A DATA-MANAGEMENT SYSTEM FOR USE IN GROUND-WATER MODELING AND RESOURCE EVALUATION

By Randall L. Fields and Edward F. Vetter

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 84-4014

Tucson, Arizona
March 1984

UNITED STATES DEPARTMENT OF THE INTERIOR

WILLIAM P. CLARK, Secretary

GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information
write to:

District Chief
U.S. Geological Survey
Box FB-44
Federal Building
301 West Congress Street
Tucson, Arizona 85701

Copies of this report can be
purchased from:

Open-File Services Section
Western Distribution Branch
U.S. Geological Survey
Box 25425, Federal Center
Denver, Colorado 80225
Telephone: (303) 234-5888

CONTENTS

ILLUSTRATIONS

TABLES

A DATA-MANAGEMENT SYSTEM FOR USE IN GROUND-WATER MODELING AND RESOURCE EVALUATION

By

Randall L. Fields and Edward F. Vetter

ABSTRACT

A data-management system was developed for use by the Southwest Alluvial Basins, Regional Aquifer-System Analysis study. Of the more than 100 computer programs developed during the study, the 9 programs described in this report form the nucleus of this ground-water data-management system. Five of the programs are used to edit and update changes to the system's master data file. The remaining programs include a polygon-definition program, two programs for data retrieval and reporting, and a program for plotting areal distributions of ground-water data.

INTRODUCTION

This report describes a data-management system developed for use by the Southwest Alluvial Basins, Regional Aquifer-System Analysis (Swab/RASA) study. The study area includes about 84,000 square miles in southern and central Arizona and small areas in California, Nevada, New Mexico, and Mexico (Anderson, 1980, p. 3). The area was divided into 72 basins for modeling purposes. Available data on which to base the models and the general analyses of the hydrologic systems were numerous in some basins and sparse in others. A major task of the Swab/RASA study was the accumulation and management of existing data on ground-water quantity and quality. The system was developed for use on local computer facilities where the data would be available for relatively quick and inexpensive access.

The large amount of available ground-water data present a significant data-management problem. In order to use the information effectively for model development, resource evaluation, and data reporting, an integrated data-handling, analysis, and plotting system is needed. This computer-based system would consist of a set of files for the different types of data; plotting and statistical routines; and retrieval, reporting, and editing programs (Anderson, 1980, p. 16-17).

This data-management system is used to supplement the U.S. Geological Survey's National Water Data Storage and Retrieval System (WATSTORE) by including a large amount of data from other sources.

The Swab/RASA data-management system represents an attempt to include all the reliable ground-water data for Arizona in a single data system.

Purpose and Scope of Report

The purpose of this report is to provide program documentation that will enable a user to implement and run the programs as the foundation of a ground-water data-management system. The scope of this report is limited to descriptions of the programs and data files and operational considerations for implementing the system.

Relation to Goals

One of the goals of the Swab/RASA study is to accumulate, analyze, and systematically file available ground-water data for use in ground-water modeling and ground-water resource evaluation. A data-management system that could provide efficient and relatively inexpensive access to ground-water data using local computer resources was necessary. During the 4 years of the Swab/RASA study, more than 100 computer programs were developed for use in the evaluation of ground-water data. The data were used to construct three-dimensional digital models of basins within the study area. The nine programs described in this report provide the updating and retrieval of the data file and areal plotting of data distributions.

The data used in the study are indexed by a master data file and stored in eight working files. The working files are segregated by general type of data—water-level data, water-quality data, and well-log data. The programs, which allow editing of changes to the master file, updating of the master file, and retrieval and reporting of all well-site data, form the nucleus of the data-management system. Programs that provide basic graphic representation of the data in the form of areal plots also are included. The relations between the system's programs and data files are shown in figure 1.

National Water Data Storage and Retrieval System

The National Water Data Storage and Retrieval System (WATSTORE) was established in November 1971 to modernize the U.S. Geological Survey's existing water-data processing procedures and techniques and to provide for the effective and efficient management of its release of information. The system is operated and maintained on the central computer facilities of the U.S. Geological Survey in Reston, Virginia.

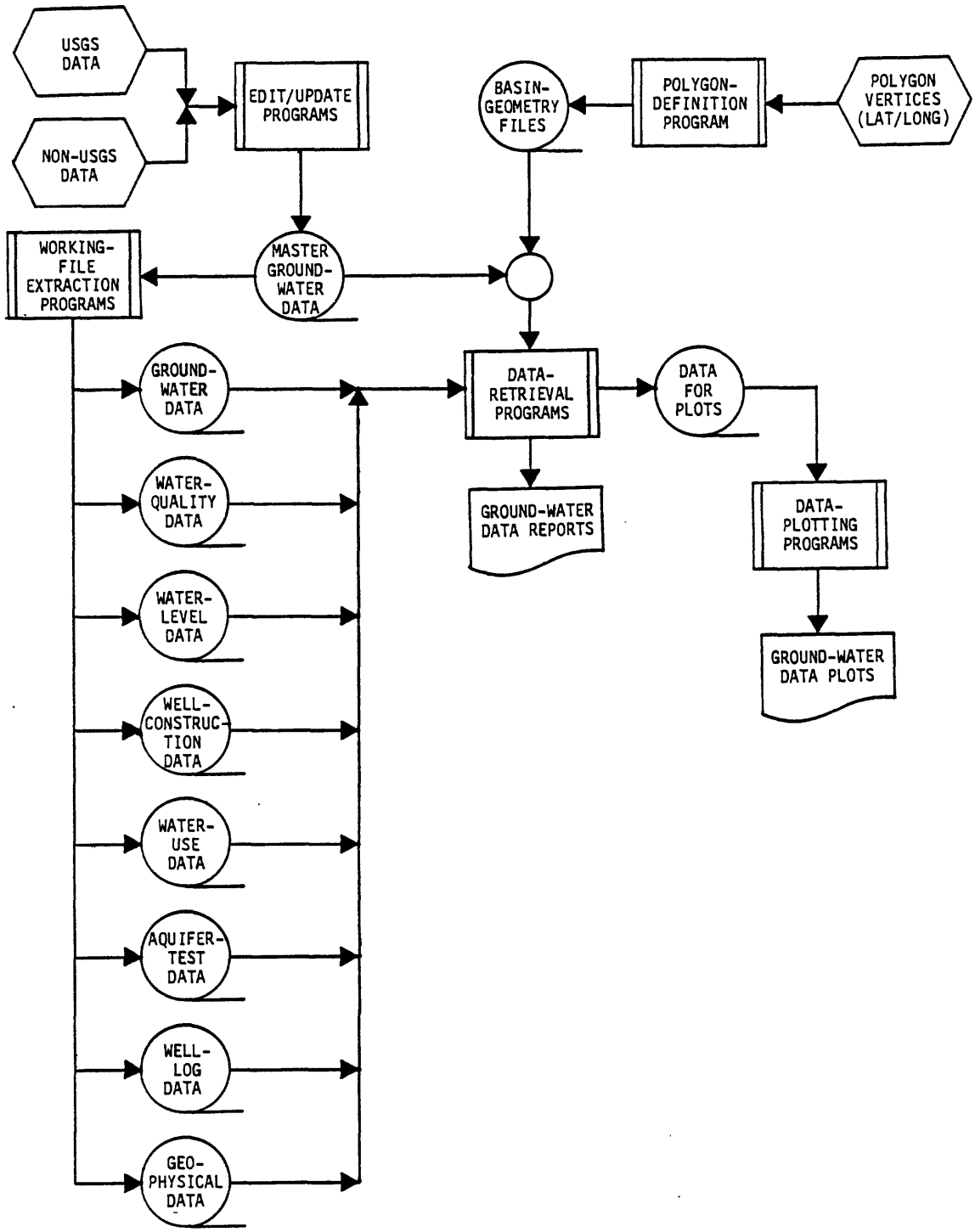


Figure 1.--Relations between programs and data files in the data-management system.

The WATSTORE system consists of several files in which data are grouped and stored by common characteristics and frequency of data collections. The system is also designed to allow for the inclusion of additional data files. Files are maintained for the storage of (1) surface-water, water-quality, and ground-water data measured on a daily or continuous basis, (2) annual peak values for streamflow stations, (3) chemical analyses for surface-water and ground-water sites, and (4) geologic and inventory data for ground-water sites. An index file of sites for which data are stored in the system is also maintained (U.S. Geological Survey, 1976).

DESCRIPTION OF THE DATA-MANAGEMENT SYSTEM

The programs in this ground-water data-management system can be divided into three major parts: (1) utility computer programs, (2) computer programs for editing and updating, and (3) programs for data retrieval and reporting. The only utility program described, POLYGON, is used to define polygons for data retrieval. Three programs edit changes to be made to the master file, and two programs perform the actual updating of the master file. Two data-retrieval programs and one graphics program, which is used to plot areal distributions of data within a given area, are discussed. The data-management system was developed for batch processing.

Computer Languages

The choice of high-level computer languages to be used in the development of this data-management system was limited by several factors. The factors included (1) the standardization of the language(s), (2) efficiency of the language(s), and (3) portability of the programs.

The criteria that the language(s) used must be standardized by the American National Standards Institute (ANSI) narrowed the field of available languages to the COmmon Business Oriented Language (COBOL) and the FORmula TRANslation (FORTRAN) language. All programs were written in Control Data Corporation* (CDC) FORTRAN IV Extended, version 4.6 and (or) CDC COBOL, version 4.7.

In terms of efficiency of the language(s) used, FORTRAN has long been the workhorse of the scientific community for numerical computing. COBOL, however, was designed for handling large amounts of data input and output and has several string-handling and table- or array-handling capabilities.

---

*The use of trade names is for descriptive purposes only and does not constitute an endorsement by the U.S. Geological Survey.

COBOL and FORTRAN, either as basic or extended languages, are available on virtually all but the smallest computer systems. Thus, portability of programs written in these languages presents fewer problems, and the programs can be moved from one computer system to another with minor modifications.

All the editing, updating, and retrieval programs in the master file are written in COBOL. FORTRAN subroutines are used by two of the editing programs to handle the conversion of latitudes and longitudes from degrees, minutes, and seconds to decimal degrees. The polygon-definition program and the areal-plot program are written in FORTRAN with no COBOL interfaces. Great care was taken during the development of the data-management system programs to insure that all source-code statements adhere to the ANSI standards for each language.

Variable-Length Records

All data files maintained by this system are made up of records with a variable-length or trailer-type record format. Each record consists of two major parts—a 22-character fixed-length record header and a variable-length part, or trailer, that may contain from 1 to 999 characters.

The fixed-length record header contains a site-identification number; a three-digit WATSTORE-type record identifier; a three-digit number, the value of which determines the length of the variable part of the record; and a one-character security or access code. The record-type identifiers currently in use in the WATSTORE system are listed in table 1. The fixed-length record header for a variable-length record has the following format:

Position	Variable	Format	Description
1-15	KEYID	9(15)	Site identifier
16-18	RNUM	999	Record identifier or repeating group number
19-21	LNUM	999	Length of variable portion in characters
22	SESAFE	X	Access and (or) security code; may be used to limit access to data record

The variable-length trailer part of the record contains as many character positions as are indicated in the length field of the record header. For example, if the length field of the record header has a value of 18, the variable-length trailer will have 18 character positions.

Table 1.--Record-identification numbers in use by WATSTORE

Record number	Description
000	GWSI entry record
001	QW header record
002	QW name record
003	QW parameter record
042	Lift data
047	Major pump data
055	Standby power data
058	Construction data
072	Hole dimensions
082	Openings data
076	Casings data
164	Minor repairs
090	Geohydrologic unit data
094	Aquifer data
098	Hydraulic data
105	Coefficients
114	Quality network
121	Level network
127	Pumpage network
134	Flow data
146	Pump-production data
158	Owners
171	Springs
183	Remarks
186	Site visits
180	Other data
189	Other identifiers
192	Field water quality
198	Available log data
203	Well group
208	Pond-Tunnel-Drain
212	Cooperator data
219	Laterals
250	Miscellaneous values
277	State water use
269	Observation-well heading
320	Measuring point

Defining New Record Types

Data from sources other than U.S. Geological Survey can be entered into all WATSTORE data files, except the quality-of-water file.

A great deal of time and money, however, must be spent editing, reformating, and reentering the data in USGS formats. The use of variable-length records allows non-WATSTORE data to be entered directly into the data files in the same format as that in which the data are received. This is accomplished by creating a record header (see section entitled "Variable-Length Records") for the new data and attaching the new data to a record header.

A new record type is defined in the following manner: A record-type number not in use by the WATSTORE system is assigned to the new data record, and the length of the data part of the record is placed in the length field of the record header. After the new record has been defined, a COBOL description of the new record is written for insertion into the programs that will handle the new record type. The process of creating a new record type from non-USGS sources is illustrated in figure 2.

Data-File Storage

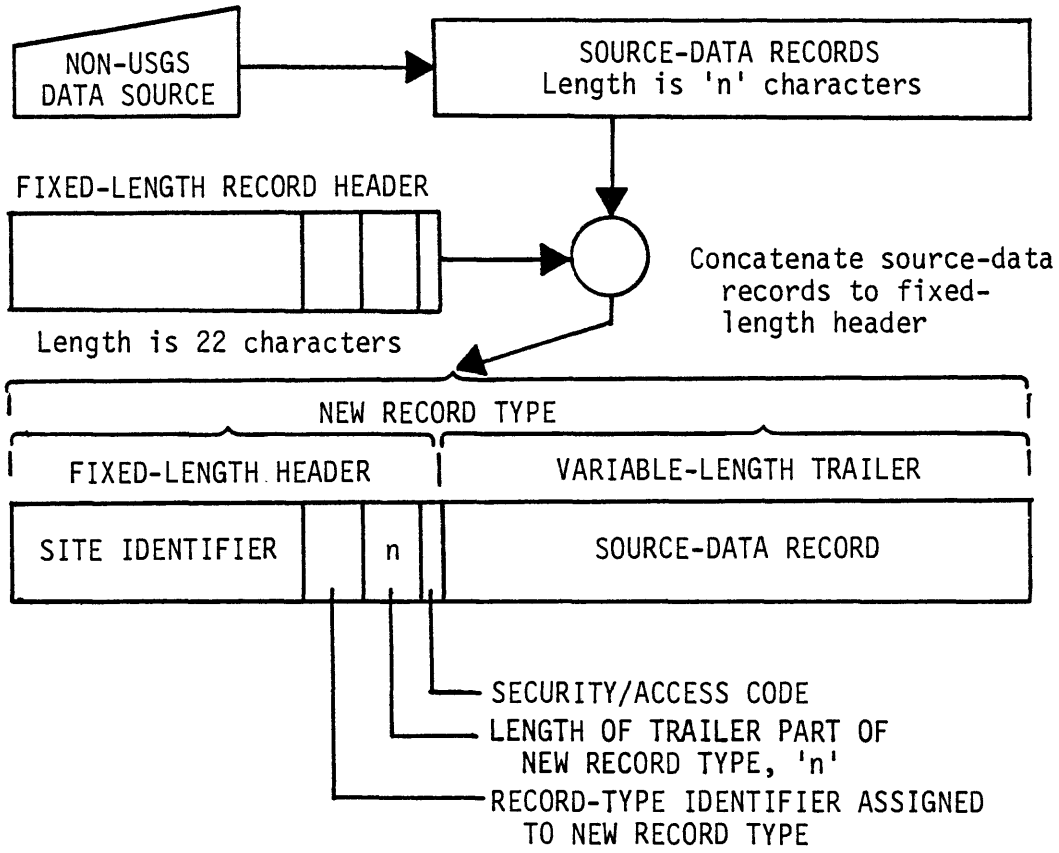
Data files in the data-management system are stored on magnetic tape. This storage medium was chosen because of the large sizes of the data files and the high cost of storing large amounts of data on an on-line storage device, such as magnetic disk. Although magnetic tapes do not allow the immediate access that magnetic disks allow, they do provide access times that are suitable for the needs of ground-water modeling and resource-evaluation studies.

Control Records for COBOL Programs

All COBOL programs developed as part of the Swab/RASA study make use of five control records for (1) controlling a program's function, (2) inputting the main and secondary titling information, and (3) signaling an end of the program's input data. The control records are identified by the letters CNL in positions 1 through 3 and a three-digit number in positions 4 through 6. The numbers 001 through 009 are reserved for controlling program functions.

In addition to the function-control number, the first control record may contain an optional two-digit number in positions 7-8, indicating the number of data files to be read by the program; however, this number is used only in the EDIT-1 and EDIT-2 programs. The remaining 72 positions of the first control record are available for comments or a brief description of the data being processed.

Three control records are used to carry information for printed page headings. These three title records are identified by the numbers 010, 011, and 012 in positions 4-6. Control record 010 is used for the principal title for the printed listing or report. There are 74 character



COBOL RECORD DESCRIPTION OF NEW RECORD TYPE:

```
                1         2         3         4   7
1----+----0----+----0----+----0----+----0...012

       01  RECORD-NAME.
           03  SITE-ID              PIC 9(15).
           03  RECORD-ID-NUM        PIC 9(3).
           03  TRAILER-LENGTH       PIC 9(3).
           03  SECURITY-CODE        PIC X(1).
           03  TRAILER-PART         PIC X(1)  OCCURS
                    1 TO 999 TIMES DEPENDING ON
                    TRAILER-LENGTH.
```

Figure 2.--Process of creating a new master-file record type with COBOL record description.

positions available for the main title. The next record, 011, is a name record used to identify the person or agency requesting the run and the person or agency that is the source of the data. The third record, 012, is used to place a subtitle on the listing or report.

The last common control record contains 999 in positions 4 through 6 and indicates to the program that all input data have been processed. The formats for these five control records are:

Record number	Positions	Variable name	Format	Description
1	1- 3	CNL-ID	XXX	Must be CNL
	4- 6	CNL-NUM	999	Must be 001 through 009
	7- 8	CNL-FLE	99	Number of files to be read from input
	9-80	CNL-REST	X(72)	Available for comments or brief description of data or program run
2	1- 3	CNL-ID	XXX	Must be CNL
	4- 6	CNL-NUM	999	Must be 010
	7-80	TITLEX	X(74)	Main title for printed page headings
3	1- 3	CNL-ID	XXX	Must be CNL
	4- 6	CNL-NUM	999	Must be 011
	7-36	GNAME	X(30)	Person initiating program run
	37-66	CNAME	X(30)	Person or agency requesting program run
	67-80	NOT USED		
4	1- 3	CNL-ID	XXX	Must be CNL
	4- 6	CNL-NUM	999	Must be 012
	7-66	SRCAGC	X(60)	Subtitle for printed page headings
	67-80	NOT USED		
5	1- 3	CNL-ID	XXX	Must be CNL
	4- 6	CNL-NUM	999	Must be 999
	7-80	NOT USED		

Audit Trail

To insure that all changes to the master file can be accounted for, an audit trail is established by the first edit program. This audit trail is updated by each successive run of one of the edit or update

programs. Audit-trail information includes the Julian dates of the present processing run and the previous processing run and the output-sequence numbers for each record accessed during the present and previous processing runs.

The audit trail allows a user to trace any given change entered for edit and update processing from its original input through the final update processing or from the final update processing back to an original input record. Audit information also includes counts of records input to a program, the number of records output, the number of lines printed, the number of input files read, and the number of records written to error files.

DATA FILE DESCRIPTIONS

The data files maintained by this system include the master data file and eight working files (fig. 1):

—Ground-water data file
—Water-quality data file
—Water-level data file
—Well-construction data file
—Water-use data file
—Aquifer-test data file
—Well-log data file
—Geophysical data file

Data contained in the working files are consolidated into a single record type from one or more of the records contained in the master file for a given site. The working files contain several tens of thousands of records less than the master file and are used for data retrieval and reporting whenever possible to reduce processing time and cost.

Master Data File

The ground-water master data file contains available data for all ground-water sites within the State of Arizona for which reliable data could be collected. Currently, 26 WATSTORE record types are in use in this file (table 2). In addition, data records from city, county, State, and other Federal agencies and from private firms are added as the data are available.

The master file began as a magnetic tape copy of the WATSTORE Ground-Water Site Inventory (GWSI) file for Arizona. The original tape was reviewed to insure that all sites for which data were present had a GWSI entry record (WATSTORE record type = 000). This check is necessary because the WATSTORE system requires each site to have a header record in its data files before any other data and (or)

Table 2.--WATSTORE identification numbers and record titles used in the Swab/RASA ground-water master file

WATSTORE code	Total length	Variable length	Record description
000	251	229	General site data
001	83	61	Water-quality header record
002	87	65	Water-quality name record
003	94	72	Water-quality parameter record
022	53	31	Opening data
042	47	25	Lift data
047	217	195	Major pump data
055	29	7	Standby power supply
058	86	64	Well-construction data
072	41	19	Dimensions of the hole constructed
076	41	19	Casing data
090	81	59	Geohydrologic unit descriptors
094	40	18	Aquifer data
114	38	16	Water-quality network data
121	37	15	Water-level data
127	38	16	Pumpage or withdrawal data
134	72	50	Production data, flowing
146	74	52	Production data, pumped
158	112	90	Owner-identification record
171	74	52	Other spring data
180	64	42	Other data available
183	70	48	Pertinent remarks
186	70	48	Site visit data
189	102	80	Other site identification
198	38	16	Available log data
234	40	18	Water-level data schedule

NOTE: Detailed descriptions of all record types can be found in the WATSTORE User's Guide, Volume 2, Chapter 2.

records can be entered for a particular site. The same method is used in this data-management system. A type "000" record must exist for each site in the master data file before any other record types can be entered in the file.

The next step was to obtain magnetic tape copies of the water-quality and water-level files for the State of Arizona. Checks were made of the water-quality tapes to insure that each site that had water-quality data also had a water-quality header record. The water-level tape was checked against the GWSI tape to insure that every site that had a water-level measurement also had a GWSI entry record in the GWSI file. These four tapes—the ground-water site inventory tape, the water-quality header file tape, the water-quality history file tape, and the water-level tape—were then sorted into descending order using the site-identification numbers. The site-identification number is composed of the latitude and longitude of the site—degrees, minutes, and seconds (DMS)—and a two-digit sequence number to differentiate wells that have the same latitude and longitude. The sorting of records is performed on descending latitude, descending longitude, ascending site-identifier, ascending record-type identifier, and sequence number. The four files were merged into a single ground-water master data file. This first copy of the master file contained about 267,000 records that represented about 26,000 ground-water sites.

Working Data Files

Ground-Water Data File

The working data file is composed of records of a fixed-length structure that have a length of 270 characters. These records can be handled by the programs as either fixed-length or variable-length records. Each record is a summary of the available ground-water data for a given site. The contents of the ground-water record were determined by an analysis of requests for information that have been handled by the Water Resources Division's Arizona District data-processing unit. Where multiple values may exist for a particular data element in the master file, the most recent value is stored in the ground-water working data file. The file is in descending order by the site-identification number. The format of the ground-water working data file is:

Position	Format	Description
1-15	9(15)	Site-identification number
16-18	999	Record-type identification number
19-21	999	Length of variable portion
22	X	Data security or access code
23-42	X(20)	Local site identifier
43-48	99v9999	Latitude of site (decimal degrees)

49-55	999v9999	Longitude of site (decimal degrees)
56	X	Type of site
57-66	X(10)	Ground-water area code
67-69	999	County code
70-76	9(5)v99	Altitude of site
77	X	Primary use of site
78	X	Primary use of water
79-85	9(5)v99	Depth of hole
86-92	9(5)v99	Depth of well
93-99	9(5)v99	Water level (most recent)
100-107	X(8)	Date water level measured
108	X	Method of measurement
109	X	Site status
110-117	X(8)	Date of construction completion
118-127	X(10)	Other site identifiers
128-142	X(15)	Assigner of other identifiers
143-150	X(8)	Date of most recent chemical analysis
165	X	Type of finish
166-172	9(5)v99	Depth to top of casing
173-179	9(5)v99	Depth to bottom of casing
180-183	9(3)v99	Diameter of the casing
185-191	9(5)v99	Depth to top of opening
192-198	9(5)v99	Depth to bottom of opening
199	X	Type of opening
200-207	X(8)	Date of production measurement
208-214	9(5)v99	Discharge
215-220	9(4)v99	Drawdown
221	X	Method of production data measurement
222-228	9(5)v99	Production water level
229-235	9(5)v99	Static water level
236-242	9(5)v99	Specific capacity
243	X	Type of lift
244-251	X(8)	Available log types
252-259	X(8)	Geohydrologic unit identifier
260	X	Contributing unit
261-266	9(5)	Number of water-level measurements

Water-Quality Data File

The water-quality working data file is sorted in descending order using the site-identification number. Each record is of variable-length structure and may range from 30 to 631 characters in length.

Each record represents a single site and the working-file data is a composite of all the water-quality data for that site for a particular date in the master file. The water-quality working file records have the following format:

Position	Format	Description
1-15	9(15)	Site-identification number
16-18	999	Record-type identification number
19-21	999	Length of variable portion
22	X	Data security or access code
23-24	99	Number of parameters in this record (0-50)
25-30	X(6)	Date of most recent analysis
31-36	9(5)	Parameter code 1
37-41	v9(5)	Parameter value 1
42-43	99	Parameter value exponent 1
.		.
.		.
.		.
620-624	9(5)	Parameter code 50
625-629	v9(5)	Parameter value 50
630-631	99	Parameter value exponent 50

A single water-quality record may contain values for at least 1 but not more than 50 chemical parameters. A parameter value in the working file is the arithmetic mean of all values for that parameter for a given site in the master file. The water-quality working file record has a value for every constituent that was recorded for a given site in the master file even though a particular constituent may not appear in every analysis.

Water-Level Data File

This sequential file contains all water-level measurement records (WATSTORE record type = 234) that have been entered into the master file. The records in this file are 40 characters in length and are fixed-length in structure. The format of the record allows them to be processed as either fixed-length or variable-length records. The file is sorted into descending order using the site-identification number as the

primary sort key and the date measured as the secondary sort key. Water-level records have this format:

Position	Format	WATSTORE code	Description
1- 15	9(15)	C1	15-digit site-identification number
16- 18	999		Record-type identification number
19- 21	999		Length of variable portion
22	X		Data security or access code
23- 30	X(8)	C235	Date on which the water level was measured
31	X	C236	Accuracy of the date
32- 37	9(4)v99	C237	Water-level measurement
38	X	C238	Site status at time of measurement
39	X	C239	Method of measurement
40	X	C240	Accuracy of the measurement

Well-Construction Data File

The well-construction data file contains as much reliable data pertaining to the construction and alteration of wells as can be obtained from contractors' records.

Water-Use Data File

The water-use data file is composed of 90-character records. The records in this file represent power consumption and reported pumpage data received from public and private sources. Each record contains data for type of power used, amount of power used, estimated or measured pumpage, and the primary use of the water pumped. The file is sorted into descending order using the site-identification number and the

year of pumpage as the primary and secondary sort keys, respectively. Water-use records have the following format:

Position	Format	Description
1- 15	9(15)	Site-identification number
16- 18	9(3)	Record number
19- 21	9(3)	Length of variable part
22	X(1)	Security or access code
23	X(1)	Quadrant
24	X(1)	Half-quadrant
25- 26	X(2)	Township
27	X(1)	Half-township
28- 29	X(2)	Range
30	X(1)	Half-range
31- 32	X(2)	Section
33	X(1)	Quarter-section
34	X(1)	Quarter-quarter-section
35	X(1)	Quarter-quarter-quarter-section
36	X(1)	Quarter-quarter-quarter-quarter-section
37- 39	X(3)	Ground-water area code
40- 45	X(6)	Owning agency
46- 48	X(3)	Energy source
49- 54	X(6)	Meter number
55- 69	X(15)	Account number
70- 71	X(2)	Revenue class
72- 73	X(2)	Reporting year
74- 81	9(8)	Power usage
82- 83	X(2)	Type of power used
84- 88	9(4)v9	Divisor
89- 90	X(2)	Year divisor was determined
91	X(1)	Source of divisor
92- 97	9(6)	Pumpage, in acre-feet per year
98- 99	X(2)	Primary use of water

Aquifer-Test Data File

The aquifer-test data file is composed of data from numerous public and private sources. The records are of variable-length structure and contain the results of from one to six individual aquifer tests or steps of a variable-discharge step-drawdown test. The file is sorted into ascending alphabetic order using the basin or ground-water-area identifier. Within a particular ground-water area, the data are sorted in descending order using the site-identification number and the date of testing as the primary and secondary sort keys. The aquifer-test records are 332 characters in length and have the following format:

Position	Format	Description
1- 15	9(15)	Site-identification number of pumped well
16- 18	9(3)	Record number (800)
19- 21	9(3)	Length of variable portion
22	X(1)	Security or access code
23- 36	X(14)	Local site identifier of pumped well
37- 42	X(6)	Basin identifier
43- 44	X(2)	Type of test
45- 52	9(8)	Beginning date of test
53- 56	9(4)	Static water level, in feet below land surface
57- 60	9(4)	Discharge measurement 1, in gallons per minute
61- 65	9(4)v9	Duration of discharge 1, in hours
66- 69	9(4)	Discharge measurement 2, in gallons per minute
70- 74	9(4)v9	Duration of discharge 2, in hours
75- 78	9(4)	Discharge measurement 3, in gallons per minute
79- 83	9(4)v9	Duration of discharge 3, in hours
84- 87	9(4)	Discharge measurement 4, in gallons per minute
88- 92	9(4)v9	Duration of discharge 4, in hours
93- 96	9(4)	Discharge measurement 5, in gallons per minute
97-101	9(4)v9	Duration of discharge 5, in hours
102-105	9(4)	Discharge measurement 6, in gallons per minute
106-110	9(4)v9	Duration of discharge 6, in hours
111-114	9(4)	Production water level, in feet below land surface
115-121	X(7)	Source of data
122	X(1)	Accuracy of water-level measurement
123	X(1)	Accuracy of discharge measurement
124-128	9(5)	Depth of pumped well, in feet
129-133	9(5)	Altitude of pumped well
134-138	9(5)	Depth to top of open section, in feet below land surface
139-143	9(5)	Depth to bottom of open section, in feet below land surface
144	X(1)	Type of opening
145-149	9(5)	Radial distance of pumped well, from observation weel, in feet
150-164	9(15)	Site identifier of observation well
165-171	9(7)	Transmissivity 1, in feet squared per day
172-177	v9(6)	Storage coefficient 1

178-192	X(15)	Method of computation 1
193-199	9(7)	Transmissivity 2, in feet squared per day
200-205	v9(6)	Storage coefficient 2
206-220	X(15)	Method of computation 2
221-227	9(7)	Transmissivity 3, in feet squared per day
228-233	v9(6)	Storage coefficient 3
234-248	X(15)	Method of computation 3
249-255	9(7)	Transmissivity 4, in feet squared per day
256-261	v9(6)	Storage coefficient 4
262-276	X(15)	Method of computation 4
277-283	9(7)	Transmissivity 5, in feet squared per day
284-289	v9(6)	Storage coefficient 5
290-304	X(15)	Method of computation 5
305-311	9(7)	Transmissivity 6, in feet squared per day
312-317	v9(6)	Storage coefficient 6
318-332	X(15)	Method of computation 6

Well-Log Data File

This file is made up of data from drillers' logs. The data are coded from copies of the driller's log and contain the driller's comments and measurements concerning the lithology encountered during construction of the well. Programs were developed as part of the Swab/RASA study whereby weighted average specific yield, hydraulic conductivity, and transmissivity could be estimated on the basis of data from drillers' logs. This in turn provided preliminary data for input to hydrologic models. The file is sorted into descending order using the site-identification number and date as primary and secondary sort keys, respectively.

Geophysical Data File

This file contains digitized data for the following types of geophysical logs:

—Gamma
—Neutron
—Density
—Resistance
—Resistivity, short
—Resistivity, long
—SP
—Caliper
—Temperature

The file is sorted into descending order using the site-identification number, ascending alphabetic order using the log-type code, and descending order using the date of the log.

UTILITY COMPUTER PROGRAMS

Polygon-Definition Program—POLYGON

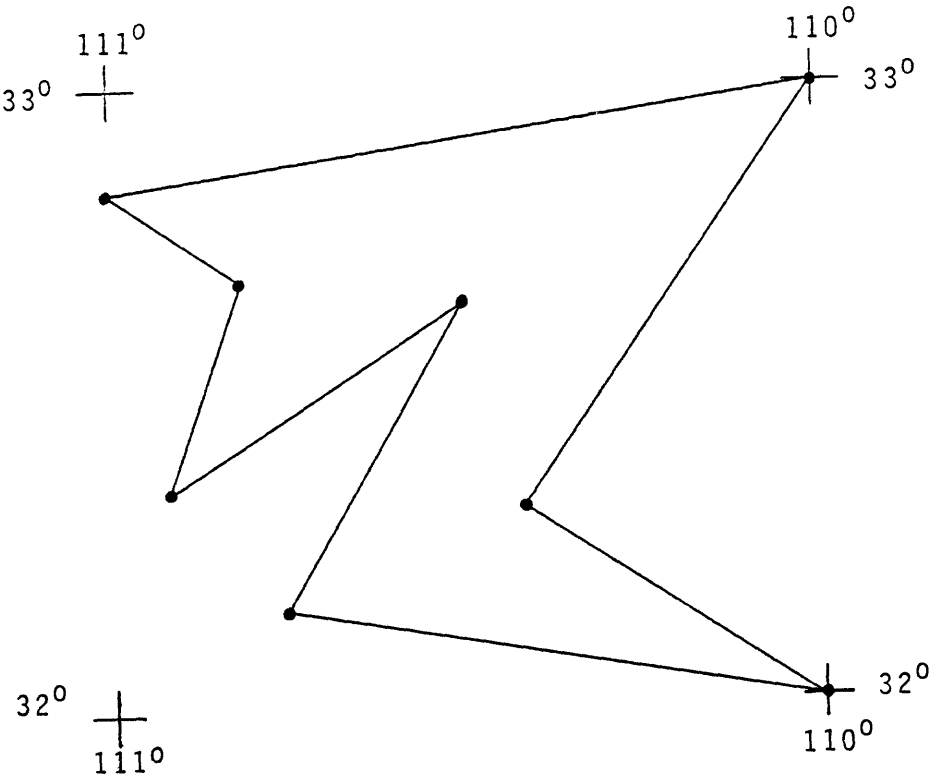
POLYGON is a FORTRAN program for computing a mathematical description of a given polygon. This description is used as a means of determining the existence of data sites within an area defined by a polygon for use in master file edit, update, and retrieval programs. The program can handle polygons defined by as many as 1,999 points.

Program Operation

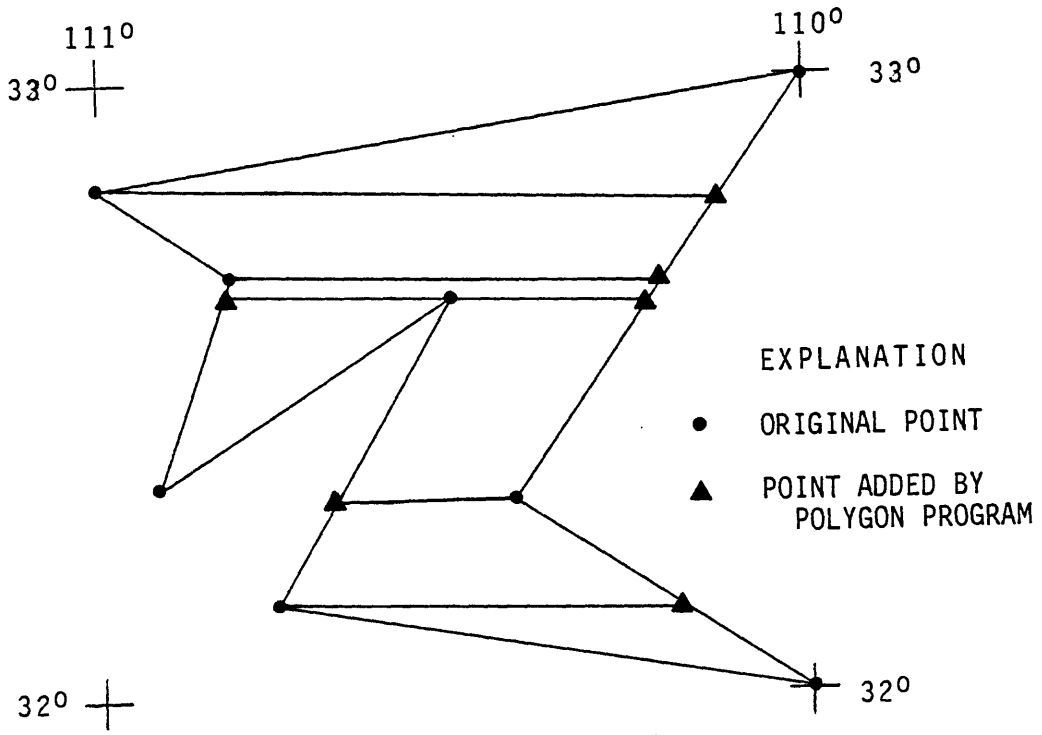
The program reads a series of points of latitude and longitude that define a boundary of a polygon as shown in figure 3A. The points must be input in sequential order around the periphery of the polygon in either a clockwise or counter-clockwise order. After all points have been read and stored in an array or table, the program closes the polygon by copying the coordinates of the first point into the position after the coordinates for the last point. For example, if a polygon is defined by 67 points, the coordinates of the first point will be copied to array position 68 to close the polygon.

The next step is to mathematically define the straight line segments that connect the points that make up the polygon's boundary. Consecutive points are paired to form directed line segments. The point that has the lower latitude of a pair becomes the beginning point of the line segment and the point with the higher latitude becomes the ending point of the line segment. If the high and low latitudes are equal, the point with the westernmost longitude becomes the line segment's beginning point. The mathematical equation for each line segment is computed. The line segments are then sorted in ascending order using the latitude of the beginning point as the sort key.

The program now views the polygon as a series of slices. Each slice is examined from left to right, additional points of corresponding latitude are added on the opposite side of the polygon, and additional line segments are defined in order to more completely define the polygon's shape. The addition of these extra points and line segments complete the definition of the polygon as a series of slices made up of trapezoids and triangles as shown in figure 3B. The line segments are sorted a final time in descending order by latitude and longitude of the beginning points, and a polygon-definition table is written to the program's output storage device (table 3).



A. As described by input data points.



B. As defined by the POLYGON program as a series of slices composed of triangles and trapezoids.

Figure 3.--Sample polygons.

Table 3. Example of "slice table" of line-segment equations for the polygon shown in figure 3

```
3216001100000330000111000002 SAMPLE POLYGON INPUT DATA
3252301110000330000110000328750-1109999330000-1099999+ 1250000- 9999999- 467499  16
3252301100614330000110000328750-1101041330000-1099999+ 5000000- 4166666- 687499  15
3246001104845325230111000327667-1108124328750-1109999+ 1083333+ 1875000-  58608  14
3246001101139325230110061432767-1101943328750-1101041+ 5000000- 4166666- 687499  13
3244301104921324600110484532741?-1108228327667-1108124+ 2500000- 1041666- 311162  12
3244301101254324600110113932741?-1102152327667-1101943+ 5000000- 4166666- 687499  11
3231001105500324430110492132516?-1109166327417-1108228+ 2500000- 1041666- 311162  10
3231001105500324430110300032516?-1109166327417-1104999+ 2250000- 4166666- 385048   9
3231001103917324430110300032516?-1106550327417-1104999+ 3666667- 2527777- 487929   8
3231001102409324430110125432516?-1104027327417-1102152+ 5000000- 4166666- 687499   7
3230001103959323100110391732500?-1106665325167-1106550+ 3666667- 2527777- 487929   6
3230001102500323100110240932500?-1104166325167-1104027+ 5000000- 4166666- 687499   5
3222301104510323000110395932375?-1107527325000-1106665+ 3666667- 2527777- 487929   4
3222301101135323000110250032375?-1101934325000-1104166+ 2333333+ 4166667- 122221   3
3216001100000322301104510322667-1099999323750-1107527+ 1083333+ 7527778+ 123730   2
3216001100000322301101135322667-1099999323750-1101934+ 2333333+ 4166667- 122221   1
000000
```

Equations of the Line Segments

The equation for each directed line segment is solved by the program. Given the coefficients for a line segment and either a latitude or longitude from any point along that line, the missing coordinate can be computed. For example, a polygon may be plotted in such a way that a portion of the polygon boundary would fall outside the limits of the plotting surface (top, bottom, left, or right side). The program that produces the plotter's instructions has enough information to determine at what point along the boundary of the polygon the line to be drawn would intersect the limits of the plotting area or surface. The longitude (sides) or latitude (top or bottom) of the plot area are known by the plotting program. The coefficients for the line segment that intersects one of these four boundaries are also known. The plot program has enough information to compute the unknown latitude or longitude of the point of intersection of the polygon boundary and the plot area boundary. The general system logic of POLYGON is illustrated in figure 4.

The line-segment equation is of the form $Ax + By = C$. Solution for the coefficients A, B, and C of each line segment in the POLYGON program is accomplished by solving the following equations:

$$A = y_2 - y_1$$

$$B = x_2 - x_1$$

$$C = (x_2y_1 - x_1y_2) + (x_1y_2 - x_2y_1)$$

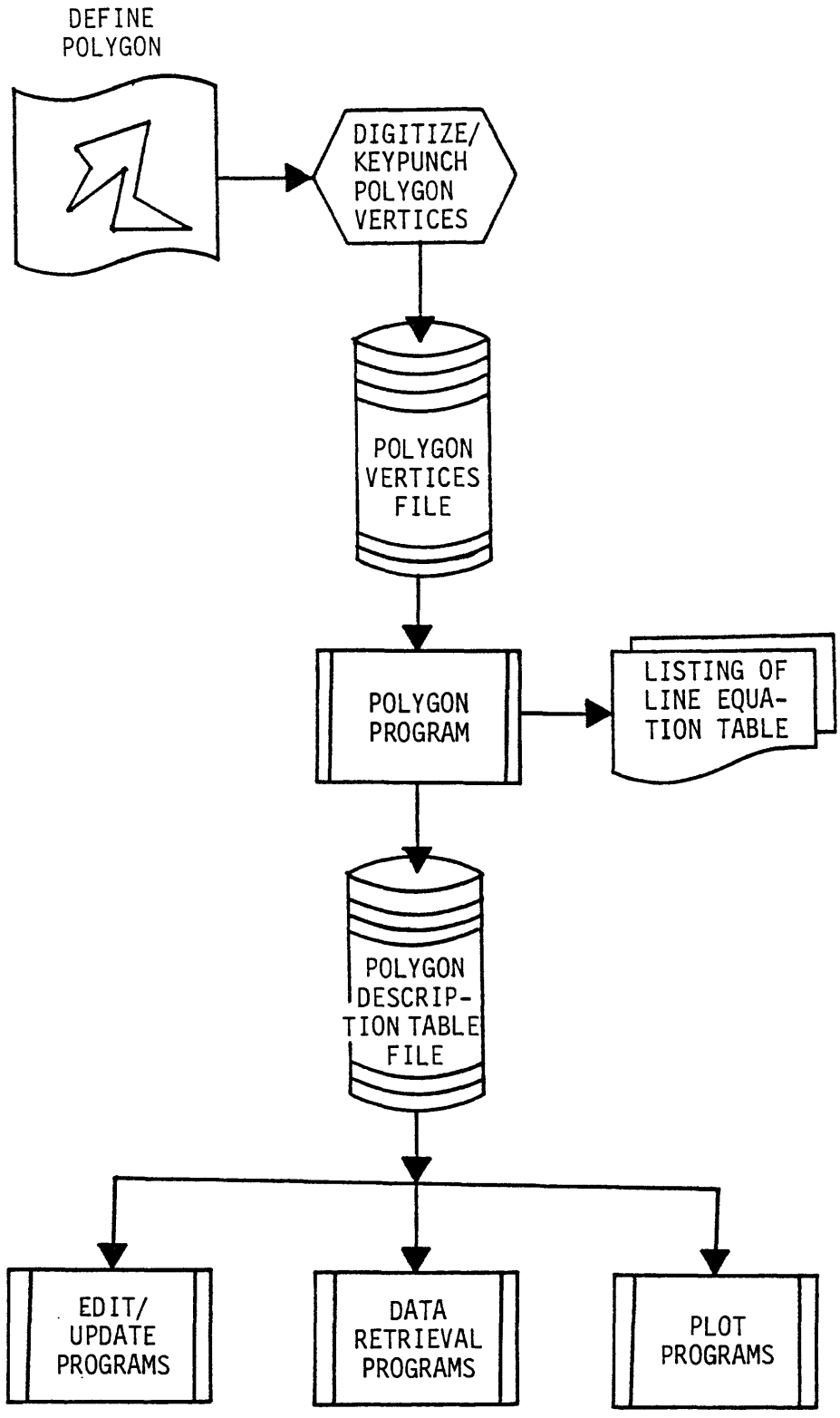


Figure 4.--General system logic for program POLYGON.

where

x_1 = longitude of the line's beginning point,

x_2 = longitude of the line's ending point,

y_1 = latitude of the line's beginning point, and

y_2 = latitude of the line's ending point.

To solve for an unknown longitude (x):

$$x = \frac{C - By}{A}$$

where

y = known latitude and
$A \neq 0$.

To solve for an unknown latitude (y):

$$y = \frac{Ax - C}{-B}$$

where

x = known longitude and
$B \neq 0$.

NOTE: Longitudes between 0° and 180° W. are expressed as negative values.

Program Input

Input to POLYGON consists of a polygon-identification record and a latitude-longitude record for each point to be defined on the polygon boundary. Latitudes and longitudes are entered in degrees, minutes, and seconds. The format for the polygon-identification record and the latitude-longitude records are:

Record	Position	Variable	Format	Description
1	1- 3		25X	Blank
	22-29		A8	Polygon ID code
	30-80		5A10	Polygon name
All other records	1- 3	NLAD	I3	Latitude, in degrees
	4		1X	Blank
	5- 6	NLAM	I2	Latitude, in minutes
	7		1X	Blank
	8- 9	NLAS	I2	Latitude, in seconds
	10-11		2X	Blank
	12-14	NLOD	I3	Longitude, in degrees
	15		1X	Blank
	16-17	NLOM	I2	Longitude, in minutes
	18		1X	Blank
	19-20	NLOS	I2	Longitude, in seconds

Program Output

POLYGON produces two output files: (1) a printed listing of the line-segment table and (2) a disk file containing the line-segment table (table 3). Each output record represents a single directed line segment and has the following format:

Position	Variable	Format	Description
1		1X	Blank
2- 7	LA1	I6	Latitude, DMS of beginning point
8-14	LN1	I7	Longitude, DMS of beginning point
15-20	LA2	I6	Latitude, DMS of ending point
21-27	LN2	I7	Longitude, DMS of ending point
28-33	KY	I6	Latitude, in decimal degrees of beginning point
34	I1	A1	Sign of latitude
35-41	KX	I7	Longitude, in decimal degrees of beginning point
42-47	KW	I6	Latitude, in decimal degrees of ending point
48	I2	A1	Sign of latitude value
49-55	KZ	I7	Longitude, in decimal degrees of ending point
56	I3	A1	Sign of coefficient A
57-64	KA	I8	Coefficient A of line equation
65	I4	A1	Sign of coefficient B
66-73	KB	I8	Coefficient B of line equation
74	I5	A1	Sign of coefficient C
75-81	KC	I8	Coefficient C of line equation
82-85	I	I4	Sequence number of line segment

POLYGON converts decimal degrees and the coefficients to integer format in order to simplify their handling by COBOL programs. Latitudes and longitudes are multiplied by 10,000 prior to output. Coefficients A and B are multiplied by 10,000,000 and coefficient C is multiplied by 10,000 prior to their being output by the program.

Error Messages

PLY01 THE CURRENT LINES ARE NOT MATCHED nnnnn

Two line segments have been encountered that are not consecutive. The value of 'nnnnn' indicates the line number of the first line segment of the mismatched pair.

PLY02 LATITUDES OF THE SECOND POINTS ARE OUT OF SEQUENCE

The first line segment of a consecutive pair of line segments should always have the lower latitude when the ending points of the line segments are compared.

PLY04 THERE IS A BREAK IN THE POLYGON nnnnn

This condition is detected in the program's sort routine. It occurs when all the following conditions are present: beginning latitude of line segment 'I' = beginning latitude of line segment 'I+1' AND the ending longitude of line segment 'I' is greater than both the beginning longitude of line segment 'I+2' and the ending longitude of line segment 'I+1'. This indicates there is no connection between line segment 'I' and line segment 'I+1'. The value 'nnnnn' indicates the sequence number of the equation for line segment 'I'.

PLY05 DUPLICATE LINE IN TABLE i, I, nI, kI, number

The line segment being inserted into the table has the same beginning and ending points as the previous line that was inserted into the line segment table.

PLY06 LAST TABLE INDEX NOT CORRECT nnnnn mmmmm

Indicates that an error has occurred while the line segment table was being expanded to accept a new line segment. The value 'nnnnn' is the last valid table-index value, 'mmmmm' is the table-index value that caused the error.

PLY07 INVALID END OF PROGRAM IN ROUTINE 514 nnnnn

PLY08 INVALID END OF PROGRAM IN ROUTINE 534 nnnnn

Either of these two errors occur when the value of the beginning latitude of line segment '1+2' is equal to 999.999 indicating that the program has attempted to sort records beyond the end of the line segment table. This error condition may never occur once the program is installed on a given machine. These messages would indicate that the number of line segments to be sorted by the program differed from the actual number of line segments encountered during the sorting operation. The number of line segments to be sorted is greater than the actual number of line segments. They were used during program development and would be useful in testing the program after or during conversion or program modification.

COMPUTER PROGRAMS FOR EDITING AND UPDATING THE GROUND-WATER MASTER FILE

Changes to the master file include addition of data, changes of existing data, and deletion of existing data from the master file. Editing of the changes to the master file is a process of five major steps:

(1) Read the changes into a temporary data set and assign initial audit-trail information to each record.

(2) Sort the changes into descending order using the site identifier as the primary sort key.

(3) Check the site identifier of each change record to insure that all records to be changed, deleted, or added are within the boundaries of the district.

(4) Convert each free-field format record into a set of formatted change records.

(5) Check the data fields of the change records for valid codes and (or) the reasonableness of data values.

After change data have been fully edited, three steps are involved in the actual updating of the master file:

(1) All records in the master file that belong to a site that has a change in the change file are written to a temporary data file for update processing.

(2) Changes are made to the temporary-change master file.

(3) The changed master is merged with the records from the old master file that were not changed, which creates a new updated copy of the master file.

Master File Edit Program—EDIT-1

This program is the first in a series of five used to edit and update the master ground-water data file. The program reads change records in the WATSTORE free-field format as shown below, initiates an

```
343234111513801 R=0$T=M$12='A-13-05 06DCB3'$
343158111513401 R=234$T=A$235#02024019815237=12.20$239=S$
343224111502401 R=0$T=M$12='A-13-05 08AAB1'$20=15060203$
342852111492401 R=0$T=M$10=111492315316=3060.$19=P$12='A-13-05 33AAB1'$
342852111492401 35=$
342852111492401 R=42$T=A$43#S$254#001$45=E$
342852111492401 R=320$T=M$321#03020019805323=-0.8$324='TOP OF CASING'$
342852111492401 R=234$T=A$235#01026019825237=127.7$239=S$
```

audit trail, checks them for location within the geographic area represented by a polygon and stores the changes for further processing. The general system logic for EDIT-1 is illustrated in figure 5.

Audit-Trail Initiation

Three data fields are added to each of the input records that serve as an audit trail. These three items are (1) a sequence number, (2) the Julian date, and (3) a change-file identifier or volume serial number (VSN).

Testing Records for Inclusion in the Master File

The latitude and longitude of each record are tested against the boundaries of a polygon. If a record's latitude or longitude are outside the polygon, a message is printed indicating that the record has been dropped from the output change file. If the latitude and longitude are within the boundaries of the polygon, a message is printed indicating that the record has been stored in the output change file.

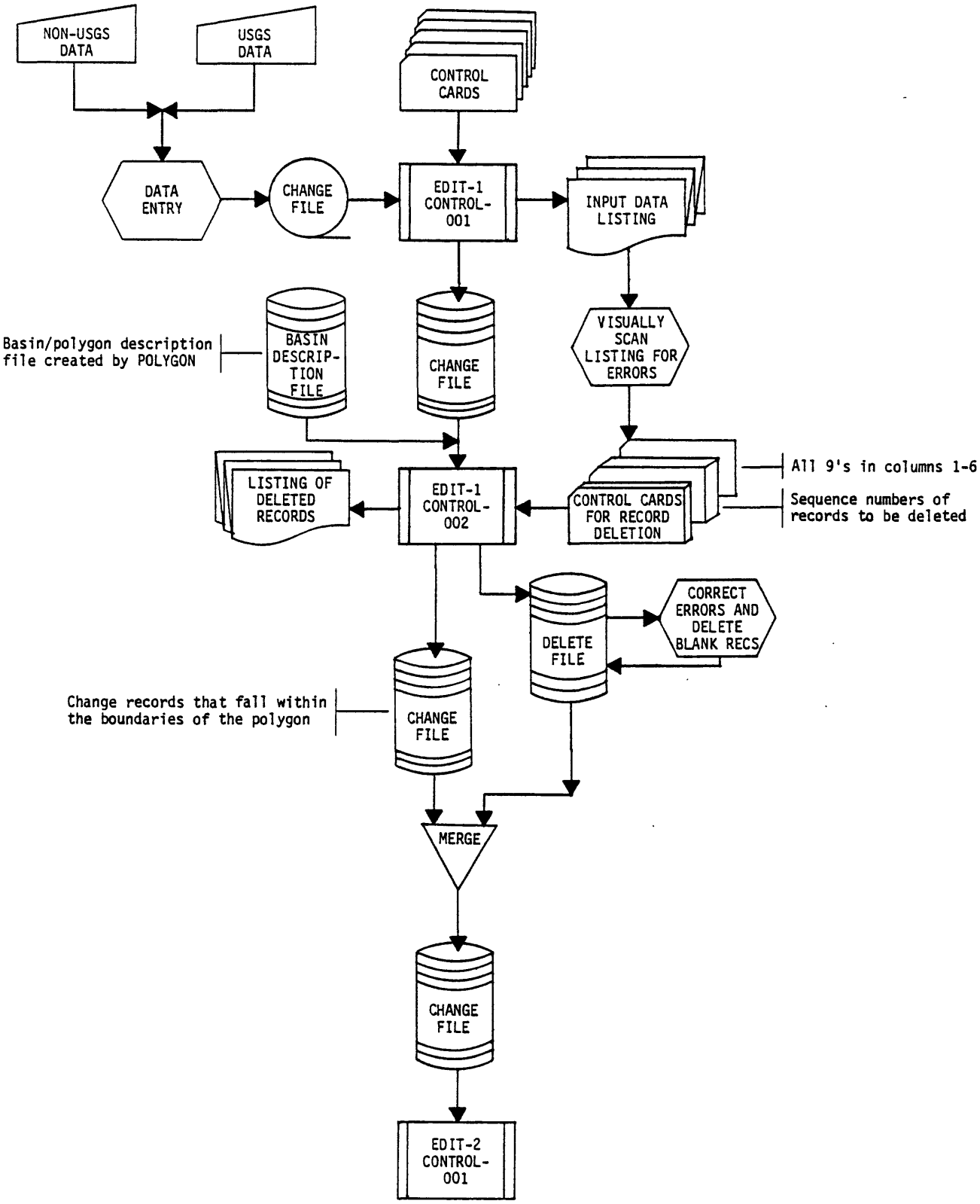


Figure 5.--General system logic for program EDIT-1.

EDIT-1 Input Data

Input data consists of three record types:

—Control records
—Free-field format change records
—Polygon-description file from the polygon program

Processing under Control-001, the program (1) reads the input free-field format data, (2) assigns a file identifier or VSN read from the input control records, (3) inserts the current Julian date into each record, (4) assigns an input sequence number to each change record, and (5) prints a listing of all input data with a message indicating whether the record was stored or dropped on output. The program, under Control-001, can also be used to delete selected records from the output file.

Processing under Control-002, the program checks the location of each site to insure that the site is within the boundaries of the polygon represented by the master file. The polygon boundary must be defined by the POLYGON program.

Control-003 allows the user to write selected records to a disk or tape file. Records can be selected by either their input sequence number or by their site-identification numbers. This function is used to aid in making corrections to the change data.

Control records.--A minimum of seven control records are required for the proper operation of EDIT-1. The first control record determines the function performed and must be one of either CNL001, CNL002, or CNL003. The next three control records, 010, 011, and 012, contain information used to title the printed output as described in the section entitled "Control Records for COBOL Programs." Two control records follow the 012 record and are required for all processing with EDIT-1. These records are a volume serial number (VSN) record and a files control record. The VSN record has a six-digit plus one-alphabetic-character identifier that is used to identify the tape or disk volume that has the input change file on it. This VSN will become an integral part of the audit trail established under control-001. The files control record has a six-digit number that passes the number of files to be read from the volume identified by the VSN record. The formats for the VSN and files records are:

Position	Format	Description
VSN Record		
1- 6		Not used
7-12	9(5)	Numeric part of the volume serial number

13	X(1)	Alphabetic part of the volume serial number
14-80		Not used

FILES Record

1- 6		Not used
7-12	9(6)	Number of files to be read from the input volume. Must be right jusified.
13-80		Not used

If control-001 is being used to delete records from the input change files, the record numbers are entered on delete records that follow the files record. These delete records have the following format:

Position	Format	Description
1- 6		Not used
7-12	9(6)	Sequence number of the record to be deleted. Sequence numbers are obtained from the printed listing from each run of EDIT-1.
13-80		Not used

The original input files should be run through EDIT-1 under control-001 until all records with obvious errors and all blank or 'junk' records are eliminated from the input files. One delete record must be in the control record set for each record to be deleted.

The last record in the control deck has the same format as the delete record and has 9's in positions 7 through 12. This record signals EDIT-1 that all control records have been processed.

Free-field format change records.--The WATSTORE type of free-field input is used in this system in order not to duplicate data entry. Positions 1 through 19 of this record type are of fixed format, and positions 20 through 80 are used to carry the change data in free-field format as described in the WATSTORE User's Guide, Volume 2, Chapter 2. The first four positions of the records are ignored by the program. Positions 5 through 19 MUST contain the 15-digit site-identification number.

The free-field format change records have the following format when read by EDIT-1:

Position	Format	Description
1- 4	X(4)	Not used by EDIT-1 ignored
5-19	9(15)	Site-identification number
20-80	X(61)	Free-field format data

A complete description of the free-field input-data format can be found in the WASTORE User's Guide, Volume 2, Chapter 2, pages B7-B12 and B98-B102.

Polygon-input records.--The polygon-definition program, POLYGON, produces a file that provides EDIT-1 with a mathematical description of the polygon for which data are being processed. The format of the polygon-definition records is:

Position	Format	Description
1	X(1)	Blank
2- 7	9(6)	Latitude, DMS of beginning point
8-14	9(7)	Longitude, DMS of beginning point
15-20	9(6)	Latitude, DMS of ending point
21-27	9(7)	Longitude, DMS of ending point
28-33	99V9(4)	Latitude, in decimal degrees of beginning point
34	X(1)	Sign of latitude
35-41	999V9(4)	Longitude, in decimal degrees of beginning point
42-47	99V9(4)	Latitude, in decimal degrees of ending point
48	X(1)	Sign of latitude value
49-55	999V9(4)	Longitude, in decimal degrees of ending point
56	X(1)	Sign of coefficient A
57-64	9V9(7)	Coefficient A of line equation
65	X(1)	Sign of coefficient B
66-73	9V9(7)	Coefficient B of line equation
74	X(1)	Sign of coefficient C
75-81	9(4)V9(4)	Coefficient C of line equation
82-85	9(4)	Sequence number of line segment

Program Output

Output from the program consists of three files—the change file, a pull file, and the line-printer listing. All three files contain essentially the image of the input records with only a few additions.

Output change file.--The output change file contains images of the input data record, in addition to the current Julian date, an input sequence number, and a file-identification or volume serial number. This is the file that is passed on to the next processing step or program as the input-data file.

Position	Format	Description
1- 6	X(6)	Volume serial number (VSN)
7-15	9(9)	Input sequence number
16-30	X(15)	Site-identification number
31-91	X(61)	Free-field change data
92-96	9(5)	Julian date

Output pull file.--This file contains exact images of the input data records. By removing erroneous and (or) blank records from this file, errors can be easily corrected and blank records deleted. After all corrections have been made, this file is re-edited to verify the corrections that were made. If the file contains only blank records and tape or file labels, the entire pull file can be discarded, only change records need be kept for correction.

Output line-printer file.--This file contains the paged listing of the input change file (55 lines per page). These records are images of the input change records with the addition of the audit-trail fields and a message that indicates whether the record was stored in the output file or deleted on input. An example of the line-printer listing from EDIT-1 program is shown below.

```
1** 8083C000000001325908112464701 R=234$T=A$235#1200801976$237=61.0$239=V$
2** 8083C000000002325908112464701 R=234$T=A$235#1202801977$237=63.0$239=V$
3** 8083C000000003325908112464701 R=234$T=A$235#0302401978$237=62.2$239=V$
4** 8083C000000004325908112464701 R=234$T=A$235#1202901978$237=56.2$239=V$
5** 8083C000000005325908112464701 R=234$T=A$235#0100401979$237=55.0$239=V$
6** 8083C000000006325908112464701 R=234$T=A$235#0101501979$237=52.9$239=V$
7** 8083C000000007325908112464501 R=234$T=A$235#0201401966$237=27.05$239=S$
8** 8083C000000008325908112464501 R=234$T=A$235#0100501967$237=32.54$239=S$
9** 8083C000000009325908112464501 R=234$T=A$235#0100201969$237=37.00$239=S$
10** 8083C000000010325908112464501 R=234$T=A$235#0100601970$237=41.31$239=S$
11** 8083C000000011325908112464501 R=234$T=A$235#0100701971$237=42.28$239=S$
12** 8083C000000012325908112464501 R=234$T=A$235#0100501972$237=49.61$239=S$
13** 8083C000000013325908112464501 R=234$T=A$235#0100401973$237=46.70$239=S$
14** 8083C000000014325908112464501 R=234$T=A$235#0300601973$237=51.84$239=S$
15** 8083C000000015325908112464501 R=234$T=A$235#0100201974$237=39.66$239=S$
16** 8083C000000016325907112462301 R=234$T=A$235#1202801977$237=54.9$239=V$
17** 8083C000000017325907112462301 R=234$T=A$235#0302401978$237=51.9$239=V$
18** 8083C000000018325907112462301 R=234$T=A$235#1202801978$237=47.3$239=V$
19** 8083C000000019325907112462301 R=234$T=A$235#1202901978$237=47.3$239=V$
20** 8083C000000020325907112462301 R=234$T=A$235#0100401979$237=44.2$239=V$
21** 8083C000000021325907112462301 R=234$T=A$235#0101501979$237=41.9$239=V$
22** 8083C000000022325942112484601 R=234$T=A$235#1200801976$237=60.0$239=V$
23** 8083C000000023325942112484601 R=234$T=A$235#1202801977$237=67.5$239=V$
24** 8083C000000024325923112483201 R=234$T=A$235#1202801978$237=55.6$239=V$
25** 8083C000000025325923112483201 R=234$T=A$235#0100401979$237=50.5$239=V$
```

Component-Format Edit Program—EDIT-2

This program reads the free-field change records, each of which contains changes to a WATSTORE repeating group, and reformats them into records containing only a single component change. The program groups these single component change records into change sets. The transition from free-field format to a one-component-per-change-record format is shown on the following page.

```
Free-field format change record:

R=234$T=A$235#1200801976$237=61.0$239=V$


Change set created by EDIT-2 from above record:


8083C82258000132590811246470 1D234235AA00101200801976
8083C82258000232590811246470 1D234237AA620461.0
8083C82258000332590811246470 1D234239AA0001V
```

Each component in the input record is examined for adherance to the rules for editing and updating of the WATSTORE GWSI file. The components that do not meet the rules as described in the WATSTORE User's Guide are written in an error file and processing continues until an end-of-file marker is reached in the input data file. The general system logic for EDIT-2 is shown in figure 6. Components are written, one per record, to allow easy access to any individual component in the change file and to simplify editing and updating of each component.

Program Input

Input to EDIT-2 consists of a deck of control records or 80-position record images and the output file from EDIT-1, Control-002.

Control records.--A minimum of four control records are required for the operation of EDIT-2. The first control record must have CNL001 or CNL002 in positions 1 through 6. This first record determines the function performed by the program. The next three records, CNL010, CNL011, and CNL012, contain information used in titling the output and to aid in tracking the listings. Under Control-003, the program requires at least one additional record with a record locator on it. These locators may be individual sequence numbers or site identifiers, or they may be ranges of sequence numbers or site identifiers. The last control record should be an end-of-processsing indicator with 9's in positions 1 through 6.

Input data records.--Input records to EDIT-2 are contained in the change file output by the EDIT-1 program. Input data read by EDIT-2 has the following format:

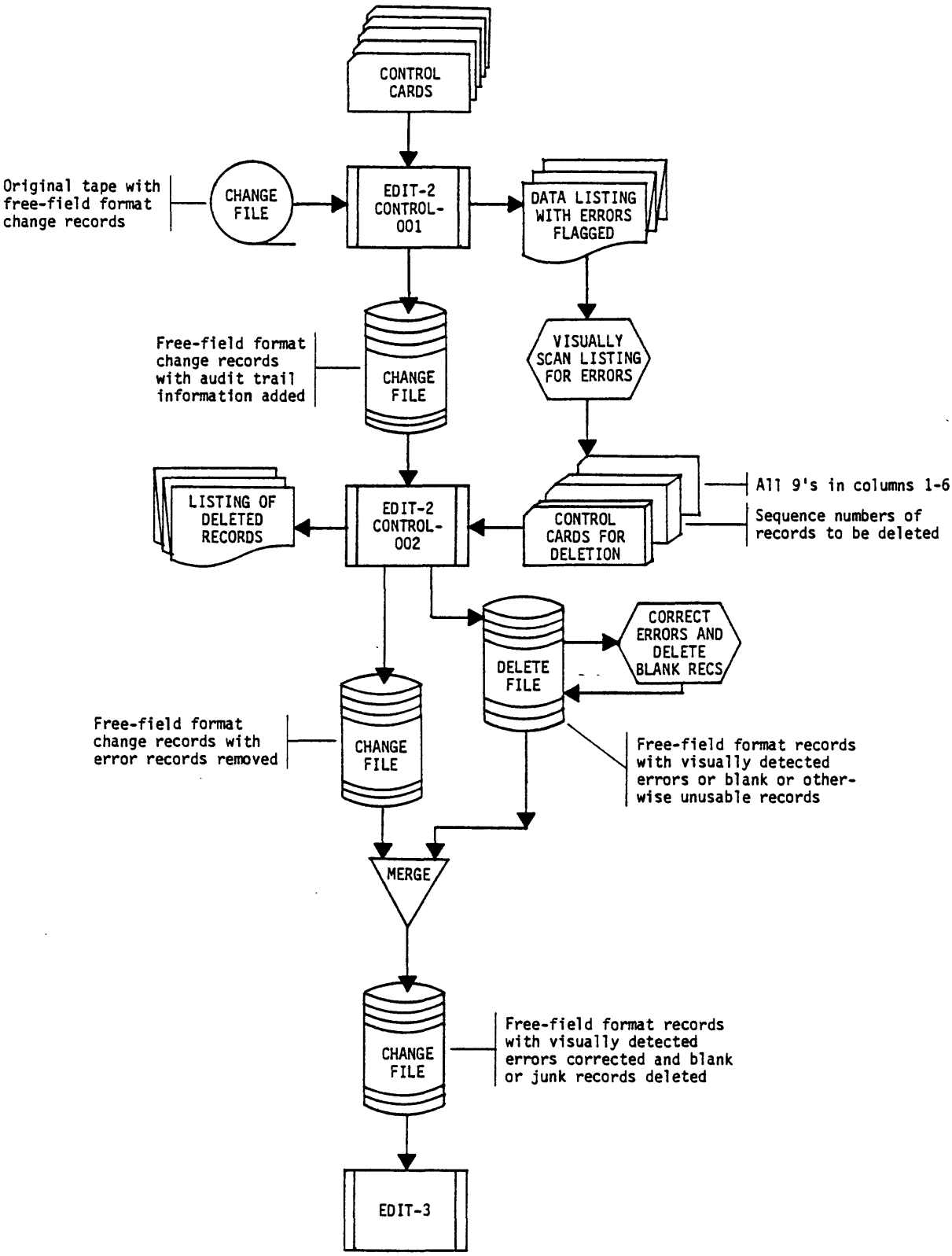


Figure 6.--General system logic for program EDIT-2.

Position	Format	Description
1- 6	X(6)	Volume serial number (VSN)
7-15	9(9)	Input sequence number
16-30	X(15)	Site-identification number
31-91	X(61)	Free-field change data
92-96	9(5)	Julian date

EDIT-2 reads the input change file of free-field format change records. Each input record is broken down into one or more records; each of these records contain the transaction type and data-component value for each WATSTORE type value pair in the input record. The program continues to add records to a change set being created from the input data until one or more of the following conditions are met: (1) a new site-identification number is read, (2) a new record-type identifier is read, or (3) a new transaction code is read. This process of converting free-field change records into change-record sets continues until an end-of-file marker is read by the program.

Program Output

Output records generated by EDIT-2 contain the following information:

—Original volume serial number
—Current Julian date
—Record's sequence number within its change-record set
—Record type that the component belongs to
—Component identifier
—Length of the change data in characters
—Component change value
—Output sequence number
—Transaction-type code
—Type-of-data code

Volume serial number.--This is the 1- to 7-character file identifier or volume serial number of the original input change file or tape.

Julian date.--The date on which the program was run is in Julian calendar format (yyddd).

Site-identification number.--This field is the 15-digit site-identification number from the input record.

Change-record set sequence number.--Each change-record set is composed of one or more change records. The record set has one change record for each component in the original input data record.

Record-type identifier.--The record-type identifier is three digits and is used in the WATSTORE GWSI file or assigned to a new record type by the user.

Component identifier.--The component identifier is a three-digit WATSTORE code that identifies a specific component field within a record type.

Data length.--This field is a count of the number of characters contained in the component-value field. It is used only for alpha and alphanumeric data.

Data-type code.--This two-digit code describes the type of data contained in the component-value field. All alpha and alphanumeric data have a data-type code of 99. Codes for numeric data are of the form nd, where n is the total number of digits which can be in the numeric item, and d is the number of digits which occur to the right of the decimal point. For example, a water-level measurement can have a length of six digits of which two digits are to the right of the decimal point. The data-type code for this water-level component is 62.

Component value.--This field is a 45-character table or array. It is processed as alphanumeric and contains as many characters as specified in the data length field for alpha and alphanumeric data. It contains as many digits as the n portion of the data-type field for numeric data. In the case of a component having no value, or having a missing value, the component value field will contain the word NULL.

Output Record Format

The output data file from EDIT-2 has the following format:

Position	Format	Description
1- 5	X(5)	Original file identification or VSN
6-10	9(5)	Julian date
11-13	999	Sequence number of record within a change set
14-28	9(15)	Site-identification number

29-31	999	Record-type identification number
32-34	999	Component identification number
35	X	Transaction code
36-37	99	Type of data
38-39	99	Length of data in characters
40-84	X(45)	Change data
85-90	9(5)	Output sequence number

Error Messages—Control-001

SKIPPING TO NEXT RECORD - FIRST CHARACTER NOT *R*

The first non-blank character in the input data field of the first change record for a new site or record type was not an 'R'. A new change record for a site must begin with 'R='.

RECORD FORMAT IN ERROR

The first non-blank character encountered after an 'R' is not an equal sign (=).

RECORD NUMBER NOT NUMERIC

One or more of the first three non-blank characters following an 'R=' is not a numeric character.

T NOT FOUND

The letter 'T' was not the first non-blank character encountered following an 'R=' value pair.

TRANSACTION FORMAT ERROR

The first non-blank character encountered following a 'T' is not an equal sign (=).

NO VALID TRANSACTION CODE

A valid transaction code (A, D, or M) was not the first non-blank character following a 'T=' value pair.

INCORRECT END OF TRANSACTION CODE

A character that is not a valid value-pair delimiter ($ or *) was encountered immediately following the transaction code.

1ST CHARACTER OF *C* NOT NUMERIC

The first character of a data-value pair was not a numeric character (0-9, =, -).

SKIPPING REST OF RECORD - TOO MANY *C* POSITIONS

A component identifier in a value pair had more than three digits. Check for a missing equal sign or pound sign (= or #).

NON-NUMERIC ENCOUNTERED - LOOKING FOR REST OF *C*

The second or third character of the component identifier of a data-value pair is not a numeric value (0-9, +, -)

COMPONENT DATA LONGER THAN 45 CHARACTERS

A component value was encountered that was longer than 45 characters. This length is an arbitrary value and may be changed to suit your needs. CAUTION: If the length is changed in EDIT-2, it must be changed in all programs that follow EDIT-2.

LOCAL WELL NUMBER DATA FIELD STARTS WITH OTHER THAN '

The local well identification number (C=12) must be enclosed in single quotes.

END OF DATA CHARACTER MISSING

A data value-pair delimiter ($ or *) was not found before the next record identifier was encountered.

Error Messages—Control-002

760M2 NO DATA INPUT

An end-of-file marker was encountered on the first attempt to read from the input data file.

761M2 END OF RUN

An end-of-file marker was encountered while attempting to read a control record image. This is normal end of processing.

763M2 MULTIPLE RANGE PARAMETERS

All values on the locator control record are zero or a blank record image has been read.

764M2 SEQUENCE COUNT WITH OTHER VALUES

An ending sequence number has been read from a range control record and a value was encountered in the 'NUMSTR' field of the same range control record.

765M2 SITE ID RANGE WITH OTHER VALUES

Both a range of site identifiers AND a range of record numbers have been specified for retrieval by the control record read by EDIT-2. Select either a range of record sequence numbers or a range of site identifiers, BUT NOT BOTH.

766M2 SEQUENCE COUNT INVALID END

A range of change-record sequence numbers has been read by the program for retrieval. The beginning value of the range to be retrieved is larger than the ending sequence number. Change the range numbers on the control record so that the beginning sequence number is less than the ending sequence number.

