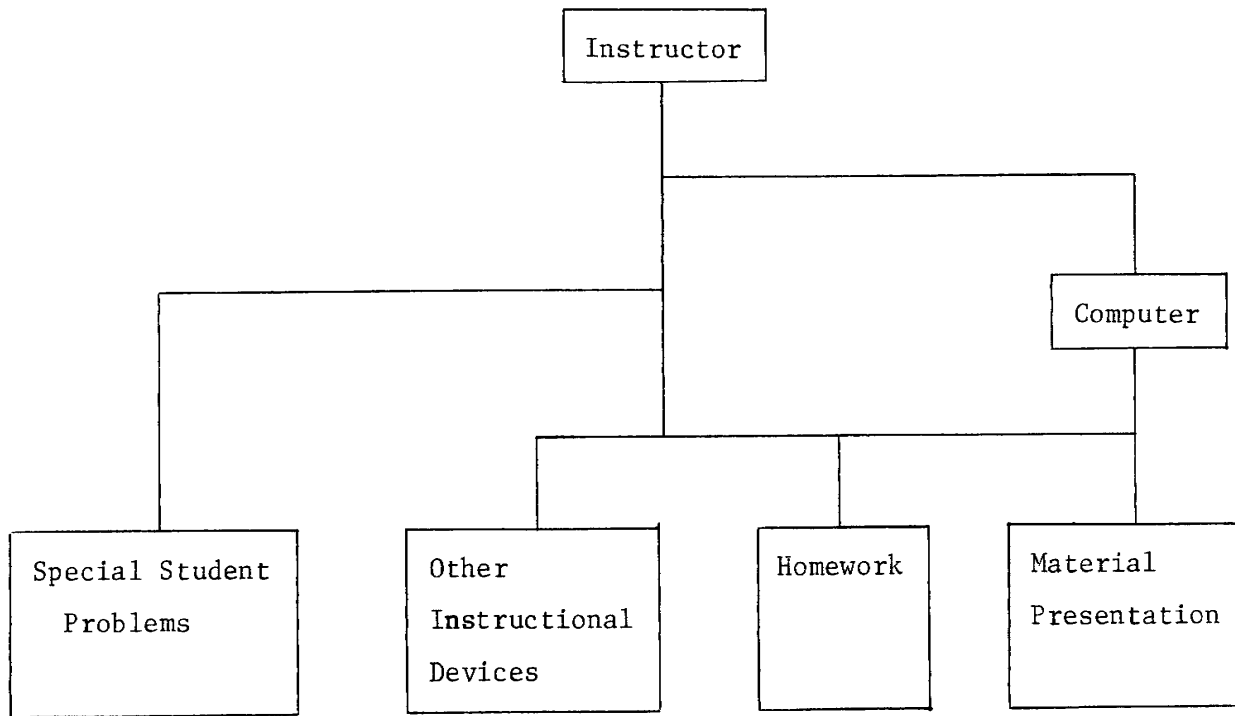


Figure 5. Organizational Chart of a Teacher Administered Course with Computer Support.

of instruction, but has brought about an advantage of flexibility. The instructor being in charge can direct the course in such a way as to gain the most from each area of the course. Through the assignment of specifically formulated problems he can rely on the terminal to present some of the variations which occur in design. This will help him realize a savings in class time.

The two extremes which limit his manner of presentation are:

- 1) To teach the course as a lecture course, using the computer to aid the student in checking his hand solutions to the problems.
- 2) To give a minimum amount of presentation required to meet the course objectives and use the computer to fill in many of the details while the student uses it to work the homework.

What is needed is a series of computer programs which will present the problems in the same fashion as the instructor would in class. The programs would require as much input and as many decisions by the student as in a hand solution.

In most structural designs there are two types of errors. The first type is one of a misinterpretation of the mathematics of the design. The second is an error in the choice of specific member, i.e., the member chosen is not adequate. The first type of error, misinterpretation of the mathematics, will not be allowed by the computer programs. This is to eliminate any negative transfer of design experience to the student and to continually emphasize the correct procedure (7). This will more closely follow the philosophy

used by the instructor in presenting problems in class. The second type of error, the member not being adequate, will be allowed by the computer programs.

This will more closely follow the normal trial and error procedure of design. Since misinterpretation of the AISC Specification is not permitted by the programs, reasons for each of the steps will be given. This will be handled in the output format.

Hayman and Lord point out the principal disadvantage when using the TAI technique in teaching. (1972, p. 2) "One difficulty with this approach (TAI) is that the main learning resource -- the classroom lecture -- is always out of phase with personal problems that occur in the student's attempts to perform the terminal objective activities. When problems occur, he must wait for the next offering of his primary resource to resolve them; even then, there is competition for recognition of personal problems. If we are interested in efficiency, from the learner's point of view, this situation is hardly ideal." (3)

The format of the output produced by the terminal will attempt to minimize this disadvantage by including reasons for the various calculations. To further minimize this disadvantage, a series of decision charts are developed. Their purpose is to give to the student a concise description of the calculations and decisions required in working a design problem. These will answer many of the questions which will occur during the students' attempt to work homework. The individual instructor can apply a strategy which will further reduce this disadvantage by allotting class time when the students are allowed to work homework and he is available to give immediate answers to their questions. This class time is freed by the proper use of the computer programs in supplementing the instructor's

lecture.

At this point, a summary of the strategy developed in the above paragraphs is warranted. A teacher administered course with computer support (TAI/CS) is being introduced. In it, the advantages of TAI and the use of the computer are amplified, while the disadvantages are minimized.

As shown in Figure 5, the responsibility for presenting the course material to the students is the responsibility of the instructor. He accomplishes this via the lectures and the assignment of specific problems to be worked by the students using the computer. Using his lectures he accomplishes the objectives of professional development and the formation of a connection between knowledge previously acquired by the student and that being taught. The lectures and the specially assigned homework are used to teach the fundamentals of steel design and begin to convey to the student the philosophy of design. The decision charts which accompany each program will answer many of the questions about the AISC Specification as they arise. Additional homework is assigned to generate design experience within the student. And finally, the special projects are used to complete teaching the overall philosophy of design and to give additional experience in steel design.

Development of the Method of Presentation for a Structural Steel

Design Course - The overall strategy defined in the previous paragraphs laid out the specifications for a lecture type presentation, supplemented by a series of conversational computer programs with supporting decision charts. One of the primary considerations of

this phase of development is to make the presentation acceptable to as many potential users as possible. To do this the presentation must communicate exactly what is required, leaving the details, which do not effect the goals of the course, up to the individual instructor.

The lecture portion of the course may be presented in different forms. The text book is a form of presentation which has been widely used in teaching. Its chief advantage is that the text can be used by both the teacher and the student as a mutual source of information. Lecture notes may also be used as a means of presentation of course content. These are sometimes reproduced and given to the student to serve as a text for the course. Both the lecture notes and the published text are excellent means of presentation of course content. They convey the course content in the exact order and precise wording of the author. This exactness, however, may be a disadvantage if the instructor of a course has a different opinion as to the best order and wording of the content. This may cause him to appear in conflict with the book while placing emphasis on other topics. It may force him to teach in a manner which is not familiar to him. Using more than one media to present the course material amplifies the disadvantages of the exactness of texts and lecture notes. Presenting the course content in the form of a topical outline minimizes this problem. The wording and emphasis is left completely up to the instructor. If the outline is set up into individual independent modules, the order of presentation is flexible.

A cross-reference of the computer programs and decision charts, included in these outlines, will assist the instructor in using them

to supplement his lectures. It is anticipated that the use of the topical outlines will require some work by the instructor in completing the details of the lecture. This amount of work will be small when compared with starting without them to prepare a lecture. Once an instructor has taught steel design, he will have compiled from these outlines, a series of custom lecture notes. He will feel comfortable teaching from these because he developed them. The outlines will have served only as a guide in the developing process. These lecture outlines are presented in Appendix II of Reference 10.

The strategy of the course required a set of computer programs to be used to save class time. They are to accomplish this by complementing the instructor's lectures through the assignment of specific problems to be worked by the students using the computer. To supplement the lecture, the programs must work the problem as the instructor would in class, printing the reasons for each step, requiring the same input and decisions. Through the use of these special problems the instructor can present new material and/or assign problems which will review material already presented.

The purpose of this phase of development is to write the necessary programs to the specification outlines by the course strategy. The computer access terminal chosen for use was an IBM 2741; the programming language is CPS-PL/I (5). It was chosen primarily because it is relatively inexpensive and available on campus. While writing the programs to meet the specifications of the course strategy, additional possible benefits of the programs began to appear. These tended to make the programs exceed the requirements set down for them.

This increased the programs' chances for success in accomplishing the savings in class time and were therefore included wherever possible.

The advantage of reviewing material already presented is amplified by the programs presenting a summary of important points before the mathematical analysis of the problem begins. In this manner the students are taught by repetition, hearing the presentation in class, and then reading a summary of that presentation when working homework.

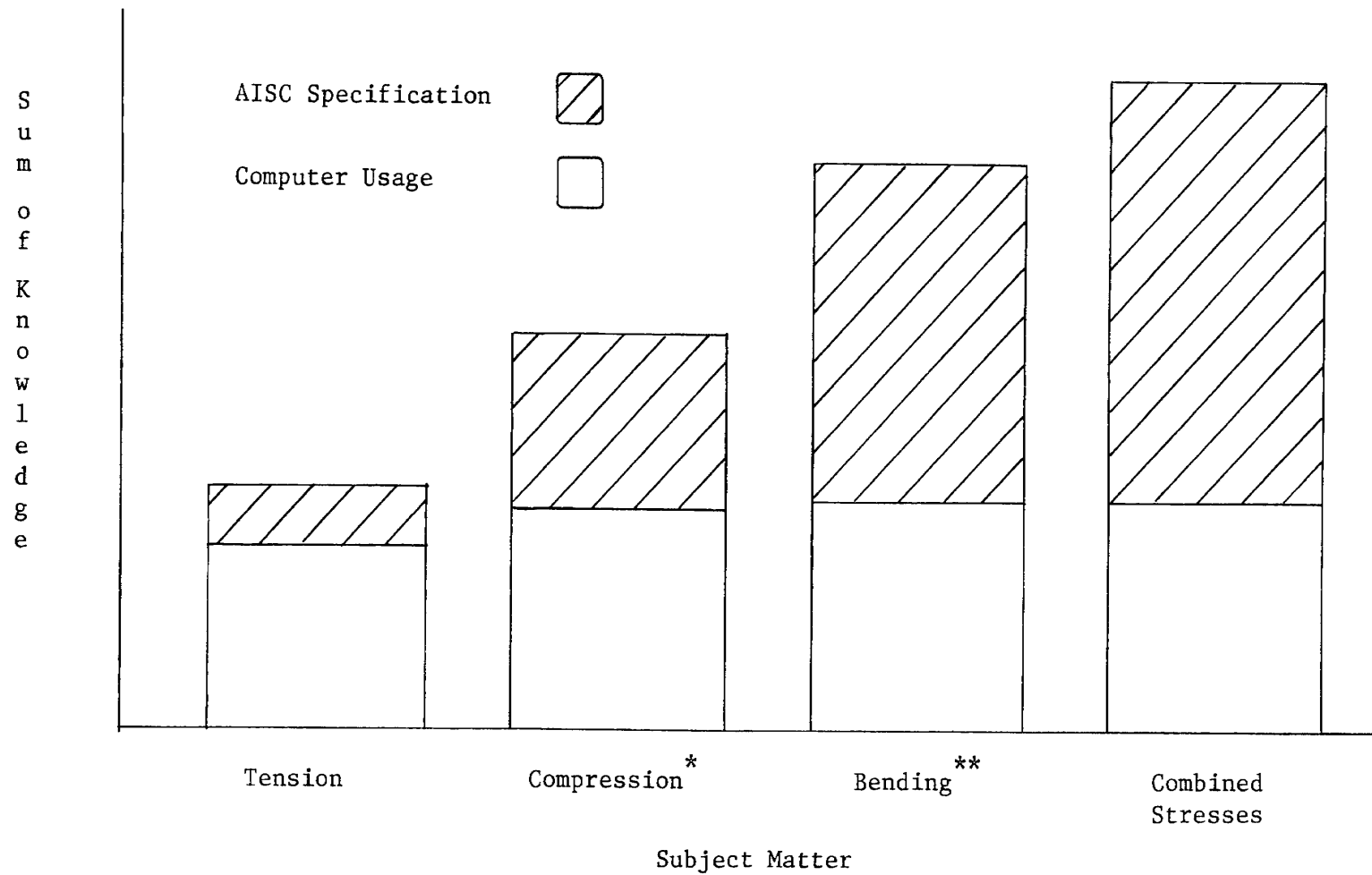
In steel design there exist many short cuts in the calculations made possible by tables and parameters readily available to the engineer. An example of one such parameter is L_c . This is the length of unbraced compression flange at which a reduction in allowable stress is required because of lateral-torsional buckling. To be effective, the programs use these type parameters. However, their use comes after the student understands exactly what they are. This requires the program to begin by performing every step of design, printing the reasons and the results. As the student progresses to more advanced problems, these short cut parameters are introduced. Only in this fashion can the programs truly act as a supplement to the lecture.

The above requirements lead to a type of program which is inefficient in its execution because of the many output titles and its interaction with the user. Once the user is familiar with the detailed calculations involved in the many equations used in design, these programs would be considered cumbersome. Because of this, it was decided to develop a second type of program. These programs efficiently

perform the design calculations, printing only enough information to help understand the answer and requiring only the minimum interaction with the user. The purpose of this second type of program is to save the student's time while working the specifically assigned homework which begins to form within him some design experience. Included in this type is a student-oriented structural analysis program which uses the direct element method to solve rigid frames, beams and trusses (9,11). Its purpose is to relieve the student time consuming hand calculations required in the special project problems.

The use of the IBM 2741 terminal will in all likelihood be new to the students enrolled in the course. To minimize the amount of time required to teach the use of the terminal, the programs begin with an explanation of their use. To further minimize the amount of explanation required before execution of the programs, they are written to cover the subject matter in a manner which starts the student with a short, easy-to-use program. After the student gains confidence in the use of the terminal, the longer and more complex programs are assigned. In this fashion, the student is gradually introduced to the use of the computer.

The basic topics in fundamental steel design are tension and compression members, beams and members subjected to combined stresses. To gain the most from the programs, it is suggested that the subjects be covered in the order stated above. Figure 6 shows the specific goals of the programs involved in teaching each topic. Equally as important as this development is the method of presenting these programs to the instructor. The purpose of the programs is to save



* Introduction to trial and error design.

** Introduction to practical design procedures, i.e., Tables, L_c , etc.

Figure 6. Goals of the Programs in Presenting Specific Topics.

time by supplementing the lecture. Therefore, the lecture outlines must clearly state the areas where the programs can be used. The final choice as to how much they are used is left up to the instructor. To assist him in his decision, an overall flow chart is included which describes, in engineering terminology, the purpose of the program. The flow chart defines the goals of the programs, their limitations, and the required information for their execution. A detailed flow chart and listing of the programs are also developed. Their purpose is to help in modifying or extending the programs. This documentation is given in Appendix IV of Reference 10.

The last item to be developed was the decision charts which are to be used by the student while working homework problems. These decision charts present the logic of the AISC Specification. This logic is the same one followed by the programs in the solution of problems. It was decided to use a flow chart format for the presentation of the decision charts. The terminology used was that of the AISC Specification with no reference to the programs. This approach makes them equally applicable in answering questions concerning the hand solutions to problems presented in class by the instructor. These are found in Appendix III of Reference 10.

SUGGESTED METHOD OF USE

The steel design course developed during this project is rather flexible. There are procedures which slightly limit this flexibility but by following them the instructor can amplify the advantages in using this method of teaching.

As shown in Figure 6, the student is gradually introduced to the use of the terminal and the concepts of steel design, if the following order of presentation of topics is used:

Design of tension members.

Design of compression members.

Design of bending members.

Design of members subjected to combined stresses.

This order of presentation insures that at any one point only one new concept is introduced by the programs. For example, when proceeding from the compression program to bending programs, the student has used the trial and error design procedure and is already familiar with the use of the terminal. In the bending programs he is introduced to the use of short cuts in design for the first time. By this gradual introduction to new material, the programs will help the instructor to be more effective in teaching because he is concerned with only one new concept at any one time.

The time to teach topics such as connections, base plates and built-up members is flexible. The only requirement is that the student have enough knowledge to understand the purpose of the material in these topics. This requirement is no different than that imposed

in a normal TAI course. However, the most effective time to present these type of topics is while the student is involved with designing the special projects. At this time he will begin to see the importance of the details in design required to have the structure act as an unit. If this procedure is followed, it is suggested that one half of the semester be used to teach the fundamentals of member design. The balance of the semester can then be spent teaching topics such as connections, base plates and built-up members.

A weekly problem session is an excellent means to be used in fulfilling the objective of professional development. During these sessions, exchange between student and teacher in the form of question and answers and/or discussions should be encouraged.

To accomplish the objectives of the course three types of homework will be assigned. The first, requiring a longhand solution, is necessary for the student to become familiar with the details of the theoretical approach to design. The second, using the computer for the solution, will accomplish the savings in class time by supplementing the lecture. The third, using the computer and/or hand solution, will be problem assignments to give the student design experience. The student should be told the specific purpose of each problem assigned. Only when he is aware of the purpose of a problem can the student be expected to work it in a manner which obtains the desired learning results (4).

The chief difficulty with a Teacher Administered Course is that the main learning resource -- the lecture -- is out of phase with the problems that occur during the students' attempt to apply the material

in working homework (3). The use of the problem sessions helps to minimize this disadvantage. It is further reduced by the decision charts covering the various sections of the AISC Specifications. These should answer many of the questions which arise while working homework assignments.

The number of terminals required to teach steel design using this TAI/CS approach depends on how much the instructor decides to use the computer. A general rule to follow in estimating the number necessary is to have enough for convenient use while working homework required to supplement the lecture. Only when the pressure, of trying to "get on" a terminal and "get off" in time to allow others to use it, is relieved can a student be expected to use the programs to learn. If this pressure exists, the tendency is to complete the homework as fast as possible with little thought as to the what and why of design.

SUMMARY AND CONCLUSIONS

One of the products of this research was a formulation of a method which could be used to create a university level course. It defined four basic steps and outlined the specifics which should be considered. The formulation was purposely made general to facilitate application to a variety of courses. Although none of the steps taken individually is particularly informative, the group taken in its entirety and stated in detail is of significant importance. It puts in chronological order a thorough process which can be used in creating a well constructed course. If such a process was applied to all courses taught, a better university educational system would result. Although the initial cost appears high, it is believed that it would be justified by others using the product of this developmental work and by the improved results. This development procedure was applied to a fundamental steel design course at the University of Missouri-Rolla. Its personal cost was approximately two thirds of a man year. This is not to say that all courses would take this amount of time. It is given only as a bench mark from which to work.

In the application of the method of development to structural steel design it was opted to use a Teacher Administered Course with Computer support (TAI/CS) as opposed to a Computer-Assisted Instruction (CAI) approach. This decision was based on the need to obtain advantages from both the TAI and CAI approach to teaching. From TAI, the advantages of the lecture presentation by a professional designer along with his personal contact with the student was used. From CAI

was taken the advantage of saving class time through the use of the computer. Along with these advantages come certain disadvantages. The problem of the lecture being out of phase with the student's problems was brought about by using TAI. The disadvantage of additional cost was caused by using the computer. An attempt was made to capitalize on the advantages and minimize the disadvantages of each.

When creating a course in which different methods of teaching are being considered, it is important to study thoroughly the advantages and disadvantages of each. In this study it would have been easy to eliminate the traditional method of teaching, i.e., TAI, based on its disadvantages without considering its benefits. A deeper study revealed that the lecture presentation was the most natural means of meeting some of the course objectives. When the fulfillment of the course objectives require the use of more than one teaching method, serious consideration should be given to the use of a combination of methods. This is what resulted from this project, i.e., the use of the computer to teach (CAI) was combined with the traditional lecture presentation and produced Teacher Administered Instruction with Computer Support (TAI/CS).

ACKNOWLEDGEMENTS

This paper is based in part on a thesis by Wendelin H. Mueller, III presented to the Civil Engineering Department of the University of Missouri-Rolla at Rolla, Missouri, in partial fulfillment of the requirements for the degree of Doctor of Philosophy. Thanks are extended to my thesis advisor, Oktay Ural, Associate Professor of Civil Engineering, Joseph H. Senne, Chairman of the Department of Civil Engineering, and to William Andrews, Professor of the Department of Civil Engineering, for their valuable suggestions and continued encouragement during this work. Acknowledgements are made to David Dearth and Michael Martin of the Department of Computer Science, University of Missouri-Rolla, for their help with the computer programming. Appreciation is also extended to the American Institute of Steel Construction, Inc. (AISC) for a Fellowship during a portion of my studies.

APPENDIX I.-REFERENCES

1. American Institute of Steel Construction, Inc., Manual of Steel Construction, 7th Ed., 1970.
2. Gerstle, Kurt H., Basic Structural Design, McGraw-Hill Book Co., 1967.
3. Hayman, R.W., and W. Lord, "A Technology-Based Educational System," Computer-Managed Learning System Laboratory, College of Engineering, Colorado State University, Fort Collins, CO, 1972.
4. Hight, Gilbert, The Art of Teaching, Alfred A. Knopf, Inc. and Random House, Inc., 1950.
5. IBM, Terminal User's Manual, "Type III Class A Program," IBM Corporation, Technical Publications Department, 1st Ed., 1970.
6. Inman, LCDR Richard P., USN, "Computer-Assisted Education at the Naval Academy," Educom, Bulletin of the Interuniversity Communications Council (EDUCOM), Vol. 4, No. 2, March 1969, pp. 3-7.
7. Kelly, William A., Ph.D., Educational Psychology, The Bruce Publishing Co., Milwaukee, 4th Ed., Revised, 1956.
8. Lekan, Helen A., Ed., Index to Computer Assisted Instruction, Instructional Media Lab., Univ. of Wisconsin-Milw., 3rd Ed., Harcourt Brace Jovanovich, Inc., Division of Instructional Systems.
9. Livesley, R.K., M.A., Ph.D., Matrix Methods of Structural Analysis, The Macmillan Co., New York, 1964.
10. Mueller, W.H. III, "Computer Assisted Teaching of Steel Design," thesis presented to the University of Missouri-Rolla, at Rolla, MO, in 1972, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.
11. Przemieniecki, J.S., Theory of Matrix Structural Analysis, McGraw-Hill Book Co., Inc., 1968.
12. Rice, John K., and John R. Rice, Introduction to Computer Science, Holt, Rinehart and Winston, Inc., 1969.
13. United States Civil Service Commission Bureau of Training, Computer Assisted Instruction: A General Discussion and Case Study, Training Systems and Technology Series: No. V, Pamphlet T-15, Aug. 1971.

VITA

Wendelin H. Mueller, III was born on February 12, 1941, in St. Louis, Missouri. He received his secondary education from Christian Brother's College, Clayton, Missouri. He received a Bachelor of Science degree in Civil Engineering from St. Louis University, St. Louis, Missouri in 1962. From June 1962 to June 1963 he was a trainee for the U.S. Army Corps of Engineers, St. Louis District. Then from June 1963 to August 1965 he had a structural engineer position with the U.S. Army Corps of Engineers, St. Louis District. While teaching as a graduate student he received a Master of Science degree in Civil Engineering from the University of Missouri-Rolla, Rolla, Missouri in August 1966. From September 1966 to May 1968 he was a research engineer for Boeing Airplane Company, Seattle, Washington. Beginning in May 1968 to September 1969 he was a structural engineer and programmer at National Computer Service, Company, Incorporated, St. Ann, Missouri.

While teaching as a half time instructor, he has been enrolled in the Graduate School of the University of Missouri-Rolla since September 1969 and held the (AISC) American Institute of Steel Construction Fellowship from September 1971 to June 1972.

APPENDIX II.-LECTURE OUTLINES

Presented in this appendix is a series of lecture outlines in modular form whose purpose is to assist the instructor in using effectively the total structural steel design course developed in this project. It is suggested that these topical outlines be used as a basis for developing a set of custom lecture notes by each instructor of the course.

The names of the applicable programs are given in the beginning of each module. A reference to the program's name within the outline indicates that that particular topic is reviewed via a discussion printed by the program. Also referenced within the outline are the applicable decision charts which are presented in Appendix III. Following the lecture outlines are examples of suggested design projects.

Introduction to the Course

Purpose: To communicate to the student the format of the course.

Applicable Programs: none

I. Computer use

A. Purpose

1. Homework
2. Lecture

B. Terminal

1. Time and place available
2. Priority for use

II. Class time

A. 50 minute lecture period

B. 3 hour laboratory period

III. Grades

A. Tests

B. Homework

C. Additional credit

Terminal Use

Purpose: To convey to the student enough information to use the computer programs.

Applicable Programs: none

- I. Activating the terminal
 - A. ON-OFF switch
 - B. LCL-COM switch
 - C. Acoustic coupler
- II. Connecting with the computer
 - A. Phone number
 - B. Answer indication
 - C. Phone connection to the acoustic coupler
- III. Login procedure
 - A. Login statement
 1. Account number
 2. Subaccount number
 - B. Name statement
 - C. Check digit
- IV. Load statement
 - A. Program name
 - B. Key
- V. XEQ statement
- VI. Underscore response
- VII. Potential problems
 - A. Computer goes down
 - B. Fatal error

Introduction on Structural Design in Metals

Purpose: To give to the student an overview of the course.

Applicable Programs: none

- I. Design
 - A. Structural designer
 - B. Specifications
 - C. Properties of a good design
- II. Structural steel design
 - A. Examples of application
 - B. Advantages
 - C. Disadvantages
- III. Introduction to the Manual of Steel Construction

Mechanical Properties of Structural Metals

Purpose: To review the more important mechanical properties of the commonly used structural metals.

Applicable Programs: none

- I. Hot rolled structural steel
 - A. Stress-strain curve
 - 1. Modulus of elasticity
 - 2. Plastic plateau
 - 3. Strain hardening
 - B. Rolled shapes
- II. Cold formed structural steel
 - A. Stress-strain curve
 - B. Shapes

Design of Tension Members

Purpose: To teach the student the design of tension members via the use of the provisions of the AISC Specification.

Applicable Programs: TENTST and TENDSN

- I. Types of tension members
 - A. Trusses
 - B. Frames (cross braces)
 - C. Others
- II. Tension stresses
 - A. Equation for stress ($F = P/A$) TENTST
 - B. Stress distribution
- III. Provisions of the AISC Specification Decision Chart 1 TENTST
 - A. Allowable tension stress (par. 1.5.1.1)
 - B. Length limitation (par. 1.8.4)
 - C. Reduced cross-section
 1. Size of holes (par. 1.14.5)
 2. Net section (par. 1.14.3)

Design of Compression Members

Purpose: To teach the student the design of compression members via the use of the provisions of the AISC Specification.

Applicable Programs: COLTST

- I. Types of compression members
 - A. Trusses
 - B. Frames
 - C. Others
- II. Compression
 - A. Equation for stress ($F = P/A$) COLTST
 - B. Buckling
 - 1. Local
 - 2. Total
 - 3. Effective length
 - C. Residual stresses
- III. Provisions of the AISC Specification Decision Chart 2 COLTST
 - A. Allowable compression stress (par. 1.5.1.3.1 and 1.5.1.3.2)
 - B. Length limitations (par. 1.8.4)
 - C. Width-thickness ratio (par. 1.9)

Design of Beams - Laterally Supported and Unsupported

Purpose: To teach the student the design of beams via the use of the provisions of the AISC Specification.

- I. Types of beams
- II. Bending stresses
 - A. Equation for stress ($F = Mc/I$) BENTST
 - B. Stress distribution
- III. Provisions of the AISC Specification for strong axis bending
Decision Chart 3 BENTST
 - A. Allowable bending stress
 1. Laterally supported compact members (par. 1.5.1.4.1)
 2. Laterally supported partially compact members
(par. 1.5.1.4.2)
 3. Laterally unsupported members (par. 1.5.1.4.6)

Note: The design portion of BENTST defines F'_y , F''_y , L_c and L_u and makes use of the first two in the design of members.

 - B. Width-thickness ratios (par. 1.9)
- IV. Provisions of the AISC Specification for weak axis bending
Decision Chart 4 (par. 1.5.1.4.3)

Design of Beam-columns

Purpose: To teach the student the design of combined stress members via the use of the provisions of the AISC Specification.

Applicable Programs: BIXDSN

- I. Types of combined stress members
 - A. Biaxial bending
 - B. Beam-columns
 - C. Biaxial bending and column action
- II. General interaction equations BIXDSN
 - A. Stress
 - B. Moment
 - C. Other
- III. Provisions of the AISC Specification Decision Chart 5 BIXDSN
 - A. Equation 1.6-1
 - B. Equation 1.6-2

Design of Connections (riveted, welded and bolted)

Purpose: To teach the student the design of connections and other details via the use of the AISC Specification.

Applicable Programs: none

- I. Types of connections (riveted, bolted and welded)
 - A. Bending
 - B. Tension
 - C. Compression
 - D. Advantages
 - E. Disadvantages
- II. Riveted connections
 - A. Types of rivets (par. 1.5.2)
 - B. Connections
 1. Shear
 2. Eccentric shear
 3. Shear and tension (par. 1.6.3)
- III. Bolted connections
 - A. Types of bolts (par. 1.5.2)
 - B. Connections
 1. Shear
 - a. Rigid
 - b. Simirigid
 2. Shear and tension (par. 1.6.3)
- IV. Welded connections
 - A. Types of welds
 - B. Connections

1. Shear

2. Shear and bending

V. Details of connections (par. 1.16 and 1.17)

VI. Base plates

Built-up Members

Purpose: To teach the student the design of built-up members via the use of the AISC Specification.

Applicable Programs: none

- I. Types of members
 - A. Bridges
 - 1. Trusses
 - 2. Beams
 - B. Buildings
 - C. Advantages
 - D. Disadvantages
- II. Angle members
 - A. Four angles
 - B. Lacing
- III. Cover plates
 - A. Plate size
 - B. Plate connection
- IV. General discussion of plate girder design

Plastic Analysis and Design

Purpose: To introduce the student to plastic analysis and design of steel structures.

Applicable Programs: PLADSN

- I. Overview of the philosophy behind plastic design PLADSN
- II. Plastic analysis
 - A. Plastic hinge PLADSN
 - B. Collapse mechanism PLADSN
 - C. Equilibrium method
 - D. Virtual work method
- III. Plastic design as per the AISC Specification PLADSN
 - A. Beams
 - B. Beam-columns

Introduction to Computer Aided Design with Available Programs

Purpose: To introduce the student to the use of commercially available computer programs.

Applicable Programs: none

- I. Overview of the types of programs
 - A. Member design
 - B. System design with optimization
- II. AISC programs
- III. ICES-STRU DL

Design a roof truss, simply supported by masonry walls, to span 30 ft. center to center bearing. Paragraph and page numbers refer to the AISC Manual of Steel Construction.

Given:

Distance between trusses - 30 ft.

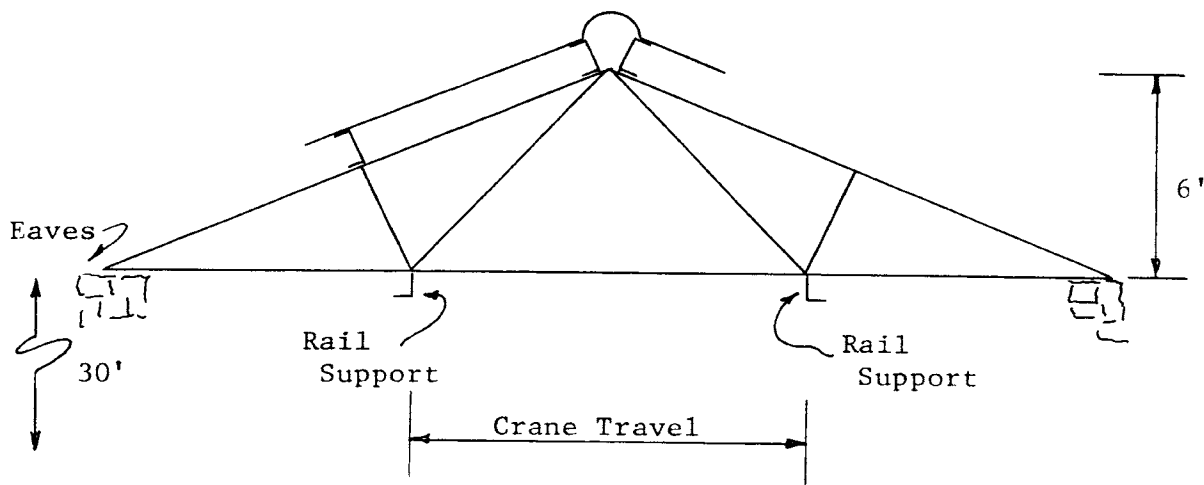
Loads:	Live	Snow - 30 psf horizontal projection (Ref. p. 5-224).
	Dead	Truss - 5 psf Roof corrugated steel (Ref. p. 6-16).
	Crane	12 ton capacity (Ref. par. 1.3.4).
	Wind	as required

Specification: AISC

Members: Truss Double angles.

Purlins Channels.

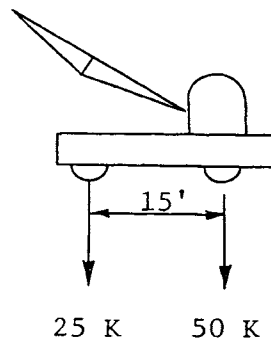
Steel: A-36



A one lane bridge is to be built over the abutments of a dam. The function of the bridge is to support a traveling crane which will be used to lift generators housed in power plants at each side of the bridge.

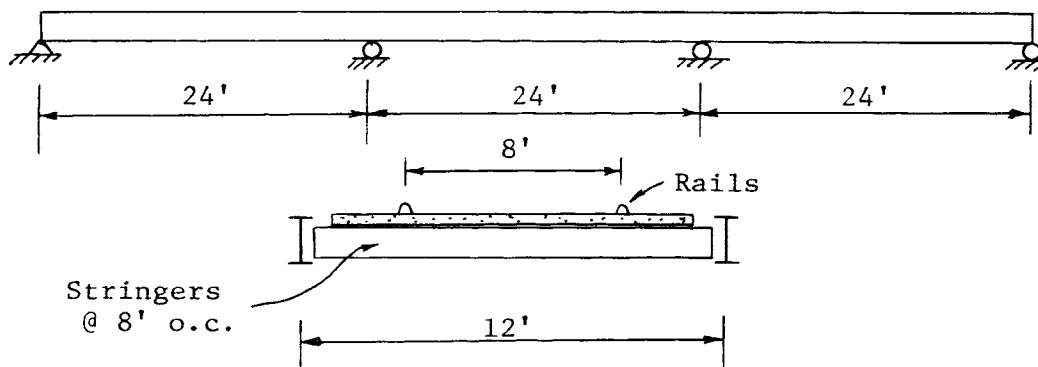
The following information is given:

1. Three span continuous bridge
2. Span length 24 ft.
3. Loads: Live - crane (see sketch)
Dead - as required (assume 8" slab)
4. W sections
5. Bolted connections
6. AISC Specification
7. A-36 steel

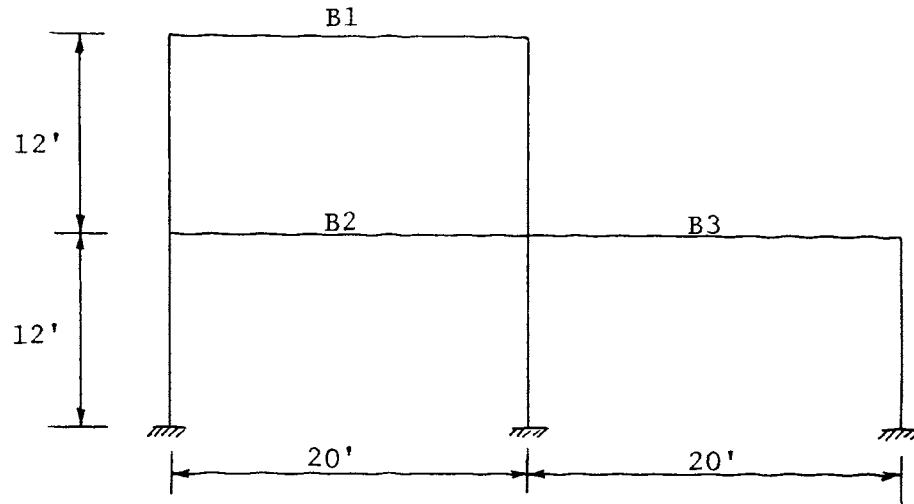


The following information is required:

1. Moment and shear envelope for the load positioned at 6' spaces.
2. Design all primary members.
3. Design the connections.



A garage for passenger cars is to be built in the following configuration.



The framing system is to be on 12 ft. centers. An external self-supporting elevator will be used to transfer the cars from the ground floor to the upper decks.

Given Loads: Live - BOCA 1970
 Dead - as required (assume 8" concrete slab)
 Use: W sections
 Bolted connections
 AISC Specification
 A-36 steel

Assume: Uniform load distribution from slab to frame.
 Top flange laterally supported for B1, B2, and B3.

Find: Design the frame and the connections.

APPENDIX III.-DECISION CHARTS

In this appendix is presented a series of decision charts. These outline the logic of the AISC Specification applied in fundamental design problems. The specific topics which they include are: tension members, compression members, beams and members subjected to combined stresses. Each of these topics are covered using the AISC Specification's terminology. The nomenclature is also from the AISC Specification and is presented in this appendix.

The object of the decision charts is to present a road map of the AISC Specification. By following the decision charts, the student is led through the various paragraphs which apply in a specific design problem. All the required decisions are presented in the proper order.

The purpose of these decision charts in teaching steel design is to give the student a concise summary of the design procedures. They are to be used while working homework either by hand or using the computer. Through their use, the student will be able to obtain immediate answers to many of the questions which arise while working homework. This purpose of the decision chart should be thoroughly explained to the class.

When teaching steel design using the technique outlined in the body of this paper, it is suggested that these decisions charts be presented to each student. This can be done either by a blackboard presentation and/or copied and handed out in class.

DECISION CHART 1 This decision chart outlines the procedure for calculating the allowable tension force in a primary member. An account is made for the reduction in area due to rivet or bolt holes

and the chain of holes provision in the AISC Specification. The eighty five percent rule of Paragraph 1.14.3 is also checked. Knowledge of the slenderness ratio $\frac{\ell}{r}$, net area and gross area is required before this decision chart can be effectively used.

DECISION CHART 2 This decision chart outlines the procedure for calculating the allowable compression stress in a primary member. Knowledge of the slenderness ratio $\frac{K\ell}{r}$ and the provision of Section 1.9 is required for effective use of this decision chart. The member may not be fully effective in resisting the load as determined by Section 1.9. Therefore the application of the provisions of Appendix C of the AISC Specifications is a possible result. Therefore, the student must either know how to apply these provisions or have been told by the instructor to stop at this point or use a new member when this situation exists. In a fundamental course such as this, the latter choice will probably be used.

DECISION CHART 3 This decision chart outlined the procedure for calculating the allowable stress in bending for W sections bent about their major axis. The axial load is assumed to be zero. The requirements for compactness and the bracing requirements are checked. Based on the results the appropriate equation to be used to calculate the allowable stress is chosen. Knowledge of the definition of bracing length and the calculations of C_b is required for effective use of this decision chart. The student must know what to do if the member is not fully effective in resisting the load as determined by Section 1.9.

DECISION CHART 4 This decision chart outlines the procedure for calculating the allowable stress in bending for W sections bent about

their minor axis. The student must know what to do if the member is not fully effective in resisting the load as determined by Section 1.9.

DECISION CHART 5 This decision chart outlines the application of the combined stress provisions of the AISC Specification as applied to W sections. It is the only decision chart which is presented on more than one page. It required six pages and should be handed out as a unit. The student must have acquired enough knowledge to calculate the allowable compression stress and allowable bending stress about the major and minor axis using the respective decision chart. He must know what to do if the member is not fully effective in resisting the load and how to calculate C_m and $F'e$.

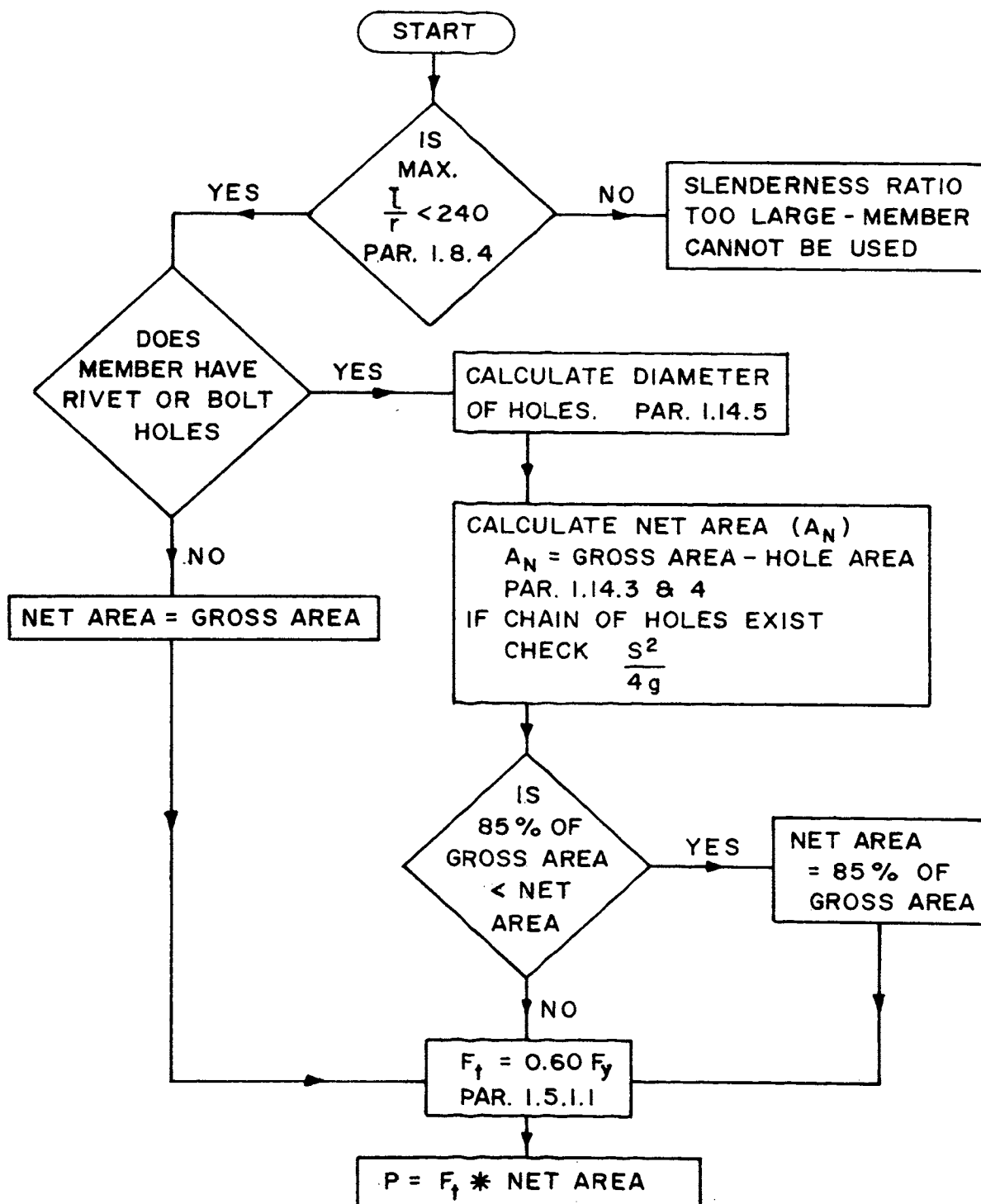
NOMENCLATURE

A_f	Area of compression flange
C_b	Bending coefficient dependent upon moment gradient
C_c	Column slenderness ratio dividing elastic and inelastic buckling
C_m	Coefficient applied to bending term in interaction formula and dependent upon column curvature caused by applied moments
E	Modulus of elasticity of steel (29,000 kips per square inch)
F_a	Axial stress permitted in the absence of bending moment
F_b	Bending stress permitted in the absence of axial force
F'_e	Euler stress divided by factor of safety
F_t	Allowable tensile stress
F_y	Specified minimum yield stress of the type of steel being used (kips per square inch). As used in this Specification, "yield stress" denotes either the specified minimum yield point (for those steels that have a yield point) or specified minimum yield strength (for those steels that do not have a yield point).
K	Effective length factor
P	Applied load (kips)
b_f	Flange width of rolled beam or plate girder
f_a	Computed axial stress
f_b	Computed bending stress
g	Transverse spacing between fastener gage lines

λ, ℓ	Actual unbraced length
r	Radius of gyration
s	Spacing (pitch) between successive holes in line of stress
t_w	Girder, beam, or column web thickness
t_f	Flange thickness

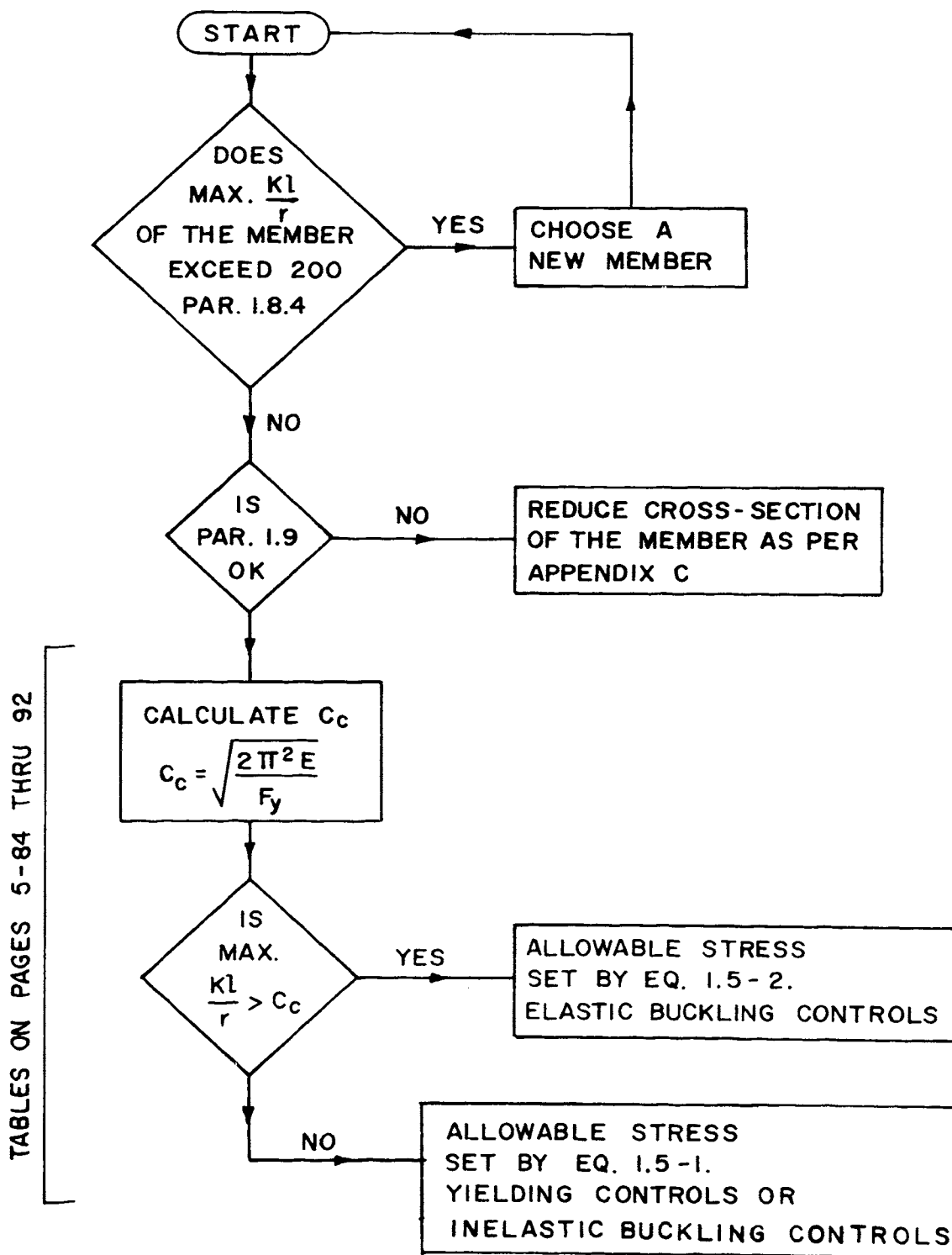
DECISION CHART I

DECISION CHART FOR CALCULATING THE ALLOWABLE TENSION FORCE IN A PRIMARY MEMBER



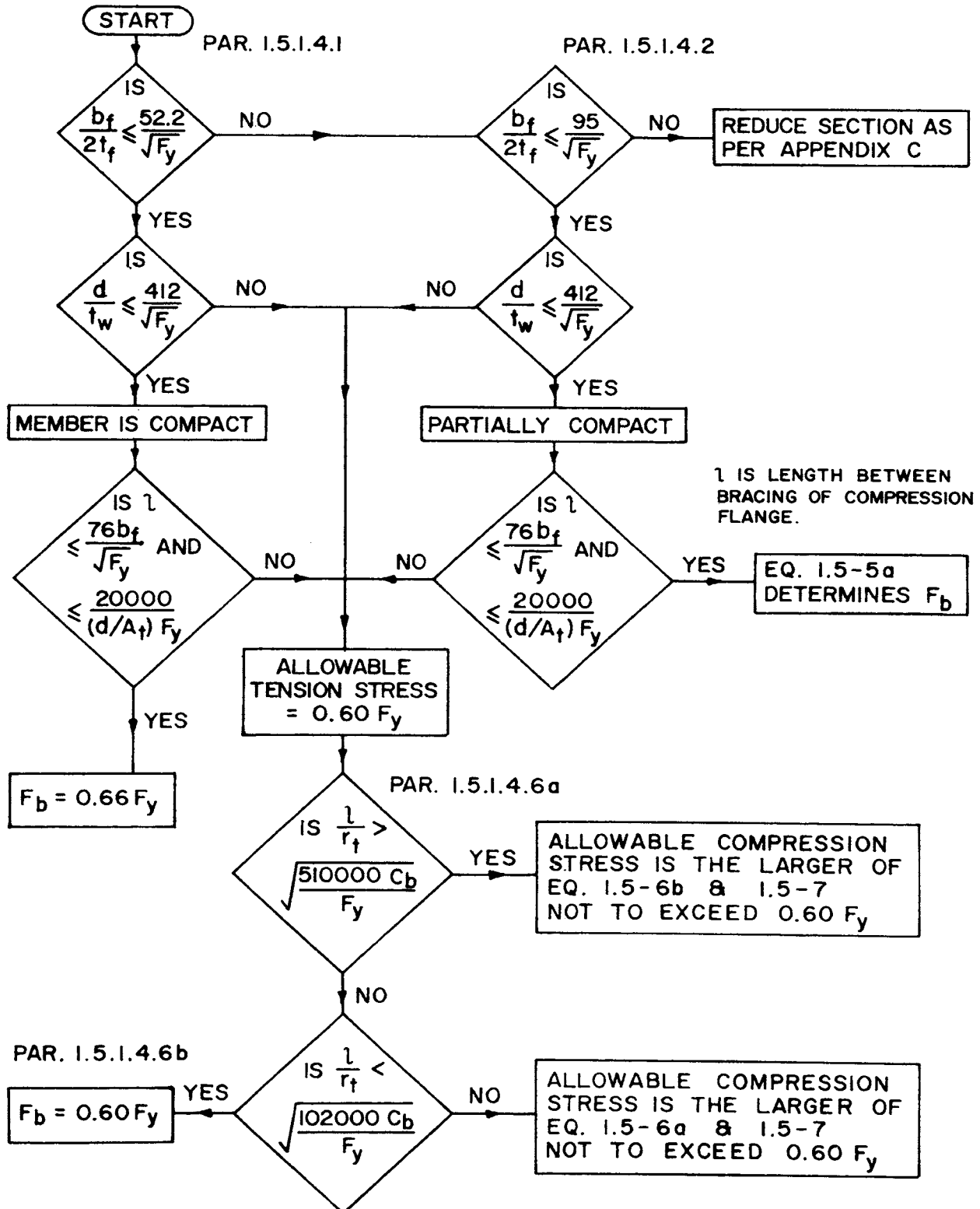
DECISION CHART 2

DECISION CHART FOR CALCULATING THE ALLOWABLE
COMPRESSION STRESS - PRIMARY MEMBERS



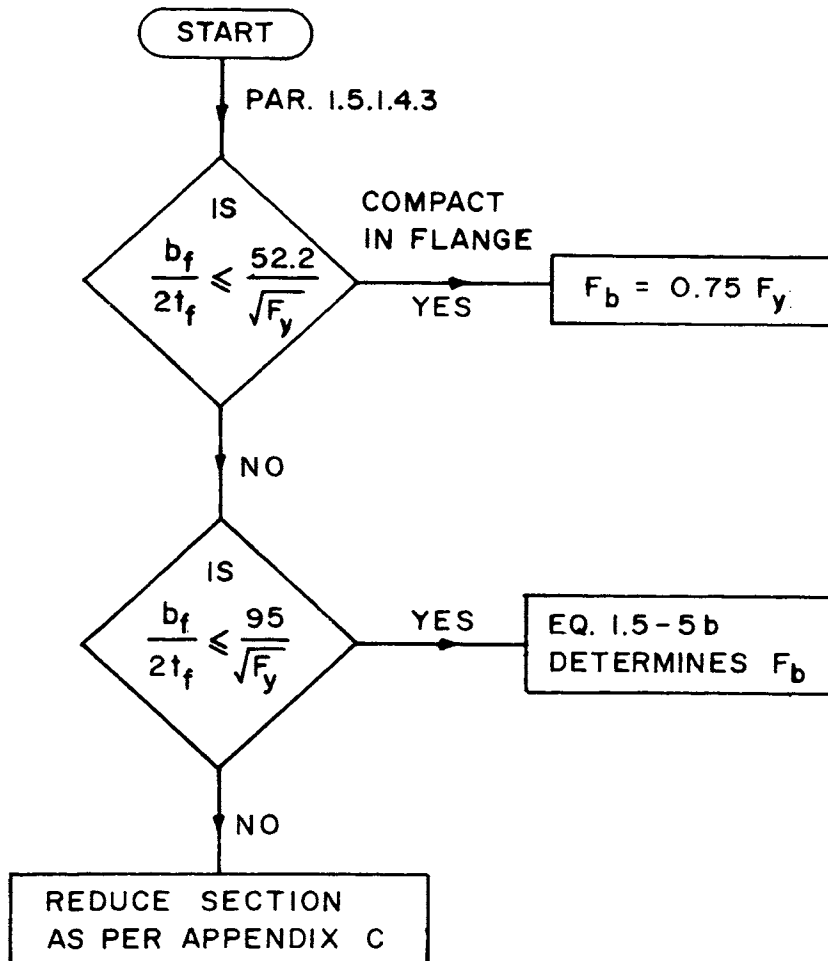
DECISION CHART 3

DECISION CHART FOR CALCULATING THE ALLOWABLE STRESS IN BENDING FOR W SECTIONS BENT ABOUT THE MAJOR AXIS - NO AXIAL LOAD



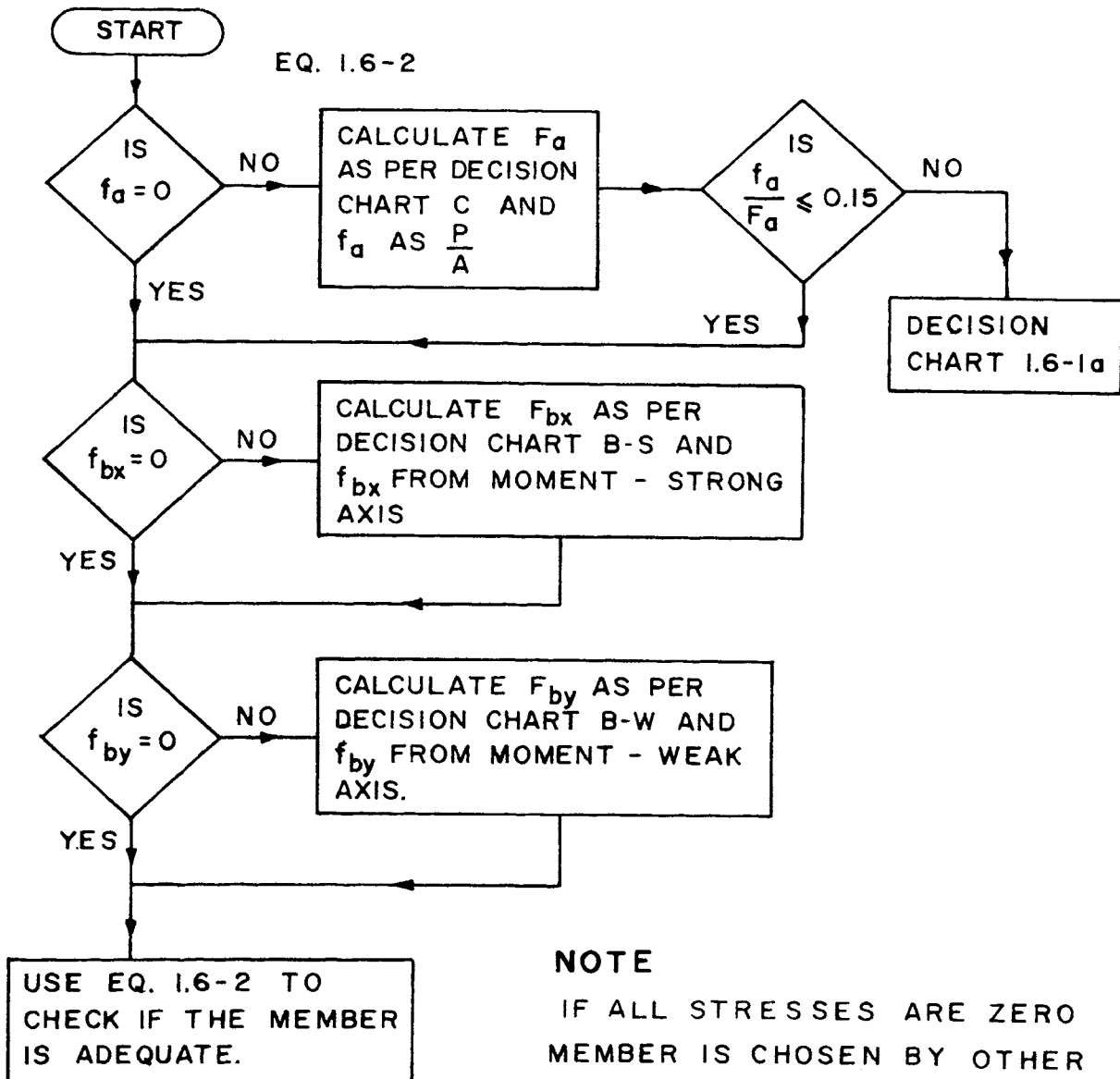
DECISION CHART 4

DECISION CHART FOR CALCULATING THE ALLOWABLE
STRESS IN BENDING FOR W SECTIONS BENT ABOUT
THE MINOR AXIS.

