
Masters Theses

Student Theses and Dissertations

1972

Computerized registration for high schools

Kenneth Lee Fore

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



Part of the [Computer Sciences Commons](#)

Department:

Recommended Citation

Fore, Kenneth Lee, "Computerized registration for high schools" (1972). *Masters Theses*. 5081.
https://scholarsmine.mst.edu/masters_theses/5081

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

COMPUTERIZED REGISTRATION
FOR HIGH SCHOOLS

BY

KENNETH LEE FORE, 1939-

A THESIS

Presented to the Faculty of the Graduate School of the

UNIVERSITY OF MISSOURI-ROLLA

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN COMPUTER SCIENCE

1972

T2743
89 pages
c.I

Approved by

AK Ryker

(Advisor)

Thomas Baird

Robert S. Montgomery

ABSTRACT

The guidance personnel in a high school are often burdened with the duty of hand scheduling the student's courses. Using the computer as a sub-optimizing tool in the registration of students, however, can cut scheduling time dramatically and relieve the guidance department of an onerous chore.

The technique described in this investigation uses a conflict matrix that schedules the student's request and keeps the class load level within the course sections.

A search of the schedule array for each course request may uncover a conflict. If no conflict occurs the course is scheduled and the remaining courses for this student are examined in the same way. If a conflict does occur, a back-tracking procedure reconsiders this partially completed schedule.

The system was implemented on an 8K IBM 1130 System at Waynesville Senior High School, Waynesville, Missouri.

ACKNOWLEDGEMENTS

The author wishes to extend his sincere appreciation to his advisor, Professor A. K. Rigler, for his guidance in this project; to Gale Townsend, Director of Guidance for the Waynesville-Fort Leonard Wood School District, for his help; and his secretary, Miss Carol Farris, for her excellent typing.

The author thanks the students of Waynesville Senior High who were the data for this project, and the principal Loyd Clarkston for his valuable assistance on the Course Master Schedule.

TABLE OF CONTENTS

| | Page |
|---|------|
| ABSTRACT..... | ii |
| ACKNOWLEDGEMENT..... | iii |
| LIST OF ILLUSTRATIONS..... | vi |
| I. INTRODUCTION..... | 1 |
| II. REVIEW OF LITERATURE..... | 3 |
| A. HISTORICAL CONTEXT OF SCHEDULING IN SECONDARY EDUCATION..... | 6 |
| B. MANUAL APPROACH TO SCHEDULING..... | 9 |
| 1. Course Master Schedule Generation..... | 9 |
| 2. Sectioning of Student Requests..... | 16 |
| C. MACHINE APPROACH TO STUDENT SECTIONING.... | 18 |
| 1. Sectioning of Student Requests..... | 18 |
| 2. Course Master Scheduling Generation... | 19 |
| III. A BLOCK SCHEDULER FOR WAYNESVILLE HIGH SCHOOL... | 36 |
| A. DESCRIPTION OF WAYNESVILLE HIGH SCHOOL CHARACTERISTICS | 36 |
| B. DESCRIPTION OF SCHEDULING..... | 37 |
| C. DETAILS OF OPERATION OF THE WAYNESVILLE HIGH SCHOOL COMPUTER PROGRAM..... | 40 |
| IV. SUMMARY..... | 57 |
| BIBLIOGRAPHY..... | 60 |
| VITA..... | 64 |

| Table of Contents (continue) | Page |
|---|------|
| APPENDICES | |
| A. 100 Uses for School Data Processing..... | 65 |
| B. Cardinal Principles of Secondary Education (1918)..... | 72 |
| C. Waynesville High School Registration Year (1971-1972)..... | 74 |
| D. Sample Conflict Matrix..... | 75 |
| E. Manual Enrollment Procedure for Waynesville High School, 1970-71..... | 76 |
| F. Sample Course Descriptions..... | 80 |
| G. Sectioning Logic Sequence a Small School..... | 81 |
| H. Scheduling Costs..... | 83 |

LIST OF ILLUSTRATION

| Figure | Page |
|--|------|
| 1. Manual "Mail Box" Sectioning of Students..... | 17 |
| 2. Section of Availability Array for t_1 | 21 |
| 3. Course Master List (CMSTR)..... | 42 |
| 4. Request File (REQST)..... | 43 |
| 5. Student Sectioning System Flowchart..... | 45 |
| 6. Course Master to Section File Index (CINDX)..... | 46 |
| 7. Priority Sorting Array of Student's Requests..... | 48 |
| 8. Student Sectioning Algorithm Flowchart..... | 50 |
| 9. Class-Time Matrix Scheduling..... | 52 |
| 10. Section File (Sect)..... | 55 |

I. INTRODUCTION

With the economic condition of public schools worsening, school administrators must rely more and more on automation and the computer in maximizing the quality of education with the amount of resources available. Many manual chores of administration that are already automated in business must receive similar attention in school districts. It is unfortunate that the staff of most secondary schools are burdened with the task of performing the detailed, yet essential, clerical functions, semester after semester, year after year, when their time is urgently needed for instruction of the students, professional counseling, guidance, and the continuous evaluation of the system to shape the policies necessary to move the education process forward.

The purpose of this study was to deal with one of the most persistent and unresolved problems facing secondary school administrators today: how can a computer be used to construct an optimum schedule of classes. If resolved, it should be possible to maximize the number of students successfully scheduled, realize the most favorable use of teachers and rooms available, minimize the number of student course request conflicts which result in the process, keep multi-section courses balanced in size, and free administrators, teachers and guidance personnel from clerical duties that can be automated.

The scheduling system developed in this study has two general phases: First, building a course master schedule, and second, sectioning the student requests. A sub-optimum approach to scheduling was used that is similar to M.I.T.'s Generalized Academic Simulation Program (GASP).^[1] This means that the generation of the schedule is derived from successive computer runs with the input of each successive run being revised through human analysis of the previous run.

The school used for this study was the Waynesville Missouri Senior High School. A unique characteristic of this institution is that out of a student body of 1,000 (1971-72 school year), about 52% were associated with the military at Fort Leonard Wood. The turnover rate of students averaged approximately 9 per week (1971-72) during the school year, and 35% over the summer vacation.

An attempt was made to create and implement a scheduling program for the Waynesville school system. Chapter II reviews the literature on scheduling which is pertinent to the specific problem and includes a discussion of traditional manual practice. Chapter III presents the sub-optimizing tool in the student scheduling problem that was developed and implemented for the Waynesville Senior High School students.

II. REVIEW OF THE LITERATURE

The public's willingness to support education is based largely on its view of how much return in pupil-teacher results it gets for each tax dollar. By making it possible to relieve the teacher of much of the clerical and non-instructional duties that consume time, effort, and money, data processing centers lower the instructional cost per-pupil. The old saying that data processing "dehumanizes" the individual no longer holds because teachers and guidance personnel have developed more as remedial clerks than as professionals using their skills to the student's advantage. The time has come for secondary schools to automate.

Goodlad, O'Toole, and Tyler^[2] have noted that the applications of computer technology in education have lagged far behind those in business, science, and government and the use of automated information systems is the exception rather than the rule. Most administrators still use very primitive techniques (paper, pencil and typewriter) for processing the wide variety of information used daily in the operation of schools. A large number of records and reports pertaining to students, teachers, budgets, and accounting transactions are produced at great expense; which could be effectively handled with computers.

Why, then, has not the education field made more use of data processing since there is surely a strong application for it? Until recently, very few educators were even aware of the scope and nature of computers and information system applications in education. The other factors involved in the slow growth of computer applications in education are the same psychological reasons found in the early application of data processing in business and industry. Educators do not relish the prospect of automation, because the unknown world of computer sciences tends to threaten their security.

Daniels and Yeates^[3], stated the possible reason why employees resist change and these ideas apply to the educational as well as the business application of computers,

- 1) "Fear of losing the job, of wage reduction, of inability to learn a new job, loss of prestige, of loss of interest in the job;"
- 2) "Suspicion of management's motives in making the change;"
- 3) "Resentment against personal attack, or a feeling that any change is a personal criticism of the way a man does his job;"
- 4) "Social upset caused by breaking up the group;"
- 5) "Ignorance, or fear of the unknown."

The loss of prestige would be the educator's resistance to change. A common question asked is, "Who ordered this change to take place?"

Educators have a high feeling for personal criticism regarding the way they are doing a job, but if it can be shown to them there is a "better" way of doing the job without any personal attack then the change over is smooth. The application of computers may mean that some of the office help will have to be reassigned to different groups. But this author found that fear of the unknown to be the most prevalent symptom in the application of this study.

Goodlad, O'Toole, and Tyler^[4] have noted that there is a serious lack of knowledge on the part of the educator about the power and potential of the computer systems for educational programs improvement. In addition, the results of successful computer installations in education have not been communicated to educators as successfully as in the business community.

Since the field of education is so large (second only to the Department of Defense on government funds spent), G. E. Anderson^[5] has listed one hundred possible applications for the use of data processing in a school district (see Appendix A). The present study deals with a small part of that list: The generation of a course master schedule and the sectioning of students to that master schedule.

A. HISTORICAL CONTEXT OF SCHEDULING IN SECONDARY EDUCATION

The settlers in America were familiar with the secondary schools of Europe and many had attended grammar schools. As a result, similar schools were soon founded in the New World. The first formal Latin Grammar School was introduced in the New World in Virginia in 1621 by private resources. The next year an Indian massacre brought it to an end. The first permanent school was established in Boston in 1635 as a free school or Latin school.

Because people wanted more education than at the grammar level, privately controlled academies were started during the early part of the 18th century. They reached a peak in number just before the Civil War. Since many people objected that education was privately controlled, public high schools began to grow. The new schools for the most part, however, were transformed academies maintained at public expense.

Brent and Kronenberg^[6] note that for a long time the academy met the liberal demands of those who were opposed to the Latin grammar school because of its formality. The academy liberalized secondary education, brought educational facilities within the reach of greater numbers, met the needs of many who desired practical training, prepared

students for college, and provided some secondary education for girls and some teacher education.

In the early days of this country (before 1850) the problem of evaluating how much education a person had if he started at one school and transferred to another did not arise because of the methods of transportation inhibited transfer of students to some extent. Later, as the United States population began to grow and the number of high school students increased, there had to be a way of handling this problem. In 1892-93 the Committee of Secondary School of the National Education Association stated that "every subject which is taught at all in a secondary school should be taught in the same way and to the same extent to every pupil so long as he pursues it, no matter what the destination of the pupil may be..." This statement is commonly referred to as the NEA Committee of Ten Report [7].

