FRI-UW-8406
August 1984

Skagit River Interim Agreement Studies

Volume III
(Limited Distribution)

Instream Flow Fish Habitat Analysis

Supplemental Programs to IFG PHABSIM System

by

K. Swanson and S. C. Crumley

Final Report

for

City of Seattle
Department of Lighting
Office of Environmental Affairs
Seattle, Washington

TABLE OF CONTENTS

ACKNOWLEDGMENTS

This study was sponsored by the City of Seattle Department of Lighting, Environmental Affairs Division, Seattle, Washington. The Fisheries Research Institute personnel responsible for conducting these studies are as follows:

Dr. Quentin J. Stober, Principal Investigator

Mr. S. C. Crumley, Project Leader

Mr. C. R. Steward, Project Biologist

Mr. Brian Winter, Project Biologist

Mr. D. E. Fast, Project Biologist

Mr. J. M. Scott, Project Biologist

Mr. E. S. Killibrew, Project Biologist

Other FRI personnel who provided field and laboratory assistance are Mr. F. Winchel, Mr. D. Martin and Ms. J. Minikin.

The outstanding programming skills of Ms. K. Swanson, and Mr. W. Meyer were instrumental to the study and are greatly appreciated. The U.S. Geological Survey provided essential equipment and advice for making discharge measurements as well as supplying timely discharge and temperature data for the Skagit River. Dr. R. Milhous from the U.S. Fish and Wildlife Instream Flow Group provided valuable advice on the Instream Flow Incremental Methodology. We thank Ms. Jean Baldrige for her critical and helpful review of this report.

SUPPLEMENTAL PROGRAMS TO IFG PHABSIM SYSTEM

A number of computer programs were developed at FRI in support of the IFG Physical Habitat Simulation (PHABSIM) system. The PHABSIM system itself is a collection of programs which when linked together provide a definition of the physical aquatic habitat. The FRI programs were designed to receive as input the output from one or more of the PHABSIM programs. Although the programs were developed specifically for the Skagit River Instream Flow Analysis, they were written in a generic style for universal applicability. A general description of the operation and purpose of each program is provided.

Program and Data Storage

All programs and data used and generated during the course of this study were written onto magnetic tape. The tapes were written on the CDC Cyber 170-750 under the NOS 1.4 operating system. They are standard labeled NOS tapes written at 6250 cpi. Two reels of tape were required to accomodate all of the files to be stored, and 3 identical tapes were made; that is, there are 6 tapes which are organized as three two-reel sets. The VSN (volume serial number) of each reel is written into its label. A list of the VSN's in each set follow:

Set	Reel #1	Reel #2
1	SKAG1	SKAG1B
2	SKAG2	SKAG2B
3	SKAG3	SKAG3B

Because it was perceived that it would be useful to store the files on tape in such a way that they could be recalled by name (as opposed to retrieving them by file position), the software package PF2TAPE/TAPE2PF was used to write the files onto tape. This package is described in the University of Washington Academic Computer Center document N85, a copy of which is included with this report (Appendix 3).

Here is an example of the job control statements that would be required to access tape SKAG1 and retrieve the files named PROGSET and PROCFIL. You want PROGSET to be saved as a direct access file (so you must DEFINE it first), and PROCFIL to be saved as an indirect access file.

```
LABEL,OLD,VSN=SKAG1,NT,D=GE,FA=S,PO=R.
DEFINE, PROGSET.
TAPE2PF,RF=PROGSET/PROCFIL.
SAVE,PROCFIL.
```

The following list gives the names of all files stored on the tapes Appendix 2, with a brief description of each file. The notation "A" in the "tape" column indicates it is on the first reel of the two-reel set; a "B" indicates it is on the second reel.

Filename	Tape	Description
TP13Ix	A	x = 1,2,3,4,5,7,8 Incubation TP13 for reach x; common criteria for all 4 species
TP13Sx	A	x = 1,2,3,4,5,7,8 Fileset with one element file for each of 4 species. Spawning TP13.
SxOUT	A	x = 1,2,3,4,5,7,8 IFG4 output with option 1=1, option 8=2
ELEVSET	A	Fileset; adjusted bed elevations for each reach (for BEDELV program)
SPNQSET	A	Fileset; input file of reach discharges for effective spawning models (all reaches/4 spp)
K13SET	B	Fileset; king (chinook) (1) output from EFSPWN, effective spawning WUA data with depth=.1, veloc=.1; input to programs WUA2EF2/WUA2EF3 (2) output from WUA2EF2/WUA2EF3, SUMTAB and MAXPROG
S13SET	B	Fileset; same as above for steelhead
P13SET	B	Fileset; same as above for pink
C13SET	B	Fileset; same as above for chum
HYDRSET	A	Fileset; output from IFG4 simulation models for all reaches: TAPE3, TAPE4, TAPE11, TAPE12; also summary data files VAF, VCE from TAPE11+12
IFG4SET	A	Fileset; calibrated IFG4 reach models
WUASET	A	Fileset; weighted usable area for all species by reach (output of program WUA1, input to WUA2)
SORCSET	A	Fileset; output of RHABINS, used for HABTAT program (criteria curve codes)
KBEDSET	A	Fileset; king (1) output of BEDELV, effective spawning WUA for adjusted bed elevations (input to WUA2EF2/WUA2EF3) (2) output from WUA2EF2/WUA2EF3, SUMTAB and MAXPROG
SBEDSET	A	Fileset; same as above for steelhead
PBEDSET	A	Fileset; same as above for pink
CBEDSET	A	Fileset; same as above for chum

Filename	Tape	Description
XEQBED	A	Fileset; xeq files for BEDELV
XEQWUA2	A	Fileset; xeq files for WUA2EF2/WUA2EF3
INCUBFS	A	Fileset; incomplete incubation model program and associated files
IFG4	A	card-image file of reduced field data; original uncalibrated input data to IFG4 hydraulic model
PROGSET	A	fileset, auxiliary programs written at FRI to process output from IFG modelling programs
PROCFIL	A	CCL procedure file, used to drive IFG programs
IFMLIB	A	IFG programs in source format
PHABSIM	A	IFG programs in compiled (binary) format

In the descriptions of individual programs below, a note will be made of which Filename on the tape (from the above list) contains the program and/or data files under discussion.

Notation Conventions

In the descriptions of program purpose and operation below, the following conventions will be used for indicating generalized file names:

All fixed parts of file names will be in capital letters

All parts of file names which can take on different values will be in lower case and will also be bolded

"**s**" will be used to stand for a species letter; in an actual file name it will have the value C,K,P or S (for chum, king, pink or steelhead).

"**x**" will be used to stand for a reach number; in an actual file name it will have the value 1,2,3,4,5,7, or 8

"**n**" will be used to stand for a sequence number; for example 1 and 2, if there are two files for the same species and/or reach that must be distinguished from each other

These "generalized file names" are used for groups of files which all have the same type of contents, but each file is for a separate species, reach, etc.

Examples:

HABOUT**x**	can be HABOUT1, HABOUT5, etc.
sADJ**n**	can be S1ADJ1,S1ADJ2 or S3ADJ1,S3ADJ2
TP13**s**S**x**	can be TP13KS7,TP13CS2

Program Description and Operation

I. Weighted Usable Area Programs

These programs, called WUA1, WUA2 and WUA3, are described in detail below. However, they all read information from a file called LABF ("label file"), so it will be described here. LABF gives a value for each of certain critical parameters of the analysis, plus a set of labels associated with each parameter. These labels are used to annotate the output of the Weighted Usable Area Programs. The parameters are:

NSTG	number of species-life stages
NC	number of tributary inflow conditions
NM	number of species weighting months
NP	number of exceedance probabilities

The structure of LABF must be as follows (see the Appendix for an example):

NSTG (format: (I2)) (maximum: 40)

then, one line for each species-life stage with the following format: (I5,5X,2A10,10F4.2), that is,

col. 1-5	species code as used in HABTAT
col. 11-30	species name (up to 20 characters)
col. 31-	species reach weighting factors, one for each reach (10 reaches maximum)

NC (format: (I2)) (maximum: 36)

then, one line for each tributary inflow condition giving the condition label using (7A10) format; ordered first by exceedance probability and then by month (where October = 1)

NM (format: (I2)) (maximum: 12)

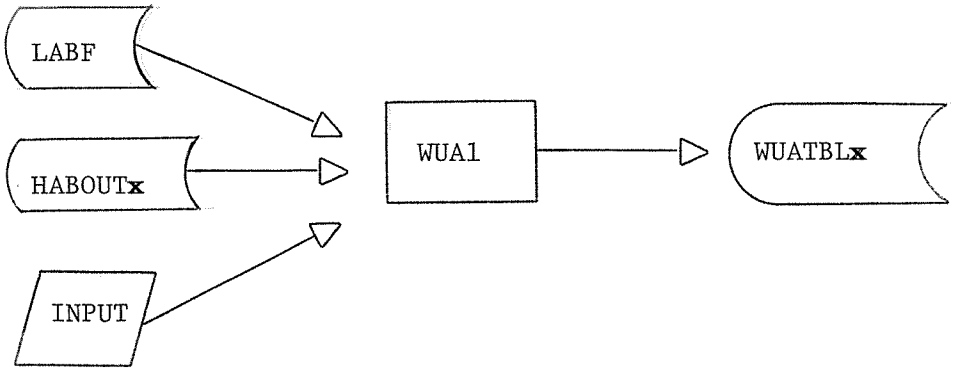
then, one line for each species weighting month giving a label for each month which usually indicates which PCTTBLx file is being used (by WUA3); (13A10)

NP (format: (I2)) (maximum: 3, must be 1, 2 or 3)

then, one line for each exceedance probability giving a label which is used for the WUA3 tables; (A3) format.

A. WUA1 - Reach Weighted Usable Area Versus Discharge Program

This program reads a file of standard output from the IFG HABTAT program (which is a formatted file with headings, carrage control, etc.) and extracts the WUA values for all species-life stages, as well as the gross area, for each discharge Q. These data are then written to a file for further processing by program WUA2. Here is a diagram of the program operation:



where:

(1) LABF is the label file described above

(2) HABOUTx is the HABTAT output data for reach x; some care must be taken in its construction. Any given run of HABTAT can only process up to 30 flows and 2 species-life stages. Therefore, it may take several HABTAT runs to produce all the data needed to cover the range of flows needed, for all the species-life stages of interest. The actual output file from HABTAT has the default name OUT. When they have all been created, they are copied together onto HABOUTx and then this file is packed (using the Cyber PACK command) to remove internal end-of-file marks.

The order in which the individual HABTAT output files are copied onto HABOUTx can be critical. If more than one run is required to process all the species-life stage codes, they must be grouped together within each set of flows. For example, if there are three sets of flows (up to 30 flows in each set) and two groups of species-life stage codes, for a total of 6 HABTAT runs, then they should be copied onto HABOUTx in the following order:

Flow set 1 - life stage group 1
Flow set 1 - life stage group 2
Flow set 2 - life stage group 1
Flow set 2 - life stage group 2
Flow set 3 - life stage group 1
Flow set 3 - life stage group 2

Furthermore, if there are enough species-life stage codes that they must be split between two runs of HABTAT (again, within a set of flows), the division should be made at the species level. That is, do not have some of the life stages for a species at the end of one group, and the rest of them at the beginning of the next group.

(3) INPUT is the standard input file, i.e., first record following the control cards in the job submitted to run WUA1. It contains three lines. For an example, see Appendix 1.

line 1: (713)
col. 1-3 = number of HABTAT runs used to create HABOUTx
col. 4-6 = number of discharges in first HABTAT run
7-9 = number of discharges in second HABTAT run
10-12 = number of discharges in third HABTAT run

13-15 = number of discharges in fourth HABTAT run
16-18 = number of discharges in fifth HABTAT run
19-21 = number of discharges in sixth HABTAT run
(only use as much of col. 4-21 as needed)

line 2: (A7)
PRINT or NOPRINT, depending on if you want some labelling information printed on the OUTPUT file.

line 3: (8A10); optional, only if PRINT is chosen above; a title for your printed OUTPUT.

(4) WUATBL**x** is the output data file for reach **x**. The following data are written to this file for each flow Q:

Q (the flow value)
up to 40 WUA's, one per species-life stage
the gross WUA for that flow

The format used is: (I5,14I9/5X,14I9/5X,13I9). Therefore, from one to three lines will be written to this file for each flow Q, depending on the number of species-life stages.

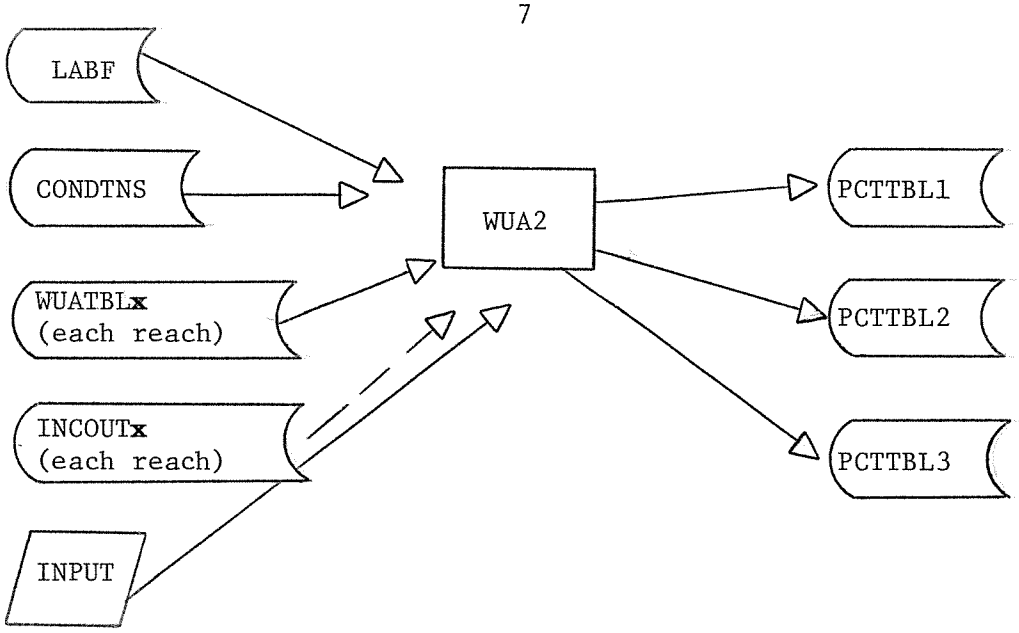
File storage: LABF and the program file WUA1 are in the fileset PROGSET on tape. The WUATBL**x** files are in the fileset WUASET.

WUA2 - River Weighted Usable Area Versus Discharge Program

The objective of this program is to produce river WUA versus discharge curves by species given the reach WUA versus discharge data (output of WUA1). The river discharge is referenced to a control point in this case Gorge Dam. The model accommodates tributary inflow below the control point as well as various extrapolation procedures for the different reaches.

This program reads the WUATBL**x** files produced by program WUA1 above, and optionally some other files of similar format, called INCOUT**x**. The INCOUT files are obtained in the following manner. Run the WUA2 program without the incubation option set. For each species identify the discharge at the control point providing the maximum WUA for the peak spawning month for the species. Calculate the reach discharge corresponding to the controlling point discharge by adding the month specific tributary inflow (median condition). Run the IFG effective spawning habitat program for this spawning discharge and lower incubation flows as needed. The INCOUT files contain incubation flows and WUA data which program WUA2 will substitute for the "normal" WUA data during incubation months only. The program produces 3 types of tables called PCTTBL1, PCTTBL2 and PCTTBL3. All tables summarize WUA across all reaches. PCTTBL1 tables WUA for each discharge as a percent of the greatest WUA, for each species-life stage. PCTTBL2 tables WUA for each discharge as a percent of the gross area, for each species-life stage. PCTTBL3 tables the percentages from PCTTBL1 divided by the gross area.

Here is a diagram of the program operation:



where:

(1) LABF is the label file described above.

(2) CONDTNS is the file containing tributary inflow conditions in cfs for each reach, exceedance probability and month. This file contains one line per reach and exceedance probability, with NM (number of months) data values on that line. Each line is read with this format: (I1,1X,A2,1X,12I5).

col. 1	reach number (identifier for this line)
col. 3-4	label for exceedance probability, e.g., 10,50,90 (optional identifier for this line)
col. 6-65	up to 12 inflow conditions for this reach and exceedance probability, one per month (12I5) beginning with October

The entries must be ordered by reach number, and then by exceedance probability within each reach number. See Appendix 1 for an example CONDTNS file.

(3) WUATBLx files are the output from program WUA1, above. They are copied onto a single file, divided by eof's, before input to program WUA2 (see submit file example in Appendix 1).

(4) INCOUTx files are output from IFG effective spawning habitat program. They are copied together onto a single file, divided by eof's, before onput to program WUA2 (see submit file example in Appendix 1).

(5) INPUT is the standard input file, i.e., first record following the control cards in the job submitted to run WUA2. For an example, see Appendix 1. There will be a variable number of lines, depending on the options chosen, as outlined below:

line 1: (2A10,5X,A10) "table option data record"

col. 1-14	the text "PERCENT TABLES"
col. 15-20	any one or more of the numbers 1,2, or 3, to select which PCTTBL tables you want

The following are only needed if you request PCTTBL3:
col. 26-33 the text "SCALE BY"
col. 34-35 habitat efficiency scaling factor for PCTTBL3, integer value between 1 and 9 which indicates a power of 10 to use as the actual scaling factor

line 2: (5A10) run label, used to label the output tables

line 3: (8I5) run parameters, namely:
col. 1-5 MINQ, smallest flow found in WUATBL's
col. 6-10 MAXQ, largest flow found in WUATBL's
col. 11-15 DIVQ, size of interval btw flows, cfs
col. 16-20 NR, number of reaches being processed
col. 21-25 BEGC, beginning condition number
col. 26-30 ENDC, ending condition number
the above two parameters allow you to table a subset of the full set of conditions and refer to the order in which the condition labels appear in LABF.
col. 31-35 NP, number of exceedance probabilities (must be equal or greater than 1)
col. 36-40 NM, number of months per exceed. prob.
Note: NC, the number of conditions for which you have entered data in the CONDTNS file, must be equal to NP x NM.

line 4 to (NR+3): (F6.3) reach weighting factors
NR lines, one per reach (NR = number of reaches)

The following lines may be omitted if there is no incubation data:

line NR+4: (A5,1X,I1) incubation data option record. Use this line to indicate if you are including incubation data via the INCOUT**x** files.

col. 1-5 the text "INCUB"
col. 7-8 NI, number of species for which there is incubation data

Then NI lines, one per incubation species: (I2,5X,6I2)

col. 1-2 species number relative to the order in which the species appear to LABF (i.e., they are numbered 1 to NSTG)

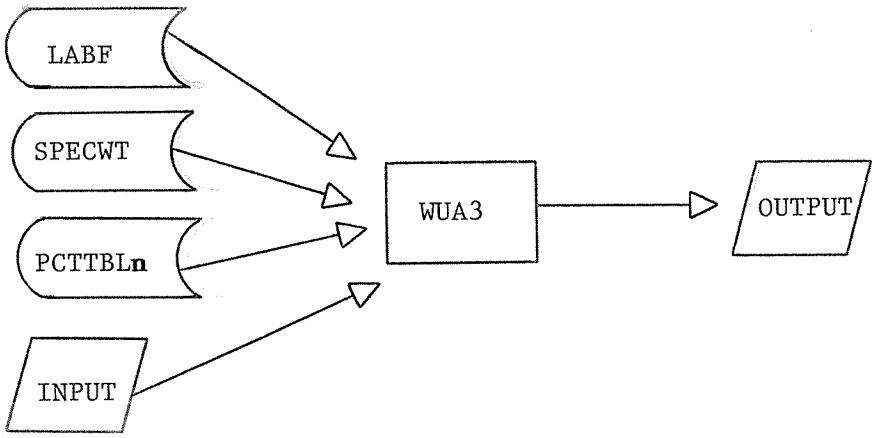
col. 8-19 up to 6 incubation month numbers for this species; these are not calendar months, but rather are numbered relative to the month that corresponds to the first tributary inflow condition (e.g., October = 1).

File storage: LABF, CONDTNS and the program file WUA3, as well as a submit file to run the program, are in the fileset PROGSET on tape. The WUATBL**x** files are in the fileset WUASET on tape. The INCOUT**x** files are not stored on tape, since that information was superceded by the output from FRI and IFG effective spawning models. PCTTBL1 are stored in WUASET as EDIT1 and EDIT2.

WUA3 - Optimized River Weighted Usable Area Versus Discharge Program

The objective of this program was the development of an optimization procedure which selected streamflows that maximized the physical habitat available for several species/life stages present simultaneously within a stream. This program reads one of the PCTTBL**n** files produced by program WUA2 above, and a file of species-life stage weighting factors, and writes out the original percentage tables weighted and summed across species-life stages. The monthly weighting factors were determined by the relative importance of each species-life stage present.

Here is a diagram of the program operation:



where:

(1) LABF is the label file described above

(2) SPECWT is the species weighting file and has the following structure:

line 1: (5A10)
up to 50 characters of title describing the weighting factors (odd or even year, and source) - printed in WUA3 tables as part of the heading

then, one line per month with up to 30 species-life stage weighting factors on each line; the months are ordered in the same order as they appear to the CONDTNS file (e.g., October = 1); use the following format:

col. 1-3 label for month (OCT, NOV, etc.)
col. 4-123 (30F4.2; decimal point can be implied or explicit) the weighting factors, using the same species order as in the LABF file

(3) PCTTBL**n** is a percent table output from program WUA2

(4) INPUT is the standard input file, i.e., first record following the control card record in the job that is submitted to run the program; see

Appendix 1 for an example submit file. It is structured as follows:

line 1: (6A10) run label, used as part of table heading

line 2: (6I5) run parameters

col. 1-5	MINQ, minimum flow value in the data
col. 6-10	MAXQ, maximum flow value in the data
col. 11-15	DIVQ, interval size in cfs for flows
col. 16-20	NR, number of reaches
col. 21-25	BEGC, beginning condition number
col. 26-30	ENDC, ending condition number

(5) OUTPUT is the standard output file

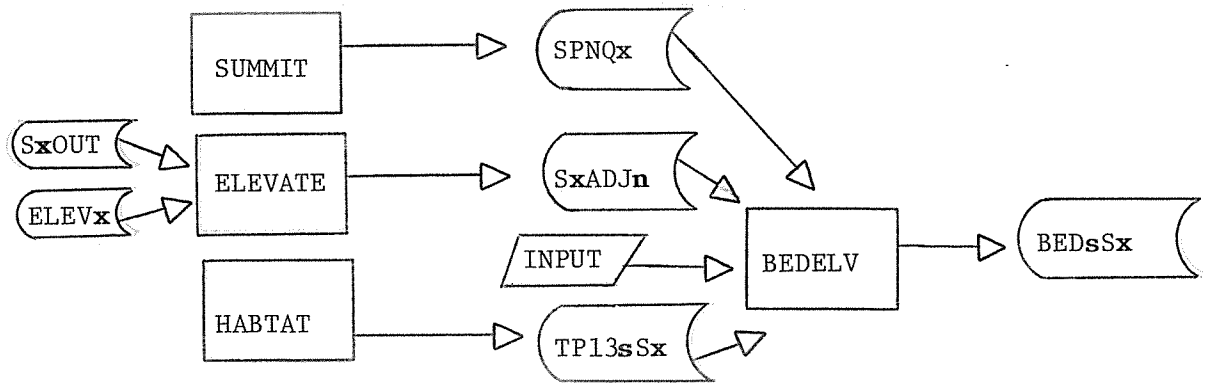
File storage: LABF, the program file WUA3 and a sample submit file are in the fileset PROGSET on tape, along with two species weighting data files called SPSWTE1 and SPSWTO1. This program was deleted from the Skagit Analysis, so there are no PCTTBL files stored.

II. EFFECTIVE SPAWNING HABITAT PROGRAM

A. BEDELV and Related Programs

The FRI effective spawning habitat program BEDELV was developed to determine the minimum incubation flows required to maintain water over the highest level of eggs located within the gravel and the amount of habitat lost for incubation flows lower than these minima. The depth below the gravel surface at which eggs are likely to be first encountered for each species were as follows: chinook salmon, 0.5 feet; pink salmon, chum salmon and steelhead trout, 0.25 feet.

The program compares the bed elevation of each cell adjusted for the species specific egg depth at a given spawning flow to the water surface elevation (WSEL) of an incubation flow of interest. If the WSEL is greater or equal to the adjusted bed elevation the cell is considered watered and the spawning WUA for that cell is summed. The effective spawning habitat is the summation of the WUA for all watered cells. The results are tabled in a "WUA1-like" data file format for a series of spawning flows that have been read from an external file. Before BEDELV can be run, some support programs must be run to produce data input files to BEDELV, namely, programs SUMMIT, ELEVATE and HABTAT. Here is a diagram of the program operation:



Support Program SUMMIT

This program is run once for each reach, producing a SPNQx file for that reach which contains spawning discharges at each exceedance probability of interest, for a range of Gorge releases. The program reads in tributary inflow values at the .90, .50 and .10 level for each species, i.e., tributary inflow during the spawning month for each species at each exceedance probability. An internal loop in the program covers the range of Gorge releases to be modified, and the tributary inflow values are added to these flow values to obtain the SPNQ file output.

where:

(1) INPUT is the standard input file, i.e., first record following the control card record in the job that is submitted to run the program (see Appendix 1 for an example submit file). It has the following format:

line 1: (4I10)
.90 level spawning month tributary inflow for pink, chum, steelhead and chinook, in that order

line 2: (4I10)
.50 level spawning month tributary inflow for pink, chum, steelhead and chinook, in that order

line 3: (4I10)
.10 level spawning month tributary inflow for pink, chum, steelhead and chinook, in that order

(2) SPNQx is the species-specific spawning month discharge file for the range of Gorge releases of interest, for reach x. For each Gorge release in the internal loop of program SUMMIT, a set of three lines are written to the SPNQ file. This set of three lines is formatted as follows:

line 1: (4I10 or 4F10.0)
for Gorge release Q, the .90 level spawning discharges for pink (col. 1-10), chum (col. 11-20), steelhead (col. 21-30) and chinook (col. 31-40).

line 2: (4I10 or 4F10.0)
for Gorge release Q, the .50 level spawning discharges for the same four species

line 3: (4I10 or 4F10.0)
For Gorge release Q, the .10 level spawning discharges for the same four species

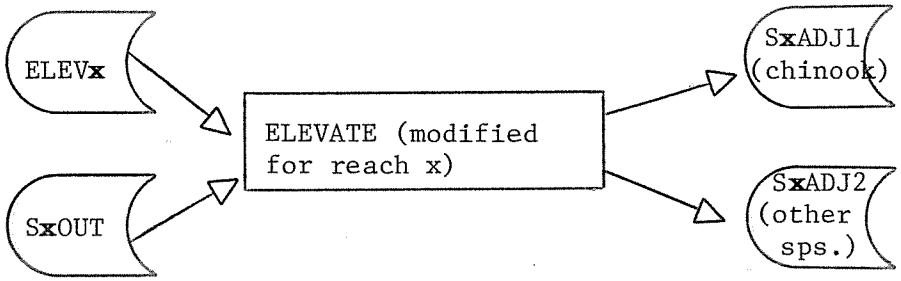
Support Program ELEVATE

Program ELEVATE computes adjusted bed elevations based on species-specific egg deposition depths. One set of adjusted bed elevations is written to a file for chinook, and another set to a file for the other three species.

The program is run once for each reach. The program contains internally set parameters which must be changed for each reach that is run. They are:

NC	number of cells in the reach
NIF	number of input flows in the SxOUT file
NSET	number of IFG4 runs used to construct SxOUT
NTRANS	number of transect in the reach
NIFSET	number of input flows per IFG4 run (30 max.)
TN	cell number of last cell in previous transect

The program operation can be diagrammed as follows:



where:

(1) ELEVx is a data file containing bed elevations for each cell in the reach, obtained by averaging bed elevations in adjacent verticals. It contains one line for each cell, with the format (1X,F10.2).

(2) SxOUT is a data file resulting from running the IFG-4 program with IDC option 1=1 and option 8=2. It contains WSEL (water surface elevation) and bed elevations for the range of spawning and incubation flows of interest. Each run of the IFG4 program can only process a maximum of 30 flows, so a number of runs must be made; each output file is copied onto SxOUT and then the file is packed to remove end-of-file marks. These are table-type files, with headings, labels, blank lines, etc.; ELEVATE looks for keywords and picks out the data values it needs. The parameters within that were described above tell the program how to read through the SxOUT file for reach x.

(3) SxADJ1 and SxADJ2: (F10.2,6F6.2,199F6.2)
The first file is adjusted by subtracting an egg depth of .5 feet for chinook, the second by substracting an egg depth of .25 feet for the other three species. Otherwise, they contain the same information, as follows:

col. 1-10	F10.2	discharge value Q
col. 11-46	6F6.2	stage, at up to 6 transects
col. 47-1240	199F6.2	adjusted bed elevations by cell

Support Program HABTAT

This is the standard IFG program HABTAT. The output files of interest are named TP13**s**S**x** (**s**=K,C,P,S for species; **x** = reach number). They contain the WUA for each individual cell in a reach for all spawning flows of interest. Also referred to as a TAPE13.

Program BEDELV

The program operation diagram is given at the beginning of this section. It was run once for each species and reach. A submit file for each species and reach is stored in the fileset XEQBED on tape; these submit files have the general name **s**BEDX**x** (e.g., CBEDX7).

(1) SPNQ**x** is output from program SUMMIT, see above.

(2) S**x**ADJ**n** is output from ELEVATE, see above.

(3) TP13**s**S**x** is output from HABTAT, see above.

(4) INPUT is the standard input file, i.e., the first record following the control card record in the submit file. An example submit file, labelled XEQBED, is in the Appendix. The run parameter line has the format (9I5) and contains the following parameters:

col. 1-5	RCH	reach number (must be 1-7, i.e., reach 7 must be entered as 6 and reach 8 as 7)
col. 6-10	SPEC	species number (1=pink, 2=chum, 3=steelhead, 4=chinook)
col. 11-15	INC1	lowest flow on S**x**ADJ**n**
col. 16-20	INC2	highest flow on S**x**ADJ**n**
col. 21-25	BEGFLO	beginning flow on TP13**s**S**x**
col. 26-30	SPN1	lowest spawning Q on SPNQ**x**
col. 31-35	SPN2	highest spawning Q on SPNQ**x**
col. 36-40	SUBQ1	lowest flow for which bed elevations were calculated (usu. same as INC1)
col. 41-45	SUBQ2	highest flow for which bed elevations were calculated (usu. same as INC2)

(5) BED**s**S**x** is the output file from BEDELV for species s and reach x; it is in the form of the WUA1 program output. The format used to write one line to this file is: (I5,14F9.0/11 (5X,14F9.0/),5X,3F9.0). Incubation flow Q is written in column 1-5 of line 1, then summed spawning WUA for each spawning flow read from SPNQ**x**, using as many lines as need with 14 values per line for a total of 171 spawning flows. The WUA is summed for cells that were watered at that incubation flow.

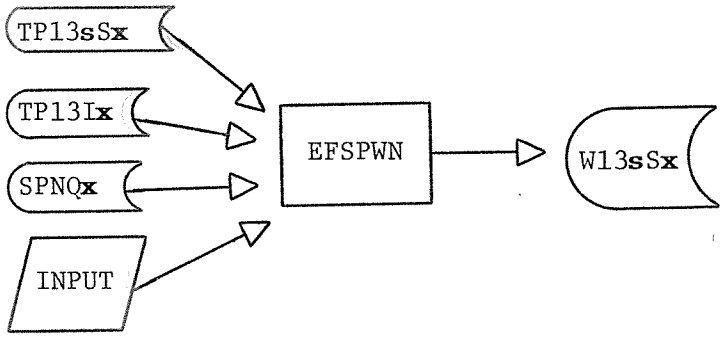
File storage: The program files SUMMIT, ELEVATE, BEDELV (and BEDELV1) are in the fileset PROGSET on tape; there is also a compiled, optimized version of BEDELV in the fileset in the element file BEDELV. The S**x**OUT files are stored individually on the tape. The ELEV**x** files are stored in the

fileset ELEVSET. The SxADJn files are also stored in ELEVSET. The TP13sSx files are stored in 7 filesets which have the generic name TP13Sx, i.e., all the files for one reach are stored together in a fileset. The BEDsSx files are stored in 4 filesets with the generic name sBEDSET, i.e., all the files for one species are stored together. The SPNQx files are stored in the fileset SPNQSET. An example submit file to run BEDELV is in PROGSET as element file XEQBED.

B. EFSPWN

This program is a redimensioned version of the IFG effective spawning habitat program designed to accommodate greater than 30 flows. This program compares the spawning habitat (WUA) predicted for the individual cells within a reach at a specified spawning flow to the WUA predicted for the same cells at an incubation flow with user specified depth and velocity incubation criteria. The lesser of the predicted spawning WUA or incubation WUA for each cell is summed to determine the effective spawning habitat for the reach.

The program operation can be diagrammed as follows:



where:

(1) TP13sSx is the spawning WUA file from running the IFG HABTAT program, described above for program BEDELV.

(2) TP13Ix is also an output file from the HABTAT program, but contains the incubation WUA by cell for all cells in a reach for each incubation flow of interest. The incubation criteria used to generate the WUA were the same for all species (p=.1, v=.1).

(3) SPNQx is the species-specific spawning month discharges for the range of Gorge discharges of interest, described above for program BEDELV.

(4) INPUT is the standard input file, i.e., the first record following the control card record in the submit file used to run the program (see the Appendix 1 for an example submit file, labeled XEQEFSP). It contains a single line with (6I5) format, and gives the following run parameters:

col. 1-5	RCH	reach number (must be 1-7 as in BEDELV)
col. 6-10	SPEC	species number (1=pink, 2=chum, 3=steelhead, 4=chinook)

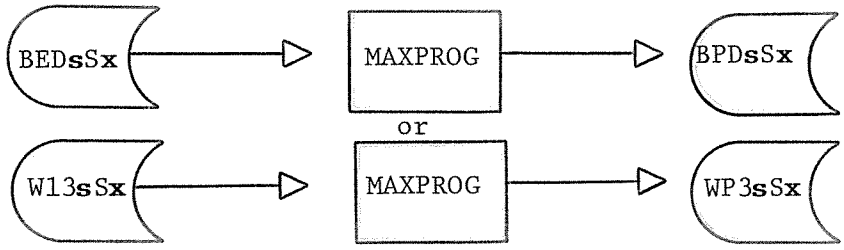
col. 11-15	MINIQ	minimum incubation Q to be tabled
col. 16-20	MAXIQ	maximum incubation Q to be tabled
col. 21-25	MINSQ	minimum spawning Q on SPNQ**x**
col. 26-30	MAXSQ	maximum spawning Q on SPNQ**x**

(5) W13**s**S**x** is the output file from program EFSPWN for species s and reach x. It is written with the format (I5,14F9.1/11(5X,14F9.0/),5X,3F9.0). This means that 13 lines will be written for each incubation flow between MINIQ and MAXIQ; each line contains 14 values (summed WUAs) except the last, which contains 3 values. The number of spawning flows is set internally in the program to 171 as gauge release from 1400-7000 cfs at 3 tributary inflow exceedance probabilities were evaluated (totalling 171 reach discharges). For each incubation flow, the first line has the incubation Q value in column 1-5.

File storage: The TP13**s**S**x** files are stored in 7 filesets on the tape, which have the generic name TP13S**x**; that is, all the files for one reach are stored together. The TP13I**x** files are each stored individually on the tape. The SPNQ**x** files are in the fileset SPNQSET. Program EFSPWN is in the fileset PROGSET, along with a submit file in element file XEQEFSP. The output data files W13**s**S**x** are stored in 4 filesets on the tape, which have the generic name **s**13SET; that is, all the files for one species are stored together.

MAXPROG - Processing output of BEDELV and/or EFSPWN

This is a small program which reads the output from BEDELV (BED**s**S**x**) or from EFSPWN (W13**s**S**x**). For each incubation flow, it determines the highest WUA value among all the spawning flows, and then recalculates each WUA as a percent of this highest WUA. The resulting percents are written out to a new file in the same format as they were read, except an F9.5 format is used instead of an F9.0 format. The number of spawning flows is encoded directly in the program to 171, which should be the number of spawning flows in the SPNQ**x** file used to run BEDELV or EFSPWN. The input format and output format also reflect this assumption.

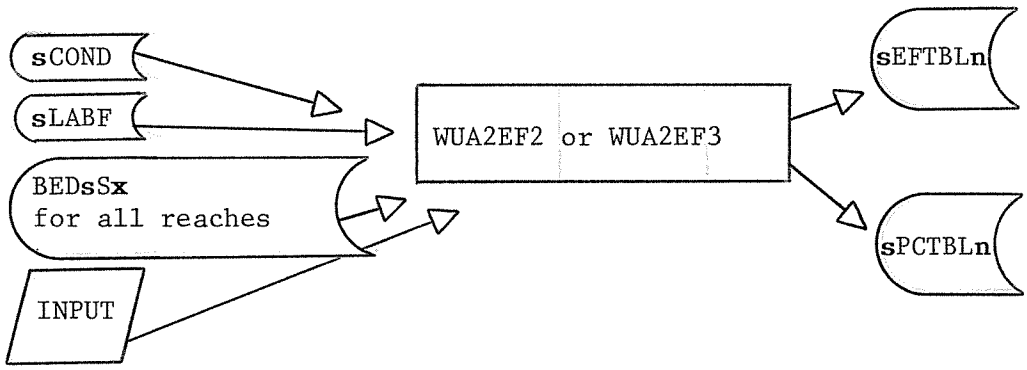


File storage: The control cards and program are in the file MAXPROG in the fileset PROGSET on tape. The BED**s**S**x** files are in the fileset for species s called **s**BEDSET. The output files BPD**s**S**x** are also stored in **s**BEDSET. The W13**s**S**x** files are in the fileset for species s called **s**13SET. The output files WP3**s**S**x** are also stored in **s**13SET.

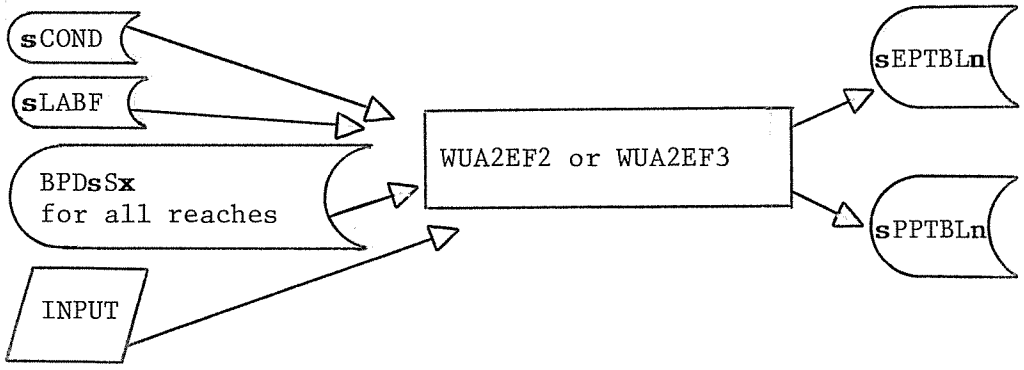
WUA2EFF - Effective Spawning Data in WUA2-like Format

This is a modified version of program WUA2; it combines reach WUA (i.e., output from BEDELV or EFSPWN) or percent WUA values (i.e., the output from BEDELV or EFSPWN that has been passed through MAXPROG), using reach extrapolation factors and adjustment for tributary inflow. The end result is river WUA values or percentages indexed to the Gorge Powerhouse. The program comes in a two or three pass form, due to a limitation on the total number of storage locations (sum of all array sizes plus individual variable storage) allowed in CDC FORTRAN. Remember that the incoming data files have 171 WUA or percent values per incubation flow (for the 171 spawning flows). The two-pass form, called WUA2EF2, handles up to 150 incubation flows. It reads the data for the first 84 spawning flows at each incubation flow on the first pass, and the last 87 spawning flows on the second pass. The three-pass form, called WUA2EF3, handles up to 185 incubation flows. It reads the data for the first 57 spawning flows at each incubation flow on the first pass, the middle 57 spawning flows (i.e., 58 through 114) on the second pass, and the last 57 spawning flows (i.e., 115 through 171) on the third pass. The optimized and compiled versions of the two and three pass versions are WUA2EB2 and WUA2EB3, respectively.

For runs made with BEDsSx (WUA value) files as input:



For runs made with BPDsSx (WUA percent) files as input:



Runs are also made with the EFSPWN output files W13sSx and the output file names are the same as with BEDsSx. Runs made with the EFSPWN via MAXPROG percentage files WP3sSx have the same output file names as those made with BPDsSx. However, the file storage for these output files are different, see file storage notes at the end of this section.

File descriptions for program operation diagrams:

(1) **s**COND contains tributary inflow to each of the 7 reaches for the incubation months for species s. For each reach, there is a set of lines, one per exceedance probability. The format is the same as for the CONDTNS file described for WUA2, except that the columns of data represent the incubation months of the species.

There are some examples of these files in the Appendix 1.

(2) **s**LABF is the label file, similar to LABF described for WUA2, except that the species-life stage labels are replaced by spawning flow-exceedance probability labels. It is structured as follows:

NSPF (I3) number of spawning flows, on first line (i.e., 171)
then, that many lines with (A10) format; each line has a "spawning flow - exceedance probability" label that corresponds with the WUA value or percent data to be processed.

NC (I3) number of conditions, which in this case is the number of incubation months times the number of exceedance probabilities.

then, that many lines containing the condition labels in (7A10) format; they are ordered by exceedance probability first, then by incubation month within exceedance probability.

There are some examples of these files in the Appendix.

Note: Steelhead were analyzed for three spawning months unlike the other species. Spawning month SCOND = May, SCONDA = March, SCONDB = April. So, first incubation month on SCONDA = April, on SCONDB = May.

(3) INPUT is the standard input file, i.e., first record following the control card record in the submit file that is used to run the program. It is structured as follows:

line 1: (7A10)
col. 1-10 species name
col. 11-70 run label

line 2: (9I5) run parameters
col. 1-5 MINQ minimum Gorge release to be tabled
col. 6-10 MAXQ maximum Gorge release to be tabled
col. 11-15 DIVQ discharge interval, in cfs
col. 16-20 NR number of reaches
col. 21-25 BEGC beginning condition number
col. 26-30 ENDC ending condition number
Note: the above two values tell the program which condition you want to see tabled; they are numbered in the order that the condition labels appear in the **s**LABF file.

col. 31-35 NP number of exceedance probabilities
col. 36-40 NM number of incubation months
col. 41-45 PASS which pass through the program this is (1 or 2

for WUA2EF2; 1, 2 or 3 for WUA2EF3)

lines 3-9: (F6.3)
reach factors, one line per reach

(4) **s**EFTBL**n:** the tabled output from pass "n", when the input data came from BED**s**S**x** for all reaches, or from W13**s**S**x** for all reaches.

(5) **s**PCTBL**n:** a coded data file, output during pass "n"; discharge providing any of 12 percentages of maximum effective spawning habitat for each spawning flow of interest. These files result when the input data came from BED**s**S**x** for all reaches, or from W13**s**S**x** for all reaches.

(6) **s**EPTBL**n:** the tabled output from pass "n", when the input data came from BPD**s**S**x** for all reaches, or from WP3**s**S**x** for all reaches.

(7) **s**PPTBL**n:** a coded data file, output during pass "n"; discharge providing any of 12 percentages of maximum effective spawning habitat for each spawning flow of interest (used as input to a plotting program). These files result when the input data came from BPD**s**S**x** for all reaches, or from WP3**s**S**x** for all reaches.

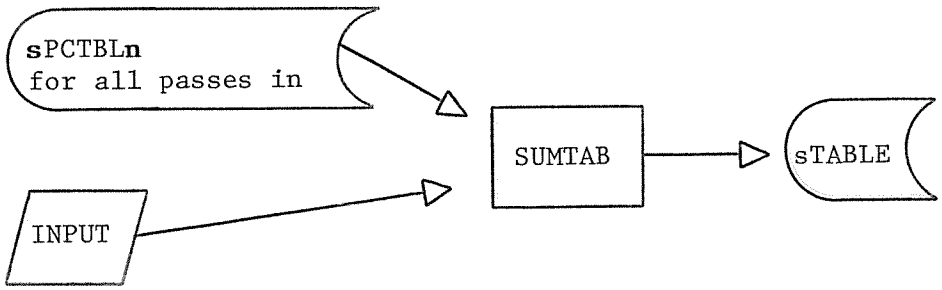
File storage: Programs WUA2EF2 and WUA2EF3 are in fileset PROGSET on tape, as are the species-specific condition and label files **s**COND and **s**LABF. The fileset XEQWUA2 on tape contains a complete series of submit files that were used to run the program. Input data files that originated with the BEDELV program, i.e., BED**s**S**x** and BPD**s**S**x**, are in the fileset for species s called **s**BEDSET. Output data files that correspond to these are also in **s**BEDSET. Input data files that originated with the EFSPWN program, i.e., W13**s**S**x** and WP3**s**S**x**, are in the fileset for species s called **s**13SET. Output data files that correspond to these are also in **s**13SET.

Note: SBEDSETA, S13SETA = steelhead spawning month of March. SDEDSETB, S13SETB = steelhead spawning month of April.

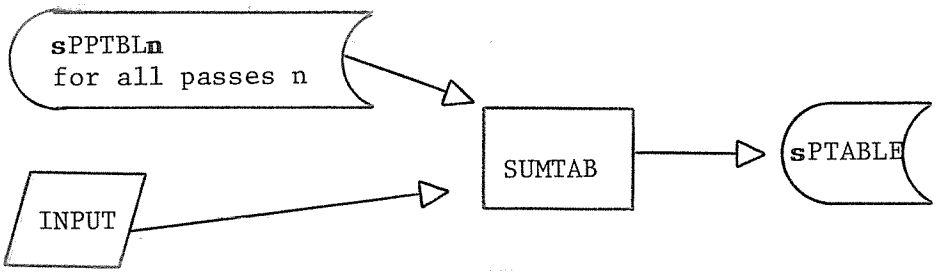
SUMTAB - Summary Tables for Final Report

This program reads the **s**PCTBL**n** or **s**PPTBL**n** files that came out of the WUA2EFF programs above, and produces summary tables for the final report giving the discharges providing any of 12 percentages of the maximum effective spawning habitat.

Operation diagram for EP I (Extrapolation Procedure I):



Operation diagram for EP II (Extrapolation Procedure II):



The input data files are adequately described in the previous section. When SUMTAB is run, the files from all passes through WUA2EF2 or WUA2EF3 are combined onto one file for input to this program.