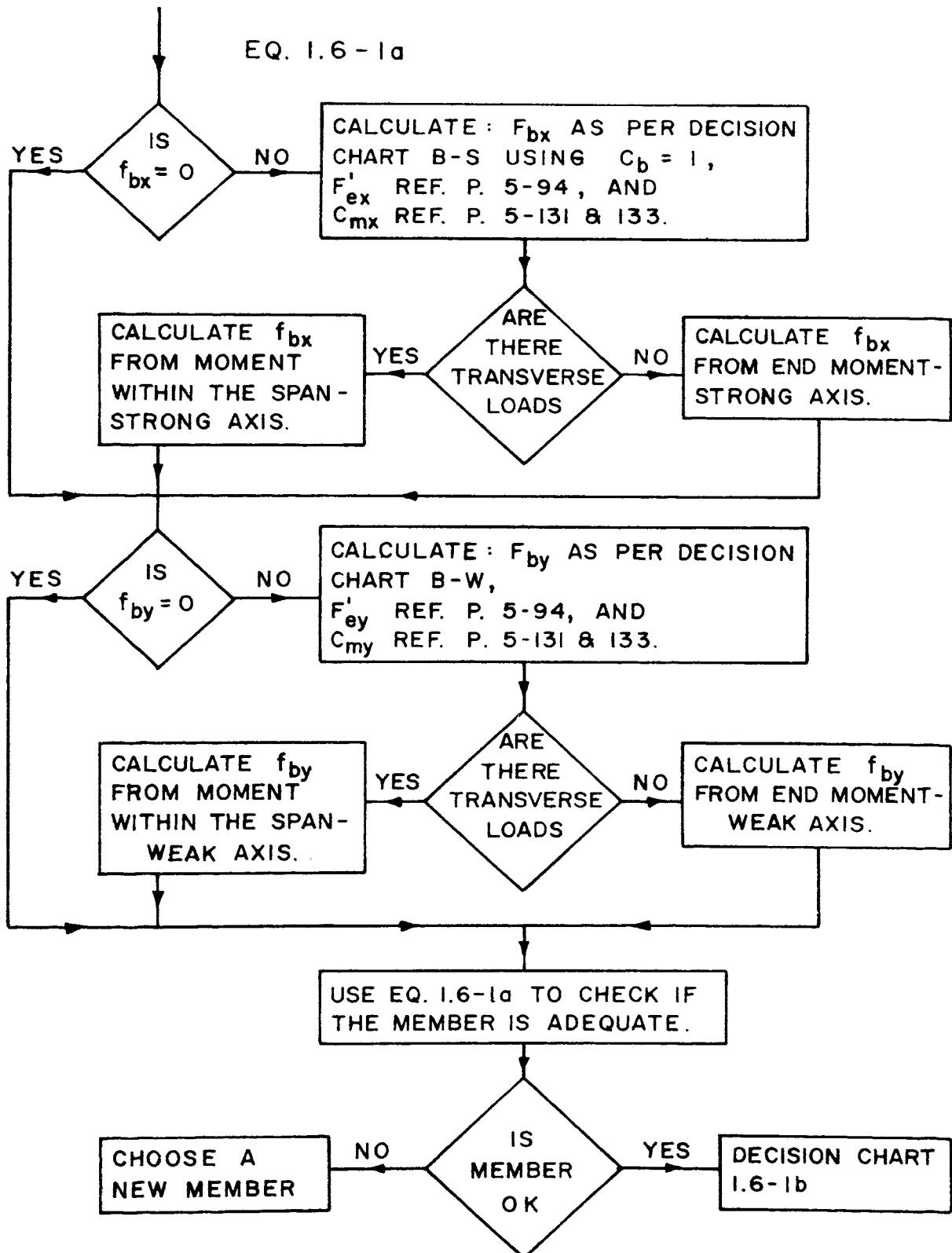


DECISION CHART 5

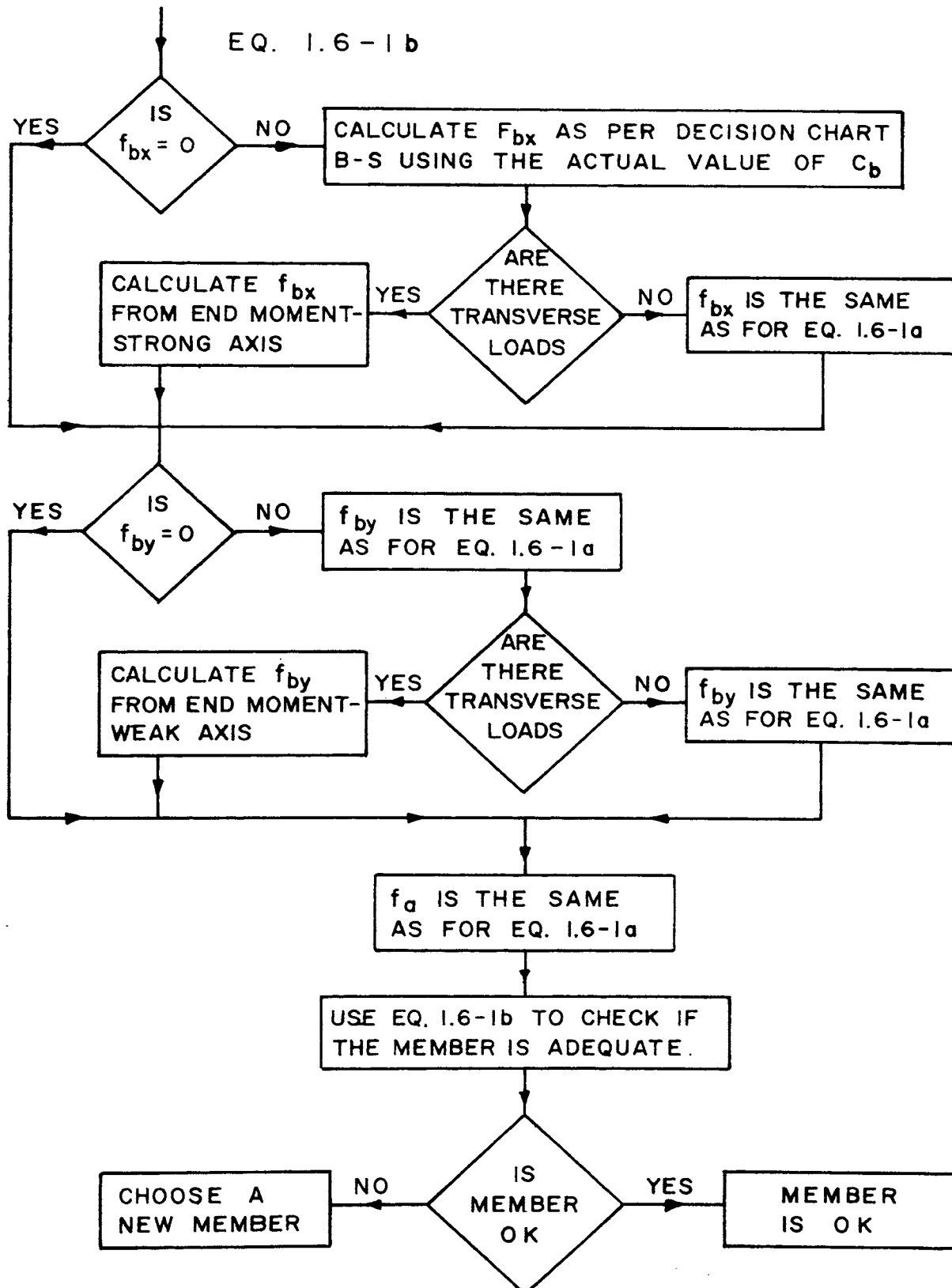
DECISION CHART FOR THE APPLICATION OF THE COMBINED
STRESS PROVISIONS OF A.I.S.C. TO W SECTIONS



DECISION CHART 1.6-1a

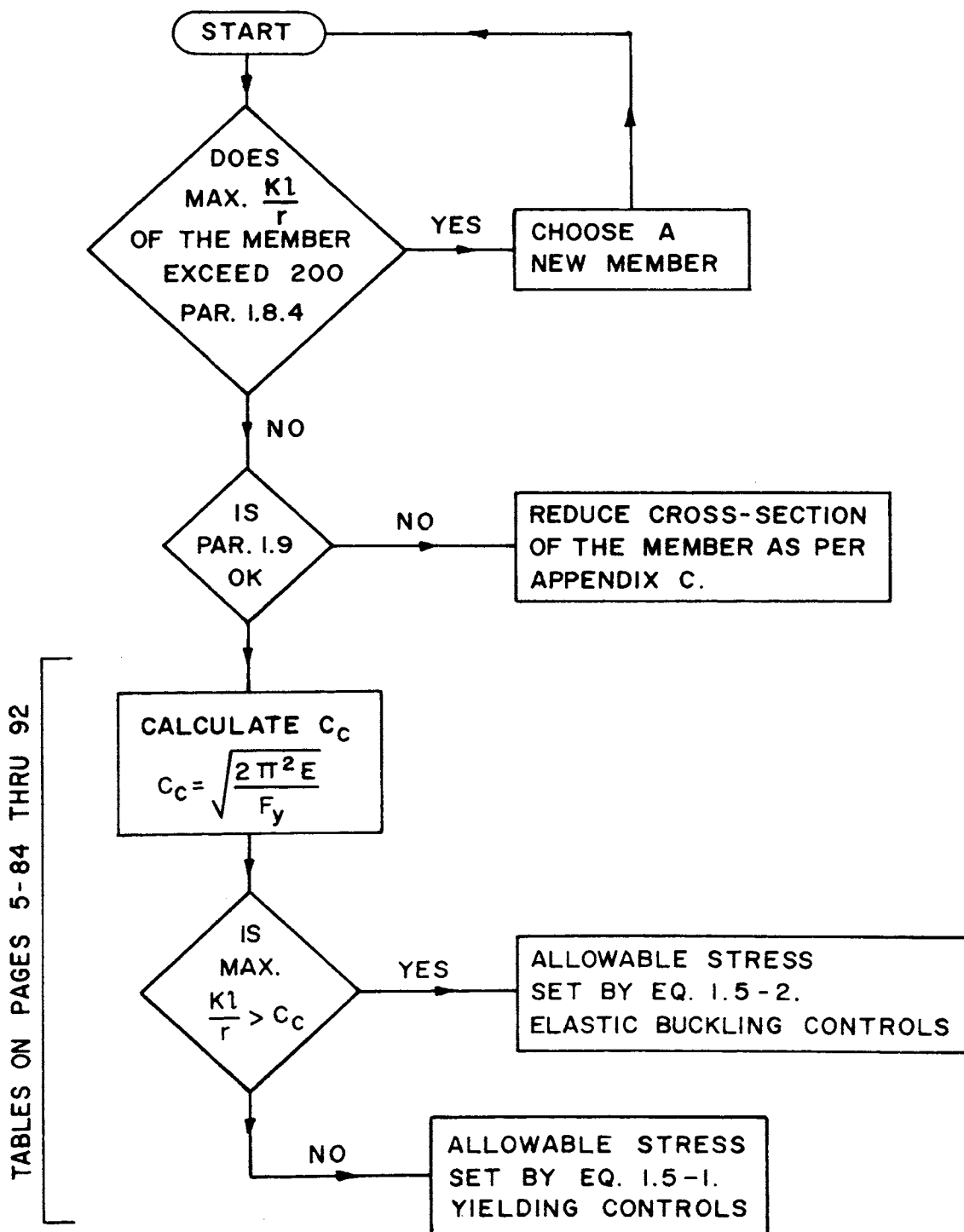


DECISION CHART 1.6-1b



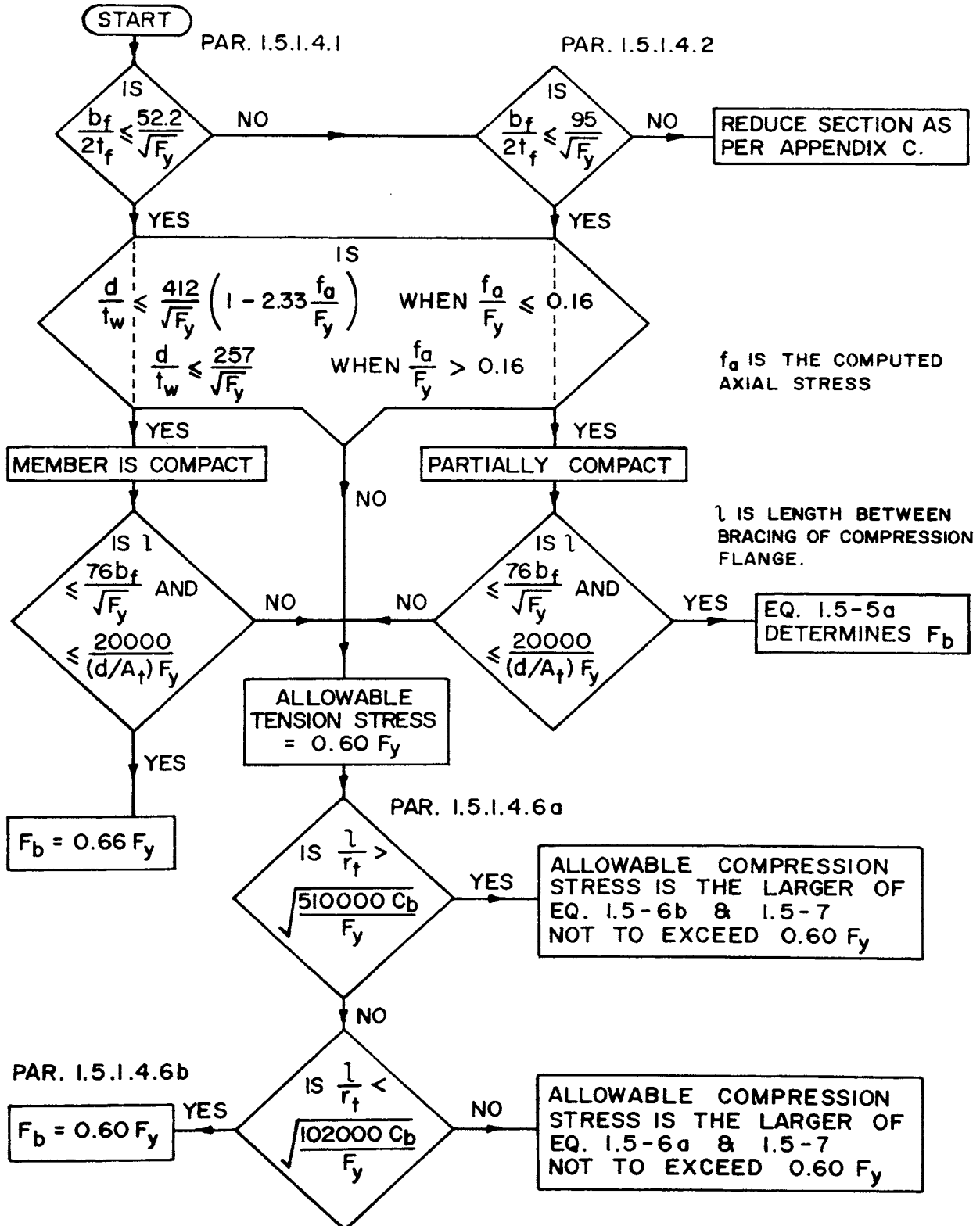
DECISION CHART C

DECISION CHART FOR CALCULATING THE ALLOWABLE COMPRESSION STRESS - PRIMARY MEMBERS



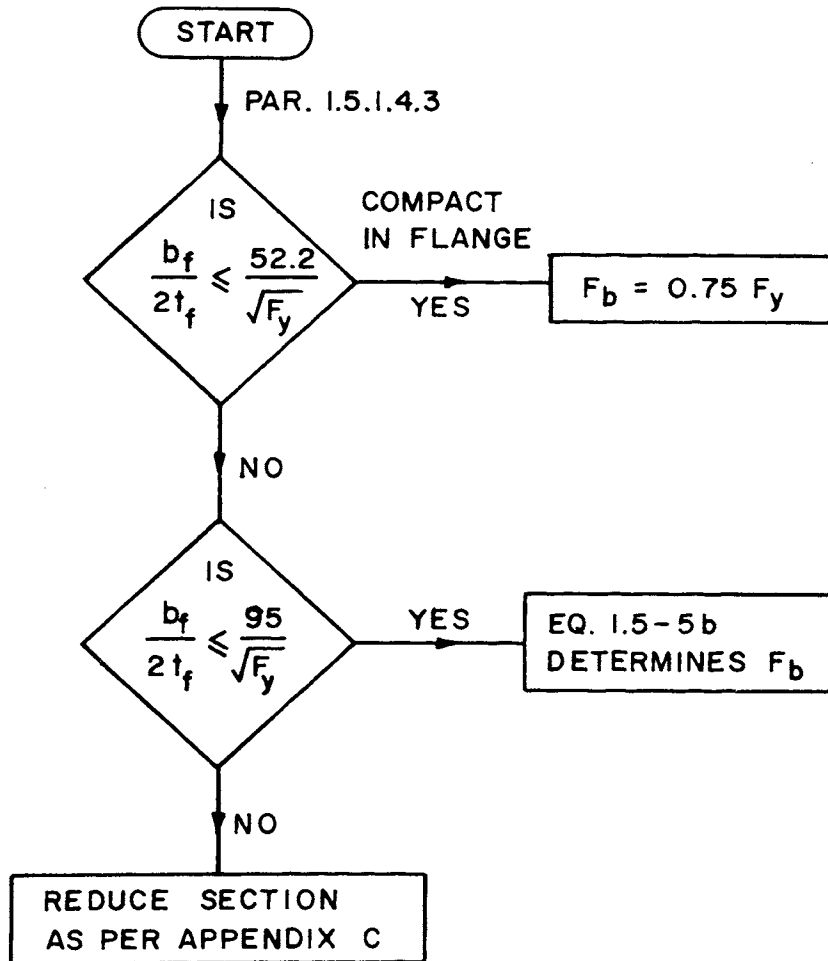
DECISION CHART B-S

DECISION CHART FOR CALCULATING THE ALLOWABLE STRESS
IN BENDING FOR W SECTIONS BENT ABOUT THE MAJOR AXIS



DECISION CHART B-W

DECISION CHART FOR CALCULATING THE ALLOWABLE STRESS IN BENDING FOR W SECTIONS BENT ABOUT THE MINOR AXIS.



APPENDIX IV.-PROGRAM DOCUMENTATION AND LISTINGS

In this appendix is presented the documentation of a series of computer programs developed to aid in teaching the fundamentals of steel design. The AISC Specification's design criteria was used. All the programs are written in CPS-PL/I for an IBM 2741 terminal. They interact with the user during the design process through the use of conversational programming. All the programs are written such that once executed, their use is explained via the output on the terminal.

Two types of programs were developed. Those whose name end in TST are learning programs with extensive output and repetitious input. DSN as the last three letters of a name indicate efficient design and investigation programs. In this type of program only the minimum input is requested and the output is reduced to only that required to indicate which design route was taken. A structural analysis program RIGID was also developed. This program will assist the student when performing the structural analysis required by the special design projects.

The documentation of the programs is presented in the following four parts:

PROGRAM DESCRIPTION - The program description consists of the program's name, its' purpose and limitations and any prerequisites and data required before execution.

OVERALL FLOW CHART - This flow chart presents an overview of what the program will do during execution. It is presented in flow chart form but uses only the AISC Specification terminology. This part of the documentation along with the program description is intended for use

by the instructor of the course. He should use them to determine which type of homework problem is to be assigned in order to accomplish a specific objective.

DETAILED FLOW CHART - This flow chart presents the logic blocks of the programs. It uses both computer programming and the AISC Specification terminology.

LISTING - The statement by statement listing of the program is presented. The detailed flow chart gives the objective of a block of program statements and the listing gives the details. These two parts of the documentation are presented for use by persons modifying or extending the programs.

INDEX TO PROGRAM
DOCUMENTATION

	Page
TENTST.....	75
TENDSN.....	82
COLTST.....	92
BENTST.....	113
BENDSN.....	142
BIXDSN.....	168
PLADSN.....	205
RIGID.....	223

Program Name:

TENTST

Purpose:

Program TENTST is a question and answer review of tension member design using the AISC Specification. It is a very simple program to use and should be the first presented to the class. In this way it will serve as an introduction to the use of the terminal.

Limitations:

This program is a review program only and will not perform design or investigation, other than two preprogrammed examples.

Prerequisites:

Read AISC Sections:

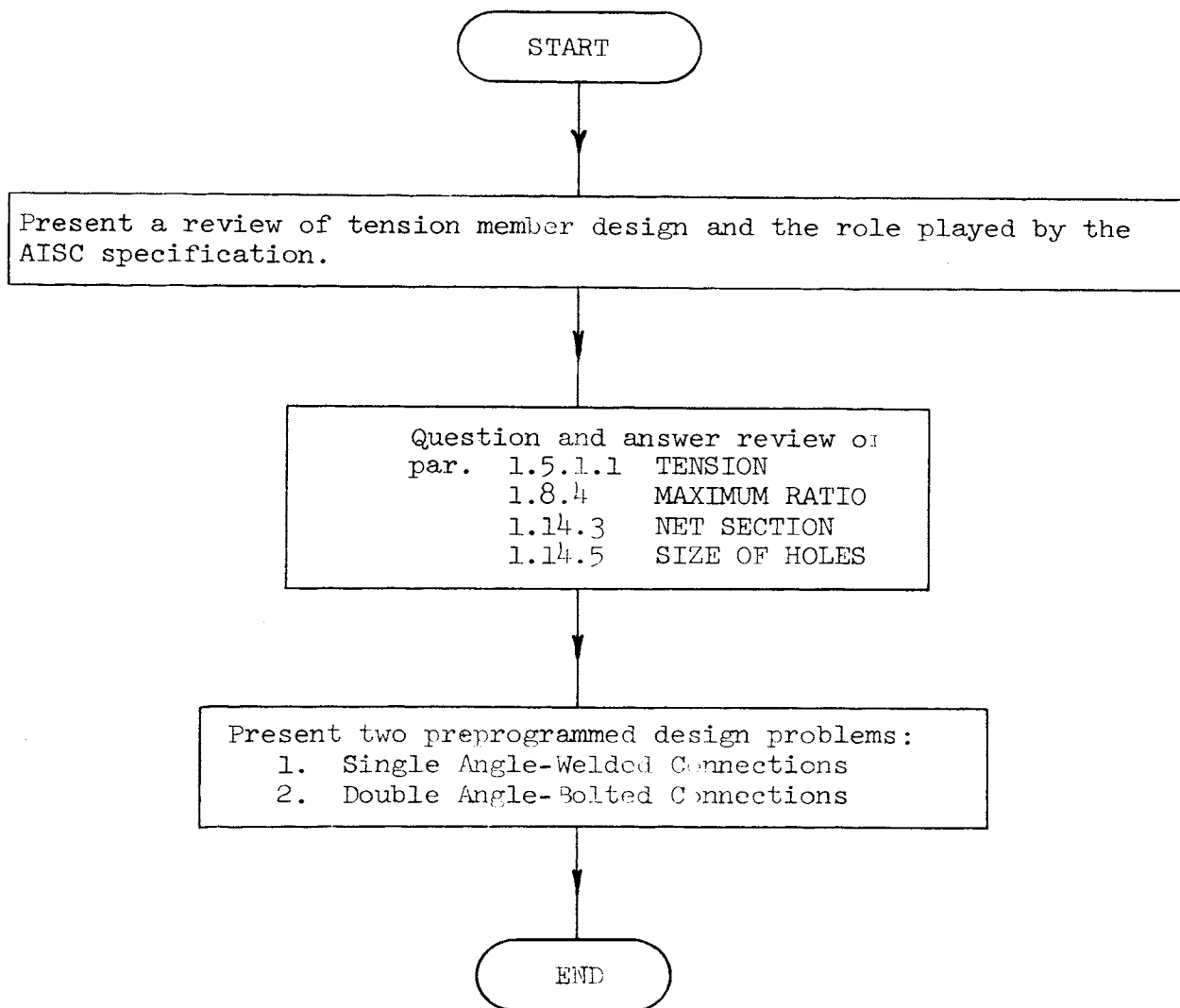
1.5.1.1	TENSION
1.8.4	MAXIMUM RATIO
1.14.3	NET SECTION
1.14.5	SIZE OF HOLE

The student must know the purpose of stitching rivets and welds.

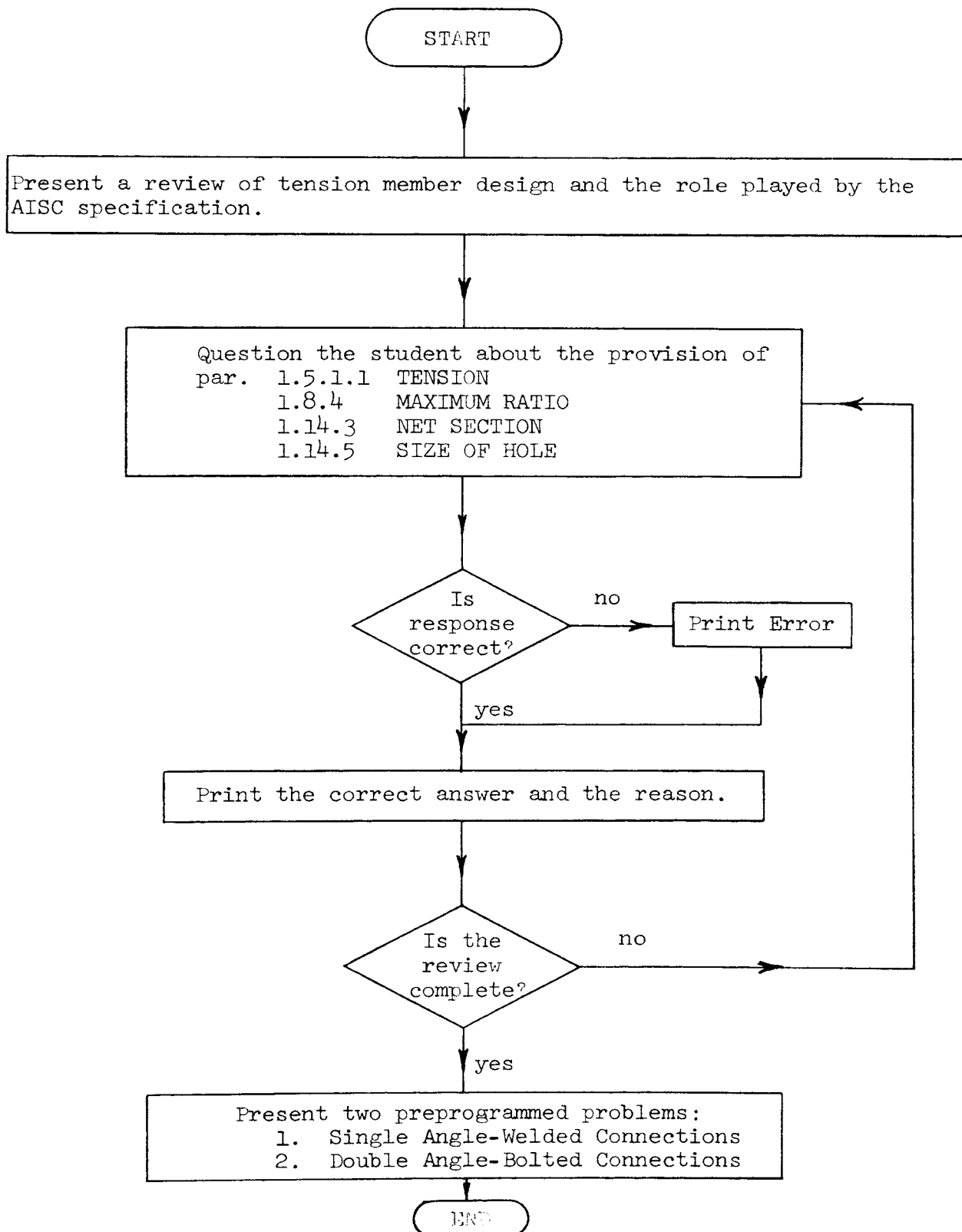
Data Required Before Execution:

None.

Overall Flow Chart TENTST



Detailed Flow Chart TENTST



```

/* PROGRAM TENTST */
PUT LIST('program TENTST');
PUT LIST(' ');
PUT LIST('PURPOSE: program TENTST will review with you the principles of tension
  member design. ');
PUT LIST(' ');
PUT LIST('Before continuing you should have read the following sections in the
  AISC code: ');
PUT LIST('      1.5.1.1 TENSION');
PUT LIST('      1.8.4   MAXIMUM RATIO');
PUT LIST('      1.14.3  NET SECTION');
PUT LIST('      1.14.5  SIZE OF HOLES. ');
PUT LIST(' ');
PUT LIST('In general the design of a tension member is based on the stress being
  equal to the force divided by the area');
PUT LIST(' or  $f = P/A$ . In design you provide the required area such that the
  stress is below some allowable value. ');
PUT LIST(' The AISC Specifications provides the allowable stress and some
  practical guidelines on allowable length');
PUT LIST(' and reduction of area due to connections. ');
PUT LIST(' ');
PUT LIST('Lets review some of these requirements. ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' Because buckling is no problem the length of a tension member is NOT
  limited by the AISC code');
PUT LIST(' true=1 false=0');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO lab2;
PUT LIST(' ERROR');
lab2: PUT LIST('The code recommends limiting the length of a tension member by
  restricting the L/r to be less than 240. ');
PUT LIST(' ');

```

```

PUT LIST('If 7/8 inch diameter bolts are to be used in a tension connection the
  reduced area of the member');
PUT LIST('  is calculated based on a ____ inch diameter hole? ');
GET LIST(ANSWER);
IF ANSWER=.9375 THEN GO TO 1ab3;
PUT LIST('ERROR');
1ab3: PUT LIST('The code requires an increase of 1/16 inch to the diameter of the bolt
  or rivet in calculating');
PUT LIST('  the area of the hole. ');
PUT LIST(' ');
PUT LIST('Given a 6 X 1/2 inch plate with two staggered 1 in. diameter holes,
  what is the net section in ');
PUT LIST('  resisting a tension force. ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' _____ ');
PUT LIST('                2" ');
PUT LIST('      o - - - - - ');
PUT LIST('                2" = g ');
PUT LIST('      o- - - - - ');
PUT LIST('                2" ');
PUT LIST(' _____ ');
PUT LIST('      <--> ');
PUT LIST('      2"=s ');
GET LIST(ANSWER);
IF ANSWER=2.25 THEN GO TO 1a1;
PUT LIST('ERROR');
1a1: PUT LIST('The effective area is the minimum of the following: ');
PUT LIST('      Area = 1/2*(6 - 1) = 2.5sq.in. ');
PUT LIST('      Area = 1/2*(6 - 2 + s*s/(4*g)) = 1/2*(6 - 2 + 4/(4*2)) = 2.25sq.in.
  this controls ');
PUT LIST('      Area = .85*6*.5 = 2.55sq.in. ');
PUT LIST(' ');
PUT LIST('What is the allowable tension stress if the Yield stress is 36 ksi? ');
GET LIST(ANSWER);

```

```

IF ANSWER=22 THEN GO TO lab4; ELSE GO TO lab5;
lab4: PUT LIST('ERROR');
lab5: PUT LIST(' The value of allowable stress for a Yield of 36 ksi is 22 ksi. It is
        calculated by  $.6*Yield$ ');
        PUT LIST(' or is found on Page 5-64 of the specifications. ');
        PUT LIST(' ');
        PUT LIST('The following design problems are presented to provide the user with a
        procedure and acceptable format. ');
        PUT LIST(' for the calculations in tension member design. ');
        PUT LIST(' ');
        PUT LIST('Design a single equal leg angle 10 ft. long to support a tension force of
        54 kips. ');
        PUT LIST(' Use a Yield stress of 36 ksi and assume welded connections. ');
        PUT LIST(' Calculations: Allowable stress  $.6*Yield = .6*36 = 22ksi$  ');
        PUT LIST(' Required area of steel Force/Allowable stress =
         $54/22 = 2.45 \text{ in.}^2$  ');
        PUT LIST(' Minimum  $r = L/240 = 12*10/240 = 0.5 \text{ in.}$  ');
        PUT LIST(' ');
        PUT LIST(' Use: Single angle  $3 \frac{1}{2} \times 3 \frac{1}{2} \times \frac{3}{8}$  ref. page 1.57 ');
        PUT LIST(' Area =  $2.48 \text{ sq.in.} > 2.45 \text{ sq.in.}$  ');
        PUT LIST('  $r_{z-z} = .687 \text{ in.} > .5 \text{ in. OK}$  ');
        PUT LIST(' ');
        PUT LIST(' Check stress: stress =  $P/area = 54/2.48 = 21.8 \text{ ksi}$  ');
        PUT LIST(' Percent understressed =  $(22 - 21.8)/22 = 0.9 \%$  ');
        PUT LIST(' ');
        PUT LIST('Design an equal leg double angle of length 12 ft. to support a tension
        load of 100 kips. ');
        PUT LIST(' Assume a  $7/8 \text{ in.}$  bolt will be used as a connector and the angles will
        be connected ');
        PUT LIST(' together or stitched on 3 ft. centers. Use A-36 steel. ');
        PUT LIST(' Calculations: Allowable stress =  $.6*Yield = .6*36 = 22ksi$  ');
        PUT LIST(' Required area of steel = Force/Allowable stress =
         $100/22 = 4.55 \text{ sq.in.}$  ');
        PUT LIST(' Minimum  $r = L/240$  ');

```

```

PUT LIST('          single angle L/240 = 3*12/240 = .166 in. ');
PUT LIST('          double angles L/240 = 12*12/240 = .6 in. ');
PUT LIST(' ');
PUT LIST('          Assume angles chosen will be approximately 1/2 in.
thick. ');
PUT LIST('          Reduction in area due to bolt holes = 7/8 + 1/16)
(1/2)*2 = .94 in. ');
PUT LIST('          (Two holes were used because bolt will go thru
the back to back legs of the angles. ');
PUT LIST(' ');
PUT LIST('          Total area required = 4.55 + .94 = 5.49sq.in. ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' Use: 4 X 4 X 3/8 Double angles ref. page 1.64 ');
PUT LIST('          Area = 5.72 - (7/8 + 1/16)*3/8*2 = 5.02sq.in. ');
PUT LIST('          or ');
PUT LIST('          Area = 85% of total area = .85*5.72 = 4.86sq.in.
< 5.02sq.in. this controls ');
PUT LIST('          Minimum r single angle = .788in. > .166in. ');
PUT LIST('          Minimum r double angle = 1.24in. > .6in. ');
PUT LIST(' ');
PUT LIST('          Check stress: f = P/A = 100/5.02 = 19.9 ksi ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST('It is suggested that you keep this output in your notebook as part of
the class notes. ');
PUT LIST(' ');
PUT LIST('If you think you have the concepts of tension design and wish to
interact with the computer ');
PUT LIST(' on some more design problems LOAD and EXECUTE TENDSN. ');

```


Program Name:

TENDSN

Purpose:

Program TENDSN will interact with the user in the design of tension members using the AISC Specification. It will design both single and built up members with bolted, riveted or welded connections. If a reduction in area is required because of rivet or bolt holes, a table of required gross area is printed. The smallest section is then chosen based on this table using the number of connection holes at a section and the thickness of the steel. If the member being designed is a built up section, a table of minimum r values for varying stitching lengths is printed. The use of these tables makes tension design a one pass calculation. The introduction to trial and error design is presented in program COLTST.

Limitations:

If staggered holes exist, interpretation in the table of required gross area or a manual check is required.

Prerequisites:

Execution of program TENTST.

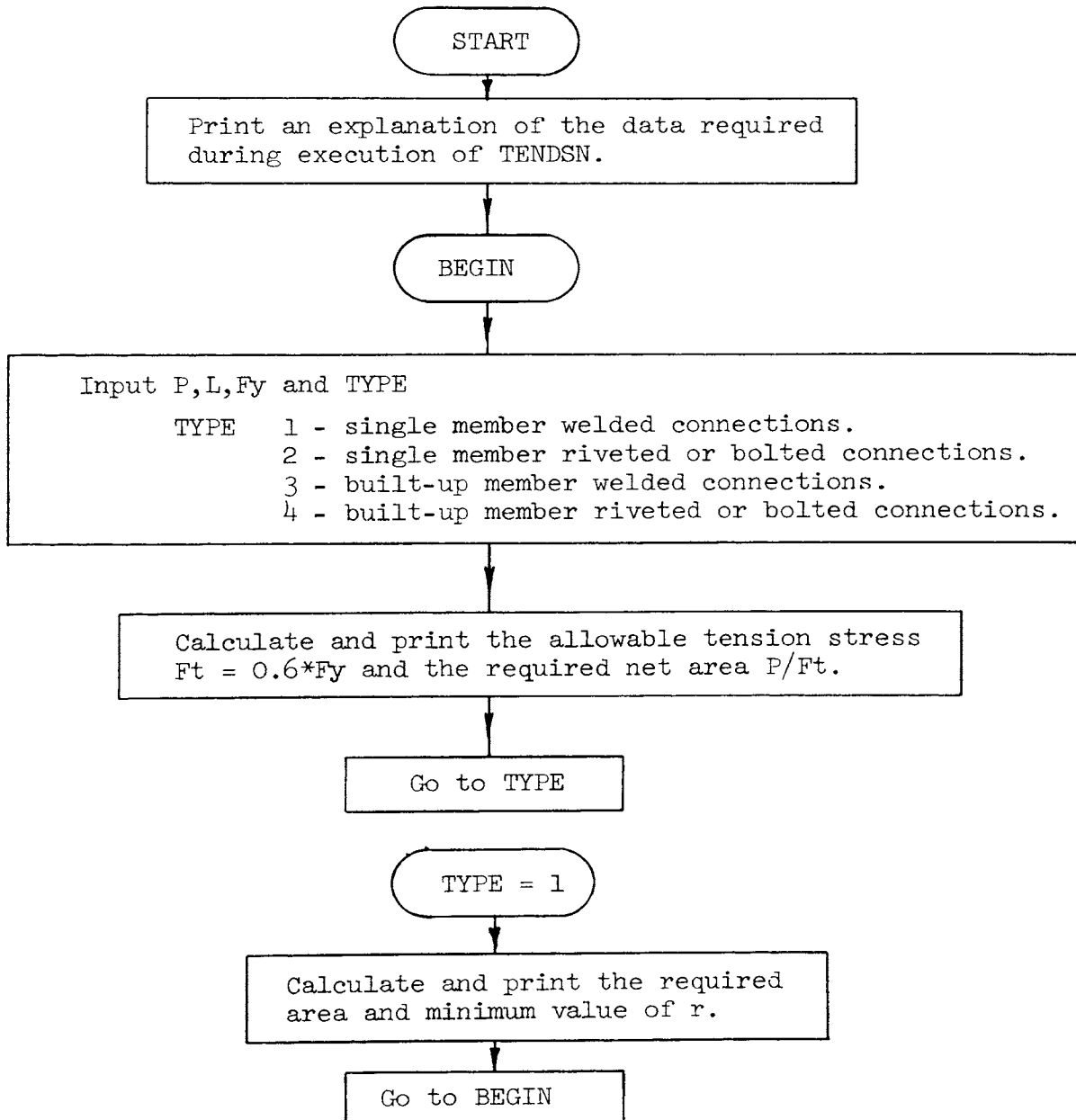
Data Required Before Execution:

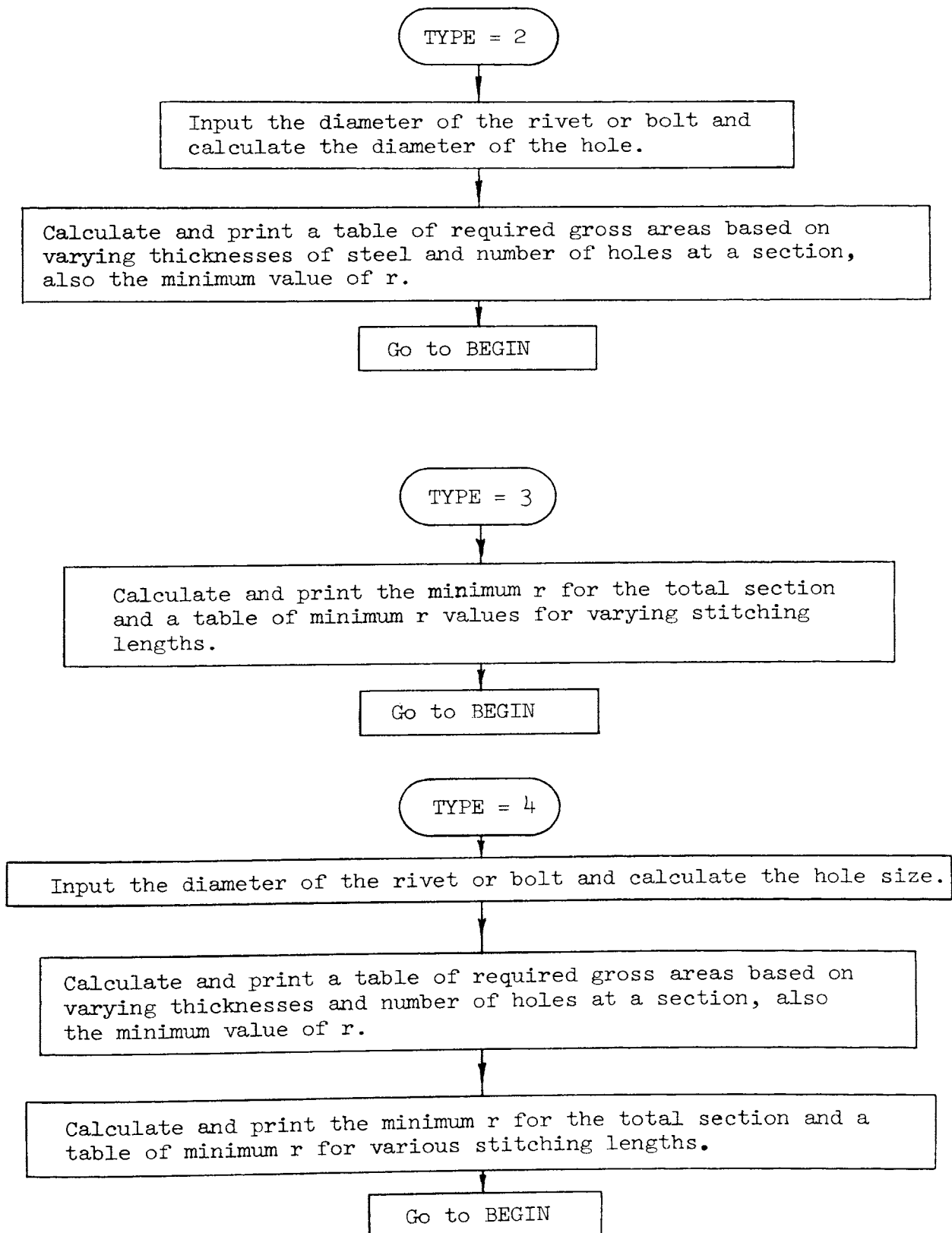
A tension design problem must be defined before execution of TENDSN.

P - Tension force.
F_y - Yield stress of the steel.
L - Length of the member.

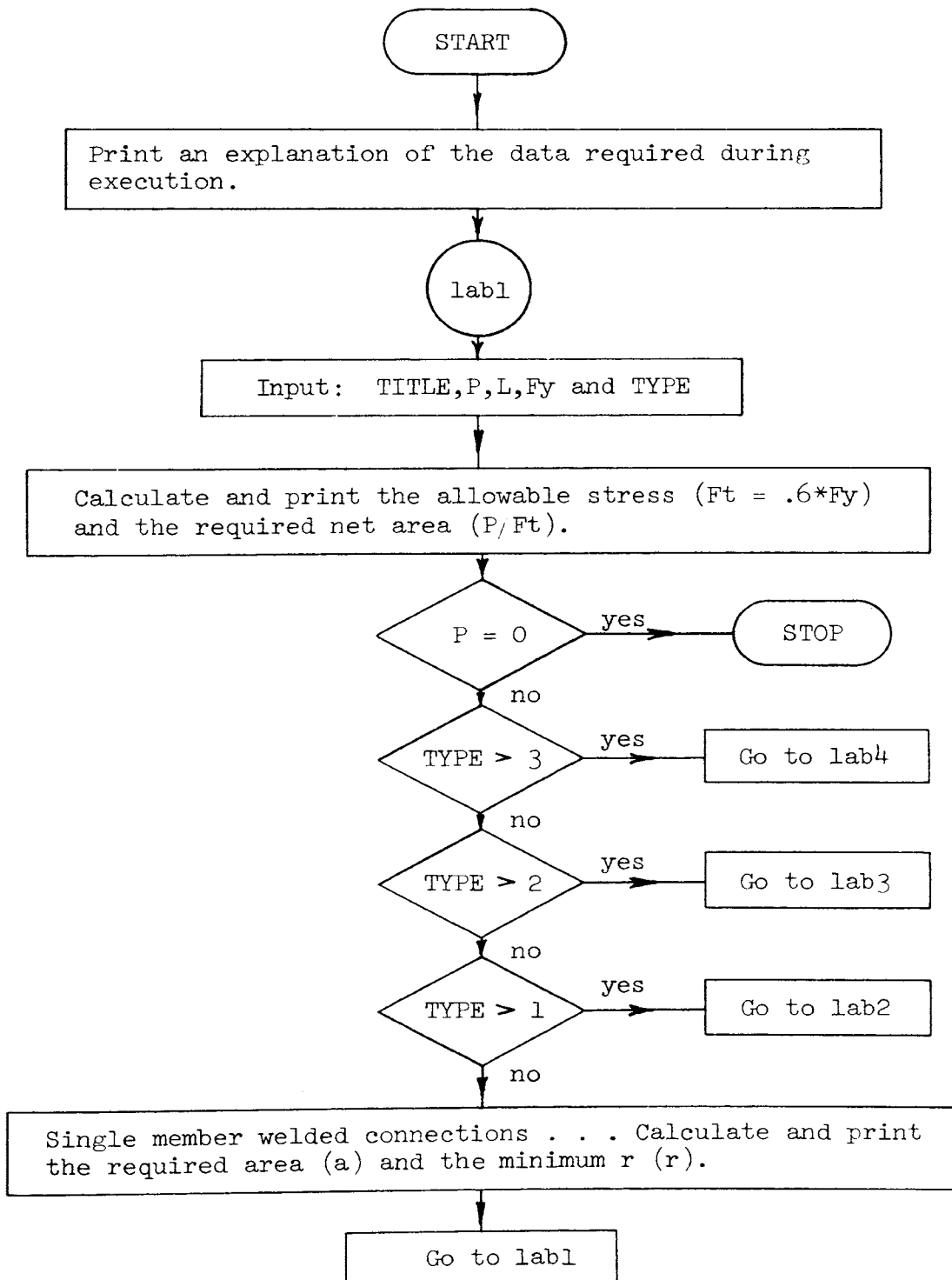
Type of member to be used i.e. single or built up members with riveted, bolted or welded connections.

