

AN INQUIRY INTO THE PROBLEM OF PREDICTING  
UNIVERSITY CLASSROOM REQUIREMENTS

by

THOMAS EDWIN COMPTON

B. S., Kansas State University, 1970

9984

A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Industrial Engineering

KANSAS STATE UNIVERSITY  
Manhattan, Kansas

1972

Approved by:

L. E. Grosh  
Major Professor

LD  
2668  
T4  
1972  
C66  
copy 2

## TABLE OF CONTENTS

	Page
CHAPTER 1 PROBLEM STATEMENT AND DISCUSSION OF LITERATURE.....	1
1.1 Introduction .....	1
1.2 Statement of Problem .....	1
1.3 Literature .....	2
1.3.1 Some General Papers .....	3
1.3.2 Some Work at Berkeley .....	3
1.3.3 Illinois State University .....	3
1.3.4 Michigan State University .....	4
1.3.5 The Toronto Group .....	4
1.3.6 The Work at Purdue .....	5
1.3.7 WICHE's Efforts .....	5
1.4 Organization of Report .....	6
CHAPTER 2 THE INDUCED COURSE LOAD MATRIX AND ITS CONSTRUCTION .....	9
2.1 The Induced Course Load Matrix .....	9
2.1.1 What Is An ICLM? .....	9
2.1.2 Some Definitions and Explanations .....	10
2.1.3 A Note on the Distribution of the Student Population .....	12
2.1.4 Other Considerations .....	12
2.2 Development of Kansas State's ICLM .....	13
2.2.1 Preliminary Considerations .....	13
2.2.2 Construction of the Matrix .....	16



	Page
CHAPTER 3 COURSE DATA, SECTIONING RULES, AND A PROJECTION .....	20
3.0 Introduction .....	20
3.1 Course Data .....	20
3.1.1 Preliminary Discussion and Definitions .....	20
3.1.2 Course Data Required .....	23
3.1.3 Possible Sources of Course Data .....	24
3.1.4 Course Data Compiled for Kansas State .	25
3.1.5 Method of Compiling Course Data .....	26
3.2 Sectioning Rules .....	31
3.3 Validating the Course Data and Sectioning Rules .....	35
3.3.1 Actual Distribution of Weekly Room Hours .....	36
3.3.2 A Projection Using the Course Data and Sectioning Rules .....	36
3.3.3 Evaluation of Projected Distribution of Weekly Room Hours .....	38
CHAPTER 4 RESULTS AND CONCLUSIONS .....	60
4.1 Introduction .....	60
4.2 The Induced Course Load Matrix .....	60
4.3 The Course Facility Requirements Data .....	61
4.4 The Sectioning Rules .....	62
4.5 Validating the Course Data and Sectioning Rules .....	63
4.6 Summary of Results .....	63
LIST OF REFERENCES .....	65

## APPENDIX

## Page

Appendix A: Program Documentation .....	67
Appendix B: Permanent Data Files .....	153

## CHAPTER 1

### PROBLEM STATEMENT AND DISCUSSION OF LITERATURE

#### 1.1 INTRODUCTION.

University planners are investigating ways to better utilize current facilities and forecast future facilities requirements. This thesis investigates one method for evaluating facilities requirements. Specifically, it investigates the problem of determining the sizes and the numbers of classrooms which are needed to meet the demands imposed by a given student population. Such information is necessary for investigating the effects on facilities requirements of, for example, shifts in the composition of the student body from one school or department to another. This problem is related to the problem of projecting facilities requirements for libraries, dormitories, and office space and to the problem of projecting future faculty requirements. The work undertaken here limits itself to investigating one method for predicting classroom requirements and does this within the framework of Kansas State University's data base.

#### 1.2 STATEMENT OF THE PROBLEM.

Briefly, the problem investigated was to find and evaluate a method of predicting classroom requirements. For investigative purposes, it was assumed that the composition of the student population would be available as input. Ideally, the output from this method would be the number and size of classrooms needed to meet these student requirements.

The preceding paragraph is a general statement of the problem. However, after reading the literature on this subject, two points were recognized. First, no organization had undertaken a full-scale investigation of this problem, although pilot studies had been made in this area (8,9,10). Second, one method in particular stood out as being worth further investigation. This was the method proposed by the Western Interstate Commission for Higher Education (WICHE) (12). Thus the problem was amended to investigating WICHE's method for predicting classroom requirements and to making this investigation on a full-scale, University wide basis.

Besides investigating WICHE's method, two other results were expected. First, an investigation of Kansas State's data base was to be made to determine what data was available and to evaluate it with respect to its suitability for use as input to WICHE's projection methods. The second result expected was the creation of an induced course load matrix (ICLM). This matrix was proposed by WICHE and was to be used in their procedure for predicting classroom requirements from a projection of the student population. This last result was to be one of the primary contributions of this investigation.

The investigation of WICHE's method for predicting classroom requirements, the investigation of Kansas State's data base, and the creation of an ICLM were the problems to be solved.

### 1.3 LITERATURE.

Several groups have worked in the general area of facilities planning. The sections which follow identify these groups and discuss the work they have done related to facilities planning.

### 1.3.1 SOME GENERAL PAPERS.

Bowman (1) and Hirsch (2) discussed two topics related to the general problem of educational planning. Bowman made a simulation of the cash flows of several budgets at Yale University. His work with this model influenced some of the budgetary decisions there. Hirsch discussed how the demand for education could be estimated. These papers discussed some aspects of educational planning, and they gave some idea of the scope of the thinking in this field. However, they are not directly related to the subject of facilities planning.

### 1.3.2 SOME WORK AT BERKELEY.

Marshall and Oliver (3) have worked on projecting student enrollments. They developed a model of undergraduate student attendance which attempts to model how undergraduates progress through the university. Their model was based on five parameters, one of which was the total work,  $w$ , required to complete a degree. Smith (4) discussed how varying the mix of classroom sizes could help accommodate the demands for classroom space imposed by courses. In this discussion he assumed a probability distribution for course enrollments and gave the probability of a given section size overflowing as a function of classroom size mix. He made the assumption that the probability distributions for course enrollments were known. He went on to test his method by assuming a normal distribution for course enrollments with the parameters determined from historical data. The author remarked that he had difficulty finding the necessary data at the Berkeley campus of the University of California.

### 1.3.3 ILLINOIS STATE UNIVERSITY.

Harden and Tchong (5) investigated two difficulties which arise when projecting student enrollments using Markovian models. The two difficulties investigated were the enlargement of the projection base due to the creation

of new departments and the restriction of enrollments by imposing departmental limits. They resolved these difficulties by developing a two-step Markovian model. They used their model to help establish guidelines for the admission of students.

#### 1.3.4 MICHIGAN STATE UNIVERSITY.

Koenig and Keeney (6) discussed a mathematical model they developed for allocating educational resources. In this article they, in very general terms, attempted to show the interactions between resources such as personnel, space, and equipment and the products of a university such as manpower, research, and public or technical services. A computer program was discussed which demonstrated some of these interactions. Theirs was a state-space model describing "...the behavioral characteristics of a system as a set of relationships among time-functions representing its inputs, outputs, and internal state." (6, p 201) Koenig (7) again in general terms, developed an input-output model of various sectors of an educational institution. A sample implementation in the academic production sector (faculty allocations) was presented.

#### 1.3.5 THE TORONTO GROUP.

Judy and Levine (8) discuss their development of a simulation model of a university. Their model "...accepts descriptions of the university's structure and statements of the levels of activities that the university is expected to perform." (8, p vii) With these inputs their model computes "...the resulting resource requirements of staff, space, materials, and money." (8, p vii) Their model is divided into four main sections. The first section uses a projected entering enrollment as input and from this computes the enrollees in each college and each course of study. From this it goes on to compute the subject-enrollees for each department. The next section uses these outputs as its inputs

and produces "...a rather complete description of the staff requirements of each department, important departmental parameters not set by explicit decision, the number of sections of pass and honors lectures and laboratories, and so on." (8, p 21) The third section computes such information as space requirements and gives this information in terms of square feet needed for a department's activities. The last part of the simulation computes budget requirements for departments. Parameters for controlling these procedures are supplied exogenously.

Two problems were apparent if this simulation model were to be used for the problem of projecting classroom demands. First, the authors did not give the details on how they process their inputs to give their outputs. Second, their outputs in the area of space requirements are in terms of total square feet. Total square feet does not indicate the number and size of required classrooms. Overall, their work appears to be too large an aggregation for the problem at hand.

#### 1.3.6 THE WORK AT PURDUE.

Blakesly and others at Purdue University have, for a number of years, been working on the solution of educational planning problems. They have done such work as computerize their class assignments and utilization review techniques and have undertaken the problem of master schedule construction. This latter undertaking is reported in (10). A Comprehensive University Scheduling System (CUSS) (10) attempts to allocate staff and room resources to meet course and student requirements and thus form a master schedule. They have attempted a full-scale simulation of master schedule construction, assignment of projected student requests to classes, and the production of complete review information.

#### 1.3.7 WICHE'S EFFORTS.

The Western Interstate Commission for Higher Education (WICHE) has been prolific in its writing of literature on



educational management and planning. Two of their efforts found useful are (11, 12). The first gives an overview of procedures that have been useful for projecting student enrollments. The second is their collection of space analysis manuals. This series of manuals gives their recommendations for planning and evaluation of classrooms and class laboratories, office and research facilities, academic support facilities, general institutional facilities, and their recommendations on other general topics related to facilities planning and evaluation.

WICHE's recommendations are explicit and realistic. They have the proper level of aggregation and seem well thought out. This is why their method of investigating facilities requirements was undertaken. The WICHE method for projecting facilities such as classrooms is given in Manual Two and discussion of an induced course load matrix is given in Manual Six. Their method starts with a distribution of the student population and, using an induced course load matrix, determines the projected course enrollments for this student population. These course enrollments are then used in conjunction with data on course facility requirements to determine what classroom facilities are required for each course. Overall facilities requirements are determined by summarizing the facilities requirements for each course. The concept of this method is simple. However, it had not been tested on a full-scale basis. Investigating the WICHE method on a full-scale basis with respect to Kansas State University was the problem undertaken.

#### 1.4 ORGANIZATION OF REPORT.

The rest of this report will be composed of three additional chapters and appendices. Chapter 2 will contain a discussion of the induced course load matrix and subjects related to it. Chapter 3 will discuss the problems encountered in deriving the course data and forming the



sectioning rules. Chapter 4 will give the results and conclusions of this investigation. The appendices will contain such items as computer program listings and the particulars on the data files created by these programs.

Figure 1.1 shows the separate parts of the investigation and their relation to one another. The course data and the sectioning rules are subject to review and manipulation by the administration; they are the parameters of the method.

**THIS BOOK  
CONTAINS  
NUMEROUS PAGES  
WITH DIAGRAMS  
THAT ARE CROOKED  
COMPARED TO THE  
REST OF THE  
INFORMATION ON  
THE PAGE.**

**THIS IS AS  
RECEIVED FROM  
CUSTOMER.**

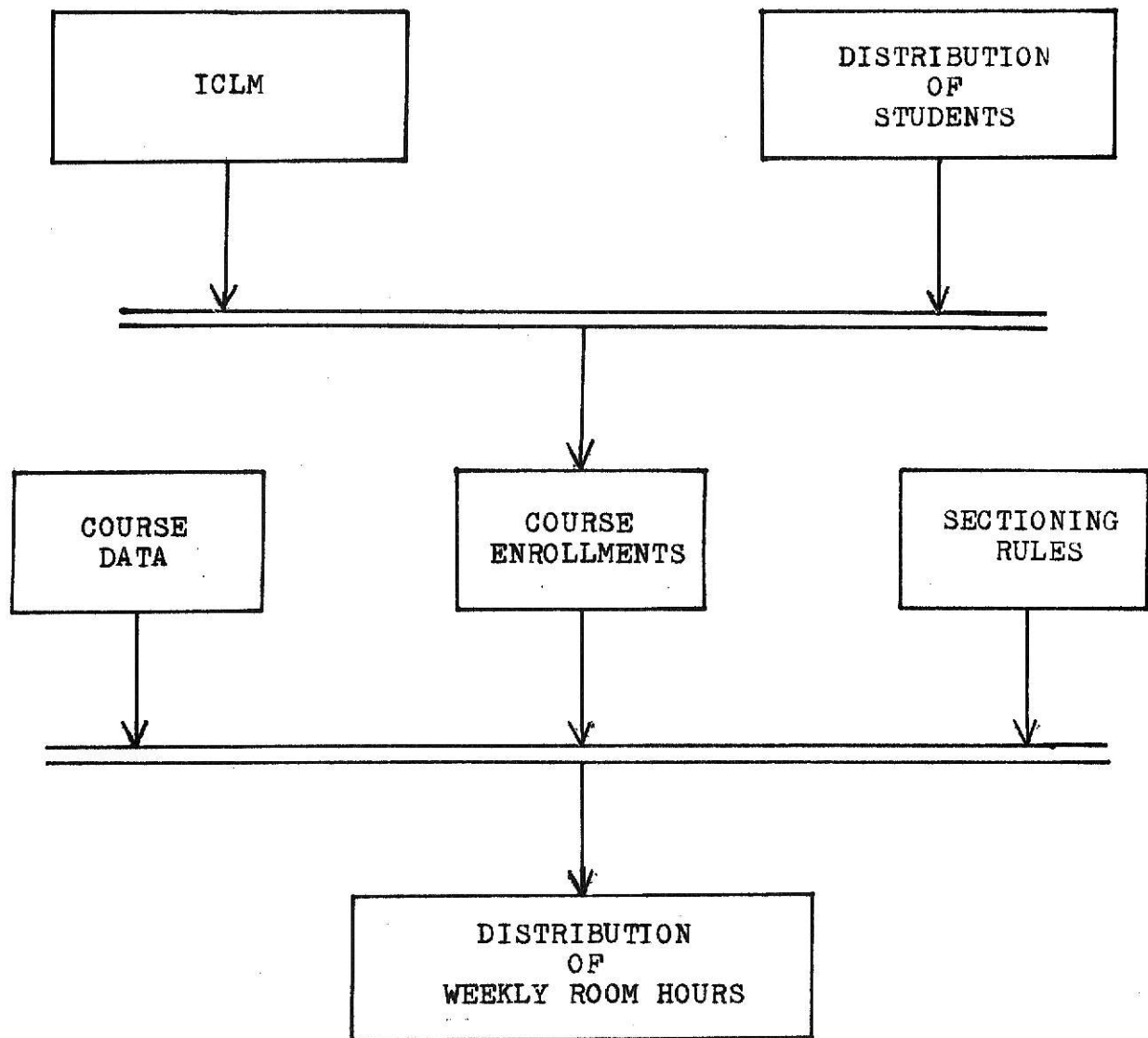


Figure 1.1--The relationship of the ICLM, the course facility requirement data, and the sectioning rules to the projection of a distribution of classroom weekly room hours. Course facility requirements and sectioning rules are the basic parameters of the projection method.

## CHAPTER 2

### THE INDUCED COURSE LOAD MATRIX AND ITS CONSTRUCTION

#### 2.1 THE INDUCED COURSE LOAD MATRIX.

This section will define an induced course load matrix (ICLM) and will discuss some of the various forms it can take. It will explain how an ICLM can be used to predict course enrollment and it will go on to pose some of the questions concerning this matrix which need further investigation.

##### 2.1.1 WHAT IS AN ICLM?

This is a matrix whose row index represents individual courses and whose column index represents identifiable homogeneous groups of students. The elements of this matrix are fractions. Each of these fractions represents the portion of an identifiable homogeneous group of students which can be expected to take a given course.

This can be stated mathematically. Let  $m$  equal the number of courses and let  $n$  equal the number of identifiable homogeneous groups of students. Let  $I$  be an  $m$  by  $n$  matrix whose elements,  $(I_{ij}, i=1,2,\dots,m, j=1,2,\dots,n)$  are the fractions of the students in a particular identifiable homogeneous group of students,  $j$ , expected to take course,  $i$ , and let  $D$  be an  $n$  dimensional column vector of the distribution of the student population, distributed by identifiable homogeneous groups of students. Then in vector notation,

$$\underline{I} \underline{D} = \underline{E} \quad (1)$$

where  $E$  is an  $m$  dimensional column vector of course enrollments and  $I$  represents the ICLM. The above equation shows

mathematically how an ICLM can be used to project course enrollments.

The ICLM and its use are simple in concept. The idea is that homogeneous groups of students can be identified and that students in these groups will have the same enrollment pattern from year to year. Thus if an enrollment prediction were to give the number of students expected in each of these groups (the vector, D) then the enrollments patterns of these groups (the ICLM, I) could be used to determine which courses they would be expected to take. By superimposing the course enrollments generated by each of these groups, total enrollments for each course could be determined.

#### 2.1.2 SOME DEFINITIONS AND EXPLANATIONS.

The definition used here for "course" is the same as WICHE's. They defined "course" as

Organized subject matter (may be a discipline specialty) in which instruction is offered within a given period of time, (usually a quarter or semester) and for which credit toward graduation or certification is usually given. (12, p 42)

The ICLM was made to predict course enrollments because courses provide a key link between student needs and facility and faculty requirements. The student enrollments in each course determine the size of classroom the course needs and, for example, in cases where multiple sections of a course are taught, student enrollments affect the number of faculty required. Of course, the size of section and the faculty required for a course are parameters under the control of the administration. Hence, by knowing course enrollments and controlling these parameters it is theorized that various facility and faculty requirements can be determined. Certainly not all facility nor are all faculty requirements are directly induced by course enrollments, but knowing course enrollments does provide a starting point for determining many of these requirements.

The column index of the ICLM represents identifiable homogeneous groups of students. These classifications can be by department, by curriculum, by level (as related to progress towards a degree), by both curriculum and level, or by any other way that may identify homogeneous groups of students. A group of students must be homogeneous in the sense that they generate the same pattern of course enrollments. Full-time students do not generate the same pattern of enrollments as part-time students. Even though a part-time student can be identified by the same department or curriculum or level as his counterpart full-time student, he does not take the same course load as a full-time student and therefore does not generate the same enrollment pattern. For a group to be homogeneous it is necessary that the proportion of a group taking a course remain relatively constant as the total number of students in the group changes.

WICHE recommended the following classification scheme for dividing the student population into homogeneous groups. Their scheme for a university with a Ph.D. program was to classify each student by curriculum and then divide each curriculum into the following levels:

1. Lower level undergraduate
2. Upper level undergraduate
3. Masters
4. Ph.D.

Curriculum is defined as a course of study. A course of study contains the courses a student must complete for a degree in a curriculum. However, curriculums do not necessarily have to lead to degrees. For example, they could be programs of study recommended for students wishing to enter professional curriculums such as dentistry or law. Level is defined by an institution and designates progress in a curriculum. Freshman, sophomores, seniors, Ph.D., etc, are examples of level designations.

The definitions for the above words are not exact. The exact meaning can change from institution to institution. However, the general sense of their meaning can be expected to remain the same.

#### 2.1.3 A NOTE ON THE DISTRIBUTION OF THE STUDENT POPULATION.

A distribution of the student population is the starting point for this facilities analysis. It is assumed that a projection of the student population can be made. Considering the present state of the art of making student population projections, it is unlikely that it will be possible to make projections distributed as finely as by college and by level within a university. Actually what is being assumed is that, what ever the aggregation of the prediction, it can be proportioned among its component identifiable homogeneous groups of students such that the proportion of the prediction allocated to any group of students remains the same with respect to time. The problem of making a projection of a student population is a prodigious one and will not be discussed further. It is worthy of a separate investigation in itself.

#### 2.1.4 OTHER CONSIDERATIONS.

This project did not set out to investigate all the possible questions which could be asked about an ICLM. It set out to determine if a detailed form of the matrix could be constructed for Kansas State University and, if it could, to report any difficulties which might be involved. Further, an attempt was to be made to demonstrate that the matrix was, indeed, usable. At no time was the purpose to investigate all the questions and ramifications presented by an ICLM.

There are certainly other ideas concerning an ICLM which could be pursued farther:

1. How can homogeneous groups of students best be identified?

2. Could projections be made by departmental levels instead of by courses? Would this be a profitable undertaking?
3. How stable are the fractions in the matrix with respect to time?
4. What effect do changes in course requirements for different curriculums have? What happens when new courses are added or old ones dropped?
5. Is this method of making enrollment projections "good enough" to be of real value?

## 2.2 DEVELOPMENT OF KANSAS STATE'S ICLM.

Construction of an induced course load matrix for Kansas State was one of the main contributions this thesis intended to make. However, as was pointed out in the preceding section, no attempt was made to investigate all the unanswered questions pertaining to induced course load matrices. Rather, this work was intended to simply investigate the feasibility of actually constructing a full-scale, detailed form of ICLM and the feasibility of using it in the analysis of facility requirements.

### 2.2.1 PRELIMINARY CONSIDERATIONS.

Before the matrix could be formed sources of data had to be considered, homogeneous groups of students had to be identified, and the physical form of the matrix had to be determined. This section will discuss some of these preliminary considerations.

#### 2.2.1.1 DATA.

To form the matrix it was necessary to find a source of data that would give the courses each student took in a particular semester and enough data on each student so that students could be sorted into identifiable homogeneous groups. Kansas State's data base contained such information. This information was on what was known as a grade tape. The grade tape was a machine processable magnetic tape used to print student grade reports at the end of a semester. These tapes were available for each semester for several



years in the past. The "grade tape" contained a short student master and a list of courses the student took that semester. The short student master contained such information as the student's address, sex and marital status, curriculum, level, etc.

#### 2.2.1.2 IDENTIFYING HOMOGENEOUS GROUPS OF STUDENTS.

The best way to identify homogeneous groups of students had not been determined. In fact a full-scale matrix had never been constructed before, so for this initial investigation students were categorized by their curriculums and, for some curriculums, by their level within their curriculums. It was hoped that this scheme would identify homogeneous groups of students. The meaning of the curriculum designation given on the tape differed slightly from the general definition given before. Here curriculums consisted of administrative categories determined by the University, not only leading to degrees, but also grouping students not having declared majors or fields of study, students who were taking courses but not working toward any degree (special students), and students in such pre-professional curriculums as pre-nursing and pre-pharmacy. Every student was classified by some curriculum at all times while he was enrolled in the University. One problem with using these curriculum designations was that often students did not report curriculum changes until several semesters had past.

A separate curriculum designation was given for each Masters and Ph.D. program of study. Thus these graduate curriculums did not need to be divided into separate levels. However, the undergraduate curriculums had to be, since juniors and seniors could not be expected to have the same enrollment pattern as freshman and sophomores. Consequently students in undergraduate curriculums were also classified by level. This level was determined from the short student master on the grade tape. The levels on the grade tape were:

- 1) Freshman
- 2) Sophomores
- 3) Juniors and veterinary medicine freshman
- 4) Seniors and veterinary medicine sophomores
- 5) Fifth year students and veterinary medicine juniors
- 6) Veterinary medicine seniors
- 7) Special
- 8) Graduate

Two levels were chosen for undergraduate curriculums. Level one was composed of items one and two above and veterinary medicine freshman and sophomores. Level two was composed of the rest of items three through six.

Kansas State did not have enough part-time students to justify including them as separate homogeneous groups. Consequently, part-time students were in no way identified in the ICLM, although they were included.

#### 2.2.1.3 PHYSICAL FORM SELECTED FOR THE ICLM.

In this initial investigation the matrix was to be constructed from the student records for one semester (fall 1970). A fall semester was chosen because the information on the grade tape concerning curriculums and levels was as up-to-date as possible and because enrollments reached their peaks in fall semesters. Enrollment peaks were of the most concern to facilities planners.

Before construction of the matrix could begin one other decision had to be made: physically, how was the matrix to be stored and how was a matrix of this size to be written onto paper for inspection by an investigator. This matrix was large. There were approximately 1400 courses offered in the fall 1970 semester and about 341 groups of students designated by a curriculum and level. This made it possible for the matrix to have 477,400 elements. However, many of these elements were zero. For example, the proportion of freshman art students regularly taking Boundary Layer Theory is zero. Freshman art students just do not take upper-level mechanical engineering courses. Since many of these elements would be expected to be zero, much space could be saved by

storing only the non-zero elements.

Considering the size of this undertaking it was obvious that the construction of this matrix would have to be undertaken with the aid of a computer. For this matrix to be useful, it had to be stored in a machine processable form. Therefore, three forms of the matrix were produced. One form was stored on machine processable magnetic tape. The two other forms of the matrix were on printed output. One printed output listed the non-zero elements of the matrix by row and the other listed the non-zero elements by column. These two forms of the matrix were to enable an investigator to see which groups of students took a given course (the row output) or to see what courses a particular group of students took (the column output).

#### 2.2.2 CONSTRUCTION OF THE MATRIX.

The matrix was constructed using the grade tape as the source of data. The procedure followed was:

1. Edit the grade tape to extract a curriculum and, where necessary, a level for each student. This information was combined with each course the student took to form an output record containing:
  - a. Course
  - b. Curriculum
  - c. Level
2. Sort these records into each row and column category.
3. Find the total number of records for each row and column category.
4. Form an element of the matrix by dividing the total number of records for a row and column category by the total number of students in the columnar category.
5. For each non-zero element of the matrix write a record onto direct access storage, each record containing a row and column identification and its corresponding matrix element.
6. Print the non-zero elements of the matrix in both row and column order.

This procedure was, of course, carried out using a computer, with programs written to perform the above steps. Each program written did not correspond to a step in the above procedure; some programs combined several steps, and some steps were performed with the aid of several programs.

Figure 2.1 shows the interrelationships of the programs used to create the matrix and the order in which these programs were used. Table 2.1 briefly explains the purpose of each program. Inputs and outputs, comments, and other data on each program are given in the Appendix.

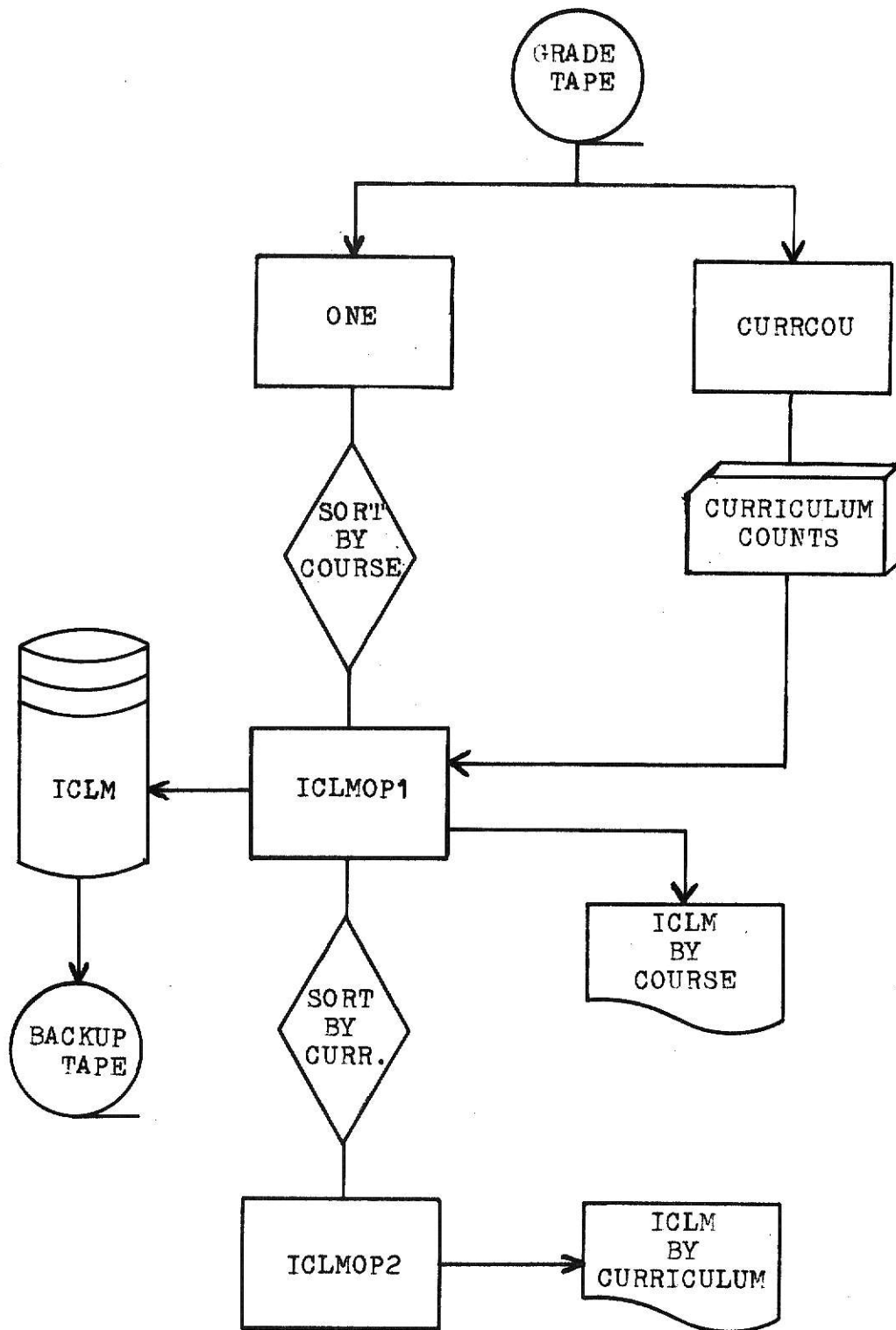


Figure 2.1--Construction of the ICLM.

TABLE 2.1

THE PURPOSE OF EACH PROGRAM  
USED IN THE CONSTRUCTION OF THE ICLM.

Program Name	Program Purpose
ONE	This program was written to edit course, curriculum, and level data from the grade tape and to write this information into a direct access data set. It also converted the University's level designation to the level designation used in this investigation. This is further explained in section 2.2.1.2.
CURRCOU	This program was written to find the total number of students in each columnar category of the matrix.
SORT/ MERGE	This was an IBM supplied utility program for sorting and manipulating records. It was used twice in the construction of the ICLM. It was first used to sort the records produced by ONE. These records were sorted by courses and then, within each course, by columnar category. The second use was to sort the records produced by ICLMOP1. These records were sorted by columnar category and, within each columnar category, by course.
ICLMOP1	<ol style="list-style-type: none"> <li>1) Find the total number of students in each column and row category.</li> <li>2) Form the elements of the ICLM.</li> <li>3) Print the matrix by row.</li> <li>4) Store the matrix in direct access storage for use by the next program.</li> </ol>
ICLMOP2	Print the ICLM in column order.

## CHAPTER 3

### COURSE DATA, SECTIONING RULES, AND A PROJECTION

#### 3.0 INTRODUCTION.

Course enrollments alone are not enough to determine the facility requirements for courses. Rules for converting course enrollments into sections of a course and data on the facility requirements of each section must be available before any attempt at projecting facility requirements can be made.

This chapter will be divided into three main sections. The first main section will explain what data is needed on the facilities requirements for each course and will also explain how this data was formed for Kansas State. The next main section will explain the rules formulated for converting course enrollments into sections of a course. The last main section will demonstrate that the data and the rules developed in the first two sections are consistent with the realities of the University.

#### 3.1 COURSE DATA.

##### 3.1.1 PRELIMINARY DISCUSSION AND DEFINITIONS.

The course data compiled gives the classroom requirements for the courses taught in the fall 1970 semester and, to a lesser extent, the class laboratory requirements for these courses. (Classroom and class laboratory will be defined subsequently.) The main objective in compiling this data was to determine classroom requirements for courses, since classrooms were a resource available to the entire University. In contrast, class laboratories were generally of use only to a single department, and hence each was not a

University wide resource and not of the same importance as a classroom. Some data on class laboratories was compiled as a by-product of compiling the classroom data, but this class laboratory data was not complete enough to be used in a projection of laboratory requirements.

Demands placed on facilities by evening classes were not considered in this study. They do not place significant demands on the University's facilities. Very few classes are held at night and consequently there is always space available for any course needing a classroom or class laboratory in the evening.

Facility requirements for courses are determined by the departments offering them. These requirements are subject to change, depending on the facilities available and the desires of the faculty and administration. These facility requirements are not always explicit in the sense that they are formally available for use in facilities analysis: they are implied by the requests of the departments for different numbers and sizes of classrooms and class laboratories. This section on the course data will show how these requirements were made explicit, and thus how the course data was extracted from Kansas State's data base and made available for general use.

Before any ideas of substance can be explained some definitions must be given.

Augments--Augments were used during the enrollment process to signify that students assigned to one section of a course must also be assigned to another particular section. One example would be where all the students in a particular recitation section were to be assigned to the same laboratory section. Augments were also used to identify sections which took more than one line in the line schedule data to present all the data on that section.

Classroom--"A room used by classes which does not require special-purpose equipment for student use. Included in this category are rooms generally referred to as lecture-demonstration rooms, seminar rooms, and general purpose classrooms." (12, Manual 1, p 40)



Class laboratory--"A room used by regularly scheduled classes which require special-purpose equipment for student participation, experimentation, observation or practice in a field of study." (12, Manual 1, p 40) These rooms can not normally be used for more than one field of study.

Kernel--An example readily demonstrates the meaning of kernel. For most cases if one student takes a course and is expected to attend a recitation and a lecture section, then all students in the course will also be expected to attend a recitation and a lecture. However, for courses with more than one kernel this is not true. One student taking one kernel of a course may attend a lecture and a recitation and another may simply attend a recitation. They both receive credit for the same course but are enrolled in different kernels of it. One particular example of an occurrence of multiple kernels (fall 1970 semester) was U. S. to 1877. This course had three kernels: lecture only, recitation only, and both lecture and recitation.

This information was available at the time students were actually assigned to sections of courses but was not retained in machine processable form after this process was completed, since no further use was usually made of it. Therefore the source of data used to construct the course data did not contain kernel information.

Lecture--Lecture and recitation are two terms used to describe types of sections requiring classroom space. Lectures usually require large classrooms and recitations small classrooms. When two sizes of classrooms are needed by a course, the larger of the two sizes is called a lecture.

Mode of instruction--Identifies the section types to which each student in a course is assigned. Some courses have more than one mode of instruction. For example, one student in a course may attend a lecture for one hour per week and a recitation for two hours per week. Another student in the same course may attend a recitation for three hours per week and not attend any lecture. In this example each student takes the same course but is taught by a different mode of instruction. Mode of instruction has a more general meaning than kernel. For example, a different mode of instruction is given for sections of a course meeting in the evening. Mode of instruction implies explicit values for data on each section of a course. See section 3.1.4 for a description of what constitutes this data.

Recitation--Identifies a section as requiring classroom space of any size. When only one section type for a course requires classroom space this section type is said to be a recitation, regardless of the size of section required.

Room data--This was a deck of punched cards giving vital data on classrooms and class laboratories. The format of these cards and the exact information they contain is given in the Appendix. This data on classrooms and class laboratories came from the University's room utilization studies. The room number and the code identifying the room as a classroom or class laboratory were the only data elements on these cards used in constructing the course data.

Section--"One or more students formally organized for instruction in a specific course under the supervision of an instructor." (12, Manual 1, p 54) Courses can be organized into more than one section. Different sections can require different kinds of facilities, such as classrooms and class laboratories. Sections requiring different kinds of facilities are of different section types. A student in a course may attend more than one section, the sections being of different types. For example, a student may attend a recitation and a laboratory section.

Section size--The number of students in a section, either the actual number or a projected number.

Section type--Identifies sections of a course requiring different sizes of classrooms or class laboratories. The requirements of each section type for each course are listed in the course data.

Weekly room hours (WRH)--Hours per week a section uses a classroom or class laboratory. This definition differs from WICHE's.

Weekly student hours (WSH)--This is a number equal to the weekly room hours multiplied by the section size.

### 3.1.2 COURSE DATA REQUIRED.

After course enrollments are determined the next step is to convert these enrollments into demands for facilities. Data on what facilities are required for each course must be known before this can be done.

WICHE recommended that the following data be compiled on each course:

- A. Organizational Unit--the academic unit offering the course.
- B. Course level--the WICHE data element dictionary lists five specific course levels. These five levels and the suggested aggregation for analytic purposes are:

- |                                |            |
|--------------------------------|------------|
| 1. Preparatory                 | } Lower    |
| 2. Lower division              |            |
| 3. Upper division              | } Upper    |
| 4. Upper division and graduate |            |
| 5. Graduate                    | } Graduate |
- C. Student Credit Hours.
- D. Weekly Student Hours of Classroom Instruction--the number of hours per week, per student, that the course meets for instruction in classrooms (lecture, recitation/discussion, seminar).
- E. Classroom Section Size--either the desired or the maximum number of students per classroom section. If a course is taught in such a way as to have classroom sections of different sizes (e.g., lecture and recitation/discussion groups), it should be treated as two courses.
- F. Weekly Student Hours of Laboratory Instruction--the number of hours per week, per student, that a course meets in laboratories.
- G. Laboratory Section Size--the desired or maximum number of students per laboratory section.
- H. Student Credit Hours of "other" instruction. (12, Manual 6, p 75)

The data compiled on Kansas State's courses became more involved than WICHE's recommendation, primarily because courses at Kansas State were taught in several different modes of instruction and each mode was often composed of several different types of sections each requiring different types or sizes of classrooms and class laboratories.

