DOCUMENT RESUME

ED 070 320 EM 010 668

AUTHOR	Otte, Richard
TITLE	Computer Technology Program (REACT). Technical Report.
INSTITUTION	Northwest Regional Educational Lab., Portland, Oreg.
SPONS AGENCY	National Inst. of Education, Washington, D. C.
BUREAU NO	BR-1-0749
PUB DATE	31 Aug 72
GRANT	OEG-0-71-4655
NOTE	214p.; See Also EM 010 204
EDRS PRICE	MF-$0.65 HC-$9.87
DESCRIPTORS	Computer Assisted Instruction; *Computer Science Education; *Course Descriptions; Formative Evaluation; Instructional Materials; Programing; *Secondary School Students
IDENTIFIERS	Computer Technology Program; REACT

ABSTRACT

Development work of the Computer Technology Program (or REACT-Relevant Educational Applications of Computer Technology) during the period of July 1, 1971 to August 31, 1972 is summarized here. Work during this period included writing and developmental testing of a course in computer careers for high schcol pupils. This phase took the course through two testing and revision cycles and brought it to the interim product state ready for field testing. The report contains a background of the program, descriptions of products and testing program, results, and actions and recommendations. In addition there is a sample of the prototype manuals produced. (Author/JY)

Technical Report

COMPUTER TECHNOLOGY PROGRAM (REACT)

August 31, 1972

Conducted Under USOE Grant OEG-0-71-4655
National Institute of Education
Richard Otte, Project Officer

by Northwest Regional Educational Laboratory
710 S.W. Second Avenue/Lindsay Building
Portland, Oregon 97204

1- 0 4 9

TABLE OF CONTENTS

INTRODUCTION

The purpose of this report is to summarize the development work of the Computer Technology Program (or REACT - Relevant Educational Applications of Computer Technology) during the period of July 1, 1971 to August 31, 1972, under USOE Grant OEG-0-71-4655. This phase included writing and developmental testing of a course in computer careers for high school pupils. This phase took the course through two testing and revision cycles and brought them to the interim product state ready for field testing. The report is organized as follows:

- Background of Computer Technology Program (REACT)
- Description of products from this project
- Testing program
    - Goals
    - Procedures
- Results
- Actions and recommendations

The report also contains in Appendix A a sample of the prototype manuals produced.

BACKGROUND OF COMPUTER TECHNOLOGY PROGRAM (REACT)

Since 1969, Program REACT of the Northwest Regional Educational Laboratory has been developing course materials for a wide variety of target audiences in the field of computers in education.

Earlier work is described in the REACT/CUES (Computer Utility for Educational Systems) Technical Reports of January 31, 1970 and January 31, 1972. This work focused primarily on developing and testing three computer courses for teachers and school administrators. Work was also done in development of student curriculum materials for the various academic disciplines.

Long-range program plans call for further work in teacher training, administrator training, pupil curriculum materials and career education in the computer field. In 1971-72, work was concentrated on a project in the latter - career education in computers. The development and testing of these materials is the subject of this report.

DESCRIPTION OF PRODUCTS FROM THIS PROJECT

The printed products of the project in computer careers make up a complete course titled The Elements of Computer Careers. It includes these manuals:

- USER'S GUIDE

  This is the guide for teachers, administrators or course managers to help them use the materials effectively. It includes a description of the materials, discussion of the hardware necessary and alternative ways to obtain it, teacher training needs, helps in organizing the course, a case study of the field test course, lists of resources, and more.

- THE ELEMENTS OF DATA PREPARATION
- THE ELEMENTS OF COMPUTER OPERATION
- THE ELEMENTS OF COMPUTER PROGRAMMING
- THE ELEMENTS OF SYSTEMS ANALYSIS
- THE ELEMENTS OF SYSTEMS PROGRAMMING
- THE ELEMENTS OF COMPUTER CENTER MANAGEMENT

  These manuals are the heart of the training system. Each covers a job area in a computer center. Each manual is divided basically into three sections. The first part presents the nature of the job itself. In this section, the student learns about salaries, working conditions, education and training requirements, career paths, and other general descriptions of the job field itself. In section two, the pupil is presented some of the skills and concepts associated with the particular job. These are not at a high level of sophistication, but enough to give the student a feel for the job and to solve some problems. The final section presents a series of problems or exercises in the particular field. A large majority of these can be solved even though a computer is not available.

- COMPUTER CAREERS IN PERSPECTIVE

This booklet is a general description of the types of career opportunities available in the computer field. It covers the fields of business and industry, scientific research, and education. Applications in these areas are described with specific examples of projects which a student might work on if he were to be employed in that area.

TESTING PROGRAM

Goals

The goals of the testing and research on this project were these:

- obtain performance and opinion data to support revision of the training manuals
- measure the effect of this type of program on a typical high school
- document the problems and procedures of implementing this type of program in a typical high school

Procedures

Test Site

The test site selected for this developmental testing phase of the program was Centennial High School in Gresham, Oregon. The school is located in a middle-class, white suburb and enrolls approximately 2,000 pupils. The school had an existing computer program, utilizing a regional computing center via the medium of a teletypewriter terminal for problem-solving and programming experience. The school had one teacher with a good computer background and several others with a very slight computer background. These factors indicate that the school, and therefore the test pupils, probably approximate closely the ultimate target population.

Testing Technique

The training manuals were tested three times by three distinct classes. Necessary revisions were made on the basis of the data obtained in each testing phase. Post-test scores were analyzed by item to pinpoint specific weak spots in the textual material. Questionnaires were administered to extract more general reactions from the students about the manuals. All testing was done in small groups (1 to 4) under direct observation of a staff member. In this way, problems (such as blockages, confusing passages or instructions, and other irregularities) could be documented as they arose. The tests given were mostly multiple-choice, with some completion items included. There was one test for each manual; the average number of items was 22, with a range of 17 to 31. A sample of the tests is shown in Appendix B.

Data

Data collected were:

- pre-test score
- post-test score
- answers to opinion questionnaire completed by pupils:

  questions related to:

  - general reactions
  - appropriateness of content
  - appropriateness of problems
  - appropriateness of level of difficulty
  - pupil descriptive data

Analysis of Data

Test scores were analyzed in this way:

- Differences in scores were computed between pre and post tests (by individual and by group) to show measures of growth. These were interpreted within the context of the recognized limitations of this technique, particularly with small N-counts.
- Item analyses of the post-test scores were made to isolate areas needing attention. When the mean performance for a group dropped below 80% for a particular item, special attention was given to the item itself or to the content which supported it. The 80% figure is an arbitrary one chosen as a "red flag" in the revision process and does not suggest that this is either a minimum or a satisfactory performance for all students.

Opinions

Opinion questionnaires were analyzed in this way:

- Responses were summarized and categorized as much as possible.
- The information extracted from the questionnaires was used in the revision process, bearing in mind the relative maturity and sophistication of the test subjects.

Effect on School

Measures of the effect of the program on the school were:

- Number of teachers using the computer in some way in their teaching. These were teachers who took some instruction in curriculum applications, then worked with students in designing computer curriculum applications for theii own classrooms.
- Number of pupils who were reached by computer-related activities, both directly and indirectly through the applications referred to above.

RESULTS

The first major goal, to obtain revision data, was realized. An example of the item analysis data collected is shown in Figure 1 on page 11, this being from the summer school sessions. Data from all manuals is available from the Laboratory but is not included in this report. Every manual was revised during the year -- some minimally (typographical errors, occasional words or sentences that were unclear, etc.) and some extensively (up to 20-30% -- in some cases, major sections being discarded and replaced). Some of the testing followed much of the major revisions, but further field testing is indicated. The evaluation plan for the field testing is in Appendix C.

The results of the questionnaires were generally positive, but were interpreted in view of the maturity level of the respondents. This type of evaluation, while general, did focus the attention of the developers to some problems or exercises that were too difficult, what was liked and disliked in terms of format, style, and so on.

The effect on the school was measured more informally. Definite impact, however, is represented by these figures. Nearly fifty students were directly involved in the program -- fifteen from an advanced computer course, twenty-five from an introductory course, and eight in a summer session. Nine teachers were directly involved -- that is, worked through the teacher training material provided them -- then worked directly with pupils in defining curriculum applications for their own subject areas. These teachers represented the following subject areas:

Mathematics	2	English	1
Science	3	Business	1
Social Studies	1	Industrial Arts	1

Approximate projections based on the class assignments of the teachers involved suggest that 300-500 pupils may have been indirectly affected by the project.

The third goal, that of documenting the project, is dealt with elsewhere in this document. Basically, the information gained in this area is contained in the User's Guide.

UNIT IV - ELEMENTS OF SYSTEMS ANALYSIS

QUESTION		PRE-TEST N = 7		POST-TEST N = 7		
NO.	PRT	CORRECT	%	CORRECT	%	GAIN
1		0	0.0	2	28.6	28.6
2		1	14.3	6	85.7	71.4
3	A	4	57.1	6	85.7	28.6
3	B	4	57.1	6	85.7	28.6
3	C	5	71.4	7	100.0	28.6
4		3	42.9	5	71.4	28.5
5		2	28.6	6	85.7	57.1
6	A	4	57.1	7	100.0	42.9
6	B	4	57.1	7	100.0	42.9
6	C	4	57.1	7	100.0	42.9
7		1	14.3	5	71.4	57.1
8		2	28.6	5	71.4	42.8
9		1	14.3	7	100.0	85.7
10		1	14.3	3	42.9	28.6
11		3	42.9	6	85.7	42.8
12		2	28.6	4	57.1	28.5
13		4	57.1	5	71.4	14.3
14		2	28.6	5	71.4	42.8
15		0	0.0	5	71.4	71.4
16	A	4	57.1	6	85.7	28.6
16	B	4	57.1	7	100.0	42.9
16	C	5	71.4	6	85.7	14.3
17		0	0.0	5	71.4	71.4
18		2	28.6	4	57.1	28.5
19		0	0.0	4	57.1	57.1
20		2	28.6	5	71.4	42.8
21	A	5	71.4	7	100.0	28.6
21	B	4	57.1	6	85.7	28.6
21	C	5	71.4	6	85.7	14.3
22		1	14.3	4	57.1	42.8
25	1	0	0.0	3	42.9	42.9
25	2	0	0.0	4	57.1	57.1
25	3	0	0.0	3	42.9	42.9
25	4	0	0.0	3	42.9	42.9
25	5	1	14.3	6	85.7	71.4
			32.65		74.68	42.03

	PRE-TEST N = 35		POST-TEST N = 35		
STUDENT	CORRECT	%	CORRECT	%	GAIN
1	16	45.7	19	54.3	8.6
2	20	57.1	32	91.4	34.3
3	11	31.4	29	82.9	51.5
4	0	0.0	18	51.4	51.4
5	17	48.6	32	91.4	42.8
6	7	20.0	30	85.7	65.7
7	9	25.7	23	65.7	40.0
		32.64		74.69	42.04

Figure 1

ACTIONS AND RECOMMENDATIONS

Actions taken as a result of the testing program were:

- All manuals were revised on the basis of cues from the test item analysis, opinion questionnaires, and observations of the pupils at work.
- The User's Guide was written to help implement the course most effectively and efficiently, given the circumstances which exist in a given school. Suggestions are made for using the materials in different ways, depending on time and hardware availability.

Further field testing is currently being planned which will test the course under more natural conditions, i.e., without the presence of development staff.

The User's Guide was the last manual to be written and includes, along with descriptions of the course, recommendations for how to use the materials. The next paragraphs are summaries from that manual.

REACT Career Training is an integrated system of student training manuals, computer hardware, and teacher training. Ideally, all three components will be implemented. It is possible, however, to use the student manuals without hardware or teacher training.

The manuals are designed so that optimum use is with a student computer center available for laboratory work. Students have the opportunity to "try out" each of the roles in a real setting as they progress through the manuals.

In order to provide students with experiences that are as close as possible to reality, the teachers or administrators in the school can act as "users," served by the student computer center. Students then act as data preparers, computer operators, programmers, analysts, systems programmers, and managers in order to develop the applications asked for by these "clients" (teachers/administrators).

Alternatives for Using the System

The manuals have been designed for flexibility in use. Ideally, pupils should progress through the manuals in the sequence shown. A teacher may elect, however, to use specific manuals out of order or in isolation. Several ways of using the materials are presented here as guides for the teacher in designing his or her own program.

Mode One - Individual Instruction

A teacher may select individual pupils to study one or more manuals on an independent basis. In this mode, students may branch out of the sequence at any point to pursue a particular occupational area (for example, computer programming) in some depth.

Mode Two - One Semester Career Orientation

If there is no computer center available on which the students can extend their knowledge gained from the books by "hands-on" practice, the manuals will still be valuable. In this mode, the total instruction time lends itself to a one semester course. By the end of the semester, students should have achieved a basic orientation to the computer-related career clusters they have studied. They should know if they have enough interest in any of these occupational areas to warrant further training.

Mode Three - Full Year Orientation

In its optimal usage, the course will last a full year and will utilize the study of the manuals, interaction with teacher/administrator "clients" on extensive projects, and student operation of a small computer center. These activities should be sprinkled liberally with books, films, guest speakers, field trips and other supplementary resources. The last section of this manual contains suggestions for these. The course may be a full course totally devoted to career orientation, or the materials could become the focus for an already existing Introduction to Computers course. Used in this way, it would cover many of the concepts probably previously included in the course, but would shift the emphasis to career orientation.

Using the Manuals

The manuals are written to be self-instructional. Experience with similar courses previously developed, however, indicates that while there will be no "teacher" in the traditional lecture-recitation sense, a teacher or course manager is necessary to supervise, control the flow of materials, evaluate progress, and generally do trouble-shooting if the materials are being used by a class-size group.

APPENDIX A

SAMPLE MANUAL

COMPUTER TECHNOLOGY PROGRAM

NORTHWEST
REGIONAL
EDUCATIONAL
LABORATORY

THE ELEMENTS OF
COMPUTER CAREERS UNIT 2

THE ELEMENTS OF COMPUTER OPERATION

2

12,

COMPUTER TECHNOLOGY PROGRAM

NORTHWEST
REGIONAL
EDUCATIONAL
LABORATORY

THE ELEMENTS OF
COMPUTER CAREERS UNIT 2

THE ELEMENTS OF COMPUTER OPERATION

Published by the Northwest Regional Educational Laboratory, a private nonprofit corporation supported in part as a regional educational laboratory by funds from the United States Office of Education, Department of Health, Education and Welfare. The opinions expressed in this publication do not necessarily reflect the position or policy of the Office of Education, and no official endorsement by the Office of Education should be inferred.

This material represents developmental work to date and is not the final validated version. Please contact the Laboratory for information regarding the status of the final product.

TABLE OF CONTENTS

Part III: Computer Software

Part IV: Procedures for Computer Operation

Part V: An Exercise in Computer Operation

INTRODUCTION

One of the key people on the production end of data processing is the computer operator. He is the individual responsible for overseeing the operation and the final output of the computer system.

In Part I of this manual, you will learn what a computer operator does, what the requirements of the job are, and how the position fits with the other jobs in the typical computer center organization. You will also learn about the career paths open to computer operators.

In the second part, you will learn about the general flow of computer processing and about the equipment which makes up a computer system. Part III describes the various aspects of computer software (that is, programs) which make the computer work. Also in this section, you will see how computers run in different modes to produce different kinds of output and how data looks inside the computer.

At several points in Parts I, II, and III, study questions are given under the heading "Check Your Understanding." At the end of each set of questions, a note tells you where to find the answer key to the questions. You should answer the questions in each set, then check and correct your answers using the key.

Part IV of this manual first takes you step-by-step through a typical day of computer operation. Some guidelines on dealing with problems in the runs and on handling the data media are then given. And, finally, an exercise in computer operation is presented in which you are expected to take the part of the operator and to perform typical tasks like filling out logs, making decisions, and interpreting information.

Once you have completed this manual, you should have a clear understanding of what a computer operator does and what importance his job has. At that point, you should have enough information to decide whether or not you are interested enough in this kind of work to study more about it.

PART I

THE WORLD OF THE COMPUTER OPERATOR

PART I THE WORLD OF THE COMPUTER OPERATOR

THE JOB

In the computer center, the person who works most directly with the computer and its related equipment is the computer operator. In fact, he is responsible for the efficient use of the computer itself and of the additional devices which feed information into and out of the computer. No computer center can function very efficiently without a competent and skillful computer operator.

When the computer center's programmer has finished writing the program for a specific task (or "job," as it is called) and has gathered together all the data to be used with the program, he is nearly ready to put the "running of the job" into the computer operator's hands. But before he does, he must have the program (which tells the computer what to do) and the data (the information the computer is to do something to) put into a form which can be fed into the computer.

Usually the programmer has his program and data put into the form of punched cards or punched paper tape by a data preparation clerk. He then takes his decks of punched cards or paper tapes to the computer operator along with an "instruction sheet" giving any special instructions involved in "running" the program. Here the computer operator's work starts.

The computer operator first schedules the running of the job on the basis of the job's importance and the availability of his computer's time. Then, at the scheduled time, he sets up his computer equipment to run the job as directed by the instruction sheet. This includes picking out and mounting appropriate printout paper on the computer printer and loading the input (cards, paper tapes, etc.) onto the appropriate input devices (card reader, paper tape reader, etc.) When all the card or tape input for a job is attached to the instruction sheet by the programmer, the computer operator has it automatically at hand when he is ready to run the job. In many cases, however, some of the programs and data needed for a job are stored on cards or tapes kept in the computer room. In these cases, the operator must gather the required input from shelves or files and then load them for use at the appropriate time.

When the computer operator is ready to put data in the form of punched cards into the reader, he loads the deck of cards into the input hopper at the top of the reader. The cards are automatically passed into a "wait" section in the reader and from here, they are moved one at a time through the "read" station.

You already know, generally, what happens in the card reader's "read" station: the punched-hole code is "read." But, more precisely, there are twelve photocells in the station which "read" one column on a card at a time and register the code punches. The data from each card is picked up this way and transferred electronically into the computer, in fact, where it can be located ("accessed") for processing by the computer the moment it is needed.

After each card is read, it is moved automatically into the card stacker on top of the card reader unit. When the cards have all gone through the reader, then, they end up stacked in order in the card stacker.

The convenience of the card readers' input hopper and card stacker allow the computer operator to load and retrieve the decks of punched cards easily.

Paper Tape Reader

A second common input device is a paper tape reader, which is designed to "read" data programs from the medium of paper tape. On a roll, blank paper tape look like this:

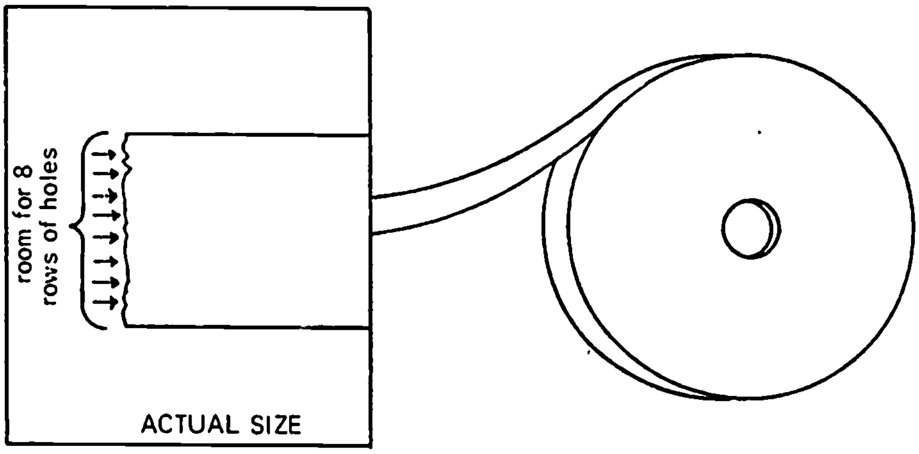


Fig. 8 Blank roll of paper tape.

Once his equipment is correctly set up for the job, the operator loads the program into the computer, so that the computer is prepared to do all the proper things to the data that is to be processed. Now, the operator starts the computer "run" and monitors the computer's operations from the control panel and the console typewriter. As you can guess, the console typewriter allows the computer operator to type in directions to the computer as required by the instruction sheet, such as telling it initially to start the run. The console typewriter is also used to type in information which the program requests, such as the current date. The control panel is equipped with buttons, switches and lights which inform the operator what the computer is doing and which allow him to regulate its operation.

When the control panel lights show that the machine has stopped for some reason, the operator has to determine if there is a problem and if so, he must correct it or call in a maintenance engineer. He also monitors the control panel, the console typewriter, and the output (data printed out by the computer) for any evidence that the run is not proceeding correctly, and he stops the job if necessary.

Whenever a run cannot be finished because of a problem in the program or data, the operator returns the cards or tape to the programmer along with the instruction sheet and a brief but precise report of all indications he observed of where the problem may lie.

When a job has been successfully run, on the other hand, the operator removes the cards or tapes used for the job and stores them in the proper place in the computer room. He then returns the instruction sheet and all output to the programmer. If the programmer has requested that cards or tapes also be returned to him, they are returned with the instruction sheet and output.

As you can probably imagine, computer rooms are usually very busy places and time is valuable. For this reason it is important for the computer operator not only to schedule the work carefully, but to keep an accurate log of the time used for each job. He is also responsible for writing periodic reports on the use of the computer he is in charge of.

In large computer centers, the computer operator usually has some help in keeping the computer room running smoothly and efficiently. Most often he will have one or more tape librarians to help with the location and handling of the magnetic tapes. He may also have assistants trained to operate some of the other equipment, including the input devices which feed information into the computer and the output devices, through which information comes out of the computer. These input and output devices include such equipment as card readers, high speed printers, and separate card handling equipment such as sorters

Like punched cards, data is recorded on paper tape by means of holes punched in coded sequence. As shown in the figure above, the inch-wide tape has room for eight rows of holes -- five running parallel to the drive sprocket holes on one side and three running parallel on the other side. The most commonly used paper tape code, ASCII, uses the first seven of these rows.

The illustration below shows a paper tape with the statement "THIS IS A SAMPLE TAPE" punched in ASCII code. Each column of punched holes represents one letter, number, or symbol. Although the eighth (or bottom) row of holes has been punched, the punches in that row are not a part of the code.*

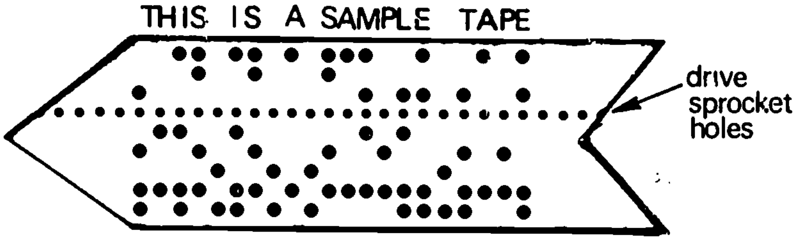


Fig. 9 Tape punched in ASCII code.

When the computer operator is ready to load data from the medium of paper tape into the computer, he mounts the roll of punched paper tape on a paper tape reader such as the one pictured on the next page.

*To learn more about ASCII code on paper tape, see pages 37 to 40 of the manual, THE ELEMENTS OF DATA PREPARATION.

and collators. In most small centers, however, the computer operator runs the computer and all the peripheral equipment himself, and he does all the handling and filing of tapes, disks and cards stored in the computer room.

REQUIREMENTS FOR THE JOB

Skills

Becoming a competent computer operator requires the development of several special skills as well as some more general ones.

Among the special skills he needs, the most important are those involved in setting up the computer equipment to run any given job and in operating the computer from the control panel and console typewriter. Both of these tasks require a basic knowledge of the various pieces of computer equipment and how they work together. Of course, the better the computer operator understands the equipment and the way each piece works with the others, the more efficiently he can run the computer system and the more quickly he can locate any trouble that might occur.

In addition to operating equipment in the computer system, it is helpful if the computer operator knows how to run the various machines which prepare and handle punched cards, including key punch machines, card sorters, card interpreters and verifiers, and card collators. He must also be proficient in the general handling and efficient storing of card decks, tape reels, magnetic disk packs, and other input/output media commonly kept in the computer room.

Good skills in basic arithmetic are important to the computer operator. Since computer printouts commonly involve numbers and results of simple arithmetic operations, the operator often checks the printout during the run for evidence of any errors or problems in the run. It is easy to see that the computer operator who can catch arithmetic errors at a glance can save a considerable amount of valuable time, both for himself and the computer.

Another skill of considerable importance to the computer operator is that of communicating effectively both in speech and in writing. He uses both forms of communication continually in his work, since he has to understand directions from and write explanations to programmers about specific computer runs, communicate with the computer

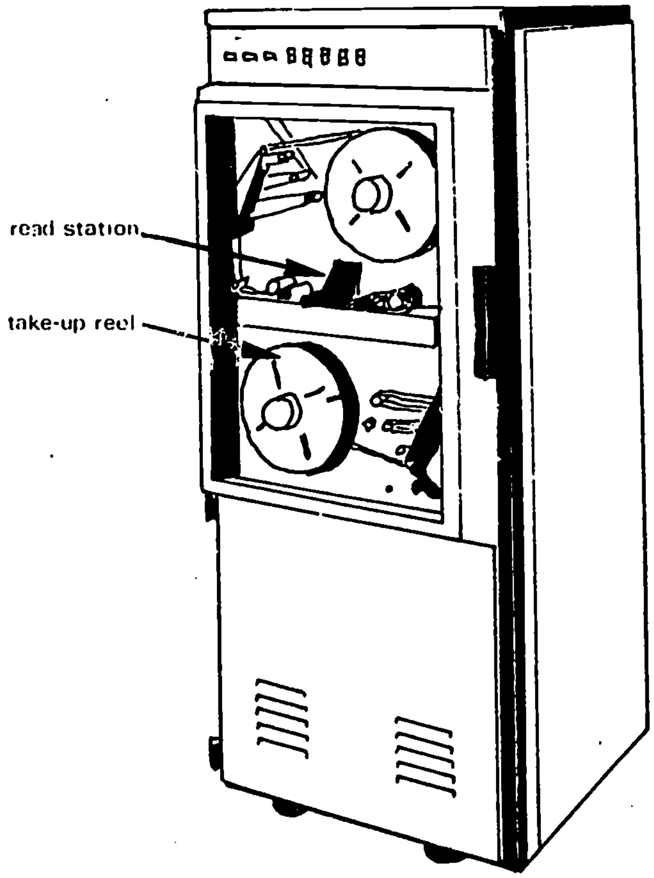


Fig. 10 A typical paper tape reader

As you can see from the illustration, the paper tape is threaded through the "read" station and wound onto a take-up reel. The "read" station is equipped with a series of sensors which read the coded holes on the tape, one column at a time. The data is then electronically transferred to the computer for use.

Most paper tape readers look something like the one pictured above, with the exception of the reader attached to the freestanding teletypewriter discussed on page 32. Each tape reader unit has a manual accompanying it which gives the computer operator explicit directions on how to mount, thread, run and rewind tape on the given device. In every case, the mechanics are relatively simple and straightforward. The computer operator needs only to follow the special directions in the manual and to be careful not to tear or perforate the tape being mounted and threaded.

through the console typewriter, and write periodic reports about the computer's use.

Education

The minimum education expected for the computer operator's position is a high school diploma. The candidate for the position should have obtained a useful general education and a good background in mathematics as well as in language arts. Most computer operators find that the high school courses which helped them the most were mathematics and English courses. Logic and psychology provide additional useful background.

Further, more specialized training for the job is desirable and can be a real boon to the high school graduate interested in a computer operator's career. Such training is readily available through special data processing schools and through classes offered by computer manufacturers. Computers centers may also make on-the-job training available.

Like others in the data processing profession, the computer operator's education doesn't stop when he gets the job. Continual changes in the data processing field in general and in computer equipment in particular make further training through on-the-job and off-hour courses necessary from time to time.

Personal Characteristics

Since the operation of machines is the "heart" of the computer operator's work, he clearly needs enough natural mechanical ability to feel comfortable learning about keyboards, control panels and the like. In addition, a computer operator must be able to use his hands easily because handling cards and tapes and operating keyboards and control panels are constant activities. Further, since the computer operator often has to lift and carry card decks, rolls of printout paper, disk packs and other bulk items, he or she needs to be generally physically fit.

The setting up and monitoring of the computer system requires constant attention to details. For this reason, the computer operator needs to be naturally alert, and should enjoy doing detailed and accurate work.

Magnetic Tape Drive

Another input device common to computer systems is the magnetic tape drive, which reads input data from magnetic tape and transfers it to the computer. You may already have guessed that the magnetic tape used in connection with computers must work something like regular magnetic recording tape. That is true. In addition, the magnetic tape drive unit works very much like a regular tape recorder. In fact, just like the average tape recorder, the magnetic tape drive can both "record" (or "write") data onto magnetic tape and "read" data from the tape.

The magnetic tape itself is always stored and handled on reels, like those pictured in Fig. 11 below. Data is carried on the magnetic tape in code in the form of magnetized spots rather than in the form of holes as in cards or paper tape.

How do the magnetic spots get "written" onto the tape? For the answer, remember what we said above about the magnetic tape units being able both to record onto and to read from tapes, and then look more closely at the illustration of the magnetic tape unit in Fig. 11.

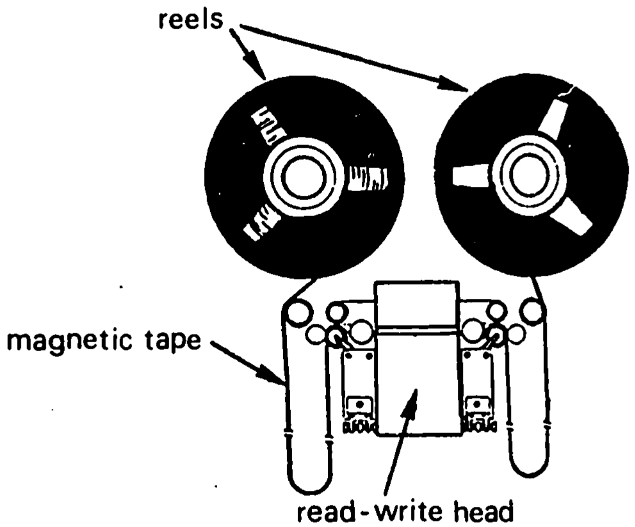


Fig. 11 Magnetic tape unit.

The computer room is a busy, sometimes hectic, place to work; there are usually hour-to-hour (sometimes minute-to-minute) changes in computer use, and work is frequently done under the pressure of deadlines. Sudden "rush" jobs may add to the bustle. Consequently, the computer operator must be adaptable and able to keep a cool head, even under pressure.

Finally, the computer operator should be able to get along well with people; he often has to work with programmers and systems analysts in scheduling and running their jobs, and with assistants in the computer room.

WORKING CONDITIONS

The computer room is the site of most of the computer operator's work, and it is usually clean, well-lighted and air-conditioned room with moderate noise level from the equipment. Smoking is generally not permitted in the computer room. The computer operator normally observes the routine eight-hour day with morning and afternoon coffee breaks and lunch hour breaks. However, it is not uncommon for special "rush" jobs or temporary computer breakdowns to make the operator's work schedule irregular.

Although the computer operator does a certain amount of work at his own desk, such as writing summary reports and filling out logs and schedules, he spends most of his working hours standing and moving about the computer in the process of setting up the equipment and monitoring its operation. In the midst of runs, he has short periods of inactivity, but these nevertheless require constant alertness because he must continually watch the equipment to be sure it is running properly.

The operator generally functions as the overseer of the computer equipment, assisting the programmers in testing their programs and making final runs of jobs on the computer. He usually reports directly to the operation supervisor who, in turn, is responsible to the manager of the computer center. In larger centers where several computer operators may be employed, there is often a "lead operator" who guides the other operators in their work.

The magnetic code is written onto the tape by the read-write head located at the center of the unit. It is called a "read-write" head because it does both the "writing" onto tape and the "reading" from tape for input into the computer. Inside the head there are usually two magnetic coils which can either sense the presence of magnetized spots (that is, "read" them), or magnetize spots on the tape's surface (that is, "write" them).

You will remember that the special machines used to punch the code holes in paper tape or cards were run from keyboards something like a typewriter's. There is also a keyboard-type of machine designed to record the magnetic spots on tape. It is a newly developed piece of equipment and, where it is installed, it is used to transcribe raw data directly onto magnetic tapes.

As yet, however, it is far more common for computer systems to use the magnetic tape drive read-write head to record data onto the medium of magnetic tape. In this case, there is no manual keyboard available. The read-write head is run electronically from the computer itself. So, if a program or some data needs to be put onto the magnetic tape, it is first put into the computer from punched cards, paper tape or any other medium, and the computer is simply instructed to output the same material on magnetic tape through the tape drive's read-write head.

It is true that magnetic tape on reels is easier to handle (and less susceptible to damage) than punched cards or perforated paper tape. In addition, a great deal more information can be stored in spot code on an inch of magnetic tape than can be stored in punched-hole code on on an inch of paper tape. So, whenever there are programs or sets of data which a computer center uses fairly often and/or needs to preserve, the material is often kept on magnetic tape. You can see then that the recent development of the keyboard-type magnetic tape writer can save time by allowing the user to skip the step of putting programs or data onto cards or paper tape, and to record the material directly onto magnetic tape. However, the tape units' read-write head will continue to be useful as a recording instrument. The reason is that often the results of computer runs need to be kept for future use; using the read-write head, the computer can output its results directly onto magnetic tape, which is more durable and easier to handle than paper media.

Now that you know the essentials of recording onto the medium of magnetic tape, let's return to our main topic, that of input devices, and look at the magnetic tape drive in this light.

COMPUTER OPERATION AS A CAREER

As more and more organizations are installing computers or using computer services, the demand for computer operators is generally increasing. This is particularly true in the metropolitan areas where the largest businesses and industries are located and where the turnover in operating personnel is relatively high.

In the larger cities, candidates for a computer operator's job may apply to any number of different kinds of organizations for possible employment: colleges and universities, data processing firms, large corporations or businesses, banks, insurance companies, city, county, and state governmental offices, the offices of major utilities, and so forth. Applicants may also want to consider taking positions as input/output clerks and decollators, tape librarians or scheduling clerks, for which they are usually well qualified.

The current salary range for computer operators is from about $95 to $200 per week,* depending on the size of the organization and the experience and capabilities of the operator.

There are several directions a computer operator's career can take. Many successful computer operators choose to continue working directly in computer operations, where changes and improvements in equipment allow them to keep expanding their knowledge and skills. In larger centers, the operator may move into the position of lead operator and, if he has supervisory abilities, he may move upward to operations supervisor. Other accomplished computer operators move into programming careers, preparing themselves either through outside programmer training courses or through on-the-job training.

On the whole, the successful computer operator has good chances for promotion; his knowledge of computer equipment and how to operate it effectively provide a good base from which to learn other specialties in the data processing field.

*U.S. Bureau of Labor statistics, 1971.

Computer centers often have a library of master magnetic tapes -- tapes encoded with often-used programs and data. When a computer run calls for the input of material from these tapes, the operator simply locates the tape in the tape library and mounts it on the magnetic tape drive. When it needs the material, the computer can then activate the tape drive's read-write head to "read" the coded spots and transfer the data electronically into the computer for use. The computer itself has complete control over activating the tape drive to input (or "access") material. But, the computer does so only according to instructions. These instructions come from the program being run or from the special instruction cards* for the run which are given to the operator by the programmer.

Most magnetic tape units look something like the unit pictured in Fig. 12, and are set-up and threaded very much as regular tape recorders are. Instructions for setting-up individual tape drives are always available in the manuals accompanying the given unit.

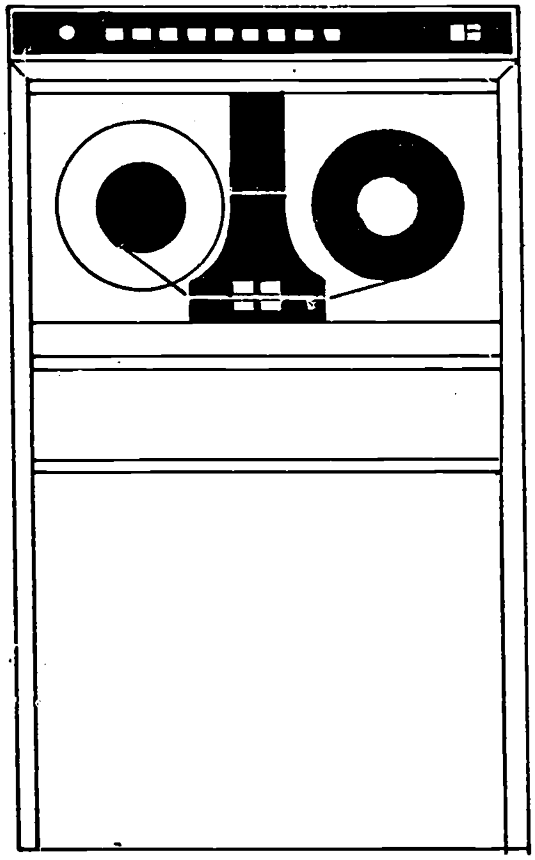


Fig. 12 Typical magnetic tape drive.

*These cards are most often referred to as "job control cards," and they are commonly used to tell the computer how to prepare for a run.

Check Your Understanding

Put a check (✓) in the box(es) beside the correct answer(s) to the following questions.

1. Which of the following tasks are commonly performed by the computer operator?

   □ a. Preparing data on punched cards or paper tape.
   □ b. Scheduling jobs on the computer.
   □ c. Writing programs.
   □ d. Running jobs on the computer.
   □ e. Setting up the computer equipment for each job.
   □ f. Monitoring the computer's control panel.

2. Which of the following skills must be developed to become a competent computer operator?

   □ a. Communication skills.
   □ b. Arithmetic skills.
   □ c. Programming skills.
   □ d. Skills in handling input and output media like cards and tapes.
   □ e. Skills in repairing machines.
   □ f. All of the above are needed.

3. The minimum education required for a computer operator's job is which of the following?

   □ a. a college degree
   □ b. a high school deploma
   □ c. a master's degree
   □ d. two years of high school

4. Which of the following are characteristics of a good computer operator?

   □ a. Dislikes commotion or noise on the job.
   □ b. Is physically fit.
   □ c. Has natural mechanical ability.
   □ d. Is naturally inclined to ignore details.
   □ e. All of the above are characteristics of a good computer operator.

□ a. Dislikes commotion or noise on the job.
□ b. Is physically fit.
□ c. Has natural mechanical ability.
□ d. Is naturally inclined to ignore details.
□ e. All of the above are characteristics of a good computer operator.

Sometimes in a computer center you may see a tape drive inputting data from a magnetic tape, but you may notice that the tape reels spin for a minute, then stop, and then spin again, sporadically. (In like manner, you could see cards in a card reader or paper tape on a paper tape reader appear to run, then stop, then run again.) What is happening?

Something very simple! In these cases, the computer is running a program (which has already been loaded into its memory) which calls for data to be "read in" by the input device one set at a time. So, the computer program calls on the input device to read in one set of data, to stop while that data is being processed and then to read in another set.

When a complete program or batch of data (called a "data file") is being input all at once by a magnetic tape drive, however, you can observe one of the things which make magnetic tape and the magnetic tape drive so useful for inputting data into the computer -- their speed!

There are two input devices, however, which are even faster than a magnetic tape drive. They are the magnetic disk drive and magnetic drum drive.

Magnetic Disk Drive

The magnetic disk drive is used to input data from magnetic disks which look very much like phonograph records.

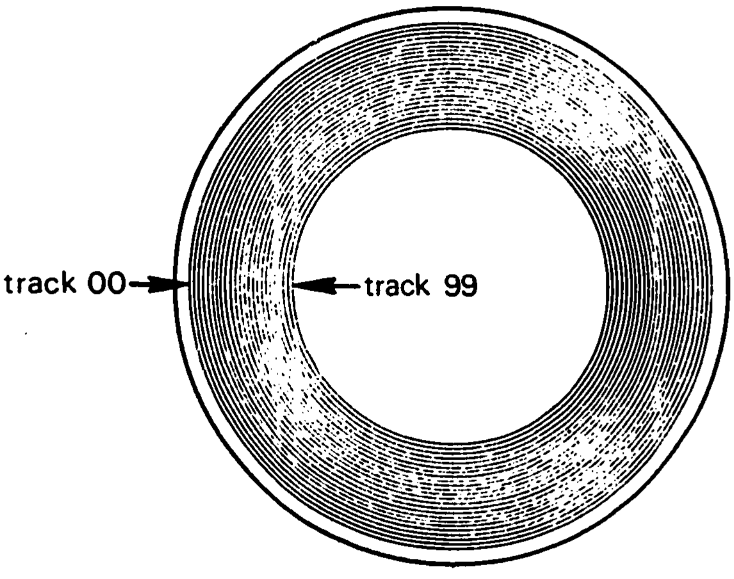


Fig. 13 Magnetic disk, showing recording tracks.

However, if you look careful at the disk illustrated in Fig. 13, you will notice that the grooves on the disk actually form circles or tracks rather than spiralling in a continuous line toward the middle, as grooves on records do. One disk, in fact, can have hundreds of tracks on each side. Data is recorded on these tracks on code in the form of magnetic spots. The coded spots are almost microscopically small, and even more information can be coded onto one side of a disk than onto an entire reel of magnetic tape.

To make them even more useful and easy to handle, disks are stored and worked with in "packs" -- that is, several disks are stacked together in one pack with a fixed distance of an inch or so between each disk.

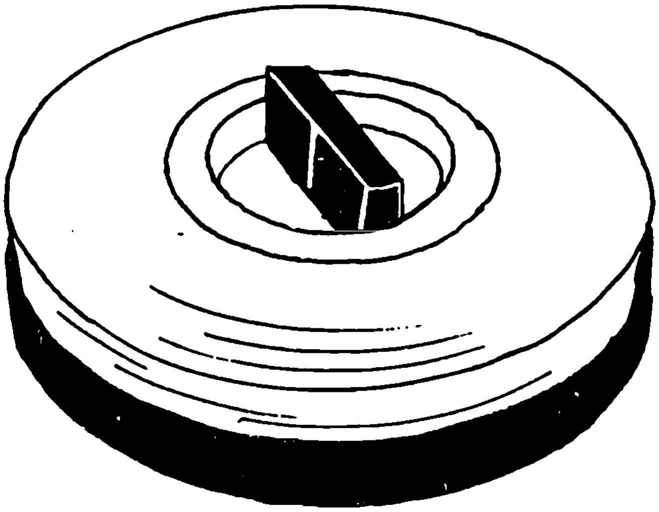


Fig. 14 A disk pack.

Since the surfaces of these disks have a magnetic coating onto which the coded spots are written, you can understand that care is needed to protect the surfaces from damage. For this reason, packs are kept in plastic covers when not in use. An added protection is given the disks in disk packs by leaving the two exposed disk surfaces (the top surface of the top disk and the bottom surface of the bottom disk) blank.

5. The conditions in which the typical computer operator must work can be best described by which of the following?

□ a. A clean, air-conditioned room with moderate noise control.
□ b. A cluttered and noisy work area.
□ c. A quiet, private office atmosphere.
□ d. A quiet, tidy, dimly-lit work area.

6. The current salary range for computer operators is

□ a. $50 to $150 per week.
□ b. $95 to $200 per week.
□ c. $110 to $250 per week.
□ d. $130 to $275 per week.

7. Which of the following positions are usually open to the competent computer operator?

□ a. Computer Center Manager
□ b. programmer
□ c. lead computer operator
□ d. operations supervisor
□ e. systems analyst

Check your answers using the key at the bottom of page 13.

PART II

COMPUTER HARDWARE

How is the magnetic spot code "written" onto magnetic disks? By read-write heads on the magnetic disk drive unit. These heads are situated in the drive unit so that there is one read-write head for each disk surface in the regular disk pack.

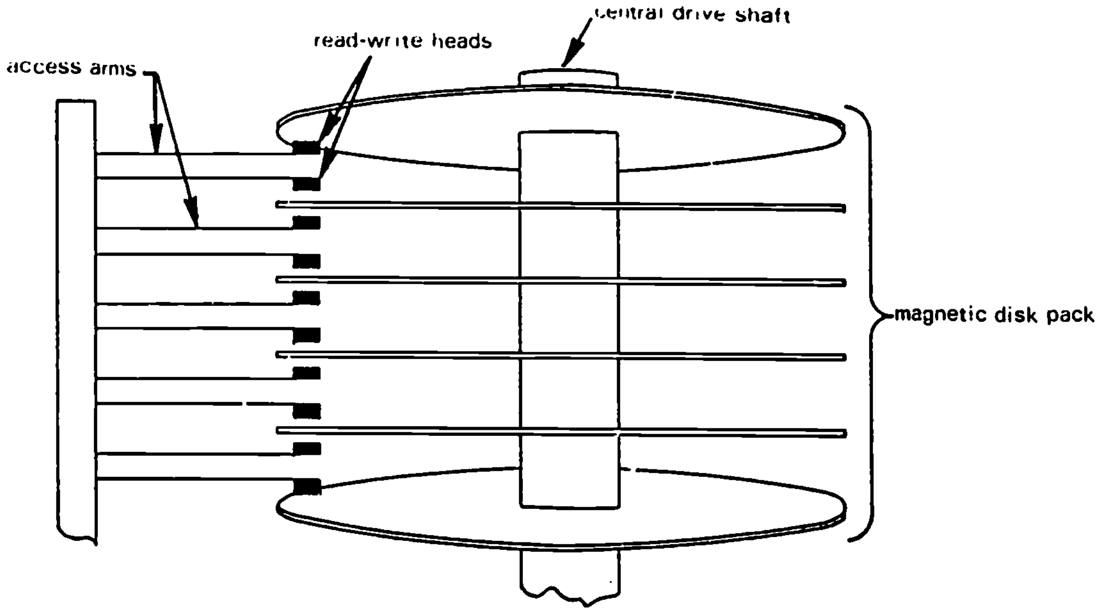


Fig. 15 Mounted disk pack with read-write heads in position.

The access arms can be moved in or out so that the heads are at any chosen track position, and the head for the desired surface can then be activated to magnetize coded spots (representing data) on the correct track of that surface.

