DOCUMENT RESUME

ED 152 274 IR 005 729

TITLE A Course of Study for Data Processing I.
INSTITUTION Montgomery County Public Schools, Rockville, Md.
PUB DATE 77
NOTE 257p.; Best copy available

EDRS PRICE MF-$0.83 HC-$14.05 Plus Postage.
DESCRIPTORS Annotated Bibliographies; *Business Education; *Computer Science Education; *Course Descriptions; *Data Processing; Guidelines; *Instructional Design; *Programing; Senior High Schools

ABSTRACT

The purpose of this publication is to serve as a course of study for Data Processing I, which is designed for 11th and 12th grade students in a business education program. Intended to give guidelines to school staff and teachers for planning and organizing instructional programs, it lists program goals, student outcomes, and unit activities. The structure of the guide permits use for traditional group, small group, or self-paced instruction. The nature of the course requires the use of equipment for hands-on training, and the material presented is oriented to the use of a computer system. An annotated bibliography includes teacher reference materials and supplementary books. Appendices include basic concepts of data processing, computer concepts, computer operations I and II, introductory and intermediate Report Program Generator Programming, and introductory and intermediate COBOL. (VT)

A Course of Study for DATA PROCESSING I

Montgomery County Public Schools
Rockville, Maryland

Table of Contents

	Page
Acknowledgments	v
Introduction	vii
Overview	viii
Units	
1. Basic Concepts of Data Processing	1
2. Computer Concepts	13
3. Computer Operations I	23
4. Computer Operations II	35
5. Introduction to Report Program Generator (RPG) Programming	49
6. Intermediate Report Program Generator (RPG) Programming	61
7. Introduction to COBOL	71
8. Intermediate COBOL	83
Annotated Bibliography	
Teacher Reference Material	95
Textbooks	96
Supplementary Books	98
Periodical	98
Appendices	
A. Basic Concepts of Data Processing	A-1
B. Computer Concepts	B-1
C. Computer Operations I	C-1
D. Computer Operations II	D-1
E. Introduction to Report Program Generator (RPG) Programming	E-1
F. Intermediate Report Program Generator (RPG) Programming	F-1
G. Introduction to COBOL	G-1
H. Intermediate COBOL	H-1

ACKNOWLEDGMENTS

During the 1974 summer curriculum workshop, the first draft of a course of study for Data Processing I was compiled under the general supervision of James J. Toquinto, Career and Vocational Division. Workshop participants were Patricia F. Bowman, Gaithersburg High School, James R. Hough (workshop leader), Albert Einstein High School, and Norman L. Miller, Montgomery Blair High School. Consultants to the workshop were:

Mary Jo Borreson, Editor
Division of Publications Services

Robert S. Kennedy, Marketing Representative
International Business Machines

John M. Lloyd, Assistant Professor
Montgomery College

James Meyerberg, Teacher Specialist
Department of Pupil and Program Appraisal

John R. Pancella, Assistant Director
Department of Pupil and Program Appraisal

Lois Parker, Coordinator
Career and Vocational Education

Joseph J. Ramach, Specialist in Data Processing
Maryland State Department of Education

William M. Richardson, Director
Department of Advance Planning and Development

Gerald Woofter, Teacher Specialist
Department of Curriculum and Instruction

During the summer of 1975, the first draft was refined and revised by the members of the earlier workshop. Consultants to this second workshop were:

James Myerberg, Teacher Specialist
Department of Pupil and Program Appraisal

Catherine Morgan, Director
Computer Assisted Instruction

Joseph J. Ramach, Specialist in Data Processing
Maryland State Department of Education

INTRODUCTION

The purpose of this publication is to serve as a course of study for Data Processing I. It is intended to give guidelines to school staffs for planning and organizing the instructional program as well as to help teachers make appropriate decisions about teaching. Program goals, student outcomes, and unit activities are listed as a guide to help the teacher in presenting, evaluating, and expanding the curriculum materials. The structure of this guide permits the teacher to use the material for traditional group instruction, for small group instruction, or for students' progressing at their own rate.

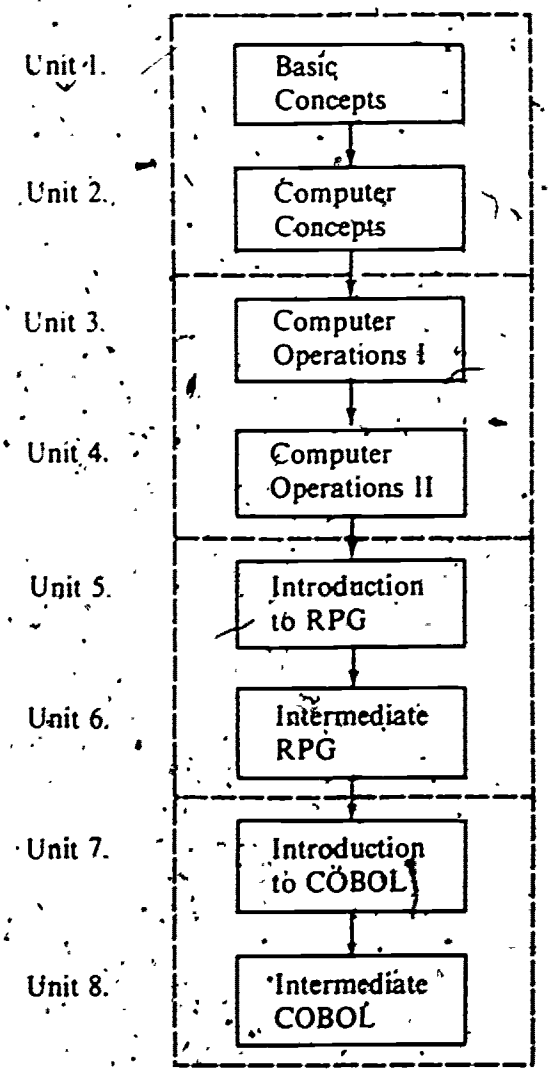
The nature of the course requires the use of equipment for *hands-on* training, and the material presented is oriented to the use of a computer system.

The block diagram which follows indicates the sequential placement of the nine-week units (modules), either for a one-hour (solid block) or a two-hour (dotted block) program.

Students will be expected to complete successfully the objectives of each unit, either by participation or by examination, before proceeding to the next unit.

DATA PROCESSING I

Nine-Week Modules



OVERVIEW

There is little doubt that sophisticated data processing is not only with us to stay but is having an increasing impact upon our lives as private citizens. Instruction in this field is necessary not only for vocational purposes but also for general basic data processing technology and systems as a consumer topic.

Data Processing I is intended for 11th and 12th grade students in the Montgomery County Public Schools as an elective offering. According to the MCPS policy on Evaluation and Reporting Student Progress, evaluation will be based on evidence of the attainment of the instructional and performance objectives for the course. Since this is an elective course not specifically required by the Maryland State Department of Education for graduation, students may elect a Credit No Credit basis for evaluation, subject to the conditions outlined in the Evaluation and Reporting Student Progress policy.

The course supports all of the objectives of the business education program and specifically aligns itself with the *primary* purpose of Business Education as stated in the *Program of Studies* ". . . to prepare students for entrance into business careers and to enable them to render efficient service in their vocations so that they may advance to higher levels of employment."[1]

Upon completion of Data Processing I, most students should:

- understand the historical development of the computer, its current and future uses
- understand the computer's role and influence on daily lives
- understand basic functions of any computer system — input, storage, control, arithmetic logic, and output
- be able to use the computer effectively during post high school education
- acquire a background of data processing concepts adequate for job entry in the field
- differentiate between computer data processing and earlier data processing
- use computer techniques to solve business application problems

While the course objectives are compatible with all the MCPS goals of education, they address themselves more specifically to the Career Development area. The course also supports the total Career Education effort in the county by making available the opportunity for specialization, the third component of the Career Education design. The Career Development area of the MCPS goals of education states.

The school must help each student gain:

- knowledge and appreciation of the wide variety and interrelationships of occupations in modern society
- opportunities to explore potential occupations in relation to personal aptitudes and interests, unrestricted by stereotypes of sex, race, or socio-economic level
- the knowledge, skills, and abilities that enable him [her] to secure satisfying employment, embark upon further training and education in a chosen career field, and adapt occupational talents to changing job demands and opportunities

1 "Business Education." *Program of Studies, Vol. 3.* Rockville: MCPS. 1976.

UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

This unit will give the student an overview of the historical developments in data processing. The unit also covers aspects of the meaning, reasons, and needs for automated means of processing information. An introductory explanation of the principles of punched card processing and computer concepts will be developed.

At the completion of the unit BASIC CONCEPTS OF DATA PROCESSING, the student should be able to:

- Describe some of the chronological developments of counting devices
- Explain the processing of data in our expanding world of business
- Identify and explain the functions of the card punch machines
- Record data in punched code by operating the card punch machine
- Classify data by operating the card sorter
- Recognize the history, types, purposes, and functions of computer devices

UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

The student should be able to describe some of the chronological developments of counting devices.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Identify some of the major counting devices, and name the inventor of each.

Given a list of devices and concepts and a list of inventors and contributors, match them.

Describe briefly the development of counting devices in this country, and know the contributors or inventors of each.

Given a list of devices and concepts and a list of inventors and contributors, match them.

UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

The student should be able to explain the processing of data in our expanding world of business.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Differentiate between the various types of information processing.

Write a description of the differences among four of the five basic methods of information processing.

Follow the flow of information and work activities through a business corporation, and relate it to the functions of the data processing department.

Draw a diagram showing the relationship between the computer department of a manufacturing corporation and the flow of data through the corporation. Show the flow lines into and out of the computer department. (Appendix A-1)

Describe the five phases of processing data.

Given an unarranged list of names and descriptions, match the name of each phase of processing data to the description.

State the needs for the change from manual to punched card processing of data.

List and explain some reasons for changing from manual to punched card processing of data.

UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

The student should be able to identify and explain the functions of the card punch machines.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Describe the characteristics of an 80-column card and code data fields using the Hollerith code.

Code your name and address using an 80-column card.

Compare the differences between the 80- and 96-column punched card.

Given an unarranged list of characteristics, align the characteristics under the heading 80-column card or 96-column card.

List the purpose and the differences between various card punch machines.

Given a written list of unique features of each card punch device, match these unique features with the machine model described.

Write a statement giving the purpose of a card punch machine.

Give the purpose and the differences between three card verifiers.

Write the purpose of the card verifier in relation to the card punch machine.

Identify the purpose and the differences between the three IBM card sorters.

State the different sorting speeds and uses of the IBM sorters.

Explain the purpose and the job applications for a card reproducer.

State the purpose of a card reproducer, and list some of the job applications it can perform.

Identify the five major uses of the collator.

State the purpose of a collator, and list some of the job applications it can perform.

Explain the purposes and the uses of the accounting machine.

Write a statement giving the purpose of an accounting machine, and list some of the job applications it can perform.

Describe the purpose and the positions of printing used by the card interpreter.

Write a statement giving the purpose of a card interpreter, and list some of the job applications it can perform.

Culminating Activity:

Match a list of card punch machine names and numbers with a list of the unique uses of each machine.

UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

The student should be able to record data in punched code by operating the card punch machine.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Name, locate, and use all parts of the card punch machine to solve a variety of problems.

Using an illustration of the IBM 29 card punch, label the machine parts by writing the part name and by drawing a line to the location of the part. (Appendix A-2)

Using an illustration, list the functional key to be depressed to move one card from one station to another station as you assume the punching of columns 1-67 without program control. (Appendix A-3)

Punch cards according to a series of predefined instructions. (Appendix A-4)

Design and punch a program card.

Design and punch program cards according to the problem directions. (Appendix A-5)

UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

The student should be able to classify data by operating the card sorter.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Name, locate, and use all parts of the card sorter to solve a variety of problems.

Using an illustration of the card sorter, label the machine parts, and indicate the card feed path. (Appendix A-6)

Identify the different sorting speeds and uses of the IBM sorters.

Label a diagram showing the basic principles of electricity as they apply to all unit record machines, except for the card punch.

Given a schematic, describe the basic principles of electricity in relation to the sorter timing cycles. (Appendix A-7)

Culminating Activity:

Given a deck of punched cards, sort the cards first numerically and then alphabetically.

Perform a card sorter selection problem using X code punch and multiple numeric code punches in one code column.

UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

The student should be able to recognize the history, types, purposes, and functions of computer devices.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Recall the contributions made to the historical development of computers.

Match a list of early computers and computer technology to a list of inventors or contributors.

Describe the functions of each of the five parts of a computer system and follow the flow of data through the computer.

Sketch, label, and explain the five computer components, and show the flow of data through the computer system.

Identify the cause and effect of the change from punched card to computer processing.

List the differences between the processing of data by card punch machines and by computers.

Identify four types of computers.

List several types of computers and describe their differences.

UNIT 1. OUTLINE	REFERENCE
I. An Historical Survey of Computing Devices	Crawford
A. Finger counting	
B. Stone counting	
C. The abacus	
D. Pascal's accounting machine (1642)	
E. Leibnitz's four functions machine (1671)	
F. Babbage's difference engine (1811)	
G. Burroughs' adding machine (1885)	
H. First use of paper tape (1728) Bouchon and Falcon	
I. First use of punched cards (1801) Joseph Jacquard	
J. Hollerith code (1887)	
K. History of IBM (1896-1924)	
L. James Powers (1908)	
M. History of UNIVAC (1911-1955)	
II. Types of Information Processing	
A. Manual, DP	
B. Mechanical, ADP	
C. Card punch machine, CPDP	
D. Integrated, IDP	
E. Electronic, EDP	
III. Business Organization	Awad
A. Principles of business management	
1. Sole proprietorship 2. Partnership 3. Corporation	
B. Three types of business activities	

C. Information flow

1. Purchasing
2. Receiving
3. Disbursements
4. Stockkeeping
5. Production
6. Sales
7. Billing and collecting
8. Delivery

IV. Data Manipulation

A. Collectpon of source documents

B. Recording and coding

C. Classifying and sorting

D. Calculating

E. Reportpng and summarizing

V. Reasons for Studying Data Processing

A. Changes in society

B. Changes in science

C. Changes in industry

VI. Need for Change from Manual to Punched Card Data Processing

A. Clerical worker shortage

B. Increased volume of data

C. Accuracy and control

D. Economy

E. Improved reporting

REFERENCE

VII. Punched Card Data Processing — Schnake

A. The punched card

1. Physical characteristics
   a) Size and shape of 80-column card (3¼ x 7⅜)
   b) Rows and columns
   c) Edges and face
   d) Identifying marks
   e) Size and shape of 96-column card
2. Recording data in Hollerith code
3. Card field format
4. Card record
5. Card file
6. Advantages and limitations of data in cards

B. The card punch machine

1. Description
2. Purpose
3. Differences between
   a) IBM 026
   b) IBM 029
   c) IBM 129
   d) IBM 5496 (96-column data recorder)
   e) UNIVAC 1710

C. The card verifier

1. Description
2. Purpose
3. Differences between
   a) IBM 056
   b) IBM 059
   c) IBM 129

D. Card sorter

1. Description
2. Purpose
3. Differences between
   a) IBM 082
   b) IBM 083
   c) IBM 084
4. Machine functions
   a) Numeric
   b) Alphabetic
   c) Selection
   d) Block sort

E. Card reproducer

1. Description
2. Purpose
3. Uses
   a) Straight reproduce
   b) Offset reproduce
   c) Gangpunch
   d) Intersperse gangpunch

F. Collator

1. Description
2. Purpose
3. Uses
   a) Sequence check
   b) Matching with selection
   c) Merging
   d) Selection
   e) Match and merge with selection

G. Accounting machine (tabulator)

1. Description
2. Purpose
3. Uses
   a) Detail print (list)
   b) Group indicate
   c) Group print
   d) Group print and summary punch

H. Interpreter

1. Description
2. Purpose
3. Uses
   a) Straight interpret
   b) Offset interpret
   c) X elimination or transfer
   d) Interpreting on upper or lower line
   e) Multiple printing

VIII. Card Punch Machine Operation — Wanous, et al.

A. Card path

B. Machine parts

C. Combination keyboard

1. Functional control keys
2. Duel and single character keys
3. Functional control switches

D. Off-registered cards

E. Program control unit

1. Program control level
2. Program drum removal and insertion

F. Program card design

1. Field definition
2. Start automatic skip
3. Start automatic duplication
4. Alphabetic shift
5. Start alphabetic duplication
6. One level program — row 12-1
7. Two level program — row 12-1 and row 4-7

IX. Card Sorter Machine Operation

A. Card path

B. Machine parts

C. Timing cycles

D. Card alignment and handling

1. Check for card positioning of corner cuts
2. Align
3. Flex
4. Fan
5. Joggle
6. Realign

E. Removing card jams

F. Replacing sort brush

G. Check brush timing

H. Uses

1. Numerical sort
2. Alphabetical sort
3. Block sort
4. Card selection

I. Card checking

1. Sight check
2. Sort needle

X. An Historical Survey of Computer Systems and Devices — Crawford

A. History

1. Difference engine
2. Mark I
3. ENIAC
4. Stored program
5. UNIVAC

B. Technology

1. First generation (1946-59)
2. Second generation (1959-65)
3. Third generation (1965-70)
4. Fourth generation (1970-)

XI. Overview of Components of a Computer System

A. Input

B. Central processing unit

1. Storage
2. Control
3. Arithmetic logic (processing)

C. Output

XII. Need for Change from Punched Card Processing to Computer Processing

A. Explosion of data

B. Speed of processing

1. Millisecond
2. Microsecond
3. Nanosecond

C. Greater problem application

XIII. Types of Computers

A. Analog

B. Digital

1. General purpose
2. Special purpose

C. Hybrid

D. Minicomputers

RESOURCES FOR UNIT 1. BASIC CONCEPTS OF DATA PROCESSING

Teacher References

Eastern Business Teachers Association. *Processing Data in Business, Education, and Government.* Somerville, N.J.: EBTA, 1972.

Schnake, M. A. *Data Processing Concepts.* New York. McGraw-Hill Book Company, 1973.

Textbooks

Awad, Elias M. *Business Data Processing.* Englewood Cliffs, N.J.. Prentice-Hall, Inc., 1971.

Crawford, F. R. *Introduction to Data Processing,* 2nd ed, Englewood Cliffs, N.J.. Prentice-Hall, Inc., 1973.

International Business Machines Corporation. *Basic Punched Card Data Processing Programmed Instruction Manuals.* White Plains, N.Y.: IBM, 1964.

Wanous, E. F., Wagner, Gerald E., and Wanous, S. J. *Fundamentals of Data Processing.* New Rochelle, N.Y.: South-Western Publishing Company, 1971.

Supplementary

Minnesota Mining and Manufacturing Company. *Data Processing,* Vols. 1-4, and a corresponding set of transparencies. St. Paul, Minn.: 3-M Company, Visual Products Division, 1966.

UNIT 2. COMPUTER CONCEPTS

This unit will give the student a knowledge of a computer system, its internal means of coding data, its base numbering system, and the concept of programming a computer system. The unit also presents the concept of the programming cycle from the initial presentation of a problem through to its solution.

At the completion of the unit COMPUTER CONCEPTS the student should be able to.

- Identify forms of computer hardware devices
- Describe different types of computer software
- Discuss the organization, methods of processing, and limitations of a data processing facility
- Recognize and apply numbering and coding systems used in computer operations
- Translate a computer problem statement into a graphic flowchart and follow the problem through the programming cycle
- Identify career opportunities in data processing

UNIT 2. COMPUTER CONCEPTS

The student should be able to identify forms of computer hardware devices.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Classify computer hardware devices as either input, output, or both input and output.

Given the names of 15 devices, indicate which ones are input only, output only, or both input and output.

Recognize various technology used in primary storage devices.

Name and briefly describe some of the types of memory used as primary storage devices.

UNIT 2. COMPUTER CONCEPTS

The student should be able to describe different types of computer software.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Explain the use of language translators.

Name the types of language translators and describe their functpons.

Identify the characteristics of several common computer programming languages.

State the major characteristics for the following programming languages. SPS, RPG, COBOL, FORTRAN, ALC, PL/1.

Explain the relationship of language translators to program languages.

Describe the function of a language translator as it is used in compiling a programmer's language.

Distinguish between hardware and software.

Draw a diagram showing the relationship between the hardware and software in a computer system. (Appendix B-1)

UNIT 2. COMPUTER CONCEPTS

The student should be able to discuss the organization, methods of processing, and limitations of a data processing facility.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Recognize the limitations of a computer system.

List some limitations of a computer system.

Distinguish between the different methods of computer processing and their application as applied to specific business needs.

State the types of data processing organizations and a unique characteristic of each.

List and explain the methods of processing data in an in-house data processing organization.

Match descriptive statements to the following terms:

in-house
time sharing
service bureaus
batch processing
multiprogramming
multiprocessing
facilities management

UNIT 2. COMPUTER CONCEPTS

The student should be able to recognize and apply numbering and coding systems used in computer operations.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Apply the concepts of computer arithmetic to counting, calculating, and converting numbers in different number bases.

Count from 0-32 in binary, octal, and hexadecimal.

Perform calculations, by adding and subtracting in binary, on a series of arithmetic problems.

Perform number conversion in three number bases.

Perform computer arithmetic quiz. (Appendix B-2)

Code alphabetic and numeric information in several coding systems.

Show a comparison of coding systems by coding your name and address in Hollerith, BCD, EBCDIC, and ASCII.

UNIT 2. COMPUTER CONCEPTS

The student should be able to translate a computer problem statement into a graphic flowchart and follow the problem through the programming cycle.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Recognize the shape and purpose for each flowchart symbol.

Identify each standardized symbol on a flowchart template.

Follow the flow of data through a business organization and through its computer system.

Draw a flowchart showing the flow of data through a business organization.

Given a problem statement and a predrawn flowchart, follow the flowchart through the processing steps to arrive at a final result. (Appendix B-3)

List and explain the meaning of the steps in the computer programming cycle.

List the steps in the programming cycle in their correct order from the receipt of a problem statement to documentation of the finished computerized job.

Given an unarranged list, arrange the steps in the programming cycle.

Classify and explain each item that makes up problem documentation.

List in sequential order the items that make up the documentation of each computer job.

UNIT 2. COMPUTER CONCEPTS

The student should be able to identify career opportunities in data processing.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Name the data processing career jobs from the lowest level job titles to the top level.

Outline the career opportunities in data processing, showing the various career paths from a beginning worker to the top level position. (Appendix B-4).

Relate the data processing career opportunities to the education and training required

Select data processing want ads from newspapers and look for the experience required and the salary offered.

Write a description for the major classes of data processing jobs, listing the following items:

entry skills required
job duties
salary range
future job outlook
advancement possibilities
types and locations of industries where jobs are available

UNIT 2. OUTLINE	REFERENCE
I. Composition of a Computer System	
A. Hardware	Marxer
1. Input devices and appropriate flowchart symbols	
a) Card reader	
b) Magnetic tape	
c) Magnetic disk	
d) Magnetic drum	
e) Punched paper tape	
f) Magnetic ink character reader	
g) Optical character reader	
h) Terminal	
i) Display station	
j) Console typewriter	
k) Data cell	
l) Other	
2. Central processing unit	
a) Primary/main storage devices	
(1) Magnetic core memory	
(2) Thin-film memory	
(3) Cryogenic memory	
(4) Photo-digital memory	
(5) Electro-optical memory	
b) Word length computers	
(1) Fixed	
(2) Variable	
c) Control	
d) Arithmetic/logic unit	

3. Output devices and appropriate flowchart symbols a) Card punch b) Magnetic tape c) Magnetic disk d) Magnetic drum e) Punched paper tape f) Line printer g) Terminal h) Display station i) Console typewriter j) Microfilm/microfiche k) Plotter l) Data cell m) Audio response	Schnake
B. Software	
1. Language translators a) Assemblers — low level b) Compilers — high level	
2. Programming languages a) Symbolic Program System (SPS) b) Report Program Generator (RPG) c) Common Business Oriented (COBOL) d) Formula Translation (FORTRAN) e) Assembly (ALC) f) Programming Language 1 (PL/1) g) Other	Marxer
C. Peopleware — the data processing staff who do:	Awad
1. Problem definition/specifications 2. System analysis and design 3. Program development — programming 4. Input data preparation 5. Program loading — operations	
II. Limitations of a Computer System	Awad
A. Inability to handle unprogrammed information	
B. Inability to make decisions independently	
C. Occasional breakdown	
D. Impracticality of use on nonrecurring or nonrepetitive calculations	

III. Computer Size — Capacity

A. Minicomputer

B. Small computer

C. Medium size computer

D. Large scale computer

IV. Processing Organizations

A. In-house

B. Time sharing

C. Service bureaus

D. Facilities management — Miller

V. Processing Methods

A. Batch

B. Multiprogramming

C. Multiprocessing

VI. Computer Arithmetic — Crawford

A. Place value concept of numbering systems

1. Binary
2. Octal
3. Hexadecimal

B. Adding and subtracting in binary

C. Number conversion

1. Binary
   a) Decimal to binary
   b) Binary to decimal
2. Octal
   a) Decimal to octal
   b) Octal to decimal
   c) Octal to binary
3. Hexadecimal
   a) Decimal to hexadecimal
   b) Hexadecimal to decimal
   c) Hexadecimal to binary

VII. Types of Internal Storage Coding Systems

A. Review of Hollerith

B. BCD

C. EBCDIC

D. ASCII

VIII. Flowcharting Concepts — Marxer

A. System flowchart — Schnake

1. Flow of data through the organization — Awad
2. Flow of data through the computer system

B. Program flowchart

1. Decision table
2. Detail problem flowchart

IX. The Programming Cycle

A. Read the problem statement.

B. Organize:

1. Input-card layout
2. Output-printer-spacing chart/carriage control tape

C. Develop problem solution:

1. Decision table
2. Flowchart

D. Explain how the formal problem specification is developed.

1. Oral/written problem statement
2. Sample input information
3. Sample output information
4. Processing to be performed
5. Explanation of application
   a) Logic for solution
   b) Equipment
   c) Computer language
   d) Source of test data
6. Job specifications

E. Code the source program.

F. Punch:

1. Source program
2. Test data
3. Job control language cards
   a) Explain their function.
   b) Explain the order of JCL cards in the run deck.

G. Test the program:

1. Print program list and desk check.
2. Compile.

H. Debug program diagnostic errors.

I. Recompile and run test data to test program logic.

J. Organize the documentation:

1. Problem statement and specifications
2. Decision table/flowchart
3. Source document sample
4. Input format — record layout
5. Output format — printer-spacing chart/carriage control tape
6. Compiled program listing
7. Output result of test data
8. Operator-run instructions

X. Career Opportunities and Training in Data Processing — Wanous, et al.

A. Data control

B. Data entry

C. Tape librarian — *Dictionary of Occupational Titles*

D. Card punch machine operator

E. Computer operator

F. Programmer — Daily newspapers

G. EDP system analyst

H. System analyst

I. Manager/director — Computer periodicals

J. Computer repair technician

K. Manufacturer software programmer — State of California

L. EDP systems design engineer

RESOURCES FOR UNIT 2. COMPUTER CONCEPTS

Teacher References

Eastern Business Teachers Association. *Processing Data in Business, Education, and Government.* Somerville, N.J.: EBTA, 1972.

Miller, Boulton B. *Computers: A User's Introduction.* Edwardsville, Ill. Bainbridge, Inc., 1974.

Schnake, M. A. *Data Processing Concepts.* New York: McGraw-Hill Book Company, 1973.

State of California Employment Development Department. *Career Guides for Entry Occupations in Data Processing* Sacramento, Cal.. State of California Emploxment Development Department, 1974.

Textbooks

Awad, Elias M. *Business Data Processing.* Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1971.

Crawford, F R *Introduction to Data Processing,* 2nd ed. Englewood Cliffs, N.J.. Prentice-Hall, Inc., 1973.

International Business Machines Corporation. *Introduction to IBM Data Processing Systems.* White Plains, N.Y.: IBM, 1969.

——. *Number Systems.* White Plains, N.Y.: IBM, 1968.

Marxer, Ellen *Elements of Data Processing.* Albany, N.Y.. Delmar Publishers by Litten Educational Publishing, Inc., 1971.

Shelly, Gary B and Cashman, Thomas J. *Introduction to Flowcharting and Computer Programming Logic.* Fullerton, Cal.: Anaheim Publishing Company, 1972.

Wanous, E F; Wagner, Gerald E.; and Wanous, S. J. *Fundamentals of Data Processing.* New Rochelle, N.Y.: South-Western Publishing Company, 1971.

Supplementary

U S Government Employment Service. *Dictionary of Occupational Titles — 1974-75.* Washington, D.C.. Government Printing Office, 1974.

UNIT 3. COMPUTER OPERATIONS I

This unit is designed to give the student knowledge of the operation of a computer and its hardware, including the card read punch, the line printer, the console panel, the console typewriter, the magnetic disk drives, and magnetic tape drives. Hands-on training will be provided.

At the completion of the unit COMPUTER OPERATIONS I, the student should be able to:

- Make operational the input and output devices
- Explain the function of keys and indicators on the input and output devices
- Perform the operational responsibilities required with the use of the card reader and printer
- Recognize and explain the use of different types of card readers and printers
- Operate disk drives and perform preventive maintenance
- Explain disk storage concepts
- Operate tape drives and perform preventive maintenance
- Explain tape storage concepts
- Recognize and explain the use of other external storage devices
- Explain the concept of stored programs and data files on external devices

UNIT 3. COMPUTER OPERATIONS I

The student should be able to make operational the input and output devices.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Describe the procedures required to:

ready the card reader/punch
ready the line printer
ready the disk drive

Perform the correct procedures to power on the system and to:

ready the card reader/punch
ready the line printer
ready the disk drive
respond to the console typewriter messages

UNIT 3. COMPUTER OPERATIONS I

The student should be able to explain the function of keys and indicators on input and output devices.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Describe the card read/punch path.

Draw a schematic of the card read/punch showing the card feed path.

Explain the function of the card read/punch indicators.

Write the function of the card read/punch indicator lights and keys.

Explain and use the card read/punch operational keys.

Write or explain orally in detail the function of each operator control key on the card read/punch.

Explain the function of the indicators and operational keys on the line printer.

Write the function of the line printer indicator lights and operational keys.

Define the functional operation of the parts of the line printer.

Write or explain orally in detail the function of each light and manually operated part on the line printer.

Operate the line printer to change and align paper, using all previously learned operating parts.

Mount the paper and carriage control tape on the line printer.

Explain the purpose of the indicators and operator keys on the CPU console panel and console typewriter.

Operate the computer by using the CPU console panel and console typewriter to power on and power off the computer.

Explain the function of the disk drive indicatoru, and use the operational keys.

Identify and explain orally or in writing the funetion of the disk drive indicators and keys.

UNIT 3. COMPUTER OPERATIONS I

The student should be able to perform the operational responsibilities required with the card reader and printer.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Remove card jams from the card read/punch.

Perform the correct procedures to remove a card jam in the card read/punch.

Design, punch, and mount a carriage control tape on the line printer.

Punch a carriage control tape from a predefined set of instructions.

Mount a carriage control tape on the line printer.

Apply the techniques to change and realign a different size form (paper) on the line printer.

Make machine adjustments to change and realign forms (paper) on the line printer.

Insert a ribbon on the line printer.

Remove the old ribbon, and insert the new ribbon on the line printer.

UNIT 3. COMPUTER OPERATIONS I

The student should be able to recognize and explain the use of different types of card readers and printers.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Distinguish between brush concepts and photoelectric concepts in punched card reading.

Describe the major differences between the two types of card reading devices.

Explain the advantages and limitations of different types of line printers.

Given a list of five types of printer devices, describe one outstanding feature of each.

UNIT 3. COMPUTER OPERATIONS I

The student should be able to operate disk drives and perform preventive maintenance.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the procedures to clean and maintain the disk pack.

Clean the disk pack cover and change the dust filter from the bottom of a disk pack.

Identify the procedures to mount and demount disk packs.

Mount a disk pack, and ready the disk drive.

UNIT 3. COMPUTER OPERATIONS I

The student should be able to explain disk storage concepts.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Describe the function of the command verbs read, write, and seek.

Given a list of verbs and a list of definitive statements, match them.

Explain the physical organization of a disk file.

Draw a schematic showing the tracks, sectors, and cylinder organization on a disk.

List advantages of disk files over tape files.

Draw a series of tape records; compare them with corresponding disk records, and show the advantage of using disk records.

Recognize the characteristics of different types of disk devices.

Describe the differences between various types of disk devices.

Culminating Activity:

Given a list of words and definitions, match the words with the definition. (Appendix C-1)

UNIT 3. COMPUTER OPERATIONS I

The student should be able to operate tape drives and perform preventive maintenance.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain the function of the tape drive indicators and keys.

Match the description with the name of the various indicator lights and keys on the magnetic tape drive.

Explain the procedures to clean tape drives.

Perform the cleaning routine on a tape drive.

Explain the procedures to mount and demount tapes.

Perform the necessary steps to mount a tape and ready the tape drive.

UNIT 3. COMPUTER OPERATIONS I

The student should be able to explain tape storage concepts.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Identify data representations as they appear on magnetic tape.

Draw an illustration to show numeric and alphabetic characters on a 7- and a 9-track magnetic tape.

List advantages and limitations of tape files over card files.

Given a card layout and a problem involving a series of 500 cards, indicate the amount of space that would be required to store the 500 records of information on a tape file.

Recognize the distinctive characteristics of different types of tape devices.

Match the descriptive statements of advantages and limitations with the different magnetic tape devices. (Appendix C-2)

UNIT 3. COMPUTER OPERATIONS I

The student should be able to recognize and explain the use of other external storage devices.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Explain the characteristics and use of magnetic drum storage.

Draw a diagram showing the physical characteristics of a magnetic drum.

Explain the characteristics and use of a data cell.

Draw a diagram showing the physical characteristics of a data cell.

Culminating Activity:

Given a list of statements, fill in the blank that completes the statement. (Appendix C-3)

UNIT 3. COMPUTER OPERATIONS I

The student should be able to explain the concept of stored programs and data files on external devices.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain the difference between manufacturer-supplied and user-written programs.

- Match a list of manufacturer- and user-written programs with a descriptive statement. (Appendix C-4)

Explain the organization and operation of the System Residence on a direct access storage disk device.

- Draw a schematic showing the relationship between the Disk Residence System Pack and the Operating System as stored in the CPU.

Teacher's Notes:

UNIT 3. OUTLINE	REFERENCE
I. Review	
A. All forms of input devices	Awad
B. All forms of output devices	
C. CPU	
1. Divisions of CPU	
2. Concepts of stored memory	
II. Cycle One Content	
A. Card reader, punch	
1. Control keys and indicators	
a) Power	
b) Start	
c) Ready	
2. Card handling review	
B. Line printer	
1. Ready printer.	
a) Close print-unit gate.	
b) Mount and restore carriage control tape.	
2. Control keys and indicators	
a) Power	
b) Start	
c). Print ready	
C. CPU console panel	
How to enable the system — disk power ON and disk light ON	
D. Console typewriter	
How to interact to message(s) on the console typewriter	
E. Disk drive	
Disk mounted, powered ON and disk light ON	

III. Cycle Two Content

A. Card reader/punch — Fuori, et al.

1. Card path — name and function of each part the card passes
2. Function of each control panel indicator light and key
   a) End of file
   b) Validity check
   c) Read check
   d) Feed stop
   e) Stacker
   f) Fuse
   g) Transport
   h) Punch check
   i) Chip box
   j) Stop key
   k) End of file key
3. How to respond to indicators and stop conditions a)-k) above

B. Line printer

1. Name and function of each manually operated part
2. Function of each control panel indicator light and key
   a) End of form
   b) Form check
   c) Print check
   d) Sync check
   e) Single cycle key
   f) Carriage space key
   g) Cover lower key
   h) Check reset key
   i) Stop key
   j) Cover raise key
   k) Carriage restore key
   l) Carriage stop key
3. Form insertion and alignment
4. Operation of each manually controlled part
5. Respond to each possible print stop condition

C. CPU console panel — Advanced Systems, Inc.

Power ON and OFF procedures

D. Console typewriter

Purpose and operation of keys and switches

E. Disk drive

1. Name and function of operating parts
2. Name and function of indicator lights and keys
3. How to respond to stop conditions

UNIT 3. OUTLINE	REFERENCE
IV. Cycle Three Content	Fuori, et al.
A. Batch processing procedure	
B. Card read/punch	
1. Removing card jams	
2. Restart procedures	
C. Line printer	
1. Plan, punch, and mount a carriage control tape.	
2. Change and align different size and thickness of paper.	
3. Remove and reinsert the line-printer ribbon.	
V. Cycle Four Content	Fuori, et al.
A. Card read/punch	
1. Different types of card readers	
a) Brush VS photoelectric reading	
b) Serial VS parallel reading	
B. Line printers	
1. Speed, advantages, and disadvantages for different types of impact printers	
a) Print wheel	
b) Wire matrix	
c) Ball (console typewriter)	
d) Chain	
e) Drum	
2. Nonimpact printers	
VI. Cycle Five Content	
A. Disk operating and handling procedures	Advanced Systems, Inc.
1. Cleaning	
2. Mounting	
3. Demounting	
4. External labels	
B. Magnetic disk medium	Fuori, et al.
1. Character representation	
2. Functions	
a) Read	
b) Write	
c) Seek	
d) Search	

3. Disk files
   a) Advantages of disk files over tape files
   b) Limitations
4. Disk organization
   a) Tracks
   b) Cylinders
   c) Sectors
5. File organization
   a) Sequential
   b) Indexed sequential
   c) Random access
6. Characteristics of different types of magnetic disk devices
   a) Removable VS nonremovable disk packs
   b) Single VS multilevel disk packs
   c) Storage capacity
   d) Speed

VII. Cycle Six Content — Advanced Systems, Inc.

A. Magnetic tape drive

1. Name and function of each tape drive operative part
2. Function of each indicator light and key
3. Types of magnetic tape drives
   a) Operating differences
   b) Advantages and disadvantages
4. Tape drive operation and handling procedures
   a) Cleaning
   b) Mounting
   c) Demounting
   d) External labels

B. Magnetic tape — IBM

1. Character representation — IBM
2. Tape files
   a) Advantages of tape files over card files
   b) Limitations
3. Type organization
   a) Blocking factor
   b) Density
   c) Header labels
   d) Trailer labels
4. Storage capacity
5. Reflective indicators
6. File protection
7. Speeds, advantages, and disadvantages for different types of magnetic tape devices
   a) Manual loading
   b) Self-loading
   c) Tape cartridge

VIII. Other Storage Devices	Fuori, et al.
A. Magnetic drum	
B. Data cell	
C. Microforms	
1. Microfilm 2. Microfiche	*Infosystems Magazine*, April 1975
IX. Central Processing Unit and Auxiliary Storage	
A. Types of stored programs	
1. Manufacturer supplied a) Compiler, assemblers, translators b) Supervisor program c) Utility programs d) Library programs 2. User written application programs	
B. SYSRES (System Residence)	Fuori, et al.
1. IPL (Initial Program Load) 2. Supervisor 3. Job control	
C. Auxiliary storage	
1. Disk pack 2. Drum 3. Tape	
D. Relate CPU to SYSRES	

RESOURCES FOR UNIT 3. COMPUTER OPERATIONS I

Teacher References

Miller, Boulton B. *Computers. A User's Introduction.* Edwardsville, Ill.. Bainbridge, Inc., 1974.