In 1919 the introduction of the Carnegie Unit helped implement the recommendation of the Committee of Ten. It usually represents the credit earned for pursuing one course meeting five times a week for an entire year. Technically, it requires a class attendance of 120 sixty-minute hours or the equivalent. If the school year is 180 days, the class period cannot be less than forty minutes to meet this requirement. The suggestion to allot a certain number of minutes per day for a course was the beginning of

the master course schedule.

Although the original purpose of the Carnegie Unit was to improve the status of retired professors for work performed by clarifying their teaching load requirements for retirement, it also clarified the distinction between secondary and higher education. For example, to be accepted into college, students were expected to have studied English in high school; there was now available a unit of measure of the amount of time in class devoted to such study. Also clarified was the educational equity that students can transfer from one high school to another. Since this method of exchange has persisted, it is of considerable concern to those that plan school programs and schedules.

In the ten year period (1908-18) the demands of industrialization and increased number of students brought about the National Education Association Seven Cardinal Principles (Appendix B)^[8]. These principles gave direction to the subject matter taught in secondary schools. During the first half of this century the Carnegie Unit was reinforced by the Smith-Hughes Act of 1917. The Smith-Hughes Act provided for the appropriation to the various states of the sum of \$1,660,000, to be increased gradually each year until 1926, when \$7,167,000 would be appropriated annually. These funds were to be used to assist the states

in paying the salaries of teachers, supervisors, and directors of vocational education in agriculture, trades, and industries.

The unit of measurement of how long a student had studied a subject matter provided the basis for scheduling the student's education. At the beginning of this century, basic reading, writing, and arithmetic were all the subject materials that were scheduled and school days were short in length. But as the industrial and technology advancements increased during the century, vocation courses were added to the curriculum, and the school administration found that the scheduling of courses became more difficult. Originally, all courses were scheduled manually. It was not until 1957 that computers were used in scheduling students.

First a few manual techniques for scheduling students will be discussed, and following this will be a description of some computer approaches. The common problems of generating course master schedules and sectioning students requests will be discussed under each technique.

B. MANUAL APPROACH TO SCHEDULING

1. Course Master Schedule Generation

Generally, the principal has the responsibility of the master schedule. Austin and Gividen^[9] have noted that

the most ignoble act of a departing principal is the destruction of all copies of old schedules of classes, teaching assignments, and time arrangements. The task of combining the various elements of resources and the need within a secondary school to produce a schedule of classes involving a minimum of conflict and a maximum of individualized student programming (sectioning) has been compared to playing between 5 to 15 games of chess simultaneously. In any case, the task is one requiring patience, practice, and a clear understanding of the purpose of the school.

In general, the main attributes that affect the generation of the course master schedule are: (1) student data, (2) teacher data, (3) time data, and (4) space data. The guidance office usually gathers the student data (Appendix C) for scheduling and the principal and other administration gather the time and space data and the teacher data (e.g., who will return for the next school year and what subjects will be taught). It is the wise principal who can resolve conflicts between various teachers' interests in subjects when he develops the course master schedule. A mathematics teacher might, for example, want to teach all algebra courses or have a mixed selection of algebra, trigonometry, and geometry. Although these problems are extremely difficult to resolve in the manual approach to scheduling, they would present little difficulty in an automated process in which levels of priorities can be

assigned to the courses a teacher wishes to teach.

In addition to problems related to student and teacher interest and classroom space, problems related to the time schedule often impose restrictions on schedules to a much greater extent than is generally realized. School bus schedules, for example, might limit the number of periods in a day and affect the opening and closing times for a school.

C. F. Lewis^[10], Holloway School, London has described the following sequence for how one should arrive at a workable timetable in England: "Start in January by talking to department heads, complete the discussion by Easter, do a preliminary sketch during the Easter holidays, complete the timetable in a month's work during the Summer term, and have the schedules printed by early July; the total time being about six months." He further states the manner in which to do your work: "Do your work yourself. The timetable job is for one person, not two. Unless one confines himself to fetching the tea and checking what the other has done, it would be difficult to see how the one person can be fully acquainted with what his colleague is doing."

Austin and Gividen^[11] have suggested team work on the schedule so that the state of the art can be refined and passed on. In preparing a course master schedule the following items might be needed: rulers, worksheets,

butchers paper, color pencils, cards, and etc. A place should be chosen where there is room and time so that the creation of the "ultimate" schedule can be accomplished without frequent interruptions. The interplay of the multi-factor task, as in the chess game, requires concentration in all but the simplest of schedules. The schedule board, whatever its form, is actually a device whereby section tickets and other cards representing periods in the day, room numbers, or the names of teachers may be moved about in relation to each other until a final "best" arrangement is realized and can be mimeographed and published.

In the generation of the course master schedule at a high school, the columns of the schedule matrix are the periods of the day with rows being the teacher's names. The section cards are then moved about in the matrix until a course master schedule is completed. Thus, the master schedule is one of the feasible solutions to a mathematical programming problem; this will be brought to our attention a little later.

Besides the individual tally of courses requested, there is one document that aids the scheduler of the course master schedule and that is a conflict matrix of the potential single courses that may conflict (see Appendix D). It is simply taking the student's requests and keeping a total sum of the courses that might get scheduled at the same

time on the student request for single section courses. As an example, the single section courses, chorus and band, could meet at the same time and out of the student population seven people have requested these two courses; thus a possible conflict in scheduling for seven students could arise.

When posting to the schedule board, a reasonable approach to the problem of priorities will increase the efficiency of this construction. A list of sample priorities might be:

- 1) Period commitments beyond the local school-teacher, facilities, or students who are shared by other schools.
- 2) Priorities based on justice to students - students in a higher class over a lower, e.g., a senior needs a class in order to graduate and gets preference over a junior.
- 3) Priorities relative to professional staff-balancing faculty load, conference periods, grade level, etc.
- 4) Priorities for group of students - slow learners, band, etc.
- 5) Priorities relative to the plant and equipment - no large class scheduled next to the cafeteria during the hour

prior to first lunch because of fumes and noise from the kitchen.

6. Priorities for a group of subjects - examination of the conflict sheet will disclose such a combination of subjects that are prerequisites for subjects.
7. Single and doubles (i.e. single - one section of course offered, doubles - two sections of course offered) - these have the most difficult demand placed on them which would give the possibility of adding or dropping a section.

The priorities listed above are not in any rated priority as they all contribute to the overall picture in scheduling.

The procedures listed below should represent the steps a high school scheduling process goes through.

- 1) Locate the single and doubles in periods which will allow for fewest conflicts among them.
- 2) Locate all other classes as to periods.
- 3) Adjust the section tickets - hold them in the same designated periods, but shift

them among teachers and rooms so that the best teacher program is completed and reasonable use of space and equipment is provided.

- 4) Check for inconsistency - e.g. will a large proportion of the eleventh grade be in physical education during the fifth period.
- 5) Test with a sample program for students.
- 6) Solicit teacher criticism - distribute a copy of the schedule, label it as "VERY TENTATIVE". Invite all teachers to examine the schedule and offer criticisms. These criticisms will most often be in regard to the individual teacher's assignment, but will frequently reveal weakness which can be readily adjusted.
- 7) Adjust the schedule - point out to the teacher that the schedule is based upon the established order of priorities.
- 8) Crystallize the schedule - after some time a final date is set in order that the actual programming may be carried out.
- 9) Student sectioning is accomplished - that is the test of the success of the master course schedule and student and class rolls are produced for opening day.

2. Sectioning of Student Requests

There are two ways students can be sectioned to their courses: guidance personnel may do the sectioning or the students may do their own sectioning on registration day, as at large institutions such as colleges and universities. At most secondary institutions the guidance personnel perform the task because it is felt that the majority of the student population is not yet mature enough to schedule themselves without some guidance.

Having placed the major responsibility for sectioning individual students in the hands of the guidance personnel, it becomes possible to group the work of the team of counselors about a central board or file box so that section balances may be assured and clerical work may be reduced to a minimum through the use of printed cards, student assistant clerks, carbon paper, and code abbreviations.

One such plan involves a "mail box" system. In the various sections of this box individual cards are provided for each student enrolled in each section of each subject, including all the doubles and singles. When a student is sectioned to an individual section, the counselor removes one card from the box for that section, writes the student's name on the card and enters that period and subject on the student's individual program card. At the end of the registration period, a basket into which the "mail box" cards have been placed will be emptied, the cards sorted

by sections, and this stack will be the official class list for the designated teacher. In turn, as a box is emptied of cards during the sectioning period, that section is closed, and unless a program change is effected and a card thus returned to a box, no more students will be placed in that section.

In a slight variation of this "mail box" approach (see Appendix E) the cards (section) can be slotted into the "mail box" instead of withdrawn as previously mentioned^[12]. Students are scheduled from pre-enrollment forms by using the course master schedule and by using a 3 x 5 card with the student's name, teacher, and section written on the card. The card is slotted in the period and teacher's name coordinates of the "mail box", and tabulated by number of cards for a designated slot, see Figure 1.

| Period | Teacher A | Teacher B | Teacher C | Teacher D |
|--------|-----------|-----------|-----------|-----------|
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

Figure 1. Manual "Mail Box" Sectioning of Students

Student help can be used to slot and tabulate the cards. A running count is kept on a copy of the course master schedule and copies of the count are distributed to each counselor so they have an accurate tally. Cards for students with conflicts are held until registration day, when the counselors can talk individually with students to resolve conflicts. Such a method of sectioning students to a schedule is defined as a counselor clinic by Austin and Gividen^[13]. The advantage of this approach is that problems in conflicts can often be reduced or eliminated by conferences between counselors.

C. MACHINE APPROACH TO STUDENT SECTIONING

1. Sectioning of Student Requests

At Purdue University in 1957, George Morgan and Gordon Sherman developed the first computer program for sectioning student requests. The sectioning of students was done with a manually prepared master course schedule and the time was one minute per student using a Datatron Computer^[14]. Later the program was submitted to the SHARE organization library as a demonstration program and then modified by IBM under the direction of Loren Bullock at the New England College Computing Center, for production use at the University of Massachusetts^[15].