INPUT is the standard input file, i.e., first record following the control card record in the submit file that is used to run the program (see Appendix 1 for an example submit file, labeled XSUMTAB). It consists of a single line using (2I3) format. Columns 1-3 contain the species code for the species being processed in this run, where 1 = chinook, 2 = pink, 3 = chum, and 4 = steelhead. Columns 4-6 contain the beginning page number for the output tables.

File storage: The input data files are stored as described in the previous section. If these files originated from the BEDELV program, then the output tables from SUMTAB are stored in the fileset sBEDSET. If the input files originated from the EFSPWN program, then the output tables from SUMTAB are stored in the fileset s13SET.

APPENDIX 1

Program, Submit File, and
Small Data File Listings

FILE: WUA1/XEQ

```
/JOB
XEQ1,P2.
/ACCOUNT
ATTACH,FILESET=PROGSET,HABOUT7/NA.
GF,LABF.
GF,WUA1.
FTN5,I=WUA1,L=0.
DEFINE,WUATBL7.
LGO,HABOUT7,LABF,WUATBL7.
REWIND,WUATBL7.
COPYSF,WUATBL7,OUTPUT.
/EOR
  5 30 30 30 30 18
PRINT
REACH 7 - GENERAL REACH WUA AND GROSS AREA BY Q
```

```
FILE: WUA1FIX/XEQ

/JOB
XEQ1FIX.
/ACCOUNT
COPYBR,INPUT,LABF.
REWIND,LABF.
ATTACH,PROGSET,HABSET,WUASET/M=W,NA.
FS,KTSET.
GF,WUA1.
FS,HABSET.
GF,*.
FS,WUASET.
FTN5,I=WUA1,L=0.
MAP=OFF.
LGO,HABNOS1,LABF,PSTBL1.
AF,PSTBL1,PSTBL1.
REWIND,PSTBL1.
COPYSF,PSTBL1,OUTPUT.
/EOR
 4
10110                CHINOOK/SPAWNING
10410                PINK/SPAWNING
10210                CHUM/SPAWING
11010                STEELHEAD/SPAWNING
/EOR
  3 30 30 27
PRINT
REACH 8- WUA AND GROSS AREA FOR KPCS/SPAWING ONLY
```

FILE: WUA2/XEQ

```
/JOB
XEQ2.
/ACCOUNT
ATTACH,FILESET=WUASET/NA.
GF,WUATBL?.
GF,INCOUT?.
ATTACH,FILESET=PROGSET/NA.
GF,WUA2.
GF,LABF.
GF,CONDTNS.
FTN5,I=WUA2.
COPYBF,WUATBL1,WUA.
COPYBF,WUATBL2,WUA.
COPYBF,WUATBL3,WUA.
COPYBF,WUATBL4,WUA.
COPYBF,WUATBL5,WUA.
COPYBF,WUATBL7,WUA.
COPYBF,WUATBL8,WUA.
REWIND,WUA.
COPYBF,INCOUT1,INC.
COPYBF,INCOUT2,INC.
COPYBF,INCOUT3,INC.
COPYBF,INCOUT4,INC.
COPYBF,INCOUT5,INC.
COPYBF,INCOUT7,INC.
COPYBF,INCOUT8,INC.
REWIND,INC.
PURGE,PCTTBL1,PCTTBL2,PCTTBL3/NA.
DEFINE,PCTTBL1,PCTTBL2,PCTTBL3.
LGO,WUA,LABF,CONDTNS,PCTTBL1,PCTTBL2,PCTTBL3,INC.
/EOR
PERCENT TABLES 1,2,3     SCALE BY 6
GORGE POWERHOUSE (RM 94.2) TO SAUK RIVER (RM 67.1)
    0 7400  100    7    1   36    3   12
26.400
11.620
16.370
11.620
13.730
34.320
23.760
INCUB 04
01       2 3 4 5 6 7
03       2 3 4 5 6 7
04       4 5 6 7 8
05      1011
```

FILE: WUA3/XEQ

```
/JOB
XEQ3.
/ACCOUNT
ATTACH,FILESET=PROGSET/NA.
GF,WUA3.
GF,LABF.
GET,SPECWT=SPSWT01.
ATTACH,PCTTBL=PCTTBL1/NA.
FTN5,I=WUA3.
LGO,PCTTBL,LABF,01SUM1,SPECWT.
SAVE,01SUM1.
/EOR
GORGE POWERHOUSE (RM 94.2) TO SAUK RIVER (RM 67.1)
    0 7400  100    7    1   36
```

FILE: SPSWT01

SPECIES WEIGHTING FOR ODD YEARS - FRI

OCT	.40		.45			.10	.05	
NOV	.35		.35	.15		.10	.05	
DEC	.30	.00	.30	.30		.05	.05	
JAN	.30	.10	.25	.25		.05	.05	
FEB	.25	.10	.25	.25	.05	.05	.05	
MAR	.20	.10	.20	.25	.15	.05	.05	
APR	.10	.10	.15	.25	.30	.05	.05	
MAY		.15		.25	.50	.05	.05	
JUN		.15			.55	.15	.10	.05
JUL		.15			.50	.20	.10	.05
AUG	.15				.50	.20	.10	.05
SEP	.40		.30			.15	.10	.05

FILE: SPSWTE1

SPECIES WEIGHTING FOR EVEN YEARS - FRI

OCT	.70					.25	.05		
NOV	.60		.20			.15	.05		
DEC	.40	.00	.45			.10	.05		
JAN	.40	.10	.35			.10	.05		
FEB	.35	.10	.35	.05		.10	.05		
MAR	.25	.10	.30	.20		.10	.05		
APR	.20	.10	.25	.30		.10	.05		
MAY		.15	.25	.45		.10	.05		
JUN		.15		.55	.15	.10	.05		
JUL		.15		.50	.20	.10	.05		
AUG	.15			.50	.20	.10	.05		
SEP	.60				.20	.15	.05		

FILE: LABF


```
26
10110      CHINOOK/SPAWNING
10100      CHINOOK/FRY
10410      PINK/SPAWNING
10210      CHUM/SPAWNING
11010      STEELHEAD/SPAWNING
11000      STEELHEAD/FRY
11001      STEELHEAD/JUVENILE
11002      STEELHEAD/ADULTS
10010      COHO/SPAWNING
10000      COHO/FRY
10001      COHO/JUVENILE
11110      RAINBOW/SPAWNING
11100      RAINBOW/FRY
11101      RAINBOW/JUVENILE
11102      RAINBOW/ADULT
13010      DOLLY VARDEN/SPAWN
13000      DOLLY VARDEN/FRY
13001      DOLLY VARDEN/JUVENIL
11210      CUTTHROAT/SPAWNING
11200      CUTTHROAT/FRY
11201      CUTTHROAT/JUVENILE
11202      CUTTHROAT/ADULT
12010      MTN WHITEFISH/SPAWN
12000      MTN WHITEFISH/FRY
12001      MTN WHITEFISH/JUVEN
12002      MTN WHITEFISH/ADULT
36
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
```

```
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
12
OCTOBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
NOVEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
DECEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JANUARY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
FEBRUARY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
MARCH - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
APRIL - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
MAY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JUNE - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JULY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
AUGUST - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
SEPTEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
 3
.90
.50
.10
```

FILE: CONDTNS

1	90	200	200	200	200	200	300	300	500	600	400	200	200
1	50	300	500	500	400	400	300	400	800	1000	700	400	300
1	10	700	700	1100	700	700	500	600	1100	1300	1100	600	500
2	90	200	300	300	200	300	400	400	700	800	600	300	300
2	50	400	600	700	600	500	400	500	1000	1300	1000	500	400
2	10	900	1000	1400	900	900	700	800	1500	1700	1500	800	600
3	90	300	400	400	300	300	500	600	900	1100	800	400	400
3	50	500	900	900	700	600	600	700	1300	1700	1200	700	600
3	10	1100	1300	1800	1100	1100	900	1000	1900	2200	2000	1100	800
4	90	400	600	600	500	600	700	800	1400	1600	1100	600	600
4	50	800	1300	1500	1200	1100	1000	1100	2000	2600	1900	1000	900
4	10	1700	2200	3000	1800	1800	1500	1600	2900	3600	3100	1700	1300
5	90	500	700	800	600	700	900	1100	1600	1900	1300	700	700
5	50	900	1500	1800	1400	1300	1200	1400	2400	3100	2300	1200	1100
5	10	2000	2700	3600	2200	2200	1800	2000	3500	4300	3800	2000	1600
6	90	900	1000	1200	1000	1000	1300	1700	2900	3600	2400	1300	1200
6	50	1700	2400	2700	2200	2000	1800	2300	4200	5300	4000	2100	1700
6	10	3300	4300	5200	3500	3400	2800	3300	6000	7400	6300	3400	2600
7	90	1100	1100	1400	1100	1100	1500	1900	3200	4000	2800	1500	1300
7	50	1900	2600	3000	2500	2200	2100	2500	4600	5900	4500	2400	1900
7	10	3700	4600	5800	4200	4000	3300	3600	6600	8200	7100	3800	2800

```
 8
10110      CHINOOK/SPAWNING      4329 6886 6306 5848 577411961 6983
10410      PINK/SPAWNING         528612173 6154 5306 540412864 6673
10210      CHUM/SPAWNING         528612173 6154 5306 540412864 6673
11010      STEELHEAD/SPAWNING    284212173 7736 6051 506431422 9012
10112      CHINOOK/SPAWNING-N    4329 6886 6306 5848 577411961 6983
10412      PINK/SPAWNING-N       528612173 6154 5306 540412864 6673
10212      CHUM/SPAWNING-N       528612173 6154 5306 540412864 6673
11012      STEELHEAD/SPAWNING-N  284212173 7736 6051 506431422 9012
36
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
12
OCTOBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
NOVEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
DECEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JANUARY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
FEBRUARY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
MARCH - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
APRIL - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
MAY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JUNE - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JULY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
AUGUST - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
```

SEPTEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH

3
.90
.50
.10

```
26
10110     CHINOOK/SPAWNING
10100     CHINOOK/FRY
10410     PINK/SPAWNING
10210     CHUM/SPAWNING
11010     STEELHEAD/SPAWNING
11000     STEELHEAD/FRY
11001     STEELHEAD/JUVENILE
11002     STEELHEAD/ADULTS
10010     COHO/SPAWNING
10000     COHO/FRY
10001     COHO/JUVENILE
11110     RAINBOW/SPAWNING
11100     RAINBOW/FRY
11101     RAINBOW/JUVENILE
11102     RAINBOW/ADULT
13010     DOLLY VARDEN/SPAWN
13000     DOLLY VARDEN/FRY
13001     DOLLY VARDEN/JUVENIL
11210     CUTTHROAT/SPAWNING
11200     CUTTHROAT/FRY
11201     CUTTHROAT/JUVENILE
11202     CUTTHROAT/ADULT
12010     MTN WHITEFISH/SPAWN
12000     MTN WHITEFISH/FRY
12001     MTN WHITEFISH/JUVEN
12002     MTN WHITEFISH/ADULT
36
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
```

APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
12
OCTOBER - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
NOVEMBER - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
DECEMBER - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
JANUARY - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
FEBRUARY - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
MARCH - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
APRIL - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
MAY - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
JUNE - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
JULY - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
AUGUST - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
SEPTEMBER - WUA AS A PERCENTAGE OF THE GREATEST WUA, DIVIDED BY THE GROSS AREA, WEIGHTED AND COMBINED FOR SPS. PRESENT (E-06)
3
.90
.50
.10

```
 8
10110      CHINOOK/SPAWNING      .116 .022 .084 .131 .258 .172 .217
10410      PINK/SPAWNING         .180 .073 .389 .144 .132 .036 .046
10210      CHUM/SPAWNING         .029 .012 .038 .055 .054 .356 .456
11010      STEELHEAD/SPAWNING    .042 .018 .045 .028 .122 .469 .276
10112      CHINOOK/SPAWNING-N    .116 .022 .084 .131 .258 .172 .217
10412      PINK/SPAWNING-N       .180 .073 .389 .144 .132 .036 .046
10212      CHUM/SPAWNING-N       .029 .012 .038 .055 .054 .356 .456
11012      STEELHEAD/SPAWNING-N .042 .018 .045 .028 .122 .469 .276
36
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
OCTOBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
NOVEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
DECEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
SEPTEMBER - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
12
OCTOBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
NOVEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
DECEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JANUARY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
FEBRUARY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
MARCH - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
APRIL - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
MAY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JUNE - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
JULY - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
AUGUST - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH
```

SEPTEMBER - WUA AS A PERCENTAGE OF GREATEST WUA, WEIGHTED AND COMBINED FOR SPECIES/LIFE STAGES PRESENT DURING THE MONTH

3
.90
.50
.10

LABF3 p.2

```
      PROGRAM WUA1(TAPE1,TAPE2,TAPE3,INPUT,OUTPUT,
     .            TAPE5=INPUT,TAPE6=OUTPUT)

C*********************************************************************
C*                                                                   *
C*     WUA (WEIGHTED USABLE AREA) PROGRAM 1:                         *
C*                                                                   *
C*     READS A FILE OF STANDARD OUTPUT FROM THE IFG HABTAT           *
C*     PROGRAM AND PRODUCES A FILE OF TABLED WUA'S AND GROSS         *
C*     AREA FOR EACH DISCHARGE Q.                                    *
C*                                                                   *
C*********************************************************************

C-----NOTES ON INPUT AND OUTPUT

C     THE IFG HABTAT FILE READ BY THIS PROGRAM MAY HAVE BEEN
C     PRODUCED BY SEVERAL HABTAT RUNS ON DIFFERENT SETS OF FLOWS.
C     WITHIN A SET OF FLOWS, MORE THAN ONE RUN MAY HAVE BEEN MADE
C     TO COVER ALL THE SPECIES-LIFE STAGE CODES.  IF SO, ALL THE
C     RUNS MADE FOR DIFFERENT LIFE STAGE CODES MUST BE GROUPED
C     TOGETHER WITHIN A SET OF FLOWS.  FOR EXAMPLE:
C         FLOW SET 1 - LIFE STAGE GROUP 1
C         FLOW SET 1 - LIFE STAGE GROUP 2
C         FLOW SET 2 - LIFE STAGE GROUP 1
C         FLOW SET 2 - LIFE STAGE GROUP 2
C         FLOW SET 3 - LIFE STAGE GROUP 1
C         FLOW SET 3 - LIFE STAGE GROUP 2
C     WHERE THERE WERE 3 SETS OF FLOWS, AND 2 GROUPS OF LIFE
C     STAGE CODES (6 HABTAT RUNS WITH THE OUTPUT COPIED ONTO
C     ONE PACKED FILE IN THE ORDER GIVEN ABOVE).
C
C     IF THE LIFE STAGE CODES MUST BE SPLIT INTO MORE THAN ONE
C     GROUP, THE DIVISION SHOULD BE MADE AT THE SPECIES LEVEL.  THAT IS,
C     DO NOT HAVE SOME OF THE LIFE STAGES FOR ONE SPECIES
C     AT THE END OF ONE GROUP, AND THE REST OF THEM AT THE
C     BEGINNING OF THE NEXT GROUP.

C     THE FOLLOWING DATA ARE WRITTEN TO TAPE3 FOR EACH FLOW Q:
C       Q (THE FLOW VALUE)
C       UP TO 40 WUA'S, ONE PER SPECIES-LIFE STAGE
C       THE GROSS WUA FOR THAT FLOW
C
C     THE FORMAT USED IS:  (I5,14I9/5X,14I9/5X,13I9)
C     HENCE, 1-3 LINES WILL BE WRITTEN TO TAPE3 PER FLOW,
C     DEPENDING ON THE NUMBER OF LIFE STAGE CODES (MAX 40)
C

C-----FILES:

C     TAPE1 = INPUT, STANDARD HABTAT OUTPUT
C     TAPE2 = INPUT, LABEL FILE (PARAMETERS AND LIFE STAGE LABELS)
C     TAPE3 = OUTPUT, TABLED WUA DATA
C     TAPE5 = STANDARD INPUT FILE, INFORMATION ABOUT THE REACH BEING
C             PROCESSED
C     TAPE6 = STANDARD OUTPUT FILE, ERROR MESSAGES AND DIAGNOSTICS

C-----LOCAL VARIABLES:
      INTEGER     MAXR,   MAXSTG, MAXSP,  SPEC,   PSPEC,  NR,     NSP,
     .            NLH(10),NSTG,   NQ(10), NRUN,   Q(30),  WUA(30,40),
```

```
     .              GROSS(30),      DFMT(2),     LINEIN(10)
      REAL          XWUA(6)
      CHARACTER*10  TITLE(8),       PRINT

*     DEFINITIONS:

*     MAXR =        MAX. NUMBER OF RUNS (REPRESENTING SETS OF FLOWS)
*     MAXSTG =      MAX. NUMBER OF SPECIES-LIFE STAGE CODES
*     MAXSP =       MAX. NUMBER OF SPECIES
*     SPEC =        SPECIES PORTION OF THE SPECIES-LIFE STAGE CODE
*     NSP =         NUMBER OF SPECIES
*     NLH(I) =      NUMBER OF LIFE STAGES FOR SPECIES I
*     NSTG =        NUMBER OF SPECIES-LIFE STAGE CODES
*     NQ(I) =       NUMBER OF FLOWS PER RUN I
*     NRUN =        COUNTS THE RUNS (SETS OF FLOWS)
*     WUA(I,J) =    WUA FOR FLOW I, SPECIES-LIFE STAGE CODE J
*     Q(I) =        FLOW VALUE I
*     GROSS(I) =    GROSS AREA FOR FLOW I
*     XWUA(K) =     WUA'S FOR ONE SPECIES AT ONE FLOW (REAL VALUE)
*     LINEIN =      LINE READ FROM HABTAT OUTPUT FILE
*     TITLE =       TITLE (OPTIONAL)
*     DFMT =        FORMAT USED TO DECODE WUA'S FROM ONE LINE

*-----CONSTANTS:
      DATA                MAXR,MAXSTG,MAXSP/10,40,10/

*-----INITIALIZATION:
      CALL FILEB(1)
      DO 700 I=1,10
  700      NLH(I) = 0
      DFMT(1) = 10H(24X, F12.
      DFMT(2) = 2H2)

*-----READ LABEL FILE AND ESTABLISH SPECIES-LIFE STAGE CONSTANTS:
      READ(2,21) NSTG
   21 FORMAT(I2)
      IF(NSTG .GT. MAXSTG) THEN
           WRITE(6,62) NSTG,MAXSTG
           STOP 1
   62      FORMAT(1HO,'ERROR...NUMBER OF LIFE STAGES (',I2,') EXCEEDS',
             ' MAXIMUM ALLOWED (',I2,')')
      ENDIF
      NSP = 0
      PSPEC = 0
      DO 701 I=1,NSTG
           READ(2,23) SPEC
   23 FORMAT(I3)
           IF(SPEC .NE. PSPEC) THEN
               NSP = NSP + 1
               PSPEC = SPEC
           ENDIF
           NLH(NSP) = NLH(NSP) + 1
  701 CONTINUE

*-----READ STANDARD INPUT FILE FOR INFO ABOUT RUNS FOR THIS REACH
      READ(5,51) NRUN,(NQ(I),I=1,NRUN)
      READ(5,52) PRINT
      IF(PRINT .EQ. 'PRINT') THEN
          READ(5,53) TITLE
          WRITE(6,1) TITLE
```

```
          DO 760 I=1,NSP
              WRITE(6,2) I,NLH(I)
  760     CONTINUE
      ENDIF
     51 FORMAT(11I3)
     52 FORMAT(A10)
     53 FORMAT(8A10)
      1 FORMAT(1H1,8A10//)
      2 FORMAT(5X,'SPECIES',I3,' HAS',I3,' LIFE STAGES.')

*-----READ HABTAT FILE, ONE SET OF FLOWS (= RUN) AT A TIME;
*     STORE, THEN WRITE OUT, DATA FOR EACH FLOW Q

      NRUN = 0

*-----FIND BEGINNING OF A SECTION OF HABTAT OUTPUT FOR A SET OF
*     FLOWS (ALL LIFE STAGES)

  601 CALL READH(1,LINEIN,10,IREP)
      IF(IREP .NE. 0) GOTO 602
      IF(LINEIN(1) .NE. 10H**********) GOTO 601
      NRUN = NRUN + 1

*-----NOW LOOP ON SPECIES, FILLING THE WUA ARRAY FOR THIS RUN

      N2 = 0
      DO 711 ISP=1,NSP
          N1 = N2 + 1
          N2 = N2 + NLH(ISP)
          CALL INTCOD(DFMT,6,1,NLH(ISP))
  611     CALL READH(1,LINEIN,10,IREP)
          IF(LINEIN(2) .NE. 10HQ VS. AVAI) GOTO 611
          CALL READH(1,LINEIN,10,IREP)
          CALL READH(1,LINEIN,10,IREP)
          DO 712 IQ=1,NQ(NRUN)
              CALL READH(1,LINEIN,10,IREP)
              DECODE(100,DFMT,LINEIN) (XWUA(J),J=1,NLH(ISP))
              DO 722 J=N1,N2
                  J2 = J - N1 + 1
                  WUA(IQ,J) = NINT(XWUA(J2))
  722         CONTINUE
  712     CONTINUE
  711 CONTINUE

*-----NOW GET THE FLOW VALUES AND THE GROSS AREA FOR THIS SET
*     OF FLOWS

  621 CALL READH(1,LINEIN,10,IREP)
      IF(LINEIN(1) .NE. 10H      Q VS. ) GOTO 621
      CALL READH(1,LINEIN,10,IREP)
      CALL READH(1,LINEIN,10,IREP)
      DO 713 IQ=1,NQ(NRUN)
          CALL READH(1,LINEIN,10,IREP)
          CALL INTMOV(LINEIN,13,11,Q(IQ))
          CALL INTMOV(LINEIN,25,11,GROSS(IQ))
  713 CONTINUE

*-----NOW WRITE OUT DATA FOR THE SET OF FLOWS IN THIS RUN
*     (ONE LINE PER FLOW)
```

```
      DO 714 I=1,NQ(NRUN)
          WRITE(3,31) Q(I),(WUA(I,J),J=1,NSTQ),GROSS(I)
   31 FORMAT(I5,14I9/5X,14I9/5X,13I9)
  714 CONTINUE

*-----READY TO GO BACK AND START A NEW SET OF FLOWS

      GOTO 601

*-----COME HERE AT END OF FILE

  602 CONTINUE
      ENDFILE 3
      STOP
      END
```

FILE: WUA2

```
      PROGRAM WUA2(TAPE1, TAPE2, TAPE3, TAPE4, TAPE7, TAPE8, TAPE9,
     .             INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT)

C*********************************************************************
C*    READS (1) TABLED WUA AND GROSS AREA BY DISCHARGE, FOR EACH     *
C*              REACH,                                               *
C*          (2) OPTIONALLY, INCUBATION FLOWS AND WUA FOR SELECTED    *
C*              SPECIES (TO OVERRIDE ABOVE IN CERTAIN MONTHS)        *
C*          (2) NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH        *
C*          (3) A LABEL FILE                                         *
C*    AND PRODUCES 3 TABLES OF OUTPUT (SUMMED ACROSS REACHES):       *
C*          (1) WUA FOR EACH DISCHARGE AS PERCENT OF THE GREATEST    *
C*              WUA, FOR EACH LIFE STAGE                             *
C*          (2) WUA FOR EACH DISCHARGE AS PERCENT OF THE GROSS       *
C*              AREA, FOR EACH LIFE STAGE                            *
C*          (3) TABLE 1 PERCENTS DIVIDED BY GROSS AREA FOR EACH      *
C*              DISCHARGE                                            *
C*********************************************************************

C-----FILES:
C
C     TAPE1 = INPUT, WUA TABLES PRODUCED BY PROGRAM WUA1, END-OF-
C             FILE BETWEEN TABLES FOR EACH REACH
C     TAPE2 = INPUT, LABEL FILE
C     TAPE3 = INPUT, NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH
C     TAPE9 = INPUT, INCUBATION DATA, EOF BETWEEN REACHES
C     TAPE4 = OUTPUT, TABLED WUA AS PERCENT OF GREATEST WUA
C     TAPE7 = OUTPUT, TABLED WUA AS PERCENT OF GROSS AREA
C     TAPE8 = OUTPUT, TABLE 1 PERCENTS DIVIDED BY GROSS AREA
C     TAPE5 = STANDARD INPUT FILE
C     TAPE6 = STANDARD OUTPUT FILE (ERROR MESSAGES, DIAGNOSTICS)

C-----LOCAL VARIABLES:

      INTEGER      MINQ, MAXQ, DIVQ, NR, NSTG, NC, MAXR, MAXSTG, MAXD,
     .             REACH, ND, Q, R, D, C, NSUB, D1, D2, NDIS(10), ADJFLO,
     .             WUA(10,150,40), GROSS(10,150), FLOW(10,150),
     .             COND(10,36), SPLABL(2,40), CLABL(7,36), QLABL(15),
     .             BEGC, ENDC, RUNLABL(6), DUM, INCUB, INCSP(5),
     .             INCMO(5,6), ICMAT(5,36), INCWUA(10,150,5),
     .             SCALE, T3SCL, TBLOPT(2)

      REAL         TBL1(150,40), SUMW(150,40), SUMG(150), WMAX(40),
     .             FACTOR(10), SFACTR(10,40), TBL3(150,40)

      LOGICAL      PTBL1, PTBL2, PTBL3

C     DEFINITIONS
C     MINQ,MAXQ = MINIMUM AND MAXIMUM DISCHARGE VALUES TO BE TABLED
C     DIVQ =      SIZE OF INTERVAL BETWEEN TABLED DISCHARGE VALUES
C     NR =        NUMBER OF REACHES BEING PROCESSED
C     NSTG =      NUMBER OF SPECIES-LIFE STAGE CODES
C     NC =        NUMBER OF TRIBUTARY INFLOW CONDITIONS TO BE TABLED
C     MAXR =      MAXIMUM NUMBER OF REACHES THAT CAN BE PROCESSED
C                 WITHOUT RE-DIMENSIONING ARRAYS
C     MAXSTG =    MAXIMUM NUMBER OF SPECIES-LIFE STAGE CODES
C     MAXD =      MAXIMUM NUMBER OF DISCHARGE LEVELS (BEFORE ADJUSTING FOR
C                 TRIBUTARY INFLOW CONDITIONS AND RESTRICTING TO MINQ-MAXQ)
```

```
*      NSUB =       NUMBER OF SUB-TABLES, WITH 15 DISCHARGE LEVELS PER
*                   SUB-TABLE
*      WUA(R,D,L) = WEIGHTED USABLE AREA FOR REACH R, DISCHARGE D,
*                   LIFE STAGE CODE L
*      GROSS(R,D) = GROSS AREA FOR REACH R, DISCHARGE D
*      FLOW(R,D) = DISCHARGE VALUE FOR REACH R, DISCHARGE D
*      COND(R,C) = TRIBUTARY INFLOW CONDITION FOR REACH R, CONDITION C
*      SPLABL(I,L) = SPECIES-LIFE STAGE LABEL FOR LIFE STAGE L
*      CLABL(I,C) = TABLE HEADING FOR CONDITION C
*      QLABL(I) =   USED IN SUB-TABLE HEADING (15 FLOW VALUES)
*      TBL1(D,L) = TABLE 1 VALUES FOR DISCHARGE LEVEL D, LIFE STAGE L
*                   (AFTER RESTRICTING ADJUSTED FLOWS TO MINQ-MAXQ)
*      SUMW(D,L) = WUA SUMMED ACROSS REACHES FOR DISCHARGE LEVEL D,
*                   LIFE STAGE L (THEN BECOMES TABLE 2 VALUES AFTER DIVIDING
*                   BY GROSS AREA)
*      TBL3(D,L) = TABLE 1 PERCENTS DIVIDED BY GROSS AREA FOR DISCHARGE D
*      SUMG(D) =    GROSS AREA SUMMED ACROSS REACHES FOR DISCHARGE LEVEL D
*      WMAX(L) =    MAXIMUM WUA ACROSS ALL DISCHARGE LEVELS FOR A GIVEN
*                   LIFE STAGE L
*      FACTOR(R) = MULTIPLICATIVE FACTOR FOR REACH R, EXPANDS WUA
*                   AND GROSS AREA VALUES TO REPRESENT A SECTION
*                   OF THE RIVER
*      SFACTR(R,L) = SPECIES-DISTRIBUTION REACH WEIGHTING FACTOR FOR
*                   REACH R AND SPECIES-LIFE STAGE L
*      BEGC,ENDC = BEGINNING AND ENDING INDEX WITHIN CONDITION
*                   ARRAYS FOR A SUBSET OF THE NC CONDITIONS TO BE USED
*                   IN ANY GIVEN RUN
*      NP =         NUMBER OF EXCEEDANCE PROBABILITES
*      NM =         NUMBER OF MONTHS AT EACH EXCEEDANCE PROBABILITY
*
*      INCUBATION DATA:
*
*      INCUB =      FLAG, BLANK IF NO INCUBATION DATA ON THIS RUN
*      INCSP(I) =   SPECIES NUMBERS FOR WHICH INCUBATION DATA WILL BE
*                   PROVIDED, WHERE INCSP(I) = L AND L IS IN THE RANGE
*                   (1 TO NSTG), I.E. THE SPECIES-LIFE STAGE NUMBERS USED
*                   IN THE BASIC DATA
*      INCMO(I,J) = J-TH MONTH NUMBER IN THE INCUBATION PERIOD FOR SPECIES
*                   L, WHERE L = INCSP(I).  MONTHS ARE NUMBERED RELATIVE
*                   TO THE FIRST MONTH USED FOR CONDITIONS, E.G. IF OCT=1
*                   THEN INCUBATION MONTH DEC=3.
*      ICMAT(I,C) = CONSTRUCTED MATRIX THAT INDICATES IF CONDITION C
*                   REPRESENTS AN INCUBATION MONTH FOR SPECIES L, WHERE
*                   L = INCSP(I)  (1=YES, 0=NO)
*      INCWUA(R,D,I) = COMPANION ARRAY TO WUA(R,D,L).  INCUBATION WUA
*                   FOR REACH R, DISCHARGE D AND SPECIES L, WHERE
*                   L = INCSP(I) (INCUBATION SPECIES I)
*
*       TABLING OPTIONS:
*
*       PTBL1,PTBL2,PTBL3 = LOGICAL FLAGS INDICATING IF PERCENT
*                   TABLES 1,2 AND/OR 3 SHOULD BE PRINTED
*       TBLOPT(2) = COL 1-20 OF TABLE OPTION RECORD - SHOULD
*                   CONTAIN THE NUMBERS 1,2 AND/OR 3
*       SCALE =     COL 26-35 OF TABLE OPTION RECORD - IF TABLE 3
*                   IS TO BE PRINTED, THIS CONTAINS A NUMBER
*                   WHICH IS THE HABITAT EFFICIENCY SCALING FACTOR
*                   (A POWER OF 10)
*       T3SCL =     HABITAT EFFICIENCY SCALING FACTOR FOR TABLE 3
```