Schnake, M. A. *Data Processing Concepts.* New York: McGraw-Hill Book Company, 1973.

Textbooks

Awad, Elias M. *Business Data Processing.* Englewood Cliffs, N.J.: Prentice-Hall, Inc.; 1971.

Fuori, William M., D'Arco, Anthony, and Orilia, Lawrence. *Introduction to Computer Operations.* New York: McGraw-Hill Book Company, 1973.

International Business Machines Corporation. *System/360 Disk Operating System Operation Training Manual, Student Text.* White Plains, N.Y.: IBM 1969.

Supplementary

Advanced Systems Incorporated. *Computer Operator Training — Video Assisted Instruction, Student Guide.* Elk Grove Village, Ill.: Advanced Systems Inc., 1972.

Periodical

"The Microfilm Picture 1975." *Infosystems,* April 1975, pp. 38-41.

UNIT 4. COMPUTER OPERATIONS II

This unit is designed to give the student an understanding of the internal operation of the central processing unit. The student will interact with the computer by means of a console typewriter and panel controls in a batch processing environment. Concepts involving teleprocessing, operating systems, security measures, and system updating will be presented.

At the completion of the unit on COMPUTER OPERATIONS II, the student should be able to.

- Explain and use the central processing control panel
- Use the console keyboard as an interacting device with the central processor
- Describe storage concepts and the flow of data into and from the central processor
- Follow a prescribed schedule, and document the activities of the computer system
- Operate the computer in a batch processing environment
- Operate the computer in a multiprogramming environment
- Explain the use of the computer in a telecommunications environment
- Describe the operating system organization
- Use security measures and backup procedures

UNIT 4. COMPUTER OPERATIONS II

The student should be able to explain and use the central processing control panel.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain the meaning and function of the various sections of the operator control panel.

Given a CPU console diagram, label the three major areas.

Explain the function of the operator dials, keys, lights, and switches on the operator intervention section of the CPU console.

Given a console diagram, indicate dial settings necessary to activate the card reader.

Explain the meaning and function of the customer engineer section.

Given a diagram of the customer engineer section of the CPU console, label the four switches and their settings.

Identify and explain the function of the two meters on the CPU console.

Display and read internally stored data.

Given a CPU console panel diagram, interpret the meaning of a series of exercises depicting the status of the CPU. (Appendix D-1)

UNIT 4. COMPUTER OPERATIONS II

The student should be able to use the console keyboard as an interacting device with the central processor.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain the purpose of indicators and keys.

Given a diagram of the keyboard, label the command keys and indicators and explain their function.

Identify the steps to perform the SYSTEM GENERATION (power ON procedures).

Perform an INITIAL PROGRAM LOAD on a computer system or list the sequential steps involved. (Appendix D-2)

List and interpret the most frequently used system-to-operator messages.

Given frequently used system-to-operator messages, interpret their meanings.

List and describe the operator response to the frequently used system messages.

Given frequently used system-to-operator messages, select the appropriate operator responses.

UNIT 4. COMPUTER OPERATIONS II

The student should be able to describe storage concepts and the flow of data into and from the central processor.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain the use of multiplexor and selector channels.

Given a computer system diagram, label the main parts of the computer, channels, and I/O devices. (Appendix D-3)

Translate the codes for storing data.

Given a series of exercises involving data codes, convert the code to the appropriate response.

Explain the use of registers to manipulate data, perform calculations, and make comparison tests.

List and explain the three types of operations that occur within a general purpose register.

Explain the use of registers by the processing supervisor to maintain address locations.

Given base and displacement for several fields, determine the effective address for each.

List and describe the various CPU program states.

Given a diagram, label the program states. (Appendix D-4)

Identify system interrupts, and recognize the cause of each interrupt.

Given a series of system interrupts, match the class of interrupt to its cause.

Recognize the meaning and use of the program status words (PSW).

Write the names of the program status words (PSW) and their usage.

Recognize the meaning and use of the channel status words (CSW).

Explain the use of the channel status words (CSW) as they relate to the total computer system.

UNIT 4. COMPUTER OPERATIONS II

The student should be able to follow a prescribed schedule, and document the activities of the computer system.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Use the computer room checklist and supplies list to prepare for the day's operations.

Given a computer room checklist, perform the tasks indicated. (Appendix D-5)

Follow the activities on the daily run schedule and system run sheets.

Given a series of jobs and a starting time, complete a daily schedule sheet.

Analyze the system-generated logs.

Given a computer usage log, determine low and high points of computer usage.

Document each job.

Gather all data generated by the system which pertains to each job and forward it to the control department. (Appendix D-6)

UNIT 4. COMPUTER OPERATIONS II

The student should be able to operate the computer in a batch processing environment.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain the procedure for setting up a job for batch processing.

Given a job run sheet, set up and run the job on the computer.

Explain the proper placement of job control cards in a batch processing environment.

Given a job task and a list of control cards, arrange the cards in proper sequence.

Identify diagnostic messages from the system and select an appropriate response.

Given a series of diagnostic messages from the system and a list of responses, match them. (Appendix D-7)

UNIT 4. COMPUTER OPERATIONS II

The student should be able to operate the computer in a multiprogramming environment.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Distinguish between batch mode and single program initiation (SPI) mode.

Given a diagram, show the relative location of the system control programs and the partitions.

State the limitation, location, and priority of partitions in storage allocation.

Given a diagram label, indicate priority of the partitions and state the amount of storage required.

Compare the processing of programs in batch mode and single program initiation (SPI) mode.

Given a series of jobs, choose the method of processing (batch or SPI), and indicate the reasons for the choice.

Use standard procedures to establish partitions, assign peripheral addresses, and react to other needs of the system.

Given a series of exercises, select the proper operator responses necessary to compile the exercises.

UNIT 4. COMPUTER OPERATIONS II

The student should be able to explain the use of the computer in a telecommunications environment.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Describe the main elements of a telecommunications system.

List the main elements of a telecommunications system.

Recognize different types of data transmittal devices.

Given a diagram of terminal devices in a telecommunications environment, label the devices.

List the advantages and limitations of a telecommunications system.

List and explain five advantages and three limitations of a telecommunications system.

Explain the concepts of multiprogramming and multiprocessing with respect to telecommunications.

Draw a schematic of a telecommunications system, showing both multiprogramming and multiprocessing possibilities. (Appendix D-8)

UNIT 4. COMPUTER OPERATIONS II

The student should be able to describe the operating system organization.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain the major organization of a disk operating system.

List the types of programs that are stored on a disk operating system.

Describe the types of control programs that are stored on the system residence (SYSRES).

List the types of control programs that are stored on the system residence (SYSRES).

Describe the processing programs that are stored on the SYSRES.

List the types of processing programs that are stored on the SYSRES.

Explain the function of libraries.

Given a list of library programs, explain the function of each.

Explain the procedure for storing user-written programs and data files.

Describe the process used in storing a user-written program in the private library.

On a schematic, draw and label the sections of a system residence and indicate the main purpose of each section. (Appendix D-9)

UNIT 4. COMPUTER OPERATIONS II

The student should be able to use security measures and backup procedures.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

State the procedures for maintaining personnel security.

List some of the procedures for maintaining personnel security in a computer center.

Identify the factors that are needed to make a computer facility physically secure.

Describe some ways for providing adequate physical security in a computer center.

Identify the methods of data security in a computer environment.

List several methods for improving data security in a computer center.

Describe the use of utility programs to produce system and data backup files.

Given a utility program, copy the SYSRES on a new disk pack (or tape, or cards).

Explore the essential elements of formal contracts and mutual agreements as they pertain to system failure.

List several advantages of mutual agreements.

Explain the job tasks necessary to prepare backup data files

List the sequential steps to produce data backup files.

UNIT 4. OUTLINE	REFERENCE
I. CPU Console Panel	Fuori, et al.
A. Operator control panel	
1. Determine and control distribution of power.	
2. Check status of the computer system (system status).	
3. Communicate with the computer.	
a) Enter data.	
b) Enter device numbers.	
4. Meaning, function, and operation of:	
a) Emergency pull switch and use meters	
b) Push button keys	
c) Display lights	
d) Rotary switches	
B. Operator intervention section	
1. Meaning, function, and operation of the data and address entry switches	
2. Meaning, function, and operation of the pushbuttons	

C. Customer engineer section

D. Display and storage registers

1. Main storage data register display
2. Main storage address register display
3. Interpret register display (lights)

II. Console Keyboard — IBM

A. Purpose and function of

1. Keyboard and switches
2. Keys and indicator lights

B. Operating procedures

1. Supply power to the system.
2. Ready the card reader.
3. Ready the disk drive.
4. Ready the printer.
5. Ready the tape drive.
6. Ready the console keyboard.
7. Initial program load.

C. Console communication

1. Console commands used by the operator
2. Operator reaction to console messages and codes
   a) Program diagnostic messages
   b) Equipment diagnostic messages
   c) Use of operators' system manuals (for the system in use)

III. Central Processing Unit Logic Flow — IBM, Fuori, et al.

A. Device control unit

B. Standard interface

C. Channels

1. Byte mode transmission — selector channel
2. Burst mode transmission — multiplexor channel

D. Main storage

1. Control section — operating system
   a) Function
   b) Types
      (1) BOS — Basic
      (2) TOS — Tape
      (3) DOS — Disk
      (4) OS — Full

2. Arithmetic logic unit
   a) Fixed point operations
   b) Variable point operations
   c) Floating point operations
3. Storage concepts
   a) Half word
   b) Full word
   c) Double word
   d) Data codes in core
      (1) Extended binary coded decimal interchange code (EBCDIC)
      (2) Hexadecimal
      (3) Zoned decimal
      (4) Packed decimal
   e) Registers
      (1) Base
      (2) General
      (3) Index

E. CPU program status — IBM

1. Program states
   a) Stopped-operating
   b) Running-waiting
   c) Supervisor-problem
   d) Marked-interruptible
2. Classes of interrupts
   a) I/O
   b) External
   c) Program check
   d) Machine check
   e) Supervisor call
3. Program status word
   a) Current PSW
   b) New PSW
   c) Old PSW
4. Channel status
   a) Channel command word
   b) Channel status word

IV. The Operator and the Job Stream — Fuori, et al.

A. Initializing the system

1. Computer room checklist
2. IPL log
3. Supplies list

B. Scheduling jobs — I/O control group (IOCG)

1. Daily-run job schedules
   a) Off-line peripherals
   b) Clerical and special
   c) System run
2. Job transmittal slip (work route sheet)

C. Documenting the system activities

1. Computer usage log
2. Error message log
3. Tape disk history log

D. Documenting the results of computer processing

1. External disk and tape labels
2. Completed output
   a) Printed output
   b) Card output
   c) New updated disks tapes

V. Operating in a Batch Environment — Fuori, et al.

A. Job run set up

1. Computer room checklists
2. Operator and job stream

B. Operator activities

1. Placement of JCL, source, and data cards in run deck
2. Recovery from invalid data
3. Detecting a continuous program loop
4. Responding to diagnostic messages
   a) System
   b) Program
5. Cancelling operations
   a) Cancelling a dump
   b) Cancel and reinitialize

VI. Operating in a Multiprogramming Environment — IBM, Miller

A. Review the control program functions

1. Task management (IPL)
2. Job management (Supervisor)
3. Data management (JCL)

B. Storage allocations

1. Single partition
2. Multiprogramming
   a) Background
   b) Foreground

C. Multiprogramming capabilities

1. Running in batch mode and/or single program initiation (SPI)
2. Partition allocation
3. Partition priorities

D. Operator (attention routine) commands

1. START
2. ALLOC
3. ASSGN
4. CANCEL
5. STOP
6. READ
7. LISTIO
8. MAP

VII. Telecommunications (Teleprocessing, Time Sharing) — Fuori, et al. IBM Miller Awad

A. Elements of telecommunicating system

1. Terminals (I/O devices)
2. Lines of communication
3. Controls
4. Computer hardware

B. Types of telecommunications terminals

1. Punched paper tape
2. Punched cards
3. Magnetic tape
4. Keyboard
5. Computer
6. Display station (CRT)
7. Analog computer
8. Audio response unit (output)

C. Types of data transmission lines

1. Telephone
2. Microwave
3. Cable

D. Concepts of operation

1. Multiprogramming
2. Multiprocessing

VIII. Operating System Concepts — Fuori, et al. IBM Miller

A. Disk system residence organization

1. Control programs
   a) Supervisor
   b) Job control
   c) IPL
2. Processing programs
   a) Language translators
   b) Service programs
      (1) Sort merge
      (2) Utility programs
   c) Linkage editor
3. Libraries
   a) Function
   b) Types
      (1) Core image
      (2) System linkage
      (3) Relocatable
      (4) Program (private)

B. User application programs

1. Payroll
2. Inventory
3. Other

C. Updating the operating system

1. Updating existing system
2. Generating a new system

IX. Security Measures — IBM

A. Personnel security

1. Screening for selection of employees
2. Screening of visitors to computer area
3. Authorized personnel in computer area
4. Separation of responsibilities in the area of
   a) Data control
   b) Systems analysis
   c) Programming
   d) Data entry
   e) Computer operations

B. Computer center and equipment security

1. Computer center location
2. Change from glass-enclosed showcase to fortress-like structures
3. Use of lead installed walls to prevent electronic espionage
4. Identification to enter computer room
   a) Photograph of employee
   b) Magnetic encoded strips
   c) Key cipher lock
   d) Double locked entry and exit with second person present
5. Fire detection systems, safeguards, and emergency procedures
6. Location of data control area
7. Electrical power

C. Data files security

1. Signout of data files
2. Use of internal codes on tape and disk files
3. Use of keyword and password to access on-line data files
4. Catalogs of eligible users
5. Scrambled data fields
6. Security over making changes to files
7. Use of a datacoder

X. Operational Backup Procedures

A. Purpose

B. Methods

RESOURCES FOR UNIT 4. COMPUTER OPERATIONS II

Teacher References

International Business Machines Corporation. *Introduction to IBM Direct-Access Storage Devices and Organization Methods*. White Plains, N.Y.: IBM, 1974.

International Business Machines Corporation. *The Considerations of Physical Security in a Computer Environment*. White Plains, N.Y.: IBM, 1972.

Miller, Boulton B. *Computers. A User's Introduction*. Edwardsville, Ill.. Bainbridge, Inc., 1974.

Schnake, M. A. *Data Processing Concepts*. New York. McGraw-Hill Book Company, 1973.

Textbooks

Awad, Elias M. *Business Data Processing*, 3rd ed. Englewood Cliffs, N.J.. Prentice-Hall, Inc., 1971.

Fuori, William M., D'Arco, Anthony, and Orilia, Lawrence. *Introduction to Computer Operations*. New York: McGraw-Hill Book Company, 1973.

International Business Machines Corporation. *Introduction to System/360 Text Programmed Instruction*. Poughkeepsie, N.Y.: IBM, 1970.

UNIT 5. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

This unit provides an introduction to the RPG vocabulary, the tasks which a programmer performs, the programmer's tools used in the performance of these tasks, and an opportunity to code instructions for business programs using the RPG programming language. The student will be required to plan, organize, code, punch, test, correct, and document business applications.

At the completion of the unit INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING, the student will be able to:

- Recognize the programming cycle and the sequence and the function of each part
- Write and document a program to read a card and to print a line
- Write and document a program to manipulate and edit numeric data and totals
- Identify and code a program with headings on each page of a report
- Write and document a program to perform multiplication and division operations and to print the result of the calculation

UNIT 5. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

The student should be able to recognize the programming cycle and the sequence and the function of each segment.

PERFORMANCE OBJECTIVES	and ASSESSMENT MEASURES
Identify and show the relationship of the seven parts of a computer-based system.	
	List the seven parts of a computer-based system.
	Arrange the seven parts of a computer-based system on a given flowchart in the correct order. (Appendix E-1)
Identify the nine segments of the programming cycle.	
	Given the nine segments of the programming cycle, arrange the segments in the correct sequence.
Identify the parts of a Printer Spacing Chart, including the carriage control tape.	
	Given a Printer Spacing Chart and a problem involving two printed columns, draw a Printer Spacing Chart.
Recognize the purpose and the parts of a card layout.	
	Given the location of three or more data fields and a card layout form, draw the card layout.

Identify the use of a flowchart and a decision table as problem-solving tools.

List and explain two problem-solving tools used in programming.

Identify the elements of a narrative.

Given a narrative, underscore and label the elements of a narrative.

Identify the relationship of the Programming Cycle elements to the coding of the program.

Given a list of the Programming Cycle elements, describe the use of each element in coding a program.

Identify the relationship between Job Control Language, Source Program, and Test Data.

Explain the purpose of each group of cards in the Run Deck.

Recognize three types of program testing and the purpose of each.

Explain the purpose of each type of program testing.

Recognize the use of Diagnostic Messages in debugging a computer program.

Given a common Diagnostic Message, explain the use and the programmer's reaction.

Identify the items to be included in program documentation.

List the items to be included in program documentation.

Culminating Activity:

Outline in sequence the nine steps in the Programming Cycle with a brief explanation of the purpose of each step. (Appendix E-2)

UNIT 5. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

The student should be able to write and document a program to read a card and to print a line.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

State the purpose of the RPG language.

List the advantages and limitations of the RPG language.

Identify the seven RPG Specifications forms.

Define the purpose of each RPG Specifications form.

Identify the essential elements of each RPG form required to run an elementary program.

Given a set of RPG coding forms, circle the essential elements on each form needed to run an elementary program.

Analyze the program statement and write the problem specification.

Write the problem specification. (Appendix E-3)

Develop the flowchart to show the solution of the problem.

Draw the flowchart for the problem.

Create the narrative to explain the problem solution.

Write a narrative for the problem.

Translate a problem statement, specification, and flowchart into a logical coded program.

Code, test, and debug the program.

Prepare the necessary forms to fully document the program.

Document the application.

UNIT 5. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

The student should be able to write and document a program to manipulate and edit numeric data and totals.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain and use editing with data.

Given a series of exercises with data to be edited, write the coding to place edit symbols in the numeric fields.

Identify the entries required to perform an *addition* calculation.

Given a series of *addition* exercises, write the coding to add the fields.

Identify the entries required to perform a *subtraction* calculation.

Given a series of *subtraction* exercises, write the statements to perform the *subtraction* operations.

Construct the entries required to print the result of a calculation as a total, as a final total, and as an edit word.

Given the previous addition and subtraction exercise statements, write the coding instructions to print the result of the calculation: (1) in an edit word in a Detail Line, 2) in a Total Line, and 3) in a Final Total Line.

Develop a problem program using the problem specification, flowchart, and narrative.

Given a programming assignment with alphanumeric and numeric edited data and calculation (+, -), write a program. (Appendix E-4)

Compile, successfully execute, and document the program.

Provide supportive documentation for the successfully executed program.

UNIT 5. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

The student should be able to identify and code a program with headings on each page of a report.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the entries required to code headings in a program.

Circle the essential entries needed for coding Heading Lines upon an RPG Output Specification form.

Identify the entries required to control the printing of Headings.

List the entries required to print Heading Lines on more than one page.

Classify the order in which Headings, Detail Lines, and Total Lines should appear in the Output Specifications.

Explain the order in which the various output lines are to be arranged.

Alter a program to print Heading Lines.

Using a previously correctly executed program, Printer Spacing Chart, and RPG coding forms, plan and code the necessary instructions to print Heading Lines for the report.

Sample Criterion:

Main heading
Edited date
Column headings

Develop a problem program using the problem specifications, flowchart, and narrative.

Write a program to include Heading Lines and a Calculation.

Compile, successfully execute, and document the program.

Test and debug the program with Heading Lines. Upon successful completion, organize the supportive documentation.

UNIT 5. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

The student should be able to write and document a program to perform multiplication and division operations and print the result of the calculation.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the placement of the factors in a multiplication instruction.

Label the entries in a series of multiplication instructions.

Define the Field Length and Decimal Positions for the multiplication instruction.

Given a series of multipliers and multiplicands, determine the Field Length and Decimal Positions for each Product.

Explain Rounding of the Result Field.

Given an exercise, write instructions to multiply, to define Factor and Result Fields, to determine Field Length, Decimal Placement, and to Half Adjust.

Identify the placement of the factors in a division instruction.

List the entries required in a single division instruction.

Define the Field Length and the Decimal Positions for a division instruction.

Write sample instructions to divide and determine the Field Length and the Decimal Positions.

Explain the operation used to save the Remainder of a division calculation for future use.

Write sample instructions to divide and to save the Remainder for future use.

Develop a problem program using the problem specifications, flowchart, and narrative.

Write a program to perform calculations including multiplication, division, and rounding.

Compile, successfully execute, and document the program.

Test and debug the program with multiplication and division. Upon successful completion of the program, organize the documentation.

UNIT 5. OUTLINE	REFERENCE
I. Introduction to Computer Programming	IBM
A: Review computer based system.	
1. Planning	
2. Collection of raw data	
3. Encoding	
4. Input	
5. Manipulation	
6. Decoding	
7. Output	
B. Hardware software	
C. Data processing terminology	
II. Review of the Programming Cycle	
A. Read the problem statement.	
B. Organize facts, definitions, assumptions to design:	
1. Printer spacing chart carriage control or output record layout	
2. Input record layout	
C. Develop problem solution through:	Shelly and Cashman National Cash Register Co.
1. Flowchart(s)	
a) Systems	
b) Program	
2. Decision table	
D. Write the narrative to include the formal:	
1. Problem specifications	
2. Explanation of application	
a) Logic for solution	
b) Equipment	
c) Computer language	
d) Source of test data	
E. Code the source program.	Shelly and Cashman
1. Input	
2. Process	
3. Output	

	REFERENCE
F. Punch:	
1. Source program	
2. Test data	
3. Job control language	
G. Test the program.	Cashman and Fletcher
1. Desk-check the program.	
2. Compile the program.	
3. Execute the program.	IBM
H. Debug the program diagnostic errors.	Shelly and Cashman
I. Organize the documentation for the application.	
III. Introduction to the RPG Language	
A. Definition	Cashman and Fletcher
B. Advantages	
C. Limitations	
D. RPG cycle	
1. Get	
2. Process	
3. Put	
IV. Introduction to RPG Specification Forms	
A. Define the purpose of each form:	
1. Control card/specification	
2. File description specification	
3. Line definition	
4. Input definition	
5. Calculations definition	
6. Output definition	Shelly and Cashman
7. File extension definition	
B. Essential RPG specifications for programming	Feingold
1. File description/control card	
2. Input	Shelly and Cashman
3. Output	

V. RPG Specification Entries for Detail Printing

A. File description specification control card

1. Common fields
   a) Page number
   b) Line number
   c) Form type
   d) Program
   e) Comment
   f) Other
2. Data files

B. Input specification

1. Record identification
2. Field description
3. Common fields

C. Output specification

1. File identification and control
   a) File name
   b) File type detail
   c) Vertical spacing forms control
2. Field description
   a) Output indicators
   b) Field name
   c) End position
3. Common fields

D. Business application problem and documentation

VI. Implementing the Programming Cycle

A. Read the problem statement.

B. Organize facts, definitions, assumptions to design.
1. Printer spacing chart carriage control or output record layout
2. Input record layout

C. Develop problem solution through

1. Flowchart(s)
   a) Systems
   b) Program
2. Decision table

D. Write the narrative to include:

1. Problem statement
2. Explanation of application
   a) Logic for solution
   b) Equipment
   c) Computer language
   d) Source of test data

E. Code the source program:

1. Input
2. Process
3. Output

F. Punch:

1. Source program
2. Test data
3. Job control language

G. Test the program.

1. Desk-check the program.
2. Compile the program.
3. Execute the program.

H. Debug the program diagnostic errors.

I. Organize the documentation for the application.

VII. Numerically Edited Data — Shelly and Cashman

A. Input data definition

B. Output editing techniques

1. Zero suppression
2. Special characters ($. b - 0, *)

C. Business application problem documentation

VIII. Data Manipulation and Final Total Printing — Cashman and Fletcher

A. Calculations specifications

1. Indicators
2. Factors
3. Mathematical operations (+, -) — Shelly and Cashman
4. Common fields

B. Output specifications

1. Total file type
2. Output indicators

C. Business application problem documentation

IX. Heading Control

A. Review the printer spacing chart

B. File description specification entry

C. Line definition specification

D. Output specifications entries

1. Channel skipping
2. Indicators
   a) Page 1
   b) Other
3. File type-head
4. Reserved work-page

E. Business application problem documentation

X. Calculations Operations — Shelly and Cashman

A. Multiplication

1. Calculation specification
   a) Factors
   b) Operation
   c) Field length
   d) Decimal positions
   e) Resulting field
   f) Rounding
   g) Indicators
2. Input, output specifications
3. Sample multiplication projects

B. Division

1. Calculation specification
   a) Factors
   b) Operation
   c) Resulting field
   d) Field length
   e) Decimal positions
   f) Rounding
   g) Store remainder
   h) Indicators
2. Input, output specifications
3. Sample division problems

C. Business application problem and documentation

RESOURCES FOR UNIT 5. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

Teacher References

Feingold, Carl *Introduction to Data Processing*. Dubuque, Iowa. Wm. C. Brown Company Publishers, 1971. o.p.

Textbooks

Cashman, Thomas J., and Fletcher, Dennis A. *IBM System/360 RPG Programming, Vol. I, Introduction*. Fullerton, Cal.: Anaheim Publishing Co., 1967.

IBM System/360 RPG Programming, Vol. II, Advanced Concepts. Fullerton, Cal.: Anaheim Publishing Company, 1967.

Feingold, Carl *Introduction to Data Processing*. Dubuque, Ia.. Wm. C. Brown Company Publishers, 1971. o.p.

International Business Machines. *A Data Processing Glossary*. White Plains, N.Y.. International Business Machines, 1972.

International Business Machines Corporation. *System/360 Report Program Generator Coding*. White Plains, N.Y.: IBM, 1969.

International Business Machines Corporation. *Systems. Introduction to RPG II*. White Plains, N.Y.. IBM, 1972.

International Business Machines. ystems Reference Library, IBM System/360 Basic Operating System Report Program Generator Specifications. White Plains, N.Y.: IBM, 1970.

Murray, Jerome T. *Programming in RPG II. IBM System/3*. New York. McGraw-Hill Book Company, 1971.

National Cash Register. *Introduction to Programming*. Dayton, O.. National Cash Register Company, 1969.

Shelly, Gary B., and Cashman, Thomas J. *Introduction to Flowcharting and Computer Programming Logic*. Fullerton, Cal.: Anaheim Publishing Company, 1972.

. *Introduction to Computer Programming RPG*. Fullerton, Cal.. Anaheim Publishing Company, 1972.

UNIT 6. INTERMEDIATE REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

The unit will provide the student with additional data processing vocabulary and techniques and experience in creating and using decision tables and flowcharts (programmer's tools). It will provide the student with information to make program decision statements, to vary the printed format of reports, to accumulate and print subtotals, and to store and use tables. The student will be required to plan, organize, code, punch, test, correct, and document business applications.

At the completion of the unit INTERMEDIATE REPORT PROGRAM GENERATOR (RPG) PROGRAMMING, the student will be able to write and document a program:

- To print and to process, depending upon the results of a test between two items
- With two multiple-indicator control instructions
- Requiring the use of a level indicator
- To print repetitive identifying information from the first record of each group
- To print one line of identifying information and the totals for each group of related records
- Using more than one control field
- To store a table in memory and to retrieve a field (Function) from the table

UNIT 6. INTERMEDIATE RPG

The student should be able to write a program to print and to process, depending upon the results of a test between two items.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the required entries for a compare (COMP) instruction.

Name the entries essential for a compare (COMP) instruction.

Explain the concept of testing.

Given a sample program, identify and explain the uses of the test instruction.

Relate the results of a test instruction to the use of Resulting Indicators.

Given a series of the test instructions including Resulting Indicators, explain the results of each test instruction.

Identify the possible Resulting Indicators when comparing two numeric fields of data.

State the Resulting Indicator entries necessary to show the results of the comparison of two numeric fields of data.

Identify the possible Resulting Indicators when comparing two alphameric fields of data.

State the Resulting Indicator entries necessary to show the results of a comparison of two alphameric fields of data.

Follow the general rules for comparisons to write sample instructions.

Given a problem statement involving comparisons, write the entries necessary to make the comparisons.

Develop a problem program with a comparison to control program processing.

Write, execute, and document a program to print and to process (depending upon the results of a comparison between two fields) using the problem specification, flowchart, and narrative.

Sample Criterion:

Print Headings on each report page.
Calculate, depending upon a decision instruction.
Print Detail Lines.
Print Totals on each report page. (Appendix F-1)

UNIT 6. INTERMEDIATE RPG

The student should be able to write and document a program with two multiple-indicator control instructions.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the use of more than one indicator to control an instruction.

Code sample multiple-indicator instructions for Input, Calculation, and Output Specifications.

Identify the relationship of indicators used on the various programming specification forms.

Given the indicators identified on the input specifications, show the relationship of the indicators on the calculation specifications and the output specifications.

Identify the relationship of indicators used in the program coding to the conditions developed in the logic flowchart or decision table.

Design a decision table, and relate the conditions on the decision table to the use of indicators in the program.

Develop a problem with at least two multiple-indicator instructions.

Write, execute, and document a program with at least two multiple-indicator control instructions, using the problem specification, flowchart, and narrative.

Sample Criterion:

Print Headings on each report page.
Calculate with two or more indicators.
Print Detail Lines with multiple-indicator control instructions.
Print Totals on each report page.

UNIT 6. INTERMEDIATE RPG

The student should be able to write and document a program requiring the use of a Level Indicator.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the use of Level Indicators.

Given Input, Calculation, and Output instructions, circle the Level Indicators.

Identify the entries required to use a Level Indicator.

Write sample Input, Calculation, and Output instructions using a Level Indicator.

Explain the use of the Blank After entry.

In a given series of Output instructions, circle the Blank After entry, and state the purpose of its use.

Develop a problem with Level Indicators.

Write, execute, and document a program requiring the use of a Level Indicator and Blank After in the Calculation and the Output Specifications, using the problem specification, flowchart, and narrative.

Sample Criterion:

Print Headings on each report page.
Calculate with one record indicator and one level indicator.
Print Detail Lines with one indicator.
Print Total Line with a level indicator.
Print Final Total.

UNIT 6. INTERMEDIATE RPG

The student should be able to write and document a program to print repetitive identifying information from the first record of each group.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the advantages of using Group Indication.

Given a problem statement, indicate why a portion of the report would be Group Indicated rather than detail listed.

Identify the entries required for Group Indication.

State what entries must be included in the Output instruction to Group Indicate a field.

Write sample instructions to Group Indicate on a report.

Alter a program to Group Indicate one field.

From a previously executed program with a level control, code the additional entries to Group Indicate the L1 control of a repetitive field on the Input and Output Specifications.

Develop a program to Group Indicate.

Write, execute, and document a program to Group Indicate at least one field, using the problem specification, flowchart, and narrative.

Sample Criterion:

Print the Headings on each report page.
Calculate with one level and one record indicator.
Print Detail Lines with one Group Indicated field.
Print Level and Final Totals.

UNIT 6. INTERMEDIATE RPG

The student should be able to write and document a program to print one line of identifying information and the totals for each group of related records.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the advantages of Group Printing.

Given a problem statement, indicate the reasons for Group Printing a report rather than Detail Printing.

Identify the entries required to Group Print.

Given a series of problem statements, write the Input and Output instructions required to Group Print.

Alter a program to Group Print.

With a previously executed program including a level control, code the instructions to double space and Group Print a report. (Appendix F-2)

Develop and document a program with Group Printing of identifying information and totals.

Given a problem statement, write, execute, and document a program to Group Print, identifying information and totals.

Sample Criterion:

Print Headings on each page.
Calculate with one record and one level indicator.
Group Print the Report.
Print Final Total.

UNIT 6. INTERMEDIATE RPG

The student should be able to write and document a program using more than one control field.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Recognize the placement of the control Level Indicators on the specifications form.

Given a sample program with more than one Level Indicator, circle the Level Indicators and explain their use.

Identify the entries required to use multiple Level Indicators.

Given a problem statement, write the necessary Input, Calculation, and Output instructions to use a series of Level Indicators.

Identify the priority of Level Indicators.

Arrange a group of Level Indicators in the order in which they will be processed by the computer.

Develop a program with more than one control field (Level Indicator) and document.

Given a problem statement, write, execute, and document a program using more than one control field (Level Indicator).

Sample Criterion:

Print Headings on each page.
Calculate with at least one record and more than one level indicator.
Print Detail Lines.
Print Total Lines with more than one Level Indicator.
Print Final Total. (optional)

UNIT 6. INTERMEDIATE RPG

The student should be able to write a program to store a table in memory and to retrieve a field (Function) from the table.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the entries required to read a table into memory.

Write the instructions required to store a two-dimensional table in memory.

Identify the entries required to use a table Function for calculation.

Write the instructions to retrieve a table Function and to use the Function in a calculation.

Identify the entries required to print data retrieved from a stored table.

Write the instructions to retrieve and to print a table Function.

Develop and document a program which stores and uses a table.

Write, execute, and document a program, given a problem statement to read a two-dimensional table into memory, to SEARCH the table for a corresponding data field (Function), to retrieve, and to print.

Sample Criterion:

Print Headings on each page.
Store a table containing the two-letter abbreviation for each of the 50 states and the corresponding name of each state.
Locate the state name.
Print and name and address, with the state name.