The read-write heads of the disk drive are run electronically by the computer. So, to put new programs or data onto magnetic disks, the material must first be fed into the computer on punched cards, paper tape, or other input meida. The computer can then be instructed to output the same material on magnetic disk through the disk drive's read-write heads.

As is the case with transcription equipment for magnetic tapes, a manual keyboard-type machine has been recently developed for recording raw data onto disks. By skipping the step of putting data first onto another medium, the keyboard machine can save much time; where new programs or data need to be put onto disks for quick and easy use in the future, they can be encoded directly onto a magnetic disk with this machine. But, once again, it is often the case that results from the computer also must be stored for frequent use in the future. In these cases magnetic disks are often chosen as the storage medium, and it is very convenient to be able to instruct the computer to output

PART II COMPUTER HARDWARE

INTRODUCTION TO PROCESSING AND HARDWARE

As you saw in Part I, the computer operator is a key figure in the modern setting where computers have taken over so much of the information (or data) processing. So let's take a minute to get a clear idea of what "data processing" is before turning our attention to the machines (or "hardware") involved in it.

From the description in Part I of the computer operator's work, the three basic elements in data processing are probably clear to you. They are:

1. A program* is put into the computer and then the data** is put in.

2. The computer does what it is programmed to do to the data.

3. The results of the computer's work are typed or printed out.

These elements can be diagrammed this way to illustrate the main steps involved in processing data:

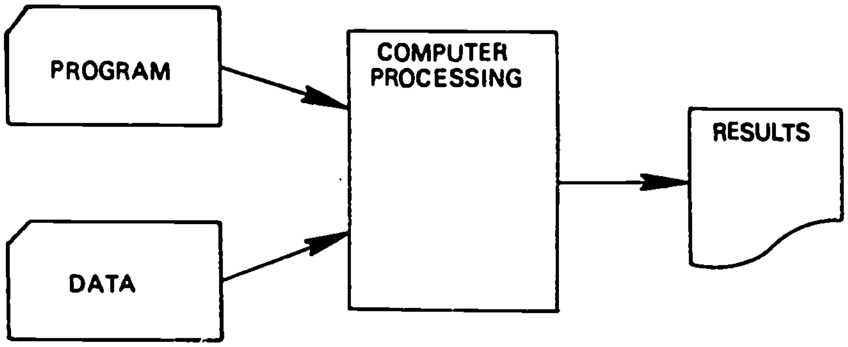


Fig. 1 The data processing flow.

*A program tells the computer what to do.
**The data is the information the computer is to do something to.

Answers to questions on pages 9 and 10:
1. (b, d, e, f) 2. (a, b, d) 3. (b)
4. (b, c) 5. (a) 6. (b) 7. (b, c, d)

its results directly onto a magnetic disk through the disk drive's read-write heads.

Large computer centers often have considerable material stored on magnetic disks, including their most frequently used programs and data. When a computer run calls for input which is stored on a disk pack, the computer operator simply removes the correct pack from the storage room and mounts it on the input device (the disk drive) very much as a stack of records would be mounted on a record player.

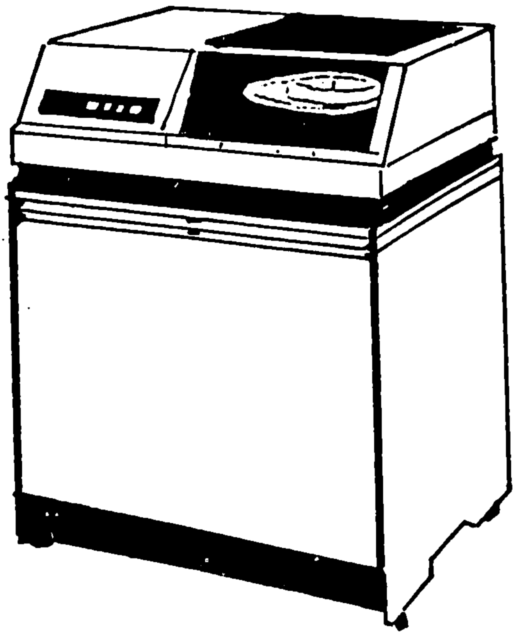


Fig. 16 A typical dtsk drive.

When the computer needs the data, it directs the drive unit to move the access arms on the read-write assembly to the correct track position and to activate the read-write head for the correct surface to "read" the coded data into the computer. As in the case of the magnetic tape drive, the computer alone can access (or locate) data on disks through the disk drive. The computer does so according to instructions given it in the program it is running or on the job control cards for the run.

From the description of accessing data above, you can probably see the feature of the disk pack which makes it a faster input medium than magnetic tape. To select out any given data on a magnetic tape, the entire tape must be run, foot by foot, through the read-vrite head until the place of the correct data is reached. To locate the given data on a magnetic disk, however, the access arms on the read-write assembly need only to be moved to the correct track position, and the needed head activated.

To get a more concrete idea of what is actually involved in these steps consider the simple example of processing some students' names to get a list of the names in alphabetical order. First, a program telling the computer how to alphabetize is loaded* into the computer.

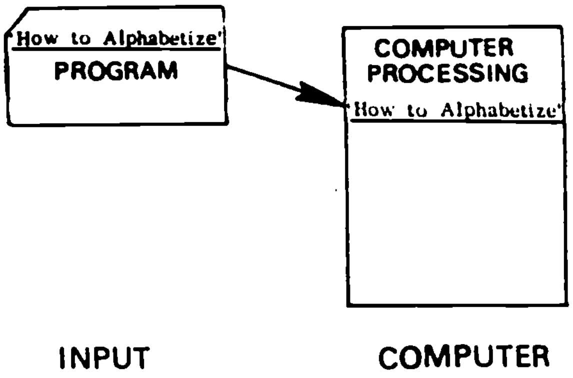


Fig. 2 Loading the program.

Now, put in the students' names and get the alphabetized list:

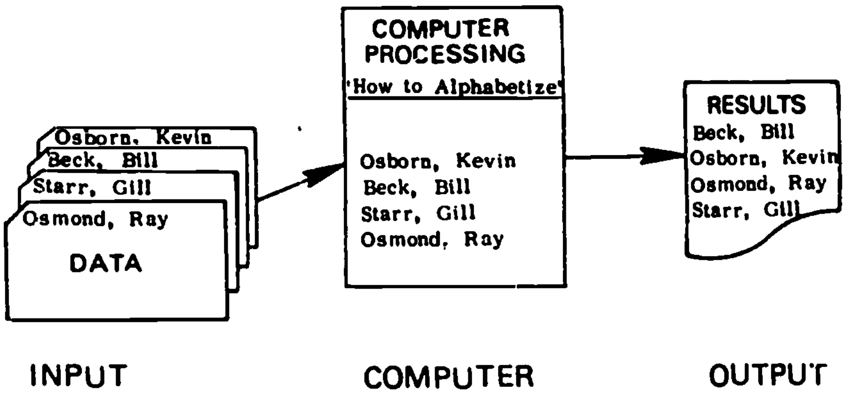


Fig. 3 Processing the data.

In every case, data processing involves the input of some original data, its processing by the computer, and the output of new, usable information. This is true whether student grades are being input to get an honor roll list or measurements are being input from instruments in a moon capsule to get an analysis of its flight condition. And, in every case, computer operators oversee the actual processing of the data.

*As you will learn later, a small set of instructions (called job control statements) are entered before each program. These job control statements are needed to tell the computer how to run the program.

Can you see that the programs and data recorded on the hundreds of tracks on the ten disk surfaces in the pack can be reached (or "accessed") immediately with just a shift of the read-write head assembly?

If a large computer center has a tremendous amount of data which it uses almost constantly, you can see that the computer operator would be better off if the data is stored in disk packs rather than on magnetic tapes. Still, disk packs would have to be located, mounted, and dismounted. One input medium even more convenient for the rapid input of continually used programs or data is the magnetic drum.

Magnetic Drum Drive

A magnetic drum is a metal cylinder coated with magnetic material. While it may be of almost any size, the typical magnetic drum is about the size of three disk packs stacked one on top of the other. Data is recorded as magnetized spots in circular tracks or bands around the surface of the drum, as shown below.

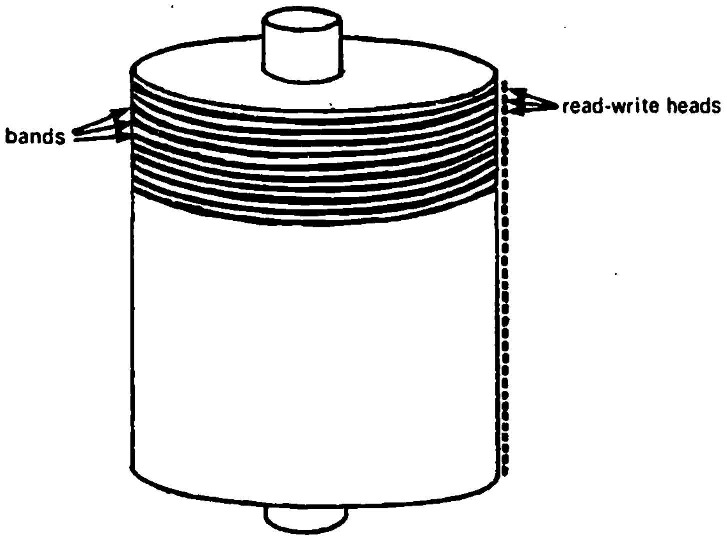


Fig. 17 Magnetic drum showing read-write heads.

Data is always recorded onto and read from magnetic disks by the read-write assembly in the magnetic drum drive unit. This assembly is actually a series of read-write heads, one for each band on the drum. The heads are positioned near the surface of the drum to record or read magnetic spots as the cylinder rotates. Since there is one read-write head for each band, all the bands on the drum can be

Now, let's turn to the equipment (called "hardware") the computer operator uses in this data processing.

To handle the three basic steps in data processing, there are three basic kinds of computer hardware that make up a data processing system:

1. input devices,

2. the central processing unit, and

3. output devices.

Our simple diagram of the data processing flow in Fig. 1 above can be expanded, now, to show just where these pieces of hardware fit.

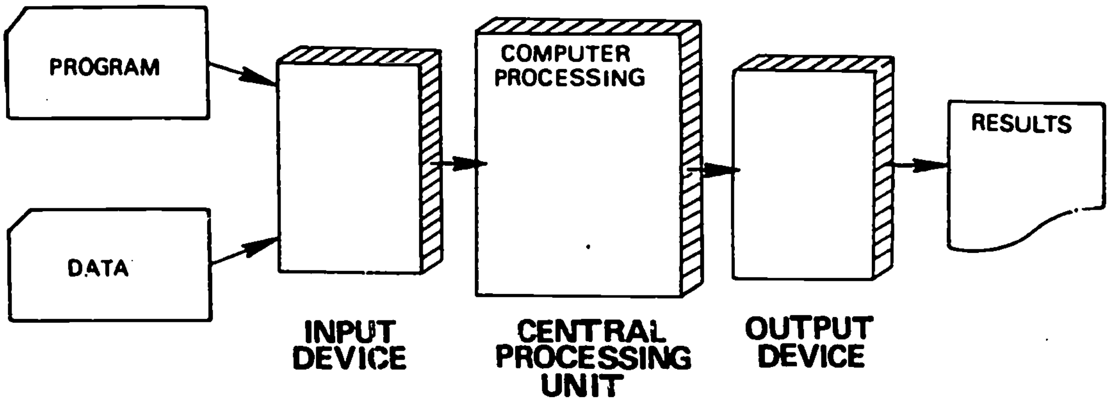


Fig. 4 The computer hardware.

Had you been wondering how the programs and data on paper tape or punched cards got into the computer and how the results got out? It's a good question. And now you can see something of the answer. There are various special devices designed to read tapes or cards and to transfer the information they carry into the central processing unit. In addition, special devices are used by a computer system to give out the results of the computer's processing in the form desired, whether in print, on tapes, or displayed on a screen. We'll consider these input and output devices in turn and then we'll examine the central processing unit with its all-important panel and console typewriter.

scanned simultaneously. Any particular data, therefore, can be located within one swift revolution of the drum. It is this feature of the drum that makes it a much faster medium than the disk, and also more expensive.

From the above description you can see that the magnetic drum would be a harder medium to carry around, store, mount and dismount than a disk or magnetic tape. This is true not only because it is often very large, but because its entire cylindrical surface, which carries the data, is highly delicate and entirely exposed. For these reasons, the magnetic drum is ordinary mounted permanently on the drive unit, where it is protected by a vacuum tight cover. Because the magnetic drum is left always in position, computer centers need to make certain that they are installing a drive unit with a drum sufficiently large to carry all the data they need to have on it.

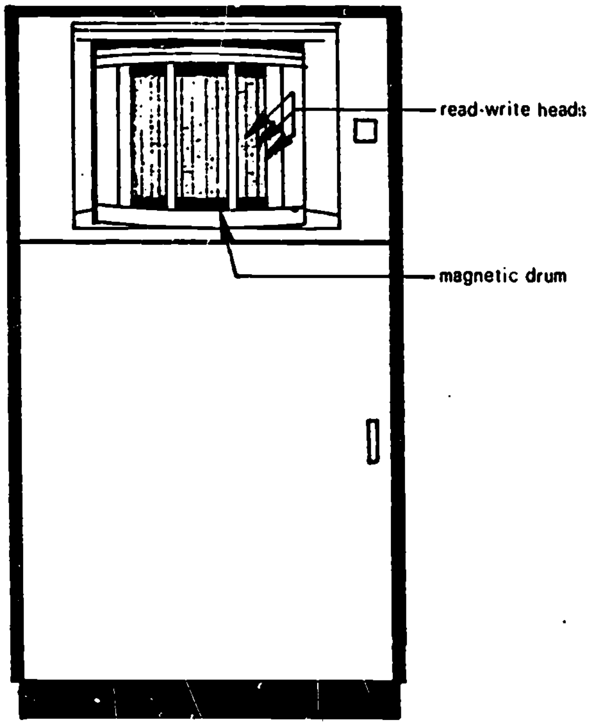


Fig. 18 A magentic drum drive.

Once a magnetic drum is encoded with a center's most constantly used programs and data, it is ready on the magnetic drum drive for input at any time. When a particular program or data file on the drum is needed for a run, the computer directs the read-write heads to scan

INPUT DEVICES

As you saw in Part I of this manual, programs and other input can come in various forms -- for example, on punched cards, magnetic or paper tape, magnetic disks, and so forth. For each kind of input medium there is a specialized input device which can "read" the information on the medium and transfer it to the central processing unit.

Punched Card Reader

Let's first consider the most commonly used medium, that of punch cards, and their related input device, the punched card reader. Punched cards, as the illustration below shows, commonly have eighty columns of figures divided into twelve rows.

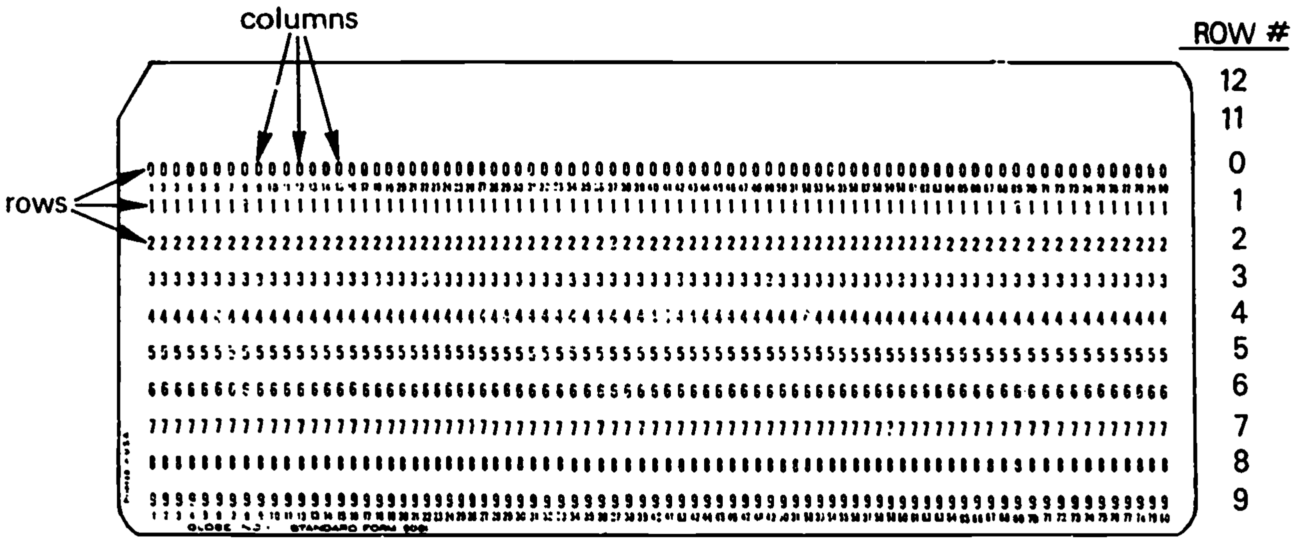


Fig. 5 Hollerith punched card.

Data is recorded on such cards in code form by punching holes. Key punch machines are used to punch the cards; generally speaking, key punch machines are operated from a keyboard something like a typewriter's and when a key is pressed, the machine automatically punches the card with holes that represent the character on that particular key. Each column on a card can carry the punched-hole code for one number, letter or symbol. As an example, the holes punched in the card on the next page are punched in the commonly used Hollerith code and represent the statement "THIS IS A SAMPLE CARD."*

*To learn more about the key punch machine and Hollerith code, refer to pages 77 to 84 of the manual THE ELEMENTS OF DATA PREPARATION.

the tracks for the correct material as the drum rotates at high speed. Once it is located, the coded material is "read" by the correct read-write head and is electronically transferred into the computer for use. The data on a magnetic drum can be accessed for input only by the computer, according to instructions given it by a program, job control cards, or the console typewriter.

Mark-Sense Reading Devices

A less common input device is the mark-sense card or sheet reader. From the illustration below you can see that this kind of input device has the same basic features as the punched card reader; that is, an input hopper, read station, and stacker.

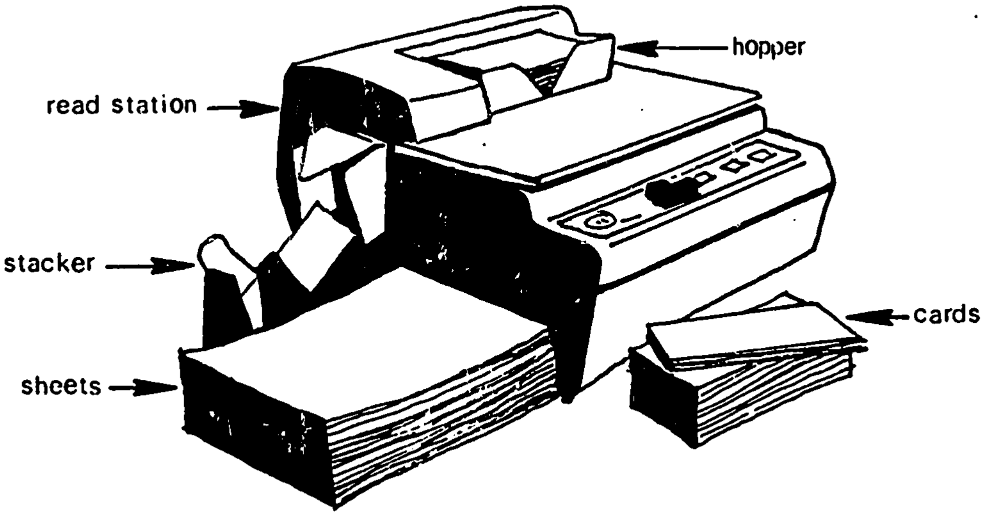


Fig. 19 Mark-sense reading device.

The average mark-sense reader can be adjusted to "read" from either cards or sheets which have data encoded on them by pencilled-in spots (the mark-sense method). Incidentally, these mark-sense spots usually follow the traditional Hollerith code sequence used for punching cards. (See pages 16 and 17 above.)

On the facing page is an example of a typical mark-sense card with the instruction ":ABORT," which is a command used to stop a computer run.

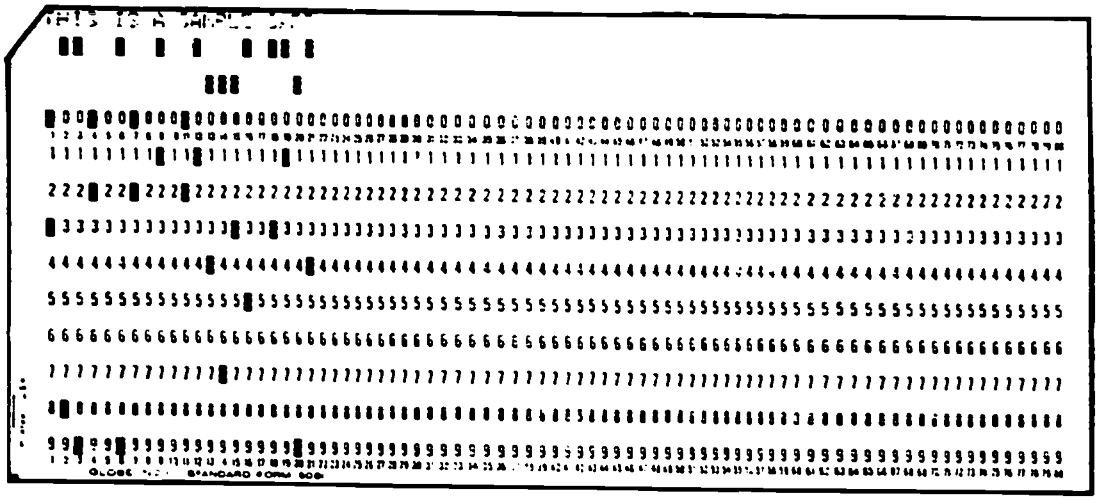


Fig. 6 Card punched in Hollerith code.

The device used to "read" punched cards is called, naturally enough, a punched card reader. While there are many different models of punched card readers in common use, most of them resemble the one illustrated below.

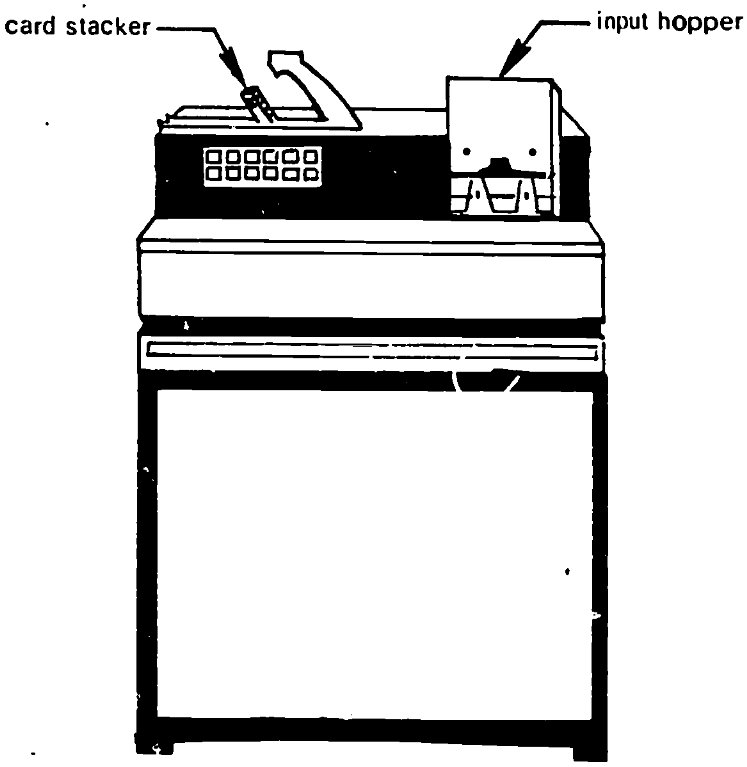


Fig. 7 Example of a card reader.

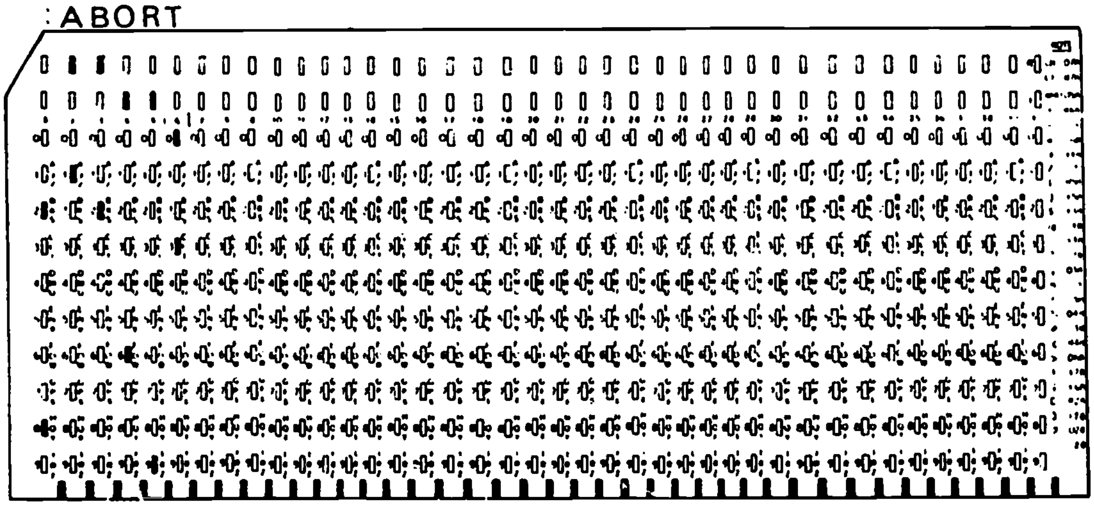


Fig. 20 Mark-sense card.

Marked cards or sheets are loaded into the input hopper in preparation for their use in a run. When the data on the cards or sheets is needed, the program stored in the computer directs the computer to activate the read station, and the cards or sheets are run through, one at a time, and "read." Usually, it "reads" by means of photo cells which register light reflected from each card; the marks reflect no light and hence are picked up by the photo cells as blank.* These registered blanks are then transmitted electronically to the computer for use. Once read, the card or sheet is deposited into the stacker. In this way, an entire file of mark-sense cards or sheets can be loaded into a reader and retrieved in an orderly stack.

As with all reading devices, there are numerous different styles and makes, and each is accompanied by a manual describing how to adjust and use the particular model.

*Punched cards can also be read by most mark-sense readers; their punched holes, like pencil marks, do not reflect light and are therefore registered as blanks. Punched card readers, however, can read punched cards much faster and are consequently nearly always used with the punched card medium.

Console Typewriter

Most computers rely on the several input devices described above for input of programs and data files for processing. There is one other input device common in the computer room, however, which is of considerable importance to the computer operator. It is the console typewriter. This input device is a part of the computer console from which the computer operator controls the computer's operation.

The console typewriter is used by the computer operator mainly to communicate directly with the computer to control its operator. You will find a complete description of the console typewriter in the section on the computer console below, pages 58 to 60.

Teletypewriter

One final common input device which the computer operator should know about is the teletypewriter. It is the one device, however, that is seldom found in the computer room. The teletypewriter (or "Tty," as it is called) is the device used specifically to input data or programs into the computer from a remote (or distant) station.

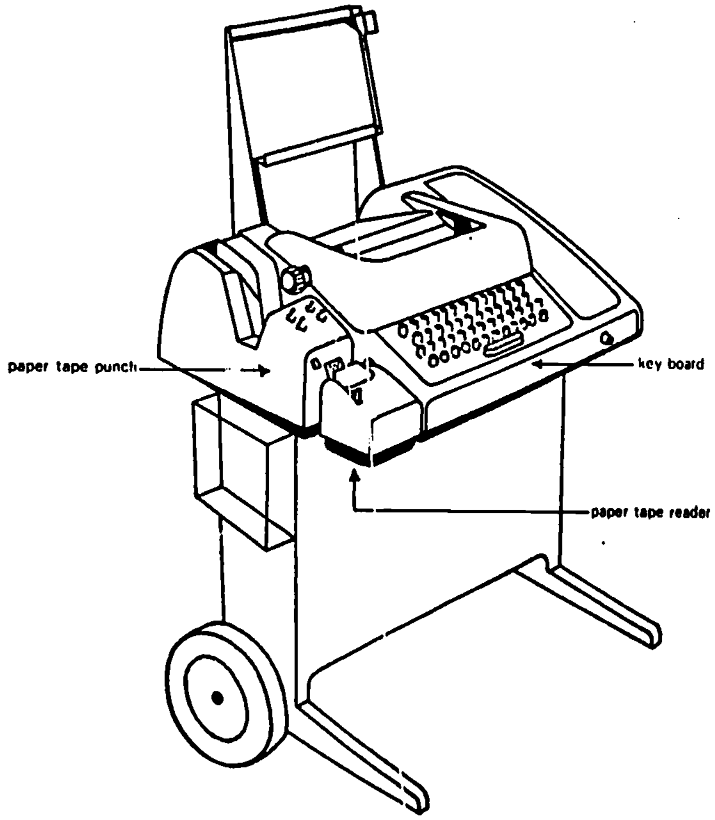


Fig. 21 A teletypewriter.

Teletypewriters (or Tty's) most often make connections with a computer by a conventional telephone line. The telephone number for the computer is dialed and, when the answering high-pitched sound is heard, the telephone receiver is placed on the coupler attached to the Tty. The coupler also emits a high-pitched sound, which is picked up by the receiver and transmitted to the computer. In that way, the connection between the two machines is made, and they can now transmit information to each other in code in the form of patterns in their respective sounds.

The Tty looks very much like a typewriter, having a similar keyboard with a platen above it which feeds paper through to be typed on. When a key is pressed, the letter, number or symbol designated by that key is typed on the paper. If the Tty is connected to (or "on-line with") the computer, the letter, number or symbol will be simultaneously transmitted directly to the computer. All information transmitted back from the computer to the Tty is translated by the Tty into the corresponding type characters and is typed out on the Tty paper for users to read.

As a remote device, the Tty actually functions as three machines in one. First, not only does it work similarly to the console typewriter but when "on-line," it can give operating instructions to the computer very much like the console typewriter does. For example, by typing appropriate commands on the keyboard, the Tty user can direct the computer to locate a certain program in storage, to process data typed in on the keyboard, and to end the run.

Secondly, when equipped with a paper tape reader as in Fig. 21, the Tty's can be used "on-line" to input data or programs from paper tape into the computer.

Finally, when equipped with a paper tape punch, the Tty can be used "off-line" to punch any paper tapes needed, or "on-line" to punch tapes of data or programs being output by the computer for storage.

Check Your Understanding

Put a check (✓) in the box(es) beside the correct answer(s) to the following questions.

1. The data processing flow follows this order:

   □ a. from programs to computer processing to input to data.
   □ b. from data and programs to output to computer processing.
   □ c. from programs and data to input to output to computer processing.
   □ d. from programs and data to input to computer processing to output.

2. Holes in Hollerith code are punched onto punched cards by a

   □ a. teletypewriter controlled by the computer.
   □ b. read/write head on a drive.
   □ c. keypunch operator on a keypunch machine.
   □ d. none of the above.

3. Paper tape is commonly used

   □ a. to mend tears in magnetic tape.
   □ b. to input data through a paper tape punch.
   □ c. to input data through a paper tape reader.
   □ d. none of the above.

4. A magnetic tape drive usually has

   □ a. multiple access arms.
   □ b. an electronic punching device.
   □ c. a head which reads and writes magnetic spots.
   □ d. a permanent drum mounted on it.

5. A disk pack is

   □ a. read by read/write heads on access arms.
   □ b. mounted on a card reader.
   □ c. a pack of separate tape reels.
   □ d. written on by a drum drive.

6. Data can be accessed most quickly from

□ a. a magnetic drum.
□ b. punched cards.
□ c. a magnetic tape.
□ d. any of the above.

7. A teletypewriter is most often used as an input device

□ a. by the computer operator in the computer room.
□ b. by someone at a remote station.
□ c. by a keypunch operator to prepare punched cards.
□ d. by none of the above.

8. The standard drum drive has

□ a. a keyboard for use in encoding data onto disks.
□ b. one large magnetic read-write head.
□ c. several sizes of drums mounted on it.
□ d. a read-write head for each drum band.

9. The computer operator is responsible for

□ a. operating all input devices by hand.
□ b. deciding by himself what medium to use to input data for each run.
□ c. loading input devices with the appropriate input media and overseeing their operation.
□ d. all of the above.

10. The disk drive's read-write head can be activated to encode data on disks

□ a. by the computer only.
□ b. by the drive operator only.
□ c. by the keypunch operator only.
□ d. by the computer operator only.

(Check your answers using the key on the bottom of page 36.)

LABORATORY -- OPERATING INPUT DEVICES

In this laboratory exercise, you will have the chance to operate the input devices that are used in your computer center. So, you should begin by arranging with your teacher to go to your center. Then, follow the sequence of activities below.

1. Look over the hardware in your computer system and identify all the input devices used by the system. Put a check (✓) beside each device listed below that is used in your system.

    ____punched card reader

    ____teletypewriter

    ____paper tape reader

    ____magnetic tape drive

    ____magnetic disk drive

    ____magnetic drum drive

    ____other (specify): ______________________________

    (Check with your teacher or computer operator to be sure you have correctly identified all the devices used.)

2. Obtain the operator's manual for each input device and return to your class room to study each booklet carefully. When you think you understand how to operate each device proceed to number 3 below.

3. Schedule a time with your teacher or computer operator when you can have his help in setting up the input device(s) for operation. Also, ask your teacher or operator to provide you with some sample input to be used in your trial run of the machines.

4. With your teacher or operator standing by, load the sample input and start the input device according to the directions given in the manual. The teacher or operator will be responsible for operating the computer for you once your input device is ready. As the run proceeds, watch your input device and see how it operates.

Answers to questions on pages 34-35:
1. (d) 2. (c) 3. (c) 4. (c) 5. (a)
6. (a) 7. (b) 8. (d) 9. (c) 10. (a)

5. When the run is finished, retrieve the sample input and switch the device off. Return the sample input to your teacher or operator.

   (Repeat steps 4 and 5 for each device you have access to in your center.)

6. If you have any trouble operating the input device, go back to study the operator's manual more carefully. Then begin at number 3 above, again.

7. When you have successfully operated the input device(s) available to you, return to your classroom. Use the space provided on the next pages to list the steps (in the proper sequence) that you went through in setting up and operating each device.

Input device ______________________________

Input device ______________________________

(Use the next page if you need more room.)

Fig. 22 Steps for setting up and operating input devices.

Input device ______________________________

Input device ______________________________

OUTPUT DEVICES

Once information has been input to the computer and processed, the resulting new information is output on an output device. In every case, the computer is connected directly to the output devices. When a program so directs it, the computer activates the appropriate device and transmits the output to it electronically. Depending on the device, the output is produced in the form of print on paper, punched holes in cards or paper tape, code on magnetic media, or even an image on the television-like screen of a cathode ray tube. There is a separate output device which produces each of these kinds of output.

As in the case of input devices, it is the computer operator's job to oversee the use of the output devices. For each computer run, the operator sets up or adjusts the appropriate devices so that they are ready to produce the output when the computer activates them. In general, the operator makes sure the devices are in good running order and are loaded with the correct output medium (paper, cards, tape, etc.) for the run.

Each model of each output device, of course, has special features and procedures for running it. These are usually outlined in an operator's manual or guidebook accompanying the device. To familiarize you with the basic features of the most commonly used output devices, however, let's take a general look at them, one at a time.

Line Printer

The most commonly used output device is the printer. It is usually called a "line printer," because it nearly always prints one complete line of information at a time. When you consider that a line of print can have up to 100 or even more characters in it, you can readily see that a line printer operates at a much greater speed than typewriter-like devices which can only print one character at a time. In fact, an average line printer can print at a speed of ten complete lines per second -- or even faster. (How many "characters per minute" can be printed at a speed of ten 100-character lines per second?

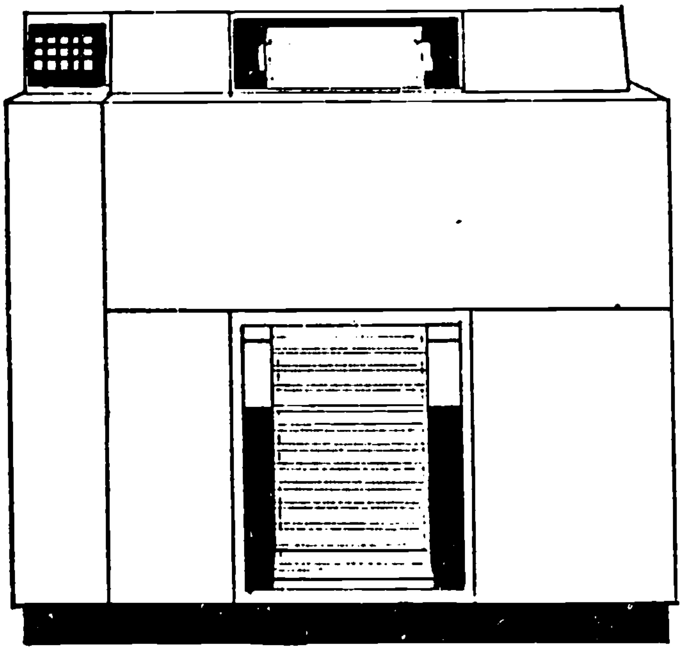
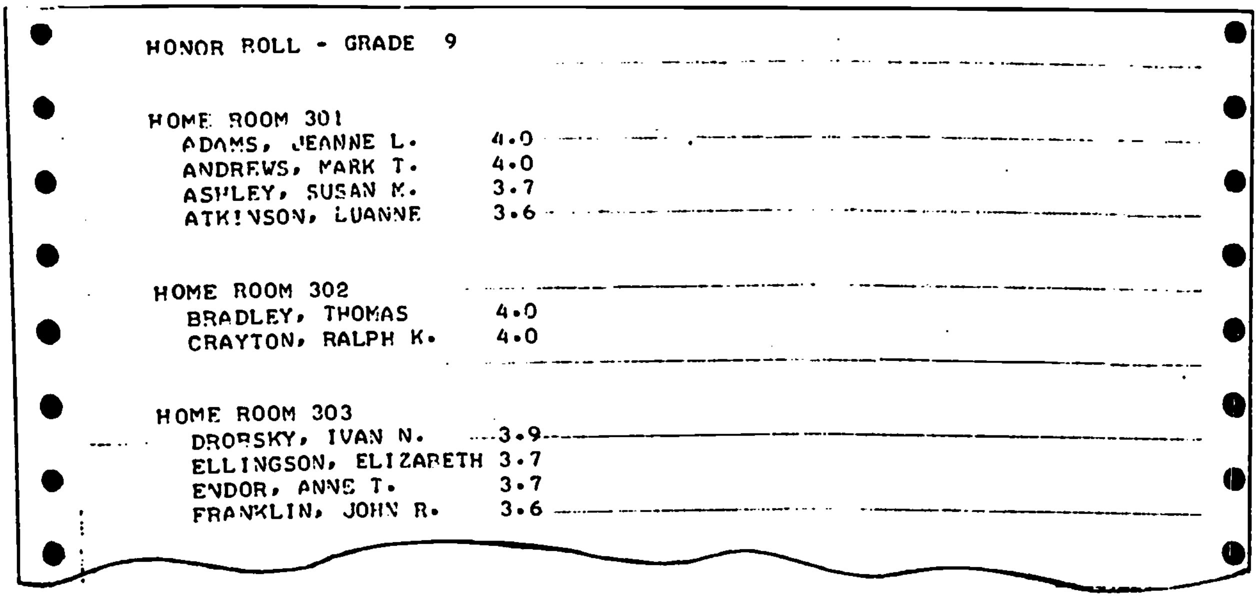


Fig. 23 A typical line printer.

The printer is always under the control of the computer which sends it electronic impulses of information. These impulses cause the printer to print out the information in the regular letters, numbers, or characters that people can read. The printout paper used by the conventional printer comes in various size rolls with holes (for notebooks or other binders) at both edges. The paper on the roll is continuous, but the page lengths are marked off by tiny perforations for easy tearing later.

The printer not only will print as directed by the computer, but it can detect when the perforations marking a page's end have been reached; it will automatically skip over the perforations and begin printing again on a fixed line (so many lines down from the perforations) on the next page. The printer can also detect when the end of the paper roll has been reached. At this point the operator must load a new roll.



```
HONOR ROLL - GRADE  9

HOME ROOM 301
  ADAMS, JEANNE L.      4.0
  ANDREWS, MARK T.      4.0
  ASHLEY, SUSAN M.      3.7
  ATKINSON, LUANNE      3.6

HOME ROOM 302
  BRADLEY, THOMAS       4.0
  CRAYTON, RALPH K.     4.0

HOME ROOM 303
  DROBSKY, IVAN N.      3.9
  ELLINGSON, ELIZABETH  3.7
  ENDOR, ANNE T.        3.7
  FRANKLIN, JOHN R.     3.6
```

Fig. 24 Output from a conventional line printer.

There are two basic types of line printers -- impact printers (which print using a type-bar or wheel pressed against the paper), and non-impact printers (which form an image by chemical or other nonimpact means). Almost all printers use an impact mechanism and produce output which looks very much like that shown in Fig. 24 above.

The most common impact printer is the chain printer. In this printer, the characters are in a circular steel chain which moves horizontally in front of the paper. Hammers in back of the paper press the paper forward against the type-characters on the chain, thus impressing the character onto the paper.

Only one revolution of the chain is required to print a line. If an "E" is to be printed 28 times in one line, the "E" character on the chain will be printed each time it passes in front of a print position on that line where an "E" is required. A hammer behind the paper presses the paper against the type face to print the character.

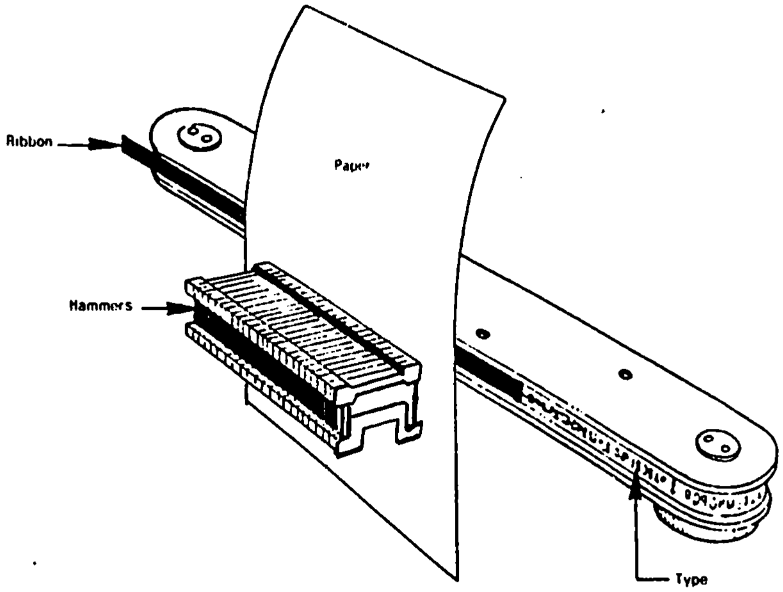


Fig. 25 The chain printer mechanism.

Another common impact printer uses a metal drum engraved with rows of characters. The drum revolves during printing, moving each row of characters past the print hammers. As the proper characters pass by, the hammers press the paper against the drum to print.

A type-wheel printer is another impact printer, with each printing character on an individual metal print wheel. Still another impact printer forms each character as a pattern of dots imprinted by the ends of small wires. As you can see, each of these printers works in essentially the same way.

The less common nonimpact printers operate with chemically impregnated paper energized by pulses of current or dusted with a deposit of electrostatically charged ink. Nonimpact printing techniques are available which print as high as 31,250 lines per minute. But you can imagine how expensive such a printer would be! (Do you think the cost would be justified by the extemely high speed in some cases -- for example, for certain kinds of output in a space flight project?)

Line printers are usually quite versatile and can be adjusted to print on various different sizes and weights of paper. They are commonly used, for example, to print reports, checks, report cards, invoices and just about any other type of output that can be printed on paper or forms.

Since it is most common for computer systems to use a printer for a major portion of its output, the computer operator usually becomes quickly familiar with loading and adjusting the printer in his system.

Card Punch

There are times when the output from a computer is data which needs to be stored on punched cards -- for example, when the data will be used at a later time as input data.

You are already familiar with the medium of punched cards, which you read about on pages 16 and 17 above. There you also learned about the keypunch machine which the human operator can use to punch data onto cards for input. As you saw, the cards are input by a card reader connected to the computer. But, how can a computer get its output punched onto cards for future use?

The device which the computer uses to output information on punched cards is, naturally enough, simply called a card punch. It is always connected directly to the computer. Frequently the card punch is combined with a card reader, as is shown in the illustration below. When the two devices are combined, the unit is usually called the card read punch.

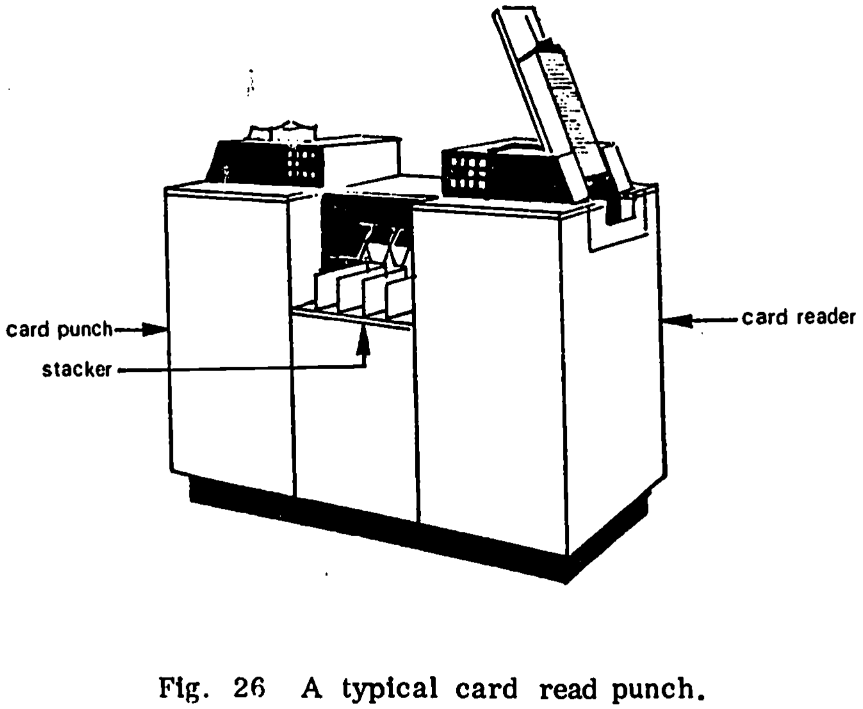


Fig. 26 A typical card read punch.