WUA 2 p.3

```
*------CONSTANTS:
      DATA MAXR,MAXSTG,MAXD/10,40,150/

*------INITIALIZATION:
      CALL FILEB(9)
      PTBL1 = PTBL2 = PTBL3 = .FALSE.

*------READ TABLE OPTION DATA RECORD AND DECODE
      READ(5,55) TBLOPT,SCALE
      CALL ISCAN(TBLOPT,1,20,1H1,1,1,I)
      IF(I .NE. 0) PTBL1 = .TRUE.
      CALL ISCAN(TBLOPT,1,20,1H2,1,1,I)
      IF(I .NE. 0) PTBL2 = .TRUE.
      CALL ISCAN(TBLOPT,1,20,1H3,1,1,I)
      IF(I .NE. 0) PTBL3 = .TRUE.
      IF(PTBL3) THEN
          CALL ISCAN(SCALE,1,10,9H123456789,1,9,I)
          IF(I .EQ. 0) THEN
              WRITE(6,60)
              STOP 0
   60         FORMAT(1HO,'ERROR...YOU HAVE REQUESTED PERCENT TABLE 3,',
     .                ' BUT HAVE NOT GIVEN A HABITAT EFFICIENCY '/
     .                6X,'SCALING FACTOR IN COL. 26-35 OF THE TABLE ',
     .                'OPTION DATA RECORD.')
          ENDIF
          CALL INTMOV(SCALE,I,1,T3SCL)
          T3SCL = 10**T3SCL
      ENDIF

*------READ PARAMETERS, SPECIES LABELS, CONDITION LABELS
      READ(5,50) RUNLABL
      READ(5,51) MINQ,MAXQ,DIVQ,NR,BEGC,ENDC,NP,NM
      DO 700 R=1,NR
          READ(5,52) FACTOR(R)
  700 CONTINUE

      INCUB = 1H
      READ(5,53) INCUB,NI
      IF(EOF(5) .NE. 0) GOTO 200
      IF(INCUB .EQ. 1H ) GOTO 200
      DO 730 I=1,NI
          READ(5,54) INCSP(I),(INCMO(I,J),J=1,6)
  730 CONTINUE
  200 CONTINUE

      READ(2,21) NSTG
      DO 701 L=1,NSTG
          READ(2,22) SPLABL(1,L),SPLABL(2,L),(SFACTR(R,L),R=1,NR)
          DO 701 R=1,NR
              IF(SFACTR(R,L).EQ.0.) SFACTR(R,L)=1.
  701 CONTINUE
      READ(2,21) NC
      IF(NC .NE. NP*NM) THEN
          WRITE(6,63) NC,NP,NM
          STOP 3
   63     FORMAT(1HO,'ERROR...NUMBER OF CONDITIONS SPECIFIED IN ',
     .           'LABEL FILE IS',I3,'.'/6X,'THIS DOES NOT EQUAL ',
     .           'THE PRODUCT OF THE NUMBER OF EXCEEDANCE ',
     .           'PROBABILITIES (',I2,') TIMES THE NUMBER OF ',
     .           'MONTHS (',I2,')'/6X,'GIVEN ON THE RUN PARAMETERS ',
```

```
                'CARD. '/6X,'PROGRAM STOPPED. ')
      ENDIF
      DO 702 C=1,NC
          READ(2,23) (CLABL(K,C),K=1,7)
  702 CONTINUE

   21 FORMAT(I2)
   22 FORMAT(10X,2A10,10F4.2)
   23 FORMAT(7A10)
   50 FORMAT(6A10)
   51 FORMAT(8I5)
   52 FORMAT(F6.3)
   53 FORMAT(A5,1X,I2)
   54 FORMAT(I2,5X,6I2)
   55 FORMAT(2A10,5X,A10)

*------IF INCUBATION IS TO BE INCLUDED IN THIS RUN, CONSTRUCT THE
*      -ICMAT- ARRAY (WHICH CONDITIONS ARE INCUBATION MONTHS FOR
*      WHICH SPECIES)
      IF(INCUB .NE. 1H ) THEN
          DO 731 I=1,NI
              DO 732 C=1,NC
                  ICMAT(I,C) = 0
  732         CONTINUE
              DO 733 J=1,6
                  IF(INCMO(I,J) .EQ. 0) GOTO 733
                  M = INCMO(I,J)
                  DO 734 P=1,NP
                      C = M + NM*(P-1)
                      ICMAT(I,C) = 1
  734             CONTINUE
  733         CONTINUE
  731     CONTINUE
      ENDIF

*------READ IN THE WUA, GROSS AREA, AND DISCHARGE VALUES FOR THE
*      NR REACHES (EACH REACH SEPERATED BY END-OF-FILE)

      DO 703 R=1,NR
          D = 0
  601     D = D + 1
          IF(D .GT. MAXD) THEN
              READ(1,11,END=602) DUM
              WRITE(6,61) MAXD,R
              STOP 1
          ENDIF
          READ(1,11,END=602) FLOW(R,D),(WUA(R,D,I),I=1,NSTG),
     .        GROSS(R,D)
          GOTO 601
  602     NDIS(R) = D - 1
*         CLEAR END-OF-FILE INDICATOR BEFORE READING IN NEXT REACH
          IF(EOF(1) .NE. 0.) CONTINUE

*------IF INCUBATION DATA IS BEING INCLUDED IN THIS RUN, READ IT IN
*      FOR THIS REACH AND BUILD -INCWUA- ARRAY
          IF(INCUB .NE. 1H ) CALL EFFSPN(R,NDIS(R),NI,INCSP,
     .        DIVQ,FLOW,WUA,    INCWUA)


  703 CONTINUE
```

```
      DO 750 R=1,NR
          WRITE(6,65) RUNLABL,R
      DO 750 D=1,NDIS(R)
          GROSS(R,D) = NINT(FACTOR(R)*FLOAT(GROSS(R,D)))
          DO 751 L=1,NSTG
              WUA(R,D,L) = NINT(FACTOR(R)*FLOAT(WUA(R,D,L))
     .            * SFACTR(R,L) )
  751     CONTINUE
          WRITE(6,66) FLOW(R,D),(WUA(R,D,L),L=1,NSTG),GROSS(R,D)
          IF(INCUB .NE. 1H ) THEN
              DO 752 I=1,NI
                  L = INCSP(I)
                  INCWUA(R,D,I) = NINT(FACTOR(R) *
     .                FLOAT(INCWUA(R,D,I)) * SFACTR(R,L))
  752         CONTINUE
          ENDIF
  750 CONTINUE

   11 FORMAT(I5,14I9/5X,14I9/5X,13I9)
   61 FORMAT(1X,'ERROR...MORE THAN',I4,' DISCRETE DISCHARGES IN ',
     .       'REACH',I3,',  PROGRAM STOPPED - MUST RE-DIMENSION ',
     .       'ARRAYS.')
   65 FORMAT(1H1,6A10/
     .       1X,'REACH',I3,', WUA AND GROSS AREA AFTER REACH',
     .       ' FACTOR ADJUSTMENT FOR REACH LENGTH AND SPECIES',
     .       ' DISTRIBUTION:'/)
   66 FORMAT(1X,I5,14I9/5X,14I9/5X,13I9)

*-----READ IN THE NC CONDITIONS FOR EACH REACH
*     ONE LINE PER EXCEEDANCE PROBABILITY, NM CONDITIONS ON A LINE

      DO 704 R=1,NR
      DO 704 P=1,NP
          I1 = 1 + NM*(P-1)
          I2 = P*NM
          READ(3,31) (COND(R,I),I=I1,I2)
  704 CONTINUE

   31 FORMAT(5X,12I5)

*-----NOW LOOP ON THE NC CONDITIONS, SUMMING WUA AND GROSS AREA
*     ACROSS REACHES AFTER ADJUSTING THE DISCHARGE VALUES.
*     TABLE THE RESULTS IN UP TO THREE TABLES.

      ND = (MAXQ - MINQ)/DIVQ + 1
      NSUB = ND/15
      LASTQ = 15
      IF(MOD(ND,15) .NE. 0) THEN
          NSUB = NSUB + 1
          LASTQ = MOD(ND,15)
      ENDIF

      DO 710 C=BEGC,ENDC

          DO 711 D=1,MAXD
          SUMG(D) = 0.
          DO 711 L=1,MAXSTG
              SUMW(D,L) = 0.
  711     CONTINUE
```

```
      DO 712 R=1,NR
          D2 = 0
          DO 713 D=1,NDIS(R)
              ADJFLO = FLOW(R,D) - COND(R,C)
              IF(ADJFLO.GE.MINQ .AND. ADJFLO.LE.MAXQ) THEN
                  D2 = D2 + 1
                  SUMQ(D2) = SUMQ(D2) + GROSS(R,D)
                  DO 714 L=1,NSTG
                      IF(INCUB .EQ. 1H ) THEN
                          SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
                        ELSE
                          IS = 0
                          DO 735 I=1,NI
                              IF(L .EQ. INCSP(I)) IS = I
735                       CONTINUE
                          IF(IS .EQ. 0) THEN
                              SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
                            ELSEIF(ICMAT(IS,C) .EQ. 0) THEN
                              SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
                            ELSE
                              SUMW(D2,L) = SUMW(D2,L)+INCWUA(R,D,IS)
                          ENDIF
                      ENDIF
714                   CONTINUE
                  ENDIF
713           CONTINUE
              IF(D2 .NE. ND) THEN
                  WRITE(6,62) R,MINQ,MAXQ,COND(R,C),(CLABL(I,C),I=1,8)
                  STOP 2
              ENDIF
 62 FORMAT(1HO,'ERROR IN REACH',I3,' - RANGE OF DISCHARGES DOES ',
     .       'NOT COVER THE DESIRED RANGE (',I5,' TO ',I5,') AFTER ',
     .       'SUBTRACTING TRIBUTARY'/' INFLOW VALUE (',I5,') FOR THE ',
     .       'FOLLOWING CONDITION: '/1X,8A10)
712       CONTINUE

          DO 715 L=1,NSTQ
              WMAX(L) = 0.
              DO 716 D=1,ND
                  WMAX(L) = MAX(WMAX(L),SUMW(D,L))
716           CONTINUE
              DO 717 D=1,ND
                  TBL1(D,L) = 100.*(SUMW(D,L)/WMAX(L))
                  IF(SUMW(D,L) .EQ. WMAX(L)) TBL1(D,L) = 100.
                  SUMW(D,L) = 100.*(SUMW(D,L)/SUMQ(D))
                  IF(PTBL3) TBL3(D,L) = (TBL1(D,L)/SUMQ(D)) * T3SCL
717           CONTINUE
715       CONTINUE

          Q = MINQ - DIVQ
          NQ = 15
          DO 718 I=1,NSUB
              IF(MOD(I,2).NE.0 .OR. NSTG.GT.27) THEN
                  IF(PTBL1) WRITE(4,41) (CLABL(K,C),K=1,7),RUNLABL
                  IF(PTBL2) WRITE(7,71) (CLABL(K,C),K=1,7),RUNLABL
                  IF(PTBL3) WRITE(8,81) (CLABL(K,C),K=1,7),RUNLABL
              ENDIF
              DO 719 J=1,15
```

```
                QLABL(J) = 1H
  719           CONTINUE
                IF(I .EQ. NSUB) NQ = LASTQ
                DO 720 J=1,NQ
                    Q = Q + DIVQ
                    CALL INTCOD(QLABL(J),1,7,Q)
  720           CONTINUE
                IF(I.EQ.1 .AND. PTBL1) WRITE(4,45) QLABL
                IF(I.GT.1 .AND. PTBL1) WRITE(4,42) QLABL
                IF(PTBL2) WRITE(7,42) QLABL
                IF(PTBL3) WRITE(8,42) QLABL
                D1 = 15*(I - 1) + 1
                D2 = MIN(D1+14,ND)
                DO 721 L=1,NSTQ
                    IF(I.EQ.1 .AND. PTBL1) THEN
                        IWMAX = NINT(WMAX(L))
                        WRITE(4,46) SPLABL(1,L),SPLABL(2,L),
     .                      (TBL1(D,L),D=D1,D2),IWMAX
                    ENDIF
                    IF(I.GT.1 .AND. PTBL1) WRITE(4,43) SPLABL(1,L),
     .                  SPLABL(2,L),(TBL1(D,L),D=D1,D2)
                    IF(PTBL2) WRITE(7,43) SPLABL(1,L),SPLABL(2,L),
     .                  (SUMW(D,L),D=D1,D2)
                    IF(PTBL3) WRITE(8,44) SPLABL(1,L),SPLABL(2,L),
     .                  (TBL3(D,L),D=D1,D2)
  721           CONTINUE
  718       CONTINUE

   41 FORMAT(1H1,'WEIGHTED USABLE AREA AS A PERCENT OF THE ',
     .        'GREATEST WUA ACROSS ALL DISCHARGES REFERENCED TO ',
     .        'GORGE POWERHOUSE, BY SPECIES/LIFE STAGE'/
     .        1X,7A10,5X,6A10)
   71 FORMAT(1H1,'WEIGHTED USABLE AREA AS A PERCENT OF THE GROSS ',
     .        'AREA FOR EACH DISCHARGE, BY SPECIES/LIFE STAGE'/
     .        1X,7A10,5X,6A10)
   81 FORMAT(1H1,'WUA AS PERCENT OF THE GREATEST WUA ACROSS ALL ',
     .        'DISCHARGES, THEN DIVIDED BY GROSS AREA FOR THAT ',
     .        'DISCHARGE (X.XXXX E-06)'/1X,7A10,5X,6A10)
   42 FORMAT(1HO,'SPECIES/LIFE STAGE',3X,15A7/
     .        1X,20(1H-),1X,105(1H-))
   43 FORMAT(1X,2A10,1X,15F7.2)
   44 FORMAT(1X,2A10,1X,15F7.4)
   45 FORMAT(1HO,'SPECIES/LIFE STAGE',3X,15A7,3X,'MAX WUA'/
     .        1X,20(1H-),1X,105(1H-),2X,8(1H-))
   46 FORMAT(1X,2A10,1X,15F7.2,I10)

  710 CONTINUE

      STOP
      END
      SUBROUTINE EFFSPN(R,MAXD,NI,INCSP,DIVQ,FLOW,WUA,   INCWUA)

*     READS INCUBATION DATA FOR ONE REACH, AND CONSTRUCTS THE
*     COMPANION ARRAY TO -WUA- WHICH CONTAINS INCUBATION WUA FOR
*     THE AFFECTED FLOWS (-INCWUA-)

*-----ARGUMENTS:
      INTEGER     NI, FLOW(10,150), WUA(10,150,40), INCWUA(10,150,5),
     .            R, MAXD, INCSP(5), DIVQ
```

```
*-----LOCAL VARIABLES:
      INTEGER     LINEIN(2), Q, NQ, NSP, MINQ, MAXQ
      REAL        XWUA

*-----INITIALIZATION:
      NSP = 0

*-----START:
  601 CALL READH(9,LINEIN,2,IREP)
      IF(IREP .EQ. -2) RETURN
      IF(LINEIN(1) .NE. 10HSPAWNING -) GOTO 601
      NSP = NSP + 1
      CALL READH(9,LINEIN,2,IREP)
      CALL READH(9,LINEIN,2,IREP)
      CALL READH(9,LINEIN,2,IREP)
      CALL INTMOV(LINEIN,8,5,MAXQ)
      CALL READH(9,LINEIN,2,IREP)
      CALL INTMOV(LINEIN,8,5,MINQ)
      NQ = (MAXQ-MINQ)/DIVQ + 1
      DO 701 Q=1,NQ-1
          CALL READH(9,LINEIN,2,IREP)
  701 CONTINUE
      L = INCSP(NSP)
      D = 0
  501 D = D + 1
      IF(FLOW(R,D) .LT. MINQ) INCWUA(R,D,NSP) = WUA(R,D,L)
      IF(FLOW(R,D) .EQ. MINQ) GOTO 502
      IF(D .EQ. MAXD) THEN
          WRITE(6,61) R,NSP,MINQ,FLOW(R,1)
   61     FORMAT(1HO,'INCONSISTENCY IN FLOW DATA...REACH',I2,
     .        ', INCUBATION SPECIES',I3,'.'/
     .            1X,'NO FLOWS IN REGULAR REACH DATA ARE LESS THAN ',
     .            'OR EQUAL TO THE LOWEST INCUBATION Q (',I5,').'/
     .            1X,'LOWEST Q IN REGULAR DATA IS: ',I5,'.'/
     .            1X,'PROGRAM STOPPED.')
          STOP 3
      ENDIF
      GOTO 501

  502 DO 702 Q=1,NQ
          CALL READH(9,LINEIN,2,IREP)
          CALL REALMOV(LINEIN,5,11,2,XWUA)
          INCWUA(R,D,NSP) = NINT(XWUA)
          D = D + 1
  702 CONTINUE
      IWUA = NINT(XWUA)
      DO 703 I=D,MAXD
          INCWUA(R,I,NSP) = IWUA
  703 CONTINUE

*-----READ UP TO END-OF-FILE ON TAPE9 (END OF REACH)
      GOTO 601

      END
```

```
FILE: WUA2A

      PROGRAM WUA2(TAPE1,TAPE2,TAPE3,TAPE4,TAPE7,TAPE8,TAPE9,
     .            INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)

C*********************************************************************
C*    READS (1) TABLED WUA AND GROSS AREA BY DISCHARGE, FOR EACH     *
C*              REACH,                                               *
C*          (2) OPTIONALLY, INCUBATION FLOWS AND WUA FOR SELECTED    *
C*              SPECIES (TO OVERRIDE ABOVE IN CERTAIN MONTHS)        *
C*          (2) NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH        *
C*          (3) A LABEL FILE                                         *
C*    AND PRODUCES 3 TABLES OF OUTPUT (SUMMED ACROSS REACHES):       *
C*          (1) WUA FOR EACH DISCHARGE AS PERCENT OF THE GREATEST    *
C*              WUA, FOR EACH LIFE STAGE                             *
C*          (2) WUA FOR EACH DISCHARGE AS PERCENT OF THE GROSS       *
C*              AREA, FOR EACH LIFE STAGE                            *
C*          (3) TABLE 1 PERCENTS DIVIDED BY GROSS AREA FOR EACH      *
C*              DISCHARGE                                            *
C*********************************************************************

C-----FILES:
C
C     TAPE1 = INPUT, WUA TABLES PRODUCED BY PROGRAM WUA1, END-OF-
C             FILE BETWEEN TABLES FOR EACH REACH
C     TAPE2 = INPUT, LABEL FILE
C     TAPE3 = INPUT, NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH
C     TAPE9 = INPUT, INCUBATION DATA, EOF BETWEEN REACHES
C     TAPE4 = OUTPUT, TABLED WUA AS PERCENT OF GREATEST WUA
C     TAPE7 = OUTPUT, TABLED WUA AS PERCENT OF GROSS AREA
C     TAPE8 = OUTPUT, TABLE 1 PERCENTS DIVIDED BY GROSS AREA
C     TAPE5 = STANDARD INPUT FILE
C     TAPE6 = STANDARD OUTPUT FILE (ERROR MESSAGES, DIAGNOSTICS)

C-----LOCAL VARIABLES:

      INTEGER     MINQ, MAXQ, DIVQ, NR, NSTQ, NC, MAXR, MAXSTQ, MAXD,
     .            REACH, ND, Q, R, D, C, NSUB, D1, D2, NDIS(10), ADJFLO,
     .            GROSS(10,150), FLOW(10,150),
     .            COND(10,36), SPLABL(2,40), CLABL(7,36), QLABL(15),
     .            BEGC, ENDC, RUNLABL(6), DUM, INCUB, INCSP(8),
     .            INCMO(8,7), ICMAT(8,36), INCWUA(10,150,8),
     .            SCALE, T3SCL, TBLOPT(2)

      REAL        TBL1(150,40), SUMW(150,40), SUMG(150), WMAX(40),
     .            FACTOR(10), SFACTR(10,40), TBL3(150,40),
     .            WUA(10,150,40)

      LOGICAL     PTBL1, PTBL2, PTBL3

C     DEFINITIONS
C     MINQ,MAXQ = MINIMUM AND MAXIMUM DISCHARGE VALUES TO BE TABLED
C     DIVQ =      SIZE OF INTERVAL BETWEEN TABLED DISCHARGE VALUES
C     NR =        NUMBER OF REACHES BEING PROCESSED
C     NSTQ =      NUMBER OF SPECIES-LIFE STAGE CODES
C     NC =        NUMBER OF TRIBUTARY INFLOW CONDITIONS TO BE TABLED
C     MAXR =      MAXIMUM NUMBER OF REACHES THAT CAN BE PROCESSED
C                 WITHOUT RE-DIMENSIONING ARRAYS
C     MAXSTQ =    MAXIMUM NUMBER OF SPECIES-LIFE STAGE CODES
C     MAXD =      MAXIMUM NUMBER OF DISCHARGE LEVELS (BEFORE ADJUSTING FOR
```

```
*                TRIBUTARY INFLOW CONDITIONS AND RESTRICTING TO MINQ-MAXQ)
*     NSUB =     NUMBER OF SUB-TABLES, WITH 15 DISCHARGE LEVELS PER
*                SUB-TABLE
*     WUA(R,D,L) = WEIGHTED USABLE AREA FOR REACH R, DISCHARGE D,
*                LIFE STAGE CODE L
*     GROSS(R,D) = GROSS AREA FOR REACH R, DISCHARGE D
*     FLOW(R,D) = DISCHARGE VALUE FOR REACH R, DISCHARGE D
*     COND(R,C) = TRIBUTARY INFLOW CONDITION FOR REACH R, CONDITION C
*     SPLABL(I,L) = SPECIES-LIFE STAGE LABEL FOR LIFE STAGE L
*     CLABL(I,C) = TABLE HEADING FOR CONDITION C
*     QLABL(I) =  USED IN SUB-TABLE HEADING (15 FLOW VALUES)
*     TBL1(D,L) = TABLE 1 VALUES FOR DISCHARGE LEVEL D, LIFE STAGE L
*                (AFTER RESTRICTING ADJUSTED FLOWS TO MINQ-MAXQ)
*     SUMW(D,L) = WUA SUMMED ACROSS REACHES FOR DISCHARGE LEVEL D,
*                LIFE STAGE L (THEN BECOMES TABLE 2 VALUES AFTER DIVIDING
*                BY GROSS AREA)
*     TBL3(D,L) = TABLE 1 PERCENTS DIVIDED BY GROSS AREA FOR DISCHARGE D
*     SUMG(D) =   GROSS AREA SUMMED ACROSS REACHES FOR DISCHARGE LEVEL D
*     WMAX(L) =   MAXIMUM WUA ACROSS ALL DISCHARGE LEVELS FOR A GIVEN
*                LIFE STAGE L
*     FACTOR(R) = MULTIPLICATIVE FACTOR FOR REACH R, EXPANDS WUA
*                AND GROSS AREA VALUES TO REPRESENT A SECTION
*                OF THE RIVER
*     SFACTR(R,L) = SPECIES-DISTRIBUTION REACH WEIGHTING FACTOR FOR
*                REACH R AND SPECIES-LIFE STAGE L
*     BEGC,ENDC = BEGINNING AND ENDING INDEX WITHIN CONDITION
*                ARRAYS FOR A SUBSET OF THE NC CONDITIONS TO BE USED
*                IN ANY GIVEN RUN
*     NP =       NUMBER OF EXCEEDANCE PROBABILITES
*     NM =       NUMBER OF MONTHS AT EACH EXCEEDANCE PROBABILITY
*
*     INCUBATION DATA:
*
*     INCUB =    FLAG, BLANK IF NO INCUBATION DATA ON THIS RUN
*     INCSP(I) =  SPECIES NUMBERS FOR WHICH INCUBATION DATA WILL BE
*                PROVIDED, WHERE INCSP(I) = L AND L IS IN THE RANGE
*                (1 TO NSTG), I.E. THE SPECIES-LIFE STAGE NUMBERS USED
*                IN THE BASIC DATA
*     INCMO(I,J) = J-TH MONTH NUMBER IN THE INCUBATION PERIOD FOR SPECIES
*                L, WHERE L = INCSP(I).  MONTHS ARE NUMBERED RELATIVE
*                TO THE FIRST MONTH USED FOR CONDITIONS, E.G. IF OCT=1
*                THEN INCUBATION MONTH DEC=3.
*     ICMAT(I,C) = CONSTRUCTED MATRIX THAT INDICATES IF CONDITION C
*                REPRESENTS AN INCUBATION MONTH FOR SPECIES L, WHERE
*                L = INCSP(I)  (1=YES, 0=NO)
*     INCWUA(R,D,I) = COMPANION ARRAY TO WUA(R,D,L).  INCUBATION WUA
*                FOR REACH R, DISCHARGE D AND SPECIES L, WHERE
*                L = INCSP(I) (INCUBATION SPECIES I)
*
*      TABLING OPTIONS:
*
*      PTBL1,PTBL2,PTBL3 = LOGICAL FLAGS INDICATING IF PERCENT
*                TABLES 1,2 AND/OR 3 SHOULD BE PRINTED
*      TBLOPT(2) = COL 1-20 OF TABLE OPTION RECORD - SHOULD
*                CONTAIN THE NUMBERS 1,2 AND/OR 3
*      SCALE =    COL 26-35 OF TABLE OPTION RECORD - IF TABLE 3
*                IS TO BE PRINTED, THIS CONTAINS A NUMBER
*                WHICH IS THE HABITAT EFFICIENCY SCALING FACTOR
*                (A POWER OF 10)
*      T3SCL =    HABITAT EFFICIENCY SCALING FACTOR FOR TABLE 3
```

```
*-----CONSTANTS:
      DATA MAXR,MAXSTG,MAXD/10,40,150/

*-----INITIALIZATION:
      CALL FILEB(9)
      PTBL1 = PTBL2 = PTBL3 = .FALSE.

*-----READ TABLE OPTION DATA RECORD AND DECODE
      READ(5,55) TBLOPT,SCALE
      CALL ISCAN(TBLOPT,1,20,1H1,1,1,I)
      IF(I .NE. 0) PTBL1 = .TRUE.
      CALL ISCAN(TBLOPT,1,20,1H2,1,1,I)
      IF(I .NE. 0) PTBL2 = .TRUE.
      CALL ISCAN(TBLOPT,1,20,1H3,1,1,I)
      IF(I .NE. 0) PTBL3 = .TRUE.
      IF(PTBL3) THEN
          CALL ISCAN(SCALE,1,10,9H123456789,1,9,I)
          IF(I .EQ. 0) THEN
              WRITE(6,60)
              STOP 0
   60         FORMAT(1HO,'ERROR...YOU HAVE REQUESTED PERCENT TABLE 3,',
     .                ' BUT HAVE NOT GIVEN A HABITAT EFFICIENCY '/
     .                6X,'SCALING FACTOR IN COL. 26-35 OF THE TABLE ',
     .                'OPTION DATA RECORD.')
          ENDIF
          CALL INTMOV(SCALE,I,1,T3SCL)
          T3SCL = 10**T3SCL
      ENDIF

*-----READ PARAMETERS, SPECIES LABELS, CONDITION LABELS
      READ(5,50) RUNLABL
      READ(5,51) MINQ,MAXQ,DIVQ,NR,BEGC,ENDC,NP,NM
      DO 700 R=1,NR
          READ(5,52) FACTOR(R)
  700 CONTINUE

      INCUB = 1H
      GO TO 200
      READ(5,53) INCUB,NI
      IF(EOF(5) .NE. 0) GOTO 200
      IF(INCUB .EQ. 1H ) GOTO 200
      DO 730 I=1,NI
          READ(5,54) INCSP(I),(INCMO(I,J),J=1,7)
  730 CONTINUE
  200 CONTINUE

      READ(2,21) NSTQ
      DO 701 L=1,NSTQ
          READ(2,22) SPLABL(1,L),SPLABL(2,L),(SFACTR(R,L),R=1,NR)
          DO 701 R=1,NR
              IF(SFACTR(R,L).EQ.0.) SFACTR(R,L)=1.
  701 CONTINUE
      READ(2,21) NC
      IF(NC .NE. NP*NM) THEN
          WRITE(6,63) NC,NP,NM
          STOP 3
   63     FORMAT(1HO,'ERROR...NUMBER OF CONDITIONS SPECIFIED IN ',
     .           'LABEL FILE IS',I3,'.'/6X,'THIS DOES NOT EQUAL ',
     .           'THE PRODUCT OF THE NUMBER OF EXCEEDANCE ',
```

```
     .            'PROBABILITIES (',I2,') TIMES THE NUMBER OF ',
     .            'MONTHS (',I2,')'/6X,'GIVEN ON THE RUN PARAMETERS ',
     .            'CARD. '/6X,'PROGRAM STOPPED. ')
          ENDIF
          DO 702 C=1,NC
              READ(2,23) (CLABL(K,C),K=1,7)
      702 CONTINUE

       21 FORMAT(I2)
       22 FORMAT(10X,2A10,7F5.4)
       23 FORMAT(7A10)
       50 FORMAT(6A10)
       51 FORMAT(8I5)
       52 FORMAT(F6.3)
       53 FORMAT(A5,1X,I2)
       54 FORMAT(I2,5X,7I2)
       55 FORMAT(2A10,5X,A10)

*-----IF INCUBATION IS TO BE INCLUDED IN THIS RUN, CONSTRUCT THE
*     -ICMAT- ARRAY (WHICH CONDITIONS ARE INCUBATION MONTHS FOR
*     WHICH SPECIES)
          IF(INCUB .NE. 1H ) THEN
              DO 731 I=1,NI
                  DO 732 C=1,NC
                      ICMAT(I,C) = 0
      732         CONTINUE
                  DO 733 J=1,7
                      IF(INCMO(I,J) .EQ. 0) GOTO 733
                      M = INCMO(I,J)
                      DO 734 P=1,NP
                          C = M + NM*(P-1)
                          ICMAT(I,C) = 1
      734             CONTINUE
      733         CONTINUE
      731     CONTINUE
          ENDIF

*-----READ IN THE WUA, GROSS AREA, AND DISCHARGE VALUES FOR THE
*     NR REACHES (EACH REACH SEPERATED BY END-OF-FILE)

          DO 703 R=1,NR
              D = 0
      601     D = D + 1
              IF(D .GT. MAXD) THEN
                  READ(1,11,END=602) DUM
                  WRITE(6,61) MAXD,R
                  STOP 1
              ENDIF
              READ(1,11,END=602) FLOW(R,D),(WUA(R,D,I),I=1,NSTG),
     .            GROSS(R,D)
              GOTO 601
      602     NDIS(R) = D - 1
*             CLEAR END-OF-FILE INDICATOR BEFORE READING IN NEXT REACH
              IF(EOF(1) .NE. 0.) CONTINUE

*-----IF INCUBATION DATA IS BEING INCLUDED IN THIS RUN, READ IT IN
*     FOR THIS REACH AND BUILD -INCWUA- ARRAY
              IF(INCUB .NE. 1H ) CALL EFFSPN(R,NDIS(R),NI,INCSP,
     .            DIVQ,FLOW,WUA,    INCWUA)
```

```
  703 CONTINUE

      DO 750 R=1,NR
          WRITE(6,65) RUNLABL,R
      DO 750 D=1,NDIS(R)
          GROSS(R,D) = NINT(FACTOR(R)*FLOAT(GROSS(R,D)))
          DO 751 L=1,NSTG
              WUA(R,D,L) = FACTOR(R)*WUA(R,D,L)
                  * SFACTR(R,L)
  751     CONTINUE
          WRITE(6,66) FLOW(R,D),(WUA(R,D,L),L=1,NSTG),GROSS(R,D)
          IF(INCUB .NE. 1H ) THEN
              DO 752 I=1,NI
                  L = INCSP(I)
                  INCWUA(R,D,I) = NINT(FACTOR(R) *
                      FLOAT(INCWUA(R,D,I)) * SFACTR(R,L))
  752         CONTINUE
          ENDIF
  750 CONTINUE

   11 FORMAT(I5,9F9.5)
   61 FORMAT(1X,'ERROR...MORE THAN',I4,' DISCRETE DISCHARGES IN ',
     .       'REACH',I3,'.  PROGRAM STOPPED - MUST RE-DIMENSION ',
     .       'ARRAYS.')
   65 FORMAT(1H1,6A10/
     .       1X,'REACH',I3,',  WUA AND GROSS AREA AFTER REACH',
     .       ' FACTOR ADJUSTMENT FOR REACH LENGTH AND SPECIES',
     .       ' DISTRIBUTION:'/)
   66 FORMAT(1X,I5,9F9.5)

*-----READ IN THE NC CONDITIONS FOR EACH REACH
*     ONE LINE PER EXCEEDANCE PROBABILITY, NM CONDITIONS ON A LINE

      DO 704 R=1,NR
      DO 704 P=1,NP
          I1 = 1 + NM*(P-1)
          I2 = P*NM
          READ(3,31) (COND(R,I),I=I1,I2)
  704 CONTINUE

   31 FORMAT(5X,12I5)

*-----NOW LOOP ON THE NC CONDITIONS, SUMMING WUA AND GROSS AREA
*     ACROSS REACHES AFTER ADJUSTING THE DISCHARGE VALUES.
*     TABLE THE RESULTS IN UP TO THREE TABLES.

      ND = (MAXQ - MINQ)/DIVQ + 1
      NSUB = ND/15
      LASTQ = 15
      IF(MOD(ND,15) .NE. 0) THEN
          NSUB = NSUB + 1
          LASTQ = MOD(ND,15)
      ENDIF

      DO 710 C=BEGC,ENDC

          DO 711 D=1,MAXD
          SUMG(D) = 0.
          DO 711 L=1,MAXSTG
```

```
          SUMW(D,L) = 0.
711       CONTINUE

          DO 712 R=1,NR
              D2 = 0
              DO 713 D=1,NDIS(R)
                  ADJFLO = FLOW(R,D) - COND(R,C)
                  IF(ADJFLO.GE.MINQ .AND. ADJFLO.LE.MAXQ) THEN
                      D2 = D2 + 1
                      SUMQ(D2) = SUMQ(D2) + GROSS(R,D)
                      DO 714 L=1,NSTQ
                          IF(INCUB .EQ. 1H ) THEN
                              SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
                            ELSE
                              IS = 0
                              DO 735 I=1,NI
                                  IF(L .EQ. INCSP(I)) IS = I
735                           CONTINUE
                              IF(IS .EQ. 0) THEN
                                  SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
                                ELSEIF(ICMAT(IS,C) .EQ. 0) THEN
                                  SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
                                ELSE
                                  SUMW(D2,L) = SUMW(D2,L)+INCWUA(R,D,IS)
                              ENDIF
                          ENDIF
714                   CONTINUE
                  ENDIF
713           CONTINUE
              IF(D2 .NE. ND) THEN
                  WRITE(6,62) R,MINQ,MAXQ,COND(R,C),(CLABL(I,C),I=1,8)
                  STOP 2
              ENDIF
 62 FORMAT(1HO,'ERROR IN REACH',I3,' -- RANGE OF DISCHARGES DOES ',
     .    'NOT COVER THE DESIRED RANGE (',I5,' TO ',I5,') AFTER ',
     .    'SUBTRACTING TRIBUTARY'/' INFLOW VALUE (',I5,') FOR THE ',
     .    'FOLLOWING CONDITION: '/1X,8A10)
712       CONTINUE


          DO 715 L=1,NSTQ
              WMAX(L) = 0.
              DO 716 D=1,ND
                  WMAX(L) = MAX(WMAX(L),SUMW(D,L))
716           CONTINUE
              DO 717 D=1,ND
                  TBL1(D,L) = 100.*(SUMW(D,L)/WMAX(L))
                  IF(SUMW(D,L) .EQ. WMAX(L)) TBL1(D,L) = 100.
                  IF (PTBL2) SUMW(D,L) = 100.*(SUMW(D,L)/SUMQ(D))
                  IF(PTBL3) TBL3(D,L) = (TBL1(D,L)/SUMQ(D)) * T3SCL
717           CONTINUE
715       CONTINUE

          Q = MINQ - DIVQ
          NQ = 15
          DO 718 I=1,NSUB
              IF(I .EQ. 1 .OR. NSTQ.GT.27) THEN
                  IF(PTBL1) WRITE(4,41) (CLABL(K,C),K=1,7),RUNLABL
                  IF(PTBL2) WRITE(7,71) (CLABL(K,C),K=1,7),RUNLABL
                  IF(PTBL3) WRITE(8,81) (CLABL(K,C),K=1,7),RUNLABL
```

```
            ENDIF
            DO 719 J=1,15
                QLABL(J) = 1H
  719       CONTINUE
            IF(I .EQ. NSUB) NQ = LASTQ
            DO 720 J=1,NQ
                Q = Q + DIVQ
                CALL INTCOD(QLABL(J),1,7,Q)
  720       CONTINUE
            IF(I.EQ.1 .AND. PTBL1) WRITE(4,45) QLABL
            IF(I.GT.1 .AND. PTBL1) WRITE(4,42) QLABL
            IF(PTBL2) WRITE(7,42) QLABL
            IF(PTBL3) WRITE(8,42) QLABL
            D1 = 15*(I - 1) + 1
            D2 = MIN(D1+14,ND)
            DO 721 L=1,NSTG
                IF(I.EQ.1 .AND. PTBL1) THEN
                    WRITE(4,46) SPLABL(1,L),SPLABL(2,L),
                        (TBL1(D,L),D=D1,D2),WMAX(L)
                ENDIF
                IF(I.GT.1 .AND. PTBL1) WRITE(4,43) SPLABL(1,L),
                    SPLABL(2,L),(TBL1(D,L),D=D1,D2)
                IF(PTBL2) WRITE(7,43) SPLABL(1,L),SPLABL(2,L),
                    (SUMW(D,L),D=D1,D2)
                IF(PTBL3) WRITE(8,44) SPLABL(1,L),SPLABL(2,L),
                    (TBL3(D,L),D=D1,D2)
  721       CONTINUE
  718   CONTINUE

   41 FORMAT(1H1,'WEIGHTED USABLE AREA AS A PERCENT OF THE ',
     .       'GREATEST WUA ACROSS ALL DISCHARGES REFERENCED TO ',
     .       'GORGE POWERHOUSE, BY SPECIES/LIFE STAGE'/
     .      1X,7A10,5X,6A10)
   71 FORMAT(1H1,'WEIGHTED USABLE AREA AS A PERCENT OF THE GROSS ',
     .       'AREA FOR EACH DISCHARGE, BY SPECIES/LIFE STAGE'/
     .      1X,7A10,5X,6A10)
   81 FORMAT(1H1,'WUA AS PERCENT OF THE GREATEST WUA ACROSS ALL ',
     .       'DISCHARGES, THEN DIVIDED BY GROSS AREA FOR THAT ',
     .       'DISCHARGE (X.XXXX E-06)'/1X,7A10,5X,6A10)
   42 FORMAT(1HO,'SPECIES/LIFE STAGE',3X,15A7/
     .      1X,20(1H-),1X,105(1H-))
   43 FORMAT(1X,2A10,1X,15F7.3)
   44 FORMAT(1X,2A10,1X,15F7.4)
   45 FORMAT(1HO,'SPECIES/LIFE STAGE',3X,15A7,3X,'MAX WUA'/
     .      1X,20(1H-),1X,105(1H-),2X,8(1H-))
   46 FORMAT(1X,2A10,1X,15F7.3,F10.3)

  710 CONTINUE

      STOP
      END
      SUBROUTINE EFFSPN(R,MAXD,NI,INCSP,DIVQ,FLOW,WUA,  INCWUA)

*     READS INCUBATION DATA FOR ONE REACH, AND CONSTRUCTS THE
*     COMPANION ARRAY TO -WUA- WHICH CONTAINS INCUBATION WUA FOR
*     THE AFFECTED FLOWS (-INCWUA-)

*-----ARGUMENTS:
      INTEGER     NI, FLOW(10,150), WUA(10,150,40), INCWUA(10,150,5),
     .            R, MAXD, INCSP(5), DIVQ
```

```
*-----LOCAL VARIABLES:
      INTEGER      LINEIN(2), Q, NQ, NSP, MINQ, MAXQ
      REAL         XWUA

*-----INITIALIZATION:
      NSP = 0

*-----START:
  601 CALL READH(9,LINEIN,2,IREP)
      IF(IREP .EQ. -2) RETURN
      IF(LINEIN(1) .NE. 10HSPAWNING -) GOTO 601
      NSP = NSP + 1
      CALL READH(9,LINEIN,2,IREP)
      CALL READH(9,LINEIN,2,IREP)
      CALL READH(9,LINEIN,2,IREP)
      CALL INTMOV(LINEIN,8,5,MAXQ)
      CALL READH(9,LINEIN,2,IREP)
      CALL INTMOV(LINEIN,8,5,MINQ)
      NQ = (MAXQ-MINQ)/DIVQ + 1
      DO 701 Q=1,NQ-1
         CALL READH(9,LINEIN,2,IREP)
  701 CONTINUE
      L = INCSP(NSP)
      D = 0
  501 D = D + 1
      IF(FLOW(R,D) .LT. MINQ) INCWUA(R,D,NSP) = WUA(R,D,L)
      IF(FLOW(R,D) .EQ. MINQ) GOTO 502
      IF(D .EQ. MAXD) THEN
         WRITE(6,61) R,NSP,MINQ,FLOW(R,1)
   61    FORMAT(1HO,'INCONSISTENCY IN FLOW DATA...REACH',I2,
     .        ', INCUBATION SPECIES',I3,'.'/
     .           1X,'NO FLOWS IN REGULAR REACH DATA ARE LESS THAN ',
     .           'OR EQUAL TO THE LOWEST INCUBATION Q (',I5,').'/
     .           1X,'LOWEST Q IN REGULAR DATA IS: ',I5,'.'/
     .           1X,'PROGRAM STOPPED.')
         STOP 3
      ENDIF
      GOTO 501

  502 DO 702 Q=1,NQ
         CALL READH(9,LINEIN,2,IREP)
         CALL REALMOV(LINEIN,5,11,2,XWUA)
         INCWUA(R,D,NSP) = NINT(XWUA)
         D = D + 1
  702 CONTINUE
      IWUA = NINT(XWUA)
      DO 703 I=D,MAXD
         INCWUA(R,I,NSP) = IWUA
  703 CONTINUE

*-----READ UP TO END-OF-FILE ON TAPE9 (END OF REACH)
      GOTO 601

      END
```

```
      PROGRAM WUA3(TAPE1,TAPE2,TAPE4,TAPE8,
     .             INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)

C*********************************************************************
C*    READS (1) A PERCENT-TABLE FILE PRODUCED BY WUA2                *
C*          (2) RUN PARAMETERS                                       *
C*          (3) A LABEL FILE                                         *
C*          (4) A FILE OF SPECIES/LIFE STAGE WEIGHTING FACTORS       *
C*              BY MONTH AND LIFE STAGE                              *
C*    AND PRODUCES A TABLE OF OUTPUT (SUMMED ACROSS REACHES):        *
C*          THE ORIGINAL PERCENTAGE TABLES AS CALCULATED IN WUA2,    *
C*          WEIGHTED AND SUMMED ACROSS SPECIES/LIFE STAGES           *
C*********************************************************************

C-----FILES:
C
C     TAPE1 = INPUT, WUA PERCENT TABLE FILE PRODUCED BY PROGRAM WUA2
C     TAPE2 = INPUT, LABEL FILE
C     TAPE4 = OUTPUT, TABLED WEIGHTED AND SUMMED WUA
C     TAPE5 = STANDARD INPUT FILE, RUN PARAMETERS
C     TAPE6 = STANDARD OUTPUT FILE (ERROR MESSAGES, DIAGNOSTICS)
C     TAPE8 = INPUT, SPECIES/LIFE STAGE PERCENTAGE WEIGHTING FACTORS,
C                BY MONTH

C-----LOCAL VARIABLES:

      INTEGER      MINQ, MAXQ, DIVQ, NR, NSTG, NC, MAXR, MAXSTG, MAXD,
     .             REACH, ND, Q, R, D, C, NSUB, D1, D2, NDIS(10), ADJFLO,
     .             SPLABL(2,40), CLABL(7,36), QLABL(15),
     .             BEGC, ENDC, RUNLABL(6), DUM, SUMLBL(12,13),
     .             PLABL(3), NP, NM, P, WTLABL(5)

      REAL         TBL1(150,40), ALLSP(36,150), SPECWT(12,40)
      INTEGER       LINEIN(14)

C     DEFINITIONS
C     MINQ,MAXQ = MINIMUM AND MAXIMUM DISCHARGE VALUES TO BE TABLED
C     DIVQ =      SIZE OF INTERVAL BETWEEN TABLED DISCHARGE VALUES
C     NR =        NUMBER OF REACHES BEING PROCESSED
C     NSTG =      NUMBER OF SPECIES-LIFE STAGE CODES
C     NC =        NUMBER OF TRIBUTARY INFLOW CONDITIONS TO BE TABLED
C     MAXR =      MAXIMUM NUMBER OF REACHES THAT CAN BE PROCESSED
C                 WITHOUT RE-DIMENSIONING ARRAYS
C     MAXSTG =    MAXIMUM NUMBER OF SPECIES-LIFE STAGE CODES
C     MAXD =      MAXIMUM NUMBER OF DISCHARGE LEVELS (BEFORE ADJUSTING FOR
C                 TRIBUTARY INFLOW CONDITIONS AND RESTRICTING TO MINQ-MAXQ)
C     NSUB =      NUMBER OF SUB-TABLES, WITH 15 DISCHARGE LEVELS PER
C                 SUB-TABLE
C     CLABL(I,C) = TABLE HEADING FOR CONDITION C
C     QLABL(I) =  USED IN SUB-TABLE HEADING (15 FLOW VALUES)
C     TBL1(D,L) = PERCENT TABLE VALUES FOR DISCHARGE LEVEL D, LIFE STAGE L
C     BEGC,ENDC = BEGINNING AND ENDING INDEX WITHIN CONDITION
C                 ARRAYS FOR A SUBSET OF THE 36 CONDITIONS TO BE USED
C                 IN ANY GIVEN RUN
C     SPECWT(M,L) = SPECIES/LIFE STAGE WEIGHTING FACTORS FOR COMBINING
C                 SPECIES PERCENTAGES FOR MONTH M (NOTE: ASSUMES 3
C                 INFLOW CONDITIONS PER MONTH) AND SPECIES/LIFE STAGE L
C     ALLSP(C,D) = TBL1, WEIGHTED BY SPECWT AND SUMMED ACROSS SPECIES
```

```
*                    FOR CONDITION C
*      NM =          NUMBER OF MONTHS
*      NP =          NUMBER OF EXCEEDANCE PROBABILITIES PER MONTH

*-----CONSTANTS:
      DATA MAXR,MAXSTG,MAXD/10,40,150/

*-----INITIALIZATION:
      CALL FILEB(1)

*-----READ PARAMETERS, SPECIES LABELS, CONDITION LABELS
      READ(5,50) RUNLABL
      READ(5,51) MING,MAXQ,DIVQ,NR,BEGC,ENDC

      READ(2,21) NSTG
      DO 701 L=1,NSTG
          READ(2,22) DUM
  701 CONTINUE
      READ(2,21) NC
      DO 702 C=1,NC
          READ(2,23) (CLABL(K,C),K=1,7)
  702 CONTINUE

   21 FORMAT(I2)
   22 FORMAT(10X,2A10,10F4.2)
   23 FORMAT(7A10)
   50 FORMAT(6A10)
   51 FORMAT(6I5)
   52 FORMAT(F6.3)

*-----WUA3: READ SUMMARY TABLE LABELS
      READ(2,21) NM
      DO 731 M=1,NM
          READ(2,24) (SUMLBL(M,I),I=1,13)
  731 CONTINUE
      READ(2,21) NP
      DO 733 P=1,NP
          READ(2,25) PLABL(P)
  733 CONTINUE

   24 FORMAT(13A10)
   25 FORMAT(A3)

*------WUA3: READ SPECIES/LIFE STAGE PERCENTAGE WEIGHTING FACTORS
      READ(8,50) WTLABL
      DO 734 M=1,NM
          READ(8,81) (SPECWT(M,L),L=1,NSTG)
  734 CONTINUE
   81 FORMAT(3X,30F4.2)


*-----NOW LOOP ON THE 36 CONDITIONS, READING THE PERCENT TABLE
*      FOR EACH CONDITION, AND CALCULATING THE NEW SUMMARY TABLE

      ND = (MAXQ - MING)/DIVQ + 1
      NSUB = ND/15
      LASTQ = 15
      IF(MOD(ND,15) .NE. 0) THEN
          NSUB = NSUB + 1
          LASTQ = MOD(ND,15)
```

```
      ENDIF

      DO 710 C=BEGC,ENDC

*------CALCULATE MONTH FROM CONDITION NUMBER (ASSUMES NM MONTHS PER
*     EXCEEDANCE PROBABILITY)
          M = 0
          DO 735 P=1,NP
              K1 = NM*(P-1) + 1
              K2 = NM*P
              IF(C.GE.K1 .AND. C.LE.K2) M = C - K1 + 1
  735     CONTINUE
          IF(M .EQ. 0) THEN
              WRITE(6,67) C,NP
              STOP 3
   67         FORMAT(1X,'ERROR IN CALCULATING MONTH FROM CONDITION',
     .             ' NUMBER',I4,'.  NUMBER OF EXCEEDANCE PROBABILITES',
     .             ' PER MONTH = ',I2)
          ENDIF

          DO 711 D=1,MAXD
              ALLSP(C,D) = 0.
  711     CONTINUE
          DO 712 I=1,NSUB
              D1 = 15*(I-1) + 1
              D2 = MIN(D1+14,ND)
  101         CALL READH(1,LINEIN,14,IREP)
              IF(IREP .NE. 0) THEN
                  WRITE(6,68) C,I
   68             FORMAT(1HO,'ERROR...END-OF-FILE ON PERCENT TABLE ',
     .                   'ENCOUNTERED WHEN SEARCHING FOR DATA FOR ',
     .                   'CONDITION',I3,', SUB-TABLE',I3,'.'/
     .                   1X,'PROGRAM STOPPED.')
                  STOP 21
              ENDIF
              IF(LINEIN(1) .NE. 10H ----------) GOTO 101
              DO 713 L=1,NSTG
                  CALL READH(1,LINEIN,14,IREP)
                  IF(IREP .NE. 0) THEN
                      WRITE(6,68) C,I
                      STOP 22
                  ENDIF
                  DECODE(140,1,LINEIN) (TBL1(D,L),D=D1,D2)
    1             FORMAT(22X,15F7.0)
  713         CONTINUE
  712     CONTINUE

          DO 715 L=1,NSTG
          DO 715 D=1,ND
              ALLSP(C,D) = ALLSP(C,D) + SPECWT(M,L)*TBL1(D,L)
  715     CONTINUE

  710 CONTINUE

*-----WUA3: PRINT SUMMARY TABLES BY MONTH
      DO 760 M=1,NM
          WRITE(4,41) RUNLABL,WTLABL,(SUMLBL(M,I),I=1,13)
          Q = MINQ - DIVQ
          NQ = 15
          DO 761 I=1,NSUB
```

```
            DO 762 J=1,15
                QLABL(J) = 1H
762         CONTINUE
            IF(I .EQ. NSUB) NQ = LASTQ
            DO 763 J=1,NQ
                Q = Q + DIVQ
                CALL INTCOD(QLABL(J),1,7,Q)
763         CONTINUE
            WRITE(4,42) QLABL
            D1 = 15*(I-1) + 1
            D2 = MIN(D1+14,ND)
            DO 764 P=1,NP
                C = M + NM*(P-1)
                WRITE(4,43) PLABL(P),(ALLSP(C,D),D=D1,D2)
764         CONTINUE
761     CONTINUE
760 CONTINUE

 41 FORMAT(1H1,6A10,10X,5A10//1X,13A10/)
 42 FORMAT(1HO,'EXCEEDANCE',51X,'DISCHARGE (CFS)'/
           1X,'PROBABILITY',5X,15A7/1X,11(1H-),5X,105(1H-)/)
 43 FORMAT(5X,A3,9X,15F7.3)

    STOP
    END
```

FILE: SUMMIT

```
/JOB
SUMMIT.
/ACCOUNT
FTN5,DB.
LGO,SPNQ.
REWIND,SPNQ.
SAVE,SPNQ.
/EOR
      PROGRAM SUMMIT(TAPE2,INPUT,OUTPUT,TAPE5=INPUT)
      DIMENSION INT(12),ISUM(12)
      DATA ISUM/12*0/
      READ (5,200)(INT(I),I=1,12)
      DO 10 J=1400,7000,100
        DO 20 I=1,12
   20     ISUM(I) = INT(I) + J
        WRITE(2,200)(ISUM(I),I=1,12)
        DO 30 I=1,12
   30     ISUM(I) = 0
   10 CONTINUE
  200 FORMAT(4I10,/,4I10,/,4I10)
      ENDFILE(2)
      STOP
      END
/EOR
```

pink	chum	steelhead	chinook	
200	200	200	200	.10
300	400	500	600	.50
700	800	800	900	.40

```
/EOF
```

FILE: ELEVATE

```
/JOB
ELVJOB.
/ACCOUNT
ATTACH,S8OUT.
ATTACH,FILESET=ELEVSET/M=W,NA.
GF,ELEV8.
FTN5.
LGO,ELEV8,S8OUT,S8ADJ1,S8ADJ2.
RF,S8ADJ1.
RF,S8ADJ2.
/EOR
      PROGRAM ELEVATE(TAPE1,TAPE2,TAPE8=/1240,
     .TAPE9=/1240,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)

      INTEGER A,B(5),NIF,NIFSET(5),NSET,NTRANS,TN(6),IDUM(2),
     .        NC,CELLNO
      REAL BED(199), BED1(150,199), BED2(150,199), BED3(150,199),
     .     DIS(150), Q, STAGE, DEPTH, STG(150,6)
      CHARACTER*10 LOCATE

      DATA (TN(I),I=1,5)/0,39,78,110,146/
      DATA (NIFSET(I),I=1,5)/30,30,30,30,30/
      DATA (B(I),I=1,5)/0,30,60,90,120/

      NC = 193
      NIF = 150
      NSET = 5
      NTRANS = 5

      DO 40 J =1,NC
        READ(1,50) BED(J)
        DO 30 I = 1,NIF
          BED1(I,J) = BED(J)
          BED2(I,J) = BED(J)
          BED3(I,J) = BED(J)
   30 CONTINUE
   40 CONTINUE
   50 FORMAT(1X,F10.2)

      DO 31 I=1,NIF
        DO 31 K=1,6
          STG(I,K)=0.
   31 CONTINUE

      DO 100 I= 1,NSET
       DO 90 J = 1,NTRANS
       A = B(I) + 0
       DO 80 K = 1,NIFSET(I)


   60  READ(2,110,END=500) LOCATE
        IF(LOCATE .NE. 'DETAILS') GO TO 60
        READ(2,130)IDUM(1)
   65   READ(2,120) Q, STAGE
          A = A + 1
          DIS(A) = Q
          STG(A,J)=STAGE
```

```
      READ(2,130) (IDUM(L),L=1,2)
 70   READ(2,140) CELLNO, DEPTH
        IF(CELLNO .EQ. 0) GO TO 80
        IF(DEPTH .LT. 0.01) GO TO 70
        CELLNO = CELLNO + TN(J)
        BED1(A,CELLNO) = STAGE-DEPTH
        BED2(A,CELLNO) = BED1(A,CELLNO)-0.5
        BED3(A,CELLNO) = BED1(A,CELLNO)-.25
        GO TO 70
 80 CONTINUE
 90 CONTINUE
100 CONTINUE

110 FORMAT(49X,A7)
120 FORMAT(47X,F9.1,11X,F8.2)
130 FORMAT(A10)
140 FORMAT(18X,I4,24X,F11.2)

500 DO 150 I = 1,NIF
      WRITE(8,160)DIS(I),(STG(I,J),J=1,6),(BED2(I,J),J=1,NC)
      WRITE(9,160)DIS(I),(STG(I,J),J=1,6),(BED3(I,J),J=1,NC)
150 CONTINUE
160 FORMAT(F10.2,6F6.2,199F6.2)

    STOP
    END
```

FILE: XEQBED

```
/JOB
BEDELV.
/ACCOUNT
ATTACH,FILESET=PROGSET/NA.
GF,BEDELV.
FTN5,I=BEDELV,L=0.
ATTACH,FILESET=SPNQSET/NA.
GF,SPNQ1.
ATTACH,FILESET=ELEVSET/NA.
GF,S1ADJ2.
ATTACH,FILESET=TP13S1/NA.
GF,TP13PS1.
ATTACH,FILESET=PBEDSET/M=W,NA.
LGO,TP13PS1,SPNQ1,S1ADJ2,BEDPS1.
RF,BEDPS1.
/EOR
     1     1   100 8700   100 1600 7700   100 8700
```

```
      PROGRAM BEDELV(TAPE1,TAPE2,TAPE3=/1246,TAPE4,INPUT,OUTPUT,
     .               TAPE5=INPUT,TAPE6=OUTPUT)

*-----PURPOSE
*
*     SUM SPAWNING WUA FOR CELLS THAT ARE WATERED AT EACH INCUBATION
*     FLOW, THEN TABLE THE RESULTS IN WUA1-LIKE DATA FILE FORMAT FOR
*     A SERIES OF SPAWNING FLOWS READ FROM AN EXTERNAL FILE
*
*     NOTE: THIS PROGRAM HAS BEEN MODIFIED TO READ IN BED ELEVATIONS
*     FOR EACH SPAWNING FLOW; THE BED ELEVATIONS ARE EITHER UNADJUSTED
*     OR ADJUSTED FOR SPECIES SPECIFIC EGG DEPOSITION DEPTHS (0.5 FT FOR
*     CHINOOK, 0.25 FT FOR PINK, CHUM, AND STEELHEAD)
*
*-----LOCAL VARIABLES:

      INTEGER     JOBID(2,10), KSPCDS(5), LSTGDS, SP, CHK, CELL,
     .            NC(6), NTR, SPEC, RCH, WTR(200),WUATBL(185,200),
     .            SPNQ, BEGFLO, SPN1, SPN2, ENDQ, SUBQ1, SUBQ2, QPOS
      REAL        SQ(130), SPNWUA(200,130), PART1(75), PART2(75),
     .            SUMWUA(185,130), ELEV(200,150),STG(150,6)

*     SPEC =      SPECIES, 1=PINK, 2=CHUM, 3=STEELHEAD, 4=CHINOOK
*     RCH =       REACH NUMBER (1-7)
*     NTR =       NO. OF TRANSECTS FOR THIS REACH
*     NC(I) =     NO. OF CELLS IN TRANSECT I
*     BEGFLO =    BEG. FLOW ON TAPE13-TYPE FILE (IN 100 CFS)
*     SPN1,SPN2 = MAX. AND MIN SPAWNING FLOWS THAT WILL BE FOUND
*                 IN THE SPAWNING FLOW FILE (TAPE 2)
*     INC1,INC2 = RANGE OF INCUBATION FLOWS TO BE TABLED
*     SUBQ1,SUBQ2 = RANGE OF FLOWS FOR WHICH BED ELEVATIONS HAVE BEEN
*                 CALCULATED
*     NSPF =      NO. OF SPAWNING FLOWS TO BE READ FROM TP13-TYPE FILE

*-----FILES:

*     TAPE1 =     INPUT, TP13 SPAWNING WUA BY CELL
*     TAPE2 =     INPUT, LIST OF SPAWNING FLOWS TO BE TABLED (BY SPECIES)
*     TAPE4 =     OUTPUT, WUA1-FORMAT DATA TABLE
*     TAPE3 =     INPUT, BED ELEVATIONS AT EACH SPAWNING FLOW
*     TAPE5 =     STANDARD INPUT FILE, RUN PARAMETERS


*-----READ RUN PARAMETERS:

      READ(5,51) RCH,SPEC,INC1,INC2,BEGFLO,SPN1,SPN2,SUBQ1,SUBQ2
   51 FORMAT(9I5)
      CALL PARAMS(RCH,  NTR,NC)

*-----READ IN SPAWNING FLOW DATA FROM TP13-TYPE FILE:

      NSPF = (SPN2 - BEGFLO)/100 + 1
      NSF = 0
      DO 100 SP=1,NSPF
          NCELLS = 0
          NSF = NSF + 1
          CHK = SP - 1
  500     CHK = CHK + 1
```

```
      READ(1,END=501) ((JOBID(I,J),J=1,10),I=1,2)
  501 IF(EOF(1).NE.0 .AND. CHK.LE.SP) GOTO 500

      KR = 2
      READ(1,END=601) IDCURV,KSPCDS,LSTGDS

      DO 200 CELL=1,NC(1)
          NCELLS = NCELLS + 1
          KR = 3
          READ(1,END=601) STAID,SQ(SP),AC,SPNWUA(NCELLS,NSF),CF
  200 CONTINUE
C
C
      DO 201 IT=2,NTR-1
          N=NC(IT)
          KR=KR+1
          DO 202 CELL=1,N
              READ(1,END=601)STAID,SQ(SP),AC,PART1(CELL),CF
  202     CONTINUE
          KR=KR+1
          DO 203 CELL=1,N
              NCELLS=NCELLS+1
              READ(1,END=601)STAID,SQ(SP),AC,PART2(CELL),CF
              SPNWUA(NCELLS,NSF) = PART1(CELL) + PART2(CELL)
  203     CONTINUE
  201 CONTINUE
C
C
      KR=KR+1
      DO 204 CELL=1,NC(NTR)
          NCELLS=NCELLS+1
          READ(1,END=601)STAID,SQ(SP),AC,SPNWUA(NCELLS,NSF),CF
  204 CONTINUE
  100 CONTINUE
C
C
*-----NOW SUM SPAWNING WUA FOR ALL CELLS THAT REMAIN WATERED AT THE
*     INCUBATION FLOW
C
*-----DETERMINE NUMBER OF INCUBATION FLOWS AND ZERO OUT
*     ARRAY TO HOLD WUA SUMS
C
C
      NIF = (INC2-INC1)/100 + 1
      IF(NIF .GT. 185) STOP2
C
      DO 300 SP=1,NSPF
      DO 300 INC=1,NIF
          SUMWUA(INC,SP)=0.
  300 CONTINUE
C
C
*-----READ IN BED ELEVATIONS FOR ALL CELLS (EITHER UNADJUSTED OR
*     ADJUSTED BY EGG DEPTH OF THE SPECIES)
C
      ENDQ = (SUBQ2 - SUBQ1)/100 + 1
        DO 350 J = 1,ENDQ
          READ(3,375)(STG(J,K),K=1,6),(ELEV(I,J),I=1,NCELLS)
  350   CONTINUE
  375 FORMAT(10X,6F6.2,200F6.2)
```

```
*-----FOR ALL SPAWNING FLOWS IN THE RANGE OF INTEREST, LOOP ON INCUBATION
*      FLOWS
C
       IF(SPN1 .LT. SUBQ1) STOP 2
       QPOS = (SPN1 -SUBQ1)/100 + 1
       IQ=QPOS-1
       K = (SPN1 - BEGFLO)/100 + 1
       DO 400 SP=K,NSPF
           IQ  =IQ + 1
           DO 401 INC=1,SP

C*******************************************
C      WRITE(6,376)IQ,Q
C      WRITE(6,377)(ELEV(J,IQ),J=1,NCELLS)
C 376 FORMAT(1X,'IQ = ',I3,3X,'Q = ',F6.0)
C 377 FORMAT(1X,20F6.2)
C*******************************************

               CALL DEWATER(STG,INC,RCH,NCELLS,NTR,NC,ELEV,IQ,    WTR)
               DO 402 CELL=1,NCELLS

C*********************************************
C      WRITE(6,378)CELL,ELEV(CELL,IQ),WTR(CELL)
C 378 FORMAT(1X,'CELL NUMBER = ',I3,' CELL ELEV = ',F6.2,
C     .' WTR = ',I1)
C*********************************************

                   IF(WTR(CELL) .EQ .1) SUMWUA(INC,SP) = SUMWUA(INC,SP)+
                   SPNWUA(CELL,SP)
  402          CONTINUE
  401      CONTINUE
  400 CONTINUE
C
C
*-----READ FILE OF SPAWNING Q'S AND CONSTRUCT WUATBL TO BE WRITTEN TO TAPE4

       NSP = 0
  505 IF(SPEC .EQ. 1) READ(2,21,END=502) SPNQ
       IF(SPEC .EQ. 2) READ(2,22,END=502) SPNQ
       IF(SPEC .EQ. 3) READ(2,23,END=502) SPNQ
       IF(SPEC .EQ. 4) READ(2,24,END=502) SPNQ
   21 FORMAT(I10)
   22 FORMAT(10X,I10)
   23 FORMAT(20X,I10)
   24 FORMAT(30X,I10)
       IF(EOF(4) .NE. 0) GOTO 502
       SP = (SPNQ - BEGFLO)/100 + 1
       IF(SP .LE. 0) THEN
           WRITE(6,61) SPNQ,BEGFLO
   61      FORMAT(1X,'HAVE READ SPAWNING FLOW FROM SPNQ FILE OF: ',I6/
                  1X,'THIS IS LESS THAN BEG FLOW ON TP13 FILE OF: ',I6)
           STOP 3
         ELSEIF(SP .GT. NSPF) THEN
           WRITE(6,62) SPNQ,SQ(NSPF)
   62      FORMAT(1X,'HAVE READ SPAWNING FLOW FROM SPNQ FILE OF: ',I6/
                  1X,'EXCEEDS MAX TP13 FLOW READ: ',I6)
           STOP 4
       ENDIF
       NSP = NSP + 1
```

```
      DO 511 INC=1,NIF
          WUATBL(INC,NSP) = IFIX(SUMWUA(INC,SP))
          IWUA=WUATBL(INC,NSP)
          IF(IWUA.EQ.0.AND.INC.GT.1)THEN
          INCR=INC-1
          WUATBL(INC,NSP)=WUATBL(INCR,NSP)
          ENDIF
  511 CONTINUE
      GOTO 505

  502 IQ = INC1 - 100
      DO 512 INC=1,NIF
          IQ = IQ + 100
          WRITE(4,41) IQ,(WUATBL(INC,SP),SP=1,NSP)
  512 CONTINUE
   41 FORMAT(I5,14I9/22(5X,14I9/))

      STOP
  601 WRITE(6,63) SP,KR,NCELLS
   63 FORMAT(1X,'END-OF-FILE ON TP13 SPAWNING FLOW FILE WITH: ',
     .       ' SP=',I3,',  KR=',I2,',  NCELLS=',I4)
      STOP

      END
      SUBROUTINE PARAMS(RCH,  NTR,NC)

      INTEGER     RCH, NTR, NC(6), NTRAN(7), NCELLS(6,7)

      DATA NTRAN/4,4,6,4,6,5,5/

      DATA (NCELLS(J,1),J=1,6)/40,74,33,36,0,0/,
     .     (NCELLS(J,2),J=1,6)/25,30,31,35, 0, 0/,
     .     (NCELLS(J,3),J=1,6)/28,30,31,36,36,37/,
     .     (NCELLS(J,4),J=1,6)/39,41,34,39, 0, 0/,
     .     (NCELLS(J,5),J=1,6)/33,28,37,30,32,29/,
     .     (NCELLS(J,6),J=1,6)/42,38,37,45,37, 0/,
     .     (NCELLS(J,7),J=1,6)/39,39,32,36,47, 0/

      NTR = NTRAN(RCH)
      DO 100 J=1,6
          NC(J) = NCELLS(J,RCH)
  100 CONTINUE

      RETURN
      END
      SUBROUTINE DEWATER(STG,INC,RCH,NCELLS,NTR,NC,ELEV,IQ, WTR)

      INTEGER     RCH, WTR(200), NTR, NC(6), NCELLS, CELL,IQ
      REAL         ELEV(200,150),STG(150,6)

      DO 100 I=1,NCELLS
          WTR(I) = 0
  100 CONTINUE

      CELL = 0

      DO 101 N=1,NTR
          DO 111 I=1,NC(N)
              CELL = CELL + 1
              IF(ELEV(CELL,IQ) .LE. STG(INC,N)) WTR(CELL) = 1
```

```
111      CONTINUE
101  CONTINUE
     RETURN
     END
```