UNIT 6. OUTLINE	REFERENCE
I. Arithmetic and Logical Testing	Shelly and Cashman
A. Definition	
B. Planning tools	
1. Flowchart	
2. Decision table	Awad National Cash Register Co.
C. Calculation program entries	
1. Record indicators	
2. Factors	
3. Operation	
4. Resulting indicators	
a) Arithmetic	
b) Logical	
c) Other	
D. Comparison rules	
E. RPG program cycle	
F. Business application and documentation	Murray

UNIT 6. OUTLINE	REFERENCE
II. Multiple-Indicator Control	Shelly and Cashman
A. Definition	
B. Record indicator entries	
1. OR	
2. AND	
3. NOT	
C. Input specification entries	Cashman and Fletcher
D. Calculation specification entries	
E. Output specification entries	
F. Relationship of indicators in instructions	
G. Planning tools review	
1. Flowchart	
2. Decision table	
H. Business application and documentation	
III. Level Indicator Control	
A. Definition	
B. Planning tools	
C. Input specification entries	
D. Calculation specification entries	
E. Output specification entries	
F. RPG program cycle	
1. Moves	
2. Headings	
3. Detail line	
4. Read	
5. Last record	
6. Control fields	
7. Record indicator	
8. Calculations/resulting indicators	
9. Output	
G. Business application and documentation	Cashman and Fletcher Shelly and Cashman

UNIT 6. OUTLINE	REFERENCE
IV. Printing Control	
A. Detail printing review	
B. Group indication	
1. Definition 2. Usage 3. Input and output entries	Cashman and Fletcher
C. RPG program cycle	
D. Business application and documentation	
V. Group Printing	Shelly and Cashman
A. Definition	
B. Usage	
C. Entries	
1. Input 2. Output	
D. RPG program cycle	
E. Business application and documentation	Cashman and Fletcher
VI. Multi-Level Indicator Control	
A. Review printing	Shelly and Cashman
1. Detail print 2. Group print 3. RPG program cycle	
B. Group indication and group printing	
C. Business application and documentation	Cashman and Fletcher
VII. Tables	
A. Definition	
B. File description entries	Shelly and Cashman

C. File extension entries	
1. File name	
2. Table name	Cashman and Fletcher
3. Entries per record	
4. Entries per table	Shelly and Cashman
5. Entry length	
6. Decimal positions	
7. Sequence	
8. Alternate table name (Function)	
9. Entry length	
D. Calculation specification entries	
1. Factors	
2. Operation	
3. Result field	
4. Indicators	
E. Output specification entries	
1. Data from the table	
2. Data not in the table	
F. Job control language statements	
G. Business application and documentation	

RESOURCES FOR UNIT 6. INTERMEDIATE REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

Textbook

Awad, Elias M. *Business Data Processing*. Third edition. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1971.

Cashman, Thomas J., and Fletcher, Dennis A. *IBM System/360 RPG Programming, Vol. I, Introduction*. Fullerton, CA: Anaheim Publishing Co., 1967.

IBM System/360 RPG Programming, Vol. II, Advanced Concepts. Fullerton, CA: Anaheim Publishing Company, 1967.

International Business Machines Corporation. *System/360 Report Program Generator Coding*. White Plains, N.Y.: IBM, 1969.

International Business Machines Corporation. *Systems. Introduction to RPG II*. White Plains, N.Y.: IBM, 1972.

International Business Machines. *Systems Reference Library, IBM System/360 Basic Operating System Report Program Generator Specifications*. White Plains, N.Y.: IBM, 1970.

Murray, Jerome T. *Programming in RPG II: IBM System/3*. New York. McGraw-Hill Book Company, 1971.

National Cash Register. *Introduction to Programming*. Dayton, O.. National Cash Register Company, 1969.

Shelly, Gary B., and Cashman, Thomas J. *Introduction to Flowcharting and Computer Programming Logic*. Fullerton, Cal.: Anaheim Publishing Company, 1972.

. *Introduction to Computer Programming RPG*. Fullerton, Cal.. Anaheim Publishing Company, 1972.

UNIT 7. INTRODUCTION TO COBOL

This unit is designed to give the student a working knowledge of the business oriented language, COBOL. Students will write elementary programs, develop COBOL flowcharts, and learn the techniques of documentation.

At the completion of the unit INTRODUCTION TO COBOL, the student should be able to.

- Identify the structure of the COBOL language
- Identify and describe each of the COBOL divisions
- Develop a decision table from a set of instructions
- Create an appropriate flowchart from a job description
- Write the various sections of a COBOL program on a coding sheet
- Test the program
- Document the job
- Write and document a program with headings on each page of a report

UNIT 7. INTRODUCTION TO COBOL

The student should be able to identify the structure of the COBOL language.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Recall and describe six elements that are contained in the programming cycle.

List in order the parts of the programming cycle.

Recognize the advantages and limitations of the COBOL language.

Name at least one advantage and one disadvantage of the COBOL language.

Identify and use the mechanics of the language.

Given a list of arithmetic verbs and a group of definitions, match them. State at least two rules governing spacing in a program. State at least two rules governing punctuation in a program.

Recognize the use of reserved words and programmer-supplied words in the language.

State the difference between a reserved word and a programmer-supplied word and give an example of each.

UNIT 7. INTRODUCTION TO COBOL

The student should be able to identify and describe each of the COBOL divisions.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify and describe the Identification Division.

Given a problem statement, write the Identification Division for the program.

Identify and describe the Environment Division.

Name two sections that are within the Environment Division and state their function.

Identify and describe the Data Division.

State the purpose of the Data Division and name two major areas.

Identify and describe the Procedure Division.

Name two types of statements that are part of the Procedure Division and state their purpose.

Culminating Activity:

List in order the four major divisions of any COBOL program and briefly describe each. (Appendix G-1, Part A)

UNIT 7. INTRODUCTION TO COBOL

The student should be able to develop a decision table from a set of instructions.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify the conditions of a given problem.

Given a problem in narrative form, write the conditions of the problem.

Identify the actions of the problem.

Given a problem statement and the conditions, write the actions to be taken.

Determine the condition and action entries.

Given a problem statement, conditions, and actions, prepare the condition and action entries.

Transfer the elements to graphic form.

Given a problem statement with identifying conditions underscored, prepare a decision table.

Test the conditions for accuracy and completeness.

Given a decision table and a problem statement, test the decision table. (Appendix G-1, Part B)

UNIT 7. INTRODUCTION TO COBOL

The student should be able to create an appropriate flowchart from a given job description.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Develop a flowchart to show the solution of a problem.

Given a problem statement, construct a simple flowchart. (Appendix G-2)

Modify the flowchart.

Given the problem statement and a completed flowchart, make modifications to the flowchart to show improvement in processing. (Appendix G-3)

Culminating Activity:

Write the flowchart for problem 2: (Appendix G-4)

Sample Criterion:

With all elements present
With no missing connectors
With no incorrect decisions
With correct symbols

UNIT 7. INTRODUCTION TO COBOL

The student should be able to write the various sections of a COBOL program on a coding sheet.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Code the Identification Division.

Given a problem statement, write the Identification Division for the problem. (Appendix G-5, Part A).

Code the Environment Division.

Using the previous problem statement, write the Environment Division for the problem. (Appendix G-5, Part B)

Code the Data Division.

Using the same problem statement, write the Data Division for the problem. (Appendix G-5, Part C)

Code the Procedure Division.

Using the same problem statement, write the Procedure Division for the problem. (Appendix G-5, Part D)

UNIT 7. INTRODUCTION TO COBOL

The student should be able to test the program.

PERFORMANCE OBJECTIVES	and ASSESSMENT MEASURES
Identify the relationship between job control, the coded program, and test data.	Explain the purpose of each group of cards in the COBOL run deck.
Recognize three types of program testing and the purpose of each.	List three types of program testing, and state when testing of a program will occur.
Remove all errors that occur before and during compile time.	Given a problem statement, and completed coding sheets, keypunch the source deck. Note and correct compile errors.
Develop test data to measure all criteria established in the set of instructions.	Given a problem statement, state all conditions that must be met, and develop data to test each condition.
Remove all errors that occur during execution time.	Given a computer printout of an executed problem which contains errors, locate the errors and determine the action to be taken. (Appendix G-6)
Analyze a Core Dump.	Given a computer Core Dump, examine the dump for error location.

UNIT 7. INTRODUCTION TO COBOL

The student should be able to document the job.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Finalize the narrative to explain the problem solution.

Write a description of the job in narrative form.

Prepare necessary forms to fully document the program.

Complete the forms and gather the documents in proper order for a given problem. (Appendix G-7)

UNIT 7. INTRODUCTION TO COBOL

The student should be able to write and document a program with headings on each page of a report.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Identify entries required to control printing of headings.

List one or more methods used to control the printing of headings, explain their use.

Develop simple IF . . . ELSE statements.

Write an IF . . . ELSE statement to cause a loop back to AGAIN until a counter called MAX is equal to 66. When MAX is equal to 66, branch to HEADING-ROUTINE.

Develop a heading in working storage.

Given a problem statement with a heading, create the heading in working storage using the VALUE clause.

Indicate vertical and horizontal spacing of heading and body lines on a spacing chart.

Given a problem statement and a spacing chart, sketch the location of the heading and detail lines on the chart.

Develop a flowchart involving headings.

Given a problem statement involving headings, draw that section of the flowchart to show the headings.

Develop a problem program using the problem statement, flowchart, and narrative.

Plan and code a program, given a problem statement, printer spacing chart, and coding forms.

Develop documentation for a successfully executed program.

Given a compiled program, test and debug the program. Upon successful completion, organize the documentation.

Culminating Activity:

Given a written problem assignment to read and print a class list:

Write the problem statement.
Draw the flowchart or decision table.
Write the COBOL coding instructions.
Compile the program.
Test and debug the program.
Fully document the problem.

(Appendix G-8)

UNIT 7. OUTLINE	REFERENCE
I. Review Programming Cycle	Shelly and Cashman
A. Problem statement	
B. Problem analysis and organization	
C. Problem definition	
1. Decision table	
2. Flowcharting	
D. Problem solution	
1. Narrative	
2. Coding	
E. Testing and debugging	
F. Documentation	

II. Overview

A. Language development

1. Why COBOL
   a) Advantages
   b) Limitations
2. Who developed COBOL
   a) CODASYL
   b) ANSI

B. Mechanics of the language

1. EBCDIC character set
   a) Letters
   b) Numbers
   c) Special characters
   d) Editing characters
   e) Relational operators
2. Punctuation
3. Word formation
   a) Nouns
   b) Verbs
   c) Reserved words

C. Types of names

1. Data names
2. External names
3. Programmer names
4. Paragraph names

D. Functions of statement

1. Input/output
2. Conditional
3. Arithmetic
4. Branching
5. Data manipulation

E. Major restrictions

F. Perform appropriate exercises

III. Program Organization — Burroughs

A. The COBOL coding sheet

1. Sequence numbers
2. Continuation indicators
3. Source program statements
4. Program identification

UNIT 7. OUTLINE	REFERENCE
B. Identification Division	Shelly and Cashman
1. General description	
2. Requirements	
3. Optional segments	
C. Environment Division	
1. General description	
2. Configuration section	
3. Input-output section	
D. Data Division	
1. General description	
2. File section	
3. Working-storage section	
E. Procedure Division	
1. General description	
2. Open and close statements	
IV. Decision Tables	Awad
A. Review decision tables	
B. Development of the table	
1. Common elements	
2. Creating the table	
3. Checking accuracy	
V. Develop the Flowchart	Shelly and Cashman
A. Review flowchart symbols	
B. Create the COBOL flowchart	
1. Organization	
a) Input/output	
b) Junction points	
c) Terminal points	
d) Looping	
2. Perform the COBOL flowchart	

VI. Code the Problem

A. Code the Identification Division

1. General rules
2. Syntax rules
3. Optional segments

B. Code the Environment Division — Shelly and Cashman

1. General rules
2. Syntax rules
3. Configuration section
   a) Source-computer
   b) Object-computer
   c) Special-names
4. Input-output section
   a) File-control
   b) I-O-control

C. Code the Data Division

1. General rules
2. File section
   a) File description entries
   b) Record description entries
3. Working-storage section
   a) Level Indicators
   b) Formats
      (1) Alphabetic
      (2) Numeric
      (3) Alphameric (alphanumeric)
   c) Editing features

D. Code the Procedure Division

1. General rules
2. I-O statements
   a) Open
   b) Close
   c) Read
   d) Write
3. Data manipulation — move
4. Conditional statement — apply overflow
5. Sequence control
   a) Go to
   b) Stop

VII. Job Control Language — Shelly and Cashman

A. Job statement.

B. Option statement

C. Exec COBOL

D. Exec LNKEDIT

E. Exec

F. Miscellaneous

VIII. Test the Program.

A. Techniques of debugging

1. Compile messages
2. Execution messages

B. Prelist the program.

C. Debug the program.

1. Desk check
2. Error correction
3. Rerun

D. Compile the program.

1. Desk check
2. Error correction
3. Rerun

E. Develop test data.

1. Design input record.
2. Meet all criteria.

F. Execute the program.

G. Abnormal end to a program

IX. Document the Job. — Shelly and Cashman

A. The narrative

B. Machine configuration form

C. The checklist

1. Layout sheet
   a) Card
   b) Printer
2. Program listing
3. Control card listing
4. Flowchart

D. Operating instructions

E. Sample output

F. Perform appropriate exercises

X. Horizontal and Vertical Spacing

A. Environment Division

1. I O control – apply form overflow
2. ANSI method – special names

B. Working-storage section

1. Value clause
2. Vertical spacing control – 77 level
3. Headings

C. Procedure Division

D. Flowchart with headings

XI. Report Preparation

A. Editing features

1. Sending fields
2. Receiving fields

B. Headings

1. Major headings
2. Columnar headings

C. Page control – counter method

1. Single IF – ELSE statements
2. Counter
   a) Initialize
   b) Increment
   c) Reset

RESOURCES FOR UNIT 7. INTRODUCTION TO COBOL

Teacher References

Burroughs Corporation. *Burroughs B2500, 3500 Information Processing System COBOL Reference Manual.* Detroit, Mich.: Burroughs Corporation, 1969.

Bernard, Solomon M. *System 360 COBOL.* Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1968.

Textbooks

Cashman, Thomas J. *Introduction to Computer Programming System 360 COBOL.* Fullerton, Cal.: Anaheim Publishing Company, 1970.

Shelly, Gary B., and Cashman, Thomas J. *Introduction to Computer Programming ANSI COBOL.* Fullerton, Cal.: Anaheim Publishing Company, 1973.

. *Introduction to Flowcharting and Computer Programming Logic.* Fullerton, Cal.: Anaheim Publishing Company, 1972.

UNIT 8. INTERMEDIATE COBOL

This unit is designed to give the student an increased knowledge of the business oriented language COBOL. Students will write programs, develop flowcharts, and document problems involving arithmetic operations, conditional and unconditional loops, subscripts, control fields, subprograms, and table files.

At the completion of the unit INTERMEDIATE COBOL, the student should be able to.

- Write appropriate COBOL statements that involve arithmetic calculations
- Design and use IF statements that result from compound conditions
- Prepare flowchart outlines that visually identify compound conditions
- Write a COBOL program that includes minor, intermediate, and major control field totals
- Develop COBOL statements that use the perform verb
- Write a COBOL program that continues the concept of subscripting
- Construct a COBOL subroutine
- Write a COBOL subprogram
- Prepare single and multilevel table files and extract elements from the files

UNIT 8. INTERMEDIATE COBOL

The student should be able to write appropriate COBOL statements that involve arithmetic calculations.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Develop statements that use arithmetic operators in symbolic and verb form.

Given a series of statements involving arithmetic calculations, write COBOL statements to add, subtract, multiply, and divide.

Explain the use of the GIVING option.

Given the series of COBOL statements prepared in the previous exercise, rewrite the statements using the GIVING option.

Explain the use of COMPUTE and ROUNDED options.

Given the series of COBOL statements prepared in the exercise above, rewrite the statements using COMPUTE and ROUNDED verbs.

Design work areas and receiving fields that are appropriate for arithmetic computations.

Given a series of problem statements involving arithmetic values, construct working-storage areas to show those values in binary, packed decimal, and display formats.

UNIT 8. INTERMEDIATE COBOL

The student should be able to design and use IF statements that result from compound conditions.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Develop condition names using the 88 level in working storage.

In working storage, use the 88 level to write an elementary item called PAYCODE which uses a code of 1 for BAD, 2 for FAIR, and 3 for GOOD. (Appendix H-1, Part A)

Design IF statements that involve AND conditions.

Using the working-storage section developed in the previous problem, write an IF statement to meet the following conditions. when MAX is equal to TOTAL and PAYCODE is equal to a value of 1, go to NO-CREDIT. (Appendix H-1, Part B)

Design IF statements that involve OR conditions

Using the previous working-storage section, write an IF statement to meet the following conditions. when TOTAL is smaller than MAX or the PAYCODE is equal to a value of 3, go to OK-ROUTINE. (Appendix H-1, Part C)

UNIT 8. INTERMEDIATE COBOL

The student should be able to prepare flowchart outlines that visually identify compound conditions.

PERFORMANCE OBJECTIVES
and ASSESSMENT MEASURES

Develop a flowchart using IF-ELSE concepts.

Construct a flowchart segment that will show graphically the following procedure statement IF ANSWER = 2 GO TO NEXT-ROUTINE, ELSE GO TO MAIN-LINE.

Design a flowchart using AND conditions.

Construct a flowchart segment to show the following: IF MAX IS EQUAL TO TOTAL AND PAYCODE IS BAD, GO TO NO-CREDIT.

Design a flowchart using OR conditions.

Construct a flowchart segment to show the following: IF TOTAL IS LESS THAN MAX OR PAYCODE IS GOOD, GO TO OK-ROUTINE.

Design a flowchart using GO TO . . . DEPENDING ON conditions.

Given the statement: GO TO (receipts, payments, customer credit, general) DEPENDING ON A-CODE draw the flowchart to show the interrelationship of the elements.

UNIT 8. INTERMEDIATE COBOL

The student should be able to write a COBOL program that includes minor, intermediate, and major control field totals.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain and use the concept of control breaks.

Given a problem containing minor, intermediate, and major control fields, with an amount that is to be accumulated and transferred, state the result of a change in the intermediate field.

Develop a program segment containing minor, intermediate, and major totals.

Write a procedure segment for the following problem. A record contains a store number (minor), a city number (intermediate), a state number (major), and an inventory amount. The store number, the amount, and the accumulated totals are to be printed. The totals are to be transferred according to rules governing control breaks.

Prepare a flowchart which involves minor, intermediate, and major totals.

Draw a flowchart to show the relationship of the minor, intermediate, and major routines for the preceding problem.

Prepare a printer layout for a problem containing multiple totals and group indication.

Given a printer layout sheet, sketch the layout for the previously defined problem.

Design a Procedure Division containing group indication and group printing.

For the previously defined problem, rewrite the procedure segment to show printing of the store number only at the beginning and when there is a control break (group indiciation).

UNIT 8. INTERMEDIATE COBOL

The student should be able to develop COBOL statements that use the PERFORM verb.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Develop a PERFORM statement to enter a mainline segment of a program.

Given a problem involving an end-of-page condition and a heading routine, write a statement to perform the heading routine.

Develop a PERFORM statement that involves multiple execution determined by a variety of controls.

Write the PERFORM statement to increment a counter from an initial value of 1 to a final value of 100.

Design the work areas that are appropriate for controls that are used in the PERFORM statement.

Write the working-storage sentence to establish a county that will eventually contain a number as large as 100.

Redesign a program to use the PERFORM concept.

Given a previously executed program involving a heading, code the instructions to PERFORM the heading routine.

UNIT 8. INTERMEDIATE COBOL

The student should be able to write a COBOL program that contains the concept of subscripting.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Develop subscripted values in working storage through the use of REDEFINES and OCCURS statements.

Code the working-storage section of the data division to provide for a table containing three elements of two digits each. (Appendix H-2; Part A)

Given a small table in working storage and a problem statement, show the redefining and subscripting of the elements of the table. (Appendix H-2, Part B)

Develop the procedure segment to move subscripted elements.

Write the procedure segment for the DAYS-OF-CREDIT table to initialize a COUNTER, move one subscripted value to a place called PRINTOUT; go to a paragraph called OUT after exhausting the table elements. (Appendix H-2, Part C)

Explain the use of the INDEXED BY and SET statements.

Rewrite the working-storage section for the table DAYS-OF-CREDIT to include INDEXED BY. Write a statement to set the index to a value of 4 in the above problem. (Appendix H-2, Parts D and E)

UNIT 8. INTERMEDIATE COBOL

The student should be able to construct a COBOL subroutine.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Develop a PERFORM statement that causes linkage to a subroutine.

Given a problem statement involving a heading, write the PERFORM statement to enter a HEADING-ROUTINE.

Develop a subroutine segment in the procedure division.

Write the procedure segment to print a heading from an output area called A-line; and show the segment at the end of the program. (Appendix H-3, Part A)

Prepare a flowchart for the subroutine section of a program.

Using the previous problem, draw a flowchart to show the branching to and returning from a subroutine. (Appendix H-3, Part B)

UNIT 8. INTERMEDIATE COBOL

The student should be able to write a COBOL subprogram.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Explain and use the COBOL CALL statement.

Write the CALL statement to use an Assembly program which is already in storage to write the current date. The pas field is CURRENT-DATE.

Develop the identification division of a subprogram.

Given a problem statement and specification, construct the working-storage and linkage sections of the subprogram. (Appendix H-4)

Develop the procedure division of a subprogram.

Given a problem statement and specifications, complete working-storage and linkage sections; write procedure statements for the subprogram.

Develop the identification division of a main program (calling program).

Given a problem statement and specifications, write the identification division for the mainline program. (Appendix H-4-2)

Develop the working-storage and linkage sections of a main program.

Given a problem statement and specification, construct the working-storage and linkage sections of the mainline program.

Develop the procedure division of the main program.

Given a problem statement and specifications, complete working-storage and linkage sections; write procedure statements for the main program.

The student should be able to prepare single and multi-level table files and extract elements from the files.

PERFORMANCE OBJECTIVES and ASSESSMENT MEASURES

Develop the working-storage section using the REDEFINES and OCCURS clauses.

> Given a problem statement containing a STATE-NAME (14 positions), a STATE-NUMBER (2 positions), and a STATE-AMOUNT (9 positions), initialized to zero, for each of the 50 states, construct the table in working storage. Redefine the table so that two of the nine positions of the STATE-AMOUNT will be decimal, and so that the table can be referenced by subscripts. (Appendix H-5)

Create a search argument holding area in working storage.

> Given a two position number in a card, create a search argument using the 77 level.

Develop the procedure segment that initializes a search, finds the item, and processes the item.

> Given the previous problem, write a statement to search for an element in the table If the number exceeds the limits of the table, go to ERROR-ROUTINE. Given the table above, write a statement to move the table items found to TYPEOUT

Develop multi-level table files (two or more dimensional files) using the REDEFINES and OCCURS clauses.

> Write a multi-level table in working storage which contains two class codes and four train fares. Redefine the table so that it can be referenced by subscripts. (Appendix H-6, Part A)

Create the procedure segment that moves subscripted values.

> Given the previous problem, write a statement to move the coach fare to Orlando into an output area called FARE-COST. (Appendix H-6, Part B)

Develop the procedure segment that tests limitations of the table and indicates the end of the table file.

> Given the previous problem, write a statement to test the elements and branch to READ-AGAIN if the limits of the table have been exceeded, or to continue processing if the elements are within bounds of the table.

UNIT 8. OUTLINE	REFERENCE
I. Review COBOL Flowchart and Documentation	
II. Review Elements of a COBOL Program	
A. Identification division	
B. Environment division	
C. Data division	
D. Procedure division	
III. Arithmetic and Related Statements	
A. Arithmetic operators	
1. Add.	
2. Subtract.	
3. Multiply.	
4. Divide.	
B. Giving option	Shelly and Cashman
C. Compute option	
D. Rounded option	
E. On-size error	
F. Usage statements in working storage	
1. Binary	
2. Packed decimal	
3. Display	
G. Perform appropriate exercises.	
IV. Conditional Statements	
A. Review simple IF statement	
B. Level 88 in data division	
C. Compound conditions — Boolean logic	
1. AND operators with IF statement	
2. OR operators with IF statement	

D. Flowchart using — Shelly and Cashman

1. AND operators
2. OR operators

E. Perform appropriate exercises.

F. GO TO depending on clause.

G. Alter statement.

V. Major, Intermediate, and Minor Totals — Shelly and Cashman

A. Control field

1. Minor
2. Intermediate with minor
3. Major with intermediate and minor

B. Flowchart

C. Group printing and indicating

D. Printer layout

E. Problem application

VI. Looping and Indexing

A. Perform — varying statements

1. Examples
2. Flowcharting
3. Program coding
   a) Data division
      (1) 77 level
      (2) Other levels
   b) Procedure division
4. Perform appropriate exercises.

B. Subscripting

1. Rules for subscripting
2. Use of redefines clause
3. Use of occurs clause
4. Problem coding
5. Perform appropriate exercises.

C. Indexing

1. Rules for indexing
2. Use of indexed by clause
3. Use of set statement
4. Perform appropriate exercises.

VII. Subroutines and Subprograms — Shelly and Cashman

A. Review the perform verb.

B. Use of perform with a subroutine

C. Flowchart

D. Perform appropriate exercises.

E. Use of Assembly subprogram

1. Advantages
2. Simple subprogram segment
3. COBOL call statement

F. Other types of subprograms

1. Advantages
2. Simple subprogram segment statement

G. Subprogram development

1. Identification division
2. Data division
   a) Working-storage section
   b) Linkage section
3. Procedure division
   a) Enter linkage
   b) Entry — using statement
   c) Enter COBOL
   d) Enter linkage
   e) Return

H. Calling program development

1. Identification division
2. Data division
3. Procedure division
   a) Enter linkage
   b) Call subprogram
   c) Enter COBOL
   d) Note

I. Problem application

VIII. Table Lookup — One-Dimensional File — Shelly and Cashman

A. Data division

1. File section
2. Working-storage section
   a) Single file
   b) Redefines clause
   c) Occurs clause

B. Procedure division

1. Table limit — error routine
2. Movement of subscripted item

C. Problem application

IX. Table Search — Extracting One Item

A. Data division

1. File section
2. Working-storage section
   a) Hold area — 77 level
   b) File segments
   c) Redefines clause
   d) Occurs clause

B. Procedure division

1. Initialize hold area
2. Search initialized
3. Search item found
4. Search item not found
5. End of search routine

C. Problem application

X. Table Lookup — Multilevel File — Shelly and Cashman

A. Data division

1. File section
2. Working-storage section
   a) Two-dimensional file
   b) Redefines clause
   c) Occurs clause

B. Procedure division

1. Table limit — error routine
2. Movement of subscripted item

C. Problem application

XI. Table Search — Extracting Several Items

A. Data division

1. File section
2. Working-storage section
   a) File segments
   b) Output area

B. Procedure division

1. Transfer of search items found
2. End of search routine

C. Problem application

RESOURCES FOR UNIT 8. INTERMEDIATE COBOL

Teacher References

Burroughs Corporation. *Burroughs B2500/3500 Information Processing System COBOL Reference Manual.* Detroit, Mich.: Burroughs Corporation, 1969.

Bernard, Solomon M. *System 360 COBOL.* Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1968.

Textbooks

Cashman, Thomas J. *Introduction to Computer Programming System 360 COBOL.* Fullerton, Cal.: Anaheim Publishing Company, 1970.

Shelly, Gary B. and Cashman, Thomas J. *Introduction to Computer Programming ANSI COBOL.* Fullerton, Cal.: Anaheim Publishing Company, 1973.

Introduction to Flowcharting and Computer Programming Logic. Fullerton, Cal.: Anaheim Publishing Company, 1972.

BIBLIOGRAPHY

TEACHER REFERENCE MATERIAL

Bernard, Solomon M. *System 360 COBOL.* Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1968.
Descriptive course outline and reference text are especially helpful to the teacher in planning to teach the fundamentals and program development of the COBOL Language.

Burroughs Corporation. *Burroughs B2500/3500 Information Processing System COBOL Reference Manual.* Detroit, Mich.: Burroughs Corporation, 1969.
This reference manual contains a complete listing of COBOL terms. The arrangement of material by individual terms within the division is especially helpful for the *beginning teacher.*

Board of Education, City of Chicago. *Curriculum Guide for Introduction to Data Processing.* Chicago. Board of Education of the City of Chicago, 1967.
This curriculum guide can be used as a teacher reference book for the first 9-week unit, Basic Concepts of Data Processing, which is part of the MCPS Data Processing I course of study.

Eastern Business Teachers Association. *Processing Data in Business, Education, and Government.* Somerville, N.J.: EBTA, 1972.
This book can be used as a teacher reference for the first two 9-week units, Basic Concepts of Data Processing and Computer Concepts, which are parts of the MCPS Data Processing I course of study.

Feingold, Carl. *Introduction to Data Processing.* Dubuque, Ia.. Wm. C. Brown Company Publishers, 1971.
This book can be used as a teacher reference. The information on Preparation of a Report Using RPG, page 130, will serve as a learning aid. This book contains a number of photographs of equipment used in the industry, and it also gives an outline of the following programming languages. RPG, Fortran IV, COBOL, PL/1. It can be used as a student text or reference.

International Business Machines Corporation. *Introduction to IBM Direct-Access Storage Devices and Organization Methods.* White Plains, N.Y.: IBM, 1974.
This book can be used as a teacher reference in the unit on Computer Operations II in the MCPS Data Processing I course of study.

——. *The Considerations of Physical Security in a Computer Environment.* White Plains, N.Y.. IBM, 1972.
This is one of the few references available on the topic of security within a data processing organization. It addresses the problems of personnel security, equipment security, equipment failure, and data security as included in the unit Computer Operations II.

Miller, Boulton B. *Computers. A User's Introduction.* Edwardsville, Ill.. Bainbridge, Inc., 1974.
This publication can be used as either a teacher or student reference for many broad topics in the units on Computer Operations I and II of the MCPS Data Processing I course of study.

Schnake, M. A. *Data Processing Concepts.* New York. McGraw-Hill Book Company, 1973.
This publication can be used as a teacher reference for many broad topics in the units on Basic Concepts of Data Processing, Computer Concepts, Computer Operations I, and Computer Operations II of the MCPS Data Processing I course of study.

State of California Employment Development Department. *Career Guides for Entry Occupations in Data Processing* Sacramento, Cal.. State of California Employment Development Department, 1974.
This book can be used as both a teacher and a student reference for the career section in the Computer Concepts unit of the MCPS Data Processing I course of study.

TEXTBOOKS

Awad, Elias M. *Business Data Processing*, 3rd ed. Englewood Cliffs, N.J.. Prentice-Hall, Inc., 1971.
This is an excellent book which can be used as a student text, student reference, or teacher text for the first two 9-week units, Basic Concepts of Data Processing and Computer Concepts, which are parts of the MCPS Data Processing I course of study.

Cashman, Thomas J. *Introduction to Computer Programming System 360 COBOL*. Fullerton, Cal.. Anaheim Publishing Company, 1970.
This student text uses a problem-oriented approach in introducing programming concepts. Basic concepts such as hardware, software, internal data representation, flowcharting, and programming are included.

Cashman, Thomas J., and Fletcher, Dennis A. *IBM System/360 RPG Programming, Vol. I, Introduction*. Fullerton, Cal.. Anaheim Publishing Co., 1967.
The content covers File Description, Input, Calculation, and Output Specifications for RPG. Although it was designed as a textbook, it can best serve as both a teacher reference and as a student reference. Most entries for a given specification form are located in one chapter. The supplementary problems can be used selectively with the classroom text. This can be used selectively with the RPG programming and macro statements.

——. *IBM System/360 RPG Programming, Vol. II, Advanced Concepts*. Fullerton, Cal.. Anaheim Publishing Company, 1967.
The content covers fixed logic, branching, looping, table lookup, multiple files, and subroutines. The appendices provide an easy reference source of Core Storage Saving Rules, Core Storage Requirements, Summary of Indicators, Error Check List, EBCDIC Code Structure, Differences Between Versions of S. 360 RPG. This can be used as a teacher reference, as a student reference, and as a source of a few additional problems.

Crawford, F. R. *Introduction to Data Processing*, 2nd ed. Englewood Cliffs, N.J.. Prentice-Hall, Inc., 1973.
This book can be used as a student text, student reference, or teacher reference for the second 9-week unit, Computer Concepts, which is part of the MCPS Data Processing I course of study.

Feingold, Carl. *Introduction to Data Processing*. Dubuque, Ia.. Wm. C. Brown Company Publishers, 1971.
This may be used as a teacher reference and as a student text or reference. The information on Preparation of a Report Using RPG, page 130, will serve as a learning aid. This book contains a number of photographs of equipment used in the industry and it also gives an outline of the following programming languages: RPG, Fortran IV, COBOL, PL/1.

Fuori, William M., D'Arco, Anthony, and Orilia, Lawrence. *Introduction to Computer Operations*. New York: McGraw-Hill Book Company, 1973.
This book can be used as a student text or a teacher reference for the units on operation, Computer Operations I and Computer Operations II, of the MCPS course of study for Data Processing I.

International Business Machines Corporation, *A Data Processing Glossary*. White Plains, N.Y.. IBM, 1972.
A Data Processing Glossary serves as a dictionary of data processing terms for the students.

——. *Basic Punched Card Data Processing Programmed Instruction Manuals*. White Plains, N.Y.. IBM, 1964.
This is a series of manuals which can be used as a student text, student reference, or teacher reference for the first 9-week unit, Basic Concepts of Data Processing, which is part of the MCPS Data Processing I course of study.

——. *Introduction to IBM Data Processing Systems.* White Plains, N.Y.: IBM, 1969.
This book can be used as a student reference or a teacher reference for the second 9-week unit, Computer Concepts, which is a part of the MCPS Data Processing I course of study.

——. *Introduction to System/360 Text Programmed Instruction.* White Plains, N.Y.. IBM, 1970.
This book can be used as a student text, student reference, or teacher reference for the second unit, Computer Concepts, of the MCPS Data Processing I course of study.

——. *Number Systems.* White Plains, N.Y.: IBM, 1968.
This book can be used as a student reference or teacher reference for the second unit, Computer Concepts.

—— *System/360 Disk Operating System Operation Training Manual, Student Text.* White Plains, N.Y., IBM, 1969.
This material goes with the Advanced Systems Incorporated COT Series of books and the video tapes, or can be used as a separate text, supplementary material, or reference material for the unit on Computer Operations I in the MCPS course of study for Data Processing I.

——. *System/360 Report Program Generator Coding.* White Plains, N.Y.: IBM, 1969.
This technical manual can be used as a teacher reference and as a programmed instruction text for a student working independently when programming RPG in a System/360 environment.

——. *Systems: Introduction to RPG II.* White Plains, N.Y.: IBM, 1972.
This technical manual can be used as a teacher reference and as an advanced student text. Clear charts, graphs, and illustrations are included. Since RPG II was initially designed to be used on an IBM System 3 Computer, the references to machine cycles and functions may be directly related to that hardware.

—— *Systems Reference Library, IBM System/360 Basic Operating System Report Program Generator Specifications.* White Plains, N.Y.: IBM, 1970.
This technical manual can be used as a teacher reference and also as a student reference when programming RPG in a System/360 environment.

Marxer, Ellen *Elements of Data Processing.* Albany, N.Y.. Delmar Publishers by Litton Educational Publishing, Inc., 1971.
This book can be used as a student text, student reference, or teacher reference for the second unit, Computer Concepts, of the MCPS course of study for Data Processing I.

Murray, Jerome T *Programming in RPG II. IBM System/3.* New York: McGraw-Hill Book Company, 1971.
This is the second of two textbooks designed to teach the machine operation of the IBM System/3 computer and the program language RPG II — the main language of the System/3. This text could be used as a teacher reference and as a student textbook, especially in the area of macro statements.

National Cash Register Company. *Introduction to Programming.* Dayton, Ohio. NCR, 1969.
This textbook could be used as a teacher reference and as a student reference. There is a good example of a decision table in the text.

Shelly, Gary B, and Cashman, Thomas J. *Introduction to Computer Programming ANSI COBOL.* Fullerton, Cal.: Anaheim Publishing Company, 1973.
This student text uses a problem-oriented approach to introduce COBOL programming concepts. Basic concepts including flowcharting and programming are included. American National Standard COBOL concepts are used throughout the text.

. *Introduction to Flowcharting and Computer Programming Logic.* Fullerton, Cal.. Anaheim Publishing Company, 1972.
This book may be used as a companion text to *Introduction to Computer Programming ANSI COBOL.* The development of the flowchart by segments which are related to illustrations is extremely helpful in preparing the student to understand the relationship of the parts to the whole flowchart.

. *Introduction to Computer Programming RPG.* Fullerton, Cal. Anaheim Publishing Company, 1972.
This book is designed as a student text. The book has clear examples in the introduction to each topic. Four activities for each chapter, review questions, short coding problems, debugging RPG programs, and two problems are given.

Wanous, E. F., Wagner, Gerald E., Wanous, S. J. *Fundamentals of Data Processing.* New Rochelle, N.Y.. Southwestern Publishing Company, 1971.
This book can be used as a student text, student reference, or teacher reference for the first two units, Basic Concepts of Data Processing and Computer Concepts, of the MCPS course of study for Data Processing I.

SUPPLEMENTARY BOOKS

Advanced Systems, Incorporated. *Computer Operator Training — Video Assisted Instruction, Student Guide.* Elk Grove Village, Ill.: Advanced Systems, Inc., 1972.
This book, together with 10 video tapes and the IBM related manuals, can be used as a student text for the unit Computer Operations I in the MCPS course of study for Data Processing I.