767M2 SITE ID RANGE INVALID - END LOW

The ending site identifier of the specified range of change records to be retrieved is larger than the beginning site identifier specified on the control record. Since the change file is sorted in descending order on the site-identifier field, the beginning site identifier on the retrieval record must be larger than the ending site identifier.

768M2 NO RECORDS PRINTED NUMFST = nnnnn
CNRECN = mmmmm

The beginning sequence number read from the retrieval control record is greater than the number of records contained in the change record file. Correct the range of records to be retrieved on the control record or check to see that you have the correct change file attached to the program.

769M2 END OF DATA - REQUEST INCOMPLETE

The ending record sequence number of the range of records to be retrieved from the change file is larger than the number of records in the change file. Check to see that you have the correct change file attached to the program or change the ending value in your retrieval request.

Component-Value Edit Program—EDIT-3

Program EDIT-3 performs editing of the component-value field to insure that reasonable and (or) valid data values are being entered into the master data file. The general system flow chart describing EDIT-3 is included as figure 7.

EDIT-3 reads a complete set of change records for a particular site and record type. This change set is stored in a table so that all the components contained in the change set can be examined. If the change set represents a new record type to be added to a given site or the addition of a new site, each of the component identifiers is examined to insure that all mandatory components are present in the change set. Mandatory components are checked in the paragraphs "STA000" through "STA320" in the Procedure Division of EDIT-3. A component's status as mandatory or optional may be changed by adding or deleting the checks in the appropriate "STAnnn" paragraph, where the "nnn" corresponds to the record-type identifier. Each component value is edited to insure that only reasonable values or valid codes are entered into the master file. If the change set represents a change to data values already in the file, each component is edited for reasonable data values and valid codes.

EDIT-3 conforms to all edits as described in the WATSTORE User's Guide, Volume II, Chapter 2. If no errors are encountered in the editing of a change set, the entire change set is written to the output-update file; otherwise, the entire change set is written to an error file for correction and re-editing.

Program Input

Input to the program consists of the output changes from the component-field format-edit program, EDIT-2. EDIT-3 has only one function, and therefore no control records are required as input. Input-data records to EDIT-3 have the following format:

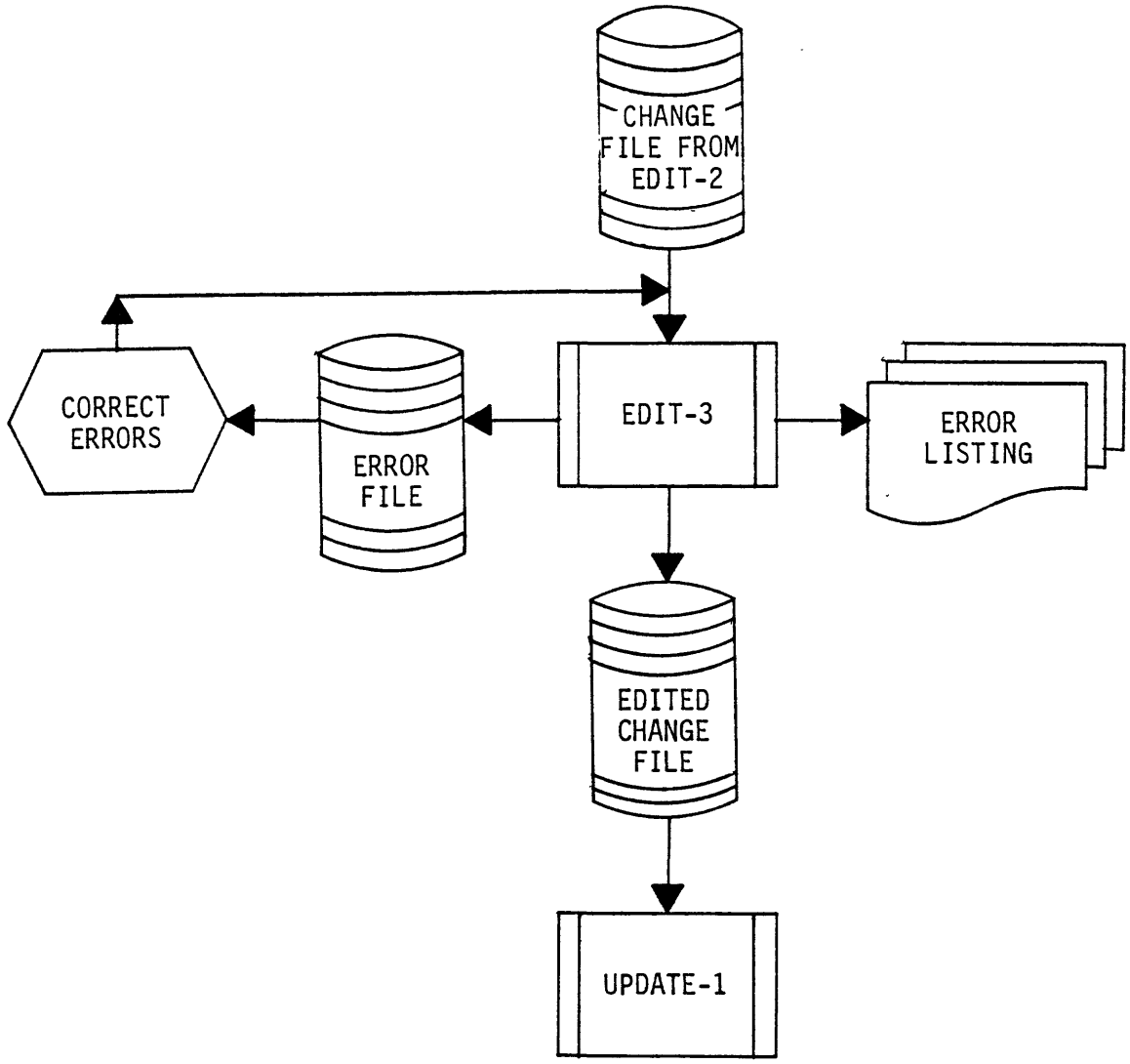


Figure 7.--General system logic for program EDIT-3.

Position	Format	Description
1- 5	X(5)	Original file identification or VSN
6-10	9(5)	Julian date
11-13	999	Sequence number of record within a change set
14-28	9(15)	Site-identification number
29-31	999	Record-type identification number
32-34	999	Component identification number
35	X	Transaction code
36-37	99	Type of data
38-39	99	Length of data in characters
40-84	X(45)	Change data
85-90	9(5)	Output sequence number

Program Output

Output from EDIT-3 is unchanged from the input records in terms of record format. Output from EDIT-3 consists of an output-update change file, an error file, and a printed listing of all input change sets that contain errors.

Update change file.--The output-update change file contains change sets in which every member record has passed the value edits for its particular component value.

Output error file.--If an error is encountered in the component value of any member of a change set, the entire change set is written to the error file. Editing of component values continues through the entire change set even though an error may have been detected in a previous member record of the change set.

Error Messages—Edit Errors

INVALID RECORD NUMBER - RNUM = nnn

A record number has been encountered that is neither a valid WATSTORE identifier nor a valid user-defined record identifier. Correct the record number or insert a new record description in the program with tests for the new record identification in paragraphs 'EXAM-CHANGE' and 'DIST-CHECK'.

REQUIRED DATA FIELD MISSING

A mandatory component field is missing from the change record set.

CHANGE > 45 CHARACTERS

The DLEN field of a change record has a value greater than 45. This message is a warning only. The update programs will only insert the first 45 characters into the updated record component. If characters are present beyond the 45-character length limit or the length defined for the component in the update programs, they will be truncated.

DISTRICT CODE IS NOT '04' - AZ

The value for the district-code component (C=6) is not '04'.

INVALID ALPHA CODE IN CHANGE

An alphanumeric character was encountered in a numeric data item.

STATE CODE NOT '04' - AZ

The value for the district-code component (C=6) is not '04'.

COUNTY CODE IS NOT VALID CODE

INVALID QUAD IN LOCAL ID

The first character of the local identification number is not 'A', 'B', 'C', or 'D'.

INVALID LOCAL IDENTIFIER

ALTITUDE OUTSIDE ACCEPTABLE RANGE - 0:12633

DATE IS NOT NUMERIC VALUE

DEPTH OF HOLE IS NOT VALID

A depth-of-hole value was encountered that is negative or greater than 3,000 feet or less than depth of well.

DEPTH OF WELL IS NOT VALID

A depth-of-well value was encountered that is negative or greater than 3,000 feet or greater than depth of hole.

ENTRY NUMBER IS NOT NUMERIC

An entry number used to identify multiple entries of the same record type for a given site has a non-numeric character.

INVALID LITHOLOGIC CODE

An invalid lithologic code has been encountered that is not identified in the data description—'GEO-CODES'.

SOURCE AGENCY NOT VALID

An invalid source-agency code has been encountered.

DATE IS LATER THAN TODAY'S DATE

A date has been found that places the value at some time in the future.

MANDATORY FIELD IS BLANK

A mandatory component field is null or has all blank characters.

QW CODE NOT VALID STORET CODE

A field water-quality code has been found that is not defined in the QW-CODES data description in EDIT-3.

Error Messages—Programmed Stops

STOP5 FILEA-UPCHNG EMPTY
*** CHECK FOR ATTACH OF CHANGE FILE IN JCL***

The change file was in an end-of-file condition on the first attempt to read it.

ERROR IN RECORD SEQUENCE STOP8 AZERRS
KTEST = KEYIDD =

The input change file has at least one record out of sequence. The change file should have the same sort order (descending site identification) as the master data file.

Master File Update Preparation Program—UPDATE-1

This program separates the master ground-water data file into two files. One file contains the records for sites not having changes in the update change file; the other contains all records for each site having changes in the update change file (fig. 8).

UPDATE-1 reads the update change file and the master file. If the site-identification number of the change record matches the site-identification number of the master file record, all the records with the same site identifier are written from the master file to a change volume of the master file. The change file is then read until a new site identifier is encountered; the master file is read until the new site identifier is either located or not found. A site is considered 'not found' when the site identifier read from the master file is less than the site identifier of the current change record. If no change records are in the change file for a given site, the master file records for that site are written to an unchanged volume of the master file. These two volumes of the master file will be merged after the updating of the master file change volume is completed.

Program Input

Input to UPDATE-1 consists of the current volume of the master ground-water data file and the update change file that was output by the component edit program, EDIT-3.

Program Output

Two master file volumes are created by UPDATE-1. These two files contain variable-length records of the same format and description as discussed in the section entitled "Master Data File." Appendix A contains the COBOL record descriptions of the WATSTORE record types contained in the master file. By splitting the master file into two files, only those sites for which change records are present in the update change file will be accessed by the final update program. This reduces access time and processing costs and helps to maintain the integrity of the data for those sites not being changed.

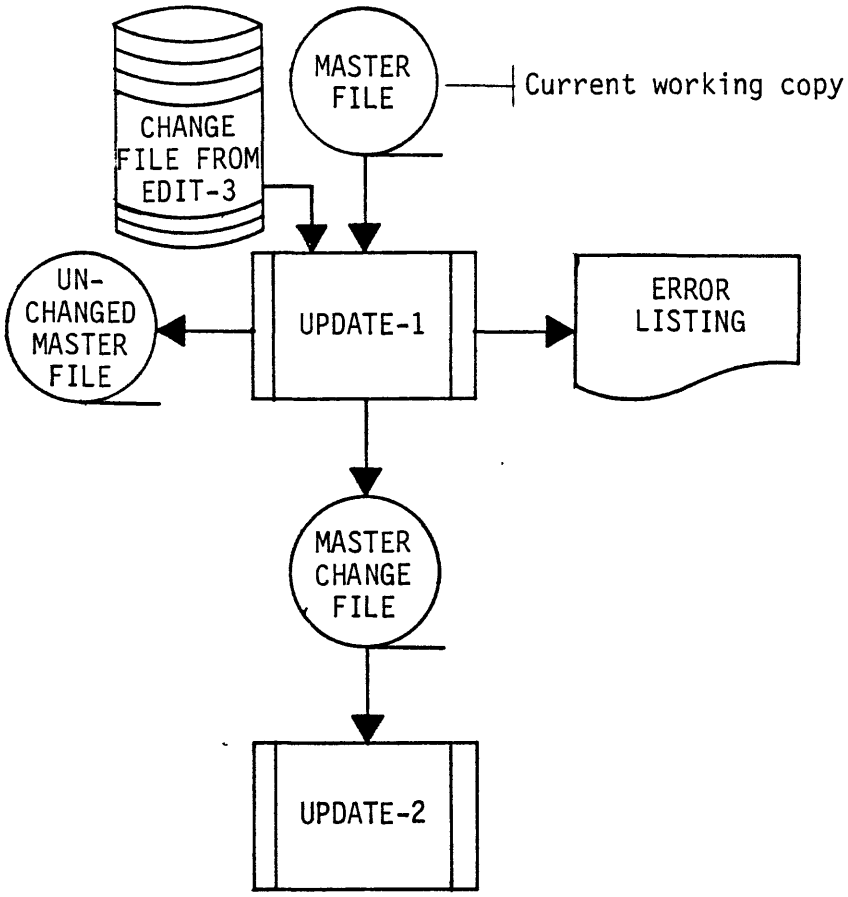


Figure 8.--General system logic for program UPDATE-1.

Master File Update Program—UPDATE-2

This program makes the changes indicated in the update change file to the records contained in the change volume of the master ground-water data file. Figure 9 shows the general system flow chart for UPDATE-2.

UPDATE-2 reads the update data file and the change master file from UPDATE-1. Each record in the change master file for which there is a change-record set in the update data file is updated and then written to an updated master volume.

Valid Update Transactions

The program recognizes three transaction codes as valid: A)dd, M)odify, and D)elete.

A)dd transaction.--A new site will be added only when an R=000 record for that site does not exist in the master change file and the first record for that site in the update change file is an R=000 record.

A new record will be added for a site only when an R=000 record exists in the master change file and no record with the same record-type identifier exists except where multiple records of the same type are allowed.

A component field will be added only when an R=000 record and the record owning the component to be added already exist in the master change file. The component value in the owning record must be a null value in order to add the component.

M)odify transaction.--Records and (or) components will be modified only where all prerequisites for modification exist. The prerequisites for modifying the contents of a given record or component are described in detail in the WATSTORE User's Guide, Volume II, Chapter 2.

D)elete transaction.--A site will be deleted only if an R=000 record exists in the master change file. If the delete is valid, all records for that site will be deleted from the master file.

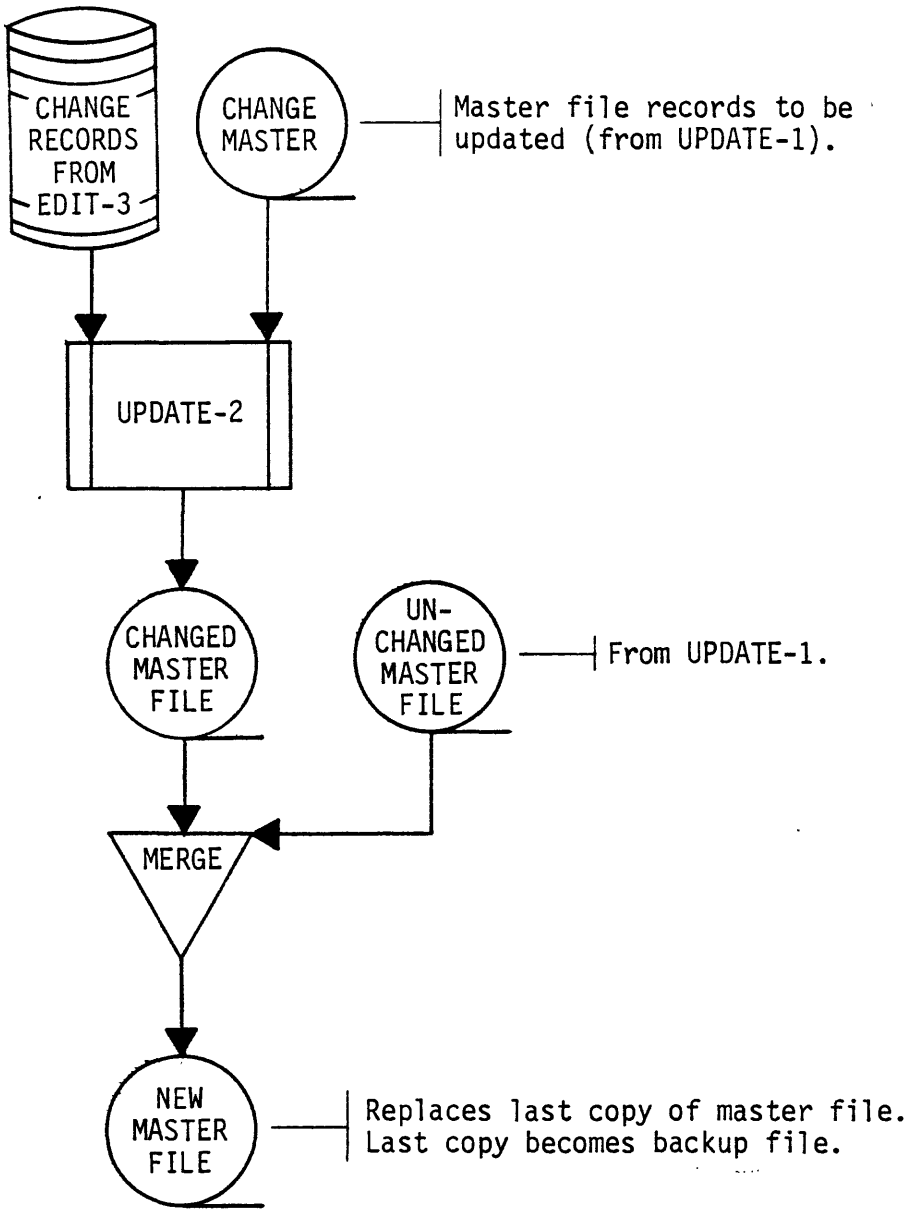


Figure 9.--General system logic for program UPDATE-2.

A record will be deleted only where the site has an R=000 record in the master change file and the record to be deleted is present in the master change file. If other records have any dependency upon the record being deleted, those dependent records also will be deleted.

A component can be deleted only when an R=000 record exists for the site, the record that owns the component exists, and the component to be deleted has a value other than null. Deleting a component on which other records or components are dependent will cause those records or components to be deleted also.

Null values.--A component change record which contains the word NULL in the data field, causes the program to assign the COBOL literal 'Low-Values' (000 octal) to the component field in the master file record.

Program Input

Input to UPDATE-2 consists of the update change file from EDIT-3 and the master change file ouput by the master file preparation program, UPDATE-1.

Program Output

The program's output consists of an updated master change file and an error file. A listing is produced of the errors with appropriate error messages.

Updated master ground-water data file.--This file contains all the records from the input change master file and the valid changes made to them except for the sites, records, and component fields that have been deleted.

Error output file.--The error file contains all change-record sets and (or) individual change records that contain invalid transactions or transactions that cannot be completed because modification prerequisites were not present. These errors are noted on the output listing and may be rerun with the program after corrections have been made to them.

PROGRAMS FOR DATA RETRIEVAL AND REPORTING

Ground-Water Site Inventory (GWSI) Data Retrieval Program—BASIN-3

BASIN-3 is a COBOL program designed to provide an inventory of selected data for all ground-water sites in a given geographic area. Data are retrieved and stored for each site located within the boundaries of a given polygon. BASIN-3 reads consecutive slices of the input polygon and the input data file. If the location of the site read falls within the boundaries of the polygon, data for that site are written to the output files. If the site location falls below the current slice, the program reads the next slice of the polygon. The program stops when all input data have been read, or a record is read whose site location falls below the lowest point of the polygon. The general system logic is shown in figure 10.

Program Input

Input to the program consists of two control records, a polygon-description file, and a ground-water data file.

Control records.--The program is initially controlled by two user-supplied control records. The control records required for the operation of BASIN-3 are control records 001 and 010.

Polygon-description file.--The polygon-description file is read from a disk file by the program. This disk file is created by the polygon-definition program, POLYGON, and contains the polygon-definition table.

Ground-water data file.--Data records input to BASIN-3 may be either the variable-length records of the master file or the 270-character records of the working ground-water data file. The length of the variable portion of the record is defined by LNUM (positions 19-21) of each record. The format of the variable portion is defined by RNUM and a detailed description of each WATSTORE record type can be found in the WATSTORE User's Guide, Volume 2, Chapter II, pages B14 through B97.1.

Program Output

BASIN-3 produces two output files: (1) a printed report of available site data, which is shown in the following example, and (2) a disk file of the available site data for use with other programs.

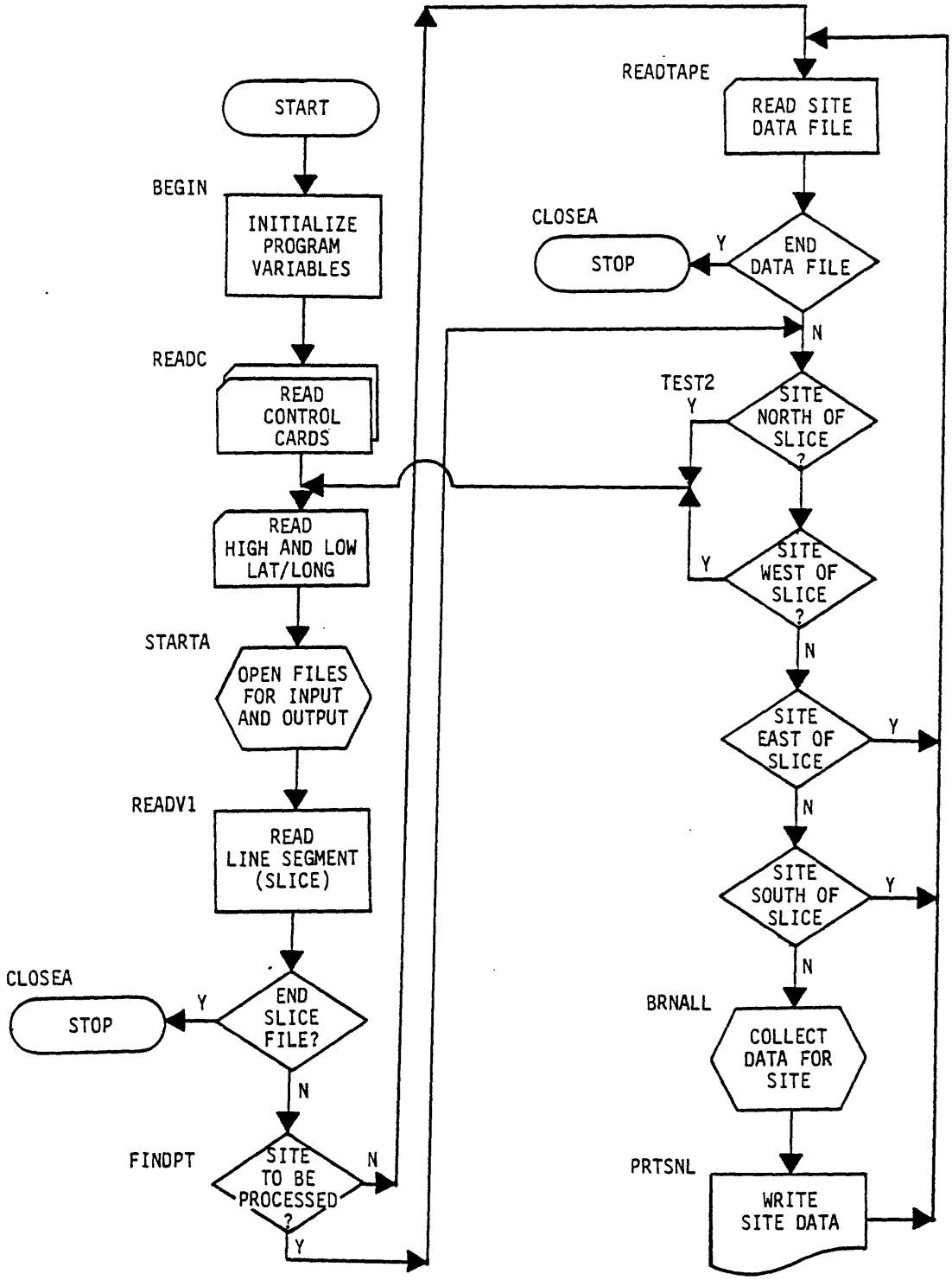


Figure 10.--General system logic for program BASIN-3.

REC NUM	SITE IDENTIFICATION LAT	LONG	LOCAL ID	DEPTH MEAS	SRC	ALT	CONST YEAR	LOG TYPES	WATER LEVELS MEAS	YR	MTH	DISCHRG GPM	YR	USE	OPENINGS TOP	TYP	CASINGS DIAM	F	RECORD CNT WL	Q1	Q2	Q3
1	324634	1110621 01 D-07-12 33D		550		2780	1969	G						U					0	0	0	0
2	324545	1111935 01 D-08-10 04BBB2		600		1979				1970				H					12	1	1	13
3	324543	1111937 01 D-08-10 04RBB1		580		1935			293	1948	S			Z					0	1	1	8
4	324520	1111257 01 D-08-11 09ABC		1470		2293	1969	G	632	1969	S			N	600.00	X	3.00	X	0	0	0	0
5	324507	1110133 01 D-08-13 08BD02		570		3245			13	1952	S			S					0	1	1	5
6	324507	1110132 01 D-08-13 08BDD1		250		3245	1952		9	1952	S			S			10.00		0	0	0	0
7	324459	1110814 01 D-08-12 07DAA		717		2675	1969	EJD				2		U	113.00	P	12.00	P	0	0	0	0
8	324442	1110705 01 D-08-12 09CC		536		2769			439	1965	S								0	0	0	0
9	324431	1105725 01 D-09-14 30CCC		350		3440	1972	D	280	1972	R				280.00	P	0.00	P	0	0	0	0
10	324350	1105923 01 D-08-13 15DCC				3580			50	1941	S			S					0	1	1	9
11	324312	1110511 01 D-08-12 22DAA				2910			136	1972	S			S					0	0	0	0
12	324304	1110821 01 D-08-12 19DDC		38		2640				1970	S			H					33	1	1	30
13	324258	1105544 01 D-08-14 20CCC		248		3990	1966	D	177	1966	R				155.00	P	8.00	P	0	0	0	0
14	324236	1110522 01 A-15-12 09RBB		778		6440	1947	D	628	1966	S			S	13.00	X	8.00	X	0	1	1	10
15	324224	1105741 01 D-08-13 25CBA		400		3690	1966	D	240	1966	P			S	250.00	P	6.00	P	0	1	1	8
16	324222	1105744 01 D-08-13 25CBB				3510			295	1972	S			S					0	0	0	0
17	324212	1110520 01 D-08-12 27DDB1				2960			30	1972	E								0	0	0	0
18	324201	1110312 01 D-08-12 25DDC				3090			32	1972	S			S					0	0	0	0
19	324157	1111531 01 D-08-11 31RBB		936		2105	1963	GD		1970	S			U	440.00	P	14.00	P	8	1	1	24
20	324138	1110501 01 D-08-12 35BCC				2920			12	1972	S			U					0	1	1	8
21	324135	1110848 01 D-09-12 31CAA		40		2615			49	1952	S			S					0	1	1	5
22	324130	1110000 01 D-08-13 33DAA		330	D	3400	1975	D							210.00	P	4.50	P	0	0	0	0
23	324105	1110506 01 D-09-12 03AAA		85		2925			16	1967	S			U					0	0	0	0
24	324104	1110531 01 D-09-12 03ABB		275		2895	1974	D						H			8.00		0	0	0	0
25	324104	1110320 01 D-09-12 01ABA		178		3100	1978	D									8.00		0	0	0	0
26	324056	1110509 01 D-09-12 03ADA				2930	1972		20	1972	S			H			8.00		0	0	0	0
27	324055	1111754 01 D-08-10 34DBA		483		1970		J	290		S								0	0	0	0
28	324046	1110547 01 D-09-12 03CAB				2840	1972		31		R			H					0	0	0	0
29	324045	1110536 01 D-09-12 03BDD				2880	1972												0	0	0	0
30	324039	1110536 01 D-09-12 03CAA		300		2880	1971	D	27	1971	P						8.00	P	0	0	0	0
31	324038	1105853 01 D-09-13 02CBB		100		3480	1971	D	30	1971	R				20.00	P	8.00	P	0	0	0	0
32	324037	1110743 01 D-09-13 05DAA				3480								S					0	0	0	0
33	324024	1110338 01 D-09-12 01CD9		34		3040			17	1972	S			H					0	1	1	8
34	324019	1110306 01 D-09-12 01DDD1		21		3102			19	1952	S			S					0	0	0	0
35	323952	1110214 01 D-09-13 07DBB		40		3160			25	1952	S			S					0	0	0	0
36	323949	1110234 01 D-09-13 07BDD		100		3160			46	1952	S			S					0	0	0	0
37	323942	1110302 01 D-09-13 07CBC		275		3115				1978	S			S					32	1	1	5
38	323912	1112402 01 D-09-09 10DBA		278		1776			230	1953	S			U					0	0	0	0
39	323909	1110203 01 D-09-13 18ADA		270		3205								H					0	0	0	0
40	323856	1110246 01 D-09-13 18CAB		40		3110			16	1967	S			S					0	0	0	0
41	323854	1110759 01 D-09-12 17CBA				2740			43	1952	S			S					0	1	1	8
42	323853	1110431 01 D-09-13 14DBB		500		3400								S					0	1	1	5
43	323853	1109841 01 D-09-13 14CBA		520		3390			395	1972	S			S					0	0	0	0
44	323842	1112317 01 D-09-09 14BAA		280		1796			164	1931	S								0	0	0	0
45	323836	1112318 01 D-09-09 14BAD		600		1798	1955	G	215	1964	S								0	0	0	0

Site-inventory report.--The site-data listing contains all the available data for the parameters for each site located within the boundaries of the polygon read by the program. The last page of the well-site report (table 4) lists all the valid codes and their meanings for the following parameters: (1) source-of-depth data, (2) methods of measurement, (3) log types, (4) use of water, (5) type of finish, and (6) type of opening.

Site-data output file.--The data file written to disk storage for use by other programs is a compact version of the printed report. The records in this file are 105 characters in length and have the following format:

Position	Variable	Format	Description
1- 4	CNRECR	9(4)	Sequence number
5- 19	KEYID	X(15)	Site identifier
20- 39	LCWLID	X(20)	Local site identifier
40- 44	DDEPTH	9(5)	Depth of well
45	SDEPTH	X	Source of depth data
46- 50	ALTUDE	9(5)	Altitude of site above or below sea level

51- 55	PPDISH	9(5)	Rate of discharge
56- 59	PPYEAR	9(5)	Year of discharge measurement
60- 63	FCYEAR	X(4)	Year construction was completed
64	WTRUSE	X	Primary use of water
65- 72	INLOGS	X(8)	Available log types
73- 79	OPNTOP	9(5)V99	Depth to top of the open section
80	OPNTYP	X	Type of opening
81- 85	CDIAM	999V99	Diameter of casing
86	FINISH	X	Type of finish
87- 92	WTRLVL	-9(4)V99	Most recent water-level measurement
93- 96	WLYEAR	X(4)	Year of most recent water-level measurement
97- 99	WLCNT	999	Number of water-level measurement records available
100-105	SPCAP	9(4)V99	Specific capacity

Error Messages

BSN3 CONTROL INPUT FILE EMPTY

Indicates that no data are present in the input stream on the first read attempt.

BSN3 ID MISSING OR INCORRECT IN CONTROL RECORD

A control record has been read that has an identification that was not recognized by the program. Check the control records for one of the following conditions:

—Control records not in required sequence

—Control record(s) not present

—'CNL' is not in positions 1-3

STOP4 SOURCE ID RECORDS MISSING

An end-of-file marker was encountered while trying to read the second control record. Check to insure that both control records are present in the input job stream.

Table 4. Parameter codes and their meaning as printed by the BASIN-3 program

DEPTH SOURCE		WATER LEVELS MEAS METHOD[MTH]		AVAILABLE LOG TYPES		USE OF WATER		TYPE OPENINGS		FINISH [F]	
S	RPTING AGENCY	A	AIRLINE	A	TIME	A	AIR CONDITIONING	F	FRACTURE	C	POROUS,CONCRETE
D	DRILLER	C	CAL AIRLINE	B	COLLAR	B	BOTTLING	L	LOUVERED	F	GRAVEL W/PERF
O	OWNER	E	ESTIMATED	C	CALIPER	C	COMMERCIAL	M	MESH	G	GRAVEL,SCREEN
A	OTHER GOV'T	G	PRESSURE GAGE	D	DRILLER'S	D	DEWATER	P	PERFORATED	H	HORIZ GALLERY
R	OTHER REPORTED	H	CAL PRESSURE GAGE	E	ELECTRIC	E	POWER	R	WIRE WOUND	O	OPEN END
L	LOGS	L	GEOPHYSICAL LOGS	F	FLUID CONDUCT	F	FIRE	S	SCREEN(UNK)	P	PERFORATED
G	GEOLOGIST	M	MANOMETER	G	GEOLOGIST	H	DOMESTIC	T	SAND,POINT	S	SCREEN
Z	OTHER	R	REPORTED	H	MAGNETIC	I	IRRIGATION	W	WALLED	T	SAND,POINT
		S	STEEL TAPE	I	INDUCTION	J	INDUSTRIAL(COOLING)	X	OPEN HOLE	W	WALLED
		T	ELECTRIC TAPE	J	GAMMA RAY	K	MINING	Z	OTHER	X	OPEN HOLE
		Y	CAL ELECTRIC TAPE	K	DIPMETER	M	MEDICINAL			Z	OTHER
		Z	OTHER	L	LATERLOG	N	INDUSTRIAL				
				M	MICROLOG	P	PUBLIC SUPPLY				
				N	NEUTRON	Q	AQUACULTURE				
				O	LATER	R	RECREATION				
				P	PHOTO	S	STOCK				
				Q	RADIOACTIVE	T	INSTITUTION				
				S	SONIC	U	UNUSED				
				T	TEMP	Y	DESALINATION				
				U	GAMMA-GAMMA	Z	OTHER				
				V	FLUID VELOCITY						
				Z	OTHER						

STOP5 DATA INPUT FILE END INCORRECT
EOF ENCOUNTERED AT READB OR READV

An end-of-file marker was encountered while reading the polygon-description file before an end-of-polygon indicator was read from the file. Check the last record in the polygon-definition table to see that it contains 9's in positions 1-15.

STOP6 SECOND RECORD OF PAIR MISSING

Each slice in the polygon-definition table is defined by at least two line-segment records. A slice may be defined by more than two records, but the records are always some multiple of two. If this error occurs, re-run the polygon-definition program, PLGN01.

STOP7 FIRST RECORD NOT '000' OR '001'

The first record encountered for a site was neither a GWSI-type header record nor a water-quality type header record.

STOP8 FILE OUT OF SEQUENCE

The data file being read has a record out of sequence. The data file must be sorted in descending order by the site identifier.

STOP9 RECORD TYPE NOT FOUND

A record was read from the data file that was neither a WATSTORE record type nor a user-defined record type. Insert a record description of the record into the program and add a test for the record type number in {paragraphs}.

Water-Level History Program—BASIN-4

BASIN-4 reads a ground-water data file or water-level working file and a polygon-description file to produce a water-level history for all sites that fall within the given polygon. The general system logic of BASIN-4 is shown in figure 11.

BASIN-4 reads consecutive slices of the input polygon and the input water-level file. If the location of the site read falls within the

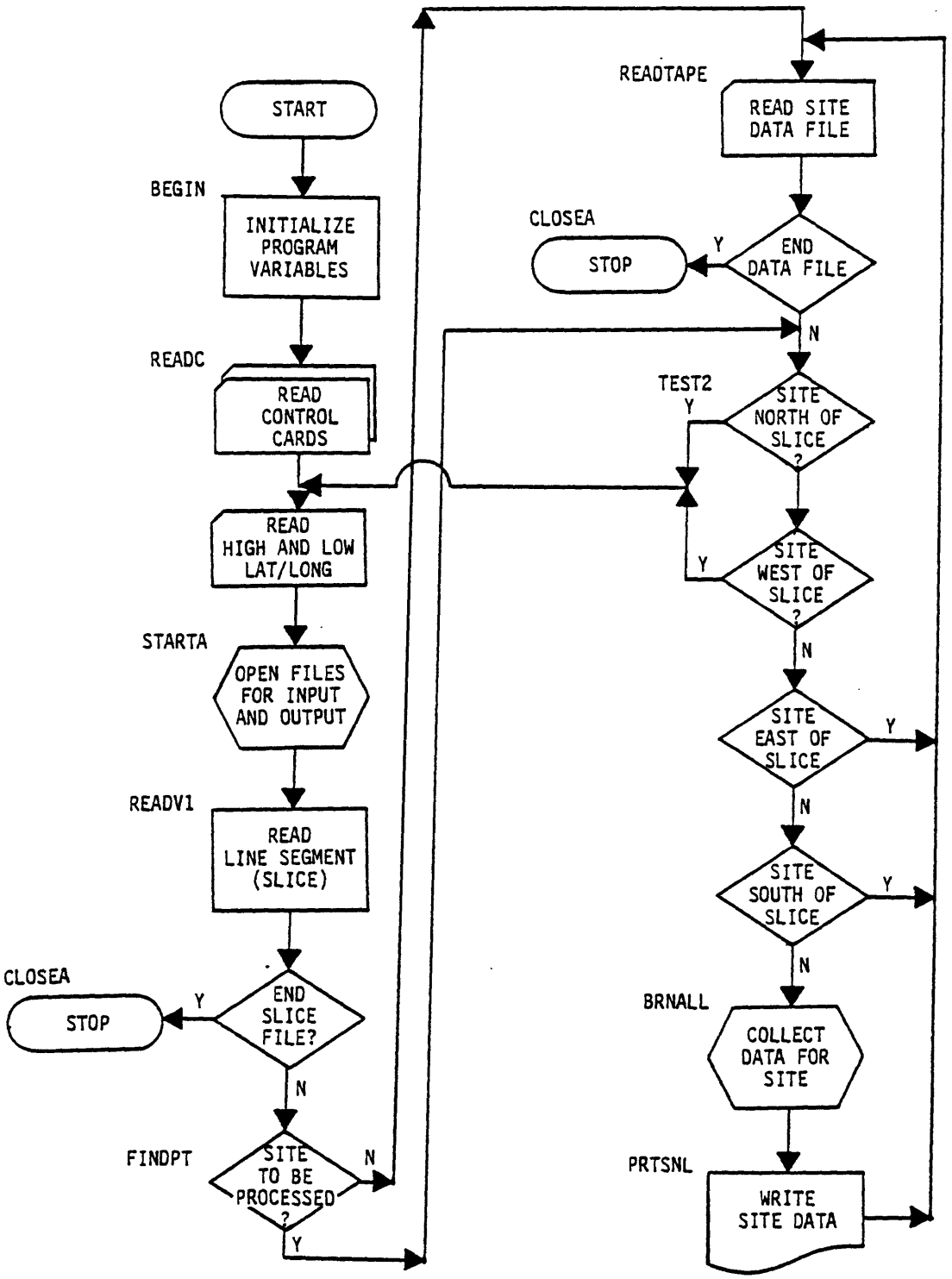


Figure 11.--General system logic for program BASIN-4.

boundaries of the polygon, all water levels for that site are written to the output files. If the site location falls below the current slice, the program reads the next slice of the polygon. The program stops when all water-level data has been read or a water-level record is read for which the site location falls below the lowest point of the polygon.

Program Input

The program requires the following input data: (1) two control records, (2) a polygon-description file, and (3) a water-level working file or ground-water data file.

Control records.--The program is initially controlled by two user-supplied control records. The control records required for the operation of BASIN-3 are control records 001 and 010.

Polygon-description file.--The polygon-description file is read from a disk file by the program. This disk file is created by the polygon-definition program, POLYGON, and contains the polygon description as described in the section covering POLYGON in this report.

Water-level data file.--Water levels may be read from either the ground-water master file or from a water-level working file. In either case, the water-level records have the following format:

Position	Format	WATSTORE code	Description
1- 15	9(15)	C1	15-digit site-identification number
16- 18	999		Record-type identification number
19- 21	999		Length of variable portion
22	X		Data security or access code
23- 30	X(8)	C235	Date on which the water level was measured
31	X	C236	Accuracy of the date
32- 37	9(4)V99	C237	Water-level measurement
38	X	C238	Site status at time of measurement
39	X	C239	Method of measurement
40	X	C240	Accuracy of the measurement

Program Output

Output from the program consists of a printed water-level history for each site within the polygon and a compressed water-level data file that can be used as input to other programs, such as plot programs.

A water-level history for a given site includes the following data:

—Output sequence number of the selected site within this particular retrieval

—Site-identification number

—Local site-identification number or site name

—Site type, such as well, spring

—Altitude of the site

—Depth of the well

—All water-level measurements available for the site

—Date of each water-level measurement

The output water-level data file contains the same information as the printed listing, plus a two-digit record identifier that is used by a hydrograph program included as part of a ground-water modeling package used by the Swab/RASA study. Format of the output water-level data record is:

Position	Format	Description
1- 2	99	Record identifier for hydrograph program, must be '88'
3- 6	9(4)	Sequence number of the site
7-10	X(4)	Blank
11-25	9(15)	Site-identification number
26-45	X(20)	Local site identifier
46-50	9(5)	Altitude of the site
51-55	9(5)	Depth of well below land surface
56-62	9(4)V99	Water-level measurement
63-70	9(8)	Date of measurement

The water-level history report printed by the BASIN-4 program is of the following type:

REC NUM	SITE IDENTIFIER LAT	LONG	DUP	LOCAL SITE IDENTIFIER	SITE TYPE	ALTI-TUDE	WELL DEPTH	WATER LEVEL HISTORY LEVEL	DATE	LEVEL	DATE	LEVEL	DATE
1	324545	1111935	01	D-08-10 04BABP		1939		309.	3/ 4/1965	309.3	3/ 4/1965	310.5	2/23/1966
								307.	2/21/1967	316.9	5/ 2/1968	318.8	1/16/1970
								327.6	12/29/1970	324.9	2/18/1972	324.2	2/ 7/1973
								324.3	1/30/1974	327.8	3/11/1975	329.8	1/22/1976
								330.5	11/23/1976				
2	324543	1111937	01	D-08-10 04BABB1		1935		293.	11/12/1948				
3	324520	1111257	01	D-08-11 09ABC		2293		632.	4/18/1969				
4	324507	1110133	01	D-08-13 08BDD2		3245		13.	10/15/1952				
5	324507	1110132	01	D-08-13 08BDD1		3245		9.	10/15/1952				
6	324442	1110705	01	D-08-12 09CC		2769		439.	5/ 1/1965				
7	324431	1105729	01	D-19-14 30CCC		3440		280.	9/ 1/1972				
8	324350	1105923	01	D-08-13 15DCC		3580		50.	11/ 1/1941				
9	324312	1110511	01	D-08-12 22DAA		2910		136.	11/ 1/1972				
10	324304	1110821	01	D-08-12 19DDC		2680		14.	10/ 7/1952	14.1	10/ 7/1952	14.3	1/ 8/1953
								16.5	7/ 7/1953	13.2	12/ 9/1953	14.8	2/25/1954
								13.7	5/13/1954	5.7	8/19/1954	12.	12/ 6/1954
								8.5	9/23/1955	16.9	9/25/1956	9.2	9/13/1957
								5.6	3/26/1958	15.7	2/ 4/1959	5.	2/10/1960
								15.7	2/ 8/1961	9.3	3/12/1962	13.5	3/10/1963
								13.1	3/ 4/1964	6.1	3/ 3/1965	4.	2/25/1966
								12.4	2/21/1967	6.7	5/ 2/1968	13.2	8/29/1969
								10.9	1/16/1970	11.6	12/30/1970	6.1	2/17/1972
								6.1	2/17/1972	5.6	2/ 7/1973	12.	1/30/1974
								9.9	3/10/1975	8.2	1/19/1976	6.6	11/23/1976
								12.5	12/12/1977				
11	324258	1105544	01	D-08-14 20CCC		3990		177.	12/ 1/1966				
12	324236	1110527	01	A-19-12 09BBB		6440		428.	12/ 1/1966				
13	324224	1105741	01	D-08-13 25CBA		3490		240.	12/ 1/1966				
14	324222	1105744	01	D-08-13 25CBB		3510		295.	11/ 1/1972				
15	324212	1110520	01	D-08-12 27DDB1		2960		30.	11/10/1972				
16	324201	1110317	01	D-08-12 25DDC		3090		32.	11/10/1972				
17	324157	1111531	01	D-08-11 31BAB		2105		438.	5/10/1968	438.3	5/10/1968	442.1	5/ 1/1969
								437.9	1/16/1970	437.8	12/30/1970	439.1	5/10/1975
								431.3	1/22/1976	437.3	11/23/1976	441.	12/12/1977

Error Messages

BSN4 CONTROL INPUT FILE EMPTY

Indicates that no data are present in the input stream on the first read attempt.

BSN4 ID MISSING OR INCORRECT IN CONTROL RECORD

A control record has been read that has an identification that was not recognized by the program. Check the control records for one of the following conditions:

—Control records not in required sequence
—Control record(s) not present
—'CNL' is not in positions 1-3

STOP4 SOURCE ID RECORDS MISSING

An end-of-file marker was encountered while trying to read the second control record. Check to insure that both control records are present in the input job stream.

STOP5 DATA INPUT FILE END INCORRECT
EOF ENCOUNTERED AT READB OR READV

An end-of-file marker was encountered while reading the polygon-description file before an end-of-polygon indicator was read from the file. Check the last record in the polygon description to see that it contains 9's in positions 1-15.

STOP6 SECOND RECORD OF PAIR MISSING

Each slice in the polygon description is defined by at least two line-segment records. A slice may be defined by more than two records, but the records are always some multiple of two. If this error occurs, re-run the polygon-definition program, PLGN01.

STOP7 FIRST RECORD NOT '000' OR '001'

The first record encountered for a site was neither a GWSI-type header record nor a water-quality type header record.

STOP8 FILE OUT OF SEQUENCE

The data file being read has a record out of sequence. The data file must be sorted in descending order by the site identifier.

STOP9 RECORD TYPE NOT FOUND

A record was read from the data file that was neither a WATSTORE record type nor a user-defined record type.

Areal Plot of Selected Ground-Water Site Data—BSNPLT

BSNPLT is a FORTRAN program designed to plot the areal location of selected ground-water sites. Thirteen site-selection parameters are available in BSNPLT. These parameters allow the user to select only those sites having the specific individual charactersitics of interest for study. Virtually any combination of available parameters may

be used. For example, a plot may be needed of only those wells having all the following characteristics:

—Constructed after 1951
—Depth of well greater than 1,000 feet
—Water levels greater than 400 feet
—Depth to top of the opening greater than 300 feet
—Sites with drillers' logs and (or) electric logs
—Primary water use is municipal
—Number of water-level records greater than 5

The program would select all sites within the polygon that meet all these specifications.

Selection parameters are grouped into three classes:

(1) Time-span parameters:

—Year of construction
—Year of water-level measurement

(2) Numeric parameters:

—Depth of well
—Water-level measurement
—Depth to top of open section
—Number of water-level measurements
—Altitude of water level
—Specific capacity
—Casing diameter

(3) Alpha-coded parameters:

—Log types available
—Type of water use
—Type of finish

Time-Span Selection Parameters

Selection of a site by a time-span parameter is governed by the following rules:

A. If the value in the beginning-year field is zero or blank and the ending-year field is greater than zero or blank, a site is selected only if its value for the time-span parameter is less than or equal to the value in the ending-year field: For example, if the beginning-year field is zero or blank and the ending-year field is the current calendar year, the entire period of record is selected.

B. If the beginning-year field is greater than zero or blank and the ending-year field is greater than zero or blank, a site is selected only if its value for the time-span parameter is greater than or equal to the beginning-year field and less than or equal to the ending-year field.

C. If the beginning-year field is zero or blank and the ending-year field is zero or blank, the parameter is not used in site selection.

D. If the beginning-year field is greater than zero or blank and the ending-year field is zero or blank, a site is selected only if its value for the parameter is greater than or equal to the beginning-year field.

For example,

Begin	End	Site data	Program
0	1960	1961	REJECT
	1960		ACCEPT
	1930		ACCEPT
	0		ACCEPT
1960	0	1961	ACCEPT
1960		1960	ACCEPT
1960		1930	REJECT
1960		0	REJECT
1950	1960	1961	REJECT
1950	1960	1960	ACCEPT
1950	1960	1950	ACCEPT
1950	1960	1952	ACCEPT
1950	1960	1930	REJECT

Numeric Selection Parameters

Selection of a site based on the numeric parameters is governed by the following rules:

A. If the input parameter is positive (+), a site is selected only if its value for the parameter is greater than or equal to the selection parameter.

B. If the input parameter is negative (-), a site is selected only if its value for the parameter is less than or equal to the selection parameter.

C. If the input parameter = 0, the parameter is not used in the selection process.

Example:

Parameter	Site data	Program
400	400	ACCEPT
400	500	ACCEPT
400	395	REJECT
-400	400	ACCEPT
-400	500	REJECT
-400	395	ACCEPT

Alpha-Coded Selection Parameters

Selection of a site based on the alpha-coded parameters is governed by the following rules:

A. If the selection parameter is left blank, the parameter is not used in the selection process.

B. If one or more codes are entered in the selection parameter, a site will be selected only if it has one or more matching codes for that parameter.

NOTE: All alpha-coded parameters used for selection are single-character codes.

For example,

Parameter	Site data	Program
D G	AUG	ACCEPT
D G	SAD	ACCEPT
D G		REJECT
D G	UA	REJECT

Site-Labeling Options

The manner in which a site is labeled on the plot is dependent on the value of the variables NID, NID1, and NID2 on control record 4.

If NID is

Zero: No label is plotted with the site symbol.

Blank: No label is plotted with the site symbol.

1: A single user-selected value will be plotted on the right side of the symbol. The value used for the label is determined by the value of NID1.

2: Two user-selected values will be plotted on the right side of the marker, one above the other. The values used are dependent upon the values of NID1 and NID2. The value for NID1 is plotted over the value for NID2.

For example,

NID = 1 and NID1 = 12

• 1520

NID = 2; NID1 = 1, NID2 = 12

• $\frac{413}{1520}$

The codes for NID1 and NID2 are

Code	Description
1	Sequence number on output listing
2	Depth of well
3	Altitude of site
4	Year of construction
5	Water-level measurement
6	Year of water-level measurement
7	Specific capacity
8	Depth to top of the open section
9	Diameter of the casing
10	Number of available water levels
11	Altitude of bottom of well
12	Altitude of water level
13	Altitude of top of the open section

Map-Border Options

Four border options are available to the user. Selection of a particular border option is made using the variable NOLNE on control record 3. The codes for borders and their meanings are

NEITHER or ' ': The plot area will have neither a boxed border nor polygon outlines. The corners of the plot area will be drawn and labeled.

MAP: A boxed or neatline border is drawn around the entire plot area and the outlines of polygons are not drawn.

BASIN: Corners of the plot area are drawn and labeled, and the outlines of all polygons input will be drawn.

BOTH: Both a boxed or neatline border and all polygon outlines are drawn.

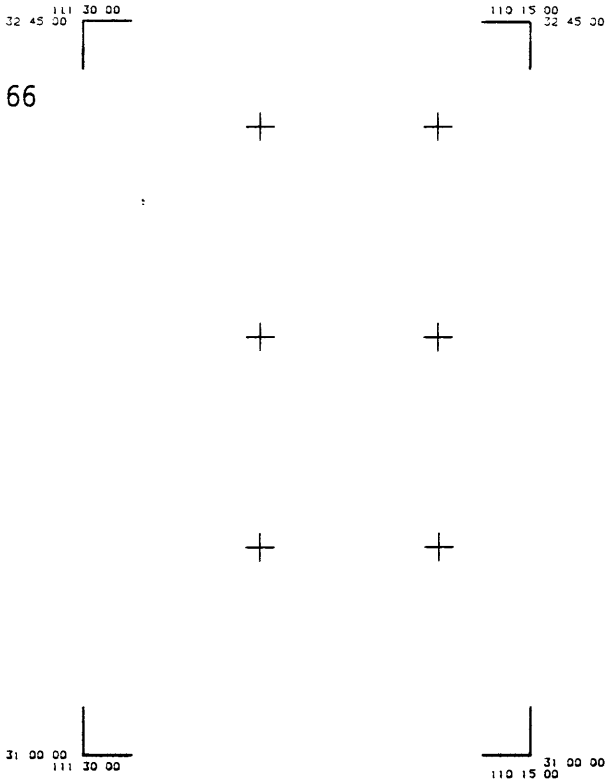
Examples of the output of these four options are shown in figure 12.

Program Input

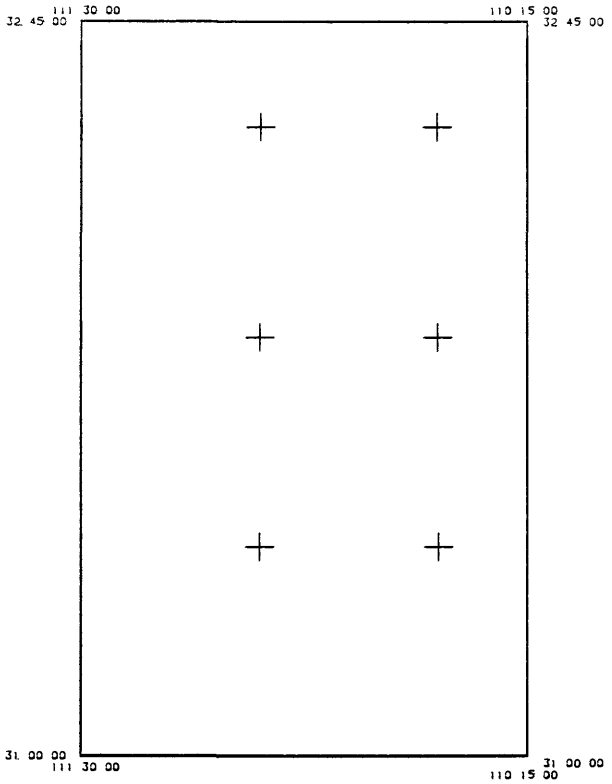
Input to BSNPLT consists of four control records, a BASIN-3 output data file, and a polygon-description file. The polygon file is optional (see section entitled "Map-Border Options").

Control records.--The following four control records are required for the operation of the program.

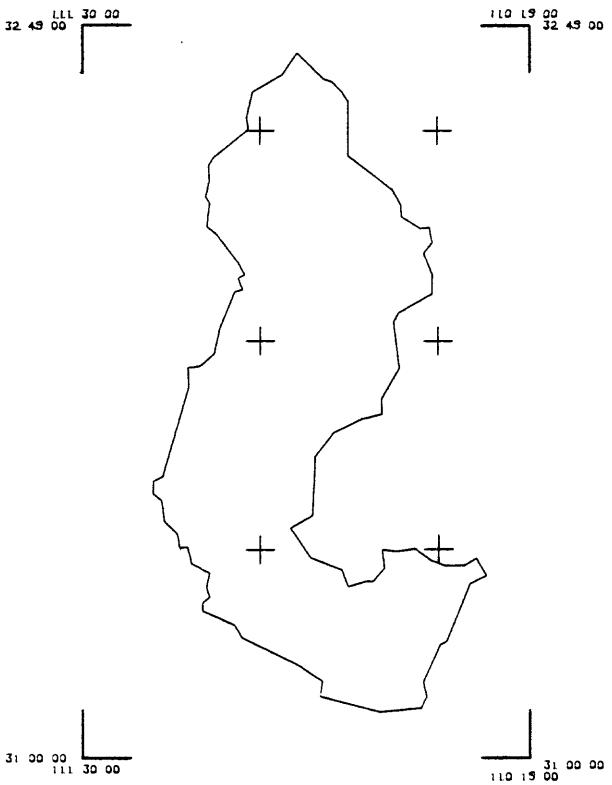
Record	Position	Variable	Format	Description
1	1- 2	IDC1	I2	Control record number must be '01'
	3-80	TITLE	7A10,A8	Title for plot; center is position 40
2	1- 2	IDC2	I2	Control record number must be '02'
	3- 4	LLAD	I2	Low latitude of plot, in degrees
	5- 6	LLAM	I2	Low latitude of plot, in minutes
	7- 8	LLAS	I2	Low latitude of plot, in seconds
	9-11	LLND	I3	Low longitude of plot, in degrees
	12-13	LLNM	I2	Low longitude of plot, in minutes
	14-15	LLNS	I2	Low longitude of plot, in seconds
	16-17	KHLAD	I2	High latitude of plot, in degrees
	18-19	KHLAM	I2	High latitude of plot, in minutes
	20-21	KHLAS	I2	High latitude of plot, in seconds
	22-24	KHLND	I3	High longitude of plot, in degrees
	25-26	KHLNM	I2	High longitude of plot, in minutes
	27-28	KHLNS	I2	High longitude of plot, in seconds
	29-30	KCMD	I3	Central meridian, in degrees



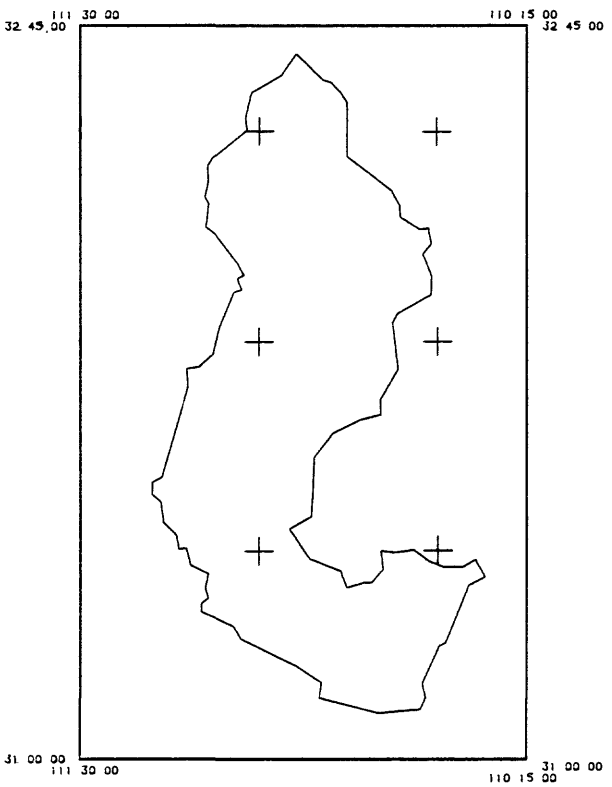
A. 'NEITHER' border option.



B. 'MAP' - only border option.



C. 'BASIN' - only border option.



D. 'BOTH' border option.

Figure 12.--Border options for areal plot of ground-water site data using program BSNPLT.

	32-33	KCMM	I2	Central meridian, in minutes
	34-35	KCMS	I2	Central meridian, in seconds
	36-37	KHPROJ[1]	I2	Type of map projection
	38-44	KSC	I7	Map scale of plot
	45-46	NWELLS	I2	Well-site plot switch
	47-48	NBRDR	I2	Plotter pen number to be used for polygon boundaries
	49-50	NHT	I2	Height of labels; inch/100
	51-52	NSHT	I2	Height of symbols; inch/100
3	1- 2	IDC3	I2	Control record number must be '03'
	3-12	ATIC1	A10	'TIC MARKS'
	13-15	ATIC2	A3	' AT'
	16-17	ITIC	I2	Tic mark interval in full minutes
	18-27	ATIC3	A10	'MIN MARKS'
	28-35	ATIC4	A8	'VALS ON'
	36-37	INTIC	I2	Full minute on which tic marks are to be drawn
	38-47	ATIC5	A10	'MIN MARKS'
	48-57	ATIC6	A10	'OUTLINE OF'
	58-59			
	60-66	NOLNE	A7	Map border option
4	1- 2	IDC4	I2	Control record number must be '04'
	3	NID	I1	Number of site-label parameters
	4- 5	NID1	I2	First label parameter code
	6- 7	NID2	I2	Second label parameter code
	8-11	CNYR1	F4.0	Beginning year of construction
	12-15	CNYR2	F4.0	Ending year of construction
	16-19	WLYR1	I4	Beginning water-level year
	20-23	WLYR2	I4	Ending water-level year
	24-28	XDEPTH	F5.0	Depth of well
	29-33	WTRLVL	F5.0	Water level
	34-38	OPNTOP	F5.0	Depth to top of opening
	39	NOPN	I1	Number of type of opening codes to be searched for
	40-42	IOPN	3I1	Type of opening codes to be selected
	43	NLOG	I1	Number of log types to be searched for
	44-51	ILOG	8A1	Log types to be selected
	52	IWUSE	I1	Number of water uses to be searched for
	53-55	IWUSE	3I1	Water uses to be selected
	56	NFIN	I1	Number of finish types to be searched for

[1]Only Lambert Conformal Conic Projection is active.

57-59	IFIN	3A1	Finishes to be selected
60-64	SPCAP	F5.2	Specific capacity
65-67	NWLVL	I3	Number of water-level records
68-72	WTRALT	F5.0	Altitude of water level
73-78	CDIAM	F6.3	Casing diameter

Program Output

Output from the program consists of a plotter-command file (punch file) and a printed listing of the options used and sites selected for plotting.

Site-data report and plot-option listing.--The listing printed from the program has three main parts:

—Descriptive information about the plot
—Listing of all selection parameters
—Data for selected sites

Plot description.--The first page of the listing provides information about the physical characteristics of the plot. This information includes:

—Plot title
—Map projection used
—Map scale of the plot
—High and low latitudes and longitudes represented by the four corners of the plot area
—Longitude that represents the central meridian of the plot area
—Height, in inches of the site symbols
—Height, in inches of the site labels
—Whether or not well sites that met selection criteria were plotted
—Plotter-pen number used to plot the polygon outlines, if plotted
—Interval and placement of tic marks, in minutes

Selection information listing.--The second page of the listing identifies the site-label options used, if any, the time ranges used in selection, if any, and the limits of physical parameters used in selection, if any.

Site-data listing.--Only those sites that meet all selection criteria are written to the listing and the output data file. The only

difference between the data listed by BSNPLT and those listed by BASIN-3 is the presence of two sequence numbers for each site in the BSNPLT listing. The first number is the sequence number from the BSNPLT program, and the second is the sequence number from the BASIN-3 program.

Error Messages

EITHER THE DATA FILE WAS EMPTY OR ALL SITES OUT OF RANGE

Input data file reached an at-end condition on the first read attempt or all sites in the input data file were outside the plot boundaries.

JOB ABORTED -- TOO LARGE

Width of the plot will exceed 33 inches; reduce the distance between high and low longitudes or increase the value of the map scale.

STRIP CONTROL OUT OF BOUND

The program plots a band of data that roughly corresponds to the width of a slice from the polygon-definition table. The width of the plot band is either negative or very large. The most likely cause is a central meridian outside the range between the low and high longitudes.

FILE OUT OF SEQUENCE

The program assumes that the input ground-water data is sorted in descending order using the site-identification number.

EOF WHILE READING OPTIONS AND CONTROLS

The input data file was in an at-end condition on the first read attempt.

FIRST CONTROL RECORD NOT '01' ID = __

The characters '01' do not appear in positions 1 and 2 of the first control record.

SECOND CONTROL RECORD NOT '02' ID = __

The characters '02' do no appear in positions 1 and 2 of the second control record.

THIRD CONTROL RECORD NOT '03' ID = __

The characters '03' do not appear in positions 1 and 2 of the third control record

FOURTH CONTROL RECORD NOT '04' ID =__

The characters '04' do not appear in positions 1 and 2 of the fourth control record.

LOW LATITUDE OUTSIDE STATE RANGE

LOW LONGITUDE OUTSIDE STATE RANGE

HIGH LATITUDE OUTSIDE STATE RANGE

HIGH LONGITUDE OUTSIDE STATE RANGE

The high and low latitudes and (or) longitudes for the State of Arizona are tested against the input plot corners, in the four if-statements following line 200 in subroutine NREAD. Either comment these if-statements or change the latitude and (or) longitude values.

LOW LONGITUDE GREATER THAN HIGH LONGITUDE

LOW LATITUDE GREATER THAN HIGH LATITUDE

Check the sequence of low and (or) high latitudes and longitudes on the control records.

NID IS GREATER THAN 2

A value greater than 2 has been entered in position 3 of control record 4.

NID1 HAS INVALID VALUE

NID1 (positions 4-5, control record 4) has a value less than zero or greater than 13.

NID2 HAS INVALID VALUE

NID2 (positions 6-7, control record 4) has a value less than zero or greater than 13.

BEGIN CONSTRUCTION YEAR IS GREATER THAN ENDING CONSTRUCTION YEAR

BEGIN WATER-LEVEL YEAR IS GREATER THAN ENDING WATER-LEVEL YEAR

Check the sequence of values for years on the control records

NLOG HAS INVALID VALUE

Number of log-type codes is less than zero or greater than 8.

NWUS HAS INVALID VALUE

Number of water-use codes is less than zero or greater than 3.

NFIN HAS INVALID VALUE

Number of finish codes is less than zero or greater than 3.

NOPN HAS INVALID VALUE

Number of type-of-opening codes is less than zero or greater than 3.

REFERENCES CITED

Anderson, T. W., 1980, Study plan for the regional aquifer-system analysis of alluvial basins in south-central Arizona and adjacent States: U.S. Geological Survey Open-File Report 80-1197, 22 p.

U.S. Geological Survey, 1976, WATSTORE, U.S. Geological Survey's water data storage and retrieval system: U.S. Geological Survey pamphlet INF-74-23, 15 p.

