General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)
LACIE PERFORMANCE PREDICTOR
FINAL OPERATIONAL CAPABILITY
PROGRAM DESCRIPTION
VOLUME II

MAY 1976

Prepared for
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lyndon B. Johnson Space Center
Houston, Texas

Contract Number NAS-9-14547
This document, in three volumes, describes the FOC version of the LACIE Performance Predictor produced under Contract NAS9-14547.

# Table of Contents

<table>
<thead>
<tr>
<th>Volume I</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book I</td>
<td>EPHEMS</td>
</tr>
<tr>
<td>Book II</td>
<td>GRID</td>
</tr>
<tr>
<td>Book III</td>
<td>LUMP</td>
</tr>
<tr>
<td>Book IV</td>
<td>SEE</td>
</tr>
<tr>
<td>Volume II</td>
<td></td>
</tr>
<tr>
<td>Book V</td>
<td>SAGE</td>
</tr>
<tr>
<td>Book VI</td>
<td>SACS</td>
</tr>
<tr>
<td>Book VII</td>
<td>POUT</td>
</tr>
<tr>
<td>Volume III</td>
<td></td>
</tr>
<tr>
<td>Book VIII</td>
<td>LEM</td>
</tr>
</tbody>
</table>

---

28234-6028-RU-00
Page ii
### Table of Contents

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part I</td>
<td>Problem Description</td>
<td>1</td>
</tr>
<tr>
<td>Part II</td>
<td>Common Block Definitions</td>
<td>31</td>
</tr>
<tr>
<td>Part III</td>
<td>List of Subroutines and Subroutine Call Structure</td>
<td>43</td>
</tr>
<tr>
<td>Part IV</td>
<td>Subroutine Descriptions and Flowcharts</td>
<td>47</td>
</tr>
<tr>
<td>Part V</td>
<td>Subroutine Listings</td>
<td>70</td>
</tr>
</tbody>
</table>
PART I

PROBLEM DESCRIPTION
Problem Description for SAGE Program

1.0 SCOPE

1.1 PROGRAM CAPABILITIES

Given the swath table files, the segment set for one country and cloud cover data, this program determines how many times and under what conditions each segment is accessed by satellites as defined on the swath table. The program writes a record for each segment on the segment reference data file which contains the pertinent acquisition data. On option, a utility function can be performed to generate the weather data file from a NASA supplied tape. This option can be exercised prior to the segment processing or as a stand-alone utility.

1.2 METHOD OF PROGRAM DEVELOPMENT

This program will be developed in FORTRAN and use of machine dependent routines will be kept to a minimum. The direct access file processing is the only known problem in this area. Modular programming techniques will be used throughout to make program development, modifications and debugging easier.

1.3 OPERATIONAL ASSUMPTIONS

- A maximum of 426 days can be specified in NODAY—the number of days in the run.
- It is assumed that only 1 year of data, starting January 1st, is to be generated on the weather data file (366 days).
- The use of the weather file will be such that the look-up date will be determined modulo 1 year, i.e., given a run day of 400, then the look-up day will be \( 400 - 366 + 1 = 35 \).
- Only 1 case is run at a time.
- Only 1 country may be run at a time.
- The segment reference data file and the weather file are regenerated each time the program is run. There is no update capability.
- A maximum of 150 acquisitions for any one segment is allowed.
2.0 INPUT

There are four files and 1 card set required for input to the program if the segment reference data file is to be generated. If just the weather file is to be generated, then only the header card and NASA weather tape is required. If both modes of operation are exercised, the NASA weather tape would replace the standard weather file as an input file.

Mode 1 Inputs - Generate weather file only.
1. Header card with option set to 1.
2. NASA weather tape.

Mode 2 Inputs - Generate segment reference data file only.
1. Card set as defined below with option set to 2.
2. Swath table file.

Mode 3 Inputs - Generate both files.
The same as for Mode 2 except Item 5 is replaced with the NASA weather tape and the option flag is set to 3.

2.1 CARDS

2.1.1 List of Data Quantities

See Input Data Description sheet on Pages 4 and 5.

2.1.2 Card Formats

"SAGE" is punched in card columns 75-78 of all cards. A sequence number is punched in card columns 79-80.

See Data Card Formats sheet on Page 6 for details.

2.1.3 Deck Set Up

1. Header card - sequence 01
2. Data card - sequence 02
## List of Data Quantities

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Dimension</th>
<th>Nominal Value</th>
<th>Range</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IHEAD</td>
<td>-</td>
<td>18</td>
<td>Blanks</td>
<td></td>
<td></td>
<td>72 character case header which prints out at the top of every page.</td>
</tr>
<tr>
<td>ICASE</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0-9999</td>
<td></td>
<td>4 digit case no. to identify the printed output and the segment reference data file.</td>
</tr>
<tr>
<td>ICSESW</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0-9999</td>
<td></td>
<td>Case number identifying the swath table and reference files.</td>
</tr>
<tr>
<td>ICSESG</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0-9999</td>
<td></td>
<td>Case number identifying the segment ID file.</td>
</tr>
<tr>
<td>ISTIME</td>
<td>-</td>
<td>3</td>
<td>0</td>
<td></td>
<td></td>
<td>ISTIME(1) - (Year - 1900) ISTIME(2) - month no. ISTIME(3) - day no. Run start date</td>
</tr>
<tr>
<td>N DAYS</td>
<td>-</td>
<td>1</td>
<td>426</td>
<td>1-426</td>
<td>Days</td>
<td>No. of days to process in the run.</td>
</tr>
<tr>
<td>IOPT</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0-2</td>
<td></td>
<td>Program run option. 0 - run sage only 1 - run weather file generation utility only 2 - run both the utility and SAGE</td>
</tr>
<tr>
<td>IVEH</td>
<td>-</td>
<td>2</td>
<td>0</td>
<td>0-2</td>
<td></td>
<td>List of vehicle numbers to process in this run.</td>
</tr>
<tr>
<td>IREPT</td>
<td>-</td>
<td>1</td>
<td>F</td>
<td>T or F</td>
<td></td>
<td>Flag to indicate whether access report is to be produced. T - yes, F - no.</td>
</tr>
<tr>
<td>DECR</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0-100</td>
<td>Kilom.</td>
<td>Swath decrement</td>
</tr>
<tr>
<td>RAND</td>
<td>-</td>
<td>1</td>
<td>1.0</td>
<td></td>
<td></td>
<td>Random no. seed used to obtain daily weather data. Must be odd integer.</td>
</tr>
<tr>
<td>IGRDN</td>
<td>-</td>
<td>1</td>
<td>1600</td>
<td>1-16000</td>
<td></td>
<td>No. of grid points to be written on the weather file (debug only)</td>
</tr>
</tbody>
</table>
### Input Data Description (cont'd)

#### List of Data Quantities (cont'd)

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Dimension</th>
<th>Nominal Value</th>
<th>Range</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISWRT0*</td>
<td>-</td>
<td>1</td>
<td>1101</td>
<td>4-1101</td>
<td>-</td>
<td>The number of records written on the swath table file. This is equal to (NO OF DAYS FOR EACH VEHICLE) + 2 + NO OF VEHICLES</td>
</tr>
<tr>
<td>NLATSW</td>
<td>-</td>
<td>1</td>
<td>100</td>
<td>1-100</td>
<td>-</td>
<td>The number of latitude points written on the swath table record</td>
</tr>
</tbody>
</table>

*Not needed for UNIVAC
Card Formats

Data Card Formats

Header Card

The header information is entered in C.C. 1-72; "SAGE" is entered in C.C. 75-78, and 01 is entered in C.C. 79-80.

<table>
<thead>
<tr>
<th>C.C. 1</th>
<th>5</th>
<th>6</th>
<th>9</th>
<th>13</th>
<th>17</th>
<th>23</th>
<th>25</th>
<th>26</th>
<th>32</th>
<th>44</th>
<th>49</th>
<th>53</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICASE</td>
<td>IOPT</td>
<td>NDAYS</td>
<td>ICSESW</td>
<td>ICSESIG</td>
<td>ISTIME</td>
<td>IVEH</td>
<td>IREP</td>
<td>DECR</td>
<td>RAND</td>
<td>LGRDN</td>
<td>ISW.R.TO</td>
<td>NLATSW</td>
</tr>
<tr>
<td>14</td>
<td>11</td>
<td>13</td>
<td>14</td>
<td>14</td>
<td>312</td>
<td>211</td>
<td>L1</td>
<td>F6.2</td>
<td>F12.0</td>
<td>15</td>
<td>14</td>
<td>13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>75</th>
<th>79</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAGE</td>
<td>02</td>
</tr>
</tbody>
</table>

| A4 | 12 |
2.1.4 Rules for Entering Data on Cards

2.1.4.1 General

1. Integers must be entered right justified.
2. F format numbers must have the decimal point present, i.e., F6.2 - XXX.XX.
3. The card sequence numbers in C.C. 79-80 must be present in all data cards.

2.1.4.2 Rules for Specific Fields

- ICSESW must match the case number on the swath reference and swath table files.
- ICSESG must match the case number on the segment ID file.
- The start time in ISTIME must not be less than the earliest vehicle start date on the swath table file. (Note, if only 1 of the vehicles is to be processed, then ISTIME will be checked against that vehicle's start time only. If ISTIME is not input, it will be assumed that the earliest vehicle start date is the run start date.
- NDAYS must be ≥ 1 and ≤ 426.
- LVEH must have entries of 0, 1 or 2.

2.2 FILES

When running the SAGE option alone, the following five files must be present:

1. Swath Table - SWATH
   See Section 2.4 of the Users Manual.
2. Swath Reference - SWATHR
   See Section 2.4 of the Users Manual.
3. Segment ID File - SEGID
   See Section 2.4 of the Users Manual for a description.

4. Weather Data File - WEATHER
   Required as input if IOPT = 0 and produced as output if IOPT ≠ 0.
   See Section 2.4 of the Users Manual for a description.

5. NASA Weather Tape - WEATAP
   This tape is required input if IOPT = 0.
   See Section 2.4 of the Users Manual for a description.
3. Segment ID File

This file is generated from NASA data cards for use in the SAGE program. This file is only valid for DAPTS and IOC.

Access Method: Sequential with fixed length records

Status: Permanent during life of IOC

Sort: Country, then region, then zone, then strata, then substrata, then segment, 4800 records

Media: Disk

Record Formats:

Header Record:
- Name (2) - 8 bytes, file name - 'SEGMENT'
- ICASE - case no. 20 bytes filler

Detail Record:
- Country ID - 4 bytes, 4 alpha characters
- Region ID - 2 bytes, 3 digit no., set to 1
- Zone ID - 2 bytes, 3 digit no., 1 to 400
- Strata ID - 2 bytes, 4 digit no., 1 to 1600
- Substrata ID - 2 bytes, 1 to 4800
- Segment ID - 2 bytes, 4 digit no., 1 to 4800
- Training Seg Ind - 2 bytes, 0 - normal, 1 - training
- Latitude - 4 bytes, in radians, $\pm \frac{\pi}{2}$
- Longitude - 4 bytes, in radians, 0 - $2\pi$
- Grid No. - 2 bytes, 1-16,000

Crop Proportion Table
- Crop Cat % mixed crop - 4 bytes, %, 0 to 100
- Crop Cat % winter wheat - 4 bytes, %, 0 to 100

Set to a value or 0 for DAPTS.
Crop Cat % spring wheat - 4 bytes, %, 0 to 100
Set to value or 0 for DAPTS
(opposite from Crop Cat %
winter wheat).

Last detail record has 'ZZZZ' in the first 4 bytes
followed by 0's in the remainder of the record.
Total: 32 bytes, 13 words

Block Factor: 20
File Size: 153,800 bytes, 62,400 words
Usage: Used by SAGE to get lat/lon and crop data to write
on segment reference data file.

4. Weather Data File
This file contains cloud cover % for 366 days and 16000 grid
points. This file is generated from a NASA supplied weather
tape.

Access Method: Direct with fixed length records
Status: Semi-permanent. Can be regenerated for each use of
SAGE or kept as a permanent file for IOC studies.
Sort: Grid point, then day
To get to a particular record N which represents 5 grid
points:
\[ N = 1 + \frac{\text{GRID}}{5} + 1 \text{ (if remainder)} \]
\[ + 0 \text{ (if no remainder)} \]

Media: Disk
Record Formats:
Header Record:
Name (2) - 8 bytes, file name = 'WEATHER'
NMAX - No. of days in a record, 2 bytes, 1 to 366

Total: 12 bytes or 4 words
171 byte filler

Detail Record - For 1 grid - all days
Each day entry is a value 0 to 8 representing 8ths of 100%. 8-4 bit entries are stored right justified in a word; for 366 days it would require \[ \frac{366 \times 4}{8} = 183 \text{ bytes or 46 words.} \]

Blocking Factor: 5

File Size: 2,928,000 bytes, 736,000 words

Usage: This file is used by SAGE to obtain the cloud cover data. For FOC on the PDP, the file will be stored in terms of 16 bit words. There will be three entries per PDP word. This will result in a 25% increase in space requirements.

When running the weather file generation utility alone, the NASA weather tape must be present for input.

NASA Weather Tape Format

There is one file on tape written in binary mode (odd parity) with sequential I/O routines. The tape is 7 track written at 800 BPI density. There will be 1600 physical records on tape and each record will be 500 words long. There will be 10 logical index point records per physical record. Each logical record will be 50 words long and will contain BCD information in the following format:

\[
\text{INDEX } \rightarrow \text{GRID} \rightarrow \text{LAT} S \mid \text{LONG} W \mid \text{YEARS} \% \text{FREQ} \mid \text{MEAN} \%
\]

\((15, 2X, 13, 2X, 13, 3X, 312, A1, LX, I3, 2I2, A1, 12, 12 (I3, 8I2), 1213)\)

If both the utility and SAGE are run, then the NASA weather tape would replace the weather data file as input. The weather data file would be computed prior to SAGE execution.
3.0 PROCESSING

3.1 OVERVIEW

The total program is divided into 2 phases. The phase which would be executed first (if selected) would be the weather data file creator. SAGE would be executed second if both phases are selected. The NASA weather tape must be supplied by NASA in the format shown in Section 2.2. In order to generate a weather data file, most of the input required by SAGE will come from the input data files. The segment location file will control SAGE since each segment identified on the file will be processed for acquisitions. In Section 3.2 a program block diagram illustrates the logic flow and the order in which the basic functions are performed. In Section 3.3 an ordered list of the procedures and equations is given.

3.2 PROGRAM FLOW

See Page for a block diagram giving an overview of program functions to be performed, and their order. Segment no. is the key control item for SAGE.

3.3 PROCEDURES AND EQUATIONS

3.3.1 Creation of the Weather Data File

1. Create Weather Data File

   1. A record for 1 grid point is read from the NASA weather tape to get grid no.-GRID and f_j - IFREQ (frequency data set).

   2. The following computations are performed on f_j set for each month Jan through December.

      Form set F_k = \sum_{j=0}^{k} f_j \quad \text{where} \quad 0 \leq k \leq 8

      such that F_0 = f_0', F_1 = f_0' + f_1', etc.

   3. Test F_8 for 0; if it is then Step 4 is not done. Instead, a constant normal cumulative distribution probability table for the current month will be used for F_0-F_8. A message will print if F_8 is not within 99 to 101.
4. Form set \( Y_K = \frac{F_k}{F_8} \) where \( 0 \leq k \leq 8 \) and \( K = k + 1 \)

\( Y_0 = 0. \)

5. The following computations are performed for each day of each month.

Form \( \Omega \) - random number using RAND as the seed.
Determine which pair of \( Y_K \)'s \( 0 \leq K \leq 9 \) \( \Omega \) lies between.
If \( \Omega \) lies between \( Y_K \) and \( Y_{K+1} \), then PC - cloud cover in eighths is set to \( K \). In this manner the PC value for NMAX days of the year (assume 366 day max) is computed.

6. The PC table for IGRID is then written on the weather data file as one logical record.

3.3.2 SAGE

1. Input Procedures

- Read in card data and check fields for errors.
- Read the header records of all input files, store required data in working storage and check for correct files.
- Determine the number of vehicles and if only 1, which one.
- Determine the key time points in the run as shown in the example below:

Data Required: IV1TIM, IV2TIM, ISOSTR and NODAY from the swath table file.
ISTIME, NDAYS from cards.

```
Swath and N. Lat. Start Run Start S. Lat. Start N. Lat. End R. End
Vehicle 1 Start ISOSTR
Vehicle 2 Start
Swath and S. Lat. End
```

Time in Days for Swath

1 2 3 4 5 6 7
Initialize Flags, Ctrs, Storage for the Case

Call RDCD to Read Input Card Data, Account for Nominal Values and Store in Working Storage

Call ERRDET to Check Card Input Errors

Write Status Information for This Phase, Wrap Up Writing of Files

Call WEATH to Read NASA Tape and Generate Weather File

= 1 or 2

Test IOPT for 0, 1 or ≥ 2

= 0

RUN SAGE ONLY

= 1

RUN SAGE

Call INIT to Control the Set Up of Times, Read File Header Records, Check for File Errors and the Writing of Seg. Ref. File Header

Perform Initialization for a Segment

1

Write Trailer Record on Seg. Ref. File, Rewind Files and Print Status Information

STOP

Read 1 Record from the Seg. Id. File

YES

STOP

NO

Is This the Trailer Record

Call SEGPRO to Control the Processing of All Acquisitions for 1 Segment

Write 1 Record on the Seg. Ref. File

Is IREPT True

YES

Call SEGREP to Print Data for 1 Segment

NO

1
Explanation of diagram above:

- Time 1 and time 7 represent the duration of the swath table and reference files.
- Time 2 is ISTIME and represents the desired run start date for access.
- Time 1 and time 3 are IV1TIM and IV2TIM.
- ISOSTR represents the delay in the utilization of the southern latitude band data which starts at time 4 = time 1 + ISOSTR.
- Time 5 represents the last day of northern latitude band day of northern latitude band day utilization and is computed by time 5 = time 7 - ISOSTR.
- Time 6 is = to ISTIME + NDAYS - 1.
- Time 7 is = to the smallest of IV1TIM and IV2TIM + (NODAY - 1).
- In this example, accesses for the northern latitudes and vehicle 1 would start at time 2. Accesses for northern latitudes and both vehicles would start at time 3. Both southern and northern latitudes would be processed starting at time 4. The processing for northern latitudes would stop at time 5 and the run would end at time 6.

2. Processing for 1 Segment

The following procedure is performed for all segments as read from the segment location file:

1) Determine if current segments LAT-LON will be accessed by 1 or 2 vehicles and whether it is in an active hemisphere. If not in an active segment, the remainder of the steps are skipped and the next segment is initiated. Determine day search limits on swath files.

2) If the THETSG is in the band about the equator not in LATNO table, an information message is printed and the next segment is initiated. Given the grid point for the current segment lat/lon, the cloud cover data is read from the weather file for 1 year of daily data, this is stored into ICLDCV array.
3) Establish Potential Access

The following steps are done for each day within the specified time limits and each vehicle:

a. Compute \( \Delta \Omega_{\text{MIN}} \) and \( \Delta \Omega_{\text{MAX}} \)

Determine latitude \( j \) and \( j + 1 \) from LATNO table such that LATNO(\( j \)) \( \geq \theta_T \geq \) LATNO(\( j + 1 \))

\( \Delta \Omega_{\text{MIN}} = \text{MIN} [\Delta \Omega_1(j), \Delta \Omega_1(j + 1)] \)

\( \Delta \Omega_{\text{MAX}} = \text{MAX} [\Delta \Omega_2(j), \Delta \Omega_2(j + 1)] \)

\( \text{LAT} = j; \text{LAT}1 = j + 1 \)

b. Find a day no. - JDAY, rev no. - IREV, vehicle no. - IVEH on the segment reference file such that

\( \Delta \Omega_{\text{MIN}} \leq (\phi_{\text{NODE}} - \phi_{\text{SEG}}) \mod 2\pi \leq \Delta \Omega_{\text{MAX}} \)

If \( (\phi_{\text{NODE}} - \phi_{\text{SEG}}) \) is not within these bounds, then no access is possible. The processing for the next day is initiated. Otherwise, step 4 is next.

4) Access Verification

a. The record on the swath table for day - JDAY and vehicle - IVEH is read to get TIME, LAT and DLONG for latitudes LAT and LAT1. IDAY and IZDAY are computed from JDAY.

b. Letting \( j = \text{LAT} \) and \( j + 1 = \text{LAT}1 \), the following equation set is computed:

\( \text{DELTA} = \text{THETSG} - \text{LATNO}(j) \)

Compute following equations for \( i = 1, 3 \):

\( \text{XTIME}(i) = \text{TIME}(i, 1) + \text{DELTA} \ast [\text{TIME}(i, 2) - \text{TIME}(i, 1)] \)

\( \text{XDLONG}(i) = \text{DLONG}(i, 1) + \text{DELTA} \ast [\text{DLONG}(i, 2) - \text{DLONG}(i, 1)] \)

\( \text{XALT} = \text{ALT}_1 + \frac{\text{ALT}_2 - \text{ALT}_1}{\text{TIME}(2, 2) - \text{TIME}(2, 1)} \ast (\text{XTIME}(2) - \text{TIME}(2, 1)) \)

\( \text{SIGMA} = \arcsin \left[ \left( \frac{\text{RADIUS} + \text{XALT}}{\text{RADIUS}} \right) \ast \sin(\text{SA}(3)) \right] - \text{SA}(3) \)
DDLNGL = \frac{-DECR}{\text{RADIUS} \times \text{SIGMA}} \times (\text{XDLONG}(3) - \text{XDLONG}(2))

DDLNGR = \frac{-DECR}{\text{RADIUS} \times \text{SIGMA}} \times (\text{XDLONG}(2) - \text{XDLONG}(1))

\text{CHECKL} = \text{XDLONG}(3) + \text{DDLNGL}

\text{CHECKR} = \text{XDLONG}(1) - \text{DDLNGR}

\text{CHLON} = (\text{PHISEG} - \text{PHINDE}) \mod 360 \ (2\pi)

Check for valid access:

If CHLON < than CHECKL, then no access is possible and step 3 is retried for the next day.

If CHLON > than CHECKR, then no access is possible and step 3 is retried for the next day.

If neither of these conditions exist, then there is an access and step 5 is performed.

5) Compute imaging conditions:

If \ (\text{CHLON} - \text{XDLONG}(2)) \ is -
< 0 access is on left side, execute a.
= 0 access on ground track, execute b.
> 0 access is on right side, execute c.

a. \quad \text{TS} = \text{XTIME}(2) + \left[ \text{CHLON} - \text{XDLONG}(2) \right] \times \left[ \frac{\text{XTIME}(3) - \text{XTIME}(2)}{\text{XDLONG}(3) - \text{XDLONG}(2)} \right] + \text{TMNODE}

\text{CTA} = +\tan^{-1} \left[ \left( \frac{\text{CHLON} - \text{XDLONG}(2)}{\text{XDLONG}(3) - \text{XDLONG}(2)} \right) \times \tan(\text{SA}(3)) \right]

\text{YALT} = \text{ALT}(1) + \left[ \frac{\text{TS} - \text{TIME}(2, 1)}{\text{TIME}(2, 2) - \text{TIME}(2, 1)} \right] \times [\text{ALT}(2) - \text{ALT}(1)]

b. \quad \text{TS} = \text{XTIME}(2) + \text{TMNODE}

\text{CTA} = 0

\text{YALT} = \text{XALT}
c. \[ TS = XTIME(2) + \left[ \frac{CHLONG - XDLOG(2)}{XDLOG(1) - XDLOG(2)} \right] \times \left[ XTIME(1) - XTIME(2) \right] + TMNODE \]
\[ CTA = -\tan^{-1} \left[ \left( \frac{CHLONG - XDLOG(2)}{XDLOG(1) - XDLOG(2)} \right) \tan(SA(3)) \right] \]
\[ YALT = ALT(1) + \left[ \frac{TS - TIME(2, 1)}{TIME(2, 2) - TIME(2, 1)} \right] \times [ALT(2) - ALT(1)] \]

6) Compute sun angle - EL

This calculation is performed by using an existing subroutine ALPHA.

Store in ICLOD the % value from the cloud cover day table for IDAY.

The following data is stored by acquisition day for each acquisition from the computed quantities above. I represents the count on number of acquisitions, 1 \( \leq I \leq 150 \).

IZDAYT(I) = IZDAY, ICLODT(I) = ICLOD
IREVT(I) = IREV, TST(I) = TS
IVEHT(I) = IVEH, YALTT(I) = YALT
CTAT(I) = CTA
ELT(I) = EL

7) The data from segment location file is then combined with the data from step 6 and a record on the segment reference data file is written.

3. If IREPT is true, the segment acquisition data report is generated.

4. Steps 2 and 3 are repeated for each segment on the segment location file.
<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Description</th>
<th>Source</th>
<th>Used</th>
<th>Range</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLOMMN</td>
<td>ΔΩ₁</td>
<td>The minimum Δ longitude at a given latitude from the ascending node.</td>
<td>Sw. Tab.</td>
<td>3</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>DLONMX</td>
<td>ΔΩ₂</td>
<td>The maximum Δ longitude at a given latitude from the ascending node.</td>
<td>Sw. Tab.</td>
<td>3</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>ICLDCV</td>
<td></td>
<td>% * 10 of cloud cover for each day of a year.</td>
<td>Weath. File</td>
<td>2</td>
<td>0-1000</td>
<td>Radians</td>
</tr>
<tr>
<td>PHINOD</td>
<td>φNODE</td>
<td>Longitude of ascending node for all revs for a day.</td>
<td>Sw. Ref.</td>
<td>3</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>PHISEG</td>
<td>φSEG</td>
<td>Longitude of segment.</td>
<td>Seg. Loc.</td>
<td>1, 2, 3, 4, 7</td>
<td>0-2π</td>
<td>Radians</td>
</tr>
<tr>
<td>THETSG</td>
<td>θ_T</td>
<td>Latitude of segment.</td>
<td>Seg. Loc.</td>
<td>1, 2, 3, 4, 7</td>
<td>0-π/2</td>
<td>Radians</td>
</tr>
<tr>
<td>LATNO</td>
<td></td>
<td>Table of latitude no.'s to be studied.</td>
<td>Sw. Tab.</td>
<td>3</td>
<td>0-±65</td>
<td>Deg</td>
</tr>
<tr>
<td>IZDAY</td>
<td></td>
<td>Zulu day no. for which there is an acquisition.</td>
<td>4</td>
<td></td>
<td></td>
<td>Day</td>
</tr>
<tr>
<td>IREV</td>
<td></td>
<td>Day rev no. of acquisition.</td>
<td>3</td>
<td>7</td>
<td></td>
<td>--</td>
</tr>
<tr>
<td>IVEH</td>
<td></td>
<td>Veh no. associated with acquisition day.</td>
<td>3</td>
<td>7</td>
<td></td>
<td>--</td>
</tr>
<tr>
<td>IGRID</td>
<td></td>
<td>Grid point no. for current segment latitude-longitude</td>
<td>Seg. Loc. File</td>
<td>2</td>
<td></td>
<td>--</td>
</tr>
<tr>
<td>LAT</td>
<td></td>
<td>Latitude ptr. into LATNO which gives upper bound latency on θ_T.</td>
<td>3</td>
<td>4</td>
<td></td>
<td>--</td>
</tr>
<tr>
<td>LAT1</td>
<td></td>
<td>Latitude ptr. into LATNO which gives lower bound latitude on θ_T.</td>
<td>3</td>
<td>4</td>
<td></td>
<td>--</td>
</tr>
<tr>
<td>TMNODE</td>
<td>tNODE</td>
<td>Time ascending node crosses equator for ZDAY, IREV and IVEH.</td>
<td>Sw. Ref.</td>
<td>5</td>
<td>0</td>
<td>Sec</td>
</tr>
</tbody>
</table>

*New
### SAGE Symbol Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Description</th>
<th>Source</th>
<th>Used</th>
<th>Range</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLNMIN</td>
<td>$\Delta \Omega_{\text{MIN}}$</td>
<td>The conservative minimum $\Delta$ longitude which bounds the segment.</td>
<td>3</td>
<td>3</td>
<td>$0-2\pi$</td>
<td>Radians</td>
</tr>
<tr>
<td>DLNMAX</td>
<td>$\Delta \Omega_{\text{MAX}}$</td>
<td>The conservative maximum $\Delta$ longitude which bounds the segment.</td>
<td>3</td>
<td>3</td>
<td>$0-2\pi$</td>
<td>Radians</td>
</tr>
<tr>
<td>DLONG</td>
<td>$\Delta \phi$</td>
<td>Vehicle and swath $\Delta$ longitudes for LAT and LAT1.</td>
<td>Sw. Tab., 3, 4</td>
<td>4, 5</td>
<td>$0-2\pi$</td>
<td>Radians</td>
</tr>
<tr>
<td>TIME</td>
<td>(3, 2)</td>
<td>Times associated with DLONG</td>
<td>Sw. Tab.</td>
<td>3, 4, 5</td>
<td>1-84,600</td>
<td>Sec</td>
</tr>
<tr>
<td>ALT</td>
<td>(2)</td>
<td>Vehicle altitude at LAT and LAT1</td>
<td>Sw. Tab., 3</td>
<td>4, 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PHINDE</td>
<td>$\phi_{\text{NODE}}$</td>
<td>Longitude of ascending node for IREV, IDAY and IVEH.</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SA</td>
<td>(3)</td>
<td>Reference scan angle</td>
<td>Sw. Ref.</td>
<td>4, 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DELTA</td>
<td></td>
<td>Latitude interpolation difference.</td>
<td>4</td>
<td>4</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>XTIME</td>
<td>(3)</td>
<td>Vehicle time at latitude of interest for $i^{\text{th}}$ swath position: left, center right.</td>
<td>4</td>
<td>4, 5</td>
<td></td>
<td>Sec</td>
</tr>
<tr>
<td>XDLONG</td>
<td>(3)</td>
<td>Delta longitude at latitude of interest for $i^{\text{th}}$ swath position: i = left, center, right.</td>
<td>4</td>
<td>5, 4</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>XALT</td>
<td></td>
<td>Vehicle altitude at latitude of interest.</td>
<td>4</td>
<td>4</td>
<td></td>
<td>Kilom.</td>
</tr>
<tr>
<td>SIGMA</td>
<td></td>
<td>Half earth centered angle for swath.</td>
<td>4</td>
<td>4</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>DDLNGL</td>
<td></td>
<td>Longitude decrement for left side.</td>
<td>4</td>
<td>4</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>DDLNGR</td>
<td></td>
<td>Longitude decrement for right side.</td>
<td>4</td>
<td>4</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>CHECKL</td>
<td></td>
<td>Relative longitude for left side of swath.</td>
<td>4</td>
<td>4</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>CHECKR</td>
<td></td>
<td>Relative longitude for right side of swath.</td>
<td>4</td>
<td>4</td>
<td></td>
<td>Radians</td>
</tr>
</tbody>
</table>

*New
### SAGE Symbol Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Description</th>
<th>Source</th>
<th>Used</th>
<th>Range</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECR</td>
<td></td>
<td>Swath decrement input from cards.</td>
<td>Cd. In.</td>
<td>4</td>
<td>1 to 100</td>
<td>Kilom.</td>
</tr>
<tr>
<td>TS</td>
<td>$T_S$</td>
<td>Image time.</td>
<td>5</td>
<td>5, 7</td>
<td></td>
<td>Sec</td>
</tr>
<tr>
<td>CTA</td>
<td></td>
<td>Crosstrack angle.</td>
<td>5</td>
<td>5, 7</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>YALT</td>
<td></td>
<td>Altitude of vehicle at time of acquisition.</td>
<td>5</td>
<td>5, 7</td>
<td></td>
<td>Kilom.</td>
</tr>
<tr>
<td>RADIUS</td>
<td></td>
<td>Radius of earth - 6376.436</td>
<td>Const.</td>
<td>4</td>
<td></td>
<td>Deg</td>
</tr>
<tr>
<td>EL</td>
<td></td>
<td>Sun elevation angle.</td>
<td>6</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CIS</td>
<td></td>
<td>Obliquity of the ecliptic - .409280.</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td>Deg</td>
</tr>
<tr>
<td>CNUV</td>
<td></td>
<td>Constant-true anomaly of the earth at vernal equinox - 1.35695.</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td>Deg</td>
</tr>
<tr>
<td>ECCE</td>
<td></td>
<td>Constant-eccentricity of the earth's orbit - .167263E-1.</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td>Deg</td>
</tr>
<tr>
<td>JDAY</td>
<td></td>
<td>Day no. from 1 to 549 to compute swath table record no.</td>
<td>Sw. Ref.</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDAY</td>
<td></td>
<td>Day no. of the current year (current date).</td>
<td>4</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PI</td>
<td>$\pi$</td>
<td>3.1415926</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PI2</td>
<td>$\pi/2$</td>
<td>1.5707963</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PI32</td>
<td>$3/2\pi$</td>
<td>4.712389</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RADIAN</td>
<td></td>
<td>57.29578</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWOPI</td>
<td>$2\pi$</td>
<td>6.2831852</td>
<td>Const.</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHLONG</td>
<td></td>
<td>Relative longitude difference of segment of interest.</td>
<td>4</td>
<td>4, 5</td>
<td></td>
<td>Radians</td>
</tr>
<tr>
<td>ICLOD</td>
<td></td>
<td>Cloud cover % * 10 for day of acquisition.</td>
<td>6</td>
<td>7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.0 OUTPUT

Where appropriate, the output discussion will refer to SAGE and to the weather data file utility. If no mention is made of the utility or SAGE, then it can be assumed the information refers to SAGE only.

4.1 PRINTED DATA

4.1.1 Reports

There is only one report and it is printed by segment no. The report lists the pertinent data for each time a given segment is acquired by one of the vehicles. The format of the report is as follows:

<table>
<thead>
<tr>
<th>ACCESS COUNT</th>
<th>ACQUISI MO/DY/YR</th>
<th>REV</th>
<th>VEH</th>
<th>CTA</th>
<th>EL</th>
<th>CLOUD COVER</th>
<th>ACQUISI TIME</th>
<th>ALT.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>nn/nm/nm</td>
<td>nm</td>
<td>n</td>
<td>nnn</td>
<td>nnn</td>
<td>nnn</td>
<td>nnn</td>
<td>nnnn</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>150</td>
<td>nn/nm/nm</td>
<td>nm</td>
<td>n</td>
<td>nnn</td>
<td>nnn</td>
<td>nnn</td>
<td>nnn</td>
<td>nnnn</td>
</tr>
</tbody>
</table>

There is a maximum of 150 acquisitions possible requiring 4 pages of print, but in general, the total acquisitions will fit on 1 page.

4.1.2 Intermediate Debug

There is no debug output in this program.

4.1.3 Status Information

Weather Data File Utility

The following information is printed upon successful completion of the utility:
***WEATHER DATA FILE UTILITY COMPLETED
WEATHER DATA FILE:
   NO OF DAYS IN A RECORD: NNN
   NO OF RECORDS READ: NNNN
   LAST RANDOM NO SEED: NNNNNNNNNNN
SAGE Program
The following information prints out upon successful completion of
the SAGE program:
***SAGE HAS SUCCESSFULLY WRITTEN THE SEGMENT
REFERENCE DATA FILE
SEGMENT REFERENCE DATA FILE:
   REFERENCE DATE: NN/NN/NN
   SWATH FILES CASE USED: NNNN
   NO OF RECORDS WRITTEN: NNNN
SEGMENT ID FILE:
   REFERENCE DATE: NN/NN/NN
   SEGMENT ID FILE CASE USED: NNNN
   NO OF RECORDS READ: NNNN

4.1.4 Echo Print of Input Control Cards
The print format is as follows:

HEADER
AAA > AAA
ICASE IOPT NDAYS ICSESW ICSESG ISTEME IREPT DECR
N NNN N NNN NNNN NNNN NN L N NN

4.2 FILES
There are two possible output files. The weather data file utility
produces the weather data file - WEATHR. This file is defined already
in Section 2.2. SAGE generates the segment reference file - SEGREF.
See Section 2.4 of the Users Manual for a description.
SEGMENT REFERENCE FILE

Access Method: Sequential with fixed length records

Status: Semi-permanent. Regenerated normally less than once per week. Will normally be kept on tape.

Sort: Country, then region, then zone, then strata, then substrata, and then segment. There are 4800 segments for DAPTS and IOC; 10000 for FOC.

Media: Disk or tape, tape for IOC

Record Formats:

Header Record:

Name (2) - 8 bytes, file name: 'SEG REFEl'

ICASE - 2 bytes, case no. - 4 digits

Reference Date - No. of days since 1900 (2 bytes)
    Used in conjunction with acquisition date.

NMAX - 2 bytes, no. of days in study; range 1 to 426

No. of Crops - 3 for IOC, max of 26 for FOC

ICSESW - 2 bytes, swath files case no. - 4 digits

ICSESG - 2 bytes, segment ID file case no. - 4 digits

Total: 20 bytes or 8 words + 2360 bytes filler for IOC

Detail Record:

Country ID - 4 bytes, 4 alpha characters

Region ID - 2 bytes, 3 digit no., 1 to 100

Zone ID - 2 bytes, 3 digit no., 1 to 400

Strata ID - 2 bytes, 4 digit no., 1 to 1600

Substrata ID - 2 bytes, 4 digit no., 1 to 4800

Segment ID - 2 bytes, 5 digit no., 1 to 10000

Training Seg. Ind. - 2 bytes, 0 - normal, 1 - training
Training Seg. Ptr. - 2 bytes, segment no. of associated training segment (FOC)

Latitude - 4 bytes, in radians, $\pm \pi$

Longitude - 4 bytes, in radians, 0-2$\pi$

Atmospheric Type - 2 bytes, 2 alphanumeric characters (FOC)

Group No. - 2 bytes (FOC)

Field Size - 2 bytes, kilometers$^2 \times 10$, range 10-20000 (FOC)

Cropping Practice Flag - 2 bytes (FOC)

Crop Proportion Table - IOC (FOC has 26 entries)

Crop Cat % mixed crop - 4 bytes, %, 0 to 100

Crop Cat % winter wheat - 4 bytes, %, 0 to 100

Crop Cat % spring wheat - 4 bytes, %, 0 to 100

Acquisition Data - 150 entries each

Acquisition Day - 2 bytes, Zulu date (no. of days since 1900)

Revolution No. - 2 bytes, range 1 to 17

Vehicle No. - 2 bytes, range 1 to 2

Crosstrack Angle - 4 bytes, Radians, range $\pm 10^\circ$

Sun Elevation Angle - 4 bytes, Radians, range $\pm 90^\circ$

Cloud Cover % - 2 bytes, % * 10

Time of Acquisition - 4 bytes, sec, 1 to 86,400

Vehicle Altitude - 2 bytes, kilometers $\times 10$, 10 to 10000

Station Contact Ind. - 2 bytes (FOC)

Atmosphere Attenuation Factor - 2 bytes (FOC)

Snow Cover % - 2 bytes (FOC)

Last detail record has 'ZZZZ' in the first 4 bytes followed by 0's in the remainder of the record.
Total Length: IOC - 2,740 bytes, 1,217 words  
FOC - 3,686 bytes, 1,690 words

Block Factor: 1

File Size: IOC - 2,425,212 words  
FOC - 14,700,008 words

Usage: This file will be used as follows:

1. DAPTS
   Used by SACS in conjunction with the crop window file to determine when accesses occur and conditions of access. This data is then represented on the segment acquisition file.

2. IOC
   Same usage as above. In addition, it will be used in conjunction with the data acquisition file by the CAMS module.

3. FOC
   Same usage as IOC except crop calendar information is obtained from substrata properties truth file.
5.0 ERROR PROCESSING

5.1 GENERAL

The program will attempt to find as many errors during the input card processing as possible. The program will continue checking for further input errors upon detecting any input error. There are 2 levels of error. These are:

Level 1 - continue processing

Level 2 - job fatal

When a level 1 error occurs, the program will print an informative message and continue. When a level 2 error occurs, the program will print an informative message and return control back to the computer system.

5.2 INPUT DATA ERRORS

Level 2

1. A check is made to see if ICSESW matches the case no. on the swath files. Message:

***ICSESW DOES NOT MATCH THE CASE NO. ON THE SWATH INPUT FILES OR SWATH FILES NOT MOUNTED.

2. A check is made to see if ICSESG matches the case no. on the segment ID file. Message:

***ICSESG DOES NOT MATCH THE CASE NO. ON THE SEGMENT ID FILE OR THIS FILE HAS NOT BEEN MOUNTED.
3. A check is made to make sure that ISTIME is not less than the earliest vehicle start time as specified on the swath table file. Message:

***ISTIME IS LESS THAN THE EARLIEST VEHICLE START TIME I1V1T1M OR I1V2T1M.

4. A check is made to make sure NDAYS is between 1 and 426. Message:

***NDAYS IS NOT BETWEEN 1 AND 426.

5. A check is made to make sure that the vehicle no. list in IVEH is between 0 and 2 and that there is at least one NONZERO entry. Message:

*** IVEH HAS AN ENTRY NOT BETWEEN 0 AND 2 OR DOES NOT HAVE AT LEAST 1 NONZERO ENTRY.

6. A check is made to make sure that a weather data file has been mounted. Message

***THE WEATHER DATA FILE HAS NOT BEEN MOUNTED.

7. A check is made to make sure that the input quantity IVEH does not specify vehicles not on the swath files. Message.

*** IVEH LIST IS NOT COMPATIBLE WITH NVEH AS SPECIFIED ON THE SWATH FILES.

8. A check is made to make sure IGRDN is between 1 and 16000.

***IGRDN DOES NOT HAVE A VALUE BETWEEN 1 AND 16000

5.3 PROCESSING ERRORS

Level 1

20. A check is made while forming the acquisition list for a segment that no more than 150 acquisitions occur. Message:

***IN PROCESSING SEGMENT NNNN MORE THAN 150 ACQUISITIONS HAVE OCCURRED. NO MORE ACQUISITIONS WILL BE PROCESSED.
21. A check will be made to make sure at least 1 acquisition of a segment occurs. Message:

***IN PROCESSING SEGMENT NNNN NO VALID ACQUISITION OCCURRED.

22. A check is made to see if a segment's latitude is in the band about the equator which is not accounted for in LATNO table. Message:

***SEGMENT NNNN LATITUDE NN NN NN IS NOT IN THE LATITUDE BAND SPECIFIED BY LATNO TABLE.

23. A check is made in the weather file utility to make sure $99 \leq F_8 \leq 101$. Message.

5.4 INPUT/OUTPUT ERRORS

For sequential I/O the FORTRAN system on the UNIVAC or PDP takes control and prints a message identifying the problem and will either continue processing or abandon the job. If processing continues, the system counts the number of times this error re-occurs and if it happens a certain number of times, the system will abandon the job.

For direct access I/O, the UNIVAC or PDP D.A. I/O package prints out an informative message, sets an error flag and allows processing to continue. In this program the swath table file is the only direct access file. Immediately after the informative message, the following message will print:

40. ***AN IRRECOVERABLE I/O ERROR HAS OCCURRED IN READING/Writing A RECORD ON THE ________ FILE. THE JOB IS BEING ABANDONED.

The blanks will be filled in with 'SWATH TB', or 'WEATHER', depending on the direct access file involved.
PART II

COMMON BLOCK DEFINITIONS
### Common Storage Allocation

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEADER</td>
<td>18</td>
<td>R</td>
<td>Contains 72 CHAR HEADER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICASE</td>
<td>1</td>
<td>I</td>
<td>Case no of access run</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JCSESW</td>
<td>1</td>
<td>I</td>
<td>Swath files case no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICSESG</td>
<td>1</td>
<td>I</td>
<td>Segment ID file case no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISTIME</td>
<td>3</td>
<td>I</td>
<td>Run start date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISDAY</td>
<td>1</td>
<td>I</td>
<td>Start day=ISTIME only in Zulu time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NDAYS</td>
<td>1</td>
<td>I</td>
<td>No. of days in the run set to 426</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IOPT</td>
<td>1</td>
<td>I</td>
<td>Program run option -0- run SAGE, 1-Run weather, run both</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LVEH</td>
<td>2</td>
<td>I</td>
<td>List of vehicle no.'s to process set to 1, 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IREPT</td>
<td>1</td>
<td>L</td>
<td>Report print indicator F-No Print, T-Print</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NVEH</td>
<td>1</td>
<td>I</td>
<td>No. of vehicles on swath files</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NODAY</td>
<td>1</td>
<td>I</td>
<td>No. of days in swath files</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IVTIM</td>
<td>(3)</td>
<td>I</td>
<td>Vehicle 1 swath start date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IGRID</td>
<td>1</td>
<td>I</td>
<td>Current weather grid no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IV2TIM</td>
<td>(3)</td>
<td>I</td>
<td>Vehicle 2 swath calen, start date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LYR</td>
<td>1</td>
<td>I</td>
<td>Current 2 digit years</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LMO</td>
<td>1</td>
<td>I</td>
<td>Current month no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Dimension</td>
<td>Format</td>
<td>Description</td>
<td>Symbol</td>
<td>Units</td>
</tr>
<tr>
<td>-------</td>
<td>-----------</td>
<td>--------</td>
<td>-------------------------------------------------------</td>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>L</td>
<td></td>
<td>1</td>
<td>Current day of month no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDA</td>
<td>1</td>
<td>1</td>
<td>Zulu start day of swath table = earliest swath day</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISWDST</td>
<td>1</td>
<td>1</td>
<td>Swath table file day no used to get record no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JDAY</td>
<td>1</td>
<td>1</td>
<td>Day no. of current year</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDAY</td>
<td>1</td>
<td>1</td>
<td>Previous day no. of current year</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPDAY</td>
<td>1</td>
<td>1</td>
<td>Day no. of run relative to 1950 - current day</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IYEAR</td>
<td>1</td>
<td>1</td>
<td>Day no. of run relative to 1950 - current day</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IYEBR</td>
<td>1</td>
<td>1</td>
<td>Previous year including 1950</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICDAY</td>
<td>1</td>
<td>1</td>
<td>Zulu - start day of vehicle 1 swath</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISWVST</td>
<td>2</td>
<td>1</td>
<td>Zulu - start day of vehicle 2 swath</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISLATS</td>
<td>1</td>
<td>1</td>
<td>Zulu - start day of southern latitude</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INLATE</td>
<td>1</td>
<td>1</td>
<td>Zulu - end day of northern latitude</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IRUNE</td>
<td>1</td>
<td>1</td>
<td>Zulu - end day of northern latitude</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISWATE</td>
<td>1</td>
<td>1</td>
<td>Zulu - end day of southern latitude</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMAX</td>
<td>1</td>
<td>1</td>
<td>No. of days in a weather data record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NLAT</td>
<td>1</td>
<td>1</td>
<td>No. of latitudes in LATNO band and in swath file rec.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISO_STR</td>
<td></td>
<td>1</td>
<td>Delay for southern hemisphere start relative to north lat months</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

Name: WORK  
Size: 414

Function: Contains Data Pertinent to the Acquisition Calculations.