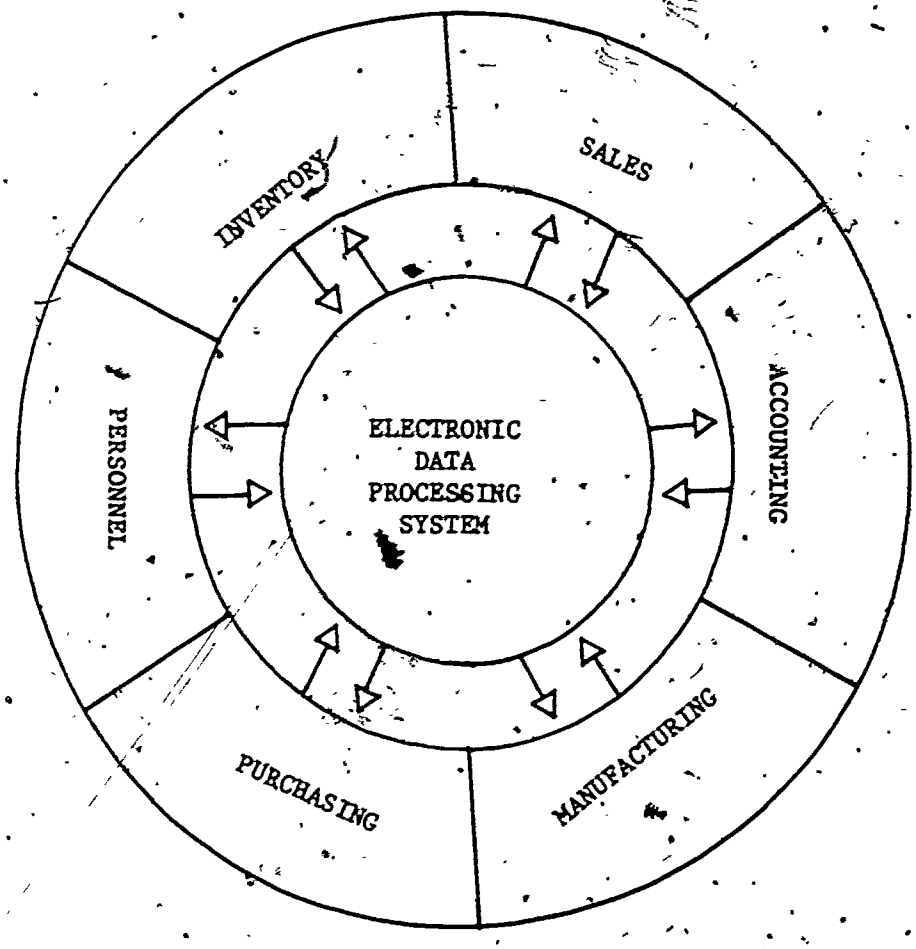
Minnesota Mining and Manufacturing Company. *Data Processing, Vols. 1-4,* and a corresponding set of transparencies. St. Paul, Minn.: 3-M Company Visual Products Division, 1966.
The four volumes can be used as a teacher reference, while the corresponding set of transparencies can be used as student visual materials.

U.S. Government Employment Service. *Dictionary of Occupational Titles — 1974-75.* Washington, D.C.. Government Printing Office, 1974.
This publication can be used as a handy reference guide to data processing and related job titles. Student and teacher reference.

PERIODICAL

"The Microfilm Picture 1975." *Infosystems,* April 1975, pp. 38-41.

APPENDIX A. BASIC CONCEPTS OF DATA PROCESSING

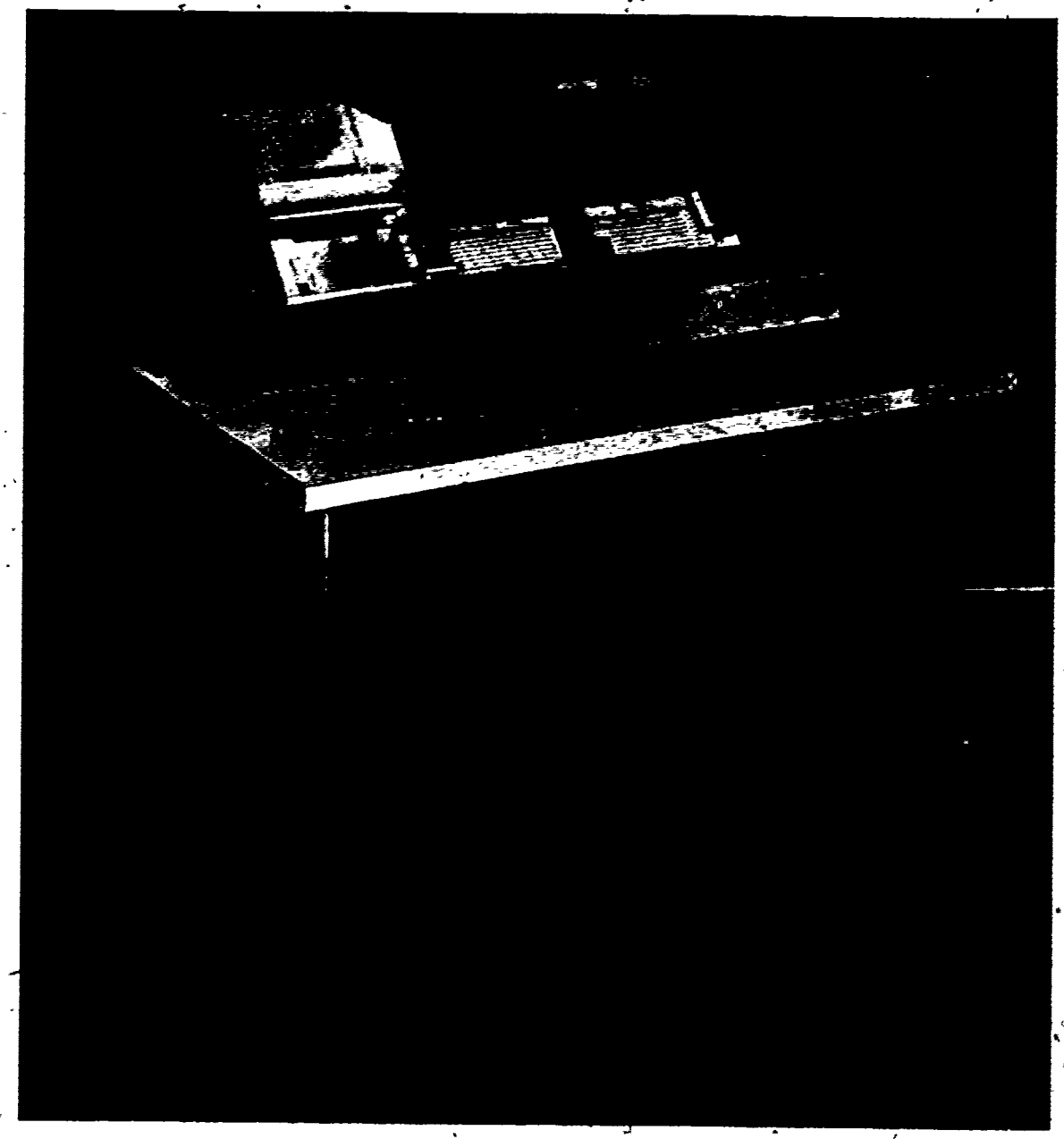


RELATIONSHIP BETWEEN

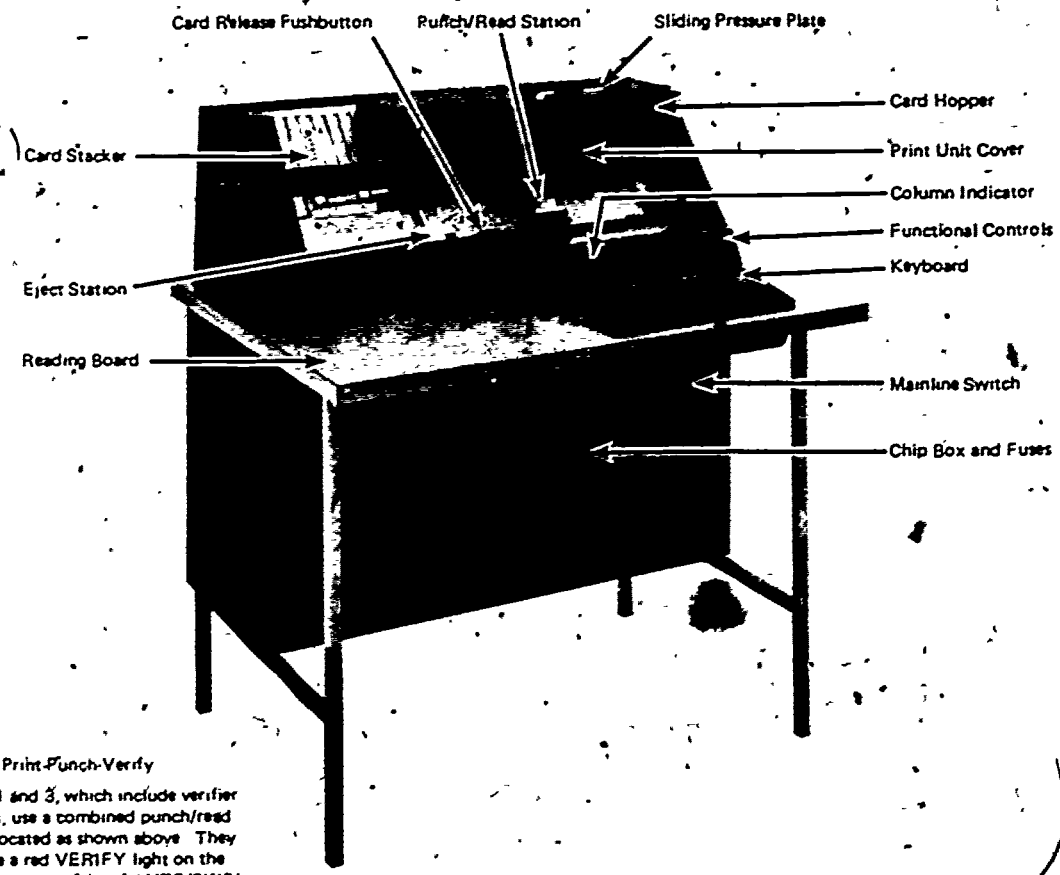
THE COMPUTER DEPARTMENT

AND OTHER DEPARTMENTS

OF A BUSINESS



IBM 29 Card Punch



Model 3 Print-Punch-Verify

Models 1 and 3, which include verifier function, use a combined punch/read station located as shown above. They also have a red VERIFY light on the keyboard to the right of AUTO/SKIP/-DUP switch, and PUNCH/VERIFY switch at left side of keyboard console.

CARD FEED PATH

Key Depressed	Card Moves	
	From	To

CARD PUNCH PROBLEMS TO TEACH SELECTED CONCEPTS

Problem 1

INSTRUCTIONS. Punch one card and then duplicate one copy of the same card and pass in both cards. When this is done correctly, the teacher will return one copy of the card so that you can tape it to the bottom of this instruction sheet.

Field Name	Card Column
Last name	1-14
First name	15-19
Initial	20
Street number	21-25
Street name	26-45
City	46-60
State	61-62
Zip code	63-67
Blank	68-80

CARD FORMAT

Last Name	First Name	Initial	Street No.	Street Name	City	State	Zip Code	Blank
1-14	15-19	20	21-25	26-45	46-60	61-62	63-67	68-80

APPENDIX A (Cont'd.).

Be sure there are NO CARDS in the card hopper for Problems 2, 3, and 4.

Problem 2

The teacher will return one copy of card punch Problem 1 to be used for this exercise.

Duplicate one copy of a master card by inserting the master card in the read bed and manually feeding one card in the punch card bed.

- Pass in both completed cards to the teacher.
- Fasten one completed card (after it has been checked by the teacher) to the back of this instruction sheet.

Problem 3

Use the master card given to you which has two errors, and duplicate correcting the errors. See • in Problem 2.

Problem 4

Use the card given to you, and interpret it on the card punch machine. Be sure to return the functional control switches to DOWN when finished.

- Pass in the one card when it is finished.
- Fasten the completed card (after it has been checked by the teacher) to the back of this instruction sheet.

Problem 5

The purpose of this problem is to lay out the *planning card* for a program card.

DIRECTIONS:

1. Prepare a program *planning card* for the fields listed below.
2. Mark each field with a vertical line.
3. Identify what kind of field is being planned, e.g., alphabetic, numeric, duplicate, sk
4. Across the top of each field, darken the rows and columns to indicate where the holes in the program card are to be punched.
5. *Print* above the 12 row *which key* you will *depress* to get the punch you want for each card column.
6. Pass in the planning card for check by teacher before you go to the card punch machine.
7. Card format

Card Column	
1-20	alphabetic field
21-30	numeric/dup field
31-45	numeric (manual punching) field
46-60	alphabetic dup field
61-80	skip field

Problem 6

Prepare a *planning card* only for the following card format:

CC 1-20	automatic alphabetic shift
CC 21-25	automatic skip
CC 26-37	numeric
CC 38-48	automatic duplication
CC 49-50	numeric
CC 51-70	automatic skip
CC 71-75	automatic duplication
CC 76-77	numeric
CC 78-80	automatic skip

Problem 7

Prepare first a *planning card* and then punch the *program card.*

CC 1-5	automatic duplication
CC 6-20	automatic skip
CC 21-41	automatic alphabetic shift and duplicate
CC 42-50	numeric
CC 51-70	automatic duplication
CC 71-80	automatic skip

Problem 8

Prepare a *program card* only while at the card punch *but* without any preplanning or planning card. *Punch while you think.*

CC 1-15	automatic skip
CC 16-20	numeric and duplicate
CC 21-40	alphabetic shift
CC 41-59	automatic duplication
CC 60-61	numeric
CC 62-70	alphabetic shift
CC 71-75	alphabetic duplication
CC 76-80	automatic skip

Problem 9

Repunch a program card combining Problems 7 and 8 on one program card. Be sure to turn the print switch off before starting.

Problem 7 on the upper program level
Problem 8 on the lower program level

Problem 10

Purpose of this problem is to:

1. Punch a program card
2. Insert the program card onto the program drum
3. Use the star wheels
4. Use your own program card to punch three data cards

Directions:

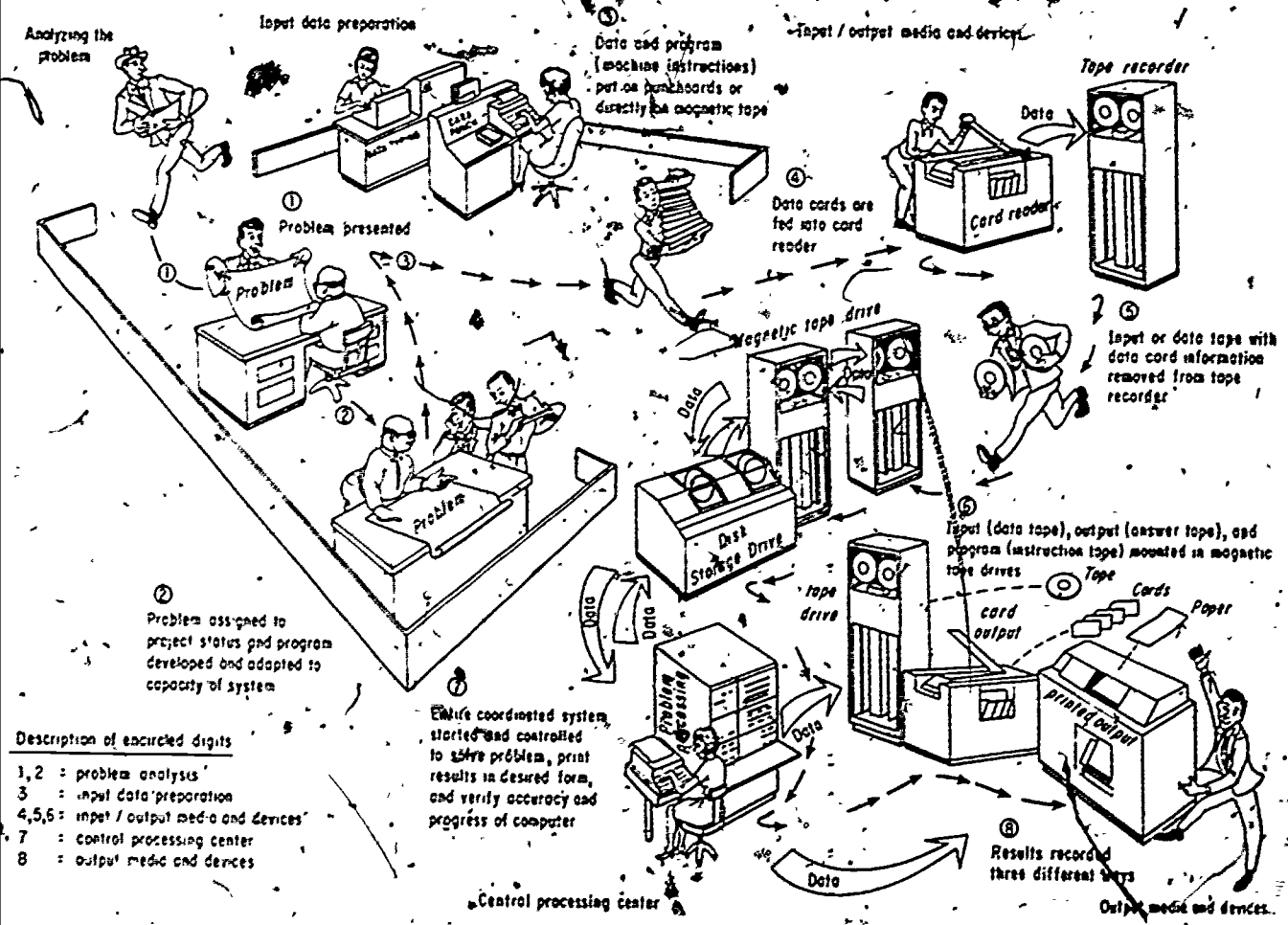
1. The card FORMAT is:

   CC 1-20 name field (last, first initial)
   CC 21-25 blank field
   CC 26-29 grade and section number field
   CC 30-35 current data field (to be duplicated)
   CC 36-80 blank field

2. Clear the card bed of all cards before starting.
3. For the first card, keep the AUTO DUP switch off.
   For the first card, turn AUTO FEED key on and feed two cards.
4. Punch the first card. Then turn AUTO DUP-AUTO SKIP switch on for the second and the third cards.
5. Data to be punched:

   1st card SMITH ALBERT J 10-24 (current date)
   2nd card HASHBARGER EMANUEL H 11-02 (current date)
   3rd card FREDRICKSON THEODORE S 12-17 (current date)

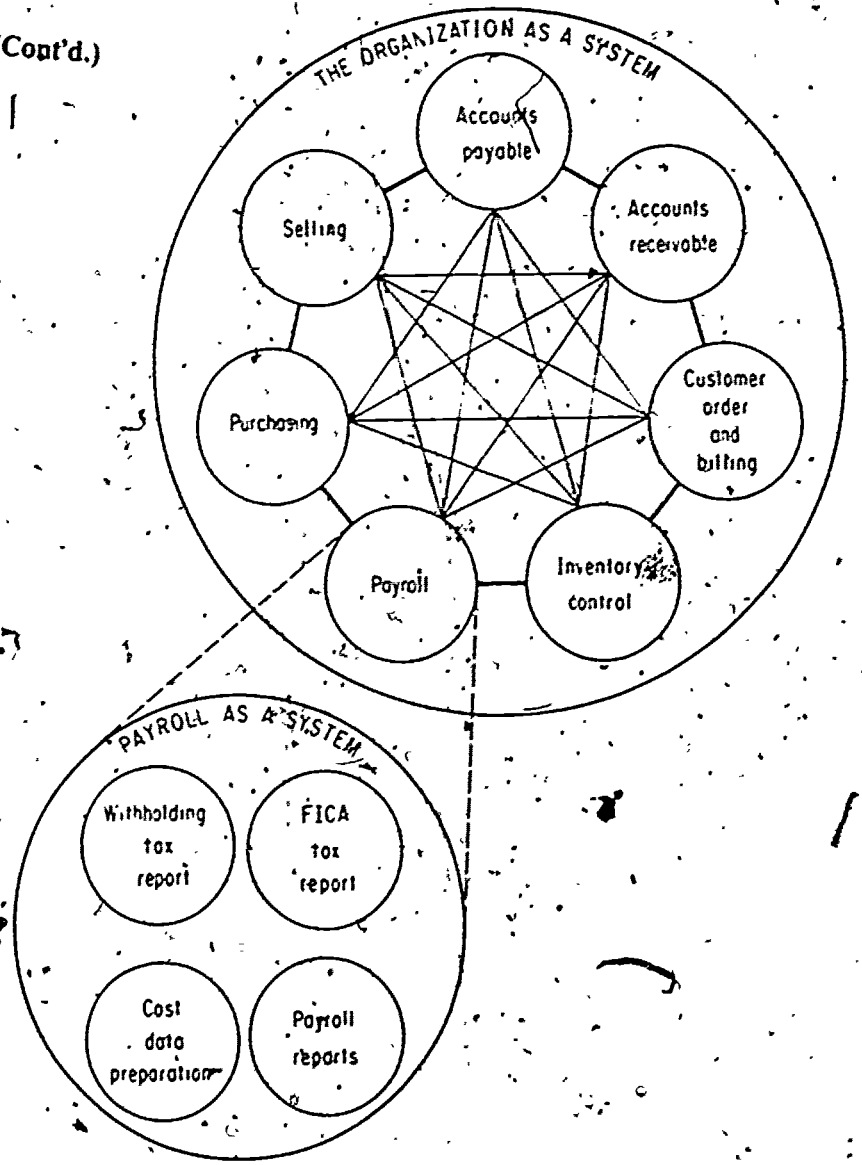
NOTE: Each card must be entirely correct to receive credit for Problem 10.



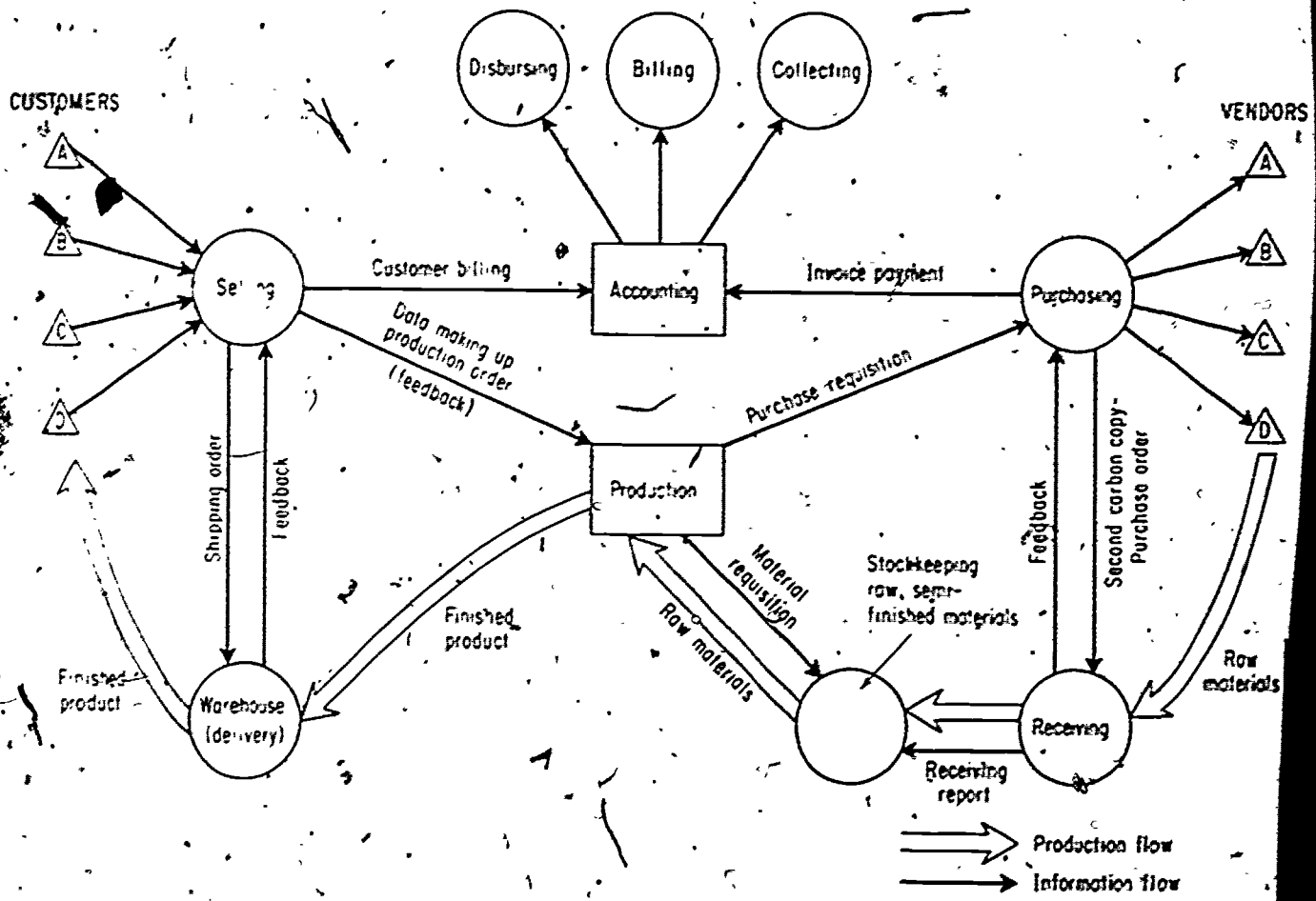
Description of encircled digits

1, 2 : problem analysis
3 : input data preparation
4,5,6 : input / output media and devices
7 : control processing center
8 : output media and devices

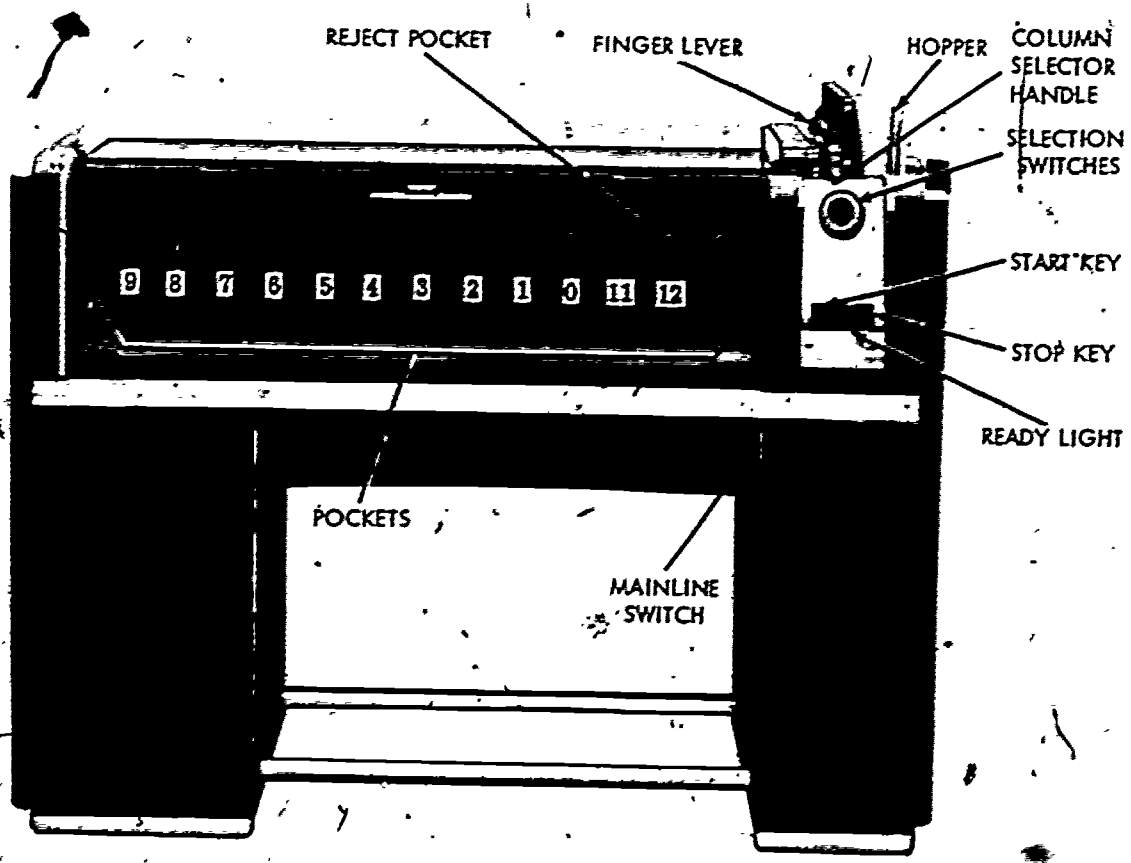
A typical business application of an electronic data-processing system



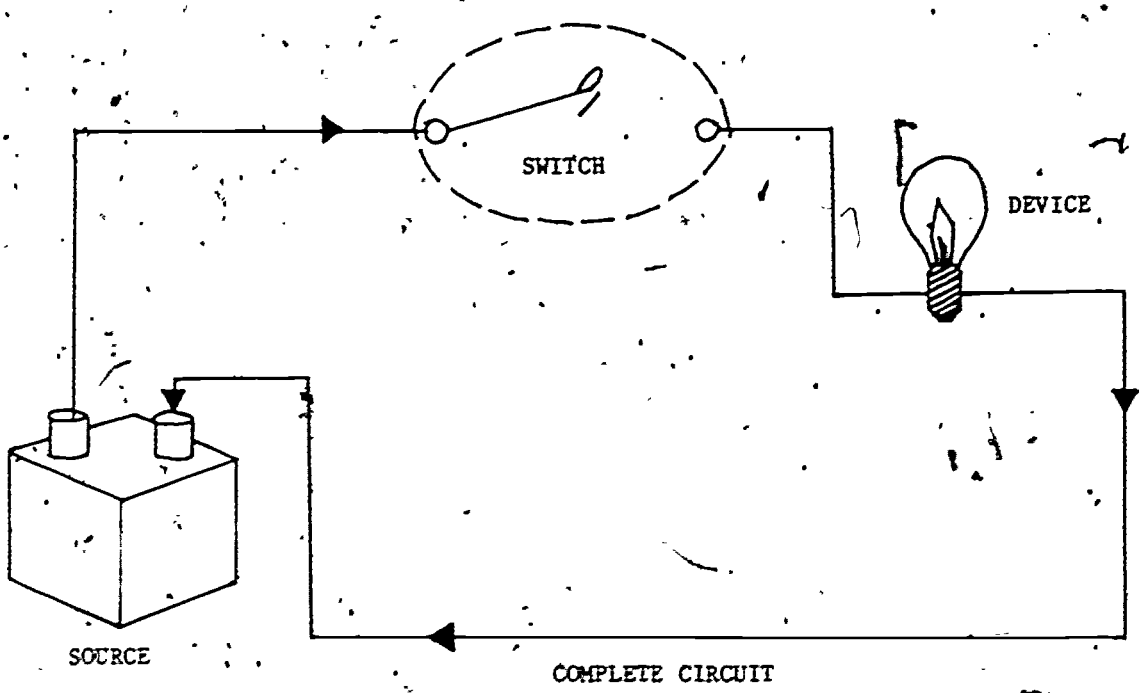
An organization as a system with seven information-generating subsystems

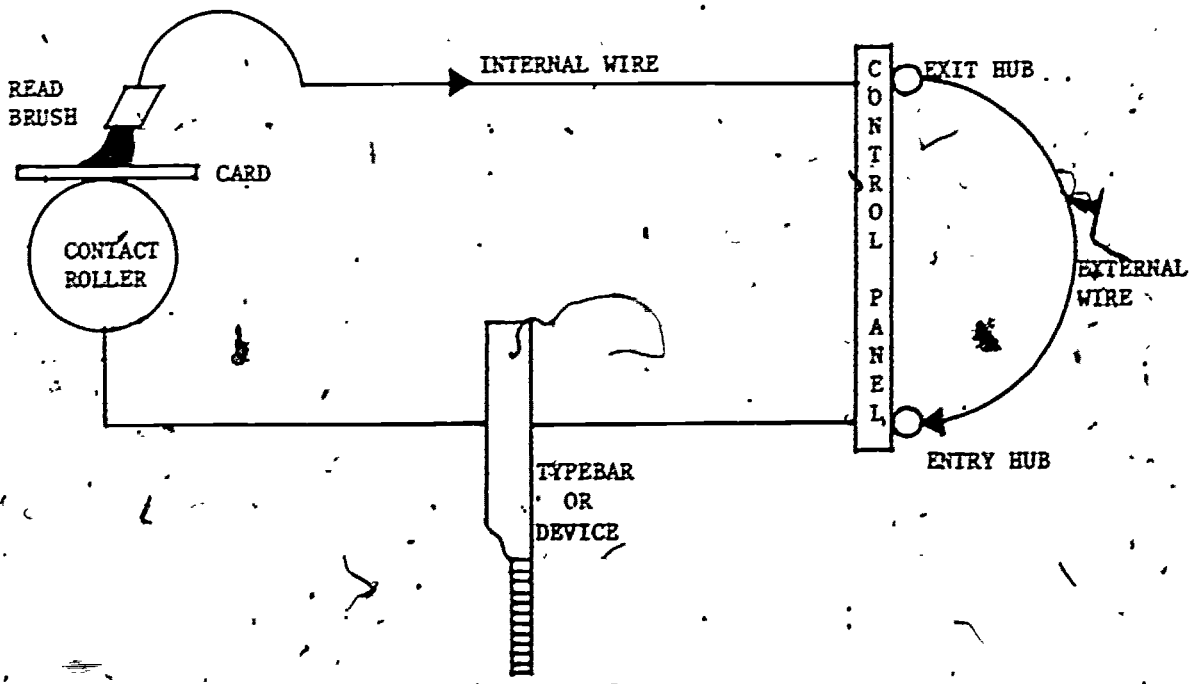


Schematic diagram of information flow in a manufacturing concern

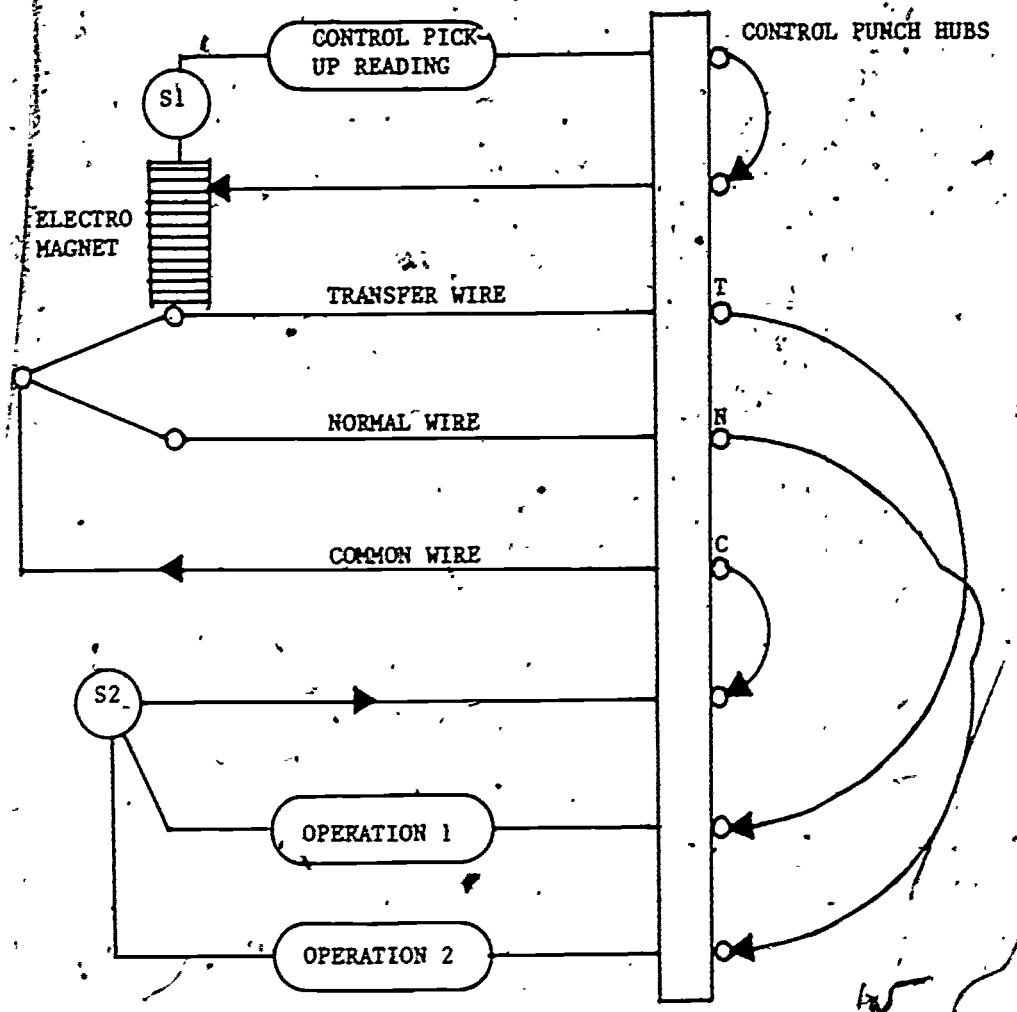

REJECT POCKET
FINGER LEVER
HOPPER
COLUMN SELECTOR HANDLE
SELECTION SWITCHES
START KEY
STOP KEY
READY LIGHT
POCKETS
MAINLINE SWITCH
9 8 7 6 5 4 3 2 1 0 11 12

