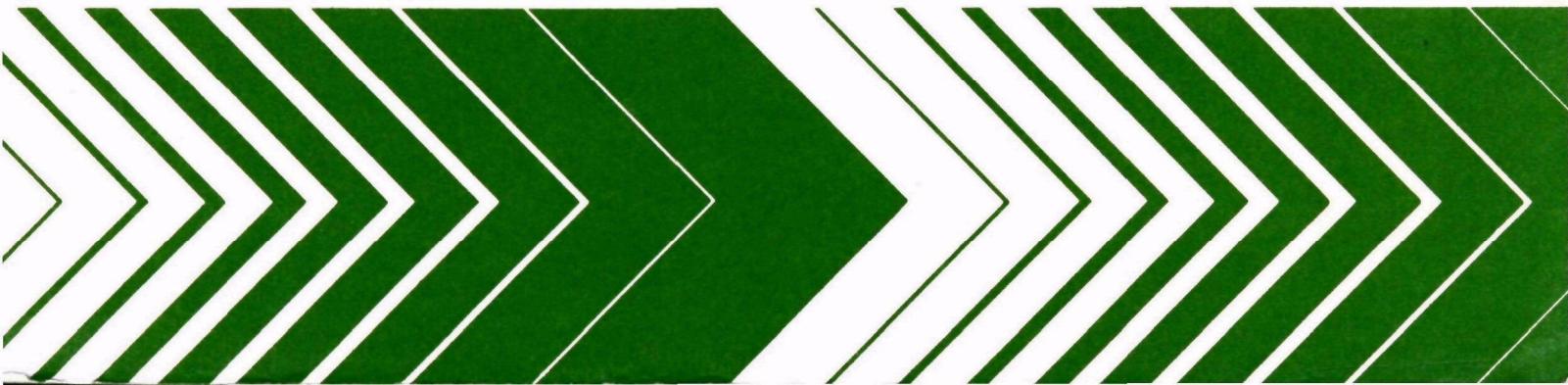

Research and Development



Development and Application of a Water Supply Cost Analysis System

Volume II



RESEARCH REPORTING SERIES

Research reports of the Office of Research and Development, U.S. Environmental Protection Agency, have been grouped into nine series. These nine broad categories were established to facilitate further development and application of environmental technology. Elimination of traditional grouping was consciously planned to foster technology transfer and a maximum interface in related fields. The nine series are

1. Environmental Health Effects Research
2. Environmental Protection Technology
3. Ecological Research
4. Environmental Monitoring
5. Socioeconomic Environmental Studies
6. Scientific and Technical Assessment Reports (STAR)
7. Interagency Energy-Environment Research and Development
8. "Special" Reports
9. Miscellaneous Reports

This report has been assigned to the ENVIRONMENTAL PROTECTION TECHNOLOGY series. This series describes research performed to develop and demonstrate instrumentation, equipment, and methodology to repair or prevent environmental degradation from point and non-point sources of pollution. This work provides the new or improved technology required for the control and treatment of pollution sources to meet environmental quality standards.

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

EPA-600/2-80-012b
July 1980

DEVELOPMENT AND APPLICATION OF A
WATER SUPPLY COST ANALYSIS SYSTEM

Volume II

by

James I. Gillean
Rex D. Osborn
William L. Britton, Jr.
ACT Systems, Inc.
Winter Park, Florida 32789

Robert M. Clark
Municipal Environmental Research Laboratory
Cincinnati, Ohio 45268

Contract No. 68-03-2506

Project Officer

Robert M. Clark
Drinking Water Research Division
Municipal Environmental Research Laboratory
Cincinnati, Ohio 45268

MUNICIPAL ENVIRONMENTAL RESEARCH LABORATORY
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
CINCINNATI, OHIO 45268

DISCLAIMER

This report has been reviewed by the Municipal Environmental Research Laboratory, U.S. Environmental Protection Agency, and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the U.S. Environmental Protection Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.

FOREWORD

The Environmental Protection Agency was created because the public and the federal government were increasingly concerned about the dangers of pollution on the health and welfare of the American people. Noxious air, foul water, and spoiled land are tragic testimonies to the deterioration of our natural environment. The complexity of that environment and the interplay among its components require a concentrated and integrated attack on the problem.

Research and development is that first step in problem solution, and involves defining the problem, measuring its impact, and searching for solutions. The Municipal Environmental Research Laboratory develops new and improved technology and systems: 1) to prevent, treat, and manage wastewater, solid and hazardous waste, and pollutant discharges from municipal and community sources; 2) to preserve and treat public drinking water supplies; and 3) to minimize the adverse economic, social, health, and aesthetic effects of pollution. This publication is a product of that research and is a most vital communications link between the researcher and the user community.

The Safe Drinking Water Act of 1974 establishes primary health-related standards and secondary aesthetic-related but nonenforceable guidelines for drinking water supplies. These standards will bring about a fundamental examination of the way water is handled before it is delivered to the consumer. Many of these changes will have an economic impact on the affected water utilities. This report describes a cost analysis system that was developed to analyze a water utility's operating and maintenance cost data. The cost analysis system provides the manager and other interested parties with an insight into a water utility's performance, and defines, by unique cost centers, those areas where costs can be monitored.

Volume I of this report outlines the background, development, and application of the cost analysis system. Volume II provides the computer program documentation developed to operate the system.

Francis T. Mayo
Director
Municipal Environmental Research Laboratory

ABSTRACT

The Environmental Protection Agency is concerned with the Safe Drinking Water Act's economic impact on the water supply industry. The Environmental Protection Agency initiated an ongoing program to gather and examine cost data from utilities across the country on which to establish a data base for water supply cost.

Experience has shown that although most utilities maintain an accounting system, the data provided by the systems are maintained in different formats and are therefore incompatible among utilities. Few systems provide continuous and adequate cost information directly.

As a result of this effort, the Environmental Protection Agency and many utility managers agree that there is a need for a cost analysis system having features especially applicable to the water supply industry. Based on the perceived need and because of the Act's emphasis on economics, the Drinking Water Research Division initiated a research program to develop standardized techniques for analyzing costs within a utility accounting framework. This program is described in a two-volume report. Volume I describes the development and functions of this system and its application to Kenton County, Kentucky, Water District No. 1.

Volume II contains the programming documentation and operating instructions for the cost analysis system.

This report is submitted by ACT Systems, Inc., in partial fulfillment of Contract No. 68-03-2506, Section II, under the sponsorship of the U.S. Environmental Protection Agency.

CONTENTS

	<u>Page</u>
Foreword	iii
Abstract	iv
Figures	v
Acknowledgments	vi
1. Introduction	1
2. Cost Analysis System Computer Operations	3
System Flow Diagrams	3
Computer Program Directory	3
System Operating Instructions	8
Routine Operating Sequence	41
3. Data Format	43
Keypunching Source Data	43
System File Layout	66
4. Program Flow Diagrams	77
5. Program Listing	124

FIGURES

<u>Number</u>		<u>Page</u>
1	Cost analysis system flow diagram	4
2	Cost data flow through the cost analysis system	7

ACKNOWLEDGMENTS

The staff of Kenton County, Kentucky, Water District No. 1 participated in the development and implementation of the cost analysis system described in this report. The cooperation and active support of Mr. Victor C. Fender, General Manager, and Mr. Malvern Connell, Office Manager, are gratefully acknowledged.

SECTION 1

INTRODUCTION

Volume I describes the water supply cost analysis system in detail from concept to final report generation. Volume II explains the computer operations and is intended to be used along with Volume I.

From its inception, the water supply cost analysis system was designed to have broad applications to the water supply industry while maintaining the capability of performing detailed checks of the input data and handling that data to meet the specific needs of a utility. This adds up to the requirement that the cost analysis system be both general and specific at the same time. In addition, it was recognized that the program must have the built-in capability to interface with many different utilities each with a different operating configuration and financial system.

Based on experience in examining the financial records of over 50 water utilities, it was also recognized that most utilities do not identify cost to the level required by the cost analysis system, and it is difficult to attempt a rapid change in the total accounting system of a utility during the implementation of a cost analysis module. Therefore, it was necessary to design, within the cost analysis system, flexible inputs and the capability of allocating costs when they are not specifically identified.

To accomplish the objectives of the cost analysis system, all cost input information must come from the utility's financial documentation. Since this information will vary from utility to utility, a concept was established for developing interface programs unique to each utility. These interface programs will accept the information in a specific utility format and convert it to a standardized format. However, the cost analysis system program cannot be entirely standard because each utility maintains a different chart of accounts and cost centers. For this reason, it is necessary to adjust the standard program to contain the chart of accounts and the cost centers on a utility-by-utility basis. Once this is accomplished, the program will handle all utilities in a standard mode while producing reports compatible with each utility's particular needs.

As described in Volume I, from a formatting standpoint, the principal variation between reports of different utilities will be in the Level IV and II reports which will reflect specific utility cost centers and chart of accounts. Level I and III reports will appear exactly the same in

format but will be produced with each utility's individual data.

The following sections of this report provide the details of the cost analysis operating system.

SECTION 2

COST ANALYSIS SYSTEM COMPUTER OPERATIONS

SYSTEM FLOW DIAGRAMS

Figure 1 (a,b,c) represents the overall routine monthly operating flow diagram of the cost analysis system. In general, the cost analysis system takes the source input data, verifies it, allocates or assigns it to the proper cost centers, and produces the reports. There are actually fourteen separate modules of programs performing the necessary functions of the analysis and six modules performing the interface functions. The interface programs receive the input data in various formats, edit the data for validity, and condition the data for entry into the cost analysis system. It is from these interface programs that reports are generated to link the automated cost analysis system with the Kenton County Water District's financial reporting system. The interface edit programs perform such functions as verifying the validity of each account number to assure that it is appropriate, monitoring each cost for correct designation to a cost center, and allocating costs not assigned to proper cost centers. These cost elements are stored in a large matrix from which each element is withdrawn and utilized in the various reports. Figure 2 shows the types of information input to the system, the flow, and application of that information within the system. In addition, the figure includes a brief description of the types of reports generated.

COMPUTER PROGRAM DIRECTORY

The purpose of this directory is to record the location of each program within the Operating Instructions, Flow Diagrams, and Program Listings Sections of this report so that each program may be analyzed individually in its entirety.

The programs listed are in the order used to create the system and are not necessarily in their routine operating order. That order is recorded in the Routine Operating Sequence part of Section 2.

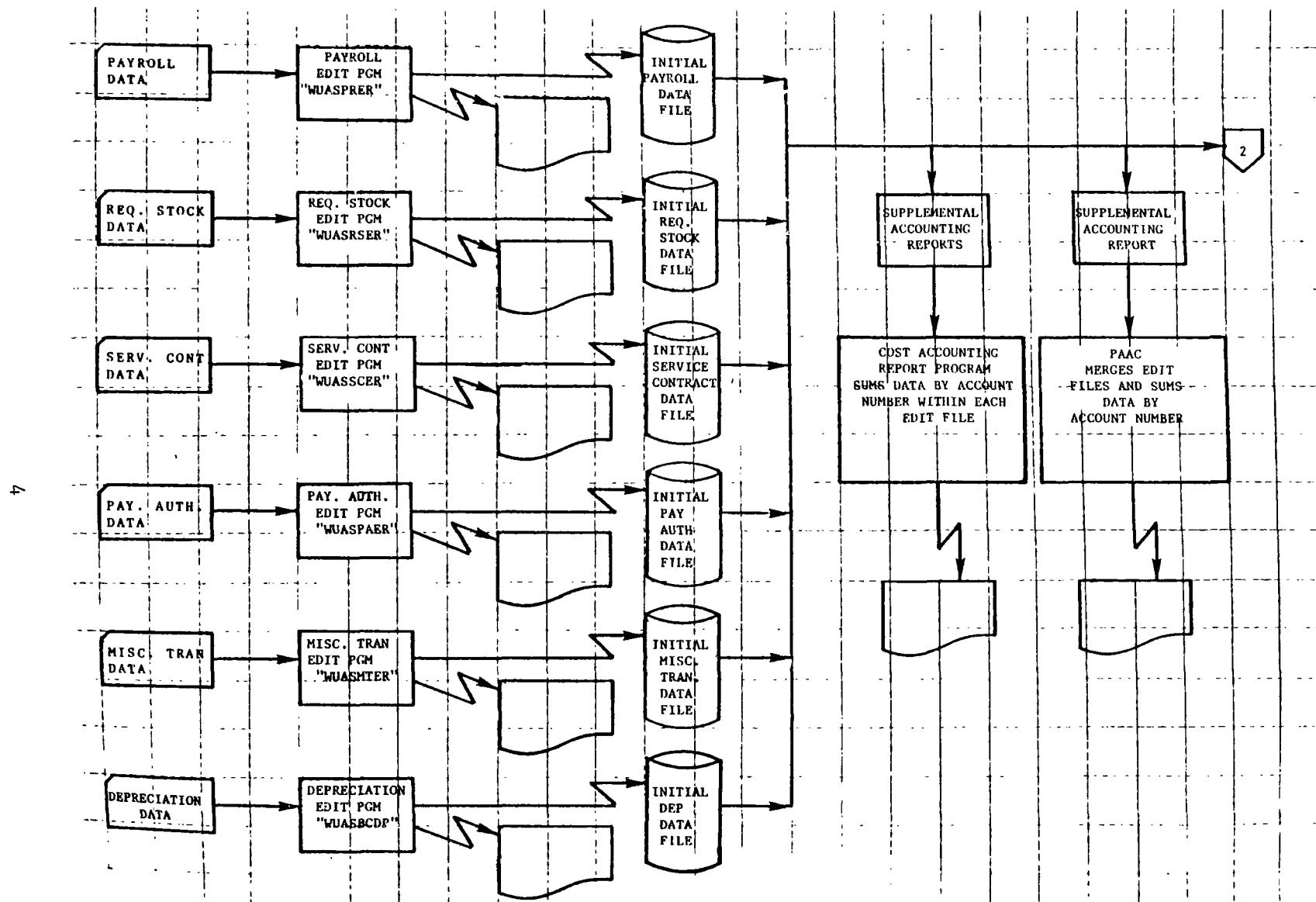


Figure 1(a). Cost analysis system routing operation flow diagram.

5

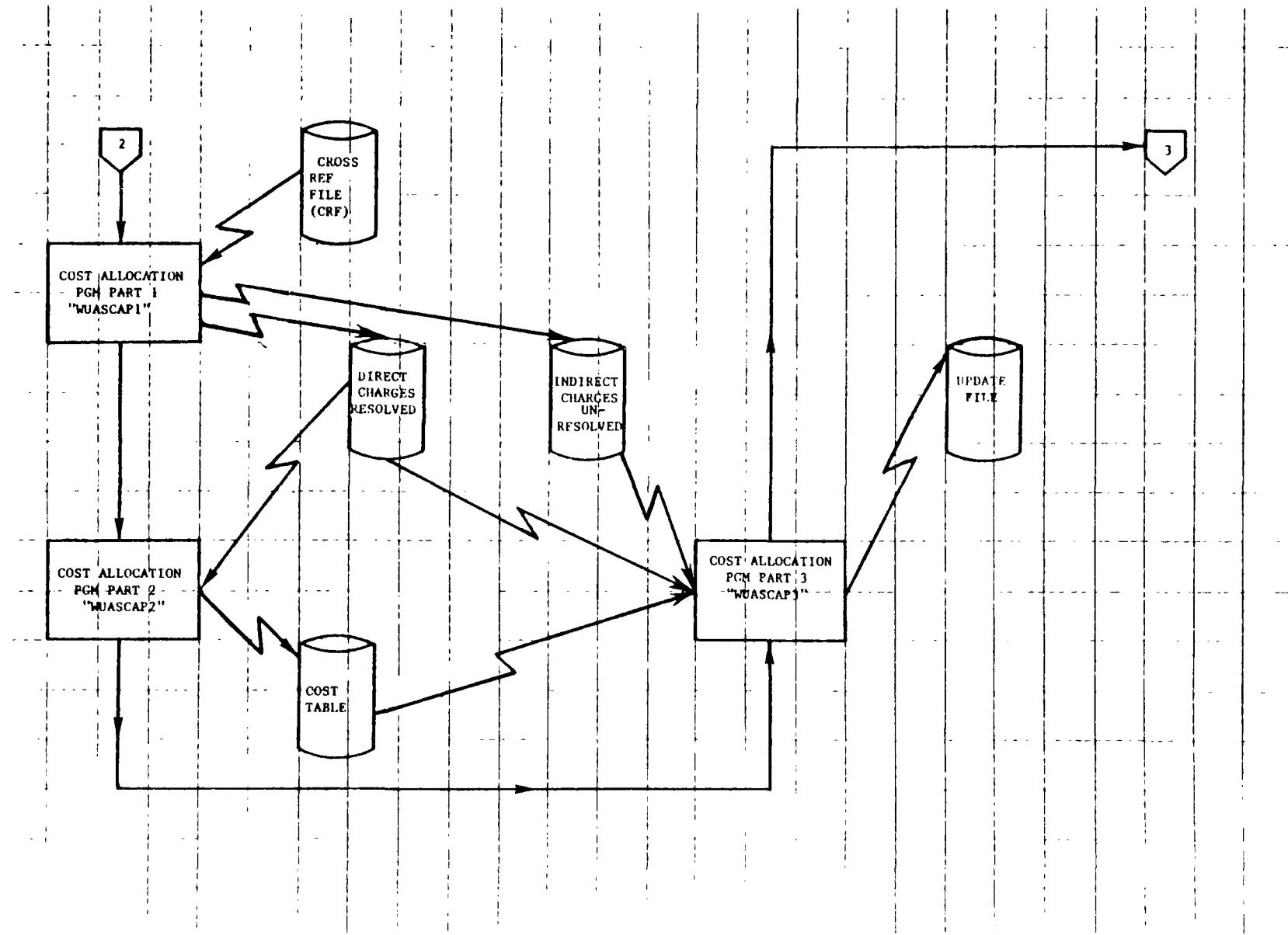


Figure 1(b)

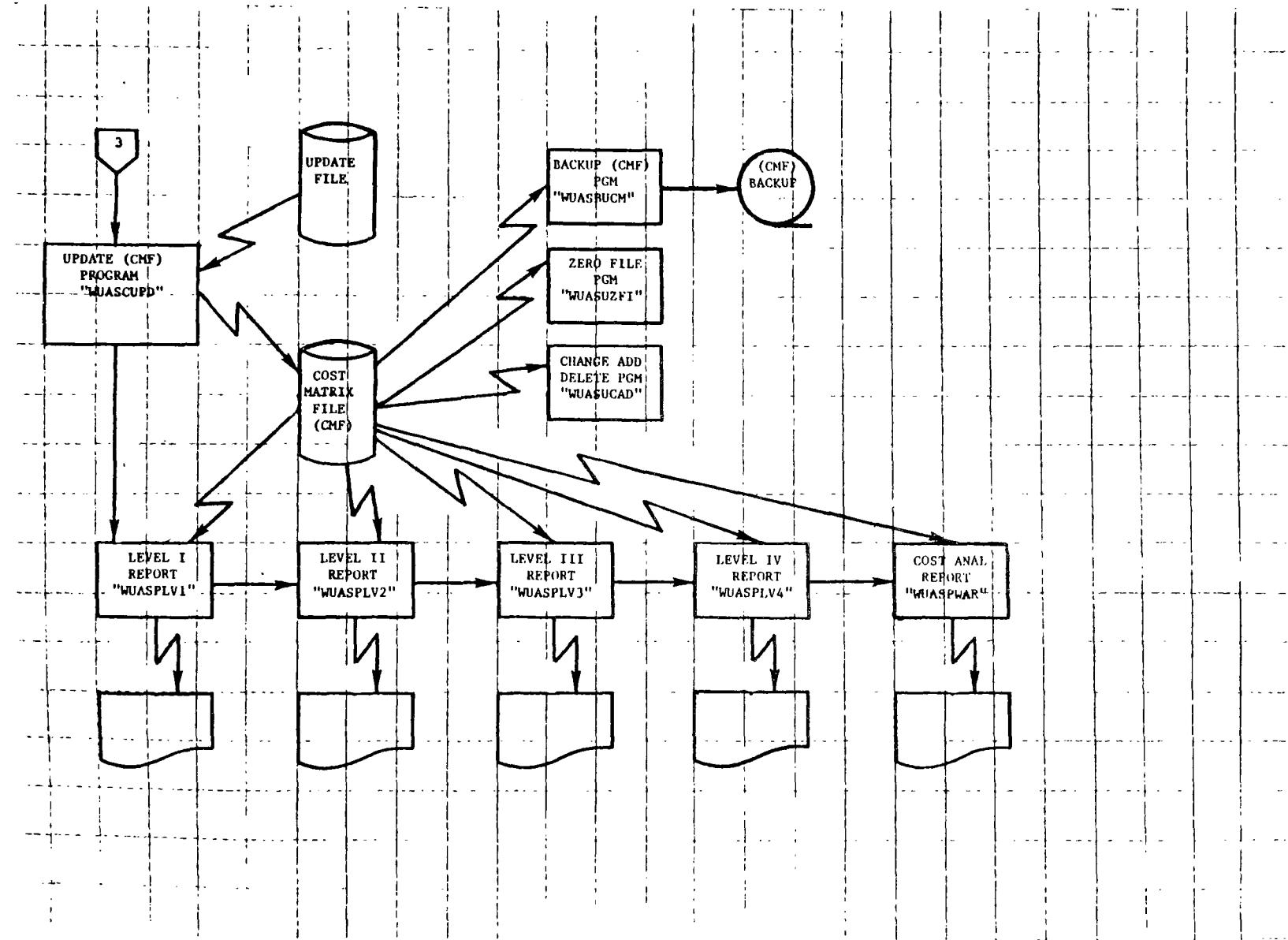


Figure 1(c)

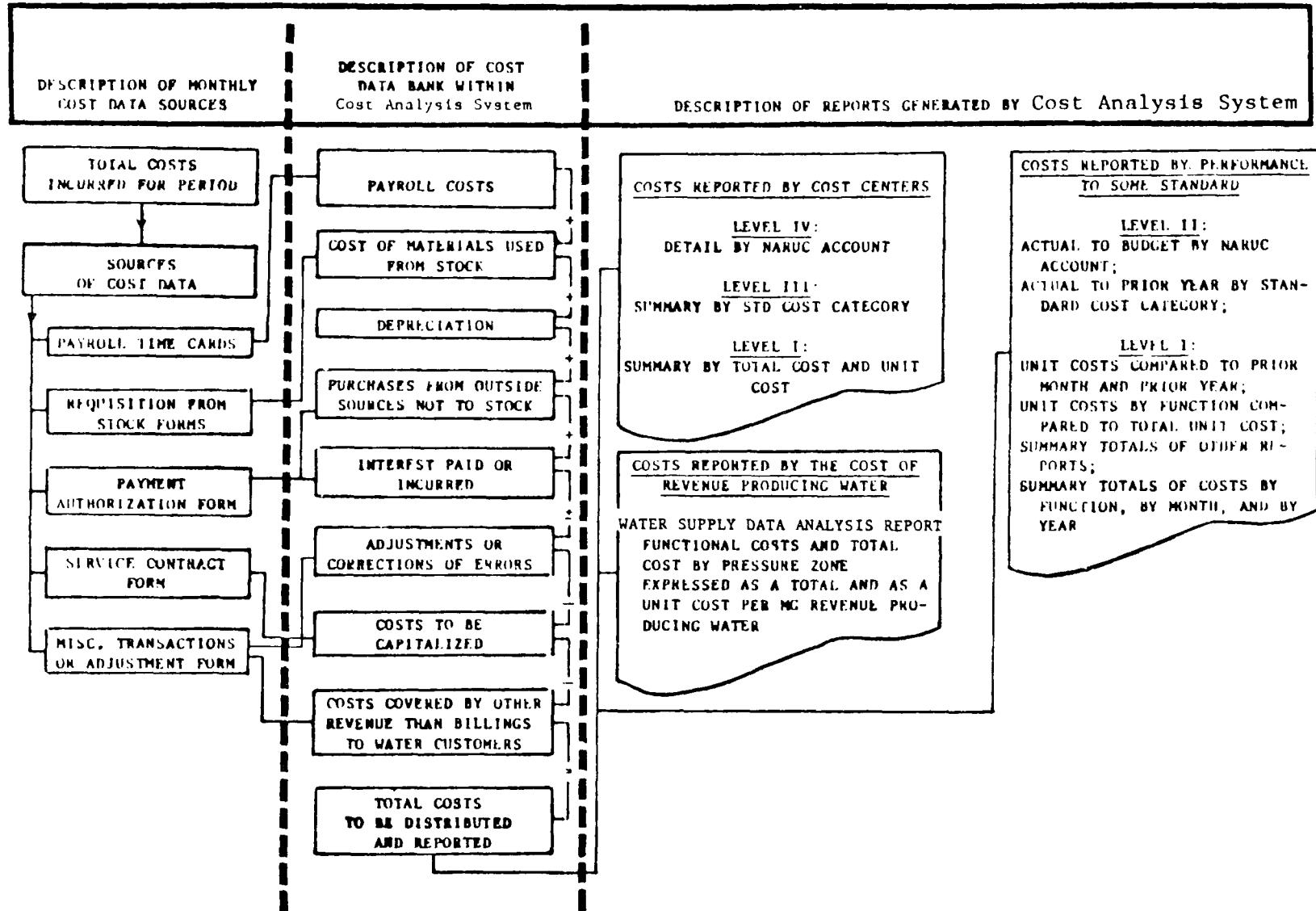


Figure 2. Cost data flow through the cost analysis system.

PROGRAM	SECTION 2 OPERATING INSTRUCTIONS	SECTION 4 FLOW DIAGRAMS	SECTION 5 PROGRAM LISTINGS
	Page No.	Page No.	Page No.
1. WUASBCCT	9	78	125
2. WUASBSAT	10	79	130
3. WUASBFAT	12	80	134
4. WUASBCRF	13	82	140
5. WUASBCMF	14	83	148
6. WUASBDPT	15	85	157
7. WUASBCDP	16	86	161
8. WUASSCER	17	88	167
9. WUASMTER	18	90	174
10. WUASPAER	19	92	182
11. WUASRSER	20	94	191
12. WUASPRER	21	96	200
13. WUASCARP	22	98	209
14. WUASPAAC	23	99	214
15. WUASCAP1	24	101	219
16. WUASCAP2	25	103	227
17. WUASCAP3	26	104	231
18. WUASUZFI	27	106	242
19. WUASCUPD	29	108	245
20. WUASPLV4	30	109	249
21. WUASPLV3	32	114	284
22. WUASPLV2	33	115	292
23. WUASPLV1	34	116	311
24. WUASPWAR	36	119	318
25. WUASBUCM	37	120	331
26. WUASUCAD	38	121	337

SYSTEM OPERATING INSTRUCTIONS

The purpose of this part of the report is to provide a general description of each program and give specific keypunching and system operating instructions. All routine listings should normally be printed on one-part paper; however, the number of report copies should be determined by the utility's requirements.

Operating instructions for each program begin with an objective or definition, and then the instructions are broken down into keypunching and operation instructions.

By following these instructions, the cost analysis system can be operated successfully. If a variation in procedure is encountered, contact the System Programmer.

"WUASBCCT"

OBJECTIVE: THIS PROGRAM CREATES A FILE THAT CONTAINS ALL THE VALID COST CENTERS AND THEIR DESCRIPTIONS USED IN THE COST ANALYSIS SYSTEM.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA USED TO BUILD THE COST CENTER TABLE FILE.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE THE DATE-TITLE CARD FOLLOWED BY THE COST CENTERS DATA BEHIND THE "//SYSIN DD *" CARD IN THE "WUASBCCT" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER LISTING THE DATA FILE.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASBSAT"

OBJECTIVE: THIS PROGRAM IS USED TO CREATE SPECIAL TABLE FILES THAT ARE USED THROUGHOUT THE COST ANALYSIS SYSTEM TO VERIFY OR REFERENCE CERTAIN DATA. CURRENTLY, THERE ARE THREE FILES THAT ARE CREATED BY THIS PROGRAM.

- (1). BSAT.CCC COST CENTER CONVERSION:
 USED TO CONVERT TWO-DIGIT COST CENTERS TO
 FOUR-DIGIT COST CENTERS.
- (2). BSAT.CCT COST CATEGORY TABLE:
 USED TO VERIFY THE EXISTENCE OF A VALID
 COST CATEGORY AND THE NAME OF THE
 CATEGORY.
- (3). BSAT.FAT FUND TITLE TABLE:
 USED TO VERIFY AND PROVIDE VALID
 FUND TITLES REQUIRED IN THE (CMF)
 BUILDING PROGRAM WUASBCMF.

OPERATING INSTRUCTIONS

- (A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see section 3)
 - (1). ONE DATE-TITLE CARD.
 - (2). DATA USED TO BUILD THE SPECIFIC FILE.

"WUASBSAT" (continued)

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE THE DATE-TITLE CARD FOLLOWED BY THE
SPECIFIC FILE DATA BEHIND THE "//SYSIN DD *" CARD
IN THE "WUASBSAT" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER LISTING THE FILE.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASBFAT"

OBJECTIVE: THIS PROGRAM CREATES A FILE THAT IS USED IN BUILDING THE COST MATRIX FILE AND THE CROSS REFERENCE FILE. IN ADDITION, IT IS USED TO VERIFY THE EXISTENCE OF VALID ACCOUNT NUMBERS IN THE ANALYSIS SYSTEM. THIS FILE CONTAINS THE ACCOUNT NAME IN TWO (2) 20-CHARACTER FIELDS.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA USED TO BUILD THE FINANCIAL ACCOUNT TABLE FILE.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE THE DATE-TITLE CARD FOLLOWED BY THE FINANCIAL DATA BEHIND THE "//SYSIN DD *" CARD IN THE "WUASBFAT" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER LISTING THE FINANCIAL ACCOUNT TABLE FILE.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASBCRF"

OBJECTIVE: THIS PROGRAM CREATES A FILE THAT IS USED TO ALLOCATE COSTS TO COST CENTERS. IN ADDITION, THIS FILE IS USED BY THE "WUASBCMF" PROGRAM TO CREATE THE COST MATRIX FILE.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see section 3)

- (1). ONE DATE-TITLE CARD.
- (2). CROSS REFERENCE DATA NEEDED TO CREATE THE CROSS REFERENCE FILE (CRF).

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE THE DATE-TITLE CARD FOLLOWED BY THE CRF DATA BEHIND THE "//SYSIN DD *" CARD IN THE "WUASBCRF" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER LISTING THE CROSS REFERENCE FILE.
- (4). THIS PROGRAM CAN BE RUN AS MANY TIMES AS REQUIRED BUT SHOULD ONLY HAVE TO BE RERUN WHEN NEW ACCOUNT NUMBERS AND/OR NEW COST CENTERS ARE ADDED TO THE COST ANALYSIS SYSTEM.

"WUASBCMF"

OBJECTIVE: THIS PROGRAM CREATES THE COST MATRIX FILE. GENERALLY, THIS PROGRAM SHOULD BE USED ONLY ONCE TO INITIATE THE FILE. HOWEVER, IF THE FILE IS DELETED, THE ORIGINAL FILE CAN BE RE-CREATED AND UPDATED FROM THE MOST CURRENT BACKUP TAPE USING PROGRAM "WUASUCAD".

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see section 3)

- (1). DATE-TITLE CARD.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). NOTE: MAKE SURE THE PROGRAM "WUASBCRF" IS SUCCESSFULLY COMPLETED PRIOR TO RUNNING THIS JOB BECAUSE THE FILE BUILT IN THAT PROGRAM IS USED TO CREATE THIS FILE.
- (2). PLACE THE DATE-TITLE CARD BEHIND THE "//SYSIN DD *" CARD IN THE "WUASBCMF" JCL DECK.
- (3). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (4). OUTPUT WILL BE ON ONE-PART PAPER.
- (5). LISTING WILL SHOW WHAT THE FILE CONTAINS.
- (6). THIS PROGRAM CAN BE RUN AS MANY TIMES AS REQUIRED.

"WUASBDPT"

OBJECTIVE: THIS PROGRAM IS USED TO CREATE A TABLE FILE OF THE YEARLY DEPRECIATION AMOUNTS ASSIGNED TO VARIOUS COST CENTERS. THE TABLE FILE WILL THEN BE USED IN THE PROGRAM "WUASBCDP" WHICH CONVERTS THE YEARLY DEPRECIATION AMOUNTS INTO A CURRENT DEPRECIATION AMOUNT.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA FOR BUILDING THE DEPRECIATION TABLE.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE THE DATE-TITLE CARD FOLLOWED BY THE DEPRECIATION AMOUNTS BEHIND THE "//SYSIN DD *" CARD IN THE "WUASBDPT" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER LISTING THE THE DEPRECIATION AMOUNTS.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED BUT IS USUALLY RUN ONCE OR UNTIL A COST CENTER DEPRECIATION VALUE CHANGES.

"WUASBCDP"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ THE FILE WHICH CONTAINS THE YEARLY DEPRECIATION COSTS BY COST CENTERS THEN CALCULATE THE CURRENT PERIOD DEPRECIATION VALUES (MONTHLY, QUARTERLY, ETC.).

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE SPECIAL DATE-TITLE CARD.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD BEHIND THE "//SYSIN DD *"
CARD IN THE "WUASBCDP" JCL DECK.
- (2). PUT DECK IN CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS
REQUIRED.

"WUASSCER"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ AND EDIT THE SERVICE CONTRACT SOURCE DATA CARDS. THIS PROGRAM BUILDS A DATA FILE USED IN THE COST ANALYSIS SYSTEM.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA FROM SERVICE CONTRACT SOURCE DOCUMENTS SHOULD BE PUNCHED AND VERIFIED.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWED BY THE SERVICE CONTRACT DATA BEHIND THE "//SYSIN DD *" CARD IN THE "WUASSCER" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). UNDER THE COMMENT COLUMN OF THE LISTING, DESCRIPTIVE COMMENTS WILL BE PRINTED CONCERNING ERROR(S) FOUND IN THE DATA.
- (5). IF ANY ERRORS WERE DETECTED, CORRECT THEM AND GO BACK TO (1).
- (6). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASMTER"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ AND EDIT THE MISCELLANEOUS TRANSACTIONS SOURCE DATA CARDS. THIS PROGRAM BUILDS A DATA FILE USED IN THE COST ANALYSIS SYSTEM.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA FROM MISCELLANEOUS TRANSACTIONS SOURCE DOCUMENTS SHOULD BE PUNCHED AND VERIFIED.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWED BY THE MISCELLANEOUS TRANSACTIONS DATA BEHIND THE " //SYSIN DD *" CARD IN THE "WUASMTER" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). UNDER THE COMMENT COLUMN OF THE LISTING, DESCRIPTIVE COMMENTS WILL BE PRINTED CONCERNING ERROR(S) FOUND IN THE DATA.
- (5). IF ANY ERRORS WERE DETECTED, CORRECT THEM AND GO BACK TO (1).
- (6). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASPAER"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ AND EDIT THE PAYMENT AUTHORIZATIONS SOURCE DATA CARDS. THIS PROGRAM BUILDS A DATA FILE USED IN THE COST ANALYSIS SYSTEM.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA FROM PAYMENT AUTHORIZATION SOURCE DOCUMENTS SHOULD BE PUNCHED AND VERIFIED.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWED BY THE PAYMENT AUTHORIZATION DATA BEHIND THE "//SYSIN DD *" CARD IN THE "WUASPAER" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). UNDER THE COMMENT COLUMN OF THE LISTING, DESCRIPTIVE COMMENTS WILL BE PRINTED CONCERNING ERROR(S) FOUND IN THE DATA.
- (5). IF ANY ERROR(S) WERE DETECTED, CORRECT THEM AND GO BACK TO (1).
- (6). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASRSER"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ AND EDIT THE REQUISITION FROM STOCK SOURCE DATA CARDS. THIS PROGRAM BUILDS A DATA FILE USED IN THE COST ANALYSIS SYSTEM.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA FROM STOCK REQUISTITION SOURCE DOCUMENTS SHOULD BE PUNCHED AND VERIFIED.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWED BY THE REQUISITION FROM STOCK DATA BEHIND THE "/SYSIN DD *" CARD IN THE "WUASPRSER" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). UNDER THE COMMENT COLUMN OF THE LISTING, DESCRIPTIVE COMMENTS WILL BE PRINTED CONCERNING ERROR(S) FOUND IN THE DATA.
- (5). IF ANY ERROR(S) WERE DETECTED, CORRECT THEM AND GO BACK TO (1).
- (6). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASPRER"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ AND EDIT THE PAYROLL SOURCE DATA CARDS. THIS PROGRAM BUILDS A DATA FILE USED IN THE COST ANALYSIS SYSTEM.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). DATA FROM PAYROLL SOURCE DOCUMENTS SHOULD BE PUNCHED AND VERIFIED.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWED BY THE PAYROLL DATA BEHIND THE "//SYSIN DD *" CARD IN THE "WUASPRER" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). UNDER THE COMMENT COLUMN OF THE LISTING, DESCRIPTIVE COMMENTS WILL BE PRINTED CONCERNING ERROR(S) FOUND IN THE DATA.
- (5). IF ANY ERROR(S) WERE DETECTED, CORRECT THEM AND GO BACK TO (1).
- (6). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASCARP"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ THE FILES CREATED BY THE INITIAL EDIT PROGRAMS AND THEN LIST THE COST BY ACCOUNT NUMBER WITHIN TYPE OF DATA (PAYROLL, SERVICE CONTRACT, ETC.).

OPERATING INSTRUCTIONS

(A) KEY PUNCH: REQUIRE NO DATE-TITLE OR DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PUT THE "WUASCARP" JCL DECK IN THE CARD READER AND READ THE DECK.
- (2). OUTPUT SHOULD BE ON MULTI-PART PAPER DEPENDING ON THE NUMBER OF REQUIRED COPIES.
- (3). UNDER THE COMMENT COLUMN OF THE LISTING, DESCRIPTIVE COMMENTS WILL BE PRINTED INDICATING ERROR(S) FOUND IN THE DATA.
- (4). IF ERRORS ARE DETECTED, THE CORRECTIONS MUST BE MADE TO THE INITIAL INPUT DATA. THE APPROPRIATE EDIT PROGRAM MUST THEN BE RERUN.
- (5). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASPAAC"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO SUM, WITHIN ACCOUNT NUMBER ORDER, THE INPUT DATA FILES CREATED BY THE INITIAL EDIT PROGRAMS.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD BEHIND THE "//SYSIN DD *" CARD IN THE "WUASPAAC" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT CAN BE ON MULTI-PART PAPER.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASCAP1"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ THE FILES CREATED BY THE INITIAL EDIT PROGRAMS AND MERGE THESE FILES INTO TWO FILES; ONE FILE CONTAINING RECORDS OF DIRECT COST CENTERS AND CHARGES, AND THE OTHER FILE CONTAINING THE INDIRECT CHARGES.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). REQUIRES NO DATE-TITLE CARD INPUT.
- (2). ONE UTILITY DESCRIPTION CARD. (UTILITY NAME OR NUMBER.)

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE UTILITY DESCRIPTION CARD BEHIND THE "//SYSIN DD *" CARD IN THE "WUASCAP1" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASCAP2"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ THE DIRECT CHARGES FILE (RESOLVED COSTS) CREATED BY THE "WUASCAP1" PROGRAM, SUM WITHIN COST CENTERS, AND CREATE A FILE THAT CONTAINS THE TOTAL AMOUNT CHARGED TO THE COST CENTERS.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWING THE "//SYSIN DD *"
CARD IN THE "WUASCAP2" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASCAP3"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO READ THE FILE CREATED BY THE "WUASCAP2" PROGRAM AND, USING A PREDETERMINED ALLOCATION METHOD, DETERMINE THE PERCENTAGES ASSOCIATED WITH EACH COST CENTER. WHEN THE RATIO OF DIRECT CHARGES TO TOTAL CHARGES IS DETERMINED, THE NON-DIRECT COST CREATED IN THE "WUASCAP1" PROGRAM ARE ALLOCATED TO COST CENTERS. THE NEW RECORDS ARE THEN ADDED TO THE RESOLVED COST CENTERS FILE CREATED IN "WUASCAP1".

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWING THE "//SYSIN DD *"
CARD IN THE "WUASCAP2" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER.
- (4). THIS PROGRAM MAY NOT BE RERUN. IF PROGRAM IS NOT
COMPLETED, YOU MUST START OVER AGAIN AT THE
"WUASCAP1" LEVEL.

DO NOT RERUN THIS PROGRAM.

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO ADJUST OR ELIMINATE COST BY ACCOUNT NUMBER IN CERTAIN DATA FIELDS THROUGHOUT THE COST MATRIX FILE (CMF). THIS PROGRAM MOVES THE CURRENT PERIOD'S COST TO THE PREVIOUS PERIOD'S COST FIELDS AND THEN ZEROS THE CURRENT PERIOD'S COST FIELD. THE PROGRAM CAN ALSO MAKE OTHER CHANGES TO THE (CMF) BY USING ONE OF THE FOLLOWING OPTION CARDS:

- (1) BY INCLUDING A "PHYSICAL" OPTION CARD, IN ADDITION TO ACCOMPLISHING THE ABOVE ADJUSTMENT, THE CURRENT YEAR-TO-DATE COST WILL BE MOVED TO THE PREVIOUS YEAR-TO-DATE COST FIELDS AND THEN THE CURRENT YEAR-TO-DATE COST FIELDS ARE ZEROED.
- (2) BY INCLUDING A "ZERO" OPTION CARD, ALL THE COST DATA FIELDS WILL BE ZEROED. (SHOULD BE USED NORMALLY FOR INITIAL SYSTEM TESTING PERIOD).

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED. (see Section 3)

- (1) NO DATE-TITLE CARD IS NEEDED.
- (2) FOR "PHYSICAL" OPTION--PUNCH THE WORD "PHYSICAL" IN THE FIRST SEVEN POSITIONS OF THE DATA CARD.
- (3) FOR "ZERO" OPTION--PUNCH THE WORD "ZERO" IN THE FIRST FOUR POSITIONS OF THE DATA CARD.

"WUASUZFI" (continued)

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

NOTE: USE CARE WHEN RUNNING THIS JOB, AS FILE COULD BE ERASED IN ERROR. THIS JOB MUST BE RUN PRIOR TO RUNNING THE PROGRAM "WUASCUPD" WHICH IS THE UPDATE UTILITY PROGRAM.

- (1). IF USING AN OPTION CARD, PLACE THE OPTION CARD AFTER THE "//SYSIN DD *" CARD IN THE "WUZSUZFI" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON ONE-PART PAPER. ONLY A MESSAGE WILL BE PRINTED STATING WHAT TYPE OF OPTION WAS USED AND IF IT WAS COMPLETED.
- (4). DO NOT RERUN THIS JOB...IF JOB SHOULD NOT BE COMPLETED OR THE WRONG OPTION WAS USED, THE FILE WILL HAVE TO BE REBUILT.

"WUASCUPD"

OBJECTIVE: THE PURPOSE OF THIS PROGRAM IS TO UPDATE THE COST MATRIX FILE (CMF) WITH CURRENT VALUES USING THE DIRECT CHARGES FILE CREATED BY THE "WUASCAP3" PROGRAM.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: REQUIRES NO DATE-TITLE OR DATA CARD(S) INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

NOTE: MAKE SURE PROGRAM "WUASUZFI" WAS EXECUTED PRIOR TO BEGINNING THIS JOB.

- (1). PLACE THE "WUASCUPD" JCL DECK IN THE CARD READER AND READ THE DECK.
- (2). OUTPUT CONSISTING OF A MESSAGE INFORMING THE OPERATOR THAT THE COST MATRIX FILE HAD BEEN UPDATED WILL BE ON ONE-PART PAPER.
- (3). DO NOT RERUN THIS JOB. IF JOB ABORTS, CONTACT PROGRAMMER OR DATA BASE MANAGER BEFORE ATTEMPTING FURTHER ACTION.

"WUASPLV4"

OBJECTIVE: THIS LEVEL 4 REPORT PROGRAM IS BROKEN DOWN INTO THE FOLLOWING FOUR SEPARATE PROGRAMS: "WUASLV41", "WUASLV42", "WUASLV43", AND "WUASLV44". THE PROGRAM READS THE COST MATRIX FILE AND PRODUCES A LISTING OF THE COSTS BY ACCOUNT NUMBERS WITHIN COST CENTERS.

THE "WUASLV41" PROGRAM LISTS ALL COSTS PERTAINING TO THE ACQUISITION FUNCTION.

THE "WUASLV42" PROGRAM LISTS ALL COSTS PERTAINING TO THE TREATMENT FUNCTION.

THE "WUASLV43" PROGRAM LISTS ALL COSTS PERTAINING TO THE DELIVERY FUNCTION.

THE "WUASLV44" PROGRAM LISTS ALL COSTS PERTAINING TO THE SUPPORT SERVICE FUNCTION.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD IS NEEDED FOR EACH PROGRAM.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARDS IN THE "WUASLV41", "WUASLV42", "WUASLV43", AND "WUASLV44" JCL DECKS.

"WUASPLV4" (continued)

- (2). PUT THE DECKS IN THE CARD READER AND READ THE DECKS.
- (3). OUTPUT WILL BE ON MULTI-PART PAPER DEPENDING UPON
THE NUMBER OF REQUIRED COPIES.
- (4). THESE PROGRAMS MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASPLV3"

OBJECTIVE: THIS LEVEL 3 REPORT PROGRAM READS THE COST MATRIX FILE, THEN SUMS THE CURRENT COSTS AND YTD COST BY COST CATEGORIES WITHIN FUNCTIONS OF ACQUISITION, TREATMENT, DELIVERY, SUPPORT SERVICES, TAXES, AND INTEREST.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD BEHIND THE "//SYSIN DD *" CARD IN THE "WUASPLV3" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON MULTI-PART PAPER DEPENDING UPON THE NUMBER OF COPIES REQUIRED.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASPLV2"

OBJECTIVE: THIS LEVEL 2 REPORT PROGRAM READS THE COST MATRIX FILE AND PRODUCES A TWO-PART LISTING. THE FIRST PART IS A COST COMPARISON OF THE ACTUAL COSTS TO THE BUDGET BY ACCOUNT NUMBERS. THE SECOND PART IS A COST COMPARISON OF ACTUAL COSTS TO THE PREVIOUS YEAR'S COST.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). REQUIRES NO DATA CARD INPUT.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD BEHIND THE "//SYSIN DD *" CARD IN THE "WUASPLV2" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON MULTI-PART PAPER DEPENDING UPON THE NUMBER OF COPIES REQUIRED.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASPLV1"

OBJECTIVE: THIS LEVEL 1 REPORT PROGRAM READS THE COST MATRIX FILE, THEN SUMS THE CURRENT COSTS AND YTD COSTS BY COST CATEGORIES WITHIN FUNCTIONS OF ACQUISITION, TREATMENT, DELIVERY, SUPPORT SERVICES, TAXES AND INTEREST. THIS PROGRAM USES THESE VALUES TO PRODUCE A LIST OF THE COSTS IN TERMS OF WATER PRODUCTION UNITS.
(USUALLY THE UNIT IS IN MILLIONS OF GALLONS.)

OPERATING INSTRUCTIONS:

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). ONE DATA CARD CONTAINING THE AMOUNT OF WATER USED FOR THE PREVIOUS MONTH IS NEEDED.
- (3). ONE DATA CARD CONTAINING THE AMOUNT OF WATER USED FOR THE CURRENT MONTH IS NEEDED.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

NOTE: THIS PROGRAM CANNOT BE RUN UNTIL LEVEL 2 REPORT HAS BEEN SUCCESSFULLY COMPLETED SINCE PART OF THE DATA FOR THIS PROGRAM IS TAKEN FROM LEVEL 2 (DATA TAKEN FROM LEVEL 2 = ACTUAL TO BUDGET PERCENT).

- (1). PLACE THE DATE-TITLE CARD FOLLOWED BY THE PREVIOUS MONTH WATER CARD AND THE CURRENT MONTH WATER CARD BEHIND THE "//SYSIN DD *" CARD IN THE "WUASPLV1" JCL DECK.

"WUASPLV1" (continued)

- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE ON MULTI-PART PAPER DEPENDING UPON THE NUMBER OF COPIES REQUIRED.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASPCAR"

OBJECTIVE: THIS WATER SUPPLY DATA ANALYSIS PROGRAM READS THE COST MATRIX FILE, ANALYZES THE COST ASSOCIATED WITH WATER PRODUCTION WITHIN VARIOUS ZONES IN THE WATER UTILITY SYSTEM, AND THEN PRINTS A REPORT.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). ONE CURRENT ZONE-USAGE DATA CARD.
- (3). ONE YTD ZONE-USAGE DATA CARD.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE DATE-TITLE CARD FOLLOWED BY THE CURRENT AND YTD DATA CARDS BEHIND THE "//SYSIN DD *" CARD IN THE "WUASPCAR" JCL DECK.
- (2). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (3). OUTPUT WILL BE PRINTED ON MULTI-PART PAPER DEPENDING UPON THE NUMBER OF REQUIRED COPIES.
- (4). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASBUCM"

OBJECTIVE: THIS PROGRAM IS USED TO LIST AND/OR BACK UP THE COST MATRIX FILE (CMF) ONTO A TAPE FILE. THE PROGRAM WILL NORMALLY PRODUCE A LISTING OF THE DATA ON THE CMF, HOWEVER, BY INSERTING AN OPTION CARD THE FILE WILL BE ALSO BACKED UP ON TAPE.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). ONE OPTION CARD CONTAINING THE WORD "BACK-UP" IN THE FIRST SEVEN POSITIONS OF THE CARD IF ACTUAL BACK-UP IS TO TAKE PLACE. OTHERWISE, NO OPTION CARD REQUIRED.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

- (1). PLACE THE DATE-TITLE CARD FOLLOWING THE "//SYSIN DD *" CARD IN THE "WUASBUCM" JCL DECK.
- (2). OPTION: IF COST MATRIX FILE IS TO BE BACKED UP ON TAPE THEN PLACE THE OPTION CARD AFTER THE DATE-TITLE CARD.
- (3). PUT THE DECK IN THE CARD READER AND READ THE DECK.
- (4). OUTPUT WILL BE ON ONE-PART PAPER.
- (5). THIS PROGRAM MAY BE RUN AS MANY TIMES AS REQUIRED.

"WUASUCAD"

OBJECTIVE: THIS PROGRAM IS USED FOR MAKING CHANGES, ADDITIONS, AND/OR DELETIONS TO RECORDS IN THE COST MATRIX FILE (CMF). IN ADDITION, THE (CMF) MAY BE RE-CREATED FROM A (CMF) BACKUP TAPE.

OPERATING INSTRUCTIONS

(A) KEY PUNCH: THE FOLLOWING INPUT DATA ARE REQUIRED: (see Section 3)

- (1). ONE DATE-TITLE CARD.
- (2). ONE OPTION CARD WITH THE WORD '****UPDATE' IN THE FIRST 10 POSITIONS OF THE CARD.
- (3). DATA FOR MAKING CHANGES, ADDITIONS, AND/OR DELETIONS. DATA RECORDS (CARDS) MUST BE IN THE EXACT IMAGE OF THE COST MATRIX FILE RECORDS.

(B) OPERATION: PROGRAM OPERATION/INFORMATION IS AS FOLLOWS:

NOTE: BY LEAVING OUT THE '****UPDATE' CARD, THE FILE WILL NOT BE PHYSICALLY CHANGED. ONLY A SIMULATION OF UPDATING WILL TAKE PLACE. THIS WILL ALLOW FOR A TEST RUN ON LARGE AMOUNTS OF DATA TO REVIEW CHANGES.

THERE ARE TWO (2) DIFFERENT OPERATING PROCEDURES AVAILABLE TO BE USED IN THIS PROGRAM, AS DESCRIBED BELOW.

(1) CHANGING, ADDING, DELETING PROCEDURE.

IN ORDER TO CHANGE A RECORD, THE ACCOUNT NUMBER, COST CENTER,

"WUASUCAD" (continued)

AND COST CATEGORY FIELDS MUST BE IDENTICAL TO THOSE OF THE RECORD BEING CHANGED. IF THESE KEY FIELDS ARE INVALID OR CANNOT BE LOCATED WITHIN THE COST MATRIX FILE, THE CHANGE WILL BE CONSIDERED A NEW RECORD AND WILL BE ADDED TO THE EXISTING FILE. EACH AMOUNT FIELD MUST CONTAIN A VALUE, OTHERWISE, IF LEFT BLANK, THE FIELD WILL BE ZEROED.

IN ORDER TO ADD A NEW RECORD, THE ACCOUNT NUMBER, COST CENTER, AND COST CATEGORY FIELDS MUST BE COMPLETED. THE AMOUNT FIELDS SHOULD CONTAIN WHATEVER VALUES ARE REQUIRED. IN ADDITION, THE NEW RECORD MUST INCLUDE A UNIQUE MATCH CODE IN ORDER TO POSITION THE INFORMATION WITHIN THE OUTPUT REPORTS.

IN ORDER TO DELETE AN EXISTING RECORD, THE ACCOUNT NUMBER, COST CENTER, AND COST CATEGORY FIELDS MUST BE IDENTICAL TO THOSE OF THE RECORD BEING DELETED. ALL OTHER FIELDS MUST BE BLANK.

TO OPERATE THE CHANGE, ADD, OR DELETE PROGRAM:

- (a) PLACE THE DATE-TITLE AND THEN THE OPTION CARD, IF USED, FOLLOWED BY THE DATA CARDS BEHIND THE "//SYSIN DD *" CARD IN THE "WUASUCAD" JCL DECK.
- (b) PUT THE DECK IN THE CARD READER AND READ THE DECK.

"WUASUCAD" (continued)

- (c) OUTPUT WILL BE ON ONE-PART PAPER LISTING ALL CHANGES, ADDS, OR DELETES. A COMMENT WILL STATE WHAT ACTION WAS TAKEN FOR EACH RECORD (i.e. RECORD CHANGED, ADDED, OR DELETED). IN ADDITION AT THE END OF THE LISTING, A COUNT OF ALL CHANGES WILL BE PRINTED.
 - (d) THIS JOB CAN BE RERUN, BUT USE CARE AS THE COST MATRIX FILE CAN BE DRASTICALLY CHANGED.
- (2) THE PROCEDURE FOR RECREATING THE (CMF) FROM (CMF) BACKUP TAPE. NO DATA CARDS ARE USED IN THIS PROGRAM--ONLY THE DATE-TITLE, OPTION CARD, AND THE (CMF) BACKUP FILE. TO OPERATE THE RECREATING PROGRAM:
- (a). PLACE THE DATE-TITLE CARD FOLLOWED BY THE OPTION CARD BEHIND THE "SYSIN DD *" CARD IN THE "WUASUCAD" JCL DECK.
 - (b) PUT THE DECK IN THE CARD READER AND READ THE DECK.
 - (c) OUTPUT WILL BE ON ONE-PART PAPER, LISTING ALL THE RECORDS THAT WERE RECREATED.
 - (d) THIS JOB CAN BE RERUN IF REQUIRED.

ROUTINE OPERATING SEQUENCE

The following routine operating sequence should normally be followed unless otherwise stated in the special operating instructions. In operating the cost analysis system, it is important to note that each program instruction must be understood and that certain procedures be completed prior to performing the subsequent operational functions.

- (A) Initial Edits - WUASPRER (Payroll), WUASRSER (Requisition from Stock), WUASPAER (Payment Authorization), WUASSCER (Service Contract), WUASMTER (Miscellaneous Transactions), and WUASBCDP (Depreciation).

Special Instructions:

- (1) The WUASBCDP (Depreciation) program needs to be run only once until the depreciation values change.
- (2) All other edit programs must be run even if current data is not supplied.
- (3) The edit programs may be run in any order.

- (B) Cost Allocation - WUASCAP1 (1st Allocation Program), WUASCAP2 (2nd Allocation Program), and WUASCAP3 (3rd Allocation Program).

- (C) Updating Cost Matrix File - WUASUZFI (Zero File) and WUASCUPD (Update File).

Special Instructions:

- (1) Each program must be run in the order listed.
- (2) Do not re-run these programs without consulting with Systems Programmer.

- (D) Print Output Reports - WUASLV41, WUASLV42, WUASLV43, WUASLV44 (Level 4 Reports); WUASPLV3 (Level 3 Report); WUASPLV2 (Level 2 Report); WUASPLV1 (Level 1 Report); and WUASPWAR (Water Supply Data Analysis Report).

Special Instructions:

- (1) Programs may be run in any order, except WUASPLV1; because this program needs data from WUASPLV2.

- (2) Programs may be run as many times as required.

- (E) Back-Up Cost Matrix File - WUASBUCL (Back-Up file).

Special Instructions:

- (1) Program may be run as many times as required.
- (2) To produce a current listing of the Cost Matrix File only, run program WUASBUCM without "Back-Up" option card.
- (3) To back-up the Cost Matrix File and then produce a listing run program WUASBUCM, with "Back-Up" option card.

SECTION 3

DATA FORMAT

KEYPUNCHING SOURCE DATA

The keypunch format is provided to describe the card image layout of all input data. This keypunch format should be used by the keypunch operator in setting-up drum cards and keypunching the data onto an 80-column card. Any variation in these formats should be discussed with the Systems Programmer.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASBCCT

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT BUILD COST CENTER TABLE

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASBSAT (" .CCC ", ".CCT", ".FAT")

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT BUILDING SPECIAL ACCT. TABLES

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASBFAT

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT FINANCIAL ACCT TABLE INPUT SHEET

TYPE OF CARDS 80 COLUMN

SOURCE CARD DEPT. JOB NO.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASBCRF

SHEET NO. 1 OF 2

PROGRAM CARD NO. 1

SOURCE DOCUMENT CROSS REFERENCE FILE INPUT SHEET

TYPE OF CARDS 80 COLUMN

SOURCE FIRST CARD DEPT. JOB NO.

L7

FIELD NO.	FIELD NAME	COLMNS FROM	THRU	NO. OF COLUMNS	TYPE N	A	JUS-	VER-	REMARKS AND INSTRUCTIONS
1	CARD NUMBER	1	1	1	X				MUST BE A '1'
	BLANK	2	2	1					
2	ACCOUNT NUMBER	3	12	10	X	X	LJ		
3	ALLOCATION CODE	13	13	1		X			
	BLANK	14	14	1					
4	COST CAT (NON LABOR)	15	16	2	X				
5	COST CAT (LABOR)	17	18	2	X				
	BLANK	19	20	2					
6	COST CENTER	21	24	4	X				
7	MATCH CODE	25	28	4	X				
8	COST CENTER	29	32	4	X				
9	MATCH CODE	33	36	4	X				
10	COST CENTER	37	40	4	X				
11	MATCH CODE	41	46	4	X				
12	COST CENTER	45	48	4	X				
13	MATCH CODE	49	52	4	X				
14	COST CENTER	53	56	4	X				
15	MATCH CODE	57	60	4	X				
16	COST CENTER	61	64	4	X				
17	MATCH CODE	65	68	4	X				
18	COST CENTER	69	72	4	X				
19	MATCH CODE	73	76	4	X				
	BLANK	77	80	4					

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASBCRF

SHEET NO. 2 OF 2

PROGRAM CARD NO. 2

SOURCE DOCUMENT CROSS REFERENCE FILE INPUT SHEET

TYPE OF CARDS 80 COLUMN

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASBDPT

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT BUILD DEPRECIATION TABLE

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASSCER

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT SERVICE CONTRACT

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASMASTER

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT MISCELLANEOUS TRANSACTIONS

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASPAER

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT PAYMENT AUTHORIZATIONS

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASRSE

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT REQUISITION FROM STOCK

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASPRER

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT PAYROLL

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASCAP1

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT UTILITY DATA CARD FOR COST ALCTN. PT 1

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO. _____

OPTION CARD

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASUZET

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASPLV1

SHEET NO. 1 OF 1

PROGRAM CARD NO. 1

SOURCE DOCUMENT WATER FLOW SHEET

TYPE OF CARDS 80 COLUMN

SOURCE CARD DEPT. JOB NO. _____

* TAKEN FROM LV II ACTUAL TO BUDGET PERCENTAGE

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASPWAR

SHEET NO. 1 OF 1

PROGRAM CARD NO. 1

SOURCE DOCUMENT WATER FLOW SHEET

TYPE OF CARDS 80 COLUMN

SOURCE CARD DEPT. JOB NO.

OPTION CARD

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASBUCM

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT BACK UP OPTION CARD

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

OPTION CARD

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASUCAD

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT CHANGE, ADD, DELETE OPTION CARD

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME WUASUCAD

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT UTILITY. TO CHANGE, ADD, OR DLTE. CM RCDs.

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME DATE-TITLE CARD

SHEET NO. 1 OF 1

PROGRAM CARD NO. _____

SOURCE DOCUMENT STANDARD DATE-TITLE CARD

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO.

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME _____ **DATE-TITLE CARD**

SHEET NO. 1 OF 1

PROGRAM CARD NO. SPECIAL

SOURCE DOCUMENT FOR PROGRAM WUASBDPT

TYPE OF CARDS _____

SOURCE DEPT. JOB NO. - - -

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME DATE-TITLE

SHEET NO. 1 OF 1

PROGRAM CARD NO. SPECIAL

SOURCE DOCUMENT FOR PROGRAM WIASBCDP

TYPE OF CARDS 80 COLUMN

SOURCE _____ DEPT. _____ JOB NO. _____

PUNCHING OR VERIFICATION INSTRUCTIONS

NAME _____ **DATE-TITLE**

SHEET NO. 1 OF 1

PROGRAM CARD NO. SPECIAL

SOURCE DOCUMENT FOR PROGRAMS WUASPLV2, WUASPCAR

TYPE OF CARDS 80 COLUMN

SOURCE DEPT. JOB NO.

SYSTEM FILE LAYOUT

The system file layout is provided to illustrate each cost accounting system program file.

FILE LAYOUT FORMAT

NAME WUASBCCT
COST CENTER TABLE FILE

SHEET NO. 1 OF 1

FILE RECORD NO. 1

RECORD LENGTH 30

TYPE OF FILE INDEXED SEQUENTIAL

SOURCE COST CENTER DESCRIPTION SHEET

FILE LAYOUT FORMAT

NAME WUASBSAT. "XXX" SPECIAL ACCT. TABLES (".CCC", ".CCT", ".FAT", ".DPT") SHEET NO. 1 OF 2

FILE RECORD NO. 1 RECORD LENGTH 22

TYPE OF FILE SEQUENTIAL SOURCE SPECIAL ACCOUNTS INPUT DESCRIPTION SHEET

FILE LAYOUT FORMAT

NAME WUASBSAT
 SPECIAL ACCT. TABLE

SHEET NO. 2 OF 2

FILE RECORD NO. 2

RECORD LENGTH 22

TYPE OF FILE SEQUENTIAL

SOURCE _____

FILE LAYOUT FORMAT

WUASBFAT
NAME FINANCIAL ACCOUNT TABLE FILE

SHEET NO. 1 OF 1

FILE RECORD NO. 1

RECORD LENGTH 58

TYPE OF FILE INDEXED SEQUENTIAL

SOURCE FINANCIAL ACCOUNT INPUT DESCRIPTION SHEET

WUASBCRF

FILE LAYOUT FORMAT

NAME _____ CROSS REFERENCE FILE (CRF)

SHEET NO. 1 OF 1

FILE RECORD NO. 1

RECORD LENGTH 265

TYPE OF FILE INDEXED SEQUENTIAL

SOURCE CROSS REFERENCE INPUT DESCRIPTION SHEET

* THE FOLLOWING TWO ITEMS OCCURS 31 TIMES

FILE LAYOUT FORMAT

NAME WUASBCMF
COST MATRIX FILE (CMF)

SHEET NO. 1 OF 1

FILE RECORD NO.

RECORD LENGTH 80

TYPE OF FILE INDEXED SEQUENTIAL

SOURCE CROSS REFERENCE FILE

FILE LAYOUT FORMAT

NAME WUASBUCK
COST MATRIX FILE BACKUP (CMF BACKUP)

SHEET NO. 1 OF 1

FILE RECORD NO. _____

RECORD LENGTH 80

TYPE OF FILE SEQUENTIAL (TAPE)

SOURCE COST MATRIX FILE

FILE LAYOUT FORMAT

NAME EDIT FILES *

SHEET NO. 1 OF 3

FILE RECORD NO. 1

RECORD LENGTH 26

TYPE OF FILE SEQUENTIAL

SOURCE PAYROLL, PAY AUTH. REQ STOCK, SERVICE CONTRACT,
DEPRECIATION, MISC. TRANS.

* NOTE: EACH EDIT FILE HAS THE SAME FORMAT.

FILE LAYOUT FORMAT

NAME EDIT FILES

SHEET NO. 2 OF 3

FILE RECORD NO. 2

RECORD LENGTH 26

TYPE OF FILE SEQUENTIAL

SOURCE

FILE LAYOUT FORMAT

NAME EDIT FILES

SHEET NO. 3 OF 3

FILE RECORD NO. (DATA) 3

RECORD LENGTH 26

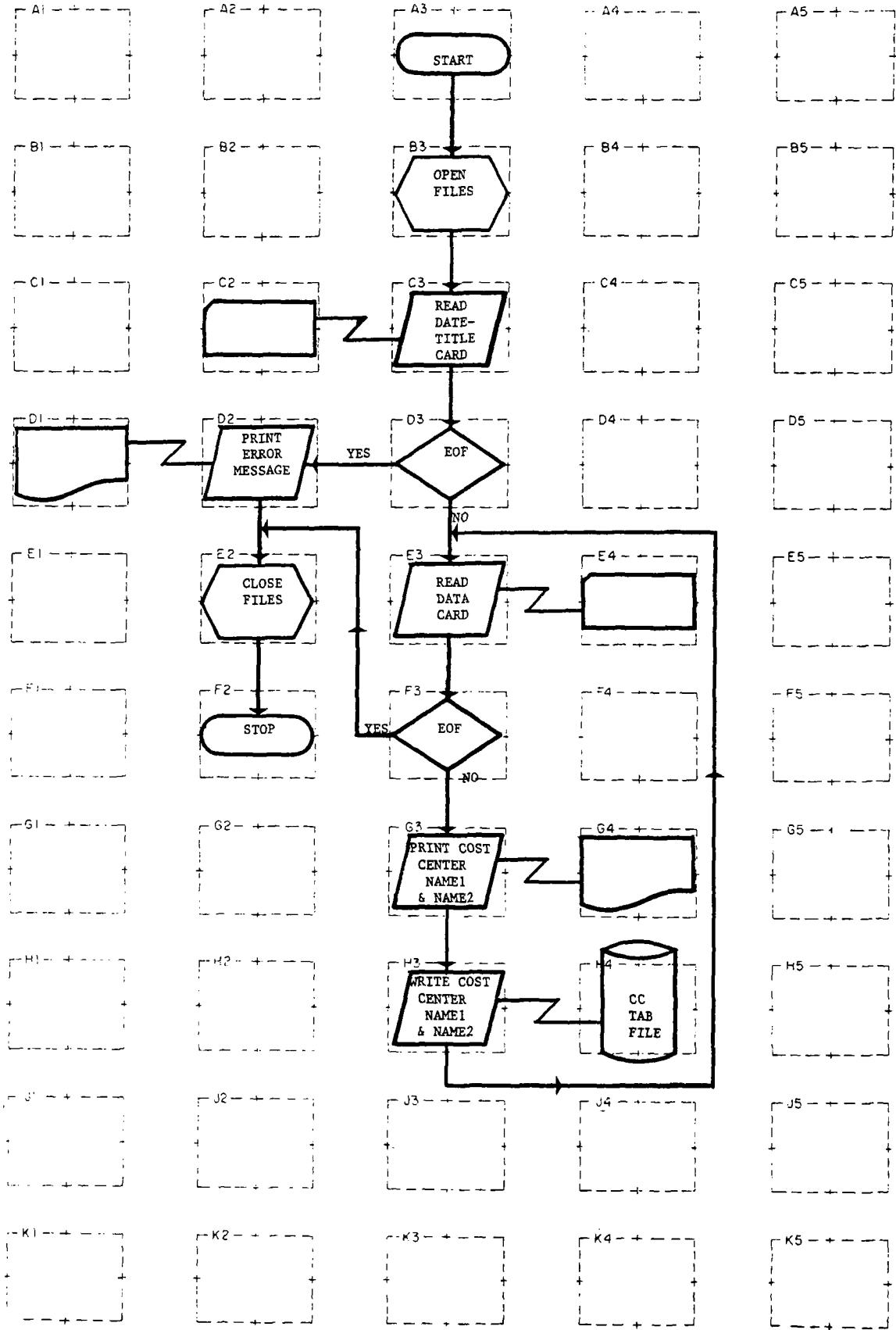
TYPE OF FILE SEQUENTIAL

SOURCE 26

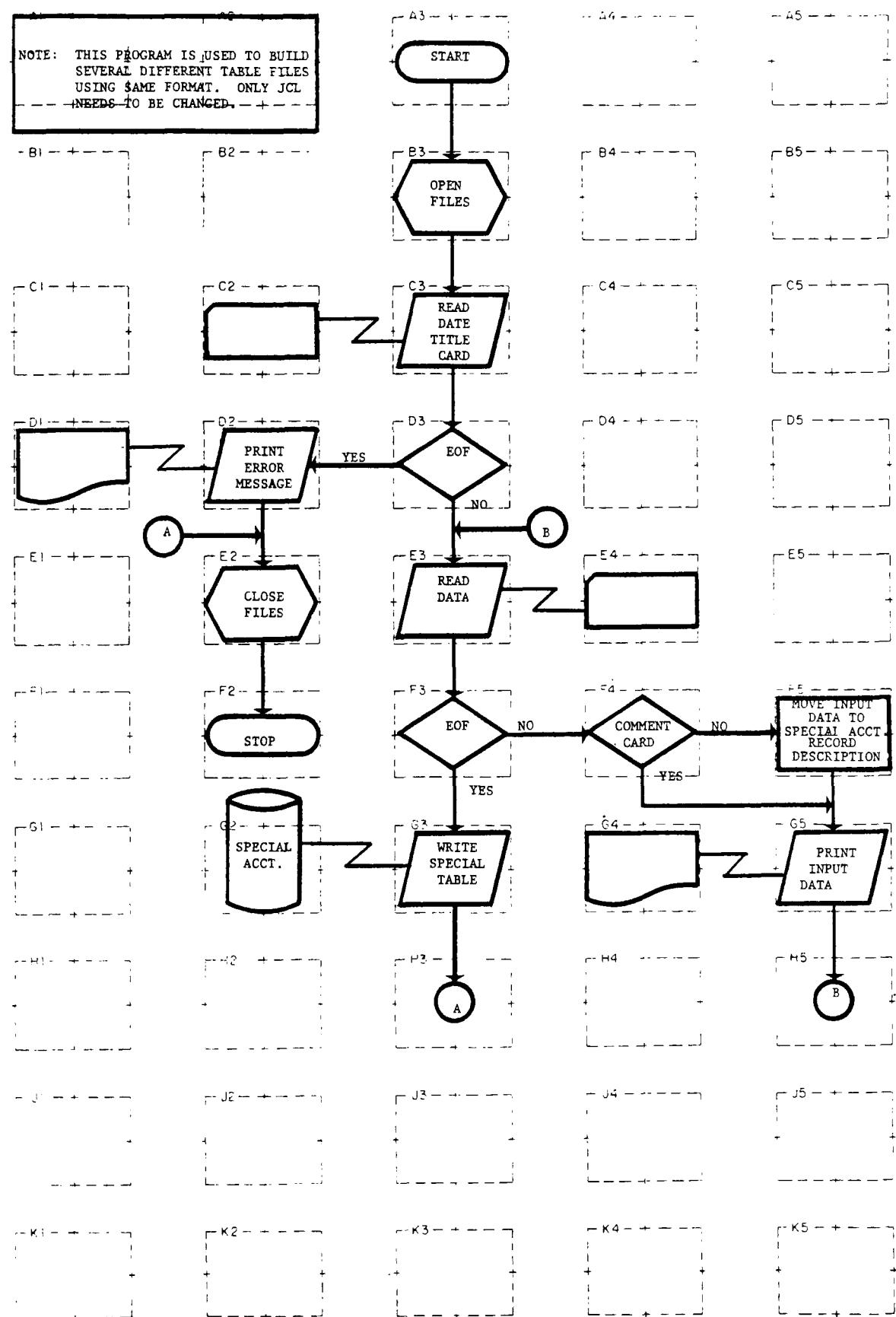
SECTION 4

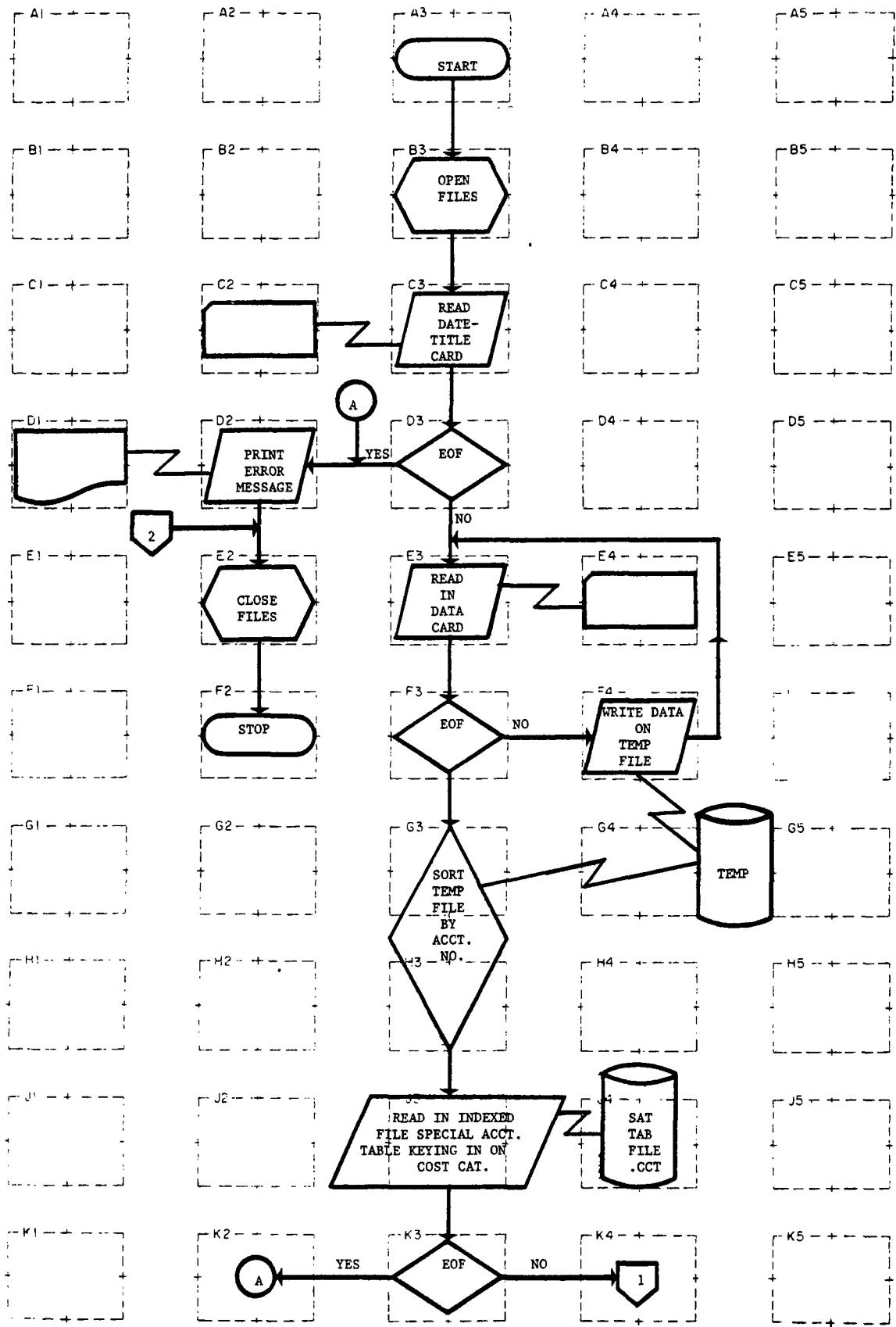
PROGRAM FLOW DIAGRAMS

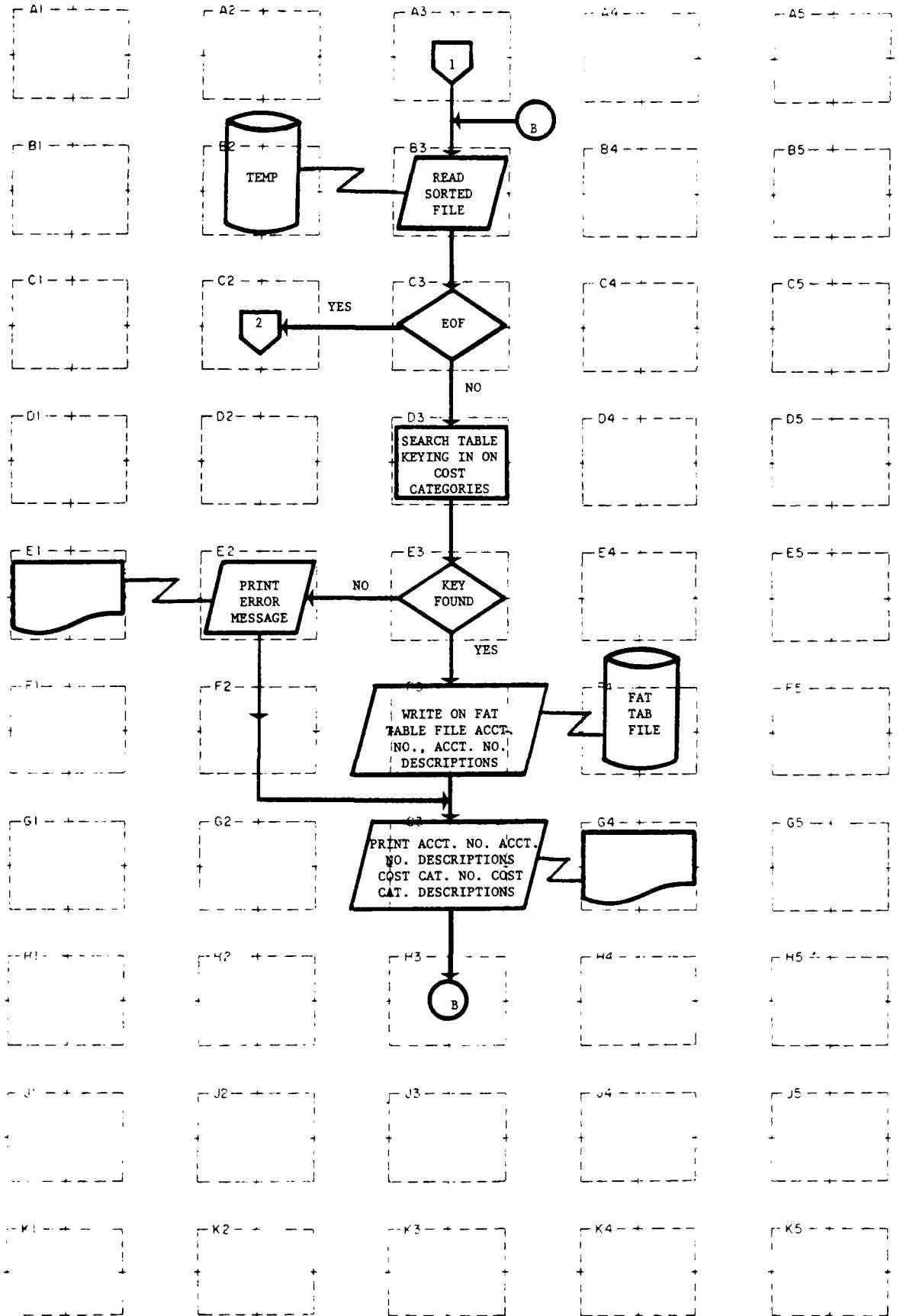
The following program flow diagrams are provided to illustrate the normal activity sequence within each program.

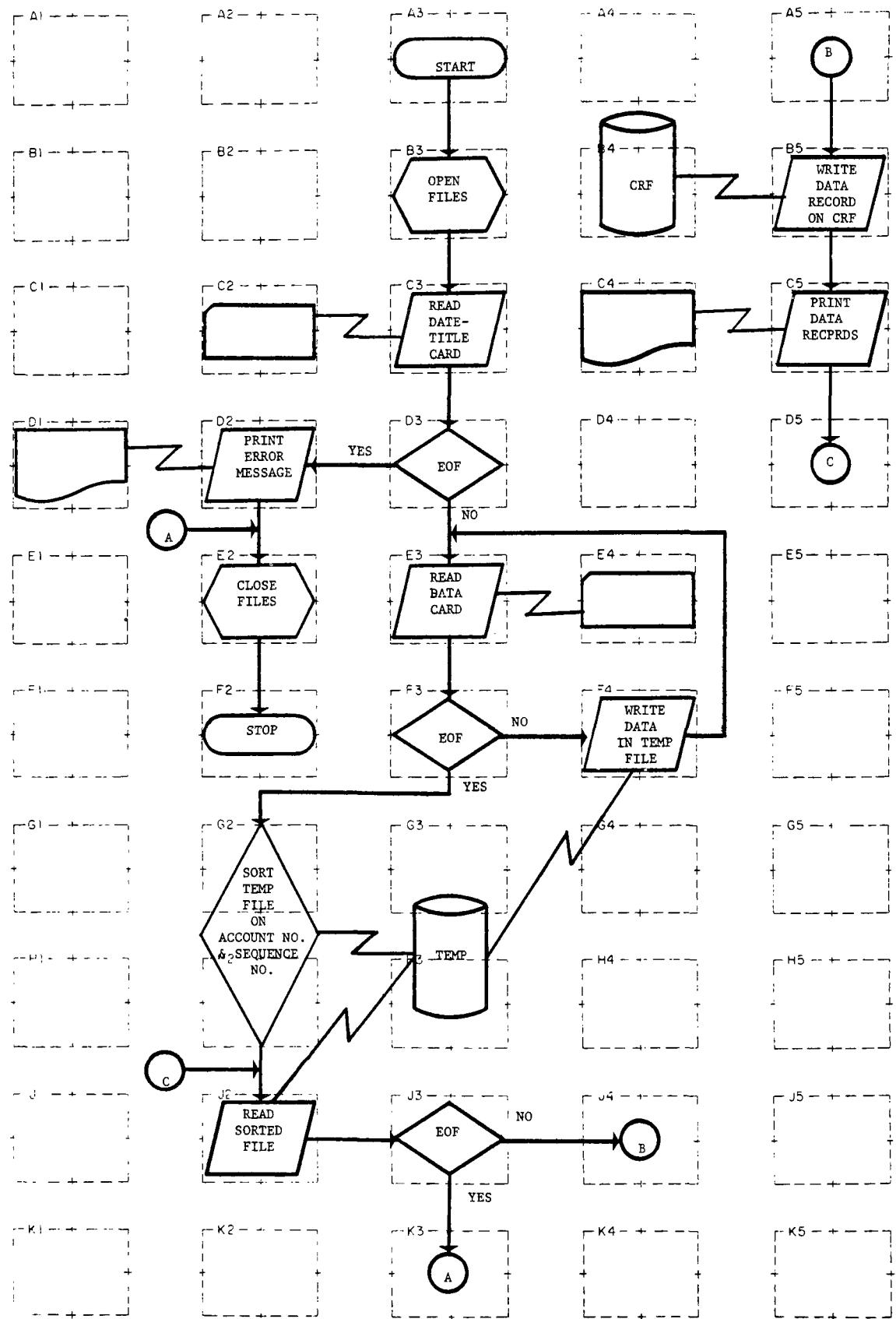


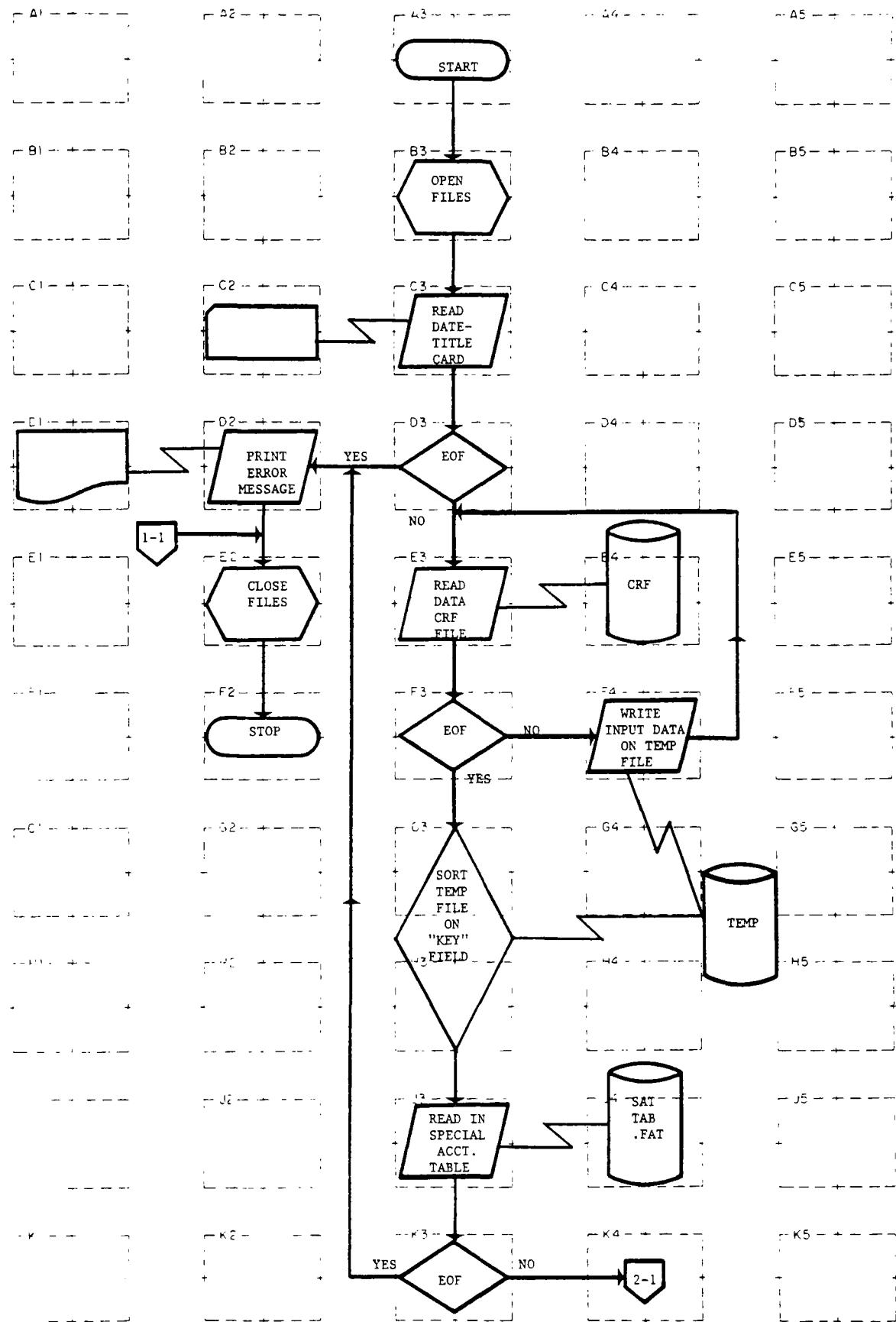
NOTE: THIS PROGRAM IS USED TO BUILD
SEVERAL DIFFERENT TABLE FILES
USING SAME FORMAT. ONLY JCL
NEEDS TO BE CHANGED.