### 3.1.3 POSSIBLE SOURCES OF COURSE DATA.

The only data on courses contained in Kansas State's data base was the line schedule. The line schedule data was compiled to:

1. Make known the courses offered in each semester.
2. Make known how many sections of a course were to be offered.
3. Make known the scheduled meeting times of each section of each course.
4. Be used in the grade recording and reporting process.
5. Be used in facilities utilization reports.

There were basically two forms of the line schedule data available:

1. Printed and published form used by students and advisors during enrollment.
2. Punched card form containing additional information: the number of students in each section and the room number in which each section was taught. This additional data was compiled towards the middle of each semester for use in reporting and evaluating facility utilization rates.

This data set was called "line schedule" because each line on each page of the published form of this data set corresponded to one section of a course and contained all the data given on that section. The basic "one line, one section" organization was followed in the punched card form of the line schedule data. The format of the punched card form of the line schedule data and the data available on these cards is given in the Appendix.

#### 3.1.4 COURSE DATA COMPILED FOR KANSAS STATE.

The following data was compiled on all courses taught in the fall 1970 semester. The data elements listed are the elements recorded in the machine processable form of the course data. The format of this data is given in the Appendix.

Course Number--A seven digit number assigned to a course.

Digits 1-3--Identify the department and college offering the course.

Digit 4 --Identifies the level of the course. Different levels of courses can be taken for various kinds of credit (graduate, undergraduate).

Digits 4-6--Identify the course within the department.

Digit 7 --Ordinarily courses are known by their six digit number, but in the assignment of students to sections of courses a seventh digit, the "hidden digit", is appended to the course number. This seventh digit indicates that the assignment routine is to assign each student to one section of each different type (type as indicated by the seventh digit).

Number of Modes of Instruction--See section 3.1.1.

Course Name--Self-explanatory.

Credit Hours--Self-explanatory.

For each mode of instruction for a course:

Number of Section Types--Each mode of instruction of a course usually requires several types of sections. This is the number of different types of sections required for a mode of instruction.

Number of Laboratory Rooms--In compiling the course data on sections requiring class laboratory space, the room numbers of rooms used by each section were recorded. This is the number of such rooms. For sections not requiring class laboratory facilities a zero was entered here.

For each section type of each mode of instruction for a course:

Type of Section--This is a character appearing after each course number, the "hidden" digit. These characters designate different types of sections.

Section Size--This is the average section size in which sections of a course have been taught or the average size set by the administration.

Weekly Room Hours--The number of hours each week the section requires a facility.

Number of Sections on Which Section Size is Based--The section sizes in the course data were formed by averaging the historic enrollments of sections of a course. This is the number of sections averaged. It is used in the sectioning rules.

Type of Facility--An "R" identifies the section type as requiring a recitation (classroom) and an "L" identifies the section type as requiring a laboratory (class laboratory).

Evening or Daytime--This identifies section types which meet at night. An evening designation usually will result in an extra mode of instruction being given for a course. An "E" signifies evening and an "D" signifies daytime.

Indicated Class Laboratories--For those section types which met in class laboratory facilities, the room numbers of these rooms were recorded in this space.

### 3.1.5 METHOD OF COMPILING COURSE DATA.

Once the data to be compiled was decided upon, its extraction from available data sources could begin. The procedure for compiling the course data was essentially a



two step process: CRSEDIT compiled initial values for the course data and hand adjustments were made for situations unforeseen when CRSEDIT was programmed. (A description of the inputs, outputs, general algorithm, and other details of CRSEDIT is given in the Appendix.) The remainder of this section will explain the method used to compile each element of course data and will detail the situations requiring hand adjustments, explaining what changes in the course data were necessary and why. As will be shown in section 3.3 this method produced reasonable course data.

Listed below are the elements of the course data and the procedure followed to form each one. These procedures were primarily carried out by CRSEDIT.

Course Number--Derived directly from the records of the line schedule data.

Number of Modes of Instruction--This number was added to the data for courses taught in more than one mode. After the hand adjustments were made most courses had only one mode of instruction.

Course Name--Derived directly from the records of the line schedule data.

Credit Hours--Derived directly from the records of the line schedule data.

Number of Section Types--This was computed by CRSEDIT. Some modifications were made for some courses when the hand adjustments were made. CRSEDIT made a test as each new record was processed to determine if a new type of section had been found for the current course. If a new type had been found the value of the number of section types was increased by one.

Number of Laboratory Rooms--This value was computed by CRSEDIT. Some modifications were made for some courses when the hand adjustments were made. As each type of section was processed a facility requirement was determined from a code on the room data cards. For sections requiring laboratory rooms, the room numbers they required were recorded. This value lists the number of different class laboratory rooms for each section type. The number of laboratory rooms for each section is a vector of values, a value for each section type requiring special-purpose facilities. Sections not requiring special-purpose facilities had a value of zero recorded here.

Type of Section--This was determined by CRSEDIT. As each record of the line schedule data was processed and a new section type was recognized, this character was stored. This is a character appearing at the end of the course number (the seventh digit).

Section Size--A section size was given in the data for each section. This value was summed over each section type of each mode. The number of sections of a section type was also determined. These two quantities were used to form an average section size for the section type. For section types with only one section having been taught, the section size was set equal to the section size given in the line schedule data.

Weekly Room Hours--Weekly room hours for each section were determined by the procedure TIME in CRSEDIT. The value of weekly room hours recorded in the course data was an average over all sections of a common type taught in a common time of day (daytime or evening). The procedure TIME determined if a section met in the daytime or evening and determined the nominal hours per week each section met by finding the nominal hours per day and multiplying it by the number of days per week. Sections said to meet for a nominal hour actually meet for 50 minutes with a 10 minute passing time. Some sections meet for more than a nominal hour or an integer multiple thereof. Consequently the total actual meeting time for these sections was calculated and rounded to the nearest half hour so the section's weekly room hours would be on the same nominal basis as other sections.

An average value over all sections was recorded because all sections of a common type for a course did not always meet for the same nominal length of time. This happened in two situations. The first was when two cards of the line schedule data were used to present the data for one section of a course (augments). Since CRSEDIT was not able to recognize this situation it computed an incorrect number of sections on which to base the average weekly room hours. (An example was Gen Chem 1, 221110). The other situation was two different lengths of meeting times for sections of a common type. An example of this was Arch Const, 105311, in which some sections met nine nominal hours per week and others met eight.

Number of Sections on Which Section Size is Based--This value was increased by one as each new section of a common type was recognized. It was recorded and set to one when a new section type was encountered.

Type of Facility--This data element contained the classroom or class laboratory requirements for a section type. There

were two receiving fields, designated REC and LAB, for this data element. As each section of the line schedule data was processed, the room where the section met was checked against the room data. If the section did not require a room in the room data these fields were not changed. If the room was a classroom, an "R" was put in the REC field. If the room was a class laboratory, an "L" was put in the LAB field. After all sections of a common type were processed, the REC and LAB fields were checked. If the fields were still blank (they were initialized to blank), this signified that the section type did not require any classrooms or class laboratories. If one field was blank and the other contained an "R" or an "L", then the fields signified the type of facility required for the section type. If both an "R" and an "L" were present, then this signified that some sections of a common type met in classrooms and some in class laboratories. In this case the LAB field was changed to blank, signifying that the section type did not require special facilities. (If one section does not need special facilities, then the other sections also do not require special facilities.)

Evening or Daytime--An over punch in the starting time field of the line schedule data signified that a section met in the evening. Courses with sections meeting in both the daytime and the evening were given an extra mode of instruction--an evening mode. In making projections, facility requirements for courses and modes of courses meeting in the evening were not processed. (Facilities demands in the evening can always be met since there are so few evening classes.)

Indicated Class Laboratories--The facilities requirements for each section type were determined as explained under "Type of Facility" above. For section types requiring laboratory space, the room numbers of the rooms indicated in the line schedule data for these sections were recorded in this space.

When CRSEDIT was programmed it was thought that kernel and augment information would not be necessary to the process of extracting course data from the line schedule. However, after the output of CRSEDIT was available it was obvious that, in a few courses, adjustments which made use of kernel and augment information needed to be made. (This kernel and augment information was made available in printed form from the Office of Admissions and Records.) Additionally, separate



modes of instruction also needed to be designated for sections of courses taught in the evening. Examples of these situations and the adjustments made for them follow.

Example 1. Two records of the line schedule data were used to describe the two different meeting times of one section. This caused CRSEDIT to act as if two sections were present, each with different weekly room hours. The result was an erroneous number of weekly room hours. Augment information was used to determine that the two records belonged together. An example of this is Reliability and Quality Assurance, course 550742.

Example 2. One section of a course met in the evening. This caused CRSEDIT to make an extra type of section for the course, requiring that all students in the course be assigned to a daytime class meeting and an evening one. These courses should have been taught in two modes of instruction: evening and daytime. The course data was changed to show two modes of instruction for courses in this situation. An example was Audio-Visual Instruction, course 415602.

Example 3. Differences in section types within the sections of a course were recognized by differences in the seventh digit of each course number of each line of the line schedule data processed. Sometimes when augments were present, for example, designating that all the students in one laboratory were to all be in the same recitation, there was not a different digit for the laboratory and recitation sections. This caused CRSEDIT not to recognize the need for both recitation and laboratory section types. The augment information was used to determine the correct numbers and types of sections for the course. An example of this situation was General Chemistry Laboratory, course 221110.

Example 4. A course was taught in several modes of instruction and CRSEDIT was able to recognize this. An example of a course with three modes of instruction was U. S. to 1877.

Some students took this course as a lecture course, some students took it as a recitation course, and some students took it as a combination lecture and recitation course. Before adjustments were made, CRSEDIT interpreted the course as being taught in one mode of instruction with a typical student taking one of four recitation sections of 62 students, one of eight recitation sections of 22 students, and a lecture section of 194. Adjustments were made using kernel information so that a typical student would take the course in a lecture section of 92 students, or a recitation section of 32 students, or a lecture section of 194 and a recitation section of 22 students. Kernel information was used to separate the sections of the course into their correct modes of instruction.

### 3.2 SECTIONING RULES.

Once course enrollments are projected and course data compiled there still remains one other problem to be solved before classroom and class laboratory demands can be projected. This is the problem of determining how courses will be divided into sections when course enrollments change from their historical values. For example, if there is an increase in a course's enrollment, will the increase be met by adding another section or be met by distributing the increase among the present number of sections?

The specific rules proposed by this investigation for use in sectioning courses will now be presented. It must be emphasized that these rules were selected arbitrarily; there was no rigorous, logical derivation of them made. Their worth is demonstrated in section 3.3 where the course data and sectioning rules are validated. These rules are contained within the logic of the program, DISTWRH, discussed in section 3.3.

#### RULE 1

Situation:

1. Only one section of a given type had been taught in the past.
2. The historical section size as determined by the line

schedule data was less than 80.

3. The projected enrollment for the section type was less than 80.

Rule: Project one section with enrollment equal to that of the section type.

## RULE 2

Situation:

1. Only one section of a given type had been taught in the past.
2. The historical section size as determined by the line schedule data was less than 80.
3. The projected enrollment for the section type was greater than 80.

Rule: Form two sections of equal size.

Justification for Rule 1 and Rule 2:

In these situations there was no history of multiple sections on which to base an average section size. Departments faced with these situations made their decisions based on available rooms, available faculty, level of course, etc., and their method of making these decisions was not recorded. From a strict facilities planning point of view there were no specific rules found for determining sectioning rules under the above conditions. A section size of 80 was arbitrarily chosen because a value was needed before the analysis could continue.

## RULE 3

Situation:

1. Only one section of a given type had been taught in the past.
2. The historical section size as determined by the line schedule data was greater than 80.

Rule: Project one section with enrollment equal to that of the section type.

Justification:

Section types having been taught in large sections can continue to be taught in large sections. No provision was made for determining if the projected section size was greater than the maximum room size available. It may be possible to modify these rules so they will consider maximum size of available classrooms.

RULE 4

## Situation:

More than one section of a given type had been taught in the past.

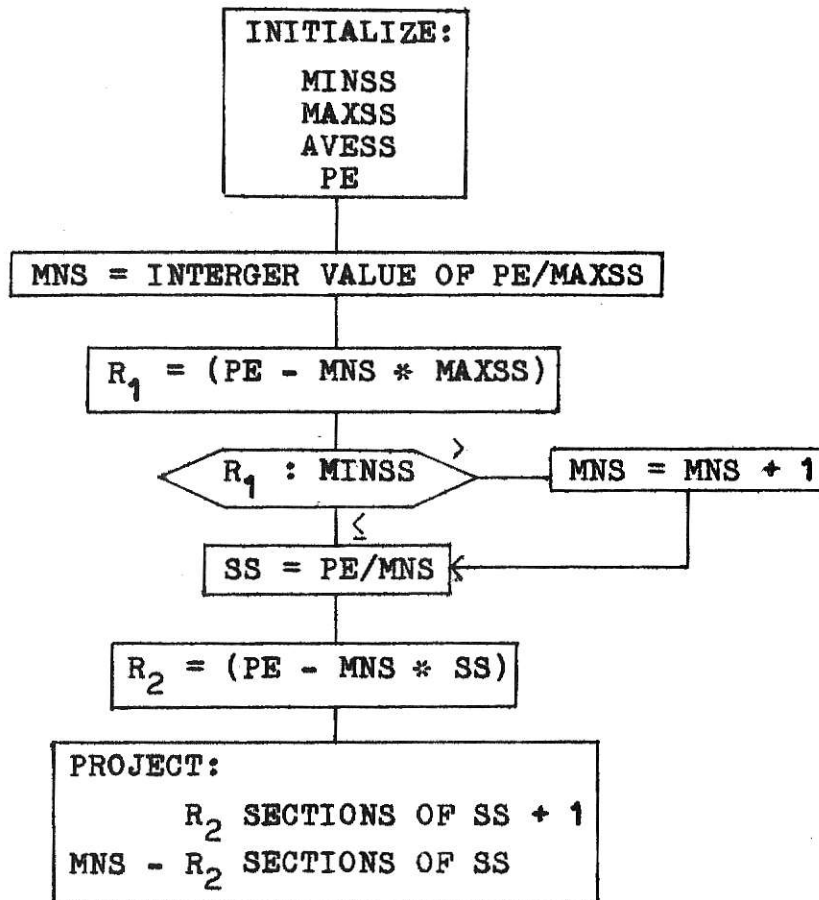
Rule: Generally speaking, divide the projected enrollment into sections of size equal to the average section size determined for the section type. However, dividing the projected enrollment into sections of a certain average size presents some problems when the division can not be performed exactly. Therefore the following procedure was formulated. This procedure is presented schematically in Figure 3.1.

1. Set a minimum section size. This is determined by consulting a table and performing an operation upon the appropriate entry. (See below for further details.)
2. Set the maximum section size equal to the average section size.
3. Divide the projected enrollment by the maximum section size and truncate the result to the nearest integer value. This gives the initial value for the maximum number of sections, MNS.
4. Multiply the MNS by the maximum section size and subtract this value from the projected enrollment to get a remainder.
5. If the remainder is greater than the minimum section size increase the maximum number of sections by unity; if not, leave it as it is.
6. Divide the projected enrollment by the maximum number of sections to get the section size, SS, to be projected.
7. Multiply this section size by the MNS and subtract this from the projected enrollment to form another remainder, R.
8. Project R sections of size  $SS+1$  and  $(MNS-R)$  sections of size SS.

## Justification:

Step 5 determined if the remainder is large enough to justify another section. If another section is justified, the section size projected will be less than the average section size. If not, the projected section size will be greater than the average section size.

Not all sections can be the same size because the MNS does not always divide evenly into the projected enrollment. In cases where it does not, a remainder, R, is formed and



Where:

MINSS = minimum section size  
 MAXSS = maximum section size  
 AVESS = average section size  
 PE = projected enrollment  
 MNS = maximum number of sections  
 $R_1$  = first remainder  
 $R_2$  = second remainder  
 SS = projected section size

Figure 3.1--Method used to section courses having histories of multiple sections.

distributed over the enrollment in the projected sections. Thus R sections are increased by one and (MNS-R) sections are left at section size SS. This distribution of the remainder is performed in step 8.

The minimum section size was determined by consulting a table and performing an operation upon the appropriate entry. The values of this table are listed below. This table was available to DISWRH, discussed in section 3.3.

<u>Index Number</u>	<u>Value</u>
1	- 5
2	- 7
3	-15
4	-18
5	-20
6	-25
7	-30
8	-40
9	-15
10	-10
11	-20
12	-30
13	-40
14	-40
15	-50

To determine the minimum section size, the average section size is divided by ten. This gives an index number for the table above. If the index number is less than one, the minimum section size is set to five. If the index number is greater than 15, it is set to 15. The minimum section size is then set equal to the result of adding the value in the table (corresponding to the previously determined index number) to the average section size. The values were initially determined arbitrarily, but were later adjusted to be consistent with available data. The above values are the adjusted values.

### 3.3 VALIDATING THE COURSE DATA AND SECTIONING RULES.

The next step after compiling the course data and formulating the sectioning rules was to demonstrate their usefulness and validity. Since the course data was compiled from fall 1970 data, a projection of the distribution of classroom hours made using the actual course enrollments, course data, and the proposed sectioning rules should be similar to the actual distribution which did occur. This section



discusses how the actual distribution of weekly room hours was tabulated and how a projection of classroom weekly room hours was made and goes on to compare the two distributions.

### 3.3.1 ACTUAL DISTRIBUTION OF WEEKLY ROOM HOURS.

The actual distribution of classroom weekly room hours was tabulated from the line schedule data. CUMDIST was the name of the program written to do this. Using the room data, CUMDIST processed each section to determine if it met in a classroom. If it did, the section's weekly room hours were calculated and added to the tabulation. If it did not meet in a classroom, no changes were made in the tabulation and CUMDIST started processing a new record.

In situations where sections of a common section type met in both classrooms and class laboratories, only the sections meeting in classrooms were counted in the actual tabulation. In the same situation the course data would have caused the class laboratory sections to have been projected as classrooms. The justification being that if one section of a section type did not need special facilities then neither did the other sections. The actual distribution also included evening classes in the tabulation where, in the same situation, the course data would have not.

Listed in the Appendix with the description of CUMDIST is the output CUMDIST produced. Listed there is the tabulation by section size (in increments of one from 1 to 350) of the following:

1. Classroom weekly room hours.
2. Cumulative classroom weekly room hours.
3. Classroom weekly student hours (WRH multiplied by section enrollment).
4. Cumulative classroom weekly student hours.

### 3.3.2 A PROJECTION USING THE COURSE DATA AND SECTIONING RULES.

To test the course data and the sectioning rules a program (DISTWRH) was written. It made projections of weekly room hours for classrooms and class laboratories. It used the

course enrollments for fall 1970 as input, converting these course enrollments into facilities requirements by using the course data and the sectioning rules. The output was a distribution by section size of the weekly room hours projected for classrooms and a distribution by section size and department of weekly room hours projected for class laboratories.

Class laboratory weekly room hours were projected by this program, but they were not tested against the actual weekly room hours for class laboratories. (Since class laboratories are under the direct control of departments, they are the ones in the best position to determine class laboratory requirements.) The line schedule data was never meant to contain all of the information needed to determine class laboratory requirements. The line schedule data could indicate that a class laboratory was held, but it could not indicate that a given class laboratory room could be used by only one course or by several courses in a department. Without knowing specific class laboratory requirements beyond those implicit within the line schedule data, it is impossible to determine exact facility requirements.

DISTWRH utilized the data used to form the ICLM as its source of course enrollments. Each record of the ICLM contained not only the identifying row and column index and corresponding element, but also the total number of students used in forming each element. By summing the number of students contained on each record over all records for a course, a figure for the course's fall 1970 enrollment could be determined. However, this number was not exactly the same as the number given by the line schedule data and used in the compilation of the actual distribution of weekly room hours. The difference was the result of changes which occurred in course enrollments between the time the line schedule data was produced and the time when the grade tape was produced.



DISTWRH used the following procedure to generate the distribution of weekly room hours.

1. Determine a course enrollment and find the course data for this course.
2. For courses with more than one mode of instruction, proportion the course enrollment between the separate modes of instruction. Use the same proportion as determined from the historical enrollments contained in the course data.
3. For each section type of each mode of instruction which requires classroom space, project the appropriate number of weekly room hours of classroom space as determined by the course data and the sectioning rules. The number of weekly room hours projected is then added to the weekly room hours for the appropriate section size. Weekly room hours are computed by multiplying the weekly room hours in the course data by the number of sections projected.
4. After all courses have been processed the distribution of weekly room hours for classrooms is then printed out and also punched onto cards. The distribution for class laboratories is also printed by this program but, as was explained above, nothing further was done with this distribution.

Table 3.1 lists some of the courses for which projections were made. For each of these courses the size of each section projected is listed with the size of each section listed for the course in the line schedule data. This was done to demonstrate that DISTWRH does, indeed, make projections similar to the apparent manner in which courses were intended to be taught.

### 3.3.3 EVALUATION OF PROJECTED DISTRIBUTION OF WEEKLY ROOM HOURS.

Using the course enrollments for fall 1970, the course data, and the sectioning rules, DISTWRH produced a distribution, by section size, of weekly room hours. CUMDIST determined the same distribution from the actual historic data, the fall 1970 line schedule data. Figures 3.2-3.5 give histograms of the weekly room hours for various section size intervals. In each figure the actual and the projected values are given side-by-side. The difference between these four figures is



**TABLE 3.1--Continued**

[illegible]

TABLE 3.1--Continued

Course Number	Course Name	P/A	Sectioned Into:																	
259165	INTR PHIL	P A	REC	58	59	59	59													
				73	55	67	61													
265211	GEN PHYS 1L	P A P A P A	REC  LAB  LEC	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21
				26	26	31	31	25	21	17	23	17	17	16	17	16	16	19	18	18
				19	19	19	20	20	20	20	20	20	20	20	20	20	20	20	20	20
				24	19	18	22	20	21	21	20	12	18	17	27	23	26	24	20	20
				317																
				314																
269110	INT POL SCI	P A	LEC	108	109	109	109													
				90	92	93	92	83												
269444	AMER DEMOC	P A	REC	48	48	49														
				50	39	58														
273110	GEN PSYCH	P A	LEC	271	271	271	272	272	272	272	331									
				96	268	313	317	329	331											
277211	INTRO SOC	P A P A	REC  LEC	27	27	27	28	28	28	28	28	28	28	28	28	28	28	28	28	28
				25	27	26	28	28	29	29	30									
				249	202	202	202	202	203											
				221	181	153	184	190	184											
285321	ECO STAT	P A	REC	52	53	53														
				51	51	58														
286315	FUN COMP PRO	P A	REC	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
				25	42	26	23	18	37	26	23	22	38	26	40	30	34			

TABLE 3.1--Continued

Course Number	Course Name	P/A	Sectioned Into:																	
305275	FUND ACCTG	P	REC	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29
		A		29	28	33	32	32	32	30	32	30	30	30	30	30	30	30	30	30
		P	LEC	205	205															
		A		220	196															
305425	BUS LAW I	P	REC	48	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49	49
		A		52	51	49	49	49	50	49	49	49	49	49	49	49	49	49	49	49
405302	ED PSYCH II	P	REC	54	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55
		A		60	51	55	55	56	53	62										
405616	EDUC SOC LG	P	REC	29	29	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
		A		15	22	34	35	35	36	34										
		P	LEC	178																
		A		178																
415451	PRIN SEC ED	P	REC	31	32	32														
		A		41	28	26														
		P	LEC	95																
		A		94																
510412	DYNAMICS	P	REC	19	19	19	19	19	19	19	20									
		A		21	21	24	12	16	18											
560212	GRAPH COMM	P	LAB	15	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
		A		14	15	19	18	15	14	15										
611101	DES CONT LIV	P	LEC	183	183															
		A		184	184															

TABLE 3.1--Continued

Course Number	Course Name	P/A	Sectioned Into:	
620325	PRESCH CHILD	P	REC	23 23 24
		A		23 23 24
		P	LEC	172 70
		A		168 70
640300	MEAL MGMT L	P	REC	34 34
		A		37 33
		P	LAB	17 17 17 17
		A		20 14 19 17
740431	ANAT PHYSIOL	P	LAB	41 42 41 124
		A		45 43 38 126
740600	GROSS ANAT I	P	LEC	85
		A		85
		P	LAB	42 43
		A		42 43

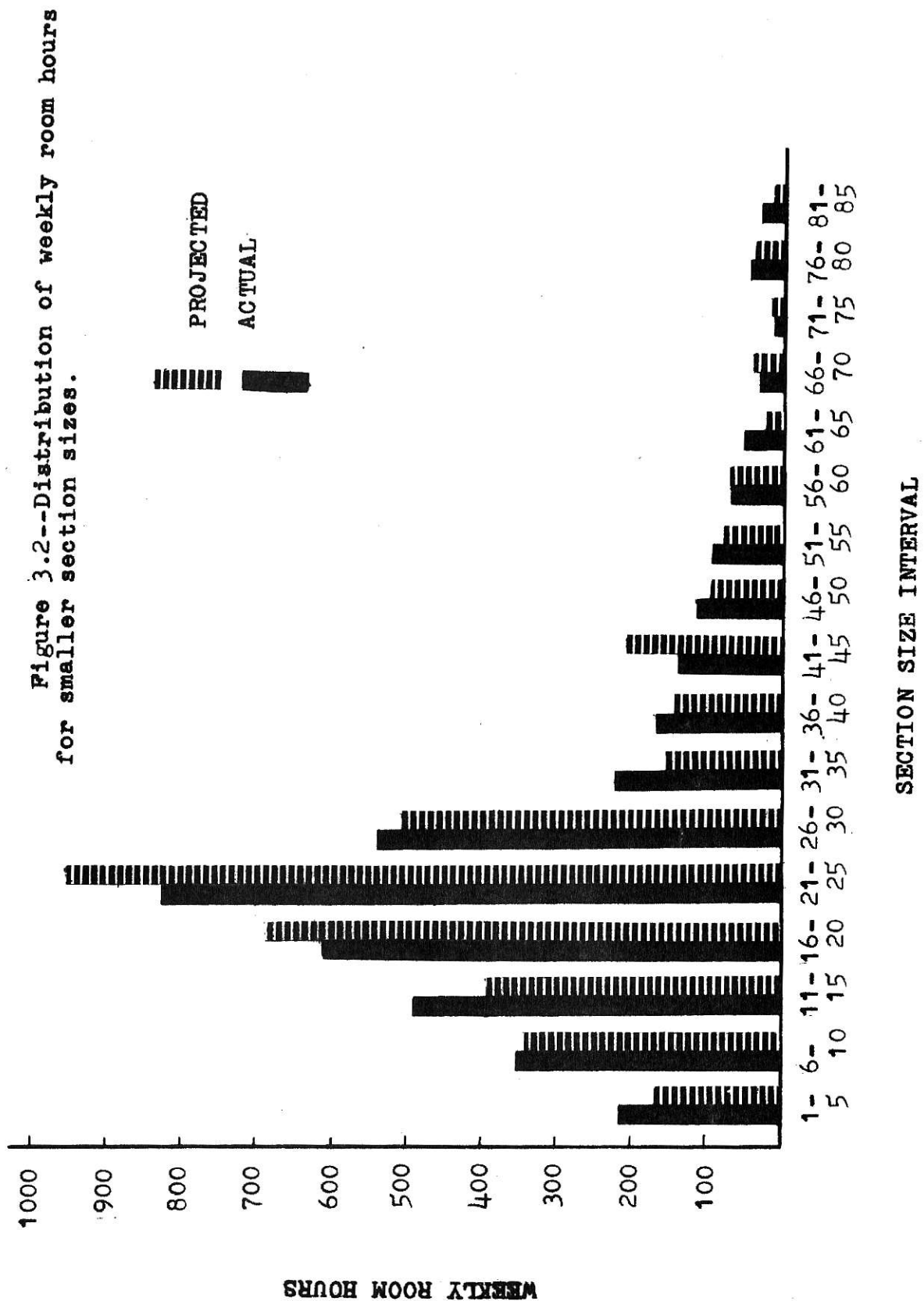
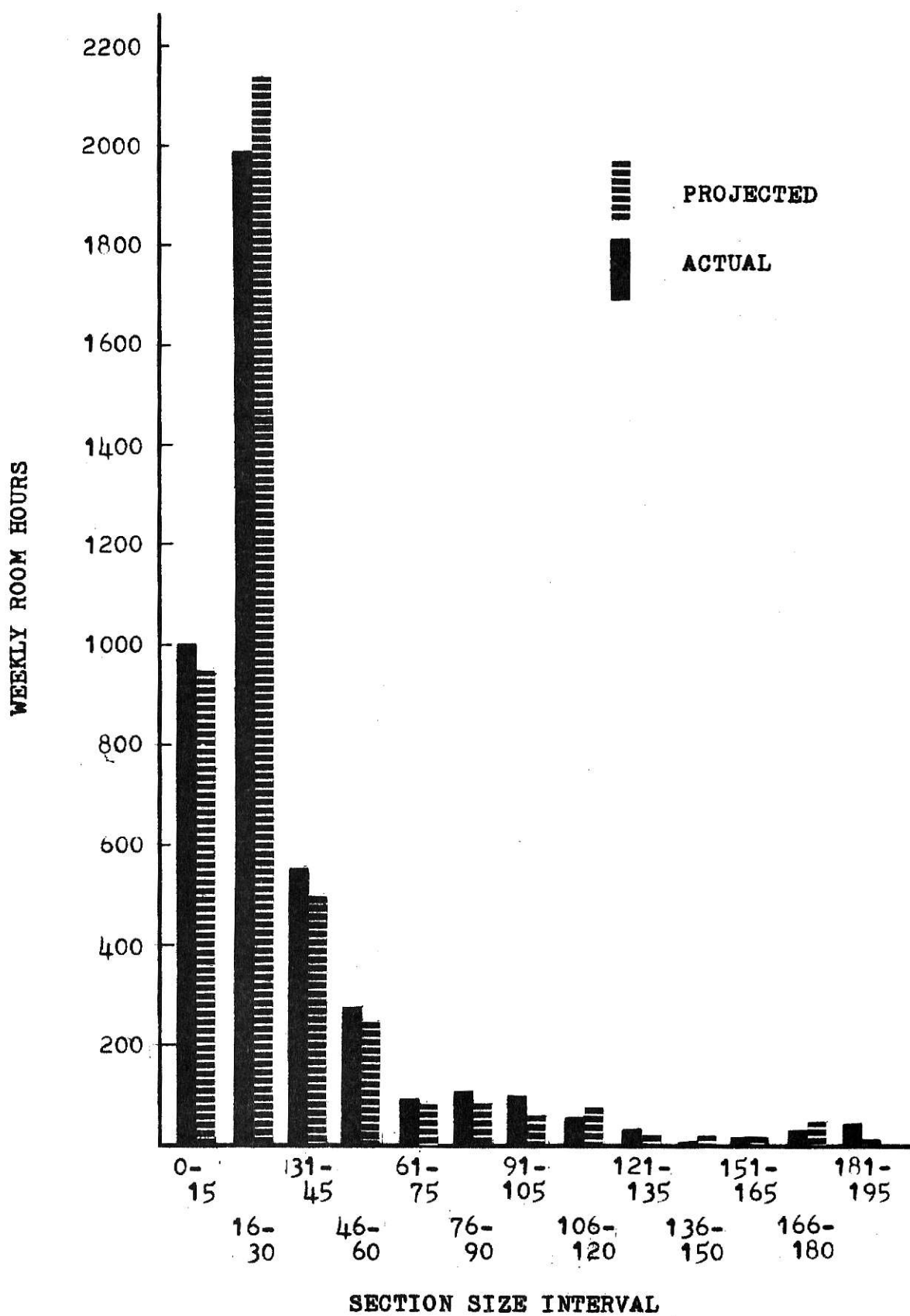




Figure 3.3--Distribution of weekly room hours in intervals of 15 from 0 to 195.



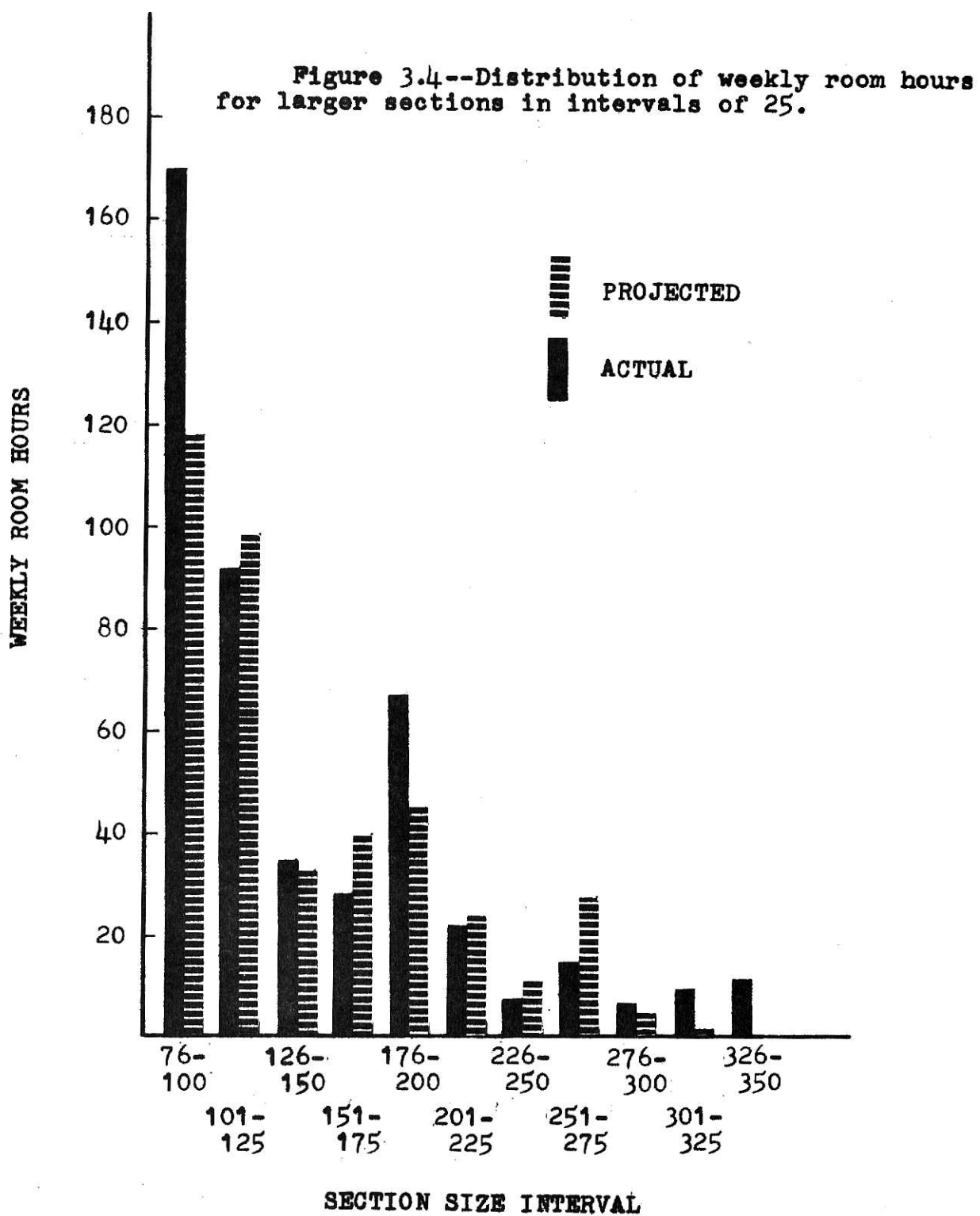
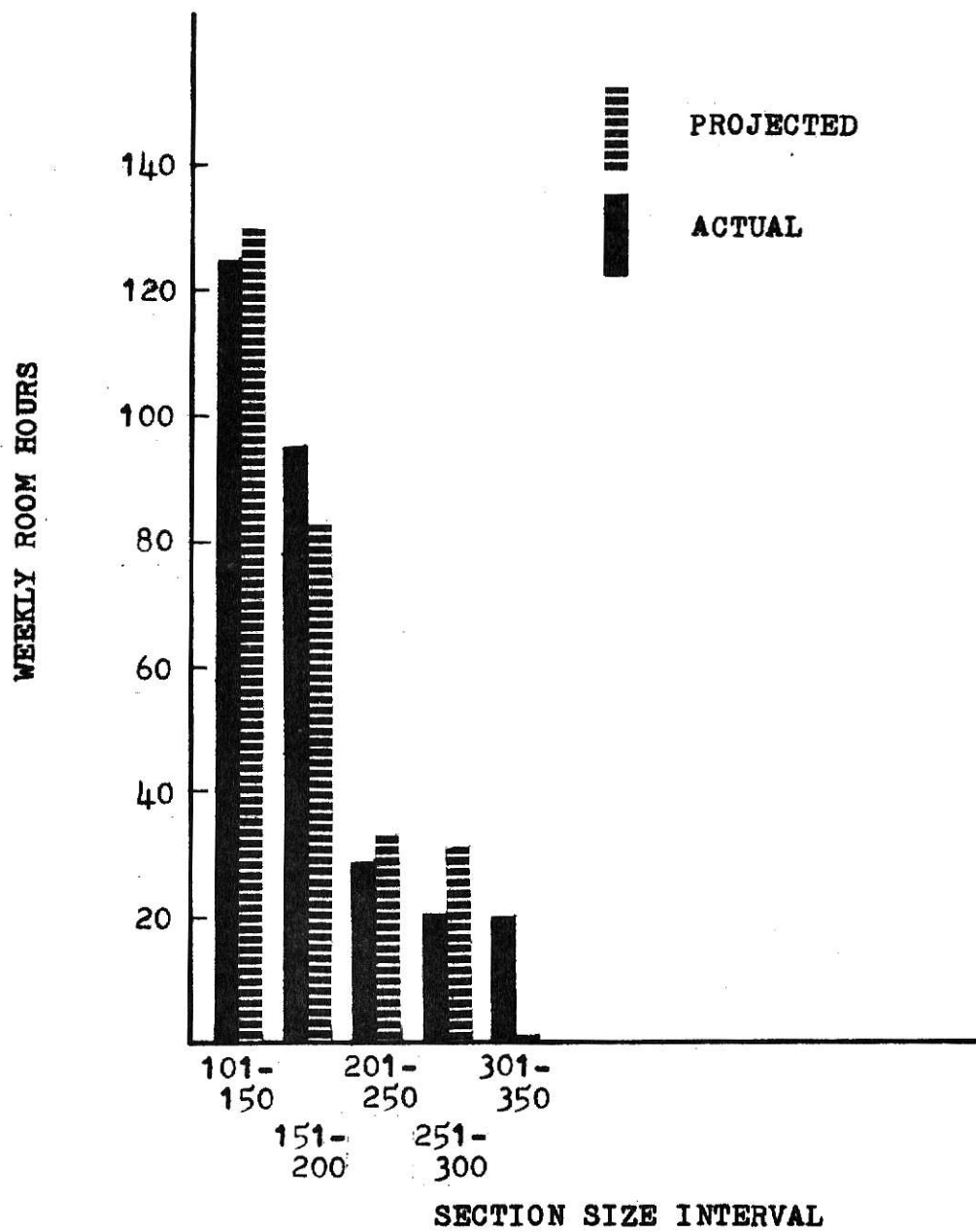


Figure 3.5--Distribution of weekly room hours for larger sections in intervals of 50.



the scale used for the ordinate and the interval used for the section size. The data for Figures 3.2-3.5 is presented in Tables 3.2-3.5. The data for these tables came from the output of CUMDIST and DISTWRH and is listed in the Appendix with these programs.