Since the advent of Purdue's student sectioning computer program, several other institutions or educational groups have produced similar programs. Examples are: IBM's CLASS (Class Load and Student Scheduling), California Richmond Union High School's computer program named SOCRATES (Scheduling of Classes Realized Automatically Through Effortless Systemization), and System Development Corporation's MASTER (Matching Available Student Time to Educational Resources).

2. Course Master Scheduling Generation

One of the earliest works in the area of class master construction was by C. C. Gotlieb, 1962^[16]. Due to the complexity of the university master schedule, Gotlieb directed his efforts to a rigorous mathematical development for determining time assignment for teachers to classes in high schools. Gotlieb constructed small timetables by iteration involving boolean matrices. It was assumed that weekly requirements could be sub-divided into five sets of daily requirements. This assumption set the pattern structure for each day of the week. To insure full utilization of certain critical facilities (e.g. lunch, science class, and gym) parts of the timetable could be filled in advance. Gotlieb's construction, therefore, was one of completing a partially filled timetable.

Gotlieb's method of timetable construction takes the form of a requirements matrix \tilde{R} with (integer) elements \tilde{r}_{ij} giving the number of times per day teacher t_i must meet with class c_j . The timetable is developed in the form of a 3-dimensional scheduling array \tilde{S}^k with elements \tilde{s}_{ij}^k which are 1 or 0 according to whether t_i does, or does not meet c_j at hour h_k . Therefore, we have

$$\sum_k \tilde{s}_{ij}^k = \tilde{r}_{ij} ; \sum_i \tilde{s}_{ij}^k \leq 1 ; \sum_j \tilde{s}_{ij}^k \leq 1 \quad (1)$$

Furthermore, it is assumed that some preassignments are given in the form of an initial assignment array S^k , and we must have $\tilde{s}_{ij}^k \geq s_{ij}^k$. A residual requirements matrix R is defined by

$$R = \tilde{R} - \sum_k S^k. \quad (2)$$

Because \tilde{R} represents an assignment problem, it is desirable to complete \tilde{R} by adding dummy teachers and classes as necessary in order that the row and column sums equal the number of periods in a day. To apply the scheduling method, an availability array A^k is constructed from S_1^k such that $a_{ij}^k = 1$ if t_i is free to meet with c_j at h_k , and 0 otherwise. For S^k to exist, certain inequality relations have to hold among the elements of A^k and R . Whenever an equality arises in one of these relations, the effect is that some elements of the availability array may have to be changed from a 1 to a 0.

a_{ij}^k

| Class j | Hour k | | | | | | | | | r_{ij} |
|-----------|------------------|------------------|------------------|------------------|------------------|---|------------------|------------------|------------------|----------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | |
| 1 | a \emptyset | d \emptyset | a \emptyset | b \emptyset | b \emptyset | 1 | 0 | c \emptyset | 0 | 1 |
| 2 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 3 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 4 | 0 | 0 | a \emptyset | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 5 | a \emptyset | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 6 | a \emptyset | 0 | 0 | b \emptyset | 0 | 0 | 1 1 0 | | | 1 |
| 7 | 0 | 0 | a \emptyset | 0 | b \emptyset | 0 | 1 0 1 | | | 1 |
| 8 | a \emptyset | 0 | 0 | b \emptyset | 0 | 0 | 0 1 1 | | | 1 |
| 9 | a \emptyset | 1 | a \emptyset | b \emptyset | 0 | 0 | c \emptyset | c \emptyset | c \emptyset | 1 |

Figure 2. Section of Availability Array for t_1

Consider, for example, a section of the availability array for t_i shown in Figure 2. The element in row 8 and column 4 is 1 because t_1 and c_3 are both free at h_4 . In addition, note that t_1 must meet both c_2 and c_3 and the only hours when this is possible are h_1 and h_3 (shaded left). A combination of a teacher and a set of classes where the number of available hours is just sufficient to satisfy the residual requirements is called "tight", and it follows that the available hours must be reserved for that set of classes. That is, since

$$\sum_k a_{12}^k \vee a_{13}^k = r_{12} + r_{13} \quad (3)$$

we must make $a_{1j}^1 = a_{1j}^3 = 0$ for all $j \neq 2, 3$. This results in all the elements marked "a" being changed from 1 to 0. One now finds that C_4 and C_5 make a tight combination with t_1 and all the elements marked "b" must now be changed from 1 to 0. Next C_6 , C_7 , and C_8 (shaded right) form a tight combination giving rise to further changes in the elements marked "c"; finally h_2 must be reserved for C_9 so that A_{11}^2 is changed from 1 to 0 ("d").

The completed availability array can be displayed as a set of 2-dimension tables (Figure 2), one for each teacher. Alternatively, we can display it as a set of 2-dimensional tables, for each hour. Each time as availability is reduced in one display because of a tight

combination, it must also be reduced in the other two, since the same information is represented. To apply the proposed method of testing whether a timetable is possible in the light of the preassignments, the three displays of the availabilities are reduced iteratively until

- 1) no further reduction of availabilities are possible, in which case the availabilities are said to be reduced, or
- 2) a set of rows is found such that the number of elements in the logical union of the rows is less than the sum of the corresponding requirements.

The tight combinations of teachers, classes, and hours are resolved systematically. At the end of a set of final availability vectors which produced no new tight combinations are obtained. The argument shows that the existence of the final availability vector is necessary for a feasible solution. The sufficiency of the final availability vector is conjectured, but it is not known. If the conjectured sufficiency is correct, then the decomposition which is made into a tight combinations will be a feasible set of matrices. Thus, a schedule is really constructed from the final availabilities. It is stated that the

algorithm described has been tested by hand calculations and by limited computer runs on small timetables and it yields valid results^[17].

G. A. Sherman^[18] formulated the scheduling problem into a problem of finding an optimal partition of a finite set of points with the criterion being the minimization of the expected number of conflicts for a randomly drawn student. The problem of selecting the "best" set function amounts to finding an optimal set partition of a finite set. In an approach related to Sherman's formulated problem, P. J. Kiviat^[19] procedure is one of aggregation of student course demand and creating a course master schedule to which individual students are assigned. The course master schedule is a probabilistic reflection of individual student demand that assigns individual students by means of an iterative assignment algorithm which converges to a "best" overall student assignment.

John Lions^[20] showed that the application of Konig's Theorem^[21] to the problem of matrix reduction was needed in Gotlieb's method for timetable generation. An examination of the 2-dimension array reveals that the array has two stages.

Stage 1. The existence of at least one possible schedule is determined

(Feasibility Test - known as part of the assignment problem^[22]).

Stage 2. Any non zero element which does not belong to some possible schedule is changed to zero (Matrix Reduction).

In the context of the timetable problem, Stage 2 is an important one because of the dimension of the matrix size for computer application. The need for a more efficient algorithm was demonstrated by Lions, because 2^n operations, where n is the number of classes, are needed making this timetable method impractical once n exceeds 20.

Lions used an EXPAND program modeled after Marshall Hall's Algorithm for District Representatives^[23] for efficient testing of the feasibility of an element. If the elements tested were tagged as being feasible and not tested again, then this alone on the average reduced the amount of time that EXPAND was used by half.

Master schedule building programs have been developed to go a step further in the scheduling operation. A heuristic simulation approach to master schedule construction was developed at M.I.T. under the direction of R. E. Holz in 1962. The program was called GASP (General Academic Simulation Program). The objective of the GASP program was to provide a more flexible class arrangement

for students at M.I.T. As a result, this initial work has been centered on the college scheduling problems rather than those found in the public schools.

GASP^[24] is uniquely different in several respects from either manual or computer-based class loading routines in that it does not require that the principal "construct" a master schedule of courses to be offered. Significant departures from traditional methods are the following:

- 1) GASP has the capability to construct a master schedule of classes based on certain data furnished as input material.
- 2) The program balances class loads through an iterative process.
- 3) Any time period configuration of instructional modules can be produced. The modular pattern then provides for virtually any desired arrangement of instructional time elements; i.e., almost complete flexibility of teaching arrangements.

In acquiring these attributes, GASP is now in its third generation form and must be used on the IBM 360 computer system.

GASP is a fantastically complex program both from the input (raw data coding) stage and from the actual programming of data via feedback information (technical phase).

One of the most important aspects of GASP is that it allows the administration to simulate a number of master schedules in order to check curriculum planning and select the one best suited to their needs. In no way, however, does it take away the decision-making prerogative of the administration.

The frustration in the application of this program system is that a minimum of 9 months of "lead time" is required. GASP is not a "one-time through" program! Thus, one must conclude that if any other routine (SOCRATES, SSSS, MASTER) will satisfy the requirements of the curriculum, it should be used rather than GASP.

One may consider GASP as a four-dimensional problem involving:

- 1) Time - available within which course may be taught.
- 2) Space - in which course may be taught; e.g. rooms.
- 3) Student's course requests.
- 4) Staff - available to teach the course (by level/teams) in the curriculum.

GASP usually proceeds in two phases: (1) room and staff assignments are made, and (2) student's course requests

become part of the program.

The idea of GASP can be thought of as a tool to aid the production of schedules and will not take over the job of the scheduling, but will be a good tool to help the scheduler do his job better. Important scheduling decisions will remain with the scheduler, e.g. resources allocation.

Holt's GASP program uses a sample of the student's requests as input for the development of the master schedule. In the program, staff, space, and students are assigned a weight factor by the school authorities. These weights are used in the selection of the time for the course to meet. Student request cards are arranged in the order of their preference for the sequence of assignment.

The computer converts the input into binary vectors or matrices indicating availability of rooms, teachers, students, and times. A student also has a vector generated which indicates the particular courses he requests. Room and teacher vectors or matrices are also generated to indicate courses to which they can be assigned.

The GASP program constructs the entire master schedule before making student assignments. After the matrices have been formed, the sequential assignment of classes to

time begins. The time pattern of a particular course is satisfied by manipulation of staff, space, and sample student matrices. Selection of the time to which the section is scheduled is made on the basis of the weights applied to the resources input. Instructors, room and sample students are then assigned to that class, and then their availability matrices are up-dated. The next section offering is scheduled in the same manner until the master schedule is completed, ending phase one. The second phase starts with the student assignment in sequence; GASP orders a student's course request with the course having the least number of sections being arranged first. A conflict matrix is generated to indicate the conflicting course sections. A section is selected randomly from the field of possible choices. The next course requested is considered to determine the time choices that do not conflict with the previously assigned sections. The time choice can be random or on the basis of a weight applied according to the proportion of remaining seats in each possible sections.