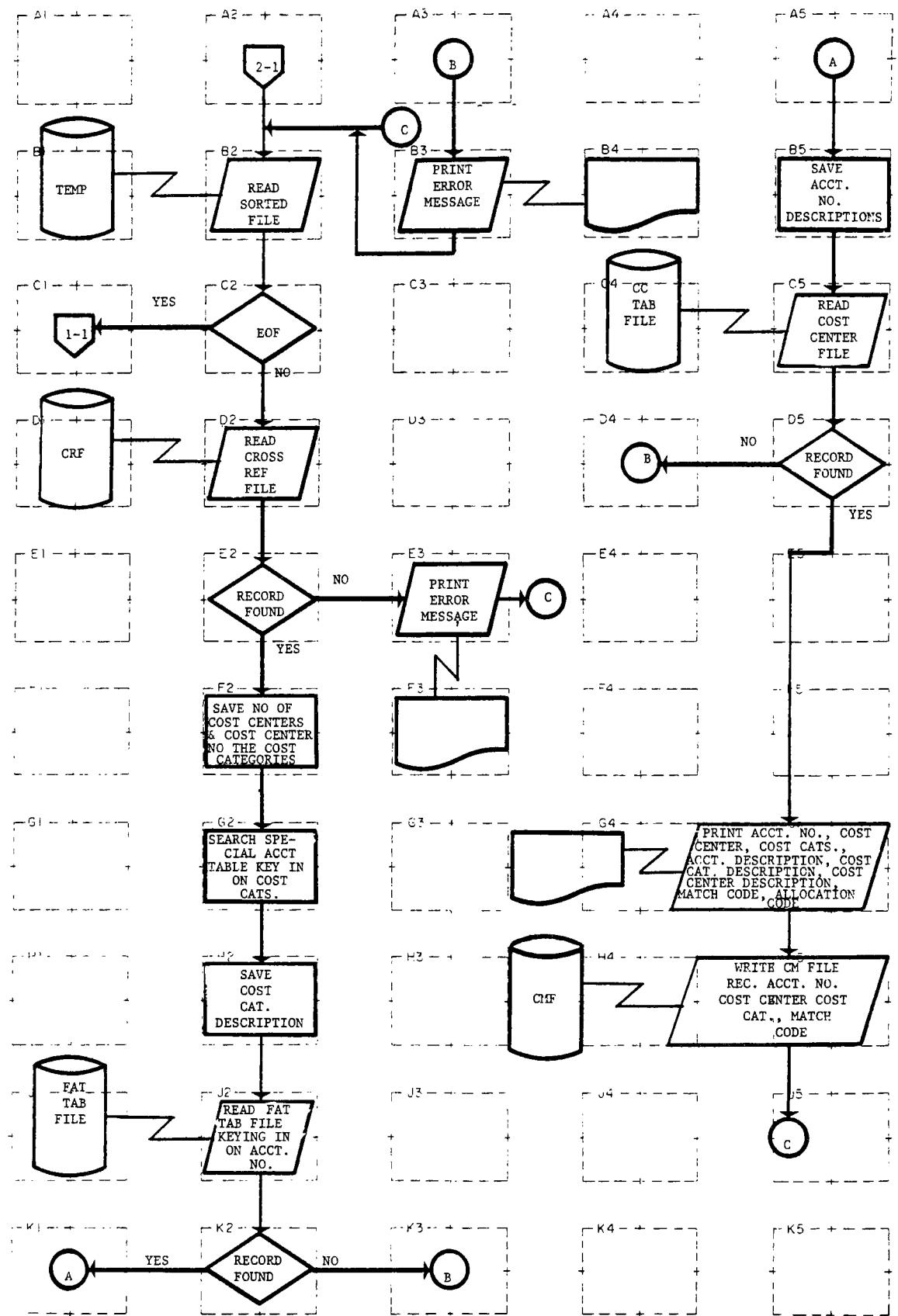


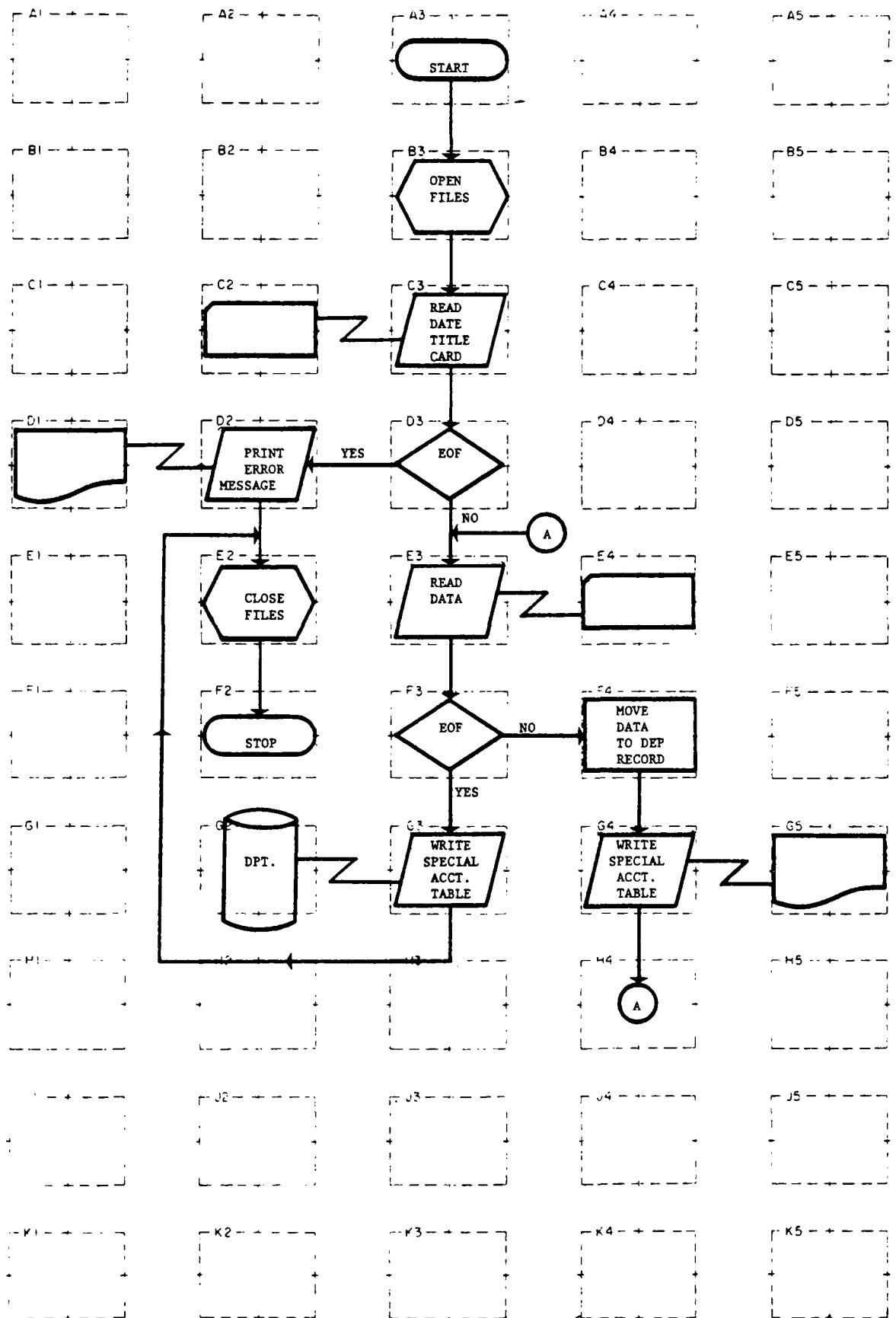


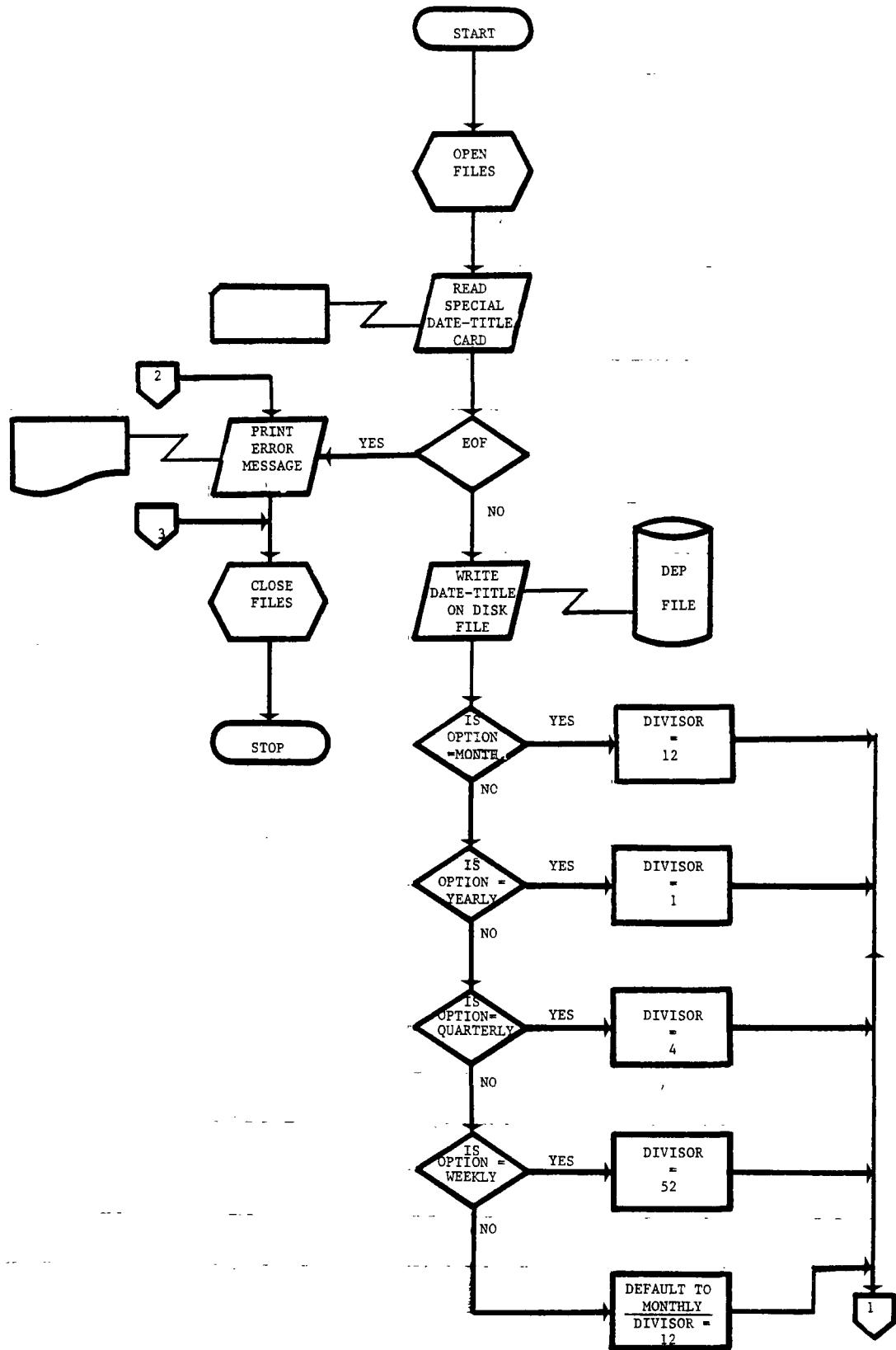


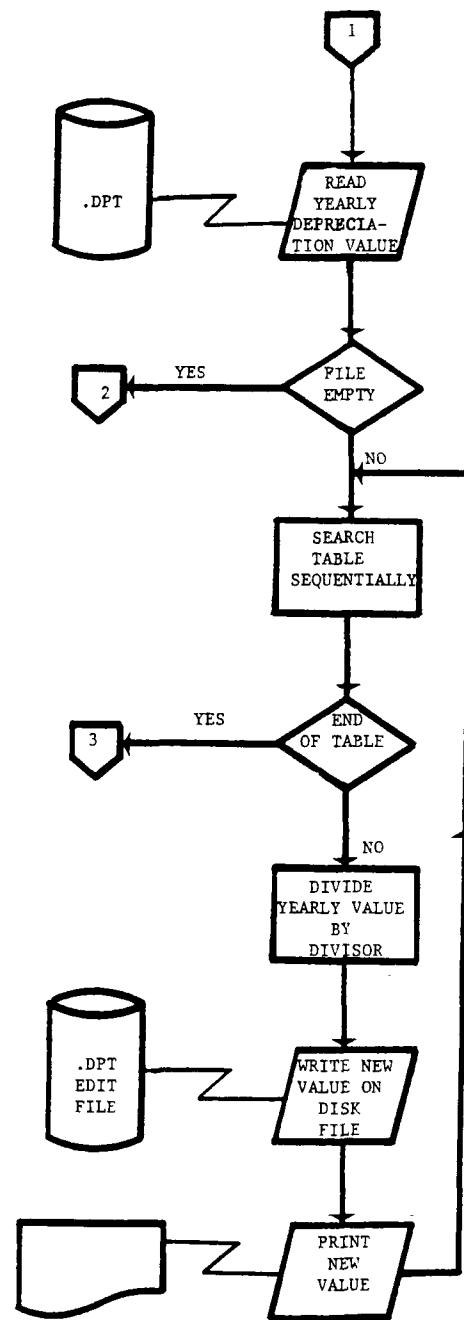


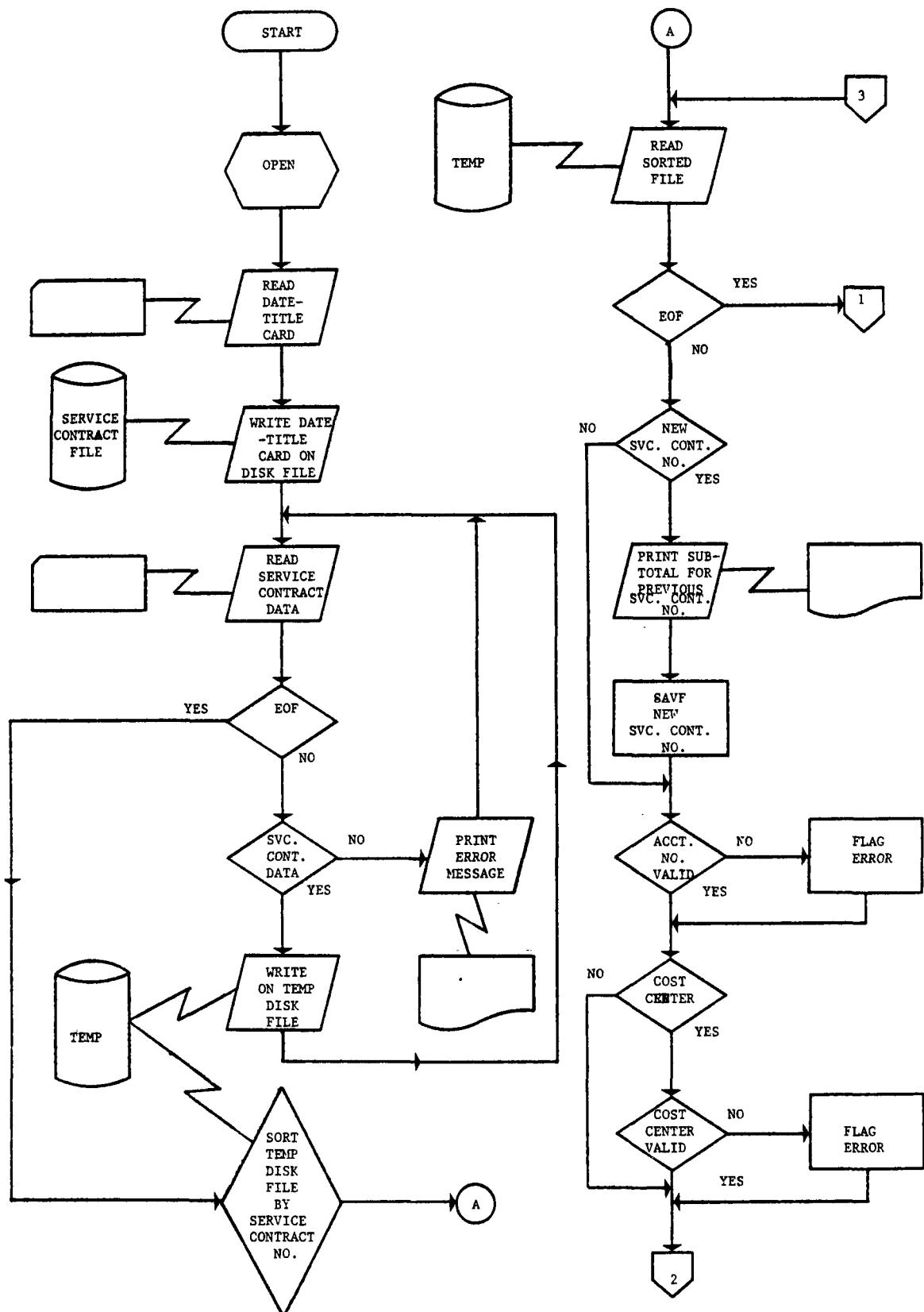


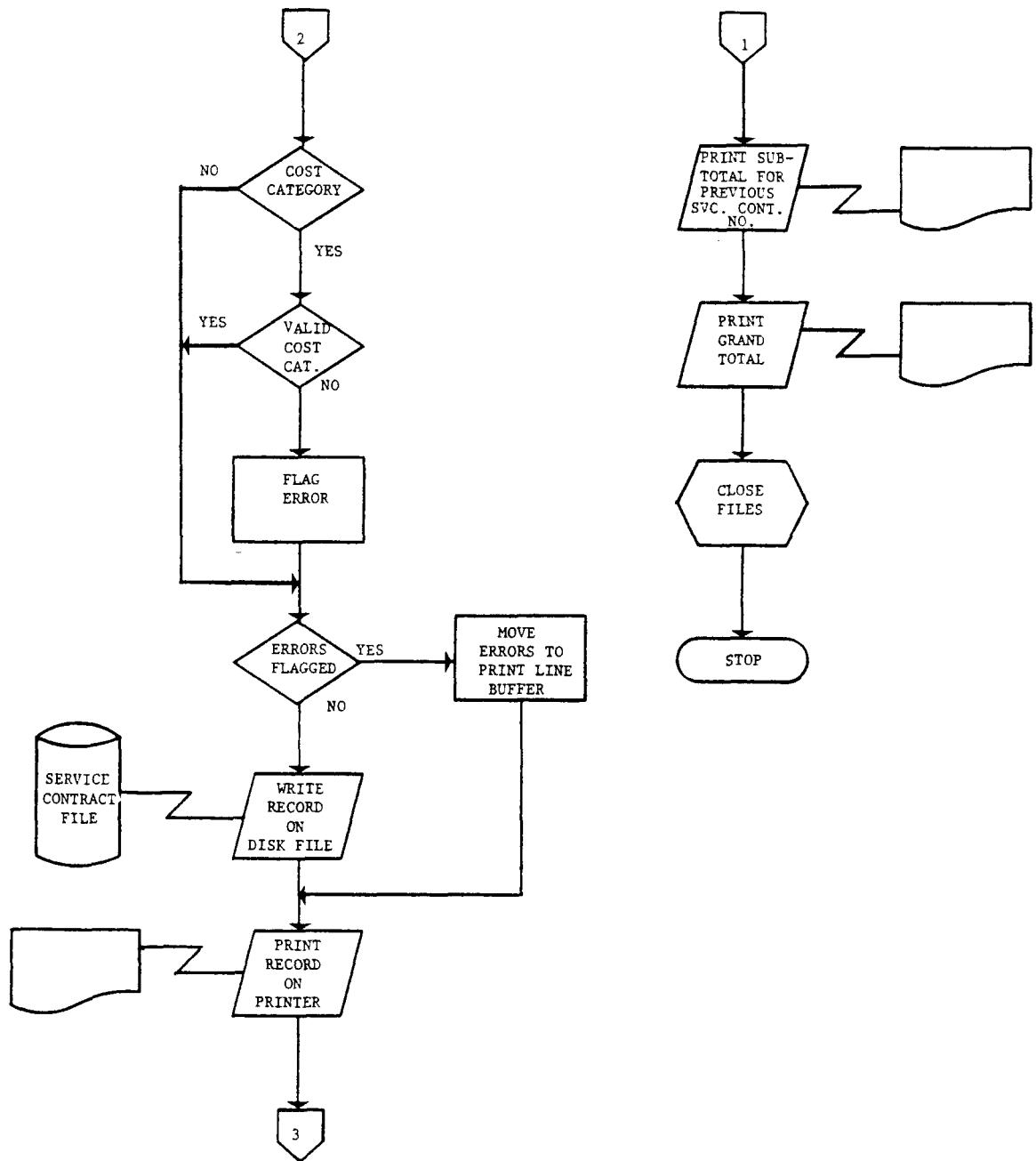


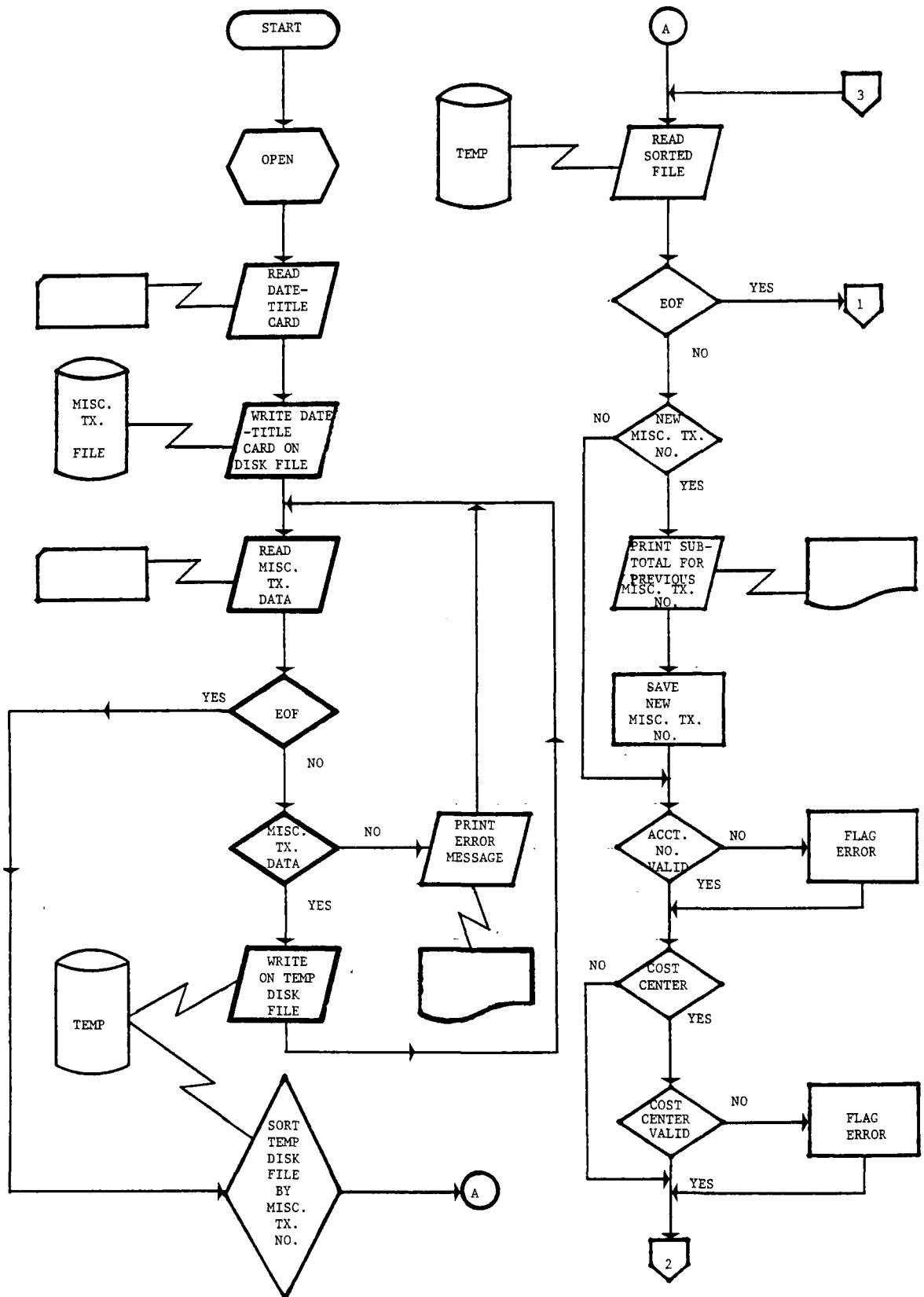


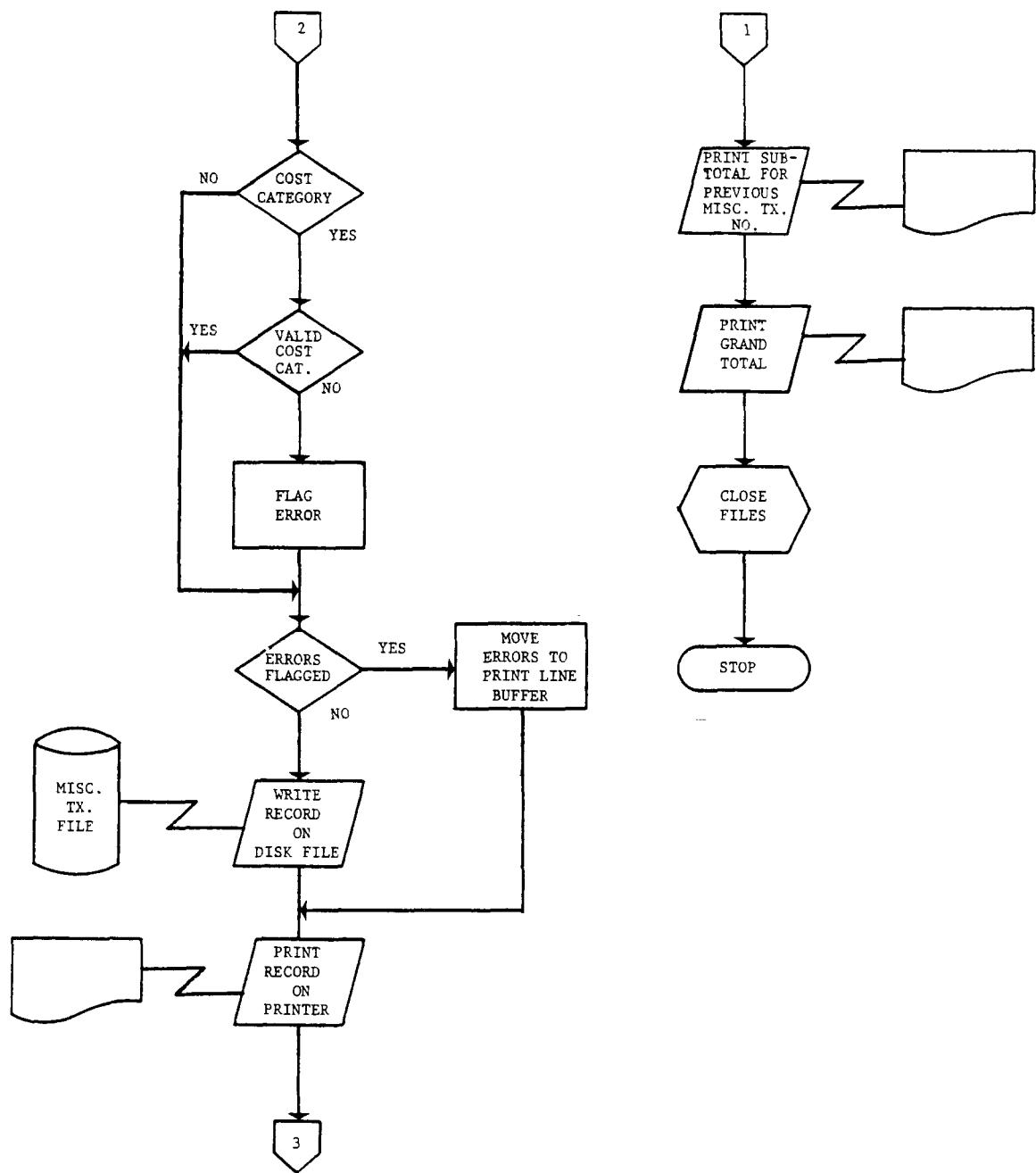


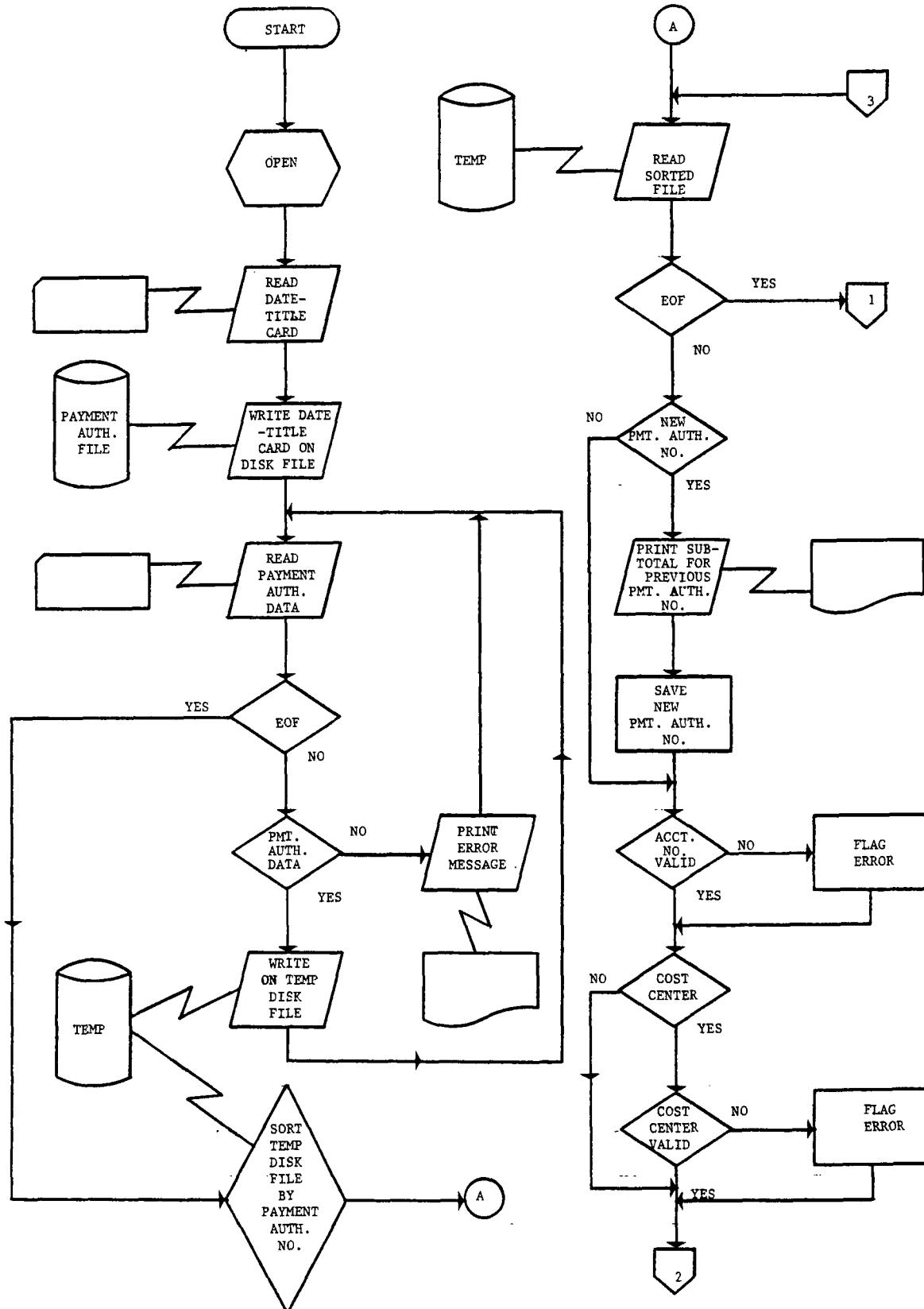


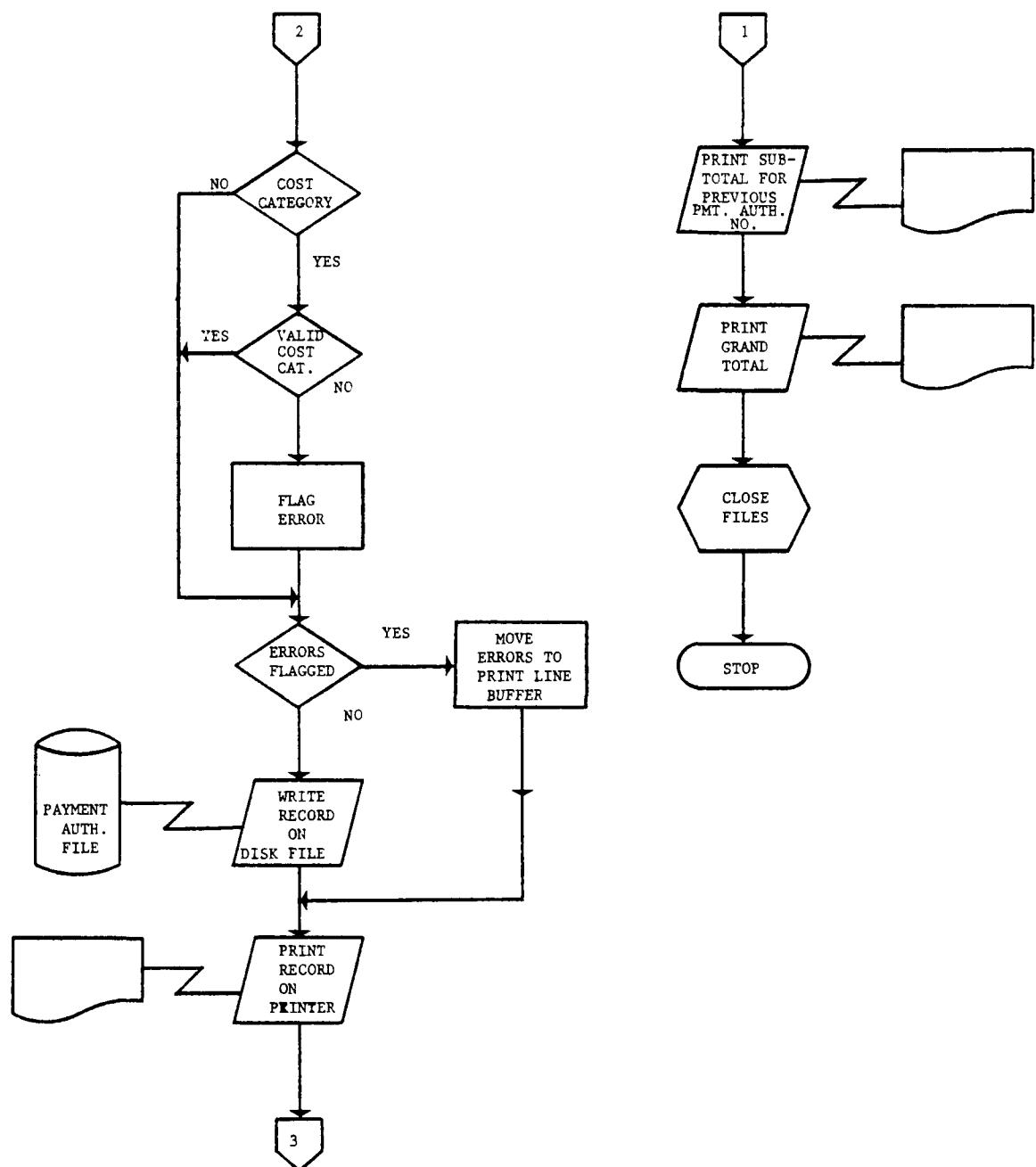


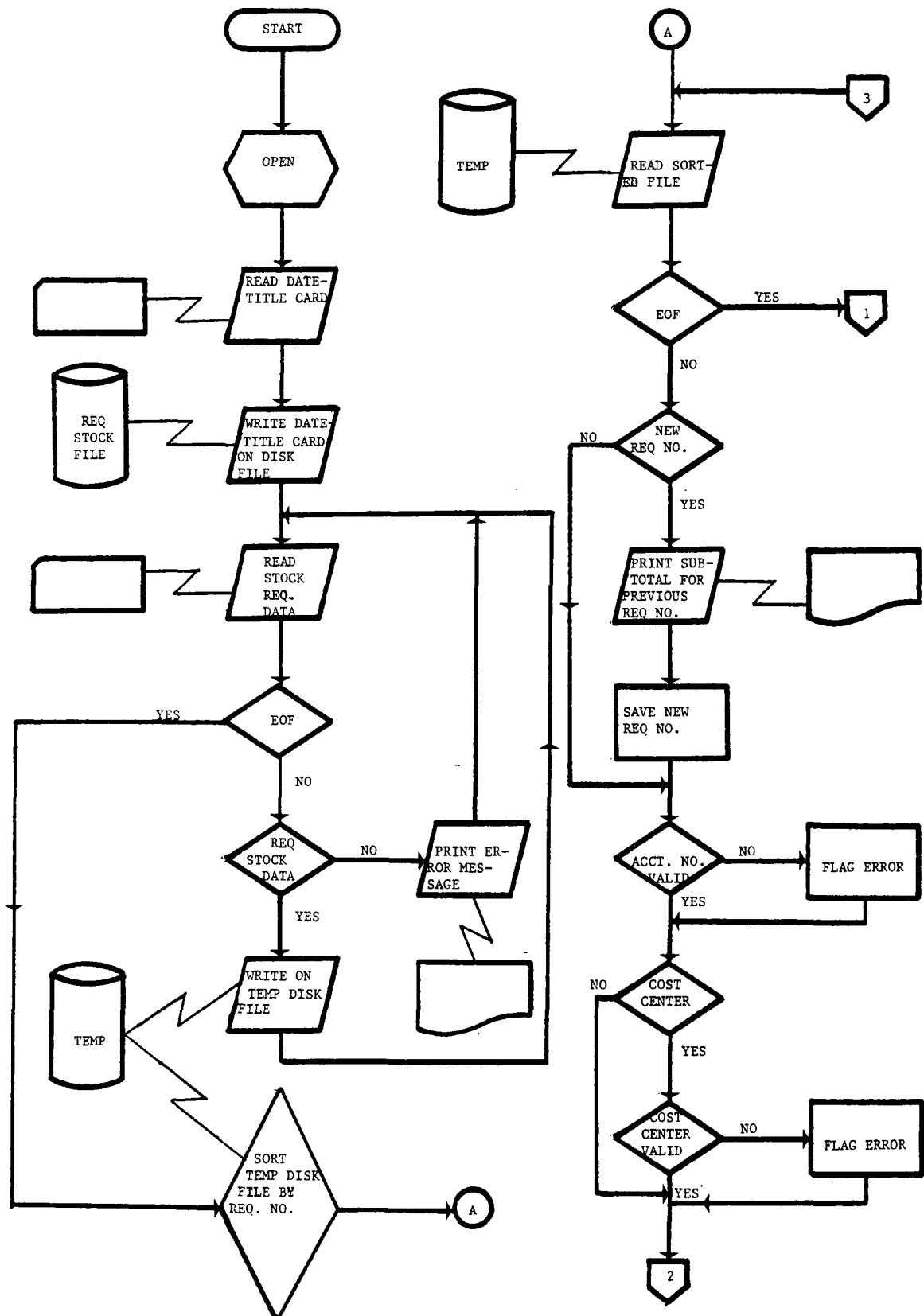


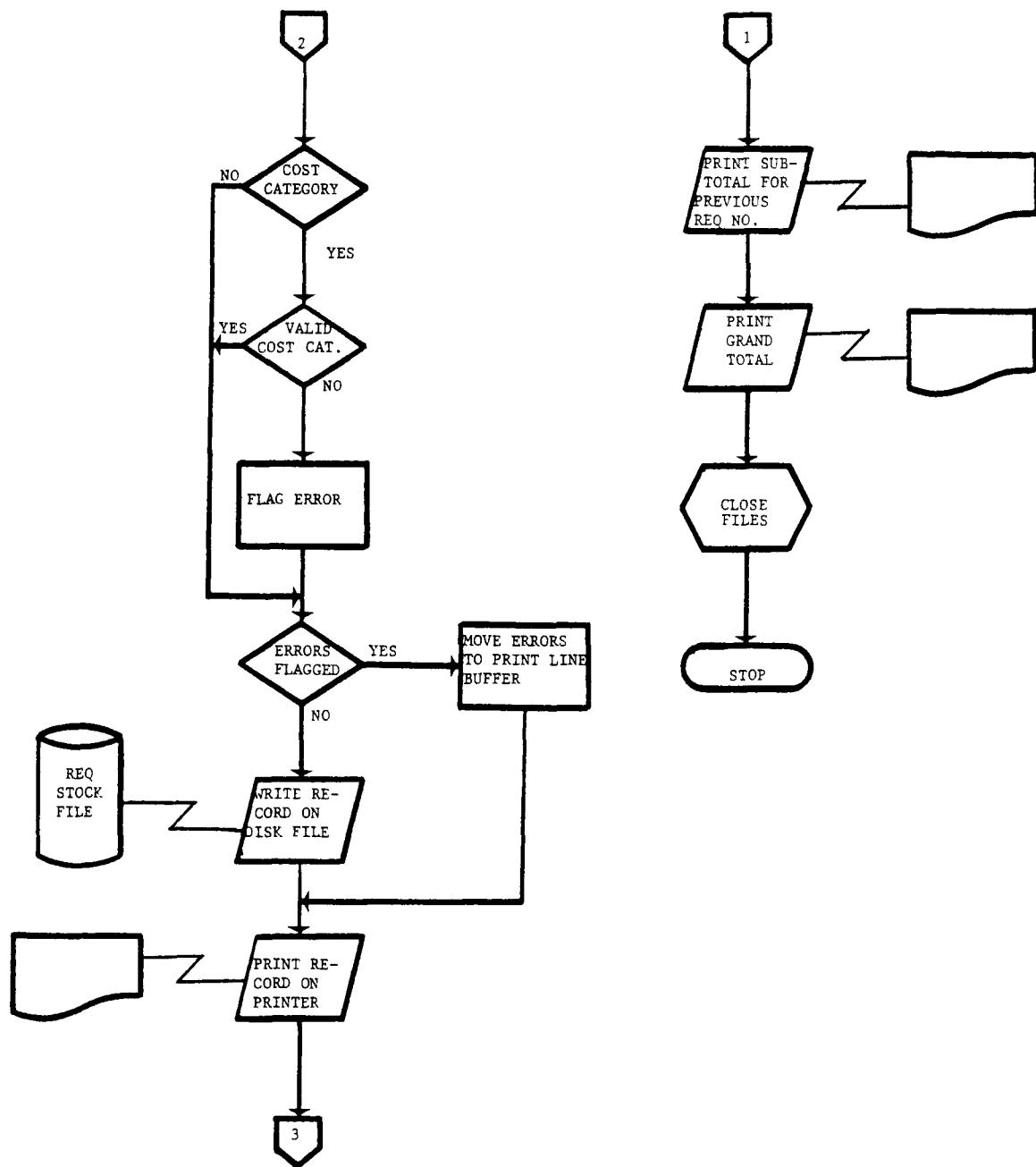


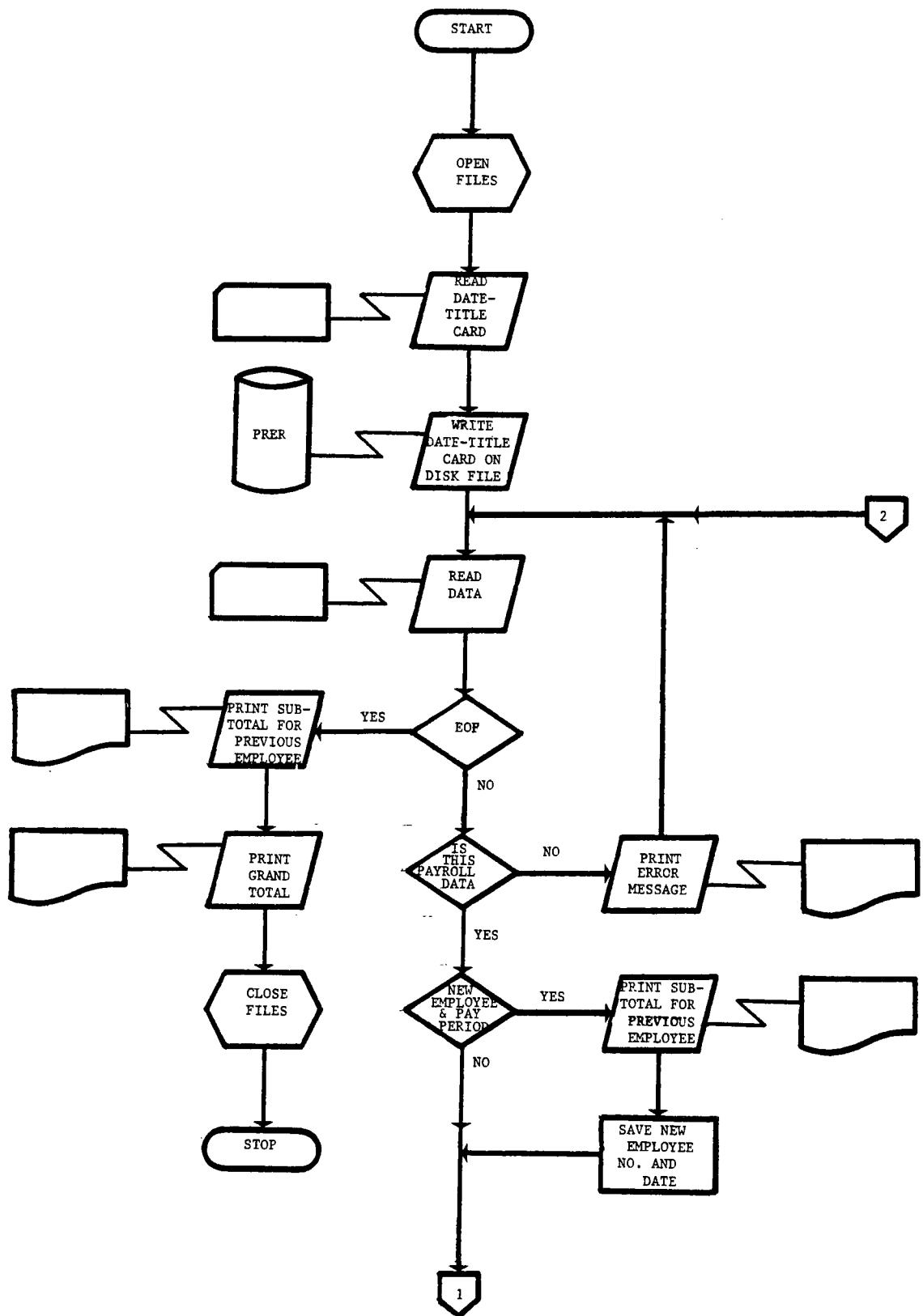


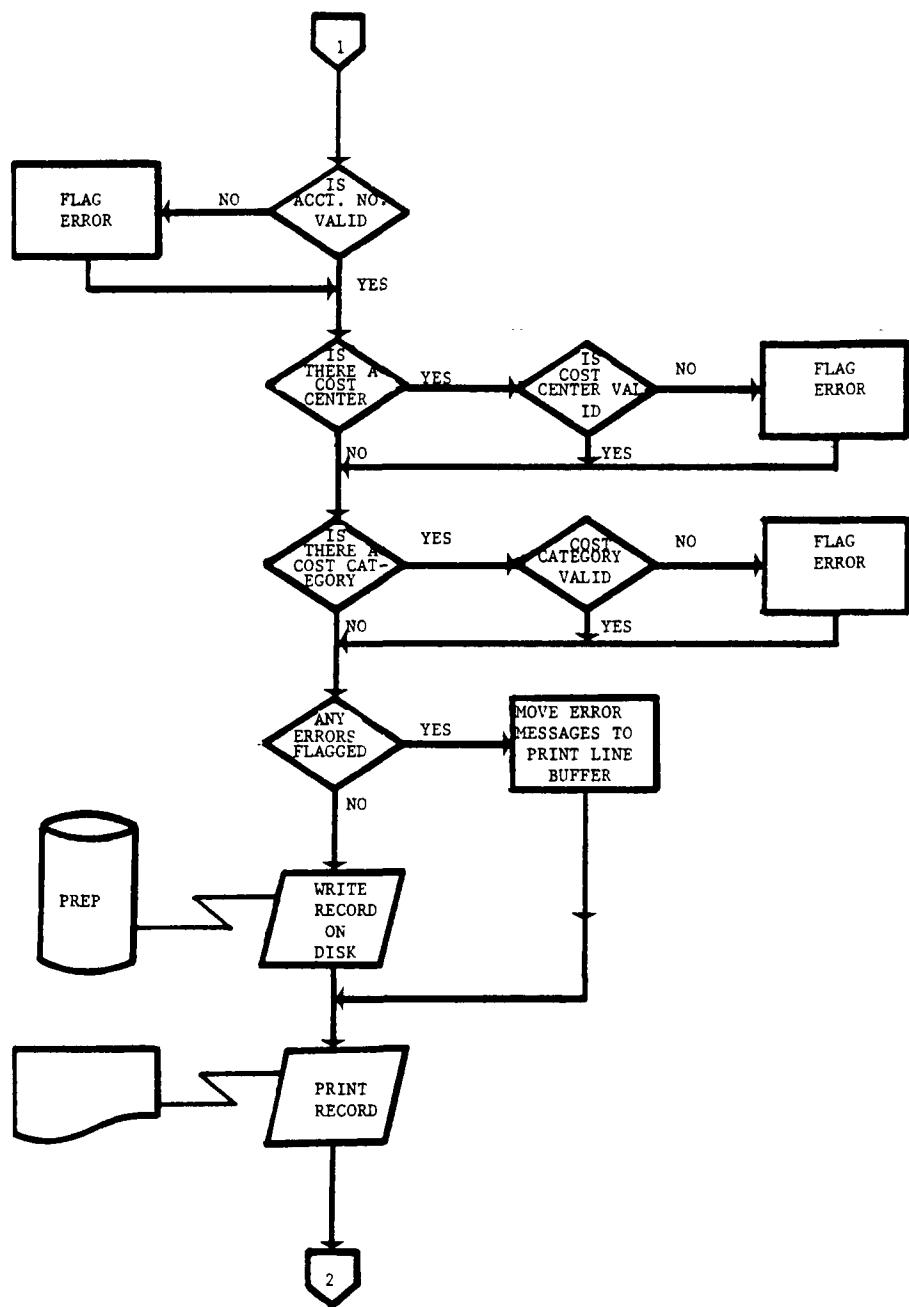


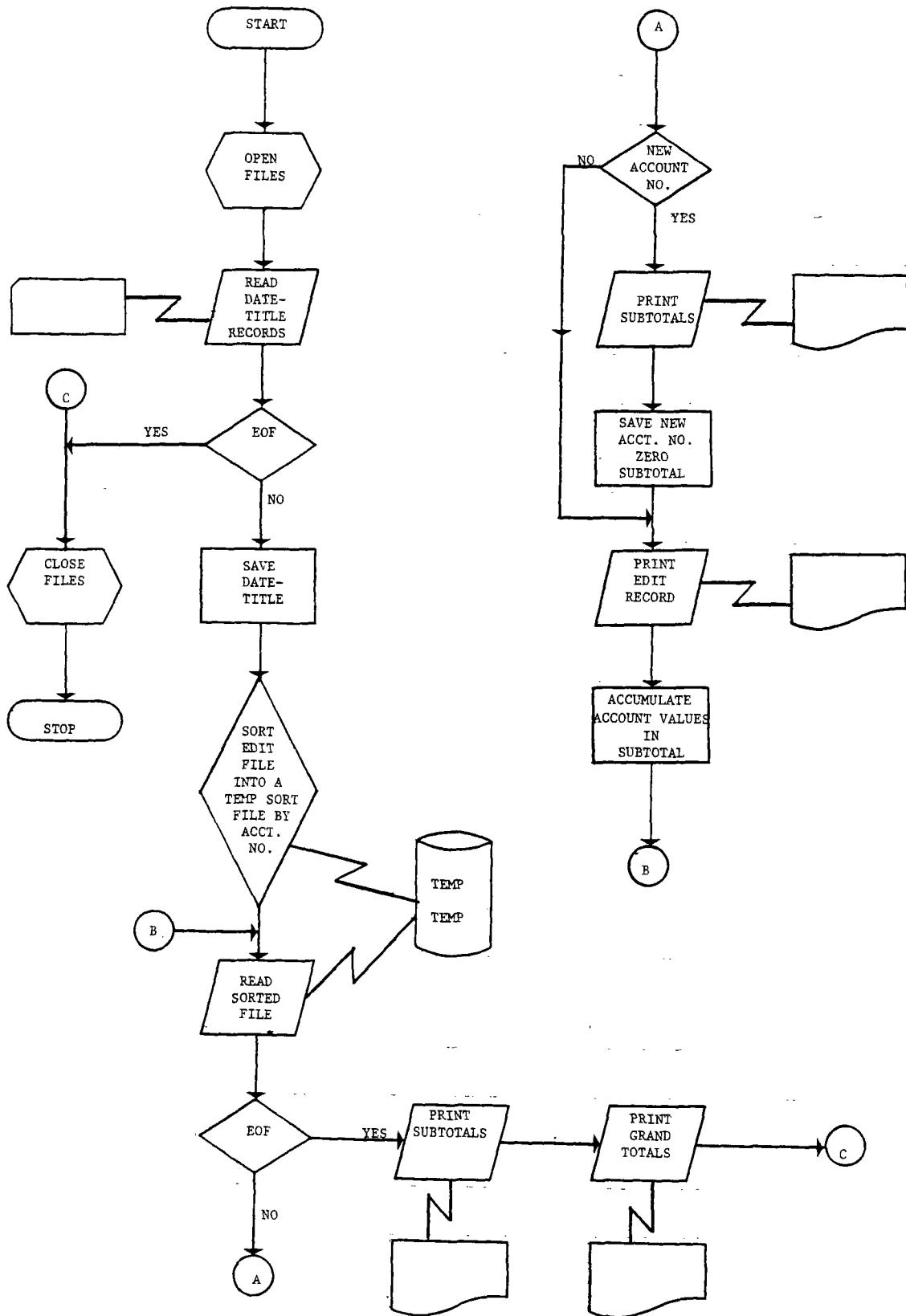


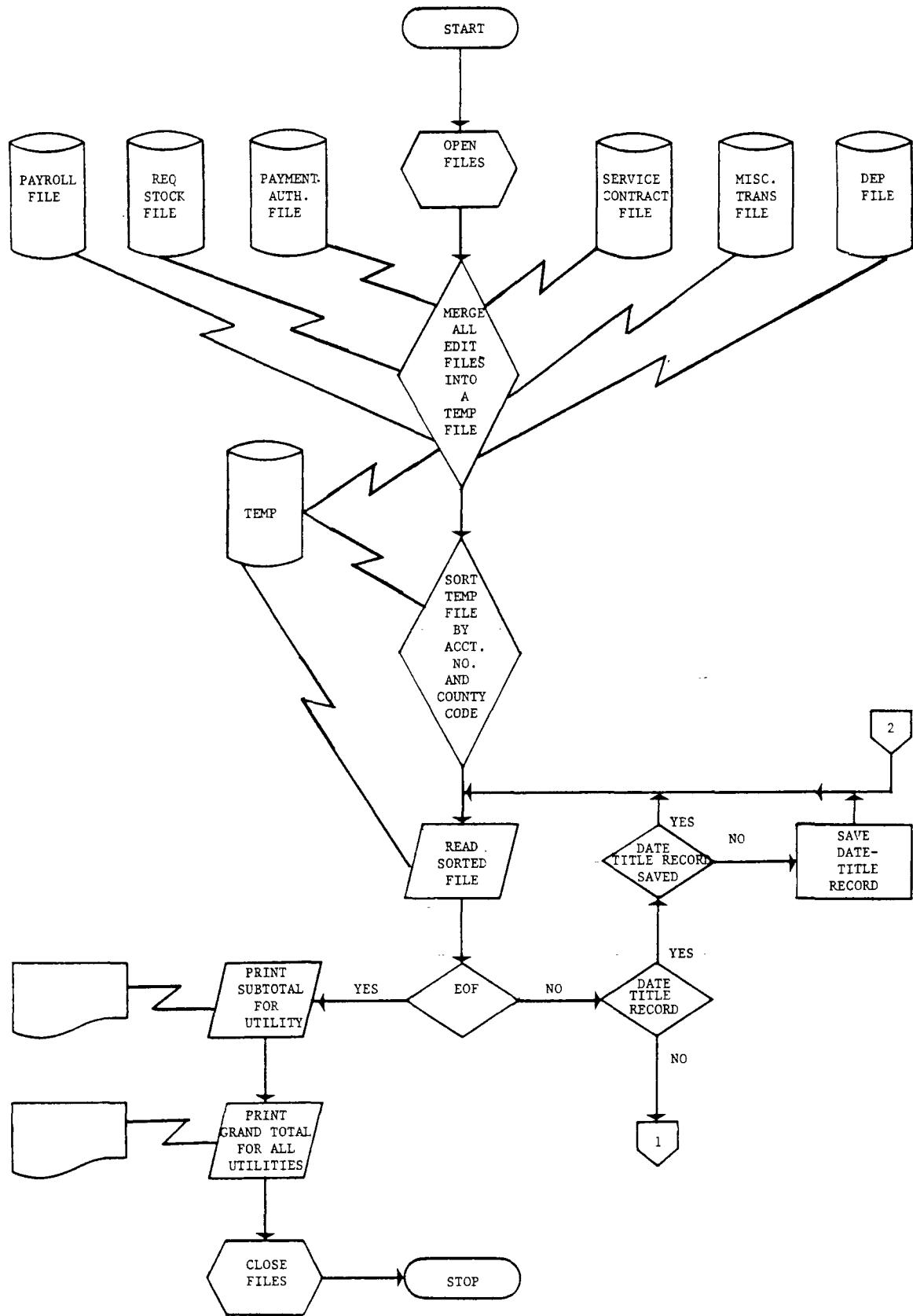


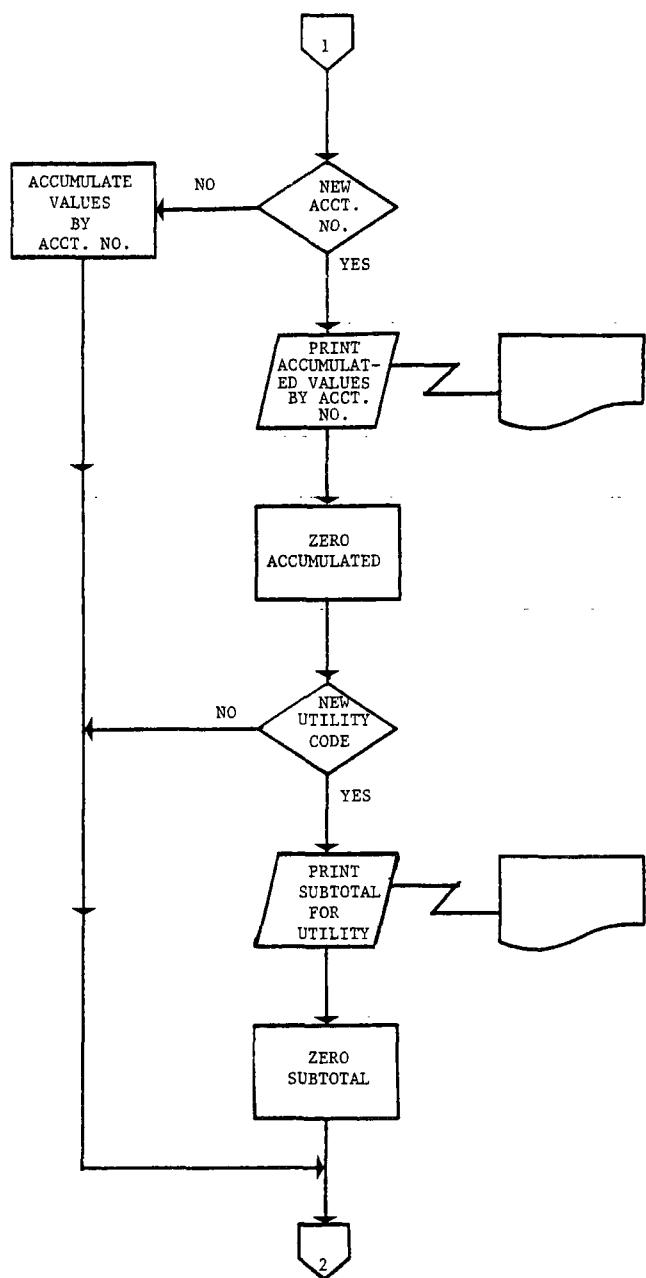


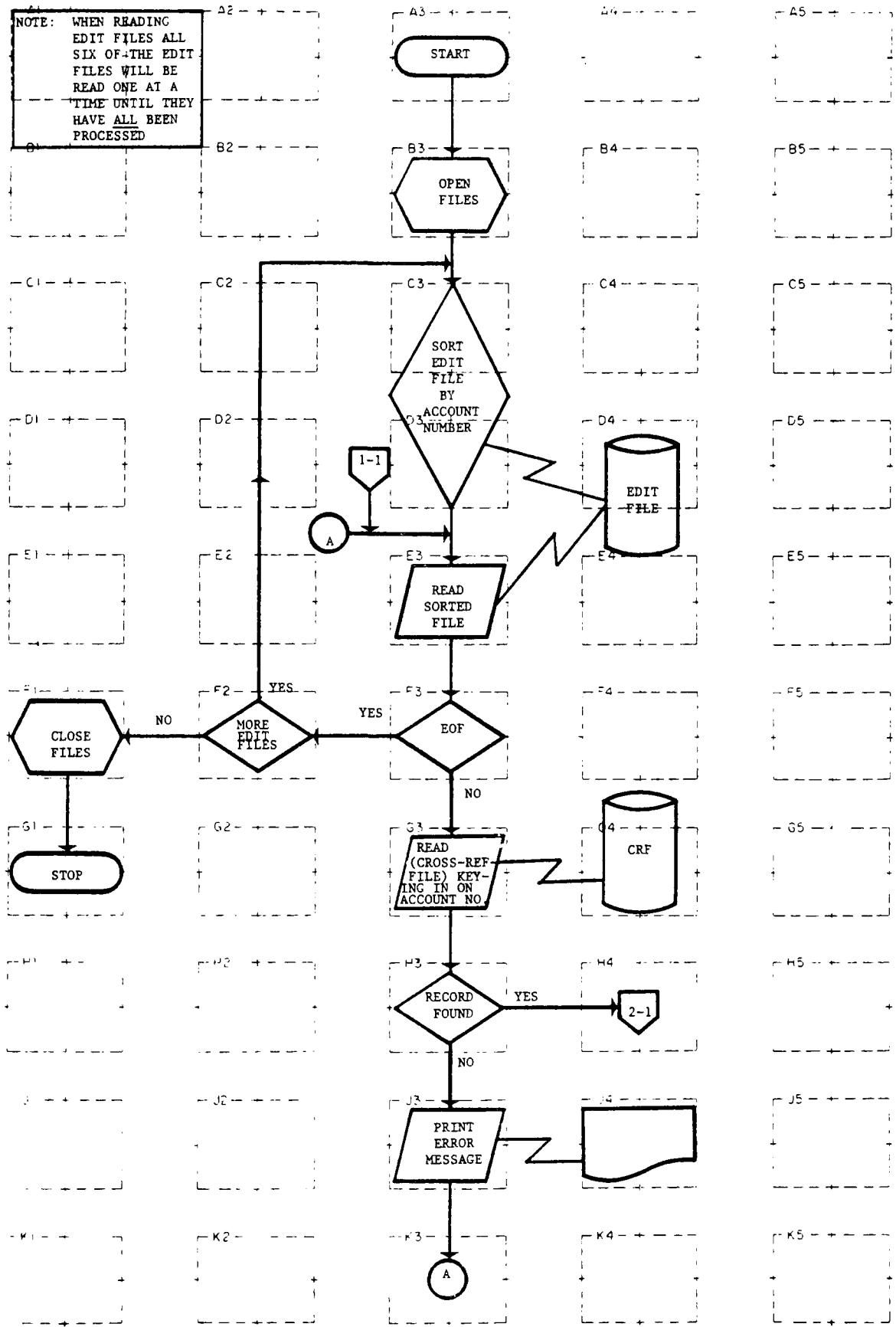


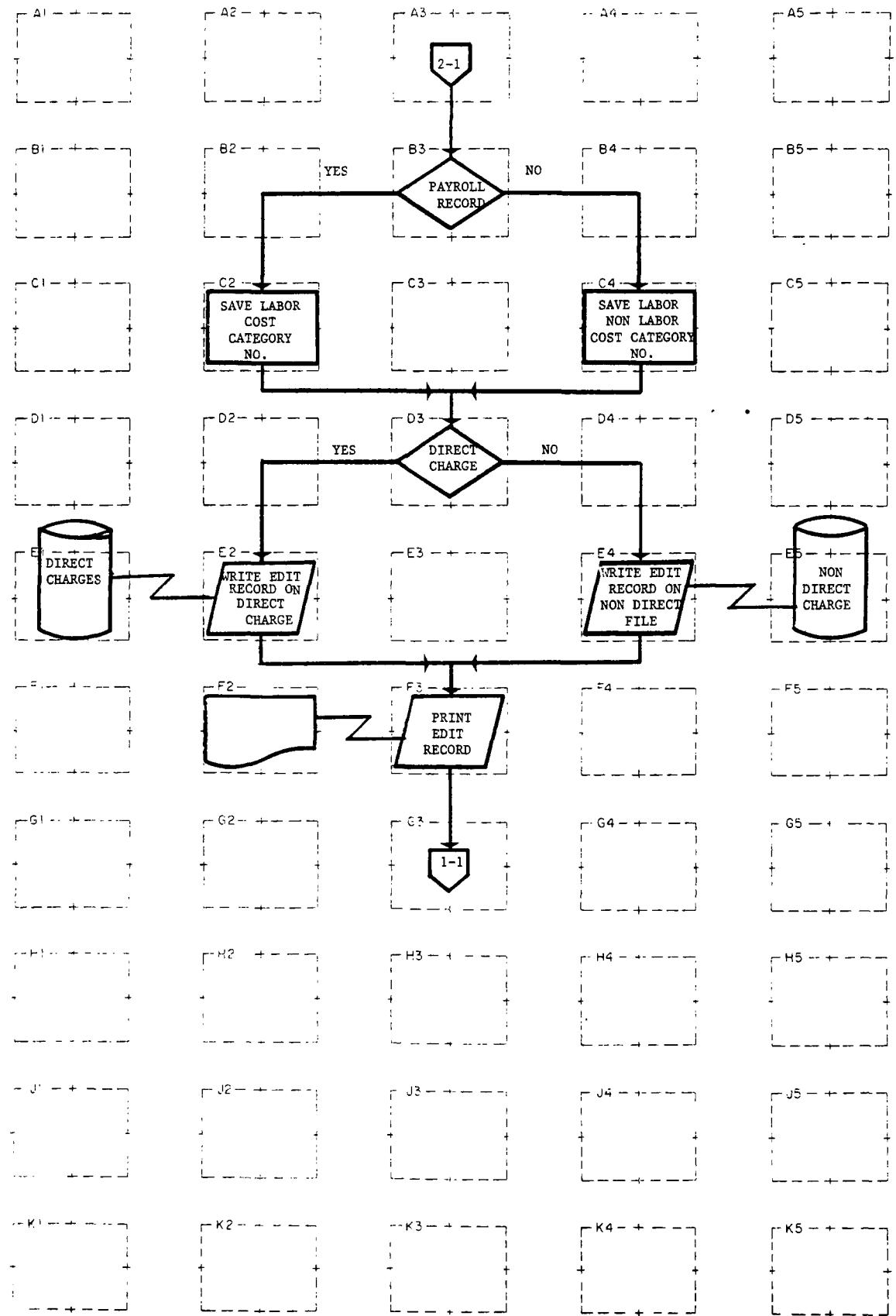


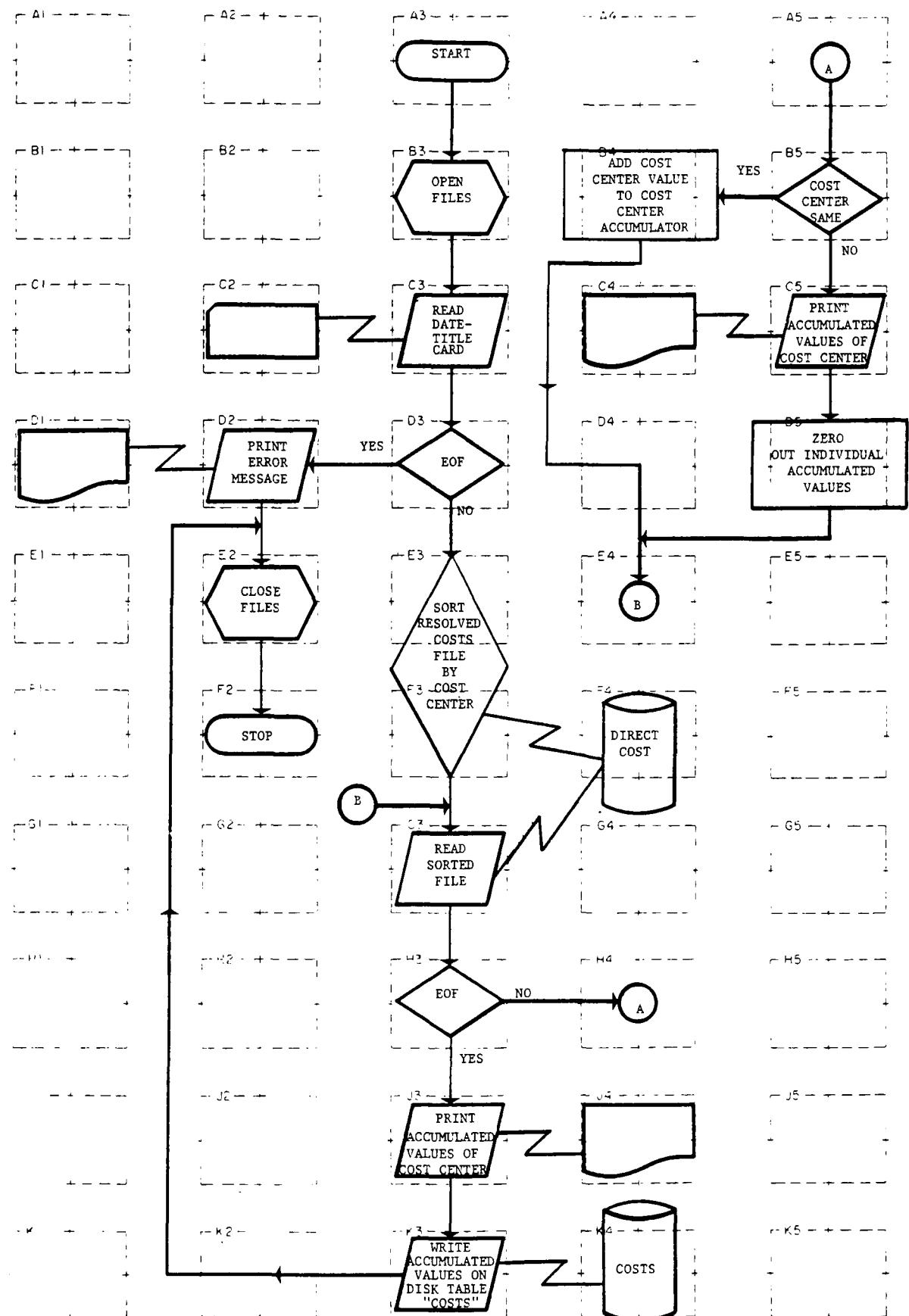


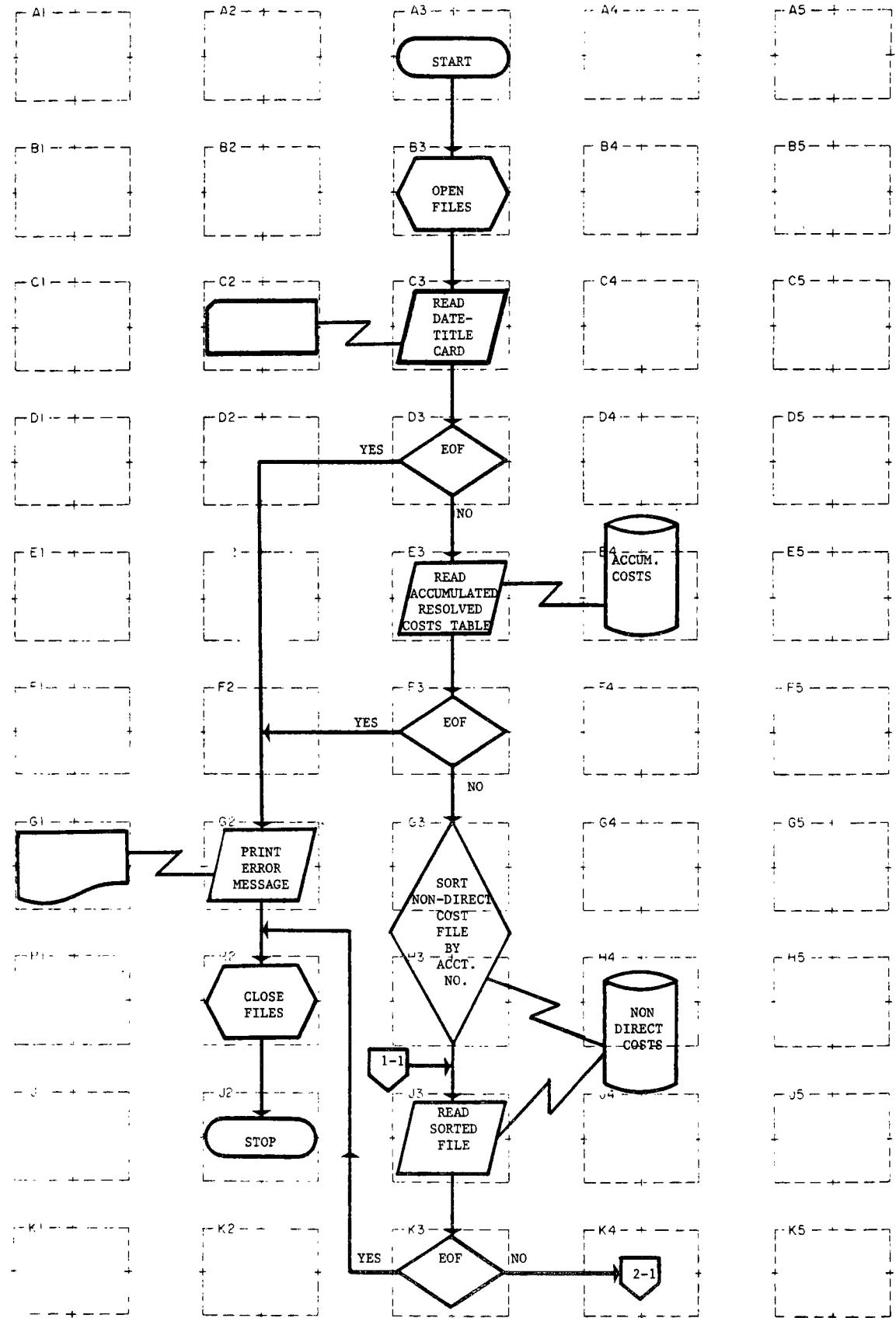


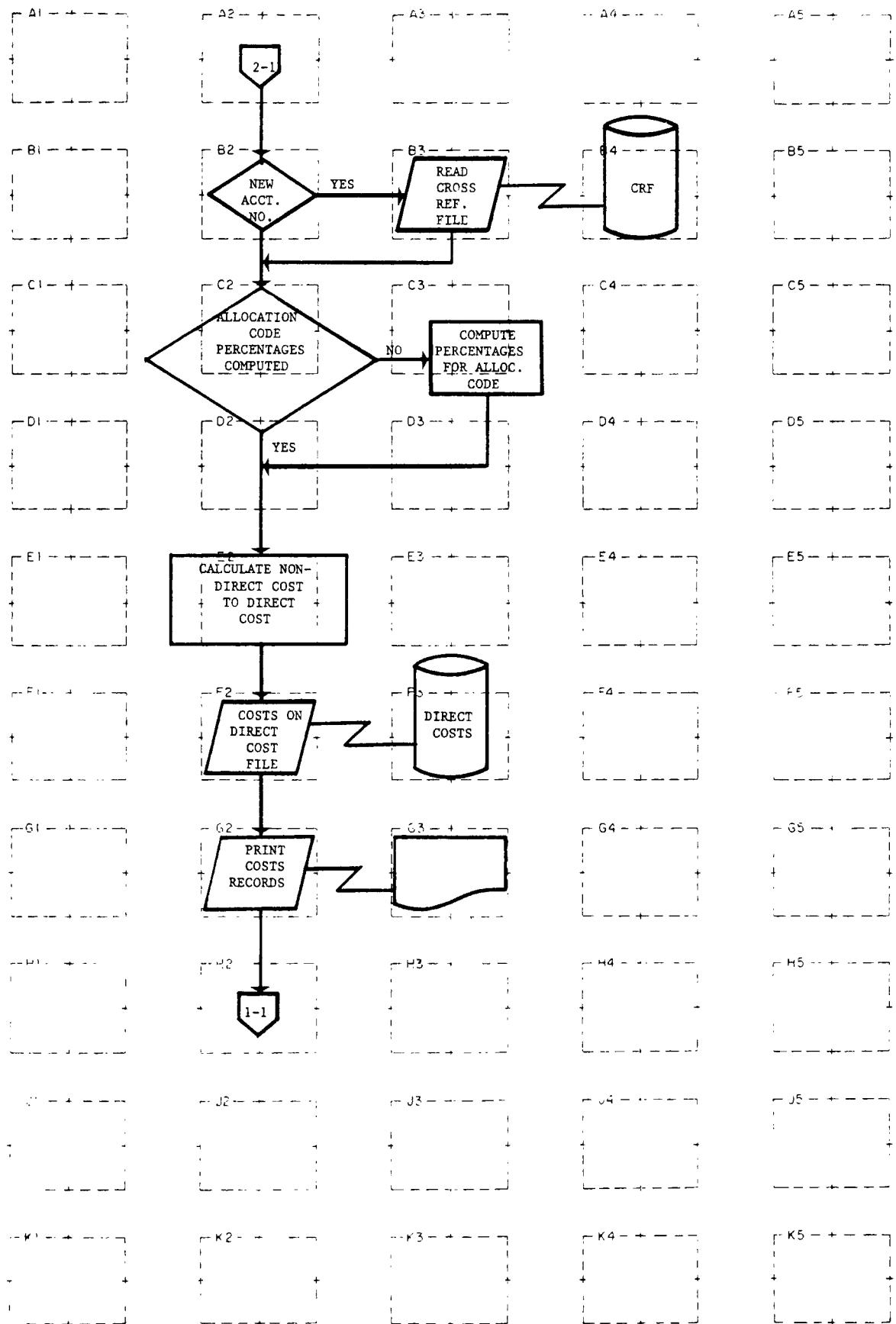


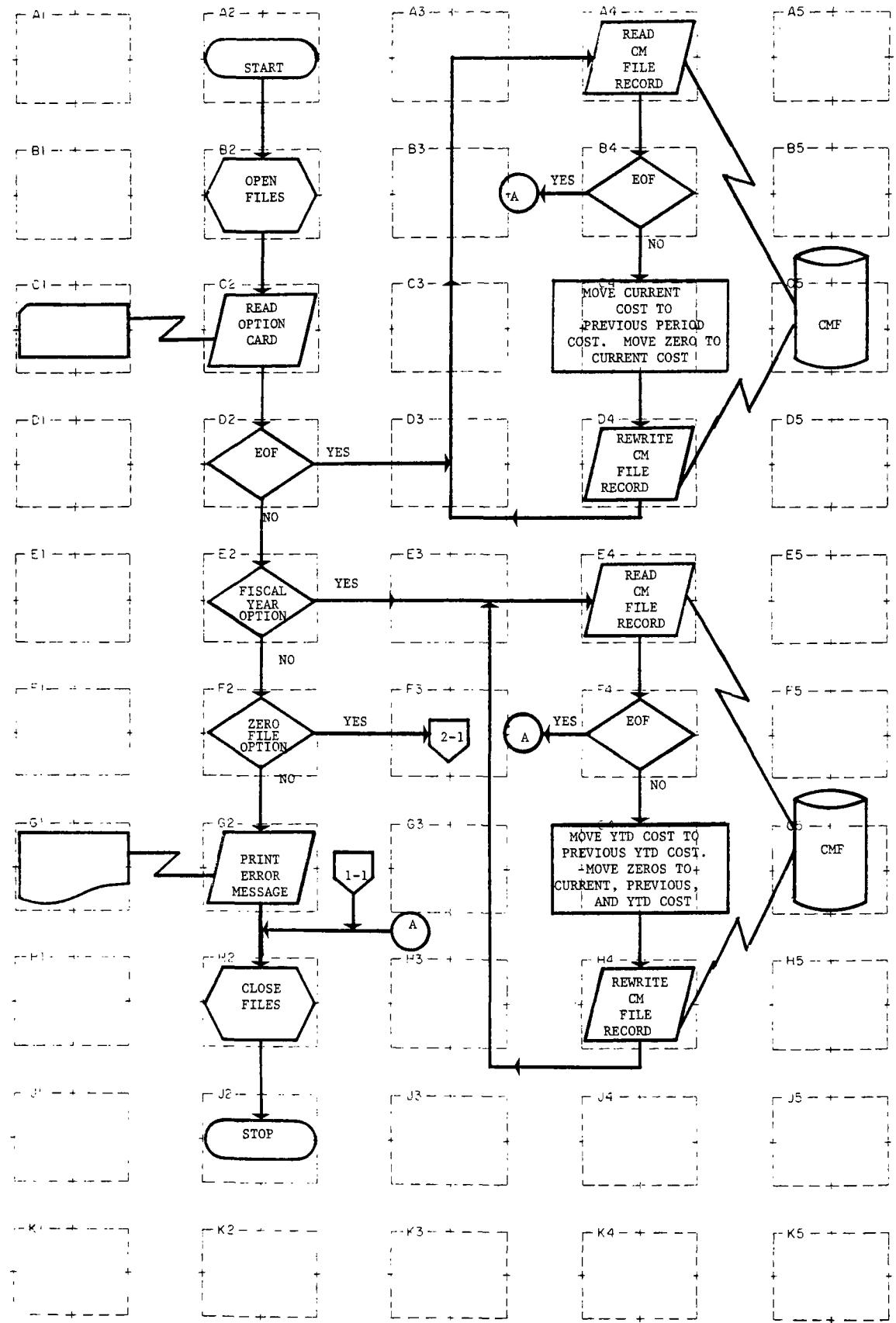


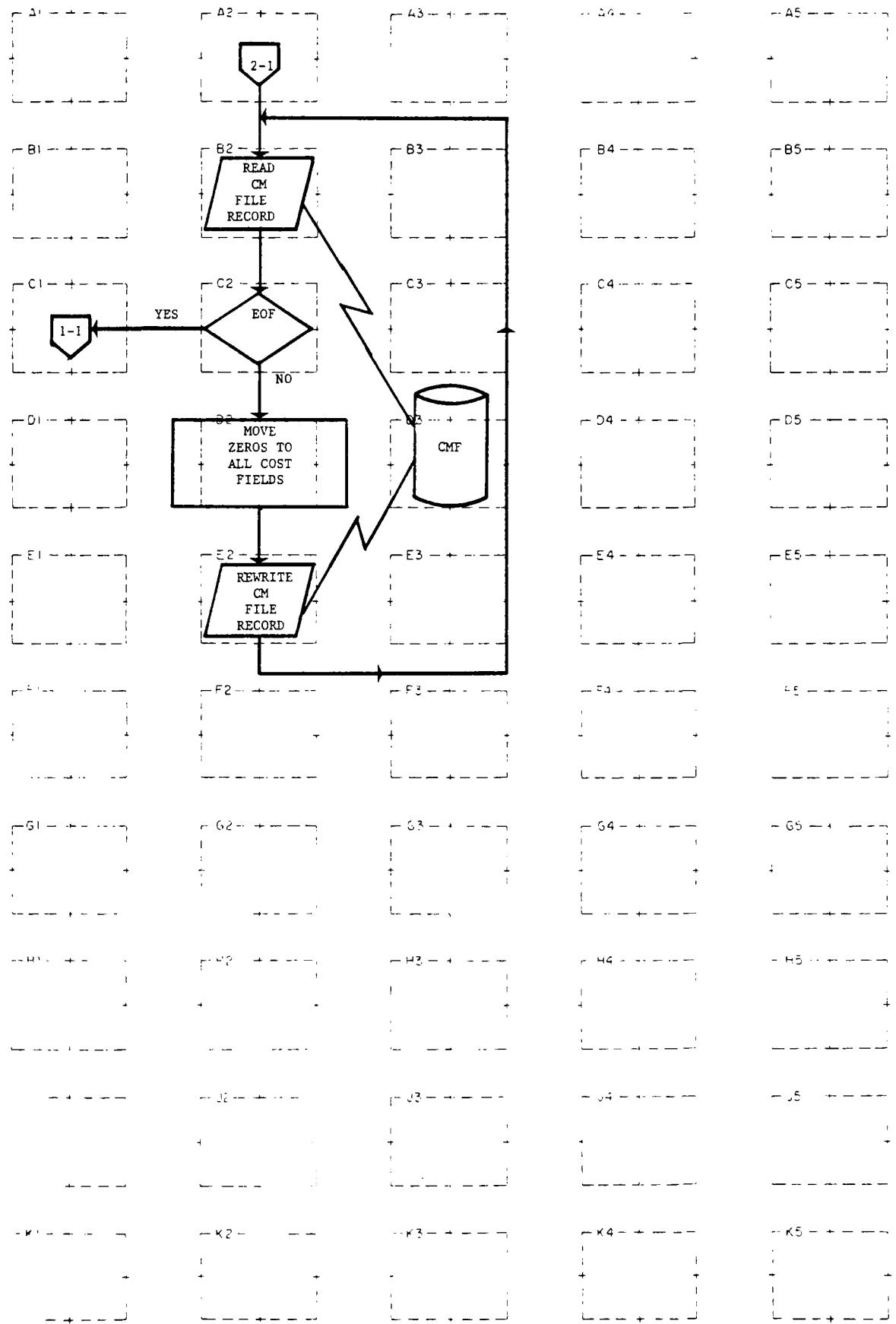


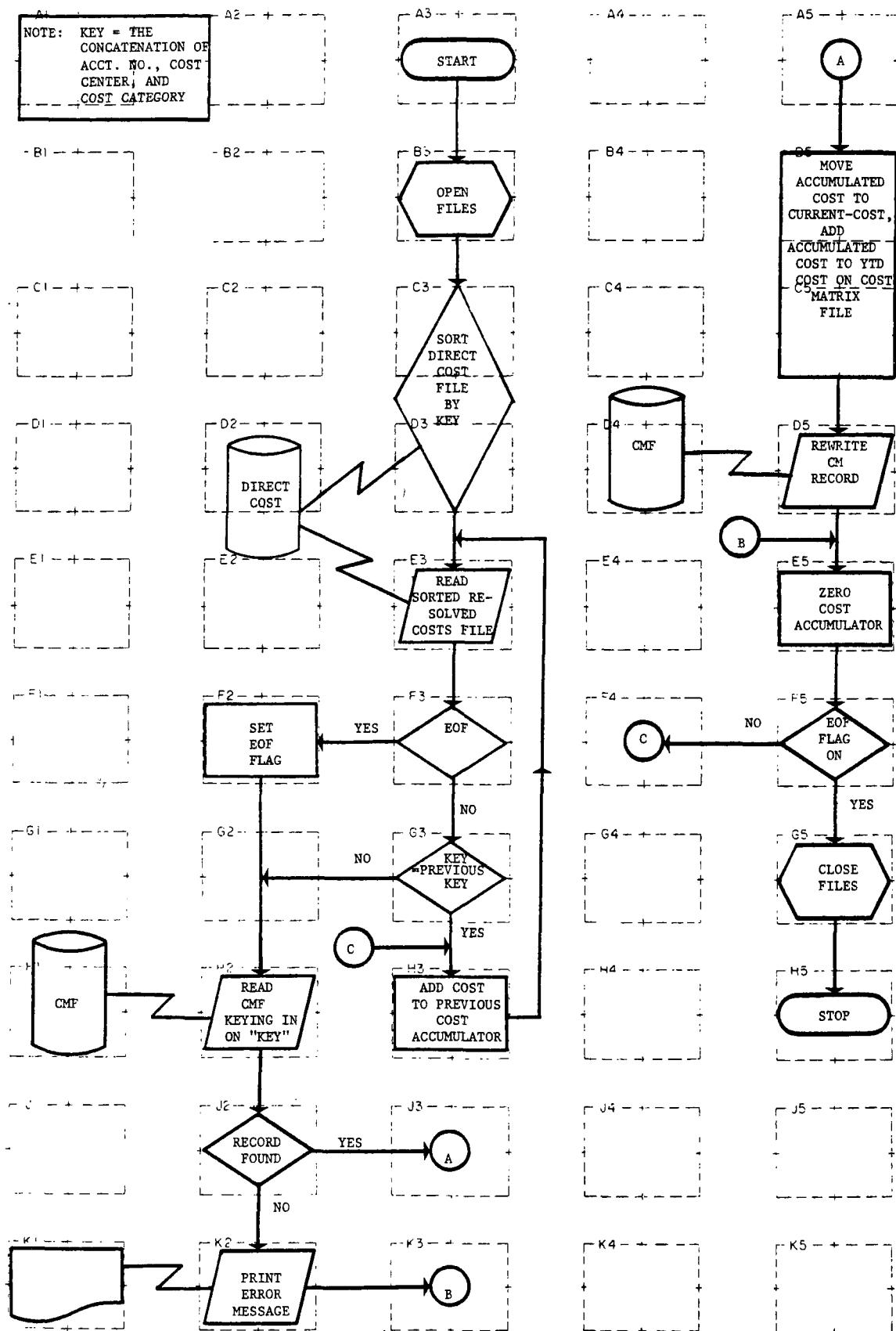


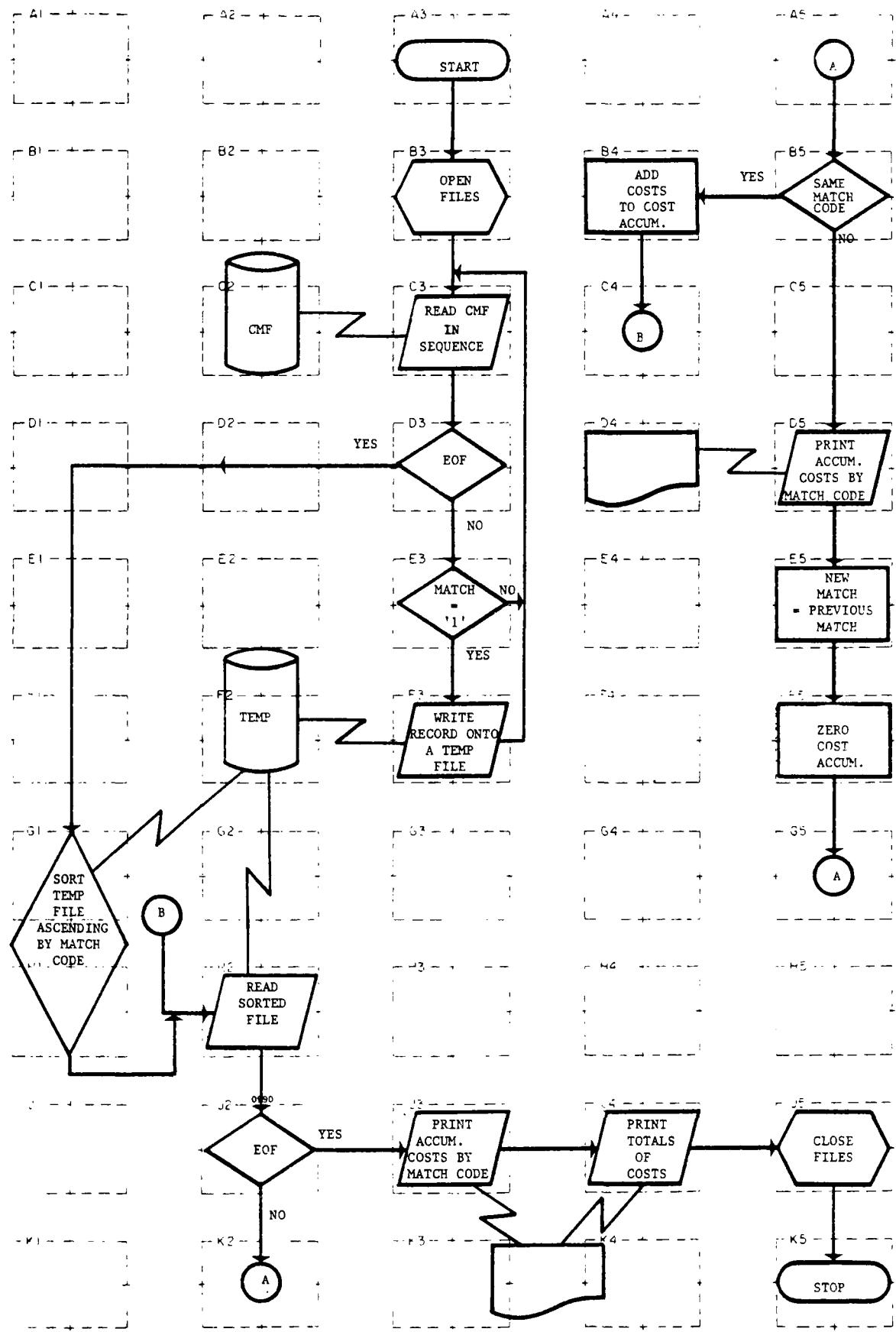


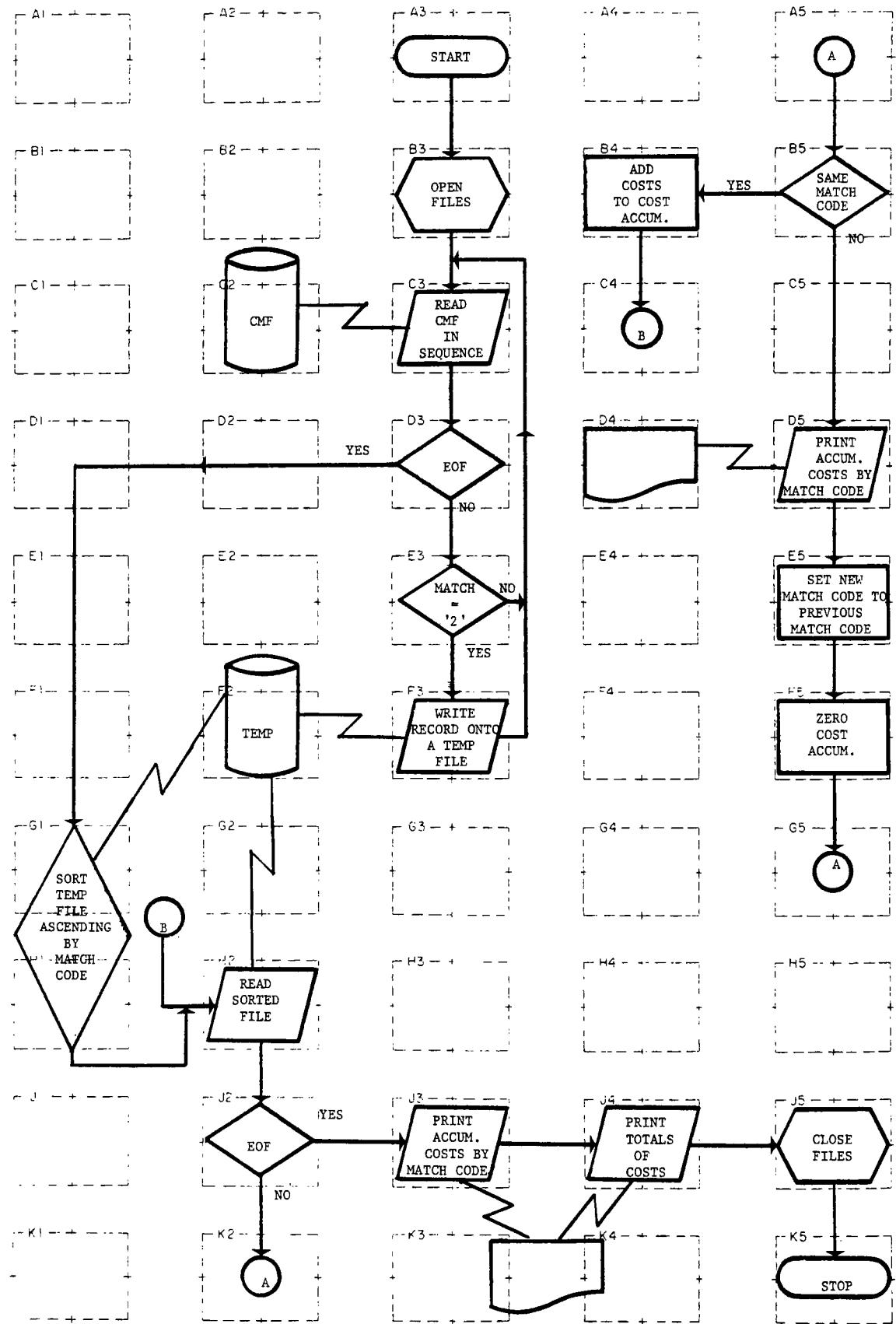


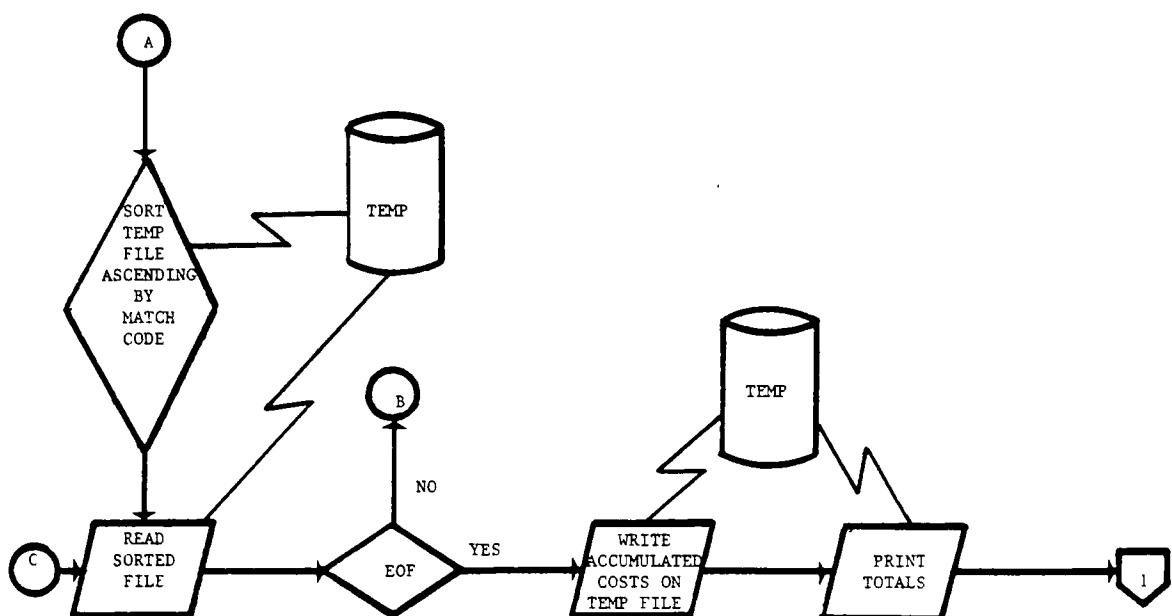
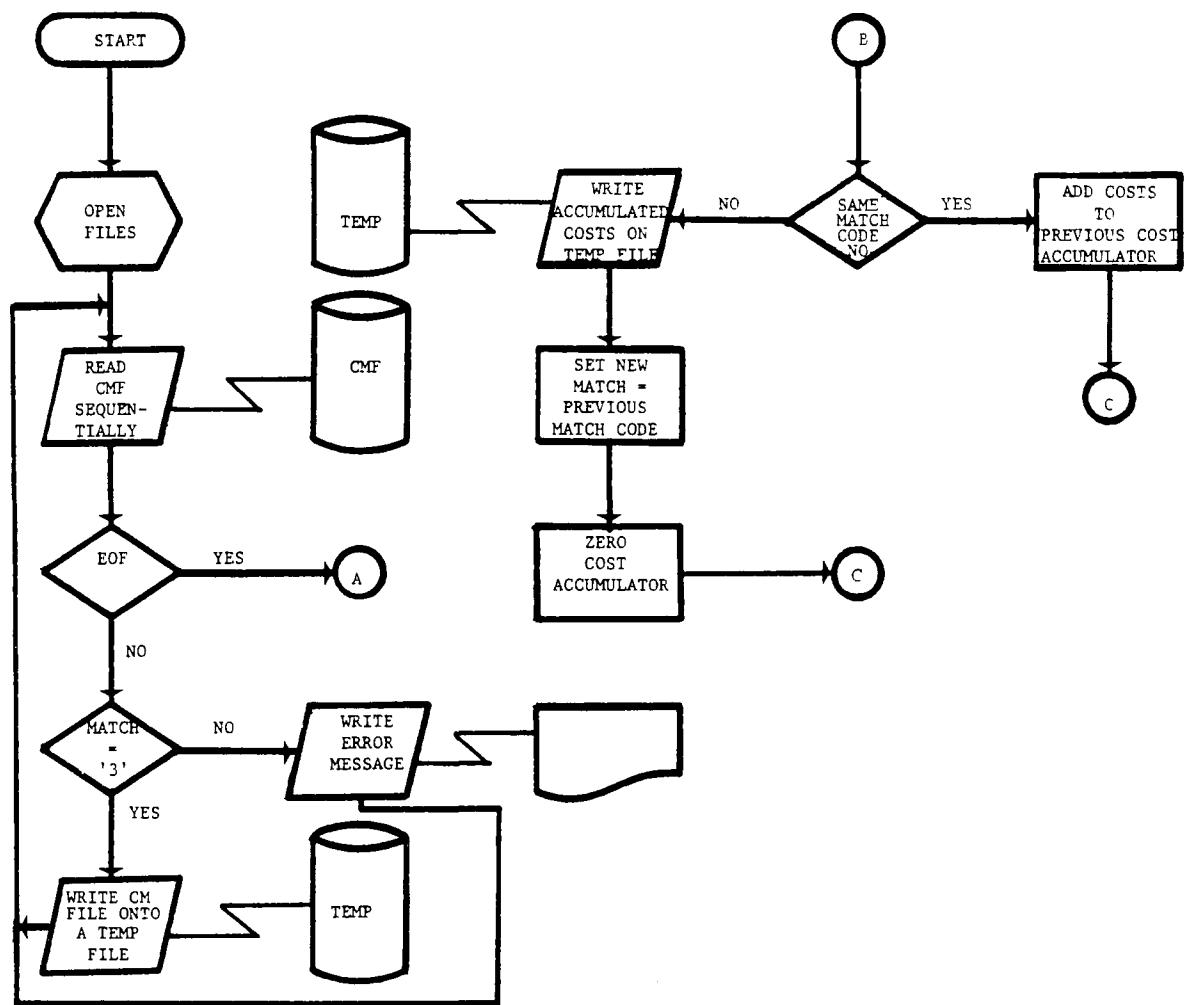


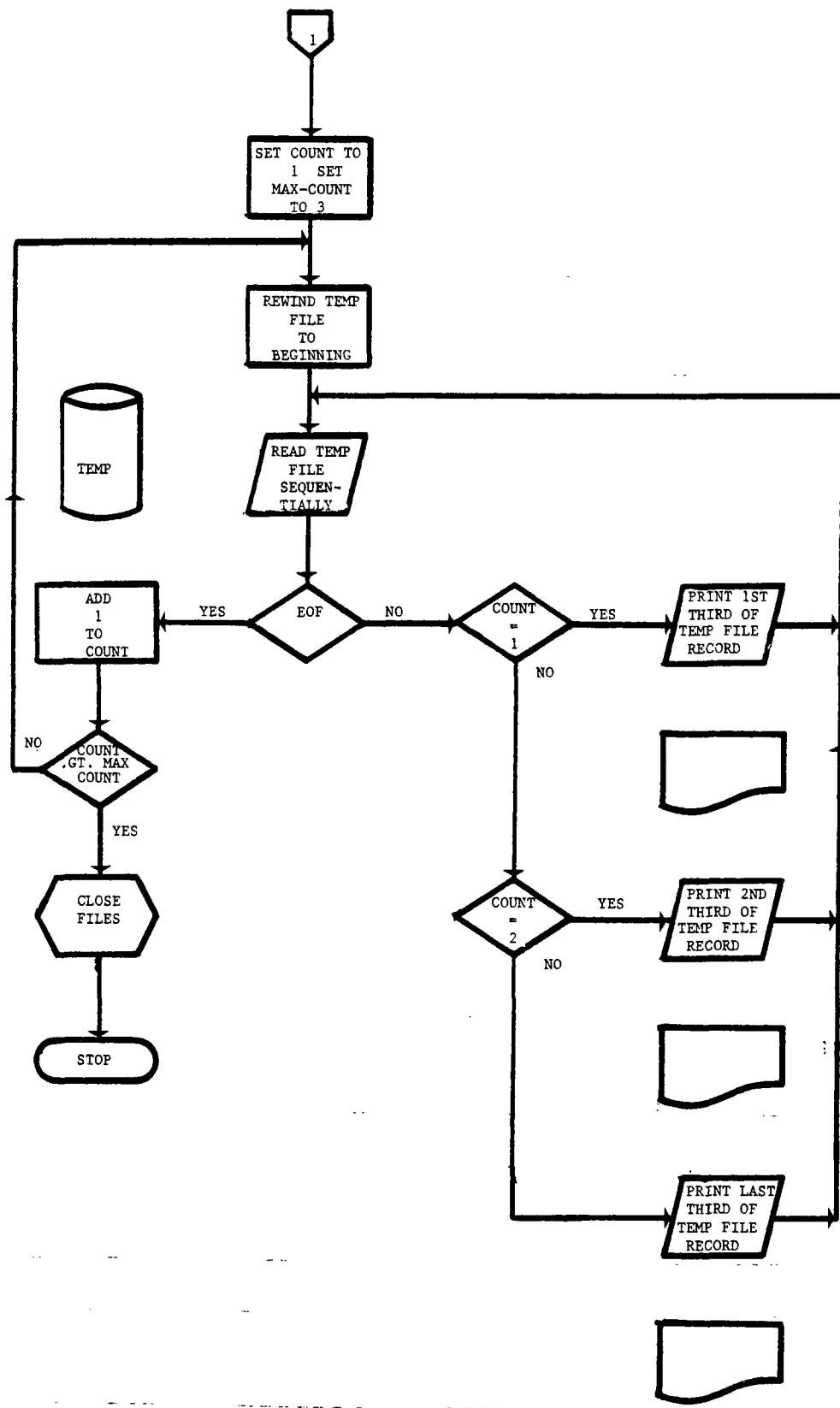


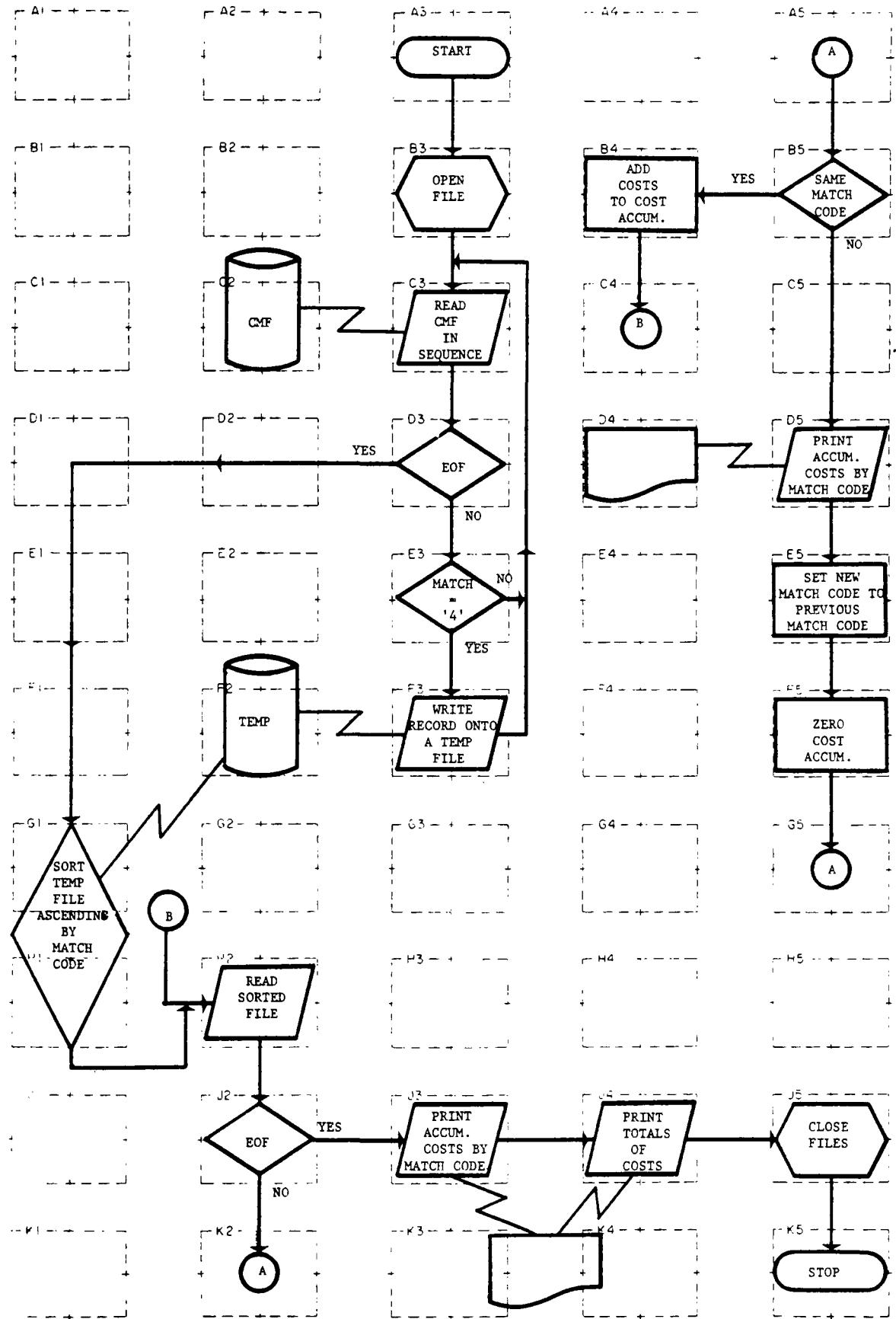


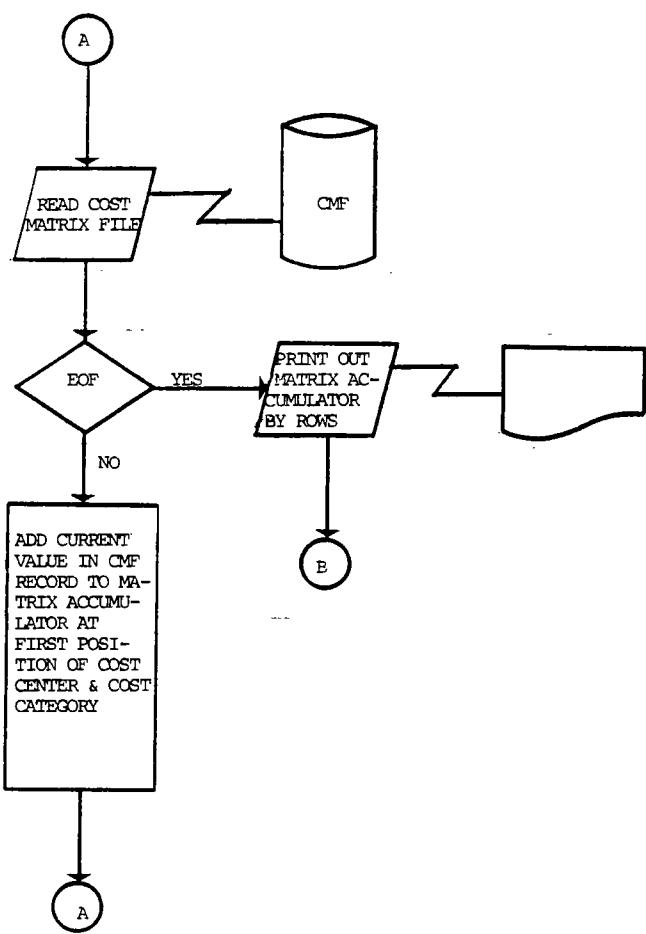
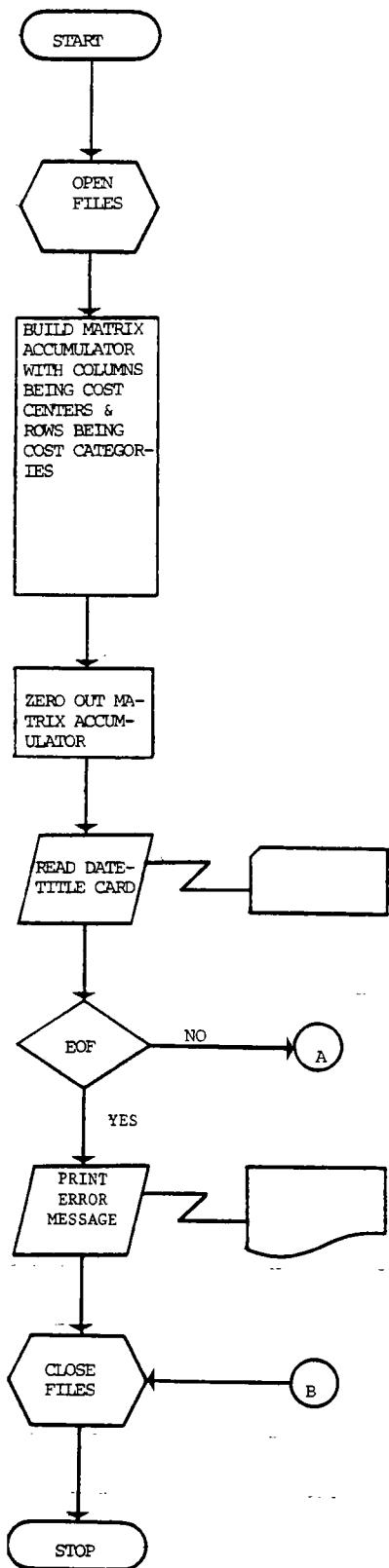


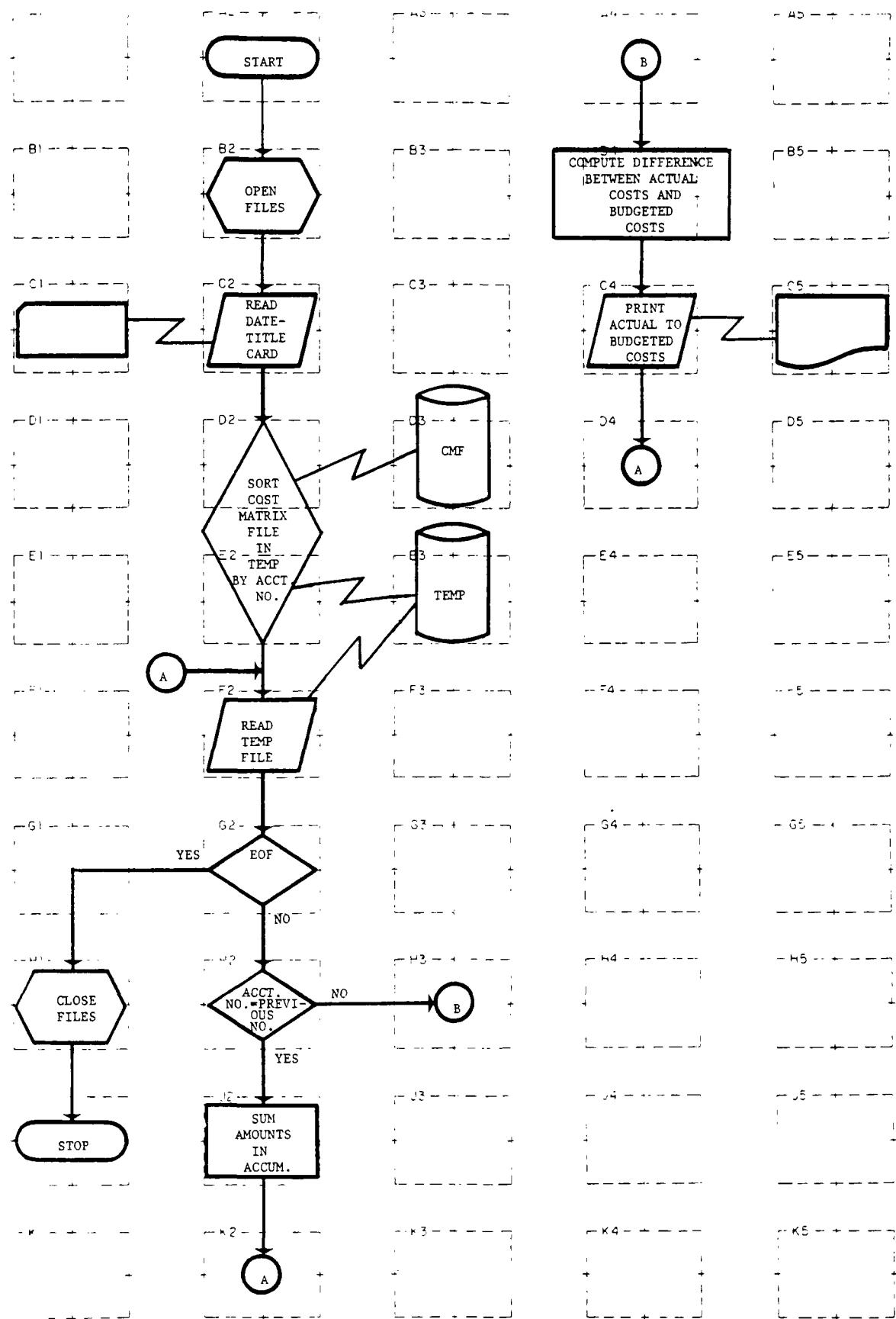


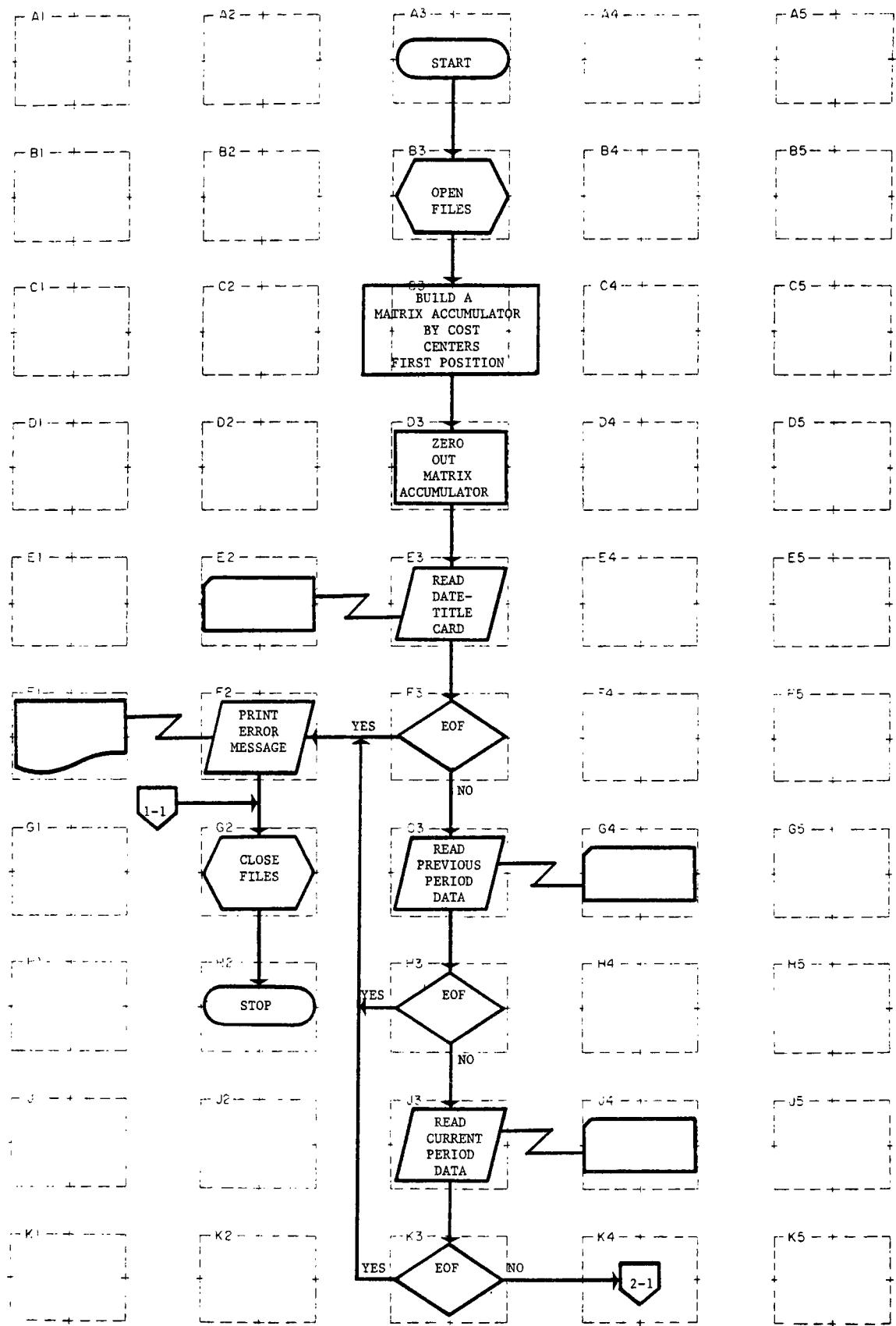


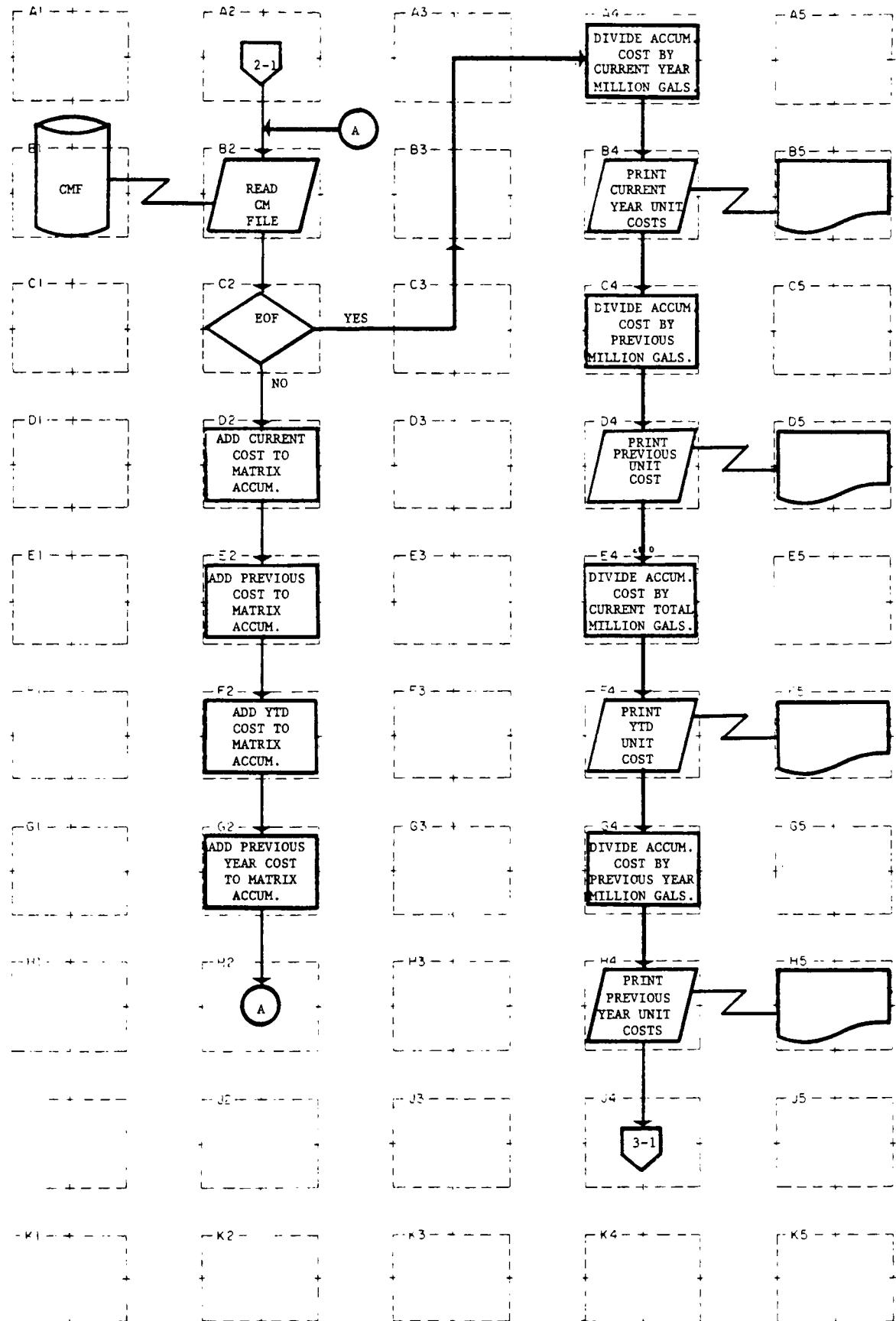


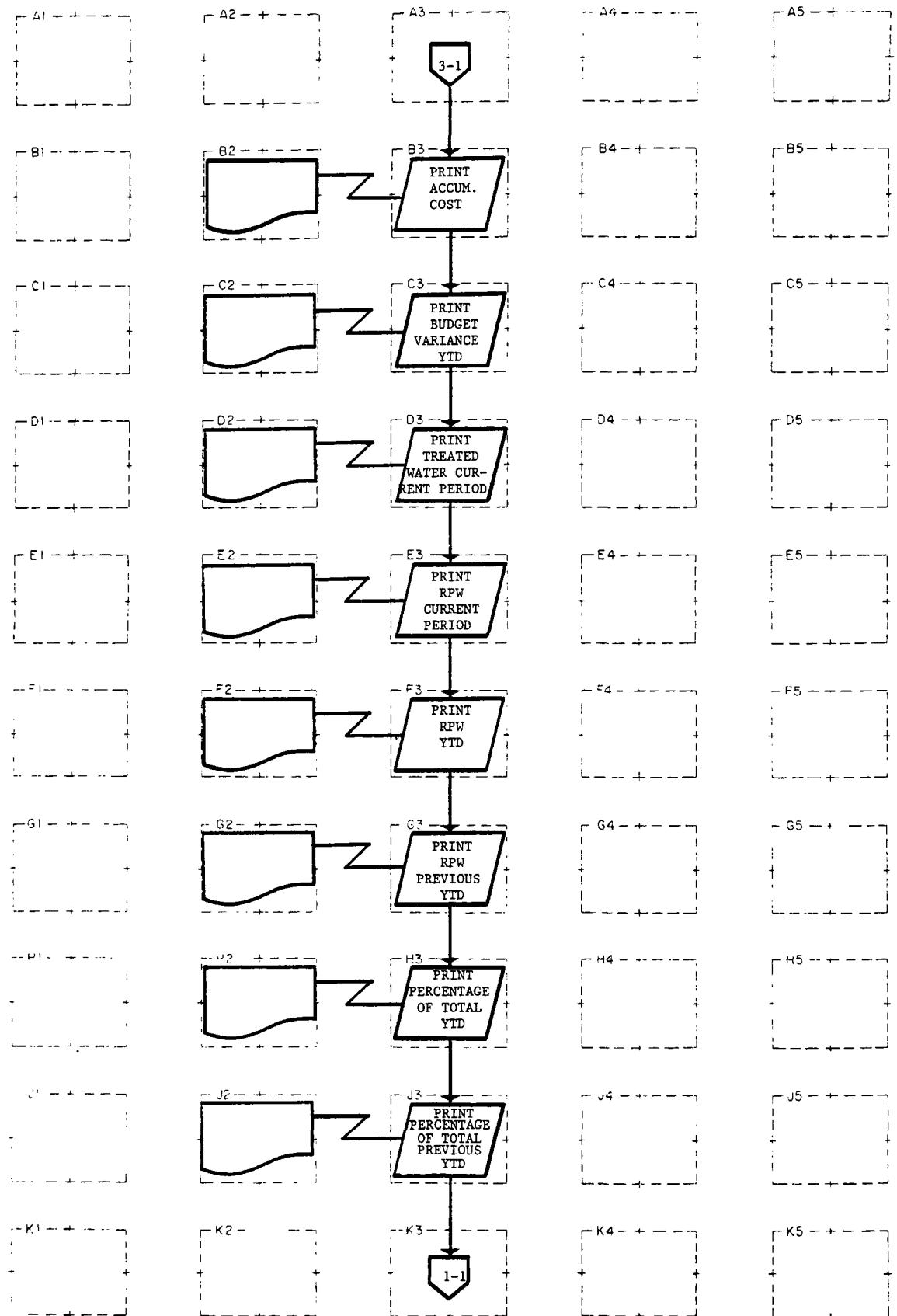


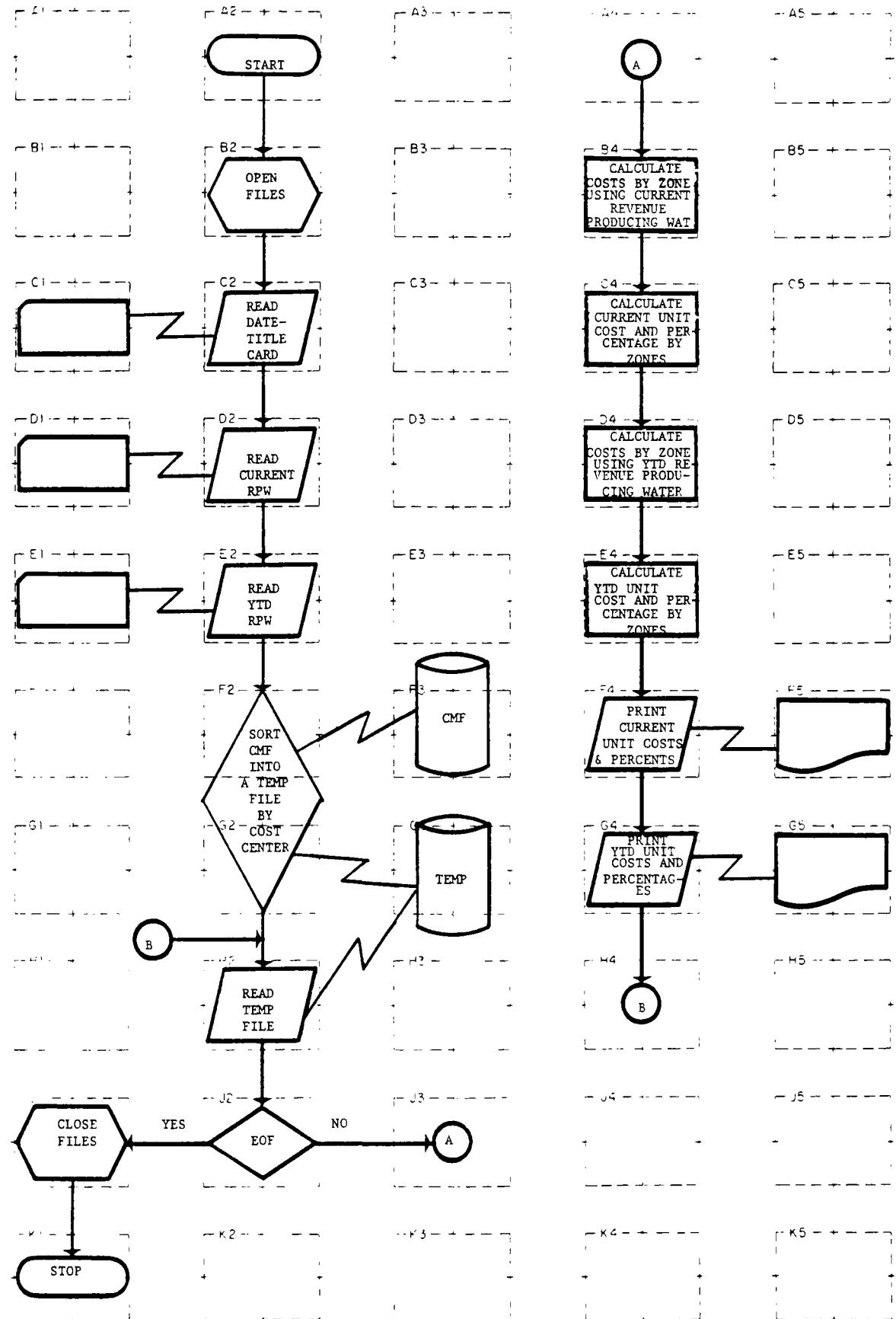


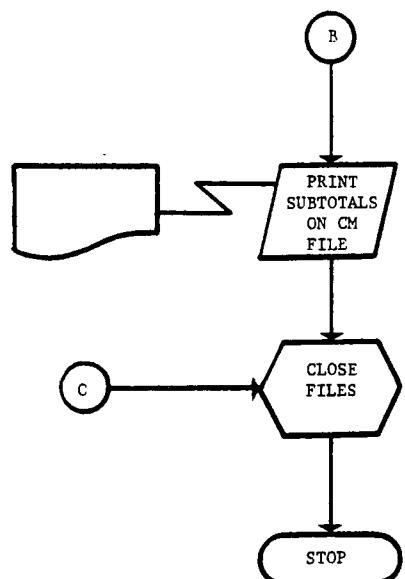
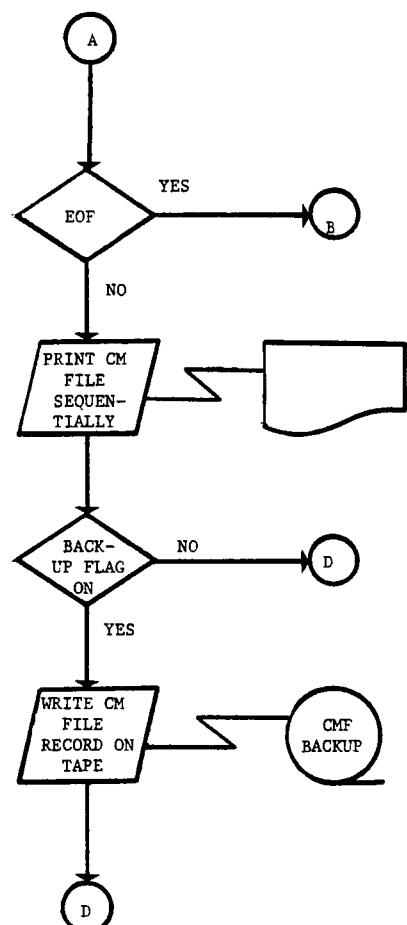
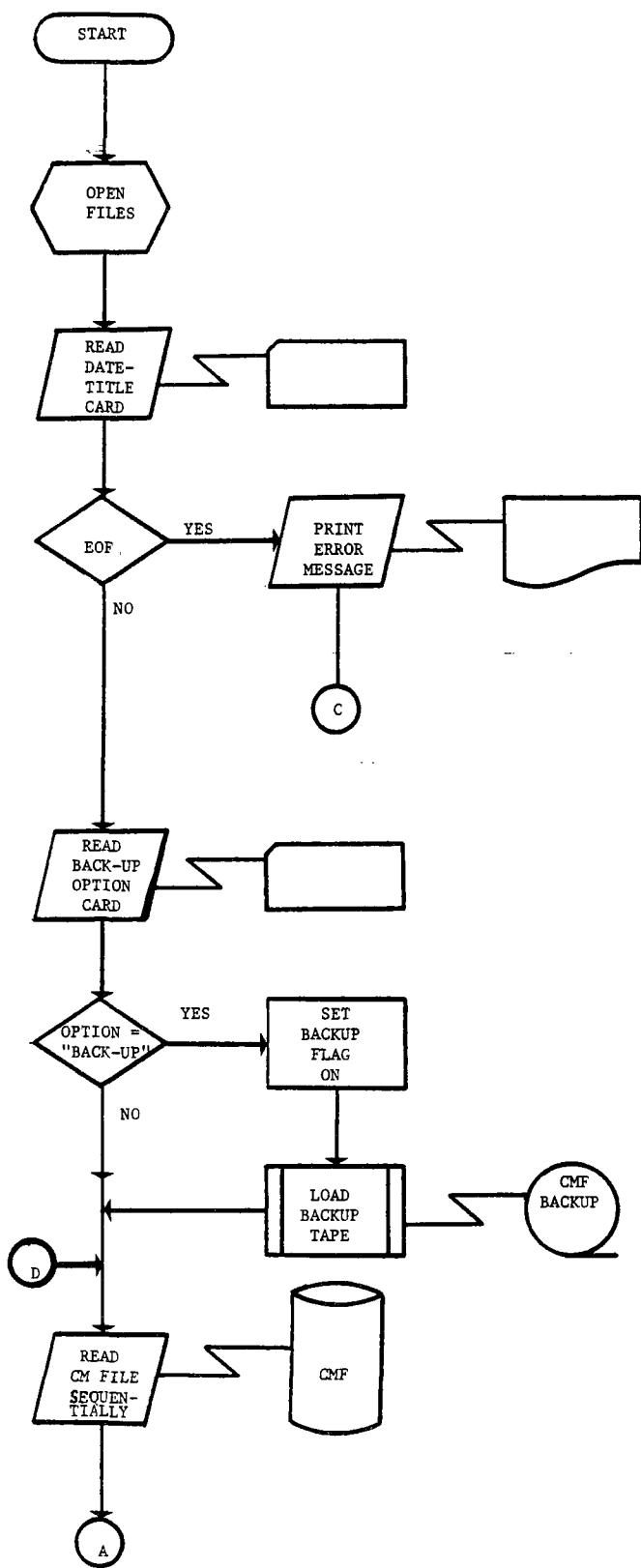


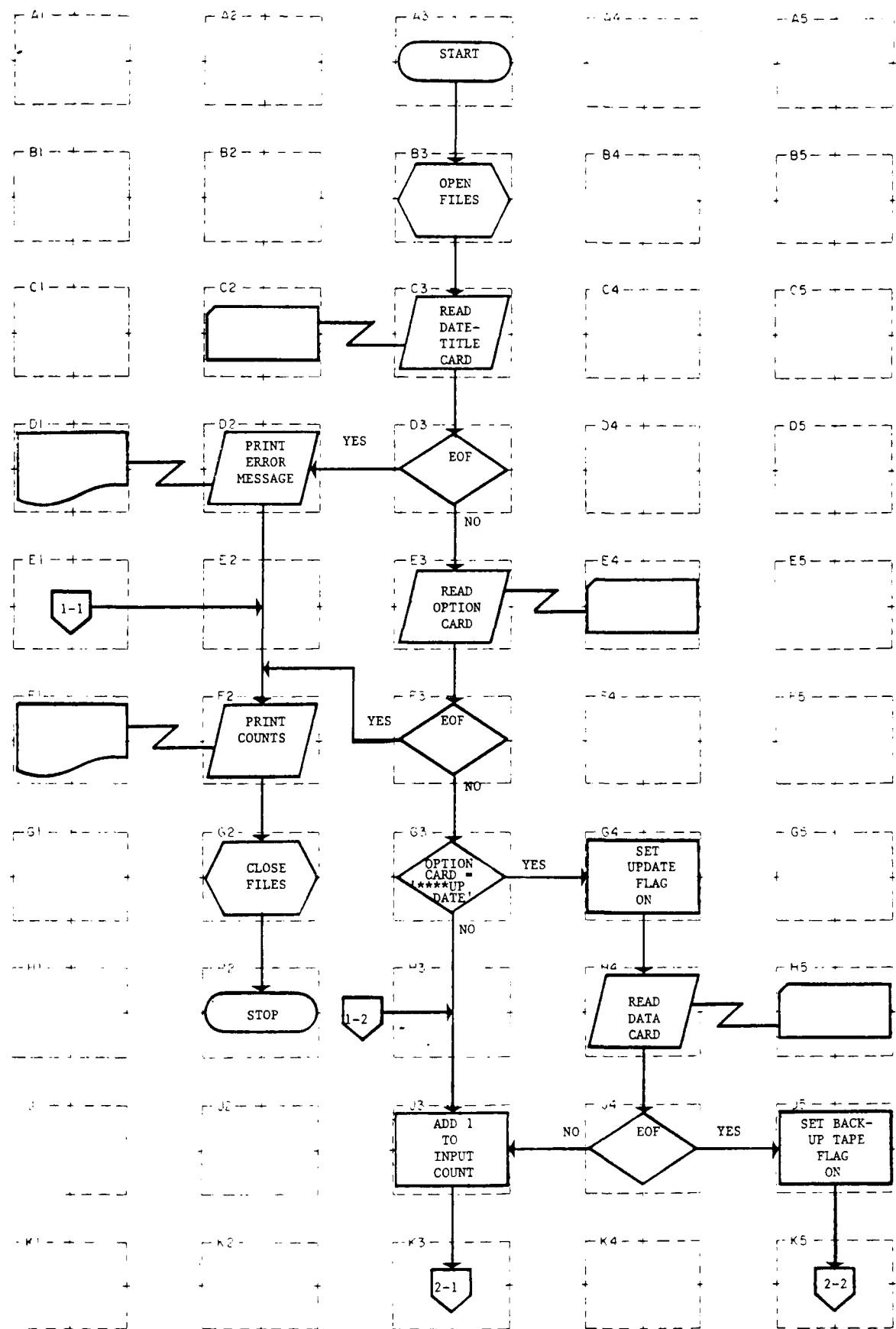


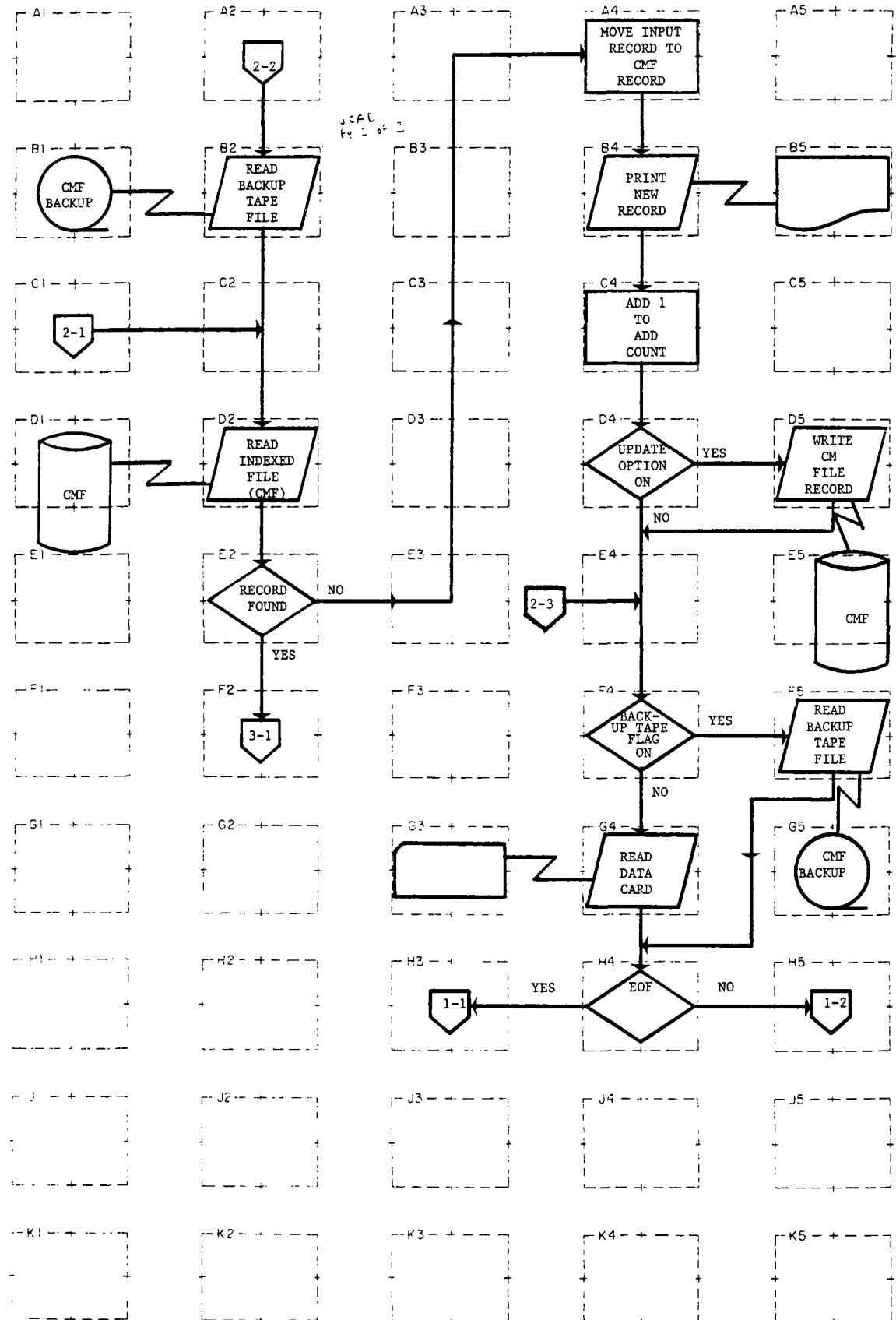


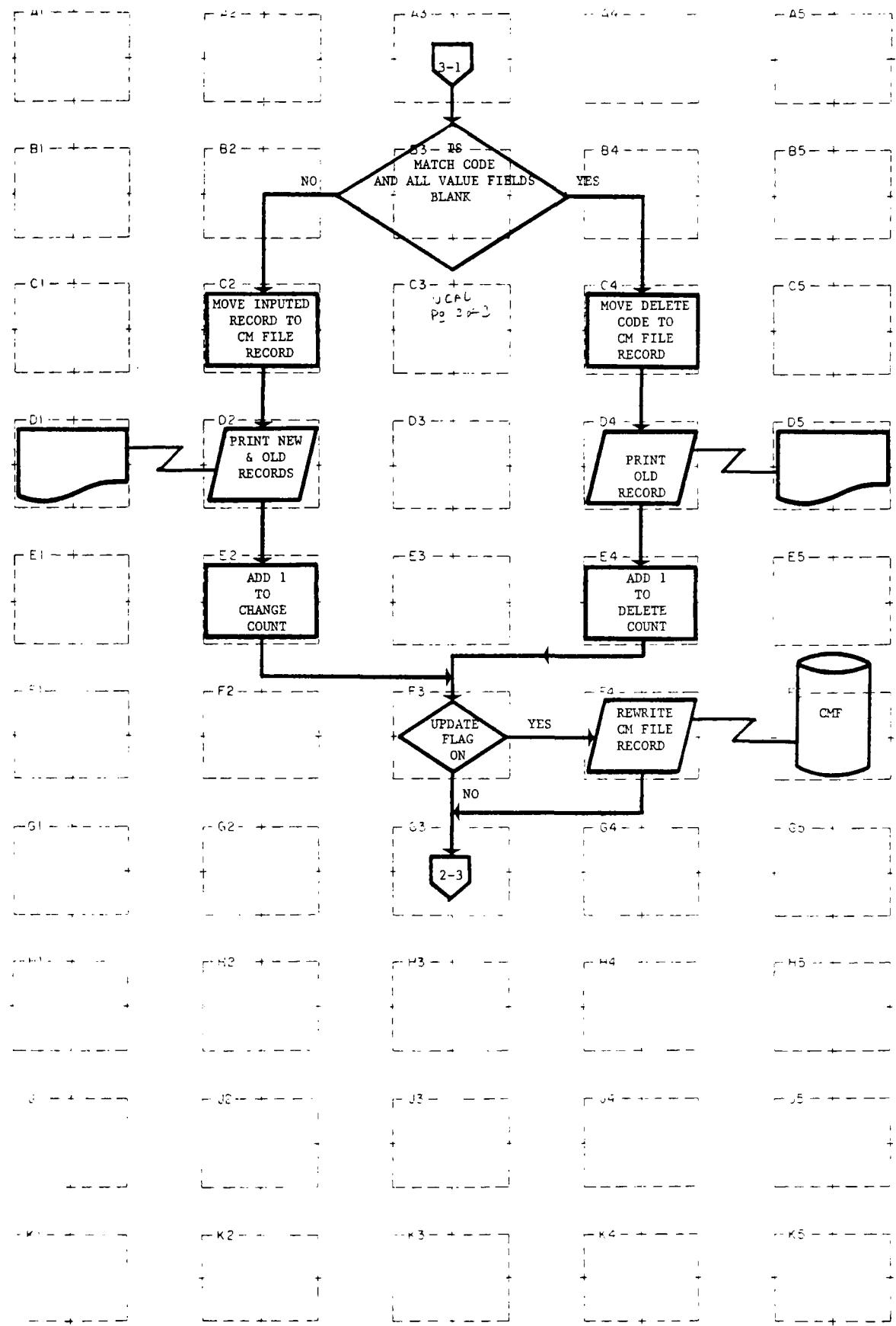












SECTION 5

PROGRAM LISTING

This section includes a source listing of all the computer programs used in the Cost Analysis System. The programs are written in American Standard COBOL within an IBM 370 environment, and thus, each program listing will provide an identification division, an operating environment division, a data format division, and a procedures division.

PGMNAME=WUASBCCT
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASBCCT.
AUTHOR. ACT SYSTEMS, INC
SUITE 200
807 W MURSE BLVD
WINTER PARK, FL 32789.
DATE-WRITTEN. MAY 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM EDITS THE COST CENTER DESCRIPTION
CARDS, FLAGS ANY DISCREPANCIES, AND BUILDS THE
COST CENTER TABLE FILE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
CO1 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-IN
ASSIGN TO UR-2540R-S-CARDIN
ACCESS MODE IS SEQUENTIAL.
SELECT PRINT-OUT
ASSIGN TO UR-1403-S-PRINTOUT
ACCESS MODE IS SEQUENTIAL.
SELECT COST-CNTR-TABFI
ASSIGN TO DA-3330-I-CCTABFI
ACCESS MODE IS SEQUENTIAL
RECORD KEY IS CC-NUMBER.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
05 CC-NO PIC X(4).
05 OVFLOW1 PIC X.
05 CC-NAME1 PIC X(13).
05 OVFLOW2 PIC X.
05 CC-NAME2 PIC X(13).
05 OVFLOW3 PIC X.
05 FILLER PIC X(47).
01 DATE-CARD.
05 DATE-IN PIC X(18).
05 TITLE-IN PIC X(24).
05 FILLER PIC X(38).
FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS

LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.

01 PRINT-LINE PIC X(133).

FD COST-CNTR-TABFI
 BLOCK CONTAINS 8 RECORDS
 RECORD CONTAINS 30 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS CC-REC.

01 CC-REC.
 05 CC-NUMBER PIC 9(4).
 05 CC-NAME1 PIC X(13).
 05 CC-NAME2 PIC X(13).

WORKING-STORAGE SECTION.

77 TITLE-1 PIC X(24).
 77 LINE-CT PIC 9(4) COMP VALUE 0.
 77 PAGE-NO PIC 9(4) COMP VALUE 1.
 77 MAX-LINE PIC 99 COMP VALUE 56.
 77 CC PIC 9 COMP.
 77 DD PIC 9 COMP.
 77 LN PIC 9 COMP.
 77 IC PIC 9 COMP.
 77 LAST-2 PIC 99.
 77 LAST-3 PIC 999.

01 LAST-CC-NO.
 05 LAST-F PIC 9(4) COMP.
 05 LAST-S PIC 9(4) COMP.
 05 LAST-I PIC 9(4) COMP VALUE 0.

01 FLAGS.
 05 BUF-FLG OCCURS 3 TIMES PIC 9.

01 HEADING-LINE.
 05 FILLER PIC X(51) VALUE SPACES.
 05 TITLE PIC X(24).
 05 FILLER PIC X(58) VALUE SPACES.

01 DATE-PAGE-LINE.
 05 FILLER PIC X VALUE SPACE.
 05 FILLER PIC X(6) VALUE 'DATE: '.
 05 DATE-1 PIC X(18).
 05 FILLER PIC X(36) VALUE SPACES.
 05 FILLER PIC X(3) VALUE 'FUR'.
 05 FILLER PIC X(58) VALUE SPACES.
 05 FILLER PIC X(5) VALUE 'PAGE '.
 05 PAGE-1 PIC ZZZ9.
 05 FILLER PIC XX VALUE SPACES.

01 TITLE-LINE.
 05 FILLER PIC X(7) VALUE SPACES.
 05 FILLER PIC X(8) VALUE 'FUNCTION'.
 05 FILLER PIC X(32) VALUE SPACES.
 05 FILLER PIC X(11) VALUE 'SUBFUNCTION'.
 05 FILLER PIC X(29) VALUE SPACES.
 05 FILLER PIC X(14) VALUE 'IDENTIFICATION'.

```
      05 FILLER PIC X(32) VALUE SPACES.  
01 DATA-OUT.  
02 DATA-LINE OCCURS 2 TIMES.  
    05 FILLER PIC X.  
    05 DATA-BLOCK OCCURS 3 TIMES.  
      10 CC-NO          PIC X(4).  
      10 FILLER        PIC XX.  
      10 CC-NAME        PIC X(13).  
      10 FILLER        PIC X.  
      10 ERROR-MSG      PIC X(20).  
    05 FILLER        PIC X(12).  
PROCEDURE DIVISION.  
OPEN-FILES.  
  OPEN INPUT CARD-IN.  
  OPEN OUTPUT PRINT-OUT, COST-CNTR-TABFI.  
CLEAR-FIELDS.  
  MOVE SPACES TO DATA-OUT.  
  MOVE ZEROS TO FLAGS.  
READ-CARDIN.  
  READ CARD-IN AT END GO TO CLOSE-ROUTINE.  
FILL-HEADING.  
  MOVE DATE-IN TO DATE-1.  
  MOVE TITLE-IN TO TITLE-1.  
PRINT-HEADING.  
  PERFORM BLANK-LINE.  
  WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.  
  MOVE 'CUST CENTER DESCRIPTIONS' TO TITLE.  
  MOVE HEADING-LINE TO PRINT-LINE.  
  PERFORM WRITE-OUT.  
  MOVE PAGE-NU TO PAGE-1.  
  MOVE DATE-PAGE-LINE TO PRINT-LINE.  
  PERFORM WRITE-OUT.  
  MOVE TITLE-1 TO TITLE.  
  MOVE HEADING-LINE TO PRINT-LINE.  
  PERFORM WRITE-OUT.  
  PERFORM BLANK-LINE THRU WRITE-OUT.  
  MOVE TITLE-LINE TO PRINT-LINE.  
  PERFORM WRITE-OUT.  
  PERFORM BLANK-LINE THRU WRITE-OUT.  
START-EDIT.  
  PERFORM READ-CARDIN.  
EDIT-STEPS.  
  MOVE .1 TO LN.  
  IF CC-NO OF INPUT-CARD NOT NUMERIC  
    MOVE 1 TO CC, DD  
    PERFORM FULL-BUFFER UNTIL FLAGS = ZEROES  
    MOVE 'NOT NUMERIC' TO ERROR-MSG (LN, CC)  
    MOVE 2 TO LN  
    PERFORM CHECK-OVERRUN THRU CHECK-END  
    PERFORM FILL-BUFFER THRU WRITE-DISK
```

```

GO TO START-EDIT.
MOVE CC-NO OF INPUT-CARD TO LAST-3, CC-NUMBER.
IF LAST-3 = 0
  MOVE 1 TO CC, DD
  PERFORM FULL-BUFFER UNTIL FLAGS = ZEROES
  MOVE CC-NO OF INPUT-CARD TO LAST-F
  PERFORM CHECK-SEQ-I THRU CHECK-END
  PERFORM FILL-BUFFER THRU WRITE-DISK
  GO TO START-EDIT.
MOVE CC-NO OF INPUT-CARD TO LAST-2.
IF LAST-2 = 0
  MOVE 2 TO CC, DD
  PERFORM FULL-BUFFER UNTIL FLAGS = ZEROES OR DD > 3
  MOVE CC-NO OF INPUT-CARD TO LAST-S
  PERFORM CHECK-SEQ-F THRU CHECK-END
  PERFORM FILL-BUFFER THRU WRITE-DISK
  GO TO START-EDIT.
EDIT-END.
MOVE 3 TO CC, DD.
PERFORM FULL-BUFFER.
PERFORM CHECK-SEQ-S THRU CHECK-END.
PERFORM FILL-BUFFER THRU WRITE-DISK.
GO TO START-EDIT.
CHECK-SEQ-S.
  IF CC-NUMBER / 100 * 100 NOT = LAST-S
    MOVE 'NO SUBFUNCTION' TO ERROR-MSG (LN, 2).
CHECK-SEQ-F.
  IF CC-NUMBER / 1000 * 1000 NOT = LAST-F
    MOVE 'NO FUNCTION' TO ERROR-MSG (LN, 1).
CHECK-SEQ-I.
  IF CC-NUMBER NOT > LAST-I
    MOVE 'SEQUENCE ERROR' TO ERROR-MSG (LN, CC)
    MOVE 2 TO LN.
  MOVE CC-NUMBER TO LAST-I.
CHECK-OVERRUN.
  IF OVFLOW1 NOT = SPACE OR OVFLOW2 NOT = SPACE
    OR OVFLOW3 NOT = SPACE
    MOVE 'FIELD OVERRUN' TO ERROR-MSG (LN, CC)
    MOVE 2 TO LN.
CHECK-TEXT.
  IF CC-NAME1 OF INPUT-CARD = SPACES
    AND CC-NAME2 OF INPUT-CARD = SPACES
    MOVE 'NO TEXT' TO ERROR-MSG (LN, CC)
    MOVE 2 TO LN.
CHECK-END.
  EXIT.
FULL-BUFFER.
* BUFFER EMPTY THEN FLAGS=0 BUFFER FULL THEN FLAGS=1
  IF BUF-FLG (DD) NOT = 0
    MOVE 0 TO IC

```

```
PERFORM WRITE-DATA 2 TIMES
MOVE ZERODES TO FLAGS
PERFORM BLANK-LINE THRU WRITE-OUT
PERFORM PAGE-OVFLOW.

ADD 1 TO DD.

WRITE-DATA.
ADD 1 TO IC.
MOVE DATA-LINE (IC) TO PRINT-LINE.
PERFORM WRITE-OUT.
MOVE SPACES TO DATA-LINE (IC).

FILL-BUFFER.
MOVE CC-NAME1 OF INPUT-CARD TO CC-NAME (1, CC).
MOVE CC-NAME2 OF INPUT-CARD TO CC-NAME (2, CC).
MOVE 1 TO BUF-FLG (CC).
MOVE CC-NO OF INPUT-CARD TO CC-NO OF DATA-LINE (1, CC).

WRITE-DISK.
IF LN NOT > 1
  MOVE CORR INPUT-CARD TO CC-REC
  WRITE CC-REC INVALID KEY
    MOVE INPUT-CARD TO PRINT-LINE
    PERFORM WRITE-OUT
    PERFORM BLANK-LINE THRU WRITE-OUT.

BLANK-LINE.
MOVE SPACES TO PRINT-LINE.

WRITE-OUT.
WRITE PRINT-LINE BEFORE ADVANCING 1.
ADD 1 TO LINE-CT.

PAGE-OVFLOW.
IF LINE-CT > MAX-LINE
  ADD 1 TO PAGE-NO
  MOVE 0 TO LINE-CT
  PERFORM PRINT-HEADING.

CLOSE-ROUTINE.
IF FLAGS > 0
  MOVE 0 TO IC