BASIC PRINCIPLES OF ELECTRICITY

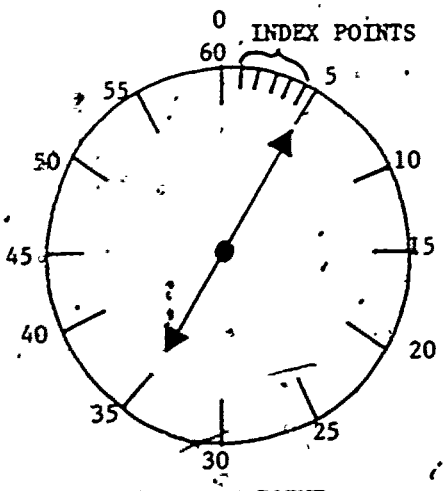




BASIC PRINCIPALES OF ELECTRICITY

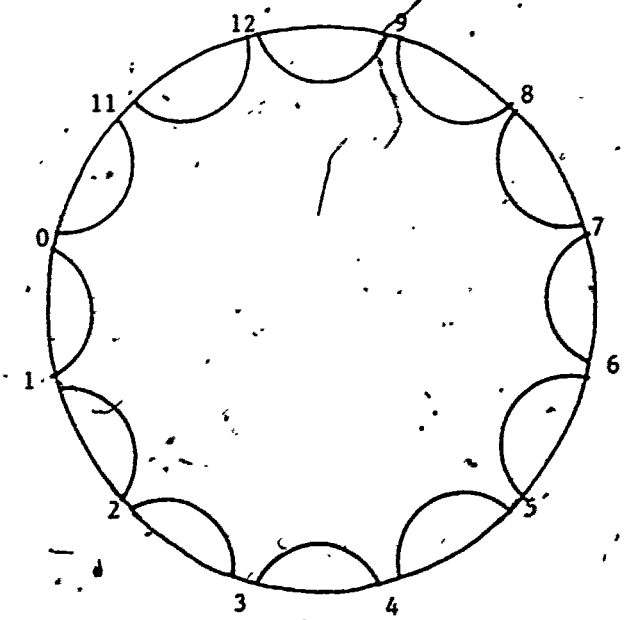


TIMING CYCLE



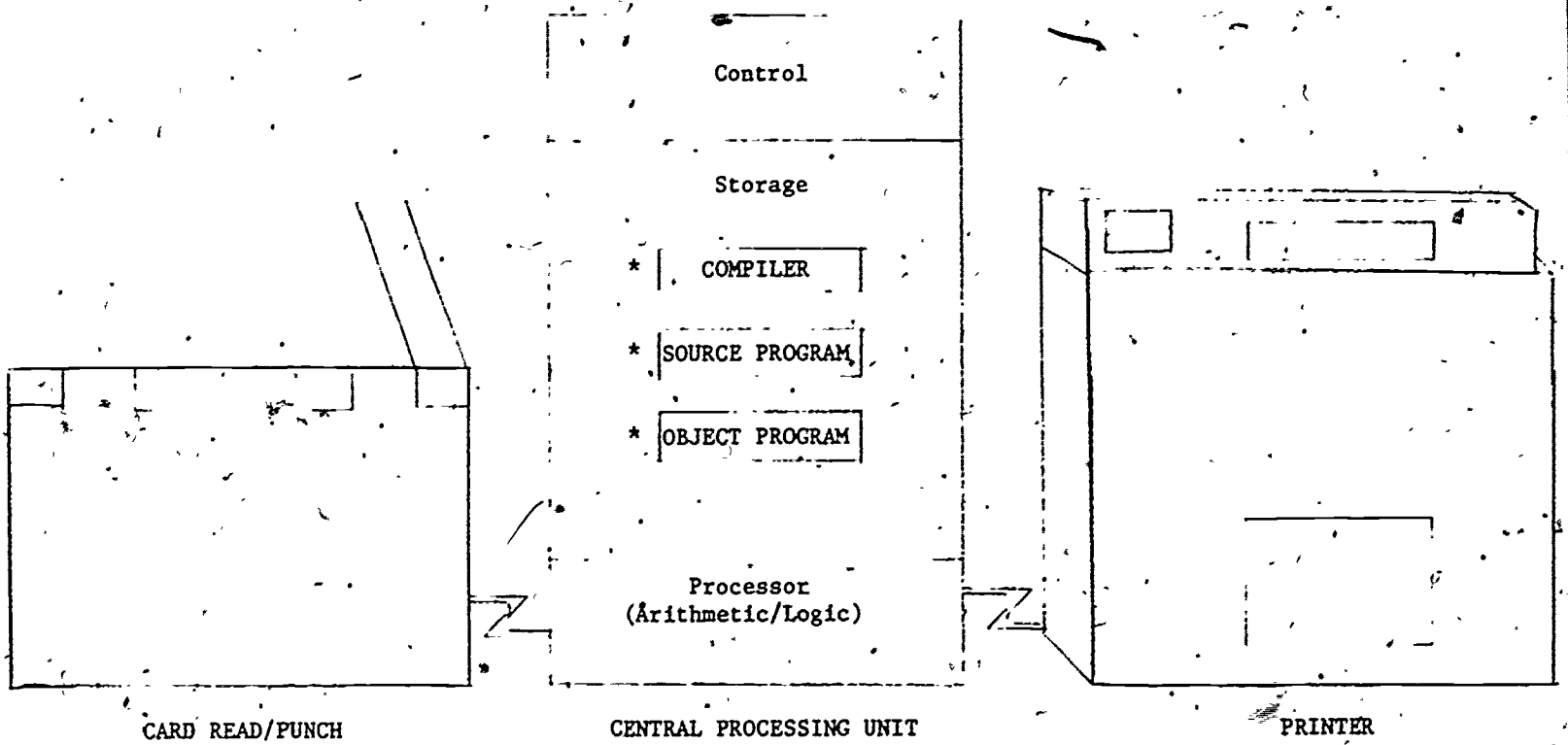
A-60 INDEX POINT
CYCLE = 1 HOUR

SORTER TIMED IMPULSES



APPENDIX B. COMPUTER CONCEPTS

RELATIONSHIP BETWEEN HARDWARE AND SOFTWARE IN A COMPUTER SYSTEM



*Software shown within the Storage Section of the CPU

ARITHMETIC QUIZ

1. Count from 0-32 in

   A. Binary

   B. Octal

   C. Hexadecimal

2. Add the following binary numbers:

   A. 1 1 1 0 B. 1 1 1 0 1
      0 1 0 1 1 0 0 1 1

3. Subtract the following binary numbers:

   A. 1 0 0 1 B. 1 1 0 0
      0 1 1 0 1 0 0 1

4. Convert the decimal number 102 into:

   A. Binary

   B. Octal

   C. Hexadecimal

5. Convert the hexadecimal number 7A7 into:

   A. Binary

   B. Decimal

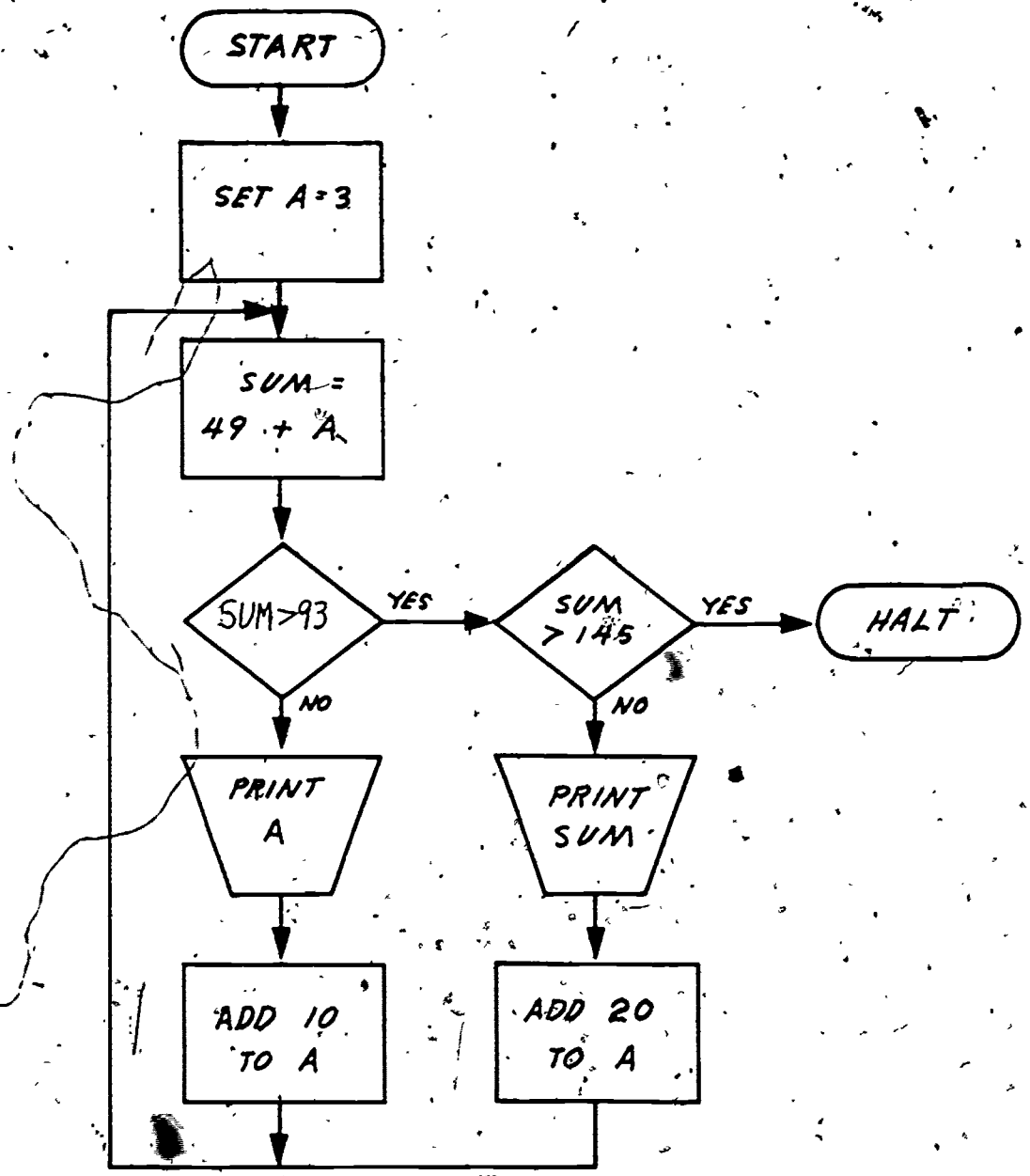
6. Code the name of your school in:

   A. Hollerith code

   B. BCD

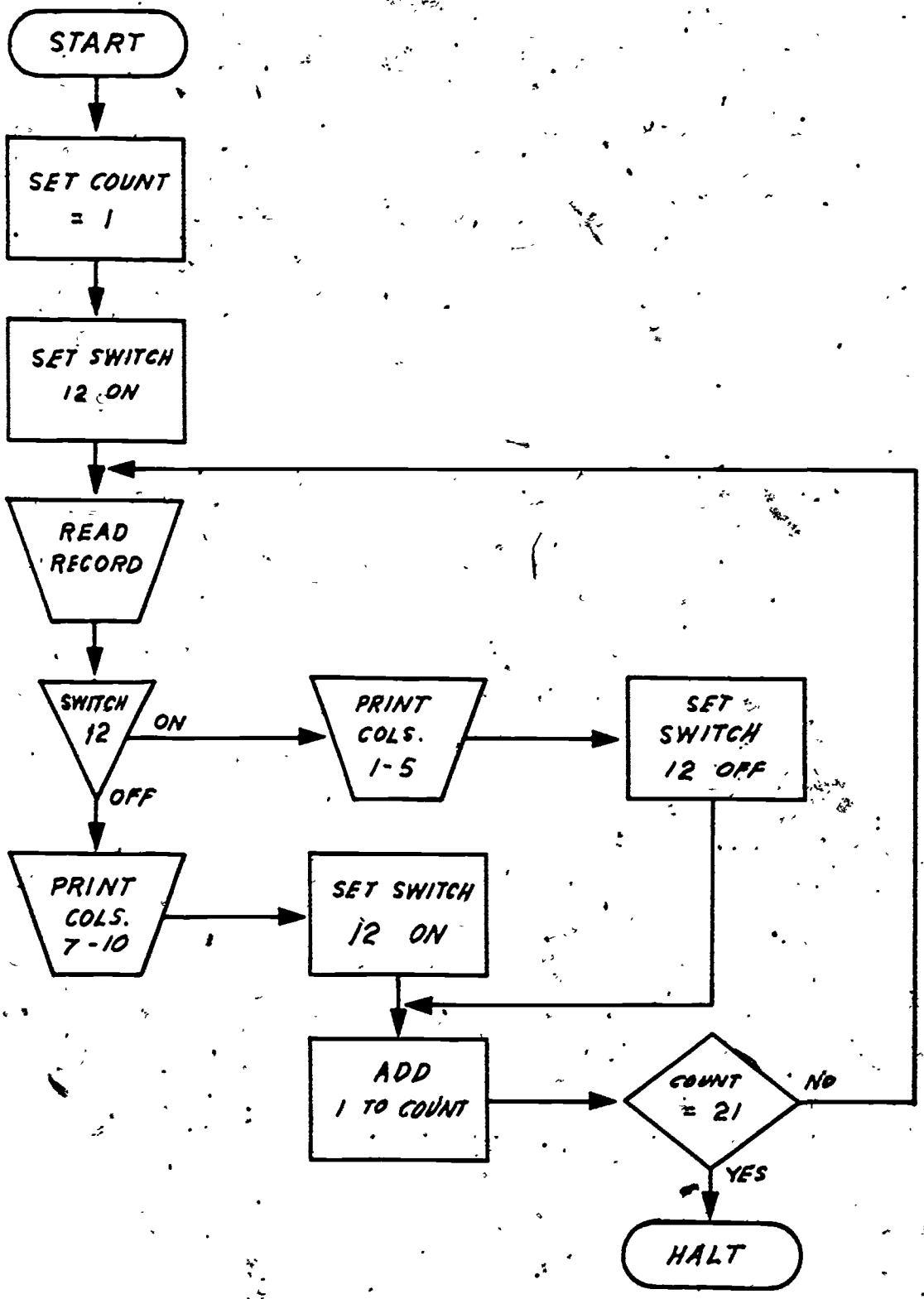
   C. EBCDIC

Look at the logic chart below and answer the following questions.



(a) "A" will be printed _____ times.
(b) _____ is the list of the values of "A" as they will be printed.
(c) "Sum" will be printed _____ times.
(d) _____ is the list of the values of "sum" as they will be printed.

Look at the logic chart below and answer the questions on the following page:



(a) Columns 1-5 of an input record will be printed _____ times.

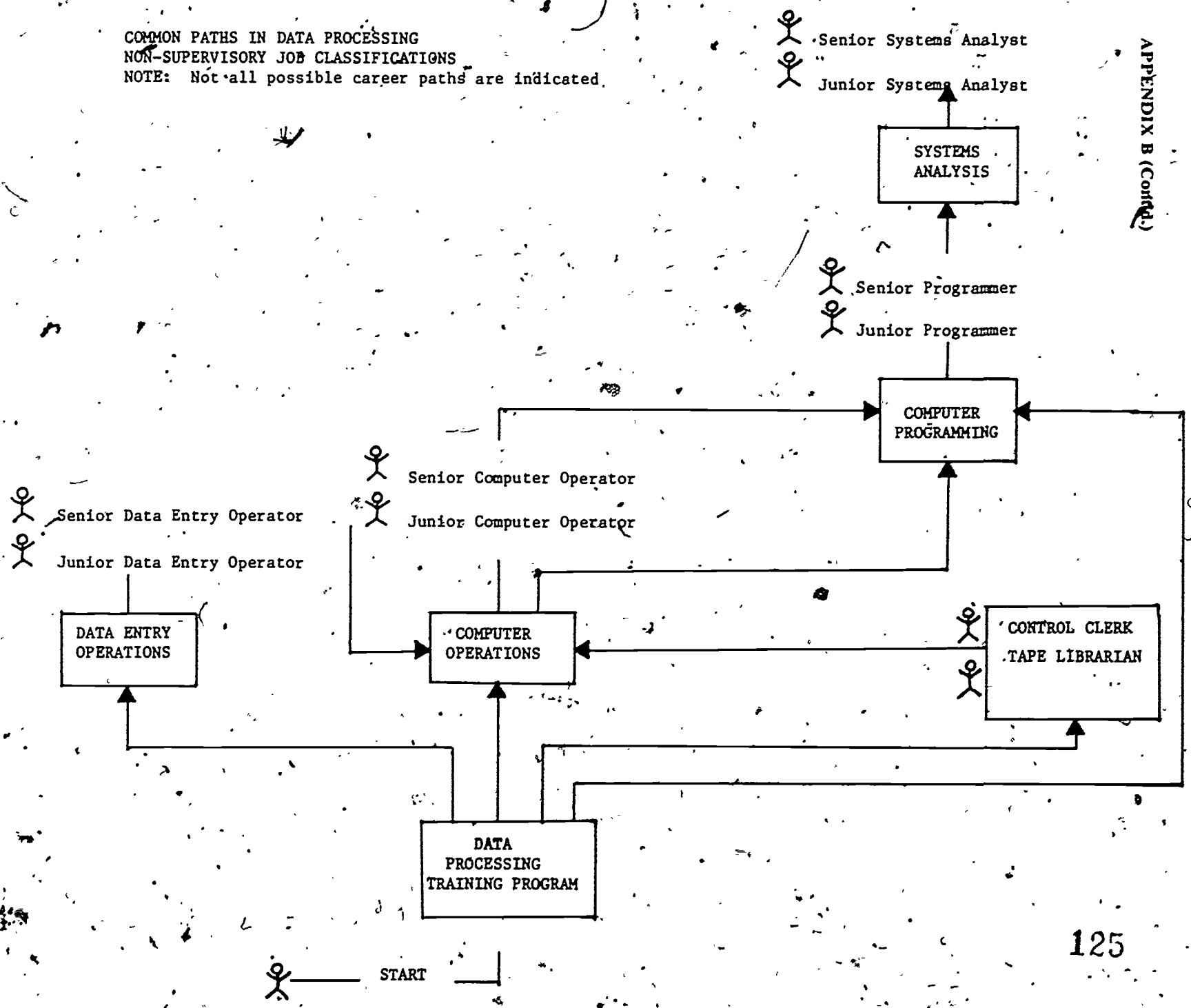
(b) Columns 7-10 of an input record will be printed _____ times.

(c) The printer will print _____ times.

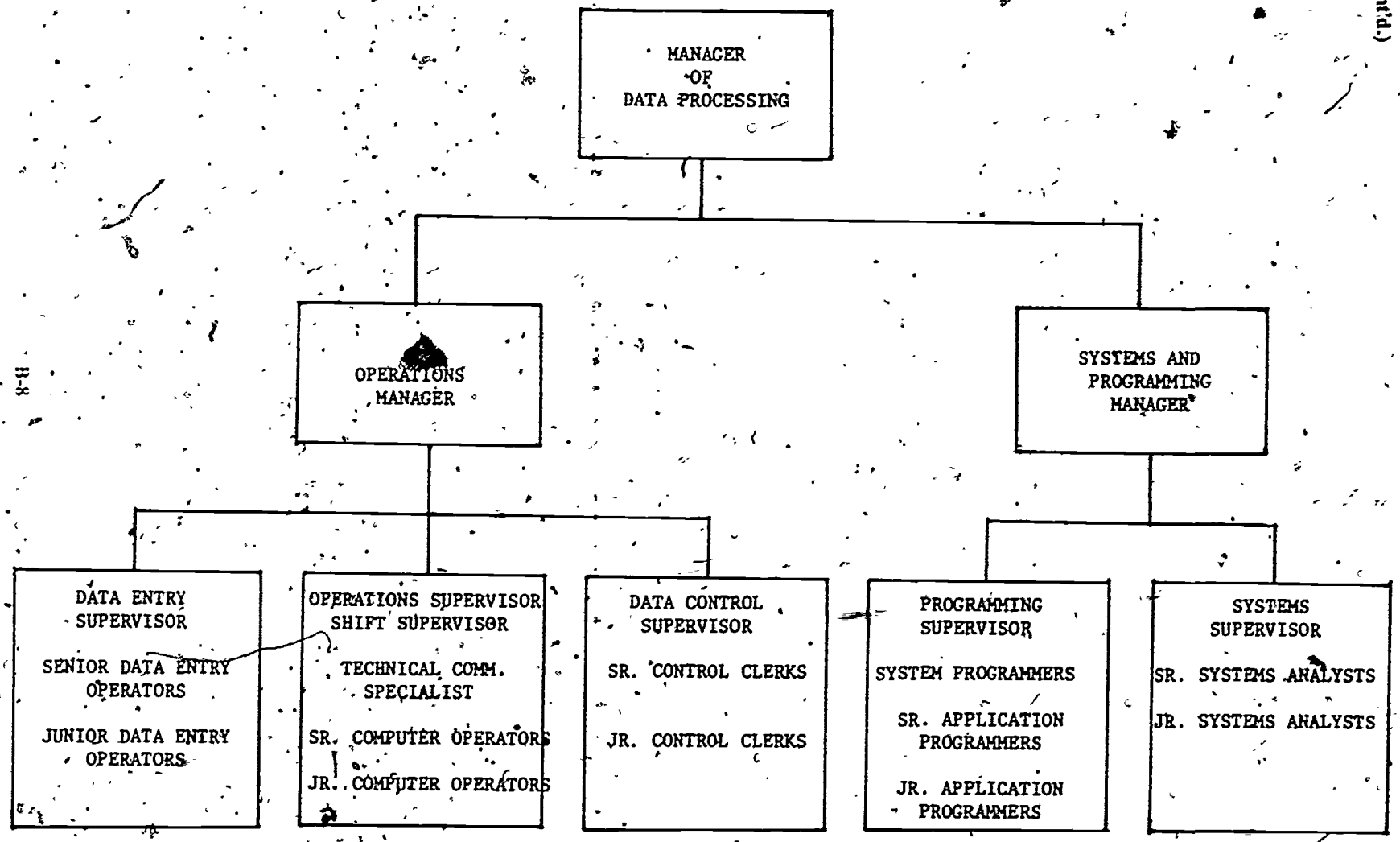
(d) The _____ block controls the number of printings.

(e) The _____ block controls which columns of a record are to be printed.

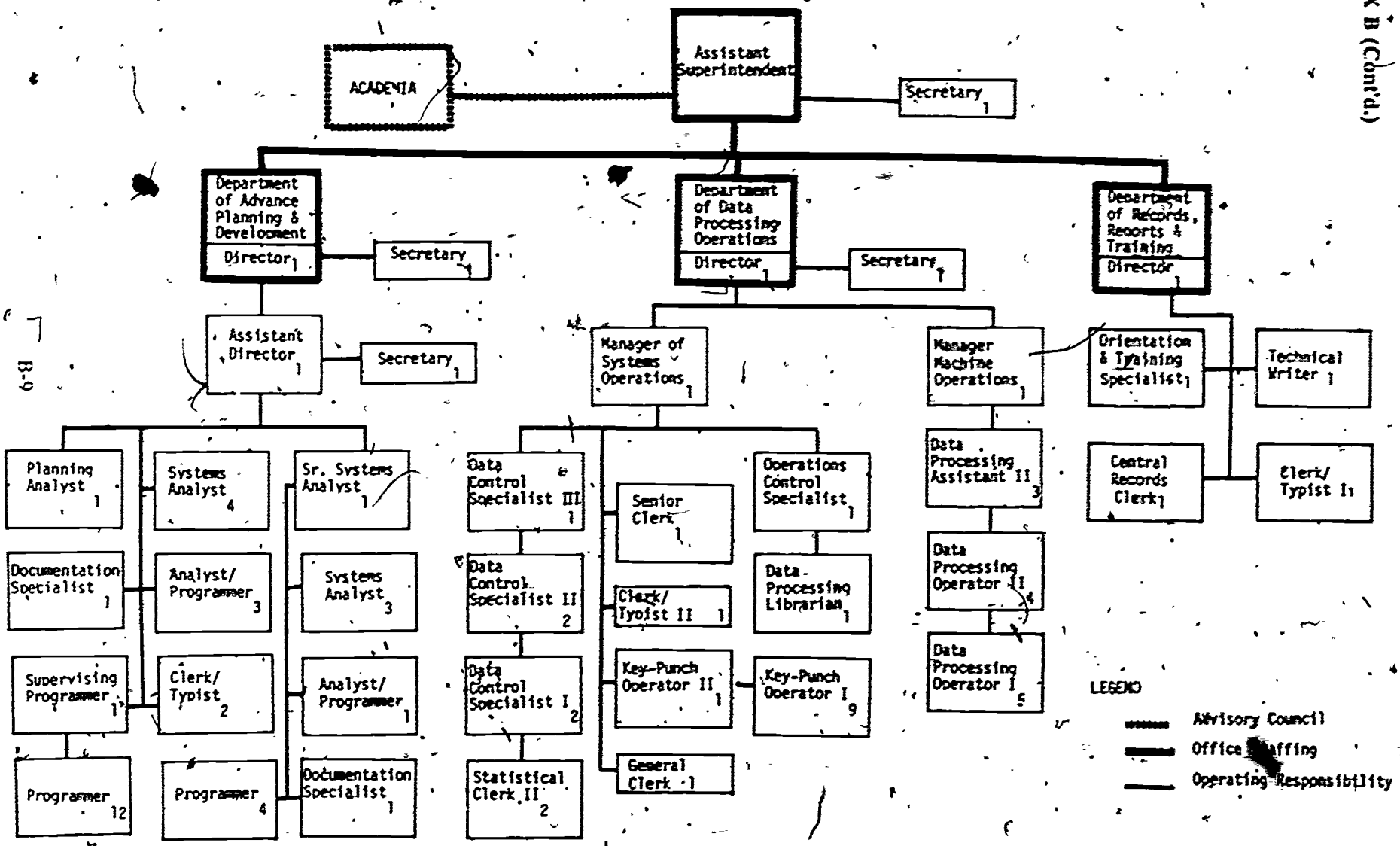
COMMON PATHS IN DATA PROCESSING
NON-SUPERVISORY JOB CLASSIFICATIONS
NOTE: Not all possible career paths are indicated.



TYPICAL ORGANIZATION STRUCTURE OF A MEDIUM OR LARGE DATA PROCESSING



ORGANIZATION CHART



APPENDIX C. COMPUTER OPERATIONS I

APPENDIX C

Match the following words with their function or explanation:

a.	Read	1.	Movement of the disk arm to find a certain record.
b.	Write	2.	Recording data onto the disk pack.
c	Seek	3.	Successive layers on the disk pack which permit processing without the movement of the access arm
d.	Track	4.	Placing or obtaining records in an ordered way
e	Random access	5.	A concentric circle on the disk pack
f	Sequential	6.	Transferring data from the disk pack.
g	Cylinder	7.	Placing or obtaining records in an unordered way

Answers

1 c
2 b
3 g
4 f
5 d
6 a
7 e

Fill in the blank that completes the statement.

1. The _____ appears at the end of the tape file and usually contains the count of the number of records.

2. The _____ prevents the tape from running off the tape reel.

3. The organization of a tape file into several logical records with a gap between them is known as a _____.

4. The ability to store information onto a tape file in terms of 800 or 1600 characters to the inch is known as tape _____.

5. The ability of the tape drive to thread tape through the transport automatically is an indication that the tape drive is _____.

Answers

1. Trailer label
2. Reflector indicator
3. Block
4. Density
5. Self-loading

Fill in the blank that completes the statement.

1. The device that removes one strip of magnetically coded data and places it on a drum for processing is known as a _____.

2. The device that has a fixed read/write head positioned over each track is known as a _____

3. Because the read/write arm does not need to be moved, the speed of processing is considered to be fast on the _____.

4. Approximately 400 million characters can be stored on the _____.

Answers

1. Data cell
2. Drum
3. Drum
4. Data cell

Match a list of manufacturer- and user-written programs with a descriptive statement.

a. Compiler	1. The manufacturer-supplied program that is used to operate the system
b. Utility program	2. The English-like language that is written by the programmer
c. Source program	3. The translator that converts the programmer's language to the computer's language
d. COBOL	4. A tape-to-printer program
e. Supervisor	5. Any language that is written by the programmer
f. Library programs	6. Usually small programs supplied by the computer manufacturer that are stored on the disk

Answers

1. e
2. c and d
3. a
4. b
5. c
6. f

APPENDIX D. COMPUTER OPERATIONS II

APPENDIX D

IPL PROCEDURE (START THE SYSTEM)

STEP	PROCEDURE	ACTION	COMMENTS
1	Mount disk pack (system) on disk drive. Ready the device.		
2	Place job control statements in SYSRDR. Ready the device.		
3	Dial load switches to address (unit) of disk drive.		
4	Press LOAD button.		IPL and Supervisor loaded: System is in WAIT state.
5	Press REQUEST button.	Message: GIVE IPL CONTROL STATEMENTS	
6	Enter DEL statements (if necessary). Enter ADD statements (if necessary).		To delete devices from PUB Table. To add devices to PUB Table.
7	Enter SET command.	SET DATE = mm/dd/yy, CLOCK = hh/mm/ss Message: DOS IPL COMPLETE	System requires operator to set date and clock. Control is given to Job Control and Supervisor.

IPL USING CONSOLE KEYBOARD

APPENDIX D (Cont'd.)

Match the following words (terms) with their explanation. Not all words are used.

a. Burst mode	1. The physical connection that is made between the CPU and its input/output devices
b. Byte mode	2. The amount of data that is usually moved between internal storage and a register
c. Interface	3. A character that has been stored after the zone portion has been removed
d. Full word	4. A shorthand method of writing in binary
e. Hexadecimal	5. The movement of large amounts of information
f Packed decimal	6. Stopping the computer under abnormal conditions will usually generate this action.
g. Base registers	
h. Half word	
i. Program status word	

Answers

1. c
2. d
3. f
4. e
5. a

(Teacher Key)

PROGRAM STATE	CPU FUNCTIONING	STATUS INDICATION	HOW SWITCHED
Stopped	Incapable of any function	"Manual" light on console	"Stop" key on console
Operating	Capable of executing instructions and being interrupted	"System" or "Wait" on console	"Start" key on console
Running	Instruction fetching and execution proceed normally	A zero bit in position 14 of the program status word (PSW) "System" light on console	I-O interrupt External interrupt
Waiting	No instruction processing. I-O and external interrupts accepted unless masked. Timer is up[illegible]	A one bit in position 14 of the program status word (PSW) "Wait" light on console	Load PSW instruction Any interrupt
Supervisor	All instru[illegible] are valid.	A zero bit in position 15 of the PSW	Any interrupt
Problem	All I-O instructions and a group of control instructions are invalid.	A one bit in position 15 of the PSW	Load PSW instruction
Masked	I-O, External, and Machine-check interrupts (individually masked) remain pending. Program interrupts are ignored.	Zero bits in the system mask, program mask, and machine-check mask fields of the PSW	Set Program Mask instruction Set System Mask instruction Load PSW instruction Any interrupt
Interruptible	Interrupts of all unmasked classes accepted.	One bits in the system mask, program mask, and machine-check mask fields of the PSW	Same as "Masked" above

(Student Handout)

Write short statements to explain the effect of each program state.

PROGRAM STATE	CPU FUNCTIONING	STATUS INDICATION	HOW SWITCHED
Stopped			
Operating			
Running			
Waiting			
Supervisor			
Problem			
Masked			
Interruptible			

COMPUTER ROOM CHECK LIST

Physical facilities

____ Main power switches on

____ Computer room temperature adequate

____ Computer room humidity adequate

____ Room clean

____ Storage areas unlocked

Power to computer and peripherals

____ Computer console ready

____ Card reader ready

____ Printer ready

____ Tape drive ready

____ Disk drive ready

____ Other

Tapes and disk packs

____ Scratch tapes mounted

____ System residence pack mounted

Initialize the system

____ Initial program load

____ Set clock and date

____ Establish partitions and mode of processing

Job processing (Input/output control department)

____ Job transmittal sheets

____ Cards for processing

____ Paper forms and carriage control tapes

____ Tape reels

____ Disk packs

Job processing (Peripherals)

____ Proper cards in reader

____ Card reader made ready

____ Proper tapes mounted and loaded

____ Proper disk packs mounted

____ Proper forms mounted on printer

____ Proper carriage control tape mounted

Teacher's Notes:

The following seven pages are forms that are used in the MCPS central office data processing installation as part of the operator's job documentation.

DAILY MACHINE SCHEDULE

System: _____

Date: ___/___/___ Revision: _____

Process Time	Job Number	Step Number	Line	Action Description	Flow To	Actual Start Time	Total Process Time	Op. No.	COMMENTS	

084 SORTER STEP INSTRUCTION SHEET

ACTION _____ Date: ___ / ___ / ___

		Job No.	
		Step No.	Time Est.
USER NAME	Phone Ext.	Job Name	
File Input Name -		Card Volume	

Sort #1	Field Name	Columns	Alphabetic / Alphanumeric / Numeric

OPERATING INSTRUCTIONS:

Cards in order before Sort? Yes ( ) No ( ) If Yes, indicate sequence.

Rejects expected? Yes ( ) No ( ) If any, what action?

Sort #2	Field Name	Columns	Alphabetic / Alphanumeric / Numeric

OPERATING INSTRUCTIONS:

Rejects expected? Yes ( ) No ( ) If any, what action?

Sort #3	Field Name	Columns	Alphabetic / Alphanumeric / Numeric

OPERATING INSTRUCTIONS:

Rejects expected? Yes ( ) No ( ) If any, what action?

Sort #4	Field Name	Columns	Alphabetic / Alphanumeric / Numeric

OPERATING INSTRUCTIONS:

Rejects expected? Yes ( ) No ( ) If any, what action?

SPECIAL INSTRUCTIONS

Disposition of Cards: _____

APPENDIX D (Cont'd.)

CLERICAL Step Instruction Sheet

ACTION _____ Date: _____

		Job No.	
		Step. No.	Time Est.
USER NAME	Phone Ext.	Job Name	
REPORT HEADING		As of Date	

SPECIAL INSTRUCTIONS

APPENDIX D (Conf'd.)

SPECIAL STEP INSTRUCTION SHEET

Action: _____ Date: ___/___/___

	Job No.:	Step No.:	Time Est.:
	Job Name:		

SYSTEM RUN SHEET

SYSTEM RUN SHEET

Job Number	Est. Time	Class	Priority	Run Date
Job Name		Job Subname		

DSNAME	SOURCE	DISPOSITION/RETENTION

APPENDIX D (Cont'd.)

WORK ROUTE SHEET

Date: / / Due Date. / /

	Job No.	
	Status No	Status Name
Job Name	Job Subname	

Step No.	Actuop By	Act. Class	Est. Time	Date Compl.	Operator Init.	Operator No	Bin	Remarks

COMMENTS

APPENDIX D (Cont'd.)

JOB TROUBLE REPORT

To

Job Number	Job Name	
Step Number	Program Number	Production ☐ Test ☐
Operator Name		Submitted By
		TOTAL TIME LOST

PROBLEM AREA

A. ☐ INSTRUCTIONS
B. ☐ INPUT DATA
C. ☐ PROGRAM PROBLEMS
D. ☐ CONTROL CARDS
E. ☐ CARRIAGE TAPE
F. ☐ REMOVED FROM SCHEDULE
G. ☐ JOB RUNNING UNREASONABLY PAST TIME ESTIMATE
H. ☐ ABNORMAL JOB TERMINATION
I. ☐ OTHER (Specify Below)

IN ALL CASES OF JOB TERMINATION DUE TO APPARENT SYSTEMS OR PROGRAMMING PROBLEMS PLEASE DO THE FOLLOWING.

1. Check one of the above and explain in detail below.
2. Provide a storage dump if the System doesn't provide a dump.
3. Provide a console typeout (carbon copy), and any printout.
4. Hold all tape and disk.