Several workable schedules are produced after a set number of trials, and the "best" schedule is selected according to a measure of relative goodness of each one. The number of runs needed varies between 8 to 10.

The approach to scheduling through GASP may be delineated by stating that an exact solution is forfeited in return for realizing the ability to consider all four of the vectors: student course requests, times, staff, and space for a large secondary school which requires modular time pattern (the periods during the day are of different time lengths), because of a trade-off between the parameters. An easy trap for the school administrator is to attempt to create a conflict-free schedule. In a modular GASP arrangement this can be achieved but in the ultimate sense it will probably be a rather unsatisfactory timetable in the computer time needed and the sacrifice of other variables such as several small modules (15 minutes each) being inserted between large instructional modules causing the back-to-back arrangement of the large instructional modules to be sacrificed.

The Stanford School Scheduling System^[25] (SSSS), of 1961, was based on the R. V. Oakford algorithm solution, which allowed for more pre-selection decision-making by using the computer system. Scheduling effectiveness, therefore, was a function of the system algorithms. It was assumed that the components of the scheduling problem could be accurately and sequentially defined in sufficient detail to program the computer. In contrast to GASP, the SSSS

program did flexible scheduling in high schools. The terms "flexible" or "variable" refers to class meetings of different lengths and frequency.

The input for each course section in the SSSS includes the time pattern which indicates if the section meets for a multiple time period, the number of days per week, or the specific days of the week. A course is defined by structure type, which presets the number of consecutive time periods a section meets. Section input arrangement is in the order of desired scheduling as the program schedules each class sequentially.

Teachers are designated for each subject they are qualified to instruct. Teachers can have preferences for certain sections or can be assigned to instruct certain sections. The school administration indicates first and second choices of teachers for courses. Special features of the rooms are specified as input. Rooms may be restricted to a specific course or group of courses.

The SSSS procedure selects a teacher and room combination for a particular course, and then finds the students who are free at the same time period chosen. The first step is to find the available times of the first choice teacher for a particular course. Teachers are assigned priority on the course subjects they are qualified to teach. If the first teacher does not have

enough free time periods to satisfy the course requirements, then second choice teacher's availability vector is scanned. Once a teacher is found for the course, which has been ranked by enrollment, largest to smallest, then a room is located with the same free time as the teacher's time. If a specific room had been designated, then its available vector is the only one checked.

Now a search of student course vectors (requests) locates the students who request the course. Their free time vector (time period not committed to a course) indicates the number of available students for that course and time period. If this number does not exceed the capacity of the room size then the teacher, room, and students are assigned to that section time period. This continues with each course in turn.

Student sectioning conflicts may be resolved by an interchanging of students between sections of a course. If it is a single section course and all students requesting the course are not free, the free students are scheduled, with output messages indicating the section and the unscheduled students. It may be possible to resolve a conflict when a course has multi-sections. The conflict resolution techniques involve boolean operations of binary vectors.

A specific output from the SSSS program is a tabulation of time periods, room, and teacher vector formats which describe the student-course request, student free time, and teacher-room availability.

The output of GASP gives a completed master schedule, with the number of students assigned to the course section and the remaining seats. A summary report on the resource utilization can also be included. Listings pertaining to course sections and time patterns for staff are printed out.

These two master scheduling generating programs, GASP and SSSS, do not guarantee an optimal schedule (one which minimized the number of conflicts), nevertheless, they are a value to school administrators in construction of more efficient master schedules.

There is one other timetable program that became operational in 1967, the OSSP (Ontario School Scheduling Program)^[26]. This program was designed for block scheduling (fixed time periods during the week), but can do some modular scheduling (periods having different times during the week). Like the GASP program it is heavily dependent upon input requirement forms, about 300 different ones (resource allocation) depending on the size and complexity of the school. The requirement forms list all teachers (can be assigned a priority rating), class, and subject.

A second set of forms may be optionally included. These are the room preference forms. Each teacher needing a room assignment is represented by a room preference form. One form might show up to ten room preferences per teacher. The rooms are assigned after the teacher-class assignments have been made.

The work required by a school administrator in filling out these forms is approximately equal to setting up the procedure to do the class master schedule by hand with the aid of the assignment board.

The output from the OSSP is in three stages with the first stage being optional, which includes several sets of tables useful in correcting the input by manual means. The second stage of the output contains a display of the conditions existing as each special requirement is handled, a special requirement being anything but one teacher, one class, one subject combination with no time restriction. This stage also contains the teacher, class and room timetable by day. A final stage produces a timetable by week.

The OSSP program is based on a method described by Gotlieb (1962) and Csima and Gotlieb (1964) but with the "tight search" replaced by an equivalent but more efficient algorithm based on the Hungarian Method introduced in this context by Lions^[27]. Lions' testing of the OSSP showed that the algorithm was workable on application to different

schools and their needs. The major disadvantage of this program is the number of computer runs needed to reach a satisfactory point. The problem is not the computer time but the time which elapses between the initial data submission and the creation of a final timetable. Also, any school timetable is necessarily a compromise. In particular, it is usually impossible to incorporate in a single timetable all the requirements a school principal could conceivably wish for.

It is noted that from 1967-to date, the dearth of literature published on the progress of machine approach to student sectioning may be explained by the proprietary rights of machine manufacturers.

III. A BLOCK SCHEDULER FOR WAYNESVILLE HIGH SCHOOL

A. DESCRIPTION OF WAYNESVILLE HIGH SCHOOL CHARACTERISTICS

In the Spring of 1971, the Waynesville-Fort Leonard Wood School District had access to an IBM 1130 System; at that time they were interested in a feasibility study on how they could use the computer system to aid them in their high school in making student schedules, class rosters, grade reports, etc.

The most important requirement was to have the computer program balance out the sections of a course when sectioning students. In the past, the hand sectioning of the students had skewed the balancing of the student count in the different sections because of the highly transient nature of the military related student body. The computer is an excellent tool to do the sectioning of students because of its speed. The closer to the opening day of school that the sectioning of students to their classes could be done, the fewer number of drops and adds would have to be handled by manual clerical means. This does not mean that the clerical chores would be eliminated; only that there would be a fewer number to handle.

The principal also wanted full control of the decisions on the course master scheduling via computer because of the complexity of scheduling resources (teacher, room,

and time) inherent in the design of a master schedule at this institution. As an example, some of the faculty were wives of military personnel and might be transferred out of the school district during the summer and their replacements could not be expected to have identical qualifications. Also, the class size may determine the room location of the class or the size of the physical room may be the limiting factor to the class size, i.e.; welding or auto body repair class.

B. DESCRIPTION OF SCHEDULING

The pre-registration was held in May after a course description was generated in April (Appendix F). The course sections were not available at that time but the course description was available. By having the course titles in alphabetical order, the courses then could be numbered consecutively, (Figure 4). With that accomplished in early May the administration was able to start planning for the coming school year. The determination of the number of sections needed for the coming school year could be accomplished in one of two ways; either by hand or using the computer.

If the number of sections was to be determined by hand, a hand tally method would be done with paper, pencil, and a lot of patience. When the total number of students

for a course was known then that total was divided by the number of students that was suggested for the size of the class for a course (e.g. 30 students). The appropriate class size of a subject was suggested by the North Central Association of Secondary Schools and Colleges.

The other method was to use the computer in recommending the number of sections needed. This was accomplished by using the master course file, which was the numerical equivalent of the student course requests, and by using the suggested class size of the North Central Association of Secondary Schools and Colleges or the principal's suggestion. (The latter is preferred under the guide lines of NCASSC.)

To arrive at the number of sections for a course the computer tallied the number of requests for each course and divided the total count by the suggested class size. An added bonus in using the computer approach was that the conflict matrix of the single section courses could be generated at that time. The conflict matrix showed the number of students that had two or possibly more single sections requested (see Appendix E). The principal then used this information for scheduling potential conflicts at different times. The conflict matrix was not used in determining room assignments.

Since scheduling normally takes place (in the school calender) before the end of the semester when grade cards are prepared, it allows the principal and other administrators to discuss and arrive at some conclusions of the information provided by the computer while the computer is used to help close (i.e. grades, transcripts labels, etc.) the school calender.

Timing is important because the high school administrators leave soon for their summer vacation and should be in on the decision making that will affect their work performance during the registration just before the starting of a new school calender. This point can not be emphasized enough since they will be the ones working with the computerized results. The most prevalent complaint from any administrator was:

"I was not involved in that decision making policy. Why?"

The change over to a computer aided decision making process is a benefit in that the person in charge has only the facts and is not influence by other human personalities involved in the decision making process. But since the computer is an aid and not the decision maker, it is imperative to have all the persons involved in the procedure to

understand what has to take place in a computer aided decision making system.

A final item that should be mentioned is the sequence number that is used in alphabetizing the students' names. Although this could be performed in several ways, the method used in Waynesville is to code the students' names (in alphabetical sequence) and course requests in an IBM card and let the computer program punch (numbering) the student sequence number into the card. This is then the number the student will use internally within the computerized system until graduation when that number will be available for reassignment. In order to keep the student records accurate during his or her school time, the social security number is collected or applied for (the number is sent to the high school and then distributed to the student) during the enrollment of the student. This social security number is then the student's identification number (ID) for his records while in school and after graduation.

C. DETAILS OF OPERATION OF THE WAYNESVILLE HIGH SCHOOL COMPUTER PROGRAM

In order for the principal to use the computer as a decision making tool when he generates the course master schedule, the data files for the scheduling of students have to be flexible so that the decision of the principal

on the course master schedule can be implemented quickly to check the results of his decisions in the generation of the course master schedule.

The technique must include the number of sections a course is to have in the course master file, the credit value of the course, and the time periods of the day the section meets. Consider, for example, the arrangement in Figure 3. American History needs 12 sections as determined by the course tally of requests and the number of students the class size will allow. The time vector for this course is 213222, if the individual numerical values of the vector are added, the sum is 12 for the number of sections for the course. This means the number of sections for a period can be changed by changing the individual numerical value in the string. As an example, the vector string 213222 means that the third period, third position of the string can be changed to 4 sections at that third period by decreasing the fourth position to one section and the resultant vector string becomes 214122^[28]. The course credit is 0.5 for the semester, meaning the course meets for one hour time period per day.