Figure 3.2 shows that the shapes of the projected and actual distributions are very close for smaller (less than 80) section sizes. Figure 3.3 compares the distributions using a larger section size interval and a larger range of section sizes than Figure 3.2. The distributions in Figure 3.3 appear to be about the same for section sizes up to 180.

It is difficult to determine from Figures 3.4 and 3.5 if the projected and actual distributions are alike in the over 175 section size range. Since sections requiring classrooms in the over 175 section size range were of some concern to University administrators, a close examination of the reasons for the differences in the distributions was in order. Table 3.6 lists the courses which, in either the line schedule data or the projected data, have sections of size greater than 175. Not all sections of a course are listed in this table. Only the sections for section types requiring large sections, over 175, are listed. Both the projected and actual sizes of sections are listed for each course. Weekly room hours required for the section type are also given. Comments are given in the footnotes of Table 3.6 explaining why a difference occurred and telling what has to be done to correct it. Figure 3.6 gives the corrected distribution for the over 175 section size range, and Table 3.7 gives the data for this figure.

After the adjustments in Table 3.6 were made, a close examination of it revealed that, in most cases, the differences in section sizes were due to the differences in the enrollments in courses as given by the line schedule data and the data used to form the ICLM. The manner in which the course data

TABLE 3.2

DATA FOR FIGURE 3.2

Section Size Interval	WRH	
	Actual	Projected
1- 5	165	215
6- 10	349	340
11- 15	484	392
16- 20	617	668
21- 25	833	957
26- 30	533	506
31- 35	228	152
36- 40	170	140
41- 45	137	208
46- 50	113	98
51- 55	90	77
56- 60	69	71
61- 65	49	22
66- 70	29	41
71- 75	12	17
76- 80	46	38
81- 85	30	14

TABLE 3.3

DATA FOR FIGURE 3.3

Section Size Interval	WRH	
	Actual	Projected
1- 15	998	947
16- 30	1983	2131
31- 45	535	500
46- 60	272	246
61- 75	90	80
76- 90	107	82
91-105	95	54
106-120	56	76
121-135	29	13
136-150	7	22
151-165	15	14
166-180	24	42
181-195	49	11

TABLE 3.4

DATA FOR FIGURE 3.4

Section Size Interval	WRH	
	Actual	Projected
76-100	169	117
101-125	91	98
126-150	34	32
151-175	28	39
176-200	67	44
201-225	21	23
226-250	7	10
251-275	14	27
276-300	6	4
301-325	9	1
326-350	11	0



TABLE 3.5

DATA FOR FIGURE 3.5

Section Size Interval	Actual	Projected
101-150	125	130
151-200	95	83
201-250	29	33
251-300	20	31
301-350	20	1

TABLE 3.6

COURSES WITH SECTION SIZES OF OVER 175 IN EITHER  
THE ACTUAL OR PROJECTED DISTRIBUTIONS OF WRH

---- Value removed from distribution

—— Value added to distribution

Course Number	Course Name	WRH	Actual Section Size	Projected Section Size
025102	PR ANI SCI	3	250	244
035100	AGRI SOCIETY	2	296	281
105270	ARCH GRAPH	3	183	174
110101	LAND DESIGN	3	181	170
209195	SUR ART HIS	3	225, 25	199
215200	PUB HLT BACT	3	257	256
215205	GEN ZOO L	2	179, 151	163, 163
215210	GEN BOTANY	2	179	177
221210	CHEMISTRY I	3	192, 204, 153,	170, 171, 171, 171
221230	CHEMISTRY II	3	197	<sup>170</sup> 374 <sup>a</sup> , 208
229370	BOOKS MEN I	3	160, 56	218 <sup>b</sup> , 117, 57
241251	U. S. to 1877	2	194	179
245010	INTERM ALG L	2	280	260
245100	COLLEGE ALG	2	323, 356, 347	478, 478 <sup>c</sup> , 318, 318, 319
245150	PLANE TRIG R	2	256	240
245220	AN GM CALC 2	2	268, 264	470 <sup>d</sup> , 235, 235
245222	AN GM CALC 3	2	206	196
259170	INTRO LOGIC	3	176	151
265101	MAN P WLD 1	3	187, 182, 183 179	176, 176, 176, 176
265211	GEN PHYS I L	1	314	317
265310	ENGR PHYS I	1	195	192
273110	GEN PSYCH	3	268, 313, 317 329, 331, 96	271, 271, 271, 272, 272, 272
273405	ABNORMAL PSYCH	3	204	201

TABLE 3.6--Continued

Course Number	Course Name	WRH	Actual Section Size	Projected Section Size
273420	PERSNLTY DEV	3	232	238
277211	INTRO SOC	2	221	249
		3	184, 181, 153, 184, 190	202, 202, 202, 205
278200	INT CUL ANTH	3	182, 218	197, 198
281105	ORAL COMM I	1	205, 246, 275, 285, 266	260, 260, 260, 260
281106	ORAL COMM IA	1	284, <u>284</u> <sup>e</sup>	285, 285
285320	EL STATISTIC	3	<u>340</u> <sup>f</sup>	11 sections of 41
305275	FUND ACCTG L	2	220, 196	205, 205
405616	EDUC SOC L	1	178	178
530115	ENGG ASSEM	1	203	205
560115	ENGG ASSEM	1	193	196
610131	SOCIO ECON	2	191	189
611101	DES CONT LIV	3	184, 184	183, 183
620250	HUMAN REL	1	196, 191	192, 192
640133	FOOD FOR MAN	2	192, 42, 46	91, 91, 91
650110	INTRO HE	2	198, 202	198, 198

## FOOTNOTES TO TABLE 3.6

<sup>a</sup>The number of students used in projecting facilities requirements for this course was inflated because not all the students counted in the enrollment, as supplied by the ICLM data (see section 3.3.2), took the course in the same way. Part of the students, 208 of 374, took the course and used the facilities projected for it. The other 166 students received credit for the course by "quizzing out" of Chemistry I, and this is why they were included in the enrollment. However, these "quiz out" students used facilities projected for Chemistry I (they were handled as a separate mode of instruction for Chemistry I). The enrollment used for the projection was changed to 208 and the projection made again with this adjustment considered.

<sup>b</sup>This course should have had an evening mode of instruction provided for it in the course data. However, since this extra mode was not present, DISTWRH projected one lecture section of 218 and one evening section of 218. After an evening mode of instruction was provided for this course, the projection was one lecture section of 117 and one evening section of 57.

<sup>c</sup>The section size used for a minimum was set too large for this section size range. Consequently, two very large sections were projected. After the value of -250 was set for an index number of 34 (see section 3.2 for a discussion of how minimum section sizes are determined), the projection was made again with two lecture sections of 318 and one lecture section of 319 projected.

<sup>d</sup>The section size used for a minimum was set too large for this section size range. Consequently, one very large section was projected. After the value of -150 was set for an index number of 26 (see section 3.2 for discussion of how

minimum section sizes are determined), the projection was made again with two lecture sections of 235 projected.

<sup>e</sup>Apparently a card for one section of this course was missing from the line schedule data. For comparison purposes a section of 284 was added to the actual distribution.

<sup>f</sup>An apparent keypunch error on a card in the line schedule data caused this large number. This value was eliminated from the actual distribution.

Figure 3.6--Distribution of weekly room hours, in intervals of 50 for larger section sizes, after the adjustments explained in Table 3.6 were made.

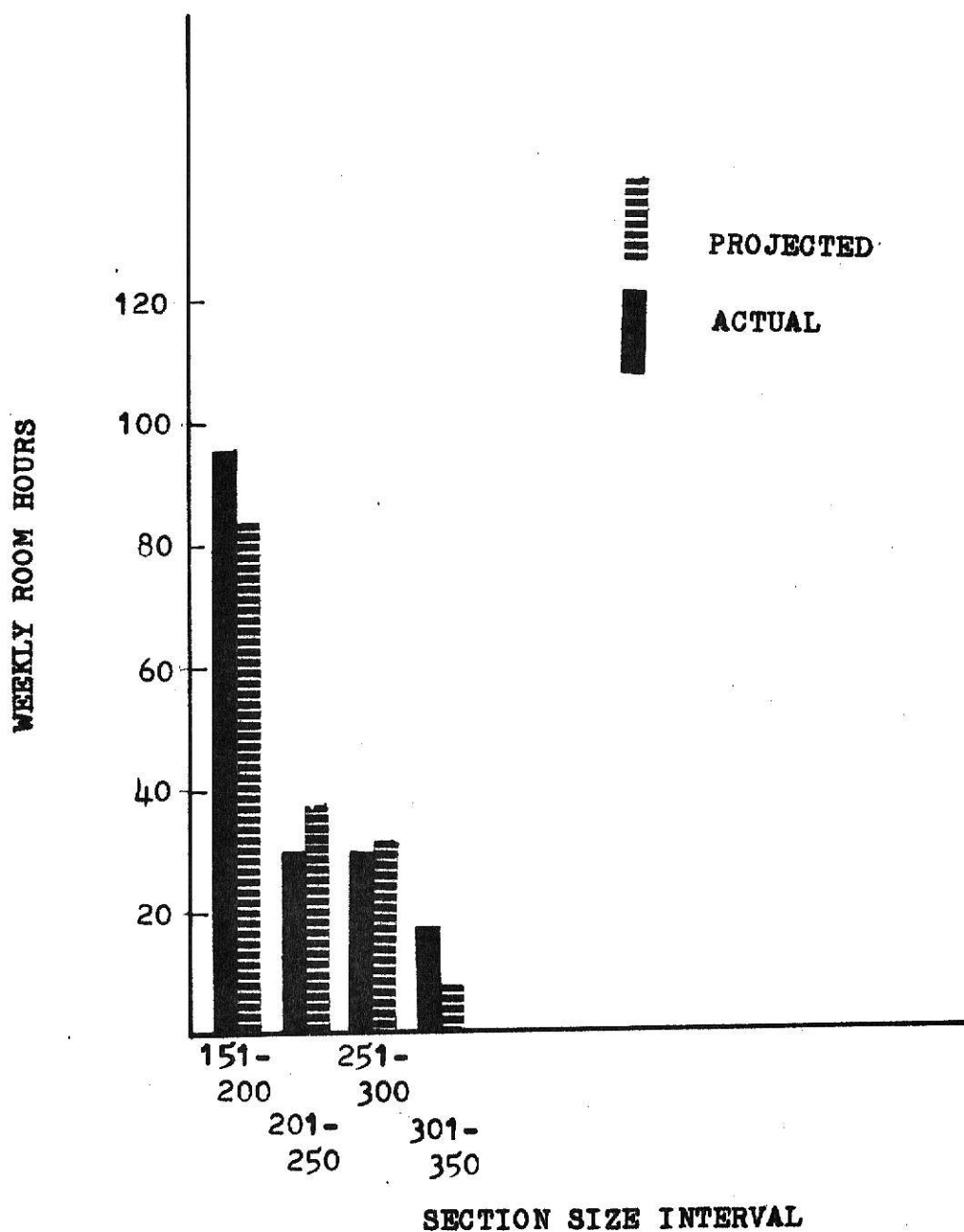


TABLE 3.7

DATA FOR FIGURE 3.6

Section Size Interval	Actual	Projected
151-200	95	83
201-250	29	37
251-300	21	31
301-350	17	7



and the sectioning rules sectioned the courses was, indeed, consistent with the way the courses were sectioned in the line schedule. The conclusion drawn is that the course data and sectioning rules can be used to project classrooms in this section size range, and, from an overall point of view, the sectioning rules and course data can be used to determine the classroom requirements for courses with sections in all section size ranges.

## CHAPTER 4

### RESULTS AND CONCLUSIONS

#### 4.1 INTRODUCTION.

Course data was compiled and sectioning rules were formed to enable the ICLM to be used to project classroom demands for a student population. The course data was compiled not with the intent of compiling the most sophisticated data possible but with the intent of compiling data which would be an initial data base in a continuing process of data compilation and refinement. An arbitrary set of sectioning rules was formed to enable a projection to be made, but it was not the purpose of this investigation to form an optimal set of rules. The objectives of the investigation were to demonstrate that:

1. An ICLM could be formed using data contained within Kansas State's data base.
2. Data on course facility requirements could be extracted from Kansas State's data base.
3. The method presented for projecting classroom requirements for a student population was reasonable and valid.

#### 4.2 THE INDUCED COURSE LOAD MATRIX.

Chapter 2 describes how an ICLM was successfully constructed from Kansas State's data base. The matrix constructed fulfilled the expectations of the investigation. However, as was noted in section 2.1.4 there remain several questions of interest which require further study and examination.

This investigation did not disclose any analytical or theoretical problems in constructing an ICLM: the construction of the matrix was suprisingly straight forward, and the utilization of an ICLM shows promise of being a useful tool of

educational administrators. The concept of an ICLM warrants further theoretical investigation and, most importantly, continual refinement and evaluation of the elements of the matrix.

The first priority for further investigation should be to show that, with a distribution of the student body in the aggregation expected of a projected enrollment, the ICLM can be used to give reasonable course enrollments. This can be done by using the ICLM with enrollment data for a fall semester other than 1970 in an attempt to generate course enrollments for that semester. Comparing these course enrollments with the actual enrollments should prove the worth of the matrix.

If further investigation is to be done requiring construction of more matrices, serious consideration should be given to incorporating CURRCOU into ONE and to re-writing the programs to operate in a more efficient fashion (efficient from a data processing point of view with re-programming emphasis on the input-output instructions of the computer programs).

#### 4.3 THE COURSE FACILITY REQUIREMENTS DATA.

Course data and its compilation were discussed in section 3.1. There were two reasons for compiling this data. The first was to provide data to be used in demonstrating and validating the procedure for projecting classroom requirements. The second was to provide an initial data base for further data compilation and refinement, should Kansas State's administration elect to continue investigating methods of projecting facilities requirements. The method used to compile the course data was secondary. The main point to be made is that data on the facility requirements of courses was assembled and that this data accurately represents the classroom facility requirements of courses.

The course data was compiled from the line schedule data. The line schedule data contained the implicit intentions of faculty and administrators concerned with determining course facilities requirements. It contained the results of their deliberations and compromises. What it did not contain and consequently the course data did not contain were their explicit free-choice preferences for facilities. This kind of information can only be determined by some sort of poll of the faculty and administration.

After the course data was compiled several observations could be made:

1. Basic classroom facility requirements for courses were successfully assembled from the line schedule data.
2. Programming the compilation of the course data was hampered by not knowing all the special cases which could arise. (An example was the need for kernel and augment information.)
3. Even though the line schedule data was never meant to be used as it was in this investigation, it was possible to extract its information on course facility requirements.
4. The compiled course data could be used as a base for further refinement of course facility requirements.

If more investigation necessitating the use of course facility requirement data is contemplated three recommendations are made:

1. Data on alternative course facility requirements needs to be compiled.
2. Procedures must be formed for continually reviewing and up-dating course facility requirements.
3. Consideration should be given to compiling course facility requirements for space other than classrooms.

#### 4.4 THE SECTIONING RULES.

The sectioning rules were explained in section 3.2. It must be emphasized that these rules were selected arbitrarily to make the projection procedure possible, and, as was presented in section 3.3, they successfully did this.

Sectioning rules are influenced by such factors as faculty and administrative preferences, available facilities, and, most importantly, by budgetary considerations. The sectioning rules may be viewed as parameters of the projection procedure subject to the control of administrators.

#### 4.5 VALIDATING THE COURSE DATA AND SECTIONING RULES.

Section 3.3 demonstrated the procedure for projecting a distribution of classroom weekly room hours from a student population. The demonstration was made to validate the course data and the sectioning rules. An actual distribution of classroom weekly room hours for the fall 1970 semester was amassed. Section 3.3.3 compared the actual distribution of classroom weekly room hours with the distribution formed when the procedure was demonstrated using fall 1970 student enrollments. As was expected, the procedure successfully duplicated the distribution of weekly room hours, validating the courses data and sectioning rules.

This was the simplest possible fashion in which to test the procedure. However, it would be interesting to make a projection (starting with a distribution of the student population) of the distribution of classroom weekly room hours for another semester, for example the fall 1971 semester. If this projection were made using the present (fall 1970) ICLM, sectioning rules, and course data and compared with the actual distribution for this other semester, more conclusive evidence on the merits of the projection method would be available. It is recommended that this test be included in any further investigation of this projection method.

#### 4.6 SUMMARY OF RESULTS.

The procedure studied in this investigation produced a distribution of weekly room hours of classroom space from a distribution of the student population. Further work needs to be done to convert this distribution of weekly room hours

into actual numbers and sizes of classrooms, and work needs to be done to evaluate present facilities in the light of projected weekly room hours. At the other end of the procedure, work has to be continued on schemes to accurately project student enrollments. The procedure described in this report can be used to solve part of the problem of determining facility requirements for a student population.

The results of this investigation are:

1. Kansas State's data base contains information which can be used to construct induced course load matrices.
2. The construction of an induced course load matrix for Kansas State is a feasible undertaking.
3. The usefulness of an induced course load matrix for predicting classroom facilities requirements has been demonstrated.
4. Classroom facility requirements for courses were determined from available data.
5. Alternative facility requirements for courses need to be compiled.
6. Procedures for continually reviewing and up-dating course facility requirements and the elements of the ICLM should be implemented.
7. It has been successfully demonstrated that realistic distributions of classroom weekly room hours can be produced when the procedure described is used in conjunction with the course data compiled and the sectioning rules formed.

## LIST OF REFERENCES

1. Bowman, Edward H. "A Budget Model of a University", Socio-Economic Planning Sciences, II (April, 1969), 175-178.
2. Hirsch, Werner Z. "Demand for Education, and Its Estimation", Socio-Economic Planning Sciences, II (April, 1969), 217-224.
3. Marshall, Kneale T. and Oliver, Robert M. "A Constant-Work Model for Student Attendance and Enrollment", Operations Research, XVIII (March-April, 1970), 193-206.
4. Smith, R. Longworth. "Accommodating Student Demand for Courses by Varying the Classroom-Size Mix", Operations Research, XIX (July-August, 1971), 862-874.
5. Harden, Warren R. and Teheng, Mike T. Projection of Student Enrollment by a Two-Step Markovian Process. Paper presented at the 39th meeting of the National Operations Research Society of America. Normal, Illinois: Office of Academic Planning and Institutional Research, Illinois State University, 1971.
6. Koenig, Herman E. and Keeney, Martin G. "A Prototype Planning and Resource Allocation Program for Higher Education", Socio-Economic Planning Sciences, II (April, 1969), 201-216.
7. Koenig, Herman E. "A Systems Model for Management, Planning, and Resource Allocation in Institutions of Higher Education", Journal of Engineering Education, 59 (April, 1969), 963-966.
8. Judy, Richard W. and Levine, Jack B. A New Tool for Educational Administrators. A report to the Commission on the Financing of Higher Education. Canada: University of Toronto Press, 1965.
9. Judy, Richard W. "Systems Analysis and University Planning", Socio-Economic Planning Sciences, II (April, 1969), 179-200.
10. Abell, V. A. et al. A Comprehensive University Scheduling System. A Presentation to the 10th College and University Machine Records Conference, Michigan State University, May 1965. Purdue University, March 1965.

11. Lovell, C. C. Student Flow Models: A Review and Conceptualization. Western Interstate Commission for Higher Education, 1971.
12. Dahnke, Harold L., et. al. Higher Education Facilities Planning and Management Manuals. Boulder, Colorado: Western Interstate Commission for Higher Education, November 1970.



## APPENDIX A

### DOCUMENTATION OF PROGRAMS

ONE

PURPOSE--This program was written to edit course, curriculum, and level data from the grade tape and to write this information onto a direct access data set.

METHOD--Curriculum and level designations for each student were determined from the short student master on the grade tape, and the courses each student took were determined from the course files. A record was written onto NEWTAPE for each course each student took. Each of these records contained course, curriculum, and level data. The procedure used to determine the level designation for a student is explained in section 2.2.1.2.

INPUT--The only input for this program was the grade tape, documented in Appendix B. The JCL for the grade tape as used in this program is listed at the end of the program listing (GO.ORGTAPE).

OUTPUT--The output of this program was written onto a direct access data set, NEWTAPE. A record of NEWTAPE contained the following information:

Bytes:

1-6 Course Number.

8-10 Curriculum Number.

12 Level.

13-19 Sequence number of record as given by ONE. This value was used only for padding the record to reach minimum record length.

This record was written with the following PL/1 format:

(P'999999',X(1),P'999',X(1),P'9',F(7));

The JCL requirements for NEWTAPE are listed at the end of the program listing (GO.NEWTAPE).



## ONE

## SOURCE LISTING

# **ILLEGIBLE DOCUMENT**

**THE FOLLOWING  
DOCUMENT(S) IS OF  
POOR LEGIBILITY IN  
THE ORIGINAL**

**THIS IS THE BEST  
COPY AVAILABLE**

```

// JCB (FH67U6CE,80,9),COMPTON,CLASS=B
// EXEC PL1LFCLG
//PL1L.SYSIN DD *
CNE: PROCEDURE OPTIONS(MAIN);
DECLARE ORGTAPE INPUT RECORD BUFFERED,
NEWTAPE OUTPUT RECORD BUFFERED,
NEW_DATA CHARACTER(20),
(FLAG1,FLAG) FIXED BINARY(1) INITIAL(0),
DATA CHARACTER(81),
      (SSN,OLDSSN) CHARACTER(9),
      CLASS FIXED DECIMAL(1),
      (OLDCOURSE,COURSE) PICTURE'9999999',
      CURRICULUM PICTURE'999',
      (CLASS_CODE(8) INITIAL('1'),LEVEL) CHARACTER(1),
      J FIXED DECIMAL(7);
ON ENDFILE(ORGTAPE) BEGIN:
GO TO STOP;
END;

CN CCVERSION BEGIN;
PUT SKIP EDIT('***POS. ERROR ',UNSOURCE,I,J,DATA) (A,A(10),F(5),X(3)),
      F(7),X(3),A(81));
IF FLAG=0 THEN DO;
  IF CNCHAR='H' THEN DO; ONCHAR='8'; GOTO END;ELSE;
  IF CNCHAR='B' THEN DO; ONCHAR='2'; GOTO END;ELSE;
  IF CNCHAR='C' THEN DO; ONCHAR='3'; GOTO END;ELSE;
  IF CNCHAR='F' THEN DO; ONCHAR='6'; GOTO END;ELSE;
  IF CNCHAR='I' THEN DO; ONCHAR='9'; GOTO END;ELSE;
  IF CNCHAR='A' THEN DO; ONCHAR='1'; GOTO END;ELSE;
  IF CNCHAR='G' THEN DO; ONCHAR='7'; GOTO END;ELSE;
  IF CNCHAR='D' THEN DO; ONCHAR='4'; GOTO END;ELSE;
  IF CNCHAR='E' THEN DO; ONCHAR='5'; GOTO END;ELSE;
L=L+1;
GOTO NEXT;
END;
ELSE DO; IF CNCHAR='P' THEN DO; ONCHAR='3'; GOTO END; ELSE;
  IF CNCHAR='T' THEN DO; ONCHAR='3'; GOTO END; ELSE;
  IF CNCHAR='-' THEN DO; UNSOURCE = '998' ; GO TO END; ELSE;
  CLASS I=SSN;

```

```

READ: READ FILE(URGTAPE) INTO (DATA);
      J=J+1;
      GET STRING(DATA) EDIT(SSN) (A(9));
      IF OLDSSN NE SSN THEN DO;
        GO TO START;
      END; ELSE;
        L=L+1;
        PUT SKIP EDIT('***COURSE NOT PROCESSED ',I,J,DATA) (A,F(5),X(3),
          F(7),X(3),A(81));
        GO TO READ;
      END;
      END: END;
OPEN  FILE(SYSPRINT) STREAM PRINT LINESIZE(130),
      FILE(NEWTAPE)OUTPUT,
      FILE(ORGTAPE) INPUT;
      CLASS_CODE(3)='2';
      CLASS_CODE(4)='2';
      CLASS_CODE(5)='2';
      CLASS_CODE(6)='2';
      J=1;
      I=0;
      K=1;
      L=0;
      N=0;
      READ FILE(ORGTAPE) INTO (DATA);
      GET STRING(DATA)EDIT(SSN)(A(9));
      START:
      FLAG=1;
      I=I+1;
      GET STRING(DATA) EDIT(CLASS,CURRICULUM) (X(29),F(1),X(24),P'999');
      CLC=SSN;
      IF CURRICULUM=702
        THEN IF 5 >= CLASS
          THEN LEVEL='1';
          ELSE LEVEL='2';
          ELSE LEVEL=CLASS_CODE(CLASS);
      COURSE='999999';

```

```

FLAC=0;
NEXT:
READ FILE(ORGTAPE) INTO (DATA);
J=J+1;
GET STRING(DATA) EDIT(SSN)(A(9));
IF OLDSSN /= SSN THEN GO TO START;
  OLDCOURSE=COURSE;
  GET STRING(DATA) EDIT(COURSE)(X(32),P'999999');
  IF CCOURSE=CLDCOURSE THEN DO;
    M=M+1;
    GOTO NEXT; END;
  PUT STRING(NEW_DATA)EDIT(COURSE,CURRICULUM,LEVEL,K)(P'999999',X(1),
    P'999',X(1),P'9',F(7));
  WRITE FILE(NEWTAPE) FROM (NEW_DATA);
  K=K+1;
GO TO TC NEXT;
STOP:
  CLCSE FILE(ORGTAPE),FILE(NEWTAPE);
  PUT SKIP EDIT('NUMBER OF RECORDS=',J) (A,F(7));
  PUT SKIP EDIT('NUMBER OF RECORDS WRITTEN ON NEWTAPE=',K-1)(A,F(7));
  PUT SKIP EDIT('NUMBER OF RECORDS NOT PROCESSED=',L)(A,F(7));
  PUT SKIP EDIT('NUMBER OF DUPLICATE COURSE NUMBERS=',M)(A,F(7));
  PUT SKIP EDIT('NUMBER OF STUDENTS=',I) (A,F(7));
  END ONE;

/*
//GC,CRGTAPE DD DSN=ORGTAPE,UNIT=TAPE7,DISP=(SHR,DELETE),
//          VOL=SER=18851,LABEL=(,BLP),
//          DCB=(LRECL=61,BLKSIZE=1215,RECFM=FB,DEN=1,TRTCH=TE,
//          BUFENC=8,OPTCD=C)
//GC,NEWTAPE DD DSN=COFH67,NEWTAPE,UNIT=2314,DISP=(OLD,KEEP),
//          SPACE=(2000,(1080,50)),VOL=SER=222222,
//          DCB=(RECFM=FB,LRECL=20,BLKSIZE=2000,BUFNO=8,OPTCD=C)
//
/*

```



# CURRCOU

PURPOSE--This program was written to determine the number of students in each curriculum and level category. This information was necessary for the formation of elements of the ICLM. This deck was used by ICLMOP1 and ICLMOP2. (See section 2.2.2.)

METHOD--This program read the short student masters on the grade tape and tabulated the number of students in each curriculum and level category. It produced a punched-card deck as output.

INPUT--The only input for this program was the grade tape, documented in Appendix B. The JCL for the grade tape as used in this program is listed at the end of the program listing (GO.ORGTAPE).

OUTPUT--A deck of punched-cards recording the number of students in each curriculum and level category. This deck contained the elements of the following data structure:

- 1 CUR\_INDEX(999,2),
- 2 CURCOUNT FIXED(5) INITIAL((1998) 0),
- 2 ALPHCUR CHARACTER(3) INITIAL ((1998)(1)' '),

The elements were initialized as above, but as the tabulation was performed, the appropriate element was indexed accordingly. CURCOUNT contained the count for a category and ALPHCUR contained the curriculum's abbreviation. Referencing an element of the data structure by a curriculum and level number gave the appropriate count or alphabetic code for a curriculum. For example, CURCOUNT(211,1) would give the number of level one undergraduate art students.

This data structure was punched with the following PL/1 code:

```
DO I= 1 TO 999;
DO I= 1 TO 2;
PUT FILE(PUNCH) EDIT(I,J,CUR_INDEX.ALPHCUR(I,J),
CUR_INDEX.CURCOUNT(I,J))(X(8),F(3),F(1),A(3),F(5));
END;
END;
```

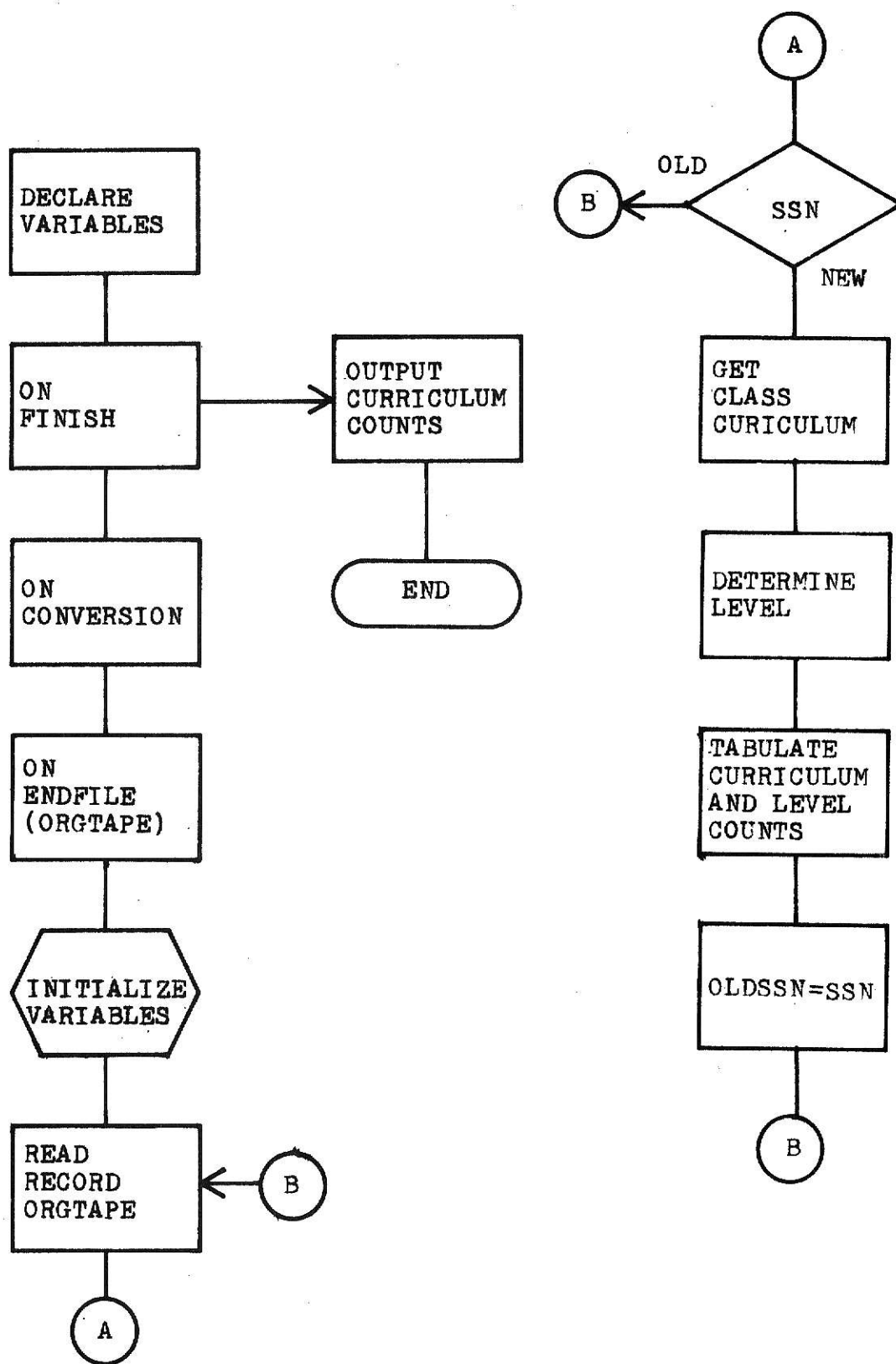


Figure A.2--Flowchart of program CURRCOU.