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICLDCV</td>
<td>366</td>
<td>I</td>
<td>%×10 Cloud cover for each day of year</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IZDAY</td>
<td>1</td>
<td>I</td>
<td>Zulu day no. for which there is an acquisition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IREV</td>
<td>1</td>
<td>I</td>
<td>Rev no. on acquisition day IZDAY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IVEH</td>
<td>1</td>
<td>I</td>
<td>Veh no. on acquisition day IZDAY</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAT</td>
<td>1</td>
<td>I</td>
<td>Latitude pointer into LATNO which gives upper bound latitude on ( \theta_T )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LAT1</td>
<td>1</td>
<td>I</td>
<td>Latitude PTR into LATNO which gives lower bound lat on ( \theta_T )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TMNODE</td>
<td>1</td>
<td>R</td>
<td>Time ascending node cross's equator for IDAY, IREV and IVEH</td>
<td>( t_{\text{node}} )</td>
<td></td>
</tr>
<tr>
<td>DLNMIN</td>
<td>1</td>
<td>R</td>
<td>The conservative minimum ( \Delta ) longitude which bounds the seg.</td>
<td>( \Delta \omega_{\text{min}} )</td>
<td></td>
</tr>
<tr>
<td>DLNMAX</td>
<td>1</td>
<td>R</td>
<td>The conservative maximum ( \Delta ) longitude which bounds the seg.</td>
<td>( \Delta \omega_{\text{max}} )</td>
<td></td>
</tr>
<tr>
<td>DLONG</td>
<td>3, 2</td>
<td>R</td>
<td>Vehicle and swath ( \Delta ) longitudes for lat and lat1</td>
<td>( \Delta \phi )</td>
<td></td>
</tr>
<tr>
<td>TIME</td>
<td>3, 2</td>
<td>R</td>
<td>Times associated with DLONG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALT</td>
<td>2</td>
<td>R</td>
<td>Vehicle altitude at lat and lat1</td>
<td></td>
<td>KILO</td>
</tr>
<tr>
<td>PHINDE</td>
<td>1</td>
<td>R</td>
<td>Longitude of ascending node for IREV, IDAY and IVEH</td>
<td>( \phi_{\text{Node}} )</td>
<td>RAD.</td>
</tr>
<tr>
<td>SA</td>
<td>3</td>
<td>R</td>
<td>Reference scan angle</td>
<td></td>
<td>RAD.</td>
</tr>
<tr>
<td>DELTA</td>
<td>1</td>
<td>R</td>
<td>See problem definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XTIME</td>
<td>3</td>
<td>R</td>
<td>See problem definition</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## COMMON STORAGE ALLOCATION

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>XDLONG</td>
<td>3</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XALT</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SIGMA</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DDLNLG</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DDLNCR</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHECKL</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHECKR</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCER</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAND</td>
<td>1</td>
<td>D.P</td>
<td>Random Number generator input seed (set to 1.0)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TS</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td>T_s</td>
<td></td>
</tr>
<tr>
<td>CTA</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YALT</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICLOD</td>
<td>1</td>
<td>I</td>
<td>Cloud Cover % See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EL</td>
<td>1</td>
<td>R</td>
<td>Sun elevation angle</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHLONG</td>
<td>1</td>
<td>R</td>
<td>See Problem Definition</td>
<td></td>
<td>RAD</td>
</tr>
<tr>
<td>IACCT</td>
<td>1</td>
<td>I</td>
<td>Count of no. of acquisitions for a Segment</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## COMMON STORAGE ALLOCATION

Name **CONS**

Size **28**

Function: Contains constants, file unit numbers and page control data for printing

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>INP</td>
<td>1</td>
<td>I</td>
<td>Card reader = 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KO</td>
<td>1</td>
<td>I</td>
<td>Printer = 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NLINE</td>
<td>1</td>
<td>I</td>
<td>Current line count = 40 initially</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LINMAX</td>
<td>1</td>
<td>I</td>
<td>Max lines on a page = 39</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NPAGE</td>
<td>1</td>
<td>I</td>
<td>Page count = 0 - initially</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISEGF</td>
<td>1</td>
<td>I</td>
<td>Segment file = 1 ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISWR</td>
<td>1</td>
<td>I</td>
<td>Swath reference file = 9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISWTABLE</td>
<td>1</td>
<td>I</td>
<td>Swath Table file = 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IREFF</td>
<td>1</td>
<td>I</td>
<td>Seg. reference file = 10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IWTHF</td>
<td>1</td>
<td>I</td>
<td>Weather data file = 11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IWTAPE</td>
<td>1</td>
<td>I</td>
<td>NASA weather tape = 33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RADIUS</td>
<td>1</td>
<td>R</td>
<td>6,376.436 radius of the earth</td>
<td></td>
<td>KILOM</td>
</tr>
<tr>
<td>CIS</td>
<td>1</td>
<td>R</td>
<td>409,280 obliquity of the ecliptic</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CNUV</td>
<td>1</td>
<td>R</td>
<td>1.35695 - See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECCE</td>
<td>1</td>
<td>R</td>
<td>.167263 E-1 See Problem Definition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PI</td>
<td>1</td>
<td>R</td>
<td>3.1415926</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PI2</td>
<td>1</td>
<td>R</td>
<td>1.5707963</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PI32</td>
<td>1</td>
<td>R</td>
<td>4.712389</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### COMMON STORAGE ALLOCATION

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>RADIAN</td>
<td>1</td>
<td>R</td>
<td>57.29578</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWOPI</td>
<td>1</td>
<td>R</td>
<td>6.2831852</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAME</td>
<td>2</td>
<td>R</td>
<td>'SEGb REFER'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOCROP</td>
<td>1</td>
<td>I</td>
<td>= 3 No. of crops IOC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBLKF</td>
<td>1</td>
<td>I</td>
<td>Blocking factor for weather tape = 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPCK</td>
<td>1</td>
<td>I</td>
<td>No. of weather items packed in a word = 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NPREC</td>
<td>1</td>
<td>I</td>
<td>= 230 - No. of words in physical rec. of IWTHF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NLREC</td>
<td>1</td>
<td>I</td>
<td>= 46 - No. of words in logical rec. of IWTHF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISWRTO</td>
<td>1</td>
<td>I</td>
<td>Total no. of records on swath table file - nominal 1101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IVCT</td>
<td>1</td>
<td>I</td>
<td>Count on no. of vehicles</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NLATSW</td>
<td>1</td>
<td>I</td>
<td>No. of latitudes in the (input data) swath table - nominal 100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IGRDN</td>
<td>1</td>
<td>I</td>
<td>No. of grid points to read and write on weather file</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>nominal = 16000- can be changed for debug</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## COMMON STORAGE ALLOCATION

### Name: DIRAC  |  Size: 5378

#### Function: Direct access file storage

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDEXSW</td>
<td>1104</td>
<td>I</td>
<td>Swath table file index CDC only</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IA</td>
<td>800</td>
<td>-</td>
<td>Swath buffer storage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDEXW</td>
<td>3201</td>
<td>I</td>
<td>Weather file index CDC only</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IB</td>
<td>230</td>
<td>-</td>
<td>Weather file storage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPC</td>
<td>46</td>
<td>I</td>
<td>Contains the weather % cloud cover in eights - 4 bits pack</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8 per word right justified</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COMMON STORAGE ALLOCATION

Name: WEATDT  Size: 477

Function: Contains weather data from weather tape

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA</td>
<td>300</td>
<td>I</td>
<td>Contains 10 logical records of data direct for weather tape, UNIVAC format</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IB</td>
<td>50</td>
<td>I</td>
<td>Contains 1 logical record converted to BCD code right justified with</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IGRIDPT</td>
<td>1</td>
<td>I</td>
<td>Grid point no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPERWT</td>
<td>19, 12</td>
<td>I</td>
<td>% frequency table for 9 eights entries 0-8 for 12 months</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IC</td>
<td>30</td>
<td>I</td>
<td>Contains 1 logical record = IB only packed 10 char per word for decode</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# COMMON STORAGE ALLOCATION

Name: **ACQUIS**  
Size: **1200**

**Function:** Contains acquisition and weather data  
**Data for writing on seg. ref. file**

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>IZDAYT</td>
<td>150</td>
<td>I</td>
<td>See IZDAY def. in work com.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IREVT</td>
<td>150</td>
<td>I</td>
<td>See IREV def. in work com.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CTAT</td>
<td>150</td>
<td>R</td>
<td>See CTA def. in work com.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ELT</td>
<td>150</td>
<td>R</td>
<td>See ELT def. in work com.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICLGDT</td>
<td>150</td>
<td>I</td>
<td>See ICLGDT def. in work com.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TST</td>
<td>150</td>
<td>R</td>
<td>See TS def. in work com.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YALTT</td>
<td>150</td>
<td>R</td>
<td>See YALTT def. in work com.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IVEHT</td>
<td>150</td>
<td>I</td>
<td>See IVEHT def. in work com.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COMMON STORAGE ALLOCATION

Name: SEGFIL
Size: 13

Function: Contains data for one segment record segment ID file

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUNTR</td>
<td>2</td>
<td>R</td>
<td>Alphas country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IREG</td>
<td>1</td>
<td>I</td>
<td>Region no. 1 to 100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IZONE</td>
<td>1</td>
<td>I</td>
<td>Zone no. 1 to 400</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISTRAT</td>
<td>1</td>
<td>I</td>
<td>Strata no. 1 to 1600</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISUBST</td>
<td>1</td>
<td>I</td>
<td>Substrata 1 to 4800</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEGMENT</td>
<td>1</td>
<td>I</td>
<td>Segment ID 1 to 4800</td>
<td></td>
<td></td>
</tr>
<tr>
<td>THETSG</td>
<td>1</td>
<td>R</td>
<td>Segment latitude ±π/2</td>
<td>θ SEG</td>
<td>RAD</td>
</tr>
<tr>
<td>PHISEG</td>
<td>1</td>
<td>R</td>
<td>Segment longitude 0 - 2π</td>
<td>φ SEG</td>
<td>RAD</td>
</tr>
<tr>
<td>ICROP</td>
<td>3</td>
<td>I</td>
<td>Crop proportion table</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ITRSEG</td>
<td>1</td>
<td>I</td>
<td>Training segment indicator</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### COMMON STORAGE ALLOCATION

**Name**: SWATH  
**Size**: 1041

**Function**: Contains swath data working area

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>LATNO</td>
<td>100</td>
<td>I</td>
<td>List of latitudes from north to south ±65</td>
<td></td>
<td>DEG</td>
</tr>
<tr>
<td>IALT</td>
<td>100</td>
<td>I</td>
<td>Vehicle altitudes vs LATNO</td>
<td></td>
<td>KILOM</td>
</tr>
<tr>
<td>TIME1</td>
<td>100, 3</td>
<td>R</td>
<td>Vehicle and swath edge times 0-86,400 vs LATNO</td>
<td></td>
<td>SEC</td>
</tr>
<tr>
<td>DLONG1</td>
<td>100, 3</td>
<td>R</td>
<td>Vehicle and swath edge Δ longitude 0-2π vs LATNO</td>
<td>Δφ</td>
<td>RAD</td>
</tr>
<tr>
<td>DLONMN</td>
<td>100</td>
<td>R</td>
<td>Minimum Δ longitude at a given latitude vs LATNO, 0-2π</td>
<td>ΔΩ₁</td>
<td>RAD</td>
</tr>
<tr>
<td>DLONMX</td>
<td>100</td>
<td>R</td>
<td>Maximum Δ longitude at a given latitude vs LATNO, 0-2π</td>
<td>ΔΩ₂</td>
<td>RAD</td>
</tr>
<tr>
<td>KVEH</td>
<td>1</td>
<td>I</td>
<td>Current vehicle no. from swath ref 1-2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KDAY</td>
<td>1</td>
<td>I</td>
<td>Current day no. 1-549 from swath ref</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NREV</td>
<td>1</td>
<td>I</td>
<td>No. of revolutions for current day and vehicle - from swath ref.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIMNOD</td>
<td>17</td>
<td>R</td>
<td>Time ascending node cross's equator for each rev swath ref.</td>
<td>tₙode</td>
<td>SEC</td>
</tr>
<tr>
<td>PHINOD</td>
<td>17</td>
<td>R</td>
<td>Longitude of ascending node for each rev swath ref</td>
<td>φₙode</td>
<td>RAD</td>
</tr>
<tr>
<td>INLAT</td>
<td>2</td>
<td>I</td>
<td>North latitude band (1) - low, (2) high</td>
<td></td>
<td>DEG</td>
</tr>
<tr>
<td>ISLAT</td>
<td>2</td>
<td>I</td>
<td>South latitude band Positive entries (1) - high (2) - low</td>
<td></td>
<td>DEG</td>
</tr>
</tbody>
</table>
PART III

LIST OF SUBROUTINES

AND

SUBROUTINE CALL STRUCTURE
# LIST OF ROUTINES IN SAGE

<table>
<thead>
<tr>
<th>NAME</th>
<th>FUNCTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. MAIN</strong></td>
<td>Main control</td>
</tr>
<tr>
<td><strong>2. RDCD</strong></td>
<td>Controls reading of data cards, checking and storing</td>
</tr>
<tr>
<td><strong>3. WEATH</strong></td>
<td>Reads NASA weather tape and writes weather data file</td>
</tr>
<tr>
<td><strong>4. INIT</strong></td>
<td>Performs initialization of data and reads header records from input files, and writes header on seg. ref. file</td>
</tr>
<tr>
<td><strong>5. READSW</strong></td>
<td>Reads a record on swath file</td>
</tr>
<tr>
<td><strong>6. WEATF</strong></td>
<td>Reads a record from the weather file</td>
</tr>
<tr>
<td><strong>7. LFPA</strong></td>
<td>Computes days (Zulu Time) 'common change'</td>
</tr>
<tr>
<td><strong>8. DAY</strong></td>
<td>Computes IDAY-DAY no. of year</td>
</tr>
<tr>
<td><strong>9. FZULU</strong></td>
<td>Given Zulu day, compute month, day and year</td>
</tr>
<tr>
<td><strong>10. SOL</strong></td>
<td>Compute EL sun elevation angle</td>
</tr>
<tr>
<td><strong>11. POTEN</strong></td>
<td>Determines potential access for a day</td>
</tr>
<tr>
<td><strong>12. ACCVER</strong></td>
<td>Verify access for a day</td>
</tr>
<tr>
<td><strong>13. IMAGE</strong></td>
<td>Compute image conditions and sun elevation angle</td>
</tr>
<tr>
<td><strong>14. SEGPRO</strong></td>
<td>Controls the processing for acquisitions for all days for one segment</td>
</tr>
<tr>
<td><strong>15. SEGREP</strong></td>
<td>Prints the output report for one segment</td>
</tr>
<tr>
<td><strong>16. WRITWT</strong></td>
<td>Write records on the weather file</td>
</tr>
<tr>
<td><strong>17. RDMIA</strong></td>
<td>Random number generator</td>
</tr>
<tr>
<td><strong>18. UNPKPK</strong></td>
<td>Data from or to IPC-file storage form</td>
</tr>
<tr>
<td>NAME</td>
<td>FUNCTIONS</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>19. EJECT</td>
<td>Restore page, print header</td>
</tr>
<tr>
<td>20. PAGER</td>
<td>Keep line count and restore page automatically</td>
</tr>
<tr>
<td>21. PIMOD</td>
<td>Convert an angle to be between 0-2π</td>
</tr>
<tr>
<td>22. DEGMOD</td>
<td>Convert an angle from radians to deg., min., sec.</td>
</tr>
<tr>
<td>23. RDWETR</td>
<td>Read NASA weather tape data into storage</td>
</tr>
<tr>
<td></td>
<td>(need a routine for UNIVAC and CDC)</td>
</tr>
</tbody>
</table>
SUBROUTINE CALL STRUCTURE

SAGE (MAIN)
   RDCD
   PAGER
   EJECT
   INIT
   WEATF
   PAGER
   READSW
   PAGER
   PAGER
   LFPA
   SEGPRO
   WEATF
   UNPKPK
   READSW
   DAY
   POTEN
   PIMOD
   ACCVER
   READSW
   PIMOD
   IMAGE
   ALPHA
   PAGER
   DEGMOD
   SEGREP
   EJECT
   PAGER
   FZULU
   DEGMOD
   WEATF
   READSW
   WEATH
   WRITWT
   RDWETR
   PAGER
   RDMIA
   UNPKPK
PART IV

SUBROUTINE DESCRIPTIONS AND FLOWCHARTS
SUPPLIED UTILITY ROUTINES

Routine Day

Call Day (IYMD, IDAY)
Given IYMD (3) where
IYMD (1) IS Day No.
IYMD (2) IS Month No.
IYMD (3) IS Year No.

Compute year day no. in IDAY

Routine PIMOD

Call PIMOD (A)
Convert ±A in radians to an angle 0-2π

Routine SOL (Entry ALPHA)

Call ALPHA (IFLAG)
For ephemeris usage as called by hector
computes ALPHAM and ALPHAT and IFLAG = 1

Routine PAGER (Entry Eject)

Call PAGER (NLINES)
Updates line count in NLINE with NLINES
NPAGE = 0 causes page to be restored prior to print.
NPAGE - page no.
HEADER- 80 char. 20A5
ICASE- case no.
KO - 6 print unit
INMAX is max no. of lines allowed
Initially NLINE should be set > LINMAX and NPAGE = 0
SUPPLIED UTILITY ROUTINES
(CONTINUED)

Call EJECT (NLINES)

Causes page to be restored automatically and then prints headers.

Routine CLDAY

Call CLDAY

Given IDAY-DAY no. of the year compute in LMO-the month
and in LDA the day no.
Need: IYEAR = 0 - Leap Year, ≠ 0 not Leap Year

Routine KEPLER

Call KEPLER (XM, XECC, XE, ERROR)

Given XM - Mean anomaly, XECC - eccentricity
Compute: E-eccentric anomaly, error = 0 means OK

Routine LFPA

Call LFPA [ FLDA, LMO, LYR, ALFGM (can be dummy), DAYS ]

Given: FLDA - day of month no., LMO - month no.,
LYR - year no. compute ALFGM - right ascension and
DAYS - Zulu day no.

Routine DEGMOD

Call DEGMOND (RAD, IDEG)

Given: angle rad in radians store the angle in deg., min., sec.,
in IDEG(1) - (3).

Routine FZULU

Call FZULU (IOATE, IOUT)

Given Zulu date in IDATE, compute year, month and day in
IOUT(1) - IOUT(3).

Routine RDMIA

Call RDMIA(FL, U)

Given double precision random no. seed in FL, compute random
no. U (0-1) based on uniform distribution.
SUBROUTINE MAIN BLOCK DIAGRAM

Initialize flags, CT, ISGCT, and IREFCT = 0

CALL RDCD(IE) to read, check and store data

IF IE 0 ERROR

IF IOPT 0, OR 0

CALL WEATH to read NASA weather tape and write weather file

WRITE STATUS INFORMATION AS PER P.D.

PRINT RAND NO. OUT

STOP

CALL INIT((I)ER) to control the initialization of files and data

IS IER 0 meaning error cond.

READ A RECORD FROM ISEG

IS COUNTRY 'ZZZZ'

WRITE STATUS INFORMATION

CALL WEATF(2) to close files

WRITE TRAILER RECORD ON IREF

WRITE EOF ON IREF REWIND ISEG, ISEG, ISEG, AND IREF

ADD 1 TO ISEG

MULTIPLY VEHICLE ALT. BY 10; STORE IN YALT

WRITE A RECORD ON IREF

ADD 1 TO IREFCT

IS ISEG 0

IS IREFCT TRUE

CALL SEGREP TO PRINT DATA FOR 1 SEGMENT

STOP
Subroutine RDCD

CALLING SEQUENCE:
Call RDCD (IE)

IE = 0 - OK, IE ≠ 0 - ERROR

PURPOSE:
The purpose of this routine is to read the header card and data card into working storage and then perform error checks on the input data.

INPUT: CONS COMMONS
       INP, KO

OUTPUT: FLAG COMMON
       HEADER, ICASE, ICSESW, ICSESG, ISTIME, NDAys, IOPT, LVEH, IREPTr, NMON,
       WORK COMMON
       DECR, RAND
       CONS: IVCT, IGRON, NLATSW

SUBROUTINES USED:
PAGER, EJECT

METHOD/PROCEDURE
1. Read header and store in HEADER, set IE = 0
2. Read data card ISWRTO, IGRDN, NLATSW, NDAys, LVEH, NMON, ISWRTO, IGRDN, NLATSW and RAND are read to temporary locations. If these quantities are 0 then set NDAys = 426, LVEH=1,2; NMON=12; and RAND=1.0, ISWRTO=1101, IGRDN=16000, NLATSW=100
3. Call EJECT (2)
4. Perform checks 4 and 5, 8 and set IE=0 if any errors.
5. Return
Subroutine WEATH

CALLING SEQUENCE:
Call WEATH

PURPOSE:
This routine controls the reading of the NASA weather tape and writing of the weather file.

INPUT:
WEATDT COMMON
    IGRIDPT, IPERWT
CONS COMMON
    IWRAPE
FLAG COMMON
    NMON
WORK COMMON: RAND

OUTPUT:
COMMON FLAG
    NMAX, IGRID
DIRAC COMMON
    IPC
COMMON WORK ICLDCV - Intermediate storage of PC data by day

SUBROUTINES USED: RDWETR, WRITWT, UNPKPK, RDMIA

METHOD/PROCEDURE
See flow chart.
SUBROUTINE WEATH BLOCK DIAGRAM

REWIND IWTAPE

SET I = 1, COUNT ON NO OF WEATHER RECORDS

CALL RDWETR (IREC) TO READ 1 REC FROM WEATHER TAPE

SET IGRID = IGRDPT

EXECUTE PROCEDURE 2 - 5 SEC, 3,3,2 P. DEF. GIVEN, IPERWT, RAND TO PRODUCE ICLDCV

CALL UNPKPK(0) TO COMPUTE DATA FROM ICLDCV INTO IPC

CALL WRITWT(1) TO WRITE 1 RECORD ON WEATHER FILE

ADD 1 TO I

CALL WRITWT(0) TO INITIALIZE WRITING OF WEATHER FILE

SET IREC = 1

RETURN

CALL WRITWT(2) TO CLOSE WEATHER FILE

IF I > IGRN

NO

YES
SUBROUTINE INIT BLOCK DIAGRAM

PURPOSE: THIS ROUTINE INITIALIZES CORE, FILES AND DOES SOME PRELIMINARY CALCULATIONS.

CALLING SEQUENCE: CALL INIT(IER)
IER = 0 - NO FATAL ERROR
IER #0 - FATAL ERROR

10 - 15
INITIALIZE COUNTERS
FLAGS
IER = 0, ICLDCV
IDEXSW AND
IDEXW ARRAYS TO 0

CALL WEAT(F(0))
TO READ HEADER
RECORD FOR
WEATHER FILE
AND CHECK TO
MAKE SURE FILE
IS THERE

CALL READSW(I/O)
TO READ HEADER
RECORD, CHECK
CASE ID COMPATIBILITY,
IF OK TO READ SEARCH
DATA RECORD

20 - 30
REWIND ISGF,
ISWR AND READ
THE HEADER RECORD
OF EACH, CHECK
ICSESG AND
ICSESW FILE CASE
SWATH, REE FILE CASE

35 - 40
MAKE SURE LVEH
LIST DOES NOT SPECIFY
A VEHICLE NO NOT ON
SWATH FILE VIA NVEH.
REARRANGE LVEH IN
ASCENDING ORDER

45
STORE ZULU TIME FOR
VEHICLE STARTS IN
ISWST ARRAY FROM
IV1TIM AND IV2TIM
USE CALL TO LFPA

CALL LFPA
TO CONVERT
ISTIME TO
ZULU IN ISDAY

50 - 55
CHECK TO
SEE IF ISDAY
IS 2 TO THE
EARLIEST ISWST
TIME

60
ADD ISDAY*30
TO EARLIEST
VEHICLE START
DAY
STORE IN ISLATS

ADD (INDAYS - 1)
TO ISDAY, STORE
IN IRUNE AS TEMP.
RUN END

ADD (NDAYS - 1)
TO EARLIEST
VEH ST. DAY
STORE IN ISWATE

IF ISWATE
IS < THAN
IRUNE, SET
IRUNE = ISWATE

RETURN
Subroutine READSW

CALLING SEQUENCE:

Call READSW (IND)

IND = 0 - initial, read header record and check for
correct file
IND = 1 - normal read 1 data record into working
storage
IND = 3 - termination - close file
activity
IND = 2 - read search table data record

PURPOSE:

This routine reads the header record or data record from the
swath table file - ISWTB and stores the data into working
storage. ISWTB is a direct access file

INPUT:
CONS COMMON
KO, ISWTB, NLATSW, ISWRTO
DIRAC COMMON
IDEXSW, A
WORK COMMON
IREV, IVEH
FLAG COMMON
JDAY, JCSESW, NODAY
ISWTB FILE - SWATH TABLE

OUTPUT:
FLAG COMMON
ISOSTR, NLAT, NVEH, IV1TIM, IV2TIM, NODAY
SWATH COMMON
LATNO, IALT, TIME1, DLONG1, DLONMX, DLONMN, INLAT,
ISLAT
METHOD/PROCEDURE

For options 0, 1, 2 array A is cleared

\[ \text{IND} = 0 \]
- Define file is given then the header record is read into A, the case no. is compared against ICSCSN and the first 2 words check to be 'SWATH TB' if not, an error message is printed with a STOP. ISOSTR, NLAT, INLAT, and ISLAT are stored and check LVEH vs. NVEH.

\[ \text{IND} = 1 \]
- IREC = record no. = \( (\text{NODAY} + 1) \times (\text{IVEH} - 1) + \text{JDAY} + 1 \). IREC is read into A and then using NLAT the data is moved from A to SWATH COMMON. NOWDS = 8*NLAT.

\[ \text{IND} = 2 \]
- IREC = record no. = \( (\text{NODAY} - 1) \times (\text{IVEH}-1) + \text{NODAY} + 2 \). Read search record from ISWTB (NOWDS = 8*NLAT) into A and move the search data to LATNO DLONMX and DLONMN using NLAT.

\[ \text{IND} = 3 \]
- Return
Subroutine WEATF

CALLING SEQUENCE:

Call WEATF (IND)

IND = 0 - read header record and check to make sure the file is weather file.
IND = 1 - normal read a physical record and move desired logical rec to IPC
IND = 2 - termination

PURPOSE:

This routine, given the grid no., will fetch the weather data for that grid from file IWTHF and store it into IPC for use by the application program.

INPUT:  
CONS COMMON
        IWTHF, IBLKF, NPREC, NLREC, KO
FLAG COMMON
        IGRID
DIRAC COMMON
        INDEXC, IB
The file as defined in sec. 2.2, item 4.

OUTPUT: FLAG COMMON
        NMAX
DIRAC COMMON
        IPC

METHOD/PROCEDURE
See flow chart

SUBROUTINES USED:
SUBROUTINE WEATF BLOCK-DIAGRAM

ENTER

5
CLEAR IB AND IPC TO 0

10

= 0
TEST IND FOR 0, 1, OR 2

= 2
TERM

= 1
NORMAL

15
RETURN

20
DEFINE FILE TO READ INDEX I' TO IDEXW

READ REC 1 INTO IB (IWTWF FILE)

25
STOP

30
MOVE IB(3) TO NMAX

1

YES

TEST IB(1), IB(2) = 'WEATHER'

NO

200
PRINT ERROR MESSAGE 6

100

COMPUTE IREC = (IGRID / IBLKF) + 1

COMPUTE INDEX = (IGRID / IBLKF) REM

IF INDEX = 0 SET IT = 5

READ IREC RECORD INTO IB

110
MOVE IB(I) INTO IPC

1
Subroutine POTEN

CALLING SEQUENCE:

Call POTEN (IND)

IND = 0 - potential access, IND ≠ 0 no access

PURPOSE:

This routine determines for a given day and vehicle whether there is a potential access of a given segment. It does this by reading the segment reference file and executing equation set 3.3.2 item 2.3.

INPUT:  SWATH COMMON

    DLONMN - Ω₂, LATNO, IALT

SEGFIL COMMON

    THETSG - θSEG, PHISEG - αSEG

CONS COMMON

    RADIAN, ISWR

SWATH REFERENCE FILE RECORD WITH FOLLOWING DATA FOR JDAY AND IVEH IN SWATH COMMON.

    KVEH, KDAY, NREV, TIMNOD - τNODE, PHINOD - φNODE

FLAG COMMON

    JDAY

WORK COMMON

    IVEH

OUTPUT:  WORK COMMON

    IREV, LAT, LAT1, PHINDE, DLNMIN, DLNMAX

METHOD/PROCEDURE

1. Set IND = 0
2. Read data record from ISWR file
3. Compute DLNMIN, DLNMAX, LAT, LAT1
4. Compute IREV where there is an access if no access set IND ≠ 0 return.
5. If access, store PHINOD(IREV) into PHINDE and TIMNOD(IREV) into TMNODE
6. Return

SUBROUTINES CALLED: PIMOD, AMAX1, AMIN1
Subroutine ACCVER

CALLING SEQUENCE:

Call ACCVER (IND)

IND = 0 - Access verified, IND ≠ no access

PURPOSE:

This routine verifies that a segment is actually accessed on JDAY for IVEH and IREV. It reads the record from ISWTB file specified by JDAY and IVEH, determines if an access is possible then computes IZDAY. See section 3.3.2 set 4 for equations.

INPUT:

FLAG COMMON
JDAY, ICDAY
WORK COMMON
LAT, LAT1, PHINDE, SA, IVEH, DECR
CONS COMMON
RADIUS
SEGFIL COMMON
THETSG
ISWTB SWATH FILE
   ORIGINALLY IN SWATH COMMON
IALT, TIME1, DLONG1

OUTPUT:

WORK COMMON
DELAT, DLONG, TIME, XTIME, XDLONG, XALT, SIGMA,
DDLNG1, IZDAY, DDLNGR, CHECKL, CHECKR, CHLONG,
TMNODE

METHOD/PROCEDURE

1. IND is set = 0
2. A call to READSW is made to get IALT, TIME1, DLONG1
3. DLONG1, TIME1, and IALT are moved into DLONG, TIME and ALT for latitude subscripts LAT and LAT1. Note IALT is integer KILOM * 10
4. Equation set 4 b is executed
5. If no access set IND ≠ 0 and return
6. ICDAY is stored in IZDAY
Subroutine ACCVER (Continued)

METHOD/PROCEDURE (continued)

7. Return

SUBROUTINES CALLED: READSW
Subroutine IMAGE

CALLING SEQUENCE:
Call IMAGE

PURPOSE:
This routine computes the imaging conditions of the access which are to be written out on the segment reference file. See section 3.3.2 of the problem description equation set 5, 6 for equations.

INPUT:
WORK COMMON
SA, ALT, XTIME, XDLONG, CHLONG, TMNODE

OUTPUT:
WORK COMMON
TS, YALT, CTA, EL

METHOD/PROCEDURE
1. Execute equation set 5.
2. Call ALPHA to compute EL.

SUBROUTINES CALLED: ALPHA
r

r~

1

28234-6028_R V-OO

Page 63

1:

SUBROUTINE SEGPHO BLl'CK DIAGRAM

PURPOse,

THIS IS A R.)UTINE WHICH CONTROLS -I-LL
'ROCESSI'iG FOR ONE SEGMENT.

C'lLING SEOUENCE· CALL SEGPROIISEGI
'SEG,' 0 - OK, 'SE~ 0 NO GOOD
ENTER

SET 'SEG-O
IACOCT- 0
(ACOUIS. CTRI
SET IV • I (VEH CTRI

CALL ACCVE R IIACI
TO VERIFY ACCESS
HASOCCURREO

TEST lAC
• 0 - ACCESS
~O- NONE

MOVE ICLDCVIIDAYI
INTO ICLOD

~o

(STHETSG OF
CURRENT SEGMENT
IN ACTIVE LAT,
BAND ASSPECIFIED
IN INLAT. ISLAT

Sl:T LAT"u· 0
FOR NORTH LAT
BAND," 1 FOR
S LAT. BAND

10

~,IOVE AcaUIS.
DATA,I.E.
IZDAY,IREV
CTA, ECT, INTO
AcaUIS COMMON

RETURN

15

•
COMPUTE IVER
FROM LVEHIIV)
CALL READSWI2)
TO GET SEARCH
DATA FOR I'IEH

IF ISVIDST':; ICD.W;;
ISLATS SET LATYP = 0 - N
IF ISLATS'~ ICDAY '" INLATE
SET LATYP = 2 -IN + SIIF
ICDAY > INLATE SET LATYPE
= 1 - 5, LATYP IND. ACTIV HEM.

130

50-60
EXECUTE CODe
166 - 215 FROM
EPHEM PROG
ROUTINe'MAIN
(DAY BOOKKEEP)

COMPUTE CURRENT
DAY OUANTITIES FOR
TIME BOOKKEEPING

--C.\LLDAYTOGET - - - IDAY FROM ISTIME,

~':tARR~ ~~~i~~,\~ME 11)
1900-1,ICDAY = I S D A Y I~9,,-

______.,

200
SKIP THE
UNPROCESSED
DAY RECORDS
ON ISWR

10
CALL WEATF(I)
TO READ WEATHER
DATA INTO IPC FOR
IGRID
REWIND ISWR; SPACE
BY HEADER RECORD

CALL
UNPKPK(21
TO UNPACK
IOCTO
ICLDCV

IFTHERE ARE
1 VEHICLES ON
THE SWATH REF.
FILE AND ONLY
THE 2ND ONE
. IS TO BE
PROCESSED,
SPACE BY THE
1ST VEHICLE ON
ISWR - SWATH
REF. FILE

SET 10
= 1- DAY
COUNTER

RETURN

•


Subroutine SEGREP

CALLING SEQUENCE:
Call SEGREP

PURPOSE:
This routine prints the data for all acquisitions for 1 segment.
See problem description for the format.

INPUT:
WORK COMMON
IACCCT
ACQUIS COMMON
IZDAYT, IREV, CTAT, ELT, ICLOD, ICEHT, TST, YALTT
SEGFIL COMMON
COUNTR, SEGMNT
CONS COMMON
KO

OUTPUT: See problem description Section 4.1.1

METHOD/PROCEDURE
1. Call EJECT(7) to restore page print run header
2. Print report headers
3. Print column headers
4. For each acquisition - IACCT
   Call PAGER(1)
   Call FZULU to convert IZDAYT to YR MON DAY
   Call DEGMOD to convert EL and CTA to DEG, MIN, SEC
   Print 1 line of data

SUBROUTINES CALLED: DEGMOD, EJECT, PAGER, FZULU
Subroutine WRITWT

CALLING SEQUENCE:

Call WRITWT (IND)

IND = 0 - write header record, IND = 1-normal, IND = 2,
   - termination

PURPOSE: This routine accumulates logical weather data records into
       a record block and writes the record on the direct access
       file IWTHF. This routine is called for each grid point.

INPUT:

CONS COMMON
   IWTHF, IBLKF, NPREC, NLREC
FLAG COMMON
   IGRID, NMAX
DIRAC COMMON
   IDEXW, I$B$, IPC

OUTPUT: Either 1 logical record IPC moved into 6 or 8 written out
        on the IWTHF file, see sec. 2.2 item 4. for format details
        in problem description.

SUBROUTINES USED:

METHOD/PROCEDURE

See flow chart.
SUBROUTINE WRITWT BLOCK DIAGRAM

10 - 15
CLEAR IB TO 0

MOVE 'WEATHER' TO IB(1), IB(2) AND NMAX TO IB(3)

DEFINE 'FILE:' TO INITIALIZE IDEXW AND FILE IWTHF

WRITE 1 PHYSICAL RECORD ON IWTHF

SET IWRT = 0 INDICATING NO WRITE FOR NEXT CALL

CLEAR IB TO 0

10
INIT. = 0

TEST IND FOR 0, 1 OR 2

TERM. = 2

WRITE PHYSICAL RECORD FROM IB ONTO IWTHF REC, IREC

100
TEST IWRT FOR FOR ≠ 0

#0
WRITE IB = REC ON FILE IWTHF

105
COMPUTE INDEX = (GRID) / IBLKF REM

160
CLEAR IB TO 0 AND SET IWRT TO 0

110
COMPUTE I = (INDEX - 1)* NLREC + 1

MOVE IPC TO IB(I)

1
SUBROUTINE UNPKPK BLOCK DIAGRAM (PAGE 1 OF 2)

CALLING SEQUENCE:
CALL UNPKPK(IND)
IND = 0 – PACK CLOUD DATA FROM ICLDCV INTO IPC
IND #0 – UNPACK CLOUD COVER DATA FROM IPC AND STORE IN ICLDCV

PURPOSE: DEPENDING ON IND THIS ROUTINE EITHER CONVERTS INTEGERS 0-8 IN
ICLDCV AND THEN PACKS THESE 4 BIT NUMBERS 8 TO A IPC WORD OR
CONVERTS AND UNPACKS THE INTEGERS FROM IPC INTO %*10 AND
STORES THEM INTO A ICLDCV WORD.

INPUT: WORK COMMON
ICLDCV
DIRAC COMMON
IPC
FLAG COMMON
NMAX
CONS COMMON
NLREC, IPCK

OUTPUT: SAME AS ABOVE EXCEPT FOR NLREC AND IPCK

METHOD/PROCEDURE
IND = 0

1 = 1
DAY COUNT

J = 1
IPC WORD COUNT

5
CLEAR IPC ARRAY TO 0

10
K = 1
COUNT ON ITEMS IN IPC(J)

IPC(D) = IPC(J) + ICLDCV/I%MULT(K)

ADD 1 TO I
20
IS K > IPCK

YES
ADD 1 TO J

RETURN

NO

IS I > NMAX

NO
ADD 1 TO K

YES

1
METHOD/PROCEDURE (CONTD)

IND #0

100

I = 1
DAD COUNT

110

J = 1
IPC WORD
COUNT

ITEMP1 = IPC(J)

K = 1
COUNT ON
ITEMS IN
IPC(J)

ITEMP = ITEMP1
MULT(K)

ITEMP1 = ITEMP1 - ITEMP*MULT(K)

ICLDCV(I) = ITEMP*125

ADD 1 TO J

ADD 1 TO K

YES

IS K > IPCK

NO

IS I > NMAX

RETURN

TABLE OF MULTIPLIERS/DIVISORS MULT

<table>
<thead>
<tr>
<th>K</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2^{28}</td>
</tr>
<tr>
<td>2</td>
<td>2^{24}</td>
</tr>
<tr>
<td>3</td>
<td>2^{20}</td>
</tr>
<tr>
<td>4</td>
<td>2^{16}</td>
</tr>
<tr>
<td>5</td>
<td>2^{12}</td>
</tr>
<tr>
<td>6</td>
<td>2^{8}</td>
</tr>
<tr>
<td>7</td>
<td>2^{4}</td>
</tr>
<tr>
<td>8</td>
<td>2^{0}</td>
</tr>
</tbody>
</table>
Subroutine RDWETR

CALLING SEQUENCE:

Call RDWETR (IRC)

IRC is logical record count 1 to 10 initialized to 1 by calling routine for first call only. Routine then adds to counter and resets it as needed to 1.

PURPOSE:

To read one physical record from the NASA tape unit IWTAPE and move 1 logical record into working storage

INPUT:

CONS COMMON NASA WEATHER TAPE
IWTAPE
WEATDT
IA - See Section 2.2 of problem description for format of each logical record

OUTPUT:

WEATDT COMMON
IGROPT, IPERWR

METHOD/PROCEDURE

Use NTRAN to perform the read from the weather tape.

SUBROUTINES CALLED: NTRAN, PAGER
PART V

SUBROUTINE LISTINGS
SUBROUTINE ALPHA(IFLAG)

CALL ALPHA(IFLAG)

INPUT = IFLAG = 1 COMPUTE RT. ASC. OF I

COMMON /CONS/
IAPlust, CIS*EMU, ECC*PI, PI2*PI32, RADIANT, TOPH1, NAME(?) + INC*KD*NIL*INF
2*LINMAX*1PAM, 1SE*IGN*IGN*INT*INT*INT*INT*INT*INT*INT*INT*INT*INT*INT
TICK, KUP*IVCT, TIL**LAT*SH*IGN*IGN
MT NAME
COMMON /FLAG/
THEADEY*IF, ICASE, ICSES*ICSES*ICSES*ICSES*TIME*TS DAY*NDAYS*UPT*LVEN*Z*FLAG
PIVET* forces*forces*forces*forces*forces*forces*forces*forces*forces*forces*forces
PLSAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY*DAY
PLSAY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY*XY
LOGICAL IEPH
COMMON /HANKY/
RAN
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.
.who
C DECLINATION OF SUN
65 XNUF = TWOPI + XNUF
70 TILS = ASTN(SINCIS)*SIN(XNUF-CNUV))
    DIF = XNUF-CNUV
    COSDIF = COS(DIF)
    SINDIF = SIN(DIF)
    IF(CUSDIF) 90, 75, 90
C PAASC OF TRUE SUN
85 ALPHAT = 3.141/2
    GD TO 150
90 ALPHAT = ATAN(COS(CIS)*SINDIF/CUSDIF)
    IF(CUSDIF) 105, 95, 95
95 IF(ALPHAT) 100, 150, 150
100 ALPHAT = TWOPI + ALPHAT
    GD TO 150
C RA. ASC. OF MEAN SUN
105 ALPHAT = (279.6666 + 360.0/365.25*DAY)/Radian
C PHIS = (43200.0 - TS)*TWOPI/86400.0 - ALPHAM*ALPHAT - EH
    ARPH = COS(TIHS + TILTS)
    ARPH = COS(PHIS - PHISE)
    ARPH = COS(TIHS + TILTS)
    EL = PI7 - ACOS(0.5*ARPH*(ARPH2 - 1.0) - (ARPH3*(ARPH2 + 1.0)))
1000 RETURN
END
SUBROUTINE DEGMOD(RAD, IDEG)

DEGMOD CONVERTS RADIANS INTO DEGREES, MINUTES, AND SECONDS
AND MODS THE DEGREES BETWEEN -360 TO +360

IDEG(1)=DFGFEFS
IDEG(2)=MINUTES
IDEG(3)=SECONDS
RAD = RADIANS
RADIAN = DEGREES PER RADIUS
HOUR = H+4 TEMPORARY STORAGE

INTEGER IDEG(3)

DATA RADIUS/57.295/R/
CONVERT RADIANS TO DEGREES, MINUTES, AND SECONDS

HOUR = RAD*RADIUS
IDEG(1)=HOUR
HOUR=ABS(HOUR-IDEG(1))*60.
IDEG(2)=HOUR
IDEG(3)=(HOUR-IDEG(2))*60.+5
C
IF (IDEG(3).LT.60) GO TO 10
IDEG(3)=IDEG(3)-60
IDEG(2)=IDEG(2)+1
GO TO 10
C
IF (IDEG(2).LT.60) GO TO 20
IDEG(2)=IDEG(2)-60
C
ISIGN=1
IF (IDEG(1).LT.0) ISIGN=-1
IDEG(1)=IDEG(1)+ISIGN
C
CONTINUE
C
IDEG(1)=MOD(IDEG(1)+360)
END

...
SUBROUTINE FZU(JDATE,TOUT)
C
C ROUINE FZUL
C
C PURPOSE TO CONVERT ZULU DATE TO YEAR MONTH AND DAY
C
C LINKAGE CALL FZUL(I1ATE,TOUT)
C
C I1ATE, BINARY INTEGER ZULU DATE
C
C TOUT, CALENDAR DATE DIMENSION 5 FOR INTEGER YEAR, MONTH AND DAY RESPECTIVELY
C
C ROUTINES CALLED NONE
C
C LOCAL VARIABLES
C
C IDAYS VECTOR CONTAINING NO. DAYS OF YEAR ON
C
C LAST DAY OF MONTHS 0 THROUGH 12 FOR NORMAI ZULU
C
C YEAR FOLLOWED BY 13 MONTHS OF LEAP YEAR
C
C NLEAP* NO. OF LEAP YEARS SINCE 1900 TO SPECIFIED FZUL
C
C NFLG* SUBSCRIPT INTO IDAYS VECTOR
C
C START=1* NOT LEAP YEAR
C
C START=19* LEAP YEAR
C
C JDAY* JULIAN DAY OF YEAR
C
C I* LOCAL USE
C
C REMARKS
C
C DIMENSION TOUT(5)
C
C D1MENSION IDAYS(24)
C
DATA IDAYS/0,51,59,90,120,151,182,212,243,273,304,335,365/,
C 1
C IF CURRENT YEAR ISLEAP YEAR-SET NFLG=1 OR 14
C
C I=NOUT(I)/4
C 3
C NFLG=1
C 4
C 10 IF (NOUT(I) - I) = 10,10,20
C NFLG=18
C 70 JDAY = IDATE - (NOUT(I) * 365) - NLEAP + 1
C C GET DAY MONTH FROM TABLE SEARCH
C 30 IF (JDAY-IDAYS(NFLG)) = 50*50*40
C 40 NOUT(I)=NOUT(I)+1
C 50 NFLG=NFLG-1
C 60 JDAY=JDAY-IDAYS(NFLG)
C IN I11;
C ENDF
C
SUBROUTINE IMAGE

THIS ROUTINE COMPUTES THE IMAGING CONDITIONS OF AN ACCESS

COMMON /HND/, RA, TEMP, I, P, L

IF (I+1.0 > TEMP) THEN
  TEMP = I + P
ENDIF

CALL ALPHA(10)
RETURN
000059  901 FORMAT(1H0+89H*** ICSFSW DOES NOT MATCH THE CASE NO ON THE SHATH INIT
000060     INPUT FILES OR SHATH FILES NOT MOUNTED) INIT
000061  19 IF LVLH(1) EQ. 0300 TO 40 INIT
000062        IF LVLH(2) EQ. 0300 TO 45 INIT
000063        IF LVLH(1) EQ. LVLH(2) GO TO 40 INIT
000064        IF LVLH(2) GT. LVLH(1) GO TO 45 INIT
000065         TLH(1) = LVLH(1) INIT
000066         TLH(2) = LVLH(1) INIT
000067         TLH(1) = TLH(2) INIT
000068         TLH(2) = TLH(2) INIT
000069         IF TLH(1) GT. 10 GO TO 45 INIT
000070        CALL PAF(2) INIT
000071               WRITL(KNF+90) INIT
000072  00 CALL PAF(2) INIT
000073             LED TO SHATH TABLES) INIT
000074  45 XVIT = TVIT1M(5) GO TO 50 INIT
000075        CALL I+PAFXVIT *IVITIM(2)+IVITIM(1)+X+DAYS) INIT
000076           LSXVIT(1) = DAYS INIT
000077             XVIT = TVIT1M(5) INIT
000078               CALL I+PAFXVIT*TVIT1M(2)+TVIT1M(1)+X+DAYS) INIT
000079                  LSXVIT(2) = DAYS INIT
000080                      STIM = 1STIME(1) INIT
000081                        CALL I+PAFXSTIM *1STIME(2)+1STIME(1)+X+DAYS) INIT
000082                          ISDAY = DAYS INIT
000083                              IF ISVST(1) LT. 1SWST(2) GO TO 50 INIT
000084                                  ISWST = TVST(2) INIT
000085                                    GO TO 54 INIT
000086                                 50 ISWST = TVST(1) INIT
000087                                    CALL PAF(2) INIT
000088                           WRITL(KNF+90) INIT
000089                                       CALL PAF(2) INIT
000090                                          WRITL(KNF+90) INIT
000091                                             CALL PAF(2) INIT
000092  903 FORMAT(1H0+89H*** ISTIME IS LESS THAN THE EARLIEST VEHICLE START TIME INIT
000093      1STIME INVITIM OR 1STTIME INIT
000094                         GO TO 500 INIT
000095  60 ISLATS = ISWST * ISOYST+30 INIT
000096        IPUN = ISDAY * NPATS + 1 INIT
000097           ISLATS = ISWST * NPAYS + 1 INIT
000098                 IF ISLATS GT. IT. IPUN = ISWST INIT
000099                   IMAJ = ISWST + 426 INIT
000100                     IF IMAJ GT. IT. INMAJ = IPUN INIT
000101                         INIT INTINIT
000102                           INGCT = IA+WIN INIT
000103                               INECE = IACX14 + 2 INIT
000104                                    IF ILE = WRITL(KNF+1)(XKMAK(1),1=1+2,ICASE*ISDAY*NPATS*NCRUP+ICGESW*ICSEGSM1N1T
000105                                            I+VITIM(I+ILLE+I+1ADL)) INIT
000106   500 LTH = 1 INIT
000107   505 INIT INIT
000110     END
SUBROUTINE LFPA(FLDA,LH,M,LYR,AFLG,M,DAYS)

GIVEN DAY, MONTH, YEAR - SUBR. LFPA RETURNS THE RIGHT ASCENSION OF
GREENWICH AT MIDNIGHT OF A GIVEN DAY

REFERENCE EPOCH IS 0 HOUR 1 JAN 1950
DATA RADIAN /57.29578/

COMPUTE DAYS IN FULL YEARS FROM EPOCH TO LYR.
DAYS TO 1 JAN 1964 IS 4744.

DAYS=74.
NOTF--LYR MUST BE GREATER THAN OR EQUAL TO (19)64

LST=LYR-1
DO 20 Z=631,1,LST
KRED=HDU((1-60)4)
IF(KRED.LT.0) GO TO 10

DAYS=FLDA+566
GO TO 20

10 DAYS=DAYS+565
20 CONTINUE
IF((LH-1170)+.90)+.30
20 CONTINUE
IF((LH-170)+.50)+.60
40 DAYS=DAYS+FLDA-1.
GO TO 270

50 DAYS=DAYS+FLDA+30.
GO TO 270

60 KRED=MOD((LYR-60)+4)
IF(KRED.GT.10) GO TO 70

DAYS=DAYS+59.
GO TO 90

70 DAYS=DAYS+59.
70 IF((LH-40)+.20)+.90
70 IF((LH-1)+.60)+.30
170 DAYS=DAYS+FLDA-1.
GO TO 270

100 DAYS=DAYS+FLDA+41.
GO TO 270

170 DAYS=DAYS+FLDA+41.
GO TO 270

200 DAYS=DAYS+FLDA+92.
GO TO 270

210 DAYS=DAYS+FLDA+122.
GO TO 270

220 DAYS=DAYS+FLDA+153.
GO TO 270

230 DAYS=DAYS+FLDA+184.
GO TO 270

240 DAYS=DAYS+FLDA+215.
GO TO 270

250 DAYS=DAYS+FLDA+247.
GO TO 270

260 DAYS=DAYS+FLDA+278.
270 CONTINUE
FUNCTION PI0RD(A)
CGLT POSITIVE ARGUMENT OF AN ANGLE IN RADIANS BETWEEN 0 AND 2PI
C
B=6.2831853
10 IF(A)<0 A=0
20 IF(A)>2*PI A=2*PI
30 IF(A<0) A=0
40 IF(A>2*PI) A=0
50 RETURN
END
SUBROUTINE PUTEN(NIND)
THIS SUBROUTINE DETERMINES WHETHER THERE IS A POTENTIAL ACCESS ON

COMMON/SHATH/

NTHP(1,100,3)+DLOM(1,100,3)+DLOM(100)+DLOM(100)+TIMNOD(17)*

COMMON/SEGFL/

1COUT(F[?]+THESEG,PHISEG,IREG,IZONE,ISTRAT,ISUPST,SEGNT,ITHPRL(6)

COMMON/CONS/

1NEW

INITFY SELUNT

COMMON/IUP/

SF11

COMMON/IUL/

SF12

1NEW

1HNU(15,15,CH,LECCF,PI,PI32,RAJIAN,ITWFP,NAMEF(?)+WO+KONLINFCONS

COMMON/IRAG/

SF18

1HNU(10,ICASE,ICESG,ICSEG,ISTTIME(9)+ISDAY+NDAYS+IOPL+LYLM(2)+FLAG

COMMON/LAG/

SF23

1TIMH(1,1CHIN+DLOMAX+DLOM(3)+TIME(3)+A13(2)+PHINOL,SA(5)+

COMMON/FAT/

SF74

2UTA*TIM(3)+XUONG(3)+XAT+SIGMA+UPHOL+CHECKL+CHECKRT+FAC

COMMON/ALAT/

SF75

4CLD-UALT

SF77

DOUBLE YLSTION RAND

COMMON/PAT/

SF79

LDM = 0

COMMON/PFTN/

5DO 10 I=1,IMAT

COMMON/AT/

SF11

1ATNO = IATNO(1)

COMMON/HAPIAN

SF11

1I = 1

PHIN = PHN

SF11

1ATNO = IATNO(1)

PHIN2 = PHIN2

SF11

IF (THSG .LE. RLAT1 .AND. THETSG .GE. RLAT2) GO TO 15

10 CONTINUE

15 LAT = 11

PHIN = PHN

PHILF = PHILF

PHISEG = PHISEG

FOR 12 = 1 TO 25

CONTINUE

END = 1

ENDIF

25 IREV = ISAV

DO 20 I=1,IMREV

DO 20 I=1,IMREV

DO 20 I=1,IMREV

DO 20 I=1,IMREV

DO 20 I=1,IMREV

Do 20 I=1,IMREV

END
000001 C
000002 SHIODR: INR RDM1A(FL*U)
000003 C
000004 DATA C1*G*ONE*Z*EO35*ONE*Z*EO35*ONE
000005 T
000006 XMOD(T) = UMOT(T+1.054)
000007 YMOD(T) = UMOT(T+52164.0)
000008 T = H
000009 IF T .LT. ZEHU(T) I = ONE
000010 H = YMOD(T)
000011 H1 = T - H2
000012 T = XMOD(XMOD(C1*H2+C2*R1) + C2*H2)
000013 U = T/YHOT5
000014 FL = T
000015 RTURN = 1
000016 END
SUBROUTINE UNPKPK(IND)

COMMON /WORK/ RANK
COMMON /ANG/ T,U
COMMON /TH/ TH1,TH2
COMMON /TIME/ TIME(4)
COMMON /LAT/ LAT(3)
COMMON /MID/ MID(2)
COMMON /DAYS/ DAYS(2)
COMMON /DAY/ DAY(2)
COMMON /UPP/ UPP(2)
COMMON /INT/ INT(2)
COMMON /IN/ IN(2)
COMMON /INN/ INN(2)
COMMON /N/ N(2)
COMMON /THR/ THR(2)
COMMON /THR1/ THR1(2)

10 DATA TIME(3) = 1,2,3
60 DATA LAT(2) = 60.0,30.0
70 DATA UPP(2) = 1.0,0.5
80 DATA INN(2) = 1.0,0.5
90 DATA H(2) = 1.0,0.5
100 DATA N(2) = 5000.0,2500.0
110 DATA THR1(2) = 1.0,0.5

20 K = IND
30 J = 1
40 DO 5 N = 1, K
50 M = N/2
60 IF (M .GT. K) GO TO 5
70 IF (N .EQ. M) GO TO 120
80 IF (M .GT. N) GO TO 130
90 IF (N .GT. M) GO TO 135

110 DO 20 K = 1, J
20 J = J + 1

120 DO 20 K = 1, M
20 K = K + 1

130 DO 20 K = 1, M
20 K = K + 1

135 DO 20 K = 1, M
20 K = K + 1

SUBROUTINE UNPACKS CLOUD COVER DATA BETWEEN ICLDCV AND UPKPK

COMMON /WORK/ RANK
COMMON /ANG/ T,U
COMMON /TH/ TH1,TH2
COMMON /TIME/ TIME(4)
COMMON /LAT/ LAT(3)
COMMON /MID/ MID(2)
COMMON /DAYS/ DAYS(2)
COMMON /DAY/ DAY(2)
COMMON /UPP/ UPP(2)
COMMON /INT/ INT(2)
COMMON /IN/ IN(2)
COMMON /INN/ INN(2)
COMMON /N/ N(2)
COMMON /THR/ THR(2)
COMMON /THR1/ THR1(2)

10 DATA TIME(3) = 1,2,3
60 DATA LAT(2) = 60.0,30.0
70 DATA UPP(2) = 1.0,0.5
80 DATA INN(2) = 1.0,0.5
90 DATA H(2) = 1.0,0.5
100 DATA N(2) = 5000.0,2500.0
110 DATA THR1(2) = 1.0,0.5

20 K = IND
30 J = 1
40 DO 5 N = 1, K
50 M = N/2
60 IF (M .GT. K) GO TO 5
70 IF (N .EQ. M) GO TO 120
80 IF (M .GT. N) GO TO 130
90 IF (N .GT. M) GO TO 135

110 DO 20 K = 1, J
20 J = J + 1

120 DO 20 K = 1, M
20 K = K + 1

130 DO 20 K = 1, M
20 K = K + 1

135 DO 20 K = 1, M
20 K = K + 1
000001 SUBROUTINE WEATH
000002 C THIS ROUTINE CONTROLS THE GENERATION OF THE WEATHFR FILE
000003 COMM /MATUT/
000004 IATT(40)+IR(50)+IP防护(9,12)
000005 COMM /CONS/
000006 RELATED/CIS,CNVX+CCF,PI+PT+PS+RADIAN+THOM1,NAME(2)+1UP+K0+ONLINECONS
000007 write(3)*SPACE+IGSF+LSH+_INH+1MHH*1IHF+1MTAPE+INCR+1B+1LKFK
000008 COMM /MATUT/
000009 COMM /ONL/
000010 COMM /FLAG/
000011 IF(10).ICASS+LS8,16,ICSF;E+1STICE(5)+TSDAY+DU0S+1UP+LVL0(2)+E0AG
000012 IF(10).ICASS+LS8,16,ICSF;E+1STICE(5)+TSDAY+DU0S+1UP+LVL0(2)+E0AG
000013 IF(10).ICASS+LS8,16,ICSF;E+1STICE(5)+TSDAY+DU0S+1UP+LVL0(2)+E0AG
000014 CICASS+LS8,16,ICSF;E+1STICE(5)+TSDAY+DU0S+1UP+LVL0(2)+E0AG
000015 COMPL/ADJ/
000016 IF(10).ICASS+LS8,16,ICSF;E+1STICE(5)+TSDAY+DU0S+1UP+LVL0(2)+E0AG
000017 IF(10).ICASS+LS8,16,ICSF;E+1STICE(5)+TSDAY+DU0S+1UP+LVL0(2)+E0AG
000018 IF(10).ICASS+LS8,16,ICSF;E+1STICE(5)+TSDAY+DU0S+1UP+LVL0(2)+E0AG
000019 MAKE_T/ADJ/
OF POOR QUALITY

ORIGINAL PAGE IS

FULL PAGE-PRINTING A LOGICAL OR PHYSICAL RECORD ON THE MEANER

PRINT

Page 105.
28234-6028-RU-00
# Table of Contents

<table>
<thead>
<tr>
<th>Part</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part I</td>
<td>Problem Description</td>
<td>108</td>
</tr>
<tr>
<td>Part II</td>
<td>Common Block Definitions</td>
<td>126</td>
</tr>
<tr>
<td>Part III</td>
<td>List of Subroutines and Subroutine Call Structure</td>
<td>134</td>
</tr>
<tr>
<td>Part IV</td>
<td>Subroutine Descriptions and Flowcharts</td>
<td>137</td>
</tr>
<tr>
<td>Part V</td>
<td>Subroutine Listings</td>
<td>165</td>
</tr>
</tbody>
</table>
PART I

PROBLEM DESCRIPTION
SACS PROBLEM DESCRIPTION

1.0 SCOPE

This document describes the requirements and processing logic for the Segment Acquisition Selector Program (SACS). This program is an integral part of the Large Area Crop Inventory Experiment (LACIE) System.

1.1 Program Capabilities. This program selects data from the Segment Reference File based upon data input manually and from a Crop Window File. It writes the extracted data to a Data Acquisition file and prints two summary reports describing the selected data. A processing flow diagram is shown in Figure 1-1.

1.2 Program Development and Organization. The program will be developed on the CDC 6600 Time Sharing System using ANSI Standard FORTRAN. The development procedures provide for adherence to the following guidelines:

a. Capabilities used in CDC FORTRAN will be compatible with UNIVAC 1108 compiler capabilities to provide for conversion with minimum amount of effort.

b. The program, its data and system subroutines will require no more than 24K words of core for the UNIVAC 1110.

c. Subroutines will be no longer than 100 FORTRAN statements and will clearly identify in a header comment block any PDP incompatibilities.
d. Variable naming conventions - Each variable will be started with one letter identifying its source or the common block in which it resides. It will be followed by 4 or 5 characters identifying the variable a prefix of I or X will be used to denote integer or real respectively, if required.

1.3 Operational Assumptions. The file restrictions are outlined in the file descriptions (see Appendix A Segment Reference Data File, Appendix B Crop Window File and Appendix C Data Acquisition File). Additional restrictions are outlined as follows:

a. All detected errors cause the program to abort and the output files and reports are terminated without completion.

b. The following program data maximums are checked.
1. Only one country may be selected for any one program run.
2. Only one case of input parameters will be processed against a single Segment Reference Data File and Crop Window File.
3. A maximum of 426 days is allowed for the acquisition period.
4. A maximum of 4 windows is allowed.
Figure 1-1  SACS Processing Flow Diagram
a. Name - SEG REFER

b. Access Method - Fixed length record sequential access method, standard FORTRAN access.

c. Status or retention cycle - not applicable.

d. Sort - A Header Record precedes the file and is followed by data records then a trailer record. The data records are in sort (from major to minor) as follows:
   1. Country ID (ZZZZ for trailer record)
   2. Region ID
   3. Zone ID
   4. Strata ID
   5. Substrata ID
   6. Segment ID

e. Media - Disk or Tape

2.2.2 Crop Window File. This input file contains a set of time windows for each geographical segment that specify growing periods for a crop. The file is generated by the LACIE Utility Maintenance Program (LUMP) and is used solely by this program (SACS) for the selection of data in the SEG REFER file. A detailed file description is found in Section 2.4 of the Users Manual. The file attributes are summarized as follows:

a. Name - CROPWIND


c. Status or Retention Cycle - Not applicable

d. Sort - A header record precedes the file and is followed by data records then a trailer record. The data records are in sort (from major to minor) as follows:
   1. Country ID (ZZZZ for trailer record)
   2. Region ID
   3. Zone ID
   4. Strata ID
   5. Substrata ID
2.0 INPUTS

Inputs to this program are supplied by the following:

a. Segment Reference Data File
b. Crop Window File
c. Manually generated parameter card input

2.1 Card Inputs

2.1.1 Card Input Data Quantities. Each parameter value entered via the input cards must be entered unless explicitly stated. All integers must be right justified, all hollerith fields must be left justified. Each parameter value is outlined as follows:

a. XITTL - Card 01, Columns 1 through 48. Contains the case title that is to be printed at the top of each output page.

b. IACASE - Card 02, Columns 1 through 4. Contains the integer case number that identifies the ACQUISI file and each printed output page.

c. ISCASE - Card 02, Columns 5 through 8. Contains the integer case number of the Segment Reference Data File (if 0 or blank no check on input case No. will be made).

d. INCASE - Card 02, Columns 9 through 12. Contains the integer case number of the Crop window input file (if 0 or blank no check on input case No. will be made).

e. IPCC - Card 02, Columns 13 through 15. Contains an integer that specifies the maximum percent cloud cover to be used in selection of ACQUISI file segments. This number is expressed in tenths of a percent (e.g., 15.4% would be expressed as 154). All ACQUISI File cloud cover percents must be less than or equal to this value.
f. ISANG - Card 02, Column 16 through 19. Contains an integer that specifies the minimum degrees sun angle to be used in selection of ACQUISSI file segments. This number is expressed in hundreds of degrees and ranges between 0 and 90 degrees (e.g., 12.50 degrees would be expressed as 1250). All ACQUISSI file sun angles must be greater than or equal to the specified angle.

g. ICOUN - Card 02, columns 20 through 23. Contains 4 hollerith characters that identify the country to be selected.

h. IREG - Card 02, Columns 24 through 26. Contains a three digit integer that identifies the region to be selected.

i. IZONE - Card 02, Columns 27 through 29. Contains a three digit integer that identifies the zone to be selected. A zero entry allows all zones in a region to be selected.

j. Output options, Columns 30 through 31. Contain a logical flag that specifies whether the program reports are to be generated. A T specifies print the report; an F specifies don't print the report.

