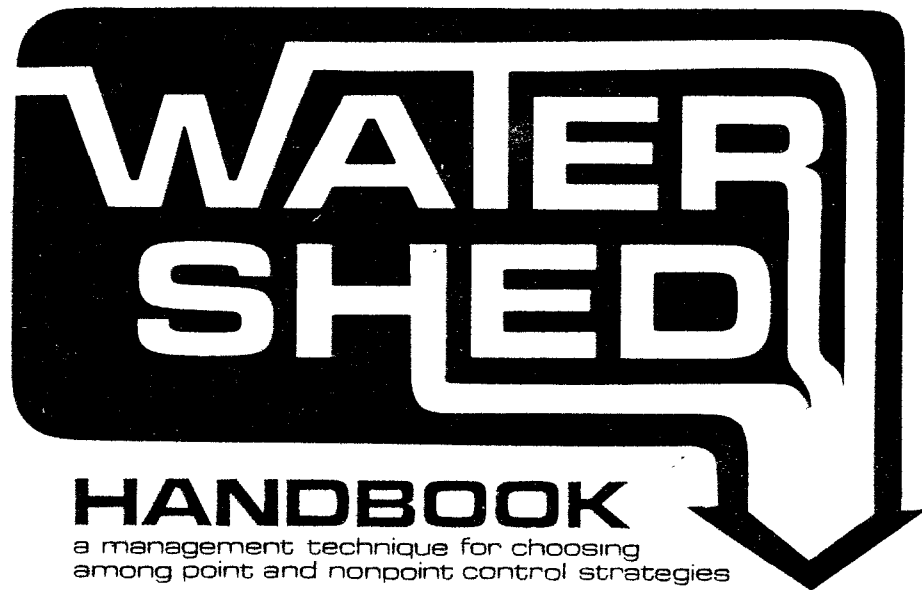
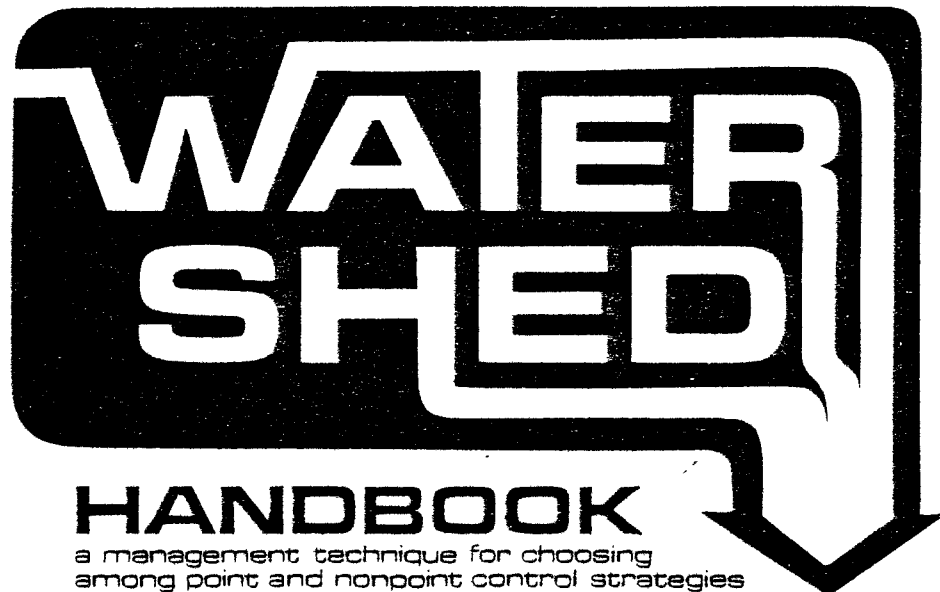




Appendix G
Users Manual for WATERSHED Computer Program



Appendix G

Users Manual for WATERSHED Computer Program

By

Michael R. Donihue
Consultant

Timothy J. Monteith
Great Lakes Basin Commission Staff

August 1981

Prepared for
U.S. Environmental Protection Agency
Great Lakes National Program Office
536 South Clark Street, Room 932
Chicago, Illinois 60605
(312)353-2117

DISCLAIMER

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

ACKNOWLEDGEMENTS

The authors wish to thank the Great Lakes National Program Office Staff and Don Urban of the USDA Soil Conservation Service for their support and guidance throughout this project. We also wish to acknowledge Marie Murrell for the secretarial support.

PROJECT OFFICERS

Great Lakes National Program Office
Kent Fuller

Great Lakes Basin Commission
Timothy Monteith

TABLE OF CONTENTS

	<u>PAGE NO.</u>
LIST OF TABLES	3
LIST OF FIGURES	5
INTRODUCTION	7
SECTION 1 - MAIN WATERSHED COMPUTER PROGRAM	9
Overview of Commands	9
Command Details	11
Sample Terminal Session	16
Sample Worksheet Output	23
Program Flow Chart and Listing	33
Data Files	94
SECTION 2 - NETWORK PROGRAM	101
Commands	101
Sample Terminal Session	101
Computer Flow Chart and Listing	103
Program Output	113

LIST OF TABLES

<u>NUMBER</u>		<u>PAGE NO.</u>
1	Legal Commands for WATERSHED Program	9
2	Legal Commands for NETWORK Program	101

LIST OF FIGURES

<u>NUMBER</u>		<u>PAGE NO.</u>
1	Flow of Commands for Typical Program Run	10
2	WATERSHED Program Flow Chart	34
3	NETWORK Program Flow Chart	104

APPENDIX G

USERS MANUAL FOR WATERSHED COMPUTER PROGRAM

INTRODUCTION

This computer program was developed to assist the user in manipulating WATERSHED worksheet data. The program is particularly useful when analyzing a network of river basins or when making a number of changes in the original assumptions.

In order to use this program properly the user must have a good understanding of the basic WATERSHED process presented in the main handbook. There is no substitution for using the worksheets presented in the handbook to organize the initial assumptions. Once the data requirements of the worksheets have been met the computer program can be implemented to assist with calculations.

The program is written in Fortran and is runnable on most medium to large size computers. The output format necessitates a wide carriage terminal or full 132 character printer. The data input and manipulation is done interactively and may be accomplished on any type of computer terminal. The program utilizes a "command" based system that requires the user be familiar with the 14 commands, legal key words, and option lists available for driving the program. Although the program prompts the user at many points for specific information, the user should still become familiar with the program commands presented in the following chapter.

There are actually two programs that make up the full WATERSHED computerized system. The first is called WATERSHED. This program performs all operations of the WATERSHED process for one river basin. The second is a sub-program called NETWORK. This program reads in the output data from WATERSHED for up to 7 different river basins and produces a Worksheet 8 for the entire network. This manual is divided into two sections. Section 1 describes the operation of the main WATERSHED program and Section 2 describes the NETWORK program.

SECTION 1 - MAIN WATERSHED COMPUTER PROGRAM

Overview of Commands

Each user must determine how their computer calls compiles, and runs a Fortran program and how such a program interacts with their hardware. The input and output subroutines utilize numbers to identify various data files. The number 8 is used for the input file associated with the READ command. The number 9 is used for the outputting of data, (other than final Worksheets) in the SAVE command. Final worksheets will default to the terminal unless routed to a data file identified by the proper number with the WORKSHEET command.

The full list of acceptable commands is shown in Table 1. As can be seen many of the commands have keywords that are required to modify the command. Once the computer run begins the program will prompt "enter command" and then wait for further instruction. The user must then select one of the 14 commands from table 1 to properly respond to this prompt.

TABLE 1

Legal Commands for WATERSHED Programs

CONTROL	Keyword (optionlist)
COST	Keyword
XCOST	Keyword
HELP	
INPUT	Keyword
LOAD	Keyword
PERCENT	Keyword
XPERCENT	Keyword
READ	Keyword
SAVE	Keyword
STOP	
TITLE	(value) Title list
TRANSMISSION	(position number)
WORKSHEET	(file number) Worksheet number

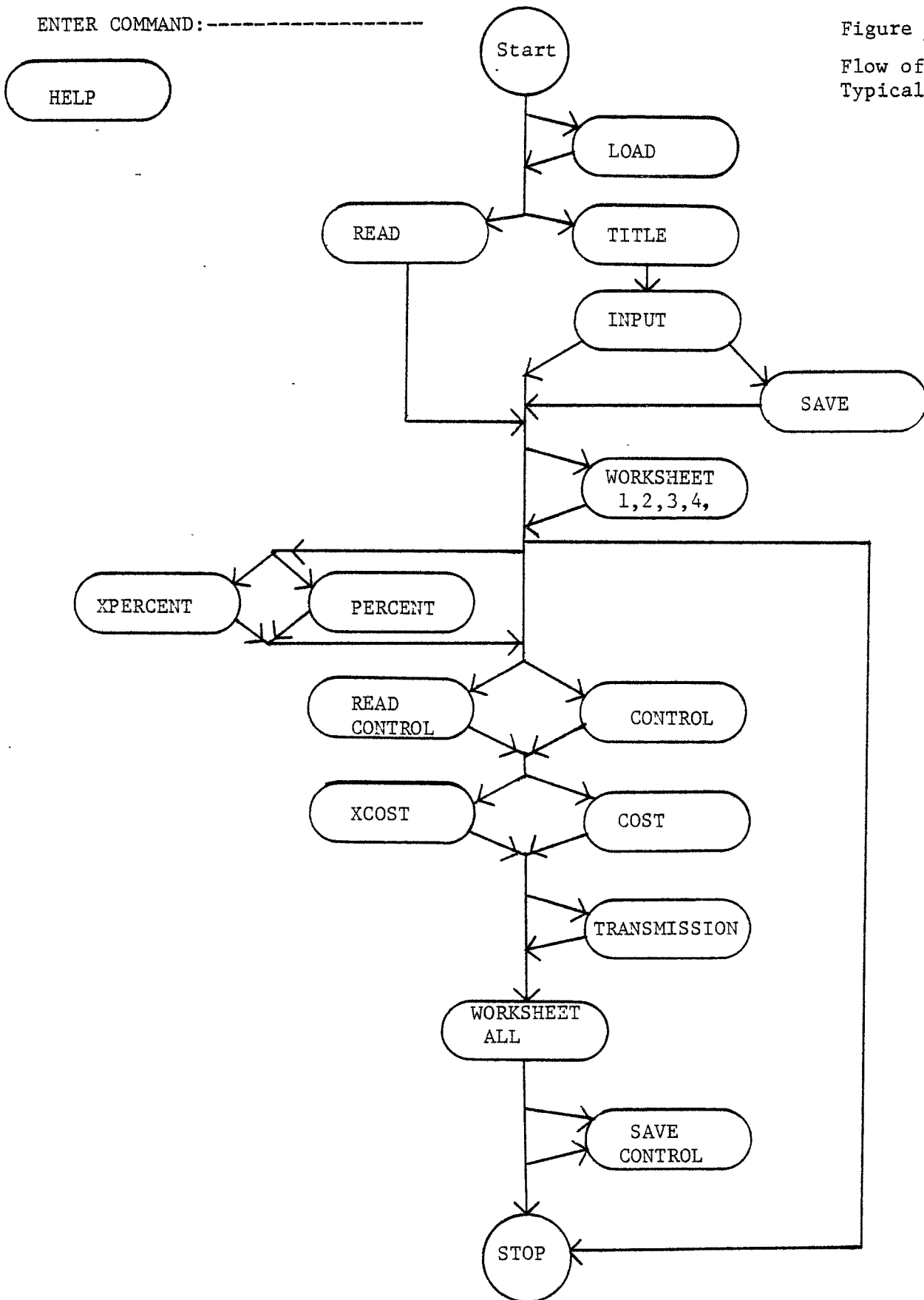
Many of the commands depend upon the completion of operations carried out under other commands on the list. Figure 1 shows a flow of commands for a typical WATERSHED run. Once the computer run has begun, HELP and STOP may be used at any point. STOP should be used with caution because data can be lost if it has not been properly saved. As can be seen in Figure 1, the use of the LOAD command is optional although it should be activated early in the process. The user may then use the READ command to bring in data from the previous run or may proceed through the TITLE, INPUT, and SAVE commands to build new Worksheets 1 through 4. The data may then be displayed using the WORKSHEET command to check for errors before moving on to the pollutant control options. It should be noted that the only way errors can be corrected is by STOPping the program after the SAVE command and editing the output file or starting over by re-entering all of the data.

Figure 1 shows the optional PERCENT and XPERCENT commands which may be used before moving into the actual CONTROL options. Again controls may be

ENTER COMMAND:-----

Figure 1

Flow of Commands for
Typical Program Run



put in from the keyboard or read from a file. The cost of the controls are then entered (COST or XCOST) and transmissions may be modified with the TRANSMISSION command. Worksheets can be printed to the terminal or data file using the WORKSHEET ALL command and, the control options may be saved.

Command Details

The units used in this program are the same as those displayed on the worksheets in the WATERSHED handbook. Where prompting for specific units does not occur, worksheet units should be used. As previously mentioned, some commands must be preceded by other commands in the list. Figure 1 provides a general guideline although more detail can be found in this section.

CONTROL keyword (optionlist)

The CONTROL command accepts input of control strategy data. Each command for the respective keywords increments the stage of controls for which values are to be entered. No more than three stages of controls may be specified. Each keyword has a set of optional arguments which may be specified within parenthesis in the command line following the keyword. Multiple options separated by commas within the set of parenthesis are accepted. The user is then prompted for the values of the controls specified. If no options are specified appropriate defaults are used.

keyword - URBAN for the input of point and urban non-point control strategies. If no options are specified the default optionlist is -- (FLOW,CONC,AREA,UAL).

(optionlist) - any of the following are valid options to identify particular source values for controls.

FLOW - to control flow values for point sources

CONC - to control concentrations for point sources

AREA - to control area values for all non-point sources

AREA/sourcenam - to control area values for a specific source (e.g. AREA/STORM or AREA/COMBINED)

UAL - to control unit area loads for all non-point sources

UAL/sourcenam - to control unit area loads for a specific source

LOAD - to control load values directly for all urban sources.
Note: this is the default if the LOAD URBAN command has been given.

keyword - RURAL to input control values for rural cropland sources. The default value of the optionlist includes the five USLE coefficients (R,K,LS,C,P).

(optionlist) - any of the following are valid options

R,K,LS,C,P - any combination of the USLE coefficients may be specified using these identifiers.

EFF - to alter the efficiency ratio for a particular source. The default value for all sources is 1.0

UAL - to alter unit area loads if USLE coefficients are not used. Note: this is the default if the LOAD CROPLAND(UAL) command has been given.

LOAD - to control load values directly for all cropland sources. Note: this is the default if the LOAD CROPLAND command has been given.

COST keyword

Accepts input on the cost of controls. Each command for the respective keyword increments the stage number for which the command applies. Multiple keywords separated by commas may be used in a single command. The user is prompted for the appropriate costs of control. ² Note: The default sets costs the same within any source group i.e. \$1000/km² to reduce stormwater runoff loads by 10% in all cities.

keyword - valid keywords are URBAN and RURAL corresponding to costs of control for urban point and non-point sources and rural cropland sources, respectively.

XCOST keyword

This command is identical to the COST command with the important distinction that costs may differ within source groups. The user is prompted for "per unit costs of control" for each source. Valid keywords are the same as those above.

HELP

This command prints a brief list and description of the valid WATERSHED commands.

INPUT keyword

This command prompts the user for values for the initial state of the WATERSHED area of study. Various keywords may be specified and multiple keywords separated by commas are accepted. The default if no keyword is specified is SOURCE.

keyword - SOURCE

prompts the user for type code, source name, point of entry character, area or units served, and optional surface features. Must be specified before additional keywords will be accepted.

The following specifications are prompted for:
type code - an integer value equal to one of the following codes:

- 1 - for rural non-cropland sources
- 2 - for rural cropland sources
- 3 - for municipal (point) sources

- 4 - for industrial (point) sources
- 12 - is a valid type code causing the corresponding source name to apply to both a noncropland and a cropland source.
- 21 - the same as 12 except cropland is prompted before noncropland.

source name - an alphanumeric identifier for each source of not more than 28 characters.

A blank carriage return signifies the end of source input.

point of entry code - a single character, alphanumeric, marking the position of the source in the WATERSHED area of study.

area or units served - is a real (floating point) number of the source area in square kilometers or the number of units for point sources (e.g. households) served by a particular source.

surface features - is an optional alphanumeric descriptor of the source. Up to 12 characters are allowed.

keyword - URBAN

To input flow (mgd) and concentration (mg/L) data for point sources and unit area loads ($\text{kg}/\text{km}^2/\text{yr}$) for non-point urban sources. All values must be entered as real (floating point) numbers. Default urban areas are point, storm sewer and combined sewer. Two additional areas such as construction or unsewered may be specified by the TITLE command.

keyword - NONCROPLAND

To enter area (km^2) and unit area loads ($\text{kg}/\text{km}^2/\text{yr}$) for noncropland rural areas. All values must be entered as real (floating point) numbers. Default areas are grassland and woodland. Three other area names may be optionally specified by the TITLE command.

keyword - CROPLAND

To enter values for either USLE coefficients, unit area loads ($\text{kg}/\text{km}^2/\text{yr}$), or total loads for rural cropland areas, (the latter two when the LOAD command has been used for cropland). This is also where the pollutant delivery ratio is entered or calculated based on the total rivermouth load. Again, all values must be entered as real numbers.

LOAD keyword

This command is used to cause WATERSHED to be run using load figures directly rather than unit area loads and USLE coefficients. This command must be activated early in the run (see Figure 1). Multiple keywords separated by commas are accepted.

keyword - URBAN

for point and urban non-point sources

NONCROPLAND
for rural non-cropland sources

CROPLAND
for rural cropland sources

CROPLAND(UAL)
to load unit area loads for rural cropland sources rather
than USLE coefficients.

PERCENT keyword

Used if control values to be entered are in terms of percent reduction from the initial state to stage I or between subsequent stages of control. User is prompted following the CONTROL command for percent reduction for the appropriate sources. Values entered must be in decimal form (i.e. 75% is entered as .75) Multiple keywords separated by commas are accepted. Note: reductions apply to all sources within a keyword type.

keywords: URBAN or RURAL

XPERCENT keyword

Identical to the PERCENT command but percent reductions are not assumed to be the same for all sources within a keyword. Following the CONTROL command the user is prompted for percent reductions for each source.

READ keyword

This command reads from a WATERSHED data file input values which were SAVED during a previous program run. File number 8 must be identified as the data file when the program is run.

keyword - is optional. If omitted only values for the initial state are read in. The only valid keyword is CONTROL and causes the values for the various control strategies to be read in.

SAVE keyword

This command writes to a data file the existing data from a WATERSHED computer session. File number 9 must be identified as the worksheet data file when the program is run.

keyword - is optional. If omitted only values for the initial state are stored. The only valid keyword is CONTROL and causes the values for the various control strategies to be stored.

STOP keyword

This command terminates the WATERSHED computer program. NOTE: If this command is used before the SAVE command, all data generated in that run will be lost.

TITLE(value) titlelist

Enables the user to specify up to four titles identifying the WATERSHED computer session output as a heading for Worksheet 1. Also enables the user to specify optional, other urban, or rural noncropland sources. The actual title or name follows the value in parenthesis. Maximum title length is 40 characters and maximum source name length is 12 characters.

- (value) - may be an integer from 1 through 4 to indicate the title(s).
- (U1) - to specify one other urban area (such as unsewered)
- (U2) - to specify a second other urban area, named in titlelist.
- (R1) - to specify one other rural noncropland area besides the woodland and grassland default.
- (R2) - to specify a second other rural noncropland area.
- (R3) - to specify a third other rural noncropland area.

TRANSMISSION(n)

This command is used to alter the effective transmissions for each source. The value in parenthesis is the integer position number of the source. The user is then prompted for the value of the transmission. If "n" is not specified, the user is prompted for the effective transmissions of all sources.

This is the one major difference between this program and the Handbook. In this program the TRANSMISSION command would be used to modify effective transmission and bioavailability as shown in Worksheet 7 in the Handbook. Because the Handbook columns a_7 and b_7 are always multiplied together, the computer program eliminates the bioavailable column b_7 . This variable can still be factored in by altering the transmission value accordingly. For example, an initial T of 0.9 (column a_7) and bioavailable fraction of 0.5 (column b_7) would result in a Load to the Mouth (column d_7) of 0.45 times the load to the surface water (column c_7). In this program an initial TRANSMISSION would be entered as 0.45 rather than 0.9.

WORKSHEET (m) n

This command controls the final output of the WATERSHED sub-routines. The full worksheet tables are outputted according to worksheet numbers as specified in the manual.

- (m) - is an optional output file number to which the tables will be output. The default is to output to the terminal.

- n - is a worksheet number 1 through 8. Multiple worksheets separated by commas are also accepted.
- ALL - is a valid keyword to output all available worksheets.

Sample Terminal Session

The following terminal session provides an example of how a WATERSHED analysis can be run using the WATERSHED computer program. The data were obtained from the Sandusky river basin in north central Ohio for example purposes only. These input data and the resulting WATERSHED worksheets do not necessarily represent actual conditions in this river basin.

In this example run a large Amdahl computer was used running the Michigan Terminal System (MTS). The data file WATER.TMP contained input information and was assigned to unit number 8. Data file WATER.DAT was used to save new input data and was assigned to unit number 9. Finally, data file -OUTFIL was a temporary storage area for the final worksheet forms, assigned to unit number 3. All prompts from the computer program are indicated by a triangle in the left hand margin. Lines without a triangle were entered by the user on this sample run.

▷ #EXECUTION BEGINS

7

Δ

△

△

2

Δ

△

Δ

A

Δ

A

Δ

A

Δ

A

△

A

Δ

8

Δ

三

△

33

△

8

7

10

△

10

Δ

8

> NON-CROPLAND TYMOCHTEE CREEK
 C,56.17
 > CROPLAND TYMOCHTEE CREEK
 C,528.4
 > NON-CROPLAND MIDDLE SANDUSKY
 C,85.23
 > CROPLAND MIDDLE SANDUSKY
 C,542.36
 > NON-CROPLAND HONEY CREEK
 D,47.24
 > CROPLAND HONEY CREEK
 D,343.55
 > MUNICIPAL TIFFIN CITY
 D,26000.
 > - STORM
 D,7.0
 > - COMBINED
 D,10.4
 > NON-CROPLAND WOLF CREEK
 E,30.46
 > CROPLAND WOLF CREEK
 E,343.11
 > NON-CROPLAND ROCK CREEK
 E,68.51
 > CROPLAND ROCK CREEK
 E,409.9
 > MUNICIPAL FREMONT CITY
 E,19730.
 > - STORM
 E,1.4
 > - COMBINED
 E,12.3
 > NON-CROPLAND LOWER SANDUSKY (SOIL 1)
 F,84.54
 > CROPLAND LOWER SANDUSKY (SOIL 1)
 F,243.55
 > NON-CROPLAND LOWER SANDUSKY (SOIL 2)
 F,8.25
 > CROPLAND LOWER SANDUSKY (SOIL 2)
 F,100.0

> INPUT FOR WORKSHEET 2 - POINT & URBAN RUNOFF LOADS

> FOR POINT SOURCES: ENTER FLOW(MGD),CONCENTRATION(MG/L)
 > FOR NON-POINT SOURCES: ENTER UNIT AREA LOADS(KG/KM2/YR)

> BUCYRUS CITY
 2.5,4.0
 > STORM
 250.
 > COMBINED
 900.
 > UPPER SANDUSKY CITY
 1.5,4.0
 > STORM
 0.0
 > COMBINED
 250.
 > TIFFIN CITY
 3.2,4.0
 > STORM
 250.
 > COMBINED
 1000.