ATTACHMENTS

ATTACHMENT A

COBOL record descriptions for WATSTORE record types.

```
**********      GENERAL SITE DATA      **********

 01  R0-GENERAL-SITE-DATA.
     03  R0-KEYID.
         05  R0-KEY-LATITUDE        PIC 9(6).
         05  R0-KEY-LONGITUDE       PIC 9(7).
         05  R0-KEY-DUP-NUM         PIC 99.
     03  R0-REC-NUM                 PIC 999.
     03  R0-REC-LENGTH              PIC 999.
     03  R0-ACCESS-CODE             PIC X.
     03  C2-SITE-TYPE               PIC X.
     03  C3-DATA-RELIABILITY        PIC X.
     03  C4-SOURCE-AGENCY           PIC X(5).
     03  C5-PROJECT-NUM.
         05  C5-BASIN-1             PIC X(7).
         05  C5-BASIN-2             PIC X(6).
     03  C6-DISTRICT                PIC 999.
     03  C7-STATE                   PIC 99.
     03  C8-COUNTY                  PIC 999.
     03  C9-LATITUDE                PIC 9(6).
     03  C10-LONGITUDE              PIC 9(7).
     03  C11-LAT-LONG-ACURACY       PIC X.
     03  C12-LOCAL-NUMBER.
         05  C12-LOCAL-SITE         PIC X(14).
         05  C12-LOCAL-REST         PIC X(6).
     03  C13-LAND-NET-LOC           PIC X(30).
     03  C14-LOCATION-MAP-ID        PIC X(30).
     03  C15-MAP-SCALE              PIC 9(6).
     03  C16-ALTITUDE               PIC 9(5)V99.
     03  C17-ALTITUDE-METHOD        PIC X.
     03  C18-ALTITUDE-ACCURACY      PIC XXX.
     03  C19-TOPO-SETTING           PIC X.
     03  C20-OWDC-HYDRO-UNIT        PIC X(8).
     03  C21-DATE-CONSTRUCTED.
         05  C21-MONTH              PIC XX.
         05  C21-DAY                PIC XX.
         05  C21-YEAR               PIC XXXX.
     03  C23-SITE-USE               PIC X.
     03  C24-WATER-USE              PIC X.
     03  C25-SECOND-WATER-USE       PIC X.
     03  C26-THIRD-WATER-USE        PIC X.
     03  C27-HOLE-DEPTH             PIC 9(5)V99.
     03  C28-WELL-DEPTH             PIC 9(5)V99.
     03  C29-WELL-DEPTH-SOURCE      PIC X.
     03  C30-WATER-LEVEL            PIC 9(5)V99.
     03  C31-WATER-LEVEL-DATE       PIC 9(8).
     03  C32-DATE-ACCRCY-WL         PIC X.
     03  C33-WATER-LEVEL-SOURCE     PIC X.
     03  C34-MEAS-METHOD-WL         PIC X.
     03  C35-PUMP-USED              PIC X.
     03  C36-GEOHYDRO-DATA-SOURCE   PIC X.
     03  C37-SITE-STATUS            PIC X.
     03  C40-LAST-UPDATE            PIC X(8).
```

```
**********      LIFT DATA      **********

 01  R42-LIFT-DATA.
     03  R42-KEYID.
         05  R42-KEY-LATITUDE      PIC 9(6).
         05  R42-KEY-LONGITUDE     PIC 9(7).
         05  R42-KEY-DUP-NUM       PIC 99.
     03  R42-REC-NUM               PIC 999.
     03  R42-REC-LENGTH            PIC 999.
     03  R42-ACCESS-CODE           PIC X.
     03  C43-LIFT-TYPE             PIC X.
     03  C254-LIFT-ENTRY-NO        PIC 999.
     03  C44-INTAKE-SETTING        PIC 9(5).
     03  C45-POWER-TYPE            PIC X.
     03  C38-LIFT-DATE             PIC X(10).
     03  C46-HORSEPOWER            PIC 9(4)V9.


**********      MAJOR PUMP DATA      **********

 01 R47-MAJOR-PUMP-DATA.
     03  R47-KEYID.
         05  R47-KEY-LATITUDE      PIC 9(6).
         05  R47-KEY-LONGITUDE     PIC 9(7).
         05  R47-KEY-DUP-NUM       PIC 99.
     03  R47-REC-NUM               PIC 999.
     03  R47-REC-LENGTH            PIC 999.
     03  R47-ACCESS-CODE           PIC X.
     03  C43-LIFT-TYPE             PIC X.
     03  C48-MANUFACTURER          PIC X(8).
     03  C49-SERIAL-NO             PIC X(12).
     03  C50-POWER-COMPANY         PIC X(12).
     03  C51-ACCOUNT               PIC X(10).
     03  C52-METER                 PIC X(12).
     03  C53-CONSUMPTION           PIC 99V9(5).
     03  C54-PUMP-MAINTAINER       PIC X(12).
     03  C254-LIFT-ENTRY-NO        PIC 999.
     03  C255-ADDL-LIFT            PIC 999.
     03  C268-PUMP-CAPACITY        PIC 99999.
```

```
**********     STANDBY POWER DATA      **********


 01  R55-STANDBY-POWER-DATA.
     03  R55-KEYID.
         05  R55-KEY-LATITUDE      PIC 9(6).
         05  R55-KEY-LONGITUDE     PIC 9(7).
         05  R55-KEY-DUP-NUM       PIC 99.
     03  R55-REC-NUM               PIC 999.
     03  R55-REC-LENGTH            PIC 999.
     03  R55-ACCESS-CODE           PIC X.
     03  C43-LIFT-TYPE             PIC X.
     03  C56-STNDBY-POWER-TYPE     PIC X.
     03  C57-STANDBY-HORSEPOWER    PIC 9(4)V99.
     03  C254-LIFT-ENTRY-NO        PIC 999.


**********     WELL CONSTRUCTION DATA      **********


 01  R58-WELL-CONST-DATA.
     03  R58-KEYID.
         05  R58-KEY-LATITUDE      PIC 9(6).
         05  R58-KEY-LONGITUDE     PIC 9(7).
         05  R58-KEY-DUP-NUM       PIC 99.
     03  R58-REC-NUM               PIC 999.
     03  R58-REC-LENGTH            PIC 999.
     03  R58-ACCESS-CODE           PIC X.
     03  C59-CONST-SEQ-NO          PIC 9(3).
     03  C60-DATE-COMPLETED        PIC X(10).
     03  C64-CONST-DATA-SOURCE     PIC X.
     03  C63-CONTRACTOR            PIC X(12).
     03  C65-CONST-METHOD          PIC X.
     03  C66-FINISH                PIC X.
     03  C67-SEAL-TYPE             PIC X.
     03  C68-SEAL-BOTTOM           PIC 9(4).
     03  C69-DEVELOPMENT-METHOD    PIC X.
     03  C70-DEVELOPMENT-DURATION  PIC 999.
     03  C71-SPECIAL-TREATMENT     PIC X.
```

```
**********    DIMENSION OF HOLE CONSTRUCTED    **********

 01  R72-DIMENSION-HOLE-CONST.
     03  R72-KEYID.
         05  R72-KEY-LATITUDE      PIC 9(6).
         05  R72-KEY-LUNGITUDE     PIC 9(7).
         05  R72-KEY-DUP-NUM       PIC 99.
     03  R72-REC-NUM               PIC 999.
     03  R72-REC-LENGTH            PIC 999.
     03  R72-ACCESS-CODE           PIC X.
     03  C59-CONST-SEQ-NO          PIC 999.
     03  C60-DATE-COMPLETED        PIC X(8).
     03  C73-HOLE-TOP              PIC 9(5)V99.
     03  C74-HOLE-BOTTOM           PIC 9(5)V99.
     03  C75-HOLE-DIAMETER         PIC 999V99.


**********    CASING DATA    **********

 01  R76-CASING-DATA.
     03  R76-KEYID.
         05  R76-KEY-LATITUDE      PIC 9(6).
         05  R76-KEY-LONGITUDE     PIC 9(7).
         05  R76-KEY-DUP-NUM       PIC 99.
     03  R76-REC-NUM               PIC 999.
     03  R76-REC-LENGTH            PIC 999.
     03  R76-ACCESS-CODE           PIC X.
     03  C59-CONST-SEQ-NO          PIC 999.
     03  C60-DATE-COMPLETED        PIC X(10).
     03  C77-CASING-TOP            PIC 9(5)V99.
     03  C78-CASING-BOTTOM         PIC 9(5)V99.
     03  C79-CASING-DIAMETER       PIC 9(3)V99.
     03  C80-CASING-MATERIAL       PIC X.
     03  C81-CASING-THICKNESS      PIC 99V999.
```

```
**********      OPENINGS SCHEDULE      **********

 01  R82-OPENINGS-SCHEDULE.
     03  R82-KEYID.
         05  R82-KEY-LATITUDE       PIC 9(6).
         05  R82-KEY-LONGITUDE      PIC 9(7).
         05  R82-KEY-DUP-NUM        PIC 99.
     03  R82-REC-NUM                PIC 999.
     03  R82-REC-LENGTH             PIC 999.
     03  R82-ACCESS-CODE            PIC X.
     03  C59-CONST-SEQ-NO           PIC 999.
     03  C60-DATE-COMPLETED         PIC X(10).
     03  C83-OPENING-TOP            PIC 9(5)V99.
     03  C84-OPENING-BOTTOM         PIC 9(5)V99.
     03  C85-OPENING-TYPE           PIC X.
     03  C86-SCREEN-MATERIAL        PIC X.
     03  C87-OPENING-DIAMETER       PIC 999V99.
     03  C88-OPENING-WIDTH          PIC 99V99°.
     03  C89-OPENING-LENGTH         PIC 999V99.


**********      GEOHYDROLOGIC UNIT DESCRIPTION      **********

 01  R90-GEOHYDRO-UNIT-DESCRIP.
     03  R90-KEYID.
         05  R90-KEY-LATITUDE       PIC 9(6).
         05  R90-KEY-LONGITUDE      PIC 9(7).
         05  R90-KEY-DUP-NUM        PIC 99.
     03  R90-REC-NUM                PIC 999.
     03  R90-REC-LENGTH             PIC 999.
     03  R90-ACCESS-CODE            PIC X.
     03  C256-GEOHYDRO-SEQ-NO       PIC 999.
     03  C91-TOP-DEPTH              PIC 9(5)V99.
     03  C92-BOTTOM-DEPTH           PIC 9(5)V99.
     03  C93-GEOHYDRO-UNIT          PIC X(8).
     03  C304-GEOHYDRO-C-UNIT       PIC X.
     03  C96-LITHOLOGY              PIC X(4).
     03  C97-LITHOLOGIC-MODIFIER    PIC X(16).
```

```
**********      AQUIFER DATA      **********

 01  R94-AQUIFER-DATA.
     03  R94-KEYID.
         05  R94-KEY-LATITUDE      PIC 9(6).
         05  R94-KEY-LONGITUDE     PIC 9(7).
         05  R94-KEY-DUP-NUM       PIC 99.
     03  R94-REC-NUM               PIC 999.
     03  R94-REC-LENGTH            PIC 999.
     03  R94-ACCESS-CODE           PIC X.
     03  C256-GEOHYDRO-SEQ-NO      PIC 999.
     03  C95-AQUIFER-DATE          PIC X(10).
     03  C126-AQUIFR-STATIC-LEVEL  PIC 9(5)V99.
     03  C132-AQUIFR-CONTRIBUTION  PIC 999.


**********      WATER QUALITY DATA COLLECTION      **********

 01  R114-QW-DATA-COLLECT.
     03  R114-KEYID.
         05  R114-KEY-LATITUDE     PIC 9(6).
         05  R114-KEY-LONGITUDE    PIC 9(7).
         05  R114-KEY-DUP-NUM      PIC 99.
     03  R114-REC-NUM              PIC 999.
     03  R114-REC-LENGTH           PIC 999.
     03  R114-ACCESS-CODE          PIC X.
     03  C115-QN-BEGIN-YEAR        PIC 9999.
     03  C116-QN-END-YEAR          PIC 9999.
     03  C117-QN-DATA-SOURCE       PIC X(5).
     03  C118-QN-FREQUENCY         PIC X.
     03  C257-QN-NETWORK           PIC X.
     03  C120-QN-ANALYSIS-TYPE     PIC X.
     03  C307-QN-ANAL-AGENCY       PIC X(5).
```

```
**********     WATER LEVEL DATA COLLECTION     **********

 01  R121-WL-DATA-COLLECT.
     03  R121-KEYID.
         05  R121-KEY-LATITUDE     PIC 9(6).
         05  R121-KEY-LONGITUDE    PIC 9(7).
         05  R121-KEY-DUP-NUM      PIC 99.
     03  R121-REC-NUM              PIC 999.
     03  R121-REC-LENGTH           PIC 999.
     03  R121-ACCESS-CODE          PIC X.
     03  C122-LN-BEGIN-YEAR        PIC 9999.
     03  C123-LN-END-YEAR          PIC 9999.
     03  C124-LN-DATA-SOURCE       PIC X(5).
     03  C125-LN-FREQUENCY         PIC X.
     03  C258-LN-NETWORK           PIC X.


**********     WATER PUMPAGE/WITHDRAWAL DATA COLLECTION     **********

 01  R127-PUMP-WITHDRAWAL-DATA.
     03  R127-KEYID.
         05  R127-KEY-LATITUDE     PIC 9(6).
         05  R127-KEY-LONGITUDE    PIC 9(7).
         05  R127-KEY-DUP-NUM      PIC 99.
     03  R127-REC-NUM              PIC 999.
     03  R127-REC-LENGTH           PIC 999.
     03  R127-ACCESS-CODE          PIC X.
     03  C128-PN-BEGIN-YEAR        PIC 9(4).
     03  C129-PN-END-YEAR          PIC 9(4).
     03  C130-PN-DATA-SOURCE       PIC X(5).
     03  C131-PN-FREQUENCY         PIC X.
     03  C132-PN-NETWORK           PIC X.
     03  C133-PN-DATA-METHOD       PIC X.
```

```
********** PRODUCTION DATA      **********
**********      R = 134 -- FLOWING
**********      R = 146 -- PUMPED

 01   R134-46-PRODUCTION-DATA.
      03  R134-46-KEYID.
          05  R134-46-KEY-LATITUDE   PIC 9(6).
          05  R134-46-KEY-LONGITUDE  PIC 9(7).
          05  R134-46-KEY-DUP-NUM    PIC 99.
      03  R134-46-REC-NUM            PIC 999.
      03  R134-46-REC-LENGTH         PIC 999.
      03  R134-46-ACCESS-CODE        PIC X.
      03  C147-PUMP-SEQ-NO           PIC 999.
      03  C148-PUMP-MEAS-DATE        PIC 9(5)V99.
      03  C150-PUMP-DISCHARGE        PIC 9(5)V99.
      03  C151-PUMP-DISCHRGE-SOURCE  PIC X.
      03  C152-PUMP-DISCHRGE-METHOD  PIC X.
      03  C153-PUMP-PROD-LEVEL       PIC 9(5)V99.
      03  C154-PUMP-STATIC-LEVEL     PIC 9(5)V99.
      03  C155-PUMP-LEVEL-SOURCE     PIC X.
      03  C309-PUMP-DRAWDOWN         PIC 999V99.
      03  C272-PUMP-SPECIFIC-CAP     PIC 9(5)V99.
      03  C156-PUMP-LEVEL-METHOD     PIC X.
      03  C157-PUMP-PERIOD           PIC 9(4)V99.



**********    OWNER IDENTIFICATION    **********


 01   R158-OWNER-IDENTIFICATION.
      03  R158-KEYID.
          05  R158-KEY-LATITUDE     PIC 9(6).
          05  R158-KEY-LONGITUDE    PIC 9(7).
          05  R158-KEY-DUP-NUM      PIC 99.
      03  R158-REC-NUM              PIC 999.
      03  R158-REC-LENGTH           PIC 999.
      03  R158-ACCESS-CODE          PIC X.
      03  C159-OWNERSHIP-DATE       PIC X(10).
      03  C161-LAST-NAME            PIC X(10).
      03  C162-FIRST-NAME           PIC X(8).
      03  C163-MIDDLE-INITIAL       PIC X.
```

```
**********      OTHER SPRING DATA      **********


 01  R171.
     03  C1-KEYID.
         05  R171-KEY-LATITUDE     PIC 9(6).
         05  R171-KEY-LONGITUDE    PIC 9(7).
         05  R171-KEY-DUP-NUM      PIC 99.
     03  R171-REC-NUM              PIC 999.
     03  R171-REC-LENGTH           PIC 999.
     03  R171-ACCESS-CODE          PIC X.
     03  C172-SPRING-NAME          PIC X(20).
     03  C173SPRING-TYPE           PIC X.
     03  C174-PERMANENCE           PIC X.
     03  C175-DISCHARGE-SPHERE     PIC X.
     03  C176-IMPROVEMENTS         PIC X.
     03  C177-NO-SPRING-OPENINGS   PIC 999.
     03  C178-FLOW-VARIABLITIY     PIC 999.
     03  C179-FLOW-VAR-ACCURACY    PIC  X.



**********      OTHER DATA AVAILABLE      **********


 01  R180-OTHER-DATA-AVAIL.
     03  R180-KEYID.
         05  R180-KEY-LATITUDE     PIC 9(6).
         05  R180-KEY-LONGITUDE    PIC 9(7).
         05  R180-KEY-DUP-NUM      PIC 99.
     03  R180-REC-NUM              PIC 999.
     03  R180-REC-LENGTH           PIC 999.
     03  R180-ACCESS-CODE          PIC X.
     03  C312-OD-ENTRY-NO          PIC XXX.
     03  C181-TYPE-DATA            PIC X(10).
     03  C181-DATA-LOCATION        PIC X.
     03  C261-DATA-FORMAT          PIC X.
```

```
********** PERTINENT REMARKS          **********

 01  R183-PERTINENT-REMARKS.
     03  R183-KEYID.
         05  R183-KEY-LATITUDE      PIC 9(6).
         05  R183-KEY-LONGITUDE     PIC 9(7).
         05  R183-KEY-DUP-NUM       PIC 99.
     03  R183-REC-NUM               PIC 999.
     03  R183-REC-LENGTH            PIC 999.
     03  R183-ACCESS-CODE           PIC X.
     03  C311-PR-SEQ-NO             PIC 999.
     03  C185-PERT-REMARK           PIC X(40).


**********       SITE VISIT DATA     **********

 01  R186-SITE-VISIT-DATA.
     03  C1-KEYID.
         05  R186-KEY-LATITUDE      PIC 9(6).
         05  R186-KEY-LONGITUDE     PIC 9(7).
         05  R186-KEY-DUP-NUM       PIC 99.
     03  R186-REC-NUM               PIC 999.
     03  R186-REC-LENGTH            PIC 999.
     03  R186-ACCESS-CODE           PIC X.
     03  C187-INVENTORY-DATE        PIC X(10).
     03  C188-INVENTORY-PERSON      PIC X(10).
```

```
**********     OTHER SITE IDENTIFICATION NUMBERS    **********

 01  R189-OTHER-SITE-ID-NOS.
     03  R189-KEYID.
         05  R189-KEY-LATITUDE     PIC 9(6).
         05  R189-KEY-LONGITUDE    PIC 9(7).
         05  R189-KEY-DUP-NUM      PIC 99.
     03  R189-REC-NUM              PIC 999.
     03  R189-REC-LENGTH           PIC 999.
     03  R189-ACCESS-CODE          PIC X.
     03  C190-OTHER-ID             PIC X(10).
     03  C191-OTHER-ID-ASSIGNER    PIC X(15).


**********     FIELD WATER QUALITY MEASUREMENT     **********

 01  R192-FWQ-MEASUREMENT.
     03  R192-KEYID.
         05  R192-KEY-LATITUDE     PIC 9(6).
         05  R192-KEY-LONGITUDE    PIC 9(7).
         05  R192-KEY-DUP-NUM      PIC 99.
     03  R192-REC-NUM              PIC 999.
     03  R192-REC-LENGTH           PIC 999.
     03  R192-ACCESS-CODE          PIC X.
     03  C193-FWQ-SAMPLE-DATE      PIC X(10).
     03  C195-FWQ-GEOHYDRO-UNIT    PIC X(8).
     03  C196-FQW-PARAMETER        PIC 9(5).
     03  C197-FWQ-MEASUREMENT      PIC 9(6)V9.
```

```
**********      AVAILABLE LOG DATA      **********

 01  R198-AVIAL-LOG-DATA.
     03  R198-KEYID.
         05  R198-KEY-LATITUDE     PIC 9(6).
         05  R198-KEY-LONGITUDE    PIC 9(7).
         05  R198-KEY-DUP-NUM      PIC 99.
     03  R198-REC-NUM              PIC 999.
     03  R198-REC-LENGTH           PIC 999.
     03  R198-ACCESS-CODE          PIC X.
     03  C199-LOG-TYPE             PIC X.
     03  C200-LOG-TOP              PIC 9(5)V99.
     03  C201-LOG-BOTTOM           PIC 9(5)V99.
     03  C202-LOG-SOURCE           PIC X.


**********      WATER LEVEL MEASUREMENT SCHEDULE      **********

 01  R234-WL-MEASUREMENT.
     03  R234-KEYID.
         05  R234-KEY-LATITUDE     PIC 9(6).
         05  R234-KEY-LONGITUDE    PIC 9(7).
         05  R234-KEY-DUP-NUM      PIC 99.
     03  R234-REC-NUM              PIC 999.
     03  R234-REC-LENGTH           PIC 999.
     03  R234-ACCESS-CODE          PIC X.
     03  C235-WL-MEAS-DATE         PIC X(10).
     03  C237-WL-MEASUREMENT       PIC 9(5)V99.
     03  C238-WL-REMARK            PIC X.
     03  C239-WL-MEAS-METHOD       PIC X.
     03  C240-WL-TIME-PICK         PIC X.
```

```
**********      MEASURING POINT      **********

 01   R320-MEASURING-POINT.
      03   R320-KEYID.
           05   R320-KEY-LATITUDE      PIC 9(6).
           05   R320-KEY-LONGITUDE     PIC 9(7).
           05   R320-KEY-DUP-NUM       PIC 99.
      03   R320-REC-NUM                PIC 999.
      03   R320-REC-LENGTH             PIC 999.
      03   R320-ACCESS-CODE            PIC X.
      03   C321-MP-BEGIN-YEAR          PIC 9(4).
      03   C322-MP-END-YEAR            PIC 9(4).
      03   C323-MP-HEIGHT              PIC 999V99.
      03   C324-MP-REMARK              PIC X(40).
```