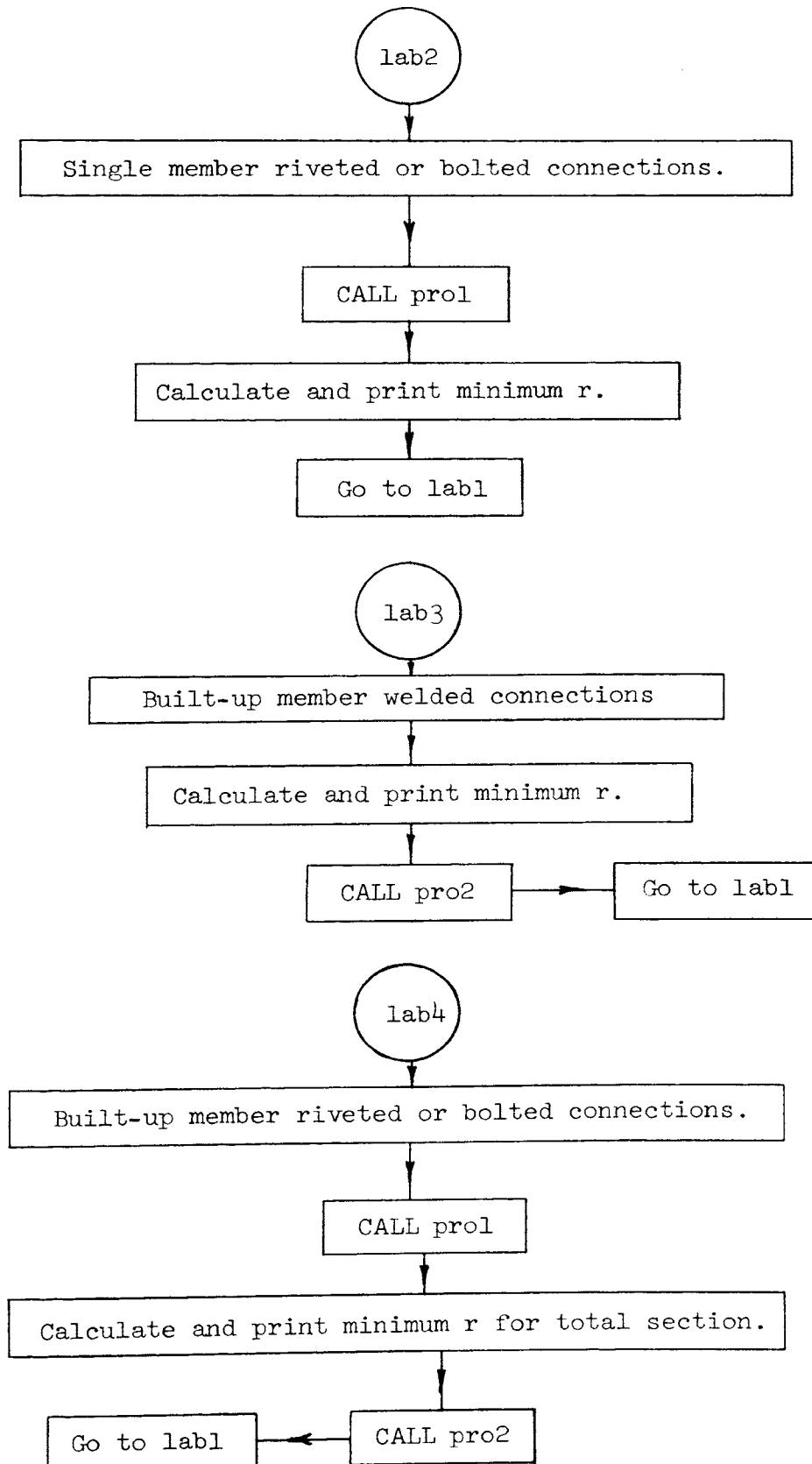
Overall Flow Chart TENDSN

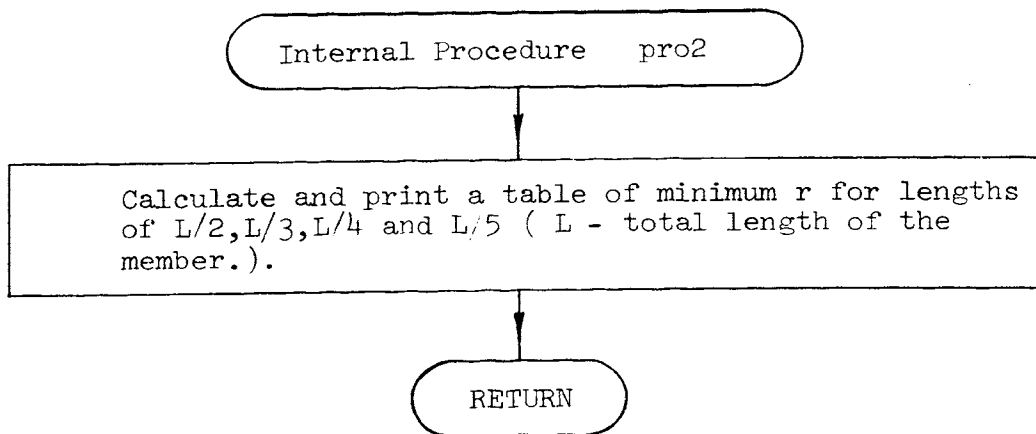
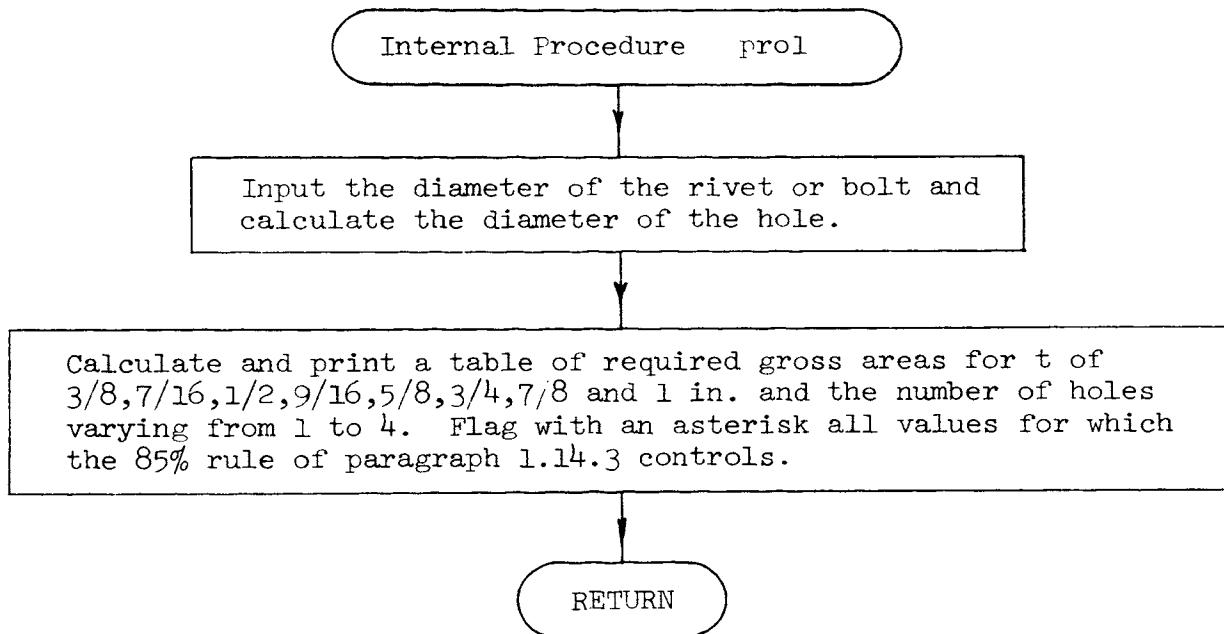




Detailed Flow Chart TENDSN







```

/*PROGRAM TENDSN*/
DECLARE fra(8) CHAR(4);
DECLARE ast(4) CHAR(1);
DECLARE area(4) ;
DECLARE TITLE CHAR(80);
fra(1)='3/8 ' ;
fra(2)='7/16' ;
fra(3)='1/2 ' ;
fra(4)='9/16' ;
fra(5)='5/8 ' ;
fra(6)='3/4 ' ;
fra(7)='7/8 ' ;
fra(8)='1 ' ;
LET round(x)=ceil(x*100)/100;
PUT LIST(' program TENDSN');
PUT LIST(' ');
PUT LIST('Purpose: program TENDSN will interact with the user in the design of
tension members. ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST('The following are parameters which will be required by TENDSN as the
design procedes. ');
PUT LIST(' TITLE - the title of the design for use by the user in identification
of the output. ');
ast(1)='';
PUT LIST(' (The title must be enclosed in single quotes ie',ast(1),
- - - title - - -,ast(1),'. ');
PUT LIST(' P - tension force (kips). ');
PUT LIST(' L - unbraced length of the member (ft.). ');
PUT LIST(' Fy- yield stress of the member (ksi). ');
PUT LIST(' TYPE - type of design');
PUT LIST(' 1 single member welded connections. ');
PUT LIST(' 2 single member riveted or bolted connections. ');
PUT LIST(' 3 built up member welded connections. ');
PUT LIST(' 4 built up member riveted or bolted connections. ');
PUT LIST(' To end the program input the tension force (P) as zero (0). ');

```

```

lab1:  PUT LIST(' ');
        PUT LIST(' ');
        GET LIST(TITLE);
        GET LIST(P);
        IF P=0 THEN GO TO enn;
        GET LIST(L,Fy);
        Ft=Fy*.6;
        Ft=round(Ft);
        IF Fy=36 THEN Ft=22;
        PUT LIST('Allowable tensile stress = Ft = .6*Fy = ',Ft,' ksi');
        PUT LIST('      (this must be < .5*minimum tension strength of material)');
        a=P/Ft;
        a=round(a);
        PUT LIST('Required net area = P/Ft = ',a,' sq.in.');
```

```

        PUT LIST(' ');
        r=L*12/240;
        r=round(r);
        GET LIST(TYPE);
        IF TYPE>3 THEN GO TO lab4;
        IF TYPE>2 THEN GO TO lab3;
        IF TYPE>1 THEN GO TO lab2;
        PUT LIST('Single member welded connections');
        PUT LIST(' ');
        PUT LIST('Required area = ',a,' sq.in.');
```

```

        PUT LIST('Minimum r = L/240 = ',r,' in.');
```

```

        GO TO lab5;
lab2:  PUT LIST('Single member riveted or bolted connections');
```

```

        PUT LIST(' ');
        CALL pro1;
        PUT LIST(' ');
        PUT LIST('Minimum r = L/240 = ',r,' in.');
```

```

        GO TO lab5;
lab3:  PUT LIST('Built up member welded connections');
```

```

        PUT LIST(' ');

```



```

PUT LIST('Minimum r for total section = L/240 = ',r,' in. ');
CALL pro2;
GO TO lab5;
lab4: PUT LIST('Built up member riveted or bolted connections');
PUT LIST(' ');
CALL pro1;
PUT LIST(' ');
PUT LIST('Minimum r for total section = L/240 ' ,r,'in. ');
CALL pro2;
lab5: PUT LIST(' ');
PUT LIST(TITLE);
PUT LIST(' ');
PUT LIST('Member chosen is _____.' );
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' ');
GO TO lab1;
enn: PUT LIST('Execution of TENDSN complete. ');
END ;
pro1: PROCEDURE ;
PUT LIST('What is the diameter of rivet or bolt to be used in the connections
(d) in. ');
GET LIST(d);
dh=d+1/16;
PUT LIST('The diameter of the hole = d+1/16= ',dh,' in. ');
PUT LIST(' ');
PUT LIST(' TABLE of required gross areas');
PUT LIST(' Number of Holes');
PUT IMAGE(1)(im3);
im3: IMAGE:
Thickness of steel 1 1 1 2 1 3 1 4
PUT LIST(' @ holes');
PUT LIST('
_____');

```

```

c=5/16;
loop1: DO j=1 to 8;
c=c+1/16;
IF j>5 THEN c=c+1/16;
loop2: DO i=1 to 4;
area(i)=max(c*dh*i+a,a/.85);
IF c*dh*i+a>a/.85 THEN ast(i)=' '; ELSE ast(i)='*';
END loop2;
PUT IMAGE(fra(j),area(1),ast(1),area(2),ast(2),area(3),ast(3),area(4),ast(4))(im1);
im1: IMAGE;
----- | ----.--- - | ----.--- - | ----.--- - | ----.--- -
END loop1;
PUT LIST(' * 85% gross section governs. AISC 1.14.3
');
RETURN ;
END pro1;
pro2: PROCEDURE ;
PUT LIST(' ');
PUT LIST('TABLE of minimum r for individual members of the built up section. ');
PUT LIST(' ');
PUT LIST('Length between stitching Minimum r ');
PUT LIST(' Rivets or Welds | ');
PUT LIST('_____ ');
loop3: DO i=2 TO 5;
PUT IMAGE(i,L*12/(240*i))(im2);
END loop3;
im2: IMAGE;
L/- | ----.---
RETURN ;
END pro2;

```

Program Name:

COLTST

Purpose:

Program COLTST will interact with the user in the design and investigation of compression members. Because the subject matter which it covers is relatively simple, COLTST was written primarily to serve as a vehicle for introducing trial and error design.

Limitations:

Only W, single and double angle sections may be designed using COLTST.

Prerequisite:

Read the following Sections of the AISC Specification:

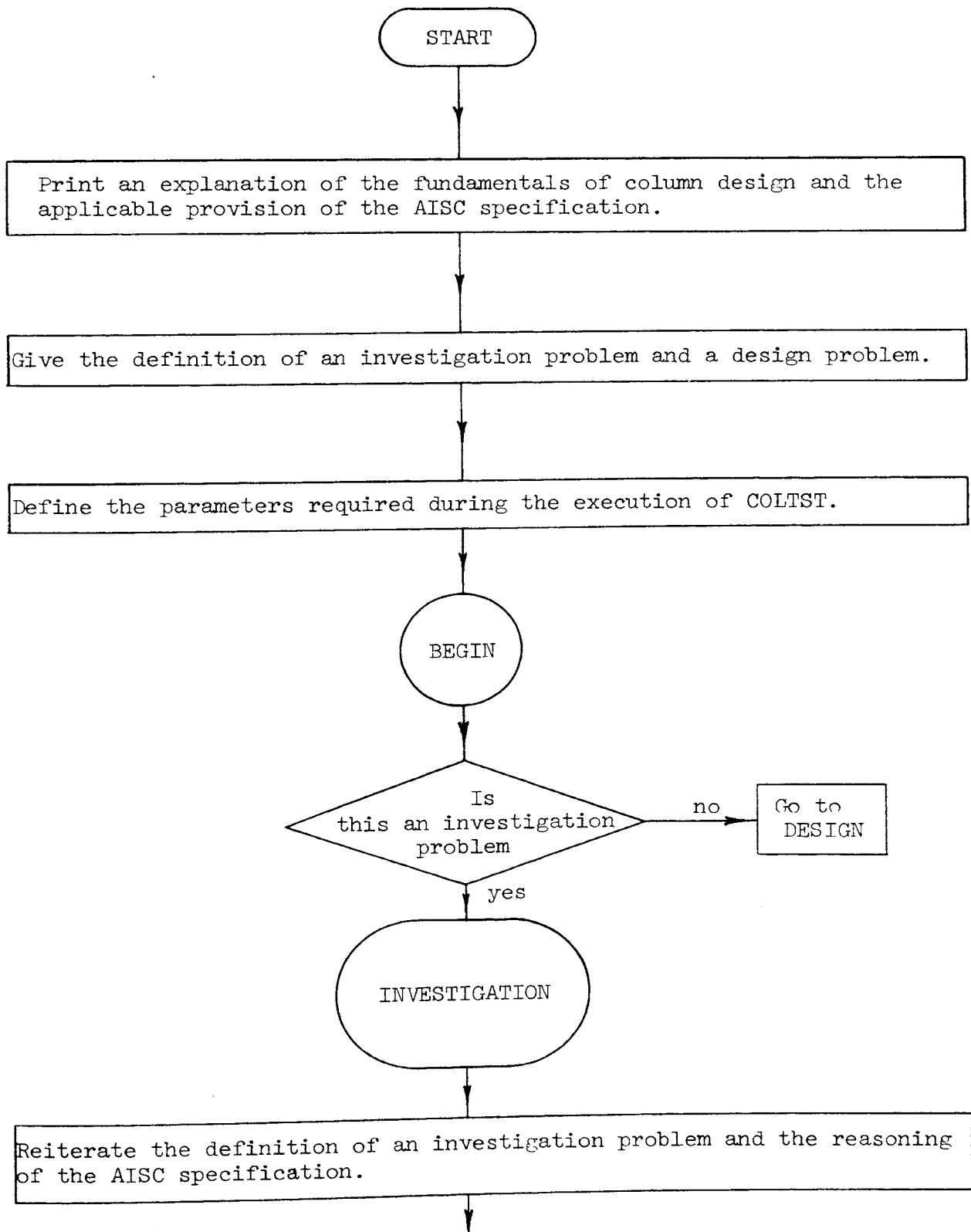
- 1.5.1.3 COMPRESSION.
- 1.8 STABILITY AND SLENDERNESS RATIOS.
- 1.9 WIDTH-THICKNESS RATIOS.

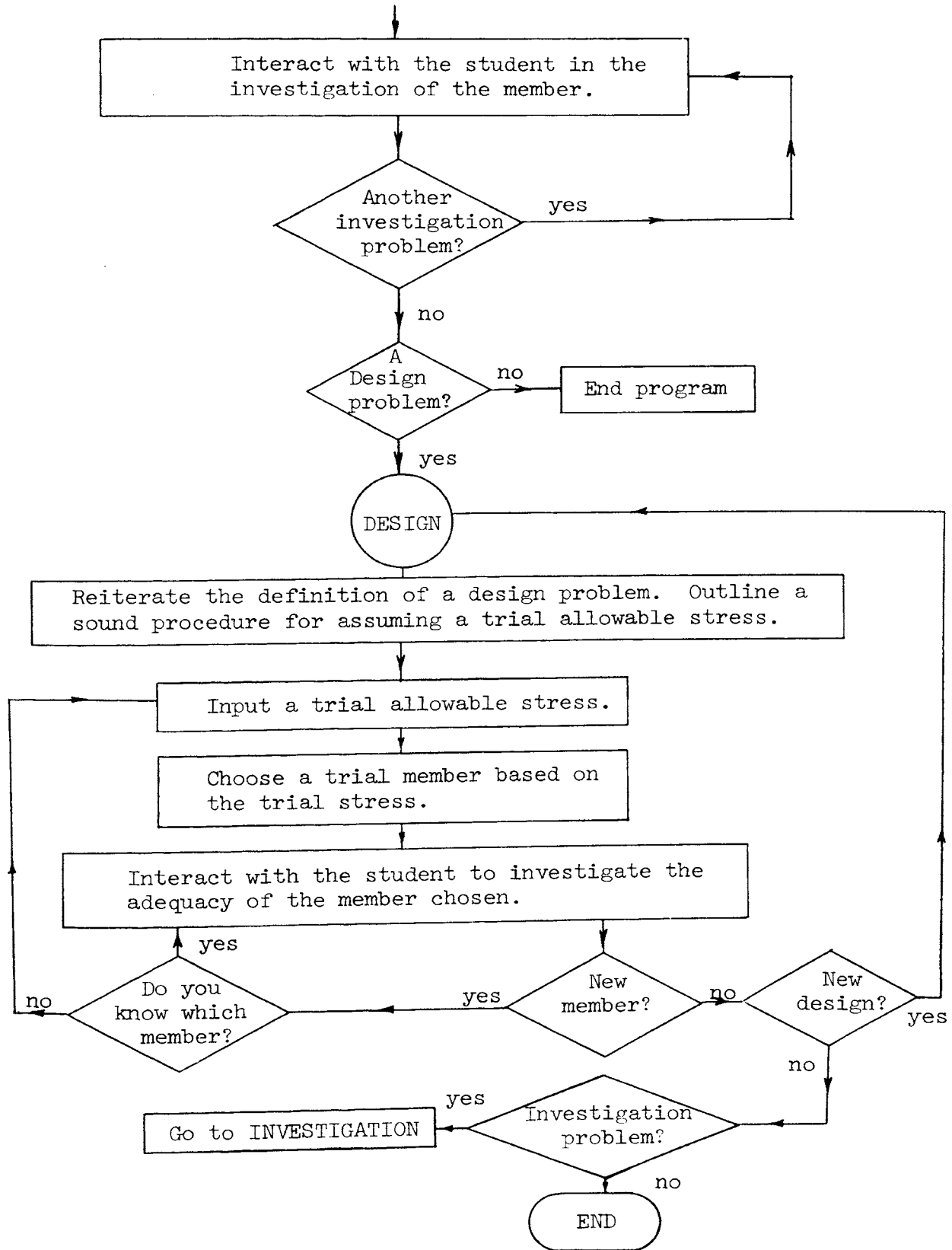
The student must have knowledge of the effective length factor K and how to calculate it.

Data Required Before Execution:

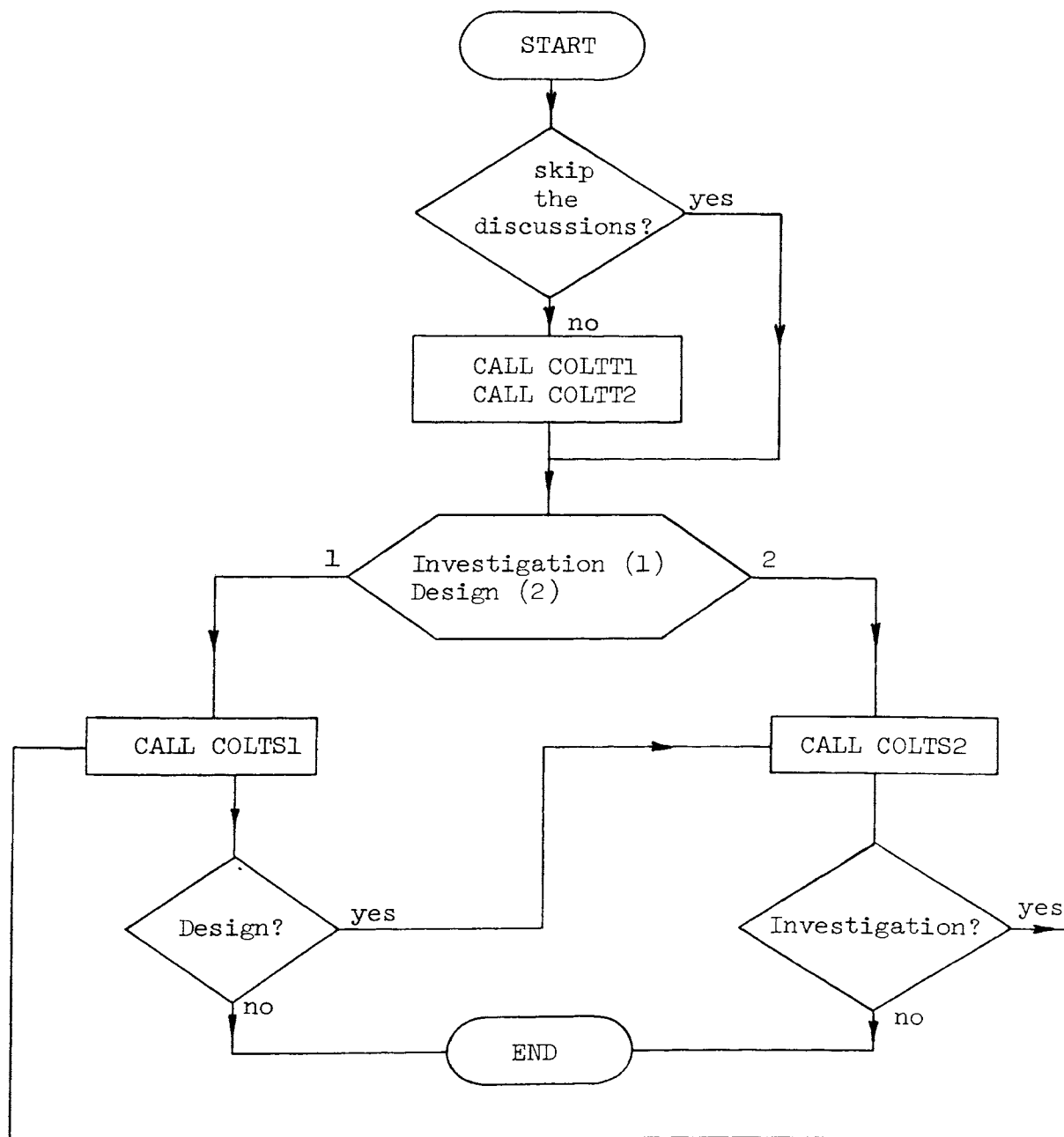
The column configuration is necessary, i.e. KL for the x, y and z axis. If a design problem is being worked the axial load is required; investigation of a member requires the member properties.

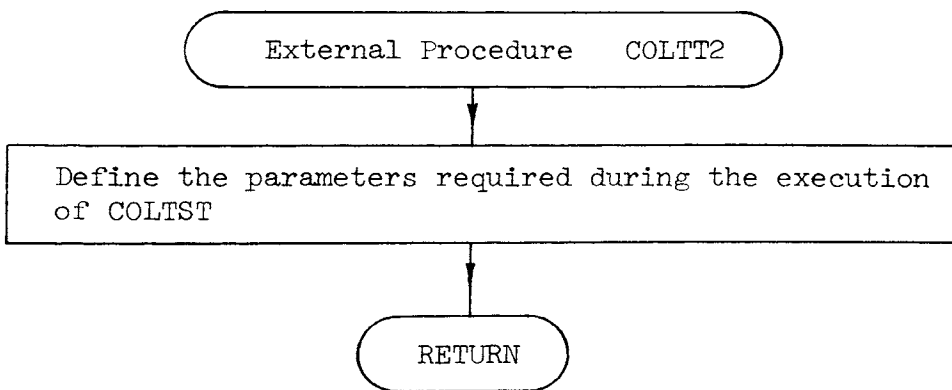
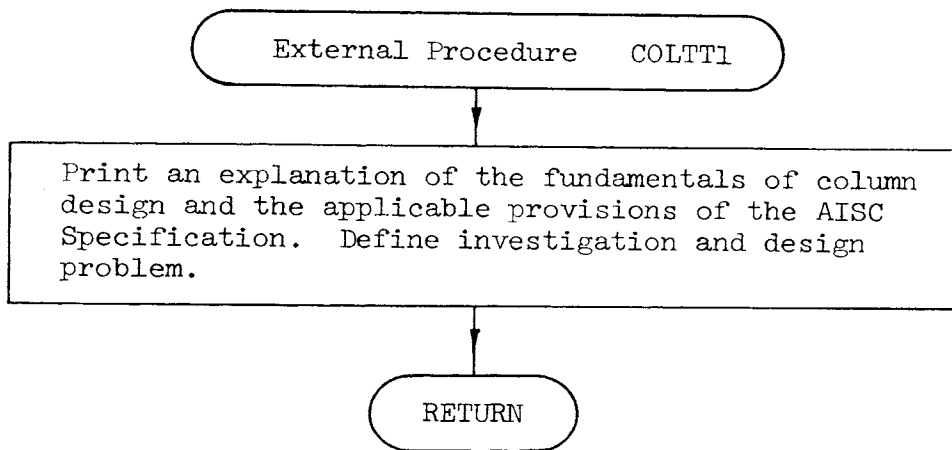
Overall Flow Chart COLTST

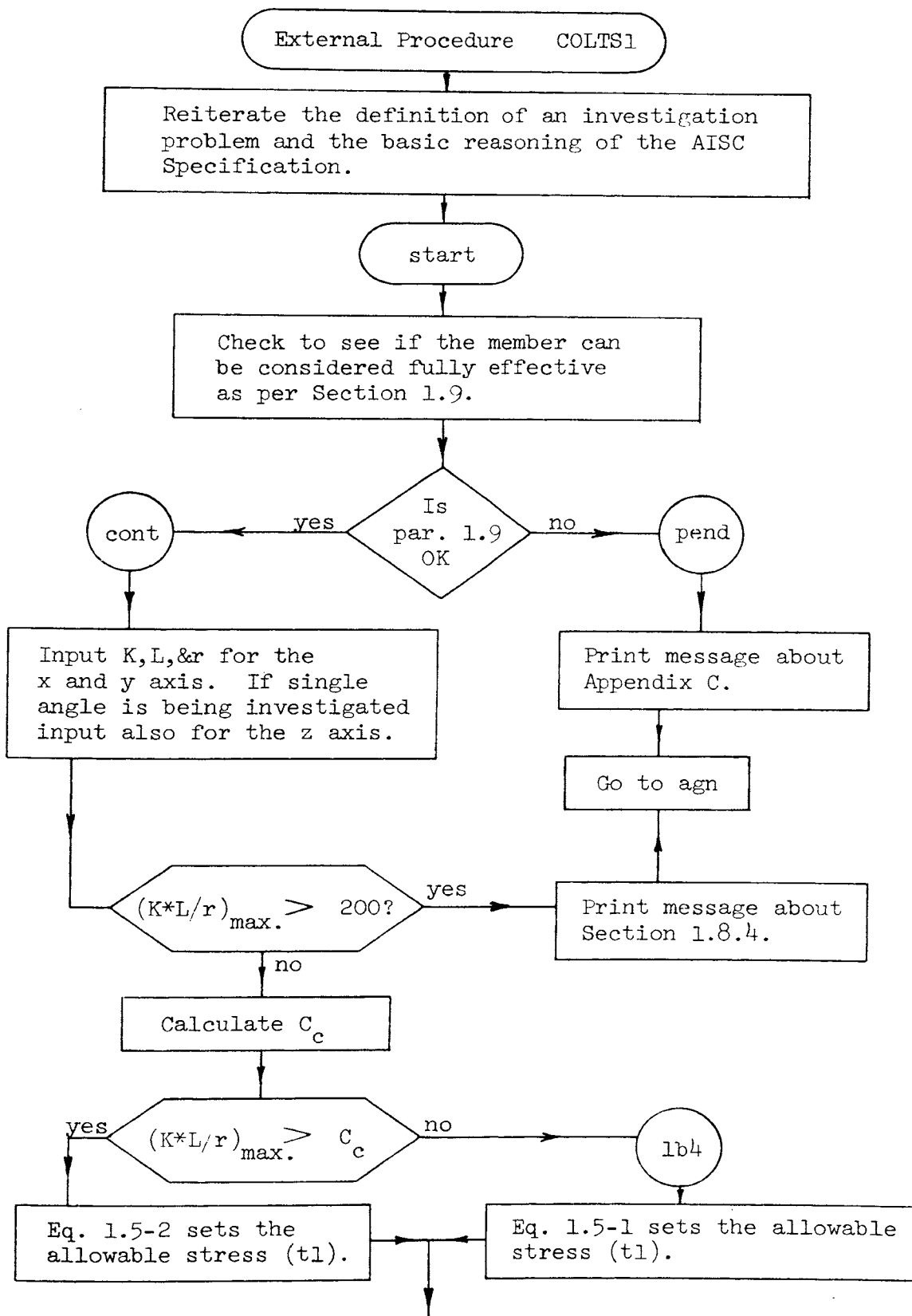


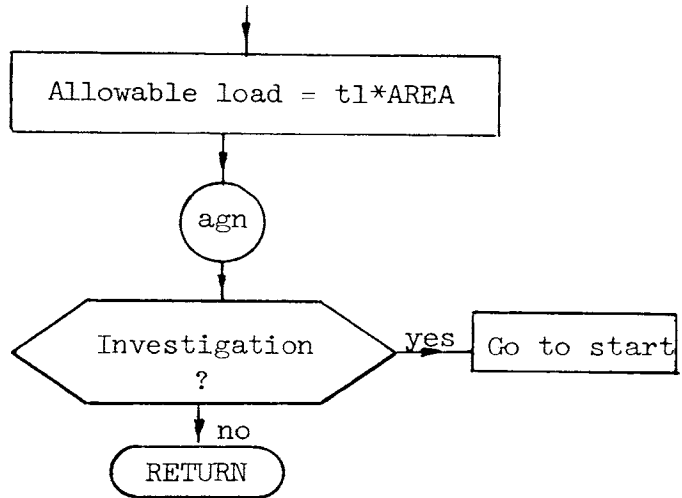


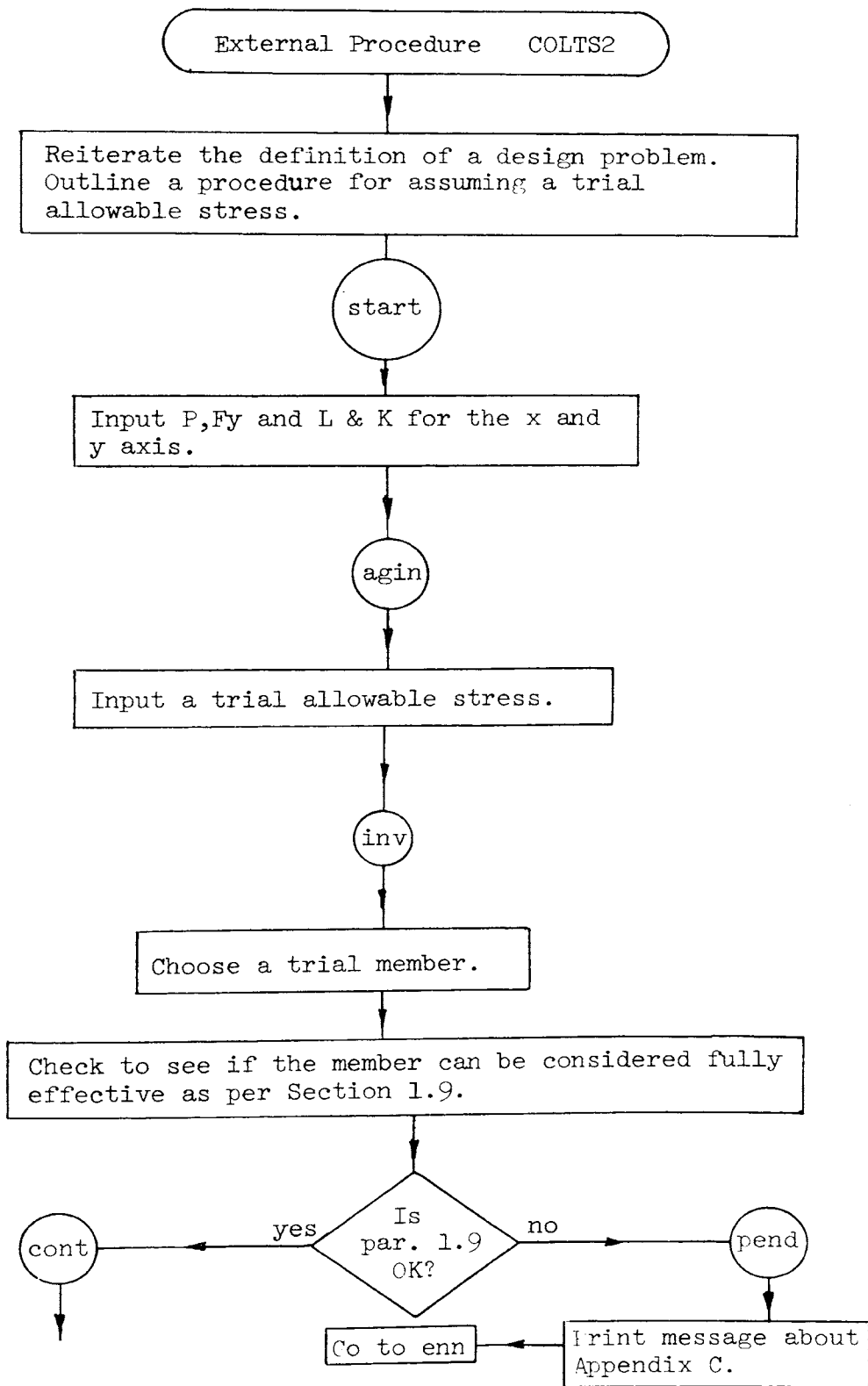
Detailed Flow Chart COLTST

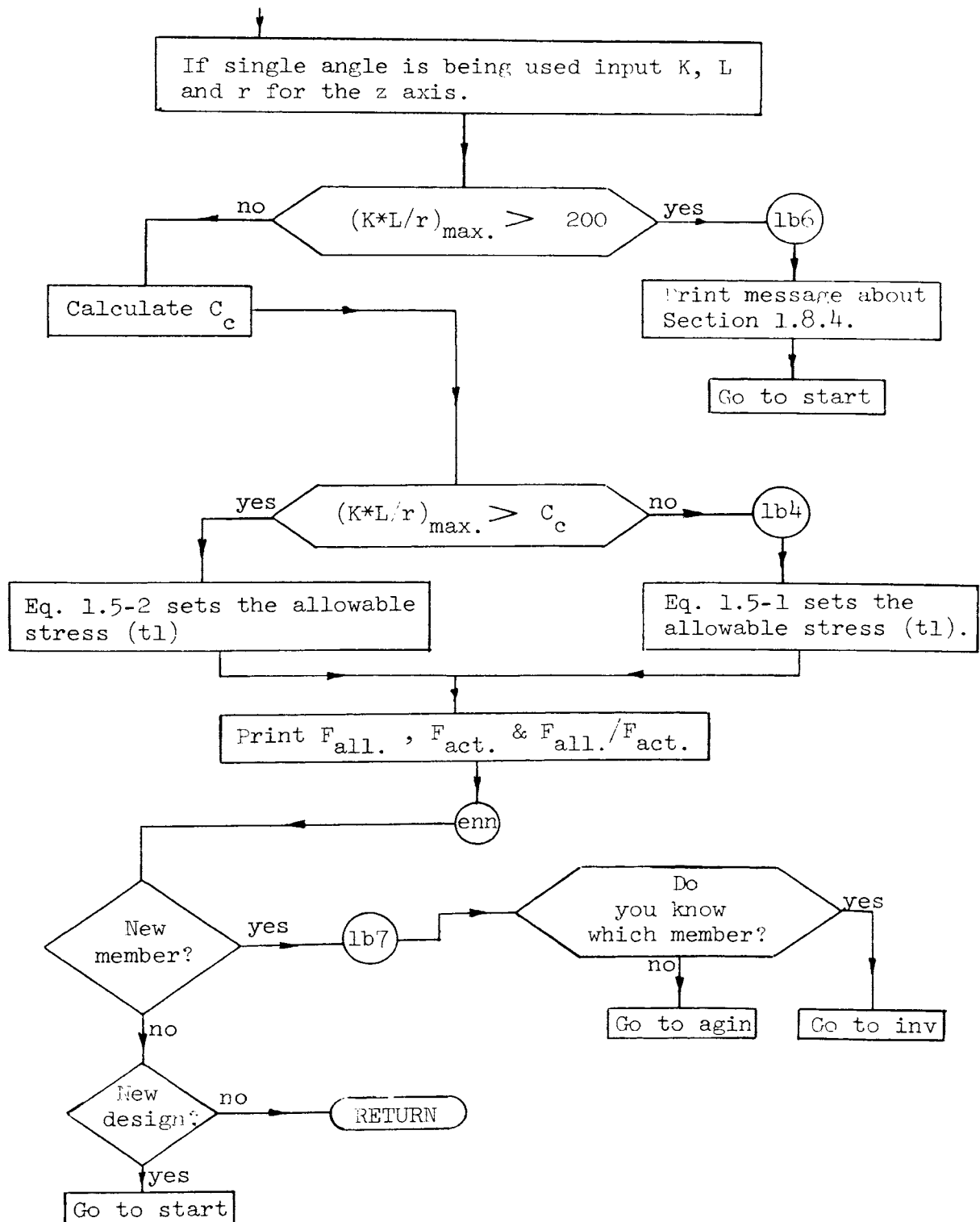












```

/*PROGRAM COLTST*/
DECLARE COLTS1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE COLTS2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE COLTT1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE COLTT2 ENTRY EXT KEY(whmIII) LIB(USER2);
PUT LIST('Do you want the discussion of the program printed? (type 1 for yes
and 0 for no)');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO not;
CALL COLTT1;
CALL COLTT2;
not: PUT LIST(' ');
PUT LIST(' ');
PUT LIST('Which type of problem do you want to work? ( Investigation = 1 Design
= 2 )');
GET LIST(ANSWER);
IF ANSWER=2 THEN GO TO dsn;
inv: CALL COLTS1;
PUT LIST('Would you like to work a design problem?');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO lab2;
dsn: CALL COLTS2;
PUT LIST('Would you like to try an investigation problem?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO inv;
lab2: END ;

```

```

COLTT1: PROCEDURE ;
        PUT LIST(' ');
        PUT LIST('Program COLTST will review with you the fundamentals of column
        design.');
```

design.');

```

        PUT LIST(' ');
        PUT LIST('Before proceeding you should have read the following sections:');
        PUT LIST(' 1.5.1.3 Compression.');
```

1.5.1.3 Compression.');

```

        PUT LIST(' 1.8 Stability and Slenderness Ratio.');
```

1.8 Stability and Slenderness Ratio.');

```

        PUT LIST(' 1.9 Width-Thickness Ratios.');
```

1.9 Width-Thickness Ratios.');

```

        PUT LIST(' ');
        PUT LIST('In general the design of a column is based on the mechanics of
        materials equation');
```

materials equation');

```

        PUT LIST('      f = P/A');
```

f = P/A');

```

        PUT LIST(' where:');
```

where:');

```

        PUT LIST('      P - axial load');
```

P - axial load');

```

        PUT LIST('      A - cross section area of the member');
```

A - cross section area of the member');

```

        PUT LIST('      f - the stress due to the load P');
```

f - the stress due to the load P');

```

        PUT LIST(' ');
        PUT LIST('The AISC sets forth the allowable stress due to compression.
        Basically, it sets the');
```

Basically, it sets the');

```

        PUT LIST(' allowable stress at 3/5*Fy by Eq. 1.5-1. This allowable is
        further reduced by the');
```

allowable stress at 3/5*Fy by Eq. 1.5-1. This allowable is further reduced by the');

```

        PUT LIST(' other terms in Eq. 1.5-1 to account for residual stresses and
        end restraints.');
```

other terms in Eq. 1.5-1 to account for residual stresses and end restraints.');

```

        PUT LIST(' Eq. 1.5-2 sets an allowable stress for the condition of elastic
        buckling, it is ');
```

Eq. 1.5-2 sets an allowable stress for the condition of elastic buckling, it is ');

```

        PUT LIST(' simply Eulers equation times a factor of safety. The effective
        length and the');
```

simply Eulers equation times a factor of safety. The effective length and the');

```

        PUT LIST(' radius of gyration of the member determines which equation is
        applicable.');
```

radius of gyration of the member determines which equation is applicable.');

```

        PUT LIST('In design two types of problems are encountered. One gives the
        section and asks for');
```

In design two types of problems are encountered. One gives the section and asks for');

```

        PUT LIST(' the allowable load. The other gives the load and asks for a
        member which will ');
```

the allowable load. The other gives the load and asks for a member which will ');

```
PUT LIST(' support it economically. The first is referred to as an  
  investigation problem;');  
PUT LIST(' the latter is a design problem.');
```

```
RETURN ;
```

```

COLTT2:  PROCEDURE ;
          PUT LIST(' ');
          PUT LIST('The following are parameters which are required in the execution of
COLTST:');
          PUT LIST('  Fy - yield stress of the steel (ksi).');
          PUT LIST('  Fa - allowable stress of the member (ksi).');
          PUT LIST('  E - modulus of elasticity (29000 ksi).');
          PUT LIST('  Ly ' distance between bracing against bending about the Y axis
(ft).');
          PUT LIST('  Lx - distance between bracing against bending about the X axis
(ft).');
          PUT LIST('  Lz - distance between bracing against bending about the Z axis
(ft).');
          PUT LIST('  Ky - effective length factor for bending about the Y axis. ');
          PUT LIST('  Kx - effective length factor for bending about the X axis. ');
          PUT LIST('  Kz - effective length factor for bending about the Z axis. ');
          PUT LIST('      (recommended values of K are found on Page 5-138 of the
AISC Specifications.)');
          PUT LIST('  rx - radius of gyration about the X axis (in.).');
          PUT LIST('  ry - radius of gyration about the Y axis (in.).');
          PUT LIST('  rz - radius of gyration about the Z axis (in.).');
          PUT LIST('      ( Data for the Z axis will be required only for single angle
members.)');
          PUT LIST('  P - axial load (kips)');
          PUT LIST('  ANSWER - usually requiring a yes or no indication (1 = yes  0 =
no).');
          PUT LIST('  AREA - cross section area (in**2).');
          RETURN ;
END COLTT2;

```

```

COLTS1: PROCEDURE ;
DECLARE ttt CHAR(1);
DECLARE t1 DEC(4), Cc DEC(4), t2 DEC(4), t3 DEC(4);
PUT LIST(' ');
ON ATTENTION GO TO start;
PUT LIST('In an investigation problem the engineer is required to calculate the
axial');
PUT LIST(' load a certain member will resist, i.e. the column capacity.');
```

```

PUT LIST(' ');
PUT LIST('As previously stated the AISC Specifications sets the allowable stress
at');
```

```

PUT LIST(' 3/5*Fy plus a reduction due to column configuration, residual
stresses and');
```

```

PUT LIST(' elastic buckling. Therefore since the member is known the
allowable stresses');
```

```

PUT LIST(' can be calculated. Using this allowable stress the allowable load
is found');
```

```

PUT LIST(' by  $P = Fa \cdot A$ .');
```

```

PUT LIST(' ');
start: PUT LIST('Choose any W, single or double angle member and column configuration
and lets go thru an investigation');
```

```

PUT LIST(' of it for the allowable axial load.');
```

```

PUT LIST(' ');
PUT LIST('For your reference type any information which will help in
identification of the problem');
```

```

GET EDIT(ttt)(A(1));
PUT LIST(' ');
PUT LIST('INPUT:');
```

```

FET LIST(Fy,AREA);
sfy=sqrt(Fy);
PUT LIST('We must first check to see if the section may be considered fully
effective in resisting');
```

```

PUT LIST(' the axial compressive stress.');
```



```

PUT LIST('According to Section 1.9.1.2 the unstiffened elements may be considered
  as fully effective if');
PUT LIST('  width to thickness ratio is less than');
t1=76/sfy;
PUT LIST('    76/Fy**.5 = ',t1,' for single angle struts; double angle struts
  with separators.');
```

```

t1=95/sfy;
PUT LIST('    95/Fy**.5 = ',t1,' for double angle struts in contact; flanges
  of beams.');
```

```

PUT LIST('Is this criteria satisfied?');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO pend;
t1=253/sfy;
PUT LIST('According to Section 1.9.2.2 stiffened elements may be considered
  fully effective if');
PUT LIST('    253/Fy**.5 = ',t1,' for webs of compression elements.');
```

```

PUT LIST('Is this criteria satisfied" (indicate yes if angles are being used.)');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO pend;
GO TO cont;
pend: PUT LIST('The member cannot be considered as fully effective in resisting the
  load.');
```

```

PUT LIST('  The reduced area may be calculated as per Appendix C of the AISC
  Specifications.');
```

```

PUT LIST('  Since the provisions of this Appendix have not been included in
  COLTST you');
```

```

PUT LIST('  have the option now of terminating the program and calculating the
  reduced');
```

```

PUT LIST('  area by hand or you may try a new member.');
```

```

PUT LIST('Do you want to terminate the program?');
GET LIST(ANSWER);
IF ANSWER=1 THEN PUT LIST('What guts!'); ELSE GO TO 1b1;
STOP ;
1b1: PUT LIST('I thought you would!);
GO TO agn;
```

```

cont:  PUT LIST('We must now check to see which of equations 1.5-1 and 1.5-2 sets the
        allowable stress. ');
        PUT LIST('Is this a single angle member? ');
        GET LIST(ANSWER);
        angle=ANSWER;
        PUT LIST('INPUT: ');
        GET LIST(rx,ry,Lx,Ly,Kx,Ky);
        IF angle=1 THEN GET LIST(rz,Lz,Kz);
        Cc=sqrt(2*9.86959*29000/Fy);
        PUT LIST('Cc = (2*(pi)**2*E/Fy)**.5 = ',Cc);
        t1=Kx*Lx*12/rx;
        t2=Ky*Ly*12/ry;
        IF angle=1 THEN t3=Kz*Lz*12/rz; ELSE t3=0;
        IF angle=1 THEN GO TO an1;
        PUT LIST('K*L/r      X = ',t1,'   Y = ',t2);
        GO TO an2;
an1:  PUT LIST('K*L/r      X = ',t1,'   Y = ',t2,'   Z = ',t3);
an2:  t1=max(t1,t2,t3);
        IF t1>200 THEN GO TO 1b6;
        PUT LIST('K*/r = ',t1,' controls the allowable stress. ');
        IF t1>Cc THEN GO TO 1b4;
        PUT LIST('Allowable stress set by Eq 1.5-1. ');
        t2=t1**2;
        t3=t2*t1;
        c2=Cc**2;
        c3=Cc**3;
        t1=(1-t2/(2*c2))*Fy/(5/3+3*t1/(8*Cc)-t3/(8*c3));
        GO TO 1b5;
1b4:  PUT LIST('Allowable stress set by Eq 1.5-2. ');
        t1=12*9.86959*29000/(23*t1**2);
1b5:  PUT LIST('Fa = ',t1);
        t1=AREA*t1;
        PUT LIST('The allowable axial load is ',t1,' kips. ');

```

```
agn:    PUT LIST(' ');
        PUT LIST('Would you like to try another investigation problem?');
        GET LIST(ANSWER);
        IF ANSWER=1 THEN GO TO start;
        RETURN ;
1b6:    PUT LIST('K*L/r > 200 therefore according to Section 1.8.4 this member cannot
        be used. ');
        GO TO agn;
        END ;
```

```

COLTS2:  PROCEDURE ;
          DECLARE ttt CHAR(1);
          DECLARE t1 DEC(4), Cc DEC(4), t2 DEC(4), t3 DEC(4);
          PUT LIST(' ');
          ON ATTENTION GO TO start;
          PUT LIST('Design is basically a trial and error procedure. The designer
                    chooses a');
          PUT LIST(' member based on his experience and checks to see if the member is
                    adequate.');
```

PUT LIST(' If the member is not satisfactory, he then chooses a new member
 based on');
 PUT LIST(' the experience just gained from the previous member.');

PUT LIST(' ');
 PUT LIST('Since the largest allowable stress the AISC Specifications will
 allow is $3/5 * F_y$, a');

PUT LIST(' good first choice is a member chosen based on an allowable stress
 of something');

PUT LIST(' smaller than this value. How much smaller depends on the value
 of $K * L / r$ of the');

PUT LIST(' member. The smaller this ratio the closer the chosen value should
 be.');

start: PUT LIST('Type any information which will help you identify this output.');

PUT LIST(' ');
 GET EDIT(ttt)(A(1));
 PUT LIST('INPUT:');

GET LIST(P,Fy);
 GET LIST(Lx,Ly,Kx,Ky);
 sfy=sqrt(Fy);
 t1=3/5*Fy;

PUT LIST('3/5*Fy = ',t1);

agin: PUT LIST('Assume a trial allowable stress.');

GET LIST(Fa);
 t1=P/Fa;
 PUT LIST('Based on this trial stress the required area is ',t1,' in**2.');

```

PUT LIST(' ');
PUT LIST('Choose a W, single or double angle member with this area keeping in
mind that a large value of r helps
inv: PUT LIST(' keep the allowable stress high. ');
PUT LIST('Type the name of the member chosen. ');
PUT LIST(' ');
GET EDIT(ttt)(A(1));
GET LIST(AREA);
PUT LIST('We must first check to see if the member may be considered fully
effective. ');
PUT LIST('According to Section 1.9.1.2 the unstiffened elements may be considered
as fully effective if');
PUT LIST(' width to thickness ratio is less than');
t1=76/sfy;
PUT LIST(' 76/Fy**.5 = ',t1,' for single angle struts; double angle struts
with separators. ');
t1=95/sfy;
PUT LIST(' 95/Fy**.5 = ',t1,' for double angle struts in contact; flanges
of beams. ');
t1=253/sfy;
PUT LIST('According to Section 1.9.2.2 the stiffened elements may be considered
fully effective if');
PUT LIST(' the width to thickness is less than');
PUT LIST(' 253/Fy**.5 = ',t1);
PUT LIST('Where applicable, is this criteria satisfied?');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO pend;
GO TO cont;
pend: PUT LIST('The member cannot be considered as fully effective in resisting the
load. ');
PUT LIST(' The reduced area may be calculated as per Appendix C of the AISC
Specifications. ');
PUT LIST(' Since the provisions of this Appendix have not been included in
COLTST you');

```

```

        PUT LIST('  have the option now of terminating the program and calculating
          the reduced');
        PUT LIST('  area by hand or you may try a new member. ');
        PUT LIST('Do you want to terminate the program? ');
        GET LIST(ANSWER);
        IF ANSWER=1 THEN PUT LIST('What guts! '); ELSE GO TO 1b1;
        STOP ;
1b1:    PUT LIST('I thought you would! ');
        GO TO enn;
cont:  PUT LIST('We must now check to see which of equations 1.5-1 and 1.5-2 sets the
          allowable stress. ');
        PUT LIST('Is this a single angle member? ');
        GET LIST(ANSWER);
        angle=ANSWER;
        PUT LIST('INPUT: ');
        GET LIST(rx,ry);
        IF angle=1 THEN GET LIST(rz,Lz,Kz);
        Cc=sqrt(2*9.86959*29000/Fy);
        PUT LIST('Cc = (2*(pi)**2*E/Fy)**.5 = ',Cc);
        t1=Kx*Lx*12/rx;
        t2=Ky*Ly*12/ry;
        IF angle=1 THEN t3=Kz*Lz*12/rz; ELSE t3=0;
        IF angle=1 THEN GO TO an1;
        PUT LIST('K*L/r      X = ',t1,'  Y = ',t2);
        GO TO an2;
an1:  PUT LIST('K*L/r      X = ',t1,'  Y = ',t2,'  Z = ',t3);
an2:  t1=max(t1,t2,t3);
        IF t1>200 THEN GO TO 1b6;
        PUT LIST('K*L/r = ',t1,' controls the allowable stress. ');
        IF t1>Cc THEN GO TO 1b4;
        PUT LIST('Allowable stress set by Eq 1.5-1. ');
        t2=t1**2;
        t3=t2*t1;
        c2=Cc**2;
        c3=Cc**3;

```

```

t1=(1-t2/(2*c2))*Fy/(5/3+3*t1/(8*Cc)-t3/(8*c3));
GO TO 1b5;
1b4: PUT LIST('Allowable stress set by Eq 1.5-2. ');
t1=12*9.86959*29000/(23*t1**2);
1b5: PUT LIST('Fa = ',t1);
t2=P/AREA;
PUT LIST('Actual stress = ',t2,' ksi. ');
t1=t1/t2;
PUT LIST('F-all/F-act = ',t1);
enn: PUT LIST('Would you like to try a new member? ');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b7;
PUT LIST('Would you like to try a new design problem? ');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO start;
RETURN ;
1b6: PUT LIST('K*L/r > 200 therefore according to Section 1.8.4 this member cannot
be used. ');
GO TO start;
1b7: PUT LIST('Do you know which member? ');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO agin; ELSE GO TO inv;
END ;

```

Program Name:

BENTST

Purpose:

The purpose of BENTST is to teach the procedure of investigation and design of a bending member. It is purposely detailed in its output to convey to the user exactly what is happening as the problem unfolds. It is a learning tool and is meant to be used by the student when working problems specifically assigned to supplement the lecture.

Limitations:

Only W members bent about the strong axis may be investigated or designed using BENTST. A C_p of one is used. An axial compressive stress of zero is assumed.

Prerequisite:

Read AISC Sections:

1.5.1.4 BENDING
1.9 WIDTH-THICKNESS RATIOS

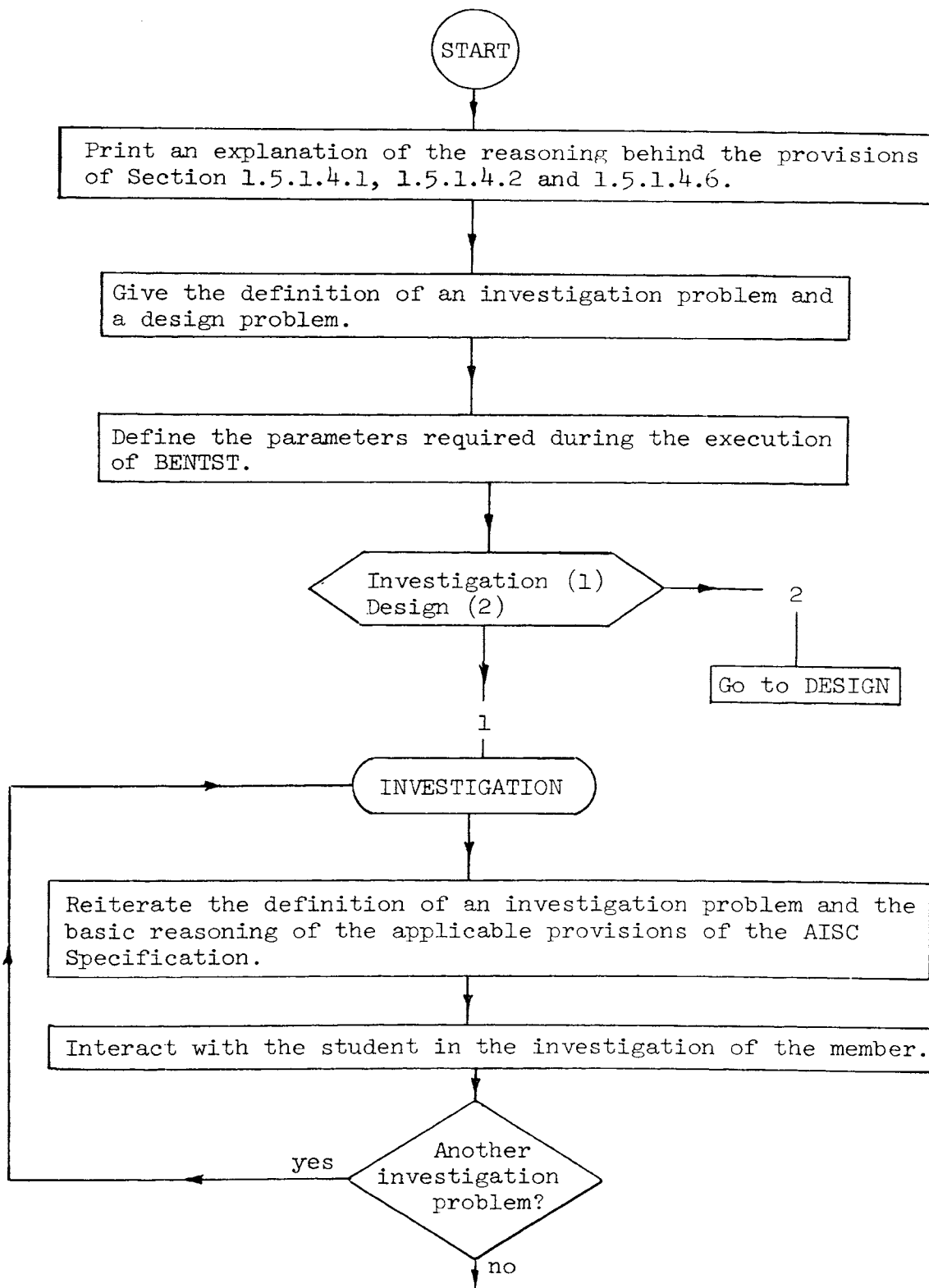
Data Required Before Execution:

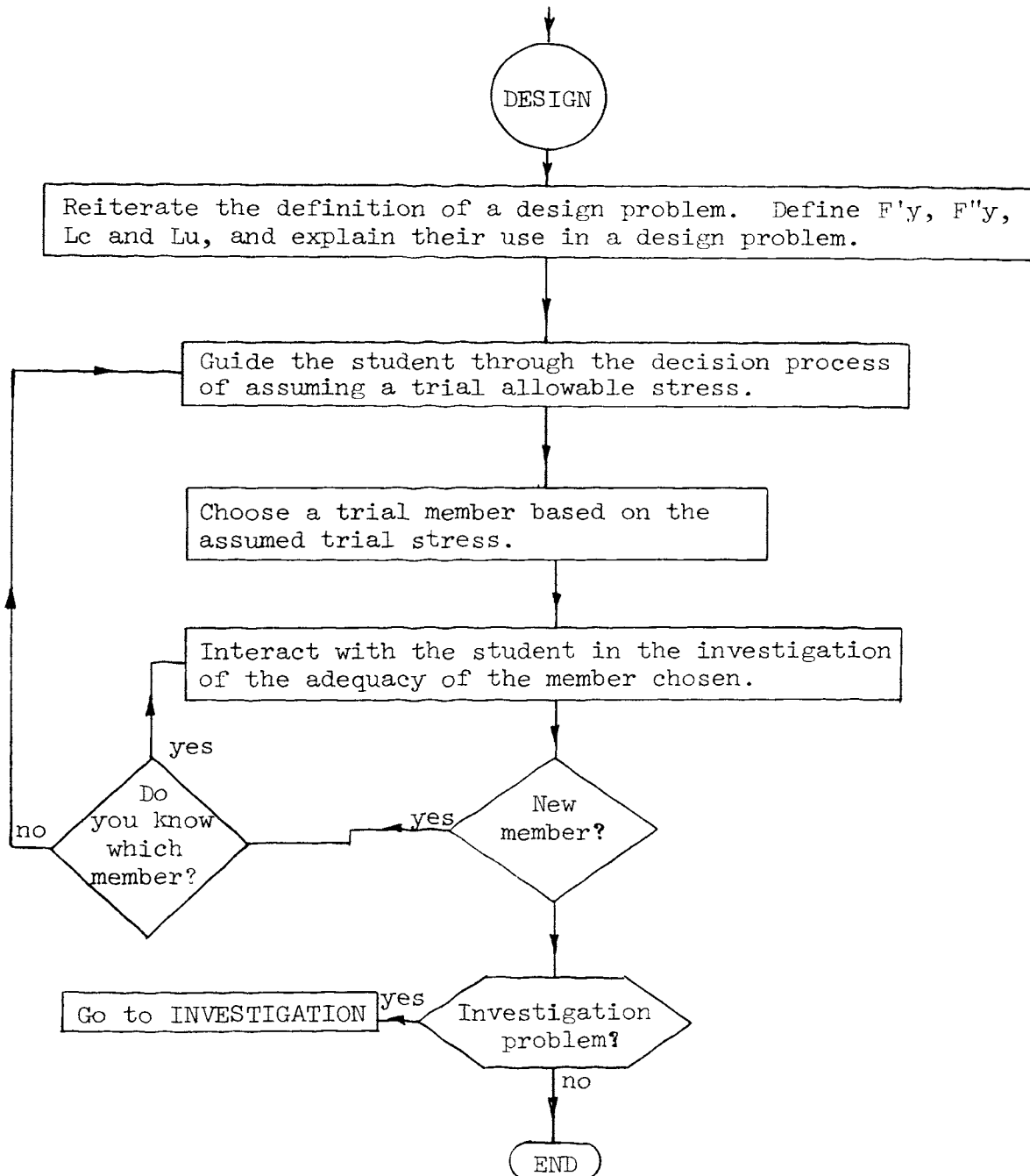
A bending investigation and/or design problem must be defined before execution of BENTST. The distance between cross-sections braced against twist or lateral displacement of the compression flange (L) must be defined.