▷ FREMONT CITY
 5.1,4.0
 ▷ STORM
 250.
 ▷ COMBINED
 1000.
 ▷ INPUT FOR WORKSHEET 3 - RURAL NON-CROPLAND AREAS
 ▷ ENTER AREA,UAL FOR: GRASSLAND,WOODLAND
 ▷ LOSS CREEK
 11.93,10.,25.04,10.
 ▷ BROKEN SWORD
 3.59,10.,18.17,10.
 ▷ UPPER SANDUSKY RIVER
 12.81,10.,34.38,10.
 ▷ TYMOCHTEE CREEK
 9.04,25.,47.13,10.
 ▷ MIDDLE SANDUSKY
 17.27,10.,57.96,10.
 ▷ HONEY CREEK
 3.17,10.,44.07,10.
 ▷ WOLF CREEK
 7.57,10.,22.89,10.
 ▷ ROCK CREEK
 12.14,10.,56.47,10.
 ▷ LOWER SANDUSKY (SOIL 1)
 10.0,25.,74.54,10.
 ▷ LOWER SANDUSKY (SOIL 2)
 8.25,10.,0.,10.
 ▷ INPUT FOR WORKSHEET 4 - RURAL CROPLAND AREAS
 ▷ IS A POLLUTANT DELIVERY RATIO KNOWN FOR EACH SOURCE ?
 NO
 ▷ ENTER TOTAL POLLUTANT LOAD AT MOUTH OF BASIN(KG/YR)
 421000.
 ▷ ENTER USLE COEFFICIENTS R,K,LS,C,P FOR EACH SOURCE
 ▷ LOSS CREEK
 125.,.35,.402,.233,1.
 ▷ BROKEN SWORD
 125.,.38,.424,.233,1.
 ▷ UPPER SANDUSKY RIVER
 130.,.42,.426,.245,1.
 ▷ TYMOCHTEE CREEK
 138.,.32,.357,.290,1.
 ▷ MIDDLE SANDUSKY
 138.,.38,.381,.260,1.
 ▷ HONEY CREEK
 125.,.35,.338,.237,1.
 ▷ WOLF CREEK
 125.,.29,.256,.237,1.
 ▷ ROCK CREEK
 125.,.34,.427,.237,1.
 ▷ LOWER SANDUSKY (SOIL 1)
 125.,.32,.410,.268,1.
 ▷ LOWER SANDUSKY (SOIL 2)
 125.,.28,.434,.237,1.
 ▷ ENTER COMMAND
 SAVE

▷ ENTER COMMAND
PERCENT URBAN

▷ ENTER COMMAND
CONTROL URBAN(CONC,UAL/STORM,UBL/COMBINED)

▷ STAGE 1 CONTROL STRATEGIES FOR URBAN SOURCES

▷ ENTER PERCENT REDUCTIONS FOR SOURCE CONC
▷ IN DECIMAL FORM (E.G. ENTER 25% AS .25)
.75

▷ ENTER PERCENT REDUCTIONS FOR NONPOINT STORM SEWER UAL
▷ IN DECIMAL FORM (E.G. ENTER 25% AS .25)
.25

▷ ENTER PERCENT REDUCTIONS FOR NONPOINT COMBINED UAL
▷ IN DECIMAL FORM (E.G. ENTER 25% AS .25)
.06

▷ ENTER COMMAND
CON URB(CON,UAL/STO,UAL/COM)

▷ STAGE 2 CONTROL STRATEGIES FOR URBAN SOURCES

▷ ENTER PERCENT REDUCTIONS FOR SOURCE CONC
▷ IN DECIMAL FORM (E.G. ENTER 25% AS .25)
.50

▷ ENTER PERCENT REDUCTIONS FOR NONPOINT STORM SEWER UAL
▷ IN DECIMAL FORM (E.G. ENTER 25% AS .25)
.25

▷ ENTER PERCENT REDUCTIONS FOR NONPOINT COMBINED UAL
▷ IN DECIMAL FORM (E.G. ENTER 25% AS .25)
.25

```

> ENTER COMMAND
  CONTROL RURAL(C)

>      STAGE 1 CONTROL STRATEGIES FOR RURAL SOURCES

> ENTER CONTROL VALUES PER SOURCE FOR C ,
> LOSS CREEK
  .108
> BROKEN SWORD
  .099
> UPPER SANDUSKY RIVER
  .103
> TYMOCHTEE CREEK
  .108
> MIDDLE SANDUSKY
  .110
> HONEY CREEK
  .101
> WOLF CREEK
  .108
> ROCK CREEK
  .105
> LOWER SANDUSKY (SOIL 1)
  .118
> LOWER SANDUSKY (SOIL 2)
  .099

> ENTER COMMAND
  CON RUR(C)

>      STAGE 2 CONTROL STRATEGIES FOR RURAL SOURCES

> ENTER CONTROL VALUES PER SOURCE FOR C ,
> LOSS CREEK
  .034
> BROKEN SWORD
  .035
> UPPER SANDUSKY RIVER
  .034
> TYMOCHTEE CREEK
  .032
> MIDDLE SANDUSKY
  .035
> HONEY CREEK
  .034
> WOLF CREEK
  .032
> ROCK CREEK
  .033
> LOWER SANDUSKY (SOIL 1)
  .036
> LOWER SANDUSKY (SOIL 2)
  .033

> ENTER COMMAND
  COST URBAN,RURAL,URBAN,RURAL

>      STAGE 1 COSTS OF CONTROL FOR URBAN SOURCES

> ENTER PER UNIT COSTS FOR POINT,STORM,COMBINED SOURCES
  2.4,7400.,7400.

```

```

> STAGE 1 COSTS OF CONTROL FOR RURAL SOURCES
> ENTER PER UNIT COSTS FOR RURAL SOURCES
  65.

> STAGE 2 COSTS OF CONTROL FOR URBAN SOURCES
> ENTER PER UNIT COSTS FOR POINT,STORM,COMBINED SOURCES
  3.6,8600.,24700.

> STAGE 2 COSTS OF CONTROL FOR RURAL SOURCES
> ENTER PER UNIT COSTS FOR RURAL SOURCES
  3000.

> ENTER COMMAND
  WORKSHEET(3) ALL

> ENTER COMMAND
  SAVE CONTROL

> ENTER COMMAND
  STOP
> #EXECUTION TERMINATED

```

Sample Worksheet Output

Once the command WORKSHEET(3) ALL is given, all final worksheet forms are written to the data file on unit number 3. An example of the printed output from this data file follows. These output sheets do not necessarily correspond to the example terminal session, although most of the names and data are the same.

SANDUSKY RIVER

PHYSICAL LAYOUT

LAKE ERIE

WORKSHEET 1

SOURCE	TYPE	POSITION	POINT OF ENTRY	AREA KM2	UNITS SERVED	SURFACE FEATURES
LOSS CREEK	NON-CROPLAND	1	A	36.9		
LOSS CREEK	CROPLAND	2	A	189.8		
BUCYRUS CITY	MUNICIPAL	3	A		13500.	
STORM		4	A	1.3		
COMBINED		5	A	7.2		
BROKEN SWORD	NON-CROPLAND	6	B	21.8		
BROKEN SWORD	CROPLAND	7	B	187.2		
UPPER SANDUSKY RIVER	NON-CROPLAND	8	B	47.2		
UPPER SANDUSKY RIVER	CROPLAND	9	B	274.6		
UPPER SANDUSKY CITY	MUNICIPAL	10	B		6250.	
STORM		11	B	0.0		
COMBINED		12	B	6.7		
TYMOCHTEE CREEK	NON-CROPLAND	13	C	56.2		
TYMOCHTEE CREEK	CROPLAND	14	C	528.4		
MIDDLE SANDUSKY	NON-CROPLAND	15	C	85.2		
MIDDLE SANDUSKY	CROPLAND	16	C	542.4		
HONEY CREEK	NON-CROPLAND	17	D	47.2		
HONEY CREEK	CROPLAND	18	D	343.6		
TIFFIN CITY	MUNICIPAL	19	D		26000.	
STORM		20	D	7.0		
COMBINED		21	D	10.4		
WOLF CREEK	NON-CROPLAND	22	E	30.5		
WOLF CREEK	CROPLAND	23	E	343.1		
ROCK CREEK	NON-CROPLAND	24	E	68.6		
ROCK CREEK	CROPLAND	25	F	409.9		
FREMONT CITY	MUNICIPAL	26	E		19730.	
STORM		27	E	1.4		
COMBINED		28	E	12.3		
LOWER SANDUSKY (SOIL 1)	NON-CROPLAND	29	F	84.5		
LOWER SANDUSKY (SOIL 1)	CROPLAND	30	F	243.6		
LOWER SANDUSKY (SOIL 2)	NON-CROPLAND	31	F	8.3		
LOWER SANDUSKY (SOIL 2)	CROPLAND	32	F	100.0		
TOTAL				3695.1	65480.	

POINT AND URBAN RUNOFF LOADS
TOTAL POLLUTANT LOADS

WORK SHEET 2

SOURCE	POSITION	FLOW MGD	POINT CONC. MG/L	LOAD KG/YR	STORM			COMBINED		
					AREA KM2	UAL KG/KM2/YR	LOAD KG/YR	AREA KM2	UAL KG/KM2/YR	LOAD KG/YR
INITIAL CONDITION										
BUCYRUS CITY	3	2.5	4.0	13820.	1.3	250.0	325.	7.2	900.0	5480.
UPPER SANDUSKY CITY	10	1.5	4.0	8292.	0.0	0.0	0.	6.7	250.0	1675.
TIFFIN CITY	19	3.2	4.0	17690.	7.0	250.0	1750.	10.4	1000.0	10400.
FREMONT CITY	26	5.1	4.0	28193.	1.4	250.0	350.	12.3	1000.0	12300.
TOTAL				67994.			2425.			30855.
STAGE 1										
BUCYRUS CITY	3	2.5	1.0	3455.	1.3	187.5	244.	7.2	846.0	6091.
UPPER SANDUSKY CITY	10	1.5	1.0	2073.	0.0	-0.0	0.	6.7	235.0	1574.
TIFFIN CITY	19	3.2	1.0	4422.	7.0	187.5	1313.	10.4	940.0	9776.
FREMONT CITY	26	5.1	1.0	7048.	1.4	187.5	262.	12.3	940.0	11562.
TOTAL				16999.			1819.			29004.
STAGE 2										
BUCYRUS CITY	3	2.5	0.5	1728.	1.3	140.6	183.	7.2	534.5	4569.
UPPER SANDUSKY CITY	10	1.5	0.5	1037.	0.0	-0.0	0.	6.7	176.3	1181.
TIFFIN CITY	19	3.2	0.5	2211.	7.0	140.6	984.	10.4	705.0	7332.
FREMONT CITY	26	5.1	0.5	3524.	1.4	140.6	197.	12.3	705.0	8672.
TOTAL				8499.			1364.			21753.

RURAL NON-CROPLAND AREAS
TOTAL POLLUTANT LOADS

WORKSHEET 3

SOURCE	POSITION	GRASSLAND			WOODLAND			TOTAL LOAD KG/YR
		AREA KM2	UAL KG/KM2/YR	LOAD KG/YR	AREA KM2	UAL KG/KM2/YR	LOAD KG/YR	
LOSS CREEK	1	11.8	10.0	118.3	25.0	10.0	250.4	368.7
BRCKEN SWORD	6	3.6	10.0	35.9	18.2	10.0	181.7	217.6
UPPER SANDUSKY RIVER	8	12.8	10.0	128.1	34.4	10.0	343.8	471.9
TYMOCHTEE CREEK	13	9.0	25.0	226.0	47.1	10.0	471.3	597.3
MIDDLE SANDUSKY	15	17.3	10.0	172.7	68.0	10.0	679.6	852.3
HONEY CREEK	17	3.2	10.0	31.7	44.1	10.0	440.7	472.4
WOLF CREEK	22	7.6	10.0	75.7	22.9	10.0	228.9	304.6
ROCK CREEK	24	12.1	10.0	121.4	56.5	10.0	564.7	686.1
LOWER SANDUSKY (SOIL 1)	29	10.0	25.0	250.0	74.5	10.0	745.4	995.4
LOWER SANDUSKY (SOIL 2)	31	8.3	10.0	82.5	0.0	10.0	0.0	82.5
TOTAL		95.7		390.6	1242.3		3906.5	5148.8

RURAL CROPLAND AREAS
UNIVERSAL SOIL LOSS EQUATION/TOTAL POLLUTANT LOAD ESTIMATES

WORK SHEET 4

SOURCE	POSITION	CROPLAND AREA (HA)	R	USLF	COEFFICIENTS	LS	C	P	SOIL LOSS T/AC/YR	SOIL LOSS MT/HA/YR	TOTAL EROSION MT/YR	DELIVERY RATIO	REDUCTION EFFICIENCY	TOTAL LOAD MT/YR
INITIAL CONDITION														
LOSS CREEK	2	18984.	125.	0.35	0.402	0.233	1.00		4.1	9.2	174492.	0.000105		18.3
BROKEN SWORD	7	18719.	125.	0.38	0.424	0.233	1.00		4.7	10.5	197027.	0.000105		20.7
UPPER SANDUSKY RIVER	9	27460.	130.	0.42	0.426	0.245	1.00		5.7	12.8	350992.	0.000105		36.9
TYMOCHTEE CREEK	14	52840.	138.	0.32	0.357	0.260	1.00		4.1	9.2	485805.	0.000105		51.0
MIDDLE SANDUSKY	16	54236.	138.	0.38	0.381	0.260	1.00		5.2	11.7	631942.	0.000105		66.4
HONEY CREEK	18	34355.	125.	0.35	0.338	0.237	1.00		3.5	7.9	270061.	0.000105		28.4
WOLF CREEK	23	34311.	125.	0.29	0.256	0.237	1.00		2.2	4.9	169262.	0.000105		17.8
ROCK CREEK	25	40990.	125.	0.34	0.427	0.237	1.00		4.3	9.6	395432.	0.000105		41.5
LOWER SANDUSKY (SOIL 1)	30	24355.	125.	0.32	0.410	0.268	1.00		4.4	9.9	240102.	0.000105		25.2
LOWER SANDUSKY (SOIL 2)	32	10000.	125.	0.28	0.434	0.237	1.00		3.6	8.1	80749.	0.000105		8.5
TOTAL		316250.							41.8	93.7	2995861.	0.000105	NA	314.6
STAGE 1														
LOSS CREEK	2	18984.	125.	0.35	0.402	0.108	1.00		1.9	4.3	90881.	0.000105	1.00	8.5
BROKEN SWORD	7	18719.	125.	0.38	0.424	0.099	1.00		2.0	4.5	83715.	0.000105	1.00	8.8
UPPER SANDUSKY RIVER	9	27460.	130.	0.42	0.426	0.103	1.00		2.4	5.4	147560.	0.000105	1.00	15.5
TYMOCHTEE CREEK	14	52840.	138.	0.32	0.357	0.108	1.00		1.7	3.8	201796.	0.000105	1.00	21.2
MIDDLE SANDUSKY	16	54236.	138.	0.38	0.381	0.110	1.00		2.2	4.9	267360.	0.000105	1.00	28.1
HONEY CREEK	18	34355.	125.	0.35	0.338	0.101	1.00		1.5	3.4	115089.	0.000105	1.00	12.1
WOLF CREEK	23	34311.	125.	0.29	0.256	0.108	1.00		1.0	2.2	77132.	0.000105	1.00	8.1
ROCK CREEK	25	40990.	125.	0.34	0.427	0.105	1.00		1.9	4.3	175191.	0.000105	1.00	18.4
LOWER SANDUSKY (SOIL 1)	30	24355.	125.	0.32	0.410	0.116	1.00		1.9	4.3	103925.	0.000105	1.00	10.9
LOWER SANDUSKY (SOIL 2)	32	10000.	125.	0.28	0.434	0.099	1.00		1.5	3.4	33730.	0.000105	1.00	3.5
TOTAL		316250.							18.0	40.4	1286378.	0.000105		135.1
STAGE 2														
LOSS CREEK	2	18984.	125.	0.35	0.402	0.034	1.00		0.5	1.3	25452.	0.000105	1.00	2.7
BROKEN SWORD	7	18719.	125.	0.38	0.424	0.035	1.00		0.7	1.6	29596.	0.000105	1.00	3.1
UPPER SANDUSKY RIVER	9	27460.	130.	0.42	0.426	0.034	1.00		0.8	1.8	48709.	0.000105	1.00	5.1
TYMOCHTEE CREEK	14	52840.	138.	0.32	0.357	0.032	1.00		0.5	1.1	59791.	0.000105	1.00	6.3
MIDDLE SANDUSKY	16	54236.	138.	0.38	0.381	0.035	1.00		0.7	1.6	85069.	0.000105	1.00	8.9
HONEY CREEK	18	34355.	125.	0.35	0.338	0.034	1.00		0.5	1.1	38743.	0.000105	1.00	4.1
WOLF CREEK	23	34311.	125.	0.29	0.256	0.032	1.00		0.3	0.7	22854.	0.000105	1.00	2.4
ROCK CREEK	25	40990.	125.	0.34	0.427	0.033	1.00		0.6	1.3	55060.	0.000105	1.00	5.8
LOWER SANDUSKY (SOIL 1)	30	24355.	125.	0.32	0.410	0.036	1.00		0.6	1.3	32252.	0.000105	1.00	3.4
LOWER SANDUSKY (SOIL 2)	32	10000.	125.	0.28	0.434	0.033	1.00		0.5	1.1	11243.	0.000105	1.00	1.2
TOTAL		316250.							5.9	13.0	408781.	0.000105		42.9

LOADING SUMMARY
TOTAL POLLUTANT LOADS

WORKSHEET 5

SOURCE	POSITION	INITIAL LOAD TO WATERSHED BASIN KG/YR	CONTROLLED LOAD KG/YR		LOAD REDUCTIONS KG/YR	
			STAGE 1	STAGE 2	STAGE 1	STAGE 2
LOSS CREEK	1	368.7				
CROPLAND	2	18321.7	8492.5	2673.6	9829.2	5818.9
BUCYRUS CITY	3	13820.0	3455.0	1727.5	10365.0	1727.5
STORM	4	325.0	243.8	182.8	81.2	60.9
COMBINED	5	6180.0	6091.2	4568.4	388.8	1522.8
BROKEN SWORD	6	217.6				
CROPLAND	7	20687.9	8790.1	3107.6	11897.8	5682.5
UPPER SANDUSKY RIVER	8	471.9				
CROPLAND	9	36854.2	15493.8	5114.5	21360.4	10379.3
UPPER SANDUSKY CITY	10	8292.0	2073.0	1036.5	6219.0	1036.5
STORM	11	0.0	0.0	0.0	0.0	0.0
COMBINED	12	1675.0	1574.5	1180.9	100.5	393.6
TYMOCHTEE CREEK	13	697.3				
CROPLAND	14	51009.5	21188.6	5279.1	29320.9	14910.5
MIDDLE SANDUSKY	15	952.3				
CROPLAND	16	66353.8	28072.8	8932.3	38281.0	19140.6
HONEY CREEK	17	472.4				
CROPLAND	18	20356.4	12084.4	4068.0	16272.0	8016.4
TIFFIN CITY	19	17689.6	4422.4	2211.2	13267.2	2211.2
STORM	20	1750.0	1312.5	984.4	437.5	328.1
COMBINED	21	10400.0	9776.0	7332.0	624.0	2444.0
WOLF CREEK	22	304.6				
CROPLAND	23	17772.5	8098.8	2399.7	9673.7	5699.2
ROCK CREEK	24	686.1				
CROPLAND	25	41520.3	18395.1	5791.3	23125.2	12513.8
FREMONT CITY	26	28192.8	7048.2	3524.1	21144.6	3524.1
STORM	27	350.0	262.5	196.9	87.5	65.6
COMBINED	28	12300.0	11562.0	8671.5	738.0	2990.5
LOWER SANDUSKY (SOIL 1)	29	995.4				
CROPLAND	30	25210.7	10912.1	3386.5	14298.6	7525.6
LOWER SANDUSKY (SOIL 2)	31	82.5				
CROPLAND	32	8478.6	3541.7	1180.6	4936.9	2361.1
TOTAL		420987.9	182890.6	74538.1	232948.3	108352.4

PROGRAM COSTS
TOTAL POLLUTANT REDUCTIONS

WORKSHEET 6

SOURCE	POSITION	AREA TREATED KM2	UNITS SERVED	PER UNIT STAGE 1	COSTS STAGE 2	TOTAL STAGE 1	COSTS STAGE 2
LOSS CREEK	2	189.8		65.	3000.	12340.	569520.
BUCYRUS CITY	3		13500.	2.	4.	32400.	48600.
STORM	4	1.3		7400.	8600.	9620.	11180.
COMBINED	5	7.2		7400.	24700.	53280.	177840.
BROKEN SWORD	7	187.2		65.	3000.	12167.	561570.
UPPER SANDUSKY RIVER	9	274.6		65.	3000.	17849.	823800.
UPPER SANDUSKY CITY	10		6250.	2.	4.	15000.	22500.
STORM	11	3.3		7400.	8600.	0.	0.
COMBINED	12	6.7		7400.	24700.	49590.	165490.
TYMOCHTEE CREEK	14	528.4		65.	3000.	34346.	1585199.
MIDDLE SANDUSKY	16	542.4		65.	3000.	35253.	1627080.
HONEY CREEK	18	343.6		65.	3000.	22331.	1030650.
TIFFIN CITY	19		26000.	2.	4.	62400.	93600.
STORM	20	7.0		7400.	8600.	51800.	60200.
COMBINED	21	10.4		7400.	24700.	76950.	256880.
WOLF CREEK	23	343.1		65.	3000.	22302.	1029330.
ROCK CREEK	25	409.9		65.	3000.	26643.	1229699.
FREMONT CITY	26		19730.	2.	4.	47352.	71028.
STORM	27	1.4		7400.	8600.	10360.	12040.
COMBINED	28	12.3		7400.	24700.	31020.	303810.
LOWER SANDUSKY (SOIL 1)	30	243.6		65.	3000.	15811.	730650.
LOWER SANDUSKY (SOIL 2)	32	100.0		65.	3000.	6500.	300000.
TOTAL						705334.	10710662.

COST-EFFECTIVENESS ANALYSIS
TOTAL POLLUTANT LOADS

WORKSHEET 7

SOURCE	POSITION	EFFECTIVE TRANSMISSION	BIO- AVAILABLE FRACTION	TOTAL LOAD TO SURFACE WATER (KG/YR)	LOAD AT MOUTH (KG/YR)	LOAD REDUCTION AT MOUTH (KG/YR)	COST OF PROGRAM (\$/YR)	COST	
								PER UNIT REMOVED AT MOUTH (\$/KG)	COST- EFFECTIVE RANK
STAGE 1						STAGE 1	STAGE 1	STAGE 1	
LOSS CREEK	1	1.0		369.	369.				
CROPLAND	2	1.0		18322.	18322.	9829.	12340.	1.3	7
BUCYRUS CITY	3	1.0		13820.	13820.	10365.	32400.	3.1	13
STORM	4	1.0		325.	325.	81.	9620.	118.4	31
COMBINED	5	1.0		6480.	6480.	389.	53280.	137.0	36
BROKEN SWORD	6	1.0		218.	218.				
CROPLAND	7	1.0		20688.	20688.	11898.	12167.	1.0	3
UPPER SANDUSKY RIVER	8	1.0		472.	472.				
CROPLAND	9	1.0		36854.	36854.	21360.	17849.	0.8	1
UPPER SANDUSKY CITY	10	1.0		8292.	9292.	5219.	15000.	2.4	12
STORM	11	1.0		0.	0.				
COMBINED	12	1.0		1675.	1675.	101.	49580.	493.3	42
TYMOCHTEE CREEK	13	1.0		697.	697.				
CROPLAND	14	1.0		51009.	51009.	29821.	34346.	1.2	5
MIDDLE SANDUSKY	15	1.0		852.	852.				
CROPLAND	16	1.0		66354.	66354.	38281.	35253.	0.9	2
HONEY CREEK	17	1.0		472.	472.				
CROPLAND	18	1.0		28356.	28356.	16272.	22331.	1.4	9
TIFFIN CITY	19	1.0		17690.	17690.	13267.	52400.	4.7	14
STORM	20	1.0		1750.	1750.	438.	51800.	118.4	30
COMBINED	21	1.0		10400.	10400.	624.	76960.	123.3	32
WOLF CREEK	22	1.0		305.	305.				
CROPLAND	23	1.0		17772.	17772.	9674.	22302.	2.3	11
ROCK CREEK	24	1.0		686.	686.				
CROPLAND	25	1.0		41520.	41520.	23125.	26643.	1.2	5
FREMONT CITY	26	1.0		28193.	28193.	21145.	47352.	2.2	10
STORM	27	1.0		350.	350.	88.	10360.	118.4	29
COMBINED	28	1.0		12300.	12300.	738.	91020.	123.3	33
LOWER SANDUSKY (SOIL 1)	29	1.0		995.	995.				
CROPLAND	30	1.0		25211.	25211.	14299.	15831.	1.1	4
LOWER SANDUSKY (SOIL 2)	31	1.0		83.	83.				
CROPLAND	32	1.0		8479.	8479.	4937.	6500.	1.3	8
SUBTOTAL						232949.			
TOTAL				420988.	420988.		705334.		