**CURRCOU**  
**SOURCE LISTING**

```

// JCB (FH67U6C2,15,1,600),COMPEN,CLASS=B
// EXEC PL1LFCLG
// PL1L,SYSIN CC *
CURRCOU: PROCEDURE OPTIONS(MAIN);
DECLARE
  1 CUR_INDEX(999,2),
  2 CURCOUNT FIXED(5) INITIAL((1998)0),
  3 ALPHCUR CHARACTER(3) INITIAL((1998)(1)' '),
  (CLASS,LEVEL,CLASS_CODE(8) INITIAL((8)1)) FIXED DECIMAL(1),
  K FIXED BINARY(31) INITIAL(0),
  CUR FIXED(3),
  CLDSSN CHARACTER(9),
  DATA BASED(P) CHARACTER(81),
  SSN BASED(P) CHARACTER(9),
  ALPH CHARACTER(3);
CN FINISH BEGIN;
  IF FLAG=1 THEN GO TO THERE;
  PUT SKIP EDIT('ERROR ON RECORD',K,DATA)(A,F(6),X(3),A(81));
  CC TO READ;
  THERE: END;
  FLAG=0;
CN CONVERSION BEGIN;
  IF ONCHAR='P' THEN DO; ONCHAR='3'; GO TO END; END;
  IF ONCHAR='T' THEN DO; ONCHAR='3'; GO TO END; END;
  IF ONCHAR='-' THEN DO; UNSOURCE='998'; GO TO END;END;
END: END;
CN ENDFILE(ORGTAPE) CALL QUIT;
CLDSSN='999999999';
PUT EDIT('CURRICULUM NUMBERS,ALPHBETIC CODE,AND COUNT')(X(10),A);
PUT SKIP;
CLASS_CODE(3)=2;
CLASS_CODE(4)=2;
CLASS_CODE(5)=2;
CLASS_CODE(6)=2;
  (ONLY FILE(ORGTAPE) STREAM OUTPUT;
  FILE: READ FILE(ORGTAPE) SET(P);
  PER4;
  IF NOV=CLDSSN THEN GO TO READ;

```

```

GET STRING(DATA) EDIT(CLASS,CUR,ALPH)(X(29),F(1),X(24),F(3),X(17),
A(3));
IF CUR=702 THEN IF >>CLASS
THEN LEVEL=1;
ELSE LEVEL=2;
ELSE LEVEL=CLASS_CODE(CLASS);
CUR_INDEX,CURCOUNT(CUR,LEVEL)=CUR_INDEX,CURCOUNT(CUR,LEVEL)+1;
CUR_INDEX,ALPHCUR(CUR,LEVEL)=ALPH;
OLUSSN=SSN;
GO TO READ;
QUIT: PROCEDURE;
DO I=1 TO 999;
DO J=1 TO 2;
PUT FILE(PUNCH) EDIT(I,J,CUR_INDEX,ALPHCUR(I,J),CUR_INDEX,CURCOUNT
(I,J))(X(8),F(3),F(1),A(3),F(5));
PUT EDIT(I,J,CUR_INDEX,ALPHCUR(I,J),CUR_INDEX,CURCOUNT(I,J))
(X(8),F(3),F(1),A(3),F(5));
END;
END: -END;
PUT SKIP EDIT('NUMBER OF RECORDS READ=',K)(A,F(5));
END QUIT;
FLAG=1;
STOP: END CURCOU ;
/*
//SU,CRTAPE DD DS=ORGTAPE,UNIT=TAPE7,DISP=(SHR,DELETE),
// VCL=SER=18851,LABEL=(,DLP),
// EOB=(LRECL=61,BLKSIZE=1215,RECFM=FB,DEN=1,
// PRTCH=TE,BUFNO=8,OPTCD=C)
//SU,PUNCH DD SYSOUT=B,DCB=BLKSIZE=80
/*

```

ICCMOP1

PURPOSE--This program was written to form the elements of the ICLM and to print the matrix by course.

METHOD--The basic procedure followed was to count all the records of one category (same course, same curriculum, and same level), form an element of the ICLM (divide this count by the total number of students in the curriculum level category), write the element of the matrix onto direct access storage, and print the element of the matrix in course order (print the matrix by row).

INPUT--Input to this program consisted of one direct access data set and three punched-card data sets. These data sets were:

1. Output from ONE sorted by course and, within each course, by curriculum and level. The contents of this data set are described in Appendix B under "Back-up Tape--Data Set 1". The JCL requirements for this data set are specified at the end of the program listing (GO.SORTED).
2. Curriculum counts given by CURRCOU. These curriculum counts are described in the program description for CURRCOU.
3. Department abbreviations for use in the printed output. The data for the department abbreviations consisted of the three digit department number and the five character department abbreviation. This data was read with the following PL/1 statement:

```
GET EDIT(J,DEPT_INDEX(J))(X(10),F(3),X(2),A(5));
```

where J is the department number and DEPT\_INDEX(J) is the department abbreviation. (The above PL/1 statement was executed four times per punched-card.)

4. Title cards consisting of identifying data which is to be printed with the output. This can be any number of cards as long as the last card is (starting in cc 1) "END DATA". These cards are read and printed with the following PL/1 statements:

```
DO WHILE(TITLE1 = 'END DATA');  
GET EDIT(TITLE)(A(80));  
IF TITLE = 'END DATA' THEN GOTO SKIP;  
PUT SKIP EDIT(TITLE)(X(17),A(80));  
SKIP:  END;
```

OUTPUT--Output for this program consisted of the following items:

1. A listing of department abbreviations.
2. A listing of the title cards.
3. A listing of the curriculum counts.
4. A print-out of the ICLM by course.
5. The ICLM on direct access storage. The format of records of this data set is explained in Appendix B under "Back-up Tape--Data Set 2". The JCL requirements for writing the ICLM onto direct access storage are given at the end of the program listing (GO.CURPROC).

A sample print-out for the ICLM listed by course is given in Figure A.3.

DEPT	COURSE	COUNT FRAC	CUR	LEV	COUNT FRAC	CUR	LEV	COUNT FRAC	CUR	LEV	COUNT FRAC	CUR	LEV
		4 .0307 SPC 998 1											
		***COURSE TOTAL= 1040											
281106		9 .1216 AG 005 1	1 .0166 AEC 010 1	1 .0454 AGR 015 1	1 .0509 NRC 016 1	5 .0458 AH 020 1							
		3 .1428 HRT 035 1	6 .1333 PF 036 1	1 .0714 RF 037 1	4 .0307 PVA 041 1	1 .0416 AED 075 1							
		1 .0136 AED 075 2	1 .0060 AR 115 1	1 .0045 AR 115 2	2 .0033 ARI 150 1	1 .0232 LAP 180 1							
		5 .0806 WL 201 1	1 .0499 BIO 205 1	6 .0618 ART 211 1	4 .0328 SP 212 1	2 .0384 ENG 214 2							
		3 .0652 ML 216 1	1 .1999 PHL 217 1	2 .3333 PHL 217 2	2 .1176 MUS 219 1	7 .1372 MTH 221 1							
		1 .0769 PHY 222 1	6 .1153 CS 223 1	3 .1250 CH 224 1	10 .0900 PLS 231 1	7 .0958 PLS 232 1							
		2 .1052 ANT 233 1	16 .1066 PSY 235 1	1 .0079 PSY 235 2	3 .1176 SCS 236 1	1 .0666 EC 236 1							
		59 .1755 PEL 270 1	21 .1399 PSD 271 1	9 .1754 MED 272 1	2 .0437 MED 272 2	10 .0925 MPE 273 1							
		2 .0185 MPE 273 2	7 .0823 APE 274 1	11 .1466 PRV 280 1	2 .2837 AMJ 281 1	3 .0025 BPM 282 1							
		5 .1086 PRO 283 1	12 .1498 PRL 285 1	6 .0340 PRV 286 1	6 .1532 PTH 287 1	3 .0483 BPM 288 1							
		15 .1665 TJ 289 1	167 .1960 GEN 290 1	8 .1095 BIS 291 1	2 .1111 HUM 292 1	7 .1372 SCS 294 1							
		1 .0178 SCS 294 2	5 .1315 SPA 295 1	1 .0555 PRP 296 1	5 .0377 TJ 299 1	44 .1078 BA 360 1							
		3 .0094 BA 360 2	20 .1639 HAA 361 1	3 .1153 PBE 376 1	1 .0393 FED 410 1	1 .0124 GE 500 1							
		1 .0199 CHE 520 1	3 .0441 CE 525 1	3 .0312 ME 560 1	18 .1374 TC 610 1	5 .2173 TC 612 1							
		10 .1098 TC 613 1	6 .0869 FCD 620 1	8 .1038 FCD 621 1	1 .0909 FCD 622 1	1 .0909 FCD 640 1							
		1 .3333 FN 641 1	72 .3272 NE 650 1	3 .1111 HEL 651 1	3 .1250 DIM 660 1	10 .1136 MED 672 1							
		1 .1250 HEJ 680 1	2 .0153 SPC 998 1										
		***COURSE TOTAL= 716											
281107		1 .0060 AR 115 1	1 .0714 ARS 120 1	2 .0434 ML 216 1	1 .0056 PRV 286 1	3 .0593 CHE 520 1							
		1 .0147 CE 525 1	1 .0075 EE 530 1	1 .0076 SPC 998 1									
		***COURSE TOTAL= 11											
281108		1 .0029 PEL 270 1	1 .0208 BMT 282 1	1 .0217 PRD 283 1	1 .0129 PRL 285 1	1 .0056 PAV 286 1							
		3 .0035 GEN 290 1	1 .0555 HUR 292 1	1 .0124 GE 500 1	2 .0399 CHE 520 1	1 .0075 EE 530 1							
		2 .0208 ME 560 1	1 .0045 ME 650 1	1 .0416 DIM 660 1									
		***COURSE TOTAL= 17											
281109		1 .0161 WL 201 1	1 .1111 ZOC 208 1	1 .0103 ART 211 1	1 .0217 ML 216 1	1 .0555 MUS 218 1							
		2 .0332 PTH 221 1	1 .0136 HIS 232 1	1 .0066 PSY 235 1	1 .0147 SCC 236 1	5 .0333 PEL 271 1							
		2 .0387 MED 272 1	2 .0235 APE 274 1	1 .0133 PRN 280 1	1 .0358 BMT 282 1	3 .0337 PRL 283 1							
		3 .0170 PRV 286 1	11 .0129 GEN 290 1	1 .0136 BIS 291 1	1 .0779 PSC 293 1	1 .0555 PRP 296 1							
		1 .0024 BA 360 1											
		***COURSE TOTAL= 42											
281110		1 .0045 AR 115 2	1 .0161 WL 201 1	1 .0024 BA 360 1	1 .0147 CE 525 1	1 .0415 DIM 660 1							
		***COURSE TOTAL= 5											
281120		1 .2500 BCH 060 1	15 .3333 SP 212 1	5 .1785 SP 212 2	1 .0217 ML 216 1	3 .0437 PLS 231 1							
		1 .0136 HIS 232 1	2 .0215 HIS 232 2	3 .0199 PSY 235 1	1 .0079 PSY 235 2	1 .0455 EC 236 1							
		1 .0029 PEL 270 1	2 .0133 PSD 271 1	1 .0208 BMT 282 1	4 .0514 PRL 285 1	1 .0455 PLS 232 1							
		1 .0256 BPM 298 2	8 .0093 GEN 290 1	1 .0136 BIS 291 1	1 .0355 HUM 292 1	1 .0769 PSC 293 1							
		1 .0243 SPA 295 1	1 .0175 TJ 299 1	1 .0277 SED 422 2	1 .0124 GE 500 1	2 .0333 CHE 520 1							
		1 .1250 HEJ 680 1	1 .0076 SPC 998 1										
		***COURSE TOTAL= 62											
281145		1 .3333 AJL 045 1	20 .4444 SP 212 1	2 .0714 SP 212 2	1 .0179 ENG 214 1	1 .0192 ENG 214 2							
		1 .0132 WL 216 2	1 .0066 PSY 235 1	1 .0147 SCC 236 1	3 .0139 PSD 271 1	1 .0335 VEC 272 1							
		1 .0184 BPM 288 2	5 .0058 GEN 290 1	1 .0399 HUM 292 2	3 .0326 TJ 299 1	1 .0244 TJ 299 2							
		1 .0004 BA 360 1	1 .5000 SED 429 1	1 .0238 ME 560 2	1 .0175 TC 613 2	1 .0409 PA 640 1							
		***COURSE TOTAL= 48											
281176		2 .0361 AEC 010 2	1 .0714 BM 058 1	1 .2500 BCH 060 1	1 .5000 AR 115 1	1 .0455 PLS 231 1							
		2 .0444 SP 212 1	1 .0199 ENG 214 1	7 .0693 PLS 231 1	1 .0136 HIS 232 1	5 .0333 PEL 271 1							

Figure A.3--A sample page of the ICLM listed by course.



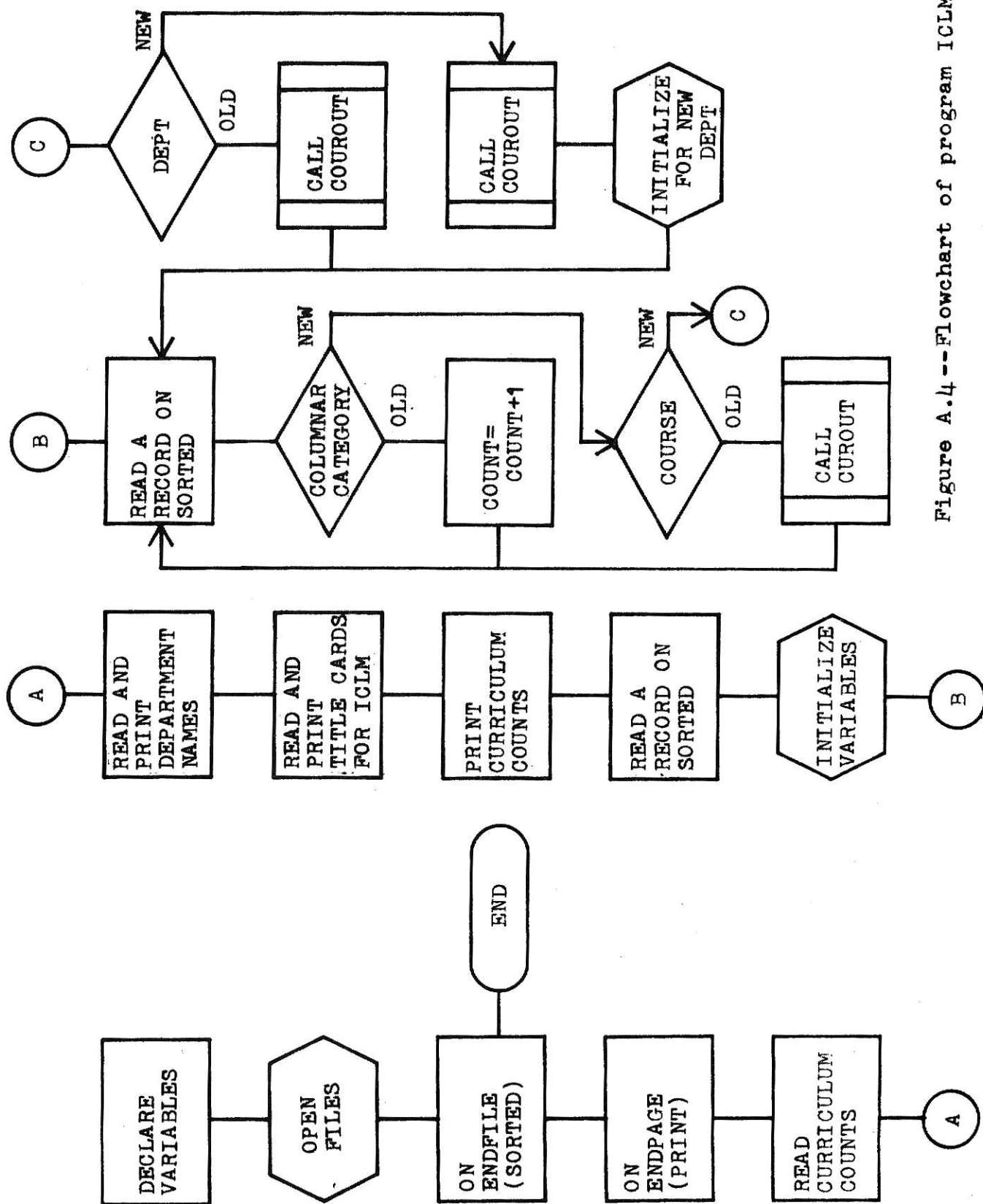


Figure A.4 --Flowchart of program ICLMOP1.

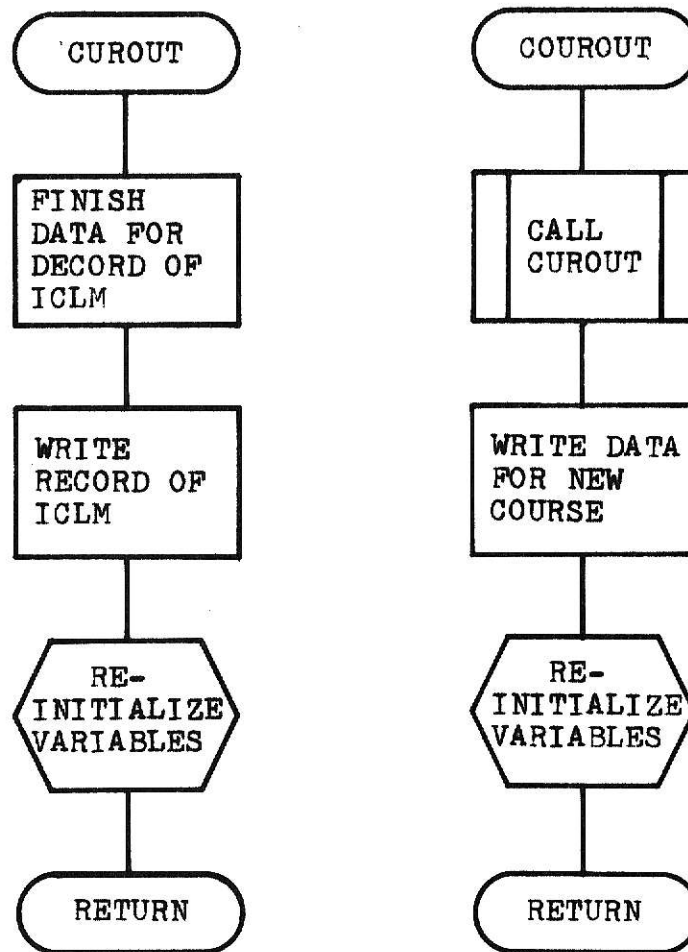


Figure A.4 --(Continued).

ICLMOP1

SOURCE LISTING

```

JCB (ACA0S660,40,10),COMPTON,CLASS=8

// EXEC PL1LFCLG
//PL1L.SYSIN DD *
ICLMP1: PROCCEDURE OPTIONS(MAIN);
DECLARE
  1 DATA BASED(P),
    2 COURSE CHARACTER(6),
    2 BLANK CHARACTER(1),
    2 CUR_LEV CHARACTER(5),
    DEPT BASED(P) CHARACTER(3),
    CCOURSE_CUR_LEV CHARACTER(12) BASED(P),
    (COURSE,CCOUNT) FIXED BINARY(15) INITIAL(0),
    III FIXED BINARY(15) INITIAL(5),
    K FIXED BINARY(31),
    1 CUR_INDEX(999),
    2 CURCOUNT(2) FIXED(5),
    2 ALPH_CUR CHARACTER(3) INITIAL((999)(1)' '),
    ALPH CHARACTER(3),
    HOLD1(132) CHARACTER(40) INITIAL((132)(40)' '),
    HCLD2(112) CHARACTER(30) INITIAL((112)(30)' '),
    DEPT_INDEX(0:750) CHARACTER(5) INITIAL((751)(1)'*****'),
    TITLE CHARACTER(80),
    OLDCOURSE CHARACTER(6),
    CLDCUR_LEV CHARACTER(5),
    GLDDEPT CHARACTER(3),
    OLDCOURSE_CUR_LEV CHARACTER(12),
    1 OUTDATA BASED(Q),
    2 COURS CHARACTER(6),
    2 CUR PICTURE'999',
    2 LEV PICTURE'9',
    2 CURALPH CHARACTER(3),
    2 FRAC PICTURE'V.9999',
    2 COUNTED FIXED BINARY(15),
    F FLCAT DECIMAL(4),
    KK FIXED BINARY INITIAL(0),
    CCUR FIXED DECIMAL(3),
    LLEV FIXED DECIMAL(1);
  OPEN FILE(SYSPRINT) STREAM PRINT LINESIZE(120),

```

```

FILE(PRINT) STREAM PRINT LINESIZE(130) PAGESIZE(59),
FILE(SORTED) INPUT RECORD SEQUENTIAL,
FILE(CURPROC) OUTPUT RECORD SEQUENTIAL;

/**/
/* ON UNITS */
CN ENDFILE(SORTED) BEGIN;
  LOCATE OUTDATA FILE(CURPROC);
  GET STRING(OLDCUR_LEV) EDIT(CCUR,LLEV) (F(3),X(1),F(1));
  CUR=CCUR;
  CURALPH=CUR_INDEX.ALPH_CUR(CCUR);
  LEV=LLEV;
  COUNTED=COUNT;
  CCURS=OLDCOURSE;
  C=COUNT;
  CC=CUR_INDEX.CURCOUNT(CCUR,LLEV);
  F=C/CC;
  FRAC=F;
  IF III=5 THEN DO;
    PUT FILE(PRINT) EDIT(COUNT,FRAC,CURALPH,CUR,LEV,'')(COLUMN(16),
      X(1),F(5),X(1),P'V.9999',X(1),A(3),X(1),P'999',X(1),F(1),A);
    III=1;
  END;
  ELSE DO;
    PUT FILE(PRINT) EDIT(COUNT,FRAC,CURALPH,CUR,LEV,'')(
      X(1),F(5),X(1),P'V.9999',X(1),A(3),X(1),P'999',X(1),F(1),A);
    III=III+1;
  END;
  COURSEC=COURSEC+COUNT;
  PUT FILE(PRINT) SKIP EDIT('**COURSE TOTAL=',COURSEC)(X(16),A,F(5));
  PUT FILE(PRINT) PAGE EDIT('NUMBER OF RECORDS READ=',K)(A,F(10));
  PUT FILE(PRINT) PAGE EDIT('NUMBER OF RECORDS SKIPPED DUE TO BUFFER
    PADDING=',KK)(A,F(7));
  CLOSE FILE(SORTED),
  FILE(PRINT),
  FILE(CURPROC);
GO TO STOP;
END;
CN ENDPAGE( PRINT) BEGIN;

```

```

PUT FILE(PRINT) PAGE EDIT('DEPT','COURSE')(X(1),A(4),X(2),A(8));
DO I=1 TO 5;
PUT FILE(PRINT) EDIT('COUNT FRAC CUR LEV')(A(23));
END;
PUT FILE(PRINT) SKIP(2);
END;
/**/
/* READ IN VALUES OF CURRICULUM INDEX */
DO II=1 TO 999;
DO JJ=1 TO 2;
GET EDIT(I,J,ALPH
          (X(8),F(3),F(1),A(3),F(5));
          ,CUR_INDEX.CURCOUNT(I,J))
IF CUR_INDEX.ALPH_CUR(I)=, THEN CUR_INDEX.ALPH_CUR(I)=ALPH;
END;
END;
GET SKIP;
/**/
/* READ IN VALUES OF DEPARTMENT INDEX*/
START: GET EDIT(J,DEPT_INDEX(J)) (X(10),F(3),X(2),A(5));
      PUT EDIT(J,DEPT_INDEX(J)) (X(10),F(3),X(2),A(5));
      IF J -> 750 THEN GO TO START;
      GET SKIP;
      PUT PAGE;
/**/
/* PRINT TITLE OF REPORT */
DO WHILE( TITLE -> 'END DATA');
GET EDIT(TITLE)(A(80));
IF TITLE='END DATA' THEN GOTO SKIP;
PUT SKIP EDIT(TITLE)(X(17),A(80));
SKIP: END;
/**/
/* PRINT TOTAL CURRICULUM COUNTS */
PUT PAGE;
PUT SKIP(2); EDIT('STUDENT COUNTS BY CURRICULUM AND LEVEL')(X(40),A);
PUT EDIT('UNDERGRADUATE CURRICULUMS')(SKIP(3),X(46),A );
PUT SKIP(3);
DO I=1 TO 3;
PUT EDIT('COUNT',')(X(20),A(5),X(14),A);

```

```

END;
DO I=1 TO 3;
  PUT EDIT('CURR.,',LEVEL 1,'LEVEL 2', ' ')(X(9),A(5),X(1),A(7),
    X(2),A(7),X(8),A);
END;
J=0;
DO I=1 TO 799;
  IF CUR_INDEX.ALPH_CUR(I)=' ' THEN GO TO END;
  J=J+1;
  PUT STRING(HOLD1(J)) EDIT(CUR_INDEX.ALPH_CUR(I),I,
    CUR_INDEX.CURCOUNT(I,1),CUR_INDEX.CURCOUNT(I,2))(X(7),
    A(3),X(1),P'999',X(1),F(5),X(4),F(5));
END;
END;
DO I=1 TO 44;
  PUT EDIT(HOLD1(I),HOLD1(I+44),HOLD1(I+88))(3(A(40)));
END;
PUT PAGE;
PUT SKIP(2) EDIT('GRADUATE CURRICULUMS: MASTER AND PHD ')(X(35),A);
PUT SKIP EDIT(' (SPECIAL STUDENTS ARE CLASSIFIED UNDER CURRICULUM 998
  )(X(27),A );
PUT SKIP(3);
DO I=1 TO 4;
  PUT EDIT('CURR.,',COUNT', ' ')(X(10),A(5),X(2),A(5),X(7),A(1));
END;
PUT SKIP(2);
J=0;
DO I=800 TO 999;
  IF CUR_INDEX.ALPH_CUR(I)=' ' THEN GO TO END3;
  J=J+1;
  PUT STRING(HOLD2(J)) EDIT(CUR_INDEX.ALPH_CUR(I),I,
    CUR_INDEX.CURCOUNT(I,1)+CUR_INDEX.CURCOUNT(I,2))
    (X(8),A(3),X(2),F(3),X(1),F(5))
    ;
END3: END;
DO I=1 TO 28;
  PUT EDIT(HOLD1(I),HOLD2(I+28),HOLD2(I+56),HOLD2(I+84))(4(A(36)));
END;
CLOSE FILE(SYSDEPT);
/ **/

```

```

/* PRINT OUT HEADINGS FOR ICLM */
PUT FILE(PRINT) EDIT('INDUCED COURSE LUAF MATRIX--LISTED BY COURSE')
  (X(25),A);
PUT FILE(PRINT) SKIP(4);
/**/
/* PRINT COLUMN HEADINGS */
PUT FILE(PRINT) EDIT('DEPT','COURSE')(X(1),A(4),X(2),A(8));
LC I= 1 TO 5;
PUT FILE(PRINT) EDIT(' COUNT FRAC CUR LEV')(A(23));
END;
PUT FILE(PRINT) SKIP;
/**/
/* START THE ACTUAL PROGRAM */
READ FILE(SORTED) SET(P);
K=1;
CLDCOURSE=COURSE;
OLDCUR_LEV=CUR_LEV;
OLCDEPT=DEPT;
CLDCOURSE_CUR_LEV=COURSE_CUR_LEV;
PUT FILE(PRINT) SKIP EDIT(DEPT_INDEX(DEPT),COURSE,'')(A(5),X(2),
  A(6),X(1),A);
COUNT=COUNT+1;
/**/
/**/
/**/
/* THE PROGRAM WILL LOOP FROM THIS POINT ON */
READ: READ FILE(SORTED) SET(P);
K=K+1;
IF COURSE='999999' THEN DO;
  KK=KK+1;
  GO TO READ; END; ELSE;
IF OLDCOURSE_CUR_LEV =COURSE_CUR_LEV
/* TO DO CLASSIFICATION */
  THEN DO;
    COUNT=COUNT+1;
    GO TO READ;
  END;
  LET IF CLDCOURSE=COURSE

```



```

/* NEW CUR_LEV SO OUTPUT CUR_LEV */
THEN DO;
  CALL CUROUT;
  CUROUT: PROCEDURE;
  LOCATE OUTDATA FILE(CURPROC);
  GET STRING(ULDCUR_LEV) EDIT(CCUR,LLEV) (F(3),X(1),F(1));
  CUR=CCUR;
  CURALPH=CUR_INDEX.ALPH_CUR(CCUR);
  LEV=LLEV;
  CCOUNTED=CCOUNT;
  CCURS=ULDCOURSE;
  C=CCOUNT;
  CC=CUR_INDEX.CURCOUNT(CCUR,LLEV);
  F=C/CC;
  FRAC=F;
  IF III=5 THEN DO;
    PUT FILE(PRINT) EDIT(CCOUNT,FRAC,CURALPH,CUR,LEV,'')(COLUMN(16),
      X(1),F(5),X(1),P'V,5999',X(1),A(3),X(1),P'999',X(1),F(1),A);
    III=1;
  END;
  ELSE DO;
    PUT FILE(PRINT) EDIT(CCOUNT,FRAC,CURALPH,CUR,LEV,'')(
      X(1),F(5),X(1),P'V,9999',X(1),A(3),X(1),P'999',X(1),F(1),A);
    III=III+1;
  END;
  COURSEC=COURSE+COUNT;
  CCOUNT=1;
  CURCOURSE_CUR_LEV=COURSE_CUR_LEV;
  FILECUR_LEV=CUR_LEV;
  ARE CUROUT;
  GO TO READ;
END;
LOCATE;
IF LLEVPRI=DEPT
/* CURSEC CURALPH */
/* F(3) X(1) F(1) */
CALL CCOUNT;
CCOUNT: PROCEDURE;

```

```

CALL COUROUT;
PUT FILE(PRINT) SKIP EDIT( '***COURSE TOTAL=',COURSEC)(X(16),A,F(5));
PUT FILE(PRINT) SKIP;
COURSEC=0;
OLDCOURSE=COURSE;
END COUROUT;
PUT FILE(PRINT) SKIP EDIT(COURSE)(X(7),A(6));
III=5;
GO TO READ;
END;
ELSE DO;
  IF DEPT_INDEX(OLDDDEPT)=DEPT_INDEX(DEPT) THEN DO;
    CALL COUROUT;
    OLDDDEPT=DEPT;
    PUT FILE(PRINT) SKIP EDIT(COURSE)(X(7),A(6));
    III=5;
    GO TO READ;
  END;
ELSE DO;
  CALL COUROUT;
  OLDDDEPT=DEPT;
  PUT FILE(PRINT) SKIP EDIT(DEPT_INDEX(DEPT),COURSE)(A(5),X(2),A(6));
  III=5;
  GO TO READ;
END;
END;
END;
END;
STEP1: END PROGRAM;
/*
//CL,SYST,PD

```

\*\*\*\*\*

CARDS CONTAINING  
IN FORM OF STUDENTS IN EACH CURRICULUM AND LEVEL CATEGORY

\*\*\*\*\*

\*\*\*\*\*

# CARDS CONTAINING INFORMATION TO BE USED AS TITLE CARDS ON THE OUTPUT

\*\*\*\*\*

```

/*
//GO,SORTED DD DSN=COFH67,SORTED,UNIT=2314,DISP=(OLD,KEEP),
//  SPACE=(2000,(1080,50)),VOL=SER=222222,
//  DCB=(RECFM=FB,LRECL=20,BLKSIZE=2000)
//GO,PRINT DD SYSOUT=A
//GC,CURPRC DD DSN=COFH67,CURPRC,UNIT=2314,DISP=(OLD,KEEP),
//  SPACE=(2000,(200,50)),VOL=SER=222222,DCB=(RECFM=FB,LRECL=20,
//  BLKSIZE=2000,BUFNO=4,OPTCD=C)
/*

```

## ICLMOP2

PURPOSE--This program was written to print the elements of the ICLM in curriculum-level order.

METHOD--The elements of the ICLM written onto direct access storage by ICLMOP1 were sorted by curriculum and level and, within each curriculum and level category, by course. It was this new, sorted data set that was printed by this program. The program recognized when a new curriculum was being processed and identified each new curriculum in the output.

### INPUT--

1. Curriculum counts--These cards are explained in this Appendix under "CURRCOU".
2. Title cards--These cards are explained in this Appendix under "ICLMOP1".
3. Departmental abbreviations--These cards are explained in this Appendix under "ICLMOP1".
4. Curriculum names--These cards gave the full name of each curriculum. They were used to identify sections of the output and are listed with it. These cards were read with the following PL/1 statement:

```
GET SKIP EDIT(I,CURNAME(I))(X(6),F(3),X(3),A(39));
```

where I is the three digit curriculum number and  
CURNAME(I) is the complete curriculum name.

5. ICLM--This was the ICLM on direct access storage. The JCL for this data set is given at the end of the program listing (GO.CURPROC). The format of records of this data set is explained in Appendix B under "Back-up Tape--Data Set 2".

OUTPUT--Output consisted of an echo-check of items 1-4 above and a listing of the ICLM by curriculum and level. A sample of this output is given in Figure A.5.



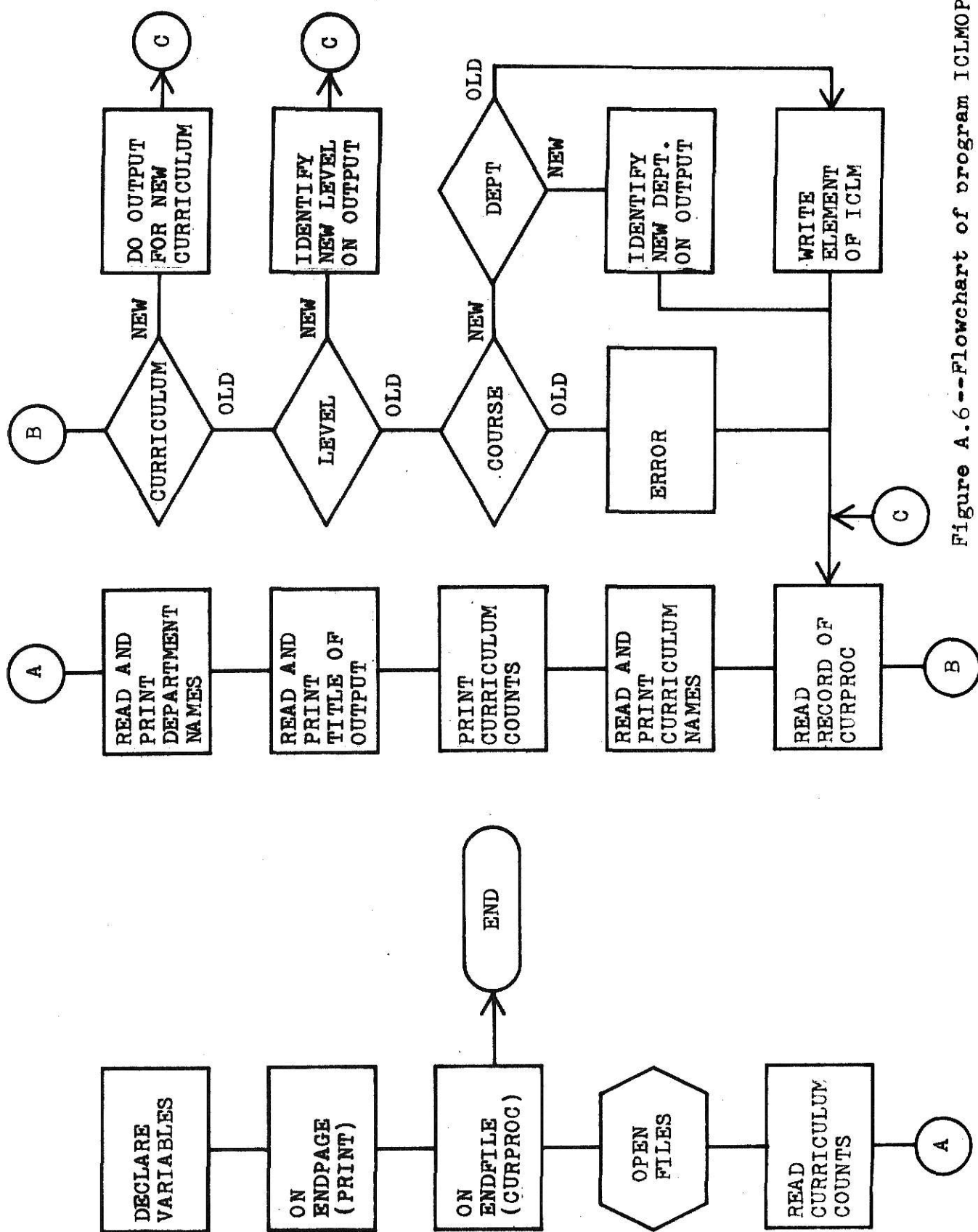


Figure A.6--Flowchart of program ICLMOP2.