ATTACHMENT B

Program listings.

```
      PROGRAM PLGN01(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
     $TAPE15,TAPE16)
      DIMENSION EQA(2000),EQB(2000),EQC(2000),LA1(2000),
     $LN1(2000),LA2(2000),LN2(2000),X(2000),Y(2000),Z(2000),W(2000)
      DO 500 I=1,2000
C.....
C.....ZERO OUT ALL TABLES
C.....
      EQA(I) = 0.
      EQB(I) = 0.
      EQC(I) = 0.
      LA1(I) = 0
      LN1(I) = 0
      LA2(I) = 0
      LN2(I) = 0
      X(I) = 0.0
      Y(I) = 0.0
      Z(I) = 0.0
      W(I) = 0.0
  500 CONTINUE
C.....
C.....STORE INITIAL VALUES FOR SETTING UP THE HIGH AND
C.....LOW LATITUDES AND LONGITUDES
C.....
      LATL = 999999
      LONL = -9999999
      LATH = 0
      LONH = 0
C.....
C.....READ THE BASIN IDENTIFIER RECORD
C.....
      READ(5,600)BCODE,BEXPL1,BEXPL2,BEXPL3,BEXPL4,BEXPL5
  600 FORMAT(26X,A8,4A10,A6)
      I=1
C.....
C.....READ AND STORE THE VERTICES IN THE SEQUENCE AS READ
C.....STORE BOTH FIXED AND FLOATING POINT VALUES
C.....
C.....THE FOUR "IF" STATEMENTS SELECT THE HIGH AND LOW
C.....VALUES FOR LATITUDE AND LONGITUDE
C.....
   18 READ(5,601)NLAD,NLAM,NLAS,MLOD,MLOM,MLOS
      IF(EOF(5))14,13
  601 FORMAT(I3,1X,I2,1X,I2,2X,I3,1X,I2,1X,I2)
C.....  13 WRITE(6,602)NLAD,NLAM,NLAS,MLOD,MLOM,MLOS
C.....602 FORMAT(1X,I3,1X,I2,1X,I2,2X,I3,1X,I2,1X,I2)
   13 LA1(I) = NLAD*10000+NLAM*100+NLAS
      LN1(I) = -(MLOD*10000+MLOM*100+MLOS)
      XLOD = MLOD
      XLOM = MLOM
      XLOS = MLOS
      X(I) = -(XLOD+(XLOM/60.0)+(XLOS/3600.0))
      YLAD = NLAD
      YLAM = NLAM
      YLAS = NLAS
```

```
      Y(I)=YLAD+(YLAM/60.0)+(YLAS/3600.0)
      IF (LA1(I).LT.LATL) LATL = LA1(I)
      IF (LA1(I).GT.LATH) LATH = LA1(I)
      IF (LN1(I).GT.LONL) LONL = LN1(I)
      IF (LN1(I).LT.LONH) LONH = LN1(I)
      I=I+1
      GO TO 18
C.....
C.....SET NUMBER OF POINTS TO "NUMVER"
C.....STORE FIRST VERTEX IN THE LAST PLACE TO CLOSE POLYGON
C.....
   14 NUMVER=I-1
      K = I
      LA1(I) = LA1(1)
      LN1(I) = LN1(1)
      X(I)=X(1)
      Y(I)=Y(1)
C.....      DO 501 I=1,K
C.....      WRITE(6,603)LA1(I),Y(I),LN1(I),X(I),I
C..... 603 FORMAT(1X,"LA1= ",I7," Y(I) = ",F11.8,
C.....9"  LN1 =  ",I9,"  X(I) =  ",F13.8," INDEX  ",I5)
C..... 501 CONTINUE
C.....      WRITE(6,604)LATL,LATH,LONL,LONH
C..... 604 FORMAT(1X,"LOW LAT = ",I6,"  HIGH LAT = ",I6,"  LOW LNG = ",
C.....9I9,"  HIGH LNG = ",I9)
C.....      WRITE(6,605)
C..... 605 FORMAT(1X,"FOLLOWING IS THE TABLE OF THE MATCHED PAIRS")
C.....
C.....LINE SEGMENTS DEFINED BY PUTTING SECOND POINTS IN TABLE
C.....
      DO 502 I=1,NUMVER
      LA2(I) = LA1(I+1)
      LN2(I) = LN1(I+1)
      Z(I) = X(I+1)
      W(I) = Y(I+1)
C.....      WRITE(6,606)X(I),Y(I),Z(I),W(I),I
C..... 606 FORMAT(1X," X(I) = ",F13.8," Y(I) = ",F12.8,
C.....9" Z(I) = ",F13.8," W(I) = ",F12.8,I5)
  502 CONTINUE
C.....      WRITE(6,607)
C..... 607 FORMAT(1X,"TABLE WITH LOWEST LATITUDE FIRST")
C.....
C.....REARRANGE POINTS SO LINE SEGMENTS ARE DIRECTED FROM
C.....LOWER TO HIGHER LATITUDE
C.....
C.....      DO 103 I=1,NUMVER
C.....      WRITE(6,104)X(I),Y(I),Z(I),W(I),I
C..... 104 FORMAT(1X," X(I) = ",F13.8," Y(I) = ",F12.8,
C.....9" Z(I) = ",F13.8," W(I) = ",F12.8,I5)
C..... 103 CONTINUE
      DO 503 I=1,NUMVER
      IF (W(I).GT.Y(I)) GO TO 503
      TEMP = Y(I)
      Y(I) = W(I)
```

```
      W(I) = TEMP
      TEMP = X(I)
      X(I) = Z(I)
      Z(I) = TEMP
      KEMP = LA1(I)
      LA1(I) = LA2(I)
      LA2(I) = KEMP
      KEMP = LN1(I)
      LN1(I) = LN2(I)
      LN2(I) = KEMP
  503 CONTINUE
C.....      DO 504 I=1,NUMVER
C.....      WRITE(6,608)X(I),Y(I),Z(I),W(I),I
C..... 608 FORMAT(1X," X(I) = ",F13.8," Y(I) = ",F12.8,
C.....9" Z(I) = ",F13.8," W(I) = ",F12.8,I5)
C..... 504 CONTINUE
C.....
C..... ROUTINE TO ELIMINATE PARALELLS
C.....
      NUM = NUMVER
      I = 0
  505 IF (I+1.GE.NUMVER) GO TO 507
      I = I + 1
      IF (Y(I).NE.W(I)) GO TO 505
C.....
C.....THIS DROPS THE LINE AND REDUCES NUMBER OF LINES
C.....
C.....      WRITE(6,609)X(I),LN1(I),Y(I),LA1(I),Z(I),LN2(I),W(I),LA2(I)
C..... 609 FORMAT(1X,"X",F13.8," LN1",I8," Y",F12.8," LA1",I7," Z",
C.....9F13.8," LN2",I8," W",F12.8," LA2",I7)
      NUMVER = NUMVER - 1
      J = I
C.....
C..... LAST LINE IN TABLE TEST
C.....
  506 IF (J.GT.NUMVER) GO TO 505
      X(J) = X(J+1)
      Y(J) = Y(J+1)
      Z(J) = Z(J+1)
      W(J) = W(J+1)
      LA1(J) = LA1(J+1)
      LA2(J) = LA2(J+1)
      LN1(J) = LN1(J+1)
      LN2(J) = LN2(J+1)
      J = J + 1
      GO TO 506
C.....
C.....THE SORT ROUTINE COMPARES TWO LINES AND MOVES
C.....THE ONE WITH THE LOWEST FIRST LATITUDE TO THE TOP
C.....IF TWO FIRST LATITUDES ARE EQUAL THEN IT CHECKS THE
C.....LONGITUDES OF THE FIRST POINTS AND MOVES THE LARGEST TO
C.....THE TOP POSITION
C.....
C.....
```

```
  507 K = NUMVER - 1
      LSSW = 23
      DO 508 I = 1,K
      IF (Y(I).LT.Y(I+1)) GO TO 508
      IF (Y(I).EQ.Y(I+1)) GO TO 51
   50 TEMP = Y(I)
      Y(I) = Y(I+1)
      Y(I+1) = TEMP
      TEMP = X(I)
      X(I) = X(I+1)
      X(I+1) = TEMP
      TEMP = Z(I)
      Z(I) = Z(I+1)
      Z(I+1) = TEMP
      TEMP = W(I)
      W(I) = W(I+1)
      W(I+1) = TEMP
      KEMP = LN1(I)
      LN1(I) = LN1(I+1)
      LN1(I+1) = KEMP
      KEMP = LA1(I)
      LA1(I) = LA1(I+1)
      LA1(I+1) = KEMP
      KEMP = LN2(I)
      LN2(I) = LN2(I+1)
      LN2(I+1) = KEMP
      KEMP = LA2(I)
      LA2(I) = LA2(I+1)
      LA2(I+1) = KEMP
      LSSW = -23
      GO TO 508
 1508 IF(Z(I).LT.Z(I+1))GO TO 50
      GO TO 508
   51 IF (X(I).EQ.X(I+1)) GO TO 1508
      IF (X(I).LT.X(I+1)) GO TO 50
      GO TO 508
  508 CONTINUE
      IF (LSSW.LT.0) GO TO 507
C.....
C.....THIS IS THE END OF THE SORT ROUTINE
C.....
C.....NOW THE SORTED TABLE IS PRINTED OUT THIS WILL BE
C.....MADE INACTIVE AFTER THE PROGRAM IS TESTED
C.....
C.....
C.....NOW THE COEFFICIENTS OF THE LINES ARE COMPUTED AND STORED
C.....
C.....      DO 509 I=1, NUMVER
C.....      WRITE(6,610)X(I),LN1(I),Y(I),LA1(I),Z(I),LN2(I),W(I),LA2(I)
C..... 610 FORMAT(1X,"X",F13.8," LN1",I8," Y",F12.8," LA1",I7," Z",
C.....9F13.8," LN2",I8," W",F12.8," LA2",I7)
C..... 509 CONTINUE
```

```
      DO 510 I=1,NUMVER
      EQA(I) = W(I)-Y(I)
      EQB(I) = X(I)-Z(I)
      EQC(I) = X(I)*W(I)-Z(I)*Y(I)
  510 CONTINUE
C.....
C.....THIS PRINTS OUT THE FIXED POINT VALUES OF THE POINTS FOR
C.....EACH LINE AND THE COMPUTED COEFFICIENTS IN FLOATING POINT
C.....THIS TO BE DEACTIVATED AFTER PROGRAM TEST
C.....
C.....      DO 511 I=1,NUMVER
C.....      WRITE(6,611)LN1(I),LA1(I),LN2(I),LA2(I),EQA(I),EQB(I),EQC(I)
C..... 611 FORMAT(1X,4I8,3F13.8)
C..... 511 CONTINUE
C.....
C.....PUT IN TABLE END INDICATORS IN THE LAST PLUS ONE
C.....POSITION IN THE TABLE SO THAT TABLE END AND
C.....PROGRAM END CAN BE RECOGNIZED
C.....
      I = NUMVER +1
      LA1(I) = 9999999
      Y(I) = 999.9990
      X(I) = 999.9999
      Z(I) = 999.9999
      W(I) = 999.9999
      I = -1
C.....
C.....THIS IS THE CONTROLLING ROUTINE FOR PRODUCING THE
C.....ADDITIONAL LINE SEGMENTS TO PRODUCE SLICES OF THE
C.....POLYGON AS A SET OF TRAPEZOIDS OR TRIANGLE
C.....
  512 I = I+2
      LLSW = +23
      IF (I.GT.NUMVER) GO TO 900
  513 IF (Y(I).NE.Y(I+1)) GO TO 800
      IF (W(I).NE.W(I+1)) GO TO 514
      IF ((W(I).GT.Y(I+2)).AND.(W(I+1).GT.Y(I+2))) GO TO 517
      IF ((W(I).GT.Y(I+2)).AND.(Y(I+2).LT.999.9)) GO TO 802
      IF (Y(I+2).EQ.999.999) GO TO 900
      IF (X(I).GE.X(I+1)) GO TO 512
  520 TEMP = Y(I)
      Y(I) = Y(I+1)
      Y(I+1) = TEMP
      TEMP = X(I)
      X(I) = X(I+1)
      X(I+1) = TEMP
      TEMP = Z(I)
      Z(I) = Z(I+1)
      Z(I+1) = TEMP
      TEMP = W(I)
      W(I) = W(I+1)
      W(I+1) = TEMP
      KEMP = LN1(I)
      LN1(I) = LN1(I+1)
```

```
      LN1(I+1) = KEMP
      KEMP = LA1(I)
      LA1(I) = LA1(I+1)
      LA1(I+1) = KEMP
      KEMP = LN2(I)
      LN2(I) = LN2(I+1)
      LN2(I+1) = KEMP
      KEMP = LA2(I)
      LA2(I) = LA2(I+1)
      LA2(I+1) = KEMP
      TEMP = EQA(I)
      EQA(I) = EQA(I+1)
      EQA(I+1) = TEMP
      TEMP = EQB(I)
      EQB(I) = EQB(I+1)
      EQB(I+1) = TEMP
      TEMP = EQC(I)
      EQC(I) = EQC(I+1)
      EQC(I+1) = TEMP
      IF (LLSW) 750,750,512
C.....
C.....THIS ROUTINE IS ENTERED WITH Y(I) = Y(I+1) AND
C.....W(I) NOT = TO W(I+1)   TEST WILL BE MADE AND A BRANCH
C.....TO 517 "IF ((W(I).GT.Y(I+2)).AND.(w(I+1).GT.Y(I+2)))"
C.....BRANCH TO 534 TAKEN IF W(I).GT.W(I+1)
C.....
  514 IF ((W(I).GT.Y(I+2)).AND.(W(I+1).GT.Y(I+2))) GO TO 517
      IF (W(I).GT.W(I+1)) GO TO 534
      IF (W(I).NE.Y(I+2)) GO TO 898
      TX = 0.
      TY = W(I)
      TZ = Z(I+1)
      TW = W(I+1)
      TA = EQA(I+1)
      TB = EQB(I+1)
      TC = EQC(I+1)
      KX = 0
      KY = LA2(I)
      KZ = LN2(I+1)
      KW = LA2(I+1)
      LA2(I+1)=LA2(I)
      W(I+1) = W(I)
      IF (TA.EQ.0.0) GO TO 803
      TX = (TC-TB*TY)/TA
      Z(I+1) = TX
      KD = TX
      XKD = KD
      TX = TX-XKD
      KS = TX*3600.+.5
      KM = KS/60
      KS = KS-(KM*60)
      KX = KD*10000+KM*100+KS
      LN2(I+1) = KX
      TX = Z(I+1)
      GO TO 515
```

```
  534 IF (Y(I+2).EQ.999.9990) GO TO 834
      IF (W(I+1).NE.Y(I+2)) GO TO 899
      TX = 0.0
      TY = W(I+1)
      TZ = Z(I)
      TW = W(I)
      TA = EQA(I)
      TB = EQB(I)
      TC = EQC(I)
      KX = 0
      KY = LA2(I+1)
      KZ = LN2(I)
      KW = LA2(I)
      LA2(I)=LA2(I+1)
      W(I) = W(I+1)
      IF (TA.EQ.0.0) GO TO 803
      TX = (TC-TB*TY)/TA
      Z(I) = TX
      KD = TX
      XKD = KD
      TX = TX-XKD
      KS = Tx*3600.+.5
      KM = KS/60
      KS = KS-(KM*60)
      KX = KD*10000+KM*100+KS
      LN2(I) = KX
      TX = Z(I)
      GO TO 515
C.....
C.....THIS ROUTINE IS ENTERED WITH Y(I) = Y(I+1)
C.....AND W(I) > Y(I+2)
C.....
C.....      X        Y        Z        W
C.....10.0     4   11.0      7
C.....10.1     4   11.0      9      I+1
C.....10.0     4   11.1     11      I+2
C.....11.0     4   10.0     14      I+3
C.....
  517 IF (Y(I+2).NE.Y(I+3)) GO TO 802
      IF (Y(I+2).EQ.Y(I)) GO TO 1517
      IF (Y(I+2).LT.Y(I)) GO TO 802
      TR = Y(I+2)
      KR = LA1(I+2)
      LLSW = -23
      MI = I -1
  730 IF (MI.GE.I+1) GO TO 519
      MI= MI +1
      IF(W(MI).LE.TR)GO TO 730
      TX = 0.0
      TY = TR
      TZ = Z(MI)
      TW = W(MI)
      TA = EQA(MI)
      TB = EQB(MI)
      TC = EQC(MI)
```

```
      KX = 0
      KY = KR
      KZ = LN2(MI)
      KW = LA2(MI)
      LA2(MI) = KR
      W(MI) = TR
      IF (TA.EQ.0.0) GO TO 803
      TX = (TC-TB*TY)/TA
      Z(MI) = TX
      KD = TX
      XKD = KD
      TX = TX-XKD
      KS = TX*3600.+.5
      KM = KS/60
      KS = KS-(KM*60)
      KX = KD*10000+KM*100+KS
      LN2(MI) = KX
      TX = Z(MI)
      GO TO 515
  519 MI = I
      KN = I +2
C.....
C.....FIND LAST LINE TO BE SORTED
C.....
  717 IF (Y(KN).NE.Y(KN+1)) GO TO 749
      KN = KN +1
      GO TO 717
C.....
C..... KN = INDEX OF LAST LINE TO BE SORTED
C.....
C.....SORT ALL LINES  WITH EQUAL FIRST LATITUDES
C.....BY THEIR FIRST LONGITUDES
C.....
  749 NVER = +23
      I = MI +2
  750 IF (X(I).GT.X(I+1)) GO TO 751
      IF((X(I).EQ.X(I+1)).AND.(Z(I).GE.Z(I+1)))GO TO 751
      NVER = -23
      GO TO 520
  751 IF (I.GE.KN-1) GO TO 752
      I = I +1
      GO TO 750
  752 IF (NVER.LT.0) GO TO 749
C.....
C.....END OF SORT BY FIRST LONGITUDES
C.....
C.....  RETURN TO CONTROL ROUTINE
C.....
      I = MI
      GO TO 512
```

```
C NOW FIND LOWEST OF SECOND LATITUDES
C..... AND THE NEXT FIRST LATITUDE
C.....AND USE AS THE SECOND LATITUDE FOR
C.....ALL THE LINES STARTING WITH THE
C.....SAME FIRST LATITUDE
C.....THIS WILL PRODUCE A NEW SET OF LINES
C.....
C.....
C.....Y(I)=Y(I+1)=Y(I+2)=Y(I+3)
C.....
C.....  W(I) > Y(I) AND W(I+1) > Y(I)
C.....
C.....
 1517 KN = I
 1518 IF (Y(KN).NE.Y(KN+1)) GO TO 1519
      KN = KN+1
      GO TO 1518
 1519 LLSW = 0
      MI = I
      TR = 99.9
  761 IF (TR.LT.W(I)) GO TO 762
      TR = W(I)
      KR = LA2(I)
  762 IF (I.GE.KN) GO TO 1763
      I = I +1
      GO TO 761
 1763 IF (TR.LT.Y(KN+1)) GO TO 763
      TR = Y(KN+1)
      KR = LA1(KN+1)
  763 I = MI -1
  764 I = I +1
      IF (TR.EQ.W(I)) GO TO 765
      TX = 0.0
      TY = TR
      TZ = Z(I)
      TW = W(I)
      TA = EQA(I)
      TB = EQB(I)
      TC = EQC(I)
      KX = 0
      KY = KR
      KZ = LN2(I)
      KW = LA2(I)
      LA2(I) = KR
      W(I) = TR
      IF (TA.EQ.0.0) GO TO 803
      TX = (TC-TB*TY)/TA
      Z(I) = TX
      KD = TX
      XKD = KD
      TX = TX-XKD
      KS = TX*3600.+.5
      KM = KS/60
      KS = KS-(KM*60)
```

```
      KX = KD*10000+KM*100+KS
      LN2(I) = KX
      TX = Z(I)
      GO TO 515
  765 IF (TR.GT.W(I)) GO TO 802
      IF (I.LT.KN) GO TO 764
      I = I-1
      GO TO 512
C.....
C.....THIS ROUTINE FINDS THE LOCATION
C.....IN THE TABLE FOR THE NEW LINE
C.....
C.....THE TABLE MUST FIRST BE SHIFTED DOWN
C.....BY ONE LINE TO MAKE ROOM FOR THE NEW LINE
C.....
C.....
  515 DO 530 L=1,NUMVER
      IF (TY.LT.Y(L)) GO TO 531
      IF (TY.GT.Y(L)) GO TO 530
      IF (TX.GT.X(L)) GO TO 531
      IF (TX.LT.X(L)) GO TO 530
      IF (TW.GT.W(L)) GO TO 530
      IF (TW.LT.W(L)) GO TO 531
      IF (TZ.GT.Z(L)) GO TO 531
      IF (TZ.EQ.Z(L)) GO TO 804
  530 CONTINUE
  531 NL = NUMVER +1
      KL = NUMVER + 2
      K = L
      NUMVER = NUMVER + 1
      IF (LA1(NL).NE.9999999) GO TO 805
  532 X(KL) = X(NL)
      Y(KL) = Y(NL)
      Z(KL) = Z(NL)
      W(KL) = W(NL)
      LA1(KL) = LA1(NL)
      LA2(KL) = LA2(NL)
      LN1(KL) = LN1(NL)
      LN2(KL) = LN2(NL)
      EQA(KL) = EQA(NL)
      EQB(KL) = EQB(NL)
      EQC(KL) = EQC(NL)
      KL = NL
      NL = NL - 1
      IF (NL.GE.K) GO TO 532
      X(K) = TX
      Y(K) = TY
      Z(K) = TZ
      W(K) = TW
      EQA(K) = TA
      EQB(K) = TB
      EQC(K) = TC.....
      LN1(K) = KX
      LN2(K) = KZ
```

```
      LA1(K) = KY
      LA2(K) = KW
      IF (LLSW) 730,765,513
  999 STOP
C.....
C..... ERROR STOPS IF INCORRECT CONDITIONS ENCOUNTERED
C.....
  800 WRITE(6,850)I
  850 FORMAT(1X,"THE CURRENT LINES ARE NOT MATCHED",I5)
      GO TO 900
C.....
C.....THE FIRST LINE OF A PAIR SHOULD ALWAYS HAVE THE LOWER
C.....LATITUDE IN COMPARING THE SECOND POINTS
C.....
  801 WRITE(6,852)I
  852 FORMAT(1X,"THE LATITUDES OF THE SECOND POINTS ARE OUT OF SEQUENCE"
     $,I5)
      GO TO 900
C.....
C.....
C.....TA = 0.0 INDICATES A PARALELL LINE WHICH SHOULD HAVE
C.....BEEN ELIMINATED EARLIER
C.....
  803 WRITE(6,853)
  853 FORMAT(1X," A PARALELL LINE HAS BEEN DETECTED ")
      GO TO 900
C.....
C..... WITH Y(I) = Y(I+1) AND W(I)> BOTH Y(I+2) AND
C..... W(I+1) THERE IS NO CONNECTION TO THE NEXT SLICE
C..... THIS INDICATES AN ERROR CONDITION
C.....
  802 WRITE(6,854)I
  854 FORMAT(1X,"THERE IS A BREAK IN THE POLYGON",I5)
      GO TO 900
C.....
C.....LINE IN TABLE HAS COMMON FIRST POINT TO LINE BEING
C.....INSERTED AND LATITUDE OF SECOND POINTS IS ALSO EQUAL
C.....
  804 WRITE(6,855)I,L,NL,KL,NUMVER
  855 FORMAT(1X,"DUPLICATE LINE IN TABLE",5I5)
      GO TO 900
C.....
C.....TABLE MOVE NOT BEING STARTED AT RIGHT LOCATION
C.....
  805 WRITE(6,856)NL,KL
  856 FORMAT(1X,"LAST TABLE INDEX NOT CORRECT",2I5)
      GO TO 900
C.....
C..... INVALID END OF PROGRAM IN ROUTINE 514
C..... FOUND Y(I+2) = 999.9990
C.....
  814 WRITE(6,864)I
  864 FORMAT(1X,"INVALID END OF PROGRAM IN ROUTINE 514 ",I5)
      GO TO 900
```

```
C.....
C.....INVALID END OF PROGRAM IN ROUTINE 534
C.....FOUND Y(I+2) = 999.9990
C.....
  834 WRITE(6,884)I
  884 FORMAT(1X,,"INVALID END OF PROGRAM IN ROUTINE 534 ",I5)
      GO TO 900
C.....
C..... THIS THE CORRECT END TO THE POLYGON PROGRAM
C..... THE TABLE THAT IS PRINTED AT THIS TIME IS A COPY
C.....OF THE FILE THAT WILL BE USED TO IDENTIFY THE SITES
C..... FOR THIS BASIN
C.....
  898 WRITE(6,1900)NUMVER,I
 1900 FORMAT(1X,"ERROR-OUT OF SEQUENCE",2I5)
      GO TO 900
  899 WRITE(6,1900)NUMVER,I
  900 WRITE(6,901)NUMVER,I
  901 FORMAT(1X,"NORMAL JOB TERMINATION",2I5)
      IF (LONL.LT.0.) LONL = -LONL
      IF (LONH.LT.0.) LONH = -LONH
      WRITE(6,902)LATL,LONL,LATH,LONH,BCODE,BEXPL1,BEXPL2,
     $BEXPL3,BEXPL4,BEXPL5
  902 FORMAT(1X,2(I6,I7),A8,4A10,A6)
      WRITE(16,903)LATL,LONL,LATH,LONH,BCODE,BEXPL1,BEXPL2,
     $BEXPL3,BEXPL4,BEXPL5
  903 FORMAT(2(I6,I7),A8,4A10,A6)
      DO 910 J=1,NUMVER
      I=NUMVER+1-J
      IF (LN1(I).LT.0.0) LN1(I) = -LN1(I)
      IF (LN2(I).LT.0.0) LN2(I) = -LN2(I)
      KY = Y(I)*10000.+.5
      KX = X(I)*10000.+.5
      KW = W(I)*10000.+.5
      KZ = Z(I)*10000.+0.5
      I1 = "+"
      I2 = "+"
      IF (KX.LT.0) I1 = "-"
      IF (KX.LT.0) KX = -KX
      IF (KZ.LT.0) I2 = "-"
      IF (KZ.LT.0) KZ = -KZ
      KA = EQA(I)*10000000. + 0.5
      KB = EQB(I) * 10000000. + 0.5
      KC = EQC(I) * 10000. +0.5
      I3 = "+"
      I4 = "+"
      I5 = "+"
      IF (KA .LT. 0) I3 = "-"
      IF (KA .LT. 0) KA = -KA
      IF (KB .LT. 0) I4 = "-"
      IF (KB .LT. 0) KB = -KB
      IF (KC .LT. 0) I5 = "-"
      IF (KC .LT. 0) KC = -KC
```

```
      WRITE(6,911)LA1(I),LN1(I),LA2(I),LN2(I),KY,I1,KX,KW,I2,KZ,
     $I3,KA,I4,KB,I5,KC,I
      WRITE(16,912)LA1(I),LN1(I),LA2(I),LN2(I),KY,I1,KX,KW,I2,KZ,
     $I3,KA,I4,KB,I5,KC,I
  911 FORMAT(1X,2(I6,I7),2(I6,A1,I7),2(A1,I8),A1,I7,I4)
  912 FORMAT(2(I6,I7),2(I6,A1,I7),2(A1,I8),A1,I7,I4)
  910 CONTINUE
      IZERO = "000000"
      WRITE(16,913)IZERO
  913 FORMAT(A6,74X)
      GO TO 999
      END
```

```
IDENTIFICATION DIVISION.
PROGRAM-ID. EDIT-1.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600:
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEA ASSIGN TO TAPEIN.
    SELECT SLCFLE ASSIGN TO SLC.
    SELECT ACHNG ASSIGN TO ADCHNG.
    SELECT RNCARD ASSIGN TO RNCHNG.
    SELECT PULLCG ASSIGN TO PULHNG.
DATA DIVISION.
FILE SECTION.
FD  CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
01  CNL-CARD.
    02  CNLXXX.
        03 ID-CNL PIC IS XXX.
        03 ID-NUM PIC 999.
    02  DELNUM REDEFINES CNLXXX PIC 9(6).
    02  VSNA PIC X.
    02  ID-DES PIC X(73).
FD  LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
01  LIST-LINE.
    03  CC PIC X.
    03  LLPRT PIC X(132).
FD  FILEA
    DATA RECORD IS RSNCRD
    LABEL RECORDS ARE OMITTED.
01  RSNCRD.
    03  FILLER PIC X(4).
    03  IMAGE.
        05  KEYID.
            07  KEYLAT PIC 9(6).
            07  KEYLNG PIC 9(7).
            07  KEYDUP PIC 99.
        05  VFMTIN PIC X(61).
FD  SLCFLE
    DATA RECORD IS SLCREC
    LABEL RECORDS ARE OMITTED.
01  SLCREC.
    03  LA1 PIC 9(6).
    03  LN1 PIC 9(7).
    03  LA2 PIC 9(6).
    03  LN2 PIC 9(7).
    03  DA1 PIC 99V9999.
    03  LS1 PIC X.
```

```
    03  DN1 PIC 999V9999.
    03  DA2 PIC 99V9999.
    03  LS2 PIC X.
    03  DN2 PIC 999V9999.
    03  ASIGN PIC X.
    03  AM PIC 9V9999999.
    03  BSIGN PIC X.
    03  BM PIC 9V9999999.
    03  CSIGN PIC X.
    03  CK PIC 999V9999.
    03  SSEQ PIC 9999.
FD  ACHNG
    DATA RECORD IS AUDREC
    LABEL RECORDS ARE OMITTED.
01  AUDREC PIC X(96).
FD  PULLCG
    DATA RECORD IS PULREC
    LABEL RECORDS ARE OMITTED.
01  PULREC PIC X(96).
FD  RNCARD
    DATA RECORD IS RECOUT
    LABEL RECORDS ARE OMITTED.
01 RECOUT.
    03  RVSN PIC 9(5).
    03  RVSNA PIC X.
    03  DSKSTR.
        05  DSEQ PIC 9(9) VALUE 0.
        05  KEYID.
            07  KEYLAT.
                09  LATDEG PIC 99.
                09  LATMIN PIC 99.
                09  LATSEC PIC 99.
            07  KEYLNG.
                09  LNGDEG PIC 999.
                09  LNGMIN PIC 99.
                09  LNGSEC PIC 99.
            07  KEYDUP PIC 99.
        05  VFMTIN PIC X(61).
    03  PRD2 PIC 9(5).
COMMON-STORAGE SECTION.
01  TDA1.
    03  EDA1 USAGE COMP-2 OCCURS 10 TIMES.
01  TDN1.
    03  EDN1 USAGE COMP-2 OCCURS 10 TIMES.
01  TDA2.
    03  EDA2 USAGE COMP-2 OCCURS 10 TIMES.
01  TDN2.
    03  EDN2 USAGE COMP-2 OCCURS 10 TIMES.
01  TAM.
    03  EAM USAGE COMP-2 OCCURS 10 TIMES.
01  TBM.
    03  EBM USAGE COMP-2 OCCURS 10 TIMES.
01  TCK.
    03  ECK USAGE COMP-2 OCCURS 10 TIMES.
```

```
WORKING-STORAGE SECTION.
77  ACOUNT PIC 9(5) VALUE 0.
77  ALT PIC X VALUE "T".
77  ALX PIC X VALUE "X".
77  ALY PIC X VALUE "Y".
77  ALZ PIC X VALUE "Z".
77  A1 USAGE COMP-2.
77  A2 USAGE COMP-2.
77  A3 USAGE COMP-2.
77  CNRECN  PIC 9(6) VALUE ZEROES.
77  COUNTA PICTURE 9(6) VALUE 0.
77  COUNTN PIC 9(6) VALUE 1.
77  ERMSG1 PIC X(27) VALUE "  SITE ID OUT OF SEQUENCE  ".
77  ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
77  ERRCNT PIC 9(9) VALUE 0.
77  ERRSW PIC X VALUE "Y".
77  ESW PIC 9 VALUE 0.
77  FLECNT PIC 9(6).
77  FRSW PIC X VALUE "F".
77  G1 USAGE COMP-2.
77  G2 USAGE COMP-2.
77  G3 USAGE COMP-2.
77  HSW PIC X VALUE "T".
77  INCRDS PIC 9(6) VALUE 0.
77  IND PIC 99.
77  JULD PIC 9(5).
77  KDE PIC 999.
77  LAT USAGE COMP-2.
77  LATREC PIC 9(6).
77  LID PIC X(20).
77  LNCNT PIC 99 VALUE 70.
77  LNG USAGE COMP-2.
77  LOWL PIC 999999 VALUE 999999.
77  MNS PIC X VALUE "-".
77  NCDE PIC 9.
77  NDATIN PIC 9(6) VALUE 0.
77  NDETLN PIC 9(6) VALUE 0.
77  NFLEIA PIC 9(6) VALUE 0.
77  NFLEOA PIC 9(6) VALUE 0.
77  NU1 PIC 9 VALUE 1.
77  NU2 PIC 9 VALUE 2.
77  NU10 PIC 99 VALUE 10.
77  PGCNT PIC 999 VALUE 0.
77  PULMSG PIC X(6) VALUE "PULLED".
77  SEQCHK PIC 9(6) VALUE 0.
77  STRKEY PIC 9(15) VALUE 0.
77  TFILEA PIC 9(6) VALUE 0.
77  TFILEB PIC 9(6) VALUE 0.
77  TFILEC PIC 9(6) VALUE 0.
77  WRTSW PIC 99 VALUE 0.
77  X PIC 99.
77  XCDE USAGE COMP-2.
77  X6 PIC X(6) VALUE "XXXXXX".
```

```
01 WKDATE.
    03  YY PIC 99.
    03  MM PIC 99.
    03  DY PIC 99.
01 NAME-MONTH.
    03  JAN PIC XXXX VALUE " JAN".
    03  FEB PIC XXXX VALUE " FEB".
    03  MAR PIC XXXX VALUE " MAR".
    03  APR PIC XXXX VALUE " APR".
    03  MAY PIC XXXX VALUE " MAY".
    03  JUNE PIC XXXX VALUE "JUNE".
    03  JULY PIC XXXX VALUE "JULY".
    03  AUG PIC XXXX VALUE " AUG".
    03  SEPT PIC XXXX VALUE "SEPT".
    03  OCT PIC XXXX VALUE " OCT".
    03  NOV PIC XXXX VALUE " NOV".
    03  DEC PIC XXXX VALUE " DEC".
01 MONTH-TABLE REDEFINES NAME-MONTH.
    03  MONTHY PIC XXXX OCCURS 12 TIMES INDEXED BY NMM.
01  JOBCNL.
    03  CC PIC X VALUE "1".
    03  PCNL.
        05  CNTYPE PIC XXX.
        05  GO-CNL PIC 999.
        05  PCNLX PIC X(74).
    03  FILLER PIC X(52) VALUE SPACES.
01  JBCNLX.
    03  CC PIC X VALUE "0".
    03  PSTATS PIC X(14) VALUE "RECORD COUNTS".
    03  FILLER PIC X(6) VALUE SPACES.
    03  XCARD PIC X(8) VALUE "CARDS IN".
    03  CSTAT PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDRLNS PIC X(12) VALUE "DETAIL LINES".
    03  DRLNS PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDATIZN PIC X(7) VALUE "DATA IN".
    03  DATAIN PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XFAIN PIC X(11) VALUE "OLD-DATA I".
    03  FADATA PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XNOUT PIC X(12) VALUE "FMT DATA OUT".
    03  FOUT PIC ZZZZZ9.
01  HDR1.
    03  CC1 PIC 9 VALUE 1.
    03  FILLER PIC X.
    03  LDATE PIC X(6) VALUE "DATE: ".
    03  RMTH PIC X(4).
    03  FILLER PIC X.
    03  RDY PIC Z9.
    03  XB PIC XXXX VALUE ", 19".
    03  RYR PIC XX.
    03  FILLER PIC X(21).
```

```
    03  TITLEA PIC X(60).
    03  FILLER PIC X(21).
    03  PGE PIC X(4) VALUE "PAGE".
    03  PNUM PIC ZZZ9.
    03  FILLER PIC X(2).
01  HDR2.
    03  CC PIC 9 VALUE 0.
    03  USGS PIC X(7) VALUE " USGS: ".
    03  GSNAME PIC X(20).
    03  FILLER PIC XXX.
    03  BASINL PIC X(7) VALUE "SOURCES".
    03  FILLER PIC X.
    03  SRCEID PIC X(50).
    03  SRCNAM PIC X(30).
    03   CNLZZZ PIC X(6).
    03  FILLER PIC X(8).
01  HDR3C3.
    03  CC PIC X VALUE "0".
    03  FILLER PIC X(8) VALUE "  NUMBER".
    03  FILLER PIC XXX.
    03  FILLER PIC XXX VALUE "VSN".
    03  FILLER PIC X(29).
    03  FILLER PIC X(16) VALUE "ORGINAL CHANGE ".
    03  FILLER PIC X(17) VALUE "RECORD BETWEEN **".
    03  FILLER PIC X(40).
    03  FILLER PIC X(4) VALUE "DATE".
    03  FILLER PIC X(12).
01  PNEWRC.
    03  CC PIC X VALUE SPACE.
    03  DSEQ PIC Z(9).
    03  FILLER PIC X.
    03  RVSN PIC 9(5).
    03  RVSNA PIC X.
    03  DSKSTR.
        05  FILLER PIC XXXX VALUE "  **".
        05  FILLER PIC XXXX.
        05  KEYID.
            07  KEYLAT.
                09  LATDEG PIC 99.
                09  LATMIN PIC 99.
                09  LATSEC PIC 99.
            07  KEYLNG.
                09  LNGDEG PIC 999.
                09  LNGMIN PIC 99.
                09  LNGSEC PIC 99.
            07  KEYDUP PIC 99.
        05  VFMTIN PIC X(61).
    03  FILLER PIC XX VALUE "**".
    03  FILLER PIC X(13).
    03  PRD2 PIC 9(5).
    03  FILLER PIC X(12).
```

```
01   PRNTLN.
     03  CC PIC X VALUE SPACE.
     03  CNRECP PIC Z(9).
     03  FILLER PIC XX VALUE "**".
     03  RECPRT.
         05  RVSN PIC 9(5).
         05  VSNA PIC X.
         05  DSEQ PIC 9(9).
         05  CIMAGE PIC X(76).
         05  RDATE PIC 9(5).
     03  FILLER PIC XX VALUE "**".
     03  NOTESX PIC X(14).
     03  PERRSS PIC Z(8).
01   DATAA.
     03  ID-CNL PIC IS XXX.
     03  ID-NUM PIC IS 999.
     03  TITLEX.
         05  SRCAGC.
             07  GNAME PIC X(20).
             07  CNAME PIC X(30).
         05  FILLER PIC X(10).
     03  FILLER PIC X(14).
01   KEYIDL.
     03  KEYLAT.
         05  LATDEG PIC 99 VALUE 99.
         05  LATMIN PIC 99 VALUE 99.
         05  LATSEC PIC 99 VALUE 99.
     03 KEYLNG.
         05  LNGDEG PIC 999 VALUE 999.
         05  LNGMIN PIC 99 VALUE 99.
         05  LNGSEC PIC 99 VALUE 999.
     03  KEYDUP PIC 99 VALUE 99.
01   RANGEL.
     03  NUMFST PIC 9(6).
     03  NUMSTR PIC 9(6).
     03  NUMLST PIC 9(6).
     03  KEYIDS PIC 9(15).
     03  KEYIDE PIC 9(15).
     03  FILLER PIC X(32).


PROCEDURE DIVISION.
NOTE-A.
      NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION
      ROUTINE THAT OPENS THE INPUT AND LISTING FILES
      AND THEN THE DATE STORED IN THE SYSTEM IS READ
      AND USED TO SET UP THE DATE IN THE FIRST
      HEADING LINE.  NO RETURN FROM THE MAINLINE
      ROUTINE IS MADE TO THIS PROCEDURE.
```

```
BEGIN.
     OPEN INPUT CARD-FILE.
     OPEN OUTPUT LIST-FILE.
     ACCEPT WKDATE FROM DATE.
     ACCEPT JULD FROM DAY.
     SET NMM TO MM.
     MOVE MONTHY (NMM) TO RMTH OF HDR1.
     MOVE DY TO RDY OF HDR1.
     MOVE YY TO RYR OF HDR1.
NOTE-B.
      NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES
      THE GENERAL DESCRIPTION OF THE REPORT TO THE
      HEADING AREA.  THEN A BRANCH IS TAKEN TO THE
      APPROPRIATE STARTING ROUTINE DEPENDING ON THE
      NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
      THERE IS NO RETURN TO THIS ROUTINE.
READC.
     READ CARD-FILE AT END GO TO STOP1.
     IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
     IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
     ADD 1 TO INCRDS.
     MOVE CNL-CARD TO PCNL.
     MOVE CNLXXX TO CNLZZZ.
     READ CARD-FILE INTO DATAA AT END GO TO STOP4.
     IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
     IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
         TO TITLEA OF HDR1 ELSE GO TO STOP4.
     ADD 1 TO INCRDS.
     READ CARD-FILE INTO DATAA AT END GO TO STOP4.
     IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
     IF ID-NUM OF DATAA = 011 MOVE GNAME TO GSNAME
         MOVE CNAME TO SRCNAM ELSE GO TO STOP4.
     ADD 1 TO INCRDS.
     READ CARD-FILE INTO DATAA AT END GO TO STOP4.
     IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
     IF ID-NUM OF DATAA = 012 MOVE SRCAGC TO SRCEID,
         MOVE GO-CNL TO CNLZZZ ELSE GO TO STOP4.
     ADD 1 TO INCRDS.
     IF GO-CNL = 001, GO TO STARTA.
     IF GO-CNL = 002, GO TO STARTB.
     IF GO-CNL = 003, GO TO STARTC, ELSE GO TO STOP2.
NOTE-C.
     NOTE NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
     THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
     ROUTINES.
NOTE-CNL1.
     NOTE CONTROL 1 PROCESSING TO READ THE ORIGINAL DISTRICT
     UPDATE FILE AND/OR DELETE SELECTED RECORDS THAT FAIL VISUAL
     VERIFICATION OF LISTING.  THIS MAY BE RUN SEVERAL TIMES TO
     GET ALL JUNK RECORDS OUT OF THE INPUT FILE.
```

```
STARTA.
    READ CARD-FILE AT END DISPLAY
    "****700M1 C1 INPUT STREAM EOF *VSN*", STOP RUN.
    MOVE JULD TO PRD2 OF RECOUT. MOVE DELNUM TO RVSN OF RECOUT.
    MOVE VSNA OF CNL-CARD TO RVSNA OF RECOUT.
    READ CARD-FILE AT END DISPLAY
    "****701M1 C1 INPUT STREAM EOF *FILES?* ", STOP RUN.
    MOVE DELNUM TO FLECNT.
READCD.
    READ CARD-FILE AT END GO TO CLSEAC.
    IF DELNUM > SEQCHK, MOVE DELNUM TO SEQCHK ELSE GO TO STOP10.
READTP.
    IF FRSW = "F", PERFORM STARA1.
    READ FILEA AT END GO TO CLOSEA.
REDTP1.
    ADD 1 TO TFILEA, CNRECN, NFLEIA.
    MOVE CNRECN TO CNRECP OF PRNTLN.
    IF DELNUM = CNRECN,
     MOVE "DELETED RECORD" TO NOTESX OF PRNTLN ELSE
    ADD 1 TO ACOUNT, MOVE " STORED RECORD" TO NOTESX OF PRNTLN.
    MOVE IMAGE OF RSNCRD TO CIMAGE OF PRNTLN.
    MOVE KEYID OF RSNCRD TO KEYID OF DSKSTR OF RECOUT.
    MOVE VFMTIN OF RSNCRD TO VFMTIN OF DSKSTR OF RECOUT.
    MOVE CNRECN TO DSEQ OF RECOUT, DSEQ OF PRNTLN.
    MOVE JULD TO RDATE OF PRNTLN, PRD2 OF RECOUT.
    MOVE RVSN OF RECOUT TO RVSN OF PRNTLN.
    MOVE RVSNA OF RECOUT TO VSNA OF PRNTLN.
    PERFORM PRTSNL. IF DELNUM NOT = CNRECN,
       WRITE RECOUT, ADD 1 TO NFLEOA.
    IF DELNUM > CNRECN, GO TO READTP,
        ELSE GO TO READCD.
STARA1.
    OPEN INPUT FILEA. OPEN OUTPUT RNCARD.
    MOVE "X" TO FRSW.
CLOSEA.
    ADD 1 TO COUNTA, COUNTN.
    IF FLECNT = 1, GO TO CLSEA1,
       ELSE SUBTRACT 1 FROM FLECNT.
    PERFORM HDPRNT.
    DISPLAY "END OF FILE  ", COUNTA.
    DISPLAY "START FILE   ", COUNTN.
    READ FILEA AT END GO TO EROR07.
    GO TO REDTP1.
CLSEA1.
    CLOSE RNCARD, FILEA.  GO TO STOP3.
CLSEAC.
    DISPLAY "999999 MISSING ON END OF DELETE FILE".
    MOVE 999999 TO DELNUM.
    GO TO READTP.
```

```
EROR07.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "MORE THAN ONE TAPE MARK BETWEEN FILES".
    PERFORM HDPRNT.
    DISPLAY "END OF FILE  ", COUNTA.
    DISPLAY "START FILE   ", COUNTN.
    ADD 1 TO FLECNT. GO TO CLSEA1.
NOTE-CNL2.
    NOTE CONTROL 2 PROCESSING TO FIND IF SITE ID IS WITHIN
    BOUNDARY OF USGS DISTRICT AS DEFINED BY POLYGON SLICE FILE.
STARTB.
    OPEN INPUT SLCFLE, RNCARD. OPEN OUTPUT ACHNG.
    READ SLCFLE AT END GO TO STOP21.
    READ SLCFLE AT END GO TO STOP21.
STARB1.
    READ RNCARD AT END GO TO CLSEB.
    ADD NU1 TO NDATIN.
    IF KEYID OF RECOUT = STRKEY, GO TO WDISK.
    MOVE 0 TO WRTSW. MOVE KEYID OF RECOUT TO STRKEY.
    MOVE NU2 TO XCDE. MOVE LATDEG OF RECOUT TO A1.
    MOVE LATMIN OF RECOUT TO A2. MOVE LATSEC OF RECOUT TO A3.
    MOVE LNGDEG OF RECOUT TO G1. MOVE LNGMIN OF RECOUT TO G2.
    MOVE LNGSEC OF RECOUT TO G3. MOVE ZEROS TO LAT, LNG.
    ENTER DMSDEC USING A1, A2, A3, G1, G2, G3, LAT, LNG, XCDE.
STARB2.
    MOVE KEYLAT OF RECOUT TO LATREC.
    IF LATREC < LOWL, GO TO STRS. MOVE ZEROS TO XCDE.
    ENTER DSTEDT USING LAT, LNG, XCDE.
    IF XCDE NOT = ZERO, MOVE NU1 TO WRTSW.
WDISK.
    IF WRTSW = 0, MOVE JULD TO PRD2 OF RECOUT, WRITE AUDREC FROM
     RECOUT, ADD NU1 TO NFLEOA, ELSE MOVE RECOUT TO RECPRT,
     WRITE LIST-LINE FROM PRNTLN, ADD NU1 TO NDETLN.
    GO TO STARB1.
STRS.
    MOVE ZERO TO X.
STRS1.
    EXAMINE LA1 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE LN1 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE LA2 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE LN2 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE DA1 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE DN1 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE DA2 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE DN2 OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE AM OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE BM OF SLCREC REPLACING ALL SPACES BY ZERO.
    EXAMINE CK OF SLCREC REPLACING ALL SPACES BY ZERO.
    IF LATREC < LA1 OF SLCREC, GO TO STRS2.
    IF X = ZERO, MOVE LA1 OF SLCREC TO LOWL.
    ADD NU1 TO X. IF X > NU10, GO TO STP903.
    IF LA1 OF SLCREC NOT = LOWL, GO TO STARB2.
    IF LS1 = MNS, COMPUTE EDN1 (X) = - DN1,
        ELSE COMPUTE EDN1 (X) =  + DN1.
```

```
    IF LS2 = MNS, COMPUTE EDN2 (X) = - DN2,
        ELSE COMPUTE EDN2 (X) = + DN2.
    IF ASIGN = MNS, COMPUTE EAM (X) = - AM,
        ELSE COMPUTE EAM (X) = + AM.
    IF BSIGN = MNS, COMPUTE EBM (X) = - BM,
        ELSE COMPUTE EBM (X) = + BM.
    IF CSIGN = MNS, COMPUTE ECK (X) = - CK,
        ELSE COMPUTE ECK (X) = + CK.
    COMPUTE EDA1 (X) = + DA1.
    COMPUTE EDA2 (X) = + DA2.
STRS2.
    READ SLCFLE AT END GO TO STOP22. GO TO STRS1.
CLSEB.
    GO TO STOP3.
    STOP RUN.
STARTC.
    READ CARD-FILE INTO RANGEL AT END DISPLAY
    "****760M1 NO INPUT STREAM FILE", STOP RUN.
    IF NUMFST = 999999, MOVE NU1 TO WRTSW, OPEN OUTPUT ACHNG,
      OPEN OUTPUT PULLCG, ELSE MOVE ZERO TO WRTSW.
    OPEN INPUT RNCARD. MOVE ALT TO HSW.
    MOVE NU1 TO CNRECN, NDATIN.
    READ RNCARD AT END DISPLAY
    "****761M1 DATA CHANGE FILE, RNCHNG, EMPTY" STOP RUN.
STARC1.
    READ CARD-FILE INTO RANGEL AT END DISPLAY
    "****762M1 END OF RUN" GO TO STOP3.
    EXAMINE RANGEL REPLACING ALL " " BY ZERO.
    IF NUMFST NOT = ZERO, MOVE ALX TO HSW, GO TO X3.
    IF NUMSTR NOT = ZERO, MOVE ALY TO HSW, GO TO X3.
    IF KEYIDS NOT = ZERO, MOVE ALZ TO HSW.
X3.
    IF HSW = ALT, DISPLAY
    "****763M1 RANGE INCORRECT" GO TO STARC1.
    IF HSW = ALX, ADD NUMSTR, NUMLST, KEYIDS, KEYIDE
      GIVING KEYIDE IF KEYIDE NOT = 0 DISPLAY
    "****764M1 MULTIPLE RANGE PARAMETERS" GO TO STARC1.
    IF HSW = ALY, ADD NUMFST, KEYIDS, KEYIDE GIVING
       KEYIDE IF KEYIDE NOT = ZERO, DISPLAY
    "****765M1 SEQUENCE COUNT WITH OTHER VALUES", GO TO STARC1.
    IF HSW = ALZ, ADD NUMFST, NUMSTR, NUMLST GIVING
       NUMLST, IF NUMLST NOT = ZERO, DISPLAY
    "****766M1 SITE ID RANGE WITH OTHER VALUES", GO TO STARC1.
    IF HSW = ALY AND NUMLST < NUMSTR, DISPLAY
    "****767M1 SEQUENCE COUNT INVALID END", GO TO STARC1.
    IF HSW = ALZ AND KEYIDE > KEYIDS, DISPLAY
    "****768M1 SITE ID RANGE INVALID END HIGH", GO TO STARC1.
    IF HSW = ALX AND NUMFST < CNRECN, DISPLAY
    "****769M1 NO RECORDS PRINTED NUMFST = ",NUMFST,
    " CNRECN = ", CNRECN, GO TO STARC1.
STARC2.
    IF WRTSW = NU1, WRITE AUDREC FROM RECOUT, ADD NU1 TO NFLEOA.
    IF HSW = ALX AND CNRECN NOT > NUMFST, GO TO PRNOUT.
    IF HSW = ALY AND CNRECN NOT < NUMSTR AND CNRECN NOT > NUMLST
       GO TO PRNOUT.
```

```
    IF HSW = ALZ AND KEYID OF RECOUT NOT > KEYIDS AND KEYID OF
       RECOUT NOT < KEYIDE, GO TO PRNOUT.
STARC3.
    READ RNCARD AT END DISPLAY
    "****770M1 END OF DATA REQUESTS INCOMPLETE" GO TO STOP3.
    ADD NU1 TO CNRECN, NDATIN.
    IF HSW = ALX AND CNRECN > NUMFST GO TO STARC1.
    IF HSW = ALY AND CNRECN > NUMLST GO TO STARC1.
    IF HSW = ALZ AND KEYID OF RECOUT < KEYIDE, GO TO STARC1.
    GO TO STARC2.
PRNOUT.
    IF LNCNT > 60, PERFORM HDPRNT, PERFORM HDCN3.
    ADD NU1 TO LNCNT, NDETLN.
    MOVE CORRESPONDING RECOUT TO PNEWRC.
    MOVE DSEQ OF RECOUT TO DSEQ OF PNEWRC.
    WRITE LIST-LINE FROM PNEWRC.
    IF WRTSW = NU1, WRITE PULREC FROM RECOUT.
    GO TO STARC3.
HDCN3.
    WRITE LIST-LINE FROM HDR3C3.
HDPRNT.
    ADD 1 TO PGCNT.
    MOVE PGCNT TO PNUM OF HDR1.
    MOVE 4 TO LNCNT.
    WRITE LIST-LINE FROM HDR1.
    WRITE LIST-LINE FROM HDR2.
PRTSNL.
    IF LNCNT > 60, PERFORM HDPRNT,
        ELSE ADD 1 TO LNCNT, NDETLN.
    WRITE LIST-LINE FROM PRNTLN.
    MOVE " " TO CC OF PRNTLN. ADD 1 TO NDETLN.
STOP1.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "****710M1 CONTROL INPUT FILE EMPTY".
    STOP RUN.
STOP2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.
STOP3.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY TFILEB.
    MOVE INCRDS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FADATA.
    MOVE NDETLN TO DRLNS.
    MOVE NFLEOA TO FOUT.
    WRITE LIST-LINE FROM JBCNLX.
    DISPLAY "NORMAL JOB TERMINATION".
    CLOSE CARD-FILE, LIST-FILE.
    STOP RUN.
STOP4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "SOURCE ID RECORDS MISSING".
    STOP RUN.
```

```
STOP5.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "DATA INPUT FILE END INCORRECT".
    DISPLAY "EOF ACCURRED AT READB OR READV".
    STOP RUN.
STOP6.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "SECOND CARD OF PAIR MISSING".
    STOP RUN.
STOP7.
    DISPLAY "VSN AND/OR DATE RECORD MISSING INPUT EMPTY".
    DISPLAY "STOP7".  STOP RUN.
STOP9.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "RECORD TYPE INVALID".
    STOP RUN.
STOP10.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ERROR IN RECORD SEQUENCE", "STOP10".
    DISPLAY SEQCHK, "   ", DELNUM.
    STOP RUN.
STOP21.
    DISPLAY "****921MU1-C2 SLICE FILE EMPTY".
    STOP RUN.
STOP22.
    DISPLAY "****922MU1-C2 CHANGE PAST END OF SLICE FILE".
    STOP RUN.
STP903.
    DISPLAY "****903M1 SLICE TABLE OVERFLOW". STOP RUN.
STP904.
    DISPLAY "****904M1 EOF ON SLICE FILE CHANGES INVALID".
    GO TO STOP3.
```

```
IDENTIFICATION DIVISION.
PROGRAM-ID. EDIT-2.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEA ASSIGN TO DELFLE.
    SELECT RNCARD ASSIGN TO ADCHNG.
    SELECT CHNGEL ASSIGN TO SRCHNG.
DATA DIVISION.
FILE SECTION.
FD  CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
01  CNL-CARD.
    02  CNLXXX.
        03 ID-CNL PIC IS XXX.
        03 ID-NUM PIC 999.
    02  DELNUM REDEFINES CNLXXX PIC 9(6).
    02  VSNA PIC X.
    02  ID-DES PIC X(73).
FD  LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
01  LIST-LINE.
    03  CC PIC X.
    03  LLPRT PIC X(132).
FD  FILEA
    DATA RECORD IS ERRREC
    LABEL RECORDS ARE OMITTED.
01  ERRREC.
    03  CNRECR PIC 9(6).
    03  KSEQ PIC 9(4).
    03  DSKSTR.
        05  DSEQ PIC 9(4).
        05  KEYID.
            07  KEYLAT.
                09  LATDEG PIC 99.
                09  LATMIN PIC 99.
                09  LATSEC PIC 99.
            07  KEYLNG.
                09  LNGDEG PIC 999.
                09  LNGMIN PIC 99.
                09  LNGSEC PIC 99.
            07  KEYDUP PIC 99.
    03  VFMTIN PIC X(61).
    03  SNDATE.
        05  VSNTA PIC 9999.
        05  VSNA PIC X.
        05  PRDATE PIC 9(5).
```

```
FD  RNCARD
    DATA RECORD IS RECOUT
    LABEL RECORDS ARE OMITTED.
01 RECOUT.
    03  RVSN PIC 9(5).
    03  RVSNA PIC X.
    03  DSKSTR.
        05  DSEQ PIC 9(9) VALUE 0.
        05  KEYID.
            07  KEYLAT.
                09  LATDEG PIC 99.
                09  LATMIN PIC 99.
                09  LATSEC PIC 99.
            07  KEYLNG.
                09  LNGDEG PIC 999.
                09  LNGMIN PIC 99.
                09  LNGSEC PIC 99.
            07  KEYDUP PIC 99.
        05  VFMTIN PIC X(61).
    03  PRD2 PIC 9(5).
FD  CHNGEL
    DATA RECORD IS CNGREC
    LABEL RECORDS ARE OMITTED.
01  CNGREC.
    03  VSN PIC 99999.
    03  VSNA PIC X.
    03  PRD2 PIC 9(5).
    03  KSEQ PIC 9(4).
    03  KEYID.
        05  KEYLAT.
            07  LATDEG PIC 99.
            07  LATMIN PIC 99.
            07  LATSEC PIC 99.
        05  KEYLNG.
            07  LNGDEG PIC 999.
            07  LNGMIN PIC 99.
            07  LNGSEC PIC 99.
        05  KEYDUP PIC 99.
    03  CSCDE PIC X.
    03  RNUM PIC 999.
    03  CNUM PIC 999.
    03  CCR PIC X.
    03  CCSR PIC X.
    03  TYPDAT PIC 99.
    03  DLEN PIC 99.
    03  NEWDAT PIC X(45).
    03  NSEQ PIC 9(8).
WORKING-STORAGE SECTION.
77  ACOUNT PIC 9(6) VALUE 0.
77  ALD PIC X VALUE "D".
77  ALFLAT PIC X(6).
77  ALFLNG PIC X(7).
77  ALPHA6 PIC X(6).
77  ALPHA7 PIC X(7).
77  ALR PIC X VALUE "R".
```

```
77  ALT PIC X VALUE "T".
77  ALX PIC X VALUE "X".
77  ALY PIC X VALUE "Y".
77  ALZ PIC X VALUE "Z".
77  BT PIC 999 VALUE 0.
77  BTA PIC 9(4).
77  BLNK1 PIC X VALUE SPACE.
77  CMPLAT PIC 9(6).
77  CMPLNG PIC 9(7).
77  CNRECN  PIC 9(6) VALUE ZEROES.
77  DELSW PIC X VALUE "Z".
77  DEM1 PIC X(15) VALUE "  DATA ENTRY    ".
77  DEM2 PIC X(10) VALUE "*UNCHECKED".
77  DYZ PIC 9.
77  ERMSG1 PIC X(27) VALUE "  AZ FORMAT SORTED CHANGES ".
77  ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
77  ERRCNT PIC 9(9) VALUE 0.
77  ERRSW PIC X VALUE "Y".
77  ESW PIC 99 VALUE 0.
77  FLECNT PIC 9(6).
77  FRSW PIC X VALUE "F".
77  HSW PIC X VALUE "X".
77  INCRDS PIC 9(6) VALUE 0.
77  JULD PIC 9(5).
77  LNCNT PIC 99 VALUE 70.
77  MSG1 PIC X(25) VALUE "CHECKS OK DROP NOT NEEDED".
77  MSG2 PIC X(25) VALUE "KEYID NOT = LAT-LNG FLDS".
77  NDATIN PIC 9(6) VALUE 0.
77  NDETLN PIC 9(6) VALUE 0.
77  NFLEIA PIC 9(6) VALUE 0.
77  NFLEOA PIC 9(6) VALUE 0.
77  NU1 PIC 9 VALUE 1.
77  PGCNT PIC 999 VALUE 0.
77  SEQCHK PIC 9(6) VALUE 0.
77  TFILEA PIC 9(6) VALUE 0.
77  TFILEB PIC 9(6) VALUE 0.
77  TFILEC PIC 9(6) VALUE 0.
01 WKDATE.
    03  YY PIC 99.
    03  MM PIC 99.
    03  DY PIC 99.
01 NAME-MONTH.
    03  JAN PIC XXXX VALUE " JAN".
    03  FEB PIC XXXX VALUE " FEB".
    03  MAR PIC XXXX VALUE " MAR".
    03  APR PIC XXXX VALUE " APR".
    03  MAY PIC XXXX VALUE " MAY".
    03  JUNE PIC XXXX VALUE "JUNE".
    03  JULY PIC XXXX VALUE "JULY".
    03  AUG PIC XXXX VALUE " AUG".
    03  SEPT PIC XXXX VALUE "SEPT".
    03  OCT PIC XXXX VALUE " OCT".
    03  NOV PIC XXXX VALUE " NOV".
    03  DEC PIC XXXX VALUE " DEC".
```

```
01 MONTH-TABLE REDEFINES NAME-MONTH.
    03  MONTHY PIC XXxx OCCURS 12 TIMES INDEXED BY NMM.
01  JOBCNL.
    03  CC PIC X VALUE "1".
    03  PCNL.
        05  CNTYPE PIC XXX.
        05  GO-CNL PIC 999.
        05  PCNLX PIC X(74).
    03  FILLER PIC X(52) VALUE SPACES.
01  JBCNLX.
    03  CC PIC X VALUE "0".
    03  PSTATS PIC X(14) VALUE "RECORD COUNTS".
    03  FILLER PIC X(6) VALUE SPACES.
    03  XCARD PIC X(8) VALUE "CARDS IN".
    03  CSTAT PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDRLNS PIC X(12) VALUE "DETAIL LINES".
    03  DRLNS PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDATIZN PIC X(7) VALUE "DATA IN".
    03  DATAIN PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XFAIN PIC X(11) VALUE "OLD-DATA I".
    03  FADATA PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XNOUT PIC X(12) VALUE "FMT DATA OUT".
    03  FOUT PIC ZZZZZ9.
01  HDR1.
    03  CC1 PIC 9 VALUE 1.
    03  FILLER PIC X.
    03  LDATE PIC X(6) VALUE "DATE: ".
    03  RMTH PIC X(4).
    03  FILLER PIC X.
    03  RDY PIC Z9.
    03  X8 PIC XXXX VALUE ", 19".
    03  RYR PIC XX.
    03  FILLER PIC X(21).
    03  TITLEA PIC X(60).
    03  FILLER PIC X(21).
    03  PGE PIC X(4) VALUE "PAGE".
    03  PNUM PIC ZZZ9.
    03  FILLER PIC X(2).
01  HDR2.
    03  CC PIC 9 VALUE 0.
    03  USGS PIC X(7) VALUE " USGS: ".
    03  GSNAME PIC X(20).
    03  FILLER PIC XXX.
    03  BASINL PIC X(7) VALUE "SOURCES".
    03  FILLER PIC X.
    03  SRCEID PIC X(50).
    03  SRCNAM PIC X(30).
    03   CNLZZZ PIC X(6).
    03  FILLER PIC X(8).
```

```
01  HDR3.
    03  CC PIC 9 VALUE 0.
    03  FILLER PIC X(26).
    03  ERRMSG PIC X(27) VALUE " ERRMSG NOT INITIALIZED    ".
    03  FILLER PIC X(45).
    03  FILLER PIC X(18) VALUE "CURRECTION NOTES".
    03  FILLER PIC X(16).
01  PRNTLN.
    03  CC PIC X VALUE SPACE.
    03  CNRECP PIC ZZZZZZZ9.
    03  FILLER PIC XXXX VALUE "**  ".
    03  FILLER PIC X(4) VALUE "****".
    03  CIMAGE PIC X(96).
    03  FILLER PIC X(4) VALUE "****".
    03  NOTESX PIC X(11).
01 CCARD.
    03  CRDSEQ PIC 9999.
    03  KEYID.
        05  KEYLAT PIC 9(6).
        05  KEYLNG PIC 9(7).
        05  KEYDUP PIC 99.
    03  RECTYP PIC X(4).
    03  FILLER PIC X.
    03  TRANST PIC X(4).
    03  FILLER PIC X.
    03  CNUMBR PIC 99.
    03  FILLER PIC XX.
    03  GLCWLD.
         05  POSITION-1    PIC X.
         05  POSITION-2    PIC X.
         05  FILLER        PIC X(18).
    03  FILLER PIC X.
    03  QLCWLD PIC X(20).
    03  FILLER PIC X(4).
    03  FILETY PIC XX.
01  DATAA.
    03  ID-CNL PIC IS XXX.
    03  ID-NUM PIC IS 999.
    03  TITLEX.
        05  SRCAGC.
            07  GNAME PIC X(20).
            07  CNAME PIC X(30).
        05  FILLER PIC X(10).
    03  FILLER PIC X(14).
01  FMTSLD.
    03  CHRSLD PIC X OCCURS 61 TIMES INDEXED BY SC.
01  KEYIDL.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 99.
```

```
01  DATSLD.
    03  UDAT PIC X OCCURS 45 TIMES INDEXED BY DC.
01  RANGEL.
    03  NUMFST PIC 9(6).
    03  NUMSTR PIC 9(6).
    03  NUMLST PIC 9(6).
    03  KEYIDS PIC 9(15).
    03  KEYIDE PIC 9(15).
    03  FILLER PIC X(32).
01  HDR3C3.
    03  CC PIC X VALUE "0".
    03  FILLER PIC X(132) VALUE "  VSN   PRD  SEQ   LAT     LNG
-   "DUP     REC   C        SZ          DATA
-   "                 SEQUENCE ".
01  PCNGRC.
    03  CC  PIC X VALUE "0".
    03  VSN PIC 9(5).
    03  VSNA PIC X.
    03  FILLER PIC X.
    03  PRD2 PIC 9(5).
    03  KSEQ PIC ZZZ9.
    03  KEYID.
        05  FILLER PIC X.
        05  KEYLAT PIC 9(6).
        05  FILLER PIC X.
        05  KEYLNG PIC 9(7).
        05  FILLER PIC X.
        05  KEYDUP PIC 99.
    03  FILLER PIC XX.
    03  CSCDE PIC Xx.
    03  RNUM PIC ZZ999.
    03  CNUM PIC ZZ999.
    03  FILLER PIC XX.
    03  CCR PIC XX.
    03  CCSR PIC XX.
    03  DLEN PIC ZZ9.
    03  FILLER PIC XXX VALUE " **".
    03  NEWDAT PIC X(45).
    03  FILLER PIC XXX VALUE "** ".
    03  NSEQ PIC Z(8).



PROCEDURE DIVISION.
NOTE-A.
     NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION
     ROUTINE THAT OPENS THE INPUT AND LISTING FILES
     AND THEN THE DATE STORED IN THE SYSTEM IS READ
     AND USED TO SET UP THE DATE IN THE FIRST
     HEADING LINE.  NO RETURN FROM THE MAINLINE
     ROUTINE IS MADE TO THIS PROCEDURE.
```

```
BEGIN.
    OPEN INPUT CARD-FILE.
    OPEN OUTPUT LIST-FILE.
    ACCEPT WKDATE FROM DATE.
    ACCEPT JULD FROM DAY.
    SET NMM TO MM.
    MOVE MONTHY (NMM) TO RMTH OF HDR1.
    MOVE DY TO RDY OF HDR1.
    MOVE YY TO RYR OF HDR1.
NOTE-B.
     NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES
     THE GENERAL DESCRIPTION OF THE REPORT TO THE
     HEADING AREA.  THEN A BRANCH IS TAKEN TO THE
     APPROPRIATE STARTING ROUTINE DEPENDING ON THE
     NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
     THERE IS NO RETURN TO THIS ROUTINE.
READC.
    READ CARD-FILE AT END GO TO STOP1.
    IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
    IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
    ADD 1 TO INCRDS.
    MOVE CNL-CARD TO PCNL.
    MOVE CNLXxx TO CNLZZZ.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
        TO TITLEA OF HDR1 ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 011 MOVE GNAME TO GSNAME
        MOVE CNAME TO SRCNAM ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 012 MOVE SRCAGC TO SRCEID,
        MOVE GO-CNL TO CNLZZZ ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    IF GO-CNL = 001, GO TO STARTA.
    IF GO-CNL = 002, GO TO STARTB.
    IF GO-CNL = 003, GO TO STARTC, ELSE GO TO STOP2.
NOTE-C.
    NOTE NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
    THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
    ROUTINES.
STARTA.
    OPEN INPUT RNCARD. OPEN OUTPUT CHNGEL, FILEA.
    MOVE ERMSG1 TO ERRMSG OF HDR3. MOVE ZEROS TO NSEQ OF CNGREC.
STARA1.
    READ RNCARD AT END CLOSE FILEA RNCARD CHNGEL, DISPLAY
    "****700M2 END OF INPUT DATA",  GO TO STOP3.
    SET SC TO 0. ADD NU1 TO NDATIN, TFILEB, NFLEDA.
    IF KEYID OF RECOUT > KEYIDL, DISPLAY
    "****701M2 INPUT FILE OUT OF SEQUENCE", KEYID OF RECOUT,
        " > ", KEYIDL GO TO STOP3.
```

```
    MOVE VFMTIN OF RECOUT TO FMTSLD.
    IF KEYID OF RECOUT < KEYIDL, GO TO EDFRST.
    PERFORM SRCHNB. IF CHRSLD (SC) = ALR,
        SET SC DOWN BY NU1, GO TO EDFRST.
    SET SC DOWN BY NU1. IF HSW = ALX, GO TO WRTERR.
    IF HSW = "A", GO TO EDR1.
    IF HSW = "B", GO TO EDR2.
    IF HSW = "C", GO TO RNUM23.
    IF HSW = ALD, GO TO TRA1.
    IF HSW = "E", GO TO TRA2.
    IF HSW = "F", GO TO TRA3.
    IF HSW = "G", GO TO TRA4.
    IF HSW = "H", GO TO CTRANS.
    IF HSW = "I", GO TO CNUM23.
    IF HSW = "J", GO TO CDATA.
    IF HSW = "K", SET SC TO 1,  GO TO PC12A.
    IF HSW = "L", GO TO LCMVE.
    IF HSW = "M", GO TO CDAT1.
    DISPLAY "SWITCH CODE ERROR HSW = ", HSW.
    STOP RUN.
EDFRST.
    MOVE ZEROS TO TYPDAT OF CNGREC. MOVE ALD TO CSCDE OF CNGREC.
    MOVE NU1 TO KSEQ OF CNGREC. MOVE KEYID OF RECOUT TO KEYIDL.
    MOVE KEYID OF RECOUT TO KEYID OF CNGREC.
    PERFORM SRCHNB. IF CHRSLD (SC) NOT = "R",
      DISPLAY CHRSLD (SC), HSW,
      DISPLAY "SKIPPING TO NEXT RECORD FIRST CHARACTER NOT *R*",
      GO TO WRTERR.
    MOVE "A" TO HSW.
    MOVE DSEQ OF RNCARD TO NOTESX.
EDR1.
    PERFORM SRCHNB. IF CHRSLD (SC) NOT = "=",
      DISPLAY "RECORD FORMAT IN ERROR", GO TO WRTERR.
    MOVE "B" TO HSW.
EDR2.
    PERFORM SRCHNB. IF CHRSLD (SC) NOT < 0, MOVE CHRSLD (SC)
         TO BT, MOVE 1 TO YY, MOVE "C" TO HSW,
        ELSE DISPLAY "RECORD NUMBER NON-NUMERIC", GO TO WRTERR.
RNUM23.
    PERFORM SRCHNB. IF CHRSLD (SC) = "$" OR "*",
      MOVE ALD TO HSW, GO TO TRANSA.
    ADD 1 TO YY. IF YY > 3, DISPLAY "RECORD CODE ERROR",
        GO TO WRTERR.
    IF CHRSLD (SC) NOT < 0, MULTIPLY 10 BY BT,
        MOVE CHRSLD (SC) TO DYZ, ADD DYZ TO BT,
      GO TO RNUM23, ELSE DISPLAY "NON-NUMERIC IN FIELD",
        GO TO WRTERR.
TRANSA.
    MOVE BT TO RNUM OF CNGREC.
TRA1.
    PERFORM SRCHNB. IF CHRSLD (SC) NOT = "T",
     DISPLAY "T NOT FOUND", GO TO WRTERR.
    MOVE "E" TO HSW.
```

```
TRA2.
    PERFORM SRCHNB. IF CHRSLD (SC) NOT = "=",
      DISPLAY "TRANSACTION FORMAT ERROR",
      GO TO WRTERR.
    MOVE "F" TO HSW.
TRA3.
    PERFORM SRCHNB. MOVE "G" TO HSW.
    IF CHRSLD (SC) = "A" OR "M" OR "D" OR "V",
     MOVE CHRSLD (SC) TO CCR OF CNGREC, CCSR OF CNGREC,
     GO TO TRA4 ELSE DISPLAY RECOUT, CHRSLD (SC),
     "NO VALID TRANSACTION CODE", GO TO WRTERR.
TRA4.
    PERFORM SRCHNB. IF CHRSLD (SC) = "$" OR "*",
      MOVE "H" TO HSW, GO TO TRA5 ELSE DISPLAY
      "INCORRECT END OF TRANS CODE", GO TO WRTERR.
TRA5.
    IF (RNUM OF CNGREC = ZERO AND CCR OF CNGREC = ALD)
        MOVE ALX TO CCR OF CNGREC, GO TO PRTCNG.
CTRANS.
    MOVE 1 TO YY.
    PERFORM SRCHNB. IF CHRSLD (SC) < 0,
     DISPLAY "1ST CHARACTER OF *C NON-NUMERIC",
     GO TO WRTERR.
    MOVE CHRSLD (SC) TO BT. MOVE "I" TO HSW.
CNUM23.
    PERFORM SRCHNB. IF CHRSLD (SC) = "=" OR "#",
     MOVE "J" TO HSW, MOVE BT TO CNUM OF CNGREC,
         GO TO CDATA.
    ADD 1 TO YY. IF YY > 3, DISPLAY
     "SKIPPING REST OF RECORD-TOO MANY *C POSITIONS",
     GO TO WRTERR.
    IF CHRSLD (SC) < 0, DISPLAY
     "NON-NUMERIC ENCOUNTERED LOOKING FOR REST OF *C",
     GO TO WRTERR.
    MULTIPLY 10 BY BT. MOVE CHRSLD (SC) TO DYZ.
    ADD DYZ TO BT. GO TO CNUM23.
CDATA.
    MOVE SPACES TO DATSLD. MOVE 0 TO BTA.
    MOVE ALL ";" TO NEWDAT OF CNGREC.
    PERFORM SRCHNB.  SET DC TO 1.
    IF CNUM OF CNGREC = 12 OR 185, GO TO PC12.
CDAT1.
    IF CHRSLD (SC) = "*" OR "$", MOVE DATSLD TO NEWDAT OF CNGREC,
      MOVE BTA TO DLEN OF CNGREC, GO TO LLCHEK.
    MOVE CHRSLD (SC) TO UDAT (DC). ADD 1 TO BTA.
    SET SC UP BY 1. SET DC UP BY 1.
    IF DC > 45, GO TO YMVE.
    IF SC > 61, MOVE "M" TO HSW, GO TO STARA1,
        ELSE GO TO CDAT1.
LLCHEK.
    IF CNUM OF CNGREC = 9, GO TO LATCMP.
    IF CNUM OF CNGREC = 10, GO TO LNGCMP,
        ELSE GO TO PRTCNG.
```

```
LATCMP.
     MOVE NEWDAT OF CNGREC TO ALPHA6.
     MOVE ALPHA6 TO CMPLAT.
     MOVE KEYLAT OF RECOUT TO ALFLAT.
     IF CMPLAT = ALFLAT, MOVE "X" TO DELSW,
        ELSE MOVE "Y" TO DELSW.
     GO TO PRTCNG.
LNGCMP.
     MOVE NEWDAT OF CNGREC TO ALPHA7.
     MOVE ALPHA7 TO CMPLNG.
     MOVE KEYLNG OF RECOUT TO ALFLNG.
     IF CMPLNG = ALFLNG, MOVE "X" TO DELSW,
        ELSE MOVE "Y" TO DELSW.
     GO TO PRTCNG.
YMVE.
     DISPLAY "CDAT OVER 45 CHARACTERS".
     STOP RUN.
PC12.
     IF CHRSLD (SC) NOT = "'", DISPLAY
      "LOCAL WELL NUMBER DATA FIELD STARTS",
      " WITH OTHER THAN ' ", GO TO WRTERR,
       ELSE SET SC UP BY 1.
PC12A.
     MOVE CHRSLD (SC) TO UDAT (DC). ADD 1 TO BTA.
     SET SC UP BY 1. SET DC UP BY 1.
     IF DC > 45, GO TO XMVE.
     IF SC > 61, MOVE "K" TO HSW, GO TO STARA1.
     IF CHRSLD (SC) = "'", MOVE "L" TO HSW,
       GO TO LCMVE, ELSE GO TO PC12A.
LCMVE.
     PERFORM SRCHNB. IF CHRSLD (SC) = "$" OR "*",
         MOVE BTA TO DLEN OF CNGREC,
         MOVE DATSLD TO NEWDAT OF CNGREC, GO TO PRTCNG,
       ELSE DISPLAY "END OF DATA CHARACTER MISSING",
          GO TO WRTERR.
XMVE.
     DISPLAY "OVER 45 CHARACTERS".
     STOP RUN.
PRTCNG.
     MOVE RVSN OF RECOUT TO VSN OF CNGREC.
     MOVE RVSNA OF RECOUT TO VSNA OF CNGREC.
     MOVE JULD TO PRD2 OF CNGREC.
     IF DELSW = ALZ, ADD NU1 TO NSEQ OF CNGREC.
     MOVE ALD TO CSCDE OF CNGREC. MOVE CNGREC TO CIMAGE OF PRNTLN.
     ADD NU1 TO CNRECN. MOVE CNRECN TO CNRECP OF PRNTLN.
     MOVE ZEROS TO TYPDAT OF CNGREC.
     MOVE "H" TO HSW. IF DELSW  = "Z", WRITE CNGREC.
     IF DELSW = "Y", MOVE MSG2 TO NEWDAT OF CNGREC,
       MOVE CNGREC TO ERRREC, WRITE ERRREC.
     MOVE "Z" TO DELSW. ADD NU1 TO KSEQ OF CNGREC.
     PERFORM PRTSNL. GO TO CTRANS.
SRCHNB.
     SET SC UP BY 1. IF SC > 61, GO TO STARA1.
     IF CHRSLD (SC) = BLNK1, GO TO SRCHNB.
```

```
WRTERR.
    MOVE "X" TO HSW. ADD 1 TO ACOUNT.
    WRITE ERRREC FROM RECOUT. GO TO STARA1.
STARTB.
    STOP RUN.
STARTC.
    OPEN INPUT CHNGEL. MOVE ALT TO HSW.
    READ CHNGEL AT END DISPLAY
    "****760M2 NO DATA INPUT" STOP RUN.
    MOVE NU1 TO CNRECN, NDATIN.
STARC1.
    READ CARD-FILE INTO RANGEL AT END DISPLAY
    "****761M2 END OF RUN" GO TO STOP3.
    EXAMINE RANGEL REPLACING ALL " " BY ZERO.
    IF NUMFST NOT = ZERO, MOVE ALX TO HSW.
    IF NUMSTR NOT = ZERO, MOVE ALY TO HSW.
    IF KEYIDS NOT = ZERO, MOVE ALZ TO HSW.
    IF HSW = ALT, DISPLAY
    "****762M2 RANGE INCORRECT" GO TO STARC1.
    IF HSW = ALX, ADD NUMSTR, NUMLST, KEYIDS, KEYIDE
      GIVING KEYIDE IF KEYIDE NOT = 0 DISPLAY
    "****762M2 MULTIPLE RANGE PARAMETERS" GO TO STARC1.
    IF HSW = ALY, ADD NUMFST, KEYIDS, KEYIDE GIVING
       KEYIDE IF KEYIDE NOT = ZERO, DISPLAY
    "****764M2 SEQUENCE COUNT WITH OTHER VALUES", GO TO STARC1.
    IF HSW = ALZ, ADD NUMFST, NUMSTR, NUMLST GIVING
       NUMLST, IF NUMLST NOT = ZERO, DISPLAY
    "****765M2 SITE ID RANGE WITH OTHER VALUES", GO TO STARC1.
    IF HSW = ALY AND NUMLST < NUMSTR, DISPLAY
    "****766M2 SEQUENCE COUNT INVALID END", GO TO STARC1.
    IF HSW = ALZ AND KEYIDE < KEYIDS, DISPLAY
    "****767M2 SITE ID RANGE INVALID END LOW", GO TO STARC1.
    IF HSW = ALX AND NUMFST < CNRECN, DISPLAY
    "****768M2 NO RECORDS PRINTED NUMFST = ",NUMFST,
    " CNRECN = ", CNRECN, GO TO STARC1.
STARC2.
    IF HSW = ALX AND CNRECN NOT > NUMFST, GO TO PRNOUT.
    IF HSW = ALY AND CNRECN NOT < NUMSTR AND CNRECN NOT > NUMLST
       GO TO PRNOUT.
    IF HSW = ALZ AND KEYID OF CNGREC NOT > KEYIDS AND KEYID OF
       CNGREC NOT < KEYIDE, GO TO PRNOUT.
STARC3.
    READ CHNGEL AT END DISPLAY
    "****769M2 END OF DATA REQUESTS INCOMPLETE" GO TO STOP3.
    ADD NU1 TO CNRECN, NDATIN.
    IF HSW = ALX AND CNRECN > NUMFST GO TO STARC1.
    IF HSW = ALY AND CNRECN > NUMLST GO TO STARC1.
    IF HSW = ALZ AND KEYID OF CNGREC < KEYIDE, GO TO STARC1.
    GO TO STARC2.
PRNOUT.
    IF LNCNT > 60, PERFORM HDPRNT, PERFORM HDCN3.
    ADD NU1 TO LNCNT, NDETLN.
    MOVE CORRESPONDING CNGREC TO PCNGRC.
    WRITE LIST-LINE FROM PCNGRC.
    GO TO STARC3.
```

```
HDCN3.
     WRITE LIST-LINE FROM HDR3C3.
HDPRNT.
     ADD 1 TO PGCNT.
     MOVE PGCNT TO PNUM OF HDR1.
     MOVE 4 TO LNCNT.
     WRITE LIST-LINE FROM HDR1.
     WRITE LIST-LINE FROM HDR2.
PRTSNL.
     IF LNCNT > 60, PERFORM HDPRNT,
         ELSE ADD 1 TO LNCNT.
     WRITE LIST-LINE FROM PRNTLN.
     MOVE " " TO CC OF PRNTLN.
STOP1.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "CONTROL INPUT FILE EMPTY".
     STOP RUN.
STOP2.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
     STOP RUN.
STOP3.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY TFILEB.
     MOVE INCROS TO CSTAT.
     MOVE NDATIN TO DATAIN.
     MOVE NFLEIA TO FADATA.
     MOVE NDETLN TO DRLNS.
     MOVE NFLEOA TO FOUT.
     WRITE LIST-LINE FROM JBCNLX.
     DISPLAY "NORMAL JOB TERMINATION".
     DISPLAY "ERRORS = ", ACOUNT.
     CLOSE CARD-FILE, LIST-FILE.
     STOP RUN.
STOP4.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "STOP4".
     STOP RUN.
STOP5.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "STOP5".
     STOP RUN.
STOP6.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "STOP6".
     STOP RUN.
STOP7.
     DISPLAY "STOP7".  STOP RUN.
```

```
STOP8.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP8".
    STOP RUN.
STOP9.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP9".
    STOP RUN.
STOP10.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP10".
    STOP RUN.
```

```
IDENTIFICATION DIVISION.
PROGRAM-ID. EDIT-3.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEA ASSIGN TO UPCHNG.
    SELECT FILEB ASSIGN TO ERCHNG.
    SELECT CHANGE-FILE ASSIGN TO EDCHNG
           RESERVE 10 ALTERNATE AREAS.
DATA DIVISION.
FILE SECTION.
FD  CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
01  CNL-CARD.
    02  CNLXXX.
        03 ID-CNL PIC IS XXX.
        03 ID-NUM PIC 999.
    02  ID-DES PIC X(74).
FD  LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
01  LIST-LINE.
    03  CC PIC X.
    03  LLPRT PIC X(132).
FD  FILEA
    DATA RECORD IS RECOUT
    LABEL RECORDS ARE OMITTED.
01  RECOUT.
    03  VSN     PIC 9(5).
    03  VSNA    PIC X.
    03  PRD2    PIC 9(5).
    03  KSEQ    PIC 9(4).
    03  KEYID.
        05 KEYLAT.
           07  LATDEG   PIC 99.
           07  LATMIN   PIC 99.
           07  LATSEC   PIC 99.
        05 KEYLNG.
           07  LNGDEG   PIC 999.
           07  LNGMIN   PIC 99.
           07  LNGSEC   PIC 99.
        05 KEYDUP PIC 99.
    03  CSCDE      PIC X.
    03  RNUM       PIC 999.
    03  CNUM       PIC 999.
    03  CCR        PIC X.
    03  CCSR       PIC X.
    03  TYPDAT     PIC 99.
```

```
    03  DLEN       PIC 99.
    03  NEWDAT     PIC X(45).
    03  NSEQ       PIC 9(8).
FD  FILE8
    DATA RECORD IS AUDREC
    LABEL RECORDS ARE OMITTED.
01  AUDREC.
    03  VSN     PIC 9(5).
    03  VSNA    PIC X.
    03  PRD2    PIC 9(5).
    03  KSEQ    PIC 9(4).
    03  KEYID.
        05 KEYLAT.
           07  LATDEG   PIC 99.
           07  LATMIN   PIC 99.
           07  LATSEC   PIC 99.
        05 KEYLNG.
           07  LNGDEG   PIC 999.
           07  LNGMIN   PIC 99.
           07  LNGSEC   PIC 99.
        05 KEYDUP PIC 99.
    03  CSCDE      PIC X.
    03  RNUM       PIC 999.
    03  CNUM       PIC 999.
    03  CCR        PIC X.
    03  CCSR       PIC X.
    03  TYPDAT     PIC 99.
    03  DLEN       PIC 99.
    03  NEWDAT     PIC X(45).
    03  NSEQ       PIC 9(8).
FD  CHANGE-FILE
    DATA RECORD IS CNGREC
    LABEL RECORDS ARE OMITTED.
01  CNGREC.
    03  VSN     PIC 9(5).
    03  VSNA    PIC X.
    03  PRD2    PIC 9(5).
    03  KSEQ    PIC 9(4).
    03  KEYID.
        05 KEYLAT.
           07  LATDEG   PIC 99.
           07  LATMIN   PIC 99.
           07  LATSEC   PIC 99.
        05 KEYLNG.
           07  LNGDEG   PIC 999.
           07  LNGMIN   PIC 99.
           07  LNGSEC   PIC 99.
        05 KEYDUP PIC 99.
    03  CSCDE      PIC X.
    03  RNUM       PIC 999.
    03  CNUM       PIC 999.
    03  CCR        PIC X.
    03  CCSR       PIC X.
    03  TYPDAT     PIC 99.
```

```
    03  DLEN       PIC 99.
    03  NEWDAT     PIC x(45).
    03  NSEQ       PIC 9(8).
WORKING-STORAGE SECTION.
77  ALFAA PIC X VALUE "A".
77  ALFAB PIC X VALUE "B".
77  ALFAC PIC X VALUE "C".
77  ALFAD PIC X VALUE "D".
77  ALFAE PIC X VALUE "E".
77  ALFAF PIC X VALUE "F".
77  ALFAG PIC X VALUE "G".
77  ALFAH PIC X VALUE "H".
77  ALFAI PIC X VALUE "I".
77  ALFAJ PIC X VALUE "J".
77  ALFAK PIC X VALUE "K".
77  ALFAL PIC X VALUE "L".
77  ALFAM PIC X VALUE "M".
77  ALFAN PIC X VALUE "N".
77  ALFAO PIC X VALUE "O".
77  ALFAP PIC X VALUE "P".
77  ALFAQ PIC X VALUE "Q".
77  ALFAR PIC X VALUE "R".
77  ALFAS PIC X VALUE "S".
77  ALFAT PIC X VALUE "T".
77  ALFAU PIC X VALUE "U".
77  ALFAV PIC X VALUE "V".
77  ALFAW PIC X VALUE "W".
77  ALFAX PIC X VALUE "X".
77  ALFAY PIC X VALUE "Y".