Whenever a computer run calls for output to be produced in the form of punched cards, the operator is responsible for loading the card punch hopper with blank cards and for readying the device for use when the computer calls on it. When the computer activates the device, the cards are automatically drawn, one at a time, from the hopper through the punch station where the data is punched in coded holes onto the card. Once punched, the card is deposited in the unit's stacker.

Since they are punched and depcsited one at a time, the cards are always kept in order. When the run is through, the operator has only to remove the complete card deck and store it or route it to the programmer, depending on what the instruction sheet specifics.

Paper Tape Punch

You will remember from the discussion on pages 18 and 19 that paper tapes can be punched by a machine operator using a paper tape punch or a teletypewriter. Both of these devices are equipped with keyboards on which the operator may type the data to be punched.

When the computer needs to output data in the form of paper tape, it most often uses the output device called a paper tape punch, which is connected directly to the computer. Once the computer operator has loaded the paper tape punch with a roll of blank tape, the computer can activate the punch at anytime. There is, of course, no keyboard on this device. The computer controls the punch mechanism entirely electronically-- sending impulses to the punch station to punch the proper sequence of holes to represent each piece of data. When all output has been produced on the paper tape for any given run, the operator removes the coded tape and stores or routes in according to the instruction sheet for the run.

If you will go back and reread page 33, you will see that the computer can also use the paper tape punch attached to a teletypewriter as a paper tape output device. The Teletype need only be connected by phone to the computer and the punch turned on for the computer to be able to use it for output purposes. But, remember, this is a remote terminal used only at a distance from the computer. The computer operator ordinarily is concerned with output on paper tape only when it is being output by the punch in his computer room.

Magnetic Output Devices

The magnetic tape, disk or drum devices you studied on pages 21 to 30 as input devices can serve as output units as well. Whenever a program requires output onto a magnetic medium, the computer electronically activates the read/write assembly of the appropriate device and transmits all data to it to be "written" (encoded in magnetic spots) on the medium. Any information output onto a magnetic tape, disk or drum can be easily stored in the computer room for future use.

When a computer program specifies a magnetic medium, say a disk, for output, the computer operator loads the appropriate disk or disk pack onto the disk drive. When the computer comes to the part in the program requiring this output means, it picks up the disk and track location specified by the program and transmits that location to the disk-drive's read/write head. The read/write head is activated, and it reads until it reaches the desired location. Then the computer transmits the data that is to be written on the disk to that read/write head, which records the data onto the disk.

Once all the information has been output onto the magnetic disk or disk pack, the operator is responsible for storing the medium in its proper place for future use. The same would be true were magnetic tapes used. As you will recall, however, a magnetic drum is fixed in in its drum drive and not moved about by the computer operator.

Teletypewriter Output

In the previous section on input devices, you saw how someone at a remote place from a computer could communicate with the computer by telephone using a teletypewriter. The teletypewriter (Tty) can be used, in fact, for both input and output of data.

On the Tty information can be output in two different media: it can be printed on paper or punched on paper tape. In the case of printing out the data, the computer simply electronically controls the Tty assembly and activates it to print data out on the regular paper above the keyboard. In the case of outputting data on paper tape, the computer electronically activates the Tty's paper tape punch and the data is is punched, item by item, onto paper tape.

Cathode Ray Tube Devices

Another common remote-terminal device is the cathode ray tube (or CRT), which looks something like a television screen.

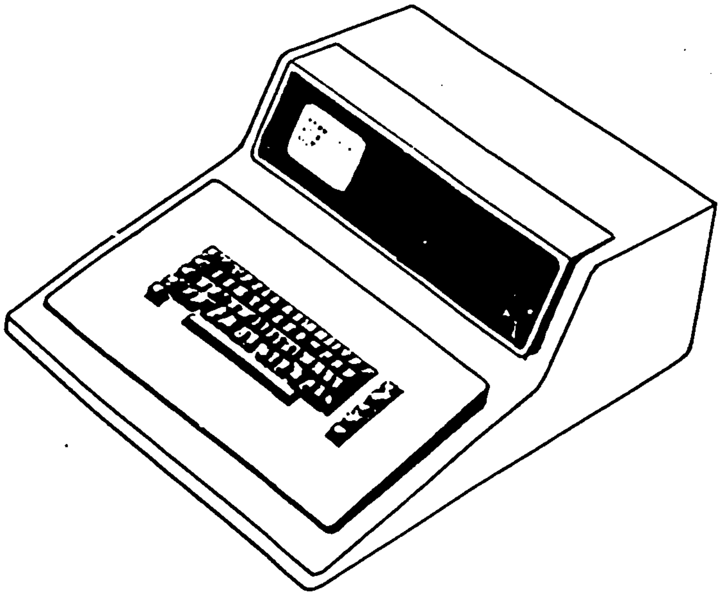


Fig. 27 A cathode ray tube with a typewriter keyboard attached.

The display screen of the CRT is most frequently used as an output device only. In these cases, the computer outputs data on a CRT by transmitting electronic signals which are transformed by the device into a display on the screen. You can pıobably guess that the CRT is a particularly useful output device when the output is a graph, a blue print, or some other graphic data.

In some cases, CRT's are used which are especially designed to act as an input as well as an output device. In such instances, the mechanism is built so that information can be sent to the computer by you touching the screen with a light-pen (a pen with a light-bulb as its tip). Such CRT's are commonly used in engineering work. Here drawings stored in the computer may be displayed on CRT's whenever needed and changes may be made in these drawings simply by using the light-pen.

Sometimes CRT's are used with keyboards attached to them, as in the illustration in Fig. 27. Such keyboards are, of course, used as input devices themselves and function in the same manner as the conventional teletypewriter keyboard.

Check Your Understanding

Put a check in the box(es) beside the correct answer(s) to the following questions.

1. The card punch used by the computer as an output device is

   □ a. the keypunch machine.
   □ b. often combined v·ith a card reader.
   □ c. is often a part of a teletypewriter unit.
   □ d. none of the above.

2. A CRT is

   □ a. another name for a teletypewriter.
   □ b. a device used for outputting information on a permanent medium.
   □ c. never found inside the computer room.
   □ d. an output (or input/output) device with a television-like screen.

3. As an output medium, paper tape can be punched

   □ a. only by the teletypewriter operator using a Tty.
   □ b. only by the computer operator using the Tty.
   □ c. only by the teletypist using a paper tape punch.
   □ d. only by the computer using a paper tape punch device.

4. Magnetic tapes, disks, and drums can be used

   □ a. for both input and output.
   □ b. for output only.
   □ c. for input only.
   □ d. none of the above.

5. With respect to the operation of output devices, the computer operator has

   □ a. nothing to do.
   □ b. to cue the computer when it should activate an output device.
   □ c. to load appropriate media onto output devices for computer runs.
   □ d. to control the output devices by means of the CRT.

Check your answers using the key at the bottom of page 50.

LABORATORY -- OPERATING OUTPUT DEVICES

You will now have a chance to learn to operate the output device(s) used in your computer center. First, you will need to arrange with your teacher for a time when you can go to your computer center. Then, follow the sequence of activities below.

1. Identify all the output devices used by your computer system. Check (✓) each one in the list below that your system uses.

   ____line printer

   ____card punch

   ____paper tape punch

   ____magnetic tape drive

   ____magnetic disk drive

   ____magnetic drum drive

   ____teletypewriter

   ____CRT

   ____Other (specify): ______________________________

   (Check with your teacher or operator to be sure you have correctly identified all the devices.)

2. Get the operator's manual for each output device and return to your classroom to study them. When you understand how to operate each device, go on to number 3 below.

3. Schedule a time with your teacher or operator when you can have assistance in setting up each output device for a computer run. For each output device to be used, you must ask your teacher operator to provide some appropriate sample input. (For example, to use the line printer, your input must require the use of the line printer for output.) Also, ask him to provide you with a quantity of media for each output device. (For example, if you will be using a card punch, you will need some blank cards.

4. With your teacher or operator standing by, load the first output device with a quantity of its appropriate media, according to the directions provided in the manual. Then, follow the correct procedures for switching the device on.

5. Ask the teacher or operator to give you the appropriate sample input to make a run using the now-ready output device. Load and start the correct input device, following the procedures you learned in your last laboratory exercise.

6. Your teacher or operator will be responsible for operating the computer for you once your input and output devices are ready. As the run proceeds, you should pay close attention to the output device to see how it operates.

7. When the run is finished, retrieve the sample input and the output and switch the devices off. Return the input to your teacher or operator, and save or discard the output.

   (Repeat steps 4 to 7 for each output device to be used.)

8. If you have any trouble operating one of the input or output devices, go back to study the operator's manual more carefully. Then, begin at number 3 above, again.

9. When you have successfully operated the output devices available to you, return to your classroom. Use the space provided on the next pages to list the steps (in the proper sequence) that you went through in setting up and operating each output device.

Answers to questions on page 48:
1. (b) 2. (d) 3. (d) 4. (a)
5. (c)

Output device ______________________________

Output device ______________________________

(Use the next page if you need more room.)

Fig. 28 Steps for setting up and operating output devices.

Output device ____

Output device ____

CENTRAL PROCESSING UNIT

Now that you are familiar with the input and output devices most often used in computer systems, you can spend some time learning about the heart of the system: the central processing unit (or CPU). It is here that the input data is processed to result in new output information.

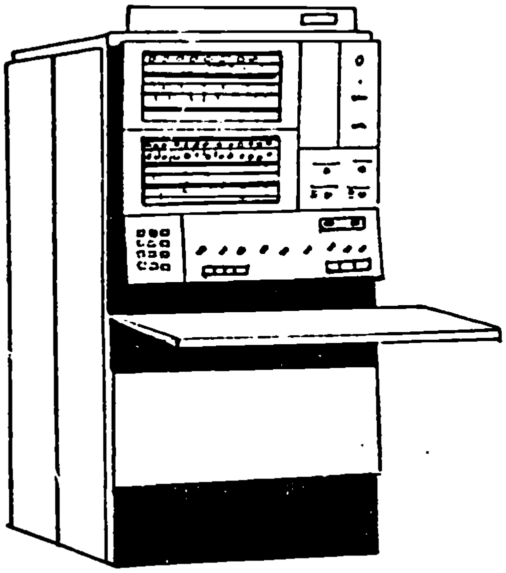


Fig. 29 A central processing unit.

In most computers, the CPU is housed in a single cabinet, or console, sometimes called the "mainframe." Although the central processing unit appears to be a single piece of equipment, it actually has three distinct parts: one which stores the programs and data for use in the run (the storage unit), one which takes care of the actual processing or manipulating of data (the arithmetic/logic unit), and one which coordinates the flow of data through the system (the control unit). We'll look at these three units separately, starting with the storage unit.

The Storage Unit

You have already seen that a computer system must have both a program and the data in it before it can process the data or produce results. These are usually loaded into the computer through input devices just before the particular run is made. Normally, the program is loaded in first and then the data to be processed is input.

Did you wonder where the program and data went once "loaded" in? Here's your answer: all input is loaded directly into the CPU's storage unit, usually called "main storage" (or "core storage").

As you may have already guessed, the information (programs and data) in main storage is temporary -- if it weren't, the computer operator would only need to load a given program once and be done with the punched cards or magnetic medium from which it was input. Any information in main storage remains there normally only until the data has been processed and the results have been output. Once a run is finished, new input is loaded into the computer and it replaces the old program and data in main storage. (This is the reason programs and data files are kept in permanent form on various media like paper tape, punched cards, and magnetic tape, for storage in the computer room.)

To understand how programs and data are "stored " temporarily in main storage, you can think of main storage as resembling a post office with a lot of numbered boxes like this:

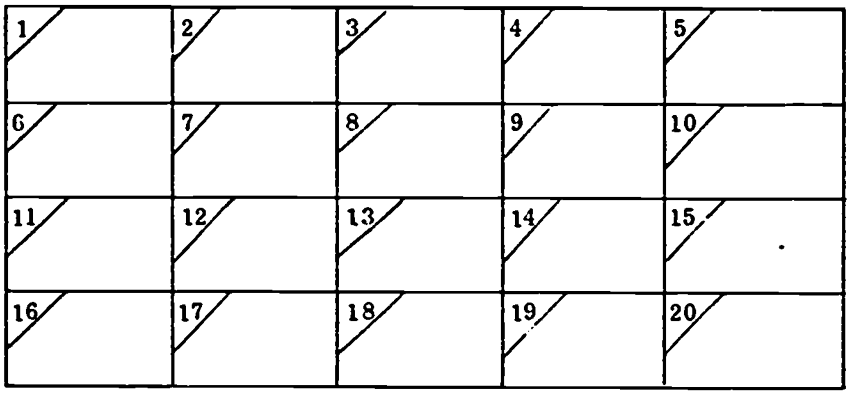


Fig. 30 Main storage with 20 empty boxes.

The numbers in the corners of the boxes indicate the location or address of the box, and each box can store only one piece of data. Using location numbers, you can see that a piece of data stored in any box can be "found" simply by calling on its location number.

If we continue to use the post office diagram, here's how main storage might look once it has been loaded with a program for adding two numbers (which will be input later to boxes 6 and 7.)

1 READ BOX 6	2 ADD BOX 7	3 STORE TOTAL IN BOX 8	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20

Fig. 31 Main storage with a program in it.

Now, let's assume the two numbers (the data) to be added for our first run are 1143 and 921. They would be input after the program, and main storage would then look like this:

1 READ BOX 6	2 ADD BOX 7	3 STORE TOTAL IN BOX 8	4	5
6 1143	7 921	8	9	10
11	12	13	14	15
16	17	18	19	20

Fig. 32 Main storage with program and data.

Once the data is processed, main storage would look like this:

1 READ BOX 6	2 ADD BOX 7	3 STORE TOTAL IN BOX 8	4	5
6 1143	7 921	8 2064	9	10
11	12	13	14	15
16	17	18	19	20

Fig. 33 Main storage after processing of data.

This example is, of course, a terribly simple one. It is just to give you a clear idea of how data is stored in main storage and how it can be found there for use when the processing must be done. Most programs are much longer and data for a run can be extremely long as, for example, in the case of complete records from kindergarten on for all students in a high school.

Once in main storage, programs and data can be "accessed" (or found) at any time for use by the computer -- that is, until new data has been loaded into the same main storage locations for the next run. This, in fact, is the key to the temporary nature of main storage: once a run has been completed, the program and data for the next run are loaded into storage and replace (or erase) the data held in those locations before.

The Arithmetic/Logic Unit

It is in the arithmetic/logical unit that the real "processing" of data takes place. This unit performs calculations and makes decisions, according to a program stored in memory.

The logical operations that are performed here include comparing information and making a decision based on the results of that comparison. A typical comparison might be:

> "Is the answer less than zero? (Compare the answer with zero.) If so, stop the program. If not, go on to the next step."

The arithmetic operations add, subtract, multiply and devide are also performed by this unit. Any complex calculations can always be reduced to a combination of these four operations, so that even the most difficult (for us) calculation is performed effortlessly and automatically by the arithmetic/logic unit of the computer. So, in the example we showed above in the section on main storage, it is the arithmetic/logic unit which registers the number 1143 and then adds to it the number 921 and comes up with the total 2064, which is then stored in main storage.

If main storage just holds data at prescribed locations and the arithmetic/unit performs operations, then -- how does anything know when or how to go where for what processing? What keeps everything from happening at once? The control unit!

The Control Unit

The control unit directs the flow of information through the computer system. It controls the movement of information between input/output devices and main storage, and directs the sequence in which operations will be carried out. Basically, it simply allows one step of the program to be acted on at a time, one after the other. Using our example from the section above on main storage again, it is the control unit which would first indicate that box 1 should be acted on -- that is, the number 1143 should be registered by the arithmetic unit. After that, it allows box 2 to be acted on, and so forth, until the run is completed.

You might think of this unit as acting like the conductor of a symphony orchestra: following the musical score (the stored program) and cueing the various instruments (input, output and storage devices) when it's their turn to play. The conductor keeps the orchestra together, playing harmoniously in the sequence prescribed by the musical score. The output then is predictable and pleasing, as is the output from an efficiently controlled computer program.

AUXILIARY STORAGE DEVICES

You will remember that the information that is in main storage is temporary. That is, it remains in main storage only long enough to be processed and output. Then new information is input to main storage, and the old is erased. (This is very much like a home tape recorder -- when you record a new song, whatever was on there before is automatically erased.)

Frequently, however, we have a need to store a large volume of information, and to store it permanently. For example, the complete pupil records from kindergarten on for every student in a high school would be too much information to be stored permanently in main storage. Even if a computer had such a large enough main storage unit, it would need to be kept available for the temporary storage of programs and data currently being processed.

Auxiliary storage devices, then, can be connected to the CPU to provide information from large permanent storage files. Since the control unit has to go outside of the CPU to get data from the auxiliary storage devices, the information is not as quickly accessible as information in main storage. The loss in speed, however, is offset by the larger capacity and the lower cost of auxiliary storage as compared to main storage.

Many of the input/output devices you have studied about can be used as auxiliary storage devices as well. Magnetic tape, disk and drum units are the most common. Even punched cards and paper tape can be considered as "auxiliary storage" in that they permanently store information outside the CPU.

There are other special auxiliary storage devices which can hold much more information than tapes, disks or drums. One is called a "Datacell;" this storage device holds billions of characters of information on magnetic strips. The strips hang from rods with a read-cylinder at the top of each rod. When the computer needs to read some data, the appropriate strip on the rod is pulled up and wrapped around the read-cylinder for the data to be read. The datacell would be used in a system that required the permanent storage of massive amounts of readily accessible information.

COMPUTER CONSOLE

One of the major values of a computer is that, once started working on a job, it can basically proceed to the end without human intervention. A stored program provides the machine with the instructions it needs to obtain information, process it, and output new information.

In practice, of cource, the operator must communicate with the computer in order to get the computer ready for running a program, and he must occasionally intervene during the data processing.

As you have seen, there is a wide variety of input-output devices available for exchanging information between a computer and its human masters. All of these devices allow an indirect kind of communication. The only direct communication with the computer is via the computer console, found on the CPU.

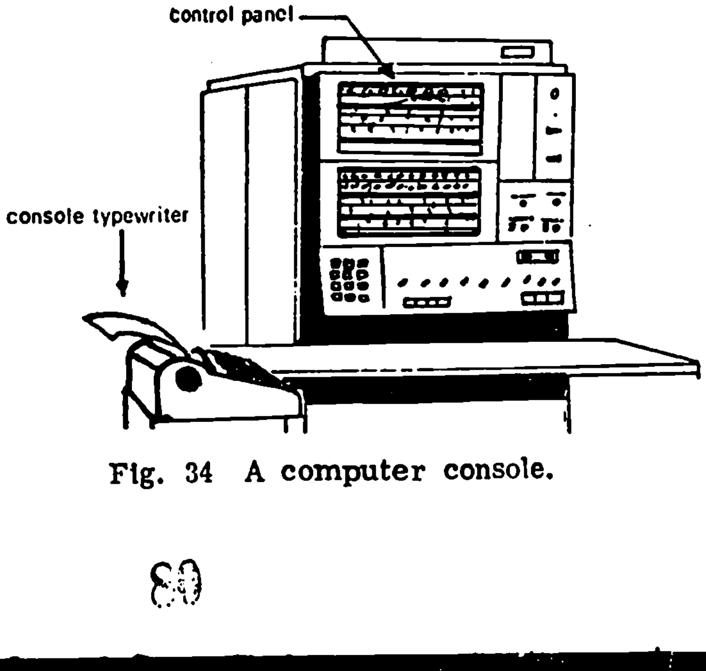


Fig. 34 A computer console.

Each computer console model has its own special features, but the typical console contains two devices for manual communication with the computer: the console typewriter and the control panel.

Console Typewriter

The console typewriter looks like an ordinary typewriter, with a standard keyboard and paper output. This typewriter can be used both for output from the computer and input to it.

At the direction of the program being run, the computer can type out messages to the operator on the console typewriter like:

"MOUNT TAPE REEL #4"

or "PLEASE TYPE TODAY'S DATE"

or "WHAT SIZE LABELS?"

or "HOW MANY COPIES ARE NEEDED"

The operator can answer these queries from the computer by typing on the keyboard. When a key is pressed, the corresponding number, letter, or symbol is typed on the typewriter paper and is simultaneously transmitted into the computer in electronic code.

The console typewriter could actually be used by the operator to input any kind of material into the computer -- data, programs, or the operating instructions called job control statements. But, can you see why it usually isn't used for these things? First, when data or programs have once been encoded into cards, tapes, or other media, they can be kept indefinitely; when they are needed for use, the medium can be mounted and loaded into the computer in a matter of minutes -- even seconds! If you input a program or data file from a console typewriter, it could take a great deal of time. Not only that, but you would have to retype it on the keyboard every time you wanted to use it. Second, the instructions given to the computer in the form of job control statements are usually basic directions needed by the computer to properly run the program involved. Since one set of job control statements is always needed for the running of each program, it is more convenient to have the set of statements on cards or paper tape for instantaneous input than to have them typed on the console typewriter each time the program is to be run.

Generally speaking, the console typewriter is the computer operator's main means of communicating with the computer to control its operation; the main input is therefore instructions for control purposes.

Commonly, the operator uses the console typewriter initially to instruct the computer to get ready for a specific run and then to activate the input devices to "read in" appropriate programs and data. Further, when during the processing of data the program calls for responses from the operator such as in the examples above, the computer operator types in the answer on the console typewriter. Sometimes instructions must be input by the operator in mid-run; these are always clearly noted on the instruction sheet given to the operator by the programmer, and the information is input from the console typewriter. Finally, the operator uses the console typewriter to instruct the computer to end a job either when a problem has arisen or when the job is done. The console typewriter can also be used to directly examine and to alter the contents of any storage location. The procedure involves simply typing in a request that the data in a specific location be typed out by the computer and then typing in any alterations to be stored.

Each computer has an operator's manual or directions booklet which outlines the procedures the operator must use in controlling routine runs from the console typewriter.

The Control Panel

The control panel is like a window into the computer. It has sets of small incandescent lights which display the contents of certain internal registers. It also has a means for manually displaying, examining, and altering the contents of storage.

While the computer is running a program, the lights change too rapidly for the contents to be read. Whenever the computer stops because of a problem, though, the light display showing the register contents is often a valuable aid in determining what has happened.

The control panel is often used for detecting program mistakes -- a process usually called "debugging." An operator can set switches on the control panel to stop operations in order to read the content of a storage location, observe a program instruction, or examine the registers of the arithmetic unit after each instruction. This kind of debugging by step-by-step operation and observation of the program is, however, usually ruled out as inefficient in a large computer system.

The computer operator also uses the control panel to monitor the general operation of the system. Indicator lights on the control panel show the operation or condition of important parts of the CPU and in such equipment as the card reader, mangetic tape units and the printer. For example, there may be a temperature warning light, overflow lights, error indicators, and lights to signal some off-normal condition, such as the printer being out of paper.

Probably the one switch always found on a control panel is the START key. When depressed, it causes the computer to start running a program, or to continue after an operator intervention.

Between the console typewriter and the control panel, a computer operator can stay in very close communication and control of the computer he is responsible for.

Check Your Understanding

Put a check (✓) in the box(es) beside the correct answer(s) to the following questions.

1. Main storage in the CPU differs from storage on an input/output medium by

   □ a. being less permanent.
   □ b. having less space available.
   □ c. being more immediately accessible to the computer.
   □ d. all of the above.

2. Each CPU storage location has

   □ a. a location number.
   □ b. a program in it.
   □ c. the capacity to store entire data files.
   □ d. none of the above.

3. When information is placed in a main storage location, the old information in that location is automatically

   □ a. moved to permanent storage.
   □ b. destroyed.
   □ c. added to the new.
   □ d. output.

4. The CPU contains the following three parts:

□ a. an arithmetic/logic unit, auxiliary storage, and a control panel.
□ b. a control unit, an arithmetic/logic unit, and input/output devices.
□ c. a control unit, a control panel, and a typewriter.
□ d. an arithmetic/logic unit, a storage unit and a control unit.

5. The part of the CPU which acts as the coordinator of the system's operation is

□ a. the control panel.
□ b. the logic unit.
□ c. the "main frame."
□ d. none of the above.

6. The arithmetic/logic unit

□ a. keeps track of all data in storage.
□ b. can perform four arithmetic operations: addition, subtraction, multiplication and division.
□ c. controls the movement of all data to and from main storage.
□ d. all of the above.

7. The console typewriter is

□ a. often substituted for a control panel.
□ b. used as a general input device.
□ c. part of auxiliary storage.
□ d. the operator's main device for communication with the computer.

8. What is the common term for the set of instructions needed by the the computer to run a program?

□ a. job control statements
□ b. console instructions
□ c. program data
□ d. programming statements

Check your answers using the key at the bottom of page 64.

LABORATORY -- OPERATING THE COMPUTER

You are now ready to try your hand at operating the computer yourself during a run. Begin by arranging with your teacher for a time when you can go to your computer center to watch the operator run a job on the computer. Take note of the number of times the operator has to communicate with the computer during the run, and how much the control panel or the console typewriter are used.

1. From your teacher or computer operator, get the operator's manual for your particular computer and return to your classroom. Study the part of the operator's manual which explains how to run a job on this particular computer. When you understand how to run a job, go on to number 2 below.

2. Schedule a time with your teacher or operator when you can have help in running a complete job on the computer yourself. Ask your teacher or operator for

   a. program and data input for a computer run.
   b. quantities of appropriate unused output media for the run.
   c. a list of the appropriate job control statements* for the run and information about the media to be used for inputting those statements.

3. Using the operator's manual for reference, prepare any job control input that is required for the run. (For example, if job control statements are to be entered from the console typewriter, all you will need is a list of the exact statements in the correct order, for your reference. If punched cards are to be used, of course, they must be punched on a key punch machine and inserted in the proper places among the program and data cards.)

*As will be explained in the manual to your computer, job control statements must be input at various points to give the computer direction during a run. Your computer manual will list all the control statements your machine uses. Your operator can tell you specifically which of those statements to use with your particular input.

4. With your teacher or operator standing by, set by and start the output device(s) for your run. Then load and start the input device(s).

    a. If the job control statements are to be entered from the console typewriter, go to the console typewriter with your list of statements. Enter the first statement(s) to start the run, and enter the remaining statements as they are required. Throughout the run, observe the operation of the system.

    b. If the job control statements are to be entered with the program and data input, the starting of the input device will, in effect, start the computer run. The entire run will be conducted automatically by the job control input. You should just observe carefully how the system operates throughout the run.

5. If you have any trouble during the run ask your teacher or operator to help you locate and correct the problem.

6. When the run is finished, retrieve the sample input and the output and switch the devices off. Return the input to your teacher or operator, and save or discard the output.

7. When you have successfully operated the computer system, return to your classroom. Using the space provided on the next page, list the steps (in the proper sequence) that you went through in preparing the job control statements and running the computer.

Answers to questions on pages 61 and 62:
1. (d) 2. (a) 3. (b) 4. (d) 5. (d)
6. (b) 7. (d) 8. (a)

(Use the back of this page if you need more room.)

Fig. 35 Steps for preparing job control statements and running the computer system.

PART III

COMPUTER SOFTWARE

PART III COMPUTER SOFTWARE

INTRODUCTION TO SOFTWARE

As you are now aware, the computer operator will need to recognize and and be able to operate the various pieces of equipment or hardware we have discussed so far. Computer hardware is, of course, the most visible part of a computer system, but it is not the most important. Even more important than the hardware is the software.

Software is the word used to refer to all of the programs that make the computer function. Without software, a computer is like a camera without film -- all the button-pushing and switch-flipping on earth won't produce anything.

Early computers used very simple programs. A computer programmer would write a set of instructions in a numeric code telling the computer how to solve a problem or process some data. Then the computer operator would take over. First, he would load the program from punched cards or whatever into the computer. Then, he would put the data in an input device -- usually a card reader -- and start the program run. During the run he might have to interrupt the operation of the computer several times -- to provide more data, to mount new magnetic tapes for output, to prepare the printer for output, to check the output for accuracy and even, perhaps, to make manual changes in the program itself by operating the control panel switches.

When one job (or program run) was completed, the operator would stop the computer, retrieve the data cards used, gather up all the output, and prepare for the next job. Meanwhile, the computer waited. Then the process would begin again: load a program, provide data for input, control and keep track of the input and output, and manually monitor the operation of the computer during the program run.

As you can see, early computers spent a lot of time both during and between program runs just sitting around waiting for the computer operator to perform his tasks.

Today's computers are far closer to being true "labor-saving" devices. Sophisticated developments in hardware have been matched by an increase in sophistication of the computer's software. Now days many of the tasks of the computer operator have been taken over by software, so that jobs can be run more quickly and efficiently and automatically, with less waiting time for the computer and less intervention by the computer operator.

Just as developments in software have increased efficiency of computer use and lightened the computer operator's load, so have software developments made the job of the programmer easier.

It used to be necessary that programs be written in the specific numeric code that the given computer used. Now, though the computer still works with numeric code, programmers write instructions in ordinary English words. New software can now translate these instructions into the required numeric code, saving much time and trouble for the programmer.

When the decision is made to acquire a computer system, it is as important to select a system with good software as it is to select good hardware. A computer without proper software will add considerably to to the cost of operating the installation -- software with low efficiency may increase computer running time by 25% or more.

The manufacturer usually supplies a complete "software package," which will include special programs to make the computer operate most efficiently, programs to do standard, often-used routines, and translators which will convert new programs into the machine's numeric code. This software package is usually called the operating system.

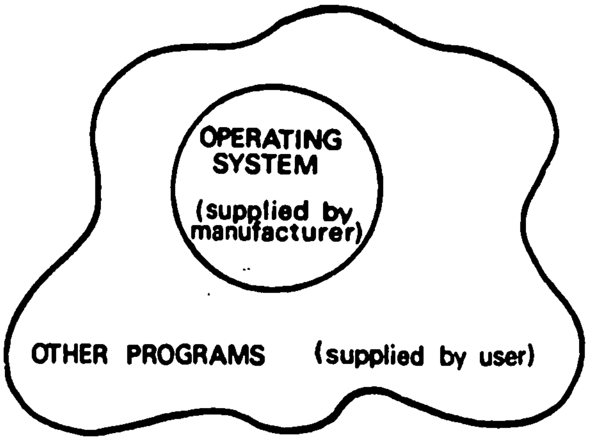


Fig. 36 The computer software.

In addition to the operating system, the computer software includes all of the "other programs." These other programs are supplied by the user and are designed to perform specific jobs as the need arises.

The programs in the operating system are stored on an auxiliary storage device, often a disk or drum. The illustration on the facing page shows how these programs may be organized.

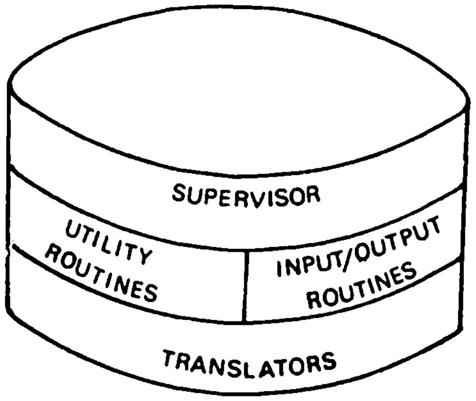


Fig. 37 Operating system stored on an auxiliary storage device.

The Operating System

An operating system is an organized collection of programs which increases the productivity of a computer by aiding in the preparaticn, translation, loading and execution of programs. Intervention by the computer operator is reduced when an operating system is available; this usually means that an operator can use the time during one run to set up the equipment for the next run. Often, several jobs can be run in an uninterrupted sequence, under the control of the operating system.

The programs most likely to be found in an operating system will be discussed here. However, the programs available may vary from one operating system to another, and certain programs may be called by other names than those used here. This manual should be a starting point. When you have an opportunity to operate a real computer system, you should carefully read the manuals that come with the system. These manuals usually give complete descriptions of the software included.

The types of programs most often available in an operating system in include:

- A supervisor (often called a monitor or an executive)
- Input/output control
- Utility programs
- Translators

Now let's look at these programs one at a time to see just how they increase efficiency and ease of operation for not only the computer, but the programmer and operator as well.

The Supervisor

The supervisor program or routine is the most important and complex of all programs in the operating system, because it sets up, monitors, and controls the operation of other programs. While most software is placed in auxiliary storage (disk or drum) until needed, the supervisor routine is loaded into the computer each morning usually to remain in main storage throughout the day's operation. It organizes and supervises the running of all other programs in the operating system.

A basic supervisor routine controls the execution of one program by clearing the storage area (placing zeros or blanks in work areas), then loading the program into storage.

Once the program is running, other programs in the operating system take care of the flow of operation. When errors are encountered during processing -- for example, when a program exceeds the storage capacity -- control is once again transferred to the supervisor routine which takes corrective action or halts execution.

An important part of the supervisor routine is the "job control" program. This program handles a series of programs sequentially with a minimum of operator intervention. To do this, the computer requires initial instructions. The programmer usually has these instructions punched on cards and gives them to the computer operator for the run. These cards are called the job control cards. The instructions on the job control cards may give the sequence of steps to be performed, and may indicate what programs in the operating system are to be used.

The supervisor routine starts and stops the programs and performs many functions normally otherwise done by the operator, but it does them faster and more accurately. Where operator intervention is required, the supervisor will print out operator instructions on the console typewriter.

In some more complex, modern computer systems, the central processor is a <u>multiprocessor</u> -- that is, it has more than one arithmetic-logical unit. This means that several programs can be run simultaneously. Of course, the supervisor routine and its job control programs must be even more sophisticated to schedule and monitor the execution of more than one program at once.

In other systems, a technique called multiprogramming is used, where there is only one arithmetic-logical unit, but several programs are in the computer at one time. The supervisor routine divides the computing time among the different programs, transferring control from one to another to process them all at the same time.* Again, a sophisticated supervisor routine is required to schedule and control the concurrent execution of the programs.

As an example of multiprogramming, a program to read and print cards may be run concurrently with a program to sort records on magnetic tapes. The supervisor routine initiates the instruction to "read a card" for the first program, then may transfer control to the sort program. When a signal indicates the card has :en read, the second program would be interrupted and control returned to the read-print program for moving the data into position for printing. Since the read-print program can proceed no faster than the reader and printer can operate, the CPU would be idle much of the time if the supervisor routine did not use this idle time to run another program.

Input-Output Control Programs

The control of input and output (I/O) operation requires a separate set of programs because these are mechanical functions performed by mechanical devices. If the speedy electronic central processor had to stop and wait during each I/O operation, its efficiency would be greatly reduced.

If data were read directly into the CPU, or directly out, the fast CPU would be tied up as the slow mechanical reader or printer operated. Iowever, most computers now use electronic buffers to cut down this delay. When data is read from an input device, it goes into an input buffer, where it is held until the CPU is ready for it -- then it is transferred from the buffer to the CPU at a fast electronic speed rather than a slow mechanical speed. Likewise, data is transferred at electronic speed from the CPU to the output buffer and output when the output device is ready.

The scheduling and overlapping of these buffered I/O operations is complicated, and must be controlled by the Input-Output Control Programs.

These control programs also handle such operations as I/O error detection, switching to an alternate tape drive at the end of a reel, and searching for the proper items to be input from a reel of tape, a disk or a drum.

*This technique is called time-sharing.

Utility Programs

Utility programs support the production work of a computer center, by servicing jobs which recur frequently. While the specific utility programs available will differ from one system to another, a fairly complete set would include:

- a program loader.
- programs to transfer data from one medium to another (card-to-tape, tape-to-card, card-to-disk, disk-to-card, tape-to-printer, disk-to-printer, etc.
- "diagnostics" or test routines to help identify and correct machine failures or program errors.
- special programs to protect permanent files, sort and merge data, duplicate tapes or disks onto new tapes or disks, "memory dumps" (output the entire contents of main storage on the printer).

The program loader is used to enter new programs into storage. Even the programs in the operating system must be initially introduced into storage by the loader.

A program is stored in memory in the same way as data. It is read in, probably from punched cards or tape. The loader supervises the reading in, allocates storage locations for each program instruction, and finally transfers control to the first instruction of the program just loaded, so the program can be executed.

You may be asking -- "But how does the loader itself get into storage?" It's a good question. The original loader is stored manually by the operator, one instruction at a time, accessing main storage directly through the switches on the console control panel. Since this is a rather laborious and time-consuming process, woe be to the person who inadvertently erases that part of storage that contains the loader!

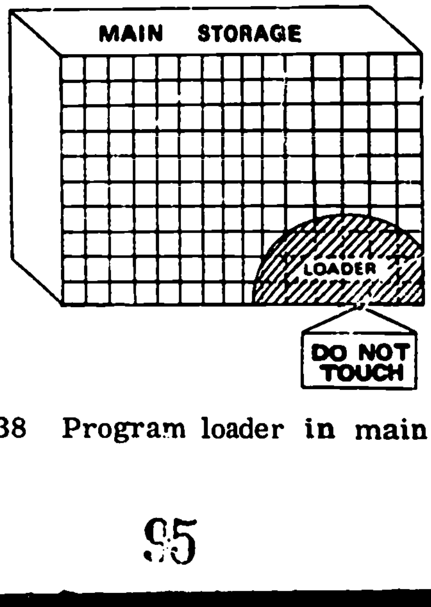


Fig. 38 Program loader in main storage.

Transferring data from one medium to another is often necessary and must be controlled by special programs such as card-to-tape, tape-to-printer and the like. Likewise, there are special programs for duplicating tapes or disks onto new tapes or disks.

Diagnostics of test routines may function automatically to check for machine or program error during operation of programs, and also may be called into action when there appears to be a malfunction in a program. If a programmer wishes to use a test routine designed to help him locate a program error, he would probably call for a "trace" or a "dump."

A trace routine will print the results as each step of the program is executed, to allow the programmer to closely observe the operation of the program.

A dump routine, often called a memory dump or memory print, will print the contents of each storage location at any specified check point in a program. The programmer can also specify the boundaries of the storage area he wishes to examine.

Special programs take care of frequently performed operations, such as copying all files from tape or disk at intervals as a protection against accidental erasure or loss of important data.

Another frequently used utility program found in most systems is a Sort-Merge routine. It should be able to sort either numeric or alphanumeric records into ascending or descending order, merge one set of records with another set of records, add and delete data, and provide for input and output on various media.

Translators

In the early days of computers, programmers wrote instructions entirely in a numeric code directly understandable by the computer. This meant that the programmer had to translate each instruction to a corresponding numeric code.

For example, if a programmer wished to write the program instruction

"SUBTRACT TOTAL DEDUCTIONS FROM GROSS PAY"

he would first translate it to a code that might look like this:

07 326 145

where:

07 = "SUBTRACT"

326 = The address of the location where "TOTAL DEDUCTIONS" is stored

145 = The address of the location where "GROSS PAY" is stored

As you can see, the job of the programmer could be quite time-consuming, tedious and error-prone.

But computers were invented to save work for humans, not to create more! It wasn't long before some programmer with writer's cramp realized that a sophisticated translator program could automatically translate

"SUBTRACT TOTAL DEDUCTIONS FROM GROSS PAY"

to

"07 326 145"

much faster and much more accurately than the programmer could translate it himself.

Since computers are designed especially to manipulate symbols (like letters and numbers), translating words into numeric codes is much easier to the computer than it is for humans. Although computers still operate exclusively in numeric codes, seldom does a programmer have to actually use numeric codes now days. Over the years, several different kinds of translators have been developed to ease the task of the programmer.

The most simple kind of translator is called an assembler. It requires more of the translating effort to be done by the programmer and less by the computer. Most operating systems, however, include at least one assembler.

The type of translator used most often by programmers is the compiler. Compilers allow the programmer to write instructions in a language much like conversational English. The translating to numeric code is almost completely carried out by the computer.

For example, the programmer can write one instruction like:

"GROSS PAY — (SOCIAL SECURITY + OTHER DEDUCTIONS) = NET PAY"

and the compiler will "compile" or translate this to a series of numeric codes:

```
"06 325 324
 15 326 000
 07 326 145
 15 327 000"
```

One common compiler is used primarily to translate programs used in business. It is called COBOL: COmmon Business Oriented Language.

Another widely used compiler is called FORTRAN, for FORmula TRANslator. FORTRAN was designed to translate scientific programs to numeric code.

BASIC (Beginner's All-Purpose Symbolic Instruction Code) is a compiler used primarily with remote terminals, where the user carries on an interactive "conversation" with the computers.

Any or all of these compilers, and perhaps other compilers as well, may be found in the operating system supplied by the computer manufacturer.

Some operating systems include special-purpose translators that cannot be called either assemblers or compilers. They may be called report program generators or interpreters, but they still serve as translators of our language to the machine's numeric code.

Check Your Understanding

Put a check (✓) in the box(es) beside the correct answer(s) to the following questions.

1. The software of a computer is

   □ a. the electrical circuitry of a system.
   □ b. the documentation of a system.
   □ c. the computer programs.
   □ d. the data cards, tapes, etc.

2. The following is not a part of an operating system:

   □ a. utility programs.
   □ b. the control unit.
   □ c. the supervisor routine.
   □ d. translators.

3. A program loader is

   □ a. usually loaded into main storage from a magnetic storage medium.
   □ b. loaded by hand from the control panel.
   □ c. loaded into the computer as a part of each program to be run.
   □ d. none of the above.

4. A translator is

   □ a. a programmer specializing in program languages.
   □ b. a program used to control input and output.
   □ c. a program which converts programs from words to numeric codes.
   □ d. none of the above.

5. Assemblers and compilers are both

   □ a. input devices.
   □ b. programers.
   □ c. utility programs.
   □ d. translators.

Check your answers using the key at the bottom of page 80.

MODES OF COMPUTER OPERATION

A computer operator trainee will probably have his first experience with a computer which operates in what is called batch processing mode. This is the mode of operation used by virtually all computers manufactured during the first decade of electronic data handling, and is still found in many data processing installations.

A more recent evolution in computer operating modes is on-line processing.

There are advantages and disadvantages to both of these modes of operation. We will talk about both here, since one or both of these methods of processing data are used in nearly every installation.

Batch Processing

Batch processing is characterized by the "batching" or grouping of transactions so that as much work as possible will be handled all at one time. A good example of this can be seen in a city water department. Each customer's account normally has only two transactions a month: one bill is calculated and one payment is posted. It is not worthwhile to update the master file of customers' records each time a transaction occurs. Instead, billing is done in one batch for all customers once a month. Payments come in on a random schedule through the month, and a card is punched for each payment as it is received and deposited. These cards are collected, or batched, and processed as a batch once a month.

This method of processing saves time and money. But, this procedure means that the computer printout of customers' accounts that is available for answering queries is, on the average, two weeks old. If a customer calls requesting information about payments, balances and account status, he may get the reply, "Wait for next month's bill and see if it has been taken care of."

In a batch processing operation, master files are usually kept on magnetic tape in numeric or alphabetic order. When the batch of input cards is to be used to update the master file, it is first sorted into the same sequence as the magnetic tape records. Then the new input data is run against the master tape to create a new master.

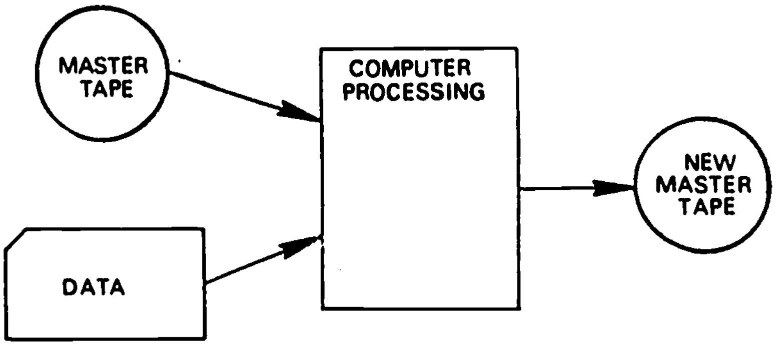


Fig. 39 Batch processing flow.

Obviously, it would not be practical in a system like this to process transactions as they occur. If every water payment were processed against the master tape the day it came in, the entire master tape would have to be run each time to find the proper record to be updated. The serial nature of magnetic tape makes it an impractical device when fast look-up of a single record is desirable.

As you can see, however, for an application like water department billing there is no urgent need to update the master file frequently. Batching the transactions to run all at once and using magnetic tape for master files is the most economical and convenient and the least time-consuming way to function.

There are many applications, however, which do not lend themselves well to batch processing. The batching time is often longer than the actual process time. In many cases, if data is held up to be batched all at once, other problems can be encountered:

- certain data may be out of date even before it is processed.
- reports may be produced too late to be helpful in making decisions.
- requests for information, such as current account balances, can be difficult to answer.
- since input data and all files are maintained serially, much money and time is spent in sorting data.

Answers to questions on page 78:
1. (c) 2. (b) 3. (b) 4. (c) 5. (d)

For applications that require information in large volumes and on short notice, these disadvantages are even more obvious.

Some examples of applications not well suited to batch processing are airline reservations systems, banking operations, credit bureaus, and stock brokerage houses, where there are frequent customer inquiries that must be answered immediately. Applications of this nature must be part of an on-line processing system.

On-line Processing

In an on-line processing system, data may be processed as soon as they become available regardless of the order in which they arrive. Input data are not sorted before being processed. This requires that the master file be stored on a "random access" device -- usually, a magnetic disk or drum. Since random access storage allows any part of the disk or drum to be read or written immediately, master files may be'referred to or updated at any time.