The credit value of the course is also used in creation of the sections. If a course such as Air Conditioning has a credit value of 1.5, it means that it is a half day course (3 periods out of a possible 6). The course should have this vector time string, 222111, depending on the school

| NUMBER OF SECTIONS | COURSE NAME | CREDIT | SEATS | COURSE MASTER | TIME |
|-----------------------|------------------------|--------|-------|------------------|--------|
| 2 | ADVANCE ALGEBRA & TRIG | 0.5 | 32 | 1 | 010001 |
| 1 | ADVANCE BIOLOGY | 0.5 | 32 | 2 | 000100 |
| 1 | AIR CONDITION I | 1.5 | 12 | 3 | 111000 |
| 1 | AIR CONDITION II | 1.5 | 12 | 4 | 000111 |
| 4 | ALGEBRA | 0.5 | 32 | 5 | 111100 |
| 12 | AMERICAN HISTORY | 0.5 | 32 | 6 | 213222 |

Figure 3. Course Master List (CMSTR)

| Student's Number | Student's Name | Request | | | | | | | Option | |
|---------------------|------------------|---------|----|----|----|----|----|----|--------|--|
| 100 | ABBOTT, TED | 84 | 92 | 48 | 90 | 64 | 6 | 26 | | |
| 300 | ABSHIRE, REBECCA | 66 | 15 | 48 | 90 | 64 | 6 | 81 | 56 | |
| 400 | ADAMS, JENNIFER | 92 | 76 | 48 | 73 | 63 | 96 | 56 | | |
| 500 | ADAMS, KATHERINE | 57 | 81 | 15 | 47 | 64 | 6 | 59 | | |
| 600 | ADKINS, JANET | 57 | 80 | 78 | 33 | 26 | 89 | 36 | 83 | |
| 700 | ADKINSON, DONNA | 56 | 73 | 47 | 35 | 63 | 96 | 29 | 48 | |
| 900 | AGEE, ELBERT | 27 | 84 | 90 | 64 | 6 | 0 | 14 | 34 | |
| 1000 | ALEXANDER, MIKE | 49 | 48 | 76 | 72 | 63 | 94 | 47 | 95 | |

Figure 4. Request File (REQST)

resources, indicating that there are two sections meeting the first, second, and third periods of the day; with the third section meeting the fourth, fifth, and sixth periods of the day. This type of file organization allows a great flexibility during the planning stages of the course master schedule, by allowing the program to set up the sections for the scheduling of students. The sections file information is kept internally on disk file and eliminates the need of course section file cards. When the principal is generating the course master schedule, he is allowed a greater degree of ease of testing the master schedule than would be possible by using a card generated section file.

When the final schedule run is ready to be produced, the program (via a logic switch) punches out the section course cards with all the information needed except the teacher's name and room number. These are keypunched into the section cards and loaded back to disk so that the student's class schedule card can be printed from the student schedule file. For a system flowchart of the scheduling system, see Figure 5.

In connection with the course master file (Figure 3) a pointing vector (shown in Figure 6) is generated during the breakdown of the course master file into the individual sections for the course. This results in the portion of the program for sectioning (scheduling) of student requests having access to all of the sections of a course. As an

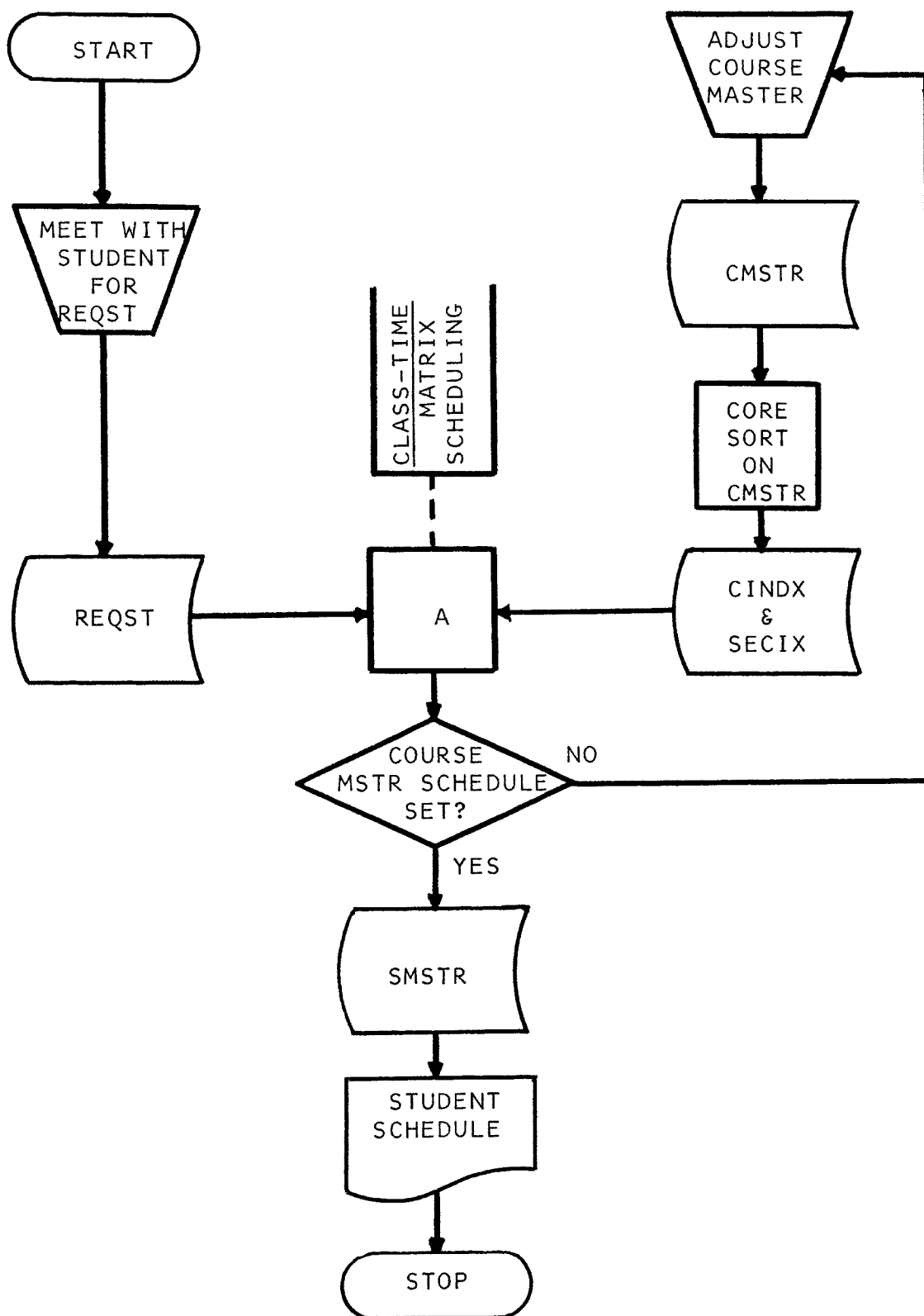


Figure 5. Student Sectioning System Flowchart

| | | |
|------------|---|----|
| CINDX (1) | = | 1 |
| CINDX (2) | = | 3 |
| CINDX (3) | = | 4 |
| CINDX (4) | = | 5 |
| CINDX (5) | = | 6 |
| CINDX (6) | = | 10 |
| CINDX (7) | = | 22 |
| CINDX (8) | = | 23 |
| CINDX (9) | = | 25 |
| CINDX (10) | = | 26 |

Figure 6. Course Master to Section File Index (CINDX)

example, see Figure 6, where $CINDEX(1) = 1$ (Course INDEX) means that a request was made for Advanced Algebra class having two sections because Advanced Biology ($CINDEX(2) = 3$) is pointed to the third section in the file.

The next logical step is to catalog the student requests in some kind of course section priority^[29]. There are several priorities that can be used in sectioning students to courses. The priority used here for sectioning is that if there is only one section to a course, it will be the first request of the student to be scheduled; because a multiple section course has a greater likelihood of not causing a conflict. Therefore, the multiple section courses will be the last courses to be scheduled, instead of being the first. The student's request file is established on disk under the name of REQST (REQUEST), then is sorted in the minimum section priority by using the CMSTR file.

As seen in Figure 7, the program arranges the section priority of a student's request in the following manner. An array is set up in core (6 x 13) with the 6 rows of the matrix representing the maximum number of courses that can be requested and the 13 columns representing the maximum number of sections offered for a course. The algorithm takes the first request (represented by row one) and using the course master file (CMSTR) retrieves from the file the

| | | | | | | | | | | | | | |
|--------|--|----|---|---------------------------------|----|----|----|----|----|----|---|---|----|
| | | | | Requests before sorting | 58 | 96 | 72 | 13 | 48 | 35 | | | |
| | | 0 | 0 | 58 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 96 |
| Array | | 0 | 0 | 0 | 0 | 72 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 x 13 | | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | 0 | 0 | 0 | 48 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 35 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | | | | | | | |
| | | | | Priority of array after sorting | 13 | 58 | 48 | 72 | 35 | 96 | | | |

Figure 7. Priority Sorting Array of Student's Requests

number of sections (represented by the columns) for a course and places the student's request number in that row and column of the matrix. After the sixth request has been processed, a FORTRAN DO statement gathers up the priorities by going down the columns of the matrix from left to right and sorting the numerical values of the requests under the file name of REQST. Also, the order of the request card file is such that the Seniors, Juniors, and Sophomores are sectioning in this order because a Senior has less time to get a course than a Junior. A Senior, therefore, is scheduled for any given course before a Junior^[30].

The scheduling (sectioning) algorithm flowchart is shown in Figure 8. The algorithm performs in this manner. If the course is not closed, the student's first course request is scheduled. If it is closed, then an option course is scheduled in the place of the first course request. A code is generated to the student sequence number so that a message is printed out with the student's schedule stating that an option course is used and that it could be handled with the counselor at enrollment time, if necessary.