- IPRPT-30- Daily Processing Load Report
- ICRPT-31- Crop Window Report (segment acquisition)

2.1.2 Card Formats. The SACS program requires two input cards. Each card has fixed field inputs. The card formats are shown in Figure 2-1 and each field is described in Section 2.1.1.

2.1.3 Deck Setup. Each of the two input cards must be provided and must be supplied in card number order (see columns 79 and 80).

2.1.4 Rules for Entering Data on Cards. (See 2.1.1)

2.2 Input Files.

2.2.1 Segment Reference Data File. This input file is generated by the Segment Access Generator Program and contains a data base of crop status for selected geographical segments of the world. The file is described in Section 2.4 of the Users Manual. The file attributes are summarized as follows:
Figure 2-1 Input Card Formats
3.0 PROCESSING

3.1 Overview. The program control flow for the SACS program is shown in Figure 3-1. The routines that perform this processing are summarized as follows:

a. SACS - Main control routine that controls the program flow and sets up all vendor dependent program variables.

b. AINIL - Reads the manually generated card input parameters and performs all file label processing.

c. AREAD - Reads data records from both the Crop Window File and the Segment Reference Data File until a Segment Reference Data file record for a required substrata is located or an end of file or zone is encountered on either file.

d. APRPT - Prints the Daily Processing Load Report.

e. AFIND - Selects a Segment Reference Data file record based upon the crop time window and the cloud cover sun angle criteria.

f. ACHRPT - Writes the Segment Data Acquisition report.

g. AWRTA - Writes the data and trailer records of the Data Acquisition file.

h. ABARF - This routine writes all error messages.

i. FZULU - Converts a date from zulu to MM DD YY
2A

Sort Access Dates into increasing Order

Acquisition date within Window Period

Yes

Advance Access Counter by

Training Segment

Yes

Cloud Cover ≥ maximum

No

Sun Angle < minimum

Yes

Add 1 to acquisition counter, save access index

Any previous acquisitions of this segment within the current window

No

Yes

To Next Access for this Window

To Next Window

25 acquisitions for this window

≤ 25
Write Data Acquisition File and Print Segment Acquisition Report

Read New SEG REFER Record

3A

3B

1A
4.0 OUTPUT

Output from the SAC's program consists of the following:

a. Data Acquisition File (unit IAFILE)
b. Crop window report, under option control (see ICRPT, printed output unit IOFILE)
c. Daily Processing Load Report under option control (see IPRPT, printed output unit IOFILE)

4.1 Print Data.

4.1.1 Reports.

4.1.1.1 Control Card Echo Print

4.1.1.2 Crop Window Report. The format of the Crop Window Report is shown in Figure 4-1.

4.1.1.3 Daily Processing Report. The format of the Daily Processing Report is shown in Figure 4-2.

4.1.2 Card Outputs. Not Applicable.

4.1.3 Output Files. This program outputs the Data Acquisition File (see Section 2.4 of the Users Manual) the file indexes selected segment acquisitions in the Segment Reference Data File. The file attributes are summarized as follows:

a. Name - ACQUISI

b. Access Method - Fixed length record sequential access method standard FORTRAN access.

c. Status or Retention cycle - Not applicable.

d. Sort - A header record precedes the data records and they are followed by a trailer. The data records are in sort (from major to minor) as follows:

The print format is:

HEADER

AAAA --- > AAA

ICASE ICESR ICSECW IPCC ISANG ICOUN IREG IZONE IPRPT ICRPT
NNNN NNNN NNNN NNN NNNN NNNN NNN NNN NNN L L
1. Country ID
2. Region ID
3. Zone ID
4. Strata ID
5. Substrata ID
6. Segment ID
<table>
<thead>
<tr>
<th>CASE XXXX</th>
<th>SEGMENT ACQUISITION DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>18 Selected Acquisitions</td>
</tr>
<tr>
<td>1</td>
<td>Single Space Between Lines of Same Segment</td>
</tr>
<tr>
<td>2</td>
<td>Double Space Between Segments</td>
</tr>
<tr>
<td>5</td>
<td>ID-TOTAL ACCESS</td>
</tr>
<tr>
<td>12</td>
<td>TOTAL XXXX</td>
</tr>
<tr>
<td>18</td>
<td>SELECTED ACQUISITIONS XXXX</td>
</tr>
<tr>
<td>33</td>
<td>COUNTRY XXXX</td>
</tr>
<tr>
<td>34</td>
<td>REGION XXXX</td>
</tr>
<tr>
<td>45</td>
<td>ZONE XXXX</td>
</tr>
</tbody>
</table>

**SPACING**

1. Single Space Between Lines of Same Segment
2. Double Space Between Segments
3. If no acquisitions = 0, selected acquisitions = 0, and none selected, print "ALL ILLEGAL"

**PAGING**

1. 39 Lines/Page CDC 6600
2. 58 Lines/Page UNIVAC, PDP

**PRINTING**

1. If no acquisitions = 0, print "NONE"
Figure 4-2 Format of Daily Processing Load Report

<table>
<thead>
<tr>
<th>RELATIVE NO.</th>
<th>DAY</th>
<th>ACQUISITIONS</th>
<th>RELATIVE NO.</th>
<th>DAY</th>
<th>ACQUISITIONS</th>
<th>RELATIVE NO.</th>
<th>DAY</th>
<th>ACQUISITIONS</th>
<th>RELATIVE NO.</th>
<th>DAY</th>
<th>ACQUISITIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>02</td>
<td>03</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>51</td>
<td>51</td>
<td>76</td>
<td>76</td>
<td>76</td>
<td>76</td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
<td>XXXXX</td>
</tr>
</tbody>
</table>

UP TO 6 PAGES EACH PAGE CONTAINING UP TO 100 DAYS
5.0 ERROR PROCESSING

All fatal errors detected by this program shall result in termination of program execution. The detected errors are listed in Table 5-1. The processing logic is described as follows:

a. Each time an error is detected, the routine ABARF is called with the error ID.

b. The routine prints the error and forces the program to terminate execution if the error is fatal or returns control if the error is non-fatal.

Non-fatal errors shall be printed and execution continued.
**Table 5-1 Error Messages**

<table>
<thead>
<tr>
<th>ERROR ID</th>
<th>MESSAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Cropwind file has invalid label name xxxxxxxx</td>
</tr>
<tr>
<td>02</td>
<td>Seg refer file has invalid label name xxxxxxxx</td>
</tr>
<tr>
<td>03</td>
<td>Cropwind input case xxx not equal to label case xxx</td>
</tr>
<tr>
<td>04</td>
<td>Seg refer input case xxx not equal to label case xxx</td>
</tr>
<tr>
<td>05</td>
<td>Invalid country input xxxx cropwind xxxx seg refer xxxx</td>
</tr>
<tr>
<td>06</td>
<td>No data selected for ACQUISI file</td>
</tr>
<tr>
<td>07</td>
<td>Seg refer acquisition day too large xxx, max-xxx, record=xxxxx</td>
</tr>
<tr>
<td>08</td>
<td>Input cards invalid or out of sequence xxxxxxx (Column 75-80)</td>
</tr>
<tr>
<td>09</td>
<td>Input sun angle less than 0 or greater than 90.00 xxxxxx</td>
</tr>
<tr>
<td>10</td>
<td>Percent cloud cover less than 0 or greater than 100.0 xxx</td>
</tr>
<tr>
<td>11</td>
<td>No. days in study exceeds 426</td>
</tr>
<tr>
<td>12</td>
<td>Crop window is missing a zone xxx</td>
</tr>
</tbody>
</table>

*Non-fatal error message*
PART II

COMMON BLOCK DEFINITIONS
# Common Storage Allocation

**Name** PBLK  
**Size** 

**Function** This Common Block Contains Potpourrie Variables

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPEOF</td>
<td>1</td>
<td>L</td>
<td>End of file flag ( F = ) more data, ( T = ) end file</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPFTF</td>
<td>1</td>
<td>I1</td>
<td>First time Flag ( 0 = ) first time, ( 1 = ) subsequent times, ( -1 = ) last time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPTYPE</td>
<td>1</td>
<td>I1</td>
<td>Type wheat ( (0 = ) unidentified, ( S = ) spring, ( 1 = ) winter) initially zero</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPERR</td>
<td>1</td>
<td>I3</td>
<td>Acquisition day when it exceeds maximum</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBCD</td>
<td>14</td>
<td>11A1</td>
<td>Numbers in Hollerith ( 0, 1, 3 \ldots 9, ) blank, ( - , 0 , \ldots , ) Comma</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ITEN</td>
<td>5</td>
<td>I5</td>
<td>Powers of 10 ( 1, 10, 100, 1000, 10000 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPEFLG</td>
<td>13*1</td>
<td>12L</td>
<td>Flag to denote whether errors 1 through 12 are fatal ( (T = yes; F = no) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPWRL</td>
<td>150*1</td>
<td></td>
<td>Integer work area</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XPWRK</td>
<td>150*1</td>
<td></td>
<td>Real work area</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

Name: IBLK  
Size:  
Page 1 of ___

**Function**  This Common Block Contains all the Manually Generated Input Parameters and the Vendor Dependent Variables

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICID</td>
<td>3*2</td>
<td>3A2</td>
<td>Columns 75-80 input card</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XITTL</td>
<td>12, 1:</td>
<td>12A4</td>
<td>Report case oriented page title</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IACASE</td>
<td>1</td>
<td>A4</td>
<td>Case number that identifies the acquisition data file</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISCASE</td>
<td>1</td>
<td>I4</td>
<td>Case number that identifies the segment reference data file (0, implies case number is not used)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IWCASE</td>
<td>1</td>
<td>I4</td>
<td>Case number that identifies the crop window file (0, implies case number is not used)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPCC</td>
<td>1</td>
<td>I3</td>
<td>The maximum percent cloud cover to be used in selection of acquisition data file segments, the number is in tenths of a percent</td>
<td>10*90</td>
<td></td>
</tr>
<tr>
<td>ISANG</td>
<td>1</td>
<td>I4</td>
<td>The minimum degrees of sun angle to be used in selection of acquisition data file segments. Hundreds of degrees and ranges between 0 and 90 degrees sun angle in radians</td>
<td>Deg*100</td>
<td></td>
</tr>
<tr>
<td>XICOUN</td>
<td>1</td>
<td>A4</td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IREG</td>
<td>1</td>
<td>I3</td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IZONE</td>
<td>1</td>
<td>I3</td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPRPT</td>
<td>1</td>
<td>L</td>
<td>Processing load report flag, T=print</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICRPT</td>
<td>1</td>
<td>L</td>
<td>Crop window report flag, T=print</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>IIFILE</td>
<td>1</td>
<td>12</td>
<td>Fortran unit number for manually generated input</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IOFILE</td>
<td>1</td>
<td>12</td>
<td>Fortran unit number for crop window and segment acquisition reports</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISFILE</td>
<td>1</td>
<td>12</td>
<td>Fortran unit number for segment reference data file</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICFILE</td>
<td>1</td>
<td>12</td>
<td>Fortran unit number for crop window file</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IAFILE</td>
<td>1</td>
<td>12</td>
<td>Fortran unit number for data acquisition file</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INLINE</td>
<td>1</td>
<td>12</td>
<td>Number of lines per page</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XTWTTL</td>
<td>5, 8</td>
<td>5A4</td>
<td>Crop window titles 4 spring windows followed by 4 winter</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### COMMON STORAGE ALLOCATION

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACRN</td>
<td>1</td>
<td>F6</td>
<td>Current record number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AFNAM</td>
<td>2, 1</td>
<td>2A4</td>
<td>File name initialized by data statement &quot;ACQUI$&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AFTRL</td>
<td>1</td>
<td>IA4</td>
<td>TRAILERID.&quot;ZZZZ&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IACNT</td>
<td>4, 1</td>
<td>I2</td>
<td>Total number of accesses for each window. IACNT(1)=window 1,...</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>IACNT(4)=WINDOW 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IATOT</td>
<td>1</td>
<td>I3</td>
<td>Total number accesses all windows</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IAPNT</td>
<td>25, 4</td>
<td>I3</td>
<td>Pointer to the segment reference file that identifies an acquisition entry that has been selected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IAPAG</td>
<td>1</td>
<td>I3</td>
<td>Current page number (initially 0)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IANLIN</td>
<td>1</td>
<td>I2</td>
<td>Number of lines remaining on page (initially 0)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANACQ</td>
<td>426</td>
<td>R1</td>
<td>Number of acquisitions per day. Subscript equals day relative to given start date (ISDATE)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

**Name**: CBLK  
**Size**:  
**Page** of **Page**

**Function**: This common block contains the data elements from the crop window file.

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICCRN</td>
<td>1</td>
<td>FG</td>
<td>Current record number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CFNAMI</td>
<td>2, 1</td>
<td>2A4</td>
<td>File name that must appear in header initially set to &quot;CROPWIND&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CFNAM2</td>
<td>2, 1</td>
<td>2A4</td>
<td>Actual name of file</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICCASE</td>
<td>1</td>
<td>D</td>
<td>Case number from file header</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICWIND</td>
<td>1</td>
<td>I</td>
<td>Number of crop windows (8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCOUN</td>
<td>1</td>
<td>A4</td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICREG</td>
<td>1</td>
<td>D</td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICZONE</td>
<td>1</td>
<td>D</td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICSTRA</td>
<td>1</td>
<td>I</td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICSUBS</td>
<td>1</td>
<td>I</td>
<td>Substrata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICLOW</td>
<td>8, 1</td>
<td>I</td>
<td>Start date relative to Jan. 1, 1950</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICHI</td>
<td>8, 1</td>
<td>I</td>
<td>Stop date relative to Jan. 1, 1950</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Note first 4 represent winter wheat windows  
last 4 represent spring wheat windows.
### COMMON STORAGE ALLOCATION

**Name** SBLK  
**Size**  
**Page** of  

**Function** This common block contains the variables extracted from the segment reference data file (Seg Refr)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCRNR</td>
<td>1</td>
<td>FG</td>
<td>Current record number</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| SFNAM1 | 2,1       | 2A4    | File name that must appear in header 'seg. refr'
| SFNAM2 | 2,1       | 2A4    | Actual name of file                             |        |       |
| SFTRL  | 1         | 4A4    | Trailer ID "ZZZZ"                               |        |       |
| ISSCAS | 1         | I3     | Case number from file header                    |        |       |
| ISDATE | 1         | I5     | Reference date, integer no. of days since 1 Jan. 1950 |
| ISDAY  | 1         | I3     | Number of days in study (1 to 426)              |        |       |
| ISNAC  | 1         | I3     | No. of acquisitions in record                   |        |       |
| SCOUNT | 1         | 1A4    | Country ID                                      |        |       |
| ISREG  | 1         | I3     | Region ID                                       |        |       |
| ISZONE | 1         | I3     | ZONE ID                                         |        |       |
| ISSTRA | 1         | I4     | Strata ID                                       |        |       |
| ISUBS  | 1         | I4     | Substrata ID                                    |        |       |
| ISSEG  | 1         | I5     | Segment ID                                      |        |       |
| SPCSW  | 1         | F6.2   | Percent spring wheat                            |        |       |
| SPCWW  | 1         | F6.2   | Percent winter wheat                            |        |       |
| ISTRN  | 1         | I1     | Training segment indicator, 0=normal, 1=training |        |       |
**COMMON STORAGE ALLOCATION**

**Name** SBLK

**Size**

**Page 2 of 2**

**Function** This common block contains the variables extracted from the segment reference data file (Seg. Refr.)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISRDAY</td>
<td>150, 1</td>
<td>I3</td>
<td>Acquisition day for each of 150 acquisition entries</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSANG</td>
<td>150, 1</td>
<td>I4</td>
<td>Sun angle radians</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISPCC</td>
<td>150, 1</td>
<td>I4</td>
<td>Percent cloud cover * 10</td>
<td></td>
<td>90*90</td>
</tr>
</tbody>
</table>
PART III

LIST OF SUBROUTINES

AND

SUBROUTINE CALL STRUCTURE
### LIST OF ROUTINES IN SACS

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ABARF</td>
<td>Processes errors and prints messages</td>
</tr>
<tr>
<td>2. ACRPT</td>
<td>Print acquisition report</td>
</tr>
<tr>
<td>3. AFIND</td>
<td>Determines segment acquisitions from the segment reference file</td>
</tr>
<tr>
<td>4. AINIL</td>
<td>Performs initialization functions</td>
</tr>
<tr>
<td>5. APRPT</td>
<td>Prints the daily processing load report</td>
</tr>
<tr>
<td>6. AREAD</td>
<td>Reads one record from the crop window and segment reference data files.</td>
</tr>
<tr>
<td>7. AWRTA</td>
<td>Writes one record on the acquisition file.</td>
</tr>
<tr>
<td>8. BINBCD</td>
<td>Converts an integer to display code and places it in a hollerith string for printing</td>
</tr>
<tr>
<td>9. FZULU</td>
<td>Converts a zulu date to day, month and year</td>
</tr>
<tr>
<td>10. MAIN</td>
<td>Main control</td>
</tr>
<tr>
<td>11. SORTAG</td>
<td>Internal array sort routine</td>
</tr>
</tbody>
</table>
SACS (MAIN)
   AINIL
   ABARF
   AWRTA
   AREAD
   ABARF
   AFINP
   ABARF
   AWRTA
   ACRPT
   BINBCD
   APRPT
   FZULU
PART IV

SUBROUTINE DESCRIPTIONS AND FLOWCHARTS
SUPPLIED UTILITY ROUTINES

Routine Day

Call Day (IYMD, IDAY)
Given IYMD (3) where

IYMD (1) IS Day No.
IYMD (2) IS Month No.
IYMD (3) IS Year No.

Compute year day no. in IDAY

Routine PIMOD

Call PIMOD (A)

Convert ±A in radians to an angle 0-2π

Routine SOL (Entry ALPHA)

Call ALPHA (IFLAG):
For ephemeris usage as called by hector
computes ALPHAM and ALPHAT and IFLAG = 1

Routine PAGER (Entry Eject)

Call PAGER (NLINES)

Updates line count in NLINE with NLINES
NPAGE = 0 causes page to be restored prior to print.
NPAGE - page no.
HEADER - 80 char. 20A5
ICASE - case no.
KO - 6 print unit
INMAX is max no. of lines allowed
Initially NLINE should be set > LINMAX and NPAGE = 0
SUPPLIED UTILITY ROUTINES
(CONTINUED)

Call EJECT (NLINES)
Causes page to be restored automatically and then prints headers.

Routine CLDAY:

Call CLDAY

Call CLDAY
Given IDAY-DAY no. of the year compute in LMO-the month
and in LDA the day no.
Need: 1YEAR = 0 - Leap Year, ≠ 0 not Leap Year

Routine KEPLER:

Call KEPLER (XM, XECC, XE, ERROR)

Given XM - Mean anomaly, XECC - eccentricity
Compute: E-eccentric anomaly, error = 0 means OK

Routine LFPA

Call LFPA

[FLDA, LMO, LYR, ALFGM (can be dummy), DAYS]
Given: FLDA - day of month no., LMO - month no.,
LYR - year no. compute ALFGM - right ascension and
DAYS - Zulu day no.

Routine DEGMOD

Call DEGMOND (RAD, IDEG)

Given: angle rad in radians store the angle in deg., min., sec.,
in IDEG(1) - (3).

Routine FZULU

Call FZULU (IOATE, IOUT)

Given Zulu date in IDATE, compute year, month and day in
IOUT(1) - IOUT(3).

Routine RDM1A

Call RDM1A(FL, U)

Given double precision random no. seed in FL, compute random
no. U (0-1) based on uniform distribution.
SUBROUTINE A-BARF

Purpose:
This routine prints the error messages for the SACS program and returns control to the operating system.

Input:
The inputs to this routine are provided via the IBLK, CBLK, SBLK and ABLK common blocks. Table 6 shows the variables.

Output:
The outputs from this routine consists of a line of outputs printed on the output file (IOFILE). Table 5-I shows the error messages in the problem description.

Calling Sequence:
Call A-BARF (id)

id, error number

Subroutines Used:
None.

Processing:
The routine shall use a "COMPUTED GO TO" to access a write statement that prints the proper error message.
Table 6. Routine ABARF Inputs

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFNAM2</td>
<td>x</td>
<td></td>
<td>Cropwind file name</td>
</tr>
<tr>
<td>SFNAM2</td>
<td>x</td>
<td></td>
<td>SEG REFER file name</td>
</tr>
<tr>
<td>ICCASE</td>
<td>x</td>
<td></td>
<td>Cropwind case number from file</td>
</tr>
<tr>
<td>IWCASE</td>
<td>x</td>
<td></td>
<td>Input cropwind case number</td>
</tr>
<tr>
<td>ISCASE</td>
<td>x</td>
<td></td>
<td>Input SEG REFER case number</td>
</tr>
<tr>
<td>ISCASE</td>
<td>x</td>
<td></td>
<td>SEG REFER case number from file</td>
</tr>
<tr>
<td>ICOUN</td>
<td>x</td>
<td></td>
<td>Input card country ID</td>
</tr>
<tr>
<td>CCOUN</td>
<td>x</td>
<td></td>
<td>Cropwind country ID</td>
</tr>
<tr>
<td>SCOUN</td>
<td>x</td>
<td></td>
<td>SEG REFER country ID</td>
</tr>
<tr>
<td>SCRN</td>
<td>x</td>
<td></td>
<td>Current SEG REFER record number</td>
</tr>
<tr>
<td>ISNDAY</td>
<td>x</td>
<td></td>
<td>Maximum number of days in study</td>
</tr>
<tr>
<td>IPERR</td>
<td>x</td>
<td></td>
<td>Acquisition day when it exceeds maximum</td>
</tr>
<tr>
<td>PCRDID</td>
<td>x</td>
<td></td>
<td>Columns 75-78 of current input card</td>
</tr>
<tr>
<td>PCRDNO</td>
<td>x</td>
<td></td>
<td>Columns 79-80 of current input card</td>
</tr>
<tr>
<td>ISANG</td>
<td>x</td>
<td></td>
<td>Sun angle</td>
</tr>
<tr>
<td>IPCC</td>
<td></td>
<td></td>
<td>% cloud cover</td>
</tr>
<tr>
<td>ISNDAY</td>
<td>x</td>
<td></td>
<td>Number of days in study</td>
</tr>
</tbody>
</table>
Subroutine ACRPT

Purpose:

Prints the acquisition report - one record per entry. See the problem description for the report format.
SUBROUTINE AFIND

Purpose:
This subroutine selects acquisitions from a SEG REFER Data Record based upon the following:

a. The acquisition date is within a specified period
b. The cloud cover is less than a specified maximum
c. The sun angle is greater than a specified minimum

Input:
The inputs to this routine are provided via the IBLK; SBLK and CBLK. Table 4 identifies the variables.

Output:
The outputs from this routine are placed into the ABLK. Table 4 identifies the variables.

Calling Sequence:
Call AFIND

Subroutines Called:
ABARF - Error routine.

Processing:
A functional flow diagram is attached.
Error Exits

07-SEG REFER acquisition day too large xxx
maximum = xxx, record = xxx
Table 4. Routine AFIND Inputs and Outputs

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPTYPE</td>
<td></td>
<td>x</td>
<td>Type wheat (1=spring, 8=winter)</td>
</tr>
<tr>
<td>IPERR</td>
<td>x</td>
<td></td>
<td>Acquisition day for out of range date</td>
</tr>
<tr>
<td>IPNCW</td>
<td></td>
<td>x</td>
<td>No. crop windows</td>
</tr>
<tr>
<td>IPCC</td>
<td></td>
<td>x</td>
<td>Maximum percent cloud cover *10 to be used in acquisition selection</td>
</tr>
<tr>
<td>ISANG</td>
<td></td>
<td>x</td>
<td>Minimum degrees of sun angle *100 to be used in acquisition selection</td>
</tr>
<tr>
<td>IACNT</td>
<td>x</td>
<td></td>
<td>Total number of accesses for each window - IACNT (1) → IACNT (4)</td>
</tr>
<tr>
<td>IATOT</td>
<td>x</td>
<td></td>
<td>Total number of accesses</td>
</tr>
<tr>
<td>IAPNT</td>
<td>x</td>
<td></td>
<td>Pointer to SEGREFER acquisition entry</td>
</tr>
<tr>
<td>ANACQ</td>
<td>x</td>
<td></td>
<td>Number of acquisitions per day</td>
</tr>
<tr>
<td>ICLOW</td>
<td></td>
<td>x</td>
<td>Start date of window</td>
</tr>
<tr>
<td>ICHI</td>
<td></td>
<td>x</td>
<td>End date of window</td>
</tr>
<tr>
<td>ISDATE</td>
<td></td>
<td>x</td>
<td>SEG REFER FILE reference date</td>
</tr>
<tr>
<td>ISNDAY</td>
<td></td>
<td>x</td>
<td>Number of days in study</td>
</tr>
<tr>
<td>ISTRN</td>
<td></td>
<td>x</td>
<td>Training segment indicator (1 = training)</td>
</tr>
<tr>
<td>ISRDAY</td>
<td></td>
<td>x</td>
<td>Acquisition day (1 to 150)</td>
</tr>
<tr>
<td>ISSANG</td>
<td></td>
<td>x</td>
<td>Sun angle (1 to 150)</td>
</tr>
<tr>
<td>ISPCC</td>
<td></td>
<td>x</td>
<td>Percent cloud cover (1 to 150)</td>
</tr>
</tbody>
</table>
SUBROUTINE AFIND BLOCK DIAGRAM

ENTER AFIND

SET WINDOW INDEX TO SPRING OR WINTER
\( \alpha = \text{IPTYPE} \)

FOR EACH WINDOW

FOR EACH ACQUISITION

\% CLOUD COVER > MAX
(ISPCC > IPCC)

YES

NO

SUN ANGLE ≥ MINIMUM
(ISANG ≥ ISSANG)

YES

NO

ACQUISITION DATE = REF.
DATE + REL. DATE

ACQUISITION DATE WITHIN WINDOW

YES

NO

PUT ACQUISITION INDEX IN ACQUIRE RECORD (IAPNT)

ACQ. DATE < 426

YES

1A

NO

ABARF
ERROR 07
DATE OUT OF RANGE

1A

ADD 1 TO PRODUCTION DAY VECTOR ANACQ (ISRDAY)

TRAINING SEGMENT

YES

NO

25 ACQUISITIONS FOUND?

YES

NO

TO NEXT ACQUISITION

TO NEXT WINDOW

END ALL WINDOWS

RETURN

ORIGINAL PAGE IS OF POOR QUALITY
SUBROUTINE AINIL

Purpose:
This subroutine performs all initialization functions for the SACS program. The functions performed are listed as follows:

a. Reads the manually generated input cards.
b. Reads and verifies the headers for SEG REFER and CROP WIND files.
c. Writes the header for the ACQUISI FILE
d. Initializes common variables.

Input:
The inputs to the routine consist of common blocks, and 3 input files. Table 1 identifies the common variables used. The files are outlined as follows.

a. SEG REFER - logical unit ISFILE, format shown in Appendix A.
b. CROPWIND - logical unit ICFILE, format shown in Appendix A.
c. CARD INPUT - logical unit IIFILE, format shown in Section 2.1.1.

Output:
The outputs from this routine consist of common variables and an output tape label. Table 1 identifies the variables. The ACQUISI file header is shown in Appendix C.

Calling Sequence:
Call AINIL

Subroutines Called:
ABARF and AWRTA

Processing:
A functional flow diagram is shown in figure 3.2.2-1.

Error Exits:

- CROPWIND file has invalid label name
- SEGREFER file has invalid label name
- CROPWIND input case xxx not equal label case xxx
- SEGREFER input case xxx not equal label case xxx
- No. of days in study exceeds 426
- Input cards invalid or out of sequence
- Input sun angle less than 0 or greater than 90.00
- Percent cloud cover less than 0 or greater than 100.0
### Table 1. Routine AINIL Inputs and Outputs

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICID</td>
<td>x</td>
<td></td>
<td>Card ID columns 75-80</td>
</tr>
<tr>
<td>IPEOF</td>
<td>x</td>
<td></td>
<td>End of file flag (false)</td>
</tr>
<tr>
<td>IPTYPE</td>
<td>x</td>
<td></td>
<td>Type of wheat (unidentified, 0)</td>
</tr>
<tr>
<td>ICREG</td>
<td>x</td>
<td></td>
<td>Cropwind region ID (0)</td>
</tr>
<tr>
<td>ICZONE</td>
<td>x</td>
<td></td>
<td>Cropwind zone ID (0)</td>
</tr>
<tr>
<td>ICSTRA</td>
<td>x</td>
<td></td>
<td>Cropwind strata ID (0)</td>
</tr>
<tr>
<td>ICSUBS</td>
<td>x</td>
<td></td>
<td>Cropwind substrata ID (0)</td>
</tr>
<tr>
<td>GCOUN</td>
<td>x</td>
<td></td>
<td>Cropwind Country (XICOIN)</td>
</tr>
<tr>
<td>XITTL</td>
<td>x</td>
<td></td>
<td>Report case oriented page title</td>
</tr>
<tr>
<td>IACASE</td>
<td>x</td>
<td></td>
<td>Case number (IIFILE) for ACQUISE file</td>
</tr>
<tr>
<td>ISCASE</td>
<td>x</td>
<td></td>
<td>Case number (IIFILE) for Seg. Refer File</td>
</tr>
<tr>
<td>IWCASE</td>
<td>x</td>
<td></td>
<td>Case number (IIFILE) for crop wind file</td>
</tr>
<tr>
<td>IPCC</td>
<td>x</td>
<td></td>
<td>Percent cloud cover</td>
</tr>
<tr>
<td>ISANG</td>
<td>x</td>
<td></td>
<td>Minimum degrees sun angle (degrees)</td>
</tr>
<tr>
<td>XISANG</td>
<td>x</td>
<td></td>
<td>Minimum degrees sun angle (radians)</td>
</tr>
<tr>
<td>XICOUN</td>
<td>x</td>
<td></td>
<td>Country ID</td>
</tr>
<tr>
<td>IREG</td>
<td>x</td>
<td></td>
<td>Region ID</td>
</tr>
<tr>
<td>IZONE</td>
<td>x</td>
<td></td>
<td>Zone ID</td>
</tr>
<tr>
<td>IPRPT</td>
<td>x</td>
<td></td>
<td>Processing load report flag</td>
</tr>
<tr>
<td>ICRPT</td>
<td>x</td>
<td></td>
<td>Crop window report flag</td>
</tr>
<tr>
<td>IAPAG</td>
<td>x</td>
<td></td>
<td>Crop window report page number (0)</td>
</tr>
<tr>
<td>IANLIN</td>
<td>x</td>
<td></td>
<td>No. lines remaining on page (0)</td>
</tr>
<tr>
<td>ANACQ</td>
<td>x</td>
<td></td>
<td>No. acquisitions/day (0)</td>
</tr>
</tbody>
</table>
Table 1. Routine AINIL Inputs and Outputs
(Continued)

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCRN</td>
<td>x</td>
<td>x</td>
<td>CROPWIND record number (1)</td>
</tr>
<tr>
<td>CFNAM1</td>
<td></td>
<td></td>
<td>Name of CROPWIND file</td>
</tr>
<tr>
<td>CFNAM2</td>
<td>x</td>
<td>x</td>
<td>Cropwind file name from label</td>
</tr>
<tr>
<td>ICCASE</td>
<td>x</td>
<td>x</td>
<td>Case number from CROPWIND label</td>
</tr>
<tr>
<td>ICWIND</td>
<td></td>
<td></td>
<td>No. of crop windows</td>
</tr>
<tr>
<td>SCRN</td>
<td>x</td>
<td></td>
<td>Seg. Refer record number (1)</td>
</tr>
<tr>
<td>SFNAM1</td>
<td></td>
<td>x</td>
<td>Seg. Refer file name</td>
</tr>
<tr>
<td>SFNAM2</td>
<td>x</td>
<td>x</td>
<td>Seg. Refer file name from label</td>
</tr>
<tr>
<td>ISCASE</td>
<td>x</td>
<td>x</td>
<td>Case number from header</td>
</tr>
<tr>
<td>ISDATE</td>
<td></td>
<td></td>
<td>Reference date</td>
</tr>
<tr>
<td>ISNDAY</td>
<td>x</td>
<td>x</td>
<td>No. of days in study (1-426)</td>
</tr>
</tbody>
</table>
SUBROUTINE APRPT

Purpose:

This subroutine prints the daily processing load report.

Input:

The inputs to this routine are provided via the IBLK and ABLK, Common Blocks. The variables are shown in Table 3.

Output:

The outputs from this routine consist of the printed Daily Processing Load report.

Calling Sequence:

Call APRPT

Subroutines Called:

FZULU convert Zulu date to MMDDYY

Processing:

A functional flow diagram is attached.
Table 3. Routine APRPT Inputs and OUTPUTS

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XITTL</td>
<td>x</td>
<td></td>
<td>Report case title</td>
</tr>
<tr>
<td>IOFFLE</td>
<td>x</td>
<td></td>
<td>FORTRAN unit number for report</td>
</tr>
<tr>
<td>INLINE</td>
<td>x</td>
<td></td>
<td>No. lines per page</td>
</tr>
<tr>
<td>ANACQ</td>
<td>x</td>
<td></td>
<td>No. acquisitions per day</td>
</tr>
<tr>
<td>ISDATE</td>
<td>x</td>
<td></td>
<td>Reference date Zulu</td>
</tr>
<tr>
<td>XICOUN</td>
<td>x</td>
<td></td>
<td>Country ID</td>
</tr>
<tr>
<td>IREG</td>
<td>x</td>
<td></td>
<td>Region ID</td>
</tr>
<tr>
<td>IZONE</td>
<td>x</td>
<td></td>
<td>Zone ID</td>
</tr>
<tr>
<td>IPTYPE</td>
<td>x</td>
<td></td>
<td>Type of wheat 0, unidentified 1=spring; 5=winter</td>
</tr>
<tr>
<td>ISNDAY</td>
<td>x</td>
<td></td>
<td>No. of days in study</td>
</tr>
</tbody>
</table>
SUBROUTINE APRPT BLOCK DIAGRAM

ENTER APRPT

NO. LINES = \( \frac{\text{NO. DAYS} + 3}{4} \)

PAGE NO. = 0
\( J_1 = 1 \)
\( J_2 = 25 \)
\( J_3 = 51 \)
\( J_4 = 76 \)

1A

PAGE NO. = PAGE NO. + 1
NO. LINES / PAGE = 25

PRINT PAGE HEADERS

1B

FOR EACH LINE

PRINT ELEMENT
\( J_1, J_2, J_3, J_4 \)

INCREMENT J
\( J_1, J_2, J_3, J_4 \)

NO. LINES FINISHED

RETURN

YES

NO

\( 25 \) LINES PRINTED THIS PAGE

YES

1A

NO
SUBROUTINE AREAD

Purpose:
This subroutine reads both the CROP WINDOW File and the Segment Reference Data File and returns a record of each or sets a flag indicating that an end of data has been encountered.

Input:
Inputs to this routine are provided by the IBLK, PBLK, CBLK and SBLK. Table 2 shows the variables.

Output:
Outputs from this routine are placed into the PBLK, CBLK and SBLK. Table 2 shows the variables.

Calling Sequence:
Call AREAD

Subroutines Called:
ABARF - Error routine.

Processing:
A functional flow diagram is attached.
Table 2. Routine AREAD Inputs and Outputs

<table>
<thead>
<tr>
<th>Variance Name</th>
<th>Set</th>
<th>Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPEOF</td>
<td>x</td>
<td></td>
<td>End of file flag (T=EOF)</td>
</tr>
<tr>
<td>IPTYPE</td>
<td>x</td>
<td>x</td>
<td>Type wheat (0, unidentified; 1, spring; 5, winter)</td>
</tr>
<tr>
<td>XICOUN</td>
<td></td>
<td>x</td>
<td>Input country ID</td>
</tr>
<tr>
<td>IREG</td>
<td></td>
<td></td>
<td>Input region ID</td>
</tr>
<tr>
<td>IZONE</td>
<td></td>
<td></td>
<td>Input zone ID</td>
</tr>
<tr>
<td>ISFILE</td>
<td></td>
<td></td>
<td>Fortran unit number segment reference data file</td>
</tr>
<tr>
<td>ICFILE</td>
<td></td>
<td></td>
<td>Fortran unit number crop window file</td>
</tr>
<tr>
<td>CCRN</td>
<td></td>
<td></td>
<td>Crop window file current record number</td>
</tr>
<tr>
<td>CCOUN</td>
<td></td>
<td></td>
<td>Crop window Country ID</td>
</tr>
<tr>
<td>ICREG</td>
<td></td>
<td></td>
<td>Crop window Region ID</td>
</tr>
<tr>
<td>ICZONE</td>
<td></td>
<td></td>
<td>Crop window Zone ID</td>
</tr>
<tr>
<td>ICSTRA</td>
<td></td>
<td></td>
<td>Crop window Strata ID</td>
</tr>
<tr>
<td>ICSUBS</td>
<td></td>
<td></td>
<td>Crop window Substrata ID</td>
</tr>
<tr>
<td>ICLOW</td>
<td></td>
<td>x</td>
<td>Crop window, window start dates</td>
</tr>
<tr>
<td>ICHI</td>
<td></td>
<td></td>
<td>Crop window, window ending dates</td>
</tr>
<tr>
<td>SCRN</td>
<td>x</td>
<td></td>
<td>SEG REFER current record number</td>
</tr>
<tr>
<td>SCOUN</td>
<td></td>
<td>x</td>
<td>SEG REFER country ID</td>
</tr>
<tr>
<td>ISREG</td>
<td>x</td>
<td></td>
<td>SEG REFER region ID</td>
</tr>
<tr>
<td>ISZONE</td>
<td>x</td>
<td></td>
<td>SEG REFER zone ID</td>
</tr>
<tr>
<td>ISSTRA</td>
<td>x</td>
<td></td>
<td>SEG REFER strata ID</td>
</tr>
<tr>
<td>ISUBS</td>
<td>x</td>
<td></td>
<td>SEG REFER substrata ID</td>
</tr>
<tr>
<td>ISSEG</td>
<td>x</td>
<td></td>
<td>SEG REFER Segment ID</td>
</tr>
<tr>
<td>ISTRN</td>
<td>x</td>
<td></td>
<td>SEG REFER Training Segment Indicator</td>
</tr>
<tr>
<td>ISRDAY</td>
<td>x</td>
<td></td>
<td>SEG REFER relative acquisition day</td>
</tr>
<tr>
<td>ISSANG</td>
<td>x</td>
<td></td>
<td>SEG REFER sun angle</td>
</tr>
<tr>
<td>ISPCC</td>
<td>x</td>
<td>x</td>
<td>SEG REFER % cloud cover</td>
</tr>
<tr>
<td>ISPCSW</td>
<td>x</td>
<td>x</td>
<td>Per cent spring wheat *10</td>
</tr>
<tr>
<td>ISPCWW</td>
<td>x</td>
<td>x</td>
<td>Percent winter wheat *10</td>
</tr>
<tr>
<td>SFTRL</td>
<td></td>
<td>x</td>
<td>Trailer ID &quot;ZZZZ&quot;</td>
</tr>
</tbody>
</table>
ERROR EXITS

05 Invalid Country Input xxxx CROPWIND xxxx SEGREFEL xxxx

Country should be equal from all three sources
SUBROUTINE AREAD BLOCK DIAGRAM (PAGE 2 OF 2)

2A

TRAILER (SCOUN = ZZZZ)

NO

1B

IS ISPRW = 0

NO

SET TYPE WHEAT TO SPRING
  IPTYPE = 1

YES

SET TYPE WHEAT TO WINTER
  IPTYPE = 5

1D

1D
SUBROUTINE AWRTA

Purpose:
This subroutine writes the records of the Data Acquisition File. These records include the Header, data records and trailer.

Input:
The inputs to this routine come from the IBLK and the ABLK. The variables are shown in Table 5.

Output:
The outputs from this routine consist of records written to the "ACQUISI" file and variables updated in the ABLK. The variables are shown in Table 5 and the record formats are shown in Table 5.

Calling Sequence:
Call AWRTA.

Subroutines Called:
None.

Processing:
A functional flow diagram is attached.
Table 5. Routine AWRTA Inputs and Outputs

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Set</th>
<th>Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPFTF</td>
<td></td>
<td>x</td>
<td>First time flag 0=first time; 1= subsequent times; 1=last time</td>
</tr>
<tr>
<td>IPNEW</td>
<td></td>
<td>x</td>
<td>Number of crop windows</td>
</tr>
<tr>
<td>XICOUN</td>
<td></td>
<td>x</td>
<td>Input country</td>
</tr>
<tr>
<td>IACASE</td>
<td></td>
<td>x</td>
<td>Case number of acquisition file</td>
</tr>
<tr>
<td>IAFILE</td>
<td></td>
<td>x</td>
<td>Fortran unit number of acquisition file</td>
</tr>
<tr>
<td>AFNAM</td>
<td></td>
<td>x</td>
<td>File name</td>
</tr>
<tr>
<td>ABLANK</td>
<td></td>
<td>x</td>
<td>Record padding</td>
</tr>
<tr>
<td>IACNT</td>
<td></td>
<td>x</td>
<td>Total number accesses per window</td>
</tr>
<tr>
<td>IAPNT</td>
<td></td>
<td>x</td>
<td>Pointer to SEG REFER acquisitions selected</td>
</tr>
<tr>
<td>ACRN</td>
<td></td>
<td>x</td>
<td>Current record number</td>
</tr>
<tr>
<td>ICCASE</td>
<td></td>
<td>x</td>
<td>Case number of cropwind file</td>
</tr>
<tr>
<td>ISNDAY</td>
<td></td>
<td>x</td>
<td>Number days in study</td>
</tr>
<tr>
<td>ISCASE</td>
<td></td>
<td>x</td>
<td>Case number from REFER file</td>
</tr>
<tr>
<td>KX</td>
<td></td>
<td>x</td>
<td>Contains list of acquisition dates for writing out on acquisition file</td>
</tr>
</tbody>
</table>
SUBROUTINE AWRTA BLOCK DIAGRAM

ENTER
AWRTA

WRITE
HEADER
RECORD

0
YES

FIRST
TIME FLAG
VALUE

= 1

WRITE
DATA
RECORD

-1
LAST
TIME

WRITE
TRAILER

INCREMENT
RECORD WITHIN
COUNT
ACRN = ACRN+1

RETURN
SUBROUTINE BINBCD BLOCK DIAGRAM

PURPOSE: TO CONVERT AN INTEGER TO DISPLAY CODE CALLING SEQUENCE:

CALL BINBCD (IVAL, NDIG, IOUT)
IVAL — BINARY INTEGER
NDIG — NO. OF DIGITS TO OUTPUT
IOUT — ORIGIN OF THE OUTPUT CHARACTER ARRAY

<table>
<thead>
<tr>
<th>VARIABLES</th>
<th>0</th>
<th></th>
<th>1</th>
<th></th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBCD ARAY</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>ITEM</td>
<td>1</td>
<td>10</td>
<td>100</td>
<td></td>
<td>10000</td>
</tr>
<tr>
<td>I = SUBSCRIPT INTO OUTPUT ARRAY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J = CURRENT VALUE OF INPUT INTEGER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K = NO. DIGITS REMAINING TO PROCESS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N = CURRENT DIGIT VALUE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ENTER

J = IVALUE
I = 1
K = NDIGIT-1
IBCD(1) = BLANK

OVALUE(I) = "-" YES

J < 0 TEST MINUS

NO

J = J
I = I + 1
K = K - 1

OVALUE(I) = "-" NO

J = 0 TEST ZERO

OVALUE(NDIGIT) = 0 YES

NO

J > 10^K

YES

PUT * IN LAST DIGIT OUTPUT OVALUE(NDIGIT)

NO

FOR VALUE OF K = K, K-1, ... 0

N = J - 10^K

OVALUE(K) = IBCD(N)
N = N + 10^K

J = J - N
K = K - 1

YES

N = 0

IBCD(1) = "0"

NO
SUBROUTINE MAIN

Purpose:
This subroutine is the main control routine for the SACS program. It is entered by the operating system each time the SACS program is executed.

Input:
The inputs to this routine are provided via common block supplied by the routines it calls. Since this routine is a control routine the inputs consist mainly of flags and counters.

Output:
The outputs from this routine consist of the setting of flags.

Calling Sequence:
Main Fortran routine.

Subroutines Collect
This routine calls the following routines.

a. AIN1 - initialization routine to read input cards and process file headers
b. AREAD - Read data records from both crop windows and segment reference data file
c. APRPT - Prints the daily processing load report.
d. AFIND - Selects segment reference file acquisitions based upon crop windows and specified input parameters.
e. AWRTA - writes the data and trailer records of the data acquisition file.
f. ACRPT - writes the Segment Data Acquisition report (crop window)

Processing:
A functional flow diagram is attached.
SUBROUTINE MAIN BLOCK DIAGRAM

ENTER SACS

AINIL
READ INPUT PROCESS HEADERS

AFIND
FIND SELECTED ACQUISITIONS IN SEG REFER FILE

SET FIRST TIME FLAG TO END (-1) (IPFTF)

1A
WRITE RECORD TO DATA ACQUISITION FILE

AWRTA
WRAP UP DATA ACQUISITION FILE

ACRPT
PRINT CROP WINDOW REPORT

APRPT
PRINT DAILY PRODUCTION REPORT

1B
END OF FILE IPEOF=T

YES

1A

1B

1C
EXIT PROGRAM
Subroutine SORTAG

Purpose:

Sorts Array A into increasing order from A(II) to A(JJ). The array tag is permuted the same as Array A. This routine processes arrays that can fit in core only. (A CDC library routine.)