FILE: BEDELV1

```
      PROGRAM BEDELV1(TAPE1,TAPE2,TAPE3=/1246,TAPE4,INPUT,OUTPUT,
     .                TAPE5=INPUT,TAPE6=OUTPUT)

*-----PURPOSE
*
*     SUM SPAWNING WUA FOR CELLS THAT ARE WATERED AT EACH INCUBATION
*     FLOW, THEN TABLE THE RESULTS IN WUA1-LIKE DATA FILE FORMAT FOR
*     A SERIES OF SPAWNING FLOWS READ FROM AN EXTERNAL FILE
*
*     NOTE: THIS PROGRAM HAS BEEN MODIFIED TO READ IN BED ELEVATIONS
*     FOR EACH SPAWNING FLOW; THE BED ELEVATIONS ARE EITHER UNADJUSTED
*     OR ADJUSTED FOR SPECIES SPECIFIC EGG DEPOSITION DEPTHS (0.5 FT FOR
*     CHINOOK, 0.25 FT FOR PINK, CHUM, AND STEELHEAD)
*
*-----LOCAL VARIABLES:

      INTEGER     JOBID(2,10), KSPCDS(5), LSTGDS, SP, CHK, CELL,
     .            NC(6), NTR, SPEC, RCH, WTR(200),WUATBL(150,200),
     .            SPNQ, BEGFLO, SPN1, SPN2, ENDQ, SUBQ1, SUBQ2, QPOS
      REAL        SQ(130), SPNWUA(200,130), PART1(75), PART2(75),
     .            SUMWUA(150,130), ELEV(200,150),STG(150,6)

*     SPEC =      SPECIES, 1=PINK, 2=CHUM, 3=STEELHEAD, 4=CHINOOK
*     RCH =       REACH NUMBER (1-7)
*     NTR =       NO. OF TRANSECTS FOR THIS REACH
*     NC(I) =     NO. OF CELLS IN TRANSECT I
*     BEGFLO =    BEG. FLOW ON TAPE13-TYPE FILE (IN 100 CFS)
*     SPN1,SPN2 = MAX. AND MIN SPAWNING FLOWS THAT WILL BE FOUND
*                 IN THE SPAWNING FLOW FILE (TAPE 2)
*     INC1,INC2 = RANGE OF INCUBATION FLOWS TO BE TABLED
*     SUBQ1,SUBQ2 = RANGE OF FLOWS FOR WHICH BED ELEVATIONS HAVE BEEN
*                 CALCULATED
*     NSPF =      NO. OF SPAWNING FLOWS TO BE READ FROM TP13-TYPE FILE

*-----FILES:

*     TAPE1 =     INPUT, TP13 SPAWNING WUA BY CELL
*     TAPE2 =     INPUT, LIST OF SPAWNING FLOWS TO BE TABLED (BY SPECIES)
*     TAPE4 =     OUTPUT, WUA1-FORMAT DATA TABLE
*     TAPE3 =     INPUT, BED ELEVATIONS AT EACH SPAWNING FLOW
*     TAPE5 =     STANDARD INPUT FILE, RUN PARAMETERS


*-----READ RUN PARAMETERS:

      READ(5,51) RCH,SPEC,INC1,INC2,BEGFLO,SPN1,SPN2,SUBQ1,SUBQ2
   51 FORMAT(9I5)
      CALL PARAMS(RCH,  NTR,NC)

*-----READ IN SPAWNING FLOW DATA FROM TP13-TYPE FILE:

      NSPF = (SPN2 - BEGFLO)/100 + 1
      NSF = 0
      DO 100 SP=1,NSPF
          NCELLS = 0
          NSF = NSF + 1
          CHK = SP - 1
  500     CHK = CHK + 1
```

```
      READ(1,END=501) ((JOBID(I,J),J=1,10),I=1,2)
  501 IF(EOF(1).NE.0 .AND. CHK.LE.SP) GOTO 500

      KR = 2
      READ(1,END=601) IDCURV,KSPCDS,LSTGDS

      DO 200 CELL=1,NC(1)
          NCELLS = NCELLS + 1
          KR = 3
          READ(1,END=601) STAID,SQ(SP),AC,SPNWUA(NCELLS,NSF),CF
  200 CONTINUE
C
C
      DO 201 IT=2,NTR-1
          N=NC(IT)
          KR=KR+1
          DO 202 CELL=1,N
              READ(1,END=601)STAID,SQ(SP),AC,PART1(CELL),CF
  202     CONTINUE
          KR=KR+1
          DO 203 CELL=1,N
              NCELLS=NCELLS+1
              READ(1,END=601)STAID,SQ(SP),AC,PART2(CELL),CF
              SPNWUA(NCELLS,NSF) = PART1(CELL) + PART2(CELL)
  203     CONTINUE
  201 CONTINUE
C
C
      KR=KR+1
      DO 204 CELL=1,NC(NTR)
          NCELLS=NCELLS+1
          READ(1,END=601)STAID,SQ(SP),AC,SPNWUA(NCELLS,NSF),CF
  204 CONTINUE
  100 CONTINUE
C
C
*-----NOW SUM SPAWNING WUA FOR ALL CELLS THAT REMAIN WATERED AT THE
*     INCUBATION FLOW
C
*-----DETERMINE NUMBER OF INCUBATION FLOWS AND ZERO OUT
*     ARRAY TO HOLD WUA SUMS
C
C
      NIF = (INC2-INC1)/100 + 1
      IF(NIF .GT. 185) STOP2
C
      DO 300 SP=1,NSPF
      DO 300 INC=1,NIF
          SUMWUA(INC,SP)=0.
  300 CONTINUE
C
C
*-----READ IN BED ELEVATIONS FOR ALL CELLS (EITHER UNADJUSTED OR
*     ADJUSTED BY EQQ DEPTH OF THE SPECIES)
C
      ENDQ = (SUBQ2 - SUBQ1)/100 + 1
        DO 350 J = 1,ENDQ
          READ(3,375)(STQ(J,K),K=1,6),(ELEV(I,J),I=1,NCELLS)
  350   CONTINUE
  375 FORMAT(10X,6F6.2,200F6.2)
```

```
*-----FOR ALL SPAWNING FLOWS IN THE RANGE OF INTEREST, LOOP ON INCUBATION
*      FLOWS
C
       IF(SPN1 .LT. SUBQ1) STOP 2
       QPOS = (SPN1 -SUBQ1)/100 + 1
       IQ=QPOS-1
       K = (SPN1 - BEGFLO)/100 + 1
       DO 400 SP=K,NSPF
           IQ  =IQ + 1
           DO 401 INC=1,SP

C*****************************************
C      WRITE(6,376)IQ,Q
C      WRITE(6,377)(ELEV(J,IQ),J=1,NCELLS)
C 376 FORMAT(1X,'IQ = ',I3,3X,'Q = ',F6.0)
C 377 FORMAT(1X,20F6.2)
C*****************************************

              CALL DEWATER(STG,INC,RCH,NCELLS,NTR,NC,ELEV,IQ,    WTR)
              DO 402 CELL=1,NCELLS

C*****************************************
C      WRITE(6,378)CELL,ELEV(CELL,IQ),WTR(CELL)
C 378 FORMAT(1X,'CELL NUMBER = ',I3,' CELL ELEV = ',F6.2,
C     .' WTR = ',I1)
C*****************************************

                  IF(WTR(CELL) .EQ .1) SUMWUA(INC,SP) = SUMWUA(INC,SP)+
     .            SPNWUA(CELL,SP)
  402         CONTINUE
  401     CONTINUE
  400 CONTINUE
C
C
*-----READ FILE OF SPAWNING Q'S AND CONSTRUCT WUATBL TO BE WRITTEN TO TAPE4

       NSP = 0
  505 IF(SPEC .EQ. 1) READ(2,21,END=502) SPNQ
       IF(SPEC .EQ. 2) READ(2,22,END=502) SPNQ
       IF(SPEC .EQ. 3) READ(2,23,END=502) SPNQ
       IF(SPEC .EQ. 4) READ(2,24,END=502) SPNQ
    21 FORMAT(I10)
    22 FORMAT(10X,I10)
    23 FORMAT(20X,I10)
    24 FORMAT(30X,I10)
       IF(EOF(4) .NE. 0) GOTO 502
       SP = (SPNQ - BEGFLO)/100 + 1
       IF(SP .LE. 0) THEN
           WRITE(6,61) SPNQ,BEGFLO
    61     FORMAT(1X,'HAVE READ SPAWNING FLOW FROM SPNQ FILE OF: ',I6/
     .           1X,'THIS IS LESS THAN BEG FLOW ON TP13 FILE OF: ',I6)
           STOP 3
         ELSEIF(SP .GT. NSPF) THEN
           WRITE(6,62) SPNQ,SQ(NSPF)
    62     FORMAT(1X,'HAVE READ SPAWNING FLOW FROM SPNQ FILE OF: ',I6/
     .          1X,'EXCEEDS MAX TP13 FLOW READ: ',I6)
           STOP 4
       ENDIF
       NSP = NSP + 1
```

```
      DO 511 INC=1,NIF
          WUATBL(INC,NSP) = IFIX(SUMWUA(INC,SP))
          IWUA=WUATBL(INC,NSP)
          IF(IWUA.EQ.0.AND.INC.GT.1)THEN
          INCR=INC-1
          WUATBL(INC,NSP)=WUATBL(INCR,NSP)
          ENDIF
  511 CONTINUE
      GOTO 505

  502 IQ = INC1 - 100
      DO 512 INC=1,NIF
          IQ = IQ + 100
          WRITE(4,41) IQ,(WUATBL(INC,SP),SP=1,NSP)
  512 CONTINUE
   41 FORMAT(I5,14I9/22(5X,14I9/))

      STOP
  601 WRITE(6,63) SP,KR,NCELLS
   63 FORMAT(1X,'END-OF-FILE ON TP13 SPAWNING FLOW FILE WITH:',
     .        ' SP=',I3,',  KR=',I2,',  NCELLS=',I4)
      STOP

      END
      SUBROUTINE PARAMS(RCH,  NTR,NC)

      INTEGER     RCH, NTR, NC(6), NTRAN(7), NCELLS(6,7)

      DATA NTRAN/4,4,6,4,6,5,5/

      DATA (NCELLS(J,1),J=1,6)/40,74,33,36,0,0/,
     .     (NCELLS(J,2),J=1,6)/25,30,31,35, 0, 0/,
     .     (NCELLS(J,3),J=1,6)/28,30,31,36,36,37/,
     .     (NCELLS(J,4),J=1,6)/39,41,34,39, 0, 0/,
     .     (NCELLS(J,5),J=1,6)/33,28,37,30,32,29/,
     .     (NCELLS(J,6),J=1,6)/42,38,37,45,37, 0/,
     .     (NCELLS(J,7),J=1,6)/39,39,32,36,47, 0/

      NTR = NTRAN(RCH)
      DO 100 J=1,6
          NC(J) = NCELLS(J,RCH)
  100 CONTINUE

      RETURN
      END
      SUBROUTINE DEWATER(STG,INC,RCH,NCELLS,NTR,NC,ELEV,IQ, WTR)

      INTEGER     RCH, WTR(200), NTR, NC(6), NCELLS, CELL,IQ
      REAL         ELEV(200,150),STG(150,6)

      DO 100 I=1,NCELLS
          WTR(I) = 0
  100 CONTINUE

      CELL = 0

      DO 101 N=1,NTR
          DO 111 I=1,NC(N)
              CELL = CELL + 1
              IF(ELEV(CELL,IQ) .LE. STG(INC,N)) WTR(CELL) = 1
```

```
111      CONTINUE
101  CONTINUE
     RETURN
     END
```

FILE: XEQEFSP

```
/JOB
EFSPWN,PO.
/ACCOUNT
ATTACH,FILESET=PROGSET/NA.
GF,EFSPWN.
FTN5,I=EFSPWN.
ATTACH,FILESET=SPNQSET/NA.
GF,SPNQ5.
ATTACH,FILESET=TP13S5/NA.
GF,TP13PS5.
ATTACH,TP13I5/NA.
LGO,TP13PS5,TP13I5,SPNQ5,W13PS5.
ATTACH,FILESET=P13SET/M=W,NA.
RF,W13PS5.
/EOR
    5     1  30011900 1900 9000
```

FILE: EFSPWN

```
      PROGRAM EFSPWN(TAPE1,TAPE2,TAPE3,TAPE4,INPUT,OUTPUT,
     +TAPE5=INPUT,TAPE6=OUTPUT)
C
      DIMENSION SPNWUA(200,150),INCWUA(200,150),SUMWUA(150,171),
     +TEMPS(75),TEMPS1(75),TEMPI(75),TEMPI1(75),WUASUM(150,171),
     +JOBID(2,10),KSPCDS(5)
C
      REAL STAID,SQ,IQ,AC,SPNWUA,CF,INCWUA,TEMPS,TEMPS1,
     +TEMPI,TEMPI1,SUMWUA,WUASUM
C
      INTEGER JOBID,KSPCDS,LSTCDS,SP,INC,CELL,NCELLS,MINSQ,MAXSQ,
     +MINIQ,MAXIQ,NSPF,CHK,FLOW,RCH,SPEC,NC(6),NTR
C
C
*-----READ RUN PARAMETERS@D
      READ(5,51) RCH,SPEC,MINIQ,MAXIQ,MINSQ,MAXSQ
   51 FORMAT(6I5)
      CALL PARAMS(RCH,  NTR,NC)
      MINIQ = MINIQ/100
      MAXIQ = MAXIQ/100
      MINSQ = MINSQ/100
      MAXSQ = MAXSQ/100

      NSPF=171
C
C
      DO 65 SP=1,MAXSQ
        IF(SP .LT. MINIQ) GO TO 65
        NCELLS = 0
      KS=SP-1
    5 KS=KS+1
      READ(1,END=10)((JOBID(I,J),J=1,10),I=1,2)
   10   IF(EOF(1).NE.0 .AND. KS.LE.SP) GOTO 5
      KS=SP-1
   15 KS=KS+1
      READ(2,END=20)((JOBID(I,J),J=1,10),I=1,2)
   20   IF(EOF(2).NE.0 .AND. KS.LE.SP) GO TO 15
      CHK = 1
        READ(1,END=130)IDCURV,KSPCDS,LSTCDS
        READ(2,END=135)IDCURV,KSPCDS,LSTCDS
C
C
      DO 25 CELL=1,NC(1)
        NCELLS=NCELLS+1
      CHK = 2
        READ(1,END=130)STAID,SQ,AC,SPNWUA(NCELLS,SP),CF
        READ(2,END=135)STAID,IQ,AC,INCWUA(NCELLS,SP),CF
   25 CONTINUE
C
C
      DO 300 IT=2,NTR-1
      N = NC(IT)
          CHK = CHK + 1
          DO 30 CELL=1,N
               READ(1,END=130)STAID,SQ,AC,TEMPS(CELL),CF
               READ(2,END=135)STAID,IQ,AC,TEMPI(CELL),CF
   30     CONTINUE
```

```
C
C
          CHK = CHK + 1
          DO 35 CELL=1,N
             NCELLS=NCELLS+1
             READ(1,END=130)STAID,SQ,AC,TEMPS1(CELL),CF
             READ(2,END=135)STAID,IQ,AC,TEMPI1(CELL),CF
             SPNWUA(NCELLS,SP)=TEMPS(CELL)+TEMPS1(CELL)
             INCWUA(NCELLS,SP)=TEMPI(CELL)+TEMPI1(CELL)
   35     CONTINUE
  300 CONTINUE
C
C
       DO 60 CELL=1,NC(NTR)
         NCELLS=NCELLS+1
         CHK = CHK + 1
         READ(1,END=130)STAID,SQ,AC,SPNWUA(NCELLS,SP),CF
         READ(2,END=135)STAID,IQ,AC,INCWUA(NCELLS,SP),CF
   60 CONTINUE
C
C
   65 CONTINUE
C
C
       DO 75 SP=1,NSPF
         DO 70 INC=1,MAXIQ
         SUMWUA(INC,SP)=0.
         WUASUM(INC,SP)=0.
   70 CONTINUE
   75 CONTINUE
C
C
C
       DO 90 SP=MINSQ,MAXSQ
       DO 85 INC=MINIQ,SP
         DO 80 CELL=1,NCELLS
          IF(SPNWUA(CELL,SP).LE.INCWUA(CELL,INC))THEN
           SUMWUA(INC,SP)=SUMWUA(INC,SP)+SPNWUA(CELL,SP)
          ELSE
         SUMWUA(INC,SP)=SUMWUA(INC,SP)+INCWUA(CELL,INC)
       ENDIF
   80 CONTINUE
   85 CONTINUE
   90 CONTINUE
C
C
       DO 115 SP=1,NSPF
         IF(SPEC .EQ. 1) READ(3,301)FLOW
         IF(SPEC .EQ. 2) READ(3,302) FLOW
         IF(SPEC .EQ. 3) READ(3,303) FLOW
         IF(SPEC .EQ. 4) READ(3,304) FLOW
         FLOW=FLOW/100
         DO 110 INC=MINIQ,MAXIQ
         WUASUM(INC,SP)=SUMWUA(INC,FLOW)
         IWUA=WUASUM(INC,SP)
         IF(IWUA .EQ. 0 .AND. INC .GT. 1)THEN
            INCR=INC-1
            WUASUM(INC,SP)=WUASUM(INCR,SP)
         ENDIF
```

```
  301 FORMAT(I10)
  302 FORMAT(10X,I10)
  303 FORMAT(20X,I10)
  304 FORMAT(30X,I10)
  110 CONTINUE
  115 CONTINUE
C
C
      DO 125 INC=MINIQ,MAXIQ
        IIQ=INC*100
        WRITE(4,120)IIQ,(WUASUM(INC,SP),SP=1,NSPF)
  120 FORMAT(I5,14F9.0/22(5X,14F9.0/))
  125 CONTINUE
C
      STOP

  130 WRITE(6,140) SP,CHK,NCELLS
      STOP
  135 WRITE(6,145) SP,CHK,NCELLS
      STOP
  140 FORMAT(1X,'END-OF-FILE ON SPAWNING FLOW FILE WITH@D',
     .       ' SP=',I3,',  CHK=',I2,',  NCELLS=',I4)
  145 FORMAT(1X,'END-OF-FILE ON INCUBATION FLOW FILE WITH@D',
     .       ' SP=',I3,',  CHK=',I2,',  NCELLS=',I4)
      END

      SUBROUTINE PARAMS(RCH,NTR,NC)
      INTEGER RCH,NTR,NC(6),NTRAN(7),  NCELLS(6,7)
      DATA NTRAN/4,4,6,4,6,5,5/
      DATA (NCELLS(J,1),J=1,6)/40,74,33,36,0,0/,
     .     (NCELLS(J,2),J=1,6)/25,30,31,35,0,0/,
     .     (NCELLS(J,3),J=1,6)/28,30,31,36,36,37/,
     .     (NCELLS(J,4),J=1,6)/39,41,34,39,0,0/,
     .     (NCELLS(J,5),J=1,6)/33,28,37,30,32,29/,
     .     (NCELLS(J,6),J=1,6)/42,38,37,45,37,0/,
     .     (NCELLS(J,7),J=1,6)/39,39,32,36,47,0/

      NTR=NTRAN(RCH)
      DO 100 J=1,6
         NC(J)=NCELLS(J,RCH)
  100 CONTINUE

      RETURN
      END
```

FILE: MAXPROG

```
/JOB
A,P3.
/ACCOUNT
ATTACH,FILESET=K13SET/M=W,NA.
GF,W13KS1.
FTN5,DB.
LGO,W13KS1,WP3KS1.
AF,WP3KS1.
/EOR
      PROGRAM MAXPROG(TAPE1,TAPE3,OUTPUT)
C***THIS PROGRAM IS IN TWO PARTS.  THE FIRST PART FINDS THE FMAXIMUM WUA
C    VALUES IN THE EFFECTIVE SPAWNING REACH FILES AND REPLACES ERRONEOUS
C    ZEROES WITH THE FMAXIMUM VALUE.   THE SECOND PART OF THE PROGRAM
C    COMPUTES THE PERCENTAGE OF EACH WUA VALUE WITH RESPECT TO THE
C    FMAXIMUM WUA VALUE.
C***FILE USE:
C    FILE1 = REACH DATA FILES
C    FILE2 = CORRECTED REACH DATA
C    FILE3 = PERCENTAGE OF FMAX FILE
C***PROGRAM WRITTEN BY W. MEYER, SPRING 1984 FOR THE FTN5 COMPILER.
C
      DIMENSION WUA(171), FMAX(171), PRCNT(171)
      DATA FMAX/171*0/
C***PART 1; DETERMINE FMAXIMUM WUA VALUES AND WRITE CORRECTED FILE
   10 READ (1,100,END=99) IFLOW,(WUA(I),I=1,171)

      DO 20 I=1,171
        IF (WUA(I) .GT. FMAX(I)) FMAX(I) = WUA(I)
   20 CONTINUE

      GO TO 10
  100 FORMAT (I5,14F9.0,/,11(5X,14F9.0,/),5X,3F9.0)
   99 REWIND (1)
C***PART 2; CALCULATE AND PRINT WUA PERCENTAGES.
   30 READ (1,100,END=999) IFLOW,(WUA(I),I=1,171)

      DO 40 I=1,171
   40   PRCNT(I) = (WUA(I)/FMAX(I)) * 100

      WRITE (3,300) IFLOW,(PRCNT(I),I=1,171)
  300 FORMAT(I5,14F9.5,/,11(5X,14F9.5,/),5X,3F9.5)
      GO TO 30
  999 ENDFILE (3)
      STOP
      END
```

FILE: CCOND

1	90	200	200	300	300	500	600	400	200	200
1	50	400	400	300	400	800	1000	700	400	300
1	10	700	700	500	600	1100	1300	1100	600	500
2	90	200	300	400	400	700	800	600	300	300
2	50	600	500	400	500	1000	1300	1000	500	400
2	10	900	900	700	800	1500	1700	1500	800	600
3	90	300	300	500	600	900	1100	800	400	400
3	50	700	600	600	700	1300	1700	1200	700	600
3	10	1100	1100	900	1000	1900	2200	2000	1100	800
4	90	500	600	700	800	1400	1600	1100	600	600
4	50	1200	1100	1000	1100	2000	2600	1900	1000	900
4	10	1800	1800	1500	1600	2900	3600	3100	1700	1300
5	90	600	700	900	1100	1600	1900	1300	700	700
5	50	1400	1300	1200	1400	2400	3100	2300	1200	1100
5	10	2200	2200	1800	2000	3500	4300	3800	2000	1600
6	90	1000	1000	1300	1700	2900	3600	2400	1300	1200
6	50	2200	2000	1800	2300	4200	5300	4000	2100	1700
6	10	3500	3400	2800	3300	6000	7400	6300	3400	2600
7	90	1100	1100	1500	1900	3200	4000	2800	1500	1300
7	50	2500	2200	2100	2500	4600	5900	4500	2400	1900
7	10	4200	4000	3300	3600	6600	8200	7100	3800	2800

FILE: CLABF

171
1400 .90
1400 .50
1400 .10
1500 .90
1500 .50
1500 .10
1600 .90
1600 .50
1600 .10
1700 .90
1700 .50
1700 .10
1800 .90
1800 .50
1800 .10
1900 .90
1900 .50
1900 .10
2000 .90
2000 .50
2000 .10
2100 .90
2100 .50
2100 .10
2200 .90
2200 .50
2200 .10
2300 .90
2300 .50
2300 .10
2400 .90
2400 .50
2400 .10
2500 .90
2500 .50
2500 .10
2600 .90
2600 .50
2600 .10
2700 .90
2700 .50
2700 .10
2800 .90
2800 .50
2800 .10
2900 .90
2900 .50
2900 .10
3000 .90
3000 .50
3000 .10
3100 .90
3100 .50
3100 .10
3200 .90
3200 .50
3200 .10

3300	.90
3300	.50
3300	.10
3400	.90
3400	.50
3400	.10
3500	.90
3500	.50
3500	.10
3600	.90
3600	.50
3600	.10
3700	.90
3700	.50
3700	.10
3800	.90
3800	.50
3800	.10
3900	.90
3900	.50
3900	.10
4000	.90
4000	.50
4000	.10
4100	.90
4100	.50
4100	.10
4200	.90
4200	.50
4200	.10
4300	.90
4300	.50
4300	.10
4400	.90
4400	.50
4400	.10
4500	.90
4500	.50
4500	.10
4600	.90
4600	.50
4600	.10
4700	.90
4700	.50
4700	.10
4800	.90
4800	.50
4800	.10
4900	.90
4900	.50
4900	.10
5000	.90
5000	.50
5000	.10
5100	.90
5100	.50
5100	.10
5200	.90
5200	.50
5200	.10

5300 .90
5300 .50
5300 .10
5400 .90
5400 .50
5400 .10
5500 .90
5500 .50
5500 .10
5600 .90
5600 .50
5600 .10
5700 .90
5700 .50
5700 .10
5800 .90
5800 .50
5800 .10
5900 .90
5900 .50
5900 .10
6000 .90
6000 .50
6000 .10
6100 .90
6100 .50
6100 .10
6200 .90
6200 .50
6200 .10
6300 .90
6300 .50
6300 .10
6400 .90
6400 .50
6400 .10
6500 .90
6500 .50
6500 .10
6600 .90
6600 .50
6600 .10
6700 .90
6700 .50
6700 .10
6800 .90
6800 .50
6800 .10
6900 .90
6900 .50
6900 .10
7000 .90
7000 .50
7000 .10
15
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)

JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JANUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
FEBRUARY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MARCH - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)

CLABF p.4

FILE: SCOND

1	90	600	400	200	200
1	50	1000	700	400	300
1	10	1300	1100	600	500
2	90	800	600	300	300
2	50	1300	1000	500	400
2	10	1700	1500	800	600
3	90	1100	800	400	400
3	50	1700	1200	700	600
3	10	2200	2000	1100	800
4	90	1600	1100	600	600
4	50	2600	1900	1000	900
4	10	3600	3100	1700	1300
5	90	1900	1300	700	700
5	50	3100	2300	1200	1100
5	10	4300	3800	2000	1600
6	90	3600	2400	1300	1200
6	50	5300	4000	2100	1700
6	10	7400	6300	3400	2600
7	90	4000	2800	1500	1300
7	50	5900	4500	2400	1900
7	10	8200	7100	3800	2800

FILE: SLABF

```
171
1400    .90
1400    .50
1400    .10
1500    .90
1500    .50
1500    .10
1600    .90
1600    .50
1600    .10
1700    .90
1700    .50
1700    .10
1800    .90
1800    .50
1800    .10
1900    .90
1900    .50
1900    .10
2000    .90
2000    .50
2000    .10
2100    .90
2100    .50
2100    .10
2200    .90
2200    .50
2200    .10
2300    .90
2300    .50
2300    .10
2400    .90
2400    .50
2400    .10
2500    .90
2500    .50
2500    .10
2600    .90
2600    .50
2600    .10
2700    .90
2700    .50
2700    .10
2800    .90
2800    .50
2800    .10
2900    .90
2900    .50
2900    .10
3000    .90
3000    .50
3000    .10
3100    .90
3100    .50
3100    .10
3200    .90
3200    .50
3200    .10
```

3300	.90
3300	.50
3300	.10
3400	.90
3400	.50
3400	.10
3500	.90
3500	.50
3500	.10
3600	.90
3600	.50
3600	.10
3700	.90
3700	.50
3700	.10
3800	.90
3800	.50
3800	.10
3900	.90
3900	.50
3900	.10
4000	.90
4000	.50
4000	.10
4100	.90
4100	.50
4100	.10
4200	.90
4200	.50
4200	.10
4300	.90
4300	.50
4300	.10
4400	.90
4400	.50
4400	.10
4500	.90
4500	.50
4500	.10
4600	.90
4600	.50
4600	.10
4700	.90
4700	.50
4700	.10
4800	.90
4800	.50
4800	.10
4900	.90
4900	.50
4900	.10
5000	.90
5000	.50
5000	.10
5100	.90
5100	.50
5100	.10
5200	.90
5200	.50
5200	.10

```
5300    .90
5300    .50
5300    .10
5400    .90
5400    .50
5400    .10
5500    .90
5500    .50
5500    .10
5600    .90
5600    .50
5600    .10
5700    .90
5700    .50
5700    .10
5800    .90
5800    .50
5800    .10
5900    .90
5900    .50
5900    .10
6000    .90
6000    .50
6000    .10
6100    .90
6100    .50
6100    .10
6200    .90
6200    .50
6200    .10
6300    .90
6300    .50
6300    .10
6400    .90
6400    .50
6400    .10
6500    .90
6500    .50
6500    .10
6600    .90
6600    .50
6600    .10
6700    .90
6700    .50
6700    .10
6800    .90
6800    .50
6800    .10
6900    .90
6900    .50
6900    .10
7000    .90
7000    .50
7000    .10
  9
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
```

AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
AUGUST - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)

FILE: SCONDA

1	90	300	500	600	400	200	200
1	50	400	800	1000	700	400	300
1	10	600	1100	1300	1100	600	500
2	90	400	700	800	600	300	300
2	50	500	1000	1300	1000	500	400
2	10	800	1500	1700	1500	800	600
3	90	600	900	1100	800	400	400
3	50	700	1300	1700	1200	700	600
3	10	1000	1900	2200	2000	1100	800
4	90	800	1400	1600	1100	600	600
4	50	1100	2000	2600	1900	1000	900
4	10	1600	2900	3600	3100	1700	1300
5	90	1100	1600	1900	1300	700	700
5	50	1400	2400	3100	2300	1200	1100
5	10	2000	3500	4300	3800	2000	1600
6	90	1700	2900	3600	2400	1300	1200
6	50	2300	4200	5300	4000	2100	1700
6	10	3300	6000	7400	6300	3400	2600
7	90	1900	3200	4000	2800	1500	1300
7	50	2500	4600	5900	4500	2400	1900
7	10	3600	6600	8200	7100	3800	2800

FILE: SLABFA

```
171
1400   .90
1400   .50
1400   .10
1500   .90
1500   .50
1500   .10
1600   .90
1600   .50
1600   .10
1700   .90
1700   .50
1700   .10
1800   .90
1800   .50
1800   .10
1900   .90
1900   .50
1900   .10
2000   .90
2000   .50
2000   .10
2100   .90
2100   .50
2100   .10
2200   .90
2200   .50
2200   .10
2300   .90
2300   .50
2300   .10
2400   .90
2400   .50
2400   .10
2500   .90
2500   .50
2500   .10
2600   .90
2600   .50
2600   .10
2700   .90
2700   .50
2700   .10
2800   .90
2800   .50
2800   .10
2900   .90
2900   .50
2900   .10
3000   .90
3000   .50
3000   .10
3100   .90
3100   .50
3100   .10
3200   .90
3200   .50
3200   .10
```

3300	.90
3300	.50
3300	.10
3400	.90
3400	.50
3400	.10
3500	.90
3500	.50
3500	.10
3600	.90
3600	.50
3600	.10
3700	.90
3700	.50
3700	.10
3800	.90
3800	.50
3800	.10
3900	.90
3900	.50
3900	.10
4000	.90
4000	.50
4000	.10
4100	.90
4100	.50
4100	.10
4200	.90
4200	.50
4200	.10
4300	.90
4300	.50
4300	.10
4400	.90
4400	.50
4400	.10
4500	.90
4500	.50
4500	.10
4600	.90
4600	.50
4600	.10
4700	.90
4700	.50
4700	.10
4800	.90
4800	.50
4800	.10
4900	.90
4900	.50
4900	.10
5000	.90
5000	.50
5000	.10
5100	.90
5100	.50
5100	.10
5200	.90
5200	.50
5200	.10

5300 .90
5300 .50
5300 .10
5400 .90
5400 .50
5400 .10
5500 .90
5500 .50
5500 .10
5600 .90
5600 .50
5600 .10
5700 .90
5700 .50
5700 .10
5800 .90
5800 .50
5800 .10
5900 .90
5900 .50
5900 .10
6000 .90
6000 .50
6000 .10
6100 .90
6100 .50
6100 .10
6200 .90
6200 .50
6200 .10
6300 .90
6300 .50
6300 .10
6400 .90
6400 .50
6400 .10
6500 .90
6500 .50
6500 .10
6600 .90
6600 .50
6600 .10
6700 .90
6700 .50
6700 .10
6800 .90
6800 .50
6800 .10
6900 .90
6900 .50
6900 .10
7000 .90
7000 .50
7000 .10
12
APRIL- TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)

MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
APRIL - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)

FILE: SCONDB

1	90	500	600	400	200	200
1	50	800	1000	700	400	300
1	10	1100	1300	1100	600	500
2	90	700	800	600	300	300
2	50	1000	1300	1000	500	400
2	10	1500	1700	1500	800	600
3	90	900	1100	800	400	400
3	50	1300	1700	1200	700	600
3	10	1900	2200	2000	1100	800
4	90	1400	1600	1100	600	600
4	50	2000	2600	1900	1000	900
4	10	2900	3600	3100	1700	1300
5	90	1600	1900	1300	700	700
5	50	2400	3100	2300	1200	1100
5	10	3500	4300	3800	2000	1600
6	90	2900	3600	2400	1300	1200
6	50	4200	5300	4000	2100	1700
6	10	6000	7400	6300	3400	2600
7	90	3200	4000	2800	1500	1300
7	50	4600	5900	4500	2400	1900
7	10	6600	8200	7100	3800	2800

FILE: SLABFB


```
171
1400    .90
1400    .50
1400    .10
1500    .90
1500    .50
1500    .10
1600    .90
1600    .50
1600    .10
1700    .90
1700    .50
1700    .10
1800    .90
1800    .50
1800    .10
1900    .90
1900    .50
1900    .10
2000    .90
2000    .50
2000    .10
2100    .90
2100    .50
2100    .10
2200    .90
2200    .50
2200    .10
2300    .90
2300    .50
2300    .10
2400    .90
2400    .50
2400    .10
2500    .90
2500    .50
2500    .10
2600    .90
2600    .50
2600    .10
2700    .90
2700    .50
2700    .10
2800    .90
2800    .50
2800    .10
2900    .90
2900    .50
2900    .10
3000    .90
3000    .50
3000    .10
3100    .90
3100    .50
3100    .10
3200    .90
3200    .50
3200    .10
```

3300	.90
3300	.50
3300	.10
3400	.90
3400	.50
3400	.10
3500	.90
3500	.50
3500	.10
3600	.90
3600	.50
3600	.10
3700	.90
3700	.50
3700	.10
3800	.90
3800	.50
3800	.10
3900	.90
3900	.50
3900	.10
4000	.90
4000	.50
4000	.10
4100	.90
4100	.50
4100	.10
4200	.90
4200	.50
4200	.10
4300	.90
4300	.50
4300	.10
4400	.90
4400	.50
4400	.10
4500	.90
4500	.50
4500	.10
4600	.90
4600	.50
4600	.10
4700	.90
4700	.50
4700	.10
4800	.90
4800	.50
4800	.10
4900	.90
4900	.50
4900	.10
5000	.90
5000	.50
5000	.10
5100	.90
5100	.50
5100	.10
5200	.90
5200	.50
5200	.10

```
5300    .90
5300    .50
5300    .10
5400    .90
5400    .50
5400    .10
5500    .90
5500    .50
5500    .10
5600    .90
5600    .50
5600    .10
5700    .90
5700    .50
5700    .10
5800    .90
5800    .50
5800    .10
5900    .90
5900    .50
5900    .10
6000    .90
6000    .50
6000    .10
6100    .90
6100    .50
6100    .10
6200    .90
6200    .50
6200    .10
6300    .90
6300    .50
6300    .10
6400    .90
6400    .50
6400    .10
6500    .90
6500    .50
6500    .10
6600    .90
6600    .50
6600    .10
6700    .90
6700    .50
6700    .10
6800    .90
6800    .50
6800    .10
6900    .90
6900    .50
6900    .10
7000    .90
7000    .50
7000    .10
  9
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .90 (DRY)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
```

JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .50 (MEDIAN)
MAY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JUNE - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)
JULY - TRIBUTARY INFLOW EXCEEDANCE PROBABILITY = .10 (WET)