77  ALFAZ PIC X VALUE "Z".
77  BT PIC 9999 VALUE 0.
77  BTA PIC 999.
77  BTB PIC 999.
77  CNRECN  PIC 9(6) VALUE ZEROES.
77  DCOUNT PIC 9(4) VALUE 0.
77  DELSW PIC X.
77  DECNT  PIC 99 VALUE 0.
77  DECSW  PIC 9 VALUE 0.
77  DGCNT  PIC 999 VALUE 0.
77  EDMSG1 PIC X(22) VALUE "   EDITED -C-NUMBER = ".
77  EMSG1 PIC X(22) VALUE " *** REJECT ERROR NO =".
77  EOFSW PIC S99 VALUE -1.
77  EQSW PIC 9999 VALUE 0.
77  ERCODE  PIC X.
77  ERMSG1 PIC X(27) VALUE "DELETED CHANGES EDIT ERRORS".
77  ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
77  ERNUM  PIC 9(5) VALUE 0.
77  ERRCNT PIC 9(9) VALUE 0.
77  ERRSW PIC X VALUE "Y".
77  ESMSG1 PIC X(20) VALUE "FILE OUT OF SEQUENCE".
77  ESMSG2 PIC X(26) VALUE "CORRECT SEQUENCE AND RERUN".
77  ESW PIC 9 VALUE 0.
77  FLECNT PIC 9(6).
77  HIALT  PIC X(6) VALUE "12633".
77  INCRDS PIC 9(6) VALUE 0.
```

```
77  KTEST PIC 9(15).
77  LNCNT PIC 99 VALUE 70.
77  LOALT  PIC X(6) VALUE "0".
77  MAXDEP  PIC X(5) VALUE "3000.".
77  NBLNK1 PIC X VALUE " ".
77  NDATIN PIC 9(6) VALUE 0.
77  NDETLN PIC 9(6) VALUE 0.
77  NEGSW  PIC 9 VALUE 0.
77  NFLEIA PIC 9(6) VALUE 0.
77  NFLEOA PIC 9(6) VALUE 0.
77  NMM PIC 9999.
77  NUMS  PIC 999.
77  NXX PIC 999.
77  NU0 PIC 9 VALUE 0.
77  NU1 PIC 9 VALUE 1.
77  NU2 PIC 9 VALUE 2.
77  NU3 PIC 9 VALUE 3.
77  NU4 PIC 9 VALUE 4.
77  NU5 PIC 9 VALUE 5.
77  NU6 PIC 9 VALUE 6.
77  NU7 PIC 9 VALUE 7.
77  NU8 PIC 9 VALUE 8.
77  NU9 PIC 9 VALUE 9.
77  NU10 PIC 99 VALUE 10.
77  NU11 PIC 99 VALUE 11.
77  NU12  PIC 99 VALUE 12.
77  NU13 PIC 99 VALUE 13.
77  NU14  PIC 99 VALUE 14.
77  NU15  PIC 99 VALUE 15.
77  NU16  PIC 99 VALUE 16.
77  NU17  PIC 99 VALUE 17.
77  NU18  PIC 99 VALUE 18.
77  NU19  PIC 99 VALUE 19.
77  NU20  PIC 99 VALUE 20.
77  NU21  PIC 99 VALUE 21.
77  NU22  PIC 99 VALUE 22.
77  NU23  PIC 99 VALUE 23.
77  NU24  PIC 99 VALUE 24.
77  NU25  PIC 99 VALUE 25.
77  NU26  PIC 99 VALUE 26.
77  NU27  PIC 99 VALUE 27.
77  N1 PIC 9.
77  N2 PIC 99.
77  N3 PIC 999.
77  N4 PIC 9999.
77  N5 PIC 9(5).
77  N6 PIC 9(6).
77  PGCNT PIC 999 VALUE 0.
77  SC PIC 99.
77  RDCNT  PIC 9(5) VALUE 0.
77  SE PIC 99.
77  SEQCHK PIC 9(6) VALUE 0.
77  TDYDAT  PIC X(8).
77  TFILEA PIC 9(6) VALUE 0.
77  TFILEB PIC 9(6) VALUE 0.
```

```
77  TFILEC PIC 9(6) VALUE 0.
77  UDCHK  PIC X.
77  UPNUM PIC 9(5) VALUE 0.
77  VALC27  PIC X(4).
77  XBLNK1 PIC X VALUE " ".
77  XBLNK2 PIC XX VALUE SPACES.
77  XBLNK3 PIC XXX VALUE SPACES.
77  XBLNK4 PIC X(4) VALUE SPACES.
77  XBLNK5 PIC X(5) VALUE SPACES.
77  XBLNK6 PIC X(6) VALUE SPACES.
77  XBLNK7 PIC X(7) VALUE SPACES.
77  XBLNK8 PIC X(8) VALUE SPACES.
77  XBLNK9 PIC X(9) VALUE SPACES.
77  XBLNK10 PIC X(10) VALUE SPACES.
77  XBLNK11 PIC X(11) VALUE SPACES.
77  XBLNK12 PIC X(12) VALUE SPACES.
77  XBLNK13 PIC X(13) VALUE SPACES.
77  XBLNK14 PIC X(14) VALUE SPACES.
77  XBLNK15 PIC X(15) VALUE SPACES.
77  XBLNK16 PIC X(16) VALUE SPACES.
77  XBLNK17 PIC X(17) VALUE SPACES.
77  XBLNK18 PIC X(18) VALUE SPACES.
77  XBLNK19 PIC X(19) VALUE SPACES.
77  XBLNK20 PIC X(20) VALUE SPACES.
77  XBLNK21 PIC X(21) VALUE SPACES.
77  XBLNK22 PIC X(22) VALUE SPACES.
77  XBLNK23 PIC X(24) VALUE SPACES.
77  XBLNK25 PIC X(25) VALUE SPACES.
77  XZERO PIC X VALUE "0".
77  X1 PIC X(1).
77  X2 PIC X(2).
77  X3 PIC X(3).
77  X4 PIC X(4).
77  X5 PIC X(5).
77  X6 PIC X(6).
77  X7 PIC X(7).
77  X8 PIC X(8).
77  X9 PIC X(9).
77  X10 PIC X(10).
77  X11 PIC X(11).
77  X12 PIC X(12).
77  X13 PIC X(13).
77  X14 PIC X(14).
77  X15 PIC X(15).
77  X16 PIC X(16).
77  X17 PIC X(17).
77  X18 PIC X(18).
77  X19 PIC X(19).
77  X20 PIC X(20).
77  X21 PIC X(21).
77  X22 PIC X(22).
77  X23 PIC X(23).
77  X24 PIC X(24).
77  X25 PIC X(25).
77  NUMER  PIC 9(11)V9(7) USAGE IS COMP.
```

```
01  SWITCHES.
    03  C-END      PIC X VALUE "N".
        88  END-CHANGE VALUE "Y".
    03  PROC-SW    PIC X VALUE "N".
        88  PROC-COMPLETE   VALUE "Y".
    03  SEQSW  PIC X VALUE "N".
    03  RNSW   PIC X VALUE "X".
01  QWCUDES.
    03  Qw1  PIC 9(5) VALUE 00010.
    03  Qw2  PIC 9(5) VALUE 00095.
    03  Qw3  PIC 9(5) VALUE 00300.
    03  Qw4  PIC 9(5) VALUE 00400.
    03  Qw5  PIC 9(5) VALUE 00405.
    03  Qw6  PIC 9(5) VALUE 00410.
    03  Qw7  PIC 9(5) VALUE 00430.
    03  Qw8  PIC 9(5) VALUE 00440.
    03  Qw9  PIC 9(5) VALUE 00445.
    03  Qw10 PIC 9(5) VALUE 00900.
    03  Qw11 PIC 9(5) VALUE 00940.
    03  Qw12 PIC 9(5) VALUE 00945.
    03  Qw13 PIC 9(5) VALUE 00950.
    03  Qw14 PIC 9(5) VALUE 01045.
    03  Qw15 PIC 9(5) VALUE 31501.
    03  Qw16 PIC 9(5) VALUE 31625.
    03  Qw17 PIC 9(5) VALUE 31673.
    03  Qw18 PIC 9(5) VALUE 71820.
    03  Qw19 PIC 9(5) VALUE 71830.

01  NUMSOUT.
    03  NUM02  PIC V99.
    03  NUM07  PIC V9(7).
    03  NUM23  PIC 99V999.
    03  NUM25  PIC 99V9(5).
    03  NUM32  PIC 999V99.
    03  NUM41  PIC 9999V9.
    03  NUM42  PIC 9999V99.
    03  NUM52  PIC 9(5)V99.
    03  NUM54  PIC 9(5)V9(4).
    03  NUM61  PIC 9(6)V9.
    03  NUM63  PIC 9(6)V999.
01  SIGNUMS.
    03  SNUM32  PIC S999V99.
    03  SNUM42  PIC S9999V99.
    03  SNUM52  PIC S99999V99.
    03  TSNUM32  PIC 999.99-.
    03  TSNUM42  PIC 9999.99-.
    03  TSNUM52  PIC 99999.99-.
01  TESTNUM.
    03  TN02  PIC .99.
    03  TN07  PIC .9(7).
    03  TN23  PIC 99.999.
    03  TN25  PIC 99.9(5).
    03  TN32  PIC 999.99.
    03  TN41  PIC 9999.9.
    03  TN42  PIC 9999.99.
```

```
    03  TN52  PIC 9(5).99.
    03  TN54  PIC 9(5).9999.
    03  TN61  PIC 9(6).9.
    03  TN63  PIC 9(6).999.
01  NNUM.
    03  NDIGIT PIC 9 OCCURS 11 TIMES INDEXED BY NLN.
01  ALFNUM.
    03  ADIGIT  PIC X OCCURS 20 TIMES INDEXED BY ALN.
01  GEO-CODES.
    03  ALVM  PIC X(5) VALUE "ALVM".
    03  ANDR  PIC X(5) VALUE "ANDR".
    03  ANDRS PIC X(5) VALUE "ANDRS".
    03  ARKS  PIC X(5) VALUE "ARKS".
    03  BSLT  PIC X(5) VALUE "BSLT".
    03  BNTN  PIC X(5) VALUE "BNTN".
    03  BLDR  PIC X(5) VALUE "BLDR".
    03  BLSD  PIC X(5) VALUE "BLSD".
    03  BLSC  PIC X(5) VALUE "BLSC".
    03  BRCC  PIC X(5) VALUE "BRCC".
    03  CLAY  PIC X(5) VALUE "CLAY".
    03  CLSN  PIC X(5) VALUE "CLSN".
    03  COAL  PIC X(5) VALUE "COAL".
    03  COBB  PIC X(5) VALUE "COBB".
    03  COSD  PIC X(5) VALUE "COSD".
    03  COSC  PIC X(5) VALUE "COSC".
    03  CLVM  PIC X(5) VALUE "CLVM".
    03  CGLM  PIC X(5) VALUE "CGLM".
    03  CQUN  PIC X(5) VALUE "CQUN".
    03  DIBS  PIC X(5) VALUE "DIBS".
    03  GLCL  PIC X(5) VALUE "GLCL".
    03  GNSS  PIC X(5) VALUE "GNSS".
    03  GRNT  PIC X(5) VALUE "GRNT".
    03  GRGN  PIC X(5) VALUE "GRGN".
    03  GRVL  PIC X(5) VALUE "GRVL".
    03  GRCM  PIC X(5) VALUE "GRCM".
    03  GRCL  PIC X(5) VALUE "GRCL".
    03  GRDS  PIC X(5) VALUE "GRDS".
    03  GRSC  PIC X(5) VALUE "GRSC".
    03  GRCK  PIC X(5) VALUE "GRCK".
    03  HROP  PIC X(5) VALUE "HROP".
    03  IGNS  PIC X(5) VALUE "IGNS".
    03  LGNT  PIC X(5) VALUE "LGNT".
    03  LMSN  PIC X(5) VALUE "LMSN".
    03  LMDM  PIC X(5) VALUE "LMDM".
    03  LOAM  PIC X(5) VALUE "LOAM".
    03  LOSS  PIC X(5) VALUE "LOSS".
    03  MRRL  PIC X(5) VALUE "MRBL".
    03  MARL  PIC X(5) VALUE "MARL".
    03  MRLS  PIC X(5) VALUE "MRLS".
    03  MMPC  PIC X(5) VALUE "MMPC".
    03  MUCK  PIC X(5) VALUE "MUCK".
    03  MUD   PIC X(5) VALUE "MUD".
    03  MDSN  PIC X(5) VALUE "MDSN".
    03  RSDM  PIC X(5) VALUE "RSDM".
    03  RYLT  PIC X(5) VALUE "RYLT".
```

```
    03  ROCK  PIC X(5) VALUE "ROCK".
    03  RBBL  PIC X(5) VALUE "RBBL".
    03  SAND  PIC X(5) VALUE "SAND".
    03  SDCL  PIC X(5) VALUE "SDCL".
    03  SDGL  PIC X(5) VALUE "SDGL".
    03  SDST  PIC X(5) VALUE "SDST".
    03  SGVC  PIC X(5) VALUE "SGVC".
    03  SNDS  PIC X(5) VALUE "SNDS".
    03  SRAN  PIC X(5) VALUE "SRAN".
    03  SHLE  PIC X(5) VALUE "SHLE".
    03  SILT  PIC X(5) VALUE "SILT".
    03  STCL  PIC X(5) VALUE "STCL".
    03  SLSN  PIC X(5) VALUE "SLSN".
    03  SLTE  PIC X(5) VALUE "SLTE".
    03  SOIL  PIC X(5) VALUE "SOIL".
    03  SYNT  PIC X(5) VALUE "SYNT".
    03  TILL  PIC X(5) VALUE "TILL".
    03  TUF   PIC X(5) VALUE "TUF".
01 WKDATE.
    03  YY PIC 99.
    03  MM PIC 99.
    03  DY PIC 99.
01 NAME-MONTH.
    03  JAN PIC XXXX VALUE " JAN".
    03  FEB PIC XXXX VALUE " FEB".
    03  MAR PIC XXXX VALUE " MAR".
    03  APR PIC XXXX VALUE " APR".
    03  MAY PIC XXXX VALUE " MAY".
    03  JUNE PIC XXXX VALUE "JUNE".
    03  JULY PIC XXXX VALUE "JULY".
    03  AUG PIC XXXX VALUE " AUG".
    03  SEPT PIC XXXX VALUE "SEPT".
    03  OCT PIC XXXX VALUE " OCT".
    03  NOV PIC XXXX VALUE " NOV".
    03  DEC PIC XXXX VALUE " DEC".
01 MONTH-TABLE REDEFINES NAME-MONTH.
    03  MONTHY PIC XXXX OCCURS 12 TIMES.
01  JOBCNL.
    03  CC PIC X VALUE "1".
    03  PCNL.
        05  CNTYPE PIC XXX.
        05  GO-CNL PIC 999.
        05  PCNLX PIC X(74).
    03  FILLER PIC X(52) VALUE SPACES.
01  JBCNLX.
    03  CC PIC X VALUE "0".
    03  PSTATS PIC X(14) VALUE "RECORD COUNTS".
    03  FILLER PIC X(6) VALUE SPACES.
    03  XCARD PIC X(8) VALUE "CARDS IN".
    03  CSTAT PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDRLNS PIC X(12) VALUE "DETAIL LINES".
    03  DRLNS PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDATIZN PIC X(7) VALUE "DATA IN".
```

```
    03  DATAIN PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XFAIN PIC X(11) VALUE "OLD-DATA I".
    03  FADATA PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XNOUT PIC X(12) VALUE "FMT DATA OUT".
    03  FOUT PIC ZZZZZ9.
01  HDR1.
    03 CC1 PIC 9 VALUE 1.
    03 FILLER PIC X.
    03 LDATE PIC X(6) VALUE "DATE: ".
    03 RMTH PIC X(4).
    03 FILLER PIC X.
    03 RDY PIC Z9.
    03 X8 PIC XxXX VALUE ", 19".
    03 RYR PIC XX.
    03 FILLER PIC X(21).
    03 TITLEA PIC X(60).
    03 FILLER PIC X(21).
    03 PGE PIC X(4) VALUE "PAGE".
    03 PNUM PIC ZZZ9.
    03 FILLER PIC X(2).
01  HDR2.
    03 CC PIC 9 VALUE 0.
    03 USGS PIC X(7) VALUE " USGS: ".
    03 GSNAME PIC X(20).
    03 FILLER PIC XXX.
    03  FILLER  PIC X(7) VALUE "SOURCES".
    03  FILLER PIC X.
    03 SRCEID PIC X(50).
    03 SRCNAM PIC X(30).
    03  CNLZZZ PIC X(6).
    03 FILLER PIC X(8).
01  HDR3.
    03  CC PIC 9 VALUE 0.
    03  PCNG PIC X(5) VALUE " CNG ".
    03  PLCWL PIC X(20) VALUE "  LOCAL WELL ID     ".
    03  PSITE PIC X(16) VALUE "    SITE ID    ".
    03  PDUP PIC XX VALUE " D".
    03  PCR PIC X(5) VALUE " C**R".
    03  PITEM PIC X(5) VALUE "  C* ".
    03  PNDAT PIC X(25) VALUE "        NEW DATA          ".
    03  PCOND PIC X(25) VALUE "         CONDITION        ".
    03  PSRCE PIC X(12) VALUE "  SOURCE    ".
    03  PCCR PIC X VALUE "D".
    03  FILLER PIC XX VALUE "--".
    03  PCCSR PIC X VALUE "S".
    03  PNOTE PIC X(12) VALUE "    NOTES  ".
01  PRNTLN.
    03  CC  PIC X VALUE SPACE.
    03  VSN    PIC 9(5).
    03  VSNA   PIC X.
    03  PRD2   PIC 9(5).
    03  KSEQ   PIC 9(4).
    03  KEYID.
```

```
        05 KEYLAT.
           07  LATDEG    PIC 99.
           07  LATMIN    PIC 99.
           07  LATSEC    PIC 99.
        05 KEYLNG.
           07  LNGDEG    PIC 999.
           07  LNGMIN    PIC 99.
           07  LNGSEC    PIC 99.
        05 KEYDUP PIC 99.
     03 CSCDE      PIC X.
     03 RNUM       PIC 999.
     03 CNUM       PIC 999.
     03 CCR        PIC X.
     03 CCSR       PIC X.
     03 TYPDAT     PIC 99.
     03 DLEN       PIC 99.
     03 NEWDAT     PIC X(45).
     03 NSEQ       PIC 9(8).
     03 ERPRINT    PIC X(30).
01 CCARD.
     03 VSN     PIC 9(5).
     03 VSNA    PIC X.
     03 PRD2    PIC 9(5).
     03 KSEQ    PIC 9(4).
     03 KEYID.
        05 KEYLAT.
           07  LATDEG    PIC 99.
           07  LATMIN    PIC 99.
           07  LATSEC    PIC 99.
        05 KEYLNG.
           07  LNGDEG    PIC 999.
           07  LNGMIN    PIC 99.
           07  LNGSEC    PIC 99.
        05 KEYDUP PIC 99.
     03 CSCDE      PIC X.
     03 RNUM       PIC 999.
     03 CNUM       PIC 999.
     03 CCR        PIC X.
     03 CCSR       PIC X.
     03 TYPDAT     PIC 99.
     03 DLEN       PIC 99.
     03 NEWDAT     PIC X(45).
     03 NSEQ       PIC 9(8).
     03 FILETY PIC XX.
01 SITELN.
     03 CC PIC X VALUE "0".
     03 FILLER PIC X(5) VALUE " *** ".
     03 FILLER PIC X(7) VALUE "ERROR =".
     03 BTE PIC ZZ9.
     03 FILLER PIC X(5).
     03 FILLER PIC X(5) VALUE " *** ".
     03 FILLER PIC X(112).
```

```
01  DATAA.
    03  ID-CNL PIC IS XXX.
    03  ID-NUM PIC IS 999.
    03  TITLEX.
        05  SRCAGC.
            07  GNAME PIC X(20).
            07  CNAME PIC X(30).
        05  FILLER PIC X(10).
    03  FILLER PIC X(14).
01  KEYIDL.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 99.
01  KEYIDC.
    03  KEYLAT.
        05  LATDEG PIC 99.
        05  LATMIN PIC 99.
        05  LATSEC PIC 99.
    03 KEYLNG.
        05  LNGDEG PIC 999.
        05  LNGMIN PIC 99.
        05  LNGSEC PIC 99.
    03  KEYDUP PIC 99.
01  KEYIDD.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 89.
01  DELTSS.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 99.
01  ANYTAB.
    02  RECTAB OCCURS 50 TIMES.
    03  VSN    PIC 9(5).
    03  VSNA   PIC X.
    03  PRD2   PIC 9(5).
```

```
    03  KSEQ   PIC 9(4).
    03  KEYID.
        05 KEYLAT.
           07  LATDEG   PIC 99.
           07  LATMIN   PIC 99.
           07  LATSEC   PIC 99.
        05 KEYLNG.
           07  LNGDEG   PIC 999.
           07  LNGMIN   PIC 99.
           07  LNGSEC   PIC 99.
        05 KEYDUP PIC 99.
    03  CSCDE      PIC X.
    03  RNUM       PIC 999.
    03  CNUM       PIC 999.
    03  CCR        PIC X.
    03  CCSR       PIC X.
    03  TYPDAT     PIC 99.
    03  DLEN       PIC 99.
    03  NEWDAT     PIC X(45).
    03  NSEQ       PIC 9(8).
    03  ERRCODE    PIC 99.
01 CHKSIT.
    03  LC1 PIC X.
    03  LC2 PIC X.
    03  LC3 PIC 99.
    03  LC4 PIC X.
    03  LC5 PIC 99.
    03  LC6 PIC X.
    03  LC7 PIC 99.
    03  LC8 PIC X.
    03  LC9 PIC X.
    03  LC10 PIC X.
    03  LC11 PIC X.
    03  RESTLC.
        05 R1 PIC X.
        05 R2 PIC X.
        05 R3 PIC X.
        05 R4 PIC X.
        05 R5 PIC X.
        05 R6 PIC X.
01  OLDMAS.
    03  VSN     PIC 9(5).
    03  VSNA    PIC X.
    03  PRD2    PIC 9(5).
    03  KSEQ    PIC 9(4).
    03  KEYID.
        05 KEYLAT.
           07  LATDEG   PIC 99.
           07  LATMIN   PIC 99.
           07  LATSEC   PIC 99.
        05 KEYLNG.
           07  LNGDEG   PIC 999.
           07  LNGMIN   PIC 99.
           07  LNGSEC   PIC 99.
        05 KEYDUP PIC 99.
```

```
    03  CSCDE      PIC X.
    03  RNUM       PIC 999.
    03  CNUM       PIC 999.
    03  CCR        PIC X.
    03  CCSR       PIC X.
    03  TYPDAT     PIC 99.
    03  DLEN       PIC 99.
    03  NEWDAT     PIC X(45).
    03  NSEQ       PIC 9(8).
    03  ERRCODE    PIC 99.
01  ERRCNG.
    03  VSN     PIC 9(5).
    03  VSNA    PIC X.
    03  PRD2    PIC 9(5).
    03  KSEQ    PIC 9(4).
    03  KEYID.
        05 KEYLAT.
           07  LATDEG    PIC 99.
           07  LATMIN    PIC 99.
           07  LATSEC    PIC 99.
        05 KEYLNG.
           07  LNGDEG    PIC 999.
           07  LNGMIN    PIC 99.
           07  LNGSEC    PIC 99.
        05 KEYDUP PIC 99.
    03  CSCDE      PIC X.
    03  RNUM       PIC 999.
    03  CNUM       PIC 999.
    03  CCR        PIC X.
    03  CCSR       PIC X.
    03  TYPDAT     PIC 99.
    03  DLEN       PIC 99.
    03  NEWDAT     PIC X(45).
    03  NSEQ       PIC 9(8).
01  ALLSWS.
    03  S001 PIC 9.
    03  S002 PIC 9.
    03  S003 PIC 9.
    03  S004 PIC 9.
    03  S005 PIC 9.
    03  S006 PIC 9.
    03  S007 PIC 9.
    03  S008 PIC 9.
    03  S009 PIC 9.
    03  S010 PIC 9.
    03  S011 PIC 9.
01  STAR-LINE.
    03  FILLER  PIC X VALUE SPACES.
    03  FILLER  PIC X(130) VALUE ALL "*".
01  EDIT-ERRORS.
    03  ED-ER1  PIC X(30) VALUE "INVALID RECORD NUMBER-RNUM  ".
    03  ED-ER2  PIC X(30) VALUE "REQUIRED DATA FIELD MISSING".
    03  ED-ER3  PIC X(30) VALUE "CHANGE < 1 OR > 25 CHARS".
    03  ED-ER4  PIC X(30) VALUE "DISTRICT CODE NOT '04' - AZ".
    03  ED-ER5  PIC X(30) VALUE "INVALID ALPHA CODE IN CHANGE".
```

```
    03  ED-ER6  PIC X(30) VALUE "STATE CODE NOT '04' - AZ".
    03  ED-ER7  PIC X(30) VALUE "COUNTY TD IS NOT VALID CODE".
    03  ED-ER8  PIC X(30) VALUE "INVALID QUAD IN LOCAL ID".
    03  ED-ER9  PIC X(30) VALUE "INVALID LOCAL IDENTIFIER".
    03  ED-ER10 PIC X(30) VALUE "ALT OUTSIDE ACCEPTABLE RANGE".
    03  ED-ER11 PIC X(30) VALUE "DATE IS NOT NUMERIC".
    03  ED-ER12 PIC X(30) VALUE "DEPTH OF HOLE IS NOT VALID".
    03  ED-ER13 PIC X(30) VALUE "DEPTH OF WELL IS NOT VALID".
    03  ED-ER14 PIC X(30) VALUE "ENTRY NUMBER IS NOT NUMERIC".
    03  ED-ER15 PIC X(30) VALUE "INVALID LITHOLOGIC CODE".
    03  ED-ER16 PIC X(30) VALUE "SOURCE AGENCY NOT VALID".
    03  ED-ER17 PIC X(30) VALUE "DATE IS LATER THAN TODAY DATE".
    03  ED-ER18 PIC X(30) VALUE "MANDATORY FIELD IS BLANK".
    03  ED-ER19 PIC X(30) VALUE "QW CODE NOT VALID STORET CODE".
    03  ED-ER20 PIC X(30) VALUE "MESSAGE 20 AVAILABLE".
    03  ED-ER21 PIC X(30) VALUE "MESSAGE 21 AVAILABLE".
    03  ED-ER22 PIC X(30) VALUE "MESSAGE 22 AVAILABLE".
    03  ED-ER23 PIC X(30) VALUE "MESSAGE 23 AVAILABLE".
    03  ED-ER24 PIC X(30) VALUE "MESSAGE 24 AVAILABLE".
    03  ED-ER25 PIC X(30) VALUE "MESSAGE 25 AVAILABLE".
01  ERR-MSG-TABLE REDEFINES EDIT-ERRORS.
    03  ER-MSG  PIC X(30) OCCURS 25 TIMES.



PROCEDURE DIVISION.
MAIN-PROC-00.
    PERFORM BEGIN.
    PERFORM READ-C.
    PERFORM INIT-EDIT.
    PERFORM EDIT-CHANGE UNTIL EOFSW > -1.
    PERFORM FINALA.
BEGIN.
    OPEN INPUT CARD-FILE, CHANGE-FILE.
    OPEN OUTPUT LIST-FILE, FILEA, FILEB.
    ACCEPT WKDATE FROM DATE.
    MOVE WKDATE TO TDYDAT.
    MOVE MM TO NMM.
    MOVE MONTHY (NMM) TO RMTH OF HDR1.
    MOVE DY TO RDY OF HDR1.
    MOVE YY TO RYR OF HDR1.
READ-C.
    READ CARD-FILE AT END GO TO STOP1.
    IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
    IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
    ADD 1 TO INCRDS.
    MOVE CNL-CARD TO PCNL.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
        TO TITLEA OF HDR1 ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
```

```
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 011
         MOVE GNAME TO GSNAME
         MOVE CNAME TO SRCNAM
    ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    READ CARD-FILE INTO DATAA
        AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 012
        MOVE SRCAGC TO SRCEID,
    ELSE GO TO STOP4.
    MOVE GO-CNL TO CNLZZZ.
    ADD 1 TO INCRDS.
INIT-EDIT.
    PERFORM CLEAR-PROC.
    PERFORM READ-CHANGE.
    IF EOFSW > -1, GO TO STOP3.
    MOVE KEYID OF CNGREC TO KEYIDL.
CLEAR-PROC.
    MOVE ALFAM TO ERRSW.
    MOVE NU0 TO NMM.
    ADD NU1 TO BT, LNCNT.
    MOVE NU0 TO DCOUNT, MM, YY, CC OF PRNTLN.
    MOVE BT TO NSEQ OF PRNTLN, NSEQ OF RECOUT
    MOVE ALFAN TO C-END, PROC-SW.
EDIT-CHANGE.
    PERFORM STORE-REC UNTIL C-END = ALFAY OR EOFSW > -1.
    PERFORM ALL-PROC UNTIL PROC-SW = ALFAY.
    PERFORM WRITE-LIST.
READ-CHANGE.
    READ CHANGE-FILE,
        AT END MOVE 1 TO EOFSW.
    IF EOFSW < NU1,
        ADD NU1 TO CNRECN, NFLEIA, TFILEC.
STORE-REC.
    IF NMM > 50 GO TO STOP9.
    PERFORM KSEQ-CHECK.
    IF SEQSW = ALFAY,
        ADD NU1 TO NMM
        MOVE CNGREC TO RECTAB (NMM)
        MOVE ALFAN TO C-END,
        PERFORM READ-CHANGE
    ELSE
        MOVE ALFAY TO C-END.
KSEQ-CHECK.
    IF KSEQ OF CNGREC > DCOUNT,
        MOVE KSEQ OF CNGREC TO DCOUNT,
        MOVE ALFAY TO SEQSW,
    ELSE
        MOVE ALFAN TO SEQSW.
```

```
ALL-PROC.
     ADD NU1 TO MM.
     IF (MM = NMM OR MM > NMM)
         MOVE ALFAY TO PROC-SW
     ELSE
         MOVE ALFAN TO PROC-SW.
     MOVE RECTAB (MM) TO OLDMAS.
     MOVE ZEROS TO ERCODE.
     MOVE ALFAN TO ERRSW.
     IF CCSR OF OLDMAS = ALFAA
         PERFORM DIST-CHK.
     IF (CCSR OF OLDMAS = ALFAD OR ALFAV)
         PERFORM ENDEDT
     ELSE
         PERFORM CHNG-EXAM.
CHNG-EXAM.
     PERFORM NEW-EXAM.
     PERFORM EXAM-CHANGE.
NEW-EXAM.
     MOVE NU0 TO DECSW, DGCNT, DECNT, NEGSW.
     MOVE ZEROS TO NUMER, NUMSOUT, SIGNUMS, TESTNUM, NNUM.
     IF DLEN OF OLDMAS = 01, MOVE NEWDAT OF OLDMAS TO X1.
     IF DLEN OF OLDMAS = 02, MOVE NEWDAT OF OLDMAS TO X2.
     IF DLEN OF OLDMAS = 03, MOVE NEWDAT OF OLDMAS TO X3.
     IF DLEN OF OLDMAS = 04, MOVE NEWDAT OF OLDMAS TO X4.
     IF DLEN OF OLDMAS = 05, MOVE NEWDAT OF OLDMAS TO X5.
     IF DLEN OF OLDMAS = 06, MOVE NEWDAT OF OLDMAS TO X6.
     IF DLEN OF OLDMAS = 07, MOVE NEWDAT OF OLDMAS TO X7.
     IF DLEN OF OLDMAS = 08, MOVE NEWDAT OF OLDMAS TO X8.
     IF DLEN OF OLDMAS = 09, MOVE NEWDAT OF OLDMAS TO X9.
     IF DLEN OF OLDMAS = 10, MOVE NEWDAT OF OLDMAS TO X10.
     IF DLEN OF OLDMAS = 11, MOVE NEWDAT OF OLDMAS TO X11.
     IF DLEN OF OLDMAS = 12, MOVE NEWDAT OF OLDMAS TO X12.
     IF DLEN OF OLDMAS = 13, MOVE NEWDAT OF OLDMAS TO X13.
     IF DLEN OF OLDMAS = 14, MOVE NEWDAT OF OLDMAS TO X14.
     IF DLEN OF OLDMAS = 15, MOVE NEWDAT OF OLDMAS TO X15.
     IF DLEN OF OLDMAS = 16, MOVE NEWDAT OF OLDMAS TO X16.
     IF DLEN OF OLDMAS = 17, MOVE NEWDAT OF OLDMAS TO X17.
     IF DLEN OF OLDMAS = 18, MOVE NEWDAT OF OLDMAS TO X18.
     IF DLEN OF OLDMAS = 19, MOVE NEWDAT OF OLDMAS TO X19.
     IF DLEN OF OLDMAS = 20, MOVE NEWDAT OF OLDMAS TO X20.
     IF DLEN OF OLDMAS = 21, MOVE NEWDAT OF OLDMAS TO X21.
     IF DLEN OF OLDMAS = 22, MOVE NEWDAT OF OLDMAS TO X22.
     IF DLEN OF OLDMAS = 23, MOVE NEWDAT OF OLDMAS TO X23.
     IF DLEN OF OLDMAS = 24, MOVE NEWDAT OF OLDMAS TO X24.
     IF DLEN OF OLDMAS = 25, MOVE NEWDAT OF OLDMAS TO X25.
     IF DLEN OF OLDMAS < NU1 OR DLEN OF OLDMAS > 25,
         MOVE ALFAX TO ERRSW,
         MOVE NU3 TO ERRCODE OF OLDMAS.
EXAM-CHANGE.
     IF RNUM OF OLDMAS = 000
         MOVE ALFAY TO RNSW
         PERFORM PTR000.
```

```
IF RNUM OF OLDMAS = 001
    MOVE ALFAY TO RNSW
    PERFORM PTR001.
IF RNUM OF OLDMAS = 002
    MOVE ALFAY TO RNSW
    PERFORM PTR002.
IF RNUM OF OLDMAS = 003
    MOVE ALFAY TO RNSW
    PERFORM PTR003.
IF RNUM OF OLDMAS = 058
    MOVE ALFAY TO RNSW
    PERFORM PTR058.
IF RNUM OF OLDMAS = 076
    MOVE ALFAY TO RNSW
    PERFORM PTR076.
IF RNUM OF OLDMAS = 198
    MOVE ALFAY TO RNSW
    PERFORM PTR198.
IF RNUM OF OLDMAS = 042
    MOVE ALFAY TO RNSW
    PERFORM PTR042.
IF RNUM OF OLDMAS = 082
    MOVE ALFAY TO RNSW
    PERFORM PTR082.
IF RNUM OF OLDMAS = 002
    MOVE ALFAY TO RNSW
    PERFORM PTR002.
IF RNUM OF OLDMAS = 001
    MOVE ALFAY TO RNSW
    PERFORM PTR001.
IF RNUM OF OLDMAS = 158
    MOVE ALFAY TO RNSW
    PERFORM PTR158.
IF RNUM OF OLDMAS = 192
    MOVE ALFAY TO RNSW
    PERFORM PTR192.
IF RNUM OF OLDMAS = 146
    MOVE ALFAY TO RNSW
    PERFORM PTR146.
IF RNUM OF OLDMAS = 186,
    MOVE ALFAY TO RNSW,
    PERFORM PTR186.
IF RNUM OF OLDMAS = 090,
    MOVE ALFAY TO RNSW,
    PERFORM PTR090.
IF RNUM OF OLDMAS = 114,
    MOVE ALFAY TO RNSW,
    PERFORM PTR114.
IF RNUM OF OLDMAS = 121,
    MOVE ALFAY TO RNSW,
    PERFORM PTR121.
IF RNUM OF OLDMAS = 189,
    MOVE ALFAY TO RNSW,
    PERFORM PTR189.
```

```
IF RNUM OF OLDMAS = 094,
    MOVE ALFAY TO RNSW,
    PERFORM PTR094.
IF RNUM OF OLDMAS = 183,
    MOVE ALFAY TO RNSW,
    PERFORM PTR183.
IF RNUM OF OLDMAS = 072,
    MOVE ALFAY TO RNSW,
    PERFORM PTR072.
IF RNUM OF OLDMAS = 047,
    MOVE ALFAY TO RNSW,
    PERFORM PTR047.
IF RNUM OF OLDMAS = 134,
    MOVE ALFAY TO RNSW,
    PERFORM PTR134.
IF RNUM OF OLDMAS = 171,
    MOVE ALFAY TO RNSW,
    PERFORM PTR171.
IF RNUM OF OLDMAS = 180,
    MOVE ALFAY TO RNSW,
    PERFORM PTR180.
IF RNUM OF OLDMAS = 127,
    MOVE ALFAY TO RNSW,
    PERFORM PTR127.
IF RNUM OF OLDMAS = 055
    MOVE ALFAY TO RNSW,
    PERFORM PTR055.
IF RNUM OF OLDMAS = 098,
    MOVE ALFAY TO RNSW,
    PERFORM PTR098.
IF RNUM OF OLDMAS = 105,
    MOVE ALFAY TO RNSW,
    PERFORM PTR105.
IF RNUM OF OLDMAS = 164,
    MOVE ALFAY TO RNSW,
    PERFORM PTR164.
IF RNUM OF OLDMAS = 203,
    MOVE ALFAY TO RNSW,
    PERFORM PTR203.
IF RNUM OF OLDMAS = 208,
    MOVE ALFAY TO RNSW,
    PERFORM PTR208.
IF RNUM OF OLDMAS = 212,
    MOVE ALFAY TO RNSW,
    PERFORM PTR212.
IF RNUM OF OLDMAS = 219,
    MOVE ALFAY TO RNSW,
    PERFORM PTR219.
IF RNUM OF OLDMAS = 250,
    MOVE ALFAY TO RNSW,
    PERFORM PTR250.
IF RNUM OF OLDMAS = 277,
    MOVE ALFAY TO RNSW,
    PERFORM PTR277.
```

```
     IF RNUM OF OLDMAS = 234,
         MOVE ALFAY TO RNSW,
         PERFORM PTR234.
     IF RNUM OF OLDMAS = 320,
         MOVE ALFAY TO RNSW,
         PERFORM PTR320.
      IF NOT RNSW = ALFAY,
            MOVE ALFAX TO ERRSW,
            MOVE NU1 TO ERCODE,
            MOVE ALFAY TO RNSW.
DIST-CHK.
     MOVE ZEROS TO ALLSWS.
     MOVE NU1 TO BTA, BTB.
     IF RNUM OF OLDMAS = 000, PERFORM STA000 VARYING BTA
         FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 042, PERFORM STA042 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 047, PERFORM STA047 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 055, PERFORM STA055 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 058, PERFORM STA058 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 072, PERFORM STA072 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 076, PERFORM STA076 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 082, PERFORM STA082 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 090, PERFORM STA090 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 094, PERFORM STA094 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 114, PERFORM STA114 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 121, PERFORM STA121 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 127, PERFORM STA127 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 134, PERFORM STA134-146 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 146, PERFORM STA134-146 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 158, PERFORM STA158 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 180, PERFORM STA180 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 183, PERFORM STA183 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 186, PERFORM STA186 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
     ELSE IF RNUM OF OLDMAS = 189, PERFORM STA189 VARYING BTA
               FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
```

```
    ELSE IF RNUM OF OLDMAS = 192, PERFORM STA192 VARYING BTA
             FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
    ELSE IF RNUM OF OLDMAS = 198, PERFORM STA198 VARYING BTA
             FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
    ELSE IF RNUM OF OLDMAS = 234, PERFORM STA234 VARYING BTA
             FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM,
    ELSE IF RNUM OF OLDMAS = 320, PERFORM STA320 VARYING BTA
             FROM 1 BY 1 UNTIL BTA = NMM OR BTA > NMM.
PTR000.
    IF CNUM OF OLDMAS = 002, PERFORM EC002.
    IF CNUM OF OLDMAS = 003, PERFORM EC003.
    IF CNUM OF OLDMAS = 004, PERFORM EC004.
    IF CNUM OF OLDMAS = 005, PERFORM EC005.
    IF CNUM OF OLDMAS = 006, PERFORM EC006.
    IF CNUM OF OLDMAS = 007, PERFORM EC007.
    IF CNUM OF OLDMAS = 008, PERFORM EC008.
    IF CNUM OF OLDMAS = 011, PERFORM EC011.
    IF CNUM OF OLDMAS = 012, PERFORM EC012.
    IF CNUM OF OLDMAS = 013, PERFORM EC013.
    IF CNUM OF OLDMAS = 014, PERFORM EC014.
    IF CNUM OF OLDMAS = 015, PERFORM EC015.
    IF CNUM OF OLDMAS = 016, PERFORM EC016.
    IF CNUM OF OLDMAS = 017, PERFORM EC017.
    IF CNUM OF OLDMAS = 018, PERFORM EC018.
    IF CNUM OF OLDMAS = 019, PERFORM EC019.
    IF CNUM OF OLDMAS = 021, PERFORM EC021.
    IF CNUM OF OLDMAS = 023, PERFORM EC023.
    IF CNUM OF OLDMAS = 024, PERFORM EC024.
    IF CNUM OF OLDMAS = 027, PERFORM EC027.
    IF CNUM OF OLDMAS = 028, PERFORM EC028.
    IF CNUM OF OLDMAS = 029, PERFORM EC029.
    IF CNUM OF OLDMAS = 030, PERFORM EC030.
    IF CNUM OF OLDMAS = 031, PERFORM EC031.
    IF CNUM OF OLDMAS = 033, PERFORM EC033.
    IF CNUM OF OLDMAS = 034, PERFORM EC034.
    IF CNUM OF OLDMAS = 266, PERFORM EC266.
    IF CNUM OF OLDMAS = 267, PERFORM EC267.
    IF CNUM OF OLDMAS = 301, PERFORM EC301.
    IF CNUM OF OLDMAS = 302, PERFORM EC302.
PTR001. EXIT.
PTR002. EXIT.
PTR003. EXIT.
PTR042.
    IF CNUM OF OLDMAS = 254, PERFORM EC254.
    IF CNUM OF OLDMAS = 043, PERFORM EC043.
    IF CNUM OF OLDMAS = 044, PERFORM EC044.
    IF CNUM OF OLDMAS = 045, PERFORM EC045.
    IF CNUM OF OLDMAS = 046, PERFORM EC046.
    IF CNUM OF OLDMAS = 038, PERFORM EC038.
PTR047.
    IF CNUM OF OLDMAS = 048, PERFORM EC048.
    IF CNUM OF OLDMAS = 043, PERFORM EC043.
    IF CNUM OF OLDMAS = 254, PERFORM EC254.
    IF CNUM OF OLDMAS = 049, PERFORM EC049.
    IF CNUM OF OLDMAS = 050, PERFORM EC050.
```

```
    IF CNUM OF OLDMAS = 051, PERFORM EC051.
    IF CNUM OF OLDMAS = 255, PERFORM EC255.
    IF CNUM OF OLDMAS = 266, PERFORM EC266.
PTR055.
    IF CNUM OF OLDMAS = 043, PERFORM EC043.
    IF CNUM OF OLDMAS = 254, PERFORM EC254.
    IF CNUM OF OLDMAS = 056, PERFORM EC056.
    IF CNUM OF OLDMAS = 057, PERFORM EC057.
PTR058.
    IF CNUM OF OLDMAS = 059, PERFORM EC059.
    IF CNUM OF OLDMAS = 060, PERFORM EC060.
    IF CNUM OF OLDMAS = 063, PERFORM EC063.
    IF CNUM OF OLDMAS = 064, PERFORM EC064.
    IF CNUM OF OLDMAS = 065, PERFORM EC065.
    IF CNUM OF OLDMAS = 066, PERFORM EC066.
    IF CNUM OF OLDMAS = 067, PERFORM EC067.
    IF CNUM OF OLDMAS = 068, PERFORM EC068.
    IF CNUM OF OLDMAS = 069, PERFORM EC069.
    IF CNUM OF OLDMAS = 070, PERFORM EC070.
    IF CNUM OF OLDMAS = 071, PERFORM EC071.
PTR072.
    IF CNUM OF OLDMAS = 059, PERFORM EC059.
    IF CNUM OF OLDMAS = 060, PERFORM EC060.
    IF CNUM OF OLDMAS = 073, PERFORM EC073.
    IF CNUM OF OLDMAS = 074, PERFORM EC074.
    IF CNUM OF OLDMAS = 075, PERFORM EC075.
PTR076.
    IF CNUM OF OLDMAS = 059, PERFORM EC059.
    IF CNUM OF OLDMAS = 060, PERFORM EC060.
    IF CNUM OF OLDMAS = 077, PERFORM EC077.
    IF CNUM OF OLDMAS = 078, PERFORM EC078.
    IF CNUM OF OLDMAS = 079, PERFORM EC079.
    IF CNUM OF OLDMAS = 080, PERFORM EC080.
PTR082.
    IF CNUM OF OLDMAS = 059, PERFORM EC059.
    IF CNUM OF OLDMAS = 060, PERFORM EC060.
    IF CNUM OF OLDMAS = 083, PERFORM EC083.
    IF CNUM OF OLDMAS = 084, PERFORM EC084.
    IF CNUM OF OLDMAS = 085, PERFORM EC085.
    IF CNUM OF OLDMAS = 086, PERFORM EC086.
    IF CNUM OF OLDMAS = 087, PERFORM EC087.
    IF CNUM OF OLDMAS = 088, PERFORM EC088.
    IF CNUM OF OLDMAS = 089, PERFORM EC089.
PTR090.
    IF CNUM OF OLDMAS = 256, PERFORM EC256.
    IF CNUM OF OLDMAS = 091, PERFORM EC091.
    IF CNUM OF OLDMAS = 092, PERFORM EC092.
    IF CNUM OF OLDMAS = 093, PERFORM EC093.
    IF CNUM OF OLDMAS = 096, PERFORM EC096.
    IF CNUM OF OLDMAS = 097, PERFORM EC097.
    IF CNUM OF OLDMAS = 304, PERFORM EC304.
PTR094.
    IF CNUM OF OLDMAS = 256, PERFORM EC256.
    IF CNUM OF OLDMAS = 095, PERFORM EC095.
    IF CNUM OF OLDMAS = 126, PERFORM EC126.
    IF CNUM OF OLDMAS = 132, PERFORM EC132.
```

```
PTR098. EXIT.
PTR105. EXIT.
PTR114.
    IF CNUM OF OLDMAS = 115, PERFORM EC115.
    IF CNUM OF OLDMAS = 116, PERFORM EC116.
    IF CNUM OF OLDMAS = 117, PERFORM EC117.
    IF CNUM OF OLDMAS = 118, PERFORM EC118.
    IF CNUM OF OLDMAS = 257, PERFORM EC257.
    IF CNUM OF OLDMAS = 120, PERFORM EC120.
    IF CNUM OF OLDMAS = 307, PERFORM EC307.
PTR121.
    IF CNUM OF OLDMAS = 122, PERFORM EC122.
    IF CNUM OF OLDMAS = 123, PERFORM EC123.
    IF CNUM OF OLDMAS = 124, PERFORM EC124.
    IF CNUM OF OLDMAS = 125, PERFORM EC125.
    IF CNUM OF OLDMAS = 258, PERFORM EC258.
PTR127.
    IF CNUM OF OLDMAS = 128, PERFORM EC128.
    IF CNUM OF OLDMAS = 129, PERFORM EC129.
    IF CNUM OF OLDMAS = 130, PERFORM EC130.
    IF CNUM OF OLDMAS = 131, PERFORM EC131.
    IF CNUM OF OLDMAS = 259, PERFORM EC259.
    IF CNUM OF OLDMAS = 133, PERFORM EC133.
PTR134.
    IF CNUM OF OLDMAS = 147, PERFORM EC147.
    IF CNUM OF OLDMAS = 148, PERFORM EC148.
    IF CNUM OF OLDMAS = 150, PERFORM EC150.
    IF CNUM OF OLDMAS = 151, PERFORM EC151.
    IF CNUM OF OLDMAS = 152, PERFORM EC152.
    IF CNUM OF OLDMAS = 153, PERFORM EC153.
    IF CNUM OF OLDMAS = 154, PERFORM EC154.
    IF CNUM OF OLDMAS = 156, PERFORM EC156.
    IF CNUM OF OLDMAS = 157, PERFORM EC157.
    IF CNUM OF OLDMAS = 272, PERFORM EC272.
    IF CNUM OF OLDMAS = 309, PERFORM EC309.
PTR146.
    IF CNUM OF OLDMAS = 147, PERFORM EC147.
    IF CNUM OF OLDMAS = 148, PERFORM EC148.
    IF CNUM OF OLDMAS = 150, PERFORM EC150.
    IF CNUM OF OLDMAS = 151, PERFORM EC151.
    IF CNUM OF OLDMAS = 152, PERFORM EC152.
    IF CNUM OF OLDMAS = 153, PERFORM EC153.
    IF CNUM OF OLDMAS = 154, PERFORM EC154.
    IF CNUM OF OLDMAS = 156, PERFORM EC156.
    IF CNUM OF OLDMAS = 157, PERFORM EC157.
    IF CNUM OF OLDMAS = 272, PERFORM EC272.
    IF CNUM OF OLDMAS = 309, PERFORM EC309.
PTR158.
    IF CNUM OF OLDMAS = 159, PERFORM EC159.
    IF CNUM OF OLDMAS = 161, PERFORM EC161.
    IF CNUM OF OLDMAS = 162, PERFORM EC162.
    IF CNUM OF OLDMAS = 163, PERFORM EC163.
PTR164. EXIT.
PTR171. EXIT.
```

```
PTR180.
     IF CNUM OF OLDMAS = 181, PERFORM EC181.
     IF CNUM OF OLDMAS = 182, PERFORM EC182.
     IF CNUM OF OLDMAS = 261, PERFORM EC261.
     IF CNUM OF OLDMAS = 312, PERFORM EC312.
PTR183.
     IF CNUM OF OLDMAS = 311, PERFORM EC311.
     IF CNUM OF OLDMAS = 185, PERFORM EC185.
PTR186.
     IF CNUM OF OLDMAS = 187, PERFORM EC187.
     IF CNUM OF OLDMAS = 188, PERFORM EC188.
PTR189.
     IF CNUM OF OLDMAS = 190, PERFORM EC190.
     IF CNUM OF OLDMAS = 191, PERFORM EC191.
PTR192.
     IF CNUM OF OLDMAS = 193, PERFORM EC193.
     IF CNUM OF OLDMAS = 195, PERFORM EC195.
     IF CNUM OF OLDMAS = 196, PERFORM EC196.
     IF CNUM OF OLDMAS = 197, PERFORM EC197.
PTR198.
     IF CNUM OF OLDMAS = 199, PERFORM EC199.
     IF CNUM OF OLDMAS = 200, PERFORM EC200.
     IF CNUM OF OLDMAS = 201, PERFORM EC201.
     IF CNUM OF OLDMAS = 202, PERFORM EC202.
PTR203. EXIT.
PTR208. EXIT.
PTR212. EXIT.
PTR219. EXIT.
PTR234.
     IF CNUM OF OLDMAS = 235, PERFORM EC235.
     IF CNUM OF OLDMAS = 237, PERFORM EC237.
     IF CNUM OF OLDMAS = 238, PERFORM EC238.
     IF CNUM OF OLDMAS = 239, PERFORM EC239.
PTR250. EXIT.
PTR277. EXIT.
PTR320.
     IF CNUM OF OLDMAS = 321, PERFORM EC321.
     IF CNUM OF OLDMAS = 322, PERFORM EC322.
     IF CNUM OF OLDMAS = 323, PERFORM EC323.
     IF CNUM OF OLDMAS = 324, PERFORM EC324.
STA000.
     MOVE ALFAM TO CCSR OF RECTAB (BTA).
     IF CNUM OF RECTAB (BTA) = 002, ADD NU1 TO S002.
     IF CNUM OF RECTAB (BTA) = 003, ADD NU1 TO S003.
     IF CNUM OF RECTAB (BTA) = 004, ADD NU1 TO S004.
     IF CNUM OF RECTAB (BTA) = 006, ADD NU1 TO S006.
     IF CNUM OF RECTAB (BTA) = 007, ADD NU1 TO S007.
     IF CNUM OF RECTAB (BTA) = 008, ADD NU1 TO S008.
     IF BTA = NMM PERFORM CHK000.
STA042.
     MOVE ALFAM TO CCSR OF RECTAB (BTA).
     IF CNUM OF RECTAB (BTA) = 043, ADD NU1 TO S002.
     IF BTA = NMM PERFORM CHK042.
```

```
STA047.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 043, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK047.
STA055.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 043, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 056, ADD NU1 TO S003.
    IF CNUM OF RECTAB (BTA) = 254, ADD NU1 TO S004.
    IF BTA = NMM PERFORM CHK055.
STA058.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK058.
STA072.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 073, ADD NU1 TO S003.
    IF BTA = NMM PERFORM CHK072.
STA076.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 077, ADD NU1 TO S003.
    IF CNUM OF RECTAB (BTA) = 079, ADD NU1 TO S004.
    IF BTA = NMM PERFORM CHK076.
STA082.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 059, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 083, ADD NU1 TO S003.
    IF CNUM OF RECTAB (BTA) = 085, ADD NU1 TO S004.
    IF BTA = NMM PERFORM CHK082.
STA090.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 256, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 093, ADD NU1 TO S003.
    IF CNUM OF RECTAB (BTA) = 304, ADD NU1 TO S004.
    IF BTA = NMM PERFORM CHK090.
STA094.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 256, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 095, ADD NU1 TO S003.
    IF BTA = NMM PERFORM CHK094.
STA114.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 115, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK114.
STA121.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 122, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK121.
STA127.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 128, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK127.
```

```
STA134-146.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 147, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 150, ADD NU1 TO S003.
    IF BTA = NMM PERFORM CHK134-146.
STA158.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 161, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK158.
STA180.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 181, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK180.
STA183.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 311, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 185, ADD NU1 TO S003.
    IF BTA = NMM PERFORM CHK183.
STA186.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 187, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK186.
STA189.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 190, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 191, ADD NU1 TO S003.
    IF BTA = NMM PERFORM CHK189.
STA192.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 193, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 196, ADD NU1 TO S003.
    IF CNUM OF RECTAB (BTA) = 197, ADD NU1 TO S004.
    IF BTA = NMM PERFORM CHK192.
STA198.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 199, ADD NU1 TO S002.
    IF BTA = NMM PERFORM CHK198.
STA234.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 235, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 237, ADD NU1 TO S003.
    IF BTA = NMM PERFORM CHK234.
STA320.
    MOVE ALFAM TO CCSR OF RECTAB (BTA).
    IF CNUM OF RECTAB (BTA) = 321, ADD NU1 TO S002.
    IF CNUM OF RECTAB (BTA) = 323, ADD NU1 TO S003.
    IF BTA = NMM PERFORM CHK320.
CHK000.
    IF (NU1 = S002 AND S003 AND S004 AND S006 AND S007 AND S008),
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU2 TO ERRCODE OF OLDMAS,
        PERFORM WRITE-LIST.
```

```
CHK042.
     IF (NU1 = S002 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK047.
     IF (NU1 = S002 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK055.
     IF (NU1 = S002 AND S003 AND S004 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK058.
     IF (NU1 = S002)
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK072.
     IF (NU1 = S002 AND S003 )
          MOVE NU0 TO ERRCODE OF OLDMAS,
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK076.
     IF (NU1 = S002 AND S003 AND S004 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK082.
     IF (NU1 = S002 AND S003 AND S004 )
          MOVE NU0 TO ERRCODE OF OLDMAS,
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK090.
     IF (NU1 = S002 AND S003 AND S004 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
```

```
CHK094.
     IF (NU1 = S002 AND S003 )
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK114.
     IF (NU1 = S002 )
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK121.
     IF (NU1 = S002 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK127.
     IF (NU1 = S002 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK134-146.
     IF (NU1 = S002 AND S003 )
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK158.
     IF (NU1 = S002 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK180.
     IF (NU1 = S002 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
CHK183.
     IF (NU1 = S002 AND S003 )
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
```

```
CHK186.
     IF (NU1 = S002  )
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU2 TO ERRCODE OF OLDMAS
         PERFORM WRITE-LIST.
CHK189.
     IF (NU1 = S002 AND S003 )
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU2 TO ERRCODE OF OLDMAS
         PERFORM WRITE-LIST.
CHK192.
     IF (NU1 = S002 AND S003 AND S004 )
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU2 TO ERRCODE OF OLDMAS
         PERFORM WRITE-LIST.
CHK198.
     IF (NU1 = S002 )
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU2 TO ERRCODE OF OLDMAS
         PERFORM WRITE-LIST.
CHK234.
     IF (NU1 = S002 AND S003 )
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU2 TO ERRCODE OF OLDMAS
         PERFORM WRITE-LIST.
CHK320.
     IF (NU1 = S002 AND S003 )
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU2 TO ERRCODE OF OLDMAS
          PERFORM WRITE-LIST.
EC002.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAC OR ALFAD OR ALFAH OR ALFAI OR ALFAM
             OR ALFAP OR ALFAT OR ALFAW OR ALFAX)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
```

```
EC003.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAC OR ALFAU )
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC004.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF NEWDAT OF OLDMAS = "USGS"
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC005.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (NEWDAT OF OLDMAS > XBLNK10)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*   EC006 IS NUMERIC  --  RESTON PICTURE IS 99
EC006.
     MOVE 00 TO TYPDAT OF OLDMAS.
     MOVE X2 TO N2.
     IF N2 = 04
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU4 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*   EC007 IS NUMERIC  --  RESTON PICTURE IS 99
EC007.
     MOVE 00 TO TYPDAT OF OLDMAS.
     MOVE X2 TO N2.
     IF N2 = 04
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU6 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*   EC008 IS NUMERIC  --  RESTON PICTURE IS 999
EC008.
     MOVE 00 TO TYPDAT OF OLDMAS.
     MOVE X3 TO N3.
     IF ( N3 = NU1 OR NU3 OR NU5 OR NU7 OR NU9 OR NU11
          OR NU13 OR NU15 OR NU17 OR NU19 OR NU21 OR NU23 OR
          NU25 OR NU27)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU7 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
```

```
EC011.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAS OR ALFAF OR ALFAT OR ALFAM)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC012.
    MOVE 99 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO CHKSIT.
    IF CCSR OF OLDMAS = ALFAM,
        MOVE ALFAR TO CCSR OF OLDMAS.
    IF (LC1 = ALFAA OR ALFAB OR ALFAC OR ALFAD)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE NU8 TO ERRCODE OF OLDMAS
        MOVE ALFAX TO ERRSW.
    IF LC8 = NBLNK1 MOVE ALFAU TO LC8.
    IF LC9 = NBLNK1, MOVE ALFAU TO LC9.
    IF LC10 = NBLNK1, MOVE ALFAU TO LC10.
    IF (ALFAU = LC8 OR LC9 OR LC10), MOVE NBLNK1 TO LC11.
    MOVE SPACES TO RESTLC, NEWDAT OF OLDMAS.
    PERFORM CLCSIT.
    MOVE CHKSIT TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.
EC013.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC014.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
*   EC015 IS NUMERIC  --  RESTON PICTURE IS 999999
EC015.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
*   EC016 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC016.
    MOVE 02 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO ALFNUM.
    PERFORM ALFTONUM.
    MOVE NUMER TO NUM52.
    IF (NUM52 < LOALT OR NUM52 > HIALT)
        MOVE ALFAX TO ERRSW
        MOVE NU10 TO ERRCODE OF OLDMAS
    ELSE
        MOVE NU0 TO ERRCODE OF OLDMAS
        MOVE NUM52 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.
EC017.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAL OR ALFAM)
        MOVE NU0 TO ERRCODE OF OLDMAS
```

```
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC018.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC019.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE OR ALFAF
       OR ALFAG OR ALFAH OR ALFAK OR ALFAL OR ALFAM OR ALFAO
       OR ALFAP OR ALFAS OR ALFAT OR ALFAU OR ALFAV OR ALFAW)
        MOVE ALFAX TO ERRSW
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC020.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC021.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 > XBLNK10)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU11 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC022.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC023.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA  OR ALFAD OR ALFAE OR ALFAG
       OR ALFAH OR ALFAM OR ALFAO OR ALFAP OR ALFAR OR ALFAS
       OR ALFAT OR ALFAU OR ALFAW OR ALFAX OR ALFAZ)
        MOVE ALFAX TO ERRSW
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC024.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR
     ALFAE OR ALFAF OR ALFAH OR ALFAI  OR ALFAM OR ALFAN
     OR ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAU OR ALFAY
     OR ALFAZ)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
```

```
EC025.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE
         OR ALFAF OR ALFAH OR ALFAI OR ALFAM OR ALFAN OR ALFAP
         OR ALFAR OR ALFAS OR ALFAT OR ALFAU OR ALFAY OR ALFAZ)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC026.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR
         ALFAE OR ALFAF OR ALFAH OR ALFAI OR ALFAM OR ALFAN
         OR ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAU OR ALFAY
        OR ALFAZ)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*    EC027 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC027.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     IF (NUM52 > MAXDEP OR NUM52 < NU0)
         MOVE ALFAX TO ERRSW
         MOVE NU12 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*    EC028 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC028.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     IF (NUM52 > MAXDEP OR NUM52 < NU0)
         MOVE ALFAX TO ERRSW
         MOVE NU13 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC029.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAG OR ALFAL OR ALFAO OR ALFAR
         OR ALFAD OR ALFAS OR ALFAZ)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*    EC030 IS NUMERIC  --  RESTON PICTURE IS S99999V99
EC030.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
```

```
    MOVE NUMER TO SNUM42.
    IF NOT (NEGSW = NU0), COMPUTE SNUM42 = (-1 * SNUM42).
    MOVE SNUM42 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.
EC031.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 > XBLNK10)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU11 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC032.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC033.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO
        OR ALFAR OR ALFAS OR ALFAZ)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC034.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAC OR ALFAE OR ALFAG OR ALFAH OR ALFAL
         OR ALFAM OR ALFAR OR ALFAS OR ALFAT OR ALFAV OR ALFAZ)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC036.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO
        OR ALFAR OR ALFAS OR ALFAZ)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC037.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAD OR ALFAF OR ALFAG OR ALFAH OR ALFAO
        OR ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAV OR
        ALFAX OR ALFAZ)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC038.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
```

```
EC043.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAJ OR ALFAP
         OR ALFAR OR ALFAS OR ALFAT OR ALFAU OR ALFAZ )
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC044.
     MOVE 00 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC045.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAD OR ALFAE OR ALFAG OR ALFAH OR ALFAL
         OR  ALFAN OR ALFAW OR ALFAZ OR XBLNK1)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*   EC046 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC046.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM42.
     MOVE NUM42 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
EC048.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC049.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC050.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC052.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
*   EC053 IS NUMERIC  --  RESTON PICTURE IS 99V99999
EC053.
     MOVE 05 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM25.
     MOVE NUM25 TO NEWDAT OF OLDMAS.
EC051.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC056.
     MOVE 99 TO TYPDAT OF OLDMAS.
       IF (X1 = ALFAD OR ALFAE OR ALFAG OR ALFAH OR
           ALFAL OR ALFAN OR ALFAW OR ALFAZ)
         MOVE NU0 TO ERRCODE OF OLDMAS
```

```
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC057.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM42.
     MOVE NUM42 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
*    EC059 IS NUMERIC  --  RESTON PICTURE IS 999
 EC059.
     MOVE 00 TO TYPDAT OF OLDMAS.
     IF (X3 > XBLNK3)
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU14 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC060.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC063.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC064.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO OR
          ALFAR OR ALFAS OR ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC065.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAH OR
          ALFAJ OR ALFAP OR ALFAR OR ALFAT OR ALFAV OR ALFAw OR
          ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC066.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAC OR ALFAF OR ALFAG OR ALFAH OR ALFAO OR
          ALFAP OR ALFAS OR ALFAT OR ALFAw OR ALFAX OR ALFAZ OR
          XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
```

```
 EC067.
      MOVE 99 TO TYPDAT OF OLDMAS.
      IF NOT (X1 = ALFAB OR ALFAC OR ALFAG OR ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
*    EC068 IS NUMERIC  --  RESTON PICTURE IS 9999
 EC068.
      MOVE 00 TO TYPDAT OF OLDMAS.
      PERFORM ENDEDT.
 EC069.
      MOVE 99 TO TYPDAT OF OLDMAS.
       IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAJ OR ALFAN
           OR ALFAP OR ALFAS OR ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
*    EC070 IS NUMERIC  --  RESTON PICTURE IS 999
 EC070.
      MOVE 00 TO TYPDAT OF OLDMAS.
      PERFORM ENDEDT.
 EC071.
      MOVE 99 TO TYPDAT OF OLDMAS.
      IF (X1 = ALFAC OR ALFAD OR ALFAE OR ALFAH OR ALFAM
          OR ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
*    EC073 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC073.
      MOVE 02 TO TYPDAT OF OLDMAS.
      IF (NEWDAT OF OLDMAS = XBLNK5)
          MOVE ALFAX TO ERRSW
          MOVE NU17 TO ERRCODE OF OLDMAS
      ELSE
          MOVE NEWDAT OF OLDMAS TO ALFNUM
          PERFORM ALFTONUM
          MOVE NUMER TO NUM52
          MOVE NUM52 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
*    EC074 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC074.
      MOVE 02 TO TYPDAT OF OLDMAS.
      MOVE NEWDAT OF OLDMAS TO ALFNUM.
      PERFORM ALFTONUM.
      MOVE NUMER TO NUM52.
      MOVE NUM52 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
```

```
*   EC075 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC075.
      MOVE 02 TO TYPDAT OF OLDMAS.
      MOVE NEWDAT OF OLDMAS TO ALFNUM.
      PERFORM ALFTONUM.
      MOVE NUMER TO NUM52.
      MOVE NUM52 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
*   EC077 IS NUMERIC  --  RESTON PICTURE IS S9999V99
 EC077.
      MOVE 02 TO TYPDAT OF OLDMAS.
      IF (NEWDAT OF OLDMAS = XBLNK5)
          MOVE ALFAX TO ERRSW
          MOVE NU17 TO ERRCODE OF OLDMAS
      ELSE
          MOVE NU0 TO ERRCODE OF OLDMAS
          MOVE NEWDAT OF OLDMAS TO ALFNUM
          PERFORM ALFTONUM
          MOVE NUMER TO SNUM42
          IF (NEGSW > NU0)
              COMPUTE SNUM42 = (-1 * SNUM42).
      IF ERRSW = ALFAM
          MOVE SNUM42 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
*   EC078 IS NUMERIC  --  RESTON PICTURE IS 999V99
 EC078.
      MOVE 02 TO TYPDAT OF OLDMAS.
      MOVE NEWDAT OF OLDMAS TO ALFNUM
      PERFORM ALFTONUM
      MOVE NUMER TO NUM32
      MOVE NUM32 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
*   EC079 IS NUMERIC  --  RESTON PICTURE IS 999V99
 EC079.
      MOVE 02 TO TYPDAT OF OLDMAS.
      IF (NEWDAT OF OLDMAS = XBLNK5)
          MOVE ALFAX TO ERRSW
          MOVE NU17 TO ERRCODE OF OLDMAS
      ELSE
          MOVE NEWDAT OF OLDMAS TO ALFNUM
          PERFORM ALFTONUM
          MOVE NUMER TO NUM32
          MOVE NUM32 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
 EC080.
      MOVE 99 TO TYPDAT OF OLDMAS.
      IF (X1 = ALFAB OR ALFAC OR ALFAG OR ALFAI OR ALFAM OR
          ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAU OR ALFAW OR
          ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
```

```
*   EC083 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC083.
      MOVE 02 TO TYPDAT OF OLDMAS.
      MOVE NEWDAT OF OLDMAS TO ALFNUM.
      PERFORM ALFTONUM.
      MOVE NUMER TO NUM52.
      MOVE NUM52 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
*   EC084 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC084.
      MOVE 02 TO TYPDAT OF OLDMAS.
      MOVE NEWDAT OF OLDMAS TO ALFNUM.
      PERFORM ALFTONUM.
      MOVE NUMER TO NUM52.
      MOVE NUM52 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
 EC085.
      MOVE 99 TO TYPDAT OF OLDMAS.
      IF (X1 = ALFAF OR ALFAL OR ALFAM OR ALFAP OR ALFAR
          OR ALFAS OR ALFAT OR ALFAW OR ALFAX OR ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
 EC086.
      MOVE 99 TO TYPDAT OF OLDMAS.
      IF (X1 = ALFAB OR ALFAC OR ALFAG OR ALFAI OR ALFAM OR
          ALFAP OR ALFAR OR ALFAS OR ALFAT OR ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
*   EC087 IS NUMERIC  --  RESTON PICTURE IS 999V99
 EC087.
      MOVE 02 TO TYPDAT OF OLDMAS.
      MOVE NEWDAT OF OLDMAS TO ALFNUM.
      PERFORM ALFTONUM.
      MOVE NUMER TO NUM32.
      MOVE NUM32 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
*   EC088 IS NUMERIC  --  RESTON PICTURE IS 99V999
 EC088.
      MOVE 03 TO TYPDAT OF OLDMAS.
      MOVE NEWDAT OF OLDMAS TO ALFNUM.
      PERFORM ALFTONUM.
      MOVE NUMER TO NUM23.
      MOVE NUM23 TO NEWDAT OF OLDMAS.
      PERFORM ENDEDT.
```

```
*   EC089 IS NUMERIC  --  RESTON PICTURE IS 999V99
 EC089.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM32.
     MOVE NUM32 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC090.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
*   EC091 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC091.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     MOVE NUM52 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
*   EC092 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC092.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     MOVE NUM52 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC093.
     MOVE 99 TO TYPDAT OF OLDMAS.
      IF (NEWDAT OF OLDMAS = XBLNK8)
          MOVE ALFAX TO ERRSW
          MOVE NU14 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC095.
     IF (X8 > XBLNK8)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU11 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC096.
     MOVE 99 TO TYPDAT OF OLDMAS.
      IF (NEWDAT OF OLDMAS = ALVM OR ANDR OR ANDRS OR ARKS
          OR BLDR OR BLSD OR BLSC OR BRCC OR CLAY OR CLSN OR COAL
          OR COBB OR COSD OR COSC OR CLVM OR CGLM OR CGUN OR DIBS
          OR GLCL OR GNSS OR GRNT OR GRGN OR GRVL OR GRCL OR GRCM
          OR GRDS OR GRSC OR GRCK OR HROP OR IGNS OR LGNT OR LMSN
          OR LMDM OR LOAM OR LOSS OR MRBL OR MARL OR MRLS OR MMPC
          OR MUCK OR MUD OR MDSN OR RSDM OR RYLT OR ROCK OR RBBL
          OR SAND OR SDCL OR SDGL OR SDST OR SGVC OR SNDS OR SRAN
          OR SHLE OR SILT OR STCL OR SLSN OR SLTE OR BSLT OR BNTN
         OR SOIL OR SYNT OR TILL OR TUF OR XBLNK5)
         MOVE NU0 TO ERRCODE OF OLDMAS
```

```
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU15 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC097.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC115.
    MOVE 00 TO TYPDAT OF OLDMAS.
    IF (X4 > XBLNK4)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU11 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC116.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC117.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC118.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAF OR
        ALFAI OR ALFAM OR ALFAO OR ALFAQ OR ALFAS OR ALFAW OR
        ALFAZ OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC120.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE
        OR ALFAF OR ALFAG OR ALFAH OR ALFAJ OR ALFAK OR ALFAL
        OR ALFAM OR ALFAZ OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC122.
    MOVE 00 TO TYPDAT OF OLDMAS.
    IF (X4 > XBLNK4)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU11 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC123.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
```

```
EC124.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X4 = "USGS" OR XBLNK4)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU16 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC125.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAF OR
        ALFAI OR ALFAM OR ALFAO OR ALFAQ OR ALFAS OR ALFAW OR
        ALFAZ OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
*   EC126 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC126.
    MOVE 00 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO ALFNUM.
    PERFORM ALFTONUM.
    MOVE NUMER TO NUM52.
    MOVE NUM52 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.
EC128.
    MOVE 00 TO TYPDAT OF OLDMAS.
    IF (X4 > XBLNK4)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU11 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC129.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC130.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC131.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAF OR
        ALFAM OR ALFAO OR ALFAQ OR ALFAS OR ALFAW OR ALFAZ OR
        ALFAI OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC132.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
```

```
EC133.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAC OR ALFAE OR ALFAM OR ALFAU OR ALFAZ OR
        XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC147.
    MOVE 00 TO TYPDAT OF OLDMAS.
    IF (X3 > XBLNK3)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU14 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC148.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
*   EC150 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC150.
    MOVE 02 TO TYPDAT OF OLDMAS.
    IF (NEWDAT OF OLDMAS > XBLNK5)
        MOVE NU0 TO ERRCODE OF OLDMAS
        MOVE NEWDAT OF OLDMAS TO ALFNUM
        PERFORM ALFTONUM
        MOVE NUMER TO NUM52
        MOVE NUM52 TO NEWDAT OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU20 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC151.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO
        OR ALFAS OR ALFAR OR ALFAZ OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC152.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAB OR ALFAC OR ALFAE OR ALFAF OR ALFAM
        OR ALFAO OR ALFAP OR ALFAT OR ALFAU OR ALFAV OR
        ALFAR OR ALFAW OR ALFAZ OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
```

```
*   EC153 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC153.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     PERFORM ENDEDT.