## ICLMOP2

## SOURCE LISTING

```

// JC8 (ACAUS660,35,30,3),CUMPTON,CLASS=F
// EXEC PLILFCLG
//PLILSYSIN DD *
ICLMPC: PROCEDURE OPTIONS(MAIN);
DECLARE
  DEPT BASED(P) CHARACTER(3),
  K FIXED BINARY(31),
  1 CUR_INDEX(999),
    2 CURCOUNT(2) FIXED(5),
    2 ALPH_CUR CHARACTER(3) INITIAL((999)(1) ' '),
      ALPH CHARACTER(3),
HOLD1(132) CHARACTER(40) INITIAL((132)(40) ' '),
HOLD2(112) CHARACTER(30) INITIAL((112)(30) ' '),
  LEPT_INDEX(0:750) CHARACTER(5) INITIAL((751)(1) '*****'),
  TITLE CHARACTER(80),
  OLDCUR CHARACTER(3),
    1 CURDATA BASED(P),
      2 COURSE CHARACTER(6),
      2 CUR_PICTURE'999',
      2 LEV CHARACTER(1),
      2 ALPHCUR CHARACTER(3),
      2 FRAC_PICTURE'V.9999',
      2 COUNT FIXED BINARY(15),
    2 OLDCURDATA,
      2 CLDCOURSE CHARACTER(6),
      2 CLDCUR_PICTURE'999',
      2 OLDEY CHARACTER(1),
      2 CLDALPHCUR CHARACTER(3),
      2 CLDFRAC_PICTURE'V.9999',
      2 CLDCOUNT FIXED BINARY(15),
  CURNAME(999) CHARACTER(39),
  (KK,III) FIXED BINARY(15) INITIAL(0),
  CCCR FIXED DECIMAL(3);
  C ENDPAGE(PRINT) BEGIN;
  PUT FILE(PRINT) PAGE EDIT(' L=VAL DEPT')(A);
  C 1=1 TO 3;
  PUT FILE(PRINT) EDIT(' COUNT_COURSE___FRAC')(A);
  END;

```



```

PUT FILE(PRINT) SKIP(2);
END;
CN ENDFILE(CURPROC) BEGIN;
PUT FILE(PRINT) SKIP EDIT('***TOTAL NUMBER OF STUDENTS FOR THIS CUR
RICULUM AND LEVEL =',CURCOUNT(OLDCUR,OLDLEV))(A,F(5));
PUT FILE(PRINT) PAGE EDIT('NUMBER OF RECORDS READ=',K)(A,F(7));
PUT FILE(PRINT) PAGE EDIT('NUMBER OF DUPLICATE RECORDS=',KK)(A,F(7));
CLOSE FILE(PRINT);
FILE(CURPROC);
GO TO STOP;
END;
CPEN FILE(SYSPRINT) STREAM PRINT LINESIZE(120),
FILE(PRINT) STREAM PRINT LINESIZE(132) PAGESIZE(59),
FILE(CURPROC) INPUT RECORD SEQUENTIAL;
/**/
/* READ IN VALUES OF CURRICULUM INDEX */
DO II=1 TO 999;
DO JJ=1 TO 2;
GET EDIT(1,J,ALPH
(X(8),F(3),F(1),A(3),F(5)),CUR_INDEX,CURCOUNT(I,J))
IF CUR_INDEX.ALPH_CUR(I)=' ' THEN CUR_INDEX.ALPH_CUR(I)=ALPH;
END;
END;
GET SKIP;
/**/
/* READ IN VALUES OF DEPARTMENT INDEX*/
START: GET EDIT(J,DEPT_INDEX(J)) (X(10),F(3),X(2),A(5));
PUT EDIT(J,DEPT_INDEX(J)) (X(10),F(3),X(2),A(5));
IF J = 750 THEN GO TO START;
GET SKIP;
PUT PAGE;
/**/
/* PRINT TITLE OF REPORT */
DO WHILE( TITLE = 'END DATA');
GET EDIT(TITLE)(A(20));
IF TITLE='END DATA' THEN GO TO SKIP;
PUT SKIP EDIT(TITLE)(X(17),A(20));
SKIP;

```

```

/**/
/* PRINT TOTAL CURRICULUM COUNTS */
PUT PAGE;
PUT SKIP(2) EDIT('STUDENT COUNTS BY CURRICULUM AND LEVEL')(X(40),A);
PUT EDIT('UNDERGRADUATE CURRICULUMS')(SKIP(3),X(46),A );
PUT SKIP(3);
DO I=1 TO 3;
  PUT EDIT('COUNT', ' ')(X(20),A(5),X(14),A);
END;
DO I=1 TO 3;
  PUT EDIT('CURR.', 'LEVEL 1', 'LEVEL 2', ' ')(X(9),A(5),X(1),A(7),
  X(2),A(7),X(8),A);
END;
J=0;
DO I=1 TO 799;
  IF CUR_INDEX.ALPH_CUR(I)=' ' THEN GO TO END;
  J=J+1;
  PUT STRING(HOLD1(J)) EDIT(CUR_INDEX.ALPH_CUR(I),I,
  CUR_INDEX.CURCOUNT(I,1),CUR_INDEX.CURCOUNT(I,2))(X(7),
  A(3),X(1),P'999',X(1),F(5),X(4),F(5));
END;
DO I=1 TO 44;
  PUT EDIT(HOLD1(I),HOLD1(I+44),HOLD1(I+88))(3(A(40)));
END;
PUT SKIP(2) EDIT('UNDERGRADUATE TOTALS')(A);
PUT SKIP(2) EDIT('TOTAL LEVEL 1 = 6385')(A);
PUT SKIP EDIT('TOTAL LEVEL 2 = 5159')(A);
PUT SKIP EDIT('TOTAL UNDERGRADUATE= 11544')(A);
PUT SKIP(2) EDIT('GRADUATE CURRICULUMS: MASTER AND PHD')(X(35),A);
PUT SKIP EDIT('SPECIAL STUDENTS ARE CLASSIFIED UNDER CURRICULUM 998
')(X(17),A );
PUT SKIP(3);
DO I=1 TO 4;
  PUT EDIT('CURR.', 'COUNT', ' ')(X(10),A(5),X(2),A(5),X(7),A(1));
END;
PUT SKIP(2);
J=0;
DO I=900 TO 999;

```

```

IF CUR_INDEX,ALPH_CUR(I)=' ' THEN GO TO END3;
J=J+1;
PUT STRING(HOLD2(J)) EDIT(CUR_INDEX,ALPH_CUR(I),I,
(CUR_INDEX,CURCOUNT(I,1)+CUR_INDEX,CURCOUNT(I,2)))
(X(8),A(3),X(2),F(3),X(1),F(5)) ;
END3: END;
DO I=1 TO 28;
PUT EDIT(HOLD2(I),FOLD2(I+28),HOLD2(I+56),HOLD2(I+84))(4(A(30)));
END;
PUT SKIP(2)EDIT('GRADUATE TOTALS')(A);
PUT SKIP(2)EDIT('TOTAL MASTERS = 1420')(A);
PUT SKIP EDIT('TOTAL PHE = 710')(A);
PUT SKIP EDIT('TOTAL GRADUATE= 2130')(A);
PUT SKIP(3) EDIT('OVERALL TOTAL = 13674')(A);
/**/
/* READ IN DATA FOR CURRICULUM NAMES*/
PUT PAGE;
PUT EDIT ('CURRICULUM CODE NUMBERS', ' ')(X(41),A,SKIP(2),X(119),A);
GET: GET SKIP EDIT(I,CURNAME(I))(X(6),F(3),X(3),A(39));
PUT EDIT(I,CURNAME(I))(X(14),P'999',X(4),A(39));
IF I <= 998 THEN GO TO GET;
CLOSE FILE(SYSPRINT);
PUT FILE(PRINT) EDIT('INDUCED COURSE LOAD MATRIX--LISTED BY CURRICULUM
AND LEVEL')(X(35),A);
READ FILE(CURPRCC) SET(P);
K=1;
CLOSE CURDATA;
CLOSE DEPT;
CCUR=CUR;
PUT FILE(PRINT) SKIP(3) EDIT(CURNAME(CCUR), ' ',ALPHCUR,CUR,')(X(37),A
(39),SKIP(2),X(42),A,A(4),P'999',A);
PUT FILE(PRINT) SKIP(2) EDIT(' ' LEVEL DEPT')(A);
DO I=1 TO 5;
PUT FILE(PRINT) EDIT(' COUNT_COURSE____FRAC')(A);
END;
PUT FILE(PRINT) SKIP(2) EDIT('LEVEL ',LEV,DEPT_INDEX(DEPT),
COUNT,COURSE,FRAC)(X(3),A(6),A(1),X(2),A(5),F(8),X(2),A(6),X(2),P'V.999
9');

```

```

REAL:  READ FILE(CURPROC)SET(P);
      K=K+1;
      IF CUR ^= CLDCUR THEN GO TO CNE; ELSE;
      IF LEV ^= OLDLEV THEN GO TO TWO; ELSE;
      IF COURSE ^= OLDCOURSE THEN GO TO THREE; ELSE;
      PUT FILE(PRINT) SKIP EDIT('***TOTAL NUMBER OF STUDENTS FOR THIS C
      COURSE,CUR,LEV,ALPHCUR,FRAC,COUNT')(A,X(2),A(8),P'999',F(3),X(2),
      A(3),X(2),P'V.9999',F(7));
      PUT FILE(PRINT) SKIP EDIT('**ERROR**A DUPLICATE RECORD HAS BEEN FOUN
      D')(A);
      KK=KK+1;
      GO TO READ;
      CNE: CCUR=CUR;
      PUT FILE(PRINT) SKIP EDIT('***TOTAL NUMBER OF STUDENTS FOR THIS CUR
      RICULUM AND LEVEL =',CURCOUNT(OLDCUR,OLDLEV))(A,F(5));
      PUT FILE(PRINT) PAGE EDIT(CURNAME(CCUR),'(',ALPHCUR,CUR,')');
      (SKIP(3),X(37),A(39),SKIP(2),X(42),A,A(4),P'999',A);
      PUT FILE(PRINT) SKIP(2) EDIT(' LEVEL DEPT')(A);
      DO I=1 TO 5;
      PUT FILE(PRINT) EDIT(' COUNT_COURSE___FRAC')(A);
      END;
      PUT FILE(PRINT) SKIP(2) EDIT('LEVEL ',LEV,DEPT_INDEX(DEPT),COUNT,COURSE
      ,FRAC)(X(3),A(6),A(1),X(2),A(5),F(8),X(2),A(6),X(2),P'V.9999');
      III=1;
      OLDEPT=DEPT;
      CLDCURDATA=CURDATA;
      GO TO READ;
      TWO: PUT FILE(PRINT) SKIP EDIT('***TOTAL NUMBER OF STUDENTS FOR THIS C
      URRICULUM AND LEVEL =',CURCOUNT(OLDCUR,OLDLEV))(A,F(5));
      PUT FILE(PRINT) SKIP(3) EDIT('LEVEL ',LEV,DEPT_INDEX(DEPT),COUNT,COURSE
      ,FRAC)(X(3),A(6),A(1),X(2),A(5),F(8),X(2),A(6),X(2),P'V.9999');
      III=1;
      OLDEPT=DEPT;
      CLDCURDATA=CURDATA;
      GO TO READ;
      THREE: IF DEPT ^= OLDEPT THEN GO;
      PUT FILE(PRINT) SKIP EDIT(DEPT_INDEX(DEPT),COUNT,COURSE,FRAC)(CUL

```

```

LN(23),A(5),F(5),X(2),A(6),X(2),P,V,9999');
III=1;
CLCCEPT=DEPT;
CLC CURDATA=CURDATA;
GO TO READ;
END;
ELSE DO;
  PUT FILE(PRINT) EDIT(COUNT,COURSE,FRAC) (COLUMN(18+III*23),F(8),X(2)
  ,A(6),X(2),P,V,9999');
  IF III=4 THEN III=0;
    ELSE III=III+1;
  GO TO READ;
END;
STOP: END ICLMCP2;
/*
//GO.SYSIN DD *

```

\*\*\*\*\*

CARDS CONTAINING  
THE NUMBER OF STUDENTS IN EACH CURRICULUM AND LEVEL CATEGORY

\*\*\*\*\*

CARDS CONTAINING  
INFORMATION TO BE USED AS TITLE CARDS ON THE OUTPUT

\*\*\*\*\*

CARDS CONTAINING  
DEPARTMENTAL ABBREVIATIONS

\*\*\*\*\*

CARDS CONTAINING  
CURRICULUM NAMES, NUMBERS, AND ABBREVIATIONS

\*\*\*\*\*

```
//GL.CURPRC      DD DSN=COFFG7,ICLM,UNIT=2314,DISP=(OLD,KEEP),  
//              VOL=SER=222222,SPACE=(2000,(200,50)),DCB=(LRECL=20,  
//              BLKSIZE=1000,RECFM=FB,BUFNO=4,UPTCD=C)  
/*
```

CRSEDIT

PURPOSE--This program was written to edit course facility requirement data from the line schedule data (see section 3.1.5).

METHOD--This program processed each record of the line schedule data sequentially, determining the required course data.

INPUT--

1. Room data--These cards are explained in Appendix B under "Room Data".
2. Line schedule data--These are the line schedule cards as explained in Appendix B under "Line Schedule Data". These cards were available to this program on direct access storage. The JCL for this data set is given at the end of the program listing. The records of this data set consist of 80 character card images of each card in the line schedule data.

OUTPUT--

1. The edited course data was written by this program onto direct access storage. The JCL requirements for this data set are given at the end of the program listing (CRSDATA).
2. A print-out of the line schedule records for each course and the course data compiled on each course. A sample page of this output is given in Figure A.7.

COMMENT--This output data set was later punched onto cards for hand adjustments (see section 3.1.5). The final form for the course data is explained in Appendix B under "Course Data".

ALPHEER	NAME	CR TYPES R1 R2	TYPE SIZE	WRF	N REC	T LAB	RECM	RCOM
	281106 CPAL CCMM IAC3	3 0 0 0	4 255 1.CC00 5 R E					
	CC240052		SFCF 28110643030CRAL CCMM IAC3	F 1130	A1107			
	CC240000		SFCF 281106030250RAL CCMM IA	T T 0330	J 21			
	CC240000		SFCF 281106030250RAL CCMM IA	T T 0230	F 208			
	CC240000		SFCF 281106030270RAL CCMM IA	T T 1230	J 228			
	CC240000		SFCF 281106030260RAL CCMM IA	T T 0830	F 208			
	CC240000		SFCF 281106030250RAL CCMM IA	T T 0730	F 208			
	CC240000		SFCF 281106030240RAL CCMM IA	M W 0430	J 21			
	CC240000		SFCF 281106030230RAL CCMM IA	M W 0830	J 21			
	CC240000		SFCF 281106030220RAL CCMM IA	M W 0730	J 21			
	CC240000		SFCF 281106030210RAL CCMM IA	M W 0730	J 11			
	CC240000		SFCF 281106030200RAL CCMM IA	M W 0730	F 208			
	CC240000		SFCF 281106030190RAL CCMM IA	T T 0830	J 11			
	CC240000		SFCF 281106030200RAL CCMM IA	T T 1030	F 208			
	CC240000		SFCF 281106030310RAL CCMM IA	T T 1130	F 208			
	CC240000		SFCF 281106030300RAL CCMM IA	T T 0130	J 26			
	CC240000		SFCF 281106030290RAL CCMM IA	T T 0130	J 20			
	CC240000		SFCF 281106030340RAL CCMM IA	T T 1230	J 11			
	CC240000		SFCF 281106030330RAL CCMM IA	T T 1130	J 11			
	CC190000		SFCF 281106031800RAL CCMM IA	T T 0330	J 26			
	CC220000		SFCF 281106031750RAL CCMM IA	M W 0230	J 228			
	CC000018		SFCF 281106030630RAL CCMM IAC3M	M W F 0330	J 124			
	CC270001		SFCF 281106230640RAL CCMM IAC3M	M W F 1130	J 234			
	CC220000		SFCF 281106030370RAL CCMM IA	M W 0830	J 20			
	CC220004		SFCF 281106230450RAL CCMM IAC3M	M W F 0130	F 208			
	CC200004		SFCF 281106230440RAL CCMM IAC3M	M W F 1130	J 225			
	CC270001		SFCF 281106230420RAL CCMM IAC3	T T 09051030J	228			
	CC240000		SFCF 281106030360RAL CCMM IA	M W 0330	J 21			
	CC210003		SFCF 281106230420RAL CCMM IAC3M	M W F 1030	J 21			
	CC230000		SFCF 281106030410RAL CCMM IA	T T 0330	J 20			
	CC240000		SFCF 281106030400RAL CCMM IA	T T 0230	J 20			
	CC230000		SFCF 281106030350RAL CCMM IA	M W 0330	J 20			
	281106 CPAL CCMM IAC3	3 0 0 0	4 284 1.CC00 3 R L					
	CC000024		B 24 2.CC00 24 R E					
	CC000000		2 24 3.CC00 6 R E					
	281107 CRAL CCMM IAC3	2 1 1	SFCF 2811072311720RAL CCMM IAC3	T T 0930	J 26			
			SFCF 2811072311710RAL CCMM IAC3	M W F 0330	J 26			
	CC160032		2 E 2.CC00 1 E L J 26					
	281108 CRAL CCMM IAC2	1 0	E 3.CC00 1 E L J 26					
			SFCF 281108230470RAL CCMM IAC2	T 7000840J	20			
	CC230092		2 16 2.CC00 1 R E					
	CC230100		SFCF 281109230480RAL CCMM IAC4M	M W F 0830	F 208			
	281109 CRAL CCMM IAC4	1 0	SFCF 281109230450RAL CCMM IAC4M	M W F 1230	F 208			
	CC060012		2 24 3.CC00 2 R E					
	281110 TAC SPKG VCIC2	1 0	SFCF 281110230811TAC SPKG VCIC2APPT	CC				
	CC000057		2 6 C.CC00 1					
	281120 SF DRAM PART	1 0	SFCF 281120230500SF DRAM PART	APPT CC				
	CC530159		3 60 C.CC00 1					
	281145 INTR THEATREC3	1 0	SFCF 281145231211INTR THEATREC3M	M W F 0130	J 15			

Figure A.7--A sample page of the output of CRSEDT showing the line schedule data for each course and the course data compiled on each course.



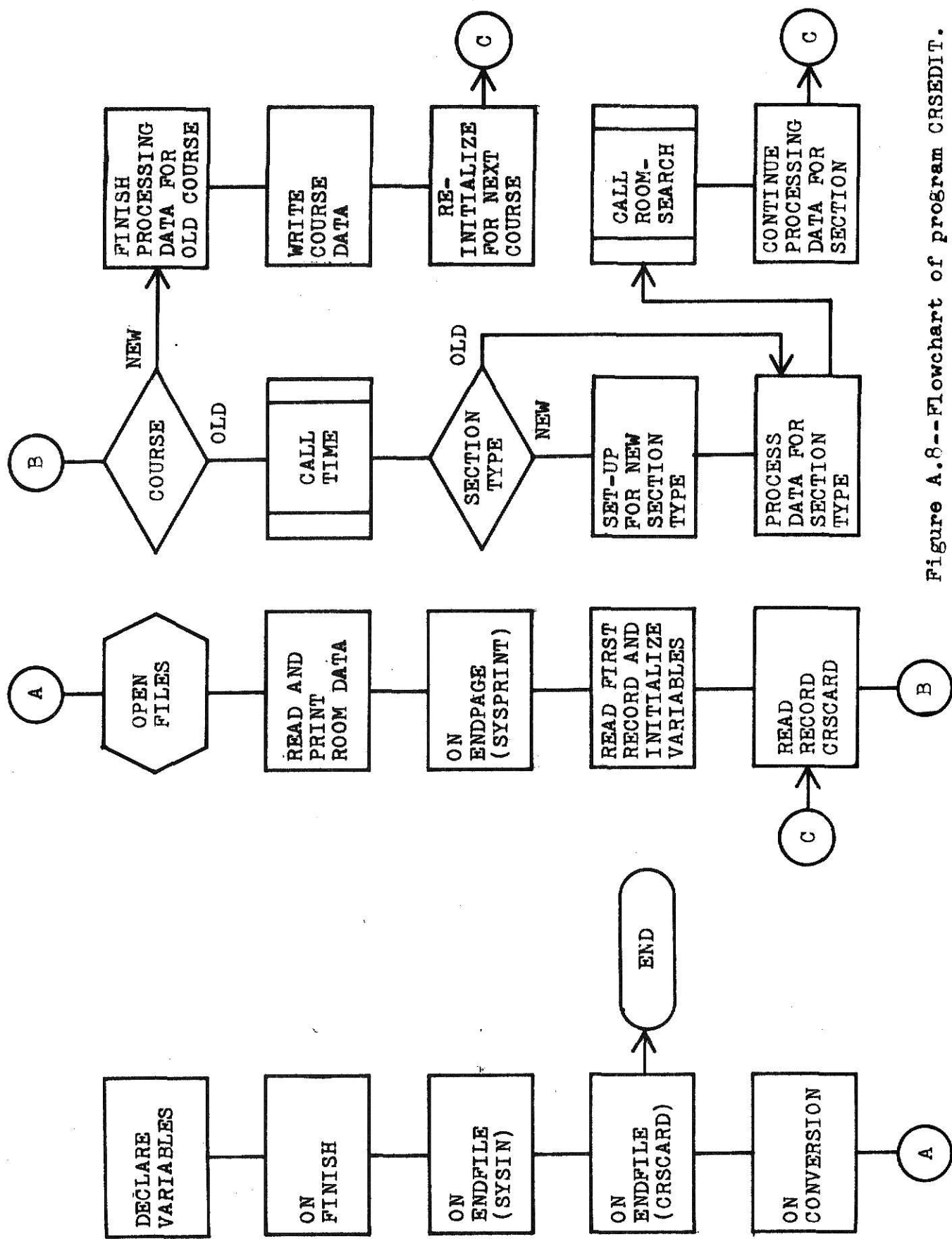


Figure A.8--Flowchart of program CRSEDIT.

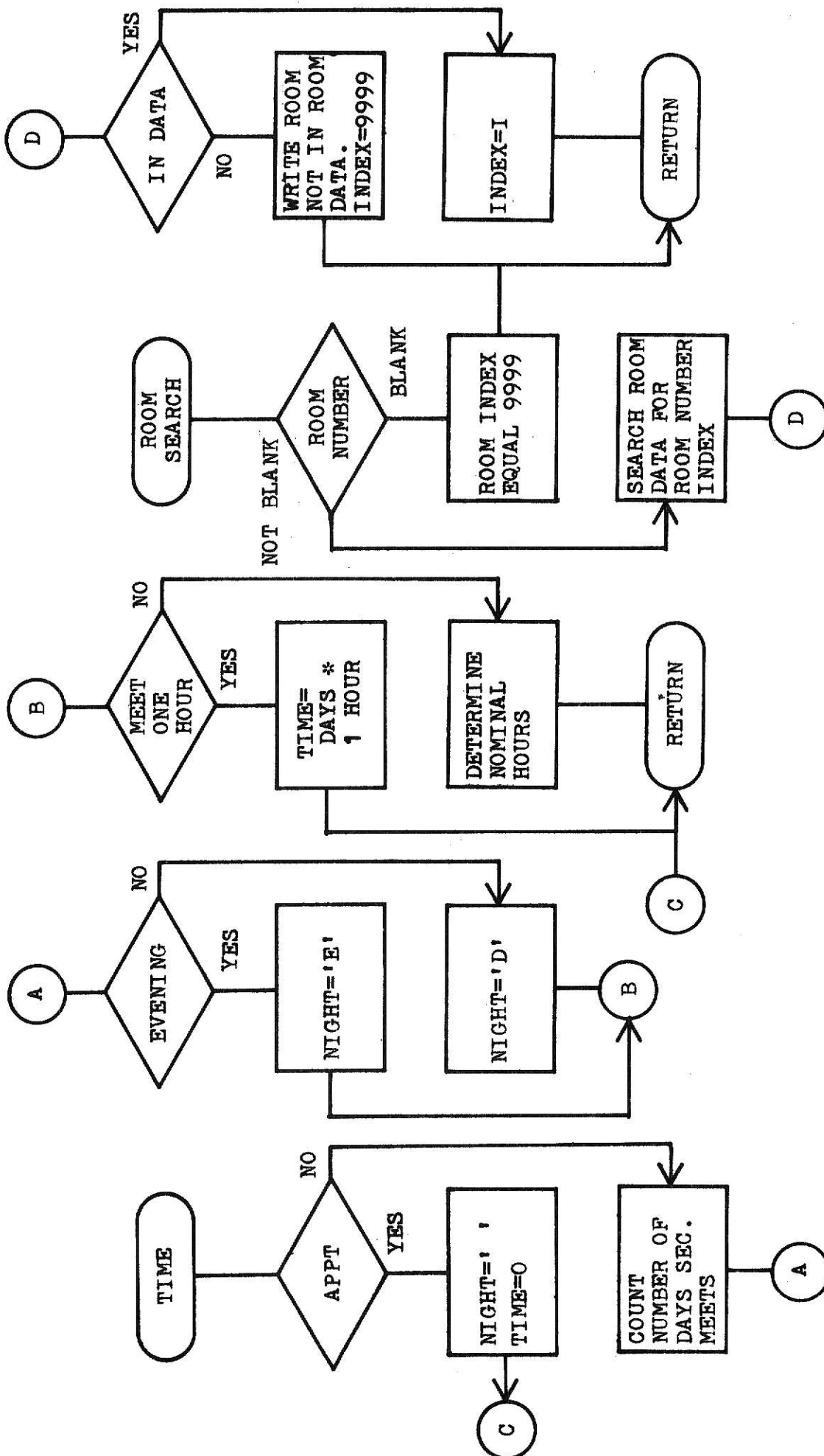


Figure A.8--(continued) Flowcharts of procedures ROOMSEARCH and TIME.

CRSEDIT

SOURCE LISTING

```

// JCB (FM67U6CZ,20,10),COMPTON,CLASS=B
// EXEC PL1LFCLG
//PL1L,SYSIN DD *
CRSECT: PROCEDURE OPTIONS(MAIN);
DECLARE
    INPUTRECORD CHARACTER(80) BASED(P),
    LINECOUNT FIXED BINARY(31),
    NUMBER PICTURE '999999',
    NAME CHARACTER(12),
    TYPECODE CHARACTER(1),
    CRHK CHARACTER(2),
    CAYS(6) CHARACTER(1),
    FROMHR FIXED BINARY(31),
    (TOHR,TCMIN,FROMMIN) FIXED BINARY(31),
    RCOMUSED CHARACTER(6),
    1 COURSEDATA,
    2 CCOURSEID,
    3 COURSE CHARACTER(6),
    3 ALPHAID CHARACTER(12),
    3 CR CHARACTER(2) INITIAL(' '),
    3 SECTYPES FIXED BINARY(15),
    2 COUNTROGMS(20) FIXED BINARY(15) INITIAL((20) 0),
    2 SECTYPE(15),
    3 SECDATA,
    4 SECTYPE CHARACTER(1),
    4 SECSIZE FIXED BINARY(31) INITIAL((15) 0),
    4 WRH FIXED BINARY(31,7) INITIAL((15) 0),
    4 QUANTITY FIXED BINARY(31) INITIAL((15) 0),
    4 REC CHARACTER(1) INITIAL((15) 1),
    4 EVENING CHARACTER(1),
    4 LAB CHARACTER(1) INITIAL((15) 1),
    3 ROOMS(20) CHARACTER(6),
    WS FIXED BINARY(31,7),
    CLOC(COURSE CHARACTER(6),
    TYPE(7)) CHARACTER(3),
    POUNDC(470) CHARACTER(6),
    POUNCAPD CHARACTER(60) BASED(Q),
    RECORDS FIXED BINARY(15) INITIAL(0),

```

```

COURSES FIXED BINARY(15),
NIGHT CHARACTER(1),
CUTDATA CHARACTER(2122) VARYING;
CN FINISH BEGIN;
  PUT SKIP EDIT('NUMBER OF RECORDS READ=',RECORDS)(A,F(6));
  PUT SKIP EDIT('NUMBER OF COURSES PROCESSED=',COURSES)(A,F(6));
END;
CN ENCFILE(SYSIN) BEGIN;
  IRMAX=1;
  FLAG=0;
END;
CN ENCFILE(CRSCARD) BEGIN;
  DO I=1 TO SECTYPES;
    SECSIZE(I)=SECSIZE(I)/QUANTITY(I);
    WRH(I)=WRH(I)/( QUANTITY(I));
    IF REC(I) ^= ' ' & LAB(I) ^= ' ' THEN DO;
      LAB(I)=' ';
      COUNTROOMS(I)=0;
    END;
  END;
END;
PUT SKIP EDIT(COURSEID,(COUNTROOMS(I) DO I=1 TO SECTYPES),(SECDATA(J),(
ROOMS(J,L) DO L=1 TO COUNTROOMS(J)), ' ' DO J=1 TO SECTYPES))(A(7),A(12
),A(2),
F(6),(SECTYPES) F(3),(SECTYPES) -SKIP,X(39),A(2),F(5),F(8,4),F(3),X(1),
2 A(2),(COUNTROOMS(J)) A(7),A(1));
PUT STRING(CUTDATA) EDIT(COURSEID,(COUNTROOMS(I) DO I=1 TO SECTYPES),(S
ECDATA(J),(ROOMS(J,L) DO L=1 TO COUNTROOMS(J)), ' ' DO J=1 TO SECTYPES
)(A(6),
A(13),A(2),F(1),(SECTYPES) F(2),(SECTYPES) (A(1),F(3),F(8,4),F(2),
3 A(1),(COUNTROOMS(J)) A(6),A(1));
WRITE FILE(CRSDATA) FROM (CUTDATA);
GO TO STOP;
END;
CN CONVERSION BEGIN;
  IF COURSEID=' ' THEN DO; UNSOURCE='00'; GOTO END; END;
  IF CCHAR='H' THEN DO; CNCHAR='0'; GOTO END; END;
  IF CCHAR='B' THEN DO; CNCHAR='2'; GOTO END; END;
  IF CCHAR='C' THEN DO; CNCHAR='3'; GOTO END; END;

```

```

IF CNCHAR='F' THEN DO; ONCHAR='6'; GOTO END;END;
IF CNCHAR='I' THEN DO; ONCHAR='9'; GOTO END;END;
IF CNCHAR='A' THEN DO; ONCHAR='1'; GOTO END;END;
IF CNCHAR='G' THEN DO; ONCHAR='7'; GOTO END;END;
IF CNCHAR='D' THEN DO; ONCHAR='4'; GOTO END;END;
IF CNCHAR='E' THEN DO; ONCHAR='5'; GOTO END;END;

END; END;

OPEN FILE(SYSIN) INPUT SEQUENTIAL RECDR BUFFERED,
FILE(CRSCARD) INPUT SEQUENTIAL RECDR BUFFERED,
FILE(CRSDATA) OUTPUT SEQUENTIAL RECDR BUFFERED,
FILE(SYSPRINT) PAGESIZE(59) LINESIZE(132) PRINT OUTPUT;

PUT EDIT('ROOM DATA, CARD IMAGES')(X(35),A);
PUT SKIP(2);
FLAG=1;
DO I=1 TO 400;
READ FILE(SYSIN) SET(Q);
IF FLAG=0 THEN GO TO CONT;
GET STRING(ROOMCARD) EDIT(TYPE(I),ROOMNUM(I))(X(10),A(3),X(58),A(6)
);
PUT SKIP EDIT(ROOMCARD)(A(80));
END;
CCNT:
CN ENDPAGE(SYSPRINT) BEGIN;
PUT PAGE EDIT('NUMBER NAME CR TYPES R1 R2 TYPE SIZE WRH
# REC T LAB ROOM ROOM')(A);
PUT SKIP(3);

END;
PUT PAGE EDIT('COURSE CARD DATA,CARD BY CARD/ EDITED COURSE DATA')(SKIP
(2),X(30),A);
PUT SKIP(3) EDIT('NUMBER NAME CR TYPES R1 R2 TYPE SIZE WRH
# REC T LAB ROOM ROOM')(A);
PUT SKIP;
COURSE=C;
READ FILE(CRSCARD) SET(P);
RECORDS=RECORDS+1;
PUT SKIP EDIT(OUTPUTRECDR)(X(15),A(80));
GET STRING(OUTPUTRECDR) EDIT(LINECOUNT,NUMBER,TYPECODE,NAME,CRHR,(DAYS(
I) CC I=1 TO 6),FROMCHR,FROMMIN,TOCHR,TOMIN,ROOMUSED)(F(6),X(26),

```

```

P'995999',A(1),X(4),A(12),A(2),6 A(1),P'R9',3 F(2),A(6));
A: CCOURSE=NUMBER;
ALPHID=NAME;
CLDCOURSE=COURSE;
CR=CRFR;
SECTYPE(1)=TYPECCDE;
SECTYPES=1;
QUANTITY(1)=1;
SECSIZE(1)=LINECOUNT;
CALL TIME(WRH(1));
EVENING(1)=NIGHT;
CALL ROOMSEARCH(ROOMUSED,I);
IF I=999
  THEN GO TO READ;
IF TYPE(1)='I10'
  THEN REC(1)='R';
ELSE DC;
  LAB(1)='L';
  ROOMS(1,1)=ROOMUSED;
  COUNTROOMS(1)=COUNTROOMS(1)+1;
END;
READ: READ FILE(CRSCARD ) SET (P);
RECORDS =RECORDS+1;
GET STRING(INPUTRECORD) EDIT(LINECOUNT,NUMBER,TYPECODE,NAME,CRHR,(DAYS(
I) DC I=1 TO 6),FROMHR,FROMMIN,TOHR,TOMIN,ROOMUSED)(F(6),X(26),
P'995999',A(1),X(4),A(12),A(2),6 A(1),P'R9',3 F(2),A(6));
IF CLDCOURSE = NUMBER
  THEN GO TO NEWCOURSE;
PUT SKIP EDIT(INPUTRECORD)(X(15),A(8C));
CALL TIME(WR);
DO KK=1 TO SECTYPES;
  IF SECTYPE(KK)=TYPECODE
    THEN IF EVENING(KK)=NIGHT
      THEN DO;
        K=KK;
        GO TO NEXTSTEP;
      END;
    ELSE;

```

```

ELSE;
END;
SECTYPES=SECTYPES+1;
SECTYPE(SECTYPES)=TYPECODE;
K=SECTYPES;
EVENING(K)=NIGHT;
NEXTSTEP: WRH(K)=WRH(K)+WR;
IF CRPR = ' ' ;
THEN CR=CRHR;
QUANTITY(K)=QUANTITY(K)+1;
SECSIZE(K)=SECSIZE(K)+LINECOUNT;
CALL RCCMSEARCH(ROOMUSED,I);
IF I=9999
THEN GO TO READ;
IF TYPE(I)='110'
THEN REC(K)='R';
ELSE CC;
LAB(K)='L';
DO J=1 TO COUNTROOMS(K);
IF ROOMS(K,J)=ROOMUSED
THEN GO TO READ;
ELSE;
END;
COUNTROOMS(K)=COUNTROOMS(K)+1;
ROOMS(K,COUNTROOMS(K))=ROOMUSED;
END;
GO TO READ;
NEWACOURSE: DO I=1 TO SECTYPES;
SECSIZE(I)=SECSIZE(I)/QUANTITY(I);
WRH(I)=WRH(I)/( QUANTITY(I));
IF REC(I) = ' ' & LAB(I) = ' ' THEN CC;
LAB(I)=' ' ;
COUNTROOMS(I)=0;
END;
END;
PUT SKIP EDIT(COURSEID,(COUNTROOMS(I) DO I=1 TO SECTYPES),(SECTA(J),(
RCCMS(J,L) DO L=1 TO COUNTROOMS(J)), ' ' DO J=1 TO SECTYPES))(A(7),A(12
),A(2),

```



```

F(1),(SECTYPES) F(3),(SECTYPES) (SKIP,A(39),A(2),F(5),F(8,4),F(3),X(1),
3 F(2),(COUNTRCMS(J)) A(7),A(1));
PUT STRING(CUTOATA) EDIT(COURSEID,(COUNTRCMS(I) DO I=1 TO SECTYPES),(S
ECLATAFJ),(ROOMS(J,L) DO L=1 TO COUNTRCMS(J)),* ; DO J=1 TO SECTYPES)
)(A(6),
A(12),A(2),F(2),(SECTYPES) F(2),(SECTYPES) (A(1),F(3),F(8,4),F(2),
3 A(1),(COUNTRCMS(J)) A(6),A(1));
WRITE FILE(CRSDATA) FROM (OUTDATA);
PUT SKIP EDIT(INPUTRECORD)(X(15),A(60));
DO I=1 TO 20;
COUNTROOMS(I)=0;
END;
DO I=1 TO 15;
WKH(I)=0;
SECTYPE(I)=' ';
EVENING(I)=' ';
SECSIZE(I)=0;
QUANTITY(I)=0;
REC(I)=' ';
LAB(I)=' ';
END;
COURSES=COURSES+1;
CC TO A;
LOOPSEARCH: PROCEDURE(RCOM,INDEX);
DECLARE ROOM CHARACTER(6);
IF RCOM=
THEN DO;
INDEX=9999;
GO TO END;
END;
INDEX=1;
IF HIGH=IRMAX;
IF ROOM=RCOMQUA(HIGH) THEN DO;
INDEX=INDEX;
GO TO END;
END;
IF ROOM=RCOMQUA(LOW) THEN DO;
INDEX=INDEX;

```

```

GO TO END;
END;

I=I*MAX/2;
ONE: IF ROOM= ROOMNUM(I)
  THEN DO;
    INDEX=I;
    GO TO END;
  END;
  IF (I*HIGH-I*LOW)=1
  THEN DO;
    I=9999;
    PUT SKIP EDIT('*****ROOM ',ROOM,' NOT ON ROOM LIST')(A,A(6
    ),A);
    GO TO END;
  END;
  IF ROOM > ROOMNUM(I)
  THEN DO;
    ILOW=I;
    I=I*HIGH-( I*HIGH-I*LOW)/2;
    GO TO ONE;
  END;
  ELSE DO;
    I*HIGH=I;
    I=ILOW+( I*HIGH-ILOW)/2;
    GO TO ONE;
  END;
END;

END;
END ROOMSEARCH;
TIME: PROCEDURE(R1);
DECLARE RT FIXED BINARY(31,7);
IF DAYS(1)='A'
  THEN DO;
    SIGHT=' ' ;
    RT=C;
    GO TO END;
  END;
  GO=C;
  I=1 TO 6;
  IF DAYS(I) = ' '

```

```

      IF A JJ=JJ+1;
END;
IF C > FROMHR
  THEN LU;
      NIGHT='E';
      FROMHR= -FROMHR;
END;
      ELSE NIGHT='D';
      IF TCHR=C
        THEN LU;
          RT=JJ;
          GO TO END;
        END;
      FROMR=FLOOR(FROMHR*2+DIVIDE(FROMMIN,30,31,7));
      TCHR=CEIL(TCHR*2+DIVIDE(TOMIN,30,31,7));
      IF FROMHR>TCHR
        THEN RT=DIVIDE((24-FROMHR+TOHR)*JJ,2,31,7);
        ELSE RT=DIVIDE((TOHR-FROMHR)*JJ,2,31,7);
      END;
      END TIME;
      STOP: END CRSEDT;
/*
//CC,SYSIA EE *
*****

      DATA ON CLASSROOMS AND CLASS LABORATORIES

*****

/*
//CRSEDATA CC CSN=COFH67,CRSDATA,SPACE=(TRK,(40,5)),VOL=SER=222222,
//      LISP=(CLP,KEEP),DDB=(RECFM=VB,LEFCL=2006,BLKSIZE=6022),UNIT=2314
//CC,CRSCARD UT CSN=COFH67,CRSCDSC,VOL=SER=222222,UNIT=2314,
//      LISP=(CLP,KEEP)
/*

```

## DISTWRH

PURPOSE--This program was written to use the sectioning rules and course data to divide course enrollments into sections of courses and to form a distribution of weekly room hours for the projected sections. (See section 3.3.2 for further details.)