FILE: WUA2EF2

```
      PROGRAM WUA2EF2(TAPE1,TAPE2,TAPE3,TAPE4,TAPE7,
     .                INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)

C*********************************************************************
C*    READS (1) TABLED WUA AND GROSS AREA BY DISCHARGE, FOR EACH     *
C*              REACH,                                               *
C*          (2) NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH        *
C*          (3) A LABEL FILE                                         *
C*    AND PRODUCES A TABLE  OF OUTPUT (SUMMED ACROSS REACHES):       *
C*          (1) WUA FOR EACH DISCHARGE AS PERCENT OF THE GREATEST    *
C*              WUA, FOR EACH LIFE STAGE (EFFECTIVE SPAWNING WUA)    *
C*********************************************************************

C-----FILES:
C
C     TAPE1 = INPUT, WUA TABLES PRODUCED BY PROGRAM WUA1, END-OF-
C             FILE BETWEEN TABLES FOR EACH REACH
C     TAPE2 = INPUT, LABEL FILE
C     TAPE3 = INPUT, NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH
C     TAPE4 = OUTPUT, TABLED WUA AS PERCENT OF GREATEST WUA
C     TAPE5 = STANDARD INPUT FILE
C     TAPE6 = STANDARD OUTPUT FILE (ERROR MESSAGES, DIAGNOSTICS)
C     TAPE7 = OUTPUT, FLOWS WHERE CRITICAL PERCENTS ARE REACHED

C-----LOCAL VARIABLES:

      INTEGER      MINQ, MAXQ, DIVQ, NR, NSTG, NC, MAXR, MAXSTG, MAXD,
     .             REACH, ND, Q, R, D, C, NSUB, D1, D2, NDIS(7), ADJFLO,
     .             FLOW(7,150), PASS, G, POS, CRIT(12), DUM2(2),
     .             COND(7,36), SPLABL(2,87), CLABL(7,36), QLABL(15),
     .             BEGC, ENDC, RUNLABL(6), DUM, SPNAME, DMAX(87)

      REAL         SUMW(150,87), WMAX(87), PCTS(12), WUA(7,150,87),
     .             FACTOR(7)

C     DEFINITIONS
C     PASS = 1   READS FIRST 6 LINES FOR EACH FLOW AND PROCESSED
C                FIRST 84 SPAWNING Q'S
C     PASS = 2   READS LINES 7-13 FOR EACH FLOWS AND PROCESSSS LAST
C                87 SPAWNING Q'S
C
C     MINQ,MAXQ = MINIMUM AND MAXIMUM DISCHARGE VALUES TO BE TABLED
C     DIVQ =      SIZE OF INTERVAL BETWEEN TABLED DISCHARGE VALUES
C     NR =        NUMBER OF REACHES BEING PROCESSED
C     NSTG =      NUMBER OF SPECIES-LIFE STAGE CODES
C     NC =        NUMBER OF TRIBUTARY INFLOW CONDITIONS TO BE TABLED
C     MAXR =      MAXIMUM NUMBER OF REACHES THAT CAN BE PROCESSED
C                 WITHOUT RE-DIMENSIONING ARRAYS
C     MAXSTG =    MAXIMUM NUMBER OF SPECIES-LIFE STAGE CODES
C     MAXD =      MAXIMUM NUMBER OF DISCHARGE LEVELS (BEFORE ADJUSTING FOR
C                 TRIBUTARY INFLOW CONDITIONS AND RESTRICTING TO MINQ-MAXQ)
C     NSUB =      NUMBER OF SUB-TABLES, WITH 15 DISCHARGE LEVELS PER
C                 SUB-TABLE
C     WUA(R,D,L) = WEIGHTED USABLE AREA FOR REACH R, DISCHARGE D,
C                 LIFE STAGE CODE L
C     GROSS(R,D) = GROSS AREA FOR REACH R, DISCHARGE D
C     FLOW(R,D) = DISCHARGE VALUE FOR REACH R, DISCHARGE D
C     COND(R,C) = TRIBUTARY INFLOW CONDITION FOR REACH R, CONDITION C
```

```
*     SPLABL(I,L) = SPECIES-LIFE STAGE LABEL FOR LIFE STAGE L
*     CLABL(I,C) = TABLE HEADING FOR CONDITION C
*     QLABL(I) =  USED IN SUB-TABLE HEADING (15 FLOW VALUES)
*     SUMW(D,L) = WUA SUMMED ACROSS REACHES FOR DISCHARGE LEVEL D,
*                 LIFE STAGE L (THEN BECOMES TABLE 1 VALUES AFTER DIVIDING
*                 BY MAX WUA)
*     WMAX(L) =   MAXIMUM WUA ACROSS ALL DISCHARGE LEVELS FOR A GIVEN
*                 LIFE STAGE L
*     FACTOR(R) = MULTIPLICATIVE FACTOR FOR REACH R, EXPANDS WUA
*                 AND GROSS AREA VALUES TO REPRESENT A SECTION
*                 OF THE RIVER
*     BEGC,ENDC = BEGINNING AND ENDING INDEX WITHIN CONDITION
*                 ARRAYS FOR A SUBSET OF THE NC CONDITIONS TO BE USED
*                 IN ANY GIVEN RUN
*     NP =        NUMBER OF EXCEEDANCE PROBABILITES
*     NM =        NUMBER OF MONTHS AT EACH EXCEEDANCE PROBABILITY
*
*-----CONSTANTS:
      DATA MAXR,MAXSTQ,MAXD/7,87,150/
      DATA PCTS/100., 99.99, 99.95, 99.9, 99.75, 99.5, 99.25, 99.0,
     .          97.5, 95.0, 92.5, 90./
      NSP1 = 84
      NSP2 = 87
      NPCT = 12
*-----READ PARAMETERS, SPECIES LABELS, CONDITION LABELS
      READ(5,50) SPNAME,RUNLABL

      WRITE(6,50) SPNAME,RUNLABL

      READ(5,51) MINQ,MAXQ,DIVQ,NR,BEGC,ENDC,NP,NM,PASS

      WRITE(6,51) MINQ,MAXQ,DIVQ,NR,BEGC,ENDC,NP,NM,PASS

      DO 700 R=1,NR
          READ(5,52) FACTOR(R)
  700 CONTINUE
      IF(PASS .EQ. 1) NSPQ = NSP1
      IF(PASS .EQ. 2) NSPQ = NSP2

      READ(2,21) NSTQ
      IF(PASS .EQ. 1) THEN
          DO 701 L=1,NSP1
              READ(2,22) SPLABL(1,L),SPLABL(2,L)
  701     CONTINUE
          DO 773 L=1,NSP2
              READ(2,22) DUM2
  773     CONTINUE
        ELSE
          DO 771 L=1,NSP1
              READ(2,22) DUM2
  771     CONTINUE
          DO 772 L=1,NSP2
              READ(2,22) SPLABL(1,L),SPLABL(2,L)
  772     CONTINUE
      ENDIF
      READ(2,21) NC
      IF(NC .NE. NP*NM) THEN
          WRITE(6,63) NC,NP,NM
```

```
      STOP 3
 63   FORMAT(1HO,'ERROR...NUMBER OF CONDITIONS SPECIFIED IN ',
     .       'LABEL FILE IS',I3,'. '/6X,'THIS DOES NOT EQUAL ',
     .       'THE PRODUCT OF THE NUMBER OF EXCEEDANCE ',
     .       'PROBABILITIES (',I2,') TIMES THE NUMBER OF ',
     .       'MONTHS (',I2,')'/6X,'GIVEN ON THE RUN PARAMETERS ',
     .       'CARD. '/6X,'PROGRAM STOPPED. ')
      ENDIF
      DO 702 C=1,NC
          READ(2,23) (CLABL(K,C),K=1,7)
 702  CONTINUE

  21  FORMAT(I3)
  22  FORMAT(2A10)
  23  FORMAT(7A10)
  50  FORMAT(7A10)
  51  FORMAT(9I5)
  52  FORMAT(F6.3)
  53  FORMAT(A5,1X,I2)
  54  FORMAT(I2,5X,6I2)
  55  FORMAT(2A10,5X,A10)

*-----READ IN THE WUA, GROSS AREA, AND DISCHARGE VALUES FOR THE
*     NR REACHES (EACH REACH SEPERATED BY END-OF-FILE)

      DO 703 R=1,NR
          D = 0
 601      D = D + 1
          IF(D .GT. MAXD) THEN
              READ(1,11,END=602) DUM
              WRITE(6,61) MAXD,R
              STOP 1
          ENDIF
      IF(PASS .EQ. 1)
     .    READ(1,11,END=602) FLOW(R,D),(WUA(R,D,I),I=1,NSPQ)
      IF(PASS .EQ. 2)
     .    READ(1,12,END=602) FLOW(R,D),(WUA(R,D,I),I=1,NSPQ)
          GOTO 601
 602      NDIS(R) = D - 1
*         CLEAR END-OF-FILE INDICATOR BEFORE READING IN NEXT REACH
          IF(EOF(1) .NE. 0.) CONTINUE

 703  CONTINUE

      DO 750 R=1,NR
      DO 750 D=1,NDIS(R)
          DO 751 L=1,NSPQ
              WUA(R,D,L) = FACTOR(R)*WUA(R,D,L)
 751      CONTINUE
 750  CONTINUE

  11  FORMAT(I5,14F9.0/5(5X,14F9.0/)///////)
  12  FORMAT(I5//////6(5X,14F9.0/),5X,14F9.0)
  61  FORMAT(1X,'ERROR...MORE THAN',I4,' DISCRETE DISCHARGES IN ',
     .       'REACH',I3,'.  PROGRAM STOPPED - MUST RE-DIMENSION ',
     .       'ARRAYS. ')

*-----READ IN THE NC CONDITIONS FOR EACH REACH
*     ONE LINE PER EXCEEDANCE PROBABILITY, NM CONDITIONS ON A LINE
```

```
      DO 704 R=1,NR
      DO 704 P=1,NP
          I1 = 1 + NM*(P-1)
          I2 = P*NM
          READ(3,31) (COND(R,I),I=I1,I2)
  704 CONTINUE

   31 FORMAT(5X,12I5)

*-----NOW LOOP ON THE NC CONDITIONS, SUMMING WUA AND GROSS AREA
*     ACROSS REACHES AFTER ADJUSTING THE DISCHARGE VALUES.
*     TABLE THE RESULTS IN UP TO THREE TABLES.

      ND = (MAXQ - MINQ)/DIVQ + 1
      NSUB = ND/15
      LASTQ = 15
      IF(MOD(ND,15) .NE. 0) THEN
          NSUB = NSUB + 1
          LASTQ = MOD(ND,15)
      ENDIF

      DO 710 C=1,NC

          IF(C .LE. NM) THEN
              M = C
              PROB = .90
            ELSEIF(C .LE. 2*NM) THEN
              M = C - NM
              PROB = .50
            ELSE
              M = C - 2*NM
              PROB = .10
          ENDIF

          DO 711 D=1,MAXD
          DO 711 L=1,MAXSTQ
              SUMW(D,L) = 0.
  711     CONTINUE

          DO 712 R=1,NR
              D2 = 0
              DO 713 D=1,NDIS(R)
                  ADJFLO = FLOW(R,D) - COND(R,C)
                  IF(ADJFLO.GE.MINQ .AND. ADJFLO.LE.MAXQ) THEN
                      D2 = D2 + 1
                      DO 714 L=1,NSPQ
                          SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
  714                 CONTINUE
                  ENDIF
  713         CONTINUE
              IF(D2 .NE. ND) THEN
                  WRITE(6,62) R,MINQ,MAXQ,COND(R,C),(CLABL(I,C),I=1,8)
                  STOP 2
              ENDIF
   62 FORMAT(1HO,'ERROR IN REACH',I3,' - RANGE OF DISCHARGES DOES ',
     .       'NOT COVER THE DESIRED RANGE (',I5,' TO ',I5,') AFTER ',
     .       'SUBTRACTING TRIBUTARY'/' INFLOW VALUE (',I5,') FOR THE ',
     .       'FOLLOWING CONDITION: '/1X,8A10)
  712     CONTINUE
```

```
      DO 715 L=1,NSPQ
          WMAX(L) = 0.
          DO 716 D=1,ND
              WMAX(L) = MAX(WMAX(L),SUMW(D,L))
716       CONTINUE
          DO 785 I=1,NPCT
              CRIT(I) = -99
785       CONTINUE
          Q = MINQ - DIVQ
          D = 0
717       IF(CRIT(1).NE.-99 .OR. D.EQ.ND) GOTO 727
              D = D + 1
              Q = Q + DIVQ
              IF(SUMW(D,L) .EQ. WMAX(L)) THEN
                  SUMW(D,L) = 100.
                ELSE
                  SUMW(D,L) = 100.*(SUMW(D,L)/WMAX(L))
              ENDIF
              DO 786 I=1,NPCT
                  IF(SUMW(D,L).GE.PCTS(I) .AND.
                        CRIT(I).EQ.-99) CRIT(I) = Q
786           CONTINUE
              IF(CRIT(1) .NE. -99) DMAX(L) = D
          GOTO 717
727       CONTINUE

*     FIND SPNQ GROUP NUMBER AND POSITION IN GROUP FOR THIS VALUE OF L

          IF(PASS .EQ. 1) L2 = L
          IF(PASS .EQ. 2) L2 = L + NSP1
          IF(MOD(L2+2,3) .EQ. 0) THEN
              Q = (L2+2)/3
              POS = 1
            ELSEIF(MOD(L2+1,3) .EQ. 0) THEN
              Q = (L2+1)/3
              POS = 2
            ELSEIF(MOD(L2,3).EQ.0) THEN
              Q = L2/3
              POS = 3
          ENDIF
          WRITE(7,71) M,PROB,Q,POS,SPLABL(1,L),(CRIT(I),I=1,NPCT)
 71       FORMAT(I2,F4.2,I3,I2,1X,A10,12I6)
715   CONTINUE

      Q = MINQ - DIVQ
      NQ = 15
      DO 718 I=1,NSUB
          WRITE(4,41) SPNAME,(CLABL(K,C),K=1,7),RUNLABL
          DO 719 J=1,15
              QLABL(J) = 1H
719       CONTINUE
          IF(I .EQ. NSUB) NQ = LASTQ
          DO 720 J=1,NQ
              Q = Q + DIVQ
              CALL INTCOD(QLABL(J),1,7,Q)
720       CONTINUE
          IF(I.EQ.1) WRITE(4,45) QLABL
          IF(I.GT.1) WRITE(4,42) QLABL
          D1 = 15*(I - 1) + 1
```

```
      DO 721 L=1,NSPQ
        IF(L .EQ. 55) THEN
          WRITE(4,41) SPNAME,(CLABL(K,C),K=1,7),RUNLABL
          IF(I.EQ.1) WRITE(4,45) QLABL
          IF(I.GT.1) WRITE(4,42) QLABL
        ENDIF
        IF(D1 .LE. DMAX(L)) THEN
          D2 = MIN(D1+14,DMAX(L))
        ELSE
          D2 = 0
        ENDIF
        IF(I.EQ.1) THEN
          IWMAX = NINT(WMAX(L))
          WRITE(4,43) SPLABL(1,L),SPLABL(2,L),
     .        (SUMW(D,L),D=D1,D2)
          WRITE(4,46) IWMAX
        ELSEIF(D2 .NE. 0) THEN
          WRITE(4,43) SPLABL(1,L),
     .    SPLABL(2,L),(SUMW(D,L),D=D1,D2)
        ELSE
          WRITE(4,43) SPLABL(1,L)
        ENDIF
721     CONTINUE
718   CONTINUE

 41 FORMAT(1H1,A10,20X,'WEIGHTED USABLE AREA AS A PERCENT OF THE ',
     .    'MAXIMUM EFFECTIVE SPAWNING WUA'/
     .    1X,7A10,5X,6A10/60X,'GORGE DISCHARGE')
 42 FORMAT(1X,'SPAWNING Q/EXC. P',4X,15A7/
     .    1X,20(1H-),1X,105(1H-))
 43 FORMAT(1X,2A10,1X,15F7.3)
 44 FORMAT(1X,2A10,1X,15F7.4)
 45 FORMAT(1X,'SPAWNING Q/EXC. P',4X,15A7,3X,'MAX WUA'/
     .    1X,20(1H-),1X,105(1H-),2X,8(1H-))
 46 FORMAT(1H+,126X,I10)

710 CONTINUE

    STOP
    END
```

```
      PROGRAM WUA2EF3(TAPE1, TAPE2, TAPE3, TAPE4, TAPE7,
     .                INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT)

C*********************************************************************
C*    READS (1) TABLED WUA AND GROSS AREA BY DISCHARGE, FOR EACH     *
C*              REACH,                                               *
C*          (2) NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH        *
C*          (3) A LABEL FILE                                         *
C*    AND PRODUCES A TABLE  OF OUTPUT (SUMMED ACROSS REACHES):       *
C*          (1) WUA FOR EACH DISCHARGE AS PERCENT OF THE GREATEST    *
C*              WUA, FOR EACH LIFE STAGE (EFFECTIVE SPAWNING WUA)    *
C*********************************************************************

C-----FILES:
C
C     TAPE1 = INPUT, WUA TABLES PRODUCED BY PROGRAM WUA1, END-OF-
C             FILE BETWEEN TABLES FOR EACH REACH
C     TAPE2 = INPUT, LABEL FILE
C     TAPE3 = INPUT, NC TRIBUTARY INFLOW CONDITIONS FOR EACH REACH
C     TAPE4 = OUTPUT, TABLED WUA AS PERCENT OF GREATEST WUA
C     TAPE5 = STANDARD INPUT FILE
C     TAPE6 = STANDARD OUTPUT FILE (ERROR MESSAGES, DIAGNOSTICS)
C     TAPE7 = OUTPUT, FLOWS WHERE CRITICAL PERCENTS ARE REACHED

C-----LOCAL VARIABLES:
      INTEGER      MINQ, MAXQ, DIVQ, NR, NSTQ, NC, MAXR, MAXSTQ, MAXD,
     .             REACH, ND, Q, R, D, C, NSUB, D1, D2, NDIS(7), ADJFLO,
     .             FLOW(7,185), PASS, Q, POS, CRIT(12), DUM2(2),
     .             COND(7,36), SPLABL(2,57), CLABL(7,36), QLABL(15),
     .             BEGC, ENDC, RUNLABL(6), DUM, SPNAME, DMAX(57)

      REAL         SUMW(185,57), WMAX(57), PCTS(12), WUA(7,185,57),
     .             FACTOR(7)

C     DEFINITIONS
C     PASS = 1   READS FIRST 6 LINES FOR EACH FLOW AND PROCESSED
C                FIRST 84 SPAWNING Q'S
C     PASS = 2   READS LINES 7-13 FOR EACH FLOWS AND PROCESSSS LAST
C                57 SPAWNING Q'S
C
C     MINQ,MAXQ = MINIMUM AND MAXIMUM DISCHARGE VALUES TO BE TABLED
C     DIVQ =      SIZE OF INTERVAL BETWEEN TABLED DISCHARGE VALUES
C     NR =        NUMBER OF REACHES BEING PROCESSED
C     NSTQ =      NUMBER OF SPECIES-LIFE STAGE CODES
C     NC =        NUMBER OF TRIBUTARY INFLOW CONDITIONS TO BE TABLED
C     MAXR =      MAXIMUM NUMBER OF REACHES THAT CAN BE PROCESSED
C                 WITHOUT RE-DIMENSIONING ARRAYS
C     MAXSTQ =    MAXIMUM NUMBER OF SPECIES-LIFE STAGE CODES
C     MAXD =      MAXIMUM NUMBER OF DISCHARGE LEVELS (BEFORE ADJUSTING FOR
C                 TRIBUTARY INFLOW CONDITIONS AND RESTRICTING TO MINQ-MAXQ)
C     NSUB =      NUMBER OF SUB-TABLES, WITH 15 DISCHARGE LEVELS PER
C                 SUB-TABLE
C     WUA(R,D,L) = WEIGHTED USABLE AREA FOR REACH R, DISCHARGE D,
C                  LIFE STAGE CODE L
C     GROSS(R,D) = GROSS AREA FOR REACH R, DISCHARGE D
C     FLOW(R,D) = DISCHARGE VALUE FOR REACH R, DISCHARGE D
C     COND(R,C) = TRIBUTARY INFLOW CONDITION FOR REACH R, CONDITION C
```

```
*     SPLABL(I,L) = SPECIES-LIFE STAGE LABEL FOR LIFE STAGE L
*     CLABL(I,C) = TABLE HEADING FOR CONDITION C
*     QLABL(I) =  USED IN SUB-TABLE HEADING (15 FLOW VALUES)
*     SUMW(D,L) = WUA SUMMED ACROSS REACHES FOR DISCHARGE LEVEL D,
*                 LIFE STAGE L (THEN BECOMES TABLE 1 VALUES AFTER DIVIDING
*                 BY MAX WUA)
*     WMAX(L) =   MAXIMUM WUA ACROSS ALL DISCHARGE LEVELS FOR A GIVEN
*                 LIFE STAGE L
*     FACTOR(R) = MULTIPLICATIVE FACTOR FOR REACH R, EXPANDS WUA
*                 AND GROSS AREA VALUES TO REPRESENT A SECTION
*                 OF THE RIVER
*     BEGC,ENDC = BEGINING AND ENDING INDEX WITHIN CONDITION
*                 ARRAYS FOR A SUBSET OF THE NC CONDITIONS TO BE USED
*                 IN ANY GIVEN RUN
*     NP =        NUMBER OF EXCEEDANCE PROBABILITES
*     NM =        NUMBER OF MONTHS AT EACH EXCEEDANCE PROBABILITY
*
*-----CONSTANTS:
      DATA MAXR,MAXSTG,MAXD/7,57,185/
      DATA PCTS/100., 99.99, 99.95, 99.9, 99.75, 99.5, 99.25, 99.0,
     .          97.5, 95.0, 92.5, 90./
      NSPQ = 57
      NSP1 = 114
      NPCT = 12
*-----READ PARAMETERS, SPECIES LABELS, CONDITION LABELS
      READ(5,50) SPNAME,RUNLABL
      READ(5,51) MING,MAXQ,DIVQ,NR,BEGC,ENDC,NP,NM,PASS
      DO 700 R=1,NR
          READ(5,52) FACTOR(R)
  700 CONTINUE

      READ(2,21) NSTG
      IF(PASS .EQ. 1) THEN
          DO 701 L=1,NSPQ
              READ(2,22) SPLABL(1,L),SPLABL(2,L)
  701     CONTINUE
          DO 773 L=1,NSP1
              READ(2,22) DUM2
  773     CONTINUE
        ELSEIF(PASS .EQ. 2) THEN
          DO 771 L=1,NSPQ
              READ(2,22) DUM2
  771     CONTINUE
          DO 772 L=1,NSPQ
              READ(2,22) SPLABL(1,L),SPLABL(2,L)
  772     CONTINUE
          DO 774 L=1,NSPQ
              READ(2,22) DUM2
  774     CONTINUE
        ELSE
          DO 775 L=1,NSP1
              READ(2,22) DUM2
  775     CONTINUE
          DO 776 L=1,NSPQ
              READ(2,22) SPLABL(1,L),SPLABL(2,L)
  776     CONTINUE
      ENDIF
      READ(2,21) NC
```

```
      IF(NC .NE. NP*NM) THEN
          WRITE(6,63) NC,NP,NM
          STOP 3
   63     FORMAT(1HO,'ERROR...NUMBER OF CONDITIONS SPECIFIED IN ',
     .           'LABEL FILE IS',I3,'. '/6X,'THIS DOES NOT EQUAL ',
     .           'THE PRODUCT OF THE NUMBER OF EXCEEDANCE ',
     .           'PROBABILITIES (',I2,') TIMES THE NUMBER OF ',
     .           'MONTHS (',I2,')'/6X,'GIVEN ON THE RUN PARAMETERS ',
     .           'CARD. '/6X,'PROGRAM STOPPED. ')
      ENDIF
      DO 702 C=1,NC
          READ(2,23) (CLABL(K,C),K=1,7)
  702 CONTINUE

   21 FORMAT(I3)
   22 FORMAT(2A10)
   23 FORMAT(7A10)
   50 FORMAT(7A10)
   51 FORMAT(9I5)
   52 FORMAT(F6.3)
   53 FORMAT(A5,1X,I2)
   54 FORMAT(I2,5X,6I2)
   55 FORMAT(2A10,5X,A10)

*-----READ IN THE WUA, GROSS AREA, AND DISCHARGE VALUES FOR THE
*     NR REACHES (EACH REACH SEPERATED BY END-OF-FILE)

      DO 703 R=1,NR
          D = 0
  601     D = D + 1
          IF(D .GT. MAXD) THEN
              READ(1,11,END=602) DUM
              WRITE(6,61) MAXD,R
              STOP 1
          ENDIF
      IF(PASS .EQ. 1)
     .    READ(1,11,END=602) FLOW(R,D),(WUA(R,D,I),I=1,NSPQ)
      IF(PASS .EQ. 2)
     .    READ(1,12,END=602) FLOW(R,D),(WUA(R,D,I),I=1,NSPQ)
      IF(PASS .EQ. 3)
     .    READ(1,13,END=602) FLOW(R,D),(WUA(R,D,I),I=1,NSPQ)
          GOTO 601
  602     NDIS(R) = D - 1
*         CLEAR END-OF-FILE INDICATOR BEFORE READING IN NEXT REACH
          IF(EOF(1) .NE. 0.) CONTINUE

  703 CONTINUE

      DO 750 R=1,NR
      DO 750 D=1,NDIS(R)
          DO 751 L=1,NSPQ
              WUA(R,D,L) = FACTOR(R)*WUA(R,D,L)
  751     CONTINUE
  750 CONTINUE

   11 FORMAT(I5,14F9.0/3(5X,14F9.0/),5X,F9.0////////)
   12 FORMAT(I5////14X,13F9.0/3(5X,14F9.0/),5X,2F9.0////)
   13 FORMAT(I5////////23X,12F9.0/3(5X,14F9.0/),5X,3F9.0)
   61 FORMAT(1X,'ERROR...MORE THAN',I4,' DISCRETE DISCHARGES IN ',
     .         'REACH',I3,'.  PROGRAM STOPPED - MUST RE-DIMENSION ',
```

```
     .        'ARRAYS. ')

*------READ IN THE NC CONDITIONS FOR EACH REACH
*      ONE LINE PER EXCEEDANCE PROBABILITY, NM CONDITIONS ON A LINE

       DO 704 R=1,NR
       DO 704 P=1,NP
           I1 = 1 + NM*(P-1)
           I2 = P*NM
           READ(3,31) (COND(R,I),I=I1,I2)
  704  CONTINUE

   31  FORMAT(5X,12I5)

*------NOW LOOP ON THE NC CONDITIONS, SUMMING WUA AND GROSS AREA
*      ACROSS REACHES AFTER ADJUSTING THE DISCHARGE VALUES.
*      TABLE THE RESULTS IN UP TO THREE TABLES.

       ND = (MAXQ - MINQ)/DIVQ + 1
       NSUB = ND/15
       LASTQ = 15
       IF(MOD(ND,15) .NE. 0) THEN
           NSUB = NSUB + 1
           LASTQ = MOD(ND,15)
       ENDIF

       DO 710 C=1,NC

           IF(C .LE. NM) THEN
               M = C
               PROB = .90
             ELSEIF(C .LE. 2*NM) THEN
               M = C - NM
               PROB = .50
             ELSE
               M = C - 2*NM
               PROB = .10
           ENDIF

           DO 711 D=1,MAXD
           DO 711 L=1,MAXSTQ
               SUMW(D,L) = 0.
  711      CONTINUE

           DO 712 R=1,NR
               D2 = 0
               DO 713 D=1,NDIS(R)
                   ADJFLO = FLOW(R,D) - COND(R,C)
                   IF(ADJFLO.GE.MINQ .AND. ADJFLO.LE.MAXQ) THEN
                       D2 = D2 + 1
                       DO 714 L=1,NSPQ
                           SUMW(D2,L) = SUMW(D2,L) + WUA(R,D,L)
  714                  CONTINUE
                   ENDIF
  713          CONTINUE
               IF(D2 .NE. ND) THEN
                   WRITE(6,62) R,MINQ,MAXQ,COND(R,C),(CLABL(I,C),I=1,8)
                   STOP 2
               ENDIF
   62  FORMAT(1HO,'ERROR IN REACH',I3,' - RANGE OF DISCHARGES DOES ',
```

```
     .        'NOT COVER THE DESIRED RANGE (',I5,' TO ',I5,') AFTER ',
     .        'SUBTRACTING TRIBUTARY'/' INFLOW VALUE (',I5,') FOR THE ',
     .        'FOLLOWING CONDITION: '/1X,8A10)
  712    CONTINUE


         DO 715 L=1,NSPQ
             WMAX(L) = 0.
             DO 716 D=1,ND
                 WMAX(L) = MAX(WMAX(L),SUMW(D,L))
  716        CONTINUE
             DO 785 I=1,NPCT
                 CRIT(I) = -99
  785        CONTINUE
             Q = MINQ - DIVQ
             D = 0
  717        IF(CRIT(1).NE.-99 .OR.  D.EQ.ND) GOTO 727
                 D = D + 1
                 Q = Q + DIVQ
                 IF(SUMW(D,L) .EQ.  WMAX(L)) THEN
                     SUMW(D,L) = 100.
                   ELSE
                     SUMW(D,L) = 100.*(SUMW(D,L)/WMAX(L))
                 ENDIF
                 DO 786 I=1,NPCT
                     IF(SUMW(D,L).GE.PCTS(I) .AND.
                         CRIT(I).EQ.-99) CRIT(I) = Q
  786            CONTINUE
                 IF(CRIT(1) .NE.  -99) DMAX(L) = D
             GOTO 717
  727        CONTINUE

*    FIND SPNQ GROUP NUMBER AND POSITION IN GROUP FOR THIS VALUE OF L

             IF(PASS .EQ.  1) L2 = L
             IF(PASS .EQ.  2) L2 = L + NSPQ
             IF(PASS .EQ.  3) L2 = L + NSP1
             IF(MOD(L2+2,3) .EQ.  0) THEN
                 G = (L2+2)/3
                 POS = 1
               ELSEIF(MOD(L2+1,3) .EQ.  0) THEN
                 G = (L2+1)/3
                 POS = 2
               ELSEIF(MOD(L2,3).EQ.0) THEN
                 G = L2/3
                 POS = 3
             ENDIF
             WRITE(7,71) M,PROB,G,POS,SPLABL(1,L),(CRIT(I),I=1,NPCT)
    71       FORMAT(I2,F4.2,I3,I2,1X,A10,12I6)
  715    CONTINUE

         Q = MINQ - DIVQ
         NQ = 15
         DO 718 I=1,NSUB
             WRITE(4,41) SPNAME,(CLABL(K,C),K=1,7),RUNLABL
             DO 719 J=1,15
                 QLABL(J) = 1H
  719        CONTINUE
             IF(I .EQ.  NSUB) NQ = LASTQ
             DO 720 J=1,NQ
```

```
            Q = Q + DIVQ
            CALL INTCOD(QLABL(J),1,7,Q)
720         CONTINUE
            IF(I.EQ.1) WRITE(4,45) QLABL
            IF(I.GT.1) WRITE(4,42) QLABL
            D1 = 15*(I - 1) + 1
            DO 721 L=1,NSPQ
               IF(L .EQ. 55) THEN
                  WRITE(4,41) SPNAME,(CLABL(K,C),K=1,7),RUNLABL
                  IF(I.EQ.1) WRITE(4,45) QLABL
                  IF(I.GT.1) WRITE(4,42) QLABL
               ENDIF
               IF(D1 .LE. DMAX(L)) THEN
                  D2 = MIN(D1+14,DMAX(L))
                ELSE
                  D2 = 0
               ENDIF
               IF(I.EQ.1) THEN
                  IWMAX = NINT(WMAX(L))
                  WRITE(4,43) SPLABL(1,L),SPLABL(2,L),
     .                (SUMW(D,L),D=D1,D2)
                  WRITE(4,46) IWMAX
                ELSEIF(D2 .NE. 0) THEN
                  WRITE(4,43) SPLABL(1,L),
     .            SPLABL(2,L),(SUMW(D,L),D=D1,D2)
                ELSE
                  WRITE(4,43) SPLABL(1,L)
               ENDIF
721         CONTINUE
718      CONTINUE

 41 FORMAT(1H1,A10,20X,'WEIGHTED USABLE AREA AS A PERCENT OF THE ',
     .       'MAXIMUM EFFECTIVE SPAWNING WUA'/
     .       1X,7A10,5X,6A10/60X,'GORGE DISCHARGE')
 42 FORMAT(1X,'SPAWNING Q/EXC. P',4X,15A7/
     .       1X,20(1H-),1X,105(1H-))
 43 FORMAT(1X,2A10,1X,15F7.3)
 44 FORMAT(1X,2A10,1X,15F7.4)
 45 FORMAT(1X,'SPAWNING Q/EXC. P',4X,15A7,3X,'MAX WUA'/
     .       1X,20(1H-),1X,105(1H-),2X,8(1H-))
 46 FORMAT(1H+,126X,I10)

710 CONTINUE

    STOP
    END
```

FILE: XSUMTAB

```
/JOB
A.
/ACCOUNT
ATTACH,FILESET=C13SET/M=W.
GF,CPPTBL1.
GF,CPPTBL2.
GF,CPPTBL3.
COPYBF,CPPTBL1,DATA.
COPYBF,CPPTBL2,DATA.
COPYBF,CPPTBL3,DATA.
REWIND,DATA.
PACK,DATA.
GET,SUMTAB.
FTN5,I=SUMTAB.
LGO,DATA,DUM,CPTABLE.
RF,CPTABLE.
/EOR
  3271
```

```
      PROGRAM SUMTAB (TAPE1,TAPE2,TAPE3,INPUT,OUTPUT,
     *TAPE5=INPUT,TAPE6=OUTPUT)
C***THIS PROGRAM PRODUCES SUMMARY TABLES OF GORGE DISCHARGES THAT
C    PROVIDE PERCENTAGES OF WUA. PROGRAMMED FOR THE SKAGIT RIVER
C    PROJECT BY W. MEYER, SPRING 1984.
C    FOR USE WITH THE FTN5 COMPILER
C***FILE USE:
C    TAPE1 = RAW DATA FILE
C    TAPE2 = DUMMY FILE OF SORTED DATA
C    TAPE3 = OUTPUT FILE OF TABLES
C    TAPE5 = INPUT FILE OF SPECIES CODE AND INITIAL PAGE NUMBER
C    TAPE6 = SYSTEM OUTPUT FILE
C
C***PROGRAM IS IN THREE PARTS; 1- INITIALIZATION, 2- MAIN READ LOOP
C    AND 3- MAIN READ AND WRITE LOOP.
C***PROGRAM UTILIZES FASTIO, SORTMERGE, AND STRING MANIPULATION ROUTINES.
C
      DIMENSION IREC(10), IREC2(11), IFISH(4), MONTH(11), ISPMON(4)
      COMMON/A/ MONTH, IFISH, ISPMON
      DATA IREC/10*10H          /
      DATA IREC2/11*10H           /
      DATA MONTH/8HOCTOBER ,8HNOVEMBER,8HDECEMBER,8HJANUARY ,
     *8HFEBRUARY,8HMARCH   ,8HAPRIL   ,8HMAY     ,8HJUNE    ,
     *8HJULY    ,8HAUGUST  /
      DATA IFISH/10HCHINOOK:  ,10HPINK:      ,10HCHUM:      ,
     *10HSTEELHEAD:/
      DATA ISPMON/9HSEPTEMBER,9HOCTOBER  ,9HDECEMBER ,9HMAY      /
      IPSPWN = 4H
      IPSQUE = 3H
      IBLANK = 4H
      IPMON = 0

C***PART 1.  INITIALIZE SPECIES CODE AND PAGE NUMBER
C    READ FROM ONE LINE DATA FILE FOLLOWING COMMAND FILE
C    ENTER SPECIES FOLLOWED BY PAGE NUMBER IN I3 FORMAT
C***SPECIES 1= CHINOOK, 2 = PINK, 3 = PINK, 4 = STEELHEAD
      READ (5,50) ISPEC, IPAGE
   50 FORMAT (2I3)

C***PART 2.  SORT RECORDS BY MONTH, SPAWNING QUE, SPAWNING EXC, AND
C            INCUBATION EXC.
      CALL SMSORT (100)
      CALL SMFILE ('SORT','FORMATTED',1,'REWIND')
      CALL SMFILE ('OUTPUT','FORMATTED',2,'REWIND')
      CALL SMKEY (2,1,1,0,'DISPLAY','DISPLAY','A')
      CALL SMKEY (13,1,4,0,'DISPLAY','DISPLAY','A')
      CALL SMKEY (21,1,2,0,'DISPLAY','DISPLAY','D')
      CALL SMKEY (5,1,2,0,'DISPLAY','DISPLAY','D')
      CALL SMEND

C***PART 3.  OPEN FASTIO FILE, BEGIN MAIN READ AND WRITE LOOP
      CALL FILEC(2)
   10 CALL READH (2,IREC,10,IREP)
      IF (IREP .LT. 0) GO TO 99
C***READ MONTH, SPAWNING QUE, AND SPAWNING EXC
      CALL INTMOV (IREC,2,1,IMON)
      CALL STRMOV (IREC,13,4,ISPAWN,1)
      CALL STRMOV (IREC,20,3,ISQUE,1)
```

```
C***WRITE NEW PAGE IF MONTH CHANGES OR END OF PAGE
      IF (IMON .NE. IPMON) THEN
        CALL HEADER (LINE, IPAGE, IMON, ISPEC)
        IPMON = IMON
      ELSEIF (LINE .GE. 36) THEN
        CALL HEADER (LINE, IPAGE, IMON, ISPEC)
      ENDIF
C***ONLY WRITE SPAWNING EXC, AND QUE IF CHANGED FROM PREVIOUS RECORD
      IF (ISPAWN .NE. IPSPWN) THEN
        CALL STRMOV (ISPAWN, 1, 4, IREC2, 10)
        IPSPWN = ISPAWN
      ELSE
        CALL STRMOV (IBLANK, 1, 4, IREC2, 10)
      ENDIF
      IF (ISQUE .NE. IPSQUE) THEN
        CALL STRMOV (ISQUE, 1, 3, IREC2, 15)
        IPSQUE=ISQUE
      ELSE
        CALL STRMOV (IBLANK, 1, 3, IREC2, 15)
      ENDIF
      CALL STRMOV (IREC, 4, 3, IREC2, 22)
      CALL STRMOV (IREC, 23, 72, IREC2, 29)

      WRITE (3, 301) (IREC2(I), I=1, 11)
  301 FORMAT (11A10)
      LINE = LINE + 1
      GO TO 10

   99 STOP
      END
      SUBROUTINE HEADER (LINE, IPAGE, IMON, ISPEC)
C***THIS SUBROUTINE WRITES THE PAGE HEADER INFORMATION.
      COMMON/A/ MONTH, IFISH, ISPMON
      DIMENSION MONTH(11), IFISH(4), ISPMON(4)
C***ADJUST MONTH INDEX DEPENDING ON SPECIES CODE
      IF (ISPEC .EQ. 1) INDEX = IMON
      IF (ISPEC .EQ. 2) INDEX = IMON + 1
      IF (ISPEC .EQ. 3) INDEX = IMON + 3
      IF (ISPEC .EQ. 4) INDEX = IMON + 8

      WRITE (3, 300) IPAGE, ISPEC, IFISH(ISPEC), ISPMON(ISPEC), MONTH(INDEX)
  300 FORMAT (1H1, /, 92X, 'PAGE ', I3, /,
     *8X, 'TABLE ', I1, '. ', ' GORGE RELEASE (CFS) PROVIDING INDICATED % ',
     *'OF MAX EFFECTIVE SPAWNING WUA - ', A10, /,
     *9X, 'SPAWNING MONTH - ', A9, 38X, 'INCUBATION MONTH - ', A8, //,
     *9X, 'SPAWNING INCUBATION', /,
     *10X, 'Q    EXC.    EXC.     100.0 99.99 99.95 99.90 99.75 99.50',
     *' 99.25 99.00 97.50 95.00 92.50 90.00', /,
     *9X, 91('-'))
      IPAGE = IPAGE + 1
      LINE = 0
      RETURN
      END
```

APPENDIX 2

Fileset listings and file itemizations
for files stored on magnetic tape

CATALOG OF S2 FILE 1 84/04/11. 17.06.14. PAGE 1

REC	NAME	TYPE	LENGTH	CKSUM	DATE	COMMENTS
1	XEQBED	ABS	6162	6222	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
2	XEQWUA2	ABS	11214	3474	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
3	WUASET	ABS	624244	3157	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
4	SORCSET	ABS	4502	7231	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
5	INCUBFS	ABS	16120	7723	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
6	PROCFIL	ABS	12010	4727	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
7	HYDRSET	ABS	651365	2054	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
8	IFMLIB	ABS	222411	0520	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
9	IFG4SET	ABS	155637	1412	84/04/07.	ARCHIVED, 84/04/07. 00.27.47.
10	SBEDSET	ABS	4111624	4026	84/04/06.	ARCHIVED, 84/04/06. 15.02.24.
11	IFG4	ABS	17362	5010	84/04/06.	ARCHIVED, 84/04/06. 15.02.24.
12	CBEDSET	ABS	4363123	4747	84/04/06.	ARCHIVED, 84/04/06. 15.02.24.
13	PHABSIM	ABS	272367	0672	84/04/06.	ARCHIVED, 84/04/06. 15.02.24.
14	KBEDSET	ABS	4256320	1537	84/04/06.	ARCHIVED, 84/04/06. 15.02.24.
15	PROGSET	ABS	1310077	6350	84/04/06.	ARCHIVED, 84/04/06. 15.02.24.
16	PBEDSET	ABS	4265534	4462	84/04/06.	ARCHIVED, 84/04/06. 15.02.24.
17	S5OUT	ABS	703301	7654	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
18	S1OUT	ABS	460731	2644	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
19	S3OUT	ABS	645517	1262	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
20	S2OUT	ABS	355576	2463	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
21	SPNQSET	ABS	56123	1512	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
22	S8OUT	ABS	1127235	7075	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
23	ELEVSET	ABS	1306012	4777	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
24	S4OUT	ABS	543067	6733	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
25	S7OUT	ABS	1135457	4631	84/04/06.	ARCHIVED, 84/04/06. 09.04.52.
26	TP13S8	ABS	4367501	1076	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
27	TP13I8	ABS	1053627	2445	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
28	TP13S7	ABS	4606161	1472	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
29	TP13S3	ABS	3246755	2042	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
30	TP13S2	ABS	1531165	7472	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
31	TP13S1	ABS	3107125	5320	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
32	TP13I5	ABS	710302	1756	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
33	TP13I1	ABS	467656	5347	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
34	TP13I4	ABS	472130	3757	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
35	TP13S5	ABS	3535415	5075	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
36	TP13I2	ABS	314706	6540	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
37	TP13I3	ABS	634076	4710	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
38	TP13I7	ABS	1116273	0427	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
39	TP13S4	ABS	2422735	3512	84/04/05.	ARCHIVED, 84/04/05. 15.21.27.
40	* EOF *		SUM = 71604514			

SKAG1, SKAG2, SKAG3

CATALOG OF NEW2 FILE 1 84/04/11. 16.47.54. PAGE 1

REC	NAME	TYPE	LENGTH	CKSUM	DATE	COMMENTS
1	C13SET	ABS	3303231	4304	84/04/11.	ARCHIVED, 84/04/11. 16.35.34.
2	P13SET	ABS	4213302	3670	84/04/11.	ARCHIVED, 84/04/11. 16.35.34.
3	S13SET	ABS	4233345	0150	84/04/11.	ARCHIVED, 84/04/11. 16.35.34.
4	K13SET	ABS	4467012	2644	84/04/11.	ARCHIVED, 84/04/11. 16.35.34.

SKAG1B, SKAG2B, SKAG3B

* EOI * SUM = 20441112

84/07/16 19.38.24. PROGSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
BDELV	1763	84/03/22 20.40.28	84/03/22 20.40.28	COMPILED (BINARY) BEDELV
BEDELV	14	84/02/15 18.32.41	84/03/21 23.30.53	PROGRAM TO PRODUCE WUA1-TYPE DATA TABLES FROM SUMMED WUA OF WATE
BEDELV1	14	84/03/22 09.20.54	84/07/11 19.11.56	
CCOND	3	84/02/24 15.42.45	84/03/30 19.31.25	
CLABF	7	84/02/24 09.34.20	84/02/24 15.29.15	
CONDTNS	4	83/09/21 09.50.47	84/04/13 01.03.38	36 CONDITIONS (3 LINES WITH 12 VALUES EACH) FOR THE 7 REACHES
EFSPWN	9	84/03/17 15.59.10	84/03/17 21.36.30	
ELEVATE	4	84/03/19 22.07.42	84/03/21 23.07.44	
EXTH	1	83/08/31 17.15.20	83/08/31 17.15.20	BATCH JOB TO EXTRACT HEADER LINES FROM FISHFIL
KCOND	4	84/02/24 15.31.36	84/03/21 07.25.24	
KLABF	8	84/02/24 09.30.09	84/02/24 15.28.36	
LABF	8	83/09/13 19.13.06	83/11/28 14.39.10	LABEL FILE USED BY PROGRAM WUA1 AND WUA2
LABF1	8	84/01/11 11.53.07	84/04/02 12.24.22	
LABF2	8	83/11/28 10.36.03	83/11/29 08.56.40	
LABF3	8	84/04/02 12.25.07	84/04/02 19.00.59	
LHNAME	5	83/08/31 17.18.08	83/08/31 17.18.08	HEADER RECORDS EXTRACTED FROM FISHFIL
MAXPROG	3	84/07/12 09.50.01	84/07/12 09.50.01	
PATCH	2	83/10/18 14.30.41	83/12/20 10.58.52	
PATCH2	2	84/01/09 09.41.30	84/01/09 11.54.57	
PCOND	4	84/02/28 07.40.17	84/03/21 07.22.35	
PLABF	8	84/02/24 09.31.40	84/02/24 15.27.38	
SCOND	2	84/02/24 15.44.37	84/03/21 08.09.05	
SCONDA	3	84/04/13 01.03.45	84/04/13 01.03.45	
SCONDB	3	84/04/13 01.07.37	84/04/13 01.07.37	
SLABF	7	84/02/24 09.36.07	84/04/13 07.36.39	
SLABFA	7	84/04/13 07.36.47	84/04/13 07.38.15	
SLABFB	7	84/04/13 07.37.39	84/04/13 07.37.39	
SPSWTE1	1	83/12/17 13.28.22	83/12/17 13.28.22	
SPSWTO1	1	83/12/17 13.28.34	83/12/17 13.28.34	
SUMMIT	2	84/07/12 09.50.06	84/07/12 09.50.06	
SUMTAB	8	84/07/12 09.50.12	84/07/12 09.50.12	
WUA1	13	83/09/13 11.13.13	83/09/20 15.33.40	WUA PROGRAM 1. EXTRACT WUA DATA FROM HABTAT OUTPUT
WUA2	33	83/11/22 18.21.06	83/11/29 10.15.47	NEW VERSION WITH INCUBATION DATA PROCESSING ADDED
WUA2A	33	84/01/25 16.22.44	84/04/03 10.33.57	
WUA2EB2	1752	84/03/23 10.43.15	84/03/23 10.43.15	COMPILED (BINARY) WUA2EF2
WUA2EB3	1442	84/03/25 08.32.16	84/03/25 08.32.16	COMPILED (BINARY) WUA2EF3
WUA2EFF	19	84/03/07 14.04.24	84/03/07 14.04.24	
WUA2EF2	22	84/03/07 14.17.37	84/07/11 19.10.41	
WUA2EF3	22	84/03/07 14.05.16	84/07/11 19.11.27	
WUA3	13	83/11/28 14.04.16	83/11/28 15.32.33	READS A PERCENT TABLE FILE, APPLIES SPS. WEIGHTING AND SUMS ACRO
XEQBED	1	84/02/15 18.52.46	84/03/17 16.51.43	SAMPLE XEQ FILE FOR BEDELV
XEQEFSP	1	84/03/17 16.57.20	84/03/19 21.31.58	
XSUMTAB	3	84/07/12 09.50.16	84/07/12 09.50.16	
OLD/WUA2	18	83/09/19 17.31.29	83/10/27 10.51.49	
OLD/WUA3	20	83/10/31 17.01.29	83/11/02 11.01.45	WAS WUA2B, SUMMED+WEIGHTED PERCENT TABLES ACROSS SPS.
PATCH/XEQ	1	83/10/18 14.45.40	83/10/18 14.45.40	
WUA1/XEQ	1	83/09/13 19.28.08	83/09/22 08.18.32	RUN PROGRAM WUA1 FOR REACH 1
WUA1FIX/XEQ	1	83/12/19 10.12.53	84/01/09 11.38.36	
WUA2/XEQ	2	83/09/21 09.47.46	84/07/11 19.15.26	
WUA3/XEQ	1	83/11/01 11.39.21	84/07/11 19.15.55	

50 ELEMENTS, 99.1 PERCENT OF CURRENT FILESET SPACE
(NE= 250, DIRECTORY 31.6 PERCENT FULL)

84/04/07 00.22.13. XEQBED CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
CBEDX1	1	84/03/19 15.47.12	84/03/23 07.21.03	
CBEDX2	1	84/03/19 15.47.21	84/03/23 07.09.05	
CBEDX3	1	84/03/19 15.47.31	84/03/23 07.23.01	
CBEDX4	1	84/03/19 15.47.39	84/03/23 07.24.28	
CBEDX5	1	84/03/19 16.06.08	84/03/23 07.25.27	
CBEDX7	1	84/03/19 16.15.36	84/03/23 07.26.32	
CBEDX8	1	84/03/19 16.20.44	84/03/23 07.30.51	
KBEDX1	1	84/03/19 15.47.51	84/03/23 07.39.52	
KBEDX2	1	84/03/19 15.47.58	84/03/23 07.40.47	
KBEDX3	1	84/03/19 15.48.06	84/03/23 07.41.51	
KBEDX4	1	84/03/19 15.48.13	84/03/23 07.42.32	
KBEDX5	1	84/03/19 16.09.42	84/03/23 07.43.25	
KBEDX7	1	84/03/19 16.11.44	84/03/23 07.44.23	
KBEDX8	1	84/03/19 16.23.41	84/03/23 07.45.07	
PBEDX1	1	84/03/19 15.50.27	84/03/23 07.04.03	
PBEDX2	1	84/03/19 15.50.35	84/03/23 07.10.01	
PBEDX3	1	84/03/19 15.50.41	84/03/23 07.10.56	
PBEDX4	1	84/03/19 15.50.49	84/03/23 07.13.23	
PBEDX5	1	84/03/19 16.07.48	84/03/23 07.14.09	
PBEDX7	1	84/03/19 16.13.56	84/03/23 07.15.35	
PBEDX8	1	84/03/19 16.22.02	84/03/23 07.16.31	
SBEDX1	1	84/03/19 15.50.54	84/03/23 07.31.27	
SBEDX2	1	84/03/19 15.51.00	84/03/23 07.33.45	
SBEDX3	1	84/03/19 15.51.06	84/03/23 07.34.39	
SBEDX4	1	84/03/19 15.51.13	84/03/23 07.35.26	
SBEDX5	1	84/03/19 15.51.20	84/03/23 07.36.52	
SBEDX7	1	84/03/19 16.16.57	84/03/23 07.37.37	
SBEDX8	1	84/03/19 16.19.27	84/03/23 07.38.27	