*   EC154 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC154.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52
     MOVE NUM52 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC156.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAC OR ALFAE OR ALFAG OR ALFAH
         OR ALFAL OR ALFAM OR ALFAR OR ALFAS OR ALFAT OR ALFAV
         OR ALFAZ OR XBLNK1)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*   EC157 IS NUMERIC  --  RESTON PICTURE IS 9999V9
 EC157.
     MOVE 01 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM41
     MOVE NUM41 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC159.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC161.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X10 > XBLNK10)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU17 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC162.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC163.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC181.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (NEWDAT OF OLDMAS > XBLNK10)
         MOVE NU0 TO ERRCODE OF OLDMAS
```

```
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU17 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC182.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAC OR ALFAD OR ALFAZ OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC185.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC187.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 > XBLNK10)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU11 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC188.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC190.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 >  XBLNK10)
        MOVE ALFAX TO ERRSW
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU17 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC191.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC193.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X10 = XBLNK10)
        MOVE ALFAX TO ERRSW
        MOVE NU11 TO ERRCODE OF OLDMAS
    ELSE
        MOVE NU0 TO ERRCODE OF OLDMAS.
    IF (X10 > TDYDAT)
        MOVE ALFAX TO ERRSW
        MOVE NU18 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC195.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC196.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (NEWDAT OF OLDMAS = QW1 OR QW2 OR QW3 OR QW4 OR QW5
    OR QW6 OR QW7 OR QW8 OR QW9 OR QW10 OR QW11 OR QW12 OR QW13
```

```
     OR QW14 OR QW15 OR QW16 OR QW17 OR QW18 OR QW19)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU18 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*    EC197 IS NUMERIC  --  RESTON PICTURE IS 999999V9
 EC197.
     MOVE 01 TO TYPDAT OF OLDMAS.
     IF (NEWDAT OF OLDMAS > XBLNK6)
         MOVE NEWDAT OF OLDMAS TO ALFNUM
         PERFORM ALFTONUM
         MOVE NUMER TO NUM61
         MOVE NUM61 TO NEWDAT OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU17 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
 EC199.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAB OR ALFAC OR ALFAD OR ALFAE
         OR ALFAF OR ALFAG OR ALFAH OR ALFAI OR ALFAJ OR ALFAK
         OR ALFAL OR ALFAM OR ALFAN OR ALFAO OR ALFAP OR ALFAQ
         OR ALFAS OR ALFAT OR ALFAU OR ALFAV OR ALFAZ)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*    EC200 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC200.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     MOVE NUM52 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
*    EC201 IS NUMERIC  --  RESTON PICTURE IS 99999V99
 EC201.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     MOVE NUM52 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
 EC202.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAD OR ALFAG OR ALFAL OR ALFAO
         OR ALFAS OR ALFAR OR ALFAZ OR XBLNK1)
         MOVE NU0 TO ERRCODE OF OLDMAS
      ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
      PERFORM ENDEDT.
```

```
EC235.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
*    EC237 IS NUMERIC  --  RESTON PICTURE IS 9999V99
EC237.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO SNUM42.
     MOVE SNUM42 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
EC238.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = XBLNK1 OR ALFAD OR ALFAE OR ALFAF OR ALFAG
         OR ALFAH OR ALFAO OR ALFAP OR ALFAR OR ALFAS OR ALFAT
         OR ALFAV OR ALFAX OR ALFAZ)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC239.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = XBLNK1 OR ALFAA OR ALFAC OR ALFAE OR ALFAG OR
         ALFAH OR ALFAL OR ALFAM OR ALFAR OR ALFAS OR ALFAT OR
         ALFAV OR ALFAZ)
         MOVE NU0.TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC254.
     MOVE 00 TO TYPDAT OF OLDMAS.
     IF (X3 > XBLNK3)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU14 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC255.
     MOVE 00 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC256.
     MOVE 00 TO TYPDAT OF OLDMAS.
     IF (X3 > XBLNK3)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU14 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
```

```
EC257.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAY OR XBLNK1)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
EC258.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC259.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC261.
     MOVE 99 TO TYPDAT OF OLDMAS.
      IF (X1 = ALFAF OR ALFAM OR ALFAP OR ALFAZ OR XBLNK1)
          MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
          MOVE ALFAX TO ERRSW
          MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
*   EC266 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC266.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO SNUM32.
     MOVE ZEROS TO NUMER.
     IF (NEGSW > NU0)
         COMPUTE SNUM32 = (-1 *  SNUM32)
         MOVE SNUM32 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
EC267.
     MOVE 99 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
EC268.
     MOVE 00 TO TYPDAT OF OLDMAS.
     PERFORM ENDEDT.
*   EC272 IS NUMERIC  --  RESTON PICTURE IS 99999V99
EC272.
     MOVE 02 TO TYPDAT OF OLDMAS.
     MOVE NEWDAT OF OLDMAS TO ALFNUM.
     PERFORM ALFTONUM.
     MOVE NUMER TO NUM52.
     PERFORM ENDEDT.
EC301.
     MOVE 99 TO TYPDAT OF OLDMAS.
     IF (X1 = ALFAA OR ALFAC OR ALFAD OR ALFAE OR ALFAG
         OR ALFAH OR ALFAM OR ALFAO OR ALFAP OR ALFAR OR ALFAS
         OR ALFAT OR ALFAU OR ALFAW OR XBLNK1)
         MOVE NU0 TO ERRCODE OF OLDMAS
     ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
     PERFORM ENDEDT.
```

```
EC302.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAA OR ALFAC OR ALFAD OR ALFAE OR ALFAG
        OR ALFAH OR ALFAM OR ALFAO OR ALFAP OR ALFAR OR ALFAS
        OR ALFAT OR ALFAU OR ALFAW OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC304.
    MOVE 99 TO TYPDAT OF OLDMAS.
    IF (X1 = ALFAN OR ALFAP OR ALFAS OR XBLNK1)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
         MOVE ALFAX TO ERRSW
         MOVE NU5 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC307.
    MOVE 99 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC309.
    MOVE 02 TO TYPDAT OF OLDMAS.
    MOVE NEWDAT OF OLDMAS TO ALFNUM
    PERFORM ALFTONUM
    MOVE NUMER TO NUM42
    MOVE NUM42 TO NEWDAT OF OLDMAS.
    PERFORM ENDEDT.
EC311.
    MOVE 00 TO TYPDAT OF OLDMAS.
    IF (X3 > XBLNK3)
        MOVE NU0 TO ERRCODE OF OLDMAS
    ELSE
        MOVE ALFAX TO ERRSW
        MOVE NU14 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC312.
    MOVE 00 TO TYPDAT OF OLDMAS.
    IF (X3 > XBLNK3)
       MOVE NU0 TO ERRCODE OF OLDMAS
   ELSE
       MOVE ALFAX TO ERRSW
       MOVE NU14 TO ERRCODE OF OLDMAS.
    PERFORM ENDEDT.
EC321.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC322.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
EC323.
    MOVE 02 TO TYPDAT OF OLDMAS.
    IF (NEWDAT OF OLDMAS = XBLNK5)
        MOVE ALFAX TO ERRSW
        MOVE NU17 TO ERRCODE OF OLDMAS
```

```
    ELSE
        MOVE NEWDAT OF OLDMAS TO ALFNUM
        PERFORM ALFTONUM
        MOVE NUMER TO SNUM42
        IF (NEGSW > NU0)
            COMPUTE SNUM42 = (SNUM42 * -1).
     MOVE SNUM42 TO NEWDAT OF OLDMAS.
     PERFORM ENDEDT.
EC324.
    MOVE 00 TO TYPDAT OF OLDMAS.
    PERFORM ENDEDT.
ENDEDT.
    MOVE ERRSW TO CCSR OF OLDMAS.
    MOVE OLDMAS TO RECTAB (MM).
CLCSIT.
    IF NOT (LC1 = "A" OR "B" OR "C" OR "D"),
       MOVE ALFAC TO ERRSW.
    IF NOT (LC2 = "H" OR "-"), MOVE ALFAC TO ERRSW.
    IF ((LC1 = "A" OR "B") AND (LC3 > 42 OR < 1)),
        MOVE ALFAC TO ERRSW.
    IF ((LC1 = "C") AND (LC3 > 19 OR < 1)),
        MOVE ALFAC TO ERRSW.
    IF ((LC1 = "D") AND (LC3 > 24 OR < 1)),
        MOVE ALFAC TO ERRSW.
    IF NOT (LC4 = "H" OR "-"), MOVE ALFAC TO ERRSW.
    IF ((LC1 = "A" OR "D") AND (LC5 > 32 OR < 1)),
        MOVE ALFAC TO ERRSW.
    IF ((LC1 = "B") AND (LC5 > 23 OR < 1)),
        MOVE ALFAC TO ERRSW.
    IF ((LC1 = "C") AND (LC5 > 25 OR < 1)),
        MOVE ALFAC TO ERRSW.
    IF NOT (LC6 = " " OR "N" OR "E" OR "W" OR "S"),
        MOVE ALFAC TO ERRSW.
    IF LC7 > 36 OR < 1, MOVE ALFAC TO ERRSW.
    IF NOT (LC8 = ALFAU OR ALFAA OR ALFAB OR ALFAC OR ALFAD),
       MOVE ALFAC TO ERRSW.
    IF NOT (LC9 = ALFAU OR ALFAA OR ALFAB OR ALFAC OR ALFAD),
       MOVE ALFAC TO ERRSW.
    IF NOT (LC10 = ALFAU OR ALFAA OR ALFAB OR ALFAC OR ALFAD),
       MOVE ALFAC TO ERRSW.
    IF NOT (LC11 = " " OR "1" OR "2" OR "3" OR "4" OR
     "5" OR "6" OR "7" OR "8" OR "9"), MOVE ALFAC TO ERRSW.
    IF ERRSW = ALFAC
       MOVE ALFAX TO ERRSW
       MOVE NU9 TO ERRCODE OF OLDMAS.
ALFTONUM.
     PERFORM NEGTST.
     PERFORM DGCNTR VARYING ALN FROM 1 BY 1 UNTIL
        (ADIGIT (ALN) = XBLNK1).
     COMPUTE NUMS = ( 11 - DGCNT ) + 1.
     SET ALN TO DLEN OF OLDMAS.
     PERFORM DGMOVE VARYING NLN FROM 11 BY -1 UNTIL  NLN < NUMS.
     EXAMINE NNUM REPLACING ALL " " BY 0.
     MOVE NNUM TO NUMER.
     EXAMINE NUMER REPLACING ALL " " BY 0.
```

```
     COMPUTE DECNT = DECNT - 1.
     COMPUTE NUMER = NUMER DIVIDED BY ( 10 ** DECNT).
     COMPUTE NUMER = NUMER + .0000001.
NEGTST.
     IF ( ADIGIT (1) = "-" ) ADD NU1 TO NEGSW,
          MOVE NU0 TO ADIGIT (1).
ALLMOVE.
     MOVE ADIGIT (ALN + 1) TO ADIGIT (ALN).
DGCNTR.
     IF (ADIGIT (ALN) = ".") ADD NU1 TO DECSW.
     IF NOT ( DECSW > NU0)
          ADD NU1 TO DGCNT
     ELSE ADD NU1 TO DGCNT, DECNT.
     IF (DECSW > NU0) MOVE ADIGIT (ALN + 1) TO ADIGIT (ALN).
DGMOVE.
     SET ALN DOWN BY 1.
     MOVE ADIGIT (ALN) TO NDIGIT (NLN).
CLOSEA.
    MOVE 1 TO EOFSW.
    MOVE ALL ZEROS TO DELTSS.
CLSEAC.
    IF (KEYIDD = 0000000000000 ),
         MOVE ZEROS TO DELTSS.
    MOVE ZEROS  TO KEYIDC.
CLSEAD.
    IF (KEYIDC = 0000000000000),
         MOVE ZEROS TO DELTSS.
    MOVE ZEROS TO KEYIDD.
FINALA.
    CLOSE CHANGE-FILE, FILEA, FILEB.
    GO TO STOP3.
STARTB.
    OPEN INPUT CHANGE-FILE.
REEDB.
    READ CHANGE-FILE AT END GO TO CLOSEB.
    ADD NU1 TO TFILEB.
    MOVE CNGREC TO LLPRT.
    WRITE LIST-LINE.
    GO TO REEDB.
CLOSEB.
    GO TO STOP3.
STARTC.
    DISPLAY "STARTC ".
    STOP RUN.
HDPRNT.
    ADD 1 TO PGCNT.
    MOVE PGCNT TO PNUM OF HDR1.
    MOVE 4 TO LNCNT.
    WRITE LIST-LINE FROM HDR1.
    WRITE LIST-LINE FROM HDR2.
    WRITE LIST-LINE FROM HDR3.
```

```
WRITE-LIST.
     IF LNCNT > 60,
         PERFORM HDPRNT,
     ELSE
         ADD 1 TO LNCNT, NDETLN.
     ADD NU1 TO YY.
     MOVE RECTAB (YY) TO OLDMAS.
     MOVE RECTAB (YY) TO PRNTLN.
     IF (ERRSW = ALFAX)
          ADD 1 TO ERNUM,
          PERFORM ERR-MSG
          PERFORM WRTERR
     ELSE
          ADD 1 TO UPNUM
          PERFORM WRTUPD.
     WRITE LIST-LINE FROM PRNTLN.
     MOVE SPACES TO CC OF PRNTLN.
     IF (YY = NMM OR YY > NMM) PERFORM CLEAR-PROC.
WRTERR.
     MOVE RECTAB (YY) TO AUDREC.
     MOVE ERNUM TO NSEQ OF AUDREC.
     WRITE AUDREC.
WRTUPD.
     MOVE RECTAB (YY) TO RECOUT.
     WRITE RECOUT.
ERR-MSG.
     IF ERRCODE OF RECTAB (YY) = NU0 OR " "
         MOVE SPACES TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU1
         MOVE ER-MSG (NU1) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU2
         MOVE ER-MSG (NU2) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU3
         MOVE ER-MSG (NU3) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU4
         MOVE ER-MSG (NU4) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU5
         MOVE ER-MSG (NU5) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU6
         MOVE ER-MSG (NU6) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU7
         MOVE ER-MSG (NU7) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU8
         MOVE ER-MSG (NU8) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU9
         MOVE ER-MSG (NU9) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU10
         MOVE ER-MSG (NU10) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU11
         MOVE ER-MSG (NU11) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU12
         MOVE ER-MSG (NU12) TO ERPRINT OF PRNTLN.
     IF ERRCODE OF RECTAB (YY) = NU13
         MOVE ER-MSG (NU13) TO ERPRINT OF PRNTLN.
```

```
    IF ERRCODE OF RECTAB (YY) = NU14
        MOVE ER-MSG (NU14) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU15
        MOVE ER-MSG (NU15) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU16
        MOVE ER-MSG (NU16) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU17
        MOVE ER-MSG (NU17) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU18
        MOVE ER-MSG (NU18) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU19
        MOVE ER-MSG (NU19) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU20
        MOVE ER-MSG (NU20) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU21
        MOVE ER-MSG (NU21) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU22
        MOVE ER-MSG (NU22) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU23
        MOVE ER-MSG (NU23) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU24
        MOVE ER-MSG (NU24) TO ERPRINT OF PRNTLN.
    IF ERRCODE OF RECTAB (YY) = NU25
        MOVE ER-MSG (NU25) TO ERPRINT OF PRNTLN.
STOP1.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "CONTROL INPUT FILE EMPTY".
    STOP RUN.
STOP2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.
STOP3.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY TFILEB.
    MOVE INCRDS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FADATA.
    MOVE NDETLN TO DRLNS.
    MOVE NFLEOA TO FOUT.
    WRITE LIST-LINE FROM JBCNLX.
    DISPLAY "NORMAL JOB TERMINATION".
    CLOSE CARD-FILE, LIST-FILE.
    STOP RUN.
STOP4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "SOURCE ID RECORDS MISSING".
    STOP RUN.
STOP5.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP5 FILEA-UPCHNG EMPTY",
      "  *** CHECK FOR ATTACH OF UPCHNG IN JCL".
    STOP RUN.
```

```
STOP6.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "SECOND CARD OF PAIR MISSING".
    STOP RUN.
STOP7.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP7 ", ESMSG1.
    DISPLAY "INPUT STREAM CARD RECORDS".
    DISPLAY ESMSG2.
    DISPLAY "CORRECT SEQUENCE AND RERUN".
    STOP RUN.
STOP8.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ERROR IN RECORD SEQUENCE", "STOP8  AZERRS".
    DISPLAY ERRCNG.
    DISPLAY "KTEST = ", KTEST, "KEYIDD = ", KEYIDD.
    STOP RUN.
STOP9.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP9 ", ESMSG1.
    DISPLAY "INPUT AZSR-NUMSER FILE ", ESMSG2.
    DISPLAY "KEYID OF CNGREC = ", KEYID OF CNGREC.
    DISPLAY "KEYIDL          = ", KEYIDL.
    STOP RUN.
STOP10.
    STOP RUN.
ERROR8.
    DISPLAY "****************************************".
    DISPLAY "ERROR IN RECORD SEQUENCE", "ERROR8  AZERRS".
    DISPLAY ERRCNG.
    DISPLAY "KTEST = ", KTEST, "KEYIDD = ", KEYIDD.
    DISPLAY "****************************************".
    ADD NUS TO LNCNT.
ERR011.
    DISPLAY OLDMAS.
    DISPLAY "STOP 11", "RNUM OF CNGREC NOT FOUND".
    DISPLAY "R=*** NOT VALID ", RNUM OF CNGREC.
    DISPLAY "******* ", CNGREC.
```

```
IDENTIFICATION DIVISION.
PROGRAM-ID. UPDATE-1.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEA ASSIGN TO MASIN.
    SELECT FILEB ASSIGN TO MASOUT.
    SELECT FILEC ASSIGN TO CMAST.
    SELECT CHNGEL ASSIGN TO DCHNG.
DATA DIVISION.
FILE SECTION.
FD  CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
01  CNL-CARD.
    02  CNLXXX.
        03 ID-CNL PIC IS XXX.
        03 ID-NUM PIC 999.
    02  DELNUM REDEFINES CNLXXX PIC 9(6).
    02  VSNA PIC X.
    02  ID-DES PIC X(73).
FD  LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
01  LIST-LINE.
    03  CC PIC X.
    03  LLPRT PIC X(132).
FD  FILEA
    DATA RECORD IS OLDMAS RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
    CHARACTERS LABEL RECORDS ARE STANDARD.
01  OLDMAS.
    03  KEYIDM.
        05  KYLATM.
            07  KLD PIC 99.
            07  KLM PIC 99.
            07  KLS PIC 99.
        05  KYLNGM.
            07  KLGD PIC 999.
            07  KLGM PIC 99.
            07  KLGS PIC 99.
        05  KYDUPM PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
           OF OLDMAS.
```

```
FD  FILEB
    DATA RECORD IS NEWMAS RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
    CHARACTERS LABEL RECORDS ARE STANDARD.
01  NEWMAS.
    03  KEYIDM.
        05  KYLATM.
            07  KLD PIC 99.
            07  KLM PIC 99.
            07  KLS PIC 99.
        05  KYLNGM.
            07  KLGD PIC 999.
            07  KLGM PIC 99.
            07  KLGS PIC 99.
        05  KYDUPM PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
           OF OLDMAS.
FD  FILEC
    DATA RECORD IS CNGMAS RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
    CHARACTERS LABEL RECORDS ARE OMITTED.
01  CNGMAS.
    03  KEYIDM.
        05  KYLATM.
            07  KLD PIC 99.
            07  KLM PIC 99.
            07  KLS PIC 99.
        05  KYLNGM.
            07  KLGD PIC 999.
            07  KLGM PIC 99.
            07  KLGS PIC 99.
        05  KYDUPM PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
           OF OLDMAS.
FD  CHNGEL
    DATA RECORD IS CNGREC
    LABEL RECORDS ARE OMITTED.
01  CNGREC.
    03  VSN PIC 99999.
    03  VSNA PIC X.
    03  PRD2 PIC 9(5).
    03  KSEQ PIC 9(4).
    03  KEYID.
        05  KEYLAT.
            07  LATDEG PIC 99.
            07  LATMIN PIC 99.
            07  LATSEC PIC 99.
```

```
        05  KEYLNG.
            07  LNGDEG PIC 999.
            07  LNGMIN PIC 99.
            07  LNGSEC PIC 99.
        05  KEYDUP PIC 99.
    03  CSCDE PIC X.
    03  RNUM PIC 999.
    03  CNUM PIC 999.
    03  CCR PIC X.
    03  CCSR PIC X.
    03  TYPDAT PIC 99.
    03  DLEN PIC 99.
    03  NEWDAT PIC X(45).
    03  NSEQ PIC 9(8).
WORKING-STORAGE SECTION.
77  ACOUNT PIC 9(6) VALUE 0.
77  ALA PIC X VALUE "A".
77  ALD PIC X VALUE "D".
77  ALFLAT PIC X(6).
77  ALFLNG PIC X(7).
77  ALPHA6 PIC X(6).
77  ALPHA7 PIC X(7).
77  ALR PIC X VALUE "R".
77  ALT PIC X VALUE "T".
77  ALX PIC X VALUE "X".
77  ALY PIC X VALUE "Y".
77  ALZ PIC X VALUE "Z".
77  BT PIC 999 VALUE 0.
77  BTA PIC 9(4).
77  BLNK1 PIC X VALUE SPACE.
77  CMPLAT PIC 9(6).
77  CMPLNG PIC 9(7).
77  CNRECN  PIC 9(6) VALUE ZEROES.
77  DELSW PIC X VALUE "Z".
77  DEM1 PIC X(15) VALUE "  DATA ENTRY    ".
77  DEM2 PIC X(10) VALUE "*UNCHECKED".
77  DYZ PIC 9.
77  ERMSG1 PIC X(27) VALUE "  AZ FORMAT SORTED CHANGES ".
77  ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
77  ERRCNT PIC 9(9) VALUE 0.
77  ERRSW PIC X VALUE "Y".
77  ESW PIC 99 VALUE 0.
77  FLECNT PIC 9(6).
77  FRSW PIC X VALUE "F".
77  HSW PIC X VALUE "X".
77  INCRDS PIC 9(6) VALUE 0.
77  JULD PIC 9(5).
77  LNCNT PIC 99 VALUE 70.
77  MSG1 PIC X(25) VALUE "CHECKS OK DROP NOT NEEDED".
77  MSG2 PIC X(25) VALUE "KEYID NOT = LAT-LNG FLDS".
77  NDATIN PIC 9(6) VALUE 0.
77  NDETLN PIC 9(6) VALUE 0.
77  NFLEIA PIC 9(6) VALUE 0.
77  NFLEOA PIC 9(6) VALUE 0.
77  NU1 PIC 9 VALUE 1.
```

```
77  PGCNT PIC 999 VALUE 0.
77  SEQCHK PIC 9(6) VALUE 0.
77  TFILEA PIC 9(6) VALUE 0.
77  TFILEB PIC 9(6) VALUE 0.
77  TFILEC PIC 9(6) VALUE 0.
01 WKDATE.
    03  YY PIC 99.
    03  MM PIC 99.
    03  DY PIC 99.
01 NAME-MONTH.
    03  JAN PIC XXXX VALUE " JAN".
    03  FEB PIC XXXX VALUE " FEB".
    03  MAR PIC XXXX VALUE " MAR".
    03  APR PIC XXXX VALUE " APR".
    03  MAY PIC XXXX VALUE " MAY".
    03  JUNE PIC XXXX VALUE "JUNE".
    03  JULY PIC XXXX VALUE "JULY".
    03  AUG PIC XXXX VALUE " AUG".
    03  SEPT PIC XXXX VALUE "SEPT".
    03  OCT PIC XXXX VALUE " OCT".
    03  NOV PIC XXXX VALUE " NOV".
    03  DEC PIC XXXX VALUE " DEC".
01 MONTH-TABLE REDEFINES NAME-MONTH.
    03  MONTHY PIC XXXX OCCURS 12 TIMES INDEXED BY NMM.
01  JOBCNL.
    03  CC PIC X VALUE "1".
    03  PCNL.
        05  CNTYPE PIC XXX.
        05  GU-CNL PIC 999.
        05  PCNLX PIC X(74).
    03  FILLER PIC X(52) VALUE SPACES.
01  JBCNLX.
    03  CC PIC X VALUE "0".
    03  PSTATS PIC X(14) VALUE "RECORD COUNTS".
    03  FILLER PIC X(6) VALUE SPACES.
    03  XCARD PIC X(8) VALUE "CARDS IN".
    03  CSTAT PIC ZZZZZ9.
    03  XDRLNS PIC X(20) VALUE " SITES TO BE CHANGED".
    03  DRLNS PIC ZZZZZ9.
    03  XDATIZN PIC X(15) VALUE " CHANGE RECORDS".
    03  DATAIN PIC ZZZZZ9.
    03  XFAIN PIC X(19) VALUE " UNCHANGED MASTERS ".
    03  FADATA PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XNOUT PIC X(12) VALUE " NEW MASTER ".
    03  FOUT PIC ZZZZZ9.
01  HDR1.
    03  CC1 PIC 9 VALUE 1.
    03  FILLER PIC X.
    03  LDATE PIC X(6) VALUE "DATE: ".
    03  RMTH PIC X(4).
    03  FILLER PIC X.
    03  RDY PIC Z9.
    03  XB PIC XXXX VALUE ", 19".
    03  RYR PIC XX.
```

```
    03  FILLER PIC X(21).
    03  TITLEA PIC X(60).
    03  FILLER PIC X(21).
    03  PGE PIC X(4) VALUE "PAGE".
    03  PNUM PIC ZZZ9.
    03  FILLER PIC X(2).
01  HDR2.
    03  CC PIC 9 VALUE 0.
    03  USGS PIC X(7) VALUE " USGS: ".
    03  GSNAME PIC X(20).
    03  FILLER PIC XXX.
    03  BASINL PIC X(7) VALUE "SOURCES".
    03  FILLER PIC X.
    03  SRCEID PIC X(50).
    03  SRCNAM PIC X(30).
    03   CNLZZZ PIC X(6).
    03  FILLER PIC X(8).
01  DATAA.
    03  ID-CNL PIC IS XXX.
    03  ID-NUM PIC IS 999.
    03  TITLEX.
        05  SRCAGC.
            07  GNAME PIC X(20).
            07  CNAME PIC X(30).
        05  FILLER PIC X(10).
    03  FILLER PIC X(14).
01  KEYIDL.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 99.
01  KEYIDX.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 99.
```

```
PROCEDURE DIVISION.
NOTE-A.
     NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION
     ROUTINE THAT OPENS THE INPUT AND LISTING FILES
     AND THEN THE DATE STORED IN THE SYSTEM IS READ
     AND USED TO SET UP THE DATE IN THE FIRST
     HEADING LINE.  NO RETURN FROM THE MAINLINE
     ROUTINE IS MADE TO THIS PROCEDURE.
BEGIN.
    OPEN INPUT CARD-FILE.
    OPEN OUTPUT LIST-FILE.
    ACCEPT WKDATE FROM DATE.
    ACCEPT JULD FROM DAY.
    SET NMM TO MM.
    MOVE MONTHY (NMM) TO RMTH OF HDR1.
    MOVE DY TO RDY OF HDR1.
    MOVE YY TO RYR OF HDR1.
NOTE-B.
     NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES
     THE GENERAL DESCRIPTION OF THE REPORT TO THE
     HEADING AREA.  THEN A BRANCH IS TAKEN TO THE
     APPROPRIATE STARTING ROUTINE DEPENDING ON THE
     NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
     THERE IS NO RETURN TO THIS ROUTINE.
READC.
    READ CARD-FILE AT END GO TO STOP1.
    IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
    IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
    ADD 1 TO INCRDS.
    MOVE CNL-CARD TO PCNL.
    MOVE CNLXXX TO CNLZZZ.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
        TO TITLEA OF HDR1 ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 011 MOVE GNAME TO GSNAME
        MOVE CNAME TO SRCNAM ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    READ CARD-FILE INTO DATAA AT END GO TO STOP4.
    IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
    IF ID-NUM OF DATAA = 012 MOVE SRCAGC TO SRCEID,
        MOVE GO-CNL TO CNLZZZ ELSE GO TO STOP4.
    ADD 1 TO INCRDS.
    IF GO-CNL = 001, GO TO STARTA.
    IF GO-CNL = 002, GO TO STARTB.
    IF GO-CNL = 003
         GO TO STARTC
     ELSE
         GO TO STOP2.
```

```
NOTE-C.
    NOTE NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
    THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
    ROUTINES.
STARTA.
    OPEN INPUT CHNGEL
               FILEA.
    OPEN OUTPUT FILEB
                FILEC.
    READ FILEA AT END DISPLAY
    "****700D1 NO MASTER FILE INPUT DATA", STOP RUN.
    MOVE NU1 TO NFLEIA.
    MOVE 0 TO NDETLN, NDATIN, NFLEOA.
    MOVE KEYIDM OF OLDMAS TO KEYIDX.
STARA1.
    READ CHNGEL AT END DISPLAY  "****701D1 END OF THE EDIT RUN"
        MOVE ALA TO HSW
        GO TO CMP.
    ADD NU1 TO NDATIN.
    IF KEYID OF CNGREC = KEYIDL
         GO TO STARA1.
    IF KEYID OF CNGREC > KEYIDL
         DISPLAY "****702D1 CHANGE FILE OUT OF SEQUENCE"
         STOP RUN
     ELSE
         MOVE KEYID OF CNGREC TO KEYIDL.
CMP.
*   IF NDETLN > 5 GO TO STOP3.
    IF (HSW = ALA OR KEYIDM OF OLDMAS > KEYIDL)
        WRITE NEWMAS FROM OLDMAS
        ADD NU1 TO NFLEOA
        GO TO RDMAS.
*   IF KEYIDM OF OLDMAS = KEYIDL GO TO TEST-RNUM.
    IF KEYIDM OF OLDMAS = KEYIDL
        ADD NU1 TO NDETLN
        WRITE CNGMAS FROM OLDMAS
        GO TO RDMAS.
    IF (KEYIDM OF OLDMAS < KEYIDL AND HSW NOT = ALA)
        GO TO STARA1.
TEST-RNUM.
          IF RNUM OF OLDMAS = RNUM OF CNGREC
              ADD NU1 TO NDETLN
              WRITE CNGMAS FROM OLDMAS
              GO TO RDMAS
          ELSE
             WRITE NEWMAS FROM OLDMAS
             GO TO RDMAS.
RDMAS.
    READ FILEA AT END CLOSE CHNGEL FILEA FILEB FILEC
        DISPLAY "****703D1 NORMAL END OF RUN"
        GO TO STOP3.
    IF KEYIDM OF OLDMAS > KEYIDX
         DISPLAY "****704D1 MASTER FILE OUT OF SEQUENCE"
         STOP RUN
```

```
      ELSE
          MOVE KEYIDM OF OLDMAS TO KEYIDX.
      ADD NU1 TO NFLEIA.
      GO TO CMP.
STARTB.
     STOP RUN.
STARTC.
     STOP RUN.
STOP1.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "CONTROL INPUT FILE EMPTY".
     STOP RUN.
STOP2.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
     STOP RUN.
STOP3.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY TFILEB.
     MOVE INCRDS TO CSTAT.
     MOVE NDATIN TO DATAIN.
     MOVE NFLEIA TO FADATA.
     MOVE NDETLN TO DRLNS.
     MOVE NFLEOA TO FOUT.
     WRITE LIST-LINE FROM JBCNLX.
     DISPLAY "NORMAL JOB TERMINATION".
     DISPLAY "ERRORS = ", ACOUNT.
     CLOSE CARD-FILE, LIST-FILE.
     STOP RUN.
STOP4.
     WRITE LIST-LINE FROM JOBCNL.
     DISPLAY "STOP4".
     STOP RUN.
```

```
IDENTIFICATION DIVISION.
PROGRAM-ID. UPDATE-2.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. 6600.
OBJECT-COMPUTER. 6600.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEA ASSIGN TO MASIN.
    SELECT FILEB ASSIGN TO MASOUT.
    SELECT FILEC ASSIGN TO CMAST.
    SELECT CHNGEL ASSIGN TO DCHNG.
DATA DIVISION.
FILE SECTION.
FD  CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
01  CNL-CARD.
    02  CNLXXX.
        03 ID-CNL PIC IS XXX.
        03 ID-NUM PIC 999.
    02  DELNUM REDEFINES CNLXXX PIC 9(6).
    02  VSNA PIC X.
    02  ID-DES PIC X(73).
FD  LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
01  LIST-LINE.
    03  CC PIC X.
    03  LLPRT PIC X(132).
FD  FILEA
    DATA RECORD IS OLDMAS RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
    CHARACTERS LABEL RECORDS ARE OMITTED.
01  OLDMAS.
    03  KEYIDM.
        05  KYLATM.
            07  KLD PIC 99.
            07  KLM PIC 99.
            07  KLS PIC 99.
        05  KYLNGM.
            07  KLGD PIC 999.
            07  KLGM PIC 99.
            07  KLGS PIC 99.
        05  KYDUPM PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
           OF OLDMAS.
```

```
FD  FILEB
    DATA RECORD IS NEWMAS RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
    CHARACTERS LABEL RECORDS ARE OMITTED.
01  NEWMAS.
    03  KEYIDM.
        05  KYLATM.
            07  KLD PIC 99.
            07  KLM PIC 99.
            07  KLS PIC 99.
        05  KYLNGM.
            07  KLGD PIC 999.
            07  KLGM PIC 99.
            07  KLGS PIC 99.
        05  KYDUPM PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
           OF NEWMAS.
FD  FILEC
    DATA RECORD IS CNGMAS RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY BLOCK CONTAINS 4793 TO 5120
    CHARACTERS LABEL RECORDS ARE OMITTED.
01  CNGMAS.
    03  KEYIDM.
        05  KYLATM.
            07  KLD PIC 99.
            07  KLM PIC 99.
            07  KLS PIC 99.
        05  KYLNGM.
            07  KLGD PIC 999.
            07  KLGM PIC 99.
            07  KLGS PIC 99.
        05  KYDUPM PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES DEPENDING ON LNUM
           OF OLDMAS.
FD  CHNGEL
    DATA RECORD IS CNGREC
    LABEL RECORDS ARE OMITTED.
01  CNGREC.
    03  VSN PIC 99999.
    03  VSNA PIC X.
    03  PRD2 PIC 9(5).
    03  KSEQ PIC 9(4).
    03  KEYID.
        05  KEYLAT.
            07  LATDEG PIC 99.
            07  LATMIN PIC 99.
            07  LATSEC PIC 99.
```

```
           05  KEYLNG.
               07  LNGDEG PIC 999.
               07  LNGMIN PIC 99.
               07  LNGSEC PIC 99.
           05  KEYDUP PIC 99.
       03  CSCDE PIC X.
       03  RNUM PIC 999.
       03  CNUM PIC 999.
       03  CCR PIC X.
       03  CCSR PIC X.
       03  TYPDAT PIC 99.
       03  DLEN PIC 99.
       03  NEWDAT PIC X(45).
       03  NSEQ PIC 9(8).
WORKING-STORAGE SECTION.
77  ACOUNT PIC 9(6) VALUE 0.
77  ALA PIC X VALUE "A".
77  ALD PIC X VALUE "D".
77  ALFLAT PIC X(6).
77  ALFLNG PIC X(7).
77  ALM PIC X VALUE "M".
77  ALPHA6 PIC X(6).
77  ALPHA7 PIC X(7).
77  ALR PIC X VALUE "R".
77  ALT PIC X VALUE "T".
77  ALX PIC X VALUE "X".
77  ALY PIC X VALUE "Y".
77  ALZ PIC X VALUE "Z".
77  BT PIC 999 VALUE 0.
77  BTA PIC 9(4).
77  BLNK1 PIC X VALUE SPACE.
77  CMPLAT PIC 9(6).
77  CMPLNG PIC 9(7).
77  CNRECN  PIC 9(6) VALUE ZEROES.
77  CSW PIC X VALUE "X".
77  DELSW PIC X VALUE "Z".
77  DEM1 PIC X(15) VALUE "  DATA ENTRY    ".
77  DEM2 PIC X(10) VALUE "*UNCHECKED".
77  DSW PIC 9 VALUE 0.
77  ERMSG1 PIC X(27) VALUE "  AZ FORMAT SORTED CHANGES ".
77  ERMSG2 PIC X(27) VALUE " GW & QW LOCAL IDS UNEQUAL ".
77  ERRCNT PIC 9(9) VALUE 0.
77  ERRSW PIC X VALUE "Y".
77  ESW PIC 99 VALUE 0.
77  FLECNT PIC 9(6).
77  FRSW PIC X VALUE "F".
77  HSW PIC X VALUE "X".
77  INCRDS PIC 9(6) VALUE 0.
77  JULD PIC 9(5).
77  LNCNT PIC 99 VALUE 70.
77  MSG1 PIC X(25) VALUE "CHECKS OK DROP NOT NEEDED".
77  MSG2 PIC X(25) VALUE "KEYID NOT = LAT-LNG FLDS".
77  MSW PIC X VALUE "X".
77  NDATIN PIC 9(6) VALUE 0.
```

```
77  NDETLN PIC 9(6) VALUE 0.
77  NFLEIA PIC 9(6) VALUE 0.
77  NFLEOA PIC 9(6) VALUE 0.
77  NU1 PIC 9 VALUE 1.
77  PGCNT PIC 999 VALUE 0.
77  SEQCHK PIC 9(6) VALUE 0.
77  TFILEA PIC 9(6) VALUE 0.
77  TFILEB PIC 9(6) VALUE 0.
77  TFILEC PIC 9(6) VALUE 0.
01 WKDATE.
    03  YY PIC 99.
    03  MM PIC 99.
    03  DY PIC 99.
01 NAME-MONTH.
    03  JAN PIC XXXX VALUE " JAN".
    03  FEB PIC XXXX VALUE " FEB".
    03  MAR PIC XXXX VALUE " MAR".
    03  APR PIC XXXX VALUE " APR".
    03  MAY PIC XXXX VALUE " MAY".
    03  JUNE PIC XXXX VALUE "JUNE".
    03  JULY PIC XXXX VALUE "JULY".
    03  AUG PIC XXXX VALUE " AUG".
    03  SEPT PIC XXXX VALUE "SEPT".
    03  OCT PIC XXXX VALUE " OCT".
    03  NOV PIC XXXX VALUE " NOV".
    03  DEC PIC XXXX VALUE " DEC".
01 MONTH-TABLE REDEFINES NAME-MONTH.
    03  MONTHY PIC XXXX OCCURS 12 TIMES INDEXED BY NMM.
01  JOBCNL.
    03  CC PIC X VALUE "1".
    03  PCNL.
        05  CNTYPE PIC XXX.
        05  GO-CNL PIC 999.
        05  PCNLX PIC X(74).
    03  FILLER PIC X(52) VALUE SPACES.
01  JBCNLX.
    03  CC PIC X VALUE "0".
    03  PSTATS PIC X(14) VALUE "RECORD COUNTS".
    03  FILLER PIC X(6) VALUE SPACES.
    03  XCARD PIC X(8) VALUE "CARDS IN".
    03  CSTAT PIC ZZZZZ9.
    03  XDRLNS PIC X(20) VALUE " SITES TO BE CHANGED".
    03  DRLNS PIC ZZZZZ9.
    03  XDATIZN PIC X(15) VALUE " CHANGE RECORDS".
    03  DATAIN PIC ZZZZZ9.
    03  XFAIN PIC X(19) VALUE " UNCHANGED MASTERS ".
    03  FADATA PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XNOUT PIC X(12) VALUE " NEW MASTER ".
    03  FOUT PIC ZZZZZ9.
```

```
01  HDR1.
    03  CC1 PIC 9 VALUE 1.
    03  FILLER PIC X.
    03  LDATE PIC X(6) VALUE "DATE: ".
    03  RMTH PIC X(4).
    03  FILLER PIC X.
    03  RDY PIC Z9.
    03  XB PIC XXXX VALUE ", 19".
    03  RYR PIC XX.
    03  FILLER PIC X(21).
    03  TITLEA PIC X(60).
    03  FILLER PIC X(21).
    03  PGE PIC X(4) VALUE "PAGE".
    03  PNUM PIC ZZZ9.
    03  FILLER PIC X(2).
01  HDR2.
    03  CC PIC 9 VALUE 0.
    03  USGS PIC X(7) VALUE " USGS: ".
    03  GSNAME PIC X(20).
    03  FILLER PIC XXX.
    03  BASINL PIC X(7) VALUE "SOURCES".
    03  FILLER PIC X.
    03  SRCEID PIC X(50).
    03  SRCNAM PIC X(30).
    03   CNLZZZ PIC X(6).
    03  FILLER PIC X(8).
01  DATAA.
    03  ID-CNL PIC IS XXX.
    03  ID-NUM PIC IS 999.
    03  TITLEX.
        05  SRCAGC.
            07  GNAME PIC X(20).
            07  CNAME PIC X(30).
        05  FILLER PIC X(10).
    03  FILLER PIC X(14).
01  KEYIDL.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 99.
01  KEYIDX.
    03  KEYLAT.
        05  LATDEG PIC 99 VALUE 99.
        05  LATMIN PIC 99 VALUE 99.
        05  LATSEC PIC 99 VALUE 99.
    03 KEYLNG.
        05  LNGDEG PIC 999 VALUE 999.
        05  LNGMIN PIC 99 VALUE 99.
        05  LNGSEC PIC 99 VALUE 99.
    03  KEYDUP PIC 99 VALUE 99.
```

```
01  RTW000.
    05 C1-KEYID.
       07 KEY-LATITUDE              PIC 9(6).
       07 KEY-LONGITUDE             PIC 9(7).
       07 KEY-DUP-NUM               PIC 99.
    05 REC-NUM                      PIC 999.
    05 REC-LENGTH                   PIC 999.
    05 ACCESS-CODE                  PIC X.
    05 C2-SITE-TYPE                 PIC X.
    05 C3-DATA-RELIABILITY          PIC X.
    05 C4-SOURCE-AGENCY             PIC X(5).
    05 C5-PROJECT-NUM.
        07  BASIN-1                 PIC X(7).
        07  BASIN-2                 PIC X(6).
    05 C6-DISTRICT                  PIC 999.
    05 C7-STATE                     PIC 99.
    05 C8-COUNTY                    PIC 999.
    05 C9-LATITUDE                  PIC 9(6).
    05 C10-LONGITUDE                PIC 9(7).
    05 C11-LAT-LONG-ACURACY         PIC X.
    05 C12-LOCAL-NUMBER.
        07  LOCAL-SITE              PIC X(14).
        07  LOCAL-REST              PIC X(6).
    05 C13-LAND-NET-LOC             PIC X(30).
    05 C14-LOCATION-MAP-ID          PIC X(30).
    05 C15-MAP-SCALE                PIC X(6).
    05 C16-ALTITUDE                 PIC 9(5)V99.
    05 C17-ALTITUDE-METHOD          PIC X.
    05 C18-ALTITUDE-ACCURACY        PIC XXX.
    05 C19-TOPO-SETTING             PIC X.
    05 C20-OWDC-HYDRO-UNIT          PIC X(8).
    05 C21-DATE-CONSTRUCTED.
        07  F-C-MONTH               PIC XX.
        07  F-C-DAY                 PIC XX.
        07  F-C-YEAR                PIC XXXX.
    05 C23-SITE-USE                 PIC X.
    05 C24-WATER-USE                PIC X.
    05 C25-SECOND-WATER-USE         PIC X.
    05 C26-THIRD-WATER-USE          PIC X.
    05 C27-HOLE-DEPTH               PIC 9(5)V99.
    05 C28-WELL-DEPTH               PIC 9(5)V99.
    05 C29-WELL-DEPTH-SOURCE        PIC X.
    05 C30-WATER-LEVEL              PIC 9(5)V99.
    05 C31-WATER-LEVEL-DATE         PIC 9(8).
    05 C32-DATE-ACCRCY-WL           PIC X.
    05 C33-WATER-LEVEL-SOURCE       PIC X.
    05 C34-MEAS-METHOD-WL           PIC X.
    05 C35-PUMP-USED                PIC X.
    05 C36-GEOHYDRO-DATA-SOURCE     PIC X.
    05 C37-SITE-STATUS              PIC X.
    05 C40-LAST-UPDATE              PIC X(8).
    05 C41-VERIFIED                 PIC X.
    05 MEAS-PNT-HEIGHT              PIC 999V99.
    05 MEAS-PNT-DATE                PIC 9(8).
    05 CURECD                       PIC X.
```

```
PROCEDURE DIVISION.
NOTE-A.
     NOTE THE PROGRAM STARTS WITH THIS INITIALIZATION
     ROUTINE THAT OPENS THE INPUT AND LISTING FILES
     AND THEN THE DATE STORED IN THE SYSTEM IS READ
     AND USED TO SET UP THE DATE IN THE FIRST
     HEADING LINE.  NO RETURN FROM THE MAINLINE
     ROUTINE IS MADE TO THIS PROCEDURE.
BEGIN.
    OPEN INPUT CARD-FILE.
    OPEN OUTPUT LIST-FILE.
    ACCEPT WKDATE FROM DATE.
    ACCEPT JULD FROM DAY.
    SET NMM TO MM.
    MOVE MONTHY (NMM) TO RMTH OF HDR1.
    MOVE DY TO RDY OF HDR1.
    MOVE YY TO RYR OF HDR1.
NOTE-B.
     NOTE THIS ROUTINE READS THE CONTROL CARDS AND MOVES
     THE GENERAL DESCRIPTION OF THE REPORT TO THE
     HEADING AREA.  THEN A BRANCH IS TAKEN TO THE
     APPROPRIATE STARTING ROUTINE DEPENDING ON THE
     NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
     THERE IS NO RETURN TO THIS ROUTINE.
NOTE-C.
    NOTE NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
    THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
    ROUTINES.
STARTA.
    OPEN INPUT CHNGEL, FILEA. OPEN OUTPUT FILEB, FILEC.
    READ FILEA AT END DISPLAY
    "****700D2 NO MASTER FILE INPUT DATA", MOVE ALA TO MSW,
         MOVE ZEROS TO KEYIDM OF OLDMAS.
    MOVE NU1 TO NFLEIA. MOVE 0 TO NDETLN, NDATIN, NFLEOA.
    MOVE KEYIDM OF OLDMAS TO KEYIDX.
RDCHNG.
    READ CHNGEL AT END DISPLAY
    "****701D2 END OF CHANGE INPUT", MOVE ALA TO CSW, GO TO CMC.
    IF KEYID OF CNGREC > KEYIDL, ADD NU1 TO TFILEC,  DISPLAY
    "****702D2 CHANGE FILE OUT OF SEQUENCE", TFILEC,
        STOP RUN ELSE ADD NU1 TO NDATIN.
    IF KEYID OF CNGREC = KEYIDL AND RNUM OF CNGMAS = RNUM OF
        CNGREC, GO TO PCHG.
    IF KEYID OF CNGREC = KEYIDL AND RNUM OF CNGMAS NOT = RNUM OF
        CNGREC, MOVE ZEROS TO KEYIDL.
    IF KEYID OF CNGREC < KEYIDL, MOVE KEYID OF CNGREC TO KEYIDL.
    IF DSW = 1, PERFORM EDTMAC.
CMPMC.
    IF KEYIDX = KEYIDL AND RNUM OF OLDMAS = RNUM OF CNGREC,
        MOVE OLDMAS TO CNGMAS, GO TO PCHG.
    IF KEYIDX = KEYIDL AND RNUM OF OLDMAS < RNUM OF CNGREC,
        GO TO RDMAS.
    IF KEYIDX > KEYIDL, GO TO RDMAS.
    IF KEYIDX < KEYIDL, GO TO RDCHNG.
```

```
RDMAS.
     IF MSW = ALA, GO TO CMC.
     READ FILEA AT END DISPLAY
     "****703D1 END OF MASTER FILE", MOVE ALA TO MSW,
        GO TO CMC.
     IF KEYIDM OF OLDMAS > KEYIDX, DISPLAY
     "****704D1 MASTER FILE OUT OF SEQUENCE", STOP RUN
         ELSE MOVE KEYIDM OF OLDMAS TO KEYIDX.
     GO TO CMPMC.
CMC.
     IF DSW = 1 AND CSW = ALA, PERFORM EDTMAC.
     DISPLAY "OLDMAS     ", OLDMAS.
     DISPLAY "NEWMAS     ", NEWMAS.
     DISPLAY "CNGMAS     ", CNGMAS.
     DISPLAY "CNGREC     ", CNGREC.
     IF ( CSW = ALA AND MSW = ALA ) GO TO STOP3.
     IF CSW = ALA, MOVE ZEROS TO KEYIDL, GO TO RDMAS.
     IF MSW = ALA, MOVE ZEROS TO KEYIDX, GO TO RDCHNG.
     DISPLAY "****728D2 INCORRECT END OF FILES". STOP RUN.
P000.
     MOVE CNGMAS TO RTW000.
     IF CNUM OF CNGREC = 012 AND CCSR OF CNGREC = ALM, MOVE NU1
         TO DSW, MOVE NEWDAT OF CNGREC TO C12-LOCAL-NUMBER.
     IF CNUM OF CNGREC = 012 AND CCSR OF CNGREC = ALD, MOVE NU1
         TO DSW, MOVE SPACES TO C12-LOCAL-NUMBER.
     IF CNUM OF CNGREC = 014 AND CCSR OF CNGREC = ALM, MOVE NU1
         TO DSW, MOVE NEWDAT OF CNGREC TO C14-LOCATIUN-MAP-ID.
     IF CNUM OF CNGREC = 014 AND CCSR OF CNGREC = ALD, MOVE NU1
         TO DSW, MOVE SPACES TO C14-LOCATION-MAP-ID.
     IF CNUM OF CNGREC = 015 AND CCSR OF CNGREC = ALM, MOVE NU1
         TO DSW, MOVE NEWDAT OF CNGREC TO C15-MAP-SCALE.
     IF CNUM OF CNGREC = 015 AND CCSR OF CNGREC = ALD, MOVE NU1
         TO DSW, MOVE SPACES TO C15-MAP-SCALE.
     MOVE RTW000 TO CNGMAS. GO TO RDCHNG.
P001.
     GO TO RDCHNG.
P002.
     GO TO RDCHNG.
P005.
     GO TO RDCHNG.
P042.
     GO TO RDCHNG.
P047.
     GO TO RDCHNG.
P055.
     GO TO RDCHNG.
P058.
     GO TO RDCHNG.
P072.
     GO TO RDCHNG.
P076.
     GO TO RDCHNG.
P082.
     GO TO RDCHNG.
```

```
P090.
    GO TO RDCHNG.
P094.
    GO TO RDCHNG.
P098.
    GO TO RDCHNG.
P105.
    GO TO RDCHNG.
P114.
    GO TO RDCHNG.
P121.
    GO TO RDCHNG.
P127.
    GO TO RDCHNG.
P134.
    GO TO RDCHNG.
P146.
    GO TO RDCHNG.
P158.
    GO TO RDCHNG.
P164.
    GO TO RDCHNG.
P171.
    GO TO RDCHNG.
P180.
    GO TO RDCHNG.
P183.
    GO TO RDCHNG.
P186.
    GO TO RDCHNG.
P189.
    GO TO RDCHNG.
P192.
    GO TO RDCHNG.
P198.
    GO TO RDCHNG.
P203.
    GO TO RDCHNG.
P208.
    GO TO RDCHNG.
P212.
    GO TO RDCHNG.
P219.
    GO TO RDCHNG.
P234.
    GO TO RDCHNG.
P250.
    GO TO RDCHNG.
P277.
    GO TO RDCHNG.
P320.
    GO TO RDCHNG.
```

```
PCHG.
    DISPLAY "PCHG" NDATIN.
    IF RNUM OF CNGREC = 000, GO TO P000.
    IF RNUM OF CNGREC = 001, GO TO P001.
    IF RNUM OF CNGREC = 002, GO TO P002.
    IF RNUM OF CNGREC = 003, GO TO P003.
    IF RNUM OF CNGREC = 042, GO TO P042.
    IF RNUM OF CNGREC = 047, GO TO P047.
    IF RNUM OF CNGREC = 055, GO TO P058.
    IF RNUM OF CNGREC = 058, GO TO P058.
    IF RNUM OF CNGREC = 072, GO TO P072.
    IF RNUM OF CNGREC = 076, GO TO P076.
    IF RNUM OF CNGREC = 082, GO TO P082.
    IF RNUM OF CNGREC = 090, GO TO P090.
    IF RNUM OF CNGREC = 094, GO TO P094.
    IF RNUM OF CNGREC = 098, GO TO P098.
    IF RNUM OF CNGREC = 105, GO TO P105.
    IF RNUM OF CNGREC = 114, GO TO P114.
    IF RNUM OF CNGREC = 121, GO TO P121.
    IF RNUM OF CNGREC = 127, GO TO P127.
    IF RNUM OF CNGREC = 134, GO TO P134.
    IF RNUM OF CNGREC = 146, GO TO P146.
    IF RNUM OF CNGREC = 158, GO TO P158.
    IF RNUM OF CNGREC = 164, GO TO P164.
    IF RNUM OF CNGREC = 171, GO TO P171.
    IF RNUM OF CNGREC = 180, GO TO P180.
    IF RNUM OF CNGREC = 183, GO TO P183.
    IF RNUM OF CNGREC = 186, GO TO P189.
    IF RNUM OF CNGREC = 189, GO TO P189.
    IF RNUM OF CNGREC = 192, GO TO P192.
    IF RNUM OF CNGREC = 198, GO TO P198.
    IF RNUM OF CNGREC = 203, GO TO P203.
    IF RNUM OF CNGREC = 208, GO TO P208.
    IF RNUM OF CNGREC = 212, GO TO P212.
    IF RNUM OF CNGREC = 219, GO TO P219.
    IF RNUM OF CNGREC = 234, GO TO P234.
    IF RNUM OF CNGREC = 250, GO TO P250.
    IF RNUM OF CNGREC = 277, GO TO P277.
    IF RNUM OF CNGREC = 320, GO TO P320.
    DISPLAY "****729D2 INVALID RECORD TYPE", RNUM OF CNGREC.
    DISPLAY CNGREC.  GO TO RDCHNG.
EDTMAC.
    DISPLAY "EDTMAC" NDATIN.
    IF RNUM OF CNGMAS = 000, PERFORM E000.
    IF RNUM OF CNGMAS = 001, PERFORM E001.
    IF RNUM OF CNGMAS = 002, PERFORM E002.
    IF RNUM OF CNGMAS = 003, PERFORM E003.
    IF RNUM OF CNGMAS = 042, PERFORM E042.
    IF RNUM OF CNGMAS = 047, PERFORM E047.
    IF RNUM OF CNGMAS = 055, PERFORM E058.
    IF RNUM OF CNGMAS = 058, PERFORM E058.
    IF RNUM OF CNGMAS = 072, PERFORM E072.
    IF RNUM OF CNGMAS = 076, PERFORM E076.
    IF RNUM OF CNGMAS = 082, PERFORM E082.
    IF RNUM OF CNGMAS = 090, PERFORM E090.
```

```
    IF RNUM OF CNGMAS = 094, PERFORM E094.
    IF RNUM OF CNGMAS = 098, PERFORM E098.
    IF RNUM OF CNGMAS = 105, PERFORM E105.
    IF RNUM OF CNGMAS = 114, PERFORM E114.
    IF RNUM OF CNGMAS = 121, PERFORM E121.
    IF RNUM OF CNGMAS = 127, PERFORM E127.
    IF RNUM OF CNGMAS = 134, PERFORM E134.
    IF RNUM OF CNGMAS = 146, PERFORM E146.
    IF RNUM OF CNGMAS = 158, PERFORM E158.
    IF RNUM OF CNGMAS = 164, PERFORM E164.
    IF RNUM OF CNGMAS = 171, PERFORM E171.
    IF RNUM OF CNGMAS = 180, PERFORM E180.
    IF RNUM OF CNGMAS = 183, PERFORM E183.
    IF RNUM OF CNGMAS = 186, PERFORM E189.
    IF RNUM OF CNGMAS = 189, PERFORM E189.
    IF RNUM OF CNGMAS = 192, PERFORM E192.
    IF RNUM OF CNGMAS = 198, PERFORM E198.
    IF RNUM OF CNGMAS = 203, PERFORM E203.
    IF RNUM OF CNGMAS = 208, PERFORM E208.
    IF RNUM OF CNGMAS = 212, PERFORM E212.
    IF RNUM OF CNGMAS = 219, PERFORM E219.
    IF RNUM OF CNGMAS = 234, PERFORM E234.
    IF RNUM OF CNGMAS = 250, PERFORM E250.
    IF RNUM OF CNGMAS = 277, PERFORM E277.
    IF RNUM OF CNGMAS = 320, PERFORM E320.
    WRITE NEWMAS FROM CNGMAS. MOVE ZERO TO DSW.
E000.
    DISPLAY CNGMAS.
E001.
    DISPLAY CNGMAS.
E002.
    DISPLAY CNGMAS.
E003.
    DISPLAY CNGMAS.
E042.
    DISPLAY CNGMAS.
E047.
    DISPLAY CNGMAS.
E055.
    DISPLAY CNGMAS.
E058.
    DISPLAY CNGMAS.
E072.
    DISPLAY CNGMAS.
E076.
    DISPLAY CNGMAS.
E082.
    DISPLAY CNGMAS.
E090.
    DISPLAY CNGMAS.
E094.
    DISPLAY CNGMAS.
E098.
    DISPLAY CNGMAS.
```

```
E105.
    DISPLAY CNGMAS.
E114.
    DISPLAY CNGMAS.
E121.
    DISPLAY CNGMAS.
E127.
    DISPLAY CNGMAS.
E134.
    DISPLAY CNGMAS.
E146.
    DISPLAY CNGMAS.
E158.
    DISPLAY CNGMAS.
E164.
    DISPLAY CNGMAS.
E171.
    DISPLAY CNGMAS.
E180.
    DISPLAY CNGMAS.
E183.
    DISPLAY CNGMAS.
E186.
    DISPLAY CNGMAS.
E189.
    DISPLAY CNGMAS.
E192.
    DISPLAY CNGMAS.
E198.
    DISPLAY CNGMAS.
E203.
    DISPLAY CNGMAS.
E208.
    DISPLAY CNGMAS.
E212.
    DISPLAY CNGMAS.
E219.
    DISPLAY CNGMAS.
E234.
    DISPLAY CNGMAS.
E250.
    DISPLAY CNGMAS.
E277.
    DISPLAY CNGMAS.
E320.
    DISPLAY CNGMAS.
STARTB.
    STOP RUN.
STARTC.
    STOP RUN.
STOP1.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "CONTROL INPUT FILE EMPTY".
    STOP RUN.
```

```
STOP2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.
STOP3.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY TFILEB.
    MOVE INCRDS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FADATA.
    MOVE NDETLN TO DRLNS.
    MOVE NFLEOA TO FOUT.
    WRITE LIST-LINE FROM JBCNLX.
    DISPLAY "NORMAL JOB TERMINATION".
    DISPLAY "ERRORS = ", ACOUNT.
    CLOSE CARD-FILE, LIST-FILE.
    STOP RUN.
STOP4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP4".
    STOP RUN.
```

```
IDENTIFICATION DIVISION.
PROGRAM-ID. BASIN3.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER.  CDC CYBER-175.
OBJECT-COMPUTER. CDC CYBER-175.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEB ASSIGN TO MGQWIN
    RESERVE 4 ALTERNATE AREAS
    ORGANIZATION IS SEQUENTIAL.
    SELECT PLTFLE ASSIGN TO PLTDSK
    ORGANIZATION IS SEQUENTIAL.
    SELECT ERRFLE ASSIGN TO ERDISK.
DATA DIVISION.
FILE SECTION.
FD  CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
01  CNL-CARD.
    03 ID-CNL PIC IS XXX.
    03 ID-NUM PIC 999.
    03 ID-DES PIC X(74).
FD  LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
01  LIST-LINE .
    03  CC PIC X.
    03  LLPRT.
        05  FILLER PIC X.
        05  P0 PIC X(10).
        05  FILLER PIC XXX.
        05  P1 PIC X(10).
        05  FILLER PIC XXX.
        05  P2 PIC X(10).
        05  FILLER PIC XXX.
        05  P3 PIC X(10).
        05  FILLER PIC XXX.
        05  P4 PIC X(10).
        05  FILLER PIC XXX.
        05  P5 PIC X(10).
        05  FILLER PIC XXX.
        05  P6 PIC X(10).
        05  FILLER PIC XXX.
        05  P7 PIC X(10).
        05  FILLER PIC XXX.
        05  P8 PIC X(10).
        05  FILLER PIC XXX.
        05  P9 PIC X(10).
    03  PIND PIC X(4).
```

```
FD  FILEB
    DATA RECORD IS OLDMAS
    RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY
    BLOCK CONTAINS 4793 TO 5120 CHARACTERS
    LABEL RECORDS ARE STANDARD.
01  OLDMAS.
    03  KEYID.
        05 KEYLAT.
           07  KLD  PIC 99.
           07  KLM  PIC 99.
           07  KLS  PIC 99.
        05 KEYLNG.
           07  KLGD  PIC 999.
           07  KLGM  PIC 99.
           07  KLGS  PIC 99.
        05 KEYDUP PIC 99.
    03  RNUM PIC IS 999.
    03  LNUM PIC IS 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES,
            DEPENDING ON LNUM OF OLDMAS.
FD  PLTFLE
    DATA RECORD IS RECOUT
    LABEL RECORDS ARE OMITTED.
01 RECOUT .
    03  CNRECR PIC 9(4).
    03  KEYID.
        05 KEYLAT PIC 9(6).
        05 KEYLNG PIC 9(7).
        05 KEYDUP PIC 99.
    03  LCWLID.
        05  LCSITE PIC X(14).
        05  LCREST PIC X(6).
    03  DDEPTH PIC 9(5).
    03  IDEPTH PIC X.
    03  ALTUDE PIC 9(5).
    03  PPDISH PIC 9(5).
    03  PPYEAR PIC X(4).
    03  FCYEAR PIC X(4).
    03  WTRUSE PIC X.
    03  INLOGS PIC X(8).
    03  OPNTOP  PIC 9(5)V99.
    03  OPNTYP  PIC X.
    03  CDIAM  PIC 999V99.
    03  FINISH  PIC X.
    03  WTRLVL2  PIC -9(4)V99.
    03  WLYEAR PIC X(4).
    03  WLMETH2  PIC X.
    03  WLCNT2  PIC 99.
    03  CNTQ10  PIC 99.
    03  CNTQ20  PIC 99.
    03  CNTQ30  PIC 99.
FD  ERRFLE
     LABEL RECORDS ARE OMITTED
     DATA RECORD IS ERREC.
01  ERREC.
     03  ERPRT  PIC X(132).
```

```
WORKING-STORAGE SECTION.
77  AC  PIC 9(4).
77  ACOUNT PIC 9(6) VALUE 0.
77  BC  PIC 9(4).
77  BLKCNT  PIC 9 VALUE 0.
77  BT PIC 999.
77  BTA PIC 999.
77  BTB PIC 999.
77  CNBLNK PIC 9(6) VALUE 0.
77  CNRECN  PIC 9(6) VALUE ZEROES.
77  CNTQW1  PIC 99 VALUE ZEROS.
77  CNTQW2  PIC 99 VALUE ZEROS.
77  CNTQW3  PIC 99 VALUE ZEROS.
77  CK  PIC 9(4).
77  COUNTA PICTURE 9(3) VALUE 0.
77  COUNTN PIC 9(6) VALUE 0.
77  COUNT2 PIC 9(6) VALUE 0.
77  COUNT2G PIC 9(6) VALUE 0.
77  CTRTOT PIC 9(8) VALUE 0.
77  DA1  PIC 9(4).
77  DA2  PIC 9(4).
77  DA3  PIC 9(4).
77  DAN  PIC 9(4).
77  DCOUNT PIC 9(6) VALUE 0.
77  DN1  PIC 9(4).
77  DN2  PIC 9(4).
77  EKEYID PIC 9(15) VALUE ZEROS.
77  ELNCNT  PIC 999 VALUE 000.
77  ENTEMP  PIC 999 VALUE ZEROS.
77  EPGCNT  PIC 999 VALUE 001.
77  ERRCNT PIC 9(6) VALUE 0.
77  ERRSW PIC X VALUE "Y".
77  ESEQ PIC 9(5) VALUE 0.
77  ESW PIC X VALUE "X".
77  EXTSW  PIC X VALUE "Y".
77  E5P0 PIC ZZZZZ.
77  FRSW PIC X VALUE "F".
77  FRTSW PIC X VALUE "F".
77  FDCLAT USAGE IS COMP-2.
77  FLATD USAGE IS COMP-2.
77  FLATM USAGE IS COMP-2.
77  FLATS USAGE IS COMP-2.
77  FDCLNG USAGE IS COMP-2.
77  FLNGD USAGE IS COMP-2.
77  FLNGM USAGE IS COMP-2.
77  FLNGS USAGE IS COMP-2.
77  GWCOUNT PIC 9(6) VALUE 0.
77  HCOUNT PIC 9(6) VALUE 0.
77  HSW PIC X VALUE "X".
77  INCRDS PIC 9(6) VALUE 0.
```

```
77  KEYIDL PIC 9(15) VALUE 0.
77  KTEST USAGE IS COMP-2.
77  LATLNA PIC 9(13).
77  LATLNB PIC 9(13).
77  LA1  PIC 9(4).
77  LA2  PIC 9(4).
77  LNCNT PIC 99 VALUE 70.
77  LN1  PIC 9(4).
77  LN2  PIC 9(4).
77  LOGSW  PIC X VALUE "N".
77  LSW PIC X VALUE "X".
77  LTEST USAGE IS COMP-2.
77  LX  PIC 9(4).
77  NDATIN PIC 9(6) VALUE 0.
77  NDETLN PIC 9(6) VALUE 0.
77  NFLETA PIC 9(6) VALUE 0.
77  NFLEOA PIC 9(6) VALUE 0.
77  NHOLD PIC 9(6).
77  NLATMS PIC 9(6).
77  NLATDC PIC 99V9999.
77  NLNGMS PIC 9(7).
77  NLNGDC PIC 999V9999.
77  NMM  PIC 999.
77  NSW PIC X VALUE "X".
77  OCOUNT PIC 9(6) VALUE 0.
77  PCOUNT PIC 9(6) VALUE 0.
77  PGCNT PIC 999 VALUE 0.
77  PRTSW PIC X VALUE "F".
77  SCOUNT PIC 9(6) VALUE 0.
77  SLSW  PIC X VALUE "Y".
77  TFILEB PIC 9(6) VALUE 0.
77  WTLCNT  PIC 99 VALUE ZEROS.
77  WLDTMP  PIC X(4) VALUE SPACES.
77  WLTEST  PIC X(7).
01  LABELF.
    03  IDENTX PIC XXXX.
    03  FILLER PIC X(17).
    03  REKCNT PIC 9(6).
    03  FILLER PIC X(53).
01 WKDATE.
    03  YY PIC 99.
    03  MM PIC 99.
    03  DY PIC 99.
01 NAME-MONTH.
    03  JAN PIC XXXX VALUE " JAN".
    03  FEB PIC XXXX VALUE " FEB".
    03  MAR PIC XXXX VALUE " MAR".
    03  APR PIC XXXX VALUE " APR".
    03  MAY PIC XXXX VALUE " MAY".
    03  JUNE PIC XXXX VALUE "JUNE".
    03  JULY PIC XXXX VALUE "JULY".
    03  AUG PIC XXXX VALUE " AUG".
    03  SEPT PIC XXXX VALUE "SEPT".
    03  OCT PIC XXXX VALUE " OCT".
    03  NOV PIC XXXX VALUE " NOV".
    03  DEC PIC XXXX VALUE " DEC".
```

```
01 MONTH-TABLE REDEFINES NAME-MONTH.
    03  MONTHY PIC XXXX OCCURS 12 TIMES .
01  JOBCNL .
    03  CC PIC X VALUE "1".
    03  PCNL.
        05  CNTYPE PIC XXX.
        05  GO-CNL PIC 999.
        05  PCNLX PIC X(74).
    03  FILLER PIC X(52) VALUE SPACES.
01  JBCNLX .
    03  CC PIC X VALUE "0".
    03  PSTATS PIC X(14) VALUE "RECORD COUNTS".
    03  FILLER PIC X(6) VALUE SPACES.
    03  XCARD PIC X(8) VALUE "CARDS IN".
    03  CSTAT PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDRLNS PIC X(12) VALUE "DETAIL LINES".
    03  DRLNS PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDATIZN PIC X(7) VALUE "DATA IN".
    03  DATAIN PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XFAIN PIC X(11) VALUE "OLD-DATA I".
    03  FADATA PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XNOUT PIC X(12) VALUE "FMT DATA OUT".
    03  FOUT PIC ZZZZZ9.
01  HDR1 .
    03 CC1 PIC 9 VALUE 1.
    03 FILLER PIC X.
    03 LDATE PIC X(6) VALUE "DATE: ".
    03 RMTH PIC X(4).
    03 FILLER PIC X.
    03 RDY PIC Z9.
    03 XB PIC XXXX VALUE ", 19".
    03 RYR PIC XX.
    03 FILLER PIC X(21).
    03 TITLEA PIC X(60).
    03 FILLER PIC X(21).
    03 PGE PIC X(4) VALUE "PAGE".
    03 PNUM PIC ZZZ9.
    03 FILLER PIC X(2).
01  HDR2 .
    03 CC PIC 9 VALUE 0.
    03 USGS PIC X(7) VALUE " USGS: ".
    03 GSNAME PIC X(20).
    03 FILLER PIC XXX.
    03 BASINL PIC X(7) VALUE "SOURCES".
    03 FILLER PIC X.
    03 SRCEID PIC X(50).
    03 SRCNAM PIC X(30).
    03 FILLER PIC X(14).
```

```
01  HDR3 .
    03  CC PIC 9 VALUE 0.
    03  FILLER PIC X VALUE SPACES.
    03  FILLER PIC X(3) VALUE "REC".
    03  FILLER PIC X(5).
    03  FILLER PIC X(19) VALUE "SITE IDENTIFICATION".
    03  FILLER PIC X(19).
    03  FILLER PIC X(5) VALUE "DEPTH".
    03  FILLER PIC X(7).
    03  FILLER  PIC X(5) VALUE "CONST".
    03  FILLER  PIC XXX.
    03  FILLER PIC X(3) VALUE "LOG".
    03  FILLER PIC X(5).
    03  FILLER PIC X(12) VALUE "WATER LEVELS".
    03  FILLER PIC XX.
    03  FILLER PIC X(7) VALUE "DISCHRG".
    03  FILLER PIC X(5).
    03  FILLER PIC X(8) VALUE "OPENINGS".
    03  FILLER PIC XX.
    03  FILLER PIC X(7) VALUE "CASINGS".
    03  FILLER PIC X.
    03  FILLER PIC X(10) VALUE "RECORD CNT".
    03  FILLER PIC X.

01  HDR4.
    03  CC PIC X VALUE SPACES.
    03  FILLER PIC X VALUE SPACES.
    03  FILLER PIC X(3) VALUE "NUM".
    03  FILLER PIC X(3).
    03  FILLER PIC X(3) VALUE "LAT".
    03  FILLER PIC X(4).
    03  FILLER PIC X(4) VALUE "LONG".
    03  FILLER PIC X(7).
    03  FILLER PIC X(8) VALUE "LOCAL ID".
    03  FILLER PIC X(12).
    03  FILLER PIC X(4) VALUE "MEAS".
    03  FILLER PIC X.
    03  FILLER PIC XXX VALUE "SRC".
    03  FILLER PIC X.
    03  FILLER PIC X(3) VALUE "ALT".
    03  FILLER PIC XX.
    03  FILLER PIC X(4) VALUE "YEAR".
    03  FILLER  PIC XXXX.
    03  FILLER PIC X(5) VALUE "TYPES".
    03  FILLER PIC XXX.
    03  FILLER PIC X(4) VALUE "MEAS".
    03  FILLER PIC X.
    03  FILLER PIC XX VALUE "YR".
    03  FILLER PIC X.
    03  FILLER PIC X(3) VALUE "MTH".
    03  FILLER PIC XX.
    03  FILLER PIC X(3) VALUE "GPM".
    03  FILLER PIC XX.
    03  FILLER PIC XX VALUE "YR".
    03  FILLER PIC X.
```

```
     03  FILLER PIC XXX VALUE "USE".
     03  FILLER PIC X(3).
     03  FILLER PIC X(3) VALUE "TOP".
     03  FILLER PIC X.
     03  FILLER PIC XXX VALUE "TYP".
     03  FILLER PIC XX.
     03  FILLER PIC X(4) VALUE  "DIAM".
     03  FILLER PIC X.
     03  FILLER PIC X VALUE "F".
     03  FILLER PIC X.
     03  FILLER PIC XX VALUE "WL".
     03  FILLER PIC X.
     03  FILLER PIC XX VALUE "Q1".
     03  FILLER PIC X.
     03  FILLER PIC XX VALUE "Q2".
     03  FILLER PIC X.
     03  FILLER PIC XX VALUE "Q3".
 01  HDR5.
     03  CC PIC 9 VALUE 0.
     03  FILLER PIC X(40).
     03  FILLER PIC X(43) VALUE "SEE LAST PAGE FOR EXPLANATION OF
-"GWSI CODES".
 01  ERHD1.
     03  CC  PIC 9 VALUE 1.
     03  FILLER  PIC X(25) VALUE SPACES.
     03  ETITLE PIC X(80).
     03  FILLER PIC X(14) VALUE SPACES.
     03  FILLER  PIC X(6) VALUE "PAGE :".
     03 EPNUM PIC ZZ9.
 01  ERHD2.
     03  FILLER PIC X(10) VALUE SPACES.
     03  FILLER PIC X(12) VALUE "SWAB/RASA : ".
     03  FILLER PIC X(14) VALUE "STAFF USE ONLY".
     03  FILLER PIC X(14) VALUE SPACES.
     03  FILLER  PIC X(25) VALUE "LISTING OF REJECTED SITES".
 01  ERHD3.
     03  FILLER PIC X(5) VALUE SPACES.
     03  FILLER PIC X(6) VALUE "REJECT".
     03  FILLER PIC XX VALUE SPACES.
     03  FILLER PIC X(5) VALUE "INPUT".
     03  FILLER PIC X(9) VALUE SPACES.
     03  FILLER PIC X(4) VALUE "SITE".
     03  FILLER PIC X(14) VALUE SPACES.
     03  FILLER PIC X(5) VALUE "LOCAL".
     03  FILLER PIC X(20) VALUE SPACES.
     03  FILLER PIC X(10) VALUE "REASON FOR".
     03  FILLER PIC X(25) VALUE SPACES.
     03  FILLER PIC X(10) VALUE "CORRECTION".
 01  ERHD4.
     03  FILLER PIC X(5) VALUE SPACES.
     03  FILLER PIC X(6) VALUE "NUMBER".
     03  FILLER PIC XX VALUE SPACES.
     03  FILLER PIC X(6) VALUE "SEQNCE".
     03  FILLER PIC X(5) VALUE SPACES.
     03  FILLER PIC X(10) VALUE "IDENTIFIER".
```

```
    03  FILLER PIC X(10) VALUE SPACES.
    03  FILLER PIC X(7) VALUE "WELL ID".
    03  FILLER PIC X(19) VALUE SPACES.
    03  FILLER PIC X(9) VALUE "REJECTION".
    03  FILLER PIC X(28) VALUE SPACES.
    03  FILLER PIC X(5) VALUE "NOTES".
01  ERMSGS.
    03  EMSG1  PIC X(25) VALUE "LATITUDE OUTSIDE POLYGON ".
    03  EMSG2  PIC X(25) VALUE "LONGITUDE EAST OF POLYGON".
    03  EMSG3  PIC X(25) VALUE "LONGITUDE WEST OF POLYGON ".
01  ERECLN.
    03  FILLER  PIC X(5) VALUE SPACES.
    03  ERSEQ  PIC ZZZZZ9.
    03  FILLER  PIC XX VALUE SPACES.
    03  INSEQ  PIC ZZZZZ9.
    03  FILLER PIC XXX VALUE SPACES.
    03  ERIDNT  PIC X(15).
    03  FILLER  PIC XXX VALUE SPACES.
    03  ELCLID  PIC X(20).
    03  FILLER  PIC XXX VALUE SPACES.
    03  ENOTE  PIC X(80).
01  DICT-HDR1.
    03  FILLER PIC 9 VALUE 0.
    03  FILLER PIC X(50).
    03  FILLER  PIC X(26) VALUE "EXPLANATION OF GWSI CODES ".
    03  FILLER  PIC X(50).
01  DICT-HDR2.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(9).
    03  FILLER  PIC X(5) VALUE "DEPTH".
    03  FILLER  PIC X(14).
    03  FILLER  PIC X(12) VALUE "WATER LEVELS".
    03  FILLER  PIC X(11).
    03  FILLER  PIC X(9)  VALUE "AVAILABLE".
    03  FILLER  PIC X(14).
    03  FILLER  PIC X(3) VALUE "USE".
    03  FILLER  PIC X(21).
    03  FILLER  PIC X(4) VALUE "TYPE".
    03  FILLER  PIC X(14).
    03  FILLER  PIC X(6) VALUE "FINISH".
    03  FILLER  PIC X(9).
01  DICT-HDR3.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(9).
    03  FILLER  PIC X(6) VALUE "SOURCE".
    03  FILLER  PIC X(12).
    03  FILLER  PIC X(17) VALUE "MEAS METHOD[MTH]".
    03  FILLER  PIC X(7).
    03  FILLER  PIC X(9)  VALUE "LOG TYPES".
    03  FILLER  PIC X(12).
    03  FILLER  PIC X(8) VALUE "OF WATER".
    03  FILLER  PIC X(17).
    03  FILLER  PIC X(8) VALUE "OPENINGS".
```

```
           03  FILLER  PIC X(11).
           03  FILLER  PIC X(5) VALUE "[ F ]".
           03  FILLER  PIC X(10).
       01  DICT-LN1.
           03  FILLER  PIC 9 VALUE 0.
           03  FILLER  PIC X(4).
           03  FILLER  PIC X(17) VALUE "S  RPTING AGENCY".
           03  FILLER  PIC X(4).
           03  FILLER  PIC X(11) VALUE "A  AIRLINE".
           03  FILLER  PIC X(13).
           03  FILLER  PIC X(8) VALUE "A  TIME".
           03  FILLER  PIC X(12).
           03  FILLER  PIC X(19) VALUE "A  AIR CONDITIONING".
           03  FILLER  PIC X(6).
           03  FILLER  PIC X(12) VALUE "F  FRACTURE".
           03  FILLER  PIC X(6).
           03  FILLER  PIC X(19) VALUE "C  POROUS,CONCRETE".
       01  DICT-LN2.
           03  FILLER  PIC 9 VALUE 0.
           03  FILLER  PIC X(4).
           03  FILLER  PIC X(11) VALUE "D  DRILLER".
           03  FILLER  PIC X(10).
           03  FILLER  PIC X(15) VALUE "C  CAL AIRLINE".
           03  FILLER  PIC X(9).
           03  FILLER  PIC X(10) VALUE "B  COLLAR".
           03  FILLER  PIC X(10).
           03  FILLER  PIC X(11) VALUE "B  BOTTLING".
           03  FILLER  PIC X(14).
           03  FILLER  PIC X(12) VALUE "L  LOUVERED".
           03  FILLER  PIC X(6).
           03  FILLER  PIC X(19) VALUE "F  GRAVEL W/PERF  ".
       01  DICT-LN3.
           03  FILLER  PIC 9 VALUE 0.
           03  FILLER  PIC X(4).
           03  FILLER  PIC X(9) VALUE "O  OWNER".
           03  FILLER  PIC X(12).
           03  FILLER  PIC X(13) VALUE "E  ESTIMATED".
           03  FILLER  PIC X(11).
           03  FILLER  PIC X(11) VALUE "C  CALIPER".
           03  FILLER  PIC X(9).