For the initial run of the course master schedule, the previous year's master schedule is used with the modification added that reflects the adding or dropping of courses offered. The student course tally also shown when

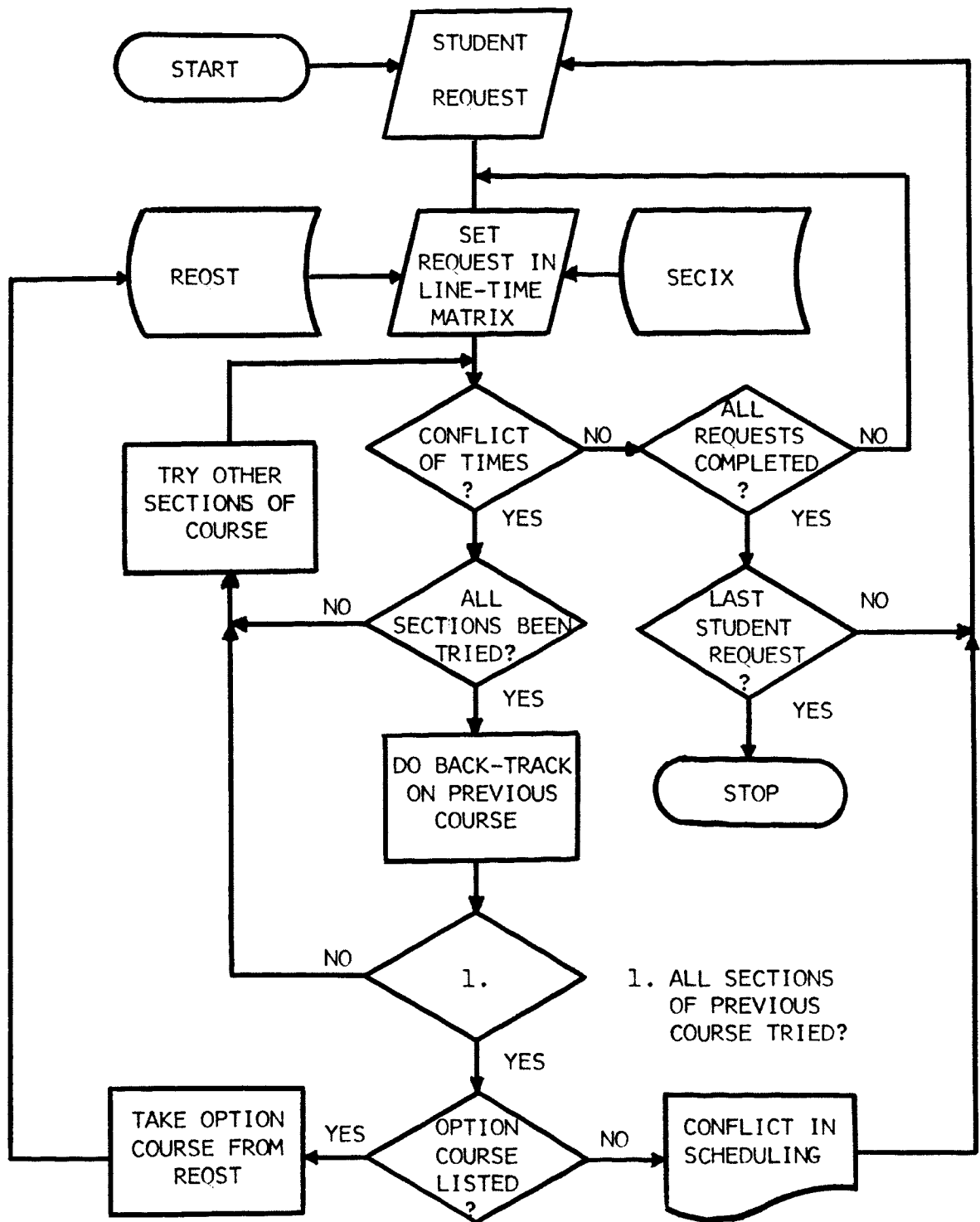


Figure 8. Student Sectioning Algorithm Flowchart

enough students request a course to justify its existence (given that there are enough funds to hire a teacher for the course). At the same time of this initial run the principal uses the previously mentioned assignment board in a parallel operation shown in the flowchart in Figure 5. Since the algorithm is not using the extreme points of a feasible region in order to optimize an objective function, the term sub-optimum used here does not mean that this is a mathematical program problem. Optimality is really feasibility at this stage and the objective function which is used is probably a rather nebulous idea in the principal's mind.

As shown in Figure 9, a 6 x 6 conflict matrix is set up in core to test the feasibility of the student sectioning process. Since the time pattern is represented in binary form (see Figure 10) it can be tested in two ways. Martin Faulkner^[31] uses a logical OR and logical AND to test the time bits for a conflict in sectioning students at the University of Washington, (Appendix G). The time bits for the first course assigned to the student are combined with the student's time vector with a logical OR. The time bits for the next request are AND'ed to the student's time vector and tested. If the result of the AND is a word (or words) containing all zeros, there is no conflict and a logical OR is performed to add this request to the student's time vector. If, as a result of the AND,

Course Time Patterns

41 000010
 36 001000
 33 010100
 20 001100
 97 100110
 50 010011

Student's Course Request Numbers

41 36 33 20 97 50

If the fourth period of Dramatics had been scheduled then a conflict would exist between Dramatics and Chemistry the fourth period and Economics and Chemistry the third period. The conflict can be eliminated by back-tracking to the scheduling of Dramatics to the second period and scheduling Chemistry for the fourth period, thus generating a successful schedule.

| COURSE TITLE | PERIOD | | | | | |
|-------------------|--------|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 |
| FAMILY RELATIONS | 0 | 0 | 0 | 0 | 1 | 0 |
| ECONOMICS | 0 | 0 | 1 | 0 | 0 | 0 |
| DRAMATICS | 0 | 1 | 0 | 1 | 0 | 0 |
| CHEMISTRY | 0 | 0 | 1 | 1 | 0 | 0 |
| WRITING & GRAMMAR | 1 | 0 | 0 | 1 | 1 | 0 |
| GEOMETRY | 0 | 1 | 0 | 0 | 1 | 1 |

Figure 9. Class-Time Matrix Scheduling

the word contains any non-zero bits, a conflict is indicated and appropriate action must be taken.

In this study, the investigator chose the addition of the columns of the matrix (6 x 6) because the IBM 1130 FORTRAN has no logical statements. If the sum of any column of the matrix exceeds one, then a time conflict occurs in the student's schedule and another section of the course will be tried. Of prime concern in automated sectioning of students is the ability to fill all sections evenly^[32]. This can be accomplished by making a sort of the present count in all the sections of a course requested. The sort is in ascending order so that the first section to be tried will have the lowest class count, if this section can not be scheduled because of a time conflict, then the next section is tried and the process continues in this manner until all sections that are not closed have been tried. The sections that are closed, because of full capacity, are not included in the count sort. If this fails, then a back-tracking procedure takes place. The back-tracking method employed in this section algorithm eliminates the conflicts in sectioning caused by the sorting method of section balancing. If this was not included in the algorithm, the number of conflicts of student schedules would have increased by 175%. The only true conflicts then are the ones that have two or more single sections meeting

at the same time since all of the sections are closed that would give a non-conflict schedule.

The back-tracking operation takes place when a conflict in the time matrix occurs and all the sections of the present course being scheduled have been tried, as shown in Figure 8. The back-tracking is now tried by taking the previous course request and trying the combination of the previous course section, eliminate the one that causes the conflict, with the present course request section. The prime concern at this point is to schedule the student and not worry about a balance in section population. Each time a course is slotted into the time-conflict matrix; a test is performed on the matrix for a time conflict.

If the testing of the section and the back-tracking do not eliminate the conflict then an option course is tried in the schedule in order to generate a successful schedule for the student. If this fails then the partially completed schedule is saved and printed out stating there is a course conflict in the sectioning process, allowing the counselors to intervene at this point.

This sectioning of students continues until the last student request has been processed and a section list is printed. This document (see Figure 10) shows the class size, the number scheduled, number of overflow, and class period time. The section file document is used to point out to the

| Section Number | Class Size | Number Scheduled | Number of Over-Flow | Class Time | Course Name |
|-------------------|---------------|---------------------|------------------------|---------------|---------------|
| 137 | 32 | 24 | 0 | 000010 | LANG. ART III |
| 138 | 32 | 24 | 0 | 000010 | LANG. ART III |
| 139 | 32 | 23 | 0 | 000001 | LANG. ART III |
| 140 | 32 | 23 | 0 | 000001 | LANG. ART III |
| 141 | 32 | 24 | 0 | 010000 | LIT. ENGLAND |
| 142 | 32 | 32 | 21 | 100000 | MERCH. (JR) |
| 143 | 32 | 5 | 0 | 010000 | MERCH. (SR) |
| 144 | 25 | 20 | 0 | 000100 | METALS |

Figure 10. Section File (Sect)

person making the course master schedule where conflicts are and what sections of a multiple section course can be increased or decreased during the periods of the day in order to minimize student section conflicts. This procedure is part of the flowchart labeled A in Figure 5, and is repeated until the course master schedule is set for the final run by the principal.

IV. SUMMARY

A practical algorithm for the semi-automatic student sectioning and master course schedule generator was developed and implemented for the Waynesville Senior High School. The main unique points in the algorithm are the back-tracking procedures used to eliminate conflicts in the student schedule (discussed below) and the ease of encoding the addition or deletion of courses and/or sections in the course master file (schedule) that the principal wishes to test for minimum student conflicts. To alleviate the burdensome problem of changing the card file (a card for every section and time a course meets) the method employed codes the course master file cards with the number of sections and a time string that represents the meeting of the sections. The file is generated on disk and thus eliminated the need for a large card file (section cards).

The biggest advantage of the Waynesville Senior High School student scheduling algorithm is the ability to do a back-tracking operation when a conflict in the student schedule is encountered. As far as this author knows there has not been any published material on a back-tracking procedure in any of today's operational computerized scheduling systems. This dearth of literature may come from the proprietary rights of the computer manufacturers and soft ware companies while all of the major computer manufacturers (IBM, NCR, Honeywell, ect.) offer, to

some degree, student enrollment packages for the users of their computer systems, the back-tracking technique described in the present investigation may be entirely new to this application.