84/04/07 00.22.14. XEQWUA2 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
CBPXEQ1	2	84/03/23 11.24.02	84/03/24 17.24.59	
CBPXEQ2	2	84/03/23 11.25.10	84/03/24 17.26.14	
CBXEQ1	1	84/03/23 11.20.30	84/03/24 17.11.20	
CBXEQ2	2	84/03/23 11.22.27	84/03/24 17.14.36	
CPXEQ1	2	84/03/18 20.32.39	84/03/21 07.30.25	
CPXEQ2	2	84/03/18 20.37.34	84/03/21 07.31.01	
CPXEQ3	2	84/03/18 20.44.36	84/03/21 07.31.33	
CXEQ1	2	84/03/07 14.58.22	84/03/23 11.15.53	
CXEQ2	2	84/03/07 14.58.24	84/03/19 22.03.25	
CXEQ3	2	84/03/07 15.08.59	84/03/18 20.45.10	
KBPXEQ1	2	84/03/23 11.30.40	84/03/24 17.29.29	
KBPXEQ2	2	84/03/23 11.31.54	84/03/24 17.31.09	
KBXEQ1	1	84/03/23 11.27.27	84/03/24 17.17.01	
KBXEQ2	2	84/03/23 11.28.49	84/03/24 17.18.41	
KPXEQ1	2	84/03/18 20.55.09	84/03/21 07.28.12	
KPXEQ2	2	84/03/18 20.58.25	84/03/21 07.29.34	
KXEQ1	1	84/03/07 14.58.40	84/03/18 20.53.03	
KXEQ2	2	84/03/07 14.58.42	84/03/18 20.56.19	
PBPXEQ1	2	84/03/23 11.39.10	84/03/24 17.32.56	
PBPXEQ2	2	84/03/23 11.40.22	84/03/24 17.34.37	
PBXEQ1	2	84/03/23 11.36.12	84/03/24 17.21.14	
PBXEQ2	2	84/03/23 11.35.27	84/03/24 17.22.19	
PPXEQ1	2	84/03/18 20.23.12	84/03/21 07.35.19	
PPXEQ2	2	84/03/18 20.26.55	84/03/21 07.36.00	
PXEQ1	2	84/03/07 14.56.27	84/03/23 11.37.33	
PXEQ2	2	84/03/07 14.58.33	84/03/18 20.24.26	
SBPXEQ1	2	84/03/23 11.43.39	84/03/25 09.09.39	
SBPXEQ2	2	84/03/23 11.45.09	84/03/25 09.12.17	
SBPXEQ3	2	84/03/25 09.14.58	84/03/25 09.14.58	
SBXEQ1	2	84/03/23 11.41.31	84/03/25 09.01.34	
SBXEQ2	2	84/03/23 11.42.38	84/03/25 09.07.10	
SBXEQ3	2	84/03/25 09.07.35	84/03/25 09.07.35	
SPXEQ1	2	84/03/18 21.16.35	84/03/21 07.36.32	
SPXEQ2	2	84/03/18 21.18.36	84/03/21 07.36.55	
SXEQ1	2	84/03/07 21.36.56	84/03/07 21.36.56	
SXEQ2	2	84/03/07 14.58.45	84/03/07 21.35.41	

84/04/05 17.18.13. ELEVSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
ELEV1	18	84/02/15 23.27.21	84/02/15 23.27.21	
ELEV2	12	84/02/15 23.28.11	84/02/15 23.28.11	
ELEV3	19	84/02/15 23.29.07	84/02/15 23.29.07	
ELEV4	15	84/02/15 23.29.44	84/02/15 23.29.44	
ELEV5	18	84/02/15 23.30.20	84/02/15 23.30.20	
ELEV7	19	84/02/15 23.30.51	84/02/15 23.30.51	
ELEV8	19	84/02/15 23.31.47	84/02/15 23.31.47	
HOPE1	169	84/03/22 16.15.00	84/03/22 19.01.45	
HOPE2	120	84/03/22 16.14.32	84/03/22 19.03.17	
HOPE4	181	84/03/22 16.02.07	84/03/22 19.06.03	
LST1J2	169	84/03/24 13.27.06	84/03/24 13.27.06	
LST2J2	120	84/03/24 13.27.32	84/03/24 13.27.32	
LST3J2	212	84/03/24 13.27.20	84/03/24 13.27.20	
LST4J2	181	84/03/24 13.26.34	84/03/24 13.26.34	
LST5J2	221	84/03/24 13.48.04	84/03/24 13.51.53	
LST7J2	317	84/03/24 13.26.50	84/03/24 13.26.50	
LST8J2	308	84/03/24 13.27.49	84/03/24 13.27.49	
S1ADJ1	157	84/03/12 21.05.32	84/03/23 11.54.44	
S1ADJ2	157	84/03/12 21.05.42	84/03/23 11.54.45	
S2ADJ1	111	84/03/12 21.21.41	84/03/20 21.06.41	
S2ADJ2	111	84/03/12 21.21.42	84/03/20 21.06.41	
S3ADJ1	196	84/03/12 21.26.20	84/03/22 20.51.30	
S3ADJ2	196	84/03/12 21.26.21	84/03/22 20.51.31	
S4ADJ1	169	84/03/12 21.28.47	84/03/20 21.12.21	
S4ADJ2	169	84/03/12 21.28.48	84/03/20 21.12.22	
S5ADJ1	205	84/03/12 21.32.32	84/03/24 13.47.09	
S5ADJ2	205	84/03/12 21.32.33	84/03/24 13.47.10	
S7ADJ1	293	84/03/12 21.38.26	84/03/22 21.14.59	
S7ADJ2	293	84/03/12 21.38.27	84/03/22 21.15.00	
S8ADJ1	284	84/03/12 21.44.59	84/03/21 22.59.16	
S8ADJ2	284	84/03/12 21.45.01	84/03/21 22.59.17	

84/04/05 17.16.12. SPNQSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
A	95	84/02/18 09.22.34	84/02/18 09.32.13	
COND	4	84/02/14 14.54.10	84/02/14 14.54.10	
RSPNO	119	84/03/21 11.19.50	84/03/21 11.19.50	
SPNQ	2	84/02/15 20.32.27	84/02/15 20.32.27	
SPNQ1	15	84/02/14 15.19.38	84/02/14 15.19.38	
SPNQ2	15	84/02/14 15.13.09	84/02/14 16.12.30	
SPNQ3	15	84/02/14 15.11.35	84/02/14 15.22.21	
SPNQ4	15	84/02/14 15.10.36	84/02/14 15.10.36	
SPNQ5	15	84/02/14 15.12.33	84/03/17 12.34.35	
SPNQ7	15	84/02/14 15.14.59	84/03/17 12.34.49	
SPNQ8	15	84/02/14 15.15.50	84/03/17 12.34.42	

84/04/07 00.11.22. WUASET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
EDIT1	153	84/04/03 20.45.25	84/04/03 20.45.25	
EDIT2	153	84/04/03 21.02.10	84/04/03 21.02.45	
LABF1	1	84/01/30 07.33.57	84/01/30 07.33.57	
NEWTBL1	38	84/01/09 12.13.34	84/01/09 12.13.34	
NEWTBL2	40	84/01/09 11.59.08	84/01/09 11.59.08	
NEWTBL3	44	84/01/09 11.58.45	84/01/09 11.58.45	
NEWTBL4	46	84/01/09 11.56.05	84/01/09 11.56.05	
NEWTBL5	51	84/01/09 12.01.08	84/01/09 12.01.08	
NEWTBL7	65	84/01/09 12.01.34	84/01/09 12.01.34	
NEWTBL8	65	84/01/09 12.02.00	84/01/09 12.02.00	
NTBL1	14	84/01/18 10.41.25	84/01/18 10.41.25	
NTBL2	14	84/01/18 11.02.07	84/01/18 11.02.07	
NTBL3	16	84/01/18 11.02.23	84/01/18 11.02.23	
NTBL4	17	84/01/18 11.02.46	84/01/18 11.02.46	
NTBL5	18	84/01/19 15.55.31	84/01/19 15.55.31	
NTBL7	23	84/01/18 11.17.43	84/01/30 07.23.37	
NTBL8	23	84/01/18 11.21.43	84/01/30 07.31.29	
NXP1	24	84/01/30 14.51.17	84/01/30 15.39.05	
NXP2	25	84/01/30 14.56.09	84/02/01 08.32.57	
NXP3	27	84/01/30 15.00.24	84/01/30 15.00.24	
NXP4	30	84/01/30 15.01.10	84/01/30 15.01.10	
NXP5	32	84/01/30 15.03.18	84/01/30 15.03.18	
NXP7	40	84/01/30 15.04.13	84/01/30 15.04.13	
NXP8	40	84/01/30 15.04.53	84/01/30 15.04.53	
PNTBL1	12	84/04/02 11.13.08	84/04/02 11.13.08	
PNTBL2	13	84/04/02 11.15.19	84/04/02 11.15.19	
PNTBL3	14	84/04/02 11.15.47	84/04/02 11.15.47	
PNTBL4	15	84/04/02 11.16.20	84/04/02 11.16.20	
PNTBL5	16	84/04/02 11.16.51	84/04/02 11.16.51	
PNTBL7	20	84/04/02 11.18.13	84/04/02 11.18.13	
PNTBL8	20	84/04/02 11.17.50	84/04/02 11.17.50	
PSTBLN1	10	84/01/11 15.54.15	84/01/11 15.54.15	
PSTBLN2	10	84/01/11 15.53.42	84/01/11 15.53.42	
PSTBLN3	11	84/01/11 15.53.02	84/01/11 15.53.02	
PSTBLN4	12	84/01/11 15.52.28	84/01/11 15.52.28	
PSTBLN5	12	84/01/11 15.51.19	84/01/19 15.51.20	
PSTBLN7	16	84/01/11 15.50.09	84/01/11 15.50.09	
PSTBLN8	16	84/01/11 15.49.35	84/01/11 15.49.35	
PSTBL1	10	84/01/11 14.39.50	84/01/19 15.40.54	
PSTBL2	10	84/01/11 15.18.54	84/01/11 15.23.35	
PSTBL3	11	84/01/11 15.40.17	84/01/11 15.40.17	
PSTBL4	12	84/01/11 15.41.13	84/01/11 15.41.13	
PSTBL5	12	84/01/11 15.41.40	84/01/19 15.46.23	
PSTBL7	16	84/01/11 15.42.30	84/01/11 15.42.30	
PSTBL8	16	84/01/11 15.43.20	84/01/11 15.43.20	
RCURV1	153	84/04/03 18.56.58	84/04/03 18.56.58	
RCURV2	153	84/04/03 18.57.15	84/04/03 18.57.16	
SEGP1	24	84/01/30 15.11.00	84/02/01 08.19.27	
SEGP2	25	84/01/30 15.16.17	84/01/31 15.22.15	
SEGP3	27	84/01/30 15.16.32	84/01/31 15.23.17	
SEGP4	30	84/01/30 15.17.09	84/01/31 15.23.39	
SEGP5	32	84/01/30 15.17.27	84/01/31 15.24.11	
SEGP7	40	84/01/30 15.22.21	84/01/31 15.27.26	
SEGP8	40	84/01/30 15.18.04	84/02/01 08.14.44	
SEG1	13	84/01/30 08.50.13	84/02/01 08.18.14	
SEG2	13	84/01/30 08.50.20	84/01/31 15.28.34	
SEG3	15	84/01/30 08.50.24	84/01/31 15.27.15	
SEG4	16	84/01/30 08.50.32	84/01/31 15.27.22	
SEG5	17	84/01/30 08.50.36	84/01/31 15.27.30	
SEG7	22	84/01/30 08.50.43	84/01/31 15.27.36	
SEG8	22	84/01/30 08.50.49	84/01/31 15.27.43	

WPCTBL1	74	84/02/08 11.45.06	84/02/08 15.06.15
WPCTBL2	77	84/02/08 15.10.31	84/02/08 15.10.31
WPCTBL3	85	84/02/08 15.19.14	84/02/08 15.19.14
WPCTBL4	93	84/02/08 15.19.38	84/02/08 15.19.38
WPCTBL5	98	84/02/08 15.20.09	84/02/08 15.20.09
WPCTBL7	126	84/02/08 15.21.03	84/02/08 15.21.03
WPCTBL8	126	84/02/08 15.21.31	84/02/08 15.21.31
WUATBL1	37	84/02/07 16.07.53	84/02/08 09.14.40
WUATBL2	39	84/02/08 10.00.05	84/02/08 10.00.05
WUATBL3	43	84/02/08 10.00.45	84/02/08 10.00.45
WUATBL4	47	84/02/08 10.00.34	84/02/08 10.00.34
WUATBL5	50	84/02/08 09.59.37	84/02/08 09.59.37
WUATBL7	64	84/02/08 09.59.50	84/02/08 09.59.50
WUATBL8	64	84/02/08 10.00.19	84/02/08 10.00.19

WUASET p.2

84/04/05 11.16.43. TP13S1 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
TP13CS1	2438	84/02/11 10.14.32	84/02/11 10.45.37	
TP13KS1	2438	84/02/11 10.12.28	84/02/11 10.43.52	
TP13PS1	2438	84/02/11 10.13.30	84/02/11 10.44.44	
TP13SS1	2438	84/02/11 10.15.41	84/02/11 10.46.33	

84/04/05 11.16.44. TP13S2 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
TP13CS2	1638	84/02/11 11.06.35	84/02/11 11.06.35	
TP13KS2	1638	84/02/11 11.04.41	84/02/11 11.04.41	
TP13PS2	1638	84/02/11 11.05.37	84/02/11 11.05.37	
TP13SS2	1638	84/02/11 11.07.28	84/02/11 11.07.28	

84/04/05 11.16.45. TP13S3 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
TP13CS3	3232	84/02/11 11.19.13	84/02/11 11.19.13	
TP13KS3	3232	84/02/11 11.16.49	84/02/11 11.16.49	
TP13PS3	3232	84/02/11 11.17.58	84/02/11 11.17.58	
TP13SS3	3232	84/02/11 11.20.38	84/02/11 11.20.38	

84/04/05 11.16.46. TP13S4 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
TP13CS4	2442	84/02/11 11.24.20	84/02/11 11.24.20	
TP13KS4	2442	84/02/11 11.21.57	84/02/11 11.21.57	
TP13PS4	2442	84/02/11 11.23.09	84/02/11 11.23.09	
TP13SS4	2442	84/02/11 11.25.34	84/02/11 11.25.34	

84/04/05 11.16.47. TP13S5 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
TP13CS5	3627	84/02/11 11.29.52	84/02/11 11.29.52	
TP13KS5	3627	84/02/11 11.27.08	84/02/11 11.27.08	
TP13PS5	3627	84/02/11 11.28.29	84/02/11 11.28.29	
TP13SS5	3627	84/02/11 11.31.26	84/02/11 11.31.26	

84/04/05 11.16.47. TP13S7 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
TP13CS7	4650	84/02/11 11.59.56	84/02/11 11.59.56	
TP13KS7	4650	84/02/11 11.56.14	84/02/11 11.56.14	
TP13PS7	4650	84/02/11 11.58.08	84/02/11 11.58.08	
TP13SS7	4650	84/02/11 12.02.21	84/02/11 12.02.21	

84/04/05 11.16.53. TP13S8 CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
TP13CS8	4350	84/02/23 09.22.21	84/02/23 09.22.21	
TP13KS8	4350	84/02/23 09.16.55	84/02/23 09.16.55	
TP13PS8	4350	84/02/23 09.19.17	84/02/23 09.19.17	
TP13SS8	4350	84/02/23 09.25.29	84/02/23 09.25.29	

84/04/06 13.11.58. KBEDSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
BEDKS1	234	84/03/23 07.56.26	84/03/24 07.06.30	
BEDKS2	245	84/03/23 07.58.58	84/03/23 07.58.58	
BEDKS3	272	84/03/23 07.57.45	84/03/23 07.57.45	
BEDKS4	299	84/03/23 08.06.57	84/03/23 08.06.57	
BEDKS5	315	84/03/23 08.05.05	84/03/23 08.05.05	
BEDKS7	404	84/03/23 08.05.46	84/03/23 08.05.46	
BEDKS8	404	84/03/23 08.06.27	84/03/23 08.06.27	
BPDKS1	235	84/03/23 08.48.57	84/03/24 07.29.24	
BPDKS2	246	84/03/23 08.25.24	84/03/23 08.25.24	
BPDKS3	273	84/03/23 08.25.05	84/03/23 08.25.05	
BPDKS4	300	84/03/23 08.24.39	84/03/23 08.24.39	
BPDKS5	316	84/03/23 08.24.09	84/03/23 08.24.09	
BPDKS7	405	84/03/23 08.23.49	84/03/23 08.23.49	
BPDKS8	405	84/03/23 08.22.59	84/03/23 08.22.59	
KEFTBL1	974	84/03/24 18.20.23	84/03/24 18.20.23	
KEFTBL2	1582	84/03/24 18.14.05	84/03/24 18.14.05	
KEPTBL1	974	84/03/24 18.24.56	84/03/24 18.24.56	
KEPTBL2	1582	84/03/24 18.32.28	84/03/24 18.32.28	
KPCTBL1	276	84/03/24 18.20.27	84/03/24 18.20.27	
KPCTBL2	286	84/03/24 18.14.10	84/03/24 18.14.10	
KPPTBL1	276	84/03/24 18.24.58	84/03/24 18.24.58	
KPPTBL2	286	84/03/24 18.32.36	84/03/24 18.32.36	
KPTABLE	716	84/03/25 10.19.30	84/03/25 10.19.30	
KTABLE	716	84/03/25 10.24.10	84/03/25 10.24.10	
RBED	2196	84/03/23 08.40.21	84/04/03 21.22.05	
RBPD	2196	84/03/23 08.40.27	84/04/03 21.22.10	

84/04/06 13.12.00. SBEDSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
BEDSS1	275	84/03/23 07.33.10	84/03/25 09.58.16	
BEDSS2	288	84/03/23 07.34.47	84/03/25 09.52.05	
BEDSS3	312	84/03/23 07.37.28	84/03/25 07.41.44	
BEDSS4	342	84/03/23 07.37.33	84/03/25 07.47.51	
BEDSS5	358	84/03/23 07.55.52	84/03/25 07.54.39	
BEDSS7	444	84/03/23 07.54.58	84/03/25 07.58.32	
BEDSS8	444	84/03/23 07.54.09	84/03/25 08.03.30	
BPDSS1	276	84/03/23 08.43.19	84/03/25 10.01.40	
BPDSS2	289	84/03/23 08.43.07	84/03/25 10.02.04	
BPDSS3	313	84/03/23 08.41.07	84/03/25 08.06.47	
BPDSS4	343	84/03/23 08.40.46	84/03/25 08.07.02	
BPDSS5	359	84/03/23 08.40.22	84/03/25 08.07.21	
BPDSS7	445	84/03/23 08.40.06	84/03/25 08.07.36	
BPDSS8	445	84/03/23 08.39.45	84/03/25 08.14.26	
RBED	2486	84/03/23 08.52.39	84/04/03 21.23.57	
RBPD	2486	84/03/23 08.52.44	84/04/03 22.23.43	
SEFTBL1	357	84/03/25 09.03.05	84/03/25 09.03.05	
SEFTBL2	469	84/03/25 09.05.36	84/03/25 10.08.50	
SEFTBL3	593	84/03/25 09.53.01	84/03/25 10.02.50	
SEPTBL1	357	84/03/25 09.10.24	84/03/25 09.10.24	
SEPTBL2	469	84/03/25 09.13.10	84/03/25 10.09.26	
SEPTBL3	593	84/03/25 10.07.11	84/03/25 10.07.11	
SPCTBL1	81	84/03/25 09.03.06	84/03/25 09.03.06	
SPCTBL2	81	84/03/25 09.05.37	84/03/25 10.08.52	
SPCTBL3	81	84/03/25 09.53.03	84/03/25 10.02.52	
SPPTBL1	81	84/03/25 09.10.25	84/03/25 09.10.25	
SPPTBL2	81	84/03/25 09.13.11	84/03/25 10.09.28	
SPPTBL3	81	84/03/25 10.07.13	84/03/25 10.07.13	
SPTABLE	307	84/03/25 10.37.24	84/03/25 10.37.24	
STABLE	307	84/03/25 10.36.18	84/03/25 10.36.18	

84/04/06 13.12.04. PBEDSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
BEDPS1	234	84/03/23 07.06.34	84/03/24 07.07.25	
BEDPS2	245	84/03/23 07.10.50	84/03/23 07.10.50	
BEDPS3	272	84/03/23 07.11.55	84/03/23 07.11.55	
BEDPS4	299	84/03/23 07.14.12	84/03/23 07.14.12	
BEDPS5	315	84/03/23 07.14.58	84/03/23 07.14.58	
BEDPS7	404	84/03/23 07.16.23	84/03/23 07.16.23	
BEDPS8	404	84/03/23 07.17.40	84/03/23 07.17.40	
BPDPS1	235	84/03/23 08.33.09	84/03/24 07.30.07	
BPDPS2	246	84/03/23 08.32.34	84/03/23 08.32.34	
BPDPS3	273	84/03/23 08.33.27	84/03/23 08.33.27	
BPDPS4	300	84/03/23 08.33.42	84/03/23 08.33.42	
BPDPS5	316	84/03/23 08.32.53	84/03/23 08.32.53	
BPDPS7	405	84/03/23 08.46.29	84/03/23 08.46.29	
BPDPS8	405	84/03/23 08.34.05	84/03/23 08.34.05	
PEFTBL1	1000	84/03/24 18.19.30	84/03/24 18.19.30	
PEFTBL2	1513	84/03/24 18.26.26	84/03/24 18.26.26	
PEPTBL1	1000	84/03/24 18.32.04	84/03/24 18.32.04	
PEPTBL2	1513	84/03/24 18.33.33	84/03/24 18.33.33	
PPCTBL1	237	84/03/24 18.19.33	84/03/24 18.19.33	
PPCTBL2	245	84/03/24 18.26.30	84/03/24 18.26.30	
PPPTBL1	237	84/03/24 18.32.08	84/03/24 18.32.08	
PPPTBL2	245	84/03/24 18.33.36	84/03/24 18.33.36	
PPTABLE	612	84/03/25 10.26.36	84/03/25 10.30.27	
PTABLE	612	84/03/25 10.23.53	84/03/25 10.31.06	
RBED	2196	84/03/23 08.51.41	84/04/03 21.23.01	
RBPD	2196	84/03/23 08.51.45	84/04/03 21.23.07	

84/04/06 13.12.06. CBEDSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
BEDCS1	315	84/03/23 07.21.53	84/03/24 06.59.10	
BEDCS2	326	84/03/23 07.09.43	84/03/23 07.09.43	
BEDCS3	272	84/03/23 07.24.05	84/03/23 07.24.05	
BEDCS4	299	84/03/23 07.25.15	84/03/23 07.25.15	
BEDCS5	396	84/03/23 07.26.15	84/03/24 16.54.22	
BEDCS7	404	84/03/23 07.27.50	84/03/23 07.27.50	
BEDCS8	404	84/03/21 10.34.17	84/03/23 07.31.26	
BPDCS1	316	84/03/23 08.32.53	84/03/24 07.31.21	
BPDCS2	327	84/03/23 08.33.12	84/03/23 08.33.12	
BPDCS3	273	84/03/23 08.33.30	84/03/23 08.33.30	
BPDCS4	300	84/03/23 08.34.04	84/03/23 08.34.04	
BPDCS5	397	84/03/23 08.34.21	84/03/24 17.01.24	
BPDCS7	405	84/03/23 08.35.26	84/03/23 08.35.26	
BPDCS8	405	84/03/22 10.07.42	84/03/22 10.07.42	
CEFTBL1	864	84/03/24 17.13.16	84/03/24 17.13.16	
CEFTBL2	1287	84/03/24 18.14.59	84/03/24 18.14.59	
CEPTBL1	864	84/03/24 18.25.47	84/03/24 18.25.47	
CEPTBL2	1287	84/03/24 18.24.48	84/03/24 18.24.48	
CPCTBL1	197	84/03/24 17.13.18	84/03/24 17.13.18	
CPCTBL2	204	84/03/24 18.15.03	84/03/24 18.15.03	
CPPTBL1	197	84/03/24 18.25.50	84/03/24 18.25.50	
CPPTBL2	204	84/03/24 18.24.53	84/03/24 18.24.53	
CPTABLE	510	84/03/25 10.32.12	84/03/25 10.32.12	
CTABLE	510	84/03/25 10.30.00	84/03/25 10.30.00	
RBED	2439	84/03/23 08.46.07	84/04/03 21.20.52	
RBPD	2439	84/03/23 08.46.12	84/04/03 21.20.58	

84/04/07 00.22.12. IFG4SET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
S1AIFG4	17	83/09/19 16.43.52	84/03/02 20.25.13	
S1BIFG4	17	83/09/19 17.13.36	84/03/02 20.25.34	
S1CIFG4	17	83/09/19 17.15.28	84/03/04 12.51.01	
S2AIFG4	13	83/09/19 17.17.01	84/03/02 20.26.12	
S2BIFG4	13	83/09/19 17.18.09	84/03/02 20.26.29	
S2CIFG4	13	83/09/19 17.19.40	84/03/04 12.59.12	
S2DIFG4	12	84/03/04 13.00.10	84/03/04 13.00.10	
S3AIFG4	19	83/09/19 17.22.01	84/03/02 20.27.14	
S3BIFG4	19	83/09/19 17.23.28	84/03/02 20.27.31	
S3CIFG4	19	83/09/19 17.25.21	84/03/02 20.27.50	
S3DIFG4	19	83/10/04 15.31.35	84/03/04 13.07.31	
S4AIFG4	15	83/09/19 17.26.45	84/03/02 20.28.52	
S4BIFG4	15	83/09/19 17.27.52	84/03/02 20.29.14	
S4CIFG4	15	83/09/19 17.29.01	84/03/02 20.29.32	
S4DIFG4	15	83/09/19 17.30.11	84/03/04 14.22.05	
S5AIFG4	18	83/09/19 17.31.30	84/03/02 20.30.51	
S5BIFG4	18	83/09/19 17.32.36	84/03/02 20.31.06	
S5CIFG4	18	83/09/19 17.34.02	84/03/02 20.31.18	
S5DIFG4	18	83/09/19 17.35.14	84/03/02 20.31.40	
S7AIFG4	19	83/09/19 17.37.14	84/03/02 20.31.59	
S7BIFG4	19	83/09/19 17.38.16	84/03/02 20.32.11	
S7CIFG4	19	83/09/19 17.40.03	84/03/02 20.32.27	
S7DIFG4	19	83/09/19 17.46.36	84/03/02 20.32.41	
S7EIFG4	19	83/09/19 17.48.22	84/03/02 20.32.54	
S8AIFG4	19	83/09/19 17.49.58	84/03/02 20.33.14	
S8BIFG4	19	83/09/19 17.51.01	84/03/02 20.33.27	
S8CIFG4	19	83/09/19 17.52.07	84/03/02 20.33.46	
S8DIFG4	19	83/09/19 17.53.44	84/03/02 20.34.18	
S8EIFG4	19	83/09/19 17.56.06	84/03/02 20.34.31	

84/04/07 00.22.12. SORCSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
CI	1	83/11/14 14.08.44	83/11/14 14.08.44	
CS	1	83/11/14 14.08.37	83/11/14 14.08.37	
KI	1	83/11/14 14.08.28	84/02/15 20.56.10	
KPCS	1	84/01/03 16.07.25	84/01/03 16.07.25	
KPCSNOS	1	84/01/03 16.00.12	84/01/03 16.00.12	
KS	1	83/11/14 14.08.13	83/11/14 14.08.13	
PI	1	83/11/14 14.09.01	83/11/14 14.09.01	
PS	1	83/11/14 14.08.54	83/11/14 14.08.54	
SI	1	83/11/14 14.09.26	83/11/14 14.09.26	
SS	1	83/11/14 14.09.12	83/11/14 14.09.12	
REACH1/CNOS	1	84/01/03 15.31.41	84/01/03 15.31.41	
REACH1/KNOS	1	84/01/03 15.19.43	84/01/03 15.19.43	
REACH1/KPCS	1	84/01/03 16.08.09	84/01/03 16.08.09	
REACH1/KPCSNOS	1	84/01/03 15.47.33	84/01/03 16.09.12	
REACH1/PNOS	1	84/01/03 15.29.15	84/01/03 15.29.15	
REACH1/SNOS	1	84/01/03 15.39.38	84/01/03 15.39.38	
REACH2/CNOS	1	84/01/03 15.34.17	84/01/03 15.34.17	
REACH2/KNOS	1	84/01/03 15.22.35	84/01/03 15.22.35	
REACH2/KPCSNOS	1	84/01/03 15.45.17	84/01/03 16.09.39	
REACH2/PNOS	1	84/01/03 15.28.28	84/01/03 15.28.28	
REACH2/SNOS	1	84/01/03 15.40.53	84/01/03 15.40.53	
REACH3/CNOS	1	84/01/03 15.31.59	84/01/03 15.31.59	
REACH3/KNOS	1	84/01/03 15.20.24	84/01/03 15.20.24	
REACH3/KPCSNOS	1	84/01/03 15.58.16	84/01/03 16.10.08	
REACH3/KPSCNOS	1	84/01/03 15.48.03	84/01/03 15.48.03	
REACH3/PNOS	1	84/01/03 15.29.34	84/01/03 15.29.34	
REACH3/SNOS	1	84/01/03 15.40.04	84/01/03 15.40.04	
REACH4/CNOS	1	84/01/03 15.35.18	84/01/03 15.35.18	
REACH4/KNOS	1	84/01/03 15.23.03	84/01/03 15.23.03	
REACH4/KPCSNOS	1	84/01/03 15.45.44	84/01/03 16.10.34	
REACH4/PNOS	1	84/01/03 15.28.06	84/01/03 15.28.06	
REACH4/SNOS	1	84/01/03 15.41.09	84/01/03 15.41.09	
REACH5/CNOS	1	84/01/03 15.36.24	84/01/03 15.36.24	
REACH5/KNOS	1	84/01/03 15.24.20	84/01/03 15.24.20	
REACH5/KPCSNOS	1	84/01/03 15.49.16	84/01/03 16.11.11	
REACH5/PNOS	1	84/01/03 15.26.07	84/01/03 15.26.07	
REACH5/SNOS	1	84/01/03 15.38.56	84/01/03 15.38.56	
REACH7/CNOS	1	84/01/03 15.35.34	84/01/03 15.35.34	
REACH7/KNOS	1	84/01/03 15.23.21	84/01/03 15.23.21	
REACH7/KPCSNOS	1	84/01/03 15.46.32	84/01/03 16.11.39	
REACH7/PNOS	1	84/01/03 15.27.35	84/01/03 15.27.35	
REACH7/SNOS	1	84/01/03 15.41.22	84/01/03 15.41.22	
REACH8/CNOS	1	84/01/03 15.37.04	84/01/03 15.37.04	
REACH8/KNOS	1	84/01/03 15.24.50	84/01/03 15.24.50	
REACH8/KPCSNOS	1	84/01/03 15.49.39	84/01/03 16.12.22	
REACH8/PNOS	1	84/01/03 15.26.26	84/01/03 15.26.26	
REACH8/SNOS	1	84/01/03 15.38.39	84/01/03 15.38.39	

84/04/07 00.22.11. HYDRSET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
JOB1A	1	84/01/04 16.02.39	84/01/04 16.02.39	
TP12B	14	84/02/22 10.54.11	84/02/22 10.54.11	
TP12C	14	84/02/22 10.54.40	84/02/22 10.54.40	
REACH1/TAPE11	25	84/01/04 16.10.52	84/01/04 16.54.06	
REACH1/TAPE12	23	84/01/04 16.54.34	84/02/27 11.15.49	
REACH1/TAPE3	10	84/01/04 16.10.53	84/01/05 08.12.47	
REACH1/TP4A	72	84/01/04 16.10.51	84/01/04 16.54.05	
REACH1/TP4B	86	84/01/04 16.11.20	84/01/04 16.54.20	
REACH1/TP4C	81	84/01/04 16.11.37	84/01/04 16.54.33	
REACH1/VAFS	8	84/03/02 14.53.40	84/03/05 09.11.06	
REACH1/VCE	3	84/02/26 17.31.16	84/02/26 18.55.04	
REACH2/TAPE11	19	84/01/04 16.12.06	84/01/05 07.23.28	
REACH2/TAPE12	25	84/01/05 07.23.56	84/02/27 11.15.08	
REACH2/TAPE3	7	84/01/05 07.23.56	84/01/05 08.18.52	
REACH2/TP4A	55	84/01/04 16.12.05	84/01/05 07.23.28	
REACH2/TP4B	63	84/01/04 16.12.32	84/01/05 07.23.38	
REACH2/TP4C	66	84/01/04 16.12.57	84/01/05 07.23.48	
REACH2/TP4D	3	84/01/04 16.13.14	84/01/05 07.23.56	
REACH2/VAFS	8	84/03/02 14.51.29	84/03/05 09.31.50	
REACH2/VCE	3	84/02/26 17.31.23	84/02/26 18.46.03	
REACH3/TAPE11	31	84/01/05 07.25.35	84/01/05 07.39.20	
REACH3/TAPE12	41	84/01/05 07.39.52	84/02/27 11.14.34	
REACH3/TAPE3	11	84/01/05 07.39.53	84/01/05 08.25.55	
REACH3/TP4A	88	84/01/05 07.25.34	84/01/05 07.39.19	
REACH3/TP4B	103	84/01/05 07.25.47	84/01/05 07.39.32	
REACH3/TP4C	106	84/01/05 07.39.43	84/01/05 07.39.43	
REACH3/TP4D	39	84/01/05 07.39.51	84/01/05 07.39.51	
REACH3/VAFS	11	84/03/02 14.48.40	84/03/05 10.00.24	
REACH3/VCE	4	84/02/26 17.31.30	84/02/26 18.37.34	
REACH4/TAPE11	23	84/01/04 16.13.56	84/01/05 07.27.37	
REACH4/TAPE12	30	84/01/05 07.28.08	84/02/27 11.13.54	
REACH4/TAPE3	9	84/01/05 07.49.43	84/01/05 08.30.09	
REACH4/TP4A	70	84/01/04 16.13.56	84/01/05 07.27.37	
REACH4/TP4B	80	84/01/04 16.14.19	84/01/05 07.27.48	
REACH4/TP4C	81	84/01/04 16.14.41	84/01/05 07.27.58	
REACH4/TP4D	57	84/01/04 16.15.01	84/01/05 07.28.07	
REACH4/VAFS	10	84/03/02 14.47.15	84/03/05 12.17.08	
REACH4/VCE	3	84/02/26 17.31.37	84/02/26 18.25.50	
REACH5/TAPE11	28	84/01/05 07.29.55	84/01/05 07.29.55	
REACH5/TAPE12	47	84/01/05 07.30.29	84/02/27 19.07.11	
REACH5/TAPE3	11	84/01/05 07.30.29	84/01/05 09.39.01	
REACH5/TP4A	88	84/01/05 07.29.54	84/01/05 07.29.54	
REACH5/TP4B	100	84/01/05 07.30.06	84/01/05 07.30.06	
REACH5/TP4C	102	84/01/05 07.30.17	84/01/05 07.30.17	
REACH5/TP4D	96	84/01/05 07.30.28	84/01/05 07.30.28	
REACH5/VAFS	12	84/03/02 14.46.30	84/03/05 12.42.50	
REACH5/VCE	4	84/02/26 17.31.43	84/02/26 18.11.04	
REACH7/TAPE11	31	84/01/04 16.15.37	84/01/05 07.31.58	
REACH7/TAPE12	50	84/01/05 07.32.56	84/02/27 11.10.39	
REACH7/TAPE3	11	84/01/05 07.32.56	84/01/05 08.45.34	
REACH7/TP4A	91	84/01/04 16.15.37	84/01/05 07.31.58	
REACH7/TP4B	101	84/01/04 16.15.57	84/01/05 07.32.12	
REACH7/TP4C	102	84/01/04 16.16.14	84/01/05 07.32.32	
REACH7/TP4D	104	84/01/04 16.16.34	84/01/05 07.32.44	
REACH7/TP4E	105	84/01/04 16.16.53	84/01/05 07.32.55	
REACH7/VAFS	15	84/03/02 14.41.32	84/03/05 13.02.26	
REACH7/VCE	3	84/02/26 17.31.52	84/02/26 18.01.39	
REACH8/TAPE11	30	84/01/05 07.34.17	84/01/05 07.34.17	
REACH8/TAPE12	50	84/01/05 07.35.08	84/02/27 19.06.25	

REACH8/TP12C	10	84/02/23	08.47.01	84/02/23	08.47.01
REACH8/TP4A	84	84/01/05	07.34.16	84/01/05	07.34.16
REACH8/TP4B	97	84/01/05	07.34.28	84/02/23	08.46.46
REACH8/TP4C	101	84/01/05	07.34.40	84/02/23	08.47.01
REACH8/TP4D	102	84/01/05	07.34.52	84/01/05	07.34.52
REACH8/TP4E	103	84/01/05	07.35.05	84/01/05	07.35.05
REACH8/VAFS	15	84/02/27	19.12.22	84/03/05	13.03.41
REACH8/VCE	3	84/02/26	17.32.00	84/02/26	17.53.42

HYDRSET p. 2

84/04/11 16.28.06. K13SET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
KEFTBL1	1048	84/03/18 21.05.37	84/03/18 21.05.37	
KEFTBL2	1691	84/03/18 21.08.52	84/03/18 21.08.52	
KEPTBL1	1048	84/03/18 21.11.25	84/03/21 07.32.29	
KEPTBL2	1691	84/03/18 21.10.24	84/03/21 07.48.50	
KPCTBL1	276	84/03/18 21.05.39	84/03/18 21.05.39	
KPCTBL2	286	84/03/18 21.08.56	84/03/18 21.08.56	
KPPTBL1	276	84/03/18 21.11.28	84/03/21 07.32.33	
KPPTBL2	286	84/03/18 21.10.27	84/03/21 07.48.54	
KPTABLE	716	84/03/21 21.47.40	84/03/21 21.47.40	
KTABLE	716	84/03/21 07.49.18	84/03/21 07.49.18	
RWP3	2182	84/03/18 19.47.35	84/03/18 19.47.35	
RW13	2240	84/03/18 19.47.30	84/03/18 19.47.30	
WP3KS1	235	84/03/18 19.47.22	84/03/18 19.47.22	
WP3KS2	246	84/03/18 19.47.23	84/03/18 19.47.23	
WP3KS3	273	84/03/18 19.47.24	84/03/18 19.47.24	
WP3KS4	300	84/03/18 19.47.25	84/03/18 19.47.25	
WP3KS5	316	84/03/18 19.47.26	84/03/18 19.47.26	
WP3KS7	405	84/03/18 19.47.27	84/03/18 19.47.27	
WP3KS8	405	84/03/18 19.47.29	84/03/18 19.47.29	
W13KS1	248	84/03/18 19.47.15	84/03/18 19.47.15	
W13KS2	259	84/03/18 19.47.16	84/03/18 19.47.16	
W13KS3	288	84/03/18 19.47.17	84/03/18 19.47.17	
W13KS4	299	84/03/18 19.47.17	84/03/18 19.47.17	
W13KS5	315	84/03/18 19.47.19	84/03/18 19.47.19	
W13KS7	404	84/03/18 19.47.20	84/03/18 19.47.20	
W13KS8	404	84/03/18 19.47.21	84/03/18 19.47.21	

26 ELEMENTS, 94.8 PERCENT OF CURRENT FILESET SPACE
(NE= 250, DIRECTORY 11.6 PERCENT FULL)

84/04/11 16.28.07. S13SET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
RWP3	2182	84/03/18 13.32.09	84/03/18 19.25.19	
RW13	2224	84/03/18 13.32.05	84/03/18 19.25.11	
SEFTBL1	580	84/03/18 21.20.01	84/03/18 21.20.01	
SEFTBL2	756	84/03/18 21.21.01	84/03/18 21.21.01	
SEPTBL1	580	84/03/18 21.22.51	84/03/21 07.41.43	
SEPTBL2	756	84/03/18 21.23.40	84/03/21 07.46.29	
SPCTBL1	119	84/03/18 21.20.03	84/03/18 21.20.03	
SPCTBL2	123	84/03/18 21.21.03	84/03/18 21.21.03	
SPPTBL1	119	84/03/18 21.22.53	84/03/21 07.41.46	
SPPTBL2	123	84/03/18 21.23.42	84/03/21 07.46.32	
SPTABLE	307	84/03/21 22.07.51	84/03/21 22.07.51	
STABLE	307	84/03/21 22.06.51	84/03/21 22.06.51	
WP3SS1	235	84/03/11 17.24.08	84/03/11 17.24.08	
WP3SS2	246	84/03/11 17.23.48	84/03/11 17.23.48	
WP3SS3	273	84/03/18 13.28.22	84/03/18 13.28.22	
WP3SS4	300	84/03/11 17.23.03	84/03/11 17.23.03	
WP3SS5	316	84/03/11 17.22.43	84/03/18 12.13.04	
WP3SS7	405	84/03/11 17.22.18	84/03/18 12.14.04	
WP3SS8	405	84/03/11 17.21.36	84/03/18 12.14.28	
W13SS1	248	84/02/18 16.22.54	84/03/08 11.52.48	
W13SS2	259	84/02/18 16.23.30	84/03/02 09.41.37	
W13SS3	272	84/02/18 16.24.20	84/03/18 13.25.06	
W13SS4	299	84/02/18 16.25.15	84/02/23 10.22.26	
W13SS5	315	84/02/18 16.25.50	84/03/18 11.56.05	
W13SS7	404	84/02/18 16.26.25	84/03/18 11.58.19	
W13SS8	404	84/02/18 16.27.01	84/03/18 11.59.58	

26 ELEMENTS, 75.5 PERCENT OF CURRENT FILESET SPACE
(NE= 250, DIRECTORY 14.4 PERCENT FULL)

84/04/11 16.28.09. P13SET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
A	1	84/03/18 19.31.45	84/03/18 19.31.45	
PEFTBL1	1024	84/03/18 20.30.51	84/03/18 20.30.51	
PEFTBL2	1511	84/03/18 20.35.31	84/03/18 20.35.31	
PEPTBL1	1024	84/03/18 20.41.38	84/03/21 07.45.54	
PEPTBL2	1511	84/03/18 20.52.24	84/03/21 07.51.25	
PPCTBL1	237	84/03/18 20.30.53	84/03/18 20.30.53	
PPCTBL2	245	84/03/18 20.35.34	84/03/18 20.35.34	
PPPTBL1	237	84/03/18 20.41.40	84/03/21 07.45.57	
PPPTBL2	245	84/03/18 20.52.28	84/03/21 07.51.29	
PPTABLE	612	84/03/21 22.04.13	84/03/21 22.09.41	
PTABLE	612	84/03/21 22.05.14	84/03/21 22.10.38	
RWP3	2223	84/03/18 19.38.40	84/03/18 19.38.40	
RW13	2267	84/03/18 19.38.33	84/03/18 19.38.33	
WP3PS1	276	84/03/18 19.33.25	84/03/18 19.33.25	
WP3PS2	246	84/03/18 19.33.26	84/03/18 19.33.26	
WP3PS3	273	84/03/18 19.33.27	84/03/18 19.33.27	
WP3PS4	300	84/03/18 19.33.27	84/03/18 19.33.27	
WP3PS5	316	84/03/18 19.33.28	84/03/18 19.33.28	
WP3PS7	405	84/03/18 19.33.29	84/03/18 19.33.29	
WP3PS8	405	84/03/18 19.33.31	84/03/18 19.33.31	
W13PS1	275	84/03/18 19.33.18	84/03/18 19.33.18	
W13PS2	259	84/03/18 19.33.19	84/03/18 19.33.19	
W13PS3	288	84/03/18 19.33.19	84/03/18 19.33.19	
W13PS4	299	84/03/18 19.33.20	84/03/18 19.33.20	
W13PS5	315	84/03/18 19.33.21	84/03/18 19.33.21	
W13PS7	404	84/03/18 19.33.22	84/03/18 19.33.22	
W13PS8	404	84/03/18 19.33.23	84/03/18 19.33.23	

27 ELEMENTS, 98.3 PERCENT OF CURRENT FILESET SPACE
(NE= 250, DIRECTORY 11.6 PERCENT FULL)

84/04/11 16.28.10. C13SET CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
CEFTBL1	583	84/03/30 19.56.57	84/03/30 19.56.57	
CEFTBL2	844	84/03/30 19.57.08	84/03/30 19.57.08	
CEFTBL3	1154	84/03/31 07.08.29	84/03/31 07.08.29	
CEPTBL1	583	84/03/30 19.57.24	84/03/30 19.57.24	
CEPTBL2	790	84/03/30 19.58.19	84/03/30 19.58.19	
CEPTBL3	1154	84/03/31 07.12.40	84/03/31 07.12.40	
CPCTBL1	134	84/03/30 19.58.44	84/03/30 19.58.44	
CPCTBL2	134	84/03/30 19.59.03	84/03/30 19.59.03	
CPCTBL3	134	84/03/31 07.08.32	84/03/31 07.08.32	
CPPTBL1	134	84/03/30 19.59.11	84/03/30 19.59.11	
CPPTBL2	134	84/03/30 19.59.22	84/03/30 19.59.22	
CPPTBL3	134	84/03/31 07.12.43	84/03/31 07.12.43	
CPTABLE	510	84/03/31 07.28.17	84/03/31 10.03.41	
CTABLE	510	84/03/31 07.27.34	84/03/31 10.02.58	
JOB	2	84/03/30 19.59.32	84/03/30 19.59.32	
WP3CS1	343	84/03/30 19.54.04	84/03/31 07.03.49	
WP3CS2	348	84/03/30 19.54.12	84/03/31 07.04.36	
WP3CS3	367	84/03/30 19.54.23	84/03/30 19.54.23	
WP3CS4	394	84/03/30 19.54.32	84/03/30 19.54.32	
WP3CS5	397	84/03/30 19.54.40	84/03/30 19.54.40	
WP3CS7	485	84/03/30 19.54.51	84/03/30 19.54.51	
WP3CS8	488	84/03/30 19.55.01	84/03/30 19.55.01	
W13CS1	362	84/03/30 19.55.11	84/03/30 20.06.02	
W13CS2	367	84/03/30 19.55.18	84/03/31 06.58.28	
W13CS3	387	84/03/30 19.55.25	84/03/31 06.59.21	
W13CS4	393	84/03/30 19.55.32	84/03/31 07.00.18	
W13CS5	396	84/03/30 19.55.41	84/03/31 07.01.07	
W13CS7	484	84/03/30 19.55.52	84/03/30 19.55.52	
W13CS8	487	84/03/30 19.56.11	84/03/30 19.56.11	

29 ELEMENTS, 97.0 PERCENT OF CURRENT FILESET SPACE
(NE= 250, DIRECTORY 12.8 PERCENT FULL)

84/04/06 13.12.08. INCUBFS CONTAINS

GROUP/ELEMENT	SIZE	ADDED	LAST MODIFIED	COMMENTS
INCUB	28	84/02/07 11.08.51	84/02/07 22.45.09	
PROCFIL	2	84/02/07 10.14.24	84/02/07 10.14.24	
QUERY	15	84/02/02 13.29.55	84/02/07 22.20.48	INTERACTIVE "FRONT-END" FOR INCUB

PAGE 1 84/04/06.
RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM, IFG4.