Magnetic tape, being a serial access medium, cannot practically be used in this way.

An on-line data processing system is significantly different from a batch processing system in many ways. A computer operator should be aware of these differences, as he may have opportunities to work with either type of system.

As was mentioned above, the storage medium used is likely to be random access rather than serial.

The major difference with on-line processing is the way the CPU is used. Instead of working on onepprogram at a time, the computer's time and main storage may be divided so that many inquiries can be processed at the same time. In an airline reservations system, for instance, cathode-ray-tube (CRT) terminals are located at ticket counters all over the country. They are connected to a central computer via standard telephone lines, and information goes to and from the terminal over these lines. As one ticket agent is typing in a question about space availability on a flight, the computer is sending confirmation of a reservation to a CRT terminal at another ticket counter.

As you can see, the input and output is likely to take place at a remotely-located device called a "terminal," and this terminal is likely to be only one of many which use the computer at the same time. This shared use of a single computer by many users all in remote locations is called "time-sharing."

The decision to implement a batch processing system or an on-line processing system depends on the volume of data to be produced, the need for immediacy, and the need for economy. Obviously, an on-line system would be much more expensive to install and maintain than a batch system. The software requirements are more complex in an on-line system. Often, however, the cost factors are offset by the value in having current data instantly accessible at the time it is needed.

Check Your Understanding

Put a check (✓) in the box(es) beside the correct answer(s) to the following questions.

1. Batch processing is characterized by

   □ a. master files kept on magnetic tape.
   □ b. transactions collected and run in a batch.
   □ c. output reports which are not often current.
   □ d. all of the above.

2. On-line processing is characterized by

   □ a. data processed immediately.
   □ b. many users at one time.
   □ c. punched card input.
   □ d. none of the above.

3. Usually an on-line processing system is

   □ a. a time-sharing system.
   □ b. very practical for schools.
   □ c. easier to maintain.
   □ d. outdated.

4. The best system to use for processing school grades would be

   □ a. batch processing.
   □ b. on-line processing.
   □ c. neither of the above.
   □ d. both of the above.

Check your answers using the key at the bottom of page 85.

DATA IN THE COMPUTER

You have seen how a computer accepts input, processes the data and produces output. What you haven't seen is what goes on inside the computer during the process.

Most input/output devices are connected to main storage by an electric cable. Most data comes from some outside medium such as punched cards. Data from the punched card is converted to electrical pulses which are transferred to the main storage unit of the computer. In main storage, the information represented by the electrical pulse is stored in a tiny ring made of iron called a "core." The core, illustrated below, can be used to represent either of two possible states: "on" or "off".

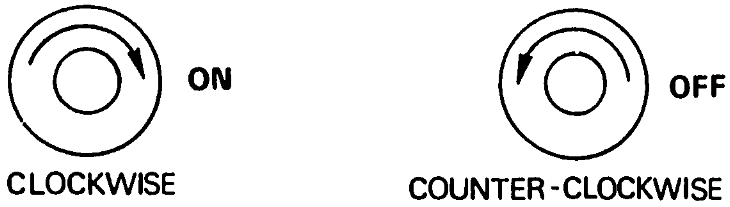


Fig. 40 "On" and "off" cores.

If the circuit is magnetized in a clockwise direction the computer has stored an "on." If the magnetization is counter-clockwise, to the left, we can say the computer stored an "off." Characters, numbers, letters and symbols can all be represented in code by groups of these magnetized cores.

When the processing is finished, the cores stored in groups of on and off cores are converted back to characters to be printed or punched.

Imagine the letter "H" being read into the computer from paper tape and stored by a group of cores in main storage. One code that is used to represent an "H" on paper tape is shown here:

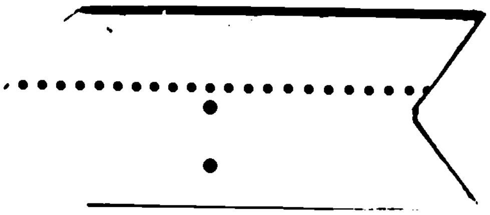


Fig. 41 "H" on paper tape.

When this 7-position pattern of holes and no holes (ooo●oo●) is translated into electrical pulses, it would look like this:

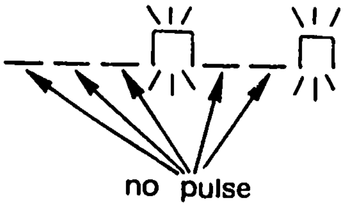


Fig. 42 "H" in electrical pulses.

And when the same pattern is translated from electrical pulses into magnetized cores, it would look like this:

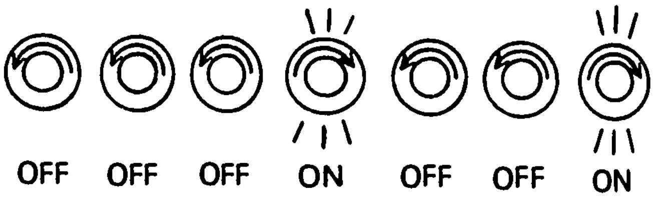


Fig. 43 "H" in magnetic cores.

This same code, representing the letter "H" could be displayed in a register on the control panel, by lights which are on or off.

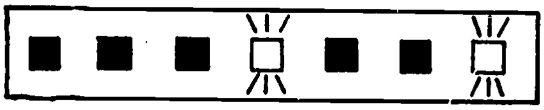
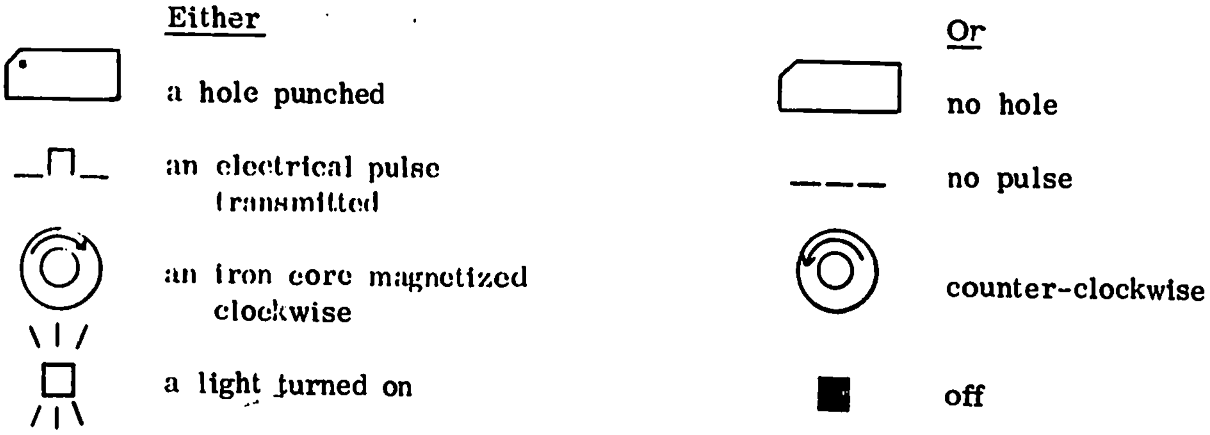


Fig. 44 "H" on a control panel register.

As you can see, a handy way to code data for input, transmission, storage and register display would be to use a code with only two possible states:



Either	Or
a hole punched	no hole
an electrical pulse transmitted	no pulse
an iron core magnetized clockwise	counter-clockwise
a light turned on	off

Fig. 45 Examples of two-state codes.

To explore the possibilities for such a two-state code further, we can look at auxiliary storage devices:

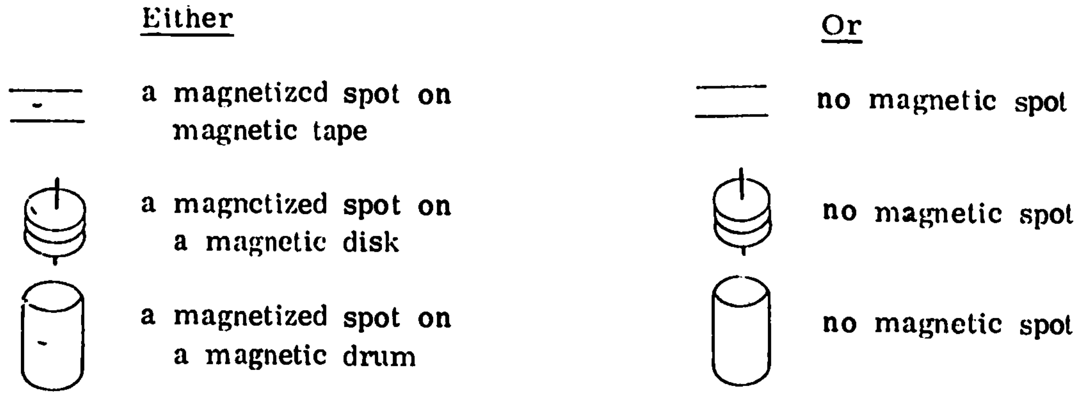

Either
Or
a magnetized spot on magnetic tape
no magnetic spot
a magnetized spot on a magnetic disk
no magnetic spot
a magnetized spot on a magnetic drum
no magnetic spot

Fig. 46 Two-state code on storage devices.

Obviously, we can use a two-state code (called a binary code) to represent all the letters of the alphabet, dollar signs and decimal points, percent signs -- in fact, any character we choose to include.

Because of the two-state nature of most computer components, a convenient system to use for representing numbers in a computer is the binary number system.* Since binary means "having two elements," we can see that this number system would be based on only two digits: zero and one. In computer slang, we call a binary digit a bit for short.

The Binary System

In the decimal number system, since decimal means "having ten elements," there are ten digits -- 0 through 9.

When counting in the decimal system, after you reach 9, you need another place to represent the number 10. Likewise, when you reach 99, you need still another place to represent the number 100, and so on.

*If the decimal system were used, 10 possible positions for each digit would be required, instead of just two. This would mean the computer would have to use 10-sprocket gears instead of simple 2-state mechanisms like switches, cores and electrical pulses.

Answers to questions on page 82:
1. (d) 2. (a, b) 3. (a) 4. (a)

The place values for the decimal number system are shown below.

100,000	10,000	1,000	100	10	1

Fig. 47 Decimal system place values.

Note that each place to the left is 10 times larger, since 10 is the base.

In the binary number system, since 2 is the base, each place to the left will be 2 times larger.

128	64	32	16	8	4	2	1

Fig. 48 Binary system place values.

To write numbers greater than 1 then, you must use more places. The numeral 2 would look like this:

128	64	32	16	8	4	2	1
						1	0

Fig. 49 The numeral 2 in binary.

This means one value of 2, and no values of 1.

The numerals through 10 look like this in binary.

Numerals	8	4	2	1
1				1
2			1	0
3			1	1
4		1	0	0
5		1	0	1
6		1	1	0
7		1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0

Fig. 50 Numbers 1 through 10 in binary.

If we wished to write the number 74 in binary, it would look like this:

64	32	16	8	4	2	1
1	0	0	1	0	1	0

Fig. 51 The number 74 in binary.

This combination of ones and zeros means "one value of 64 plus one value of 8 plus one value of 2."

You can represent a number of any size in binary, even very large numbers, if you don't mind how many places you use! In a computer, however, there is a limit to the number of binary places that may be used, which limits the size of the number we can represent.

For example, assume we have a computer which is limited to 8 bits or place value positions.

128	64	32	16	8	4	2	1

Fig. 52 Eight "bits" or place value positions.

The highest number possible in this system would be 11111111, or 255 (add 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1).

A group of 8 bits like this is called a byte. The size of a computer's storage area is usually measured in bytes. A typical computer might have more than 100,000 bytes of main storage available (100,000 bytes = 800,000 bits).

Check Your Understanding

Put a check (✓) in the box(es) beside the correct answer(s) to the following questions.

1. In main storage, information is stored on:

   □ a. magnetic tape.
   □ b. binary numbers.
   □ c. counters.
   □ d. magnetic cores.

2. Input data is carried to main storage

   □ a. in holes.
   □ b. by electrical impulses.
   □ c. by cores of data.
   □ d. in bits of data.

3. The binary code is

   □ a. a two-state code.
   □ b. used to code data for computers.
   □ c. based on a binary number system.
   □ d. all of the above.

4. The number 15 would be represented in binary by:

   □ a. 1101
   □ b. 1011
   □ c. 1001
   □ d. 1111

5. The number 60 would be represented in binary by:

   □ a. 111100
   □ b. 100101
   □ c. 101011
   □ d. none of the above.

Check your answers using the key at the bottom of page 91.

PART IV

PROCEDURES FOR COMPUTER OPERATION

PART IV PROCEDURES FOR COMPUTER OPERATION

INTRODUCTION -- A TYPICAL DAY

From starting up the computer system in the morning to filling in the last report at night, there are many routine tasks the computer operator performs every working day. To familiarize you with these tasks this section describes the work the computer operator does step-by-step through a typical day.

You will encounter the schedules and logs on which the operator keeps track of jobs, the basics of operating the computer system, and the procedures used in handling problems. Follow this typical day carefully. You will need to use many of these techniques to complete the exercise given in the last section of this manual.

Starting Up The Computer

The operator's first task of the day is to start up the computer. Each system has its own method of turning on the power which is described in the operating manual accompanying the system. The same general steps, however, are usually followed in any "power-on" procedure.

First, the computer operator always checks to see if anyone is working on equipment connected to the computer. Sometimes a repairman comes in during off-hours to do maintenance on equipment. If no one is working on equipment, the computer operator can press the power-on button of the computer located on the computer console. This switches on the central processor. The length of time it takes the computer to warm up depends on the computer system. Generally, a ready-light will go on indicating when the system is warmed up.

Each input/output device connected to the central processor usually has a power-on button also. The computer operator switches on each device by depressing its power-on button. These devices also require some time to warm up, and an indicator light tells the operator when each is ready to run. He should always wait until all the devices indicate a ready state. Each part of the computer should be ready before the operator begins to run a job.

If any device fails to turn on, the operator must try to locate and correct the problem by checking to see if the device is plugged in, if all the switches are in the correct position, and so forth. The operator's

Answers to questions on page 88:
1. (d) 2. (b) 3. (d) 4. (d) 5. (a)

manual for each piece of equipment provices guidelines for checking the machine out in such a situation. If the regular procedures fail to get the device to start, the operator calls in a customer engineer to repair the device.

Here is a summary of the steps in the "power-on" procedure.

- Power-On Steps -

1. Check to see that no one is working on equipment attached to the computer.

2. Press the power-on button.

3. Wait until a ready light is on. (The length of time for warm-up of the computer depends on the system.)

4. Press the power-on button for each input/output device and wait for each to indicate a ready condition.

5. If any device fails to power-on, call a customer engineer.

The failure of a computer device to power-on is just one of the problems to be solved by the customer engineer (or "C.E."); whenever a piece of equipment does not fuction properly and the operator cannot get it to work, the C.E. is usually called in.* Most companies which sell and lease computers offer repair services through customer engineers. Normally, the steps for obtaining good service from the C.E. are:

- Repair Call Steps -

1. The name and phone contact of your C.E. should be posted at the computer console or near the telephone.

C.E. Name ______________________________
Phone Number ____________________________
Night Number ____________________________

*In some centers, the Operations Manager has the responsibility for making any final decision to call a repairman. In such cases the computer operator must report all problems to the Operation Manager.

2. Wh·n you call, give as much information as you can. Some information the C. E. may need is:

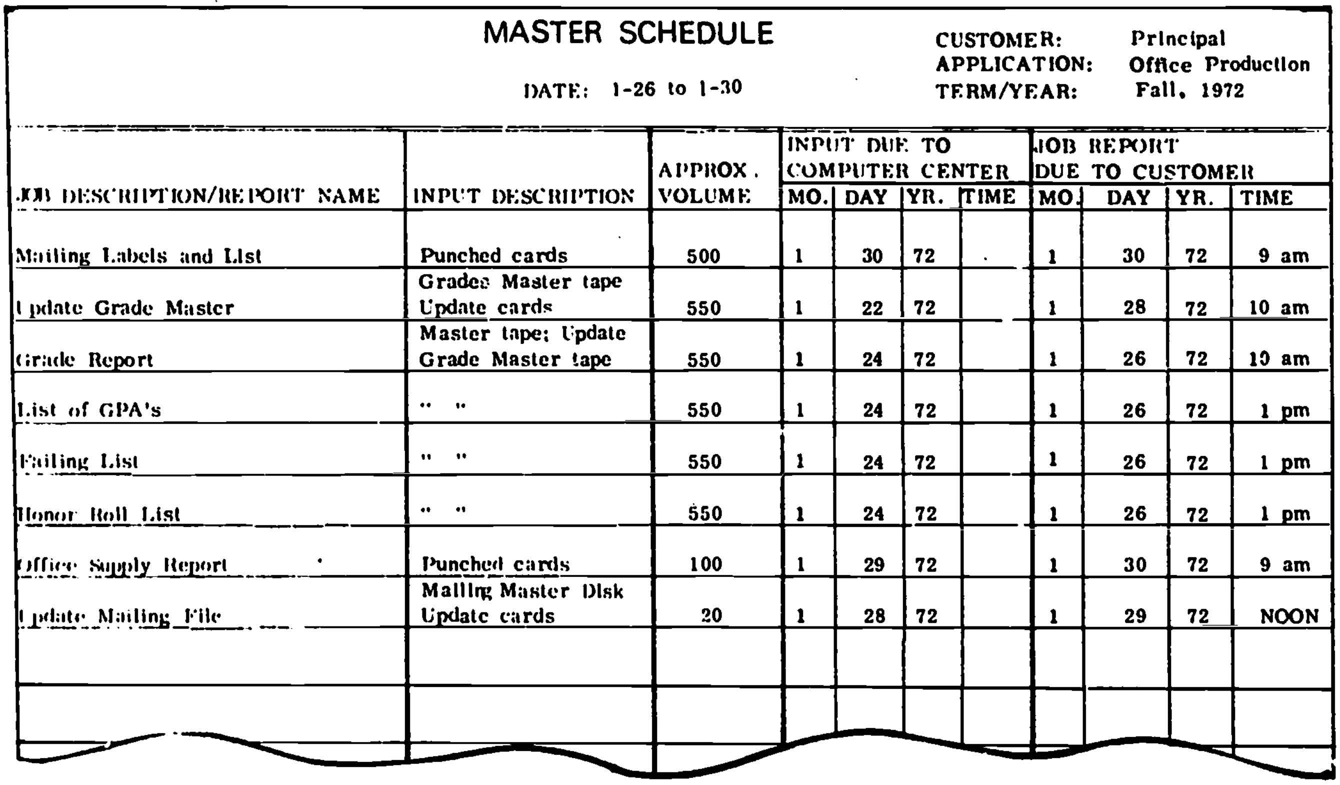
   a. Your name, company and phone number.
   b. How urgent is your request for service?
   c. If the problem is in the hardware, which device is causing the problem?
   d. If the operating system is in error, give the C. E. any error messages from the console printout.
   e. The C. E. may need directions to get to your location.

PREPARING THE DAILY WORK SCHEDULE

While the computer equipment is warming up, the operator usually makes up a schedule of the jobs to be run that day. That schedule includes the routine jobs run by the center and all special jobs requested to be run.

In most centers, the Computer Center Manager or Operations Manager provides the computer operator with a weekly or monthly Master Schedule or the routine "production" jobs which must be run that week or month and the date and time each job is due. When the operator makes his daily work schedule, the "production" jobs scheduled for that day are always given priority because they are jobs which keep the organization going, such as the calculation and printing of paychecks for a company, or the attendance and grade reports for a school.

On the next page is an example of a weekly Master Schedule of production runs a manager might give a computer operator.



MASTER SCHEDULE

DATE: 1-26 to 1-30

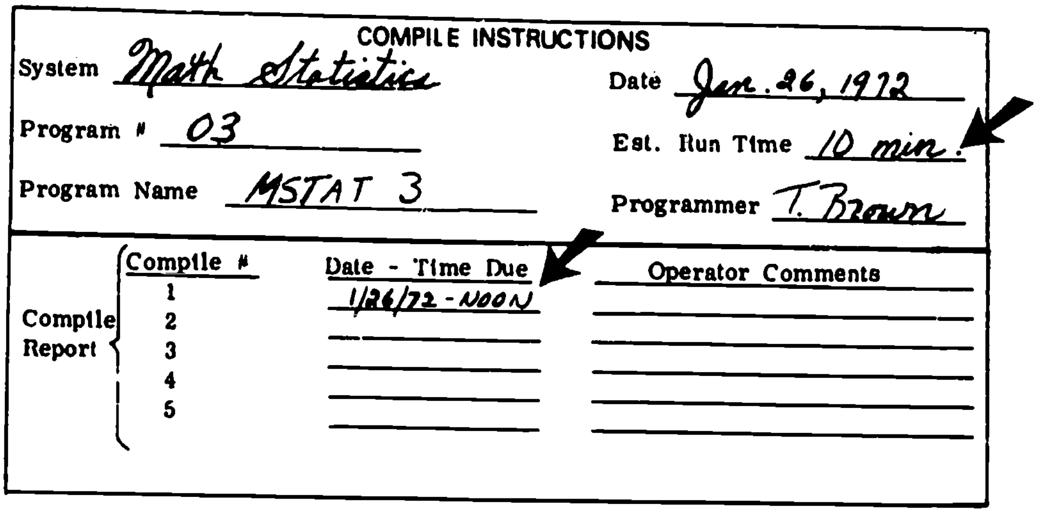
CUSTOMER: Principal
APPLICATION: Office Production
TERM/YEAR: Fall, 1972

			INPUT DUE TO COMPUTER CENTER				JOB REPORT DUE TO CUSTOMER			
JOB DESCRIPTION/REPORT NAME	INPUT DESCRIPTION	APPROX. VOLUME	MO.	DAY	YR.	TIME	MO.	DAY	YR.	TIME
Mailing Labels and List	Punched cards	500	1	30	72		1	30	72	9 am
Update Grade Master	Grades Master tape Update cards	550	1	22	72		1	28	72	10 am
Grade Report	Master tape; Update Grade Master tape	550	1	24	72		1	26	72	10 am
List of GPA's	" "	550	1	24	72		1	26	72	1 pm
Failing List	" "	550	1	24	72		1	26	72	1 pm
Honor Roll List	" "	550	1	24	72		1	26	72	1 pm
Office Supply Report	Punched cards	100	1	29	72		1	30	72	9 am
Update Mailing File	Mailing Master Disk Update cards	20	1	28	72		1	29	72	NOON

Fig. 53 Sample Master Schedule.

But the production jobs aren't the only jobs the operator may have to schedule for the day. Programmers are continually developing new programs, and part of their checking procedure is to have the computer "compile" the new program -- that is, to translate it to machine coding and to print out all problems encountered in the translating. A new program may have to be "compiled" several times before the programmer gets all the problems smoothed out.

Program compile requests are given to the computer operator in the form of an instruction (or identification) card together with the punched cards or paper tape on which the program has been prepared. The instruction card always indicates clearly what program is being compiled, approximately how long the compile run should take, and when the programmer needs the compile to be finished. On the facing page is an example of a compile instruction card, showing estimated run time and date-time due figures:



COMPILE INSTRUCTIONS

System Math Statistics Date Jan. 26, 1972

Program # 03 Est. Run Time 10 min.

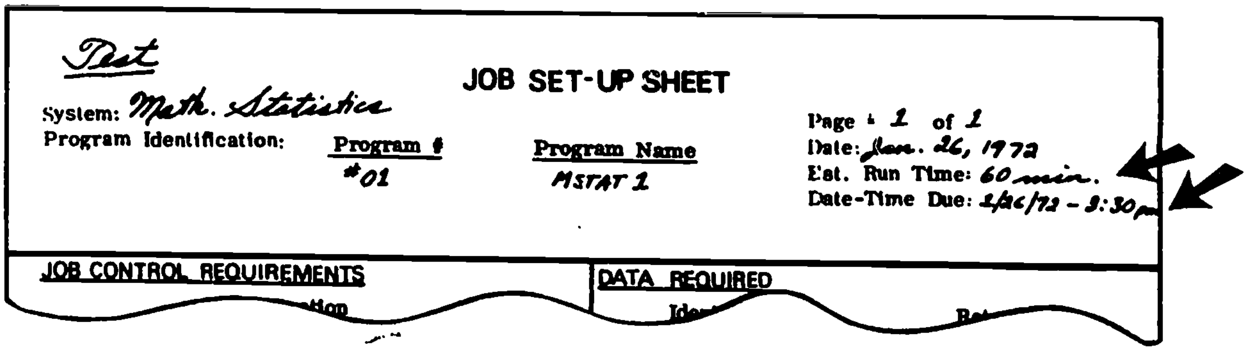
Program Name MSTAT 3 Programmer T. Brown

	Compile #	Date - Time Due	Operator Comments
Compile Report	1	1/26/72 - NOON	
	2		
	3		
	4		
	5		

Fig. 54 Sample compile instruction card.

Once a program has been correctly compiled, it still must be tested by being run with data. These are called "test" runs and constitute another kind of job which the operator may be requested to schedule and run.

Like compile requests, test run requests are given to the operator with an instruction sheet and the tapes or card decks for the program and test data. The instruction (or job set-up) sheet for test runs is usually more detailed than the instruction card used for compile runs, because test runs usually involve more things for the operator to do in setting up the equipment for the job. However, the test run instructions will always specify the approximate run time for the test and the date and time by which it must be completed. Following is an example of the top section of a test run's set-up sheet showing the run and date-time due figures.



Test

JOB SET-UP SHEET

System: Math. Statistics

Program Identification: Program # #01 Program Name MSTAT 1

Page 1 of 1
Date: Jan. 26, 1972
Est. Run Time: 60 min.
Date-Time Due: 1/26/72 - 3:30pm

JOB CONTROL REQUIREMENTS

DATA REQUIRED

Fig. 55 Top section of a sample test run instruction sheet.

Generally, compile and test runs do not have priority over production runs, but the computer operator will always try to fit them into his daily schedule to meet the programmer's time-due as nearly as possible.

In summary, then, when preparing to make up his daily schedule, the operator has the list of production jobs and the instruction sheets for all compile and test jobs to be run that day. For each job, he knows the time it is due; in addition, the instruction sheet for each compile and test job gives its estimated run time.

The approximate run time for each production job must also be known if all the jobs are to be scheduled to "fit" into the day. Usually, the experienced computer operator is familiar enough with the production runs that he knows how long each takes. But, if he doesn't know the usual run time for a given job, he can always refer to the notebook of job set-up sheets for the production runs which is kept in the computer center. These instruction sheets are usually of the same form as those used for test runs, and always show the run time for the given production job.

With the times-due and estimated run times for all production, compile and test runs for the day, the operator can proceed to set-up his daily work schedule.

Let's see how he would schedule jobs for a typical day -- say, January 26, 1972. From the Master Schedule, for the period of January 26-30, shown in Fig. 53 above, the operator sees he will need to schedule the following jobs to be run on this day:

Grade Report	--	Due: 10:00 a. m.
List of GPA'S	--	Due: 1:00 p. m.
Failing List	--	Due: 1:00 p. m.
Honor Roll List	--	Due: 1:00 p. m.

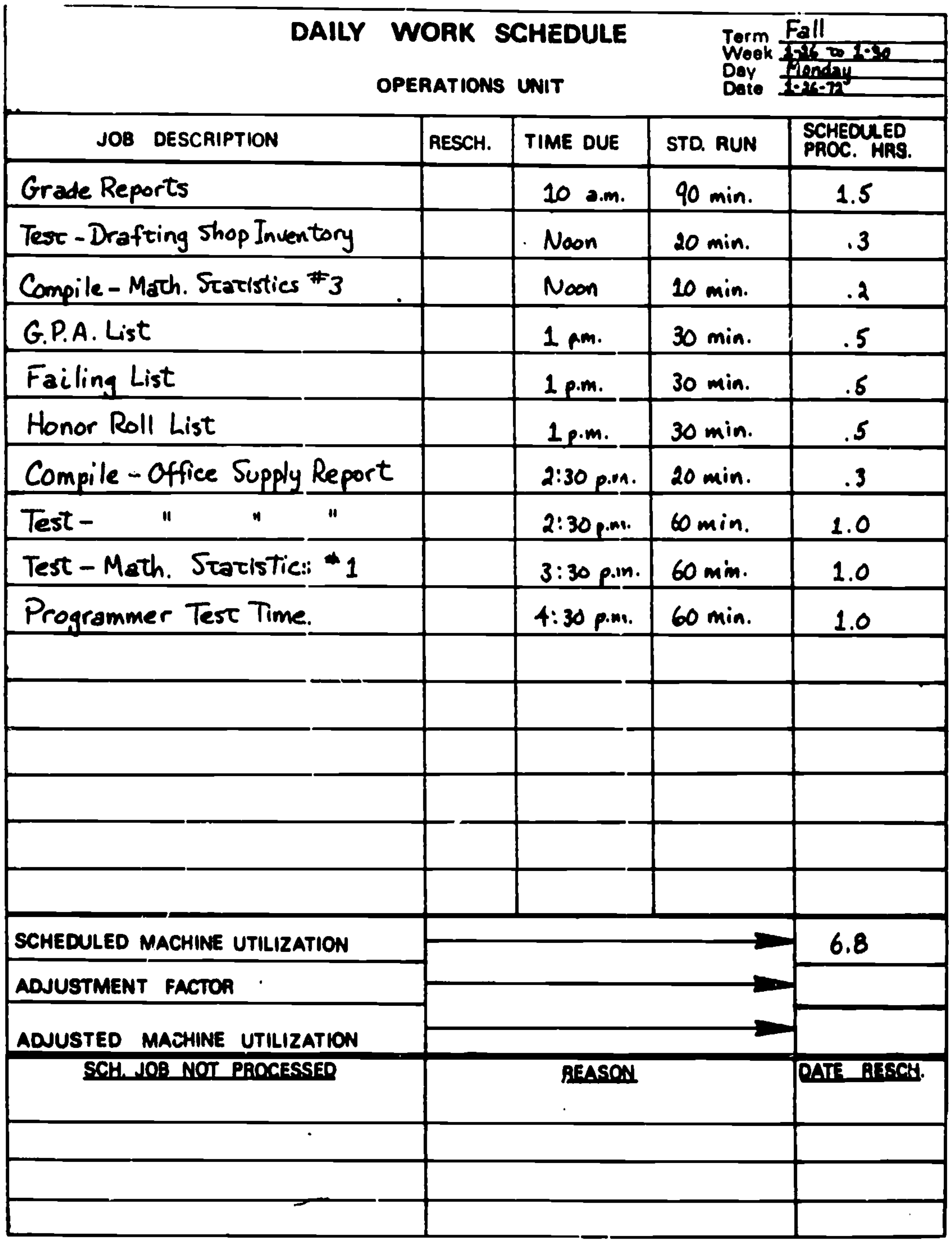
Assume he knows that the Grade Report run time is 90 minutes and he finds from the job set-up sheets that each of the three listing runs takes 30 minutes.

In addition, suppose the instruction sheets for compile and test runs left on his desk indicates that these runs must also be scheduled for January 26:

Compile -- Math Statistics, Program 03	Due: NOON	Run Time: 10 min.
Test -- Math Statistics, Program 01	Due: 3:30 p. m.	Run Time: 60 min.
Compile -- Office Supply Report	Due: 2:30 p. m.	Run Time: 20 min.
Test -- Office Supply Report	Due: 2:30 p. m.	Run Time: 60 min.
Test -- Drafting Shop Inventory	Due: NOON	Run Time: 20 min.

Finally, to make the day a really typical one, assume that one programmer has asked the computer operator to schedule 60 minutes during the afternoon in which he can make his own test runs of a program.

The operator's final daily schedule for all these runs might look like the example below.



DAILY WORK SCHEDULE

OPERATIONS UNIT

Term Fall
Week 1-26 to 1-30
Day Monday
Date 1-26-72

JOB DESCRIPTION	RESCH.	TIME DUE	STD. RUN	SCHEDULED PROC. HRS.
Grade Reports		10 a.m.	90 min.	1.5
Test - Drafting Shop Inventory		Noon	20 min.	.3
Compile - Math. Statistics #3		Noon	10 min.	.2
G.P.A. List		1 p.m.	30 min.	.5
Failing List		1 p.m.	30 min.	.5
Honor Roll List		1 p.m.	30 min.	.5
Compile - Office Supply Report		2:30 p.m.	20 min.	.3
Test - " " "		2:30 p.m.	60 min.	1.0
Test - Math. Statistics #1		3:30 p.m.	60 min.	1.0
Programmer Test Time.		4:30 p.m.	60 min.	1.0

SCHEDULED MACHINE UTILIZATION →	6.8
ADJUSTMENT FACTOR →	
ADJUSTED MACHINE UTILIZATION →	

SCH. JOB NOT PROCESSED	REASON	DATE RESCH.

Fig. 56 Sample daily work schedule.

Now, let's look at the completed scheduled in detail. First, you can see that all the production runs have been scheduled to be completed by the due time given on the Master Schedule. Programmer compile or test requests are fitted into the day so that they do not interfere with higher priority jobs, but so that they will accomodate the programmers' due-times as nearly as possible.

Notice on the schedule that the estimated run time for each job has been entered in the column labelled "Std. Run," which stands for the commonly used phrase "Standard Run Time."

Also notice the far right-hand column on the schedule headed "Scheduled Proc. Hrs." -- this heading stands for "Scheduled Process Hours" and the figures in the column are the parts of an hour (in decimal form) that each job will take. The first run, for example, has a "Std. Run" of 90 minutes and a "Scheduled Proc. Hrs." of 1.5 hours (or 1 1/2 hours.) The second run has a 20 minute "Std. Run," hence a "Scheduled Proc. Hrs." of .3 hours.*

The decimal representations of run times in the "Scheduled Proc. Hrs." column aid the operator in seeing quickly how many hours of the day have been scheduled. When the schedule is finished the operator adds up the total "Scheduled Proc. Hrs." for the day and enters the total hours on the line for Scheduled Machine Utilization.

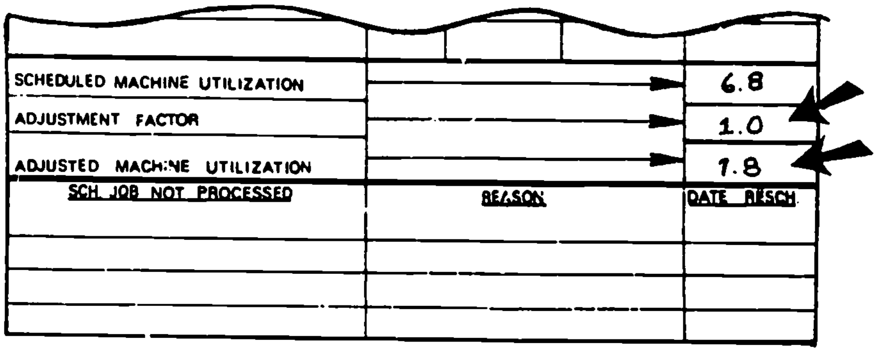
If the machine is scheduled to be used for more hours than there are in in the working day, something must be rescheduled. If all the jobs on the daily work schedule absolutely must be completed, the computer operator will work overtime to finish the daily tasks. Usually, however, one of the programmer's compile or test runs can be processed another day.

If, for any reason, a job is not completed during the day, the operator will enter that job in the space provided at the bottom of the daily work schedule and will reschedule it for later.

Notice that just under the "Scheduled Machine Utilization" line there is a line for "Adjustment Factor" and one for "Adjusted Machine Utilization." These lines are used only when some unscheduled job is run during the day. When such a run is made, the run time is written down (in decimal form) as the "Adjustment Factor."

*To translate from minutes to parts of hours, simply divide the minutes by 60 (minutes per hour and round the answer off to the nearest tenth. For example, 90 ÷ 60 = 1.5 and 20 ÷ 60 = .3

If, on a typical day, one unscheduled run were made that took one hour, the bottom of the schedule would look like this at the end of the day.



SCHEDULED MACHINE UTILIZATION	6.8
ADJUSTMENT FACTOR	1.0
ADJUSTED MACHINE UTILIZATION	7.8

SCH JOB NOT PROCESSED	REASON	DATE RESCH

Fig. 57 Adjustment figures on a daily work schedule.

Check Your Understanding

Put a check (✓) in the box(es) beside the correct answer(s) to the following questions.

1. If no one is working on the computer equipment, the operator will start up the equipment by following which of the procedures below?

   □ a. Switch on the main control switch for the entire system and everything will start.
   □ b. Switch on each piece of equipment one right after the other until they are all started.
   □ c. Switch on only the central processor. Other equipment will be started only when it is needed.
   □ d. Switch on the central processor and wait for the ready light, then proceed to switch on the other equipment.

2. Which of the following correctly describes the Master Schedule?

   □ a. It lists run times for all production runs to be made each day.
   □ b. It is usually a weekly or monthly list of production runs to be made showing the date and time each run is due.
   □ c. It gives a complete time schedule of all runs to be made on the computer each week.
   □ d. It is usually a weekly list of compile and test tuns showing when each is due.

3. Which of the following is not referred to when making up the daily work schedule.

□ a. The Master Schedule.
□ b. Compile instruction cards.
□ c. Job set-up instruction sheets.
□ d. All of the above are referred to.

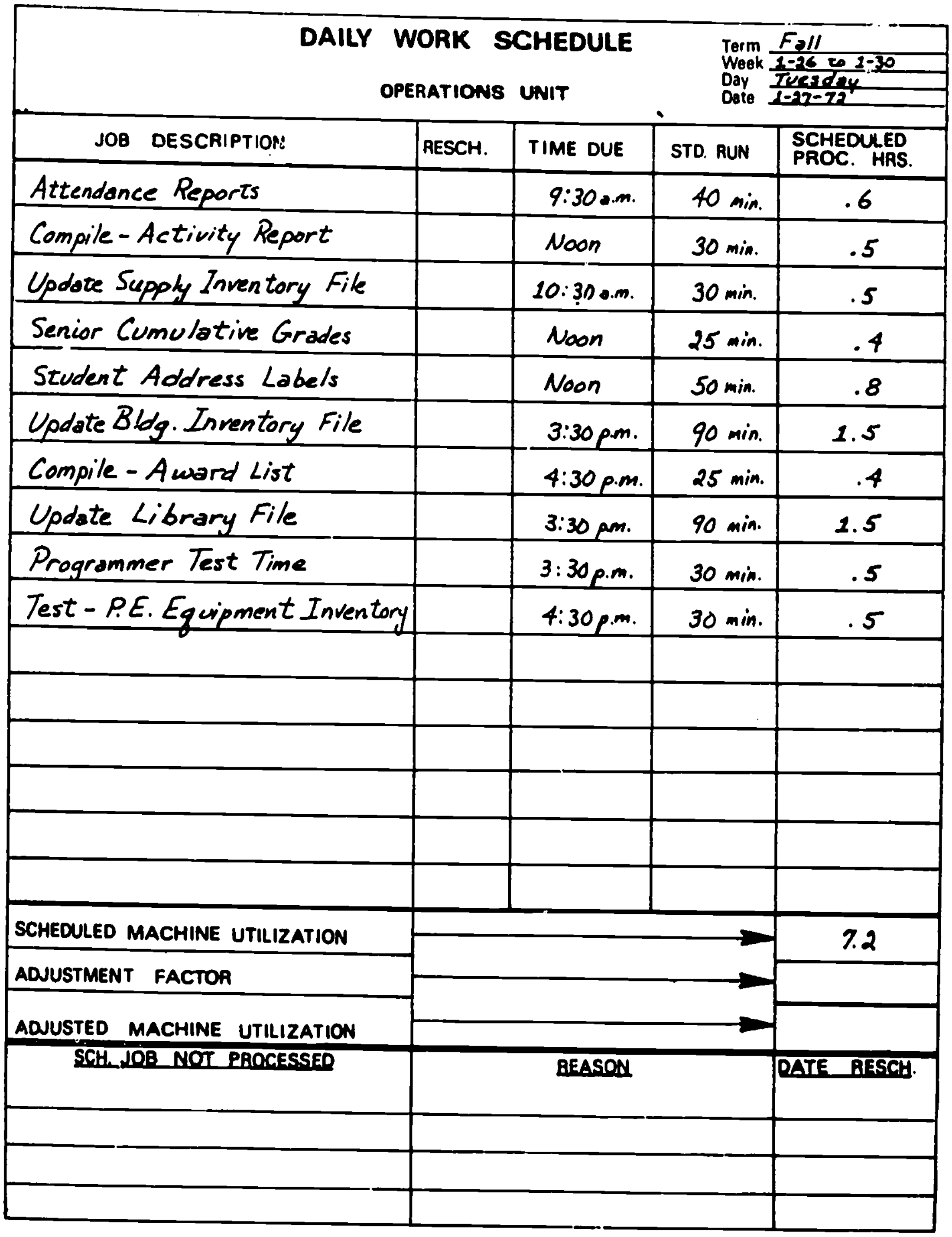
4. All of the following information is included on the usual daily work schedule, except

□ a. the time the input will be given to the operator.
□ b. the time it will take to run the job.
□ c. the title of the job.
□ d. the time the run is due to be finished.
□ e. the total hours the computer is scheduled for use.
□ f. parts of hours each job will run.
□ g. total hours the computer is actually used.

Check your answers using the key at the bottom of page 102.

5. Assume you are the computer operator who has made up the daily work schedule on the opposite page on the morning of January 27, 1972.

Suppose that, at noon, a programmer brought you a course listing program that he needed compiled in a hurry, and suppose you compiled it at 4:00 p. m., using 20 minutes of computer time. Fill in the schedule given to show how you would record this compile run on your daily schedule. Then, fill in the total number of hours the computer was actually used that day.



DAILY WORK SCHEDULE

OPERATIONS UNIT

Term Fall
Week 1-26 to 1-30
Day Tuesday
Date 1-27-72

JOB DESCRIPTION	RESCH.	TIME DUE	STD. RUN	SCHEDULED PROC. HRS.
Attendance Reports		9:30 a.m.	40 min.	.6
Compile - Activity Report		Noon	30 min.	.5
Update Supply Inventory File		10:30 a.m.	30 min.	.5
Senior Cumulative Grades		Noon	25 min.	.4
Student Address Labels		Noon	50 min.	.8
Update Bldg. Inventory File		3:30 p.m.	90 min.	1.5
Compile - Award List		4:30 p.m.	25 min.	.4
Update Library File		3:30 p.m.	90 min.	1.5
Programmer Test Time		3:30 p.m.	30 min.	.5
Test - P.E. Equipment Inventory		4:30 p.m.	30 min.	.5
SCHEDULED MACHINE UTILIZATION	→			7.2
ADJUSTMENT FACTOR	→			
ADJUSTED MACHINE UTILIZATION	→			

SCH. JOB NOT PROCESSED	REASON	DATE RESCH.

When you have completed your daily work schedule, check it against the answer shown on page 159 of the Appendix. Correct your schedule if it differs from the one shown in the Appendix.

HOUSEKEEPING CHORES

Once the computer is warmed up, there are several tasks performed by the computer operator to make the computer ready for daily runs. These are what you might call computer "housekeeping" chores. The computer's memory is cleared of old data. This task is often called "clearing core." In addition, daily dates are stored inside the computer and it is a part of the housekeeping tasks to change the date from day to day.

Both clearing core and changing the date are usually done by manipulating switches on the control panel of the computer. The exact sequence of steps in each case, however, depends on the computer model and is always described in the operating manual which comes with the system.

After the computer operator has performed the housekeeping tasks a few times, he no longer needs to refer to a list of the steps involved. Since programmers or substitute operators also use the computer on occasion, however, a list of the steps for their routine tasks is usually kept on the console for reference.

SETTING UP THE COMPUTER SYSTEM

Once the computer system has warmed up and all the housekeeping chores have been performed, the computer operator is ready to start work on the jobs of the day. There are actually four distinct steps in the complete handling of the jobs:

1. Setting up for the jobs
2. Running the jobs
3. Documenting the jobs
4. Routing the outputs

So, we'll start here to follow the operator through the first step, setting up the job, on our typical day, January 26, 1972.

The task of setting up for the day's runs has two phases: first, the gathering together of supplies and input to be used in the day's runs, and second, the setting up of the equipment with the input and with the output forms needed for the first run.

The computer operator usually starts off by checking to see that all the standard supplies for the computer equipment are in place, including paper in the console typewriter, standard printing paper in the computer

Answers to questions on pages 99-100:
1. (d) 2. (b) 3. (d) 4. (a)

printer, blank cards ready in the card punch, a scratch disk on the disk drive, blank tape in the paper tape punch and so forth. Any of these standard supplies that need replenishing would be collected from the stock room and placed in position, now.