           03  FILLER  PIC X(13) VALUE "C  COMMERCIAL".
           03  FILLER  PIC X(12).
           03  FILLER  PIC X(8) VALUE "M  MESH".
           03  FILLER  PIC X(10).
           03  FILLER  PIC X(17) VALUE "G  GRAVEL,SCREEN".
           03  FILLER  PIC XX.
       01  DICT-LN4.
           03  FILLER  PIC 9 VALUE 0.
           03  FILLER  PIC X(4).
           03  FILLER  PIC X(15) VALUE "A  OTHER GOV'T".
           03  FILLER  PIC X(6).
           03  FILLER  PIC X(17) VALUE "G  PRESSURE GAGE".
           03  FILLER  PIC X(7).
           03  FILLER  PIC X(13) VALUE "D  DRILLER'S".
```

```
    03  FILLER  PIC X(7).
    03  FILLER  PIC X(10) VALUE "D  DEWATER".
    03  FILLER  PIC X(15).
    03  FILLER  PIC X(14) VALUE "P  PERFORATED".
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(17) VALUE "H  HORIZ GALLERY".
    03  FILLER  PIC XX.
01  DICT-LN5.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(18) VALUE "R  OTHER REPORTED".
    03  FILLER  PIC X(3).
    03  FILLER  PIC X(21) VALUE "H  CAL PRESSURE GAGE".
    03  FILLER  PIC X(3).
    03  FILLER  PIC X(12) VALUE "E  ELECTRIC".
    03  FILLER  PIC X(8).
    03  FILLER  PIC X(8) VALUE "E  POWER".
    03  FILLER  PIC  X(17).
    03  FILLER  PIC X(14) VALUE "R  WIRE WOUND".
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(12) VALUE "O  OPEN END".
    03  FILLER  PIC X(5).
01  DICT-LN6.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(8) VALUE "L  LOGS".
    03  FILLER  PIC X(13).
    03  FILLER  PIC X(20) VALUE "L  GEOPHYSICAL LOGS".
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(17) VALUE "F  FLUID CONDUCT".
    03  FILLER  PIC X(3).
    03  FILLER  PIC X(7) VALUE "F  FIRE".
    03  FILLER  PIC X(18).
    03  FILLER  PIC X(15) VALUE "S  SCREEN(UNK)".
    03  FILLER  PIC X(3).
    03  FILLER  PIC X(14) VALUE "P  PERFORATED".
    03  FILLER  PIC X(5).
01  DICT-LN7.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(13) VALUE "G  GEOLOGIST".
    03  FILLER  PIC X(8).
    03  FILLER  PIC X(13) VALUE "M  MANOMETER".
    03  FILLER  PIC X(11).
    03  FILLER  PIC X(13) VALUE "G  GEOLOGIST".
    03  FILLER  PIC X(7).
    03  FILLER  PIC X(11) VALUE "H  DOMESTIC".
    03  FILLER  PIC X(14).
    03  FILLER  PIC X(14) VALUE "T  SAND,POINT".
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(10) VALUE "S  SCREEN".
    03  FILLER  PIC X(9).
```

```
01  DICT-LN8.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(4).
    03  FILLER  PIC X(9) VALUE "Z  OTHER".
    03  FILLER  PIC X(12).
    03  FILLER  PIC X(12) VALUE "R  REPORTED".
    03  FILLER  PIC X(12).
    03  FILLER  PIC X(12) VALUE "H  MAGNETIC".
    03  FILLER  PIC X(8).
    03  FILLER  PIC X(13) VALUE "I  IRRIGATION".
    03  FILLER  PIC X(12).
    03  FILLER  PIC X(10) VALUE "W  WALLED".
    03  FILLER  PIC X(8).
    03  FILLER  PIC X(14) VALUE "T  SAND,POINT".
    03  FILLER  PIC X(5).
01  DICT-LN9.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(25).
    03  FILLER  PIC X(14) VALUE "S  STEEL TAPE".
    03  FILLER  PIC X(10).
    03  FILLER  PIC X(13) VALUE "I  INDUCTIUN".
    03  FILLER  PIC X(7).
    03  FILLER  PIC X(22) VALUE "J  INDUSTRIAL(COOLING)".
    03  FILLER  PIC X(3).
    03  FILLER  PIC X(13) VALUE "X  OPEN HOLE".
    03  FILLER  PIC X(5).
    03  FILLER  PIC X(10) VALUE "W  WALLED".
01  DICT-LN10.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(25).
    03  FILLER  PIC X(17) VALUE "T  ELECTRIC TAPE".
    03  FILLER  PIC X(7).
    03  FILLER  PIC X(13) VALUE "J  GAMMA RAY".
    03  FILLER  PIC X(7).
    03  FILLER  PIC X(9) VALUE "K  MINING".
    03  FILLER  PIC X(16).
    03  FILLER  PIC X(9) VALUE "Z  OTHER".
    03  FILLER  PIC X(9).
    03  FILLER  PIC X(13) VALUE "X  OPEN HOLE".
    03  FILLER  PIC X(5).
01  DICT-LN11.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(25).
    03  FILLER  PIC X(21) VALUE "Y  CAL ELECTRIC TAPE".
    03  FILLER  PIC X(3).
    03  FILLER  PIC X(12) VALUE "K  DIPMETER".
    03  FILLER  PIC X(8).
    03  FILLER  PIC X(12) VALUE "M  MEDICINAL".
    03  FILLER  PIC X(31).
    03  FILLER  PIC X(9) VALUE "Z  OTHER".
    03  FILLER  PIC X(10).
01  DICT-LN12.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(25).
    03  FILLER  PIC X(9) VALUE "Z  OTHER".
    03  FILLER  PIC X(15).
    03  FILLER  PIC X(12) VALUE "L  LATERLOG".
    03  FILLER  PIC X(8).
    03  FILLER  PIC X(13) VALUE "N  INDUSTRIAL".
```

```
01  DICT-LN13.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(12) VALUE "M  MICROLOG".
    03  FILLER  PIC X(8).
    03  FILLER  PIC X(16) VALUE "P  PUBLIC SUPPLY".
01  DICT-LN14.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(11) VALUE "N  NEUTRON".
    03  FILLER  PIC X(9).
    03  FILLER  PIC X(14) VALUE "Q  AQUACULTURE".
01  DICT-LN15.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(9) VALUE "O  LATER".
    03  FILLER  PIC X(11).
    03  FILLER  PIC X(13) VALUE "R  RECREATION".
01  DICT-LN16.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(9) VALUE "P  PHOTO".
    03  FILLER  PIC X(11).
    03  FILLER  PIC X(8) VALUE "S  STOCK".
01  DICT-LN17.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(15) VALUE "Q  RADIOACTIVE".
    03  FILLER  PIC X(5).
    03  FILLER  PIC X(14) VALUE "T  INSTITUTION".
01  DICT-LN18.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(8) VALUE "S  SONIC".
    03  FILLER  PIC X(12).
    03  FILLER  PIC X(9) VALUE "U  UNUSED".
01  DICT-LN19.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(7) VALUE "T  TEMP".
    03  FILLER  PIC X(13).
    03  FILLER  PIC X(15) VALUE "Y  DESALINATION".
01  DICT-LN20.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(14) VALUE "U  GAMMA-GAMMA".
    03  FILLER  PIC X(6).
    03  FILLER  PIC X(8) VALUE "Z  OTHER".
01  DICT-LN21.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(17) VALUE "V  FLUID VELOCITY".
```

```
01  DICT-LN22.
    03  FILLER  PIC 9 VALUE 0.
    03  FILLER  PIC X(49).
    03  FILLER  PIC X(8) VALUE "Z  OTHER".
01  DBUGLNS.
    03  STRLN.
        05  FILLER  PIC 9 VALUE 1.
        05  STRS  PIC X(130) VALUE ALL "*".
    03  DBUG1.
        05  DB11  PIC 9 VALUE 0.
        05  DB12  PIC X(11) VALUE " SITE ID : ".
        05  LTD   PIC XX.
        05  DB12  PIC X VALUE SPACES.
        05  LTM   PIC XX.
        05  DB13  PIC X VALUE SPACES.
        05  LTS   PIC XX.
        05  DB14  PIC XXX VALUE SPACES.
        05  LGD   PIC XXX.
        05  DB15  PIC X VALUE SPACES.
        05  LGM   PIC XX.
        05  DB16  PIC X VALUE SPACES.
        05  LGS   PIC XX.
        05  FILLER PIC X(10) VALUE SPACES.
        05  FILLER  PIC X(20) VALUE "DEBUG CALLED FROM : ".
        05  DBSRC  PIC X(20).
    03  DBUG2.
        05  DB21  PIC 9 VALUE 0.
        05  DB22  PIC X(8) VALUE "TEST2 : ".
    03  DBUG3.
        05  DB31  PIC 9 VALUE 0.
        05  DB32  PIC X(10) VALUE " NLATMS : ".
        05  ELATMS PIC 9(6).
        05  DB33  PIC X(17) VALUE "   LATUDE-1(1) : ".
        05  ELATUDE-1  PIC 9(6).
        05  DB34  PIC X(17) VALUE "   LATUDE-2(1) : ".
        05  ELATUDE-2  PIC 9(6).
        05  DB35  PIC X(12) VALUE "   NLNGMS : ".
        05  ELNGMS PIC 9(7).
        05  DB36  PIC X(11) VALUE "   LOLNG : ".
        05  ELOLNG  PIC 9(7).
        05  DB37  PIC X(11) VALUE "   HILNG : ".
        05  EHILNG  PIC 9(7).
    03  DBUG4.
        05  DB41  PIC 9 VALUE 0.
        05  DB41  PIC X(8) VALUE "VCARD : ".
        05  EVSEQ PIC 9999.
    03  DBUG5.
        05  DB51  PIC 9 VALUE 0.
        05  DB52  PIC X(12) VALUE " LATUDE-L : ".
        05  ELAT-L  PIC 9(6).
        05  DB53  PIC X(14) VALUE "   LNGTUD-L : ".
        05  ELNG-L  PIC 9(7).
        05  DB54  PIC X(14) VALUE "   LATUDE-H : ".
        05  ELAT-H  PIC 9(6).
```

```
    05  DB55  PIC X(14) VALUE "   LNGTUD-H : ".
    05  ELNG-H  PIC 9(7).
    05  DB56  PIC X(13) VALUE "   DECLAT-L : ".
    05  EDL-L  PIC 99.9999.
    05  DB57  PIC X(9) VALUE "   LS1 : ".
    05  ELS1  PIC X.
    05  DB58  PIC X(12) VALUE "  DECLNG-L: ".
    05  EDLG-L  PIC 999.9999.
03  DBUG6.
    05  DB61  PIC 9 VALUE 0.
    05  DB62  PIC X(9) VALUE "  ASIGN: ".
    05  EASIN  PIC X.
    05  DB63  PIC X(13) VALUE "   ACOEFF : .".
    05  EACOF  PIC V99999999.
    05  DB64  PIC X(11) VALUE "   BSIGN : ".
    05  EBSIN  PIC X.
    05  DB648  PIC X(13) VALUE "   BCOEFF : .".
    05  EBCOF  PIC 9V9999999.
    05  DB65  PIC X(19) VALUE SPACES.
    05  DB66  PIC X(10) VALUE "DECLAT-H: ".
    05  EDL-H  PIC 99.9999.
    05  DB67  PIC X(9) VALUE "   LS2 : ".
    05  ELS2  PIC X.
    05  DB68  PIC X(12) VALUE "  DECLNG-H: ".
    05  EDLG-H PIC 999.9999.
03  DBUG7.
    05  DB71  PIC 9 VALUE 0.
    05  DB72  PIC X(9) VALUE " CSIGN : ".
    05  ECSIN  PIC X.
    05  DB73  PIC X(12) VALUE "   KNSTNT : ".
    05  EKNSTNT PIC 9999999.
03  DBUG8.
    05  DB81 PIC 9 VALUE 0.
    05  DB82 PIC X(8) VALUE "LATDMS : ".
    05  ELTDMS  PIC 9(6).
    05  DB83  PIC X(13) VALUE "     FDCLAT: ".
    05  EFDCLT  PIC 9(6).
    05  DB84  PIC X(26) VALUE SPACES.
    05  DB85  PIC X(8) VALUE "LNGDMS: ".
    05  ELGDMS  PIC 9(7).
    05  DB86  PIC X(14) VALUE "     FDCLNG : ".
    05  EFDCLG  PIC 9(7).
03  DBUG9.
    05  DB91  PIC 9 VALUE 0.
    05  DB92  PIC X(10) VALUE SPACES.
    05  DB93  PIC X(10) VALUE "   LATD : ".
    05  ELATD  PIC 99.
    05  DB94  PIC X(10) VALUE "   LATM : ".
    05  ELATM  PIC 99.
    05  DB95  PIC X(10) VALUE "   LATS : ".
    05  ELATS  PIC 99.
    05  DB96  PIC X(26) VALUE SPACES.
    05  DB97  PIC X(10) VALUE "   LNGD : ".
```

```
        05  ELNGD  PIC 999.
        05  DB98  PIC X(10) VALUE "   LNGM : ".
        05  ELNGM  PIC 99.
        05  DB99  PIC X(10) VALUE "   LNGS : ".
        05  ELNGS PIC 99.
    03  DBUG10.
        05  DB101  PIC 9 VALUE 0.
        05  DB102  PIC X(9) VALUE SPACES.
        05  DB103  PIC X(11) VALUE "   FLATD : ".
        05  EFLATD  PIC 99.
        05  DB104  PIC X(11) VALUE "   FLATM : ".
        05  EFLATM  PIC 99.
        05  DB105  PIC X(11) VALUE "   FLATS : ".
        05  EFLATS  PIC 99.
        05  DB106  PIC X(26) VALUE SPACES.
        05  DB107  PIC X(11) VALUE "   FLNGD : ".
        05  EFLNGD  PIC 999.
        05  DB108  PIC X(11) VALUE "   FLNGM : ".
        05  EFLNGM  PIC 99.
        05  DB109  PIC X(11) VALUE "   FLNGS : ".
        05  EFLNGS  PIC 99.
    03  DBUG11.
        05  DB111  PIC 9 VALUE 0.
        05  DB112  PIC X(8) VALUE "TEST21 :".
    03  DBUG12.
        05  DB121  PIC 9 VALUE 0.
        05  DB122  PIC X(8) VALUE "  DA1 : ".
        05  EDA1  PIC 99.
        05  DB123  PIC X(17) VALUE "   ACOEFS(DA1) : ".
        05  ECOF-DA1  PIC .9999999999.
        05  DB124  PIC X(19) VALUE "   ACOEFS(DA1+1) : ".
        05  ECOF-DA2  PIC .9999999999.
        05  DB125  PIC X(17) VALUE "   BCOEFS(DA1) : ".
        05  EBCOF-DA1 PIC .9999999999.
        05  DB126  PIC X(19) VALUE "   BCOEFS(DA1+1) : ".
        05  EBCOF-DA2  PIC .9999999999.
    03  DBUG13.
        05  DB131  PIC 9 VALUE 0.
        05  DB132  PIC X(10) VALUE "  KTEST : ".
        05  EKTEST  PIC 999999.999999.
        05  DB133  PIC X(12)  VALUE "    LTEST : ".
        05  ELTEST PIC 999999.999999.
        05  DB134  PIC X(17) VALUE "   KNSTNS(DA1) : ".
        05  EKNSTNS  PIC 9999.9999.
        05  DB135  PIC X(19) VALUE "   KNSTNS(DA1+1) : ".
        05  EKNSTNS-2 PIC 9999.9999.
        05  DB136  PIC X(7) VALUE "  LN1: ".
        05  ELN1  PIC 9999.
        05  DB137  PIC X(7) VALUE "  LN2: ".
        05  ELN2  PIC 9999.
01 SITELN.
    03  CC PIC 9 VALUE SPACE.
    03  CNRECP PIC ZZZZ9.
    03  FILLER PIC X.
```

```
    03  KEYID.
        05 KEYLAT PIC 9(6).
        05 FILLER PIC X.
        05 KEYLNG PIC 9(7).
        05 FILLER PIC X.
        05 KEYDUP PIC 99.
    03  FILLER PIC X.
    03  LCWLID.
        05  LCSITE PIC X(14).
        05  LCREST PIC X(6).
    03  FILLER PIC X.
    03  DDEPTH PIC ZZZZZ.
    03  FILLER PIC X.
    03  IDEPTH PIC X.
    03  FILLER PIC X.
    03  ALTUDE PIC ZZZZZ.
    03  FILLER PIC X.
    03  FCYEAR PIC X(4).
    03  FILLER PIC X.
    03  INLOGS PIC X(8).
    03  FILLER PIC X.
    03  WTRLVL3 PIC -ZZZZ.
    03  FILLER PIC X.
    03  WLYEAR PIC X(4).
    03  FILLER PIC X.
    03  WLMETH3  PIC X.
    03  FILLER PIC X.
    03  PPDISH PIC ZZZZ9.
    03  FILLER PIC X.
    03  PPYEAR PIC X(4).
    03  FILLER PIC X.
    03  WTRUSE PIC X.
    03  FILLER PIC X.
    03  OPNTOP PIC ZZZZ.99.
    03  FILLER PIC X.
    03  OPNTYP PIC X.
    03  FILLER PIC X.
    03  CDIAM  PIC ZZZ.99.
    03  FILLER PIC X.
    03  FINISH PIC X.
    03  FILLER PIC X.
    03  WLCNT3  PIC Z9.
    03  FILLER PIC X.
    03  CNTQ1S  PIC Z9.
    03  FILLER PIC X.
    03  CNTQ2S PIC Z9.
    03  FILLER PIC X.
    03  CNTQ3S  PIC Z9.
01  DATAA .
    03  ID-CNL PIC IS XXX.
    03  ID-NUM PIC IS 999.
    03  TITLEX.
```

```
        05  SRCAGC.
            07  GNAME PIC X(20).
            07  CNAME PIC X(30).
        05  FILLER PIC X(10).
    03  FILLER PIC X(13).
01  BCARD.
    03  LOLAT PIC 9(6).
    03  LOLNG PIC 9(7).
    03  HILAT PIC 9(6).
    03  HILNG PIC 9(7).
    03  BASIN1 PIC X(7).
    03  BASINX REDEFINES BASIN1.
        05  BASIN2 PIC X(6).
        05  FILLER PIC X.
    03  BTYPE PIC X.
    03  BDESRP PIC X(50).
01  VCARD.
    03  LATUDE-L PIC 9(6).
    03  LNGTUD-L PIC 9(7).
    03  LATUDE-H PIC 9(6).
    03  LNGTUD-H PIC 9(7).
    03  DECLAT-L PIC 99V9999.
    03  LS1 PIC X.
    03  DECLNG-L PIC 999V9999.
    03  DECLAT-H PIC 99V9999.
    03  LS2 PIC X.
    03  DECLNG-H PIC 999V9999.
    03  ASIGN PIC X.
    03  ACOEFF PIC 9V9999999.
    03  BSIGN PIC X.
    03  BCOEFF PIC 9V9999999.
    03  CSIGN PIC X.
    03  KNSTNT PIC 999V9999.
    03  VSEQ PIC 9999.

01  SLN.
    03  CC PIC X VALUE SPACE.
    03  LATUDE-L PIC 9(6).
    03  FILLER PIC XX.
    03  LNGTUD-L PIC 9(7).
    03  FILLER PIC XX.
    03  LATUDE-H PIC 9(6).
    03  FILLER PIC XX.
    03  LNGTUD-H PIC 9(7).
    03  FILLER PIC XX.
    03  DECLAT-L PIC 99V9999.
    03  FILLER PIC XX.
    03  DECLNG-L PIC 999V9999.
    03  FILLER PIC XX.
    03  DECLAT-H PIC 99V9999.
    03  FILLER PIC XX.
    03  DECLNG-H PIC 999V9999.
    03  FILLER PIC XX.
    03  ACOEFF PIC 9.999999.
    03  ASIGN PIC X.
```

```
    03  FILLER PIC XX.
    03  BCOEFF PIC 9.9999999.
    03  BSIGN PIC X.
    03  FILLER PIC XX.
    03  KNSTNT PIC ZZZ.9999.
    03  CSIGN PIC X.
    03  FILLER PIC XX.
    03  VSEQ PIC ZZZZ.
01  LATONE.
    03  LATUDE-1 PIC 9(6) OCCURS 50 TIMES .
01  LNGONE.
    03  LNGTUD-1 PIC 9(7) OCCURS 50 TIMES .
01  LATTWO.
    03  LATUDE-2 PIC 9(6) OCCURS 50 TIMES .
01  LNGTWO.
    03  LNGTUD-2 PIC 9(7) OCCURS 50 TIMES .
01  DATONE.
    03  DECLAT-1 PIC 99V9999 OCCURS 50 TIMES .
01  DNGONE.
    03  DECLNG-1 PIC 999V9999 OCCURS 50 TIMES .
01  DATTWO.
    03  DECLAT-2 PIC 99V9999 OCCURS 50 TIMES .
01  DNGTWO.
    03  DECLNG-2 PIC 999V9999 OCCURS 50 TIMES .
01  ACOEF.
    03  ACOEFS USAGE IS COMP-2 OCCURS 50 TIMES .
01  BCOEF.
    03  BCOEFS USAGE IS COMP-2 OCCURS 50 TIMES .
01  CCOEF.
    03  KNSTNS USAGE IS COMP-2 OCCURS 50 TIMES .
01  LATDMS.
    03  LATD PIC 99.
    03  LATM PIC 99.
    03  LATS PIC 99.
01  LNGDMS.
    03  LNGD PIC 999.
    03  LNGM PIC 99.
    03  LNGS PIC 99.
01  TYPLOG.
    03  LOGTYP PIC X OCCURS 8 TIMES .

01  RTW000.
    03  KEYID.
        05 KEYLAT PIC 9(6).
        05 KEYLNG PIC 9(7).
        05 KEYDUP PIC 99.
    03 RNUM PIC 999.
    03 LNUM PIC 999.
    03  SESAFE PIC X.
    03 SITETY PIC X.
    03 ORELIX PIC X.
    03 REPAGY PIC X(5).
    03  PROJNU.
        05  BASIN1 PIC X(7).
        05  BASIN2 PIC x(6).
```

```
    03 DISTRT PIC XXX.
    03 STATEW PIC XX.
    03 COUNTY PIC XXX.
    03 LLACUR PIC X.
    03 LCWLID.
        05  LCSITE PIC X(14).
        05  LCREST PIC X(6).
    03  LCLNET PIC X(30).
    03 LCLMAP PIC X(30).
    03 SCALEN PIC X(6).
    03 ALTUDE PIC 9(5)V99.
    03 MMEASR PIC X.
    03 MACCUR PIC XXX.
    03 TOPSET PIC X.
    03 OWDCUR PIC X(8).
    03  FCMNTH PIC XX.
    03  FCCDAY PIC XX.
    03  FCYEAR PIC XXXX.
    03 FILLER PIC X.
    03 SITUSE PIC X.
    03 WTRUSE PIC X.
    03 SWTRUS PIC X.
    03 TWTRUS PIC X.
    03 HDEPTH PIC 9(5)V99.
    03 WDEPTH PIC 9(5)V99.
    03 WDSRCE PIC X.
    03 WTRLVL PIC S9(5)V99.
    03  WLDAT.
        05  WLMNTH PIC XX.
        05  WLLDAY PIC XX.
        05  WLYEAR PIC XXXX.
    03 WLDACR PIC X.
    03 WLSRCE PIC X.
    03 WLMETH PIC X.
    03  PUMPUS PIC X.
    03 GHSRCE PIC X.
    03 XTATUS PIC X.
    03 LUPDAT PIC X(8).
    03 VERIFY PIC X.
    03  MSHGHT PIC 999V99.
    03  MSDATE PIC 9(8).
    03  CURECD PIC X.
01  RTQ001.
    03  KEYID.
        05 KEYLAT PIC 9(6).
        05 KEYLNG PIC 9(7).
        05 KEYDUP PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  LATLNG.
        05  LATUDE PIC 9(6).
        05  LNGTUD PIC 9(7).
        05  DUPERK PIC 99.
```

```
    03  STATEC PIC XX.
    03  DISTRT PIC XX.
    03  CONTYC PIC XXX.
    03  SITECD PIC XX.
    03  HYDROC PIC X(8).
    03  TDRAIN PIC X(7).
    03  CDRAIN PIC X(7).
    03  DATUMX PIC X(8).
    03  WDEPTH PIC 9(7)V99.
    03  EPACDE PIC X.
01  RTQ002.
    03  KEYID.
        05 KEYLAT PIC 9(6).
        05 KEYLNG PIC 9(7).
        05 KEYDUP PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  LCWLID.
        05  LCSITE PIC X(14).
        05  LCREST PIC X(6).
    03  FILLER PIC X(28).
    03  GEOLUC PIC X(8).
    03  AOTYPE PIC X.
    03  FILLER PIC X(6).
    03  EPACDE PIC X.
01  RTW058.
    03  KEYID  PIC 9(15).
    03  RNUM  PIC 999.
    03  LNUM  PIC 999.
    03  SESAFE  PIC X.
    03  ENTNUM   PIC 999.
    03  ENTDT.
        05  DTDAY  PIC XX.
        05  DTMNTH  PIC XX.
        05  DTYEAR  PIC XXXX.
    03  DTACCR  PIC X.
    03  DRILNAM   PIC X(12).
    03  FILLER  PIC X(28).
    03  DATSRC  PIC X.
    03  METHOD  PIC X.
    03  FINISH  PIC X.
    03  BOTTOM  PIC 999.
    03  DVMETH  PIC X.
    03  SPCLTR  PIC X.

01  RTW076.
    03  KEYID PIC 9(15).
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  CSGTOP  PIC 9(5)V99.
    03  CSGBOT  PIC 9(5)V99.
    03  CSGDIAM  PIC 999V99.
    03  CSGMTL  PIC X.
    03  CSGTHK  PIC 99V999.
```

```
01  RTW082.
    03  KEYID  PIC 9(15).
    03  RNUM  PIC 999.
    03  LNUM  PIC 999.
    03  SESAFE  PIC X.
    03  OPNTOP  PIC 9(5)V99.
    03  OPNBTM  PIC 9(5)V99.
    03  OPNTYP  PIC X.
    03  TYPMTL  PIC X.
    03  OPNDIAM  PIC 999V99.
    03  OPNWID  PIC 99.999.
    03  OPNLNG  PIC 999.99.

01  RTW198.
    03  KEYID PIC 9(15).
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  TYPELG PIC X.
    03  BDEPTH PIC 9(5)V99.
    03  EDEPTH PIC 9(5)V99.
    03  SRCLOG PIC X.
01  RTW146.
    03 KEYID PIC 9(15).
    03 RNUM PIC 999.
    03 LNUM PIC 999.
    03 SESAFE PIC X.
    03 PPSQNO PIC 999.
    03 PPDATE.
        07  PPMNTH PIC XX.
        07  PPPDAY PIC XX.
        07  PPYEAR PIC XXXX.
    03  PPDTAC PIC X.
    03  PPDISH PIC 9(5)V99.
    03  PPDISS PIC X.
    03  PPMETH PIC X.
    03  PPPWL PIC 9(5)V99.
    03  PPSWL PIC 9(5)V99.
    03  PPLSRC PIC X.
    03  PPLMTH PIC X.
    03  PPPERD PIC 9(4)V9.
    03  PPSCAP PIC 9(5)V99.
    03  FILLER PIC XXX.
01  RTW234.
    03  KEYID  PIC 9(15).
    03  RNUM  PIC 999.
    03  LNUM  PIC 999.
    03  SESAFE  PIC X.
    03  WLDATE.
        05  WLMNTH  PIC XX.
        05  WLDAY  PIC XX.
        05  WLYEAR  PIC XXXX.
```

```
    03  DTACCR  PIC X.
    03  WTRLVL1 PIC S9(4)V99.
    03  WLSTAT  PIC X.
    03  WLMETH1  PIC X.
    03  MEASAC  PIC X.

01  RTEST.
    03  KEYID.
        05 KEYLAT PIC 9(6) VALUE 340900.
        05 KEYLNG PIC 9(7) VALUE 1133650.
        05 KEYDUP PIC 99.
    03  RNUM PIC 999.
    03  LNUM PIC 999.
    03  SESAFE PIC X.
    03  LATLNG.
        05  LATUDE PIC 9(6).
        05  LNGTUD PIC 9(7).
        05  DUPERK PIC 99.
    03  STATEC PIC XX.
    03  DISTRT PIC XX.
    03  CONTYC PIC XXX.
    03  SITECD PIC XX.
    03  HYDROC PIC X(8).
    03  TDRAIN PIC X(7).
    03  CDRAIN PIC X(7).
    03  DATUMX PIC X(8).
    03  WDEPTH PIC 9(7)V99.
    03  EPACDE PIC X.
01  LNTEST.
    03  CC PIC X VALUE SPACE.
    03  FILLER PIC X(8) VALUE " KTEST= ".
    03  KTESTE PIC ZZZ.9(8).
    03  FILLER PIC X(8) VALUE " LTEST= ".
    03  K1E PIC X(12) VALUE " KNSTNS-1 = ".
    03  KNE1 PIC ZZZ.9(8).
    03  L1E PIC X(12) VALUE " KNSTNS-1 = ".
    03  LNE1 PIC ZZZ.9(8).



PROCEDURE DIVISION.
DECLARATIVES.
SFILEB SECTION.
    USE AFTER STANDARD ENDING FILE LABEL
        PROCEDURE ON FILEB.
LFILEB.
    MOVE FILE-LABEL TO LABELF.
    MOVE TFILEB TO REKCNT.
    MOVE LABELF TO FILE-LABEL.
    DISPLAY FILE-LABEL, TFILEB, REKCNT.
END DECLARATIVES.
```

```
*NOTE-A.
*      THE PROGRAM STARTS WITH THIS INITIALIZATION
*      ROUTINE THAT OPENS THE INPUT AND LISTING FILES
*      AND THEN THE DATE STORED IN THE SYSTEM IS READ
*      AND USED TO SET UP THE DATE IN THE FIRST
*      HEADING LINE.  NO RETURN FROM THE MAINLINE
*      ROUTINE IS MADE TO THIS PROCEDURE.
 BEGIN.
      OPEN INPUT CARD-FILE.
      OPEN OUTPUT LIST-FILE.
      ACCEPT WKDATE FROM DATE.
      MOVE MM TO NMM.
      MOVE MONTHY (NMM) TO RMTH OF HDR1.
      MOVE DY TO RDY OF HDR1.
      MOVE YY TO RYR OF HDR1.
      MOVE SPACES TO WLDTMP.
      MOVE ZEROS TO ENTEMP.
*NOTE-B.
*      THIS ROUTINE READS THE CONTROL CARDS AND MOVES
*      THE GENERAL DESCRIPTION OF THE REPORT TO THE
*      HEADING AREA.  THEN A BRANCH IS TAKEN TO THE
*      APPROPRIATE STARTING ROUTINE DEPENDING ON THE
*      NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
*      THERE IS NO RETURN TO THIS ROUTINE.
 READC.
      READ CARD-FILE AT END GO TO STOP1.
      IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
      IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
      ADD 1 TO INCRDS.
      MOVE CNL-CARD TO PCNL.
      READ CARD-FILE INTO DATAA AT END GO TO STOP4.
      IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
      IF ID-NUM OF DATAA = 010 MOVE TITLEX OF DATAA
          TO TITLEA OF HDR1 ELSE GO TO STOP4.
      ADD 1 TO INCRDS.
      IF GO-CNL = 001, GO TO STARTA, ELSE GO TO STOP2.
 STARTA.
      OPEN INPUT FILEB.
      OPEN OUTPUT ERRFLE.
      OPEN OUTPUT PLTFLE.
 READB.
      READ CARD-FILE INTO BCARD AT END GO TO STOP5.
      ADD 1 TO INCRDS.
*NOTE-C.
*     NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
*     THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
*     ROUTINES.  A RETURN TO READB1 IS MADE FROM THE
*     ROUTINE ENDFLE WHEN A TAPE RECORD IS LESS THAN THE
*     THE LOWEST LATITUDE IN THE CURRENT BASIN AND THERE
*     IS NOT A TERMINATION RECORD SIGNALLED BY 000000 IN
*     THE FIRST SIX POSITIONS OF THE CURRENT RECORD.
```

```
 READB1.
      MOVE BDESRP OF BCARD TO SRCEID OF HDR2.
      MOVE BASIN1 OF BCARD TO BASINL OF HDR2.
*NOTE-D.
*     THE ROUTINES FROM HERE TO THE ROUTINE
*     FINDPT READ AND STORE THE SLICES FOR THE
*     BASIN WHILE EDITING AND COMPUTING THE
*     COEFFICIENTS FOR THE LINE SEGMENTS.
*     EXIT IS TAKEN WHEN THE FIRST RECORD
*     OF THE NEXT SLICE IS READ AND A BRANCH
*     IS TAKEN TO THE ROUTINE, FINDPT.
*     THE RETURN IS TO READV1 SINCE THE FIRST
*     RECORD OF THE NEXT SLICE HAS NOT BEEN
*     STORED IN THE TABLE.
 READV.
      READ CARD-FILE INTO VCARD AT END GO TO STOP5.
      ADD 1 TO INCRDS.
 READV1.
      IF FRSW = "F", PERFORM CLRVSS VARYING LA1 FROM 1 BY
          1 UNTIL LA1 = 51,
          MOVE "Y" TO FRSW, SET LA1 TO 1,
          MOVE LATUDE-L OF VCARD TO NHOLD.
      IF LATUDE-L OF VCARD < NHOLD, GO TO FINDPT.
 STRVCD.
      IF LATUDE-L OF VCARD > NHOLD, GO TO STOP8.
      EXAMINE LATUDE-L OF VCARD REPLACING ALL " " BY 0.
      MOVE LATUDE-L OF VCARD TO LATUDE-1 (LA1).
      EXAMINE LNGTUD-L OF VCARD REPLACING ALL " " BY 0.
      MOVE LNGTUD-L OF VCARD TO LNGTUD-1 (LA1).
      EXAMINE LATUDE-H OF VCARD REPLACING ALL " " BY 0.
      MOVE LATUDE-H OF VCARD TO LATUDE-2 (LA1).
      EXAMINE LNGTUD-H OF VCARD REPLACING ALL " " BY 0.
      MOVE LNGTUD-H OF VCARD TO LNGTUD-2 (LA1).
      EXAMINE DECLAT-L OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLAT-L OF VCARD TO DECLAT-1 (LA1).
      EXAMINE DECLNG-L OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLNG-L OF VCARD TO DECLNG-1 (LA1).
      EXAMINE DECLAT-H OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLAT-H OF VCARD TO DECLAT-2 (LA1).
      EXAMINE DECLNG-H OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLNG-H OF VCARD TO DECLNG-2 (LA1).
      EXAMINE ACOEFF OF VCARD REPLACING ALL " " BY 0.
      MOVE ACOEFF OF VCARD TO ACOEFS (LA1).
      IF ASIGN OF VCARD = "-", COMPUTE ACOEFS (LA1) =
           ACOEFS (LA1) - 2 * ACOEFS (LA1),
        ELSE MOVE "+" TO ASIGN OF VCARD.
      EXAMINE BCOEFF OF VCARD REPLACING ALL " " BY 0.
      MOVE BCOEFF OF VCARD TO BCOEFS (LA1).
      IF BSIGN OF VCARD = "-", COMPUTE BCOEFS (LA1) =
           BCOEFS (LA1) - 2 * BCOEFS (LA1),
        ELSE MOVE "+" TO BSIGN OF VCARD.
      EXAMINE KNSTNT OF VCARD REPLACING ALL " " BY 0.
      MOVE KNSTNT OF VCARD TO KNSTNS (LA1).
```

```
      IF CSIGN OF VCARD = "-", COMPUTE KNSTNS (LA1) =
           KNSTNS (LA1) - 2 * KNSTNS (LA1),
        ELSE MOVE "+" TO CSIGN OF VCARD.
      SET LN2 TO LA1.
      SET LA1 UP BY 1.
      GO TO READV.
*NOTE-E.
*     THIS IS THE TAPE READ ROUTINE THAT CONTROLS THE MAIN
*      PROGRAM.  THE KEY ROUTINES ARE READTAPE, BRNALL,
*     CHNG01, CHNG02, TEST2.
*     THIS PROCEDURE TESTS THE FIRST RECORD TAPE
*     SWITCH FOR THE SITE BEING OUTSIDE THE BASIN
*     AND CONTINUES TO READ THE MASTER FILE UNTIL
*     A SITE WITHIN THE BASIN IS FOUND.
*     RECORDS ARE PROCESSED FOR THAT SITE UNTIL
*     A CHANGE OF SITE KEYID IS DETECTED.  A
*     BRANCH IS MADE TO CHNG01 WHERE THE OUTPUT
*     FOR THE PREVIOUS IS MADE IF APPLICABLE.
*     THEN IN CHNG02 THE LAST SITE RECORDS ARE
*     CLEARED AND THE NEW SITE IS SET UP FOR
*     SEARCH AND PROCESSING.
 FINDPT.
      MOVE "F" TO FRSW.
      IF (SLSW = "X") MOVE "Y" TO SLSW, GO TO TEST2.
      GO TO READTAPE.
 READTAPE.
      READ FILEB AT END GO TO CLOSEA.
      ADD 1 TO TFILEB, NFLEIA.
      IF KEYID OF OLDMAS NOT = KEYIDL, GO TO CHNG01.
*NOTE-WELL.
*     THE BRANCH INSTRUCTIONS IN THE PROCEDURE, BRNALL,
*     ARE ORDERED BY THEIR RELATIVE FREQUENCY IN THE
*     WRDM MASTER FILE.  REARRANGEMENT WOULD SLOW
*     THE OPERATION.
 BRNALL.
      IF FRTSW = "X", GO TO READTAPE.
      IF RNUM OF OLDMAS = 234, GO TO PTR234.
      IF RNUM OF OLDMAS = 000, GO TO PTR000.
      IF RNUM OF OLDMAS = 001, GO TO PTR001.
      IF RNUM OF OLDMAS = 002, GO TO PTR002.
      IF RNUM OF OLDMAS = 003, GO TO PTR003.
      IF RNUM OF OLDMAS = 058, GO TO PTR058.
      IF RNUM OF OLDMAS = 076, GO TO PTR076.
      IF RNUM OF OLDMAS = 198, GO TO PTR198.
      IF RNUM OF OLDMAS = 042, GO TO PTR042.
      IF RNUM OF OLDMAS = 082, GO TO PTR082.
      IF RNUM OF OLDMAS = 002, GO TO PTR002.
      IF RNUM OF OLDMAS = 001, GO TO PTR001.
      IF RNUM OF OLDMAS = 158, GO TO PTR158.
      IF RNUM OF OLDMAS = 192, GO TO PTR192.
      IF RNUM OF OLDMAS = 146, GO TO PTR146.
      IF RNUM OF OLDMAS = 186, GO TO PTR186.
      IF RNUM OF OLDMAS = 090, GO TO PTR090.
      IF RNUM OF OLDMAS = 114, GO TO PTR114.
      IF RNUM OF OLDMAS = 121, GO TO PTR121.
```

```
     IF RNUM OF OLDMAS = 189, GO TO PTR189.
     IF RNUM OF OLDMAS = 094, GO TO PTR094.
     IF RNUM OF OLDMAS = 183, GO TO PTR183.
     IF RNUM OF OLDMAS = 072, GO TO PTR072.
     IF RNUM OF OLDMAS = 047, GO TO PTR047.
     IF RNUM OF OLDMAS = 134, GO TO PTR134.
     IF RNUM OF OLDMAS = 171, GO TO PTR171.
     IF RNUM OF OLDMAS = 180, GO TO PTR180.
     IF RNUM OF OLDMAS = 127, GO TO PTR127.
     IF RNUM OF OLDMAS = 055, GO TO PTR055.
     IF RNUM OF OLDMAS = 098, GO TO PTR098.
     IF RNUM OF OLDMAS = 105, GO TO PTR105.
     IF RNUM OF OLDMAS = 164, GO TO PTR164.
     IF RNUM OF OLDMAS = 203, GO TO PTR203.
     IF RNUM OF OLDMAS = 208, GO TO PTR208.
     IF RNUM OF OLDMAS = 212, GO TO PTR212.
     IF RNUM OF OLDMAS = 219, GO TO PTR219.
     IF RNUM OF OLDMAS = 250, GO TO PTR250.
     IF RNUM OF OLDMAS = 277, GO TO PTR277.
     GO TO STOP9.
 CHNG01.
     IF FRTSW = "Y", PERFORM PRTSNL.
*  THE FOLLOWING CONDITION TO BE USED FOR WITHDRAWING SPECIFIC LOG
*     IF FRTSW = "Y" AND LOGSW =  "Y", PERFORM PRTSNL.
 CHNG02.
     MOVE KEYID OF OLDMAS TO KEYIDL.
     MOVE SPACES TO SITELN.
     MOVE SPACES TO RECOUT.
     MOVE "X" TO FRTSW, HSW, NSW, LSW.
     MOVE KEYLAT OF OLDMAS TO NLATMS.
     MOVE KEYLNG OF OLDMAS TO NLNGMS.
     GO TO TEST2.
*NOTE-P.
*     THE FOLLOWING ROUTINES ARE IN PLACE TO
*     ALLOW PROCESSING OF ANY RECORD IN THE
*     (WRDM) WATER RESOURCES MASTER TAPE FILE
*     DATA MAY BE EXTRACTED AND STORED IN THE
*     LAST 41 POSITIONS OF "SITELN" AND "RECOUT"
*     FOR STORING ON YOUR CYBER DISK FILE AND
*     PRINTING ON THE LISTING.
*
*     ADDITIONAL OUT PUT MAY BE HANDLED BY
*     SETTING UP ADDITIONAL OUTPUT RECORDS
*     AND INSERTING THE PROPER INSTRUCTIONS
*     TO ADD TO THE OUTPUT
 PTR000.
     MOVE "Y" TO HSW.
     MOVE "N" TO LOGSW.
     MOVE OLDMAS TO RTW000.
 PRMOVE.
     EXAMINE WDEPTH OF RTW000 REPLACING ALL " " BY 0.
     EXAMINE HDEPTH OF RTW000 REPLACING ALL " " BY 0.
     EXAMINE WTRLVL OF RTW000 REPLACING ALL " " BY 0.
     EXAMINE PPDISH OF RTW146 REPLACING ALL " " BY 0.
     EXAMINE ALTUDE OF RTW000 REPLACING ALL " " BY 0.
```

```
    IF WDEPTH OF RTW000 = 0, MOVE HDEPTH OF RTW000 TO DDEPTH OF
            SITELN, DDEPTH OF RECOUT
    ELSE
    MOVE WDEPTH OF RTW000 TO DDEPTH OF SITELN, DDEPTH OF RECOUT.
    MOVE WDSRCE OF RTW000 TO IDEPTH OF SITELN, IDEPTH OF RECOUT.
    MOVE WTRLVL OF RTW000 TO WTRLVL2, WTRLVL3.
    MOVE WLYEAR OF RTW000 TO WLYEAR OF SITELN.
    MOVE WLDAT TO WLDTMP.
    MOVE WLMETH OF RTW000 TO WLMETH2, WLMETH3.
    MOVE CORR RTW000 TO SITELN.
    MOVE CORR RTW000 TO RECOUT.
    GO TO READTAPE.
PTR001.
    COMPUTE CNTQW1 = CNTQW1 + 1.
    IF (HSW = "Y" AND WDEPTH OF RTW000 NOT = 0) GO TO READTAPE.
    MOVE "H" TO HSW.
    MOVE OLDMAS TO RTQ001.
    EXAMINE WDEPTH OF RTQ001 REPLACING ALL " " BY 0.
    IF HSW = "Y", MOVE WDEPTH OF RTQ001 TO DDEPTH
       OF SITELN, DDEPTH OF RECOUT, GO TO READTAPE.
    GO TO PRMOVE.
PTR002.
    MOVE "N" TO NSW.
    ADD 1 TO COUNTN, CNTQW2.
    IF HSW = "Y", GO TO READTAPE.
    MOVE OLDMAS TO RTQ002.
    MOVE LCWLID OF RTQ002 TO LCWLID OF SITELN.
    MOVE LCWLID OF RTQ002 TO LCWLID OF RECOUT.
    GO TO READTAPE.
PTR003.
    COMPUTE CNTQW3 = CNTQW3 + 1.
    GO TO READTAPE.
PTR042.
    GO TO READTAPE.
PTR047.
    GO TO READTAPE.
PTR055.
    GO TO READTAPE.
PTR058.
    MOVE OLDMAS TO RTW058.
    IF (ENTNUM > ENTEMP),
      MOVE ENTNUM TO ENTEMP,
    MOVE FINISH OF RTW058 TO FINISH OF RECOUT, FINISH OF SITELN.
    GO TO READTAPE.
PTR072.
    GO TO READTAPE.
PTR076.
    MOVE OLDMAS TO RTW076.
    EXAMINE RTW076 REPLACING ALL " " BY 0.
    MOVE CSGDIAM TO CDIAM OF RECOUT, CDIAM OF SITELN.
    GO TO READTAPE.
```

```
PTR082.
    MOVE OLDMAS TO RTW082.
    EXAMINE OPNTOP OF RTW082 REPLACING ALL " " BY 0.
    MOVE OPNTOP OF RTW082 TO OPNTOP OF RECOUT, OPNTOP OF SITELN
    MOVE OPNTYP OF RTW082 TO OPNTYP OF RECOUT, OPNTYP OF SITELN.
    GO TO READTAPE.
PTR090.
    GO TO READTAPE.
PTR094.
    GO TO READTAPE.
PTR098.
    GO TO READTAPE.
PTR105.
    GO TO READTAPE.
PTR114.
    GO TO READTAPE.
PTR121.
    GO TO READTAPE.
PTR127.
    GO TO READTAPE.
PTR134.
    GO TO READTAPE.
PTR146.
    MOVE OLDMAS TO RTW146.
    EXAMINE PPYEAR OF RTW146 REPLACING ALL " " BY 0.
    EXAMINE PPDISH OF RTW146 REPLACING ALL " " BY 0.
    MOVE PPYEAR OF RTW146 TO PPYEAR OF RECOUT.
    IF PPYEAR OF RTW146 = 0000 MOVE SPACES TO PPYEAR OF SITELN
    ELSE MOVE PPYEAR OF RTW146 TO PPYEAR OF SITELN.
    MOVE PPDISH OF RTW146 TO PPDISH OF RECOUT.
    MOVE PPDISH OF RTW146 TO PPDISH OF SITELN.
    GO TO READTAPE.
PTR158.
    GO TO READTAPE.
PTR164.
    GO TO READTAPE.
PTR171.
    GO TO READTAPE.
PTR180.
    GO TO READTAPE.
PTR183.
    GO TO READTAPE.
PTR186.
    GO TO READTAPE.
PTR189.
    GO TO READTAPE.
PTR192.
    GO TO READTAPE.
PTR198.
    MOVE OLDMAS TO RTW198.
    IF LSW = "X", SET LX TO 1,
        MOVE "Y" TO LSW,
    ELSE SET LX UP BY 1.
    IF LX > 8, GO TO READTAPE.
    MOVE TYPELG OF RTW198 TO LOGTYP (LX).
```

```
*  USE THE FOLLOWING STATEMENT TO PULL SPECIFIC LOG TYPES
*     IF (TYPELG OF RTW198 = "G" OR "I" OR "J" OR "N" OR "Q" OR
*         "S" OR "U") , MOVE "Y" TO LOGSW.
      GO TO READTAPE.
 PTR203.
      GO TO READTAPE.
 PTR208.
      GO TO READTAPE.
 PTR212.
      GO TO READTAPE.
 PTR219.
      GO TO READTAPE.
 PTR234.
      MOVE OLDMAS TO RTW234.
      MOVE WTRLVL1 TO WLTEST.
      IF (CNRECN > 421 OR CNRECN < 425) MOVE 000000 TO WTRLVL1,
          WTRLVL2, WTRLVL3.
      IF NOT (WLTEST > "000000" OR WLTEST ="000000" OR
          WLTEST < "999999") MOVE ZEROS TO WTRLVL1.
      EXAMINE WLDATE  REPLACING ALL " " BY 0.
      EXAMINE WTRLVL1 REPLACING ALL " " BY 0.
      EXAMINE WTRLVL1 REPLACING ALL "." BY 0.
      COMPUTE WTLCNT = WTLCNT + 1.
      IF (WLDATE > WLDTMP),
          MOVE WLDATE TO WLDTMP,
          MOVE WLYEAR OF RTW234 TO WLYEAR OF SITELN,
                      WLYEAR OF RECOUT,
          IF (WTRLVL1 IS NOT NUMERIC) MOVE ZEROS TO WTRLVL1.
          MOVE WTRLVL1 TO WTRLVL2, WTRLVL3,
          MOVE WLMETH1 TO WLMETH2, WLMETH3.
      GO TO READTAPE.
 PTR250.
      GO TO READTAPE.
 PTR277.
      GO TO READTAPE.
*NOTE-T.
*     THIS ROUTINE TESTS THE LINE SEGMENT EQUATIONS
*     TO DETERMINE IF THE SITE IS WITHIN THE BASIN.
*     THE FRTSW SWITCH IS SET TO "Y" IF THE SITE
*     IS WITHIN THE BASIN.  EXIT IS TAKEN TO ENOFLE
*     IF THE LATITUDE OF THE SITE BEING TESTED IS
*      GREATER THAN THE HIGH LATITUDE OF THE
*     CURRENT SLICE.   EXIT IS TO READTAPE IF THE
*     SITE IS OUTSIDE THE RECTANGLE THAT BOUNDS
*     THE CURRENT SLICE.    IF THE SITE IS WITHIN
*     THE RECTANGLE BUT OUTSIDE THE BASIN EXIT
*     IS TAKEN TO READTAPE.  ONLY IF THE SITE IS
*     WITHIN THE BASIN IS THE BRANCH TO BRNALL
*     TAKEN.
 TEST2.
      IF NLATMS < LATUDE-1(1), GO TO ENDFLE.
      IF NLATMS > LATUDE-2(1), MOVE EMSG1 TO ENOTE, PERFORM DMPSIT,
         GO TO READTAPE.
      IF NLNGMS < LOLNG OF BCARD, MOVE EMSG2 TO ENOTE,
            PERFORM DMPSIT, GO TO READTAPE.
```

```
    IF NLNGMS > HILNG OF BCARD,  MOVE EMSG3 TO ENOTE,
        PERFORM DMPSIT, GO TO READTAPE.
    MOVE NLATMS TO LATDMS.
    MOVE LATD TO FLATD.
    MOVE LATM TO FLATM.
    MOVE LATS TO FLATS.
    COMPUTE FDCLAT = FLATD + FLATM / 60 + FLATS / 3600.
    MOVE NLNGMS TO LNGDMS.
    MOVE LNGD TO FLNGD.
    MOVE LNGM TO FLNGM.
    MOVE LNGS TO FLNGS.
    COMPUTE FDCLNG = FLNGD + FLNGM / 60 + FLNGS / 3600.
    COMPUTE FDCLNG = FDCLNG - 2 * FDCLNG.
    SET DA1 TO 1.
TEST21.
    COMPUTE KTEST = ACOEFS (DA1) * FDCLNG + BCOEFS (DA1)
        * FDCLAT.
    COMPUTE LTEST = ACOEFS (DA1 + 1) * FDCLNG
        + BCOEFS (DA1 + 1 ) * FDCLAT.
    IF (KTEST < KNSTNS (DA1) OR LTEST > KNSTNS (DA1 + 1))
          GO TO NEXTB, ELSE GO TO SITEIN.
NEXTB.
    IF (DA1 + 1 = LN2), GO TO READTAPE,
        ELSE SET DA1 UP BY 2, GO TO TEST21.
SITEIN.
    MOVE "Y" TO FRTSW.
    GO TO BRNALL.
ENDFLE.
    IF LATUDE-L OF VCARD = 000000, GO TO CLOSEA.
    IF LATUDE-L OF VCARD < LOLAT OF BCARD,
        MOVE VCARD TO BCARD, GO TO READB1,
        ELSE MOVE "F" TO FRSW, MOVE "X" TO SLSW, GO TO READV1.
PRTSNL.
    IF LNCNT > 60, PERFORM HDPRNT,
        ELSE ADD 1 TO LNCNT.
    MOVE IDEPTH OF SITELN TO IDEPTH OF RECOUT.
    MOVE TYPLOG TO INLOGS OF SITELN, INLOGS OF RECOUT.
    MOVE WTLCNT TO WLCNT2,WLCNT3.
    MOVE CNTQW1 TO CNTQ1O, CNTQ1S.
    MOVE CNTQW2 TO CNTQ2O, CNTQ2S.
    MOVE CNTQW3 TO CNTQ3O, CNTQ3S.
    ADD 1 TO CNRECN, NDETLN.
    MOVE CNRECN TO CNRECP OF SITELN.
    MOVE CNRECN TO CNRECR OF RECOUT.
    WRITE RECOUT.
    WRITE LIST-LINE FROM SITELN.
    ADD 1 TO BLKCNT.
    IF BLKCNT = 5, MOVE 0 TO BLKCNT, MOVE SPACES TO
        LIST-LINE, WRITE LIST-LINE, ADD 1 TO LNCNT.
    MOVE SPACES TO TYPLOG, SITELN, RECOUT, WLDTMP.
    MOVE ZEROS TO ENTEMP, WTLCNT, CNTQW1, CNTQW2, CNTQW3,CNTQ1O,
         CNTQ1S, CNTQ2O, CNTQ2S, CNTQ3O, CNTQ3S, WLCNT2, WLCNT3,
         RTW234,RTW058,RTW076,RTW082.
     MOVE "N" TO LOGSW.
```

```
HDPRNT.
     ADD 1 TO PGCNT.
     MOVE PGCNT TO PNUM OF HDR1.
     MOVE 0 TO BLKCNT.
     MOVE 8 TO LNCNT.
     WRITE LIST-LINE FROM HDR1.
     WRITE LIST-LINE FROM HDR2.
     WRITE LIST-LINE FROM HDR5.
     WRITE LIST-LINE FROM HDR3.
     WRITE LIST-LINE FROM HDR4.
DBUGGER.
     WRITE ERREC FROM STRLN AFTER ADVANCING 2 LINES.
     MOVE KLD TO LTD, MOVE KLM TO LTM, MOVE KLS TO LTS, MOVE KLGM
         TO LGM, MOVE KLGD TO LGD, MOVE KLGS TO LGS.
     WRITE ERREC FROM DBUG1 AFTER ADVANCING 2 LINES.
     WRITE ERREC FROM DBUG2 AFTER ADVANCING 2 LINES.
     MOVE NLATMS TO ELATMS, MOVE LATUDE-1(1) TO ELATUDE-1, MOVE
         LATUDE-2(1) TO ELATUDE-2, MOVE NLNGMS TO ELNGMS, MOVE
         LOLNG TO ELOLNG, MOVE HILNG TO EHILNG.
     WRITE ERREC FROM DBUG3 AFTER ADVANCING 2 LINES.
     MOVE LATUDE-L OF VCARD TO ELAT-L , MOVE LNGTUD-L OF VCARD
          TO ELNG-L, MOVE LATUDE-H OF VCARD TO ELAT-H, MOVE
         LNGTUD-H OF VCARD TO ELNG-H, MOVE DECLAT-L OF VCARD
              TO EDL-L, MOVE LS1 TO ELS1, MOVE DECLNG-L OF VCARD
         TO EDLG-L.
     WRITE ERREC FROM DBUG4 AFTER ADVANCING 2 LINES.
     WRITE ERREC FROM DBUG5 AFTER ADVANCING 2 LINES.
     MOVE ASIGN OF VCARD TO EASIN, MOVE ACOEFF OF VCARD TO EACOF,
         MOVE BSIGN OF VCARD TO  EBSIN, MOVE BCOEFF OF VCARD TO
         EBCOF, MOVE DECLAT-H OF VCARD TO EDL-H, MOVE
         LS2 TO ELS2, MOVE DECLNG-H OF VCARD TO EDLG-H,
              MOVE CSIGN OF VCARD  TO ECSIN, MOVE KNSTNT OF VCARD
              TO EKNSTNT.
      WRITE ERREC FROM DBUG6 AFTER ADVANCING 2 LINES.
      WRITE ERREC FROM DBUG7 AFTER ADVANCING 2 LINES.
      MOVE LATDMS TO ELTDMS, MOVE FDCLAT TO EFDCLT, MOVE LNGDMS
          TO ELGDMS, MOVE FDCLNG TO EFDCLG.
      WRITE ERREC FROM DBUG8 AFTER ADVANCING 2 LINES.
      MOVE LATD TO ELATD, MOVE LATM TO ELATM, MOVE LATS TO ELATS,
          MOVE LNGD TO ELNGD, MOVE LNGM TO ELNGM, MOVE LNGS TO
          ELNGS.
      WRITE ERREC FROM DBUG9 AFTER ADVANCING 2 LINES.
      MOVE FLATD TO EFLATD, MOVE FLATM TO EFLATM, MOVE FLATS TO
          EFLATS, MOVE FLNGD TO EFLNGD, MOVE FLNGM TO EFLNGM,
           MOVE FLNGS TO EFLNGS.
      IF (FRSW = "F" ) SET DA1 TO 1.
      IF DA1 < 1, SET DA1 TO 1.
      WRITE ERREC FROM DBUG10 AFTER ADVANCING 2 LINES.
      MOVE KTEST TO EKTEST, MOVE LTEST TO ELTEST,
          MOVE KNSTNS (DA1) TO KNSTNS (DA1),
          COMPUTE DA3 = DA1 + 1, MOVE KNSTNS (DA3) TO EKNSTNS-2,
          MOVE LN1 TO ELN1, MOVE LN2 TO ELN2.
       WRITE ERREC FROM DBUG11.
```

```
       MOVE DA1 TO EDA1, MOVE ACOEFS (DA1) TO ECOF-DA1, MOVE
          ACOEFS (DA3) TO ECOF-DA2, MOVE BCOEFS (DA1) TO
          EBCOF-DA1, MOVE BCOEFS (DA3) TO EBCOF-DA2.
       WRITE ERREC FROM DBUG12.
       WRITE ERREC FROM DBUG13.
 CLOSEA.
     IF FRTSW = "Y", PERFORM PRTSNL.
*     IF FRTSW = "Y" AND LOGSW =  "Y", PERFORM PRTSNL.
     CLOSE FILEB.
     GO TO STOP3.
 CLRVSS.
     MOVE 0 TO LATUDE-1 (LA1).
     MOVE 0 TO LNGTUD-1 (LA1).
     MOVE 0 TO LATUDE-2 (LA1).
     MOVE 0 TO LNGTUD-2 (LA1).
     MOVE 0 TO DECLAT-1 (LA1).
     MOVE 0 TO DECLNG-1 (LA1).
     MOVE 0 TO DECLAT-2 (LA1).
     MOVE 0 TO DECLNG-2 (LA1).
     MOVE 0 TO ACOEFS (LA1).
     MOVE 0 TO BCOEFS (LA1).
     MOVE 0 TO KNSTNS (LA1).
  PRNTSL.
     MOVE LATUDE-1 (LN1) TO LATUDE-L OF SLN.
     MOVE  LNGTUD-1 (LN1) TO LNGTUD-L OF SLN.
     MOVE LATUDE-2 (LN1) TO LATUDE-H OF SLN.
     MOVE LNGTUD-2 (LN1) TO LNGTUD-H OF SLN.
     MOVE DECLAT-1 (LN1) TO DECLAT-L OF SLN.
     MOVE DECLNG-1 (LN1) TO DECLNG-L OF SLN.
     MOVE DECLAT-2 (LN1) TO DECLAT-H OF SLN.
     MOVE DECLNG-2 (LN1) TO DECLNG-H OF SLN.
     MOVE ACOEFS (LN1) TO ACOEFF OF SLN.
     MOVE BCOEFS (LN1) TO BCOEFF OF SLN.
     MOVE KNSTNS (LN1) TO KNSTNT OF SLN.
     IF ACOEFS (LN1) > 0, MOVE "+" TO ASIGN OF SLN
       ELSE MOVE "-" TO ASIGN OF SLN.
     IF BCOEFS (LN1) > 0, MOVE "+" TO BSIGN OF SLN
       ELSE MOVE "-" TO BSIGN OF SLN.
     IF KNSTNS (LN1) > 0, MOVE "+" TO CSIGN OF SLN
       ELSE MOVE "-" TO CSIGN OF SLN.
     IF LNCNT > 60 PERFORM SHPRT, ELSE ADD 1 TO LNCNT.
     WRITE LIST-LINE FROM SLN.
 SHPRT.
     ADD 1 TO PGCNT.
     MOVE PGCNT TO PNUM OF HDR1.
     WRITE LIST-LINE FROM HDR1.
     WRITE LIST-LINE FROM HDR2.
      MOVE 4 TO LNCNT.
 RITDICT.
     ADD 1 TO PGCNT.
     MOVE PGCNT TO PNUM OF HDR1.
     WRITE LIST-LINE FROM HDR1.
     WRITE LIST-LINE FROM HDR2.
     WRITE LIST-LINE FROM DICT-HDR1.
     WRITE LIST-LINE FROM DICT-HDR2.
```

```
    WRITE LIST-LINE FROM DICT-HDR3.
    WRITE LIST-LINE FROM DICT-LN1.
    WRITE LIST-LINE FROM DICT-LN2.
    WRITE LIST-LINE FROM DICT-LN3.
    WRITE LIST-LINE FROM DICT-LN4.
    WRITE LIST-LINE FROM DICT-LN5.
    WRITE LIST-LINE FROM DICT-LN6.
    WRITE LIST-LINE FROM DICT-LN7.
    WRITE LIST-LINE FROM DICT-LN8.
    WRITE LIST-LINE FROM DICT-LN9.
    WRITE LIST-LINE FROM DICT-LN10.
    WRITE LIST-LINE FROM DICT-LN11.
    WRITE LIST-LINE FROM DICT-LN12.
    WRITE LIST-LINE FROM DICT-LN13
    WRITE LIST-LINE FROM DICT-LN14.
    WRITE LIST-LINE FROM DICT-LN15.
    WRITE LIST-LINE FROM DICT-LN16.
    WRITE LIST-LINE FROM DICT-LN17.
    WRITE LIST-LINE FROM DICT-LN18.
    WRITE LIST-LINE FROM DICT-LN19.
    WRITE LIST-LINE FROM DICT-LN20.
    WRITE LIST-LINE FROM DICT-LN21.
    WRITE LIST-LINE FROM DICT-LN22.
DMPSIT.
    IF EXTSW = "Y", MOVE PCNLX TO ETITLE, MOVE "X" TO EXTSW,
          PERFORM ERHEAD.
     COMPUTE ESEQ = ESEQ + 1
    MOVE ESEQ TO ERSEQ.
     MOVE TFILE8 TO INSEQ.
    MOVE KEYID OF OLDMAS TO ERIDNT.
    IF RNUM OF OLDMAS = 000, MOVE OLDMAS TO RTW000,
          MOVE LCWLID OF RTw000 TO ELCLID.
    IF RNUM OF OLDMAS = 001, MOVE OLDMAS TO RTQ001,
           MOVE SPACES TO ELCLID.
    IF RNUM OF OLDMAS = 002, MOVE OLDMAS TO RTQ002,
           MOVE LCWLID OF RTQ002 TO ELCLID.
    IF RNUM OF OLDMAS = 234, MOVE SPACES TO ELCLID.
    WRITE ERREC FROM ERECLN.
    MOVE SPACES TO ERECLN, ERREC.
    ADD 1 TO ELNCNT.
    IF ELNCNT > 60, ADD 1 TO EPGCNT, MOVE EPGCNT TO EPNUM,
        PERFORM ERHEAD.
ERHEAD.
    MOVE ZEROS TO ELNCNT.
    WRITE ERREC FROM ERHD1.
    WRITE ERREC FROM ERHD2.
    WRITE ERREC FROM ERHD3.
    WRITE ERREC FROM ERHD4.
    MOVE SPACES TO ERREC.
    WRITE ERREC.
    WRITE ERREC.
    ADD 5 TO ELNCNT.
```

```
STOP1.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "CONTROL INPUT FILE EMPTY".
    STOP RUN.
STOP2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.
STOP3.
    PERFORM RITDICT.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY TFILE8.
    MOVE INCRDS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FADATA.
    MOVE NDETLN TO DRLNS.
    MOVE NFLEOA TO FOUT.
    WRITE LIST-LINE FROM JBCNLX.
    DISPLAY "NORMAL JOB TERMINATION".
    DISPLAY " COUNT OF BLANKS ", CN8LNK.
    CLOSE CARD-FILE, LIST-FILE, ERRFLE.
    STOP RUN.
STOP4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP4 SOURCE ID RECORDS MISSING".
    STOP RUN.
STOP5.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP5 DATA INPUT FILE END INCORRECT".
    DISPLAY "EOF ACCURRED AT READ8 OR READV".
    STOP RUN.
STOP6.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP6 SECOND CARD OF PAIR MISSING".
    STOP RUN.
STOP7.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP7 FIRST RECORD NOT 000 OR 001".
    DISPLAY KEYIDL, " 7 ERROR 7 ", KEYID OF OLDMAS.
    STOP RUN.
STOP8.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP8 FILE OUT OF SEQUENCE".
    DISPLAY NHOLD, "LATUDE-L = ", LATUDE-L OF VCARD.
    DISPLAY   "LATUDE-H", LATUDE-H OF VCARD.
    DISPLAY KEYIDL, " 8 ERROR 8 ", KEYID OF OLDMAS.
    STOP RUN.
STOP9.
    DISPLAY "STOP9 RECORD TYPE NOT FOUND".
    DISPLAY OLDMAS.
    STOP RUN.
```

```
IDENTIFICATION DIVISION.
PROGRAM-ID. BASN04.
REMARKS.  BASN04 IS USED TO OUTPUT A WATER LEVEL HISTORY REPORT
          FOR A PARTICULAR BASIN FROM THE SWAB/RASA MASTER FILE.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER.  CDC CYBER 175.
OBJECT-COMPUTER. CDC CYBER 175.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT CARD-FILE ASSIGN TO "INPUT".
    SELECT LIST-FILE ASSIGN TO "OUTPUT".
    SELECT FILEB ASSIGN TO MGQWIN
    RESERVE 4 ALTERNATE AREAS
    ORGANIZATION IS SEQUENTIAL.
    SELECT PLTFLE ASSIGN TO PLTDSK
    ORGANIZATION IS SEQUENTIAL.
DATA DIVISION.
FILE SECTION.
FD  CARD-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS CNL-CARD.
01  CNL-CARD.
    03 ID-CNL PIC IS XXX.
    03 ID-NUM PIC 999.
    03 ID-DES PIC X(74).
FD  LIST-FILE
    LABEL RECORD OMITTED
    DATA RECORD IS LIST-LINE.
01  LIST-LINE .
    03  CC PIC X.
    03  LLPRT.
        05  FILLER PIC X.
        05  P0 PIC X(10).
        05  FILLER PIC XXX.
        05  P1 PIC X(10).
        05  FILLER PIC XXX.
        05  P2 PIC X(10).
        05  FILLER PIC XXX.
        05  P3 PIC X(10).
        05  FILLER PIC XXX.
        05  P4 PIC X(10).
        05  FILLER PIC XXX.
        05  P5 PIC X(10).
        05  FILLER PIC XXX.
        05  P6 PIC X(10).
        05  FILLER PIC XXX.
        05  P7 PIC X(10).
        05  FILLER PIC XXX.
        05  P8 PIC X(10).
        05  FILLER PIC XXX.
        05  P9 PIC X(10).
    03  PIND PIC X(4).
```

```
FD  FILEB
    DATA RECORD IS OLDMAS
    RECORD CONTAINS 23 TO 327 CHARACTERS
    RECORDING MODE IS BINARY
    LABEL RECORDS ARE STANDARD.
01  OLDMAS.
    03  KEYID.
        05 KEYLAT.
           07  KLD  PIC 99.
           07  KLM  PIC 99.
           07  KLS  PIC 99.
        05 KEYLNG.
           07  KLGD  PIC 999.
           07  KLGM  PIC 99.
           07  KLGS  PIC 99.
        05 KEYDUP PIC 99.
    03  RNUM PIC IS 999.
    03  LNUM PIC IS 999.
    03  SESAFE PIC X.
    03  RESTRX PIC X OCCURS 1 TO 305 TIMES,
            DEPENDING ON LNUM OF OLDMAS.
FD  PLTFLE
    DATA RECORD IS RECOUT
    LABEL RECORDS ARE OMITTED.
01 RECOUT.
   03  CCR PIC X.
    03  CNRECR PIC 9(4).
    03  XSP PIC X(4).
    03  INDO PIC 99.
    03  KEYID.
        05 KEYLAT PIC 9(6).
        05 KEYLNG PIC 9(7).
        05 KEYDUP PIC 99.
    03  LCWLID.
        05  LCSITE PIC X(14).
        05  LCREST PIC X(6).
    03  ALTUDE PIC 9(5).
    03  RWDEPTH PIC 9(5).
    03  RWTLEV  PIC 9(5)V99.
    03  RWDATE  PIC 9(8).
    03  RWRST  PIC X(5).

WORKING-STORAGE SECTION.
77  ALFAA  PIC X VALUE "A".
77  ALFABLNK PIC X VALUE " ".
77  ALFAY  PIC X VALUE "Y".
77  ALFAX  PIC X VALUE "X".
77  ALFAZ  PIC X VALUE "Z".
77  CNBLNK PIC 9(6) VALUE 0.
77  CNTSL  PIC 9(5) VALUE 0.
77  COUNTA PICTURE 9(3) VALUE 0.
77  COUNT2 PIC 9(6) VALUE 0.
77  COUNT2G PIC 9(6) VALUE 0.
77  CNROUT  PIC 9(5) VALUE 0.
```

```
77  DA1  PIC 9(4).
77  ELNCNT  PIC 999 VALUE 000.
77  ENTEMP  PIC 999 VALUE ZEROS.
77  EPGCNT  PIC 999 VALUE 001.
77  ERRCNT PIC 9(6) VALUE 0.
77  ESEQ PIC 9(5) VALUE 0.
77  EXTSW  PIC X VALUE "Y".
77  FRSW PIC X VALUE "F".
77  FRTSW PIC X VALUE "F".
77  FDCLAT USAGE IS COMP-2.
77  FLATD USAGE IS COMP-2.
77  FLATM USAGE IS COMP-2.
77  FLATS USAGE IS COMP-2.
77  FDCLNG USAGE IS COMP-2.
77  FLNGD USAGE IS COMP-2.
77  FLNGM USAGE IS COMP-2.
77  FLNGS USAGE IS COMP-2.
77  HSW PIC X VALUE "X".
77  INCRDS PIC 9(6) VALUE 0.
77  KEYIDL PIC 9(15) VALUE 0.
77  KTEST USAGE IS COMP-2.
77  LA1  PIC 9(4).
77  LDSW  PIC X VALUE "Y".
77  LNCNT PIC 99 VALUE 70.
77  LN1  PIC 9(4).
77  LN2  PIC 9(4).
77  LSW PIC X VALUE "X".
77  LTEST USAGE IS COMP-2.
77  NDATIN PIC 9(6) VALUE 0.
77  NDETLN PIC 9(6) VALUE 0.
77  NFLEIA PIC 9(6) VALUE 0.
77  NFLEOA PIC 9(6) VALUE 0.
77  NHOLD PIC 9(6).
77  NLATMS PIC 9(6).
77  NLNGMS PIC 9(7).
77  NMM  PIC 999.
77  NSW PIC X VALUE "X".
77  PGCNT PIC 999 VALUE 0.
77  PRTSW PIC X VALUE "Z".
77  RECNT PIC 9(4) VALUE 0.
77  SLSW  PIC X VALUE "Y".
77  SW0  PIC X VALUE "Y".
77  SW1  PIC X VALUE "Y".
77  SW2  PIC X VALUE "Y".
77  TFILEB PIC 9(6) VALUE 0.
77  WTLCNT  PIC 99 VALUE ZEROS.
77  WLOTMP  PIC X(4) VALUE SPACES.
01  LABELF.
    03  IDENTX PIC XXXX.
    03  FILLER PIC X(17).
    03  REKCNT PIC 9(6).
    03  FILLER PIC X(53).
```

```
01 WKDATE.
    03  YY PIC 99.
    03  MM PIC 99.
    03  DY PIC 99.
01 NAME-MONTH.
    03  JAN PIC XXXX VALUE " JAN".
    03  FEB PIC XXXX VALUE " FEB".
    03  MAR PIC XXXX VALUE " MAR".
    03  APR PIC XXXX VALUE " APR".
    03  MAY PIC XXXX VALUE " MAY".
    03  JUNE PIC XXXX VALUE "JUNE".
    03  JULY PIC XXXX VALUE "JULY".
    03  AUG PIC XXXX VALUE " AUG".
    03  SEPT PIC XXXX VALUE "SEPT".
    03  OCT PIC XXXX VALUE " OCT".
    03  NOV PIC XXXX VALUE " NOV".
    03  DEC PIC XXXX VALUE " DEC".
01 MONTH-TABLE REDEFINES NAME-MONTH.
    03  MONTHY PIC XXXX OCCURS 12 TIMES .
01  JOBCNL .
    03  CC PIC X VALUE "1".
    03  PCNL.
        05  CNTYPE PIC XXX.
        05  GO-CNL PIC 999.
        05  PCNLX PIC X(74).
    03  FILLER PIC X(52) VALUE SPACES.
01  JBCNLX .
    03  CC PIC X VALUE "0".
    03  PSTATS PIC X(14) VALUE "RECORD COUNTS".
    03  FILLER PIC X(6) VALUE SPACES.
    03  XCARD PIC X(8) VALUE "CARDS IN".
    03  CSTAT PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDRLNS PIC X(12) VALUE "DETAIL LINES".
    03  DRLNS PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XDATIZN PIC X(7) VALUE "DATA IN".
    03  DATAIN PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XFAIN PIC X(11) VALUE "OLD-DATA I".
    03  FADATA PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  XNOUT PIC X(12) VALUE "FMT DATA OUT".
    03  FOUT PIC ZZZZZ9.
01  JOBCNX2.
    03  CC  PIC X VALUE "0".
    03  JC21  PIC X(10) VALUE "SITES READ".
    03  STAT2A  PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  JC22  PIC X(12) VALUE "WATER LEVELS".
    03  STAT22 PIC ZZZZZ9.
    03  FILLER PIC X(8) VALUE SPACES.
    03  JC2G  PIC X(13) VALUE "LINES PRINTED".
    03  STAT2G PIC ZZZZZ9.
```

```
01  HDR1 .
    03 CC1 PIC 9 VALUE 1.
    03 FILLER PIC X.
    03 LDATE PIC X(6) VALUE "DATE: ".
    03 RMTH PIC X(4).
    03 FILLER PIC X.
    03 RDY PIC Z9.
    03 XB PIC XXXX VALUE ", 19".
    03 RYR PIC XX.
    03 FILLER PIC X(21).
    03 TITLEA PIC X(60).
    03 FILLER PIC X(21).
    03 PGE PIC X(4) VALUE "PAGE".
    03 PNUM PIC ZZZ9.
    03 FILLER PIC X(2).
01  HDR2 .
    03 CC PIC 9 VALUE 0.
    03 USGS PIC X(7) VALUE " USGS: ".
    03 GSNAME PIC X(20).
    03 FILLER PIC XXX.
    03 BASINL PIC X(7) VALUE "SOURCES".
    03 FILLER PIC X.
    03 SRCEID PIC X(50).
    03 SRCNAM PIC X(30).
    03 FILLER PIC X(14).
01  HDR3 .
    03  CC PIC 9 VALUE 0.
    03  FILLER PIC X VALUE SPACES.
    03  FILLER PIC X(3) VALUE "REC".
    03  FILLER PIC XX.
    03  FILLER PIC X(15) VALUE "SITE IDENTIFIER".
    03  FILLER PIC X(8).
    03  FILLER PIC X(10) VALUE "LOCAL SITE".
    03  FILLER PIC X(9).
    03  FILLER PIC X(4) VALUE "SITE".
    03  FILLER PIC X.
    03  FILLER  PIC X(5) VALUE "ALTI-".
    03  FILLER  PIC XX.
    03  FILLER PIC X(4) VALUE "WELL".
    03  FILLER PIC X(16).
    03  FILLER PIC X(19) VALUE "WATER LEVEL HISTORY".
    03  FILLER PIC X(30).

01  HDR4.
    03  CC PIC X VALUE SPACES.
    03  FILLER PIC X VALUE SPACES.
    03  FILLER PIC X(3) VALUE "NUM".
    03  FILLER PIC X(4).
    03  FILLER PIC X(3) VALUE "LAT".
    03  FILLER PIC X(4).
    03  FILLER PIC X(4) VALUE "LONG".
    03  FILLER PIC XX.
    03  FILLER PIC XXX VALUE "DUP".
```

```
    03  FILLER PIC X(5).
    03  FILLER PIC X(10) VALUE "IDENTIFIER".
    03  FILLER PIC X(9).
    03  FILLER PIC X(4) VALUE "TYPE".
    03  FILLER PIC X.
    03  FILLER PIC X(4) VALUE "TUDE".
    03  FILLER PIC X(3).
    03  FILLER PIC X(5) VALUE "DEPTH".
    03  FILLER PIC X(4).
    03  FILLER PIC X(5) VALUE "LEVEL".
    03  FILLER PIC X(4).
    03  FILLER PIC X(4) VALUE "DATE".
    03  FILLER  PIC X(7).
    03  FILLER PIC X(5) VALUE "LEVEL".
    03  FILLER PIC X(4).
    03  FILLER PIC X(4) VALUE "DATE".
    03  FILLER PIC X(7).
    03  FILLER PIC X(5) VALUE "LEVEL".
    03  FILLER PIC X(4).
    03  FILLER PIC X(4) VALUE "DATE".
    03  FILLER PIC X(7).
01 SITELN.
    03  CC PIC 9 VALUE SPACE.
    03  SITEHD.
        05  CNRECP PIC ZZZ9.
        05  FILLER PIC XXX.
        05  KEYID.
            07 KEYLAT PIC 9(6).
            07 FILLER PIC X.
            07 KEYLNG PIC 9(7).
            07 FILLER PIC X.
            07 KEYDUP PIC 99.
        05  FILLER PIC XXX.
        05  LCWLID.
            07  LCSITE PIC X(14).
            07  LCREST PIC X(6).
        05  FILLER PIC XX.
        05  STYP  PIC X.
        05  FILLER PIC XX.
        05  ALTUDE PIC ZZZZZ.
        05  FILLER PIC XX.
        05  SDEPTH PIC ZZZZZ.
        05  FILLER PIC X(4).
    03  LEVDAT  PIC X(62).
01  LEVLN.
    03  LVDAT OCCURS 3 TIMES.
        05  LINT  PIC ZZZZZ.
        05  LPT   PIC X.
        05  LDEC  PIC Z.
        05  FILLER  PIC XX.
        05  LMO  PIC ZZ.
        05  LSYM1  PIC X.
        05  LDAY PIC ZZ.
        05  LSYM2  PIC X.
        05  LYR  PIC ZZZZ.
        05  FILLER  PIC XX.
```

```
01  TMPLVL.
    03  INTLVL  PIC 99999V.
     03  DECLVL  PIC 9.
01  DATAA .
    03  ID-CNL PIC IS XXX.
    03  ID-NUM PIC IS 999.
    03  TITLEX.
        05  SRCAGC.
            07  GNAME PIC X(20).
            07  CNAME PIC X(30).
        05  FILLER PIC X(10).
    03  FILLER PIC X(14).
01  BCARD.
    03  LOLAT PIC 9(6).
    03  LOLNG PIC 9(7).
    03  HILAT PIC 9(6).
    03  HILNG PIC 9(7).
    03  BASIN1 PIC X(7).
    03  BASINX REDEFINES BASIN1.
        05  BASIN2 PIC X(6).
        05  FILLER PIC X.
    03  BTYPE PIC X.
    03  BDESRP PIC X(50).
01  VCARD.
    03  LATUDE-L PIC 9(6).
    03  LNGTUD-L PIC 9(7).
    03  LATUDE-H PIC 9(6).
    03  LNGTUD-H PIC 9(7).
    03  DECLAT-L PIC 99V9999.
    03  LS1 PIC X.
    03  DECLNG-L PIC 999V9999.
    03  DECLAT-H PIC 99V9999.
    03  LS2 PIC X.
    03  DECLNG-H PIC 999V9999.
    03  ASIGN PIC X.
    03  ACOEFF PIC 9V9999999.
    03  BSIGN PIC X.
    03  BCOEFF PIC 9V9999999.
    03  CSIGN PIC X.
    03  KNSTNT PIC 999V9999.
    03  VSEQ PIC 9999.
01  LATONE.
    03  LATUDE-1 PIC 9(6) OCCURS 50 TIMES .
01  LNGONE.
    03  LNGTUD-1 PIC 9(7) OCCURS 50 TIMES .
01  LATTWO.
    03  LATUDE-2 PIC 9(6) OCCURS 50 TIMES .
01  LNGTWO.
    03  LNGTUD-2 PIC 9(7) OCCURS 50 TIMES .
01  DATONE.
    03  DECLAT-1 PIC 99V9999 OCCURS 50 TIMES .
01  DNGONE.
    03  DECLNG-1 PIC 999V9999 OCCURS 50 TIMES .
```

```
01  DATTWO.
    03  DECLAT-2 PIC 99V9999 OCCURS 50 TIMES .
01  DNGTWO.
    03  DECLNG-2 PIC 999V9999 OCCURS 50 TIMES .
01  ACOEF.
    03  ACOEFS USAGE IS COMP-2 OCCURS 50 TIMES .
01  BCOEF.
    03  BCOEFS USAGE IS COMP-2 OCCURS 50 TIMES .
01  CCOEF.
    03  KNSTNS USAGE IS COMP-2 OCCURS 50 TIMES .
01  LATDMS.
    03  LATD PIC 99.
    03  LATM PIC 99.
    03  LATS PIC 99.
01  LNGDMS.
    03  LNGD PIC 999.
    03  LNGM PIC 99.
    03  LNGS PIC 99.
01  TYPLOG.
    03  LOGTYP PIC X OCCURS 8 TIMES .
01  RTW000.
    03  KEYID.
        05 KEYLAT PIC 9(6).
        05 KEYLNG PIC 9(7).
        05 KEYDUP PIC 99.
    03 RNUM PIC 999.
    03 LNUM PIC 999.
    03  SESAFE PIC X.
    03 SITETY PIC X.
    03 DRELIX PIC X.
    03 REPAGY PIC X(5).
    03  PROJNU.
        05  BASIN1 PIC X(7).
        05  BASIN2 PIC X(6).
    03 DISTRT PIC XXX.
    03 STATEW PIC XX.
    03 COUNTY PIC XXX.
    03 LLACUR PIC X.
    03 LCWLID.
        05  LCSITE PIC X(14).
        05  LCREST PIC X(6).
    03  LCLNET PIC X(30).
    03 LCLMAP PIC X(30).
    03 SCALEN PIC X(6).
    03 ALTUDE PIC 9(5)V99.
    03 MMEASR PIC X.
    03 MACCUR PIC XXX.
    03 TOPSET PIC X.
    03 OWDCOR PIC X(8).
    03  FCMNTH PIC XX.
    03  FCCDAY PIC XX.
    03  FCYEAR PIC XXXX.
    03 FILLER PIC X.
    03 SITUSE PIC X.
    03 WTRUSE PIC X.
    03 SWTRUS PIC X.
```

```
    03 TWTRUS PIC X.
    03 HDEPTH PIC 9(5)V99.
    03 WDEPTH PIC 9(5)V99.
    03 WDSRCE PIC X.
    03 WTRLVL PIC 9(5)V99.
    03  WLDAT.
        05  WLMNTH PIC XX.
        05  WLLDAY PIC XX.
        05  WLYEAR PIC XXXX.
    03 WLDACR PIC X.
    03 WLSRCE PIC X.
    03 WLMETH PIC X.
    03  PUMPUS PIC X.
    03 GHSRCE PIC X.
    03 XTATUS PIC X.
    03 LUPDAT PIC X(8).
    03 VERIFY PIC X.
    03  MSHGHT PIC 999V99.
    03  MSDATE PIC 9(8).
    03  CURECD PIC X.
01  RTW234.
    03  KEYID.
        05  KEYLAT  PIC 9(6).
        05  KEYLNG  PIC 9(7).
        05  KEYDUP  PIC 99.
    03  RNUM  PIC 999.
    03  LNUM  PIC 999.
    03  SESAFE  PIC X.
    03  WLDATE.
        05  WLMNTH  PIC 99.
        05  WLDAY   PIC 99.
        05  WLYEAR  PIC 9999.
    03  DTACCR  PIC X.
    03  WTRLVL PIC 9(4)V999.
    03  WLMETH PIC X.
    03  MEASAC PIC X.
01  RTQ002.
    03  KEYID  PIC 9(15).
    03  RNUM   PIC 999.
    03  LNUM   PIC 999.
    03  SESAFE PIC X.
    03  LCWLID  PIC X(20).
    03  LCREST  PIC X(44).
01  BLNKHD.
    03  HEAD  PIC X(69).
    03  BLVL  PIC X(62).
```

```
 PROCEDURE DIVISION.
 DECLARATIVES.
 SFILEB SECTION.
     USE AFTER STANDARD ENDING FILE LABEL
         PROCEDURE ON FILEB.
 LFILEB.
     MOVE FILE-LABEL TO LABELF.
     MOVE TFILEB TO REKCNT.
     MOVE LABELF TO FILE-LABEL.
     DISPLAY FILE-LABEL, TFILEB, REKCNT.
 END DECLARATIVES.
*NOTE-A.
*      THE PROGRAM STARTS WITH THIS INITIALIZATION
*      ROUTINE THAT OPENS THE INPUT AND LISTING FILES
*      AND THEN THE DATE STORED IN THE SYSTEM IS READ
*      AND USED TO SET UP THE DATE IN THE FIRST
*      HEADING LINE.  NO RETURN FROM THE MAINLINE
*      ROUTINE IS MADE TO THIS PROCEDURE.
 BEGIN.
     OPEN INPUT CARD-FILE.
     OPEN OUTPUT LIST-FILE.
     ACCEPT WKDATE FROM DATE.
     MOVE MM TO NMM.
     MOVE MONTHY (NMM) TO RMTH OF HDR1.
     MOVE DY TO RDY OF HDR1.
     MOVE YY TO RYR OF HDR1.
     MOVE SPACES TO WLOTMP.
     MOVE ZEROS TO ENTEMP.
*NOTE-B.
*      THIS ROUTINE READS THE CONTROL CARDS AND MOVES
*      THE GENERAL DESCRIPTION OF THE REPORT TO THE
*      HEADING AREA.  THEN A BRANCH IS TAKEN TO THE
*      APPROPRIATE STARTING ROUTINE DEPENDING ON THE
*      NUMBER IN COLUMNS 4-6 IN THE FIRST CONTROL CARD.
*      THERE IS NO RETURN TO THIS ROUTINE.
 READC.
     READ CARD-FILE AT END GO TO STOP1.
     IF ID-CNL OF CNL-CARD NOT = "CNL" GO TO STOP2.
     IF ID-NUM OF CNL-CARD = 999 GO TO STOP3.
     ADD 1 TO INCRDS.
     MOVE CNL-CARD TO PCNL.
     READ CARD-FILE INTO DATAA AT END GO TO STOP4.
     IF ID-CNL OF DATAA NOT = "CNL" GO TO STOP4.
     IF ID-NUM OF DATAA = 010 MOVE TITLEx OF DATAA
         TO TITLEA OF HDR1 ELSE GO TO STOP4.
     ADD 1 TO INCRDS.
     IF GO-CNL = 001, GO TO STARTA, ELSE GO TO STOP2.
 STARTA.
     OPEN INPUT FILEB.
     OPEN OUTPUT PLTFLE.
 READB.
     READ CARD-FILE INTO BCARD AT END GO TO STOP5.
     ADD 1 TO INCRDS.
```

```
*NOTE-C.
*     NO RETURNS ARE MADE TO ANY OF THE ROUTINES BEFORE
*     THIS POINT IN THE PROGRAM FROM ANY OF THE FOLLOWING
*     ROUTINES.  A RETURN TO READB1 IS MADE FROM THE
*     ROUTINE ENDFLE WHEN A TAPE RECORD IS LESS THAN THE
*     THE LOWEST LATITUDE IN THE CURRENT BASIN AND THERE
*     IS NOT A TERMINATION RECORD SIGNALLED BY 000000 IN
*     THE FIRST SIX POSITIONS OF THE CURRENT RECORD.
 READB1.
      MOVE BDESRP OF BCARD TO SRCEID OF HDR2.
      MOVE BASIN1 OF BCARD TO BASINL OF HDR2.
*NOTE-D.
*     THE ROUTINES FROM HERE TO THE ROUTINE
*     FINDPT READ AND STORE THE SLICES FOR THE
*     BASIN WHILE EDITING AND COMPUTING THE
*     COEFFICIENTS FOR THE LINE SEGMENTS.
*     EXIT IS TAKEN WHEN THE FIRST RECORD
*     OF THE NEXT SLICE IS READ AND A BRANCH
*     IS TAKEN TO THE ROUTINE, FINDPT.
*     THE RETURN IS TO READV1 SINCE THE FIRST
*     RECORD OF THE NEXT SLICE HAS NOT BEEN
*     STORED IN THE TABLE.
 READV.
      READ CARD-FILE INTO VCARD AT END GO TO STOP5.
      ADD 1 TO INCRDS.
 READV1.
      IF FRSW = "F", PERFORM CLRVSS VARYING LA1 FROM 1 BY
          1 UNTIL LA1 = 51,
          MOVE "Y" TO FRSW, SET LA1 TO 1,
          MOVE LATUDE-L OF VCARD TO NHOLD.
      IF LATUDE-L OF VCARD < NHOLD, GO TO FINDPT.
 STRVCD.
      IF LATUDE-L OF VCARD > NHOLD, GO TO STOP8.
      EXAMINE LATUDE-L OF VCARD REPLACING ALL " " BY 0.