Calling Sequence:

CALL SORTAG(A, II, JJ, TAG)
PART V

SUBROUTINE LISTINGS
SUBROUTINE ABADF(IREP)

C**SUBROUTINE ABADF(IREP)**

C
CROUTINE ABADF
C
CPURPOSE
THIS ROUTINE PRINTS ERROR MESSAGES AND ABORTS THE
PROGRAM IF THE ERROR IS FATAL
C
CCOMMON
INL-MANUAL INPUT VARIABLES
PULK-PROGRAM FLAGS,COUNTERS,CONSTANTS
SULK-SEGMENTED FILE RECORD
AIK-ACQUISITION FILE RECORD
CULK-CROPPING FILE RECORD
C
CLINKAGE
CALL ABADF(IREP)
IF(IHE) ERROR ID
C
CRoutines CALLED
NONE
C
CLOCAL VARIABLES
C
C**COMM ON BLOCK INL-MANUAL INPUT PARAMETERS CONSTANTS**
C
COMMON /IPLK/
IX1ITL(12),IXIS,IXCOUN,IXITL(16),
PICRD(3),JACAS,ISCAS,INCAS,PICCC,IXANG,IREG,IZONE,ITFILE,ITOFILE,
44PFILE,44PFIL

LOGICAL INPLK,ITPLK
C
CNAMES DESCRIPTION
C
XIITL REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)
IXISANG MINIMUM SUN ANGLE IN RADIANS FROM INPUT
IXCOUN COUNTRY ID FROM INPUT (16)
IXITL CROPP WINDOW TITLES, TITLE FOR EACH WINDOW OCCUPIES 0 WORDS
JIM
ITFILE AT 2 CHARACTERS PER WORD
JCUMN COLUMNS 75-80 OF INPUT CARD 5A2
JACAS ACQUISITION FILE CASE NO. FROM INPUT
JICAS MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (12)
JACAS5 MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (14)
JCROF REPORT ID FROM INPUT (16)
JICROF LOGICAL UNIT NO. FOR CARD INPUT
JICTFILE LOGICAL UNIT NO. FOR REPORT OUTPUT
JICTFILE LOGICAL UNIT NO. FOR CROPPING DATA FILE
JITFILE LOGICAL UNIT NO. FOR ACQUISITION FILE
JITLINES LIMITS PER PAGE
JITPRINT PRINTING LAYOUT REPORT FLAG (T=PRINT REPORT)
JITPRINT ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)
COMMON BLOCK PULK-PROGRAM COUNTERS AND CONSTANTS

C
C NAME DESCRIPTION
C SCRNN SEGREFER CURRENT RECORD NO.
C SFNAME SEGREFER FILE NAME PROTOTYPE*SEGREFER*
C SFNAME SEGREFER FILE NAME FROM LABEL
C SPTL IDPLMN 10 #ZZZ#
C SOUN SEGREFER COUNTRY TO (FO2)
C SPECSP PERCENT SPRING WHEAT (FO2)
C SPECWM PERCENT WINTER WHEAT (FO2)
C Sraw SUN ACCEL/IN RADIAN (FOR EACH OF 150 SEGMENTS)
C I5SCAS SEGREFER CASE NO. FROM LABEL
C I5DATE SEGREFER REFERENCE DATE-JULIA
C I5RDAY NO. DAYS IN STUDY (+*26)
C I5RAC NO. ACQUISITIONS IN RECORD (0-150)
C I5RFG SEGREFER REGION ID
C I5RFL SEGREFER COUNTRY ID
C I5RST SEGREFER STRATA ID
C I5RSTR SEGREFER STRATA TO
C I5RSL SEGREFER SEGMENT ID
C I5RSLN SEGREFER SEGMENT INDICATOR (NORMAL=1; TRAINING=0)
C I5RDAY ACQUISITION DAY FOR 150 ACQUISITIONS-JULIA
C I5IZEND 150 END OF ALL ACQUISITIONS IN RECORD
C I5PCC PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS
C I5VLHC NO OF VEHICLES IN THE SKMATE TAPE
C I5LEH ORIGNED ORIGIN OF ACCESS DATES AS READ FROM SEQ. REF. FILE
C
C COMMUN /PRINT/ INFER
C INFER INPUT ERROR FLAG. SET = 1 IN AWF OF THERE ARE ANY FATAL
C ERRORS DETECTED ON THE INPUT CARDS BY AWFIL
C
C IF THERE WERE ANY INPUT ERRORS, ABORT RUN.
C IF ( IERR .LT. 0 ) GO TO 210
C
C GO TO WRITE STATEMENT FOR ERROR
C IF(IERR.GT.15) GOTO 210
C
C WRITE (IOFILE,#1) CFNAME
C G*HT 200
C
C FORMAT (6H001) + ThiCHWIND FILE HAS INVALID LABEL NAME +2AQ)
C AWFLO
C
C WRITE (IOFILE#21) SFNAME
C G*HT 200
C
C FORMAT (6H002) +37FHSEGREFER FILE HAS INVALID LABEL NAME +2AQ)
C AWFLO
C
C WRITE (IOFILE#31) TWCASE,1CCASF
C GO TO 190
C
C FORMAT (6H003) +20HCHWIND INPUT CASE.ID3.25H NOT EQUAL TO LABEL)
C AWFLO
C
C CASE .15)
C
C WRITE (IOFILE#41) I5SCAS,15SCAS
C GO TO 190
C
C FORMAT (6H004) +3HSEGREFER INPUT CASE.ID3.25H NOT EQUAL TO LABEL)
C AWFLO
C
C CASE .15)
C
C WRITE (IOFILE#51) XICOUNT,SCOUNT
C G*HT 200
C
C FORMAT (6H005) +2HINVALID COUNTRY INPUT+4A+10H CHWIND+4A+10H)
C AWFLO
C
C CASE .15)
C
C
C ORDER ORIGNAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE
C COMMUN ILCK-CHUPIND HEADER AND DATA
C
COMMUN /CHLX/
1CCNW,CFH(N),CFH(N),CCOUN,
2CSCHL,ICCHF,ICCHF,ICCHF,ICCHF,ICCHF,ICCHF,ICCHF,ICCHF,ICCHF,
3NAME,DESCRIPT
C NAME DESCRIPTION
C CFH(N) COMMILH RECORD NO.
C CFH(N) NAME, CCHUPIND (255)
C CFH(N) LABEL NAME =I.MIST MATCH CFH(N)
C CFH(N) CHUPIND CHUNK ID (40)
C ICCHF CHUPIND FILE INFO FROM LABEL
C ICCHF CHUPIND REGID ID
C ICCHF CHUPIND ZUSL ID
C ICCHF CHUPIND OTMATA ID
C ICCHF CHUPIND ZUSSTKATA ID
C ICCHF START DATE OF 6 WINDOWS ZULU-RELATIVE JAN 1950
C ICCHF STOP DATE OF 6 WINDOWS ZULU-RELATIVE JUN 1950
C
DIMENSION ILEG(11),JIMAG(61)
C EQUIVALENT (IPANK,JIMAG)
C DATA 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60
C DATA NONE / IH+1OH,1HN,HHE /
C TEST LAST LINE ENTERED IF YES PRINT STATISTICS
IF(IPFTF.LT.0) GOTO 50
C IF (IPFTF.LT.0) PRINT WANTED
IF(,IHT,1017)GOTO 60
C CUMULIF LINE COUNT FROM MAX NO. SEGMENTS SELECTED
NMAX=1
DO 5 I=1,4+1
J3 (IACIT(1)+1)/4
IF (NMAY,GE,0) GOTO 5
NMAX=J3
CONTINUE
S 5 TEST PAGE HEADERS REQUIRED (NO. LINES ON PAGE TOO SMALL)
IAM1N=IAM1N-NMAX-1
IAM1N=IAM1N+1
IF(IAM1N,GE,0) GOTO 10
C PRINT PAGE HEADERS, INCREMENT PAGE, RESET LINE COUNT
IAM1N=INLTNE-8
JAPAG=JAPAG+1
WRITE(IPFIL,200) ICASEXITIL,JAPAG
WRITE(IPFIL,250) ( I,J=1,4 )
WRITE(IPFIL,250)
WRITE(IPFIL,250)
C INITIALIZE FIRST LINE OF SEGMENT AND PRINT
C
10 ISSUE=0
C INITIALIZE ACQUISITION POINTERS TO BLANKS
DO 20 J=1,64,1
JIMAG[J]=100000
20 CONTINUE
C IF(ISSMAC.NE.0) GOTO 22
C IF (ISSMAC.NE.0) PRINT NULL LINE AND EXIT
WRITE(IPFIL,200) ISSSEG
GOTO 60
C TEST IF ANY WINDOW REQUIRES NONE OR ALL ILLEGAL
C 22 IF (FLG=2)
   DO 25 J=1,4
   J=ISUM(1)-1+1
   IF (IACNT(J) .NE. 0) GO TO 23
   CONTINUE
C 25 CONTINUE
   SET UP TO PRINT NONE
   DO 21 K=1,4
   JIMAG(J)=NONE(K)
   J=J+1
   CONTINUE
C 21 CONTINUE
   IF (IACNT(1) .NE. 0) GO TO 25
   SET UP TO PRINT ALL ILLLEGAL
   DO 24 K=1,11
   JIMAG(J)=ILLLEGAL(K)
   J=J+1
C 24 CONTINUE
   SET UP NEXT LINE FOR EACH OF 4 WINDOWS
   L=ISUM(2)*J-1)
   DO 35 J=1,4
      J=J+1
      IF ((ISUM=K) .GT. IACNT(1)) GO TO 40
      L=L+1
      CALL RNDUD (IAPNT(L+1),J,JIMAG(J))
   CONTINUE
C 35 CONTINUE
   L=ISUM+1
   DO 46 J=1,4
      IF (FLG=3) GO TO 45
      IF (FLG=2) GO TO 45
      IF (FLNPLT .EQ. 1) ISMFLG=JIM
      NPLT (LJL+270) TSKFLG=ISSEG.ISANAC.IACNT(1) . GT. JIMAG(L) . LT. 1) \)
      I=LAC(1) .GT. JIMAG(L) . LT. 1) \)
      I=LAC(1) .GT. JIMAG(L) . LT. 1) \)
      IF (FLNG=1) GO TO 46
   CONTINUE
C 45 WRITE SUBSEQUENT LINES
C 46 WRITE (TJFILF,270) (JIMAG(L),L=1,60)
   NMAX=NMAX+1
   IF (NMAX .LE. 0) GO TO 60
   CONTINUE
C 47 CONTINUE
   BLANK OUT ACQUISITION POINTERS FOR NEXT PRINT LINE.
   DO 57 L=1,60
      JIMAG(L)=JAC(1)
   CONTINUE
   GO TO 30
C 50 WRITE (TJFILF,500) ACPN
C 51 RETURN
   FORMAT (HEADCASE,14,77H SEGMENT ACQUISITION DATA +1Z49+16M LPP SIMF
   ILLATION +NHRAGE +14)
C 210 FORMAT (1HD0.1X.4ACQUISITION +A4.2X7REGION +15.2X5MZONE +15.2XHINSTACRP
   CRA +T4X2 +10HSSIP +0ATA +71)
C 215 FORMAT (/12H0V7X SEGMENT +7.XX (11XXXXXXXXX WTPAR(2,7) 2))))
C 240 FORMAT (X,X,0.1X.=TOTAL +7.XX(45.2X8PSEIETF +9))
C 250 FORMAT (X,0.1X,HRTX +A4.HAC(55),J,HAC(11,11,5.5X))
C 270 FORMAT (A1,1X,15,13X,8(12.3X,15A14))
260 FORMAT(1H, 20X, 0(13A1, 9X))
290 FORMAT(1HO, 19, 9H-000, 3X, 4HONE)
300 FORMAT(1HI, 30H TOTAL ACQUISITION RECORDS WRITTEN=, F7.0)
END
### SUBROUTINE AFINDD

#### PURPOSE
This routine selects acquisitions from a seafloor data record based upon:

- Cloud cover LT maximum
- Sun angle LT minimum
- Acquisition date within crop window

#### LOCAL VARIABLES
- \( i \): Current window number
- \( j \): Current acquisition number
- \( k \): Index into window date—either spring or winter
- \( m \): Number of acquisitions current window

#### ROUTINES CALLED
- AGANF—ERROR ROUTINE
- O7—REFERENCE DATE OUT OF RANGE

#### COMMON BLOCK
- TALK=MANUAL INPUT PARAMETERS
- CONSTANTS

#### NAME
- DESCRIPTION
- XITITL INPUT CASE ORIENTED PAGE TITLE FROM INPUT
- XSANG MINIMUM SUN ANGLE IN RADIANS FROM INPUT
- XITCUN COUNTRY ID FROM INPUT
- XINTL CROP WINDOW TITLE. TITLE FOR EACH WINDOW OCCUPIES 8 WORDS
- XICID COLUMNS 75-90 OF INPUT CARD 5A2
- XICASE SEAFLOOR CASE NO. FROM INPUT
- XICSF SEAFLOOR CASE NO. FROM INPUT
- XICNT CROP CASE NO. FROM INPUT
- XICTC MAXIMUM PERCENT CLINN COVER < 15% 10 TO 10 DEG SELECT SEGMENT
- XLONG MINIMUM DEGREES SUN ANGLE = 100 (IN DEGREES) (14)
- XICPF COUNTRY ID FROM INPUT (13)
- XIFILE LOGICAL UNIT NO. FOR CAMP INPUT
- XIFILE LOGICAL UNIT NO. FOR REPORT OUTPUT
- XIFILE LOGICAL UNIT NO. FOR SEAFLOOR DATA FILE
- XIFILE LOGICAL UNIT NO. FOR CROPWINDOW FILE
- XIFILE LOGICAL UNIT NO. FOR ACQUISITION FILE
- XILINES LINES PER PAGE
- XIPRPT PRINTING LUAL REPORT FLAG (1=PRINT REPORT)
- XIPRNT ACQUISITION CROP WINDOW REPORT FLAG (1=PRINT REPORT)
- XITITLE CROP WINDOW TITLE FOR REPORT
C CASE CROPHIND CASE NO. FROM LAHEL
C CROPHIND NO. OF CROPH WINDOWS (A)
C CROPHIND REGION ID
C CROPHIND ZONE ID
C CROPHIND STRATA ID
C CROPHIND SUBSTRA ID
C CROPHIND START DATE OF & WINDOWS ZULU-RELATIVE JAN 1950
C CROPHIND STOP DATE OF & WINDOWS ZULU-RELATIVE JAN 1950
C
C COMMUN BLOCK AND ACQUI XIT FILE AND REPORT VARIABLES
C
C NAME DESCRIPTION
C ACRTN CURRENT RECORD NO.
C AFAM LABLE FILE NAME ACQUIXIT *
C ARTRL TRATER ID */Z */ Q *
C ANACO NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS
C IACT TOTAL NO. OF ACQUISITIONS FOR EACH OF 4 WINDOWS
C IAOI TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS
C IAPNT INDEX TO SEGMENT ACQUISITION (1-150) THAT IDENTIFIES
C A SELECTED SEGMENT. IAPNT(I);= WINDOW 1 THROUGH 4
C IAPAG ACQUISITION REPORT CURRENT PAGE NUMBER
C IANLIN ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE
C IACINT NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS
C
C DIMENSION IAPNT(100)
C C OUTLINE (KAPNT(I),IAPNT(I))
C C FIND NO. ACQUISITIONS IN RECORD
C DO 5 I=1,150
C IF(ISRDAY(I),10,0) GOTO 6
C I=1
C 5 CONTINUE
C DO 6 J=1,100
C KAPNT(I)=0
C 6 CONTINUE
C IAOI=0
C 10 CONTINUE
C
C INITIALIZE CROPHIND WINDOW INDEX. K=0 WINTER, K=4 SPRING
C K=IPJYF+1
C FOR EACH WINDOW I, TEST ALL SEGMENT ACQUISITIONS
C DO 16 J=1,150
C K=K+1
C INITIALIZE WINDOW LOOPVARIABLES K AND NACQ
C NACQ=0
C NACQ=0
C FOR EACH OF 150 SEGMENTS=SEF IF ACQUISITION IS WITHIN
C WINDOW AND SATISFIERS CLOUD AND SUN CRITERIA
C DD 60 J=1,150
C IS ACCESS DATE WITHIN WINDOW I
C IF(ICH(K),LI,ISRDAY(J)) GOTO 60
C IF(ICH(K),LJ,ISRDAY(J)) GOTO 40
C TEST DAY WITHIN SIZE OF VECTOR
C I=1,ISDAY(J)-ISDAY1+1
C IF(ISDAY(J),LI,ISDAY1) GOTO 40
C ERROR 7 DAY OUT OF RANGE OF VECTOR
C IERR=JRDAY
C CALL ISDAY(7)
C IF(ISDAY1,J)GOTO 50
ACCESS DATE IS WITHIN WINDOW 1. ADVANCE NACQ1
(TOTAL ACCESS COUNT FOR WINDOW 1)
NACQ1 = NACQ1 + 1

ANY PREVIOUS LEGAL ACQUISITIONS FOR CURRENT SEGMENT
WITHIN WINDOW 1.
IF ( NACQ1 .LE. 0 ) GO TO 50
YES, IF THIS IS NOT A TRAINING SEGMENT, SKIP CLOUD
COVER AND SUN ANGLE TEST.
IF(ISTRN, .NE. 0) GO TO 60
NOW TEST FOR LEGAL ACQUISITION SATISFYING CLOUD COVER
L.T. MAXIMUM AND SUN ANGLE .GE. MINIMUM.
50 IF ( ISPCC(J) .NE. TPCC ) GO TO 40
IF ( SSANG(J) .LT. XISANG ) GO TO 60
SEGMENT SELECTED FOR WINDOW 1. UPDATE COUNT (NACQ) AND JIM
INDEX ARRAY.
NACQ1 = NACQ1 + 1
IAPHI(NACQ1) = TURKIN(J)
ANAC(IINDAY) = AANAL(IINDAY) + 1
C ACCEPT MAXIMUM OF 25 ACQUISITIONS FOR EACH WINDOW.
IF(NACQ1 .GE. 25) GO TO 70
END OF SEGMENT
C CONTINUE
60 CONTINUE
END OF WINDOW. POST TOTAL ACQUISITION COUNT
IACHT(1) = NACQ1
IACHT(1) = NACQ1
IANT = IANT + NACQ1
C CONTINUE
RETURN
END
**SUBROUTINE AINIL**

**Purpose:** This routine initializes the SACS program.

- Reads special label.
- Reads common block label.
- Reads input parameter cards.

**Linkage:** CALL AINIL

**Routines Called:** None

**Local Variables:** None

**Errors Detected:** 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11

---

**COMMON BLOCK IBLK-MANUAL INPUT PARAMETERS CONSTANTS**

**NAME** | DESCRIPTION
---|---
IE4TILI | PAGE ORIENTED PAGE TITLE FROM INPUT (1244)
IE1XIT4 | SAME AS IBLK
TSANG | HOURS SIN ANGLE IN RADIANS FROM INPUT
ITCOUN | COUNTRY ID FROM INPUT (44)
ICINTL | CROP WINDOW TITLES, TITLE FOR EACH WINDOW OCCUPIES 8 CHAR.
ISEC | AT 2 CHARACTERS PER ITEM.
ICICID | COLUMNS 8-HRU INPUT CARD SAG
ITCASC | ACQUIREL FILE CASE NO. FROM INPUT
ICASC | CASE acquirer CASE NO. FROM INPUT
ICASC | CROP WINDOW CASE NO. FROM INPUT
ITPCC | MAXIMUM PERCENT STIG TOP COVERED 1-10 TO BE USED IN SEGMENT SELECT
ITISP | MAXIMUM FIELDS SIN ANGLE = 100 (IN DEGREES) (14)
ITPLS | HISTORY ID FROM INPUT (15)
ITPGNZ | ZONE ID FROM INPUT (15)
ITFLF | LOGICAL UNIT NO. FOR CARD INPUT
ITFLG | LOGICAL UNIT NO. FOR REPORT OUTPUT
ITFLF | LOGICAL UNIT NO. FOR SEGMENT DATA FILE
ITFLF | LOGICAL UNIT NO. FOR CROP-WINDOW FILE
ITFLF | LOGICAL UNIT NO. FOR ACQUISITION FILL
ITPLS | LINE PER PAGE
ITFRT | PROCESSING LOG REPORT FLAG (TAGPRINT REPORT)
ITFRT | ACQUISITION (CROP WINDOW) REPORT FLAG (TAGPRINT REPORT)

---

**COMMON BLOCK IPHI-BLOCK PROGRAM COUNTS AND CONSTANTS**

**NAME** | DESCRIPTION
---|---
ITPLK | PROCESSING LOG REPORT FLAG (TAGPRINT REPORT)
ITFRT | ACQUISITION (CROP WINDOW) REPORT FLAG (TAGPRINT REPORT)
000179 1 IF (ICMC(1) .NE. ICCN) GO TO 14
000180 IF (ICMC(1) .EQ. 2) GO TO 15
000181 14 CALL ABANFK(2)
000182 CONTINUE
000183 C CHECK SEGREGER AND CROPWIND CASE NUMBERS
000184 IF (ISCASE(0,0) .GT. 16) GOTO 16
000185 IF (ISCASE(0,1) .GT. 16) GOTO 16
000186 CALL ABANFK(4)
000187 16 IF (ISCASE(0,0) .LE. 16) GOTO 18
000188 IF (ISCASE(0,0), ICCASL) GOTO 18
000189 CALL ABANFK(3)
000190 18 IF (ISANG(0,0) .GT. 22)
000191 CALL ABANFK(4)
000192 GO TO 23
000193 22 IF (ISANG(0,0) .GT. 20) GOTO 20
000194 CONTINUE
000195 IF (IPCE(0,0) .LT. 20) GOTO 20
000196 CALL ABANFK(10)
000197 24 IF (IPCE(0,0) .LT. 100) GOTO 24
000198 IF (IPCE(0,0) .LT. 100) GOTO 24
000199 GO TO 28
000200 26 IF (IPCE(0,0) .GT. 100) GOTO 26
000201 IF (IPCE(0,0) .GT. 100) GOTO 26
000202 C IF THERE WERE ANY INPUT ERRORS, ABORT RUN.
000203 C 28 IF (IERROR .NE. 0) CALL ABANFK(-1)
000204 C SUM ANGLE TO RADIANs
000205 XISANG=XISANG/5729.578
000206 XISANG=XISANG/5729.578
000207 C INITIALIZE FLAGS COUNTERS
000208 IF (IPCL=0)
000209 IF (IPCM=0)
000210 IF (IPCM=0)
000211 IF (IPCM=0)
000212 IF (IPCM=0)
000213 IF (IPCM=0)
000214 IF (IPCM=0)
000215 IF (IPCM=0)
000216 IF (IPCM=0)
000217 IF (IPCM=0)
000218 IF (IPCM=0)
000219 IF (IPCM=0)
000220 C CONTINUE
000221 30 CALL WHITE ACQUISITION LABEL
000222 CALL AMPTA
000223 KFACT
000224 100 FORMAT(12A0*26X*2A2*12)
000225 105 FORMAT(11,5HCASE +14,8X+120A+100+5IX+6HEADER+10 +30X12A4)
000226 106 FORMAT(11,5HCASE +14,8X+120A+100+5IX+6HEADER+10 +30X12A4)
000227 107 FORMAT(11,5HCASE +14,8X+120A+100+5IX+6HEADER+10 +30X12A4)
000228 108 FORMAT(11,5HCASE +14,8X+120A+100+5IX+6HEADER+10 +30X12A4)
000229 109 FORMAT(11,5HCASE +14,8X+120A+100+5IX+6HEADER+10 +30X12A4)
000230 END
**SUBROUTINE APNPT**

C ***---------------------------------------------APNPT***
C
C ROUTINE  APNPT
C
C PURPOSE  THIS ROUTINE PRINTS THE DAILY PROCESSING LOAD REPORT
C
C LINKAGE  CALL APNPT
C
C ROUTINES CALLED
C
C FLULU, CONVERT ZULU DATE TO MM-DD-YY
C
C LOCAL variables
C
C THRT-YY-MM-DD REFERENCE DATE
C
C NCPAG-NO. OF CURRENT PAGE
C
C NLIN=TOTAL NO. DETAIL LINES TO BE PRINTED
C
C NLINP-NO. OF DETAIL LINES REMAINING TO BE PRINTED
C
C NACUM-ACQUISITION NO. FOR COLUMNS 1 THROUGH 4
C
C***---------------------------------------------APNPT***
C
C COMMON BLOCK 1BLK=MANUAL INPUT PARAMETERS CONSTANTS
C
C
C COMMON 1BLK
C
C IVILK
C
C IXITTL(12),XISANG,XICOUN,INTIL(16)
C
C XI1ND(3),IXACAF,EXCASE,IECASE,IPCC,XISANG,INREG,IZONF,IIFILE,IOFILE
C
C IHPLK,IPRT
C
C LOGICAL IPRT,ICHP
C
C NAME IPRT ASSUMPTION
C
C XTITL REPORT CASE UMFNLN PAGE TITLE FROM INPUT(124)
C
C XISANG MAXIMUM SUN ANGLE IN RADIANS FROM INPUT
C
C XICOUN LONY FD FROM INPUT (AU)
C
C INTIL CPUR RIDPA TITLES, TITLE FOR EACH WINDOW OCCUPIES 8 WORDS
C
C IE CASE AT 2 CHARACTERS PER WDN.
C
C ICID COLUMNS 74-90 OF INPUT CARD 342
C
C ICASE ACQUIS FILE CASE NO. FROM INPUT
C
C IECASE CHKWINDOW CASE NO. FROM INPUT
C
C INPCC MAXIMUM PERCENT CLOUD COVER * 10 TO BE USED IN SEGMENT SELECT
C
C XISANG MINIMUM DEGREES SUN ANGLE IN DEGREES (10)
C
C IHPLK H BTN UP FROM INPUT (15)
C
C IUPH ZOH FD FROM INPUT (15)
C
C IIFILE LOGICAL UNIT NO. FOR CARD INPUT
C
C IFILE LOGICAL UNIT NO. FOR REPORT OUTPUT
C
C IFFILE LOGICAL UNIT NO. FOR SFRGHTER DATA FILE
C
C IXFILE LOGICAL UNIT NO. FOR CHKWINDOW FILE
C
C IXFILE LOGICAL UNIT NO. FOR ACQUSITION FILE
C
C INRIDE NO. LINES PER PAGE
C
C IPRTD PRINTING LOAD REPORT FLAG (T=PRINT REPORT)
C
C IECHP ACHRITION (CHP WINDOW) REPORT FLAG (T=PRINT REPORT)
C
C COMMON BLOCK 1BLK=PROGRAM COUNTERS AND CONSTANTS
C
C COMMON 1BLK
C
C IXPARK(140),ITRL,IIPType,IPRUN,INL(15),IPNDFK(150),ILTV(5)
C
C
2IPLEC(13), IPFEU
LOGICAL IPOF

C NAME DESCRIPTION
C XPMWK FLOATING POINT WORK AREA, DIMENSION 150
C IPFIRST FIRST TIME FLAG (0=FIRST TIME, 1=SECOND, 2=LAST)
C INTYPE TYPE OF ACQUISITION (0=IDENTIFICATION, 1=TRAINING, 2=SPRING)
C IPYRN ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM RECORD PRINT
C IPCD HOLLENITH ARRAY (0/1/2/3/4/5/6/7/8/9) = 0/0/0/1/
C ITEN POINTS OF TEN (110, 100, 1000, 10000)
C IPMWK INTENER AREA, DIMENSION 150
C IPSEQ END OF FILE FLAG (SEQUENCE FILE)
C IPEFPL FATAL LENGTH FLAG FOR EACH 1-15 (T=FATAL)
C COMMON BLOCK SFLK-SEQUENCE FILE HEADER AND DATA RECORD VARIABLES
C COMMON/SHARE
10IRK=IPNAME(2) 5TRL=SCOUNH SPCHN=SPCWH SSANG(SSANG(150))
21INC=ISDAY=ISAC=ISTJ=ISTUR=ISTRA,
3ISURR, ISTSC, IJSNK, ISNKDAY(150), TSPCC(150) NVEN
4, JUHR(150), IDAY(150)

C NAME DESCRIPTION
C SFNAM SFLK HEADER FILE NAME PHONETIC=SLCHAFCER*
C SFNAM2 SFLK HEADER FILE NAME FROM LABEL
C SFLTPL TRAILER ID *779*
C SCOUN SFLK COUNTRY ID (LQ-2)
C SPCN CHAINING SPACING WHAT (FC=1)
C SPCW HIGHT HIGHT WHAT (FC=2)
C SSANG SSANG 450 VLAU(LH) FOR EACH OF 150 SEGMENTS
C SSSAS SFLK HEADER CASE NO. FROM LABEL
C SDAFF SFLK HEADER REFERENCE DATE-LDDT1-
C ISDAY NO. DAYS IN STUDY (-999)
C ISAC NO. ACQUISITIONS IN RECORD (0-150)
C ISHE SFLK HEADER REGION ID
C ISZDE SFLK HEADER ZONE ID
C ISTRA SFLK HEADER STRATA ID
C LTSN SFLK HEADER SUBSTRATA ID
C ISSEG SFLK HEADER SEGMENT ID
C ISMHO TRAINING SEGMENT INDICATION(0=NORMAL, 1=TRAINING)
C ISDAY ACQUISITION DAY FOR 150 ACQUISITIONS-3012-
C IF END OF ALL ACQUISITIONS IN RECORD
C ISAC PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS
C IF LAST NOT ACQ IN TABLE
C INOK ORIGIN UNDER ACESS DATES AS READ FROM SFLK. REF. FILE
C COMMON/MARK
10JACNA=NAMEx(2) 5TRL=ARAC(406)
15JACNEX(+5) JAP(25,4) IAPM, IAD, IACH(4,0)
C NAME DESCRIPTION
C SFNAM SFLK HEADER FILE NAME *ACQUISI *
C SFNAM2 SFLK HEADER FILE NAME FROM LABEL
C SFLTPL TRAILER ID *779*
C SCOUN SFLK COUNTRY ID (LQ-2)
C SPCN CHAINING SPACING WHAT (FC=1)
C SPCW HIGHT HIGHT WHAT (FC=2)
C SSSAS SFLK HEADER CASE NO. FROM LABEL
C SDAFF SFLK HEADER REFERENCE DATE-LDDT1-
C J NIK ORIGIN UNDER ACESS DATES AS READ FROM SFLK. REF. FILE
C COMMON/ADDK
10JACNA=NAMEx(2) 5TRL=ARAC(406)
15JACNEX(+5) JAP(25,4) IAPM, IAD, IACH(4,0)
C NAME DESCRIPTION
C SFNAM SFLK HEADER FILE NAME *ACQUISI *
C SFNAM2 SFLK HEADER FILE NAME FROM LABEL
C SFLTPL TRAILER ID *779*
C SCOUN SFLK COUNTRY ID (LQ-2)
C SPCN CHAINING SPACING WHAT (FC=1)
C SPCW HIGHT HIGHT WHAT (FC=2)
C SSSAS SFLK HEADER CASE NO. FROM LABEL
C SDAFF SFLK HEADER REFERENCE DATE-LDDT1-
C J NIK ORIGIN UNDER ACESS DATES AS READ FROM SFLK. REF. FILE
C COMMON/ADDK/
C IAPAG  ACQUISITION REPORT CURRENT PAGE NUMBER
C JAMLIN  ACQUISITION REPORT NO. LINFS REMAINING CURRENT PAGE
C JACMIL  NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 8 WINDOWS
  DIMENSION IRDATE(S)
  IF (LNOT. IUPPT) GO TO 10
C INITIALIZE VARIABLES
  CALL JNHU (ISDATE, IUPDAI)
  NCPAG = 0
  NACO = 0
  NLIN = (TODAY+3)/8
  NLINP = NLIN
C INITIALIZE PAGE VARIABLES AND PRINT PAGE/COLUMN TITLES
  NCPAG = NCPAG + 1
  NLINP = NLINP (NLIN + 9)
  NACO = NACO + 1
  NACO = NACO + 1
  NACO = NACO + 1
  NACO = NACO + 1
  NACO = NACO + 1
  NACO = NACO + 1
  NACO = NACO + 1
  WRITL (TOPFILE, 100) NACASE, XITTL, NCPAG
  WRITL (TOPFILE, 110) IRDATE(2), IRDATE(3), IRDATE(4), XICOUN, IREG, IZONE
C WRIT LP IN 25 DETAIL LINES PER PAGE
  DO 30 IR = NLINP
    NACO = NACO + 1
    NACO = NACO + 1
    NACO = NACO + 1
    NACO = NACO + 1
    NACO = NACO + 1
    NACO = NACO + 1
    NACO = NACO + 1
    NACO = NACO + 1
    WRITL (TOPFILE, 160) NACO1, ANACU1, NACO2, ANACU2, NACO3, ANACU3
  CONTINUE
C 30 CONTINUE
  IF (NLINP .GT. 0) GO TO 10
  WRITE (TOPFILE, 170)
  EXIT
  WRITE (TOPFILE, 170)
  NLINP = NLINP - 25
  NLINP = NLINP - 25
  STOP
C 100 FORMAT (AHICASE, 14, 50H DAILY PROCESSING LOAD REPORT, 1241, 16H LPP
C ISIMULATION , 5UPARGS , H1)
C 110 FORMAT (1HI0, 9X, 1HSTADI UAIF , 12, 1H-12, 1H-12, 1H-2X, 6HCOUNTRY , 1A4, 2X, 4HPP
C 173HPRF(1N13+2X15HIZONE+13/7))
C 130 FORMAT (110)
C 140 FORMAT (10H)
C 150 FORMAT (10H)
C 160 FORMAT (X5, SINU1, 6X, HRELATIVE5X, SINH1, 6X, AMPLATIVE1H)
C 170 FORMAT (X5, SINU1, 6X, HRELATIVE5X, SINH1, 6X, AMPLATIVE1H)
C 180 FORMAT (10H)
C 190 FORMAT (10H)
C 200 FORMAT (10H)
C 210 FORMAT (10H)
C 220 FORMAT (10H)
C 230 FORMAT (10H)
C 240 FORMAT (10H)
C 250 FORMAT (10H)
C 260 FORMAT (10H)
C 270 FORMAT (10H)
C 280 FORMAT (10H)
C 290 FORMAT (10H)
C 300 FORMAT (10H)
C 310 FORMAT (10H)
C 320 FORMAT (10H)
C 330 FORMAT (10H)
C 340 FORMAT (10H)
C 350 FORMAT (10H)
C 360 FORMAT (10H)
C 370 FORMAT (10H)
C 380 FORMAT (10H)
C 390 FORMAT (10H)
C 400 FORMAT (10H)
C 410 FORMAT (10H)
C 420 FORMAT (10H)
C 430 FORMAT (10H)
C 440 FORMAT (10H)
C 450 FORMAT (10H)
C 460 FORMAT (10H)
C 470 FORMAT (10H)
C 480 FORMAT (10H)
C 490 FORMAT (10H)
C 500 FORMAT (10H)
C 510 FORMAT (10H)
C 520 FORMAT (10H)
C 530 FORMAT (10H)
C 540 FORMAT (10H)
C 550 FORMAT (10H)
C 560 FORMAT (10H)
C 570 FORMAT (10H)
C 580 FORMAT (10H)
C 590 FORMAT (10H)
C 600 FORMAT (10H)
C 610 FORMAT (10H)
C 620 FORMAT (10H)
C 630 FORMAT (10H)
C 640 FORMAT (10H)
C 650 FORMAT (10H)
C 660 FORMAT (10H)
C 670 FORMAT (10H)
C 680 FORMAT (10H)
C 690 FORMAT (10H)
C 700 FORMAT (10H)
C 710 FORMAT (10H)
C 720 FORMAT (10H)
C 730 FORMAT (10H)
C 740 FORMAT (10H)
C 750 FORMAT (10H)
C 760 FORMAT (10H)
C 770 FORMAT (10H)
C 780 FORMAT (10H)
C 790 FORMAT (10H)
C 800 FORMAT (10H)
C 810 FORMAT (10H)
C 820 FORMAT (10H)
C 830 FORMAT (10H)
C 840 FORMAT (10H)
C 850 FORMAT (10H)
C 860 FORMAT (10H)
C 870 FORMAT (10H)
C 880 FORMAT (10H)
C 890 FORMAT (10H)
C 900 FORMAT (10H)
C 910 FORMAT (10H)
C 920 FORMAT (10H)
C 930 FORMAT (10H)
C 940 FORMAT (10H)
C 950 FORMAT (10H)
C 960 FORMAT (10H)
C 970 FORMAT (10H)
C 980 FORMAT (10H)
C 990 FORMAT (10H)
C 1000 FORMAT (10H)
C 1010 FORMAT (10H)
C 1020 FORMAT (10H)
C 1030 FORMAT (10H)
C 1040 FORMAT (10H)
C 1050 FORMAT (10H)
C 1060 FORMAT (10H)
C 1070 FORMAT (10H)
C 1080 FORMAT (10H)
C 1090 FORMAT (10H)
C 1100 FORMAT (10H)
C 1110 FORMAT (10H)
C 1120 FORMAT (10H)
C 1130 FORMAT (10H)
C 1140 FORMAT (10H)
C 1150 FORMAT (10H)
C 1160 FORMAT (10H)
C 1170 FORMAT (10H)
C 1180 FORMAT (10H)
C 1190 FORMAT (10H)
C 1200 FORMAT (10H)
C 1210 FORMAT (10H)
C 1220 FORMAT (10H)
C 1230 FORMAT (10H)
C 1240 FORMAT (10H)
C 1250 FORMAT (10H)
C 1260 FORMAT (10H)
C 1270 FORMAT (10H)
C 1280 FORMAT (10H)
C 1290 FORMAT (10H)
C 1300 FORMAT (10H)
C 1310 FORMAT (10H)
C 1320 FORMAT (10H)
C 1330 FORMAT (10H)
C 1340 FORMAT (10H)
C 1350 FORMAT (10H)
C 1360 FORMAT (10H)
C 1370 FORMAT (10H)
C 1380 FORMAT (10H)
C 1390 FORMAT (10H)
C 1400 FORMAT (10H)
C 1410 FORMAT (10H)
C 1420 FORMAT (10H)
C 1430 FORMAT (10H)
C 1440 FORMAT (10H)
C 1450 FORMAT (10H)
C 1460 FORMAT (10H)
C 1470 FORMAT (10H)
C 1480 FORMAT (10H)
C 1490 FORMAT (10H)
C 1500 FORMAT (10H)
C 1510 FORMAT (10H)
C 1520 FORMAT (10H)
C 1530 FORMAT (10H)
C 1540 FORMAT (10H)
C 1550 FORMAT (10H)
C 1560 FORMAT (10H)
C 1570 FORMAT (10H)
C 1580 FORMAT (10H)
C 1590 FORMAT (10H)
C 1600 FORMAT (10H)
C 1610 FORMAT (10H)
C 1620 FORMAT (10H)
C 1630 FORMAT (10H)
C 1640 FORMAT (10H)
C 1650 FORMAT (10H)
C 1660 FORMAT (10H)
C 1670 FORMAT (10H)
C 1680 FORMAT (10H)
C 1690 FORMAT (10H)
C 1700 FORMAT (10H)
C 1710 FORMAT (10H)
C 1720 FORMAT (10H)
C 1730 FORMAT (10H)
C 1740 FORMAT (10H)
C 1750 FORMAT (10H)
C 1760 FORMAT (10H)
C 1770 FORMAT (10H)
C 1780 FORMAT (10H)
C 1790 FORMAT (10H)
C 1800 FORMAT (10H)
C 1810 FORMAT (10H)
C 1820 FORMAT (10H)
C 1830 FORMAT (10H)
SUBROUTINE AREAD

C******************************************************************************
C ROUTINE AREAD
C PURPOSE
THIS ROUTINE READS BOTH THE CROP WIND AND SEGREGATION DATA FILES AND
озвращает a record of each or sets a flag to INDICATE AN END OF DATA HAS BEEN ENCOUNTERED.
C LINKAGE
CALL AREAD
C LOCAL VARIABLES
T     THROW AWAY INTEGER ARRAY DIMENSION 150
Y     THROW AWAY REAL ARRAY DIMENSION 150
TOFLG FLAG TO INDICATE ACQUISITIONS ARE TO BE COUNTED
ONLY 0=ACCEPT AND COUNT, 1=COUNT ONLY
C ROUTINES CALLED
ADARE (HOUR ROUTINE)
15-COUNTRY NOT EQUAL FOR INPUT AND 2 FILES
16-CROPWIND MISSING SEGREGATION DATA
13-RECORD SKIPPED OUT TO 0 PERCENT WHEAT
C******************************************************************************

C COMMON_BLOCK_IBLK=MACHINE INPUT PARAMETERS CONSTANTS

COMMON /IBLK/
  IXITLT(2),IXISANG,IXICOUN,IXITLL(14),
  2IFIN(3),IACAS,ICASE,ICHALSE,1PCG,ISANG,INEG,IZONE,TIFILE,IOFILE;
  JIM
  3ISFILE,1CFILE,1AFILE,1IM,L;
  4IPRT1,IPRT2
  5LOGICAL IPRT1,IPRT2

C NAME DESCRIPTION
C IXITLT REPORT CASE UNIDENTIFIED PAGE TITLE FROM INPUT (12A4) IBLK
C IXISANG MINIMUM SUN ANGLE IN RADIANS FROM INPUT IBLK
C IXICOUN COUNTRY ID FROM INPUT (42) IBLK
C IXITLL WINDOW TITLE TITLE FOR EACH WINDOW OCCUPIES 8 WORDS IBLK
C JIM AT 2 CHARACTERS PER WORD JIM
C 1ICD LIGHTS /-RAO DF INPUT CARD 3A2 IBLK
C 1ACAS ACQUISITION FILE CASE NO. FROM INPUT IBLK
C 1ICASE SEGREGATION CASE NO. FROM INPUT IBLK
C 1ICHALSE CHALSE CASE NO. FROM INPUT IBLK
C 1IPCG MAXIMUM PERCENT CLOUD COVER * 10 TO BE USED IN SEGMENT SELECT IBLK
C 1ISANG MINIMUM DEGREES SUN ANGLE * 100 (IN DEGREES) (14) IBLK
C 1IREG HEIGHT TO FRONT INPUT (13) IBLK
C 1IZONE ZONE TO FRONT INPUT (13) IBLK
C 1IFILE LOGICAL UNIT NO. FOR FRONT INPUT IBLK
C 1ITFILE LOGICAL UNIT NO. FOR REPORT OUTPUT IBLK
C 1ISFILE LOGICAL UNIT NO. FOR SEGREGATION DATA FILE IBLK
C 1ITFLG LOGICAL UNIT NO. FOR CROPWIND FILE IBLK
C 1ITLL LOGICAL UNIT NO. FOR ACQUISITION FILE IBLK
C 1ILINT NO. LINES PER PAGE IBLK
C 1IPRT1 PRINTING LOG REPORT FLAG (T=PRINT REPORT) IBLK
C 1IPRT2 ACQUISITION: (T=AN WINDOW) REPORT FLAG (T=PRINT REPORT) IBLK
C ICASN CASE NO. FROM LABEL
C ICNT NO. OF CRIP WINDOWS (R)
C ICPRG CRIP REGION ID
C ICZONE CRIP ZONE ID
C ICSTHA CRIP STATION ID
C ICSPHS CRIP SUBSTATION ID
C ICLWD START DATE OF 6 WINDOWS ZULU-RELATIVE JAN 1950
C ICHR STOP DATE OF 6 WINDOWS ZULU-RELATIVE JAN 1950
C DIMENSION 111500*X(150),ITEMP(8)*ITEMP(8)
C EQUIVALENCE (TEMP(+1)
C EQUVALANCE (MRK+X)
C IACCT = 7*

C READ A RECORD FROM SEGREFER FILE
C READ A RECORD FROM SEGREFER FILE
C SCHR=SCRN
C IF(SCRN,*,PSTRING) GOTO 200
C IF(SCRN,*,PSTRING) GOTO 21
C IF(TEMP(5),*,PSTRING) GOTO 21
C IF(NO, *, PSTRING) = EXIT TO END FILE EXIT
C CALL ADRF(5)
C GOTO 200
C IDENTIFY TYPE OF WHEAT: I = WINTER, S = SPRING
C IF NO PPMR 13
C CONTINUE
C IF(ISPRM,*,PSTRING) GOTO 215
C IPTYPE = 5
C GOTO 20
C IPTYPE = 1
C COMARE INPUT INPUT ID FIELDS WITH SEGREFER ID FIELDS
C GT SKIP SEGREFER RECORD
C IF(ISZONE,*,PSTRING) GOTO 21
C IF(ISZONE,*,PSTRING) GOTO 21
C IF(ISZONE,*,PSTRING) GOTO 21
C PROPER SEGREFER RECORD COMARE WITH CRIP INPUT RECORD
C IF(ISZONE, *, PSTRING) GOTO 21
C IF(ISZONE, *, PSTRING) GOTO 21
C IF(ISZONE, *, PSTRING) GOTO 21
C IF(ISZONE, *, PSTRING) GOTO 21
C IF(ISZONE, *, PSTRING) GOTO 21
C READ RECORD FROM CRIP INPUT FILE
C READ RECORD FROM CRIP INPUT FILE
C READ RECORD FROM CRIP INPUT FILE
C READ RECORD FROM CRIP INPUT FILE
C READ RECORD FROM CRIP INPUT FILE
C CONTINUE
C INCREDENT NO. RECORDS READ AND CHECK FOR TRAILER
C CONTINUE
C CONTINUE
C CONTINUE
C CONTINUE
C CONTINUE
C CONTINUE
C CONTINUE
C CONTINUE
C CONTINUE
C TEST CROPSWIND COUNTRY LEGAL IF NOT ERROR

C TEMPORARY BYPASS FOR CHECKOUT
IF (XICOUN .GT. 0.) THEN.
  IF(COUNCFL. XICOUN) GO TO 26
ENDIF

90 IF (ICOUNT .EQ. ICOUNF) 120, 120+120
100 IF (ICOUNT = ICOUNF) 120, 120+120
110 IF (ICOUNT = ICOUNF) 120, 120+120

C CROPWIND MISING A ZONE FOR SEGREFER-PRINT ERROR
120 CALL AHANT(12)
GO TO 10

C END OF FILE EXIT
200 IF (EF = 'TRUE')
GO TO 900

C COUNT NO. OF ACCESS DATES AND ARRANGE THEM IN INCREASING ORDER IRRESPECTIVE OF VEHICLE NUMBER.

210 NDDAY = 0
DO 220 J = 1, 1150
IF (ISHDAY(J) .EQ. 0) GO TO 230

220 CONTINUE
NDDAY = NDDAY + 1
J

230 DO 250 IZ = 1, 1150

240 CONTINUE
ISHDAY(IZ) = ISRNAY(IZ)

250 CONTINUE
CALL SURTAG(ISHDAY + 1, NDDAY, ORDER).

C NOW PERMUTE SUN ANGLE AND PERCENT CLOUD COVER ARRAYS TO JIM

C AGREE WITH SORTED ORDER OF DATES.

C

DO 260 J = 1, NDDAY
XPWRK(J) = SSAM(J)

260 CONTINUE
IPRCC(J) = 1

DO 290 J = 1, NDDAY
J = 10KIC(J)
SSAM(J) = XPWRK(J)

290 CONTINUE
IPRCC(J) = 1

900 RETURN
END

ORIGINAL PAGE IS OF POOR QUALITY.
COMMON BLOCK FILE=COUNCIL HEAD AND DATA

COMMUN(PLK)
  ICNAM1(FANAM1)IICNAM2(FANAM2)ICOUNT
  ICNAM1 FCNAM1 REORD NO.
  ICNAM2 LABEL FILE NAME MUST MATCH ICNAM1
  COUNT COUNCIL COUNTRY TO AQ
  ICNAME COUNCIL NAME FROM LABEL
  ICNAME SUB OF COUNCIL (R)
  ICNAME COUNCIL REGION
  ICNAME COUNCIL ZONE ID
  ICNAME COUNCIL STRATA ID
  ICNAME COUNCIL SUBSTRATA ID
  IIFLOW START DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950
  ICINT STOP DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950

COMMON BLOCK IFLK=AQUIVISI FILE AND REPORT VARIABLES

COMMUN/PLK/
  IAQIDUS(25)+IAQIDUP+IAQIDUP(25)

NAME DESCRIPTION
  IAQIDUS CURRENT RECORD NO.
  IAQIDUP LAPEL FILE NAME=AQUIVISI *
  IAQIDUP TRAILER ID A
  IAQIDUS NO. OF ACQUISITIONS SELECTED FOR EACH OF 262 DAYS
  IAQIDUS TOTAL NO. OF ACQUISITIONS FOR EACH OF 4 WINDOWS
  IAQINT TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS
  IAQINT INDEX TO SPREADER ACQUISITION (1-150) THAT IDENTIFIES
  IAQINT A SELECTED SPREADER, IAQINT(1,J,J)=WINDOW 1 THROUGH 4
  IAQINT ACQUISITION REPORT CURRENT PAGE NUMBER
  IAQINT ACQUISITION REPORT NO. LING REMAINING CURRENT PAGE
  IAQINT NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS

COMMON BLOCK PFLK=MANUAL INPUT PARAMETERS CONSTANTS

COMMUN/PLK/
  IFLK(1)+IFLKD+IFLKD(16)+
  IFLK(1)+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLKD+IFLK
C INTTL CHUMP WINDOW TITLE, TITLE FOR EACH WINDOW OCCUPIES 8 WORDS
C AT 2 CHARACTERS PER WORD.
C ICID COLUMNS 7 TO 9 OF INPUT CARD 342
C IACASE ACQUISS FILE CASE NO. FROM INPUT
C ICASE SLGHEED CASE NO. FROM INPUT
C ICASE CHMUND CASE NO. FROM INPUT
C IPECC MAXIMUM PERCENT CLOUD COVER * 10 TO BE USED IN SEGMENT SELECT
C ISANG MINIMUM DEGREES SUN ANG = 100 (IN DEGREES) (14)
C IREG REGION TO FROM INPUT (15)
C IPUNE ZONE TO FROM INPUT (16)
C IIFILE LOGICAL UNIT NO. FOR CARD INPUT
C IIPF FILE LOGICAL UNIT NO. FOR REPORT OUTPUT
C ISPHE LOGICAL UNIT NO. FOR SLGHEED DATA FILE
C IIPF FILE LOGICAL UNIT NO. FOR CHUMPWINDOW FILE
C IAFILE LOGICAL UNIT NO. FOR ACQUISITION FILE
C IUNL:NE # LINES PER PAGE
C IPRINT PRINTING LOAD REPORT FLAG (T=PRINT, Report)
C ICMPX ACQUISITION CHUMP WINDOW REPORT FILE (T=PRINT REPORT)
C ICHMPX ACQUISITION CHUMP WINDOW REPORT FILE (T=PRINT REPORT)
C DENT DIMENSION KANP(100)
C EQUIVALENCE(KANP(1),TAPNT(1))
C T TEST FIRST TIME FLAG
C J IF 0, WRITE THAILER
C E 1 0, WRITE HEAOER
C G 1 0, WRITE ACQUISITION RECORD
C WRITE HEAOER RECORD
DO 11 1=1,6A
IPMK(1)=1
11 CONTINUE
C WRITE (IAFILE) AFNAM,IACASE,ISMDAY,IPNCH,(INTTL(L),L=1,16)
C 2 =ICASE, ISECASE* (IPMK(L),L=1,94)
C IPF=1
C ACH=0
C G010 40
C WRITE THAILER- SET UP FAKE DETAIL RECORD
DO 20 71=1,100
KX(1)=0
20 CONTINUE
C WRITE (IAFILE) SCWHE,AFTRL
C ISHF=099
C ISZNF=999
C G0 TO 65
C CONTINUE
DO 60 60=1,100
IX = KANP(1)
INX (IX .LT. 0)GO TO 45
KX(1)=0
60 CONTINUE
45 KX(1)= IKDAY(IX)
C CONTINUE
C WRITE (IAFILE) SCWHE, ISHFR, ISZNF, ISSTRA, ISURS, ISSF
C 1(*(XX1),L=1,100), IPNCH
C IF IPF=LT .NU END FILE IAPFILE
C INCREMENT RECORD COUNT
C ACHR=ACRH+1
C RETURN
C END
SUBROUTINE BINUC(Ival,NDIG+OUT)

DIMENSION IOUT(10)

CROUTINE

C

CPURPOSE TO CONVERT AN INTEGER TO DISPLAY CODE AND PLACE CHAR.
STRING OUTPUT IN AN ARRAY MA1 (N IS INPUT)

C

CLINKAGE CALL 'BINUC(Ival,NDIG+OUT)

IVAL BINARY INTEGER
NDIG NO. OF DIGITS TO OUTPUT
IOUT ORIGIN OF THE OUTPUT CHARACTER ARRAY

C

CRoutines Called: None

C

CLocal Variables

I CURRENT VALUE OF LEAST SIG. DIGITS
J
K NO. OF DIGITS REMAINING TO PROCESS
N VALUE OF CURRENT MOST SIGNIFICANT DIGIT

C

C

COMMENTS ALGORITHM IS DESCRIBED AS FOLLOWS

A PRODUCTS RIGHT JUSTIFIED 0
B PRODUCTS LEFT JUSTIFIED
C PRODUCED FOR OUT OF RANGE
D BLANK PRODUCED FOR LEADING ZEROS
E NO. GENERATED FROM LEFT TO RIGHT BY REDUCING
F NO. BY DECREASING PUNS OF 10

C

COMMON BLOCK PULK=PROGRAM COUNTERS AND CONSTANTS

C

COMMON PULK/

IPIF(NK(150), IPIFF, IPIFTYPE, IPIFGR, ICD(15), IPIFRR(150), IOE(5),
PIFFLR(15), IPIFUR)

LOGICAL IPEF

C

NAME DESCRIPTION

XPIF GR FLOATING POINT WORK AREA DIMENSION 150

JPIFF FIRST TIME FLAG (0=FIRST TIME, 1=SUBSEQUENT FUTL. = LAST)

IPETYPE TYPE WaTeR (O=UNIDENTIFIED, 1=INTER, 5=SPRING)

IPFGR ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM (LWNORK PRINT)

IPFRR MILLIRH March 1/2/3/4/5/6/7/8/9/10/11/12

JPEE PMNS OF FFN (1=10, 100=1000, 10000)

IPFRRM INTEGER WORK AREA DIMENSION 150

IPFRM END OF FILE FLAG (REQUESTED FILE)

IPFREL FLOATING ERROR FLAG FOR ERROR 1-13 (I FAI TAL)

JIVAL

I=1

J=0

ICD(I)=ICD(I)

C TEST INPUT VALUE 0, NEGATIVE, OR OUT OF RANGE
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000049</td>
<td>C</td>
<td>IF(J) = 0:20:30</td>
</tr>
<tr>
<td>000050</td>
<td>10</td>
<td>VALUE IS NEGATIVE</td>
</tr>
<tr>
<td>000051</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000052</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000053</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000054</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000055</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000056</td>
<td>K = K - 1</td>
<td></td>
</tr>
<tr>
<td>000057</td>
<td>I = I + 1</td>
<td></td>
</tr>
<tr>
<td>000058</td>
<td>J = J</td>
<td></td>
</tr>
<tr>
<td>000059</td>
<td>GOTO 70</td>
<td></td>
</tr>
<tr>
<td>000060</td>
<td>C</td>
<td>VALUE IS ZERO</td>
</tr>
<tr>
<td>000061</td>
<td>20</td>
<td>NOT (N) = IBCD</td>
</tr>
<tr>
<td>000062</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000063</td>
<td>00</td>
<td>GOTO 100</td>
</tr>
<tr>
<td>000064</td>
<td>C</td>
<td>TEST INPUT VALUE IN RANGE</td>
</tr>
<tr>
<td>000065</td>
<td>30</td>
<td>N = J - I - IBCD(K + 1)</td>
</tr>
<tr>
<td>000066</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000067</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000068</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000069</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000070</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000071</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000072</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000073</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000074</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000075</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000076</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000077</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000078</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000079</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000080</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000081</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000082</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000083</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000084</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000085</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000086</td>
<td></td>
<td></td>
</tr>
<tr>
<td>000087</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**ORIGIN: PAGE 29 OF POOR QUALITY**

**PAGE 1932**

**28246028-RO-00**
SUBROUTINE FZULU(I DATE, I OUT)

C
C ROUTINE FZULU
C
C PURPOSE TO CONVERT ZULU DATE TO YEAR MONTH AND DAY
C
C LINKAGE CALL FZULU (I DATE, I OUT)
C I DATE: BINARY INTEGER ZULU DATE
C I OUT: CALENDER DATE+DIMENSION 3 FOR INTEGER
C YEAR+ MONTH AND DAY RESPECTIVELY
C
C ROUTINES CALLED NONE
C
C LOCAL VARIABLES
C IDAYS VECTOR CONTAINING NO. DAYS OF YEAR ON
C LAST DAY OF MONTHS 0 THROUGH 12 FOR NORMAL YEAR
C FOLLOWED BY 13 MONTHS OF LEAP YEAR
C
C NLLEAP: NO. OF LEAP YEARS SINCE 1900 TO SPECIFIED
C DATE
C
C NFLG: SUBSCRIPT IN TO IDAYS VECTOR
C START=1, NOT LEAP YEAR
C START=14, LEAP YEAR
C
C JDAY: JULIAN DAY OF YEAR
C
C comments
C
C LOCAL USE ('NFLG'
C
C
C DIMENSION IOUT (3)
C DIMENSION IDAYS(28)
C DATA IDAYS/0,31,59,90,120,151,181,212,243,273,304,335,365,
C 1 0,31,59,90,121,152,182,213,244,274,305,336,365,
C 5 0,31,59,90,121,152,182,213,244,274,305,336,365,
C 0
C
C CONTINUE
C
C FIND NO. LEAP YEARS SINCE 1900 AND GET CURRENT YEAR
C
C NLLEAP=(1995-I DATE)/400
C I OUT(1)=(I DATE-NLLEAP)/365
C
C REF IF CURRENT YEAR IS LEAP YEAR=SET NFLG=1 OR 14
C
C I=IOUT(1)/4
C I=I4
C NFLG=1
C IF (IOUT(1)-1) 19+10+20
C NFLG=14
C
C JULIAN DAY = ZULU DAY+NO YEARS+365+NO LEAP YEARS
C
C 20 JDAY=I DATE-(IOUT(1)*365)+NLLEAP+1
C
C GET DAY MONTH FROM TABLE SEARCH
C
C 30 IF (JDAY+IDAYS(NFLG))50+50+40
C 40 IOUT(2)=IOUT(2)+1
C NFLG=NFLG+1
C LOUT 40
C
C 50 NFLG=NFLG-1
C IOUT(1)+JDAY-IDAYS(NFLG)
C NFLG=0
C
C 70 CONTINUE
C
C 70 IPFTR=1 WRITE TRAILER ACQUISITION FILE
C CALL AWRTA PRINT STATISTICS CRUPE WIND REPORT
C CALL ACRPT PRINT DAILY PRODUCTION REPORT
CALL APRT STOP
END
SUBROUTINE SORITAG(A(1:1), L)  
C SORTS ARRAY A INTO INCREASING ORDER, FROM A(1) TO A(JJ)  
C ARRAY TAG IS PERMUTED THE SAME AS ARRAY A  
C ORDERING IS BY INTEGER SUBTRACTION; THUS FLOATING POINT  
C NUMBERS MUST BE IN NORMALIZED FORM.  
C ALWAYS IN(I) AND IJ(I) PERMUT SORTING UP TO 2*K(I)+1 ELEMENTS  
C CDC 6600 TIME IS 5.2A SEC. FOR 10,000 RANDOM ITEMS.  
C AND MACHINE TIME PROPORTIONAL TO N*LOG2(N)  
C  
C *  STURLINGTON & SPICHERER 1968  
C  
C INFORMAL A(I), IJ(I), IJ(I) , TAG(I)  
K=1  
I=15  
J=11  
5 IF (I .GE. J) GO TO 70  
10 K=1  
I=I(J+1)  
T=A(I(J))  
IF (T(I), I.E. T) GO TO 20  
A(I(J))=A(I)  
A(I)=T  
T=A(I(J))  
TAG(I(J))=TAG(I())  
20 =J  
I=I(J)  
IF (A(I(J)) .GE. T) GO TO 40  
A(I(J))=A(I)  
A(I)=T  
T=A(I(J))  
TAG(I(J))=TAG(I())  
40 L=J  
I=I(J+1)  
IF (A(I(J)) .LT. T) GO TO 40  
A(I(J))=A(I)  
A(I)=T  
T=A(I(J))  
TAG(I(J))=TAG(I())  
GO TO 40  
30 A(I(J))=A(K)  
A(K)=T  
T=A(I(J))  
TAG(I(J))=TAG(K)  
TAG(K)=T  
GO TO 30  
40 L=K  
IF (A(I(J)) .LT. L) GO TO 40  
T=A(I(J))  
50 K=K+1  
IF (A(K) .LT. L) GO TO 50  
K=K  
I=I(J+1)  
IF (I(J) .LT. J) GO TO 50  
I=I(J)  
I=I(J)  
I=I(J)  
I=I(J)  
J=K  
I=I(J)  
GO TO 40
## Table of Contents

<table>
<thead>
<tr>
<th>Part</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part I</td>
<td>Problem Description</td>
<td>203</td>
</tr>
<tr>
<td>Part II</td>
<td>Common Block Definitions</td>
<td>236</td>
</tr>
<tr>
<td>Part III</td>
<td>List of Subroutines and Subroutine Call Structure</td>
<td>263</td>
</tr>
<tr>
<td>Part IV</td>
<td>Subroutine Descriptions and Flowcharts</td>
<td>268</td>
</tr>
<tr>
<td>Part V</td>
<td>Subroutine Listings</td>
<td>313</td>
</tr>
</tbody>
</table>
PART I

PROBLEM DESCRIPTION FOR

THE POST PROCESSOR
## Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 Scope</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Program Capabilities</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Method of Program Development</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Operational Assumptions</td>
<td>2</td>
</tr>
<tr>
<td>2.0 Input</td>
<td>3</td>
</tr>
<tr>
<td>2.1 Cards</td>
<td>3</td>
</tr>
<tr>
<td>2.2 Files</td>
<td>4</td>
</tr>
<tr>
<td>3.0 Processing</td>
<td>11</td>
</tr>
<tr>
<td>3.1 Overview</td>
<td>11</td>
</tr>
<tr>
<td>3.2 Program Flow</td>
<td>12</td>
</tr>
<tr>
<td>3.3 Procedures and Equations</td>
<td>17</td>
</tr>
<tr>
<td>3.4 POUT Symbol Table</td>
<td>23</td>
</tr>
<tr>
<td>4.0 Output</td>
<td>26</td>
</tr>
<tr>
<td>4.1 Printed Data</td>
<td>26</td>
</tr>
<tr>
<td>5.0 Error Processing</td>
<td>30</td>
</tr>
<tr>
<td>5.1 Input Errors</td>
<td>30</td>
</tr>
<tr>
<td>5.2 Processing Errors</td>
<td>31</td>
</tr>
</tbody>
</table>
Problem Description for the Post Processor

1.0 SCOPE

1.1 PROGRAM CAPABILITIES

Depending on the user specified option flags, the POUT program will read from associated LACIE file(s) and produce a printed report.