COST-EFFECTIVENESS ANALYSIS
TOTAL POLLUTANT LOADS

WORK SHEET 7

SOURCE	POSITION	EFFECTIVE TRANSMISSION	BIO-AVAILABLE FRACTION	TOTAL LOAD TO SURFACE WATER (KG/YR)	LOAD AT MOUTH (KG/YR)	LOAD REDUCTION AT MOUTH (KG/YR)	COST OF PROGRAM (\$/YR)	COST PER UNIT REMOVED AT MOUTH (\$/KG)	COST-EFFECTIVE RANK
STAGE 2						STAGE 2	STAGE 2	STAGE 2	
LOSS CREEK	1	1.0		369.	369.				
CROPLAND	2	1.0		18322.	18322.	5819.	569520.	97.9	23
BUCYRUS CITY	3	1.0		13820.	13820.	1728.	48600.	28.1	17
STORM	4	1.0		325.	325.	61.	11180.	183.5	38
COMBINED	5	1.0		6480.	5480.	1523.	177840.	116.8	28
BROKEN SWORD	6	1.0		218.	218.				
CROPLAND	7	1.0		20688.	20688.	5682.	561570.	98.8	24
UPPER SANDUSKY RIVER	8	1.0		472.	472.				
CROPLAND	9	1.0		36854.	36854.	10379.	823800.	79.4	19
UPPER SANDUSKY CITY	10	1.0		8292.	8292.	1037.	22500.	21.7	16
STORM	11	1.0		0.	0.				
COMBINED	12	1.0		1675.	1675.	394.	165490.	420.4	41
TYMOCHTEE CREEK	13	1.0		697.	697.				
CROPLAND	14	1.0		51009.	51009.	14910.	1585199.	106.3	27
MIDDLE SANDUSKY	15	1.0		852.	852.				
CROPLAND	16	1.0		66354.	65354.	19141.	1527080.	85.0	20
HONEY CREEK	17	1.0		472.	472.				
CROPLAND	18	1.0		28356.	28356.	8016.	1030650.	128.6	35
TIFFIN CITY	19	1.0		17690.	17690.	2211.	93600.	42.3	18
STORM	20	1.0		1750.	1750.	328.	60200.	183.5	39
COMBINED	21	1.0		10400.	10400.	2444.	256880.	105.1	25
WOLF CREEK	22	1.0		305.	305.				
CROPLAND	23	1.0		17772.	17772.	5699.	1029330.	180.6	37
ROCK CREEK	24	1.0		686.	686.				
CROPLAND	25	1.0		41520.	41520.	12614.	1229699.	97.5	22
FREMONT CITY	26	1.0		28193.	28193.	3524.	71028.	20.2	15
STORM	27	1.0		350.	350.	66.	12040.	183.5	40
COMBINED	28	1.0		12300.	12300.	2891.	303810.	105.1	26
LOWER SANDUSKY (SOIL 1)	29	1.0		995.	995.				
CROPLAND	30	1.0		25211.	25211.	7526.	730650.	97.1	21
LOWER SANDUSKY (SOIL 2)	31	1.0		83.	83.				
CROPLAND	32	1.0		8479.	8479.	2361.	300000.	127.1	34
SUBTOTAL						108352.			
TOTAL				420988.	420988.	341301.	10710662.		

SUMMARY OF PROGRAMS
TOTAL POLLUTANT LOADS

WORKSHEET 8

SOURCE	RANK	STAGE	LOAD REDUCTION (KG/YR)	SUM LOAD REDUCTION (KG/YR)	PERCENT REDUCTION	COST OF REDUCTION (\$/YR)	SUM OF REDUCTION COSTS (\$/YR)
CROPLAND - UPPER SANDUSKY RIVER	1	1	21360.	21360.	5.1	17849.	17849.
CROPLAND - MIDDLE SANDUSKY	2	1	38281.	59641.	14.2	35253.	53102.
CROPLAND - BROKEN SWORD	3	1	11898.	71539.	17.0	12167.	65270.
CROPLAND - LOWER SANDUSKY (SOIL 1)	4	1	14299.	85838.	20.4	15831.	81101.
CROPLAND - TYMOCHTEE CREEK	5	1	29821.	115659.	27.5	34346.	115446.
CROPLAND - ROCK CREEK	6	1	23125.	138784.	33.0	26643.	142090.
CROPLAND - LOSS CREEK	7	1	9829.	148613.	35.3	12340.	154429.
CROPLAND - LOWER SANDUSKY (SOIL 2)	8	1	4937.	153550.	36.5	5500.	160929.
CROPLAND - HONEY CREEK	9	1	16272.	169822.	40.3	22331.	183260.
MUNICIPAL - FREMONT CITY	10	1	21145.	190966.	45.4	47352.	230612.
CROPLAND - WOLF CREEK	11	1	9674.	200640.	47.7	22302.	252914.
MUNICIPAL - UPPER SANDUSKY CITY	12	1	6219.	206859.	49.1	15000.	267914.
MUNICIPAL - BUCYRUS CITY	13	1	10365.	217224.	51.6	32400.	300314.
MUNICIPAL - TIFFIN CITY	14	1	13267.	230491.	54.8	62400.	362714.
MUNICIPAL - FREMONT CITY	15	2	3524.	234015.	55.6	71028.	433742.
MUNICIPAL - UPPER SANDUSKY CITY	16	2	1037.	235052.	55.8	22500.	456242.
MUNICIPAL - BUCYRUS CITY	17	2	1728.	236779.	56.2	48600.	504842.
MUNICIPAL - TIFFIN CITY	18	2	2211.	238990.	56.8	93600.	598442.
CROPLAND - UPPER SANDUSKY RIVER	19	2	10379.	249370.	59.2	823800.	1422242.
CROPLAND - MIDDLE SANDUSKY	20	2	19141.	268510.	63.8	1627080.	3049322.
CROPLAND - LOWER SANDUSKY (SOIL 1)	21	2	7526.	276036.	65.6	730650.	3779972.
CROPLAND - ROCK CREEK	22	2	12614.	288650.	68.6	1229699.	5009671.
CROPLAND - LOSS CREEK	23	2	5819.	294468.	69.9	569520.	5579190.
CROPLAND - BROKEN SWORD	24	2	5682.	300151.	71.3	561570.	6140760.
COMBINED - TIFFIN CITY	25	2	2444.	302595.	71.9	256880.	6397639.
COMBINED - FREMONT CITY	26	2	2891.	305485.	72.6	303810.	6701449.
CROPLAND - TYMOCHTEE CREEK	27	2	14910.	320396.	76.1	1585199.	8286648.
COMBINED - BUCYRUS CITY	28	2	1523.	321919.	76.5	177840.	8464487.
STORM - FREMONT CITY	29	1	88.	322006.	76.5	10360.	8474846.
STORM - TIFFIN CITY	30	1	438.	322444.	76.6	51800.	8526646.
STORM - BUCYRUS CITY	31	1	81.	322525.	76.6	9620.	8536266.
COMBINED - TIFFIN CITY	32	1	624.	323149.	76.8	76960.	8613225.
COMBINED - FREMONT CITY	33	1	738.	323887.	76.9	91020.	8704245.
CROPLAND - LOWER SANDUSKY (SOIL 2)	34	2	2361.	326248.	77.5	300000.	9004245.
CROPLAND - HONEY CREEK	35	2	8016.	334264.	79.4	1030650.	10034895.
COMBINED - BUCYRUS CITY	36	1	389.	334653.	79.5	53280.	10088174.
CROPLAND - WOLF CREEK	37	2	5699.	340352.	80.8	1029330.	11117504.
STORM - BUCYRUS CITY	38	2	61.	340413.	80.9	11180.	11128684.
STORM - TIFFIN CITY	39	2	328.	340741.	80.9	60200.	11188884.
STORM - FREMONT CITY	40	2	66.	340807.	81.0	12040.	11200923.
COMBINED - UPPER SANDUSKY CITY	41	2	394.	341200.	81.0	165490.	11366412.
COMBINED - UPPER SANDUSKY CITY	42	1	101.	341301.	81.1	49580.	11415991.

Program Flow Chart and Listing

The flow chart shown in Figure 2 indicates that the program is built around a command interpreter. This interpreter screens all commands, checks their validity, and then activates the proper subroutine (indicated in script print). The fourteen legal commands are also shown in Figure 2.

The following program listings are organized by major subroutines indicated in Figure 2. Documentation has been included in the form of comments throughout the program code to assist in more detailed understanding of its operation. This code is fairly extensive so that the program would run in a simple manner for the users. Individuals interested in obtaining a tape of these programs, rather than entering all the lines from scratch, should contact Dr. William C. Sonzogni, at the Great Lakes Environmental Research Laboratory, NOAA, Department of Commerce, Ann Arbor, Michigan 48104, phone: 313/668-2249.

This main program has been broken up into 4 source files for ease in writing and editing. To compile the WATERSHED main program four files must be linked together. These four files are identified by the major subroutine in each file. Each file shows the line numbers starting with number "1". An example of this chain on the Michigan Terminal System is as follows:

```
$RUN *FTN PAR=S=WTRSHD+WRKSHT+WTRSAV+WTRCTL L=WATERSHED
```

```
where: *FTN calls the Fortran Compiler,  
       : WTRSHD, WRKSHT, WTRSAV, WTRCTL are the four surface files,  
       : and WATERSHED is the object file for the compiled program.
```

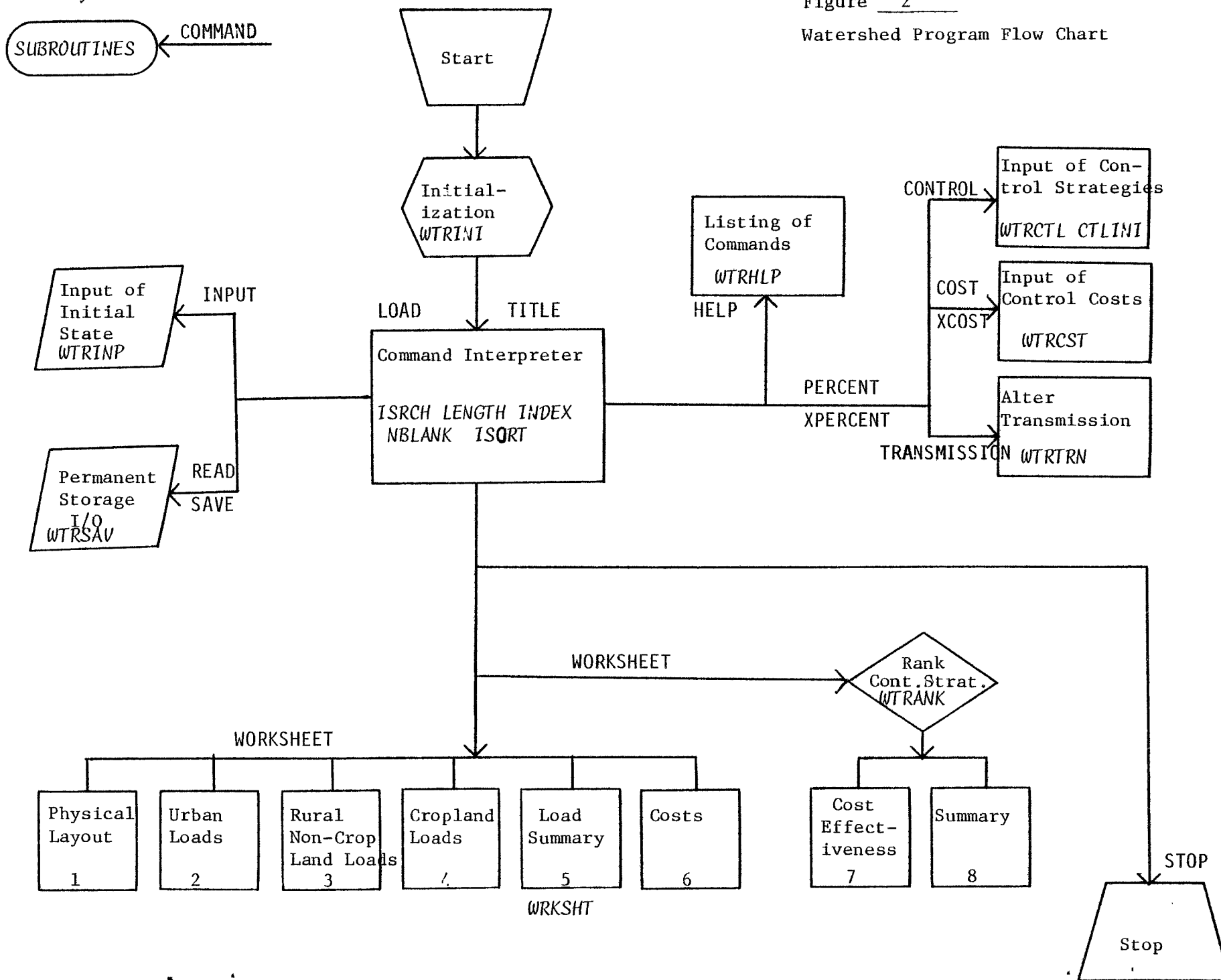
The start of the command and subrouting sections of the program are marked with a bracket and underline.

Key



Figure 2

Watershed Program Flow Chart



The first source file is called

WTRSHD

```

1 C
2 C MAIN PROGRAM FOR WATERSHED POLLUTION MANAGEMENT PACKAGE
3 C ACCEPTS COMMANDS AND CALLS SUBROUTINES RANKING
4 C COST-EFFECTIVENESS OF ALTERNATIVE CONTROL STRATEGIES
5 C
6 C WRITTEN FOR GREAT LAKES BASIN COMMISSION ANN ARBOR, MI
7 C JAN - JUN 1981 BY MICHAEL DONIHUE
8 C
9 C INTEGER CMD(80),BLNK,COMA,ALL,NJMWRK(10),OPTJRB(5),JPTRUL(8),
10 C OPTVAL(8),JVAL(3)
11 C *****
12 LOGICAL LINPUT,LODURB,LODRNC,LODRCP,L4UAL,LIN1,LIN2,LIN3,LIN4
13 LOGICAL URCTL,RULCTL,UCTLOD(3),RCTLOD(3),UPRCNT,UXPCNT,RPRCNT,
14 C RXPCNT,URBCST,RULCST,LPRANK
15 INTEGER OTHURB,OTHRUL,SRCE(210,28),TYPCOD(9,12)
16 DIMENSION ITYP(210),NPOS(210),AUNIT(210,2),KTITLE(4,40),
17 C SURFEA(210,3),CTL0D(210,3),CTUSLE(3,30,5)
18 DIMENSION STMCTL(30,3),STMAPA(30,3),COMCTL(30,3),COMARA(30,3),
19 C OT1CTL(30,3),OT1ARA(30,3),OT2CTL(30,3),OT2ARA(30,3)
20 DIMENSION PFLOW(30),PCONC(30),UALS(30),JALC(30),W2UAL1(30),
21 C W2UAL2(30),TRANSM(210)
22 INTEGER TLEUR1(12),TLEUR2(12),TLERU1(12),TLERU2(12),TLERU3(12),
23 C APOE(210),AREACN(4),UALCON(4)
24 DIMENSION GRASA(30),UALG(30),WLNOA(30),UALW(30),W3OTA1(30),
25 C W3UAL1(30),W3UAL2(30),W3UAL3(30),W3OTA2(30),W3OTA3(30)
26 REAL LODRNL(30),LODRN2(30),LODRW3(30),LODRW3(30),LODRW3(30)
27 DIMENSION USLER(30),USLEK(30),USLELS(30),USLEF(30),W4UAL(30),
28 C USLEP(30),SOILS1(30),SOILS2(30),TTLER(30),RATIO(30)
29 DIMENSION PDIF(210,3),TCOST(210,3),PUCOST(210,3)
30 DIMENSION PMOUTH(210),DIFMTH(210,3),CPURFM(210,3),KERANK(210,3)
31 COMMON /WTRIN/ IN,IOUT,NSRCE,KTITLE,NTLE,NURB,NRJVC,NRUCR
32 COMMON /WTRTC/ LINPUT,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHURB,
33 C OTHRUL,LODRCP,TYPCOD,JIN,JOUT,L4UAL,KOUT,LPRANK
34 COMMON /WRKSH1/ SRCE,ITYP,NPOS,APOE,AUNIT,SURFEA,TLEUR1,TLEUR2
35 COMMON /WRKSH2/ PFLOW,PCONC,UALS,UALC,W2UAL1,W2UAL2
36 COMMON /WRKSH3/ GRASA,UALG,WLNOA,UALW,LODRNL,LODRN2,
37 C W3OTA1,W3UAL1,W3UAL2,W3UAL3,W3OTA2,W3OTA3,
38 C TLERU1,TLERU2,TLERU3,LODRW3,LODRW3,LODRW3
39 COMMON /WRKSH4/ USLER,USLEK,USLELS,USLEF,USLEP,SOILS1,
40 C SOILS2,TTLER,W4UAL,RATIO
41 COMMON /WRKSH6/ PDIF,TCOST,PUCOST,TRANSM,PMOUTH,DIFMTH,CPURFM,
42 C KERANK
43 COMMON /WTRCON/ UCTLOD,RCTLOD,URCTL,RULCTL,UPRCNT,UXPCNT,RPRCNT,
44 C RXPCNT,OPTVAL,NOPT,URBCST,RULCST,AREACN,UALCON
45 DIMENSION CTLEFW(30,3),CTLCNC(30,3)
46 DIMENSION CTLEF(30,3),SOILC1(30,3),SOILC2(30,3),CTLEROI(30,3),
47 C CUALW4(30,3)
48 DIMENSION TOTCST(3),TOTREM(3),KNDX(540),FINCST(540),FINREM(540),
49 C ISFIN(540)
50 INTEGER FINSRC(540,43)
51 COMMON /WRKSH9/ TOTCST,TOTREM,TOTLOD,TOTMTH,KNDX,FINSRC,FINCST,
52 C FINREM,ISFIN,NUM
53 COMMON /UBNCTL/ CTLEFW,CTLCNC,CTLOD,STMCTL,STMAPA,COMCTL,
54 C COMARA,OT1CTL,OT1ARA,OT2CTL,OT2ARA
55 COMMON /RFOCTL/ CTUSLE,CTLEF,SOILC1,SOILC2,CTLEROI,CUALW4
56 C *****
57 DATA BLNK,COMA,ALL,NJMWRK/' ',' ',' ','A','1','2','3','4','5','6','7',
58 C '8','9','0'/'
59 DATA LPAREN,RPAREN,ISLASH,ITEE,IDD,MMM/'(',')','/','T','O','M'/'
60 DATA ISORCE,IUBAN,NCOPLAN,ICOPLAN,IRUAL/'S','U','N','C','R'/'

```

```

61 DATA OPTURB/'F','C','A','U','L'/
62 DATA OPTRUL/'R','K','L','C','P','E','U','L'/
63 N=30
64 ISR=1
65 ISU=1
66 ICSR=1
67 ICSU=1
68 KNTF=0
69 KNTU=0
70 KNTF2=0
71 KNTU2=0
72 C INITIALIZE ARRAYS & LOGICAL FLAGS
73 CALL WTRINI(N)
74 WRITE(10UT,2)
75 2 FORMAT(' ',10X,'WATERSHED MANAGEMENT CONTROL PACKAGE')
76 DO 12 I=1,80
77 12 CMD(I)=BLNK
78 C READ COMMAND LINE
79 WRITE(10UT,15)
80 15 FORMAT(' -ENTER COMMAND')
81 READ(IN,16) (CMD(I),I=1,80)
82 16 FORMAT(A8A1)
83 C FIND COMMAND NUMBER, LENGTH OF COMMAND, POSITION OF 1ST BLANK SPACE
84 NUMCMD=ISRCH(CMD,LEN,IPOS)
85 IF(NUMCMD .EQ. 0) GO TO 800
86 C
87 C NUMCMD COMMAND
88 C 1 CONTROL
89 C 2 COST
90 C 3 STOP
91 C 4 HELP
92 C 5 INPUT
93 C 6 LOAD
94 C 7 PERCENT
95 C 8 READ
96 C 9 SAVE
97 C 10 TITLES
98 C 11 WORKSHEETS
99 C 12 XCOST
100 C 13 XPERCENT
101 C 14 TRANSMISSION
102 C
103 GO TO(100,200,900,140,160,350,375,220,240,250,260,200,375,
104 400),NUMCMD
105 C
106 C CONTROL COMMAND
107 C
108 100 KEYWRD=0
109 K=0
110 NOPT=0
111 DO 102 J=1,8
112 102 OPTVAL(J)=0
113 DO 103 J=1,4
114 103 ARFACN(J)=0
115 103 UALCON(J)=0
116 IF(LEN.LE.IPOS) GO TO 115
117 C SCAN COMMAND LINE FOR "URBAN" OR "RURAL"
118 104 IPOS1=MINBLNK(CMD,IPOS)
119 IF(CMD(IPOS1).EQ.'URBAN') KEYWRD=1
120 IF(CMD(IPOS1).EQ.'RURAL') KEYWRD=2

```

```

121      IF(KEYWRD.EQ.0) GO TO 802
122      C      NOW BEGIN SEARCH FOR OPTION VALUES
123      IPOS1=INDEX(CMD,LPAREN,IPOS1)
124      IF(IPOS1.EQ.-1) GO TO 114
125      LPOS=INDEX(CMD,IRPARN,IPOS1)
126      IF(LPOS.EQ.-1) GO TO 802
127      106 IPOS1=IPOS1+1
128      IPOS1=NBLANK(CMD,IPOS1)
129      K=K+1
130      IF(KEYWRD.EQ.2) GO TO 109
131      C      HERE FOR URBAN OPTION VALUES
132      DO 107 J=1,5
133      M=J
134      IF(CMD(IPOS1).NE.OPTURB(J)) GO TO 107
135      C      LOOK FOR "AREA" OR "UAL"
136      IF(M.L1.3 .OR. M.EQ.5) GO TO 112
137      IPOS1=INDEX(CMD,ISLASH,IPOS1)
138      IF(IPOS1.NE.-1) GO TO 142
139      DO 141 JZ=1,4
140      IF(M.EQ.3) AREACN(JZ)=1
141      IF(M.EQ.4) UALCON(JZ)=1
142      141 CONTINUE
143      GO TO 112
144      C      "AREA/" OR "UAL/" FOR SPECIFIC SOURCES
145      142 IPOS1=IPOS1+1
146      IPOS1=NBLANK(CMD,IPOS1)
147      108 MF=0
148      IPOS2=IPOS1+1
149      IPOS3=IPOS1+2
150      IF(CMD(IPOS1).EQ.ISORCE .AND. CMD(IPOS2).EQ.ITEE .AND.
151      & CMD(IPOS3).EQ.I00) MF=1
152      IF(CMD(IPOS1).EQ.ICPLAN .AND. CMD(IPOS2).EQ.I00 .AND.
153      & CMD(IPOS3).EQ.MMM) MF=2
154      IF(OTHPUL.GT.0 .AND. MF.EQ.0) GO TO 133
155      IF(MF.EQ.0) GO TO 805
156      IF(M.EQ.4) GO TO 131
157      IF(MF.EQ.1) AREACN(1)=1
158      IF(MF.EQ.2) AREACN(2)=1
159      GO TO 112
160      131 IF(MF.EQ.1) UALCON(1)=1
161      IF(MF.EQ.2) UALCON(2)=1
162      GO TO 112
163      C      HERE FOR OTHER URBAN AREAS
164      133 IF(CMD(IPOS1).EQ.TLEUR1(1) .AND. CMD(IPOS2).EQ.TLEUR1(2) .AND.
165      & CMD(IPOS3).EQ.TLEUR1(3)) MF=3
166      IF(MF.EQ.0 .AND. OTHPUL.EQ.2) GO TO 135
167      IF(MF.EQ.0) GO TO 805
168      IF(M.EQ.3) AREACN(3)=1
169      IF(M.EQ.4) UALCON(3)=1
170      GO TO 112
171      135 IF(CMD(IPOS1).EQ.TLEUR2(1) .AND. CMD(IPOS2).EQ.TLEUR2(2) .AND.
172      & CMD(IPOS3).EQ.TLEUR2(3)) MF=4
173      IF(MF.EQ.0) GO TO 805
174      IF(M.EQ.3) AREACN(4)=1
175      IF(M.EQ.4) UALCON(4)=1
176      GO TO 112
177      107 CONTINUE
178      GO TO 805
179      C      HERE FOR RURAL OPTION VALUES
180      109 DO 110 J=1,8

```

```

181      M=J
182      IF(CMD(IPOS1).EQ.)PTRUL(J)) GO TO 112
183 110 CONTINUE
184      GO TO 805
185  C      NOW STORE OPTION VALUE AND LOOK FOR NEXT OPTION
186 112 OPTVAL(K)=M
187      NOPT=K
188      IPOS1=INDEX(CMD,COMA,IPOS1)
189      IF(IPOS1.GT.LPOS) GO TO 802
190      IF(IPOS1.EQ.-1) GO TO 114
191      GO TO 106
192  C      FIRST SORT OPTIONS FOR WTRCTL THEN DETERMINE STAGE NUMBER
193 114 IF(NOPT.GT.1) CALL ISORT(OPTVAL,NOPT,8)
194      IF(KEYWRD.EQ.2) GO TO 125
195      IF(KNTU.EQ.ISU) ISU=ISU+1
196      CALL WTRCTL(ISU,KEYWRD,KNTU,IER)
197      GO TO 130
198 125 IF(KNTR.EQ.ISR) ISR=ISR+1
199      CALL WTRCTL(ISR,KEYWRD,KNTR,IER)
200 130 ISTAGE=MAX0(ISR,ISU)
201      IF(IER.EQ.0) GO TO 10
202      GO TO(806,807,805),IER
203  C      HERE FOR UNSPECIFIED KEYWORD IN COMMAND
204 115 WRITE(1OUT,116)
205 116 FORMAT('ENTER "URBAN" OR "RURAL" WITH OPTION VALUES, IF ANY,',
206      &' IN PARENTHESIS SEPARATED BY COMMAS')
207      DO 117 J=1,80
208 117 CMD(J)=BLNK
209      READ(1N,16)(CMD(J),J=1,80)
210      IPOS=1
211      LEN=LENGTH(CMD,80)
212      GO TO 104
213  C
214  C      HELP COMMAND
215  C
216 140 CALL WTRHLP
217      GO TO 10
218  C
219  C      INPUT COMMAND
220  C
221 160 KEYWRD=1
222      IPOS1=IPOS
223      IF(LEN.LE.IPOS) GO TO 190
224      IPOS1=NRLANK(CMD,IPOS)
225  C      SCAN FOR OPTION VALUES
226 165 KEYWRD=0
227      IF(CMD(IPOS1).EQ.ISORCF) KEYWRD=1
228      IF(CMD(IPOS1).EQ.IURAN) KEYWRD=2
229      IF(CMD(IPOS1).EQ.NCPLAN) KEYWRD=3
230      IF(CMD(IPOS1).EQ.ICPLAN) KEYWRD=4
231      IF(KEYWRD.EQ.0) GO TO 802
232 190 CALL WTRINP(KEYWRD,IER)
233      IF(IER.NE.0) GO TO 804
234  C      LOOK FOR ANOTHER OPTION
235      IPOS=INDEX(CMD,COMA,IPOS1)
236      IF(IPOS.EQ.-1) GO TO 10
237      IPOS=IPOS+1
238      IPOS1=NRLANK(CMD,IPOS)
239      GO TO 165
240  C