1	00B	7663	7135B	*******	S1IFG4	SKAGIT RIVER ISLAND REACH IFG DATA SET FOR 3 FLOWS AND 4 XSECTIONS Q MEASURED
2	17B	0	0000B			
		7663	7135B			

TOTAL

7663	7135B

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

```
FILE   1 ITEM,PROCFIL.
     1  008      38   4271B  *******   .PROC,RNMOD,F,FMOD.          *      F=WSP DATA DECK BEFORE N MODIFICATION      *      FMO
     2  008      31   7152B  *******   .PROC,RHABIN,F.     *      F=FILE CREATED AS INPUT FILE FOR HABTAT          NOTE(OUTPUT,N
     3  008      25   2027B  *******   .PROC,RCORDIN,F.    *      F=FILE OF COORDINATE POINTS FOR EITHER WSP OR IFG4   IFE,.NOT.F
     4  008      38   5215B  *******   .PROC,RQMOD,F,FMOD.          *      F=WSP FILE BEFORE Q MODIFICATION           *      FMO
     5  008      23   5643B  *******   .PROC,RWPQARD,F.    *      F=FILE OF PARD AND QARD CARDS    IFE,.NOT.FILE(PHABSIM,MS),CDPT
     6  008      39   7340B  *******   .PROC,RHABINS,F.    *      F=INPUT FILE CREATED FOR HABTAT USING IFG'S ≠FISHFIL≠
     7  008      22   3521B  *******   .PROC,RIFG4IN,F.    *      F=FILE CREATED          IFE,.NOT.FILE(PHABSIM,MS),CDPT2.
     8  008      41   3542B  *******   .PROC,RADDQ,F,FMOD.          *      F=ORIGINAL IFG4 SOURCE FILE      *      FMOD=IFG4 SOU
     9  008      37   1734B  *******   .PROC,RADDSZF,F,FMOD.        *      F=ORIGINAL IFG4 SOURCE FILE      *      FMOD=IFG4 SOU
    10  008      26   2412B  *******   .PROC,RROUGH,F.     *      F=FILE CONSISTING OF ROUGHNESS VALUES FOR WSP DATA DECK
    11  008      41   0213B  *******   .PROC,RCRVFIL,F,FMOD.        *      F=CARD IMAGE FILE      *      FMOD=UNFORMATTED FISHFI
    12  008      43   4061B  *******   .PROC,RLDIR,F,DIRCTY.        *      F= ≠FISHFIL≠ TYPE FILE,DIRCTY=DIRECTORY    RETURN,DIR
    13  008      45   4226B  *******   .PROC,RSTART,S,TAPE3,TP4.    *      S=FILE WITH BOTH CROSS SECTION   *      DATA AND HY
    14  008      60   1657B  *******   .PROC,RWSP,INT,OUT.          *      INT=COMPLETE WSP DATA FILE       *      OUT=WSP OUTPU
    15  008      29   4215B  *******   .PROC,FOK,F.       *      F=FILE TO BE PRINTED AT TERMINAL          IFE,.NOT.FILE(F,MS),
    16  008      33   5377B  *******   .PROC,RSUBMDC,I,TAPE3,TP3.    REWIND,TAPE3,TP3.   IFE,.NOT.FILE(I,MS),CDPT2.    GET,I.
    17  008      69   4561B  *******   .PROC,RHBIF4,IH,IS,OUT,FISHFIL,TAPE8.   *      IH=IFG4 INPUT              *      IS=HABTAT INP
    18  008      61   7317B  *******   .PROC,RWSPC,INT,OUT.         *      INT= WSP DATA FILE    *      OUT= WSP RESULTS
    19  008      89   4505B  *******   .PROC,RHBWSP,IH,IS,OUT,FISHFIL.      *      IH=WSP INPUT, IS=HABTAT INPUT    *      IF
    20  008      53   6747B  *******   .PROC,RHABPLT,TAPE7,TAPE2.   *      TAPE7=SOURCE OF DATA TO BE PLOTTED.        *      NOT
    21  008      31   6462B  *******   .PROC,RSTORE,S,TAPE3,TP4.    *      S=FILE WITH BOTH CROSS SECTION   *      DATA AND HY
    22  008      45   1757B  *******   .PROC,RSUBMD,TAPE3,TP4.      *      THIS PROCEDURE ALLOWS USER TO    *      ADD SUBSTRATE
    23  008      37   2245B  *******   .PROC,RWSPPLT,INT,OUT.       *      INT=WSP OF IFG4 DATA DECK        *      OUT=OUTPUT OF
    24  008      70   6055B  *******   .PROC,RHABTAT,INT,OUT,FISHFIL,TAPE8,TAPE3,TP4,TAPE7.       *      INT=INPUT FILE PRODUCED
    25  008      22   5176B  *******   .PROC,RGENFIL,F.    *      F=FILE CREATED          IFE,.NOT.FILE(PHABSIM,MS),CDPT2.
    26  008      39   5640B  *******   .PROC,RCWRT,F,FO.   *     F=FISHFIL TYPE FILE     *      FO=CARD IMAGE FILE         IFE,.NOT.F
    27  008      26   2721B  *******   .PROC,RENTSSD,F.    *      F=FILE CREATED OF SUBSTRATE DATA           IFE,.NOT.FILE(PHABSI
    28  008      22   5640B  *******   .PROC,RGCURV,F.     *      F=FILE AT CARD IMAGE CURVE SETS  IFE,.NOT.FILE(F,MS),CDPT2.
    29  008      26   2350B  *******   .PROC,RGI4CV,F.     *      F=FILE CREATED FOR IFG4 DECK THAT          *      CONTAINS VELO
    30  008      31   7322B  *******   .PROC,RGI4NS,F.     *      F=FILE CREATED FOR IFG4 DECK THAT          *      CONTAINS MANN
    31  008      30   0136B  *******   .PROC,RGI4QD,F.     *      F=FILE CREATED FOR IFG4 DECK THAT          *      CONTAINS THE
    32  008      38   5024B  *******   .PROC,RGI4XS,F,FMOD.          *      F= COORDINATE DATA FILE          *      FMOD=MODIFIED
    33  008      45   4202B  *******   .PROC,RCKI4,INT,OUT.         *      INT=IFG4 DATA FILE    *      OUT=OUTPUT FILE WITH AN
    34  008      32   2223B  *******   .PROC,RIFG1A,INT,OUT.        *      INT=INPUT FILE, O=OUTPUT FILE    IFE,.NOT.FILE(INT,MS
    35  008      52   0131B  *******   .PROC,RIFG4,INT,OUT,TAPE3,TP4.       *      INT=COMPLETE IFG4 DATA DECK      *      OUT
    36  008     103   4750B  *******   .PROC,RHBWSPS,IH,IS,ISS,FISHFIL,OUT,TAPE8.   *      IH=WSP INPUT, IS=HABTAT INPUT
    37  008      37   1515B  *******   .PROC,RAVDPTH,INT,OUT.       *      INT=SOURCE(DIRECT INPUT OF W.S. ELEV.)     *      OUT
    38  008      45   5244B  *******   .PROC,RIFG4V,INT,OUT.        *      INT=SOUTUTURCE, O=RESULTS          IFE,.NOT.FILE(PHABSI
    39  008      70   4446B  *******   .PROC,RWSPO,INT,OUT.         *      INT=WSP INPUT FILE    *      OUT=WSP RESULTS USED FO
    40  008      45   6135B  *******   .PROC,RHABTAD,INT,OUT,FISHFIL.        *      INT=SOURCE(DIRECT INPUT OF HABITAT DATA)
    41  008      31   0277B  *******   .PROC,RHYSCF,TP3,TP4,TP1.    *     TP3=CROSS SECTION DATA  *     TP4=FLOW RELATED DATA
    42  008      34   4330B  *******   .PROC,RHYSCF1,TP3,TDUM,TP1.  *     TP3=CROSS SECTION DATA  *     TDUM=AN EMPTY FILE
    43  008      32   0034B  *******   .PROC,RHYSBF,TP1,TP3,TP4.    *     TP1=CARD IMAGE FILE     *     TP3=CROSS SECTION DATA
    44  008      34   4253B  *******   .PROC,RHYSBF1,TP1,TP3,TDUM.  *     TP1=CARD IMAGE FILE     *     TP3=CROSS SECTION DATA
    45  008      40   6750B  *******   .PROC,RHYSCF2,TP4,TP1.       *     TP4=FILE OF FLOW RELATED DATA     *     TP1=FILE IN CA
    46  008      37   7106B  *******   .PROC,RHYSBF2,TP4,TP1,TDUM.  *     TP4=FILE OF FLOW RELATED DATA     *     TP1=CARD IMAGE
    47  008      24   6473B  *******   .PROC,PHABNEW.      *GIVES UPDATES TO THE PHABSIM SYSTEM    *AND LISTS IMPORTANT MESSAGES
    48  008      66   5154B  *******   .PROC,RSPLTP8,F8,F16.         *         *     RUNS THE SPLITP8 PROGRAM           *      WHIC
    49  008      61   1227B  *******   .PROC,RLSTVD,TAPE3,TAPE4,OUTPUT.      *         *         *     RUNS THE LSTVD PROGRAM  *      WHIC
    50  008      44   2276B  *******   .PROC,RWMOD,TP3,TP3A,INPUT,OUTPUT.    *         *         *   RUNS THE WMOD PROGRAM     *    WHICH
    51  008      71   4351B  *******   .PROC,RADIN14,F,GDAVD,TAPE4.  *         *  PREPARES AN INPUT FILE USING              *   A TAPE4
    52  008      56   7172B  *******   .PROC,RADINWP,F,GDAVD,TAPE4.  *         *  MAKES A FILE FOR THE AVDEPTH              *   PROGRAM
    53  008       0   0000B
    54  008      64   6456B  *******   .PROC,RWSPS,INT,OUT,FNAME.    *      INT=WSP INPUT FILE     *      OUT=WSP RESULTS USED FO
    55  008      67   0061B  *******   .PROC,RAVDEPT,INPT,OUT,TAPE8.         *         *     RUNS THE AVDEPTH PROGRAM.
    56  008      65   6337B  *******   .PROC,RAVDTP4,IWSP,IAVDPT,TAPE4.      *         *       RUNS THE AVDTP4 PROGRAM
    57  008      53   4202B  *******   .PROC,RDIK14,IN14,TP3,TP4.    *         *   RUNS THE DIK14 PROGRAM     *   WHICH CREATES TA
```

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

```
FILE   1 ITEM,PROCFIL.
    58  008       34    3637B  *******    .PROC,R8NOTE,TAPE8,TP8.        *  PROC TO ADD A NOTE TO THE  *  TAPE8 GENERATED BY HABTAT
    59  008       37    0013B  *******    .PROC,RHBOUT,TAPE8,OUT.        *          *  RUNS THE HABOUT PROGRAM    *          RETURN,ZMA
    60  008       43    4721B  *******    .PROC,RHBOUTA,TAPE8,OUT,ZIN=ZINF.       *          *   RUNS THE HABOUTA PROGRAM  *
    61  008       54    1447B  *******    .PROC,RADBEND,TAPE3,TAPE31.    *          *  RUNS THE ADDBEND PROGRAM WHICH        *  ADDS TH
    62  008       47    1125B  *******    .PROC,RLSTRLW,TAPE31,OUTPUT.   *          *     RUNS THE LSTRLW PROGRAM            *     WHIC
    63  008       43    1551B  *******    .PROC,RMODIOC,ORGF,MODF.       *          *  RUNS THE MODIOC PROGRAM WHICH         *  CAN BE
    64  008       48    1166B  *******    .PROC,RAVDWSP,IWSP,IAVDPT.     *          *      RUNS THE AVDWSP PROGRAM           *
    65  008       50    0315B  *******    .PROC,RHBTIN,F,TAPE8.          * PROC TO GENERATE HABITAT TIME SERIES PROGRAM INPUT FILE
    66  008       58    7242B  *******    .PROC,RTSPLT,T1,T2,T3,T4,T5,TSOUT.      *          * THIS PROC RUNS THE TIME SERIES PRINTER
    67  008       75    3637B  *******    .PROC,RHABT,HAB,FLW,POUT,HWHAB,ANDX.    *          * HABITAT TIME SERIES GENERATOR PROC.  I
    68  008       36    0767B  *******    .PROC,LIBF,F,SF.    * -LIBF(F=FILE BEING NAMED AND ADDED TO SF) *     RETURN,TEMP.
    69  008       25    1126B  *******    .PROC,MODTLIB,TLIB,SF,TEXT.    * MODTLIB(SF=SOURCE FILE       *     TLIB=FILE MODIFIED
    70  008       30    0030B  *******    .PROC,ADDTLIB,TLIB,SF,TEXT.    *     SF=SOURCE FILE,          *     TLIB=FILE ADDED TO.
    71  008       65    0313B  *******    .PROC,RGET1,TLIB,F,TEXT.       *        F=RECORDED TO BE COPIED          *  FROM TEXT LIBRARY
    72  008      152    3167B  *******    .PROC,RTITL,FTEXT,FOUT,NLIN=3.          *          *   READ TITLE LINES ON THE   *   FILE F
    73  008       62    4014B  *******    .PROC,RWSEI4,FD,FMOD,TP4.      *          *          RUNS THE WSEI4 PROGRAM WHICH
    74  008       46    7067B  *******    .PROC,RCRVCK,FCRV,OUT.         *          *  RUNS THE CURVCK PROGRAM WHICH         *  CHECKS
    75  008       31    5743B  *******    .PROC,RQFMT,FLNW.   *          *  CREATES A FILE OF MONTHY FLOWS (OR WHAT EVER)  *  USING T
    76  008       49    2410B  *******    .PROC,RSUMVAF,TAPE12,OUT.      *          *  RUNS THE I4VAF PROGRAM WHICH SUMMARIZES VELOCIT
    77  008       51    6152B  *******    .PROC,RSUMVCE,TAPE11,OUT.      *          *  RUNS THE I4VCE PROGRAM WHICH SUMMARIZES THE VEL
    78  008       47    1243B  *******    .PROC,RTKNWSP,WSP1,WSP2.       *          *  USES THE PROGRAM TAKNWSP TO STRIP N CARDS FROM
    79  008      107    5204B  *******    .PROC,RCRVTIT,FISHY,FISHY2.    *          *    REMOVES THE TITLE LINES AND EOR'S FROM
    80  008       35    4245B  *******    .PROC,RMONDUR,FTS,FORDER.      *          * RUNS THE MONTHLY DURATION PROGRAM     *
    81  008       37    6767B  *******    .PROC,RERRTF,FILM,FILS,OUT.    *          * CALCULATES ERRORS BETWEEN TWO FILES   * FILE FIL
    82  008       56    3670B  *******    .PROC,RVELERR,TP41,TP42,OUT,TAPE3.      *          *  RUNS THE VELERR PROGRAM WHICH COMPARE
    83  008       43    6316B  *******    .PROC,RMVRL,ZTP31,ZTP32.       *          *  MOVES REACH LENGTHS OF C/S FILE ZTP31
    84  008       40    2016B  *******    .PROC,RARTP4,TAPE4,TP4.        *          *   TAPE4 = FLOW RELATED DATA FROM A HYDRAULIC SIM
    85  008       37    3250B  *******    .PROC,RLISTH,ZFISH,OUT.        *          *  LISTS THE H CARDS IN A CRITERIA CURVE DATA SET
    86  008       39    5621B  *******    .PROC,RMKBND1,ZTP31,ZTP32.     *          * MAKES BEND WEIGHTING FACTORS EQUAL TO ONE ON A C
    87  008       65    1240B  *******    .PROC,REFSPWN,TP13S,TP131,EFSPT,OUT.    *          *  RUNS THE EFFECTIVE SPAWNING PROGRAM
    88  008      139    1464B  *******    .PROC,LPTINFO.      REWIND,ZFILE.       COPY,ZFILE,OUTPUT.  .DATA,ZFILE.          13 LINE PR
    89  008       95    4736B  *******    .PROC,LPGP,PRGRM,DATFIL1,Z2,OUT.        *          *  PROCEDURE TO RUN LINE-PRINTER GRAPHIC
    90  008       84    7762B  *******    .PROC,LPGP1,PRGRM,DATFIL1,OUT.          *          *  PROCEDURE TO RUN LINE-PRINTER GRAPHIC
    91  008       90    5341B  *******    .PROC,LPGP2,PRGRM,DATFIL1,Z2,OUT.       *          *  PROCEDURE TO RUN LINE-PRINTER GRAPHIC
    92  008      109    6667B  *******    .PROC,LPGP5,DATFIL1,Z2,Z3,Z4,Z5,OUT.    *          *  PROCEDURE TO RUN LINE-PRINTER GRAPHIC
    93  008       89    4055B  *******    .PROC,LPGPC,DATFIL1,OUT.       *          *  PROCEDURE TO RUN LINE-PRINTER GRAPHICS PROGRAM
    94  008      100    0400B  *******    .PROC,READ13,TAPE13,FMT13.     RETURN,FMT13,LGO.   FTN5,L=DUM,I=READ13.          MAP,OFF.
    95  008       57    0206B  *******    .PROC,WRIT13,FMT13,TAPE13.     RETURN,TAPE13,LGO.  FTN5,L=DUM,I=WRIT13.          MAP,OFF.
    96  008       91    5555B  *******    .PROC,LST13,NREC,TAPE13,FMT13.          FTN5,L=DUM,I=LST13.           MAP,OFF.  LOAD,LGO.
  NO EOF    -------   -----
             4869    7276B

  TOTAL     -------   -----
             4869    7276B
```

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

```
FILE   1 ITEM,IFMLIB.
     1  00B      166    7363B  *******   ADDQSI          PROGRAM ADDQSI(TAPE1,TAPE2,TAPE3,INPUT=/80,OUTPUT,          UTAPE5=INPUT,TA
     2  00B      152    1413B  *******   ADSZFI          PROGRAM ADSZFI(INPUT,OUTPUT,TAPE2,TAPE1,TAPE5=INPUT,TAPE6=OUTPUT)
     3  00B      358    0232B  *******   CORDIN           PROGRAM CORDIN(INPUT,OUTPUT,TAPE1,TAPE2,TAPE5=INPUT,                6  TA
     4  00B       47    3070B  *******   ELVO            SUBROUTINE ELVO                DIMENSION VEL(15)              INTEGER VEL
     5  00B      200    0624B  *******   LDIRCTY         PROGRAM LDIRCTY(TAPE1=/200,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,          U
     6  00B       55    1525B  *******   QTOA            SUBROUTINE QTOA(N,Q,IAQ,QMAX)          DIMENSION Q(100),IQ(100)
     7  00B      516    3471B  *******   ROUGH           PROGRAM ROUGH(INPUT,OUTPUT,TAPE1,TAPE5=INPUT,TAPE6=OUTPUT)          DIME
     8  00B     1384    3154B  *******   HABIN           PROGRAM HABIN(INPUT,TAPE3,OUTPUT,TAPE5=INPUT)                INTEGER TITL(8
     9  00B       30    3260B  *******   RBLKS           SUBROUTINE RBLKS(IN,OUT)                IT=IN                DO 10 I=1,10
    10  00B      228    7456B  *******   AVDWSP          PROGRAM AVDWSP(TAPE2,TAPE1,INPUT,OUTPUT,TAPE6=OUTPUT,               U   T
    11  00B      131    0762B  *******   GENFIL          PROGRAM GENFIL(INPUT,OUTPUT,TAPE3,TAPE5=INPUT,TAPE6=OUTPUT)     C
    12  00B      423    2326B  *******   GCURV           PROGRAM GCURV(INPUT,TAPE13,OUTPUT,TAPE6=OUTPUT,TAPE5=INPUT)     C
    13  00B       41    7110B  *******   FINDLOW         SUBROUTINE FINDLOW(Y,NP,NL)          C      ROUTINE TO LOCATE THE LOWEST POINT
    14  00B       81    4241B  *******   FIT             SUBROUTINE FIT(A,B,R2,OPT,NEGVL)     C      ROUTINE TO COMPUTE THE CONSTANTS F
    15  00B       71    3645B  *******   ACCUM           SUBROUTINE ACCUM(TX,TY,OPT,NEGVL,IW)          C      ROUTINE TO ACUMULATE THE
    16  00B       32    2120B  *******   INITFIT         SUBROUTINE INITFIT      C      ROUTINE TO INITIALIZE THE ACCUMULATORS FOR S
    17  00B       32    5577B  *******   EVAL            FUNCTION EVAL(A,B,X,OPT,NEGVL)       C      ROUTINE TO EVALUATE THE FUNCTION
    18  00B      155    7767B  *******   DSCAN           SUBROUTINE DSCAN(D,WMAX,WSUM)        C      FOR EACH Q PROCESSED THROUGH A SEC
    19  00B      219    2522B  *******   SEARCH          SUBROUTINE SEARCH(COMPQ,DESIRQ,WSEL)          C      ROUTINE TO SEARCH FOR TH
    20  00B      302    1467B  *******   GETN      82/02/27      SUBROUTINE GETN(IX)          C      ROUTINE TO ESTABLISH AN  N  VALUE
    21  00B      595    1614B  *******   CROSSEC   82/06/10      SUBROUTINE CROSSEC(ISET,Q,WSEL,QSTAR,IXL,IXR)          C      ROUT
    22  00B      253    0750B  *******   SLIC2           SUBROUTINE SLIC2(ISET,IX,Q,WSEL,XR,XL,YR,YL,NEGVL)     C      ROUTINE TO HAN
    23  00B      340    5250B  *******   SUMQ            SUBROUTINE SUMQ(ICNTRL)              C      ROUTINE TO HANDLE SUMMARIZING THE
    24  00B      202    4322B  *******   PROCSEC         SUBROUTINE PROCSEC      C      ROUTINE TO PROCESS ALL OF THE SECTIONS IN A
    25  00B       28    0351B  *******   EVQWS           FUNCTION EVQWS(X,A,B,Z1,Z2)          C          C          X=(A*(Y-Z2)**B) + Z1
    26  00B      136    5506B  *******   CENTER          SUBROUTINE CENTER(JTITLE)            C          C      SUBROUTINE TO CENTER AN
    27  00B      202    1435B  *******   ENTSSD          PROGRAM ENTSSD(TAPE1,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)     C
    28  00B      158    2245B  *******   PAGEJCT         SUBROUTINE PAGEJCT(N,MAXLCT,PROGNM,DAT,TIM,JOBID,IW)  C          IFG SYSTEM
    29  00B      101    2535B  *******   PHABARR         PROGRAM PHABARR(TAPE4,TAPE1,OUTPUT)                DIMENSION V(100)
    30  00B      508    0466B  *******   STAGQ           SUBROUTINE STAGQ(Q,S,DS,N,A,B,LOS,LT,NV,IW )          C          C
    31  00B      242    4752B  *******   SUBMOD          PROGRAM SUBMOD(TAPE3,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
    32  00B      265    1021B  *******   SUBMODC         PROGRAM SUBMODC(INPUT,TAPE3,TAPE2,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
    33  00B      101    2135B  *******   VTEST           PROGRAM VTEST(TAPE5,TAPE1,OUTPUT,TAPE6=OUTPUT)               DIMENSION VEL(
    34  00B      407    4641B  *******   VTEST2          PROGRAM VTEST2(INPUT,TAPE1,OUTPUT,TAPE6=OUTPUT,TAPE5=INPUT)     C234567890
    35  00B      675    1022B  *******   WSPPLOT         PROGRAM WSPPLOT(TAPE1,TAPE2,OUTPUT=TAPE2)    C          C      PROGRAM TO REA
    36  00B      680    5626B  *******   WSPPLT2         PROGRAM WSPPLT2(TAPE1,TAPE2,OUTPUT=TAPE2)    C          C      PROGRAM TO REA
    37  00B     3972    4343B  *******   XPRPLT2         SUBROUTINE XPRPLT2 (Y,X,NELM,NCRV,MNPTS,ISYMBOL,IDENT,NEWPG,ISCALEPRPL001
    38  00B      285    2756B  *******   PROCQS          SUBROUTINE PROCQS (LIFSTG)           C          C      THIS ROUTINE DRIVES THE
    39  00B       34    6635B  *******   FINDLW          SUBROUTINE FINDLW(Y,IX,NP,NL)        C      ROUTINE TO LOCATE THE LOW POINT IN
    40  00B      170    0603B  *******   INTRPL          SUBROUTINE INTRPL(X,Y,J1,J2)         C          C      THIS ROUTINE FINDS THE A
    41  00B      317    2105B  *******   PRNMAT          SUBROUTINE PRNMAT(NC,NR,NP,CB,RB,ITYPE)      C          C      ROUTINE TO PRI
    42  00B      193    2532B  *******   LOCATE          SUBROUTINE LOCATE(IWANT)             C      ROUTINE TO POSITION THE SEQUENTIAL
    43  00B      106    5746B  *******   SORTER          SUBROUTINE SORTER(L,M,N,IAS)         C      THIS ROUTINE SORTS A LIST OF INTEG
    44  00B       73    0575B  *******   SORT2           SUBROUTINE SORT2(L,K,N,IAS)          C      THIS ROUTINE SUPPORTS SUBROUTINE S
    45  00B      104    7715B  *******   INDICES         SUBROUTINE INDICES(I)    C      ROUTINE TO SET THE INDICIES SO THAT A CURVE
    46  00B      680    0663B  *******   ACQUIR          SUBROUTINE ACQUIR        C      ROUTINE TO GET THE REQUIRED CURVE SET DEFINI
    47  00B       65    0746B  *******   RECAP           SUBROUTINE RECAP         C          C      SUBROUTINE TO RESET THE VELOCITY
    48  00B      404    5324B  *******   BRIDG           SUBROUTINE BRIDG         C          C      SUBROUTINE TO PROCESS WEIR AND OR
    49  00B      142    0432B  *******   FORF            FUNCTION FORF(PARM)      C          C      FUNCTION TO COMPUTE STARTING ELEV
    50  00B      573    4370B  *******   ERRORS          SUBROUTINE ERRORS(TO,NO)             C          C      SUBROUTINE TO PRINT ALL
    51  00B       91    6503B  *******   ABNORM          SUBROUTINE ABNORM(LOT)   C          C      SUBROUTINE TO HANDLE TERMINATION
    52  00B       80    5765B  *******   BALCOM          SUBROUTINE BALCOM                COMMON/ERRS/      CALC(2),MTAB,NN(2),IWR,MCO
    53  00B       96    1557B  *******   FIXUP           SUBROUTINE FIXUP(Y)              COMMON/LOBS/      HF(2),HV(2),HF1(2),REACHM(
    54  00B      240    5726B  *******   HYDTAB          SUBROUTINE HYDTAB                COMMON/WSPB/      CN(2,100)              COMM
    55  00B      103    4132B  *******   COMPZ1          SUBROUTINE COMPZ1(Z11,Z,ICON)        C          C      NEWTONS METHOD REGULA F
    56  00B      124    4736B  *******   COMPZ2          SUBROUTINE COMPZ2(Z,ICON,Z10,Z11)            C          C      NEWTONS METHO
    57  00B      108    4362B  *******   BALOBS          SUBROUTINE BALOBS                COMMON/AGAINST/   OBSELE(100),ELA(100)
```

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITFM,IFMLIB.

58	00B	240	2031B	*******	FIRSTA	SUBROUTINE FIRSTA C C SUBROUTINE TO INITIATE PROCESSING
59	00B	452	4222B	*******	CRITIC	SUBROUTINE CRITIC(Z) COMMON/ERRS/ CALC(2),MTAB,NN(2),IWR,MC
60	00B	94	3526B	*******	CENTRE	SUBROUTINE CENTRE(JTITLE,J) C C SUBROUTINE TO CENTER AN
61	00B	268	7301B	*******	BLKDATA	BLOCK DATA BLKDATA C C INITIALIZATION OF ALL DATA ELEMEN
62	00B	776	4577B	*******	IFG01A	PROGRAM IFG01A(INPUT,OUTPUT,ALTAPE,TAPE5=INPUT,TAPE6=OUTPUT, 1TAPE
63	00B	216	0372B	*******	READ2	SUBROUTINE READ2(DEPTH,DIST,NOM,DEPT,DAX,NDC,IFM,FLIM1,FLIM2,DTYP
64	00B	39	4004B	*******	SORT	SUBROUTINE SORT (DEPTH,DIST,NOM) DIMENSION DEPTH(200),DIST(200)
65	00B	141	0565B	*******	SAGTP	SUBROUTINE SAGTP (DAX,DEPT,DIST,DEPTH,NOM,DISTX,TEN,TWPF) DIME
66	00B	85	6360B	*******	ELEVFD	SUBROUTINE ELEVFD (DEPT,DIST,NOM,NLEV,DFLEV,FLEV,LDOE) DIME
67	00B	285	0077B	*******	HYCOMP	SUBROUTINE HYCOMP(SLEV,DAX,DEPT,NOM,RLEV,NRLEV,DATUM,DS,QMEA 1,SLO
68	00B	206	0672B	*******	STAGF	SUBROUTINE STAGF(A,B,DS,DALEV,DATUM,DTYP) DIMENSION S(8),Q(8)
69	00B	210	0163B	*******	COMP	SUBROUTINE COMP(DEPT,DIST,NOM,CON,XCON) DIMENSION IS(12) ,AT(21)
70	00B	83	2705B	*******	COPYTLB	PROGRAM COPYTLB(TAPE15,TAPE16,OUTPUT) DIMENSION LINER(8)
71	00B	57	3746B	*******	LSTCNUM	PROGRAM LSTCNUM(TAPE1,OUTPUT,TAPE6=OUTPUT) C C PROGRAM TO
72	00B	258	1665B	*******	GI4NS	PROGRAM GI4NS(TAPE1,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT) DIME
73	00B	156	1402B	*******	GI4QD	PROGRAM GI4QD(TAPE1,INPUT,OUTPUT) C DIMENSION QS(1
74	00B	60	0314B	*******	PAGETS	SUBROUTINE PAGETS(IL,ITITLE,IPN,IW) DIMENSION ITITLE(16)
75	00B	142	7044B	*******	CURVWRT	PROGRAM CURVWRT(TAPE3,TAPE1) DIMENSION IC(5), LKS(5),IT(5),X(30
76	00B	89	2076B	*******	STRIP1	PROGRAM STRIP1(TAPE11,TAPE12,OUTPUT,TAPE6=OUTPUT) DIMENSION LI
77	00B	366	2766B	*******	HABPLT	PROGRAM HABPLT(TAPE1,TAPE2,TAPE7,OUTPUT=TAPE2) C C P
78	00B	459	1564B	*******	WSPFA	FUNCTION WSPFA(Z2,N) C C THIS FUNCTION IS USED TO OBTAIN T
79	00B	462	6250B	*******	WSPFB	FUNCTION WSPFB(Z2,N) C C THIS FUNCTION IS USED TO OBTAIN T
80	00B	467	7377B	*******	WSPFC	FUNCTION WSPFC(Z2,N) C C THIS FUNCTION IS USED TO OBTAIN T
81	00B	460	4202B	*******	WSPFD	FUNCTION WSPFD(Z2,N) C C THIS FUNCTION IS USED TO OBTAIN T
82	00B	459	6701B	*******	WSPFE	FUNCTION WSPFE(Z2,N) C C THIS FUNCTION IS USED TO OBTAIN T
83	00B	219	7414B	*******	CORDFF	PROGRAM CORDFF(INPUT,OUTPUT,TAPE1,TAPE2,TAPE5=INPUT, 6 TA
84	00B	246	7077B	*******	HAINTP	PROGRAM HAINTP(TAPE1,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT) DIME
85	00B	199	0431B	*******	STAGWSP	83/01/18. PROGRAM STAGWSP(INPUT,OUTPUT,TAPE1,TAPE2,TAPE5=INPUT,
86	00B	121	1032B	*******	STRIP14	PROGRAM STRIP14(TAPE1,TAPE2,OUTPUT,TAPE6=OUTPUT) C C
87	00B	224	4107B	*******	HYSCF	PROGRAM HYSCF(TAPE3,TAPE4,TAPE1,INPUT,OUTPUT,TAPE5=INPUT, U TAP
88	00B	60	7401B	*******	TITLE	PROGRAM TITLE(INPUT,TAPE13,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT) DIME
89	00B	86	4201B	*******	PRINTC	PROGRAM PRINTC(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
90	00B	117	1463B	*******	CONVCD	PROGRAM CONVCD(TAPE11,TAPE12,OUTPUT) DIMENSION LINER(9), X(12
91	00B	110	6731B	*******	PHABAR2	PROGRAM PHABAR2(TAPE4,TAPE1,OUTPUT) DIMENSION V(100)
92	00B	145	3607B	*******	GI4XS	PROGRAM GI4XS(TAPE2,TAPE1,INPUT,OUTPUT) DIMENSION LINER(9)
93	00B	194	4722B	*******	HYSBF	PROGRAM HYSBF(TAPE1,TAPE3,TAPE4,OUTPUT,TAPE6=OUTPUT) DIMENSION JOBI
94	00B	178	0120B	*******	ADDSLP	PROGRAM ADDSLP(TAPE10,TAPE11,INPUT,OUTPUT,TAPE5=INPUT, U
95	00B	747	4476B	*******	WSPCK	PROGRAM WSPCK(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT) C
96	00B	174	7043B	*******	READR	SUBROUTINE READR(IDNO,JFMT,JI,JF) COMMON/CK/SX(400),SY(400
97	00B	274	5532B	*******	CHECK	SUBROUTINE CHECK(IXI,IXF,ITYPE) C C THIS SUBROUTINE CHECKS T
98	00B	37	6224B	*******	PAGE132	SUBROUTINE PAGE132(IL,IPN,IW) 101 FORMAT(1H1,7X,*PROGRAM NAME - *,A1
99	00B	490	1472B	*******	LSTCRVI	PROGRAM LSTCRVI(TAPE1,TAPE66,INPUT,OUTPUT,TAPE5=INPUT, U
100	00B	107	1346B	*******	GI4WE	PROGRAM GI4WE(TAPE1,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT) C
101	00B	360	0515B	*******	DIRI4	82/06/10 PROGRAM DIRI4(TAPE1,TAPE3,TAPE4,OUTPUT) C C
102	00B	116	5546B	*******	SPLITP8	PROGRAM SPLITP8(TAPE8,TAPE18,INPUT,OUTPUT,TAPE5=INPUT, U T
103	00B	361	4344B	*******	LSTVD	PROGRAM LSTVD(TAPE3,TAPE4,OUTPUT,TAPE6=OUTPUT) C C
104	00B	152	0612B	*******	WMOD	PROGRAM WMOD(TAPE1,TAPE2,INPUT,OUTPUT,TAPE6=OUTPUT,TAPE5=INPUT)
105	00B	182	4572B	*******	CURVWTB	PROGRAM CURVWTB(TAPE3,TAPE1,OUTPUT,INPUT,TAPE5=INPUT, U
106	00B	159	1734B	*******	CURVWTA	PROGRAM CURVWTA(TAPE3,TAPE1) DIMENSION IC(5), LKS(5),IT(5),X(30
107	00B	998	1130B	*******	CKI4	83/08/08. PROGRAM CKI4(TAPE1,TAPE2,OUTPUT=TAPE2) C
108	00B	838	6132B	*******	ACQUIR2	SUBROUTINE ACQUIR2(IDCRVS,NCURV,ISPCDS,LFSDES,INDATE,SX,SY, U
109	00B	143	3214B	*******	LOCATE2	SUBROUTINE LOCATE2(IWANT,IW,IT,IFLOW) C ROUTINE TO POSITION THE
110	00B	349	5544B	*******	LSTCRV	PROGRAM LSTCRV(TAPE1,INPUT,OUTPUT,TAPE5=INPUT, U TAPE6=OUT
111	00B	202	6747B	*******	AVDTP4	PROGRAM AVDTP4(TAPE2,TAPE1,TAPE4,INPUT,OUTPUT,TAPE6=OUTPUT, U T
112	00B	210	5232B	*******	STGQI	81/06/04. PROGRAM STGQ1(TAPE99,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OU
113	00B	160	0706B	*******	PAGEI	81/06/04. SUBROUTINE PAGEI(N,MAXLCT,PROGNM,DAT,TIM,JOBID,IW)
114	00B	714	6111B	*******	PNTVA	81/06/04. PROGRAM PNTVA(TAPE3,TAPE4,TAPE41,TAPE6,INPUT,OUTPUT,

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM,IFMLIB.

115	00B	132	3522B	*******	TCORD	81/06/04.	PROGRAM TCORD(TAPE15,TAPE16,INPUT,OUTPUT,TAPE5=INPUT,T
116	00B	456	2741B	*******	GETCRV	81/06/05.	PROGRAM GETCRV(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,
117	00B	101	4553B	*******	TP8NOTE	81/06/05.	PROGRAM TP8NOTE(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,T
118	00B	693	4562B	*******	HABOUT	81/06/05.	PROGRAM HABOUT(TAPE8,TAPE9,INPUT,OUTPUT,TAPE5=INPUT,TA
119	00B	245	1634B	*******	I4TOWSP	PROGRAM I4TOWSP(TAPE1,TAPE2,OUTPUT,TAPE6=OUTPUT) C C	
120	00B	248	1470B	*******	WSEI4	83/05/19.	PROGRAM WSEI4(TAPE1,TAPE2,TAPE4,OUTPUT) C
121	00B	222	3577B	*******	ADDCNI4	81/07/04.	PROGRAM ADDCNI4(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,
122	00B	405	2512B	*******	NMODWSP	81/10/29.	PROGRAM NMODWSP(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,
123	00B	204	1330B	*******	ADDWS	81/07/04.	PROGRAM ADDWS(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,TAP
124	00B	430	4225B	*******	MODQARD	81/08/06.	PROGRAM MODQARD(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,
125	00B	305	0251B	*******	CGNI4	81/07/04.	PROGRAM CGNI4(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT
126	00B	503	5640B	*******	MODIOC	81/08/06.	PROGRAM MODIOC(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,TA
127	00B	353	0757B	*******	NMODC	81/07/05.	PROGRAM NMODC(TAPE1,TAPE2,INPUT,OUTPUT,TAPE5=INPUT,
128	00B	659	3513B	*******	AVDEPTH	83/01/22.	PROGRAM AVDEPTH(INPUT,OUTPUT,TAPE8,TAPE5=INPUT,TAPE6=O
129	00B	254	4240B	*******	AVDTP8	81/09/15.	SUBROUTINE AVDTP8(ITIT,NS,NQ,STA,Q,W,WP,WD,DEP,CSA)
130	00B	50	4021B	*******	START	81/07/05.	SUBROUTINE START(NSW,L,N,NN,K) COMMON ELEV(10
131	00B	90	5550B	*******	END	81/07/05.	SUBROUTINE END(NSW,L,N,NN,K,AREA,WIDTH,WP) COMM
132	00B	81	4751B	*******	WFD	81/07/05.	SUBROUTINE WFD(N2,WSL,RDEP,WD) DIMENSION DEPR
133	00B	269	0614B	*******	COMPA	81/07/05.	SUBROUTINE COMPA(NSECT,NELEV,STA2) DIM
134	00B	68	0411B	*******	VCBLD	81/07/05.	PROGRAM VCBLD(TAPE17,INPUT,OUTPUT,TAPE6=OUTPUT,TAPE5=I
135	00B	108	0067B	*******	PNTVC	81/07/05.	PROGRAM PNTVC(TAPE4,TAPE41,INPUT,OUTPUT,TAPE5=INPUT,TA
136	00B	326	6100B	*******	HABTIN	81/09/14.	PROGRAM HABTIN(TAPE1,TAPE8,INPUT,OUTPUT,TAPE5=INPUT,TA
137	00B	714	4011B	*******	HABT	81/09/14.	PROGRAM HABT(INPUT,OUTPUT,TAPE2,TAPE10,TAPE11,TAPE23,
138	00B	184	5625B	*******	HABANQ	81/07/09.	SUBROUTINE HABANQ(N, Q,HA,QH,NPTS,NTRAN,AT,BT,TYPE,H
139	00B	228	2724B	*******	FLWIN	81/07/09.	SUBROUTINE FLWIN(TITLE,AM,A,FT,IOUT,NQ,AU,Q,YER,N,IR,I
140	00B	63	4711B	*******	HADERM	81/07/09.	SUBROUTINE HADERM(TITLE,AM,STAN,INFO,XNFO,AU,IEND,IR,I
141	00B	196	7665B	*******	HADDOE	81/07/09.	SUBROUTINE HADDOE(TITLE,AM,STAN,INFO,XNFO,AU,IEND,IR,I
142	00B	72	2043B	*******	AVEFLW	81/07/09.	SUBROUTINE AVEFLW(Q,N,A,IOUT,IW) DIMENSION Q(13
143	00B	106	0507B	*******	QWRITE	81/07/09.	SUBROUTINE QWRITE(TITLE,AM,IOUT,NQ,AU,Q,YER,N,IW)
144	00B	257	0435B	*******	PLOTC	81/07/09.	SUBROUTINE PLOTC(CON,XCON,DEPT,YER,NOM,IW) C
145	00B	56	3641B	*******	IWORD	81/07/09.	FUNCTION IWORD(IPOS) C C FUNCTION TO
146	00B	101	0453B	*******	HADGS	81/07/09.	SUBROUTINE HADGS(TITLE,AM,STAN,INFO,XNFO,AU,IEND,IR,IW
147	00B	124	2714B	*******	PAGEQ	81/07/09.	SUBROUTINE PAGEQ(N,MAXLCT,PROGNM,DAT,TIM,JOBID,IW)
148	00B	418	2041B	*******	TSPLT	81/08/14.	PROGRAM TSPLT(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE4,TAP
149	00B	203	4340B	*******	REDAYR	81/08/14.	SUBROUTINE REDAYR(IFIRST,ILAST,IEND) C
150	00B	103	7321B	*******	TBLPRT	81/08/14.	SUBROUTINE TBLPRT(LPOUT) C COMM
151	00B	83	4066B	*******	PRTPLT	81/08/14.	SUBROUTINE PRTPLT(IOPT,LPOUT) C COMM
152	00B	215	0515B	*******	SETUP	81/08/14.	C SUB SETUP INITIALIZES PARAMETERS SUBROUTINE SET
153	00B	166	1373B	*******	PPLOT	81/08/14.	SUBROUTINE PPLOT(LPOUT,K) C COMM
154	00B	216	3136B	*******	DEFSCL	81/08/14.	SUBROUTINE DEFSCL C COMMON /GPARMS
155	00B	44	4206B	*******	ORDERM	81/08/14.	SUBROUTINE ORDERM(X,N,IH) C FIND ORDER OF MAGN
156	00B	616	1015B	*******	QFORMT	81/07/09.	PROGRAM QFORMT(INPUT,OUTPUT,TAPE1,TAPE5=INPUT,TAPE6=OU
157	00B	96	7147B	*******	ADDBEND	81/08/06.	PROGRAM ADDBEND(TAPE3,TAPE31,INPUT,OUTPUT,TAPE6=OUTPUT
158	00B	93	2604B	*******	LSTRLW	81/08/06.	PROGRAM LSTRLW(TAPE3,OUTPUT,TAPE6=OUTPUT) C
159	00B	158	7626B	*******	PAGE	81/08/07.	SUBROUTINE PAGE(N,LINES,IWR) C C SUB
160	00B	113	2246B	*******	FILNAME	80/00/00	PROGRAM FILNAME(TAPE11,TAPE12,INPUT,OUTPUT,TAPE5=INPUT,
161	00B	482	4207B	*******	CURVCK	81/08/14.	PROGRAM CURVCK(TAPE5,TAPE6,OUTPUT) C
162	00B	546	7247B	*******	WSPQARD	81/08/14.	PROGRAM WSPQARD(TAPE1,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=O
163	00B	740	0157B	*******	IFG4IN	81/08/14.	PROGRAM IFG4IN(INPUT,OUTPUT,TAPE1,TAPE2,TAPE5=INPUT)
164	00B	164	5536B	*******	GI4CV	81/10/26.	PROGRAM GI4CV(TAPE1,INPUT,OUTPUT) DIME
165	00B	482	1734B	*******	I4VAF	81/10/26.	PROGRAM I4VAF(TAPE11,OUTPUT,TAPE2,INPUT,TAPE5=INPUT,TA
166	00B	251	7751B	*******	STATSTC	81/10/26.	SUBROUTINE STATSTC(N,ISW,STA,J,Q,ERR) DIME
167	00B	332	2134B	*******	I4VCE	81/10/26.	PROGRAM I4VCE(TAPE12,OUTPUT,TAPE2,INPUT,TAPE5=INPUT,TA
168	00B	198	2354B	*******	STATSTB	81/10/26.	SUBROUTINE STATSTB(N,ISW,STA,Q,ERR) DIME
169	00B	309	4037B	*******	CURVFIL	81/10/26.	PROGRAM CURVFIL(TAPE2,TAPE3,INPUT,OUTPUT,TAPE6=OUTPUT,
170	00B	45	3777B	*******	TAKNWSP	81/10/27.	PROGRAM TAKNWSP(TAPE11,TAPE12,OUTPUT) C
171	00B	101	3147B	*******	MONDUR	82/02/04.	PROGRAM MONDUR(TAPE1,TAPE2,OUTPUT) C

ECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

```
ILE   1 ITEM,IFMLIB.
  172  00B       64   0516B  *******   OPDER      82/02/04.                SUBROUTINE ORDER(A,IN,N,INDEX,IDEM)          C
  173  00B       98   6705B  *******   READN      82/02/04.                SUBROUTINE READN(Q,IYER,N,ND,ISTA,IR,IW)     C
  174  00B       87   3445B  *******   WRITEQ     82/02/04.                SUBROUTINE WRITEQ(Q,IYER,N,ND,IFM,IW)        C
  175  00B      557   5564B  *******   ARCOMP     82/12/09.                SUBROUTINE ARCOMP(Y)    C          C       SUBROUTINE TO
  176  00B      287   5614B  *******   LPTP8C                    PROGRAM LPTP8C(TAPE81,TAPE82,OUTPUT,TAPE6=OUTPUT)      C          C
  177  00B      201   0350B  *******   LPTTP8                    PROGRAM LPTTP8(TAPE8,OUTPUT,TAPE6=OUTPUT)    C          C  PLOTS EACH LIFE S
  178  00B      306   4102B  *******   LPTCRV                    PROGRAM LPTCRV(TAPE6=513,TAPE2=513,OUTPUT=TAPE2)       C          C      PROG
  179  00B      827   6507B  *******   LPTQHA                    PROGRAM LPTQHA(TAPE1,TAPE2,TAPE33,OUTPUT,INPUT,                1
  180  00B      911   4757B  *******   LPTDURA                   PROGRAM LPTDURA(TAPE1,TAPE2,TAPE33,INPUT,OUTPUT,               U TAPE5=INPUT)
  181  00B      827   6450B  *******   LPTDUR                    PROGRAM LPTDUR(TAPE1,TAPE2,TAPE33,INPUT,OUTPUT                 U  ,TAPE5=INPUT
  182  00B      658   6131B  *******   LPTDAN                    PROGRAM LPTDAN(TAPE1,TAPE2,TAPE33,INPUT,OUTPUT,                U  TAPE5=INPUT)
  183  00B      470   6617B  *******   LPTSAN                    PROGRAM LPTSAN(TAPE1,TAPE2,TAPE33,OUTPUT,INPUT,                1
  184  00B      199   7715B  *******   LPTQM                     PROGRAM LPTQM(TAPE1,TAPE6,INPUT,OUTPUT,TAPE10=INPUT)   C          C  USES MA
  185  00B      278   0530B  *******   LPTP8A                    PROGRAM LPTP8A(TAPE81,OUTPUT,TAPE6=OUTPUT)   C          C       PLOTS D
  186  00B       79   7415B  *******   LPTVCE                    PROGRAM LPTVCE(TAPE11,TAPE6,OUTPUT)          C          C  PLOTS VELOCITY CA
  187  00B       79   0612B  *******   LPTQVF                    PROGRAM LPTQVF(TAPE12,TAPE6,OUTPUT)          C          C  PLOTS VELOCITY AD
  188  00B      105   3030B  *******   READF                     SUBROUTINE READF(Q,IYER,NY,ISTA,IFM,IN,ND)   C          C READS EITHER A USG
  189  00B       38   5675B  *******   WRITEAQ                   SUBROUTINE WRITEAQ(QD,IYR,N,ND,IW)           C          C   WRITE A TABLE OF
  190  00B      210   6633B  *******   DURSTAT                   SUBROUTINE DURSTAT(XPNT,QD,QD2,HAVE,HAVE2,QD50,QD250,QD90,QD290,
  191  00B      423   7224B  *******   LPTTSN                    PROGRAM LPTTSN(INPUT,OUTPUT,TAPE1,TAPE2,TAPE3,TAPE4,TAPE5,                ^
  192  00B      308   1752B  *******   MAKTP6                    PROGRAM MAKTP6(TAPE1,TAPE6,OUTPUT,TAPE2=OUTPUT)                DIMENSION IC(1
  193  00B      698   4010B  *******   MAPA       83/01/28.                SUBROUTINE MAPA (NN,H,V,LO,LH,XL,XH,YL,YH,HN,VN,LHT,KT
  194  00B      370   5374B  *******   MAPA1      83/01/28.                SUBROUTINE MAPA1        C          C-----**************
  195  00B      684   0775B  *******   MAPA2      83/01/28.                SUBROUTINE MAPA2        C          C-----**************
  196  00B      336   4233B  *******   MAPA3      83/01/28.                SUBROUTINE MAPA3  ( H,V )       C          C----------****
  197  00B      201   0056B  *******   MAPA4      83/01/28.                SUBROUTINE MAPA4        C          C-----**************
  198  00B      152   6754B  *******   MAPA5      83/01/28.                SUBROUTINE MAPA5  ( NV,NH,NCHAR )              C
  199  00B      100   3223B  *******   MAPA6      83/01/28.                    IDENT  MAPA6                       ENTRY  MAPA6
  200  00B      286   3370B  *******   MAPA7      83/01/28.                SUBROUTINE MAPA7  (  CMINA,CMAXA,NSC,N)       C
  201  00B       93   3735B  *******   TRNC       83/01/28.                FUNCTION TRNC(ARG)      C          C     **************
  202  00B      201   1255B  *******   ERRTF      82/07/12.                PROGRAM ERRTF(TAPE11,TAPE12,TAPE1,OUTPUT,TAPE6=OUTPUT)
  203  00B       32   7143B  *******   ERRC       82/07/12.                SUBROUTINE ERRC(QSUM,QASUM,Q2SUM,NVQ,ERR,ERRA,SDIV)
  204  00B       94   6753B  *******   MVRL       82/07/12.                PROGRAM MVRL(TAPE31,TAPE32,OUTPUT)             C
  205  00B       90   4013B  *******   LISTH      82/07/12.                PROGRAM LISTH(TAPE5,OUTPUT,TAPE6=OUTPUT)       C
  206  00B       52   7205B  *******   MKBND1     82/07/12.                PROGRAM MKBND1(TAPE31,TAPE32,OUTPUT)           C
  207  00B      223   6103B  *******   HABTAT     82/07/31.                PROGRAM HABTAT(TAPE1,TAPE2,TAPE6,TAPE3,TAPE4,TAPE8,TAP
  208  00B      689   6660B  *******   CROSEC     82/12/14.                SUBROUTINE CROSEC(NP,NVS,WSEL,STAID,RL,W,A,UA)
  209  00B      318   3463B  *******   REACH      82/07/31.                SUBROUTINE REACH(TA,TUA)               C       ROUTINE TO PRO
  210  00B      566   7557B  *******   VELERR     82/07/31.                PROGRAM VELERR(TAPE3,TAPE41,TAPE42,TAPE1,OUTPUT,TAPE6=
  211  00B      110   4573B  *******   LSTTP3     82/08/22.                PROGRAM LSTTP3(TAPE3,OUTPUT,TAPE6=OUTPUT)      C
  212  00B      360   7017B  *******   EFFSPWN    82/09/24.                PROGRAM EFFSPWN(TAPE13,TAPE23,TAPE33,TAPE2,OUTPUT,INPU
  213  00B       43   5155B  *******   PAGEA      82/09/24.                SUBROUTINE PAGEA(IL,IPN,IW)                    DIMENSION ITIT
  214  00B      558   5747B  *******   SPCIES     82/12/13.                SUBROUTINE SPCIES       C          C       THIS ROUTINE D
  215  00B      666   4056B  *******   HABOUTA    82/12/15.                PROGRAM HABOUTA(TAPE8,TAPE9,INPUT,OUTPUT,TAPE5=INPUT,T
  216  00B      429   1222B  *******   SLICE      83/08/29.                SUBROUTINE SLICE(STAID,XL,XR,YL,YR,S,WSEL,VM,RL,W,AREA
  217  00B     1422   3572B  *******   INITAL     83/05/10.                SUBROUTINE INITAL       C          C       SUBROUTINE TO
  218  00B       46   1203B  *******   HNOSV      83/05/04.                SUBROUTINE HNOSV(V,VN,D,DN,ITYP,D65,A,B)       C
  219  00B       61   0556B  *******   WFTEST     83/05/04.                SUBROUTINE WFTEST(CF,V,D,S,IW)         C          C          D
  220  00B      273   7155B  *******   IFG4       83/05/07.                PROGRAM IFG4(TAPE1,TAPE2,TAPE3,TAPE4,OUTPUT=TAPE2,
  221  00B      209   5771B  *******   NVQB       83/05/07.                SUBROUTINE NVQB(ACON,BCON,CALVEL,NOBS,NP,Q,NSETS,ITYP,
  222  00B      504   1133B  *******   PROCQ      83/08/31.                SUBROUTINE PROCQ(Q,WSEL)                       C       ROUTINE TO PRO
  223  00B      514   5345B  *******   INITIAL    83/05/21.                SUBROUTINE INITIAL      C          C       ROUTINE TO INI
  224  00B     1241   5647B  *******   CALIB      83/08/29.                SUBROUTINE CALIB(QA1,QB1)                      C          C          L
  225  00B      807   1534B  *******   READER     83/08/06.                SUBROUTINE READER(WSGVN)                       C       ROUTINE TO REA
  226  00B      373   1733B  *******   HABINS     83/08/31.                PROGRAM HABINS(TAPE1,INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OU
  227  00B      135   3721B  *******   HABBAN     83/09/05.                SUBROUTINE HABBAN(DAT,TIM,IW)          C          C
  228  00B      115   4573B  *******   WSPBAN     83/09/09.                SUBROUTINE WSPBAN(DAT,TIM,IW)          C          C
```

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM, IEMLIB.

229	00B	128	4226B	*******	IFG4BAN	83/09/05.
230	00B	677	6615B	*******	WSP	83/08/31.
231	00B	1678	7157B	*******	PERUSE	83/08/31.
232	00B	605	6621B	*******	F	83/08/31.
233	00B	1505	1334B	*******	SCRIBE	83/08/31.
234	17B	0	0000B			
		72492	4550B			
TOTAL		72492	4550B			

```
      SUBROUTINE IFG4BAN(DAT,TIM,IW)     C          C
      PROGRAM WSP(INPUT,OUTPUT,TAPE3=512,TAPE4=512,TAPE8, TA
      SUBROUTINE PERUSE                  C          C          C
      FUNCTION F(Z2,N)                   C          C       THIS FUNCTION
      SUBROUTINE SCRIBE(H)               C          C       SUBROUTINE TO
```

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM, PHABSIM.

1	00B	1077	6032B	PHABSIM	83/09/05.								
2	00B	377	7663B	ADDQSI	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
3	00B	327	1153B	ADSZFI	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
4	00B	592	3651B	CORDIN	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
5	00B	704	1767B	CURVFIL	81/10/26.	23.28.37	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
6	00B	89	0540B	ELVO	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
7	00B	524	1474B	LDIRCTY	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
8	00B	92	1505B	QTOA	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
9	00B	863	2046B	ROUGH	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
10	00B	2120	3517B	HABIN	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
11	00B	47	3105B	RBLKS	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
12	00B	583	4032B	HABINS	83/08/31.	22.35.15	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
13	00B	470	2622B	AVDWSP	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
14	00B	345	6420B	GENFIL	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
15	00B	682	7554B	GCURV	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
16	00B	45	7531B	FINDLOW	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
17	00B	89	6417B	FIT	80/08/20.	15.13.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
18	00B	90	2317B	ACCUM	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
19	00B	39	3056B	INITFIT	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
20	00B	54	5336B	EVAL	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION	OPT=2
21	00B	76	6750B	DSCAN	80/08/20.	15.13.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
22	00B	111	5020B	SEARCH	80/08/20.	15.13.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
23	00B	165	3260B	GETN	82/06/10.	20.18.52	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
24	00B	518	3010B	CROSSEC	82/06/10.	20.18.52	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
25	00B	152	6652B	SLIC2	80/08/20.	15.13.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
26	00B	255	3257B	SUMQ	80/08/20.	15.13.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
27	00B	185	7260B	PROCSEC	80/08/20.	15.13.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
28	00B	50	5452B	EVQWS	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION	OPT=2
29	00B	98	6021B	CENTER	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
30	00B	480	2427B	ENTSSD	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
31	00B	122	1047B	PAGEJCT	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
32	00B	278	7253B	PHABARR	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
33	00B	612	2420B	STAGQ	80/08/22.	18.05.57	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
34	00B	472	4567B	SUBMOD	80/08/26.	16.44.17	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
35	00B	473	5435B	SUBMODC	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
36	00B	207	1724B	VTEST	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
37	00B	682	2407B	VTEST2	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
38	00B	636	5173B	WSPPLOT	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
39	00B	638	5331B	WSPPLT2	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
40	00B	1196	1137B	XPRPLT2	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
41	00B	369	5105B	HABTAT	82/07/31.	22.14.10	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=1
42	00B	193	7327B	PROCQS	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
43	00B	46	1653B	FINDLW	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
44	00B	87	4254B	INTRPL	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
45	00B	371	2072B	PRNMAT	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
46	00B	147	4573B	LOCATE	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
47	00B	93	1702B	SORTER	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
48	00B	79	4356B	SORT2	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
49	00B	120	1153B	INDICES	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
50	00B	663	4535B	ACQUIR	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
51	00B	601	2164B	CROSEC	82/12/14.	08.58.43	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
52	00B	687	4230B	WSP	83/08/31.	22.52.58	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
53	00B	65	0026B	RECAP	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
54	00B	310	7124B	BRIDG	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
55	00B	333	7427B	ARCOMP	82/12/09.	22.34.40	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
56	00B	148	6735B	FORF	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION	OPT=2
57	00B	895	7250B	ERRORS	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM,PHABSIM.

58	00B	100	3113B	ABNORM	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
59	00B	97	2200B	BALCOM	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
60	00B	326	0150B	F	83/08/31.	22.52.58	NOS 1.4	FTN	4.8552	6464	I	FUNCTION OPT=2
61	00B	89	5002B	FIXUP	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
62	00B	257	1173B	HYDTAB	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
63	00B	87	7361B	COMPZ1	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
64	00B	105	2600B	COMPZ2	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
65	00B	101	7262B	BALOBS	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
66	00B	222	5606B	FIRSTA	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
67	00B	300	1575B	CRITIC	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
68	00B	66	4124B	CENTRE	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
69	00B	372	1045B	BLKDATA	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	BLOCKDATA OPT=2
70	00B	743	4424B	IFG01A	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
71	00B	326	3257B	READ2	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
72	00B	42	5521B	SORT	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
73	00B	212	7770B	SAGTP	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
74	00B	92	3341B	ELEVFD	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
75	00B	324	1522B	HYCOMP	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
76	00B	396	4545B	STAGF	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
77	00B	272	7731B	COMP	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
78	00B	184	6622B	COPYTLB	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
79	00B	154	3304B	LSTCNUM	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
80	00B	307	7540B	GI4CV	81/10/26.	21.27.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
81	00B	440	0247B	GI4NS	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
82	00B	333	3517B	GI4QD	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
83	00B	82	2224B	PAGETS	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2
84	00B	335	3747B	CURVWRT	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
85	00B	198	2562B	STRIP1	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
86	00B	481	5551B	HABPLT	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
87	00B	260	0456B	WSPFA	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION OPT=2
88	00B	258	1226B	WSPFB	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION OPT=2
89	00B	264	2002B	WSPFC	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION OPT=2
90	00B	259	1670B	WSPFD	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION OPT=2
91	00B	257	7421B	WSPFE	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	FUNCTION OPT=2
92	00B	410	3372B	CORDFF	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
93	00B	507	2401B	HAINTP	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
94	00B	494	7670B	STAGWSP	83/01/18.	21.17.23	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
95	00B	238	1716B	STRIPI4	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
96	00B	465	3110B	HYSCF	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
97	00B	179	4140B	TITLE	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
98	00B	223	3747B	PRINTC	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
99	00B	253	5747B	CONVCD	80/07/14.	22.33.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
100	00B	285	3121B	PHABAR2	80/08/25.	23.24.52	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=2
101	00B	561	6336B	DIRI4	82/06/10.	20.18.52	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
102	00B	325	2702B	GI4XS	80/09/17.	19.34.07	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
103	00B	421	6564B	HYSBF	80/09/17.	19.34.07	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
104	00B	1114	0430B	IFG4IN	81/08/14.	08.06.51	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
105	00B	997	4735B	WSPQARD	81/08/14.	08.06.51	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
106	00B	359	5446B	ADDSLP	80/09/17.	19.34.07	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
107	00B	810	2144B	WSPCK	80/09/17.	19.34.07	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
108	00B	505	0224B	CURVCK	81/08/14.	08.06.51	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
109	00B	192	1611B	READR	80/09/17.	19.34.07	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
110	00B	304	2726B	CHECK	80/09/17.	19.34.07	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
111	00B	68	6605B	PAGE132	80/09/17.	19.34.07	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
112	00B	752	5242B	LSTCRVI	80/09/25.	20.49.33	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
113	00B	848	4466B	READER	83/08/06.	09.17.02	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
114	00B	220	6717B	GI4WE	80/09/22.	18.21.37	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM,PHABSIM.

115	00B	265	1054B	SPLITP8	81/06/01.	23.46.27	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
116	00B	479	1325B	LSTVD	81/06/02.	09.39.42	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
117	00B	394	3103B	WMOD	81/06/02.	10.28.04	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
118	00B	446	6164B	CURVWTB	81/06/02.	12.23.46	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
119	00B	367	6533B	CURVWTA	81/06/02.	12.24.12	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
120	00B	1169	7222B	CKI4	83/08/08.	19.42.28	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
121	00B	838	6563B	ACQUIR2	81/06/02.	15.15.36	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
122	00B	152	2537B	LOCATE2	81/06/02.	15.15.36	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
123	00B	511	5622B	LSTCRV	81/06/02.	16.00.20	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
124	00B	418	4656B	AVDTP4	81/06/04.	00.02.00	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
125	00B	420	5403B	STGQI	81/06/04.	22.07.46	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
126	00B	80	2341B	PAGEI	81/06/04.	22.07.46	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
127	00B	922	5722B	PNTVA	81/06/04.	22.18.22	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
128	00B	301	6022B	TCORD	81/06/04.	22.26.24	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
129	00B	611	4161B	GETCRV	81/06/05.	10.50.21	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
130	00B	252	7637B	TP8NOTE	81/06/05.	10.50.21	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
131	00B	1281	6013B	HABOUT	81/06/05.	10.50.21	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
132	00B	1038	0144B	HABOUTA	82/12/15.	00.06.26	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
133	00B	387	5264B	I4TOWSP	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
134	00B	419	2341B	WSEI4	83/05/19.	20.42.15	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
135	00B	401	1606B	ADDCNI4	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
136	00B	762	3644B	NMODWSP	81/10/29.	23.31.09	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
137	00B	392	6550B	ADDWS	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
138	00B	753	0171B	MODGARD	81/08/06.	23.46.48	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
139	00B	465	6232B	CGNI4	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
140	00B	636	0732B	MODIOC	81/08/06.	23.31.24	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
141	00B	634	5213B	NMODC	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
142	00B	942	0071B	AVDEPTH	83/01/22.	19.21.57	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
143	00B	430	1255B	AVDTP8	81/09/15.	08.30.03	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
144	00B	63	6433B	START	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
145	00B	107	0524B	END	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
146	00B	97	4666B	WFD	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
147	00B	313	6000B	COMPA	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
148	00B	200	1614B	VCBLD	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
149	00B	308	0274B	PNTVC	81/07/06.	15.46.56	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
150	00B	1125	2375B	HABT	81/09/14.	21.56.14	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=2
151	00B	318	1304B	HABANQ	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
152	00B	338	1031B	FLWIN	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
153	00B	103	7212B	HADERM	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
154	00B	168	1657B	HADDOE	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
155	00B	93	5474B	AVEFLW	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
156	00B	161	3251B	QWRITE	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
157	00B	228	1117B	PLOTC	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
158	00B	48	5340B	IWORD	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	FUNCTION	OPT=2
159	00B	132	5226B	HADGS	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
160	00B	70	7120B	PAGEQ	81/07/09.	22.02.30	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=2
161	00B	677	3043B	HABTIN	81/09/14.	21.55.05	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
162	00B	740	1312B	QFORMT	81/07/09.	23.14.30	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
163	00B	277	7376B	ADDBEND	81/08/06.	09.57.20	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
164	00B	198	7301B	LSTRLW	81/08/06.	09.58.00	NOS 1.4	FTN	4.8508	6464	I	PROGRAM	OPT=1
165	00B	1633	7247B	PERUSE	83/08/31.	22.52.58	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
166	00B	1710	7363B	SCRIBE	83/08/31.	22.52.58	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
167	00B	140	1550B	WSPBAN	83/09/05.	14.17.05	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
168	00B	119	6547B	PAGE	81/08/07.	22.35.05	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINE	OPT=1
169	00B	357	2243B	REACH	82/07/31.	22.14.10	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=1
170	00B	159	6077B	HABBAN	83/09/05.	14.17.05	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
171	00B	153	2527B	IFG4BAN	83/09/05.	14.17.05	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM, PHABSIM.

172	00B	261	0006B	FILNAME	81/07/03.	23.12.45	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
173	00B	695	6726B	TSPLT	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
174	00B	171	3223B	REDAYR	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
175	00B	117	7306B	TBLPRT	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
176	00B	101	7562B	PRTPLT	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
177	00B	225	0017B	SETUP	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
178	00B	142	5162B	PPLOT	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
179	00B	213	6160B	DEFSCL	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
180	00B	68	5514B	ORDERM	81/08/14.	08.43.31	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
181	00B	88	2744B	ORDER	81/09/15.	23.07.43	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
182	00B	155	5613B	READN	81/09/15.	23.07.43	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
183	00B	128	3526B	WRITEQ	81/09/15.	23.07.43	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
184	00B	492	0627B	I4VAF	81/10/26.	20.56.06	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
185	00B	233	1241B	STATSTC	81/10/26.	20.56.06	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
186	00B	346	2750B	I4VCE	81/10/26.	20.56.06	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
187	00B	173	1617B	STATSTB	81/10/26.	20.56.06	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=1
188	00B	256	6440B	LSTTP3	82/08/22.	14.09.14	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=1
189	00B	136	5342B	TAKNWSP	81/10/26.	23.56.54	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
190	00B	307	7071B	MONDUR	82/02/04.	20.41.11	NOS 1.4	FTN	4.8508	6464	I	PROGRAM OPT=1
191	00B	840	6633B	VELERR	82/07/31.	22.14.10	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=1
192	00B	365	1501B	ERRTF	82/07/11.	20.08.24	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=1
193	00B	50	2000B	ERRC	82/07/11.	20.08.24	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=1
194	00B	271	1713B	MVRL	82/07/12.	08.58.01	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=1
195	00B	194	7300B	LISTH	82/07/12.	10.48.11	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=1
196	00B	211	6610B	MKBND1	82/07/12.	11.05.39	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=1
197	00B	660	7700B	EFFSPWN	82/09/24.	11.38.27	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=1
198	00B	74	4043B	PAGEA	82/09/24.	11.38.27	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=1
199	00B	607	3257B	SPCIES	82/12/13.	23.30.07	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
200	00B	373	1231B	SLICE	83/08/29.	20.24.32	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
201	00B	1731	3501B	INITAL	83/05/10.	13.54.18	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
202	00B	86	5626B	HNOSV	83/05/04.	21.35.00	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
203	00B	69	3662B	WFTEST	83/05/04.	21.35.00	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
204	00B	283	6545B	IFG4	83/05/07.	06.47.43	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
205	00B	1428	7341B	CALIB	83/08/29.	19.10.45	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
206	00B	208	1764B	NVQB	83/05/07.	06.47.43	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
207	00B	521	7405B	PROCQ	83/08/31.	18.53.03	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
208	00B	632	3642B	INITIAL	83/05/21.	09.29.42	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINEOPT=2
209	00B	459	0452B	MAPA1	80/05/25.	13.16.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2 ROUND=+-*/ TRACE
210	00B	479	5273B	MAPA2	80/05/25.	13.16.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2 ROUND=+-*/ TRACE
211	00B	438	2441B	MAPA3	80/05/25.	13.16.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2 ROUND=+-*/ TRACE
212	00B	197	0616B	MAPA4	80/05/25.	13.16.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2 ROUND=+-*/ TRACE
213	00B	57	0163B	MAPA5	80/05/25.	13.16.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2 ROUND=+-*/ TRACE
214	00B	38	0524B	MAPA6	80/05/25.	13.17.36.	NOS 1.4	COMPASS	3.6508	2		
215	00B	230	2503B	MAPA7	80/05/25.	13.16.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2 ROUND=+-*/ TRACE
216	00B	86	6443B	TRNC	80/06/20.	08.20.28	NOS 1.4	FTN	4.8508	6464	I	FUNCTION OPT=1
217	00B	606	0755B	MAPA	80/05/25.	13.16.15	NOS 1.4	FTN	4.8508	6464	I	SUBROUTINEOPT=2 ROUND=+-*/ TRACE
218	00B	558	7543B	LPTP8C	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
219	00B	371	4302B	LPTTP8	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
220	00B	399	3154B	LPTCRV	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
221	00B	1034	0310B	LPTQHA	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
222	00B	1188	0321B	LPTDURA	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
223	00B	1210	2502B	LPTDUR	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
224	00B	931	3116B	LPTDAN	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
225	00B	708	4540B	LPTSAN	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
226	00B	412	7614B	LPTQM	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
227	00B	449	2303B	LPTP8A	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2
228	00B	228	7653B	LPTVCE	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM OPT=2

RECORD LEVEL LENGTH CHECKSUM PREFIX TABLE

FILE 1 ITEM, PHABSIM.

229	00B	228	0204B	LPTQVF	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
230	00B	176	6614B	READF	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
231	00B	96	4564B	WRITEAQ	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
232	00B	190	7710B	DURSTAT	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	SUBROUTINE	OPT=2
233	00B	755	1024B	LPTTSN	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
234	00B	518	6365B	MAKTP6	83/05/24.	23.41.46	NOS 1.4	FTN	4.8552	6464	I	PROGRAM	OPT=2
235	00B	484	2361B	PHABSIM	83/09/05.								
236	17B	0	0000B										
		92328	7327B										

TOTAL

92328	7327B ✓

APPENDIX 3

UW Academic Computer Center Document N85
PF2TAPE/TAPE2PF

Chapter 1

Introduction

The purpose of this manual is to describe how to use the Academic Computer Center file archiving systems, PF2TAPE and TAPE2PF. PF2TAPE transfers data or programs from permanent file to magnetic tape and TAPE2PF does just the reverse.

The Academic Computer Center recommends that you have backups for any important data, programs, procedures, or files. Using the programs PF2TAPE and TAPE2PF, you can create and maintain backup files on magnetic tapes.

Another use of PF2TAPE/TAPE2PF is to keep infrequently used data or programs on tape instead of paying file storage costs to keep them online. This allows you to save money and the Computer Center to manage the permanent file space more efficiently and keep our costs of operation down.

The programs PF2TAPE and TAPE2PF and this manual are based on programs and documentation received from the Indiana University Wrubel Computing Center. The documentation received from Indiana was prepared by: D. Forman, P. Glenn and J. Hettmer. This Academic Computer Center document is taken almost exactly from the Reference Manual used at Oregon State University Computer Center.

Chapter 2

How PF2TAPE and TAPE2PF Work

When a file is archived using PF2TAPE, it is written as a single record on a permanent storage device such as a magnetic tape. The PF2TAPE library file thus created is called NEW. PF2TAPE makes the record on NEW look like an ABS (absolute) record, but the structure and content of the data are actually unchanged (see Chapter 6 for details).

If you already have a PF2TAPE library (OLD), and are adding new files to it, PF2TAPE first writes the new files on NEW as described above. PF2TAPE then copies records from the OLD library to the NEW library. Any archived file from OLD which has the same name as an archived file already on NEW is not copied. Thus PF2TAPE will retain only one version of each file.

TAPE2PF uses the ABS record on the PF2TAPE library to reconstruct an exact copy of the original file, which is then left as a local file.

NOTE:

1. TAPE2PF does not automatically save retrieved files as permanent files, unless the R parameter is used.

2. A local file that has the same name as a file being retrieved is lost.

The form of the PF2TAPE and TAPE2PF commands is:

PF2TAPE,parameters.

TAPE2PF,parameters.

The parameters are listed in Chapters 7 and 8.

The use of defaults is discussed in Chapter 6.

Chapter 3

PF2TAPE Examples

A. A sample job to create a PF2TAPE library:

You wish to archive your permanent files TEST, CREATEU, and PROC (these are all the permanent files that you have). You do not have an OLD PF2TAPE library file.

INPUT:

```
LABEL,NEW,VSN=nnnnnn,PO=W,NT,D=PE,W.
PF2TAPE,P=0.
```

OUTPUT:

```
1              PF2TAPE    78/09/26     08.14.37

     ACTION    NAME          DATE          TIME     TYPE

   ARCHIVED*  CREATEU     78/09/26.    08.14.19.PFILE
   ARCHIVED*  PROC        78/09/26.    08.14.19.PFILE
   ARCHIVED*  TEST        78/09/26.    08.14.19.PFILE

            END OF PF2TAPE LISTING.
  PF2TAPE COMPLETE.
```

This example would be used only to create a new PF2TAPE library because the P=0 option on the PF2TAPE statement specifies that an OLD backup library does not exist. The NEW PF2TAPE library is written onto a backup tape (VSN=nnnnnn). PF2TAPE default parameters specify that all of your permanent files, but none of the local files, be archived on to the NEW backup library. The file type, PFILE, indicates a permanent file; and the date and time displayed is when the PF2TAPE was done.

B. A sample command sequence to update a PF2TAPE library:

You wish to add two more permanent files, TEST2 and MINE, to your PF2TAPE library.

INPUT:

```
RESOURCE,PE=2.                Two 9-track tape drives will be used
LABEL,OLD,VSN=nnnnn1,PO=R,NT,D=PE,R.
LABEL,NEW,VSN=nnnnn2,PO=W,NT,D=PE,W.
PF2TAPE,PF=TEST2/MINE.
```

OUTPUT:

```
1              PF2TAPE    78/09/26  09.32.46

   ACTION     NAME          DATE      TIME     TYPE

   ARCHIVED*  MINE        78/09/26. 09.32.46.PFILE
   ARCHIVED*  TEST2       78/09/26. 09.32.46.PFILE
   COPIED *   CREATEU     78/09/26. 08.14.37.PFILE
   COPIED *   PROC        78/09/26. 08.14.37.PFILE
   COPIED *   TEST        78/09/26. 08.14.37.PFILE

              END OF PF2TAPE LISTING
  PF2TAPE COMPLETE.
```

In this example two new permanent files (TEST2 and MINE) were archived and then the contents of the OLD library were copied to the NEW library.

Unless additional parameters specify otherwise, PF2TAPE combines all of the files on OLD with all of the user's permanent files, creating a file named NEW which contains only the most recent versions of all the files.

C. Archiving updated copies of permanent files:

You wish to replace your archived files (TEST, CREATEU, MINE) with new versions.

INPUT:

```
RESOURCE,PE=2.              Two 9-track tape drives will be used
LABEL,OLD,VSN=nnnnn1,PO=R,NT,D=PE,R.
LABEL,NEW,VSN=nnnnn2,PO=W,NT,D=PE.
PF2TAPE,PF=TEST/CREATEU/MINE.
```

OUTPUT:

```
1              PF2TAPE    78/09/27  13.24.56.
   ACTION     NAME          DATE      TIME     TYPE

   ARCHIVED*  CREATEU     78/09/27. 13.24.56.PFILE
   ARCHIVED*  PROC        78/09/27. 13.24.56.PFILE
   ARCHIVED*  TEST        78/09/27. 13.24.56.PFILE
   DELETED    CREATEU     78/09/26. 08.14.19.PFILE
   DELETED    PROC        78/09/26. 08.14.19.PFILE
   DELETED    TEST        78/09/26. 08.14.19.PFILE

              END OF PF2TAPE LISTING
  PF2TAPE COMPLETE.
```

This example illustrates the archiving of permanent files that were previously archived onto an OLD library. The OLD library in this example is the PF2TAPE library created in Example A. The permanent files were archived and then all files on the OLD library that have names matching those just archived are "deleted" from the NEW library by not copying them forward from the OLD library onto the NEW library.

Chapter 4

TAPE2PF Examples

A. A sample command sequence to retrieve four files from a PF2TAPE library and use them as local files:

INPUT:

```
LABEL,OLD,VSN=nnnnnn,PO=R,NT,D=PE.
TAPE2PF,RF=QWERT/A/X/TAPE1.
```

OUTPUT:

```
RETRIEVING QWERT
RETRIEVING A
RETRIEVING X
RETRIEVING TAPE1
          4 FILES RETRIEVED.
RETRIEVE COMPLETE.
```

This job retrieves files QWERT, A, X, and TAPE1 from the PF2TAPE library tape. Default TAPE2PF parameters specify that these files be retrieved from PF2TAPE file OLD and then rewound. Files QWERT, A, X, and TAPE1 are local files only and must be SAVEd or DEFINEd by the user in order to be made permanent.

B. A sample command sequence to retrieve four files from a PF2TAPE library and save two of them as permanent files:

INPUT:

```
LABEL,OLD,VSN=nnnnnn,PO=R,NT,D=PE.
DEFINE,QWERT.
TAPE2PF,RF=QWERT/A/X/TAPE1.
```

OUTPUT:

```
RETRIEVING QWERT
RETRIEVING A
RETRIEVING X
RETRIEVING TAPE1
          4 FILES RETRIEVED.
RETRIEVE COMPLETE.
SAVE,TAPE1.
```

This example is the same as Example A except that retrieved file QWERT will be retained as a direct access permanent file and retrieved file TAPE1 is SAVEd as an indirect access permanent file.

Chapter 5

More Examples

A. A sample job to catalog a PF2TAPE tape and list the names of the files stored on the tape:

INPUT:

```
LABEL,OLD,VSN=nnnnnn,PO=R,NT,D=PE.
CATALOG,OLD,N,R.
```

OUTPUT:

```
          CATALOG OF OLD                    FILE      1
   REC    NAME       TYPE          LENGTH     CKSUM      DATE

     1    TEST2      ABS             1760      5721     78/09/26.
     2    MINE       ABS               77      6602     78/09/26.
     3    TEST       ABS              115      3633     78/09/26.
     4    CREATEU    ABS             1351      1225     78/09/26.
     5    PROC       ABS              124      7446     78/09/26.

          * EOI *          SUM =    61406
CATALOG COMPLETE.
```

The program CATALOG is used to list the archived files on an OLD library. Files will appear as type ABS which is the result of archiving the files and is of no concern to you. The date listed for each file will be the date that the file was archived. The N parameter specifies a CATALOG to end-of-information. The R parameter specifies that file OLD be rewound before and after the CATALOG. CATALOG is described in the NOS Reference Manual, Volume 1.

B. A sample job to archive both permanent and local files and to update an old PF2TAPE library:

INPUT:

```
RESOURC,PE=2.                Two 9-track tape drives are requested.
LABEL,OLD,VSN=nnnnn1,PO=R,NT,D=PE.
LABEL,NEW,VSN=nnnnn2,PO=W,NT,D=PE.
PF2TAPE,PF-BIGFILE,LF=TAPE1/TAPE2.
```

OUTPUT:

```
1            PF2TAPE    78/09/27.  11.24.45.

    ACTION     NAME         DATE        TIME     TYPE

    ARCHIVED*  BINPROG    78/09/27.  11.24.45.PFILE
    ARCHIVED*  SOURCE     78/09/27.  11.24.45.PFILE
    ARCHIVED*  TAPE1      78/09/27.  11.24.45.LOCAL FILE
    ARCHIVED*  TAPE2      78/09/27.  11.24.45.LOCAL FILE
    COPIED  *  CREATEU    78/09/26.  09.32.46.PFILE
    COPIED  *  MINE       78/09/26.  09.32.46.PFILE
    COPIED  *  PROC       78/09/26.  09.32.46.PFILE
    COPIED  *  TEST       78/09/26.  09.32.46.PFILE
    DELETED    BIGFILE    78/09/26.  09.32.46.PFILE
```

This job archives all permanent files except BIGFILE. The only local files archived are TAPE1 and TAPE2.

The contents of the OLD library are then transferred to the library NEW. As the contents of OLD are transferred, any files that have the same name as files that were archived in this run are dropped.

Note that OLD is not changed by this PF2TAPE run, only NEW contains the results of the PF2TAPE run.

Chapter 6

Recommended Procedures and Possible Pitfalls

A. Using PF2TAPE defaults

By simply using the PF2TAPE command with no parameters, you will:

1. Archive all permanent files (but no local files) onto a new library.
2. Transfer all other files from the OLD library to the NEW library.
3. Get a listing of all files archived onto the NEW library. The examples in Chapters 3 and 4 contain sample output from PF2TAPE runs.
4. Leave both the OLD and NEW libraries rewound.

B. Using TAPE2PF Defaults

The control statement TAPE2PF causes all archived files (maximum of 200) to be retrieved from library file OLD and created as local files with the same name as the archived files. The file OLD and all retrieved files are rewound. Unless you want to retrieve all files on the library OLD, we recommend:

```
     TAPE2PF,RF=RF1/RF2/RFN.
or
     TAPE2PF,RF-RF1/RF2/RFN).
```

(See Chapter 8 for more details.)

C. Warnings and Restrictions

When you wish to PF2TAPE your permanent files, make sure your local files do not have the same names as your permanent files; if you ask for archiving of permanent files and local files having the same name, only the local files will be archived. The permanent files will be ignored.

Local files that are archived are returned after they have been processed.

The following local files will not be archived: INPUT, OUTPUT, LGO, PUNCH, PUNCHB, TAPE, OLD, NEW, SCR, SCR1, SCR2, SCR3, OLDPL, OPL and all file names beginning with ZZZZZ. These files are protected from local file archiving because they tend to be created without any explicit action. These files will not be archived when you ask to have all local files archived. If you wish to archive one of these files then you must use the RENAME statement to change its name;

e.g.,

```
RENAME,BINPROG=LGO.      This statement will change the name
                           of the LGO file to BINPROG
```

Local files described with the P, I, L and N parameters on the PF2TAPE control statement will be protected from archiving. Any magnetic tape resident local file or any file assigned to the terminal will also be protected from archiving.

TAPE2PF has a maximum limit of 200 files. If more than 200 files are specified TAPE2PF will stop after the two-hundredth file.

Remember that TAPE2PF does not automatically save retrieved files as permanent files.

When a file is archived, it is prefixed with a loader table to make the resultant archived record look like an ABS record. This "formatting" permits records other than PF2TAPE files to be stored on your PF2TAPE library. Note that the creation of the ABS record by PF2TAPE does not affect the contents of any record or file you might have archived, including any ABS records or files.

WARNING:

If you use LIBEDIT to alter the comments field of any PF2TAPE ABS record, you might make all of the information stored in that record unrecognizable to PF2TAPE or TAPE2PF.

Chapter 7

PF2TAPE Parameter List

1. Selecting Permanent Files for PF2TAPE

ALLPF (default)

This instructs PF2TAPE to archive all permanent files. To archive individual files see the PF= parameter.

PF=file1/file2/file3/.../filen

Specifies that only the named permanent files are to be archived.

PF-file1/file2/file3/.../filen

Specifies that all permanent files EXCEPT those named are to be archived.

NOPF

Specifies that NO permanent files are to be archived. This is the same as PF= with no files named.

2. Selecting Local Files for PF2TAPE

ALLLF

This instructs PF2TAPE to archive all local files. To archive individual files, see the LF= parameter.

LF=file1/file2/file3/.../filen

Specifies that only the named local files be archived.

LF-file1/file2/file3/.../filen

Specifies that all local files except those named be archived.

NOLF (default)

Specifies that NO local files be archived.

WARNINGS ABOUT LOCAL FILES AND PF2TAPE

(A) Any permanent files to be archived will be made local first, which means that any local files by the same name will be lost.

(B) Any local files to be archived will be returned as soon as they have been archived, so they will be lost.

(C) See additional warnings in Chapter 6.

3. Deleting Unwanted Files from a PF2TAPE Library

DF=file1/file2/file3/.../filen

Causes the named files to be omitted from the new library, rather than being copied over from the old library when the new one is being built. The default is to carry all old files forward forever.

4. Selecting Old and New Library Names

P=filename (default: P=OLD)

The named file is used as the old library, from which the new one will be constructed.

P=0 (zero)

Tells PF2TAPE that there is no old library, as is the case on a first (creation) run.

N=filename (default: N=NEW)

The new library will be placed on the named file.

5. Miscellaneous PF2TAPE Options

L=filename (default: L=OUTPUT)

Name of the file to receive the output requested by the LO=F parameter.

LIST

When used with no other parameters (except P, L, I, or LO), LIST produces a listing of all files on the archive file without generating a new archive file.

LO=F

This instructs PF2TAPE to use CATLIST format on the listing specified by the L= parameter. Appendix A contains a sample of the listing produced when LO=F is used.

NR

No rewind. If this parameter is present, PF2TAPE will not rewind the OLD or NEW library file. Default is to rewind both OLD and NEW before and after the PF2TAPE run.

I=filename

Used to designate an input file containing further PF2TAPE parameters. If this parameter is not used, all parameters must appear on the PF2TAPE control statement. The single character I is used to specify the file INPUT. If the parameters are entered from a file, the contents of the file should follow the same formats as if the parameters were being entered on the PF2TAPE statement.

Chapter 8

TAPE2PF Parameter List

1. Selecting Files to be Retrieved from a PF2TAPE Library

RF=file1/file2/file3/.../filen

Causes the named files to be extracted from the PF2TAPE library and left as local files. Remember, they will not automatically be made permanent; you must do this yourself.

RF-file1/file2/file3/.../filen

Causes all files in the entire library (except those named) to be retrieved.

Note that RF= with no files named will retrieve nothing, and RF- with no files named (the default) will retrieve the first 200 files in the library, which can be very time consuming. Do not do this unless you really mean it.

2. Error Control During the TAPE2PF Operation

A

This parameter instructs TAPE2PF to abort if either no files were retrieved or not all files specified with the RF= parameter were retrieved.

IGNORE

Instructs TAPE2PF to ignore unrecognizable records on the old file. If this parameter is not present, TAPE2PF will abort if the OLD file is not in PF2TAPE format. The IGNORE parameter allows you to store records other than PF2TAPE files on the PF2TAPE library.

3. Miscellaneous TAPE2PF Options

P=filename (default: P=OLD)

Used to specify the name of the old library file.

L=filename (default: L=OUTPUT)

Used to specify the file to receive the output requested when the LO=F option is selected. Appendix B contains a sample listing produced when the L= option is used without

the LO=F option.

LO=F

This instructs TAPE2PF to use CATLIST format on the listing specified by the L= parameter. Default is no listing. Appendix C contains a sample listing produced when the LO=F option is used.

NR

No rewind. If this parameter is present, TAPE2PF will not rewind the OLD file, or any retrieved files. Normally, OLD and all retrieved files are automatically rewound by TAPE2PF.

R

Causes all retrieved files to be saved as direct or indirect access permanent files, depending on their state at the time of archiving. Old versions of these files will be lost and all are returned.

Chapter 9

Error Messages

The following are possible error messages that can occur while using PF2TAPE and TAPE2PF. The text following each message explains what caused the message to appear.

PF2TAPE COMPLETE

Self-explanatory.

ABORT OPTION CHOSEN, TAPE2PF ABORTED

The A option was selected on the TAPE2PF statement, errors have occurred and the run was aborted.

BAD FILE NAME IN RECORD NAMED x

The ABS loader table in the record named x on the OLD library file did not contain a valid file name. Run aborted.

BAD FORMAT ON LIBRARY TAPE FILE

The magnetic tape file specified as the OLD library was unreadable by PF2TAPE. The tape must be in NOS internal format.

DUPLICATE PARAMETER

Only one occurrence of each argument is allowed. Run aborted.

MORE THAN 128 LOCAL FILES, ERROR.

Attempting to archive too many files. Return enough local files to reduce the total count of local files to 128 or less and rerun PF2TAPE. The PF2TAPE run was partially completed but should be rerun.

NULL ARGUMENT AFTER x

The specified argument x must be equated to some value or name. Run aborted.

NOT ALL RF FILES WERE ENCOUNTERED

Some of the files specified with the RF argument were not found during the run. All files that were found on the PF2TAPE library were processed.

NO FILES WERE RETRIEVED

None of the requested files were retrieved from the OLD library because none of the file names were found on the library.

NO/ALL NOT LEGAL WITH =/- TYPE PARAMS

The prefix NO or ALL cannot be used at the same time as = or -. For example, ALLPF=QWERT/BIGFILE is illegal.

TAPE2PF FILE LIMIT HAS BEEN REACHED

A maximum of 200 files may be retrieved by one TAPE2PF statement.

TAPE2PF COMPLETE

Self-explanatory.

SEE CONSULTANTS -ACC- 543-5227 (1)

Internal error in PF2TAPE/TAPE2PF. Notify the consultants immediately because of possible system errors.

SEE CONSULTANTS -ACC- 543-5227 (2)

Internal error in PF2TAPE/TAPE2PF. Notify the consultants immediately because of possible system error.

TOO MANY NAMES FOR ONE PARAMETER AFTER x

Self-explanatory. The last accepted parameter was the parameter x as listed in the error message.

UNKNOWN FIELD AFTER x

An unknown parameter was entered. Run aborted.

UNRECOGNIZED INPUT

PF2TAPE/TAPE2PF cannot understand the information provided on the input file.

UNRECOGNIZED RECORD

PF2TAPE/TAPE2PF could not understand a record read from the OLD library. Either the library is bad or the IGNORE option was not specified and non-PF2TAPE files exist on the library. Run aborted.

n FILES RETRIEVED

Self-explanatory.

n FILES RETRIEVED FROM TAPE

Self-explanatory.

n LOCAL FILES ARCHIVED

Self-explanatory.

n PFILES ARCHIVED

n permanent files were archived.

n RECORDS COPIED FROM OLD

Self-explanatory.

** WARNING ** BUSY PFILE WAS SKIPPED.

The permanent file listed in the PF2TAPE output immediately prior to this message was busy (in use for writing or modification by this or some other job). The file was not archived.

** WARNING ** PFILE WAS ARCHIVED AS LOCAL filename

A local file, "filename," existed with the same name as a permanent file tnat was to be archived. The local file was archived in place of the permanent file. The file name is listed in the PF2TAPE output message immediately prior to this message.

Appendix A

Sample Output of PF2TAPE(LO=F,P=0)

```
1        PF2TAPE    78/10/02. 13.19.13.

    FILE NAME ACCESS FILE-TYPE LENGTH DN CREATION LAST ACCESS LAST MOD
          ACTION   INDEX  PERM. SUBSYS DATE/TIME DATE/TIME DATE/TIME

    1 PROC     IND. PRIVATE         1    77/10/25. 78/09/01. 77/10/27.
       ARCHIVED*          WRITE         12.47.22. 06.03.27. 13.34.21.

    2 MEMO     IND. PRIVATE         3    78/01/03. 78/09/01. 78/01/03.
       ARCHIVED*          WRITE         08.32.45. 06.05.38. 10.43.34.

    3 LSCAN    IND. PRIVATE        71    78/09/01. 78/09/26. 78/09/01.
       ARCHIVED*          WRITE         14.10.38. 08.03.13. 14.10.38.

          END OF PF2TAPE LISTING.
```

Appendix B

Sample Listing Produced by TAPE2PF(L=filename,RF=LSCAN...

```
1          TAPE2PF      78/10/02. 13.26.08.

  ACTION    NAME          DATE         TIME    TYPE

 SKIPPED    PROC       78/10/02. 13.19.13.PFILE
 SKIPPED    MEMO       78/10/02. 13.19.13.PFILE
 RETRIEVE*  LSCAN      78/10/02. 13.19.13.PFILE

            END OF TAPE2PF LISTING.
```

Appendix C

Sample Listing Produced by TAPE2PF(LO=F,RF=LSCAN...

```
1          TAPE2PF     78/10/02. 13.29.04.

    FILE NAME ACCESS FILE-TYPE LENGTH DN CREATION LAST ACCESS LAST MOD
              ACTION   INDEX  PERM. SUBSYS DATE/TIME DATE/TIME DATE/TIME

    1 PROC      IND. PRIVATE          1    77/10/25. 78/09/01. 77/10/27.
           SKIPPED           WRITE        12.47.22. 06.03.27. 13.34.21.

    2 MEMO      IND. PRIVATE          3    78/01/03. 78/09/01. 78/01/03.
           SKIPPED           WRITE        06.32.45. 06.05.38. 10.43.34.

    3 LSCAN     IND. PRIVATE         71    78/09/01. 78/09/26. 78/09/01.
         RETRIEVE*           WRITE        14.10.38. 08.03.13. 14.10.38.

          END OF TAPE2PF LISTING.
```

0.18.22.43UL, 1.621, , FBSM000, HV, 00