The application of the back-tracking procedure to the scheduling of the students at Waynesville Senior High School eliminated 64% (73 out of 114) of the conflicts in student schedules (12% of the student population) with the result being that 5% of the student population had conflicts that had to be resolved by manual means. The percent of unresolved conflicts is defined as being a function of the "best" course master schedule that is available. An example of a perfect course master schedule would be 0% of conflicts.

The cost of the application of this system to the Waynesville High School was approximately sixty four cents (64¢) per student (includes computer time, administrative, and supply expenditures). This figure is based on a student population of 1,000 and using the computer for a total of five runs (+ or - runs) for class master course "tinkering". This contrasts sharply with the University of Missouri Department of Education's charge of one dollar (\$1.00) per head for scheduling students in Missouri high schools and approximately \$1.25 per head, with a consultant fee of several hundred dollars charged by commercial computer

service companies. Even these figures are small, however, when compared to the cost of the manual operation of \$2.57 per head (Appendix H) during the 1970 scheduling operation at Waynesville Senior High.

BIBLIOGRAPHY

1. Richardson, Duane: "Pupil Scheduling by Computer",
National Data Processing: New Dimensions and Prospects,
R. A. Kaimann (ed), Boston: Houghton Mifflin, 1967,
pp. 190-193.
2. Goodlad, John I., O'Toole, John F. Jr., and Tyler,
Louise L.: "Computer and Information System in
Education", New York: Harcourt, Brace and World, 1966,
pp. 28.
3. Daniels, A. and Yeates, D.: "System Analysis", Palo
Alto: Science Research Associates, Inc., 1971,
pp. 1-11
4. Goodlad, J. I., O'Toole, J. F. Jr., and Tyler, L.L.:
"Computer and Information System in Education", New
York: Harcourt, Brace and World, 1966, pp. 29.
5. Anderson, G. E. Jr.: "100 Uses for School Data
Processing", National Data Processing: New Dimensions
and Prospects, R. A. Kaimann (ed), Boston: Houghton
Mifflin, 1967, pp. 206-209.
6. Bent, Rudyard K. and Kronenberg, Henry H.: "Principles
of Secondary Education", 5th Ed., New York: McGraw-Hill,
1966, pp. 10.
7. Ibid, pp. 38-41.

8. Austin, David B. and Gividen, Noble: "The High School Principal and Staff Develop the Master Schedule", New York: Bureau of Publications, Teachers College, Columbia University, 1960, pp. V.
9. Ibid, pp. 26.
10. Lewis, C. F.: "The School Time-Table", London: Cambridge University Press, 1961, pp. 1-2.
11. Austin, David B. and Gividen, Noble: "The High School Principal and Staff Develop the Master Schedule:", New York: Bureau of Publications, Teachers College, Columbia University, 1960, pp. 55-64.
12. Farris, Carolyn S.: Personal interview, June 25, 1972.
13. Austin, David B. and Gividen, Noble: "The High School Principal and Staff Develop the Master Schedule:", New York: Bureau of Publications, Teachers College, Columbia University, 1960, pp. 71.
14. Blakesley, James F.: "Registration is Now a Matter of Minutes", College and University Business, Vol. 25, No. 5, Nov. 1959, pp. 38-44.
15. Faulkner, M. A.: "Computer Sectioning and Class Scheduling for a University", Automated Education System, Enoch Haga (ed), Elmhursts, Ill.: Business Press, 1967, pp. 233.

16. Gotlieb, C. C.: "The Construction of Class-Teacher Time-Table:", Proc. IFIP Congress 62 (Munic) North Holland Publ. Co., 1963, pp. 73-77.
17. Csima, J. and Gotlieb, C. C.: "Test on a Computer Method for Constructing School Timetables", Communications of the ACM, Vol. 7, No. 3, March 1964, pp. 160-163.
18. Holzman, A. G. and Turkes, W. R.: "Optimal Scheduling in Educational Institutions", Cooperative Research Project No. 1323, University of Pittsburgh, Pittsburgh, Pa., 1964, pp. 27.
19. Kiviat, P. J.: "Probabilistic Aggregate Approach", Cooperative Research Project No. 1323, A. G. Holzman, University of Pittsburgh, Pittsburgh, Pa., 1964, pp. 338-351.
20. Lions, John: "Matrix Reduction Using the Hungarian Method for the Generation of School Timetables", Communications of the ACM, Vol. 9, No. 5, May 1966, pp. 349-354.
21. Kaufmann, A. and Faure, R.: "Introduction to Operations Research", New York: Academic Press, 1968, pp. 90.
22. Ackoff, R. L. (ed): "Progress in Operation Research", New York: Wiley, 1961, pp. 149-150.
23. Hall, M.: "An Algorithm for District Representatives", American Math. Month. 63 (Dec. 1956), pp. 716-717.

24. Kenney, J. B. and Rentz, R. R.: "Automation and Control of Public School Instruction Records", Itasca, Ill.: F. E. Peacock, Publishers, 1970, pp. 48-58.
25. Richardson, Duane: "Pupil Scheduling by Computer", National Data Processing: New Dimensions and Prospects, R. A. Kalman (ed), Boston: Houghton Mifflin Co., 1967, pp. 191.
26. Schwartz, H. J.: "Experiences With the Ontario School Scheduling Program", AEDS Monitor, (March 1968), pp. 3-4.
27. Lions, J.: "The Ontario School Scheduling Program", The Computer Journal, Vol. 10, No. 1, (May 1967), pp. 14-21.
28. Winters, W. K.: "A Scheduling Algorithm for a Computer Assisted Registration System", Communications of ACM, Vol. 14, No. 3 (March 1971), pp. 166-171.
29. Jensen, George: "Student Scheduling Phase of Computer Assisted Registration at the University of Tennessee", CARS Reprot No. 4, (May 6, 1969), University of Tennessee, Knoxville, Tennessee, pp. 3.
30. Faulkner, Martin: "Computer Sectioning and Class Scheduling", Datamation, Vol. 11, No. 6 (June 1964), pp. 36.
31. Faulkner, Martin: "Computer Sectioning and Class Scheduling for a University", Automated Education Systems, Enoch Haga (ed), Elmhurst, Ill.: Business Press, 1967, pp. 234-236.

VITA

Kenneth Lee Fore was born to George Washington and Mayme Mason Fore on June 27, 1939, in Joplin, Missouri where he received his primary and secondary education. He received a Bachelor of Science degree in Electrical Engineering from the Missouri School of Mines, Rolla, Missouri in May 1962.

Since June 1971 he has been employed as a teacher of Data Processing for the Waynesville-Fort Leonard Wood School District, Waynesville, Missouri; also having work experiences with Western Union Telegraph Company, Eagle-Picher Industries, and Missouri Southern College.

APPENDIX A

100 Uses for School Data Processing

BUDGETING AND ACCOUNTING

1. Request and estimate analysis.
2. Allocation to specific categories.
3. Maintenance of amount available in each category after each transaction affecting that category.
4. Notification when a budget category is not being sent as anticipated.
5. Detail accounting of how the resources in each category were spent.
6. Cost analyses of goods and services, and of educational programs.
7. Projected costs in each budget category of proposed actions.
8. Income versus outgo projections.
9. Financial reports to boards, the state, individual schools, and so forth.

PAYROLL

10. Preparation of salary checks.
11. Maintenance of school and employee accounts for withholding tax, insurance, retirement, and so forth.

12. Preparation of documents for tax, retirement and insurance officials.
13. Preparation of employee withholding tax statement.
14. Salary accounting from various budget categories and to various educational programs.

PURCHASING

15. Accumulation of requisitions for quantity discounts.
16. Encumbrance and release of funds.
17. Product and vendor statistical information.
18. Analysis of disposition of purchased goods and services.
19. Follow-up for uncompleted purchase orders.

SUPPLIES AND INVENTORY

21. Preparation of requisition procedures for stock items.
22. Automatic reordering.
23. Allocation of costs to budget categories and projects.
24. Preparation of delivery schedules and routing.
25. Inventory of materials in use or stocked.

ACCOUNTS PAYABLE AND RECEIVABLE

26. Generation of bills.

27. Crediting and debiting of proper accounts.
28. Follow-up for unpaid bills.

MAINTENANCE

29. Scheduling of preventive maintenance.
30. Repair schedule and costing.
31. Replacement scheduling.
32. Analysis of product durability.

CAFETERIA ACCOUNTING

33. Analysis of available foods and prices in relation to diet and consumption habits of pupils.
34. Food ordering, inventory and payment.

INSTRUCTION MATERIALS RECORDS

(A-V, books)

35. Ordering.
36. Cataloging.
37. Requisitioning.
38. Analysis of actual use by teacher, department, subject and pupil.
39. Analysis of maintenance and cost.

PERSONNEL

40. Analysis of applicant interview and qualifications.
41. Experience and qualifications of staff members.

42. Job evaluation.
43. Salary information.
44. Academic credits.
45. Leave record, including illness.
46. Studies of geographic origin, education certification and professional mobility patterns.

PUPIL CENSUS

47. Enrollment predictions.
48. Attendance law compliance, particularly for pupils in other schools.
49. Source of verified birth date for test and legal purposes.
50. Address directors for capitation purposes.
51. Federal employment of parents survey.
52. Age in grade reports.
53. Bus transportation planning and control.
54. Mailing of school communications (one per household).
55. Pupil directories for all school offices.
56. Preschool surveys.

REGISTRATION AND SCHEDULING

57. Recording of course requests.
58. Comparison of requests with anticipated

instructional program for pupil (and for school if there is a choice of schools).

59. Check for prerequisites.
60. Summer school courses.
61. List of pupils requesting specified courses for screening (advanced placement, band, courses to be withdrawn).
62. Interaction or potential conflict analysis.
63. Projection of enrollment in higher level courses based on requests for beginning courses.
64. Construction of the master schedule.
65. Simulation of proposed master schedules for purpose of refinement.
66. Pupil schedules.
67. Class lists.
68. Homeroom lists.
69. Unhonored request lists for counselor action.
70. Room characteristics (for scheduling purposes).
71. Room utilization studies.
72. Extracurricular activity time assignment.
73. Locker assignment and lock combination records.

ATTENDANCE ACCOUNTING

74. Daily attendance bulletin.
75. Period-by-period accounting of pupils not present and not on the daily bulletin.

76. Preparation of register pages for manual recording of optical scanner sheets for machine analysis.
77. Lists of pupils with unusual attendance characteristics.
78. Recording, posting and summarizing: by pupils, by classroom, by school, and by district.