The four major types of reports that can be produced are:

1. Substrata Reference Data Report - the data is processed from the Substrata Historical File (SUBHST).

2. Population Mean, Standard Deviation and Histogram Reports. There are five parameter types in this group. Each parameter produces its own report.
   a. Population Sampling Error (Segment Truth File - SEGTRU)
   b. Population CAMS Error (CAMS Output File - CAMSF)
   c. Population Yield Error (YES Output File - YESOUT)
   d. Population Area Error (CAS Cum Output File - CASF)
   e. Population Production Error (CAS Cum Output File - CASF)

3. Histograms of Monte Carlo Statistics Reports. There are four parameter types in this group. Each parameter produces its own report except for Confidence Level which produces six reports. All data is processed from the CAS Distribution Output File (CASDIS).
   a. Monte Carlo Area Error
   b. Monte Carlo Production Error
   c. Monte Carlo Yield Error
   d. Confidence Level
      - Area Confidence Level Est/Est
      - Area Confidence Level True/Est
4. Frequency of Sample Segment Acquisitions Report - the data is processed from the Data Acquisition File (ACQUIS).

1.2 METHOD OF PROGRAM DEVELOPMENT

The program will be developed in FORTRAN. The direct access file processing is the only known conversion problem and will be handled in one routine. Modular programming techniques will be used throughout to make the program development, modification and debugging easier.

1.3 OPERATIONAL ASSUMPTIONS

- Only one major type of printed report will be produced per run.
- Population reports will be produced by option for either zone, region or country level.
- For Population or Monte Carlo reports any one, a set, or all of the parameters can be selected in one run.
- The Substrata Historical File will also be required to produce the first three Population reports.
- Printed report control will be by card input.
- All control card input data will be echo printed.
- All control card input data will be checked for errors before any error will cause the processing to terminate in the middle of a case.
- External print units will be in English, an optional override is available for metric units via the control card input.
- The Header Card and the four Data Cards are required input.
- All data to be processed will be obtained from disk (or tape) files.
- All input data files will be checked for correct case numbers.
2.0 INPUT

Program input will be in two forms.

2. File input for requested option data.

2.1 CARDS

2.1.1 List of Data Quantities

See Table 2-2 for Input Data Description.

2.1.2 Card Formats

"POUT" is punched in card columns 75-78 of all cards. A sequence number is punched in card columns 79-80.

See Figure 2-1 for Data Card Formats.

2.1.3 Deck Set Up

1. Header Card - sequence 01
2. Data Card 1 - sequence 02
3. Data Card 2 - sequence 03
4. Data Card 3 - sequence 04
5. Data Card 4 - sequence 05

2.1.4 Rules for Entering Data on Cards

2.1.4.1 General

1. Integers must be entered right-justified.
2. Alphanumeric names must be entered left-justified.
3. F format numbers must have the decimal point present, i.e., F6.1 +XXX.X
4. The card sequence numbers in CC. 79-80 must be present on all cards.
2.1.4.2 Specific Fields

The correspondence between the RPTYPE, PARMTR and ICASIN input values, and the required input files versus the produced reports is shown in Table 2-1 below.

Histogram ranges, PARMTR, BIOWD, WPRTY, IPRD input values are required only if RPTYPE=02 and =03. LEVEL is required only if RPTYPE=02.

Table 2-1

<table>
<thead>
<tr>
<th>RPTYPE</th>
<th>PARMTR(I) where I =</th>
<th>ICASIN(I) where I =</th>
<th>Req. Input Files</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>-</td>
<td>1</td>
<td>SUBHST</td>
<td>Substrata Reference Data</td>
</tr>
<tr>
<td>02</td>
<td>1*</td>
<td>1</td>
<td>SEGTRU</td>
<td>Population Sampling Error</td>
</tr>
<tr>
<td></td>
<td>2*</td>
<td>2</td>
<td>CAMSF</td>
<td>Population CAMS Error</td>
</tr>
<tr>
<td></td>
<td>3*</td>
<td>3</td>
<td>YESOUT</td>
<td>Population Yield Error</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4</td>
<td>CASF</td>
<td>Population Area Error</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
<td>CASF</td>
<td>Population Production Error</td>
</tr>
<tr>
<td>03</td>
<td>1</td>
<td>1</td>
<td>CASDIS</td>
<td>Monte Carlo Area Error</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>CASDIS</td>
<td>Monte Carlo Production Error</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>CASDIS</td>
<td>Monte Carlo Yield Error</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4</td>
<td>CASDIS</td>
<td>Confidence Level</td>
</tr>
<tr>
<td>04</td>
<td>-</td>
<td>1</td>
<td>ACQUIS</td>
<td>Frequency of Sample Segment Acquisition</td>
</tr>
</tbody>
</table>

*SUBHST file also required for each report.

2.2 FILES

Any one of the following files, depending on the report option selected, will be input. The Substrata Historical File will always be input when any of the first three Population reports are selected.

SUBHST - Substrata Historical File
SEGTRU - Segment Truth File
CAMSF - CAMS Output File
CASF - CAS Cum Output File
CASDIS - CAS Distribution Output File
ACQUIS - Data Acquisition File
YESOUT - YES Output File

Complete descriptions of all files used by the POUT program can be found in the Reference File Description Document.
Table 2-2. Input Data Description

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Dimension</th>
<th>Nominal Value</th>
<th>Range</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IHEADR</td>
<td>-</td>
<td>12</td>
<td>Blanks</td>
<td></td>
<td>-</td>
<td>72 character case header which prints out at the top of every page</td>
</tr>
<tr>
<td>RPTYPE</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>1-4</td>
<td>-</td>
<td>Major type of report identification</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>01 - Substrata Reference Data Report</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>02 - Population Mean, Standard Deviation and Histogram Report</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>03 - Histograms of Monte Carlo Statistics Report</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>04 - Frequency of Sample Segment Acquisitions Reports</td>
</tr>
<tr>
<td>AUNITS</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0, 1</td>
<td>-</td>
<td>External print units flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0 - English units</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Wheat area value in 10,000 acres</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Production value in 100,000 bushels</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yield value in bushels/acre</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>#0 - Metric units</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Wheat area value in 1000 hectares</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Production value in 1000 metric tons</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yield value in quintals/hectare</td>
</tr>
<tr>
<td>START</td>
<td>-</td>
<td>1</td>
<td>-500.0</td>
<td>-999.9 to 999.9</td>
<td>-</td>
<td>Histogram interval start value</td>
</tr>
<tr>
<td>INTVL1</td>
<td>-</td>
<td>1</td>
<td>100.0</td>
<td>0.1 to 100.0</td>
<td>-</td>
<td>Histogram interval value in percent</td>
</tr>
<tr>
<td>BREAK1</td>
<td>-</td>
<td>1</td>
<td>-100.0</td>
<td>-999.9 to 999.9</td>
<td>-</td>
<td>Histogram breakpoint to change interval value</td>
</tr>
<tr>
<td>INTVL2</td>
<td>-</td>
<td>1</td>
<td>5.0</td>
<td>0.1-100.0</td>
<td>-</td>
<td>Histogram interval value in percent</td>
</tr>
<tr>
<td>BREAK2</td>
<td>-</td>
<td>1</td>
<td>100.0</td>
<td>-999.9 to 999.9</td>
<td>-</td>
<td>Histogram breakpoint to change interval value</td>
</tr>
</tbody>
</table>
Table 2-2. Input Data Description (cont'd)

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Dimension</th>
<th>Nominal Value</th>
<th>Range</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTVL3</td>
<td>-</td>
<td>1</td>
<td>100.0</td>
<td>0.1-100.0</td>
<td>-</td>
<td>Histogram interval value in percent</td>
</tr>
<tr>
<td>STOP</td>
<td>-</td>
<td>1</td>
<td>500.0</td>
<td>-999.9 to 999.9</td>
<td>-</td>
<td>Histogram interval terminal value</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A maximum of 51 range intervals is allowed although less may be used. Intervals may not overlap.</td>
</tr>
<tr>
<td>PARMTR</td>
<td>-</td>
<td>5</td>
<td>0</td>
<td>0-3</td>
<td>-</td>
<td>Report parameter type option flags for RPTYPE = 02 or 03. = 0 - no report; ≠ 0 - produce report.</td>
</tr>
</tbody>
</table>

- PARMTR(1)≠0: RPTYPE = 02 Produce Population Sampling Error Report
- PARMTR(1)≠0: RPTYPE = 03 Produce Monte Carlo Area Error Report
- PARMTR(2)≠0: RPTYPE = 02 Produce Population CAMS Error Report
- PARMTR(2)≠0: RPTYPE = 03 Produce Monte Carlo Production Error Report
- PARMTR(3)≠0: RPTYPE = 02 Produce Population Yield Error Report
- PARMTR(3)≠0: RPTYPE = 03 Produce Monte Carlo Yield Error Report
- PARMTR(4)≠0: RPTYPE = 02 Produce Population Area Error Report
- PARMTR(4)≠0: RPTYPE = 03 Produce Confidence Level Report
- PARMTR(5)≠0: RPTYPE = 02 Produce Population Production Error Report

If RPTYPE = 02 or 03 and all PARMTR values are zero, then all reports of that RPTYPE will be produced.
Table 2-2. Input Data Description (cont'd)

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Dimension</th>
<th>Nominal Value</th>
<th>Range</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICASIN</td>
<td>-</td>
<td>5</td>
<td>0</td>
<td>0-9999</td>
<td>-</td>
<td>Case number identification associated with the required input file. A case number must be input for every report produced. Relationship between RPTYPE, PARMTR and ICASIN follows. ICASIN(1) = case no. RPTYPE = 01 = 02 PARMTR(1)#0 = 03 PARMTR(1)#0 = 04 ICASIN(2) = case no. RPTYPE = 02 PARMTR(2)#0 = 03 PARMTR(2)#0 ICASIN(3) = case no. RPTYPE = 02 PARMTR(3)#0 = 03 PARMTR(3)#0 ICASIN(4) = case no. RPTYPE = 02 PARMTR(4)#0 = 03 PARMTR(4)#0 ICASIN(5) = case no. RPTYPE = 02 PARMTR(5)#0 If all PARMTR values are zero for a RPTYPE = 02 or 03, then all the ICASIN values (case number) must be entered for that RPTYPE value.</td>
</tr>
<tr>
<td>LEVEL</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>0-3</td>
<td>-</td>
<td>Parameter report level indicator for RPTYPE = 2. = 1 - reports produced at zone level; = 2 - at region level; = 3 - at country level.</td>
</tr>
<tr>
<td>ICSESH</td>
<td>-</td>
<td>1</td>
<td>0</td>
<td>0-9999</td>
<td>-</td>
<td>Case number identification for the SUBHST input file. Required only when RPTYPE = 2 and PARMTR = 1, 2 or 3.</td>
</tr>
<tr>
<td>Name</td>
<td>Symbol</td>
<td>Dimension</td>
<td>Nominal Value</td>
<td>Range</td>
<td>Units</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>--------</td>
<td>-----------</td>
<td>---------------</td>
<td>--------</td>
<td>-------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BIOWD</td>
<td>-</td>
<td>4</td>
<td>0</td>
<td>0,1</td>
<td></td>
<td>Prediction bio-window flags: BiowD(n) = 1 to process bio-window n = 0 otherwise for RPTYPE = 02 and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PARMTR(2) ≠ 0 Selects by bio-window in record.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PARMTR(3) ≠ 0 Uses last yield date in record.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PARMTR(4) ≠ 0 Selects by bio-window in record.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PARMTR(5) ≠ 0 Selects by bio-window in record.</td>
</tr>
<tr>
<td>WPRTY</td>
<td>-</td>
<td>4</td>
<td>0</td>
<td>0-4</td>
<td></td>
<td>Bio-window priorities: List of bio-windows in decreasing order of priority, e.g., 4, 1, 3, 2 or 3, 1, 0, 0. Used only by RPTYPE = 02 and PARMTR(2) ≠ 0 in conjunction with the prediction dates below.</td>
</tr>
<tr>
<td>∞</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Prediction dates (up to 14 dates) for the selection of data for RPTYPE = 02 or 03,</td>
</tr>
<tr>
<td>IPRD</td>
<td>-</td>
<td>3, 14</td>
<td>0</td>
<td>year: &gt; 64</td>
<td></td>
<td>IPRD(1, n) = year - 1900</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>month: 01-12</td>
<td></td>
<td>IPRD(2, n) = month</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>day: 01-31</td>
<td></td>
<td>IPRD(3, n) = day</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>The dates must be in ascending order. The first zero date terminates the list.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>For RPTYPE = 02 and if PARMTR(2) ≠ 0 Prediction date selected as a function of priority above and latest date less than or equal to the input prediction date.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PARMTR(3) ≠ 0 Selects the latest yield date less than or equal to the input prediction date.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PARMTR(4) or (5) ≠ 0 the input prediction dates must match with file dates; otherwise a message is printed and that date skipped.</td>
</tr>
</tbody>
</table>
Figure 2-1. Data Card Formats
Figure 2-1. Data Card Formats (cont'd)
Figure 2-1. Data Card Formats (cont'd)
3.0 PROCESSING

3.1 OVERVIEW

The POUT program is essentially subdivided into four areas of processing. The area executed depends on the major type option selected on the card input. The main program POUT will read the card input and then after error-checking will transfer to the selected option. The required file(s) will be processed and the printed report(s) will be produced.
3.2 PROGRAM FLOW

Initialization of constants, set nominals

Read set of data cards

Input error checking

Output error messages → STOP

RPTYPE=01?

YES → SUBREF: Substrata historical data processed

NO → FILSEQ: Reads SUBHST file, checks ID, formats data into FILBUF and transmits data to SUBREF

SUBHST

Substrata Historical File (SUBHST)

F

Substrata reference data report created

REPORT
ANPERR Process CASF data for report

FILDA

CASF

MCHIST Process the CAS distribution output data to create Monte Carlo histograms

FILDA Passes data from direct access file to create report

C

NO

YES

REPT2 Population production error report

REPORT

REPRT2 Monte Carlo area error prod. error yield error & confid. level

REPORTS

E

CAS Distrib. Output File (CASDIS)
FREQAC
Processes the data acquisitions file data to create the freq. of sample segment acquisitions

FILSEQ
Reads, checks formats, and passes file data through FILBUF to FREQAC

REPRT3
Frequency of sample segment acquisitions

Data Acq. File (ACQUIS)

REPORT

STOP
3.3 PROCEDURES AND EQUATIONS

3.3.1 Initialization

The general initialization tasks are performed by the main program POUT.

1. Initialize flags and constants; set nominals.
2. Read data cards.
3. Process data cards for input errors.
4. Determine report type to be produced.

3.3.2 Substrata Reference Data Report

The Substrata Reference Data will be created and produced by subroutine SUBREF from the Substrata Historical File.

1. The Substrata Historical File will be accessed sequentially through the file interface subroutine FILSEQ with RPTYPE = 01 and ICASIN(I) = valid case number.

FILSEQ is a subroutine that reads a requested sequential file, checks for a match on CASE ID and returns one detail record at a time to the requesting subroutine through the FILBUF common block.

2. Process and sum the data by strata, zone, region and country for the following equations:

\( XR(K) = \frac{\text{AREA}(K)}{\text{XNA} \times \text{AREAPS}} \) where \( K \) is any substratum

\[ HWA = \sum_{I=K}^{J} \frac{XN(I) \times XR(I) \times 10289.712 \times XHPW(I)}{100} \]

\[ TWA = \sum_{I=K}^{J} \frac{XN(I) \times XR(I) \times 10289.712 \times XTPW(I)}{100} \]

\[ XINBS = \sum_{I=K}^{J} \frac{XBPW(I) \times 100}{HWA} \times \frac{XN(I) \times XR(I) \times 10289.712}{HWA} \]
(5) $XSIMBS = \frac{TWA - HWA}{HWA}$

(6) $XINCV = \left[ \sum_{I=K}^{J} (XN(I) \times XR(I) \times 10289.712 \times \frac{XTPW(I)}{100} \times XCV2(I) ) \right]^{\frac{1}{2}} \times \frac{1}{TWA}$

(7) $XSIMCV = \left[ \frac{J}{J-1} \left[ \sum_{I=K}^{J} (TWA(I) - HWA(I) )^{2} \right] \right]^{\frac{1}{2}} \times \frac{1}{TWA}$

3. Output of the printed report for each strata, zone, region and country as processed is produced by calls to subroutine REPRT1.

3.3.3 Population Mean, Standard Deviation and Histogram Reports

Population reports will be selected (by input option), created and produced by control subroutine POPDRV.

1. For $\text{PARMTR}(1) \neq 0$ the Segment Truth File along with the Substrata Historical File (SUBHST) will be accessed sequentially through subroutine FILSEQ with $\text{RPTYPE} = 02$, $\text{PARMTR}(1) \neq 0$ and $\text{ICASIN}(1) = \text{valid case number}$ to produce the Population Sampling Error Report.

2. For $\text{PARMTR}(2) \neq 0$ the CAMS Output File along with the SUBHST File will be accessed sequentially through subroutine FILSEQ with $\text{RPTYPE} = 02$, $\text{PARMTR}(2) \neq 0$ and $\text{ICASIN}(2) = \text{valid case number}$ to produce the Population CAMS Error Report.

3. For $\text{PARMTR}(3) \neq 0$ the YES Output File along with the SUBHST File will be accessed sequentially through subroutine FILSEQ with $\text{RPTYPE} = 02$, $\text{PARMTR}(3) \neq 0$ and $\text{ICASIN}(3) = \text{valid case number}$ to produce the Population Yield Error Report.

4. For $\text{PARMTR}(4) \neq 0$ or $\text{PARMTR}(5) \neq 0$ the CAS Cum Data File will be accessed through the direct access subroutine FILDA. The required input data for either of these accesses
is PARMTR(4)≠0, RTYPE=02 and ICASIN(4) = valid case number or PARMTR(5)≠0, RTYPE=02, and ICASIN(5) = valid case number.

5. Each of the five parameter reports computes and sums the same equations except for the computation of the specific error and reference values which appear below the basic equations.

(8) \[ XMVAL = \frac{100}{J*\text{REF}(J)} \sum_{I=K}^{J} \text{DELERR}(I) \]

(9) \[ \text{STDDEV} = \frac{100}{\text{REF}(J)} \left[ \frac{1}{J-1} \left\{ \sum_{I=K}^{J} (\text{DELERR}(I))^2 \right\} - \frac{1}{J} \left( \sum_{I=K}^{J} \text{DELERR}(I) \right)^2 \right]^{1/2} \]

(10) \[ XMPSUM = J*XMVAL \]

(11) \[ SDPSUM = J^{1/2} * \text{STDDEV} \]

(12) \[ \text{PCDERR} = \frac{100}{\text{REF}(J)} * \text{DELERR} \]

where each parameter type (below) for any K substratum or S stratum

a. Population Sampling Error

(13) \[ \text{DELERR}(K) = \left( \frac{1}{XM(K)} \sum_{L=1}^{XM(K)} (XTPW(L) - XTPW(K)) \right) * \text{REF}(J) * 10289.712 \]

(14) \[ \text{REF}(J) = \sum_{I=K}^{J} XTPW(I) * \text{REF}(I) * 10289.712 \]

b. Population CAMS Error

(15) \[ \text{DELERR}(K) = \left[ \frac{1}{XM(K)} \sum_{L=1}^{XM(K)} (XEPW(L) - XTPW(L)) \right] * \text{REF}(J) * 10289.712 \]

(16) \[ \text{REF}(J) = \sum_{I=K}^{J} XTPW(I) * \text{REF}(I) * 10289.712 \]
c. Population Yield Error

\[
(17) \text{DELELLR}(S) = (\text{XEYLD}(S) - \text{XTYLD}(S)) \times \sum_{I=K}^{S} \frac{\text{XTPW}(I) \times \text{XN}(I)}{10289.712} \times \text{XR}(I)
\]

\[
(18) \text{REF}(J) = \sum_{I=K}^{S} \frac{\text{XTPW}(I) \times \text{XN}(I) \times \text{XR}(I) \times 10289.712 \times \text{XTYLD}(S)}{S}
\]

d. Population Area Error

\[
(19) \text{DELERR}(S) = \frac{\text{AERRS}}{\text{XNT}}
\]

\[
(20) \text{REF}(J) = \sum_{I=S}^{S} \frac{\text{TWAS}(I)}{\text{XNT}}
\]

e. Population Production Error

\[
(21) \text{DELERR}(S) = \frac{\text{PRERRS}}{\text{XNT}}
\]

\[
(22) \text{REF}(J) = \sum_{I=S}^{S} \frac{\text{TPRODS}(I)}{\text{XNT}}
\]


3.3.4 Histograms of Monte Carlo Statistics

Monte Carlo Histograms will be selected (by input option), created and produced by subroutine MCHIST.

1. The CAS Distribution Output File will be accessed through the direct access subroutine FILDA with RPTYPE = 03, PARMTR and ICASIN values for each parameter of this type required.

2. Each of the nine reports computes and sums the equations (8), (9) and (12) of Section 3.3.3 with \( J = \text{XNT} \). Each parameter type is shown below where \( L = \text{zone (Z)}, \text{region (R) or country (C) level.} \)
a. Monte Carlo Area Error

(23) \( \text{DELERR(L)} = \text{AERRL} \)

(24) \( \text{REF} = \frac{\text{AEREFL}}{\text{XNT}} \)

b. Monte Carlo Production Error

(25) \( \text{DELERR(L)} = \text{PRERRL} \)

(26) \( \text{REF} = \frac{\text{PEREFL}}{\text{XNT}} \)

c. Monte Carlo Yield Error

(27) \( \text{DELERR(L)} = \text{YERRL} \)

(28) \( \text{REF} = \text{YEREFL} = 100 \)

d. Confidence Level (Country Level Only)

(29) \( \text{DELERR(C)} = \text{XCL} \)

where \( \text{XCL} = \text{CLEWA}, \text{CLATEC}, \text{CLATWC}, \text{CLEPRD}, \text{CLPTEC}, \text{CLPTWC} \)

(30) \( \text{REF} = \text{CLALL} = 100 \)

3. Output of the printed reports is produced by subroutine REPRT2.

3.3.5 Frequency of Sample Segment Acquisitions

The Frequency of Sample Segment Acquisitions Report will be created and produced by subroutine FREQAC.

1. The Data Acquisition File will be accessed sequentially through subroutine FILSEQ with \( \text{RPTYPE} = 04 \) and \( \text{ICASIN}(1) = \text{valid case number} \).
2. The percentage of the number of segments will be tallied by zone, region and country for the bio-window combinations:

0 1 2 3 4 1&2 1&3 1&4 2&3 2&4 3&4 1,2&3 1,2&4
1,3&4 2,3&4 1,2,3&4

3. Output of the printed report is produced by subroutine REPRT3.

3.3.6 Method of Parameter Range and Histogram Computations for 3.3.3 and 3.3.4 Above

If a set of range values is not input, then the nominal values will be used for all parameters; see Table 2-2 for these values.

As each parameter value is read from a file, the value is tallied according to the range interval it fits into. When all values have been read, the tally entries for each range are normalized by the total number of values read. The Histogram table is generated from these normalized values in the following manner.

\[ f_1, f_2, \ldots, f_{51} \] are number values in range 1, 2, \ldots, 51

Maximum of 51 range intervals are allowed.

<table>
<thead>
<tr>
<th>Value</th>
<th>Sum of Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_1 )</td>
<td>( f_1 )</td>
</tr>
<tr>
<td>( f_2 )</td>
<td>( f_1 + f_2 )</td>
</tr>
<tr>
<td>( \vdots )</td>
<td>( \vdots )</td>
</tr>
<tr>
<td>( f_{51} )</td>
<td>( f_1 + f_2 + \cdots + f_{51} )</td>
</tr>
<tr>
<td>Name</td>
<td>Symbol</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>J</td>
<td>j</td>
</tr>
<tr>
<td>K</td>
<td>K</td>
</tr>
<tr>
<td>AREA</td>
<td>NA</td>
</tr>
<tr>
<td>XNA</td>
<td>N</td>
</tr>
<tr>
<td>AREAPS</td>
<td></td>
</tr>
<tr>
<td>XR</td>
<td>R</td>
</tr>
<tr>
<td>XN</td>
<td>N</td>
</tr>
<tr>
<td>XHPW</td>
<td>~PW</td>
</tr>
<tr>
<td>HWA</td>
<td>H</td>
</tr>
<tr>
<td>HWA(K)</td>
<td>H</td>
</tr>
<tr>
<td>XTPW</td>
<td>PW</td>
</tr>
<tr>
<td>TWA</td>
<td>T</td>
</tr>
<tr>
<td>TWA(K)</td>
<td>T</td>
</tr>
<tr>
<td>XBTWP</td>
<td>δPW</td>
</tr>
<tr>
<td>XINBS</td>
<td></td>
</tr>
<tr>
<td>XSIMBS</td>
<td></td>
</tr>
<tr>
<td>XCV2</td>
<td>CV₂</td>
</tr>
<tr>
<td>Name</td>
<td>Symbol</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
</tr>
<tr>
<td>KinCV</td>
<td>KINCV</td>
</tr>
<tr>
<td>XsimCV</td>
<td>XSIMCV</td>
</tr>
<tr>
<td>DeLerr</td>
<td>ΔE</td>
</tr>
<tr>
<td>Ref</td>
<td>REF</td>
</tr>
<tr>
<td>XmVal</td>
<td>XMVAL</td>
</tr>
<tr>
<td>Stddev</td>
<td>STDDEV</td>
</tr>
<tr>
<td>XmpsSum</td>
<td>XMPUSM</td>
</tr>
<tr>
<td>SdpSum</td>
<td>SDPSUM</td>
</tr>
<tr>
<td>PcdErr</td>
<td>%ΔE</td>
</tr>
<tr>
<td>Xm</td>
<td>M</td>
</tr>
<tr>
<td>XepW</td>
<td>ð</td>
</tr>
<tr>
<td>XeYld</td>
<td>Ý</td>
</tr>
<tr>
<td>Xtyld</td>
<td>Y</td>
</tr>
<tr>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Xnt</td>
<td>N_T</td>
</tr>
<tr>
<td>A errs</td>
<td>EA_S</td>
</tr>
</tbody>
</table>
### POUT Symbol Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Symbol</th>
<th>Description</th>
<th>Source</th>
<th>Used</th>
<th>Range</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWAS</td>
<td>WAS</td>
<td>True wheat area for the $S^{th}$ stratum</td>
<td>CASF</td>
<td>20</td>
<td></td>
<td>Hectares</td>
</tr>
<tr>
<td>PRERRS</td>
<td>EPₜₜ</td>
<td>Production error</td>
<td>CASF</td>
<td>21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPRODS</td>
<td>PRDₜₜ</td>
<td>True production</td>
<td>CASF</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AERRL</td>
<td>AER</td>
<td>Area error</td>
<td>CASDIS</td>
<td>23</td>
<td></td>
<td>Quntials</td>
</tr>
<tr>
<td>AEREFL</td>
<td>AERFL</td>
<td>Reference value of area error</td>
<td>CASDIS</td>
<td>24</td>
<td></td>
<td>Hectares</td>
</tr>
<tr>
<td>PRERRL</td>
<td>RER</td>
<td>Production error</td>
<td>CASDIS</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PEREFL</td>
<td>REFL</td>
<td>Reference value of production error</td>
<td>CASDIS</td>
<td>26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YERRL</td>
<td>YER</td>
<td>Yield error</td>
<td>CASDIS</td>
<td>27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YEREFL</td>
<td>REFL</td>
<td>Reference value of yield error (= 100)</td>
<td></td>
<td>28</td>
<td></td>
<td>Hectares</td>
</tr>
<tr>
<td>CLEWA</td>
<td>CLE</td>
<td>Area confidence level Est/Est</td>
<td>CASDIS</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLATEC</td>
<td>CATE</td>
<td>Area confidence level True/Est</td>
<td>CASDIS</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLATWC</td>
<td>TWC</td>
<td>Area confidence level True/WC</td>
<td>CASDIS</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLEPRD</td>
<td>LPRD</td>
<td>Production confidence level Est/Est</td>
<td>CASDIS</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLPTEC</td>
<td>PTEC</td>
<td>Production confidence level True/Est</td>
<td>CASDIS</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLPTWC</td>
<td>PTWC</td>
<td>Production confidence level True/WC</td>
<td>CASDIS</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLALL</td>
<td>ALL</td>
<td>Reference value for all confidence levels area and production (= 100)</td>
<td>30</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.0 OUTPUT

The major product of this program is its printed reports.

4.1 PRINTED DATA

4.1.1 Reports

The type of report is determined by option on the input cards. See Tables 4.1-4.3 for format and report contents.

4.1.2 Echo Print Input Card Images

The control card inputs to POUT are printed out in the following format after all nominal values have been stored.

HEADER

AAA-------------------------AAA LPP SIMULATION PAGE 1

<table>
<thead>
<tr>
<th>RPTYPE</th>
<th>PARAMTR</th>
<th>ICASIN</th>
<th>LEVEL</th>
<th>ICSESH</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>N N N N</td>
<td>NNNN</td>
<td>N N N</td>
<td>N N N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INTERVAL</th>
<th>START</th>
<th>INTVL1</th>
<th>BREAK1</th>
<th>INTVL2</th>
<th>BREAK2</th>
<th>INTVL3</th>
<th>STOP</th>
</tr>
</thead>
<tbody>
<tr>
<td>+XXX.X</td>
<td>XXX.X</td>
<td>+XXX.X</td>
<td>XXX.X</td>
<td>+XXX.X</td>
<td>XXX.X</td>
<td>+XXX.X</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BIOWD</th>
<th>WPRTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>NNNN</td>
<td>NNNN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IPRD</th>
</tr>
</thead>
<tbody>
<tr>
<td>YY</td>
</tr>
<tr>
<td>MM</td>
</tr>
<tr>
<td>DD</td>
</tr>
<tr>
<td>YY</td>
</tr>
<tr>
<td>MM</td>
</tr>
<tr>
<td>DD</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>HEAD</th>
<th>COUNTRY</th>
<th>AREA</th>
<th>CASE NUMBER</th>
<th>N/A</th>
<th>COUNTRY REFERENCE DATA</th>
<th>LPP SIMULATION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>XXXX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>XXXX</td>
<td>XXXX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE</td>
<td>XXXX</td>
<td>XXX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE</td>
<td>XXXX</td>
<td>XXX</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 4-1:**
<table>
<thead>
<tr>
<th>Parameter Title for each type of Report</th>
<th>Population or Monte Carlo</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILE NAME</td>
<td>AAAA/AAA</td>
</tr>
<tr>
<td>(PROBABILITY) VALUE: 1 XXX.XXX TP XXX.XXX 12 XXX.XXX TP XXX.XXX 35 XXX.XXX TP XXX.XXX</td>
<td></td>
</tr>
<tr>
<td>DATA TYPE</td>
<td>AAAA/AAAAA</td>
</tr>
<tr>
<td>HEURISTIC DATA</td>
<td>1 XXX.XXX XXX.XXX XXX.XXX XXX.XXX XXX.XXX XXX.XXX XXX.XXX</td>
</tr>
<tr>
<td>N OF POPULATION</td>
<td>XXX.XXX</td>
</tr>
<tr>
<td>MINIMUM READING</td>
<td>XXX.XXX</td>
</tr>
<tr>
<td>COUNTY</td>
<td>AAA</td>
</tr>
</tbody>
</table>

* Not valid for Monte Carlo Statistics Reports

**FIGURE 4-2**
### Frequency of Sample Segment Acquisitions

<table>
<thead>
<tr>
<th>Case Number</th>
<th>File Name</th>
<th>Case Name</th>
<th>LPR Simulation</th>
<th>Page NN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### SEGMENTS

<table>
<thead>
<tr>
<th>ZONE</th>
<th>X</th>
<th>XX</th>
<th>XXX</th>
<th>XXXX</th>
<th>XXXXX</th>
<th>XXXXXX</th>
<th>XXXXXXX</th>
<th>XXXXXXXX</th>
<th>XXXXXXXXX</th>
<th>XXXXXXXXXX</th>
<th>XXXXXXXX</th>
<th>XXXXXXX</th>
<th>XXXXXXXX</th>
<th>XXXXXXXXX</th>
<th>XXXXXXXXXX</th>
<th>XXXXXXXXX</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 4-8.**
5.0 ERROR PROCESSING

5.1 INPUT ERRORS

All input errors in this program will be fatal.

1. Label and sequence checking on control cards. Message

*** IMPROPER LABEL AND SEQUENCE NUMBER ON POUT CONTROL CARD NO.____. LABEL AND SEQ. NO. = ____.

2. Range testing on RPTYPE value. Message

*** RPTYPE VALUE OUT OF RANGE. RPTYPE = ____.

3. Range testing on Histogram interval values. Messages

*** START VALUE GREATER THAN BREAK1 VALUE. 
START = ____ BREAK1 = ____.

*** BREAK1 VALUE GREATER THAN BREAK2 VALUE. 
BREAK1 = ____ BREAK2 = ____.

*** BREAK2 VALUE GREATER THAN STOP VALUE. 
BREAK2 = ____ STOP = ____.

*** THE SEGMENT BETWEEN START AND BREAK1 WILL NOT DIVIDE INTO EQUAL INTVL1 INTERVALS. 
BREAK1 - START = ___. INTVL1 = ____.

*** THE SEGMENT BETWEEN BREAK1 AND BREAK2 WILL NOT DIVIDE INTO EQUAL INTVL2 INTERVALS. 
BREAK2 - BREAK1 = ___. INTVL2 = ____.

*** THE SEGMENT BETWEEN BREAK2 AND STOP WILL NOT DIVIDE INTO EQUAL INTVL3 INTERVALS. 
STOP - BREAK2 = ___. INTVL3 = ____.

*** REQUESTED NO. OF HISTOGRAM INTERVALS EXCEEDS MAXIMUM OF 51. REQUESTED NO. OF INTERVALS THAT WOULD BE GENERATED = ____.

4. Range testing on PARMTR values. Message

*** PARMTR(i) VALUE OUT OF RANGE. PARMTR(i) = ____.
5.2 PROCESSING ERRORS

1. *** INPUT PREDICTION DATE DOES NOT MATCH DATE ON FILE.
   RPTYPE = ___. PARMTR(i) = ___. INPUT DATE = ___.

   Message is non-fatal. The input date is skipped.

2. Case number on control card does not match case number on file. Message

   *** ICASIN(i) DOES NOT MATCH CASE NUMBER ON FILE.
   ICASIN(i) = ____ FILE CASE NO. = ____ FILE NAME = ____.
PART II

COMMON BLOCK
DEFINITIONS
COMMON STORAGE ALLOCATION

Name: BWCMBN  Size: 17

Function: Holds the bio-window combinations data for printing

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSEGS</td>
<td></td>
<td>I</td>
<td>Number of segments used in the bio-window combinations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BIOCMB</td>
<td>16</td>
<td></td>
<td>Bio-window combinations</td>
<td></td>
<td>percent</td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

**Name** CARDIN  
**Size** 15

**Function** Holds the second and third input data cards except for histogram range input

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RPTYPE</td>
<td></td>
<td>I</td>
<td>Report type selection flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>01 - Substrata Reference Data</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>02 - Population Histogram</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>03 - Monte Carlo Histogram</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>04 - Freq. of Sample Seg. Acquisition</td>
</tr>
<tr>
<td>AUNITS</td>
<td></td>
<td>I</td>
<td>External print units flag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0 = English, 1 = Metric</td>
</tr>
<tr>
<td>PARMTR</td>
<td>5</td>
<td>I</td>
<td>Parameter type option flags for RPTYPE = 02 or 03</td>
</tr>
<tr>
<td>ICASIN</td>
<td>5</td>
<td>I</td>
<td>Case number identification associated with the required input file</td>
</tr>
<tr>
<td>LEVEL</td>
<td></td>
<td>I</td>
<td>Parameter report level indicators</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1 = zone, 2 = region, 3 = country</td>
</tr>
<tr>
<td>JPARM</td>
<td></td>
<td>I</td>
<td>Which PARMTR (index) being processed</td>
</tr>
<tr>
<td>IERR</td>
<td></td>
<td>I</td>
<td>Error flag for multiple report types (no match on case number, etc.)</td>
</tr>
</tbody>
</table>
## COMMON STORAGE ALLOCATION

**Name**: CONVRT  
**Size**: 6  
**Page**: 1 of 1

Contains the print units

**Function**: conversion factors

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTAMER</td>
<td>3</td>
<td>R</td>
<td>Conversion factors to convert internal metric units to external English units</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CTAMER(1) = Wheat area = 0.002471044</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CTAMER(2) = Production = 3.6743544</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CTAMER(3) = Yield = 1.4369664</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CTMTRC</td>
<td>3</td>
<td>R</td>
<td>Conversion factors to convert internal metric units to external metric units</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CTMTRC(1) = Wheat area = 0.001</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CTMTRC(2) = Production = 0.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CTMTRC(3) = Yield = 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

**Name:** FACQUS  
**Size:** 107  

Holds 1 detail record from Data Function Acquisition File, input file (ACQUIS)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIYB6</td>
<td></td>
<td>R</td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGN7</td>
<td></td>
<td>I</td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE7</td>
<td></td>
<td>I</td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRTA7</td>
<td></td>
<td>I</td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBSTA7</td>
<td></td>
<td>I</td>
<td>Substrata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEGMT7</td>
<td></td>
<td>I</td>
<td>Segment ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IWINDO</td>
<td>25, 4</td>
<td>I</td>
<td>2nd dimension = 4 crop windows</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1st dimension = up to 25 zulu dates/window in ascending order</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ITOTL</td>
<td></td>
<td>I</td>
<td>Total number of accesses</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

Name: FBLKBX  
Size: (28x18)  

Holds 1 header/detail record from CAS  
Function: cum. output file - input file (CASF)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>FBLKBX</td>
<td>28x18</td>
<td>R</td>
<td>Header record, or detail record sectioned into 18 groups</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## COMMON STORAGE ALLOCATION

**Name**: FCAMSF  
**Size**: 19  
**Holds**: 1 detail record from CAMS  
**Function**: Output File - Input file (CAMSF)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUN3</td>
<td></td>
<td>A4</td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGN3</td>
<td></td>
<td>I</td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE3</td>
<td></td>
<td>I</td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRTA3</td>
<td></td>
<td>I</td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBSTA3</td>
<td></td>
<td>I</td>
<td>Substrata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEGMT3</td>
<td></td>
<td>I</td>
<td>Segment ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPW3</td>
<td></td>
<td>R</td>
<td>True proportion of wheat for segment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EWINDO</td>
<td>3, 4</td>
<td>R</td>
<td>Bio-window values where zulu date of acquisition (integer). Est. proportion of wheat. Error in proportion of wheat estimate</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FCAMSF

EWINDO = IWIN

ZULU(I) = IWIN(1, I)

EPW(I) = EWINDO(2, I)

I = 1 to 4
**COMMON STORAGE ALLOCATION**

Name: **FCASCM**  
Size: **28**

Holds 1 header/detail group of a record

Function: from common block **FBLKBX**

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME5</td>
<td></td>
<td>A6</td>
<td>File name 'CASCUM'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICASE5</td>
<td>I</td>
<td></td>
<td>Case number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COUN5</td>
<td>A6</td>
<td></td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NT5</td>
<td>I</td>
<td></td>
<td>Current Monte Carlo iteration number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NREGS5</td>
<td>I</td>
<td></td>
<td>Number of regions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NZTOT5</td>
<td>I</td>
<td></td>
<td>Total number of zones</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NSTRT5</td>
<td>I</td>
<td></td>
<td>Total number of strata</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NBW5</td>
<td>I</td>
<td></td>
<td>Number of bio-windows</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NPD5</td>
<td>I</td>
<td></td>
<td>Number of prediction dates</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BWNBR5</td>
<td>4</td>
<td>I</td>
<td>Bio-window numbers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDNBR5</td>
<td>14</td>
<td>I</td>
<td>Prediction dates in zulu</td>
<td></td>
<td></td>
</tr>
<tr>
<td>filler</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Dimension</td>
<td>Format</td>
<td>Description</td>
<td>Symbol</td>
<td>Units</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>--------</td>
<td>------------------------------------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>REGN5</td>
<td>I</td>
<td></td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE5</td>
<td>I</td>
<td></td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRTA5</td>
<td>I</td>
<td></td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>filler</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HWA5</td>
<td>R</td>
<td></td>
<td>Historical wheat area (WA)</td>
<td>WA</td>
<td>WA</td>
</tr>
<tr>
<td>TWA5</td>
<td>R</td>
<td></td>
<td>True WA</td>
<td>WA</td>
<td>WA</td>
</tr>
<tr>
<td>EWA5</td>
<td>R</td>
<td></td>
<td>Estimated WA</td>
<td>~WA</td>
<td>WA</td>
</tr>
<tr>
<td>AERR5</td>
<td>R</td>
<td></td>
<td>Error in WA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVAR5</td>
<td>R</td>
<td></td>
<td>Variance in WA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPROD5</td>
<td>R</td>
<td></td>
<td>True production</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EPROD5</td>
<td>R</td>
<td></td>
<td>Estimated production</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRERR5</td>
<td>R</td>
<td></td>
<td>Production error</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Remainder of record is not used)
## COMMON STORAGE ALLOCATION

**Name**: FCASDS  
**Size**: 303  
**Page**: 1 of 3

Holds 1 header/detail record from CAS Dist.

**Function**: Output File - input file (CASDIS)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME6</td>
<td></td>
<td>A6</td>
<td>File name 'CASDIS'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICASE6</td>
<td></td>
<td>I</td>
<td>Case number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COUN6</td>
<td></td>
<td>A6</td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NT6</td>
<td></td>
<td>I</td>
<td>Current Monte Carlo iteration number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NREGS6</td>
<td></td>
<td>I</td>
<td>Number of regions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NZTOT6</td>
<td></td>
<td>I</td>
<td>Total number of zones</td>
<td></td>
<td></td>
</tr>
<tr>
<td>filler</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NBW6</td>
<td></td>
<td>I</td>
<td>Number of bio-windows</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NPD6</td>
<td></td>
<td>I</td>
<td>Number of prediction dates</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BWNBR6</td>
<td>4</td>
<td>I</td>
<td>Bio-window numbers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PDNBR6</td>
<td>14</td>
<td>I</td>
<td>Prediction dates insulu</td>
<td></td>
<td></td>
</tr>
<tr>
<td>filler</td>
<td></td>
<td></td>
<td>(276 words of filler)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |

| filler  |           |        |                                                  |        |       |
## COMMON STORAGE ALLOCATION

**Name**: FCASDS  
**Size**: 303  
**Function**: Holds 2 header/detail record (country records)  
**_from CAS Dist. Output File - input file (CASDIS)***

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description 1st Country Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>AEREF</td>
<td></td>
<td>R</td>
<td>Reference value for area error</td>
</tr>
<tr>
<td>PEREF</td>
<td></td>
<td>R</td>
<td>Reference value for production error</td>
</tr>
<tr>
<td>YEREF</td>
<td></td>
<td>R</td>
<td>Reference value for yield error</td>
</tr>
<tr>
<td>AERR</td>
<td>100</td>
<td>R</td>
<td>Word N+3 specifies the area error for the nth Monte Carlo iteration</td>
</tr>
<tr>
<td>PRERR</td>
<td>100</td>
<td>R</td>
<td>Word N + 103 specifies the production error for the nth Monte Carlo iteration</td>
</tr>
<tr>
<td>YERR</td>
<td>100</td>
<td>R</td>
<td>Word N + 203 specifies the yield error for the nth Monte Carlo iterations</td>
</tr>
</tbody>
</table>

**2nd Country Record**

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description 2nd Country Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLEARF</td>
<td></td>
<td>R</td>
<td>Ref. value of area confidence level Est/Est</td>
</tr>
<tr>
<td>CLEPRF</td>
<td></td>
<td>R</td>
<td>Ref. value of prod. confidence level Est/Est</td>
</tr>
<tr>
<td>CLTARF</td>
<td></td>
<td>R</td>
<td>Ref. value of area confidence level True/Est</td>
</tr>
<tr>
<td>CLEW6</td>
<td>100</td>
<td>R</td>
<td>Area confidence level Est/Est</td>
</tr>
<tr>
<td>CLEPR6</td>
<td>100</td>
<td>R</td>
<td>Production confidence level Est/Est</td>
</tr>
<tr>
<td>CLATC6</td>
<td>100</td>
<td>R</td>
<td>Area confidence level True/Est</td>
</tr>
</tbody>
</table>

---

**Notes**

- Reference values are used for error and confidence level calculations.
- Dimensions indicate the number of rows or columns in the dataset.
- Formats specify how data is stored (e.g., R for real numbers).
- Units are not specified in this table.
### COMMON STORAGE ALLOCATION

**Name**: PCASDS  
**Size**: 303

Holds 2 header/detail record (country and region or zone)

**Function**: from CAS Dist. Output File - input file (CASDIS)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLTPRF</td>
<td></td>
<td>R</td>
<td>Ref. value of prod. confidence level True/Est</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLAWCR</td>
<td></td>
<td>R</td>
<td>Ref. value of area confidence level True/ WC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLPWC</td>
<td></td>
<td>R</td>
<td>Ref. value of prod. confidence level True/ WC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLPTC6</td>
<td>100</td>
<td>R</td>
<td>Prod. confidence level True/Est</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLATW6</td>
<td>100</td>
<td>R</td>
<td>Area confidence level True/WC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GLPTW6</td>
<td>100</td>
<td>R</td>
<td>Prod. confidence level True/WC</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Region or Zone Record

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>AEREF</td>
<td></td>
<td>R</td>
<td>Ref. value for area error</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PEAREF</td>
<td></td>
<td>R</td>
<td>Ref. value for prod. error</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YEAREF</td>
<td></td>
<td>R</td>
<td>Ref. value for yield error</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AERR</td>
<td>100</td>
<td>R</td>
<td>Area error, N-Monte Carlo iterations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRERR</td>
<td>100</td>
<td>R</td>
<td>Prod. error, N-Monte Carlo iterations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>YERR</td>
<td>100</td>
<td>R</td>
<td>Yield error, N-Monte Carlo iterations</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COMMON STORAGE ALLOCATION

Name: FHEADR  Size: 4
Holds the first 4 words of each
Function: sequential file

Header Record

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>2</td>
<td>(I) A4</td>
<td>File name; 8 characters; 4 char/word</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICASEF</td>
<td>I</td>
<td></td>
<td>File case number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IMXSEG</td>
<td>I</td>
<td></td>
<td>Maximum number of segments in any substrata (SUBHST)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### COMMON STORAGE ALLOCATION

**Name**: FILBUF  
**Size**: 134

Holds file data returned by file access Function routines to the calling routine

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILENM</td>
<td>2</td>
<td>A6</td>
<td>File name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICNTRY</td>
<td>A6</td>
<td></td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IREGN</td>
<td>I</td>
<td></td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IZONE</td>
<td>I</td>
<td></td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISTRTA</td>
<td>I</td>
<td></td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISBSTA</td>
<td>I</td>
<td></td>
<td>Substrata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICASE</td>
<td>I</td>
<td></td>
<td>Case number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MXSEG</td>
<td>I</td>
<td></td>
<td>Maximum number of segments in any substrata (SUBHST only)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KBUF</td>
<td>125</td>
<td></td>
<td>Bucket area that the detail info is equivalenced into</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KBUF</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>WORD</td>
<td>SUBBRF</td>
<td>SAMERR</td>
<td>CAMSER</td>
<td>YLDERR</td>
<td>AREAER</td>
</tr>
<tr>
<td>1</td>
<td>FILENM(1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>FILENM(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>ICNTRY</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>IREGN</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>IZONE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>ISTRTA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>ISBSTA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>ICASE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>MXSEG</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>NSEG</td>
<td>NT</td>
<td>NT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>IGPNO</td>
<td>NSTRAT</td>
<td>NREGS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>XHPW</td>
<td>(filler)</td>
<td>NZTOT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>AREA</td>
<td>NBW</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>XTPW</td>
<td>NPD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>N</td>
<td>IBIOWD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>NA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>XBTPW</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>XCV2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>XTPWI</td>
<td>IPRDAT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>PRDATE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### COMMON STORAGE ALLOCATION

**Name**: FSBHST  
**Size**: 168  
**Page**: 1 of 1

**Function**: Holds one detail record from Substrata

**Historical File, input file (SUBHST)**

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUN1</td>
<td>A4</td>
<td></td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGN1</td>
<td>I</td>
<td></td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE1</td>
<td>I</td>
<td></td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRTA1</td>
<td>I</td>
<td></td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBSTA1</td>
<td>I</td>
<td></td>
<td>Substrata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INEG</td>
<td>I</td>
<td></td>
<td>List of sample segments</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NGRP</td>
<td>I</td>
<td></td>
<td>Group no.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HPW</td>
<td>R</td>
<td></td>
<td>Hist. proportion of wheat</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AREA1</td>
<td>R</td>
<td></td>
<td>Land area</td>
<td></td>
<td>kilometers²</td>
</tr>
<tr>
<td>TPW1</td>
<td>R</td>
<td></td>
<td>True proportion of wheat for substrata</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAGR</td>
<td>I</td>
<td></td>
<td>No. of agricultural segments in substrata</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NA1</td>
<td>I</td>
<td></td>
<td>No. of allocated segments in substrata</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BTPW</td>
<td>R</td>
<td></td>
<td>Bias of true proportion of wheat</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RTMP</td>
<td>R</td>
<td></td>
<td>Ratio of true mixed pixels</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV1</td>
<td>R</td>
<td></td>
<td>Coef. of variation for within county variation of PW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV2</td>
<td>R</td>
<td></td>
<td>Coef. of variation for within county variation of PW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV3</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CV4</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# COMMON STORAGE ALLOCATION

Name: FSGTRU  
Size: 16  

Function: Holds one detail record from Segment Truth File, input file (SEGTRU)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUN2</td>
<td></td>
<td>A4</td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGN2</td>
<td></td>
<td>I</td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE2</td>
<td></td>
<td>I</td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRTA2</td>
<td></td>
<td>I</td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBSTA2</td>
<td></td>
<td>I</td>
<td>Substrata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEGMT2</td>
<td></td>
<td>I</td>
<td>Segment ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ITS</td>
<td></td>
<td>DUME2</td>
<td>Dim 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ITSPRI</td>
<td>6</td>
<td>DUME2</td>
<td>Dim 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWFLG</td>
<td></td>
<td>I</td>
<td>DUME2 Dim 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPW2</td>
<td>R</td>
<td></td>
<td>True proportion of wheat for segment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TPM2</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COMMON STORAGE ALLOCATION

Name FYESOT, Size 23, Function: Holds one detail record from YES Output
File: input file (YESOUT)

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUN4</td>
<td>A4</td>
<td>Country ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGN4</td>
<td>I</td>
<td>Region ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE4</td>
<td>I</td>
<td>Zone ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRTA4</td>
<td>I</td>
<td>Stratum ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YSTR</td>
<td>R</td>
<td>True yield for stratum</td>
<td></td>
<td></td>
<td>Quintals/Hectare</td>
</tr>
<tr>
<td>IZPRDD</td>
<td>6, I</td>
<td>Zulu prediction date</td>
<td></td>
<td></td>
<td>Quintals/Hectare</td>
</tr>
<tr>
<td>YSCI</td>
<td>6, R</td>
<td>Est. yield for prediction date</td>
<td></td>
<td></td>
<td>Quintals/Hectare</td>
</tr>
<tr>
<td>YSYCI</td>
<td>6, R</td>
<td>Std. deviation of yield error</td>
<td></td>
<td></td>
<td>Quintals/Hectare</td>
</tr>
</tbody>
</table>

YSPNT = IYSPNT
IZPRDD(I) = IYSPNT(1,I), I = 1 to 4
YSCI(I) = YSPNT(2,I)
COMMON STORAGE ALLOCATION

**Name**: HSTOGM  
**Size**: 114

**Function**: Holds the Histogram data values for printing

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>XMVAL</td>
<td></td>
<td>R</td>
<td>Mean value of the error</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>STDDEV</td>
<td></td>
<td>R</td>
<td>Standard deviation of the error</td>
<td>σ</td>
<td></td>
</tr>
<tr>
<td>XMPSUM</td>
<td></td>
<td>R</td>
<td>Mean of population sum for RPTYPE = 02 only</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDPSUM</td>
<td></td>
<td>R</td>
<td>Std. deviation of population sum for RPTYPE = 02 only</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NRANGE</td>
<td></td>
<td>I</td>
<td>Number of range intervals</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IRANGE</td>
<td>51</td>
<td>I</td>
<td>Number of readings per each range interval</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NTOTL</td>
<td></td>
<td>I</td>
<td>Total number of readings</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RANGES</td>
<td>52</td>
<td>R</td>
<td>Range interval points</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>MINVAL</td>
<td></td>
<td>R</td>
<td>Minimum reading value</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>MAXVAL</td>
<td></td>
<td>R</td>
<td>Maximum reading value</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>DATPRD</td>
<td>3</td>
<td>I</td>
<td>Prediction date if RPTYPE = 02 or 03 and if 02 PARMTR(n), n = 2, 3, 4 &amp; 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NZRNG</td>
<td></td>
<td>I</td>
<td>Index of the zero range</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REF</td>
<td></td>
<td></td>
<td>Reference value</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COMMON STORAGE ALLOCATION

Name: IXRCD  
Size: 1136

Function: An array containing the index record for the direct access routine RANACF

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>IXRCD</td>
<td>1136</td>
<td></td>
<td>Subroutine RANACF index record array</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

**Name** PRNTID  **Size** 15  **Page 1 of 1**

**Function** Holds identification information for report printing

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNTRY</td>
<td></td>
<td>A4</td>
<td>Country ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REGION</td>
<td></td>
<td>I</td>
<td>Region ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZONE</td>
<td></td>
<td>I</td>
<td>Zone ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STRATA</td>
<td></td>
<td>I</td>
<td>Strata ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICASE</td>
<td></td>
<td>I</td>
<td>Case number</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILNAM</td>
<td>2</td>
<td>A6</td>
<td>For RPTYPE = 02 or 03 file name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ITOTYP</td>
<td>I</td>
<td></td>
<td>Type of information to be printed 1=strata, 2=zone, 3=region, 4=country</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAMFMT</td>
<td>I</td>
<td></td>
<td>File name format flag (REPRT2) 0 = 2A4 format, 1 = A6 format</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### COMMON STORAGE ALLOCATION

Name: PRCTRL  
Size: 4

**Function:** Print control flags

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NPAGE</td>
<td></td>
<td>I</td>
<td>Current page</td>
</tr>
<tr>
<td>NLINES</td>
<td></td>
<td>I</td>
<td>Line counter</td>
</tr>
<tr>
<td>MAXLIN</td>
<td></td>
<td>I</td>
<td>Maximum lines per page</td>
</tr>
<tr>
<td>KOUT</td>
<td></td>
<td>I</td>
<td>Print output unit number</td>
</tr>
<tr>
<td>HEADER</td>
<td>16</td>
<td>I</td>
<td>Run title output (print) area</td>
</tr>
</tbody>
</table>
### COMMON STORAGE ALLOCATION

Name: **RANGE**  
Size: **7**

Function: Holds the histogram range input values

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>START</td>
<td></td>
<td>R</td>
<td>Histogram start value</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>INTVL1</td>
<td></td>
<td>R</td>
<td>Histogram 1st interval value</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>BREAK1</td>
<td></td>
<td>R</td>
<td>Histogram breakpoint to change interval</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>INTVL2</td>
<td></td>
<td>R</td>
<td>Histogram 2nd interval value</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>BREAK2</td>
<td></td>
<td>R</td>
<td>Histogram breakpoint to change next interval</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>INTVL3</td>
<td></td>
<td>R</td>
<td>Histogram 3rd interval value</td>
<td></td>
<td>percent</td>
</tr>
<tr>
<td>STOP</td>
<td></td>
<td>R</td>
<td>Histogram terminal value</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

Name: **READNG**  
Size: **5600**

**Function:** Intermediate storage area for the "error readings" of the Population and Monte Carlo Reports. Intermediate Buffer for Confidence Levels.