  PERFORM WRITE-DATA 2 TIMES.

CLOSE-FILES.
CLOSE CARD-IN, PRINT-OUT, COST-CNTR-TABFI.

PROGRAM-END.
STOP RUN.
```

PGMNAME=WUASBSAT
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASBSAT.
AUTHOR. ACT SYSTEMS, INC
SUITE 200
807 W MORSE BLVD
WINTER PARK, FL 32789.
DATE-WRITTEN. JUNE 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM EDITS THE SPECIAL ACCOUNT DESCRIPTION
CARDS, FLAGS ANY DISCREPANCIES, AND BUILDS THE SPECIAL
ACCOUNT SEARCH TABLE FILE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-IN
ASSIGN TO UR-2540R-S-CARDIN
ACCESS MODE IS SEQUENTIAL.
SELECT PRINT-OUT
ASSIGN TO UR-1403-S-PRINTOUT
ACCESS MODE IS SEQUENTIAL.
SELECT SPECIAL-ACT-TABFI
ASSIGN TO UT-3330-S-SATABFI
ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
05 SA-NO PIC XX.
05 SA-OVFLOW1 PIC X.
05 SA-NAME PIC X(20).
05 SA-OVFLOW2 PIC X.
05 FILLER PIC X(56).
01 DATE-CARD.
05 DATE-IN PIC X(18).
05 TITLE-IN PIC X(24).
05 FILLER PIC X(38).
FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.
01 PRINT-LINE PIC X(133).

FD SPECIAL-ACCT-TABFI
 BLOCK CONTAINS 0 RECORDS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS SPEC-ACCT-REC.
 01 SPFC-ACCT-REC.
 05 RECORD-COUNT.
 10 NO-OF-RECS PIC 9(4).
 10 FILLER PIC X(18).
 05 SPEC-ACCT OCCURS 1 TO 99 TIMES
 DEPENDING ON NO-OF-RECS
 ASCENDING KEY IS SA-NO INDEXED BY IA.
 10 SA-NO PIC XX.
 10 SA-NAME PIC X(20).
 WORKING-STORAGE SECTION.
 77 LAST-SA PIC 99 VALUE 0.
 77 LINE-CT PIC 9(4) COMP VALUE 0.
 77 PAGE-NO PIC 9(4) COMP VALUE 1.
 77 MAX-LINE PIC 99 COMP VALUE 56.
 77 IE PIC 9 COMP.
 01 HEADING-LINE.
 05 FILLER PIC X(52) VALUE SPACES.
 05 FILLER PIC X(28) VALUE 'SPECIAL ACCOUNT DESCRIPTIONS'.
 05 FILLER PIC X(53) VALUE SPACES.
 01 DATE-LINE.
 05 FILLER PIC X VALUE SPACE.
 05 FILLER PIC X(6) VALUE 'DATE: '.
 05 DATE-1 PIC X(18).
 05 FILLER PIC X(40) VALUE SPACES.
 05 FILLER PIC X(3) VALUE 'FOR'.
 05 FILLER PIC X(53) VALUE SPACES.
 05 FILLER PIC X(5) VALUE 'PAGE '.
 05 PAGE-1 PIC ZZZ9.
 05 FILLER PIC XX VALUE SPACES.
 01 TITLE-LINE.
 05 FILLER PIC X(54) VALUE SPACES.
 05 TITLE PIC X(24).
 05 FILLER PIC X(55) VALUE SPACES.
 01 DATA-OUT.
 05 FILLER PIC X(41).
 05 SA-NO PIC XX.
 05 FILLER PIC X(8).
 05 SA-NAME PIC X(20).
 05 FILLER PIC X(5).
 05 SA-ERROR1 PIC X(20).
 05 FILLER PIC X(5).
 05 SA-ERROR2 PIC X(20).
 05 FILLER PIC X(12).
 01 ERR-FIELDS.
 05 ERROR-FLD OCCURS 2 TIMES PIC X(20).
 PROCEDURE DIVISION.

```

OPEN-FILES.
  OPEN INPUT CARD-IN.
  OPEN OUTPUT PRINT-OUT, SPECIAL-ACCT-TABFI.
CLEAR-FIELDS.
  MOVE SPACES TO DATA-OUT.
  MOVE SPACES TO ERR-FIELDS.
  MOVE ZERO TO NO-OF-RECS.
  SET IA TO 1.
READ-CARDIN.
  READ CARD-IN AT END GO TO CLOSE-ROUTINE.
FILL-HEADING.
  MOVE DATE-IN TO DATE-1.
  MOVE TITLE-IN TO TITLE.
PRINT-HEADING.
  PERFORM BLANK-LINE.
  WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
  MOVE HEADING-LINE TO PRINT-LINE.
  PERFORM WRITE-OUT.
  MOVE PAGE-NO TO PAGE-1.
  MOVE DATE-LINE TO PRINT-LINE.
  PERFORM WRITE-OUT.
  MOVE TITLE-LINE TO PRINT-LINE.
  PERFORM WRITE-OUT.
START-EDIT.
  PERFORM READ-CARDIN.
  IF SA-NO OF INPUT-CARD = '***'
    PERFORM BLANK-LINE THRU WRITE-OUT
    MOVE SA-NAME OF INPUT-CARD TO SA-NAME OF DATA-OUT
    MOVE DATA-OUT TO PRINT-LINE
    PERFORM WRITE-OUT
    PERFORM BLANK-LINE THRU WRITE-OUT
    GO TO START-EDIT.
  MOVE 1 TO IE.
ERROR-CHECKS.
  IF SA-NO OF INPUT-CARD IS NOT NUMERIC
    MOVE 'NOT NUMERIC' TO ERROR-FLD (IE)
    MOVE 2 TO IE.
  IF SA-NAME OF INPUT-CARD = SPACES
    MOVE 'NO TEXT' TO ERROR-FLD (IE)
    MOVE 2 TO IE.
  IF SA-OVFLOW1 NOT = SPACE OR SA-OVFLOW2 NOT = SPACE
    MOVE 'FIELD OVERRUN' TO ERROR-FLD (IE)
    MOVE 2 TO IE.
  IF SA-NO OF INPUT-CARD NOT > LAST-SA
    AND SA-NO OF INPUT-CARD NOT = ZERUS
    MOVE 'SEQUENCE ERROR' TO ERROR-FLD (IE)
    MOVE 2 TO IE.
  IF IE < 2
    PERFORM FILL-DATA.
  MOVE SA-NO OF INPUT-CARD TO LAST-SA.

```

```
WRITE-DATA.  
    MOVE CORR INPUT-CARD TO DATA-OUT.  
    MOVE ERROR-FLD (1) TO SA-ERRCR1.  
    MOVE ERROR-FLD (2) TO SA-ERROK2.  
    MOVE DATA-OUT TO PRINT-LINE.  
    PERFORM WRITE-OUT.  
    MOVE SPACES TO ERR-FIELDS, DATA-OUT.  
    GO TO START-EDIT.  
FILL-DATA.  
    MOVE CORR INPUT-CARD TO SPEC-ACCT (IA).  
    ADD 1 TO NO-OF-RECS.  
    SET IA UP BY 1.  
BLANK-LINE.  
    MOVE SPACES TO PRINT-LINE.  
WRITE-OUT.  
    WRITE PRINT-LINE BEFORE ADVANCING 1.  
    ADD 1 TO LINE-CT.  
    IF LINE-CT > MAX-LINE  
        ADD 1 TO PAGE-NO  
        MOVE 0 TO LINE-CT  
        PERFORM PRINT-HEADING.  
CLOSE-ROUTINE.  
    WRITE SPEC-ACCT-REC.  
    CLOSE CARD-IN, PRINT-OUT, SPECIAL-ACCT-TABFI.  
PROGRAM-END.  
    STOP RUN.
```

PGMNAME=WUASBFAT
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASBFAT.
AUTHOR. ACT SYSTEMS, INC
 SUITE 200
 807 W MORSE BLVD
 WINTER PARK, FL 32789.
DATE-WRITTEN. JUNE 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM EDITS THE FINANCIAL ACCOUNT DESCRIPTION
 CARDS, FLAGS ANY DISCREPANCIES, AND BUILDS THE FINANCIAL
 ACCOUNT TABLE FILE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
 C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT CARD-IN
 ASSIGN TO UR-2540R-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
 SELECT FILE-SORT
 ASSIGN TO UT-3330-S-FILESORT
 ACCESS MODE IS SEQUENTIAL.
 SELECT SORTED-FILE
 ASSIGN TO UT-3330-S-SORTEDFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT SPECIAL-ACCT-TABFI
 ASSIGN TO UT-3330-S-SATABFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT PRINT-OUT
 ASSIGN TO UR-1403-S-PRINTOUT
 ACCESS MODE IS SEQUENTIAL.
 SELECT FINANCIAL-ACCT-TABFI
 ASSIGN TO DA-3330-I-FATABFI
 ACCESS MODE IS SEQUENTIAL
 RECORD KEY IS FA-NUMBER.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS INPUT-RECS.
01 INPUT-RECS PIC X(80).
01 DATE-CARD.
 05 DATE-IN PIC X(18).
 05 TITLE-IN PIC X(24).
 05 FILLER PIC X(38).

SD FILE-SORT
 RECORD CONTAINS 80 CHARACTERS
 DATA RECORD IS SORT-INPUT.
 01 SORT-INPUT.
 05 SORT-FIRST-10 PIC X(10).
 05 FILLER PIC X(70).
 FD SORTED-FILE
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS INPUT-CARD.
 01 INPUT-CARD.
 05 FA-NO PIC X(10).
 05 OVFLOW1 PIC X.
 05 FA-NAME1 PIC X(20).
 05 OVFLOW2 PIC X.
 05 FA-NAME2 PIC X(20).
 05 OVFLOW3 PIC X.
 05 FN-TITLE PIC XX.
 05 OVFLOW4 PIC X.
 05 GA-TITLE PIC XX.
 05 OVFLOW5 PIC X.
 05 FA-INTERNAL PIC X(4).
 05 OVFLOW6 PIC X.
 05 FILLER PIC X(16).
 FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.
 01 PRINT-LINE PIC X(133).
 FD FINANCIAL-ACCT-TABF1
 BLOCK CONTAINS 8 RECORDS
 RECORD CONTAINS 58 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS FA-REC.
 01 FA-REC.
 05 FA-NUMBER PIC X(10).
 05 FA-NAME1 PIC X(20).
 05 FA-NAME2 PIC X(20).
 05 FN-TITLE PIC XX.
 05 GA-TITLE PIC XX.
 05 FA-INTERNAL PIC X(4).
 FD SPECIAL-ACCT-TABF1
 BLOCK CONTAINS 0 RECORDS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS SPEC-ACCT-REC.
 01 SPEC-ACCT-REC.
 05 RECORD-COUNT.
 10 NO-OF-RECS PIC 9(4).
 10 FILLER PIC X(18).
 05 SPEC-ACCT OCCURS 1 TO 99 TIMES

```

DEPENDING ON NO-OF-RECS
ASCENDING KEY IS SA-NO INDEXED BY IA.
10 SA-NO      PIC XX.
10 SA-NAME    PIC X(20).

WORKING-STORAGE SECTION.
77 IFA      PIC 9 COMP.
77 ISA      PIC 9 COMP.
77 IND      PIC 9 COMP.
77 LAST-FA   PIC X(10) VALUE SPACES.
77 LINE-CT   PIC 9(4) COMP VALUE 0.
77 PAGE-NO   PIC 9(4) COMP VALUE 1.
77 MAX-LINE  PIC 99 COMP VALUE 56.
77 SEARCH-NO PIC XX.

01 COLUMN-HEADING.
    05 FILLER PIC X(21) VALUE SPACES.
    05 FILLER PIC X(9)  VALUE 'ACCOUNT #'.
    05 FILLER PIC X(10) VALUE SPACES.
    05 FILLER PIC X(12) VALUE 'ACCOUNT NAME'.
    05 FILLER PIC X(34) VALUE SPACES.
    05 FILLER PIC X(18) VALUE 'FUND/ACCOUNT TITLE'.
    05 FILLER PIC X(7)  VALUE SPACES.
    05 FILLER PIC X VALUE '#'.
    05 FILLER PIC X(21) VALUE SPACES.

01 DATE-LINE.
    05 FILLER PIC X VALUE SPACE.
    05 FILLER PIC X(6)  VALUE 'DATE: '.
    05 DATE-1  PIC X(18).
    05 FILLER PIC X(40) VALUE SPACES.
    05 FILLER PIC X(3)  VALUE 'FOR'.
    05 FILLER PIC X(53) VALUE SPACES.
    05 FILLER PIC X(5)  VALUE 'PAGE'.
    05 PAGE-1  PIC ZZZ9.
    05 FILLER PIC XX VALUE SPACES.

01 HEADING-LINE.
    05 FILLER PIC X(51) VALUE SPACES.
    05 FILLER PIC X(30) VALUE 'FINANCIAL ACCOUNT DESCRIPTIONS'.
    05 FILLER PIC X(52) VALUE SPACES.

01 TITLE-LINE.
    05 FILLER PIC X(54) VALUE SPACES.
    05 TITLE   PIC X(24).
    05 FILLER PIC X(55) VALUE SPACES.

01 DATA-OUT.
    05 DATA-LINE OCCURS 2 TIMES.
        10 FILLER PIC X(21).
        10 FA-NO.
            15 FA-NO-1 PIC XX.
            15 FILLER PIC X.
            15 FA-NO-2 PIC XX.
            15 FILLER PIC X.
            15 FA-NO-3 PIC X(4).