Note:

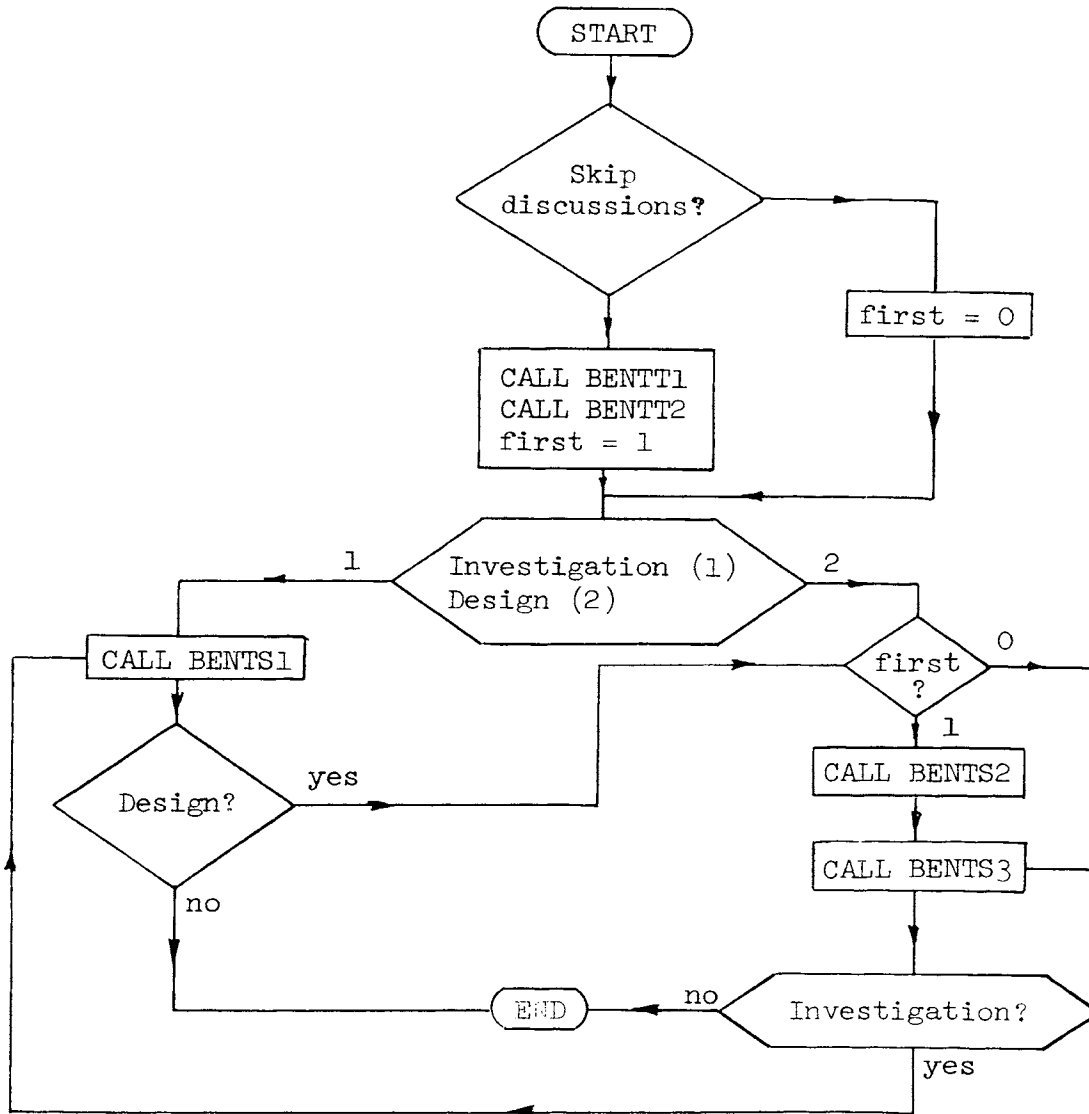
In the design portion of BENTST the use of the values of F_y' , F_y'' , L_c and L_u are introduced.

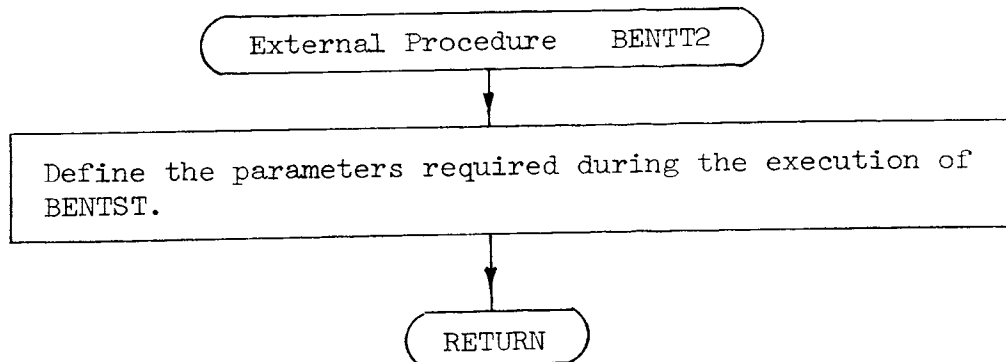
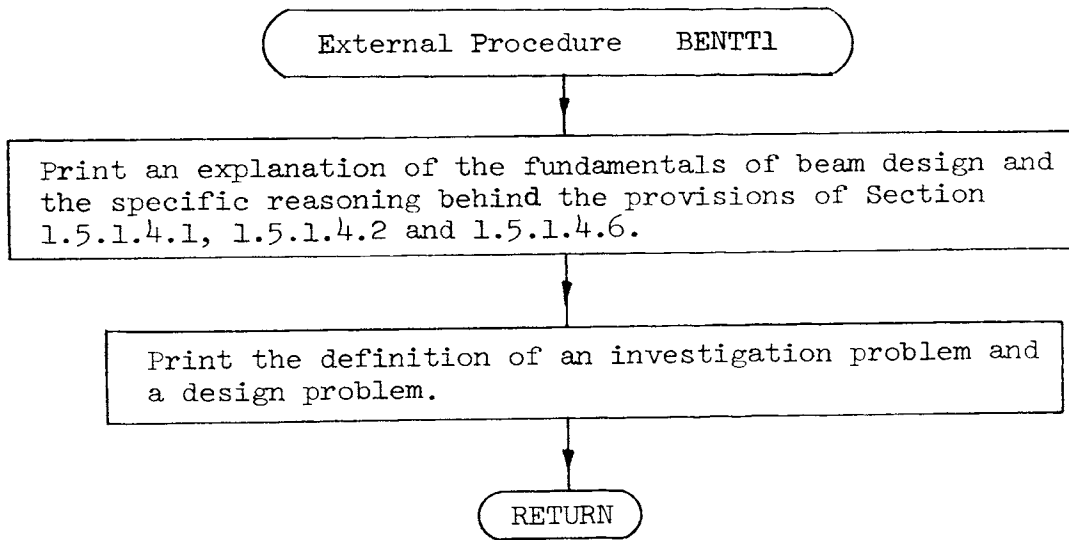
Overall Flow Chart BENTST

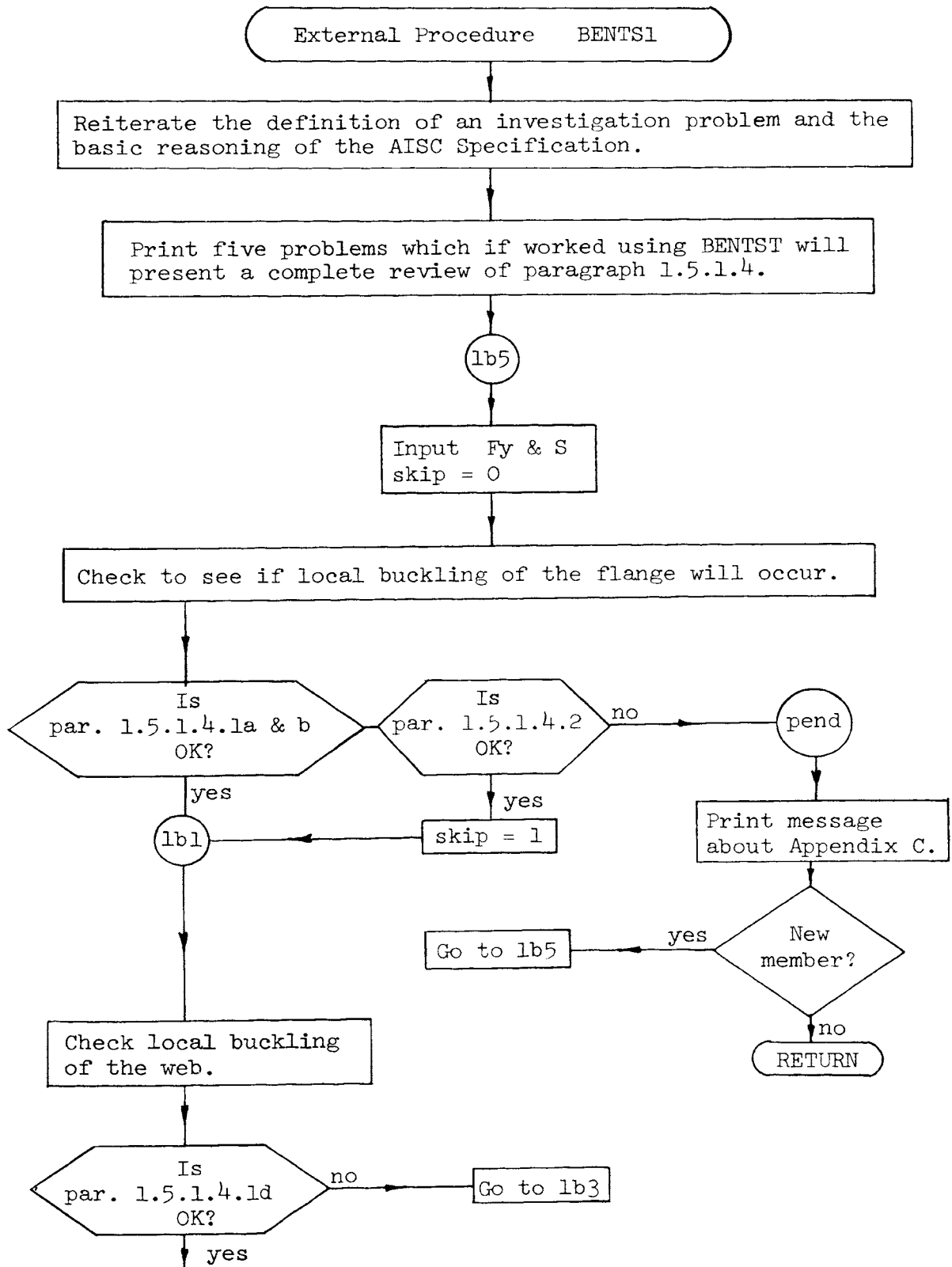


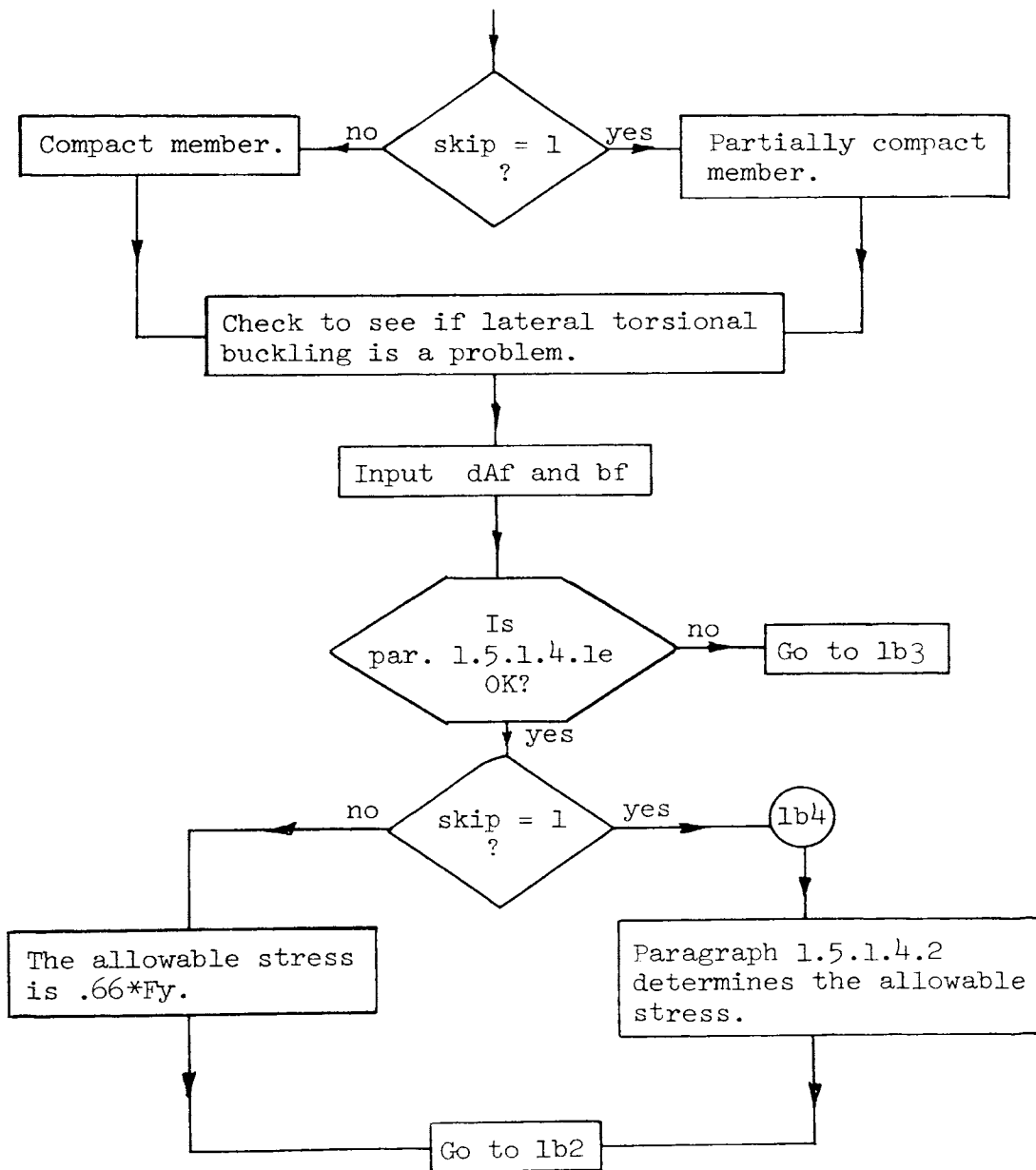


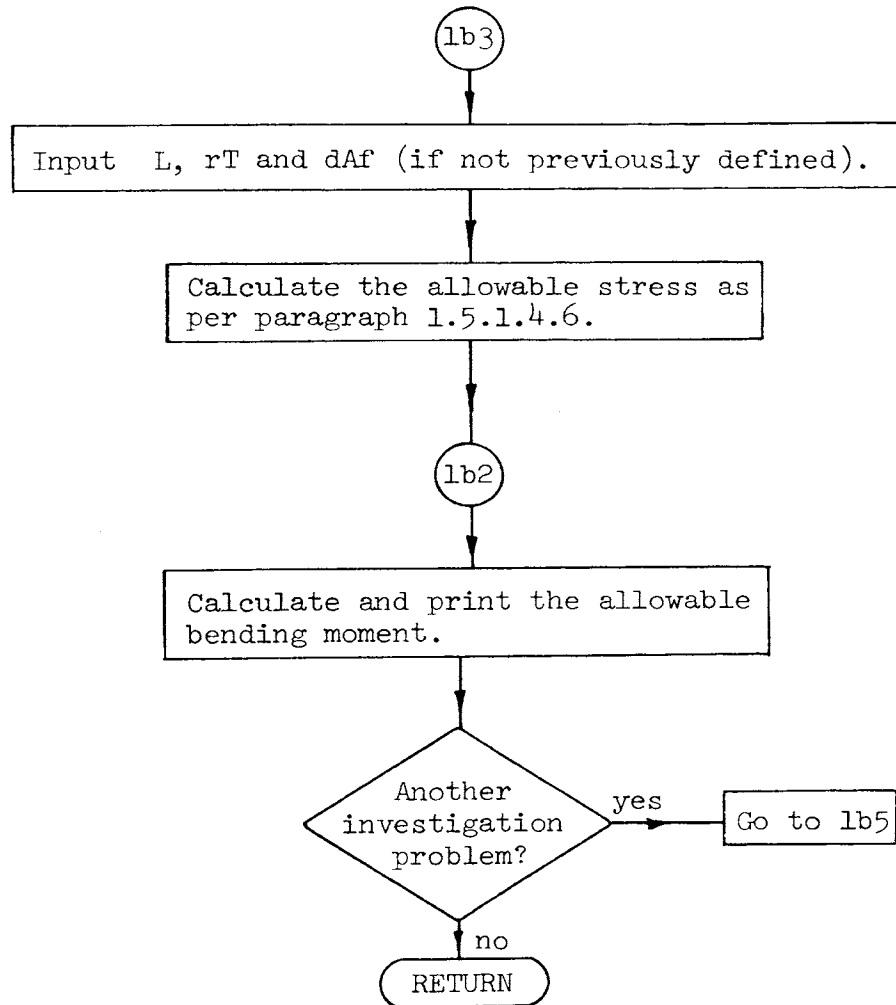
Detailed Flow Chart BENTST

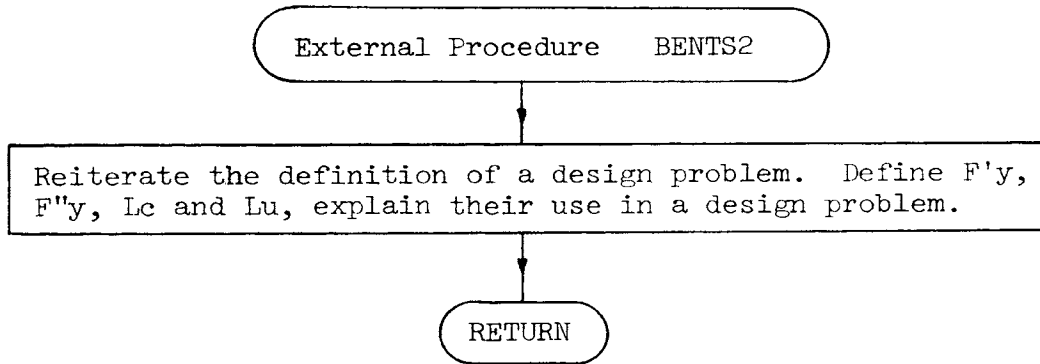


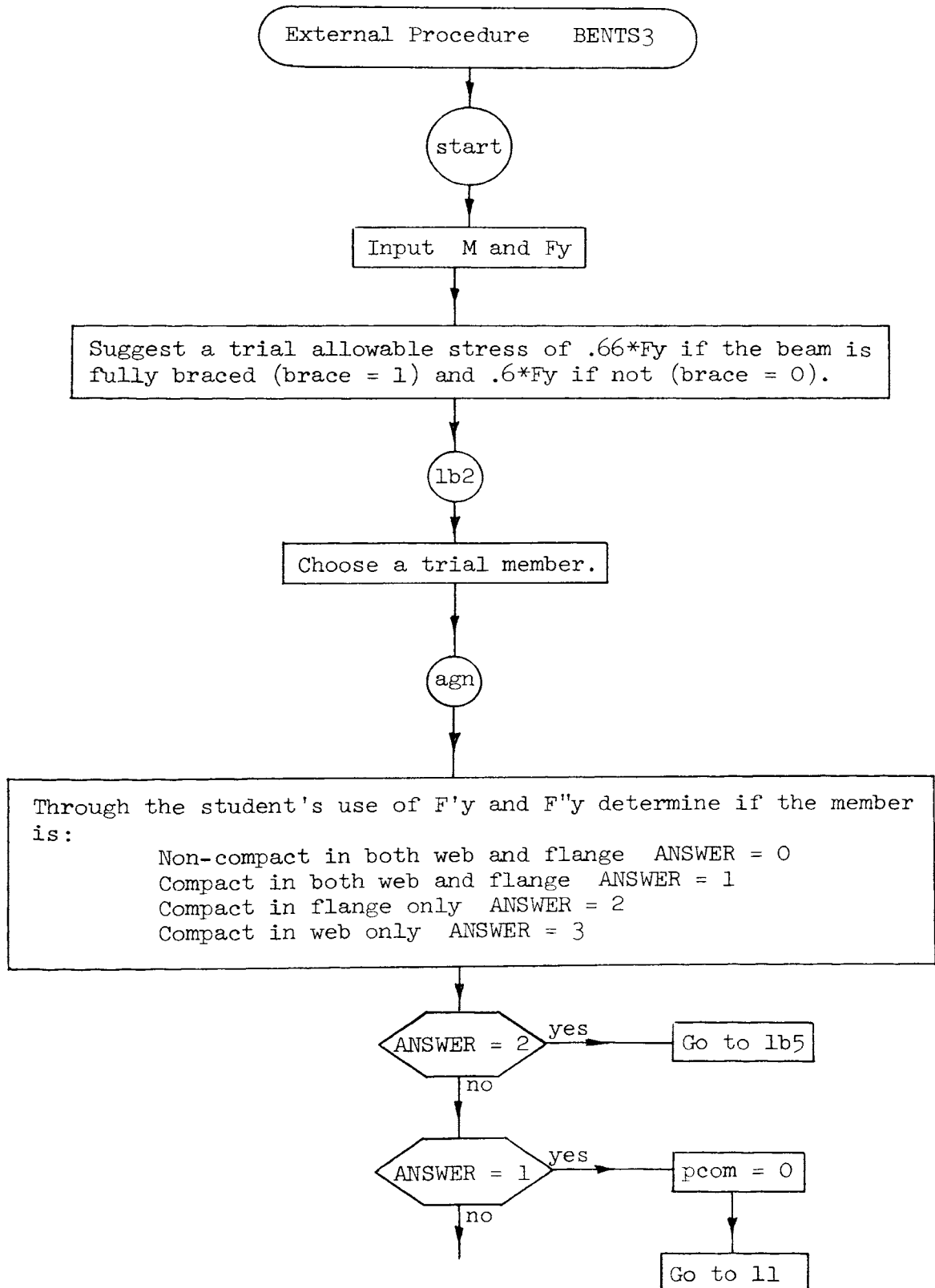


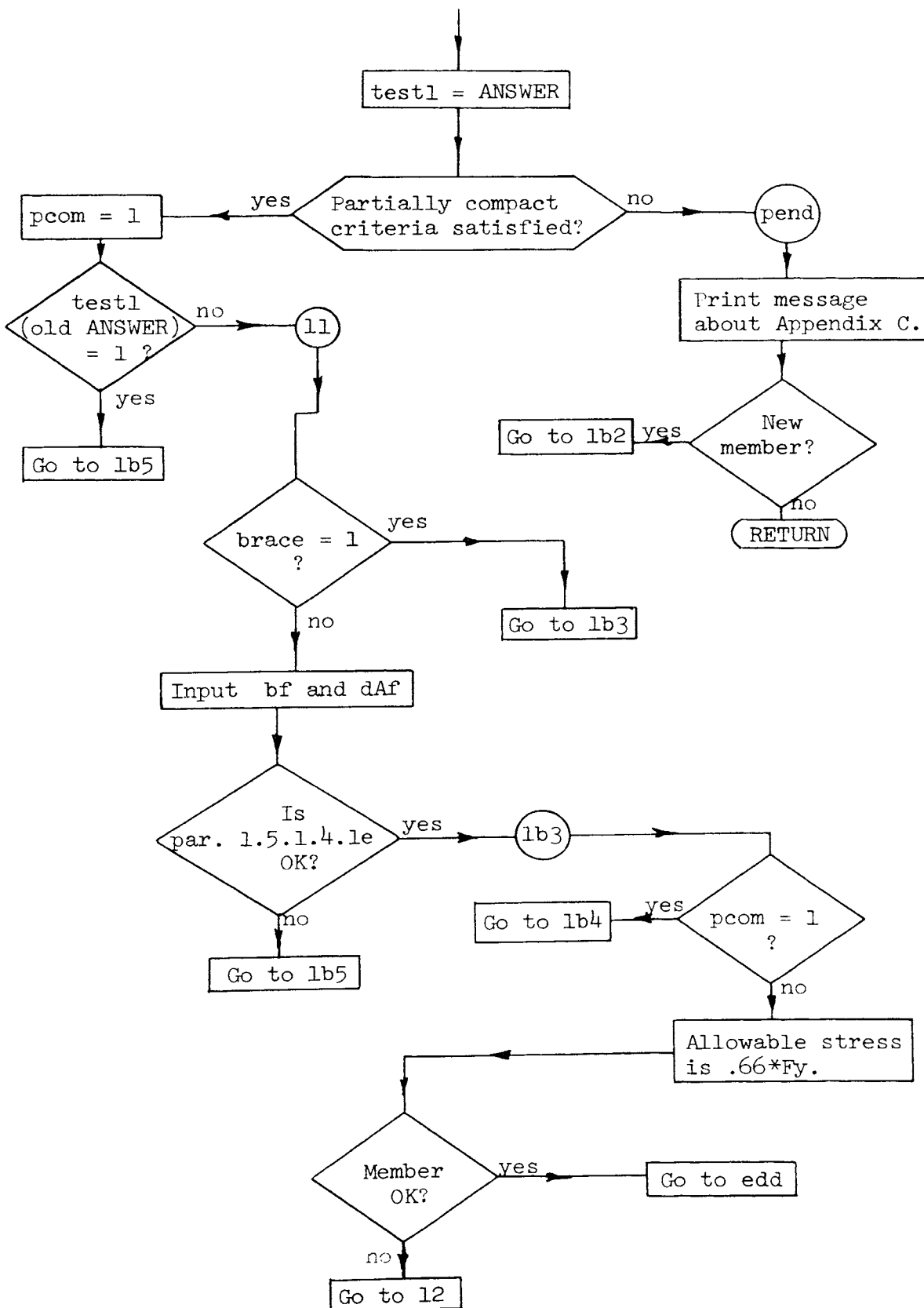


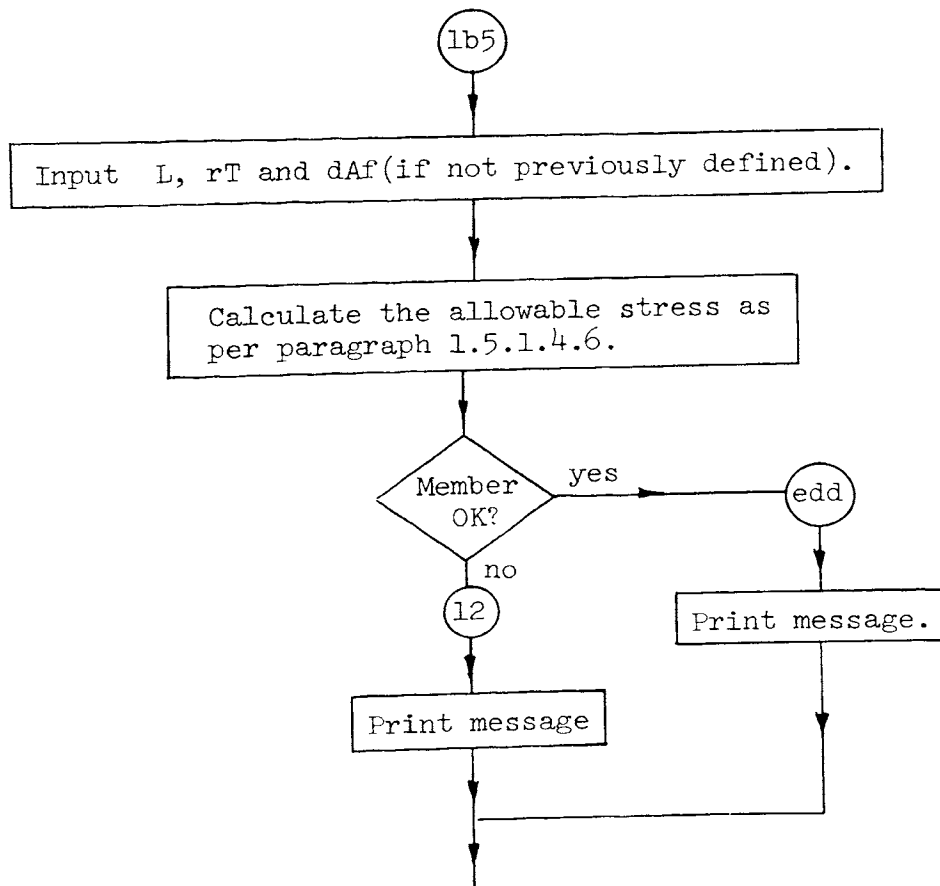
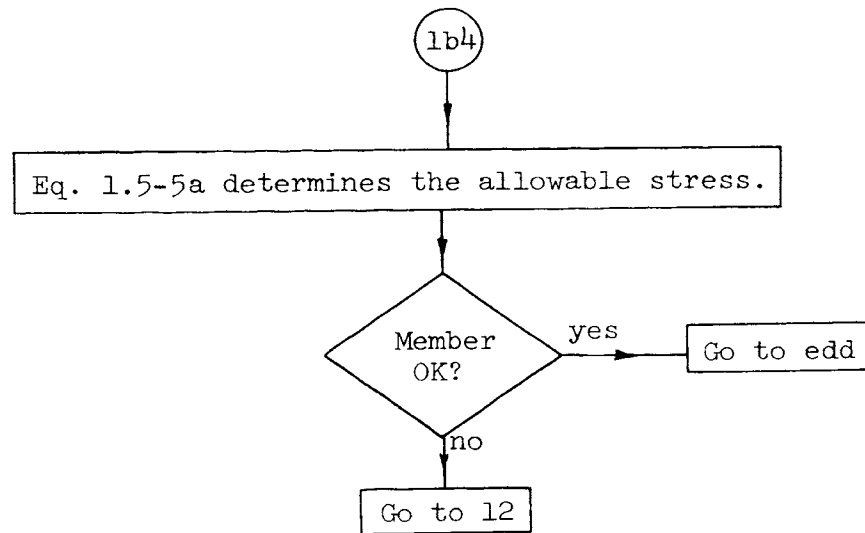


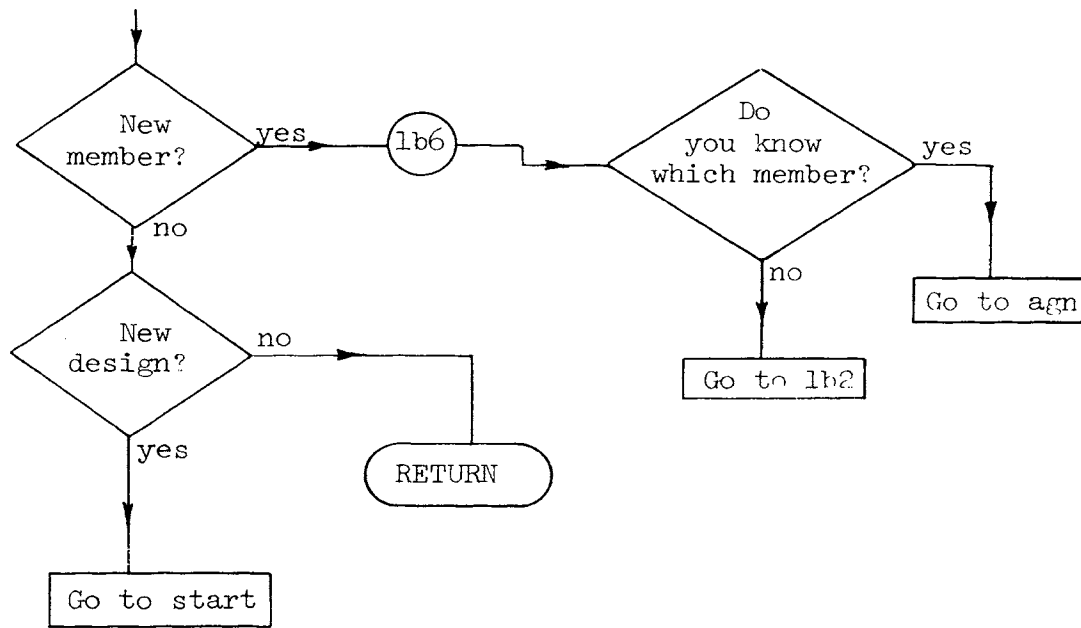












```

/*PROGRAM BENTST*/
DECLARE BENTT1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENTT2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENTS1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENTS2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENTS3 ENTRY EXT KEY(whmIII) LIB(USER2);
first=1;
PUT LIST('Would you like to skip the explanation of the program?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b1;
CALL BENTT1;
CALL BENTT2;
1b1: IF ANSWER=1 THEN first=0;
PUT LIST('Which type of problem would you like to review? (Investigation = 1
design = 2)');
GET LIST(ANSWER);
IF ANSWER=2 THEN GO TO dsn;
inv: CALL BENTS1;
PUT LIST('Would you like to try a design problem?');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO ed;
dsn: IF first=1 THEN CALL BENTS2;
first=0;
CALL BENTS3;
PUT LIST('Would you like to try an investigation problem?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO inv;
ed: PUT LIST(' ');
PUT LIST('You have just completed the review of investigation and design for
bending. ');
PUT LIST(' ');
PUT LIST('To perform design and analysis for bending with less output titles
and thus ');
PUT LIST(' more efficiently execute BENDSN. ');

```

```

BENTT1: PROCEDURE ;
PUT LIST(' ');
PUT LIST('Program BENTST will review with you the fundamentals of beam design.');
```

PUT LIST(' ');

PUT LIST(' ');

PUT LIST('Before proceeding you should have read the following sections:');

PUT LIST(' 1.5.1.4 Bending');

PUT LIST(' 1.9.1 Unstiffened Elements Under Compression');

PUT LIST(' 1.9.2 Stiffened Elements Under Compression');

PUT LIST(' ');

PUT LIST('In general the design of a bending member or a beam is based on the mechanics of materials');

PUT LIST(' equation for the stress due to bending, i.e.');

PUT LIST(' $f = M*c/I$ ');

PUT LIST(' where');

PUT LIST(' f - stress due to bending');

PUT LIST(' M - bending moment at a particular section');

PUT LIST(' c - distance to the extreme fiber as measured from the neutral axis');

PUT LIST(' I - moment of inertia');

PUT LIST(' ');

PUT LIST('The AISC Specification in paragraph 1.5.1.4 sets forth the allowable stress');

PUT LIST(' due to bending. Basically it sets the allowable stress at .66 times the yield');

PUT LIST(' stress of the steel, Ref. Paragraph 1.5.1.4.1. If the section, because of its');

PUT LIST(' lack of sufficient lateral support, buckles before the yield stress is reached');

PUT LIST(' this allowable stress is reduced as per Paragraph 1.5.1.4.6a or b. Such a');

PUT LIST(' beam will fail because of lateral torsional buckling. If the section,');

PUT LIST(' because of its shape, buckles locally before the yield stress is reached the');

```
PUT LIST(' allowable stress is reduced as per Paragraph 1.5.1.4.2 or 1.5.1.4.6a  
or b.');
```

```
PUT LIST(' Such a beam will fail because of local buckling, such a');
```

```
PUT LIST(' section is called non-compact.');
```

```
PUT LIST(' ');
```

```
PUT LIST('In design two types of problems are encountered. One gives the  
section and asks for');
```

```
PUT LIST(' the resisting moment or allowable load. The other gives the  
moment or load and');
```

```
PUT LIST(' asks for a section which will support it economically. The first  
is referred to');
```

```
PUT LIST(' as an investigation problem, the latter is a design problem.');
```

```
RETURN ;
```

```
END ;
```

```

BENTT2:  PROCEDURE ;
          PUT LIST('The following are parameters which are required in the execution
                of BENTST:');
          PUT LIST('  Fy - yield stress of the steel (ksi).');
          PUT LIST('  fa - allowable stress of the steel (ksi).');
          PUT LIST('  bf - width of the flange (in.).');
          PUT LIST('  tf - thickness of the flange (in.).');
          PUT LIST('  dAf - d/Af or the depth of the member divided by the area of the
                flange (1/in.).');
          PUT LIST('  rT - radius of gyration of a section comprising the compression
                flange plus');
          PUT LIST('          1/3 the compression web area taken about an axis in the
                plane of the web (in.).');
          PUT LIST('  L - distance between bracing of the compression flange (ft.).');
          PUT LIST('  S - section modulus of the member (in**3).');
          PUT LIST('  M - Bending Moment (ft.kips).');
          PUT LIST('  ANSWER - usually requiring a yes or no indication by 1 = yes or
                0 = no. ');
          RETURN ;
          END BENTT2;

```



```

BENTS1:  PROCEDURE ;
          DECLARE ttt CHAR(1);
          DECLARE test1 DEC(4);
          DECLARE test DEC(4);
          DECLARE t1 DEC(4);
          PUT LIST('In an investigation problem the engineer is required to calculate
                  what bending moment a');
          PUT LIST('    certain section will resist i.e. the resisting moment. ');
          PUT LIST('As previously stated the allowable stress is equal to .66 times the
                  yield stress of the');
          PUT LIST('    steel unless a reduction is specified by the AISC Specifications
                  either because of the');
          PUT LIST('    geometry of the section or the lateral bracing. Therefore since
                  the section is known');
          PUT LIST('    the allowable stress as per the AISC Specification can be
                  calculated. ');
          PUT LIST('Rearranging the terms in our stress equation we have  $M=I*f/c$  ');
          PUT LIST('An investigation of the following members will give the user a
                  review of bending. ');
          PUT LIST('    W18X55   Fy = 36ksi   L = 5ft. ');
          PUT LIST('    W18X55   Fy = 36ksi   L = 10ft. ');
          PUT LIST('    W21X55   Fy = 45ksi   L = 5ft. ');
          PUT LIST('    W21X55   Fy = 36ksi   L = 10ft. ');
          PUT LIST('    W21X55   Fy = 60ksi   L = 10ft. ');
1b5:     PUT LIST('Choose any W section and lets go thru an investigation of it for the
                  resisting moment. ');
          PUT LIST('    For your reference type the name of the member. ');
          GET EDIT(ttt)(A(1));
          PUT LIST('What is the section modulus of the member? ');
          GET LIST(S);
          GET LIST(Fy);
          tf=0;
          skip=0;
          dAf=0;
          sfy=sqrt(Fy);

```

```

PUT LIST('We will first check to see if local buckling is a concern.');
```

```

PUT LIST('If Section 1.5.1.4.1a and b are satisfied local buckling of the flange
  will not occur.');
```

```

t1=52.2/sfy;
PUT LIST('  f2.2/Fy**.5 = ',t1);
PUT LIST('Are these two Sections satisfied?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b1;
PUT LIST('Member is partially compact if by Section 1.5.1.4.2  52.2/Fy**.5
  < bf/2tf < 95/Fy**.5 and');
```

```

PUT LIST('  all other provisions of Section 1.5.1.4.1 are satisfied.');
```

```

t1=95/sfy;
PUT LIST('    95/Fy**.5 = ',t1);
PUT LIST('Is bf/2tf < 95/Fy**.5 ?');
```

```

/* skip= 1 member is partially compact. */;
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO pend; ELSE skip=1;
```

```

1b1: PUT LIST('If Section 1.5.1.4.1d is satisfied local buckling of the web will not
  occur.');
```

```

t1=412/sfy;
PUT LIST('  412(1-2.33fa/Fy)/Fy**.5 = ',t1,'  fa = 0 because no compression
  load.');
```

```

t1=257/sfy;
PUT LIST('    257/Fy**.5 = ',t1);
PUT LIST('Is Section 1.5.1.4.1d satisfied?');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO 1b3;
IF skip=1 THEN GO TO s1;
PUT LIST('The member is COMPACT i.e. no local buckling will occur.');
```

```

GO TO s2;
```

```

s1: PUT LIST('The member is partially compact.');
```

```

s2: PUT LIST(' ');
PUT LIST('Section 1.5.1.4.1e sets the requirements for lateral bracing of the
  compression flange.');
```

```

PUT LIST(' If this section is satisfied lateral torsional buckling will not
  occur. ');
GET LIST(dAf,bf);
t1=76*bf/sfy;
PUT LIST(' 76bf/Fy**.5 = ',t1,' in. ');
t1=20000/(dAf*Fy);
PUT LIST(' 20000/((d/Af)*Fy) = ',t1,' in. ');
PUT LIST('Is the bracing adequate? i.e. is L(in) < both of the above. ');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO 1b3;
t1=.66*Fy;
IF Fy=36 THEN t1=24;
IF skip=1 THEN GO TO 1b4;
PUT LIST('The allowable stress in tension and compression is equal to .66*Fy
  = ',t1);
GO TO 1b2;
1b4: PUT LIST('The allowable stress in tension and compression is determined by
  section 1.5.1.4.2 ');
IF tf=0 THEN GET LIST(tf);
t1=Fy*(.733-.0014*bf/(2*tf)*Fy**.5);
PUT LIST(' Fb=Fy(.733 - 0.0014(bf/2tf)Fy**.5) = ',t1);
GO TO 1b2;
1b3: IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
PUT LIST('The allowable tension stress is .6*Fy = ',t1,' as per section
  1.5.1.4.5. ');
GET LIST(L,rT);
IF dAf=0 THEN GET LIST(dAf);
t1=L*12/rT;
test=sqrt(102000/Fy);
IF t1<test THEN GO TO 1b7;
test1=sqrt(510000/Fy);
IF t1>=test1 THEN GO TO 1b8;
PUT LIST('L/rT is between (102000*Cb/Fy)**.5 = ',test,' and (510000*Cb/Fy)
  **.5 = ',test1);

```

```

PUT LIST('Therefore the allowable compression stress on the extreme fiber is
  the larger of');
t1=(2/3-Fy*(L*12/rT)**2/1530000)*Fy;
PUT LIST('  Fb = (2/3 - Fy(L/rT)**.5/1530000*Cb)Fy = ',t1);
PUT LIST('    and');
test=12000/(L*12*dAf);
PUT LIST('  Fb = 12000*Cb/Ld/Af = ',test);
IF Fy=36 THEN test1=22; ELSE test1=Fy*.6;
PUT LIST('    not to exceed Fb = .6*Fy = ',test1);
t1=max(t1,test);
t1=min(t1,test1);
PUT LIST('  use:  Fb = ',t1);
GO TO 1b2;
1b8: PUT LIST('L/rT = ',t1,' which is greater than (510000*Cb/Fy)**.5 = ',test1);
test1=12000/(L*12*dAf);
test=170000/(L*12/rT)**2;
PUT LIST('Therefore the allowable compression stress on the extreme fiber is
  the larger of');
PUT LIST('  Fb = 170000*Cb/(L/rT)**2 = ',test);
PUT LIST('    and');
PUT LIST('  Fb = 12000*Cb/Ld/Af = ',test1);
IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
PUT LIST('    not to exceed .6*Fy = ',t1);
test=max(test,test1);
t1=min(t1,test);
PUT LIST('  use:  Fb = ',t1);
GO TO 1b2;
1b7: PUT LIST('L/rT = ',t1,' which is less than (102000*Cb/Fy)**.5 = ',test);
IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
PUT LIST('Therefore the allowable stress on the extreme fiber is equal to
  .6*Fy = ',t1);
1b2: test=t1*S/12;
PUT LIST('The allowable bending moment is ',test,' ft. kips. ');
PUT LIST(' ');

```

```
      PUT LIST('Do you want to try another investigation problem');
      GET LIST(ANSWER);
      IF ANSWER=1 THEN GO TO 1b5;
      RETURN ;
pend:  PUT LIST(' ');
      PUT LIST('Because of a violation of Section 1.9 the member you have chosen');
      PUT LIST('    will not be 100% effective in resisting the load. The reduction
              in');
      PUT LIST('    section may be calculated as per Appendix C. This procedure has
              not');
      PUT LIST('    been included in BENTST. Therefore you have the option of
              calculating');
      PUT LIST('    this reduction by hand or of trying a new member. ');
      PUT LIST('Would you like to try a new member?');
      GET LIST(ANSWER);
      IF ANSWER=1 THEN GO TO 1b5;
      RETURN ;
      END BENTS1;
```

```

BENTS2: PROCEDURE ;
DECLARE hyp CHAR(1);
hyp='';
PUT LIST('Design is basically a trial and error procedure. The designer
  chooses a member based on');
PUT LIST('  his experience and checks to see if the section is adequate. If
  the section is not');
PUT LIST('  satisfactory he then chooses a new section based on the experience
  gained from the ');
PUT LIST('  previous member.');
```

```

PUT LIST(' ');
PUT LIST('The AISC Manual of Steel Construction helps expedite this process by
  listing in the section');
PUT LIST('  "Properties for Designing", two parameters which can be used to
  see if the member is compact.');
```

```

PUT LIST('  These are Fy',hyp,'and Fy".');
PUT LIST('    Fy',hyp,' - stress at which the member is no longer compact
  because of the flanges.');
```

```

PUT LIST('    Fy" - stress at which the member is no longer compact because
  of the web.');
```

```

PUT LIST('  A dash (-) indicates the member is compact to a yield stress of
  100 ksi. Fy" has been');
```

```

PUT LIST('  calculated using an axial stress of zero i.e. pure bending.');
```

```

PUT LIST('  Two other parameters which are of interest in design are Lc and
  Lu. These have to do ');
PUT LIST('  with lateral bracing.');
```

```

PUT LIST('    Lc - if the length of lateral bracing is less than Lc and the
  section is compact');
```

```

PUT LIST('    the allowable stress will be .66*Fy.');
```

```

PUT LIST('    Lu - if the length of lateral bracing is less than Lu the
  allowable stress may be');
```

```

PUT LIST('    taken as .6*Fy.');
```

```

PUT LIST('    NOTE: If the length between bracing is much greater than
  Lu the allowable stress');
```

```

PUT LIST('    will be much less than .6*Fy.');
```

```
RETURN ;  
END ;
```

```

BENTS3:  PROCEDURE ;
          DECLARE ttt CHAR(1);
          DECLARE t1 DEC(4);
          DECLARE fat DEC(4);
          DECLARE test DEC(4);
          DECLARE test1 DEC(4);
          DECLARE S DEC(4);
im1:     IMAGE;
          Choose a W member with a section modulus of -----.-- in**3 about the x axis
          (Ref. Page 2.7 thru 2.12).
im1a:    IMAGE;
          For your reference type the name of the member chosen.
im2:     IMAGE;
          Based on Fy' and Fy'' is the member chosen compact
im3:     IMAGE;
          What is the section modulus of the member chosen (in**3).
im4:     IMAGE;
          Actual stress = -----.-- ksi    allowable stress = -----.-- ksi.
im5:     IMAGE;
          Required section modulus based on the assumed stress = S = M*12/fa =
          -----.-- in**3.
start:   PUT LIST('What is the design moment (ft.kips).');
          GET LIST(M,Fy);
          sfy=sqrt(Fy);
          PUT LIST('Is the compression flange laterally braced over its total length?');
          GET LIST(ANSWER);
          brac=0;
          IF ANSWER=0 THEN GO TO 1b1;
          brac=1;
          /* brac=1 full lateral support.*/;
          IF Fy=36 THEN t1=24; ELSE t1=.66*Fy;
          PUT LIST('Since bracing is no problem and the final member chosen will probably
          be compact');
          PUT LIST('    a good choice of an allowable stress is .66*Fy = ',t1,' ksi');
          GO TO 1b2;

```



```

1b1:    IF Fy=36 THEN t1=22; ELSE t1=Fy*.6;
        PUT LIST('Since bracing may be a problem a good choice of an allowable stress
            is .6*Fy = ',t1,' ksi');
1b2:    PUT LIST('Choose a trial allowable stress. ');
        GET LIST(fa);
        S=M*12/fa;
        PUT IMAGE(S)(im5);
        PUT IMAGE(S)(im1);
agn:    PUT IMAGE(1)(im1a);
        bf,tf,dAf=0;
        GET EDIT(ttt)(A(1));
        PUT IMAGE(1)(im3);
        GET LIST(S);
        fat=M*12/S;
        PUT IMAGE(1)(im2);
        PUT LIST(' (web only = 3   flange only = 2   web & flange = 1   none of these
            = 0)? ');
        GET LIST(ANSWER);
        IF ANSWER=2 THEN GO TO 1b5; ELSE test1=ANSWER
        pcom=0;
        IF ANSWER=1 THEN GO TO 11;
        test=95/sfy;
        PUT LIST('Is bf/2tf < 95/Fy**.5 = ',test);
        GET LIST(ANSWER);
        pcom=1;
        IF ANSWER=0 THEN GO TO pend;
        IF test1=0 THEN GO TO 1b5;
11:    IF brac=1 THEN GO TO 1b3;
        GET LIST(bf,dAf);
        test=76*bf/sfy;
        test1=20000/(dAf*Fy);
        PUT LIST('Is the compression flange laterally braced closer than 76*bf/Fy**.5
            = ',test,' in. ');
        PUT LIST(' and 20000/(d/Af)*Fy = ',test1,' in. ');
        GET LIST(ANSWER);
        IF ANSWER=0 THEN GO TO 1b5;
1b3:    IF pcom=1 THEN GO TO 1b4;

```

```

IF Fy=36 THEN test=24; ELSE test=.66*Fy;
PUT LIST('Allowable stress determined by section 1.5.1.4.1 Fb = .66*Fy = ',
  test,' actual stress = ',fat);
t1=test;
IF fat>test THEN PUT LIST('SECTION TRIED IS NOT ADEQUATE!'); ELSE GO TO edd;
GO TO 12;
1b4: IF bf=0 THEN GET LIST(bf);
GET LIST(tf);
test1=Fy*(.733-.0014*bf/(2*tf)*sfy);
PUT LIST('Allowable stress determined by EQ. 1.5-1 Fb = ',test1,' actual
  stress = ',fat);
t1=test1;
IF fat>test1 THEN PUT LIST('SECTION TRIED IS NOT ADEQUATE!'); ELSE GO TO edd;
GO TO 12;
1b5: PUT LIST('Paragraph 1.5.1.4.6a will determine the allowable stress in
  compression. ');
PUT LIST('Input the following: ');
GET LIST(L,rT);
IF dAf=0 THEN GET LIST(dAf,bf);
t1=L*12/rT;
test=sqrt(102000/Fy);
IF t1<test THEN GO TO 1b7;
test1=sqrt(510000/Fy);
IF t1>=test1 THEN GO TO 1b8;
PUT LIST('L/rT is between (102000*Cb/Fy)**.5 = ',test,' and
  (510000*Cb/Fy)**.5 = ',test1);
PUT LIST('Therefore the allowable compression stress on the extreme fiber is
  the larger of ');
t1=(2/3-Fy*(L*12/rT)**2/1530000)*Fy;
PUT LIST(' Fb = (2/3 - Fy(L/rT)**2/1530000*Cb)Fy = ',t1);
PUT LIST(' and ');
test=12000/(L*12*dAf);
PUT LIST(' Fb = 12000*Cb/Ld/Af = ',test);
IF Fy=36 THEN test1=22; ELSE test1=Fy*.6;
PUT LIST(' not to exceed Fb = .6*Fy = ',test1,' ksi. ');

```

```

t1=max(t1,test);
t1=min(t1,test1);
PUT LIST('    use:  Fb = ',t1);
GO TO 1b9;
1b8: PUT LIST('L/rT = ',t1,' which is greater than (510000*Cb/Fy)**.5 = ', test1);
test1=12000/(L*12*dAf);
test=170000/(L*12/rT)**2;
PUT LIST('Therefore the allowable compression stress on the extreme fiber is
the larger of');
PUT LIST('    Fb = 170000*Cb/(L/rT)**2 = ',test);
PUT LIST('    and');
PUT LIST('    Fb = 12000*Cb/Ld/Af = ',test1);
IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
PUT LIST('    not to exceed .6*Fy = ',t1);
test=max(test,test1);
t1=min(t1,test);
PUT LIST('    use:  Fb = ',t1);
GO TO 1b9;
1b7: PUT LIST('L/rT = ',t1,' which is less than (102000*Cb/Fy)**.5 = ',test);
IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
PUT LIST('Therefore the allowable stress on the extreme fiber is equal to
.6*Fy = ',t1);
1b9: PUT IMAGE(fat,t1)(im4);
IF t1<fat THEN PUT LIST('SECTION NOT ADEQUATE'); ELSE GO TO edd;
GO TO 12;
edd: PUT LIST('Section is adequate  actual stress = ',fat,'    allowable stress
= ',t1);
12:  test=t1/fat;
PUT LIST('f-all/f-act = ',test);
PUT LIST('Do you want to try a new member?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b6;
PUT LIST('Would you like to try a new design problem?');
GET LIST(ANSWER);

```

```

IF ANSWER=1 THEN GO TO start;
RETURN ;
pend:  PUT LIST(' ');
       PUT LIST('Because of a violation of Section 1.9 the member that you have
              chosen');
       PUT LIST(' will not be 100% effective in resisting the load. The reduction
              in');
       PUT LIST(' section may be calculated as per Appendix C. This procedure has
              not');
       PUT LIST(' been included in BENTST. Therefore you have the option of
              calculating');
       PUT LIST(' this reduction by hand or of trying a new member. ');
       PUT LIST('Would you like to try a new member? ');
       GET LIST(ANSWER);
       IF ANSWER=1 THEN GO TO 1b2;
       RETURN ;
1b6:   PUT LIST('Do you know which member? ');
       GET LIST(ANSWER);
       IF ANSWER=0 THEN GO TO 1b2; ELSE GO TO agn;
       END BENTS3;

```

Program Name:

BENDSN

Purpose:

BENDSN will investigate and design beams using a minimum amount of input and giving just enough output to define the design route. It should be used only after the fundamentals of beam design are understood. It is ideally suited for the design of structures and parametric studies.