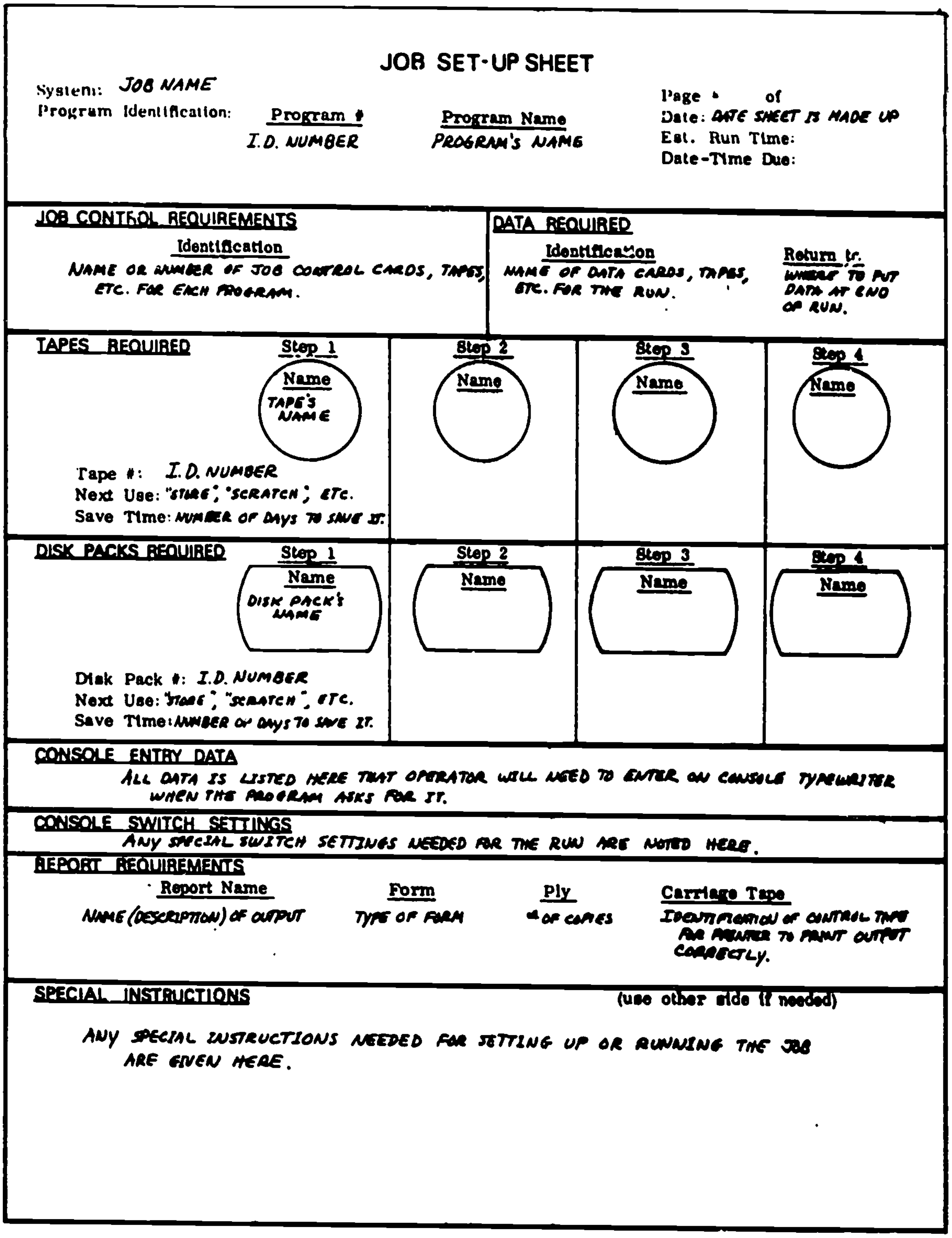
Next, the operator would gather the card decks, tape or disks, and any special forms or supplies needed to run the day's jobs. The card decks or tapes for the test or compile runs of the day are almost always right at hand, because programmers make it a practice to send their input along with the instruction sheet to the operator for each such run. So, collecting these and placing them conveniently near the computer is easy.

The card decks, tape reels and disks with the input for a center's production runs are all stored in the computer room in card cabinets or on tape and disk shelves. These are usually quite easy for the experienced operator to gather together for the day's jobs; he or she needs only tc note from the daily work schedule what production runs are scheduled to know what card decks, tape reels and disks need to be on hand at the computer. Locating them in the cabinets and on the shelves is also simple for the operator who is familiar with the computer room. Since the input for production runs is also usually described briefly on the Master Schedule (see page 94), the operator can also glance at that schedule to refresh his or her memory on the specific cards or tapes needed.

Even for an experienced operator, however, there may be new or infrequently run production jobs with which he or she is not at all familiar. In such cases, the brief description of input on the Master Schedule will not be specific enough. In these cases, the question will go to the job set-up sheets kept in a notebook in the computer room. The operator needs only to locate the job set-up sheet for the given run to get complete instructions on what input is used in the run and even what special printout forms or whatever may be needed.

The Job Set-Up Sheet

You have already seen the top section of a sample job set-up sheet. While job set-up sheets differ from computer system to computer system, they all provide the computer operator with basically the same kinds of information. Here is an example of a typical job set-up sheet, with explanations of exactly what information is entered by the programmer in each section of the sheet. Study the following example carefully.



JOB SET-UP SHEET

System: JOB NAME

Program Identification: Program # I.D. NUMBER — Program Name PROGRAM'S NAME

Page of
Date: DATE SHEET IS MADE UP
Est. Run Time:
Date-Time Due:

JOB CONTROL REQUIREMENTS

Identification
NAME OR NUMBER OF JOB CONTROL CARDS, TAPES, ETC. FOR EACH PROGRAM.

DATA REQUIRED

Identification	Return to
NAME OF DATA CARDS, TAPES, ETC. FOR THE RUN.	WHERE TO PUT DATA AT END OF RUN.

TAPES REQUIRED

Step 1	Step 2	Step 3	Step 4
Name TAPE'S NAME	Name	Name	Name

Tape #: I.D. NUMBER
Next Use: "STORE", "SCRATCH", ETC.
Save Time: NUMBER OF DAYS TO SAVE IT.

DISK PACKS REQUIRED

Step 1	Step 2	Step 3	Step 4
Name DISK PACK'S NAME	Name	Name	Name

Disk Pack #: I.D. NUMBER
Next Use: "STORE", "SCRATCH", ETC.
Save Time: NUMBER OF DAYS TO SAVE IT.

CONSOLE ENTRY DATA

ALL DATA IS LISTED HERE THAT OPERATOR WILL NEED TO ENTER ON CONSOLE TYPEWRITER WHEN THE PROGRAM ASKS FOR IT.

CONSOLE SWITCH SETTINGS

ANY SPECIAL SWITCH SETTINGS NEEDED FOR THE RUN ARE NOTED HERE.

REPORT REQUIREMENTS

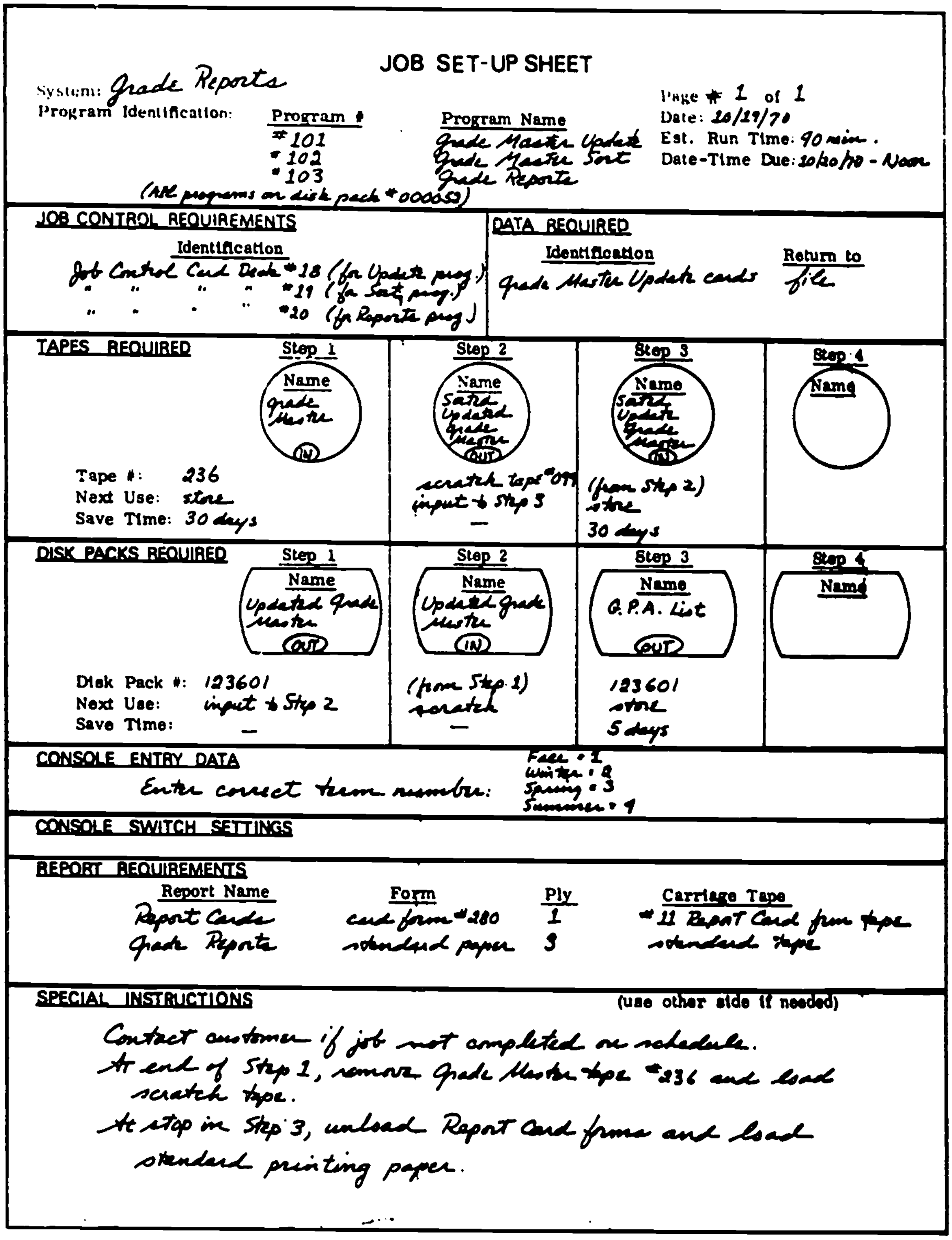
Report Name	Form	Ply	Carriage Tape
NAME (DESCRIPTION) OF OUTPUT	TYPE OF FORM	# OF COPIES	IDENTIFICATION OF CONTROL TAPE FOR PRINTER TO PRINT OUTPUT CORRECTLY.

SPECIAL INSTRUCTIONS (use other side if needed)

ANY SPECIAL INSTRUCTIONS NEEDED FOR SETTING UP OR RUNNING THE JOB ARE GIVEN HERE.

Fig. 58 Sample job set-up sheet with explanations.

Now, look at an example of the job set-up sheet which the operator might find for the Grade Reports production job on our typical day.



JOB SET-UP SHEET

System: Grade Reports

Page # 1 of 1
Date: 10/19/70
Est. Run Time: 90 min.
Date-Time Due: 10/20/70 - Noon

Program Identification:

Program #	Program Name
#101	Grade Master Update
#102	Grade Master Sort
#103	Grade Reports

(All programs on disk pack #000053)

JOB CONTROL REQUIREMENTS

Identification

Job Control Card Deck #18 (for Update prog.)
" " " " #19 (for Sort prog.)
" " " " #20 (for Reports prog.)

DATA REQUIRED

Identification	Return to
Grade Master Update cards	file

TAPES REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	Grade Master IN	Sorted Updated Grade Master OUT	Sorted Update Grade Master IN	Name
Tape #:	236	scratch tape #099	(from Step 2)	
Next Use:	store	input to Step 3	store	
Save Time:	30 days	—	30 days	

DISK PACKS REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	Updated Grade Master OUT	Updated Grade Master IN	G.P.A. List OUT	Name
Disk Pack #:	123601	(from Step 1)	123601	
Next Use:	input to Step 2	scratch	store	
Save Time:	—	—	5 days	

CONSOLE ENTRY DATA

Enter correct term number:
Fall = 1
Winter = 2
Spring = 3
Summer = 4

CONSOLE SWITCH SETTINGS

REPORT REQUIREMENTS

Report Name	Form	Ply	Carriage Tape
Report Cards	card form #280	1	#11 Report Card form tape
Grade Reports	standard paper	3	standard tape

SPECIAL INSTRUCTIONS (use other side if needed)

Contact customer if job not completed on schedule.
At end of Step 1, remove Grade Master tape #236 and load scratch tape.
At stop in Step 3, unload Report Card forms and load standard printing paper.

Fig. 59 Job set-up sheet for Grade Reports job.

From this sheet, the computer operator can determine just what input is needed to run the job.

- Production programs #101, 102, 103, (on disk pack #000052)
- Job control card decks #18, 19 and 20
- Data cards: "Grade Master Update Cards"
- 2 tapes: #236 and a "scratch" (or blank) tape #099
- Disk pack #123601

Also note that there are special supplies listed in the REPORTS REQUIREMENTS section of the sheet.

- Report card forms (with the #11 Report card form tape needed for the printer
- Standard printing paper (3-ply)

Compare these lists with the job set-up sheet in Fig. 59. Do you see where each item is located on the sheet?

Let's consider each of these items separately, so that you will know what each one is and how it would be gathered for the job.

First, whatever programs are needed for the jobs are listed at the top of the sheet. In this case, the three production programs #101, 102 and 103 are stored on a disk pack #000052. Frequently, a computer center will store all its production programs together on one disk pack and this disk pack will be so often used that it will be kept on the disk drive. If the needed disk is not on the disk drive, however, the operator would retrieve the correct disk pack #000052 and ready it on the drive.

Usually, the job control cards or tapes for production run programs are already prepared and stored in cabinets in the computer room. So when the control statements are on sets of cards, as in our example, the computer operator must get the cards from the card cabinet. The cards in our example may look like the ones illustrated on the facing page.

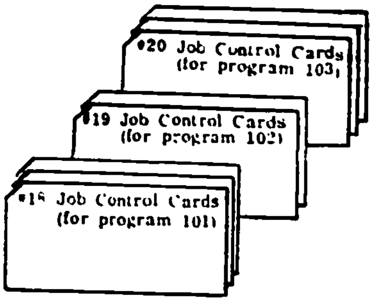


Fig. 60 Job control cards for grade report job.

The data cards for the job (in this case, the Grade Master Update cards) would probably have been received from the keypunch room and stored in a card file until the job is ready to run.

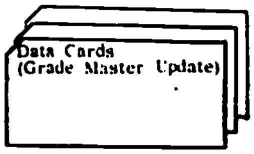


Fig. 61 Data cards for grade report job.

Once gathered, the job control card decks and the data card decks would be set conveniently near the card reader to be loaded. We will consider the order in which they are to be loaded in a minute.

Next, the magnetic tapes #236 and "scratch" tape #099 would be retrieved from the tape shelves in the computer room. In this case, tape #236 is the Grade Master tape which will be input in Step 1. (Notice the circled word "IN" in the tape figure in Step 1 on the sheet. This identifies the tape as input.) Step 2 shows that the "scratch" tape will become the "Sorted Updated Grade Master" tape that is output. (Notice "OUT" in the figure for the step.)

Once the two magnetic tapes are gathered for the run, the tape for Step 1 is mounted on the tape drive. Notice that clear instructions are given under SPECIAL INSTRUCTIONS for when, during the run, to replace tape #236 with the scratch tape on the tape drive.

Now, consider the disk packs needed for the job. From the set-up sheet it is clear that there is only one disk pack for the run beside the one with the programs on it, and that is disk pack #123601. This would be retrieved from the disk pack storage shelves and mounted on the disk drive.

It is true that there is already a disk pack on the disk drive -- the pack #000051 with production programs on it. Most disk drives, however, have two stations on which disk packs can be mounted, so the operator on our typical day can proceed to load disk pack #123601 onto the the drive without difficulty.

Finally, once all the input devices have been set-up, output forms specified in the REPORT REQUIREMENT section of the set-up sheet can be located by the operator. For the printing of report cards, it is clear that the special card form #280 must be retrieved from the stock room and loaded into the printer. In addition, the printer must be set-up with a special carriage tape, #11, which will adapt the printer to print correctly on these special forms.

Since standard, 3-ply printing paper is needed for the grade report printout the operator will also want to obtain a supply of this paper and have it on hand for use. Notice that the special instructions at the bottom of the sheet indicate when this report paper is to be loaded in the printer during the run.

You can see that gathering input and setting up the computer system is made quite simple by the instructions on the typical job set-up sheet. See if you can now interpret a job set-up sheet by yourself by answering the following questions.

Check Your Understanding

Read over the job set-up sheet on page 110 and then answer the following questions.

1. What programs are needed for the run? (Give names and identification numbers.)

____

____

2. What medium are the programs on?

____

3. What job control cards are needed? ____________________

4. Describe any data card decks needed for the run, giving the name of each.

____

____

5. What magnetic tapes are needed for the run. (Give names and numbers.)

____

____

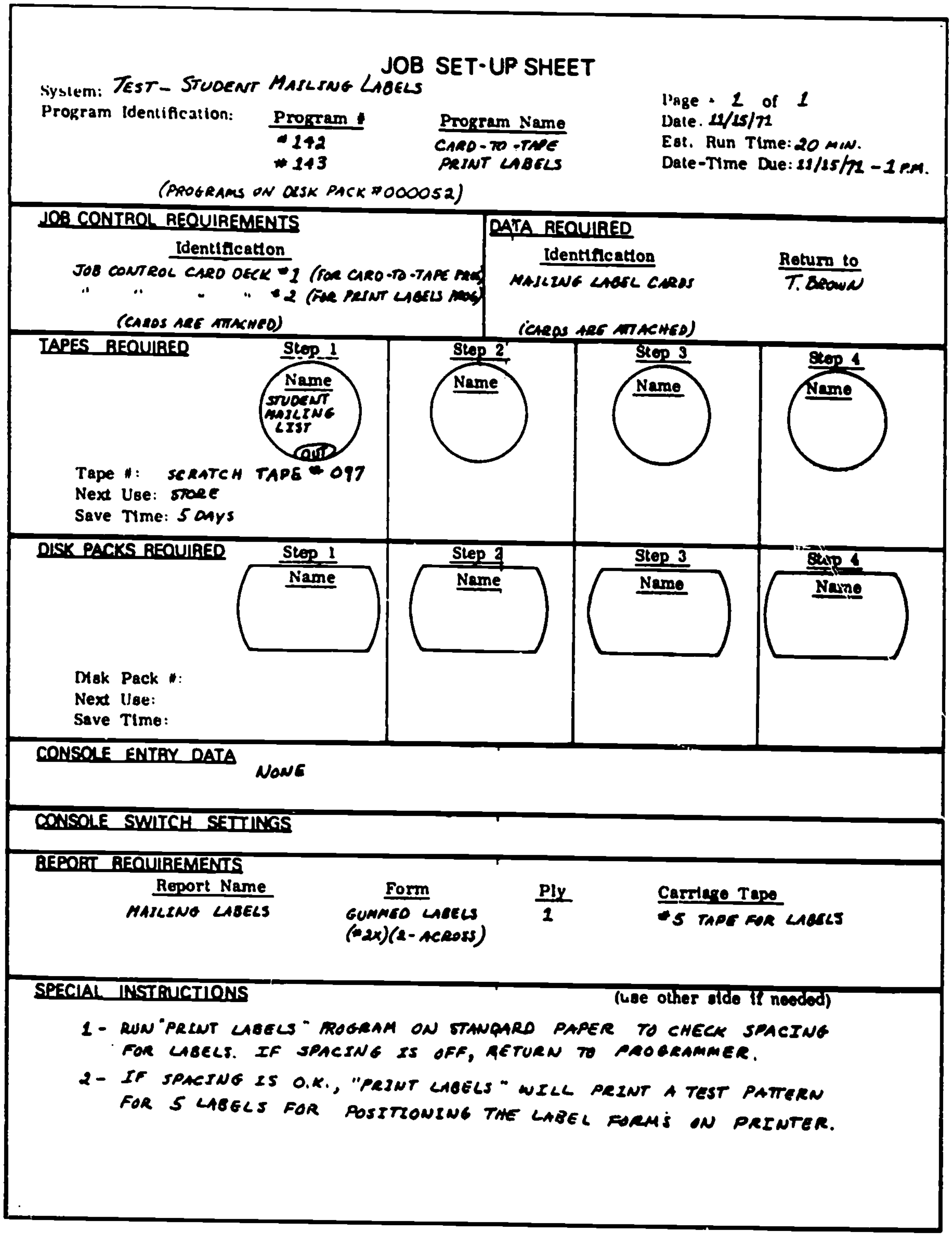
6. What disk packs are needed for the run? (Give names and numbers.)

____

7. Describe any special forms needed for this job. (Give names and descriptions of forms.)

____

____



JOB SET-UP SHEET

System: TEST – STUDENT MAILING LABELS

Program Identification:

Program #	Program Name
#142	CARD-TO-TAPE
#143	PRINT LABELS

(PROGRAMS ON DISK PACK #000052)

Page · 1 of 1
Date. 11/15/71
Est. Run Time: 20 MIN.
Date-Time Due: 11/15/71 – 1 P.M.

JOB CONTROL REQUIREMENTS

Identification

JOB CONTROL CARD DECK #1 (FOR CARD-TO-TAPE PROG)
" " " " #2 (FOR PRINT LABELS PROG)

(CARDS ARE ATTACHED)

DATA REQUIRED

Identification	Return to
MAILING LABEL CARDS	T. BROWN

(CARDS ARE ATTACHED)

TAPES REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	STUDENT MAILING LIST OUT			

Tape #: SCRATCH TAPE #097
Next Use: STORE
Save Time: 5 DAYS

DISK PACKS REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name				

Disk Pack #:
Next Use:
Save Time:

CONSOLE ENTRY DATA

NONE

CONSOLE SWITCH SETTINGS

REPORT REQUIREMENTS

Report Name	Form	Ply	Carriage Tape
MAILING LABELS	GUMMED LABELS (#2X)(2-ACROSS)	1	#5 TAPE FOR LABELS

SPECIAL INSTRUCTIONS (use other side if needed)

1 - RUN "PRINT LABELS" PROGRAM ON STANDARD PAPER TO CHECK SPACING FOR LABELS. IF SPACING IS OFF, RETURN TO PROGRAMMER.

2 - IF SPACING IS O.K., "PRINT LABELS" WILL PRINT A TEST PATTERN FOR 5 LABELS FOR POSITIONING THE LABEL FORMS ON PRINTER.

Do you think you have identified all the items of input and output forms needed for running this job? Check your answers with those given on page 160 of the Appendix.

Arranging Job Control Cards and Data Cards

We have followed the operator through the steps of gathering all the input a and the output forms and loading the disk packs, the first tape, and the first printer forms for the Grade Reports job. But, what about the loading of the job control cards and the data card deck onto the card reader?

These cards must be loaded onto the card reader in the correct order -- that is, in the order in which they will be used. The typical order is simple: before each program is run, job control cards for that program must be in the card reader, ready to be read. So, if a job calls for three programs (#1, 2 and 3) to be run one after the other, their job control cards would be loaded on the card reader in the same order.

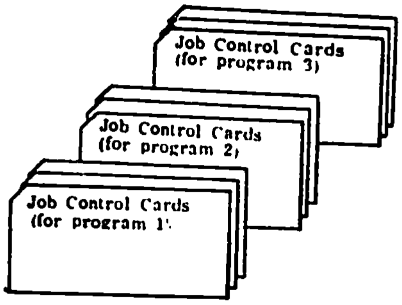

Job Control Cards (for program 3)
Job Control Cards (for program 2)
Job Control Cards (for program 1)

Fig. 62 Ordering job control cards for loading onto card reader.

If data cards are to be run with any program, the data card deck is placed just behind the job control cards for that program.

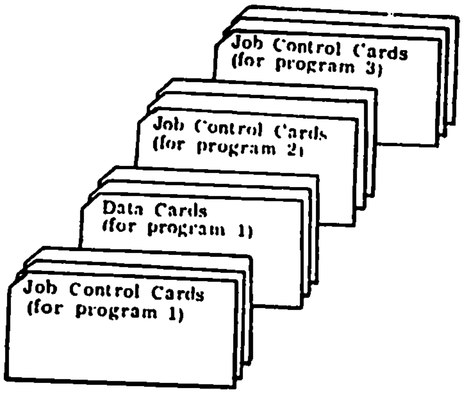
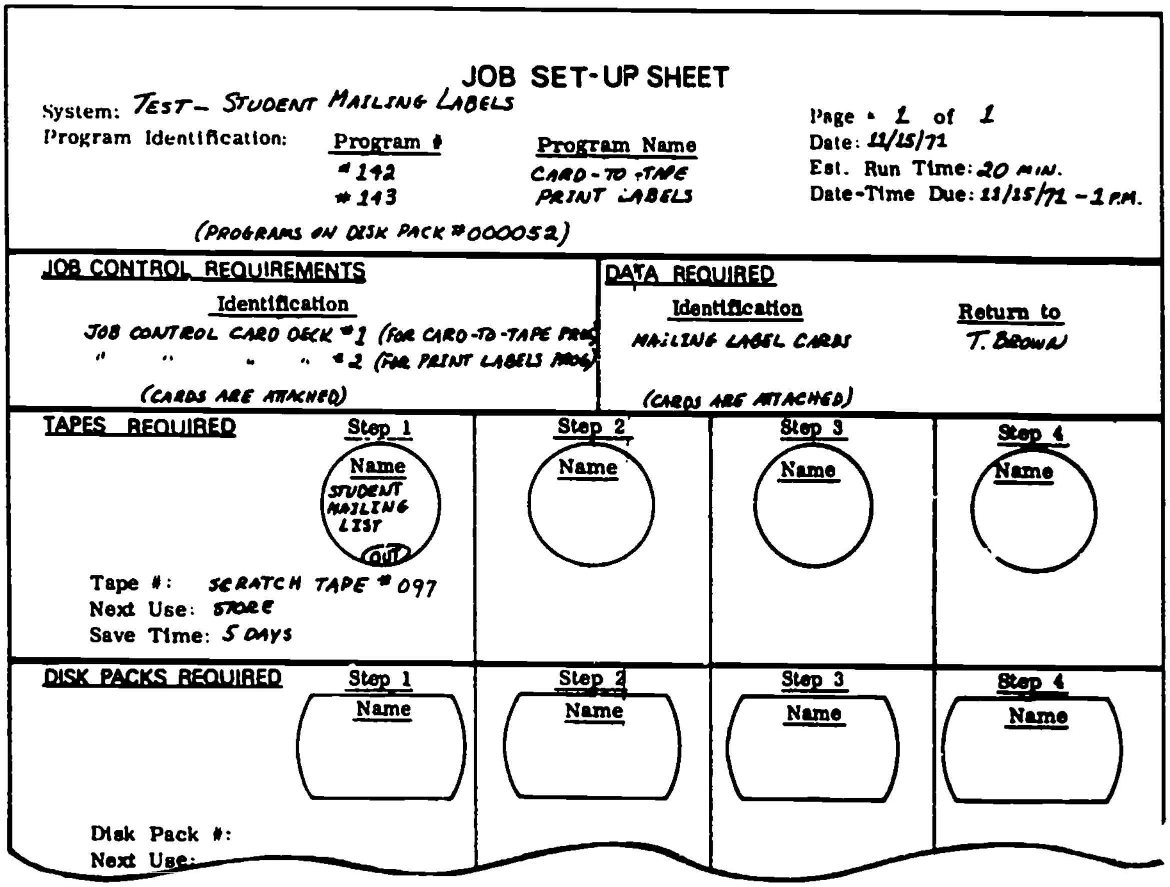

Job Control Cards (for program 3)
Job Control Cards (for program 2)
Data Cards (for program 1)
Job Control Cards (for program 1)

Fig. 63 Ordering of job control cards and data card decks for loading onto card reader.

Of course, if job control statements are to be entered on paper tape, the tapes are entered in the same order as the card decks illustrated above.

When a job is fairly simple to run, the job set-up sheet usually makes it clear how the job control cards to tape and data input must be ordered for the run. For example, look at the top portion of the job set-up sheet for the Student Mailing Labels job:



JOB SET-UP SHEET

System: TEST – STUDENT MAILING LABELS

Page • 1 of 1

Program Identification:

Program #	Program Name
#142	CARD-TO-TAPE
#143	PRINT LABELS

Date: 11/15/71

Est. Run Time: 20 MIN.

Date-Time Due: 11/15/71 – 1 P.M.

(PROGRAMS ON DISK PACK #000052)

JOB CONTROL REQUIREMENTS

Identification

JOB CONTROL CARD DECK #1 (FOR CARD-TO-TAPE PROG)
" " " " #2 (FOR PRINT LABELS PROG)

(CARDS ARE ATTACHED)

DATA REQUIRED

Identification	Return to
MAILING LABEL CARDS	T. BROWN

(CARDS ARE ATTACHED)

TAPES REQUIRED

Step 1	Step 2	Step 3	Step 4
Name STUDENT MAILING LIST OUT	Name	Name	Name

Tape #: SCRATCH TAPE #097

Next Use: STORE

Save Time: 5 DAYS

DISK PACKS REQUIRED

Step 1	Step 2	Step 3	Step 4
Name	Name	Name	Name

Disk Pack #:

Next Use:

Fig. 64 Sample job set-up sheet for simple run.

Can you see the order of events in the run? First, you can see that the input for the job includes:

- A card-to-tape program (to transfer data from "cards-to-tape.")
- A label printing program. (It will print data onto labels.)
- Job control cards; #1 and #2 for the programs.
- Data: the mailing label cards.

Clearly, the order in the run must be:

1. Card-to-tape program (#142) will process the mailing labels data cards and transfer the data onto tape.
2. The label printing program (#143) will then process the mailing list data on the tape and print it onto labels.

So, the job control and data input must be in this order: job control statements for program #142, data cards, and job control statements for program #143.

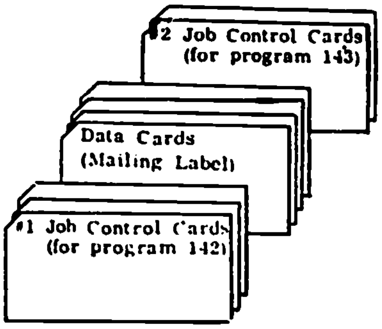
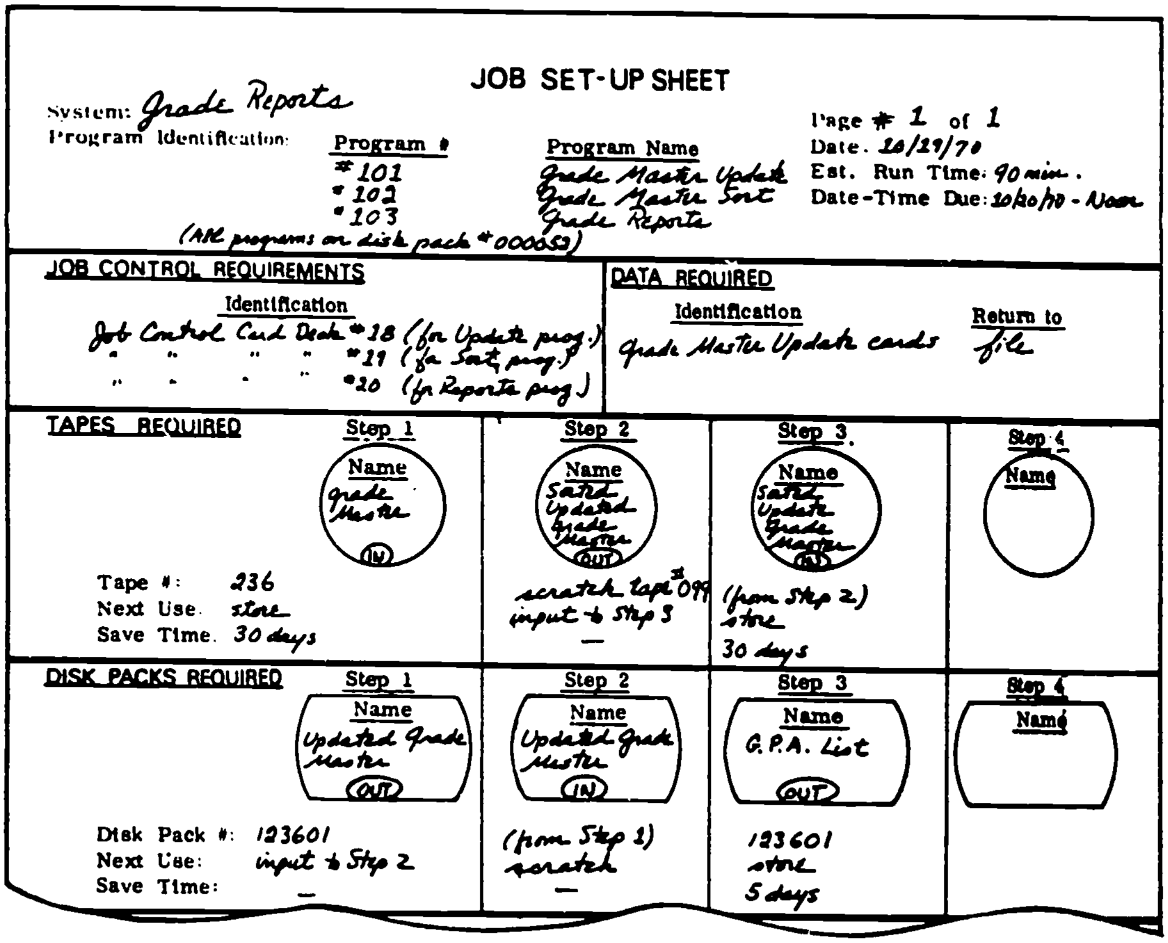


Fig. 65 Card input for Student Mailing Labels job.

Not all jobs, however, are so straightforward. For example, look again at the upper section of the Grade Report job set-up sheet. You can see that the order of events in this job is quite a lot more complex.



JOB SET-UP SHEET

System: Grade Reports

Page # 1 of 1

Program Identification:

Program #	Program Name
#101	Grade Master Update
#102	Grade Master Sort
#103	Grade Reports

(All programs on disk pack #000653)

Date. 10/29/70

Est. Run Time: 90 min.

Date-Time Due: 10/30/70 - Noon

JOB CONTROL REQUIREMENTS

Identification
Job Control Card Deck #18 (for Update prog.)
" " " " #19 (for Sort prog.)
" " " " #20 (for Reports prog.)

DATA REQUIRED

Identification	Return to
Grade Master Update cards	file

TAPES REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	Grade Master IN	Sorted Updated Grade Master OUT	Sorted Update Grade Master IN	
Tape #:	236	scratch tape #099	(from Step 2)	
Next Use.	store	input to Step 3	store	
Save Time.	30 days	—	30 days	

DISK PACKS REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	Updated Grade Master OUT	Updated Grade Master IN	G.P.A. List OUT	
Disk Pack #:	123601	(from Step 1)	123601	
Next Use:	input to Step 2	scratch	store	
Save Time:	—	—	5 days	

Fig. 66 Upper section of Grade Report set-up sheet.

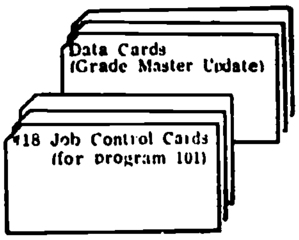
Only careful study of the set-up sheet will clarify the flow of events in this job and therefore make it clear how the job control cards and data cards should be ordered.

Look at the information given on this set-up sheet, paying close attention to what data on tape or disk is input (marked "IN") and what is output (marked "OUT") in each step of the run. Try to see the relationship between the programs (listed at the top of the sheet) and the data on cards, tapes, and disks that are processed. With careful study you should see that the flow of events and the consequent order of job control and data cards are as follows:

Step 1 -- The Grade Master tape #236 is input and Updated Grade Master data is output on disk. This means simply that the data on tape #236 is to be updated with new data and output onto disk. To update it, obviously the Grade Master Update program #101 must be used along with the Grade Master Update data cards.

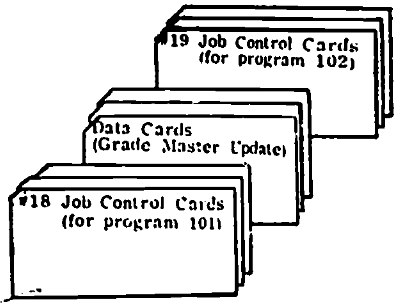
The job control cards for the Update program #101 would therefore be placed at the front of the card stack for this job and the deck of update cards would go second.

Card Order Begins



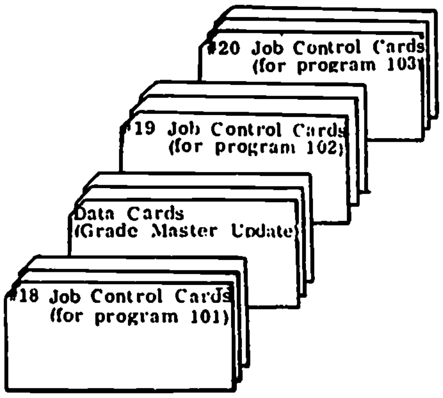
Step 2 -- The Updated Grade Master data on disk (created in Step 1) is input and Sorted Updated Grade Master data is output on tape. To sort the data, clearly the Grade Master Sort program #102 must be used. So, the job control cards for the Sort program would be placed next in the stack for this job.

Card Order Becomes



Step 3 -- The Sorted Grade Master data now on tape (created in Step 2) is input and the G. P. A. data is output on disk while two reports are printed. What program must be used to process the Sorted Grade Master data to produce the G. P. A. data and print the report? It must be the Grade Reports program #103, because it has not yet been used and there is no further step in which to use it. So, the job control cards for this program are placed at the end of the stack.

Card Order Becomes



You can see from this example that the job set-up sheet clearly identifies the input data and the tapes and disk packs needed for even a complex job, but it may not give a very clear picture of the step by step flow of programs and data in the job. Consequently, for complex jobs, programmers usually provide not only a job set-up sheet but also a flowchart which clearly shows the sequence of events in the run.

Such flowcharts are usually called job stream flowcharts. Here is an example of a job stream flowchart for the Grade Reports job.

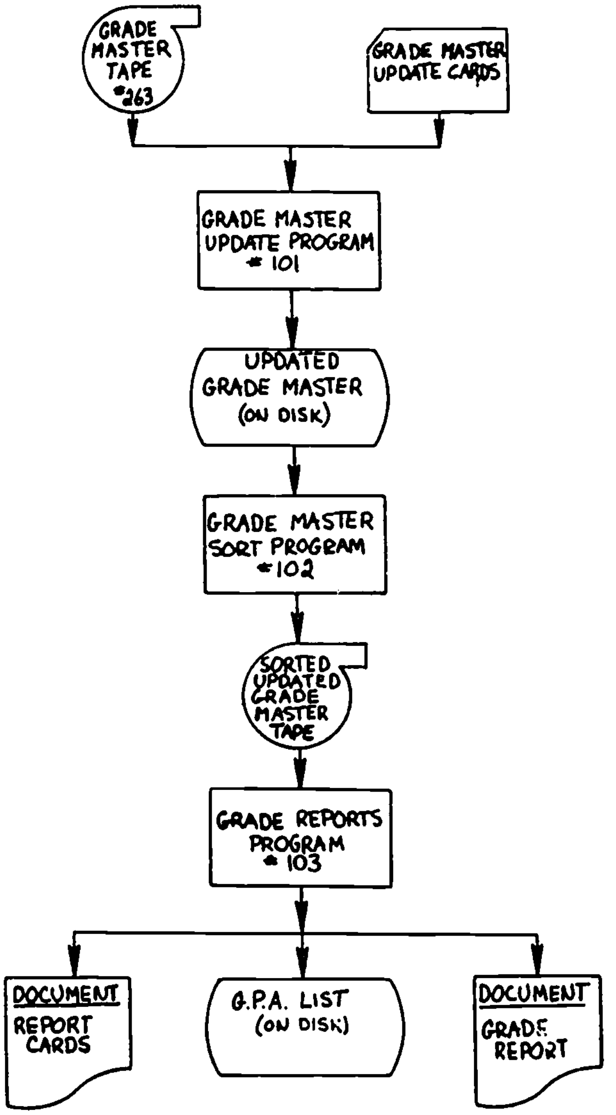

GRADE MASTER TAPE #263
GRADE MASTER UPDATE CARDS
GRADE MASTER UPDATE PROGRAM # 101
UPDATED GRADE MASTER (ON DISK)
GRADE MASTER SORT PROGRAM #102
SORTED UPDATED GRADE MASTER TAPE
GRADE REPORTS PROGRAM # 103
DOCUMENT
REPORT CARDS
G.P.A. LIST (ON DISK)
DOCUMENT
GRADE REPORT

Fig. 67 Job stream flowchart for Grade Reports job.

Notice how clear this flowchart makes the flow between the programs and the data on tapes, cards and disks. In addition, the chart also shows the output very clearly: two documents (report cards and grade reports) and a G.P.A. list (of data) on disk.

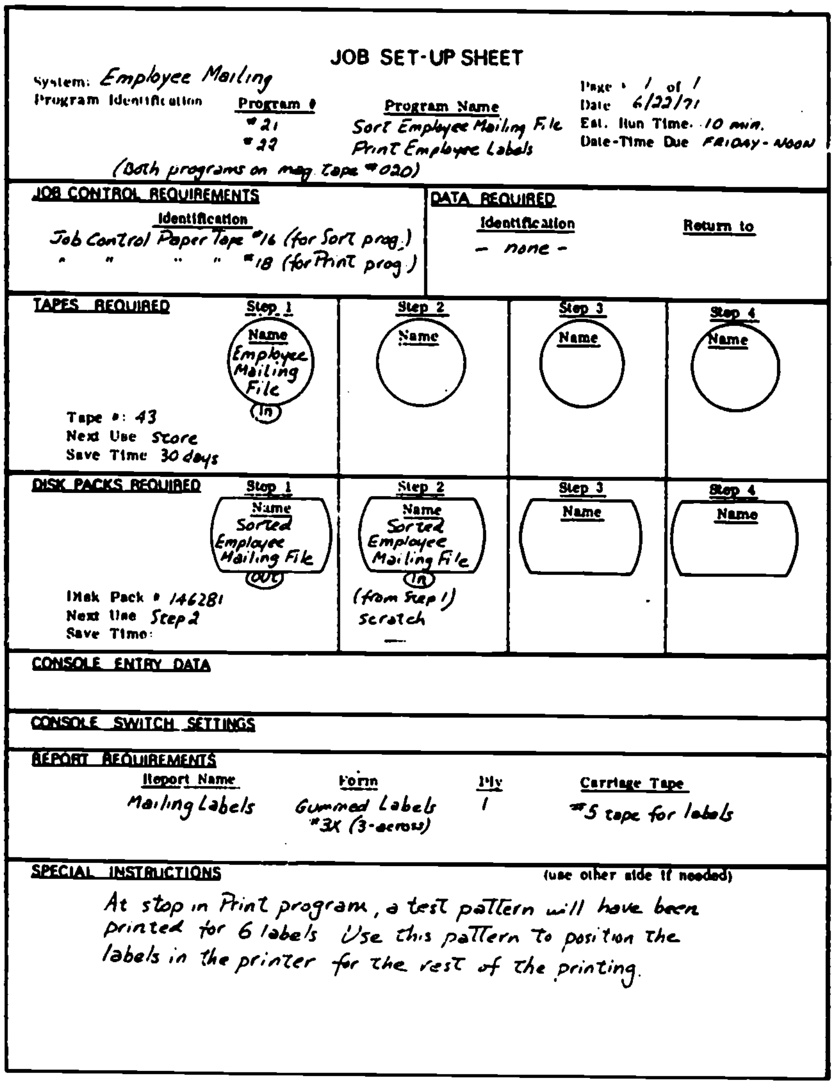
If you had this flowchart to work from, you would have no doubt what programs were used when and with what data cards. Consequently, the ordering of job control cards and data cards would be very simple. The job control cards would be put in the same order as the programs they control. Any data card deck would then be inserted behind the job control cards for the program that is used to process it.

Check Your Understanding

1. Show how you would order the following job control cards and data cards for a job by writing the numbers 1, 2, 3, 4, and 5 on the line beside the appropriate cards.

    ____Job control cards for second program
    ____Job control cards for third program
    ____Job control cards for first program
    ____Data cards for first program
    ____Data cards for second program

2. Study the job set-up sheet and the job stream flowchart shown below for Student Mailing Labels job.



JOB SET-UP SHEET

System: *Employee Mailing*

Page *1* of *1*
Date *6/22/71*
Est. Run Time *10 min.*
Date-Time Due *FRIDAY - NOON*

Program Identification

Program #	Program Name
#21	*Sort Employee Mailing File*
#22	*Print Employee Labels*

(Both programs on mag. tape #020)

JOB CONTROL REQUIREMENTS

Identification
Job Control Paper Tape #16 (for Sort prog.)
" " " " #18 (for Print prog.)

DATA REQUIRED

Identification	Return to
- none -	

TAPES REQUIRED

Step 1	Step 2	Step 3	Step 4
Name *Employee Mailing File* *In*	Name	Name	Name

Tape #: *43*
Next Use *Store*
Save Time *30 days*

DISK PACKS REQUIRED

Step 1	Step 2	Step 3	Step 4
Name *Sorted Employee Mailing File* *OUT*	Name *Sorted Employee Mailing File* *In* *(from Step 1)* *Scratch* *—*	Name	Name

Disk Pack # *146281*
Next Use *Step 2*
Save Time:

CONSOLE ENTRY DATA

CONSOLE SWITCH SETTINGS

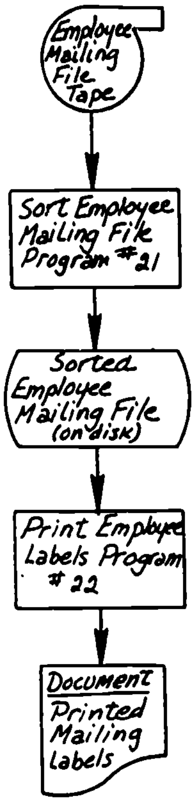
REPORT REQUIREMENTS

Report Name	Form	Ply	Carriage Tape
Mailing Labels	*Gummed Labels* *#3X (3-across)*	*1*	*#5 tape for labels*

SPECIAL INSTRUCTIONS (use other side if needed)

At stop in Print program, a test pattern will have been printed for 6 labels Use this pattern to position the labels in the printer for the rest of the printing.

Job Set-Up Sheet


Employee Mailing File Tape
Sort Employee Mailing File Program #21
Sorted Employee Mailing File (on disk)
Print Employee Labels Program #22
Document Printed Mailing Labels

Job Stream Flowchart

a. Describe any job control cards or tape and/or data cards or tapes needed for this job. (Give names and identification numbers.)