```

```
10 FILLER PIC X(5).
10 FA-NAME PIC X(20).
10 FILLER PIC X(5).
10 FA-ERR PIC X(20).
10 FILLER PIC X(5).
10 SA-NAME PIC X(20).
10 FILLER PIC X(5).
10 SA-NO PIC XX.
10 FILLER PIC X(5).
10 SA-ERR PIC X(10).
10 FILLER PIC X(5).

PROCEDURE DIVISION.

OPEN-FILES.
    OPEN INPUT CARD-IN, SPECIAL-ACCT-TABFI.
    OPEN OUTPUT PRINT-OUT, FINANCIAL-ACCT-TABFI.

READ-CARDIN.
    READ CARD-IN AT END GO TO CLOSE-ROUTINE.

FILL-HEADING.
    MOVE DATE-IN TO DATE-1.
    MOVE TITLE-IN TO TITLE.

SORT-ROUTINE.
    OPEN OUTPUT SORTED-FILE.
    SORT FILE-SORT ON ASCENDING KEY
        SORT-FIRST-1C
    INPUT PROCEDURE READ-SORT-INPUT THRU
        READ-SORT-INPUT-END
    OUTPUT PROCEDURE WRITE-SORT-OUTPUT THRU
        WRITE-SORT-OUTPUT-END.

CLOSE-SORT-FILES.
    CLOSE SORTED-FILE, CARD-IN.
    OPEN INPUT SORTED-FILE.

PRINT-HEADING.
    PERFORM BLANK-LINE.
    WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
    MOVE HEADING-LINE TO PRINT-LINE.
    PERFORM WRITE-OUT.
    MOVE PAGE-NO TO PAGE-1.
    MOVE DATE-LINE TO PRINT-LINE.
    PERFORM WRITE-OUT.
    MOVE TITLE-LINE TO PRINT-LINE.
    PERFORM WRITE-OUT.
    PERFORM BLANK-LINE THRU WRITE-OUT.
    MOVE COLUMN-HEADING TO PRINT-LINE.
    PERFORM WRITE-OUT.
    PERFORM BLANK-LINE THRU WRITE-OUT.

READ-SPECIAL-ACCTS.
    READ SPECIAL-ACCT-TABFI AT END
        MOVE 'SPECIAL-ACCT-TABFI CONTAINS NO DATA' TO PRINT-LINE
        PERFORM WRITE-OUT
        GO TO CLOSE-ROUTINE.
```

```

START-EDIT.
  READ SORTED-FILE AT END GO TO CLOSE-ROUTINE.
  MOVE 1 TO IFA, ISA.
  MOVE SPACES TO DATA-OUT.

CHECK-ACCT-NO.
  IF FA-NO OF INPUT-CARD = SPACES
    MOVE 'BLANK ACCOUNT #' TO FA-ERR (IFA)
    MOVE 2 TO IFA.
  IF FA-NO OF INPUT-CARD NOT > LAST-FA
    MOVE 'SEQUENCE ERROR' TO FA-ERR (IFA)
    MOVE 2 TO IFA.
  MOVE FA-NO OF INPUT-CARD TO LAST-FA, FA-NO OF DATA-LINE (1),
    FA-NUMBER.

CHECK-ACCT-NAME.
  IF FA-NAME1 OF INPUT-CARD = SPACES AND
    FA-NAME2 OF INPUT-CARD = SPACES
    MOVE 'NL TEXT' TO FA-ERR (IFA)
    MOVE 2 TO IFA.
  MOVE FA-NAME1 OF INPUT-CARD TO FA-NAME (1).
  MOVE FA-NAME2 OF INPUT-CARD TO FA-NAME (2).

CHECK-OVFLOW.
  IF OVFLOW1 NOT = SPACE OR OVFLOW2 NOT = SPACE OR
    OVFLOW3 NOT = SPACE OR OVFLOW4 NOT = SPACE OR
    OVFLOW5 NOT = SPACE OR OVFLOW6 NOT = SPACE
    MOVE 'FIELD OVERFUN' TO FA-ERR (IFA)
    MOVE 2 TO IFA.

CHECK-INTERNAL-ACCT.
  EXAMINE FA-INTERNAL OF INPUT-CARD REPLACING ALL ' ' BY 0.
  IF FA-INTERNAL OF INPUT-CARD NOT NUMERIC
    MOVE 'NOT NUMERIC' TO FA-ERR (IFA)
    MOVE 2 TO IFA.
  MOVE FA-INTERNAL OF INPUT-CARD TO FA-NO-3 (2).

SEARCH-SPEC-ACCTS.
  MOVE 1 TO IND.
  MOVE FN-TITLE OF INPUT-CARD TO SEARCH-NO, FA-NO-1 (2).
  PERFORM SEARCH-ROUTINE.
  MOVE 2 TO IND.
  MOVE GA-TITLE OF INPUT-CARD TO SEARCH-NO, FA-NO-2 (2).
  PERFORM SEARCH-ROUTINE.

CHECK-ERRORS.
  IF IFA < 2 AND ISA < 2
    PERFORM FILL-DATA.
  MOVE 1 TO IND.
  PERFORM WRITE-DATA 2 TIMES.
  PERFORM BLANK-LINE THRU WRITE-DUT.
  PERFORM PAGE-OVFLOW.
  GO TO START-EDIT.

SEARCH-ROUTINE.
  SET IA TO 1.
  SEARCH ALL SPEC-ACCT AT END

```

```

MOVE 'NO TITLE FOUND' TO SA-NAME OF DATA-LINE (IND)
MOVE 2 TO ISA
MOVE ALL '*' TO SA-ERR (IND)
MOVE SEARCH-NU TO SA-NO OF DATA-LINE (IND)
WHEN SA-NO OF SPEC-ACCT (IA) = SEARCH-NU
    MOVE CORR SPEC-ACCT (IA) TO DATA-LINE (IND).

WRITE-DATA.
    MOVE DATA-LINE (IND) TO PRINT-LINE.
    PERFORM WRITE-OUT.
    ADD 1 TO IND.

FILL-DATA.
    MOVE CORR INPUT-CARD TO FA-REC.
    WRITE FA-REC INVALID KEY
        MOVE 'INVALID KEY' TO FA-ERR (IFA).

BLANK-LINE.
    MOVE SPACES TO PRINT-LINE.

WRITE-OUT.
    WRITE PRINT-LINE BEFORE ADVANCING 1.
    ADD 1 TO LINE-CT.

PAGE-OVFLOW.
    IF LINE-CT > MAX-LINE
        ADD 1 TO PAGE-NU
        MOVE 0 TO LINE-CT
        PERFORM PRINT-HEADING.

*
*      INPUT AND OUTPUT SORT ROUTINES
*

READ-SORT-INPUT.
    READ CARD-IN AT END
        GO TO READ-SORT-INPUT-END.
    RELEASE SORT-INPUT FROM INPUT-RECS.
    GO TO READ-SORT-INPUT.

READ-SORT-INPUT-END.
    EXIT.

WRITE-SORT-OUTPUT.
    RETURN FILE-SORT RECORD INTO INPUT-CARD AT END
        GO TO WRITE-SORT-OUTPUT-END.
    WRITE INPUT-CARD.
    GO TO WRITE-SORT-OUTPUT.

WRITE-SORT-OUTPUT-END.
    EXIT.

*
CLOSE-ROUTINE.
    CLOSE CARD-IN, PRINT-OUT, SPECIAL-ACCT-TABFI, SORTED-FILE,
        FINANCIAL-ACCT-TABFI.

PROGRAM-END.
    STOP RUN.

```

```
PGMNAME=WUASBCRF
      DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASBCRF.
AUTHOR. ACT SYSTEMS, INC
      SUITE 200
      807 W MORSE BLVD
      WINTER PARK, FL 32789.
DATE-WRITTEN. SEPTEMBER 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM BUILDS A CROSS REFERENCE FILE USED TO BUILD
      THE COST MATRIX FILE AND FOR COST ALLOCATION.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
      CO1 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
      SELECT CARD-IN
          ASSIGN TO UR-2540R-S-CARDIN
          ACCESS MODE IS SEQUENTIAL.
      SELECT FILE-SORT
          ASSIGN TO UT-3330-S-FILESORT
          ACCESS MODE IS SEQUENTIAL.
      SELECT SORTED-FILE
          ASSIGN TO UT-333-S-SORTEDFI
          ACCESS MODE IS SEQUENTIAL.
      SELECT CROSS-REF-FILE
          ASSIGN TO DA-3330-I-CRTABFI
          ACCESS MODE IS SEQUENTIAL
          RECORD KEY IS ACCT-NO.
      SELECT PRINT-OUT
          ASSIGN TO UR-1403-S-PRINTOUT
          ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
      RECORD CONTAINS 80 CHARACTERS
      LABEL RECORDS ARE OMITTED
      DATA RECORDS ARE INPUT-CARD, DATE-CARD.
01 INPUT-CARD.
      05 CODE-NO      PIC X.
      05 FILLER       PIC X(79).
01 DATE-CARD.
      05 DATE-IN     PIC X(18).
      05 TITLE-IN    PIC X(24).
      05 FILLER       PIC X(38).
SD FILE-SORT
```

RECORD CONTAINS 80 CHARACTERS
 DATA RECORD IS SORT-INPUT.
 01 SORT-INPUT.
 05 SORT-CODE PIC X.
 05 FILLER PIC X.
 05 SORT-FA-NO PIC X(10).
 05 FILLER PIC X(68).
 FD SORTED-FILE
 RECORD CONTAINS 80 CHARACTERS
 BLOCK CONTAINS 5 RECORDS
 LABEL RECORDS ARE STANDARD
 DATA RECORDS ARE INPUT-REC, INPUT-1, INPUT-2.
 01 INPUT-REC.
 05 FILLER PIC X(80).
 01 INPUT-1.
 05 CODE-NO PIC X.
 05 FILLER PIC X.
 05 FA-NO PIC X(10).
 05 C-ALLOC PIC X.
 05 FILLER PIC X.
 05 C-CAT-BLOCK.
 10 C-CAT OCCURS 2 TIMES PIC XX.
 05 FILLER PIC XX.
 05 C-CNTR-BLK OCCURS 7 TIMES.
 10 C-CNTR PIC X(4).
 10 MATCH-NO PIC X(4).
 05 FILLER PIC X(4).
 01 INPUT-2.
 05 CODE-NO PIC X.
 05 FILLER PIC X.
 05 FA-NO PIC X(10).
 05 C-CNTR-BLK OCCURS 8 TIMES.
 10 C-CNTR PIC X(4).
 10 MATCH-NO PIC X(4).
 FD CROSS-REF-FILE
 RECORD CONTAINS 265 CHARACTERS
 BLOCK CONTAINS 4 RECORDS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS CR-REC.
 01 CR-REC.
 05 ACCT-NO PIC X(10).
 05 C-ALLOC PIC X.
 05 C-CATS.
 10 C-CAT OCCURS 2 TIMES PIC XX.
 05 ND-CC PIC 99.
 05 C-CNTR-BLK OCCURS 31 TIMES.
 10 C-CNTR PIC X(4).
 10 MATCH-NO PIC X(4).
 FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS

LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.

01 PRINT-LINE PIC X(133).

WORKING-STORAGE SECTION.

77 LINE-CT PIC 9(4) COMP VALUE 0.
77 PAGE-NO PIC 9(4) COMP VALUE 1.
77 MAX-LINE PIC 99 COMP VALUE 56.
77 I PIC 9 COMP.
77 IC PIC 99 COMP.
77 IR PIC 9 COMP.
77 ERR-HOLD PIC X.
77 J PIC 9 COMP.
77 IE PIC 9 COMP.
77 CODE-HOLD PIC 9 COMP.

01 ACCT-COUNTERS.

05 STL-ACCTS PIC 9(6) COMP VALUE 0.
05 TTL-ACCTS PIC 9(6) COMP VALUE 0.
05 STL-GOOD PIC 9(6) COMP VALUE 0.
05 TTL-GOOD PIC 9(6) COMP VALUE 0.

01 ACCT-LINE.

05 FILLER PIC X VALUE SPACE.
05 TEXT-1 PIC X(4) VALUE 'PAGE'.
05 FILLER PIC X(11) VALUE ' TOTALS '.
05 FILLER PIC X(4) VALUE 'ALL '.
05 ALL-ACCTS PIC Z(5)9.
05 FILLER PIC XX VALUE SPACES.
05 FILLER PIC X(5) VALUE 'GOOD '.
05 GOOD-ACCTS PIC Z(5)9.
05 FILLER PIC X(94) VALUE SPACES.

01 SORT-ERR-LINE.

05 FILLER PIC X VALUE SPACE.
05 CARD-FLD PIC X(80).
05 FILLER PIC X(5) VALUE SPACES.
05 FILLER PIC X(15) VALUE 'SORT CODE ERROR'.
05 FILLER PIC X(32) VALUE SPACES.

01 HEADING-LINE.

05 FILLER PIC X(54) VALUE SPACES.
05 FILLER PIC X(26) VALUE 'CROSS REFERENCE FILE INPUT'.
05 FILLER PIC X(53) VALUE SPACES.

01 DATE-LINE.

05 FILLER PIC X VALUE SPACE.
05 FILLER PIC X(6) VALUE 'DATE: '.
05 DATE-1 PIC X(18).
05 FILLER PIC X(40) VALUE SPACES.
05 FILLER PIC X(3) VALUE 'FOR'.
05 FILLER PIC X(53) VALUE SPACES.
05 FILLER PIC X(4) VALUE 'PAGE'.
05 PAGE-1 PIC ZZZ9.
05 FILLER PIC X(4) VALUE SPACES.

01 TITLE-LINE.

```

05 FILLER PIC X(54) VALUE SPACES.
05 TITLE PIC X(24).
05 FILLER PIC X(55) VALUE SPACES.
01 COL-HEADING-1.
05 FILLER PIC X VALUE SPACE.
05 FILLER PIC X(30) VALUE 'FINANCIAL CATEGORY          COST'.
05 FILLER PIC X(26) VALUE '                      COST / MATCH'.
05 FILLER PIC X(76) VALUE SPACES.
01 COL-HEADING-2.
05 FILLER PIC XX VALUE SPACES.
05 FILLER PIC X(22) VALUE 'ACCOUNT NON-L LABOR '.
05 FILLER PIC X(26) VALUE 'ALLOCATION # CENTERS /'.
05 FILLER PIC X(6) VALUE ' CODE'.
05 FILLER PIC X(71) VALUE SPACES.
05 FILLER PIC X(6) VALUE 'ERRORS'.
01 DATA-OUT.
05 DATA-LINE OCCURS 4 TIMES.
10 FILLER PIC X.
10 FA-NO PIC X(10).
10 C-CAT-BLK OCCURS 2 TIMES.
15 FILLER PIC XX.
15 C-CAT PIC XX.
10 FILLER PIC X(9).
10 C-ALLOC PIC X.
10 FILLER PIC X(7).
10 NO-CC PIC Z9.
10 FILLER PIC XX.
10 CC-MC-BLK OCCURS 8 TIMES.
15 FILLER PIC XX.
15 CC-ND PIC X(4).
15 DELIM PIC X.
15 MATCH PIC X(4).
10 FILLER PIC X.
10 ERRS.
15 ERR-NO OCCURS 4 TIMES PIC X.

```

PROCEDURE DIVISION.

OPEN-FILES.

OPEN INPUT CARD-IN.

OPEN OUTPUT PRINT-OUT, CROSS-REF-FILE, SORTED-FILE.

READ-CARDIN.

READ CARD-IN AT END GO TO CLOSE-FILES.

FILL-HEADING.

MOVE DATE-IN TO DATE-1.

MOVE TITLE-IN TO TITLE.

SORT-ROUTINE.

SORT FILE-SORT ON ASCENDING KEY

SORT-FA-NO

SORT-CODE

INPUT PROCEDURE READ-SORT-INPUT THRU

READ-SORT-INPUT-END

```
OUTPUT PROCEDURE WRITE-SORT-OUTPUT THRU
    WRITE-SORT-OUTPUT-END.
CLOSE-SORT-FILES.
    CLOSE SORTED-FILE, CARD-IN.
    OPEN INPUT SORTED-FILE.
PRINT-HEADING.
    PERFORM BLANK-LINE.
    WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
    MOVE HEADING-LINE TO PRINT-LINE.
    PERFORM WRITE-OUT.
    MOVE PAGE-NO TO PAGE-1.
    MOVE DATE-LINE TO PRINT-LINE.
    PERFORM WRITE-OUT.
    MOVE TITLE-LINE TO PRINT-LINE.
    PERFORM WRITE-OUT.
    PERFORM BLANK-LINE THRU WRITE-OUT.
    MOVE COL-HEADING-1 TO PRINT-LINE.
    PERFORM WRITE-OUT.
    MOVE COL-HEADING-2 TO PRINT-LINE.
    PERFORM WRITE-OUT.
    PERFORM BLANK-LINE THRU WRITE-OUT.
START-EDIT.
    MOVE SPACES TO DATA-OUT, CR-REC.
    MOVE O TO CODE-HOLD.
READ-SORTED-FILE.
    MOVE O TO IE.
    READ SORTED-FILE AT END GO TO CLOSE-ROUTINE.
    MOVE CODE-NO OF INPUT-1 TO IR.
    IF IR NOT > CODE-HOLD
        PERFORM WRITE-DISK
        PERFORM WRITE-DATA VARYING I FROM 1 BY 1
            UNTIL I > 4 OR DATA-LINE (I) = SPACES
        PERFORM BLANK-LINE THRU CHECK-OVFLOW
        PERFORM START-EDIT.
    MOVE IR TO CODE-HOLD.
CHECK-INPUTS.
    IF CODE-NO OF INPUT-1 = '1'
        PERFORM CHECK-INPUT-1 THRU CHECK-INPUT-1-END
    ELSE PERFORM CHECK-INPUT-2 THRU CHECK-INPUT-2-END.
    GO TO READ-SORTED-FILE.
CHECK-INPUT-1.
    PERFORM CHECK-FA-NO THRU CHECK-C-ALLOC.
    MOVE FA-NO OF INPUT-1 TO FA-NO OF DATA-LINE (1), ACCT-NO.
    MOVE O TO IC.
    MOVE 2 TO I.
    PERFORM CHECK-C-CNTRS THRU CHECK-C-CNTRS-END.
CHECK-INPUT-1-END.
    EXIT.
CHECK-INPUT-2.
    PERFORM CHECK-FA-NO.
```

```

PERFORM CHECK-FA-MATCH.
MOVE 1 TO I.
PERFORM CHECK-C-CNTRS THRU CHECK-C-CNTRS-END.
CHECK-INPUT-2-END.
EXIT.
CHECK-FA-NO.
IF FA-NO OF INPUT-1 = SPACES
  MOVE '1' TO ERR-HOLD
  PERFORM ERROR-ROUTINE.
CHECK-C-CAT.
PERFORM FILL-C-CAT VARYING I FRUM 1 BY 1 UNTIL I > 2
IF C-CAT-BLOCK OF INPUT-1 = SPACES
  MOVE '2' TO ERR-HOLD
  PERFORM ERROR-ROUTINE
  GO TO CHECK-C-CAT-END.
EXAMINE C-CAT-BLOCK OF INPUT-1 REPLACING ALL ' ' BY '0'.
IF C-CAT-BLOCK OF INPUT-1 NOT NUMERIC
  MOVE '2' TO ERR-HOLD
  PERFORM ERROR-ROUTINE.
CHECK-C-CAT-END.
EXIT.
CHECK-C-ALLOC.
IF C-ALLOC OF INPUT-1 = SPACE
  MOVE '3' TO ERR-HOLD
  PERFORM ERROR-ROUTINE.
MOVE C-ALLOC OF INPUT-1 TO C-ALLOC OF DATA-LINE (1),
  C-ALLOC OF CR-REC.
FILL-C-CAT.
MOVE C-CAT OF C-CAT-BLOCK (I) TU C-CAT OF C-CAT-BLK (1, I),
  C-CAT OF C-CATS (I).
CHECK-C-CNTRS.
IF C-CNTR-BLK OF INPUT-2 (I) = SPACES
  GO TO CHECK-C-CNTRS-RCHK.
IF C-CNTR OF INPUT-2 (I) = SPACES OR
  MATCH-NO OF INPUT-2 (I) = SPACES
    MOVE '4' TO ERR-HOLD
    PERFORM ERROR-ROUTINE
    GO TO CHECK-C-CNTR-CONT.
IF C-CNTR OF INPUT-2 (I) NOT NUMERIC OR
  MATCH-NO OF INPUT-2 (I) NOT NUMERIC
    MOVE '4' TO ERR-HOLD
    PERFORM ERROR-ROUTINE.
CHECK-C-CNTR-CONT.
ADD 1 TO IC.
IF IC > 31 MOVE 'V' TO ERR-HOLD
  PERFORM ERROR-ROUTINE
  GO TO CHECK-C-CNTRS-END.
IF IR = 1 THEN MOVE IC TO J
ELSE MOVE I TO J.
MOVE C-CNTR OF INPUT-2 (I) TO CC-NO (IR, J),

```

```

C-CNTR OF CR-REC (IC).
MOVE MATCH-NO OF INPUT-2 (I) TO MATCH (IR, J),
      MATCH-NO OF CR-REC (IC).
MOVE IC TO NO-CC OF DATA-LINE (1), NO-CC OF CR-REC.
MOVE '/' TO DELIM (IR, J).
MOVE IC TO NO-CC OF DATA-LINE (1), NO-CC OF CR-REC.
ADD 1 TO I
IF I NOT > 8 GO TO CHECK-C-CNTRS.
GO TO CHECK-C-CNTRS-END.
CHECK-C-CNTRS-RCHK.
IF IR = 1 AND I = 2 OR IR > 1 AND I = 1
    MOVE '4' TO ERR-HOLD
    PERFORM ERROR-ROUTINE.
CHECK-C-CNTRS-END.
EXIT.
CHECK-FA-MATCH.
IF FA-NO OF INPUT-2 NOT = FA-NO OF DATA-LINE (1)
    MOVE '1' TO ERR-HOLD
    PERFORM ERROR-ROUTINE
    MOVE FA-NO OF INPUT-2 TO FA-NO OF DATA-LINE (2).
ERROR-ROUTINE.
IF IE NOT > 3
    ADD 1 TO IE
ELSE MOVE 1 TO IE.
IF IE = 1 MOVE ALL '*' TO ERRS OF DATA-LINE (IR).
MOVE ERR-HOLD TO ERR-NO (IR, IE).
WRITE-DATA.
MOVE DATA-LINE (I) TO PRINT-LINE.
PERFORM WRITE-OUT.
WRITE-DISK.
ADD NO-CC OF CR-REC TO STL-ACCTS, TTL-ACCTS.
IF ERRS OF DATA-LINE (1) = SPACES AND
    ERRS OF DATA-LINE (2) = SPACES AND
    ERRS OF DATA-LINE (3) = SPACES AND
    ERRS OF DATA-LINE (4) = SPACES
        ADD NO-CC OF CR-REC TO STL-GOOD, TTL-GOOD
        WRITE CR-REC INVALID KEY
        SUBTRACT NO-CC OF CR-REC FROM STL-GOOD, TTL-GOOD
        MOVE ALL '*' TO ERRS OF DATA-LINE (1)
        MOVE 'INVLD KEY' TO CC-MC-BLK (1, 8).
BLANK-LINE.
MOVE SPACES TO PRINT-LINE.
WRITE-OUT.
WRITE PRINT-LINE BEFORE ADVANCING 1.
ADD 1 TO LINE-CT.
CHECK-OVFLOW.
IF LINE-CT > MAX-LINE
    PERFORM FILL-STL-ACCT-LINE
    PERFORM PRINT-ACCT-LINE
    MOVE 0 TO STL-ACCTS, STL-GOOD

```

```

        ADD 1 TO PAGE-NO
        MOVE 0 TO LINE-CT
        PERFORM PRINT-HEADING.

READ-SORT-INPUT.
    READ CARD-IN AT END
        GO TO READ-SORT-INPUT-END.
    IF CODE-NO OF INPUT-CARD = '1' OR '2' OR '3' OR '4'
        RELEASE SORT-INPUT FROM INPUT-CARD
    ELSE MOVE INPUT-CARD TO CARD-FLD OF SORT-ERR-LINE
        MOVE PRINT-LINE TO SORT-ERR-LINE
        PERFORM WRITE-OUT.
    GO TO READ-SORT-INPUT.

READ-SORT-INPUT-END.
    EXIT.

WRITE-SORT-OUTPUT.
    RETURN FILE-SORT RECORD INTO INPUT-REC AT END
    GO TO WRITE-SORT-OUTPUT-END.
    WRITE INPUT-REC.
    GO TO WRITE-SORT-OUTPUT.

WRITE-SORT-OUTPUT-END.
    EXIT.

FILL-STL-ACCT-LINE.
    MOVE STL-ACCTS TO ALL-ACCTS.
    MOVE STL-GOOD TO GOOD-ACCTS.

FILL-TTL-ACCT-LINE.
    MOVE 'CUMM' TO TEXT-1.
    MOVE TTL-ACCTS TO ALL-ACCTS.
    MOVE TTL-GOOD TO GOOD-ACCTS.

PRINT-ACCT-LINE.
    PERFORM BLANK-LINE THRU WRITE-OUT.
    MOVE ACCT-LINE TO PRINT-LINE.
    PERFORM WRITE-OUT.

CLOSE-ROUTINE.
    PERFORM WRITE-DISK.
    IF DATA-OUT NOT = SPACES
        PERFORM WRITE-DATA VARYING I FROM 1 BY 1
            UNTIL I > 4 OR DATA-LINE (I) = SPACES.
    PERFORM FILL-STL-ACCT-LINE.
    PERFORM PRINT-ACCT-LINE.
    PERFORM FILL-TTL-ACCT-LINE THRU PRINT-ACCT-LINE.

CLOSE-FILES.
    CLOSE PRINT-OUT, SORTED-FILE, CROSS-REF-FILE.

PROGRAM-END.
    STOP RUN.

```

PGMNAME=WUASBCMF
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASBCMF.
AUTHOR. ACT SYSTEMS, INC
 SUITE 200
 807 W MORSE BLVD
 WINTER PARK, FL 32789.
DATE-WRITTEN. AUGUST 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM EDITS THE CUST MATRIX ACCOUNT CARDS, FLAGS
ANY DISCREPANCIES, AND BUILDS THE COST ACCOUNT MATRIX
FILE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370-168.
OBJECT-COMPUTER. IBM-370-168.
SPECIAL-NAMES.
 C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT CARD-IN
 ASSIGN TO UR-2540K-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
 SELECT CROSS-REF-FILE
 ASSIGN TO DA-3330-I-CRTABFI
 ACCESS MODE IS SEQUENTIAL
 RECORD KEY IS ACCT-NO.
 SELECT COST-MATRIX-FILE
 ASSIGN TO DA-3330-I-CMFILE
 ACCESS MODE IS SEQUENTIAL
 RECORD KEY IS CM-NUMBER.
 SELECT FINANCIAL-ACCT-TABFI
 ASSIGN TO DA-3330-I-FATABFI
 ACCESS MODE IS RANDOM
 NOMINAL KEY IS FA-NOMINAL
 RECORD KEY IS FA-NUMBER.
 SELECT COST-CNTR-TABFI
 ASSIGN TO DA-3330-I-CCTABFI
 ACCESS MODE IS RANDOM
 NOMINAL KEY IS CC-NOMINAL
 RECORD KEY IS CC-NUMBER.
 SELECT SPECIAL-ACCT-TABFI
 ASSIGN TO UT-3330-S-SATABFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT FILE-SORT
 ASSIGN TO UT-3330-S-FILESURT
 ACCESS MODE IS SEQUENTIAL.
 SELECT SORTED-FILE
 ASSIGN TO UT-3330-S-SORTEDFI
 ACCESS MODE IS SEQUENTIAL.

```
SELECT PRINT-OUT
      ASSIGN TO UR-1403-S-PRINTOUT
      ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
      RECORD CONTAINS 80 CHARACTERS
      LABEL RECORDS ARE OMITTED
      DATA RECORD IS DATE-CARD.
01 DATE-CARD.
      05 DATE-IN      PIC X(18).
      05 TITLE-IN     PIC X(24).
      05 FILLER       PIC X(38).
FD CROSS-REF-FILE
      RECORD CONTAINS 265 CHARACTERS
      BLOCK CONTAINS 4 RECORDS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS CR-REC.
01 CR-REC.
      05 ACCT-NO      PIC X(10).
      05 C-ALLOC      PIC X.
      05 C-CATS.
          10 C-CAT OCCURS 2 TIMES PIC XX.
      05 NO-CC        PIC 99.
      05 C-CNTR-BLK  OCCURS 31 TIMES.
          10 C-CNTR      PIC X(4).
          10 MATCH-NO    PIC X(4).
FD FINANCIAL-ACCT-TABFI
      BLOCK CONTAINS 8 RECORDS
      RECORD CONTAINS 58 CHARACTERS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS FA-REC.
01 FA-REC.
      05 FA-NUMBER    PIC X(10).
      05 FA-NAME      PIC X(20) OCCURS 2 TIMES.
      05 FILLER       PIC X(8).
FD COST-CNTR-TABFI
      BLOCK CONTAINS 8 RECORDS
      RECORD CONTAINS 30 CHARACTERS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS CC-REC.
01 CC-REC.
      05 CC-NUMBER    PIC X(4).
      05 CC-NAME      PIC X(13) OCCURS 2 TIMES.
FD SPECIAL-ACCT-TABFI
      BLOCK CONTAINS 0 RECORDS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS SPEC-ACCT-REC.
01 SPEC-ACCT-REC.
      05 RECORD-COUNT.
```

```

        10 NO-OF-RECS PIC 9(4).
        10 FILLER      PIC X(18).
05 SPEC-ACCT OCCURS 1 TO 99 TIMES
    DEPENDING ON NO-OF-RECS
    ASCENDING KEY IS SA-NO INDEXED BY IA.
        10 SA-NO      PIC XX.
        10 SA-NAME    PIC X(20).

SD FILE-SORT
RECORD CONTAINS 21 CHARACTERS
DATA RECORD IS SORT-INPUT.

01 SORT-INPUT.
    05 SORT-FIRST-16 PIC X(16).
    05 FILLER      PIC X(5).

FD SORTED-FILE
RECORD CONTAINS 21 CHARACTERS
BLOCK CONTAINS 10 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-REC.

01 INPUT-REC.
    05 FA-NO.
        10 FA-NO-1    PIC X.
        10 FA-NO-2    PIC X(9).
    05 CC-NO      PIC X(4).
    05 SA-NO      PIC XX.
    05 MATCH-NO    PIC X(4).
    05 C-ALLOC     PIC X.

FD COST-MATRIX-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CM-REC.

01 CM-REC.
    05 MATCH-NO    PIC X(4).
    05 CM-NUMBER.
        10 FA-NO     PIC X(10).
        10 CC-NO     PIC X(4).
        10 SA-NO     PIC XX.
    05 AMTS       PIC 9(8)V99 OCCURS 6 TIMES.

FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.

01 PRINT-LINE PIC X(133).

WORKING-STORAGE SECTION.
77 LINE-CT      PIC 9(4) COMP VALUE 0.
77 PAGE-NO     PIC 9(4) COMP VALUE 1.
77 MAX-LINE    PIC 99 COMP VALUE 56.
77 IB          PIC 9 COMP.
77 IC          PIC 99 COMP.
77 NF-ERR      PIC X(20) VALUE 'RECORD NOT FOUND' .

```

```

77 FA-NOMINAL PIC X(10).
77 CC-NOMINAL PIC X(4).
01 ACCT-COUNTERS.
    05 STL-ACCTS    PIC 9(6) COMP VALUE 0.
    05 TTL-ACCTS    PIC 9(6) COMP VALUE 0.
    05 STL-GOOD     PIC 9(6) COMP VALUE 0.
    05 TTL-GOOD     PIC 9(6) COMP VALUE 0.
01 ACCT-LINE.
    05 FILLER PIC X VALUE SPACE.
    05 TEXT-1 PIC X(4) VALUE 'PAGE'.
    05 FILLER PIC X(11) VALUE ' TOTALS '.
    05 FILLER PIC X(4) VALUE 'ALL '.
    05 ALL-ACCTS    PIC Z(5)9.
    05 FILLER PIC XX VALUE SPACES.
    05 FILLER PIC X(5) VALUE 'GOOD '.
    05 GOOD-ACCTS   PIC Z(5)9.
    05 FILLER PIC X(94) VALUE SPACES.
01 DATA-BLK.
    05 FA-NO      PIC X(10).
    05 CC-NO      PIC X(4).
    05 SA-NO      PIC XX.
    05 MATCH-NU    PIC X(4).
    05 C-ALLOC     PIC X.
01 HEADING-LINE.
    05 FILLER PIC X(55) VALUE SPACES.
    05 FILLER PIC X(22) VALUE 'COST MATRIX FILE INPUT'.
    05 FILLER PIC X(56) VALUE SPACES.
01 DATE-LINE.
    05 FILLER PIC X VALUE SPACE.
    05 FILLER PIC X(6) VALUE 'DATE: '.
    05 DATE-1     PIC X(18).
    05 FILLER PIC X(40) VALUE SPACES.
    05 FILLER PIC X(3) VALUE 'FOR'.
    05 FILLER PIC X(53) VALUE SPACES.
    05 FILLER PIC X(4) VALUE 'PAGE'.
    05 PAGE-1     PIC ZZZ9.
    05 FILLER PIC X(4) VALUE SPACES.
01 TITLE-LINE.
    05 FILLER PIC X(54) VALUE SPACES.
    05 TITLE     PIC X(24).
    05 FILLER PIC X(55) VALUE SPACES.
01 COL-HEADING-1.
    05 FILLER PIC X(7) VALUE SPACES.
    05 FILLER PIC X(17) VALUE 'FINANCIAL ACCOUNT'.
    05 FILLER PIC X(17) VALUE SPACES.
    05 FILLER PIC X(11) VALUE 'COST CENTER'.
    05 FILLER PIC X(21) VALUE SPACES.
    05 FILLER PIC X(13) VALUE 'COST CATEGORY'.
    05 FILLER PIC X(10) VALUE SPACES.
    05 FILLER PIC X(15) VALUE 'COST ALLOCATION'.

```

```

05 FILLER PIC X(4) VALUE SPACES.
05 FILLER PIC X(10) VALUE 'MATCH CODE'.
05 FILLER PIC X(3) VALUE SPACES.
05 FILLER PIC X(5) VALUE 'ERROR'.

01 COL-HEADING-2.
05 FILLER PIC X(6) VALUE SPACES.
05 FILLER PIC X VALUE '#'.
05 FILLER PIC X(14) VALUE SPACES.
05 FILLER PIC X(4) VALUE 'NAME'.
05 FILLER PIC X(15) VALUE SPACES.
05 FILLER PIC X VALUE '#'.
05 FILLER PIC X(11) VALUE SPACES.
05 FILLER PIC X(4) VALUE 'NAME'.
05 FILLER PIC X(16) VALUE SPACES.
05 FILLER PIC X VALUE '#'.
05 FILLER PIC X(10) VALUE SPACES.
05 FILLER PIC X(4) VALUE 'NAME'.
05 FILLER PIC X(14) VALUE SPACES.
05 FILLER PIC X(6) VALUE 'METHOD'.
05 FILLER PIC X(12) VALUE SPACES.
05 FILLER PIC X VALUE '#'.
05 FILLER PIC X(9) VALUE SPACES.
05 FILLER PIC X(4) VALUE 'FLAG'.

01 DATA-OUT.
05 DATA-LINE OCCURS 2 TIMES.
10 FILLER PIC X.
10 FA-NO PIC X(10).
10 FILLER PIC X(3).
10 FA-NAME PIC X(20).
10 FILLER PIC X(5).
10 CC-NO PIC X(4).
10 FILLER PIC X(3).
10 CC-NAME PIC X(20).
10 FILLER PIC X(5).
10 SA-NO PIC XX.
10 FILLER PIC X(3).
10 SA-NAME PIC X(20).
10 FILLER PIC X(7).
10 C-ALLOC PIC X.
10 FILLER PIC X(14).
10 MATCH-NO PIC X(4).
10 FILLER PIC X(7).
10 CM-ERR PIC XX.
10 FILLER PIC XX.

```

PROCEDURE DIVISION.

OPEN-FILES.

```

OPEN INPUT CARD-IN, FINANCIAL-ACCT-TABFI, CROSS-REF-FILE,
COST-CNTR-TABFI, SPECIAL-ACCT-TABFI.
OPEN OUTPUT PRINT-OUT, COST-MATRIX-FILE.

```

READ-CARDIN.

```

        READ CARD-IN AT END GO TO CLOSE-ROUTINE.
FILL-HEADING.
        MOVE DATE-IN TO DATE-1.
        MOVE TITLE-IN TO TITLE.
SORT-ROUTINE.
        OPEN OUTPUT SORTED-FILE.
        SORT FILE-SORT ON ASCENDING KEY
            SORT-FIRST-16
        INPUT PROCEDURE READ-SORT-INPUT THRU
            READ-SORT-INPUT-END
        OUTPUT PROCEDURE WRITE-SORT-OUTPUT THRU
            WRITE-SORT-OUTPUT-END.
CLOSE-SORT-FILES.
        CLOSE SORTED-FILE, CARD-IN.
        OPEN INPUT SORTED-FILE.
PRINT-HEADING.
        PERFORM BLANK-LINE.
        WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
        MOVE HEADING-LINE TO PRINT-LINE.
        PERFORM WRITE-OUT.
        MOVE PAGE-N0 TO PAGE-1.
        MOVE DATE-LINE TO PRINT-LINE.
        PERFORM WRITE-OUT.
        MOVE TITLE-LINE TO PRINT-LINE.
        PERFORM WRITE-OUT.
        PERFORM BLANK-LINE THRU WRITE-OUT.
        MOVE COL-HEADING-1 TO PRINT-LINE.
        PERFORM WRITE-OUT.
        MOVE COL-HEADING-2 TO PRINT-LINE.
        PERFORM WRITE-OUT.
        PERFORM BLANK-LINE THRU WRITE-OUT.
READ-SPECIAL-ACCTS.
        READ SPECIAL-ACCT-TABFI AT END
            MOVE 'SPECIAL-ACCT-TABFI CONTAINS NO DATA' TO PRINT-LINE
            PERFORM WRITE-OUT
            GO TO CLOSE-ROUTINE.
START-EDIT.
        READ SORTED-FILE AT END
            GO TO CLOSE-ROUTINE.
        MOVE SPACES TO DATA-OUT.
FIND-FA-INFO.
        MOVE FA-NO OF INPUT-REC TO FA-NOMINAL,
            FA-NO OF DATA-LINE (1).
        IF FA-NO-1 = '-'
            MOVE 'CONTRA ACCOUNT' TO FA-NAME OF DATA-LINE (1)
            GO TO FIND-FA-INFO-CONT.
        READ FINANCIAL-ACCT-TABFI INVALID KEY
        MOVE NF-ERR TO FA-NAME OF DATA-LINE (1)
        MOVE ALL '*' TO CM-ERR OF DATA-LINE (1)
        GO TO FIND-CC-INFO.

```

```

MOVE FA-NAME OF FA-REC (1) TO FA-NAME OF DATA-LINE (1).
MOVE FA-NAME OF FA-REC (2) TO FA-NAME OF DATA-LINE (2).
FIND-FA-INFO-CONT.
  MOVE FA-NUMBER TO FA-NO OF CM-NUMBER.
FIND-CC-INFO.
  MOVE CC-NO OF INPUT-REC TO CC-NOMINAL,
    CC-NO OF DATA-LINE (1).
READ COST-CNTR-TABFI INVALID KEY
  MOVE NF-ERR TO CC-NAME OF DATA-LINE (1)
  MOVE ALL '*' TO CM-ERR OF DATA-LINE (1)
  GO TO FIND-SA-INFO.
MOVE CC-NUMBER TO CC-NO OF CM-NUMBER.
MOVE CC-NAME OF CC-REC (1) TO CC-NAME OF DATA-LINE (1).
MOVE CC-NAME OF CC-REC (2) TO CC-NAME OF DATA-LINE (2).
FIND-SA-INFO.
  MOVE 1 TO IB.
  PERFORM SEARCH-ROUTINE.
CHECK-COST-ALLOC.
  IF C-ALLOC OF INPUT-REC NOT = SPACE
    MOVE C-ALLOC OF INPUT-REC TO C-ALLOC OF DATA-LINE (1)
  ELSE MOVE ALL '*' TO CM-ERR OF DATA-LINE (1),
    C-ALLOC OF DATA-LINE (1).
CHECK-MATCH-NO.
  IF MATCH-NO OF INPUT-REC NOT = SPACES
    MOVE MATCH-NO OF INPUT-REC TO MATCH-NO OF CM-REC,
      MATCH-NO OF DATA-LINE (1)
  ELSE MOVE ALL '*' TO CM-ERR OF DATA-LINE (1),
    MATCH-NO OF DATA-LINE (1).
CHECK-ERRORS.
  ADD 1 TO STL-ACCTS, TTL-ACCTS.
  IF CM-ERR OF DATA-LINE (1) = SPACES
    MOVE 1 TO IB
    PERFORM ZERO-AMTS 6 TIMES
    PERFORM FILL-DATA.
  MOVE 1 TO IB.
  PERFORM WRITE-DATA 2 TIMES.
  PERFORM BLANK-LINE THRU WRITE-OUT.
  PERFORM PAGE-OVFLOW.
  GO TO START-EDIT.
SEARCH-ROUTINE.
  MOVE SA-NO OF INPUT-REC TO SA-NO OF DATA-LINE (IB).
  SET IA TO 1.
  SEARCH ALL SPEC-ACCT AT END
    MOVE NF-ERR TO SA-NAME OF DATA-LINE (IB)
    MOVE ALL '*' TO CM-ERR OF DATA-LINE (IB)
  WHEN SA-NO OF SPEC-ACCT (IA) = SA-NO OF INPUT-REC
    MOVE SA-NAME OF SPEC-ACCT (IA) TO
      SA-NAME OF DATA-LINE (IB)
    MOVE SA-NO OF SPEC-ACCT (IA) TO SA-NO OF CM-NUMBER.
FILL-DATA.

```

```

ADD 1 TO STL-GOOD, TTL-GOOD.
WRITE CM-REC INVALID KEY
    SUBTRACT 1 FROM STL-GOOD, TTL-GOOD
    MOVE ALL '*' TO CM-ERR OF DATA-LINE (1)
    MOVE 'INVALID KEY' TO FA-NAME OF DATA-LINE (1).

WRITE-DATA.
    IF DATA-LINE (IB) NOT = SPACES
        MOVE DATA-LINE (IB) TO PRINT-LINE
        PERFORM WRITE-OUT.

    ADD 1 TO IB.

BLANK-LINE.
    MOVE SPACES TO PRINT-LINE.

WRITE-OUT.
    WRITE PRINT-LINE BEFORE ADVANCING 1.
    ADD 1 TO LINE-CT.

PAGE-OVFLW.
    IF LINE-CT > MAX-LINE
        PERFORM FILL-STL-ACCT-LINE
        PERFORM PRINT-ACCT-LINE
        MOVE 0 TO STL-ACCTS, STL-GOOD
        ADD 1 TO PAGE-NO
        MOVE 0 TO LINE-CT
        PERFORM PRINT-HEADING.

ZERO-AMTS.
    MOVE ZERO TO AMTS OF CM-REC (IB).
    ADD 1 TO IB.

*
*      INPUT AND OUTPUT SORT ROUTINES
*

READ-SORT-INPUT.
    READ CROSS-REF-FILE AT END
        GO TO READ-SORT-INPUT-END.
    MOVE SPACES TO DATA-BLK.
    MOVE ACCT-NU TO FA-NO OF DATA-BLK.
    MOVE C-ALLOC OF CR-REC TO C-ALLUC OF DATA-BLK.
    MOVE 1 TO IB, IC.
    PERFORM STORE-C-CAT 2 TIMES.
    GO TO READ-SORT-INPUT.

STORE-C-CAT.
    IF C-CAT (IB) NOT = SPACES
        MOVE C-CAT (IB) TO SA-NO OF DATA-BLK
        PERFORM RELEASE-INPUT VARYING IC FROM 1 BY 1
            UNTIL IC > NO-CC.

        ADD 1 TO IB.

RELEASE-INPUT.
    MOVE C-CNTR (IC) TO CC-NO OF DATA-BLK.
    MOVE MATCH-NO OF C-CNTR-BLK (IC) TO MATCH-NO OF DATA-BLK.
    RELEASE SORT-INPUT FROM DATA-BLK.

READ-SORT-INPUT-END.
    EXIT.

```

```
WRITE-SORT-OUTPUT.  
    RETURN FILE-SORT RECORD INTO INPUT-REC AT END  
        GO TO WRITE-SORT-OUTPUT-END.  
    WRITE INPUT-REC.  
    GO TO WRITE-SORT-OUTPUT.  
WRITE-SORT-OUTPUT-END.  
    EXIT.  
*  
FILL-STL-ACCT-LINE.  
    MOVE STL-ACCTS TO ALL-ACCTS.  
    MOVE STL-GOOD TO GOOD-ACCTS.  
FILL-TTL-ACCT-LINE.  
    MOVE 'CUMM' TO TEXT-1.  
    MOVE TTL-ACCTS TO ALL-ACCTS.  
    MOVE TTL-GOOD TO GOOD-ACCTS.  
PRINT-ACCT-LINE.  
    PERFORM BLANK-LINE THRU WRITE-OUT.  
    MOVE ACCT-LINE TO PRINT-LINE.  
    PERFORM WRITE-OUT.  
CLOSE-ROUTINE.  
    PERFORM FILL-STL-ACCT-LINE.  
    PERFORM PRINT-ACCT-LINE.  
    PERFORM FILL-TTL-ACCT-LINE THRU PRINT-ACCT-LINE.  
    CLOSE PRINT-OUT, SPECIAL-ACCT-TABFI, SORTED-FILE,  
        FINANCIAL-ACCT-TABFI, COST-CNTR-TABFI, COST-MATRIX-FILE,  
        CROSS-REF-FILE.  
PROGRAM-END.  
    STOP RUN.
```

```

PGMNAME=WUASBDPT
.SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASBDPT.
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
    807 W MURSE BLVD
    WINTER PARK, FL 32789.
DATE-WRITTEN. JUNE 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM EDITS THE SPECIAL ACCOUNT DESCRIPTION
CARDS, FLAGS ANY DISCREPANCIES, AND BUILDS THE SPECIAL
DEPRECIATION TABLE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT SPECIAL-ACCT-TABFI
        ASSIGN TO UT-3330-S-SATABFI
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE OMITTED
    DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
    05 CUST-CNTR          PIC X(4).
    05 FILLER              PIC X.
    05 DEP-AMOUNT          PIC 9(8)V99.
    05 FILLER              PIC X.
    05 COUNTY-CODE         PIC X.
    05 FILLER              PIC X(63).
01 SPECIAL-TITLES.
    05 SA-NO               PIC XX.
    05 FILLER              PIC X.
    05 SPECIAL-TITLE       PIC X(24).
    05 FILLER              PIC X(53).
01 DATE-CARD.
    05 DATE-IN             PIC X(18).

```

```

05 TITLE-IN      PIC X(24).
05 DEP-ACCT-IN    PIC X(10).
05 FILLER        PIC X(28).

FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.

01 PRINT-LINE      PIC X(133).

FD SPECIAL-ACCT-TABFI
BLOCK CONTAINS 0 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS SPEC-ACCT-REC.

01 SPEC-ACCT-REC.

05 RECORD-COUNT.
10 NO-OF-RECS   PIC 9(4).
10 DEP-ACCT-DUT PIC X(10).
10 FILLER       PIC X(8).

05 SPEC-ACCT OCCURS 1 TO 99 TIMES
DEPENDING ON NO-OF-RECS
ASCENDING KEY IS SA-NU INDEXED BY IA.
10 COST-CNTR    PIC X(4).
10 DEP-AMOUNT   PIC 9(8)V99.
10 COUNTY-CODE  PIC X.
10 FILLER       PIC X(7).

WORKING-STORAGE SECTION.

77 LINE-CT      PIC 9(4) COMP VALUE 0.
77 PAGE-NO      PIC 9(4) COMP VALUE 1.
77 MAX-LINE     PIC 99 COMP VALUE 56.
77 AMT-ACCUM    PIC 9(8)V99 VALUE 0.

01 HEADING-LINE.
05 FILLER       PIC X(52) VALUE SPACES.
05 FILLER       PIC X(28) VALUE 'SPECIAL ACCOUNT DESCRIPTIONS'.
05 FILLER       PIC X(53) VALUE SPACES.

01 DATE-LINE.
05 FILLER       PIC X VALUE SPACE.
05 FILLER       PIC X(6) VALUE 'DATE: '.
05 DATE-1       PIC X(18).
05 FILLER       PIC X(40) VALUE SPACES.
05 FILLER       PIC X(3) VALUE 'FGR'.
05 FILLER       PIC X(53) VALUE SPACES.
05 FILLER       PIC X(5) VALUE 'PAGE '.
05 PAGE-1       PIC ZZZ9.
05 FILLER       PIC XX VALUE SPACES.

01 TITLE-LINE.
05 FILLER       PIC X(54) VALUE SPACES.
05 TITLE        PIC X(24).
05 FILLER       PIC X(55) VALUE SPACES.

01 DATA-OUT.
05 FILLER       PIC X(52).
05 COST-CNTR    PIC X(4).

```

```

05 FILLER          PIC X(8).
05 DEP-AMOUNT      PIC $BZ(7)9.99.
05 FILLER          PIC X(56).

PROCEDURE DIVISION.
OPEN-FILES.
  OPEN INPUT CARD-IN.
  OPEN OUTPUT PRINT-OUT, SPECIAL-ACCT-TABFI.
CLEAR-FIELDS.
  MOVE SPACES TO DATA-OUT.
  MOVE ZERO TO NO-OF-RECS.
  SET IA TO 1.

READ-CARDIN.
  READ CARD-IN AT END GO TO CLOSE-ROUTINE.

FILL-HEADING.
  MOVE DATE-IN TO DATE-1.
  MOVE TITLE-IN TO TITLE.
  MOVE DEP-ACCT-IN TO DEP-ACCT-OUT.

PRINT-HEADING.
  PERFORM BLANK-LINE.
  WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
  MOVE HEADING-LINE TO PRINT-LINE.
  PERFORM WRITE-OUT.
  MOVE PAGE-NO TO PAGE-1.
  MOVE DATE-LINE TO PRINT-LINE.
  PERFORM WRITE-OUT.
  MOVE TITLE-LINE TO PRINT-LINE.
  PERFORM WRITE-OUT.

WRITE-OUT-RECS.
  PERFORM READ-CARDIN.
  IF SA-NO = '***' THEN
    PERFORM BLANK-LINE THRU WRITE-OUT
    MOVE SPECIAL-TITLE TO TITLE
    MOVE TITLE-LINE TO PRINT-LINE
    PERFORM WRITE-OUT
    PERFORM BLANK-LINE THRU WRITE-OUT
    GO TO WRITE-OUT-RECS.

FILL-DATA.
  MOVE CORR INPUT-CARD TO DATA-OUT, SPEC-ACCT (IA).
  MOVE DATA-OUT TO PRINT-LINE.
  PERFORM WRITE-OUT.
  MOVE SPACES TO DATA-OUT.
  ADD DEP-AMOUNT OF INPUT-CARD TO AMT-ACCUM.
  ADD 1 TO NO-OF-RECS.
  SET IA UP BY 1.
  GO TO WRITE-OUT-RECS.

BLANK-LINE.
  MOVE SPACES TO PRINT-LINE.

WRITE-OUT.
  WRITE PRINT-LINE BEFORE ADVANCING 1.
  ADD 1 TO LINE-CT.

```

```
IF LINE-CT > MAX-LINE
    ADD 1 TO PAGE-NO
    MOVE 0 TO LINE-CT
    PERFORM PRINT-HEADING.

CLOSE-ROUTINE.
    WRITE SPEC-ACCT-REC.
    PERFORM BLANK-LINE THRU WRITE-OUT.
    PERFORM WRITE-OUT.
    MOVE AMT-ACCUM TO DEP-AMOUNT OF DATA-OUT.
    MOVE DATA-OUT TO PRINT-LINE.
    PERFORM WRITE-OUT.
    CLOSE CARD-IN, PRINT-OUT, SPECIAL-ACCT-TABFI.

PROGRAM-END.
STOP RUN.
```

PGMNAME=WUASBCDP
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASBCDP.
AUTHOR. ACT SYSTEMS, INC
SUITE 200
807 W MORSE BLVD
WINTER PARK, FL 32789.
DATE-WRITTEN. DECEMBER, 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM READS ONE CARD
AND BUILDS THE CURRENT DEPRECIATION
FILE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-IN
ASSIGN TO UR-2540R-S-CARDIN
ACCESS MODE IS SEQUENTIAL.
SELECT PRINT-OUT
ASSIGN TO UR-1403-S-PRINTUUT
ACCESS MODE IS SEQUENTIAL.
SELECT DEPRECIATION-TABFI
ASSIGN TO UT-3330-S-DPTABFI
ACCESS MODE IS SEQUENTIAL.
SELECT COST-CNTR-TABFI
ASSIGN TO DA-3330-I-CCTABFI
ACCESS MODE IS RANDOM
NOMINAL KEY IS CC-NOMINAL
RECORD KEY IS CC-NUMBER.
SELECT FINANCIAL-COST-ALLOCATION
ASSIGN TO UT-3330-S-FICALFI
ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
05 DATE-IN PIC X(18).
05 TITLE-IN PIC X(24).
05 PERIOD-IN.
10 PERIOD1 PIC X.
10 FILLER PIC X(8).

```

      05 FILLER          PIC X(29).
FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
      05 FILLER          PIC X.
      05 PRINT-DATA       PIC X(132).
FD DEPRECIATION-TABFI
BLOCK CONTAINS 0 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS DEPRECIATION-REC.
01 DEPRECIATION-REC.
      05 RECORD-COUNT.
          10 NO-OF-RECS   PIC 9(4).
          10 ACCT-NO      PIC X(10).
          10 FILLER        PIC X(8).
      05 DEP-REC    OCCURS 1 TO 99 TIMES
          DEPENDING ON NO-OF-RECS
          ASCENDING KEY IS COST-CNTR INDEXED BY CC.
          10 COST-CNTR     PIC X(4).
          10 AMOUNT        PIC 9(8)V99.
          10 COUNTY-CODE   PIC X.
          10 FILLER        PIC X(7).
FD FINANCIAL-CUST-ALLOCATION
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS DISK-OUT.
01 DISK-OUT.
      05 DISK-ACCT-NO.
          10 FILLER        PIC X(9).
          10 DISK-COUNTY-CODE PIC X.
      05 DISK-COST-CNTR   PIC X(4).
      05 DISK-COST-CAT    PIC XX.
      05 DISK-CURRENT-AMT PIC S9(8)V99.
01 FIRST-TWO-RECS.
      05 FILLER          PIC X.
      05 SORT-SEQUENCE   PIC X.
      05 DISK-TITLE.
          10 DISK-DATE      PIC X(18).
          10 FILLER        PIC X(6).
FD COST-CNTR-TABFI
BLOCK CONTAINS 8 RECORDS
RECORD CONTAINS 30 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CC-REC.
01 CC-REC.
      02 CC-NUMBER       PIC X(4).
      02 CC-NAME         PIC X(13) OCCURS 2 TIMES.

```

WORKING-STORAGE SECTION.

```

77 SOURCE-CODE          PIC X VALUE '6'.
77 LINE-CT              PIC 99 COMP VALUE 0.
77 PAGE-CT              PIC 9(4) CLMP VALUE 1.
77 PERIOD               PIC 99 VALUE 12.
77 CC-NOMINAL           PIC X(4).
77 NO-OF-COST-CNTRS    PIC 99 COMP VALUE 0.
77 AMT-HOLD              PIC 9(8)V99.
77 MAX-LINE              PIC 99 COMP VALUE 56.
77 YEARLY-DEP-TOTAL     PIC 9(8)V99 VALUE ZERO.
77 CURRENT-DEP-TOTAL    PIC 9(8)V99 VALUE ZERO.

01 HEAD-1.
    05 FILLER             PIC X(56) VALUE SPACES.
    05 FILLER             PIC X(18) VALUE 'DEPRECIATION TABLE'.
    05 FILLER             PIC X(58) VALUE SPACES.

01 HEAD-2.
    05 FILLER             PIC X(15) VALUE 'PERIOD ENDING: '.
    05 DATE-OUT            PIC X(18).
    05 FILLER             PIC X(31) VALUE SPACES.
    05 FILLER             PIC X(3)  VALUE 'FOR'.
    05 FILLER             PIC X(55) VALUE SPACES.
    05 FILLER             PIC X(6)  VALUE 'PAGE: '.
    05 PAGE-OUT            PIC Z(4).

01 HEAD-3.
    05 FILLER             PIC X(54) VALUE SPACES.
    05 TITLE-OUT           PIC X(24).
    05 FILLER             PIC X(54) VALUE SPACES.

01 HEAD-4.
    05 FILLER             PIC X(29) VALUE 'DEPRECIATION PERIOD ENT
                           'ERED: '.

    -                    PIC X VALUE QUOTE.
    05 FILLER             PIC X(7).
    05 PERIOD-OUT         PIC X VALUE QUOTE.
    05 FILLER             PIC X(54) VALUE SPACES.
    05 FILLER             PIC X(41) VALUE SPACES.

01 HEAD-5.
    05 FILLER             PIC X(10) VALUE SPACES.
    05 FILLER             PIC X(7)  VALUE 'ACCOUNT'.
    05 FILLER             PIC X(10) VALUE SPACES.
    05 FILLER             PIC X(4)  VALUE 'COST'.
    05 FILLER             PIC X(11) VALUE SPACES.
    05 FILLER             PIC X(19) VALUE 'YEARLY DEPRECIATION'.
    05 FILLER             PIC X(14) VALUE SPACES.
    05 FILLER             PIC X(14) VALUE 'CURRENT PERIOD'.
    05 FILLER             PIC X(43) VALUE SPACES.

01 HEAD-6.
    05 FILLER             PIC X(10) VALUE SPACES.
    05 FILLER             PIC X(6)  VALUE 'NUMBER'.
    05 FILLER             PIC X(10) VALUE SPACES.
    05 FILLER             PIC X(6)  VALUE 'CENTER'.

```

```

05 FILLER          PIC X(16) VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'AMOUNT'.
05 FILLER          PIC X(19) VALUE SPACES.
05 FILLER          PIC X(19) VALUE 'DEPRECIATION AMOUNT'.
05 FILLER          PIC X(21) VALUE SPACES.
05 FILLER          PIC X(8)  VALUE 'COMMENTS'.
05 FILLER          PIC X(11) VALUE SPACES.

01 DATA-LINE.
05 FILLER          PIC X(10).

05 ACCT-NO-OUT.
  10 FILLER          PIC X(9).
  10 COUNTY-CODE-OUT PIC X.

05 FILLER          PIC X(7).
05 COST-CNTR-OUT   PIC X(4).
05 FILLER          PIC X(13).
05 YEARLY-AMT-OUT  PIC -(8)---.
05 FILLER          PIC X(20).
05 CURRENT-AMT-OUT PIC -(8)---.
05 FILLER          PIC X(19).
05 COMMENT-OUT    PIC X(25).
05 FILLER          PIC XX.

PROCEDURE DIVISION.
OPEN-IO-FILES.
  OPEN INPUT CARD-IN,
    DEPRECIATION-TABFI,
    COST-CNTR-TABFI,
  OUTPUT PRINT-OUT,
    FINANCIAL-COST-ALLOCATION.

HOUSE-KEEP.
  MOVE SPACES TO DISK-OUT, DATA-LINE.
  PERFORM MOVE-SPACES.

READ-CARDIN.
  READ CARD-IN AT END MOVE
    'NO DATE CARD ** JOB ABORTED' TO PRINT-DATA,
  PERFORM PRINT-LINE-OUT,
  GO TO CLOSE-IO-FILES.

MOVE DATE-IN TO DATE-OUT.
MOVE TITLE-IN TO TITLE-OUT.

TEST-PERIOD.
  IF PERIOD1 = 'M' THEN
    MOVE ' MONTH '  TO PERIOD-OUT
    GO TO WRITE-FIRST-TWO-RECS.
  IF PERIOD1 = 'Y' THEN
    MOVE 1 TO PERIOD,
    MOVE ' YEAR '   TO PERIOD-OUT,
    GO TO WRITE-FIRST-TWO-RECS.
  IF PERIOD1 = 'Q' THEN
    MOVE 4 TO PERIOD,
    MOVE 'QUARTER'  TO PERIOD-OUT,
    GO TO WRITE-FIRST-TWO-RECS.

```

```

IF PERIOD1 = 'W' THEN
    MOVE 52 TO PERIOD,
    MOVE ' WEEK ' TO PERIOD-OUT,
    GO TO WRITE-FIRST-TWO-RECS.
INVALID-PERIOD-FALLTHRU.
    MOVE PERIOD-IN TO PERIOD-OUT.
    MOVE ' **** INVALID PERIOD ENTERED - DEFAULTED TO MONTH ****'
        TO COM1.
WRITE-FIRST-TWO-RECS.
    MOVE SPACES TO DISK-OUT.
    MOVE DATE-IN TO DISK-DATE.
    WRITE DISK-OUT.
    MOVE SPACES TO DISK-OUT.
    MOVE TITLE-IN TO DISK-TITLE.
    MOVE SOURCE-CODE TO SORT-SEQUENCE.
    WRITE DISK-OUT.
READ-DEPRECIATION-TABLE.
    READ DEPRECIATION-TABFI AT END
        MOVE 'DEPRECIATION TABLE CONTAINS NO DATA' TO PRINT-DATA
        PERFORM PRINT-LINE-OUT
        GO TO CLOSE-IO-FILES.
WRITE-HEADINGS.
    PERFORM MOVE-SPACES.
    WRITE PRINT-LINE BEFORE NEW-PAGE.
    MOVE PAGE-CT TO PAGE-OUT.
    ADD 1 TO PAGE-CT.
    MOVE 0 TO LINE-CT.
    MOVE HEAD-1 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-2 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-3 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-4 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-5 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-6 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PRINT-LINE-OUT.
SET-POINTERS.
    SET CC TO 0.
    MOVE NO-OF-RECS TO NO-OF-COST-CNTRS.
START-EDIT.
    SET CC UP BY 1.

```

```
IF CC > NO-OF-COST-CNTRS THEN
    GO TO CLOSE-IO-FILES.
MOVE ACCT-NO TO DISK-ACCT-NO, ACCT-NO-OUT.
MOVE COUNTY-CODE (CC) TO DISK-COUNTY-CODE COUNTY-CODE-OUT.
MOVE COST-CNTR (CC) TO DISK-COST-CNTR, COST-CNTR-OUT,
    CC-NUMINAL.
MOVE AMOUNT (CC) TO AMT-HOLD, YEARLY-AMT-OUT.
ADD AMOUNT (CC) TO YEARLY-DEP-TOTAL.
DIVIDE AMT-HOLD BY PERIOD GIVING DISK-CURRENT-AMT.
MOVE DISK-CURRENT-AMT TO CURRENT-AMT-OUT.
ADD DISK-CURRENT-AMT TO CURRENT-DEP-TOTAL.
PERFORM CHECK-COST-CNTR.
MOVE DATA-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE SPACES TO COMMENT-OUT.
WRITE DISK-OUT.
IF LINE-CT > MAX-LINE THEN
    PERFORM WRITE-HEADINGS.
GO TO START-EDIT.
CHECK-COST-CNTR.
READ COST-CNTR-TABFI INVALID KEY
    MOVE 'INVALID COST CENTER *****' TO COMMENT-OUT.
MOVE-SPACES.
MOVE SPACES TO PRINT-DATA.
PRINT-LINE-OUT.
    WRITE PRINT-LINE BEFORE 1.
    ADD 1 TO LINE-CT.
CLOSE-IO-FILES.
    PERFORM MOVE-SPACES.
    PERFORM PRINT-LINE-OUT 2 TIMES.
    MOVE SPACES TO DATA-LINE.
    MOVE YEARLY-DEP-TOTAL TO YEARLY-AMT-OUT.
    MOVE CURRENT-DEP-TOTAL TO CURRENT-AMT-OUT.
    MOVE ' DEPRECIATION TOTALS' TO COMMENT-OUT.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
CLOSE CARD-IN, DEPRECIATION-TABFI, COST-CNTR-TABFI,
    PRINT-OUT, FINANCIAL-COST-ALLOCATION.
E-O-J.
STOP RUN.
```

```
PGMNAME=WUASSCER
.SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASSCER.
AUTHOR. ACT SYSTEMS, INC
      SUITE 200
      807 W MORSE BLVD
      WINTER PARK, FLA 32789 .
DATE-WRITTEN. SEPT 23 1977.
DATE-COMPILED. SEPTEMBER 30 1977.
REMARKS. THIS PROGRAM EDITS THE SERVICE CONTRACT DATA
         CARDS, TOTALS ALL COSTS, FLAGS ANY DISCREPANCIES.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
      CO1 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
      SELECT CARD-IN
          ASSIGN TO UR-2540R-S-CARDIN
          ACCESS MODE IS SEQUENTIAL.
      SELECT FILE-SORT
          ASSIGN TO UT-3330-S-FILESORT
          ACCESS MODE IS SEQUENTIAL.
      SELECT SORTED-FILE
          ASSIGN TO UT-3330-S-SORTEDFI
          ACCESS MODE IS SEQUENTIAL.
      SELECT FINANCIAL-COST-ALLOCATION
          ASSIGN TO UT-3330-S-FICALFI
          ACCESS MODE IS SEQUENTIAL.
      SELECT PRINT-OUT
          ASSIGN TO UR-1403-S-PRINTOUT
          ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
  RECORD CONTAINS 80 CHARACTERS
  LABEL RECORDS ARE OMITTED
  DATA RECORD IS INPUT-CARD.
01 INPJT-CARD.
  02 ID1 PIC X.
  02 ID2 PIC X(79).
01 DATE-CARD.
  02 DATE-IN      PIC X(18).
  02 TITLE-IN     PIC X(24).
  02 FILLER       PIC X(38).
SD FILE-SORT
  RECORD CONTAINS 80 CHARACTERS
```

```

        DATA RECORD IS SORT-INPUT.
01  SORT-INPUT.
    02  S-SOURCE-CODE-NO  PIC X.
    02  FILLER            PIC X.
    02  S-CONTRACT-NO    PIC X(8).
    02  FILLER            PIC X(70).

FD  SORTED-FILE
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-RECS.

01  INPUT-RECS.
    02  SOURCE-CODE-NO PIC X.
    02  FILLER          PIC X.
    02  CONTRACT-NO    PIC X(8).
    02  DATA-ITEMS.
        05  NUMS OCCURS 7 TIMES PIC X(8).
    02  BALANCE          PIC X(8).
    02  FILLER            PIC X(6).

01  INPUT-RECS-NUMERIC.
    02  FILLER          PIC X(10).
    02  DAT-IN.
        05  NUMS2 OCCURS 7 TIMES PIC S9(6)V99.
    02  BAL-IN           PIC S9(6)V99.
    02  FILLER            PIC X(6).

FD  FINANCIAL-COST-ALLOCATION
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS DISK-OUT.

01  DISK-OUT.
    05  DISK-ACCT         PIC X(10).
    05  DISK-COST-CNTR    PIC X(4).
    05  DISK-COST-CAT     PIC X(2).
    05  DISK-AMOUNT        PIC S9(8)V99.

01  FIRST-TWO-RECS.
    05  FILLER            PIC X.
    05  SORT-SEQUENCE     PIC X.
    05  DISK-TITLE         PIC X(24).

FD  PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.

01  PRINT-LINE          PIC X(133).

WORKING-STORAGE SECTION.

77  LINE-CT             PIC 99 COMP VALUE 1.
77  PAGE-CT             PIC 9(4) COMP VALUE 1.
77  SOURCE-CODE          PIC X VALUE '3'.
77  SEARCH-NO            PIC XX VALUE SPACES.
77  ACCT-N01             PIC X(10) VALUE '680-30' '.
77  ACCT-N02             PIC X(10) VALUE '680-31' '.

```

```

77 TOTAL-ALL          PIC S9(8)V99  COMP VALUE 0.
77 BALANCE-ALL        PIC S9(8)V99  COMP VALUE 0.
77 CONTRACT-NO-HOLD   PIC X(8) VALUE '99999999'.
77 NUMERIC-HOLD        PIC X(8).
77 I                  PIC 9 COMP VALUE 0.
77 J                  PIC 9 COMP VALUE 0.
77 MAX-LINE           PIC 99  COMP VALUE 56.
77 BALANCE-ACCUM      PIC S9(6)V99  COMP VALUE 0.
77 TOTAL-ACCUM         PIC S9(6)V99  COMP VALUE 0.
77 SUB-TOTAL-BALANCE  PIC S9(6)V99  COMP VALUE 0.
77 SUB-TOTAL-TOTAL    PIC S9(6)V99  COMP VALUE 0.
01 ACCUM-ALL-SUBTOTAL.
05 ACCUM OCCURS 7 TIMES PIC S9(6)V99.
01 ACCUM-ALL-TOTAL.
05 ACCUM-ALL OCCURS 7 TIMES PIC S9(6)V99.
01 HEADING-1.
02 FILLER  PIC X(54) VALUE SPACES.
02 FILLER  PIC X(23) VALUE 'SERVICE CONTRACT REPORT'.
02 FILLER  PIC X(56) VALUE SPACES.
01 HEADING-2.
02 FILLER          PIC X VALUE SPACES.
02 FILLER          PIC X(13) VALUE 'DATE ENDING: '.
02 DATE-OUT         PIC X(18) VALUE SPACES.
02 FILLER          PIC X(33) VALUE SPACES.
02 FILLER          PIC X(3) VALUE 'FOR'.
02 FILLER          PIC X(53) VALUE SPACES.
02 FILLER          PIC X(5) VALUE 'PAGE '.
02 PAGE-OUT         PIC ZZZ9.
02 FILLER          PIC X(3) VALUE SPACES.
01 HEADING-3.
02 FILLER          PIC X(54) VALUE SPACES.
02 HEAD             PIC X(24) VALUE SPACES.
02 FILLER          PIC X(55) VALUE SPACES.
01 HEADING-4.
02 FILLER          PIC X VALUE SPACES.
02 FILLER          PIC X(20) VALUE 'CONTRACT # MATERIAL'.
02 FILLER          PIC X(5) VALUE SPACES.
02 FILLER          PIC X(16) VALUE 'LABOR      TRACTOR'.
02 FILLER          PIC X(3) VALUE SPACES.
02 FILLER          PIC X(20) VALUE 'COMPRESSOR      TRUCKS'.
02 FILLER          PIC X(5) VALUE SPACES.
02 FILLER          PIC X(17) VALUE 'OTHER      OVERHEAD'.
02 FILLER          PIC X(5) VALUE SPACES.
02 FILLER          PIC X(18) VALUE 'BALANCE      TOTAL'.
02 FILLER          PIC X(8) VALUE SPACES.
02 FILLER          PIC X(8) VALUE 'COMMENTS'.
02 FILLER          PIC X(7) VALUE SPACES.
01 DATA-LINE.
02 FILLER          PIC X VALUE SPACES.
02 CONTRACT-OUT      PIC X(8).

```

```

02 FILLER          PIC XX VALUE SPACES.
02 DATA-BLK OCCURS 7 TIMES.
    05 DAT          PIC -(6)---.
    05 FILLER       PIC XX.
02 BALANCE-OUT    PIC -(8)---.
02 TOTAL-OUT      PIC -(8)---.
02 FILLER          PIC XX VALUE SPACES.
02 COMMENT         PIC X(20) VALUE SPACES.
02 FILLER          PIC X VALUE SPACES.

01 ERROR-LIST.
    02 ERR-LIST-A.
        05 ERR-LIST-1 PIC X(100) VALUE      'INVALID CONTRACT NUM
-                                     'MATERIAL NON NUMERIC
-                                     'LABOR NON NUMERIC
-                                     'TRACTOR NON NUMERIC
-                                     'COMPRESSOR N-NUMERIC'.
-                                     'TRUCKS NON NUMERIC
-                                     'OTHER NON NUMERIC
-                                     'OVERHEAD NON NUMERIC
-                                     'BALANCE NON NUMERIC
-                                     'OUT OF BALANCE '.

        05 ERR-LIST-2 PIC X(100) VALUE      '.

02 ERR-LIST-B REDEFINES ERR-LIST-A.
    05 ERR-LIST OCCURS 10 TIMES PIC X(20).

01 ERROR-CODE.
    02 ERR-CODE OCCURS 10 TIMES PIC 9.

PROCEDURE DIVISION.

OPEN-FILES.
    OPEN INPUT CARD-IN.
    OPEN OUTPUT PRINT-OUT, FINANCIAL-COST-ALLOCATION.

READ-CARD.
    READ CARD-IN AT END GO TO CLOSE-FILES.
    MOVE SPACES TO DISK-OUT.
    MOVE DATE-IN TO DATE-OUT, DISK-TITLE.
    WRITE DISK-OUT.
    MOVE SPACES TO DISK-OUT.
    MOVE TITLE-IN TO HEAD, DISK-TITLE.
    MOVE SOURCE-CODE TO SORT-SEQUENCE.
    WRITE DISK-OUT.

WRITE-ERROR-HEADING.
    MOVE SPACES TO PRINT-LINE.
    MOVE ' THE FOLLOWING CARDS HAVE A INVALID SOURCE CODE NUMBER'
          TO PRINT-LINE.

    PERFORM PRINT-LINE-OUT THRU MOVE-SPACES 2 TIMES.

SORT-RECORDS.
    OPEN OUTPUT SORTED-FILE.
    SORT FILE-SORT ON ASCENDING KEY
        S-CONTRACT-NO
    INPUT PROCEDURE READ-SORT-INPUT THRU
        READ-SORT-INPUT-END
    OUTPUT PROCEDURE WRITE-SORT-OUTPUT THRU

```

```

        WRITE-SORT-OUTPUT-END.
CLOSE-SORT-FILES.
    CLOSE SORTED-FILE, CARD-IN.
    OPEN INPUT SORTED-FILE.
HOUSE-KEEPING.
    PERFORM WRITE-HEADINGS.
    MOVE ZEROS TO ERROR-CODE, ACCUM-ALL-SUBTOTAL,
                  ACCUM-ALL-TOTAL.

READ-INPUT-REC.
    READ SORTED-FILE AT END GO TO CLOSE-FILES.
DATA-TEST-SECTION.
    IF CONTRACT-NO NOT = TO CONTRACT-NO-HOLD THEN
        MOVE CONTRACT-NO TO CONTRACT-OUT
        ELSE MOVE SPACES TO CONTRACT-OUT.
    MOVE CONTRACT-NO TO CONTRACT-NO-HOLD.
    EXAMINE INPUT-RECS REPLACING ALL ' ' BY '0'.
    IF (CONTRACT-NO NOT NUMERIC) OR (CONTRACT-NO = ZERO) THEN
        MOVE 1 TU ERR-CODE (1).
    PERFORM TEST-FOR-NUMERIC VARYING I FROM 1 BY 1 UNTIL I = 8.
    MOVE BAL-IN TU BALANCE-OUT.
    ADD BAL-IN TO SUB-TOTAL-BALANCE.
    ADD BALANCE-ACCUM TO TOTAL-ACCUM.
    IF BAL-IN NOT = TO BALANCE-ACCUM THEN
        MOVE 1 TO ERR-CODE (10).
    MOVE ZEROS TO BALANCE-ACCUM.

READ-NEXT-INPUT-REC.
    PERFORM READ-INPUT-REC.

TOTAL-TEST.
    MOVE CONTRACT-NO TO NUMERIC-HOLD.
    IF NUMERIC-HOLD NOT EQUAL TO CONTRACT-NO-HOLD THEN
        MOVE TOTAL-ACCUM TO TOTAL-OUT,
        ADD TOTAL-ACCUM TO SUB-TOTAL-TOTAL,
        MOVE ZEROS TO TOTAL-ACCUM,
        ELSE MOVE ZEROS TO TOTAL-OUT.

ERROR-DETECTION.
    MOVE ZEROS TO I.
    IF ERROR-CODE NOT EQUAL TO ZEROS THEN
        PERFORM PRINT-ERRORS THRU PRINT-ERRORS-END UNTIL
        ERROR-CODE = ZEROS
    ELSE MOVE SPACES TO COMMENT,
        MOVE DATA-LINE TO PRINT-LINE,
        PERFORM PRINT-LINE-OUT THRU MOVE-SPACES.

TOTAL-TEST-END.
    IF TOTAL-OUT NOT = TO ZEROS PERFORM PAGE-OVFLOW.
    GO TO DATA-TEST-SECTION.

PRINT-ERRORS.
    ADD 1 TU I.
    IF ERR-CODE (I) EQUAL TO 1 THEN
        MOVE ERR-LIST (I) TO COMMENT
        MOVE DATA-LINE TO PRINT-LINE,

```

```
        PERFORM PRINT-LINE-OUT THRU MOVE-SPACES,
        MOVE SPACES TO DATA-LINE
        MOVE ZERO TO ERR-CODE (I).
PRINT-ERRORS-END.
        EXIT.
TEST-FOR-NUMERIC.
        ADD 1 I GIVING J.
        MOVE NUMS2 (I) TO DAT (I).
        ADD NUMS2 (I) TO BALANCE-ACCUM, ACCUM (I).
WRITE-HEADINGS.
        MOVE PAGE-CT TO PAGE-OUT.
        MOVE 0 TO LINE-CT.
        WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
        MOVE HEADING-1 TO PRINT-LINE.
        PERFORM PRINT-LINE-OUT THRU MOVE-SPACES.
        MOVE HEADING-2 TO PRINT-LINE.
        PERFORM PRINT-LINE-OUT THRU MOVE-SPACES.
        MOVE HEADING-3 TO PRINT-LINE.
        PERFORM PRINT-LINE-OUT THRU MOVE-SPACES 2 TIMES.
        MOVE HEADING-4 TO PRINT-LINE.
        PERFORM PRINT-LINE-OUT THRU MOVE-SPACES 2 TIMES.
        ADD 1 TO PAGE-CT.
PRINT-LINE-OUT.
        WRITE PRINT-LINE BEFORE 1.
        ADD 1 TO LINE-CT.
MOVE-SPACES.
        MOVE SPACES TO PRINT-LINE.
PAGE-OVFLOW.
        IF LINE-CT > MAX-LINE THEN
            PERFORM WRITE-SUB-TOTAL
            PERFORM WRITE-HEADINGS.
PAGE-OVFLOW-END.
        EXIT.
WRITE-SUB-TOTAL.
        PERFORM PRINT-LINE-OUT.
        PERFORM MOVE-OUT-ACCUMS VARYING I FROM 1 BY 1 UNTIL I = 8.
        MOVE 'SUBTOTAL' TO CONTRACT-OUT.
        MOVE SUB-TOTAL-BALANCE TO BALANCE-OUT.
        MOVE SUB-TOTAL-TOTAL TO TOTAL-OUT.
        ADD SUB-TOTAL-BALANCE TO BALANCE-ALL.
        ADD SUB-TOTAL-TOTAL TO TOTAL-ALL.
        MOVE ZEROS TO SUB-TOTAL-BALANCE, SUB-TOTAL-TOTAL,
                           ACCUM-ALL-SUBTOTAL.
        MOVE DATA-LINE TO PRINT-LINE.
        PERFORM PRINT-LINE-OUT THRU MOVE-SPACES.
MOVE-OUT-ACCUMS.
        MOVE ACCUM (I) TO DAT (I).
        ADD ACCUM (I) TO ACCUM-ALL (I).
MOVE-OUT-TOTAL-ACCUMS.
        MOVE ACCUM-ALL (I) TO DAT (I).
```

```

*
*      SORT ROUTINE
*
READ-SORT-INPUT.
  READ CARD-IN AT END GO TO READ-SORT-INPUT-END.
  IF ID1 NOT = TO SOURCE-CODE THEN
    WRITE PRINT-LINE FROM INPUT-CARD AFTER ADVANCING 1
  PERFORM MOVE-SPACES
    GO TO READ-SORT-INPUT.
  RELEASE SORT-INPUT FROM INPUT-CARD.
  GO TO READ-SORT-INPUT.
READ-SORT-INPUT-END.
  EXIT.
WRITE-SORT-OUTPUT.
  RETURN FILE-SORT RECORD INTO INPUT-RECS
    AT END GO TO WRITE-SORT-OUTPUT-END.
  WRITE INPUT-RECS.
  GO TO WRITE-SORT-OUTPUT.
WRITE-SORT-OUTPUT-END.
  EXIT.
*
*      END SORT ROUTINE
*
CLOSE-FILES.
  MOVE ZEROS TO CONTRACT-NO-HOLD.
  PERFORM TUTAL-TEST THRU ERRUR-DETECTION.
  PERFORM WRITE-SUB-TOTAL.
  PERFORM PRINT-LINE-OUT 2 TIMES.
  MOVE 'TOTAL' TO CONTRACT-OUT.
  PERFORM MOVE-OUT-TOTAL-ACUMS VARYING I
    FROM 1 BY 1 UNTIL I = 8.
  MOVE BALANCE-ALL TO BALANCE-OUT.
  MOVE TOTAL-ALL TO TOTAL-OUT.
  MOVE DATA-LINE TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
WRITE-DISK-RECORDS.
  MOVE SPACES TO DISK-OUT.
  MOVE ACCT-N01 TO DISK-ACCT.
  ADD ACCUM-ALL (3)
    ACCUM-ALL (4)
    ACCUM-ALL (5)
    ACCUM-ALL (6) GIVING DISK-AMOUNT.
  WRITE DISK-OUT.
  MOVE ACCT-N02 TO DISK-ACCT.
  MOVE ACCUM-ALL (7) TO DISK-AMOUNT.
  WRITE DISK-OUT.
  CLOSE CARD-IN, SORTED-FILE, PRINT-OUT,
    FINANCIAL-COST-ALLOCATION.
STOP-RUN.
  STOP RUN.

```

PGMNAME=WUASMTER
.SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASMTER.
AUTHOR. ACT SYSTEMS, INC
 SUITE 200
 807 W MORSE BLVD
 WINTER PARK, FLA 32789 .
DATE-WRITTEN. OCT 5 1977.
DATE-COMPILED. OCT 7 1977.
REMARKS. THIS PROGRAM EDITS THE MISCELLANEOUS TRANSACTIONS
CARDS, TOTALS ALL COSTS, FLAGS ANY DISCREPANCIES.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
 C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT CARD-IN
 ASSIGN TO UR-2540R-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
 SELECT FILE-SORT
 ASSIGN TO UT-3330-S-FILESORT
 ACCESS MODE IS SEQUENTIAL.
 SELECT SORTED-FILE
 ASSIGN TO UT-3330-S-SORTEDFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT PRINT-OUT
 ASSIGN TO UR-1403-S-PRINTOUT
 ACCESS MODE IS SEQUENTIAL.
 SELECT FINANCIAL-ACCT-TABFI
 ASSIGN TO DA-3330-I-FATABFI
 ACCESS MODE IS RANDOM
 NOMINAL KEY IS FA-NOMINAL
 RECORD KEY IS FA-NUMBER.
 SELECT COST-CNTR-TABFI
 ASSIGN TO DA-3330-I-CCTABFI
 ACCESS MODE IS RANDOM
 NOMINAL KEY IS CC-NOMINAL
 RECORD KEY IS CC-NUMBER.
 SELECT SPECIAL-ACCT-TABFI
 ASSIGN TO DA-3330-S-SATABFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT FINANCIAL-COST-ALLOCATION
 ASSIGN TO UT-3330-S-FICALFI
 ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.

FD CARD-IN
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS INPUT-CARD.

01 INPUT-CARD.
 02 ID1 PIC X.
 02 ID2 PIC X(79).

01 DATE-CARD.
 02 DATE-IN PIC X(18).
 02 TITLE-IN PIC X(24).
 02 FILLER PIC X(38).

SD FILE-SORT
RECORD CONTAINS 80 CHARACTERS
DATA RECORD IS SORT-INPUT.

01 SORT-INPUT.
 02 S-SOURCE-CODE-NO PIC X.
 02 FILLER PIC X.
 02 S-ENTRY-NO PIC X(3).
 02 FILLER PIC X.
 02 S-ACCT-NO PIC X(10).
 02 FILLER PIC X(64).

FD SORTED-FILE
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-RECS.

01 INPUT-RECS.
 02 SOURCE-CODE-NO PIC X.
 02 FILLER PIC X.
 02 ENTRY-NO PIC X(3).
 02 FILLER PIC X.
 02 ACCT-IN PIC X(10).
 02 FILLER PIC X.
 02 COST-CTR-IN.
 05 CC-FIRST-2-DIGITS PIC XX.
 05 CC-SECOND-2-DIGITS PIC XX.
 02 FILLER PIC X.
 02 INPUT-AMTS.
 05 AMOUNT-IN PIC X(10).
 05 AMT-IN REDEFINES AMOUNT-IN PIC S9(8)V99.
 02 FILLER PIC X(48).

FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.

01 PRINT-LINE PIC X(133).

FD COST-CNTR-TABFI
BLOCK CONTAINS 8 RECORDS
RECORD CONTAINS 30 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CC-REC.

```

01 CC-REC.
  02 CC-NUMBER      PIC 9(4).
  02 CC-NAME        PIC X(13) OCCURS 2 TIMES.

FD FINANCIAL-ACCT-TABFI
  BLOCK CONTAINS 8 RECORDS
  RECORD CONTAINS 58 CHARACTERS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS FA-REC.

01 FA-REC.
  05 FA-NUMBER      PIC X(10).
  05 FA-NAME        PIC X(20) OCCURS 2 TIMES.
  05 FILLER         PIC X(8).

FD SPECIAL-ACCT-TABFI
  BLOCK CONTAINS 0 RECORDS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS COST-CENTERS.

01 COST-CENTERS.
  05 RECORD-COUNT.
    10 NO-OF-RECS    PIC 9(4).
    10 FILLER        PIC X(18).

  05 COST-CNTRS OCCURS 1 TO 99 TIMES
    DEPENDING ON NO-OF-RECS
    ASCENDING KEY IS CC-SIZE-2
    INDEXED BY CC.
    10 CC-SIZE-2     PIC X(2).
    10 CC-SIZE-4     PIC X(4).
    10 FILLER        PIC X(16).

FD FINANCIAL-COST-ALLOCATION
  BLOCK CONTAINS 20 RECORDS
  RECORD CONTAINS 26 CHARACTERS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS DISK-OUT.

01 DISK-OUT.
  05 DISK-ACCT       PIC X(10).
  05 DISK-COST-CNTR  PIC X(4).
  05 DISK-COST-CAT   PIC X(2).
  05 DISK-AMOUNT     PIC S9(8)V99.

01 FIRST-TWO-RECS.
  05 FILLER         PIC X.
  05 SORT-SEQUENCE  PIC X.
  05 DISK-TITLE      PIC X(24).

WORKING-STORAGE SECTION.
  77 SOURCE-CODE      PIC X VALUE '5'.
  77 SEARCH-NO        PIC XX VALUE SPACES.
  77 LINE-CT          PIC 99 COMP VALUE 0.
  77 PAGE-CT          PIC 9(4) COMP VALUE 0.
  77 CR-AMT-ACCUM    PIC S9(8)V99 COMP VALUE ZERO.
  77 DB-AMT-ACCUM    PIC S9(8)V99 COMP VALUE ZERO.
  77 BALANCE-AMOUNTS PIC S9(8)V99 COMP VALUE ZERO.
  77 CR-TOTAL         PIC S9(8)V99 COMP VALUE ZERO.

```

```

77 DB-TOTAL          PIC S9(8)V99 COMP VALUE ZERO.
77 ENTRY-NO-HOLD    PIC X(3) VALUE ZEROS.
77 ACCT-HOLD         PIC X(10) VALUE ZERO.
77 I                PIC 9 COMP VALUE 0.
77 MAX-LINE         PIC 99 COMP VALUE 56.
77 FA-NOMINAL       PIC X(10).
77 CC-NOMINAL       PIC X(4).

01 HEADING-1.
  02 FILLER          PIC X(49) VALUE SPACES.
  02 FILLER          PIC X(33) VALUE 'MISCELLANEOUS TRANSACTI
                     'ONS REPORT'.
  -                  PIC X(51) VALUE SPACES.

01 HEADING-2.
  02 FILLER          PIC X VALUE SPACES.
  02 FILLER          PIC X(13) VALUE 'DATE ENDING: '.
  02 DATE-OUT         PIC X(18) VALUE SPACES.
  02 FILLER          PIC X(33) VALUE SPACES.
  02 FILLER          PIC X(3) VALUE 'FOR'.
  02 FILLER          PIC X(53) VALUE SPACES.
  02 FILLER          PIC X(5) VALUE 'PAGE '.
  02 PAGE-OUT         PIC ZZZ9.
  02 FILLER          PIC X(3) VALUE SPACES.

01 HEADING-3.
  02 FILLER          PIC X(54) VALUE SPACES.
  02 TITLE-OUT        PIC X(24) VALUE SPACES.
  02 FILLER          PIC X(55) VALUE SPACES.

01 HEADING-4.
  02 FILLER          PIC X(2) VALUE SPACES.
  02 FILLER          PIC X(12) VALUE 'ENTRY NUMBER'.
  02 FILLER          PIC X(5) VALUE SPACES.
  02 FILLER          PIC X(14) VALUE 'ACCOUNT NUMBER'.
  02 FILLER          PIC X(5) VALUE SPACES.
  02 FILLER          PIC X(11) VALUE 'CUST CENTER'.
  02 FILLER          PIC X(9) VALUE SPACES.
  02 FILLER          PIC X(6) VALUE 'AMOUNT'.
  02 FILLER          PIC X(6) VALUE SPACES.
  02 FILLER          PIC X(12) VALUE 'CREDIT TOTAL'.
  02 FILLER          PIC X(6) VALUE SPACES.
  02 FILLER          PIC X(11) VALUE 'DEBIT TOTAL'.
  02 FILLER          PIC X(14) VALUE SPACES.
  02 FILLER          PIC X(7) VALUE 'COMMENT'.
  02 FILLER          PIC X(13) VALUE SPACES.

01 DATA-LINE.
  02 FILLER          PIC X(7).
  02 ENTRY-NO-OUT     PIC X(3).
  02 FILLER          PIC X(12).
  02 ACCT-OUT         PIC X(10).
  02 FILLER          PIC X(10).
  02 COST-CTR-OUT    PIC X(4).
  02 FILLER          PIC X(8).