COMMENTS

HARDWARE MALFUNCTION REPORT

Date _____

Device Type and/or Address _____

_____

Time C.E. Called _____

Did C.E. Call? ☐ Yes ☐ No Time _____

Date and Time C.E. Arrived _____

Problem Resolved? ☐ No ☐ Yes Time Resolved _____

Symptom _____

_____

_____

If Not Resolved _____

_____

_____

Problem Reported By _____

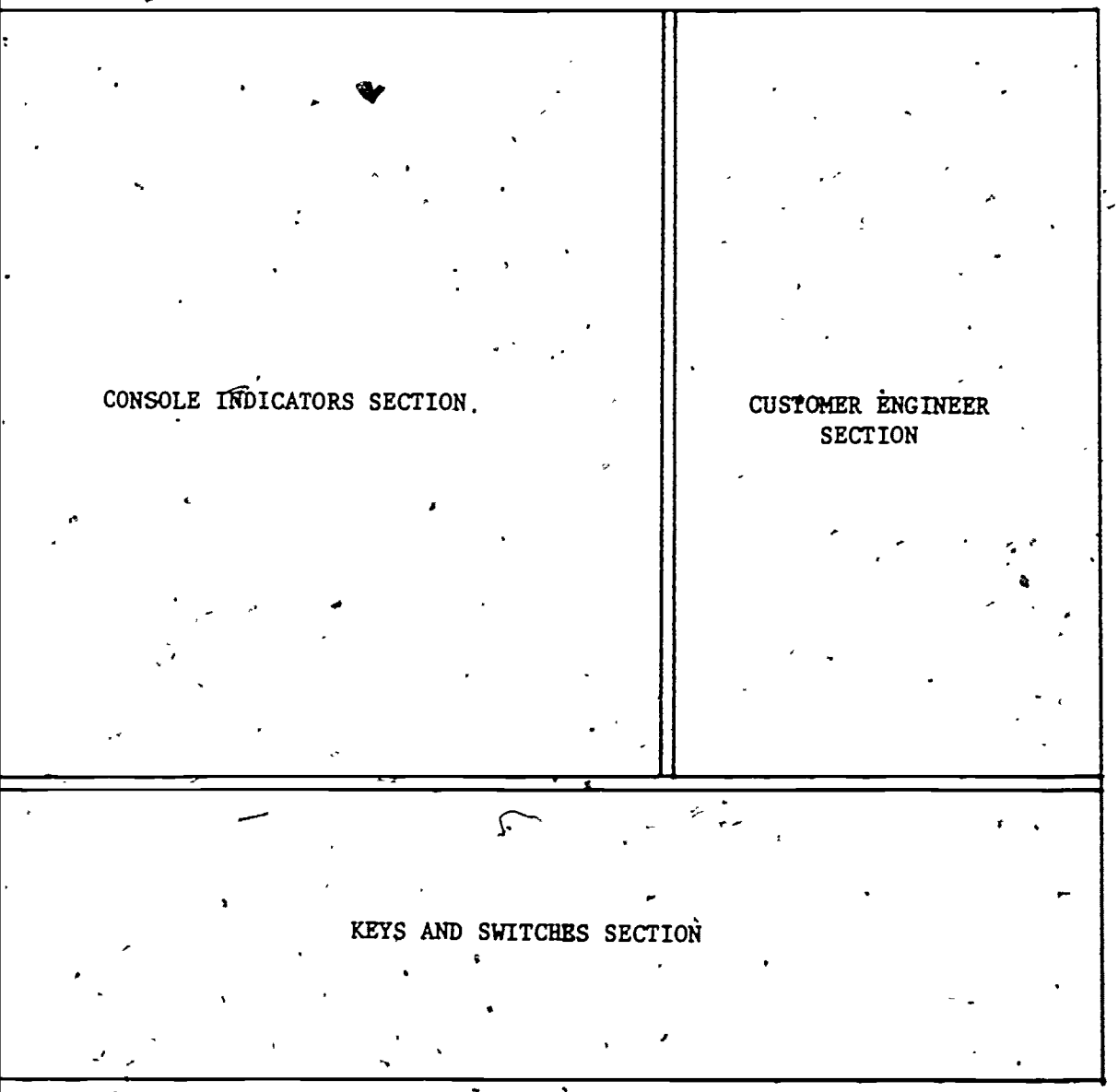
(Division of Data Processing Operations Use Only)

APPENDIX D

(Teacher Key)

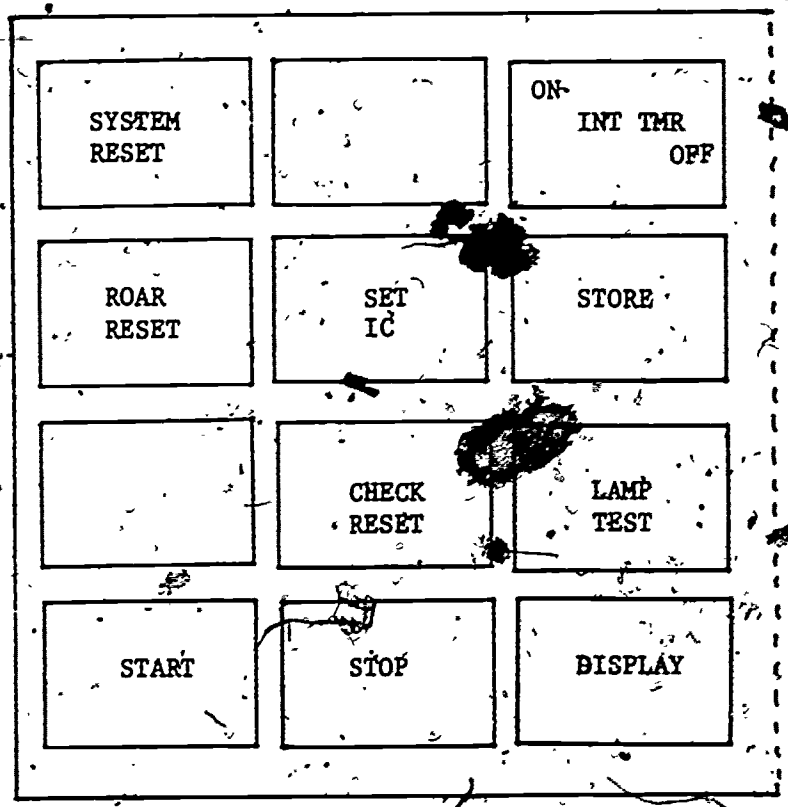
Reference: *Introduction to Computer Operations* by Fuori/D'Arco/Orilia

IBM S/360 CENTRAL PROCESSING UNIT CONSOLE

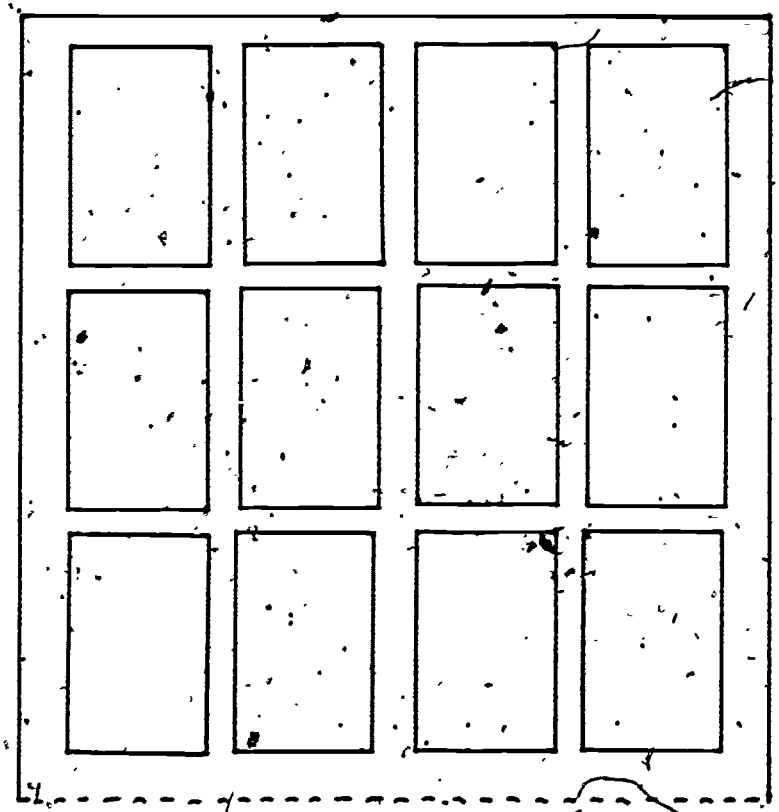


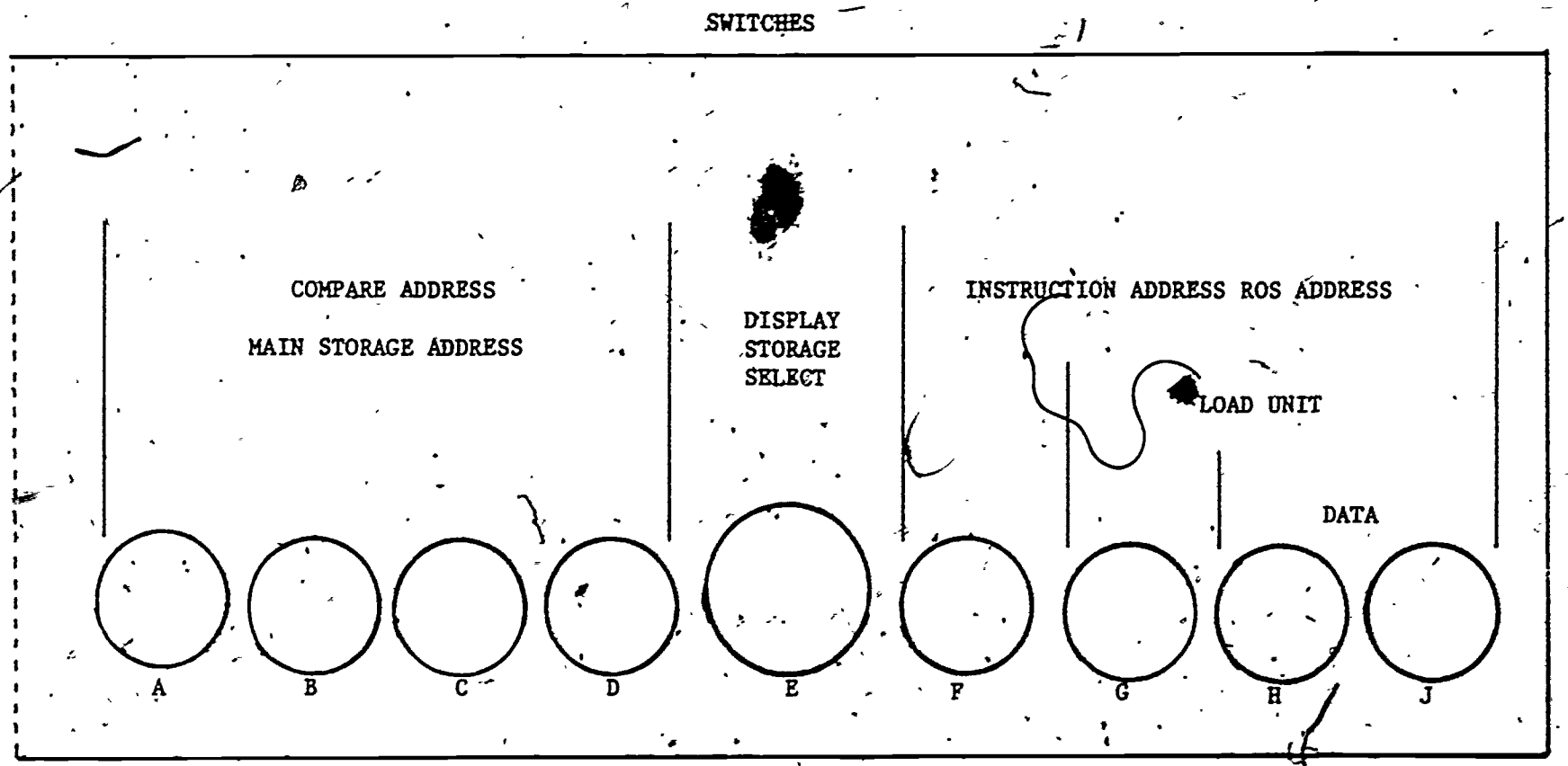
(Student Handout)

(Teacher Key)

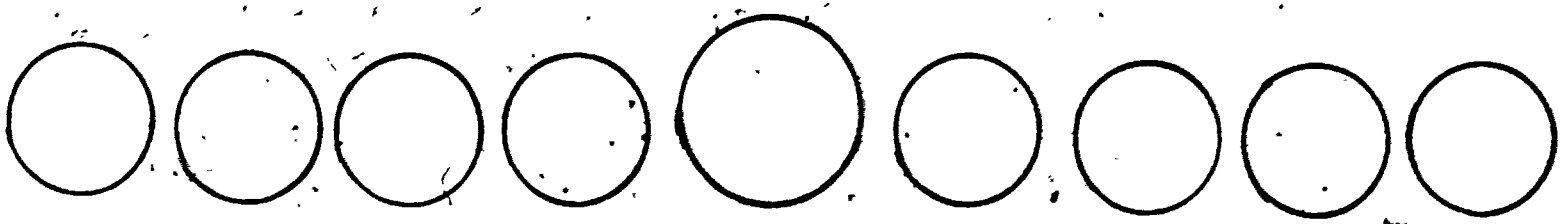

SYSTEM RESET
ON
INT TMR
OFF
ROAR RESET
SET IC
STORE
CHECK RESET
LAMP TEST
START
STOP
DISPLAY

(Student Handout)




SWITCHES
COMPARE ADDRESS
MAIN STORAGE ADDRESS
DISPLAY
STORAGE
SELECT
INSTRUCTION ADDRESS ROS ADDRESS
LOAD UNIT
DATA
A
B
C
D
E
F
G
H
J

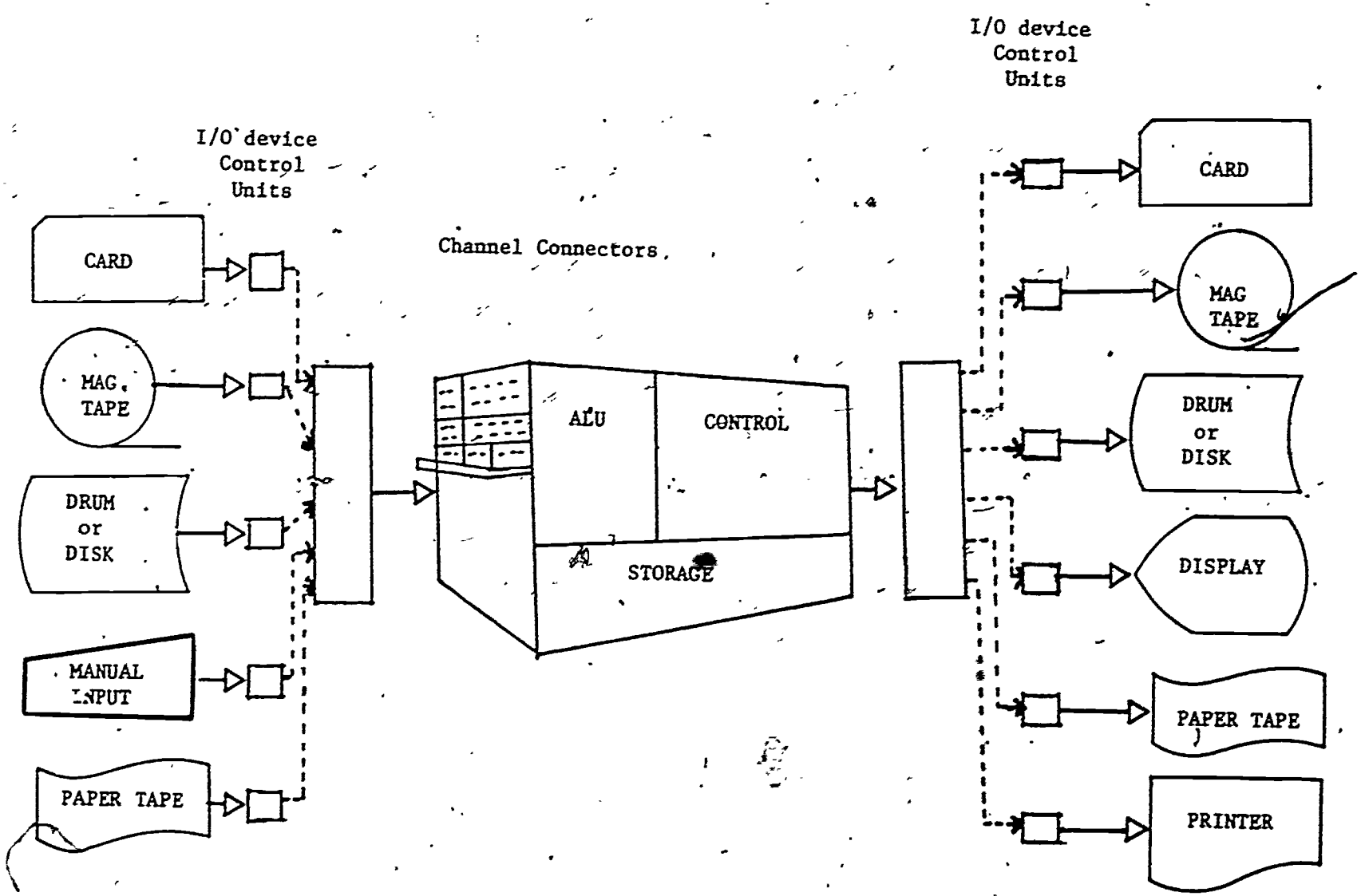
(Student Handout)



D-21

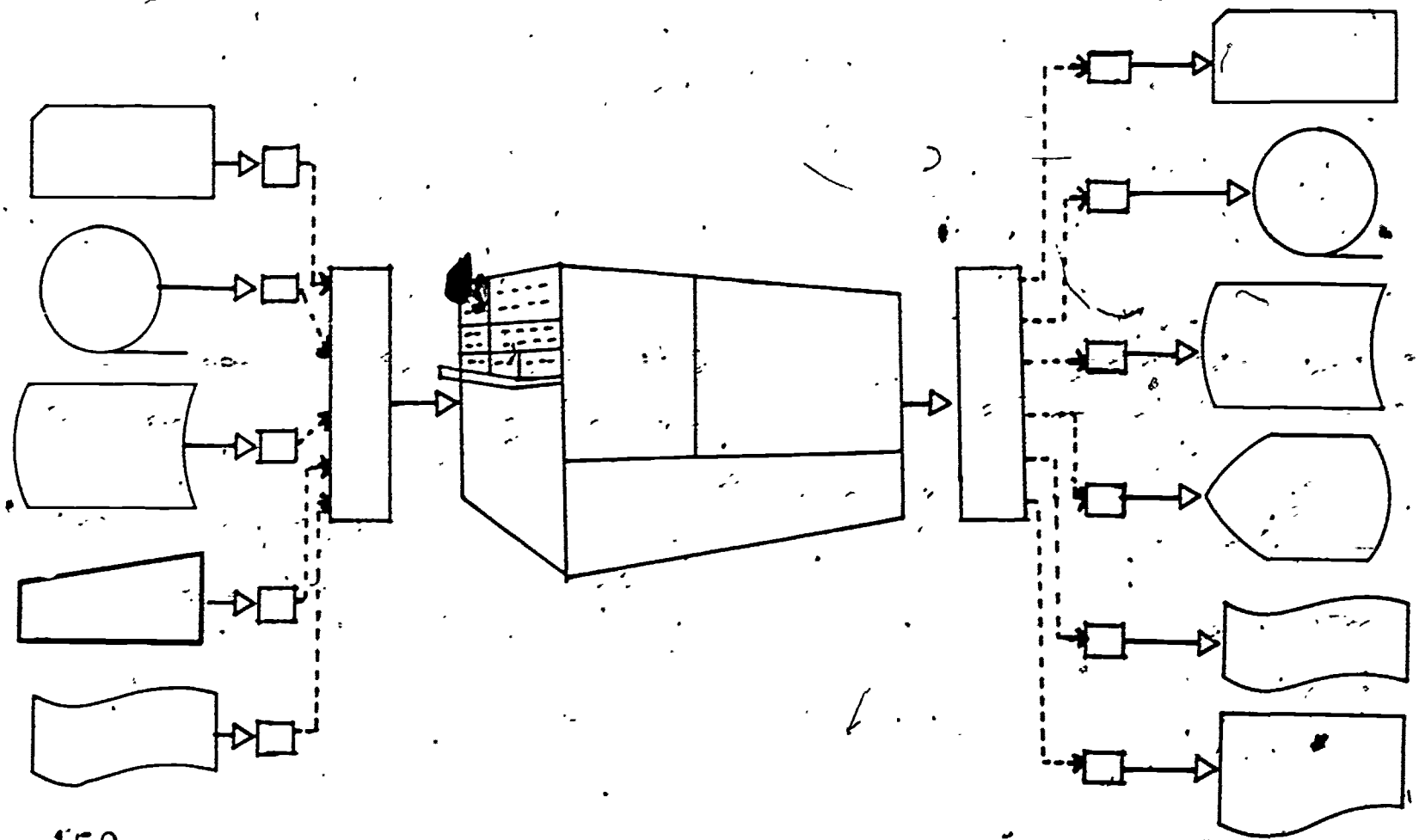
Label all components of the total computer system showing names of input and output devices, the location of the I O device control units, the channel connectors, and components of the CPU.

(Teacher Key)



Label all components of the total computer system showing names of input and output devices, the location of the I O device control units, the channel connectors, and components of the CPU.

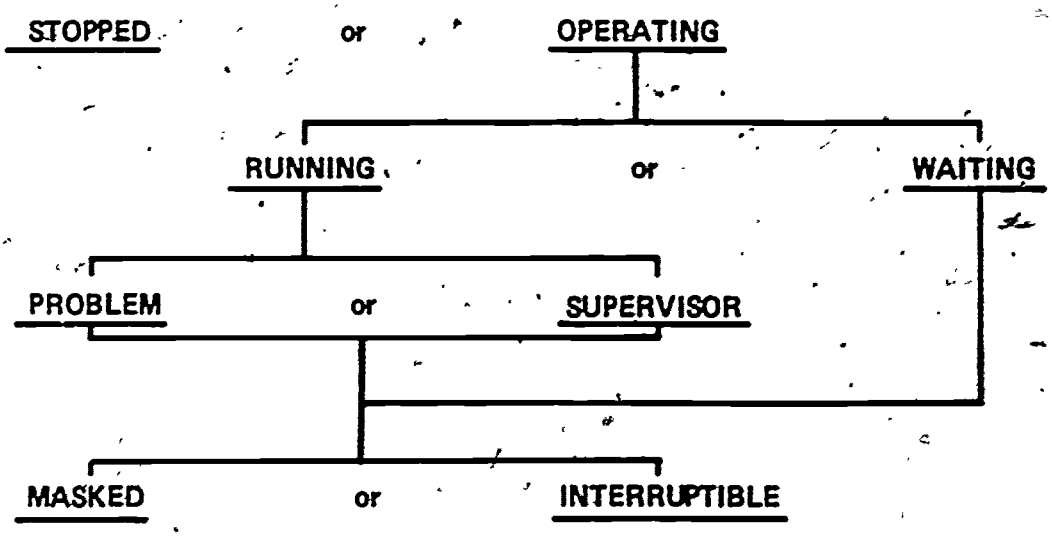
(Student Handout)



(Teacher Key)

IBM S/360 PROGRAM STATES

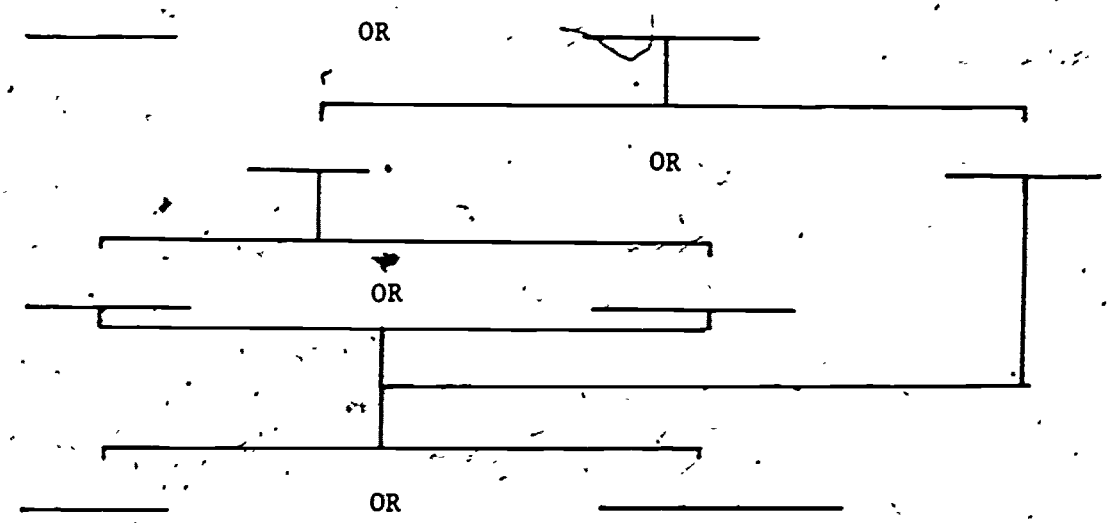
STOPPED	or	OPERATING
RUNNING	or	WAITING
SUPERVISOR	or	PROBLEM
MASKED	or	INTERRUPTIBLE



(Student Handout)

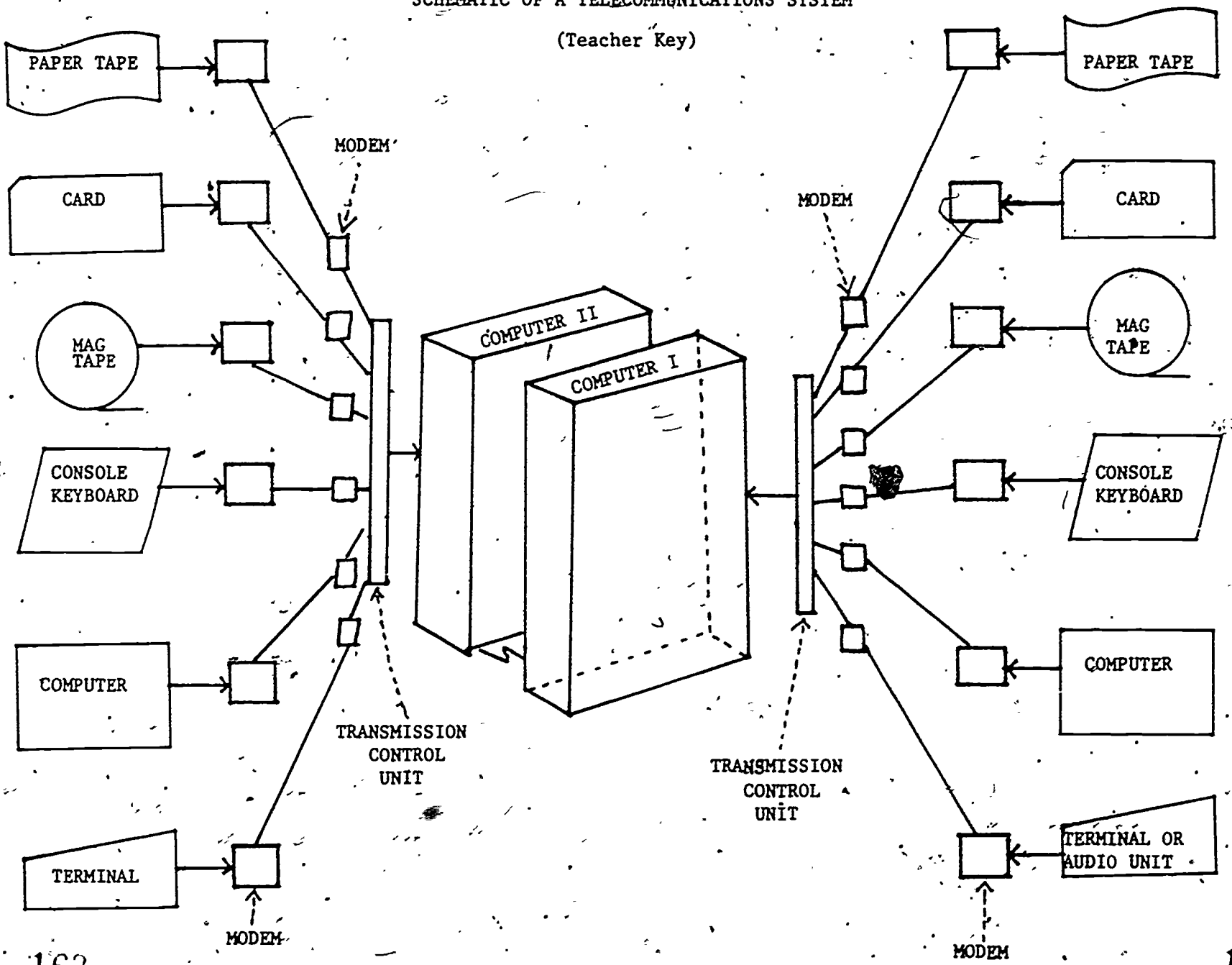
IBM S/360 PROGRAM STATES

Write the names of the different program states on the short horizontal lines.



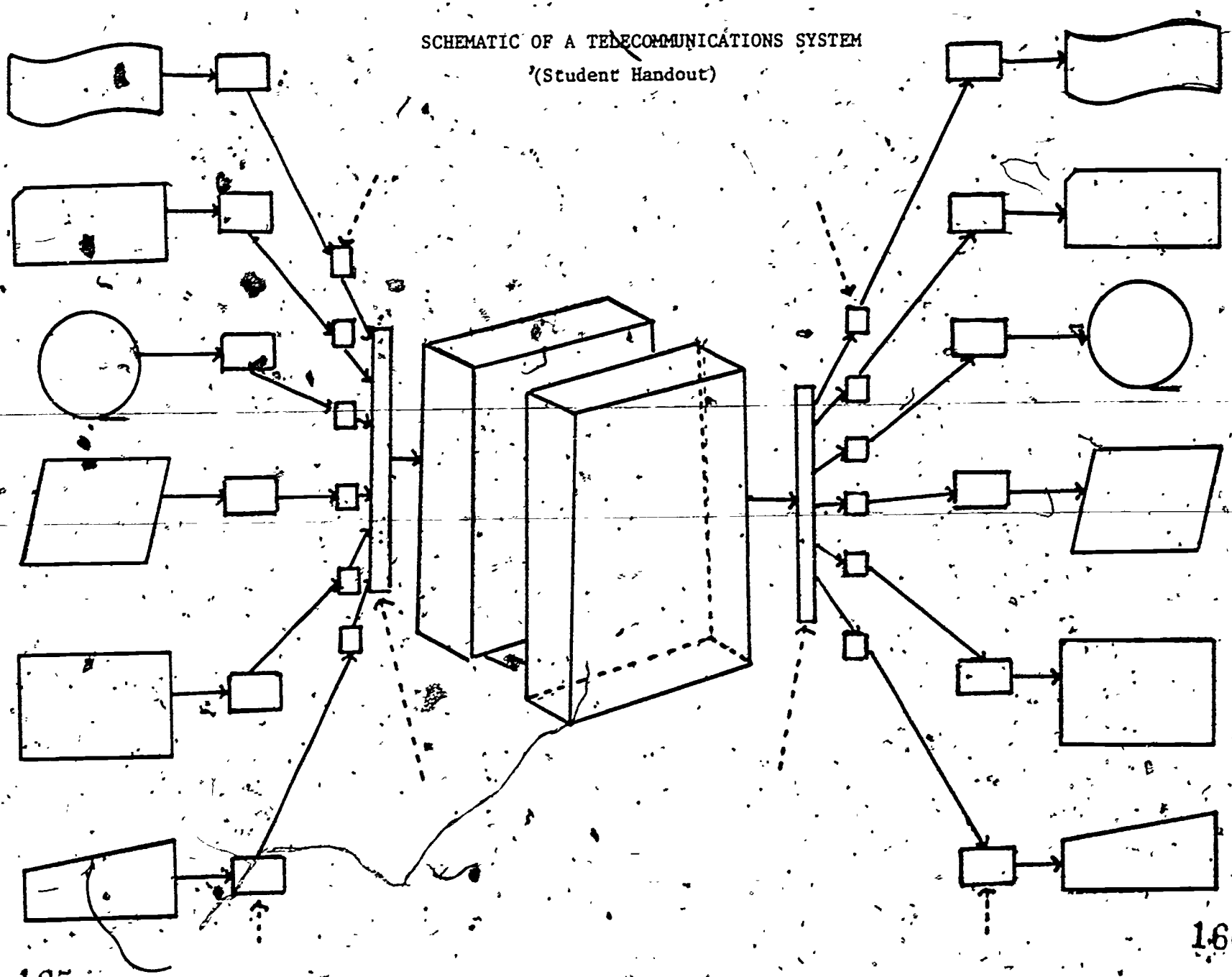
SCHEMATIC OF A TELECOMMUNICATIONS SYSTEM

(Teacher Key)



SCHEMATIC OF A TELECOMMUNICATIONS SYSTEM

(Student Handout)



Match a list of diagnostic messages printed on the console typewriter and a group of interactions by the operator. Not all interactions are used.

INTERACTION	MESSAGE
a) Non-recoverable error. Type the verb CANCEL on the console.	1. Set command not given
b) Punch unit not ready. Run out cards, ready the punch unit, and type RETRY on console.	2. Bg OPO8 A interv req
c) Type the command on the console to establish the date and clock.	3. Error has occurred during linkage editing.
d) No action. System will cancel the job.	4. Cat name not in library
e) User program not found in library. Type IGNORE on console.	5. Punch check-outcard
f) Background is indicating that a divice is not ready. Ready the device.	

Answers:

1. c
2. f
3. d
4. a
5. b

IBM DISK OPERATING SYSTEM

CONTROL PROGRAMS

1. TASK MANAGEMENT

(Supervisor)

2. JOB MANAGEMENT

(Job Control)

3. DATA MANAGEMENT

(IOCS)

PROCESSING PROGRAMS

1. LANGUAGE TRANSLATORS

(Source Code -- High Level)

2. SERVICE PROGRAMS

(Librarian and Utilities)

3. USER-WRITTEN PROGRAMS

APPENDIX E. INTRODUCTION TO REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

APPENDIX E

PREFACE

This program was designed to be used by the teacher and the class to discuss each step of the programming cycle.

Problem Statement

Using a Student Name file of punched cards prepared by the computer at the end of the last school schedule run, sort the cards on the First Name field within the Last Name field within the Class Period field within the Course Number field.

The input record is made up of the following fields:

Last Name	1-15	Identification Number	42-47
First Name	16-25	Course Number	48-52
Middle Initial	26	Class Period	53
Grade	40-41	Sex	80

The output record is to be a printed class list including the following:

HEADING

Name of the school	Teacher name	CLASS LIST
Name of the school system	Course number	Column headings
Name of the city and the state in which the school is located	Course name	

DETAIL

Identification number Grade

Student name — Last, First, Middle Sex

TOTAL

Footing: NUMBER OF STUDENTS

Total number of students in the class

Write, execute, and document a program to print the Class List for a Data Processing class 5370. Use all the steps in the programming cycle.

JOB: CLASS LIST

A printed report is to be developed listing the student name, identification number, grade, and sex of each student in a class — Data Processing, 5370. The page is to have headings including: name of school, school system, city and state in which the school is located, teacher name, course name and number, and column headings for the listed information.

At the end of each Class List, there is to be a Total line containing the footing: NUMBER OF STUDENTS and the total number of students enrolled in the class.

Test data will come from the Student Name input file.

The name of the Input File is STUDENT and the name of the Output File is LIST.

This program is to be compiled and executed on an IBM S/370, using the Report Programming Generator (RPG) language.

State in sequence the nine steps in the Programming Cycle.

Answer.

1. Read the problem statement.
2. Organize the facts and prepare the problem specifications.
3. Develop the problem solution — flowchart or decision table.
4. Write the narrative.
5. Code the source program.
6. Punch the program.
7. Test the program.
8. Debug the diagnostic errors.
9. Organize the documentation.