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>READNG</td>
<td>5000</td>
<td>R</td>
<td>&quot;Error readings&quot; at substrata, strata, zone, region or country level</td>
</tr>
<tr>
<td>CLEWA</td>
<td>100</td>
<td>R</td>
<td>Area confidence level</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Est/Est</td>
</tr>
<tr>
<td>CLEPRD</td>
<td>100</td>
<td>R</td>
<td>Production confidence level</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Est/Est</td>
</tr>
<tr>
<td>CLATEC</td>
<td>100</td>
<td>R</td>
<td>Area confidence level</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>True/Est</td>
</tr>
<tr>
<td>CLPTEC</td>
<td>100</td>
<td>R</td>
<td>Production confidence level</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>True/Est</td>
</tr>
<tr>
<td>CLATWC</td>
<td>100</td>
<td>R</td>
<td>Area confidence level</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>True/WC</td>
</tr>
<tr>
<td>CLPTWC</td>
<td>100</td>
<td>R</td>
<td>Production confidence level</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>True/WC</td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

Name: SELCTN  Size: 50

Function: Holds the input bio-window and prediction date selection criteria for RPTYPE = 02 or 03

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIOWD</td>
<td>4</td>
<td>I</td>
<td>Bio-windows to be processed flags</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WPRTY</td>
<td>4</td>
<td>I</td>
<td>Bio-window order of priority</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPRD</td>
<td>3, 14</td>
<td>I</td>
<td>Prediction dates to be processed</td>
<td></td>
<td>year, month, day</td>
</tr>
</tbody>
</table>
**COMMON STORAGE ALLOCATION**

**Name**: SUBVAL  
**Size**: 11  
**Function**: Holds the Substrata Historical Data values for printing

<table>
<thead>
<tr>
<th>Name</th>
<th>Dimension</th>
<th>Format</th>
<th>Description</th>
<th>Symbol</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>HWA</td>
<td></td>
<td>R</td>
<td>Historical wheat area</td>
<td>WA</td>
<td>Hectares</td>
</tr>
<tr>
<td>TWA</td>
<td></td>
<td>R</td>
<td>True wheat area</td>
<td>WA</td>
<td>Hectares</td>
</tr>
<tr>
<td>NSURGP</td>
<td>3</td>
<td>I</td>
<td>Number in each substrata group type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NSEGPP</td>
<td>2</td>
<td>I</td>
<td>Number of segments in groups 1 and 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XINBS</td>
<td></td>
<td>R</td>
<td>Input bias</td>
<td></td>
<td>% Hist.</td>
</tr>
<tr>
<td>XSIMBS</td>
<td></td>
<td>R</td>
<td>Simulation bias</td>
<td></td>
<td>% Hist.</td>
</tr>
<tr>
<td>XINCV</td>
<td></td>
<td>R</td>
<td>Input coefficient of variation</td>
<td></td>
<td>% True</td>
</tr>
<tr>
<td>XSIMCV</td>
<td></td>
<td>R</td>
<td>Simulation coefficient of variation</td>
<td></td>
<td>% True</td>
</tr>
</tbody>
</table>
PART III

LIST OF SUBROUTINES AND SUBROUTINE CALL STRUCTURE
<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>POUT</td>
<td>Main control subroutine for POUT program.</td>
</tr>
<tr>
<td>SUBREF</td>
<td>This subroutine processes the substrata reference data report - RPTYPE = 1.</td>
</tr>
<tr>
<td>POPDRV</td>
<td>This subroutine controls the generation of the population reports - RPTYPE = 2.</td>
</tr>
<tr>
<td>SAMERR</td>
<td>This subroutine produces the population segment sampling error report - RPTYPE = 2, PARMTR = 1.</td>
</tr>
<tr>
<td>CAMSER</td>
<td>This subroutine produces the population CAMS error report - RPTYPE = 2, PARMTR = 2.</td>
</tr>
<tr>
<td>FREQAC</td>
<td>This subroutine produces the frequency of sample segment acquisitions report - RPTYPE = 4.</td>
</tr>
<tr>
<td>YLDERR</td>
<td>This subroutine produces the population yield error report - RPTYPE = 2, PARMTR = 3.</td>
</tr>
<tr>
<td>REPRT1</td>
<td>This subroutine prints the substrata reference data report.</td>
</tr>
<tr>
<td>REPRT2</td>
<td>This subroutine prints the histogram data reports.</td>
</tr>
<tr>
<td>REPRT3</td>
<td>This subroutine prints the frequency of sample segments acquisitions.</td>
</tr>
<tr>
<td>PAGHDR</td>
<td>This routine controls count on printed output and provides a line of printed heading on top of each output page.</td>
</tr>
<tr>
<td>GENRNG</td>
<td>This subroutine generates the histogram ranges from the interval input.</td>
</tr>
<tr>
<td>FILSEG</td>
<td>This subroutine reads the required sequential files determined by the RPTYPE flag and returns the required data in common block FILBUF.</td>
</tr>
<tr>
<td>CLMOVE</td>
<td>This routine (used by MCHIST) moves the confidence levels, one at a time, from the reading buffer to FILBUF.</td>
</tr>
<tr>
<td>PUNITS</td>
<td>This routine sets up the proper print and conversion units for the population and Monte Carlo reports.</td>
</tr>
<tr>
<td>MCHIST</td>
<td>This subroutine produces the Monte Carlo reports - RPTYPE = 3.</td>
</tr>
<tr>
<td>ANPERR</td>
<td>This subroutine produces the population area error report or the population production error report - RPTYPE = 2, PARMTR = 4, 5.</td>
</tr>
<tr>
<td>Name</td>
<td>Function</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>FILDA</td>
<td>This subroutine controls the reading of the CASF file or CASDIS file.</td>
</tr>
<tr>
<td>RANACF</td>
<td>This routine is a generalized random access I/O routine (utilizing define file on the Univac).</td>
</tr>
</tbody>
</table>
POUT Subroutine Call Structure

POUT
  PAGHDR
  SUBREF
    FILSEQ
      PAGHDR
      REPRT1
      PAGHDR
  POPDRV
    GENRNG
    SAMERR
      FILSEQ
      REPRT2
        PUNIT5
        PAGHDR
    CAMSER
      FILSEQ
      LFPA
      FZULU
      REPRT2
      PAGHDR
    YLDERR
      FILSEQ
      LFPA
      PAGHDR
      FZULU
      REPRT2
    ANPERR
      FILDA
      PAGHDR
      RANACF
      LFPA
      REPRT2
MCHIST
FILDA
PAGHDR
GENRNG
LFPA
CLMOVE
FZULU
REPT2
FREQAC
FILSEQ
REPT3
PAGHDR
PART IV

SUBROUTINE DESCRIPTIONS AND FLOWCHARTS
Program Main Routine POUT

Purpose:

This is the driver for the Post Processor (POUT) program. The driver reads in the control card input and error check all values except the file case numbers (ICASIN(i)). All card input is checked before any error will cause processing to terminate. All card input is Echo printed and all errors are noted with an appropriate error message. The routine then calls the requested RPTYPE report subroutine.

Input:

/CARDIN/  IHEADR
          RPTYPE
          AUNITS
          PARMTR (5)
          ICASIN (5)
          LEVEL

/RANGE/   START
          INTVL1
          BREAK1
          INTVL2
          BREAK2
          INTVL3
          STOP

/SELCNT/  BIOWD (4)
          WPRTY (4)
          IPRD (3, 14)

Output:

/CARDIN/
/RANGE/
/SELCNT/
/PRTCTRL/  NPAGE
           NLINES
           MAXLIN
           KOUT
           HEADER (16)

/CONVRT/  CTAMER (3)
          CTMTRC (3)
Linkage:

Main program.

Subroutine Used:

CALL SUBREF  process and produce Substrata Reference Data Report if RPTYPE = 01.
CALL POPDRV   process and produce Population Reports if RPTYPE = 02.
CALL MCHIST   process and produce Monte Carlo Reports if RPTYPE = 03.
CALL FREQAC   process and produce Frequency of Sample Segments Acquisitions Report.
CALL PAGHDR   controls count on printed output and provides a line of printed heading on the top of each output page with page number.

Local Variables:

NOMRNG = 0
    = 1.  Histogram range values input
    Nominal ranges to be used
RNGNML =  Histogram nominal range values
AMCVT  =  American conversion factors
XMCVT  =  Metric conversion factors
IN = 5 Standard read (input) unit
IOUT = 6 Standard write (output) unit
LNMAX = 45 Maximum no. of print lines per page
LABEL = "POUT" Card label identifier
NCARD = "01, 02, etc." Card label numbers
MRPTYP = 4 Maximum RPTYPE value
XMXRGE = 51 Maximum no. of histogram ranges
INERRS  Logical flag indicating that an input error has occurred. At the completion of input this flag is tested for run termination due error(s).
Processing:

1. Zero out arrays and preset constants.

2. Read Header Card
   Save header
   Check card label and sequence no.
   Print error message and set INERRS = true

3. Read Data Card 1
   Check card label and sequence no.
   Check RPTYPE in range
   Check RANGE values for nominal usage
   Check input RANGE values for interval or size errors

4. Read Data Card 2
   Check card label and sequence
   Check PARMTR values and if set on ICASIN values
   Check LEVEL value if RPTYPE = 02; use nominal of 1 if = 0

5. Read Data Cards 4 and 5
   Check card labels and sequences
   Check only if RPTYPE = 02 or 03

6. Echo print all input values with all nominals set where necessary.

7. Terminate run if INERRS = true.

8. Run case according to RPTYPE value.

9. Stop.
Subroutine SUBREF

Purpose:

This routine produces the Substrata Reference Data Report. It accesses the Substrata Historical File (SUBHST) through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT1 for printing.

Input:

/FILBUF/ ICNTRY IREGN IZONE ISTRTA ICASE NSEG NSEG GPNO XHPW AREA XTPW N NSEG NA XBTPW XCV2

Output:

/PRNTID/ CNTRY REGION ZONE STRATA ICASE ITOTYP

/SUBVAL/ HWA TWA NSUBGP (3) NSEG GP (2) XINBS XSIMBS XINCV XSIMCV

Linkage:

CALL SUBREF called by POUT.
Subroutines Used:

CALL FILSEQ (IH) reads data file SUBHST sequentially.
IH = 1 read header record.
IH = 0 read detail record.
Return of IH < 0 end of file on file being read.

CALL REPRT1 prints the Substrata Reference Data Report.

Local Variable Description:

AREAPS = 10289.712, area per segment in Hectares.
XNA Flt. pt., no. of allocated segments.
XR Ratio of the true substratum land area to the gross-
  pseudo substratum area.
XN Flt. pt., no. of agricultural segments in substrata.
HWAK Historical wheat area, substrata level.
HWAS Historical wheat area, strata level.
HWAZ Historical wheat area, zone level.
HWAR Historical wheat area, region level.
HWAC Historical wheat area, country level.
TWAK True wheat area, substrata level.
TWAS True wheat area, strata level.
TWAZ True wheat area, zone level.
TWAR True wheat area, region level.
TWAC True wheat area, country level.
XNR XN * XR * 10289.72, substrata level.
J No. of substrata in a stratum.
XINBSS Input bias, strata level.
XINBSZ Input bias, zone level.
XINBSR Input bias, region level.
XINBSC Input bias, country level.
XSMBSZ Simulation bias, zone level.
XSMBSR Simulation bias, region level.
XSMBSC Simulation bias, country level.
XINCVS Input coefficient of variation, strata level.
XINCVZ Input coefficient of variation, zone level.
XINCVR Input coefficient of variation, region level.
XINCVC Input coefficient of variation, country level.
XSMCVS(2) Simulation coefficient of variation, strata level.
XSMCVZ Simulation coefficient of variation, zone level.
XSMCVR Simulation coefficient of variation, region level.
XSMCVC Simulation coefficient of variation, country level.
IH Flag to subroutine FILSEQ where IH = 1 read header
  record or IH = 0 read detail record.
KSBSTA Substrata counter.
Processing:

1. \( IH = 1 \), call FILSEQ to read header record.

2. Verification done by FILSEQ.

3. SAVE ICASE in common PRNTID.

4. SET KSBSTA = 0 and \( J = 0 \) and zero out accumulative variables.
   Call FILSEQ to read detail record.
   
   \[
   KSBSTA = KSBSTA + 1
   \]
   IF (KSBSTA, EQ 1) SAVE COUNTRY ID
   REGION ID
   ZONE ID
   STRATA ID
   in PRNTID

5. Test for a change in identification area
   
   STRATA
   ZONE
   REGION

6. If same, perform calculations

   \( XNA = NA \)
   \( XN = N \)
   \( XR = AREA/(XNA * AREAPS) \)
   \( XNR = XN * XR * AREAPS \)
   \( HWAK = XNR * XHPW/100 \)
   \( HWAS = HWAS + HWAK \)
   \( TWA = XNR * XTPW/100 \)
   \( TWAS = TWAS + TWA \)
   \( NSUBGP (IGPNO) = NSUBGP (IGPNO) + 1 \)
   IF (IGPNO, EQ. 1 or 2)
   \( NSEGPG (IGPNO) = NSEGPG (IGPNO) + NSEG \)
   \( XINBSS = XBTPW * HWAK + XINBSS \)
   \( XINCVS = (XCV2 * TWA) **2 + XINCNS \)
   \( XSMCV = TWA - HWAK \)
   \( XSMCVS(1) = XSMCVS(1) + XSMCV **2 \)
   \( XSMCVS(2) = XSMCV(2) + XSMCV \)
   \( J = J + 1 \)
   Read another detail record and go to 5.

7. Change in strata ID

   \( HWA' = HWAS \)
   \( TWA' = TWAS \)
   \( XINBS = XINBSS/HWA \)
   \( XSIMBS = (TWA' - HWA')/HWA \)
   \( XINCV = (XINCVS)^{1/2}/TWA \)
   \( XSMCV = (J/(J - 1) * (XSMCVS(1) - 1/J * XSMCVS(2)^2))^{1/2} \)
\text{NSUBGP}(i) = \text{NSBGP}(i)
\text{NSEGGP}(j) = \text{NSGGP}(j) \quad \text{where} \quad i = 1, 2 \& 3; \quad j = 1 \& 2

Call 
\text{REPRT1} \text{ with } ITOTYP = 1

\text{HWAZ} = \text{HWAZ} + \text{HWA}
\text{TWA} = \text{TWAZ} + \text{TWA}
\text{XINBSZ} = \text{XINBSZ} + \text{XINBS}
\text{XSMBSZ} = \text{XSMBSZ} + \text{XSIMBS}
\text{XINCVZ} = \text{XINCVZ} + \text{XINCV}
\text{XSMCVZ} = \text{XSMCVZ} + \text{XSMCV}
\text{NSBGPZ}(i) = \text{NSBGP}(i) + \text{NSUBGP}(i)
\text{NSGGPZ}(j) = \text{NSGGP}(j) + \text{NSEGGP}(j)

\text{HWAS} = 0
\text{TWAS} = 0
\text{XINBS} = 0
\text{XINCVS} = 0
\text{XSMCVS}(1) = 0
\text{XSMCVS}(2) = 0
\text{J} = 0
\text{NSBGP}(i)
\text{NSGGP}(j)

\text{If no change in zone ID, read another detail record and go to 5.}

8. \text{Change in zone ID}

\text{HWA} = \text{HWAZ}
\text{TWA} = \text{TWAZ}
\text{XINBS} = \text{XINBSZ}
\text{XSIMBS} = \text{XSMBSZ}
\text{XINCV} = \text{XINCVZ}
\text{XSIMCV} = \text{XSMCVZ}
\text{NSUBGP}(i) = \text{NSBGP}(i)
\text{NSEGGP}(j) = \text{NSGGP}(j)

Call 
\text{REPRT1} \text{ with } ITOTYP = 2

\text{HWAR} = \text{HWAR} + \text{HWA}
\text{TWAR} = \text{TWAR} + \text{TWA}
\text{XINBSR} = \text{XINBSR} + \text{XINBS}
\text{XSMBSR} = \text{XSMBSR} + \text{XSMBS}
\text{XINCVR} = \text{XINCVR} + \text{XINCV}
\text{XSIMCVR} = \text{XSIMCVR} + \text{XSIMCV}
\text{NSBGPZ}(i) = \text{NSBGPZ}(i) + \text{NSUBGP}(i)
\text{NSGGPZ}(j) = \text{NSGGPZ}(j) + \text{NSEGGP}(j)

\text{HWAZ} = 0
\text{TWA} = 0
\text{XINBS} = 0
\text{XSMBS} = 0
\text{XINCVZ} = 0
\text{XSMCVZ} = 0
\text{NSBGPZ}(i) = 0
\text{NSGGPZ}(j) = 0

\text{If no change in region ID, read another detail record and go to 5.}
9. Change in region ID
   Move region values to common variables as in 8 above.
   CALL REPR T1 with ITOTYP = 3.
   Sum region values into country values.
   Zero out region values.
   If more data to be read, read another detail record and go to 5.

10. All records have been processed.
    Move country values to common variables as in 8 above.
    CALL REPR T1 with ITOTYP = 4

11. Return to POUT.
Subroutine POPDRV

Purpose:

This routine is the driver routine for the generation of the Population Reports (RPTYPE = 02).

Input:

/CARDIN/ RPTYPE
PARMTR(5)
IERR

Output:

/CARDIN/ JPARM
IERR Reset to zero each subroutine call

Linkage:

CALL POPDRV called by POUT

Subroutines Used:

CALL SAMERR Generates the Population Sampling Error Report (PARMTR(1) ≠ 0)
CALL CAMSER Generates the Population CAMS Error Report (PARMTR(2) ≠ 0)
CALL YLDERR Generates the Population Yield Error Report (PARMTR(3) ≠ 0)
CALL ANPERR
CALL GENRNG Generates ranges for histogram

Local Variable:

JERR

Processing:

1. CALL GENRNG generates the range intervals for the histogram data.
2. IF (PARMTR(1) ≠ 0) set JPARM = 1
   CALL SAMERR
3. IF PARMTR(2) ≠ 0
   set JPARM = 2
   CALL CAMSER

4. IF PARMTR(3) ≠ 0
   set JPARM = 3
   CALL YLDERR

5. IF PARMTR(4) ≠ 0
   set JPARM = 4
   CALL ANPERR

6. IF PARMTR(5) ≠ 0
   set JPARM = 5
   CALL ANPERR

7. Return.
Subroutine SAMERR

Purpose:

This routine produces the Population Sampling Error Report. It accesses the SUBHST and Segment Truth (SEGTRU) Files through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT2 for printing.

Input:

/CARDIN/ LEVEL
/HSTOGM/ NRANGE RANGE (52) NZRNG
/FILBUF/ FILENM (2) ICNTRY IREGN IZONE ISTRTA ISBSTA ICASE
NSEG Number of sample segments in substrata
AREA Land area of substrata
XTPW True proportion of wheat, Kth substratum
N No. of agricultural segments in substrata
NA No. of allocated segments in substrata
XTPWI True proportion of wheat ith segment

Output:

/PRNTID/ CNTRY REGION ZONE STRATA ICASEN FILNAM (2) ITOTYP NAMFMT

/READNG/ READNG (3200)
/HSTOGM/ XMVAL
STDDEV
XMPSUM
SDPSUM
NRANGE
IRANGE (51)
NTOTL
RANGES (52)
MINVAL
MAXVAL
REF
NZRNG

Linkage:

CALL SAMERR called by POPDRV

Subroutines Used:

CALL FILSEQ (IH) reads data files SUBHST and SEGTRU.
IH - header/detail record flag
  = 1 read header records
  = 0 read detail records

CALL REPRT2 prints the Population Sampling Error Report.

Local Variable Description:

IH Flag passed to routine FILSEQ whether IH = 1, read
header records; IH = 0, read detail records.

AREAPS
XNA
XR
XN
XM
J Substrata counter at zone, region or country level
DELSB Sum of proportion of wheat at substrata level
DELERR
DELSQ Square of DELERR
KXM No. of segments in substrata

Processing:

1. Initialize arrays, constants, and work areas.

2. Get header record.
   Check if IERR = 0 - no match on case ID.
   Save case number and file name.

3. Get detail records.
   Check for EOF - go to 8.
4. First time through initialize AREA ID.

5. If not first time through check for change in substrata ID and go sum segments (8.).

6. Accumulate true proportion wheat values.

7. Read another detail record.

8. Compute the sum for the substrata.

9. Save the reading in READNG.

10. Find minimum and maximum reading.

11. Add reading to sums.

12. Test level complete - otherwise go to 3.

13. If all readings processed for level compute the mean, standard deviation, etc. Tally the percentage of reading against histogram ranges.


15. Return.
Subroutine CAMSER

Purpose:

This routine produces the Population CAMS Error Report. It accesses the SUBHST and CAMS Output (CAMSF) Files through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT2 for printing.

Input:

/CARDIN/  LEVEL
/HSTOGM/  NRANGE
          RANGES
          NZRNG
/FILBUF/  FILENM (2)
          ICNTRY
          IREGN
          IZONE
          ISTRTA
          ISBSTA
          ICASE
          NSEG
          AREA
          XTPW
          N
          NA
          XTPWI
          PRDATE (4)
          XEPW (4)

/SELCTN/  BIOWD (4)
          WPRTY (4)
          IPRD (3, 14)

/PRTCTL/  KOUT

Output:

/PRNTID/  CNTRY
          REGION
          ZONE
          STRATA
          ICASEN
          FILNAM (2)

No. of sample seg. in substrata
Land area of substrata
True proportion of wheat, Kth substratum
No. of agricultural seg. in substrata
No. of allocated seg. in substrata
True proportion of wheat, Ith segment
Prediction dates of the four bio-windows
Est. proportion of wheat of the four bio-windows
/HSTOGM/  XMVAL  
      STDDEV  
      XMPSUM  
      SDPSUM  
      NRANGE  
      IRANGE (51)  
      NTOTL  
      RANGES (52)  
      MINVAL  
      MAXVAL  
      PRDATE (3)  
      NZRNG  
      REF

/READNG/  
Linkage:  
CALL CAMSER  called by POPDRV  
Subroutines Used:  
CALL FILSEQ (IH)  reads data files SUBHST and CAMSF.  
      IH = 1  read header records  
      = 0  read detail records  
CALL FZULU (IDATE, IOUT)  converts Zulu date to calendar date.  
      IDATE = Zulu date, integer  
      IOUT(1) = year, integer  
      IOUT(2) = month, integer  
      IOUT(3) = day, integer  
CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS)  converts calendar date to Zulu.  
      FLDA = day, flt. pt.  
      LMO = month, integer  
      LYR = year, integer  
      ALFGM = dummy  
      DAYS = Zulu date, flt. pt.  
CALL REPRT2  prints the Population CAMS Error Report.  
CALL PAGHDR  

Local Variable Description:  
IH  
AREAPS  
XNA  
XR
Processing:

1. Initialize arrays, constants and work areas.
2. Get bio-window number or prediction date.
3. Get header record.
   Verify if case number match file.
4. Get detail records.
   Check for EOF, go to 12.
5. Set ID area if first time through.
6. If not first time through check for change in substrata ID;
   if so, go to 12.
7. Accumulate est. proportion wheat for bio-window.
9. Get matching prediction date values if prediction date convert
to Zulu, Check against file dates - if no match, skip to next
   prediction point, go to 2.
10. Accumulate information for prediction date.
11. Read another detail for prediction date.
12. Compute sum of substrata.
13. Save the reading.
14. Find minimum and maximum reading.
15. Add reading to sums.
16. Test level complete - otherwise go to 4.
17. If complete, process all values for level. Compute mean, standard
deviation, etc. Tally readings against histogram ranges.
18. Call REPRT2.
19. Go get another prediction point.
20. If done, return.
Subroutine FREQAC

Purpose:
This routine produces the Frequency of Sample Segment Acquisition Report by scanning the Data Acquisition File (ACQUIS) and tallying the combination of accesses for each segment.

Input:
/FILBUF/ FILEN (2)
ICNTRY
IREGN
IZONE
ISTRTA
ICASE
KWIN (4)

Output:
/PRNTID/ CNTRY
REGION
ZONE
STRATA
ICASEN
FILNAM (2)
ITOTYP

/BWCMBN/ NSEGS
BIOCMB (16)

Linkage:
CALL FREQAC called by POUT

Subroutines Used:
CALL FILSEQ
REPT3

Local Variable Description:
KNTBIO'(16,3) (I, J) I - bio-window count
J - zone, region, country level
counts of segment acquisitions
KNT (3) Total no. of segments (readings) at each level
(zone, etc.)
KBIOS (16) ≡ KNTBIO(I, 1)
KBIOR (16) ≡ KNTBIO(I, 2)
KBIOS (16) ≡ KNTBIO(I, 3)
IFIRST  First time flag = 0 - first time 
      ≠ not first time
IH
KNTZ  ≡ KNT(1) -
KNTR  ≡ KNT(2) -
KNTC  ≡ KNT(3) -

Processing:

1. Initialize storage arrays.
2. Get header record and save ID information.
   If EOF, go to 8.
4. If first time through save ID (area) information.
5. If not first time through, check for change in ID area and 
   go to 8.
6. Test bio-windows for accesses and tally the combinations.
7. Go to 3 to get another detail record.
8. Compute percentage of combinations at zone level and print 
   that zone.
9. Add zone values to region. If change in zero, compute per-
   centage and print.
10. Add region to country values. If EOF, compute percentages 
    and print.
11. When completed, return.
Subroutine YLDERR

Purpose:

This routine produces the Population Yield Error Report. It accesses the SUBHST and YES Output (YESOUT) Files through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT2 for printing.

Input:

/CARDIN/ LEVEL
/HSTOGM/ NRANGE RANGES NZRNG
/FILBUF/ FILENM (2) ICNTRY IREGN IZONE ICASE AREA XTPW N NA PRDATE (6) XEYLD (6) XTYLD
/SELCNT/ BIOWD (4) IPRD (3, 14)
/PRTCTL/ KOUT

Output:

/PRNTID/ CNTRY REGION ZONE ICASEN FILNAM (2) NAMFMT
/HSTOGM/  XMVAL
       STDDEV
       XMPSUM
       SDPSUM
       NTOTL
       MINVAL
       MAXVAL
       PRDATE (3)
       REF

/READNG/  READNG (500)

Linkage:

CALL YLDERR  called by POPDRV

Subroutines Used:

CALL FILSEQ  reads data files SUBHST and YESOUT
CALL FZULU (IDATE, IOUT)
CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS)
CALL REPRT2  prints yield error
CALL PAGHDR

Local Variable Description:

IH
AREAPS
XNA
XN
XNR
J
KXS
DELSTA
REFSTA
DELERR
DELSQ

Processing:

1. Initialize arrays, constants, etc.
2. Get bio-window number or prediction date.
3. Get header record,
   Verify match on case number.
4. Get detail records.
   Check EOF, go to 10.
5. Set ID area if first time through.
6. If not first time through check for change in level ID.
7. Determine prediction point processing.
8. Accumulate information.
9. Read another detail record.
10. Compute sum of strata.
11. Save reading.
12. Find minimum and maximum readings.
13. Add reading to sums.
14. Test level complete, otherwise go to 4.
15. If complete, process all values for level. Compute mean,
    standard deviation, etc. Tally readings against ranges.
17. Get another prediction point.
18. Otherwise, return.
Subroutine REPRT1

Purpose:
Control and print the Substrata Reference Data Report.

Input:
/CARDIN/  AUNITS
/PRTCTL/  NPAGE
           NLINES
           MAXLIN
           KOUT
           HEADER
/PRNTID/  CNTRY
           REGION
           ZONE
           STRATA
           ICASEN
/SUBVAL/  HWA
           TWA
           NSUBGP (3)
           NSEGGP (2)
           XINBS
           XSIMBS
           XINCV
           XSSIMCV
           ITOTYP
/CONVRT/  CTAMER (3)
           CTMTRC (3)

Output:
The printed report.

Linkage:
CALL REPRT1  called from SUBREF

Subroutines Used:
None.

Local Variable Description:
Subroutine REPRT2

Purpose:

Subroutine REPRT2 produces the printed reports for the Population and Monte Carlo Reports.

The first time through for any run, the histogram ranges are printed for all parameters. If the confidence levels are requested as part of a larger case, they have their own ranges. The ranges are printed only once just before the confidence level reports.

For each report, the histogram values are computed from the number of values in each range. The percentages are summed at each range point. These percentages are then printed as the histogram data followed by its mean, standard deviation, reference value, minimum and maximum readings, and number of readings. For each population report the mean and standard deviation of the population sum are also printed.

Finally, the identification line containing country, region, zone and prediction point is printed.
Subroutine REPRT3

Purpose:

Subroutine REPRT3 produces the printed report of the Frequency of Sample Segments Acquisitions.

Input:

- /PRNTID/  CNTRY
- REGION
- ZONE
- ICASEN
- FILNAM (2)
- ITOTYP
- /BWCMBN/  NSEGS
- BIOCMCB (16)
- /PRTCTL/  NPAGE

Output:

The printed report.

Linkage:

CALL REPRT3

Subroutines Used:

CALL PAGHDR (N) controls the line count on the printed output.

Local Variable Descriptors:

- MPAGE  checked against NPAGE to determine if the page number has been incremented and if so, prints the report headings.
- JFIRST  first time flag.

Processing:

1. Print page headings if JFIRST = 0.
2. Go to 3. if ITOTYP = 1
   5. if ITOTYP = 2
   7. if ITOTYP = 3
3. Print report headings if NPAGE ≠ MPAGE.
4. Print the zone level values for the 16 combinations. Go to 9.
5. Print report headings if NPAGE ≠ MPAGE.
6. Print the region level values for the 16 combinations. Go to 9.
7. Print report headings if NPAGE ≠ MPAGE.
8. Print the country level values for the 16 combinations.
9. Return.
Subroutine PAGHDR (N)

Purpose:
Subroutine PAGHDR controls the line count on printed output and provides a line of printed heading at the top of each output page including the page number.

Input:
/PRTCTL/  NPAGE
          NLINES
          MAXLIN
          KOUT
          HEADER (16)

Linkage:
CALL PAGHDR (N)
where  N = count of lines to be printed
      -N = skip to a new page before printing N lines

Local Variable Descriptions:
LPRINT = a flag which indicates to the routine what the previous command was
        = 1 increment the line count
        = 0 skip to the bottom of page
        = -1 the first time the routine called, always skip to new page

NLINES = the current starting line number of a page

Processing:
1. Test LPRINT value for previous command.
2. LPRINT less than zero; set constants N1 and N2 and reset LPRINT = 1. Go to 11.
3. LPRINT equals zero; set LPRINT = 1. Go to 11.
4. Test N for less than or equal zero. If yes, go to 11.
5. LPRINT greater than zero. Compute J = NLINES+N.
6. Test J if greater than N1 constant and if so, go to 8.
7. Otherwise set NLINES = J; go to 13.

8. If N not equal zero, go to 13.

9. Move line printer to bottom of page by printing an appropriate number of blank lines.

10. Set LPRINT = 0. Go to

11. Increment page number and print header at the top of the new page.

12. Reset NLINES value.

13. Return to calling routine.
Subroutine GENRNG (ICL)

Purpose:

Subroutine GENRNG generates the histogram ranges from the input interval or nominal interval values except for the confidence levels of RPTYPE = 03 which have a special set of intervals.

Input:

/RANGE/  START
         INTVL1
         BREAK1
         INTVL2
         BREAK2
         INTVL3
         STOP

Output:

/HSTOGM/  NRANGE
          RANGES (52)
          NZRNG

Linkage:

CALL GENRNG (ICL) where

ICL = 0  generate standard ranges for histogram data
ICL ≠ 0  generate special ranges for the confidence levels; that is, no RNGMIN or RNGMAX entries.

Local Variable Descriptions:

K    is count of the number of range values during generation.
N1   is the number of intervals between any two breakpoint values.
VALU is the actual range value computed from the initial breakpoint value plus the interval increment.
RNGMIN, RNGMAX very large minimum and maximum extreme points
Processing:

1. Zero all range values.
2. Set NRANGE and NZRNG to zero.
4. If ICL = 0 then RANGES(K) = RNGMIN.
5. If ICL ≠ 0 then K = 0.
7. Generate the N1 RANGES values continually searching for the special case of zero.
8. Add N1 to NRANGE value.
9. If INTVL2 value ≠ 0, go to 11.
10. Set RANGES(K+1) = BREAK1. Go to 18.
11. Compute N1 for next set of ranges.
12. Repeat the process for generation of the RANGES for the next set.
13. Add N1 to NRANGE value.
14. If INTVL3 value ≠ 0, go to 16.
15. Set RANGES(K+1) = BREAK2. Go to 18.
16. Compute N1 for last set of ranges.
17. Repeat the generation process of the RANGES for the final set.
18. If ICL ≠ 0, go to 21.
19. Set RANGES(K+2) = RNGMAX.
20. Add 2 to NRANGE.
Subroutine FILSEQ

Purpose:

Reads the required sequential file determined by RPTYPE and returns the required data in FILBUF common block.

Input:

/CARDIN/  RPTYPE
          PARMTR (5)
          ICASIN (5)

/FSBHST/

/FSGTRU/

/FCAMSF/

/FYESOT/

/FACQUS/

/PRTCTL/

Output:

/FILBUF/  FILNAM (2)
          ICNTRY
          IREGN
          IZONE
          ISTRTA
          ISBSTA
          ICASE
          KBUF (100)  see individual routines for which quantities used

Linkage:

CALL FILSEQ (IH)

Inputs:

IH = 1  Reads header record
IH = 0  Reads detail record(s)

Output:

IH = -1  EOF indicator

Local Variable Description:
Subroutine CLMOVE (NCL)

Purpose:

Subroutine CLMOVE moves one set of confidence levels for RPTYPE = 03 and JPARM = 4 from the /READNG/ buffer to /FILBUF/.

Input:

/READNG/   CLEWA (100)
           CLEPRD (100)
           CLATEC (100)
           CLPTEC (100)
           CLATWC (100)
           CLPTWC (100)

Output:

/FILBUF/   MCREF  confidence level reference value = 100.
            MCERR  confidence level values.

Linkage:

CALL CLMOVE (NCL) where

NCL = 1  Area confidence level est/est
       = 2  Production confidence level est/est
       = 3  Area confidence level true/est
       = 4  Production confidence level true/est
       = 5  Area confidence level true/WC
       = 6  Production confidence level true/WC

Local Variable Descriptions:

M = 4900 + 100 * NCL  location of the requested confidence level in /READNG/ buffer.

Processing:

1. Compute the location of the confidence level requested in /READNG/ buffer.
2. Move the 100 values from /READNG/ to array MCERR.
3. Return.
Subroutine PUNITS (FACTOR, IUNITS)

Purpose:
Subroutine PUNITS sets up the proper print and conversion units for the Population and Monte Carlo Reports.

Input:
/CARDIN/ RPTYPE
         JPARM
         AUNITS

/CONVRT/ CTAMER (3)
         CTMTRC (3)

Linkage:
CALL PUNITS (FACTOR, IUNITS) where
FACTOR = the conversion factor
IUNITS = the descriptor of the units

Local Variable Descriptions:
JFAC = an array of indices for the CTAMTER or CTMTRC array
IFAC = index into the JFAC array
NFAC = index into the CTAMER or CTMTRC array
IAMERU = an array of English units descriptors that correspond to the units in the CTAMER array.
IMTRCU = an array of metric units descriptors that correspond to the units in the CTMTRC array.

Processing:
1. Set IFAC = 0.
2. If RPTYPE = 03, set IFAC = IFAC + 5.
3. IFAC = IFAC + JPARM.
4. NFAC = JFAC (IFAC).
5. If AUNITS ≠ 0, go to 12.
7. FACTOR = 1.0. If RPTYPE = 03 and NFAC = 3, go to 9.
8. FACTOR = CTAMER (NFAC).
9. MFAC = (NFAC - 1) * 3 (the index into IAMERU array).
10. Move descriptor from IAMERU to IUNITS array.
11. Go to 17.
13. FACTOR = 1.0. If RPTYPE = 03 and NFAC = 3, go to 15.
14. FACTOR = CTMTRC (NFAC).
15. MFAC = (NFAC - 1) * 3 (the index into IMTRCU array).
16. Move descriptor from IMTRCU to IUNITS array.
17. Return.
Subroutine MCHIST

Purpose:

This routine produces the Monte Carlo Reports. It accesses the CAS Distribution Output File (CASDIS) through the direct access file read routine FILDA. It computes the required quantities and passes them through common to routine REPRT2 for printing.

Input:

/CARDIN/  LEVEL
         JPARM
         PARMTR
         ICASIN

/HSTOGM/  NRANGE
         RANGES (52)
         NZRNG

/FILBUF/  FILENM (1)
          ICNTRY
          IREGN
          IZONE
          ICASE
          NT
          NREGS
          NZTOT
          NBW
          NPD
          IBIOWD (4)
          IPRDAT (14)
          MCREF
          MCERR (100)

/SELECN/  BIOWD (4)
           IPRD (3, 14)

/PRTLCTL/  KOUT

/RANGE/  START
         INTVL1
         BREAK1
         INTVL2
         BREAK2
         INTVL3
         STOP
Output:

/PRNTID/    CNTRY
             REGION
             ZONE
             ICASEN
             FILNAM (1)
             NAMFMT.

/HSTOGM/    XMVAL
             STDDEV
             REF
             MINVAL
             MAXVAL
             NTOTL
             DATPRD (3)

/READNG/    READNG (5600)

Linkage:

CALL MCHIST     called by POUT

Subroutines Used:

CALL FILDA (IH, IREC)
CALL FZULU (IDATE, IOUT)
CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS)
CALL REPRT2
CALL PAGHDR

Local Variable Description:

IH
REC
J
DELERR
DELSQ
Processing:

1. Initialize arrays, constants, etc.

2. Get header record; verify case numbers on PARMTR values requested.
   DO 1 = 1, 4
   If PARMTR(1) ≠ 0, then verify ICASIN(I) = ICASE6.
   If not equal, write message "ICASIN(I) does not match case number on file."
   Set PARMTR(I) = -PARMTR(I)
   Save in /PRNTID/
       case number
       file name
       country ID
   Set NAMFMT = 1 (file name format type).

3. Loop on PARMTR(I), I = 1 to 4. If PARMTR(I) ≤ 0 skip, otherwise JPARAM = I.

4. Loop on LEVEL (set from 1 to 3) if PARMTR(4) ≠ 0 LEVEL = 3 only.

5. Set IPNT: Get bio-window or prediction date.
   For BIOWD(I), IREC = I, I = 1 to 4.
   For IPRD(3, I), IREC = 4 + I, I = 1 to 14.

6. If LEVEL = 1, then get all NZ TOT records.
   If LEVEL = 2, then get all NREGS records.
Direct Access File Requirements for
RPTYPE = 03 the
CAS Distribution Output File

IH = 1  Read header record and save in /FILBUF/

FILENM (1) = NAME6
ICNTRY = COUN6
ICASE = ICASE6
NT = NT6
NREGS = NREG6
NZTOT = NZTOT6
NBW = NBW6
NPD = NPD6
IBIOWD (4) = BWNBR6
IPRDAT (14) = PDNBR6

IH = 0, JPARM = 1, 2, or 3, NREC = IREC = N and LEVEL = 1 or 2
Read detail record.
Subroutine ANPERR

Purpose:

This routine produces if PARMTR(4) ≠ 0 the Population Area Error Report or if PARMTR(5) ≠ 0 the Population Production Error Report. It accesses the CAS.Cum Output File (CASF) through the direct access file read routine FILDA. It computes the required quantities and passes them through common to routine REPR2 for printing.

Input:

/CARDIN/ LEVEL
        JPARAM

/HSTOGM/ NRANGE
        RANGES (52)
        NZRNG

/FILBUF/ FILENM (1)
         ICNTRY
         IRECN
         IZONE
         ICASE
         NT
         NSTRAT
         NBW
         NPD
         IBIOWD (4)
         IPRDAT (14)
         APVAL (2)
         Note: APVAL(1) = AERRS = PRERRS
         APVAL(2) = TWAS = TPRODS

/SELCNT/ BIOWD (4)
         IPRD (3, 14)

/PRTCTL/ KOUT

Output:

/PRNTID/ CNTRY
         REGION
         ZONE
         ICASEN
         FILNAM (1)
         NAMFMT.
/HSTOGM/
  XMVAL
  STDDEV
  XMPSUM
  SDPSUM
  NTOTL
  MINVAL
  MAXVAL
  DATPRD (3)
  REF

/READNG/
  READNG (500)

Linkage:
  CALL ANPERR     called by POPDRV

Subroutines Used:
  CALL FILDA     reads data file CASF
  CALL FZULU (IDATE, IOUT)
  CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS)
  CALL REPRT2    prints report
  CALL PAGHDR

Local Variable Description:
  IH
  J
  DELSTA
  REFSTA
  DELELR = \sum (READNG)
  DELSQ = \sum (READNG)

Processing:
  1. Initialize arrays, constants, etc.
  2. Get bio-window number or prediction date.
  3. Get header record.
     Verify match on case number.
     Move case number, file name, and country ID to /PRNTID/.
     Set NAMFMT flag.
  4. Get detail record (strata records only).
     If all NSTRAT done, go to 10.
5. Set ID area if first time through REGION and ZONE in /PRNTID/.

6. If not first time through, check for change in LEVEL ID.

7. Determine if prediction point valid for record.

8. Accumulate values (Eqns. 19 and 20, or 21 and 22).
   \[ \text{READNG}(J) = \text{DELERRS}(S) \]
   \[ \text{MINVAL}, \text{MAXVAL} \]
   \[ \text{DELERR} = \text{DELERR} + \text{READNG}(J) \]

9. Read another detail record (go to 4).

10. Compute sums (Eqns. 8, 9, 10, 11, and 12).

11. Tally readings against ranges.

12. Call REPRT2.

13. Get another prediction point at 2.

14. Otherwise, return.
Subroutine FILDA (IH, IPNT, NREC)

Purpose:

This routine calls the direct access routines to read the direct access files determined by RPTYPE = 02 or 03 and JPARM value, and returns the required data in common blocks FILBUF and READNG.

Input:

/CARDIN/  RPTYPE
         JPARAM
         LEVEL

/FBLKBX/  FBLKBX (504)

/FCASCM/  FCASCM (28)

/FCASDS/  FCASDS (303)

Output:

/FILBUF/  FILENM (1)
          ICNTRY
          IREGN
          IZONE
          ICASE
          NT
          NSTRAT
          NREGS
          NZTOT
          NBW
          NPD
          IBIOWD
          IPRDAT
          APVAL (2)
          MGREF
          MCERR (100)

/IXRCD/   IXRCD (1135)

/READNG/  CLEWA (100)
          CLEPRD (100)
          CLATEC (100)
          CIPTEC (100)
          CLATWC (100)
          CLPTWC (100)
Linkage:

CALL FILDA (IH, IPNT, NREC)

Inputs:

IH = 1  Read header record
IH = 0  Read detail record

IPNT = for RPTYPE = 02 the number of the prediction point
(or group no. within each detail record)

NREC = for all RPTYPE the number of the record to be read

Subroutines Used:

CALL RANACF (IFILE, IREC, BUF, N, IX, L, IOPT)

Local Variable Description:

IFILE = 14 for RPTYPE = 02
        4 for RPTYPE = 03

IREC = 1 for header record
       for RPTYPE = 02
          = 63 through 387
       for RPTYPE = 03
          = NREC

BUF = for RPTYPE = 02  FBLKBX
       for RPTYPE = 03  FCASDS

N = NWRDS = 504 for RPTYPE = 02
    303 for RPTYPE = 03

IX = IXRCD an array dimensioned 1135

L = 387 for RPTYPE = 02
    1135 for RPTYPE = 03

IOPT = Entry point option
       = 0  to open the file
       = 1  to read the file
       = -1 to close the file

IH = 1  Read the header record
       0  Read a detail record

IPNT = Prediction point number
       = 1-4 for bio-windows 1-4
       = 5-18 for prediction dates 1-14

NREC = No. of the detail record to read
Processing:

1. If IH = 1
   Open file and read header record
   a. file CASF if RPTYPE = 02
   b. file CASDIS if RPTYPE = 03

2. For RPTYPE = 02
   If IH = 0
   Read detail record no. NREC into FBLKBX.
   Move IPNT group to FCASCM.
   See note 1.

3. For RPTYPE = 03
   IH = 0
   Read detail record no. NREC into FCASDS.
   See note 2.

Note 1:

For CAS Cum Output File (RPTYPE = 02)

Read a detail record into FBLKBX.

Move group IREC into FCASCM.

For all JPARM values
   Move REGN5 to IREGN
   ZONE5 to IZONE

For JPARM = 4
   move TWAS to APVAL(1)
   AERRS to APVAL(2)

For JPARM = 5
   move TPRODS to APVAL(1)
   PRERRS to APVAL(2)

Note 2:

For CAS Distribution File (RPTYPE = 03)

If LEVEL = 3 (country)

Read first country record to FCASDS.

For JPARM = 1
   move AEREF to MCREF
   AERR(I) to MCERR(I), I = 1, 100

For JPARM = 2
   move PEREF to MCREF
   PRERR(I) to MCERR(I), I = 1, 100

For JPARM = 3
   move 100 to MCREF
   YERR(I) to MCERR(I), I = 1, 100

For JPARM = 4
   go to get second country record.

JPARM < 4
   Return
Read third country record to FCASDS.

Move
CLPTC6(I) to CLPTEC(I)
CLATW6(I) to CLATWC(I)
CLPTW6(I) to CLPTWC(I), I = 1, 100

Return.

If LEVEL = 1 or 2 (zone or region)

Read zone or region record into FCASDS.