```

```

241 C      COST (NUMCMD=2) & XCOST (NUMCMD=12) COMMANDS
242 C
243 200 KEYWRD=0
244 ICFLG=0
245 C      FLAG IF COSTS DIFFER BETWEEN SOURCES (XCOST)
246 IF (NUMCMD.EQ.12) ICFLG=1
247 IF (LEN.LE.IPOS) GO TO 208
248 IPOS1=IPOS
249 202 IPOS1=NBLANK(CMD,IPOS1)
250 IF (CMD(IPOS1).EQ.IURAN) KEYWRD=1
251 IF (CMD(IPOS1).EQ.IPUAL) KEYWRD=2
252 IF (KEYWRD.EQ.0) GO TO 802
253 IF (KEYWRD.EQ.2) GO TO 203
254 C      COUNT STAGES
255 IF (KNTU2.EQ.ICSU) IC SU=IC SU+1
256 CALL WTRCST(ICFLG,KEYWRD,ICSU,KNTU2,IER)
257 GO TO 204
258 203 IF (KNTR2.EQ.ICSR) ICSR=ICSR+1
259 CALL WTRCST(ICFLG,KEYWRD,ICSR,KNTR2,IER)
260 204 IF (IER.EQ.0) GO TO 205
261 GO TO (807,806),IER
262 C      LOOK FOR ANOTHER KEYWORD
263 205 IPOS1=INDEX(CMD,COMA,IPOS1)
264 IF (IPOS1.EQ.-1) GO TO 10
265 IPOS1=IPOS1+1
266 KEYWRD=0
267 GO TO 202
268 C      HERE FOR UNSPECIFIED KEYWORD
269 208 WRITE(10UT,386)
270 DO 210 I=1,80
271 210 CMD(I)=BLNK
272 READ(1N,16)(CMD(I),I=1,80)
273 IPOS1=1
274 LEN=LENGTH(CMD,80)
275 GO TO 202
276 C
277 C      READ COMMAND
278 C
279 220 KEYWRD=0
280 IF (LEN.LE.IPOS) GO TO 235
281 IPOS1=NBLANK(CMD,IPOS)
282 IF (CMD(IPOS1).EQ.ICPLAN) KEYWRD=1
283 IF (KEYWRD.EQ.0) GO TO 805
284 235 CALL WTRSAV(1,ISTAGF,KEYWRD)
285 GO TO 10
286 C
287 C      SAVE COMMAND
288 C
289 240 KEYWRD=0
290 IF (LEN.LE.IPOS) GO TO 245
291 IPOS1=NBLANK(CMD,IPOS)
292 IF (CMD(IPOS1).EQ.ICPLAN) KEYWRD=1
293 IF (KEYWRD.EQ.0) GO TO 805
294 245 CALL WTRSAV(3,ISTAGF,KEYWRD)
295 GO TO 10
296 C
297 C      TITLES COMMAND
298 C
299 250 IPOS1=INDEX(CMD,LPAREN,3)
300 IF (IPOS1.EQ.-1) GO TO 802

```

```

301      IPDS1=IPDS1+1
302      IPDS2=MBLANK(CMD,IPDS1)
303      IF(IPDS2.EQ.-1) GO TO 802
304      IUFLG=0
305      C      GET TITLE NUMBER
306      251 DO 252 J=1,4
307          K=J
308          IF(CMD(IPDS2).EQ.NUMWRK(J)) GO TO 255
309      252 CONTINUE
310      IF(IUFLG.NE.0) GO TO 802
311      C      FLAG IF SPECIFYING OTHER URBAN OR RURAL AREAS
312      IF(CMD(IPDS2).EQ.IUBAN) IUFLG=1
313      IF(CMD(IPDS2).EQ.IRUAL) IUFLG=2
314      IPDS2=IPDS2+1
315      IPDS2=MBLANK(CMD,IPDS2)
316      GO TO 251
317      C
318      255 IPDS1=INDEX(CMD,IPARN,IPDS2)
319      IF(IPDS1.EQ.-1) GO TO 802
320      J=0
321      IF(IUFLG.EQ.0) NTLF=K
322      IF(IUFLG.EQ.1) OTHURB=K
323      IF(IUFLG.EQ.2) OTHRUL=K
324      IF(OTHURB.GT.2.OR.OTHRUL.GT.3) GO TO 802
325      IREG=IPDS1+1
326      C      GET TITLE FROM COMMAND LINE
327      DO 258 I=IREG,LEN
328          J=J+1
329          IF(IUFLG.NE.0.AND.J.GT.12) GO TO 258
330          IF(IUFLG.EQ.0.AND.J.GT.40) GO TO 258
331          IF(IUFLG.EQ.0) KTITLE(NTLF,J)=CMD(I)
332          IF(IUFLG.EQ.0) GO TO 258
333          IF(IUFLG.EQ.2) GO TO 257
334          IF(OTHURB.EQ.1) TLFUR1(J)=CMD(I)
335          IF(OTHURB.EQ.1) TYP00(8,J)=CMD(I)
336          IF(OTHURB.EQ.2) TLFUR2(J)=CMD(I)
337          IF(OTHURB.EQ.2) TYP00(9,J)=CMD(I)
338          GO TO 258
339      257 IF(OTHRUL.EQ.1) TLERU1(J)=CMD(I)
340          IF(OTHRUL.EQ.2) TLERU2(J)=CMD(I)
341          IF(OTHRUL.EQ.3) TLERU3(J)=CMD(I)
342      258 CONTINUE
343      GO TO 10
344      C
345      C      WORKSHEET COMMAND
346      C
347      260 KEYW0D=0
348      LOOK=LFV
349      IF(LEN.LE.IPDS) GO TO 282
350      IPDS2=MBLANK(CMD,IPDS)
351      LOOK=IPDS2
352      C      LOOK FOR OUTPUT DEVICE NUMBER -- KOUT
353      IPDS2=INDEX(CMD,LPAREN,1)
354      IF(IPDS2.NE.-1) GO TO 290
355      IPDS2=LOOK
356      KOUT=IPUT
357      GO TO 265
358      C      GET WORKSHEET NUMBER
359      262 IPDS2=INDEX(CMD,COMA,LOOK)
360      IF(IPDS2.EQ.-1.AND.LOOK.LT.LEN) GO TO 802

```

```

361      IF(IPOS2.EQ.-1) GO TO 10
362      264 IPOS2=IPOS2+1
363      LOOK=NBLANK(CMD,IPOS2)
364      IF(LOOK.EQ.-1) GO TO 10
365      265 KEYWRD=CMD(LOOK)
366      IF(KEYWRD.EQ.ALL) GO TO 275
367      DO 268 J=1,9
368      KTABLE=J
369      IF(KEYWRD.EQ.NUMWRK(J)) GO TO 277
370      268 CONTINUE
371      GO TO 802
372      C      HERE FOR "ALL" WORKSHEETS
373      275 KEYWRD=9
374      LOOK=LEN
375      GO TO 280
376      277 KEYWRD=KTABLE
377      280 CALL WRKSHT(ISTAGE,KEYWRD,IER)
378      IF(IER.NE.0) GO TO 806
379      GO TO 262
380      C      NO WORKSHEET NUMBER SPECIFIED
381      282 WRITE(1OUT,284)
382      284 FORMAT('OENTER WORKSHEET NUMBERS SEPARATED BY COMMAS OR "ALL"')
383      DO 285 I=1,80
384      285 CMD(I)=BLNK
385      READ(IN,16)(CMD(I),I=1,80)
386      IPOS=1
387      LEN=LENGTH(CMD,80)
388      GO TO 260
389      C      HERE FOR OUTPUT TO A FILE -- KOUT
390      290 IPOS2=IPOS2+1
391      LPOS=INDEX(CMD,IRPARN,IPOS2)
392      IF(LPOS.EQ.-1) GO TO 802
393      IPOS2=NBLANK(CMD,IPOS2)
394      DO 292 I=1,9
395      M2=I
396      IF(CMD(IPOS2).EQ.NUMWRK(I)) GO TO 293
397      292 CONTINUE
398      GO TO 802
399      293 KOUT=M2
400      IPOS2=LPOS
401      GO TO 264
402      C
403      C      LOAD COMMAND
404      C
405      350 KEYWRD=0
406      IF(LEN.LE.IPOS) GO TO 365
407      IPOS1=IPOS
408      354 IPOS1=NBLANK(CMD,IPOS1)
409      C      GET KEYWORD
410      IF(CMD(IPOS1).EQ.IUBAN) KEYWRD=1
411      IF(CMD(IPOS1).EQ.ICPLAN) KEYWRD=2
412      IF(CMD(IPOS1).EQ.NCPLAN) KEYWRD=3
413      IF(KEYWRD.EQ.0) GO TO 802
414      IF(KEYWRD.EQ.1) LODJPB=.TRUE.
415      IF(KEYWRD.EQ.3) LODRNC=.TRUE.
416      IF(KEYWRD.NE.2) GO TO 362
417      C      HERE FOR RURAL CONPLAN KEYWRD - SEARCH FOR OPTION (UAL)
418      IPOS1=INDEX(CMD,LPAREN,IPOS1)
419      IF(IPOS1.EQ.-1) GO TO 360
420      LPOS=INDEX(CMD,IRPARN,IPOS1)

```

```

421      IF(IPOS.EQ.-1) GO TO 802
422      IPOS1=IPOS1+1
423      IPOS1=NBANK(CMD,IPOS1)
424      IF(CMD(IPOS1).NE.IJBN) GO TO 805
425      L4UAL=.TRUE.
426      LODRCP=.FALSE.
427      GO TO 362
428      360 LODRCP=.TRUE.
429      L4UAL=.FALSE.
430      C      LOOK FOR ANOTHER KEYWORD
431      362 IPOS1=INDEX(CMD,COMA,IPOS1)
432      IF(IPOS1.EQ.-1) GO TO 10
433      IPOS1=IPOS1+1
434      KEYWRD=0
435      GO TO 354
436      C      HERE FOR UNSPECIFIED KEYWORD IN COMMAND
437      365 WRITE(IOUT,366)
438      366 FORMAT('CENTER "URBAN" OR "NON-CROPLAND" OR "CROPLAND" OR ',
439      &      '"CROPLAND(UAL)"')
440      DO 368 I=1,80
441      369 CMD(I)=BLNK
442      READ(IN,16)(CMD(I),I=1,80)
443      IPOS1=1
444      LEN=LENGTH(CMD,80)
445      GO TO 354
446      C
447      C      PERCENT (NUMCMD=7) & XPERCENT (NJMCMD=13) COMMANDS
448      C
449      375 KEYWRD=0
450      IF(LEN.LE.IPOS) GO TO 385
451      IPOS1=IPOS
452      376 IPOS1=NBANK(CMD,IPOS1)
453      IF(CMD(IPOS1).EQ.IJBN) KEYWRD=1
454      IF(CMD(IPOS1).EQ.IRUAL) KEYWRD=2
455      IF(KEYWRD.EQ.0) GO TO 802
456      IF(KEYWRD.EQ.1) UPRCNT=.TRUE.
457      IF(KEYWRD.EQ.2) RPRCNT=.TRUE.
458      IF(NUMCMD.EQ.7) GO TO 390
459      C      HERE FOR XPERCENT COMMAND
460      IF(KEYWRD.EQ.1) UXPCNT=.TRUE.
461      IF(KEYWRD.EQ.2) RXPCNT=.TRUE.
462      C      LOOK FOR ANOTHER KEYWORD
463      380 IPOS1=INDEX(CMD,COMA,IPOS1)
464      IF(IPOS1.EQ.-1) GO TO 10
465      IPOS1=IPOS1+1
466      KEYWRD=0
467      GO TO 376
468      C      HERE FOR UNSPECIFIED KEYWORD IN COMMAND
469      385 WRITE(IOUT,386)
470      386 FORMAT('CENTER "URBAN" OR "RURAL"')
471      DO 388 I=1,80
472      388 CMD(I)=BLNK
473      READ(IN,16)(CMD(I),I=1,80)
474      IPOS1=1
475      LEN=LENGTH(CMD,80)
476      GO TO 376
477      C
478      C      TRANSMISSION COMMAND
479      C
480      400 KEYWRD=0

```

```

481      IF (LEN .LE. IPOS) GO TO 420
482      IPOS1=HBLANK(CMD,IPOS)
483      ISIZE=0
484      404 CONTINUE
485      IF (CMD(IPOS1).EQ.COMA) GO TO 410
486      C      GET POSITION NUMBER
487      DO 405 I=1,10
488          JV=I
489          IF (CMD(IPOS1).EQ.NUMWRK(I)) GO TO 438
490      405 CONTINUE
491      GO TO 802
492      C      TEST FOR NUMBER OF DIGITS IN POSITION NUMBER
493      408 ISIZE=ISIZE + 1
494      IF (ISIZE .GT. 3) GO TO 802
495      JVAL(ISIZE)=JV
496      IPOS1=IPOS1+1
497      IF (IPOS1 .GT. LEN) GO TO 410
498      GO TO 404
499      410 IF (ISIZE .EQ. 0) GO TO 802
500      GO TO(411,414,417),ISIZE
501      C
502      411 KEYWRD=JVAL(1)
503      GO TO 420
504      414 IF (JVAL(2).EQ.10) JVAL(2)=0
505      KEYWRD=JVAL(1)*10 + JVAL(2)
506      GO TO 420
507      417 IF (JVAL(2).EQ.10) JVAL(2)=1
508      IF (JVAL(3).EQ.10) JVAL(3)=0
509      KEYWRD=(JVAL(1)*100)+(JVAL(2)*10)+JVAL(3)
510      IF (KEYWRD .GT. NSRCE) GO TO 802
511      C
512      420 CALL WTRTRN(KEYWRD)
513      LRANK=.FALSE.
514      IPOS1=IPOS1+1
515      IF (IPOS1 .GT. LEN) GO TO 10
516      ISIZE = 0
517      GO TO 404
518      C      ERRORS
519      800 WRITE(IOUT,301)
520      301 FORMAT('ILLEGAL COMMAND')
521      GO TO 10
522      C
523      802 WRITE(IOUT,302)
524      302 FORMAT(1H0,'ILLEGAL SYNTAX OF COMMAND')
525      GO TO 10
526      C
527      804 WRITE(IOUT,303) KEYWRD
528      303 FORMAT(1H0,'INPUT DATA EXISTS FOR WORKSHEET',I2)
529      GO TO 10
530      C
531      805 WRITE(IOUT,304)
532      304 FORMAT(1H0,'ILLEGAL OPTION VALUE')
533      GO TO 10
534      C
535      806 WRITE(IOUT,305) KEYWRD
536      305 FORMAT(1H0,'INPUT DATA DOES NOT EXIST FOR WORKSHEET',I2)
537      GO TO 10
538      C
539      807 WRITE(IOUT,306)
540      306 FORMAT(1H0,'CANNOT ENTER MORE THAN 3 STAGES OF CONTROLS')

```

```

541      GO TO 10
542      C
543      900 CONTINUE
544      STOP
545      END
546      C
547      FUNCTION ISRCH(CMD,LEN,IPOS)
548      C
549      C      WTRSHD FUNCTION TO DETERMINE COMMAND NUMBER, LENGTH OF COMMAND,
550      C      AND POSITION OF FIRST SPACE FOLLOWING THE COMMAND
551      C
552      INTEGER CMD(80),COMAND(14,3),BLNK
553      DATA COMAND
554      &  /'C','C','S','H','I','L','P','R','S','T','W','X','X','T',
555      &  'O','O','T','E','M','O','E','E','A','I','O','C','P','R',
556      &  'N','S','O','L','P','A','R','A','V','T','R','O','E','A'/
557      DATA BLNK/' '/
558      ISRCH=0
559      DO 10 I=1,14
560      K=I
561      DO 5 J=1,3
562      IF(CMD(J).NE.COMAND(I,J)) GO TO 10
563      IFOUND=J
564      5 CONTINUE
565      IF(IFOUND.EQ.3) GO TO 35
566      10 CONTINUE
567      RETURN
568      C
569      35 ISRCH=K
570      LEN=LENGTH(CMD,80)
571      IPOS=INDEX(CMD,BLNK,1)
572      RETURN
573      END
574      C
575      FUNCTION LENGTH(CMD,N)
576      C
577      C      FUNCTION TO FIND THE LENGTH OF CMD WITH DIMENSION N
578      C
579      INTEGER CMD(N),BLNK
580      DATA BLNK/' '/
581      K=N+1
582      DO 10 I=1,N
583      K=K-1
584      IF(CMD(K).NE.BLNK) GO TO 15
585      10 CONTINUE
586      LENGTH=-1
587      RETURN
588      C
589      15 LENGTH=K
590      RETURN
591      END
592      C
593      FUNCTION INDEX(CMD,ISTVAL,IBEG)
594      C
595      C      FUNCTION TO LOCATE ISTVAL IN CMD BEGINNING SEARCH AT IBEG
596      C
597      INTEGER CMD(80),ISTVAL
598      I=IBEG-1
599      10 I=I+1
600      IF(I.GT.80) GO TO 15

```

```

601      IF (CMD(I).NE.TSTVAL) GO TO 10
602      INDEX=I
603      RETURN
604      C
605      15 INDEX=-1
606      RETURN
607      END
608      C
> 609      FUNCTION NBLANK(CMD, IREG)
610      C
611      C      FUNCTION TO FIND THE FIRST NON-BLANK CHARACTER
612      C      IN CMD BEGINNING SEARCH AT IREG
613      C
614      INTEGER CMD(80),BLNK
615      DATA BLNK/' '/
616      I=IREG-1
617      10 I=I+1
618      IF (I.GT.80) GO TO 15
619      IF (CMD(I).EQ.BLNK) GO TO 10
620      NBLANK=I
621      RETURN
622      C
623      15 NBLANK=-1
624      RETURN
625      END
626      C
> 627      SUBROUTINE ISORT(IRAY,NVAL,NDIM)
628      C
629      C      ROUTINE TO BUBBLE SORT THE NVAL VALUES IN IRAY IN ASCENDING ORDER
630      C
631      DIMENSION IRAY(NDIM)
632      1 IFLAG=0
633      N=NVAL-1
634      DO 5 I=1,N
635          IF (IRAY(I).LE. IRAY(I+1)) GO TO 5
636          IFLAG=1
637          ITMP=IRAY(I)
638          IRAY(I)=IRAY(I+1)
639          IRAY(I+1)=ITMP
640      5 CONTINUE
641      IF (IFLAG .NE. 0) GO TO 1
642      RETURN
643      END
644      C
> 645      SUBROUTINE WTRHLP
646      C
647      C      HELP LIST OF WTRSHD COMMANDS
648      C
649      DIMENSION KTITLE(4,40)
650      COMMON /WTRID/ IN, IOUT, NSRCE, KTITLE, NTLE, NURB, NRUNCP, NRUCRP
651      C
652      WRITE(IOUT, 5)
653      5 FORMAT(' ', 20X, 'WTRSHD COMMANDS'//2X, 'COMMAND', T33, 'DESCRIPTION' /
654      & /2X, 'CONTROL', T25, 'TO ENTER CONTROL STRATEGIES' /
655      & /2X, 'COST', T25, 'TO ENTER PER UNIT COSTS OF CONTROL' /
656      & /2X, 'STOP', T25, 'END EXECUTION'//2X, 'HELP', T25,
657      & 'PRINTS THIS TABLE' /)
658      WRITE(IOUT, 10)
659      10 FORMAT(2X, 'INPUT', T25, 'TO ENTER INITIAL STATE OF WATERSHED ',
660      & 'SOURCES'//2X, 'LOAD', T25, 'TO ENTER LOAD DIRECTLY' /)

```

```

661      G/2X,'PERCENT',T25,'TO ENTER CONTROLS AS PERCENT REDUCTIONS'
662      G//2X,'READ',T25,'READ INPUT DATA FROM PERMANENT ',
663      G'SORAGE'//)
664      WRITE(IOUT,15)
665      15 FORMAT(2X,'SAVE',T25,'SAVE WORKSHEET DATA'//
666      G2X,'TITLE',T25,'TITLE(S) FOR WATERSHED ANALYSIS'//
667      G2X,'WORKSHEET',T25,'SPECIFIED WORKSHEETS TO BE OUTPUT'//
668      G/2X,'XCOST',T25,'IF COSTS DIFFER BETWEEN SOURCES'//
669      G/2X,'XPERCENT',T25,'IF PERCENT REDUCTIONS DIFFER BETWEEN SOURCES'//
670      G/2X,'TRANSMISSION',T25,'TO ALTER EFFECTIVE TRANSMISSIONS')
671      C
672      RETURN
673      END
674      C
675      SUBROUTINE WTRININ)
676      C
677      C      INITIALIZATION ROUTINE FOR WTRSHD
678      C
679      C *****
680      LOGICAL L INPUT,L DURB,L DRNC,L DRCP,L4UAL,L IN1,L IN2,L IN3,L IN4
681      LOGICAL URCTL,RUCTL,UCTLOD(3),RCTLOD(3),UPRCNT,UXPCNT,RPRCNT,
682      G      RXPcnt,URBCST,RULCST,L RANK
683      INTEGER OTHURB,OTHRUL,SRCE(210,28),TYPCOD(9,12),OPTVAL(8)
684      DIMENSION ITYP(210),NPOS(210),AUNIT(210,2),KTITLE(4,40),
685      G      SURFEA(210,3),CTLOD(210,3),CTUSLE(3,30,5)
686      DIMENSION STMCTL(30,3),STMARA(30,3),COMCTL(30,3),COMARA(30,3),
687      G      OT1CTL(30,3),OT1ARA(30,3),OT2CTL(30,3),OT2ARA(30,3)
688      DIMENSION PLOW(30),PCONC(30),UALS(30),UALC(30),W2UAL1(30),
689      G      W2UAL2(30),TRNSMN(210)
690      INTEGER TLEUR1(12),TLEUR2(12),TLEUR1(12),TLEUR2(12),TLEUR3(12),
691      G      APOE(210),AREACN(4),UALCON(4)
692      DIMENSION GRASA(30),UALG(30),WLND(30),UALW(30),W3OTA1(30),
693      G      W3UAL1(30),W3UAL2(30),W3UAL3(30),W3OTA2(30),W3OTA3(30)
694      REAL LODRNL(30),LODRN2(30),LOD1W3(30),LOD2W3(30),LOD3W3(30)
695      DIMENSION USLER(30),USLEK(30),USLELS(30),USLEC(30),W4UAL(30),
696      G      USLEP(30),SOILS1(30),SOILS2(30),TFLER(30),RATIO(30)
697      DIMENSION PDIF(210,3),TCOST(210,3),PUCOST(210,3)
698      DIMENSION PMOUTH(210),DIFMTH(210,3),CPUREM(210,3),KERANK(210,3)
699      COMMON /WTRID/ IN,IOUT,NSRCE,KTITLE,NTLE,NURB,NRUNCP,NRUCRP
700      COMMON /WTRTF/ LINPUT,L DURB,L DRNC,L IN1,L IN2,L IN3,L IN4,OTHURB,
701      G      OTHRUL,L DRCP,TYPCOD,JIN,JOUT,L4UAL,KOUT,L RANK
702      COMMON /WRKSH1/ SRCE,ITYP,NPOS,APOE,AUNIT,SURFEA,TLEUR1,TLEUR2
703      COMMON /WRKSH2/ PLOW,PCONC,UALS,UALC,W2UAL1,W2UAL2
704      COMMON /WRKSH3/ GRASA,UALG,WLND,UALW,LODRN1,LODRN2,
705      G      W3OTA1,W3UAL1,W3UAL2,W3UAL3,W3OTA2,W3OTA3,
706      G      TLEUR1,TLEUR2,TLEUR3,LOD1W3,LOD2W3,LOD3W3
707      COMMON /WRKSH4/ USLER,USLEK,USLELS,USLEC,USLEP,SOILS1,
708      G      SOILS2,TFLER,W4UAL,RATIO
709      COMMON /WRKSH6/ PDIF,TCOST,PUCOST,TRNSMN,PMOUTH,DIFMTH,CPUREM,
710      G      KERANK
711      COMMON /WTRCON/ UCTLOD,RCTLOD,URCTL,RUCTL,UPRCNT,JXPCNT,RPRCNT,
712      G      RXPcnt,OPTVAL,NPT,URBCST,RULCST,AREACN,UALCON
713      DIMENSION CTLEFW(30,3),CTLCNC(30,3)
714      DIMENSION CTLE(30,3),SOILC1(30,3),SOILC2(30,3),CTLERO(30,3),
715      G      CUALW4(30,3)
716      DIMENSION TOTCST(3),TOTREM(3),KNOX(540),FINCST(540),FINREM(540),
717      G      ISFIN(540)
718      INTEGER FINSRC(540,43)
719      COMMON /WRKSH8/ TOTCST,TOTREM,TOTLOD,TOTMTH,KNOX,FINSRC,FINCST,
720      G      FINREM,ISFIN,NUM