      MOVE LATUDE-L OF VCARD TO LATUDE-1 (LA1).
      EXAMINE LNGTUD-L OF VCARD REPLACING ALL " " BY 0.
      MOVE LNGTUD-L OF VCARD TO LNGTUD-1 (LA1).
      EXAMINE LATUDE-H OF VCARD REPLACING ALL " " BY 0.
      MOVE LATUDE-H OF VCARD TO LATUDE-2 (LA1).
      EXAMINE LNGTUD-H OF VCARD REPLACING ALL " " BY 0.
      MOVE LNGTUD-H OF VCARD TO LNGTUD-2 (LA1).
      EXAMINE DECLAT-L OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLAT-L OF VCARD TO DECLAT-1 (LA1).
      EXAMINE DECLNG-L OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLNG-L OF VCARD TO DECLNG-1 (LA1).
      EXAMINE DECLAT-H OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLAT-H OF VCARD TO DECLAT-2 (LA1).
      EXAMINE DECLNG-H OF VCARD REPLACING ALL " " BY 0.
      MOVE DECLNG-H OF VCARD TO DECLNG-2 (LA1).
      EXAMINE ACOEFF OF VCARD REPLACING ALL " " BY 0.
      MOVE ACOEFF OF VCARD TO ACOEFS (LA1).
      IF ASIGN OF VCARD = "-", COMPUTE ACOEFS (LA1) =
           ACOEFS (LA1) - 2 * ACOEFS (LA1),
        ELSE MOVE "+" TO ASIGN OF VCARD.
```

```
     EXAMINE BCOEFF OF VCARD REPLACING ALL " " BY 0.
     MOVE BCOEFF OF VCARD TO BCOEFS (LA1).
     IF BSIGN OF VCARD = "-", COMPUTE BCOEFS (LA1) =
          BCOEFS (LA1) - 2 * BCOEFS (LA1),
       ELSE MOVE "+" TO BSIGN OF VCARD.
     EXAMINE KNSTNT OF VCARD REPLACING ALL " " BY 0.
     MOVE KNSTNT OF VCARD TO KNSTNS (LA1).
     IF CSIGN OF VCARD = "-", COMPUTE KNSTNS (LA1) =
          KNSTNS (LA1) - 2 * KNSTNS (LA1),
       ELSE MOVE "+" TO CSIGN OF VCARD.
     SET LN2 TO LA1.
     SET LA1 UP BY 1.
     GO TO READV.
*NOTE-E.
*    THIS IS THE TAPE READ ROUTINE THAT CONTROLS THE MAIN
*     PROGRAM.  THE KEY ROUTINES ARE READTAPE, BRNALL,
*    CHNG01, CHNG02, TEST2.
*    THIS PROCEDURE TESTS THE FIRST RECORD TAPE
*    SWITCH FOR THE SITE BEING OUTSIDE THE BASIN
*    AND CONTINUES TO READ THE MASTER FILE UNTIL
*    A SITE WITHIN THE BASIN IS FOUND.
*    RECORDS ARE PROCESSED FOR THAT SITE UNTIL
*    A CHANGE OF SITE KEYID IS DETECTED.  A
*    BRANCH IS MADE TO CHNG01 WHERE THE OUTPUT
*    FOR THE PREVIOUS IS MADE IF APPLICABLE.
*    THEN IN CHNG02 THE LAST SITE RECORDS ARE
*    CLEARED AND THE NEW SITE IS SET UP FOR
*    SEARCH AND PROCESSING.
 FINDPT.
     MOVE "F" TO FRSW.
     IF (SLSW = "X") MOVE "Y" TO SLSW, GO TO TEST2.
     GO TO READTAPE.
 READTAPE.
     READ FILEB INTO LIST-LINE AT END GO TO CLOSEA.
     ADD 1 TO TFILEB, NFLEIA.
     IF KEYID OF OLDMAS NOT = KEYIDL, GO TO CHNG01.
*NOTE-WELL.
*    THE BRANCH INSTRUCTIONS IN THE PROCEDURE, BRNALL,
*    ARE ORDERED BY THEIR RELATIVE FREQUENCY IN THE
*    WRDM MASTER FILE.  REARRANGEMENT WOULD SLOW
*    THE OPERATION.
 BRNALL.
     IF FRTSW = "X", GO TO READTAPE.
     IF RNUM OF OLDMAS = 234, GO TO PTR234.
     IF RNUM OF OLDMAS = 000, GO TO PTR000.
     IF RNUM OF OLDMAS = 001, GO TO PTR001.
     IF RNUM OF OLDMAS = 002, GO TO PTR002.
     IF RNUM OF OLDMAS = 003, GO TO PTR003.
     IF RNUM OF OLDMAS = 058, GO TO PTR058.
     IF RNUM OF OLDMAS = 076, GO TO PTR076.
     IF RNUM OF OLDMAS = 198, GO TO PTR198.
     IF RNUM OF OLDMAS = 042, GO TO PTR042.
     IF RNUM OF OLDMAS = 082, GO TO PTR082.
```

```
      IF RNUM OF OLDMAS = 158, GO TO PTR158.
      IF RNUM OF OLDMAS = 192, GO TO PTR192.
      IF RNUM OF OLDMAS = 146, GO TO PTR146.
      IF RNUM OF OLDMAS = 186, GO TO PTR186.
      IF RNUM OF OLDMAS = 090, GO TO PTR090.
      IF RNUM OF OLDMAS = 114, GO TO PTR114.
      IF RNUM OF OLDMAS = 121, GO TO PTR121.
      IF RNUM OF OLDMAS = 189, GO TO PTR189.
      IF RNUM OF OLDMAS = 094, GO TO PTR094.
      IF RNUM OF OLDMAS = 183, GO TO PTR183.
      IF RNUM OF OLDMAS = 072, GO TO PTR072.
      IF RNUM OF OLDMAS = 047, GO TO PTR047.
      IF RNUM OF OLDMAS = 134, GO TO PTR134.
      IF RNUM OF OLDMAS = 171, GO TO PTR171.
      IF RNUM OF OLDMAS = 180, GO TO PTR180.
      IF RNUM OF OLDMAS = 127, GO TO PTR127.
      IF RNUM OF OLDMAS = 055, GO TO PTR055.
      IF RNUM OF OLDMAS = 098, GO TO PTR098.
      IF RNUM OF OLDMAS = 105, GO TO PTR105.
      IF RNUM OF OLDMAS = 164, GO TO PTR164.
      IF RNUM OF OLDMAS = 203, GO TO PTR203.
      IF RNUM OF OLDMAS = 208, GO TO PTR208.
      IF RNUM OF OLDMAS = 212, GO TO PTR212.
      IF RNUM OF OLDMAS = 219, GO TO PTR219.
      IF RNUM OF OLDMAS = 250, GO TO PTR250.
      IF RNUM OF OLDMAS = 277, GO TO PTR277.
      GO TO STOP9.
 CHNG01.
      IF (FRTSW = "Y" AND LDSW = ALFAX)  PERFORM PRTSNL,
          MOVE ZEROS TO CNTSL, MOVE SPACES TO LIST-LINE,
          ADD 1 TO LNCNT, WRTTE LIST-LINE.
       MOVE ALFAZ TO PRTSW.
       MOVE ALFAY TO SW0, SW1, SW2, LDSW.
 CHNG02.
      MOVE KEYID OF OLDMAS TO KEYIDL.
      MOVE SPACES TO SITELN.
      MOVE SPACES TO RECOUT.
      MOVE "X" TO FRTSW, HSW, NSW, LSW.
      MOVE KEYLAT OF OLDMAS TO NLATMS.
      MOVE KEYLNG OF OLDMAS TO NLNGMS.
      GO TO TEST2.
*NOTE-P.
*     THE FOLLOWING ROUTINES ARE IN PLACE TO
*     ALLOW PROCESSING OF ANY RECORD IN THE
*     (WRDM) WATER RESOURCES MASTER TAPE FILE
*     DATA MAY BE EXTRACTED AND STORED IN THE
*     LAST 41 POSITIONS OF "SITELN" AND "RECOUT"
*     FOR STORING ON YOUR CYBER DISK FILE AND
*     PRINTING ON THE LISTING.
*
*
*     ADDITIONAL OUT PUT MAY BE HANDLED BY
*     SETTING UP ADDITIONAL OUTPUT RECORDS
*     AND INSERTING THE PROPER INSTRUCTIONS
*     TO ADD TO THE OUTPUT
```

```
PTR000.
    MOVE ALFAX TO SW0.
    SET WTLCNT TO ZERO.
     ADD 1 TO COUNTA.
    MOVE "Y" TO HSW.
    MOVE OLDMAS TO RTW000.
    IF ( SITETY OF RTW000 = "S" ),
         GO TO READTAPE.
PRMOVE.
    MOVE KEYLAT OF RTW000 TO KEYLAT OF SITELN,
         KEYLAT OF RECOUT.
    MOVE KEYLNG OF RTW000 TO KEYLNG OF RECOUT, KEYLNG OF SITELN.
    MOVE KEYDUP OF RTW000 TO KEYDUP OF RECOUT, KEYDUP OF SITELN.
    MOVE LCWLID OF RTW000 TO LCWLID OF RECOUT, LCWLID OF SITELN.
     EXAMINE ALTUDE OF RTW000 REPLACING ALL " " BY 0.
    MOVE ALTUDE OF RTW000 TO ALTUDE OF RECOUT, ALTUDE OF SITELN.
     EXAMINE WDEPTH OF RTW000 REPLACING ALL " " BY 0.
     MOVE WDEPTH OF RTW000 TO RWDEPTH.
     EXAMINE WLDAT OF RTW000 REPLACING ALL " " BY 0.
     EXAMINE WTRLVL OF RTW000 REPLACING ALL " " BY 0.
     IF (WTRLVL OF RTW000 = 0)  GO TO READTAPE , ELSE
         ADD 1 TO WTLCNT,
         MOVE WTRLVL OF RTW000 TO TMPLVL,
         MOVE INTLVL TO LINT (WTLCNT),
         MOVE DECLVL TO LDEC (WTLCNT),
         MOVE "." TO LPT (WTLCNT),
         MOVE WLMNTH OF RTW000 TO LMO (WTLCNT),
         MOVE WLLDAY OF RTW000 TO LDAY (WTLCNT),
         MOVE WLYEAR OF RTW000 TO LYR (WTLCNT),
         MOVE "/" TO LSYM1 (WTLCNT), LSYM2 (WTLCNT),
         MOVE ALFAX TO LDSW.
      GO TO READTAPE.
PTR001.
     MOVE ALFAX TO SW1.
     IF NOT (SW0 = ALFAX), PERFORM READTAPE,
        IF RNUM OF OLDMAS = 002, MOVE OLDMAS TO RTQ002,
            MOVE LCWLID OF RTQ002 TO LCWLID OF SITELN.
     GO TO READTAPE.
PTR002.
    MOVE "N" TO NSW.
    MOVE ALFAX TO SW2.
    IF LDSW = ALFAY, MOVE LCWLID OF RTQ002 TO LCWLID OF SITELN.
    IF LCWLID OF SITELN = ALFABLNK,
         MOVE "NO LOCAL ID AVAILABLE" TO LCWLID OF SITELN.
    GO TO READTAPE.
PTR003.
    GO TO READTAPE.
PTR042.
    GO TO READTAPE.
PTR047.
    GO TO READTAPE.
PTR055.
    GO TO READTAPE.
```

```
PTR058.
    GO TO READTAPE.
PTR072.
    GO TO READTAPE.
PTR076.
    GO TO READTAPE.
PTR082.
    GO TO READTAPE.
PTR090.
    GO TO READTAPE.
PTR094.
    GO TO READTAPE.
PTR098.
    GO TO READTAPE.
PTR105.
    GO TO READTAPE.
PTR114.
    GO TO READTAPE.
PTR121.
    GO TO READTAPE.
PTR127.
    GO TO READTAPE.
PTR134.
    GO TO READTAPE.
PTR146.
    GO TO READTAPE.
PTR158.
    GO TO READTAPE.
PTR164.
    GO TO READTAPE.
PTR171.
    GO TO READTAPE.
PTR180.
    GO TO READTAPE.
PTR183.
    GO TO READTAPE.
PTR186.
    GO TO READTAPE.
PTR189.
    GO TO READTAPE.
PTR192.
    GO TO READTAPE.
PTR198.
    GO TO READTAPE.
PTR203.
    GO TO READTAPE.
PTR208.
    GO TO READTAPE.
PTR212.
    GO TO READTAPE.
PTR219.
    GO TO READTAPE.
```

```
PTR234.
    IF NOT (SW0 = ALFAX) MOVE
          "NO SITE HEADER REC" TO LCWLID OF SITELN,
           MOVE KEYLAT OF RTW234 TO KEYLAT OF SITELN,
           MOVE KEYLNG OF RTW234 TO KEYLNG OF SITELN,
           MOVE KEYDUP OF RTW234 TO KEYDUP OF SITELN.
    ADD 1 TO COUNT2.
    MOVE OLDMAS TO RTW234.
    IF (WTLCNT = 3) ADD 1 TO CNTSL, PERFORM PRTSNL,
        SET WTLCNT TO 0.
    EXAMINE WLDATE OF RTW234 REPLACING ALL " " BY 0.
    EXAMINE WTRLVL OF RTW234 REPLACING ALL " " BY 0.
    IF (WTRLVL OF RTW234 = 0),   GO TO READTAPE, ELSE
       ADD 1 TO WTLCNT,
       MULTIPLY WTRLVL OF RTW234 BY 0.1 GIVING WTRLVL OF RTW234,
       MOVE WTRLVL OF RTW234 TO TMPLVL, RWTLEV,
       MOVE INTLVL TO LINT (WTLCNT),
       MOVE DECLVL TO LDEC (WTLCNT),
       MOVE "." TO LPT (WTLCNT),
       MOVE WLMNTH OF RTW234 TO LMO (WTLCNT),
       MOVE WLDAY OF RTW234 TO LDAY (WTLCNT),
       MOVE WLYEAR OF RTW234 TO LYR (WTLCNT),
       MOVE RECNT TO CNRECR.
       MOVE WLDATE TO RWDATE.
       EXAMINE RWDATE REPLACING ALL " " BY 0.
       EXAMINE RWTLEV REPLACING ALL " " BY 0.
       MOVE SPACES TO CCR, RWRST, XSP.
       MOVE 88 TO INDO.
       WRITE RECOUT.
       ADD 1 TO CNROUT.
       MOVE "/" TO LSYM1 (WTLCNT), LSYM2 (WTLCNT).
       MOVE ALFAX TO LDSW.
    GO TO READTAPE.
PTR250.
    GO TO READTAPE.
PTR277.
    GO TO READTAPE.
*NOTE-T.
*     THIS ROUTINE TESTS THE LINE SEGMENT EQUATIONS
*     TO DETERMINE IF THE SITE IS WITHIN THE BASIN.
*     THE FRTSW SWITCH IS SET TO "Y" IF THE SITE
*     IS WITHIN THE BASIN.  EXIT IS TAKEN TO ENDFLE
*     IF THE LATITUDE OF THE SITE BEING TESTED IS
*      GREATER THAN THE HIGH LATITUDE OF THE
*     CURRENT SLICE.   EXIT IS TO READTAPE IF THE
*     SITE IS OUTSIDE THE RECTANGLE THAT BOUNDS
*     THE CURRENT SLICE.    IF THE SITE IS WITHIN
*     THE RECTANGLE BUT OUTSIDE THE BASIN EXIT
*     IS TAKEN TO READTAPE.  ONLY IF THE SITE IS
*     WITHIN THE BASIN IS THE BRANCH TO BRNALL
*     TAKEN.
```

```
TEST2.
    IF NLATMS < LATUDE-1(1), GO TO ENDFLE.
    IF NLATMS > LATUDE-2(1), GO TO READTAPE.
    IF NLNGMS < LOLNG OF BCARD, GO TO READTAPE.
    IF NLNGMS > HILNG OF BCARD, GO TO READTAPE.
    MOVE NLATMS TO LATDMS.
    MOVE LATD TO FLATD.
    MOVE LATM TO FLATM.
    MOVE LATS TO FLATS.
    COMPUTE FDCLAT = FLATD + FLATM / 60 + FLATS / 3600.
    MOVE NLNGMS TO LNGDMS.
    MOVE LNGD TO FLNGD.
    MOVE LNGM TO FLNGM.
    MOVE LNGS TO FLNGS.
    COMPUTE FDCLNG = FLNGD + FLNGM / 60 + FLNGS / 3600.
    COMPUTE FDCLNG = FDCLNG - 2 * FDCLNG.
    SET DA1 TO 1.
TEST21.
    COMPUTE KTEST = ACOEFS (DA1) * FDCLNG + BCOEFS (DA1)
        * FDCLAT.
    COMPUTE LTEST = ACOEFS (DA1 + 1) * FDCLNG
        + BCOEFS (DA1 + 1 ) * FDCLAT.
    IF (KTEST < KNSTNS (DA1) OR LTEST > KNSTNS (DA1 + 1))
          GO TO NEXTB, ELSE GO TO SITEIN.
NEXTB.
    IF (DA1 + 1 = LN2), GO TO READTAPE,
        ELSE SET DA1 UP BY 2, GO TO TEST21.
SITEIN.
    MOVE "Y" TO FRTSW.
    GO TO BRNALL.
ENDFLE.
    IF LATUDE-L OF VCARD = 000000, GO TO CLOSEA.
    IF LATUDE-L OF VCARD < LOLAT OF BCARD,
        MOVE VCARD TO BCARD, GO TO READ81,
        ELSE MOVE "F" TO FRSW, MOVE "X" TO SLSW, GO TO READV1.
PRTSNL.
    ADD 1 TO COUNT2G.
    IF (CNTSL NOT > 1) ADD 1 TO RECNT.
    IF LNCNT > 59, PERFORM HDPRNT,
        ELSE ADD 1 TO LNCNT.
     ADD 1 TO CNTSL, PERFORM CLRSHD.
     IF (WTLCNT = 1) PERFORM CLRLVL 2 TIMES.
     IF (WTLCNT = 2) PERFORM CLRLVL 1 TIMES.
     MOVE LEVLN TO LEVDAT,
     MOVE RECNT TO CNRECP OF SITELN,
     IF LDSW = ALFAX,
         IF PRTSW = "A", MOVE LEVLN TO BLVL,
              WRITE LIST-LINE FROM BLNKHD
         ELSE WRITE LIST-LINE FROM SITELN.
     MOVE SPACES TO BLNKHD.
     MOVE SPACES TO LIST-LINE.
     MOVE ZEROS TO WTLCNT.
     PERFORM CLRLVL 3 TIMES.
     MOVE "A" TO PRTSW.
```

```
HDPRNT.
    ADD 1 TO PGCNT.
    MOVE PGCNT TO PNUM OF HDR1.
    SET LNCNT TO 10.
    WRITE LIST-LINE FROM HDR1.
    WRITE LIST-LINE FROM HDR2.
    WRITE LIST-LINE FROM HDR3.
    WRITE LIST-LINE FROM HDR4.
    MOVE SPACES TO LIST-LINE.
    WRITE LIST-LINE.
    WRITE LIST-LINE.
CLOSEA.
    IF FRTSW = "Y", PERFORM PRTSNL.
    CLOSE FILEB.
    GO TO STOP3.
CLRVSS.
    MOVE 0 TO LATUDE-1 (LA1).
    MOVE 0 TO LNGTUD-1 (LA1).
    MOVE 0 TO LATUDE-2 (LA1).
    MOVE 0 TO LNGTUD-2 (LA1).
    MOVE 0 TO DECLAT-1 (LA1).
    MOVE 0 TO DECLNG-1 (LA1).
    MOVE 0 TO DECLAT-2 (LA1).
    MOVE 0 TO DECLNG-2 (LA1).
    MOVE 0 TO ACOEFS (LA1).
    MOVE 0 TO BCOEFS (LA1).
    MOVE 0 TO KNSTNS (LA1).
 CLRSHD.
     MOVE SITELN TO BLNKHD.
     MOVE SPACES TO HEAD.
 CLRLVL.
     ADD 1 TO WTLCNT.
     MOVE ZEROS TO LINT(WTLCNT).
     MOVE SPACES TO LPT (WTLCNT).
     MOVE ZEROS TO LDEC (WTLCNT).
     MOVE ZEROS TO LMO (WTLCNT).
     MOVE SPACES TO LSYM1 (WTLCNT).
     MOVE ZEROS TO LDAY (WTLCNT).
     MOVE SPACES TO LSYM2 (WTLCNT).
     MOVE ZEROS TO LYR (WTLCNT).
STOP1.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "CONTROL INPUT FILE EMPTY".
    STOP RUN.
STOP2.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "ID MISSING OR INCORRECT IN CONTROL CARD".
    STOP RUN.
STOP3.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY TFILEB.
    MOVE INCRDS TO CSTAT.
    MOVE NDATIN TO DATAIN.
    MOVE NFLEIA TO FADATA.
```

```
    MOVE NDETLN TO DRLNS.
    MOVE NFLEOA TO FOUT.
    MOVE COUNTA TO STAT2A.
    MOVE COUNT2 TO STAT22.
    MOVE COUNT2G TO STAT2G.
    WRITE LIST-LINE FROM JBCNLX.
    WRITE LIST-LINE FROM JOBCNX2.
    DISPLAY "NORMAL JOB TERMINATION".
    DISPLAY " COUNT OF BLANKS ", CNBLNK.
    CLOSE CARD-FILE, LIST-FILE, PLTFLE.
    STOP RUN.
STOP4.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP4 SOURCE ID RECORDS MISSING".
    STOP RUN.
STOP5.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP5 DATA INPUT FILE END INCORRECT".
    DISPLAY "EOF ACCURRED AT READB OR READV".
    STOP RUN.
STOP8.
    WRITE LIST-LINE FROM JOBCNL.
    DISPLAY "STOP8 FILE OUT OF SEQUENCE".
    DISPLAY NHOLD, "LATUDE-L = ", LATUDE-L OF VCARD.
    DISPLAY   "LATUDE-H", LATUDE-H OF VCARD.
    DISPLAY KEYIDL, " 8 ERROR 8 ", KEYID OF OLDMAS.
    STOP RUN.
STOP9.
    DISPLAY "STOP9 RECORD TYPE NOT FOUND".
    DISPLAY OLDMAS.
    STOP RUN.
```

```
      PROGRAM BSNPLT (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT
     1,PUNCH,TAPE7=PUNCH,TAPE15,TAPE16,TAPE17,TAPE18,TAPE19
     2,TAPE20,TAPE21,TAPE22,TAPE23,TAPE24,TAPE25,TAPE26,TAPE27
     3,TAPE28,TAPE29,TAPE30,TAPE31,TAPE32,TAPE33,TAPE34,TAPE35
     4,TAPE46)
C.....
C..... MULTIPLE UNITS ARE USED TO ACCOMMODATE THE RUNNING
C..... OF MULTIPLE PLOTS ON ONE RUN OF THE PROGRAM
C.....
      DIMENSION LPROJ(4),NSTDP(2),LCMS(2),LCWLID(20)
      DIMENSION KEYID(2)
C.....
C..... THIS COMMON STORAGE AREA IS USED BY THE MAIN ROUTINE
C..... AND THE READ CONTROL CARD ROUTINE PRIMARILY
C.....
      COMMON/MALINE/JSTR,TITLE(8),XS(300),YS(300)
     1,KSYMS(300),XLS(300),YLS(300),FHLAT,FLLAT,FHLNG,FLLNG,CM,ZHT
     2,SYHT,IPAGE,NPLOT,SCALE,XHLAT,YHLNG,XLLAT,YLLNG,TITLE2(8)
     3,TITLE3(8),TLA,TLN,FITIC,LLAT,KHLAT,LLNG,KHLNG,KHOLNE,LOP,NCM
     4,TITLE4(8),TITLF5(8),TITLE6(8),TITLE7(8),NF,NOTN(20)
     5,NDS,NBRDR
C.....
      COMMON/SELCT/CNYR1,CNYR2,CDIAM,ILOG(8),IFIN(3),IOPN(3),IWUSE(3)
     $,ISTSW,NFIN,NLOG,NOPN,NWLVL,NWUS,NWELLS,SPCAP,TSTDEP,TSTOPN,TSTWL
     2,TSTWDP,TSTDIM,WLYR1,WLYR2,WTRDEP,WTRLVL,XDEPTH,OPNTOP
C.....
      COMMON/RDR/ALT,CSDM,CNSYR,DEPTH,INWU,LOGS(8),MFIN,MOPN,SPCP
     1,TPOPN,WL,WTLYR,WLCNT
C.....
      COMMON/LBLR/I,NID,NID1,NID2,OLBL1(300),OLBL2(300),BTMALT,WLALT
     1,TOPALT,SEQ
C.....
      COMMON/SYMLD/XL,YL,KSP,HT,ANGLE,NDM
C.....
C.....  DEFAULT VALUES FOR THE LAMBERT CONFORMAL PROJECTION
C.....
      DATA PH33,PH44,AA,BB,CM,SCALE/33.,45.,251109960.5,
     1250258678.8,112.,126720./
C.....
      DATA LPROJ(1),LPROJ(2),LPROJ(3),LPROJ(4)/"LAMBERT CO",
     1"NFORMAL CO","NIC PROJEC","TION      "/
C.....
      DATA NSTDP(1),NSTDP(2)/"STANDARD P","ARALLELS  "/
      DATA LCMS(1),LCMS(2)/"CENTRAL ME","RIDIAN ="/
      DATA BMSG/"BEGIN PLOT"/
      DATA HT,ANGLE,KSP/0.08,90.0,1/
C.....
C.....  NPLOT IS INCREASED BY 1 FOR EACH PLOT RUN
C.....  KDISK IS FILE CONTAINING THE SLICE TABLE FROM PLOYGON PROGRAM
C.....
      CALL PLOTS(0,0,46)
      NPLOT = 0
      NDISK = 20
      KDISK = 15
```

```
C.....
C.....  FOR MULTIPLE PLOTS :
C..... THIS IS THE RESTART POINT FOR PLOTS AFTER THE FIRST
C.....
  100 EOFSW = -1.
      I = 0
      SEQ = 0.0
      KGOSW = -1
      NFILE = 5
      NDPLT = -1
      XE = 2.0
      KEYIDL =999999
      FSLC = 1.0
      EOFSL = 0.0
      ITBSW = 0
      SLCNT = 0.0
      XHT = ZHT
      NBRDR = 1
C.....
C..... THE VARIABLES FOR THIS CONTROL CARD READ ROUTINE
C..... ARE STORED IN THE COMMON MALINE AREA
C.....
C.....  ALL HEADING FOR THE PLOT IS DONE IN THE NREAD ROUTINE
C.....  FOR OUTPUT AFTER THE PLOT FILE IS OPENED
C.....
      CALL NREAD3
      LNCNT=6
      WRITE(6,9007)TITLE,IPAGE,NPLOT,DATE(QQ)
      WRITE(6,9008)
      WRITE(6,9121)
      WRITE(6,9122)
C.....
C.....  IF BASIN BOUNDARIES ARE TO BE INCLUDED IN THE PLOT
C.....  READ THE SLICE TABLE HEADER LINE
C.....
C.....  KHOLNE MAY HAVE THE FOLLOWING CODES AND OPTIONS DEPENDING ON
C.....  NOLNE FOUND ON CONTROL CARD # 3
C.....            1       "NEITHER" OR "      "
C.....            2       "BASIN  "
C.....            3       "MAP    "
C.....            4       "BOTH   "
C.....
      IF(KHOLNE .EQ. 2 .OR. KHOLNE .EQ. 4) GO TO 110
      GO TO 120
  110 READ(KDISK,9020)LLATS,LLNGS,LHLTS,LHLNS
 9020 FORMAT(2(I6,I7),54X)
      READ(KDISK,9021)LLATS,LLNGS,LHLTS,LHLNS,DLLTS,DLLNS,DHLTS,DHLNS,
     $LS1,ARGA,LS2,ARGB,LS3,ARGC
C.....
C.....  DEGREES ARE CONVERTED TO PLOT COORDINATES
C.....  NOTE: PROJECTION ROUTINE IS INITIALIZED HERE
C.....
  120 CALL LMBINIT(FLLAT,FHLAT,AA,BB,CM)
      CALL LMBCMPT(FLLAT,FLLNG,SCALE,YLLNG,XLLAT)
      CALL LMBCMPT(FHLAT,FHLNG,SCALE,YHLNG,XHLAT)
```

```
C.....
C.....  INITIALIZE THE PLOT BAND INCREMENT AND THE
C.....  BOTTOM OF THE PLOT BAND
C.....
      BNDINC = 0.01
      BNDBTM = FHLAT-0.01
C.....
C.....  SET THE WIDTH OF THE STRIP
C.....  0.9 < WIDTH OF THE STRIP < 1.1 INCHES
C.....
  200 BNDBTM = BNDBTM-.002
      BNDINC = BNDINC+.002
      CALL LMBCMPT(BNDBTM,CM,SCALE,YCM,XCM)
      DIFF = XCM-XHLAT
      IF (DIFF.LT.0.0.OR.DIFF.GT.1.1) GO TO 4800
      IF (DIFF.LT.0.9) GO TO 200
      BNDBTM = FHLAT
C.....
C.....  INCREMENT COUNT OF PLOTS FOR THIS RUN
C.....
      EPLOT = NPLOT
      NPLOT = NPLOT +1
      BPLOT = NPLOT
      YTEST = YHLNG-YLLNG
C.....
C.....  TEST FOR WIDTH OF PLOT GREATER THAN WIDTH OF PLOTTER
C.....  PLOT BEGINNING PLOT MESSAGE
C.....
      IF (YTEST.GT.33.0) GO TO 4700
      CALL SYMBOL(0.0,18.4,0.24,BMSG,270.0,10)
      CALL NUMBER(0.0,15.8,0.24,BPLOT,270.0,-1)
      CALL SYMBOL(1.0,7.5,0.24,TITLE,90.0,78)
C.....
C.....  PLOT THE NORTH WEST CORNER AND PRINT THE
C.....  COORDINATES OUTSIDE THE PLOT
C.....
      CALL LMBCMPT(FHLAT,FHLNG,SCALE,YNW,XNW)
      XNW =2.0
      YNW = 17.0-YHLNG
C.....
C.....  IF OPTION IS "NEITHER"  SKIP THE BORDER PLOT ROUTINES
C.....  IF OPTION IS "MAP" OR "BOTH" PLOT COMPLETE BORDERS
C.....  IF OPTION IS "BASIN" ONLY, PLOT CORNERS ONLY
C.....
      IF (KHOLNE .LT. 2) GO TO 500
      IF(KHOLNE .GT. 2) GO TO 300
      CALL PLOT(XNW+0.5,YNW,3)
      CALL PLOT(XNW,YNW,2)
      CALL PLOT(XNW,YNW+0.5,2)
      CALL PLOT(XNW+0.5,YNW-.01,3)
      CALL PLOT(XNW-.01,YNW-.01,2)
      CALL PLOT(XNW-.01,YNW+0.5,2)
  300 NDM = KHLAT
      XL = XNW+HT/2.0
      YL = YNW-10.0*HT
```

```
      CALL SYMLDM
      NDM = KHLNG
      XL = XNW-HT
      YL = YNW-4.0*HT
      CALL SYMLDM
C.....
C.....  PLOT THE NORTH EAST CORNER AND PRINT THE
C.....  LATITUDE AND LONGTITUDE OF THE CORNER
C.....
      CALL LMBCMPT(FHLAT,FLLNG,SCALE,YNE,XNE)
      YNE = 17.0-YLLNG
      XNE = 2.0
      IF(KHOLNE .GT. 2) GO TO 400
      CALL PLOT(XNE,YNE-0.5,3)
      CALL PLOT(XNE,YNE,2)
      CALL PLOT(XNE+0.5,YNE,2)
      CALL PLOT(XNE+.01,YNE-0.5,3)
      CALL PLOT(XNE+.01,YNE+.01,2)
      CALL PLOT(XNE+0.5,YNE+.01,2)
  400 NDM = KHLAT
      XL = XNE+HT/2.0
      YL = YNE + HT
      CALL SYMLDM
      NDM = LLNG
      XL = XNE-HT
      YL = YNE-4.0*HT
      CALL SYMLDM
      LNCNT = 16
C.....
C.....  IF OPTION IS "MAP" OR "BOTH" CLOSE TOP OF BORDER
C.....
      IF (KHOLNE.LE.2) GO TO 500
      CALL PLOT(XNE,YNE,3)
      CALL PLOT(XNE,YNW,2)
      CALL PLOT(XNE+.01,YNE+.01,3)
      CALL PLOT(XNE+.01,YNW-.01,2)
C.....
C.....  COMPUTE THE CENTER POINT OF THE PLOT (Y-AXIS)
C.....
  500 YCNT = (YNW - YNE) / 2.0 + YNW
C.....
C.....  COMPUTE THE BOUNDARIES AT THE SIDES OF THE PLOT
C.....
  550 IF (BNDBTM.LT.FLLAT) GO TO 3800
      JSTR = -1
      I = 0
      BNDTOP = BNDBTM
      BNDBTM = BNDBTM-BNDINC
      IF (BNDBTM.LT.FLLAT) BNDBTM = FLLAT-.0001
      CALL LMBCMPT(BNDBTM,CM,SCALE,YINC,XINC)
      XF = XINC -XHLAT+2.0
      IF (TLA.LT.BNDBTM) GO TO 700
      TLNP = TLN
      JSTR = 0
```

```
C.....
C.....  COMPUTE THE LOCATION OF TIC MARKS AND STORE THEM IN THE PLOT TABLE
C.....
  600 IF (TLNP.LT.FLLNG) GO TO 700
      I = I+1
      CALL LMBCMPT(TLA,TLNP,SCALE,YLNP,XLNP)
      YS(I) = 17.0-YLNP
      XS(I) = XLNP-XHLAT+2.0
      KSYMS(I) = 0
      TLNP = TLNP-FITIC
      IF (TLNP.LT.FLLNG) TLA = TLA-FITIC
      GO TO 600
  700 IF (KHOLNE.LE.1) GO TO 900
      IF (KHOLNE.EQ.2) GO TO 800
      I = I+1
      JSTR = 0
C.....
C.....  COMPUTE THE LEFT HAND BORDER AND STORE IT IN THE PLOT TABLE
C.....
      CALL LMBCMPT(BNDTOP,FHLNG,SCALE,YCM,XCM)
      XS(I) = XCM-XHLAT+2.0
      YS(I) = 17.0-YCM
      KSYMS(I) = 1
      CALL LMBCMPT(BNDBTM,FHLNG,SCALE,YCM,XCM)
      XLS(I) = XCM-XHLAT+2.0
      YLS(I) = 17.0-YCM
      I = I+1
C.....
C.....  COMPUTE THE RIGHT HAND BORDER AND STORE IT IN THE PLOT TABLE
C.....
      CALL LMBCMPT(BNDTOP,FLLNG,SCALE,YCM,XCM)
      XS(I) = XCM-XHLAT+2.0
      YS(I) = 17.0-YCM
      KSYMS(I) = 2
      CALL LMBCMPT(BNDBTM,FLLNG,SCALE,YCM,XCM)
      XLS(I) = XCM-XHLAT+2.0
      YLS(I) = 17.0-YCM
      IF(KHOLNE .EQ. 3) GO TO 900
C.....
C.....  IF OPTION IS "BOTH" OR "BASIN" READ THE SLICE TABLE FROM DISK,
C.....  COMPUTE THE CORDINATES OF THE LINE SEGMENT, AND STORE THEM IN THE
C.....  PLOT TABLE
C.....
  800 IF(EOFSL .EQ. 1.0) GO TO 900
      ISLSW = 0
      IF(DLLTS .LT. BNDBTM .AND. SLCNT .GT. 0.0) GO TO 900
  805 IF(LS1 .EQ. "-") ARGA = -ARGA
      IF(LS2 .EQ. "-") ARGB = -ARGB
      IF(LS3 .EQ. "-") ARGC = -ARGC
 9022 FORMAT(1H0,F4.0,2(I6,2X,I7,2X),2(F7.4,2X,F8.4,2X),3(F10.0,2X))
      IF (DLLTS .LT. BNDBTM) GO TO 900
      I = I + 1
      CALL LMBCMPT(DLLTS,DLLNS,SCALE,YLNP,XLNP)
      YS(I) = 17.0 - YLNP
      XS(I) = XLNP - XHLAT + 2.0
```

```
      KSYMS(I) = 3
      CALL LMBCMPT(DHLTS,DHLNS,SCALE,YLNP,XLNP)
      XLS(I) = XLNP - XHLAT + 2.0
      YLS(I) = 17.0 - YLNP
      LSTSLC = LSCR
      LSCR = LLATS
      IF(NSLC .EQ. 1) NSLC = 2
      IF(NSLC .EQ. 2) NSLC = 1
      JSTR = 0
 9013 FORMAT(1H0,10X,7HNWELLS=,I5,5X,7H KGOSW=,I5)
      READ(KDISK,9021)LLATS,LLNGS,LHLTS,LHLNS,DLLTS,DLLNS,DHLTS,DHLNS,
     $LS1,ARGA,LS2,ARGB,LS3,ARGC
      SLCNT = SLCNT + 1.0
 9021 FORMAT(2(I6,I7),2(F6.4,1X,F7.4),3(A1,F8.0))
      IF(EOF(KDISK))802,803
  802 EOFSL = 1.
      GO TO 900
  803 IF(I .LE. 250) GO TO 800
      ISLSW = 1
      GO TO 900
C.....
C.....  IF WELLS ARE TO BE PLOTTED ENTER THE SELECTION ROUTINE
C.....  OTHERWISE GO TO THE SORT ROUTINE
C.....
  900 IF (NWELLS) 2400,1000,1000
C.....
C.....  IF "GO" SWITCH IS ZERO OR POSITIVE -- GO TO THE TEST FOR A FULL TA
C.....
 1000 IF(KGOSW)1100,1600,1600
C.....
C.....  READ IN THE FIRST SET OF SITES TO BE PLOTTED
C.....  THIS IS THE ONLY READ ROUTINE THAT IS USED FOR
C.....  READING THE SITE DATA FILE WHETHER THE FILE IS
C.....  COMING FROM THE INPUT STREAM OR FROM A DISK FILE
C.....
 1100 READ(NFILE,9003)NSEQ,LAD,LAM,LAS,LOD,LOM,LOS,KDUP,(LCWLID(JK),
     $JK=1,20),DEPTH,ALT,CNSYR,INWU,(LOGS(JK),JK=1,8),TPOPN,MOPN,CSDM
     $,MFIN,WL,WTLYR,WLCNT,SPCP
 9003 FORMAT(I4,3I2,I3,2I2,I2,20A1,F5.0,1X,F5.0,9X,F4.0,A1,8A1,F7.2,A1
     $,F5.2,A1,F7.2,F4.0,1X,I2,F5.2,15X)
C.....
C.....  AT THE FIRST END OF FILE THE UNIT NUMBER IS CHANGED FROM 5 TO 16
C.....  TO ALLOW READING FROM A DISK FILE, THEN THE UNIT NUMBER IS
C.....  STEPPED BY 1 FOR EACH END OF FILE ON UNIT 5 "INPUT" ALSO THE
C.....  EOFSW IS SET TO 0.0
C.....
      IF(EOF(NFILE))4300,1350
 1300 READ(NDISK,9003)NSEQ,LAD,LAM,LAS,LOD,LOM,LOS,KDUP,(LCWLID(JK),
     $JK=1,20),DEPTH,ALT,CNSYR,INWU,(LOGS(JK),JK=1,8),TPOPN,MOPN,CSDM
      IF(EOF(NDISK))4300,1350
```

```
C.....
C.....  TEST TO CHECK IF THE SITE IS WITHIN THE PLOT LIMITS
C.....
 1350 FLAT=FLOAT(LAD)+FLOAT(LAM)/60.+FLOAT(LAS)/3600.
      FLNG=FLOAT(LOD)+FLOAT(LOM)/60.+FLOAT(LOS)/3600.
      KEYID(1)=LAD*10000
      KEYID(1)=KEYID(1)+LAM*100
      KEYID(1)=KEYID(1)+LAS
      KEYID(2)=LOD*10000
      KEYID(2)=KEYID(2)+LOM*100
      KEYID(2)=KEYID(2)+LOS
 1400 KGOSW=+1
      IF(FLAT.GT.FHLAT.OR.FLAT.LT.FLLAT)GO TO 1100
      IF(FLNG.GT.FHLNG.OR.FLNG.LT.FLLNG)GO TO 1100
      ISTSW = 0
      CALL SLCTN
      BTMALT=ALT-DEPTH
      WLALT=ALT-WL
      TOPALT=ALT-TPOPN
      IF(ISTSW.NE.1)GO TO 1100
      ISEQ = ISEQ+1
C.....
C.....  INSERT WRITE STATEMENT HERE TO LIST SELECTED SITES
C.....
      WRITE(6,9004)ISEQ,NSEQ,LAD,LAM,LAS,LOD,LOM,LOS,KDUP,(LCWLID(JK),
     $JK=1,20),DEPTH,ALT,CNSYR,INWU,(LOGS(JK),JK=1,8),TPOPN,MOPN,CSDM
     $,MFIN,WL,WTLYR,WLCNT,SPCP
 9004 FORMAT(1H ,5X,I4,1X,I4,1X,3I2,1X,I3,3I2,2X,20A1,2(1X,F6.0),1X
     $,F5.0,1X,A1,2X,8A1,1X,F6.0,1X,A1,F6.2,1X,A1,1X,F8.2,1X,F5.0
     $,I4,1X,F6.2)
      SEQ=SEQ+1.0
C.....
C..... CONVERT MAP COORDINATES TO PLOTTER COORDINATES
C.....
 1500 CALL LMBCMPT(FLAT,FLNG,SCALE,Y,X)
      Y=17.0-Y
      X=X-XHLAT+2.0
C.....
C.....  PLOTTING IS DONE IN ONE INCH WIDE BANDS OR LESS
C.....  FIRST TEST IS DONE TO CHECK IF PAST THE ONE INCH MARK
C.....  SECOND TEST IS FOR TABLE FULL CONDITION  IF THE TABLE IS FULL
C.....  THE BAND IS REDUCED IN LENGTH TO ITS PRESENT POSITION AND THE
C.....  CURRENT BAND IS PLOTTED
C.....
 1600 IF(X.GE.XE)GO TO 2400
      IF(I.GE.298)GO TO 2300
 1700 I=I+1
C.....
C.....  NEXT SECTION STORES THE POINTS TO BE PLOTTED IN A TABLE UNTIL
C.....  ALL POINTS FOR A GIVEN BAND HAVE BEEN READ IN OR THE TABLE IS
C.....  FULL   JSTR IS SET TO 0 TO INDICATE THAT A POINT HAS BEEN STORED
C.....
      XS(I)=X
      YS(I)=Y
      KSYMS(I)=4
```

```
      IF(NID.EQ.0)GO TO 1900
      CALL SITLBL
 1900 JSTR = 0
      ISEQ=IFIX(SEQ)
      LNCNT=LNCNT+1
      IF(LNCNT.GE.60)GO TO 4200
      GO TO 1100
C.....
C..... ONCE THE STRIP IS COMPLETE, THE TABLE IS SORTED BY
C..... THE LONGITUDE COORDINATE TO REDUCE PEN TRAVEL
C.....
 2300 XE = XS(I)
      BNDBTM = BNDBTM+BNDINC/2.0
 2400 J = I-1
      JSW = 0
      DO 2600 JE=1,J
      IF (YS(JE).LE.YS(JE+1)) GO TO 2600
      XT = XS(JE)
      YT = YS(JE)
      TLAB1 = OLBL1(JE)
      TLAB2 = OLBL2(JE)
      KSYMT = KSYMS(JE)
      XLT = XLS(JE)
      YLT = YLS(JE)
      XS(JE) = XS(JE+1)
      YS(JE) = YS(JE+1)
      OLBL1(JE) =OLBL1(JE+1)
      OLBL2(JE) = OLBL2(JE+1)
      KSYMS(JE) = KSYMS(JE+1)
      XLS(JE) = XLS(JE+1)
      YLS(JE) = YLS(JE+1)
      XS(JE+1) =XT
      YS(JE+1) = YT
      OLBL1(JE+1) = TLAB1
      OLBL2(JE+1) = TLAB2
      KSYMS(JE+1) = KSYMT
      XLS(JE+1) = XLT
      YLS(JE+1) = YLT
      JSW = -1
 2600 CONTINUE
      IF (JSW) 2400,2800,2800
C.....
C.....  THE FOLLOWING FIVE STATEMENTS MAY BE USED FOR DEBUGGING
C.....
C       IF(JSW) 2400,2650,2650
C 2650 DO 2700 J=1,I
C       WRITE(6,9006)I,J,KSYMS(J),XS(J),YS(J),XLS(J),YLS(J)
C 9006 FORMAT(" ",3I10,4F12.4)
C 2700 CONTINUE
C.....
C..... PEN LOCATION IS CHECKED AND PLOTTING IS DONE FROM
C..... THE PEN LOCATION
C.....
 2800 CALL WHERE(RXPAGE,RYPAGE,RFACT)
      IF (RYPAGE-17.0) 3000,2900,2900
```

```
 2900 IJ = I +1
      IE = 0
      INC = -1
      GO TO 3100
 3000 IJ = 0
      IE = I+1
      INC = 1
 3100 IJ = IJ + INC
      IF (IJ.EQ.IE) GO TO 550
 3150 IF (KSYMS(IJ)) 3100,3300,3400
 3200 CALL SYMBOL(XS(IJ),YS(IJ),0.06,3,0.0,-1)
      CALL NUMBER(XS(IJ)+.02+0.1,YS(IJ)+0.06,0.1,OLBL1(IJ),90.0,-1)
      GO TO 3100
C.....
C.....  PLOT THE TIC MARKS
C.....
 3300 CALL SYMBOL(XS(IJ),YS(IJ),0.3,3,0.0,-1)
      GO TO 3100
 3400 NGOTO=KSYMS(IJ)
      GO TO (3500,3550,3600,3650,3650,3650,3650,3650) NGOTO
C.....
C.....  PLOT DOUBLE THICK LINE ON THE LEFT HAND BORDER SEGMENT
C.....
 3500 CALL PLOT(XS(IJ),YS(IJ),3)
      CALL PLOT(XLS(IJ),YLS(IJ),2)
      CALL PLOT(XLS(IJ),YLS(IJ)-0.01,3)
      CALL PLOT(XS(IJ),YS(IJ)-0.01,2)
      GO TO 3100
C.....
C.....  PLOT A DOUBLE THICK LINE ON THE RIGHT HAND BORDER SEGMENT
C.....
 3550 CALL PLOT(XS(IJ),YS(IJ),3)
      CALL PLOT(XLS(IJ),YLS(IJ),2)
      CALL PLOT(XLS(IJ),YLS(IJ)+0.01,3)
      CALL PLOT(XS(IJ),YS(IJ)+0.01,2)
      GO TO 3100
C.....
C.....  PLOT THE BASIN BOUNDARY LINE SEGMENTS
C.....
 3600 IF(NBRDR .EQ. 1) GO TO 3610
      CALL PLOT(XLS(IJ),YLS(IJ),3)
      CALL PLOT(XS(IJ),YS(IJ),2)
      GO TO 3100
 3610 CALL NEWPEN(2)
      CALL PLOT(XLS(IJ),YLS(IJ),3)
      CALL PLOT(XS(IJ),YS(IJ),2)
      CALL NEWPEN(1)
      GO TO 3100
C.....
C.....  PLOT SITE SYMBOL AND LABEL
C.....
 3650 CALL SYMBOL(XS(IJ),YS(IJ),ZHT,1,90.0,-1)
      IF(NID.EQ.0)GO TO 3100
      IF(NID.EQ.2)GO TO 3675
      CALL NUMBER(XS(IJ)+0.12,YS(IJ),SYHT,OLBL1(IJ),90.0,-1)
      GO TO 3100
```

```
 3675 IF(NID1.EQ.7)GO TO 3680
      SIHT=SYHT+0.02
      CALL NUMBER(XS(IJ)-0.02,(YS(IJ)+SIHT),SYHT,OLBL1(IJ)
     $,90.0,-1)
      GO TO 3690
 3680 SIHT=SYHT+0.02
      CALL NUMBER(XS(IJ)+0.12,(YS(IJ)-SIHT),SYHT,OLBL1(IJ),90.0,2)
 3690 CALL PLOT(XS(IJ),YS(IJ)+0.1,3)
      SIHT=SYHT*5.0
      CALL PLOT(XS(IJ),YS(IJ)+SIHT+0.1,2)
      IF(NID2.EQ.7)GO TO 3695
      SIHT=SYHT+0.02
      CALL NUMBER(XS(IJ)+0.1,YS(IJ)+SIHT,SYHT,OLBL2(IJ),90.0,-1)
      GO TO 3100
 3695 SIHT=SYHT+0.02
      CALL NUMBER(XS(IJ)+0.1,(YS(IJ)+SIHT),SYHT,OLBL2(IJ),90.0,2)
      GO TO 3100
C.....
C.....  PLOT SW CORNER AND PRINT COORDINATES
C.....
 3800 CALL LMBCMPT(FLLAT,FHLNG,SCALE,Y,X)
      Y = 17.0 -Y
      X = X-XHLAT+2.0
      IF(KHOLNE .GT. 2) GO TO 3900
      IF (KHOLNE.LT.2) GO TO 3900
      CALL PLOT(X,Y+.5,3)
      CALL PLOT(X,Y,2)
      CALL PLOT(X-.5,Y,2)
      CALL PLOT(X+0.01,Y+.5,3)
      CALL PLOT(X+0.01,Y-0.01,2)
      CALL PLOT(X-.5,Y-0.01,2)
 3900 TLNP = TLN
      NDM = LLAT
      XL = X+HT/2.0
      YL = Y -10.0*HT
      CALL SYMLDM
      NDM = KHLNG
      XL = X + 2.0*HT
      YL = Y-(4.0*HT)
      CALL SYMLDM
C.....
C.....  PRINT INFORMATION BLOCK AT SW CORNER
C.....
      XL = X+4.0*HT
      YL = Y
      CALL SYMBOL(XL,YL,XHT,LPROJ,90.0,40)
      XL = XL+2.0*XHT
      YL = Y
      CALL SYMBOL(XL,YL,XHT,LCMS,90.0,20)
C.....
C.....  PRINT CENTRAL MERIDIAN
C.....
      NDM = NCM
      YL = YL+XHT*25.0
      CALL SYMLDM
```

```
      XL = XL+2.0*XHT
      YL = Y
      CALL SYMBOL(XL,YL,XHT,NSTDP,90.0,20)
      YL = Y+21.0*XHT
      NDM = LLAT
      CALL SYMLDM
      YL = Y+31.0*XHT
      CALL SYMBOL(XL,YL,XHT,"AND",90.0,3)
      NDM = KHLAT
      YL = Y+37.0*XHT
      CALL SYMLDM
C.....
C.....  IF OPTION IS "MAP" OR "BOTH" CLOSE THE BOTTOM OF THE BORDER
C.....
      IF (KHOLNE.LE.2) GO TO 4000
      CALL LMBCMPT(FLLAT,FHLNG,SCALE,Y1,X1)
      Y1 = 17.0 -Y1
      X1 = X1-XHLAT+2.0
      CALL PLOT(X1,Y1,3)
      CALL LMBCMPT(FLLAT,FLLNG,SCALE,Y,X)
      Y = 17.0 -Y
      X = X-XHLAT+2.0
      CALL PLOT(X,Y,2)
 4000 CALL LMBCMPT(FLLAT,FLLNG,SCALE,Y,X)
      Y = 17.0-Y
      X = X-XHLAT+2.0
      IF(KHOLNE .GT. 2) GO TO 4150
      IF (KHOLNE.LT.2) GO TO 4150
      CALL PLOT(X,Y-.5,3)
      CALL PLOT(X,Y,2)
      CALL PLOT(X-.5,Y,2)
      CALL PLOT(X+0.01,Y-.5,3)
      CALL PLOT(X+0.01,Y-0.01,2)
      CALL PLOT(X-.5,Y-0.01,2)
 4150 IF(KHOLNE .LT. 3) GO TO 4100
      CALL PLOT(X,Y,3)
      CALL PLOT(X+0.01,Y-0.01,2)
 4100 NDM = LLAT
      XL = X+HT/2.0
      YL = Y + HT
      CALL SYMLDM
      NDM = LLNG
      XL = X + 2.0*HT
      YL = Y-(4.0*HT)
      CALL SYMLDM
C.....
C.....  CALL ROUTINE TO OUTPUT MILEAGE SCALE
C.....
      XLM = XL +1.0
      YLM = 17.0
      CALL SCMLE(XLM,YLM,SCALE)
      HT = 0.14
      X = XL+2.0
      Y = 17.0-6.0*HT
      xCNTR = XL+3.5
```

```
      YCNTR = 17.0
      CALL SYMBOL(XCNTR+2.0,7.5,0.24,TITLE,90.0,78)
      CALL SYMBOL(XCNTR+3.0,18.4,0.24," END  PLOT",270.0,10)
      CALL NUMBER(999.0,15.8,0.24,8PLOT,270.0,-1)
      CALL WHERE(XCORD,YCORD,RFAC)
      XNEW = XCORD+4.0
      CALL PLOT(XNEW,0.0,-3)
      GO TO 100
4200  IPAGE = IPAGE+1
      WRITE(6,9007)TITLE,IPAGE,NPLOT,DATE(QQ)
9007  FORMAT(1H1,10X,4HUSGS,14X,7A10,A8,19X,4HPAGE,I3
     1,/1H0,10X,4HPLOT,I2,37X,24HSWABRASA WELL SITE PLOTS,44X,A10)
      WRITE(6,9008)
9008  FORMAT(1H0,5X,3HNEW,2X,3HRPT,72X,6HTOP OF,4H TYP,9H CSNG TYP,3X
     1,12HWATER LEVELS)
      WRITE(6,9121)
9121  FORMAT(1H0,5X,3HSEQ,2X,3HSEQ,4X,3HLAT,4X,4HLONG,8X,8HLOCAL ID,7X
     $,5HDEPTH,4X,3HALT,2X,5HCNSYR,1X,3HUSE,3X,4HLOGS,2X,7HOPENING,1X
     $,3HOPN,1X,4HDIAM,1X,3HFIN,1X,4HMEAS,1X,14HYEAR CNT SPCAP)
      LNCNT = 6
      WRITE(6,9122)
9122  FORMAT(1H ,5H          )
      GO TO 1100
4300  IF(KGOSW.GE.0)NWELLS=-1
      IF(KGOSW)4400,2400,2400
4400  GO TO 100
4500  IF (NPLOT.GT.0) GO TO 100
      WRITE(6,9009)
9009  FORMAT("0","EITHER THE DATA FILE WAS EMPTY OR SITES OUT OF RANGE")
      STOP
4600  STOP
4700  WRITE(6,9010)YTEST,YHLNG,YLLNG
9010  FORMAT(" ","JOB ABORTED---TOO LARGE",3F12.4)
      STOP
4800  WRITE(6,9011)DIFF,XCM,XHLAT
9011  FORMAT(" ","STRIP CONTROL OUT OF BOUND",3F12.4)
      STOP
4900  WRITE(6,9012)KEYID,KEYIDL
9012  FORMAT(1H ,I6,I7,I2,5X,I6,21H FILE OUT OF SEQUENCE,85X)
      STOP
      END
```

```
      SUBROUTINE NREAD3
      COMMON/MALINE/JSTR,TITLE(8),XS(300),YS(300)
     $,KSYMS(300),XLS(300),YLS(300),FHLAT,FLLAT,FHLNG,FLLNG,CM,ZHT
     $,SYHT,IPAGE,NPLOT,SCALE,XHLAT,YHLNG,XLLAT,YLLNG,TITLE2(8)
     $,TITLE3(8),TLA,TLN,FITIC,LLAT,KHLAT,LLNG,KHLNG,KHOLNE,LOP,NCM
     $,TITLE4(8),TITLE5(8),TITLE6(8),TITLE7(8),NF,NOTN(20)
     $,NDS,NBRDR
C.....
      COMMON/LBLR/I,NID,NID1,NID2,OLBL1(300),OLBL2(300),BTMALT,WLALT
     $,TOPALT
C.....
      COMMON/SELCT/CNYR1,CNYR2,CDIAM,ILOG(8),IFIN(3),IOPN(3),IWUSE(3)
     $,ISTSW,NFIN,NLOG,NOPN,NWLVL,NWUS,NWELLS,SPCAP,TSTDEP,TSTOPN,TSTWL
     $,TSTWDP,TSTDIM,WLYR1,WLYR2,WTRDEP,WTRLVL,XDEPTH,OPNTOP
C.....
      DATA ZHT,SYHT/0.06,0.06/
      DATA (NOTN(I),I=1,13)/10H SEQUENCE ,10HWELL DEPTH,10H ALTITUDE ,
     $10HYR OF CNST,10H WATER LVL,10HWTR LVL YR,10HSPEC CPCTY,
     $10H TOP DEPTH,10HCASNG DIAM,10H NUM WLVLS,10HALT BOTTOM,
     $10HALT WTRLVL,10HALT TP OPN/
C.....
C..... CONTROL CARDS ARE AS FOLLOWS
C..... TITLE I2,7A10,A8
C..... LATITUDE(LOW AND HIGH) LONGITUDES (LOW AND HIGH)
C..... PROJECTION AND SCALE DATA
C.....
      IPAGE = IPAGE + 1
C.....
C..... READ TITLE RECORD AND PRINT FIRST HEADING LINE
C.....
      READ(5,9010)IDC1,TITLE
      IF (EOF(5)) 9901,100
  100 IF (IDC1.NE.1) GO TO 9801
      NPXX = NPLOT +1
      WRITE(6,9020)TITLE,IPAGE,NPXX,DATE(QQ)
C.....
C..... READ AND PRINT PLOT COORDINATES LIMITS
C.....
      READ(5,9030)IDC2,LLAD,LLAM,LLAS,LLND,LLNM,LLNS,KHLAD,KHLAM,KHLAS
     $,KHLND,KHLNM,KHLNS,KCMD,KCMM,KCMS,KHPROJ,KSC,NWELLS,NBRDR,
     $NHT,NSHT,ISYM
      IF (EOF(5)) 9901,200
  200 IF (IDC2.NE.2) GO TO 9802
      IF (LLAD.LT.30.OR.LLAD.GT.37) GO TO 9821
      IF (KHLAD.LT.30.OR.KHLAD.GT.37) GO TO 9822
      IF (LLND.LT.107.OR.LLND.GT.116)GO TO 9823
      IF (KHLND.LT.107.OR.KHLND.GT.116) GO TO 9824
      IF (LLND.GT.KHLND) GO TO 9825
      IF (LLAD.GT.KHLAD) GO TO 9826
C.....
C.....  CONVERT NOTATION HEIGHT TO DECIMAL INCHES
      IF(NHT .LE. 0) GO TO 210
      ZHT = NHT
      ZHT = ZHT * 0.01
```

```
  210 IF(NSHT .LE. 0) GO TO 220
      SYHT = NSHT
      SYHT = SYHT * 0.01
C.....
C.....  CONVERT DEGREES, MINUTES, SECONDS TO DECIMAL LONGITUDES AND
C.....  LATITUDES
C.....
  220 LLAT=LLAD*10000+LLAM*100+LLAS
      KHLAT=KHLAD*10000+KHLAM*100+KHLAS
      LLNG=LLND*10000+LLNM*100+LLNS
      KHLNG=KHLND*10000+KHLNM*100+KHLNS
C.....
C.....  CONVERT LOW LATITUDE TO DECIMAL DEGREES
C.....
      FLLAD=LLAD
      FLLAM=LLAM
      FLLAS=LLAS
      FLLAT=FLLAD+FLLAM/60.+FLLAS/3600.
C.....
C.....  CONVERT HIGH LATITUDE TO DECIMAL DEGREES
C.....
      FHLAD=KHLAD
      FHLAM=KHLAM
      FHLAS=KHLAS
      FHLAT=FHLAD+FHLAM/60.+FHLAS/3600.
C.....
C.....  CONVERT LOW LONGITUDE TO DECIMAL DEGREES
C.....
      FLLND=LLND
      FLLNM=LLNM
      FLLNS=LLNS
      FLLNG=FLLND+FLLNM/60.+FLLNS/3600.
C.....
C..... CONVERT HIGH LONGITUDE TO DECIMAL DEGREES
C.....
      FHLND=KHLND
      FHLNM=KHLNM
      FHLNS=KHLNS
      FHLNG=FHLND+FHLNM/60.+FHLNS/3600.
C.....
C..... PROJECTION AND SCALE DATA
C.....
C.....  CONVERT INPUT CENTRAL MERIDIAN TO DECIMAL DEGREES
C.....
      IF (KCMD.EQ.0) GO TO 300
      CMD = KCMD
      CMM = KCMM
      CMM = CMM/60.
      CMS = KCMS
      CMS = CMS/3600.0
      CM = CMD+CMM+CMS
      GO TO 310
```

```
C.....
C.....  IF NO CENTRAL MERIDIAN IS INPUT, COMPUTE CENTRAL MERIDIAN
C.....
 300  CM = (FLLNG+FHLNG)/2.0
      KCMD = CM
      CMD = KCMD
      CMT = CM-CMD
      CMM = CMT*60.0
      KCMM = CMM
      CMT = KCMM
      CMS = (CMM-CMT)*60.0
      KCMS = CMS
      GO TO 310
C.....
C.....  IF SCALE IS INPUT, CONVERT TO DECIMAL
C.....
 310  IF (KSC.GT.0) SCALE = KSC
      NCM = KCMD*10000+KCMM*100+KCMS
C.....
C.....  DETERMINE WHETHER PROJECTION IS LAMBERT CONFORMAL OR MERCATOR
C.....
      IF (KHPROJ.EQ.1) GO TO 320
      IF (KHPROJ.EQ.2) GO TO 330
  320 PROJ1 = "LAMBERT CO"
      PROJ2 = "NFORMAL    "
      GO TO 400
  330 PROJ1 = "MERCATOR   "
      PROJ2 = "           "
      GO TO 400
C.....
C.....  WRITE PLOT LIMITS AND PARAMTERS
C.....
  400 WRITE(6,9040)
      WRITE(6,9050)PROJ1,PROJ2
      WRITE(6,9060)KS    C.....
      WRITE(6,9070)LLAD,LLAM,LLAS,KHLAD,KHLAM,KHLAS
      WRITE(6,9080)LLND,LLNM,LLNS,KHLND,KHLNM,KHLNS
      WRITE(6,9090)KHLAD,KHLAM,KHLAS,KHLND,KHLNM,KHLNS,KHLAD,KHLAM,
     $KHLAS,LLND,LLNM,LLNS
      WRITE(6,9100)LLAD,LLAM,LLAS,KHLND,KHLNM,KHLNS,LLAD,LLAM,LLAS,
     $LLND,LLNM,LLNS
      WRITE(6,9110)KCMD,KCMM,KCMS
      WRITE(6,9130)SYHT
      WRITE(6,9140)ZHT
      IF(NWELLS .GE. 0) WRITE(6,9150)NWELLS
      IF(NWELLS .LT. 0) WRITE(6,9160)NWELLS
      IF(NBRDR .NE. 1) WRITE(6,9170)
      IF(NBRDR .EQ. 1) WRITE(6,9180)
C.....
C.....  READ TIC MARKS AND PERIMETER OPTIONS
C.....
      READ(5,9190)IDC3,ATIC1,ATIC2,ITIC,ATIC3,ATIC4,INTIC
     $,ATIC5,ATIC6,NOLNE
      IF (EOF(5)) 9803,500
  500 IF (IDC3.NE.3) GO TO 9803
```

```
C.....
C.....  CONVERT TIC MARK INTERVAL TO DECIMAL
C.....
      FITIC=ITIC
      FITIC=FITIC/60.
      FINTIC=INTIC
      FINTIC=FINTIC/60.
C.....
C.....  DETERMINE THE TYPE OF MAP OR BASIN PLOT OPTION TO BE USED
C.....
      IF (NOLNE.EQ."NEITHER") KHOLNE = 1
      IF (NOLNE.EQ."BASIN  ") KHOLNE = 2
      IF (NOLNE.EQ."MAP    ") KHOLNE = 3
      IF (NOLNE.EQ."BOTH   ") KHOLNE = 4
      IF (NOLNE.EQ."       ") KHOLNE = 1
      WRITE(6,9200)ATIC1,ATIC2,ITIC,ATIC3,ATIC4,INTIC,ATIC5,ATIC6,NOLNE
      MHLAM =KHLAM
      IF (INTIC.EQ.0) GO TO 520
      DO 510 IXK=1,60
      IIN=MHLAM/INTIC
      IF (MHLAM-(IIN*INTIC).EQ.0) GO TO 520
  510 MHLAM = MHLAM -1
  520 TLD =KHLAD
      TLM =MHLAM
      TLA = TLD + TLM/60.
      MHLNM = KHLNM
      DO 530 IXK=1,60
      IF (INTIC.EQ.0) GO TO 540
      IIN = MHLNM/INTIC
      IF (MHLNM-(IIN*INTIC).EQ.0) GO TO 540
  530 MHLNM = MHLNM -1
  540 TLND = KHLND
      TLNM = MHLNM
      TLN = TLND + TLNM/60.
C.....
C.....  READ OPTION CONTROL CARD # 4
C.....
      READ(5,9210)IDC4,NID,NID1,NID2,CNYR1,CNYR2,WLYR1,WLYR2,
     $XDEPTH,WTRLVL,OPNTOP,NOPN,(IOPN(I),I=1,3),NLOG,(ILOG(N),N=1,8)
     $,NWUS,(IWUSE(N),N=1,3),NFIN,(IFIN(N),N=1,3),SPCAP,NWLVL,WTRDEP
     $,CDIAM
      IF(EOF(5))9800,600
  600 IF(IDC4 .NE. 4) GO TO 9804
      TSTDEP = ABS(XDEPTH)
      TSTOPN = ABS(OPNTOP)
      TSTWL = ABS(WTRLVL)
      TSTWDP=ABS(WTRDEP)
      TSTDIM=ABS(CDIAM)
      WRITE(6,9020)TITLE,IPAGE,NPXX,DATE(QQ)
      WRITE(6,9220)
      IF(NWELLS)610,620,620
  610 WRITE(6,9230)
      GO TO 1700
```

```
C.....
C.....  EXAMINE SITE IDENTIFIER OPTIONS
C.....
  620 IF(NID .EQ. 0) WRITE(6,9240)
      IF(NID .NE. 1) GO TO 630
      IF(NID1 .LE. 0 .OR. NID1 .GT. 13) GO TO 9830
      WRITE(6,9250)NOTN(NID1)
  630 IF(NID .LT. 2) GO TO 635
      IF(NID .NE. 2) GO TO 9831
      IF(NID1 .LE. 0 .OR. NID1 .GT. 13) GO TO 9830
      IF(NID2 .LE. 0 .OR. NID2 .GT. 13) GO TO 9832
      WRITE(6,9260)NOTN(NID1),NOTN(NID2)
C.....
C.....  EXAMINE TIME SPAN RANGE - CONSTRUCTION YEAR
C.....
  635 IF(CNYR1 .LE. 0.0 .AND. CNYR2 .LE. 0.0) WRITE(6,9270)
      IF(CNYR1 .LE. 0.0 .AND. CNYR2 .GT. 0.0) WRITE(6,9280)CNYR2
      IF(CNYR1 .GT. 0.0 .AND. CNYR2 .LE. 0.0) WRITE(6,9290)CNYR1
      IF(CNYR1.GT.0.0.AND.CNYR2.GT.0.0.AND.CNYR1.GT.CNYR2)GO TO 9833
      IF(CNYR1 .GT. 0.0 .AND. CNYR2 .GT. 0.0) WRITE(6,9300)CNYR1,CNYR2
C.....
C.....  EXAMINE TIME SPAN RANGE - YEAR OF WATER LEVEL MEASUREMENT
C.....
      IF(WLYR1 .LE. 0.0 .AND. WLYR2 .LE. 0.0) WRITE(6,9310)
      IF(WLYR1 .LE. 0.0 .AND. WLYR2 .GT. 0.0) WRITE(6,9320)WLYR2
      IF(WLYR1 .GT. 0.0 .AND. WLYR2 .LE. 0.0) WRITE(6,9330)WLYR1
      IF(WLYR1.GT.0.0.AND.WLYR2.GT.0.0.AND.WLYR1.GT.WLYR2)GO TO 9834
      IF(WLYR1 .GT. 0.0 .AND. WLYR2 .GT. 0.0) WRITE(6,9340)WLYR1,WLYR2
      WRITE(6,9345)
C.....
C.....  EXAMINE DEPTH OF WELL
C.....
      IF(XDEPTH) 640,650,660
  640 WRITE(6,9350) TSTDEP
      GO TO 670
  650 WRITE(6,9360)
      GO TO 670
  660 WRITE(6,9370) TSTDEP
C.....
C.....  EXAMINE WATER LEVEL MEASUREMENTS
C.....
  670 IF(WTRLVL) 680,690,700
  680 WRITE(6,9380) TSTWL
      GO TO 710
  690 WRITE(6,9390)
      GO TO 710
  700 WRITE(6,9400) TSTWL
C.....
C.....  EXAMINE DEPTH TO TOP OF THE OPENING
C.....
  710 IF(OPNTOP) 720,730,740
  720 WRITE(6,9410) TSTOPN
      GO TO 750
  730 WRITE(6,9420)
      GO TO 750
  740 WRITE(6,9430) TSTOPN
```

```
C.....
C.....  EXAMINE TYPE OF OPENING PARAMETERS
C.....
  745 IF(NOPN .GT. 3)GO TO 9840
      IF(NOPN .GT. 0)GO TO 747
      WRITE(6,9445)
      GO TO 750
  747 WRITE(6,9447)NOPN,(IOPN(I),I=1,NOPN)
C.....
C.....  EXAMINE LOG PARAMETERS
C.....
  750 IF(NLOG .GT. 8) GO TO 9835
      IF(NLOG .GT. 0) GO TO 760
      WRITE(6,9440)
      GO TO 800
  760 WRITE(6,9450)NLOG,(ILOG(I),I=1,NLOG)
C.....
C.....  EXAMINE WATER USE PARAMETERS
C.....
  800 IF(NWUS .GT. 3)GO TO 9836
      IF(NWUS .GT. 0) GO TO 810
      WRITE(6,9460)
      GO TO 820
  810 WRITE(6,9470)NWUS,(IWUSE(I),I=1,NWUS)
C.....
C.....  EXAMINE FINISH PARAMETERS
C.....
  820 IF(NFIN .GT. 3) GO TO 9837
      IF(NFIN .GT. 0) GO TO 830
      WRITE(6,9480)
      GO TO 840
  830 WRITE(6,9490)NFIN,(IFIN(I),I=1,NFIN)
C.....
C.....  EXAMINE SPECIFIC CAPACITY PARAMETER
C.....
  840 IF(SPCAP .GT. 9999.99) GO TO 9838
      IF(SPCAP .GT. 0.0) GO TO 850
      IF(SPCAP .EQ. 0.0) GO TO 860
      IF(SPCAP .LT. 0.0) GO TO 870
  850 WRITE(6,9500)SPCAP
      GO TO 880
  860 WRITE(6,9510)
      GO TO 880
  870 WRITE(6,9520)SPCAP
C.....
C.....  EXAMINE WATER LEVEL COUNT PARAMETER
C.....
  880 IF(NWLVL .GT. 999) GO TO 9839
      IF(NWLVL .GT. 0) GO TO 890
      WRITE(6,9530)
      GO TO 900
  890 WRITE(6,9540)NWLVL
  900 IF(WTRDEP.EQ.0.0)GO TO 930
      IF(WTRDEP.GT.0.0)GO TO 920
```

```
      WRITE(6,9941)TSTWDP
      GO TO 940
  920 WRITE(6,9942)TSTWDP
      GO TO 940
  930 WRITE(6,9943)
  940 IF(CDIAM .EQ.0.0) GO TO 960
      IF(CDIAM.GT.0.0) GO TO 950
      WRITE(6,9944)TSTDIM
      GO TO 970
  950 WRITE(6,9945)TSTDIM
      GO TO 970
  960 WRITE(6,9946)
  970 CONTINUE
      LNCNT = 60
 1700 RETURN
C.....
C..... ERROR STOPS
C.....
 9800 WRITE(6,9899)
 9899 FORMAT(1H1,"******  EOF WHILE READING OPTIONS AND CONTROLS ****")
      STOP
 9801 WRITE(6,9811)IDC1
 9811 FORMAT(1H1,31HFIRST CONTROL RECORD NOT 01 ID=,I2)
      STOP
 9802 WRITE(6,9812)IDC2
 9812 FORMAT(1H1,32HSECOND CONTROL RECORD NOT 02 ID=,I2)
      STOP
 9803 WRITE(6,9813)IDC3
 9813 FORMAT(1H1,31HTHIRD CONTROL RECORD NOT 03 ID=,I2)
      STOP
 9804 WRITE(6,9814)IDC4
 9814 FORMAT(1H1,31HFORTH CONTROL RECORD NOT 04 ID=,I2)
      STOP
 9821 WRITE(6,9921)LLAD
      STOP
 9822 WRITE(6,9922)KHLAD
      STOP
 9823 WRITE(6,9923)LLND
      STOP
 9824 WRITE(6,9924)KHLND
      STOP
 9825 WRITE(6,9925)LLND,KHLND
      STOP
 9826 WRITE(6,9926)LLAD,KHLAD
      STOP
 9830 WRITE(6,9930)NID1
      STOP
 9831 WRITE(6,9931)NID
      STOP
 9832 WRITE(6,9932)NID2
      STOP
 9833 WRITE(6,9933)
      STOP
```

```
 9834 WRITE(6,9934)
      STOP
 9835 WRITE(6,9935)NLOG
      STOP
 9836 WRITE(6,9936)NWUS
      STOP
 9837 WRITE(6,9937)NFIN
      STOP
 9838 WRITE(6,9938)SPCAP
      STOP
 9839 WRITE(6,9939)NWLVL
      STOP
 9840 WRITE(6,9940)NOPN
      STOP
 9901 WRITE(6,9911)
 9911 FORMAT(1H1,35HINPUT DATA FILE EMPTY* END OF RUN *)
      IF (NPLOT.EQ.0) WRITE(6,9912)
      IF (NPLOT.NE.0) WRITE(6,9913)NPLOT
 9912 FORMAT(1H0,39HNO PLOT OUTPUT  ***CONTROLS MISSING *** )
 9913 FORMAT(1H0,I2,34HPLOTS COMPLETED NORMAL TERMINATION)
      CALL WHERE(XFIN,YFIN,RFIN)
      CALL PLOT(1.0,YFIN+5.0,999)
      STOP
C.....
C.....  FORMAT STATEMENTS
C.....
 9010 FORMAT(I2,7A10,A8)
 9020 FORMAT(1H1,10X,4HUSGS,14X,7A10,A8,19X,4HPAGE,I3
 1,/1H0,10X,4HPLOT,I2,37X,25HSWAB-RASA WELL SITE PLOTS,44X,A10)
 9030 FORMAT(I2,2(3I2,I3,2I2),I3,2I2,I2,I7,5I2)
 9040 FORMAT(1H0,51X,27HPLOT OPTIONS AND PARAMETERS,44X)
 9050 FORMAT(1H0,51X,14HPROJECTION IS ,2A10)
 9060 FORMAT(1H0,51X,14HSCALE  :  1 : ,I7)
 9070 FORMAT(1H0,30X,19HLATITUDES  : LOW : ,I3,1X,2(1X,I2),10X,
     $7HHIGH : ,I3,2(1X,I2))
 9080 FORMAT(1H ,30X,19HLONGITUDES : LOW : ,I3,1X,2(1X,I2),10X,
     $7HHIGH : ,I3,2(1X,I2))
 9090 FORMAT(1H0,30X,21HPLOT LIMITS :  N.W.  ,I3,2(1X,I2),3H : ,
     $I3,2(1X,I2),8X,7HN.E. : ,I3,2(1X,I2),3H : ,I3,2(1X,I2))
 9100 FORMAT(1H ,45X,6HS.W.  ,I3,2(1X,I2),3H : ,
     $I3,2(1X,I2),8X,7HS.E. : ,I3,2(1X,I2),3H : ,I3,2(1X,I2))
 9110 FORMAT(1H ,40X,20HCENTRAL MERIDIAN :  ,I3,2(1X,I2))
 9120 FORMAT(1H ,40X,I2,38H SYMBOLS WILL BE USED AS SITE LOCATORS)
 9130 FORMAT(1H ,40X,21HSITE MARKERS WILL BE ,F4.2,17H INCHES IN HEIGHT)
 9140 FORMAT(1H ,40X,25HSITE IDENTIFIERS WILL BE ,F4.2,17H INCHES IN HEI
     $GHT)
 9150 FORMAT(1H ,40X,26HWELL SITES WILL BE PLOTTED,5X,9HNWELLS = ,I3)
 9160 FORMAT(1H ,40X,29HNO WELL SITES WILL BE PLOTTED,5X,9HNWELLS = ,I3)
 9170 FORMAT(1H ,40X,35HBASIN BOUNDARIES PLOTTED WITH PEN 1)
 9180 FORMAT(1H ,40X,35HBASIN BOUNDARIES PLOTTED WITH PEN 2)
 9190 FORMAT(I2,A10,A3,I2,A10,A8,I2,A10,1X,A10,1X,A7,14X)
 9200 FORMAT(1H ,40X,A10,A3,I2,A10,A8,I2,A10,1X,A10,1X,A7,30X)
 9210 FORMAT(I2,I1,2I2,4F4.0,3F5.0,I1,3A1,I1,8A1,2(I1,3A1),F5.2,I3
     $,F5.0,F6.3)
```

```
9220 FORMAT(1H ,52X,26HOPTIONS FOR SITE SELECTION)
9230 FORMAT(1H ,30X,43HNWELLS = -1 : ALL SELECTION OPTIONS IGNORED)
9240 FORMAT(1H ,30X,24HSITE NOTATION IS OMITTED)
9250 FORMAT(1H ,30X,21HSITE IDENTIFIER IS : ,A10)
9260 FORMAT(1H ,30X,21HSITE IDENTIFIER IS : ,A10,6H OVER ,A10)
9270 FORMAT(1H ,30X,45HYEAR OF CONSTRUCTION IS NOT USED IN SELECTION,
    $17H - NO RANGE GIVEN)
9280 FORMAT(1H ,30X,44HSELECTED SITES WILL HAVE CONSTRUCTION YEARS ,
    $5HTHRU ,F5.0)
9290 FORMAT(1H ,30X,44HSELECTED SITES WILL HAVE CONSTRUCTION YEARS ,
    $5HFROM ,F5.0,11H TO PRESENT)
9300 FORMAT(1H ,30X,44HSELECTED SITES WILL HAVE CONSTRUCTION YEARS ,
    $8HBETWEEN ,F5.0,4H AND,F5.0)
9310 FORMAT(1H ,30X,41HWATER LEVEL YEAR IS NOT USED IN SELECTION,
    $17H - NO RANGE GIVEN)
9320 FORMAT(1H ,30X,43HSELECTED SITES WILL HAVE WATER LEVEL YEARS ,
    $5HTHRU ,F5.0)
9330 FORMAT(1H ,30X,43HSELECTED SITES WILL HAVE WATER LEVEL YEARS ,
    $5HFROM ,F5.0,11H TO PRESENT)
9340 FORMAT(1H ,30X,43HSELECTED SITES WILL HAVE WATER LEVEL YEARS ,
    $8HBETWEEN ,F5.0,4H AND,F5.0)
9345 FORMAT(1H ,20X,20HPHYSICAL PARAMETERS:)
9350 FORMAT(1H ,30X,25HDEPTH OF WELL : DEPTHS OF,F6.0,8H OR LESS)
9360 FORMAT(1H ,30X,37HDEPTH OF WELL : NOT USED IN SELECTION)
9370 FORMAT(1H ,30X,25HDEPTH OF WELL : DEPTHS OF,F6.0,11H OR GREATER)
9380 FORMAT(1H ,30X,30HWATER LEVELS : MEASUREMENTS OF,F6.0,8H OR LESS)
9390 FORMAT(1H ,30X,36HWATER LEVELS : NOT USED IN SELECTION)
9400 FORMAT(1H ,30X,30HWATER LEVELS : MEASUREMENTS OF,F6.0
    $,11H OR GREATER)
9410 FORMAT(1H ,30X,32HDEPTH TO TOP OPENING : DEPTHS OF,F6.0
    $,8H OR LESS)
9420 FORMAT(1H ,30X,47HDEPTH TO TOP OF OPENING : NOT USED IN SELECTION)
9430 FORMAT(1H ,30X,32HDEPTH TO TOP OPENING : DEPTHS OF,F6.0
    $,11H OR GREATER)
9440 FORMAT(1H ,30X,33HLOG TYPE IS NOT USED IN SELECTION)
9445 FORMAT(1H ,30X,40HTYPE OF OPENING IS NOT USED IN SELECTION)
9447 FORMAT(1H ,30X,I2,34HOPENING TYPES USED FOR SELECTION :,3(1X,A1))
9450 FORMAT(1H ,30X,I2,31H LOG TYPES USED FOR SELECTION :,8(1X,A1))
9460 FORMAT(1H ,30X,34HWATER USE IS NOT USED IN SELECTION)
9470 FORMAT(1H ,30X,I2,32H WATER USES USED FOR SELECTION :,3(1X,A1))
9480 FORMAT(1H ,30X,36HFINISH TYPE IS NOT USED IN SELECTION)
9490 FORMAT(1H ,30X,I2,34H FINISH TYPES USED FOR SELECTION :,3(1X,A1))
9500 FORMAT(1H ,30X,29HSPECIFIC CAPACITY : SP CAP OF,F7.2,
    $11H OR GREATER)
9510 FORMAT(1H ,30X,42HSPECIFIC CAPACITY :  NOT USED IN SELECTION)
9520 FORMAT(1H ,30X,29HSPECIFIC CAPACITY : SP CAP OF,F7.2,
    $8H OR LESS)
9530 FORMAT(1H ,30X,42HWATER LEVEL COUNT :  NOT USED IN SELECTION)
9540 FORMAT(1H ,30X,29HWATER LEVEL COUNT : COUNTS OF,I5,11H OR GREATER)
9921 FORMAT(1H1,10X,38HLOW LATITUDE OUTSIDE STATE RANGE:30-37,
    $5X,9HLOW LAT =,I3)
9922 FORMAT(1H1,10X,39HHIGH LATITUDE OUTSIDE STATE RANGE:30-37,
    $5X,10HIGH LAT =,I3)
```

```
 9923 FORMAT(1H1,10X,42HLOW LONGITUDE OUTSIDE STATE RANGE:107-116,
     $5X,10HLOW LONG =,I4)
 9924 FORMAT(1H1,10X,43HHIGH LONGITUDE OUTSIDE STATE RANGE:107-116,
     $5X,11HHIGH LONG =,I4)
 9925 FORMAT(1H1,10X,15HLOW LONGITUDE -,I4,32H - GREATER THAN HIGH LONGI
     $TUDE -,I4)
 9926 FORMAT(1H1,10X,14HLOW LATITUDE -,I3,31H - GREATER THAN HIGH LATITU
     $DE -,I3)
 9930 FORMAT(1H1,10X,27H**** NID1 HAS INVALID VALUE)
 9931 FORMAT(1H1,10X,27H **** NID IS GREATER THAN 2)
 9932 FORMAT(1H1,10X,27H**** NID2 HAS INVALID VALUE)
 9933 FORMAT(1H1,10X,34H**** BEGINNING CONSTRUCTION YEAR -,F5.0,
     $32H - IS GREATER THAN ENDING YEAR -,F5.0,/,1H0,15X,
     $17HCORRECT AND RERUN)
 9934 FORMAT(1H1,10X,33H**** BEGINNING WATER LEVEL YEAR -,F5.0,
     $32H - IS GREATER THAN ENDING YEAR -,F5.0,/,1H0,15X,
     $17HCORRECT AND RERUN)
 9935 FORMAT(1H1,10X,28H**** NLOG IS INVALID VALUE :,I3)
 9936 FORMAT(1H1,10X,28H**** NWUS IS INVALID VALUE :,I3)
 9937 FORMAT(1H1,10X,28H**** NFIN IS INVALID VALUE :,I3)
 9938 FORMAT(1H1,10X,29H**** SPCAP IS INVALID VALUE :,F10.5)
 9939 FORMAT(1H1,10X,29H**** NWLVL IS INVALID VALUE :,I5)
 9940 FORMAT(1H1,10X,28H**** NOPN IS INVALID VALUE :,I3)
 9941 FORMAT(1H0,30X,34HDEPTH TO WATER : DEPTHS LESS THAN ,F6.0)
 9942 FORMAT(1H0,30X,37HDEPTH TO WATER : DEPTHS GREATER THAN ,F6.0)
 9943 FORMAT(1H0,30X,43HDEPTH TO WATER IS NOT A SELECTION PARAMETER)
 9944 FORMAT(1H0,30X,38HCASING DIAMETER : DIAMETERS LESS THAN ,F7.3)
 9945 FORMAT(1H0,30X,41HCASING DIAMETER : DIAMETERS GREATER THAN ,F7.3)
 9946 FORMAT(1H0,30X,44HCASING DIAMETER IS NOT A SELECTION PARAMETER)
      END
```

```
      SUBROUTINE LAMMB(ARG1,ARG2,ARG3,ARG4,ARG5)
C.....  LAMBERT CONFORMAL PROJECTION, AFTER SCHWENN AND WOLFE(U. WISC.,
C.....     CALL LMBINIT(A,B,C,D,E) -----INITIALIZES OR DEFINES PROJECTION
C.....        A=STANDARD PARALLEL NO. 1  (+ = NORTH)  EG.  +33.
C.....        B=STANDARD PARALLEL NO. 2                EG.  +45.
C.....        C=LENGTH OF SEMIMAJOR AXIS OF ELLIPSOID (EG. 6378206.4
C.....        D=LENGTH OF SEMIMINOR AXIS OF ELLIPSOID (EG. 6356583.8
C.....        E=STANDARD LONGITUDE IN DEGREES  (+ FOR WEST).
C.....     CALL LMBRST(E,DUM,DUM,DUM,DUM)  -----RESETS STANDARD LONGITUDE
C.....        E=STANDARD LONGITUDE IN DEGREES.
C.....     CALL LMBCMPT(P,Q,S,X,Y) -----CONVERTS LAT/LONG TO X/Y.
C.....    P=LATITUDE IN DEGREES   (+ FOR NORTH).
C.....    Q=LONGITUDE IN DEGREES  (+ FOR WEST ).
C.....    S=MAP SCALE      (EG.  500000.)
C.....    X=OUTPUT  X  COORDINATE   (INCREASING TO W OF STND LONGITUDE)
C.....    Y=OUTPUT  Y  COORDINATE   (INCREASING TO S OF APEX O
C.....IF OUTPUT IS DESIRED IN INCHES (FOR MAP USE), C AND D MUST BE INCHES
```

```
C.....
C.....  ENTRY LMBINIT
C.....
      ENTRY LMBINIT
          STD = ARG5
          AL11 = ARG1*.0174533
          AL22 = ARG2*.0174533
          EPS = SQRT (1.-ARG4**2/ARG3**2)
          CN1 = EPS**2
          AN1 = ARG3/SQRT (1.-CN1*SIN (AL11)**2)
          AN2 = ARG3/SQRT (1.-CN1*SIN (AL22)**2)
          CN2 = SIN (1./3600.*.0174533)
          AA1 = 1./(AN1*CN2)
          AA2 = 1./(AN2*CN2)
          CN5 = EPS/2.
          COSL1 = COS (AL11)
          P1 = 90.-ARG1
          P2 = 90.-ARG2
          CN3 = COS (P1*.0174533)
          CN4 = COS (P2*.0174533)
          Q1 = TAN (P1*.0174533/2.)*((1.+EPS*CN3)/(1.-EPS*CN3))**CN5
          Q2 = TAN (P2*.0174533/2.)*((1.+EPS*CN4)/(1.-EPS*CN4))**CN5
          ALCN = ALOG10(COSL1/COS (AL22)/AA1*AA2)/ALOG10(Q1/Q2)
          AKCN = COSL1/(AA1*CN2*ALCN*Q1**ALCN)
          RCN = ALCN*EPS/2.
      RETURN
C.....
C.....  ENTRY LMBRST
C.....
      ENTRY LMBRST
          STD = ARG1
      RETURN
C.....
C.....  ENTRY LMBCMPT
C.....
      ENTRY LMBCMPT
          AMCN = STD-ARG2
          P = (90.-ARG1)*.0174533
          CN7 = EPS*COS (P)
          BASE1 = AKCN*TAN (P/2.)**ALCN
          BASE2 = ((1.+CN7)/(1.-CN7))**RCN
          BASE3 = ALCN*AMCN*.0174533
          BASE4 = BASE1*BASE2
          ARG4 = -BASE4*SIN (BASE3)/ARG3
          ARG5 = BASE4*COS (BASE3)/ARG3
      RETURN
      END
```

```
      SUBROUTINE SLCTN
C.....
      COMMON/SELCT/CNYR1,CNYR2,CDIAM,ILOG(8),IFIN(3),IOPN(3),IWUSE(3)
     $,ISTSW,NFIN,NLOG,NOPN,NWLVL,NWUS,NWELLS,SPCAP,TSTDEP,TSTOPN,TSTWL
     $,TSTWDP,TSTDIM,WLYR1,WLYR2,WTRDEP,WTRLVL,XDEPTH,OPNTOP
C.....
      COMMON/RDR/ALT,CSDM,CNSYR,DEPTH,INWU,LOGS(8),MFIN,MOPN,SPCP
     $,TPOPN,WL,WTLYR,WLCNT
C.....
C.....  TEST CONTRUCTION YEAR WITHIN SPECIFIED RANGE
C.....
      IF((CNYR1.EQ.0).AND.(CNYR2.EQ.0))GO TO 200
      IF((CNYR1.EQ.0).AND.(CNYR2.GT.0))GO TO 110
      IF((CNYR1.GT.0).AND.(CNYR2.EQ.0))GO TO 150
      IF((CNYR1.GT.0).AND.(CNYR2.GT.0))GO TO 170
      GO TO 9900
  110 IF(CNSYR.GT.CNYR2)GO TO 9900
      ISTSW=1
      GO TO 200
  150 IF(CNSYR.LT.CNYR1)GO TO 9900
      ISTSW=1
      GO TO 200
  170 IF(CNSYR.LT.CNYR1)GO TO 9900
      IF(CNSYR.GT.CNYR2)GO TO 9900
      ISTSW=1
C.....
C.....  TEST WATER LEVEL MEASUREMENT YEAR WITHIN SPECIFIED RANGE
C.....
  200 IF((WLYR1.EQ.0).AND.(WLYR2.EQ.0))GO TO 300
      IF((WLYR1.EQ.0).AND.(WLYR2.GT.0))GO TO 210
      IF((WLYR1.GT.0).AND.(WLYR2.EQ.0))GO TO 250
      IF((WLYR1.GT.0).AND.(WLYR2.GT.0))GO TO 270
      GO TO 9900
  210 IF(WTLYR.GT.WLYR2)GO TO 9900
      ISTSW=1
      GO TO 300
  250 IF(WTLYR.LT.WLYR1)GO TO 9900
      ISTSW=1
      GO TO 300
  270 IF(WTLYR.LT.WLYR1)GO TO 9900
      IF(WTLYR.GT.WLYR2)GO TO 9900
      ISTSW=1
      GO TO 300
C.....
C.....  TEST FOR DEPTH OF WELL WITHIN SPECIFIED RANGE
C.....
  300 IF(XDEPTH)310,400,350
  310 IF(DEPTH.GT.TSTDEP)GO TO 9900
      ISTSW=1
      GO TO 400
  350 IF(DEPTH.LT.TSTDEP)GO TO 9900
      ISTSW=1
      GO TO 400
```

```
C.....
C.....  TEST DEPTH TO TOP OF OPENING FOR SPECIFIED RANGE
C.....
  400 IF(OPNTOP)410,500,450
  410 IF(TPOPN.GT.TSTOPN)GO TO 9900
      ISTSW=1
      GO TO 500
  450 IF(TPOPN.LT.TSTOPN)GO TO 9900
      ISTSW=1
      GO TO 500
C.....
C.....  TEST DEPTH TO WATER FOR SPECIFIED RANGE
C.....
  500 IF(WTRDEP)510,600,510
  510 IF(ALT.LE.0.0)GO TO 9900
      IF(WL.LE.0.0)GO TO 9900
      DPWTR=ALT-WL
      IF(WTRDEP.LT.0.0)GO TO 550
      IF(DPWTR.LT.TSTWDP)GO TO 9900
      ISTSW=1
      GO TO 600
  550 IF(DPWTR.GT.TSTWDP)GO TO 9900
      ISTSW=1
C.....
C.....  TEST FOR SPECIFIED LOG TYPES
C.....
  600 IF(NLOG.EQ.0)GO TO 700
      DO 690 J=I,NLOG
         DO 680 N=1,8
            IF(LOGS(N).EQ.ILOG(J))LGSW=1
  680    CONTINUE
  690 CONTINUE
      IF(LGSW.NE.1)GO TO 9900
      ISTSW=1
C.....
C.....  TEST FOR WATER LEVEL WITHIN SPECIFIED RANGE
C.....
  700 IF(WTRLVL)710,800,750
  710 IF(WL.GT.TSTWL)GO TO 9900
      ISTSW=1
      GO TO 800
  750 IF(WL.LT.TSTWL)GO TO 9900
      ISTSW=1
C.....
C.....  TEST FOR SPECIFIED WATER USE CODES
C.....
  800 IF(NWUS.EQ.0)GO TO 900
      DO 890 J=1,NWUS
            IF(INWU.EQ.IWUSE(J))IUSW=1
  890 CONTINUE
      IF(IUSW.NE.1)GO TO 9900
      ISTSW=1
```

```
C.....
C.....  TEST FOR TYPE OF OPENINGS SPECIFIED
C.....
  900 IF(NOPN.EQ.0)GO TO 1000
      DO 990 J=1,NOPN
            IF(MOPN.EQ.IOPN(J))IOPSW=1
  990 CONTINUE
      IF(IOPSW.NE.1)GO TO 9900
      ISTSW=1
C.....
C.....  TEST FOR SPECIFIED TYPE OF FINISH CODES
C.....
 1000 IF(NFIN.EQ.0)GO TO 1100
      DO 1090 J=1,NFIN
            IF(MFIN.EQ.IFIN(J))IFSW=1
 1090 CONTINUE
      IF(IFSW.NE.1)GO TO 9900
      ISTSW=1
C.....
C.....  TEST FOR WATER LEVEL RECORD COUNT WITHIN SPECIFIED RANGE
C.....
 1100 IF(NWLVL)1110,1200,1150
 1110 IF(WLCNT.GT.IABS(NWLVL))GO TO 9900
      ISTSW=1
      GO TO 1200
 1150 IF(WLCNT.LT.IABS(NWLVL))GO TO 9900
      ISTSW=1
C.....
C.....  TEST FOR SPECIFIC CAPACITY WITHIN SPECIFIED RANGE
C.....
 1200 IF(SPCAP)1210,1300,1250
 1210 IF(SPCP.GT.ABS(SPCAP))GO TO 9900
      ISTSW=1
      GO TO 1400
 1250 IF(SPCP.LT.ABS(SPCAP))GO TO 9900
C.....
C.....  TEST FOR CASING DIAMETER WITHIN SPECIFIED RANGE
C.....
 1300 IF(CDIAM)1310,1400,1350
 1310 IF(CSDM.GT.ABS(CDIAM))GO TO 9900
      GO TO 1400
 1350 IF(CSDM.LT.ABS(CDIAM))GO TO 9900
      ISTSW=1
 1400 LGSW=0
      IFSW=0
      IUSW=0
      IOPSW=0
      IF(ISTSW.EQ.1)GO TO 9910
 9900 ISTSW=0
 9910 RETURN
      END
```

```
      SUBROUTINE LABEL
C.....
      COMMON/RDR/ALT,CSDM,CNSYR,DEPTH,INWU,LOGS(8),MFIN,MOPN,SPCP
     $,TPOPN,WL,WTLYR,WLCNT
C.....
      COMMON/LBLR/I,NID,NID1,NID2,OLBL1(300),OLBL2(300),BTMALT,WLALT
     $,TOPALT,SEQ
C.....
      IF(NID1.EQ.1)OLBL1(I)=SEQ
      IF(NID1.EQ.2)OLBL1(I)=DEPTH
      IF(NID1.EQ.3)OLBL1(I)=ALT
      IF(NID1.EQ.4)OLBL1(I)=CNSYR
      IF(NID1.EQ.5)OLBL1(I)=WL
      IF(NID1.EQ.6)OLBL1(I)=WTLYR
      IF(NID1.EQ.7)OLBL1(I)=SPCP
      IF(NID1.EQ.8)OLBL1(I)=TPOPN
      IF(NID1.EQ.9)OLBL1(I)=CSDM
      IF(NID1.EQ.10)OLBL1(I)=WLCNT
      IF(NID1.EQ.11)OLBL1(I)=BTMALT
      IF(NID1.EQ.12)OLBL1(I)=WLALT
      IF(NID1.EQ.13)OLBL1(I)=TOPALT
      IF(NID.NE.2)GO TO 9900
      IF(NID2.EQ.1)OLBL2(I)=SEQ
      IF(NID2.EQ.2)OLBL2(I)=DEPTH
      IF(NID2.EQ.3)OLBL2(I)=ALT
      IF(NID2.EQ.4)OLBL2(I)=CNSYR
      IF(NID2.EQ.5)OLBL2(I)=WL
      IF(NID2.EQ.6)OLBL2(I)=WTLYR
      IF(NID2.EQ.7)OLBL2(I)=SPCP
      IF(NID2.EQ.8)OLBL2(I)=TPOPN
      IF(NID2.EQ.9)OLBL2(I)=CSDM
      IF(NID2.EQ.10)OLBL2(I)=WLCNT
      IF(NID2.EQ.11)OLBL2(I)=BTMALT
      IF(NID2.EQ.12)OLBL2(I)=WLALT
      IF(NID2.EQ.13)OLBL2(I)=TOPALT
 9900 RETURN
      END



      SUBROUTINE SYMLDM
      DIMENSION NDX(7)
      COMMON/SYMLD/XL,YL,KSP,HT,ANGLE,NDM
      NT = NDM/10000
      NT = NT*10000
```

```
      FND = NT/10000
      CALL NUMBER(XL,YL,HT,FND,ANGLE,-1)
      YL = YL +3.0*HT
      IF (FND.GE.100.0) YL = YL+HT
      NM = NDM-NT
      DO 31 I=1,5
      J = 6-I
      IF (I.NE.3) GO TO 30
      NDX(J) = 11
      GO TO 31
   30 NMT = NM/10
      NMT = NMT*10
      NDX(J) = NM-NMT+1
      NM = NM/10
   31 CONTINUE
      DO 41 I=1,5
      NDXK = NDX(I)
      GO TO  (10,11,12,13,14,15,16,17,18,19,20),NDXK
   10 CALL SYMBOL(XL,YL,HT,"0",ANGLE,1)
      GO TO 41
   11 CALL SYMBOL(XL,YL,HT,"1",ANGLE,1)
      GO TO 41
   12 CALL SYMBOL(XL,YL,HT,"2",ANGLE,1)
      GO TO 41
   13 CALL SYMBOL(XL,YL,HT,"3",ANGLE,1)
      GO TO 41
   14 CALL SYMBOL(XL,YL,HT,"4",ANGLE,1)
      GO TO 41
   15 CALL SYMBOL(XL,YL,HT,"5",ANGLE,1)
      GO TO 41
   16 CALL SYMBOL(XL,YL,HT,"6",ANGLE,1)
      GO TO 41
   17 CALL SYMBOL(XL,YL,HT,"7",ANGLE,1)
      GO TO 41
   18 CALL SYMBOL(XL,YL,HT,"8",ANGLE,1)
      GO TO 41
   19 CALL SYMBOL(XL,YL,HT,"9",ANGLE,1)
      GO TO 41
   20 CALL SYMBOL(XL,YL,HT," ",ANGLE,1)
   41 YL = YL +HT
      RETURN
      END
```