For JPARM = 1 move AEREF to MCREF
AERR(I) to MCERR(I), I = 1, 100

For JPARM = 2 move PEREF to MCREF
PRERR(I) to MCERR(I), I = 1, 100

For JPARM = 3 move 100 to MCREF
YERR(I) to MCERR(I), I = 1, 100
PART V

SUBROUTINE LISTINGS
SUBROUTINE ANPERR
GREATENS THE POPULATION AREA ERROR REPORT OR THE POPULATION

COMMON BLOCKS
COMMON /CARDIN/RTYPE,UNITS,PARMT(5)
* * * ICASIN(5),LEVEL,JPAORM,IERR
COMMON /FILBUF/FILENR(2),CTHRY,HEMGN,IZONE,ISTRAT
* * * INSTA,ICASE,XSEG
* * * ABUF(125)

DIMENSION KLIN(4),IBINDO(4),IPRDAZ(14),PRODATE(6),XEPW(4),XEYLD(6)

APVAL(2)
MCNRR(100)

REAL MCREF,MCERR
INTEGER FILENR,IPRDAZ

EQUIVALENCE (Kphil),FILENR,IT,=KLIQ(1)

(ABUF(2),[GPIN],STRAT,HEMGS)
(ABUF(3),[GPIN],XTOT)
(KBUF(4),AREA,NBH)
(KBUF(5),XTPK,EPO)
(KBUF(6),[IPIZDO](1))
(KBUF(7),LA)
(KBUF(8),XTP)
(KBUF(9),XCV)
(KBUF(10),XTPA,1,IPRDAZ(11))
(KBUF(11),PRODATE(11))
(KBUF(12),XPKP,EPO)
(KBUF(21),XTYLDO(11))
(KBUF(24),APVAL(1),MCREF)
(KBUF(25),CFLDK(11))
(KBUF(27),XTYLD)

COMMON /HSTOGM/,AMVAL,STDEV,XKPSUM,SDPSUM
** ARANGF,RANGE(51),TOTL,RANGES(52)
** MINVAL,MAXVAL,DAIPHD(3),KZRNK,REF

REAL MINVAL,MAXVAL
INTEGER DATPD

EQUIVALENCE (LYR,DATPD(1)),(LMO,DATPD(2))
** (LCA,DATPD(3))

COMMON /PRITID/,CNTR,Y,REGION,ZONE,STRATA,ICASEN
** FILEN,FILNHM(2)
** ITUP,FNPMT

COMMON /PRITC/
** PRCFLY,APAGE,PLINES,MAYLN,KOUT,HEADER(10)

COMMON /READG/,READG(5600)

DIMENSION CLNYA(100),CLPRD(100),CLATC(100),CLPTEC(100)
** CLATC(100),CLPTEC(100)

EQUIVALENCE
** (READG(501),CLEHA(1))
** (READG(501),CLEPRD(1)), (READG(501),CLETEC(1))
* (READ(5001),CLMTEC(1))
* (READ(5001),CLMTEC(1))
COMMON /SELECT/,BIONO(4),NPRTY(4),IPHD(3,14)
C
C  INTEGER BIORD,NPRTY
C
C  INITIALIZE COUNTERS
IPHT=0
JPHD=0
KPHD=0
JN=0
DATPHD(1)=0
DATPHD(2)=0
DATPHD(3)=0
I=1
C
C  READ HEADER FROM CAS CUM FILE
CALL FILLA (IH,IM,IH)
C
C  CHECK CASE NUMBER
IF (ICASE.EQ. ICASIN(4)) GO TO 150
C
C  CALL PRED(2)
WHITE (KWE+1) ICASE,ICASIN(4)
C
C  FORMAT (5OH,****, ERROR IN SUBROUTINE ANPEHR ******/
C  1 5OH CASE NUMBER FROM CAS CUM FILE = 15
C  2 5OH CASE NUMBER FROM CAS CUM FILE = 15
STP
C
C 150 IH=0
CUTRY=ICHTYP
ICASE=ICASE
FILNAM(1)=FILNAM(1)
FILNAM(2)=FILNAM(2)
NAMF=0
K=0
C
C  PREDICTION POINT LOOP
C 200 JN=JN+1
C
C  IF (JN.LT.4) GO TO 220
C
C  JN(JN).LF.4 (BIONO-INDUK)
IH=IH+D(JN)
C
C  IF (IH.EQ.0) GO TO 220
C
C  DATPHD(1)=IH
C
C  IF (RIORD(IH).NE.0) GO TO 240
C
C  IAP=RIORD(IH).LF.0 HRT RIOND(IH)=0
C
C  (SIH CHECK 1st PROCESSED BY CAS, BUT REPORT NOT DESIRED)
IPHT = 1
GO TO 200
C
C  JN = (G7.4) (PREDICTION DATE)
C 220 JNHD=JNHD+1
C
C  IF (JNHD.GT.8) GO TO 900
C
C  APNAM = IPH(1+KPHD)
C
C  IF (LYK.EQ.0) GO TO 900
C
C  LNY=IPH(1+KPHD)
C
C  APNAM = IPH(2+KPHD)
C
C  APNAM = IPH(3+KPHD)
C
C  CALL FDLA (FLDA,LNY,LYK,ALGF,NDAY)
C
C  APNAM
IDAYS= DAYS
IF ( IDAYS .EQ. IPDAT(JPRD) ) GO TO 240
CALL PAGHM (-2)
WRITE (KOUT,2) KPRD,LHD,LOA,LYR,IDAYS,IPDAT(JPRD)
FORMAT (KPRD) ERROR IN SUBROUTINE AMPERR

1) TH THE +13.2OH-TH PREDICTION DATE +12+1H/+12+1H/+12H = 15
2) TH (ZULU DATE)/42H DOES NOT AGREE WITH THE PREDICTION DATE
3) TH 13.22H FROM THE CAS CUM FILE )
C
IF ( IDAYS .GT. IPDATA(JPRD) ) GO TO 230
C
IDAYS .LT. IPDAT(JPRD)
INPUT PREDICTION DATE .LT. PREDICTION DATE FROM FILE.
C
SKIP INPUT DATE,
IF ( KPRD .LT. 14 ) GO TO 225
GO TO 960
C
IDAYS .LT. IPDAT(JPRD)
C
SKIP DATE FROM CAS CUM FILE.
C
KPRD= KPRD + 1
IPHTA= IPHT + 1
GO TO 230
C
IPHTA= IPHT + 1
ISIPATE= 0
NPP= 62
C
ZONE, REGION, OR COUNTRY LOOP
300 DO 310 I=1,500
READ(I,J)= 0.0
C
310 CONTINUE
DO 320 J=1,500
INPFO(J)= 0.0
C
320 CONTINUE
C
MINVAL= 1.E10
MAXVAL= -1.E10
REGION= 0
ZONF= 0
DELFR= 0.0
DELSN= 0.0
KFF= 0.0
J= 0
JNUTL= 0
C
IF ( ISTHAT .LT. 4 ) GO TO 415
C
STRATA LOOP
400 ISTRATA= ISTRATA + 1
NREC= NREC + 1
C
REAL STRATA RECORD FROM CAS CUM FILE
CALL FTRA (14,IPHT+1REC)
C
IF ( ISTHAT .LT. 4 AND ISTHAT .LT. 10000 ) GO TO 410
C
CALL PACHM(-2)
C
WRITE (KOUT,3) ISTRATA,ISTRATA,NREC
C
FORMAT (KPRD) ERROR IN SUBROUTINE AMPERR
C
1) 2HI ING REAL STRATA ID #1H FOR 14:
2) 2HI-TH STRATA (RECORD #1H,17H OF CAS CUM FILE )
C
STOP
C
2824-6024-0U-00
Page 316
C 410 IF ( ZONE *NE. 0 ) GO TO 420
C FIRST STRATA OF THIS ZONE OR REGION
C
C 414 REGION = 179
C ZONE = 179
C
C 420 IF ( LEVEL = 2 ) 430, 440 + 460
C
C 420 LEVEL = 1 (ZONb
C 430 IF ( ZONE *NE. ZONE ) GO TO 500
C GO TO 500
C
C 440 IF ( REGION *NE. 179 ) GO TO 500
C
C 460 STRF POPULATION ERROR IN READING ARRAY AND UPDATE
C MIN, MAX, VALUES AND SUMS.
C
C 460 IF J = 1 + 1
C LXME APVAL(I)/XNT
C READING(J) = ERR
C MINVAL = MIN(MINVAL, ERR)
C MAXVAL = MAX(MAXVAL, ERR)
C DELPHR = UPHR + ERR
C DEL54 = DEL50 + ERR
C REF = REF * APVAL(2)
C TEST FOR END OF COUNTRY
C IF ( ISTRAT .LT. ISTRAT ) GO TO 400
C
C 500 END OF ZONE, REGION, OR COUNTRY
C CUMULATE MEAN VALUE, STANDARD DEVIATION, ETC. (Eqs. 6-12)
C
C 500 XJ = J
C 500 NTOLE = J
C 500 REF = REF/XNT
C 500 XJ=VAL = 100.0/DELEN/(XJ*REF) 
C 500 STDEV = 0.0
C 500 IF ( J .GT. 1 ) STDEV = 100.0/REF * SORT ( DELSG - DELERN**2/XJ) 
C 500 /XJ=1.0) 
C 500 XPSU= XJ*XJXVAL
C 500 SPSU = STDEV*SORT(XJ)
C 500 TALLY READINGS IN RANGES
C 500 DO 550 J = 1, NTOLE
C 500 PCDERR = 0.0
C 500 IF ( REF * NE. 0.0 ) PCDERR = 100.0*READING(J)/REF
C 500 IF ( PCDERR * NE. 0.0 ) GO TO 520
C 500 PCDERR = 0.0
C 500 IRANGE(I INFNG) = IRANGE(7KHNG) + 1
C 500 GO TO 550
C 500 PCDERR = 0.0
C 500 IRANGE(I INFNG) = IRANGE(KHNG) + 1
C 500 GO TO 550
C 520 IRANGE(K) = IRANGE(K) + 1
C 500 CONTINUE
C 550 CONTINUE
C 550 CALL REPT2
C 500 IF ( ISTRAT .LT. ISTRAT ) GO TO 300
C END OF COUNTRY, GO BACK FOR NEXT PREDICTION POINT.
C INTEGER BIWND,W,A,
C COMMON /PRTCrIL/NPAGE,NLINES,MAXLIN,KOUT,READER(16)
C COMMON /MATCH/ MATCH
C COMMON /READR/ HEADING(5000)
C DIMENSION CLEMA(100),CLEFD0(100),CLATEC(100),CLPTEC(100)

C EQUIVALENCE ((READR(5001)+CLEMA(1)),(READR(5002)+CLEFD0(1))
C (READR(5003)+CLATEC(1)),(READR(5004)+CLPTEC(1))

C *** LOCAL STORAGE
C INTEGER KSTH4
C DATA ARAPS /10289.712/
J=1J=0
JPHO = 0

C *** PROCESS FOR BIU-WINDOWS FIRST
C 10 I=1
C IF (J*II .LT. 0) GO TO 30
C 15 J=1J=J+1
C IF (J*II .GT. 4) GO TO 20
C IF (STH4(J*II)) NE.0) GO TO 50
C 20 J*II = -1
C 30 JPHO = JPHO+1
C IF (JPHO .GT. 14) GO TO 500
C IF (JPHO(1,JPHO)) EQ.0) GO TO 500
C 50 DO 60 J = 1,3200
C 60 READR(I) = 0.0
C 100 J = 0
C MXX = 0
C NTL = 0
C 101 DO 70 I = 1,51
C 70 IRAH(1) = 0
C DELSU = 0.0
C DELHL = 0.0
C REF = 0.0
C NVAL = -1.0
C MAXVAL = -1.0
C DEL1 = 0.0
C IF (JH,FU,0) GO TO 80
C 102
C *** GET HEADER RECORD
C
C CALL F(IIFC(IN))
C IF ((IENP.NE.0) GO TO 500
C ICASEN = ICASE
C FILENM(1) = FILENM(1)
C FILENM(2) = FILENM(2)
C NAMAT = 0
C IH = 0
C *** GET DETAIL RECORDS
C
C NO CALL FIEL5(IN)
C IF (INH.LT.1) GO TO 200
C IF (AXH.GT.0) GO TO 100
C IF (ATCH.EQ. -2) GO TO 210
C ENTY = ENTHY
C EGEN = EGENF
C ZHALE = ZH0F
C STHATA = ISTHATA
C SINHFA = ISTHFA
C GO TO 120
C
C TEST FOR CHANGE IN TD AREA
C 100 IF (ISTHATA.NE.ISNTHA.OR.ISTRIA.NE.STRATA) GO TO 200
C GO TO (105,110,120) LEVEL
C 105 IF (ISONE.NE.ZONE) GO TO 200
C 110 IF (INRED.IE.REGIONJ) GO TO 200
C
C 120 IF (JWIN.LT.10) GO TO 150
C IF (PRDATE(JWIN),EN>0) GO TO 80
C IDATE = FDJPAD(JWIN)
C UELSD = UELES+(XEP-(JWIN)-XTPM1)
C KXH = KXH+1
C GO TO 80
C
C *** USE PREDICTION DATE INPUT
C 150 FLDA = IPRO1(4+JPRD)
C L*O = IPRD(2+JPRD)
C LVH = IPRD(1+JPRD)
C CALL LFFAC(LDCA,LCT,LVL+ALF8M+DAYS)
C I DAYS = DAYS
C UD 160 T = 1,1
C IX = +PHIY(1)
C IF ((X.PE.3) GO TO 160
C IF ((RDATE(IY).LE.0) GO TO 160
C IF ((IY.EQ.0) +PUA1E(IY)) GO TO 160
160 CONTINUE 
    GO TO 80
C
180 IDATL = IDAYS
   DELSB = DELSH*(XEP(I) = XTPH)
   KXM = KXM + 1
C
GO TO 80
C
C *** CHANGE IN IDO AREA
C
200 CONTINUE
   IF (KXM .EQ. 0) GO TO 210
   XM = KXM
   XH = XH + 1
   XM = AREA/(XM*AREAS)
   XH = XI*XM*AREAS
   J = J + 1
   XM = KXM
C
READ(J) = DELSH/XM*XM
C
MAXVAL = MAX1(MAXVAL,READ(J))
MAXVAL = MIN1(MAXVAL,READ(J))
C
DELR = DELERR*READ(J)
REF = REF + XTPH*XM
DELS = DELS + DELR*READ(J)**2
C
210 CONTINUE
   DELSB = 0
   KXM = 0
C
IF (I .LT. J) GO TO 250
   GO TO (220, 230, 240), LEVEL
C
220 IF (IZONE .LE. ZOVL) GO TO 250
C
230 IF (INEQ .LE. REGION) GO TO 80
   GO TO 250
C
240 IF (I .EQ. 0) GO TO 80
C
250 IF (J .EQ. 0) GO TO 400
C
NTOL = J
C
U0 = 300
   I = 1
   NTOL
PCERR = 0.0
IF (REF .LE. 0.0) PCERR = 100.0*READ(I)/REF
GO TO 300
C
240 IF (PCERR .LE. 0.0) GO TO 260
C
I RANGEm (ZRING) = IRANGE(KRING)+1
GO TO 300
C
260 IF (PCERR .GE. RANGES(K+1)) GO TO 260
C
IRANGE(K) = IRANGE(K)+1
GO TO 300
**SUBROUTINE CLMOVE(NCL)**

C THIS ROUTINE CALLED BY MCHIST MOVES THE CONFIDENCE LEVELS one at a time from the reading buffer to FILBUF.

C COMMUNITY STORAGE

C COMMON /FILBUF/FILNAME(2),ICNTKY,IREGN,IZONE,ISTRTA
C ** ISSTATIICASE,MASEG
C ** XNDF(12)

C DIMENSION XNDF(4),IBIND(4),IPDAT(14),PHD(6),ICPD(4),KEYLD(6)
C ** APVAL(2)
C ** XCEHR(100)

C REAL XCHEF,XCMEM
C INTEGER FILEM,MHDAT

C EQUIVALENC (XNDF(1),NSER,INT,NHEN(NHSEG))
C ** (XNDF(2),(IPNO,ECHR,INT))
C ** (XNDF(3),XREND,KNT)
C ** (XNDF(4),XIpha,ECHR)
C ** (XNDF(5),XTPM,IPNO)
C ** (XNDF(6),...BNND(1))
C ** (XNDF(7),XK)
C ** (XNDF(8),XKPHA)
C ** (XNDF(9),...XKPHA)
C ** (XNDF(10),XTPD,IPDAT(1))
C ** (XNDF(11),PHD(1))
C ** (XNDF(12),XLD(1))
C ** (XNDF(13),...PHD(1))
C ** (XNDF(14),...PHD(1))
C ** (XNDF(27),XTYL)

C COMMON /READGY,READGA(5000)
C DIMENSION GLEA(100),CLEPRU(100),CLATEC(100),CLPTEC(100)
C ** CLATEC(100),CLPTEC(100)

C EQUIVALENCE READGU(5001),CLEHA(1)
C ** READGU(5001),CLATEC(1)
C ** READGU(5001),CLPTEC(1)
C ** READGU(5001),CLATWC(1)

C COMPUTE THE STARTING INDEX IN BUFFER

C H = 4000*ICL
C
C SET REFERENCE VALUE

C XCHEF = 100

C MOVE = C READING VALUES

D10 I = 1100
SUBROUTINE FILDA (IMNPNT,ANLRT)
CALL THE RARACF TO READ THE CAS CUMULATIVEFILDA
FILE (CASF) OR THE CAS DISTRIBUTION FILE (CASDIS)
COMMON BLKUS
COMMON /CANDIN/RPTYPE,AUNIT$,PARK(5)
* ICASIN(5)+LEVEL+PARK+ERR
INTEGER NTYPE,AUNIT$,PARK
COMMON /FILNAM/ FRAKLY(28218)
COMMON /FACUS/ FACUS(353)
DIMENSION MINUS(5),POW(100),AERR(100),MPERR(100),YERR(100)
** CLATp(100)+CLEPR(100)+CLCIT(100)+CLPLC(100)
** CLATb(100), CLPP(100)
INTEGER BMINP,BDMP,BLONG,BOUNG
LOGICAL CI
** (FCASDS(1),NATEX,AEREF), (FCASDS(2),ICASIN,PEREF)
** (FCASDS(3),COUNG)
** (FCASDS(4),NTH,AERR(1),CLAT(1),CLPTC(1))
** (FCASDS(5),NTHSDS), (FCASDS(6),NTOTO)
** (FCASDS(7),NTD), (FCASDS(7),NHTD)
** (FCASDS(10),PNOME(1),CLEPR(1),CLAT(b(1))
** (FCASDS(11),YERR(1),CLCIT(1),CLPLC(1))
** (FCASDS(12),ICUSF,ICUSF,ICUSF)
** INSTUS,ICASE,MASEG
** NODF(125)
** DIMENSION KFIN(4), KTHND(4), IPRTAT(14), FRDATE(6), XEPH(6), XLYLO(6)
** AFVAL(2)
** NFIN(100)
** REAL PCREF,NCRE,
** INTEGER FILEIN,PRDATE
EQUIVALENT (KFINF(1),MSEG$),NTH,KFINH(1)
** (KDFUF(7),IDNAM,ISRT,NDNS)
** (KDFU(5),XSPK,XTOT)
** (FCASDS(12),BRA, BPL)
** (KEEF(5),XTP,VPD)
** (KDFU(5),WSPK,WSUR)
** (FCASDS(10),XFRAT,IPRTAT(1))
** (KDFU(P),FRDATE(1))
** (KDFU(17),XEPH(1))
** (THDF(21),XLYL(1))
** (KDFU(24),AFVAL(1),NCREF)
** (KDFU(25),CNKEX(1))
** GDFU(27),XLYL)
COMMON /XKIN(3),XGUD(580)
COMMON /ANLRT/ MPAR,SL14S,MAXLIN,NOUT,HEADEH(16)
INTEGER IMNPNT
COMMON /FILNAM/ FRAKLY(28218)
COMMON /FACUS/ FACUS(353)
COMMON /FILNAM/ FRAKLY(28218)
COMMON /FACUS/ FACUS(353)
DIMENSION CLEMA(100),CLEPRD(100),CLATEC(100),CLPTEC(100)
*  CLATX(100),CLPTX(100)

EQUIVALENCE (READS(S001),CLEMA(1))

DIMENSION InLKHX(S04)
EQUIVALENCE (L=LKHX+FLBKDX)

INPUT ...

LEVEL = 1 FOR ZONE LEVEL
  = 2 FOR REGION LEVEL
  = 3 FOR COUNTRY LEVEL.

RPTYPE = REPORT TYPE
  (2 FOR POPULATION ERRORS)
  (5 FOR MORTALITY ERRORS OR CONFIDENCE LEVELS)

JHARM = INDEX WHICH DETERMINES PARAMETER TO BE REPORTED...
 IF RPTYPE = 2 AND
 IF JHARM = 4, THEN POPULATION AREA ERROR IS REPORTED.
 IF JHARM = 5, THEN POPULATION MEAN ERROR IS REPORTED.
 IF RPTYPE = 3 AND
 IF JHARM = 1, THEN AREA DATA IS EXTRACTED FROM CASDIS.
 IF JHARM = 2, THEN POPULATION DATA IS EXTRACTED FROM CASDIS.
 IF JHARM = 3, THEN EPI DATA IS EXTRACTED FROM CASDIS.
 IF JHARM = 4, THEN CONFIDENCE LEVELS ARE EXTRACTED FROM CASDIS.

LINKAGE ...
CALL FILOA (IH,INPT,NREC)
WHERE
  IN = 1 TO READ HEADER RECORD;
  N = 0 TO READ DATA (DETAIL) RECORD.
INPT = PREDICTION POINT NUMBER.
NREC = RECORD NUMBER TO READ.

LOCAL VARIABLES ...
FILE = FILE NUMBER
NWORDS = NUMBER OF WORDS TO READ
LUX = LENGTH OF INDEX RECORD (MAXIMUM RECORD NUMBER)

INTEGER CASF, CASUS

DATA CASF, LCASF, LCASUS, LCASD, LIXCAS, LIXDIS
  1, 14, 504, 4, 303, 386, 506 /
MORS = LUX-85*506 ALLOWS UP TO 8 PREDICTION POINTS
(I=FLBKX IN LKHX=504)
(I=FLBKX IN LKHX=504)

IPEC = IREC
IPF = I=1

TEST REPORT TYPE AND SET FILE NUMBER, FILE LENGTH AND FILE
NAME TO READ CAS SUM FILE IF RPTYPE = 2 OR TO READ CAS DIST.
FILE IF RPTYPE = 3.
C  IF ( RTYPE = EQ. 2 ) GO TO 120
C  IF ( RTYPE = EQ. 3 ) GO TO 130
C  ANY VALUE OF RTYPE EXCEPT 2 OR 3 IS AN ERROR
C
110 CALL PAGNH (-2)
   WRITE (KUNIT+1) RTYPE, JPARAM, LEVEL
   FORMAT ('/3H9=**** ERROR IN SUBROUTINE FILDA ****/
   1 FHM, ILLEGAL VALUE OF RTYPE = *14*12H OR JPARAM = *14*12H
   2 IONH, LEVEL = *14*
   STOP
C
C  HTYPE = 2
120 IF ( JPARAM .LT. 4 ) GO TO 110
   IFILE = CASEF
   ANAS = LCASEF
   FILEM(1) = 6HCASUM
   LIAS = LIXCAS
   IF (IM .NE. 0) DEFINE FILE 14(386+504+U+1DUM)
   GO TO 150
C  HTYPE = 3
130 IFILE = CASDIS
   ANAS = LCASL
   FILEM(1) = 6HCASDIS
   LIAS = LIXDIS
   IF (IM .NE. 0) DEFINE FILE 4(506+303+U+1DUM)
C
C  IM = 1, OPEN RANDOM ACCESS FILE AND READ HEADER RECORD
   CONTINUE
   CALL PREAD (IFILE, IBLKBMX*, FNAME*, IXRCU, LIX*1)
   CHECK FILE NAME
   IF ( IBLKBMX(1) .EQ. FILEM(1) ) GO TO 170
   IF ERROR, FILE NAME DOES NOT MATCH
   CALL PAGNH (-2)
   WRITE (KUNIT+1) FILEM(1)*FBLKBMX(1)*1
   FORMAT ('/3H9=**** ERROR IN SUBROUTINE FILDA ****/
   1 FILE NAME OF *AB*1UM FILE = *AB*3TH--DOES NOT MATCH EXPEDE
   2 ICASE VALUE )
   STOP
C  EXTRACT INFORMATION FROM HEADER RECORD
170 ICASE = IBLKBMX(3)
   LCASE = IBLKBMX(4)
   I. = IBLKBMX(5)
   ANES = IBLKBMX(6)
   A210 = IBLKBMX(7)
   IF ( RTYPE .EQ. 3) VSTAT = IBLKBMX(7)
   VSTATE = IBLKBMX(8)
   PND = IBLKBMX(9)
C  DO 160 I=1+4
   CONTINUE
C  IF ( IIF .EQ. 0 ) GO TO 200
   GO 160 I=1+4
IPRDAT(1) = IBLKBX(I+13)
CONTINUE
C GO TO 950
C READ DATA (DETAIL) RECORD AND EXTRACT DESIRED DATA
C IF ( RPTYPE .EQ. 3 ) GO TO 300
C RPTYPE = 2, READ DETAIL RECORD FROM CAS CUM FILE
C CALL HACF (IFILE,IREC,FBLKBX,NAROS,IXRCO,LIX+1)
C IF ( JPARM .EQ. 5 ) GO TO 240
C JPARM = 4, STORE TRUE WA AND AREA ERROR
C APVAL(1) = FBLKBX(1+IPP)
C APVAL(2) = FBLKBX(6+IPP)
C GO TO 900
C STORE TRUE PRODUCTION AND PRODUCTION ERROR
C APVAL(1) = FBLKBX(12+IPP)
C APVAL(2) = FBLKBX(10+IPP)
C GO TO 900
C RPTYPE = 3, READ DETAIL RECORD FROM CAS DISTR. FILE
C IF ( JPARM .EQ. 4 ) IREC = IREC + 1
C CALL HACF (IFILE,IREC,FCASOS,NAROS,IXRCO,LIX+1)
C GO TO (310,320,330,400), JPARM
C JPARM = 1
C II = 4
C GO TO 340
C JPARM = 2
C II = 104
C GO TO 400
C JPARM = 3
C II = 204
C MCREF = FCASOS(JPARM)
C DO 350 II=1,100
C MCREF(II) = FCASOS(II)
C 350 II = II + 1
C GO TO 400
C JPARM = 4
C DO 400 II=1,100
C CLEXP(1) = FCASOS(I+3)
C CLEXP(1) = FCASOS(I+3)
C 410 CLEXP(1) = FCASOS(I+3)
C READ NEXT GROUP OF CONFIDENCE LEVELS
IIFC = IIFC + 1
C CALL HACF (IFILE,IIFC,FCASOS,NAROS,IXRCO,LIX+1)
C GO TO 450
C CLEXP(1) = FCASOS(I+3)
INTEGER COUN2, REGN2, ZONE2, STRTA2, SSTA2, SEGMT2
COMMON /FCAMSF/ COUN2, REGN2, ZONE2, STRTA2, SSTA2, SEGMT2

C

DIMENSION IKIND(3,4)

C

INTEGER COUN3, REGN3, ZONE3, STRTA3, SSTA3, SEGMT3
COMMON /FCAMSF/ COUN2, REGN2, ZONE2, STRTA2, SSTA2, SEGMT2

C

DIMENSION IYSPT(3,6)

C

INTEGER COUN4, REGN4, ZONE4, STRTA4, YSTR, YSPT(3,6)
COMMON /FCAMSF/ COUN2, REGN2, ZONE2, STRTA2, SSTA2, SEGMT2

C

DIMENSION ICASE3, IMSEG3, ICASE4, IMSEG4
COMMON /SUBHT/ ICASE3, IMSEG3, ICASE4, IMSEG4

C

*** LOCAL STORAGE

C

INTEGER FCAM, FCAM, FCAM, FCAM, FCAM, FCAM
DATA SUBHT, ICASE, IMSEG, ICASE, ICASE, ICASE, ICASE, ICASE
DATA 12222, 12222, 12222

C

*** BRANCH TO READ HEADER RECORD(S) OR DETAIL RECORD(S)

C

IF (IH, FCAM, 0) GOTO 300

C

*** BRANCH BY RPTYPE TO PROCESS HEADER RECORD(S)

C

GO TO (100, 200, 300, 400, 120), RPTYPE

C

*** RPTYPE = 1, SUBSTRATA HISTORICAL FILE (SUBHT)

C

READ SUHIST AND READ HEADER RECORD

C

100 REIND FSUB

C

READ (FSUP) NAME(1), NAME(2), ICASE, IMSEG

C

SAVE HEADER INFO

C

FILSANY = NAME(1)
FILSANY = NAME(12)
ICASE = ICASE
IMSEG = IMSEG
1 = 1
M1 = MSEG + 12
GO TO 150

C

*** RPTYPE = 4, DATA ACQUISITION FILE (ACQUI)

C

120 READ FDACU
C  READ (FACD) NAME(1),NAME(2),ICASEF
C  SAVE HEADER INFO
C  FILENM(1) = NAME(1)
C  FILENM(2) = NAME(2)
C  ICASE = ICASEF
C  I = 1
C  *** TEST FOR MATCH ON REQUESTED FILE CASE NUMBER
C  150 IF (ICASIN(1),EQ,ICASEF) GO TO 500
C  *** ERROR - CASE NUMBERS DO NOT MATCH
C  CALL ERRHR(-4)
C  WRITE (KOUT,6260) I,ICASIN(1),ICASEF,NAME(1),NAME(2)
C  *** STOP  
C  GO TO -700
C  *** NTYPE = 2  
C  PARMTYP(1) = 1, SEGMENT THRU FILE (SEGTEM) + SUBST (JPARM = 1)
C  PARMTYP(2) = 1, CAMS OUTPUT FILE (CAMSF) + SUBST (JPARM = 2)
C  PARMTYP(3) = 1, YES OUTPUT FILE (YESOUT) + SUBST (JPARM = 3)
C  200 GO TO (210,220,230),JPARM
C  JPARM = 1, SEGTEM + SUBST
C  210 REMIND FSEG
C  READ (FSEG) NAME(1),NAME(2),ICASEF
C  GO TO 250,
C  JPARM = 2, CAMSF + SUBST
C  220 REMIND FCAM
C  READ (FCAM) NAME(1),NAME(2),ICASEF
C  GO TO 250
C  JPARM = 3, YESOUT + SUBST
C  230 REMIND FYES
C  READ (FYES) NAME(1),NAME(2),ICASEF
C  SAVE HEADER INFO
C  FILENM(1) = NAME(1)
C  FILENM(2) = NAME(2)
C  ICASE = ICASEF
C  I = 1
C  *** READ SUBST HEADER RECORD
C  250 REMIND FSHR
C  *** STOP
C  300 STOP
C READ (FSUB) NAME(1) NAME(2) ICASEF IMXSEG
C
C IMXSEG = IMXSEG
C
C **** TEST FOR MATCH ON REQUESTED FILE CASE NUMBERS
C
C IF ICASE (JPAKM) EQ ICASE (FNAME) GO TO 270
C
C **** ERROR - NO MATCH
C
C IFERR = JPAKM
C
C CALL PAGHDR(-4)
C
C WRITE (XOUT,1200) JPAKM, ICASE (JPAKM), ICASE (FNAME), (1) FILENM (1), (2) FILENM (2)
C
C **** TEST FOR MATCH ON SUBSET FILE
C
C 270 IF ICASE (JPAKM), ICASE (FNAME) GO TO 280
C
C MATCH = 0
C
C GO TO 500
C
C **** ERROR - NO MATCH
C
C 280 IF (IFRNF, EQ, 0) CALL PAGHDR (0)
C
C IFNRM = JPAKM
C
C CALL PAGHDR (4)
C
C WRITE (XOUT, 1200) JPAKM, ICASE (JPAKM), ICASE (NAME(1)), NAME(2)
C
C **** RETURN
C
C GO TO 500
C
C **** READ DETAIL RECORD (5)
C
C 310 GO TO (310, 340, 600, 360, 330), RPTYPE
C
C C**** RPTYPE = 1 - SUBSET FILE
C
C 340 READ ( FSUB ) COUNF, HFONZ, ZONE1, STN1A, SUB1A, INEG,
C
C #
C
C ( HFONZ (1), 1 = 1, HF02 )
C
C 350 TEST FOR END-OF-FILE MARKER
C
C IF ( COUNF, NE, IZZZZ ) GO TO 320
C
C **** END-OF-FILE
C
C IF = -1
C
C GO TO 500
C
C **** WRITE DETAIL VALUES TO FILEBUF
C
C 320 ICNTN = COUNF
C
C INEG = HFONZ
C
C ZONE = HZONE1
C
C 1STN1A = STN1A
C
C 330 IFSEG = 1 + G
C
C JPAKM = HSEG ( MSEG + 1 )
C
C CALL CONVT ( 1,1 ) FROM PERCENT TO FRACTION
C
C XHEN = XSEG ( MSEG + 2 ) * 0.01
C
C **** END-OF-FILE
C
C 350 FILEBUF
C
C 360 WRITE ( XOUT, 1200) JPAKM, ICASE (JPAKM), ICASE (FNAME), (1) FILENM (1), (2) FILENM (2)
C
C 370 END
C  CONVEXT SUBSTRATA LAND AREA FROM KM*2 TO HECTARES
    AREA = XSFSG(XSEG+3) * 100.0
C  CONVEXT TRUE SHA FROM PERCENT TO FRACTION
    N = XHFH (XSEG+5)
    NA = NSSEG (XSEG+6)
    XHTPH = XSEG (XSEG+7)
    XCVP = XSFUG (XSEG+10)
C  GO TO 500
C  *** NPTYPE = 4 ACQUISE FILE
C  330 READ (FACO) COUNT,REGION,ZONET,STRTA,SUBSTA,SEGMT
C  (*
    ((I+IND(1)+J+25)+1+41) ITOTL
C  **  TEST FOR END-OF-FILE MARKER
C  IF (COUNT.NE.IZZZ) GO TO 340
C  *** END-OF-FILE
C  IH = -1
C  GO TO 500
C  *** NPTYPE = 2
C  340 IDNTY = COUNT
C  IPEN = REGION
C  IZON = ZONET
C  ISTRTA = STRTA
C  ISUBSTA = SUBSTA
C  ILO = 150
C  350 KAUF(I) = I*IND(1,1)
C  GO TO 500
C  *** NPTYPE = 3
C  360 IF (JAPM,LE,3) GO TO 440
C  C
C  IF (MATCH,LE,1) GO TO 390
C  **  READ (FSUR) COUNT,REGION,ZONET,STRTA,SUBSTA,INEG
C  (*
C  (XSEG(1),1=1,NXRD3)
C  **  TEST FOR END-OF-FILE MARKER
C  IF (COUNT,LE,12722) GO TO 370
C  IH = -1
C  GO TO 500
C  370 IF (MATCH,LE,0) GO TO 390
C  MATCH = 0
C  *** IF (MATCH,LE,6) = REGION,1500+160
C  IF (1+6)-REGION,1500+160

ORIGINAL PAGE IS OF POOR QUALITY.
800 IF (I30NE - ZONE1) = 600, R10, 660
210 IF (I31STA = STRTA1) = 360, R20, 660
420 IF (I31STA = SUBSTA1) = 360, R30, 660
630 MATCH = 1
360 MATCH = -1
GO TO 300
GO TO 360

*** MOVE SUBHST VALUES TO FILBHF
380 IDIIHY = COIIV1
IPJIIH = REGI1
IZUNE = ZONE1
I5RTAI = STRTA1
ISUBTA = SUBSTA1

I5SEG = TISH
CONVRRT SUBHST LAA I AREA FROM KH**2 TO MECTAKES
AREA = XSEG(X5SEG+3)*100.0
CONVRRT TRUE PW FROM PERCENT TO FRACTION
XTPH = XSEG(X5SEG+4)*.01
N = H5SEG(X5SEG+5)
NA = H5SEG(X5SEG+6)

C IF (MATCH,FC,1) GO TO 509
370 IF (IPRINT,EN,7) GO TO 420
C *** JPARA =1, SEGMENT TRUTH FILE (SEGTHRU) + SUBHST
400 READ (FSEG) COUH2,KEGH2,ZONE2,STRTA2,SUBSTA2,SEGMT2

*1
(DUKE2=2,FC=1), I=1,6), TP=2, TP=2
C *** TEST FOR EOF
460 IF (COUP=ZNE,ZZZZ) GO TO 409
C
480 IF (IPFICH, NEE, ZZIZZ) GO TO 409
C
490 IF (IPFICH, NEE, ZZIZZ) GO TO 409
C
460 IF (IPFICH, NEE, ZZIZZ) GO TO 409
C
470 IF (IPFICH, NEE, ZZIZZ) GO TO 409
C
480 IF (IPFICH, NEE, ZZIZZ) GO TO 409
C
490 IF (IPFICH, NEE, ZZIZZ) GO TO 409
C
500 IF (MATCH, FC, 0) MATCH = 1
510 IF (MATCH, FC, 1) MATCH = -1
GO TO 410
GO TO 360
C
520 IF (MATCH, FC, 0) MATCH = 1
530 IF (MATCH, FC, 1) MATCH = -1
GO TO 410
GO TO 360
GO TO 500

C*** JPARH = 2+ CAMS OUTPUT FILE (CAMSF) + SUBMST
C
C 420 READ (FCAM) COU4S,REGN4,ZONE4,STRTA4,STSTA4,SEGTM4
      TP43,((ITR assigns to J(J+1))J=1:3,J=1:4)
C
C IF (COUN4,NE,IZZZZ) GO TO 425
C
C 425 IF ( ITREG = REG4 ) 429,424,420
C 424 IF ( ITZON = ZONE4 ) 429,426,420
C 426 IF ( ISTSTA = STSTA4 ) 429,427,420
C 427 IF ( ISTSTA = STSTA4 ) 429,428,420
C 428 MATCH = 1
C 429 CONTINUE
C
C IF (MATCH,EQ,0) MATCH = -1
C IF (MATCH,EQ,1) MATCH = +1
C
C IREG4 = REG4
C IZON4 = ZONE4
C ISTSTA4 = STSTA4
C ISUBSTA = SUBSTA4
C
C 430 XTPH1 = TP43*1:0.01
C
C 435 IF ( MATCH,EQ,1 ) GO TO 369
C
C 440 IF (MATCH,EQ,1) GO TO 470
C
C*** JPARF = 3+ YES OUTPUT FILE (YESOUT) + SUBMST
C
C 440 IF (MATCH,EQ,1) GO TO 470
C
C 470 READ (FYSF) COU4S,REGN4,ZONE4,STRTA4,STSTA4,SEGTM4
      TP43,((ITR assigns to J(J+1))J=1:3,J=1:4)
C
C IF (COUN4,NE,IZZZZ) GO TO 450
C
C 450 IF (MATCH,EQ,0) GO TO 460
C
C MATCH = 0
C
C IF(ITREG = REG4) 460,900,940
C 900 IF ( ITZON = ZONE4 ) 440,910,940
C 910 IF (STSTA4 = STSTA4) 460,920,940
C 920 MATCH = 1
GO TO 440
990 MATCH = -1
GO TO 440
C
C *** MOVE YESUFT VALUES TO FILBUF
C
460 ICATHY = COUNT
IREGH = REGN
170NE = ZONE
1STPA = STPA
C
XYLF = YSTE
GO 465 I = 1+6
PRINT(I) = YSPNT(1,I)
465 KEYLF(I) = YSPNT(2,I)
C
IF (MATCH.EQ.1) GO TO 500

470 READ (FSUB) COUNT,REGH,ZONE1,STPA1,INUG

C
480 IF (COUNT.EQ.1) GO TO 480
C
IM = -1
GO TO 500
C
480 IF (IREGH = REGH1) 488+482+470
482 IF (IZONE = ZONE1) 488+484+470
484 IF (1STPA = STPA1) 488+486+470
480 MATCH = 1
GO TO 490
480 CONTINUE.
C
IF (MATCH .EQ. 0) MATCH = 1
IF (MATCH .EQ. 1) MATCH = 2
C
IF (IREGH = REGH1) IREGH
IZONE = ZONE
1STPA = STPA
C
CONV5 SUBSTRATA LAND AREA FROM KM2 TO HECTARES
C
490 AREA = XSFG(XSEG*4+3)*100.0
C
CONV5 TRUE FL FROM PERCENT TO FRACTION
C
XTPH = XSFG(XSEG*4+4)*.01
N = XSEG*4+5
N = XSEG*4+6
C
IF (MATCH .EQ. -1) GO TO 440
C
C *** RETURN
C
500 RETURN
C
600 CONTINUE
C
GO TO 500
C *** NON-MATCH ON CASE NUMBER FOR FSUB (RPTYPE=1) OR FACQ (RPTYPE=4)
C
700 STOP
C *** ERROR MESSAGE
C 6200 FORMAT (50H *** ICASIN(I) DOES NOT MATCH CASE NUMBER ON FILE
* 12H ICASIN(+I+2n)=+I+14M FILE CASE NO. +14
* 11H FILE NAME+2n/14M /14+10 (OH********))
END
FILSEQ
FILSEQ
FILSEQ
FILSEQ
FILSEQ
FILSEQ
FILSEQ
FILSEQ
FILSEQ
FILSEQ
FILSEQ

ORIGINAL PAGE IS OF POOR QUALITY
IN = 1
IFIRST = 0
DO 20 I = 1,3
K = T(I) + 1
IF (T(I) = 0) GO TO 1
DO 20 J = 1,16
20 KNT10(J + 1) = 0

*** GET HEADER RECORD
CALL FILEC(1M)
ICASEM = ICASEM
FILEM(1) = FILEM(1)
FILEM(2) = FILEM(2)

*** GET DETAIL RECORD
IN = 0
50 CALL FILESEQ(IN)
IF (IM.LT.0) GO TO 200
C IF (IFIRST.GT.0) GO TO 100
C ENTRY = ENTRY
REGION = REGION
ZONE = ZONE
IFIRST = 1
GO TO 120

*** TEST FOR CHANGE IN I.D. AREA
100 IF (IZONE.GT.ZONE) GO TO 200
IF (ITRIG.GT_REGION) GO TO 200
C
120 K = 0
KI = 0
KMTZ = KMTZ + 1
UP 150 X = 1,4
IF (X = T(I) .EQ. 0) GO TO 130
K = K + 1
KI = K + 1
GO TO 120

130 CONTINUE
C IF (K.FLT.0) GO TO 140
C
140 GO TO (150, 150, 160, 180, 180, 180, 180)
C
150 KIPOL(KI + 1) = KIPOL(KI + 1) + 1
GO TO 50
C 160 IF (K*F(I) + 0.0) GO TO 170
C
KAINZ(KI+3) = KAINZ(KI+3)+1
GO TO 50
C
170 KAINZ(KI+4) = KBINZ(KI+4)+1
GO TO 50
C
180 KAINZ(KI+6) = KBINZ(KI+6)+1
GO TO 50
C
*** CHANGE IN I.D. AREA
C
200 DO 710 I = 1,16
BIKZ = KBIIZ(I)
TZ = KNTZ
KINUC+I(I) = (BIKZ/TZ)*100.
210 CONTINUE
C
*** SAVE ZONE VALUES IN REGION ACCUMULATORS
C
DO 220 I = 1,16
KBIUR(I) = KBIUR(I)+KBIIZ(I)
220 CONTINUE
C
*** CALL REPORT WITH ITOTYP=1 FOR ZONE PRINT
C
ITOTYP = 1
CALL REPS3
C
*** SAVE ZONE VALUES IN REGION ACCUMULATORS
C
DO 220 I = 1,16
KBIUR(I) = KBIUR(I)+KBIIZ(I)
220 CONTINUE
C
KTH = KTH+KNTZ
C
*** ZERO OUT ZONE VALUES
C
KNTZ = 0
DO 250 I = 1,16
250 KBIIZ(I) = 0
C
IF (IH.LT.U) GO TO 240
C
ZINF = ZZONE
IF (1REGION,EQ.,REGION) GO TO 120
C
240 DO 250 I = 1,16
BIUR = KBIUR(I)
TZ = KNTZ
KINUC+I(I) = (BIUR/TZ)*100.
250 CONTINUE
C
NSEGS = KTH
C
*** CALL REPORT WITH ITOTYP=2 FOR REGION PRINT
C
ITOTYP = 2
CALL REPS3
C
*** SAVE REGION VALUES IN COUNTRY ACCUMULATORS
C
DO 260 I = 1,16
260 KBIURC(I) = KBIUC(I)+KBIUR(I)
C      KNTC = KNTC+KNTR
C  *** ZERO OUT REGION VALUES
C      KNTR = 0
  DO 270 I = 1,16
      KNTRCH(I) = 0
C  IF (IH,LT.0) GO TO 300
C      REG5UN = INEG5
      GO TO 120
C  *** MOVE COUNTRY VALUES TO PRINT BUFFER
C  DO 320 I = 1,16
      TC = KNTC
      BI0C = BI0C(I)
      BI0CRI(I) = (BI0C/TC)*100.
  CONTINUE
C      NSC6S = KNTR
C  *** CALL REPORT WITH ITOTYP=3 FOR COUNTRY
C      ITOTYP = 3
      CALL REPT3
C  *** RETURN TO DRIVER PROGRAM
C      RETURN
C      EN
SUBROUTINE FZULU(IDATE, IOUT) MOD3
C ***************************************************************
C ROUTINE FZULU CALL FZULU (IDATE, IOUT)
C PURPOSE TO CONVERT ZULU DATE TO YEAR MONTH AND DAY
C LINKAGE CALL FZULU (IDATE, IOUT)
   IN(1)*CALC-YEAR-INTEGER ZULU DATE
   IN(2)*CALENDAR DATE+DIMENSION 3 FOR INTEGER
   YEAR, MONTH AND DAY RESPECTIVELY
C Routines Called None
C Local Variables:
   IDAYS VECTOR CONTAINING NO, DAYS OF YEAR ON
   LAST DAY OF MONTHS 0 THROUGH 12 FOR NORMAL
   YEAR FOLLOWED BY 13 MONTHS OF LEAP YEAR
   NLEAP*NO, OF LEAP YEARS SINCE 1900 TO SPECIFIED
   DATE
   NFLG*SUBSCRIPT INTO IDAYS VECTOR
   START=1* NOT LEAP YEAR
   START=14* LEAP YEAR
   JDAY*JULIAN DAY OF YEAR
   LOCAL USE
C Comments NO, LOCAL USE

DIMENSION INUT(3) FZULU
DIMENSION IDAYS(365) FZULU
DATA IDAYS/0,31,60,90,120,151,181,212,243,273,304,334,365+/ FZULU
1 0,31,60,91,121,152,182,213,244,274,305,335,366+/ FZULU
000034 IDATE = IDAT + 10203 FZULU
000035 DO 5 I=1,3 FZULU
000036 INUT(I) = 0 FZULU
000037 5 CONTINUE FZULU
000038 CONTINUE FZULU
000039 NLEAP = (1093*IDATE)/1461 FZULU
000040 IOUT(1) = (IDATE-NLEAP)/365 FZULU
000041 C SEE IF CURRENT YEAR IS LEAP YEAR-SET NFLG=1 OR 14 FZULU
000042 1 = IOUT(1)/4 FZULU
000043 I=1+4 FZULU
000044 NFLG=1 FZULU
000045 IF(IOUT(1) =1)10+10+20 FZULU
000046 NFLG=14 FZULU
000047 C JULIAN DAY = ZULU DAY-NO, YEARS*365-NO, LEAP YEARS FZULU
000048 JDAY=IDATE-(IOUT(1)*365)+NLEAP+1 FZULU
000049 C GET DAY NORTH FROM TABLE SEARCH FZULU
000050 20 IF(JDAY=IDAYS(NFLG))50+50+40 FZULU
000051 NFLG=14+1 FZULU
000052 GOTO 50 FZULU
000053 NFLG=14+1 FZULU
000054 50 NFLG=14-1 FZULU
000055 INUT(3)=JDAY-IDAYS(NFLG) FZULU
000056 IF(1104) FZULU
000057 L40 FZULU
SUBROUTINE GENRNG(INCL)
C
C THIS ROUTINE GENERATES THE HISTOGRAM RANGES FROM THE INTERVAL INPUT
C
C *** COMMON STORAGE
C
C COMMON /RANGE / START+INTVL1+BREAK1+INTVL2+BREAK2
C }}/INTVL3+STOP
C
C REAL INTVL1,INTVL2,INTVL3
C COMMON /HSTOGM/XVAL,STUDY,XPSUM,SPSDUM
C */RANGE,INTVL1,INTVL2,INTVL3
C REAL MINVAL,MAXVAL
C **
C REAL RNGHIN,INTVL1,INTVL2,INTVL3
C
C DATA RNGHIN = -1.0,6/4, RNGMAX = 1.0,6/
C 00 10 1 = 1*52
C 10 RANGES(I) = 0.0
C
C RANGE = 0
C RZHNG = 0
C
C *** GENERATE RANGE VALUES
C
C K = 1
C IF (INCL.EQ.0) RANGES(K) = RNGMIN
C IF (INCL.EQ.0) K = 0
C
C ni = (BREAK1-START)/INTVL1
C DO 20 I = 1*ni
C K = *1
C VALU = START+INTVL1*(I-1)
C IF (VALU.LT.0.0) GO TO 20
C IF (VALU.LT.0.0) GO TO 20
C RANGES(K) = 0.0
C RZHNG = K
C K = *1
C 20 RANGES = RANGES+1
C DO 30 I = 1*ni
C K = *1
C VALU = RANGE XINTVL2+BREAK1
C IF (INTVL2.LE.0.0) GO TO 30
C RANGES(K+1) = RANGE1
C GO TO 10
C 30 ni = (BREAK2-BREAK1)/INTVL2
C DO 40 I = 1*ni
C K = *1
C VALU = RANGE1+INTVL2*(I-1)
C IF (INTVL2.LT.0.0) GO TO 40
C IF (VALU.LT.0.0) GO TO 40
C
C END
RANGES(K) = 0
NZANG = K
K = K+1
N RANGE = N RANGE+1
40 RANGES(K) = VALU
N RANGE = N RANGE+1
IF (INTVL3,GE,0.0) GO TO 50
C
RANGES(K+1) = BREAK2
GO TO 100
C
50 N1 = (STOP-BREAK2)/INTVL3
C
DO 60 I = 1,N1
K = K+1
VALU = BREAK2+INTVL3*(I-1)
IF (.ZANG,.LE.0.0) GO TO 60
IF (VALU,.LT.0.0) GO TO 60
RANGES(K) = 0
NZANG = K
K = K+1
N RANGE = N RANGE+1
60 RANGES(K) = VALU
N RANGE = N RANGE+1
RANGES(K+1) = STOP
100 IF (ICL,NE,0.0) GO TO 120
RANGES(K+2) = HIGMAX
N RANGE = N RANGE+2
GO TO 150
120 CONTINUE
C
150 CONTINUE
C
RETURN
C
END
SUBROUTINE LFPA(FDA+LMIC+LYR+ALFM+DAYS)

C MODULE = P1LPFA
C OCTOBER 1, 1973
C JUNE 26, 1973

C MODULE P1LPFA = POINT TARGET PERFORMANCE PREDICTOR

C GIVEN: D.AY+MONTH+YEAR = SUBR, LFPA RETURNS THE RIGHT ASCENSION OF
C GREENWICH AT MIDNIGHT OF A GIVEN DAY.

C REFERENCES:
C EPCH IS 0 HOUR 1 JAN 1950
C DATA PARA 1957.2 1976
C
C COMPUTE DAYS IN FULL YEARS FROM EPCH TO LYR
C DAYS TO 1 JAN 1964 IS 4748.

C
C D.AY=0748
C NOT -LYR MUST BE GREATER THAN OR EQUAL TO (19)64
C LAY=LYR
C DO 20 I=04, LASTYH

C DATA=DAY+FLD=1.
C GO TO 270

C D.AY=0748

C
C 10 D.AYS+DAYS+365
C 20 C.O.N.T.I.N.U.E.
C IF(LYR=1300+40,31)
C 30 IF(LYR=26050,65)
C 40 DATA=DAY+FLDA=1.
C GO TO 270

C D.AY=0748
C DAY=DATA+30
C GO TO 270

C

C 60 KDEP=K.DEP.(LYR-60)+43
C IF(KDEP=40,70)
C DAY=DATA+59
C GO TO 270

C D.AY=0748
C 70 D.AYS+DAYS+58
C 80 IF(LYR=3) 270+170+90
C 90 IF(LYR=5) 190+190+100
C 100 IF(LYR=7) 210+210+110
C 110 IF(LYR=9) 230+230+120
C 120 IF(LYR=11) 250+250+20
C 170 D.AYS+DAYS+FLDA
C GO TO 270

C D.AY=0748
C 180 D.AYS+DAYS+31
C GO TO 270

C D.AY=0748
C 190 D.AYS+DAYS+1.
C GO TO 270

C D.AY=0748
C 200 D.AYS+DAYS+92
C GO TO 270

C D.AY=0748
C 210 D.AYS+DAYS+122
C GO TO 270

C D.AY=0748
C 220 D.AYS+DAYS+153
C GO TO 270

C D.AY=0748
C 230 D.AYS+DAYS+184.
SUBROUTINE MCHIST

*** COMMON STORAGE

COMMON /CARDIN/RTYPE,AUNITS,PARMTHK(5)
    ,ICASIN(5),LEVEL,JPARK,IEHR
INTEGER MPRTYPE,AUNITS,PARMTHK

COMMON /RANGE/START,INVL1,RANGE1+INVL2,BREAK2
    ,INVL3,STOP
REAL INVL1,INVL2,INVL3

COMMON /PRCTL/INPUTNAMES,MAXLIN,KOUT,HEADER(16)
INTEGER HEADER

COMMON /SFLCTN/BIOND(4),MPRTY(4),IPHD(3,14)
INTEGER BIOND,MPRTY

COMMON /FILBUF/FILENAME(2),ICNTRY,INREG,NZONE,ISTRATA
    ,IS USTA,ITCASE,MAXSEG
    ,KRF(165)

DIMENSION KMIN(4),H10PD(4),IPNDAT(14),PDAT(6),XEPN(4),XKEYLD(6)
REAL APVAL(2)

REAL MCREF(111)
INTEGER FILENAM,PDAT1

EQUIVALENCE (KRF(1),NSEG,KT1,KMIN(1))
    , (KRF(2),ICPU,KSTRAT,NHEGS)
    , (KRF(3),XHP,ZTUT)
    , (KRF(4),APE,DU)
    , (KRF(5),XEP,PHI)
    , (KRF(6),XJ10D(1))
    , (KRF(7),RA)
    , (KRF(9),XRTPI)
    , (KRF(9),XCV2)
    , (KRF(10),XIP+1,IPNDAT(1))
    , (KRF(11),XP+11,PDAT(1))
    , (KRF(17),XLP(1))
    , (KRF(21),XKEYLD(1))
    , (KRF(24),APVAL(1),MCREF)
    , (KRF(25),MCREF)
    , (KRF(27),XITL(1))

COMMON /MCHIST/IXVVAL,IXVSTUDY,INPMUNS,INPSUM
    ,IPRANGE,IRANGE(51),INTUL,HANGE(52)
    ,INVAL,MAXVAL,UPRPROD(3),HZRNG,REF
    ,HISTOG
INTEGER UAPPROD

REAL MINVAL,MAXVAL
INTEGER DAIPRO

MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
MCHIST
COMMON /PRNTID, ENTRY, REGION, ZONE, STRATA, ICASEN
*,
FILEN0, FILENAM(2)
*,
ITYP, NAMFT
C INTEGER ENTRY, REGION, ZONE, STRATA, FILEN0, FILENAM
C COMMON /READNG, READNG(5600)
C DIMENSION CLEMA(100), CLEPRO(100), CLATEC(100), CLTECE(100)
*,
CLEMA(100), CLETNC(100)
C EQUIVALENCE (READNG(5001), CLEMA(1)), (READNG(5201), CLATEC(1))
*,
(READNG(5301), CLEPRO(1)), (READNG(5401), CLTECE(1))
C *** LOCAL STORAGE
C DATA CLSTRT, CLIVL1, CLBRK1, CLIVL2, CLBRK2, CLIVL3, CLSIOP*
*,
IPN'T = 0, 2, 0, 0, 0, 0, 0, 0,
NREC = 1
IH = 1
ICL = 0
C *** GFI HEADER RECORD
C CALL FILDA(INH+IPN'T+NREC)
C ICASEN = ICASE
FILNAM(1) = FILENAM(1)
NAMFT = 1
ENTRY = IENTRY
C *** CHECK CASE NUMBERS
DO 50 I = 1, 4
C IF (NAMFT(I).EQ.0) GO TO 50
IF (ICASEN.NE.1) GO TO 50
CALL PATHMN(-4)
WRITE (XOUT, 6500) I, ICASEN(I), ICASE, FILNAM(1)
PARNTR(I) = -NAMFT(I)
50 CONTINUE
C MAXPHP = NAMPHPD
C *** GENERATE RANGE VALUES
C CALL GENRNG(ICL)
C *** LOGP ON PARNTR VALUES
DO 400 I = 1, 4
C IPN'T = 0
JINH = 0
JPHP = 0
C IF (NAMFT(I).LE.0) GO TO 400
C JPHP = I
LEVEL = 1

NRPT = 1
C ******************************************************************************************
C *** ALL MONTE CARLO REPORTS WILL BE PROCESSED AT COUNTRY LEVEL ONLY ***
C LEVEL = 3
C ******************************************************************************************
C *** SET VALUES FOR SPECIAL CASE PARNTH(4)=N
C IF (JPARH.LT.4) GO TO 100
C LEVEL = 3
C NRPT = 0
C STA = CLSTRT
C IN1 = CLIVL1
C IN2 = CLIVL2
C IN3 = CLIVL3
C STOP = CLSTOP
C ICL = 1
C CALL REHINT(ICO)
C ******************************************************************************************
C *** LOOP ON PREDICTIUM POINTS
C 100 CONTINUE
C IF (JYJ+1.LT.0) GU TO 130
C IF (JYJ+1.EQ.0) GO TO 110
C IG = INDWU(IPN+1)
C IF (IG .LT. JYJ+1) GO TO 110
C IF (DIOVU(JYJ+1),EQ,0) GO TO 100
C CALL PAHMO(24)
C WRITE (LICHF,,510) JYJ+1,ICASE,FLAMN(1)
C GO TO 100
C 110 IPNT = IPN+1
C IF (RI0VUG(IG),NE,0) GO TO 150
C GO TO 100
C 120 JN1 = -1
C JN1 = IPN+T
C JN1 = 1
C 130 JYJ+1 = JYJ+1
C IF (IPR0(JYJ+1),EQ,0) GU TO 400
C FLUX = JYJ+1,IPR0)
C FLU0 = IPR0(JYJ+1,IPR0)
LYR = IPRD(1+JPROD)

CALL LPPA(FLDA,LMO,LYR,ALFM,DAYS)

DATE = DAYS

DO 135 INX = JNX+NPD

IF (DATE.EQ.IPRDAT(INX)) GO TO 140

135 CONTINUE

CALL MACHDR(4).