____

____

____

b. In what order would these cards or tapes by loaded into a card or tape reader? List them (by name and number) to show the correct order:

i) ____________________________________

ii) ____________________________________

iii) ____________________________________

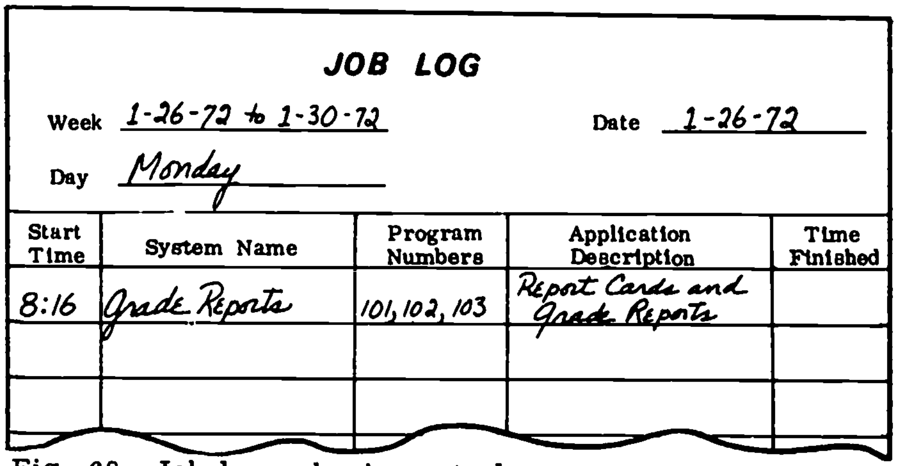
Check your answers using the key at the bottom of page 121.

The Job Log

Now, returning to our typical day, let's assume the operator has loaded not only the tapes, disk and output forms, but also the punched card input. At this point, there is only one more task he must do before he can run the job.

As you saw in the section about making out a daily work schedule, certain times (such as standard run time or estimated run time) were important. These time-notations allowed the operator to find an estimate of computer time to be used during the day. But, for each day's work, the computer center and company management need to know how much computer time is actually being used on each job. To keep track of actual run times for each job during the day the computer operator usually records the actual time a job is started and the time it is completed is a log called a "job-log."

The typical job log has a place for identifying the system and programs being run. The computer operator fills in the description of the job on the job log and records the actual time he is starting the job as is shown in the illustration on the following page.



JOB LOG

Week 1-26-72 to 1-30-72 Date 1-26-72

Day Monday

Start Time	System Name	Program Numbers	Application Description	Time Finished
8:16	Grade Reports	101,102,103	Report Cards and Grade Reports	

Fig. 68 Job log, showing actual "starting time" for a job.

Now that the actual start time has been recorded, the computer operator is ready to run the Grade Reports job.

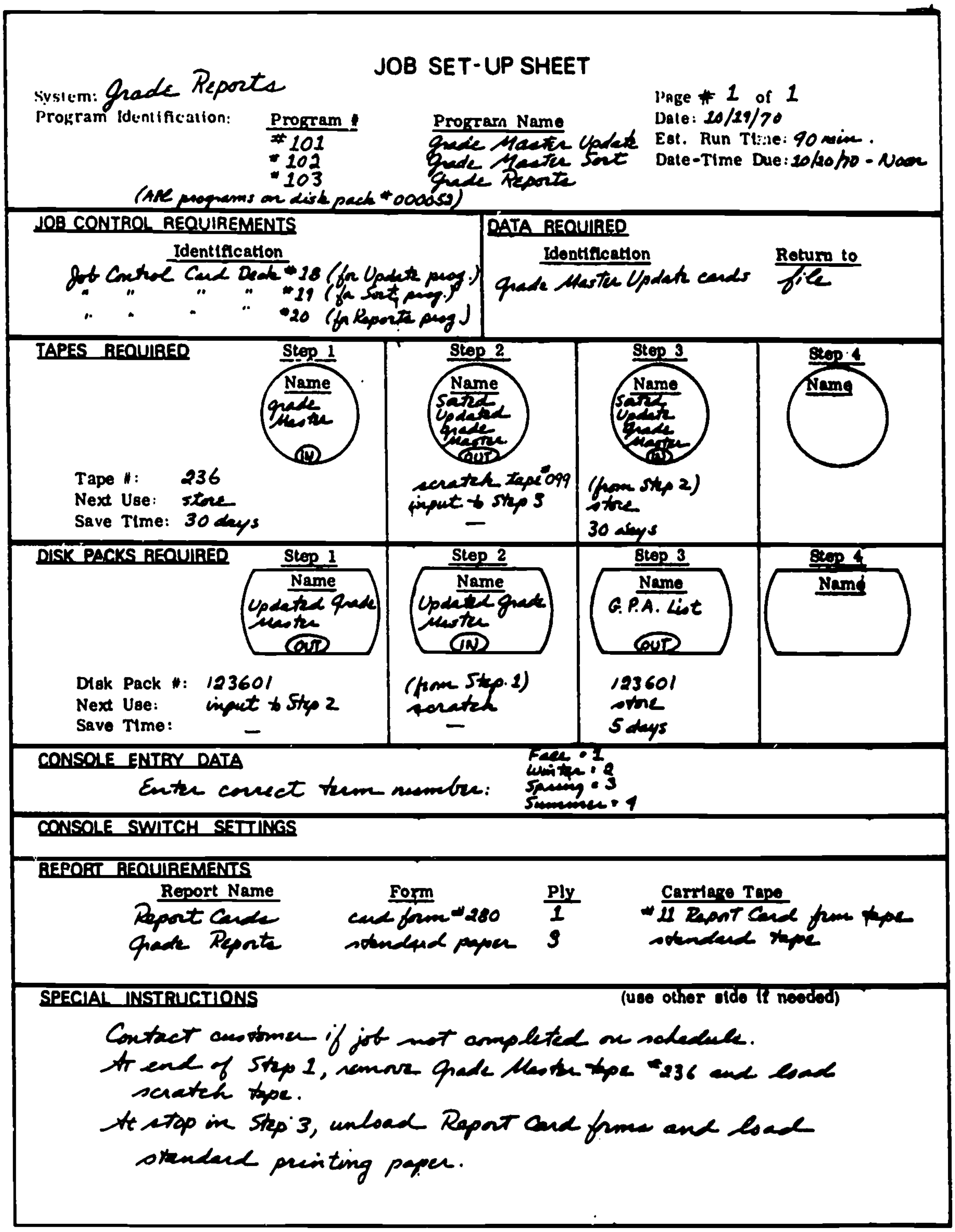
RUNNING THE JOB

To start a job running, the computer operator must always enter an instruction on the console typewriter to tell the computer to start reading the job control cards for the first program.

The specific instruction to read the job control cards is unique to each computer system. In addition, some systems require a read-instruction for only the first set of job control cards and some require an instruction for each set of job control cards in the stack. The correct use of instructions for a given system is always described in the operator's manual for the particular system.

Once the run is started, the operator's responsibility becomes simply one of monitoring the process to see that everything is proceeding correctly and of making any mid-run console typewriter entries or media changes that are required by the job. All such operator interventions are made clear on the set-up sheet for the particular job.

For example, from the set-up sheet for the Grade Reports job, it is clear that only three interventions need to be made by the operator in mid-run. Look at the copy of the set-up sheet on the next page and see if you can spot them.



JOB SET-UP SHEET

System: Grade Reports

Page # 1 of 1
Date: 10/19/70
Est. Run Time: 90 min.
Date-Time Due: 10/20/70 - Noon

Program Identification:

Program #	Program Name
#101	Grade Master Update
#102	Grade Master Sort
#103	Grade Reports

(All programs on disk pack #000053)

JOB CONTROL REQUIREMENTS

Identification

Job Control Card Deck #18 (for Update prog.)
" " " " #19 (for Sort prog.)
" " " " #20 (for Reports prog.)

DATA REQUIRED

Identification	Return to
Grade Master Update cards	file

TAPES REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	Grade Master (IN)	Sorted Updated Grade Master (OUT)	Sorted Update Grade Master (IN)	
Tape #:	236	scratch tape #099	(from Step 2)	
Next Use:	store	input to Step 3	store	
Save Time:	30 days	—	30 days	

DISK PACKS REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	Updated Grade Master (OUT)	Updated Grade Master (IN)	G.P.A. List (OUT)	
Disk Pack #:	123601	(from Step 1)	123601	
Next Use:	input to Step 2	scratch	store	
Save Time:	—	—	5 days	

CONSOLE ENTRY DATA

Enter correct term number: Fall = 1
Winter = 2
Spring = 3
Summer = 4

CONSOLE SWITCH SETTINGS

REPORT REQUIREMENTS

Report Name	Form	Ply	Carriage Tape
Report Cards	card form #280	1	#11 Report Card form tape
Grade Reports	standard paper	3	standard tape

SPECIAL INSTRUCTIONS (use other side if needed)

Contact customer if job not completed on schedule.
At end of Step 1, remove Grade Master tape #236 and load scratch tape.
At stop in Step 3, unload Report Card forms and load standard printing paper.

Fig. 69 Grade Reports job set-up sheet.

Answers to questions on pages 118-119: 1. (3,5,1,2,4) 2. (a). (Job Control Paper Tapes #16 and #18. No data cards or tapes) (b)(i) (Job Control Paper Tape #16) (b)(ii) (Job Control Paper Tape #18) (b)(iii) (none)

The first intervention is indicated under "Console Entry Data." The instructions here mean that, when asked on the console typewriter, the operator must enter the current school term, using the code "1" if it is a fall term, "2" if winter, "3" if spring, and "4" if summer. The console entry is entered into the computer on the console typewriter.

The next two operator interventions are described under "Special Instructions" on the set-up sheet. The first says when to remove the Grade Master tape and to load the scratch tape. If only one tape drive is available, here's where we solve the problem of having two magnetic tapes in one job. When the Update Grade Master program is completed, the computer will stop. At this point, the operator must remove the Grade Master tape from the tape drive and load a blank tape, called a "scratch tape" by most computer operators.

The second instruction indicates that a change in output media must be made in Step 3. Here's a case where a programmer has placed a "Stop" instruction in the middle of a program. The computer will stop to allow the operator to replace the report card form with standard printer paper. He can then start the computer run again and the printer will begin printing the final output, the Grade Report.

The entire job process is shown in the chart found below. The chart describes the interaction of the operator, the central processor and the input/output devices as the job is run.

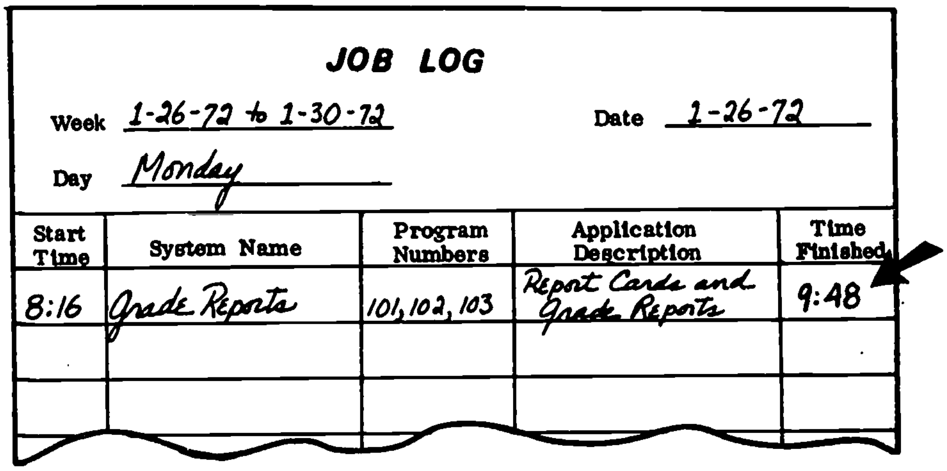
OPERATOR IN ACTION	CPU IN ACTION	I/O DEVICE IN ACTION
On console typewriter, enters instruction to read job control cards for Grade Master Update program.		
		Card reader reads job control cards.
	Gets Grade Master Update program.	Disk drive reads Update program.
		Console typewriter types request for term number.
Enters correct number for current term.		
		Card reader reads in Update cards.
	Matches Update cards to Grade Master tape.	Tape drive reads Grade Master tape. Disk drive writes Updated Grade Master.
	Stops	
Removes Grade Master tape; loads scratch tape. Starts computer running again.		
		Card reader reads in job control cards.
	Gets Sort program.	Disk drive reads Sort program.
	Sorts.	Disk drive reads Updated Grade Master.
	Stops	
Gives instruction to read job control cards		
		Card reader reads job control cards.
	Gets Grade Report program.	Disk drive reads Grade Report program.
		Printer prints Report Cards. Disk drive writes G. P. A. List.
	Stops	
Changes printer to standard paper. Starts computer again.		
		Printer prints Grade Report.
	Stops.	

Chart 70 Chart of job process.

Documenting a computer job involves such tasks as noting on the job log the time the job is completed and labelling all card decks, tapes and disks. If any computer problems or computer operator errors occur during the run process, the nature of the problem may be recorded on a trouble log, which we will discuss later.

Filling In the Job Log

Looking again at the job log, you will see a column for the time the job was completed. As the first step in documenting the job, the computer operator logs in the time the job was finished as shown on the job log below.



JOB LOG

Week 1-26-72 to 1-30-72 Date 1-26-72

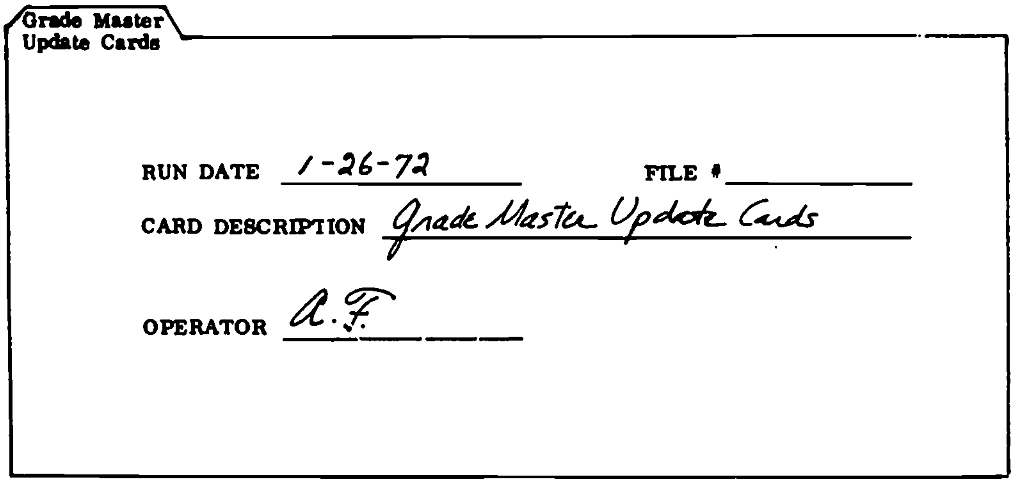
Day Monday

Start Time	System Name	Program Numbers	Application Description	Time Finished
8:16	Grade Reports	101, 102, 103	Report Cards and Grade Reports	9:48

Fig. 71 Job log, with "Time Finished" column filled in.

Labelling Input and Output

Labelling cards, tapes and disks for storage is the computer operator's next documenting task. A tabbed card, like the one illustrated on the next page, is usually used to identify card decks in the card-cabinet. When documenting the job, the operator would naturally fill-out such a tabbed card for any new card input or output to be stored in the cabinet.



Grade Master
Update Cards

RUN DATE 1-26-72 FILE # ______

CARD DESCRIPTION Grade Master Update Cards

OPERATOR A.F.

Fig. 72 Tabbed card for file of card decks.

A gummed label should then be made for each new tape or disk output which must be saved for future use. The labels are usually marked with the identifying name and number of the program or data. In the case of the Grade Report job, for example, there is a final output tape to be labelled "Sorted Updated Grade Master" and a disk file to be identified on the disk pack label as "G. P. A. List." The illustration below shows how a tape or disk pack may look after they are labelled.

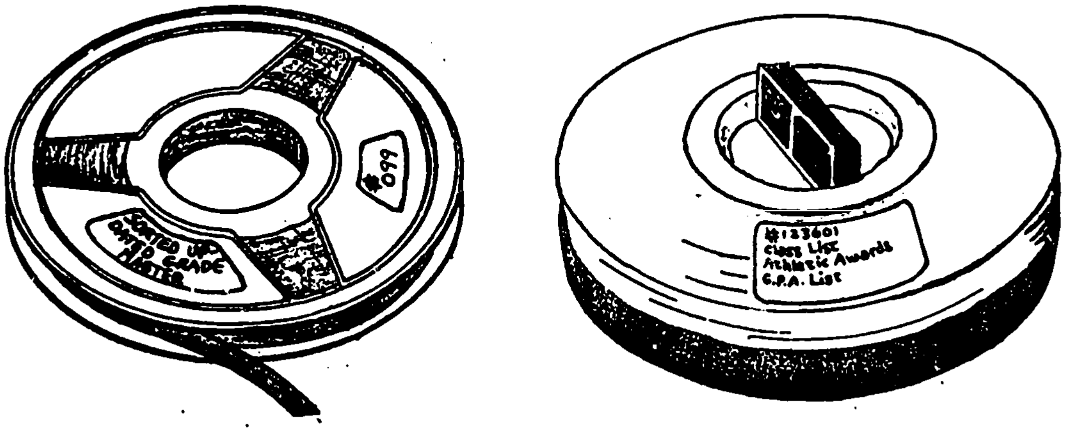


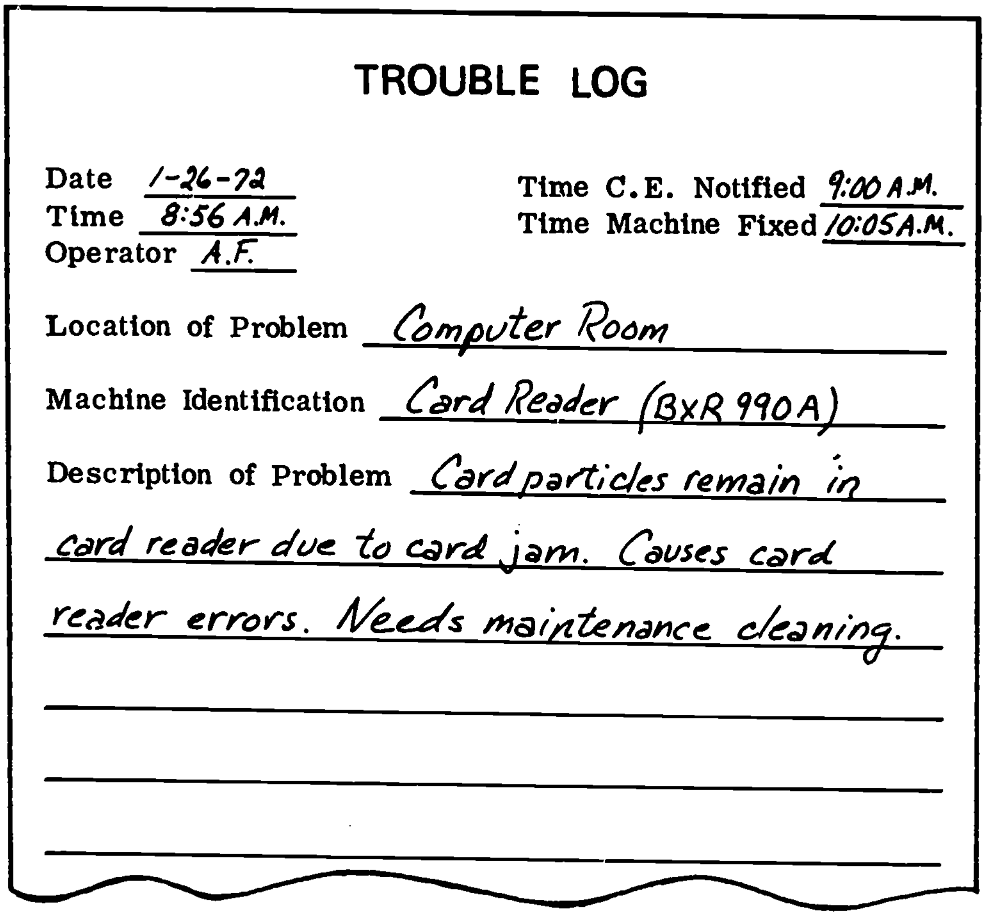
Fig. 73 Magnetic tape and disk pack with labels.

The Trouble Log

Suppose during the Grade Reports run, one of the computer devices failed to operate. Sometimes, the computer operator can bypass problems if they are minor ones. Other times he may have called a C. E.

A card jam is a common problem reported by computer operators. Sometimes when a card jams, card particles remain to cause card reader errors. The computer operator may bypass the problem, but should still note it in the trouble log. The trouble log serves as a maintenance guide to the customer engineer.

The card jam might be entered in the trouble log as it is in the illustration below.



TROUBLE LOG

Date 1-26-72
Time 8:56 A.M.
Operator A.F.

Time C.E. Notified 9:00 A.M.
Time Machine Fixed 10:05 A.M.

Location of Problem Computer Room

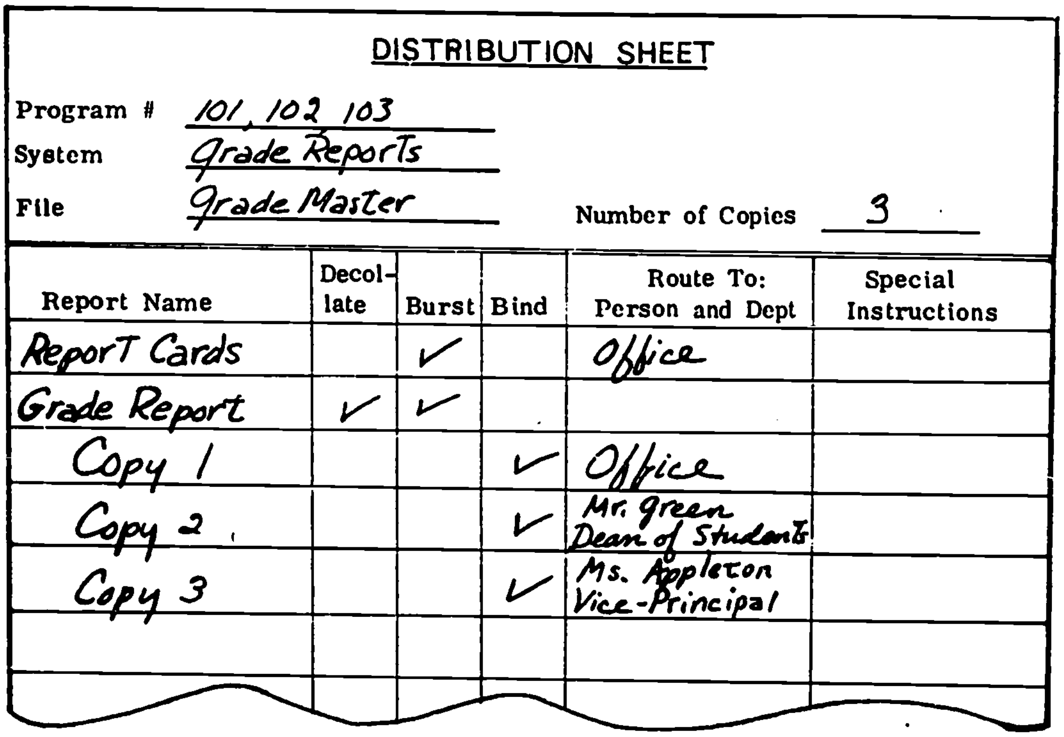
Machine Identification Card Reader (BXR 990A)

Description of Problem Card particles remain in card reader due to card jam. Causes card reader errors. Needs maintenance cleaning.

Fig. 71 Sample trouble log entry.

Along with the original set-up sheet for a job, the computer operator often gets a distribution sheet such as the one pictured in the following figure. This sheet is ordinarily filled out by the programmer to indicate how the final reports from the run are to be handled.

As you can see from the sample distribution sheet for the Grade Reports job, three routine things which can be done to output reports; they can be decollated (the carbons can be removed from report copies), burst (the pages can be separated at the perforated lines), and/or bound (pages can be fastened together with clips or other fasteners). In addition, each output may be routed to a particular person or department for use.



DISTRIBUTION SHEET

Program # 101, 102 103

System Grade RepoRTs

File Grade MasTer

Number of Copies 3

Report Name	Decol-late	Burst	Bind	Route To: Person and Dept	Special Instructions
ReporT Cards		✓		Office	
Grade Report	✓	✓			
Copy 1			✓	Office	
Copy 2			✓	Mr. Green Dean of StudenTs	
Copy 3			✓	Ms. Appleton Vice-Principal	

Fig. 75 Sample distribution sheet.

Is the sample distribution sheet perfectly clear to you? Notice the entries for the Grade Report. Only the complete Grade Report output by the job needs to be decollated and burst. After that, each of the three decollated and burst copies must be bound and routed separately.

In cases where the output needs to be routed promptly to different people, you can see how useful the distribution sheet is. It saves much time in preparing the output and forwarding it just as soon as it is produced. Whenever there is no distribution sheet, the set-up sheet always indicates to the operator where to route the output -- usually directly back to the programmer.

Check Your Understanding

Put a check (√) in the box(es) beside the correct answer(s) to the following questions.

1. Which of the following items is not written in on the typical job log?

   □ a. the time a run is started
   □ b. program number(s)
   □ c. names of cards, tapes and disks
   □ d. time a run is completed

2. Documenting a job includes all but one of the following. Which one is not a part of documentation?

   □ a. writing up the daily work schedule
   □ b. labelling tapes and disk packs
   □ c. completing the job log
   □ d. making up tabbed cards for card decks

3. In the final handling of output, when reports are "decollated", they are

   □ a. stapled together
   □ b. numbered and stacked
   □ c. separated at perforated lines
   □ d. none of the above

4. When final output is "burst,"

   □ a. it is thrown away
   □ b. it is corrected and rerun
   □ c. the carbons are removed
   □ d. the pages are separated at perforated lines

5. Which of the following items is not included on a typical distribution sheet?

   □ a. names of output documents
   □ b. instructions for what to decollate, burst or bind
   □ c. time the job is completed
   □ d. name of person or department to route output to

Check your answers using the key at the bottom of page 130.

GUIDELINES FOR TROUBLE-SHOOTING

In computer operations, there are three general types of trouble which can develop: hardware problems, software problems or job stream problems. The following section will provide you with some guidelines for handling these problems.

Common Hardware Problems

Some of the most common machine problems encountered by computer operators are described below and the general ways of handling them are indicated. To diagnose and handle any specific problems on a specific machine, however, the competent computer operator will always be certain he is thoroughly familiar with the machine and the operating procedures outlined in the operator's manual accompanying it.

Card Reader Problems

1. A card may not read properly due to improper placement in the hopper. Replacing it in the hopper in the correct position will take care of the problem. If improper reading is due to an off-punch (a punch that is out of line), the card must be repunched and placed back in the deck.

2. A card may become jammed in the hopper, read station or stacker. The jammed card must be removed and any card particles swept from the card reader. The card should be replaced by a newly punched card. If the operator is unable to remove all card particles when a card jams, a customer engineer must be called to service the card reader before it is is used again.

3. In the case of an invalid punch on a card (where the key punch has failed to punch a usuable code in a column), the card reader will stop. The card must be repunched correctly and run again.

Magnetic Tape Drive Problems

1. If the device has trouble reading a tape, the tape should be tried on the drive it was created on. If that drive works, the first drive is not functioning. The device needs to be serviced.

2. If the tape drive glass window sticks or only closes halfway, the C. E. should be called.

3. If the tape refuses to load, the door should be checked. The door must be completely closed before the device will run.

4. Errors in reading a tape may indicate oxide deposits or dust on the tape.

5. The operator should make sure all tapes are kept in covers when not in use.

6. If I/O errors persist, the tape may be worn out.

Printer Problems

1. To avoid paper jams, the operator should always check for correct insertion of paper and forms.

2. Tension control should also be watched, because incorrect tension may cause paper to jam. If a paper jam occurs or if the position of the form is incorrect, the operator should press the printer stop key and correct the condition.

3. Skipping of paper or incorrect forms spacing may be due to a worn carriage tape (if the machine uses carriage tapes), the wrong type of form being used, or an incorrect line-drive setting. All of these or their equivalent, should be checked.

4. Loss of print positions at the beginning of a report may be due to misplaced or worn ribbon, incorrect form, or incorrect position of form.

5. If the printer does not show a ready state, one of the following may be the reason:

    - An end-of-form condition may be indicated. In this case, more paper or forms must be placed in the printer and the printer restarted.

    - Forms may not be feeding properly. The position of paper in the printer should be checked and adjusted.

    - Carriage sensors or typebar may not be in ready position. The procedure for securing them to run should be checked.

Answers to questions on page 128:
1. (c) 2. (a) 3. (d) 4. (d) 5. (c)

- Often if the lower door to the printer is not secured, the printer will not function. If, while loading new paper, the lower door has been ajar, it should be secured.

Magnetic Disk Problem -- "Head Crash"

Sometimes a foreign particle or object gets caught between the magnetic surface and the read/write heads on a disk. This condition causes the read/write head to damage the disk surface. Damage to a disk surface may cause further damage if the disk pack is used on another drive. The problem should be taken care of immediately so further damage is not done.

The following indicators may tell the operator that a head crash has occurred:

- Sudden and persistent read errors.
- Unusual noise from a disk drive, which could go from a tinkling sound within the drive to a screeching sound.

The steps to be followed in case of a head crash are:

1. Stop the computer and remove the damaged disk pack.
2. Note the head crash in the trouble log.
3. Call a C. E. to repair the disk drive.
4. Proceed with computer runs only after the disk drive has been repaired.

Software Malfunctions

As you will recall, there are two kinds of software: the operating system (usually provided by the manufacturer) and the other programs (provided by user's programmers). The computer operator usually cannot solve software problems, but he can help by identifying any malfunctions and relaying this information to the manufacturer or programmer.

Whenever a problem arises in the operating system, it is usually reported directly to the manufacturer who provided this software. The computer operator will always indicate as clearly as possible the symptoms of the problem which he may have encountered.

You will remember that, before a program is ready for testing (and then use), it is first compiled. During the compile run, the program is loaded and output onto some medium as well as onto the printed paper from the printer. As the program is printed out, diagnostic messages (error messages) are printed by the printer. On most computers, these error messages may be of two kinds: warnings or "fatal" messages.

The difference between the two is just this: when, in compiling a program, the compiler notes a minor error, it prints out a warning message. When it discovers a crucial error (one that will stop the program from running if used in its present form), the compiler sends out a "fatal" message. When the printout from compiling is done, the programmer uses both kinds of error messages to help him correct his program.

The computer operator needs to be concerned with these error messages only when he is running a combined compile-then-test job for a programmer. Clearly, in such a case, if a "fatal" message is given during the compile run, he can not go on to test the program. The program would just stop at the crucial error. When running a combined job, then, a "fatal" message would cause the operator to stop the run after the compiling was done and to return the input and the printout to the programmer for correction.

Errors in Job Stream

Sometimes a programmer will make errors in filling out a job stream flowchart, a job set-up sheet, or even in his instructions to the data preparation people about preparing the program or data.

Such job stream errors are often noted by the computer operator before a run is made (as when the programmer has failed to identify what media to use for output from a run). During a run, errors in the input or output media specified by the programmer may be encountered when output "looks" odd, does not fit the forms, and so forth. Any incorrect information in these areas must be corrected by the programmer before the operator can complete the job successfully.

If a large job is being run which involves various different programs and outputs, it has often originated with the systems analyst. Any job stream difficulties encountered in these cases are, of course, referred back to the system's analyst.

HANDLING OF THE DATA MEDIA

Here are some helpful points to remember about handling data in the form of punched cards, magnetic tape or disks.

Handling Punched Cards

1. Punched card decks should be handled with care. Any crimping or marking of punch cards can produce card reader errors.
2. If a card deck appears to be worn on the edges, reproduce the entire deck to prevent unnecessary card reader errors.
3. Cards which are off-punched often cause card reader errors. A card gauge may be used to check an off-punch in a card. The card should be repunched and returned to its place in the card deck.

Handling Paper Tape

1. Punched paper tape is relatively fragile and must be handled carefully. Creased, torn or wrinkled tape will jam in the tape reader and/or cause tape reader errors.
2. Worn or damaged paper tape should be reproduced to insure a good copy of the tape for use.
3. Paper tapes should be stored in rolls in plastic or paper boxes to minimize the danger of tearing, creasing or perforation.

Handling Magnetic Tape

1. Magnetic tapes should be kept on tape reels and stored in plastic tape containers. Since the magnetic surface of tapes is highly fragile, great care should be taken in the handling of them.
2. Magnetic tape reels should be kept free of dust. A lint-free cloth may be used to remove any accumulated dust.

3. To keep these tapes in good condition, it is necessary to be sure the tape drive unit is kept clean. After about eight hours of use, the drive should be dusted with a lint-free cloth. The read/write heads and the drive mechanism should be cleaned according to the instructions in the operator's guide for the particular device.

Disk Pack Handling

1. To protect the delicate surfaces of the disks, the plastic top and bottom covers should be kept on the pack whenever the pack is not in use on the disk drive.

2. Disk packs should be stored flat on shelves -- never stacked one on top the other or stored on edge.

3. Disk packs should be stored away from sunlight or fire hazard. An enclosed metal cabinet provides good protection. The storage area should be kept at a temperature between 60 and 90 F, with humidity from 10% to 80%.

Labelling Magnetic Tapes and Disk Packs

Small, adhesive labels may be placed on the cover of the tape reel or disk pack. The label should clearly identify the data stored on the tape or disk pack.

1. Use labels that are applied or removed easily.

2. Write on the label with something which does not leave particles of dust, like a felt pen.

3. Write the label before you place it on the disk pack or tape reel.

4. On disk packs, labels should be placed only on the cover. Never place labels on disk surfaces. On tape reels a label should be placed on both the reel itself and on the reel container.

5. Use a new label for every change of identification. Erasing labels leaves particles which may damage disk or tape surfaces or read/write heads.

PART V

AN EXERCISE IN COMPUTER OPERATION

PART V AN EXERCISE IN COMPUTER OPERATION

INTRODUCTION

Now that you are familiar with the various kinds of tasks a computer operator performs on a typical day, here's a chance for you to put your understanding to work. In the exercise which follows, you will have to act as the computer operator on a typical day, June 16, 1972, filling out all the appropriate schedules and logs and running one complete job from beginning to end on the computer.*

Follow all the instructions and suggestions carefully as you go along, and be sure to check your work with the answers provided in the Appendix whenever you are instructed to.

The job you will be required to run on the computer is the Grading System job (with program NSORT and the Quarter Test Grade Sort Data) you saved from your work in the manual, THE ELEMENTS OF DATA PREPARATION. So, before proceeding with the exercise, you should retrieve the Grading System input you saved on paper tape, mark-sense cards and/or punched cards. Determine which of the input media you can use with your computer system, then keep the input which is on that medium and set the other input aside. For example, if your computer system has a punched card reader as its only input device, you would select to use the Grading System input you prepared on punched cards and to set your paper tapes and mark-sense cards aside.

PREPARING THE DAILY WORK SCHEDULE

In this exercise, you will be acting as the computer operator on the typical day of June 16, 1972. Assume that your computer and the input/output devices have all been started, and that you are ready to make up your daily work schedule.

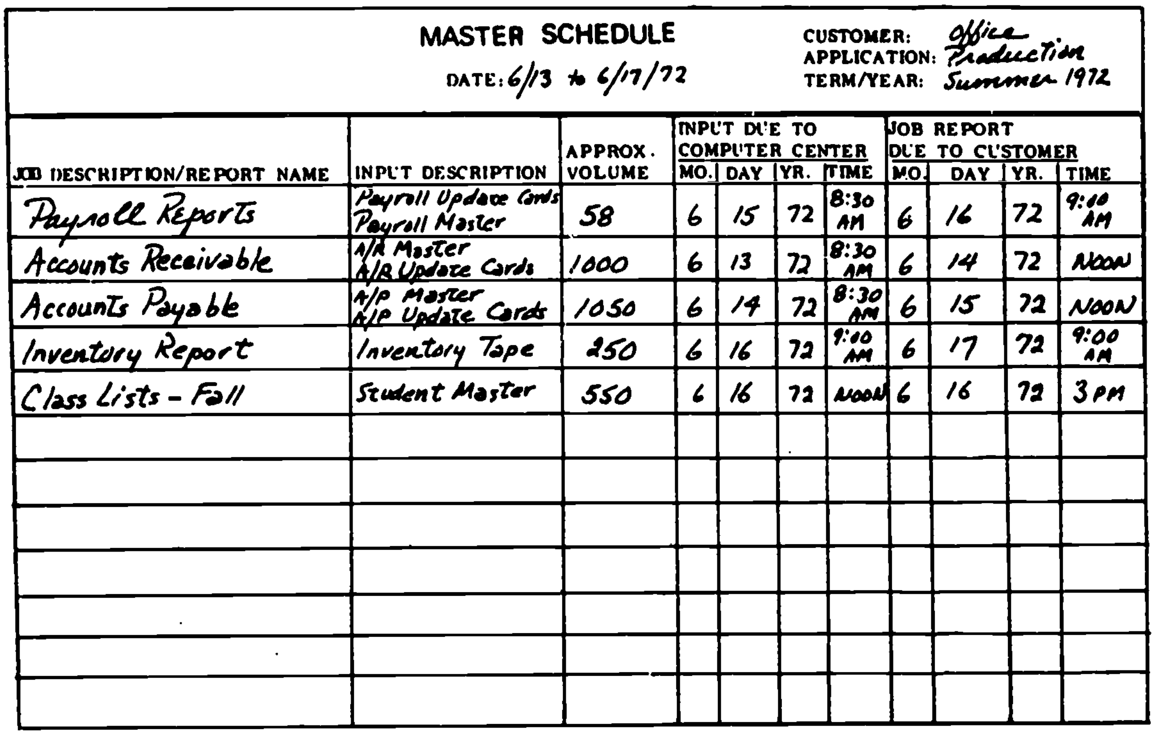
*If you do not have a computer available to you for running this job, you can still go through all of this exercise, skipping just the last section on "Running the Grading System Job."

The Master Schedule of production runs for the week of June 13-17, 1972, is shown in Fig. 76 below. In addition to the jobs scheduled there for June 16, assume that the following runs have been requested for the same day:

Test -- Faculty Reporting (three separate runs)
Computer Utilization
Compile -- Senior Absences (two runs needed, maybe three)
Grading System

Fig. 77 on page 139 shows the top portion of the job set-up sheets for each of these jobs. A copy of the instruction card for the compile run is also shown.

Using the information given on the Master Schedule and the instruction sheets, make up your daily work schedule for June 16, 1972. Fill in your complete schedule on the blank schedule form on page 140.



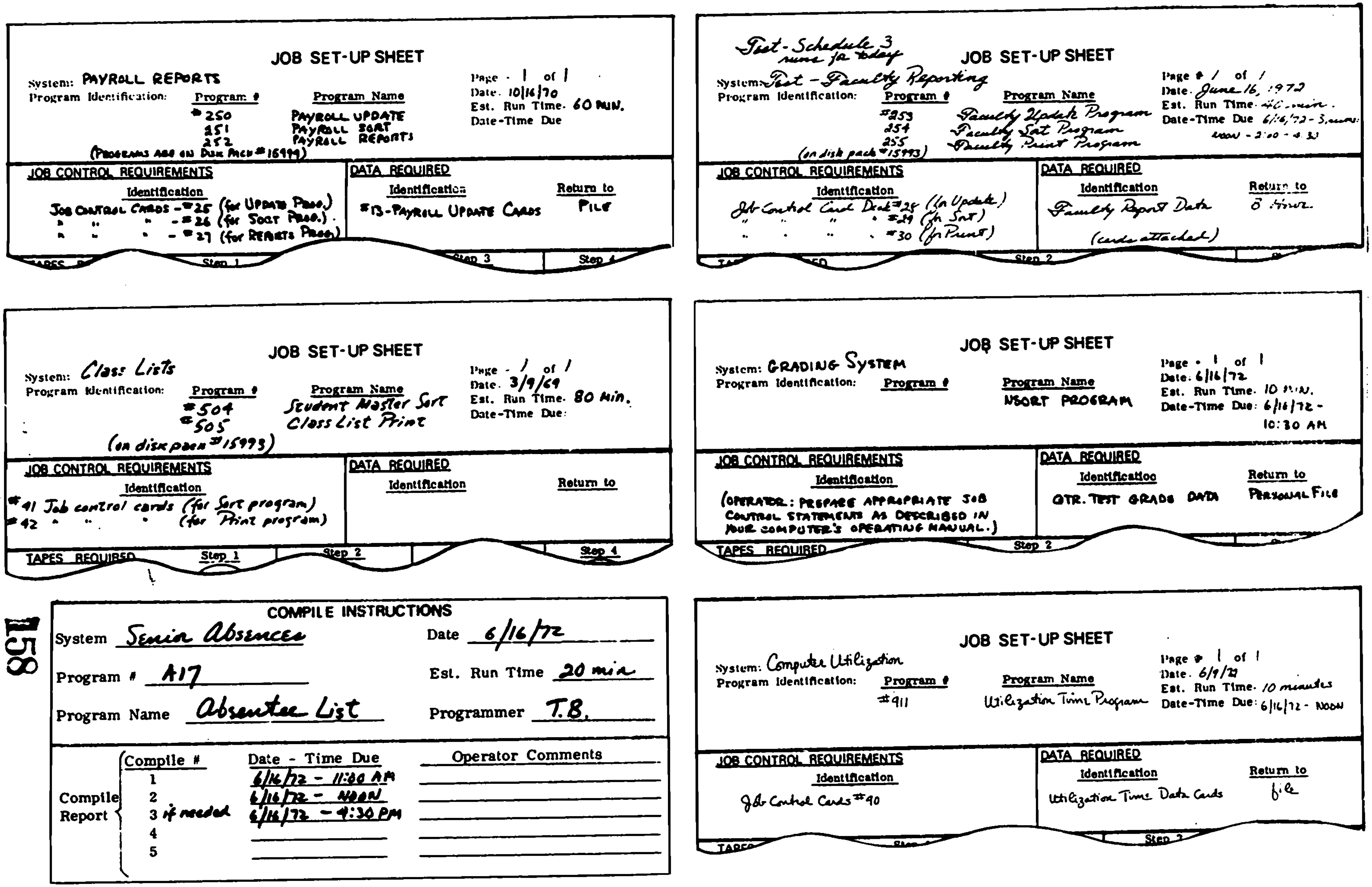
MASTER SCHEDULE

DATE: 6/13 to 6/17/72

CUSTOMER: Office
APPLICATION: Production
TERM/YEAR: Summer 1972

			INPUT DUE TO COMPUTER CENTER				JOB REPORT DUE TO CUSTOMER			
JOB DESCRIPTION/REPORT NAME	INPUT DESCRIPTION	APPROX. VOLUME	MO.	DAY	YR.	TIME	MO.	DAY	YR.	TIME
Payroll Reports	Payroll Update Cards Payroll Master	58	6	15	72	8:30 AM	6	16	72	9:10 AM
Accounts Receivable	A/R Master A/R Update Cards	1000	6	13	72	8:30 AM	6	14	72	NOON
Accounts Payable	A/P Master A/P Update Cards	1050	6	14	72	8:30 AM	6	15	72	NOON
Inventory Report	Inventory Tape	250	6	16	72	9:10 AM	6	17	72	9:00 AM
Class Lists - Fall	Student Master	550	6	16	72	NOON	6	16	72	3 PM

Fig. 76 Master Schedule for week of June 13, 1972.



JOB SET-UP SHEET

System: PAYROLL REPORTS
Program Identification:

Program #	Program Name
#250	PAYROLL UPDATE
251	PAYROLL SORT
252	PAYROLL REPORTS

(PROGRAMS ARE ON DISK PACK #16999)

Page - 1 of 1
Date. 10/16/70
Est. Run Time. 60 MIN.
Date-Time Due

JOB CONTROL REQUIREMENTS

Identification
JOB CONTROL CARDS - #25 (for UPDATE PROG.)
" " " - #26 (for SORT PROG.)
" " " - #27 (for REPORTS PROG.)

DATA REQUIRED

Identification	Return to
#13-PAYROLL UPDATE CARDS	FILE

Step 1 Step 3 Step 4

Test-Schedule 3
runs for today

JOB SET-UP SHEET

System: Test - Faculty Reporting
Program Identification:

Program #	Program Name
#253	Faculty Update Program
254	Faculty Sort Program
255	Faculty Print Program

(on disk pack #15993)

Page # 1 of 1
Date. June 16, 1972
Est. Run Time. 40 min.
Date-Time Due 6/16/72 - 3 runs:
noon - 2:00 - 4:30

JOB CONTROL REQUIREMENTS

Identification
Job Control Card Deck #28 (for Update)
" " " " #29 (for Sort)
" " " " #30 (for Print)

DATA REQUIRED

Identification	Return to
Faculty Report Data (cards attached)	8 Hour

Step 2

JOB SET-UP SHEET

System: Class Lists
Program Identification:

Program #	Program Name
#504	Student Master Sort
#505	Class List Print

(on disk pack #15993)

Page - 1 of 1
Date. 3/9/69
Est. Run Time. 80 min.
Date-Time Due:

JOB CONTROL REQUIREMENTS

Identification
#41 Job control cards (for Sort program)
#42 " " " (for Print program)

DATA REQUIRED

Identification Return to

TAPES REQUIRED Step 1 Step 2 Step 4

JOB SET-UP SHEET

System: GRADING SYSTEM
Program Identification:

Program #	Program Name
	NSORT PROGRAM

Page - 1 of 1
Date. 6/16/72
Est. Run Time. 10 MIN.
Date-Time Due: 6/16/72 -
10:30 AM

JOB CONTROL REQUIREMENTS

Identification
(OPERATOR: PREPARE APPROPRIATE JOB CONTROL STATEMENTS AS DESCRIBED IN YOUR COMPUTER'S OPERATING MANUAL.)

DATA REQUIRED

Identification	Return to
QTR. TEST GRADE DATA	PERSONAL FILE

TAPES REQUIRED Step 2

COMPILE INSTRUCTIONS

System Senior Absences Date 6/16/72
Program # A17 Est. Run Time 20 min
Program Name Absentee List Programmer T.B.