MARK REPORTING

79. Printing of report cards.
80. Summary lists for teachers, department heads, counselors.
81. Pupil transcripts.
82. Mark analysis by course and teacher.
83. Comment analysis by course and teacher.
84. Failure, near-failure, and incomplete lists.
85. Computation of various averaging and ranking statistics.
86. Notification of counselors when pupil marks or number and type of comments deviate from expectation.
87. Honor roll and rank in class determination.
88. Underachiever identification.
89. Preparation of anticipated mark distribution for each class from distribution of tested ability and school's marking practices for the course.

MISCELLANEOUS

90. Summaries of ability versus achievement.
91. Administrative roles of staff.
92. College admission studies: What kinds of pupils make good where?
93. Misconduct reports and analyses.
94. School insurance accounting.
95. Physical education skill and achievement records.
96. Health and dental records.
97. Identification of pupils with handicaps or special needs.
98. Population studies.
99. Dropout prediction.
100. Dropout analysis.

APPENDIX B
CARDINAL PRINCIPLES of SECONDARY
EDUCATION (1918)

- 1) Health - The school can teach certain facts about health, help pupils form good health habits, and build attitudes that will promote health.
- 2) Command of fundamental processes - Usually refers to basic skills in reading, writing, and arithmetic. All students should increase their skill in English and arithmetic during high school.
- 3) Worthy home membership - An understanding of the importance of the home as a social institution, and one's place and duties in it, is embodied in this aim. School life is as worthy as home life, and one learns to be a member of one through participation in the other because of common elements between them.
- 4) Vocation - Everyone at some time in life is responsible for his own welfare and usually for that of others. The multitude of different occupations, the period of training required for them, and the demands of a changing society have led to more emphasis on vocational training.

- 5) Citizenship - All school subject should foster good citizenship. Patriotism must not be confused with citizenship, although it is a desirable trait.
- 6) Worthy use of leisure - Leisure activities should be recreational, healthful, and, if possible educational. Pupils should be taught to utilize the common means at their disposal in their own homes and communities for leisure pursuits, such as games, literature, art, music, and science.
- 7) Ethical character - Crime and delinquency have revealed that youth who are incapable of directing their own conduct are a menace to society. The aim of ethical character should be taught indirectly, and practically every school subject can make a contribution to it. It is comparable to good sportmanship and fair play in athletics; honesty in all schoolwork; respect for property, law, order, and authority; and an attitude of reverence for a supreme being.

APPENDIX C

WAYNESVILLE HIGH SCHOOL REGISTRATION YEAR 1971-1972

NAME _____
 (last) (first) (middle)

ADDRESS _____

PHONE _____
 (Home) (Business)

PARENT OR GUARDIAN _____

OCCUPATION _____
 (If military give rank)

PLEASE CHECK THE FOLLOWING

10 11 12

- I plan to attend WHS during 1971-72
- I will not attend WHS
- I am uncertain about attending WHS during 71-72

SOPHOMORE

| 1st Semester | 2nd Semester |
|--------------|--------------|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |

JUNIOR

| 1st Semester | 2nd Semester |
|--------------|--------------|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |

SENIOR

| 1st Semester | 2nd Semester |
|--------------|--------------|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |

Please read the accompanying course descriptions carefully before indicating your choice of electives in that teachers will be employed to meet your needs as indicated by the subjects you choose. Thus, schedule changes at a later date will be almost impossible.

 DATE PARENT'S SIGNATURE

 DATE COUNSELOR'S SIGNATURE

APPENDIX E

MANUAL ENROLLMENT PROCEDURE
FOR WAYNESVILLE HIGH SCHOOL, 1970-71

MAY:

Pre-enrollment forms are handed out to all students, even if they will not be returning the next year.

Pre-enrollment forms are tabulated.

ABOUT JUNE OR JULY:

The master schedule is made up from the tabulation of the pre-enrollment forms.

Students are scheduled from the pre-enrollment forms and the master schedule generated by the principal. A 3 x 5 card; with student's name, teacher, and section; is made out for each student scheduled. Cards are then slotted into a "mail box" and tabulated by number of cards in designated slot, i.e.

| Period | Teacher A | Teacher B | Teacher C | Teacher D |
|--------|-----------|-----------|-----------|-----------|
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

Manual "Mail Box" Sectioning of Students

Student help is used to slot and tabulate the cards and to keep a running count of students scheduled in each class. The count of classes is kept on a copy of the master schedule and distributed to each person scheduling students, so that they will have an accurate tally of classes. When class size maximum (size set by principal) is reached, no more students are to be scheduled in that section and it is marked closed on the master schedule chart. Sections were generally closed at 30, to allow for special situations where students must be scheduled in a certain section. The count could go as high as 35, the maximum for meeting accreditation standards. (Exceptions were made for Physical Education and Music, whose count could exceed 35).

Counselors and their secretary scheduled students (student help did not). Students with conflicts (classes chosen on pre-enrollment form but not offered or two classes being offered the same period) were not scheduled but put into a separate file for registration, when counselors would talk with students individually to resolve conflicts.

FIRST OF AUGUST:

A meeting was held to determine various steps of registration, i.e.: where to get books, pay fees, pick up schedules, see counselors to resolve conflicts, etc. The scheduling of students continued.

REGISTRATION: (Middle of August)

Students without conflicts enrolled, picking up a copy of their schedule. Students with conflicts (and all graduating seniors) were seen by counselors and their schedule was adjusted according to the student's wishes. Seniors who had failed to meet graduation requirements were rescheduled. These schedules were recorded on 3 x 5 cards (same method as stated above). This generally caused an imbalance in class size for some sections and other students whose schedule could easily be changed were rescheduled to alleviate this problem. These students would then have to be notified of their class change before the beginning of school. Their cards were removed from the original class and reslotted according to change.

AFTER ENROLLMENT (Last of August)

Cards were removed from "mail box" slot, alphabetized and typed into a class roll according to teacher. (Each teacher received a list of students in each of the five sections.) The Guidance Office retained a copy to be kept up to date as students were added and deleted - this was done by hand by the secretary and/or student help. Ideally, by the first day of school teachers had an alphabetized listing of their

students by period, the Guidance Office had a list of where each student was each period, plus a complete schedule of the student's six classes.

APPENDIX F

Sample Course Descriptions

The following course descriptions will provide some information concerning courses which the student must take to meet state and local graduation requirements and should also be of help to the student as he elects additional courses from the many available to him. Since course descriptions are necessarily brief, the student or parent who desires additional information should consult with the counselor, principal, or teacher as needed.

The student may enroll in courses listed for or below his grade level.

1. COMMUNICATIVE SKILLS (3 units required)

A. LANGUAGE ARTS

Language Arts I (Grade 9) (1 unit)

This course is designed to help the student become more proficient in the use of the English language through speaking, listening, reading, writing and spelling. Through the study of functional grammar and good literature the student learns to communicate more effectively with others.

Language Arts II (Grade 10) (1 unit)

This course is designed to further increase the student's abilities to communicate effectively through speaking, reading and writing. The literature studied at this level is quite varied with the objectives of increasing the student's appreciation of many kinds of good literature.

Language Arts III (Grade 11) (1 unit)

Attention is given in this course to further refinement of the technical skills essential to effective written and oral communication. Emphasis is placed on the correct and intelligent interpretation of what is heard or read. A research paper will be written. Language Arts III includes the study of American Literature.

Literature of England (Grade 12) (1/2 unit)

This is a brief survey of writings from the Anglo-Saxon era to the contemporary period. Emphasis will be placed on enjoyment and appreciation as well as evaluation of types of literature. A research paper is required.

Appendix G

Sectioning Logic Sequence a Small School

Assume a small school that offers only five courses and meets but one day a week. The day is broken into six one-hour periods.

Students initial time vector (000000)

Time vector of each course in the time schedule

Course A (100000) meets at 8 a.m.
 Course B (000100) meets at 11 a.m.
 Course C (010001) meets at 9 a.m. and 1 p.m.
 Course D (001100) meets at 10 a.m. and 11 a.m.
 Course E (010010) meets at 9 a.m. and 12 p.m.

Student requests course A, C, D, and E

Denote the student's time vector by S. Initially S(000000)

The sectioning process is as follows:

| | | |
|---------------|-----------------|-------------|
| | S (000000) | |
| AND | A (100000) | |
| Test for zero | <u>(000000)</u> | no conflict |

| | | |
|----|----------------|-------------------------|
| OR | S (000000) | |
| | A (100000) | |
| | S (A) (100000) | first request sectioned |

| | | |
|---------------|-----------------|-------------|
| | S (A) (100000) | |
| AND | C (010000) | |
| Test for zero | <u>(000000)</u> | no conflict |

| | | |
|----|------------------|--------------------------|
| OR | S (A) (100000) | |
| | C (010001) | |
| | S (A,C) (110001) | second request sectioned |

| | | | |
|---------------|-----------|-----------------|----------------------------|
| AND | S (A,C) | (110001) | |
| Test for zero | D | <u>(001100)</u> | |
| | | (000000) | no conflicts |
| OR | S (A,C) | (110001) | |
| | D | (001100) | |
| | S (A,C,D) | (111101) | third request sectioned |
| AND | S (A,C,D) | (111101) | |
| Test for zero | E | <u>(010010)</u> | |
| | | (010000) | conflict |

Student cannot be sectioned into Course E.

Appendix H

Scheduling Costs

DEVELOPMENTAL COST

| | |
|-----------------------------|---------------|
| Programming & Computer Time | 3,000.00 |
| Clerical Workers | 1,170.00 |
| Supplies | <u>100.00</u> |
| | \$4,270.00 |

APPLICATION COST OF COMPUTER SCHEDULING SYSTEM

| | |
|-----------------|---------------|
| Computer Time | 50.00 |
| Supplies | 50.00 |
| Administrative | 200.00 |
| Clerical Worker | 190.00 |
| Royalty Fee | <u>150.00</u> |
| | \$640.00 |

HAND SCHEDULING COST

| | |
|-----------------|--------------|
| Counselors | 1,575.00 |
| Clerical Worker | 780.00 |
| Student Worker | 192.00 |
| Supplies | <u>30.00</u> |
| | \$2,577.00 |