WRITE (KOUT+6520) RTYPE,JPARK,PARTRK(JPARM),(IPRD(L,JPRD),L=1,3)

GO TO 150

160 INPT = JPHI+INX

JNX = INX+1

DO 145 K = 1,3

145 DATA(1) = IPHO(1,JPRD)

GO TO 150

C *** LOOP ON LEVEL VALUE

150 CONTINUE

GO TO (155*160*165)*LEVEL

C *** LEVEL = 1

155 INEC = 14*(IPNT-1)*63

NMAX = NTO

GO TO 170

C *** LEVEL = 2

160 INEC = 4*(IPNT-1)*63

NMAX = NREGS

GO TO 170

C *** LEVEL = 3

165 INEC = 1*(IPNT-1)*63

NMAX = 1

GO TO 170

170 CONTINUE

C *** LOOP ON NUMBER OF REPORTS 0 NMAX VALUE

DO 300 INPT = 1*NRPT

C INITIALIZE ACCUMULATOR VALUES

J = 0

IN = 0

DFLTR = 0.0

KEF = 0.0

UFLS = 0.0

MDTVAL = 1.0

MAXVAL = -1.0

NPEC = INEC

DO 175 IX = 1+51

175 INACUT(IX) = 0
C *** IF CONFIDENCE LEVELS GET ONE SET OF VALUES PER REPORT

C
C DO 200 IDY = 1*MAX
C NREC = NREC + 1
C CALL FILLD(IN,IPNT,NREC)
C IF (JPARM.EQ.0) CALL CLMVE(IRPT)
C REF = PEF+HCREF
C DO 190 IRNG = 1*NT
C J = J+1
C READING(J) = HCLERM(IRNG)
C MINVAL = MIN(MINVAL,READING(J))
C MAXVAL = MAX(MAXVAL,READING(J))
C DELFUNP = HCLERM*READING(J)
C DELSU = DELSU*REAL-HU(J)**2
C 190 CONTINUE
C 200 CONTINUE
C
C *** TALLY PERCENT RANGES

C NTOVL = J
C DO 250 NI = 1*NTOVL
C PCELR = 0.0
C IF (NI.EQ.0) PCELR = 100.0/READING(NI)/REF
C DO 220 K = 1*NHANGE
C IF (PCELR.GE.0.0) GO TO 210
C IRANGE(NZHRANGE) = IRANGE(NZHRANGE)+1
C GO TO 250
C 210 IF (PCELR.GE.RANGES(K+1)) GO TO 220
C IRANGE(K) = IRANGE(K)+1
C GO TO 250
C 220 CONTINUE
C 250 CONTINUE
C
C DATPRD(I) = JMIN
C IF (JMIN.GT.0) GO TO 280
C CALL FZULU(IDATE,DATPRD)
C 280 TJ = NTOVL
C XMVAL = 0.0
C IF (HFF.EQ.0.0) XMVAL = 100.0/(TJ*REF)*HCLERM
C XMVAL = 0.0
C IF (HFF.EQ.0.0) XMVAL = 100.0/(TJ*REF)*HCLERM
C STBLVE = 0.0
C
IF (J .GT. 1, AND, REF .NE. 0, 0) STODAY = 100 / REF 
* #SORT (1 / (TJ = 1) * (DELSP = 1 / TJ * DELERR ** 2)) 
290 CALL FEPRT2 
C 300 CONTINUE 
C IF (LEVEL .EQ. 3) GO TO 350 
C LEVEL = LEVEL + 1 
GO TO 150 
C 350 IF (IPN .LT. MAXPN) GO TO 100 
C 400 CONTINUE 
C 500 RETURN 
C *** ERROR MESSAGES 
C 6500 FORMAT (50H *** ICASIN(I) DOES NOT MATCH CASE NUMBER ON FILE/) 
* 12H ICASIN(I,11,2H) = 14,15H FILE CASE NO. = 14 
* 11H FILE NAME = 3D/1H /IX/10 (6H********) 
C 6510 FORMAT (38H *** REQUESTED BIO=INDO- NOT ON FILE/) 
* 26H BIO-INDO NO. = 11,15H FILE CASE NO. = 14 
C 6520 FORMAT (54H *** INPUT PREDICTION DATE DOES NOT MATCH DATE ON FILE/) 
* 312/1H /IX/18 (6H********)
SUBROUTINE PAGHDR(N)
C ROUTINE CONTROLS COUNT ON PRINTED OUTPUT AND PROVIDES A LINE
C OF PRINTED HEADING ON THE TOP OF EACH OUTPUT PAGE.
C *** COMMON STORAGE
C COMMON /PRCTCL/NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)
INTEGER HEADER
C *** LOCAL STORAGE
C DATA LPRINT=-1/
C IF (LPRINT) 3*4*5
C 3 N1 = MAXLIN-3
C N2 = MAXLIN-1
C LPRINT = 1
C GO TO 30
C 4 LPRINT = -1
C GO TO 30
C 5 IF (N.LE.0) GO TO 10
C J = NLINES+1
C IF (J.GT.N1) GO TO 10
C NLINES = J
C GO TO 30
C 10 CONTINUE
C IF (N.LE.0) GO TO 30
C DO 20 J = NLINES+1,N2
C 20 WRITE (KOUT,100)
C LPRINT = 0
C GO TO 50
C 30 NPAGE = NPAGE+1
C WRITE (KOUT,120) (HEADER(I)+=1+16)+NPAGE
C NLINES = 3+IAUS(4)
C 50 RETURN
C 100 FORMAT (1H)
C 120 FORMAT (1H1,6X,16AB,10/I0)
C END
SUBROUTINE POPURV

C THIS ROUTINE IS THE DRIVER FOR THE POPULATION REPORTS

C

C *** COMMON STORAGE

C COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)

C INTEGER RPTYPE,AUNITS,PARMTR

C

C *** GENERATE RANGES FOR HISTOGRAMS

C ICL = 0

C CALL GENRNG(ICL)

C

C *** PARMTR(1)

C IF (PARMTR(1).EQ.0) GO TO 20

C JPARH = 1

C CALL SAMERR

C IFHR = 0

C

C *** PARMTR(2)

C 20 IF (PARMTR(2).EQ.0) GO TO 40

C JPARH = 2

C CALL CANSEN

C IEHR = 0

C

C *** PARMTR(3)

C 40 IF (PARMTR(3).EQ.0) GO TO 60

C JPARH = 3

C CALL YLDERR

C IFHR = 0

C

C *** PARMTR(4)

C 60 IF (PARMTR(4).EQ.0) GO TO 80

C JPARH = 4

C CALL ANPERR

C IFHR = 0

C

C *** PARMTR(5)

C 80 IF (PARMTR(5).EQ.0) GO TO 100

C JPARH = 5

C CALL RDPERR

C IFHR = 0

C
80 IF (PARMTR(S) .EQ. 0) GO TO 100

C        JPARM = 5
C        CALL A+PERR
C
C        100 RETURN
C        END
**THE POST PROCESSOR PROGRAM**

**COMMON STORAGE**

```
COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5),
   ICASIN(5),LEVEL,JPARM,IERR
INTEGER RPTYPE,AUNITS,PARMTR

COMMON /RANGE/START,INTVL1,BREAK1,INTVL2,BREAK2,
   INTVL3,STOP
REAL INTVL1,INTVL2,INTVL3

COMMON /SELECTN/BIMNU(4),RPRY(4),IPRD(3,14)
```

**LOCAL STATE**

```
DIMENSION IHEADR(12)
DIMENSION RANGE(7),RNGNML(7),INPVLI(15),INPVL2(50)

AMCVT(3),XMCTV(3),ACARD(5),PMNR(4)
```

**LOGICAL INLEPS**

```
LOGITAL INLEPS:
EOUT/LENCE (RASK01(1),START), (INPVLI(1),RPTYPE)
   (INPVLI(1),RNGNML(1))
```

**DATA**

```
DATA IN,OUT /5,6/; NONING /1/; IBLNK IN / LNMAX /45/
DATA (RNGNML(1),E1(1)):500,100,45,100,25,100,500/
DATA MBREAK,MBREAK1,MBREAK2 /124,125,50,7/
DATA (AMCVT(1),E1(1)):2,4,11,3;4,3,7,114,0,2,3,0,7,4,5,
       114,0,0,1,3,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,
DATA (XMCTV(1),E1(1)):2,4,11,3;4,3,7,114,0,2,3,0,7,4,5,
       114,0,0,1,3,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,
DATA (ACARD(1),E1(1)):2,4,11,3;4,3,7,114,0,2,3,0,7,4,5,
       114,0,0,1,3,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,
```

**INITIALIZE INPUT QUANTITIES**

```
DO 10 I = 1,HMOD
   10 IHEADR(I) = IBLNK
```

**GO**

```
GO TO 1,760511, 59113
```

```
DO 20 I = 1,HMNP1
   20 INPVLI(I) = 0
```

```
DO 30 I = 1,HMNEG
   30 RANGE(I) = 0
```

```
DO 40 I = 1,HMNP2
   40 INPVL2(I) = 0
```

```
INPAR = 0
```

```
COMMON /SUBHS/ICSESH
 ```
NLINES = 0
MAXLM = LMAX
KOUT = IUOT
000061
000062
000063
000064
000065
000066
000067
000068
000069
000070
000071
000072
000073
000074
000075
000076
000077
000078
000079
000080
000081
000082
000083
000084
000085
000086
000087
000088
000089
000090
000091
000092
000093
000094
000095
000096
000097
000098
000099
000100
000101
000102
000103
000104
000105
000106
000107
000108
000109
000110
000111
000112
000113
000114
000115
000116
000117
000118
COM 50 \ I = 1+3
C'MATERE(I) = A'MC(w(I)
50 C'THKE(I) = X'HCw(I)
C DO 60 \ I = 1+4
C ON 60 \ J = 1+12
60 HDEAE(I) = HMDP(I)
C *** READ INPUT CONTROL CARDS
C CARD 1 = HEADER CARD
C HEAD (IN+5000) (1HEAER(I),I=1+12),INBL,INCRD
C DO 90 \ J = 1+12
90 HDEAE(I) = IHEAER(I)
C IF (INBL.EQ.,LABEL,AND,INCRD,EQ.,NCARD(I)) GO TO 100
C LABEL ERROR ON HEADER CARD
C INERTS = ',TRUE.
C CALL P,AGHR(3)
C WRITE (NOUT+6000) NCARD(I),INBL,INCRD
C CARD 2 = DATA CARD
C 100 HEAD (IN+5000) RPTYPE,AUNIT,5,(RANGE(I),I=1+7),INBL,INCRD
C IF (INBL.EQ.,LABEL,AND,INCRD,EQ.,NCARD(2)) GO TO 110
C LABEL ENFOR.
C INERTS = ',TRUE.
C CALL P,AGHR(3)
C WRITE (NOUT+6000) NCARD(2),INBL,INCRD
C 110 IF (RPTYPE,GT,0,AND,RPTYPE,LT,MRPTYPE+1) GO TO 115
C RPTYPE VALUE OUT OF RANGE
C INERTS = ',TRUE.
C CALL P,AGHR(2)
C WRITE (NOUT+6010) RPTYPE
C 115 IF (RPTYPE,EQ.,1,OR,RPTYPE,EQ.,2) GO TO 200
C 000119
C GO 120 \ I = 1+12
C IF (CHAGE(I),NE,0) NOWMG = 0
C CONTIUE
C 120 CONTIUE
C IF (NOWMG,NE,0) GO TO 125
C 000122
C GO 122 \ I = 1+12
C 000123
122 RANGE(1) = RGNMUL(1)
GO TO 200

C CHECK FOR INTERVAL RANGE ERRORS

125 CONTINUE
A = 0.
B = 0.
C = 0.
D = 0.
IF (START.LT.BREAK1) GO TO 130

C IF (ERR.NE.0) WRITE (KOUT,6020) START,BREAK1

130 A = BREAK1-START
U = 0. + A/INTVL1
IF (AMOD(U,IINTVL1),EQ,0) GO TO 140

C IF (ERR.NE.0) WRITE (KOUT,6030) A,IINTVL1

140 IF (INTVL2.EQ.0.0) AND,BREAK2,ED.G0) GO TO 180

C IF (ERR.NE.0) WRITE (KOUT,6030) BREAK1,BREAK2

150 B = BREAK2-BREAK1
D = U+INTVL2
IF (AMOD(U+INTVL2),EQ,0) GO TO 160

C IF (ERR.NE.0) WRITE (KOUT,6030) B,INTVL2

160 IF (INTVL3.EQ.0.0) AND,STOP,ED.G0) GO TO 180

C IF (ERR.NE.0) WRITE (KOUT,6030) BREAK2,STOP

170 C = STOP-BREAK2
D = U+C/INTVL3
IF (AMOD(U+C/INTVL3),EQ,0) GO TO 180

C IF (ERR.NE.0) WRITE (KOUT,6030) C,INTVL3

180 CONTINUE
C = 0.
GO TO 130
C CARD 3 - DATA CARD 2
C
C 200 READ (IN,5020) (PARMTR(I),I=1,5), (ICASIN(I),I=1,5)
    ILFL,ICASE,INFL,INCRD
C
C IF (ILFL.LE.ICASE.AND.INCRD.EQ.INCRD(3)) GO TO 205
C
C INERRS = .TRUE.
C CALL PGRDH(3)
C WRITE (KOUT,6000) NCARD(3),INCRD,INCRD
C
C 205 IF (KRTYPE.EQ.1 .OR. RTYPE.EQ.4) GO TO 220
C
C SETALT = 0
C DO 210 I = 1,5
C IF (PARMTR(I),IEQ.0) SETALT = 1
C 210 CONTINUE
C
C IF (SFLALL.EQ.1.) GO TO 218
C DO 215 I = 1,5
C IF (I.LE.5.AND.RTYPE.EQ.3) GO TO 218
C 215 PARMTR(I) = 1
C
C 218 IF (KRTYPE.EQ.3) GO TO 220
C
C IF (LEVEL,EQ.0) LEVEL = JLEVEL
C 220 GO TO 240
C
C IF (I.LE.5.AND.RTYPE.EQ.3) GO TO 250
C IF (KRTYPE.GT.1 .OR. RTYPE.LT.4) GO TO 230
C IF (ICASIN(I),IEQ.0) GO TO 250
C
C INERRS = .TRUE.
C CALL PGRDH(3)
C WRITE (KOUT,6090) 1,1,ICASIN(I),RTYPE
C
C GO TO 250
C 230 IF (PARMTR(I),IEQ.0) GO TO 240
C
C IF (ICASIN(I),IEQ.0) GO TO 240
C
C INERRS = .TRUE.
C CALL PGRDH(3)
C WRITE (KOUT,6100) 1,1,ICASIN(I),KPTYPE,1,PARMTR(I)
C
C 240 CONTINUE
C C CARDS 4 = 5 (DATA CARDS 3 = 4)
C
C 250 READ (IN,530) (HIDUP(I),I=1,9), (KPTYPE(I),I=1,4)
C   (IPNO(J,J),J=1,5),I=1,6), INFL, INCRD
C
C POUT POUT POUT POUT POUT POUT
C POUT POUT POUT POUT POUT POUT
C POUT POUT POUT POUT POUT POUT
C IF (INL.REL.LABEL.AND.INCRO.EQ.NCARD(4)) GO TO 260
C INERRS = .TRUE.
CALL PACHDR(3)
WRITE (KOUT,6000) NCARD(4),INLBL,INCH
C 260 HEAD (IN+5+40) (IPRO(J,J),J=1:3),I=1:14)INLBL,INCH
C IF (INL.REL.LABEL.AND.INCRO.EQ.NCARD(5)) GO TO 270
C INERRS = .TRUE.
CALL PACHDR(3)
WRITE (KOUT,6000) NCARD(5),INLBL,INCH
C *** ECHO PRINT CARD IMAGES
C 270 CONTINUE
C CALL PACHDR(-15)
C WRITE (KOUT,5500) HPYPE,(PAMTR(I),I=1:5),(ICASIN(I),I=1:5),LEVEL
C WRITE (KOUT,5510) RANGE(I),I=1:7
C WRITE (KOUT,5520) HINOH(I),I=1:4),(HPRTY(I),I=1:4)
C WRITE (KOUT,5530) (IPRO(J,J),J=1:3),I=1:14
C *** TERMINATE RUN, IF ERRORS FOUND IN INPUT DATA
C IF (.NOT.INERRS) GO TO 260
C CALL PACHDR(2)
WRITE (KOUT,110)
GO TO 500
C *** RUN CASE
C 260 GO TO (300,350,400,450,500),HPYPE,
C *** SUBSTRATA REFERENCE DATA REPORT
C 300 CALL SUBREF
C GO TO 500
C *** POPULATION REPORTS
C 350 CALL POPUV
C GO TO 500
C *** NPCONE CARLO REPORTS
C 400 CALL NPCINT
C GO TO 500
C *** CALL FREWAC
C 450 CALL FREWAC
C
GO TO 500
C
C *** FINISHED - STOP RUN
C
C
500 STOP
C
C *** INPUT FORMATS
C
500 FORMAT (12d6,2x,4x,A2)
501U FORMAT (12,d11,7F6,1,20x,A4,A2)
502U FORMAT (2x,511,5x,514,2x,11,14,35X,A4,A2)
5130 FORMAT (414,12x,60x,4x,A2)
5030 FORMAT (0532,14x,4X,A2)
5031 FORMAT (7x,16x,H4x,TR,5X,614,2x,514,2x,4X,A2)
19y,12+0,12,14x15,11/10)
5032 FORMAT (44H INTERVAL START INTVL1 BREAK1 INTVL2 BREAK2 *)
15H INTVL3 STOP/INTVL7(F6,1,2X)/110)
5033 FORMAT (7X,14x,7x,412,2x,4X,A2)
5034 FORMAT (7x,14x,7x,412,2x,4X,A2)
5035 FORMAT (7x,14X,1/7)(412,2x,4X,A2)
5036 FORMAT (412,2x,4X,A2)
C
C *** ERROR MESSAGE FORMATS
C
6000 FORMAT (47H *** 1305PROC55 LABEL AND SEQUENCE NUMBER ON POUT*)
* 1H CONTROL CARD NO.* +21H+20H LABEL AND SEQ. NO.*
* 4A+21H *)
6001 FORMAT (30H *** HP3045 VALUE OUT OF RANGE. PPTYPE=+12/11H )
6002 FORMAT (45H *** START VALUE GREATER THAN BREAK1 VALUE.*)
6003 FORMAT (7H START=-FO,1,6H INTVL1=-FO,1/1H )
6030 FORMAT (44H *** BREAK2 VALUE GBR 8.33TH. PPTYPE=+12/11H )
6031 FORMAT (27H *** BREAK2 VALUE GREATER THAN START VALUE.*)
6032 FORMAT (8H BREAK2=+FO,1,AH STOP=-FO,1/1H )
6050 FORMAT (50H *** THE SEGMENT BETWEEN START AND BREAK1 WILL NOT*)
6051 FORMAT (30H *** TIME DIVIDE INTO EQUAL INTVL1 INTERVALS.*)
6052 FORMAT (1H BREAK1=-FO,1,5H INTVL1=-FO,1/1H )
6060 FORMAT (51H *** TIME DIVIDE INTO EQUAL INTVL2 INTERVALS.*)
6061 FORMAT (30H *** TIME DIVIDE INTO EQUAL INTVL2 INTERVALS.*)
6062 FORMAT (15H INTVL2=-FO,1,5H INTVL2=-FO,1/1H )
6070 FORMAT (44H *** THE SEGMENT BETWEEN BREAK2 AND STOP WILL NOT*)
6071 FORMAT (30H *** TIME DIVIDE INTO EQUAL INTVL3 INTERVALS.*)
6072 FORMAT (1H BREAK2=-FO,1,5H INTVL3=-FO,1/1H )
6080 FORMAT (15H *** NUMBER OF HISTOGRAM INTERVALS EXCEEDS THE.*)
6081 FORMAT (1H MAXIMUM OF 500,0502H KILOSTOF0,0 OF INTERVALS THAT*
6082 FORMAT (25H MUST BE TERMINATED=+11H/1H )
6090 FORMAT (128 *** ICAS001,11,502) VALUE MISSING FOR REPORT TYPE.*)
6100 FORMAT (120 *** ICAS001,11,120,14,8H PRTYPE=+12/1H )
6110 FORMAT (120 *** ICAS001,11,120,14,8H PRTYPE=+12/1H )

END
C
C
C
C
C
C
C
SUBROUTINE PUNITS (FACTOR, UNITS)
C *** THIS ROUTINE SETS UP THE PROPER PRINT AND CONVERSION UNITS
C *** FOR THE POPULATION AND MONTE CARLO REPORTS
C *** COMMON STORAGE
C COMMON /CARDIN/RPTYPE,AUNITS, FAKTR(5)
C, CASE(5), LEVEL, JPARM, IEMR
  INTEGER RPTYPE, AUNITS, FAKTR
C COMMON /CONVRT/ CNAME(3), CONVR(3)
C
C *** LOCAL STORAGE
C DIMENSION UNITS(1)
C DIMENSION JFAC(10), IAMERU(9), ITHNCU(9)
C DATA (JFAC(I), I = 1: 10) / 1, 1, 2, 1, 2, 1, 3, 3, 0/ 1
C DATA (IAMEL(I), I = 1: 9) / 6H10**4, 6HACRELS, 6H:
C C DATA (ITHNCU(I), I = 1: 9) / 6H10**4, 6HBUHORSHEM, 6H:
C C IFAC = 0
C IF (RPTYPE.EQ.3) IFAC = IFAC+5
C IFAC = IFAC+JPARM
C C IFAC = JFAC(IFAC)
C IF (AUNITS.NE.0) GO TO 50
C C *** AMERICAN UNITS
C FACTOR = 1.0
C IF (RPTYPE.EQ.3) GO TO 70
C FAKTR = CNAME(IFAC)
C 20 IFAC = (IFAC-1)*3
C C 30 IAMERU(I) = IAMERU(J)
C C 50 IAMERU(1) = IAMERU(3)
C C 70 IAMERU(1) = IAMERU(3)
C C *** SWAN UNITS
C FACTOR = 1.0
C IF (RPTYPE.EQ.3) GO TO 70
C FAKTR = CNAME(IFAC)
C 20 IFAC = (IFAC-1)*3
C C 30 IAMERU(I) = IAMERU(J)
C C 50 IAMERU(1) = IAMERU(3)
C C 70 IAMERU(1) = IAMERU(3)
**SUROUTINE RANACF (IFILE,IREC,BUF,N,IX,L,IOPT)**

- **INPUT PARAMETERS**
  - `FILE`: Logical unit number of the random access file.
  - `IREC`: Record number to be read or written.
  - `BUF`: Array of length `N` to contain the record.
  - `N`: Length of record to be read or written.
  - `IX`: Array of dimension `L` containing the index record.
  - `L`: Length of index.
  - `IOPT`: Entry point option.
    - 0: To open the file.
    - 1: To read the file.
    - 2: To write the file.
    - 1: To close the file.

**DIMENSION**

- `BUF(N)`, `IX(L)`

**LINKAGE**

- Call RANACF (IFILE, IREC, BUF, N, IX, L, IOPT)

**RANACF IS CALLED FROM INPATH, CAS, HRAUDP**

**SUBROUTINES USED**

- OPEN, CLOSE, READ, WRITE

**THE RANDOM ACCESS I/O OPERATIONS**

**IF (IOPT, FG, -1) GO TO 400**

**IF (IOPT = 0) GO TO 300**

**IF (IOPT = 1) GO TO 300**

**IF (IOPT = 2) GO TO 300**

**GO TO 900**

**IF (IOPT = 3) GO TO 300**

**GO TO 900**

**STOP**

**999 FORMAT(1H,91X)*** An irrecoverable I/O error has occurred on read**

**STOP**
SUBROUTINE REPT1

C THIS ROUTINE CONTROLS AND PRINTS THE SUBSTRATA REFERENCE DATA REPORT

C *** COMMON STORAGE

C COMMON //CANDIN//UPTYPE,UNITSS,PARTHK(5)
C * ICASIN(5),LEVEL,PARNM,ERR
C INTEGER RPRTYP,UNITSS,PARTHK

C COMMON //PRINTIL//PAGE,HINES,MAXLIN,KOUT,HEADER(16)
C INTEGER HEADER

C COMMON //CONVT//CIAMER(3),CTRTC(3)

C COMMON //PRINT1//CTRTY,REGION,ZONE,STRATA,ICASEN
C *!, FILENO,FILNH(2)
C *!, INITYP,NAME

C INTEGER CTRTY,RESTU,ZONE,STRATA,FILENO,FILNH

C COMMON //SUBVAL//HMXTHA,KSUBGP(3),NSEGGP(2)*XINES
C *!, XSIGN+XINCN+XSIMCN

C *** LOCAL STORAGE

C LIMITATION FACTOR(3),IAMEFHI(4),IINCROC(4),IUNITSS(4)

C ** UNTISS(3)

C DATA JFIRST /0/, KPAGE /0/
C DATA (I:THUC(1),1+4) /60(THOUS+6HAND HE+6METARES+6H) / KPAGE
C DATA (I:CIAMER(1),1+4) /60(CTRTY-T+6HOUSSAN,6CH+6ENS) / KPAGE
C CHECK FOR FIRST LINE THROUGH

C IF (JFIRST.NE.0) GO TO 100

C SAVE FIRST PAGE NUMBER

C MPAGE = KPAGE+1
C JFIRST = KPAGE

C IF (UNITSS.NE.0) GO TO 50

C AMERICAN UNITS

C DO 30 1 = 1+4
C IF (1.GT.3) GO TO 30
C UNITSS(1) = CIAMER(1)
C 30 UNITSS(1) = IAMEFHI(1)
C GO TO 60

C METRIC UNITS

C 50 UN 60 1 = 1+4
C IF (1.GT.3) GO TO 60
C UNITSS(1) = CIAMER(1)
C 60 UNITSS(1) = IAMEFHI(1)
...
SUBROUTINE REPRT2

** COMMON STORAGE **

COMMON /CARDIN,HPTYPE,AUNITs,PARMTK(5)
* ICASIN(5),LEVEL,PARK+1ERR
INTEGER HPTYPE,AUNITs,PARMTK
COMMON /PRICIL/SPARE,LINES,MAXL+ACUT,HEADER(16)
INTEGER HEADER

** COMMON /PRINTD/ CTRY,REGION,ZONE,STRATA,ICASEN
** FILMN,FILNAM(2)
** ITOT,FILNAM(1)

** INTEGER CTRY,REGION,ZONE,STRATA,FILN,w,FILNAM **

CALL HINTS/V AXIVAL,STDEV,SPRVA+SUF,SUM
* MAXVAL+TXEGL(S1)+TOTL+RANGES(S2)
** MINVAL+MAXVAL+DATPND(3)+NZHNG+REF

PFAL MINVAL+MAXVAL
INTEGER DATPND

** LOCAL STORAGE **

DIMENSION IUNITS(3)

DATA (H(1)+H(4))/6HNCALRO+6HCALO
DATA (H(1)+H(4))/6H POPU+LATION+6H SAMPL+ING ER+6HMROR
* 6H POPU+LATION+6H CAYS+6HERROR+6H
* 6H POPU+LATION+6H YIELD+6H ERROR+6H
* 6H POPU+LATION+6H AREA+6HERROR+6H
* 6H MONT+6H CANLO+6H AREA+6HERROR+6H
* 6H MONT+6H CANL+6H POPU+LATION+6HERROR+6H
* 6H MONT+6H CANLO+6H YIELD+6H ERROR+6H

DATA MFIRST /2/ MFIRST /2/ IF (HPTYPF.EQ.2) IX = JPARH
IF (HPTYPE.EQ.3) IX = HPTYPE+2*(JPARN+1)

CALL PUNITS(FACTOR,IUNITS)

MINVAL = MINVAL+REF+150
MAXVAL = MAXVAL+REF+150
REF = REF+FACTOR

** PRINT HEADINGS **

CALL PANGOn(4)
IF (IX+J+1) GO TO 100
RILL (HUNIT,IRON) (HPTYPE(1+1X),1+15),ICASEN,FILNAM(1),FILNAM(2)
L2 = LOC12*(I-1)
L3 = LOC3*(I-1)

C WRITE (KOUT+1150) L1+FVAL(L1)*SUMFVL(L1)+L2+FVAL(L2)*SUMFVL(L2)+
     L3+FVAL(L3)*SUMFVL(L3)

C 250 CONTINUE
     IF(IX,,F4,,916) GO TO 290
     L1 = L1+1
     L2 = L2+1
     L3 = L3+1

C IF (L10,L2,L3) GO TO 270
     IF (L10,L2,L3) GO TO 280

C WRITE (KOUT+1150) L1+FVAL(L1)*SUMFVL(L1)+L2+FVAL(L2)*SUMFVL(L2)+
     L3+FVAL(L3)*SUMFVL(L3)
     GO TO 290

C 270 WRITE (KOUT+1165) L1+FVAL(L1)*SUMFVL(L1)
     GO TO 280

C 280 WRITE (KOUT+1177) L1+FVAL(L1)*SUMFVL(L1)+L2+FVAL(L2)*SUMFVL(L2)
     GO TO 290

C WRITE (KOUT+1200) (IDTYPE(I+JX)+I-1+2)*XMVAL+STDDEV*REF
     (LOW+311(I)+1)*3

C GO TO 270 IF (HTYPF,FO,,3) GO TO 300

C WRITE (KOUT+1052) XPS3+PS3
     WRITE (KOUT+1050) XVAL+MAXVAL+INTOL

C WRITE (KOUT+1190) CHAIN
     IF (HTYPF,FO,,3) GO TO 310

C WRITE (KOUT+1110) REGION
     IF (LEVEL,S=,3) WRITE (KOUT+1110) REGION
     IF (LEVEL,S=,1) WRITE (KOUT+1120) ZONE
     GO TO 290

C 310 IF (LEVEL,S=,1) WRITE (KOUT+1190)
     IF (LEVEL,S=,2) WRITE (KOUT+1185)
     IF (LEVEL,S=,3) WRITE (KOUT+1190)
     IF (LOW+H,,3) WRITE (KOUT+1190)

C 320 CONTINUE
     IF (IX,,E,,1,,G,,DATPRD,,1,,E,,G,,) GO TO 500
     IF (DATPRD,,1,,G,,T,,1,,3) WRITE (KOUT+1135) DATPRD,,1,,G,,T,,1,,3
     IF (DATPRD,,1,,G,,T,,4,,9) WRITE (KOUT+1135) DATPRD,,1,,G,,T,,4,,9

C * WRITE (KOUT+1130) DATPRD,,2,,G,,T,,1,,3,,G,,T,,4,,9

C 500 RETURN

C *** FORMAT

C 1000 FORMAT (19X,S5,,A6,,/12H CASE NUMBER=16,,/6H FILE NAME=4X,,A4,,/1H )

C 1010 FORMAT (19X,S5,,A6,,/12H CASE NUMBER=16,,/6H FILE NAME=4X,,A4,,/1H )

C 1020 FORMAT (19X,,/1H AREA OF+3,,A6,,+1,,A,,X,, I,,J,,+10,,H LESS THAN,,F,,4,,3)
     (1**F,,3,,A,,X,, T,,O,,F,,3,,)
     (1,,/F,,3,,A,,X,, T,,O,,F,,3,,)

C 1025 FORMAT (19X,,/1H AREA OF+3,,A6,,+1,,A,,X,, I,,J,,+10,,H GR. THAN,,F,,10,,3)
     12,,/F,,3,,A,,X,, T,,O,,F,,3,,)

C 1030 FORMAT (19X,,/1H AREA OF+3,,A6,,+1,,A,,X,, I,,J,,+10,,H GR. THAN,,F,,10,,3)
     14,,/F,,3,,A,,X,, T,,O,,F,,3,,)

C 1040 FORMAT (19X,,/1H AREA OF+3,,A6,,+1,,A,,X,, I,,J,,+10,,H GR. THAN,,F,,10,,3)
     16,,/F,,3,,A,,X,, T,,O,,F,,3,,)

C 1050 FORMAT (19X,,/1H AREA OF+3,,A6,,+1,,A,,X,, I,,J,,+10,,H GR. THAN,,F,,10,,3)
     18,,/F,,3,,A,,X,, T,,O,,F,,3,,)

C 1060 FORMAT (19X,,/1H AREA OF+3,,A6,,+1,,A,,X,, I,,J,,+10,,H GR. THAN,,F,,10,,3)
     20,,/F,,3,,A,,X,, T,,O,,F,,3,,)
1070 FORMAT (11H DATA TYPE + 2A6+5H MEAN+F8.3+22H (PCT) STD. D. IATION+MODF1
* F8.3,18H (PCT) KPP. VALUE+F11.3+1X+3A6+1H )
1080 FORMAT (24X+22H SUM OF POPULATION SUM+F8.3+)
* 22H (PCT) STD. DEVIATION+
1090 FORMAT (24X+15H MAXIMUM READING+F8.3+23H (PCT) MAXIMUM READING) NDF1
* F8.3,20H (PCT) NUMBER OF READINGS+16/1H )
1100 FORMAT (23X+12H COUNTRY +AG)
1110 FORMAT (11H+4X+5A8+8H )
1120 FORMAT (1H+8X+4PRT+1H)
1130 FORMAT (1H+4X+15H SUGGESTION DATE+14+1H/+12+1H/+12,)
1140 FORMAT (6H+1H+8X+17H+12H+1PRT NUMBER+12)
1150 FORMAT (1X+15H HISTOGRAM DATA +5+14+1H+F9.3+F10.3))
1160 FORMAT (1X+15H HISTOGRAM DATA ++)
1170 FORMAT (5X+1H+F9.3+F10.3+)
1180 FORMAT (5X+1H+F9.3+F10.3+)
1190 FORMAT (1H++4A+1H++ REGIO DATA ++)
1200 FORMAT (1H++48X+20H++ COUNTRY DATA ++)
C END
C THIS ROUTINE PROCESSES THE FREQUENCY-UP, SAMPLE SEGMENTS ACQUISITIONS
C
C DIMENSION OF AREA SEGMENTS
C
C DO THE CHECKS FOR THE FIRST RUN
C
C IF (JFIRST .NE. 0) GO TO 100
C
C IF (PAGE .LE. 100) GO TO 100
C
C PRINT PAGE AS A STARTER
C
CALL PARM(4)
WRITE (KOUT+1000) ICASEN,FILNAM(1),FILNAM(2)

C PRINT TOTAL LINE
100 GO TO (150,200,300) IOTYP

C PRINT ZONE LINE
150 CALL PARM(3)

C IF (PAGE .LE. 90) GO TO 100
C
C WRITE (KOUT+1000) ICASEN,FILNAM(1),FILNAM(2)

C GO TO 500

C PRINT REGION LINE

200 CALL PARM(6)
SUBROUTINE SAMERR
C
C *** CUT IF E REPORTS THE POPULATION SAMPLING ERROR REPORT
C
C COMMON STORAGE
C
COMMON /CARDIN/HPTYPE,AUNITS,PARMTH(S),
* ICASIN(5),LEVEL,JPARM,INCR
C INTEGER HPTYPE,AUNITS,PARMTH(S)
C
COMMON /RANGE / START,INVL1,BREAK1,INVL2,BREAK2
* INVL3,STOP
C
REAL INVL1,INVL2,INVL3
C
COMMON /FILBUF/ FILENM(2),ICNTNY,IREGN,IZONE,ISTRTA
* IISLST,ICASE,MXSLG
C
KBUF(125)
C
DIMENSION KNH(4),IPHADT(14),PHDATE(6),XEPH(4),XYLD(6)
C
C
APVAL(7)
C
MCERH(100)
C
REAL MCHEF,MCERH
C
INTEGER FILENM,PHDATE
C
EQUIVANCE (KBUF(1),ISEGNT,KMIN(1))
* (KBUF(2),IUPIN,ISTRAT,NGES)
* (KBUF(3),XHAP,INTOT)
* (KBUF(4),XRVT,REVT)
* (KBUF(5),XTPR,TPR)
* (KBUF(6),XHTP,HTP)
* (KBUF(7),XVCP)
* (KBUF(8),XVTP)
* (KBUF(9),XVPR,VP)
* (KBUF(10),XSTRAT,IPRCT)
* (KBUF(11),XHADT,DATE)
* (KBUF(12),XEPH,EPH)
* (KBUF(13),XYLD,YL)
* (KBUF(14),APVAL,MCHEF)
* (KBUF(15),ICERT)
C
C
(NEW)
STHTA = ISTHTA
SOSTHA = ISTHTA

GO TO 120

C

C TEST FOR CHANGE IN I.O. AREA

100 IF (ISTHTA .NE. SOSTHA .OR. ISTHTA .NE. STRATA) GO TO 200
105 IF (ZONE .NE. ZONE) GO TO 200
106 IF (H Điểm .NE. REGION) GO TO 200
110 IF (CHIEN, .NE. CHIEN) GO TO 200
120 DELSU = DELSU + (XTPI - XTPH)

C

GO TO 50

C

C *** CHANGE IN I.O. AREA

200 CONTINUE
205 IF (XI .EQ. 0) GO TO 210
210 XI = K
215 XR = AREA/(XI*AREAAPS)
220 X'R = XI*X'R*AREAAPS

C

J = J+1

C

HEADING(J) = DELSU/XM*XMR

C

MINVAL = AMIN(MINVAL, HEADING(J))
MAXVAL = AMAX(MAXVAL, HEADING(J))

C

DELEHN = DELPH + HEADING(J)
REF = REF + XTPH*AREAAPS
DEL3H = DELSU + HEADING(J)**2

C

210 CONTINUE
220 DELSU = 0
225 XM = 0

C

IF (I .LT. 0) GO TO 250
230 GO TO (225, 230, 250) + LEVEL

C

220 IF (ZONE .NE. ZONE) GO TO 250

C

230 IF (H Điểm .NE. REGION) GO TO 50

C

240 IF (I .EQ. 0) GO TO 50

C

250 IF (J .NE. 0) GO TO 400

C

GO 300 I = 1, N1OTL

C

IF (I .GE. 1 .AND. I .LT. N1OTL)
305 IF (DELEHN .GT. 0.0) PCXPR = 0.0
310 IF (DELEHN .GT. 0.0) PCXPR = 100.0*HEADING(I)/REF
315 IF (PCXPR .LT. 11.0) GO TO 260

C

260
C  IRANGE(MZRNG) = IRANGE(MZRNG)+1
    GO TO 300
C  260 IF (PCDEMR*GE*IRANGES(K+1)) GO TO 280
  IRANGE(K) = IRANGE(K)+1
    GO TO 300
C
280 CONTINUE
C  300 CONTINUE
C
    TJ = INTL
    XMVAL = 0.0
    IF(HEF .NE. 0.0) XMVAL = 100.0/(TJ*REF)*DELLERR
    STODEV = 0.0
    IF(HEF*HEF .GT. 1.0) GO TO 390
    IF(J .LT. 1.0) GO TO 390
    STODEV = 100.0/REF*SQRT(1.0/(TJ-1.0))
    390 XPSUM = TJ*XVAL
    SAMSUM = SAMSUM + STODEV*TJ
    CALL REPT2
C  400 IF(HEF .EQ. 0.0) GO TO 5
C  500 RETURN
C
END

ORIGINAL PAGE IS OF POOR QUALITY
SUBROUTINE SUBREF

C THIS ROUTINE PRODUCES THE SUBSTRATA REFERENCE DATA REPORT

C COMMON STORAGE

C COMMON /FILBUF/ FILENM(2),ICTRY,IREG,IZONE,ISTATA
C          ,ISUST,ISCASE,IXSREG
C          ,IAVF(2),IKRF(165)

C DIMENSION KN(4),IHDRM(4),IPRDAT(14),PDTRDATE(6),XEPH(4),XKEYLD(6)
C          ,MCREF(140)

C REAL MCREF,MCRE,
C INTEGER FILNUPRDATE

C EQUIVALENCE (SBUF(1)*NSEG+NT,KIN(1))
C          , (KIN(2),12*MIPAT,RREGS)
C          , (KIN(3),XHPR+NZDLT)
C          , (KIN(4),AMI+NB)
C          , (KIN(5),XTR+XIU)
C          , (KIN(6),XIVHND(1))
C          , (KIN(7),XIVI)
C          , (KIN(8),XIVP)
C          , (KIN(9),XCP)
C          , (KIN(10),XDP+?IPRDAT(1))
C          , (KIN(11),PDTRDATE(1))
C          , (KIN(12),XEPH(1))
C          , (KIN(21),XKEYLD(1))
C          , (KIN(24),XAVAL(1),MCREF)
C          , (KIN(25),XCHP(1))
C          , (KIN(27),XAVLD(1))

C COMMON /PRNTID/ LNTY,RFILN,ZONE,STRATA,ICASEN
C          , FLENO,FILNAM(2)
C          , ITUTYP,NAMEPI

C INTEGER CTRY,RFILN,ZONE,STRATA,FLENO,FILNAM

C COMMON /SUBVAL/ MA=IVAC,NSGPPS(3)+NSGPPZ(2)+XINBS
C          , ASIHSV,XINCV,XINCV

C *** LOCAL STORAGE

C DIMENSION SVAL(11),ZVAL(11),RVAL(11),CVAL(11)
C          , XSMCV(2),NSGPPS(3)+NSGPPZ(2)+NSGPPZ(2)
C          , RSGPP(3),KSSGMP(2)+KSSGMP(3)+KSGLP(3)

C EQUIVALENCE
C          , (SVAL(1),MASS), (SVAL(2),THAS), (SVAL(3),XINSS)
C          , (SVAL(4),XINCVS)+SVAL(5),XSMCV(1)
C          , (SVAL(7),NSGPPS(1)), (SVAL(10),NSGPPS(1))
C          , (SVAL(1),XHAR), (SVAL(2),XHAR), (SVAL(3),XINUS)
C          , (SVAL(4),XPHS), (SVAL(5),XICV), (SVAL(6),XSCV)
C          , (SVAL(7),XNCVZ(1)), (SVAL(10),XSGPPZ(1))
** (RVVAL(1)+HVAL) * (RVVAL(2)+XVAL) * (RVVAL(3)+XINSH)
** (RVVAL(4)+XSHIFT) * (RVVAL(5)+XINCVM) * (RVVAL(6)+XSHC)
** (RVVAL(7)+XSGPP(1)) * (RVVAL(10)+XSGPP(1))

C

** (CVVAL(1)+HAC) * (CVVAL(2)+IAC) * (CVVAL(3)+XINSC)
** (CVVAL(4)+XSHIFT) * (CVVAL(5)+XINCVC) * (CVVAL(6)+XSHC)
** (CVVAL(7)+XSGPPC(1)) * (CVVAL(10)+XSGPPC(1))

DATA AREAS /10299.17/

IM = 1
ASSTA = 0
J = 0

C

GO TO 1 = 1+11
RVVAL(1) = 0
ZVAL(T) = 0
HVAL(T) = 0

20 CVVAL(T) = 0:

C

GET HEADER RECORD INFO.

C

CALL FILE('IP')

C

ICASEP = ICASE
IM = 0

C

GET TRAIL RECORD

C

50 CALL FILE('I2')

C

IF (IN,L.T.0) GO TO 200
KSSSTA = +KSSSTA+1

C

IF (KSSSTA.GT.1) GO TO 100

C

CVTRY = ICNTRY
KFLTRV = INRECN
ZMT = IZMT
STATA = ISTATA

C

GO TO 120

C

TEST FOR CHANGE IN IDENTIFICATION AREA

C

100 IF (ISTATA.GT.STNATA) GO TO 200
IF (IN.LT.LP.LT.00) GO TO 200
IF (IN.GE.LP.GT.REGION) GO TO 200

C

120 XHA = 1A
XH = 1
XP = AREA/(XHA*AREA)
XIR = XIR*AREA

C

HTK = XHR*YHR
HRA = XHA+HTK

C

TVAK = XHR*TP
TVAK = TVAK*TVAK

C

NSGPS(IGPDO) = NSGPS(IGPDO)+1
IF (IN1111,LT,1) NSGPS(IGPDO) = NSGPS(IGPDO)+NSEG

C

XINSS = XINTP+XHAK+XINSS
C XINCVS = (XCVZ + THAK) * 2 * XINCVS
C XSHCV = THAK - HA XA
C XSHCVS(i) = XSHCVS(i) + XSMCV * 2
C XSHCVS(2) = XSHCVS(2) + XSHCV
C J = J + 1
C GO TO 50
C CHANGE IN I.D. AREA
C 200 H'A = HAAS
C T'A = T'A
C X1'MS = X1'MSS / HA
C XS'MS = (X1'MS / XHA) / HA
C X1'CV = SMX'TINCVS' / TWA
C XSMCV = S.C.
C IF (XSMCVS(2) * 2 / J - XSMCVS(1)) .GT. 0.0 GO TO 205
C IF (J .LT. 2) GO TO 205
C C = J
C GO TO 205
C CONTINUE
C 205 UN 210 T = 1.3
C IF (1.1 .LE. T) GO TO 210
C NSBGPZ(I) = NSBGPZ(I)
C 210 NSBGPZ(I) = NSBGPZ(I)
C CALL REPT1 WITH ITOTYP=1 FOR STRATA PRINT
C ITOTYP = 1
C CALL REPT1
C X1'MS = X1'MSS / HA
C XS'MS = XS'MSS / HA
C X1'CV = X1'CV / TWA
C XSMCV = XSMCV / TWA
C SAVE STRATA VALUES IN ZONE, ACCUMULATORS
C H'A = H'A + H'A
C T'A = T'A + T'A
C X1'MS = X1'MSS + X1'MS
C XS'MS = XS'MSS + XS'MS
C X1'CV = X1'CV + X1'CV
C XSMCV = XSMCV + XSMCV
C UN 220 I = 1.3
C IF (1.1 .LE. T) GO TO 220
C NSBGPZ(I) = NSBGPZ(I) + NSBGPZ(I)
C 220 NSBGPZ(I) = NSBGPZ(I) + NSBGPZ(I)
C C ZERO OUT STRATA ACCUMULATOR VALUES
C UN 230 I = 1.11
C 230 SVVAL(I) = 0
C J = J
C IF (J .LE. 0) STRATA = STRATA
C C UPSRT STRATA I.D.
C TEST FOR CHANGE IN ZONE I.D. OR END-OF-FILE
C IF (IH.LT.0) GO TO 240
C IF (IZONE.GT.ZONE) GO TO 240
IF (IREG.GT.REGION) GO TO 120
C
240 HMA = H+AZ
TAA = THAZ
XHM = XTSZ/HMA
XSM = XSPS/HMA
XICV = XTKCV/TAA
XICV = XSMCV/TAA
C
DO 250 I = 1,3
IF (I.EQ.3) GO TO 250
NSEGPR(I) = NSGP2(I)
250 NSGP2(I) = NSGP2(I)
C CALL REPHT WITH ITYP2 FOR ZONE PRINT
C ITYP2 = 2
CALL REHPT
C SAVE ZONE VALUES IN REGION ACCUMULATORS
C HMA = HMA+HMA
TAA = THA+TAA
XHM = XHM+XHSZ
XSM = XSM+XSMPS
XICV = XICV+XICVZ
XICV = XSMCV+XSMCV
C
DO 260 I = 1,3
IF (I.EQ.3) GO TO 260
NSGP2(I) = NSGP2(I)+NSGPR(I)
260 NSGP2(I) = NSGP2(I)+NSGPR(I)
C ZERO OUT ZONE ACCUMULATOR VALUES
C DO 270 I = 1,11
270 ZVAL(I) = 0
C RESET ZONE I.D.
C IF (IH.EQ.0) ZONE = IZONE
C TEST FOR CHANGE IN REGION I.D. OR END-OF-FILE
C IF (IH.LT.0) GO TO 280
C IF (IREG.GT.REGION) GO TO 120
C
280 HMA = H+AZ
TAA = THAZ
XHM = XHM+XHSZ
XSM = XSM+XSMPS
XICV = XICV+XICVZ
XICV = XSMCV+XSMCV
C
C  DD 290  I = 1,3
C     IF (1.E0,.3) GO TO 290
C     NSSEGPP(I) = NSSEGRR(I)
C     290  NSUPGR(I) = NSSEGRR(I)
C     CALL REPRT1 WITH ITOTYP=3 FOR REGION PRINT
C     ITOTYP = 3
C     CALL REPRT1
C     SAVE REGION VALUES IN COUNTRY ACCUMULATORS
C     HMA = H+HMA
C     TM = TM+TA
C     XHNSC = XINSC + XHNSR
C     XSNSC = XSNSC + XSNSR
C     XHNVC = XINVC + XHNCV
C     XSNCV = XSNVC + XSNCV
C     DO 300  I = 1,3
C     IF (1.EG,.3) GO TO 300
C     NSUPGR(I) = NSSEGRR(I) + NSSEGPP(I)
C     300  NSSEGPP(I) = NSSEGRR(I) + NSSEGPP(I)
C     ZERO OUT REGION ACCUMULATOR VALUES
C     DO 310  I = 1,11
C     310  HVALC(I) = 0
C     TEST FOR END-OF-FILE OR RESET REGION I.D.
C     IF (IH,L1,.0) GO TO 320
C     REGION = IRGN
C     GO TO 120
C     MOVE COUNTRY VALUES TO PRINT AREA
C     320  HMA = HMA
C     TM = TM
C     XHNS = XHNNSC/HMA
C     XSNS = XSNNSC/HMA
C     XHNVC = XHNVCT/MA
C     XSNCV = XSNVC/TMA
C     DO 330  I = 1,3
C     IF (1.EG,.3) GO TO 330
C     NSSEGPP(I) = NSSEGRR(I)
C     330  NSSEGPP(I) = NSSEGRR(I)
C     CALL REPRT1 WITH ITOTYP=4 FOR COUNTRY PRINT
C     ITOTYP = 4
C     CALL REPRT1
C     RETURN
C     END
SUBROUTINE YLDERR

*** ROUTINE PRODUCES THE POPULATION YIELD ERROR REPORT(8)

*** COMMON STORAGE

COMMON /CARIN/RPTYPE,AUNITSPARMK(5)
*, ICASIN(5),LEVEL,SPM,IERR

INTEGER RPTYPE,AUNITSPARMK(5)

COMMON /RANGE/START,INTVL1,BREAK1,INTVL2,BREAK2
*, INTVL3,STOP

COMMON /HF1/ INTVL1,INTVL2,INTVL3

COMMON /FILEUF/ FILEUNM(2),INTERH,IPRINT,IZONE,ISTRTA
*, ISDATA,ICASE,MASEG

*** COMMON /RHUF/ RHUF(125)

DIMENSION KIN(8),IMHD(4),IPHDAT(14),PRDATE(6),XEPH(4),XELYD(6)

APVAL(2)
*, MCEF(100)

REAL NCRF,NREF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MONDF

FILEUF MOND
INTEGER HEADREP
C COMMON /SFLCTN, BIOD(4), WPTY(4), IPRT(3+4)
C INTEGER BIOD(4), WPTY(4)
C COMMON /MATCH/, MATCH
C COMMON /READNG/, READNG(5600)
C DIMENSION CLEHA(100), CLEPRD(100), CLATEC(100), CLPTEC(100)
C EQUIVALENCE (READNG(5001), CLEHA(1))
C EQUIVALENCE (READNG(5001), CLEPRD(1))
C EQUIVALENCE (READNG(5001), CLATEC(1))
C EQUIVALENCE (READNG(5001), CLPTEC(1))
C *** LOCAL STORAGE
C DATA AHFAAS /10249, 712/
C JOIN = 0
C JPRD = 0
C *** PROCESS FOR BIO-WINDOWS FIRST
C 10 IN = 1
C IF (JOIN .LT. 10) GO TO 30
C 15 JOIN = JOIN + 1
C IF (JOIN .GT. 4) GO TO 20
C IF (BIOD(JOIN) .LT. 0) GO TO 50
C GO TO 15
C 20 JOIN = -1
C 30 JPRD = JPRD + 1
C IF (JPRD .GT. 14) GO TO 500
C IF (IPRT(1) .EQ. 0) GO TO 500
C 50 DO 60 I = 1, 500
C 60 READNG(I) = 0.0
C J = 0
C NKS = 0
C NTUL = 0
C 70 DO 90 I = 1, 51
C 80 IF (READNG(I) = 0)
C 90 RFSRA = 0.0
C 100 LLLSTX = 0.0
C 110 DELKR = 0.0
C 120 LFLSTG = 0.0
C 130 RFF = 0.0
C 140 MINVAL = -1.1E10
C 150 MAXVAL = 1.1E10
C IF ( IN .LT. E-10 ) GO TO 80
C 80 CONTINUE
C *** UTILITY PICTURES
CALL FILSER(1M)

IF (IERE,NE,0) GO TO 500

ICASE = ICASE
FILNAM(1) = FILENM(1)
FILNAM(2) = FILENM(2)
IRNFT = 0

IM = 0

*** GET DETAIL RECORDS

80 CALL FILSER(1M)

IF (IM,L.T,0) GO TO 200
IF (XYS .GT, 0 ) GO TO 100
IF (MATCH .EQ., -2) GO TO 210

ENTRY = ICENTRY
FWINM = INFM
XVF = IZONE
STRATA = INSTRATA

GO TO 126

*** TEST FOR CHANGE IN ID AREA

100 IF ( INSTRATA .NE. STRATA ) GO TO 200

GO TO (105,110,120), LEVEL

GO TO (121,122), LEVEL

105 IF ( IZONE .NE. ZONE LR, IREGN .NE. REGION ) GO TO 200

110 IF ( INFM .NE. REGION ) GO TO 200

120 IF ( IZONE .GT. 0 ) GO TO 100

IF (JPNL.LT.0) GO TO 150

GO 150 I = 1:6
IX = K-(I-1)
IF (PKNAT(IX),EQ,0) GO TO 130
IDATE = PKNAT(IX)

GO TO 180

CONTINUE

130

*** PREDICTION DATE INPUT

150 FLOA = IPRO(3*JPRD)

H = IPRO(2*JPRD)

LHY = IPRO(1*JPRD)

CALL LPFA(FLOA,LHY,LHY,ALFOM,DAYS)

IFATE = DAYS
DO 160 I = 1,6
IX = 6-(I-1)   
IF (PHDATE(I)) .EQ. 0) GO TO 160
IF (PHDATE(I) .LE. IDATE) GO TO 160
160 CONTINUE
C
C *** NO DATE MATCH
C
CALL PAGHOM(-2)
WRITE (KOUT+300) WPTYPE,JPARM,PARMTR(JPARM),(IPRO(I,JPRD),I=1,3)

C
GO TO 30
180 DELSTA = DELSTA*XTP*XNR
REFSTA = REFSTA*XTP*XNR*XYLD
KXS = KXS+1.
C
GO TO 80

C
C *** CHANGE IN ID AREA
C
200 IF (KXS .EQ. 0) GO TO 210
J = J+1
READG(J) = (XYLD(IX)-XYLD)*DELSTA
210 CONTINUE
C
C
C
C
DELSA = 0
REFSTA = 0
KXS = 0
IF (IH .LT. 0) GO TO 250
C
GO TO (220,230,240) LEVEL
220 IF (IZONE .NE. ZONE) GO TO 250
230 IF (IPREG,EN_REGION) GO TO 80
GO TO 250
240 IF (IH,FQ,0) GO TO 80
C
250 IF (J .EQ. 0) GO TO 400
MTOL = J
C
CD 200 I = 1,MTOL
PCDFAP = 0.0
C
200 CONTINUE
C
CD 200 K = 1,MTOL
IF (MTOL .LT. 0) PCDERR = 100.0*READNG(I)/REF
C
C
C
C
C
C
VAMF = VAMF(MZMG) = VAMF(MZMG)+1
GO TO 300
C 260 IF (MODEFR,GE,RANGES(K+1)) GO TO 280
C
C 270 CONTINUE
C 300 CONTINUE
C
C DATAGF(I) = JMIN
C IF (JGTJ, GT, 6) GO TO 310
C CALL F2JLU(DATE, DATPRO)
C 310 CONTINUE
C
C TJ = TOTAL
C XVAL = 0.0
C IF (REF .EQ. 0.0) XVAL = 100.0/(TJ*REF)*DELERR
C
C STDEV = 0.0
C IF(DELERR**2/TJ + GTMDELSQ)GO TO 390
C IF (J .GT. 1 .AND. RFF .NE. 0.0 ) STDEV = 100.0/REF*
C * TJ = ( 1.0/(TJ-1.0)*MDELSQ-1.0/TJ*DELEFR**2 )
C 390 XPSUM = TFXVAL
C SPSUM = SUM(TJ)/STDEV
C
C CALL REPRT2
C
C IF ( J * .EQ. 0 ) GO TO 50
C GO TO 10
C 400 CONTINUE
C IF ( JGTJ .LT. 0 ) GO TO 420
C CALL PARMDR (-2)
C WRITE (POINT+6310) HTYPE, JPARM, PARMTR(JPARM), JMIN
C IF ( J * .EQ. 0 ) GO TO 50
C GO TO 10
C 420 CALL PARMDR (-2)
C WRITE (654+50) HTYPE, JPARM, PARMTR(JPARM), JMIN
C IF ( J * .EQ. 0 ) GO TO 50
C GO TO 10
C 500 RETURN
C
C *** ERROR MESSAGE
C
C 6300 FORMAT (5AH *** INPUT PREDICTION DATE DOES NOT MATCH DATE ON FILE/ YLDERR
C * 12H YLDFERR
C * 3L2)
C 6310 FORMAT (43H *** 10 INPUT PREDICTION FOR 810-WINDOW SELECTED/ YLDERR
C * 12H YLDFERR
C * 11)
C END CUR