METHOD--The program reads the course data for a course and then uses the ICLM data (see section 3.3.2) to determine an enrollment for it. The program tabulates distributions of classroom and class laboratory weekly room hours and prints these distributions after all courses have been processed. The program consists of two main parts: The sectioning of courses and the printing of the distributions.

### INPUT--

1. Minimum section sizes--(The use of these cards is explained in section 3.2.) These cards are read with the following PL/1 statement:

```
GET LIST(N);
```

```
GET SKIP EDIT(SECMIN(I))(X(5),F(5));
```

where N is the number of index numbers and SECMIN is a vector of values for the respective index numbers.

2. Department abbreviations--These cards are explained in this Appendix under "ICLMOP1".
3. ICLM--This is a direct access data set. Its contents are explained in Appendix B under "Back-up Tape--Data Set 2". The JCL requirements are given at the end of the program listing.
4. JCL specifying the card punch--The JCL used in the program is given at the end of the program listing.

### OUTPUT--

1. Echo check of departmental abbreviations.
2. Echo check of section minimums.
3. Description of how each course was sectioned. A sample of this output is given in Figure A.9.

```

201106 CRAL COMW 1A
TOTAL PROJECTED ENROLLMENT= 716
SECTIONED INFO:
*****
2 REC SECTIONS OF 285 TYPE 4
6 REC SECTIONS OF 23
18 REC SECTIONS OF 24 TYPE B
*****
2 REC SECTIONS OF 23
4 REC SECTIONS OF 24 TYPE 2
*****
201107 CRAL COMW 1B
TOTAL PROJECTED ENROLLMENT= 11
SECTIONED INFO:
*****
1 LAB SECTION OF 11 TYPE 2
1 LAB SECTION OF 11 TYPE D
*****
201108 CRAL COMW 1P
TOTAL PROJECTED ENROLLMENT= 17
SECTIONED INFO:
*****
*** EVENING CLASS, THEREFORE THIS SPACE DEMAND WAS NOT PROCESSED. TYPE 2
*****
201109 CRAL COM 1AP
TOTAL PROJECTED ENROLLMENT= 42
SECTIONED INFO:
*****
2 REC SECTIONS OF 21 TYPE 2
*****
201110 TAC SPRG VCI
TOTAL PROJECTED ENROLLMENT= 5
SECTIONED INFO:
*****
*** NO CLASSROOM OR CLASS LABORATORY SPACE GENERATED
DUE TO AC ROOM NUMBER BEING GIVEN IN THE COURSE DATA. TYPE 2
*****
201111 SP SPRG PART
TOTAL PROJECTED ENROLLMENT= 42
SECTIONED INFO:
*****
*** NO CLASSROOM OR CLASS LABORATORY SPACE GENERATED
DUE TO AC ROOM NUMBER BEING GIVEN IN THE COURSE DATA. TYPE 3
*****
201115 INTR THEATRE
TOTAL PROJECTED ENROLLMENT= 48
SECTIONED INFO:
*****
1 REC SECTION OF 48 TYPE 2
*****
201116 APC THEATRE
TOTAL PROJECTED ENROLLMENT= 49
SECTIONED INFO:
*****
2 REC SECTIONS OF 16
1 REC SECTIONS OF 17 TYPE 2
*****
201120 CRAL COMW 1I
TOTAL PROJECTED ENROLLMENT= 114
SECTIONED INFO:
*****
2 REC SECTIONS OF 57 TYPE 2
6 REC SECTIONS OF 19 TYPE A
*****
201121 APC THEATRE
TOTAL PROJECTED ENROLLMENT= 32
SECTIONED INFO:
*****
1 REC SECTION OF 32 TYPE 2
*****

```

Figure A.9--A sample page of DISTWRH showing how courses were sectioned.

4. Distribution of classroom weekly room hours. This is a listing and a deck of punched-cards. The cards are punched with the following PL/1 statement:

```
PUT FILE(PUNCH) EDIT((I,RECDIST(I) DO I= 1 TO 351))
                     (4(F(8),2F(6)));
```

where I is the section size and RECDIST is the distribution of classroom weekly room hours. The listing of the distribution used in the comparison of the actual and projected distributions is given in Figure A.10.

5. The distribution of class laboratory weekly room hours. This distribution is broken down by department. The distribution for each department is listed with the printed output and written onto direct access storage. The record written onto direct access storage was written directly from the following PL/1 data structure:

```
1 LABDISTOUT,
2  LNUMBER FIXED BINARY(15),
2  LDEPTNAME CHARACTER(5),
2  LABDIST(52),
3  LABWRH FIXED BINARY(15),
3  LABWSH FIXED BINARY(15),
```

where

LNUMBER is the department number,

LDEPTNAME is the department abbreviation,

LABWRH is the distribution of class laboratory weekly room hours,

LABWSH is the distribution of class laboratory weekly student hours.

The JCL requirements for this data set are given at the end of the program listing.

LECTURE-PRECIPITATION WEEKLY ROOM HOURS AND WEEKLY STUDENT HOURS BY SECTION SIZE

SEC-SIZE	MRH	WSH	SEC-SIZE	MRH	WSH	SEC-SIZE	MRH	WSH	SEC-SIZE	MRH	WSH
1	1	1	55	31	1705	109	11	1199	163	4	652
2	66	132	56	7	392	110	3	330	164	0	0
3	61	182	57	15	855	111	1	111	165	0	0
4	49	198	58	9	522	112	3	336	166	3	498
5	38	136	59	19	1121	113	0	0	167	1	167
6	79	474	60	21	1260	114	8	912	168	0	0
7	68	477	61	3	183	115	7	805	169	0	0
8	49	392	62	6	372	116	2	232	170	6	1020
9	73	687	63	13	815	117	0	0	171	9	1539
10	71	710	64	0	0	118	7	826	172	3	516
11	55	265	65	0	0	119	5	595	173	0	0
12	60	729	66	24	1584	120	0	0	174	3	522
13	106	1282	67	7	469	121	3	363	175	0	0
14	62	868	68	0	0	122	0	0	176	12	2112
15	109	1635	69	5	345	123	0	0	177	2	354
16	163	2618	70	5	350	124	0	0	178	1	178
17	192	3287	71	0	0	125	0	0	179	2	358
18	101	1825	72	1	72	126	0	0	180	0	0
19	123	2242	73	7	511	127	2	254	181	0	0
20	89	1793	74	9	666	128	0	0	182	0	0
21	167	2507	75	0	0	129	0	0	183	6	1098
22	134	2664	76	3	228	130	2	260	184	0	0
23	118	2729	77	6	462	131	3	393	185	0	0
24	486	11664	78	2	156	132	0	0	186	0	0
25	52	1300	79	9	711	133	0	0	187	0	0
26	85	2210	80	18	1440	134	3	402	188	0	0
27	59	2693	81	10	885	135	0	0	189	2	378
28	143	4021	82	0	0	136	21	2856	190	0	0
29	95	2781	83	0	0	137	0	0	191	0	0
30	84	2250	84	0	0	138	0	0	192	3	576
31	59	1829	85	4	340	139	0	0	193	0	0
32	19	609	86	0	0	140	0	0	194	0	0
33	23	1027	87	6	578	141	0	0	195	0	0
34	32	1068	88	6	528	142	0	0	196	3	588
35	9	315	89	12	1068	143	0	0	197	3	591
36	17	612	90	6	540	144	0	0	198	7	1386
37	15	555	91	6	546	145	0	0	199	3	597
38	55	2100	92	0	0	146	0	0	200	0	0
39	23	1218	93	4	372	147	1	147	201	3	603
40	20	800	94	0	0	148	0	0	202	9	1818
41	59	2419	95	1	95	149	0	0	203	3	609
42	82	2444	96	9	864	150	0	0	204	0	0
43	35	1834	97	5	485	151	3	453	205	5	1025
44	14	614	98	0	0	152	0	0	206	0	0
45	15	675	99	4	396	153	3	459	207	0	0
46	8	362	100	6	600	154	0	0	208	0	0
47	18	846	101	3	303	155	0	0	209	0	0
48	26	1243	102	6	612	156	0	0	210	0	0
49	23	1217	103	2	206	157	1	157	211	0	0
50	13	650	104	0	0	158	0	0	212	0	0
51	15	669	105	8	840	159	0	0	213	0	0
52	15	710	106	23	2438	160	0	0	214	0	0
53	6	213	107	0	0	161	0	0	215	0	0
54	12	547	108	6	648	162	3	466	216	0	0

Figure A.10--The distribution of classroom weekly room hours produced by DISTWRH and used in section 3.3 for comparison with the actual distribution.

LECTURE-RECITATION WEEKLY ROOM HOURS AND WEEKLY STUDENT HOURS BY SECTION SIZE  
(CONTINUED)

SFC-SIZE	WRH	LSH	SEC-SIZE	WRH	LSH	SEC-SIZE	WRH	WSH	SEC-SIZE	WRH	WSH
217	0	0	271	5	2439	325	0	0			
218	3	654	272	5	2448	326	0	0			
219	0	0	273	0	0	327	0	0			
220	0	0	274	0	0	328	0	0			
221	0	0	275	0	0	329	0	0			
222	0	0	276	0	0	330	0	0			
223	0	0	277	0	0	331	0	0			
224	0	0	278	0	0	332	0	0			
225	0	0	279	0	0	333	0	0			
226	0	0	280	0	0	334	0	0			
227	0	0	281	2	562	335	0	0			
228	0	0	282	0	0	336	0	0			
229	0	0	283	0	0	337	0	0			
230	0	0	284	0	0	338	0	0			
231	0	0	285	2	570	339	0	0			
232	0	0	286	0	0	340	0	0			
233	0	0	287	0	0	341	0	0			
234	0	0	288	0	0	342	0	0			
235	0	0	289	0	0	343	0	0			
236	0	0	290	0	0	344	0	0			
237	0	0	291	0	0	345	0	0			
238	3	714	292	0	0	346	0	0			
239	0	0	293	0	0	347	0	0			
240	2	480	294	0	0	348	0	0			
241	0	0	295	0	0	349	0	0			
242	0	0	296	0	0	350	0	0			
243	0	0	297	0	0	OVER 350	3	1122			
244	3	732	298	0	0						
245	0	0	299	0	0						
246	0	0	300	0	0						
247	0	0	301	0	0						
248	0	0	302	0	0						
249	2	482	303	0	0						
250	0	0	304	0	0						
251	0	0	305	0	0						
252	0	0	306	0	0						
253	0	0	307	0	0						
254	0	0	308	0	0						
255	0	0	309	0	0						
256	3	769	310	0	0						
257	0	0	311	0	0						
258	0	0	312	0	0						
259	0	0	313	0	0						
260	6	1560	314	0	0						
261	0	0	315	0	0						
262	0	0	316	0	0						
263	0	0	317	1	317						
264	0	0	318	0	0						
265	0	0	319	0	0						
266	0	0	320	0	0						
267	0	0	321	0	0						
268	0	0	322	0	0						
269	0	0	323	0	0						
270	0	0	324	0	0						

Figure A.10---Continued



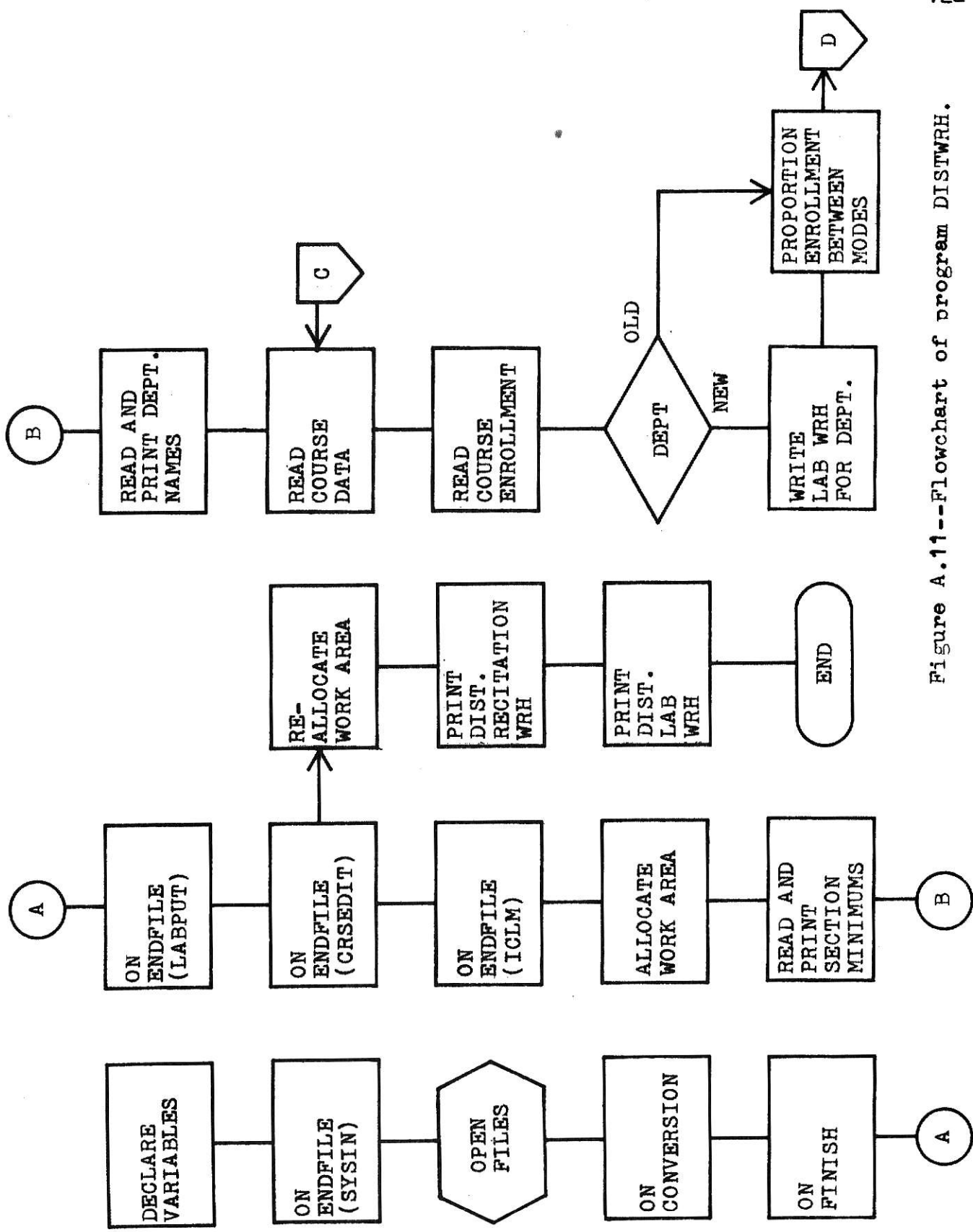


Figure A.11--Flowchart of program DISTWRH.

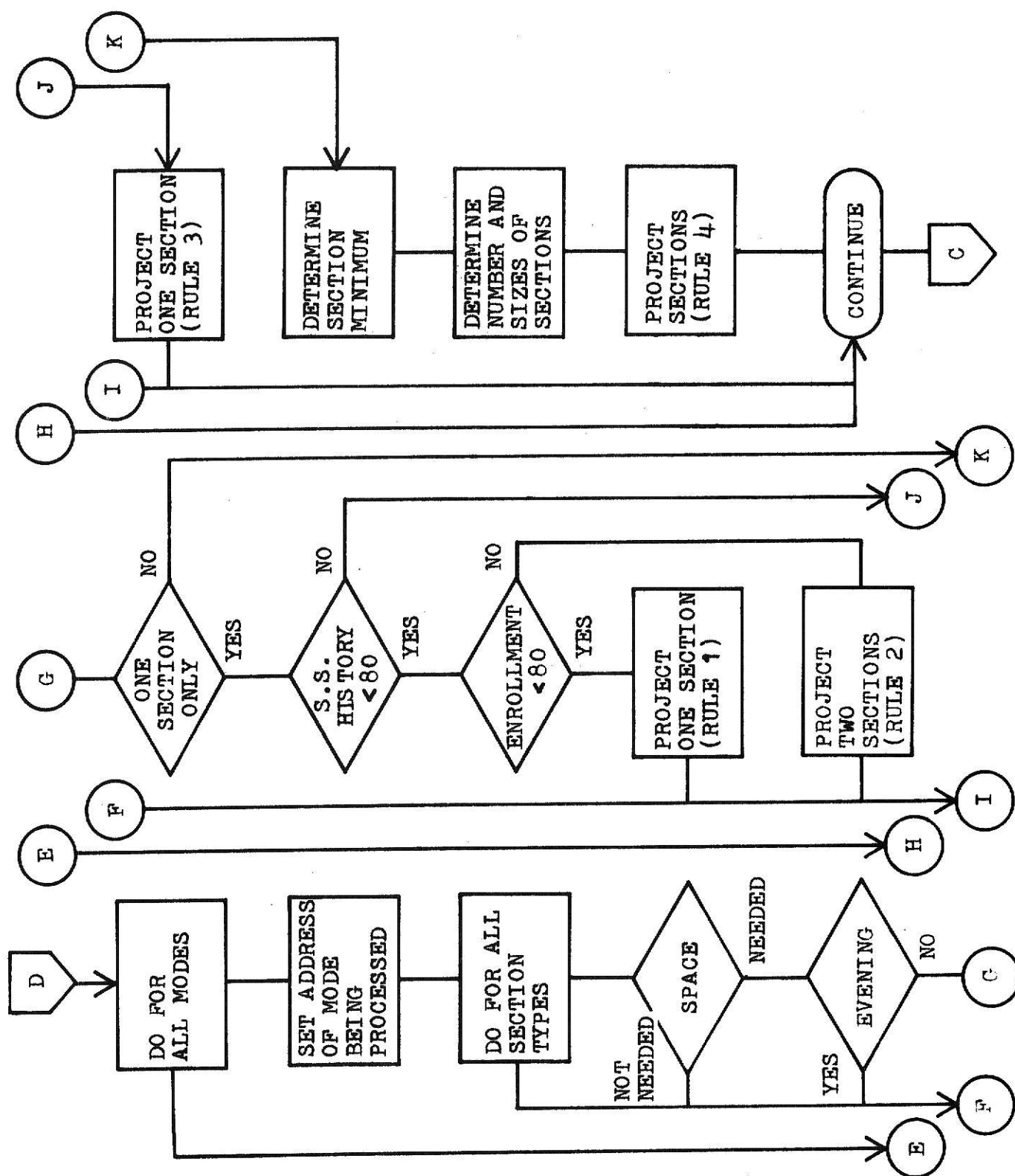


Figure A.11--Flowchart of program DISTWRH (Continued).

DISTWRH

SOURCE LISTING

```

// JOB (FH6706C2,15,10,200),COMPTON,CLASS=B
//STEP3 EXEC PL1LFCLG
//PL1L.SYSIN DD *
CISTWRH: PROCEDURE OPTIONS(MAIN);
DECLARE
  1 DATA BASED(P),
    2 COURSE CHARACTER(6),
    2 CUR CHARACTER(3),
    2 LEV CHARACTER(1),
    2 CURALPH CHARACTER(3),
    2 FRAC CHARACTER(5),
    2 COUNTER FIXED BINARY(15),
  1 COURSDATA CONTROLLED,
    2 COURSID,
    3 COURSE CHARACTER(6),
    3 INDICATOR FIXED BINARY(15),
    3 ALPHID CHARACTER(12),
    3 CR CHARACTER(2),
    2 KERNEL(3),
    3 KSECTYPES FIXED BINARY(15),
    3 KCOUNTERCOMS(20) FIXED BINARY(15),
    3 KSECTYPES(10),
    4 KSECDATA,
    5 KSECTYPE CHARACTER(1),
    5 KSECSIZE FIXED BINARY(31),
    5 KWRH FIXED BINARY(31,7),
    5 KQUANTITY FIXED BINARY(31),
    5 KREC CHARACTER(1),
    5 KEVENING CHARACTER(1),
    5 KLAB CHARACTER(1),
    4 KROOMS(20) CHARACTER(6),
  (KICTAL,KCCOUNT(3),CCURESUM) FIXED BINARY(31),
    SUM FIXED BINARY(15),
  CHECKCOURSE CHARACTER(6),
  1 TEMPORSDATA BASED(G),
    1 SECTYPES FIXED BINARY(15),
    2 COUNTERCOMS(20) FIXED BINARY(15),
    SECTYPES(10),

```

```

3 SECDATA,
  4 SECTYPE CHARACTER(1),
  4 SECSIZE FIXED BINARY(31),
  4 WRH FIXED BINARY(31,7),
  4 QUANTITY FIXED BINARY(31),
  4 REC CHARACTER(1),
  4 EVENING CHARACTER(1),
  4 LAB CHARACTER(1),
  3 ROOMS(20) CHARACTER(6),
  (III, JJJ) INITIAL(0),
  (DEPT, CLDDEPT) CHARACTER(3) INITIAL('002'),
  OLD CHARACTER(6),
  (SECTIONS, SECSKIPS) FIXED BINARY(15) INITIAL(0),
  CCURSUM FIXED BINARY(15),
  (NAMEA, CLDNAME) CHARACTER(5) INITIAL('*****'),
  1 SECMINMAX(30),
  2 MAXSS FIXED BINARY(15),
  2 MINSS FIXED BINARY(15),
  1 DEPTINDEX(0:750) CONTROLLED,
  3 NAME CHARACTER(5) INITIAL((751)(1)*****),
  2 INDEXER FIXED BINARY(15) INITIAL((751) 999),
  1 REQUIST(351),
  2 RECWRH FIXED BINARY(15),
  2 RECWSH FIXED BINARY(15),
  1 LABDISTCUT,
  2 LABUMBER FIXED BINARY(15),
  3 LDEPTNAME CHARACTER(5),
  2 LABDIST(52),
  3 LABWRH FIXED BINARY(15),
  3 LABWSH FIXED BINARY(15),
  (WKSTHR, R, SS, MAXNUMSEC) FIXED BINARY(15),
  1 PAVEREC CONTROLLED,
  2 TITLER(6) CHARACTER(120) INITIAL((6)(120)' '),
  3 BODYR(54,4) CHARACTER(30) INITIAL((4*54)(30)' '),
  1 CHARACTER(20),
  42 CHARACTER(12),
  1 FACCLAB CONTROLLED,
  2 SUFF,

```

```

3 TITL(4) CHARACTER(120) INITIAL((4)(120)' '),
3 TITL(10) CHARACTER(12) INITIAL((10)(12)' '),
3 TITL(2) CHARACTER(120) INITIAL((2)(120)' '),
2 SECID(52) CHARACTER(12) INITIAL((52)(12)' '),
2 BOCYL(52,9) CHARACTER(12) INITIAL((52*9)(12)' '),
  (FLAG1,FLAG2) INITIAL(0);
CN ENDFILE(SYSIN) GO TO CONTINUEA;
OPEN FILE(ICLM) RECORD INPUT,
  FILE(CRSECT) STREAM INPUT,
  FILE(LABPUT) OUTPUT RECORD;
ON CNVERSION PUT SKIP EDIT('**CONVERSION ERROR',ONSOURCE,ONCHAR)(A,A,
A);
CN FINISH BEGIN;
  PUT PAGE;
  PUT SKIP EDIT('RECORDS READ ON ICLM      ',III)(A,F(6));
  PUT SKIP(2) EDIT('RECORDS READ ON CRSECT =',JJJ)(A,F(6));
  PUT SKIP(2) EDIT('TOTAL NUMBER OF',SECTIONS PROCESSED',I=',
  SECTIONS)(A,SKIP,X(2),A,COLUMN(25),A(1),F(6));
  PUT SKIP(2) EDIT('NUMBER OF SECTIONS NOT',PROCESSED DO TO ',
  'EVENING CLASSES OR NC ',CLASSROOM SPACE NEEDED',I=',SECSKIPS)
  (A,SKIP,X(2),A,SKIP,X(2),A,SKIP,X(2),A,COLUMN(25),A,F(6));
  END;
CN ENDFILE(LABPUT) BEGIN;
  PUT PAGE EDIT(STUFF,(SECID(L),(BOCYL(L,LL) DO LL=1 TO 9) DO L=1 TO
  52)) (4 A(120),10 A(12),2 A(120),52C A(12));
  GO TO STOP;
END;
CN ENDFILE(ICLM) BEGIN;
  FLAG1=FLAG1+1;
  IF FLAG1=1 THEN GO TO THERE;
  IF FLAG2=1 THEN GO TO END;
  WRITE FILE(LABPUT) FROM (LABDISTOUT);
A:
  CALL READCRSPATA;
  JUJ=JUJ+1;
  PUT SKIP EDIT('**COURSE DATA FOR COURSE',COURSE,'NOT NEEDED')(A,
  X(1),A(7),A);
  GO TO A;

```

```

THERE: END;
CA ENDFILE(CRSECT) BEGIN;
  FLAG2=1;
  IF FLAG1=1 THEN GO TO END;
  WRITE FILE(LABPUT) FROM (LABDISTOUT);
  B: READ FILE(ICLM) SET(P);
  III=III+1;
  PUT SKIP EDIT('COURSE DATA FOR COURSE',COURS,'NOT GIVEN')(A,X(1)
,A(7),A);
  GO TO B;
END;
ALLOCATE COURSDATA,DEPTINDEX;
GET LIST(N);
DO I=1 TO N;
  GET SKIP EDIT(SECMINMAX(I))(2 F(5));
END;
GET SKIP;
DO J=1 TO 751;
  GET EDIT(I,NAME(I))(F(13),X(2),A(5));
  PUT EDIT(I,NAME(I))(F(13),X(2),A(5));
END;
CONTINUEA:
PUT PAGE EDIT(' SECTION MINIMUMS AND MAXIMUMS') (X(35),A);
PUT SKIP(3);
DO I=1 TO N;
  PUT SKIP EDIT(I,SECMINMAX(I))(X(40),F(5),X(2),2 F(5));
END;
PUT PAGE;
J=0;
DO I=0 TO 750;
  NAMEA=NAME(I);
  IF NAMEA = '*****'
  THEN IF NAMEA = OLDNAME
  THEN DO;
    J=J+1;
    INDEXEN(I)=J;
    CLONAME=NAMEA;
  END;

```

```

ELSE INDEXER(I)=J;
ELSE;
END;
NUMDEPT=J;
NUMSMM=N;
READ FILE(ICLM) SET(P);
III=III+1;
READ;
CALL READCRSDATA;
JJJ=JJJ+1;
AAA: IF CCURS = COURSE
THEN IF CCURS > COURSE
THEN DO;
PUT SKIP EDIT('***COURSE DATA FOR COURSE ',COURSE,' NOT NEE
DED')(A,A(6),A);
CALL READCRSDATA;
JJJ=JJJ+1;
GO TO AAA;
END;
ELSE DO;
PUT SKIP EDIT('**COURSE DATA FOR COURSE ',COURS,' NOT GIVEN'
)(A,A(6),A);
OLD=COURS;
B: READ FILE(ICLM) SET(P);
III=III+1;
IF CCURS = OLD THEN GO TO B;
GO TO AAA;
END;
ELSE DO;
COURSESUM=CCOUNTER;
OLD=COURS;
C: READ FILE(ICLM) SET(P);
III=III+1;
IF CCURS = OLD THEN DO;
COURSESUM=COURSESUM+CCOUNTER;
GO TO C;
END;
END;

```



```

SET STRING(COURSE) EDIT(DEPT)(A(3));
IF INDEXER(DEPT) /= INDEXER(OLDDEPT)
  THEN IF OLDDEPT /= '002'
    THEN DO;
      WRITE FILE(LABPUT) FROM(LABDISTOUT);
      OLDDEPT=DEPT;
      LDEPTNAME=NAME(DEPT);
      LNUMBER=INDEXER(DEPT);
      LABDIST=C;
    END;
  ELSE DO;
    OLDDEPT=DEPT;
    LDEPTNAME=NAME(DEPT);
    LNUMBER=INDEXER(DEPT);
    RECDIST=0;
    LABDIST=C;
  END;
ELSE;
KTCTAL=C;
DO I=1 TO INDICATOR;
  KCCOUNT(I)=0;
  DO J=1 TO KSECTYPES(I);
    KCOUNT(I)=KCOUNT(I)+KSECSIZE(I,J)*KQUANTITY(I,J);
  END;
  KCCOUNT(I)=DIVIDE(KCOUNT(I),KSECTYPES(I),31,0);
  KTCTAL=KTCTAL+ KCOUNT(I);
END;
PUT SKIP(2) EDIT(COURSE,ALPHID)(A(6),X(3),A(12));
PUT SKIP EDIT('KTCTAL PKJECTED ENROLLMENT=',COURSESUM)(X(4),A,F(5));
PUT SKIP EDIT('SECTIONED INTO:')(X(4),A);
DO I3=1 TO INDICATOR;
  G=ADDR(KERNEL(I3));
  COURSUM=COURSESUM#DIVIDE(KCOUNT(I3),KTCTAL,31,8);
  IF INDICATOR /= 1 THEN PUT SKIP EDIT('*****') *****;
  *****
  PUT EDIT('13,COURSE',STUDENTS')(X(4),A,F(3),F(6),X(2),A);
  DO I=1 TO SECTYPES;
    SECTION=SECTION+1;
    IF (REC(I)=' ' LAB(I)=' ')

```

```

THEN DO;
  SECSKIPS= SECSKIPS+1;
  PUT SKIP EDIT( '*** NO CLASSROOM OR CLASS LABORATORY SPACE GENERATED',
  'DUE TO NO ROOM NUMBER BEING GIVEN IN THE COURSE DATA.', ' TYPE ',
  SECTYPE(I))(X(4),A,SKIP,X(8),A,A,A(1));
  GO TO NEXT;
END;

IF EVENING(I)='E'
THEN DO;
  SECSKIPS=SECSKIPS+1;
  PUT SKIP EDIT('*** EVENING CLASS, THEREFORE THIS SPA
  CE DEMAND WAS NOT PROCESSED. TYPE ',SECTYPE(I))(X(4),A,A(1));
  GO TO NEXT;
END;

IF QUANTITY(I)=1
THEN IF SECSIZE(I)<80
THEN IF COURSUM<80
THEN IF REC(I)='R'
THEN DO;
  RECWRH(COURSE)=RECWRH(COURSE)+WRH(I);
  RECWSH(COURSE)=RECWSH(COURSE)+WRH(I)*COURSE;
  PUT SKIP EDIT(' 1 REC SECTION OF ',COURSE)(X(6),A,F(5));
  PUT EDIT('TYPE ',SECTYPE(I))(X(3),A,A(1));
  GO TO NEXT;
END;
ELSE DO;
  N=DIVIDE(COURSE,2,15,0);
  LABWRH(N)=LABWRH(N)+WRH(I);
  LABWSH(N)=LABWSH(N)+WRH(I)*COURSE;
  PUT SKIP EDIT(' 1 LAB SECTION OF ',COURSE)(X(6),A,F(5));
  PUT EDIT('TYPE ',SECTYPE(I))(X(3),A,A(1));
  GO TO NEXT;
END;

ELSE DO;
  SUB=DIVIDE(COURSE,2,15,0);
  WRH(I)=WRH(I)*2;
  IF REC(I)='R'
  THEN DO;

```

```

IF SUM>350 THEN J=351;
ELSE J= SUM;
RECWRH(J)=RECWRH(J)+WRH(I);
RECWSH(J)=RECWSH(J)+WRH(I)* SUM;
PUT SKIP EDIT(' 2 REC SECTIONS OF ', SUM(X(6),A,F(5)));
PUT EDIT('TYPE ',SECTYPE(I))(X(3),A,A(1));
GO TO NEXT;
END;
ELSE DO;
N=DIVIDE( SUM,2,15,0);
IF N>51 THEN N=52;
LABWRH(N)=LABWRH(N)+WRH(I);
LABWSH(N)=LABWSH(N)+WRH(I)* SUM;
PUT SKIP EDIT(' 2 LAB SECTIONS OF ', SUM(X(6),A,F(5)));
PUT EDIT('TYPE ',SECTYPE(I))(X(3),A,A(1));
GO TO NEXT;
END;
END;
ELSE IF REC(I)='R'
THEN DO;
IF COURSUM>350 THEN J=351;
ELSE J=COURSUM;
RECWRH(J)=RECWRH(J)+WRH(I);
RECWSH(J)=RECWSH(J)+WRH(I)*COURSUM;
PUT SKIP EDIT(' 1 REC SECTION OF ',COURSUM)(X(6),A,F(5));
PUT EDIT('TYPE ',SECTYPE(I))(X(3),A,A(1));
GO TO NEXT;
END;
ELSE DO;
N=DIVIDE(COURSUM,1,15,0);
IF N>51 THEN N=52;
LABWRH(N)=LABWRH(N)+WRH(I);
LABWSH(N)=LABWSH(N)+WRH(I)*COURSUM;
PUT SKIP EDIT(' 1 LAB SECTION OF ',COURSUM)(X(6),A,F(5));
PUT EDIT('TYPE ',SECTYPE(I))(X(3),A,A(1));
GO TO NEXT;
END;
ELSE;

```

```

J=DIVIDE(SECSIZE(I),10,15,0);
IF J>NUMSM THEN J=NUMSM;
MAXSECS=SECSIZE(I);
IF JK1 THEN MINSECS=5;
ELSE MINSECS=SECSIZE(I)+MINSS(J);
MAXNUMSEC=DIVIDE(COURSEUM,MAXSECS,15,0);
IF(COURSEUM-MAXNUMSEC*MAXSECS)>MINSECS THEN MAXNUMSEC=MAXNUMSEC+1;
SS=DIVIDE(COURSEUM,MAXNUMSEC,15,0);
R=CCOURSEUM-(MAXNUMSEC*SS);
IF REC(I)='R'
THEN DO;
NR=MAXNUMSEC-R;
NSS=SS+1;
IF NR>350 THEN NR=351;
IF R>350 THEN R=351;
RECWRH(SS)=RECWRH(SS)+NR*WRH(I);
RECWRH(NSS)=RECWRH(NSS)+R*WRH(I);
RECWSH(SS)=RECWSH(SS)+NR*WRH(I)*SS;
RECWSH(NSS)=RECWSH(NSS)+R*WRH(I)*NSS;
IF NR /= 0 THEN
PUT SKIP EDIT(NR,' REC SECTIONS OF ',SS)(X(4),F(5),A,F(5));
IF R /= 0 THEN
PUT SKIP EDIT(R,' REC SECTIONS OF ',NSS)(X(4),F(5),A,F(5));
PUT EDIT('TYPE ',SECTYPE(I))(X(3),A,A(1));
GO TO NEXT;
END;
ELSE DO;
NR=MAXNUMSEC-R;
NSS=DIVIDE(SS+1,2,15,0);
ISS=DIVIDE(SS,2,15,0);
IF NSS>51 THEN NSS=52;
IF SS>51 THEN SS=52;
LABWRH(ISS)=LABWRH(NSS)+R*WRH(I);
LABWRH(ISS)=LABWRH(ISS)+R*WRH(I);
LABWSH(ISS)=LABWSH(NSS)+R*WRH(I)*(SS+1);
LABWSH(ISS)=LABWSH(ISS)+R*WRH(I)*SS;
IF NR /= 0 THEN
PUT SKIP EDIT(NR,' LAB SECTIONS OF ',SS)(X(4),F(5),A,F(5));