```

```

02 AMOUNT-OUT      PIC Z(8).ZZCR.
02 FILLER          PIC XX.
02 CR-TOTAL-OUT   PIC Z(8).ZZCR.
02 FILLER          PIC X(5).
02 DB-TOTAL-OUT   PIC -(8)---.
02 FILLER          PIC X(7).
02 COMMENT         PIC X(25) VALUE SPACES.
02 FILLER          PIC X(3).

01 ERROR-CODE.
02 ERR-CODE OCCURS 8 TIMES PIC 9.

01 ERROR-LIST.
02 ERR-LIST-A.
    05 ERR-LIST1 PIC X(100) VALUE 'ACCOUNT NUMBER NOT FOUND
    -                                     'COST CENTER NOT NUMERIC
    -                                     'COST CENTER NOT FOUND
    -                                     'AMOUNT NOT NUMERIC
    05 ERR-LIST2 PIC X(100) VALUE 'TOTAL AMOUNT NOT NUMERIC
    -                                     'CR AMTS NOT = TO CR TOTAL
    -                                     'DB AMTS NOT = TO DB TOTAL
    -                                     'TOTALS DO NOT BALANCE '.

02 ERR-LIST-B REDEFINES ERR-LIST-A.
    05 ERR-LIST OCCURS 8 TIMES PIC X(25).

PROCEDURE DIVISION.

OPEN-FILES.
    OPEN INPUT CARD-IN.
    OPEN OUTPUT PRINT-OUT, SORTED-FILE,
          FINANCIAL-COST-ALLOCATION.

READ-TITLE-CARD.
    READ CARD-IN AT END GO TO WRITE-FINAL-TOTALS.
    MOVE SPACES TO DISK-OUT.
    MOVE DATE-IN TO DATE-OUT, DISK-TITLE.
    WRITE DISK-OUT.
    MOVE SPACES TO DISK-OUT.
    MOVE TITLE-IN TO TITLE-OUT, DISK-TITLE.
    MOVE SOURCE-CODE TO SORT-SEQUENCE.
    WRITE DISK-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    MOVE ' THE FOLLOWING CARDS HAVE A INVALID SOURCE CODE'
        TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

SORT-RECORDS.
    SORT FILE-SURT ON ASCENDING KEY
        S-ENTRY-NO
        S-ACCT-NO
    INPUT PROCEDURE READ-SORT-INPUT
        THRU READ-SORT-INPUT-END
    OUTPUT PROCEDURE WRITE-SORT-OUTPUT
        THRU WRITE-SORT-OUTPUT-END.

CLOSE-SURT-FILES.

```

```

CLOSE SORTED-FILE, CARD-IN.
OPEN INPUT SORTED-FILE, FINANCIAL-ACCT-TABFI.
                           CUST-CNTR-TABFI, SPECIAL-ACCT-TABFI.

HOUSEKEEPING.
MOVE ZEROS TO ERROR-CODE.
READ SPECIAL-ACCT-TABFI AT END
MOVE 'COST CENTER TABLE CONTAINS NO DATA' TO
PRINT-LINE,
PERFORM PRINT-LINE-OUT
GO TO CLOSE-FILES.

WRITE-HEADINGS.
MOVE 0 TO LINE-CT.
ADD 1 TO PAGE-CT.
MOVE PAGE-CT TO PAGE-OUT.
PERFORM MOVE-SPACES.
WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
MOVE HEADING-1 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-2 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-3 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
MOVE HEADING-4 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

READ-RECORDS.
READ SORTED-FILE AT END GO TO WRITE-FINAL-TOTALS.

EDIT-RECORDS.
EXAMINE INPUT-AMTS REPLACING ALL SPACES BY ZEROS.
EXAMINE COST-CTR-IN REPLACING ALL SPACES BY ZEROS.
IF ENTRY-NO NOT = TO ENTRY-NO-HOLD THEN
MOVE ENTRY-NO TO ENTRY-NO-HOLD, ENTRY-NO-OUT
ELSE MOVE SPACES TO ENTRY-NO-OUT.
IF ACCT-IN NOT = ACCT-HOLD THEN
MOVE ACCT-IN TO ACCT-HOLD, ACCT-OUT, FA-NOMINAL,
READ FINANCIAL-ACCT-TABFI INVALID KEY
MOVE 1 TO ERR-CODE (1).
MOVE ACCT-IN TO DISK-ACCT, ACCT-OUT.
IF COST-CTR-IN NOT NUMERIC
MOVE 1 TO ERR-CODE (2)
MOVE SPACES TO COST-CTR-IN, DISK-COST-CNTR.
IF COST-CTR-IN = TO ZEROS THEN
MOVE SPACES TO COST-CTR-IN, DISK-COST-CNTR
ELSE PERFORM FIND-COST-CNTR.
IF AMOUNT-IN > 0 THEN
ADD AMT-IN TO DB-AMT-ACCUM
ELSE ADD AMT-IN TO CR-AMT-ACCUM.
MOVE AMT-IN TO AMOUNT-OUT, DISK-AMOUNT.

WRITE-LINE-OUT.

```

```

IF ERROR-CODE NOT = ZERO THEN
    PERFORM MOVE-IN-ERRORS VARYING I
        FROM 1 BY 1 UNTIL ERROR-CODE = ZEROS
    ELSE MOVE DATA-LINE TO PRINT-LINE
    PERFORM PRINT-LINE-OUT.

WRITE-DISK.
    WRITE DISK-OUT
    MOVE SPACES TO DISK-OUT.
PERFORM READ-RECORDS.
IF ENTRY-NO NOT = ENTRY-NO-HOLD THEN
    PERFORM WRITE-SUBS.
IF LINE-CT > MAX-LINE THEN
    PERFORM WRITE-HEADINGS.
GO TO EDIT-RECORDS.

FIND-COST-CNTR.
    IF CC-FIRST-2-DIGITS NOT = ZEROS THEN
        MOVE COST-CTR-IN TO CC-NOMINAL
        READ COST-CNTR-TABFI INVALID KEY
        MOVE 1 TO ERR-CODE (3)
    ELSE PERFORM SEARCH-CC.
    MOVE COST-CTR-IN TO DISK-COST-CNTR.

MOVE-IN-ERRORS.
    IF ERR-CODE (I) NOT = ZEROS THEN
        MOVE ERR-LIST (I) TO COMMENT
        MOVE DATA-LINE TO PRINT-LINE
        PERFORM PRINT-LINE-OUT
        MOVE SPACES TO DATA-LINE
        MOVE ZERO TO ERR-CODE (I).

PAGE-OVFLOW.
    IF LINE-CT NOT < MAX-LINE THEN
        PERFORM WRITE-SUBS
        PERFORM WRITE-HEADINGS.

PAGE-OVFLOW-END.
    EXIT.

SEARCH-CC.
    MOVE CC-SECOND-2-DIGITS TO SEARCH-NO.
    SET CC TO 1.
    SEARCH ALL COST-CNTRS AT END
        MOVE 1 TO ERR-CODE (3)
    WHEN CC-SIZE-2 (CC) = TO SEARCH-NO
        MOVE CC-SIZE-4 (CC) TO COST-CTR-IN.

WRITE-SUBS.
    MOVE SPACES TO DATA-LINE.
    MOVE CR-AMT-ACCUM TO CR-TOTAL-OUT.
    MOVE DB-AMT-ACCUM TO DB-TOTAL-OUT.
    ADD CR-AMT-ACCUM DB-AMT-ACCUM GIVING BALANCE-AMOUNTS.
    IF BALANCE-AMOUNTS NOT = 0 THEN
        MOVE ' CREDITS & DEBITS NOT = ' TO COMMENT
    ELSE MOVE SPACES TO COMMENT.
    PERFORM WRITE-LINE-OUT.

```

```

ADD CR-AMT-ACCUM TO CR-TOTAL.
ADD DB-AMT-ACCUM TO DB-TOTAL.
MOVE ZEROS TO CR-AMT-ACCUM,
          DB-AMT-ACCUM.
MOVE SPACES TO DATA-LINE.
MOVE-SPACES.
MOVE SPACES TO PRINT-LINE.
PRINT-LINE-OUT.
WRITE PRINT-LINE BEFORE ADVANCING 1.
ADD 1 TO LINE-CT.
*   SORT ROUTINE
*
READ-SORT-INPUT.
READ CARD-IN AT END GO TO READ-SORT-INPUT-END.
IF ID1 NOT = TO SOURCE-CODE THEN
    WRITE PRINT-LINE FROM INPUT-CARD AFTER ADVANCING 1
PERFORM MOVE-SPACES
    GO TO READ-SORT-INPUT.
RELEASE SORT-INPUT FROM INPUT-CARD.
GO TO READ-SORT-INPUT.
READ-SORT-INPUT-END.
EXIT.
WRITE-SORT-OUTPUT.
RETURN FILE-SURT RECORD INTO INPUT-RECS
    AT END GO TO WRITE-SORT-OUTPUT-END.
WRITE INPUT-RECS.
GO TO WRITE-SORT-OUTPUT.
WRITE-SORT-OUTPUT-END.
EXIT.
*
*   END SORT ROUTINE
*
WRITE-FINAL-TOTALS.
PERFORM WRITE-SUBS.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
PERFORM PRINT-LINE-OUT 2 TIMES.
MOVE SPACES TO DATA-LINE.
MOVE CR-TOTAL TO CR-TOTAL-OUT.
MOVE DB-TOTAL TO DB-TOTAL-OUT.
MOVE DATA-LINE TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
CLOSE-FILES.
CLOSE SORTED-FILE, FINANCIAL-ACCT-TABFI,
          SPECIAL-ACCT-TABFI, FINANCIAL-COST-ALLOCATION,
          COST-CNTR-TABFI, PRINT-OUT.
E-O-J.
STOP RUN.

```

```
PGMNAME=WUASPAER
-SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASPAER.
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
        807 W MORSE BLVD
        WINTER PARK, FLA 32789 .
DATE-WRITTEN. OCT 5 1977.
DATE-COMPILED. OCT 6 1977.
REMARKS. THIS PROGRAM EDITS THE PAYMENT AUTHORIZATION DATA
CARDS, TOTALS ALL COSTS, FLAGS ANY DISCREPANCIES.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT FILE-SORT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
    SELECT SORTED-FILE
        ASSIGN TO UT-3330-S-SORTEDFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT FINANCIAL-ACCT-TABFI
        ASSIGN TO DA-3330-I-FATABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS FA-NOMINAL
        RECORD KEY IS FA-NUMBER.
    SELECT COST-CNTR-TABFI
        ASSIGN TO DA-3330-I-CCTABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS CC-NOMINAL
        RECORD KEY IS CC-NUMBER.
    SELECT SPECIAL-ACCT-TABFI
        ASSIGN TO DA-3330-S-SATABFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT FINANCIAL-COST-ALLOCATION
        ASSIGN TO UT-3330-S-FICALFI
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
```

FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS INPUT-CARD.
 01 INPUT-CARD.
 02 ID1 PIC X.
 02 ID2 PIC X(79).
 01 DATE-CARD.
 02 DATE-IN PIC X(18).
 02 TITLE-IN PIC X(24).
 02 FILLER PIC X(38).
 SD FILE-SORT
 RECDR CONTAINS 80 CHARACTERS
 DATA RECORD IS SORT-INPUT.
 01 SORT-INPUT.
 02 S-SOURCE-CODE-NO PIC X.
 02 FILLER PIC X.
 02 S-PAY-AUTH PIC X(10).
 02 FILLER PIC X(68).
 FD SORTED-FILE
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS INPUT-RECS.
 01 INPUT-RECS.
 02 SOURCE-CODE-NO PIC X.
 02 FILLER PIC X.
 02 PAY-AUTH-IN PIC X(10).
 02 FILLER PIC X.
 02 ACCT-IN PIC X(10).
 02 FILLER PIC X.
 02 COST-CTR-IN.
 05 CC-FIRST-2-DIGITS PIC XX.
 05 CC-SECOND-2-DIGITS PIC XX.
 02 FILLER PIC X.
 02 AMOUNT-IN PIC X(10).
 02 AMT-IN REDEFINES AMOUNT-IN PIC S9(8)V99.
 02 FILLER PIC X.
 02 TOTAL-IN PIC X(10).
 02 TOT-IN REDEFINES TOTAL-IN PIC S9(8)V99.
 02 FILLER PIC X(30).
 FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.
 01 PRINT-LINE PIC X(133).
 FD COST-CNTR-TABFI
 BLOCK CONTAINS 8 RECORDS
 RECDR CONTAINS 30 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS CC-REC.

```

01 CC-REC.
  02 CC-NUMBER      PIC 9(4).
  02 CC-NAME        PIC X(13) OCCURS 2 TIMES.

FD FINANCIAL-ACCT-TABFI
  BLOCK CONTAINS 8 RECORDS
  RECORD CONTAINS 58 CHARACTERS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS FA-REC.

01 FA-REC.
  05 FA-NUMBER      PIC X(10).
  05 FA-NAME        PIC X(20) OCCURS 2 TIMES.
  05 FILLER         PIC X(8).

FD SPECIAL-ACCT-TABFI
  BLOCK CONTAINS 0 RECORDS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS COST-CENTERS.

01 COST-CENTERS.
  05 RECORD-COUNT.
    10 NO-OF-RECS   PIC 9(4).
    10 FILLER       PIC X(18).
  05 COST-CNTRS OCCURS 1 TO 99 TIMES
    DEPENDING ON NO-OF-RECS
    ASCENDING KEY IS CC-SIZE-2
    INDEXED BY CC.
    10 CC-SIZE-2    PIC X(2).
    10 CC-SIZE-4    PIC X(4).
    10 FILLER       PIC X(16).

FD FINANCIAL-COST-ALLOCATION
  BLOCK CONTAINS 20 RECORDS
  RECORD CONTAINS 26 CHARACTERS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS DISK-OUT.

01 DISK-OUT.
  05 DISK-ACCT      PIC X(10).
  05 DISK-COST-CNTR PIC X(4).
  05 DISK-COST-CAT  PIC X(2).
  05 DISK-AMOUNT    PIC S9(8)V99.

01 FIRST-TWO-RECS.
  05 FILLER         PIC X.
  05 SORT-SEQUENCE  PIC X.
  05 DISK-TITLE     PIC X(24).

WORKING-STORAGE SECTION.
  77 SOURCE-CODE    PIC X VALUE '4'.
  77 SEARCH-NO      PIC XX VALUE SPACES.
  77 LINE-CT        PIC 99 CEMP VALUE 0.
  77 PAGE-CT        PIC 9(4) CEMP VALUE 1.
  77 AMT-ACCUM      PIC S9(8)V99  COMP VALUE 0.
  77 AMT-SUB-TOTAL  PIC S9(8)V99  COMP VALUE 0.
  77 TOTAL-SUB-TOTAL PIC S9(8)V99  COMP VALUE 0.
  77 TOTAL-HOLD     PIC S9(8)V99  COMP VALUE 0.

```

```

77 TOTAL-ALL          PIC S9(10)V99 COMP VALUE 0.
77 AMT-ALL           PIC S9(10)V99 COMP VALUE 0.
77 PAY-AUTH-HOLD    PIC X(10) VALUE SPACES.
77 NUMERIC-HOLD     PIC X(10) VALUE SPACES.
77 I                 PIC 9 COMP VALUE 0.
77 MAX-LINE          PIC 99 COMP VALUE 50.
77 FA-NOMINAL        PIC X(10).
77 CC-NOMINAL        PIC X(4).
77 PRINT-PAY-AUTH-FLAG
88 FLAG-ON           PIC X VALUE '1'.
88 FLAG-OFF          PIC X VALUE '0'.

01 HEADING-1.
02 FILLER            PIC X(51) VALUE SPACES.
02 FILLER            PIC X(28) VALUE 'PAYMENT AUTHORIZATION'.
                               'REPURT'.
02 FILLER            PIC X(53) VALUE SPACES.

01 HEADING-2.
02 FILLER            PIC X VALUE SPACES.
02 FILLER PIC X(13)  VALUE 'DATE ENDING: '.
02 DATE-OUT          PIC X(18) VALUE SPACES.
02 FILLER            PIC X(33) VALUE SPACES.
02 FILLER            PIC X(3) VALUE 'FOR'.
02 FILLER            PIC X(53) VALUE SPACES.
02 FILLER            PIC X(5) VALUE 'PAGE '.
02 PAGE-OUT          PIC ZZZ9.
02 FILLER            PIC X(3) VALUE SPACES.
02 FILLER            PIC X(4) VALUE SPACES.

01 HEADING-3.
02 FILLER            PIC X(54) VALUE SPACES.
02 TITLE-OUT          PIC X(24) VALUE SPACES.
02 FILLER            PIC X(55) VALUE SPACES.

01 HEADING-4.
02 FILLER            PIC X(17) VALUE SPACES.
02 FILLER            PIC X(15) VALUE 'PAYMENT AUTH. #'.
02 FILLER            PIC X(8) VALUE SPACES.
02 FILLER            PIC X(9) VALUE 'ACCOUNT #'.
02 FILLER            PIC X(5) VALUE SPACES.
02 FILLER            PIC X(11) VALUE 'CUST CENTER'.
02 FILLER            PIC X(9) VALUE SPACES.
02 FILLER            PIC X(6) VALUE 'AMOUNT'.
02 FILLER            PIC X(11) VALUE SPACES.
02 FILLER            PIC X(5) VALUE 'TOTAL'.
02 FILLER            PIC X(11) VALUE SPACES.
02 FILLER            PIC X(8) VALUE 'COMMENTS'.
02 FILLER            PIC X(18) VALUE SPACES.

01 DATA-LINE.
02 FILLER            PIC X(20) VALUE SPACES.
02 PAY-AUTH-OUT      PIC X(10).
02 FILLER            PIC X(10) VALUE SPACES.
02 ACCT-OUT          PIC X(10).

```

```

02 FILLER          PIC X(8) VALUE SPACES.
02 COST-CTR-OUT   PIC X(4).
02 FILLER          PIC X(8) VALUE SPACES.
02 AMOUNT-OUT     PIC -(8)---.
02 FILLER          PIC X(5) VALUE SPACES.
02 TOTAL-OUT      PIC -(8)---.
02 FILLER          PIC X(5) VALUE SPACES.
02 COMMENT         PIC X(25) VALUE SPACES.
02 FILLER          PIC X(6) VALUE SPACES.

01 ERROR-CODE.
02 ERR-CODE OCCURS 7 TIMES PIC 9.

01 ERROR-LIST.
02 ERR-LIST-A.
05 ERR-LIST1 PIC X(75) VALUE 'ACCOUNT NUMBER NOT FOUND
-                                     'PAY AUTHORITY NOT NUMERIC
-                                     'COST CENTER NOT NUMERIC '.
05 ERR-LIST2 PIC X(100) VALUE 'COST CENTER NOT FOUND
-                                     'AMOUNT NOT NUMERIC
-                                     'TOTAL NOT NUMERIC
-                                     'AMOUNTS & TOTAL NOT EQUAL'.

02 ERR-LIST-B REDEFINES ERR-LIST-A.
05 ERR-LIST OCCURS 7 TIMES PIC X(25).

01 SUB-LINE.
02 FILLER          PIC X(5) VALUE SPACES.
02 S-T-COMMENT     PIC X(11) VALUE SPACES.
02 FILLER          PIC X(52) VALUE SPACES.
02 AMT-ALL-OUT    PIC -(10)---.
02 FILLER          PIC X(3) VALUE SPACES.
02 TOTAL-ALL-OUT   PIC -(10)---.
02 FILLER          PIC X(36) VALUE SPACES.

PROCEDURE DIVISION.

OPEN-FILES.
OPEN INPUT CARD-IN.
OPEN OUTPUT PRINT-OUT, SORTED-FILE,
      FINANCIAL-COST-ALLOCATION.

READ-TITLE-CARD.
READ CARD-IN AT END GO TO WRITE-FINAL-TOTALS.
MOVE SPACES TO DISK-OUT.
MOVE DATE-IN TO DATE-OUT, DISK-TITLE.
WRITE DISK-OUT.
MOVE SPACES TO DISK-OUT.
MOVE TITLE-IN TO TITLE-OUT, DISK-TITLE.
MOVE SOURCE-CODE TO SORT-SEQUENCE.
WRITE DISK-OUT.
MOVE SPACES TO PRINT-LINE.
MOVE ' THE FOLLOWING CARDS HAVE A INVALID SOURCE CODE'
      TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

SORT-RECORDS.

```

```

SORT FILE-SORT ON ASCENDING KEY
  S-PAY-AUTH
INPUT PROCEDURE READ-SORT-INPUT
  THRU READ-SORT-INPUT-END
OUTPUT PROCEDURE WRITE-SORT-OUTPUT
  THRU WRITE-SORT-OUTPUT-END.
CLOSE-SORT-FILES.
  CLOSE SORTED-FILE, CARD-IN.
  OPEN INPUT SORTED-FILE, FINANCIAL-ACCT-TABFI,
    COST-CNTR-TABFI, SPECIAL-ACCT-TABFI.
HOUSEKEEPING.
  MOVE ZEROS TO ERROR-CODE.
  READ SPECIAL-ACCT-TABFI AT END
    MOVE 'COST CENTER TABLE CONTAINS NO DATA' TU
    PRINT-LINE.
    PERFORM PRINT-LINE-OUT
    GO TO CLOSE-FILES.
WRITE-HEADINGS.
  MOVE 0 TO LINE-CT.
  MOVE PAGE-CT TO PAGE-OUT.
  PERFORM MOVE-SPACES.
  WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
  MOVE HEADING-1 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-2 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-3 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
  MOVE HEADING-4 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
  ADD 1 TU PAGE-CT.
READ-RECORDS.
  READ SORTED-FILE AT END GO TO WRITE-FINAL-TOTALS.
EDIT-RECORDS.
  MOVE ACCT-IN TO ACCT-OUT, FA-NOMINAL, DISK-ACCT.
  READ FINANCIAL-ACCT-TABFI INVALID KEY
    MOVE 1 TO ERR-CODE (1).
  MOVE PAY-AUTH-IN TO PAY-AUTH-HOLD.
  IF FLAG-ON MOVE PAY-AUTH-IN TO PAY-AUTH-OUT
    ELSE MOVE SPACES TO PAY-AUTH-OUT.
  MOVE COST-CTR-IN TO COST-CTR-OUT.
  EXAMINE INPUT-RECS REPLACING ALL ' ' BY '0'.
  IF PAY-AUTH-IN NOT NUMERIC MOVE 1 TO ERR-CODE (2).
  IF COST-CTR-IN NOT NUMERIC MOVE 1 TO ERR-CODE (3)
    MOVE ZEROS TO COST-CTR-IN.
  IF COST-CTR-IN NOT = ZEROS THEN
    PERFORM FIND-COST-CNTR.
  MOVE AMT-IN TO AMOUNT-OUT, DISK-AMOUNT.

```

```

        ADD AMT-IN TO AMT-ACCUM, AMT-SUB-TOTAL.
        IF TOT-IN NOT = TO 0 THEN MOVE TOT-IN TO TOTAL-HOLD.
            ADD TOTAL-HOLD TO TOTAL-SUB-TOTAL.
            MOVE 0 TO TOTAL-OUT.
READ-NEXT-REC.
    PERFORM READ-RECORDS.
CHECK-NEXT-REC.
    MOVE PAY-AUTH-IN TO NUMERIC-HOLD.
    IF NUMERIC-HOLD = TU PAY-AUTH-HOLD THEN
        MOVE '0' TO PRINT-PAY-AUTH-FLAG
        GO TO WRITE-LINE-OUT.
    MOVE '1' TO PRINT-PAY-AUTH-FLAG.
        MOVE TOTAL-HOLD TO TOTAL-OUT.
        IF TOTAL-HOLD NOT = TO AMT-ACCUM THEN
            MOVE 1 TO ERR-CODE (7).
        MOVE ZEROS TO AMT-ACCUM, TOTAL-HOLD.
WRITE-LINE-OUT.
    IF ERROR-CODE NOT = TO ZEROS THEN
        PERFORM MOVE-IN-ERRORS VARYING I
            FROM 1 BY 1 UNTIL ERROR-CODE = ZEROS
        ELSE MOVE DATA-LINE TO PRINT-LINE
        PERFORM PRINT-LINE-OUT
        WRITE DISK-OUT
        MOVE SPACES TO DISK-OUT.
    IF FLAG-ON THEN
        PERFORM PAGE-OVFLOW THRU PAGE-OVFLOW-END.
    GO TO EDIT-RECORDS.
FIND-COST-CNTR.
    IF CC-FIRST-2-DIGITS NOT = ZEROS THEN
        MOVE COST-CTR-IN TO CC-NOMINAL
        READ COST-CNTR-TABFI INVALID KEY
            MOVE 1 TO ERR-CODE (4)
        ELSE PERFORM SEARCH-CC.
        MOVE COST-CTR-IN TO DISK-COST-CNTR.
MOVE-IN-ERRORS.
    IF ERR-CODE (I) NOT = ZERO THEN
        MOVE ERR-LIST (I) TO COMMENT
        MOVE DATA-LINE TO PRINT-LINE
        PERFORM PRINT-LINE-OUT
        MOVE SPACES TO DATA-LINE
        MOVE ZERO TO ERR-CODE (I).
PAGE-OVFLOW.
    IF LINE-CT NOT < MAX-LINE THEN
        PERFORM WRITE-SUB-TOTALS
        PERFORM WRITE-HEADINGS.
PAGE-OVFLOW-END.
    EXIT.
SEARCH-CC.
    MOVE CC-SECOND-2-DIGITS TO SEARCH-NO.
    SET CC TO 1.

```

```

SEARCH ALL COST-CNTRS AT END
    MOVE 1 TO ERR-CODE (4)
    WHEN CC-SIZE-2 (CC) = TO SEARCH-NO
        MOVE CC-SIZE-4 (CC) TO COST-CTR-IN.

WRITE-SUB-TOTALS.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PRINT-LINE-OUT.
    MOVE 'SUB TOTAL ' TO S-T-COMMENT.
    MOVE AMT-SUB-TOTAL TO AMT-ALL-OUT.
    MOVE TOTAL-SUB-TOTAL TO TUTAL-ALL-OUT.
    MOVE SUB-LINE TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    ADD AMT-SUB-TOTAL TO AMT-ALL.
    ADD TOTAL-SUB-TOTAL TO TOTAL-ALL.
    MOVE ZEROS TO AMT-SUB-TOTAL, TOTAL-SUB-TOTAL.
MOVE-SPACES.
    MOVE SPACES TO PRINT-LINE.
PRINT-LINE-OUT.
    WRITE PRINT-LINE BEFORE ADVANCING 1.
    ADD 1 TU LINE-CT.

*
READ-SORT-INPUT.
    READ CARD-IN AT END GO TO READ-SORT-INPUT-END.
    IF ID1 NOT = TO SOURCE-CODE THEN
        WRITE PRINT-LINE FROM INPUT-CARD AFTER ADVANCING 1
    PERFORM MOVE-SPACES
    GO TO READ-SORT-INPUT.
    RELEASE SORT-INPUT FROM INPUT-CARD.
    GO TO READ-SORT-INPUT.
READ-SORT-INPUT-END.
    EXIT.

WRITE-SORT-OUTPUT.
    RETURN FILE-SORT RECORD INTO INPUT-RECS
        AT END GO TO WRITE-SORT-OUTPUT-END.
    WRITE INPUT-RECS.
    GO TO WRITE-SORT-OUTPUT.
WRITE-SORT-OUTPUT-END.
    EXIT.

*
*      END SORT ROUTINE
*

WRITE-FINAL-TOTALS.
    IF ERROR-CODE = TO ZERO THEN
        WRITE DISK-OUT.
    MOVE TOTAL-HOLD TO TOTAL-OUT.
    MOVE DATA-LINE TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM WRITE-SUB-TOTALS.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

```

```
PERFORM PRINT-LINE-OUT.  
MOVE 'GRAND TOTAL' TO S-T-COMMENT.  
MOVE TOTAL-ALL TO TOTAL-ALL-OUT.  
MOVE AMT-ALL TO AMT-ALL-OUT.  
MOVE SUB-LINE TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
CLOSE-FILES.  
CLOSE SORTED-FILE, FINANCIAL-ACCT-TABFI,  
SPECIAL-ACCT-TABFI, FINANCIAL-COST-ALLOCATION,  
COST-CNTR-TABFI, PRINT-OUT.  
E-O-J.  
STOP RUN.
```

```
PGMNAME=WUASRSER
.SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASRSER.
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FLA 32789 .
DATE-WRITTEN. OCT 3 1977.
DATE-COMPILED. OCT 5 1977.
REMARKS. THIS PROGRAM EDITS THE REQUISITION FOR STOCK DATA
        CARDS, TOTALS ALL COSTS, FLAGS ANY DISCREPANCIES.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    CO1 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT FILE-SORT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
    SELECT SORTED-FILE
        ASSIGN TO UT-3330-S-SORTEDFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT FINANCIAL-ACCT-TABFI
        ASSIGN TO DA-3330-I-FATABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS FA-NOMINAL
        RECORD KEY IS FA-NUMBER.
    SELECT COST-CNTR-TABFI
        ASSIGN TO DA-3330-I-CCTABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS CC-NOMINAL
        RECORD KEY IS CC-NUMBER.
    SELECT SPECIAL-ACCT-TABFI
        ASSIGN TO DA-3330-S-SATABFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT FINANCIAL-COST-ALLOCATION
        ASSIGN TO UT-3330-S-FICALFI
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
```

FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS INPUT-CARD.
 01 INPUT-CARD.
 02 ID1 PIC X.
 02 ID2 PIC X(79).
 01 DATE-CARD.
 02 DATE-IN PIC X(18).
 02 TITLE-IN PIC X(24).
 02 FILLER PIC X(38).
 SD FILE-SORT
 RECORD CONTAINS 80 CHARACTERS
 DATA RECORD IS SORT-INPUT.
 01 SORT-INPUT.
 02 S-SOURCE-CODE-NO PIC X.
 02 FILLER PIC X.
 02 S-REQ-NO PIC X(10).
 02 FILLER PIC X(68).
 FD SORTED-FILE
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS INPUT-RECS.
 01 INPUT-RECS.
 02 SOURCE-CODE-NO PIC X.
 02 FILLER PIC X.
 02 REQ-IN PIC X(10).
 02 FILLER PIC X.
 02 ACCT-IN PIC X(10).
 02 FILLER PIC X.
 02 COST-CTR-IN.
 05 CC-FIRST-2-DIGITS PIC XX.
 05 CC-SECOND-2-DIGITS PIC XX.
 02 FILLER PIC X.
 02 AMOUNT-IN PIC X(10).
 02 AMT-IN REDEFINES AMOUNT-IN PIC S9(8)V99.
 02 FILLER PIC X.
 02 TOTAL-IN PIC X(10).
 02 TUT-IN REDEFINES TOTAL-IN PIC S9(8)V99.
 02 FILLER PIC X(30).
 FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.
 01 PRINT-LINE PIC X(133).
 FD COST-CNTR-TABFI
 BLOCK CONTAINS 8 RECORDS
 RECORD CONTAINS 30 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS CC-REC.

```

01 CC-REC.
  02 CC-NUMBER      PIC 9(4).
  02 CC-NAME       PIC X(13) OCCURS 2 TIMES.

FD FINANCIAL-ACCT-TABFI
  BLOCK CONTAINS 8 RECORDS
  RECORD CONTAINS 58 CHARACTERS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS FA-REC.

01 FA-REC.
  05 FA-NUMBER      PIC X(10).
  05 FA-NAME       PIC X(20) OCCURS 2 TIMES.
  05 FILLER        PIC X(8).

FD SPECIAL-ACCT-TABFI
  BLOCK CONTAINS 0 RECORDS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS COST-CENTERS.

01 COST-CENTERS.
  05 RECORD-COUNT.
    10 NO-OF-RECS   PIC 9(4).
    10 FILLER       PIC X(18).
  05 COST-CNTRS OCCURS 1 TO 99 TIMES
    DEPENDING ON NO-OF-RECS
    ASCENDING KEY IS CC-SIZE-2
    INDEXED BY CC.
    10 CC-SIZE-2   PIC X(2).
    10 CC-SIZE-4   PIC X(4).
    10 FILLER       PIC X(16).

FD FINANCIAL-COST-ALLOCATION
  BLOCK CONTAINS 20 RECORDS
  RECORD CONTAINS 26 CHARACTERS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS DISK-OUT.

01 DISK-OUT.
  05 DISK-ACCT      PIC X(10).
  05 DISK-COST-CNTR PIC X(4).
  05 DISK-COST-CAT  PIC X(2).
  05 DISK-AMOUNT    PIC S9(8)V99.

01 FIRST-TWO-RECS.
  05 FILLER        PIC X.
  05 SORT-SEQUENCE PIC X.
  05 DISK-TITLE    PIC X(24).

WORKING-STORAGE SECTION.

77 SOURCE-CODE      PIC X VALUE '2'.
77 SEARCH-NO        PIC XX VALUE SPACES.
77 LINE-CT          PIC 99 CUMP VALUE 0.
77 PAGE-CT          PIC 9(4) CUMP VALUE 1.
77 AMT-ACCUM        PIC S9(8)V99  COMP VALUE 0.
77 AMT-SUB-TOTAL   PIC S9(8)V99  COMP VALUE 0.
77 TOTAL-SUB-TOTAL PIC S9(8)V99  COMP VALUE 0.
77 TOTAL-ALL        PIC S9(10)V99 COMP VALUE 0.

```

```

77 AMT-ALL          PIC S9(10)V99 COMP VALUE 0.
77 REQ-HOLD         PIC X(10) VALUE SPACES.
77 NUMERIC-HOLD    PIC X(10) VALUE SPACES.
77 TOTAL-HOLD      PIC S9(8)V99 COMP VALUE 0.
77 I                PIC 9 COMP VALUE 0.
77 MAX-LINE        PIC 99 COMP VALUE 50.
77 FA-NOMINAL      PIC X(10).
77 CC-NOMINAL      PIC X(4).
77 PRINT-REQ-FLAG  PIC X VALUE '1'.
88 FLAG-ON          VALUE '1'.
88 FLAG-OFF         VALUE '0'.
01 HEADING-1.
02 FILLER          PIC X(51) VALUE SPACES.
02 FILLER          PIC X(29) VALUE 'REQUISITION FROM STOCK
                   ' REPORT'.
-                  PIC X(53) VALUE SPACES.
01 HEADING-2.
02 FILLER          PIC X VALUE SPACES.
02 FILLER          PIC X(13) VALUE 'DATE ENDING: '.
02 DATE-OUT         PIC X(18) VALUE SPACES.
02 FILLER          PIC X(33) VALUE SPACES.
02 FILLER          PIC X(3) VALUE 'FOR'.
02 FILLER          PIC X(53) VALUE SPACES.
02 FILLER          PIC X(5) VALUE 'PAGE '.
02 PAGE-OUT         PIC ZZZ9.
02 FILLER          PIC X(3) VALUE SPACES.
01 HEADING-3.
02 FILLER          PIC X(54) VALUE SPACES.
02 TITLE-OUT        PIC X(24) VALUE SPACES.
02 FILLER          PIC X(55) VALUE SPACES.
01 HEADING-4.
02 FILLER          PIC X VALUE SPACES.
02 FILLER          PIC X(10) VALUE SPACES.
02 FILLER          PIC X(13) VALUE 'REQUISITION #'.
02 FILLER          PIC X(7) VALUE SPACES.
02 FILLER          PIC X(9) VALUE 'ACCOUNT #'.
02 FILLER          PIC X(6) VALUE SPACES.
02 FILLER          PIC X(11) VALUE 'COST CENTER'.
02 FILLER          PIC X(8) VALUE SPACES.
02 FILLER          PIC X(6) VALUE 'AMOUNT'.
02 FILLER          PIC X(10) VALUE SPACES.
02 FILLER          PIC X(5) VALUE 'TOTAL'.
02 FILLER          PIC X(15) VALUE SPACES.
02 FILLER          PIC X(8) VALUE 'COMMENTS'.
02 FILLER          PIC X(24) VALUE SPACES.
01 DATA-LINE.
02 FILLER          PIC X(10) VALUE SPACES.
02 REQ-OUT          PIC X(10).
02 FILLER          PIC X(10) VALUE SPACES.
02 ACCT-OUT         PIC X(10).

```

```

02 FILLER          PIC X(8) VALUE SPACES.
02 COST-CTR-OUT   PIC X(4).
02 FILLER          PIC X(8) VALUE SPACES.
02 AMOUNT-OUT     PIC -(8).--.
02 FILLER          PIC X(4) VALUE SPACES.
02 TUTAL-OUT       PIC -(8).--.
02 FILLER          PIC X(9) VALUE SPACES.
02 COMMENT         PIC X(25) VALUE SPACES.
02 FILLER          PIC X(13) VALUE SPACES.

01 ERROR-CODE.
02 ERR-CODE OCCURS 7 TIMES PIC 9.

01 ERROR-LIST.
02 ERR-LIST-A.
05 ERR-LIST1 PIC X(75) VALUE 'ACCOUNT NUMBER NOT FOUND
-                                'REQUISITION-NO NONNUMERIC
-                                'COST CENTER NOT NUMERIC '.
05 ERR-LIST2 PIC X(100) VALUE 'COST CENTER NOT FOUND
-                                'AMOUNT NOT NUMERIC
-                                'TOTAL NOT NUMERIC
-                                'AMOUNTS & TOTAL NOT EQUAL'.

02 ERR-LIST-B REDEFINES ERR-LIST-A.
05 ERR-LIST OCCURS 7 TIMES PIC X(25).

01 SUB-LINE.
02 FILLER          PIC X(7) VALUE SPACES.
02 S-T-COMMENT    PIC X(11) VALUE SPACES.
02 FILLER          PIC X(40) VALUE SPACES.
02 AMT-ALL-OUT    PIC -(10).--.
02 FILLER          PIC X(2) VALUE SPACES.
02 TOTAL-ALL-OUT  PIC -(10).--.
02 FILLER          PIC X(47) VALUE SPACES.

PROCEDURE DIVISION.
OPEN-FILES.
  OPEN INPUT CARD-IN.
  OPEN OUTPUT PRINT-OUT, SORTED-FILE,
        FINANCIAL-COST-ALLOCATION.

READ-TITLE-CARD.
  READ CARD-IN AT END GO TO WRITE-FINAL-TOTALS.
  MOVE SPACES TO DISK-OUT.
  MOVE DATE-IN TO DATE-OUT, DISK-TITLE.
  WRITE DISK-OUT.
  MOVE SPACES TO DISK-OUT.
  MOVE TITLE-IN TO TITLE-OUT, DISK-TITLE.
  MOVE SOURCE-CODE TO SORT-SEQUENCE.
  WRITE DISK-OUT.
  MOVE SPACES TO PRINT-LINE.
  MOVE ' THE FOLLOWING CARDS HAVE A INVALID SOURCE CODE'
        TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

SORT-RECORDS.

```

```
SORT FILE-SORT ON ASCENDING KEY
  S-REQ-NO
INPUT PROCEDURE READ-SORT-INPUT
  THRU READ-SURT-INPUT-END
OUTPUT PROCEDURE WRITE-SORT-OUTPUT
  THRU WRITE-SORT-OUTPUT-END.
CLOSE-SORT-FILES.
  CLOSE SORTED-FILE, CARD-IN.
  OPEN INPUT SORTED-FILE, FINANCIAL-ACCT-TABFI,
    COST-CNTR-TABFI, SPECIAL-ACCT-TABFI.
HOUSEKEEPING.
  MOVE ZEROS TO ERROR-CODE.
  READ SPECIAL-ACCT-TABFI AT END
    MOVE 'COST CENTER TABLE CONTAINS NO DATA' TO
      PRINT-LINE,
    PERFORM PRINT-LINE-OUT
    GO TO CLOSE-FILES.
WRITE-HEADINGS.
  MOVE 0 TO LINE-CT.
  MOVE PAGE-CT TO PAGE-OUT.
  PERFORM MOVE-SPACES.
  WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
  MOVE HEADING-1 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-2 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-3 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
  MOVE HEADING-4 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
  ADD 1 TO PAGE-CT.
READ-RECORDS.
  READ SORTED-FILE AT END GO TO WRITE-FINAL-TOTALS.
EDIT-RECORDS.
  MOVE ACCT-IN TO ACCT-OUT, FA-NOMINAL, DISK-ACCT.
  READ FINANCIAL-ACCT-TABFI INVALID KEY
    MOVE 1 TO ERR-CODE (1).
  MOVE REQ-IN TO REQ-HOLD.
  IF FLAG-ON MOVE REQ-IN TO REQ-OUT
  ELSE MOVE SPACES TO REQ-OUT.
  MOVE COST-CTR-IN TO COST-CTR-OUT.
  EXAMINE INPUT-RECS REPLACING ALL ' ' BY '0'.
  IF REQ-IN NOT NUMERIC MOVE 1 TO ERR-CODE (2).
  IF COST-CTR-IN NOT NUMERIC MOVE 1 TO ERR-CODE (3)
    MOVE ZEROS TO COST-CTR-IN.
  IF COST-CTR-IN NOT = ZEROS THEN
    PERFORM FIND-COST-CNTR.
  MOVE AMT-IN TO AMOUNT-OUT, DISK-AMOUNT.
```

```

        ADD AMT-IN TO AMT-ACCUM, AMT-SUB-TOTAL.
        IF TOT-IN NOT = TO 0 THEN MOVE TOT-IN TO TOTAL-HOLD,
          ADD TOTAL-HOLD TO TOTAL-SUB-TOTAL.
        MOVE 0 TO TOTAL-OUT.
READ-NEXT-REC.
        PERFORM READ-RECORDS.
CHECK-NEXT-REC.
        MOVE REQ-IN TO NUMERIC-HOLD.
        IF NUMERIC-HOLD = TO REQ-HOLD THEN
          MOVE '0' TO PRINT-REQ-FLAG
          GO TO WRITE-LINE-OUT.
        MOVE '1' TO PRINT-REQ-FLAG.
        MOVE TOTAL-HOLD TO TOTAL-OUT.
        IF TOTAL-HOLD NOT = TO AMT-ACCUM THEN
          MOVE 1 TO ERR-CODE (7).
        MOVE ZEROS TO AMT-ACCUM, TOTAL-HOLD.
WRITE-LINE-OUT.
        IF ERROR-CODE NOT = ZERO THEN
          PERFORM MOVE-IN-ERRORS VARYING I
            FROM 1 BY 1 UNTIL ERROR-CODE = ZEROS
          ELSE MOVE DATA-LINE TO PRINT-LINE
          PERFORM PRINT-LINE-OUT
          WRITE DISK-OUT
          MOVE SPACES TO DISK-OUT.
        IF FLAG-ON THEN
          PERFORM PAGE-OVFLOW THRU PAGE-OVFLOW-END.
        GO TO EDIT-RECORDS.
FIND-COST-CNTR.
        IF CC-FIRST-2-DIGITS NOT = ZEROS THEN
          MOVE COST-CTR-IN TO CC-NOMINAL
          READ COST-CNTR-TABFI INVALID KEY
          MOVE 1 TO ERR-CODE (4)
        ELSE PERFORM SEARCH-CC.
        MOVE COST-CTR-IN TO DISK-COST-CNTR.
MOVE-IN-ERRORS.
        IF ERR-CODE (I) NOT = ZEROS THEN
          MOVE ERR-LIST (I) TO COMMENT
          MOVE DATA-LINE TO PRINT-LINE
          PERFORM PRINT-LINE-OUT
          MOVE SPACES TO DATA-LINE
          MOVE ZERO TO ERR-CODE (I).
PAGE-OVFLOW.
        IF LINE-CT NOT < MAX-LINE THEN
          PERFORM WRITE-SUB-TOTALS
          PERFORM WRITE-HEADINGS.
PAGE-OVFLOW-END.
        EXIT.
SEARCH-CC.
        MOVE CC-SECOND-2-DIGITS TO SEARCH-NO.
        SET CC TO 1.

```

```

SEARCH ALL COST-CNTRS AT END
    MOVE 1 TO ERR-CODE (4)
WHEN CC-SIZE-2 (CC) = TO SEARCH-NO
    MOVE CC-SIZE-4 (CC) TO COST-CTR-IN.

WRITE-SUB-TOTALS.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PRINT-LINE-OUT.
    MOVE 'SUB TOTAL ' TO S-T-COMMENT.
    MOVE AMT-SUB-TOTAL TO AMT-ALL-OUT.
    MOVE TOTAL-SUB-TOTAL TO TOTAL-ALL-OUT.
    MOVE SUB-LINE TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    ADD AMT-SUB-TOTAL TO AMT-ALL.
    ADD TOTAL-SUB-TOTAL TO TOTAL-ALL.
    MOVE ZEROS TO AMT-SUB-TOTAL, TOTAL-SUB-TOTAL.

MOVE-SPACES.
    MOVE SPACES TO PRINT-LINE.

PRINT-LINE-OUT.
    WRITE PRINT-LINE BEFORE ADVANCING 1.
    ADD 1 TO LINE-CT.

* SORT ROUTINE
*
READ-SORT-INPUT.
    READ CARD-IN AT END GO TO READ-SORT-INPUT-END.
    IF ID1 NOT = TO SOURCE-CODE THEN
        WRITE PRINT-LINE FROM INPUT-CARD AFTER ADVANCING 1
    PERFORM MOVE-SPACES
        GO TO READ-SORT-INPUT.
    RELEASE SORT-INPUT FROM INPUT-CARD.
    GO TO READ-SORT-INPUT.

READ-SORT-INPUT-END.
    EXIT.

WRITE-SORT-OUTPUT.
    RETURN FILE-SORT RECORD INTO INPUT-RECS
    AT END GO TO WRITE-SORT-OUTPUT-END.
    WRITE INPUT-RECS.
    GO TO WRITE-SORT-OUTPUT.

WRITE-SORT-OUTPUT-END.
    EXIT.

*
* END SORT ROUTINE
*
WRITE-FINAL-TOTALS.
    IF ERROR-CODE = TO ZERO THEN
        WRITE DISK-OUT.
    MOVE TOTAL-HOLD TO TOTAL-OUT.
    MOVE DATA-LINE TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    PERFORM WRITE-SUB-TOTALS.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

```

```
PERFORM PRINT-LINE-OUT.  
MOVE 'GRAND TOTAL' TO S-T-COMMENT.  
MOVE TOTAL-ALL TO TOTAL-ALL-OUT.  
MOVE AMT-ALL TO AMT-ALL-OUT.  
MOVE SUB-LINE TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
CLOSE-FILES.  
    CLOSE SORTED-FILE, FINANCIAL-ACCT-TABFI,  
        SPECIAL-ACCT-TABFI, FINANCIAL-CUST-ALLOCATION,  
        COST-CNTR-TABFI, PRINT-OUT.  
E-O-J.  
STOP RUN.
```

PGMNAME=WUASPRER
.SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASPRER.
AUTHOR. ACT SYSTEMS, INC
 SUITE 200
 807 W MORSE BLVD
 WINTER PARK, FLA 32789 .
DATE-WRITTEN. OCT 8 1977.
DATE-COMPILED. OCT 10 1977.
REMARKS. THIS PROGRAM EDITS THE PAYROLL REPORT DATA
CARDS, TOTALS ALL COSTS, FLAGS ANY DISCREPANCIES.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
 C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT CARD-IN
 ASSIGN TO UR-2540R-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
 SELECT SORTED-FILE
 ASSIGN TO UT-3330-S-SORTEDFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT PRINT-OUT
 ASSIGN TO UR-1403-S-PRINTOUT
 ACCESS MODE IS SEQUENTIAL.
 SELECT FINANCIAL-ACCT-TABFI
 ASSIGN TO DA-3330-I-FATABFI
 ACCESS MODE IS RANDOM
 NOMINAL KEY IS FA-NUMINAL
 RECORD KEY IS FA-NUMBER.
 SELECT COST-CNTR-TABFI
 ASSIGN TO DA-3330-I-CCTABFI
 ACCESS MODE IS RANDOM
 NOMINAL KEY IS CC-NOMINAL
 RECORD KEY IS CC-NUMBER.
 SELECT SPECIAL-ACCT-TABFI
 ASSIGN TO DA-3330-S-SATABFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT FINANCIAL-COST-ALLOCATION
 ASSIGN TO UT-3330-S-FICALFI
 ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE OMITTED

DATA RECORD IS INPUT-CARD.
 01 INPUT-CARD.
 02 ID1 PIC X.
 02 FILLER PIC X(11).
 02 DATE-DATA PIC X(6).
 02 FILLER PIC X(62).
 01 DATE-CARD.
 02 DATE-IN PIC X(18).
 02 TITLE-IN PIC X(24).
 02 FILLER PIC X(38).
 FD SORTED-FILE
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS INPUT-RECS.
 01 INPUT-RECS.
 02 SOURCE-CODE-NO PIC X.
 02 FILLER PIC X.
 02 EMPL-NO-IN.
 05 EMP1-IN PIC X(3).
 05 EMP2-IN PIC XX.
 05 EMP3-IN PIC X(4).
 02 FILLER PIC X.
 02 DATE-DATA-IN PIC X(6).
 02 FILLER PIC X.
 02 ACCT-IN PIC X(10).
 02 FILLER PIC X.
 02 COST-CTR-IN.
 05 CC-FIRST-2-DIGITS PIC XX.
 05 CC-SECOND-2-DIGITS PIC XX.
 02 FILLER PIC X.
 02 TOTAL-CUST-IN PIC X(10).
 02 TOT-COST-IN REDEFINES TOTAL-COST-IN PIC S9(8)V99.
 02 FILLER PIC X.
 02 GROSS-PAY-IN PIC X(10).
 02 GRO-PAY-IN REDEFINES GROSS-PAY-IN PIC S9(8)V99.
 02 FILLER PIC X(24).
 FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.
 01 PRINT-LINE PIC X(133).
 FD COST-CNTR-TABFI
 BLOCK CONTAINS 8 RECORDS
 RECORD CONTAINS 30 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS CC-REC.
 01 CC-REC.
 02 CC-NUMBER PIC 9(4).
 02 CC-NAME PIC X(13) OCCURS 2 TIMES.
 FD FINANCIAL-ACCT-TABFI

BLOCK CONTAINS 8 RECORDS
 RECORD CONTAINS 58 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS FA-REC.
01 FA-REC.
 05 FA-NUMBER PIC X(10).
 05 FA-NAME PIC X(20) OCCURS 2 TIMES.
 05 FILLER PIC X(8).
FD SPECIAL-ACCT-TABFI
 BLOCK CONTAINS 0 RECORDS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS COST-CENTERS.
01 COST-CENTERS.
 05 RECDR-COUNT.
 10 NO-OF-RECS PIC 9(4).
 10 FILLER PIC X(18).
 05 COST-CNTRS OCCURS 1 TO 99 TIMES
 DEPENDING ON NO-OF-RECS
 ASCENDING KEY IS CC-SIZE-2
 INDEXED BY CC.
 10 CC-SIZE-2 PIC X(2).
 10 CC-SIZE-4 PIC X(4).
 10 FILLER PIC X(16).
FD FINANCIAL-COST-ALLOCATION
 BLOCK CONTAINS 20 RECORDS
 RECORD CONTAINS 26 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS DISK-OUT.
01 DISK-OUT.
 05 DISK-ACCT PIC X(10).
 05 DISK-COST-CNTR PIC X(4).
 05 DISK-COST-CAT PIC X(2).
 05 DISK-AMOUNT PIC S9(8)V99.
01 FIRST-TWO-RECS.
 05 FILLER PIC X.
 05 SORT-SEQUENCE PIC X.
 05 DISK-TITLE PIC X(24).
WORKING-STORAGE SECTION.
 77 SOURCE-CODE PIC X VALUE '1'.
 77 SEARCH-NO PIC XX VALUE SPACES.
 77 LINE-CT PIC 99 COMP VALUE 0.
 77 PAGE-CT PIC 9(4) COMP VALUE 0.
 77 TOTAL-ACCUM PIC S9(8)V99 COMP VALUE 0.
 77 GRAND-TOTAL-ACCUM PIC S9(8)V99 COMP VALUE 0.
 77 TOTAL-SUBTOTAL PIC S9(8)V99 COMP VALUE 0.
 77 GROSS-SUB-TOTAL PIC S9(8)V99 COMP VALUE 0.
 77 GRAND-SUBTOTAL PIC S9(8)V99 COMP VALUE 0.
 77 TOTAL-ALL PIC S9(10)V99 COMP VALUE 0.
 77 GROSS-ALL PIC S9(10)V99 COMP VALUE 0.
 77 GRAND-ALL PIC S9(10)V99 COMP VALUE 0.

```

77 EMPL-NO-HOLD      PIC X(9) VALUE SPACES.
77 DATE-HOLD        PIC X(6) VALUE SPACES.
77 GROSS-HOLD       PIC S9(8)V99- COMP VALUE 0.
77 I                 PIC 9 COMP VALUE 0.
77 MAX-LINE         PIC 99 COMP VALUE 53.
77 FA-NOMINAL       PIC X(10).
77 CC-NOMINAL       PIC X(4).
77 PRINT-EMPL-FLAG  PIC X VALUE '1'.
88 FLAG-ON          VALUE '1'.
88 FLAG-OFF         VALUE '0'.
01 HEADING-1.
    02 FILLER          PIC X(59) VALUE SPACES.
    02 FILLER          PIC X(14) VALUE 'PAYROLL REPORT'.
    02 FILLER          PIC X(60) VALUE SPACES.
01 HEADING-2.
    02 FILLER          PIC X VALUE SPACES.
    02 FILLER          PIC X(13) VALUE 'DATE ENDING: '.
    02 DATE-OUT         PIC X(18) VALUE SPACES.
    02 FILLER          PIC X(33) VALUE SPACES.
    02 FILLER          PIC X(3) VALUE 'FOR'.
    02 FILLER          PIC X(53) VALUE SPACES.
    02 FILLER          PIC X(5) VALUE 'PAGE '.
    02 PAGE-OUT         PIC ZZZ9.
    02 FILLER          PIC X(3) VALUE SPACES.
01 HEADING-3.
    02 FILLER          PIC X(54) VALUE SPACES.
    02 TITLE-OUT        PIC X(24) VALUE SPACES.
    02 FILLER          PIC X(55) VALUE SPACES.
01 HEADING-4.
    02 FILLER          PIC X(5) VALUE SPACES.
    02 FILLER          PIC X(15) VALUE 'EMPLOYEE NUMBER'.
    02 FILLER          PIC X(6) VALUE SPACES.
    02 FILLER          PIC X(4) VALUE 'DATE'.
    02 FILLER          PIC X(4) VALUE SPACES.
    02 FILLER          PIC X(14) VALUE 'ACCOUNT NUMBER'.
    02 FILLER          PIC X(5) VALUE SPACES.
    02 FILLER          PIC X(11) VALUE 'CUST CENTER'.
    02 FILLER          PIC X(6) VALUE SPACES.
    02 FILLER          PIC X(10) VALUE 'TOTAL COST'.
    02 FILLER          PIC X(5) VALUE SPACES.
    02 FILLER          PIC X(9) VALUE 'GROSS PAY'.
    02 FILLER          PIC X(6) VALUE SPACES.
    02 FILLER          PIC X(9) VALUE 'TOTAL PAY'.
    02 FILLER          PIC X(10) VALUE SPACES.
    02 FILLER          PIC X(8) VALUE 'COMMENTS'.
    02 FILLER          PIC X(6) VALUE SPACES.
01 DATA-LINE.
    02 FILLER          PIC X(7) VALUE SPACES.
    02 EMPL-NO-OUT      PIC X(3).

```

```

      05 DASH1      PIC X VALUE '-'.
      05 EMP2-OUT    PIC XX.
      05 DASH2      PIC X VALUE '-'.
      05 EMP3-OUT    PIC X(4).
02 FILLER      PIC X(7) VALUE SPACES.
02 DATE-DATA-OUT PIC X(6).
02 FILLER      PIC X(5) VALUE SPACES.
02 ACCT-OUT    PIC X(10).
02 FILLER      PIC X(10) VALUE SPACES.
02 COST-CTR-OUT PIC X(4).
02 FILLER      PIC X(8) VALUE SPACES.
02 TOTAL-COST-OUT PIC -(8)---.
02 FILLER      PIC X(4) VALUE SPACES.
02 GROSS-PAY-OUT PIC -(8)---.
02 FILLER      PIC X(4) VALUE SPACES.
02 GRAND-PAY-OUT PIC -(8)---.
02 FILLER      PIC X(4) VALUE SPACES.
02 COMMENT     PIC X(20) VALUE SPACES.

01 ERROR-CODE.
02 ERR-CODE OCCURS 3 TIMES PIC 9.

01 ERROR-LIST.
02 ERR-LIST-A.
      05 ERR-LIST1 PIC X(80) VALUE      'ACCOUNT # NOT FOUND
-                               'EMPLOYEE # N-NUMERIC
-                               'COST CTR NOT NUMERIC
-                               'COST CNTR NOT FOUND '.
-                               'TOTAL IN NOT NUMERIC
-                               'GROSS PAY NOTNUMERIC
-                               'GROSS & TOTAL NOT =
-                               'DATE NOT NUMERIC '.

02 ERR-LIST-B REDEFINES ERR-LIST-A.
05 ERR-LIST OCCURS 8 TIMES PIC X(20).

01 SUB-LINE.
02 FILLER      PIC X(5) VALUE SPACES.
02 S-T-COMMENT  PIC X(11) VALUE SPACES.
02 FILLER      PIC X(50) VALUE SPACES.
02 TOTAL-ALL-OUT PIC -(10)---.
02 FILLER      PIC X(2) VALUE SPACES.
02 GROSS-ALL-OUT PIC -(10)---.
02 FILLER      PIC XX VALUE SPACES.
02 GRAND-ALL-OUT PIC -(10)---.
02 FILLER      PIC X(24) VALUE SPACES.

PROCEDURE DIVISION.
OPEN-FILES.
  OPEN INPUT CARD-IN.
  OPEN OUTPUT PRINT-OUT, SORTED-FILE,
    FINANCIAL-COST-ALLOCATION.

READ-TITLE-CARD.
  READ CARD-IN AT END GO TO WRITE-FINAL-TOTALS.
  MOVE SPACES TO DISK-OUT.

```



```

MOVE ACCT-IN TO ACCT-OUT, FA-NOMINAL, DISK-ACCT.
READ FINANCIAL-ACCT-TABFI INVALID KEY
  MOVE 1 TO ERR-CODE (1).
MOVE COST-CTR-IN TO COST-CTR-OUT.
MOVE EMPL-NO-IN TO EMPL-NO-HOLD.
MOVE DATE-DATA-IN TO DATE-HOLD.
EXAMINE INPUT-RECS REPLACING ALL ' ' BY '0'.
IF EMPL-NO-IN NOT NUMERIC MOVE 1 TO ERR-CODE (2).
IF DATE-DATA-IN NOT NUMERIC MOVE 1 TO ERR-CODE (8).
IF FLAG-ON THEN
  MOVE EMP1-IN TO EMP1-OUT,
  MOVE EMP2-IN TO EMP2-OUT,
  MOVE EMP3-IN TO EMP3-OUT,
  MOVE '-' TO DASH1, DASH2,
  MOVE DATE-DATA-IN TO DATE-DATA-OUT,
ELSE MOVE SPACES TO EMPL-NO-OUT, DATE-DATA-OUT.
TEST-COST-CNTR.
  IF COST-CTR-IN NOT NUMERIC MOVE 1 TO ERR-CODE (3)
    MOVE ZEROS TO COST-CTR-IN.
  IF COST-CTR-IN NOT = ZEROS THEN
    PERFORM FIND-COST-CNTR.
  MOVE TOT-COST-IN TO TOTAL-COST-OUT, DISK-AMOUNT.
    ADD TOT-COST-IN TO TOTAL-ACCUM.
  IF GRO-PAY-IN > 0 THEN MOVE GRO-PAY-IN TO GROSS-HOLD.
    MOVE ZEROS TO GROSS-PAY-OUT, GRAND-PAY-OUT.
READ-NEXT-REC.
  PERFORM READ-RECORDS.
CHECK-NEXT-REC.
  IF (EMPL-NO-IN = EMPL-NO-HOLD) AND (DATE-DATA-IN = DATE-HOLD)
    THEN MOVE '0' TO PRINT-EMPL-FLAG,
      GO TO WRITE-LINE-OUT.
MOVE-GROSS-OUT.
  MOVE '1' TO PRINT-EMPL-FLAG.
  MOVE GROSS-HOLD TO GROSS-PAY-OUT.
  IF GROSS-HOLD NOT = TOTAL-ACCUM THEN
    MOVE 1 TO ERR-CODE (7).
  ADD TOTAL-ACCUM TO GRAND-TOTAL-ACCUM, TOTAL-SUBTOTAL.
  ADD GROSS-HOLD TO GROSS-SUB-TOTAL.
  MOVE ZEROS TO TOTAL-ACCUM, GROSS-HOLD.
MOVE-GRAND-OUT.
  IF EMPL-NO-IN NOT = EMPL-NO-HOLD THEN
    MOVE GRAND-TOTAL-ACCUM TO GRAND-PAY-OUT
    ADD GRAND-TOTAL-ACCUM TO GRAND-SUBTOTAL
    MOVE ZEROS TO GRAND-TOTAL-ACCUM.
WRITE-LINE-OUT.
  IF ERROR-CODE NOT = ZERO THEN
    PERFORM MOVE-IN-ERRORS VARYING I
      FROM 1 BY 1 UNTIL ERROR-CODE = ZEROS
    ELSE MOVE DATA-LINE TO PRINT-LINE
    PERFORM PRINT-LINE-OUT

```

```

        WRITE DISK-OUT
        MOVE SPACES TO DISK-OUT.
CHECK-FOR-END-OF-PAGE.
    IF FLAG-ON THEN
        PERFORM PAGE-OVFLOW THRU PAGE-OVFLOW-END.
        GO TO EDIT-RECORDS.
FIND-COST-CNTR.
    IF CC-FIRST-2-DIGITS NOT = ZEROS THEN
        MOVE COST-CTR-IN TO CC-NOMINAL
        READ COST-CNTR-TABFI INVALID KEY
        MOVE 1 TO ERR-CODE (4)
    ELSE PERFORM SEARCH-CC.
        MOVE COST-CTR-IN TO DISK-COST-CNTR.
MOVE-IN-ERRORS.
    IF ERR-CODE (I) NOT = ZEROS THEN
        MOVE ERR-LIST (I) TO COMMENT
        MOVE DATA-LINE TO PRINT-LINE
        PERFORM PRINT-LINE-OUT
        MOVE SPACES TO DATA-LINE
        MOVE ZERO TO ERR-CODE (I).
PAGE-OVFLOW.
    IF LINE-CT NOT < MAX-LINE THEN
        MOVE '1' TO PRINT-EMPL-FLAG
        PERFORM WRITE-SUB-TOTALS
        PERFORM WRITE-HEADINGS.
PAGE-OVFLOW-END.
    EXIT.
SEARCH-CC.
    MOVE CC-SECOND-2-DIGITS TO SEARCH-NO.
    SET CC TO 1.
    SEARCH ALL COST-CNTRS AT END
    MOVE 1 TO ERR-CODE (4)
    WHEN CC-SIZE-2 (CC) = TO SEARCH-NO
        MOVE CC-SIZE-4 (CC) TO COST-CTR-IN.
WRITE-SUB-TOTALS.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PRINT-LINE-OUT.
    MOVE 'SUB TOTAL ' TO S-T-COMMENT.
    MOVE TOTAL-SUBTOTAL TO TOTAL-ALL-OUT.
    MOVE GROSS-SUB-TOTAL TO GROSS-ALL-OUT.
    MOVE GRAND-SUBTOTAL TO GRAND-ALL-OUT.
    MOVE SUB-LINE TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    ADD TOTAL-SUBTOTAL TO TOTAL-ALL.
    ADD GROSS-SUB-TOTAL TO GROSS-ALL.
    ADD GRAND-SUBTOTAL TO GRAND-ALL.
    MOVE ZEROS TO TOTAL-SUBTOTAL, GROSS-SUB-TOTAL,
                      GRAND-SUBTOTAL.
MOVE-SPACES.
    MOVE SPACES TO PRINT-LINE.

```

```
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE ADVANCING 1.  
    ADD 1 TO LINE-CT.  
WRITE-FINAL-TOTALS.  
    MOVE ZEROS TO EMPL-NO-HOLD.  
    PERFORM MOVE-GROSS-OUT THRU WRITE-LINE-OUT.  
    PERFORM WRITE-SUB-TOTALS.  
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
    PERFORM PRINT-LINE-OUT.  
    MOVE 'GRAND TOTAL' TO S-T-COMMENT.  
    MOVE TOTAL-ALL TO TOTAL-ALL-OUT.  
    MOVE GROSS-ALL TO GROSS-ALL-OUT.  
    MOVE GRAND-ALL TO GRAND-ALL-OUT.  
    MOVE SUB-LINE TO PRINT-LINE.  
    PERFORM PRINT-LINE-OUT.  
CLOSE-FILES.  
    CLOSE SORTED-FILE, FINANCIAL-ACCT-TABFI,  
        SPECIAL-ACCT-TABFI, FINANCIAL-COST-ALLOCATION,  
        COST-CNTR-TABFI, PRINT-OUT.  
E-O-J.  
    STOP RUN.
```

```
PGMNAME=WUASCARP
.SYSIN DU *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASCARP.
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FLA 32789.
DATE-WRITTEN. OCT 24 1977.
DATE-COMPILED. OCT 26 1977.
REMARKS. THIS PROGRAM READS THE COST ALLOCATION
        FILES, TOTALS THE COSTS AND PRODUCES
        A COST QLDSQTYDN RUP/RT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT FILE-SURT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
    SELECT FINANCIAL-COST-ALLOCATION
        ASSIGN TO UT-3330-S-FICALFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT FINANCIAL-ACCT-TABFI
        ASSIGN TO DA-3330-I-FATABFI
        ACCESS MODE RANDOM
        NOMINAL KEY IS FA-NOMINAL
        RECORD KEY IS FA-NUMBER.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD PRINT-OUT
    RECORD CONTAINS 81 CHARACTERS
    LABEL RECORDS ARE OMITTED
    DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
    02 FILLER          PIC X.
    02 PRINT-DATA      PIC X(80).
FD FINANCIAL-ACCT-TABFI
    BLOCK CONTAINS 8 RECORDS
    RECORD CONTAINS 58 CHARACTERS
    LABEL RECORDS ARE STANDARD
    DATA RECORD IS FA-REC.
01 FA-REC.
```

```

02 FA-NUMBER      PIC X(10).
02 FA-NAME1       PIC X(20).
02 FA-NAME2       PIC X(20).
02 FILLER         PIC X(8).

SD FILE-SORT
RECORD CONTAINS 26 CHARACTERS
DATA RECORD IS SORT-DATA.

01 SORT-DATA.
02 S-ACCT-NO     PIC X(10).
02 FILLER         PIC X(16).

FD FINANCIAL-COST-ALLOCATION
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-RECS.

01 INPUT-RECS.
02 ACCT-IN        PIC X(10).
02 COST-CNTR-IN   PIC X(4).
02 COST-CAT-IN    PIC X(2).
02 AMOUNT-IN      PIC S9(8)V99.

01 FIRST-TWO-RECS.
02 FILLER         PIC X.
02 S-CODE          PIC 9.
02 TITLE-IN.
05 DATE-IN        PIC X(18).
05 FILLER         PIC X(6).

WORKING-STORAGE SECTION.

77 DEBIT-TOTAL    PIC S9(8)V99 VALUE +0.
77 CREDIT-TOTAL   PIC S9(8)V99 VALUE +0.
77 FA-NOMINAL     PIC X(10) VALUE SPACES.
77 ACCT-TOTAL     PIC S9(8)V99 VALUE ZERO.
77 ACCT-HOLD      PIC X(10) VALUE SPACES.
77 SOURCE-CODE    PIC X.
77 LINE-CT         PIC 99 COMP VALUE 0.
77 PAGE-CT         PIC 9(4) COMP VALUE 1.
77 MAX-LINE        PIC 99 COMP VALUE 54.

01 HEAD1.
02 FILLER         PIC X(13) VALUE SPACES.
02 FILLER         PIC X(29) VALUE
'FINANCIAL COST ALLOCATION OF '.

02 TOP-TITLE      PIC X(29) VALUE SPACES.
02 FILLER         PIC X(9) VALUE SPACES.

01 HEAD2.
02 FILLER         PIC X(6) VALUE 'DATE: '.
02 DATE-OUT        PIC X(18).
02 FILLER         PIC X(14) VALUE SPACES.
02 FILLER         PIC X(3) VALUE 'FOR'.
02 FILLER         PIC X(29) VALUE SPACES.
02 FILLER         PIC X(6) VALUE 'PAGE: '.
02 PAGE-OUT        PIC Z(4).

```

```

01 HEAD3.
  02 FILLER          PIC X(28) VALUE SPACES.
  02 TITLE-OUT       PIC X(24).
  02 FILLER          PIC X(28) VALUE SPACES.

01 HEAD4.
  02 FILLER          PIC X(5)  VALUE SPACES.
  02 FILLER          PIC X(14) VALUE 'ACCOUNT NUMBER'.
  02 FILLER          PIC X(10) VALUE SPACES.
  02 FILLER          PIC X(12) VALUE 'ACCOUNT NAME'.
  02 FILLER          PIC X(11) VALUE SPACES.
  02 FILLER          PIC X(6)  VALUE 'AMOUNT'.
  02 FILLER          PIC X(6)  VALUE SPACES.
  02 FILLER          PIC X(13) VALUE 'ACCOUNT TOTAL'.
  02 FILLER          PIC X(3)  VALUE SPACES.

01 DATA-LINE.
  02 FILLER          PIC X(6).

  02 COMMENT.
    05 FILLER          PIC X(2).
    05 ACCT-OUT        PIC X(10).

  02 FILLER          PIC X(7)  VALUE SPACES.
  02 FA-NAME-OUT      PIC X(20) VALUE SPACES.
  02 FILLER          PIC X(1)  VALUE SPACES.
  02 AMOUNT-OUT       PIC -(8)9.99.
  02 FILLER          PIC X(4)  VALUE SPACES.
  02 ACCT-TOTAL-OUT   PIC -(8)9.99.
  02 FILLER          PIC X(6)  VALUE SPACES.

01 TOP-TITLES.
  02 TL1.
    05 FILLER PIC X(87) VALUE 'PAYROLL REPORT TRANSACTIONS
    -                                     ' REQUISITIONS FROM STOCK
    -                                     'SERVICE CONTRACT TRANSACTIONS'.
    05 FILLER PIC X(87) VALUE ' PAYMENT AUTHORIZATIONS
    -                                     'MISCELLANEOUS TRANSACTIONS
    -                                     'DEPRECIATION VALUES '.

  02 TL2 REDEFINES TL1.
    05 TITLES OCCURS 6 TIMES PIC X(29).

PROCEDURE DIVISION.
OPEN-SORT-FILES.
  OPEN INPUT  FINANCIAL-ACCT-TABFI
    OUTPUT PRINT-OUT.

SORT-RECORDS.
  SORT FILE-SURT ON ASCENDING KEY
    S-ACCT-NO
    USING FINANCIAL-COST-ALLOCATION
    GIVING FINANCIAL-COST-ALLOCATION.

OPEN-FCA-FILE.
  OPEN INPUT  FINANCIAL-COST-ALLOCATION.

READ-FIRST-2-REC.
  PERFORM READ-INPUT.
  MOVE DATE-IN TO DATE-OUT.

```

```

PERFORM READ-INPUT.
MOVE TITLES (S-CODE) TO TOP-TITLE.
MOVE S-CODE TO SOURCE-CODE.
MOVE TITLE-IN TO TITLE-OUT.

WRITE-HEADINGS.
  MOVE 0 TO LINE-CT.
  MOVE PAGE-CT TO PAGE-OUT.
  ADD 1 TO PAGE-CT.
  PERFORM MOVE-SPACES.
  WRITE PRINT-LINE BEFORE NEW-PAGE.
  MOVE HEAD1 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD2 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD3 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
  MOVE HEAD4 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

HOUSEKEEPING.
  MOVE SPACES TO DATA-LINE.
  PERFORM READ-INPUT.

READ-FA-FILE.
  MOVE ACCT-IN TO ACCT-HOLD, ACCT-OUT, FA-NOMINAL.
  READ FINANCIAL-ACCT-TABFI INVALID KEY
    MOVE 'ACCT # INVALID *****' TO FA-NAME1
    MOVE SPACES TO FA-NAME2.
  PERFORM MOVE-OUT-AMOUNTS.
  MOVE FA-NAME1 TO FA-NAME-OUT.
  MOVE DATA-LINE TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE SPACES TO DATA-LINE.
  MOVE FA-NAME2 TO FA-NAME-OUT.
  PERFORM READ-INPUT.
  IF ACCT-IN NOT = TO ACCT-HOLD THEN
    PERFORM WRITE-ACCT-TOTAL
    PERFORM CHECK-LINE-CT
    GO TO READ-FA-FILE.

WRITE-DATA.
  PERFORM MOVE-OUT-AMOUNTS.
  MOVE DATA-LINE TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE SPACES TO DATA-LINE.
  PERFORM CHECK-LINE-CT.
  PERFORM READ-INPUT.
  IF ACCT-IN NOT = TO ACCT-HOLD THEN
    PERFORM WRITE-ACCT-TOTAL
    PERFORM READ-FA-FILE.
  GO TO WRITE-DATA.

```

```
MOVE-DUT-AMOUNTS.  
    MOVE AMOUNT-IN TO AMOUNT-DUT.  
    ADD AMOUNT-IN TO ACCT-TOTAL.  
    IF AMOUNT-IN < 0 THEN  
        ADD AMOUNT-IN TO CREDIT-TOTAL  
    ELSE ADD AMOUNT-IN TO DEBIT-TOTAL.  
CHECK-LINE-CT.  
    IF LINE-CT NOT < MAX-LINE THEN  
        MOVE ACCT-HOLD TO ACCT-DUT  
        PERFORM WRITE-HEADINGS.  
WRITE-ACCT-TOTAL.  
    MOVE ACCT-TOTAL TO ACCT-TOTAL-DUT.  
    MOVE ZERO TO ACCT-TOTAL.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
    MOVE SPACES TO DATA-LINE.  
READ-INPUT.  
    READ FINANCIAL-COST-ALLOCATION AT END  
        GO TO WRITE-FINAL-TOTALS.  
MOVE-SPACES.  
    MOVE SPACES TO PRINT-DATA.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
    ADD 1 TO LINE-CT.  
WRITE-FINAL-TOTALS.  
    PERFORM WRITE-ACCT-TOTAL.  
    MOVE SPACES TO DATA-LINE.  
    MOVE 'DEBIT TOTAL' TO COMMENT.  
    MOVE DEBIT-TOTAL TO AMOUNT-DUT.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    MOVE 'CREDIT TOTAL' TO COMMENT.  
    MOVE CREDIT-TOTAL TO AMOUNT-DUT.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
CLOSE-ALL-FILES.  
    CLOSE FINANCIAL-COST-ALLOCATION, PRINT-OUT,  
        FINANCIAL-ACCT-TABFI.  
E-O-J.  
    STOP RUN.
```

```
PGMNAME=WUASPAAC
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASPAAC.
AUTHOR. ACT SYSTEMS INC.
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FLA 32789.
DATE-WRITTEN. APRIL 1978.
DATE-COMPILED. APRIL 1978.
REMARKS. THIS PROGRAM MERGES ALL EDIT FILES
        INTO ONE TEMP FILE (MERGED-FILE)
        THEN SUMS UP THE COST BY
        ACCOUNT NUMBERS WITHIN COUNTY CODES.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT PRINT-OUT
        ASSIGN TO UT-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT MERGED-FILE
        ASSIGN TO UT-3330-S-MERGEDFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT MERGING-FILE
        ASSIGN TO UT-3330-S-MERGERFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT PAY-ROLL-FILE
        ASSIGN TO UT-3330-S-PAYROLL
        ACCESS MODE IS SEQUENTIAL.
    SELECT REQ-STOCK-FILE
        ASSIGN TO UT-3330-S-REQSTOCK
        ACCESS MODE IS SEQUENTIAL.
    SELECT PAY-AUTH-FILE
        ASSIGN TO UT-3330-S-PAYAUTH
        ACCESS MODE IS SEQUENTIAL.
    SELECT MISC-TRANS-FILE
        ASSIGN TO UT-3330-S-MISCTRAN
        ACCESS MODE IS SEQUENTIAL.
    SELECT SERVICE-CONTRACT-FILE
        ASSIGN TO UT-3330-S-SERVCONT
        ACCESS MODE IS SEQUENTIAL.
    SELECT DEPRECIATION-FILE
        ASSIGN TO UT-3330-S-DEPREC
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
```

FILE SECTION.

FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.

01 PRINT-LINE.
 05 FILLER PIC X.
 05 PRINT-DATA PIC X(132).

FD MERGED-FILE
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-REC.

01 INPUT-REC.
 05 ACCT-NO.
 10 DATE-TITLE-CODE PIC X.
 10 FILLER PIC X(8).
 10 COUNTY-CODE PIC X.
 05 FILLER PIC X(6).
 05 AMOUNT PIC S9(8)V99.

SD MERGING-FILE
RECORD CONTAINS 26 CHARACTERS
DATA RECORD IS MERGE-REC.

01 MERGE-REC.
 05 MERGE-ACCT.
 10 SORT-ACCT PIC X(9).
 10 SORT-COUNTY PIC X.
 05 FILLER PIC X(16).

FD PAY-RD -FILE
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS PAYROLL-REC.

01 PAYROLL-REC PIC X(26).

FD REQ-STOCK-FILE
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS REQ-REC.

01 REQ-REC PIC X(26).

FD PAY-AUTH-FILE
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS PAYAUTH-REC.

01 PAYAUTH-REC PIC X(26).

FD MISC-TRANS-FILE
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD

```

        DATA RECORD IS MISC-REC.
01  MISC-REC          PIC X(26).
FD  SERVICE-CONTRACT-FILE
    BLOCK CONTAINS 20 RECORDS
    RECORD CONTAINS 26 CHARACTERS
    LABEL RECORDS ARE STANDARD
    DATA RECORD IS SERVICE-REC.
C1  SERVICE-REC        PIC X(26).
FD  DEPRECIATION-FILE
    BLOCK CONTAINS 20 RECORDS
    RECORD CONTAINS 26 CHARACTERS
    LABEL RECORDS ARE STANDARD
    DATA RECORD IS DEPRECIATION-REC.
01  DEPRECIATION-REC   PIC X(26).

WORKING-STORAGE SECTION.
77  LINE-CT            PIC 99 VALUE 0.
77  MAX-LINE           PIC 99 VALUE 56.
77  COUNTY-CODE-HOLD   PIC X VALUE SPACES.
77  DEBIT-SUBTOTAL     PIC S9(8)V99 VALUE +0.
77  CREDIT-SUBTOTAL    PIC S9(8)V99 VALUE +0.
77  DEBIT-TOTAL         PIC S9(8)V99 VALUE +0.
77  CREDIT-TOTAL        PIC S9(8)V99 VALUE +0.
77  ACCT-TOTAL          PIC S9(8)V99 VALUE +0.
77  ACCT-HOLD           PIC X(10) VALUE SPACES.
01  HEAD-1.
    05  FILLER            PIC X(10) VALUE SPACES.
    05  FILLER            PIC X(14) VALUE 'ACCOUNT NUMBER'.
    05  FILLER            PIC X(10) VALUE SPACES.
    05  FILLER            PIC X(6)  VALUE 'AMOUNT'.
    05  FILLER            PIC X(92) VALUE SPACES.
01  DATA-LINE.
    05  FILLER            PIC X(12).
    05  ACCT-OUT          PIC X(10).
    05  FILLER            PIC X(7).
    05  AMOUNT-OUT        PIC -(8).---.
    05  FILLER            PIC X(92) VALUE SPACES.

PROCEDURE DIVISION.
MERGE-FILES.
    MERGE MERGING-FILE ON ASCENDING KEY
        MERGE-ACCT
        USING  PAY-ROLL-FILE
              REQ-STOCK-FILE
              PAY-AUTH-FILE
              SERVICE-CONTRACT-FILE
              MISC-TRANS-FILE
              DEPRECIATION-FILE
        GIVING MERGED-FILE.

SORT-FILES.
    SORT    MERGING-FILE ON ASCENDING KEY
          SORT-COUNTY

```

```
        SORT-ACCT
        USING MERGED-FILE
        GIVING MERGED-FILE.
OPEN-FILES.
    OPEN INPUT MERGED-FILE
        OUTPUT PRINT-OUT.
WRITE-HEADINGS.
    MOVE 0 TO LINE-CT.
    PERFORM MOVE-SPACES.
    WRITE PRINT-LINE BEFORE NEW-PAGE.
    MOVE HEAD-1 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PRINT-LINE-OUT 2 TIMES.
    MOVE SPACES TO DATA-LINE.
READ-INPUT.
    READ MERGED-FILE AT END
        GO TO WRITE-FINAL-TOTALS.
VERIFY-DATA.
    IF DATE-TITLE-CODE = SPACES THEN
        GO TO READ-INPUT.
    IF COUNTY-CODE NOT = COUNTY-CODE-HOLD THEN
        PERFORM WRITE-SUBTOTALS THRU TOP-NEW-PAGE.
START-TOTALING.
    IF ACCT-NO NOT = ACCT-HOLD THEN
        PERFORM WRITE-DATA-LINE.
    ADD AMOUNT TO ACCT-TOTAL.
    GO TO READ-INPUT.
WRITE-DATA-LINE.
    MOVE ACCT-HOLD TO ACCT-OUT.
    MOVE ACCT-TOTAL TO AMOUNT-OUT.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE ACCT-NO TO ACCT-HOLD.
    IF ACCT-TOTAL > 0 THEN
        ADD ACCT-TOTAL TO DEBIT-SUBTOTAL
    ELSE ADD ACCT-TOTAL TO CREDIT-SUBTOTAL.
    MOVE ZERO TO ACCT-TOTAL.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    IF LINE-CT > MAX-LINE THEN
        PERFORM WRITE-HEADINGS.
WRITE-SUBTOTALS.
    MOVE ZERO TO LINE-CT.
    PERFORM WRITE-DATA-LINE.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    MOVE 'SUBTOTAL' TO ACCT-OUT.
    MOVE ZERO TO AMOUNT-OUT.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE 'DEBIT' TO ACCT-OUT.
```

```
MOVE DEBIT-SUBTOTAL TO AMOUNT-OUT.  
MOVE DATA-LINE TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE ' CREDIT' TO ACCT-OUT.  
MOVE CREDIT-SUBTOTAL TO AMOUNT-OUT.  
MOVE DATA-LINE TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
ADD DEBIT-SUBTOTAL TO DEBIT-TOTAL.  
ADD CREDIT-SUBTOTAL TO CREDIT-TOTAL.  
MOVE ZEROS TO DEBIT-SUBTOTAL, CREDIT-SUBTOTAL.  
MOVE COUNTY-CODE TO COUNTY-CODE-HOLD.  
TOP-NEW-PAGE.  
    PERFORM WRITE-HEADINGS.  
MOVE-SPACES.  
    MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
    ADD 1 TO LINE-CT.  
WRITE-FINAL-TOTALS.  
    PERFORM WRITE-SUBTOTALS.  
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
    MOVE 'GRANDTOTAL' TO ACCT-OUT.  
    MOVE ZERO TO AMOUNT-OUT.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    MOVE ' DEBIT ' TO ACCT-OUT.  
    MOVE DEBIT-TOTAL TO AMOUNT-OUT.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    MOVE ' CREDIT' TO ACCT-OUT.  
    MOVE CREDIT-TOTAL TO AMOUNT-OUT.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
CLOSE-FILES.  
    CLOSE MERGED-FILE  
        PRINT-OUT.  
E-O-J.  
STOP RUN.
```

PGMNAME=WUASCAP1
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASCAP1.
AUTHOR. ACT SYSTEMS, INC
 SUITE 200
 807 W MORSE BLVD
 WINTER PARK, FL 32789.
DATE-WRITTEN. DECEMBER 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM READS THE FINANCIAL COST ALLOCATION
INTERFACE FILES, EDITS DATA FOR ERRORS AND BUILDS
TWO FILES ONE FOR RESOLVED COST CENTERS AND ANOTHER
FOR UNRESOLVED COST CENTERS.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
 C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT PRINT-OUT
 ASSIGN TO UR-1403-S-PRINTLUT
 ACCESS MODE IS SEQUENTIAL.
 SELECT FINANCIAL-COST-ALLOCATION
 ASSIGN TO UT-3330-S-FICALFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT CROSS-REF-FILE
 ASSIGN TO DA-3330-I-CRTABFI
 ACCESS MODE IS RANDOM
 RECORD KEY IS ACCT-NO
 NOMINAL KEY IS ACCT-NOMINAL.
 SELECT COST-ALLOCATION-RESOLVED
 ASSIGN TO UT-3330-S-RESOCAFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT COST-ALLOCATION-UNRESOLVED
 ASSIGN TO UT-3330-S-UNRSCAFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT CARD-IN
 ASSIGN TO UR-2540R-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
 05 FILLER PIC X.