Compile Report

Compile #	Date - Time Due	Operator Comments
1	6/16/72 - 11:00 AM	
2	6/16/72 - NOON	
3 if needed	6/16/72 - 4:30 PM	
4		
5		

JOB SET-UP SHEET

System: Computer Utilization
Program Identification:

Program #	Program Name
#911	Utilization Time Program

Page # 1 of 1
Date. 6/9/72
Est. Run Time. 10 minutes
Date-Time Due: 6/16/72 - NOON

JOB CONTROL REQUIREMENTS

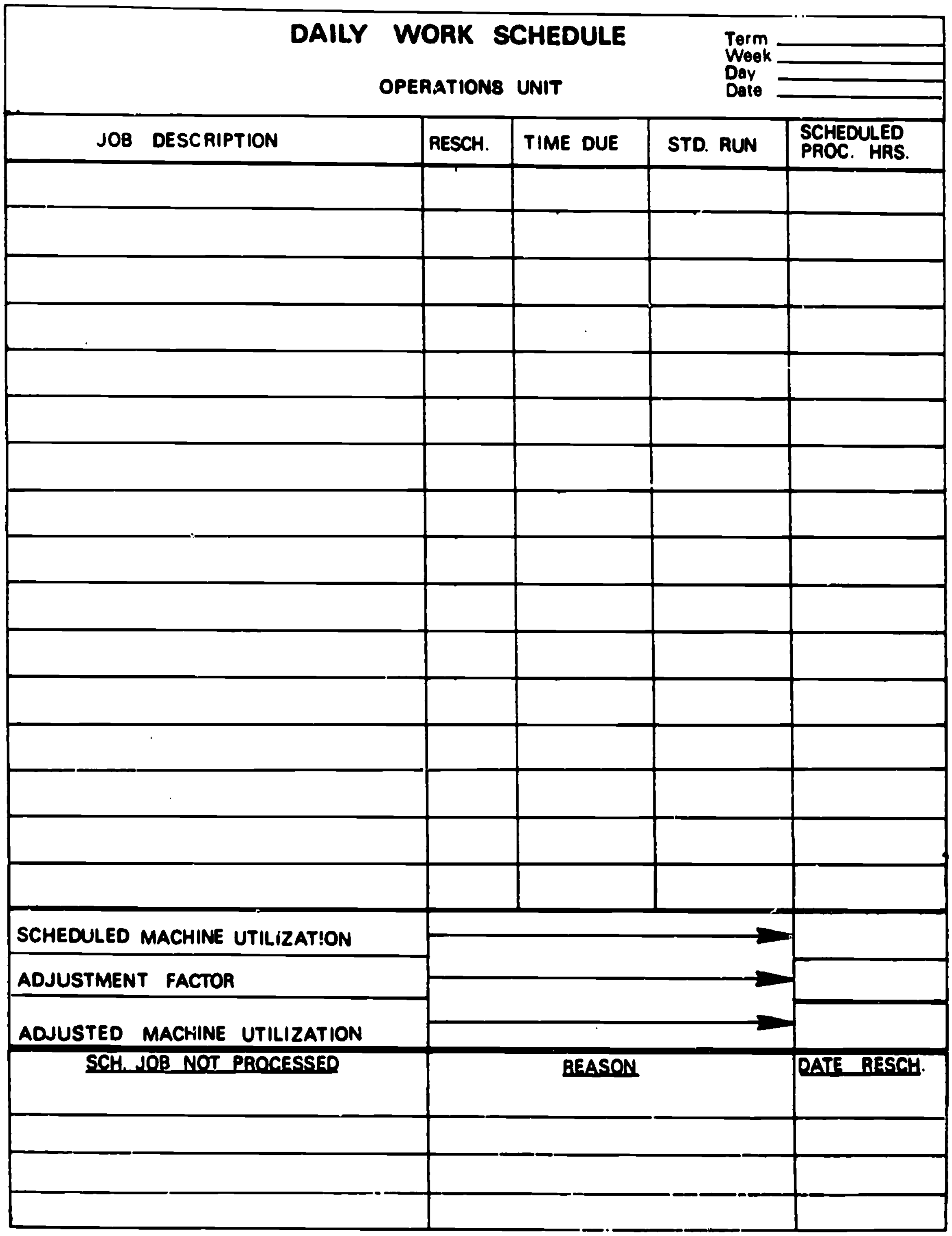
Identification
Job Control Cards #90

DATA REQUIRED

Identification	Return to
Utilization Time Data Cards	file

Step 2

Fig. 77 Top portion of job set-up sheets and compile instruction card.



DAILY WORK SCHEDULE

OPERATIONS UNIT

Term ______
Week ______
Day ______
Date ______

JOB DESCRIPTION	RESCH.	TIME DUE	STD. RUN	SCHEDULED PROC. HRS.

SCHEDULED MACHINE UTILIZATION	→	
ADJUSTMENT FACTOR	→	
ADJUSTED MACHINE UTILIZATION	→	

SCH. JOB NOT PROCESSED	REASON	DATE RESCH.

Fig. 78 Daily Work Schedule form for June 16, 1972.

When your daily schedule is complete, compare it with the version shown on page 161 of the Appendix. Does your schedule include all the runs and the appropriate due time, standard run time and scheduled processing hours for each? Does your "Scheduled Machine Utilization" figure agree with the one shown on the Appendix version? Also, check to see if you have scheduled all runs so they will be completed by their due-times. Using the sample schedule in the Appendix as a model, correct or improve your schedule in any way that you think it needs it.

RUNNING THE FIRST JOB -- ON PAPER

Now, assume that all the housekeeping chores have been done on your computer and you are ready to set up your computer equipment for the first job of the day, which should be the Payroll Reports job.

Setting Up the Job

The Payroll Reports job set-up sheet and job-stream flowchart are shown on the next two pages. Study them until you have a good understanding of what the job will involve. Then, rather than setting up the actual computer equipment to run the job, answer the preliminary questions on page 144 and then fill in the diagram on page 145 to show, on paper, how the equipment for this job should be set up.

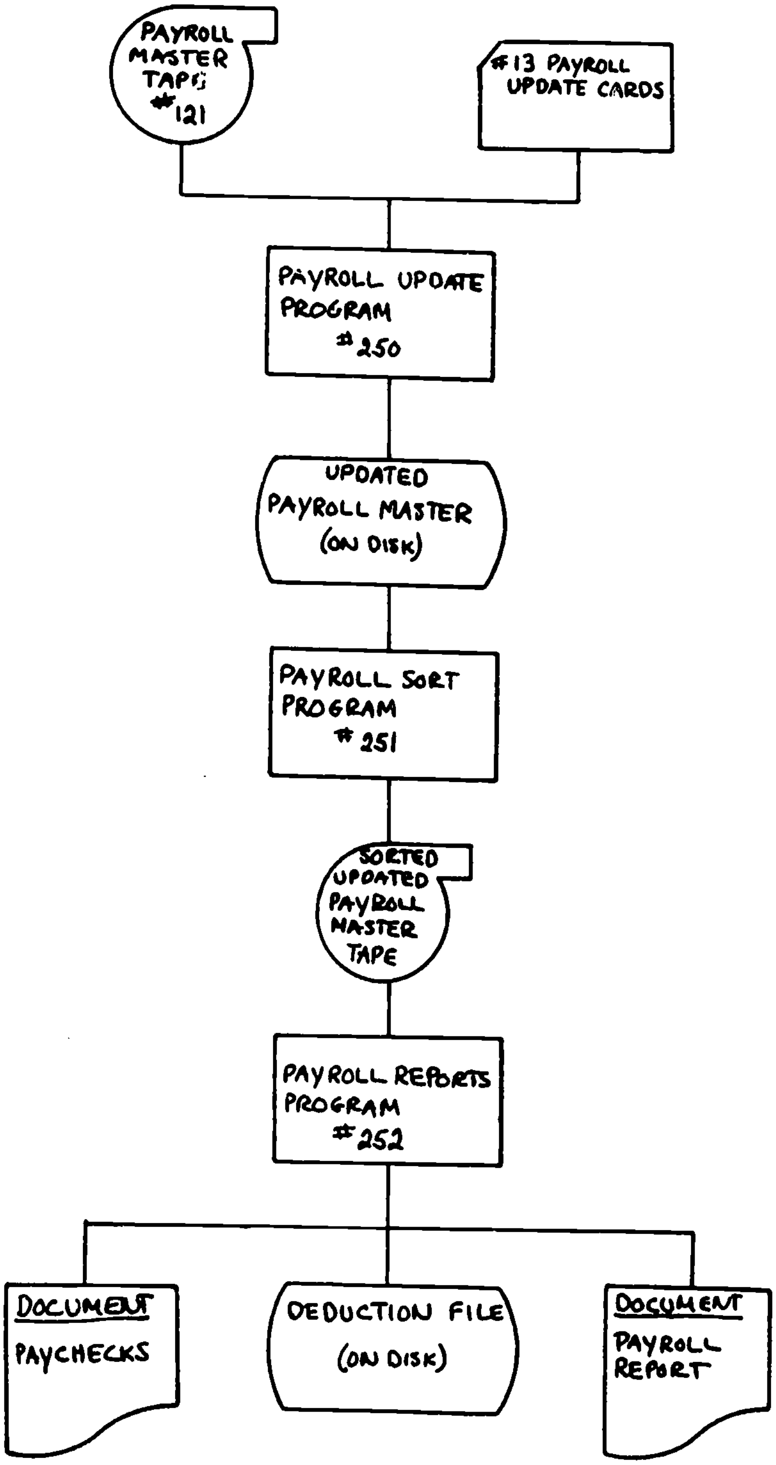
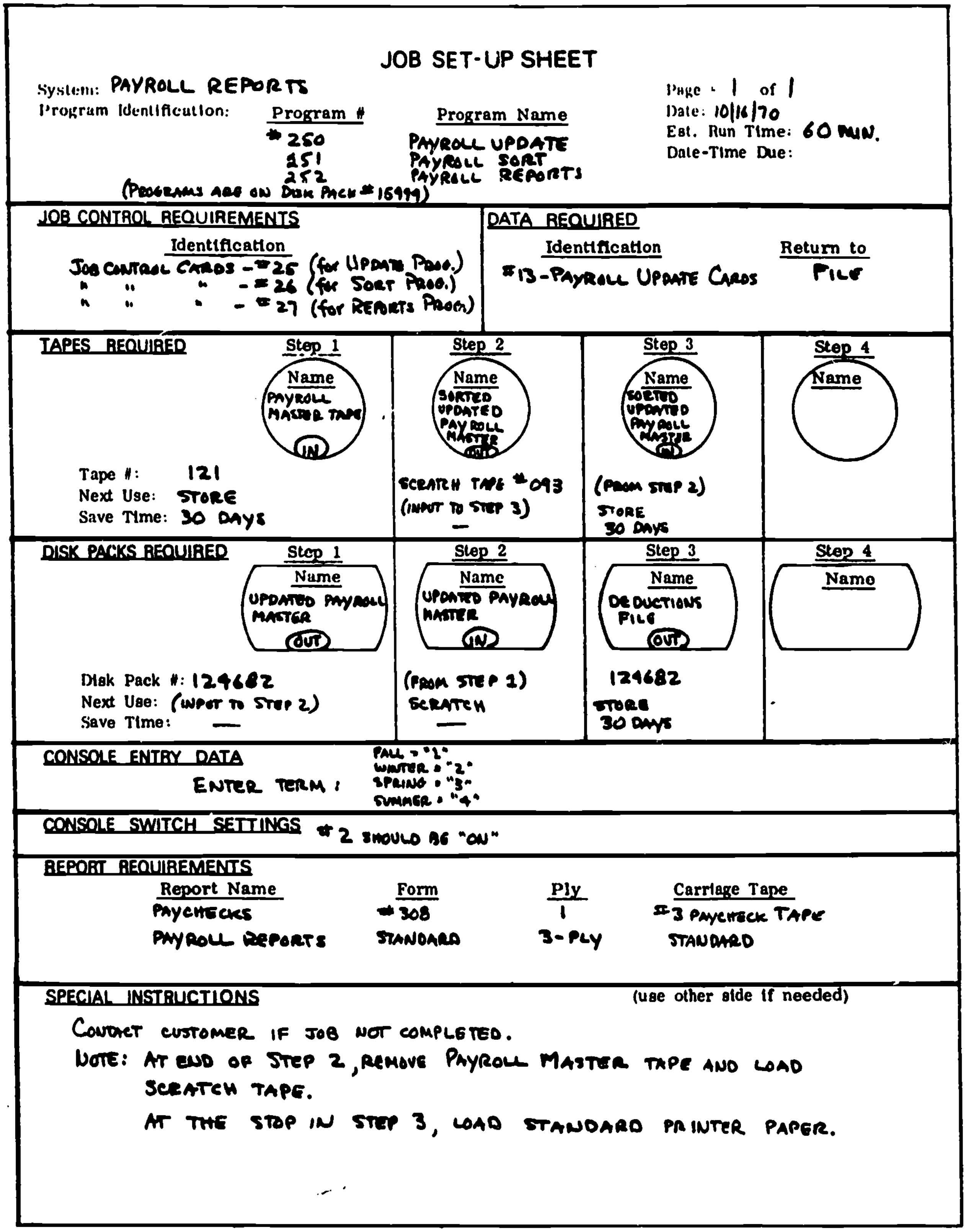

PAYROLL MASTER TAPE #121
#13 PAYROLL UPDATE CARDS
PAYROLL UPDATE PROGRAM #250
UPDATED PAYROLL MASTER (ON DISK)
PAYROLL SORT PROGRAM #251
SORTED UPDATED PAYROLL MASTER TAPE
PAYROLL REPORTS PROGRAM #252
DOCUMENT PAYCHECKS
DEDUCTION FILE (ON DISK)
DOCUMENT PAYROLL REPORT

Fig. 79 Job-stream flowchart for Payroll Reports job.



JOB SET-UP SHEET

System: PAYROLL REPORTS

Page: 1 of 1
Date: 10/16/70
Est. Run Time: 60 MIN.
Date-Time Due:

Program Identification:

Program #	Program Name
#250	PAYROLL UPDATE
251	PAYROLL SORT
252	PAYRLL REPORTS

(PROGRAMS ARE ON DISK PACK #15999)

JOB CONTROL REQUIREMENTS

Identification
JOB CONTROL CARDS - #25 (for UPDATE PROG.)
" " " - #26 (for SORT PROG.)
" " " - #27 (for REPORTS PROG.)

DATA REQUIRED

Identification	Return to
#13-PAYROLL UPDATE CARDS	FILE

TAPES REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	PAYROLL MASTER TAPE IN	SORTED UPDATED PAYROLL MASTER OUT	SORTED UPDATED PAYROLL MASTER IN	
Tape #:	121	SCRATCH TAPE #093	(FROM STEP 2)	
Next Use:	STORE	(INPUT TO STEP 3)	STORE	
Save Time:	30 DAYS	—	30 DAYS	

DISK PACKS REQUIRED

	Step 1	Step 2	Step 3	Step 4
Name	UPDATED PAYROLL MASTER OUT	UPDATED PAYROLL MASTER IN	DEDUCTIONS FILE OUT	
Disk Pack #:	124682	(FROM STEP 1)	124682	
Next Use:	(INPUT TO STEP 2)	SCRATCH	STORE	
Save Time:	—	—	30 DAYS	

CONSOLE ENTRY DATA

ENTER TERM: FALL = "1"
WINTER = "2"
SPRING = "3"
SUMMER = "4"

CONSOLE SWITCH SETTINGS #2 SHOULD BE "ON"

REPORT REQUIREMENTS

Report Name	Form	Ply	Carriage Tape
PAYCHECKS	#308	1	#3 PAYCHECK TAPE
PAYROLL REPORTS	STANDARD	3-PLY	STANDARD

SPECIAL INSTRUCTIONS (use other side if needed)

CONTACT CUSTOMER IF JOB NOT COMPLETED.

NOTE: AT END OF STEP 2, REMOVE PAYROLL MASTER TAPE AND LOAD SCRATCH TAPE.

AT THE STOP IN STEP 3, LOAD STANDARD PRINTER PAPER.

Fig. 80 Job set-up sheet for Payroll Reports job.

Preliminary Questions for Payroll Reports Job

1. What job control cards are needed? (Give identification numbers.)

    ____

2. What punched card data is required to run the job? (Give names and numbers.)

    ____

    ____

3. What magnetic tapes must be found in the tape library? (Give names and numbers.)

    ____

    ____

4. What disk packs are to be used? (Give names and numbers.)

    ____

    ____

5. What data that is output on tape or disk pack during this run will be saved at the end of the job? (Give identification number.)

    ____

    ____

6. What output forms are needed for the printer in this run? (Give names and form descriptions.)

    ____

    ____

When you have filled in the above answers, check to be sure your lists are complete by comparing your answers with those given on page 162 of the Appendix.

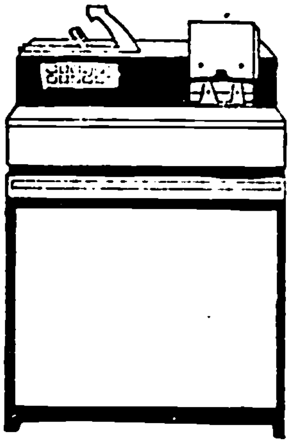
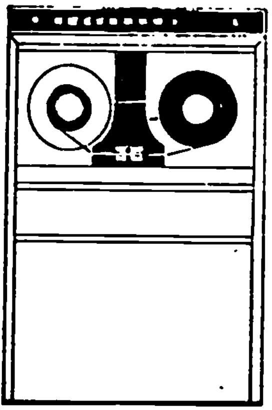
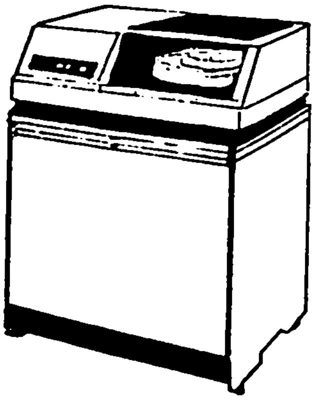
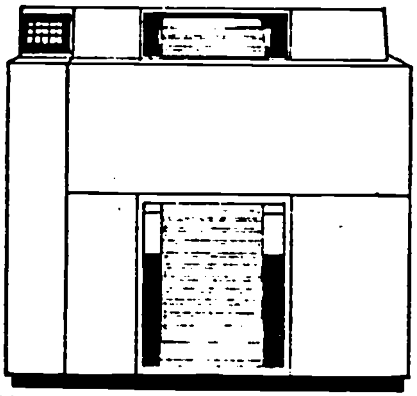
INPUT/OUTPUT DEVICES	INPUT TO BE LOADED ON DEVICE (Specify)	INPUT TO BE KEPT READY (if any)
		
		
		
		

Fig. 81 Setting up equipment for Payroll Reports job.

When you have finished filling in the set-up diagram, compare yours with the answer shown on page 163 of the Appendix. Correct your set-up up plan, if it needs it.

Running the Job

Now with the equipment set up, assume you are ready to start your run.

According to the set-up sheet for this job on page 143, what sense switches must be on at the start of the job?

Sense Switch Setting:____________

(The correct switch setting is switch number two in the "on" position.)

On the basis of the set-up sheet, decide which of the following interventions you are going to have to make during this run. Put a check in the box beside the applicable intervention(s).

□ 1. Splitting the paychecks apart.

□ 2. Loading standard printer paper during Step 3.

□ 3. Changing sense switches at Step 1.

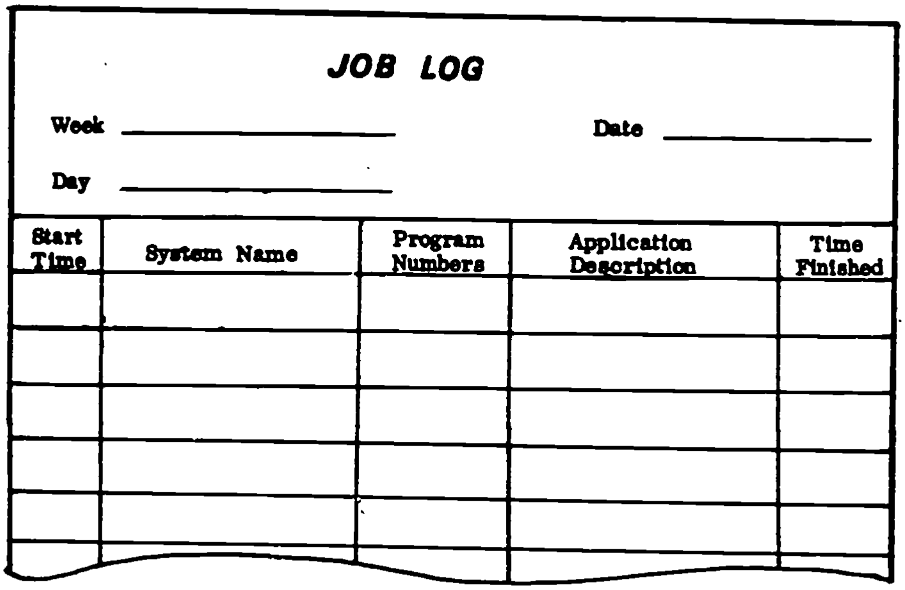
□ 4. Entering the current term as the computer requests it.

□ 5. Loading a new disk pack after Step 2.

□ 6. Loading a scratch tape after Step 2.

(Check to see if your interventions are correct. The ones you should have marked are numbers two, four and six. Get them right? If not, look again at the job set-up sheet and see why 2, 4, and 6 are the only interventions required.)

Now, assume it is 8:45 a. m. and that you have started the job. Begin your day's job log by filling in the information for this first job.



JOB LOG

Week ____________ Date ____________

Day ____________

Start Time	System Name	Program Numbers	Application Description	Time Finished

Fig. 82 Job log for June 16, 1972.

At 9:05 a.m., suppose the disk drive begins to make a loud noise and you have a "head crash" to deal with. Assume that you have a "back-up" copy of all disk packs with production programs on them for use in such emergencies.

Listed below are steps to trouble shooting you would take to correct the head crash problem. Put the steps in the order in which you would implement them.

_____ Note the problem on the trouble log.

_____ Place a duplicate disk on the disk drive and rerun the Payroll Reports job.

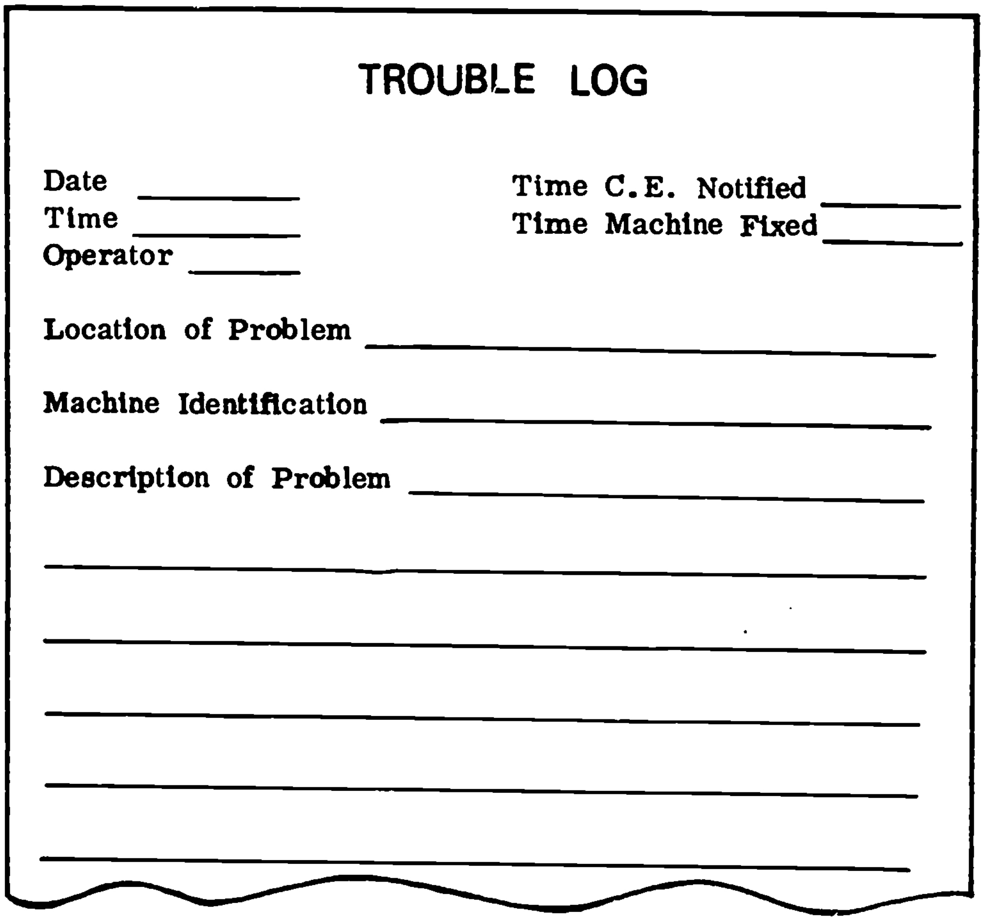
_____ Stop the running of the job and remove the damaged disk pack from the disk drive.

_____ Call a C. E. and have the head crash problem corrected.

(Check your answers using the key at the bottom of page 149.)

Since the job log for the day needs to reflect the actual machine time used by each job, you will have to show on your log that the first run of Payroll Reports job stopped at 9:05 a. m. Go back, then and enter this time in the "Time Finished" column of your day's job log on page 147. When the head crash problem is fixed, the entire Payroll Reports job will have to be run from the beginning again, so you will be entering the job in your log, again, with a new start and finish time.

The disk drive problem will also have to be written up in the trouble log for your computer. Using the sample trouble log sheet below, write up your report on the problem.



TROUBLE LOG

Date ________ Time C.E. Notified ________
Time ________ Time Machine Fixed ________
Operator ______

Location of Problem ____________________

Machine Identification ____________________

Description of Problem ____________________

____

____

____

____

____

Fig. 83 Trouble log report on
June 16, 1972 head crash.

In the case of this head crash, assume that the customer engineer came promptly and was able to put the disk drive back into working order by 10:10 a. m. At this point, you can start your Payroll Reports job over. Enter this starting time for your second run of the job on your job log on page 147.

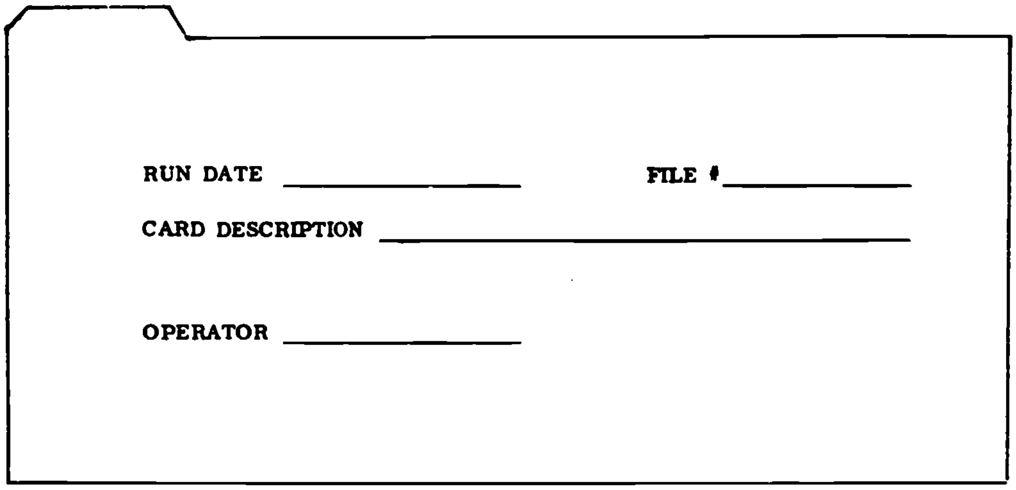
Assume that all goes well on this second run, and that the job is finished by 11:02 a. m.

(Check your log against the completed version shown on page 164 of the Appendix. Correct yours, if it is inaccurate or incomplete.)

Documenting the Job

Your first step in documenting this job would, of course, be to complete your entry of the job in your job log by filling in the "Time Finished" column. Go back to your log on page 147 and fill it in. Then, check the accuracy of your log by comparing it with the one shown on page 165 of the Appendix. Correct yours if it needs it.

Now, document the #13-Payroll Update cards by filling in the necessary information on the tabbed card form provided in Fig. 84 below.



RUN DATE ________________ FILE # ____________

CARD DESCRIPTION ____________________________________

OPERATOR ________________

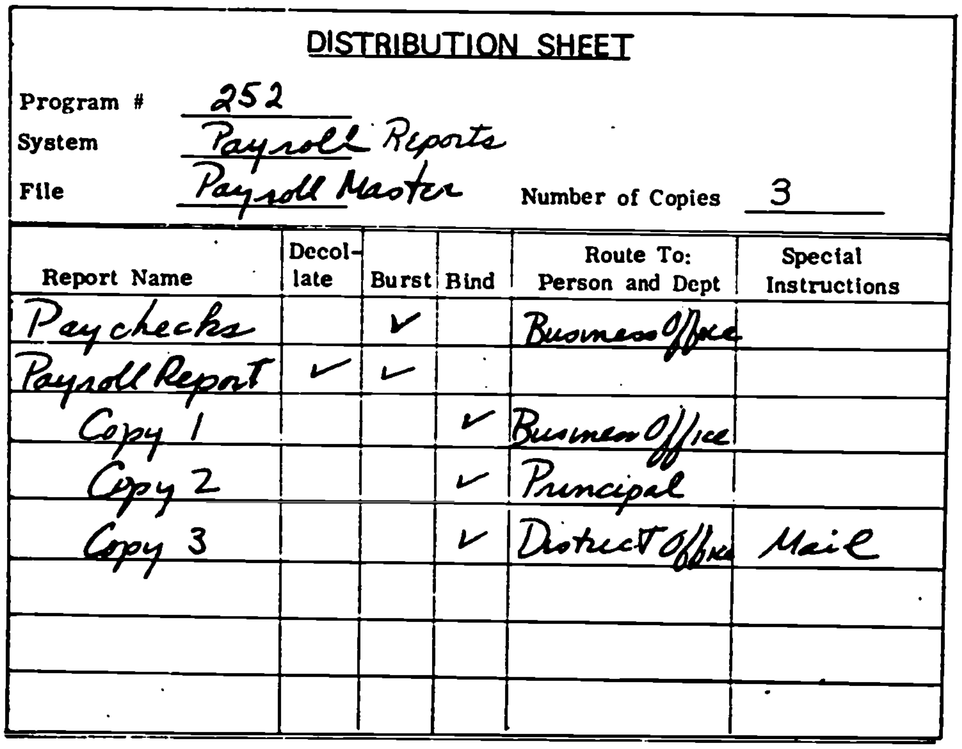
Fig. 84 Tabbed card for Payroll Update cards.

An example of how this card should be filled out is on page 166 of the Appendix. Check your card against the one given there and correct yours if it needs it.

Answers for troubleshooting question, page 147:
2, 4, 1, 3

Handling Final Output

The distribution sheet illustrated below shows what should be done with the final output from the Payroll Reports job.



DISTRIBUTION SHEET

Program # 252
System Payroll Reports
File Payroll Master Number of Copies 3

Report Name	Decol-late	Burst	Bind	Route To: Person and Dept	Special Instructions
Paychecks		✓		Business Office	
Payroll Report	✓	✓			
Copy 1			✓	Business Office	
Copy 2			✓	Principal	
Copy 3			✓	District Office	Mail

Fig. 85 Distribution sheets for payroll reports run.

On the basis of these distribution directions, what must be done to the payroll report before the three copies are bound separately and sent out? (Explain clearly and completely.)

Did you know how to handle the three-part report? (Check your explanation using the answer given on page 167 of the Appendix. Correct your description so that it is accurate and complete.)

You have now gone through some of the main steps involved in running a job on a computer. If there were any places where you felt really lost about what to do, go back and reread the part of this manual that deals with the topic. Before going on to the next section, you should feel fairly sure of the elements involved in computer operation.

RUNNING THE GRADING SYSTEM JOB -- ON THE COMPUTER

Introduction

You are now ready to run the Grading System job on your computer system. Assume it is the second job to be run on your typical day of June 16, 1972.

To get ready for your work, schedule a time for your run with your teacher or computer operator, and review your notes in Part II on running your system's equipment. Be certain you understand thoroughly how to work with the input device appropriate for your Grading System input (punched card reader, mark-sense card reader, or paper tape punch) as well as your system's printer.

You should also carefully review the procedures for running your computer's central processing unit, including how to prepare job control statements for your FORTRAN NSORT program. Study the operator's guide to your computer if you have any questions before you begin your run.

Setting Up the Job

To begin the Grading System job, be sure you have your program and data input at hand and that you understand what will be involved in the job. A job-stream flowchart and a job set-up sheet for the Grading System job are shown on page 153 for your reference.

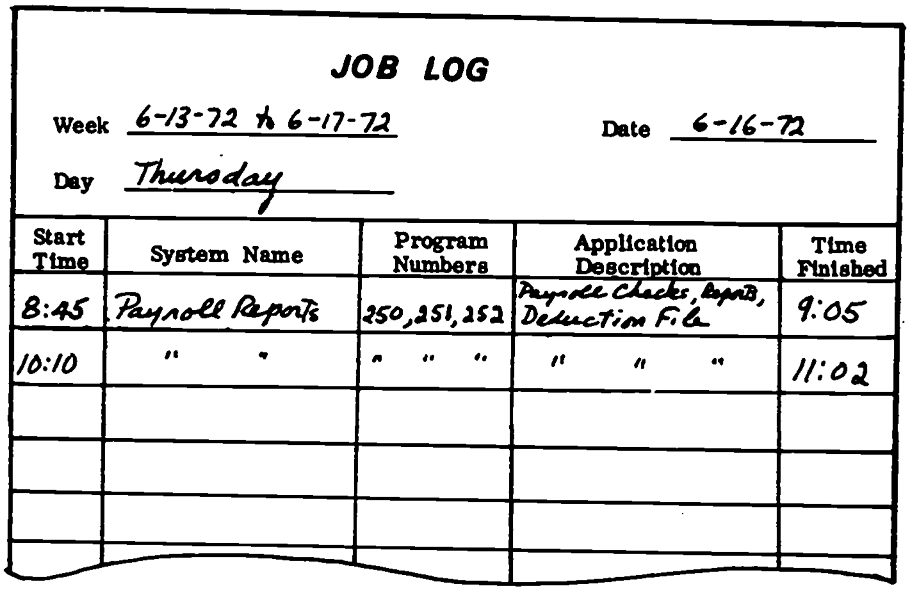
1. Prepare the required job control statements for the run. Consulting the operator's guide to your computer, your computer operator or your teacher if you need help.

2. Set up the input and output devices for the job.

Running the Job

Now, you are ready to make your run.

1. Enter this job on the job log below. (Assume this is the second job you are running on your typical day of June 16, 1972.)



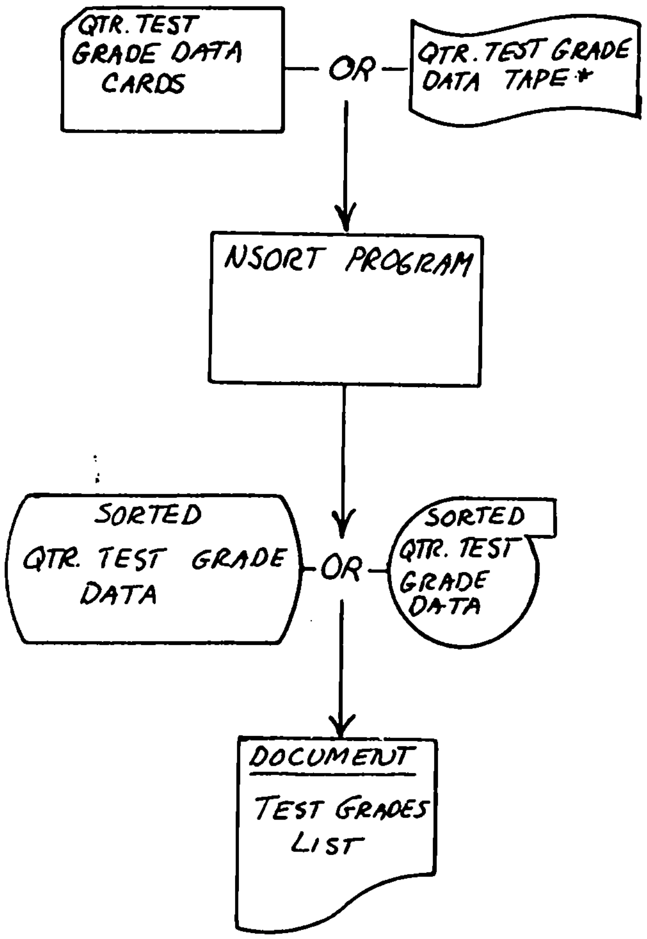
JOB LOG

Week 6-13-72 to 6-17-72 Date 6-16-72

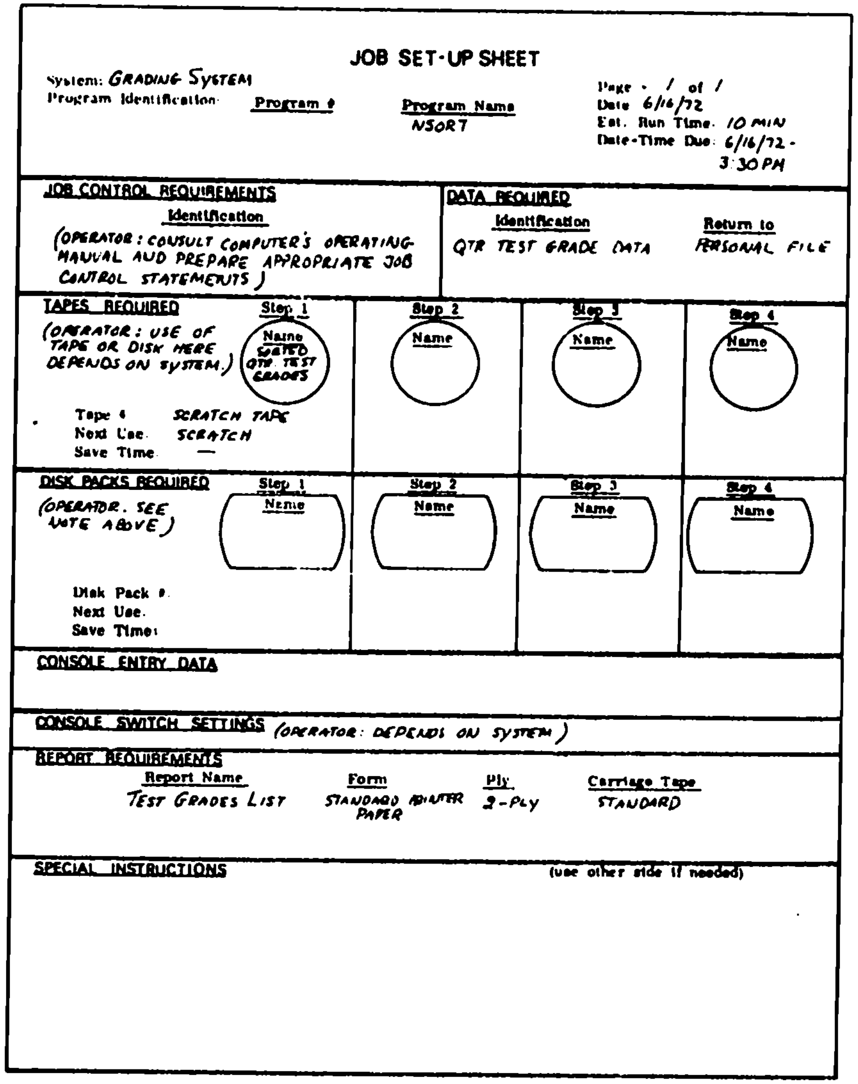
Day Thursday

Start Time	System Name	Program Numbers	Application Description	Time Finished
8:45	Payroll Reports	250, 251, 252	Payroll Checks, Reports, Deduction File	9:05
10:10	" "	" " "	" " "	11:02

Fig. 86 Job log for June 16, 1972.


QTR. TEST GRADE DATA CARDS
OR
QTR. TEST GRADE DATA TAPE *
NSORT PROGRAM
SORTED QTR. TEST GRADE DATA
OR
SORTED QTR. TEST GRADE DATA
DOCUMENT
TEST GRADES LIST

NOTE: The symbol ⌇ is used to represent paper tape.



JOB SET-UP SHEET

System: GRADING SYSTEM
Program Identification: Program # Program Name NSORT

Page - 1 of 1
Date 6/16/72
Est. Run Time. 10 MIN
Date-Time Due: 6/16/72 - 3:30 PM

JOB CONTROL REQUIREMENTS

Identification

(OPERATOR: CONSULT COMPUTER'S OPERATING MANUAL AND PREPARE APPROPRIATE JOB CONTROL STATEMENTS)

DATA REQUIRED

Identification	Return to
QTR TEST GRADE DATA	PERSONAL FILE

TAPES REQUIRED

(OPERATOR: USE OF TAPE OR DISK HERE DEPENDS ON SYSTEM.)

Step 1	Step 2	Step 3	Step 4
Name SORTED QTR. TEST GRADES	Name	Name	Name

Tape # SCRATCH TAPE
Next Use. SCRATCH
Save Time —

DISK PACKS REQUIRED

(OPERATOR. SEE NOTE ABOVE)

Step 1	Step 2	Step 3	Step 4
Name	Name	Name	Name

Disk Pack #.
Next Use.
Save Time:

CONSOLE ENTRY DATA

CONSOLE SWITCH SETTINGS (OPERATOR: DEPENDS ON SYSTEM)

REPORT REQUIREMENTS

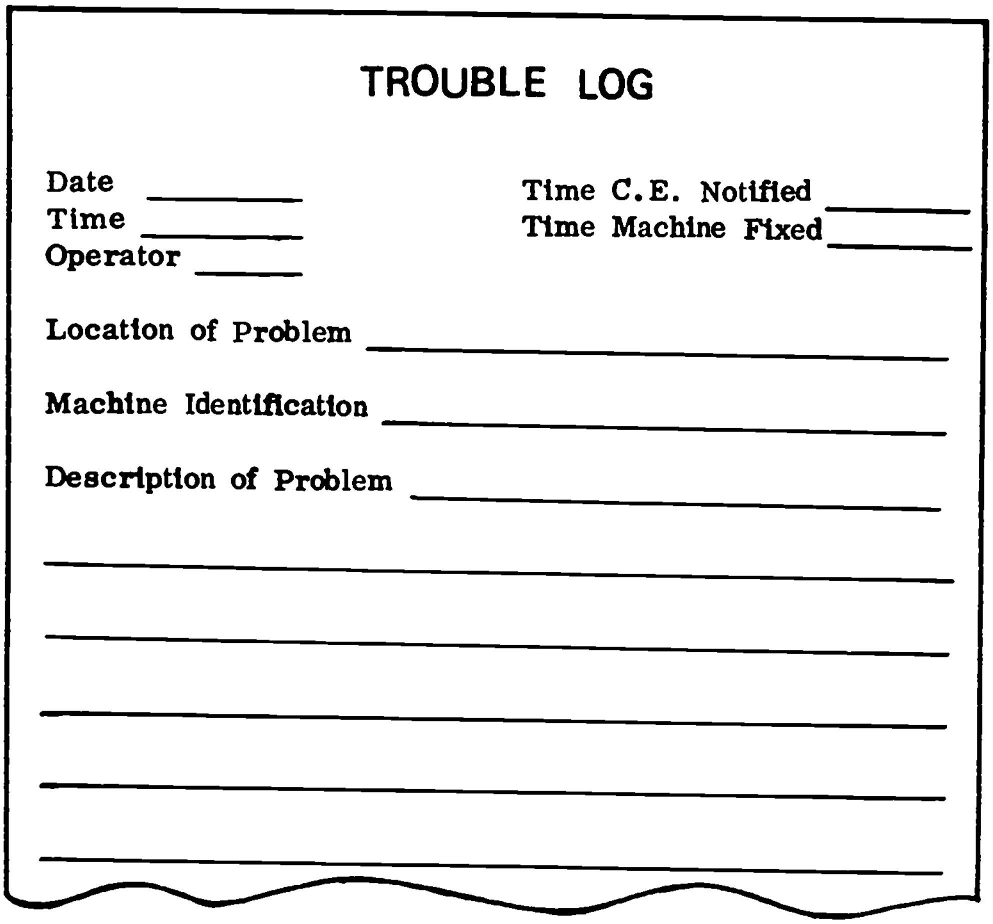
Report Name	Form	Ply	Carriage Tape
TEST GRADES LIST	STANDARD PRINTER PAPER	2-PLY	STANDARD

SPECIAL INSTRUCTIONS (use other side if needed)

Fig. 87 Job-stream flowchart and set-up sheet for Grading System job.

2. Start your run.

3. If you have any equipment trouble, follow the correct procedures for handling the problem. Enter the trouble on the sample trouble log sheet below.