```

```

721      COMMON /UBNCTL/ CTIFLW,CTL CNC,CTLDD,STMCTL,STMARA,CQ4CTL,
722      &      COMARA,OT1CTL,OT1ARA,OT2CTL,OT2ARA
723      COMMON /RFDCTL/ CTUSLF,CTLE,SOILC1,SOILC2,CTLERD,CUALW4
724      INTEGER STYP1(12),STYP2(12),STYP3(12),STYP4(12),
725      &      STYP6(12),STYP7(12)
726      DATA STYP1/'N','O','N','-', 'C','R','O','P','L','A','N','D'/
727      DATA STYP2/' ',' ','C','R','O','P','L','A','N','D',' ',' '/
728      DATA STYP3/' ','M','U','N','I','C','I','T','Y','P','A','R','A','M','E','T','E','R','S',' '/
729      DATA STYP4/' ',' ','I','N','D','U','S','T','R','Y',' ',' '/
730      DATA STYP6/' ',' ','S','T','O','R','E',' ',' ',' ',' ',' '/
731      DATA STYP7/' ',' ','C','O','M','B','I','N','E','D',' ',' ',' '/
732      C *****
733      DATA BLNK,IBLNK/' ',' '/
734      IN=5
735      IOUT=5
736      KOUT=IOUT
737      JIN=9
738      JOUT=9
739      LINPJT=.FALSE.
740      LODURB=.FALSE.
741      LODRNC=.FALSE.
742      LODRCP=.FALSE.
743      LRANK=.FALSE.
744      LIN1=.FALSE.
745      LIN2=.FALSE.
746      LIN3=.FALSE.
747      LIN4=.FALSE.
748      L4UAL=.FALSE.
749      URBCIL=.FALSE.
750      RULCTL=.FALSE.
751      URBCST=.FALSE.
752      RULCST=.FALSE.
753      UPRCNT=.FALSE.
754      RPRCNT=.FALSE.
755      UXPCNT=.FALSE.
756      RXPCNT=.FALSE.
757      OTHURB=0
758      OTHRUL=0
759      NURB=0
760      NRUNC=0
761      NRJCRP=0
762      NTIE=0
763      NSRCE=0
764      TOTLOD=0.0
765      TOTMTH=0.0
766      NUM=0
767      DO 2 I=1,12
768          TLEUR1(I)=IBLNK
769          TLEUR2(I)=IBLNK
770          TLERU1(I)=IBLNK
771          TLERU2(I)=IBLNK
772      2  TLERU3(I)=IBLNK
773      DO 4 I=1,4
774          DO 3 J=1,40
775      3  KTITLF(I,J)=IBLNK
776      4  CONTINUE
777      DO 5 J=1,3
778          RCTLOD(J)=.FALSE.
779          UCTLOD(J)=.FALSE.
780          TOTCSIP(I)=0.0

```

```

781      TOTREM(I)=0.0
782      5 CONTINUE
783      DO 10 I=1,N
784          PFLOW(I)=0.0
785          PCONC(I)=0.0
786          UALS(I)=0.0
787          UALC(I)=0.0
788          W2UAL1(I)=0.0
789          W2UAL2(I)=0.0
790          GRASA(I)=0.0
791          UALG(I)=0.0
792          WLNDIA(I)=0.0
793          UALW(I)=0.0
794          LODRN1(I)=0.0
795          LODRN2(I)=0.0
796          USLFR(I)=0.0
797          USLFR(I)=0.0
798          USLFLS(I)=0.0
799          USLEC(I)=0.0
800          JSLEFP(I)=0.0
801          SOILS1(I)=0.0
802          SOILS2(I)=0.0
803          TTLER(I)=0.0
804          RATIO(I)=0.0
805          W3OTA1(I)=0.0
806          W3OTA2(I)=0.0
807          W3OTA3(I)=0.0
808          W3UAL1(I)=0.0
809          W3UAL2(I)=0.0
810          W3UAL3(I)=0.0
811          LOD1W3(I)=0.0
812          LOD2W3(I)=0.0
813          LOD3W3(I)=0.0
814          W4UAL(I)=0.0
815      DO 8 J=1,3
816          CTLFLW(I,J)=0.0
817          CTLCNC(I,J)=0.0
818          STMCTL(I,J)=0.0
819          STMARA(I,J)=0.0
820          COMCTL(I,J)=0.0
821          COMARA(I,J)=0.0
822          OT1CTI(I,J)=0.0
823          OT1ARA(I,J)=0.0
824          OT2CTI(I,J)=0.0
825          OT2ARA(I,J)=0.0
826          CTLF(I,J)=0.0
827          SOILC1(I,J)=0.0
828          SOILC2(I,J)=0.0
829          CTLFRD(I,J)=0.0
830          CUALW4(I,J)=0.0
831      DO 6 K=1,5
832      6      CTUSLF(J,I,K)=0.0
833      8 CONTINUE
834      10 CONTINUE
835      DO 20 K=1,210
836          APDF(K)=JOB1NK
837          ITYP(K)=0
838          NPOS(K)=0
839          TRNSMN(K)=1.0
840          PMDUTH(K)=0.0

```

```

841      DO 12 K2=1,3
842          IF (K2.EQ.3) GO TO 11
843          AJNIT(K,K2)=0.0
844      11 SURFEA(K,K2)=BLNK
845          PDIF(K,K2)=0.0
846          ICOST(K,K2)=0.0
847          PUCOST(K,K2)=0.0
848          CTL3D(K,K2)=0.0
849          DIFMTH(K,K2)=0.0
850          CPUREM(K,K2)=0.0
851          KFRANK(K,K2)=0.0
852      12 CONTINUE
853          DO 15 M=1,28
854      15 SRCE(K,M)=IBLNK
855      20 CONTINUE
856          DO 25 I=1,12
857              TYPCOD(1,I)=STYP1(I)
858              TYPCOD(2,I)=STYP2(I)
859              TYPCOD(3,I)=STYP3(I)
860              TYPCOD(4,I)=STYP4(I)
861              TYPCOD(5,I)=IBLNK
862              TYPCOD(6,I)=STYP6(I)
863              TYPCOD(7,I)=STYP7(I)
864              TYPCOD(8,I)=IBLNK
865              TYPCOD(9,I)=IBLNK
866      25 CONTINUE
867          DO 30 I=1,540
868              KNDX(I)=0
869              FINCST(I)=0.0
870              FIVREM(I)=0.0
871              ISFIN(I)=0
872          DO 28 J=1,43
873      28 FINSRC(I,J)=IBLNK
874      30 CONTINUE
875      C
876          RETURN
877      END
878      C
879      C SUBROUTINE WTRINP(KEYWRD,IFR)
880      C
881      C      WORKSHEET INPUT ROUTINES FOR INITIAL STATE OF WTRSHD
882      C
883      C *****
884      LOGICAL LINPUT,LODURB,LODRNC,LODPCP,L4UAL,LIN1,LIN2,LIN3,LIN4
885      LOGICAL LODPDR
886      INTEGER OTHURB,OTHRJL,SRCE(210,28),TYPCOD(9,12)
887      DIMENSION ITYP(210),NPOS(210),AUNIT(210,2),KTITLE(4,40),
888      & SURFEA(210,3)
889      DIMENSION PFLW(30),PCONC(30),UALS(30),UALC(30),W2UAL1(30),
890      & W2UAL2(30)
891      INTEGER TLEUR1(12),TLEUR2(12),TLEUR3(12),TLEUR4(12),
892      & APOE(210)
893      DIMENSION GRASA(30),UALG(30),WLND4(30),UALW(30),W3OTA1(30),
894      & W3UAL1(30),W3UAL2(30),W3UAL3(30),W3OTA2(30),W3OTA3(30)
895      REAL LODRN1(30),LODRN2(30),LODRN3(30),LODRN4(30),LODRN5(30),
896      DIMENSION USLEF(30),USLEK(30),USLELS(30),USLEC(30),W4UAL(30),
897      & USLEP(30),SOILS1(30),SOILS2(30),TTLER(30),RATIO(30)
898      COMMON /WTR17/ IN,LOUT,NSPCE,KTITLE,NTLF,NURB,NRJNCP,NRUCRP
899      COMMON /WTR18/ LINPUT,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHRB,
900      & OTHRJL,LODRCP,TYPCOD,JIN,LOUT,L4UAL,KOUT

```

```

901      COMMON /WRKSH1/ SRCE, ITP, NPOS, APOF, AUNIT, SURFEA, TLEUR1, TLEUR2
902      COMMON /WRKSH2/ PFLOW, PCONC, UALS, JAL C, W2JAL1, W2UAL2
903      COMMON /WRKSH3/ GRASA, UALG, WLND, UALW, LODRN1, LODRN2,
904      C      W3OTA1, W3UAL1, W3UAL2, W3UAL3, W3OTA2, W3OTA3,
905      C      TLEUR1, TLEUR2, TLEUR3, LODW3, LODW3, LODW3
906      COMMON /WRKSH4/ USLFR, USLFK, USLFLS, USLEC, USLEP, SOILS1,
907      C      SOILS2, TLER, W4UAL, RATIO
908      INTERSEPT STYP(12)
909      C *****
910      DATA FIN, ANSNO, BLNK, IBLNK, ICOMA/ 'END', 'N', ' ', ' ', ' ', ' ' /
911      DATA ANSYES/ 'Y' /
912      C
913      IER=1
914      GO TO (100, 200, 300, 400), KEYWPD
915      C
916      INPUT SOURCES
917      C
918      100 IF(LIN1) GO TO 500
919      WRITE(10UT, 101)
920      101 FORMAT('O ENTER TYPE CODE, SOURCE NAME FOR ALL SOURCES')
921      I=1
922      NURB=0
923      102 READ(IN, 103) ITP(I), (SRCE(I, NAM), NAM=1, 28)
924      103 FORMAT(13, 28A1)
925      IF(ITYP(I).EQ.0.OR.SRCE(I,1).EQ.FIN) GO TO 104
926      IF(ITYP(I).GT.22.OR.ITYP(I).LT.0) GO TO 107
927      IF(ITYP(I).GT.4) GO TO 125
928      IF(ITYP(I).EQ.1) NRUNCP=NRUNCP+1
929      IF(ITYP(I).EQ.2) NRUCRP=NRUCRP+1
930      IF(ITYP(I).LT.3) GO TO 130
931      C
932      C      HERE FOR URBAN & POINT SOURCES TO ALLOW SPACE FOR
933      C      POINT, STORM, COMBINED & TWO OTHER CATEGORIES OF SOURCES
934      C
935      J=1
936      NURB=NURB+1
937      IND=2+0THJRB
938      DO 128 IPNT=1, IND
939      I=I+1
940      DO 127 J2=13, 24
941      J3=J2-12
942      IF(IPNT.EQ.1) SRCE(I, J2)=TYPCOD(6, J3)
943      IF(IPNT.EQ.2) SRCE(I, J2)=TYPCOD(7, J3)
944      IF(IPNT.EQ.3) SRCE(I, J2)=TLEUR1(J3)
945      IF(IPNT.EQ.4) SRCE(I, J2)=TLEUR2(J3)
946      127 CONTINUE
947      128 ITP(I)=5+IPNT
948      C
949      130 I=I+1
950      GO TO 102
951      C      ERROR
952      107 WRITE(10UT, 108) ITP(I)
953      108 FORMAT('O', I5, ' IS AN ILLEGAL TYPE CODE - TRY AGAIN')
954      GO TO 102
955      C
956      C      HERE FOR MULTIPLE TYPE CODE (21 OR 12) - NON-CROPLAND/CROPLAND
957      C
958      125 IF(ITYP(I).LT.10) GO TO 107
959      ITP1=ITYP(I)/10
960      ITP2=ITYP(I)-ITYP1*10

```

```

961      IF(ITYP1.GT.2.OR.ITYP2.GT.2) GO TO 107
962      IF(ITYP1.EQ.1) NRUNCP=NRUNCP+1
963      IF(ITYP1.EQ.2) NRJCRP=NRJCRP+1
964      ITP(I)=ITYP1
965      I=I+1
966      IF(ITYP2.EQ.0) GO TO 102
967      ITP(I)=ITYP2
968      IF(ITYP2.EQ.1) NRUNCP=NRUNCP+1
969      IF(ITYP2.EQ.2) NRUCRP=NRUCRP+1
970      DO 126 NAM=1,28
971      126 SRCE(I,NAM)=SRCE(I-1,NAM)
972      I=I+1
973      GO TO 102
974      C
975      104 NSRCE=I-1
976      C
977      C      INPUT ROUTINE FOR WORKSHEET 1
978      C
979      IB=1
980      WRITE(IOUT,105)
981      135 WRITE(IOUT,111)
982      105 FORMAT('O INPUT FOR WORKSHEET 1 - PHYSICAL LAYOUT PER SOURCE')
983      111 FORMAT('/' ENTER POINT OF ENTRY CODE, AREA OR UNITS SERVED, SURFACE F
984      EATURES PER SOURCE')
985      DO 120 I=IB,NSRCE
986      IE=I
987      NPOS(I)=I
988      KODE=ITYP(I)
989      DO 119 J=1,12
990      119 STYP(J)=TYPCOD(KODE,J)
991      IF(KODE.LE.5) WRITE(IOUT,106)(STYP(J),J=1,12),
992      &(SRCE(I,NAM),NAM=1,28)
993      106 FORMAT(1X,12A1,3X,28A1)
994      IF(KODE.GT.5) WRITE(IOUT,109)(STYP(J),J=1,12)
995      109 FORMAT(11X,'- ',12A1)
996      READ(IN,110,ERR=122)APOE(I),AUNIT(I,1),(SURFEA(I,NTP),NTP=1,3)
997      110 FORMAT(A1,1X,F10.0,3A4)
998      120 CONTINUE
999      LIN1=.TRUE.
1000      GO TO 450
1001      C      ERROR ON INPUT
1002      122 WRITE(IOUT,123)
1003      123 FORMAT('DEERRR IN INPUT LINE -- TRY AGAIN')
1004      IB=IE
1005      GO TO 135
1006      C
1007      C      INPUT ROUTINE FOR WORKSHEET 2
1008      C
1009      200 IF(LIN2) GO TO 500
1010      TTLWK2=0.0
1011      IF(NURP.EQ.0) GO TO 450
1012      WRITE(IOUT,201)
1013      201 FORMAT('O INPUT FOR WORKSHEET 2 - POINT & URBAN RUNOFF LOADS')
1014      IF(.NOT.LODURB) WRITE(IOUT,202)
1015      IF(LODURB) WRITE(IOUT,203)
1016      202 FORMAT('O FOR POINT SOURCES: ENTER FLOW(MGD), CONCENTRATION(MG/L)' /
1017      &' FOR NON-POINT SOURCES: ENTER UNIT AREA LOADS(KG/KM2/YR)')
1018      203 FORMAT('O ENTER TOTAL LOADS PER SOURCE(KG/YR)')
1019      222 FORMAT(2F10.0)
1020      C

```

```

1021      IP3=0
1022      IP4=0
1023      IP5=0
1024      IP6=0
1025      IP7=0
1026      I=0
1027      205 IB=I+1
1028      DO 210 I=IB,NSRCE
1029          IE=I
1030          IF(IYP(I).GE.3) GO TO 215
1031      210 CONTINUE
1032          LIN2=.TRUE.
1033          GO TO 450
1034      215 WRITE(IOUT,220)(SRCE(I,NAM),NAM=1,28)
1035      220 FORMAT(1X,28A1)
1036          IF(LDURB) GO TO 225
1037          IF(IYP(I).GT.4) GO TO 230
1038          IP3=IP3+1
1039      C      HERE FOR URBAN POINT SOURCES)
1040          READ(IN,222,ERR=223) PFLOW(IP3),PCONC(IP3)
1041          AUNIT(I,2)=PFLOW(IP3)*PCONC(IP3)*1382.0
1042          TTLWK2=TTLWK2+AUNIT(I,2)
1043          GO TO 205
1044      C      ERROR ON INPUT
1045      223 IF(PFLOW(IP3).EQ.FIN) GO TO 450
1046          WRITE(IOUT,123)
1047          WRITE(IOUT,202)
1048          I=IE-1
1049          IP3=IP3-1
1050          GO TO 205
1051      C      HERE FOR URBAN LOADS LOADED DIRECTLY
1052      225 READ(IN,222,ERR=226) AUNIT(I,2)
1053          GO TO 205
1054      C      ERROR ON INPUT
1055      226 IF(AUNIT(I,2).EQ.FIN) GO TO 450
1056          WRITE(IOUT,123)
1057          WRITE(IOUT,203)
1058          I=IE-1
1059          GO TO 205
1060      C      HERE FOR URBAN NON-POINT SOURCES
1061      230 READ(IN,222,ERR=236) UALURB
1062          KODE=IYP(I)-5
1063          GO TO(231,232,233,234),KODE
1064      231 IP4=IP4+1
1065          UALS(IP4)=UALURB
1066          GO TO 235
1067      232 IP5=IP5+1
1068          UALC(IP5)=UALURB
1069          GO TO 235
1070      233 IP6=IP6+1
1071          W2UAL1(IP6)=UALURB
1072          GO TO 235
1073      234 IP7=IP7+1
1074          W2UAL2(IP7)=UALURB
1075      C
1076      235 AUNIT(I,2)=AUNIT(I,1)*UALURB
1077          TTLWK2=TTLWK2+AUNIT(I,2)
1078          GO TO 205
1079      C      ERROR ON INPUT
1080      236 IF(UALURB.EQ.FIN) GO TO 450

```

```

1081      WRITE(IOUT,123)
1082      WRITE(IOUT,202)
1083      I=IF-1
1084      GO TO 275
1085
1086      C      INPUT ROUTINE FOR WORKSHEET 3
1087      C
1088      300 IF(LIN3) GO TO 500
1089      TILWK3=0.0
1090      IF(NRUNCP.EQ.1) GO TO 450
1091      WRITE(IOUT,305)
1092      305 FORMAT('O INPUT FOR WORKSHEET 3 - RURAL NON-CROPLAND AREAS')
1093      IDUM1=IBLNK
1094      IDUM2=IBLNK
1095      IDUM3=IBLNK
1096      IF(OTHPUL.GE.1) IDUM1=ICOMA
1097      IF(OTHRUL.GE.2) IDUM2=ICOMA
1098      IF(OTHPUL.EQ.3) IDUM3=ICOMA
1099      LEN1=LENGTH(TLERU1,12)
1100      IF(LEN1.EQ.-1) LEN1=1
1101      LEN2=LENGTH(TLERU2,12)
1102      IF(LEN2.EQ.-1) LEN2=1
1103      LEN3=LENGTH(TLERU3,12)
1104      IF(LEN3.EQ.-1) LEN3=1
1105      IF(.NOT.LODRNC) WRITE(IOUT,302) IDUM1,(TLERU1(J),J=1,LEN1),IDUM2,
1106      & (TLERJ2(J),J=1,LEN2),IDUM3,(TLERU3(J),J=1,LEN3)
1107      IF(LODRNC) WRITE(IOUT,303) IDUM1,(TLERU1(J),J=1,LEN1),IDUM2,
1108      & (TLERU2(J),J=1,LEN2),IDUM3,(TLERU3(J),J=1,LEN3)
1109      302 FORMAT('O ENTER AREA,UAL FOR: GRASSLAND,WOODLAND',39A1/
1110      & 2X,'AREA IN KM2, UAL IN KG/KM2/YR')
1111      303 FORMAT('O ENTER TOTAL LOADS FOR: GRASSLAND,WOODLAND',39A1/
1112      & 2X,'LOADS ARE KG/YR')
1113      C
1114      IP1=0
1115      I=0
1116      310 IR=I+1
1117      DO 315 I=IR,NSRCE
1118      IC=I
1119      IF(ITYP(I).EQ.1) GO TO 320
1120      315 CONTINUE
1121      LIN3=.TRUE.
1122      GO TO 450
1123      320 WRITE(IOUT,220) (SRCE(I,NAM),NAM=1,28)
1124      IP1=IP1+1
1125      IF(LODRNC) GO TO 335
1126      READ(IN,330,ERR=331) GRASA(IP1),JALG(IP1),WLND4(IP1),UALW(IP1),
1127      & W30TA1(IP1),W3UAL1(IP1),W30TA2(IP1),W3UAL2(IP1),W30TA3(IP1),
1128      & W3UAL3(IP1)
1129      330 FORMAT(10F10.0)
1130      LODRV1(IP1)=GRASA(IP1)*UAL3(IP1)
1131      LODRN2(IP1)=WLND4(IP1)*UALW(IP1)
1132      IF(OTHPUL.EQ.0) GO TO 345
1133      LOD1W3(IP1)=W30TA1(IP1)*W3UAL1(IP1)
1134      LOD2W3(IP1)=W30TA2(IP1)*W3UAL2(IP1)
1135      LOD3W3(IP1)=W30TA3(IP1)*W3UAL3(IP1)
1136      GO TO 245
1137      C      ERROR ON INPUT
1138      331 IF(GRASA(IP1).EQ.FIN) GO TO 450
1139      WRITE(IOUT,123)
1140      WRITE(IOUT,302) IDUM1,(TLERU1(J),J=1,LEN1),IDUM2,(TLERJ2(J),J=1,

```

```

1141      ELLEN2), IDUM3, (TLERU3(J), J=1, LEN3)
1142      I=IF-1
1143      IP1=IP1-1
1144      GO TO 310
1145  C
1146      335 READ(IN, 340, ERR=346) LODRN1(IP1), LODRN2(IP1), LOD1W3(IP1),
1147      & LOD2W3(IP1), LOD3W3(IP1)
1148      340 FORMAT(6F10.0)
1149      345 AJNIT(I, 2)=LODRN1(IP1)+LODRN2(IP1)+LOD1W3(IP1)+LOD2W3(IP1)+
1150      & LOD3W3(IP1)
1151      TTLWK3=TTLWK3+AJNIT(I, 2)
1152      GO TO 310
1153  C      ERROR ON INPUT
1154      346 IF(LODRN1(IP1).EQ.FIN) GO TO 450
1155      WRITE(OUT, 123)
1156      WRITE(OUT, 303) IDUM1, (TLERU1(J), J=1, LEN1), IDUM2, (TLERJ2(J), J=1,
1157      & ELLEN2), IDUM3, (TLERU3(J), J=1, LEN3)
1158      I=IF-1
1159      IP1=IP1-1
1160      GO TO 310
1161  C
1162  C      INPUT ROUTINE FOR WORKSHEET 4
1163  C
1164      400 IF(LIN4) GO TO 500
1165      IF(NRUCRP .EQ. 0) GO TO 450
1166      LODPDR=.FALSE.
1167      WRITE(OUT, 405)
1168      405 FORMAT('O INPUT FOR WORKSHEET 4 - RURAL CROPLAND AREAS')
1169      IF(LODRCP) GO TO 415
1170      WRITE(OUT, 406)
1171      406 FORMAT(' IS A POLLUTANT DELIVERY RATIO KNOWN FOR EACH SOURCE ?')
1172      READ(IN, 407) ANSWER
1173      407 FORMAT(A1)
1174      IF(ANSWER .EQ. ANSYF5) GO TO 412
1175      IF(LIN2 .AND. LIN3) GO TO 409
1176      IEP=0
1177      WRITE(OUT, 408)
1178      408 FORMAT('O CANNOT CALCULATE POLLUTANT DELIVERY LOADS PRIOR TO /
1179      & INPUT OF INITIAL STATE FOR URBAN POINT, & NON-CROPLAND SOURCES')
1180      GO TO 500
1181      409 WRITE(OUT, 410)
1182      410 FORMAT('O ENTER TOTAL POLLUTANT LOAD AT MOUTH OF BASIN(KG/YR)')
1183      READ(IN, 222) PTL0AD
1184      TTLWK4=PTL0AD - TTLWK2 - TTLWK3
1185      WRITE(OUT, 411)
1186      411 FORMAT('O ENTER USLE COEFFICIENTS R,K,LS,C,P FOR EACH SOURCE')
1187      GO TO 420
1188      412 WRITE(OUT, 413)
1189      413 FORMAT('O ENTER USLE COEFFICIENTS R,K,LS,C,P & DELIVERY RATIO FOR E
1190      EACH SOURCE')
1191      LODPDR=.TRUE.
1192      GO TO 420
1193      415 IF(.NOT.L4UAL) WRITE(OUT, 416)
1194      416 FORMAT('O ENTER TOTAL LOADS FOR EACH SOURCE')
1195      IF(L4UAL) WRITE(OUT, 417)
1196      417 FORMAT('O ENTER UNIT AREA LOADS FOR EACH SOURCE(KG/KM2/YR)')
1197      420 IC=0
1198      ER0T0T=0.0
1199      I=0
1200      422 I3=I+1