Limitations:

Only W members bent about the strong axis may be investigated or designed using BENDSN.

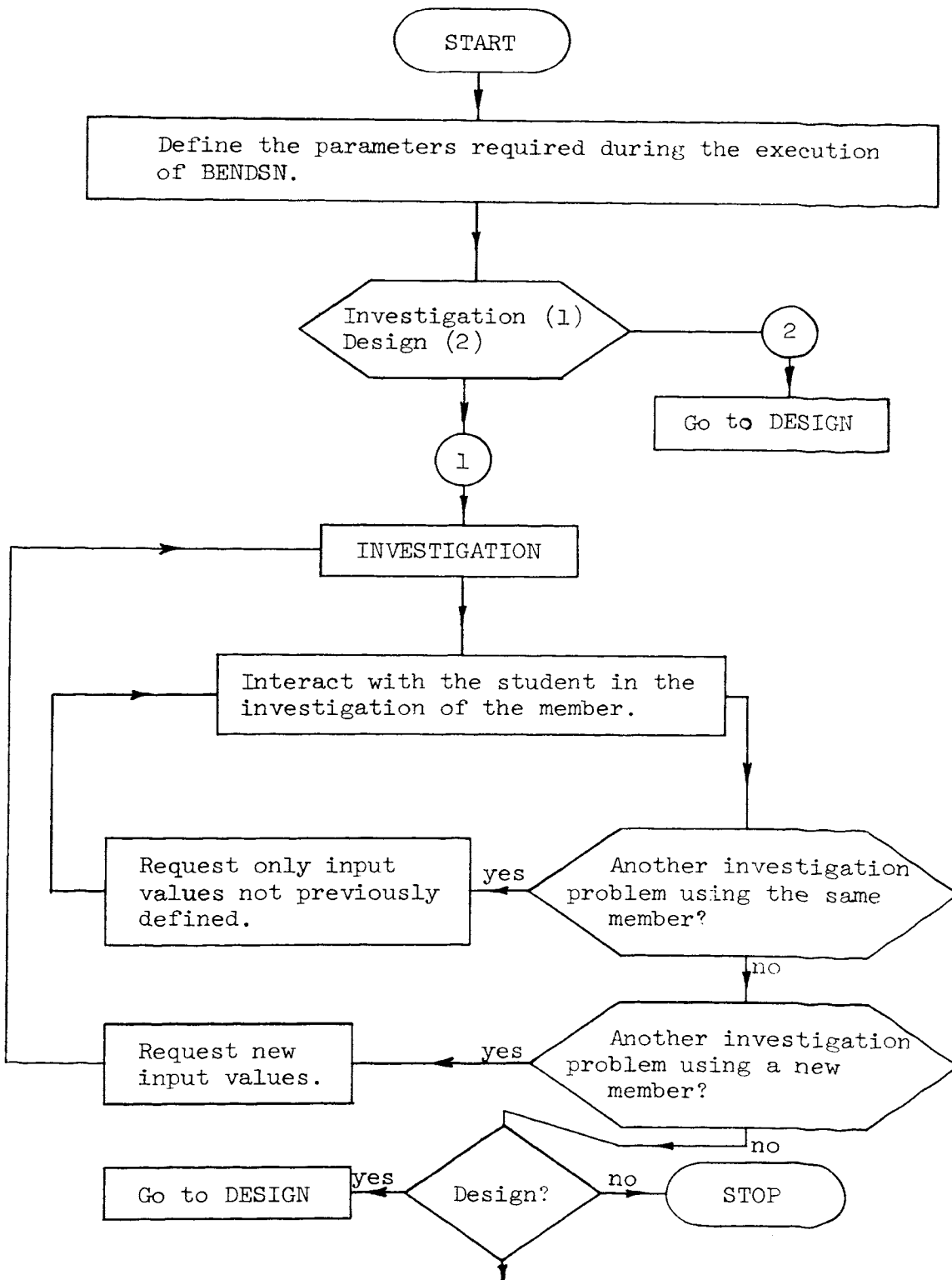
Prerequisite:

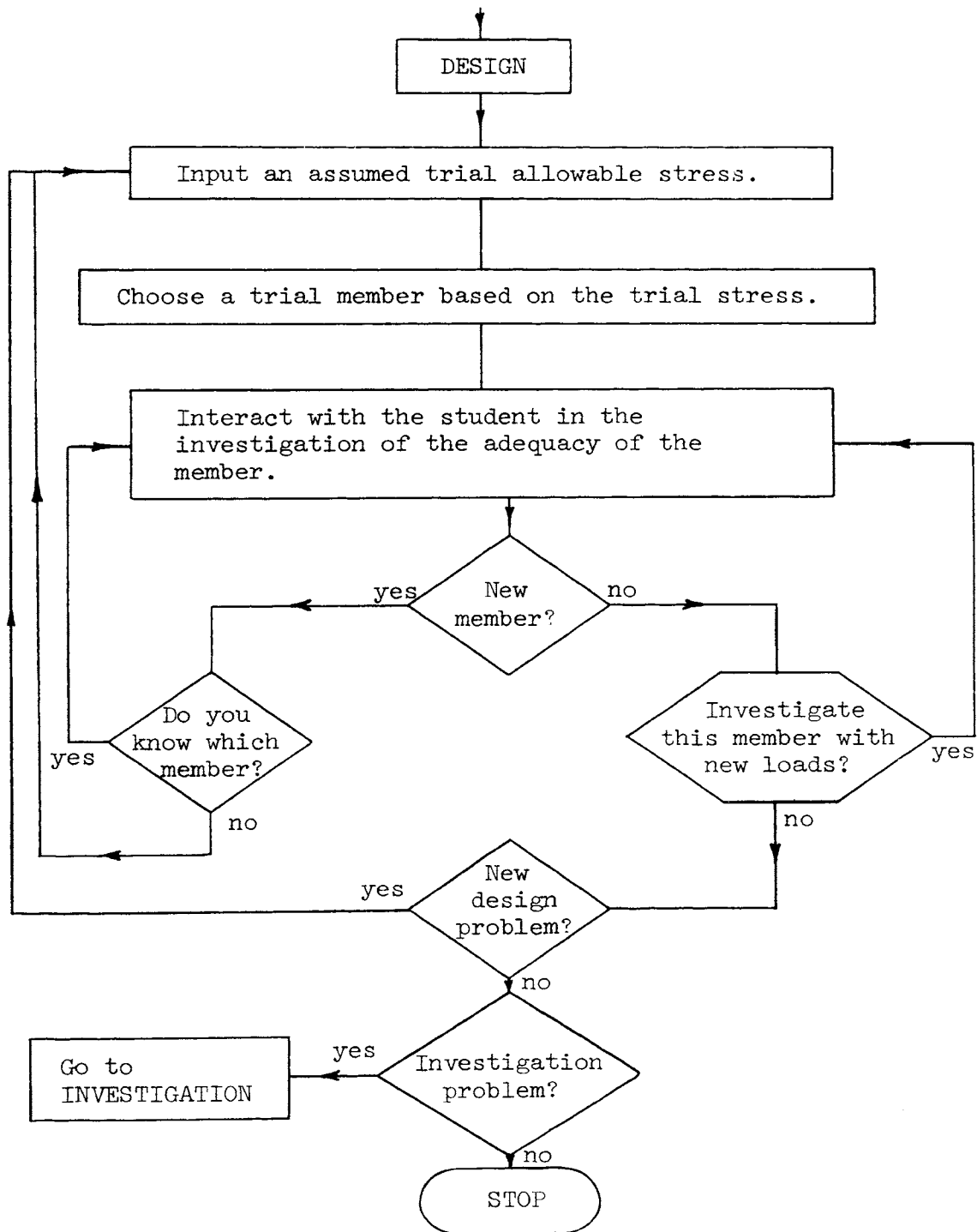
Execution of BENTST

Data Required Before Execution:

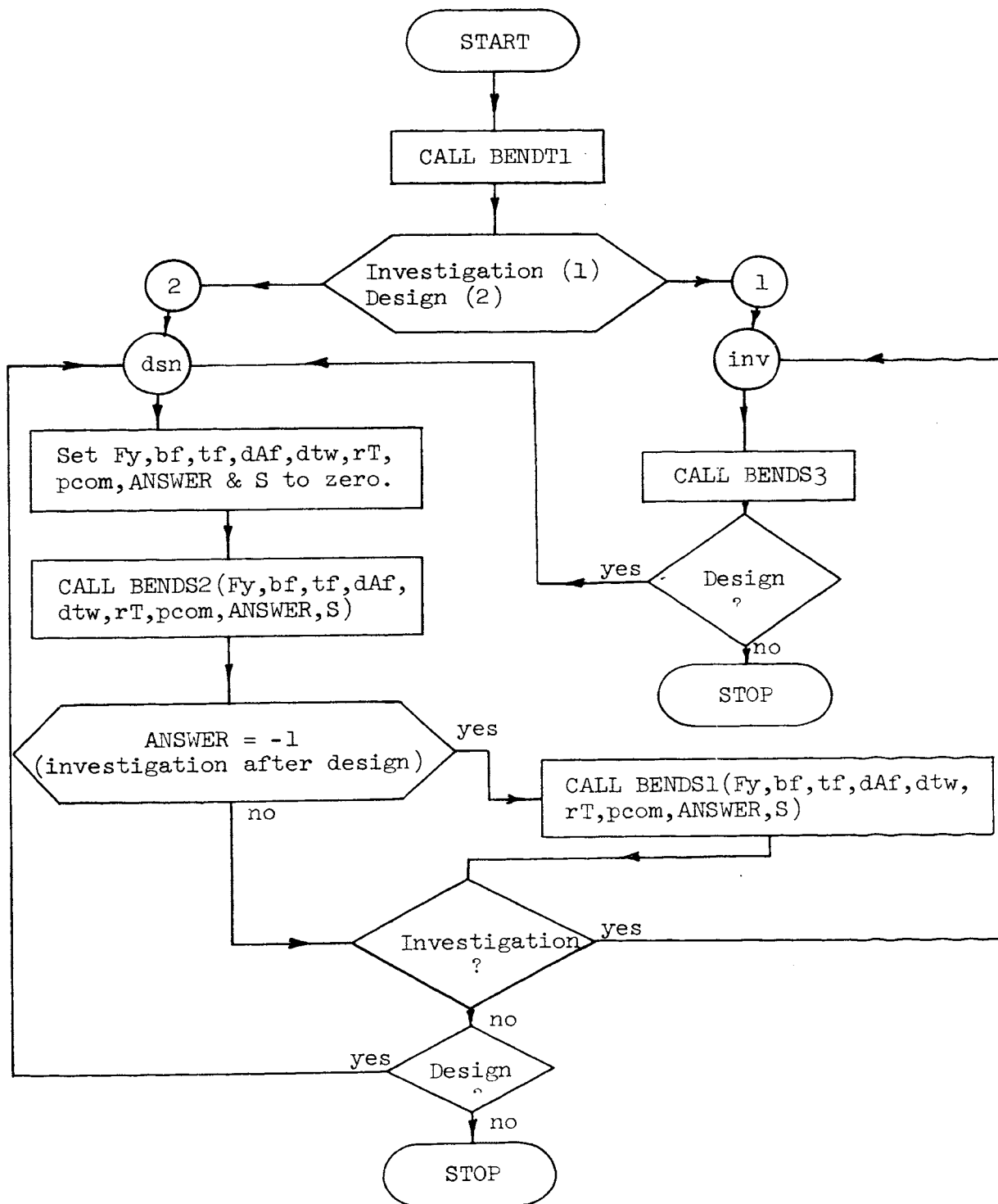
A bending design and/or investigation problem must be defined before execution of BENDSN. The length between bracing against twist (L) and the bending coefficient C_b must be defined.

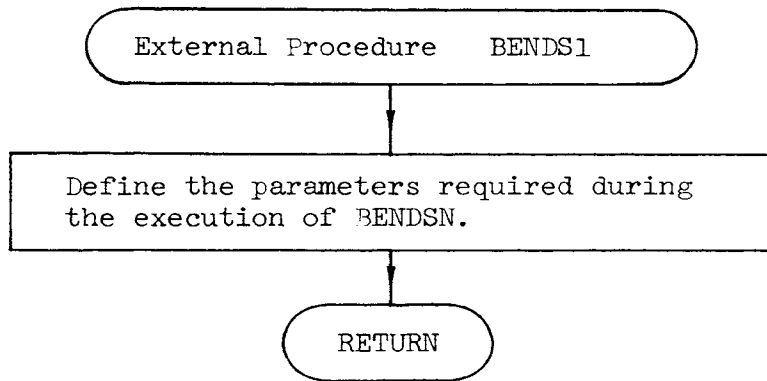
Overall Flow Chart BENDSN

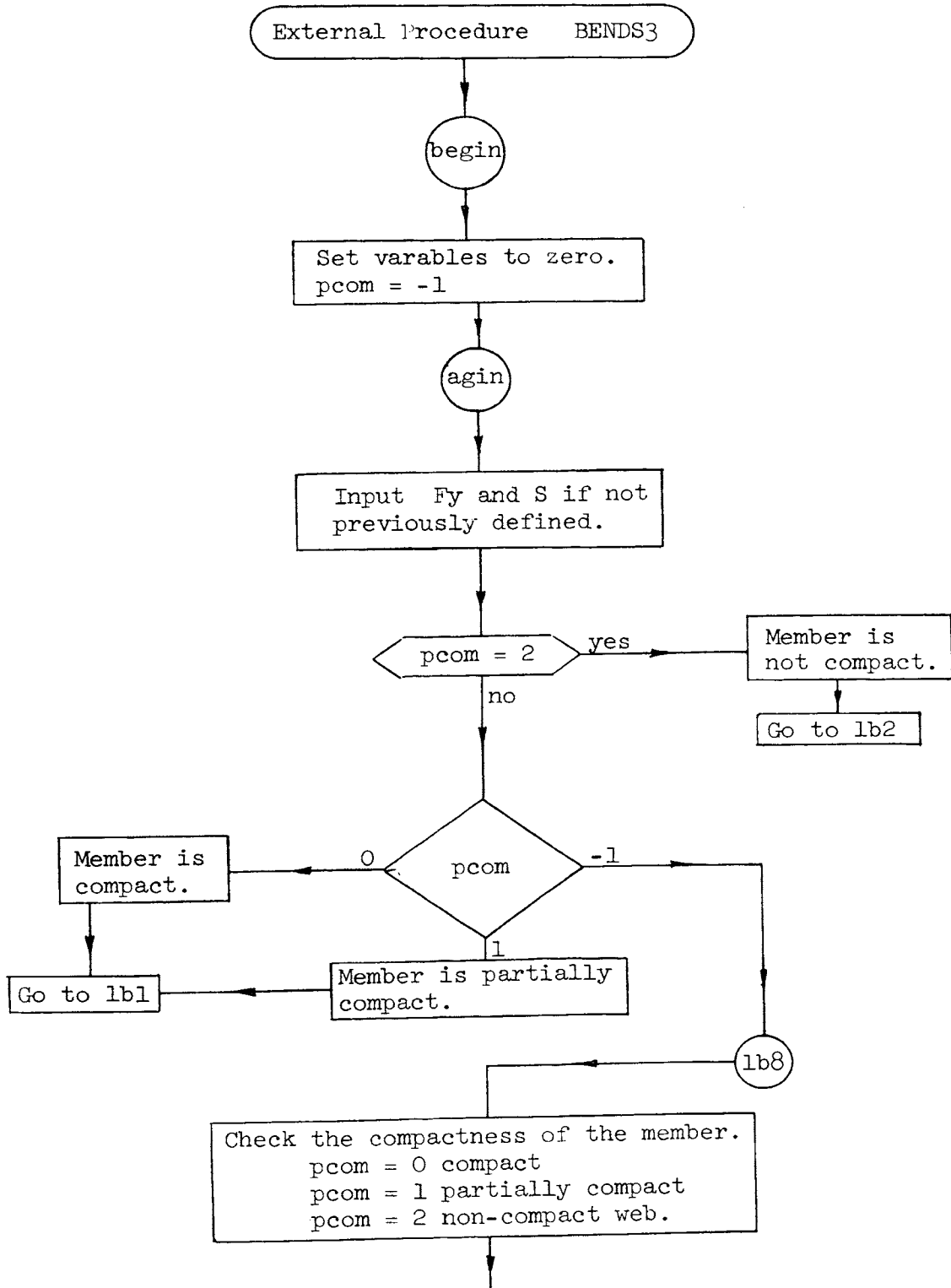


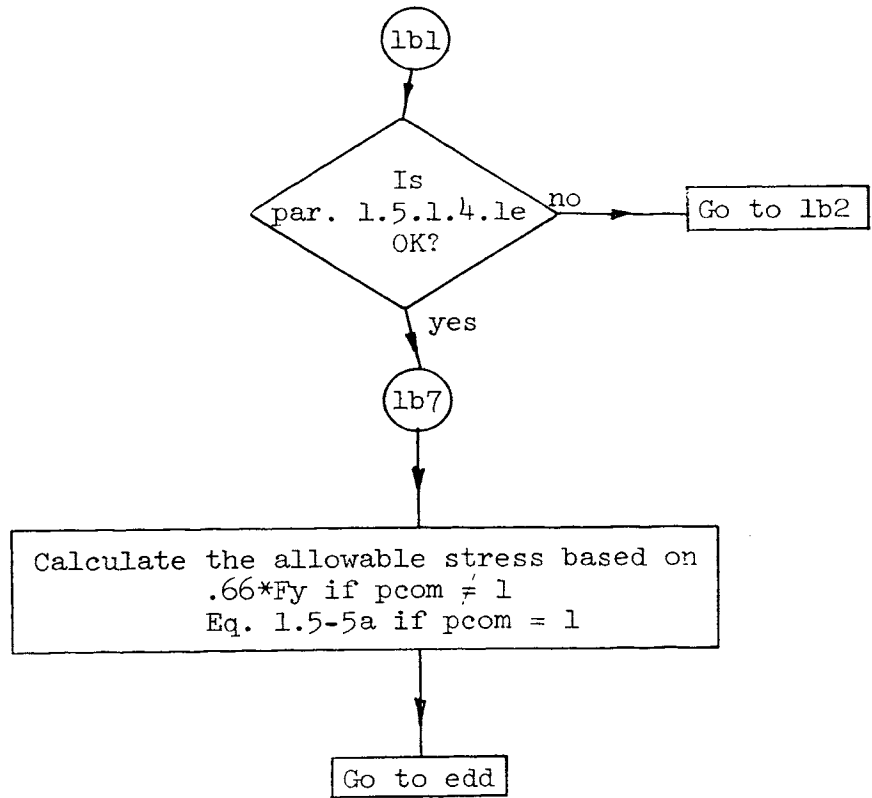


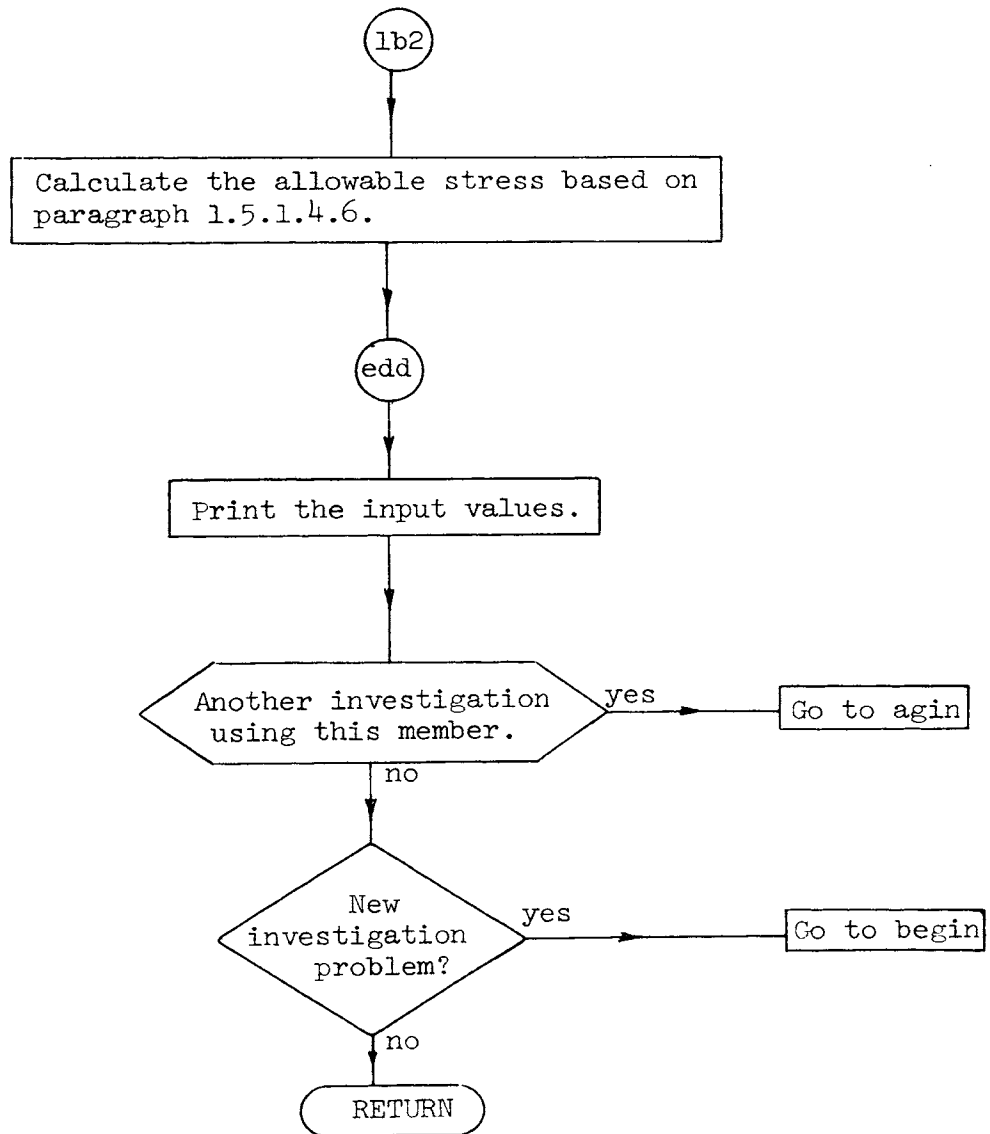
Detailed Flow Chart BENDSN

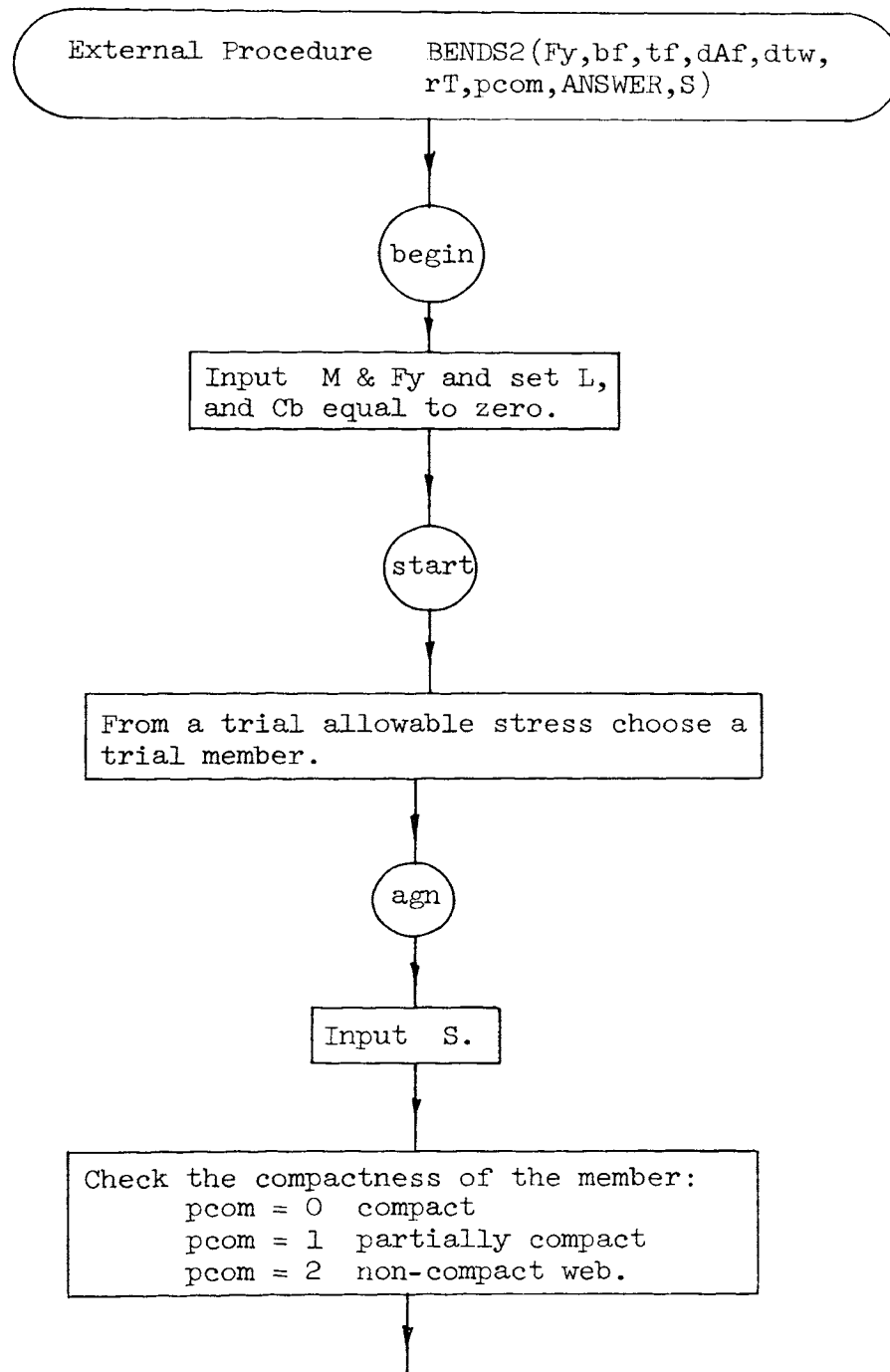


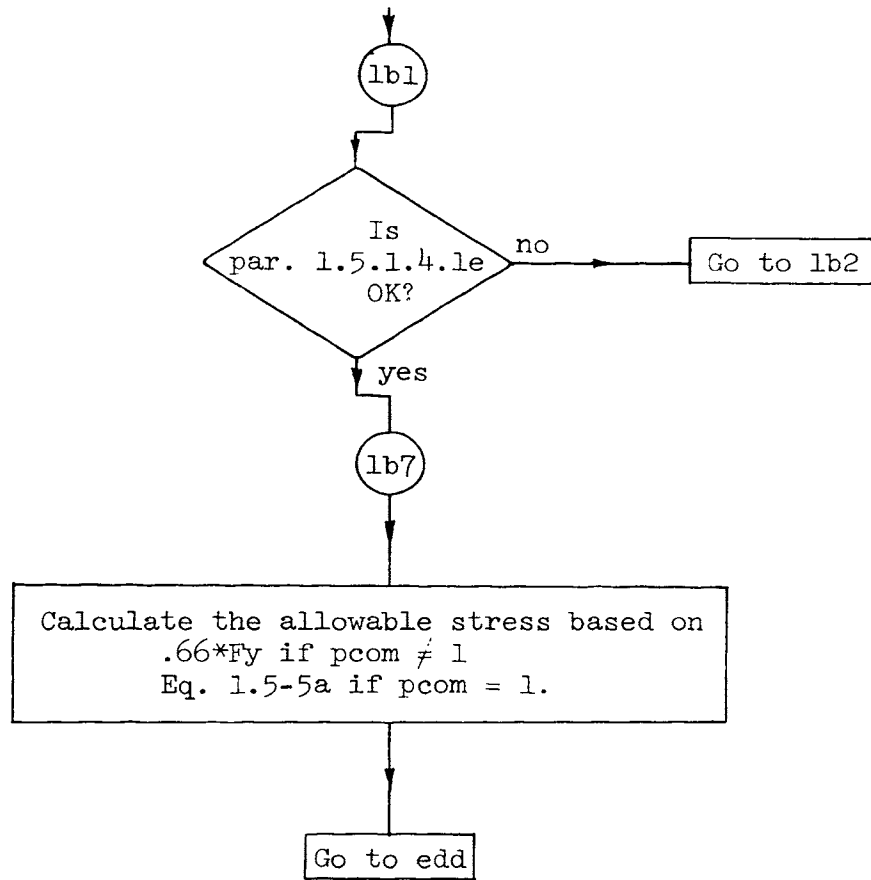


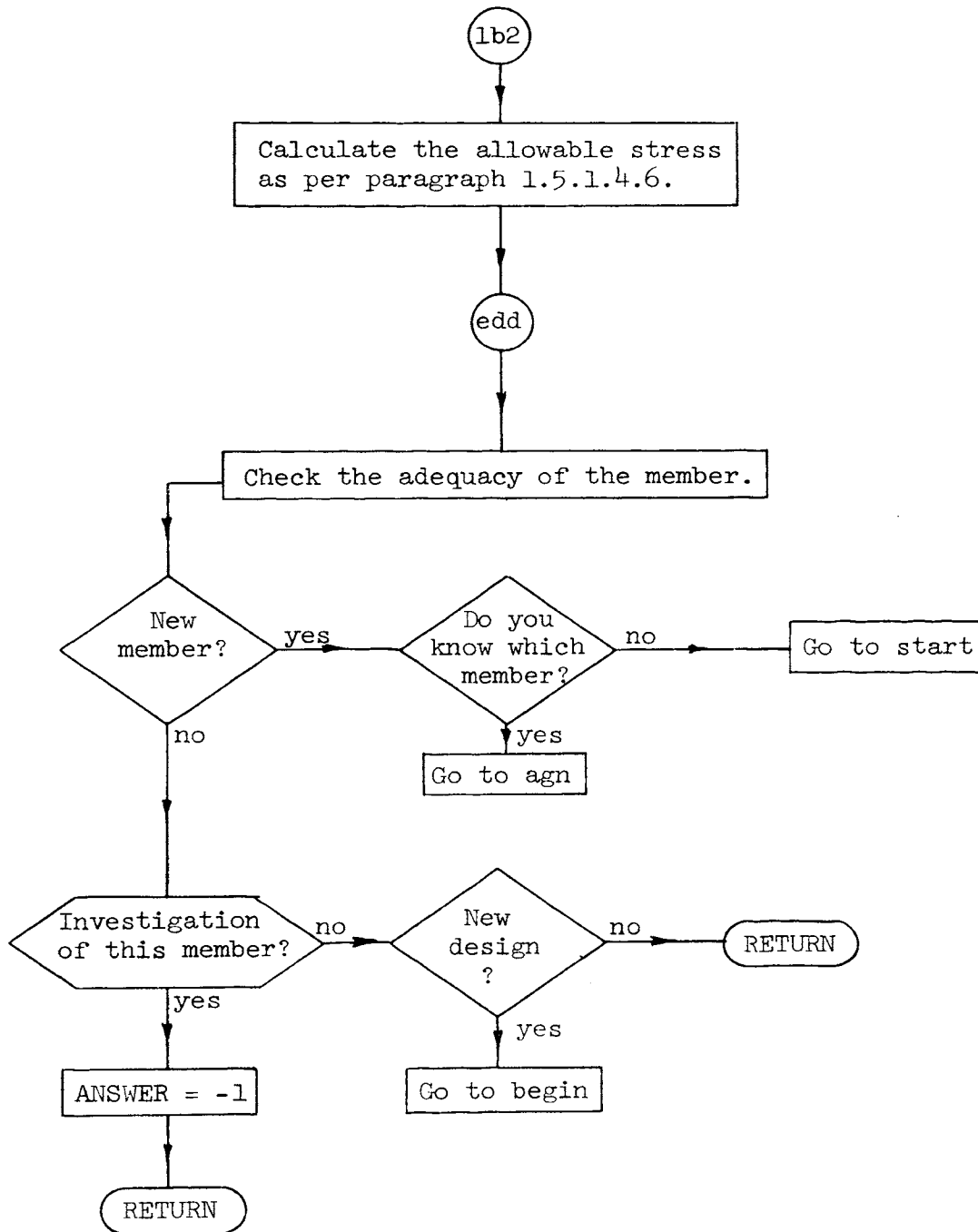


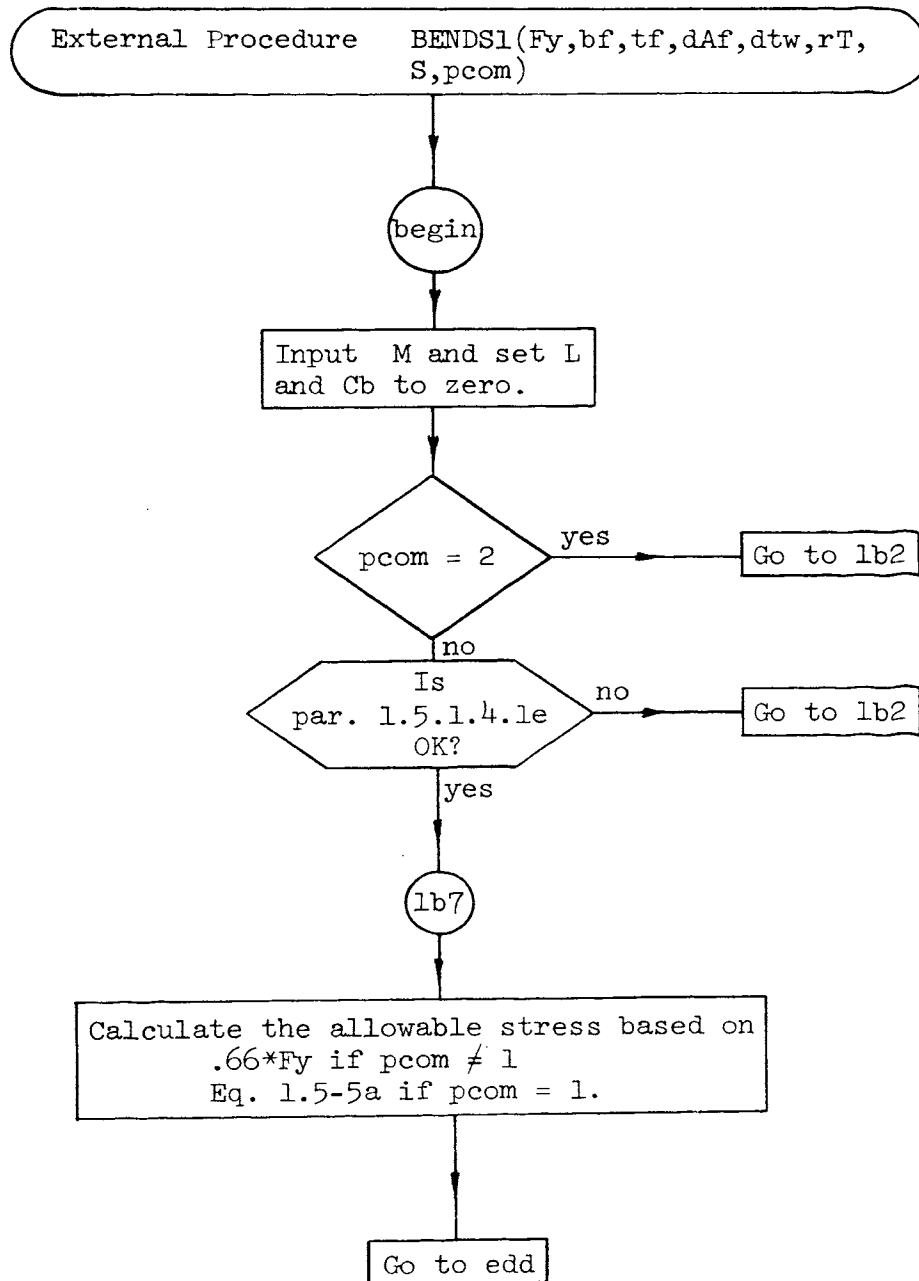


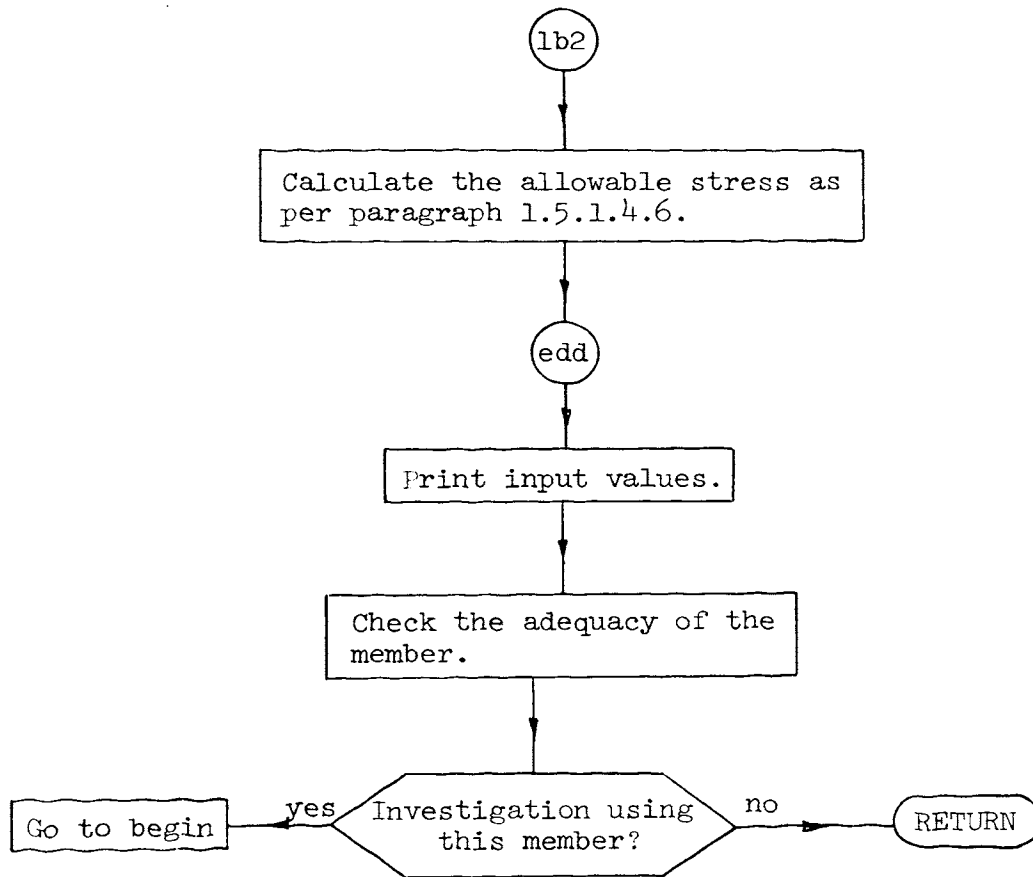












```

/*PROGRAM BENDSN */
DECLARE BENDT1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENDT2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENDS1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENDS2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BENDS3 ENTRY EXT KEY(whmIII) LIB(USER2);
CALL BENDT1;
PUT LIST('Type of problem (investigation = 1 Design = 2)');
GET LIST(ANSWER);
IF ANSWER=2 THEN GO TO dsn;
inv: CALL BENDS3;
PUT LIST('Would you like to try a design problem?');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO ed;
dsn: Fy,bf,tf,dAf,dtw,rT,pcom,ANSWER,S=0;
CALL BENDS2(Fy,bf,tf,dAf,dtw,rT,pcom,ANSWER,S);
IF ANSWER=-1 THEN CALL BENDS1(Fy,bf,tf,dAf,dtw,rT,S,pcom);
PUT LIST('Would you like to try an investigation problem?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO inv;
PUT LIST('Would you like to try a design problem?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO dsn;
ed: END ;

```

```

BENDT1: PROCEDURE ;
PUT LIST('The following are parameters which are required in the execution
of BENDSN:');
PUT LIST('  Fy - yield stress of the steel (ksi).');
PUT LIST('  bf - width of the flange (in.).');
PUT LIST('  tf - thickness of the flange (in.).');
PUT LIST('  dAf - d/Af or the depth of the member divided by the area of the
flange (1/in.).');
PUT LIST('  dtw - d/tw or the depth of the member divided by the thickness of
the flange. ');
PUT LIST('  rT - radius of gyration of a section comprising the compression
flange plus ');
PUT LIST('          1/3 the compression web area taken about an axis in the
plane of the web (in.).');
PUT LIST('  L - distance between bracing of the compression flange (ft.).');
PUT LIST('  S - section modulus of the member (in**3).');
PUT LIST('  Cb -  $C_b = 1.75 + 1.05(M1/M2) * 0.3(M1/M2)**2$  but  $< 2.3$  where M1
and M2');
PUT LIST('          are the smaller and larger moment at the ends of the unbraced
length. ');
PUT LIST('          If M1 and M2 are exceeded in this length  $C_b = 1.$  ');
PUT LIST('  ANSWER - usually requiring a yes or no indication by 1 = yes or
0 = no. ');
PUT LIST(' ');
PUT LIST('  IDENTIFICATION - Type any information which will help you in
identification of the output. ');
RETURN ;
END BENDT1;

```

```

BENDS1:  PROCEDURE (Fy,bf,tf,dAf,dT,rT,S,pcom);
          DECLARE ttt CHAR(1);
          DECLARE t1 DEC(4);
          DECLARE t2 DEC(4);
          DECLARE fat DEC(4);
          DECLARE fa DEC(4);
begin:    PUT LIST(' ');
          PUT LIST(' ');
          PUT LIST(' ');
          PUT LIST(' ');
im1:      IMAGE;
          * * * * * IDENTIFICATION * * * * *
          PUT IMAGE(1)(im1);
          GET EDIT(ttt)(A(1));
          PUT LIST('Design Moment (ft.kips).');
          L,Cb=0;
          GET LIST(M);
          PUT LIST('S = ',S);
          PUT LIST('Fy = ',Fy);
          sfy=sqrt(Fy);
          fat=M*12/S;
          IF pcom=2 THEN GO TO 1b2;
          /* pcom = 1 if partially compact. pcom = 2 if non compact. */;
1b1:      PUT LIST('Is the compression flange fully laterally supported?');
          GET LIST(ANSWER);
          IF ANSWER=1 THEN GO TO 1b7:
          IF bf=0 THEN GET LIST(bf); ELSE PUT LIST('bf = ',bf);
          IF dAf=0 THEN GET LIST(dAf); ELSE PUT LIST('dAf = ',dAf);
          t1=76*bf/sfy;
          t2=20000/(dAf*Fy);
          t1=min(t1,t2);
          IF L=0 THEN GET LIST(L); ELSE PUT LIST('L = ',L);
          IF L*12>t1 THEN GO TO 1b2;
          /* par 1.5.1.4.1 or 1.5.1.4.2 */;

```

```

1b7:    IF pcom=1 THEN GO TO 1b3;
        PUT LIST('      Paragraph 1.5.1.4.1 determines the allowable stress.');
```

$$fa = .66 * Fy;$$

```

        IF Fy=36 THEN fa=24;
        GO TO edd;
1b3:    IF bf=0 THEN GET LIST(bf);
        IF tf=0 THEN GET LIST(tf);
        PUT LIST('      Equation 1.5-5 determines the allowable stress.');
```

$$fa = Fy * (.733 - .0014 * bf / (2 * tf) * sfy);$$

```

        GO TO edd;
1b2:    PUT LIST('INPUT:');
        IF L=0 THEN GET LIST(L);
        IF rT=0 THEN GET LIST(rT); ELSE PUT LIST('rT = ',rT);
        t1=L*12/rT;
        GET LIST(Cb);
        IF t1 < sqrt(102000 * Cb / Fy) THEN GO TO 1b4;
        IF dAf=0 THEN GET LIST(dAf);
        t2=12000 * Cb / (L * 12 * dAf);
        IF t1 > sqrt(510000 * Cb / Fy) THEN GO TO 1b5;
        t1=(2/3 - Fy * t1**2 / (1530000 * Cb)) * Fy;
        GO TO 1b6;
1b5:    t1=170000 * Cb / t1**2;
1b6:    t1=max(t1,t2);
        IF Fy=36 THEN t2=22; ELSE t2=.6 * Fy;
        fa=min(t1,t2);
        PUT LIST('      Paragraph 1.5.1.4.6a determines the allowable stress.');
```

$$GO TO edd;$$

```

1b4:    IF Fy=36 THEN fa=22; ELSE fa=.6 * Fy;
        PUT LIST('      Paragraph 1.5.1.4.6b determines the allowable stress.');
```

$$edd: PUT LIST('Allowable stress = ',fa, ' Actual stress = ',fat);$$

```

        t1=fa/fat;
        PUT LIST('      f-all/f-act = ',t1);
        IF bf>0 THEN PUT LIST('bf = ',bf);
        IF tf>0 THEN PUT LIST('tf = ',tf);
```

```
IF dAf>0 THEN PUT LIST('dAf = ',dAf);
IF rT>0 THEN PUT LIST('rT = ',rT);
IF dtw>0 THEN PUT LIST('dtw = ',dtw);
PUT LIST('Would you like to try another investigation problem using this
member?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO begin;
RETURN ;
END BENDS1;
```

```

BENDS2: PROCEDURE (Fy,bf,tf,dAf,dtw,rT,pcom,ANSWER,S);
DECLARE ttt CHAR(1);
DECLARE t1 DEC(4);
DECLARE t2 DEC(4);
DECLARE fat DEC(4);
DECLARE fa DEC(4);
begin: PUT LIST(' ');
        PUT LIST(' ');
        PUT LIST(' ');
        PUT LIST(' ');
im1:    IMAGE;
        * * * * * IDENTIFICATION * * * * *
        PUT IMAGE(1)(im1);
        GET EDIT(ttt)(A(1));
        PUT LIST('Design Moment (ft.kips).');
        L,Cb=0;
        GET LIST(M);
        PUT LIST('Yield stress (KSI).');
        GET LIST(Fy);
        sfy=sqrt(Fy);
start: PUT LIST('Assume an allowable stress (ksi).');
        GET LIST(fa);
        S=M*12/fa;
        PUT LIST('Section modulus based on the assumed stress = S = ',S,' in**3. ');
        PUT LIST('Try a W section with approximately this section modulus. ');
agn:   PUT IMAGE(1)(im1);
        GET EDIT(ttt)(A(1));
        PUT LIST('What is the section modulus of the member (in**3). ');
        bf,tf,dAf,dtw,rT,S=0;
        GET LIST(S);
        fat=M*12/S;
        PUT LIST('Is the member compact? ');
        pcom=0;
        GET LIST(ANSWER);
        IF ANSWER=1 THEN GO TO 1b1; ELSE pcom=2;

```

```

/* pcom = 2 for non compact member. */;
PUT LIST('INPUT:');
GET LIST(bf,tf);
IF bf/(2*tf)>95/sfy THEN GO TO pend;
t1=412/sfy;
GET LIST(dtw);
IF dtw>t1 THEN GO TO 1b2;
pcom=1;
/* pcom = 1 if partially compact. */;
1b1: IF L>0 THEN GO TO 1b8;
PUT LIST('Is the compression flange fully laterally supported?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b7;
1b8: IF bf=0 THEN GET LIST(bf);
GET LIST(dAf);
t1=76*bf/sfy;
t2=20000/(dAf*Fy);
t1=min(t1,t2);
IF L=0 THEN GET LIST(L); ELSE PUT LIST('L = ',L);
IF L*12>t1 THEN GO TO 1b2;
/* par 1.5.1.4.1 or 1.5.1.4.2 */;
1b7: IF pcom=1 THEN GO TO 1b3;
PUT LIST(' Paragraph 1.5.1.4.1 determines the allowable stress. ');
fa=.66*Fy;
IF Fy=36 THEN fa=24;
GO TO edd;
1b3: IF bf=0 THEN GET LIST(bf);
IF tf=0 THEN GET LIST(tf);
PUT LIST(' Equation 1.5-5 determines the allowable stress. ');
fa=Fy*(.733-.0014*bf/(2*tf)*sfy);
GO TO edd;
1b2: PUT LIST('INPUT:');
IF L=0 THEN GET LIST(L);
GET LIST(rT);

```



```

t1=L*12/rT;
IF Cb=0 THEN GET LIST(Cb); ELSE PUT LIST('Cb = ',Cb);
IF t1<sqrt(102000*Cb/Fy) THEN GO TO 1b4;
IF dAf=0 THEN GET LIST(dAf);
t2=12000*Cb/(L*12*dAf);
IF t1>sqrt(510000*Cb/Fy) THEN GO TO 1b5;
t1=(2/3-Fy*t1**2/(1530000*Cb))*Fy;
GO TO 1b6;
1b5: t1=170000*Cb/t1**2;
1b6: t1=max(t1,t2);
IF Fy=36 THEN t2=22; ELSE t2=.6*Fy;
fa=min(t1,t2);
PUT LIST(' Paragraph 1.5.1.4.6a determines the allowable stress.');
```

GO TO edd;

```

1b4: IF Fy=36 THEN fa=22; ELSE fa=.6*Fy;
PUT LIST(' Paragraph 1.5.1.4.6b determines the allowable stress.');
```

GO TO edd;

```

edd: PUT LIST('Allowable stress = ',fa,' Actual stress = ',fat);
t1=fa/fat;
PUT LIST(' f=all/f-act = ',t1);
PUT LIST('Would you like to try a new member?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b10;
PUT LIST('Would you like to try an investigation of this member with a new
loading?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b9;
PUT LIST('Would you like to try a new design problem?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO begin;
RETURN ;
1b9: ANSWER=-1;
RETURN ;
```

```

pend:  PUT LIST(' ');
        PUT LIST('Because of a violation of Section 1.9 the member you have chosen');
        PUT LIST('    will not be 100% effective in resisting the load. The reduction
in');
        PUT LIST('    section may be calculated as per Appendix C. This procedure has
not');
        PUT LIST('    been included in BENDSN. Therefore you have the option of
calculating');
        PUT LIST('    this reduction by hand or of trying a new member. ');
        PUT LIST('Would you like to try a new member?');
        GET LIST(ANSWER);
        IF ANSWER=1 THEN GO TO start;
        RETURN ;
1b10:  PUT LIST('Do you know which member?');
        GET LIST(ANSWER);
        IF ANSWER=0 THEN GO TO start;
        GO TO agn;
        END BENDS2;

```

```

BENDS3:  PROCEDURE ;
          DECLARE ttt CHAR(1);
          DECLARE t1 DEC(4);
          DECLARE t2 DEC(4);
          DECLARE S DEC(4);
          DECLARE fa DEC(4);
begin:    PUT LIST(' ');
          bf,tf,dtw,Fy,rT,dAf,S=0;
          pcom=-1;
agin:    PUT LIST(' ');
im1:     IMAGE;
          * * * * * IDENTIFICATION * * * * *
          PUT IMAGE(1)(im1);
          GET EDIT(ttt)(A(1));
          L,Cb=0;
          PUT LIST('Yield stress (ksi).');
          IF Fy=0 THEN GET LIST(Fy); ELSE PUT LIST('Fy = ',Fy);
          sfy=sqrt(Fy);
          PUT LIST('Section Modulus S (in**3)');
          IF S=0 THEN GET LIST(S); ELSE PUT LIST('S = ',S);
          IF pcom=2 THEN PUT LIST('Member is not compact'); ELSE GO TO 1b9;
          GO TO 1b2;
1b9:     IF pcom=-1 THEN GO TO 1b8;
          IF pcom=0 THEN PUT LIST('Member is compact. ');
          IF pcom=1 THEN PUT LIST('Member is partially compact. ');
          GO TO 1b1;
1b8:     PUT LIST('Is the member compact?');
          pcom=0;
          GET LIST(ANSWER);
          IF ANSWER=1 THEN GO TO 1b1;
          pcom=2;
          PUT LIST('INPUT: ');
          GET LIST(bf,tf);
          IF bf/(2*tf)>95/sfy THEN GO TO pend;
          t1=412/sfy;

```

```

GET LIST(dtw);
IF dtw>t1 THEN GO TO 1b2;
pcom=1;
/* pcom = 1 if partially compact. */;
1b1: PUT LIST('Is the compression flange fully laterally supported?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO 1b7;
IF bf=0 THEN GET LIST(bf);
IF dAf=0 THEN GET LIST(dAf);
t1=76*bf/sfy;
t2=20000/(dAf*Fy);
t1=min(t1,t2);
IF L=0 THEN GET LIST(L); ELSE PUT LIST('L = ',L);
IF L*12>t1 THEN GO TO 1b2;
/* par 1.5.1.4.1 or 1.5.1.4.2 */;
1b7: IF pcom=1 THEN GO TO 1b3;
PUT LIST(' Paragraph 1.5.1.4.1 determines the allowable stress. ');
fa=.66*Fy;
IF Fy=36 THEN fa=24;
GO TO edd;
1b3: IF bf=0 THEN GET LIST(bf);
IF tf=0 THEN GET LIST(tf);
PUT LIST(' Equation 1.5-5 determines the allowable stress. ');
fa=Fy*(.733-.0014*bf/(2*tf)*sfy);
GO TO edd;
1b2: PUT LIST('INPUT: ');
IF L=0 THEN GET LIST(L);
IF rT=0 THEN GET LIST(rT);
t1=L*12/rT;
GET LIST(Cb);
IF t1<sqrt(102000*Cb/Fy) THEN GO TO 1b4;
IF dAf=0 THEN GET LIST(dAf);
t2=12000*Cb/(L*12*dAf);
IF t1>sqrt(510000*Cb/Fy) THEN GO TO 1b5;
t1=(2/3-Fy*t1**2/(1530000*Cb))*Fy;
GO TO 1b6;

```

```

1b5:      t1=170000*Cb/t1**2;
1b6:      t1=max(t1,t2);
          IF Fy=36 THEN t2=22; ELSE t2=.6*Fy;
          fa=min(t1,t2);
          PUT LIST('      Paragraph 1.5.1.4.6a determines the allowable stress. ');
          GO TO edd;
1b4:      IF Fy=36 THEN fa=22; ELSE fa=.6*Fy;
          PUT LIST('      Paragraph 1.5.1.4.6b determines the allowable stress. ');
          GO TO edd;
edd:      PUT LIST('Allowable stress = ',fa);
          t1=fa*S/12;
          IF bf>0 THEN PUT LIST(bf);
          IF tf>0 THEN PUT LIST(tf);
          IF dtw>0 THEN PUT LIST(dtw);
          IF dAf>0 THEN PUT LIST(dAf);
          IF rT>0 THEN PUT LIST(rT);
          PUT LIST('Allowable bending moment = ',t1,' ft.kips. ');
          PUT LIST('Would you like to try another investigation problem using this
            member? ');
          GET LIST(ANSWER);
          IF ANSWER=1 THEN GO TO agin;
          PUT LIST('Would you like to try another investigation problem? ');
          GET LIST(ANSWER);
          IF ANSWER=1 THEN GO TO begin;
          RETURN ;
pend:     PUT LIST(' ');
          PUT LIST('Because of a violation of Section 1.9 the member that you have
            chosen ');
          PUT LIST(' will not be 100% effective in resisting the load. The reduction
            in ');
          PUT LIST(' section may be calculated as per Appendix C. This procedure has
            not ');
          PUT LIST(' been included in BENDSN. Therefore you have the option of
            calculating ');
          PUT LIST(' this reduction by hand or of trying a new member. ');

```

```
PUT LIST('Would you like to try a new member?');  
GET LIST(ANSWER);  
IF ANSWER=1 THEN GO TO begin;  
RETURN ;  
END BENDS3;
```

Program Name:

BIXDSN

Purpose:

Program BIXDSN will interact with the user in the design of beam-columns. It may be used to design members subjected to any combination of bending about the strong axis, bending about the weak axis and compression. The use of BIXDSN is intended to give experience in design since the details of calculations of each part of BIXDSN has been covered in previous programs.