TROUBLE LOG

Date ________
Time ________
Operator ______

Time C.E. Notified ________
Time Machine Fixed ________

Location of Problem ____________________

Machine Identification ____________________

Description of Problem ____________________

____

____

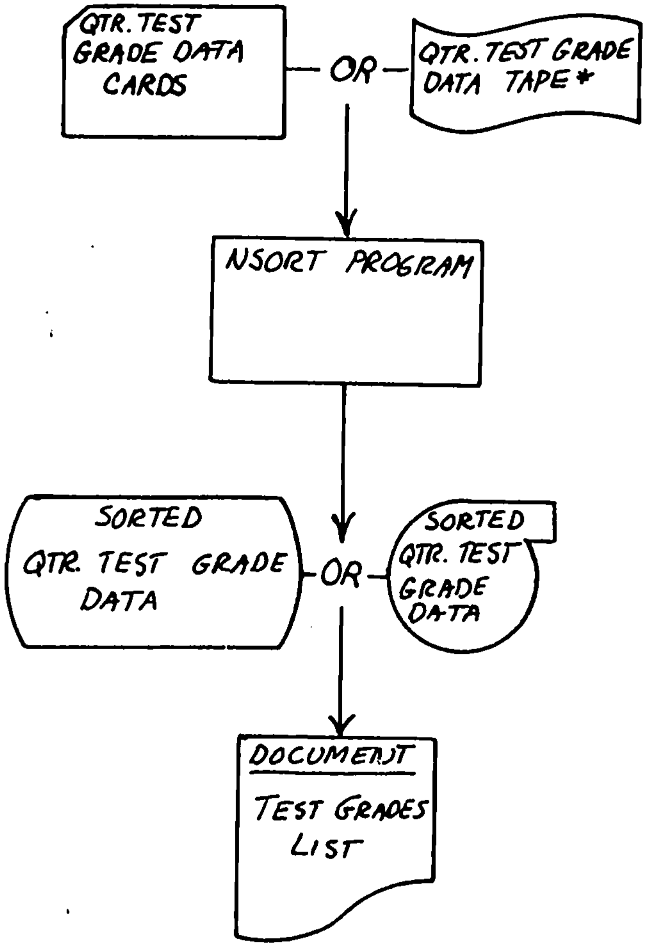
____

____

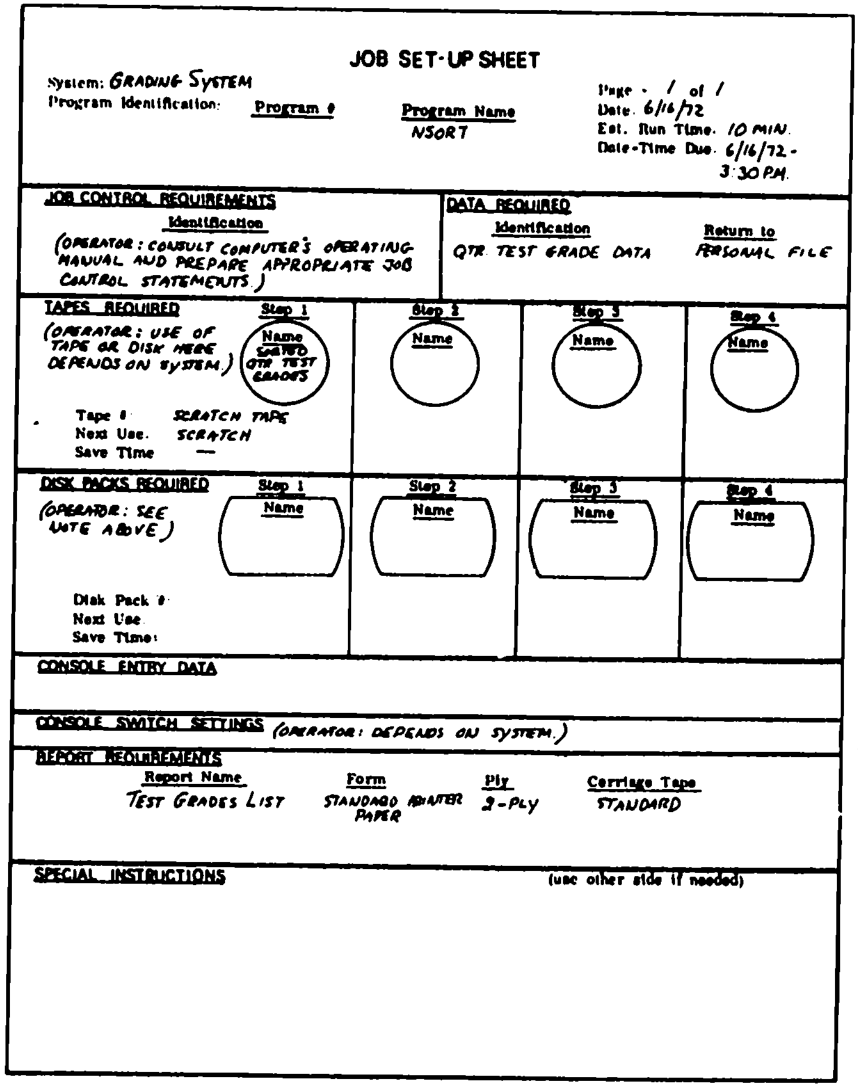
____

Fig. 88 Trouble log for Grading System job.

4. If trouble occurs and you must run the job over, do so. (Be sure to make a new entry for the run on your job log.)


QTR. TEST GRADE DATA CARDS
OR
QTR. TEST GRADE DATA TAPE *
NSORT PROGRAM
SORTED QTR. TEST GRADE DATA
OR
SORTED QTR. TEST GRADE DATA
DOCUMENT
TEST GRADES LIST

NOTE: The symbol ⌓ is used
to represent paper tape.



JOB SET-UP SHEET

System: GRADING SYSTEM
Program Identification: **Program #** **Program Name** NSORT

Page - 1 of 1
Date. 6/16/72
Est. Run Time. 10 MIN.
Date-Time Due. 6/16/72 - 3:30 PM.

JOB CONTROL REQUIREMENTS
Identification
(OPERATOR: CONSULT COMPUTER'S OPERATING MANUAL AND PREPARE APPROPRIATE JOB CONTROL STATEMENTS.)

DATA REQUIRED

Identification	**Return to**
QTR. TEST GRADE DATA	PERSONAL FILE

TAPES REQUIRED
(OPERATOR: USE OF TAPE OR DISK HERE DEPENDS ON SYSTEM.)

Step 1	**Step 2**	**Step 3**	**Step 4**
Name SORTED QTR TEST GRADES	**Name**	**Name**	**Name**

Tape #: SCRATCH TAPE
Next Use. SCRATCH
Save Time —

DISK PACKS REQUIRED
(OPERATOR: SEE NOTE ABOVE)

Step 1	**Step 2**	**Step 3**	**Step 4**
Name	**Name**	**Name**	**Name**

Disk Pack #:
Next Use
Save Time:

CONSOLE ENTRY DATA

CONSOLE SWITCH SETTINGS (OPERATOR: DEPENDS ON SYSTEM.)

REPORT REQUIREMENTS

Report Name	**Form**	**Ply**	**Carriage Tape**
TEST GRADES LIST	STANDARD PRINTER PAPER	2-PLY	STANDARD

SPECIAL INSTRUCTIONS (use other side if needed)

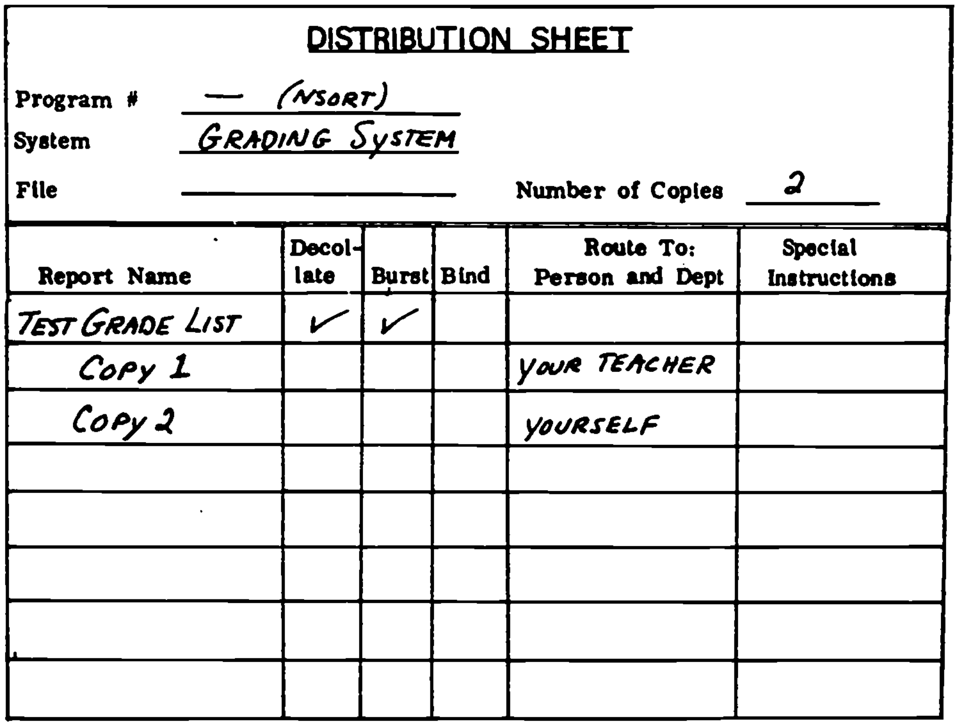
Fig. 87 Job-stream flowchart and set-up sheet for Grading System job.

Documenting the Job and Handling Output

1. Complete your job log for the run.

2. Document any input or output you need to, by making tabbed cards for card input, labelling any tape or disk output which will be saved, and so forth.

3. Assume that the following distribution instructions came with the other instruction sheets for this job. Follow the directions given.



DISTRIBUTION SHEET

Program # — (NSORT)
System GRADING SYSTEM
File ______________ Number of Copies 2

Report Name	Decol-late	Burst	Bind	Route To: Person and Dept	Special Instructions
TEST GRADE LIST	✓	✓			
COPY 1				YOUR TEACHER	
COPY 2				YOURSELF	

Fig. 89 Distribution sheet for Grading System job.

APPENDIX

COMPLETED DAILY WORK SCHEDULE FOR PROBLEM 5, PAGE 101

DAILY WORK SCHEDULE

OPERATIONS UNIT

Term Fall
Week 1-26 to 1-30
Day Tuesday
Date 1-27-72

JOB DESCRIPTION	RESCH.	TIME DUE	STD. RUN	SCHEDULED PROC. HRS.
Attendance Reports		9:30 a.m.	40 min.	.6
Compile - Activity Report		Noon	30 min.	.5
Update Supply Inventory File		10:30 a.m.	30 min.	.5
Senior Cumulative Grades		Noon	25 min.	.4
Student Address Labels		Noon	50 min.	.8
Update Bldg. Inventory File		3:30 p.m.	90 min.	1.5
Compile - Award List		4:30 pm.	25 min.	.4
Update Library File		3:30 p.m.	90 min.	1.5
Programmer Test Time		3:30 p.m.	30 min.	.5
Test - P.E. Equipment Inventory		4:30 p.m.	30 min.	.5
SCHEDULED MACHINE UTILIZATION				7.2
ADJUSTMENT FACTOR				.3
ADJUSTED MACHINE UTILIZATION				7.5

SCH. JOB NOT PROCESSED	REASON	DATE RESCH.

ANSWERS TO QUESTIONS ON PAGE 109

1. What programs are needed for the run? (Give names and identification numbers.)

   # 142 - CARD-TO-TAPE PROGRAM

   # 143 - PRINT LABELS PROGRAM

2. What medium are the programs on?

   DISK PACK (#000052)

3. What job control cards are needed DECKS #1 AND 2

4. Describe any data card decks needed for the run, giving the name of each.

   MAILING LABEL CARDS

5. What magnetic tapes are needed for the run. (Give names and num numbers.)

   SCRATCH TAPE #097

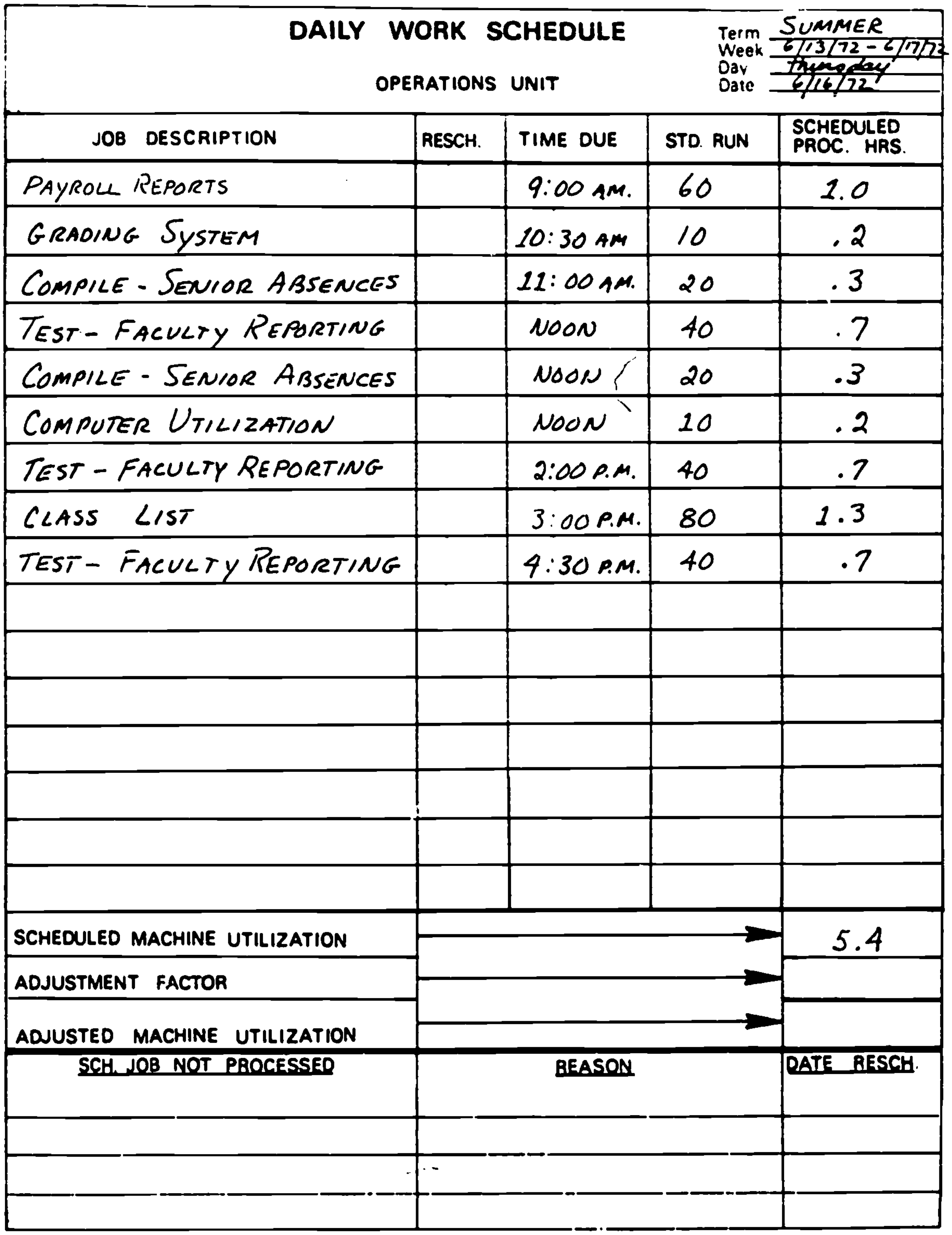
6. What disk packs are needed for the run? (Give names and numbers.)

   DISK PACK #000052 (WITH PROGRAMS ON IT) NO OTHERS.

7. Describe any special forms needed for this job. (Give names and descriptions of forms.)

   MAILING LABEL FORMS - GUMMED (#2x-2 ACROSS), ONE-PLY.

COMPLETED DAILY WORK SCHEDULE, FIG. 78, PAGE 140



DAILY WORK SCHEDULE

OPERATIONS UNIT

Term SUMMER
Week 6/13/72 - 6/17/72
Day Thursday
Date 6/16/72

JOB DESCRIPTION	RESCH.	TIME DUE	STD. RUN	SCHEDULED PROC. HRS.
PAYROLL REPORTS		9:00 AM.	60	1.0
GRADING SYSTEM		10:30 AM	10	.2
COMPILE - SENIOR ABSENCES		11:00 AM.	20	.3
TEST - FACULTY REPORTING		NOON	40	.7
COMPILE - SENIOR ABSENCES		NOON	20	.3
COMPUTER UTILIZATION		NOON	10	.2
TEST - FACULTY REPORTING		2:00 P.M.	40	.7
CLASS LIST		3:00 P.M.	80	1.3
TEST - FACULTY REPORTING		4:30 P.M.	40	.7

SCHEDULED MACHINE UTILIZATION →	5.4
ADJUSTMENT FACTOR →	
ADJUSTED MACHINE UTILIZATION →	

SCH. JOB NOT PROCESSED	REASON	DATE RESCH.

ANSWERS TO QUESTIONS, PAGE 144

Preliminary Questions for Payroll Reports Job

1. What job control cards are needed? (Give identification numbers.)

#25, 26, 27

2. What punched card data is required to run the job? (Give names and numbers.)

#13- Payroll Update Cards

3. What magnetic tapes must be found in the tape library? (Give names and numbers.)

Payroll Master Tape #121 and
Scratch tape #093

4. What disk packs are to be used? (Give names and numbers.)

Disk Pack #124682

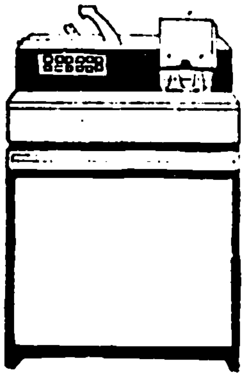
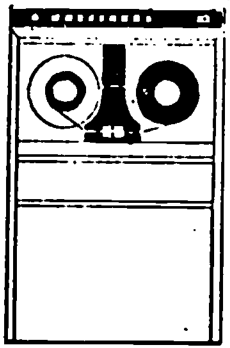
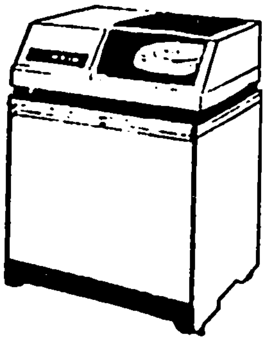
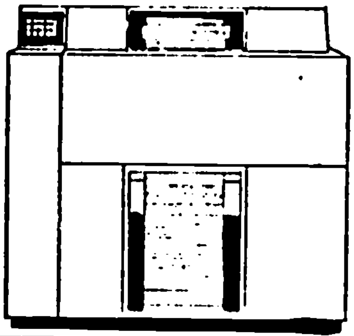
5. What data that is output on tape or disk pack during this run will be saved at the end of the job? (Give identification number.)

Sorted Updated Payroll Master Tape
and Deduction File on Disk Pack #124682

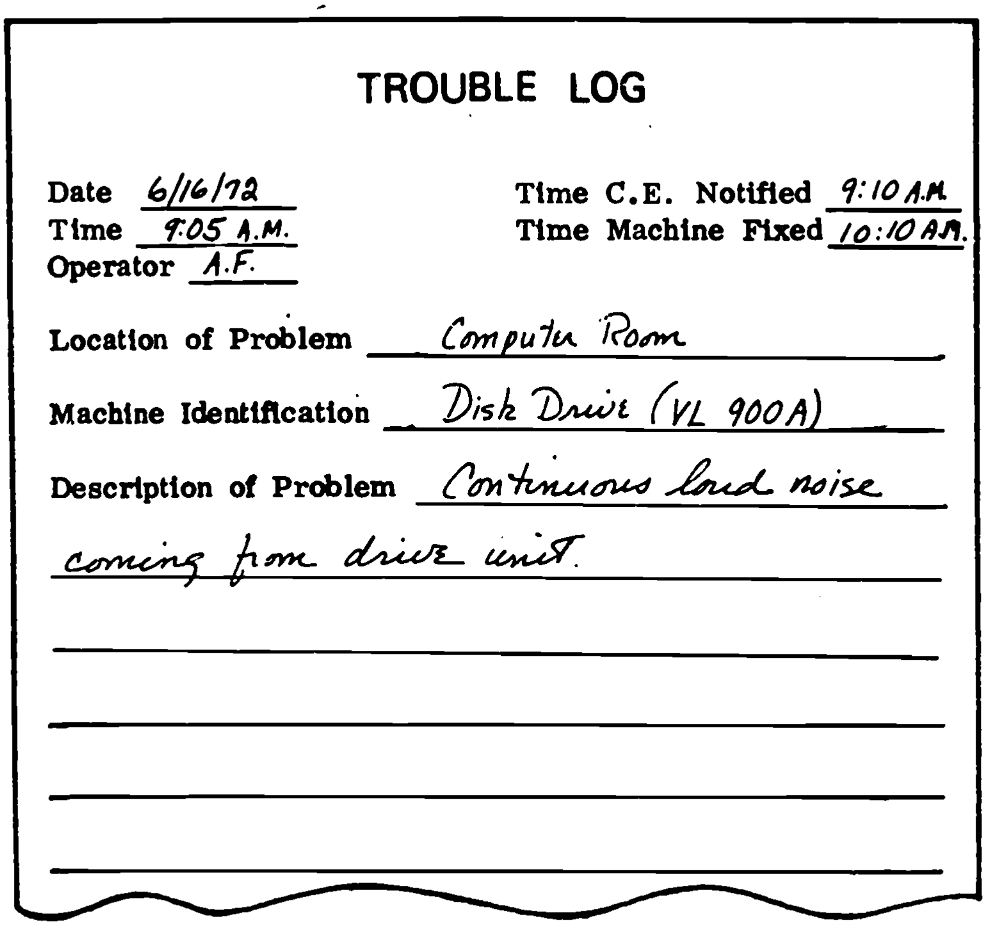
6. What outpit forms are needed for the printer in this run? (Give names ard form descriptions.)

Paychecks Form #308 (one-ply) and
Standard 3-ply report paper

COMPLETED SET-UP DIAGRAM, FIG. 81, PAGE 145

INPUT/OUTPUT DEVICES	INPUT TO BE LOADED ON DEVICE (Specify)	INPUT TO BE KEPT READY (if any)
	JOB CONTROL CARDS #25 PAYROLL UPDATE CARDS #13 JOB CONTROL CARDS #26 JOB CONTROL CARDS #27	- none -
	PAYROLL MASTER TAPE #121	SCRATCH TAPE #093
	DISK PACK # 15994 DISK PACK # 124682	.. none -
	PAYROLL CHECK FORMS #308	STANDARD PRINTING PAPER (3-PLY)

COMPLETED TROUBLE LOG, FIG. 83, PAGE 148



TROUBLE LOG

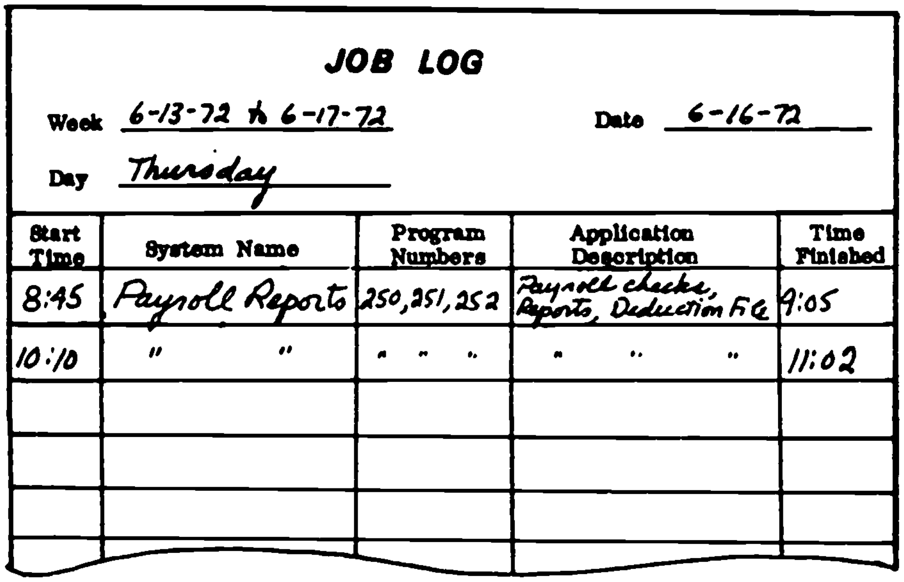
Date 6/16/72	Time C.E. Notified 9:10 A.M.
Time 9:05 A.M.	Time Machine Fixed 10:10 A.M.
Operator A.F.	

Location of Problem Computer Room

Machine Identification Disk Drive (VL 900A)

Description of Problem Continuous loud noise coming from drive unit.

COMPLETED JOB LOG, FIG. 82, PAGE 147



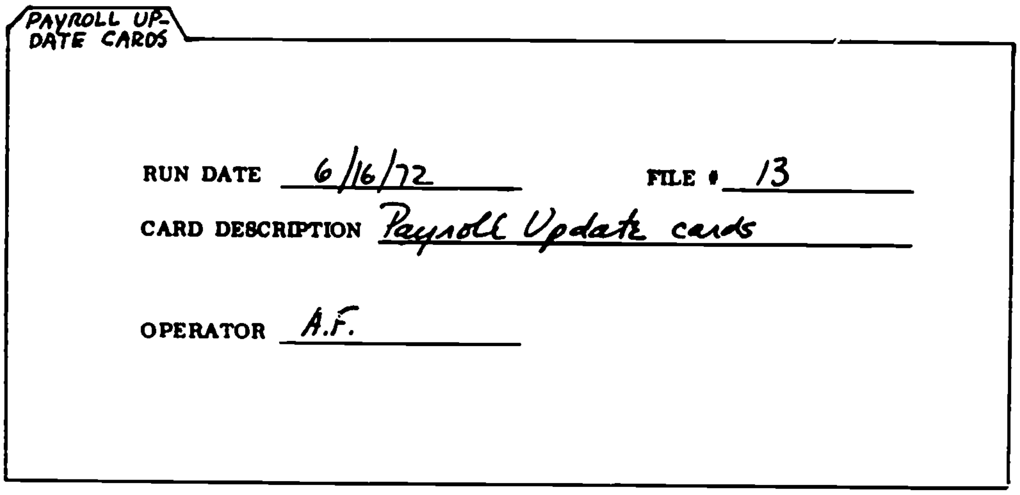
JOB LOG

Week 6-13-72 to 6-17-72 Date 6-16-72

Day Thursday

Start Time	System Name	Program Numbers	Application Description	Time Finished
8:45	Payroll Reports	250,251,252	Payroll checks, Reports, Deduction File	9:05
10:10	" "	" " "	" " "	11:02

COMPLETED TABBED CARD, FIG. 84, PAGE 149



PAYROLL UP-
DATE CARDS

RUN DATE 6/16/72 FILE # 13

CARD DESCRIPTION Payroll Update cards

OPERATOR A.F.

ANSWER TO DISTRIBUTION SHEET QUESTION, PAGE 150

"The report must be:

a) decollated (that is, the carbons are removed from between the three copies), and

b) burst (that is, the pages of each copy are separated at the perforated line)."

APPENDIX B

UNIT TEST

UNIT 3 -- THE ELEMENTS OF COMPUTER PROGRAMMING

Check the box(es) for the correct answer(s) below.

1. Which of the following specifications does the computer programmer receive from the systems analyst?

   □ a. The written analysis of an entire computer problem
   □ b. The forms in which output will be produced
   □ c. Complete operating procedures
   □ d. The form in which data will be input to the computer
   □ e. The forms of input and output to be used as well as a description of the process needed to produce the desired results

2. Which of the following describes the skills most necessary to the computer programmer?

   □ a. Thorough knowledge of key punching
   □ b. The ability to write programs in all programming languages
   □ c. Complete knowledge of computer operation
   □ d. The ability to analyze a problem and write a program to solve it

3. Which of the following is not true about the education and training required for computer programming?

   □ a. A college degree is helpful.
   □ b. On-the-job training is not available in most computer centers.
   □ c. No education is necessary for computer programming.
   □ d. A college degree is essential.
   □ e. Some training in computers and programming languages is necessary.

4. It is important that a person who wants to be a computer programmer

- □ a. has a logical and objective mind
- □ b. has the ability to understand number relationships and arithmetic
- □ c. has a highly creative mind
- □ d. requires a great deal of supervision

5. Number the steps you see below showing the order in which a programmer solves a computer problem.

______ Codes a program in a symbolic language
______ Tests the program for accurate results
______ Debugs the compiled program listing
______ Designs a flowchart of the problem
______ Documents the final program

6. The technique used to arrange steps of a problem in a specific order is called

- □ a. flowcharting
- □ b. coding the program
- □ c. debugging

7. The language which the computer understands and directly communicates in is called

- □ a. assembly language
- □ b. symbolic programming language
- □ c. machine language

8. Symbolic or "high-level" programming languages are

- □ a. used by the computer just as they are
- □ b. translated by the computer into machine language
- □ c. translated by the computer into assembly language

9. FORTRAN can solve a problem in

- □ a. complete sentences
- □ b. machine language
- □ c. numeric formulas or algebraic expressions
- □ d. numbers and letters

10. Which of the following is a symbolic language used to learn programming?

□ a. COBOL
□ b. BASIC
□ c. FORTRAN

11. Debugging a program is the process of

□ a. documenting all the results of running the program
□ b. finding and correcting all errors in the program
□ c. listing the program
□ d. writing the program

12. Tracing a program means

□ a. listing all program statements
□ b. writing a description of the program
□ c. checking step-by-step through the program logic
□ d. debugging the program cards or tape
□ e. flowcharting all the steps to a solution

13. Which of the following are points to check in the logic of a program?

□ a. Correct flowcharting symbols
□ b. Step-by-step logical sequence in the solution
□ c. Correctly spelled words
□ d. Accurately defined input and output
□ e. Correctly coded processes

14. A successfully compiled program is one which is

□ a. ready to be debugged
□ b. free of all errors
□ c. coded
□ d. ready to be run with the actual data to be used

15. Put the following debugging steps in sequence by numbering them 1 through 4.

______ The program is compiled.
______ Test run errors are corrected.
______ A test run is made.
______ Compile errors are corrected.

16. Which of the following finally proves the reliability of a program?

□ a. Accurate results or output
□ b. A test run of the program
□ c. A compiled program

17. Which of the following documentations is the responsibility of the computer programmer?

□ a. A computer operations manual
□ b. Information about how to run a program
□ c. A systems specification
□ d. General information about programs for computer center records

18. Which of the following manuals tells a programmer about computer equipment and software programs available?

□ a. An operating system manual
□ b. A programming language manual
□ c. A subroutine library reference guide
□ d. An "Introduction to Computers" manual

19. Check which of the following information might be found in a programming language manual.

□ a. "How to use paper tape"
□ b. "Elements of programming in BASIC"
□ c. "Reading data in FORTRAN"
□ d. "Arithmetic operations used in BASIC"

APPENDIX C

EVALUATION PLAN

COMPUTER TECHNOLOGY PROGRAM

"THE ELEMENTS OF COMPUTER CAREERS"

FY 1973

Northwest Regional Educational Laboratory
710 S.W. Second Avenue/Lindsay Building
Portland, Oregon 97204

Introduction

The Computer Technology Program (REACT) has developed, over the past year, a course for high school students entitled "The Elements of Computer Careers." It consists of seven manuals plus a User's Guide. Six of the manuals each specifically describe a job area in the computer field; the seventh is a general summary of opportunities in the field.

The course is designed for optimum use in situations where computer hardware is available for hands-on experience. The User's Guide, however, spells out optional approaches for schools which have only a terminal or no hardware at all.

In the 1972-73 school year, the course materials will be used in three different environments as a field test. This document explains the evaluation design which will be used to glean information from these field tests. It is organized into the following sections:

- Goals
- Sites
- Research Questions
- Measurement Techniques and Instrumentation
- Analyses

Goals

The goals of the evaluation phase are three:

- Collect data of interest to potential users
- Supply potential publishers with valid documentation
- Make final minor revisions in the materials

Sites

The course will be taught in three distinct field test settings:

- a school with a computer center
- a school with terminal access to a computer only
- a school with no hardware

The field test is really three field tests - all data will be analyzed and presented for each mode of instruction and in total. The technical report will include a section with descriptive data on test site schools, courses, and students. This data will be indicative of the degree to which the test site schools represent the ultimate target population.

Research Questions

The questions which are to be answered are in two categories (with some overlap) - information for potential users and publishers, and information for the developers.

For Potential Users and Publishers

1. Do teachers find the course useful? Will they continue to use it?
2. How long does the course run in the different modes?

3. What price will potential users pay for it?

4. How unique is the course? What does it compete with? How great is the demand or need for it?

5. How much technical support is needed? How can this be reduced?

6. How do the students respond to the course?

7. How many students can be handled at one time in the different modes? How self-instructional is the course? What level of training/competence must the teacher have? What special orientation is required?

8. What particular strengths, uniqueness, value should be stressed in disseminating the materials?

9. How satisfactory are the "compromise" modes? What unique features does the course have in the three modes? How can the greatest effectiveness be gained in each of the three modes?

For Developers

10. How well do students master the materials (post-test performance) in each of the three modes of presentation?

11. Where are blocks, gaps, errors, rough spots, etc.?

12. What further information needs to be included in the User's Guide?

13. How do the users (teachers and students) assess the style, format, exercises, problems, laboratories, etc.?

14. Are the problems realistic? Too hard, too long, too many, too confusing?

15. What would make the course more interesting or valuable?

Measurement Techniques and Instrumentation

The field tests are to be conducted with an absolute minumum of intervention by Laboratory staff. Data collection instruments, therefore, must be largely self-explanatory. Some baseline data can be collected (see Appendix A) during the brief orientation and a post-field test interview will also be held. Tests for achievement on the manuals themselves are built into the course.

Referring to the questions in the previous section, they will be answered at least partially in these ways:

Question #1	Teacher questionnaire and interview
Question #2	Observation and log book kept on each student (see sample in Appendix F)
Question #3	Interview with teachers, curriculum personnel, administrators, and consultants
Question #4	Same as #3
Question #5	Log book kept at test site and trouble log of telephone queries kept at the Laboratory (see sample in Appendix G)
Question #6	Observation and student and teacher questionnaire
Question #7	Observation, interview, log of teacher activity
Question #8	Observation, interview, questionnaire
Question #9	Same as #8
Question #10	Post-tests which are in the <u>User's Guide</u> (see sample in Appendix B)

Question #11 The students will be asked to write comments in the manuals as they go. Also, a trouble log will be kept (see sample in Appendix C)

Question #12 Questionnaire and/or interview with teachers during and after the course (see sample in Appendix D)

Question #13 Questionnaire (see sample in Appendix E)

Question #14 Same as #13

Question #15 Same as #13 supplemented by interview with teacher

Analyses

All analyses will be done as a whole and separately for each mode of instruction. Test scores will be analyzed in this way:

- Differences in scores are computed between pre and post-tests to show measures of growth.
- Post-test scores are scrutinized to see if performance objectives are being met.
- Item analyses are made to pinpoint areas needing attention. When the mean performance for a group is below 80% for a particular item, either the question or the content are revised as appropriate.

(See sample analyses in Appendix H)

Questionnaires, interview records, log books, and observations will be analyzed in this way:

- Responses are summarized and categorized as much as possible.

- The information is used in the revision process.
- A complete documentation of the experience and lessons of the field test will be included in the final technical report.

The technical report will include all data, the documentation of the field test, and recommendations for optimal use in the various modes.

APPENDIX A

STUDENT RECORD

1. Name ____

2. Grade ____________________ 3. Age ____________________

4. Approximate overall gradepoint average (C, C+, B, etc.) ______________

5. What background have you had in computers? (What courses, any other outside experience?)

6. What do you expect to learn from the course you are now in?

7. If you have any idea what you would like to be or do after high school, please explain.

APPENDIX B

UNIT TEST

UNIT 5 -- THE ELEMENTS OF SYSTEMS PROGRAMMING

Check the box(es) for the correct answer(s) below.

1. Which of the following statements best describes systems programming?

   □ a. It has replaced applications programming.
   □ b. It involves writing programs that tell the computer how to process certain data to output certain results.
   □ c. It is the kind of programming done in designing the software of the computer system.
   □ d. It is the kind of programming which uses only machine language.

2. Which of the following tasks does the manufacturer's systems programmer commonly do that is never done by the user's systems programmer?

   □ a. Make changes in the systems software
   □ b. Engineer the systems hardware
   □ c. Design the systems software
   □ d. Know how to use the software of a system

3. Which of the following skills is not a requirement for a systems programmer?

   □ a. Ability to write programs in high level languages
   □ b. Skill in programming in machine and assembly language
   □ c. Skill in written and spoken communication
   □ d. Ability to design elementary hardware for a system

4. Starting salaries for systems programmers are currently from

   □ a. $550 to $600 a month
   □ b. $600 to $675 a month
   □ c. $675 to $750 a month
   □ d. $750 to $1,000 a month

5. The lowest-level programming language and the most difficult to use for writing programs is

□ a. BASIC language
□ b. translator language
□ c. machine language
□ d. assembly language

6. What is the software of a computer system?

□ a. The wiring of the system
□ b. The input and output devices
□ c. The manuals and guides which explain how the system operates
□ d. The internal programs which make the system operate efficiently and automatically

7. Which of the following are <u>not</u> a part of systems software?

□ a. Translator programs
□ b. Applications programs
□ c. The supervisor
□ d. The operating system

8. An assembler is a translator program which translates

□ a. from assembly language into machine language
□ b. from assembly language into a high-level language
□ c. from machine language into assembly language
□ d. from a high-level language into machine language

9. A macro-instruction is

□ a. one instruction which must be translated into two or more machine instructions
□ b. several instructions together which must be translated into one machine-language instruction.
□ c. one instruction in any language which is used very often
□ d. one instruction in any language which can be used to mean many different things

10. An operating system is which of the following?

- □ a. A computer system developed by the systems analyst
- □ b. A systematic approach used by systems programmers to design solutions to problems
- □ c. A collection of programs which controls and coordinates a computer system's operation
- □ d. A central processing unit

11. Of the following software programs, which is a part of the operating system?

- □ a. The supervisor
- □ b. Input-output control programs
- □ c. The program loader
- □ d. All of the above are part of it

12. Which of the following techniques are not ordinarily used by systems programmers in their work?

- □ a. Documentation techniques
- □ b. Program debugging techniques
- □ c. Flowcharting
- □ d. None of these are used

13. The supervisor part of the operating system

- □ a. coordinates the input and output of the system
- □ b. monitors and controls all other programs in the operating system
- □ c. compiles applications programs
- □ d. does none of the above

14. A compiler is a translator program which

- □ a. converts assembly language into machine language
- □ b. converts problem-oriented languages into machine language
- □ c. translates machine language into BASIC
- □ d. translates assembly language into high-level languages

15. Which of the following describes multiprogramming?

□ a. It is programming which requires two or more arithmetic-logical units.
□ b. It is a kind of programming which uses several different programming languages at one time.
□ c. It is the programming used by the operating system.
□ d. None of the above describes multiprogramming.

16. Which of the following best describes the program loader?

□ a. A program which supervises the reading in and storage of programs
□ b. An input device used for loading programs
□ c. The switches on the computer's control panel
□ d. None of the above describes the program loader

17. Which of the programs in an operating system is the user's systems programmer responsible for writing?

□ a. The utility program
□ b. The input-output control routine
□ c. All of the programs in the operating system
□ d. None of the programs in the operating system

18. Which of the following are written as guides to the user of systems software?

□ a. Basic documentation
□ b. Reference manuals
□ c. Engineering specifications
□ d. Operation manuals

19. The first part oi systems software to be developed is usually

□ a. the supervisor
□ b. the assembler
□ c. input-output control routines
□ d. There is no usual order.

20. Following are the basic steps in developing systems software. Show the order in which the steps usually take place by putting the numbers 1, 2, 3, 4, 5, and 6 on the line beside the appropriate step.

______ Test and debug the software system as a working whole

______ Document the components of the system

______ Test and debug the individual components of the software system

______ Create the several parts of the operating system and all required compilers

______ Determine exactly what programs are to be developed for the software package

______ Teach trainers about the software system, so they can conduct training sessions for users

21. Which of the following would be a routine part of the job of the user's systems programmer?

□ a. Programming the systems software

□ b. Documenting the systems software

□ c. Decide what software to include in a system

□ d. Help other programmers to make the best use of systems software

APPENDIX C

TROUBLE LOG

When you have a problem with the Computer Careers course, please fill out a Trouble Log form. Follow these steps:

1. Try to solve the problem yourself or by using local resources. Double-check your User's Guide or other NWREL-supplied materials for the solution.

2. If you still need help, call the Computer Technology Program at the Northwest Regional Educational Laboratory in Portland Oregon (503-224-3650).

3. Fill out a Trouble Log form. Be as thorough as possible. The form will be most helpful if it helps us alleviate the problem in the future.

TROUBLE LOG

(Please use this for any kind of problem which occurs in the operation of the course)

1. Type of problem: □ Manual #__________

   □ Hardware

   □ Software

   □ Scheduling

   □ Other

2. Briefly describe the problem

3. Briefly tell how you responded to the problem

4. What do you suggest to alleviate this problem in the future?

Name________________________ □ Student

Date________________________ □ Teacher

□ Other______________________

(Use back for any further comments)

APPENDIX D

TEACHER QUESTIONNAIRE

Name____

School__________________________________ Date________________

Please answer these questions relating to the course, "Elements of Computer Careers," which you have field tested in your school. The number of respondents will be small, so please answer carefully and completely.

1. Would you continue to use these materials if you had a course they logically fit?

2. For what amount would you expect these materials to sell?

   How much would you be willing to pay for them if you were to use them?

3. How unique do you consider these materials? Are there other comparable courses you are familiar with? Do you think there is a heavy demand for materials such as these?

4. Not considering o\erhead costs (teacher salary, room space, etc.), what do you estimate it cost you to use these materials?

5. How many students completed the course in your school?

6. How many students do you feel you could have handled at one time in this course?

7. How self-instructional was the course? What level of competence does a teacher need to have? What special orientation does the teacher need?

8. How much outside support (help, trouble-shooting) was required from the Laboratory, etc.? How could this be reduced? Are there additional things you can mention that need to be in the User's Guide?

9. In the way you taught the course, how much time is appropriate (one semester, one year, etc.)?

10. How well did the students react to the materials? Comment.

11. Could you have effectively used the materials with no hardware (assuming you had some hardware)? What do you think is needed in the line of hardware to make the course most effective?

12. If you were selling the materials, what would you emphasize as strong points, uniqueness, value, etc.?

13. How do you personally evaluate the materials (style, format, problems, exercises), the teaching of the course, etc.? How could it be improved? Comment freely; use the back if necessary.

APPENDIX E

QUESTIONNAIRE

This form should be completed after finishing each manual and its test. Remember that this manual is in an experimental stage, and any comments you make will be helpful in making the manual better.

1. Name____

2. Manual #___________ 3. Date____________________________

4. What is your overall reaction to this manual (excellent, good, satisfactory, poor, etc.)?______________________________________

5. Did the manual teach what you expected or wanted to learn about the subject? Explain

6. Was the level of difficulty of the content of the manual appropriate (too easy, too hard, confusing, etc.)?

7. Were the problems and exercises appropriate (too easy, too hard, too long, too confusing, etc.)?

8. What would make this manual or the use of this manual more interesting or better in some way?

9. Make other comments (either good or bad) about the manual (length, style and format, illustrations, methods of presentation, appearance, etc.)

10. At this point in the course, do you have any suggestions for improving it?

APPENDIX F

TIME LOG

Manual	Date Started	Date Finished	Approximate number of days of Lab time during study of this manual (on computer, data preparation, projects, etc.)	Approximate number of days using outside resources during study of this manual (books, films, speakers, field trips, etc.
#1 Data Preparation				
#2 Computer Operation				
#3 Computer Programming				
#4 Systems Analysis				
#5 Systems Programming				
#6 Computer Center Management				
#7 Computer Careers in Perspective				

Name________________________________

APPENDIX G

TELEPHONE LOG

Test Site Calling______________________________

Name of Person Calling______________________________

Date____________________

Name of Person Handling Call______________________________

1. Type of Problem
   □ Manual #____________
   □ Hardware
   □ Software
   □ Scheduling
   □ Other

2. Briefiy Describe the Problem______________________________

   ____

   ____

   ____

3. What have site personnel already done to try to solve the problem?

   ____

   ____

   ____

   ____

4. Briefly tell how the problem was resolved______________________________

   ____

   ____

   ____

5. What is suggested to alleviate the problem in the future. Be thorough. (Use back of sheet if necessary.)

   ____

   ____

APPENDIX II

UNIT IV - ELEMENTS OF SYSTEMS ANALYSIS

QUESTION		PRE-TEST N = 7		POST-TEST N = 7		
NO.	PRT	CORRECT	%	CORRECT	%	GAIN
1		0	0.0	2	28.6	28.6
2		1	14.3	6	85.7	71.4
3	A	4	57.1	6	85.7	28.6
3	B	4	57.1	6	85.7	28.6
3	C	5	71.4	7	100.0	28.6
4		3	42.9	5	71.4	28.5
5		2	28.6	6	85.7	57.1
6	A	4	57.1	7	100.0	42.9
6	B	4	57.1	7	100.0	42.9
6	C	4	57.1	7	100.0	42.9
7		1	14.3	5	71.4	57.1
8		2	28.6	5	71.4	42.8
9		1	14.3	7	100.0	85.7
10		1	14.3	3	42.9	28.6
11		3	42.9	6	85.7	42.8
12		2	28.6	4	57.1	28.5
13		4	57.1	5	71.4	14.3
14		2	28.6	5	71.4	42.8
15		0	0.0	5	71.4	71.4
16	A	4	57.1	6	85.7	28.6
16	B	4	57.1	7	100.0	42.9
16	C	5	71.4	6	85.7	14.3
17		0	0.0	5	71.4	71.4
18		2	28.6	4	57.1	28.5
19		0	0.0	4	57.1	57.1
20		2	28.6	5	71.4	42.8
21	A	5	71.4	7	100.0	28.6
21	B	4	57.1	6	85.7	28.6
21	C	5	71.4	6	85.7	14.3
22		1	14.3	4	57.1	42.8
25	1	0	0.0	3	42.9	42.9
25	2	0	0.0	4	57.1	57.1
25	3	0	0.0	3	42.9	42.9
25	4	0	0.0	3	42.9	42.9
25	5	1	14.3	6	85.7	71.4
			32.65		74.68	42.03

	PRE-TEST N = 35		POST-TEST N = 35		
STUDENT	CORRECT	%	CORRECT	%	GAIN
1	16	45.7	19	54.3	8.6
2	20	57.1	32	91.4	34.3
3	11	31.4	29	82.9	51.5
4	0	0.0	18	51.4	51.4
5	17	48.6	32	91.4	42.8
6	7	20.0	30	85.7	65.7
7	9	25.7	23	65.7	40.0
		32.64		74.69	42.04