```

```

1201      DO 425 I=IB,NSRCE
1202      IF=I
1203      IF(IITYP(I).EQ.2) GO TO 430
1204 425 CONTINUE
1205      LIN4=.TRUE.
1206      IF(.NOT.LODPDR) GO TO 436
1207      GO TO 450
1208 437 IC=IC+1
1209      WRITE(IOUT,270)(SRCE(I,NAM),NAM=1,28)
1210      CRPLND=AUNIT(I,1)*100.0
1211      IF(LODRCP) GO TO 440
1212      IF(.NOT.LODPDR) GO TO 437
1213      READ(IN,340,ERR=439)USLER(IC),USLEK(IC),USLELS(IC),USLEC(IC),
1214      & USLEP(IC),RATIO(IC)
1215      GO TO 434
1216 432 READ(IN,340,ERR=439)USLER(IC),USLEK(IC),USLELS(IC),USLEC(IC),
1217      & USLEP(IC)
1218 434 SOILS1(IC)=USLER(IC)*USLEK(IC)*USLELS(IC)*USLEC(IC)*USLEP(IC)
1219      SOILS2(IC)=SOILS1(IC)*2.243
1220      TTLER(IC)=SOILS2(IC)*CRPLND
1221      EROTOT=EROTOT+TTLER(IC)
1222      IF(.NOT.LODPDR) GO TO 422
1223      AJNIT(I,2)=TTLER(IC)*RATIO(IC)
1224      AUNIT(I,2)=AUNIT(I,2)*1000.
1225      GO TO 422
1226 C      NOW CALCULATE PDR GIVEN TOTAL LOAD AND TOTAL EROSION
1227 436 RATVAL=(TTLWK4/1000.)/EROTOT
1228      LODPDR=.TRUE.
1229      IC=0
1230      DO 438 I=1,NSRCE
1231      IF(IITYP(I).NE.2) GO TO 438
1232      IC=IC+1
1233      RATIO(IC)=RATVAL
1234      AJNIT(I,2)=TTLER(IC)*RATVAL
1235      AUNIT(I,2)=AUNIT(I,2)*1000.
1236 438 CONTINUE
1237      GO TO 450
1238 C      ERROR ON INPUT
1239 439 IF(USLER(IC).EQ.FIN) GO TO 450
1240      WRITE(IOUT,123)
1241      IF(LODPDR) WRITE(IOUT,413)
1242      IF(.NOT.LODPDR) WRITE(IOUT,411)
1243      I=I-1
1244      IC=IC-1
1245      GO TO 422
1246 C
1247 440 IF(.NOT.L4UAL) GO TO 442
1248      READ(IN,222,ERR=441)W4UAL(IC)
1249      AUNIT(I,2)=AUNIT(I,1)*W4UAL(IC)
1250      GO TO 422
1251 C      ERROR ON INPUT
1252 441 IF(W4UAL(IC).EQ.FIN) GO TO 450
1253      WRITE(IOUT,123)
1254      WRITE(IOUT,417)
1255      I=I-1
1256      IC=IC-1
1257      GO TO 422
1258 C      HERE FOR INPUT OF LOAD VALUES
1259 442 READ(IN,222,ERR=443)AUNIT(I,2)
1260      GO TO 422

```

```

1251      C          ERROR ON INPUT
1262      443 IF (AUNIT(I,2).EQ.FIN) GO TO 450
1263          WRITE(IOUT,123)
1264          WRITE(IOUT,416)
1265          I=IF-1
1266          IC=IC-1
1267          GO TO 422
1268      C          DONE
1269      450 LINPUT=.TRUE.
1270          IFR=0
1271      500 CONTINUE
1272          RETURN
1273          END
END OF FILE

```

The second source file is called
WRKSHT

```

1  SUBROUTINE WPKSHT(ISTAGE,MTABLE,IER)
2
3  C
4  C ROUTINE TO OUTPUT WORKSHEETS
5  C *****
6  LOGICAL LINPUT,LODURB,LODRNC,LODRCP,L4UAL,LIN1,LIN2,LIN3,LIN4
7  LOGICAL URBCTL,RULCTL,UCTLOD(3),RCTLOD(3),UPRCNT,UXPCNT,RPRCNT,
8  & RXPcnt,URBCST,RULCST,LRANK
9  INTEGER OTHURB,OTHRUL,SRCE(210,28),TYPCOD(9,12)
10 DIMENSION IITYP(210),NPOS(210),AUNIT(210,2),KTITLE(4,40),
11 & SURFEA(210,3),CTL0D(210,3),CTUSLF(3,30,5)
12 DIMENSION STMCTL(30,3),STMARA(30,3),COMCTL(30,3),COMARA(30,3),
13 & OT1CTL(30,3),OT1ARA(30,3),OT2CTL(30,3),OT2ARA(30,3)
14 DIMENSION PFL0W(30),PCONC(30),UALS(30),UALC(30),W2UAL1(30),
15 & W2UAL2(30),TRNSMN(210)
16 INTEGER TLEUR1(12),TLEUR2(12),TLERU1(12),TLERU2(12),TLERU3(12),
17 & APCE(210),AREACN(4),UALCON(4)
18 DIMENSION GRASA(30),UALG(30),WLND(30),UALW(30),W3OTA1(30),
19 & W3UAL1(30),W3UAL2(30),W3UAL3(30),W3OTA2(30),W3OTA3(30)
20 REAL LODRNL(30),LODRN2(30),LOD1W3(30),LOD2W3(30),LOD3W3(30)
21 DIMENSION USLEF(30),USLEK(30),USLELS(30),JSLEC(30),W4UAL(30),
22 & USLEP(30),SOILS1(30),SOILS2(30),TTLEF(30),RATIO(30)
23 DIMENSION PDIF(210,3),TCOST(210,3),PUCOST(210,3)
24 DIMENSION PMOUTH(210),DIFMTH(210,3),CPJREM(210,3),KERANK(210,3)
25 COMMON /WTRID/ IN,LOUT,NSRCE,KTITLE,NTIE,NURB,NRUNC,NRUCRP
26 COMMON /WTRTF/ LINPUT,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHURB,
27 & OTHRUL,LODRCP,TYPCOD,JIN,JOUT,L4UAL,KJUT,LRANK
28 COMMON /WRKSH1/ SRCE,IITYP,NPOS,APCE,AUNIT,SURFEA,TLEUR1,TLEUR2
29 COMMON /WRKSH2/ PFL0W,PCONC,UALS,UALC,W2UAL1,W2UAL2
30 COMMON /WRKSH3/ GRASA,UALG,WLND,UALW,LODRNL,LODRN2,
31 & W3OTA1,W3UAL1,W3UAL2,W3UAL3,W3OTA2,W3OTA3,
32 & TLERU1,TLERU2,TLERU3,LOD1W3,LOD2W3,LOD3W3
33 COMMON /WRKSH4/ USLEF,USLEK,USLELS,USLEP,USLEP,SOILS1,
34 & SOILS2,TTLEF,W4UAL,RATIO
35 COMMON /WRKSH6/ PDIF,TCOST,PUCOST,TRNSMN,PMOUTH,DIFMTH,CPJREM,
36 & KERANK
37 COMMON /WTRCON/ UCTLOD,RCTLOD,URBCTL,RULCTL,UPRCNT,UXPCNT,RPRCNT,
38 & PXPcnt,OPTVAL,NOPT,URBCST,RULCST,AREACN,UALCON
39 DIMENSION CTLEFLW(30,3),CTLCNC(30,3)
40 DIMENSION CTLE(30,3),SOILC1(30,3),SOILC2(30,3),CTLER0(30,3),
41 & CUALW4(30,3)
42 DIMENSION TOTCST(3),TOTREM(3),KNDX(540),FINCST(540),FINREM(540),
43 & ISFIN(540)
44 INTEGER FINSRC(540,43)
45 COMMON /WRKSH8/ TOTCST,TOTREM,TOTLOD,TOTMTH,KNDX,FINSRC,FINCST,
46 & FINREM,ISFIN,NUM
47 COMMON /UBNCTL/ CTLEFLW,CTLCNC,CTL0D,STMCTL,STMARA,COMCTL,
48 & COMARA,OT1CTL,OT1ARA,OT2CTL,OT2ARA
49 COMMON /REFCTL/ CTUSLF,CTLE,SOILC1,SOILC2,CTLER0,CUALW4
50 C *****
51 LOGICAL SAMNAM,SCHSRC
52 DATA IPG,ITM,IBLNK/'1','2',' '
53 REAL LT1,LT2,LT3,LT4,LT5,LT6,STGTOT(3),DIFTOT(3)
54 INTEGER NWRKSH(8),STYP(12),TMPSRC(28),CRPSRC(28)
55 DO 3 I=1,28
56 CRPSRC(I)=IBLNK
57 IF(I.GT.10 .AND. I.LT.21) CRPSRC(I)=TYPCOD(12,I-10)
58 3 CONTINUE
59 IER=1

```

```

60      IFF=ITM
61      IF(KOUT.NE.IOUT) IFF=IPG
62      IF(.NOT.LINPUT .OR. MTABLE.EQ.0) GO TO 939
63      LK=8
64      DO 5 I=1,8
65          NWRKSH(I)=0
66          IF(MTABLE.EQ.9) NWRKSH(I)=I
67      5 CONTINUE
68      IF(MTABLE.NE.9) GO TO 35
69      LK=0
70      16 LK=LK+1
71          IF(LK.GT.8) GO TO 899
72      C
73      30 MTABLE=NWRKSH(LK)
74      35 GO TO(100,200,300,400,500,600,700,800),MTABLE
75      C
76      C      OUTPUT ROUTINE FOR WORKSHEET 1
77      C
78      100 WRITE(KOUT,101) IFF
79      101 FORMAT(A1,3X,'PHYSICAL LAYOUT')
80      IF(NTLE.EQ.0) GO TO 108
81      DO 105 I=1,NTLE
82      105 WRITE(KOUT,106) (KTITLE(I,J),J=1,40)
83      106 FORMAT(13X,40A1)
84      GO TO 110
85      108 WRITE(KOUT,109)
86      109 FORMAT('0')
87      110 WRITE(KOUT,111)
88      111 FORMAT(94X,'WORKSHEET 1'/1X,52('-'),52('-')/59X,'POINT',6X,'AREA',
89          66X,'UNITS'/
90          66X,'SOURCE',22X,'TYPE',8X,'POSITION',3X,'OF ENTRY',5X,'KM2',6X,
91          6'SERVED',4X,'SURFACE FEATURES'/1X,52('-'),52('-'))
92      C
93      LT1=0.0
94      LT2=0.0
95      DO 120 I=1,NSRCE
96      120 KODE=ITYP(I)
97      DO 115 J=1,12
98          STYP(J)=TYPCOD(KODE,J)
99      115 CONTINUE
100      IF(KODE.EQ.3 .OR. KODE.EQ.4) GO TO 125
101      LT1=LT1+AUNIT(I,1)
102      SAMNAM=CHSRC(I,SRCE,KODE)
103      IF(SAMNAM) GO TO 114
104      WRITE(KOUT,118) (SRCE(I,J),J=1,28),(STYP(J),J=1,12),
105          5 NPOS(I),APOE(I),AUNIT(I,1),(SURFEA(I,IS),IS=1,3)
106      118 FORMAT(1X,28A1,1X,12A1,19,10X,A1,F13.1,16X,3A4)
107      GO TO 120
108      114 WRITE(KOUT,117) (STYP(J),J=1,12),NPOS(I),APOE(I),AUNIT(I,1),
109          5 (SURFEA(I,IS),IS=1,3)
110      117 FORMAT(30X,12A1,19,10X,A1,F13.1,16X,3A4)
111      GO TO 120
112      125 LT2=LT2+AUNIT(I,1)
113      WRITE(KOUT,116) (SRCE(I,J),J=1,28),(STYP(J),J=1,12),NPOS(I),
114          5 APOE(I),AUNIT(I,1),(SURFEA(I,IS),IS=1,3)
115      116 FORMAT(1X,28A1,1X,12A1,19,10X,A1,13X,F11.0,5X,3A4)
116      GO TO 120
117      120 CONTINUE
118      WRITE(KOUT,127) LT1,LT2
119      127 FORMAT(1H0,20X,'TOTAL ',36X,F13.1,F11.0)

```

```

120      WRITE(KOUT,130)
121      FORMAT(1X,52(' '),52(' '))
122      GO TO 16
123      C
124      C      OUTPUT ROUTINE FOR WORKSHEET 2
125      C
126      200 IF(NURB.EQ.0) GO TO 16
127      NS=0
128      WRITE(KOUT,201) IFF
129      201 FORMAT(A1,43X,'POINT AND URBAN RUNOFF LOADS' /47X,
130      &'TOTAL POLLUTANT LOADS')
131      IF(OTHURB.EQ.0) WRITE(KOUT,202)
132      IF(OTHURB.EQ.1) WRITE(KOUT,203)(TLFUR1(JT),JT=1,12)
133      IF(OTHURB.EQ.2) WRITE(KOUT,204)(TLFUR1(JT),JT=1,12),
134      &(TLFUR2(JT),JT=1,12)
135      WRITE(KOUT,207)
136      207 FORMAT(5X,'INITIAL CONDITION' /5X,17(' '))
137      LT1=0.0
138      LT2=0.0
139      LT3=0.0
140      LT4=0.0
141      LT5=0.0
142      IT=0
143      I=0
144      205 IB=I+1
145      DO 206 I=IB,NSRCE
146      IF(ITYP(I).EQ.3 .OR. ITYP(I).EQ.4) GO TO 210
147      206 CONTINUE
148      GO TO 220
149      210 IT=IT+1
150      I1=I+1
151      I2=I+2
152      I3=I+3
153      I4=I+4
154      IF(OTHURB.EQ.0)WRITE(KOUT,211)(SRCE(I,J),J=1,28),NPOS(I),PFLOW(IT)
155      &.PCONC(IT),AUNIT(I,2),AUNIT(I1,1),UALS(IT),AUNIT(I1,2),AUNIT(I2,1)
156      &.UALC(IT),AUNIT(I2,2)
157      IF(OTHURB.EQ.1)WRITE(KOUT,212)(SRCE(I,J),J=1,28),NPOS(I),PFLOW(IT)
158      &.PCONC(IT),AUNIT(I,2),AUNIT(I1,1),UALS(IT),AUNIT(I1,2),AUNIT(I2,1)
159      &.UALC(IT),AUNIT(I2,2),AUNIT(I3,1),W2UAL1(IT),AUNIT(I3,2)
160      IF(OTHURB.EQ.2)WRITE(KOUT,213)(SRCE(I,J),J=1,28),NPOS(I),PFLOW(IT)
161      &.PCONC(IT),AUNIT(I,2),AUNIT(I1,1),UALS(IT),AUNIT(I1,2),AUNIT(I2,1)
162      &.UALC(IT),AUNIT(I2,2),AUNIT(I3,1),W2UAL1(IT),AUNIT(I3,2),
163      &.AUNIT(I4,1),W2UAL2(IT),AUNIT(I4,2)
164      LT1=LT1+AUNIT(I,2)
165      LT2=LT2+AUNIT(I1,2)
166      LT3=LT3+AUNIT(I2,2)
167      LT4=LT4+AUNIT(I3,2)
168      LT5=LT5+AUNIT(I4,2)
169      GO TO 205
170      220 IF(OTHURB.EQ.0)WRITE(KOUT,221) LT1,LT2,LT3
171      IF(OTHURB.EQ.1)WRITE(KOUT,222) LT1,LT2,LT3,LT4
172      IF(OTHURB.EQ.2)WRITE(KOUT,223) LT1,LT2,LT3,LT4,LT5
173      C
174      C      OUTPUT CONTROL STRATEGIES
175      C
176      IF(.NOT.URBCT) GO TO 225
177      NS=NS+1
178      WRITE(KOUT,219) NS
179      219 FORMAT(1H0,4X,'STAGE',I2,/5X,7(' '))

```

```

180      LT1=0.0
181      LT2=0.0
182      LT3=0.0
183      LT4=0.0
184      LT5=0.0
185      I=0
186      IT=0
187      214  IR= I+1
188      DO 215  I=IR,NSRCE
189          IF(ITYP(I).EQ.3.OR.ITYP(I).EQ.4) GO TO 216
190      215  CONTINUE
191      GO TO 217
192      216  IT= IT+1
193          II= I+1
194          I2= I+2
195          I3= I+3
196          I4= I+4
197          IF(OTHURB.EQ.0) WRITE(KOUT,211)(SRCE(I,J),J=1,28),NPOS(I),
198      &CTLFLW(IT,NS),CTLCNC(IT,NS),CTLQD(I,NS),STMARA(IT,NS),
199      &STMCTL(IT,NS),CTLQD(I1,NS),COMARA(IT,NS),COMCTL(IT,NS),
200      &CTLQD(I2,NS)
201          IF(OTHURB.EQ.1) WRITE(KOUT,212)(SRCE(I,J),J=1,28),NPOS(I),
202      &CTLFLW(IT,NS),CTLCNC(IT,NS),CTLQD(I,NS),STMARA(IT,NS),
203      &STMCTL(IT,NS),CTLQD(I1,NS),COMARA(IT,NS),COMCTL(IT,NS),
204      &CTLQD(I2,NS),OT1ARA(IT,NS),OT1CTL(IT,NS),CTLQD(I3,NS)
205          IF(OTHURB.EQ.2) WRITE(KOUT,213)(SRCE(I,J),J=1,28),NPOS(I),
206      &CTLFLW(IT,NS),CTLCNC(IT,NS),CTLQD(I,NS),STMARA(IT,NS),
207      &STMCTL(IT,NS),CTLQD(I1,NS),COMARA(IT,NS),COMCTL(IT,NS),
208      &CTLQD(I2,NS),OT1ARA(IT,NS),OT1CTL(IT,NS),CTLQD(I3,NS),
209      &OT2ARA(IT,NS),OT2CTL(IT,NS),CTLQD(I4,NS)
210      LT1=LT1+CTLQD(I,NS)
211      LT2=LT2+CTLQD(I1,NS)
212      LT3=LT3+CTLQD(I2,NS)
213      LT4=LT4+CTLQD(I3,NS)
214      LT5=LT5+CTLQD(I4,NS)
215      GO TO 214
216      217  IF(OTHURB.EQ.0) WRITE(KOUT,221) LT1,LT2,LT3
217          IF(OTHURB.EQ.1) WRITE(KOUT,222) LT1,LT2,LT3,LT4
218          IF(OTHURB.EQ.2) WRITE(KOUT,223) LT1,LT2,LT3,LT4,LT5
219      218  IF(NS.GE.ISTAGF) GO TO 225
220      GO TO 224
221      225  IF(OTHURB.EQ.0) WRITE(KOUT,227)
222      FORMAT(1X,22(' '),79(' '))
223      IF(OTHURB.EQ.1) WRITE(KOUT,228)
224      FORMAT(1X,50(' '),60(' '))
225      IF(OTHURB.EQ.2) WRITE(KOUT,226)
226      FORMAT(1X,64(' '),65(' '))
227      GO TO 16
228      202  FORMAT(80X,'WORKSHEET 2'/1X,22(' '),79(' ')/45X,'POINT',16X,
229      &'STORM',16X,'COMBINED'/30X,'POS FLOW CONC. LOAD',
230      &2(4X,'AREA UAL LOAD')/6X,'SOURCE',18X,'TION',4X,'MGD',4X,
231      &'MG/L',3X,'KG/YR',2(3X,'KM2 KG/KM2/YR KG/YR')/1X,22(' '),79(' '))
232      203  FORMAT(98X,'WORKSHEET 2'/1X,50(' '),60(' ')/41X,'POINT',15X,
233      &'STORM',14X,'COMBINED',11X,12A1/30X,'POS FLOW CONC. LOAD',
234      &3(4X,'AREA UAL LOAD')/6X,'SOURCE',18X,'TION MGD MG/L KG/YR'
235      &3(3X,'KM2',9X,'KG/YR')/1X,50(' '),60(' '))
236      204  FORMAT(80X,37X,'WORKSHEET 2'/1X,64(' '),65(' ')/41X,'POINT',15X,
237      &'STORM',14X,'COMBINED',4X,2(7X,12A1)/30X,'POS FLOW CONC. LOAD'
238      &4(4X,'AREA UAL LOAD')/6X,'SOURCE',18X,'TION MGD MG/L KG/YR'
239      &4(3X,'KM2',8X,'KG/YR')/1X,64(' '),65(' '))

```

```

240 211 FORMAT(1X,28A1,14,2X,3(2F7.1,F8.0))
241 212 FORMAT(1X,28A1,14,F7.1,F5.1,F8.0,3(F7.1,F6.0,F7.0))
242 213 FORMAT(1X,28A1,14,F7.1,F5.1,F8.0,4(F7.1,F6.0,F7.0))
243 221 FORMAT(21X,'TOTAL',9X,3(12X,F10.0))
244 222 FORMAT(21X,'TOTAL',T44,F10.0,T63,F10.0,T83,F10.0,T101,F10.0)
245 223 FORMAT(21X,'TOTAL',T44,F10.0,T63,F10.0,T83,F10.0,T101,F10.0,
246      ET110,F10.0)
247 C
248 C      OUTPUT ROUTINE FOR WORKSHEET 3
249 C
250 300 IF(NRUNCP.EQ.0) GO TO 16
251      WRITE(KOUT,301) IFF
252 301 FORMAT(A1,45X,'RURAL NON-CROPLAND AREAS'/47X,'TOTAL POLLUTANT ',
253      &'LOADS')
254      IF(OTHRUL.EQ.0)WRITE(KOUT,302)
255      IF(OTHRUL.EQ.1)WRITE(KOUT,303)(TLERU1(JT),JT=1,12)
256      IF(OTHRUL.EQ.2)WRITE(KOUT,304)(TLERU1(JT),JT=1,12),
257      &      (TLERU2(JT),JT=1,12)
258      IF(OTHRUL.EQ.3)WRITE(KOUT,305)(TLERU1(JT),JT=1,12),
259      &      (TLERU2(JT),JT=1,12),(TLERU3(JT),JT=1,12)
260      AT1=0.0
261      AT2=0.0
262      LT1=0.0
263      LT2=0.0
264      LT3=0.0
265      LT4=0.0
266      LT5=0.0
267      LT6=0.0
268      IT=0
269      I=0
270 306 IB=I+1
271      DO 307 I=IB,NSRCE
272          IF(ITYP(I).EQ.1) GO TO 310
273 307 CONTINUE
274      GO TO 317
275 310 IT=IT+1
276      IF(OTHRUL.EQ.0)
277          &WRITE(KOUT,311)(SRCE(I,J),J=1,28),NPOS(I),GRASA(IT),UALG(IT),
278      &      LODRN1(IT),WLND4(IT),UALW(IT),LODRN2(IT),AUNIT(I,2)
279      IF(OTHRUL.EQ.1)WRITE(KOUT,312)(SRCE(I,J),J=1,28),NPOS(I),GRASA(IT)
280      &      UALG(IT),LODRN1(IT),WLND4(IT),UALW(IT),LODRN2(IT),W3OTAL1(IT),
281      &      W3UAL1(IT),LOD1W3(IT),AUNIT(I,2)
282      IF(OTHRUL.EQ.2)WRITE(KOUT,313)(SRCE(I,J),J=1,28),NPOS(I),GRASA(IT)
283      &      UALG(IT),LODRN1(IT),WLND4(IT),UALW(IT),LODRN2(IT),W3OTAL1(IT),
284      &      W3JAL1(IT),LOD1W3(IT),W3TA2(IT),W3UAL2(IT),LOD2W3(IT),AUNIT(I,2)
285      IF(OTHRUL.EQ.3)WRITE(KOUT,314)(SRCE(I,J),J=1,28),NPOS(I),GRASA(IT)
286      &      UALG(IT),LODRN1(IT),WLND4(IT),UALW(IT),LODRN2(IT),W3OTAL1(IT),
287      &      W3UAL1(IT),LOD1W3(IT),W3TA2(IT),W3UAL2(IT),LOD2W3(IT),W3OTA3(IT),
288      &      W3UAL3(IT),LOD3W3(IT),AUNIT(I,2)
289      AT1=AT1+GRASA(IT)
290      AT2=AT2+WLND4(IT)
291      LT1=LT1+LODRN1(IT)
292      LT2=LT2+LODRN2(IT)
293      LT3=LT3+AUNIT(I,2)
294      LT4=LT4+LOD1W3(IT)
295      LT5=LT5+LOD2W3(IT)
296      LT6=LT6+LOD3W3(IT)
297      GO TO 306
298 317 IF(OTHRUL.EQ.0)WRITE(KOUT,318)AT1,AT2,LT1,LT2,LT3
299      IF(OTHRUL.EQ.1)WRITE(KOUT,319)AT1,AT2,LT1,LT2,LT4,LT3