Limitations:

Only W sections may be designed using BIXDSN.

Prerequisite:

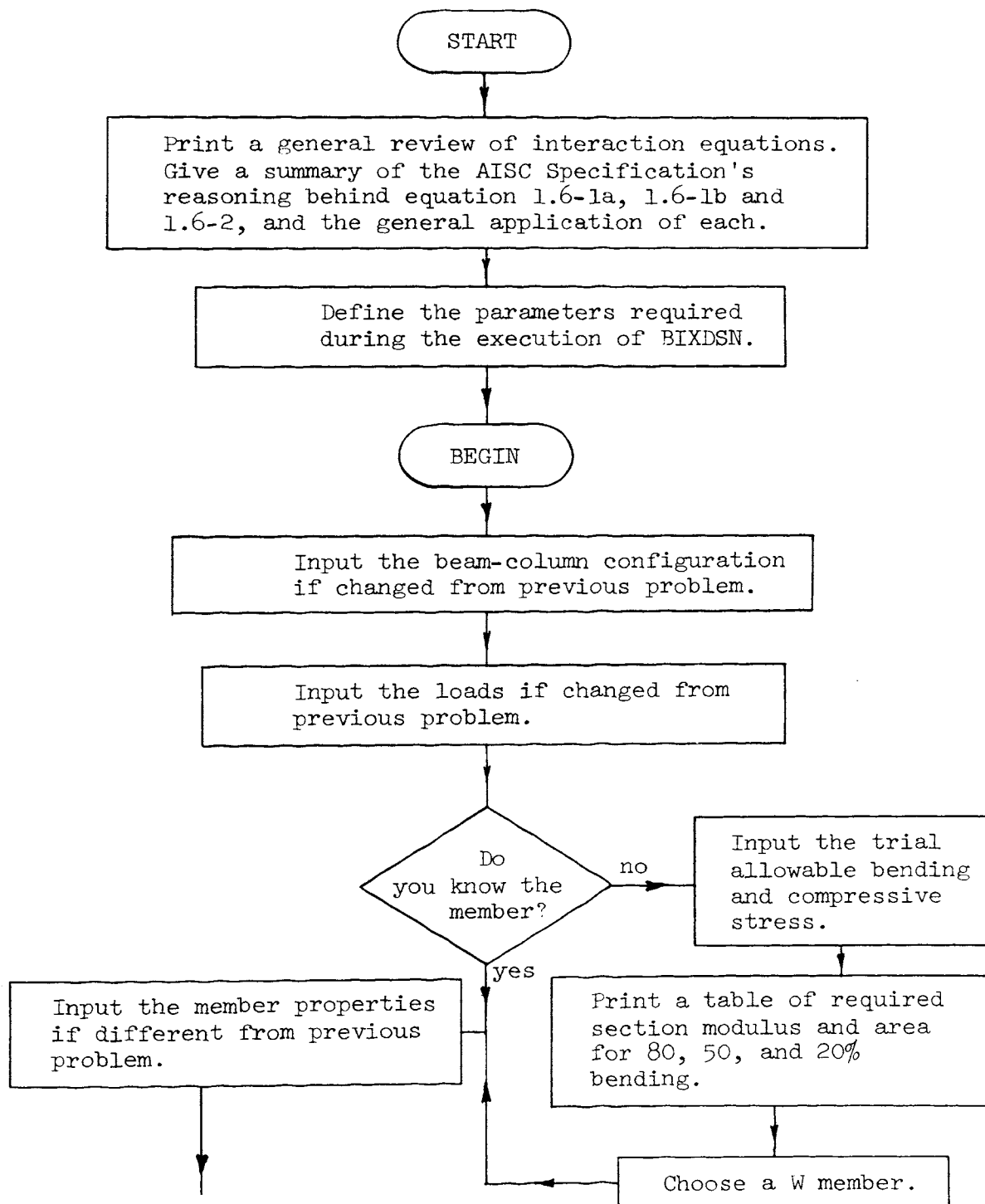
Execution of BENDSN and COLTST. Read Section 1.6 of the AISC Specifications. The student must have knowledge of bending about the weak axis.

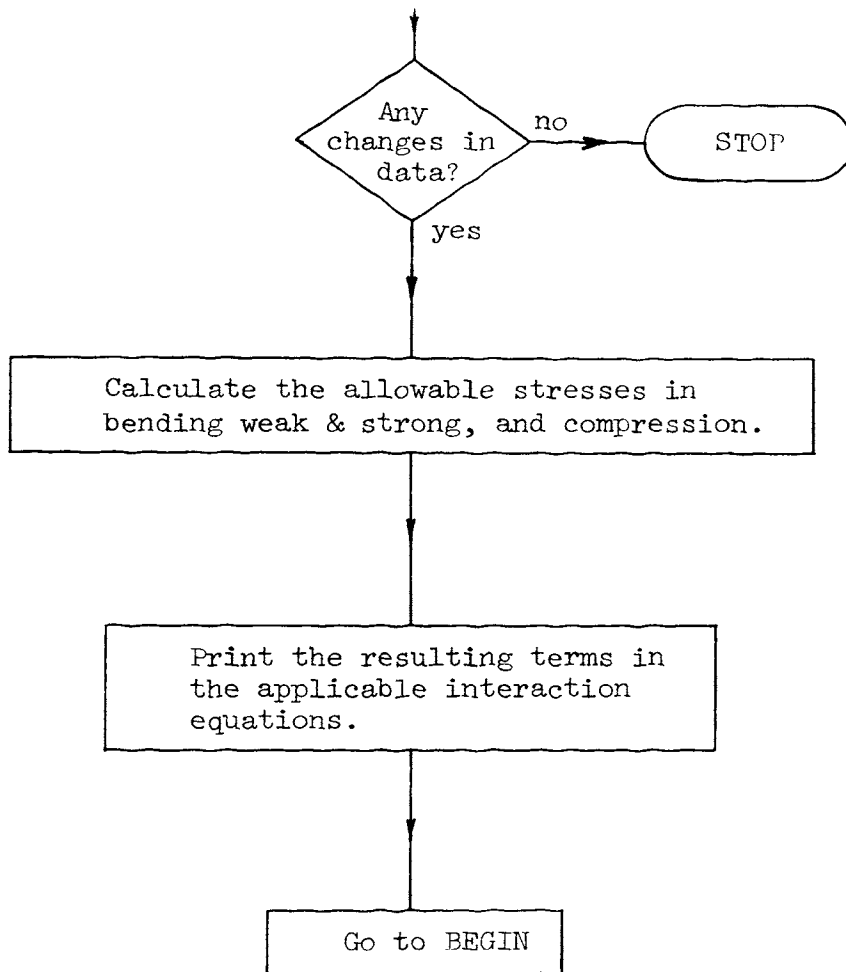
Data Required Before Execution:

A design and/or investigation problem must be defined before execution of BIXDSN.

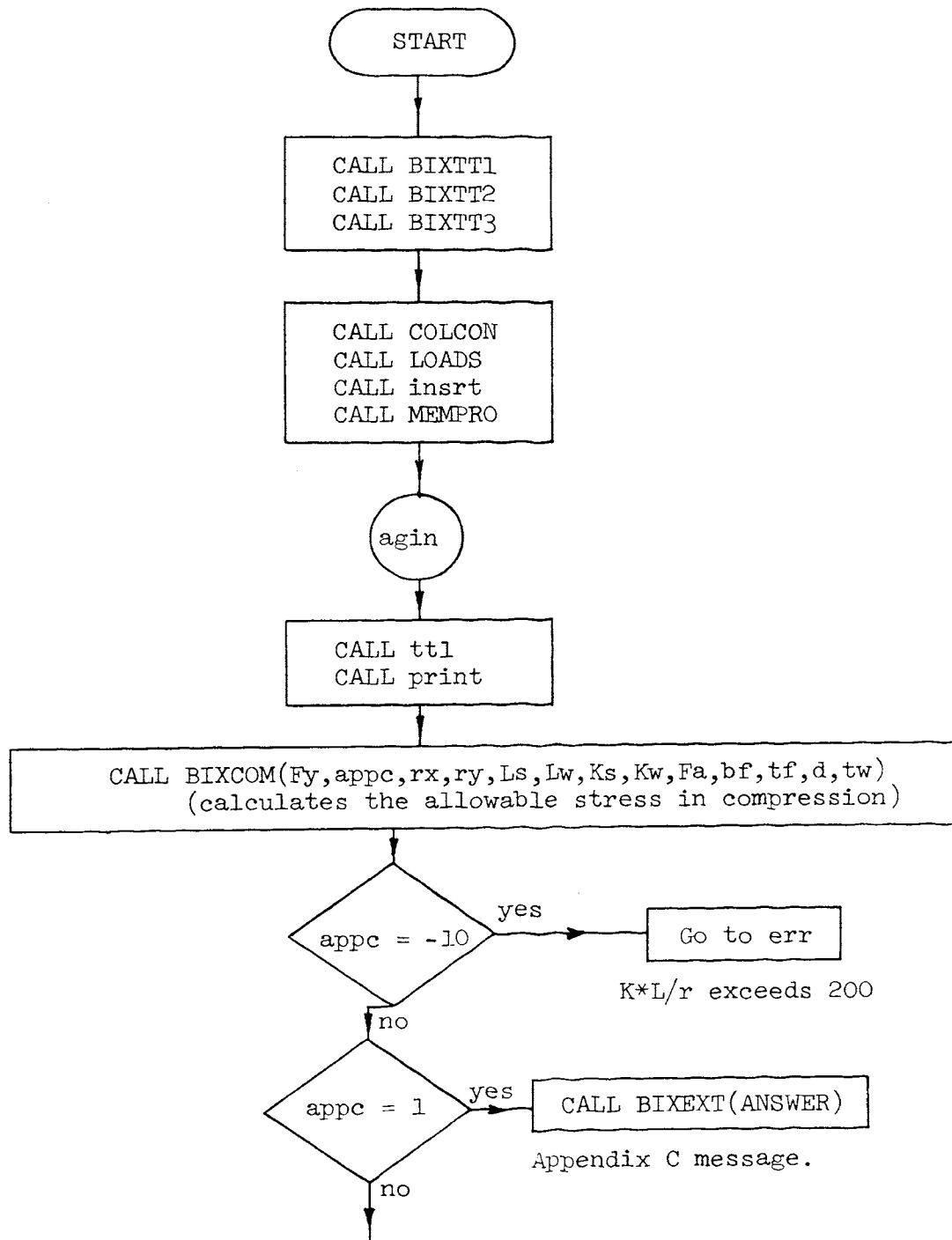
- L - length between bracing of compression flange.
- L - length between bracing in strong direction.
- L^S - length between bracing in weak direction.
- K^W - effective length factor strong direction.
- K^S - effective length factor weak direction.
- C_b^W - bending coefficient (ref. P. 5-19).
- C_b^b - bending coefficient (ref. P. 5-23).
- C_{my}^{mx} - bending coefficient (ref. P. 5-23).

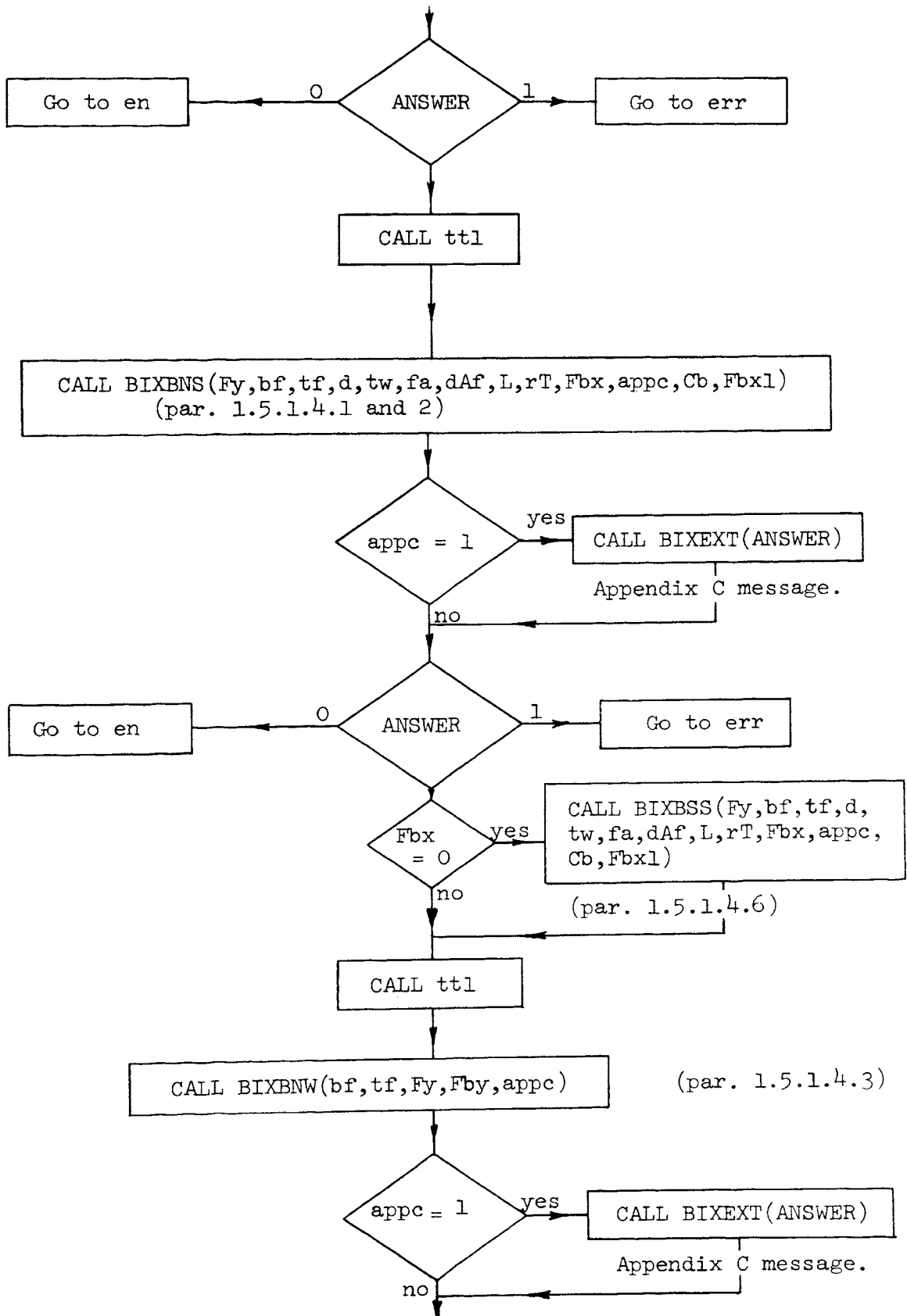
Overall Flow Chart BIXDSN

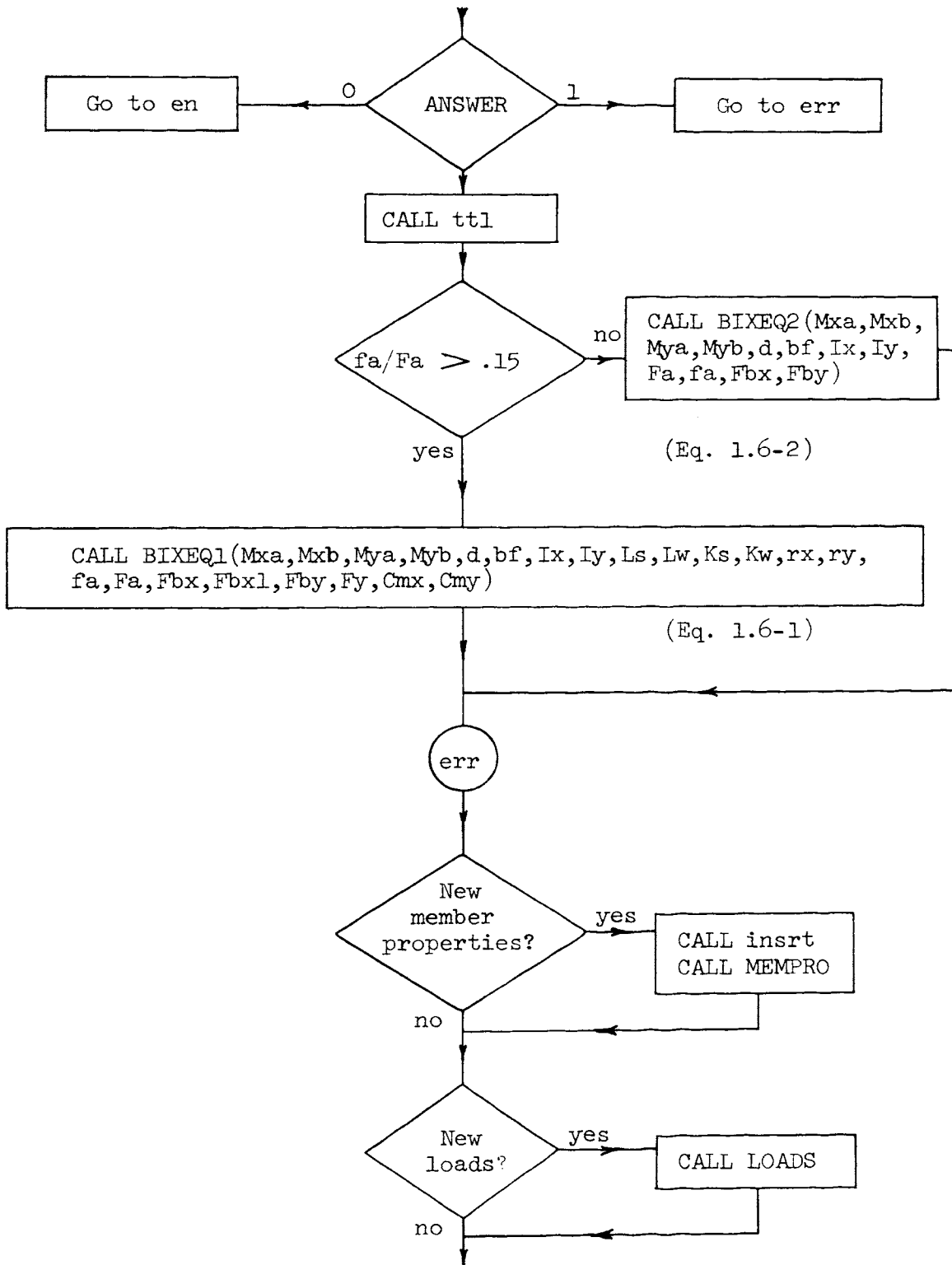


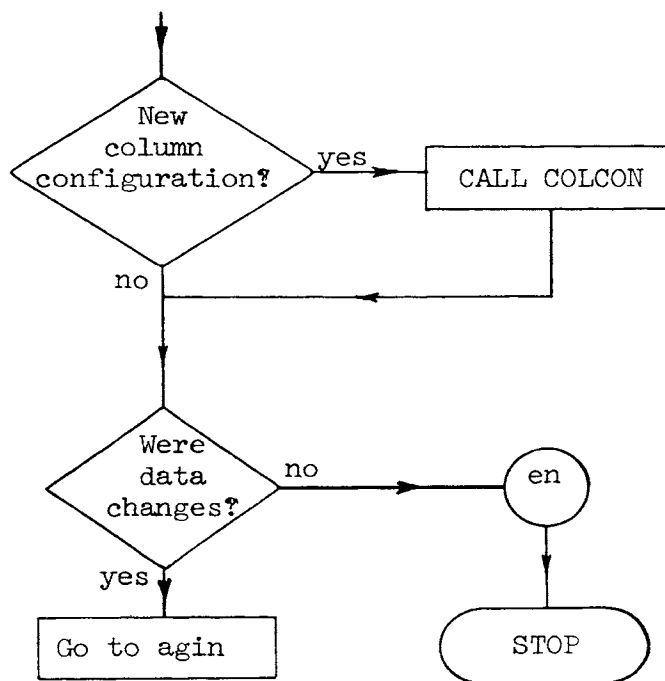


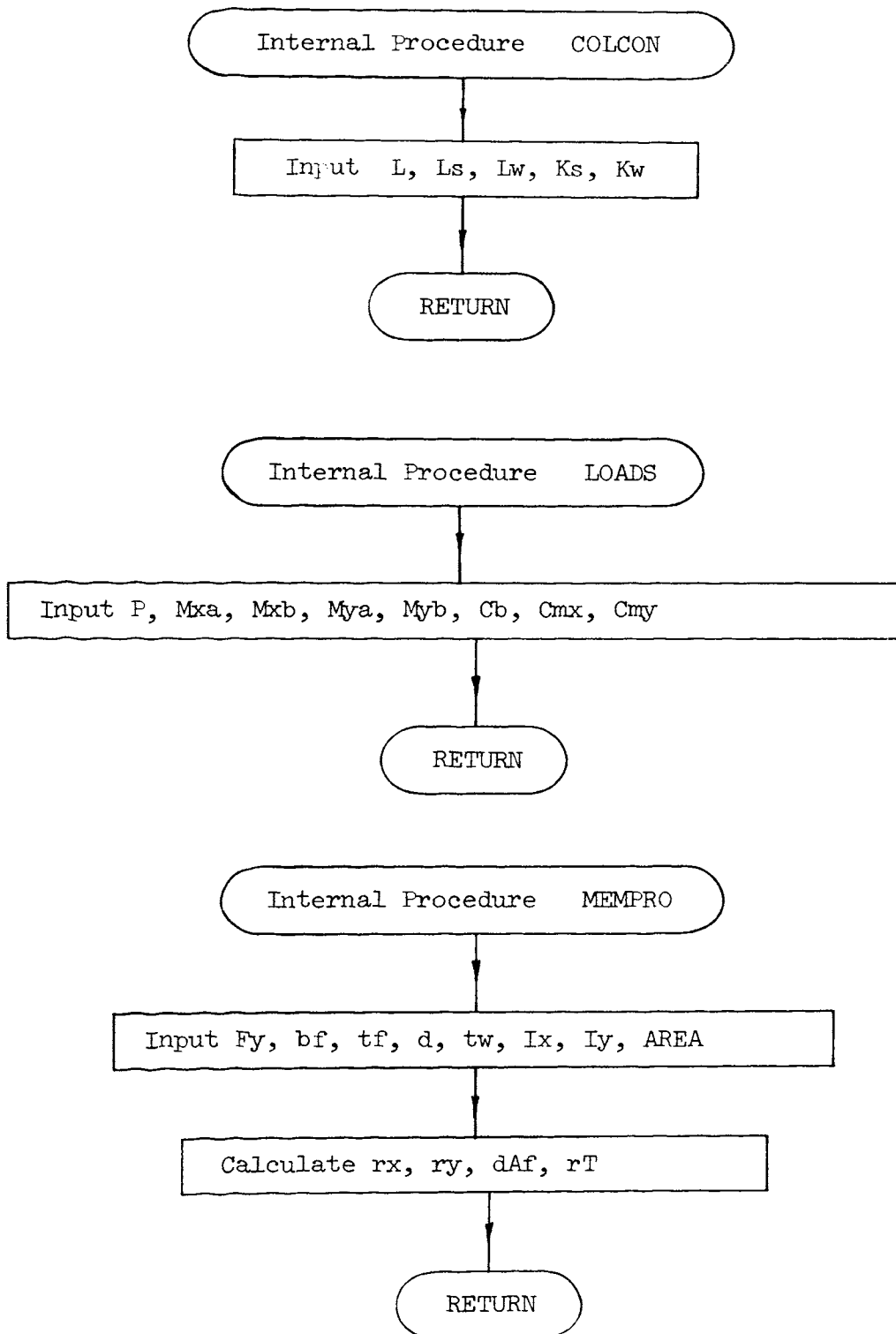
Detailed Flow Chart BIXDSN

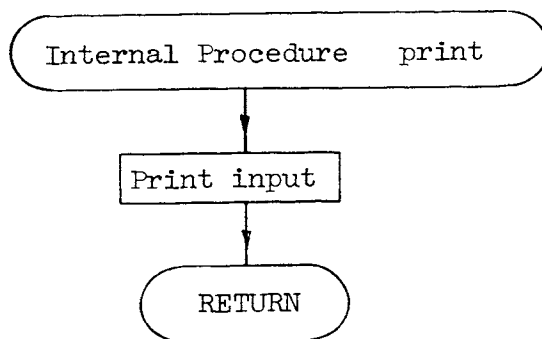
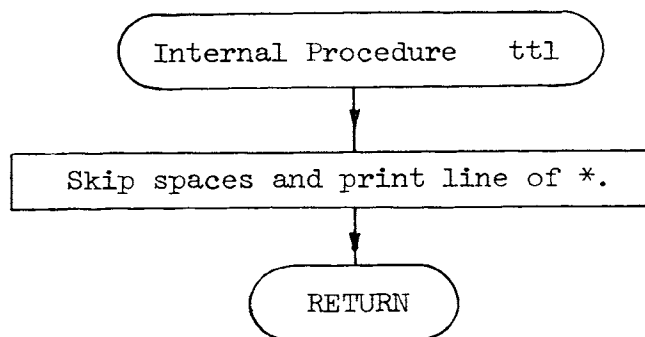
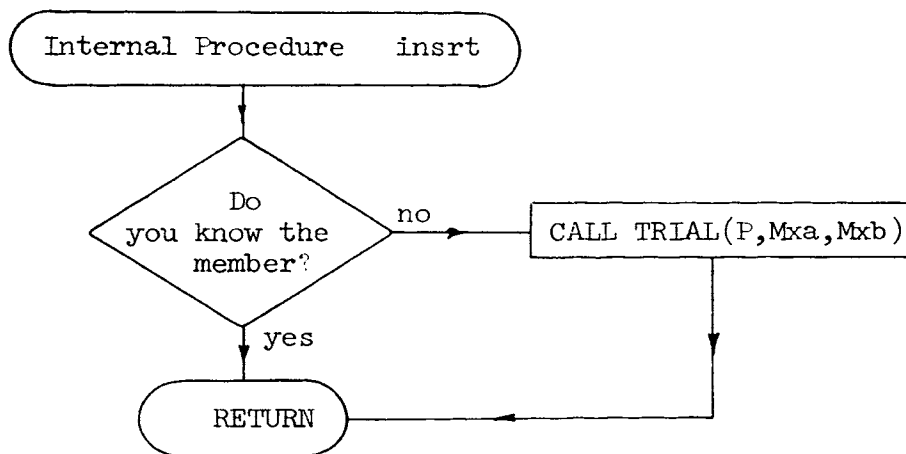


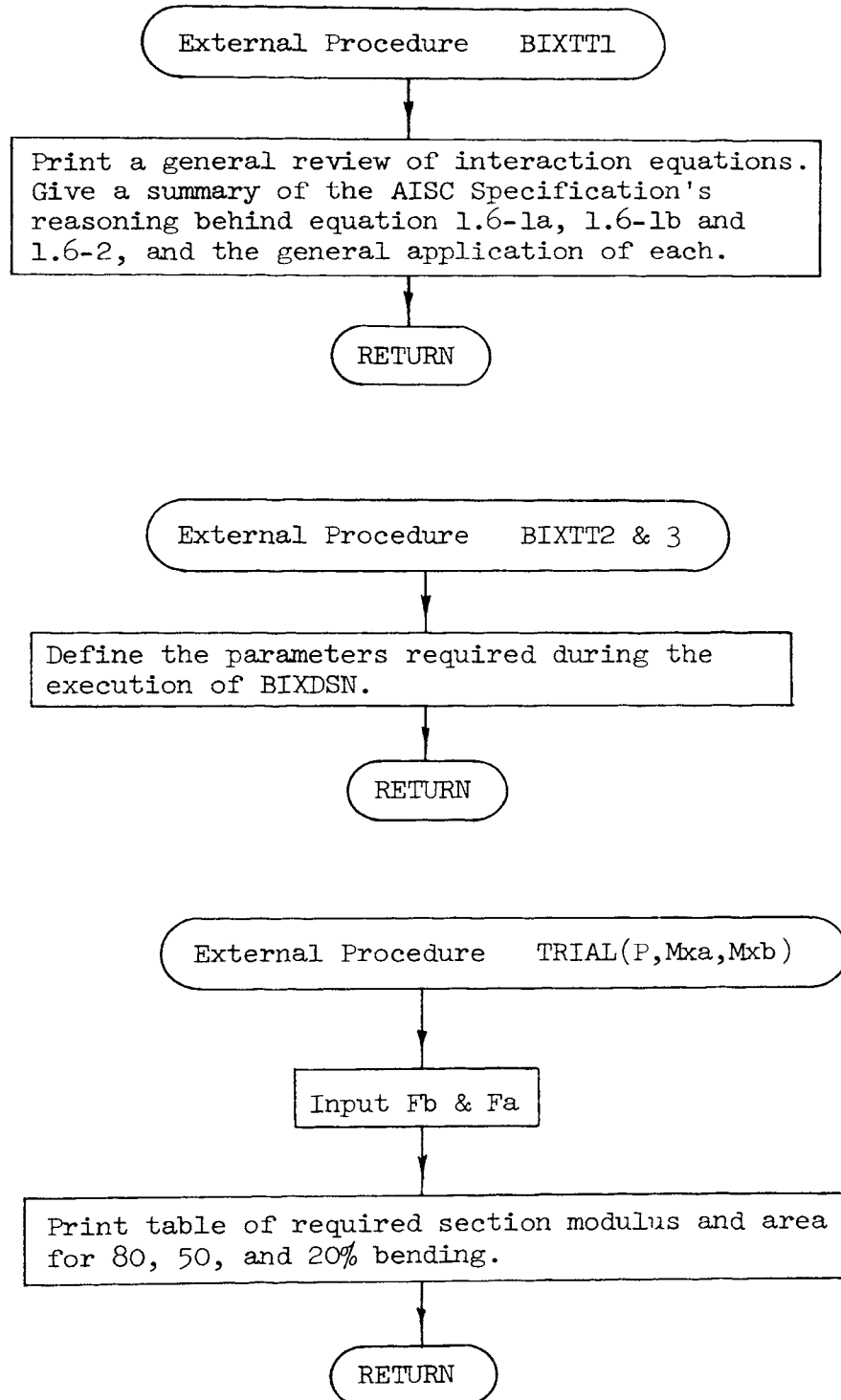


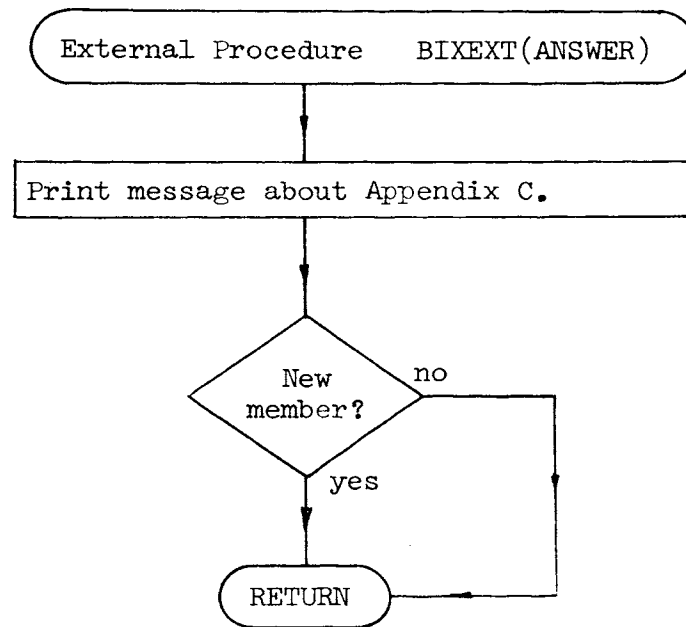


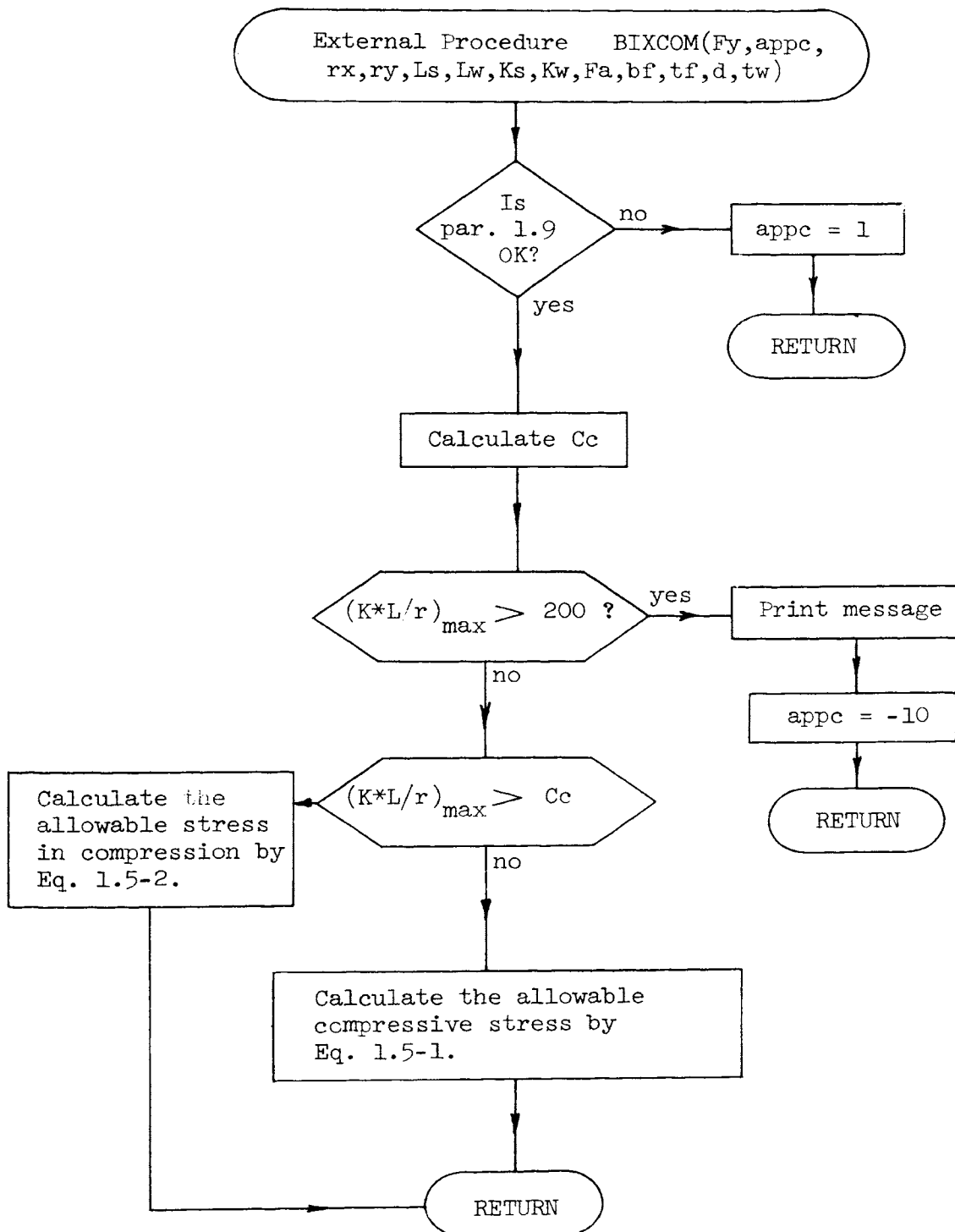


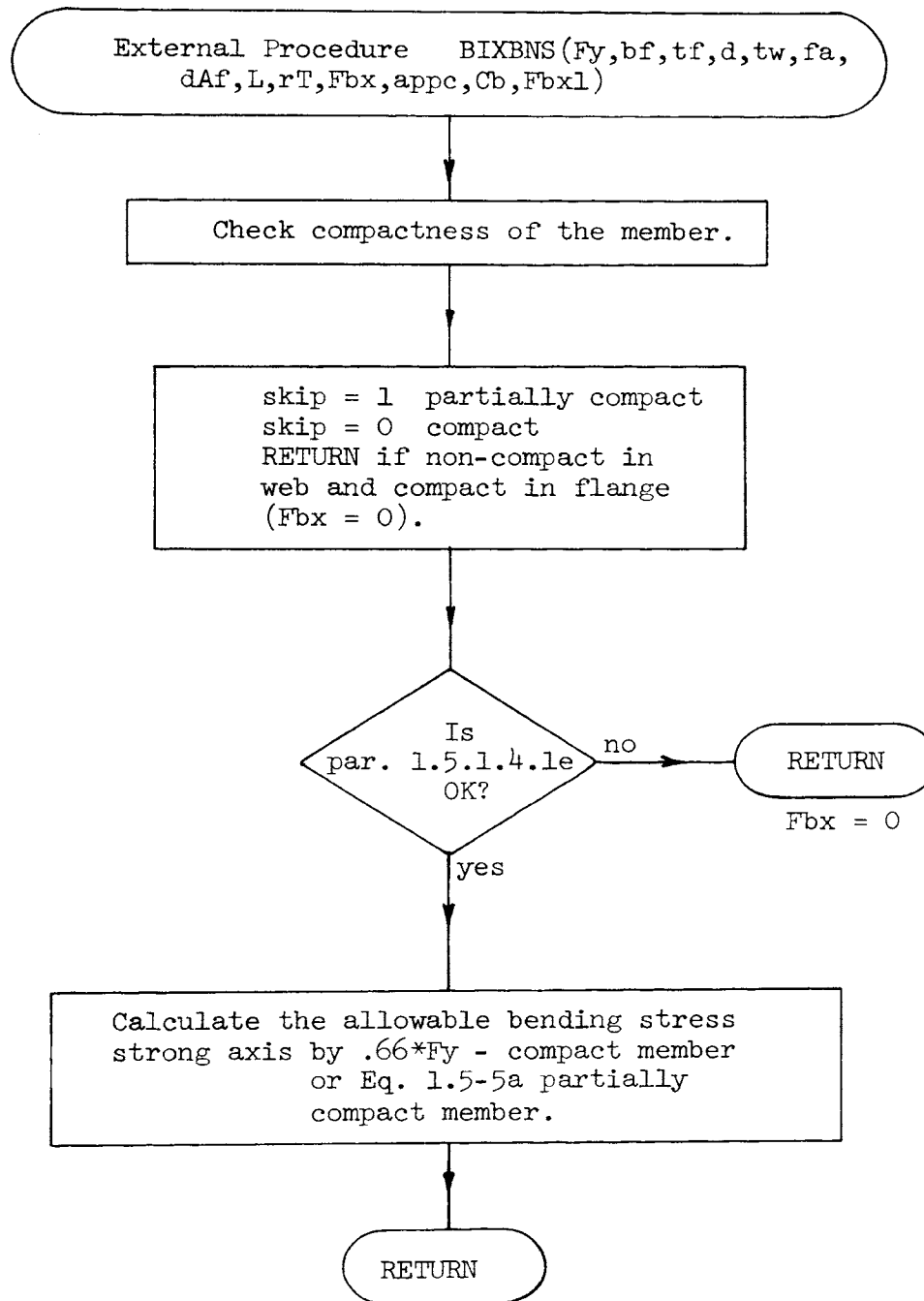


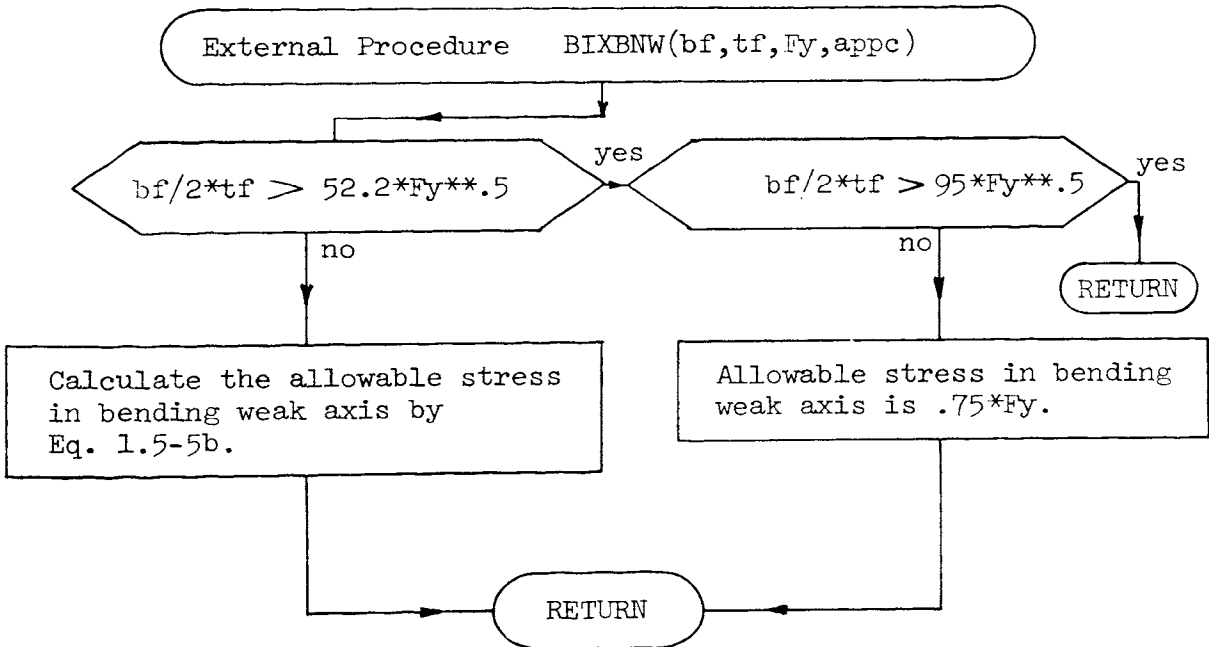
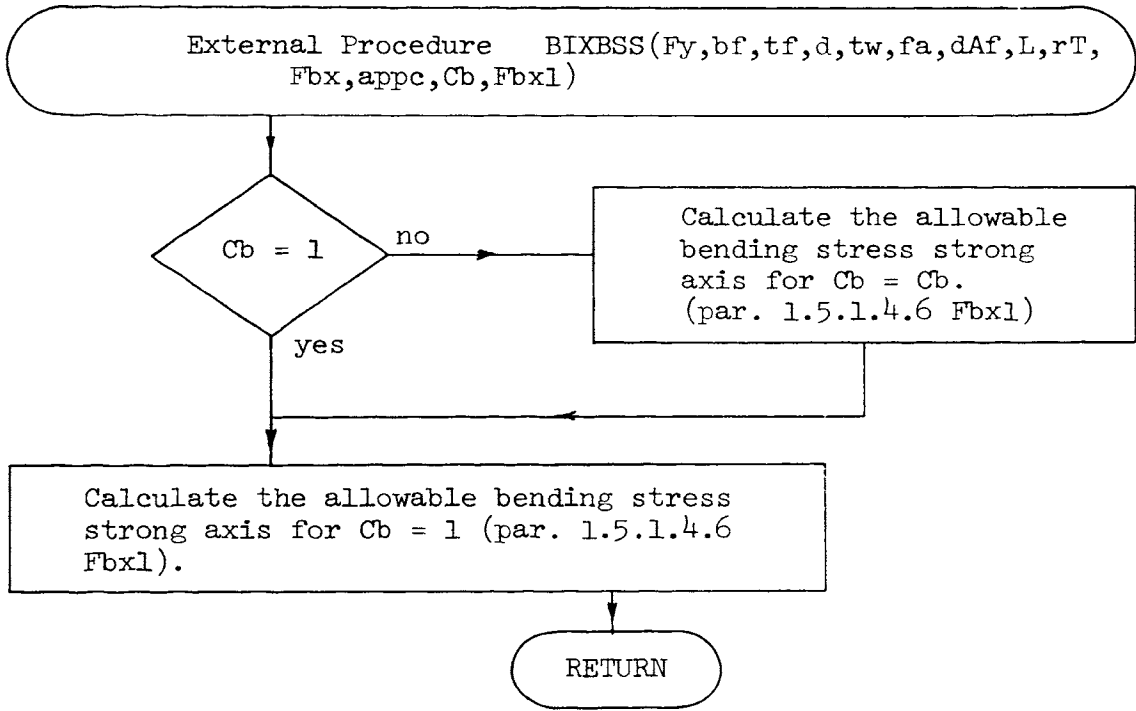


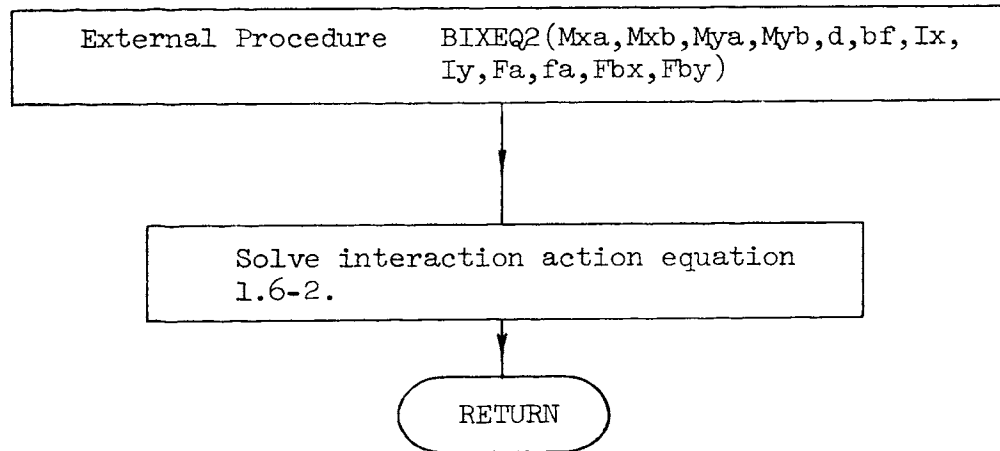
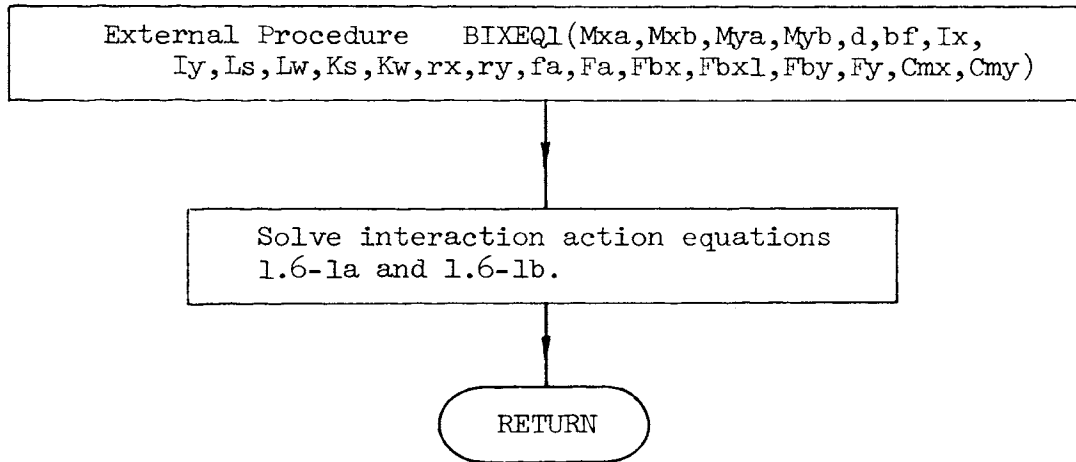












```

/* PROGRAM BIXDSN */
DECLARE BIXTT1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXTT2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXTT3 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXBNW ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXCOM ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXBNS ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXBSS ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE TRIAL ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXEXT ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXEQ1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BIXEQ2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE t1 DEC(4), t2 DEC(4), t3 DEC(4), t4 DEC(4);
DECLARE ttt CHAR(1);
DECLARE sum1 DEC(4), sum DEC(4), rT DEC(4), rx DEC(4), ry DEC(4);
PUT LIST('Would you like to skip the titles?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO skip;
CALL BIXTT1;
CALL BIXTT2;
CALL BIXTT3;
skip: CALL COLCOM;
CALL LOADS;
CALL insert;
CALL MEMPRO;
agin: CALL tt1;
PUT LIST(' * * * * * IDENTIFICATION * * * * *');
GET EDIT(ttt)(A(1));
CALL print;
CALL tt1;
CALL BIXCOM(Fy,appc,rx,ry,Ls,Lw,Ks,Fa,bf,tf,d,tw);
IF appc=-10 THEN GO TO err;
fa=P/AREA;
ANSWER=10;
IF appc=1 THEN CALL BIXEXT(ANSWER);

```

```

IF ANSWER=1 THEN GO TO err;
IF ANSWER=0 THEN GO TO en;
CALL tt1;
CALL BIXBNS(Fy,bf,tf,d,tw,fa,dAf,L,rT,Fbx,appc,Cb,Fbx1);
IF appc=1 THEN CALL BIXEXT(ANSWER);
IF ANSWER=1 THEN GO TO err;
IF ANSWER=0 THEN GO TO en;
IF Fbx=0 THEN CALL BIXBSS(Fy,bf,tf,d,tw,fa,dAf,L,rT,Fbx,appc,Cb,Fbx1);
CALL tt1;
CALL BIXBNW(bf,tf,Fy,Fby,appc);
IF appc=1 THEN CALL BIXEXT(ANSWER);
IF ANSWER=1 THEN GO TO err;
IF ANSWER=0 THEN GO TO en;
CALL tt1;
IF fa/Fa>.15 THEN GO TO egl;
CALL BIXEQ2(Mxa,Mxb,Mya,Myb,d,bf,Ix,Iy,Fa,fa,Fbx,Fby);
GO TO eq2;
eq1: CALL BIXEQ1(Mxa,Mxb,Mya,Myb,d,bf,Ix,Iy,Ls,Lw,Ks,Kw,rx,ry,fa,Fa,Fbx,Fbx1,Fby,Fy,
    Cmx,Cmy);
eq2: PUT LIST(' ');
CALL tt1;
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' ');
err: t1=1;
t2=1;
t3=1;
PUT LIST('Do you want to change the member properties? i.e. Fy,bf,tf,d,tw,Ix,Iy,
    AREA');
GET LIST(ANSWER);
IF ANSWER=1 THEN CALL insrt; ELSE t1=0;
IF ANSWER=1 THEN CALL MEMPRO;
PUT LIST('Do you want to change the loads? i.e. P,Mxa,Mxb,Mya,Myb,Cb,Cmx,Cmy');
GET LIST(ANSWER);

```

```

IF ANSWER=1 THEN CALL LOADS; ELSE t2=0;
PUT LIST('Do you want to change the column configuration? i.e. L,Ls,Lw,Ks,Kw');
GET LIST(ANSWER);
IF ANSWER=1 THEN CALL COLCON; ELSE t3=0;
IF t1+t2+t3<.01 THEN GO TO en;
GO TO agin;
COLCON: PROCEDURE ;
        PUT LIST(' ');
        PUT LIST('Input the column configuration. ');
        GET LIST(L,Ls,Lw,Ks,Kw);
        RETURN ;
        END COLCON;
LOADS:  PROCEDURE ;
        PUT LIST('Input the loads. ');
        GET LIST(P,Mxa,Mxb,Mya,Myb,Cb,Cmx,Cmy);
        RETURN ;
        END LOADS;
MEMPRO: PROCEDURE ;
        PUT LIST('Input the member properties. ');
        GET LIST(Fy,bf,tf,d,tw,Ix,Iy,AREA);
        rx=sqrt(Ix/AREA);
        ry=sqrt(Iy/AREA);
        dAf=d/(tf*bf);
        rT=bf**3*tf/12;
        rT=rT+(d/6-tf)*tw**3/12;
        rT=sqrt(rT/(bf*tf*(d/6-tf)*tw));
        RETURN ;
        END MEMPRO;
insrt:  PROCEDURE ;
        PUT LIST('Do you know your first trial member? ');
        GET LIST(ANSWER);
        IF ANSWER=1 THEN GO TO 1b1;
        CALL TRIAL(P,Mxa,Mxb);
1b1:    ANSWER=1;
        RETURN ;

```



```

BIXTT1:  PROCEDURE ;
          PUT LIST(' ');
          PUT LIST('Program BIXDSN will review and interact with the user in the design
of Beam-Columns.');
```

of Beam-Columns.');

```

          PUT LIST(' ');
          PUT LIST('Before proceeding the user must have covered beam design (BENTST and
BENDSN) and column');
```

BENTST and column');

```

          PUT LIST(' design (COLTST) and have read Section 1.6-1 in the AISC
Specifications.');
```

Specifications.');

```

          PUT LIST(' ');
          PUT LIST('Beam-Columns are members subjected simultaneously to bending and axial
loads.');
```

loads.');

```

          PUT LIST(' Basically, a Beam-Column design is satisfactory if the stresses
satisfy some type');
```

satisfy some type');

```

          PUT LIST(' of interaction equation. An interaction equation is usually of the
form');
```

of interaction equation. An interaction equation is usually of the form');

```

          PUT LIST(' ');
          PUT LIST('          fa   fbx   fby');
          PUT LIST('          -- + --- + --- <= 1.0');
```

$$\frac{f_a}{F_a} + \frac{f_{bx}}{F_{bx}} + \frac{f_{by}}{F_{by}} \leq 1.0$$

```

          PUT LIST('          Fa   Fbx   Fby');
```

where');

```

          PUT LIST('          f = actual stress');
```

f = actual stress');

```

          PUT LIST('          F = allowable stress');
```

F = allowable stress');

```

          PUT LIST(' ');
          PUT LIST('Note that each term of the equation represents a ratio of actual to
allowable');
```

Note that each term of the equation represents a ratio of actual to allowable');

```

          PUT LIST(' stress. Each of these ratios may be viewed as the percent of the
member resisting');
```

stress. Each of these ratios may be viewed as the percent of the member resisting');

```

          PUT LIST(' that type of load, ie fa/Fa = .2 says 20% of the member is used by
the');
```

that type of load, ie fa/Fa = .2 says 20% of the member is used by the');

```

          PUT LIST(' axial load and only 80% is left to resist bending. This percentage
will be used later');
```

axial load and only 80% is left to resist bending. This percentage will be used later');

```

          PUT LIST(' to perform design.');
```

to perform design.');

```

          PUT LIST(' ');
```

```

PUT LIST('For the design of combined stresses the AISC Specifications give
  three interaction equations');
PUT LIST('  Eq. 1.6-1a&b and 1.6-2. All of the form given above. Equation
  1.6-1a has an amplification');
PUT LIST('  factor applied to the bending terms. This factor is  $C_m/(1 - f_a/F_e)$ 
  and is applied to ');
PUT LIST('  account for the increase in bending moment due to the deflection,
  i.e.  $P*\Delta$  effect.');
```

Equation 1.6-1a is a stability equation, ie it accounts for buckling.');

```

PUT LIST('  In it  $F_b$ , the allowable stress in bending, is calculated based on
   $C_b = 1.$ ');
```