SAMPLE

REPORT PROGRAM GENERATOR-FILE DESCRIPTION SPECIFICATIONS
System/360

Date: 4/20/7-
Program: CLASS LIST
Programmer: SAM SMITH

Punching Instruction: Graphic ØOI; Punch HAA

Page: 1 2

Program Identification: 75 76 77 78 79 80

Line	Form Type	Filename	File Type	File Designation	End of File	Sequence	File Format	Block Length	Record Length	Mode of Processing	Length of Key Field or of Record Address Field	Record Address Type	Type of File Organization	Overflow Indicator	Key Field Starting Location	Extension Code E/L	Device	Symbolic Device	Labels (S, N, or E)	Name of Label Exit	Extent Exit for DAM	File Addition	No. Tracks for Cylinder Overflow	No. of Extents	Tape Ignored
0 1	H																								
0 2	F	*CLASS LIST APPLICATION																							
0 3	F	*																							
0 4	F	*SAM SMITH																							
0 5	F	*																							
0 6	F	*DESIGNED TO PRINT A LIST AND TO COUNT THE STUDENTS IN A CLASS																							
0 7	F	*																							
0 8	F	*APRIL 20, 197-																							
0 9	F	*																							
1 0	F	STUDENT IP					80	80									READ40								
1 1	F	LIST	O				132	132					OF				LPRINTER								
1 2	F																								
1 3	F																								
1 4	F																								
1 5	F																								
	F																								
	F																								
	F																								
	F																								
	F																								

Card Electro Number

SAMPLE

REPORT PROGRAM GENERATOR - INPUT SPECIFICATIONS
System/360

Date 4/20/7-

Program CLASSLIST

Programmer SAM SMITH

Punching Instructions: Graphic 0 0 1 6 ; Punch H A A H 6 2 A H

Page 02

Program Identification

Line	Form Type	Filename	Sequence	Number (1-N)	Option (O)	Resulting Indicator	Record Identification Codes 1 Position	Not (N)	C/Z/D	Character	2 Position	Not (N)	C/Z/D	Character	3 Position	Not (N)	C/Z/D	Character	Stacker Select	Packed (P)	Field Location From	To	Decimal Positions	Field Name	Control Level (L1-L9)	Matching Fields or Chaining Fields	Field-Record Relation	Field Indicators Plus	Minus	Zero or Blank	Starting Sign Position
01	I	STUDENT AB				18	8 HC																								
02	I																				1	15		LAST							
03	I																				16	25		FIRST							
04	I																				26	26		MIDDLE							
05	I																				40	41		GRADE							
06	I																				42	47		IDNO							
07	I																				80	80		SEX							
08	I																														
09	I																														
10	I																														
11	I																														
12	I																														
13	I																														
14	I																														
15	I																														
	I																														
	I																														
	I																														
	I																														
	I																														

SAMPLE

MONTGOMERY COUNTY PUBLIC SCHOOLS
Rockville, Maryland

REPORT PROGRAM GENERATOR - CALCULATION SPECIFICATIONS —
System/360

Date 4/28/7-

Program CLASS LIST

Programmer SAM SMITH

Punching Instruction							
	Graphic	0	8	1			
	Punch	A	H	H			

Page 03

Program Identification

Line	Term Type	Control Level (10 or 18)	Indicators			Factor 1	Operation	Factor 2	Result Field	Field Length	Decimal Positions	Half Adjust (H)	Resulting Indicators: Plus	Minus	Zero or Blank	Compare: High 1 > 2	Low 1 < 2	Equal 1 = 2	Comments
			And	And															
			2nd	2nd	2nd														
3 4 5	6	7 8	9 10 11	12 13 14	15 16 17	18 19 20 21 22 23 24 25 26 27	28 29 30 31 32	33 34 35 36 37 38 39 40 41 42	43 44 45 46 47 48	49 50 51	52	53	54	55	56	57	58	59	60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
01	C		18			+1,8	ADD	TOTAL	TOTAL	3	8								
02	C																		
03	C																		
04	C																		
05	C																		
06	C																		
07	C																		
08	C																		
09	C																		
10	C																		
11	C																		
12	C																		
13	C																		
14	C																		
15	C																		
	C																		
	C																		
	C																		
	C																		
	C																		

APPENDIX E (Cont'd.)

Card Electro Number

SAMPLE

REPORT PROGRAM GENERATOR - OUTPUT FORMAT SPECIFICATIONS
System/360

Date 4/20/7-
Program CLASSLIST
Programmer SAM SMITH

Punching Instruction: Graphic 1 0; Punch H H A

Page 04

Program Identification

Line	Form Type	Filename	Type (H/D/T)	Space Before	Space After	Skip Before	Skip After	Output Indicators	Field Name	Zero Suppress (Z)	Blank After (B)	End Position in Output Record	Packed Field (P)	Constant or Edit Word	Starting Sign Position
01	O	LIST	H		1	01		1P							
02	O		OR		1	01		OF							
03	O											22		'HOMETOWN HIGH SCHOOL'	
04	O											77		'TEACHER JANE DOE'	
05	O		H		1			1P							
06	O		OR		1			OF							
07	O											25		'SOMEWHERE COUNTY PUBLIC'	
08	O											33		'SCHOOLS'	
09	O											43		'COURSE: DATA PROCESSIN'	
10	O											44		'G'	
11	O		H		1			1P							
12	O		OR		1			OF							
13	O											25		'ANYWHERE, U.S.A. [illegible]'	
14	O											73		'NO: 5370'	
15	O		H		2			1P							
	O		OR		2			OF							
	O											47		'CLASS LIST'	
	O														

SAMPLE

REPORT PROGRAM GENERATOR - OUTPUT FORMAT SPECIFICATIONS
System/360

Date 4/20/7-
Program CLASS LIST
Programmer SAM SMITH

Punching Instruction: Graphic 1 0 3 2 ; Punch H A H A

Page 0 5

Program Identification

Line	Form Type	Filename	Type (N/O/T)	Space Before	Space After	Skip Before	Skip After	Output Indicators	Field Name	Zero Suppress (Z)	Blank After (B)	End Position in Output Record	Packed Field (P)	Constant or Edit Word	Sterling Sign Position
01	O		H		2			1P							
02	O		OR		2			OR							
03	O											15		'ID NO.'	
04	O											37		'STUDENT NAME'	
05	O											74		'GRADE SEX'	
06	O		O		1			18							
07	O					4			IDNO	Z		15			
08	O								LAST			34			
09	O								FIRST			49			
10	O								MIDDLE			53			
11	O								GRADE	Z		66			
12	O								SEX			73			
13	O		T	2				1A							
14	O							3	TOTAL	Z		35			
15	O											29		'NUMBER OF STUDENTS'	
	O														
	O														
	O														

MULTIPLE-CARD LAYOUT FORM

Company [illegible]

Application [illegible] by [illegible] Date 4/20/[illegible] Job No. [illegible] Sheet No. 2

LAST NAME	FIRST NAME	MI	GRADE	ID. NO.	COURSE NUMBER	PERIOD	SEX

HOMETOWN HIGH SCHOOL
SOMEWHERE COUNTY PUBLIC SCHOOLS
ANYWHERE, U. S. A. 00000

TEACHER: JANE DOE
COURSE: DATA PROCESSING
NO: 5374

CLASS LIST

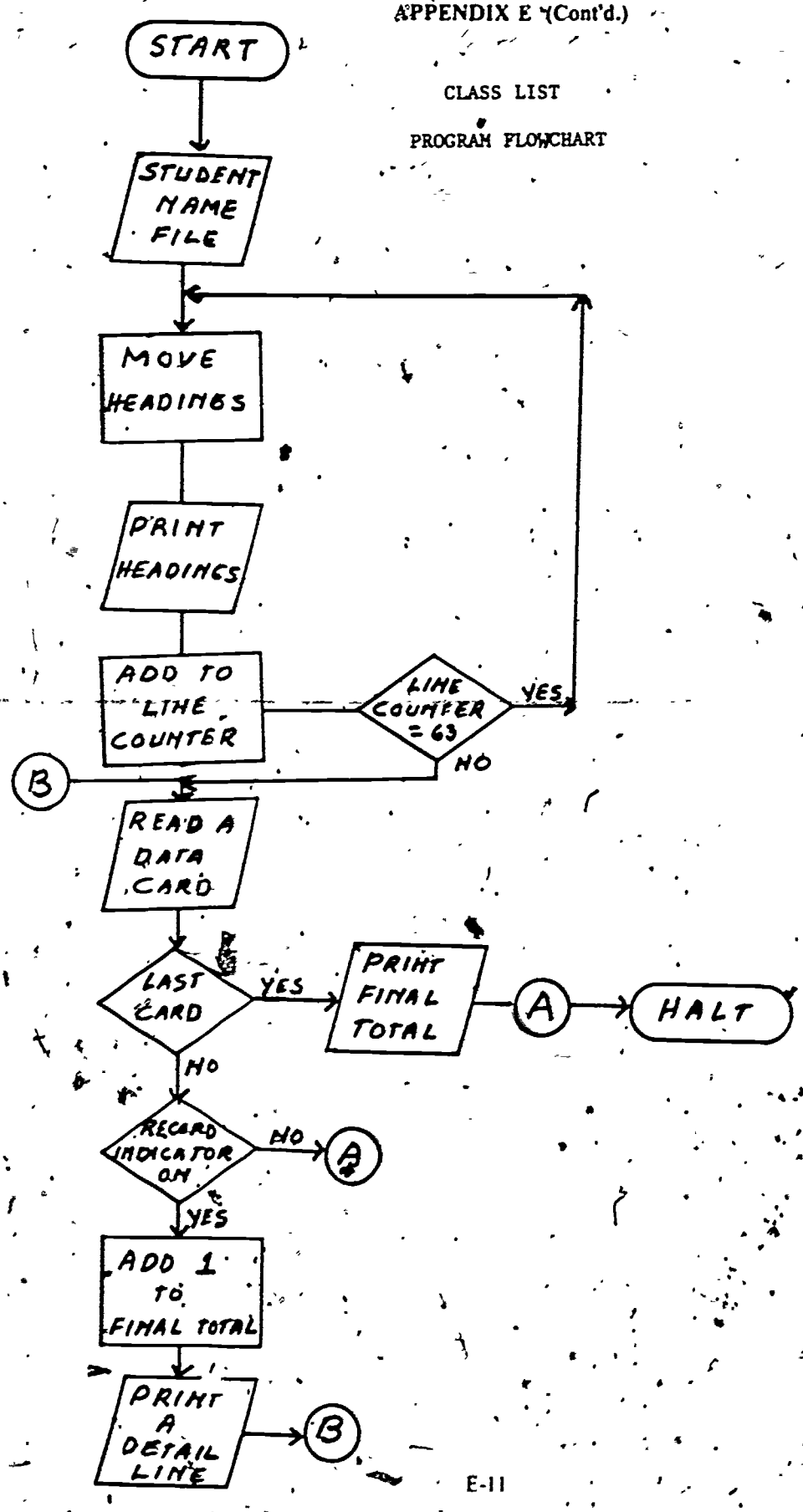
ID NO.	STUDENT NAME	GRADE	SEX
XXXXXX	XXXXXXXXXXXXXXXXXXXX	XX	X
XXXXXX	XXXXXXXXXXXXXXXXXXXX	XX	X

NUMBER OF STUDENTS XX

APPENDIX E (Cont'd.)

CLASS LIST

PROGRAM FLOWCHART



Given a list of students and the schools they attend, write a program to print the student names in one column and the school names in a second column.

Sample Criterion:

- Read the problem statement.
- Organize the Printer Spacing Chart and the Record Layout.
- Develop the problem solution using a flowchart.
- Write the narrative to include the problem specification, an explanation of the problem, the logic for the solution, the equipment and the computer language to be used, the source of the test data.
- Code the instructions for the source program.
- Punch the source program, test data, and Job Control Language.
- Test the program
- Debug and execute the program.
- Organize the documentation for the application program.

Given a program assignment, including alphameric and numeric data fields, addition and editing, and the data processing forms, write a program to calculate, edit, and print.

Sample Criterion:

- Read the problem statement.
- Organize the Printer Spacing Chart and the Record Layout.
- Develop the problem solution using a flowchart.
- Write the narrative to include the problem specification, an explanation of the problem, the logic for the solution, the equipment and the computer language to be used, the source of the test data.
- Code the instructions for the source program.
- Punch the source program, test data, and the Job Control Language.
- Test the program.
- Debug and execute the program.
- Organize the documentation for the application program.

APPENDIX F. INTERMEDIATE REPORT PROGRAM GENERATOR (RPG) PROGRAMMING

APPENDIX F

Write, execute, and document the RPG program to prepare a Weekly Payroll Report. Employees are to receive the time-and-one-half for all work in excess of 40 hours.

EXAMPLE

An employee working 42 hours at $2.00 per hour would be paid for 40 hours at $2.00 per hour and 2 hours at $3.00 per hour.

INPUT

Payroll Cards Input is to consist of Payroll Cards containing the Employee Number, Employee Name, Hours Worked, Rate of Pay, and Deductions.

OUTPUT

Payroll Register Output is to consist of a Payroll Register. The report is to contain the Employee Number, Employee Name, Total Hours Worked, Regular Rate of Pay, Regular Earnings, Overtime Earnings, Total Earnings, Deductions, Net Pay (Total Earnings minus Deductions equals Net Pay). Final Totals are to be printed of Regular Earnings, Overtime Earnings, Total Earnings, Deductions, New Pay.

SAMPLE CRITERION:

- Problem Statement
- Printer Spacing Chart
- Record Layout
- Flowchart and/or Decision Table
- Narrative, including the problem specification, an explanation of the problem, the logic for the solution, the equipment and the computer language to be used, the source of the test data
- Coded instructions for the source program
- Punched source program, test data, and Job Control Language statements
- Test Program
- Debug and execute the program
- Organize the documentation for the program

Using a previously written or executed program with a level control indicator, code the instructions to double space and Group Print the report.

- Remove the detail control card from the Output Specifications.
- Punch another card, to replace the detail control card, which contains

  0 in card column(s) 6
  T in card column(s) 15
  2 in card column(s) 18 or 19
  L1 in card column(s) 24, 25

It may be necessary to remove nonrepetitive alphanumeric fields from the Input and the Output instructions in the program.

APPENDIX G. INTRODUCTION TO COBOL

APPENDIX G

PART A

List in order the four divisions of any COBOL program and describe each.

Answer

1. Identification division — identifies the source program
2. Environment division — specifies the computer that is used, and associates files with hardware devices
3. Data division — describes all data that will be considered input and output, and establishes work areas
4. Procedure division — contains the procedure commands needed to solve the problem

PART B

Problem 1

Given the following problem statement:

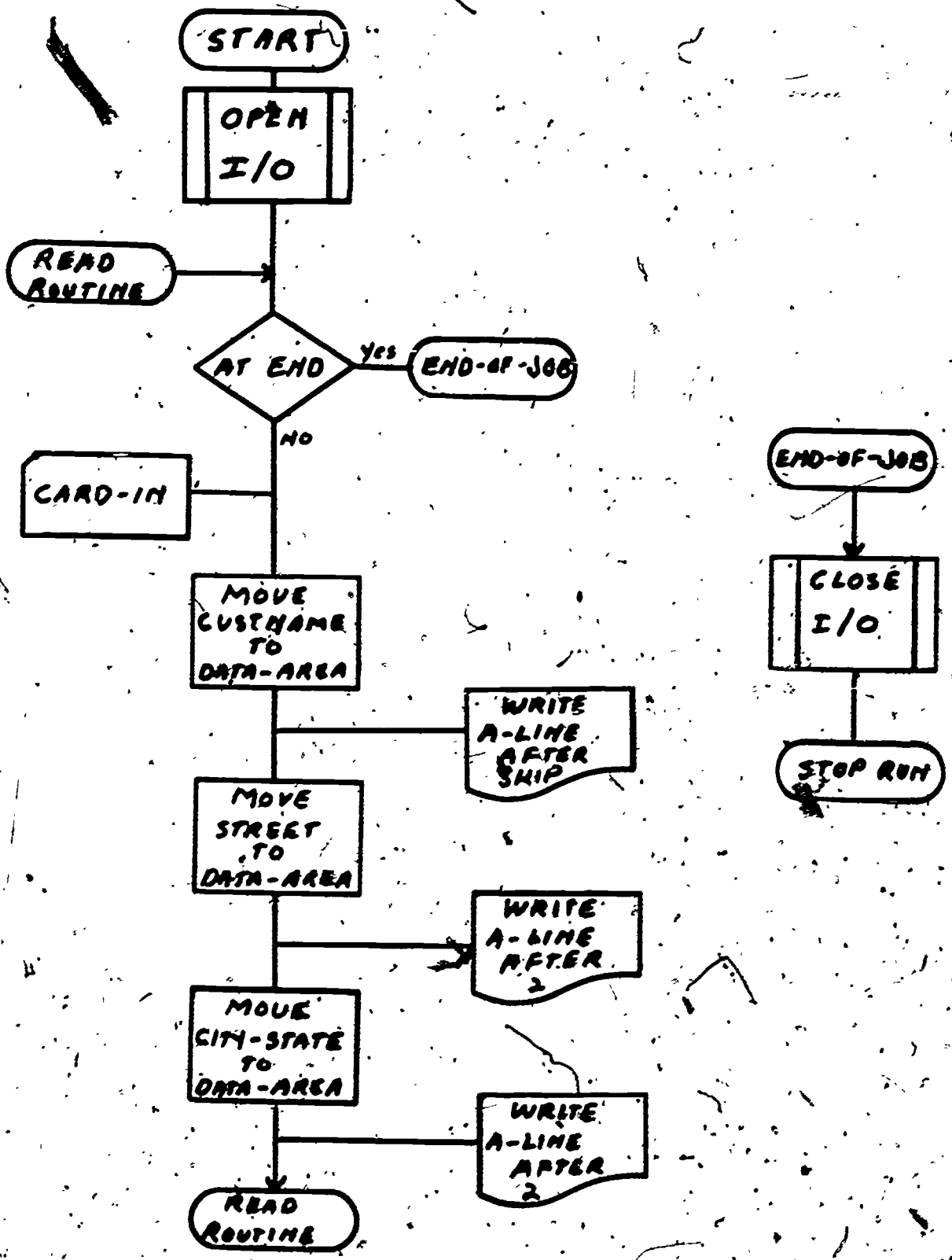
> If an employee had no raise during the last six months or earns less than $5000, provide a 5% raise otherwise no raise.

- Underscore the elements of the problem.
- Prepare the decision table.
- Test the table to determine accuracy.

Answer

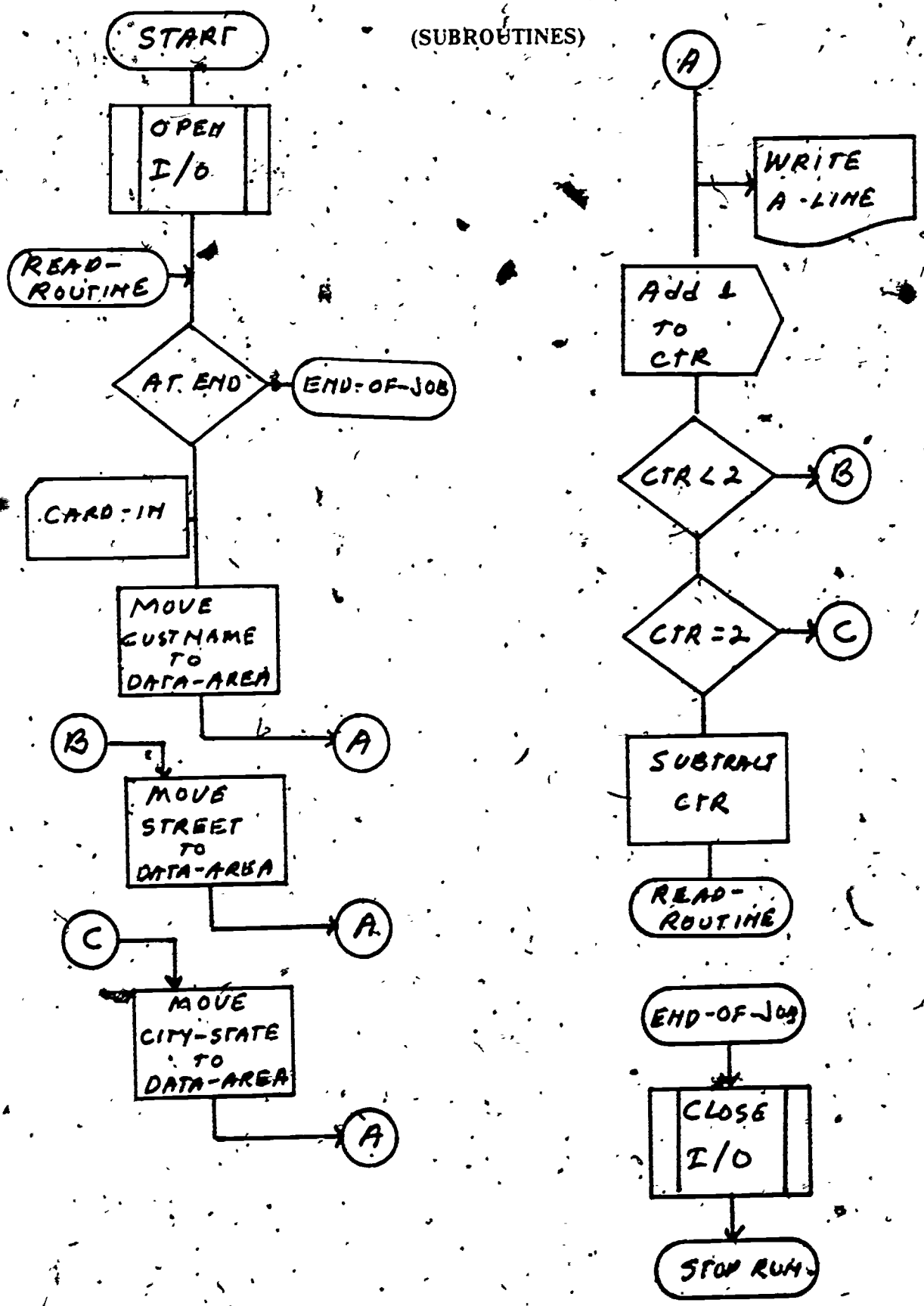
ELEMENTS	CONDITION ENTRY		
	Yes	No	No
Raise last 6 mos	x	x	
Earns at least $5,000	x		x
ACTIONS	**ACTION ENTRY**		
No raise	x		
5% raise		x	x

FLOWCHART: LABELRUN



FLOWCHART: LABELRUN

(SUBROUTINES)



Problem 2

The following is provided for a name and address card:

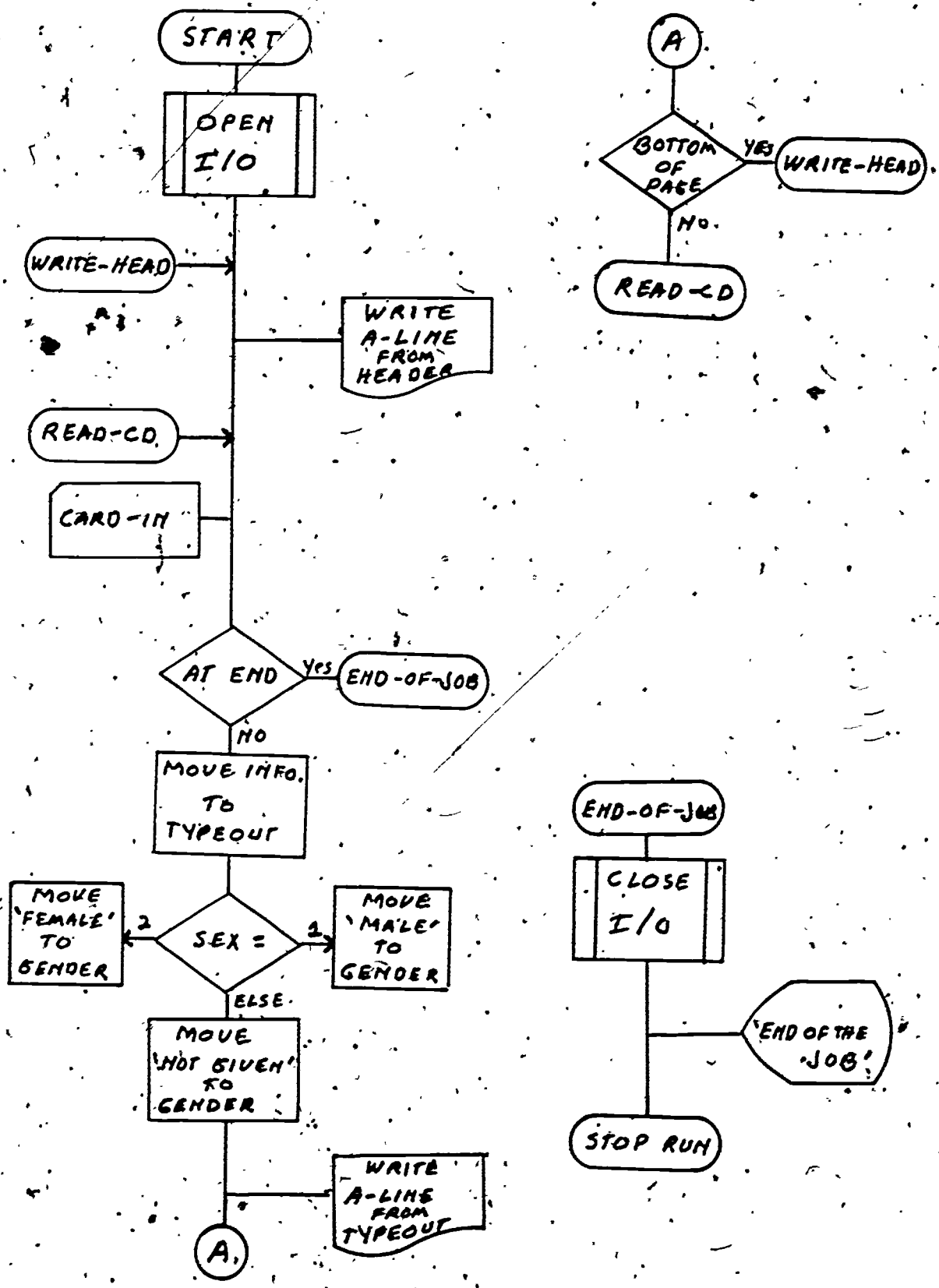
- name in card columns 1 through 30
- street address in card columns 31 through 50
- city in card columns 51 through 57
- zip code in card columns 58 through 62
- age in card columns 63 and 64
- phone number in card columns 65 through 71
- and a special code in card column 72 to represent sex:

male = 1, female = 2, and not given = any other number

The printed output will consist of a heading at the top of each page followed by detail lines, single-spaced three lines below the heading. At the end, the message *end of job* is to be printed on the computer console.

Prepare a flowchart using proper flowchart symbols for the problem. Be sure to include the heading, detail lines, all decisions, and connectors. Retain the flowchart for later use.

FLOWCHART


START
OPEN I/O
WRITE-HEAD
WRITE A-LINE FROM HEADER
READ-CD
CARD-IN
AT END
YES
END-OF-JOB
NO
MOVE INFO. TO TYPEOUT
SEX =
2
MOVE 'FEMALE' TO GENDER
1
MOVE 'MALE' TO GENDER
ELSE
MOVE 'NOT GIVEN' TO GENDER
WRITE A-LINE FROM TYPEOUT
A
A
BOTTOM OF PAGE
YES
WRITE-HEAD
NO
READ-CD
END-OF-JOB
CLOSE I/O
'END OF THE JOB'
STOP RUN

PREFACE

This program was designed to be used by the teacher and the class to discuss each step of the programming cycle and to introduce the first completely written, executed, and documented program.

PROBLEM STATEMENT

Using a customer name and address file, sort the cards into sequence on a sorter by account number, and write a program to prepare a three-line label from each of the cards.

The name and address card contains:

- the account number in card columns 1 through 6
- the customer's name in card columns 7 through 28
- the street address in card columns 29 through 50
- the city and state in card columns 51 through 72

The printed output will consist of a label using print positions 11 through 32 as follows.

- The first line at the top of the label contains the name.
- The second line, double-spaced below the first, contains the street address.
- The third line, double-spaced below the second, contains the city.

The process is to be repeated for each card until all cards have been read. Prepare a flowchart, coded instructions, sample output, and complete documentation for the problem.

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 1 of 3
Program: Problem 1		Graphic	Card Form	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page) 1–3	(Serial) 4–6	Cont. 7	A 8–11	B 12–72 (COBOL Statement)
	01		IDENTIFICATION DIVISION.	
PART	02		PROGRAM-ID. 'LABEL5'.	
A	03		AUTHOR. HOUSH.	
	04			
	05		ENVIRONMENT DIVISION.	
	06		CONFIGURATION SECTION.	
	07		SOURCE-COMPUTER. IBM-360.	
PART	08		OBJECT-COMPUTER. IBM-360.	
B	09		INPUT-OUTPUT SECTION.	
	10		FILE-CONTROL.	
	11		SELECT CARD-IN ASSIGN 'SYS005' UNIT-RECORD 2540R.	
	12		SELECT PRINTER ASSIGN 'SYS008' UNIT-RECORD 1403.	
	13			
	14		DATA DIVISION.	
	15		FILE SECTION.	
	16		FD CARD-IN,	
	17		RECORDING MODE IS F, LABEL RECORDS ARE OMITTED,	
	18		RECORD CONTAINS 80 CHARACTERS, DATA RECORD IS CD-IN.	
PART	19		01 CD-IN.	
C	20		02 ACCOUNT-NO PICTURE 9(6).	
			02 NAME PICTURE A(22).	
			02 STREET PICTURE X(22).	
			02 CITY PICTURE X(22).	
			02 FILLER PICTURE X(8).	

APPENDIX G (Cont'd.)

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 2 of 3
Program PROBLEM 1		Graphic	Card Form	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page) (Serial)	Cont.	A	B — COBOL Statement
01		FD	PRINTER, RECORDING MODE IS F, LABEL RECORDS ARE OMITTED,
02			RECORD CONTAINS 133 CHARACTERS, DATA RECORD IS A-LINE.
03		01	A-LINE PICTURE X(133).
04		WORKING-STORAGE SECTION.	
05		77	TO-A-NEW-PAGE PICTURE X VALUE '1'.
06		01	HOLD.
07			02 FILLER PICTURE X(20) VALUE SPACES.
08			02 LABEL-AREA PICTURE X(22).
09			02 FILLER PICTURE X(100) VALUE SPACES.
10			
11		PROCEDURE DIVISION.	
12		START.	
13			OPEN INPUT CARD-IN, OUTPUT PRINTER.
14		READ-CD.	
15			READ CARD-IN AT END GO TO END-OF-JOB.
16			MOVE SPACES TO A-LINE.
17			MOVE NAME TO LABEL-AREA.
18			WRITE A-LINE FROM HOLD AFTER ADVANCING TO-A-NEW-PAGE.
19			MOVE SPACES TO A-LINE.
20			MOVE STREET TO LABEL-AREA.
			WRITE A-LINE FROM HOLD AFTER ADVANCING 2.
			MOVE SPACES TO A-LINE.
			MOVE CITY TO LABEL-AREA.
			WRITE A-LINE FROM HOLD AFTER ADVANCING 2.
			MOVE SPACES TO A-LINE. GO TO READ-CD.

PART C

PART D

APPENDIX G (Cont'd.)

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 3 of 3
Program PROBLEM 2		Graphic	Card Form	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page)	Sequence (Serial)	Cont	A (8)	B (12) COBOL Statement
	01			END-OF-JOB.
PART	02			CLOSE CARD-IN, PRINTER.
D	03			STOP RUN.
	04			
	05			
	06			
	07			
	08			
	09			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			
	18			
	19			
	20			

APPENDIX G (Cont'd.)

Problem 2 Part E

DIAGNOSTIC MESSAGES

CARD	ERROR MESSAGE	
2	IF010151-E	HOU-ALT INVALID AS EXTERNAL-NAME. IGNORED.
8	IF010871-W	' INPUT-OUTPUT ' SHOULD NOT BEGIN A-MARGIN.
8	IF01004I-E	INVALID WORD INPUT-OUTPUT. SKIPPING TO NEXT RECOGNIZABLE WORD.
10	IF010021-W	INPUT-OUTPUT SECTION HEADER MISSING. ASSUMED PRESENT.
34	IF01056I-E	FILE-NAME NOT DEFINED IN A SELECT. DESCRIPTION IGNORED.
37	IF01087I-W	' NEW PAGE ' SHOULD NOT BEGIN A-MARGIN.
38	IF01087I-W	' TWO-LINES ' SHOULD NOT BEGIN A-MARGIN.
10	IF02146I-W	RECORD SIZE IN RECORD-CONTAINS CLAUSE DISAGREES WITH COMPUTED RECORD SIZE. USING MAXIMUM COMPUTED SIZE.
37	IF021281-C	VALUE CLAUSE LITERAL DOES NOT CONFORM TO PICTURE. CHANGED TO ZERO.
39	IF01087I-W	' HEAD ' SHOULD NOT BEGIN A-MARGIN.
38	IF02129I-C	VALUE CLAUSE LITERAL DOES NOT CONFORM TO PICTURE. CHANGED TO ZERO..
55	IF010871-W	' TYPE-OUT ' SHOULD NOT BEGIN A-MARGIN.
63	IF010011-C	NUMERIC LITERAL NOT RECOGNIZED AS LEVEL NUMBER BECAUSE CODE ILLEGAL AS USED. SKIPPING TO NEXT LEVEL, SECTION OR DIVISION.
71	IF010871W	' MALE ' SHOULD NOT BEGIN A-MARGIN.
72	IF01087I-W	' FEMALE ' SHOULD NOT BEGIN A-MARGIN.
73	IF010871-W	' NOT ' SHOULD NOT BEGIN A-MARGIN.
73	IF010011-C	NUMERIC LITERAL NOT RECOGNIZED AS LEVEL NUMBER BECAUSE NOT LEGAL AS USED. SKIPPING TO NEXT LEVEL, SECTION OR DIVISION.
72	IF02126I-C	VALUE CLAUSE LITERAL TOO LONG. TRUNCATED TO PICTURE SIZE.
75	IF03001I-E	PRINT-OUT NOT DEFINED. DELETING TILL LEGAL ELEMENT FOUND.
78	IF04050I-E	SYNTAX REQUIRES RECORD-NAME. FOUND DNM=1220. STATEMENT DISCARDED.

APPENDIX G (Cont'd.)

DOCUMENTATION CHECKLIST FOR COBOL PROGRAMS

Form	Form Description	Check	Remarks
1	COMPUTER CONFIGURATION SCHEMATIC		
1	NARRATIVE		
2	PROGRAM DESCRIPTION		
3	FILE LAYOUTS		
4	DATE CARD LAYOUT		
5	80 COLUMN DATA CARD LAYOUT		
6	CONSOLE MESSAGES		
7	CONTROL CARD LISTING		
8	KEYPUNCH INSTRUCTIONS		
9	CARRIAGE CONTROL SET UP		
10	PRINTER LAYOUT		
	SAMPLE REPORT FORMS		
11	PROGRAM LISTING		
	G-12		

ANALYST: PROGRAMMER: SUPERVISOR:

APPENDIX G (Cont'd.)	Program Number:
	Program Title:

NARRATIVE:

APPENDIX G (Cont'd.)

Program number:
Program title:

PROGRAM DESCRIPTION

Program language

Core required

Approximate run time

System:

Model: _____ Disk _____ Tape _____ Labels _____

Printer _____ Paper _____ Carriage tape _____

Card reader _____

Card punch _____ Console _____

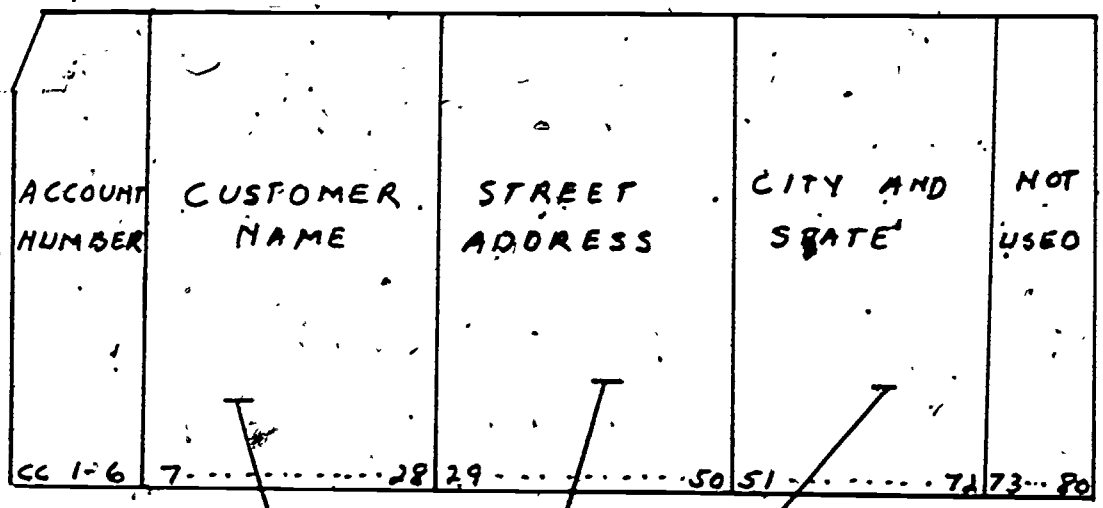
INPUT FILES:

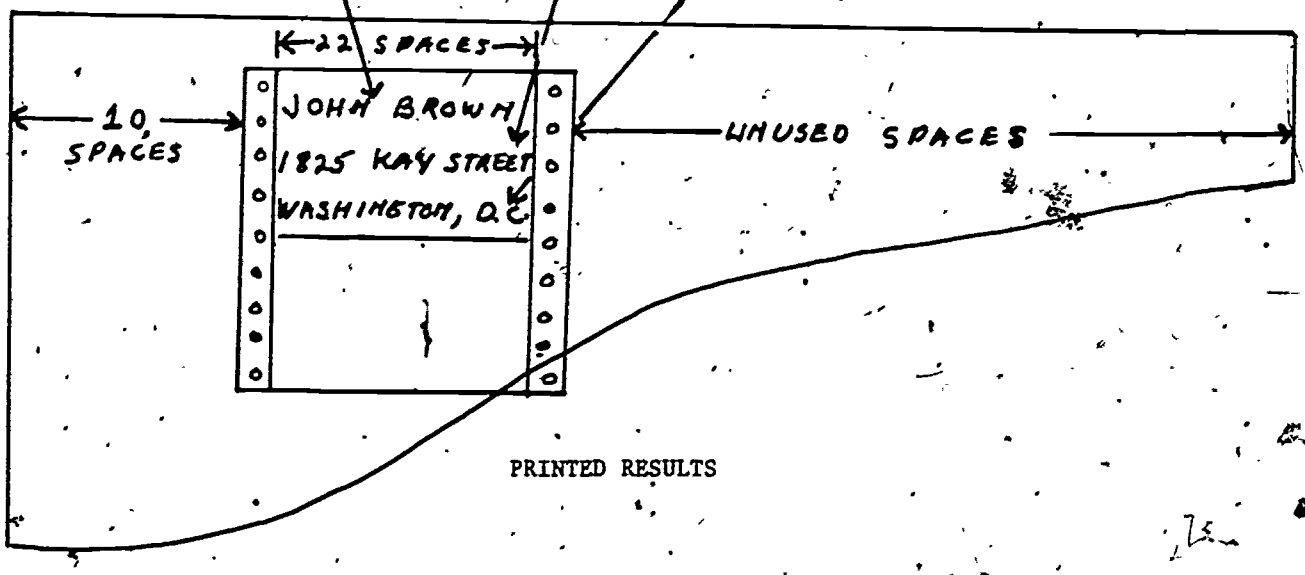
	File Name	Sequence	Device
1			
2			
3			
4			
5			
6			

OUTPUT FILES:

	File Name	Sequence	Device
1			
2			
3			
4			
5			
6			

CARD LAYOUT





PRINTED RESULTS

APPENDIX G (Cont'd.).