```

```

300      IF(OTHRUL.EQ.2)WRITE(KOUT,320)AT1,AT2,LT1,LT2,LT4,LT5,LT3
301      IF(OTHRUL.EQ.3)WRITE(KOUT,321)AT1,AT2,LT1,LT2,LT4,LT5,LT6,LT3
302      IF(OTHRUL.EQ.0)WRITE(KOUT,322)
303      322 FORMAT(1X,14(' '),79(' '))
304      IF(OTHRUL.EQ.1.OR.OTHRUL.EQ.2)WRITE(KOUT,323)
305      323 FORMAT(1X,54(' '),60(' '))
306      IF(OTHRUL.EQ.3)WRITE(KOUT,324)
307      324 FORMAT(1X,60(' '),72(' '))
308      GO TO 16
309      302 FORMAT(81X,' WORKSHEET 3'/1X,14(' '),79(' ')/43X,' GRASSLAND',15X,
310      &' WOODLAND',13X,' TOTAL'/30X,' POSI',2(5X,' AREA',3X,' UAL',4X,' LOAD ')
311      &,7X,' LOAD'/6X,' SOURCE',18X,' TION',2(4X,' KM2 KG/KM2/YR KG/YR'),7X,
312      &' KG/YR'/1X,14(' '),79(' ')/)
313      303 FORMAT(50X,52X,' WORKSHEET 3'/1X,54(' '),60(' ')/43X,' GRASSLAND',
314      &15X,' WOODLAND',13X,12A1,8X,' TOTAL'/30X,' POSI',
315      &3(5X,' AREA',3X,' UAL',4X,' LOAD '),6X,' LOAD'/6X,' SOURCE',18X,
316      &' TION',3(4X,' KM2 KG/KM2/YR KG/YR'),6X,' KG/YR'/,
317      &1X,54(' '),60(' ')/)
318      304 FORMAT(50X,52X,' WORKSHEET 3'/1X,54(' '),60(' ')/39X,' GRASSLAND',
319      &10X,' WOODLAND',9X,12A1,6X,12A1,3X,' TOTAL'/30X,' POSI',
320      &4(2X,' AREA',2X,' UAL',2X,' LOAD '),2X,' LOAD'/6X,' SOURCE',18X,' TION',
321      &4(2X,' KM2',8X,' KG/YR'),2X,' KG/YR'/1X,54(' '),60(' ')/)
322      305 FORMAT(60X,60X,' WORKSHEET 3'/1X,60(' '),72(' ')/39X,' GRASSLAND',
323      &10X,' WOODLAND',3X,3(6X,12A1),3X,' TOTAL'/30X,' POSI',
324      &5(2X,' AREA',2X,' UAL',2X,' LOAD '),2X,' LOAD'/6X,' SOURCE',18X,' TION',
325      &5(2X,' KM2',8X,' KG/YR'),2X,' KG/YR'/1X,60(' '),72(' '))
326      311 FORMAT(1X,28A1,14,2X,2(F8.1,F7.1,F8.1),F12.1)
327      312 FORMAT(1X,28A1,14,2X,3(F8.1,F7.1,F8.1),F11.1)
328      313 FORMAT(1X,28A1,14,1X,4(F7.1,F5.1,F6.1),F8.1)
329      314 FORMAT(1X,28A1,14,1X,5(F7.1,F5.1,F6.1),F8.1)
330      318 FORMAT(1H0,20X,' TOTAL',5X,F12.1,3X,F12.1,F8.1,3X,2F12.1)
331      319 FORMAT(21X,' TOTAL',T32,F12.1,T55,F12.1/1X,T49,F11.1,T71,F11.1,
332      &T94,F11.1,F12.1)
333      320 FORMAT(21X,' TOTAL',T32,F9.1,T50,F9.1/1X,T43,F10.1,T61,F10.1,T79,
334      &F13.1,T97,F10.1,F9.1)
335      321 FORMAT(21X,' TOTAL',T32,F9.1,T50,F9.1/1X,T43,F10.1,T61,F10.1,T79,
336      &F10.1,T97,F10.1,T115,F10.1,F9.1)
337      C
338      C      OUTPUT ROUTINE FOR WORKSHEET 4
339      C
340      400 IF(NRUCRP.EQ.0) GO TO 16
341      NS=0
342      IF(.NOT.LODRCP)WRITE(KOUT,401) IFF
343      IF(LODRCP) WRITE(KOUT,430) IFF
344      401 FORMAT(41,47X,' RURAL CROPLAND AREAS'/27X,' UNIVERSAL SOIL LOSS ',
345      &' EQUATION/TOTAL POLLUTANT LOAD ESTIMATES'/59X,59X,' WORKSHEET 4'/
346      &1X,65(' '),65(' ')/35X,' CROPLAND',30X,' SOIL',5X,' SOIL',4X,' TOTAL',
347      &28X,' TOTAL'/30X,' POSI',3X,' AREA',8X,' USLE COEFFICIENTS',7X,' LOSS',
348      &5X,' LOSS',3X,' EROSION',3X,' DELIVERY REDUCTION',5X,' LOAD'/6X,
349      &' SOURCE',18X,' TION',3X,' (HA)',5X,' R',4X,' K',5X,' LS',4X,' C',4X,' D',
350      &2X,' T/AC/YR MT/HA/YR MT/YR',5X,' RATIO',4X,' EFFICIENCY',4X,
351      &' MT/YR'/1X,65(' '),65(' '))
352      430 FORMAT(41,41X,' RURAL CROPLAND AREAS'/73X,' WORKSHEET 4'/1X,84(' ')/
353      &548X,' CROPLAND',24X,' TOTAL'/50X,' AREA',12X,' UAL',11X,' LOAD'/6X,
354      &' SOURCE',18X,' POSITION',12X,' KM2',10X,' KG/KM2/YR',8X,' KG/YR'/
355      &1X,84(' '))
356      WRITE(KOUT,207)
357      T1=0.0
358      T2=0.0
359      , ST3=0.0

```

```

360      IT=0
361      I=0
362      404 IB=I+1
363      DO 405 I=IB,NSRCE
364          IF(IITYP(I).EQ.2) GO TO 410
365      405 CONTINUE
366      GO TO 420
367      410 IT=IT+1
368      CRPLND=AUNIT(I,1)*100.0
369      CRPLND=AUNIT(I,2)/100.0
370      IF(L4UAL .OR. LODRCP) GO TO 413
371      WRITE(KOUT,411)(SRCE(I,J),J=1,28),NPOS(I),CRPLND,USLER(IT),
372      & USLEK(IT),USLELS(IT),USLEEC(IT),USLEP(IT),SOILS1(IT),SOILS2(IT),
373      & TLER(IT),RATIO(IT),CRPLND
374      411 FORMAT(1X,28A1,14,F10.0,F6.0,F5.2,2F6.3,F5.2,F7.1,F9.1,F10.0,
375      & F10.6,8X,F13.1)
376      GO TO 419
377      413 WRITE(KOUT,431)(SRCE(I,J),J=1,28),NPOS(I),AJNIT(I,1),W4UAL(IT),
378      & AUNIT(I,2)
379      431 FORMAT(1X,28A1,16,F20.0,F15.1,F15.0)
380      419 T1=CRPLND
381      T2=T2+TLER(IT)
382      ST3=ST3+CRPLND
383      GO TO 404
384      420 IF(.NOT.LODRCP)WRITE(KOUT,421) T1,T2,ST3
385      IF(LODRCP) WRITE(KOUT,434) T1,ST3
386      434 FORMAT(21X,'TOTAL',15X,F14.0,16X,F14.0)
387      421 FORMAT(21X,'TOTAL',T30,F14.0,T68,20X,F10.0,10X,
388      & 5X,'NA',F14.1)
389      IF(.NOT.RULCTL) GO TO 425
390      C
391      C      RURAL CROPLAND CONTROL STRATEGIES
392      C
393      412 NS=NS+1
394      WRITE(KOUT,219) NS
395      ST1=0.0
396      ST2=0.0
397      ST3=0.0
398      T2=0.0
399      IT=0
400      I=0
401      414 IB=I+1
402      DO 415 I=IB,NSRCE
403          IF(IITYP(I).EQ.2) GO TO 416
404      415 CONTINUE
405      GO TO 418
406      416 IT=IT+1
407      CRPLND=AUNIT(I,1)*100.0
408      CRPLND=CTLOD(I,NS)/100.0
409      IF(.NOT.LODRCP)WRITE(KOUT,417)(SRCE(I,J),J=1,28),NPOS(I),CRPLND,
410      & (CTUSLE(NS,IT,J),J=1,5),SOILC1(IT,NS),
411      & SOILC2(IT,NS),CYLFRO(IT,NS),RATIO(IT),CTLEF(IT,NS),CRPLND
412      & IF(LODRCP)WRITE(KOUT,431)(SRCE(I,J),J=1,28),NPOS(I),AUNIT(I,1),
413      & (UAIW4(IT,NS),CTLOD(I,NS)
414      417 FORMAT(1X,28A1,14,F10.0,F6.0,F5.2,2F6.3,F5.2,F7.1,F9.1,F10.0,F10.6
415      & ,F8.2,F13.1)
416      T2=T2+CTLFRO(IT,NS)
417      ST3=ST3+CRPLND
418      GO TO 414
419      418 IF(.NOT.LODRCP)WRITE(KOUT,424) T1,T2,ST3

```

```

420      IF (LUDRCP)      WRITE(KOUT,434) TL,ST3
421      424 FORMAT(21X,'TOTAL',T30,F14.0,T68,20X,F10.0,
422      & 18X,F13.1)
423      IF(MS.GE.1STAGE) GO TO 425
424      GO TO 412
425      IF(.NOT.LUDRCP) WRITE(KOUT,422)
426      IF (LUDRCP)      WRITE(KOUT,433)
427      433 FORMAT(1X,84(' '))
428      422 FORMAT(1X,65(' '),65(' '))
429      GO TO 16
430
431      C
432      C      OUTPUT ROUTINE FOR WORKSHEET 5
433      C
434      500 IF(.NOT.URBCTL .AND. .NOT.RULCTL) GO TO 16
435      TTL0D=0.0
436      DO 501 I=1,3
437      SIGTOT(I)=0.0
438      501 DIFTOT(I)=0.0
439      WRITE(KOUT,502) IFF
440      IF(ISTAGE .EQ. 1) WRITE(KOUT,504)
441      IF(ISTAGE .EQ. 2) WRITE(KOUT,503)
442      IF(ISTAGE .EQ. 3) WRITE(KOUT,506)
443      502 FORMAT(A1,50X,'LOADING SUMMARY',47X,'TOTAL POLLUTANT LOADS')
444      503 FORMAT(99X,'WORKSHEET 5',1X,50(' '),60(' ')/53X,
445      & 'INITIAL LOAD TO CONTROLLED LOAD',6X,'LOAD REDUCTIONS',53X,
446      & 'WATERSHED BASIN',9X,'KG/YR',17X,'KG/YR',6X,'SOURCE',22X,'TYPE',
447      & 4X,'POSITION',9X,'KG/YR',4X,2(4X,'STAGE 1',4X,'STAGE 2')/
448      & 1X,50(' '),60(' '))
449      504 FORMAT(99X,'WORKSHEET 5',1X,50(' '),60(' ')/53X,
450      & 'INITIAL LOAD TO',53X,'WATERSHED BASIN CONTROLLED LOAD',6X,
451      & 'LOAD REDUCTIONS',6X,'SOURCE',22X,'TYPE',4X,'POSITION',9X,
452      & 'KG/YR',2(14X,'KG/YR',3X)/1X,50(' '),60(' '))
453      506 FORMAT(60X,61X,'WORKSHEET 5',1X,66(' '),66(' ')/53X,
454      & 'INITIAL LOAD TO',10X,'CONTROLLED LOAD',18X,'LOAD REDUCTIONS',
455      & 53X,'WATERSHED BASIN',15X,'KG/YR',28X,'KG/YR',6X,'SOURCE',22X,
456      & 'TYPE',4X,'POSITION',8X,'KG/YR',4X,
457      & 2(4X,'STAGE 1',4X,'STAGE 2',4X,'STAGE 3')/1X,66(' '),66(' '))
458      DO 550 I=1,NSPCF
459      KODE=ITYP(I)
460      DO 507 J=1,12
461      STYP(J)=TYPCOD(KODE,J)
462      SAMNAM=SCHSRC(I,SRCE,KODE)
463      IF(ITYP(I).NE.1 .AND. AUNIT(I,2).NE.0.0) GO TO 505
464      IF(.NOT.SAMNAM) WRITE(KOUT,526) (SRCE(I,J),J=1,28),
465      & (STYP(J),J=1,12),NPOS(I),AUNIT(I,2)
466      IF(SAMNAM) WRITE(KOUT,531) (STYP(J),J=1,12),NPOS(I),AUNIT(I,2)
467      GO TO 548
468      505 CONTINUE
469      IF(SAMNAM) GO TO 530
470      IF(ISTAGE.EQ.1) WRITE(KOUT,526) (SRCE(I,J),J=1,28), (STYP(J),J=1,12),
471      & NPOS(I),AUNIT(I,2),CTL0D(I,1),PDIF(I,1)
472      IF(ISTAGE.EQ.2) WRITE(KOUT,527) (SRCE(I,J),J=1,28), (STYP(J),J=1,12),
473      & NPOS(I),AUNIT(I,2), (CTL0D(I,J),J=1,2), (PDIF(I,J),J=1,2)
474      IF(ISTAGE.EQ.3) WRITE(KOUT,527) (SRCE(I,J),J=1,28), (STYP(J),J=1,12),
475      & NPOS(I),AUNIT(I,2), (CTL0D(I,J),J=1,3), (PDIF(I,J),J=1,3)
476      GO TO 540
477      530 IF(ISTAGE.EQ.1) WRITE(KOUT,531) (STYP(J),J=1,12),
478      & NPOS(I),AUNIT(I,2),CTL0D(I,1),PDIF(I,1)
479      IF(ISTAGE.EQ.2) WRITE(KOUT,53) (STYP(J),J=1,12),
480      & NPOS(I),AUNIT(I,2), (CTL0D(I,J),J=1,2), (PDIF(I,J),J=1,2)

```

```

480      IF (ISTAGE.EQ.3) WRITE(KOUT,533) (STYP(J),J=1,12),
481      & NPOS(1),AUNIT(1,2), (CTLDD(1,J),J=1,3), (PDIF(1,J),J=1,3)
482 540 DO 545 J=1,ISTAGE
483     STGTOT(J)=STGTOT(J)+CTLDD(1,J)
484 545 DIFTOT(J)=DIFTOT(J)+PDIF(1,J)
485 548 TTLDD=TTLDD+AUNIT(1,2)
486 550 CONTINUE
487 C
488     IF (ISTAGE.EQ.1) WRITE(KOUT,554) TTLDD,STGTOT(1),DIFTOT(1)
489     IF (ISTAGE.EQ.2) WRITE(KOUT,556) TTLDD, (STGTOT(J),J=1,2),
490     & (DIFTOT(J),J=1,2)
491     IF (ISTAGE.EQ.3) WRITE(KOUT,556) TTLDD, (STGTOT(J),J=1,3),
492     & (DIFTOT(J),J=1,3)
493     IF (ISTAGE.LT.3) WRITE(KOUT,565)
494     IF (ISTAGE.EQ.3) WRITE(KOUT,566)
495 566 FORMAT(1X,66(' '),66(' '))
496 526 FORMAT(1X,28A1,1X,12A1,15,6X,F12.1,F18.1,F22.1)
497 527 FORMAT(1X,28A1,1X,12A1,15,6X,F12.1,2X,6F11.1)
498 554 FORMAT('0',20X,'TOTAL',27X,F12.1,F18.1,F22.1)
499 556 FORMAT('0',20X,'TOTAL',27X,F12.1,2X,6F11.1)
500 531 FORMAT(30X,12A1,15,6X,F12.1,F18.1,F22.1)
501 533 FORMAT(30X,12A1,15,6X,F12.1,2X,6F11.1)
502 565 FORMAT(1X,50(' '),60(' '))
503     GO TO 16
504 C
505 C     OUTPUT ROUTINE FOR WORKSHEET 6
506 C
507 600 IF (.NOT.URBCTL .AND. .NOT.RULCTL) GO TO 16
508     WRITE(KOUT,605) IFF
509 605 FORMAT(A1,51X,'PROGRAM COSTS'/44X,'TOTAL POLLUTANT REDUCTIONS')
510     IF (ISTAGE.EQ.1) WRITE(KOUT,606)
511     IF (ISTAGE.EQ.2) WRITE(KOUT,610)
512     IF (ISTAGE.EQ.3) WRITE(KOUT,612)
513 606 FORMAT(99X,'WORKSHEET 6'/1X,50(' '),60(' ')/53X,'AREA'/
514     & 51X,'TREATED',4X,'UNITS',6X,'PER UNIT COSTS',8X,'TOTAL COSTS'/
515     & 6X,'SOURCE',22X,'TYPE',4X,'POSITION',3X,'KM2',5X,'SERVED',11X,
516     & '$/YR',16X,'$/YR'/1X,50(' '),60(' '))
517 610 FORMAT(99X,'WORKSHEET 6'/1X,50(' '),60(' ')/53X,'AREA',16X,
518     & 'PER UNIT COSTS',8X,'TOTAL COSTS'/51X,'TREATED',4X,'UNITS',11X,
519     & '$/YR',16X,'$/YR'/6X,'SOURCE',22X,'TYPE',4X,'POSITION',3X,
520     & 'KM2',5X,'SERVED',2(4X,'STAGE 1',4X,'STAGE 2')/1X,50(' '),60(' '))
521 612 FORMAT(60X,61X,'WORKSHEET 6'/1X,66(' '),66(' ')/53X,'AREA',
522     & 21X,'PER UNIT COSTS',21X,'TOTAL COSTS'/51X,'TREATED',4X,'UNITS',
523     & 16X,'$/YR',29X,'$/YR'/6X,'SOURCE',22X,'TYPE',4X,'POSITION',
524     & 3X,'KM2',5X,'SERVED',2(4X,'STAGE 1',4X,'STAGE 2',4X,'STAGE 3')/
525     & 1X,66(' '),66(' '))
526 C
527     DO 615 I=1,3
528 615 STGTOT(I)=0.0
529     DO 620 I=1,NSRCE
530     IF (ITYP(I).EQ.1 .OR. AUNIT(I,2).EQ.0.0) GO TO 630
531     IF (ITYP(I).GE.3) GO TO 620
532 C
533     IF (ISTAGE.EQ.1) WRITE(KOUT,616) (SRCE(I,J),J=1,28),NPOS(1),
534     & AUNIT(1,1),PUICOST(1,1),TCOST(1,1)
535     IF (ISTAGE.EQ.2) WRITE(KOUT,619) (SRCE(I,J),J=1,28),NPOS(1),
536     & AUNIT(1,1), (PUICOST(1,J),J=1,2), (TCOST(1,J),J=1,2)
537     IF (ISTAGE.EQ.3) WRITE(KOUT,619) (SRCE(I,J),J=1,28),NPOS(1),
538     & AUNIT(1,1), (PUICOST(1,J),J=1,3), (TCOST(1,J),J=1,3)
539     GO TO 625

```

```

540 620 CONTINUE
541 KODF=ITYP(I)
542 DO 540 J=1,12
543 640 STYP(J)=TYPCOD(KODF,J)
544 IF(KODF.GT.4) GO TO 641
545 IF(ISTAGE.EQ.1) WRITE(KOUT,621)(SRCE(I,J),J=1,28),
546 & (STYP(J),J=1,12),NPOS(I),AUNIT(I,1),PUCOST(I,1),TCOST(I,1)
547 IF(ISTAGE.EQ.2) WRITE(KOUT,623)(SRCE(I,J),J=1,28),
548 & (STYP(J),J=1,12),NPOS(I),
549 & AUNIT(I,1),(PUCOST(I,J),J=1,2),(TCOST(I,J),J=1,2)
550 IF(ISTAGE.EQ.3) WRITE(KOUT,623)(SRCE(I,J),J=1,28),
551 & (STYP(J),J=1,12),NPOS(I),
552 & AUNIT(I,1),(PUCOST(I,J),J=1,3),(TCOST(I,J),J=1,3)
553 GO TO 625
554 641 IF(ISTAGE.EQ.1) WRITE(KOUT,642)(STYP(J),J=1,12),NPOS(I),
555 & AUNIT(I,1),PUCOST(I,1),TCOST(I,1)
556 IF(ISTAGE.EQ.2) WRITE(KOUT,643)(STYP(J),J=1,12),NPOS(I),
557 & AUNIT(I,1),(PUCOST(I,J),J=1,2),(TCOST(I,J),J=1,2)
558 IF(ISTAGE.EQ.3) WRITE(KOUT,643)(STYP(J),J=1,12),NPOS(I),
559 & AUNIT(I,1),(PUCOST(I,J),J=1,3),(TCOST(I,J),J=1,3)
560 625 DO 628 J=1,ISTAGE
561 628 STGTOT(J)=STGTOT(J)+TCOST(I,J)
562 632 CONTINUE
563 C
564 IF(ISTAGE.EQ.1) WRITE(KOUT,635)STGTOT(I)
565 IF(ISTAGE.EQ.2) WRITE(KOUT,637)(STGTOT(J),J=1,2)
566 IF(ISTAGE.EQ.3) WRITE(KOUT,638)(STGTOT(J),J=1,3)
567 616 FORMAT(1X,28A1,3X,'CROPLAND',17,F11.1,9X,F17.1,F20.1)
568 619 FORMAT(1X,28A1,3X,'CROPLAND',17,F11.1,9X,6F11.1)
569 621 FORMAT(1X,28A1,1X,12A1,15,11X,F10.0,F16.1,F20.1)
570 623 FORMAT(1X,28A1,1X,12A1,15,11X,F10.0,F10.1,5F11.1)
571 635 FORMAT('0',20X,'TOTAL',58X,F20.1)
572 637 FORMAT('0',20X,'TOTAL',63X,2F11.1)
573 638 FORMAT('0',20X,'TOTAL',63X,3F11.1)
574 642 FORMAT(30X,12A1,15,F11.1,10X,F16.1,F20.1)
575 643 FORMAT(30X,12A1,15,F11.1,9X,6F11.1)
576 IF(ISTAGE.EQ.3) WRITE(KOUT,565)
577 IF(ISTAGE.EQ.3) WRITE(KOUT,566)
578 GO TO 16
579 C
580 C OUTPUT ROUTINE FOR WORKSHEET 7
581 C
582 700 IF(.NOT.URBCTL .AND. .NOT.RURCTL) GO TO 16
583 IF(ISTAGE.EQ.0 .OR. NSRCE.EQ.0) GO TO 16
584 IF(.NOT.IRANK) CALL WTRANK(ISTAGE)
585 FINTOT=0.0
586 DO 775 J=1,ISTAGE
587 WRITE(KOUT,710) IFF,J,J,J,J
588 710 FORMAT(A1,44X,'COST-EFFECTIVENESS ANALYSIS',48X,'TOTAL POLLUTANT',
589 & 'LOAD',61X,60X,'WORKSHEET 7',1X,65(' '),66(' '),58X,58X,'COST',
590 & 'LOAD',15X,'PER UNIT',66X,'TOTAL LOAD TO',4X,
591 & 'LOAD AT',3X,'REDUCTION COST OF REMOVED',5X,'COST-',
592 & 'EFFECTIVE',4X,'SURFACE WATER',5X,'MOUTH',4X,
593 & 'AT MOUTH',3X,'PROGRAM',3X,'AT MOUTH EFFECTIVE',6X,'SOURCE',
594 & 'TYPE POSITION TRANSMISSION',5X,'(KG/YR)',7X,'(KG/YR)',
595 & '4X,'(KG/YR)',3X,'(1/YR)',5X,'(1/KG)',5X,'RANK',1X,65(' '),66(' '),
596 & '6/5X,'STAGE',12,79X,3(3X,'STAGE',12)/5X,7(' '),79X,3(3X,7(' ')))
597 C
598 DO 718 I=1,NSRCE
599 KODF=ITYP(I)