05 PRINT-DATA PIC X(132).
FD FINANCIAL-COST-ALLOCATION
BLOCK CONTAINS 20 RECORDS
RECORD CONTAINS 26 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-RECS.
01 INPUT-RECS.
05 ACCT-NO-IN.
10 FILLER PIC X(9).
10 COUNTY-CODE PIC X.
05 COST-CNTR-IN PIC X(4).
05 FILLER PIC X(2).
05 AMOUNT-IN PIC S9(8)V99.
01 FIRST-TWO-RECS.
05 FILLER PIC X.
05 SOURCE-CODE-IN PIC 9.
05 TITLE-IN.
10 DATE-IN PIC X(18).
10 FILLER PIC X(6).
FD CROSS-REF-FILE
BLOCK CONTAINS 4 RECORDS
RECORD CONTAINS 265 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CR-REC.
01 CR-REC.
05 ACCT-NO PIC X(10).
05 ALLOC-CODE PIC X.
05 C-CATS.
10 C-CAT OCCURS 2 TIMES PIC XX.
05 NO-OF-CC PIC 99.
05 CC-BLK OCCURS 31 TIMES.
10 COST-CNTR PIC X(4).
10 MATCH-CODE PIC X(4).
01 CR-REC-1.
05 FILLER PIC X(17).
05 CC-BLK-1 OCCURS 31 TIMES.
10 FILLER PIC X(4).
10 PERCENT PIC V9999.
FD COST-ALLOCATION-RESOLVED
BLOCK CONTAINS 5 RECORDS
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS RES-REC.
01 RES-REC.
05 RES-REC-DATA PIC X(30).
05 FILLER PIC X(50).
FD COST-ALLOCATION-UNRESOLVED
BLOCK CONTAINS 5 RECORDS
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE STANDARD

DATA RECORD IS UNRES-REC.
 01 UNRES-REC.
 05 UNRES-REC-DATA PIC X(30).
 05 FILLER PIC X(50).
 FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS INPUT-CARD.
 01 INPUT-CARD.
 05 COUNTY-CODE-IN PIC X.
 05 FILLER PIC X(79).
 WORKING-STORAGE SECTION.
 77 LINE-CT PIC 99 COMP VALUE 0.
 77 PAGE-CT PIC 9(4) COMP VALUE 1.
 77 MAX-LINE PIC 99 COMP VALUE 56.
 77 SOURCE-CODE PIC 9.
 77 ACCT-NOMINAL PIC X(10) VALUE ZERO.
 77 I PIC 99.
 77 PERCENT-3401 PIC V9(4) VALUE ZERO.
 77 PERCENT-2101 PIC V9(4) VALUE ZERO.
 77 HOLD-AMT PIC S9(8)V99.
 77 COST-CNTR-3401 PIC X(4) VALUE '3401'.
 77 COST-CNTR-2101 PIC X(4) VALUE '2101'.
 77 MATCH-3401 PIC X(4).
 77 MATCH-2101 PIC X(4).
 77 RES-FLAG PIC 9 VALUE ZERO.
 77 COUNTY-CODE-HOLD PIC X VALUE SPACES.
 01 ERROR-LIST.
 05 ERR1 PIC X(20) VALUE 'INVALID COST CENTER '.
 01 FILE-NAMES.
 05 FILENAMES1.
 10 FILLER PIC X(72) VALUE
 - 'PAYROLL
 - 'REQ OF STOCK
 - 'SERVICE
 - 'PAYMENT AUTH
 - 'MISC TRANS
 - 'DEPRECIATION'.
 05 FILENAMES REDEFINES FILENAMES1.
 10 FILE-NAME OCCURS 6 TIMES PIC X(12).
 01 HEAD-1.
 05 FILLER PIC X(44) VALUE SPACES.
 05 FILLER PIC X(34) VALUE 'INITIAL COST ALLOCATION
 - REPORT OF '.
 05 FILE-NAME-OUT PIC X(12).
 05 FILLER PIC X(42) VALUE SPACES.
 01 HEAD-2.
 05 FILLER PIC X(13) VALUE 'DATE ENDING: '.
 05 DATE-OUT PIC X(18).
 05 FILLER PIC X(33) VALUE SPACES.
 05 FILLER PIC X(3) VALUE 'FOR'.

```

05 FILLER          PIC X(55) VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'PAGE: '.
05 PAGE-OUT        PIC Z(4).

01 HEAD-3.
05 FILLER          PIC X(54) VALUE SPACES.
05 TITLE-OUT        PIC X(24).
05 FILLER          PIC X(54) VALUE SPACES.

01 HEAD-4.
05 FILLER          PIC X(15) VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'INPUT DATA'.
05 FILLER          PIC X(53) VALUE SPACES.
05 FILLER          PIC X(19) VALUE 'RESOLVED INPUT DATA'.
05 FILLER          PIC X(35) VALUE SPACES.

01 HEAD-5.
05 FILLER          PIC X(7)  VALUE 'ACCT #'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'COST/CNTR '.
05 FILLER          PIC X(8)  VALUE 'COST/CAT'.
05 FILLER          PIC X(6)  VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'AMOUNT'.
05 FILLER          PIC X(12) VALUE SPACES.
05 FILLER          PIC X(9)  VALUE 'COST-CNTR'.
05 FILLER          PIC X(3)  VALUE SPACES.
05 FILLER          PIC X(8)  VALUE 'COST-CAT'.
05 FILLER          PIC X(2)  VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'MATCH/CODE'.
05 FILLER          PIC X(5)  VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'AMOUNT '.
05 FILLER          PIC X(8)  VALUE 'RESOLVED'.
05 FILLER          PIC X(10) VALUE SPACES.
05 FILLER          PIC X(7)  VALUE 'COMMENT'.
05 FILLER          PIC X(7)  VALUE SPACES.

01 DATA-LINE.
05 ACCT-ND-OUT    PIC X(10).
05 FILLER          PIC X(3).
05 COST-CNTR-OUT-1 PIC X(4).
05 FILLER          PIC X(13).
05 AMOUNT-OUT-1   PIC -(8)---.
05 FILLER          PIC X(14).
05 COST-CNTR-OUT-2 PIC X(4).
05 FILLER          PIC X(9).
05 COST-CAT-OUT   PIC XX.
05 FILLER          PIC X(8).
05 MATCH-CODE-OUT  PIC X(4).
05 FILLER          PIC X(3).
05 AMOUNT-OUT-2   PIC -(8)---.
05 FILLER          PIC X(7).
05 RES-OUT         PIC XX.
05 FILLER          PIC X(7).
05 COMMENT-OUT    PIC X(20).

```

```

01 DISK-RECORD.
  05 DISK-ACCT-NO      PIC X(10).
  05 DISK-COST-CNTR.
    10 DISK-ALLOC-CODE PIC X.
    10 FILLER        PIC X(3).
  05 DISK-COST-CAT   PIC XX.
  05 DISK-MATCH-CODE PIC X(4).
  05 DISK-AMT OCCURS 6 TIMES PIC S9(8)V99.

PROCEDURE DIVISION.
OPEN-FILES.
  OPEN OUTPUT PRINT-OUT,
    COST-ALLOCATION-RESERVED,
    COST-ALLOCATION-UNRESERVED,
    INPUT FINANCIAL-COST-ALLOCATION,
    CARD-IN
    CROSS-REF-FILE.

HOUSE-KEEP.
  MOVE SPACES TO RES-REC, UNRES-REC, DATA-LINE.

*
*  INVALID OPTION CARD OR NO OPTION CARD DEFAULTS
*  TO A BLANK COUNTY CODE
*
READ-OPTION-CARD.
  READ CARD-IN AT END GO TO FILL-HEADINGS.
  IF COUNTY-CODE-IN = 'C' THEN
    MOVE 'C' TO COUNTY-CODE-HOLD.

FILL-HEADINGS.
  READ FINANCIAL-COST-ALLOCATION AT END
  GO TO CLOSE-FILES.
  MOVE DATE-IN TO DATE-OUT.
  READ FINANCIAL-COST-ALLOCATION AT END
  GO TO CLOSE-FILES.
  MOVE TITLE-IN TO TITLE-OUT.
  MOVE SOURCE-CODE-IN TO SOURCE-CODE.
  MOVE FILE-NAME (SOURCE-CODE) TO FILE-NAME-OUT.

WRITE-HEADINGS.
  MOVE PAGE-CT TO PAGE-OUT.
  ADD 1 TU PAGE-CT.
  MOVE 0 TO LINE-CT.
  PERFORM MOVE-SPACES.
  WRITE PRINT-LINE BEFORE NEW-PAGE.
  MOVE HEAD-1 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD-2 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD-3 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES.
  PERFORM PRINT-LINE-OUT 2 TIMES.
  MOVE HEAD-4 TO PRINT-DATA.

```

```

PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES.
PERFORM PRINT-LINE-OUT 2 TIMES.
MOVE HEAD-5 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
READ-INPUT.
  READ FINANCIAL-COST-ALLOCATION AT END
    GO TO CLOSE-FILES.
  IF AMOUNT-IN = 0 THEN
    GO TO READ-INPUT.
  IF COUNTY-CODE NOT = TO COUNTY-CODE-HOLD THEN
    GO TO READ-INPUT.
START-EDIT.
  MOVE ACCT-NO-IN TO ACCT-NOMINAL.
  READ CROSS-REF-FILE INVALID KEY
    MOVE SPACES TO DATA-LINE
    MOVE 'ACCT # NOT FOUND' TO COMMENT-OUT
    MOVE ACCT-NO-IN TO ACCT-NO-OUT
    MOVE DATA-LINE TO PRINT-DATA
    PERFORM PRINT-LINE-OUT
    MOVE SPACES TO DATA-LINE
    PERFORM PAGE-OVFLOW
    GO TO READ-INPUT.
CHECK-4-VALID-COST-CNTR.
  EXAMINE COST-CNTR-IN REPLACING ALL ' ' BY '0'.
  IF (COST-CNTR-IN NOT = ZERO) AND (ALLOC-CODE NOT = 'H') THEN
    PERFORM CHECK-COST-CENTERS
  ELSE GO TO CHECK-ALLOC-CODE.
  IF RES-FLAG = 0 THEN
    MOVE 'I' TO ALLOC-CODE
    GO TO RESOLVED-CC.
CHECK-ALLOC-CODE.
  IF ALLOC-CODE = 'A' OR 'H' OR 'I' OR 'J' THEN
    GO TO RESOLVED-CC
  ELSE GO TO UN-RESOLVED-CC.
RESOLVED-CC.
  MOVE ACCT-NO-IN TO DISK-ACCT-NO, ACCT-NO-OUT.
  IF COST-CNTR-IN NOT = ZEROES THEN
    MOVE COST-CNTR-IN TO COST-CNTR-OUT-1
  ELSE MOVE SPACES TO COST-CNTR-OUT-1.
  PERFORM CHECK-COST-CAT.
  MOVE AMOUNT-IN TO AMOUNT-OUT-1.
  IF ALLOC-CODE NOT = 'H' THEN
    GO TO CHECK-A-I-J.
  IF PERCENT-3401 = ZERO THEN
    PERFORM GET-H-PERCENTAGES.
  MULTIPLY AMOUNT-IN BY PERCENT-3401 GIVING HOLD-AMT.
  MOVE HOLD-AMT TO DISK-AMT (1), AMOUNT-OUT-2.
  MOVE MATCH-3401 TO DISK-MATCH-CODE, MATCH-CODE-OUT.

```

```

MOVE COST-CNTR-3401 TO DISK-COST-CNTR, COST-CNTR-OUT-2.
PERFORM WRITE-RES-REC.
MOVE SPACES TO ACCT-NO-OUT, COST-CNTR-OUT-1.
MOVE ACCT-NO-IN TO DISK-ACCT-NO.
MOVE ZEROS TO AMOUNT-OUT-1.
MULTIPLY AMOUNT-IN BY PERCENT-2101 GIVING HOLD-AMT.
MOVE HOLD-AMT TO DISK-AMT (1), AMOUNT-OUT-2.
MOVE MATCH-2101 TO DISK-MATCH-CODE, MATCH-CODE-OUT.
MOVE COST-CNTR-2101 TO DISK-COST-CNTR, COST-CNTR-OUT-2.
PERFORM CHECK-COST-CAT.
PERFORM WRITE-RES-REC.
GO TO READ-INPUT.

CHECK-A-I-J.
IF (ALLOC-CODE = 'I') AND (COST-CNTR-IN NOT = SPACES OR ZERO)
  THEN MOVE COST-CNTR-IN TO COST-CNTR (1)
    MOVE SPACES TO COMMENT-OUT.
MOVE AMOUNT-IN TO DISK-AMT (1), AMOUNT-OUT-2.
MOVE COST-CNTR (1) TO DISK-COST-CNTR, COST-CNTR-OUT-2.
MOVE MATCH-CODE (1) TO DISK-MATCH-CODE, MATCH-CODE-OUT.
PERFORM WRITE-RES-REC.
GO TO READ-INPUT.

WRITE-RES-REC.
MOVE DATA-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE DISK-RECORD TO RES-REC-DATA.
WRITE RES-REC.
MOVE SPACES TO DISK-RECORD, DATA-LINE.
PERFORM PAGE-OVFLOW.

UN-RESOLVED-CC.
MOVE ACCT-NO-IN TO DISK-ACCT-NO, ACCT-NO-OUT.
IF COST-CNTR-IN NOT = ZERODES THEN
  MOVE COST-CNTR-IN TO COST-CNTR-OUT-1
ELSE MOVE SPACES TO COST-CNTR-OUT-1.
PERFORM CHECK-COST-CAT.
MOVE AMOUNT-IN TO AMOUNT-OUT-1, AMOUNT-OUT-2, DISK-AMT (1).
MOVE SPACES TO DISK-MATCH-CODE, MATCH-CODE-OUT.
MOVE ALLOC-CODE TO DISK-ALLOC-CODE.
MOVE 'NO' TO RES-OUT
MOVE DATA-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE DISK-RECORD TO UNRES-REC-DATA.
WRITE UNRES-REC.
MOVE SPACES TO DISK-RECORD, DATA-LINE.
PERFORM PAGE-OVFLOW.
GO TO READ-INPUT.

PAGE-OVFLOW.
IF LINE-CT NOT < MAX-LINE THEN
  PERFORM WRITE-HEADINGS.

MOVE-SPACES.
MOVE SPACES TO PRINT-LINE.

```

```

PRINT-LINE-OUT.
  WRITE PRINT-LINE BEFORE 1.
  ADD 1 TO LINE-CT.
CHECK-COST-CAT.
  IF SOURCE-CODE = 1 THEN
    MOVE C-CAT (2) TO DISK-COST-CAT, COST-CAT-OUT
  ELSE MOVE C-CAT (1) TO DISK-COST-CAT, COST-CAT-OUT.
CHECK-COST-CENTERS.
  MOVE 0 TO RES-FLAG.
  PERFORM CHECK-COST-CNTR THRU CHECK-COST-CNTR-END
    VARYING I FROM 1 BY 1 UNTIL
      I > NO-OF-CC.
CHECK-COST-CNTR.
  IF COST-CNTR (I) = TO COST-CNTR-IN THEN
    ADD NO-OF-CC TO I
    GO TO CHECK-COST-CNTR-END.
  IF I = TO NO-OF-CC THEN
    MOVE 1 TO RES-FLAG.
CHECK-COST-CNTR-END.
  EXIT.
GET-H-PERCENTAGES.
  IF COST-CNTR (1) = '3401' THEN
    MOVE MATCH-CODE (1) TO MATCH-3401
    MOVE PERCENT (2) TO PERCENT-3401
    MOVE MATCH-CODE (3) TO MATCH-2101
    MOVE PERCENT (4) TO PERCENT-2101
  ELSE
    MOVE MATCH-CODE (1) TO MATCH-2101
    MOVE PERCENT (2) TO PERCENT-2101
    MOVE MATCH-CODE (3) TO MATCH-3401
    MOVE PERCENT (4) TO PERCENT-3401.
CLOSE-FILES.
  CLOSE FINANCIAL-COST-ALLOCATION, CROSS-REF-FILE,
    COST-ALLOCATION-RESOLVED, COST-ALLOCATION-UNRESOLVED,
    CARD-IN, PRINT-OUT.
E-O-J.
  STOP RUN.

```

PGMNAME=WUASCAP2

.SYSIN DU *

ID DIVISION.

PROGRAM-ID. WUASCAP2

AUTHOR. ACT SYSTEMS, INC

SUITE 200

807 W MORSE BLVD

WINTER PARK, FL 32789.

DATE-WRITTEN. DEC 14, 1977.

DATE-COMPILED.

REMARKS. THIS PROGRAM READS THE RESOLVED FILE
(RESOCIFI) AND BUILDS A TABLE OF COST
CENTER VALUES TO BE USED IN CAP3.

ENVIRONMENT DIVISION.

CONFIGURATION SECTION.

SOURCE-COMPUTER. IBM-370.

OBJECT-COMPUTER. IBM-370.

SPECIAL-NAMES.

C01 IS NEW-PAGE.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT PRINT-OUT

ASSIGN TO UR-1403-S-PRINTOUT

ACCESS MODE IS SEQUENTIAL.

SELECT COST-ALLOCATION-RESOLVED

ASSIGN TO UT-3330-S-RESOCIFI

ACCESS MODE IS SEQUENTIAL.

SELECT COST-ALLOCATION-RESOLVED-COSTS

ASSIGN TO UT-3330-S-RECTABFI

ACCESS MODE IS SEQUENTIAL.

SELECT FILE-SORT

ASSIGN TO UT-3330-S-FILESORT

ACCESS MODE IS SEQUENTIAL.

SELECT CARD-IN

ASSIGN TO UR-2540R-S-CARDIN

ACCESS MODE IS SEQUENTIAL.

DATA DIVISION.

FILE SECTION.

FD PRINT-OUT

RECORD CONTAINS 133 CHARACTERS

LABEL RECORDS ARE OMITTED

DATA RECORD IS PRINT-LINE.

01 PRINT-LINE.

05 FILLER PIC X.

05 PRINT-DATA PIC X(80).

05 FILLER PIC X(52).

FD COST-ALLOCATION-RESOLVED

BLOCK CONTAINS 5 RECORDS

RECORD CONTAINS 80 CHARACTERS

LABEL RECORDS ARE STANDARD

DATA RECORD IS RES-REC.

01 RES-REC.

 05 FILLER PIC X(10).

 05 COST-CNTR-IN PIC X(4).

 05 FILLER PIC X(6).

 05 AMOUNT OCCURS 6 TIMES PIC S9(8)V99.

FD COST-ALLOCATION-RESOLVED-COSTS

 BLOCK CONTAINS 0 RECORDS

 LABEL RECORDS ARE STANDARD

 DATA RECORD IS RES-COSTS.

01 RES-COSTS.

 05 RECORD-COUNT.

 10 NO-OF-RECS PIC 9(2).

 10 FILLER PIC X(12).

 05 RES-COST-REC OCCURS 1 TO 99 TIMES

 DEPENDING ON NO-OF-RECS

 ASCENDING KEY IS COST-CNTR

 INDEXED BY CC.

 10 COST-CNTR PIC X(4).

 10 CC-AMOUNT PIC S9(8)V99.

SD FILE-SORT

 RECORD CONTAINS 80 CHARACTERS

 DATA RECORD IS SORT-INPUT.

01 SORT-INPUT.

 05 FILLER PIC X(10).

 05 S-COST-CNTR PIC X(4).

 05 FILLER PIC X(66).

FD CARD-IN

 RECORD CONTAINS 80 CHARACTERS

 LABEL RECORDS ARE OMITTED

 DATA RECORD IS INPUT-CARD.

01 INPUT-CARD.

 05 DATE-IN PIC X(18).

 05 TITLE-IN PIC X(24).

 05 FILLER PIC X(38).

WORKING-STORAGE SECTION.

77 LINE-CT PIC 9(2) COMP VALUE 0.

77 PAGE-CT PIC 9(4) COMP VALUE 0.

77 MAX-LINE PIC 9(2) COMP VALUE 56.

77 COST-CNTR-HOLD PIC X(4) VALUE ZERO.

77 AMOUNT-ACCUM PIC S9(8)V99 VALUE 0.

77 COST-CNTR-CT PIC 99 VALUE 0.

01 HEAD-1.

 05 FILLER PIC X(36) VALUE SPACES.

 05 FILLER PIC X(26) VALUE 'TOTAL COST BY COST CENTERS'.

 05 FILLER PIC X(18) VALUE SPACES.

01 HEAD-2.

 05 FILLER PIC X(13) VALUE 'DATE ENDING: '.

 05 DATE-OUT PIC X(18).

 05 FILLER PIC X(17) VALUE SPACES.

```

05 FILLER          PIC X(3) VALUE 'FOR'.
05 FILLER          PIC X(19) VALUE SPACES.
05 FILLER          PIC X(6) VALUE 'PAGE: '.
05 PAGE-OUT        PIC ZZZ9.

01 HEAD-3.
05 FILLER          PIC X(37) VALUE SPACES.
05 TITLE-OUT        PIC X(24).
05 FILLER          PIC X(19) VALUE SPACES.

01 HEAD-4.
05 FILLER          PIC X(23) VALUE SPACES.
05 FILLER          PIC X(11) VALUE 'COST CENTER'.
05 FILLER          PIC X(26) VALUE SPACES.
05 FILLER          PIC X(5) VALUE 'VALUE'.
05 FILLER          PIC X(15) VALUE SPACES.

01 DATA-LINE.
05 FILLER          PIC X(27) VALUE SPACES.
05 COST-CNTR-OUT   PIC Z(4).
05 FILLER          PIC X(23).
05 AMOUNT-OUT      PIC Z(8).99CR.
05 FILLER          PIC X(13).

PROCEDURE DIVISION.

OPEN-IO-FILES.
  OPEN INPUT CARD-IN
    OUTPUT PRINT-OUT
      COST-ALLOCATION-RESOLVED-COSTS.

SORT-RES-FILE.
  SORT FILE-SORT ON ASCENDING KEY
    S-COST-CNTR
  USING COST-ALLOCATION-RESOLVED
    GIVING COST-ALLOCATION-RESOLVED.

OPEN-CAR-FILE.
  OPEN INPUT COST-ALLOCATION-RESOLVED.

HOUSE-KEEP.
  PERFORM MOVE-SPACES.

READ-CARD-IN.
  READ CARD-IN AT END
    MOVE 'NO DATE CARD ** JOB ABORTED' TO PRINT-DATA
    PERFORM PRINT-LINE-OUT
    GO TO CLOSE-IO-FILES.

MOVE DATE-IN TO DATE-OUT.
MOVE TITLE-IN TO TITLE-OUT.

WRITE-HEADINGS.
  ADD 1 TO PAGE-CT.
  MOVE 0 TO LINE-CT.
  MOVE PAGE-CT TO PAGE-OUT
  WRITE PRINT-LINE BEFORE NEW-PAGE.
  MOVE HEAD-1 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD-2 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.

```

```
MOVE HEAD-3 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES.  
PERFORM PRINT-LINE-OUT 2 TIMES.  
MOVE HEAD-4 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
SET-POINTER.  
    SET CC TO 1.  
READ-INPUT.  
    READ COST-ALLOCATION-RESOLVED AT END  
        GO TO WRITE-COST-TABLE.  
START-TOTALING.  
    MOVE COST-CNTR-IN TO COST-CNTR (CC), COST-CNTR-OUT,  
        COST-CNTR-HOLD.  
ADD-AMOUNTS.  
    ADD AMOUNT (1) TO AMOUNT-ACCUM  
    PERFORM READ-INPUT.  
    IF COST-CNTR-IN NOT = COST-CNTR-HOLD THEN  
        PERFORM WRITE-COST-CNTR  
        GO TO START-TOTALING  
    ELSE GO TO ADD-AMOUNTS.  
WRITE-COST-CNTR.  
    MOVE AMOUNT-ACCUM TO AMOUNT-OUT, CC-AMOUNT (CC).  
    MOVE ZERO TO AMOUNT-ACCUM.  
    SET CC UP BY 1.  
    ADD 1 TO COST-CNTR-CT.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    IF LINE-CT NOT < MAX-LINE THEN  
        PERFORM WRITE-HEADINGS.  
MOVE-SPACES.  
    MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
    ADD 1 TO LINE-CT.  
WRITE-COST-TABLE.  
    MOVE COST-CNTR-CT TO NO-OF-RECS.  
    PERFORM WRITE-COST-CNTR.  
    WRITE RES-COSTS.  
CLOSE-IO-FILES.  
    CLOSE COST-ALLOCATION-RESOLVED  
        COST-ALLOCATION-RESOLVED-COSTS  
        CARD-IN  
        PRINT-OUT.  
E-O-J.  
    STOP RUN.
```

PGMNAME=WUASCAP3
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASCAP3.
AUTHOR. ACT SYSTEMS, INC
 SUITE 200
 807 W MORSE BLVD
 WINTER PARK, FL 32789.
DATE-WRITTEN. DEC 15, 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM READS THE UNRESOLVED FILE
 (UNRSCAFI) AND USING THE TABLE OF COST
 CENTER VALUES, RESOLVES THE FILE AND
 ADDS THE RESOLVED RECORDS TO THE RESOLVED
 FILE (RESUCAFI).
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
 C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT CARD-IN
 ASSIGN TO UR-2540R-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
 SELECT PRINT-OUT
 ASSIGN TO UR-1403-S-PRINTOUT
 ACCESS MODE IS SEQUENTIAL.
 SELECT COST-ALLOCATION-RESOLVED
 ASSIGN TO UT-3330-S-RESOCAFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT COST-ALLOCATION-UNRESOLVED
 ASSIGN TO UT-3330-S-UNRSCAFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT COST-ALLOCATION-RESOLVED-CUSTS
 ASSIGN TO UT-3330-S-RECTABFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT CROSS-REF-FILE
 ASSIGN TO DA-3330-I-CRTABFI
 ACCESS MODE IS RANDOM
 RECORD KEY IS CRF-ACCT-NO
 NOMINAL KEY IS ACCT-NOMINAL.
 SELECT FILE-SORT
 ASSIGN TO UT-3330-S-FILESORT
 ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS

LABEL RECORDS ARE OMITTED
DATA RECORD IS INPUT-CARD.

01 INPUT-CARD.
 05 DATE-IN PIC X(18).
 05 TITLE-IN PIC X(24).
 05 FILLER PIC X(38).

FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.

01 PRINT-LINE.
 05 FILLER PIC X.
 05 PRINT-DATA PIC X(132).

FD COST-ALLOCATION-RESOLVED
 BLOCK CONTAINS 5 RECORDS
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS RES-REC.

01 RES-REC.
 05 RES-ACCT-NO PIC X(10).
 05 RES-COST-CNTR PIC X(4).
 05 RES-COST-CAT PIC XX.
 05 RES-MATCH-CODE PIC X(4).
 05 RES-AMTS OCCURS 6 TIMES PIC S9(8)V99.

FD COST-ALLOCATION-UNRESOLVED
 BLOCK CONTAINS 5 RECORDS
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS UNRES-REC.

01 UNRES-REC.
 05 ACCT-NO PIC X(10).
 05 ALLOC-CODE PIC X.
 05 FILLER PIC X(3).
 05 COST-CAT PIC XX.
 05 MATCH-CODE PIC 9(4).
 05 AMTS OCCURS 6 TIMES PIC S9(8)V99.

FD COST-ALLOCATION-RESOLVED-COSTS
 BLOCK CONTAINS 0 RECORDS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS RES-COST-RECORD.

01 RES-COST-RECORD.
 05 RECORD-COUNT.
 10 NO-OF-RECS PIC 9(2).
 10 FILLER PIC X(12).
 05 RES-COSTS OCCURS 1 TO 99 TIMES
 DEPENDING ON NO-OF-RECS
 ASCENDING KEY IS COST-CNTR
 INDEXED BY CC.
 10 COST-CNTR PIC X(4).
 10 CC-AMOUNT PIC S9(8)V99.

FD CROSS-REF-FILE
 BLOCK CONTAINS 4 RECORDS
 RECORD CONTAINS 265 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS CR-REC.

01 CR-REC.
 05 CRF-ACCT-NO PIC X(10).
 05 CRF-ALLOC-CODE PIC X.
 05 C-CATS.
 10 C-CAT OCCURS 2 TIMES PIC XX.
 05 NO-OF-CC PIC 99.
 05 CC-BLK OCCURS 31 TIMES.
 10 CRF-COST-CNTR PIC X(4).
 10 CRF-MATCH-CODE PIC X(4).

SD FILE-SORT
 RECORD CONTAINS 80 CHARACTERS
 DATA RECORD IS SORT-INPUT.

01 SORT-INPUT.
 05 S-ACCT-NO PIC X(10).
 05 FILLER PIC X(70).

WORKING-STORAGE SECTION.

77 LINE-CT	PIC 99 COMP VALUE 0.
77 PAGE-CT	PIC 9(4) COMP VALUE 0.
77 MAX-LINE	PIC 9(2) COMP VALUE 56.
77 ACCT-NOMINAL	PIC X(10) VALUE ZERO.
77 AMT-HOLD	PIC S9(8)V99.
77 AMT-HOLD-OVFLOW	PIC S9(8)V99.
77 ACCT-HOLD	PIC X(10).
77 ALLOC-CODE-HOLD	PIC X.
77 I	PIC 99.
77 J	PIC 9 COMP.
77 SLASH	PIC X VALUE '/'.
77 COMMON-AMOUNTS-ACCUM	PIC S9(8)V99.
77 SEARCH-NO	PIC X(4).
77 PERCENT-100	PIC S9V99 VALUE +1.
77 SUBTOTAL	PIC S9(8)V99 VALUE +0.
01 HEAD-1.	
05 FILLER	PIC X(50) VALUE SPACES.
05 FILLER	PIC X(32) VALUE 'UNRESOLVED COST CENTERS ' RESOLVED'.
-	
05 FILLER	PIC X(50) VALUE SPACES.
01 HEAD-2.	
05 FILLER	PIC X(13) VALUE 'DATE ENDING: '.
05 DATE-OUT	PIC X(18).
05 FILLER	PIC X(33) VALUE SPACES.
05 FILLER	PIC X(3) VALUE 'FOR'.
05 FILLER	PIC X(55) VALUE SPACES.
05 FILLER	PIC X(6) VALUE 'PAGE: '.
05 PAGE-OUT	PIC Z(4).
01 HEAD-3.	

```

05 FILLER          PIC X(54) VALUE SPACES.
05 TITLE-OUT       PIC X(24).
05 FILLER          PIC X(54) VALUE SPACES.
01 HEAD-4.
05 FILLER          PIC X(8)  VALUE 'ACCOUNT'.
05 FILLER          PIC X(5)  VALUE SPACES.
05 FILLER          PIC X(4)  VALUE 'COST'.
05 FILLER          PIC X(3)  VALUE SPACES.
05 FILLER          PIC X(5)  VALUE 'ALLOC'.
05 FILLER          PIC X(3)  VALUE SPACES.
05 FILLER          PIC X(5)  VALUE 'TOTAL'.
05 FILLER          PIC X(17) VALUE SPACES.
05 FILLER          PIC X(41) VALUE 'RESOLVED AMOUNT ALLOCAT
-                           ED COST CENTERS(S)'.
05 FILLER          PIC X(41) VALUE SPACES.
01 HEAD-5.
05 FILLER          PIC X(7)  VALUE 'NUMBER'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(8)  VALUE 'CATEGORY'.
05 FILLER          PIC X(2)  VALUE SPACES.
05 FILLER          PIC X(4)  VALUE 'CODE'.
05 FILLER          PIC X(3)  VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'AMOUNT'.
05 FILLER          PIC X(98) VALUE SPACES.
01 DATA-LINE.
05 ACCT-NO-OUT    PIC X(10).
05 FILLER          PIC X(4).
05 COST-CAT-OUT   PIC X(2).
05 FILLER          PIC X(6).
05 ALLOC-CODE-OUT PIC X.
05 AMOUNT-OUT     PIC -(7)9.99.
05 RES-DATA OCCURS 6 TIMES.
10 AMT-OUT         PIC -(7)9.99.
10 SLASH-OUT       PIC X.
10 COST-CNTR-OUT  PIC X(4).
01 B-PERCENTAGES.
05 B-PERCENT OCCURS 15 TIMES PIC 9V99.
01 C-PERCENTAGES.
05 C-PERCENT OCCURS 15 TIMES PIC 9V99.
01 D-PERCENTAGES.
05 D-PERCENT OCCURS 15 TIMES PIC 9V99.
01 E-PERCENTAGES.
05 E-PERCENT OCCURS 15 TIMES PIC 9V99.
01 F-PERCENTAGES.
05 F-PERCENT OCCURS 15 TIMES PIC 9V99.
01 G-PERCENTAGES.
05 G-PERCENT OCCURS 15 TIMES PIC 9V99.
01 COMMON-PERCENTAGES.
05 COMMON-PERCENT OCCURS 15 TIMES PIC 9V99.
01 COMMON-AMOUNTS.

```

05 COMMON-AMT OCCURS 13 TIMES PIC S9(8)V99.
PROCEDURE DIVISION.
OPEN-IO-FILES.
OPEN INPUT CARD-IN,
COST-ALLOCATION-RESOLVED-COSTS,
CROSS-REF-FILE,
OUTPUT COST-ALLOCATION-RESOLVED,
PRINT-OUT.
SORT-UNRES-FILE.
SORT FILE-SORT ON ASCENDING KEY
S-ACCT-NO
USING COST-ALLOCATION-UNRESOLVED
GIVING COST-ALLOCATION-UNRESOLVED.
OPEN-CAU-FILE.
OPEN INPUT COST-ALLOCATION-UNRESOLVED.
FILL-HEADING.
READ CARD-IN AT END MOVE
'*** NO DATE/TITLE CARD *** JOB ABORTED ***'
TO PRINT-DATA,
PERFORM PRINT-LINE-OUT
GO TO CLOSE-IO-FILES.
MOVE DATE-IN TO DATE-OUT.
MOVE TITLE-IN TO TITLE-OUT.
READ-COST-TABLE-IN.
READ COST-ALLOCATION-RESOLVED-COSTS AT END MOVE
'*** COST TABLE CONTAINS NO DATA ** JOB ABORTED ***'
TO PRINT-DATA
PERFORM PRINT-LINE-OUT
GO TO CLOSE-IO-FILES.
WRITE-HEADINGS.
ADD 1 TO PAGE-CT.
MOVE PAGE-CT TO PAGE-OUT.
MOVE ZERO TO LINE-CT.
PERFORM MOVE-SPACES.
WRITE PRINT-LINE BEFORE NEW-PAGE.
MOVE HEAD-1 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEAD-2 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEAD-3 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES.
PERFORM PRINT-LINE-OUT 2 TIMES.
MOVE HEAD-4 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEAD-5 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
HOUSE-KEEP.
MOVE SPACES TO DATA-LINE.

```

MOVE ZERO TO COMMON-PERCENTAGES,
B-PERCENTAGES,
C-PERCENTAGES,
D-PERCENTAGES,
E-PERCENTAGES,
F-PERCENTAGES,
G-PERCENTAGES.

READ-INPUT.
  READ COST-ALLOCATION-UNRESOLVED AT END
  GO TO WRITE-SUBTOTAL.

START-RESOLVING.
  MOVE ACCT-NO TO ACCT-HOLD, ACCT-NOMINAL, RES-ACCT-NO,
    ACCT-NO-OUT.
  MOVE ALLOC-CODE TO ALLOC-CODE-HOLD, ALLOC-CODE-OUT.
  MOVE COST-CAT TO RES-COST-CAT, COST-CAT-OUT.
  MOVE AMTS (1) TO AMOUNT-OUT, AMT-HOLD, AMT-HOLD-OVFLOW.
  ADD AMTS (1) TO SUBTOTAL.

READ-CRF.
  READ CROSS-REF-FILE INVALID KEY
    MOVE DATA-LINE TO PRINT-DATA
    PERFORM PRINT-LINE-OUT
    MOVE '*** ABOVE ACCOUNT NUMBER NOT FOUND ***' TO
      PRINT-DATA
    PERFORM PRINT-LINE-OUT
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT
    GO TO READ-INPUT.

  IF ALLOC-CODE = 'B' THEN
    GO TO ALLOCATION-B.
  IF ALLOC-CODE = 'C' THEN
    GO TO ALLOCATION-C.
  IF ALLOC-CODE = 'D' THEN
    GO TO ALLOCATION-D.
  IF ALLOC-CODE = 'E' THEN
    GO TO ALLOCATION-E.
  IF ALLOC-CODE = 'F' THEN
    GO TO ALLOCATION-F.
  IF ALLOC-CODE = 'G' THEN
    GO TO ALLOCATION-G.

INVALID-ALLOC-CODE-FALLTHRU.
  MOVE DATA-LINE TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE '*** ABOVE RECORD HAS INVALID ALLOC CODE ***'
    TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
  GO TO READ-INPUT.

ALLOCATION-B.
  IF B-PERCENTAGES = ZERO THEN
    PERFORM GET-B-PERCENTAGES.
  PERFORM MOVE-B-PERCENT-TO-COMMON VARYING I

```

```
    FROM 1 BY 1 UNTIL I > NO-OF-CC.  
ALLOCATION-B-1.  
    PERFORM WRITE-RESOLVED-COSTS THRU WRITE-RESOLVED-COSTS-END.  
    PERFORM READ-INPUT.  
    IF ACCT-NO = TO ACCT-HOLD THEN  
        PERFORM START-RESOLVING  
        GO TO ALLOCATION-B-1  
    ELSE GO TO START-RESOLVING.  
ALLOCATION-C.  
    IF C-PERCENTAGES = ZERO THEN  
        PERFORM GET-C-PERCENTAGES.  
    PERFORM MOVE-C-PERCENT-TO-COMMON VARYING I  
        FROM 1 BY 1 UNTIL I > NO-OF-CC.  
ALLOCATION-C-1.  
    PERFORM WRITE-RESOLVED-COSTS THRU WRITE-RESOLVED-COSTS-END.  
    PERFORM READ-INPUT.  
    IF ACCT-NO = TO ACCT-HOLD THEN  
        PERFORM START-RESOLVING  
        GO TO ALLOCATION-C-1  
    ELSE GO TO START-RESOLVING.  
ALLOCATION-D.  
    IF D-PERCENTAGES = ZERO THEN  
        PERFORM GET-D-PERCENTAGES.  
    PERFORM MOVE-D-PERCENT-TO-COMMON VARYING I  
        FROM 1 BY 1 UNTIL I > NO-OF-CC.  
ALLOCATION-D-1.  
    PERFORM WRITE-RESOLVED-COSTS THRU WRITE-RESOLVED-COSTS-END.  
    PERFORM READ-INPUT.  
    IF ACCT-NO = TO ACCT-HOLD THEN  
        PERFORM START-RESOLVING  
        GO TO ALLOCATION-D-1  
    ELSE GO TO START-RESOLVING.  
ALLOCATION-E.  
    IF E-PERCENTAGES = ZERO THEN  
        PERFORM GET-E-PERCENTAGES.  
    PERFORM MOVE-E-PERCENT-TO-COMMON VARYING I  
        FROM 1 BY 1 UNTIL I > NO-OF-CC.  
ALLOCATION-E-1.  
    PERFORM WRITE-RESOLVED-COSTS THRU WRITE-RESOLVED-COSTS-END.  
    PERFORM READ-INPUT.  
    IF ACCT-NO = TO ACCT-HOLD THEN  
        PERFORM START-RESOLVING  
        GO TO ALLOCATION-E-1  
    ELSE GO TO START-RESOLVING.  
ALLOCATION-F.  
    IF F-PERCENTAGES = ZERO THEN  
        PERFORM GET-F-PERCENTAGES.  
    PERFORM MOVE-F-PERCENT-TO-COMMON VARYING I  
        FROM 1 BY 1 UNTIL I > NO-OF-CC.  
ALLOCATION-F-1.
```

```

PERFORM WRITE-RESOLVED-COSTS THRU WRITE-RESOLVED-COSTS-END.
PERFORM READ-INPUT.
IF ACCT-NO = ACCT-HOLD THEN
    PERFORM START-RESOLVING
    GO TO ALLOCATION-F-1
ELSE GO TO START-RESOLVING.
ALLOCATION-G.
IF G-PERCENTAGES = ZERO THEN
    PERFORM GET-G-PERCENTAGES.
PERFORM MOVE-G-PERCENT-TO-COMMON VARYING I
    FROM 1 BY 1 UNTIL I > NO-OF-CC.
ALLOCATION-G-1.
PERFORM WRITE-RESOLVED-COSTS THRU WRITE-RESOLVED-COSTS-END.
PERFORM READ-INPUT.
IF ACCT-NO = ACCT-HOLD THEN
    PERFORM START-RESOLVING
    GO TO ALLOCATION-G-1
ELSE GO TO START-RESOLVING.
MOVE-SPACES.
MOVE SPACES TO PRINT-LINE.
PRINT-LINE-OUT.
WRITE PRINT-LINE BEFORE 1.
ADD 1 TO LINE-CT.
GET-COMMON-PERCENTAGES.
MOVE ZEROS TO COMMON-AMOUNTS, COMMON-AMOUNTS-ACCUM.
PERFORM SEARCH-COST-TABLE VARYING I FROM 1 BY 1
    UNTIL I > NO-OF-CC.
PERFORM TOTAL-AMTS VARYING I FROM 1 BY 1
    UNTIL I > NO-OF-CC.
IF COMMON-AMOUNTS-ACCUM = ZEROS THEN
    PERFORM MOVE-IN-EQUAL-PERCENTAGES VARYING
        I FROM 1 BY 1 UNTIL I > NO-OF-CC
ELSE MOVE 1 TO PERCENT-100
    PERFORM MOVE-IN-PERCENTAGES VARYING
        I FROM 1 BY 1 UNTIL I > NO-OF-CC
SEARCH-COST-TABLE.
MOVE CRF-COST-CNTR (I) TO SEARCH-NO.
SET CC TO 1.
SEARCH ALL RES-COSTS AT END
MOVE ZERO TO COMMON-AMT (I)
WHEN COST-CNTR (CC) = SEARCH-NO
MOVE CC-AMOUNT (CC) TO COMMON-AMT (I).
TOTAL-AMTS.
IF COMMON-AMT (I) < ZERO THEN
MOVE ZERO TO COMMON-AMT (I).
ADD COMMON-AMT (I) TO COMMON-AMOUNTS-ACCUM.
MOVE-IN-EQUAL-PERCENTAGES.
DIVIDE 1 BY NO-OF-CC GIVING COMMON-PERCENT (I) ROUNDED.
MOVE-IN-PERCENTAGES.
DIVIDE COMMON-AMT (I) BY COMMON-AMOUNTS-ACCUM

```

```

        GIVING COMMON-PERCENT (I) ROUNDED.
        SUBTRACT COMMON-PERCENT (I) FROM PERCENT-100.
        IF I = NO-OF-CC THEN
            ADD PERCENT-100 TO COMMON-PERCENT (I).
        WRITE-RESOLVED-COSTS.
        MOVE 1 TO I, J.
        FILL-PRINT-LINE.
        MOVE ACCT-NO TO RES-ACCT-NO.
        MULTIPLY COMMON-PERCENT (I) BY AMT-HOLD GIVING RES-AMTS (1).
        MOVE RES-AMTS (1) TO AMT-OUT OF RES-DATA (J).
        SUBTRACT RES-AMTS (1) FROM AMT-HOLD-OVFLOW.
        MOVE SLASH TO SLASH-OUT OF RES-DATA (J).
        MOVE CRF-COST-CNTR OF CC-BLK (I) TO
            COST-CNTR-OUT OF RES-DATA (J),
            RES-COST-CNTR.
        MOVE COST-CAT TO RES-COST-CAT.
        MOVE CRF-MATCH-CODE OF CC-BLK (I) TO
            RES-MATCH-CODE.
        BALANCE-LAST-AMOUNT.
        IF I = NO-OF-CC THEN
            ADD RES-AMTS (1) TO AMT-HOLD-OVFLOW
            MOVE AMT-HOLD-OVFLOW TO RES-AMTS (1)
                AMT-OUT OF RES-DATA (J).
        WRITE-DISK-REC.
        WRITE RES-REC.
        CHECK-NO-OF-CC.
        IF I = NO-OF-CC THEN
            GO TO WRITE-PRINT-LINE.
        IF J = 6 THEN PERFORM
            WRITE-PRINT-LINE.
            MOVE 0 TO J.
            ADD 1 TO I, J.
            GO TO FILL-PRINT-LINE.
        WRITE-PRINT-LINE.
        MOVE DATA-LINE TO PRINT-DATA.
        PERFORM PRINT-LINE-OUT.
        MOVE SPACES TO DATA-LINE.
        SKIP-A-LINE.
        PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
        IF LINE-CT > MAX-LINE THEN
            PERFORM WRITE-HEADINGS.
        WRITE-RESOLVED-COSTS-END.
        EXIT.
        GET-B-PERCENTAGES.
        PERFORM GET-COMMON-PERCENTAGES.
        PERFORM MOVE-COMMON-TO-B-PERCENT VARYING
            I FROM 1 BY 1 UNTIL I > NO-OF-CC.
        MOVE-COMMON-TO-B-PERCENT.
        MOVE COMMON-PERCENT (I) TO B-PERCENT (I).
        GET-C-PERCENTAGES.

```

```
PERFORM GET-COMMON-PERCENTAGES.  
PERFORM MOVE-COMMON-TO-C-PERCENT VARYING  
    I FROM 1 BY 1 UNTIL I > NO-OF-CC.  
MOVE-COMMON-TO-C-PERCENT.  
    MOVE COMMON-PERCENT (I) TO C-PERCENT (I).  
GET-D-PERCENTAGES.  
    PERFORM GET-COMMON-PERCENTAGES.  
    PERFORM MOVE-COMMON-TO-D-PERCENT VARYING  
        I FROM 1 BY 1 UNTIL I > NO-OF-CC.  
MOVE-COMMON-TO-D-PERCENT.  
    MOVE COMMON-PERCENT (I) TO D-PERCENT (I).  
GET-E-PERCENTAGES.  
    PERFORM GET-COMMON-PERCENTAGES.  
    PERFORM MOVE-COMMON-TO-E-PERCENT VARYING  
        I FROM 1 BY 1 UNTIL I > NO-OF-CC.  
MOVE-COMMON-TO-E-PERCENT.  
    MOVE COMMON-PERCENT (I) TO E-PERCENT (I).  
GET-F-PERCENTAGES.  
    PERFORM GET-COMMON-PERCENTAGES.  
    PERFORM MOVE-COMMON-TO-F-PERCENT VARYING  
        I FROM 1 BY 1 UNTIL I > NO-OF-CC.  
MOVE-COMMON-TO-F-PERCENT.  
    MOVE COMMON-PERCENT (I) TO F-PERCENT (I).  
GET-G-PERCENTAGES.  
    PERFORM GET-COMMON-PERCENTAGES.  
    PERFORM MOVE-COMMON-TO-G-PERCENT VARYING  
        I FROM 1 BY 1 UNTIL I > NO-OF-CC.  
MOVE-COMMON-TO-G-PERCENT.  
    MOVE COMMON-PERCENT (I) TO G-PERCENT (I).  
MOVE-B-PERCENT-TO-COMMON.  
    MOVE B-PERCENT (I) TO COMMON-PERCENT (I).  
MOVE-C-PERCENT-TO-COMMON.  
    MOVE C-PERCENT (I) TO COMMON-PERCENT (I).  
MOVE-D-PERCENT-TO-COMMON.  
    MOVE D-PERCENT (I) TO COMMON-PERCENT (I).  
MOVE-E-PERCENT-TO-COMMON.  
    MOVE E-PERCENT (I) TO COMMON-PERCENT (I).  
MOVE-F-PERCENT-TO-COMMON.  
    MOVE F-PERCENT (I) TO COMMON-PERCENT (I).  
MOVE-G-PERCENT-TO-COMMON.  
    MOVE G-PERCENT (I) TO COMMON-PERCENT (I).  
WRITE-SUBTOTAL.  
    MOVE SPACES TO DATA-LINE.  
    PERFORM MOVE-SPACES.  
    PERFORM PRINT-LINE-OUT 2 TIMES.  
    MOVE 'SUBTOTAL ' TO ACCT-NO-OUT.  
    MOVE SUBTOTAL TO AMOUNT-OUT.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
CLOSE-IO-FILES.
```

CLOSE COST-ALLOCATION-RESOLVED,
COST-ALLOCATION-UNRESOLVED,
COST-ALLOCATION-RESOLVED-COSTS,
CROSS-REF-FILE,
CARD-IN,
PRINT-OUT.

E-O-J.
STOP RUN.

PGMNAME=WUASUZFI
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASUZFI.
AUTHOR. ACT SYSTEMS, INC
SUITE 200
807 W MORSE BLVD
WINTER PARK, FL 32789.
DATE-WRITTEN. DEC 1977.
DATE-COMPILED.
REMARKS. THIS PROGRAM ZEROS OUT THE AMOUNT
FIELDS IN THE COST-MATRIX FILE
DEPENDING ON THE PARAMETER CARD(S).
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-IN
ASSIGN TO UR-2540R-S-CARDIN
ACCESS MODE IS SEQUENTIAL.
SELECT PRINT-OUT
ASSIGN TO UR-1403-S-PRINTOUT
ACCESS MODE IS SEQUENTIAL.
SELECT COST-MATRIX-FILE
ASSIGN TO DA-3330-I-CMFILE
ACCESS MODE IS SEQUENTIAL
RECORD KEY IS CM-NUMBER.
DATA DIVISION.
FILE SECTION.
FD COST-MATRIX-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-REC.
01 INPUT-REC.
05 MATCH-CODE PIC X(4).
05 CM-NUMBER PIC X(16).
05 AMOUNTS.
10 AMTS OCCURS 6 TIMES PIC S9(8)V99.

* THE ABOVE AMTS (AMOUNTS) ARE AS FOLLOWS
* (1) -- CURRENT COST AMOUNT
* (2) -- PREVIOUS MONTH COST AMOUNT
* (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)
* (4) -- BUDGET AMOUNT
* (5) -- BUDGET AMOUNT YEAR TO DATE (YTD)

* (6) -- PREVIOUS YEAR COST AMOUNT (PYC)

FD CARD-IN
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
 05 PARM-CODE PIC X.
 05 FILLER PIC X(79).
FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
 05 FILLER PIC X.
 05 PRINT-DATA PIC X(80).
 05 FILLER PIC X(52).
PROCEDURE DIVISION.
OPEN-FILES.
 OPEN I-O COST-MATRIX-FILE,
 INPUT CARD-IN,
 OUTPUT PRINT-OUT.

TOP-OF-NEWPAGE.

 PERFORM MOVE-SPACES.
 WRITE PRINT-LINE BEFORE NEW-PAGE.
 PERFORM PRINT-LINE-OUT 10 TIMES.
 MOVE 'THE CUST MATRIX FILE IS BEING CHANGED BY THE FOLLOWING METHOD' TO PRINT-DATA.
- PERFORM PRINT-LINE-OUT.
 PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

READ-CARD-IN.

 READ CARD-IN AT END GO TO STANDARD-ZERO-PROCEDURE.
 IF PARM-CODE = 'F' THEN
 GO TO FISCAL-YEAR-CHANGE-PROCEDURE.
 IF PARM-CODE = 'Z' THEN
 GO TO ZERO-OUT-FILE-PROCEDURE.

INVALID-PARM-CODE-FALLTHRU.

 MOVE 'THE FOLLOWING PARAMETER CARD WAS INVALID -- JOB ABORTED' TO PRINT-DATA.

 PERFORM PRINT-LINE-OUT.

 MOVE INPUT-CARD TO PRINT-DATA.

 PERFORM PRINT-LINE-OUT.

 GO TO CLOSE-FILES.

STANDARD-ZERO-PROCEDURE.

 MOVE 'NORMAL PERIOD CHANGE OVER' TO PRINT-DATA.
 PERFORM PRINT-LINE-OUT.

START-ZERO-1.

 PERFORM READ-CM-FILE.

 MOVE AMTS (1) TO AMTS (2).

 MOVE ZEROS TO AMTS (1).

```
    PERFORM REWRITE-CM-FILE.  
    GO TO START-ZERO-1.  
FISCAL-YEAR-CHANGE-PROCEDURE.  
    MOVE '    FISCAL YEAR CHANGE OVER ' TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
START-ZERO-2.  
    PERFORM READ-CM-FILE.  
    MOVE AMTS (1) TO AMTS (2).  
    MOVE AMTS (3) TO AMTS (6).  
    MOVE ZEROS TO AMTS (1), AMTS (3), AMTS (4), AMTS (5).  
    PERFORM REWRITE-CM-FILE.  
    GO TO START-ZERO-2.  
ZERO-OUT-FILE-PROCEDURE.  
    MOVE '    COST MATRIX FILE IS BEING ZEROED OUT' TO  
         PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
START-ZERO-3.  
    PERFORM READ-CM-FILE.  
    MOVE ZEROS TO AMOUNTS.  
    PERFORM REWRITE-CM-FILE.  
    GO TO START-ZERO-3.  
READ-CM-FILE.  
    READ COST-MATRIX-FILE AT END  
    MOVE 'REVISION OF CM FILE COMPLETED' TO PRINT-DATA  
        PERFORM PRINT-LINE-OUT  
        GO TO CLOSE-FILES.  
REWRITE-CM-FILE.  
    REWRITE INPUT-REC.  
MOVE-SPACES.  
    MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
CLOSE-FILES.  
    CLOSE COST-MATRIX-FILE,  
        CARD-IN,  
        PRINT-OUT.  
E-O-J.  
    STOP RUN.
```

```

PGMNAME=WUASCUPD
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASCUPD.
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FL 32789.
DATE-WRITTEN. DEC 1977.
DATE-COMPILED.
REMARKS.    THIS PROGRAM READS THE COST ALLOCATION RESOLVED
            FILE (RESOCIFI) AND UPDATES THE COST MATRIX FILE
            (CMFILE) WITH THE CURRENT COSTS.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT COST-ALLOCATION-RESOLVED
        ASSIGN TO UT-3330-S-RESOCIFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS RANDOM
        RECORD KEY IS CM-NUMBER
        NOMINAL KEY IS CM-NOMINAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT FILE-SORT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD COST-MATRIX-FILE
BLOCK CONTAINS 5 RECORDS
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CM-REC.
01 CM-REC.
    05 MATCH-CODE      PIC X(4).
    05 CM-NUMBER.
        10 ACCT-NO      PIC X(10).
        10 COST-CNTR     PIC X(4).
        10 COST-CAT      PIC X(2).
    05 AMOUNTS.
        10 AMT OCCURS 6 TIMES PIC S9(8)V99.

```

```

FD COST-ALLOCATION-RESOLVED
  BLOCK CONTAINS 5 RECORDS
  RECORD CONTAINS 80 CHARACTERS
  LABEL RECORDS ARE STANDARD
  DATA RECORD IS RES-REC.
01 RES-REC.
  05 RES-NUM          PIC X(16).
  05 RES-MATCH-CODE  PIC X(4).
  05 RES-AMOUNTS.
    10 RES-AMT OCCURS 6 TIMES PIC S9(8)V99.
SD FILE-SORT
  RECORD CONTAINS 80 CHARACTERS
  DATA RECORD IS SORT-INPUT.
01 SORT-INPUT.
  05 SORT-FIELD      PIC X(16).
  05 FILLER         PIC X(64).
FD PRINT-OUT
  RECORD CONTAINS 133 CHARACTERS
  LABEL RECORDS ARE OMITTED
  DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
  05 FILLER         PIC X.
  05 PRINT-DATA     PIC X(132).
WORKING-STORAGE SECTION.
77 CM-NOMINAL        PIC X(16) VALUE SPACES.
77 READ-COUNT        PIC 9(7) VALUE ZERO.
77 UPDATE-COUNT     PIC 9(7) VALUE ZERO.
77 GOOD-COUNT        PIC 9(7) VALUE ZERO.
77 BAD-COUNT         PIC 9(7) VALUE ZERO.
01 AMOUNT-HOLD.
  05 AMT-HOLD OCCURS 6 TIMES PIC S9(8)V99.
01 DATA-LINE.
  05 FILLER         PIC X(10) VALUE SPACES.
  05 COMMENT        PIC X(18) VALUE SPACES.
  05 FILLER         PIC XX   VALUE SPACES.
  05 COUNT-OUT       PIC Z(7).
  05 ERROR-REC      PIC X(80).
  05 FILLER         PIC X(15).
PROCEDURE DIVISION.
SORT-RECORDS.
  SORT FILE-SORT ON ASCENDING KEY
    SORT-FIELD
    USING COST-ALLOCATION-RESOLVED
    GIVING COST-ALLOCATION-RESOLVED.
OPEN-FILES.
  OPEN I-O  COST-MATRIX-FILE,
    INPUT COST-ALLOCATION-RESOLVED,
    OUTPUT PRINT-OUT.
HOUSE-KEEP.
  MOVE SPACES TO DATA-LINE.

```

MOVE ZEROS TO AMOUNT-HOLD.
TOP-NEW-PAGE.
 PERFORM MOVE-SPACES.
 WRITE PRINT-LINE BEFORE NEW-PAGE.
 MOVE 'UPDATE CM FILE PGM' TO COMMENT.
 MOVE DATA-LINE TO PRINT-DATA.
 PERFORM PRINT-LINE-OUT.
 PERFORM MOVE-SPACES.
 PERFORM PRINT-LINE-OUT 3 TIMES.
READ-RES-REC.
 READ COST-ALLOCATION-RESOLVED AT END
 GO TO WRITE-COUNTS.
 ADD 1 TO READ-COUNT.
MOVE-NOMINAL-KEY.
 MOVE RES-NUM TO CM-NOMINAL.
 READ COST-MATRIX-FILE INVALID KEY
 MOVE 'INVALID RECORD KEY' TO COMMENT
 MOVE RES-REC TO ERROR-REC
 MOVE DATA-LINE TO PRINT-DATA
 PERFORM PRINT-LINE-OUT
 PERFORM MOVE-SPACES THRU PRINT-LINE-OUT
 MOVE SPACES TO DATA-LINE
 ADD 1 TO BAD-COUNT
 GO TO READ-RES-REC.
 MOVE AMOUNTS TO AMOUNT-HOLD.
 MOVE ZEROS TO AMT-HOLD (1).
START-UPDATE.
 ADD RES-AMT (1) TO AMT-HOLD (1) AMT-HOLD (3).
 ADD 1 TO GOOD-COUNT.
CHECK-NEXT-REC.
 PERFORM READ-RES-REC.
 IF RES-NUM - TO CM-NOMINAL THEN
 GO TO START-UPDATE.
REWRITE-CM-REC.
 MOVE AMOUNT-HOLD TO AMOUNTS.
 REWRITE CM-REC.
 ADD 1 TO UPDATE-COUNT.
 MOVE SPACES TO CM-NUMBER.
GO-TO-MOVE-NOMINAL-KEY.
 GO TO MOVE-NOMINAL-KEY.
MOVE-SPACES.
 MOVE SPACES TO PRINT-LINE.
PRINT-LINE-OUT.
 WRITE PRINT-LINE BEFORE 1.
WRITE-COUNTS.
 IF CM-NUMBER NOT = SPACES THEN
 PERFORM REWRITE-CM-REC.
 PERFORM MOVE-SPACES.
 PERFORM PRINT-LINE-OUT 3 TIMES.
 MOVE 'INPUT RECORD COUNT' TO COMMENT.

```
MOVE READ-COUNT TO COUNT-OUT.  
PERFORM PRINT-LINE-OUT.  
MOVE 'ERROR RECORD COUNT' TO COMMENT.  
MOVE BAD-COUNT TO COUNT-OUT.  
PERFORM PRINT-LINE-OUT.  
MOVE 'GOOD RECORD COUNT' TO COMMENT.  
MOVE GOOD-COUNT TO COUNT-OUT.  
PERFORM PRINT-LINE-OUT.  
MOVE 'UPDATED REC COUNT' TO COMMENT.  
MOVE UPDATE-COUNT TO COUNT-OUT.  
PERFORM PRINT-LINE-OUT.  
CLOSE-FILES.  
CLOSE    COST-MATRIX-FILE,  
        COST-ALLOCATION-RESOLVED  
        PRINT-OUT.  
E-O-J.  
STOP RUN.
```

```
PGMNAME=WUASLV44
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASLV44.
AUTHOR. ACT SYSTEMS INC
      SUITE 200
      807 W MORSE BLVD
      WINTER PARK, FLA 32789.
DATE-WRITTEN. JAN 1978.
DATE-COMPILED.
REMARKS.   THIS PROGRAM SORTS THE COST MATRIX FILE BY
           MATCH CODE AND PRINTS OUT LEVEL IV
           (SUPPORT SERVICES) REPORT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
      C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
      SELECT CARD-IN
          ASSIGN TO UR-2540R-S-CARDIN
          ACCESS MODE IS SEQUENTIAL.
      SELECT PRINT-OUT
          ASSIGN TO UR-1403-S-PRINTOUT
          ACCESS MODE IS SEQUENTIAL.
      - SELECT COST-MATRIX-FILE
          ASSIGN TO DA-3330-I-CMFILE
          ACCESS MODE IS SEQUENTIAL
          RECLRD KEY IS CM-NUMBER.
      SELECT FINANCIAL-ACCT-TABFI
          ASSIGN TO DA-3330-I-FATABFI
          ACCESS MODE IS RANDOM
          NOMINAL KEY IS FA-NOMINAL
          RECORD KEY IS FA-NUMBER.
      SELECT FILE-SORT
          ASSIGN TO UT-3330-S-FILESORT
          ACCESS MODE IS SEQUENTIAL.
      SELECT SORTED-FILE
          ASSIGN TO UT-3330-S-SORTEDFI
          ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
      RECLRD CONTAINS 80 CHARACTERS
      LABEL RECORDS ARE OMITTED
      DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
    05 DATE-IN          PIC X(18).
```

```

      05 TITLE-IN          PIC X(24).
      05 FILLER           PIC X(38).
FD  PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
      05 FILLER           PIC X.
      05 PRINT-DATA        PIC X(132).
FD  CUST-MATRIX-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CM-REC.
01 CM-REC.
      05 MATCH-CODE-POS1  PIC X.
      05 FILLER           PIC X(3).
      05 CM-NUMBER        PIC X(10).
      05 FILLER           PIC X(66).
FD  FINANCIAL-ACCT-TABFI
BLOCK CONTAINS 8 RECORDS
RECORD CONTAINS 58 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS FA-REC.
01 FA-REC.
      05 FA-NUMBER        PIC X(10).
      05 FA-NAME          PIC X(20) OCCURS 2 TIMES.
      05 FILLER           PIC X(8).
SD  FILE-SORT
RECORD CONTAINS 80 CHARACTERS
DATA RECORD IS SORT-INPUT.
01 SORT-INPUT.
      05 S-MATCH-CODE    PIC X(4).
      05 FILLER           PIC X(16).
      05 FILLER           PIC X(60).
FD  SORTED-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-REC.
01 INPUT-REC.
      05 MATCH-CODE        PIC X(4).
      05 ACCT-NO           PIC X(10).
      05 COST-CNTR.
          10 FILLER          PIC X.
          10 CC-POS2          PIC 9.
          10 FILLER          PIC X.
          10 CC-POS4          PIC 9.
      05 COST-CAT          PIC XX.
      05 AMTS              PIC S9(8)V99 OCCURS 6 TIMES.

```

```

*****  

*      THE ABOVE AMTS ARE AS FOLLOWS  

*      (1) -- CURRENT COST AMOUNT  

*      (2) -- PREVIOUS MONTH COST AMOUNT  

*      (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)  

*      (4) -- BUDGET AMOUNT  

*      (5) -- BUDGET AMOUNT YEAR TO DATE  

*      (6) -- PREVIOUS YEAR COST AMOUNT (PYC)  

*****  

WORKING-STORAGE SECTION.  

77 PAGE-CT          PIC 9(4)  VALUE ZERO.  

77 LINE-CT          PIC 9(2)  VALUE ZERO.  

77 FA-NOMINAL       PIC X(10) VALUE SPACES.  

77 I                PIC 9.  

77 MATCH-HOLD        PIC X(4)  VALUE SPACES.  

77 MATCH-POS1        PIC X     VALUE '4'.  

77 MAX-LINE         PIC 9(2)  COMP VALUE 56.  

77 SUBTOTAL-FLAG    PIC X     VALUE '0'.  

01 HEAD-1.  

    05 FILLER          PIC X(57) VALUE SPACES.  

    05 FILLER          PIC X(18) VALUE 'LEVEL IV REPORT OF'.  

    05 FILLER          PIC X(57) VALUE SPACES.  

01 HEAD-2.  

    05 FILLER          PIC X(6)  VALUE 'DATE: '.  

    05 DATE-OUT        PIC X(18) VALUE SPACES.  

    05 FILLER          PIC X(30) VALUE SPACES.  

    05 FILLER          PIC X(26) VALUE 'SUPPORT SERVICES COSTS  

                           'FOR'.  

    -  

    05 FILLER          PIC X(42) VALUE SPACES.  

    05 FILLER          PIC X(6)  VALUE 'PAGE: '.  

    05 PAGE-OUT        PIC Z(4).  

01 HEAD-3.  

    05 FILLER          PIC X(55) VALUE SPACES.  

    05 TITLE-OUT        PIC X(24) VALUE SPACES.  

    05 FILLER          PIC X(53) VALUE SPACES.  

01 HEAD-4.  

    05 FILLER          PIC X(26) VALUE 'COST CATEGORY DESCRIPTI  

                           'ONS'.  

    -  

    05 FILLER          PIC X(2)  VALUE SPACES.  

    05 FILLER          PIC X(102) VALUE '*****  

                           ***** S U P P O R T   S E R V I  

                           C E S   *****  

                           ***'.  

    -  

    05 FILLER          PIC X(2)  VALUE SPACES.  

01 HEAD-5.  

    05 FILLER          PIC X(31) VALUE SPACES.  

    05 FILLER          PIC X(8)  VALUE 'CUSTOMER'.  

    05 FILLER          PIC X(13) VALUE SPACES.  

    05 FILLER          PIC X(9)  VALUE 'GENERAL &'.  

    05 FILLER          PIC X(12) VALUE SPACES.

```

```

      05 FILLER          PIC X(34) VALUE 'MAINTANCE OF OUTSIDE SE
                           'RVICE AREAS'.
      05 FILLER          PIC X(12) VALUE SPACES.
      05 FILLER          PIC X(7)  VALUE 'COST OF'.
      05 FILLER          PIC X(6)  VALUE SPACES.

01 HEAD-6.
      05 FILLER          PIC X(30) VALUE SPACES.
      05 FILLER          PIC X(10) VALUE 'ACCOUNTING'.
      05 FILLER          PIC X(9)  VALUE SPACES.
      05 FILLER          PIC X(14) VALUE 'ADMINISTRATIVE'.
      05 FILLER          PIC X(10) VALUE SPACES.
      05 FILLER          PIC X(11) VALUE 'TAYLOR MILL'.
      05 FILLER          PIC X(11) VALUE SPACES.
      05 FILLER          PIC X(12) VALUE 'INDEPENDENCE'.
      05 FILLER          PIC X(9)  VALUE SPACES.
      05 FILLER          PIC X(14) VALUE 'MATERIALS SOLD'.
      05 FILLER          PIC X(2)  VALUE SPACES.

01 DATA-LINE.
      05 FILLER          PIC X(2)  VALUE SPACES.
      05 DESC             PIC X(20).
      05 AMOUNTS OCCURS 5 TIMES.
          10 FILLER        PIC X(6).
          10 AMT            PIC -(8)---.
          10 FILLER        PIC X(5).

01 ACCUMS.
      05 ACCUM            PIC S9(8)V99 OCCURS 5 TIMES.

01 SUBTOTALS.
      05 SUBS             PIC S9(8)V99 OCCURS 5 TIMES.

PROCEDURE DIVISION.
OPEN-FILES.
    OPEN INPUT CARD-IN,
           INPUT CARD-IN,
           COST-MATRIX-FILE,
           FINANCIAL-ACCT-TABFI,
           OUTPUT PRINT-OUT.

READ-CARD-IN.
    READ CARD-IN AT END
        MOVE '*** NO DATE/TITLE CARD ** JOB ABORTED ***' TO
        PRINT-DATA,
        PERFORM PRINT-LINE-OUT,
        GO TO CLOSE-FILES.

    MOVE DATE-IN TO DATE-OUT.
    MOVE TITLE-IN TO TITLE-OUT.

SORT-CM-FILE.
    OPEN OUTPUT SORTED-FILE.
    SORT FILE-SORT ON ASCENDING KEY
        S-MATCH-CODE
    INPUT PROCEDURE
        READ-SORT-INPUT THRU READ-SORT-INPUT-END
    OUTPUT PROCEDURE

```

```

        WRITE-SORT-OUTPUT THRU WRITE-SORT-OUTPUT-END.
CLOSE SORTED-FILE.
OPEN INPUT SORTED-FILE.
WRITE-HEADINGS.
MOVE ZERO TO LINE-CT.
ADD 1 TO PAGE-CT.
MOVE PAGE-CT TO PAGE-OUT.
PERFORM MOVE-SPACES.
WRITE PRINT-LINE BEFORE NEW-PAGE.
MOVE HEAD-1 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEAD-2 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEAD-3 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
MOVE HEAD-4 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEAD-5 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEAD-6 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
ZERO-ACCUMS.
MOVE ZEROS TO SUBTOTALS.
HOUSE-KEEP.
MOVE SPACES TO DATA-LINE.
MOVE ZEROS TO ACCUMS.
READ-SORTED-FILE.
READ SORTED-FILE AT END PERFORM WRITE-LINE-OUT.
GO TO WRITE-TOTALS.
NEW-MATCH-CODE.
MOVE MATCH-CODE TO MATCH-HOLD.
IF (MATCH-CODE = '4815') AND (SUBTOTAL-FLAG = '0') THEN
  PERFORM WRITE-SUB-TOTAL
  MOVE '1' TO SUBTOTAL-FLAG
  GO TO READ-SORTED-FILE.
MOVE ACCT-NR TO FA-NOMINAL.
READ FINANCIAL-ACCT-TABFI. INVALID KEY
MOVE SPACES TO FA-NAME (1), FA-NAME (2).
TOTAL-COST-CNTRS.
IF CC-POS2 < 3 THEN
  ADD AMTS (1) TO ACCUM (CC-POS2), SUBS (CC-POS2)
  GO TO READ-NEXT-REC.
IF CC-POS2 = 4 THEN
  MOVE 5 TO CC-POS2
  ADD AMTS (1) TO ACCUM (CC-POS2), SUBS (CC-POS2)
  GO TO READ-NEXT-REC.
IF CC-POS4 = 2 THEN
  MOVE 4 TO CC-POS2

```

```
    ADD AMTS (1) TO ACCUM (CC-POS2), SUBS (CC-POS2)
    ELSE ADD AMTS (1) TO ACCUM (CC-POS2), SUBS (CC-POS2).
READ-NEXT-REC.
    PERFORM READ-SORTED-FILE.
    IF MATCH-CODE EQUAL TO MATCH-HOLD THEN
        GO TO TOTAL-COST-CNTRS.
WRITE-LINE-OUT.
    MOVE FA-NAME (1) TO DESC.
    PERFORM MOVE-ACCUMS VARYING I FROM 1 BY 1 UNTIL I = 6.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    IF FA-NAME (2) NOT EQUAL TO SPACES THEN
        MOVE SPACES TO DATA-LINE.
        MOVE FA-NAME (2) TO DESC.
        MOVE DATA-LINE TO PRINT-DATA,
        PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM HOUSE-KEEP.
CHECK-FOR-NEW-HEADING.
    IF LINE-CT NOT < MAX-LINE THEN
        PERFORM WRITE-HEADINGS.
    GO TO NEW-MATCH-CODE.
WRITE-TOTALS.
    PERFORM MOVE-SPACES.
    PERFORM PRINT-LINE-OUT 2 TIMES.
    PERFORM HOUSE-KEEP.
    MOVE 'TOTALS' TO DESC.
    PERFORM MOVE-AMOUNTS VARYING I FROM 1 BY 1 UNTIL I = 6.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    GO TO CLOSE-FILES.
WRITE-SUB-TOTAL.
    PERFORM MOVE-AMOUNTS VARYING I FROM 3 BY 1 UNTIL I = 6.
    MOVE 'EXPENCES COVERED BY' TO DESC.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE SPACES TO DATA-LINE.
    MOVE 'OTHER REVENUE' TO DESC.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
MOVE-SPACES.
    MOVE SPACES TO PRINT-LINE.
PRINT-LINE-OUT.
    WRITE PRINT-LINE BEFORE 1.
    ADD 1 TO LINE-CT.
MOVE-ACCUMS.
    MOVE ACCUM (I) TO AMT OF AMOUNTS (I).
MOVE-AMOUNTS.
    MOVE SUBS (I) TO AMT OF AMOUNTS (I).
```

```
MOVE ZERO TO SUBS (I).
READ-SORT-INPUT.
    READ COST-MATRIX-FILE AT END
        GO TO READ-SORT-INPUT-END.
    IF MATCH-CODE-POS1 NOT = TO MATCH-POS1
        THEN GO TO READ-SORT-INPUT.
    RELEASE SORT-INPUT FROM CM-REC.
    GO TO READ-SORT-INPUT.
READ-SORT-INPUT-END.
    EXIT.
WRITE-SORT-OUTPUT.
    RETURN FILE-SORT RECORD INTO INPUT-REC
        AT END GO TO WRITE-SORT-OUTPUT-END.
    WRITE INPUT-REC.
    GO TO WRITE-SORT-OUTPUT.
WRITE-SORT-OUTPUT-END.
    EXIT.
CLOSE-FILES.
    CLOSE    SORTED-FILE,
            CARD-IN,
            COST-MATRIX-FILE,
            FINANCIAL-ACCT-TABFI,
            PRINT-OUT.
E-O-J.
    STOP RUN.
```

```
PGMNAME=PLV43COV
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. PLV43COV.
AUTHOR. ACT SYSTEMS INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FLA 32789.
DATE-WRITTEN. JAN 1978.
DATE-COMPILED.
REMARKS. THIS PROGRAM SORTS THE COST MATRIX FILE BY
        MATCH CODE AND PRINTS OUT LEVEL IV (DELIVERY)
        REPORT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS SEQUENTIAL
        RECORD KEY IS CM-NUMBER.
    SELECT PRINT-FILE
        ASSIGN TO UT-3330-S-PRINTFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT FINANCIAL-ACCT-TABFI
        ASSIGN TO DA-3330-I-FATABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS FA-NOMINAL
        RECORD KEY IS FA-NUMBER.
    SELECT FILE-SORT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
    SELECT SORTED-FILE
        ASSIGN TO UT-3330-S-SORTEDFI
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE OMITTED
```

DATA RECORD IS INPUT-CARD.

01 INPUT-CARD.
 05 DATE-IN PIC X(18).
 05 TITLE-IN PIC X(24).
 05 FILLER PIC X(38).
FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
 05 FILLER PIC X.
 05 PRINT-DATA PIC X(132).
FD COST-MATRIX-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CM-REC.
01 CM-REC.
 05 MATCH-CODE-POS1 PIC X.
 05 FILLER PIC X(3).
 05 CM-NUMBER PIC X(10).
 05 FILLER PIC X(66).
FD PRINT-FILE
BLOCK CONTAINS 6 RECORDS
RECORD CONTAINS 240 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS PRINT-LINE-HOLD.
01 PRINT-LINE-HOLD.
 05 DESC-HOLD PIC X(20) OCCURS 2 TIMES.
 05 AMTS-HOLD PIC S9(8)V99 OCCURS 20 TIMES.
FD FINANCIAL-ACCT-TABFI
BLOCK CONTAINS 8 RECORDS
RECORD CONTAINS 58 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS FA-REC.
01 FA-REC.
 05 FA-NUMBER PIC X(10).
 05 FA-NAME PIC X(20) OCCURS 2 TIMES.
 05 FILLER PIC X(8).
SD FILE-SORT
RECORD CONTAINS 80 CHARACTERS
DATA RECORD IS SORT-INPUT.
01 SORT-INPUT.
 05 S-MATCH-CODE PIC X(4).
 05 FILLER PIC X(16).
 05 FILLER PIC X(60).
FD SORTED-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD

DATA RECORD IS INPUT-REC.

01 INPUT-REC.

 05 MATCH-CODE PIC X(4).

 05 ACCT-NO PIC X(10).

 05 COST-CNTR.

 10 FILLER PIC X.

 10 CC-POS2 PIC 9.

 10 FILLER PIC X.

 10 CC-POS4 PIC 9.

 05 COST-CAT PIC XX.

 05 AMTS PIC S9(8)V99 OCCURS 6 TIMES.

* THE ABOVE AMTS ARE AS FOLLOWS

* (1) -- CURRENT COST AMOUNT

* (2) -- PREVIOUS MONTH COST AMOUNT

* (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)

* (4) -- BUDGET AMOUNT

* (5) -- BUDGET AMOUNT YEAR TO DATE

* (6) -- PREVIOUS YEAR COST AMOUNT (PYC)

WORKING-STORAGE SECTION.

 77 PAGE-CT PIC 9(4) VALUE ZERO.

 77 LINE-CT PIC 9(2) VALUE ZERO.

 77 FA-NOMINAL PIC X(10) VALUE SPACES.

 77 I PIC 9.

 77 J PIC S99.

 77 K PIC 99.

 77 L PIC 99.

 77 M PIC 99.

 77 N PIC 99.

 77 O PIC 99.

 77 MATCH-HOLD PIC X(4) VALUE SPACES.

 77 MATCH-POS1 PIC X VALUE '3'.

 77 MAX-LINE PIC 9(2) COMP VALUE 56.

 77 MINUS-ONE PIC S9(8)V99 VALUE -0.01.

01 HEAD-1.

 05 FILLER PIC X(57) VALUE SPACES.

 05 FILLER PIC X(18) VALUE 'LEVEL IV REPORT OF'.

 05 FILLER PIC X(57) VALUE SPACES.

01 HEAD-2.

 05 FILLER PIC X(6) VALUE 'DATE: '.

 05 DATE-OUT PIC X(18) VALUE SPACES.

 05 FILLER PIC X(32) VALUE SPACES.

 05 FILLER PIC X(21) VALUE ' DELIVERY COSTS FOR '.

 05 FILLER PIC X(41) VALUE SPACES.

 05 FILLER PIC X(6) VALUE 'PAGE: '.

 05 PAGE-OUT PIC Z(4).

 05 FILLER PIC X(3) VALUE ' - '.

01 HEAD-3.

```

05 FILLER          PIC X(55) VALUE SPACES.
05 TITLE-OUT       PIC X(24) VALUE SPACES.
05 FILLER          PIC X(53) VALUE SPACES.
C1 HEAD-4.
05 FILLER          PIC X(49) VALUE SPACES.
05 FILLER          PIC X(79) VALUE '*****' .
*      DELIVERY   *****
'******'.
05 FILLER          PIC X(4) VALUE SPACES.
01 HEAD-5.
05 FILLER          PIC X(49) VALUE SPACES.
05 HEAD-OUT         PIC X(79).
05 FILLER          PIC X(4) VALUE SPACES.
01 DATA-LINE.
05 FILLER          PIC X(4) VALUE SPACES.
05 DESC             PIC X(20).
05 FILLER          PIC X(19) VALUE SPACES.
05 AMOUNTS OCCURS 5 TIMES.
10 FILLER          PIC X(6).
10 AMT              PIC -(8).--.
05 FILLER          PIC X(4).
01 ACCUMS.
05 ACCUM-A OCCURS 4 TIMES.
10 ACCUM-B OCCURS 5 TIMES.
15 ACCUM PIC S9(8)V99.
01 SUBTOTAL.
05 SUB-A OCCURS 4 TIMES.
10 SUB-B OCCURS 5 TIMES.
15 SUBS             PIC S9(8)V99.
01 HEAD-DATA-HOLD.
05 HEAD-DATA-A.
10 FILLER          PIC X(79) VALUE '-----'
*      TRANSMISSION   -----
*-----'.
10 FILLER          PIC X(79) VALUE '    ZONE 5      ZO
'NE 6           ZONE 7      ZONE 8
*      ZONE 9 '.
10 FILLER          PIC X(79) VALUE '-----'
*      DISTRIBUTION   -----
*-----'.
10 FILLER          PIC X(79) VALUE '    ZONE 5      ZO
'NE 6           ZONE 7      ZONE 8
*      ZONE 9 '.
10 FILLER          PIC X(79) VALUE '-----'
*      STORAGE        -----
*-----'.
10 FILLER          PIC X(79) VALUE '    BROMLEY      IDA
'SPENCE        DEVOG PARK
*      '.
10 FILLER          PIC X(79) VALUE '-----'

```

```

          ----- PUMP STATIONS -----
          -----
          10 FILLER      PIC X(79) VALUE ' COLD SPRINGS    IDA
          'SPENCE        DEVOG PARK           BOONE CN
          'TY            '.

          05 HEAD-DATA-B REDEFINES HEAD-DATA-A.
          10 HEAD-DATA      PIC X(79) OCCURS 8 TIMES.

01 PAGE-LETTER-HOLD.
05 PG-LTR-A      PIC X(4) VALUE 'ABCD'.
05 PG-LTR-B REDEFINES PG-LTR-A.
10 PAGE-LETTER PIC X OCCURS 4 TIMES.

PROCEDURE DIVISION.

OPEN-FILES.
  OPEN INPUT CARD-IN,
        COST-MATRIX-FILE,
        FINANCIAL-ACCT-TABFI,
  OUTPUT PRINT-OUT,
        PRINT-FILE.

READ-CARD-IN.
  READ CARD-IN AT END
    MOVE '*** NO DATE/TITLE CARD ** JOB ABORTED ***' TO
    PRINT-DATA,
    PERFORM PRINT-LINE-OUT,
    GO TO CLOSE-FILES.

  MOVE DATE-IN TO DATE-OUT.
  MOVE TITLE-IN TO TITLE-OUT.

SORT-CM-FILE.
  OPEN OUTPUT SORTED-FILE.
  SORT FILE-SORT ON ASCENDING KEY
    S-MATCH-CODE
  INPUT PROCEDURE
    READ-SORT-INPUT THRU READ-SORT-INPUT-END
  OUTPUT PROCEDURE
    WRITE-SORT-OUTPUT THRU WRITE-SORT-OUTPUT-END.
  CLOSE SORTED-FILE.
  OPEN INPUT SORTED-FILE.

ZERO-ACCUMS.
  MOVE ZEROS TO SUBTOTAL.
  MOVE MINUS-ONE TO SUBS (3, 4) SUBS (3, 5) SUBS (4, 1).

HOUSE-KEEP.
  MOVE SPACES TO PRINT-LINE-HOLD.
  MOVE ZEROS TO ACCUMS.
  MOVE MINUS-ONE TO ACCUM (3, 4) ACCUM (3, 5) ACCUM (4, 1).

READ-SORTED-FILE.
  READ SORTED-FILE AT END PERFORM WRITE-PRINT-FILE,
  GO TO WRITE-SUBTOTALS.

NEW-MATCH-CODE.
  MOVE MATCH-CODE TO MATCH-HOLD.
  MOVE ACCT-NR TO FA-NOMINAL.
  READ FINANCIAL-ACCT-TABFI INVALID KEY

```

```
MOVE SPACES TO FA-NAME (1), FA-NAME (2).
TOTAL-COST-CNTRS.
ADD AMTS (1) TO ACCUM (CC-POS2, CC-POS4)
SUBS (CC-POS2, CC-POS4).
PERFORM READ-SORTED-FILE.
IF MATCH-CODE EQUAL TO MATCH-HOLD THEN
GO TO TOTAL-COST-CNTRS.
WRITE-PRINT-FILE.
MOVE FA-NAME (1) TO DESC-HOLD (1).
MOVE FA-NAME (2) TO DESC-HOLD (2).
MOVE 1 TO K.
PERFORM MOVE-ACCUMS VARYING I FROM 1 BY 1 UNTIL I = 5.
WRITE PRINT-LINE-HOLD.
PERFORM HOUSE-KEEP.
RETURN-TO-NEW-MATCH-CODE.
GO TO NEW-MATCH-CODE.
MOVE-ACCUMS.
PERFORM MOVE-ACCUMS-A VARYING L FROM 1 BY 1 UNTIL L = 6.
MOVE-ACCUMS-A.
MOVE ACCUM (I, L) TO AMTS-HOLD (K).
ADD 1 TO K.
WRITE-SUBTOTALS.
WRITE PRINT-LINE-HOLD.
MOVE 1 TO K.
PERFORM MOVE-SUBS VARYING I FROM 1 BY 1 UNTIL I = 5.
MOVE 'TOTALS' TO DESC-HOLD (1).
MOVE SPACES TO DESC-HOLD (2).
WRITE PRINT-LINE-HOLD.
GO TO HOUSE-KEEP-1.
MOVE-SUBS.
PERFORM MOVE-SUBS-A VARYING L FROM 1 BY 1 UNTIL L = 6.
MOVE-SUBS-A.
MOVE SUBS (I, L) TO AMTS-HOLD (K).
ADD 1 TO K.
HOUSE-KEEP-1.
MOVE -1 TO J.
MOVE 0 TO O.
PERFORM START-PRINTING THRU START-PRINTING-END
VARYING I FROM 1 BY 1 UNTIL I = 5.
GO TO CLOSE-FILES.
START-PRINTING.
ADD 2 TO J, O.
PERFORM REWIND-PRINT-FILE.
WRITE-HEADINGS.
ADD 1 TO PAGE-CT.
MOVE 1 TO LINE-CT.
MOVE PAGE-CT TO PAGE-OUT.
MOVE PAGE-LETTER (I) TO PAGE-LETTER-OUT.
PERFORM MOVE-SPACES.
WRITE PRINT-LINE BEFORE NEW-PAGE.
```

```
MOVE HEAD-1 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-2 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-3 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
MOVE HEAD-4 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-DATA (J) TO HEAD-OUT.  
MOVE HEAD-5 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-DATA (D) TO HEAD-OUT.  
MOVE HEAD-5 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
READ-PRINT-FILE.  
    READ PRINT-FILE AT END GO TO START-PRINTING-END.  
MOVE-VALUES-OUT.  
    IF I = 1 THEN  
        MOVE 1 TO N.  
    IF I = 2 THEN  
        MOVE 6 TO N.  
    IF I = 3 THEN  
        MOVE 11 TO N.  
    IF I = 4 THEN  
        MOVE 16 TO N.  
    MOVE 1 TO M.  
    PERFORM MOVE-VALUES 5 TIMES.  
    MOVE DESC-HOLD (1) TO DESC.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    IF DESC-HOLD (2) NOT = SPACES THEN  
        MOVE SPACES TO DATA-LINE  
        MOVE DESC-HOLD (2) TO DESC  
        MOVE DATA-LINE TO PRINT-DATA  
        PERFORM PRINT-LINE-OUT.  
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
    IF LINE-CT > MAX-LINE THEN  
        PERFORM WRITE-HEADINGS.  
    PERFORM READ-PRINT-FILE.  
    IF PRINT-LINE-HOLD = SPACES THEN  
        PERFORM MOVE-SPACES THRU PRINT-LINE-OUT  
        PERFORM READ-PRINT-FILE.  
    GO TO MOVE-VALUES-OUT.  
START-PRINTING-END.  
    EXIT.  
MOVE-VALUES.  
    IF AMTS-HOLD (N) = MINUS-ONE THEN  
        MOVE ZERO TO AMT OF AMOUNTS (M)
```

```
ELSE MOVE AMTS-HOLD (N) TO AMT OF AMOUNTS (M)
    ADD 1 TO M.
    ADD 1 TO N.
REWIND-PRINT-FILE.
    MOVE ZERO TO PAGE-CT.
    CLOSE PRINT-FILE.
    OPEN INPUT PRINT-FILE.
MOVE-SPACES.
    MOVE SPACES TO PRINT-LINE.
PRINT-LINE-OUT.
    WRITE PRINT-LINE BEFORE 1.
    ADD 1 TO LINE-CT.
READ-SORT-INPUT.
    READ COST-MATRIX-FILE AT END
        GO TO READ-SORT-INPUT-END.
    IF MATCH-CODE-POS1 NOT = TO MATCH-POS1
        THEN GO TO READ-SORT-INPUT.
    RELEASE SORT-INPUT FRUM CM-REC.
    GO TO READ-SORT-INPUT.
READ-SORT-INPUT-END.
EXIT.
WRITE-SORT-OUTPUT.
    RETURN FILE-SORT RECORD INTO INPUT-REC
        AT END GO TO WRITE-SORT-OUTPUT-END.
    WRITE INPUT-REC.
    GO TO WRITE-SORT-OUTPUT.
WRITE-SORT-OUTPUT-END.
EXIT.
CLOSE-FILES.
    CLOSE SURTED-FILE,
        CARD-IN,
        COST-MATRIX-FILE,
        FINANCIAL-ACCT-TABFI,
        PRINT-FILE,
        PRINT-DUT.
E-O-J.
STOP RUN.
```

PGMNAME=WUASLV43
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASLV43.
AUTHOR. ACT SYSTEMS INC
SUITE 200
807 W MORSE BLVD
WINTER PARK, FLA 32789.
DATE-WRITTEN. JAN 1978.
DATE-COMPILED.
REMARKS. THIS PROGRAM SORTS THE COST MATRIX FILE BY
MATCH CODE AND PRINTS OUT LEVEL IV (DELIVERY)
REPORT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
SELECT CARD-IN
ASSIGN TO UR-2540R-S-CARDIN
ACCESS MODE IS SEQUENTIAL.
SELECT PRINT-OUT
ASSIGN TO UR-1403-S-PRINTOUT
ACCESS MODE IS SEQUENTIAL.
SELECT COST-MATRIX-FILE
ASSIGN TO DA-3330-I-CMFILE
ACCESS MODE IS SEQUENTIAL
RECORD KEY IS CM-NUMBER.
SELECT PRINT-FILE
ASSIGN TO UT-3330-S-PRINTF1
ACCESS MODE IS SEQUENTIAL.
SELECT FINANCIAL-ACCT-TABFI
ASSIGN TO DA-3330-I-FATABFI
ACCESS MODE IS RANDOM
NOMINAL KEY IS FA-NOMINAL
RECORD KEY IS FA-NUMBER.
SELECT FILE-SORT
ASSIGN TO UT-3330-S-FILESORT
ACCESS MODE IS SEQUENTIAL.
SELECT SORTED-FILE
ASSIGN TO UT-3330-S-SORTEDFI
ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE OMITTED

DATA RECORD IS INPUT-CARD.

01 INPUT-CARD.

 05 DATE-IN PIC X(18).

 05 TITLE-IN PIC X(24).

 05 FILLER PIC X(38).

FD PRINT-OUT

RECORD CONTAINS 133 CHARACTERS

LABEL RECORDS ARE OMITTED

DATA RECORD IS PRINT-LINE.

01 PRINT-LINE.

 05 FILLER PIC X.

 05 PRINT-DATA PIC X(132).

FD COST-MATRIX-FILE

RECORD CONTAINS 80 CHARACTERS

BLOCK CONTAINS 5 RECORDS

LABEL RECORDS ARE STANDARD

DATA RECORD IS CM-REC.