Program number:
Program title:

CONSOLE MESSAGES			ACTION
No.	Message	Cause	

APPENDIX G (Cont'd.)

Program number:
Program title:

CONTROL CARD LISTING

Seq	Card Description	Disposition

APPENDIX G (Cont'd.)

PROBLEM 2

The following is provided for a name and address card:

- name in card columns 1 through 30
- street address in card columns 31 through 50
- city in card columns 51 through 57
- zip code in card columns 58 through 62
- age in card columns 63 and 64
- phone number in card columns 65 through 71
- and a special code in card column 72 to represent sex:

  male = 1, female = 2, and not given = any other number

The printed output will consist of a heading at the top of each page followed by detail lines, single-spaced three lines below the heading as follows:

NAME ADDRESS CITY ZIP AGE PHONE SEX

At the end, the message *end of job* is to be printed on the computer console.

Prepare coded instructions, ~~sample~~ output, and complete documentation for the problem. Use the flowchart prepared in a prior exercise to aid in coding. Include the flowchart in the documentation.

APPENDIX G (Cont'd.)

DOCUMENTATION CHECKLIST FOR COBOL PROGRAMS

Form	Form Description	Check	Remarks
1	COMPUTER CONFIGURATION SCHEMATIC		
1	NARRATIVE		
2	PROGRAM DESCRIPTION		
3	FILE LAYOUTS		
4	DATE CARD LAYOUT		
5	80 COLUMN DATA CARD LAYOUT		
6	CONSOLE MESSAGES		
7	CONTROL CARD LISTING		
8	KEYPUNCH INSTRUCTIONS		
9	CARRIAGE CONTROL SET UP		
10	PRINTER LAYOUT		
	SAMPLE REPORT FORMS		
11	PROGRAM LISTING		
	G-19		

ANALYST: PROGRAMMER: SUPERVISOR:

APPENDIX G (Cont'd.)	Program Number:
	Program Title:

NARRATIVE:

APPENDIX G (Cont'd.)

Program number:
Program title:

PROGRAM DESCRIPTION

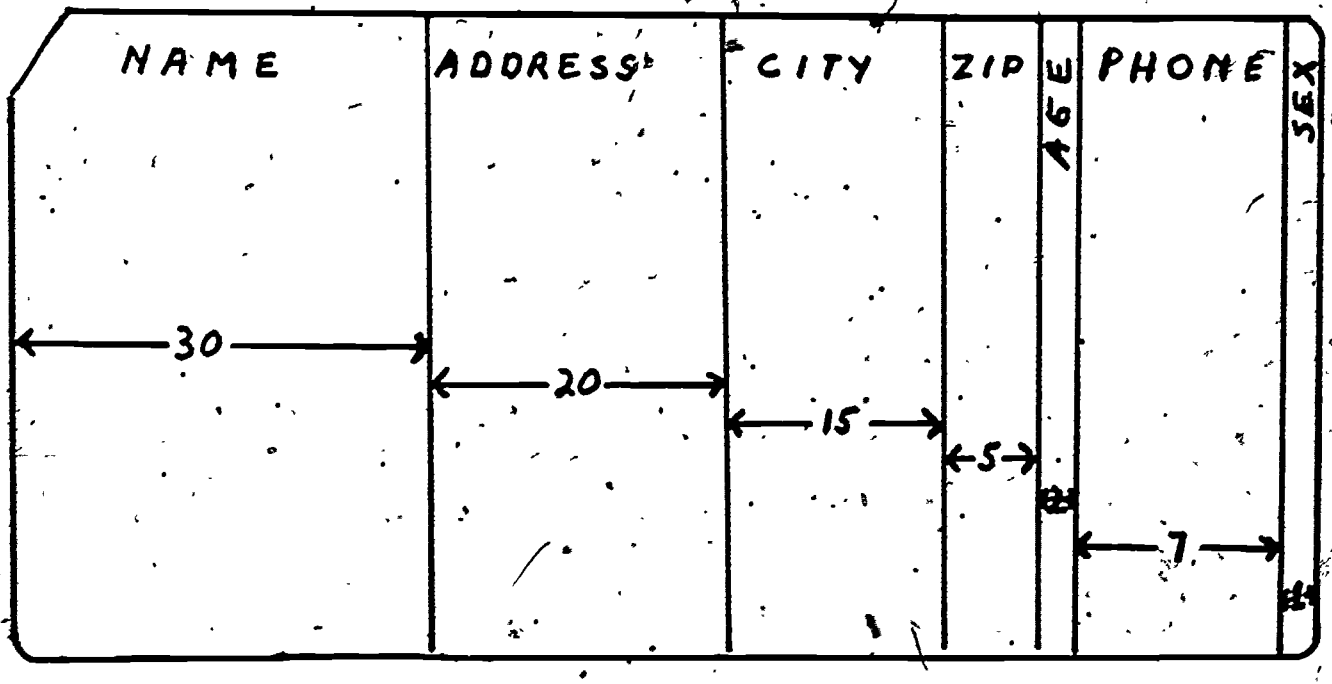
Program language
Core required
Approximate run time
System:
Model: _____ Disk _____ Tape _____ Labels _____
Printer _____ Paper _____ Carriage tape _____
Card reader _____
Card punch _____ Console _____

INPUT FILES:

	File Name	Sequence	Device
1			
2			
3			
4			
5			
6			

OUTPUT FILES:

	File Name	Sequence	Device
1			
2			
3			
4			
5			
6			



6

APPENDIX G (Cont'd.)

Program number:
Program title:

CONSOLE MESSAGES			ACTION
No.	Message	Cause	

APPENDIX G (Cont'd.)

Program number:
Program title:

CONTROL CARD LISTING

Seq.	Card Description	Disposition

SAMPLE
PRINT CHART

PROGRAM TITLE ALT PROG

CARRIAGE CONTROL

NAME	ADDRESS	CITY	ZIP	AGE	PHONE	SEX

G-25

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 1 of 5
Program Problem 2		Graphic	Card Form #	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page)	Sequence (Serial)	Cont.	A (8)	B (12) COBOL Statement
	01			IDENTIFICATION DIVISION.
PART A	02			PROGRAM-ID. 'ALTPROG'.
	03			AUTHOR. HOUGH.
	04			
	05			ENVIRONMENT DIVISION.
	06			
	07			CONFIGURATION SECTION.
	08			SOURCE-COMPUTER. IBM-360.
	09			OBJECT-COMPUTER. IBM-360.
	10			
PART B	11			INPUT-OUTPUT SECTION.
	12			FILE-CONTROL.
	13			SELECT CARD-IN ASSIGN 'SYS040' UNIT-RECORD 2540R.
	14			SELECT PRINTER ASSIGN 'SYS043' UNIT-RECORD 1403.
	15			I-O-CONTROL.
	16			APPLY OVERFLOW TO FORM-OVERFLOW ON PRINTER.
	17			
	18			
	19			
	20			

APPENDIX G (Cont'd.)

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 2 of 5
Program		Graphic	Card Form #	Identification 73 80
Programmer	Date	Punch		

Sequence (PAGE) 1–3	(SERIAL) 4–6	CONT. 7	A 8–11	B 12–72 COBOL Statement
	01		DATA DIVISION.	
	02			
	03		FILE SECTION.	
	04		FD	CARD-IN,
	05			RECORDING MODE IS F, LABEL RECORDS ARE OMITTED,
	06			RECORD CONTAINS 80 CHARACTERS; DATA RECORD IS CO-IN.
	07			
	08		01	CO-IN.
	09			02 NAME PICTURE A(30).
	10			02 STREET PICTURE X(20).
	11			02 CITY PICTURE X(15).
PART C	12			02 ZIP PICTURE 9(5).
	13			02 AGE PICTURE 99.
	14			02 PHONE PICTURE 9(7).
	15			02 SEX. PICTURE 9.
	16			
	17		FD	PRINTER,
	18			RECORDING MODE IS F,
	19			LABEL RECORDS ARE OMITTED,
	20			RECORD CONTAINS 133 CHARACTERS
				DATA RECORD IS A-LINE.
			01	A-LINE PICTURE X(133).

APPENDIX G (Cont'd.)

(1-27)

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 3 of 5
Program		Graphic	Card Form	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page, Line)	Cont.	A	B — COBOL Statement
01			WORKING-STORAGE SECTION.
02			
03		77	SKIP-TO-A-NEW-PAGE PICTURE 9 VALUE 1.
04		77	TWO-LINES PICTURE 9 VALUE 2.
05			
06		01	HEADER.
07			02 FILLER PICTURE X(18) VALUE SPACES.
08			02 PART-1 PICTURE A(4) VALUE 'NAME'.
09			02 FILLER PICTURE X(25) VALUE SPACES.
PART 10			02 PART-2 PICTURE A(7) VALUE 'ADDRESS'.
C 11			02 FILLER PICTURE X(17) VALUE SPACES.
12			02 PART-3 PICTURE X(4) VALUE 'CITY'.
13			02 FILLER PICTURE X(13) VALUE SPACES.
14			02 PART-4 PICTURE A(3) VALUE 'ZIP'.
15			02 FILLER PICTURE X(6) VALUE SPACES.
16			02 PART-5 PICTURE A(3) VALUE 'AGE'.
17			02 FILLER PICTURE X(7) VALUE SPACES.
18			02 PART-6 PICTURE A(5) VALUE 'PHONE'.
19			02 FILLER PICTURE X(9) VALUE SPACES.
20			02 PART-7 PICTURE A(3) VALUE 'SEX'.
			02 FILLER PICTURE X(9) VALUE SPACES.

APPENDIX G (cont'd.)

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 4 of 5
Program		Graphic / Punch	Card Form #	Identification (73–80)
Programmer	Date			

Sequence (Page) 1–3	Sequence (Serial) 4–6	Cont. 7	A 8	B 12–72 COBOL Statement
	01		01	TYPE-OUT.
	02			02 FILLER PICTURE X(6) VALUE SPACES.
	03			02 NAME-O PICTURE A(30).
	04			02 FILLER PICTURE X(5) VALUE SPACES.
	05			02 ADDRES PICTURE X(20).
	06			02 FILLER PICTURE X(5) VALUE SPACES.
	07			02 TOWN PICTURE X(15).
	08			02 FILLER PICTURE X(6) VALUE SPACES.
	09			02 ZP PICTURE 9(5).
	10			02 FILLER PICTURE X(6) VALUE SPACES.
PART	11			02 OLD PICTURE 99.
C	12			02 FILLER PICTURE X(6) VALUE SPACES.
	13			02 TELLA PICTURE X(7).
	14			02 FILLER PICTURE X(5) VALUE SPACES.
	15			02 GENDER PICTURE A(9).
	16			02 FILLER PICTURE X(6) VALUE SPACES.
	17			
	18		01	MALE PICTURE A(9) VALUE 'MALE ".
	19			
	20		01	FEMALE PICTURE A(9) VALUE 'FEMALE ".
			01	MISSING PICTURE A(9) VALUE 'NOT GIVEN'.

APPENDIX G (Cont'd.)

Sample COBOL Program Sheet

System		Punching Instructions		Sheet 5 of 5
Program		Graphic	Card Form #	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page) (Serial) 1–3 4–6 | Cont. 7 | A 8 | B 12–72 COBOL Statement

```
PROCEDURE DIVISION.
START.  OPEN INPUT CARD-IN,
                OUTPUT PRINTER.
WRITE-HEAD.
        WRITE A-LINE FROM HEADER AFTER SKIP-TO-A-NEW-PAGE.
        MOVE SPACES TO A-LINE.
        WRITE A-LINE AFTER ADVANCING TWO-LINES.
READ-CD.
        READ CARD-IN AT END GO TO END-OF-JOB.
        MOVE NAME TO NAME-O.
        MOVE STREET TO ADDRES.
        MOVE CITY TO TOWN.
        MOVE ZIP TO ZP. MOVE AGE TO OLD.
        MOVE PHONE TO TELLA.
        IF SEX = 1 MOVE MALE TO GENDER, GO TO NEXT-CHECK.
        IF SEX = 2 MOVE FEMALE TO GENDER, GO TO NEXT-CHECK,
        ELSE MOVE MISSING TO GENDER.
NEXT-CHECK.
        WRITE A-LINE FROM TYPE-OUT AFTER 1, IF OVERFLOW GO
        WRITE-HEAD. GO TO READ-CD.
END-OF-JOB.
        CLOSE CARD-IN, PRINTER.
        DISPLAY 'END OF THE JOB' UPON CONSOLE.
        STOP RUN.
```

PART D

APPENDIX G (Cont'd.)

APPENDIX H. INTERMEDIATE COBOL

APPENDIX H

PART 1.

In working storage, use the 88 level to write an elementary item called paycode which uses a code of 1 for bad, 2 for fair, and 3 for good.

Answer

```
02 Paycode picture 9
   88 Bad value 1
   88 Fair value 2
   88 Good value 3
```

PART 2.

Using the working-storage section developed in the previous problem, write an IF statement to meet the following conditions:

- When max is equal to total and paycode is equal to a value of 1, go to no-credit.

Answer

If max = total and paycode = bad, go to no-credit.

PART 3.

Using the previous working-storage section, write an IF statement so that, when total is smaller than max or the paycode is equal to a value of 3, go to ok-routine.

Answer

If total less than max or paycode = good, go to ok-routine.

PART 4.

Code the working-storage section of the data division to provide for a table containing three elements of two numbers. Call the table Days-of-credit, the elements Days-to-pay, and the subscript Ctr.

Answer

```
77 Ctr picture 9
01 Days-of-credit
   02 Days-to-pay picture 99 occurs 3 times.
```

APPENDIX H (Cont'd.)

PART 5.

Given a table containing the following:

```
02 Table-name
   03 One picture X(25)
   03 Two picture X(25)
   03 Three picture X(25)
```

show the realignment and subscripting of the elements in the table to contain a NAME (14 positions) and AMT (11 positions).

Answer

```
02 New-table redefines table-name occurs 3 times.
   03 One-element
      04 Name picture A(14)
      04 Amt picture 9(11)
```

PART 6.

Given a table called new-table which has been redefined and subscripted, write the following procedure statements:

1. Initialize a subscript called ctr.
2. Move name to an area called nameout.
3. Go to a paragraph called out when all elements of the table have been moved.

Answer

1. Move 1 to ctr.
2. Move name (ctr) to nameout.
3. If ctr = 4 to to out.

PART 7.

Given a table called days-of-credit which has been redefined and subscripted in a previous problem, rewrite the table to include indexed by a ctr.

Answer

```
01 Days-of-credit
   02 Days-to-pay picture 99 occurs 3 times indexed by ctr.
```

PART 8.

Given the table in the previous problem, write a statement to set the index to a value of 4.

Answer

Set ctr to 2.

PART 9.

Write the procedure segment to print a heading from an output area called A-line, and show the segment at the end of the program.

Answer

End-of-job

Close card-file, printer-file.
Stop run.

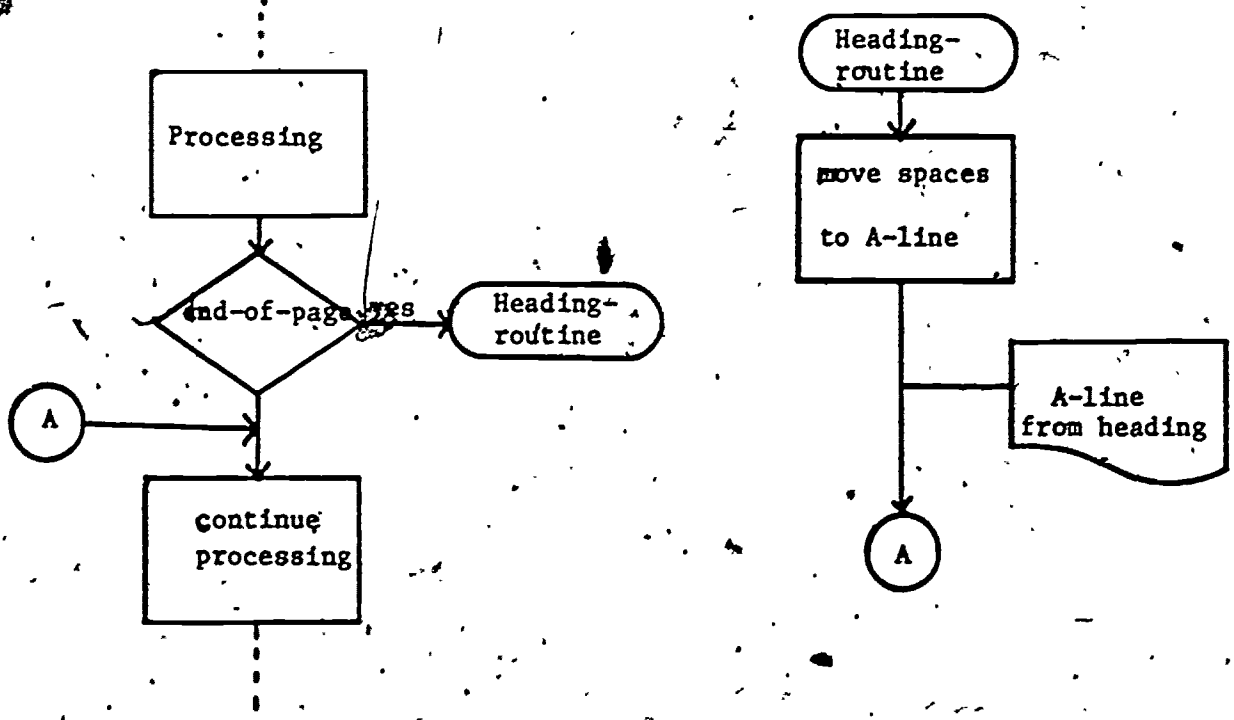
Heading-routine

Move spaces to A-line.
Write A-line from heading after advancing to-a-new-page.

PART 10.

Using the previous problem, draw a segment of the flowchart to show branching to and returning from a subroutine.

Flowchart segment



COBOL PROGRAM SHEET

System			Punching Instructions	Sheet 1 of 3
Program CALLING OR MAIN PROGRAM		Graphic	Card Form# *	Identification
Programmer	Date	Punch		73 80

SEQUENCE (PAGE) 1 3 (SERIAL) 4 6 | CONT 7 | A 8 | B 12 16 20 24 28 32 36 40 44 48 52 56 60 64 68 72

```
IDENTIFICATION DIVISION.
PROGRAM-ID. 'MAINPRGM'.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-360.
OBJECT-COMPUTER. IBM-360.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
     SELECT CARD-IN ASSIGN 'SYS004' UNIT-RECORD 2540R.
     SELECT PRINTER ASSIGN 'SYS005' UNIT-RECORD 1403.
DATA DIVISION.
FILE SECTION.
FD   CARD-IN, RECORDING MODE IS F, LABEL RECORDS ARE
     OMITTED, RECORD CONTAINS 80 CHARACTERS, DATA
     RECORD IS CARD.
01   CARD.
     02 DATA-FIELD PICTURE X(11).
     02 FILLER PICTURE X(69).
FD   PRINTER, RECORDING MODE IS F, LABEL RECORDS ARE
     OMITTED, RECORD CONTAINS 133 CHARACTERS, DATA
     RECORD IS LINE.
01   LINE.
     02 FILLER PICTURE X(27)
     02 DATA-AREA.
        03 FIELDA PICTURE X(69)
```

APPENDIX H (Cont'd.)

COBOL PROGRAM SHEET

System		Punching Instructions	Sheet 2 of 3
Program CALLING OR MAIN PROGRAM		Graphic / Card Form #	Identification
Programmer	Date	Punch	73 – 80

```
                 03 FIELDB PICTURE X(11).
              02 FILLER PICTURE X(26).
PROCEDURE DIVISION.
START. OPEN INPUT CARD-IN OUTPUT PRINTER.
READ-AGAIN.
          READ CARD-IN AT END GO TO END-OF-JOB.
          ENTER LINKAGE.
          CALL 'SUBPRGM' USING CARD.
          ENTER COBOL.
          NOTE SUBPROGRAM CHANGES INFORMATION IN CARD.
          MOVE CARD TO DATA-AREA.
          WRITE LINE AFTER ADVANCING 2.
          GO TO READ-AGAIN.
END-OF-JOB.
          CLOSE CARD-IN, PRINTER.
          STOP RUN.
```

APPENDIX H (Cont'd.)

Department of Data Systems

COBOL Program Sheet

MONTGOMERY COUNTY PUBLIC SCHOOLS

System		Punching Instructions		Sheet 3 of 3
Program SUBPROGRAM		Graphic	Card Form #	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page) 1–3	(Serial) 4–6	Cont. 7	A 8–11	B 12–72 COBOL Statement
	01			IDENTIFICATION DIVISION.
	02			PROGRAM-ID. 'FIRSTSUB'.
	03			ENVIRONMENT DIVISION.
	04			CONFIGURATION SECTION.
	05			SOURCE-COMPUTER. IBM-360.
	06			OBJECT-COMPUTER. IBM-360.
	07			
	08			DATA DIVISION.
	09			WORKING-STORAGE SECTION.
	10			77 HOLD PICTURE X(11) VALUE 'DATA PASSED'.
	11			LINKAGE SECTION.
	12			01 PASSING-FIELD.
	13			02 FILLER PICTURE X(69).
	14			02 A-FIELD PICTURE X(11).
	15			PROCEDURE DIVISION.
	16			START.
	17			ENTER LINKAGE.
	18			ENTRY 'SUBPROG' USING PASSING-FIELD.
	19			ENTER COBOL.
	20			MOVE HOLD TO A-FIELD.
				ENTER LINKAGE.
				RETURN.
				ENTER COBOL.
				NOTE PASSING-FIELD IN THIS PROGRAM IS THE SAME AS
				CARD IN THE MAIN PROGRAM.

APPENDIX H (Cont'd.)

COBOL PROGRAM SHEET

System			Punching Instructions	Sheet 1 of 13
Program		Graphic	Card Form # *	Identification
Programmer	Date	Punch		73 80

SEQUENCE PAGE SERIAL 1 3 4 5 7 | A 8 | B 12 16 20 24 28 32 36 40 44 48 52 56 60 64 68 72

```
WORKING-STORAGE SECTION.
77  CTR  PICTURE 99 VALUE ZERO.
01  HEADER.
    02 FILLER PICTURE X(40) VALUE SPACES.
    02 ONE PICTURE X(10) VALUE 'STATE-NAME'.
    02 FILLER PICTURE X(10) VALUE SPACES.
    02 TWO PICTURE X(10) VALUE 'STATE-CODE'.
    02 FILLER PICTURE X(10) VALUE SPACES.
    02 THREE PICTURE X(12) VALUE 'STATE-AMOUNT'.
    02 FILLER PICTURE X(41) VALUE SPACES.
01  TYPEOUT.
    02 FILLER PICTURE X(41) VALUE SPACES.
    02 NAME-HOLD PICTURE A(14).
    02 FILLER PICTURE X(12) VALUE SPACES.
    02 STATE-HOLD PICTURE 9(2).
    02 FILLER PICTURE X(12) VALUE SPACES.
    02 AMOUNT-HOLD PICTURE ZZZZZZ,ZZ.
    02 FILLER PICTURE X(41) VALUE SPACES.
01  TABLE.
    02 TABLE-A.
      03 FILLER PICTURE X(25) VALUE 'ALABAMA          01000000000'.
      03 FILLER PICTURE X(25) VALUE 'ALASKA           02000000000'.
      03 FILLER PICTURE X(25) VALUE 'ARIZONA          03000000000'.
      03 FILLER PICTURE X(25) VALUE 'ARKANSAS         04000000000'.
      03 FILLER PICTURE X(25) VALUE 'CALIFORNIA       05000000000'.
```

APPENDIX H (Cont'd.)

COBOL PROGRAM SHEET

System		Punching Instructions	Sheet 2 of 3
Program		Graphic / Card Form# *	Identification
Programmer	Date	Punch	73 80

```
03 FILLER PICTURE X(25) VALUE 'COLORADO                 06000000000',
03 FILLER PICTURE X(25) VALUE 'CONNECTICUT              07000000000',
03 FILLER PICTURE X(25) VALUE 'DELAWARE                 08000000000',
03 FILLER PICTURE X(25) VALUE 'FLORIDA                  09000000000',
03 FILLER PICTURE X(25) VALUE 'GEORGIA                  10000000000',
03 FILLER PICTURE X(25) VALUE 'HAWAII                   11000000000',
03 FILLER PICTURE X(25) VALUE 'IDAHO                    12000000000',
03 FILLER PICTURE X(25) VALUE 'ILLINOIS                 13000000000',
03 FILLER PICTURE X(25) VALUE 'INDIANA                  14000000000',
03 FILLER PICTURE X(25) VALUE 'IOWA                     15000000000',
03 FILLER PICTURE X(25) VALUE 'KANSAS                   16000000000',
03 FILLER PICTURE X(25) VALUE 'KENTUCKY                 17000000000',
03 FILLER PICTURE X(25) VALUE 'LOUISIANA                18000000000',
03 FILLER PICTURE X(25) VALUE 'MAINE                    19000000000',
03 FILLER PICTURE X(25) VALUE 'MARYLAND                 20000000000',
03 FILLER PICTURE X(25) VALUE 'MASSACHUSETTS            21000000000',
03 FILLER PICTURE X(25) VALUE 'MICHIGAN                 22000000000',
03 FILLER PICTURE X(25) VALUE 'MINNESOTA                23000000000',
03 FILLER PICTURE X(25) VALUE 'MISSISSIPPI              24000000000',
03 FILLER PICTURE X(25) VALUE 'MISSOURI                 25000000000',
03 FILLER PICTURE X(25) VALUE 'MONTANA                  26000000000',
03 FILLER PICTURE X(25) VALUE 'NEBRASKA                 27000000000',
03 FILLER PICTURE X(25) VALUE 'NEVADA                   28000000000',
03 FILLER PICTURE X(25) VALUE 'NEW HAMPSHIRE            29000000000',
03 FILLER PICTURE X(25) VALUE 'NEW JERSEY               30000000000',
```

COBOL PROGRAM SHEET

System		Punching Instructions	Sheet 3 of 3
Program	Graphic	Card Form # *	Identification
Programmer	Date	Punch	73 80

```
03 FILLER PICTURE X(25) VALUE 'NEW MEXICO          31000000000',
03 FILLER PICTURE X(25) VALUE NEW YORK             32000000000',
03 FILLER PICTURE X(25) VALUE NORTH CAROLINA       33000000000',
03 FILLER PICTURE X(25) VALUE NORTH DAKOTA         34000000000',
03 FILLER PICTURE X(25) VALUE OHIO                 35000000000',
03 FILLER PICTURE X(25) VALUE OKLAHOMA             36000000000',
03 FILLER PICTURE X(25) VALUE OREGON               37000000000',
03 FILLER PICTURE X(25) VALUE PENNSYLVANIA         38000000000',
03 FILLER PICTURE X(25) VALUE RHODE ISLAND         39000000000',
03 FILLER PICTURE X(25) VALUE SOUTH CAROLINA       40000000000',
03 FILLER PICTURE X(25) VALUE SOUTH DAKOTA         41000000000',
03 FILLER PICTURE X(25) VALUE TENNESSEE            42000000000',
03 FILLER PICTURE X(25) VALUE TEXAS                43000000000',
03 FILLER PICTURE X(25) VALUE UTAH                 44000000000',
03 FILLER PICTURE X(25) VALUE VERMONT              45000000000',
03 FILLER PICTURE X(25) VALUE VIRGINIA             46000000000',
03 FILLER PICTURE X(25) VALUE WASHINGTON           47000000000',
03 FILLER PICTURE X(25) VALUE WEST VIRGINIA        48000000000',
03 FILLER PICTURE X(25) VALUE WISCONSIN            49000000000',
03 FILLER PICTURE X(25) VALUE WYOMING              50000000000',
02 TABLE-1 REDEFINES TABLE-A.
03 STATE-TABLE OCCURS 50 TIMES.
   04 STATE-NAME PICTURE X(14).
   04 STATE-CODE PICTURE 99.
   04 STATE-AMT PICTURE 9(7)V99.
```

PART 11.

Write a multi-level table in working storage which contains two class codes and four train fares as follows.

	Jacksonville	Orlando	Tampa	Miami
Tourist	$120.00	$150.00	$175.00	$225.00
Coach	$ 95.00	$120.00	$140.00	$180.00

Redefine the table so that it can be referenced by subscripts.

Answer-

```
01 Table computational — 3

   02 Coach

      03 Filler picture 9(3)v99 value 95.00
      03 Filler picture 9(3)v99 value 120.00
      03 Filler picture 9(3)v99 value 140.00
      03 Filler picture 9(3)v99 value 180.00

   02 Tourist

      03 Filler picture 9(3)v99 value 120.00
      03 Filler picture 9(3)v99 value 150.00
      03 Filler picture 9(3)v99 value 175.00
      03 Filler picture 9(3)v99 value 225.00

01 New-table redefines table computations — 3

   02 Class-code occurs 2 times

      03 Train-fare occurs 4 times picture 9(3)v99
```

PART 12.

Using the previous problem, write a statement to move the Orlando coach fare to an area called Fare-cost.

Answer

```
Move Train-fare (1,2) to Fare-cost
```

Using the previous problem, write a statement to test the elements and branch to Read-again if the limits of the table have been exceeded, or to continue processing if the elements are within the bounds of the table.

Answer

```
If class-code greater than 2 or Train-fare greater than 4, go to Read-again.
```

Division of Systems Development

COBOL Program Sheet

MONTGOMERY COUNTY PUBLIC SCHOOLS

System		Punching Instructions		Sheet of
Program		Graphic	Card	Identification
Programmer	Date	Punch	Form	73 80

Sequence (PAGE) 1 3	(SERIAL) 4 6	CONT 7	A 8	B 12	COBOL Statement 16 20 24 28 32 36 40 44 48 52 56 60 64 68 72

APPENDIX H (Cont'd.)

REPORT PROGRAM GENERATOR-FILE DESCRIPTION SPECIFICATIONS
System/360

Date _____

Program _____

Programmer _____

Punching Instruction: Graphic / Punch

Page

Program Identification

Line	Form Type	Filename	File Type (I/O/U/C)	File Designation (P/S/C/D/T)	End of File (E)	Sequence (A/D)	File Format (F/V)	Block Length	Record Length	Mode of Processing (L/R)	Length of Key Field or of Record Address Field	Record Address Type (K/T)	Type of File Organization (I/D/T)	Overflow Indicator	Key Field Starting Location	Extension Code (E/L)	Device	Symbolic Device	Labels (S, N, or L)	Name of Label Exit	Extent Exit for DAM	File Addition (A)	No. Tracks for Cylinder Overflows	No. of Extents	Tape Rewind (N/U)
1 2 3	4	5	6	7	8	9	10	11 12 13 14	15 16 17 18	19	20 21 22 23	24 25 26 27	28	29 30	31	32	33 34	35 36 37 38	39	40 41 42 43 44 45 46	47 48 49 50 51 52	53	54 55 56 57 58 59	60 61 62 63 64 65	66	67 68 69	70	71 72 73 74
01	F																								
02	F																								
03	F																								
04	F																								
05	F																								
06	F																								
07	F																								
08	F																								
09	F																								
10	F																								
11	F																								
12	F																								
13	F																								
14	F																								
15	F																								
	F																								
	F																								
	F																								
	F																								
	F																								

Card Electro Number _____

REPORT PROGRAM GENERATOR - INPUT SPECIFICATIONS
System/360

Date _____

Program _____

Programmer _____

Punching Instruction: Graphic / Punch

Page 1 2

Program Identification 73 74 75 76 77 78 79 80

Line	Form Type	Filename	Sequence	Number (1 N)	Option (O)	Resulting Indicator	Record Identification Codes 1 Position	1 Not (N)	1 C/Z/D	1 Character	2 Position	2 Not (N)	2 C/Z/D	2 Character	3 Position	3 Not (N)	3 C/Z/D	3 Character	Stacker Select	Punched (P)	Field Location From	Field Location To	Decimal Positions	Field Name	Control Level (L1 L9)	Matching Fields or Chaining Fields	Field Output Suppression	Field Indicators Plus	Field Indicators Minus	Field Indicators Zero or Blank	Starting Sign Position
01	I																														
02	I																														
03	I																														
04	I																														
05	I																														
06	I																														
07	I																														
08	I																														
09	I																														
10	I																														
11	I																														
12	I																														
13	I																														
14	I																														
15	I																														
	I																														
	I																														
	I																														
	I																														
	I																														

REPORT PROGRAM GENERATOR - CALCULATION SPECIFICATIONS
System/360

Date _____

Program _____

Programmer _____

Punching Instruction: Graphic / Punch

Page

Program Identification

Line	Form Type	Control Level (L0, L1, L9, LR)	Indicators: And Not	Indicators: And Not	Indicators: And Not	Factor 1	Operation	Factor 2	Result Field	Field Length	Decimal Positions	Half Adjust (H)	Resulting Indicators: Plus	Resulting Indicators: Minus	Resulting Indicators: Zero or Blank	Compare: High 1 > 2	Compare: Low 1 < 2	Compare: Equal 1 = 2	Comments
3 4 5	6	7 8	9 10 11	12 13 14	15 16 17	18 19 20 21 22 23 24 25 26 27	28 29 30 31 32	33 34 35 36 37 38 39 40 41 42	43 44 45 46 47 48	49 50 51	52	53	54	55	56	57	58	59	60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
01	C																		
02	C																		
03	C																		
04	C																		
05	C																		
06	C																		
07	C																		
08	C																		
09	C																		
10	C																		
11	C																		
12	C																		
13	C																		
14	C																		
15	C																		
	C																		
	C																		
	C																		
	C																		
	C																		

Card Electro Number _____

APPENDIX H (Cont'd.)

REPORT PROGRAM GENERATOR - OUTPUT FORMAT SPECIFICATIONS
System/360

Date _____

Program _____

Programmer _____

Punching Instruction	Graphic							
	Punch							

Page 1 2

Program Identification 75 76 77 78 79 80

Line	Form Type	Filename	Type (N/D/T)	Skipping Before	Skipping After	Spacing Before	Spacing After	Output Indicators Not	Output Indicators And	Output Indicators Not	Output Indicators And	Output Indicators Not	Field Name	Zero Suppress (Z)	Blank After (B)	End Position in Output Record	Period Field (P)	Constant or Edit Word	Sterling Sign Position
0 1	O																		
0 2	O																		
0 3	O																		
0 4	O																		
0 5	O																		
0 6	O																		
0 7	O																		
0 8	O																		
0 9	O																		
1 0	O																		
1 1	O																		
1 2	O																		
1 3	O																		
1 4	O																		
1 5	O																		
	O																		
	O																		
	O																		

COBOL Program Sheet

System		Punching Instructions		Sheet of
Program		Graphic	Card Form	Identification (73–80)
Programmer	Date	Punch		

Sequence (Page) 1–3	Sequence (Serial) 4–6	Cont. 7	A 8–11	B 12–72 COBOL Statement
	01			
	02			
	03			
	04			
	05			
	06			
	07			
	08			
	09			
	10			
	11			
	12			
	13			
	14			
	15			
	16			
	17			
	18			
	19			
	20			

APPENDIX H (Cont'd.)

IBM

INTERNATIONAL BUSINESS MACHINES CORPORATION

MULTIPLE-CARD LAYOUT FORM

Form X24-6599-0
Printed in U.S.A.

Company _____ Date _____ Job No _____ Sheet No _____

Applications _____ By _____