```

```

IF R = 0 THEN
  PUT SKIP EDIT(R, ' LAB SECTIONS OF ', (SS+1))(X(4), F(5), A, F(5));
  PUT EDIT('TYPE ', SECTYPE(I))(X(3), A, A(1));
  GO TO NEXT;
END;
NEXT: END;
GO TO READ;
END;
G=NULL;
FREE COURSDATA, DEPTINDEX;
ALLOCATE PAGEREC, PAGELAB;
CLOSE FILE(LABPUT);
OPEN FILE(LABPUT) RECORD INPUT;
PUT STRING(TITLER(2)) EDIT('LECTURE-RECITATION WEEKLY ROOM HOURS AND WE
EY STUDENT HOURS BY SECTION SIZE')(X(20), A);
A=SEC.SIZE WRH WSH ;
PUT STRING(TITLER(5)) EDIT(A, A, A, A)(X(5), A(20), 3 (X(10), A(20)));
K=0;
DO J=1 TO 4;
  DO I=1 TO 54;
    K=K+1;
    PUT STRING(BODYR(I, J)) EDIT(K, RECWRH(K), RECWSH(K))(X(7), F(5), 2 F(6));
  END;
END;
PUT PAGE EDIT(PAGEREC)(6 A(120), 54 (4 A(30)));
BODYR= ;
PUT STRING(TITLER(3)) EDIT(' (CONTINUED)')(X(48), A);
DO J=1 TO 1;
  DO I=1 TO 54;
    K=K+1;
    PUT STRING(BODYR(I, J)) EDIT(K, RECWRH(K), RECWSH(K))(X(7), F(5), 2 F(6));
  END;
END;
DO I=1 TO 16;
  K=K+1;
  PUT STRING(BODYR(I, 5)) EDIT(K, RECWRH(K), RECWSH(K))(X(7), F(5), 2 F(6));
END;
END;
PUT STRING(BODYR(27, 3)) EDIT('OVER 350', RECWRH(351), RECWSH(351))

```

```

(X(4),A(8),2 F(6));
PUT EDIT(PAGEREC)(6 A(120),54 (4 A(30)));
PUT FILE(PUNCH) EDIT(I,RECDIST(I) DC I=1 TO 351))(4(F(8),2 F(6)));
PUT STRING(TITLEL(2)) EDIT('LABORATORY WEEKLY ROOM HOURS AND WEEKLY STUDENT HOURS BY SECTION SIZE AND DEPARTMENT')(X(20),A );
AA=' WRH WSH';
PUT STRING(TITLL(1)) EDIT('SEC. SIZE', (AA DC I=1 TO 9))(X(1),A(9),X(2),
9 A(12));
DO I=1 TO 51;
PUT STRING(SECID(I)) EDIT(I*2,'-',I*2+1)(X(3),F(3),A(1),F(3));
END;
PUT STRING(SECID(52)) EDIT('OVER 103')(X(2),A(10));
DO I=1 TO 9;
TITL=' ';
BCDYL=' ';
DO J=1 TO 9;
READ FILE(LABPUT) INTO (LABDISTOUT);
PUT STRING(TITL(J+1)) EDIT(LDEPTNAME)(X(4),A(5));
DO K=1 TO 52;
PUT STRING(BODYL(K,J)) EDIT(LABWRH(K),LABWSH(K))(2 F(6));
END;
END;
PUT PAGE EDIT(STUFF,(SECID(L),(BCDYL(L,LL) DO LL=1 TO 9) DO L=1 TO 5
2))(4 A(120),10 A(12),2 A(120),520 A(12));
END;
STOP;
FREE PAGEREC,PAGELAB;
READCRSDATA: PROCEDURE;
GET FILE(CRSEDT) EDIT(COURSEID)(A(6),F(1),A(13),A(2));
IF INDICATOR ^= 0 THEN DO;
DO I=1 TO INDICATOR;
GET FILE(CRSEDT) SKIP EDIT(CHECKCOURSE,KSECTYPES(I),
(KCOUNTRUPS(I,J) DO J=1 TO KSECTYPES(I))(A(6),X(1),F(3),20 F(3));
IF CHECKCOURSE ^= COURSE THEN PUT SKIP EDIT ('DATA READ OUT OF SEQUEN
CE COURSE=',COURSE,' CHECKCOURSE=',CHECKCOURSE)(A,A(6),A,A(6));
DO J=1 TO KSECTYPES(I);
GET FILE(CRSEDT) SKIP EDIT(CHECKCOURSE,KSECDATA(I,J),
(KRCCDS(I,J,K) DO K=1 TO KCOUNTRUPS(I,J))(A(6),X(1),A(2),F(4),

```

```

F(8,J),X(2),F(3),3 A(2),20 A(7));
IF CHECKCCOURSE /= CCOURSE THEN PUT SKIP EDIT ('DATA READ OUT OF SEQUEN
CE CCOURSE=',CCOURSE,' CHECKCCOURSE=',CHECKCCOURSE)(A,A(6),A,A(6));
END;
END;
GET FILE(CRSEDIT) SKIP;
END;
ELSE DO;
  INDICATOR=1;
  GET FILE(CRSEDIT) EDIT(KSECTYPES(1),(KCOUNTROOMS(1,J) DO J=1 TO
  KSECTYPES(1))(F(3),20 F(3));
  DO J=1 TO KSECTYPES(1);
    GET FILE(CRSEDIT) SKIP EDIT(CHECKCCOURSE,KSECDATA(1,J),(KROOMS(1,J),
    K) DO K=1 TO KCOUNTROOMS(1,J))(A(6),X(1),A(2),F(4),F(8,3),X(2),F(3),
    3 A(2),20 A(7));
    IF CHECKCCOURSE /= CCOURSE THEN PUT SKIP EDIT ('DATA READ OUT OF SEQUEN
CE CCOURSE=',CCOURSE,' CHECKCCOURSE=',CHECKCCOURSE)(A,A(6),A,A(6));
END;
GET FILE(CRSEDIT) SKIP;
END;
END READCRSDATA;
END DISTWRH;
/*
//GC.SYSIN DD *

```

\*\*\*\*\*

# CARDS CONTAINING MINIMUM SECTION SIZES FOR DIFFERENT SIZES OF SECTIONS

\*\*\*\*\*

## CARDS CONTAINING DEPARTMENTAL ABBREVIATIONS

\*\*\*\*\*

```

/*
//JCL UNIT=2314,VOL=SER=222222,DSN=CCFH67,CURPROC,DISP=(OLD,KEEP)
//CRSECT DD DSN=CCFH67,CRSEDIT,UNIT=2314,VOL=SER=222222,
// LISP=(OLD,KEEP)

```

```
//LABELT DD DSN=COFH67,LABELT,UNIT=2314,VOL=SER=222222,  
// SPACE=(TRK,(3,1)),DCB=(LRECL=216,BLKSIZE=6912,RECFM=FB),  
// DISP=(OLD,KEEP)  
//PUNCH DD SYSOUT=8,DCB=(LRECL=80,RECFM=FB,BLKSIZE=1600)  
/*
```



CUMDIST

PURPOSE--This program was written to tabulate the actual distribution of classroom weekly room hours as given by the line schedule data. (See section 3.1.1.)

METHOD--This program processed each record of the line schedule data and determined the weekly room hours required for that section. Only classroom weekly room hours were counted in the distribution of weekly room hours formed by the program. After all records were processed it printed this distribution.

INPUT--

1. Room data--These cards are explained in Appendix B under "Room Data."
2. The line schedule data on direct access storage. A record of this data was an 80 character card image of a card of the line schedule data. The card form of the line schedule data is explained in Appendix B under "Line Schedule Data." The JCL requirements for this data set are given at the end of the program listing (GO.CRSCDSC).

OUTPUT--A printed distribution of the actual classroom weekly room hours. A sample of this output is given in Figure A.12. This sample output is the distribution used for comparison in section 3.3.

SECTION--FALL 1971  
DISTRIBUTION OF WEEKLY ROOM HOURS BY SECTION SIZE

SECTION	SIZE	WH	CUMWH	WSH	CUMWSH
1	14.00	14.00	14.00	14.00	14.00
2	27.00	41.00	54.00	54.00	68.00
3	36.00	79.00	114.00	182.00	182.00
4	48.00	127.00	192.00	374.00	374.00
5	36.00	165.00	182.00	546.00	546.00
6	65.00	231.00	376.00	56.00	56.00
7	95.00	326.00	665.00	1627.00	1627.00
8	65.00	392.50	522.00	2159.00	2159.00
9	52.50	445.00	472.50	2631.50	2631.50
10	69.00	514.00	690.00	3321.50	3321.50
11	71.00	585.00	781.00	4102.50	4102.50
12	69.50	653.50	822.00	4924.50	4924.50
13	100.00	753.50	1300.00	6224.50	6224.50
14	107.50	861.00	1505.00	7729.50	7729.50
15	137.00	998.00	2055.00	9784.50	9784.50
16	135.00	1133.00	2160.00	11944.50	11944.50
17	152.00	1285.00	2584.00	14528.50	14528.50
18	120.00	1405.00	2160.00	16688.50	16688.50
19	93.00	1498.00	1747.00	18455.50	18455.50
20	117.00	1615.00	2340.00	20795.50	20795.50
21	140.00	1755.00	2940.00	23735.50	23735.50
22	133.00	1888.00	2926.00	26661.50	26661.50
23	153.00	2041.00	3519.00	30180.50	30180.50
24	187.00	2228.00	4488.00	34668.50	34668.50
25	220.00	2448.00	5500.00	40168.50	40168.50
26	196.00	2644.00	5095.00	45284.50	45284.50
27	100.00	2744.00	2750.00	47954.50	47954.50
28	74.00	2818.00	2072.00	50036.50	50036.50
29	75.00	2893.00	2179.00	52211.50	52211.50
30	89.00	2981.00	2640.00	54851.50	54851.50
31	70.00	3051.00	2170.00	57021.50	57021.50
32	33.00	3084.00	1056.00	58077.50	58077.50
33	35.50	3119.50	1171.50	59249.00	59249.00
34	46.00	3165.50	1344.00	60813.00	60813.00
35	43.50	3209.00	1322.50	62335.50	62335.50
36	39.00	3248.00	1454.00	63739.50	63739.50
37	37.50	3285.50	647.50	64387.00	64387.00
38	29.00	3314.50	1122.00	65439.00	65439.00
39	32.00	3346.50	1150.00	66737.00	66737.00
40	53.00	3399.50	2120.00	68857.00	68857.00
41	45.00	3444.50	1365.00	70702.00	70702.00
42	21.00	3465.50	1302.00	72004.00	72004.00
43	23.00	3488.50	988.00	72993.00	72993.00
44	14.00	3502.50	616.00	73609.00	73609.00
45	24.00	3516.50	1080.00	74689.00	74689.00
46	39.00	3555.50	1794.00	76483.00	76483.00
47	25.00	3580.50	1173.00	77656.00	77656.00
48	20.00	3600.50	960.00	78616.00	78616.00
49	13.00	3613.50	637.00	79253.00	79253.00
50	16.00	3629.50	500.00	80053.00	80053.00
51	24.00	3653.50	1224.00	81277.00	81277.00
52	7.00	3660.50	364.00	81641.00	81641.00

Figure A.12--The actual distribution of classroom weekly room hours as produced by CUMDIST.

ACTUAL--FALL 1972  
DISTRIBUTION OF WEEKLY ROOM HOURS BY SECTION SIZE (CONTINUED)

SECTION SIZE	WSH	CUMWRH	WSH	CUMWSH
53	21.00	3661.50	1113.00	82756.00
54	26.00	3707.50	1404.00	84160.00
55	12.00	3719.50	660.00	84820.00
56	18.00	3737.50	1008.00	85828.00
57	7.00	3744.50	359.00	86277.00
58	17.00	3761.50	926.00	87213.00
59	3.00	3769.50	472.00	87685.00
60	19.00	3789.50	1140.00	88825.00
61	13.00	3801.50	793.00	89618.00
62	11.00	3812.50	682.00	90300.00
63	11.00	3823.50	693.00	90993.00
64	3.00	3826.50	492.00	91485.00
65	11.00	3837.50	715.00	92200.00
66	1.00	3838.50	66.00	92266.00
67	6.00	3844.50	402.00	92668.00
68	2.00	3846.50	136.00	92804.00
69	6.00	3852.50	414.00	93218.00
70	14.00	3866.50	980.00	94198.00
71	6.00	3872.50	426.00	94624.00
72	0.00	3872.50	0.00	94624.00
73	3.00	3875.50	219.00	94843.00
74	0.00	3875.50	0.00	94843.00
75	3.00	3878.50	225.00	95068.00
76	5.00	3883.50	340.00	95408.00
77	10.00	3893.50	770.00	96178.00
78	7.00	3900.50	546.00	96724.00
79	8.00	3908.50	711.00	97435.00
80	15.00	3924.50	1200.00	98635.00
81	2.00	3932.50	648.00	99283.00
82	4.00	3936.50	328.00	99611.00
83	3.00	3937.50	249.00	99860.00
84	5.00	3944.50	420.00	100280.00
85	10.00	3954.50	950.00	101230.00
86	3.00	3957.50	252.00	101482.00
87	11.00	3968.50	957.00	102439.00
88	6.00	3974.50	528.00	102967.00
89	5.00	3979.50	445.00	103412.00
90	6.00	3985.50	540.00	103952.00
91	6.00	3991.50	546.00	104498.00
92	7.00	3999.50	644.00	105142.00
93	9.00	4007.50	837.00	105979.00
94	7.00	4014.50	658.00	106637.00
95	8.00	4022.50	760.00	107397.00
96	12.00	4034.50	1152.00	108549.00
97	3.00	4037.50	291.00	108840.00
98	6.00	4043.50	588.00	109428.00
99	2.00	4045.50	198.00	109626.00
100	2.00	4047.50	200.00	109826.00
101	5.00	4055.50	808.00	110634.00
102	3.00	4058.50	306.00	110940.00
103	3.00	4061.50	309.00	111249.00
104	11.00	4072.50	1145.00	112394.00
105	8.00	4080.50	846.00	113240.00

Figure A.12--Continued

ACTUAL--FALL 1971  
DISTRIBUTION OF WEEKLY ROAD HOURS BY SECTION SIZE (CONTINUED)

SECTION SIZE	WH	CUMWH	WSH	CUMWSH
106	3.00	4083.50	316.00	113197.00
107	C.00	4083.50	0.00	113197.00
108	7.00	4090.50	756.00	113953.00
109	11.00	4101.50	1159.00	115122.00
110	5.00	4106.50	550.00	115762.00
111	3.00	4109.50	333.00	116035.00
112	2.00	4111.50	224.00	116259.00
113	3.00	4114.50	339.00	116598.00
114	2.00	4116.50	228.00	116826.00
115	3.00	4119.50	345.00	117171.00
116	3.00	4122.50	348.00	117519.00
117	5.00	4127.50	585.00	118104.00
118	3.00	4130.50	354.00	118458.00
119	C.00	4130.50	0.00	118458.00
120	6.00	4136.50	720.00	119178.00
121	C.00	4136.50	0.00	119178.00
122	C.00	4136.50	0.00	119178.00
123	2.00	4138.50	246.00	119424.00
124	C.00	4138.50	0.00	119424.00
125	C.00	4138.50	0.00	119424.00
126	C.00	4138.50	0.00	119424.00
127	C.00	4138.50	0.00	119424.00
128	11.00	4149.50	1408.00	120832.00
129	3.00	4152.50	387.00	121219.00
130	11.00	4163.50	1430.00	122649.00
131	C.00	4163.50	0.00	122649.00
132	C.00	4163.50	0.00	122649.00
133	C.00	4163.50	0.00	122649.00
134	C.00	4163.50	0.00	122649.00
135	2.00	4165.50	270.00	122919.00
136	C.00	4165.50	0.00	122919.00
137	C.00	4165.50	0.00	122919.00
138	1.00	4166.50	138.00	123057.00
139	3.00	4169.50	417.00	123474.00
140	C.00	4169.50	0.00	123474.00
141	3.00	4172.50	423.00	123897.00
142	C.00	4172.50	0.00	123897.00
143	C.00	4172.50	0.00	123897.00
144	C.00	4172.50	0.00	123897.00
145	C.00	4172.50	0.00	123897.00
146	C.00	4172.50	0.00	123897.00
147	C.00	4172.50	0.00	123897.00
148	C.00	4172.50	0.00	123897.00
149	C.00	4172.50	0.00	123897.00
150	C.00	4172.50	0.00	123897.00
151	2.00	4174.50	303.00	124199.00
152	C.00	4174.50	0.00	124199.00
153	6.00	4180.50	918.00	125117.00
154	3.00	4183.50	462.00	125579.00
155	C.00	4183.50	0.00	125579.00
156	C.00	4183.50	0.00	125579.00
157	C.00	4183.50	0.00	125579.00
158	C.00	4183.50	0.00	125579.00

Figure A.12--Continued

ACTUAL--CALL 1971  
DISTRIBUTION OF WEEKLY ROOM HOURS BY SECTION SIZE (CONTINUED)

SECTION SIZE	WRH	CUMWRH	WSH	CUMWSH
159	0.00	4183.50	0.00	125579.00
160	3.00	4186.50	480.00	126059.00
161	0.00	4186.50	0.00	126059.00
162	0.00	4186.50	0.00	126059.00
163	0.00	4186.50	0.00	126059.00
164	1.00	4187.50	164.00	126223.00
165	0.00	4187.50	0.00	126223.00
166	0.00	4187.50	0.00	126223.00
167	0.00	4187.50	0.00	126223.00
168	3.00	4190.50	504.00	126727.00
169	1.00	4191.50	169.00	126896.00
170	3.00	4194.50	510.00	127406.00
171	6.00	4200.50	1026.00	128432.00
172	0.00	4200.50	0.00	128432.00
173	0.00	4200.50	0.00	128432.00
174	0.00	4200.50	0.00	128432.00
175	0.00	4200.50	0.00	128432.00
176	3.00	4203.50	528.00	128960.00
177	0.00	4203.50	0.00	128960.00
178	1.00	4204.50	178.00	129138.00
179	7.00	4211.50	1253.00	130391.00
180	0.00	4211.50	0.00	130391.00
181	6.00	4217.50	1086.00	131477.00
182	6.00	4223.50	1092.00	132569.00
183	6.00	4229.50	1098.00	133667.00
184	12.00	4241.50	2208.00	135875.00
185	0.00	4241.50	0.00	135875.00
186	1.00	4242.50	186.00	136061.00
187	3.00	4245.50	561.00	136622.00
188	0.00	4245.50	0.00	136622.00
189	0.00	4245.50	0.00	136622.00
190	3.00	4248.50	570.00	137192.00
191	3.00	4251.50	573.00	137765.00
192	5.00	4256.50	960.00	138725.00
193	1.00	4257.50	193.00	138918.00
194	2.00	4259.50	388.00	139306.00
195	1.00	4260.50	195.00	139501.00
196	2.00	4262.50	392.00	139893.00
197	3.00	4265.50	591.00	140484.00
198	2.00	4267.50	396.00	140880.00
199	0.00	4267.50	0.00	140880.00
200	0.00	4267.50	0.00	140880.00
201	0.00	4267.50	0.00	140880.00
202	2.00	4269.50	404.00	141284.00
203	1.00	4270.50	203.00	141487.00
204	6.00	4276.50	1224.00	142711.00
205	1.00	4277.50	205.00	142916.00
206	2.00	4279.50	412.00	143328.00
207	0.00	4279.50	0.00	143328.00
208	0.00	4279.50	0.00	143328.00
209	0.00	4279.50	0.00	143328.00
210	0.00	4279.50	0.00	143328.00
211	0.00	4279.50	0.00	143328.00

Figure A.12--Continued

ACTUAL--FALL 1971  
DISTRIBUTION OF WEEKLY ROOM HOURS BY SECTION SIZE (CONTINUED)

SECTION SIZE	ASH	CUMKPH	WSH	CUMKSH
212	0.00	4279.50	0.00	143328.00
213	0.00	4279.50	0.00	143328.00
214	0.00	4279.50	0.00	143328.00
215	0.00	4279.50	0.00	143328.00
216	0.00	4279.50	0.00	143328.00
217	0.00	4279.50	0.00	143328.00
218	3.00	4282.50	65.00	143982.00
219	0.00	4282.50	0.00	143982.00
220	2.00	4284.50	44.00	144422.00
221	2.00	4286.50	44.00	144864.00
222	0.00	4286.50	0.00	144864.00
223	0.00	4286.50	0.00	144864.00
224	0.00	4286.50	0.00	144864.00
225	3.00	4289.50	675.00	145539.00
226	0.00	4289.50	0.00	145539.00
227	0.00	4289.50	0.00	145539.00
228	0.00	4289.50	0.00	145539.00
229	0.00	4289.50	0.00	145539.00
230	0.00	4289.50	0.00	145539.00
231	0.00	4289.50	0.00	145539.00
232	3.00	4292.50	696.00	146235.00
233	0.00	4292.50	0.00	146235.00
234	0.00	4292.50	0.00	146235.00
235	0.00	4292.50	0.00	146235.00
236	0.00	4292.50	0.00	146235.00
237	0.00	4292.50	0.00	146235.00
238	0.00	4292.50	0.00	146235.00
239	0.00	4292.50	0.00	146235.00
240	0.00	4292.50	0.00	146235.00
241	0.00	4292.50	0.00	146235.00
242	0.00	4292.50	0.00	146235.00
243	0.00	4292.50	0.00	146235.00
244	0.00	4292.50	0.00	146235.00
245	0.00	4292.50	0.00	146235.00
246	1.00	4293.50	246.00	146481.00
247	0.00	4293.50	0.00	146481.00
248	0.00	4293.50	0.00	146481.00
249	0.00	4293.50	0.00	146481.00
250	3.00	4296.50	750.00	147231.00
251	0.00	4296.50	0.00	147231.00
252	0.00	4296.50	0.00	147231.00
253	0.00	4296.50	0.00	147231.00
254	0.00	4296.50	0.00	147231.00
255	0.00	4296.50	0.00	147231.00
256	2.00	4298.50	512.00	147743.00
257	3.00	4301.50	771.00	148514.00
258	0.00	4301.50	0.00	148514.00
259	0.00	4301.50	0.00	148514.00
260	0.00	4301.50	0.00	148514.00
261	0.00	4301.50	0.00	148514.00
262	0.00	4301.50	0.00	148514.00
263	0.00	4301.50	0.00	148514.00
264	2.00	4303.50	525.00	149042.00

Figure A.12--Continued

ACTUAL--FALL 1971  
DISTRIBUTION OF WEEKLY ROAD HOURS BY SECTION SIZE (CONTINUED)

SECTION SIZE	WRH	CUMWRH	WSH	CUMWSH
265	C-CC	4303.50	0.00	149042.00
266	1-CC	4304.50	266.00	149308.00
267	C-CC	4304.50	0.00	149308.00
268	5-CC	4309.50	1340.00	150648.00
269	C-CC	4309.50	0.00	150648.00
270	C-CC	4309.50	0.00	150648.00
271	C-CC	4309.50	0.00	150648.00
272	C-CC	4309.50	0.00	150648.00
273	C-CC	4309.50	0.00	150648.00
274	C-CC	4309.50	0.00	150648.00
275	1-CC	4310.50	275.00	150923.00
276	C-CC	4310.50	0.00	150923.00
277	C-CC	4310.50	0.00	150923.00
278	C-CC	4310.50	0.00	150923.00
279	C-CC	4310.50	0.00	150923.00
280	2-CC	4312.50	560.00	151483.00
281	0-CC	4312.50	0.00	151483.00
282	C-CC	4312.50	0.00	151483.00
283	C-CC	4312.50	0.00	151483.00
284	1-CC	4313.50	284.00	151767.00
285	1-CC	4314.50	285.00	152052.00
286	C-CC	4314.50	0.00	152052.00
287	C-CC	4314.50	0.00	152052.00
288	C-CC	4314.50	0.00	152052.00
289	C-CC	4314.50	0.00	152052.00
290	C-CC	4314.50	0.00	152052.00
291	C-CC	4314.50	0.00	152052.00
292	C-CC	4314.50	0.00	152052.00
293	C-CC	4314.50	0.00	152052.00
294	C-CC	4314.50	0.00	152052.00
295	C-CC	4314.50	0.00	152052.00
296	2-CC	4316.50	592.00	152644.00
297	C-CC	4316.50	0.00	152644.00
298	C-CC	4316.50	0.00	152644.00
299	C-CC	4316.50	0.00	152644.00
300	C-CC	4316.50	0.00	152644.00
301	C-CC	4316.50	0.00	152644.00
302	C-CC	4316.50	0.00	152644.00
303	C-CC	4316.50	0.00	152644.00
304	C-CC	4316.50	0.00	152644.00
305	C-CC	4316.50	0.00	152644.00
306	C-CC	4316.50	0.00	152644.00
307	C-CC	4316.50	0.00	152644.00
308	C-CC	4316.50	0.00	152644.00
309	C-CC	4316.50	0.00	152644.00
310	C-CC	4316.50	0.00	152644.00
311	C-CC	4316.50	0.00	152644.00
312	C-CC	4316.50	0.00	152644.00
313	3-CC	4317.50	939.00	153583.00
314	1-CC	4320.50	314.00	153897.00
315	C-CC	4320.50	0.00	153897.00
316	C-CC	4320.50	0.00	153897.00
317	3-CC	4323.50	951.00	154848.00

Figure A.12--Continued



ACTUAL--FALL 1971  
DISTRIBUTION OF WEEKLY ROAD HOURS BY SECTION (CONTINUED)

SECTION	SIZE	WPH	CUMWPH	WSP	CUMWSP	
318	C.CC	4323.50	0.00	154848.00		
319	C.CC	4323.50	0.00	154848.00		
320	C.CC	4323.50	0.00	154848.00		
321	C.CC	4323.50	0.00	154848.00		
322	C.CC	4323.50	0.00	154848.00		
323	C.CC	4323.50	6+6.00	155494.00		
324	C.CC	4323.50	0.00	155494.00		
325	C.CC	4323.50	0.00	155494.00		
326	C.CC	4323.50	0.00	155494.00		
327	C.CC	4323.50	0.00	155494.00		
328	C.CC	4323.50	0.00	155494.00		
329	C.CC	4323.50	987.00	156481.00		
330	C.CC	4323.50	0.00	156481.00		
331	C.CC	4331.50	993.00	157474.00		
332	C.CC	4331.50	0.00	157474.00		
333	C.CC	4331.50	0.00	157474.00		
334	C.CC	4331.50	0.00	157474.00		
335	C.CC	4331.50	0.00	157474.00		
336	C.CC	4331.50	0.00	157474.00		
337	C.CC	4331.50	0.00	157474.00		
338	C.CC	4331.50	0.00	157474.00		
339	C.CC	4331.50	0.00	157474.00		
340	C.CC	4334.50	1020.00	158494.00		
341	C.CC	4334.50	0.00	158494.00		
342	C.CC	4334.50	0.00	158494.00		
343	C.CC	4334.50	0.00	158494.00		
344	C.CC	4334.50	0.00	158494.00		
345	C.CC	4334.50	0.00	158494.00		
346	C.CC	4334.50	0.00	158494.00		
347	C.CC	4336.50	694.00	159188.00		
348	C.CC	4336.50	0.00	159188.00		
349	C.CC	4336.50	0.00	159188.00		
350	C.CC	4336.50	0.00	159188.00		
OVER	350	C.CC	4336.50	0.00	159188.00	

Figure A.12--Continued



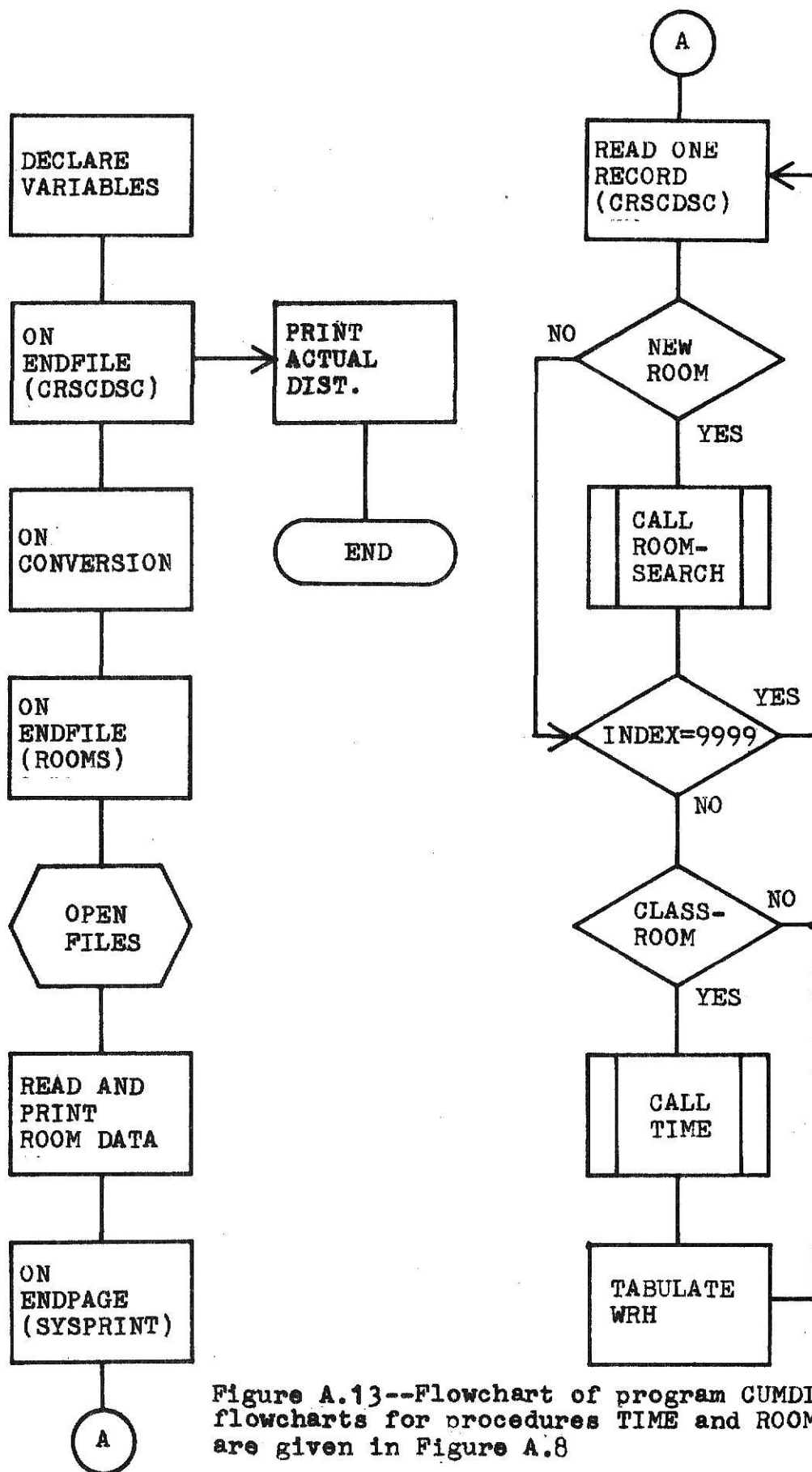


Figure A.13--Flowchart of program CUMDIST. The flowcharts for procedures TIME and ROOMSEARCH are given in Figure A.8

## CUMDIST

## SOURCE LISTING

```

// JOB (FH67U6C2,15,3),COMPTON,CLASS=B
// EXEC PL1LFCLG
//PL1L.SYSIN DD *
CUMDIST: PROCEDURE OPTIONS(MAIN);
DECLARE
    INPUTRECORD CHARACTER(80) BASED(P),
    LINECOUNT FIXED BINARY(31),
    NUMBER PICTURE '999999',
    NAME CHARACTER(12),
    TYPECODE CHARACTER(1),
    CKHR CHARACTER(2),
    DAYS(6) CHARACTER(1),
    FROMHR FIXED BINARY(31),
    (TOHR,TCMIN,FROMMIN) FIXED BINARY(31),
    RCOMUSED CHARACTER(6),
    INDEX FIXED BINARY(15),
    TYPE(4CC) CHARACTER(3),
    RCOMCARD CHARACTER(8C) BASED(Q),
    RCOMNUM(400) CHARACTER(6),
    GLDRCCM CHARACTER(6) INITIAL('999999'),
    TITLE CHAR(25),
    NIGHT CHARACTER(1),
    NR FIXED BINARY(31,7),
    (WRH(351),WSH(351)) FIXED BINARY(31,7) INITIAL((351) 0),
    (CUMWRH,CUMWSH) FIXED BINARY(31,7) INITIAL(0);
CN ENDFILE(CRSCDSC) GO TO OUTPUT;
CN CONVERSION BEGIN;
    IF CNSOURCE=1 THEN DO; CNSOURCE='00'; GOTO END; END;
    IF CNCHAR='H' THEN DO; CNCHAR='8'; GOTO END;END;
    IF CNCHAR='B' THEN DO; CNCHAR='2'; GOTO END;END;
    IF CNCHAR='C' THEN DO; CNCHAR='3'; GOTO END;END;
    IF CNCHAR='F' THEN DO; CNCHAR='6'; GOTO END;END;
    IF CNCHAR='I' THEN DO; CNCHAR='9'; GOTO END;END;
    IF CNCHAR='A' THEN DO; CNCHAR='1'; GOTO END;END;
    IF CNCHAR='S' THEN DO; CNCHAR='7'; GOTO END;END;
    IF CNCHAR='D' THEN DO; CNCHAR='4'; GOTO END;END;
    IF CNCHAR='E' THEN DO; CNCHAR='5'; GOTO END;END;
END: END;

```

```

CN FADFILE(RCCMS) GO TO NEXTA;
OPEN FILE(RCCMS) INPUT SEQUENTIAL RECORD,
FILE(SYSPRINT) OUTPUT PRINT PAGESIZE(59);
PUT SKIP EDIT('ROOM CARDS')(X(30),A);
I=0;
LOOPA: READ FILE(RCCMS) SET(Q);
I=I+1;
GET STRING(RCCMSCARD) EDIT(TYPE(I),ROOMNUM(I))(X(10),A(3),X(58),A(6));
PUT SKIP EDIT(RCCMSCARD)(X(10),A);
GO TO LOOPA;
NEXTA:
IRMAX=I;
CN ENDPAGE(SYSPRINT) BEGIN;
PUT PAGE EDIT(TITLE)(X(50),A);
PUT FILE(SYSPRINT) SKIP EDIT('DISTRIBUTION OF WEEKLY ROOM HOURS BY
SECTION SIZE (CONTINUED)')(X(40),A);
PUT FILE(SYSPRINT) SKIP(3) EDIT('SECTION SIZE WRH CUMWRH
WSH CUMWSH')(A);
PUT SKIP(2);
END;
GO TO CONTINUE;
READ: READ FILE(CRSCDSC) SET(P);
GET STRING(INPUTRECORD) EDIT(LINECOUNT,NUMBER,TYPECODE,NAME,CRHR,(DAYS(
I) CC I=1 TO 6),FROMHR,FROMMIN,TOHR,TOMIN,ROOMUSED)(F(6),X(26),
P'999999',A(1),X(4),A(12),A(2),6 A(1),P'R9',3 F(2),A(6));
IF ROOMUSED ^= OLDROOM THEN CALL RCCMSSEARCH(ROOMUSED, INDEX);
IF INDEX=999 THEN GO TO READ;
IF (TYPE(INDEX) ^= '110') THEN GO TO READ;
CALL TIME(WR);
WRH(LINECOUNT)=WRH(LINECOUNT)+WR;
WSH(LINECOUNT)=WSH(LINECOUNT)+WR#LINECOUNT;
GO TO READ;
CUTPUT:
TITLE='ACTUAL--FALL 1971';
CALL PRINT(TITLE);
CONTINUE:
PUT PAGE EDIT('DATA SET 3 (CORRECTED COURSE DATA)')(X(50),A);
CC I=1 TO 351;

```

```

GET EDIT(K,WRH(K),WSH(K))(F(8),2 F(6));
PUT EDIT(K,WRH(K),WSH(K))(F(8),2 F(6));
END;
TITLE='PROJECTED (DATA SET 3)';
CALL PRINT (TITLE);
PRINT: PROCEDURE(TITLE);
ECL TITLE CHAR(25);
PUT PAGE EDIT(TITLE)(X(50),A);
PUT SKIP EDIT('DISTRIBUTION OF WEEKLY ROOM HOURS BY SECTION SIZE')(X
(40),A);
PUT FILE(SYSPRINT) SKIP(3) EDIT('SECTION SIZE WRH CUMWRH
WSH CUMWSH')(A);
PUT SKIP(2);
CUMWRH=0;
CUMWSH=0;
DO I=1 TO 350;
CUMWRH=CUMWRH+WRH(I);
CUMWSH=CUMWSH+WSH(I);
PUT SKIP EDIT(I,WRH(I),CUMWRH,WSH(I),CUMWSH)(X(6),F(3),X(3),
F(8,2),F(11,2),F(9,2),F(13,2));
END;
CUMWRH=CUMWRH+WRH(351);
CUMWSH=CUMWSH+WSH(351);
PUT SKIP EDIT('OVER 350',WRH(351),CUMWRH,WSH(351),CUMWSH)(X(1),A(11)
,F(8,2),F(11,2),F(9,2),F(13,2));
END PRINT;
TIME: PROCEDURE(RT);
DECLARE RT FIXED BINARY(31,7);
IF DAYS(1)='A'
THEN DO;
NIGHT=' ';
RT=0;
GO TO END;
E.C;
JJ=0;
DO I=1 TO 6;
IF DAYS(I)=' '
THEN JJ=JJ+1;

```

```

END;
IF 0 > FROMHR
  THEN DO;
    NIGHT='E';
    FROMHR= -FROMHR;
  END;
  ELSE NIGHT='D';
  IF TCFR=0
    THEN DO;
      RT=JJ;
      GO TO END;
    END;
    FROMHR=FLOOR(FROMHR*2+DIVIDE(FROMMIN,30,31,7));
    TOHR=CEIL(TOHR*2+DIVIDE(TOMIN,30,31,7));
    IF FROMHR>TOHR
      THEN RT=DIVIDE((24-FROMHR+TOHR)*JJ,2,31,7);
      ELSE RT=DIVIDE((TOHR-FROMHR)*JJ,2,31,7);
    END;
  END TIME;
ROOMSEARCH: PROCEDURE(RCOM,INDEX);
  DECLARE ROOM CHARACTER(6);
  IF ROOM=' '
    THEN DO;
      INDEX=9999;
      GO TO END;
    END;
    ILCW=1;
    IHIGH=IRMAX;
    IF ROOM=ROOMNUM(IHIGH) THEN DO;
      INDEX=IHIGH;
      GO TO END;
    END;
    IF ROOM=ROOMNUM(LOW) THEN DO;
      INDEX=LOW;
      GO TO END;
    END;
  IF IRMAX/2;
CASE: IF ROOM= ROOMNUM(I)
  THEN DO;

```

```

INDEX=I;
GO TO END;
END;
IF (IHIGH-LOW)=1
  THEN DO;
    INDEX=999;
    PUT SKIP EDIT('*****RCOM ',ROOM,' NOT ON ROOM LIST.')(A,A(6
  )+A);
    GO TO END;
  END;
  IF ROOM > RCOMNUM(I)
    THEN DO;
      LOW=I;
      I=IHIGH-( IHIGH-LOW)/2;
      GO TO ONE;
    END;
  ELSE DO;
    IHIGH=I;
    I=LOW+( IHIGH-LOW)/2;
    GO TO ONE;
  END;
END;
END ROOMSEARCH;
END CUMDIST;
/*
//CC,ROOMS CC *
*****
CARDS CONTAINING
DATA ON CLASSROOMS AND CLASS LABORATORIES
*****
/*
//CC,CRSCC CC DSN=COFH7,CRSCC,UNIT=2314,VOL=SER=222222,
//  DISP=(OLD,KEEP)
//

```

## **APPENDIX B**

### **PERMANENT DATA FILES**



### GRADE TAPE

This was a tape provided by the office of Admissions and Records and used at the end of a semester for printing student grade reports. The fall 1970 tape was used in this investigation.