The actual stress f_b is calculated based on the bending moment as defined');

```

PUT LIST('  in the third column of Table C 1.6.1.1 page 5-131.');
```

Equation 1.6-1b is a yield equation or joint equation, ie it accounts for');

```

PUT LIST('  the yield at the joints. In it  $F_b$  is calculated based on  $C_b$  as
  defined for ');
PUT LIST('  bending. As above the actual stress  $f_b$  is calculated as per
  Table c 1.6.1.1.');
```

```

RETURN ;
END ;
```

```

BIXTT2:  PROCEDURE ;
im1:     IMAGE;
        * * * * *
PUT LIST(' ');
PUT LIST('Input to Bixdsn is divided into three groups.');
```

Column Configuration

```

PUT LIST('L - distance between bracing against twist of the compression flange
(ft.).');
```

Ls - distance between bracing against lateral displacement of the member in the strong direction (ft.).

```

PUT LIST('Lw - distance between bracing against lateral displacement of the
member in the weak direction (ft.).');
```

Ks - effective length factor in the strong direction.

```

PUT LIST('Kw - effective length factor in the weak direction.');
```

Loads

```

PUT LIST('P - axial load (kips).');
```

Mx - applied moment about the x or strong axis of the beam (ft-kips).

```

PUT LIST('Mxa and Mxb refers to the moment applicable in equation
1.6-1a');
```

Mxa and Mxb, respectively. Ref. column 3 of Table C 1.6.1.1 page 5-131.

```

PUT LIST('My - applied moment about the y or weak axis of the beam (ft-kips).');
```

$C_b = 1.75 + 1.05 \cdot (M_1/M_2) + 0.3 \cdot (M_1/M_2)^2 \leq 2.3$ where M_1 is the smaller and M_2 the larger bending moment at the ends of the unbraced length. Ref p. 5-19 and 5-104.

```

PUT LIST('Cmx - a coefficient to account for sidesway effects in the x or
strong direction.');
```

Ref p. 5-23 and 5-131.

```

PUT LIST('Cmy - a coefficient to account for sidesway effects in the y or weak
direction.');
```

Ref p. 5-23 and 5-131.

```

PUT LIST(' ');
PUT IMAGE(1)(im1);

```

```
PUT LIST(' Member Properties');
PUT LIST('Fy - yield stress of the steel (ksi).');
PUT LIST('bf - width of the flange (in.).');
PUT LIST('d - total depth of the member (in.).');
PUT LIST('tw - thickness of the web (in.).');
PUT LIST('Ix - moment of inertia about the x or strong axis (in**4).');
PUT LIST('Iy - moment of inertia about the y or weak axis(in**4).');
PUT LIST('AREA - cross section area of the member (in**2).');
RETURN ;
END ;
```

```

BIXTT3: PROCEDURE ;
im1: IMAGE;
*****
PUT IMAGE(1)(im1);
PUT LIST(' Other Input Items');
PUT LIST('IDENTIFICATION - when this appears type any information which will
  help you in');
PUT LIST(' later identifying the output. ');
PUT LIST('ANSWER - usually requiring a yes or no indication. (yes = 1 no = 0)');
PUT IMAGE(1)(im1);
PUT LIST(' ');
PUT LIST('By changing specific values in these input groups BIXDSN will perform
  investigation');
PUT LIST(' and design. By defining certain values of Mx,My and P, BIXDSN can be
  used to');
PUT LIST(' investigate and design columns, beams and beam-columns');
PUT LIST(' ');
PUT LIST('Since the user must work several problems using BIXDSN to obtain a
  feel for ');
PUT LIST(' Combined Stress Design, the input to BIXDSN has been streamlined as
  much as');
PUT LIST(' possible. Input parameters always remain as they were as last
  defined by ');
PUT LIST(' the user. If a parameter is requested and the user wishes NOT to
  redefine');
PUT LIST(' it, he may simply press the return key and that parameter will not
  CHANGE .');
PUT LIST(' ');
RETURN ;
END ;

```

```
BIXEXT:  PROCEDURE (ANSWER);
          PUT LIST(' ');
          PUT LIST('Because of a violation of Section 1.9 the member that you have
          chosen');
          PUT LIST('  will not be 100% effective in resisting the load. The reduction
          in');
          PUT LIST('  section may be calculated as per Appendix C. This procedure has
          not');
          PUT LIST('  been included in BIXDSN. Therefore you have the option of
          calculating');
          PUT LIST('  this reduction by hand or of trying a new member. ');
          PUT LIST('Would you like to try a new member?');
          GET LIST(ANSWER);
          RETURN ;
          END ;
```

```

TRIAL:  PROCEDURE (P,Mxa,Mxb);
        PUT LIST('Based on your experience in beam and column design assume an allowable');
        PUT LIST(' bending stress and compression stress.');
```

Percent of allowable	Section Modulus	Area
load used by		
bending		

```

im1:   IMAGE;
        -----
        t3=80;
        t2=t2/.8;
        t1=t1/.2;
        PUT IMAGE(t3,t2,t1)(im1);
        t3=50;
        t2=t2*80/50;
        t1=t1*20/50;
        PUT IMAGE(t3,t2,t1)(im1);
        t3=20;
        t2=t2*50/20;
        t1=t1*50/80;
        PUT IMAGE(t3,t2,t1)(im1);
        PUT LIST('Based on the above chart choose a W member.');
```

RETURN ;

END TRIAL;


```

BIXCOM:  PROCEDURE (Fy,appc,rx,ry,Ls,tw,Ks,Kw,Fa,bf,tf,d,tw);
          DECLARE ttt CHAR(1);
          DECLARE t1 DEC(4), Cc DEC(4), t2 DEC(4);
          PUT LIST('Find the allowable compressive stress. ');
          sfy=sqrt(Fy);
          appc=0;
          IF bf/(2*tf)>95/sfy THEN GO TO pend;
          IF (d-2*tf)/tw>253/sfy THEN GO TO pend;
          GO TO cont;
pend:     appc=1;
          RETURN ;
cont:     Cc=sqrt(2*9.86959*29000/Fy);
          PUT LIST('Cc = (2 (pi)**2*E/Fy)**.5 = ',Cc);
          t1=Kw*Lw*12/ry;
          t2=Ks*Ls*12/rx;
          PUT LIST('K*L/r weak = ',t1,' strong = ',t2);
          IF t1>t2 THEN GO TO 1b2;
          PUT LIST('Strong axis controls the allowable stress. ');
          GO TO 1b3;
1b2:     PUT LIST('Weak axis controls the allowable stress. ');
1b3:     t1=max(t1,t2);
          IF t1>200 THEN GO TO 1b6;
          IF t1>Cc THEN GO TO 1b4;
          PUT LIST('Allowable stress set by Eq 1.5-1. ');
          t2=t1**2;
          t3=t2*t1;
          c2=Cc**2;
          c3=Cc**3;
          t1=(1-t2/(2*c2))*Fy/(5/3+3*t1/(8*Cc)-t3/(8*c3));
          GO TO 1b5;
1b4:     PUT LIST('Allowable stress set by Eq 1.5-2. ');
          t1=12*9.86959*29000/(23*t1**2);
1b5:     PUT LIST('Fa = ',t1);
          Fa=t1;
          RETURN ;

```

```
1b6:  PUT LIST('K*L/r exceeds 200 member cannot be used. ');  
      appc--10;  
      RETURN ;  
      END ;
```

```

BIXBNS:  PROCEDURE (Fy,bf,tf,d,tw,fa,dAf,L,rT,Fbx,appc,Cb,Fbx1);
          DECLARE ttt CHAR(1);
          DECLARE test1 DEC(4);
          DECLARE test DEC(4);
          DECLARE t1 DEC(4);
          appc=0;
          PUT LIST('Find the allowable bending stress strong axis. ');
          Fbx=0;
          skip=0;
          sfy=sqrt(Fy);
          t1=52.2/sfy;
          IF bf/(2*tf)<=t1 THEN GO TO 1b1;
          t1=95/sfy;
          IF bf/(2*tf)>t1 THEN GO TO pend;
          /* skip= 1 member is partially compact. */;
          skip=1;
          /* paragraph d */;
1b1:      IF fa/Fy>.16 THEN t1=257/sfy; ELSE t1=412/sfy*(1-2.33*fa/Fy);
          IF d/tw>t1 THEN GO TO 1b3;
          IF skip=1 THEN GO TO s1;
          PUT LIST('The member is COMPACT i.e. no local buckling will occur. ');
          GO TO s2;
s1:      PUT LIST('The member is partially compact. ');
s2:      PUT LIST(' ');
          t1=76*bf/sfy;
          t2=20000/(dAf*Fy);
          t1=min(t1,t2);
          IF L*12>t1 THEN GO TO 1b3;
          t1=.66*Fy;
          IF Fy=36 then t1-24;
          IF skip=1 THEN GO TO 1b4;
          PUT LIST('The allowable stress in tension and compression is equal to .66*Fy
          = ',t1);
          Fbx1,Fbx=t1;
          RETURN ;

```

```
1b4:   PUT LIST('The allowable stress in tension and compression is determined by
        Section 1.5.1.4.2. ');
        t1=Fy*(.733-.0014*bf/(2*tf)*Fy**.5);
        PUT LIST('   Fb = Fy(.733 - 0.0014(bf/2tf)Fy**.5) = ',t1);
        Fbx1,Fbx=t1;
1b3:   RETURN ;
pend:  appc=1;
        RETURN ;
        END ;
```

```

BIXBSS:  PROCEDURE (Fy,bf,tf,d,tw,fa,dAf,L,rT,Fbx,appc,Cb,Fbx1);
          DECLARE test DEC(4), test1 DEC(4), t1 DEC(4), tcb DEC(4);
          sfy=sqrt(Fy);
          IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
          PUT LIST('The allowable tension stress is .6*Fy = ',t1,' as per Section
            1.5.1.4.5. ');
          tcb=Cb;
          Cb=1;
1b5:     PUT LIST(' * * * * for Cb = ',Cb);
          t1=L*12/rT;
          test=sqrt(102000*Cb/Fy);
          IF t1<test THEN GO TO 1b7;
          test1=sqrt(510000*Cb/Fy);
          IF t1>=test1 THEN GO TO 1b8;
          PUT LIST('L/rT is between (102000*Cb/Fy)**.5 and (510000*Cb/Fy)**.5');
          PUT LIST('Therefore the allowable compression stress is the larger of');
          t1=(2/3-Fy*(L*12/rT)**2/(1530000*Cb))*Fy;
          PUT LIST('    Fb = (2/3 - Fy(L/rT)**.5/1530000*Cb)Fy = ',t1);
          PUT LIST('    and');
          test=12000*Cb/(L*12*dAf);
          PUT LIST('    Fb = 12000*Cb/Ld/Af = ',test);
          IF Fy=36 THEN test1=22; ELSE test1=Fy*.6;
          PUT LIST('    not to exceed Fb = .6*Fy = ',test1);
          t1=max(t1,test);
          t1=min(t1,test1);
          PUT LIST('    USE: Fb = ',t1);
          GO TO 1b2;
1b8:     PUT LIST('L/rT is greater than (510000*Cb/Fy)**.5');
          test1=12000*Cb/(L*12*dAf);
          test=170000*Cb/(L*12*Cb/rT)**2;
          PUT LIST('Therefore the allowable compression stress is the larger of');
          PUT LIST('    Fb = 170000*Cb/(L/rT)**2 = ',test);
          PUT LIST('    and');
          PUT LIST('    Fb = 12000*Cb/Ld/Af = ',test1);
          IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
          PUT LIST('    not to exceed Fb = .6*Fy = ',t1);

```

```
test=max(test,test1);
t1=min(t1,test);
PUT LIST('    USE: Fb = ',t1);
GO TO 1b2;
1b7: PUT LIST('L/rT is less than ('L/rT is less than (102000.*Cb/Fy)**.5');
IF Fy=36 THEN t1=22; ELSE t1=.6*Fy;
PUT LIST('    USE: Fb = ',t1);
1b2: Fbx=t1;
IF Cb=1 THEN Fbx1=t1;
IF tcb=1 THEN RETURN ; ELSE Cb=tcb;
tcb=1;
GO TO 1b5;
END ;
```

```

BIXBNW:  PROCEDURE (bf,tf,Fy,Fby,appc);
          DECLARE t1 DEC(4);
          PUT LIST('Find the allowable bending stress  weak axis. ');
          appc=0;
          sfy=sqrt(Fy);
          IF bf/(2*tf)>52.2/sfy THEN GO TO 1b1;
          Fby=.75*Fy;
          t1=Fby;
          PUT LIST('Allowable stress set by  Fby = .75*Fy = ',t1);
          RETURN ;
1b1:     IF bf/(2*tf)>95/sfy THEN GO TO pend;
          Fby=Fy*(.933-.0035*(bf/(2*tf))*sqrt(Fy));
          t1=Fby;
          PUT LIST('Allowable stress set by Eq.1.5-5b Fby = ',t1);
          RETURN ;
pend:    appc=1;
          RETURN ;
          END ;

```

```

BIXEQ1:  PROCEDURE (Mxa,Mxb,Mya,Myb,d,bf,Ix,Iy,Ls,Lw,Ks,Kw,rx,ry,fa,Fa,Fbx,Fbx1,Fby,Fy,
          Cmx,Cmy);
          DECLARE t1 DEC(4), t2 DEC(4), t3 DEC(4), sum1 DEC(4);
          PUT LIST('Combined stresses');
          fbx=Mxb*12*d/2/Ix;
          fbx1=Mxa*12*d/2/Ix;
          fby=Myb*12*bf/2/Iy;
          fby1=Mya*12*bf/2/Iy;
          IF Ls=0 THEN Fex=Cmx; ELSE Fex=Cmx/(1-fa/(149331.188/(Ks*Ls*12/rx)**2));
          IF Lw=0 THEN Fey=Cmy; ELSE Fey=Cmy/(1-fa/(149331.188/(Kw*Lw*12/ry)**2));
          t1=fa/Fa;
          t2=Fex*fbx1/Fbx1;
          t3=Fey*fby1/Fby;
          PUT LIST('EQ. 1.6-1a');
          PUT IMAGE(t1,t2,t3)(im1);
          sum1=t1+t2+t3;
          PUT LIST('EQ. 1.6-1b');
          t4=.6*Fy;
          IF Fy=36 THEN t4=22;
          PUT IMAGE(fa/t4,fbx/Fbx,fby/Fby)(im1);
          sum=fa/t4+fbx/Fbx+fby/Fby;
          IF sum1>sum THEN GO TO 1b4;
          PUT LIST('EQ. 1.6-1b controls');
          t1=fa/t4;
          t2=fbx/Fbx;
          t3=fby/Fby;
          sum1=t1+t2+t3;
          GO TO 1b3;
1b4:  PUT LIST('EQ. 1.6-1a controls');
1b3:  PUT LIST(' ');
          IF sum1>1 THEN PUT LIST('Section is not adequate'); ELSE PUT LIST('Section OK');
          PUT LIST(' Sum = ',sum1);
          t1=t1*100;
          t2=t2*100;
          t3=t3*100;

```



```
im1: PUT LIST('Axial Load accounts for ',t1,' %.');  
      PUT LIST('Bending X accounts for ',t2,' %.');  
      PUT LIST('Bending Y accounts for ',t3,' %.');  
      IMAGE;  
      -.-.- + -.-.- + -.-.- ? <= 1.0  
      RETURN ;  
      END ;
```

```

BIXEQ2:  PROCEDURE (Mxa,Mxb,Mya,Myb,d,bf,Ix,Iu,Fa,fa,Fbx,Fby);
DECLARE t1 DEC(4), t2 DEC(4), t3 DEC(4), sum DEC(4);
PUT LIST('Combined stresses. ');
fbx=Mxb*12*d/2/Ix;
fbx1=Mxa*12*d/2/Ix;
fby=Myb*12*bf/2/Iy;
fby1=Mya*12*bf/2/Iy;
PUT LIST(' ');
PUT LIST('EQ. 1.6-2 Controls');
t1=fa/Fa;
t2=fbx1/Fbx;
t3=fby1/Fby;
PUT LIST('for Ma');
PUT IMAGE(t1,t2,t3)(im1);
sum1=t1+t2+t3;
t2=fbx/Fbx;
t3=fby/Fby;
sum=t1+t2+t3;
PUT LIST('for Mb');
PUT IMAGE(t1,t2,t3)(im1);
IF sum>sum1 THEN PUT LIST('Mb controls'); ELSE PUT LIST('Ma controls');
If sum>sum1 THEN GO TO 1b3;
t2=fbx1/Fbx;
t3=fby1/Fby;
sum=sum1;
1b3:  PUT LIST(' ');
IF sum>1 THEN PUT LIST('Section is not adequate'); ELSE PUT LIST('Section OK');
PUT LIST(' Sum = ',sum);
t1=t1*100;
t2=t2*100;
t3=t3*100;
PUT LIST('Axial Load accounts for ',t1,' %');
PUT LIST('Bending X accounts for ',t2,' %');
PUT LIST('Bending Y accounts for ',t3,' %');
im1:  IMAGE;

```

```
----- + ----- + ----- ? <= 1.0  
RETURN ;  
END BIXEQ2;
```

Program Name:

PLADSN

Purpose:

Program PLADSN reviews for the user the philosophy of plastic design of bending members. It will also interact with the user in the plastic design of beam-columns.

Limitations:

Only W sections subjected to bending about the strong axis and column loads may be designed using PLADSN.

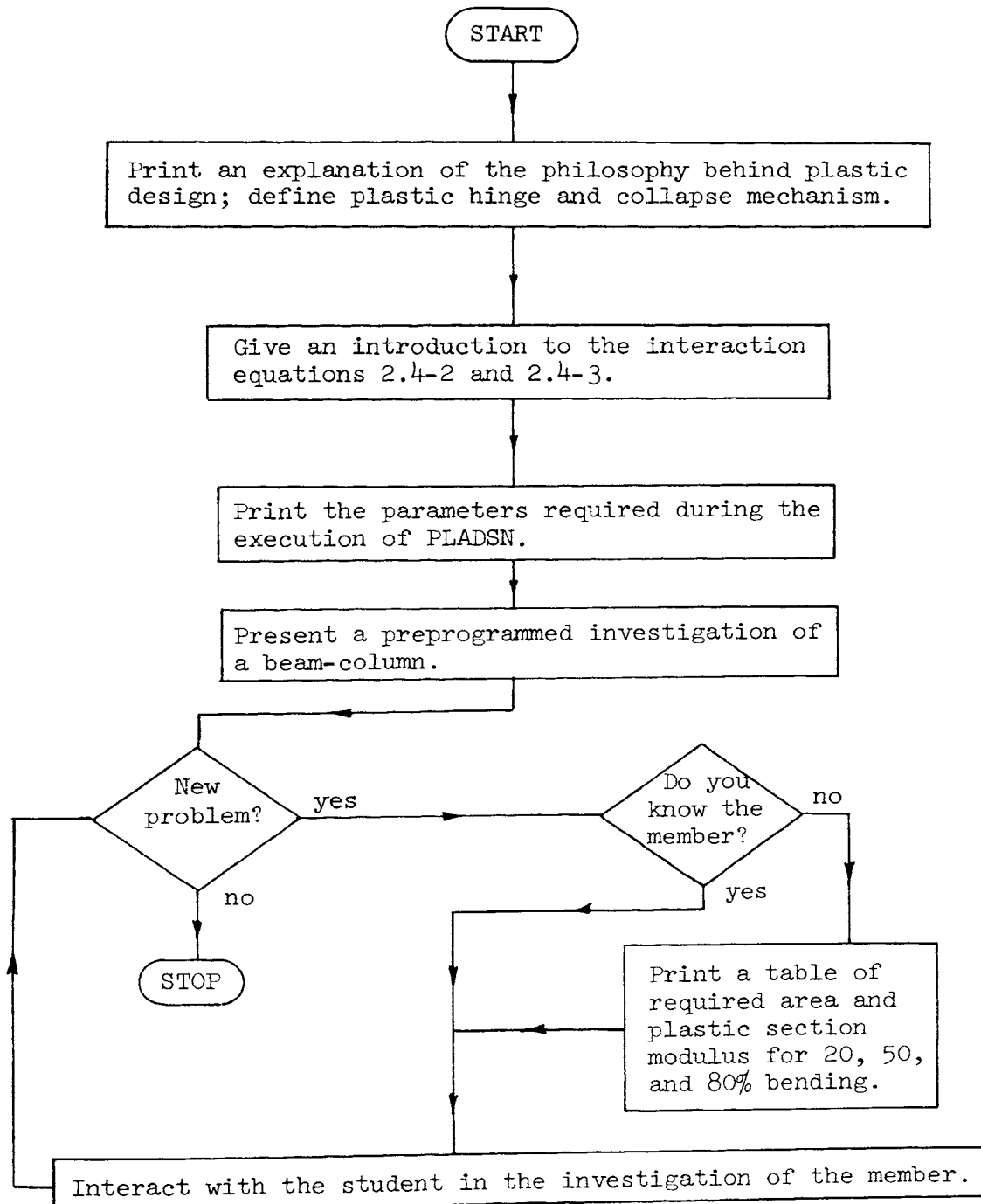
Prerequisite:

Read Part 2 of the AISC Specification.

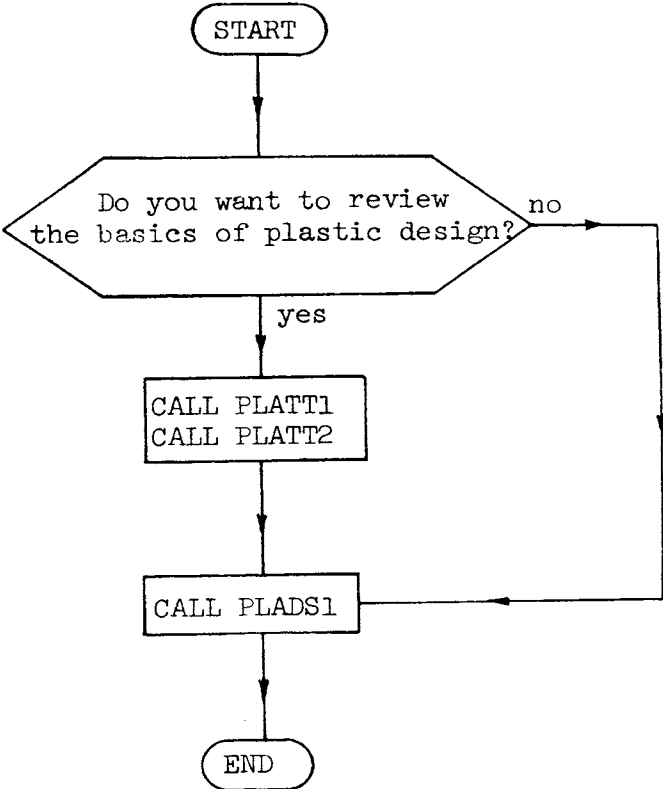
Data Required Before Execution:

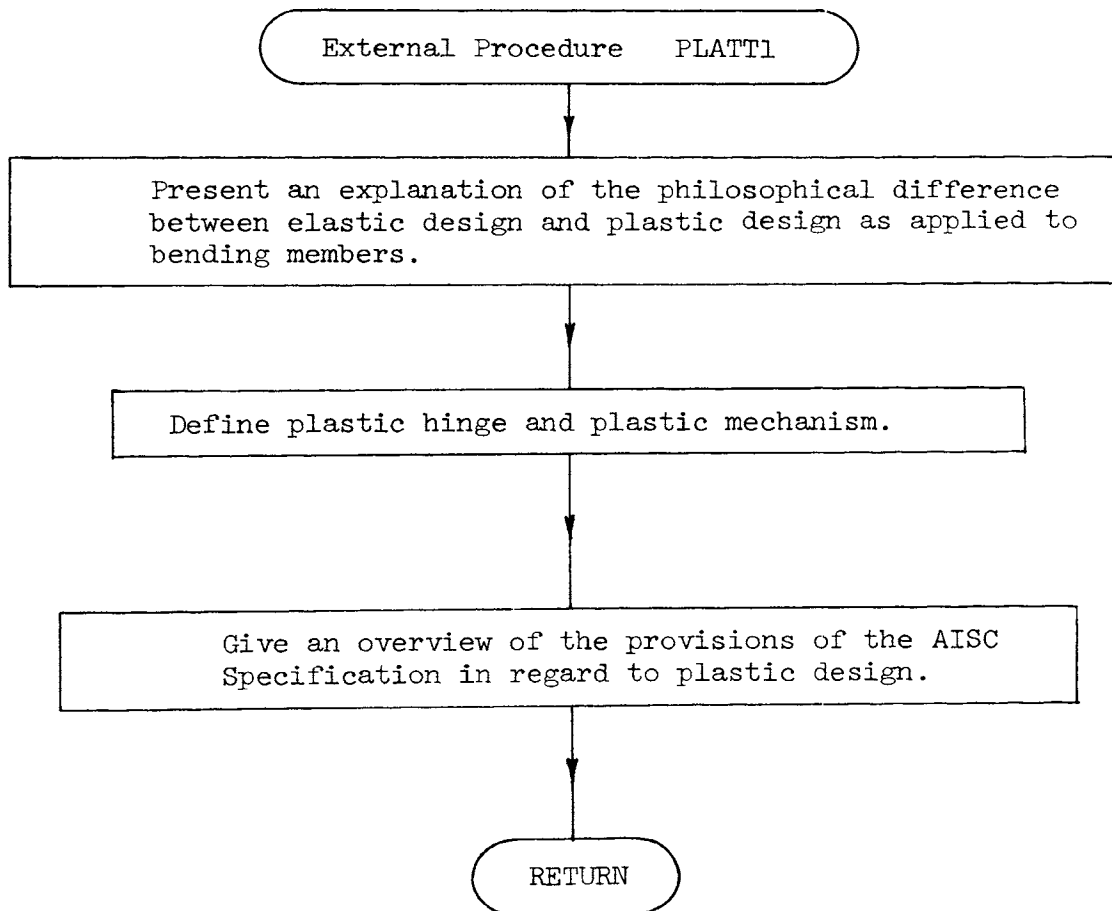
A beam-column design problem. The length between support in the strong and weak direction.

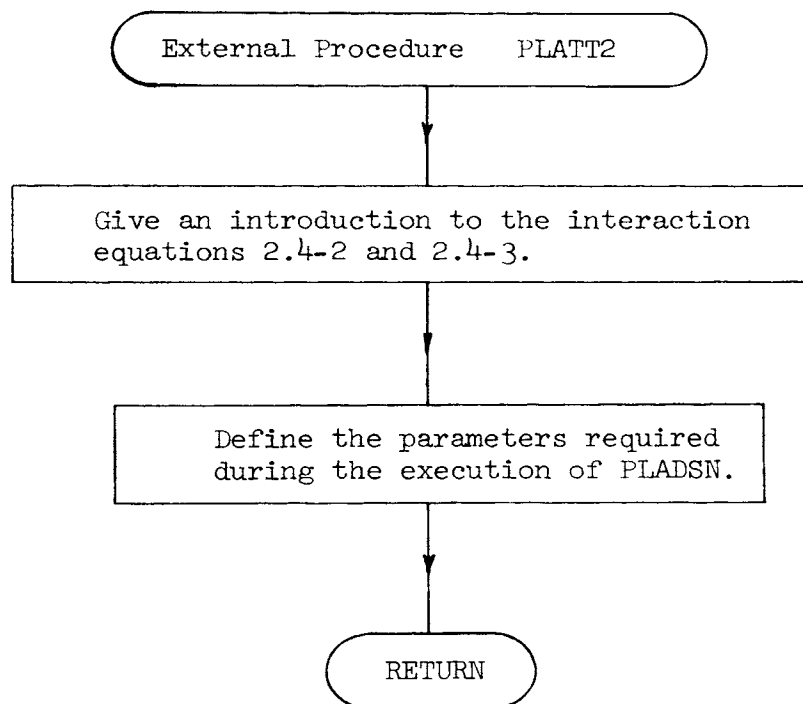
Overall Flow Chart PLADSN

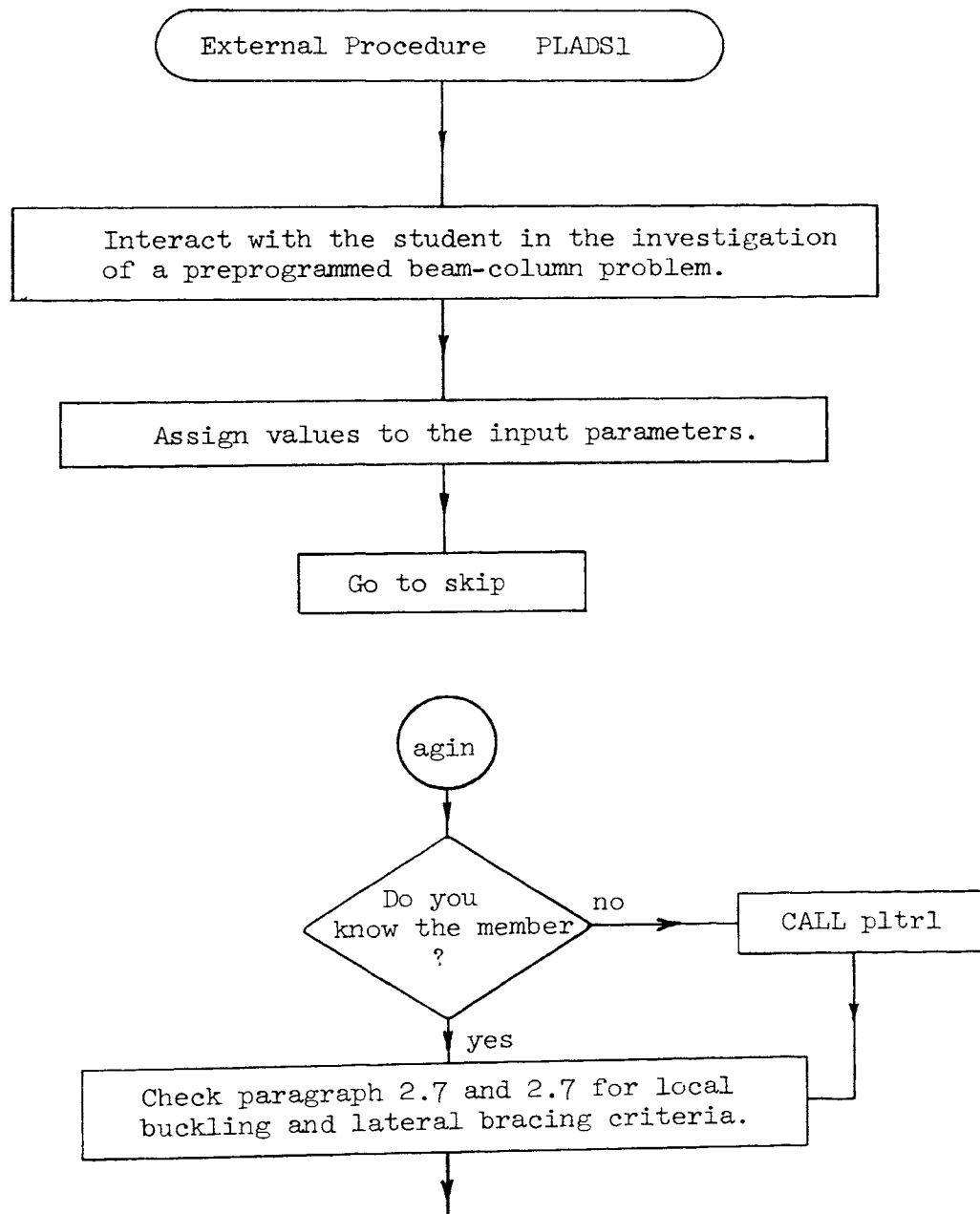


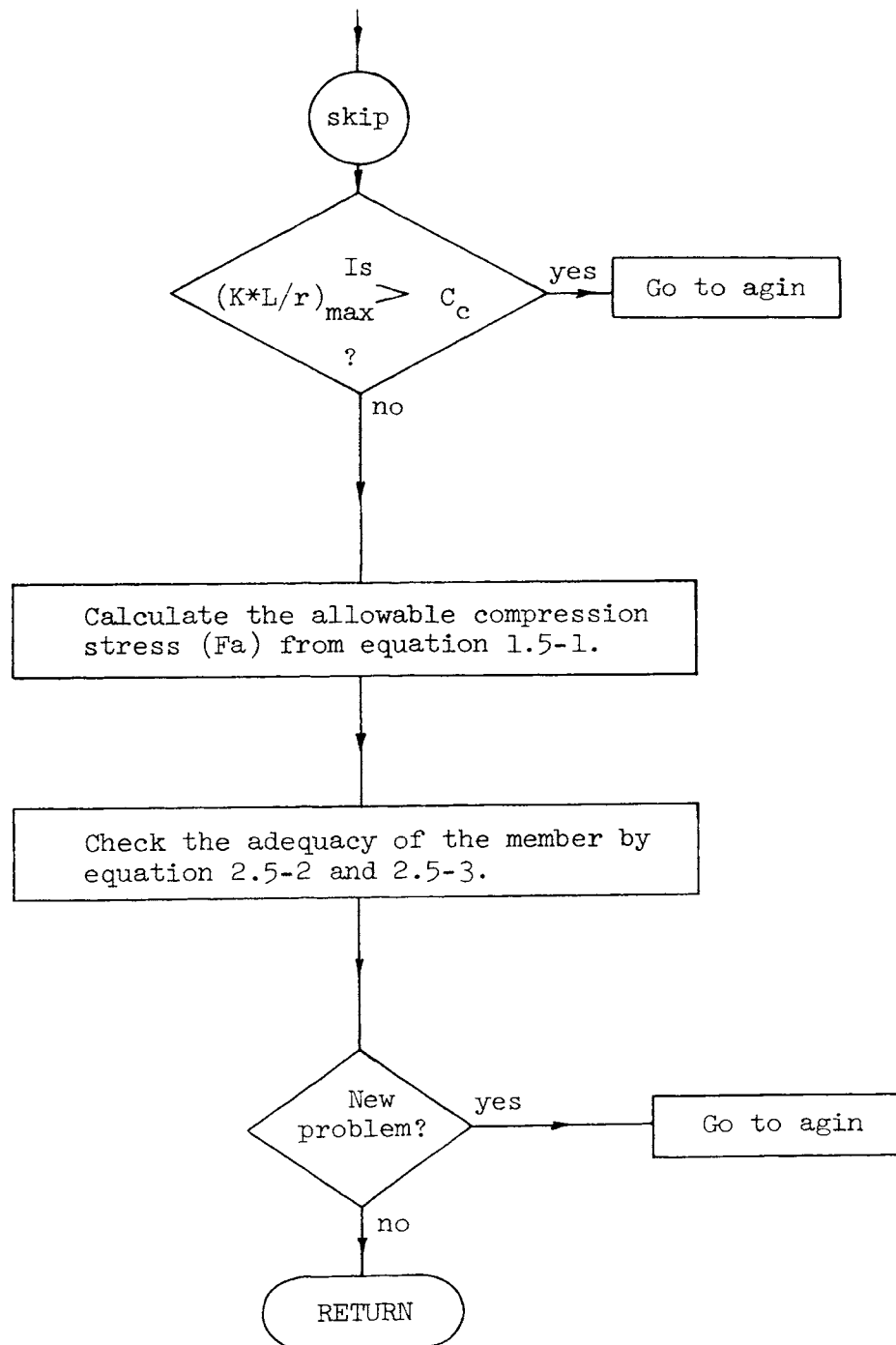
Detailed Flow Chart PLADSN

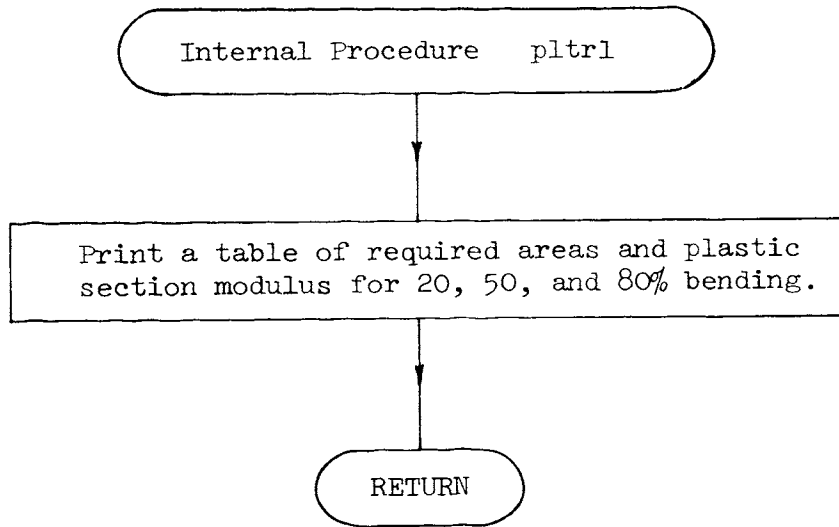












```
/* PROGRAM PLADSN */
DECLARE PLATT1 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE PLATT2 ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE PLADS1 ENTRY EXT KEY(whmIII) LIB(USER2);
PUT LIST('Would you like to review the basics of plastic design? (1 = yes
0 = no)');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO 1b1;
CALL PLATT1;
CALL PLATT2;
1b1: CALL PLADS1;
PUT LIST('This concludes PLADSN. ');
END ;
```

```

PLATT1: PROCEDURE ;
        PUT LIST(' ');
        PUT LIST('Program PLADSN will review with you plastic design of beams and beam-
        columns');
        PUT LIST('Before proceeding the user should first read Part 2 of the AISC
        Specifications');
        PUT LIST(' ');
        PUT LIST('Plastic design has a different philosophy than elastic design. To
        explain');
        PUT LIST('this difference we will consider bending. If no local buckling occur
        a bending');
        PUT LIST('member will resist no additional bending moment when the total cross
        section has');
        PUT LIST('reached the plastic or yield stress. The bending moment at this time
        is known');
        PUT LIST('as the plastic moment; and the point in the beam at which it forms is
        known as');
        PUT LIST('a plastic hinge. If the load is increased beyond this point a new
        plastic hinge');
        PUT LIST('will form at another point of high bending moment. This process
        continues until');
        PUT LIST('enough plastic hinges form to cause instability of the structure.
        This instability');
        PUT LIST('is called a collapse mechanism. The load at which this occurs is
        reduced by');
        PUT LIST('a factor of safety in plastic design to obtain the working load of the
        structure. ');
        PUT LIST('Note that for each plastic hinge to form no local or torsional
        buckling may occur. ');
        PUT LIST(' ');
        PUT LIST('In elastic design we apply a factor of safety to the yield stress and
        call this the');
        PUT LIST('allowable stress. The design of the structure is accomplished by
        keeping the stresses');

```

```

PUT LIST('below this allowable. This allowable stress is further reduced to
account');
PUT LIST('for local or torsional buckling.');
```

From the above the difference between the elastic and plastic design can be seen to be:

```

PUT LIST('one of physiological intent. In elastic design the designer intends
that the stresses');
PUT LIST('everywhere be below some set limit. In plastic design the designer
assures that no');
PUT LIST('local or lateral torsional buckling can occur and then designs based
on the load that');
PUT LIST('will cause collapse of the structure or collapse mechanism.');
```

To further emphasize the fact that in plastic design of beam the designer has only to:

```

PUT LIST('insure that no local or lateral torsional buckling will occur, look
at Section 2.7 and 2.9');
PUT LIST('of the AISC Specifications. Section 2.7 specifies the limit on
bf/2*tf and d/tw thus');
PUT LIST('controlling local buckling of the flanges and web. Section 2.9
limits the length between');
PUT LIST('bracing thus controlling the lateral torsional buckling. With these
provisions satisfied');
PUT LIST('the maximum moment the member can resist is the plastic moment.');
```

Since plastic design for bending consists of a simple check of local and lateral torsional buckling requirements let us now proceed to review plastic design of beam-columns.

```

RETURN ;
END PLATT1;
```

```

PLATT2:  PROCEDURE ;
          PUT LIST(' ');
          PUT LIST('Plastic design of beam-columns is basically the same as it is in
                    elastic design.');
```

That is, the design is satisfactory if an interaction equation is satisfied.

```

          PUT LIST(' There are two interaction equations which apply, Equation 2.4-2
                    and 2.4-3.');
```

These account for instability and load limitations respectively. They are

```

          PUT LIST(' similar to those in elastic design except that they are in terms
                    of factored');
```

load instead of stress.

```

          PUT LIST(' ');
          PUT LIST('Note that no accounting has been made for bending about the weak axis.
                    This is');
```

because simple plastic theory i.e. the use of plastic bending moment is not

```

          PUT LIST(' adequate for biaxial bending.');
```

The following parameters will be used during the execution of PLADSN:

```

          ');
          PUT LIST(' Fy - yield stress (ksi).');
```

Fa - allowable compressive stress defined by Eq. 1.5-1 (ksi).

```

          PUT LIST(' Fa - allowable compressive stress defined by Eq. 1.5-1 (ksi).');
```

Cm - a coefficient defined in Section 1.6.1

```

          PUT LIST(' Cm - a coefficient defined in Section 1.6.1');
```

Mp - Plastic moment (ft*K).

```

          PUT LIST(' Mp - Plastic moment (ft*K).');
```

M - applied factored bending moment (ft*K).

```

          PUT LIST(' M - applied factored bending moment (ft*K).');
```

(for calculation of factored load see Section 2.1)

```

          PUT LIST(' (for calculation of factored load see Section 2.1)');
```

PLADSN assumes bending about the strong axis.

```

          PUT LIST(' PLADSN assumes bending about the strong axis.');
```

P - applied factored axial load (Kips).

```

          PUT LIST(' P - applied factored axial load (Kips).');
```

(for calculation of factored load see Section 2.1)

```

          PUT LIST(' (for calculation of factored load see Section 2.1)');
```

K - effective length factor - input as 1.0 if Column is from a one or two story

```

          PUT LIST(' K - effective length factor - input as 1.0 if Column is from a one
                    or two story ');
```

unbraced planar frame - Ref. Table C 2.4.1.

```

          PUT LIST(' unbraced planar frame - Ref. Table C 2.4.1.');
```

AREA- cross section area (in**2).

```

          PUT LIST(' AREA- cross section area (in**2).');
```

```

PUT LIST(' rx - radius of gyration about the strong axis (in).');
PUT LIST(' ry - radius of gyration about the weak axis (in).');
PUT LIST(' Lx - length between bracing in the strong direction (ft).');
PUT LIST(' Ly - length between bracing in the weak direction (ft).');
PUT LIST(' Z - plastic section modulus (in**3).');
PUT LIST(' E - modulus of elasticity of steel (29,000ksi).');
PUT LIST(' ANSWER - usually requiring a yes or no indication 1 = yes 0 =
no. ');
PUT LIST(' IDENTIFICATION - anything may be typed for the future identification
of the problem. ');
PUT LIST(' ');
PUT LIST('During the execution of PLADSN if any value of input is to remain the
same ');
PUT LIST(' as that in the previous problem the value may be retyped or the
RETURN key ');
PUT LIST(' may be struck. ');
PUT LIST(' ');
PUT LIST('The logic flow of beam-column design or Paragraph 2.4 is best
explained by ');
PUT LIST(' working an investigation problem. ');
PUT LIST(' ');
RETURN ;
END PLATT2;

```



```

PLADS1:  PROCEDURE ;
          DECLARE t1 DEC(6), t2 DEC(6), t3 DEC(6), t5 DEC(6);
          DECLARE ttt CHAR(1);
          DECLARE P DEC(6), M DEC(6), Lx DEC(6), Ly DEC(6);
          DECLARE K DEC(6), rx DEC(6), ry DEC(6), Z DEC(6);
          DECLARE Fy DEC(6), AREA DEC(6), Cm DEC(6), Fa DEC(6);
          DECLARE Fe DEC(6), Pe DEC(6), Pcr DEC(6);
          DECLARE Cc DEC(6), Mm DEC(6), sum DEC(6), sum1 DEC(6);
          PUT LIST(' * * * * * IDENTIFICATION * * * * *');
          PUT LIST(' Investigation of a W14X184 P-255 M=850      unbraced in the weak
                    direction');
          P=255;
          M=850;
          Lx=14;
          Ly=14;
          K=1;
          rx=6.49;
          ry=4.04;
          Z=338;
          Fy=36;
          AREA=54.1;
          Cm=1;
          GO TO skip;
agin:    PUT LIST(' * * * * * IDENTIFICATION * * * * *');
          GET EDIT(ttt)(A(1));
          GET LIST(P,M,Fy,Lx,Ly);
          PUT LIST('Do you know the first trial member?');
          GET LIST(ANSWER);
          IF ANSWER=0 THEN CALL pltr1;
          GET LIST(ry,rx,Z,AREA,Cm,K);
          PUT LIST('Check width thickness ratio Section 2.7. ');
          PUT LIST('    is the bf/2tf provision satisfied');
          PUT LIST('    and is');
          IF P/(Fy*AREA)>.27 THEN t1=257/sqrt(Fy); ELSE t1=412/sqrt(Fy)*(1-1.4*P/
                    (Fy*AREA));

```

```

t2=P/(Fy*AREA);
PUT LIST('      d/t < ',t1,' based on P/Py = ',t2);
PUT LIST('Bracing requirements Section 2.9.');
```

$$t1 = (1375/Fy + 25) * r_y / 12$$

$$t2 = 1375 / Fy * r_y / 12$$

```

PUT LIST('      length between bracing shall be closer than');
PUT LIST('      ',t1,' ft for +1 > M/Mp > -0.5');
PUT LIST('      ',t2,' ft for -0.5 >= M/Mp > -1.');
```

```

PUT LIST('Are these provisions satisfied?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO skip;
PUT LIST('Choose a new member.');
```

```

GO TO agin;
skip: CALL print;
Cc=sqrt(2*9.86959*29000/Fy);
PUT LIST('Cc = (2*(pi)**2*E/Fy)**.5 = ',Cc);
t1=K*Lx*12/rx;
t2=Ly*12/ry;
PUT LIST('Slenderness ratio strong = ',t1,' weak = ',t2);
IF t1>=Cc THEN GO TO 1b1;
PUT LIST('Slenderness ratio strong axis greater than Cc try a new member.');
```

```

GO TO agin;
1b1: PUT LIST('Slenderness ratio strong is less than or equal to Cc. OK');
IF t1>t2 THEN GO TO 1b2;
PUT LIST('Weak axis controls the value of Fa');
GO TO 1b3;
```

```

1b2: PUT LIST('Strong axis controls the value of Fa.');
```

```

1b3: t1=max(t1,t2);
t2=t1**2;
t3=t2*t1;
c2=Cc**2;
c3=Cc**3;
Fa=(1-t2/(2*c2))*Fy/5/3+3*t1/(8*Cc)-53/(8*c3);
PUT LIST('Fa = ',Fa,' by Eq. 1.5-1');
```

```

Pcr=1.7*AREA*Fa;
PUT LIST('Pcr = 1.7*AREA*Fa = ',Pcr);
PUT LIST(' ');
t1=K*Lx*12/rx;
Fe=12*9.86959*29000/(23*t1**2);
Pe=23/12*AREA*Fe;
PUT LIST('F'e = 12*(pi)**2*E/(23*(KL/r)**2) = ',Fe);
PUT LIST('Pe = (23/12)*AREA*F'e = ',Pe);
PUT LIST('Is the column braced in the weak direction?');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO 1b4;
Mm=Fy*Z/12;
PUT LIST('Mm = Mp = Fy*Z = ',Mm,' ft*kips');
GO TO 1b5;
1b4: Mm=(1.07-Ly*12/ry*sqrt(Fy)/3160)*Fy*Z/12;
IF Mm Fy*Z/12 THEN Mm=Fy*Z/12;
PUT LIST('Mm = (1.07 - ((Ly/ry)*Fy**.5)/3160)*Mp/12 <= Mp = ',Mm,' ft*kips');
1b5: PUT LIST(' ');
PUT LIST('Interaction Equation 2.4-2');
t1=P/Pcr;
t2=Cm*M/((1-P/Pe)*Mm);
PUT IMAGE(t1,t2)(im1);
im1: IMAGE;
--.-- + --.-- < = ? 1.0
PUT LIST('Interaction Equation 2.4-3');
sum=Fy*AREA;
sum1=Fy*Z/12;
PUT LIST(' Py = Fy*AREA = ',sum);
PUT LIST(' Mp = Fy*Z/12 = ',sum1);
t3=P/(Fy*AREA);
t4=M/(1.18*Fy*Z/12);
PUT IMAGE(t3,t4)(im1);
sum=t1+t2;
sum1=t3+t4;

```

```

IF sum>sum1 THEN GO TO 1b6;
PUT LIST('Eq. 2.4-3 controls');
t1=t3;
t2=t4;
GO TO 1b7;
1b6: PUT LIST(' Eq 2.4-2 controls');
1b7: ssum=max(sum,sum1);
PUT LIST(' sum = ',sum);
t1=t1*100;
t2=t2*100;
PUT LIST('Axial load accounts for ',t1,' %');
PUT LIST('Bending accounts for ',t2,' %');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST('Do you want to try another problem?');
GET LIST(ANSWER);
IF ANSWER=1 THEN GO TO agin;
PUT LIST(' ');
RETURN ;
print: PROCEDURE ;
PUT LIST(' ');
PUT LIST(' ');
PUT LIST(' ');
PUT LIST('Fy = ',Fy);
PUT LIST(M = ',M,' P = ',P);
PUT LIST('Lx = ',Lx,' Ly = ',Ly);
PUT LIST('rx = ',rx,' ry,= ',ry);
PUT LIST('AREA = ',AREA,' Z = ',Z);
PUT LIST('Cm = ',Cm,' K = ',K);
PUT LIST(' * * * * *');
PUT LIST(' ');
RETURN ;
END print;
pltr1: PROCEDURE ;

```

```

PUT LIST(' ');
PUT LIST(' ');
PUT IMAGE(1)(im6);
PUT IMAGE(1)(im7);
lop1: DO i=2 TO 8 BY 3;
      j=i/10;
      jj=i*10;
      t1=P/((1-j)*Fy);
      t2=M*12/(j*Fy);
      PUT IMAGE(jj,t1,t2)(im8);
END lop1;
im6:  IMAGE;
      Percent of Member          Area          Z Plastic Section Modulus
im7:  IMAGE;
      Resisting Bending          (in**2)          (in**3)
im8:  IMAGE;
      --.-%          ---.-          -----
PUT LIST('Based on the above table choose a W member. ');
RETURN ;
END pltr1;
END PLADS1;

```

Program Name:

RIGID

Purpose:

Program RIGID can be used to analyze beams, plane frames and trusses. The members in the structure can be arranged in any general geometric pattern.

Limitations:

The structure may consist of a maximum of nine joints and eighteen members. The analysis is an elastic analysis. Loads and supports must be in the plane of the structure. No internal hinges are allowed.

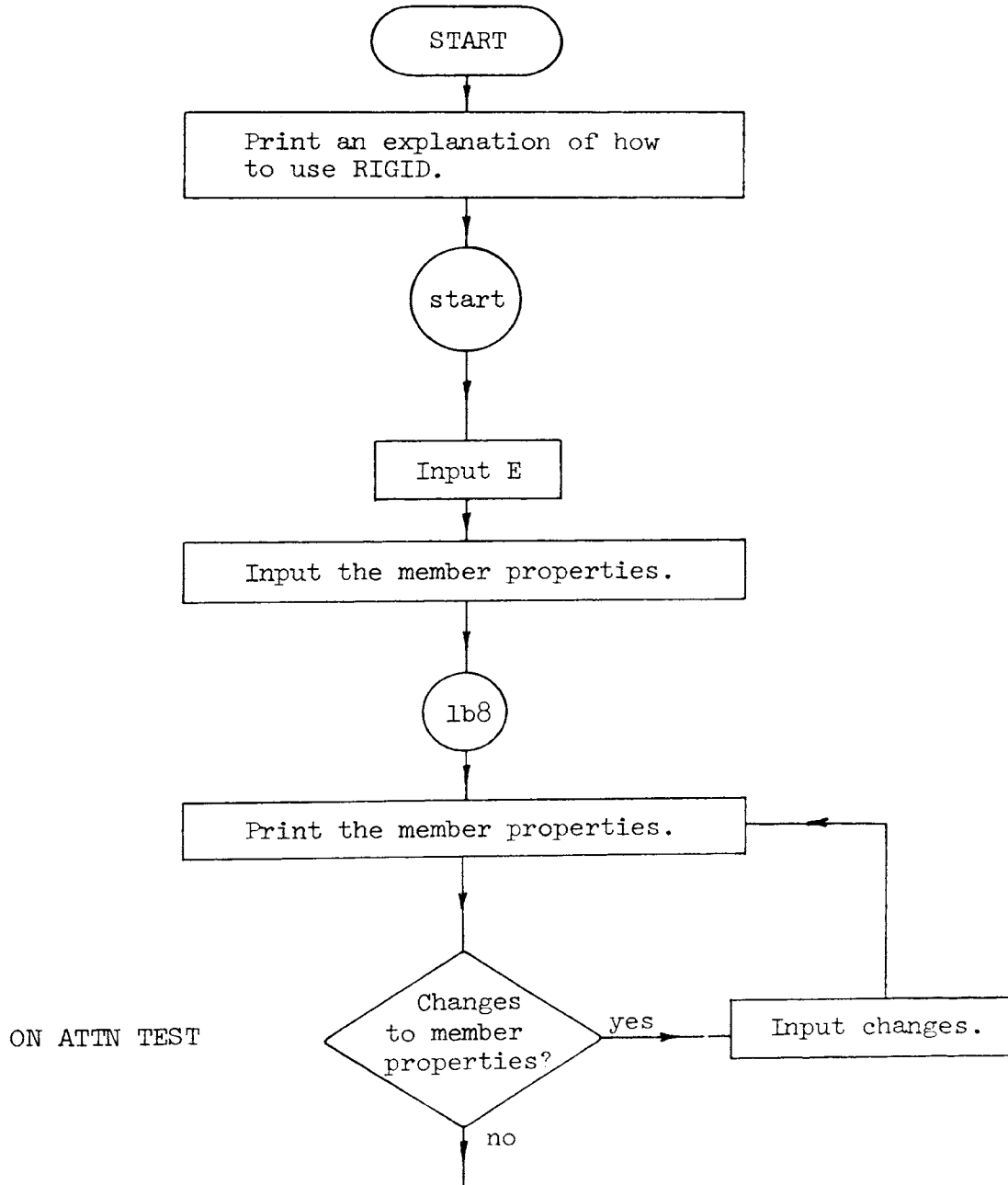
Prerequisite:

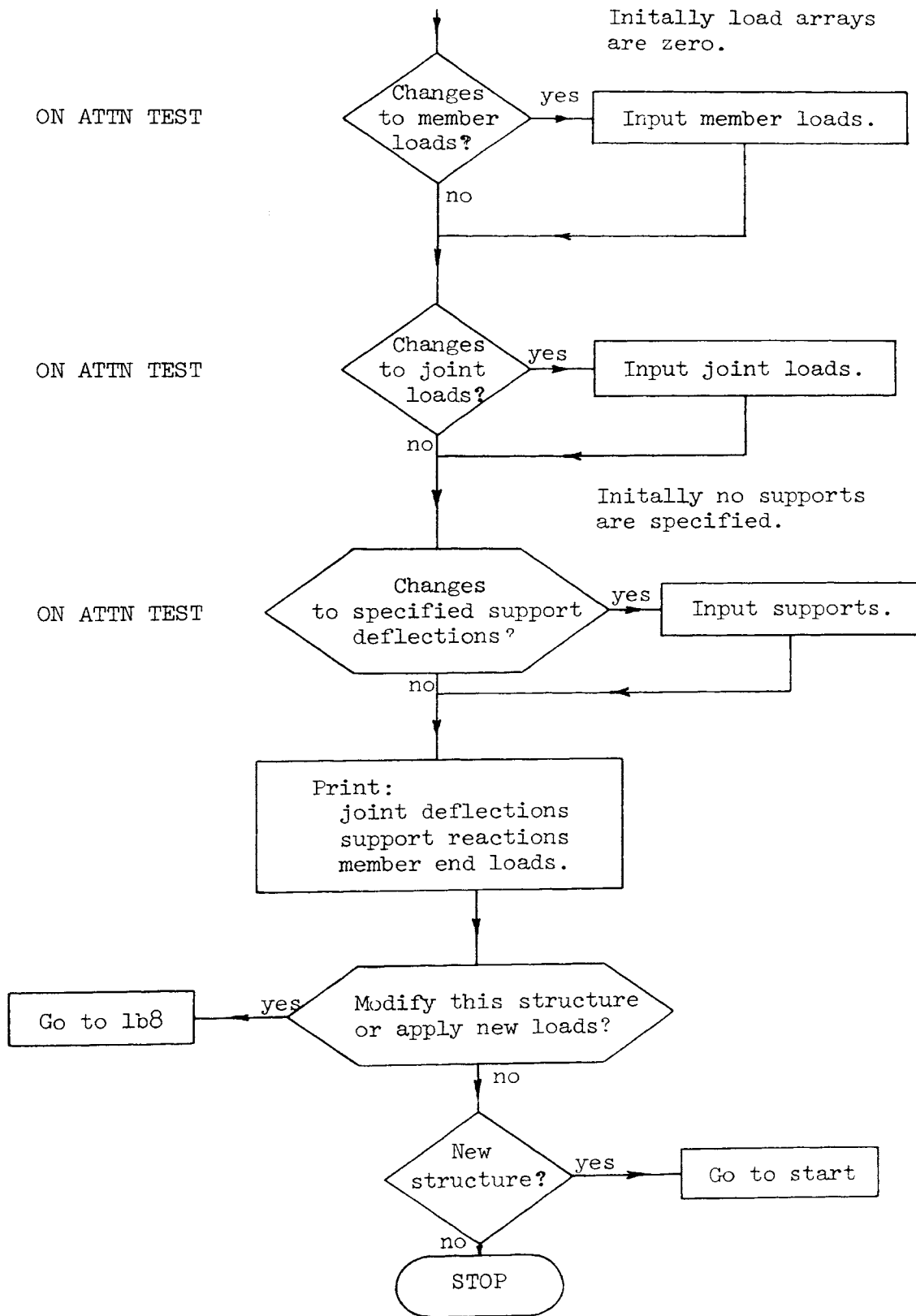
Knowledge of indeterminate structural analysis.