01 CM-REC.

 05 MATCH-CODE-POS1 PIC X.

 05 FILLER PIC X(3).

 05 CM-NUMBER PIC X(10).

 05 FILLER PIC X(66).

FD PRINT-FILE

BLOCK CONTAINS 6 RECORDS

RECORD CONTAINS 220 CHARACTERS

LABEL RECORDS ARE STANDARD

DATA RECORD IS PRINT-LINE-HOLD.

01 PRINT-LINE-HOLD.

 05 DESC-HOLD PIC X(20) OCCURS 2 TIMES.

 05 AMTS-HOLD PIC S9(8)V99 OCCURS 18 TIMES.

FD FINANCIAL-ACCT-TABFI

BLOCK CONTAINS 8 RECORDS

RECORD CONTAINS 58 CHARACTERS

LABEL RECORDS ARE STANDARD

DATA RECORD IS FA-REC.

01 FA-REC.

 05 FA-NUMBER PIC X(10).

 05 FA-NAME PIC X(20) OCCURS 2 TIMES.

 05 FILLER PIC X(8).

SD FILE-SORT

RECORD CONTAINS 80 CHARACTERS

DATA RECORD IS SORT-INPUT.

01 SORT-INPUT.

 05 S-MATCH-CODE PIC X(4).

 05 FILLER PIC X(16).

 05 FILLER PIC X(60).

FD SORTED-FILE

RECORD CONTAINS 80 CHARACTERS

BLOCK CONTAINS 5 RECORDS

LABEL RECORDS ARE STANDARD

```

DATA RECORD IS INPUT-REC.
01 INPUT-REC.
  05 MATCH-CODE      PIC X(4).
  05 ACCT-NO        PIC X(10).
  05 COST-CNTR.
    10 FILLER        PIC X.
    10 CC-POS2       PIC 9.
    10 FILLER        PIC X.
    10 CC-POS4       PIC 9.
  05 COST-CAT       PIC XX.
  05 AMTS          PIC S9(8)V99 OCCURS 6 TIMES.

*****
*      THE ABOVE AMTS ARE AS FOLLOWS
*      (1) -- CURRENT COST AMOUNT
*      (2) -- PREVIOUS MONTH COST AMOUNT
*      (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)
*      (4) -- BUDGET AMOUNT
*      (5) -- BUDGET AMOUNT YEAR TO DATE
*      (6) -- PREVIOUS YEAR COST AMOUNT (PYC)
*****

WORKING-STORAGE SECTION.
 77 PAGE-CT          PIC 9(4) VALUE ZERO.
 77 LINE-CT          PIC 9(2) VALUE ZERO.
 77 FA-NOMINAL       PIC X(10) VALUE SPACES.
 77 I                PIC 9.
 77 J                PIC S99.
 77 K                PIC 99.
 77 L                PIC 99.
 77 M                PIC 99.
 77 N                PIC 99.
 77 O                PIC 99.
 77 MATCH-HOLD        PIC X(4) VALUE SPACES.
 77 MATCH-POS1        PIC X VALUE '3'.
 77 MAX-LINE         PIC 9(2) COMP VALUE 56.
 77 MINUS-ONE        PIC S9(8)V99 VALUE -0.01.

01 HEAD-1.
  05 FILLER          PIC X(57) VALUE SPACES.
  05 FILLER          PIC X(18) VALUE 'LEVEL IV REPORT OF'.
  05 FILLER          PIC X(57) VALUE SPACES.

01 HEAD-2.
  05 FILLER          PIC X(6) VALUE 'DATE: '.
  05 DATE-OUT         PIC X(18) VALUE SPACES.
  05 FILLER          PIC X(32) VALUE SPACES.
  05 FILLER          PIC X(21) VALUE ' DELIVERY COSTS FOR '.
  05 FILLER          PIC X(41) VALUE SPACES.
  05 FILLER          PIC X(6) VALUE 'PAGE: '.
  05 PAGE-OUT         PIC Z(4).
  05 FILLER          PIC X(3) VALUE ' - '.
  05 PAGE-LETTER-OUT PIC X.

01 HEAD-3.

```

```

05 FILLER          PIC X(55) VALUE SPACES.
05 TITLE-OUT       PIC X(24) VALUE SPACES.
05 FILLER          PIC X(53) VALUE SPACES.
01 HEAD-4.
05 FILLER          PIC X(2) VALUE SPACES.
05 FILLER          PIC X(26) VALUE 'COST CATEGORY DESCRIPTI
-                   'ONS'.
05 FILLER          PIC X(6) VALUE SPACES.
05 FILLER          PIC X(96) VALUE '*****' D E L I V E R Y *
-                   '*****'.
05 FILLER          PIC X(2) VALUE SPACES.
01 HEAD-5.
05 FILLER          PIC X(34) VALUE SPACES.
05 HEAU-OUT         PIC X(96).
05 FILLER          PIC X(2) VALUE SPACES.
01 DATA-LINE.
05 FILLER          PIC X(4) VALUE SPACES.
05 DESC             PIC X(20).
05 FILLER          PIC X(2) VALUE SPACES.
05 AMOUNTS OCCURS 6 TIMES.
    10 FILLER        PIC X(6).
    10 AMT            PIC -(8).--.
05 FILLER          PIC X(2).
01 ACCUMS.
    05 ACCUM-A OCCURS 4 TIMES.
        10 ACCUM-B OCCURS 6 TIMES.
            15 ACCUM PIC S9(8)V99.
01 SUBTOTAL.
    05 SUB-A OCCURS 4 TIMES.
        10 SUB-B OCCURS 6 TIMES.
            15 SUBS      PIC S9(8)V99.
01 HEAD-DATA-HOLD.
    05 HEAD-DATA-A.
        10 FILLER        PIC X(96) VALUE '-----' TRANSMI
-                   ----- ----- DISTRIBU
-                   'SSION -----'.
        10 FILLER        PIC X(96) VALUE '     ZONE 1      ZO
-                   'NE 2      ZONE 3      ZONE 1      ZON
-                   'E 2      ZONE 3 '.
        10 FILLER        PIC X(96) VALUE '-----'
-                   ----- STORAGE -----'
-                   '-----'.
        10 FILLER        PIC X(96) VALUE '     DUDLEY PIKE      BARRIN
-                   'GTON RD      EDGEWOOD      KENTON LANDS      IND
-                   'PARK        OBLIQUE ST'.
        10 FILLER        PIC X(96) VALUE '-----'
-                   ----- PUMP STATION -----'
-                   '-----'.
        10 FILLER        PIC X(96) VALUE '     PLANT      DIXI

```

```

-           'E HWY          KYLES LANE          DUDLEY PIKE      TURKE
-           'Y FOOT          LAFAYETTE'.
05 HEAD-DATA-B REDEFINES HEAD-DATA-A.
10 HEAD-DATA    PIC X(96) OCCURS 6 TIMES.
01 PAGE-LETTER-HOLD.
05 PG-LTR-A        PIC X(3) VALUE 'ABC'.
05 PG-LTR-B REDEFINES PG-LTR-A.
10 PAGE-LETTER PIC X OCCURS 3 TIMES.

PROCEDURE DIVISION.

OPEN-FILES.
  OPEN INPUT CARD-IN,
        COST-MATRIX-FILE,
        FINANCIAL-ACCT-TABFI,
  OUTPUT PRINT-OUT,
        PRINT-FILE.

READ-CARD-IN.
  READ CARD-IN AT END
    MOVE '*** NO DATE/TITLE CARD ** JOB ABORTED ***' TO
    PRINT-DATA,
    PERFORM PRINT-LINE-OUT.
    GO TO CLOSE-FILES.
  MOVE DATE-IN TO DATE-OUT.
  MOVE TITLE-IN TO TITLE-OUT.

SORT-CM-FILE.
  OPEN OUTPUT SORTED-FILE.
  SORT FILE-SORT ON ASCENDING KEY
    S-MATCH-CODE
  INPUT PROCEDURE
    READ-SORT-INPUT THRU READ-SORT-INPUT-END
  OUTPUT PROCEDURE
    WRITE-SORT-OUTPUT THRU WRITE-SORT-OUTPUT-END.
  CLOSE SORTED-FILE.
  OPEN INPUT SORTED-FILE.

ZERO-ACCUMS.
  MOVE ZEROS TO SUBTOTAL.
  MOVE MINUS-ONE TO SUBS (1, 4) SUBS (1, 5) SUBS (1, 6)
                        SUBS (2, 4) SUBS (2, 5) SUBS (2, 6).

HOUSE-KEEP.
  MOVE SPACES TO PRINT-LINE-HOLD.
  MOVE ZEROS TO ACCUMS.
  MOVE MINUS-ONE TO ACCUM (1, 4) ACCUM (1, 5) ACCUM (1, 6)
                        ACCUM (2, 4) ACCUM (2, 5) ACCUM (2, 6).

READ-SORTED-FILE.
  READ SORTED-FILE AT END PERFORM WRITE-PRINT-FILE,
  GO TO WRITE-SUBTOTALS.

NEW-MATCH-CODE.
  MOVE MATCH-CODE TO MATCH-HOLD.
  MOVE ACCT-NO TO FA-NOMINAL.
  READ FINANCIAL-ACCT-TABFI INVALID KEY
    MOVE SPACES TO FA-NAME (1), FA-NAME (2).

```

```

TOTAL-COST-CNTRS.
    ADD AMTS (1) TO ACCUM (CC-POS2, CC-POS4)
        SUBS (CC-POS2, CC-POS4).
    PERFORM READ-SORTED-FILE.
    IF MATCH-CODE EQUAL TO MATCH-HOLD THEN
        GO TO TOTAL-COST-CNTRS.

WRITE-PRINT-FILE.
    MOVE FA-NAME (1) TO DESC-HOLD (1).
    MOVE FA-NAME (2) TO DESC-HOLD (2).
    MOVE 1 TO K.
    PERFORM MOVE-ACCUMS VARYING I FROM 1 BY 1 UNTIL I = 5.
    WRITE PRINT-LINE-HOLD.
    PERFORM HOUSE-KEEP.
RETURN-TO-NEW-MATCH-CODE.
    GO TO NEW-MATCH-CODE.

MOVE-ACCUMS.
    PERFORM MOVE-ACCUMS-A VARYING L FROM 1 BY 1 UNTIL L = 7.

MOVE-ACCUMS-A.
    IF ACCUM (I, L) NOT = MINUS-ONE THEN
        MOVE ACCUM (I, L) TO AMTS-HOLD (K)
        ADD 1 TO K.

WRITE-SUBTOTALS.
    WRITE PRINT-LINE-HOLD.
    MOVE 1 TO K.
    PERFORM MOVE-SUBS VARYING I FROM 1 BY 1 UNTIL I = 5.
    MOVE 'TOTALS' TO DESC-HOLD (1).
    MOVE SPACES TO DESC-HOLD (2).
    WRITE PRINT-LINE-HOLD.
    GO TO HOUSE-KEEP-1.

MOVE-SUBS.
    PERFORM MOVE-SUBS-A VARYING L FROM 1 BY 1 UNTIL L = 7.

MOVE-SUBS-A.
    IF SUBS (I, L) NOT = MINUS-ONE THEN
        MOVE SUBS (I, L) TO AMTS-HOLD (K)
        ADD 1 TO K.

HOUSE-KEEP-1.
    MOVE -1 TO J.
    MOVE 0 TO D.
    PERFORM START-PRINTING THRU START-PRINTING-END
        VARYING I FROM 1 BY 1 UNTIL I = 4.
    GO TO CLOSE-FILES.

START-PRINTING.
    ADD 2 TO J, D.
    PERFORM REWIND-PRINT-FILE.

WRITE-HEADINGS.
    ADD 1 TO PAGE-CT.
    MOVE 1 TO LINE-CT.
    MOVE PAGE-CT TO PAGE-OUT.
    MOVE PAGE-LETTER (I) TO PAGE-LETTER-OUT.
    PERFORM MOVE-SPACES.

```

```
WRITE PRINT-LINE BEFORE NEW-PAGE.  
MOVE HEAD-1 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-2 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-3 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
MOVE HEAD-4 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-DATA (J) TO HEAD-OUT.  
MOVE HEAD-5 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-DATA (O) TO HEAD-OUT.  
MOVE HEAD-5 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
READ-PRINT-FILE.  
    READ PRINT-FILE AT END GO TO START-PRINTING-END.  
MOVE-VALUES-OUT.  
    IF I = 1 THEN  
        MOVE 1 TO N.  
    IF I = 2 THEN  
        MOVE 7 TO N.  
    IF I = 3 THEN  
        MOVE 13 TO N.  
    MOVE 1 TO M.  
    PERFORM MOVE-VALUES 6 TIMES.  
    MOVE DESC-HOLD (1) TO DESC.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    IF DESC-HOLD (2) NOT = SPACES THEN  
        MOVE SPACES TO DATA-LINE  
        MOVE DESC-HOLD (2) TO DESC  
        MOVE DATA-LINE TO PRINT-DATA  
        PERFORM PRINT-LINE-OUT.  
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
    IF LINE-CT > MAX-LINE THEN  
        PERFORM WRITE-HEADINGS.  
    PERFORM READ-PRINT-FILE.  
    GO TO MOVE-VALUES-OUT.  
START-PRINTING-END.  
    EXIT.  
MOVE-VALUES.  
    MOVE AMTS-HOLD (N) TO AMT OF AMOUNTS (M).  
    ADD 1 TO N.  
    ADD 1 TO M.  
REWIND-PRINT-FILE.  
    MOVE ZERO TO PAGE-CT.  
    CLOSE      PRINT-FILE.
```

```
OPEN INPUT PRINT-FILE.  
MOVE-SPACES.  
    MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
    ADD 1 TO LINE-CT.  
READ-SORT-INPUT.  
    READ COST-MATRIX-FILE AT END  
        GO TO READ-SORT-INPUT-END.  
    IF MATCH-CODE-POS1 NOT = TO MATCH-POS1  
        THEN GO TO READ-SORT-INPUT.  
    RELEASE SORT-INPUT FROM CM-REC.  
    GO TO READ-SORT-INPUT.  
READ-SORT-INPUT-END.  
    EXIT.  
WRITE-SORT-OUTPUT.  
    RETURN FILE-SORT RECORD INTO INPUT-REC  
        AT END GO TO WRITE-SORT-OUTPUT-END.  
    WRITE INPUT-REC.  
    GO TO WRITE-SORT-OUTPUT.  
WRITE-SORT-OUTPUT-END.  
    EXIT.  
CLOSE-FILES.  
    CLOSE    SORTED-FILE,  
            CARD-IN,  
            COST-MATRIX-FILE,  
            FINANCIAL-ACCT-TABFI..  
            PRINT-FILE,  
            PRINT-OUT.  
E-O-J.  
    STOP RUN.
```

```
PGMNAME=WUASLV42
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASLV42.
AUTHOR. ACT SYSTEMS INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FLA 32789.
DATE-WRITTEN. JAN 1978.
DATE-COMPILED.
REMARKS. THIS PROGRAM SORTS THE COST MATRIX FILE BY
        MATCH CODE AND PRINTS OUT LEVEL IV (TREATMENT)
        REPORT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS SEQUENTIAL
        RECORD KEY IS CM-NUMBER.
    SELECT FINANCIAL-ACT-TABFI
        ASSIGN TO DA-3330-I-FATABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS FA-NOMINAL
        RECORD KEY IS FA-NUMBER.
    SELECT FILE-SORT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
    SELECT SORTED-FILE
        ASSIGN TO UT-3330-S-SORTEDFI
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE OMITTED
    DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
    05 DATE-IN          PIC X(18).
```

```

      05 TITLE-IN          PIC X(24).
      05 FILLER           PIC X(38).
FD   PRINT-OUT
      RECORD CONTAINS 133 CHARACTERS
      LABEL RECORDS ARE OMITTED
      DATA RECORD IS PRINT-LINE.
01   PRINT-LINE.
      05 FILLER           PIC X.
      05 PRINT-DATA        PIC X(132).
FD   COST-MATRIX-FILE
      RECORD CONTAINS 80 CHARACTERS
      BLOCK CONTAINS 5 RECORDS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS CM-REC.
01   CM-REC.
      05 MATCH-CODE-PDS1  PIC X.
      05 FILLER           PIC X(3).
      05 CM-NUMBER        PIC X(10).
      05 FILLER           PIC X(66).
FD   FINANCIAL-ACCT-TABFI
      BLOCK CONTAINS 8 RECORDS
      RECORD CONTAINS 58 CHARACTERS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS FA-REC.
01   FA-REC.
      05 FA-NUMBER        PIC X(10).
      05 FA-NAME          PIC X(20) OCCURS 2 TIMES.
      05 FILLER           PIC X(8).
SD   FILE-SORT
      RECORD CONTAINS 80 CHARACTERS
      DATA RECORD IS SORT-INPUT.
01   SORT-INPUT.
      05 S-MATCH-CODE    PIC X(4).
      05 FILLER           PIC X(16).
      05 FILLER           PIC X(60).
FD   SORTED-FILE
      RECORD CONTAINS 80 CHARACTERS
      BLOCK CONTAINS 5 RECORDS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS INPUT-REC.
01   INPUT-REC.
      05 MATCH-CODE       PIC X(4).
      05 ACCT-NO          PIC X(10).
      05 COST-CNTR.
          10 FILLER         PIC X.
          10 CC-PDS2        PIC 9.
          10 FILLER         PIC XX.
      05 COST-CAT         PIC XX.
      05 AMTS             PIC S9(8)V99 OCCURS 6 TIMES.
*****
```

* THE ABOVE AMTS ARE AS FOLLOWS
 * (1) -- CURRENT COST AMOUNT
 * (2) -- PREVIOUS MONTH COST AMOUNT
 * (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)
 * (4) -- BUDGET AMOUNT
 * (5) -- BUDGET AMOUNT YEAR TO DATE
 * (6) -- PREVIOUS YEAR COST AMOUNT (PYC)

WORKING-STORAGE SECTION.

```

77 PAGE-CT          PIC 9(4) VALUE ZERO.
77 LINE-CT          PIC 9(2) VALUE ZERO.
77 FA-NOMINAL      PIC X(10) VALUE SPACES.
77 I                PIC 9.
77 MATCH-HOLD       PIC X(4) VALUE SPACES.
77 MATCH-POS1       PIC X     VALUE '2'.
77 MAX-LINE         PIC 9(2) COMP VALUE 56.
01 HEAD-1.
  05 FILLER          PIC X(57) VALUE SPACES.
  05 FILLER          PIC X(18) VALUE 'LEVEL IV REPORT OF'.
  05 FILLER          PIC X(57) VALUE SPACES.
01 HEAD-2.
  05 FILLER          PIC X(6)  VALUE 'DATE: '.
  05 DATE-OUT        PIC X(18) VALUE SPACES.
  05 FILLER          PIC X(32) VALUE SPACES.
  05 FILLER          PIC X(21) VALUE ' TREATMENT COSTS FOR '.
  05 FILLER          PIC X(45) VALUE SPACES.
  05 FILLER          PIC X(6)  VALUE 'PAGE: '.
  05 PAGE-OUT        PIC Z(4).
01 HEAD-3.
  05 FILLER          PIC X(55) VALUE SPACES.
  05 TITLE-OUT        PIC X(24) VALUE SPACES.
  05 FILLER          PIC X(53) VALUE SPACES.
01 HEAD-4.
  05 FILLER          PIC X(27) VALUE SPACES.
  05 FILLER          PIC X(26) VALUE 'COST CATEGORY DESCRIPTI
  -                  'ONS'.
  05 FILLER          PIC X(18) VALUE SPACES.
  05 FILLER          PIC X(39) VALUE '***** T R E A T M E
  -                  ' N T *****'.
  05 FILLER          PIC X(22) VALUE SPACES.
01 HEAD-5.
  05 FILLER          PIC X(71) VALUE SPACES.
  05 FILLER          PIC X(12) VALUE 'PURIFICATION'.
  05 FILLER          PIC X(13) VALUE SPACES.
  05 FILLER          PIC X(14) VALUE 'PLANT OVERHEAD'.
  05 FILLER          PIC X(22) VALUE SPACES.
01 HEAD-6.
  05 FILLER          PIC X(75) VALUE SPACES.
  05 FILLER          PIC X(5)  VALUE 'RIVER'.
  05 FILLER          PIC X(20) VALUE SPACES.

```

```

      05 FILLER          PIC X(5) VALUE 'RIVER'.
      05 FILLER          PIC X(27) VALUE SPACES.
01 DATA-LINE.
      05 FILLER          PIC X(30).
      05 DESC            PIC X(20).
      05 FILLER          PIC X(6).
      05 AMOUNTS OCCURS 2 TIMES.
          10 FILLER        PIC X(14).
          10 AMT           PIC -(8).--.
      05 FILLER          PIC X(26).
01 ACCUMS.
      05 ACCUM           PIC S9(8)V99 OCCURS 2 TIMES.
01 SUBTOTALS.
      05 SUBS            PIC S9(8)V99 OCCURS 2 TIMES.
PROCEDURE DIVISION.
OPEN-FILES.
    OPEN INPUT CARD-IN,
           COST-MATRIX-FILE,
           FINANCIAL-ACCT-TABFI,
           OUTPUT PRINT-OUT.
READ-CARD-IN.
    READ CARD-IN AT END
        MOVE '*** NO DATE/TITLE CARD ** JOB ABORTED ***' TO
        PRINT-DATA,
        PERFORM PRINT-LINE-OUT,
        GO TO CLOSE-FILES.
    MOVE DATE-IN TO DATE-OUT.
    MOVE TITLE-IN TO TITLE-OUT.
SORT-CM-FILE.
    OPEN OUTPUT SORTED-FILE.
    SORT FILE-SORT ON ASCENDING KEY
        S-MATCH-CODE
    INPUT PROCEDURE
        READ-SORT-INPUT THRU READ-SORT-INPUT-END
    OUTPUT PROCEDURE
        WRITE-SORT-OUTPUT THRU WRITE-SORT-OUTPUT-END.
CLOSE SURTED-FILE.
OPEN INPUT SORTED-FILE.
WRITE-HEADINGS.
    MOVE ZERO TO LINE-CT.
    ADD 1 TO PAGE-CT.
    MOVE PAGE-CT TO PAGE-OUT.
    PERFORM MOVE-SPACES.
    WRITE PRINT-LINE BEFORE NEW-PAGE.
    MOVE HEAD-1 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-2 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-3 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.

```

```
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
MOVE HEAD-4 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-5 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-6 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
ZERO-ACCUMS.  
    MOVE ZEROS TO SUBTOTALS.  
HOUSE-KEEP.  
    MOVE SPACES TO DATA-LINE.  
    MOVE ZEROS TO ACCUMS.  
READ-SORTED-FILE.  
    READ SORTED-FILE AT END PERFORM WRITE-LINE-OUT,  
    GO TO WRITE-SUBTOTALS.  
NEW-MATCH-CODE.  
    MOVE MATCH-CODE TO MATCH-HOLD.  
    MOVE ACCT-NO TO FA-NOMINAL.  
    READ FINANCIAL-ACCT-TABFI INVALID KEY  
        MOVE SPACES TO FA-NAME (1), FA-NAME (2).  
TOTAL-COST-CNTRS.  
    ADD AMTS (1) TO ACCUM (CC-POS2), SUBS (CC-POS2).  
    PERFORM READ-SORTED-FILE.  
    IF MATCH-CODE EQUAL TO MATCH-HOLD THEN  
        GO TO TOTAL-COST-CNTRS.  
WRITE-LINE-OUT.  
    MOVE FA-NAME (1) TO DESC.  
    PERFORM MOVE-ACCUMS VARYING I FROM 1 BY 1 UNTIL I = 3.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.  
    IF FA-NAME (2) NOT EQUAL TO SPACES THEN  
        MOVE SPACES TO DATA-LINE.  
        MOVE FA-NAME (2) TO DESC.  
        MOVE DATA-LINE TO PRINT-DATA,  
        PERFORM PRINT-LINE-OUT.  
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
    PERFORM HOUSE-KEEP.  
CHECK-FOR-NEW-HEADING.  
    IF LINE-CT NOT < MAX-LINE THEN  
        PERFORM WRITE-HEADINGS.  
    GO TO NEW-MATCH-CODE.  
WRITE-SUBTOTALS.  
    PERFORM MOVE-SPACES.  
    PERFORM PRINT-LINE-OUT 2 TIMES.  
    PERFORM HOUSE-KEEP.  
    MOVE 'TOTALS' TO DESC.  
    PERFORM MOVE-AMOUNTS VARYING I FROM 1 BY 1 UNTIL I = 3.  
    MOVE DATA-LINE TO PRINT-DATA.  
    PERFORM PRINT-LINE-OUT.
```

```
GO TO CLOSE-FILES.  
MOVE-SPACES.  
    MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
    ADD 1 TO LINE-CT.  
MOVE-ACCUMS.  
    MOVE ACCUM (I) TO AMT OF AMOUNTS (I).  
MOVE-AMOUNTS.  
    MOVE SUBS (I) TO AMT OF AMOUNTS (I).  
READ-SORT-INPUT.  
    READ COST-MATRIX-FILE AT END  
        GO TO READ-SORT-INPUT-END.  
    IF MATCH-CODE-POS1 NOT = TO MATCH-POS1  
        THEN GO TO READ-SORT-INPUT.  
    RELEASE SORT-INPUT FROM CM-REC.  
    GO TO READ-SORT-INPUT.  
READ-SORT-INPUT-END.  
    EXIT.  
WRITE-SORT-OUTPUT.  
    RETURN FILE-SORT RECORD INTO INPUT-REC  
        AT END GO TO WRITE-SORT-OUTPUT-END.  
    WRITE INPUT-REC.  
    GO TO WRITE-SORT-OUTPUT.  
WRITE-SORT-OUTPUT-END.  
    EXIT.  
CLOSE-FILES.  
    CLOSE SURTED-FILE,  
        CARD-IN,  
        COST-MATRIX-FILE,  
        FINANCIAL-ACCT-TABFI.  
        PRINT-OUT.  
E-O-J.  
    STOP RUN.
```

```
PGMNAME=WUASLV41
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASLV41.
AUTHOR. ACT SYSTEMS INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FLA 32789.
DATE-WRITTEN. JAN 1978.
DATE-COMPILED.
REMARKS.    THIS PROGRAM SORTS THE COST MATRIX FILE BY
            MATCH CODE AND PRINTS OUT LEVEL IV (ACQUISITION)
            REPORT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTDUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS SEQUENTIAL
        RECORD KEY IS CM-NUMBER.
    SELECT FINANCIAL-ACCT-TABFI
        ASSIGN TO DA-3330-I-FATABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS FA-NOMINAL
        RECORD KEY IS FA-NUMBER.
    SELECT FILE-SORT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
    SELECT SORTED-FILE
        ASSIGN TO UT-3330-S-SORTEDFI
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE OMITTED
    DATA RECORD IS INPUT-CARD.
01 INPUT-CARD.
    05 DATE-IN          PIC X(18).
```

```

      05  TITLE-IN          PIC X(24).
      05  FILLER           PIC X(38).
FD   PRINT-OUT
      RECORD CONTAINS 133 CHARACTERS
      LABEL RECORDS ARE OMITTED
      DATA RECORD IS PRINT-LINE.
01   PRINT-LINE.
      05  FILLER           PIC X.
      05  PRINT-DATA        PIC X(132).
FD   COST-MATRIX-FILE
      RECORD CONTAINS 80 CHARACTERS
      BLOCK CONTAINS 5 RECORDS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS CM-REC.
01   CM-REC.
      05  MATCH-CODE-POS1  PIC X.
      05  FILLER           PIC X(3).
      05  CM-NUMBER         PIC X(10).
      05  FILLER           PIC X(66).
FD   FINANCIAL-ACCT-TABFI
      BLOCK CONTAINS 8 RECORDS
      RECORD CONTAINS 58 CHARACTERS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS FA-REC.
01   FA-REC.
      05  FA-NUMBER         PIC X(10).
      05  FA-NAME            PIC X(20) OCCURS 2 TIMES.
      05  FILLER           PIC X(8).
SD   FILE-SORT
      RECORD CONTAINS 80 CHARACTERS
      DATA RECORD IS SORT-INPUT.
01   SORT-INPUT.
      05  S-MATCH-CODE     PIC X(4).
      05  FILLER           PIC X(16).
      05  FILLER           PIC X(60).
FD   SORTED-FILE
      RECORD CONTAINS 80 CHARACTERS
      BLOCK CUNTAINS 5 RECORDS
      LABEL RECORDS ARE STANDARD
      DATA RECORD IS INPUT-REC.
01   INPUT-REC.
      05  MATCH-CODE        PIC X(4).
      05  ACCT-NO           PIC X(10).
      05  COST-CNTR.
          10  FILLER         PIC X.
          10  CC-POS2         PIC 9.
          10  FILLER         PIC XX.
      05  COST-CAT          PIC XX.
      05  AMTS              PIC S9(8)V99 OCCURS 6 TIMES.

```

* THE ABOVE AMTS ARE AS FOLLOWS
 * (1) -- CURRENT COST AMOUNT
 * (2) -- PREVIOUS MONTH COST AMOUNT
 * (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)
 * (4) -- BUDGET AMOUNT
 * (5) -- BUDGET AMOUNT YEAR TO DATE
 * (6) -- PREVIOUS YEAR COST AMOUNT (PYC)

WORKING-STORAGE SECTION.

```

77 PAGE-CT          PIC 9(4) VALUE ZERO.
77 LINE-CT          PIC 9(2) VALUE ZERO.
77 FA-NOMINAL      PIC X(10) VALUE SPACES.
77 I                PIC 9.
77 MATCH-HOLD       PIC X(4) VALUE SPACES.
77 MATCH-POS1       PIC X     VALUE '1'.
77 MAX-LINE         PIC 9(2) COMP VALUE 56.
01 HEAD-1.
  05 FILLER          PIC X(57) VALUE SPACES.
  05 FILLER          PIC X(18) VALUE 'LEVEL IV REPORT OF'.
  05 FILLER          PIC X(57) VALUE SPACES.
01 HEAD-2.
  05 FILLER          PIC X(6)  VALUE 'DATE: '.
  05 DATE-OUT        PIC X(18) VALUE SPACES.
  05 FILLER          PIC X(32) VALUE SPACES.
  05 FILLER          PIC X(21) VALUE 'ACQUISITION COSTS FOR'.
  05 FILLER          PIC X(45) VALUE SPACES.
  05 FILLER          PIC X(6)  VALUE 'PAGE: '.
  05 PAGE-OUT        PIC Z(4).
01 HEAD-3.
  05 FILLER          PIC X(55) VALUE SPACES.
  05 TITLE-OUT        PIC X(24) VALUE SPACES.
  05 FILLER          PIC X(53) VALUE SPACES.
01 HEAD-4.
  05 FILLER          PIC X(5)  VALUE SPACES.
  05 FILLER          PIC X(26) VALUE 'COST CATEGORY DESCRIPTI'.
  -                 'ONS'.
  05 FILLER          PIC X(14) VALUE SPACES.
  05 FILLER          PIC X(82) VALUE '*****'.
  -                 '*** A C Q U I S I T'.
  -                 'I O N *****'.
  -                 '*****'.
01 HEAD-5.
  05 FILLER          PIC X(5)  VALUE SPACES.
  05 FILLER          PIC X(49) VALUE SPACES.
  05 FILLER          PIC X(5)  VALUE 'RIVER'.
  05 FILLER          PIC X(14) VALUE SPACES.
  05 FILLER          PIC X(12) VALUE 'TRANSMISSION'.
  05 FILLER          PIC X(11) VALUE SPACES.
  05 FILLER          PIC X(12) VALUE 'PUMP STATION'.
  05 FILLER          PIC X(9)  VALUE SPACES.

```

```

      05 FILLER          PIC X(15) VALUE 'PURCHASED WATER'.
      05 FILLER          PIC X(5)  VALUE SPACES.
01 DATA-LINE.
      05 FILLER          PIC X(8).
      05 DESC             PIC X(20).
      05 FILLER          PIC X(4).
      05 AMOUNTS OCCURS 4 TIMES.
          10 FILLER        PIC X(12).
          10 AMT            PIC -(8).--.
      05 FILLER          PIC X(8).
01 ACCUMS.
      05 ACCUM           PIC S9(8)V99 OCCURS 4 TIMES.
01 SUBTOTALS.
      05 SUBS            PIC S9(8)V99 OCCURS 4 TIMES.
PROCEDURE DIVISION.
OPEN-FILES.
    OPEN INPUT CARD-IN,
          COST-MATRIX-FILE,
          FINANCIAL-ACCT-TABFI,
    OUTPUT PRINT-OUT.
READ-CARD-IN.
    READ CARD-IN AT END
        MOVE '*** NO DATE/TITLE CARD ** JOB ABORTED ***' TO
        PRINT-DATA,
        PERFORM PRINT-LINE-OUT,
        GO TO CLOSE-FILES.
    MOVE DATE-IN TO DATE-OUT.
    MOVE TITLE-IN TO TITLE-OUT.
SORT-CM-FILE.
    OPEN OUTPUT SORTED-FILE.
    SORT FILE-SORT ON ASCENDING KEY
        S-MATCH-CODE
    INPUT PROCEDURE
        READ-SORT-INPUT THRU READ-SORT-INPUT-END
    OUTPUT PROCEDURE
        WRITE-SORT-OUTPUT THRU WRITE-SORT-OUTPUT-END.
CLOSE SORTED-FILE.
OPEN INPUT SORTED-FILE.
WRITE-HEADINGS.
    MOVE ZERO TO LINE-CT.
    ADD 1 TO PAGE-CT.
    MOVE PAGE-CT TO PAGE-OUT.
    PERFORM MOVE-SPACES.
    WRITE PRINT-LINE BEFORE NEW-PAGE.
    MOVE HEAD-1 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-2 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-3 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.

```

```
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
MOVE HEAD-4 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-5 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
ZERO-ACCUMS.  
MOVE ZEROS TO SUBTOTALS.  
HOUSE-KEEP.  
MOVE SPACES TO DATA-LINE.  
MOVE ZEROS TO ACCUMS.  
READ-SORTED-FILE.  
READ SORTED-FILE AT END PERFORM WRITE-LINE-OUT.  
GO TO WRITE-SUBTOTALS.  
NEW-MATCH-CODE.  
MOVE MATCH-CODE TO MATCH-HOLD.  
MOVE ACCT-NO TO FA-NOMINAL.  
READ FINANCIAL-ACCT-TABFI INVALID KEY  
MOVE SPACES TO FA-NAME (1), FA-NAME (2).  
TOTAL-COST-CNTRS.  
ADD AMTS (1) TO ACCUM (CC-POS2), SUBS (CC-POS2).  
PERFORM READ-SORTED-FILE.  
IF MATCH-CODE EQUAL TO MATCH-HOLD THEN  
GO TO TOTAL-COST-CNTRS.  
WRITE-LINE-OUT.  
MOVE FA-NAME (1) TO DESC.  
PERFORM MOVE-ACCUMS VARYING I FROM 1 BY 1 UNTIL I = 5.  
MOVE DATA-LINE TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
IF FA-NAME (2) NOT EQUAL TO SPACES THEN  
MOVE SPACES TO DATA-LINE,  
MOVE FA-NAME (2) TO DESC,  
MOVE DATA-LINE TO PRINT-DATA,  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
PERFORM HOUSE-KEEP.  
CHECK-FOR-NEW-HEADING.  
IF LINE-CT NOT < MAX-LINE THEN  
PERFORM WRITE-HEADINGS.  
GO TO NEW-MATCH-CODE.  
WRITE-SUBTOTALS.  
PERFORM MOVE-SPACES.  
PERFORM PRINT-LINE-OUT 2 TIMES.  
PERFORM HOUSE-KEEP.  
MOVE 'TOTALS' TO DESC.  
PERFORM MOVE-AMOUNTS VARYING I FROM 1 BY 1 UNTIL I = 5.  
MOVE DATA-LINE TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
GO TO CLOSE-FILES.  
MOVE-SPACES.
```

```
MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
    ADD 1 TO LINE-CT.  
MOVE-ACCUMS.  
    MOVE ACCUM (I) TO AMT OF AMOUNTS (I).  
MOVE-AMOUNTS.  
    MOVE SUBS (I) TO AMT OF AMOUNTS (I).  
READ-SORT-INPUT.  
    READ COST-MATRIX-FILE AT END  
        GO TO READ-SORT-INPUT-END.  
    IF MATCH-CODE-POS1 NOT = TO MATCH-POS1  
        THEN GO TO READ-SORT-INPUT.  
    RELEASE SORT-INPUT FROM CM-REC.  
    GO TO READ-SORT-INPUT.  
READ-SORT-INPUT-END.  
    EXIT.  
WRITE-SORT-OUTPUT.  
    RETURN FILE-SORT RECORD INTO INPUT-REC  
        AT END GO TO WRITE-SORT-OUTPUT-END.  
    WRITE INPUT-REC.  
    GO TO WRITE-SORT-OUTPUT.  
WRITE-SORT-OUTPUT-END.  
    EXIT.  
CLOSE-FILES.  
    CLOSE SORTED-FILE,  
        CARD-IN,  
        COST-MATRIX-FILE,  
        FINANCIAL-ACCT-TABFI,  
        PRINT-OUT.  
E-O-J.  
    STOP RUN.
```

```

PGMNAME=WUASPLV3
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASPLV3.
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FL 32789.
DATE-WRITTEN. NOV 16, 1977.
DATE-COMPILED. NOV 18, 1977.
REMARKS. THIS PROGRAM READS THE COST MATRIX FILE,
    SUMS ALL TOTALS AND PRODUCES A LEVEL III
    REPORT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS SEQUENTIAL
        RECORD KEY IS CM-NUMBER.
    SELECT PRINTER
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT READER
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CUST-MATRIX-FILE
    RECORD CONTAINS 80 CHARACTERS
    BLOCK CONTAINS 5 RECORDS
    LABEL RECORDS ARE STANDARD
    DATA RECORD IS INPUT-REC.
01 INPUT-REC.
    05 MATCH-CODE      PIC 9(4).
    05 CM-NUMBER.
        10 ACCT-NO      PIC X(10).
        10 COST-CNTR.
            15 CC-POS1    PIC 9.
            15 FILLER     PIC 999.
            10 CUST-CAT   PIC 99.
    05 AMTS          PIC S9(8)V99 OCCURS 6 TIMES.
FD READER
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE OMITTED

```

DATA RECORD IS CARD-IN.
 01 CARD-IN.
 05 DATE-IN PIC X(18).
 05 TITLE-IN PIC X(24).
 05 PERIOD-IN PIC X(7).
 05 FILLER PIC X(31).
 FD PRINTER
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINTER-DATA.
 01 PRINTER-DATA.
 02 FILLER PIC X.
 02 PRINT-LINE PIC X(132).
 WORKING-STORAGE SECTION.
 77 PER PIC 9 COMP VALUE 0.
 77 CAT PIC 99 COMP VALUE 0.
 77 FUN PIC 9 COMP VALUE 0.
 77 I PIC 99 COMP VALUE 0.
 77 J PIC 99 COMP VALUE 0.
 77 K PIC 9 COMP VALUE 0.
 77 LINE-TOTAL-1 PIC X(10).
 77 LINE-TOTAL REDEFINES LINE-TOTAL-1 PIC S9(8)V99.
 01 ACCUM-MAT.
 05 PERIOD-1 OCCURS 2 TIMES.
 10 CATEGORY-1 OCCURS 12 TIMES.
 15 FUNCTION-1 OCCURS 6 TIMES.
 20 AMOUNT-1 PIC X(10).
 01 ACCUM-MATRIX REDEFINES ACCUM-MAT.
 05 PERIOD OCCURS 2 TIMES.
 10 CATEGORY OCCURS 12 TIMES.
 15 FUNCTION OCCURS 6 TIMES.
 20 AMOUNT PIC S9(8)V99.
 01 SUBS-1.
 05 SUB-TOTALS-1 OCCURS 2 TIMES.
 10 SUB-TOTAL-1 OCCURS 8 TIMES PIC X(10).
 01 SUBS REDEFINES SUBS-1.
 05 SUB-TOTALS OCCURS 2 TIMES.
 10 SUB-TOTAL OCCURS 8 TIMES PIC S9(8)V99.
 01 GRAND-1.
 05 GRAND-TOTALS-1 OCCURS 2 TIMES.
 10 GRAND-TOTAL-1 OCCURS 8 TIMES PIC X(10).
 01 GRAND REDEFINES GRAND-1.
 05 GRAND-TOTALS OCCURS 2 TIMES.
 10 GRAND-TOTAL OCCURS 8 TIMES PIC S9(8)V99.
 01 CATEGORY-TITLES.
 05 TITLES1.
 10 TL1 PIC X(108) VALUE ' COST CATEGORY
 ' PAYROLL COST:
 ' MANAGEMENT AND
 ' SUPERVISION

```

-           ' OPERATION
-           ' LABOR      '.
10 TL2     PIC X(108) VALUE      ' MAINTENANCE
-           ' OTHER
-           ' TOTAL
-           ' BENEFITS
-           ' TOTAL PAYROLL
-           ' COST      '.

10 TL3     PIC X(108) VALUE      ' CHEMICALS
-           ' POWER
-           ' MISCELLANEOUS
-           ' TOTAL PAYROLL AND
-           ' DEPRECIATION      '.

10 TL4     PIC X(72)  VALUE      ' TOTAL
-           ' INTEREST
-           ' TAXES
-           ' OTHER DIRECT COSTS'.

05 TITLES REDEFINES TITLES1 OCCURS 22 TIMES PIC X(18).

01 HEADING-1.
    02 PERIOD-OUT      PIC X(7).
    02 FILLER          PIC X(45) VALUE SPACES.
    02 FILLER          PIC X(26) VALUE
                      'LEVEL III SUMMARY OF COST'.
    02 FILLER          PIC X(54) VALUE SPACES.

01 HEADING-2.
    02 FILLER          PIC X(9) VALUE 'ENDING: '.
    02 DATE-OUT         PIC X(20) VALUE SPACES.
    02 FILLER          PIC X(34) VALUE SPACES.
    02 FILLER          PIC X(3) VALUE 'FOR'.
    02 FILLER          PIC X(54) VALUE SPACES.
    02 PAGE-OUT        PIC X(12) VALUE 'PAGE 1 OF 2 '.

01 HEADING-3.
    02 FILLER          PIC X(52) VALUE SPACES.
    02 LOCATION-OUT    PIC X(25) VALUE SPACES.
    02 FILLER          PIC X(55) VALUE SPACES.

01 HEADING-4.
    02 FILLER          PIC X(18) VALUE SPACES.
    02 FILLER          PIC X(11) VALUE 'FUNCTION: '.
    02 FILLER          PIC XX   VALUE SPACES.
    02 FILLER          PIC X(11) VALUE 'ACQUISITION'.
    02 FILLER          PIC X(3)  VALUE SPACES.
    02 FILLER          PIC X(9)  VALUE 'TREATMENT'.
    02 FILLER          PIC X(5)  VALUE SPACES.
    02 FILLER          PIC X(8)  VALUE 'DELIVERY'.
    02 FILLER          PIC X(5)  VALUE SPACES.
    02 FILLER          PIC X(7)  VALUE 'SUPPORT'.
    02 FILLER          PIC X(6)  VALUE SPACES.
    02 FILLER          PIC X(8)  VALUE 'INTEREST'.
    02 FILLER          PIC X(8)  VALUE SPACES.

```

```

02 FILLER          PIC X(5) VALUE 'TOTAL'.
02 FILLER          PIC X(6) VALUE SPACES.
02 FILLER          PIC X(5) VALUE 'TAXES'.
02 FILLER          PIC X(8) VALUE SPACES.
02 FILLER          PIC X(5) VALUE 'TOTAL'.
02 FILLER          PIC XX    VALUE SPACES.

01 HEADING-5.
02 FILLER          PIC X(72) VALUE SPACES.
02 FILLER          PIC X(8) VALUE 'SERVICES'.
02 FILLER          PIC X(42) VALUE SPACES.
02 FILLER          PIC X(10) VALUE 'WITH TAXES'.

01 DATA-LINE.
05 TITLE-OUT        PIC X(18) VALUE SPACES.
05 FILLER          PIC X(3) VALUE SPACES.
05 PERIODS         PIC X(7) VALUE SPACES.
05 AMOUNTS-OUT      PIC --,---.--9.99 OCCURS 8 TIMES.

PROCEDURE DIVISION.
OPEN-FILES.
  OPEN INPUT COST-MATRIX-FILE, READER,
         OUTPUT PRINTER.
READ-CARD-IN.
  READ READER AT END MOVE '***** NO DATE CARD *****'
    TO PRINT-LINE
    PERFORM PRINT-LINE-OUT
    GO TO CLOSE-FILES.
MOVE DATE-IN TO DATE-OUT.
MOVE TITLE-IN TO LOCATION-OUT.
MOVE PERIOD-IN TO PERIOD-OUT.

CLOSE-READER.
  CLOSE READER.

HOUSEKEEPING.
  MOVE SPACES TO ACCUM-MAT, LINE-TOTAL-1, GRAND-1, SUBS-1,
            DATA-LINE.

TOTAL-UP-MATRIX.
  READ COST-MATRIX-FILE AT END
    GO TO WRITE-HEADINGS.
  IF (COST-CAT < 1 OR > 12) OR (CC-POS1 < 1 OR > 6) THEN
    GO TO TOTAL-UP-MATRIX.
  MOVE 1 TO I, J.
  PERFORM ADD-AMOUNTS 2 TIMES.
  GO TO TOTAL-UP-MATRIX.

ADD-AMOUNTS.
  IF AMOUNT-1 (I, COST-CAT, CC-POS1) = SPACES THEN
    MOVE ZERO TO AMOUNT (I, COST-CAT, CC-POS1).
  ADD AMTS (J) TO AMOUNT (I, COST-CAT, CC-POS1).
  ADD 1 TO I.
  ADD 2 TO J.

WRITE-HEADINGS.
  PERFORM MOVE-SPACES.
  WRITE PRINTER-DATA BEFORE NEW-PAGE.

```

```
MOVE HEADING-1 TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
MOVE HEADING-2 TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
MOVE HEADING-3 TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
MOVE HEADING-4 TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
MOVE HEADING-5 TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
MOVE TITLES (1) TO TITLE-OUT.  
MOVE DATA-LINE TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
WRITE-SUBTITLE.  
MOVE TITLES (2) TO TITLE-OUT.  
MOVE DATA-LINE TO PRINT-LINE.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
WRITE-MANAGEMENT-SUPERVISION.  
MOVE 3 TO I.  
MOVE 4 TO J.  
PERFORM WRITE-CATEGORIES.  
WRITE-OPERATION-LABOR.  
MOVE 5 TO I.  
MOVE 6 TO J.  
PERFORM WRITE-CATEGORIES.  
WRITE-MAINTENANCE-LABOR.  
MOVE 7 TO I.  
PERFORM WRITE-CATEGORIES.  
WRITE-OTHER-LABOR.  
MOVE 8 TO I.  
PERFORM WRITE-CATEGORIES.  
WRITE-SUBTOTAL-T1.  
MOVE 9 TO I.  
MOVE 13 TO J.  
PERFORM WRITE-SUBTOTALS.  
WRITE-BENIFITS.  
MOVE 10 TO I.  
PERFORM WRITE-CATEGORIES.  
WRITE-RUNNING-SUBTOTAL-T2.  
MOVE 11 TO I.  
MOVE 12 TO J.  
PERFORM WRITE-RUNNING-SUBTOTALS.  
WRITE-SUBTITLE-ODC.  
PERFORM PRINT-LINE-OUT.  
MOVE 'OTHER DIRECT COST:' TO TITLE-OUT.  
MOVE DATA-LINE TO PRINT-LINE.
```

```
        PERFORM PRINT-LINE-OUT.  
        PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
WRITE-MAINTENANCE.  
        MOVE 7 TO I.  
        MOVE 13 TO J.  
        PERFORM WRITE-CATEGORIES.  
WRITE-CHEMICALS.  
        MOVE 14 TO I.  
        PERFORM WRITE-CATEGORIES.  
WRITE-POWER.  
        MOVE 15 TO I.  
        PERFORM WRITE-CATEGORIES.  
WRITE-MISCELLANEOUS.  
        MOVE 16 TO I.  
        PERFORM WRITE-CATEGORIES.  
WRITE-SUBTOTAL-T3.  
        MOVE 9 TO I.  
        PERFORM WRITE-SUBTOTALS.  
WRITE-SECOND-PAGE.  
        MOVE *PAGE 2 OF 2 * TO PAGE-OUT.  
        PERFORM WRITE-HEADINGS.  
        PERFORM PRINT-LINE-OUT.  
WRITE-RUNNING-SUBTOTAL-T4.  
        MOVE 17 TO I.  
        MOVE 22 TO J.  
        PERFORM WRITE-RUNNING-SUBTOTALS.  
WRITE-DEPRICINATION.  
        MOVE 18 TO I.  
        MOVE 13 TO J.  
        PERFORM WRITE-CATEGORIES.  
WRITE-RUNNING-SUBTOTAL-T5.  
        MOVE 19 TO I.  
        PERFORM WRITE-RUNNING-SUBTOTALS.  
WRITE-INTEREST.  
        MOVE 20 TO I.  
        PERFORM WRITE-CATEGORIES.  
WRITE-TAXES.  
        MOVE 21 TO I.  
        PERFORM WRITE-CATEGORIES.  
        PERFORM PRINT-LINE-OUT.  
WRITE-GRAND-TOTAL-T6.  
        MOVE 19 TO I.  
        PERFORM WRITE-RUNNING-SUBTOTALS.  
        GO TO CLOSE-FILES.  
MOVE-SPACES.  
        MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
        WRITE PRINTER-DATA BEFORE 1.  
WRITE-CATEGORIES.  
        MOVE 1 TO PER.
```

```

ADD 1 TO CAT.
PERFORM MOVE-AMOUNTS VARYING FUN FROM 1 BY 1 UNTIL FUN = 6.
PERFORM TOTAL-AMOUNTS.
MOVE TITLES (I) TO TITLE-OUT.
MOVE PERIOD-IN TO PERIODS.
MOVE DATA-LINE TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE SPACES TO DATA-LINE.
MOVE 2 TO PER.
PERFORM MOVE-AMOUNTS VARYING FUN FROM 1 BY 1 UNTIL FUN = 6.
PERFORM TOTAL-AMOUNTS.
MOVE TITLES (J) TO TITLE-OUT.
MOVE 'YTD' TO PERIODS.
MOVE DATA-LINE TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
MOVE SPACES TO DATA-LINE.

MOVE-AMOUNTS.
  IF AMOUNT-1 (PER, CAT, FUN) NOT = SPACES THEN
    PERFORM CHECK-LINE THRU ADD-TOTALS.

CHECK-LINE.
  IF LINE-TOTAL-1 = SPACES THEN
    MOVE ZERO TO LINE-TOTAL.

CHECK-SUB.
  IF SUB-TOTAL-1 (PER, FUN) = SPACES THEN
    MOVE ZERO TO SUB-TOTAL (PER, FUN).

CHECK-GRAND.
  IF GRAND-TOTAL-1 (PER, FUN) = SPACES THEN
    MOVE ZERO TO GRAND-TOTAL (PER, FUN).

ADD-TOTALS.
  MOVE AMOUNT (PER, CAT, FUN) TO AMOUNTS-OUT (FUN).
  ADD AMOUNT (PER, CAT, FUN) TO LINE-TOTAL,
    SUB-TOTAL (PER, FUN), GRAND-TOTAL (PER, FUN).

TOTAL-AMOUNTS.
  MOVE 6 TO FUN.
  PERFORM TOTAL-AMTS-FIX.
  IF AMOUNT-1 (PER, CAT, FUN) NOT = SPACES THEN
    MOVE 7 TO FUN
    PERFORM CHECK-LINE THRU CHECK-GRAND
    MOVE AMOUNT (PER, CAT, 6) TO AMOUNTS-OUT (7)
    ADD AMOUNT (PER, CAT, 6) TO LINE-TOTAL, SUB-TOTAL (PER, 7),
      GRAND-TOTAL (PER, 7).

  MOVE 8 TO FUN.
  PERFORM TOTAL-AMTS-FIX.
  MOVE SPACES TO LINE-TOTAL-1.

TOTAL-AMTS-FIX.
  IF LINE-TOTAL-1 NOT = SPACES THEN
    PERFORM CHECK-SUB THRU CHECK-GRAND
    MOVE LINE-TOTAL TO AMOUNTS-OUT (FUN)
    ADD LINE-TOTAL TO SUB-TOTAL (PER, FUN),

```

```
GRAND-TOTAL (PER, FUN).
WRITE-SUBTOTALS.
MOVE 1 TO K.
PERFORM MOVE-SUBTOT-TO-DATALINE VARYING PER
   FROM 1 BY 1 UNTIL PER = 9.
MOVE TITLES (I) TO TITLE-OUT.
MOVE PERIOD-IN TO PERIODS.
MOVE DATA-LINE TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE 2 TO K.
PERFORM MOVE-SUBTOT-TO-DATALINE VARYING PER
   FROM 1 BY 1 UNTIL PER = 9.
MOVE TITLES (J) TO TITLE-OUT.
MOVE 'YTD' TO PERIODS.
MOVE DATA-LINE TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE SPACES TO DATA-LINE, SUBS-1.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
PERFORM PRINT-LINE-OUT.
WRITE-RUNNING-SUBTOTALS.
MOVE 1 TO K.
PERFORM MOVE-GRTOT-TO-DATALINE VARYING PER
   FROM 1 BY 1 UNTIL PER = 9.
MOVE TITLES (I) TO TITLE-OUT.
MOVE PERIOD-IN TO PERIODS.
MOVE DATA-LINE TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE 2 TO K.
PERFORM MOVE-GRTOT-TO-DATALINE VARYING PER
   FROM 1 BY 1 UNTIL PER = 9.
MOVE TITLES (J) TO TITLE-OUT.
MOVE 'YTD' TO PERIODS.
MOVE DATA-LINE TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE SPACES TO DATA-LINE, SUBS-1.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
PERFORM PRINT-LINE-OUT.
MOVE-SUBTOT-TO-DATALINE.
IF SUB-TOTAL-1 (K, PER) NOT = SPACES THEN
  MOVE SUB-TOTAL (K, PER) TO AMOUNTS-OUT (PER).
MOVE-GRTOT-TO-DATALINE.
IF GRAND-TOTAL-1 (K, PER) NOT = SPACES THEN
  MOVE GRAND-TOTAL (K, PER) TO AMOUNTS-OUT (PER).
CLOSE-FILES.
CLOSE COST-MATRIX-FILE, PRINTER.
E-O-J.
STOP RUN.
```

```
PGMNAME=WUASPLV2
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASPLV2
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FL 32789.
DATE-WRITTEN. JAN 1978.
DATE-COMPILED.
REMARKS. THIS PROGRAM READS THE COST MATRIX FILE
        AND PRINTS THE LEVEL TWO WATER UTILITY
        REPORT CALLED 'COST COMPARISON'.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS SEQUENTIAL
        RECORD KEY IS CM-NUMBER.
    SELECT FINANCIAL-ACCT-TABFI
        ASSIGN TO DA-3330-I-FATABFI
        ACCESS MODE IS RANDOM
        NOMINAL KEY IS FA-NOMINAL
        RECORD KEY IS FA-NUMBER.
    SELECT FILE-SORT
        ASSIGN TO UT-3330-S-FILESORT
        ACCESS MODE IS SEQUENTIAL.
    SELECT SORTED-FILE
        ASSIGN TO UT-3330-S-SORTEDFI
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE OMITTED
    DATA RECORD IS DATE-CARD.
01 DATE-CARD.
    05 DATE-IN.
```

```

        10  MO-IN    PIC X(3).
        10  DAY-YR   PIC X(15).
05  TITLE-IN   PIC X(24).
05  FILLER      PIC X(8).
05  FISCAL-YR  PIC X(24).
05  FILLER      PIC X(4).
05  NO-WKS     PIC X.
05  RPT-PERIOD PIC X.

FD  COST-MATRIX-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CM-REC.

01  CM-REC.
05  MATCH-NU   PIC X(4).
05  CM-NUMBER.
10  FA-NU      PIC X(10).
10  CC-NU.
15  CC1        PIC X.
15  FILLER     PIC X(3).
10  SA-NU      PIC XX.
05  AMT-S      PIC S9(8)V99 OCCURS 6 TIMES.

*****  

* THE ABOVE AMOUNTS ARE AS FOLLOWS
* (1) -- CURRENT COST AMOUNT
* (2) -- PREVIOUS MONTH COST AMOUNT
* (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)
* (4) -- BUDGET AMOUNT
* (5) -- BUDGET AMOUNT YEAR TO DATE (YTD)
* (6) -- PREVIOUS YEAR COST AMOUNT (PYC)
*****  

FD  FINANCIAL-ACCT-TABFI
BLOCK CONTAINS 8 RECORDS
RECORD CONTAINS 58 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS FA-REC.

01  FA-REC.
05  FA-NUMBER   PIC X(10).
05  FA-NAME-1   PIC X(40).
05  FILLER      PIC X(8).

SD  FILE-SORT
RECORD CONTAINS 80 CHARACTERS
DATA RECORD IS SORT-INPUT.

01  SORT-INPUT.
05  FILLER      PIC X(4).
05  FIRST-10    PIC X(10).
05  CC-NDS.
10  CC-1        PIC X.
10  CC-3        PIC X(3).
05  SA-NDS     PIC XX.

```

```

05 FILLER      PIC X(30).
05 BUDGET      PIC 9(8)V99.
05 FILLER      PIC X(20).

FD SORTED-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-REC.

01 INPUT-REC.
05 MATCH-NO      PIC X(4).
05 FA-NO.
10 FA-NO-1      PIC X(2).
10 FA-NO-2      PIC X(8).
05 CC-NO.
10 CC-NO-1      PIC X.
10 CC-NO-2      PIC XXX.
05 SA-NO        PIC XX.
05 AMTS.
10 AMT-1        PIC S9(8)V99.
10 AMT-2        PIC S9(8)V99.
10 AMT-3        PIC S9(8)V99.
10 AMT-4        PIC S9(8)V99.
10 AMT-5        PIC S9(8)V99.
10 AMT-6        PIC S9(8)V99.

FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-DATA.

01 PRINT-DATA.
05 FILLER      PIC X.
05 PRINT-LINE    PIC X(132).

WORKING-STORAGE SECTION.
77 SCT-ONE      PIC 9(5) COMP VALUE 0.
77 SCT-TWO      PIC 9(5) COMP VALUE 0.
77 LINE-CT       PIC 9(4) COMP VALUE 0.
77 PAGE-CT       PIC 9(4) COMP VALUE 1.
77 MAX-LINE      PIC 99   COMP VALUE 56.
77 SUB-T1        PIC X(6) VALUE SPACES.
77 SUB-T2        PIC X(6) VALUE 'CONT-D'.
77 FA-NOMINAL    PIC X(10) VALUE SPACES.
77 BUD-ACCT      PIC X(10) VALUE SPACES.
77 NEGS-IN       PIC X(2) VALUE '***'.
77 YEAR-IN       PIC X(3) JUSTIFIED RIGHT.
77 YEAR-OUT      PIC 9(4) COMP VALUE 0.
77 MONTH         PIC 9(2) COMP VALUE 1.
77 I              PIC 99   COMP VALUE 0.
77 J              PIC 9    COMP VALUE 0.
77 K              PIC 99   COMP VALUE 0.
77 CC-J          PIC 9    COMP VALUE 0.
77 SA-K          PIC 99   COMP VALUE 0.

```

```

77 P-ERRUR      PIC S9(8)V99 COMP VALUE +0.
77 DEPRE-C      PIC S9(8)V99 CUMP VALUE +0.
77 DEPRE-P      PIC S9(8)V99 CUMP VALUE +0.
77 INT-C       PIC S9(8)V99 COMP VALUE +0.
77 INT-P       PIC S9(8)V99 COMP VALUE +0.
77 TAX-C        PIC S9(8)V99 COMP VALUE +0.
77 TAX-P        PIC S9(8)V99 CUMP VALUE +0.
77 TEMP-1       PIC S9(8)V999 COMP VALUE +0.
77 TEMP-2       PIC S9(8)V999 COMP VALUE +0.
77 TEMP-3       PIC S9(8)V999 COMP VALUE +0.
77 S-TOT-1      PIC S9(8)V99 COMP VALUE +0.
77 S-TOT-2      PIC S9(8)V99 COMP VALUE +0.
77 S-TOT-3      PIC S9(8)V99 COMP VALUE +0.
77 S-TOT-4      PIC S9(8)V99 COMP VALUE +0.
77 S-TOT-5      PIC S9(8)V99 COMP VALUE +0.
77 S-TOT-6      PIC S9(8)V99 CUMP VALUE +0.
77 TUT-1        PIC S9(8)V99 CUMP VALUE +0.
77 TOT-2        PIC S9(8)V99 COMP VALUE +0.
77 TOT-3        PIC S9(8)V99 COMP VALUE +0.
77 TOT-4        PIC S9(8)V99 COMP VALUE +0.
77 TOT-5        PIC S9(8)V99 COMP VALUE +0.
77 TOT-6        PIC S9(8)V99 COMP VALUE +0.
77 TOT-T3       PIC S9(8)V99 COMP VALUE +0.
77 TOT-T4       PIC S9(8)V99 COMP VALUE +0.
77 TOT-T5       PIC S9(8)V99 COMP VALUE +0.
77 PRINT-REQ-FLAG PIC X    VALUE '0'.
88 FLAG-0        VALUE '0'.
88 FLAG-1        VALUE '1'.
88 FLAG-2        VALUE '2'.
77 RPT-FLAG      PIC X    VALUE '1'.
88 RPT-A         VALUE '1'.
88 RPT-B         VALUE '2'.
77 INT-FLAG      PIC X    VALUE '0'.
88 FLAG-NO        VALUE '0'.
88 FLAG-YES       VALUE '1'.
01 YEAR-PTS.
05 YR-A          PIC X(2).
05 YR-B          PIC X.
01 MONTH-NAMES.
05 MO-A          PIC X(36)   VALUE      'JANFEBMARAPRMAYJUN
-                                     'JULAUGSEPOCTNOWDEC'.
05 MO-B REDEFINES MO-A.
10 MO-NAME OCCURS 12 TIMES PIC X(3).
01 NAME-LIST.
05 LIST-A.
10 NAMES-1A      PIC X(120)  VALUE
-                                     'SOURCE OF SUPPLY
-                                     'PUMPING
-                                     'WATER TREATMENT
-                                     'TRANSMISSION AND DISTRIBUTION '.

```

```

10 NAMES-2A    PIC X(120) VALUE
      'CUSTOMER ACCOUNTS
      'ADMINISTRATIVE AND GENERAL
      'NON BUDGETED COSTS
      'TOTAL ACTUAL COSTS TO BUDGET '.

05 NAME-A REDEFINES LIST-A.
  10 NAME OCCURS 8 TIMES PIC X(30).

01 LOWER-LIMITS.
  05 LIM-LA    PIC X(16) VALUE '6062646690920000'.
  05 LIM-LB REDEFINES LIM-LA.
    10 LO-LIM OCCURS 8 TIMES PIC X(2).

01 UPPER-LIMITS.
  05 LIM-UA    PIC X(16) VALUE '6163656790939999'.
  05 LIM-UB REDEFINES LIM-UA.
    10 HI-LIM OCCURS 8 TIMES PIC X(2).

01 COST-CENTERS.
  05 CC-LIST-1    PIC X(120) VALUE
      'ACQUISITION
      'TREATMENT
      'DELIVERY
      'SUPPORT SERVICES
      'INTEREST
      'TAXES '
  05 CC-LIST-2 REDEFINES CC-LIST-1.
    10 CC-NAME OCCURS 6 TIMES PIC X(20).

01 CAT-LIST.
  05 CAT-A.
    10 C-CAT-1    PIC X(100) VALUE
      'MANAGEMENT & SUPRV
      'OPERATION LABOR
      'MAINTENANCE LABOR
      'OTHER LABOR
      'BENEFITS '
    10 C-CAT-2    PIC X(100) VALUE
      'MAINTENANCE
      'CHEMICALS
      'POWER
      'MISCELLANEOUS
      'DEPRECIATION '

  05 CAT-B REDEFINES CAT-A.
    10 CAT-NAME OCCURS 10 TIMES PIC X(20).

01 HEADING-1.
  05 FILLER      PIC X(55) VALUE SPACES.
  05 FILLER      PIC X(22) VALUE 'COST COMPARISON REPORT'.
  05 FILLER      PIC X(55) VALUE SPACES.

01 HEADING-2.
  05 FILLER      PIC X(12) VALUE 'LAST ENTRY: '.
  05 DATE-DUT    PIC X(18).
  05 FILLER      PIC X(34) VALUE SPACES.
  05 FILLER      PIC X(3)  VALUE 'FOR'.

```

```

05 FILLER      PIC X(55) VALUE SPACES.
05 FILLER      PIC X(6)  VALUE 'PAGE: '.
05 PAGE-DUT    PIC Z(4).

01 HEADING-3.
05 FILLER      PIC X(12)  VALUE 'RPT PERIOD: '.
05 RPT-PD      PIC X(8).
05 WKS-DUT     PIC X(8).
05 FILLER      PIC X(26)  VALUE SPACES.
05 TITLE-DUT   PIC X(24).
05 FILLER      PIC X(54)  VALUE SPACES.

01 HEADING-12.
05 FILLER      PIC X(12)  VALUE 'YTD START: '.
05 FISC-YR     PIC X(24).
05 FILLER      PIC X(24)  VALUE SPACES.
05 FILLER      PIC X(12)  VALUE '( LEVEL II )'.
05 FILLER      PIC X(60)  VALUE SPACES.

*****  

*  

*          PART-A HEADINGS  

*  

*****
```

```

01 HEADING-4.
05 FILLER      PIC X(36) VALUE 'COMPARISON OF ACTUAL COS
-                  T TO BUDGET '.

05 SUB-DUT     PIC X(6).
05 FILLER      PIC X(90) VALUE SPACES.

01 HEADING-5.
05 FILLER      PIC X(17) VALUE 'BUDGET CATEGORIES'.
05 FILLER      PIC X(42) VALUE SPACES.
05 FILLER      PIC X(12) VALUE 'ACTUAL COSTS'.
05 FILLER      PIC X(15) VALUE SPACES.
05 FILLER      PIC X(14) VALUE 'BUDGETED COSTS'.
05 FILLER      PIC X(10) VALUE SPACES.
05 FILLER      PIC X(22) VALUE 'VARIANCE YR TO DATE '.

01 HEADING-6.
05 FILLER      PIC X(14) VALUE '***** **'.
05 FILLER      PIC X(20) VALUE '*****'.
05 FILLER      PIC X(19) VALUE SPACES.
05 FILLER      PIC X(24) VALUE '*****'.
-                  '**'.

05 FILLER      PIC X(4)  VALUE SPACES.
05 FILLER      PIC X(24) VALUE '*****'.
-                  '**'.

05 FILLER      PIC X(4)  VALUE SPACES.
05 FILLER      PIC X(21) VALUE '*****'.
05 FILLER      PIC X(2)  VALUE SPACES.

01 HEADING-7.
05 FILLER      PIC X(2)  VALUE SPACES.
05 FILLER      PIC X(18) VALUE 'ACCT #    ACCOUNT '.
05 FILLER      PIC X(4)  VALUE 'NAME'.
```

```

05 FILLER      PIC X(29) VALUE SPACES.
05 FILLER      PIC X(24) VALUE 'CURR PERIOD    YR TO DATE'.
05 FILLER      PIC X(5)  VALUE SPACES.
05 FILLER      PIC X(13) VALUE 'YR TO DATE    '.
05 FILLER      PIC X(15) VALUE 'ANNUAL AMT    '.
05 FILLER      PIC X(22) VALUE '    BALANCE PERCENT    '.

01 BUDGET-CAT.
05 NAME-OUT     PIC X(30).
05 FILLER      PIC X(102) VALUE SPACES.

01 DATA-LINE-A.
05 HEAD-STRIP-OUT PIC X(52).
05 COL-1A       PIC ----,----.
05 COL-2A       PIC --,--,-,--.
05 FILLER      PIC X(1)  VALUE SPACES.
05 COL-3A       PIC ---,---,---.
05 COL-4A       PIC ---,---,---.
05 FILLER      PIC X(1)  VALUE SPACES.
05 COL-5A       PIC ---,---,---.
05 FILLER      PIC X(2)  VALUE SPACES.
05 COL-6A       PIC -(5)..
05 FILLER      PIC X(1)  VALUE SPACES.
05 NEG-OUT      PIC X(2).

01 HEAD-STRIP-1A.
05 FILLER      PIC X(2)  VALUE SPACES.
05 ACCT-NO     PIC X(9).
05 FILLER      PIC X(1)  VALUE SPACES.
05 ACCT-NAME   PIC X(40).

01 HEAD-STRIP-2A.
05 NAME-8      PIC X(30).
05 FILLER      PIC X(22) VALUE SPACES.

*****
*
*
*          PART-B HEADINGS
*
*****
01 HEADING-8.
05 FILLER      PIC X(40) VALUE 'COMPARISON OF ACTUAL COSTS
-                                TO PRIOR YEAR '.

05 SUBS-OUT    PIC X(6).
05 FILLER      PIC X(86) VALUE SPACES.

01 HEADING-9.
05 FILLER      PIC X(6)  VALUE SPACES.
05 FILLER      PIC X(15) VALUE 'COST CATEGORIES'.
05 FILLER      PIC X(28) VALUE SPACES.
05 FILLER      PIC X(13) VALUE 'ACTUAL COSTS '.
05 Curr-YR     PIC 9(4).
05 FILLER      PIC X(10) VALUE SPACES.
05 FILLER      PIC X(13) VALUE 'ACTUAL COSTS '.
05 PREV-YR     PIC 9(4).
05 FILLER      PIC X(14) VALUE SPACES.

```

```

05 FILLER          PIC X(19) VALUE 'VARIANCE YR TO DATE'.
05 FILLER          PIC X(6)  VALUE SPACES.
01 HEADING-10.
05 FILLER          PIC X(6)  VALUE SPACES.
05 FILLER          PIC X(30) VALUE '*****'.
               '*****'.
05 FILLER          PIC X(13) VALUE SPACES.
05 FILLER          PIC X(17) VALUE '*****'.
05 FILLER          PIC X(10) VALUE SPACES.
05 FILLER          PIC X(17) VALUE '*****'.
05 FILLER          PIC X(11) VALUE SPACES.
05 FILLER          PIC X(24) VALUE '*****'.
               '*****'.
05 FILLER          PIC X(4)  VALUE SPACES.
01 HEADING-11.
05 FILLER          PIC X(53) VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'YR TO DATE'.
05 FILLER          PIC X(17) VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'YR TO DATE'.
05 FILLER          PIC X(17) VALUE SPACES.
05 FILLER          PIC X(19) VALUE ' BALANCE PERCENT'.
05 FILLER          PIC X(6)  VALUE SPACES.
01 DATA-LINE-B.
05 HEAD-OUT         PIC X(49).
05 COL-1B           PIC ---,---,---.
05 FILLER          PIC X(13) VALUE SPACES.
05 COL-2B           PIC ---,---,---.
05 FILLER          PIC X(11) VALUE SPACES.
05 COL-5B           PIC ---,---,---.
05 FILLER          PIC X(4)  VALUE SPACES.
05 COL-6B           PIC -(5)-.
05 FILLER          PIC X(4)  VALUE SPACES.
05 NEGS-OUT         PIC X(2).
01 HEAD-STRIP-1B.
05 FILLER          PIC X(10) VALUE SPACES.
05 C-CAT-OUT        PIC X(30).
05 FILLER          PIC X(9)  VALUE SPACES.
01 HEAD-STRIP-2B.
05 FILLER          PIC X(6)  VALUE SPACES.
05 FILLER          PIC X(14)  VALUE 'TOTAL COST OF '.
05 CC-DUTS          PIC X(20).
05 FILLER          PIC X(9)  VALUE SPACES.
01 DATA-LINE-D.
05 FILLER          PIC X(6)  VALUE SPACES.
05 CC-UUT           PIC X(20).
05 FILLER          PIC X(106) VALUE SPACES.
01 DATA-LINE-E.
05 FILLER          PIC X(8)  VALUE SPACES.
05 C-CAT-SUB        PIC X(20).
05 FILLER          PIC X(104) VALUE SPACES.

```

```

*****  

*  

*      START OF PROGRAM  

*  

*****  

PROCEDURE DIVISION.  

OPEN-FILES.  

    OPEN INPUT CARD-IN, COST-MATRIX-FILE,  

          FINANCIAL-ACCT-TABFI.  

        OUTPUT PRINT-OUT.  

READ-TITLE-CARD.  

    READ CARD-IN AT END GO TO TITLE-PARAGRAPH.  

    GO TO READ-TITLE-CARD.  

TITLE-PARAGRAPH.  

    PERFORM FILL-HEADINGS.  

    PERFORM TOP-NEW-PAGE.  

    PERFORM WRITE-PAGE-HEADING.  

SORT-RECORDS.  

    OPEN OUTPUT SORTED-FILE.  

    SORT FILE-SORT ON ASCENDING KEY FIRST-10  

          DESCENDING KEY BUDGET  

    INPUT PROCEDURE READ-SORT-INPUT  

          THRU READ-SORT-INPUT-END  

    OUTPUT PROCEDURE WRITE-SORT-OUTPUT  

          THRU WRITE-SORT-OUTPUT-END.  

REWIND-FILE.  

    CLOSE SORTED-FILE.  

    OPEN INPUT SORTED-FILE.  

REPORT-PART-A.  

    PERFORM WRITE-SUB-TITLE-A.  

    MOVE '0' TO PRINT-REQ-FLAG.  

    MOVE SPACES TO BUD-ACCT.  

    PERFORM NEW-BUDGET-CAT THRU TEST-ACCT-LIMIT-END VARYING I FRO  

- M 1 BY 1 UNTIL I GREATER THAN 7.  

    PERFORM WRITE-TOTALS.  

    PERFORM WRITE-DEPREC-INT-TAXES.  

    MOVE TOT-2 TO TOT-T4  

    PERFORM WRITE-TOTALS.  

    MOVE ZEROS TO DEPRE-C, DEPRE-P, INT-C, INT-P, TAX-C, TAX-P.  

    MOVE ZEROS TO SCT-DONE, TEMP-1, TEMP-2.  

    MOVE '0' TO INT-FLAG.  

    MOVE '2' TO RPT-FLAG.  

    MOVE SUB-T1 TO SUB-OUT, SUBS-OUT.  

DOUBLE-SORT.  

    CLOSE SORTED-FILE, COST-MATRIX-FILE.  

    OPEN INPUT COST-MATRIX-FILE.  

    OPEN OUTPUT SORTED-FILE.  

    SORT FILE-SORT ON ASCENDING KEY  

          CC-1  

          SA-NOS

```

```
INPUT PROCEDURE READ-SORT-INPUT
    THRU READ-SORT-INPUT-END
OUTPUT PROCEDURE WRITE-SORT-OUTPUT
    THRU WRITE-SORT-OUTPUT-END.

REPORT-PART-B.
    PERFORM TOP-NEW-PAGE.
    PERFORM WRITE-PAGE-HEADING.
    PERFORM WRITE-SUB-TITLE-B.
    PERFORM ZERO-SUB-TOTALS.
    PERFORM ZERO-TOTALS.
    PERFORM REWIND-FILE.
    PERFORM WRITE-COST-CENTER THRU COST-CENTER-END VARYING J FROM
        1 BY 1 UNTIL J GREATER THAN 4.
    PERFORM WRITE-OP-COSTS.
    PERFORM WRITE-INTEREST.
    PERFORM WRITE-TAXES.
    PERFORM WRITE-LAST-LINE.
    PERFORM TOP-NEW-PAGE.
    PERFORM WRITE-COUNTERS.
    GO TO CLOSE-FILES.
```

*

* END OF PROGRAM.

*

*

* SUBROUTINES FOLLOW,
* 1. GENERAL SUBROUTINES
* 2. ROUTINES FOR PART-A
* 3. ROUTINES FOR PART-B
*

*

*

* GENERAL SUBROUTINES
*

MOVE-SPACES.

MOVE SPACES TO PRINT-LINE.

PRINT-LINE-OUT.

WRITE PRINT-DATA BEFORE 1.

ADD 1 TO LINE-CT.

SKIP-LINE.

PERFORM MOVE-SPACES.

PERFORM PRINT-LINE-OUT.

PAGE-OVFLW.

IF LINE-CT NOT < MAX-LINE THEN

PERFORM TOP-NEW-PAGE

MOVE SUB-T2 TO SUB-OUT, SUBS-OUT

```

        PERFORM WRITE-PAGE-HEADING
        IF RPT-A THEN PERFORM WRITE-SUB-TITLE-A
        ELSE PERFORM WRITE-SUB-TITLE-B.

ZERO-SUB-TOTALS.
    MOVE ZEROS TO S-TOT-1, S-TOT-2, S-TOT-3, S-TOT-4, S-TOT-5,
    S-TOT-6.

ZERO-TOTALS.
    MOVE ZEROS TO TOT-1, TOT-2, TOT-3, TOT-4, TOT-5, TOT-6.

TOP-NEW-PAGE.
    PERFORM MOVE-SPACES.
    WRITE PRINT-DATA BEFORE NEW-PAGE.
    MOVE PAGE-CT TO PAGE-OUT.
    MOVE 0 TO LINE-CT.
    ADD 1 TO PAGE-CT.

FILL-HEADINGS.
    MOVE DATE-IN TO DATE-OUT, YEAR-IN.
    MOVE FISCAL-YR TO FISC-YR.
    MOVE 'MONTHLY' TO RPT-PD.
    IF RPT-PERIOD = '2' THEN MOVE 'QUARTER' TO RPT-PD.
    IF RPT-PERIOD = '3' THEN MOVE 'HALF YR' TO RPT-PD.
    IF RPT-PERIOD = '4' THEN MOVE 'ANNUAL' TO RPT-PD.
    MOVE SPACES TO WKS-OUT.
    IF NO-WKS = '1' MOVE '( 4 WKS)' TO WKS-OUT.
    IF NO-WKS = '2' MOVE '( 5 WKS)' TO WKS-OUT.
    IF NO-WKS = '3' MOVE '(13 WKS)' TO WKS-OUT.
    IF NO-WKS = '4' MOVE '(26 WKS)' TO WKS-OUT.
    IF NO-WKS = '5' MOVE '(52 WKS)' TO WKS-OUT.
    MOVE PAGE-CT TO PAGE-OUT.
    MOVE TITLE-IN TO TITLE-OUT.
    MOVE SUB-T1 TO SUB-OUT, SUBS-OUT.
    PERFORM FIND-YEAR.
    PERFORM FIND-MONTH VARYING I FROM 1 BY 1 UNTIL I GREATER
    THAN 12.

WRITE-PAGE-HEADING.
    MOVE HEADING-1 TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    MOVE HEADING-2 TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    MOVE HEADING-3 TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    MOVE HEADING-12 TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
    PERFORM SKIP-LINE 2 TIMES.

FIND-YEAR.
    IF YEAR-IN IS NOT NUMERIC MOVE '078' TO YEAR-IN.
    MOVE YEAR-IN TO YEAR-OUT.
    IF YEAR-OUT > 900 THEN
        SUBTRACT 900 FROM YEAR-OUT.
    ADD 1900 YEAR-OUT GIVING CURR-YR.
    ADD 1899 YEAR-OUT GIVING PREV-YR.

```

```

FIND-MONTH.
  IF MO-IN EQUAL MO-NAME (I) MOVE I TO MONTH.
  WRITE-DATA-LINE-A.
    MOVE DATA-LINE-A TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
  WRITE-DATA-LINE-B.
    MOVE DATA-LINE-B TO PRINT-LINE.
    PERFORM PRINT-LINE-OUT.
VARI-AND-PERCENT.
  MOVE SPACES TO NEG-OUT, NEGS-OUT.
  IF RPT-B AND TEMP-2 = 0 THEN MOVE 0 TO TEMP-1.
  SUBTRACT TEMP-1 FROM TEMP-2 GIVING TEMP-3.
  MOVE TEMP-3 TO COL-5A, COL-5B.
  MULTIPLY 100 BY TEMP-1.
  IF TEMP-2 > 0 DIVIDE TEMP-2 INTO TEMP-1 FALSE
                                             MOVE ZERO TO TEMP-1.
  IF TEMP-1 > 100      MOVE NEGS-IN TO NEG-OUT, NEGS-OUT.
  ADD .C45 TO TEMP-1.
  MOVE TEMP-1 TO COL-6A, COL-6B.

*****  

*  

*          SUBROUTINES FOR PART-A  

*  

*****  

WRITE-SUB-TITLE-A.
  MOVE HEADING-4 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM SKIP-LINE.
  MOVE HEADING-5 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-6 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-7 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM SKIP-LINE.

NEW-BUDGET-CAT.
  IF I EQUAL 7 THEN
    MOVE '0' TO PRINT-REQ-FLAG
    MOVE SPACES TO BUD-ACCT
    PERFORM REWIND-FILE.

  PERFORM PAGE-OVFLOW.
  MOVE NAME (I) TO NAME-OUT.
  PERFORM SKIP-LINE.
  MOVE BUDGET-CAT TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.

READ-SORTED-FILE.
  READ SORTED-FILE AT END GO TO READ-FILE-COMPLETE.
  IF AMT-4 > 0 THEN MOVE FA-NO TO BUD-ACCT.
  IF I = 7 THEN GO TO STEP-TWO.
  IF FA-NO-1 < LO-LIM (I) GO TO READ-SORTED-FILE.

```

```

IF FA-NO EQUAL BUD-ACCT THEN GO TO NEXT-STEP.
IF AMT-4 = 0 THEN GO TO TEST-ACCT-LIMIT.
GO TO NEXT-STEP.
STEP-TWO.
MOVE '0' TO INT-FLAG.
PERFORM DEPREC-INT-TAXES THRU TAXES-END.
IF FLAG-YES THEN GO TO TEST-ACCT-LIMIT.
IF FA-NO EQUAL BUD-ACCT THEN GO TO READ-SORTED-FILE.
IF AMT-4 > 0    GO TO READ-SORTED-FILE.
IF AMT-5 > 0    GO TO READ-SORTED-FILE.
IF AMT-3 = 0    GO TO READ-SORTED-FILE.
MOVE ZERO  TO COL-3A, COL-4A, COL-5A, COL-6A.
MOVE SPACES TO NEG-OUT.
NEXT-STEP.
IF FLAG-0 THEN
  MOVE FA-NO TO FA-NOMINAL, ACCT-NO
  READ FINANCIAL-ACCT-TABFI INVALID KEY
                                         MOVE SPACES TO ACCT-NAME.
IF FLAG-2 THEN
  READ FINANCIAL-ACCT-TABFI INVALID KEY
                                         MOVE SPACES TO ACCT-NAME.
IF FLAG-0 OR FLAG-2 THEN
  MOVE FA-NAME-1 TO ACCT-NAME
  MOVE HEAD-STRIP-1A TO HEAD-STRIP-OUT.
IF FA-NO NOT EQUAL FA-NOMINAL GO TO WRITE-DATA-LINE.
ADD AMT-1 TO S-TOT-1.
ADD AMT-3 TO S-TOT-2.
ADD AMT-5 TO S-TOT-3.
ADD AMT-4 TO S-TOT-4.
MOVE '1' TO PRINT-REQ-FLAG.
GO TO READ-SORTED-FILE.
WRITE-DATA-LINE.
IF I = 7 THEN
  MOVE S-TOT-1 TO COL-1A
  MOVE S-TOT-2 TO COL-2A
  ADD S-TOT-1 TO TOT-1
  ADD S-TOT-2 TO TOT-2
  ELSE PERFORM ADD-SUB-TO-TOTALS.
PERFORM WRITE-DATA-LINE-A.
PERFORM PAGE-OVFLOW.
ADD S-TOT-5 TO TOT-5.
MOVE AMT-1 TO S-TOT-1.
MOVE AMT-3 TO S-TOT-2.
MOVE AMT-5 TO S-TOT-3.
MOVE AMT-4 TO S-TOT-4.
MOVE '2' TO PRINT-REQ-FLAG.
MOVE FA-NO TO FA-NOMINAL, ACCT-NO.
TEST-ACCT-LIMIT.
IF FA-NO-1 NOT GREATER THAN HI-LIM (I)
                                         GO TO READ-SORTED-FILE.

```

```

IF FLAG-1 THEN PERFORM ADD-SUB-TD-TOTALS
    PERFORM WRITE-DATA-LINE-A
    PERFORM PAGE-OVFLOW
    MOVE ZEROS TO S-TOT-1, S-TOT-2, S-TOT-3,
        S-TOT-4, S-TOT-5, S-TOT-6
    MOVE '0' TO PRINT-REQ-FLAG.
    GO TO TEST-ACCT-LIMIT-END.
READ-FILE-COMPLETE.
IF FLAG-2 THEN
    READ FINANCIAL-ACCT-TABFI INVALID KEY
        MOVE SPACES TO ACCT-NAME.
IF FLAG-2 THEN
    MOVE FA-NAME-1 TO ACCT-NAME
    MOVE HEAD-STRIP-1A TO HEAD-STRIP-OUT.
IF FLAG-0 THEN GO TO TEST-ACCT-LIMIT-END.
IF I = 7 THEN
    MOVE S-TOT-1 TO COL-1A
    MOVE S-TOT-2 TO COL-2A
    ADD S-TOT-1 TO TOT-1
    ADD S-TOT-2 TO TOT-2
    ELSE PERFORM ADD-SUB-TD-TOTALS.
    PERFORM WRITE-DATA-LINE-A.
    MOVE SPACES TO FA-NOMINAL.
TEST-ACCT-LIMIT-END.
IF I = 6 PERFORM WRITE-TOTALS.
WRITE-TOTALS.
    PERFORM SKIP-LINE.
    MOVE NAME (8) TO NAME-8.
    IF I = 6 THEN DIVIDE 12 INTO TOT-4 GIVING TOT-3,
        MULTIPLY TOT-3 BY MONTH GIVING TOT-3.
    IF I GREATER THAN 6 MOVE 'TOTAL' TO NAME-8.
    MOVE HEAD-STRIP-2A TO HEAD-STRIP-OUT.
    MOVE TOT-1 TO COL-1A.
    MOVE TOT-2 TO COL-2A, TEMP-1.
    MOVE TOT-3 TO COL-3A, TEMP-2.
    MOVE TOT-4 TO COL-4A.
    PERFORM VARI-AND-PERCENT.
    PERFORM WRITE-DATA-LINE-A.
ADD-SUB-TD-TOTALS.
    MULTIPLY MONTH BY S-TOT-4 GIVING TEMP-3.
    DIVIDE 12 INTO TEMP-3.
    ADD .005 TO TEMP-3.
    MOVE TEMP-3 TO S-TOT-3.
    ADD S-TOT-1 TO TOT-1.
    ADD S-TOT-2 TO TOT-2.
    ADD S-TOT-3 TO TOT-3.
    ADD S-TOT-4 TO TOT-4.
    IF ACCT-NO EQUAL '932-50' THEN
        COMPUTE P-ERROR = (MONTH * TOT-4 / 12) - TOT-3;
        ADD P-ERROR TO S-TOT-3.