This tape contained two different record formats. One was a short student master and the other was the class file (grade report). One short student master was given for each student and one class file was given for each section of each course the student took. Students were arranged on the tape in order of ascending social security numbers.

The JCL necessary to use this tape follows:

```

C
C
1
//ddname DD DSN=ORGTAPE,UNIT=TAPE7,DISP=(SHR,DELETE),
//          VOL=SER=18851,LABEL=(,BLP),
//          DCB=(LRECL=81,BLKSIZE=1215,RECFM=FB,DEN=1
//          TRTCH=TE)

```

These records can be read just as punched-cards since each data element is in its character representation. The format follows.

### CLASS FILE

#### Bytes:

1- 9	Social security number
10-28	Student name
29	School
30	Class
31-32	Blank
33-38	Course number
39	Type code
40-43	Line number
44-55	Course name
56-57	Credit hours
58-63	Days of week
64-67	Hours from
78-71	Hours to
72-76	Room number
77-80	Blank
81	Record mark

## SHORT STUDENT MASTER

## Bytes:

1- 9	Social security number
10-28	Student name
29	School
30	Class
31	Residence code
32-52	Blank
53	Master tag
54	Assignment tag
55-57	Numerical curriculum
58	Veteran
59-60	High school
61	Sex
62-64	Date of entry
65-66	High school rank
67-68	Transfer college
69-71	City code
72-73	Year of birth
74	Dual curriculum
75-77	Alphabetic curriculum
78-79	Blank
80	WD code
81	Record mark

ROOM DATA

This was a deck of punched-cards used in University facility utilization and review studies. It contained information on classrooms and class laboratories.

The only information contained on these cards and used in this investigation was room number and room type code. The following information was given by each card:

Characters:

1-10	Capacity
11-13	Room type code
14-29	Building name
30-32	Building code number
33-39	Room area
40-42	Department code
43-71	Blank
72-77	Room number
78-79	Blank
80	Minus sign

These cards were listed on the print-outs of each program which used them. A listing of these cards is given in Figure B.1.

## INPLT ROOM DATA

1021CANDERSON HALL	001	192331	A 201C	-
3035CANDERSON HALL	001	754331	A 211J	-
30350ANDERSON HALL	001	754331	A 221J	-
2021CART ANNEX	086	92C3C9	AA101	-
15210ART ANNEX	086	7493C9	AA102	-
32210WEBER HALL	004	1050204	AI103	-
297110WEBER HALL	004	2475	AI107	-
50210WEBER HALL	004	14499204	AI122	-
56110WEBER HALL	004	C0831	AI129	-
81110WEBER HALL	004	1057	AI230	-
322CAUDITORIUM	133	259	AU101	-
3220AUDITORIUM	133	376	AU102	-
4011CAUDITORIUM	133	1042	AU105	-
13522CAUDITORIUM	133	1903	AU201	-
17022CAUDITORIUM	133	1903	AU204	-
2411CBIO SCIENCE	136	346	BS105	-
2411CBIO SCIENCE	136	346	BS106	-
4021CBIO SCIENCE	136	1364310	BS112	-
40210BIO SCIENCE	136	1393310	BS114	-
6011CBIO SCIENCE	136	848	BS116	-
17811CBIO SCIENCE	136	2269	BS120	-
40210BIO SCIENCE	136	1412310	BS121	-
40210BIO SCIENCE	136	1373310	BS123	-
84210BIO SCIENCE	136	3152310	BS219	-
105110BIO SCIENCE	136	1255	BS221	-
2021CBIO SCIENCE	136	815310	BS226	-
20210BIO SCIENCE	136	815310	BS227	-
40210BIO SCIENCE	136	1411310	BS228	-
40210BIO SCIENCE	136	1381310	BS230	-
2021CBIO SCIENCE	136	795310	BS319	-
20210BIO SCIENCE	136	835310	BS321	-
2021CBIO SCIENCE	136	815310	BS322	-
2021CBIO SCIENCE	136	835310	BS323	-
35110CALVIN HALL	013	914333	C 18	-
54110CALVIN HALL	013	924	C 102	-
63110CALVIN HALL	013	922	C 116	-
20210CALVIN HALL	013	7087C1	C 201	-
70110CALVIN HALL	013	916	C 202	-
56110CALVIN HALL	013	952	C 208	-
36110CALVIN HALL	013	960	C 209	-
59110CALVIN HALL	013	931	C 211	-
76110CALVIN HALL	013	912	C 212	-
30210CALVIN HALL	013	9147C1	C 217	-
60110DICKENS HALLS	018	684	D 106	-
110110DICKENS HALLS	018	1140	D 108	-
42210DICKENS HALL	018	1140310	D 201	-
25210DICKENS HALL	018	976214	D 206	-
20210DICKENS HALL	018	910310	D 207	-
15210DICKENS HALL	018	704214	D 302	-
224110DENISON	017	2508	DE113A	-
36110DENISON	017	500	DE116A	-
36110DENISON	017	500	DE116B	-
101110DENISON	017	1085	DE117	-
103110DENISON	017	1083	DE118	-
98110DENISON	017	1085	DE215	-
99110DENISON	017	1086	DE216	-
108110DENISON	017	1085	DE217	-

Figure B.1--Room data used in the investigation.

108110DENISON	017	1086	DE218	-
108110DENISON	017	1C85	OE219	-
32210CALL HALL	072	1079205	OP156	-
32210CALL HALL	072	1024205	DP160	-
35210CALL HALL	072	1368205	DP202	-
53110CALL	072	762	DP204	-
45210CALL HALL	072	1378205	DP226	-
20210SEATON HALL	102	1130263	E 4A	-
15210SEATON HALL	102	2552404	E 11	-
12210SEATON HALL	102	448404	E 12	-
14210SEATON HALL	102	714408	E 32A	-
16210SEATON HALL	102	4650402	E 35	-
16210SEATON HALL	102	4665402	E 37	-
34610SEATON	102	3067	E 63	-
15220SEATON HALL	102	3171410	E 106	-
15210SEATON	102	587410	E 107	-
50110SEATON	102	720	E 127	-
36110SEATON	102	731	E 129	-
25110SEATON	102	344	E 130	-
27110SEATON	102	360	E 130A	-
20210SEATON HALL	102	354407	E 131	-
26110SEATON	102	360	E 132	-
40210SEATON HALL	102	3480402	E 134	-
20210SEATON HALL	102	3715402	E 138	-
32110SEATON	102	608	E 143	-
50110SEATON	102	835	E 161	-
29110SEATON	102	472	E 162	-
12210SEATON HALL	102	498408	E 164G	-
12210SEATON HALL	102	770408	E 164H	-
15210SEATON HALL	102	849408	E 164I	-
12210SEATON HALL	102	850408	E 164J	-
8210SEATON HALL	102	461404	E 168	-
29110SEATON	102	558	E 202	-
41210SEATON HALL	102	837410	E 204	-
27110SEATON	102	594	E 205	-
42210SEATON HALL	102	2268263	E 206	-
24210SEATON HALL	102	1230263	E 218	-
20210SEATON HALL	102	1001263	E 219	-
20210SEATON HALL	102	720407	E 220A	-
10220SEATON HALL	102	737407	E 221	-
25210SEATON HALL	102	827407	E 222	-
25210SEATON HALL	102	1848402	E 226	-
20210SEATON HALL	102	612402	E 235	-
64110SEATON	102	835	E 236	-
15210SEATON HALL	102	666412	E 237	-
28110SEATON	102	467	E 242	-
34210SEATON HALL	102	892263	E 251	-
10210SEATON HALL	102	425263	E 252	-
20210SEATON HALL	102	1272410	E 254H	-
24210SEATON HALL	102	1712410	E 254I	-
22210SEATON HALL	102	1222410	E 254J	-
30210SEATON HALL	102	1140272	E 257	-
21210SEATON HALL	102	1465267	E 257A	-
21210SEATON HALL	102	1465267	E 257B	-
18210SEATON HALL	102	788263	E 260	-
18210SEATON HALL	102	768263	E 260B	-
40210SEATON HALL	102	1765260	E 260D	-
40210SEATON HALL	102	1916272	E 301	-
18210SEATON HALL	102	924269	E 303	-
28210SEATON HALL	102	1074260	E 305	-

Figure B.1--(Continued).

20210SEATCH HALL	102	748269	E 306	-
18210SEATCH HALL	102	1001269	E 307	-
20210ENVIR CES	102	1221	E 307A	-
20210ENVIR CES	102	1221	E 307B	-
20210SEATCH HALL	102	1221269	E 308	-
10210SEATCH HALL	102	467269	E 314	-
76210SEATCH HALL	102	736263	E 317	-
18210SEATCH HALL	102	933272	E 320	-
36210SEATCH HALL	102	2468410	E 322	-
30110EAST STADIUM	104	438	ES104A	-
10210EAST STADIUM	104	232328	ES202	-
16210FAIRCHILD	030	645	F 117	-
105110FAIRCHILD	030	923	F 202	-
62110FAIRCHILD	030	616	F 208	-
10210FEED TECHNOLOG	029	832207	FT202	-
30220HOLTON	034	511360	G 1	-
30220HOLTON	034	520360	G 4	-
55110HOLTON	034	863	G 204	-
50110HOLTON	034	560	G 206D	-
34110EISENHOWER	022	477	J 11	-
31110EISENHOWER	022	476	J 12	-
100110EISENHOWER	022	1049	J 15	-
35110EISENHOWER	022	517	J 16	-
31110EISENHOWER	022	516	J 20	-
38110EISENHOWER	022	500	J 21	-
36220EISENHOWER	022	559	J 22	-
15210EISENHOWER	022	531332	J 26	-
30110EISENHOWER	022	478	J 112	-
33110EISENHOWER	022	525	J 121	-
32110EISENHOWER	022	512	J 122	-
38110EISENHOWER	022	563	J 123	-
35110EISENHOWER	022	512	J 124	-
39110EISENHOWER	022	625	J 125	-
41110EISENHOWER	022	555	J 126	-
16110EISENHOWER	022	344	J 201	-
40110EISENHOWER	022	493	J 211	-
24110EISENHOWER	022	360	J 212	-
40110EISENHOWER	022	495	J 218	-
35110EISENHOWER	022	545	J 219	-
37110EISENHOWER	022	545	J 223	-
40110EISENHOWER	022	515	J 224	-
40110EISENHOWER	022	572	J 225	-
38110EISENHOWER	022	515	J 226	-
33110EISENHOWER	022	562	J 227	-
40110EISENHOWER	022	562	J 228	-
205110JUSTIN	041	2094	JU109	-
16210JUSTIN HALL	041	1388507	JU110	-
24210JUSTIN HALL	041	937506	JU115	-
20210JUSTIN HALL	041	1110506	JU143	-
20210JUSTIN HALL	041	1101506	JU144	-
20210JUSTIN HALL	041	1093506	JU145	-
15210JUSTIN HALL	041	1093506	JU146	-
20210JUSTIN HALL	041	879506	JU147	-
42110JUSTIN	041	510	JU148	-
48110JUSTIN	041	725	JU149	-
39110JUSTIN	041	563	JU150	-
24210JUSTIN HALL	041	790503	JU226	-
56110JUSTIN	041	570	JU249	-
20210JUSTIN HALL	041	1484503	JU250	-
99110JUSTIN	041	1448	JU251	-

Figure B.1--(Continued).

20210JUSTIN HALL	041	1449503	JU252	-
20210JUSTIN HALL	041	1484503	JU253	-
33110JUSTIN	041	590	JU254	-
60110JUSTIN	041	718	JU256	-
20210JUSTIN HALL	041	912503	JU301	-
33210JUSTIN HALL	041	1145309	JU326	-
18210JUSTIN HALL	041	1301505	JU327	-
25210JUSTIN HALL	041	1152505	JU329	-
12220JUSTIN HALL	041	703309	JU337	-
20210JUSTIN HALL	041	596309	JU338	-
52110JUSTIN	041	620	JU341	-
23210JUSTIN HALL	041	726309	JU343	-
20210JUSTIN HALL	041	728309	JU344	-
20210JUSTIN HALL	041	595309	JU345	-
20210JUSTIN HALL	041	608503	JU346	-
12210JUSTIN HALL	041	604503	JU347	-
20210JUSTIN HALL	041	606309	JU348	-
20210JUSTIN HALL	041	700503	JU349	-
20210JUSTIN HALL	041	697503	JU351	-
21210KEDZIE	071	2222	K 106	-
30210KEDZIE HALL	071	944334	K 107	-
24210KEDZIE HALL	071	1256314	K 208	-
24210KEDZIE	071	1256334	K 208L	-
35110KEDZIE	071	695	K 210	-
43110KEDZIE	071	495	K 214	-
23110KEDZIE	071	378	K 216	-
104110KING HALL	020	1178	KG 4	-
18210KING HALL	020	817314	KG101	-
18210KING HALL	020	816314	KG102	-
18210KING HALL	020	817314	KG103	-
18210KING HALL	020	831314	KG106	-
18210KING HALL	020	816314	KG107	-
18210KING HALL	020	817314	KG108	-
23110KING HALL	020	435	KG110	-
18210KING HALL	020	816314	KG202	-
18210KING HALL	020	817314	KG203	-
27110KING HALL	020	413	KG204	-
18210KING HALL	020	831314	KG205	-
18210KING HALL	020	816314	KG206	-
18210KING HALL	020	817314	KG207	-
14210KING HALL	020	1193314	KG301	-
14210KING HALL	020	817314	KG302	-
25110KING HALL	020	384	KG313	-
20210MILLING LAB W1	080	573207	MI105	-
15210MILLING LAB W1	080	1554207	MI110	-
20210MILITARY SCIEN	079	3712328	MS 1	-
62110MILITARY SCIEN	079	958	MS 7	-
100210MILITARY SCIEN	079	5070326	MS 8	-
30110MILITARY SCIEN	079	1419	MS 11	-
40110MILITARY SCIEN	079	718	MS201	-
42110MILITARY SCIEN	079	707	MS204	-
30110MILITARY SCI	079	446	MS205	-
48110MILITARY SCIEN	079	718	MS210	-
40110MILITARY SCIEN	079	669	MS211	-
50110MILITARY SCIEN	079	962	MS212	-
35110MILITARY SCIEN	079	718	MS213	-
23520NICHOLS GYM	082	4400329	N 2	-
31520NICHOLS GYM	082	5771329	N 4	-
8210WARD HALL	085	725412	NL113	-
354110PHYSICAL SCIEN	091	3603	PS101	-

Figure B.1--(Continued).

118110PHYSICAL SCIEN 091	1410	PS102	-
210110PHYSICAL SCIEN 091	2058	PS103	-
40110PHYSICAL SCIEN 091	560	PS120	-
38110PHYSICAL SCIEN 091	549	PS121	-
38110PHYSICAL SCIEN 091	549	PS122	-
39110PHYSICAL SCIEN 091	549	PS123	-
36110PHYSICAL SCIEN 091	549	PS124	-
36110PHYSICAL SCIEN 091	555	PS127	-
44110PHYSICAL SCIEN 091	691	PS129	-
45110PHYSICAL SCIEN 091	691	PS130	-
45110PHYSICAL SCIEN 091	691	PS131	-
47110PHYSICAL SCIEN 091	691	PS132	-
52110PHYSICAL SCIEN 091	804	PS143	-
46110PHYSICAL SCIEN 091	803	PS144	-
48110PHYSICAL SCIEN 091	797	PS145	-
69110PHYSICAL SCIEN 091	784	PS146	-
25210PHYSICAL SCIEN 091	1251330	PS216	-
25210PHYSICAL SCIEN 091	1241330	PS218	-
20210PHYSICAL SCIEN 091	1251330	PS219	-
20210PHYSICAL SCIEN 091	1194330	PS221	-
25210PHYSICAL SCIEN 091	1200330	PS222	-
20210PHYSICAL SCIEN 091	1193330	PS223	-
20210PHYSICAL SCIEN 091	1203330	PS224	-
20210PHYSICAL SCIEN 091	1241330	PS312	-
25520MENS GYM 073	771329	R 12	-
20520MENS GYM 073	1613329	R 101	-
100525MENS GYM 073	19096329	R 116	-
42110MENS GYM 073	452	R 204	-
42110MENS GYM 073	447	R 205	-
65110MENS GYM 073	870	R 302	-
50110MENS GYM 073	601	R 303	-
40210MENS GYM 073	1842329	R 304	-
20210ENGINEERING SH 025	1388409	S 105	-
40210ENGINEERING SH 025	3697409	S 108	-
15220ENGINEERING SH 025	408263	S 113	-
10220ENGINEERING SH 025	835267	S 114	-
25220ENGINEERING SH 025	805360	S 201	-
68110ENGINEERING SH 025	902	S 204	-
48110ENGINEERING SH 025	799	S 206	-
30110ENGINEERING SH 025	799	S 207	-
36110ENGINEERING SH 025	675	S 208	-
32210THOMPSON HALL 106	960322	T 15	-
122110THOMPSON 106	1273	T 101	-
38110THOMPSON 106	975	T 109	-
11210THOMPSON HALL 106	549322	T 201	-
15210THOMPSON HALL 106	665321	T 208	-
28210THOMPSON 106	735322	T 212	-
53110THOMPSON 106	818	T 213	-
25210THOMPSON HALL 106	724322	T 216	-
30110TRAILER C 140	589	TR C	-
40210VETERINARY HAL 112	1622604	V 10	-
85110VETERINARY HAL 112	2353	V 13A	-
33210VETERINARY HAL 112	1854310	V 201	-
18210VETERINARY HAL 112	758310	V 207	-
10220DYKSTRA HALL 019	1257603	VH174	-
80210DYSTRA HALL 019	3072	VH175	-
10220DYKSTRA HALL 019	905602	VH177	-
50210BURT HALL 009	2497601	VS102	-
86110BURT HALL 009	880	VS109	-
250110WILLARD HALL 116	2205	W 114	-

Figure B.1--(Continued).



44110WILLARD HALL	116	596	W 115	-
40210WILLARD HALL	116	1312314	W 116	-
30210WILLARD HALL	116	847314	W 120	-
40212WILLARD HALL	116	1407212	W 201	-
20210WILLARD HALL	116	839212	W 202	-
33210WILLARD HALL	116	1263314	W 215	-
56210WILLARD HALL	116	1811314	W 216	-
56110WILLARD HALL	116	681	W 218	-
48210WILLARD HALL	116	1607314	W 318	-
48210WILLARD HALL	116	1560314	W 319	-
48210WILLARD HALL	116	1324212	W 320	-
24210WATERS HALL	120	1271203	WA 9	-
18210WATERS HALL	120	726203	WA 10	-
22210WATERS HALL	120	962208	WA 18	-
20210WATERS HALL	120	774208	WA 41	-
28210WATERS HALL	120	921206	WA124A	-
12210WATERS HALL	120	614206	WA124B	-
43110WATERS HALL	120	503	WA126	-
15210WATERS HALL	120	515208	WA127	-
52110WATERS HALL	120	587	WA132	-
40210WATERS HALL	120	1378207	WA150	-
229110WATERS HALL	120	1895	WA231	-
20210WATERS HALL	120	640340	WA239	-
59110WATERS HALL	120	921	WA244	-
41110WATERS HALL	120	483	WA253	-
32210WATERS HALL	120	1195203	WA304	-
101110WATERS HALL	120	1010	WA328	-
39110WATERS HALL	120	515	WA332	-
62110WATERS HALL	120	952	WA348	-
66110WATERS HALL	120	959	WA350	-
15220WEST STADIUM	105	368309	WS104	-
15220WEST STADIUM	105	151309	WS110	-
15220WEST STADIUM	105	537309	WS116	-
10210WEST STADIUM	105	490309	WS117	-
15110WEST STADIUM	105	378309	WS118	-
15220WEST STADIUM	105	378309	WS123	-
15220WEST STADIUM	105	223309	WS211	-
20210CHEMICAL ENGG	014	2646406	XX 6	-
25110CHEM ENG	014	509	XX103	-
44110CHEM ENG	014	595	XX206	-

Figure B.1--(Continued)..

### LINE SCHEDULE DATA

This was a deck of punched-cards used in University facility utilization and review studies. It contained information on each section of each course which met in the fall 1971 semester.

The format of these cards follows:

Characters:

1- 6	Number of students in section
7-10	Credit hours multiplied by students in section
11-12	Blank
13	Minus sign
14-17	Corrected number of students in section
18-21	Corrected number of students in section multiplied by credit hours
22-27	Blank
28-32	Department abbreviation
33-38	Course number
39	Type code
40-43	Line number
44-55	Course name
56-57	Credit hours
58-63	Days of week
64-67	Hour from
68-71	Hour to
72-77	Room number
78-80	Blank

A sample listing of these cards is given in Figure B.2.

00250000	SPCH	281106B30190RAL	COMM	IA	T	T	0830	J	11
00240000	SPCH	281106B30200RAL	COMM	IA	M	W	0730	F	208
00250000	SPCH	281106B30210RAL	COMM	IA	M	W	0730	J	11
00230000	SPCH	281106B30220RAL	COMM	IA	M	W	0730	J	21
00260000	SPCH	281106B30230RAL	COMM	IA	M	W	0830	J	21
00240000	SPCH	281106B30240RAL	COMM	IA	M	W	0430	J	21
00260000	SPCH	281106B30250RAL	COMM	IA	T	T	0730	F	208
00240000	SPCH	281106B30260RAL	COMM	IA	T	T	0830	F	208
00250000	SPCH	281106B30270RAL	COMM	IA	T	T	1230	J	228
00240000	SPCH	281106B30280RAL	COMM	IA	T	T	0230	F	208
00240000	SPCH	281106B30290RAL	COMM	IA	T	T	0330	J	21
00240852	SPCH	281106430300RAL	COMM	IA03	F		1130	AI107	
00290000	SPCH	281106B30310RAL	COMM	IA	T	T	1130	F	208
00260000	SPCH	281106B30320RAL	COMM	IA	T	T	1030	F	208
00260000	SPCH	281106B30330RAL	COMM	IA	T	T	1130	J	11
00240000	SPCH	281106B30340RAL	COMM	IA	T	T	1230	J	11
00240000	SPCH	281106B30350RAL	COMM	IA	T	T	0130	J	20
00240000	SPCH	281106B30360RAL	COMM	IA	T	T	0130	J	26
00250000	SPCH	281106B30370RAL	COMM	IA	M	W	0830	J	20
00240000	SPCH	281106B30380RAL	COMM	IA	M	W	0330	J	21
00250000	SPCH	281106B30390RAL	COMM	IA	M	W	0330	J	20
00260000	SPCH	281106B30400RAL	COMM	IA	T	T	0230	J	20
00230000	SPCH	281106B30410RAL	COMM	IA	T	T	0330	J	20
00230000	SPCH	281106230420RAL	COMM	IA03M	W	F	1030	J	21
00270081	SPCH	281106230430RAL	COMM	IA03	T	T	09051030J	228	
00280084	SPCH	281106230440RAL	COMM	IA03M	W	F	1130	J	225
00280084	SPCH	281106230450RAL	COMM	IA03M	W	F	0130	F	208
00270081	SPCH	281106230460RAL	COMM	IA03M	W	F	1130	J	224
00260018	SPCH	281106230630RAL	COMM	IA03M	W	F	0330	J	224
00250000	SPCH	281106B31790RAL	COMM	IA	M	W	0230	J	228
00190000	SPCH	281106B31800RAL	COMM	IA	T	T	0330	J	26

Figure B.2---Sample of line schedule data.

### DATA SETS ON BACK-UP TAPE

There were four data sets saved on a back-up magnetic tape. Three of these were created by this investigation and the fourth is a copy of an additional ICLM, the one for the spring 1971 semester. The spring ICLM was made for another project but was recorded on this back-up tape for convenience.

The four data sets on this tape are:

1. The output of ONE sorted by course, curriculum, and level.
2. The ICLM for fall 1970 sorted by course, curriculum, and level.
3. The ICLM for fall 1970 sorted by curriculum, level, and course.
4. The ICLM for spring 1971 sorted by course, curriculum, and level.

#### Data Set 1:

To use this data set the following JCL is needed:

```

8
1
//ddname DD DSN=BACKUP,UNIT=TAPE9,DISP=(OLD,KEEP),
//          VOL=SER=907201,DCB=(LRECL=20,BLKSIZE=2000,
//          RECRM=FB),LABEL=1

```

A record of this data set contains the following information:

Bytes:	
1-6	course number
8-10	curriculum number
12	level
13-19	sequence number of record as given by ONE.
	This value is used only for padding the record to reach minimum record length.

This record was written with the following PL/1 format:

```
(P'999999'.X(1),P'999',X(1),P'9',F(7));
```

Data Set 2:

To use the second data set, the following JCL is needed:

```

&
1
//ddname DD DSN=CURPROC,VOL=SER=907201,UNIT=TAPE9,
//          DISP=(OLD,KEEP),LABEL=2,DCB=(LRECL=20,BLKSIZE=2000,
//          RECFM=FB)

```

A record of this data set contains the following information:

Bytes:	
1-6	course number
7-9	curriculum number
10	level
11-13	curriculum name (alphabetic abbreviation)
14-18	element of ICLM
19-20	number of students in column and row category.

This record was written directly from the following PL/1 data structure:

```

1 OUTDATA BASED(Q),
2 COURS CHARACTER(6),
2 CUR PICTURE '999',
2 LEV PICTURE '9',
2 CURALPH CHARACTER(3),
2 FRAC PICTURE'V.9999',
2 COUNTED FIXED BINARY(15),

```

Data Set 3:

To use the third data set the following JCL is needed:

```

&
1
//ddname DD DSN=ICLM,VOL=SER=907201,UNIT=TAPE9,
//          DISP=(OLD,KEEP),DCB=(LRECL=20,BLKSIZE=2000,
//          RECFM=FB),LABEL=3

```

The records of this data set contain the same information in the same format as the second data set.

Data Set 4:

To use the forth data set the following JCL is needed:

```

&
1
//ddname DD DSN=ICLMS71,VOL=SER907201,UNIT=TAPE9,
//          DISP=(OLD,KEEP),DCB=(LRECL=20,BLKSIZE=2000,
//          RECFM=FB),LABEL=4

```

The records of this data set contain the same information in the same format as the second data set.

### COURSE DATA

This was a deck (box and a half) of punched-cards. This information was produced by CRSEDIT and written onto a direct access device. It was later punched onto cards so hand adjustments could be made. Section 3.1 lists the data (after hand adjustments) contained in these cards for each course. Since the format for reading these cards is variable, the procedure used in DISTWRH to read the course data for one course is listed in Figure B.3.

The course data was read into a PL/1 data structure and this structure is given in Figure B.4.

The variables used in the procedure listed in Figure B.3 correspond to the following items of course data as explained in section 3.1. Other variables having a strictly data processing use are also explained.

Variable:	Description:
CRSEDIT .....	Name of the file containing the course data.
COURSID .....	Part of the above course data structure.
COURSE .....	Course number.
INDICATOR .....	Number of modes of instruction for a course. It is equal to zero if only one mode is present.
ALPHID .....	Course name.
CR .....	Credit hours.
CHECKCOURSE ....	The course number is present on all cards belonging to a course as a check against cards being out of sequence. The course number is read into this variable on all cards of a course except the first.
KERNEL .....	This names part of the course data structure. It would have been better named mode of instruction.

```

READCRSLATA: PROCEDURE;
GET FILE(CRSEDT) EDIT(CCURSIC)(A(6),F(1),A(13),A(2));
IF INDICATOR ^= 0 THEN CC;
DO I=1 TO INDICATOR;
  GET FILE(CRSEDT) SKIP EDIT(CHECKCOURSE,KSECTYPES(I),
    (KCOUNTROOMS(I,J) DO J=1 TO KSECTYPES(I))(A(6),X(1),F(3),20 F(3)));
  IF CHECKCOURSE ^= CCOURSE THEN PUT SKIP EDIT ('DATA READ OUT OF SEQUEN
    CE COURSE=',CCOURSE,' COURSE=',CHECKCOURSE)(A,A(6),A,A(6));
  DO J=1 TO KSECTYPES(I);
    GET FILE(CRSEDT) SKIP EDIT(CHECKCOURSE,KSECDATA(I,J),
    (KCOUNTROOMS(I,J,K) DO K=1 TO KCOUNTROOMS(I,J))(A(6),X(1),A(2),F(4),
    F(8,3),X(2),F(3),3 A(2),20 A(7)));
    IF CHECKCOURSE ^= CCOURSE THEN PUT SKIP EDIT ('DATA READ OUT OF SEQUEN
    CE COURSE=',CCOURSE,' COURSE=',CHECKCOURSE)(A,A(6),A,A(6));
  END;
END;
GET FILE(CRSEDT) SKIP;
ELSE DO;
  INDICATOR=1;
  GET FILE(CRSEDT) EDIT(KSECTYPES(1),(KCOUNTROOMS(1,J) DO J=1 TO
    KSECTYPES(1))(F(3),20 F(3)));
  DO J=1 TO KSECTYPES(1);
    GET FILE(CRSEDT) SKIP EDIT(CHECKCOURSE,KSECDATA(1,J),(KCOUNTROOMS(1,J,
    K) DO K=1 TO KCOUNTROOMS(1,J))(A(6),X(1),A(2),F(4),F(8,3),X(2),F(3),
    3 A(2),20 A(7)));
    IF CHECKCOURSE ^= CCOURSE THEN PUT SKIP EDIT ('DATA READ OUT OF SEQUEN
    CE COURSE=',CCOURSE,' COURSE=',CHECKCOURSE)(A,A(6),A,A(6));
  END;
GET FILE(CRSEDT) SKIP;
END;
END READCRSLATA;

```

Figure B.3--Source listing of the procedure used to read the course data for one course.

```

1 COURSDATA CONTROLLED,
  2 COURSID,
    3 COURSE CHARACTER(6),
3 INDICATOR FIXED BINARY(15),
  3 ALPHID CHARACTER(12),
    3 CR CHARACTER(2),
2 KERNEL(3),
  3 KSECTYPES FIXED BINARY(15),
  3 KCOUNTROOMS(20) FIXED BINARY(15),
  3 KSECCTYPES(10),
    4 KSFCDATA,
      5 KSECTYPE CHARACTER(1),
      5 KSECSIZE FIXED BINARY(31),
      5 KWRH FIXED BINARY(31,7),
      5 KQUANTITY FIXED BINARY(31),
      5 KREC CHARACTER(1),
      5 KEVENING CHARACTER(1),
      5 KLAB CHARACTER(1),
    4 KROOMS(20) CHARACTER(6),

```

Figure B.4--The PL/I data structure used to store the course data for one course.

```

2811062ORAL COMM IA 03 3 0 0 0
281106 2 C
281106 4 288 1.000 2R D
281106 8 24 2.000 24R D
281106 1 C
281106 2 24 3.000 6R D

```

Figure B.5--Sample cards of the course data for one course.



Variable:	Description:
KSECTYPES .....	This is the number of different types of sections for a mode of instruction of a course.
KCOUNTRROOMS ....	This is the number of laboratory rooms for each section type.
KSECCTYPES .....	Names part of the course data structure.
KSECDATA .....	Names part of the course data structure.
KSECTYPE .....	Type of section.
KSECSIZE .....	Section size.
KWRH .....	Weekly room hours.
KQUANTITY .....	Number of sections on which section size is based.
KREC .....	Type of facility.
KEVENING .....	Evening or daytime class meeting.
KLAB .....	Type of facility.
KROOMS .....	Indicated class laboratories.

Sample cards of the course data are given in Figure B.5.

AN INQUIRY INTO THE PROBLEM OF PREDICTING  
UNIVERSITY CLASSROOM REQUIREMENTS

by

THOMAS EDWIN COMPTON

B. S., Kansas State University, 1970

---

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Industrial Engineering

KANSAS STATE UNIVERSITY  
Manhattan, Kansas

1972

One method for projecting a university's classroom requirements for a predicted student population was investigated. The essentials of this method were proposed by the Western Interstate Commission for Higher Education (WICHE). Fundamental to WICHE's method was an induced course load matrix (ICLM) used to project course enrollments for a predicted student population. This matrix was successfully constructed from data available for one fall semester at Kansas State University. Data on course classroom facility requirements was successfully extracted from the University's data base and made available in machine processable form. Rules necessary for dividing course enrollments into individual sections of courses were proposed. It was successfully shown that, given course enrollments, the extracted course facility requirements and the proposed sectioning rules could be used to determine classroom requirements for courses.