Data Required Before Execution:

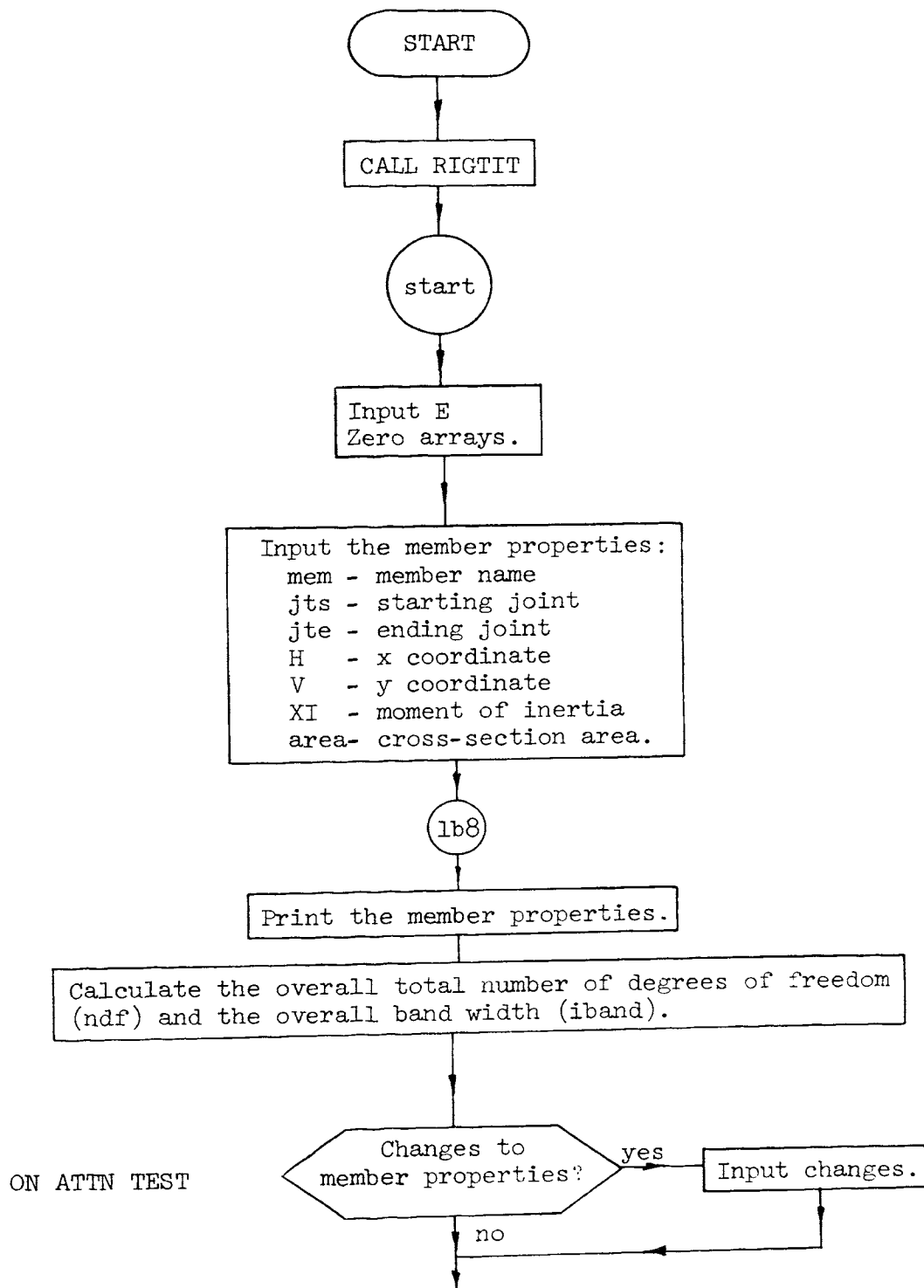
The joints of the structure should be numbered and the members named. A cross section area and moment of inertia (zero if a truss) for each member must be defined. The x and y dimension of each member should be calculated before execution of RIGID.

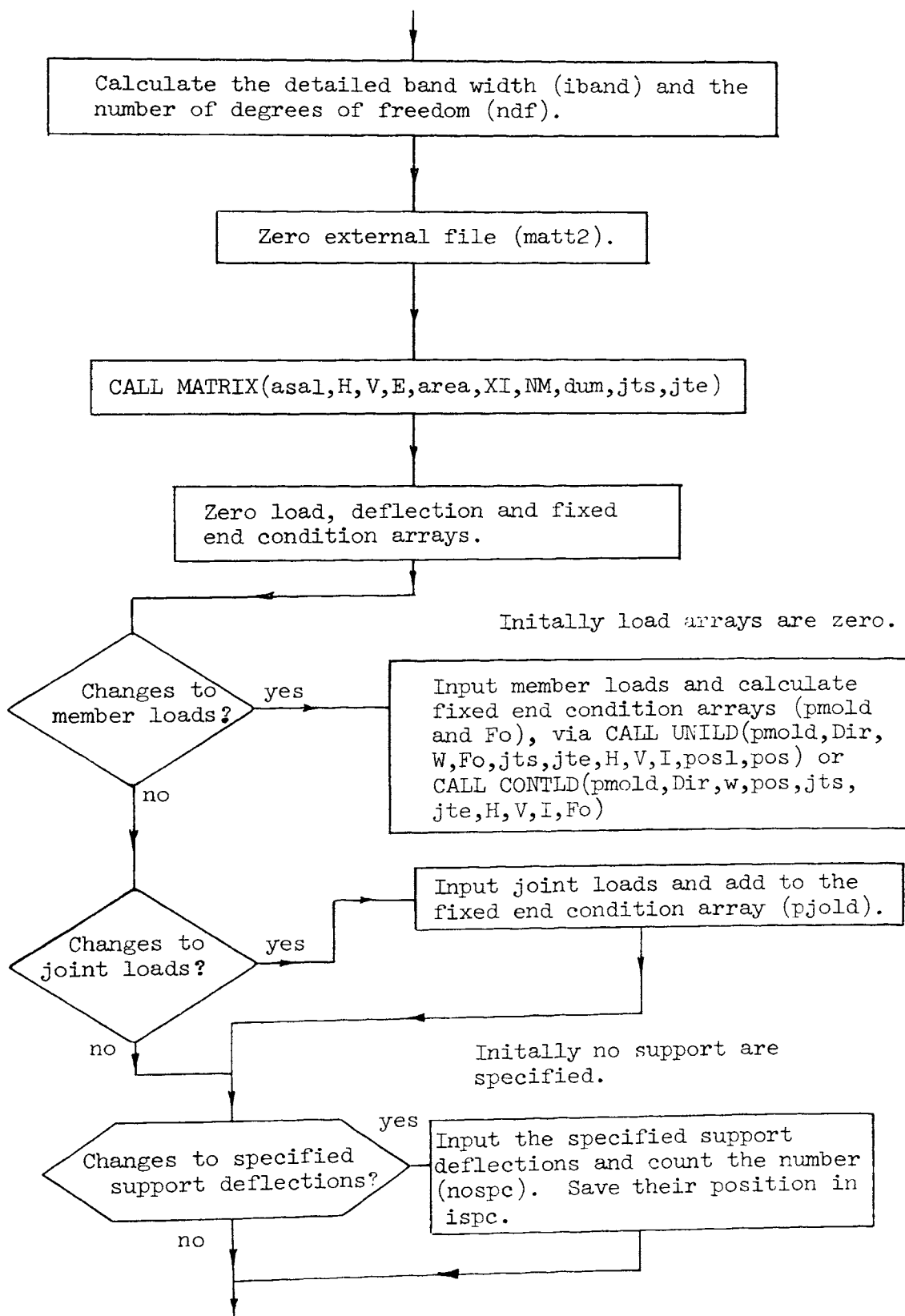
Overall Flow Chart RIGID

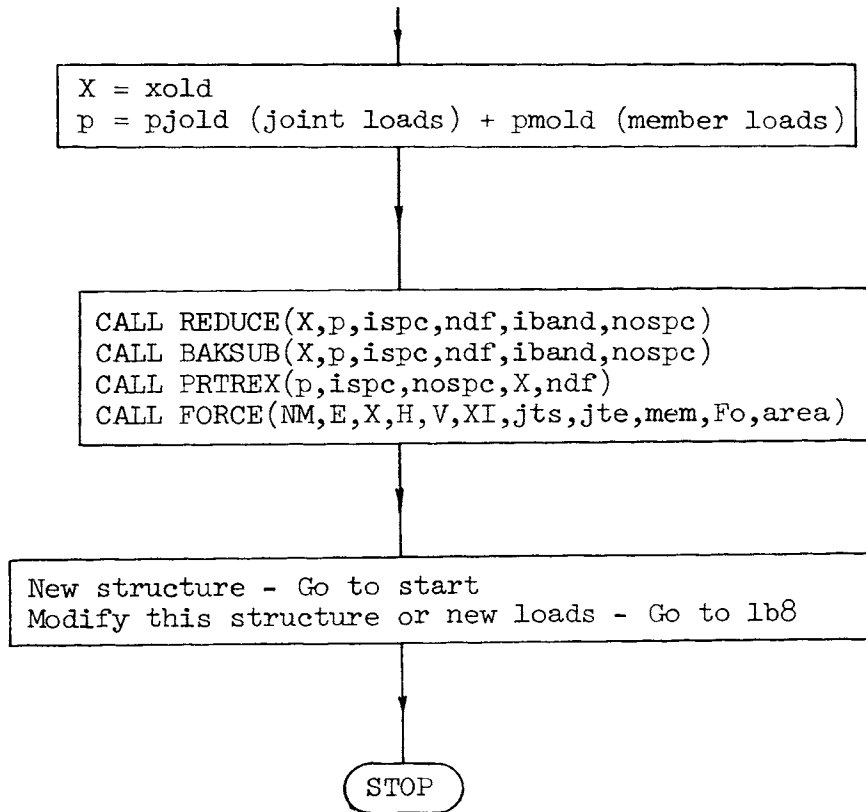


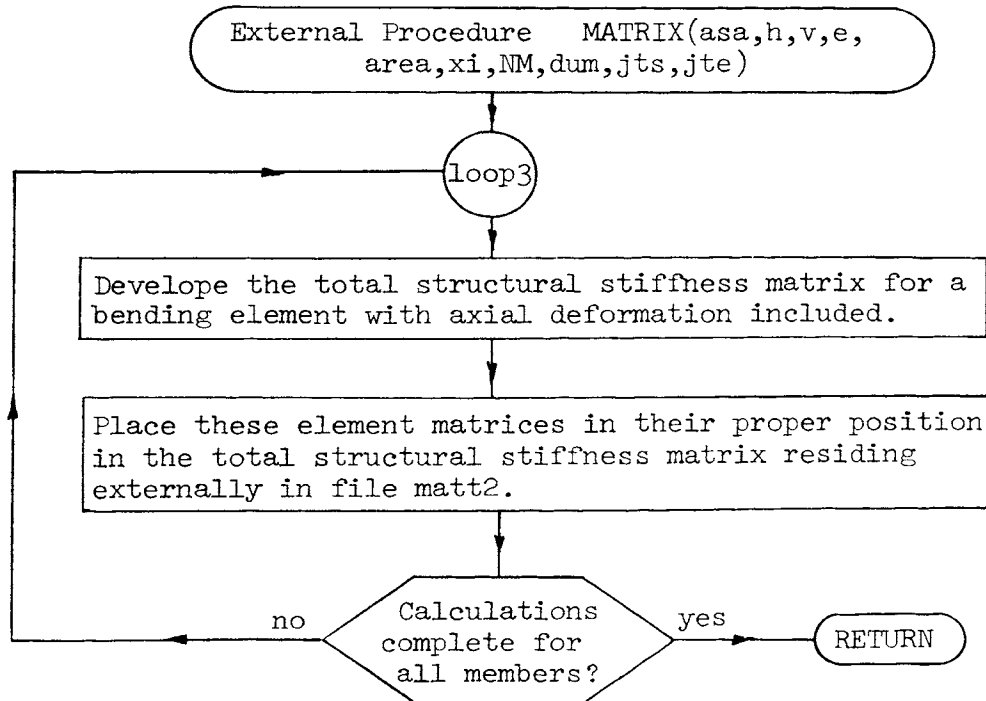
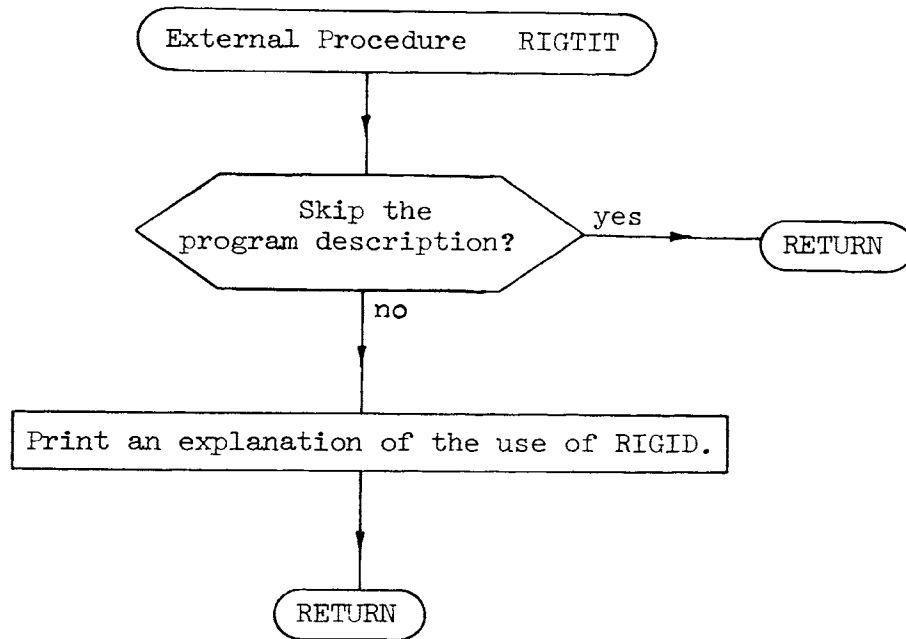


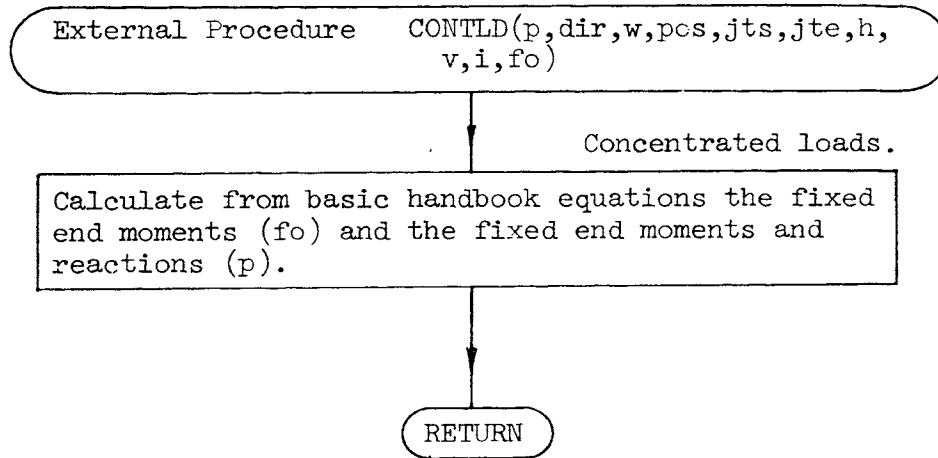
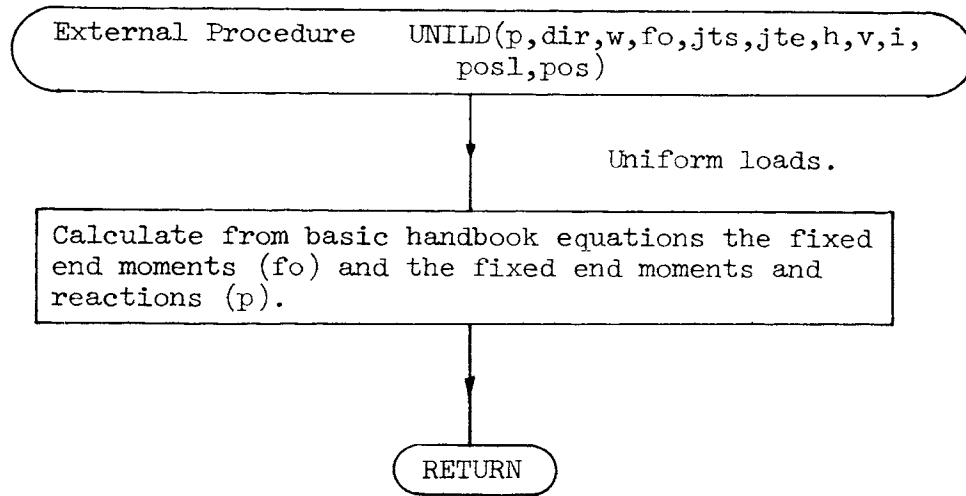
Detailed Flow Chart RIGID

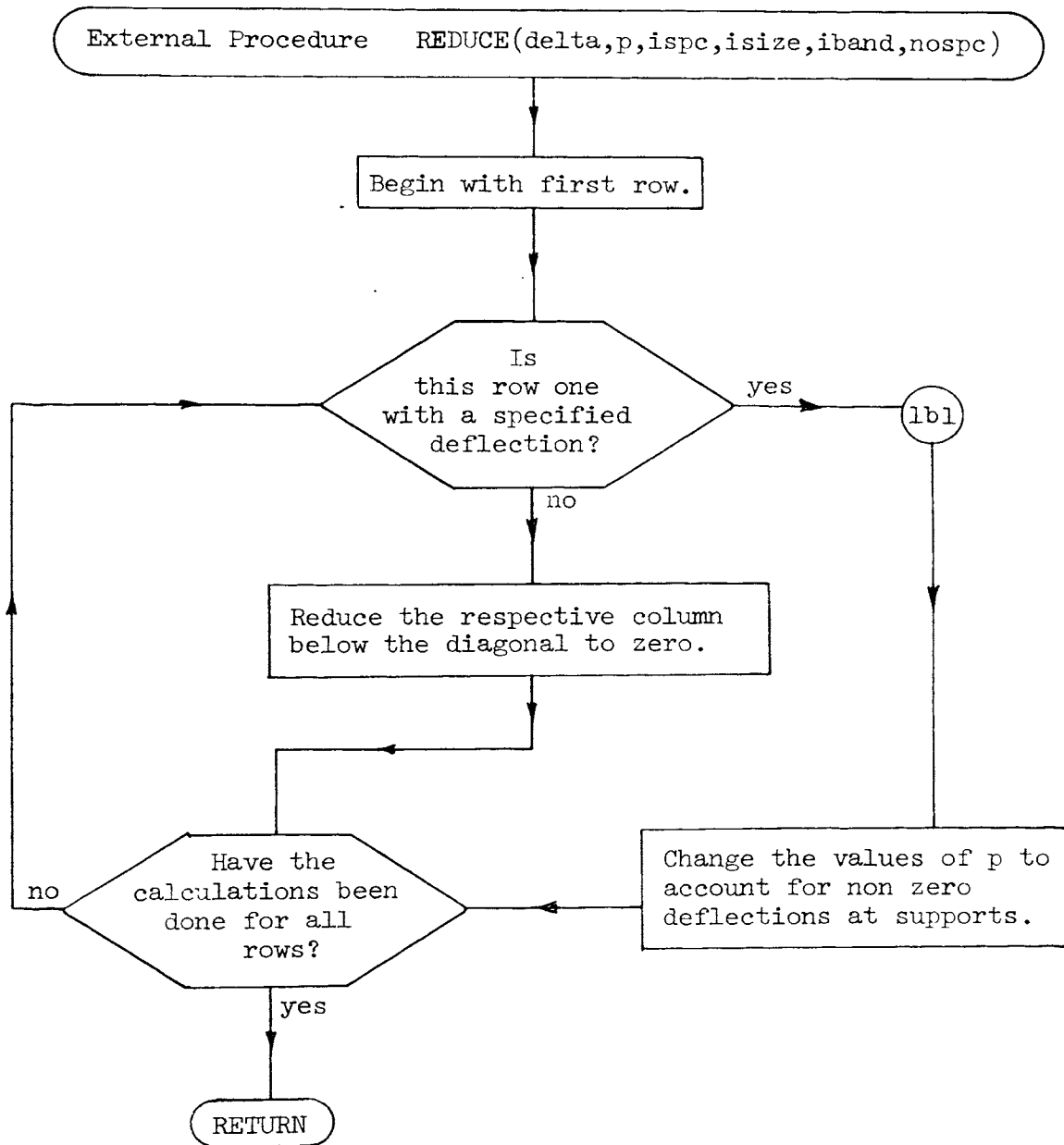


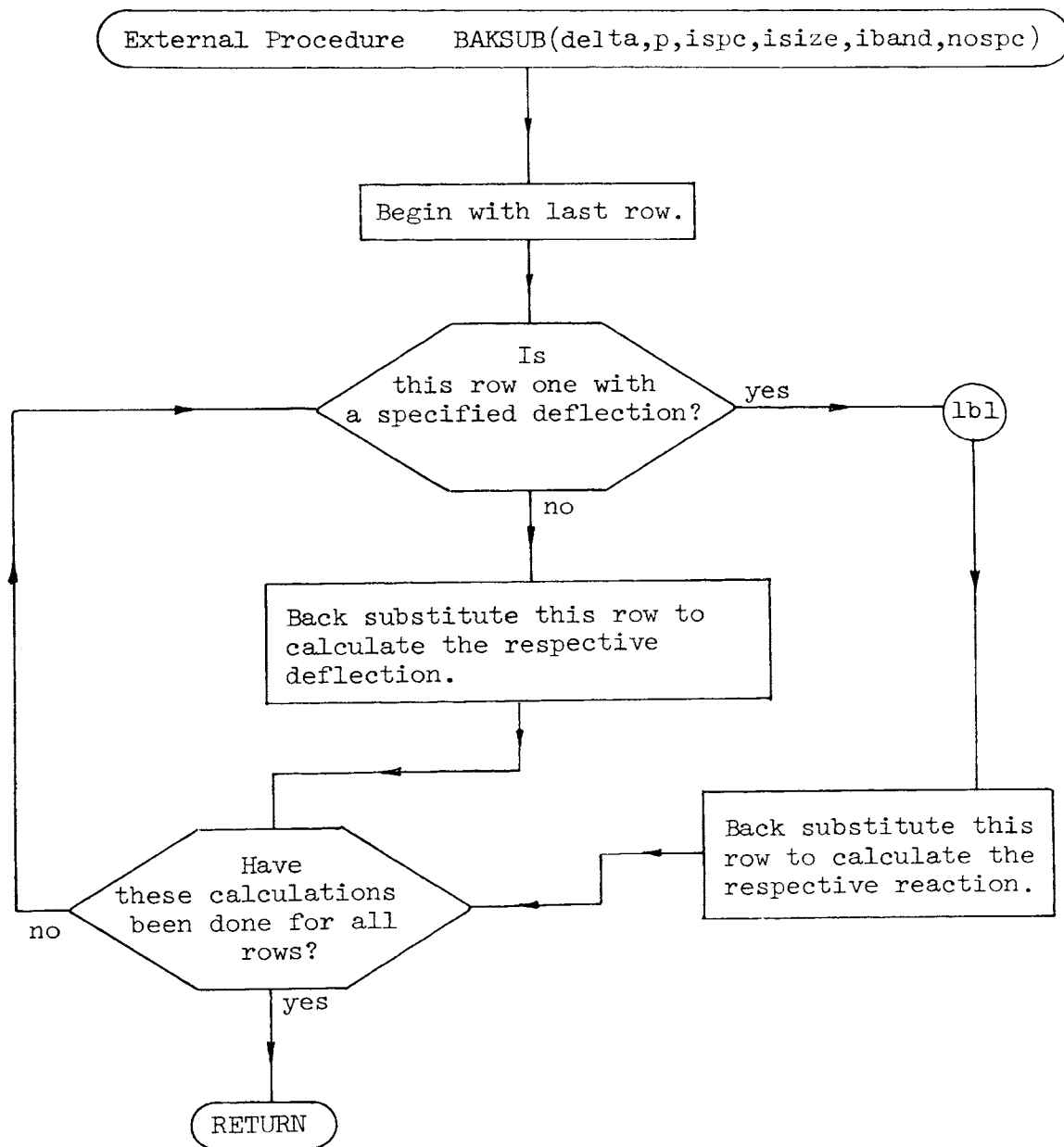


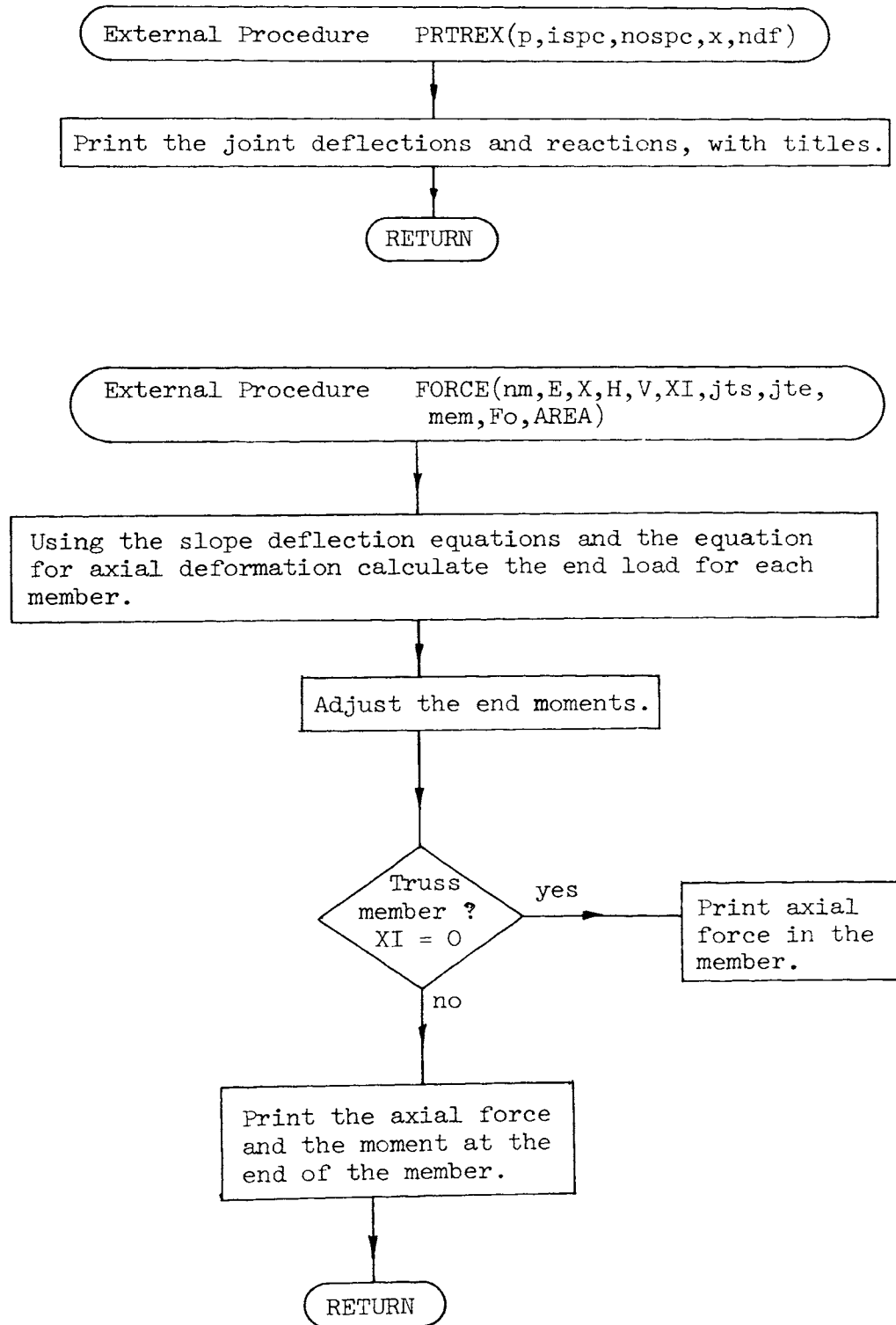













```

/* PROGRAM RIGID */
DECLARE matt2 FILE ENV( REGIONAL (1) F(216) ) ;
DECLARE type CHAR (1);
DECLARE ispc(10) DEC(6);
DECLARE dum(27) , asa1(6,6) ;
DECLARE E DEC(6), H(18) DEC(6), V(18) DEC(6), XI(18) DEC(6);
DECLARE jts(18) DEC(6), jte(18) DEC(6), mem(18) CHAR(3), Fo(18,2) DEC(6);
DECLARE CONTLD ENTRY EXT KEY(whmIII) LIB(USER2);
DECLCARE UNILD ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE X(27) , p(27) , area(18) DEC(6);
DECLARE xold(27) DEC(6), pmold(27) DEC(6), pjold(27) DEC(6);
DECLARE Dir CHAR(1);
DECLARE PRTREX ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE MATRIX ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE REDUCE ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE BAKSUB ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE FORCE ENTRY EXT KEY(whmIII) LIB(USER2);
DECLARE RIGTIT ENTRY EXT KEY(whmIII) LIB(USER2);
CALL RIGTIT;
PUT LIST(' ');
PUT LIST(' ');
start: PUT IMAGE(1)(im4);
PUT IMAGE(1)(im5);
ON ERROR MSG GO TO start;
GET EDIT(E)(X(1),F(8));
PUT IMAGE(1)(im1);
PUT IMAGE(1)(im2);
PUT IMAGE(1)(im3);
E=E*144;
XI=0;
area=0;
xold=0;
pmold=0;
pjold=0;
ON ATTENTION GO TO 1b8;
i=1;

```

```

err1:    i=i-1;
loop1:  i=i+1;
        ON ERROR MSG GO TO err1;
        NM=i-1;
        PUT LIST('Member',i);
        GET EDIT(mem(i),jts(i),jte(i),H(i),V(i),area(i),XI(i))(X(1),A(3),X(3),F(3),X(4),
          F(3),X(3),(4) F(11)));
        area(i)=area(i)/144;
        XI(i)=XI(i)/12**4;
        GO TO loop1;
1b8:    Dir=' ';
        ON ERROR SYSTEM;
        PUT EDIT(Dir)(SKIP(10),A(1));
        PUT IMAGE(1)(im1);
        PUT IMAGE(1)(im2);
        iband,ndf=0;
        k=12**4;
        l=144;
loop4:  DO i=1 TO NM;
        PUT LIST('Member',i);
        PUT IMAGE(mem(i),jts(i),jte,H(i),V(i),area(i)*l,XI(h)*k)(im15);
        /* Calculate the number of degrees of freedom ndf and iband */;
im15:  IMAGE;
        --- --- --- -----
        ndf=MAX(ndf,jts(i),jte(i));
        dif=ABS(jts(i)-jte(i));
        iband=MAX(dif,iband);
        END loop4;
        PUT LIST(' ');
        PUT LIST('Changes to Member Properties');
        ON ATTENTION GO TO 1b9;
        PUT IMAGE(1)(im1);
        PUT IMAGE(1)(im2);
        PUT IMAGE(1)(im3);
        ON ERROR MSG GO TO 1b10;

```

```

1b10:   GET LIST(Member);
        IF Member>NM THEN NM=Member;
        i=Member;
        GET EDIT(mem(i),jts(i),jte(i),H(i),V(i),area(i),XI(i))(X(1),A(3),X(3),F(3),
          X(4),F(3),X(3),(4) F(11)));
        area(i)=area(i)/144;
        XI(i)=XI(i)/12**4;
        ON ATTENTION GO TO 1b8;
        GO TO 1b10;
1b9:   iband=(iband+1)*3;
        ON ERROR SYSTEM;
        ndf=ndf*3;
        /* Zero file matt2 */;
        dum=0;
        OPEN FILE(matt2) SEQUENTIAL OUTPUT ;
loop2:  DO i=1 TO 27;
        WRITE FILE(matt2) FROM(dum) ;
        END loop2;
        CLOSE FILE(matt2) ;
        CALL MATRIX(asa1,H,V,E,area,XI,NM,dum,jts,jte);
        p=0;
        X=0;
        Fo=0;
        ON ATTENTION GO TO 1b7;
        PUT LIST('Member Loads');
        PUT IMAGE(1)(im12);
        PUT IMAGE(1)(im13);
        PUT IMAGE(1)(im14);
        i=1;
err2:  i=i-1;
1b5:  i=i+1;
        ON ERROR MSG GO TO err2;
        PUT LIST('Load ;,i);
        GET EDIT(I,type,Dir,W,pos,pos1)(X(1),F(2),X(6),A(1),X(9),A(1),X(6),(3) F(12));

```

```

IF type='c' THEN type='C';
IF Dir='y' THEN Dir='Y';
IF Dir='x' THEN Dir='X';
pos=abs(pos);
pos1=abs(pos1);
IF I>0 THEN GO TO 1b12;
pmold=0;
Fo=0;
GO TO 1b7;
1b12: IF i=1 THEN pmold=0;
      IF i=1 THEN Fo=0;
      IF type='C' THEN GO TO 1b6;
      CALL UNILD(pmold,Dir,W,Fo,jts,jte,H,V,I,pos1,pos);
      GO TO 1b5;
1b6:  CALL CONTLD(pmold,Dir,W,pos,jts,jte,H,V,I,Fo);
      GO TO 1b5;
1b7:  ON ATTENTION GO TO 1b2;
      PUT LIST('Joint Loads');
      PUT IMAGE(1)(im6);
      PUT IMAGE(1)(im7);
      PUT IMAGE(1)(im8);
      ON ATTENTION GO TO 1b2;
      i=1;
err3: i=i-1;
1b1:  i=i+1;
      ON ERROR MSG GO TO err3;
      PUT LIST('Load ',i);
      GET EDIT(I,Dir,W)(X(1),F(3),X(7),A(1),X(7),F(7));
      IF I>0 THEN GO TO 1b13;
      pjold=0;
      GO TO 1b2;
1b13: IF i=1 THEN pjold=0;
      IF Dir='X' THEN I=I*3-1;
      IF Dir='x' THEN I=I*3-1;

```

```

IF Dir='Y' THEN I=I*3;
IF Dir='y' THEN I=I*3;
IF Dir='R' THEN I=I*3-2;
IF Dir='r' THEN I=I*3-2;
pjold(I)=pjold(I)+W;
GO TO lb1;
1b2: /* Specified Deflections */;
ON ATTENTION GO TO lb4;
PUT LIST('Specified Deflections');
PUT IMAGE(1)(im9);
PUT IMAGE(1)(im10);
PUT IMAGE(1)(im11);
ON ATTENTION GO TO lb4;
i=1;
err4: i=i-1;
1b3: i=i+1;
ON ERROR MSG GO TO err4;
PUT LIST('Spec. Def. ',i);
GET EDIT(I,Dir,W)(X(1), F(3),X(7),A(1),X(7),F(7));
IF i=1 THEN xold=0;
IF i=1 THEN nospc=0;
nospc=nospc+1;
IF Dir='X' THEN I=I*3-1;
IF Dir='x' THEN I=I*3-1;
IF Dir='Y' THEN I=I*3;
IF Dir='y' THEN I=I*3;
IF Dir='R' THEN I=I*3-2;
IF Dir='r' THEN I=I*3-2;
IF Dir='X' THEN W=W/12;
IF Dir='x' THEN W=W/12;
IF Dir='Y' THEN W=W/12;
IF Dir='y' THEN W=W/12;
xold(I)=xold(I)+W;
ispc(nospc)=I;
GO TO lb3;
1b4: ON ATTENTION SYSTEM;
ON ERROR SYSTEM;

```

```

X=xold;
p=pjold+pmold;
CALL REDUCE(X,p,ispc,ndf,iband,nospc);
CALL BAKSUB(X,p,ispc,ndf,iband,nospc);
CALL PRTREX(p,ispc,nospc,X,ndf);
CALL FORCE(NM,E,X,H,V,XI,jts,jte,mem,Fo,area);
PUT LIST('New structure = 1   Modify this Structure or new loads = 2   STOP
        = 0');
GET LIST(ANSWER);
IF ANSWER=0 THEN GO TO enn;
IF ANSWER=1 THEN GO TO start; ELSE GO TO lb8;
enn:  PUT LIST('Execution Complete');
      END ;
im1:  IMAGE;
      Mem.  Start  End.    X corrd  Y coord    Area      I
im2:  IMAGE;
      Name Joint  Joint   (ft.)    (ft.)      (in**2)   (in**4)
im3:  IMAGE;
      |***| |***|  |***|  |*****|  |*****|  |*****|  |*****|
im4:  IMAGE;
      Modulus of Elasticity E(ksi)
im5:  IMAGE;
      |*****|
im6:  IMAGE;
      Joint  Direction  Joint Load
im7:  IMAGE;
      No.    (X,Y or R)  (K or K*ft)
im8:  IMAGE;
      |***|      |*|      |*****|
im9:  IMAGE;
      Joint  Direction  Spec. Def.
im10: IMAGE;
      No.    (X,Y or R) (in or rad)
im11: IMAGE;
      |***|      |*|      |*****|
im12: IMAGE;

```

	Mem.	Type	Direction	Magnitude	Perpendicular distance from start. jt to	
	No.	(U or C)	(X or Y)	(K or K/ft)	Start	End of load. (ft)
im13:	IMAGE;					
im14:	IMAGE;					
	**	*	*	*****	*****	*****

```

RIGITIT: PROCEDURE ;
ON ATTENTION GO TO enn;
PUT LIST('To skip the printing of the program description depress the ATTN
key. ');
PUT LIST(' ');
PUT LIST('Problem Description: Before input to RIGID can begin the user must
number the');
PUT LIST('joints of the structure and give up to a three character alphaneric
name to');
PUT LIST('each member. A member connects two joints and a joint must be
specified');
PUT LIST('at supports, points between members and changes in section. ');
PUT LIST(' ');
PUT LIST('Sing Convention: The sign convention used by RIGID is the standard
X-Y system, ');
PUT LIST('+Y is up +X is to the right. This convention is applicable when
defining the');
PUT LIST('member lengths and loads. For member lengths, the coordinates are
set at the');
PUT LIST('starting joint. For loads, the direction defines the sign. Positive
rotation');
PUT LIST('and moment is clockwise. ');
PUT LIST(' ');
PUT LIST('Rules for Input: ');
PUT LIST(' 1. Values of input must be typed in the columns indicated by
astericks. The');
PUT LIST(' only exception to this is in the Changes to Member Properties.
In this');
PUT LIST(' section the member number, as defined by the computer, is
requested by');
PUT LIST(' Member. The user types the member number then the RETURN key
and then the');
PUT LIST(' new member properties in the respective columns defined by
astericks. ');

```



```

PUT LIST(' 2. After one line has been typed the balance of the line must
be spaced out');
PUT LIST('      past the last asterisk, and the RETURN key struck. If the
line is acceptable,');
PUT LIST('      the terminal will respond with a request for the next line of
input. If');
PUT LIST('      an error occurs, and error message will be printed and a
request for a repeat');
PUT LIST('      of correct information will follow. If neither of the above
occurs not enough');
PUT LIST('      spaces followed the last item of input. To correct this,
space several times');
PUT LIST('      and then strike the RETURN key. ');
PUT LIST(' 3. The ATTN key causes the computer to pass to the next area of
input. It is used');
PUT LIST('      after all items of input in an area have been accepted by
the computer. ');
PUT LIST(' 4. If the user elects to modify the structure or loads the ATTN
key struck when the');
PUT LIST('      first line of input is requested will leave that area of data
unchanged. ');
PUT LIST(' 5. A line of blank spaces as the first line of input causes that
area of input to');
PUT LIST('      be set to zero, ie all member loads are zero etc. If one
item of an area is to');
PUT LIST('      change the whole area must be retyped. On the first pass all
load data is zero. ');
RETURN ;
END RIGTIT;

```

enn:

```

MATRIX:  PROCEDURE (asa,h,v,e,area,xi,NM,dum,jts,jte);
          DECLARE matt2 FILE ENV( REGIONAL(1) F(216) ) ;
          OPEN FILE(matt2) DIRECT UPDATE ;
          c=1;
loop3:   DO i1=1 TO NM;
          x1=sqrt(h(i1)**2+v(i1)**2);
          sin1=v(i1)/x1;
          cos1=h(i1)/x1;
          ea=area(i1)*e/x1;
          ei1=e*xi(i1)/x1;
          ei2=3i1/x1;
          ei3=ei2/x1;
          asa(1,1)=4*ei1;
          asa(1,2)=6*ei2*sin1*c;
          asa(1,3)=-6*ei2*cos1*c;
          asa(1,4)=asa(1,1)/2;
          asa(1,5)=-asa(1,2);
          asa(1,6)=-asa(1,3);
          asa(2,1)=asa(1,2);
          asa(2,2)=ea*cos1**2+12*ei3*sin1**2;
          asa(2,3)=ea*sin1*cos1-12*ei3*sin1*cos1;
          asa(2,4)=asa(1,2);
          asa(2,5)=-asa(2,2);
          asa(2,6)=-asa(2,3);
          asa(3,1)=asa(1,3);
          asa(3,2)=asa(2,3);
          asa(3,3)=ea*sin1**2+12*ei3*cos1**2;
          asa(3,4)=asa(1,3);
          asa(3,5)=-asa(2,3);
          asa(3,6)=-asa(3,3);
          asa(4,1)=asa(1,4);
          asa(4,2)=asa(2,4);
          asa(4,3)=asa(3,4);
          asa(4,4)=asa(1,1);

```

```

asa(4,5)=-asa(1,2);
asa(4,6)=-asa(1,3);
asa(5,1)=asa(1,5);
asa(5,2)=asa(2,5);
asa(5,3)=asa(3,5);
asa(5,4)=asa(4,5);
asa(5,5)=asa(2,2);
asa(5,6)=asa(2,3);
asa(6,1)=asa(1,6);
asa(6,2)=asa(2,6);
asa(6,3)=asa(3,6);
asa(6,4)=asa(4,6);
asa(6,5)=asa(5,6);
asa(6,6)=asa(3,3);
ii=(jts(i1)-1)*3+1;
jj=(jte(i1)-1)*3+1;
l=0;
lop1: DO k=1 TO 2;
      IF k=1 THEN iii=ii; ELSE iii=jj;
lop2: DO i=iii TO iii+2;
      l=l+1;
      READ FILE(matt2) INTO(dum) KEY(i) ;
      jjj=0;
lop3: DO j=ii TO ii+2;
      jjj=jjj+1;
      dum(j)=dum(j)+asa(l,jjj);
      END lop3;
lop4: DO j=jj TO jj+2;
      jjj=jjj+1;
      dum(j)=dum(j)+asa(l,jjj);
      END lop4;
      REWRITE FILE(matt2) FROM(dum) KEY(i) ;
      END lop2;
      END lop1;
      END loop3;

```

```
CLOSE FILE(matt2) ;  
RETURN ;  
END MATRIX;
```

```

UNILD:  PROCEDURE (p,dir,w,fo,jts,jte,h,v,i,pos1,pos);
        ii=jts(i)*3-2;
        jj=jte(i)*3-2;
        b=pos1-pos;
        IF dir='X' THEN GO TO stmt1;
        sign=h(i)/abs(h(i));
        h1=abs(h(i));
        h2=h1-pos1;
        const=b/(24*h1**2);
        const=const*(b**2*(h1+3*(h2-pos))-24*(h2+b/2)**2*(pos+b/2));
        const=abs(const);
        p(ii)=p(ii)-w*const*sign;
        fo(i,1)=fo(i,1)+w*const*sign;
        const=b/(24*h1**2);
        const=const*(b**2*(h1+3*(pos-h2))-24*(pos+b/2)**2*(h2+b/2));
        const=abs(const);
        p(jj)=p(jj)+w*const*sign;
        fo(i,2)=fo(i,2)-w*const*sign;
        const=b/(4*h1**3);
        const=const*(4*(h2+b/2)**2*(h1+2*(pos+b/2))-b**2*(h2-pos));
        const=abs(const);
        p(ii+2)=p(ii+2)+w*const;
        const=b/(4*h1**3);
        const=const*(4*(pos+b/2)**2*(h1+2*(h2+b/2))-b**w*(pos-h2));
        const=abs(const);
        p(jj+2)=p(jj+2)+w*const;
        RETURN ;
        /* Horizontal uniform load */;
stmt1:  sign=v(i)/abs(v(i));
        h1=abs(v(i));
        h2=h1-pos1;
        const=b/(24*h1**2);
        const=const*(b**2*(h1+3*(h2-pos))-24*(h2+b/2)**2*(pos+b/2));
        const=abs(const);
        p(ii)=p(ii)+w*const*sign;

```

```

fo(i,1)=fo(i,1)-w*const*sign;
const=b/(24*h1**2);
const=const*(b**2*(h1+3*(pos-h2))-24*(pos+b/2)**2*(h2+b/2));
const=abs(const);
p(jj)=p(jj)-w*const*sign;
fo(i,2)=fo(i,2)+const*sign;
const=b/(4*h1**3);
const=const*(4*(h2+b/2)**2*(h1+2*(pos+b/2))-b**2*(h2-pos));
const=abs(const);
p(ii+1)=p(ii+1)+w*const;
const=b/(4*h1**3);
const=const*(4*(pos+b/2)**2*(h1+2*(h2+b/2))-b**2*(pos-h2));
const=abs(const);
p(jj+1)=p(jj+1)+w*const;
RETURN ;
END UNILD;

```

```

CONTLD:  PROCEDURE (p,dir,w,pos,jts,jte,h,v,i,fo);
          ii=jts(i)*3-2;
          jj=jte(i)*3-2;
          IF dir='X' THEN GO TO stmt1;
          sign=h(i)/abs(h(i));
          h1=abs(h(i));
          h2=h1**2;
          h3=h1**3;
          b=h1-pos;
          p(ii)=p(ii)-w*pos*b*b/h2*sign;
          p(jj)=p(jj)+w*pos*pos*b/h2*sign;
          p(ii+2)=p(ii+2)+w*b**2*(h1+2*pos)/h3;
          p(jj+2)=p(jj+2)+w*pos**2*(h1+2*b)/h3;
          fo(i,1)=fo(i,1)+w*pos*b*b/h2*sign;
          fo(i,2)=fo(i,2)-w*pos*pos*b/h2*sign;
          RETURN ;
          /* horizontal concentrated loads */;
stmt1:   sign=v(i)/abs(v(i));
          h1=abs(v(i));
          h2=h1**2;
          h3=h1**3;
          b=h1-pos;
          p(ii)=p(ii)+w*pos*b*b/h2*sign;
          p(jj)=p(jj)-w*pos*pos*b/h2*sign;
          p(ii+1)=p(ii+1)*w*b**2*(h1+2*pos)/h3;
          p(jj+1)=p(jj+1)+w*pos**2*(h1+2*b)/h3;
          fo(i,1)=fo(i,1)-w*pos*b*b/h2*sign;
          fo(i,2)=fo(i,2)+w*pos*pos*b/h2*sign;
          RETURN ;
          END ;

```

```

REDUCE:  PROCEDURE (delta,p,isper,ysize,iband,nospc);
          DECLARE matt2 FILE ENV( REGIONAL(1) F(216) ) ;
          DECLARE k(27) , kn(27) ;
          OPEN FILE(matt2) DIRECT UPDATE ;
loop4:   DO i=1 TO ysize;
          READ FILE(matt2) INTO(k) KEY(i) ;
          /* is delta(i) specified ? */;
loop1:   DO j=1 TO nospc;
          IF ispc(j)=i THEN GO TO lb1;
          END loop1;
          IF k(i)=0 THEN GO TO lb4;
          /* delta(i) not specified. */;
          /* reduce column i to zero. */;
loop3:   DO j=1 TO iband-1;
          ii=i+j;
          /* ii is the row number. */;
          IF ii>ysize THEN GO TO lb4;
          READ FILE(matt2) INTO(kn) KEY(ii) ;
          con=kn(i)/k(i);
loop2:   DO jj=i TO i+iband-1;
          /* jj is the column number. */;
          IF jj>ysize THEN GO TO lb3;
          kn(jj)=kn(jj)-k(jj)*con;
          END loop2;
lb3:     REWRITE FILE(matt2) FROM(kn) KEY(ii) ;
          p(ii)=p(ii)-p(i)*con;
lb2:     END loop3;
          GO TO lb4;
lb1:     /* delta(i) specified. */;
          IF delta(i)=0 THEN GO TO lb4;
          /* change the values of p(). */;
loop5:   DO j=i TO iband+i-1;
          IF j>ysize THEN GO TO lb4;
          p(j)=p(j)-delta(i)*k(j);

```



```
1b4:      END loop5;  
        END loop4;  
        CLOSE FILE(matt2) ;  
        RETURN ;  
        END ;
```

```

BAKSUB: PROCEDURE (delta,p,ispce,ysize,iband,nospc);
DECLARE matt2 FILE ENV( REGIONAL(1) F(216) );
DECLARE k(27) ;
OPEN FILE(matt2) DIRECT INPUT ;
lop4: DO i=1 TO ysize;
/* ii is the row number starting with the last. */;
ii=ysize+1-i;
READ FILE(matt2) INTO(k) KEY(ii);
IF k(ii)=0 THEN GO TO lb3;
/* is ii a specified deflection? */;
lop1: DO j=1 TO nospc;
IF ispce(j)=ii THEN GO TO lb1;
END lop1;
/* ii is not a specified deflection calculate delta(ii). */;
sum=0;
lop2: DO j=1 TO iband-1;
jj=ii+j;
IF jj>ysize THEN GO TO lb2;
sum=sum+k(jj)*delta(jj);
END lop2;
lb2: delta(ii)=(p(ii)-sum)/k(ii);
GO TO lb3;
lb1: /* ii is a specified deflection calculate the reaction ie p(ii). */;
sum=0;
lop3: DO j=1 TO iband-1;
jj=ii+j;
IF jj>ysize THEN GO TO lb4;
sum=sum+k(jj)*delta(jj);
END lop3;
lb4: p(ii)=-p(ii)+sum;
lb3: END lop4;
RETURN ;
END ;

```

```

PRTREX:  PROCEDURE (p, ispc, nospc, x, ndf);
          DECLARE dir CHAR(1);
          PUT LIST(' ');
          PUT LIST('Deflections (in or rad)');
          PUT LIST('Joint #   Rotation   Def. X       Def. Y');
          j=0;
          DO i=1 TO ndf BY 3;
          j=j+1;
          PUT EDIT(j,x(i),x(i+1)*12,x(i+2)*12)(F(6),(3) (X(2),E(10,3)));
          END ;
          PUT LIST(' ');
          PUT LIST('Reactions (k or K*ft)');
          PUT LIST('Joint #   Direction   Force or Moment');
          DO i=1 TO nospc;
          j=mod(ispc(i),3);
          IF j=0 THEN dir='Y';
          IF j=2 THEN dir='X';
          IF j=1 THEN dir='M';
          j=ispc(i);
          k=ceil(ispc(i)/3);
          PUT IMAGE(k,dir,p(j))(im1);
          END ;
im1:     IMAGE;
          ---
          RETURN ;
          END ;

```

```

FORCE:  PROCEDURE (nm,E,X,H,V,XI,jts,jte,mem,Fo,AREA);
        DECLARE f(3) DEC(4);
        PUT LIST(' ');
        PUT LIST('Member End Loads (K or K*ft)');
        PUT LIST('Member Axial Load Joint Moment Joint Moment');
        PUT LIST(' (average)');
1op1:   DO i=1 TO nm;
        ii=jts(i)*3-2;
        jj=jte(i)*3-2;
        x1=sqrt(H(i)**2+V(i)**2);
        EA=E*AREA(i)/x1;
        EI1=E*XI(i)/x1;
        EI2=EI1/x1;
        sin1=V(i)/x1;
        cos1=H(i)/x1;
        f(1)=-EA*cos1*x(ii+1)-EA*sin1*x(ii+2);
        f(1)=f(1)+EA*cos1*x(jj+1)+EA*sin1*x(jj+2);
        f(2)=4*EI1*x(ii)+6*EI2*sin1*x(ii+1);
        f(2)=f(2)-6*EI2*cos1*x(ii+2)+2*EI1*x(jj);
        f(2)=f(2)-6*EI2*sin1*x(jj+1)+6*EI2*x(jj+2)*cos1;
        f(3)=f(2)-2*EI1*x(ii)+2*EI1*x(jj);
        f(2)=f(2)+Fo(i,1);
        f(3)=f(3)+Fo(i,2);
        IF XI(i)=0 THEN GO TO 1b2;
        PUT EDIT(mem(i),f(1),jts(i),f(2),jte(i),f(3))(x(2),A(3),F(10,2),x(4),(2) (x(4),
        F(2),x(4),F(10,2),x(4)));
        GO TO 1b1;
1b2:   PUT EDIT(mem(i),f(1))(x(2),A(3),F(10,2));
1b1:   END 1op1;
        RETURN ;
        END ;

```