```

```

IF MONTH EQUAL 12 THEN MOVE S-TOT-4 TO S-TOT-3.
MOVE S-TOT-1 TO COL-1A.
MOVE S-TOT-2 TO COL-2A, TEMP-1.
MOVE S-TOT-3 TO COL-3A, TEMP-2.
MOVE S-TOT-4 TO COL-4A.
PERFORM VARI-AND-PERCENT.

DEPREC-INT-TAXES.
  IF CC-NO-1 EQUAL '5' THEN
    ADD AMT-1 TO INT-C
    ADD AMT-3 TO INT-P
    MOVE '1' TO INT-FLAG
    GO TO TAXES-END.
  IF CC-NO-1 EQUAL '6' THEN
    ADD AMT-1 TO TAX-C
    ADD AMT-3 TO TAX-P
    MOVE '1' TO INT-FLAG
    GO TO TAXES-END.
  IF SA-NO EQUAL '10' THEN
    ADD AMT-1 TO DEPRE-C
    MOVE '1' TO INT-FLAG.

TAXES-END.
  EXIT.

WRITE-DEPREC-INT-TAXES.
  MOVE SPACES TO NEG-OUT.
  PERFORM SKIP-LINE.
  MOVE CAT-NAME (10) TO NAME-8.
  MOVE HEAD-STRIP-2A TO HEAD-STRIP-OUT.
  MULTIPLY MONTH BY DEPRE-C GIVING DEPRE-P.
  MOVE DEPRE-C TO COL-1A.
  MOVE DEPRE-P TO COL-2A.
  MOVE ZEROS TO COL-3A, COL-4A, COL-5A, COL-6A.
  PERFORM WRITE-DATA-LINE-A.
  PERFORM PAGE-OVFLOW.
  ADD DEPRE-C TO TOT-1.
  ADD DEPRE-P TO TOT-2.
  MOVE TOT-2 TO TOT-T3
  PERFORM WRITE-TOTALS.
  MOVE SPACES TO NEG-OUT.
  MOVE CC-NAME (5) TO NAME-8.
  MOVE HEAD-STRIP-2A TO HEAD-STRIP-OUT.
  MOVE INT-C TO COL-1A.
  MOVE INT-P TO COL-2A.
  MOVE ZEROS TO COL-3A, COL-4A, COL-5A, COL-6A.
  PERFORM WRITE-DATA-LINE-A.
  ADD INT-C TO TOT-1.
  ADD INT-P TO TOT-2.
  MOVE CC-NAME (6) TO NAME-8.
  MOVE HEAD-STRIP-2A TO HEAD-STRIP-OUT.
  MOVE TAX-C TO COL-1A.
  MOVE TAX-P TO COL-2A.

```

```

PERFORM WRITE-DATA-LINE-A.
ADD TAX-C TO TOT-1.
ADD TAX-P TO TOT-2.
*****
*
*
*          SUBROUTINES FOR PART-B
*
*****
WRITE-SUB-TITLE-B.
MOVE HEADING-8 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
PERFORM SKIP-LINE.
MOVE HEADING-9 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-10 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-11 TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
PERFORM SKIP-LINE 2 TIMES.
WRITE-COST-CENTER.
PERFORM PAGE-OVFLOW.
MOVE CC-NAME (J) TO CC-OUT, CC-GUTS.
MOVE DATA-LINE-D TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
IF J < 5 MOVE 'PAYROLL COSTS:'      ' TO C-CAT-SUB,
      MOVE DATA-LINE-E TO PRINT-LINE,
      PERFORM PRINT-LINE-OUT.
PERFORM DOUBLE-SORTED-FILE THRU DATA-LINE-EXIT      VARYING K
      FROM 1 BY 1 UNTIL K GREATER THAN 10.
PERFORM SKIP-LINE 3 TIMES.
COST-CENTER-END.
EXIT.
DOUBLE-SORTED-FILE.
MOVE '0' TO PRINT-REQ-FLAG.
READ-DS-FILE.
READ SORTED-FILE AT END GO TO DS-FILE-COMPLETE.
MULTIPLY MONTH BY AMT-6.
DIVIDE AMT-6 BY 12 GIVING AMT-6.
IF FLAG-YES THEN PERFORM INT-AND-TAXES.
IF FLAG-0 THEN
      MOVE CAT-NAME (K) TO C-CAT-OUT
      MOVE HEAD-STRIP-1B TO HEAD-OUT.
MOVE CC-NO-1 TO CC-J.
MOVE SA-NO TO SA-K.
IF CC-J NOT EQUAL J OR SA-K NOT EQUAL K THEN
      GO TO WRITE-DATA-LINES.
MOVE '1' TO PRINT-REQ-FLAG.
ADD AMT-3 TO S-TOT-1.
ADD AMT-6 TO S-TOT-2.
GO TO READ-DS-FILE.

```

```

WRITE-DATA-LINES.
    ADD S-TOT-1 TO TOT-1, TOT-3, TEMP-1.
    ADD S-TOT-2 TO TOT-2, TOT-4, TEMP-2.
    MOVE S-TOT-1 TO COL-1B.
    MOVE S-TOT-2 TO COL-2B.
    PERFORM VARI-AND-PERCENT.
    MOVE AMT-3 TO S-TOT-1.
    MOVE AMT-6 TO S-TOT-2.
    IF K EQUAL 6 AND J LESS THAN 5 THEN
        MOVE 'OTHER DIRECT COSTS:' TO C-CAT-SUB
        MOVE DATA-LINE-E TO PRINT-LINE
        PERFORM PRINT-LINE-OUT.
    PERFORM WRITE-DATA-LINE-B.
    PERFORM WRITE-CC-TOTALS.
    PERFORM PAGE-OVFLOW.
    MOVE '0' TO PRINT-REQ-FLAG.
    IF K EQUAL 10 AND J EQUAL 4 THEN
        MOVE '1' TO INT-FLAG
        GO TO READ-DS-FILE.

DATA-LINE-EXIT.
    EXIT.

WRITE-CC-TOTALS.
    IF K EQUAL 10 AND J LESS THAN 5 THEN
        MOVE TOT-3 TO COL-1B; TEMP-1
        MOVE TOT-4 TO COL-2B; TEMP-2
        MOVE HEAD-STRIP-2B TO HEAD-OUT
        PERFORM VARI-AND-PERCENT
        PERFORM WRITE-DATA-LINE-B
        MOVE ZEROS TO TOT-3, TOT-4, TOT-5, TOT-6.

DS-FILE-COMPLETE.
    IF FLAG-1 THEN
        ADD S-TOT-1 TO TOT-1, TEMP-1
        ADD S-TOT-2 TO TOT-2, TEMP-2
        MOVE S-TOT-1 TO COL-1B
        MOVE S-TOT-2 TO COL-2B
        PERFORM VARI-AND-PERCENT
        PERFORM WRITE-DATA-LINE-B
        PERFORM WRITE-CC-TOTALS.

    GO TO DATA-LINE-EXIT.

WRITE-OP-COSTS.
    MOVE 'OPERATING'           TO CC-OUTS.
    MOVE TOT-1 TO COL-1B, TEMP-1.
    MOVE TOT-2 TO COL-2B, TEMP-2.
    MOVE TOT-1 TO TOT-T5
    PERFORM VARI-AND-PERCENT.
    MOVE HEAD-STRIP-2B TO HEAD-OUT.
    PERFORM WRITE-DATA-LINE-B.

INT-AND-TAXES.
    IF CC-NO-1 EQUAL '5' THEN
        ADD AMT-3 TO INT-C

```

```

        ADD AMT-6 TO INT-P
        GO TO READ-DS-FILE.
IF CC-NO-1 EQUAL '6' THEN
        ADD AMT-3 TO TAX-C
        ADD AMT-6 TO TAX-P.
GO TO READ-DS-FILE.

WRITE-INTEREST.
PERFORM SKIP-LINE.
ADD INT-C TO TOT-1.
ADD INT-P TO TOT-2.
MOVE INT-C TO COL-1B, TEMP-1.
MOVE INT-P TO COL-2B, TEMP-2.
PERFORM VARI-AND-PERCENT.
MOVE CC-NAME (5) TO C-CAT-OUT.
MOVE HEAD-STRIP-1B TO HEAD-OUT.
PERFORM WRITE-DATA-LINE-B.

WRITE-TAXES.
ADD TAX-C TO TOT-1.
ADD TAX-P TO TOT-2.
MOVE TAX-C TO COL-1B, TEMP-1.
MOVE TAX-P TO COL-2B, TEMP-2.
PERFORM VARI-AND-PERCENT.
MOVE CC-NAME (6) TO C-CAT-OUT.
MOVE HEAD-STRIP-1B TO HEAD-OUT.
PERFORM WRITE-DATA-LINE-B.

WRITE-LAST-LINE.
PERFORM SKIP-LINE.
MOVE TOT-1 TO COL-1B, TEMP-1.
MOVE TOT-2 TO COL-2B, TEMP-2.
PERFORM VARI-AND-PERCENT.
MOVE 'ALL ITEMS' TO CC-OUTS.
MOVE HEAD-STRIP-2B TO HEAD-OUT.
PERFORM WRITE-DATA-LINE-B.
PERFORM TOP-NEW-PAGE.
IF TOT-T3 NOT EQUAL TOT-T5 GO TO WRITE-ERROR.
IF TOT-T4 NOT EQUAL TOT-1 GO TO WRITE-ERROR.

WRITE-ERROR.
PERFORM WRITE-PAGE-HEADING.
PERFORM SKIP-LINE 5 TIMES.
MOVE 'ERROR -- PART-A AND PART-B TOTALS DO NOT MATCH. '
      TO PRINT-LINE.
PERFORM PRINT-LINE-OUT.
MOVE ZERO TO COL-6B.
MOVE TOT-T3 TO COL-1B.
MOVE TOT-T5 TO COL-2B.
SUBTRACT TOT-T5 FROM TOT-T3 GIVING COL-5B.
MOVE 'PART-A, PART-B TOTALS & ERROR.' TO C-CAT-OUT.
MOVE HEAD-STRIP-1B TO HEAD-OUT.
PERFORM WRITE-DATA-LINE-B.
MOVE TOT-T4 TO COL-1B.

```

```

MOVE TOT-1 TO COL-2B.
SUBTRACT TOT-1 FROM TOT-T4 GIVING COL-5B.
PERFORM WRITE-DATA-LINE-B.
PERFORM TOP-NEW-PAGE.
PERFORM WRITE-COUNTERS.
GO TO CLOSE-FILES.

*****
*
*          FINI
*
*****

READ-SORT-INPUT.
  READ COST-MATRIX-FILE AT END
    GO TO READ-SORT-INPUT-END.
  IF CC-NU EQUAL '4301' OR '4302' OR '4401'
    OR '4321' OR '4322' OR '4421'
    GO TO READ-SORT-INPUT.
  IF SA-NU GREATER THAN '12' THEN GO TO READ-SORT-INPUT.
  RELEASE SORT-INPUT FROM CM-REC.
  ADD 1 TO SCT-ONE.
  GO TO READ-SORT-INPUT.
READ-SORT-INPUT-END.
  EXIT.

WRITE-SORT-OUTPUT.
  RETURN FILE-SORT RECORD INTO INPUT-REC
    AT END GO TO WRITE-SORT-OUTPUT-END.
  WRITE INPUT-REC.
  ADD 1 TO SCT-TWO.
  GO TO WRITE-SORT-OUTPUT.
WRITE-SORT-OUTPUT-END.
  EXIT.

WRITE-COUNTERS.
  MOVE 'NUMBER OF ACCOUNTS SORTED' TO HEAD-OUT.
  MOVE SCT-ONE TO COL-1B.
  MOVE SCT-TWO TO COL-2B.
  MOVE DATA-LINE-B TO PRINT-LINE.
  MOVE ZEROS TO COL-5B, COL-6B.
  PERFORM PRINT-LINE-OUT.
  MOVE ZEROS TO SCT-ONE, SCT-TWO.
  MOVE ZEROS TO COL-1B, COL-2B, COL-5B.

CLOSE-FILES.
  CLOSE CARD-IN, COST-MATRIX-FILE,
    FINANCIAL-ACCT-TABFI,
    SORTED-FILE, PRINT-OUT.

E-O-J.
  STOP RUN.

```

```
PGMNAME=WUASPLV1
.SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID. WUASPLV1.
AUTHOR. ACT SYSTEMS, INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FLA 32789.
DATE-WRITTEN. 02/06/78.
REMARKS. THIS PROGRAM READS COST MATRIX FILE
        AND BUILDS THE LEVEL 1 REPORT.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    CO1 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS SEQUENTIAL
        RECORD KEY IS CM-NUMBER.
DATA DIVISION.
FILE SECTION.
FD COST-MATRIX-FILE
    RECORD CONTAINS 80 CHARACTERS
    BLOCK CONTAINS 5 RECORDS
    LABEL RECORDS ARE STANDARD
    DATA RECORD IS INPUT-REC.
01 INPUT-REC.
    05 MATCH-CODE          PIC X(4).
    05 CM-NUMBER.
        10 FILLER      PIC X(10).
        10 COST-CNTR.
            15 POS1      PIC 9.
            15 FILLER    PIC XXX.
        10 COST-CAT     PIC 99.
    05 AMOUNTS.
        10 FIGS OCCURS 6 TIMES  PIC S9(8)V99.
FD CARD-IN
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE OMITTED
    DATA RECORD IS INPUT-CARD.
```

```

01 INPUT-CARD.
  05 DATE-IN          PIC X(18).
  05 TITLE-IN         PIC X(24).
  05 FILLER           PIC X(38).

01 NUMBER-CARD.
  05 MIL-GALS OCCURS 4 TIMES PIC 9(7)V999.
    05 ACTUAL-TD-BUDGET   PIC 9(3)V99.
    05 FILLER             PIC X(35).

*
* ABOVE MIL-GALS IS AS FOLLOWS
* (1) RPW (REVENUE PRODUCING WATER)
* (2) TREATED WATER MONTH
* (3) RPW YEAR TO DATE
* (4) RPW PREVIOUS YEAR.
*

FD PRINT-OUT
  RECORD CONTAINS 133 CHARACTERS
  LABEL RECORDS ARE OMITTED
  DATA RECORD IS PRINT-LINE.

01 PRINT-LINE.
  05 FILLER           PIC X.
  05 PRINT-DATA        PIC X(132).

WORKING-STORAGE SECTION.
  77 MILLION-GAL      PIC S9(5)V999 VALUE ZERO.
  77 PREVIOUS-MILLION-GAL PIC S9(5)V999 VALUE ZERO.
  77 ACCUM-ACROSS     PIC S9(8)V99 VALUE ZERO.
  77 SAVE-AMT          PIC S9(8)V99 VALUE ZERO.
  77 I                 PIC 99 VALUE 0.
  77 PAGE-CTR          PIC 99 VALUE 0.
  77 J                 PIC 99 VALUE 0.
  77 K                 PIC 99 VALUE 0.
  77 PCT-CTR           PIC V9999.
  77 FIN-PCT            PIC 999V9999.

01 HEADING-1.
  05 FILLER           PIC X(50) VALUE SPACES.
  05 FILLER           PIC X(30) VALUE 'SUMMARY OF SYSTEMS INFORMATI
  -                   'ON'.

  05 FILLER           PIC X(52) VALUE SPACES.

01 HEADING-2.
  05 FILLER           PIC X(6) VALUE 'DATE: '.
  05 DATE-OUT          PIC X(18) VALUE SPACES.
  05 FILLER           PIC X(40) VALUE SPACES.
  05 FILLER           PIC XXX VALUE 'FOR'.
  05 FILLER           PIC X(55) VALUE SPACES.
  05 FILLER           PIC X(6) VALUE 'PAGE: '.
  05 PAGE-NR          PIC Z(4).

01 HEADING-3.
  05 FILLER           PIC X(53) VALUE SPACES.
  05 TITLE-OUT          PIC X(24) VALUE SPACES.
  05 FILLER           PIC X(55) VALUE SPACES.

```

```

01  HEADING-4.
  05 FILLER      PIC X(26) VALUE '                   CATEGORY   '.
  05 FILLER      PIC X(26) VALUE '                   ACQUISITION TREAT'.
  05 FILLER      PIC X(26) VALUE 'MENT'      DELIVERY    SUPPO'.
  05 FILLER      PIC X(26) VALUE 'RT'        INTEREST    TOTAL'.
  05 FILLER      PIC X(26) VALUE '           TAXES      TOTAL '.

01  HEADING-5.
  05 FILLER      PIC X(60) VALUE SPACES.
  05 FILLER      PIC X(12) VALUE '( LEVEL I )'.
  05 FILLER      PIC X(60) VALUE SPACES.

01  HEADING-6.
  05 FILLER      PIC X(26) VALUE '           *****'.
  05 FILLER      PIC X(26) VALUE '           *****'*****'.
  05 FILLER      PIC X(26) VALUE '****'      *****'*****'.
  05 FILLER      PIC X(26) VALUE '**'        *****'*****'.
  05 FILLER      PIC X(26) VALUE '           ****'*****'.

01  DATA-LINE.
  05 FILLER      PIC X(10) VALUE SPACES.
  05 DESCRIPTION PIC X(20) VALUE SPACES.
  05 FILLER      PIC X(3) VALUE SPACES.
  05 POSIT OCCURS 8 TIMES.
    10 AMTS      PIC -(8).--.
    10 FILLER    PIC X.
  05 FILLER      PIC XXX VALUE SPACES.

01  PCT-LINE.
  05 FILLER      PIC X(10) VALUE SPACES.
  05 DESCRIPTION1 PIC X(20) VALUE SPACES.
  05 FILLER      PIC X(3) VALUE SPACES.
  05 PERCENT    PIC -(8).--- OCCURS 8 TIMES.
  05 FILLER      PIC XXX VALUE SPACES.

01  MATRIX.
  05 FUNCTIONS OCCURS 6 TIMES.
    10 SUB-FUNCTIONS OCCURS 5 TIMES.
      15 ACCT      PIC S9(8)V99.

01  TITLE-1.
  05 TITLE-ONE.
    10 FILLER PIC X(20) VALUE 'CURRENT PERIOD   '.
    10 FILLER PIC X(20) VALUE 'PREVIOUS PERIOD '.
    10 FILLER PIC X(20) VALUE 'CURRENT PERIOD   '.
    10 FILLER PIC X(20) VALUE 'CURRENT YEAR (YTD)'.
    10 FILLER PIC X(20) VALUE 'PREVIOUS YEAR (YTD)'.
  05 TITLE-TWO REDEFINES TITLE-ONE.
    10 TITLE OCCURS 5 TIMES PIC X(20).

01  TITLE-1-A.
  05 TITLE-ONE-A.
    10 FILLER PIC X(20) VALUE 'UNIT COST $/MG   '.
    10 FILLER PIC X(20) VALUE 'UNIT COST $/MG   '.
    10 FILLER PIC X(20) VALUE 'PREVIOUS YEAR $/MG '.
    10 FILLER PIC X(20) VALUE 'UNIT COST $/MG   '.
    10 FILLER PIC X(20) VALUE 'UNIT COST $/MG   '.

```

```

      05 TITLE-TWO-A REDEFINES TITLE-ONE-A.
      10 TITLE-A OCCURS 5 TIMES PIC X(20).
01  TITLE-3.
      05 TITLE-THREE.
          10 FILLER PIC X(20) VALUE 'CURRENT PERIOD' .
          10 FILLER PIC X(20) VALUE 'YEAR TO DATE' .
      05 TITLE-FOUR REDEFINES TITLE-THREE.
          10 TITLE-B OCCURS 2 TIMES PIC X(20).
01  TITLE-3-A.
      05 TITLE-THREE-A.
          10 FILLER PIC X(20) VALUE 'TOTAL COST' .
          10 FILLER PIC X(20) VALUE 'TOTAL COST' .
      05 TITLE-FOUR-A REDEFINES TITLE-THREE-A.
          10 TITLE-X OCCURS 2 TIMES PIC X(20).
01  TITLE-5.
      05 TITLE-FIVE.
          10 FILLER PIC X(20) VALUE '% OF TOTAL' .
          10 FILLER PIC X(20) VALUE '% OF PREVIOUS' .
      05 TITLE-SIX REDEFINES TITLE-FIVE.
          10 TITLE-C OCCURS 2 TIMES PIC X(20).
01  TITLE-5-A.
      05 TITLE-FIVE-A.
          10 FILLER PIC X(20) VALUE 'YEAR TO DATE' .
          10 FILLER PIC X(20) VALUE 'TOTAL (YTD)' .
      05 TITLE-SIX-A REDEFINES TITLE-FIVE-A.
          10 TITLE-D OCCURS 2 TIMES PIC X(20).

PROCEDURE DIVISION.

OPEN-FILES.
    OPEN INPUT CARD-IN, COST-MATRIX-FILE
        OUTPUT PRINT-OUT.

HOUSEKEEPING.
    MOVE ZEROS TO MATRIX, PCT-CTR.

READ-TITLE-CARD.
    READ CARD-IN, AT END GO TO END-OF-JOB.
    MOVE DATE-IN TO DATE-OUT.
    MOVE TITLE-IN TO TITLE-OUT.

READ-PREVIOUS-DIVISOR-CARD.
    READ CARD-IN AT END GO TO END-OF-JOB.
    EXAMINE NUMBER-CARD REPLACING ALL ' ' BY '0'.
    MOVE MIL-GALS (1) TO PREVIOUS-MILLION-GAL.

READ-DIVISOR-CARD.
    READ CARD-IN, AT END GO TO END-OF-JOB.
    EXAMINE NUMBER-CARD REPLACING ALL ' ' BY '0'.
    MOVE MIL-GALS (1) TO MILLION-GAL.

WRITE-HEADINGS.
    ADD 1 TO PAGE-CTR.
    PERFORM MOVE-SPACES.
    MOVE PAGE-CTR TO PAGE-NR.
    WRITE PRINT-LINE BEFORE ADVANCING NEW-PAGE.
    MOVE HEADING-1 TO PRINT-DATA.

```

```

PERFORM PRINT-LINE-OUT.
MOVE HEADING-2 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-3 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-5 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-4 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE HEADING-6 TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM SKIP-LINE 2 TIMES.

PROCESS-MATRIX.
  READ COST-MATRIX-FILE AT END GO TO PROCESS-TIME.
  IF (COST-CAT < 1 OR > 12) OR (POS1 < 1 OR > 6) THEN
    GO TO PROCESS-MATRIX.
  ADD FIGS (1) TO ACCT (POS1, 1).
  ADD FIGS (2) TO ACCT (POS1, 2).
  ADD FIGS (3) TO ACCT (POS1, 4).
  ADD FIGS (6) TO ACCT (POS1, 3).
  ADD FIGS (6) TO ACCT (POS1, 5).
  GO TO PROCESS-MATRIX.

PROCESS-TIME.
  PERFORM MOVE-AMOUNTS VARYING J FROM 1 BY 1 UNTIL J > 5.
  MOVE 1 TO J K.
  PERFORM MOVE-AMOUNTS-1 2 TIMES.
  PERFORM GET-PERCENTAGES.
  GO TO END-OF-JOB.

MOVE-AMOUNTS.
  IF J = 2 THEN
    MOVE PREVIOUS-MILLION-GAL TO MILLION-GAL.
  IF J = 4 THEN
    MOVE MIL-GALS (3) TO MILLION-GAL.
  PERFORM MOVE-AMOUNTS-A VARYING I FROM 1 BY 1 UNTIL I > 5.
  MOVE ACCUM-ACROSS TO AMTS OF POSIT (6).
  DIVIDE ACCT (6, J) BY MILLION-GAL GIVING AMTS OF POSIT (7).
  DIVIDE ACCT (6, J) BY MILLION-GAL GIVING SAVE-AMT.
  ADD ACCUM-ACROSS, SAVE-AMT GIVING AMTS OF POSIT (8).
  MOVE TITLE (J) TO DESCRIPTION.
  MOVE DATA-LINE TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE SPACES TO DATA-LINE.
  MOVE TITLE-A (J) TO DESCRIPTION.
  MOVE DATA-LINE TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
  MOVE SPACES TO DATA-LINE.
  MOVE ZEROS TO ACCUM-ACROSS.

```

MOVE-AMOUNTS-1.

PERFORM MOVE-AMOUNTS-B VARYING I FROM 1 BY 1 UNTIL I > 5.

MOVE ACCUM-ACROSS TO AMTS OF POSIT (6).

MOVE ACCT (6, J) TO AMTS OF POSIT (7).

ADD ACCT (6, J), ACCUM-ACROSS GIVING AMTS OF POSIT (8).

MOVE TITLE-B (K) TO DESCRIPTION.

PERFORM DATA-LINE-OUT.

MOVE SPACES TO DATA-LINE.

MOVE TITLE-X (K) TO DESCRIPTION.

PERFORM DATA-LINE-OUT THRU PRINT-LINE-OUT.

MOVE ZEROS TO ACCUM-ACROSS.

MOVE SPACES TO DATA-LINE.

ADD 1 TO K.

ADD 3 TO J.

MOVE-AMOUNTS-A.

DIVIDE ACCT (I, J) BY MILLION-GAL GIVING AMTS OF POSIT (1).

DIVIDE ACCT (I, J) BY MILLION-GAL GIVING SAVE-AMT.

ADD SAVE-AMT TO ACCUM-ACROSS.

MOVE-AMOUNTS-B.

MOVE ACCT (I, J) TO AMTS OF POSIT (1).

ADD ACCT (I, J) TO ACCUM-ACROSS.

GET-PERCENTAGES.

PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.

MOVE 'BUDGET VARIANCE' TO DESCRIPTION.

MOVE ACTUAL-TO-BUDGET TO AMTS OF POSIT (8).

PERFORM DATA-LINE-OUT.

MOVE SPACES TO DATA-LINE.

MOVE 'YEAR TO DATE' TO DESCRIPTION.

PERFORM DATA-LINE-OUT THRU PRINT-LINE-OUT.

MOVE 'TREATED WATER' TO DESCRIPTION.

MOVE MIL-GALS (2) TO AMTS OF POSIT (8).

PERFORM DATA-LINE-OUT.

MOVE SPACES TO DATA-LINE.

MOVE 'CURRENT PERIOD' TO DESCRIPTION.

PERFORM DATA-LINE-OUT THRU PRINT-LINE-OUT.

MOVE 'REVENUE PRODUCING' TO DESCRIPTION.

MOVE MIL-GALS (1) TO AMTS OF POSIT (8).

PERFORM DATA-LINE-OUT.

MOVE SPACES TO DATA-LINE.

MOVE 'WATER CURRENT PERIOD' TO DESCRIPTION.

PERFORM DATA-LINE-OUT THRU PRINT-LINE-OUT.

MOVE 'REVENUE PRODUCING' TO DESCRIPTION.

MOVE MIL-GALS (3) TO AMTS OF POSIT (8).

PERFORM DATA-LINE-OUT.

MOVE SPACES TO DATA-LINE.

MOVE 'WATER (YTD)' TO DESCRIPTION.

PERFORM DATA-LINE-OUT THRU PRINT-LINE-OUT.

MOVE 'REVENUE PRODUCING' TO DESCRIPTION.

MOVE MIL-GALS (4) TO AMTS OF POSIT (8).

PERFORM DATA-LINE-OUT.

MOVE SPACES TO DATA-LINE.
MOVE 'WATER PREVIOUS YTD' TO DESCRIPTION.
PERFORM DATA-LINE-OUT THRU PRINT-LINE-OUT.
MOVE 1 TO K.
MOVE 4 TO J.
PERFORM PERCENT-ROUTINE 2 TIMES.
PERCENT-ROUTINE.
MOVE ZEROS TO PCT-CTR.
PERFORM FIND-PERCENT VARYING I FROM 1 BY 1 UNTIL I > 5.
ADD ACCT (6, J) TO ACCUM-ACROSS.
PERFORM FIND-PERCENT-A VARYING I FROM 1 BY 1 UNTIL I > 5.
IF ACCUM-ACROSS > 0 THEN DIVIDE ACCT (6, J) BY ACCUM-ACROSS
GIVING PCT-CTR ROUNDED ELSE MOVE ZEROS TO PCT-CTR.
MULTIPLY PCT-CTR BY 100 GIVING PERCENT (7).
IF ACCUM-ACROSS > 0 THEN
MOVE 100 TO PERCENT (8).
MOVE TITLE-C (K) TO DESCRIPTION1.
MOVE PCT-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE SPACES TO PCT-LINE.
MOVE TITLE-D (K) TO DESCRIPTION1.
MOVE PCT-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
ADD 1 TO J K.
FIND-PERCENT.
ADD ACCT (I, J) TO ACCUM-ACROSS.
FIND-PERCENT-A.
IF ACCUM-ACROSS > 0 THEN DIVIDE ACCT (I, J) BY ACCUM-ACROSS
GIVING PCT-CTR ROUNDED ELSE MOVE ZEROS TO PCT-CTR.
MULTIPLY PCT-CTR BY 100 GIVING PERCENT (I).
DATA-LINE-OUT.
MOVE DATA-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
MOVE-SPACES.
MOVE SPACES TO PRINT-DATA.
PRINT-LINE-OUT.
WRITE PRINT-LINE BEFORE ADVANCING 1.
SKIP-LINE.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
END-OF-JOB.
CLOSE CARD-IN, COST-MATRIX-FILE
PRINT-OUT.
STOP RUN.

PGMNAME=WUASPWAR
SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASPWAR.
AUTHOR. ACT SYSTEMS, INC
SUITE 200
807 W MORSE BLVD
WINTER PARK, FL 32789.
DATE-WRITTEN. JAN 1978.
DATE-COMPILED.
REMARKS. THIS PROGRAM READS THE COST MATRIX FILE
AND PRINTS THE WATER UTILITY REPORT
CALLED 'WATER SUPPLY DATA ANALYSIS REPORT'.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT CARD-IN
 ASSIGN TO UR-2540R-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
 SELECT COST-MATRIX-FILE
 ASSIGN TO DA-3330-I-CMFILE
 ACCESS MODE IS SEQUENTIAL
 RECORD KEY IS CM-NUMBER.
 SELECT FILE-SORT
 ASSIGN TO UT-3330-S-FILESORT
 ACCESS MODE IS SEQUENTIAL.
 SELECT SORTED-FILE
 ASSIGN TO UT-3330-S-SORTEDFI
 ACCESS MODE IS SEQUENTIAL.
 SELECT PRINT-OUT
 ASSIGN TO UR-1403-S-PRINTOUT
 ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS DATE-CARD.
01 DATE-CARD.
 05 DATE-IN.
 10 MO-IN PIC X(3).
 10 DAY-YR PIC X(15).
 05 TITLE-IN PIC X(24).
 05 FILLER PIC X(8).
 05 FISCAL-YR PIC X(24).

```

05 FILLER          PIC X(4).
05 NO-WKS          PIC X.
05 RPT-PERIOD     PIC X.
01 WATER-CARD.
05 WU-NAME         PIC XX.
05 SYSTEM           PIC 99.
05 SERIES           PIC 99.
05 PARALLEL         PIC 99.
05 SER-PAR          PIC 99.
05 WATER-SULD      PIC S9(7)V999.
05 ZONE-WATER.
10 MIL-GAL OCCURS 6 TIMES PIC S9(7)V999.
FD COST-MATRIX-FILE
RECORD CONTAINS 80 CHARACTERS
BLOCK CONTAINS 5 RECORDS
LABEL RECORDS ARE STANDARD
DATA RECORD IS CM-REC.
01 CM-REC.
05 MATCH-NU        PIC X(4).
05 CM-NUMBER.
10 FA-NU           PIC X(10).
10 CC-NU           PIC X(4).
10 SA-NU           PIC XX.
05 AMT-S            PIC S9(8)V99 OCCURS 6 TIMES.
*****
*      THE ABOVE AMOUNTS ARE AS FOLLOWS
*      (1) -- CURRENT COST AMOUNT
*      (2) -- PREVIOUS MONTH COST AMOUNT
*      (3) -- CURRENT COST AMOUNT YEAR TO DATE (YTD)
*      (4) -- BUDGET AMOUNT
*      (5) -- BUDGET AMOUNT YEAR TO DATE (YTD)
*      (6) -- PREVIOUS YEAR COST AMOUNT (PYC)
*****
SD FILE-SORT
RECORD CONTAINS 80 CHARACTERS
DATA RECORD IS SORT-INPUT.
01 SORT-INPUT.
05 FILLER          PIC X(4).
05 FIRST-10         PIC X(10).
05 CC-NOS           PIC X(4).
05 SA-NOS           PIC XX.
05 FILLER          PIC X(60).
FD SORTED-FILE
BLOCK CONTAINS 5 RECORDS
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS INPUT-REC.
01 INPUT-REC.
05 MATCH-NO         PIC X(4).
05 FA-NO.

```

```

10 FA-NO-1      PIC X(2).
10 FA-NO-2      PIC X(8).

05 CC-NO.
10 CC-NO-1      PIC X.
10 CC-NO-2      PIC X.
10 CC-NO-3      PIC X.
10 CC-NO-4      PIC X.

05 SA-NO        PIC XX.

05 AMTS.
10 AMT-1        PIC S9(8)V99.
10 AMT-2        PIC S9(8)V99.
10 AMT-3        PIC S9(8)V99.
10 AMT-4        PIC S9(8)V99.
10 AMT-5        PIC S9(8)V99.
10 AMT-6        PIC S9(8)V99.

FD PRINT-DUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-DATA.

01 PRINT-DATA.
05 FILLER        PIC X.
05 PRINT-LINE    PIC X(132).

WORKING-STORAGE SECTION.
77 LINE-CT        PIC 9(4) COMP VALUE 0.
77 PAGE-CT        PIC 9(4) COMP VALUE 1.
77 MAX-LINE       PIC 99   COMP VALUE 57.
77 I               PIC 99   COMP VALUE 0.
77 J               PIC 99   COMP VALUE 0.
77 K               PIC 99   COMP VALUE 0.
77 L               PIC 99   COMP VALUE 0.
77 N               PIC 99   COMP VALUE 1.
77 ZONES          PIC 99   COMP VALUE 1.
77 DOLS            PIC S9(8)V99   VALUE +0.
77 TEMP-1          PIC S9(8)V999  VALUE +0.
77 TEMP-2          PIC S9(8)V999  VALUE +0.
77 RPT-FLAG        PIC X   VALUE '1'.
88 RPT-A           VALUE '1'.
88 RPT-B           VALUE '2'.

01 COST-CENTERS.
05 CC-LIST.
10 CC-LIST-1      PIC X(120)  VALUE
                                'ACQUISITION'
                                'TREATMENT'
                                'TRANSMISSION'
                                'DISTRIBUTION'
                                'STORAGE'
                                'PUMPING'
10 CC-LIST-2      PIC X(120)  VALUE
                                'SUPPORT SERVICES'
                                'INTEREST'

```

```

          'TAXES
          'TOTAL
          'DELIVERY

      05 CC-NAMES REDEFINES CC-LIST.
      10 CC-NAME OCCURS 12 TIMES PIC X(20).

01 MATRIX.
  05 FUNCTIONS OCCURS 10 TIMES.
    10 SUB-FUNCTIONS OCCURS 7 TIMES.
      15 POSIT PIC S9(8)V99.

01 SUB-SUB-TOTAL.
  05 SS-TOT          PIC S9(8)V99      OCCURS 7 TIMES.

01 SUB-TOTAL.
  05 S-TOT          PIC S9(8)V99      OCCURS 7 TIMES.

01 COL-TOTAL.
  05 C-TOT          PIC S9(8)V99      OCCURS 7 TIMES.

*****  

*  

*          HEADINGS  

*  

*****  

01 HEADING-1.
  05 FILLER          PIC X(56) VALUE SPACES.
  05 FILLER          PIC X(20) VALUE 'COST ANALYSIS REPORT'.
  05 FILLER          PIC X(56) VALUE SPACES.

01 HEADING-2.
  05 FILLER          PIC X(12) VALUE 'LAST ENTRY: '.
  05 DATE-OUT        PIC X(18).
  05 FILLER          PIC X(34) VALUE SPACES.
  05 FILLER          PIC X(3)  VALUE 'FOR'.
  05 FILLER          PIC X(55) VALUE SPACES.
  05 FILLER          PIC X(6)  VALUE 'PAGE: '.
  05 PAGE-OUT        PIC Z(4).

01 HEADING-3.
  05 FILLER          PIC X(12) VALUE 'RPT PERIOD: '.
  05 RPT-PD          PIC X(8).
  05 WKS-OUT          PIC X(8).
  05 FILLER          PIC X(26) VALUE SPACES.
  05 TITLE-OUT        PIC X(24).
  05 FILLER          PIC X(54) VALUE SPACES.

01 HEADING-6.
  05 FILLER          PIC X(12) VALUE 'YTD START: '.
  05 FISC-YR          PIC X(24).
  05 FILLER          PIC X(96).

01 HEADING-4.
  05 FILLER          PIC X(20) VALUE 'ITEM DESCRIPTION '.
  05 FILLER          PIC X(12) VALUE SPACES.
  05 FILLER          PIC X(14) VALUE '           ZONE 1'.
  05 FILLER          PIC X(14) VALUE '           ZONE 2'.
  05 FILLER          PIC X(14) VALUE '           ZONE 3'.

```

```

      05 Z-4          PIC X(14).
      05 Z-5          PIC X(14).
      05 FILLER       PIC X(4)  VALUE SPACES.
      05 Z-OUT        PIC X(10).
      05 FILLER       PIC X(16)  VALUE      TOTALS   '.
01  HEADING-5.
      05 FILLER       PIC X(15)  VALUE '*****'.
      05 FILLER       PIC X(15)  VALUE '*****'.
      05 FILLER       PIC X(2)   VALUE SPACES.
      05 FILLER       PIC X(14)  VALUE '*****'.
      05 FILLER       PIC X(16)  VALUE '*****'.
01  DATA-LINE-A.
      05 LTR-OUT       PIC X(2).
      05 NAME-OUT      PIC X(30).
      05 COLS          PIC ---,---,---.-- OCCURS 7 TIMES.
      05 FILLER        PIC X(2)   VALUE SPACES.
01  DATA-LINE-B.
      05 FILLER        PIC X(31)  VALUE SPACES.
      05 PERCENT OCCURS 7 TIMES.
          10 PERC        PIC BBB-(9).-.
      05 FILLER        PIC X(3)   VALUE SPACES.
01  DATA-LINE-C.
      05 FILLER        PIC X(34)  VALUE SPACES.
      05 FILLER        PIC X(14)  VALUE '      %   '.
      05 FILLER        PIC X(14)  VALUE '      %   '.
      05 FILLER        PIC X(14)  VALUE '      %   '.
      05 FILLER        PIC X(14)  VALUE '      %   '.
*****
*
*      START OF PROGRAM
*
*****
PROCEDURE DIVISION.
OPEN-FILES.
    OPEN INPUT CARD-IN, COST-MATRIX-FILE,
           OUTPUT PRINT-OUT.
READ-TITLE-CARD.
    READ CARD-IN AT END GO TO TITLE-PARAGRAPH.
    PERFORM FILL-HEADINGS.
READ-WATER-CARD.
    READ CARD-IN AT END GO TO TITLE-PARAGRAPH.
TITLE-PARAGRAPH.

```

```

    PERFORM TOP-NEW-PAGE.
    PERFORM WRITE-PAGE-HEADING.
HOUSEKEEPING.
    MOVE ZEROS TO MATRIX.
    PERFORM ZERO-SUB-TOTALS THRU ZERO-TOTALS.
SORT-RECORDS.
    OPEN      OUTPUT      SORTED-FILE.
    SORT FILE-SORT ON ASCENDING KEY CC-NOS
    INPUT PROCEDURE READ-SORT-INPUT
        THRU READ-SORT-INPUT-END
    OUTPUT PROCEDURE WRITE-SORT-OUTPUT
        THRU WRITE-SORT-OUTPUT-END.
    PERFORM REWIND-FILE.
TEST-CONFIGURATION.
    IF SYSTEM = 0 THEN GO TO WRITE-MATRIX.
    IF SYSTEM = 1 THEN GO TO SERIES-CONFIG.
    IF SYSTEM = 2 THEN GO TO PARA-CONFIG.
    IF SYSTEM = 3 THEN GO TO FAR-SER-CONFIG.
    IF SYSTEM = 4 THEN GO TO SER-PAR-SER-CONFIG.
    GO TO WRITE-MATRIX.
SERIES-CONFIG.
    MOVE ZERO TO L.
    MOVE SERIES TO ZONES.
    GO TO READ-SORTED-FILE.
PARA-CONFIG.
    MOVE ZERO TO L.
    MOVE PARALLEL TO ZONES.
READ-SORTED-FILE.
    READ SORTED-FILE AT END GO TO END-SORTED-FILE.
    IF PAGE-CT EQUAL 2 THEN MOVE AMT-1 TO DOLS
        ELSE MOVE AMT-3 TO DOLS.
    IF DOLS = 0 GO TO READ-SORTED-FILE.
    MOVE CC-NO-1 TO I.
    IF CC-NO-1 NOT EQUAL '3' THEN GO TO STEP-2.
    MOVE CC-NO-2 TO L.
    SUBTRACT 1 FROM L.
STEP-2.
    ADD L TO I.
    IF I > 10 THEN MOVE 10 TO I.
    MOVE CC-NO-4 TO J.
    IF CC-NO-1 NOT EQUAL '3' THEN ADD DOLS TO POSIT (I, 6)
        ADD DOLS TO POSIT (10, 6).
    IF WU-NAME EQUAL 'KE' PERFORM KENTON-CO-EXCEPTIONS
        THRU KENTON-END.
    IF WU-NAME EQUAL 'CO' PERFORM COVINGTON-EXCEPTIONS
        THRU COVINGTON-END.
    IF CC-NO-1 = '3' THEN ADD DOLS TO POSIT (I, J).
    IF CC-NO-1 = '3' THEN ADD DOLS TO POSIT (10, J).
    ADD DOLS TO POSIT (I, 7).
    ADD DOLS TO POSIT (10, 7).

```

GO TO READ-SORTED-FILE.
 END-SORTED-FILE.
 ADD DOLS TO POSIT (10, 7).
 ADD DOLS TO POSIT (9, 7).
 MOVE 3 TO I.
 PERFORM CALCULATE-COSTS THRU END-COSTS.
 IF WU-NAME EQUAL 'CO' THEN GO TO STEP-3.
 MOVE 6 TO I.
 PERFORM CALCULATE-COSTS THRU END-COSTS.
 STEP-3.
 MOVE 0 TO J.
 PERFORM ADD-DEL-COSTS 5 TIMES.
 PERFORM ZERO-SUB-TOTALS THRU ZERO-TOTALS.
 GO TO WRITE-MATRIX.
 ADD-DEL-COSTS.
 ADD 1 TO J.
 MOVE POSIT (3, J) TO POSIT (10, J).
 ADD POSIT (4, J), POSIT (5, J), POSIT (6, J)
 TO POSIT (10, J).
 KENTON-CO-EXCEPTIONS.
 IF CC-NO-2 EQUAL '3' THEN GO TO STORAGE-CHANGES.
 IF CC-NO-2 EQUAL '4' THEN GO TO PUMPING-CHANGES.
 GO TO KENTON-END.
 STORAGE-CHANGES.
 IF CC-NU-4 = '5' OR '6' THEN MOVE 3 TO J
 ELSE MOVE 2 TO J.
 IF CC-NU-4 = '1' THEN MOVE 1 TO J.
 GO TO KENTON-END.
 PUMPING-CHANGES.
 IF CC-NU-4 > '4' THEN MOVE 3 TO J
 ELSE MOVE 2 TO J.
 IF CC-NO-4 = '1' THEN MOVE 1 TO J.
 KENTON-END.
 IF J > 7 THEN MOVE 7 TO J.
 COVINGTON-EXCEPTIONS.
 IF CC-NO-2 EQUAL '4' THEN GO TO COV-PUMP-CHANGES.
 GO TO COVINGTON-END.
 COV-PUMP-CHANGES.
 IF CC-NO-4 > '2' THEN SUBTRACT 1 FROM J.
 IF CC-NO-4 = '2' THEN MOVE 5 TO J.
 COVINGTON-END.
 IF J > 7 THEN MOVE 7 TO J.
 CALCULATE-COSTS.
 IF WATER-SOLD = 0 THEN GO TO END-COSTS.
 MOVE POSIT (I, 1) TO S-TOT (1).
 IF SYSTEM EQUAL 2 THEN MOVE ZERG TO POSIT (I, 1)
 GO TO CALC-PARA-COSTS.
 COMPUTE POSIT (I, 1) = POSIT (I, 1) * MIL-GAL (1)
 / WATER-SOLD.
 MOVE POSIT (I, 2) TO S-TOT (2).

```

COMPUTE POSIT (I, 2) = (POSIT (I, 2) + S-TOT (1)
    - POSIT (I, 1)) * MIL-GAL (2) / (WATER-SOLD
    - MIL-GAL (1)).
SUBTRACT POSIT (I, 1), POSIT (I, 2) FROM POSIT (I, 7)
    GIVING PUSIT (I, 3).
GO TO END-COSTS.
CALC-PARA-COSTS.
    COMPUTE S-TOT (N) = S-TOT (1) * MIL-GAL (N) / WATER-SOLD.
    ADD S-TOT (N) TO POSIT (I, N).
    ADD 1 TO N.
    IF N NOT GREATER THAN PARALLEL GO TO CALC-PARA-COSTS.
    MOVE 1 TO N.
    SUBTRACT POSIT (I, 2), POSIT (I, 3), POSIT (I, 4),
        POSIT (I, 5) FROM PUSIT (I, 7) GIVING PUSIT (I, 1).
END-COSTS.
    EXIT.
NEXT-PAGE.
    IF PAGE-CT > 2 THEN GO TO CLOSE-FILES.
    PERFORM READ-WATER-CARD THRU HOUSEKEEPING.
    MOVE '1' TO RPT-FLAG.
    PERFORM REWIND-FILE.
    GO TO TEST-CONFIGURATION.
PAR-SER-CONFIG.
    GO TO WRITE-MATRIX.
SER-PAR-SER-CONFIG.
    GO TO WRITE-MATRIX.
WRITE-MATRIX.
    PERFORM WRITE-MATRIX-LINE VARYING I FROM 1 BY 1 UNTIL
        I GREATER THAN 10.
    IF RPT-B THEN GO TO NEXT-PAGE
        ELSE PERFORM SKIP-LINE 2 TIMES.
    GO TO WRITE-COSTS-BY-ZONE.
WRITE-MATRIX-LINE.
    MOVE CC-NAME (I) TO NAME-OUT.
    PERFORM MATRIX-COL VARYING J FROM 1 BY 1 UNTIL
        J GREATER THAN 7.
    PERFORM WRITE-DATA-LINE-A.
    IF RPT-B      THEN
        MOVE POSIT (I, 7) TO TEMP-1
        MOVE POSIT (10, 7) TO TEMP-2
        MOVE 7 TO J
        PERFORM VARI-AND-PERCENT
        PERFORM WRITE-DATA-LINE-C
        PERFORM WRITE-DATA-LINE-B
        PERFORM WRITE-SUB-TOTALS THRU ST-END.
    IF I = 2 THEN MOVE CC-NAME (11) TO PRINT-LINE
        PERFORM PRINT-LINE-OUT.
MATRIX-COL.
    MOVE POSIT (I, J) TO COLS (J).
    IF RPT-B PERFORM PERCENT-LINE.

```

```

PERCENT-LINE.
  MOVE C-TOT (J)      TO TEMP-2.
  MOVE POSIT (I, J) TO TEMP-1.
  PERFORM VARI-AND-PERCENT.

WRITE-SUB-TOTALS.
  IF I < 6 OR I = 7 THEN GO TO ST-END.
  IF I > 8 THEN GO TO ST-END.
  MOVE 'SUB TOTAL'   TO NAME-OUT.
  MOVE 0 TO J.
  PERFORM LOAD-SUB-TOTALS 7 TIMES.
  PERFORM WRITE-DATA-LINE-A THRU WRITE-DATA-LINE-B.

ST-END.
  EXIT.

LOAD-SUB-TOTALS.
  ADD 1 TO J.
  MOVE C-TOT (J) TO TEMP-2.
  IF I = 6 THEN MOVE SS-TOT (J) TO COLS (J), TEMP-1
    ELSE MOVE S-TOT (J) TO COLS (J), TEMP-1.
  PERFORM VARI-AND-PERCENT.

WRITE-CUSTS-BY-ZONE.
  PERFORM ZERO-SUB-TOTALS THRU ZERO-TOTALS.
  MOVE ZEROS TO TEMP-1, TEMP-2.
  MOVE 'TOTAL CUSTS BY ZONE:' TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE 'DIRECT' TO NAME-OUT.
  MOVE 0 TO J.
  PERFORM DIRECT-COSTS 5 TIMES.
  MOVE ZERO TO COLS (6).
  MOVE C-TOT (7) TO COLS (7).
  PERFORM WRITE-DATA-LINE-A.
  MOVE 'INDIRECT' TO NAME-OUT.
  MOVE 0 TO J.
  ADD POSIT (1, 7), POSIT (2, 7), POSIT (7, 7), POSIT (8, 7),
    POSIT (9, 7) TO TEMP-1.
  MOVE TEMP-1 TO COLS (7).
  ADD TEMP-1 TO C-TOT (7).
  PERFORM INDIRECT-COSTS 5 TIMES.
  PERFORM WRITE-DATA-LINE-A.
  MOVE 'TOTAL' TO NAME-OUT.
  MOVE 0 TO J.
  PERFORM LOAD-TOTALS 7 TIMES.
  PERFORM WRITE-DATA-LINE-A.
  MOVE ZERO TO K.
  PERFORM SKIP-LINE 2 TIMES.
  GO TO WRITE-WATER-PRODUCED.

DIRECT-COSTS.
  ADD 1 TO J.
  MOVE POSIT (10, J) TO COLS (J).
  ADD POSIT (10, J) TO C-TOT (J).
  ADD POSIT (10, J) TO C-TOT (7).

```

INDIRECT-COSTS.

ADD 1 TO J.

MOVE POSIT (10, 6) TO TEMP-1.

MULTIPLY MIL-GAL (J) BY TEMP-1.

IF WATER-SOLD > 0 THEN

DIVIDE TEMP-1 BY WATER-SOLD GIVING TEMP-1

ADD .005 TO TEMP-1

ELSE MOVE ZERO TO TEMP-1.

IF TEMP-1 < 0 THEN SUBTRACT .01 FROM TEMP-1.

MOVE TEMP-1 TO COLS (J).

ADD TEMP-1 TO C-TOT (J).

IF J = 1 THEN ADD .01 TO C-TOT (1).

IF WU-NAME = 'KE' AND J = 3 THEN

COMPUTE S-TOT (2) = POSIT (10, 6) - S-TOT (1)

MOVE S-TOT (2) TO COLS (3)

COMPUTE C-TOT (3) = C-TOT (7) - (C-TOT (1) +
C-TOT (2)).

ADD TEMP-1 TO S-TOT (1).

LOAD-TOTALS.

ADD 1 TO J.

MOVE C-TOT (J) TO COLS (J).

WRITE-WATER-PRODUCED.

MOVE 'RE' TO LTR-OUT.

MOVE 'VENUE PRODUCING WATER (MG)' TO NAME-OUT.

MOVE ZERO TO J.

ADD .005 TO WATER-SOLD.

MOVE WATER-SOLD TO COLS (7), TEMP-2.

MOVE 100 TO PERC (7).

PERFORM LOAD-WATER 6 TIMES.

PERFORM WRITE-DATA-LINE-A.

PERFORM WRITE-DATA-LINE-C.

PERFORM WRITE-DATA-LINE-B.

MOVE '' TO LTR-OUT.

MOVE ZERO TO J.

PERFORM ZERO-SUB-TOTALS THRU ZERO-TOTALS.

PERFORM SKIP-LINE 2 TIMES.

GO TO CALCULATE-UNIT-COSTS.

LOAD-WATER.

ADD 1 TO J.

ADD .005 TO MIL-GAL (J).

MOVE MIL-GAL (J) TO COLS (J), TEMP-1.

PERFORM VARI-AND-PERCENT.

CALCULATE-UNIT-COSTS.

MOVE 'UNIT COSTS BY ZONE:' TO PRINT-LINE.

PERFORM PRINT-LINE-OUT.

MOVE ZEROS TO POSIT (1, 6), POSIT (2, 6), POSIT (7, 6),
POSIT (8, 6), POSIT (9, 6), POSIT (10, 6).

IF ZONES > 7 THEN MOVE 7 TO ZONES.

PERFORM LOAD-UNIT-COSTS VARYING I FROM 1 BY 1 UNTIL
I GREATER THAN 10.

```

MOVE '2' TO RPT-FLAG.
GO TO WRITE-MATRIX.
LOAD-UNIT-COSTS.
  IF WATER-SOLD > 0 THEN DIVIDE PUSIT (I, 7) BY WATER-SOLD
    GIVING TEMP-1
    ADD .005 TO TEMP-1
    ELSE MOVE ZERO TO TEMP-1.
  MOVE TEMP-1 TO POSIT (I, 7).
  IF I < 7 ADD TEMP-1 TO SS-TOT (7).
  IF I < 9 ADD TEMP-1 TO S-TOT (7).
  IF I < 10 THEN ADD TEMP-1 TO C-TOT (7)
    ELSE MOVE C-TOT (7) TO POSIT (10, 7).
  PERFORM LOAD-COLS THRU LOAD-CONTD VARYING J FROM 1 BY 1 UNTIL
    J GREATER THAN ZONES.

LOAD-COLS.
  IF I < 3 OR > 6 THEN GO TO LOAD-CONTD.
  IF MIL-GAL (J) > 0 THEN DIVIDE POSIT (I, J) BY
    MIL-GAL (J) GIVING TEMP-1
    ADD .005 TO TEMP-1
    ELSE MOVE ZERO TO TEMP-1.

LOAD-CONTD.
  MOVE TEMP-1 TO POSIT (I, J).
  IF I < 7 THEN ADD TEMP-1 TO SS-TOT (J).
  IF I < 9 THEN ADD TEMP-1 TO S-TOT (J).
  IF I < 10 THEN ADD TEMP-1 TO C-TOT (J)
    ELSE MOVE C-TOT (J) TO POSIT (10, J).

*****
*
*
*          GENERAL SUBROUTINES
*
*****
MOVE-SPACES.
  MOVE SPACES TO PRINT-LINE.
PRINT-LINE-OUT.
  WRITE PRINT-DATA BEFORE 1.
  ADD 1 TO LINE-CT.
SKIP-LINE.
  PERFORM MOVE-SPACES.
  PERFORM PRINT-LINE-OUT.
PAGE-OVFLOW.
  IF LINE-CT NOT < MAX-LINE THEN
    PERFORM TOP-NEW-PAGE.
    PERFORM WRITE-PAGE-HEADING.
ZERO-SUB-TOTALS.
  MOVE ZEROS TO SUB-SUB-TOTAL.
  MOVE ZEROS TO SUB-TOTAL.
ZERO-TOTALS.
  MOVE ZEROS TO COL-TOTAL.
TOP-NEW-PAGE.
  PERFORM MOVE-SPACES.

```

```

WRITE PRINT-DATA BEFORE NEW-PAGE.
MOVE PAGE-CT TO PAGE-OUT.
MOVE 0 TO LINE-CT.
ADD 1 TU PAGE-CT.
FILL-HEADINGS.
  MOVE DATE-IN TO DATE-OUT.
  MOVE SPACES TO Z-4, Z-5.
  MOVE 'ZONES 1..3' TO Z-OUT.
  IF WU-NAME EQUAL 'CO' THEN
    MOVE '          ZONE 4' TO Z-4
    MOVE '          ZONE 5' TO Z-5
    MOVE 'ZONES 1..5' TO Z-OUT.
  MOVE FISCAL-YR TO FISC-YR.
  MOVE 'MONTHLY' TO RPT-PD.
  IF RPT-PERIOD = '2' THEN MOVE 'QUARTER' TO RPT-PD.
  IF RPT-PERIOD = '3' THEN MOVE 'HALF YR' TO RPT-PD.
  IF RPT-PERIOD = '4' THEN MOVE 'ANNUAL' TO RPT-PD.
  MOVE SPACES TO WKS-OUT.
  IF NO-WKS = '1' MOVE '( 4 WKS)' TO WKS-OUT.
  IF NO-WKS = '2' MOVE '( 5 WKS)' TO WKS-OUT.
  IF NO-WKS = '3' MOVE '(13 WKS)' TO WKS-OUT.
  IF NO-WKS = '4' MOVE '(26 WKS)' TO WKS-OUT.
  IF NO-WKS = '5' MOVE '(52 WKS)' TO WKS-OUT.
  MOVE PAGE-CT TO PAGE-OUT.
  MOVE TITLE-IN TO TITLE-OUT.
WRITE-PAGE-HEADING.
  MOVE HEADING-1 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-2 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-3 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-6 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  PERFORM SKIP-LINE 2 TIMES.
  MOVE HEADING-4 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
  MOVE HEADING-5 TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT;
  PERFORM SKIP-LINE.
  IF PAGE-CT EQUAL 2 THEN MOVE 'FUNCTION COSTS FOR PERIOD:' TO PRINT-LINE
    ELSE MOVE 'FUNCTION COSTS YEAR TO DATE:' TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
WRITE-DATA-LINE-A.
  MOVE DATA-LINE-A TO PRINT-LINE.
  PERFORM PRINT-LINE-OUT.
WRITE-DATA-LINE-C.
  MOVE DATA-LINE-C TO PRINT-LINE.

```

```

        WRITE PRINT-DATA BEFORE O.
WRITE-DATA-LINE-B.
        MOVE DATA-LINE-B TO PRINT-LINE.
        PERFORM PRINT-LINE-OUT.
VARI-AND-PERCENT.
        MULTIPLY 100 BY TEMP-1.
        IF TEMP-2 > 0 DIVIDE TEMP-2 INTO TEMP-1 ELSE
                MOVE ZERO TO TEMP-1.
        MULTIPLY 1    BY TEMP-1 GIVING PERC (J) ROUNDED.
*****
*
*          FINI
*
*****
READ-SORT-INPUT.
        READ COST-MATRIX-FILE AT END
        GO TO READ-SORT-INPUT-END.
        IF CC-NU EQUAL '4301' OR '4302' OR '4401'
            OR '4321' OR '4322' OR '4421'
                GO TO READ-SORT-INPUT.
        RELEASE SORT-INPUT FROM CM-REC.
        GO TO READ-SORT-INPUT.
READ-SORT-INPUT-END.
        EXIT.
WRITE-SORT-OUTPUT.
        RETURN FILE-SORT RECORD INTO INPUT-REC
        AT END GO TO WRITE-SORT-OUTPUT-END.
        WRITE INPUT-REC.
        GO TO WRITE-SORT-OUTPUT.
WRITE-SORT-OUTPUT-END.
        EXIT.
REWIND-FILE.
        CLOSE SORTED-FILE.
        OPEN INPUT SURTED-FILE.
CLOSE-FILES.
        CLOSE CARD-IN, COST-MATRIX-FILE,
        SORTED-FILE, PRINT-OUT.
E-O-J.
        STOP RUN.

```

PGMNAME=WUASBUCM
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASBUCM.
AUTHOR. ACT SYSTEMS, INC
 SUITE 200
 807 W MORSE BLVD
 WINTER PARK, FL 32789.
DATE-WRITTEN. JAN 1978
DATE-COMPILED.
REMARKS. THIS PROGRAM BACKS UP THE CM FILE
 AND PRODUCES A COPY OF SAME.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
 C01 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
 SELECT CARD-IN
 ASSIGN TO UK-2540R-S-CARDIN
 ACCESS MODE IS SEQUENTIAL.
 SELECT PRINT-OUT
 ASSIGN TO UR-1403-S-PRINTOUT
 ACCESS MODE IS SEQUENTIAL.
 SELECT COST-MATRIX-FILE
 ASSIGN TO DA-3330-I-CMFILE
 ACCESS MODE IS SEQUENTIAL
 RECORD KEY IS CM-NUMBER.
 SELECT TAPE-OUT
 ASSIGN TO UT-2400-S-TAPEOUT
 ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD COST-MATRIX-FILE
 BLOCK CONTAINS 5 RECORDS
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS INPUT-REC.
01 INPUT-REC.
 05 MATCH-CODE-IN.
 10 DELETE-POS PIC X.
 10 FILLER PIC X(3).
 05 CM-NUMBER.
 10 ACCT-NO-IN PIC X(10).
 10 COST-CNTR-IN PIC X(4).
 10 COST-CAT-IN PIC X(2).
 05 AMOUNTS-IN OCCURS 6 TIMES.
 10 AMT-IN PIC S9(8)V99.

FD TAPE-OUT
 BLOCK CONTAINS 5 RECORDS
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE STANDARD
 DATA RECORD IS TAPE-REC.
 01 TAPE-REC PIC X(80).
 FD PRINT-OUT
 RECORD CONTAINS 133 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS PRINT-LINE.
 01 PRINT-LINE.
 05 FILLER PIC X.
 05 PRINT-DATA PIC X(132).
 FD CARD-IN
 RECORD CONTAINS 80 CHARACTERS
 LABEL RECORDS ARE OMITTED
 DATA RECORD IS INPUT-CARD.
 01 INPUT-CARD.
 05 DATE-IN.
 10 BACKUP-OPTION PIC X(7).
 10 FILLER PIC X(11).
 05 TITLE-IN PIC X(24).
 05 FILLER PIC X(38).
 WORKING-STORAGE SECTION.
 77 LINE-CT PIC 99 COMP VALUE 0.
 77 PAGE-CT PIC 9(4) COMP VALUE 0.
 77 MAX-LINE PIC 99 COMP VALUE 56.
 77 I PIC 9.
 77 BACKUP-FLAG PIC 9 VALUE 0.
 77 ACCT-HOLD PIC X(10) VALUE SPACES.
 77 ACCT-COUNT PIC 999 VALUE ZERO.
 01 HEAD-1.
 05 FILLER PIC X(53) VALUE SPACES.
 05 FILLER PIC X(27) VALUE 'COST ALLOCATION FILE BA
 CKUP'.
 -
 05 FILLER PIC X(52) VALUE SPACES.
 01 HEAD-2.
 05 FILLER PIC X(13) VALUE 'DATE ENDING: '.
 05 DATE-OUT PIC X(18).
 05 FILLER PIC X(34) VALUE SPACES.
 05 FILLER PIC X(3) VALUE 'FOR'.
 05 FILLER PIC X(54) VALUE SPACES.
 05 FILLER PIC X(6) VALUE 'PAGE: '.
 05 PAGE-OUT PIC Z(4).
 01 HEAD-3.
 05 FILLER PIC X(55) VALUE SPACES.
 05 TITLE-OUT PIC X(24).
 05 FILLER PIC X(53) VALUE SPACES.
 01 HEAD-4.
 05 FILLER PIC X(2) VALUE SPACES.

```

05 FILLER          PIC X(35) VALUE '***** K E Y   F I E
                   'L D *****'.
05 FILLER          PIC X(28) VALUE SPACES.
05 FILLER          PIC X(7)  VALUE 'CURRENT'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(8)  VALUE 'PREVIOUS'.
05 FILLER          PIC X(5)  VALUE SPACES.
05 FILLER          PIC X(7)  VALUE 'CURRENT'.
05 FILLER          PIC X(5)  VALUE SPACES.
05 FILLER          PIC X(7)  VALUE 'YEARLY '.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(7)  VALUE 'CURRENT'.
05 FILLER          PIC X(5)  VALUE SPACES.
05 FILLER          PIC X(8)  VALUE 'PREVIOUS'.

01 HEAD-5.
05 FILLER          PIC X(2)  VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'ACCOUNT NO'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(9)  VALUE 'COST CNTR'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(8)  VALUE 'COST CAT'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'MATCH CODE'.
05 FILLER          PIC X(16) VALUE SPACES.
05 FILLER          PIC X(4)  VALUE 'COST'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'MONTH COST'.
05 FILLER          PIC X(2)  VALUE SPACES.
05 FILLER          PIC X(10) VALUE 'COST (YTD)'.
05 FILLER          PIC X(4)  VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'BUDGET'.
05 FILLER          PIC X(2)  VALUE SPACES.
05 FILLER          PIC X(12) VALUE 'BUDGET (YTD)'.
05 FILLER          PIC X(2)  VALUE SPACES.
05 FILLER          PIC X(9)  VALUE 'YEAR COST'.

01 DATA-LINE.
05 FILLER          PIC X(2).

05 CM-NUMBER-1.
10 ACCT-NO         PIC X(10).
10 FILLER          PIC X(6).
10 COST-CNTR       PIC X(4).
10 FILLER          PIC X(10).
10 COST-CAT        PIC X(2).
05 FILLER          PIC X(10).
05 MATCH-CODE      PIC X(4).
05 FILLER          PIC X(12).
05 AMOUNTS OCCURS 6 TIMES.
10 AMT             PIC -(9)---.

01 ACCOUNT-ACCUM.
05 ACCT-ACCUM      PIC S9(8)V99 OCCURS 6 TIMES.

```

```

01  SUBTOTAL.
    05  SUBS          PIC S9(8)V99 OCCURS 6 TIMES.
PROCEDURE DIVISION.
OPEN-FILES.
    OPEN INPUT COST-MATRIX-FILE,
           CARD-IN,
           OUTPUT PRINT-OUT.
FILL-HEADING.
    READ CARD-IN AT END
        PERFORM ERROR-CARD-IN
        GO TO CLOSE-FILES.
    MOVE DATE-IN TO DATE-OUT.
    MOVE TITLE-IN TO TITLE-OUT.
    READ CARD-IN AT END
        GO TO HOUSEKEEP.
    IF BACKUP-OPTION NOT = 'BACK-UP' THEN
        GO TO HOUSEKEEP.
OPEN-BACKUP-FILE.
    MOVE 1 TO BACKUP-FLAG.
    OPEN OUTPUT TAPE-OUT.
HOUSEKEEP.
    MOVE ZEROS TO SUBTOTAL, ACCOUNT-ACCUM.
WRITE-HEADINGS.
    ADD 1 TO PAGE-CT.
    MOVE PAGE-CT TO PAGE-OUT.
    MOVE 0 TO LINE-CT.
    PERFORM MOVE-SPACES.
    WRITE PRINT-LINE BEFORE NEW-PAGE.
    MOVE HEAD-1 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-2 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-3 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-4 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE HEAD-5 TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
INIT-ACCT-HOLD.
    PERFORM READ-INPUT-REC.
    MOVE ACCT-NO-IN TO ACCT-HOLD.
    GO TO START-BACKUP.
READ-INPUT-REC.
    READ COST-MATRIX-FILE AT END
        GO TO WRITE-SUBTOTAL.
START-BACKUP.
    IF ACCT-NO-IN NOT = ACCT-HOLD

```

```

THEN PERFORM SUBTOTAL-ACCT.
ADD 1 TO ACCT-COUNT.
MOVE ACCT-NO-IN TO ACCT-NO.
MOVE COST-CNTR-IN TO COST-CNTR.
MOVE COST-CAT-IN TO COST-CAT.
MOVE MATCH-CODE-IN TO MATCH-CODE.
PERFORM MOVE-AMOUNTS VARYING I FROM 1 BY 1 UNTIL I = 7.
MOVE DATA-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
IF LINE-CT > MAX-LINE
    THEN PERFORM WRITE-HEADINGS.
IF BACKUP-FLAG = 1 THEN
    MOVE INPUT-REC TO TAPE-REC
    WRITE TAPE-REC.
GO TO READ-INPUT-REC.
MOVE-SPACES.
MOVE SPACES TO PRINT-LINE.
PRINT-LINE-CUT.
WRITE PRINT-LINE BEFORE 1.
ADD 1 TO LINE-CT.
MOVE-AMOUNTS.
MOVE AMT-IN (I) TO AMT (I).
ADD AMT-IN (I) TO SUBS (I) ACCT-ACCUM (I).
MOVE-SUBTOTALS.
MOVE SUBS (I) TO AMT (I).
ERROR-CARD-IN.
MOVE '*** NO DATE/TITLE CARD *** JOB ABORTED'
    TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
SUBTOTAL-ACCT.
MOVE SPACES TO DATA-LINE.
PERFORM MOVE-ACCT-SUB VARYING I FROM 1 BY 1 UNTIL I = 7.
IF (ACCT-COUNT > 1) AND (ACCOUNT-ACCUM NOT = ZERO) THEN
    MOVE 'SUB TOTAL' TO ACCT-NO.
    MOVE DATA-LINE TO PRINT-DATA,
    PERFORM PRINT-LINE-OUT.
MOVE 1 TO ACCT-COUNT.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
MOVE ZEROS TO ACCOUNT-ACCUM.
MOVE ACCT-NO-IN TO ACCT-HOLD.
MOVE-ACCT-SUB.
MOVE ACCT-ACCUM (I) TO AMT (I).
WRITE-SUBTOTAL.
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT 2 TIMES.
MOVE SPACES TO DATA-LINE.
PERFORM MOVE-SUBTOTALS VARYING I FROM 1 BY 1 UNTIL I = 7.
MOVE 'TOTAL' TO ACCT-NO.
MOVE DATA-LINE TO PRINT-DATA.
PERFORM PRINT-LINE-OUT.
CLOSE-FILES.

```

```
CLOSE    COST-MATRIX-FILE,  
        CARD-IN,  
        PRINT-OUT.  
CLOSE-TAPE-FILE.  
    IF BACKUP-FLAG NOT = 1 THEN  
        GO TO E-O-J.  
    CLOSE TAPE-OUT.  
E-O-J.  
    STOP RUN.
```

```
PGMNAME=WUASUCAD
.SYSIN DD *
ID DIVISION.
PROGRAM-ID. WUASUCAD.
AUTHOR. ACC SYSTEMS, INC
    SUITE 200
    807 W MORSE BLVD
    WINTER PARK, FL 32789.
DATE-WRITTEN. MARCH 1978.
DATE-COMPILED.
REMARKS. THIS PROGRAM READS A CARD IMAGE OF A
COST MATRIX RECORD AND EITHER CHANGES,
ADDS OR DELETES THE RECORD.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
SPECIAL-NAMES.
    CO1 IS NEW-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-IN
        ASSIGN TO UR-2540R-S-CARDIN
        ACCESS MODE IS SEQUENTIAL.
    SELECT PRINT-OUT
        ASSIGN TO UR-1403-S-PRINTOUT
        ACCESS MODE IS SEQUENTIAL.
    SELECT COST-MATRIX-FILE
        ASSIGN TO DA-3330-I-CMFILE
        ACCESS MODE IS RANDOM
        RECORD KEY IS CM-NUMBER
        NOMINAL KEY IS CM-NOMINAL.
    SELECT TAPE-IN
        ASSIGN TO UT-2400-S-TAPEIN
        ACCESS MODE IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD COST-MATRIX-FILE
    BLOCK CONTAINS 5 RECORDS
    RECORD CONTAINS 80 CHARACTERS
    LABEL RECORDS ARE STANDARD
    DATA RECORD IS CM-REC.
01 CM-REC.
    05 MATCH-CODE.
        10 DELETE-POS    PIC X.
        10 FILLER       PIC X(3).
    05 CM-NUMBER.
        10 ACCT-NO      PIC X(10).
        10 COST-CNTR     PIC X(4).
        10 COST-CAT      PIC X(2).
```

```

05 AMOUNTS.
  10 AMTS          PIC S9(8)V99 OCCURS 6 TIMES.
FD TAPE-IN
BLOCK CONTAINS 5 RECORDS
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE STANDARD
DATA RECORD IS TAPE-REC.
01 TAPE-REC          PIC X(80).
FD CARD-IN
RECORD CONTAINS 80 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS INPUT-REC.
01 INPUT-REC.
  05 DATE-IN          PIC X(18).
  05 TITLE-IN         PIC X(24).
  05 FILLER           PIC X(38).
01 INPUT-REC-1.
  05 MATCH-CODE-IN   PIC X(4).
  05 INPUT-KEY.
    10 ACCT-IN         PIC X(10).
    10 COST-CNTR-IN   PIC X(4).
    10 COST-CAT-IN    PIC X(2).
  05 AMOUNTS-IN       PIC X(60).
01 INPUT-REC-2.
  05 FILLER           PIC X(20).
  05 AMTS-IN          PIC S9(8)V99 OCCURS 6 TIMES.
FD PRINT-OUT
RECORD CONTAINS 133 CHARACTERS
LABEL RECORDS ARE OMITTED
DATA RECORD IS PRINT-LINE.
01 PRINT-LINE.
  05 FILLER           PIC X.
  05 PRINT-DATA        PIC X(132).

WORKING-STORAGE SECTION.
77 PAGE-CT            PIC 9(4) COMP VALUE 0.
77 LINE-CT             PIC 9(2) COMP VALUE 0.
77 MAX-LINE            PIC 9(2) COMP VALUE 56.
77 INPUT-CT            PIC 9(4)V99 VALUE 0.
77 ADD-CT              PIC 9(4)V99 VALUE 0.
77 CHANGE-CT           PIC 9(4)V99 VALUE 0.
77 DELETE-CT           PIC 9(4)V99 VALUE 0.
77 ERROR-CT            PIC 9(4)V99 VALUE 0.
77 CM-NOMINAL          PIC X(16) VALUE SPACES.
77 I                   PIC 99   COMP VALUE 0.
77 UPDATE-FLAG          PIC 9 VALUE 0.
77 INPUT-DEVICE         PIC 9 VALUE 0.
01 HEAD-1.
  05 FILLER            PIC X(54) VALUE SPACES.
  05 FILLER            PIC X(24) VALUE 'CHANGES ADDS AND DELETE
                                'S'.
-
```

```

05 FILLER          PIC X(54) VALUE SPACES.
01 HEAD-2.
05 FILLER          PIC X(6) VALUE 'DATE: '.
05 DATE-OUT        PIC X(18).
05 FILLER          PIC X(41) VALUE SPACES.
05 FILLER          PIC X(2) VALUE 'ON'.
05 FILLER          PIC X(55) VALUE SPACES.
05 FILLER          PIC X(6) VALUE 'PAGE: '.
05 PAGE-OUT        PIC Z(4).

01 HEAD-3.
05 FILLER          PIC X(54) VALUE SPACES.
05 TITLE-OUT        PIC X(24).
05 FILLER          PIC X(54) VALUE SPACES.

01 HEAD-4.
05 FILLER          PIC X(58) VALUE SPACES.
05 FILLER          PIC X(16) VALUE 'COST MATRIX FILE'.
05 FILLER          PIC X(58) VALUE SPACES.

01 HEAD-5.
05 FILLER          PIC X(7) VALUE 'ACCOUNT'.
05 FILLER          PIC X(6) VALUE SPACES.
05 FILLER          PIC X(4) VALUE 'COST'.
05 FILLER          PIC X(3) VALUE SPACES.
05 FILLER          PIC X(4) VALUE 'COST'.
05 FILLER          PIC X(2) VALUE SPACES.
05 FILLER          PIC X(5) VALUE 'MATCH'.
05 FILLER          PIC X(6) VALUE SPACES.
05 FILLER          PIC X(7) VALUE 'CURRENT'.
05 FILLER          PIC X(5) VALUE SPACES.
05 FILLER          PIC X(8) VALUE 'PREVIOUS'.
05 FILLER          PIC X(9) VALUE SPACES.
05 FILLER          PIC X(3) VALUE 'YTD'.
05 FILLER          PIC X(8) VALUE SPACES.
05 FILLER          PIC X(6) VALUE 'YEARLY'.
05 FILLER          PIC X(9) VALUE SPACES.
05 FILLER          PIC X(3) VALUE 'YTD'.
05 FILLER          PIC X(6) VALUE SPACES.
05 FILLER          PIC X(8) VALUE 'PREVIOUS'.
05 FILLER          PIC X(7) VALUE SPACES.
05 FILLER          PIC X(8) VALUE 'COMMENTS'.
05 FILLER          PIC X(8) VALUE SPACES.

01 HEAD-6.
05 FILLER          PIC X(7) VALUE ' NUMBER'.
05 FILLER          PIC X(5) VALUE SPACES.
05 FILLER          PIC X(6) VALUE 'CENTER'.
05 FILLER          PIC X(2) VALUE SPACES.
05 FILLER          PIC X(3) VALUE 'CAT'.
05 FILLER          PIC X(3) VALUE SPACES.
05 FILLER          PIC X(4) VALUE 'CODE'.
05 FILLER          PIC X(7) VALUE SPACES.
05 FILLER          PIC X(6) VALUE 'AMOUNT'.

```

```

05 FILLER          PIC X(7)  VALUE SPACES.
05 FILLER          PIC X(5)  VALUE 'MONTH'.
05 FILLER          PIC X(9)  VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'AMOUNT'.
05 FILLER          PIC X(7)  VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'BUDGET'.
05 FILLER          PIC X(7)  VALUE SPACES.
05 FILLER          PIC X(6)  VALUE 'BUDGET'.
05 FILLER          PIC X(8)  VALUE SPACES.
05 FILLER          PIC X(4)  VALUE 'YEAR'.
05 FILLER          PIC X(24) VALUE SPACES.

01 DATA-LINE.
05 ACCT-OUT        PIC X(10).
05 FILLER          PIC X(3).
05 COST-CNTR-OUT   PIC X(4).
05 FILLER          PIC X(4).
05 COST-CAT-OUT    PIC X(2).
05 FILLER          PIC X(3).
05 MATCH-CODE-OUT  PIC X(4).
05 FILLER          PIC X.
05 AMOUNTS-OUT OCCURS 6 TIMES.
 10 CODE-NAME      PIC X(2).
 10 AMTS-OUT       PIC -(8)---.
05 FILLER          PIC X(3).
05 COMMENT         PIC X(20).

PROCEDURE DIVISION.
OPEN-FILES.
  OPEN I-O    COST-MATRIX-FILE,
        INPUT CARD-IN,
        OUTPUT PRINT-OUT.

FILL-HEADING.
  READ CARD-IN AT END GO TO WRITE-RECORD-COUNTS.
  MOVE DATE-IN TO DATE-OUT.
  MOVE TITLE-IN TO TITLE-OUT.

WRITE-HEADINGS.
  ADD 1 TO PAGE-CT.
  MOVE 0 TO LINE-CT.
  MOVE PAGE-CT TO PAGE-OUT.
  PERFORM MOVE-SPACES.
  WRITE PRINT-LINE BEFORE NEW-PAGE.
  MOVE HEAD-1 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD-2 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD-3 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  MOVE HEAD-4 TO PRINT-DATA.
  PERFORM PRINT-LINE-OUT.
  PERFORM MOVE-SPACES.
  PERFORM PRINT-LINE-OUT 2 TIMES.

```

```
MOVE HEAD-5 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
MOVE HEAD-6 TO PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
HOUSEKEEP.  
MOVE SPACES TO DATA-LINE.  
UPDATE-FILE-YES-OR-NO.  
READ CARD-IN AT END GO TO WRITE-RECORD-COUNTS.  
IF ACCT-IN NOT = 'UPDATE' THEN  
    ADD 1 TO INPUT-CT  
    GO TO START-UPDATE.  
MOVE 1 TO UPDATE-FLAG.  
TEST-FOR-INPUT-DEVICE.  
READ CARD-IN AT END  
MOVE 1 TO INPUT-DEVICE  
PERFORM OPEN-TAPE-FILE  
GO TO READ-INPUT.  
*  
*  
* THE ABOVE CONDITION IS FOR TAPE INPUT USED  
* TO RECREATE THE CM FILE FROM THE BACKUP TAPE  
*  
*  
* THE FOLLOWING IS FOR CARD INPUT USED TO  
* MAKE CHANGES, ADDS, OR DELETES TO THE CM-FILE.  
*  
*  
ADD 1 TO INPUT-CT.  
GO TO START-UPDATE.  
READ-INPUT.  
IF INPUT-DEVICE = 0 THEN  
    PERFORM READ-CARD-IN  
ELSE PERFORM READ-TAPE-IN.  
ADD 1 TO INPUT-CT.  
START-UPDATE.  
MOVE ACCT-IN TO ACCT-OUT.  
MOVE COST-CNTR-IN TO COST-CNTR-OUT.  
MOVE COST-CAT-IN TO COST-CAT-OUT.  
IF (MATCH-CODE-IN = SPACES) AND (AMOUNTS-IN = SPACES) THEN  
    GO TO DELETE-RECORD.  
MOVE MATCH-CODE-IN TO MATCH-CODE-OUT.  
EXAMINE AMOUNTS-IN REPLACING ALL ' ' BY '0'.  
PERFORM MOVE-AMTS-IN-TO-AMTS-OUT VARYING I  
    FROM 1 BY 1 UNTIL I = 7.  
VERIFY-KEY.  
IF (ACCT-IN = SPACES) OR (COST-CNTR-IN = SPACES) OR  
    (COST-CAT-IN = SPACES) OR (MATCH-CODE-IN = SPACES)  
THEN MOVE '*** INVALID KEY ***' TO COMMENT  
MOVE DATA-LINE TO PRINT-DATA
```

```

        PERFORM PRINT-LINE-OUT
        PERFORM MOVE-SPACES THRU PRINT-LINE-OUT
        ADD 1 TO ERROR-CT
        GO TO READ-INPUT.

READ-CM-FILE.
    MOVE INPUT-KEY TO CM-NOMINAL.
    READ COST-MATRIX-FILE INVALID KEY
        GO TO ADD-RECORD.

CHANGE-RECORD.
    MOVE '*** NEW RECORD ***' TO COMMENT.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    MOVE MATCH-CODE TO MATCH-CODE-OUT.
    PERFORM MOVE-CM-AMTS-OUT VARYING I
        FROM 1 BY 1 UNTIL I = 7.
    MOVE '*** OLD RECORD ***' TO COMMENT.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    PERFORM PAGE-OVFLOW.
    ADD 1 TO CHANGE-CT.
    MOVE MATCH-CODE-IN TO MATCH-CODE.
    PERFORM MOVE-AMTS-IN-TO-DISK-AMTS VARYING I
        FROM 1 BY 1 UNTIL I = 7.
    IF UPDATE-FLAG = 1 THEN
        REWRITE CM-REC.
    MOVE SPACES TO DATA-LINE.
    GO TO READ-INPUT.

ADD-RECORD.
    MOVE '*** RECORD ADDED ***' TO COMMENT.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.
    PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.
    MOVE ACCT-IN TO ACCT-NO.
    MOVE COST-CNTR-IN TO COST-CNTR.
    MOVE COST-CAT-IN TO COST-CAT.
    MOVE MATCH-CODE-IN TO MATCH-CODE.
    PERFORM MOVE-AMTS-IN-TO-DISK-AMTS VARYING I
        FROM 1 BY 1 UNTIL I = 7.
    MOVE CM-NUMBER TO CM-NOMINAL.
    IF UPDATE-FLAG = 1 THEN
        WRITE CM-REC.
    PERFORM PAGE-OVFLOW.
    ADD 1 TO ADD-CT.
    MOVE SPACES TO DATA-LINE.
    GO TO READ-INPUT.

DELETE-RECORD.
    MOVE '*** RECORD DELETED **' TO COMMENT.
    MOVE DATA-LINE TO PRINT-DATA.
    PERFORM PRINT-LINE-OUT.

```

```
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
ADD 1 TO DELETE-CT.  
PERFORM PAGE-OVFLOW.  
IF UPDATE-FLAG = 0 THEN  
    GO TO READ-INPUT.  
MOVE INPUT-KEY TO CM-NOMINAL.  
READ COST-MATRIX-FILE INVALID KEY  
    GO TO READ-INPUT.  
MOVE ZEROS TO MATCH-CODE, AMOUNTS.  
MOVE HIGH-VALUE TO DELETE-POS.  
REWRITE CM-REC.  
GO TO READ-INPUT.  
PAGE-OVFLOW.  
    IF LINE-CT NOT < MAX-LINE THEN  
        PERFORM WRITE-HEADINGS.  
MOVE-SPACES.  
    MOVE SPACES TO PRINT-LINE.  
PRINT-LINE-OUT.  
    WRITE PRINT-LINE BEFORE 1.  
    ADD 1 TO LINE-CT.  
MOVE-AMTS-IN-TO-AMTS-OUT.  
    MOVE AMTS-IN (I) TO AMTS-OUT OF AMOUNTS-OUT (I).  
MOVE-CM-AMTS-OUT.  
    MOVE AMTS (I) TO AMTS-OUT OF AMOUNTS-OUT (I).  
MOVE-AMTS-IN-TO-DISK-AMTS.  
    MOVE AMTS-IN (I) TO AMTS (I).  
READ-CARD-IN.  
    READ CARD-IN AT END  
    GO TO WRITE-RECORD-COUNTS.  
READ-TAPE-IN.  
    READ TAPE-IN AT END  
    GO TO WRITE-RECORD-COUNTS.  
MOVE TAPE-REC TO INPUT-REC.  
OPEN-TAPE-FILE.  
    OPEN INPUT TAPE-IN.  
WRITE-RECORD-COUNTS.  
    MOVE SPACES TO DATA-LINE.  
    PERFORM MOVE-SPACES.  
    WRITE PRINT-LINE BEFORE NEW-PAGE.  
    MOVE 'I' TO CODE-NAME OF AMOUNTS-OUT (1).  
    MOVE 'A' TO CODE-NAME OF AMOUNTS-OUT (2).  
    MOVE 'C' TO CODE-NAME OF AMOUNTS-OUT (3).  
    MOVE 'D' TO CODE-NAME OF AMOUNTS-OUT (4).  
    MOVE 'E' TO CODE-NAME OF AMOUNTS-OUT (5).  
    MOVE INPUT-CT TO AMTS-OUT OF AMOUNTS-OUT (1).  
    MOVE ADD-CT TO AMTS-OUT OF AMOUNTS-OUT (2).  
    MOVE CHANGE-CT TO AMTS-OUT OF AMOUNTS-OUT (3).  
    MOVE DELETE-CT TO AMTS-OUT OF AMOUNTS-OUT (4).  
    MOVE ERROR-CT TO AMTS-OUT OF AMOUNTS-OUT (5).  
    MOVE DATA-LINE TO PRINT-DATA.
```

```
PERFORM PRINT-LINE-OUT.  
PERFORM MOVE-SPACES THRU PRINT-LINE-OUT.  
IF UPDATE-FLAG = 1 THEN  
  MOVE '*** COST MARTIX FILE UPDATED ***' TO PRINT-DATA  
ELSE MOVE '*** COST MATRIX FILE NOT UPDATED ***' TO  
  PRINT-DATA.  
PERFORM PRINT-LINE-OUT.  
CLOSE-TAPE-FILE.  
IF INPUT-DEVICE = 0 THEN  
  GO TO CLOSE-FILES.  
CLOSE TAPE-IN.  
CLOSE-FILES.  
CLOSE  COST-MATRIX-FILE,  
      CARD-IN,  
      PRINT-OUT.  
E-O-J.  
STOP RUN.
```

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completing)

1. REPORT NO. EPA-600/2-80-012b	2.	3. RECIPIENT'S ACCESSION NO.
4. TITLE AND SUBTITLE DEVELOPMENT AND APPLICATION OF A WATER SUPPLY COST ANALYSIS SYSTEM Volume II		5. REPORT DATE July 1980 (Issuing Date) 6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) James I. Gillean, Rex D. Osborn, William L. Britton, Jr. Robert M. Clark		8. PERFORMING ORGANIZATION REPORT NO.
9. PERFORMING ORGANIZATION NAME AND ADDRESS ACT Systems Inc. Winter Park, Florida 32789		10. PROGRAM ELEMENT NO. C61C1C SOS#1, Task 39 11. CONTRACT/GRANT NO. 68-03-2506
12. SPONSORING AGENCY NAME AND ADDRESS Municipal Environmental Research Laboratory--Cin., OH Office of Research and Development U.S. Environmental Protection Agency Cincinnati, Ohio 45268		13. TYPE OF REPORT AND PERIOD COVERED Final Report 9/78-9/79 14. SPONSORING AGENCY CODE EPA/600/14
15. SUPPLEMENTARY NOTES See also Volume I (EPA-600/2-80-012a) Project Officer: Robert M. Clark (513) 684-7488		
16. ABSTRACT A cost analysis for system water supply utility management has been developed and implemented in Kenton County, Kentucky, Water District No. 1. The system description is contained in Volume I of the two volume report resulting from the study. This volume (Volume II) contains the program documentation for the cost analysis system.		
17. KEY WORDS AND DOCUMENT ANALYSIS		
a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group
Accounting Computer Program Cost Analysis Cost Center Cost Effectiveness Cost Engineering Water Supply	Product Costing	13B
18. DISTRIBUTION STATEMENT Release to Public		19. SECURITY CLASS (<i>This Report</i>) unclassified 20. SECURITY CLASS (<i>This page</i>) unclassified
		21. NO. OF PAGES 351 22. PRICE