```

```

600      DO 713 NAM=1,12
601 713 STYP(NAM)=TYP(1,NAM)
602      IF (ITYP(1).EQ.1 .OR. AUNIT(1,2).EQ.0.0) GO TO 716
603      SAMNAM=SCISRC(1,SRCE,KODE)
604      IF (SAMNAM) GO TO 715
605      WRITE(KOUT,714) (SRCE(1,NAM),NAM=1,28), (STYP(NAM),NAM=1,12),
606      &NPOS(1),TRNSMN(1),AUNIT(1,2),PMOUTH(1),DIFMTH(1,J),TCOST(1,J),
607      &CPUREM(1,J),KERANK(1,J)
608      GO TO 718
609 715 WRITE(KOUT,719) (STYP(NAM),NAM=1,12),NPOS(1),TRNSMN(1),AUNIT(1,2),
610      &PMOUTH(1),DIFMTH(1,J),TCOST(1,J),CPUREM(1,J),KERANK(1,J)
611 719 FORMAT(30X,12A1,15,F12.1,5X,F12.0,F14.0,1X,2F10.0,F10.1,18)
612      GO TO 718
613 716 WRITE(KOUT,717) (SRCE(1,NAM),NAM=1,28), (STYP(NAM),NAM=1,12),NPOS(1),
614      &TRNSMN(1),AUNIT(1,2),PMOUTH(1)
615 718 CONTINUE
616 714 FORMAT(1X,28A1,1X,12A1,15,F12.1,5X,F12.0,F14.0,1X,2F10.0,F10.1,18)
617 717 FORMAT(1X,28A1,1X,12A1,15,F12.1,5X,F12.0,F14.0)
618      FINTOT=FINTOT+TOTREM(J)
619      IF (J.EQ.1STAGE) WRITE(KOUT,720) TOTREM(J),TOTLOD,TOTMTH,FINTOT,
620      & TOTCST(J)
621      IF (J.LT.1STAGE) WRITE(KOUT,722) TOTREM(J),TOTLOD,TOTMTH,TOTCST(J)
622 720 FORMAT(1H0,16X,'SUBTOTAL',64X,F12.0/20X,'TOTAL',39X,F12.0,F14.0,
623      & 1X,2F10.0)
624 722 FORMAT(1H0,16X,'SUBTOTAL',64X,F12.0/20X,'TOTAL',39X,F12.0,F14.0,
625      & 11X,F10.0)
626      WRITE(KOUT,725)
627 725 FORMAT(1X,65(' '),66(' '))
628 775 CONTINUE
629      GO TO 16
630
631 C
632 C      OUTPUT ROUTINE FOR WORKSHEET 8
633 C
634 800 IF (.NOT.URBCTL .AND. .NOT.RURCTL) GO TO 16
635      IF (IS.EQ.0 .OR. NSRCE.EQ.0) GO TO 16
636      IF (.NOT.LRANK) CALL WTRANK(1STAGE)
637      SUM1=0.0
638      SUM2=0.0
639      WRITE(KOUT,815) IF
640 815 FORMAT(A1,48X,'SUMMARY OF PROGRAMS'/47X,'TOTAL POLLUTANT LOADS'/
641      & 55X,56X,'WORKSHEET 8'/1X,60(' '),61(' ')/57X,57X,'SUM OF'/65X,
642      & 'LOAD',6X,'SUM OF LOAD',15X,'COST OF',5X,'REDUCTION'/63X,
643      & 'REDUCTION',4X,'REDUCTION',4X,'PERCENT',4X,'REDUCTION',6X,'COSTS'/
644      & 67X,'SOURCE',34X,'RANK',4X,'STAGE',5X,'(KG/YR)',6X,'(KG/YR)',4X,
645      & 'REDUCTION',4X,'($/YR)',7X,'($/YR)/1X,60(' '),61(' '))
646 C
647      DO 875 I=1,NUM
648      M2=KNDX(I)
649      SUM1=SUM1+FINREM(M2)
650      SUM2=SUM2+FINCST(M2)
651      PCTRED=(SUM1/TOTLOD)*100.0
652      WRITE(KOUT,820) (FINSRC(M2,J),J=1,43),I,ISFIN(M2),FINREM(M2),SUM1,
653      & PCTRED,FINCST(M2),SUM2
654 875 CONTINUE
655      WRITE(KOUT,850)
656 850 FORMAT(1X,60(' '),61(' '))
657 820 FORMAT(1X,43A1,15,F14.0,F13.0,F11.1,2F13.0)
658      GO TO 16
659 C
660 899 IFR=0

```

```

660      900 CONTINUE
661      RETURN
662      END
663      C
664      C
665      LOGICAL FUNCTION SCHSRC(I,SRCE,KODE)
666      C
667      C      LOGICAL FUNCTION TO SEE IF CURRENT SOURCE NAME IS IDENTICAL TO
668      C      THE PREVIOUS SOURCE NAME FOR WORKSHEET OUTPUT
669      C
670      INTEGER SRCE(210,28),CURSRC(28),TSTSRC(28)
671      SCHSRC=.FALSE.
672      IF(I.EQ.1) RETURN
673      IF(KODE.GT.4) GO TO 25
674      J=I-1
675      DO 10 K=1,28
676          TSTSRC(K)=SRCE(J,K)
677          CURSRC(K)=SRCE(I,K)
678      10 CONTINUE
679      LENTST=LENGTH(TSTSRC,28)
680      LENCUR=LENGTH(CURSRC,28)
681      IF(LENTST.NE.LENCUR) RETURN
682      DO 20 K=1,LENTST
683          IF(TSTSRC(K).NE.CURSRC(K)) GO TO 30
684      20 CONTINUE
685      25 SCHSRC=.TRUE.
686      30 RETURN
687      END
END OF FILE

```

The third source file is called

WTRSAV

```

1  SUBROUTINE WTRSAV(ISWTCN,ISTAGF,KEYWPD)
2  C
3  C   ROUTINE TO WRITE TO AND READ FROM PERMANENT STORAGE
4  C   DATA FOR WTRSHD   ISWTCN=1 FOR READ COMMAND
5  C   JIN  - INPUT DEVICE INITIALIZED IN WTRINI AT 8
6  C   JOU  - OUTPUT DEVICE INITIALIZED IN WTRINI AT 9
7  C
8  C *****
9  LOGICAL LINPUT,LODURB,LODRNC,LODRCP,L4UAL,LIN1,LIN2,LIN3,LIN4
10 LOGICAL URBCST,RULCTL,UCTLOD(3),RCTLOD(3),UPRCNT,JXPCNT,RPRCNT,
11 C   RXPCNT,URBCST,RULCST,LKANK
12 INTEGER OTHURB,OTHRUL,SRCE(210,28),TYPCOD(9,12),OPTVAL(8)
13 DIMENSION ITYP(210),NPOS(210),AUNIT(210,2),KTITLE(4,40),
14 C   SURFEA(210,3),CTLOD(210,3),CTUSLF(3,30,5)
15 DIMENSION STMCTL(30,3),STMARA(30,3),COMCTL(30,3),COMARA(30,3),
16 C   OT1CTL(30,3),OT1ARA(30,3),OT2CTL(30,3),OT2ARA(30,3)
17 DIMENSION PFLOW(30),PCONC(30),UALS(30),UALC(30),W2UAL1(30),
18 C   W2UAL2(30),TRANSMN(210)
19 INTEGER TLEUR1(12),TLEUR2(12),TLERU1(12),TLERU2(12),TLERU3(12),
20 C   APOE(210),AREACN(4),UALCON(4)
21 DIMENSION GRASA(30),UALG(30),WLND(30),UALW(30),W3OTA1(30),
22 C   W3UAL1(30),W3UAL2(30),W3UAL3(30),W3OTA2(30),W3OTA3(30)
23 REAL LODRN1(30),LODRN2(30),LOD1W3(30),LOD2W3(30),LOD3W3(30)
24 DIMENSION USLER(30),USLEK(30),USLELS(30),USLEFC(30),W4UAL(30),
25 C   USLEP(30),SOILS1(30),SOILS2(30),TTLER(30),RATIO(30)
26 DIMENSION PDIF(210,3),TCOST(210,3),PUCOST(210,3)
27 DIMENSION PMOUTH(210),DIFMTH(210,3),CPUREM(210,3),KERANK(210,3)
28 COMMON /WTRIO/  IN,IOUT,NSRCE,KTITLE,NTLE,NURB,NRUNCN,NRUCRP
29 COMMON /WTRIF/  LINPUT,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHURB,
30 C   OTHRUL,LODRCP,TYPCOD,JIN,JOU,L4UAL,KOUT,LKANK
31 COMMON /WRKSH1/ SRCE,ITYP,NPOS,APOE,AUNIT,SURFEA,TLEUR1,TLEUR2
32 COMMON /WRKSH2/ PFLOW,PCONC,UALS,UALC,W2UAL1,W2UAL2
33 COMMON /WRKSH3/ GRASA,UALG,WLND,UALW,LODRN1,LODRN2,
34 C   W3OTA1,W3UAL1,W3UAL2,W3UAL3,W3OTA2,W3OTA3,
35 C   TLERU1,TLERU2,TLERU3,LOD1W3,LOD2W3,LOD3W3
36 COMMON /WRKSH4/ USLER,USLEK,USLELS,USLEFC,USLEP,SOILS1,
37 C   SOILS2,TTLER,W4UAL,RATIO
38 COMMON /WRKSH6/ PDIF,TCOST,PUCOST,TRANSMN,PMOUTH,DIFMTH,CPUREM,
39 C   KERANK
40 COMMON /WTRCON/ UCTLOD,RCTLOD,URBCST,RULCTL,UPRCNT,JXPCNT,RPRCNT,
41 C   RXPCNT,OPTVAL,NJPT,URBCST,RULCST,AREACN,UALCON
42 DIMENSION CTLFLW(30,3),CTLCNC(30,3)
43 DIMENSION CTLEF(30,3),SOILC1(30,3),SOILC2(30,3),CTLER0(30,3),
44 C   CUALW4(30,3)
45 DIMENSION TOTCST(3),TOTREM(3),KNOX(540),FINCST(540),FINRFM(540),
46 C   ISFIN(540)
47 INTEGER FINSRC(540,43)
48 COMMON /WRKSH8/ TOTCST,TOTREM,TOTLOD,TOTMTH,KNOX,FINSRC,FINCST,
49 C   FINRFM,ISFIN,NUM
50 COMMON /UBNCTL/ CTLFLW,CTLCNC,CTLOD,STMCTL,STMARA,COMCTL,
51 C   COMARA,OT1CTL,OT1ARA,OT2CTL,OT2ARA
52 COMMON /RFDCTL/ CTUSLF,CTLE,SOILC1,SOILC2,CTLER0,CUALW4
53 C *****
54 DATA ALPHA1,ALPHA2/'####','$$$$'/
55 IF(ISWTCN.EQ.1) GO TO 100
56 IF(.NOT.LINPUT) GO TO 700
57 C
58 C   SAVE COMMAND
59 C

```

```

60      IP1=0
61      IP2=0
62      IP3=0
63      IP4=0
64      IF (KFYWRD .GT. 0) GO TO 200
65      IF (LODURB) IP1=1
66      IF (LODRNC) IP2=1
67      IF (LODRCP) IP3=1
68      IF (LQUAL) IP4=1
69      WRITE(JOUT,1) NSRCE,OTHRB,OTHRUL,NTLF,IP1,IP2,IP3,IP4
70      1 FORMAT(8I5)
71      IF (NTLF.EQ.0) GO TO 7
72      DO 5 I=1,NTLF
73          WRITE(JOUT,2)(KTITLE(I,J),J=1,40)
74      2 FORMAT(40A1)
75      5 CONTINUE
76      7 IF (OTHRB.GE.1) WRITE(JOUT,8)(TLER1(J),J=1,12)
77      IF (OTHRB.EQ.2) WRITE(JOUT,8)(TLER2(J),J=1,12)
78      IF (OTHRUL.GE.1) WRITE(JOUT,8)(TLERU1(J),J=1,12)
79      IF (OTHRUL.GE.2) WRITE(JOUT,8)(TLERU2(J),J=1,12)
80      IF (OTHRUL.EQ.3) WRITE(JOUT,8)(TLERU3(J),J=1,12)
81      8 FORMAT(12A1)
82      IP1=0
83      IP2=0
84      IP3=0
85      IP4=0
86      IP5=0
87      IP6=0
88      IP7=0
89      C
90      DO 80 I=1,NSRCE
91          IF (ITYP(I).EQ.2) AUNIT(I,2)=AUNIT(I,2)/1000.
92          WRITE(JOUT,10)(SRCE(I,J),J=1,28),ITYP(I),APOE(I),(AUNIT(I,J),
93      6 J=1,2),(SURFFA(I,J),J=1,3)
94      10 FORMAT(28A1,13,A1,2F15.4,3A4)
95          IF (ITYP(I).EQ.1) GO TO 60
96          IF (ITYP(I).EQ.2) GO TO 40
97      C      HERE FOR URBAN POINT SOURCES
98          IF (LODURB) GO TO 80
99          IF (ITYP(I).GT.5) GO TO 20
100         IP3=IP3+1
101         WRITE(JOUT,15) PFLOW(IP3),PCONC(IP3)
102      15 FORMAT(2F15.4)
103         GO TO 80
104      C      HERE FOR URBAN NON-POINT SOURCES
105      20 KODE=ITYP(I)-5
106         GO TO(21,22,23,24),KODE
107      21 IP4=IP4+1
108         WRITE(JOUT,15) UALS(IP4)
109         GO TO 80
110      22 IP5=IP5+1
111         WRITE(JOUT,15) UALC(IP5)
112         GO TO 80
113      23 IP6=IP6+1
114         WRITE(JOUT,15) W2UAL1(IP6)
115         GO TO 80
116      24 IP7=IP7+1
117         WRITE(JOUT,15) W2UAL2(IP7)
118         GO TO 80
119      C      HERE FOR RURAL CROPLAND SOURCES

```

```

120      40 IP2=IP2+1
121      AUNIT(1,2)=AUNIT(1,2)*1000.
122      IF(LODRCP) GO TO 50
123      WRITE(JOUT,42)USLER(IP2),USLEK(IP2),USLELS(IP2),USLEC(IP2),
124      & USLEP(IP2),RATIO(IP2)
125      42 FORMAT(5F15.4,F15.8)
126      GO TO 90
127  C
128      50 IF(.NOT.L4UAL) GO TO 80
129      WRITE(JOUT,15) W4UAL(IP2)
130      GO TO 80
131  C      HERE FOR RURAL NON-CROPLAND SOURCES
132      60 IP1=IP1+1
133      IF(LODRNC) GO TO 70
134      WRITE(JOUT,62)GRASA(IP1),UALG(IP1),WLNDG(IP1),UALW(IP1),
135      & W3OTA1(IP1),W3UAL1(IP1),W3OTA2(IP1),W3UAL2(IP1),W3OTA3(IP1),
136      & W3UAL3(IP1)
137      62 FORMAT(13F15.4)
138      GO TO 80
139  C
140      70 WRITE(JOUT,62)LODRN1(IP1),LODRN2(IP1),LOD1W3(IP1),LOD2W3(IP1),
141      & LOD3W3(IP1)
142  C
143      80 CONTINUE
144  C
145      GO TO 800
146  C
147  C      TO SAVE CONTROLS
148  C
149      200 IF(ISTAGE.EQ.0) GO TO 700
150      WRITE(JOUT,202) ALPHA1
151      202 FORMAT(A4)
152      IF(URACTL) IP1=1
153      IF(RULCTL) IP2=1
154      IF(LRANK) IP3=1
155      WRITE(JOUT,1) IP1,IP2,IP3,ISTAGE
156  C
157      DO 250 J=1,ISTAGE
158      IP1=0
159      IP2=0
160      DO 240 I=1,NSRCE
161      IF(I1YP(I).EQ.1) GO TO 240
162      WRITE(JOUT,15) CTLOC(I,J)
163      IF(I1YP(I).EQ.2) GO TO 225
164  C      HERE FOR URBAN & POINT SOURCES
165      IP1=IP1+1
166      WRITE(JOUT,62) CTLEW(IP1,J),CTLCNC(IP1,J),STMARA(IP1,J),
167      & STMCCTL(IP1,J),COMARA(IP1,J),COMCTL(IP1,J),OTLARA(IP1,J),
168      & OTLCCTL(IP1,J),OT2ARA(IP1,J),OT2CTL(IP1,J)
169      GO TO 240
170  C      HERE FOR RURAL CROPLAND SOURCES
171      225 IP2=IP2+1
172      WRITE(JOUT,62)(CTUSIE(J,IP2,IP3),IP3=1,5),SOILC1(IP2,J),
173      & SOILC2(IP2,J),CTLEFR(IP2,J),CTLE(IP2,J),CUALW4(IP2,J)
174  C
175      240 CONTINUE
176      250 CONTINUE
177  C      TO SAVE RANKINGS
178      WRITE(JOUT,202) ALPHA2
179      WRITE(JOUT,252) NUM,TOTLOC,TOTMTH

```

```

180      252 FORMAT(15,2F15.4)
181      WRITE(JOUT,62)(TOTREM(J),J=1,ISTAGE),(TOTCST(J),J=1,ISTAGE)
182      DO 260 I=1,NSRCE
183          WRITE(JOUT,15) TRNSMN(I),PMOUTH(I)
184          DO 255 J=1,ISTAGE
185              WRITE(JOUT,262) POIF(I,J),PUCOST(I,J),TCOST(I,J),DIFMTH(I,J),
186              & CPJREM(I,J),KERANK(I,J)
187      262 FORMAT(5F15.4,15)
188      255 CONTINUE
189      260 CONTINUE
190  C
191      DO 275 I=1,NUM
192          WRITE(JOUT,270)(FINSRC(I,J),J=1,43),KNDX(I),ISFIN(I),FINREM(I),
193          & FINCST(I)
194      270 FORMAT(43A1,2I5,2F15.4)
195      275 CONTINUE
196  C
197      GO TO 800
198  C
199  C      READ COMMAND
200  C
201      100 IF(KEYWPD.GT.0) GO TO 300
202      READ(JIN,1) NSRCE,OTHURB,OTHRUL,NTLE,IP1,IP2,IP3,IP4
203      IF(IP1.EQ.1) LODURB=.TRUE.
204      IF(IP2.EQ.1) LODRNC=.TRUE.
205      IF(IP3.EQ.1) LODRCP=.TRUE.
206      IF(IP4.EQ.1) L4UAL=.TRUE.
207      IF(NTLE.EQ.0) GO TO 107
208      DO 105 I=1,NTLE
209          READ(JIN,2)(KTITLE(I,J),J=1,40)
210      105 CONTINUE
211      107 IF(OTHURB.GE.1) READ(JIN,8)(TLEUR1(J),J=1,12)
212          IF(OTHURB.EQ.2) READ(JIN,8)(TLEUR2(J),J=1,12)
213          IF(OTHRUL.GE.1) READ(JIN,8)(TLERU1(J),J=1,12)
214          IF(OTHRUL.GE.2) READ(JIN,8)(TLERU2(J),J=1,12)
215          IF(OTHRUL.EQ.3) READ(JIN,8)(TLERU3(J),J=1,12)
216          IP1=0
217          IP2=0
218          IP3=0
219          IP4=0
220          IP5=0
221          IP6=0
222          IP7=0
223          NURB=0
224          NRUCRP=0
225          NRUNCP=0
226  C
227      DO 180 I=1,NSRCE
228          NPOS(I)=1
229          READ(JIN,10)(SRCE(I,J),J=1,28),ITYP(I),APOF(I),(AUNIT(I,J),J=1,2)
230          & (SURFFA(I,J),J=1,3)
231          IF(ITYP(I).EQ.1) GO TO 160
232          IF(ITYP(I).EQ.2) GO TO 140
233  C      HERE FOR URBAN POINT SOURCES
234          IF(LODURB) GO TO 180
235          IF(ITYP(I).GT.5) GO TO 120
236          IP3=IP3+1
237          NURB=NURB+1
238          READ(JIN,15) PFLOW(IP3),PCONC(IP3)
239          GO TO 180

```

```

240      C      HERE FOR URBAN NON-POINT SOURCES
241      120 READ(JIN,15) UALURB
242      KODE=ITYP(1)-5
243      GO TO(121,122,123,124),KODE
244      121 IP4=IP4+1
245      UALS(IP4)=UALURB
246      GO TO 180
247      122 IP5=IP5+1
248      UALC(IP5)=UALURB
249      GO TO 180
250      123 IP6=IP6+1
251      W2UAL1(IP6)=UALURB
252      GO TO 180
253      124 IP7=IP7+1
254      W2UAL2(IP7)=UALURB
255      GO TO 180
256      C      HERE FOR RURAL CROPLAND SOURCES
257      140 NRUCRP=NRUCRP+1
258      AUNIT(1,2)=AUNIT(1,2)*1000.
259      IP2=IP2+1
260      IF(LDRCP) GO TO 150
261      CRPLND=AUNIT(1,1)*100.0
262      READ(JIN,42)USLER(IP2),USLEK(IP2),USLFLS(IP2),USLEC(IP2),
263      & USLEP(IP2),RATIO(IP2)
264      SOILS1(IP2)=USLER(IP2)*USLEK(IP2)*USLFLS(IP2)*USLEC(IP2)*
265      & USLEP(IP2)
266      SOILS2(IP2)=SOILS1(IP2)*2.243
267      TILER(IP2)=SOILS2(IP2)*CRPLND
268      GO TO 180
269      C
270      150 IF(.NOT.L4UAL) GO TO 180
271      READ(JIN,15) W4UAL(IP2)
272      GO TO 180
273      C      HERE FOR RURAL NON-CROPLAND SOURCES
274      160 MRUNCP=MRUNCP+1
275      IP1=IP1+1
276      IF(LDRNC) GO TO 170
277      READ(JIN,62)GRASA(IP1),UALG(IP1),WLND1(IP1),UALW(IP1),W3OTA1(IP1)
278      & ,W3UAL1(IP1),W3OTA2(IP1),W3UAL2(IP1),W3OTA3(IP1),W3UAL3(IP1)
279      LODRN1(IP1)=GRASA(IP1)*UALG(IP1)
280      LODRN2(IP1)=WLND1(IP1)*UALW(IP1)
281      IF(OTHRUL.EQ.0) GO TO 180
282      LOD1W3(IP1)=W3OTA1(IP1)*W3UAL1(IP1)
283      LOD2W3(IP1)=W3OTA2(IP1)*W3UAL2(IP1)
284      LOD3W3(IP1)=W3OTA3(IP1)*W3UAL3(IP1)
285      GO TO 180
286      C
287      170 READ(JIN,62)LODRN1(IP1),LODRN2(IP1),LOD1W3(IP1),LOD2W3(IP1),
288      & LOD3W3(IP1)
289      C
290      180 CONTINUE
291      C
292      LINPUT=.TRUE.
293      GO TO 800
294      C
295      C      TO READ CONTROLS
296      C
297      300 READ(JIN,202,END=400) ALPTST
298      IF(ALPTST.NE.ALPHA1) GO TO 300
299      C

```

```

300      READ(JIN,1) IP1,IP2,IP3,ISTAGE
301      IF(IP1.EQ.1) URBCTL=.TRUE.
302      IF(IP2.EQ.1) RULCTL=.TRUE.
303      IF(IP3.EQ.1) LRANK=.TRUE.
304      IF(ISTAGE.EQ.0) GO TO 700
305      DO 350 J=1,ISTAGE
306      IP1=0
307      IP2=0
308      DO 340 I=1,NSRCE
309      IF(ITYP(I).EQ.1) GO TO 340
310      READ(JIN,15) CTLOD(I,J)
311      IF(ITYP(I).EQ.2) GO TO 325
312      C      HERE FOR URBAN & POINT SOURCES
313      IP1=IP1+1
314      READ(JIN,62) CTFLW(IP1,J),CTLCNC(IP1,J),STMARA(IP1,J),
315      & STMCTI(IP1,J),COMARA(IP1,J),COMCTL(IP1,J),OTIARA(IP1,J),
316      & OTICTL(IP1,J),OT2ARA(IP1,J),OT2CTL(IP1,J)
317      GO TO 340
318      C      HERE FOR RURAL CROPLAND SOURCES
319      325 IP2=IP2+1
320      READ(JIN,62) CTUSLE(J,IP2,IP3),IP3=1,5),SOILC1(IP2,J),
321      & SOILC2(IP2,J),CTLERD(IP2,J),CTLE(IP2,J),CUALW4(IP2,J)
322      C
323      340 CONTINUE
324      350 CONTINUE
325      C      HERE TO READ RANKINGS
326      READ(JIN,202) ALPTST
327      IF(ALPTST.NE. ALPHA2) GO TO 400
328      READ(JIN,252) NUM,TOTLOD,TOTMTH
329      READ(JIN,62) (TOTREM(J),J=1,ISTAGE), (TOTCST(J),J=1,ISTAGE)
330      C
331      DO 360 I=1,NSRCE
332      READ(JIN,15) TRNSMN(I),PMOUTH(I)
333      DO 355 J=1,ISTAGE
334      READ(JIN,262) POFI(I,J),PUCOST(I,J),TCOST(I,J),DIFMTH(I,J),
335      & CPUREM(I,J),KERANK(I,J)
336      355 CONTINUE
337      360 CONTINUE
338      C
339      DO 375 I=1,NUM
340      READ(JIN,270) (FINSRC(I,J),J=1,43),KNDX(I),ISFIN(I),FINREM(I),
341      & FINGST(I)
342      375 CONTINUE
343      C
344      GO TO 800
345      C
346      400 WRITE(10UT,402)
347      402 FORMAT('OFPROP IN INPUT/OUTPUT OF CONTROL VALUES')
348      C
349      C
350      700 WRITE(10UT,701)
351      701 FORMAT('0 DATA HAS NOT YET BEEN INPUT')
352      C
353      800 CONTINUE
354      RETURN
355      END

```

END OF FILE

The fourth source file is called

WTRCTL

```

1  > SUBROUTINE WTRCTL (IS,KEYWRD,KOUNTN,IER)
2  C
3  C ROUTINE TO ACCEPT OF CONTROL STRATEGIES FOR WATERSHED
4  C IS - STAGE CODE MAXIMUM IS 3 STAGES
5  C KEYWRD - INPUT SWITCH CONTROLLING PROGRAM FLOW TO ACCEPT
6  C CONTROL STRATEGIES FOR URBAN (POINT & NON-POINT) SOURCES
7  C AND RURAL CROPLAND SOURCES
8  C *****
9  LOGICAL LINPUT,LODURB,LODRNC,LODRCP,L4UAL,LIN1,LIN2,LIN3,LIN4
10 LOGICAL URBCTL,RULCTL,UCTLOD(3),RCTLOD(3),UPRCNT,UXPCNT,RPRCNT,
11 C RXPCNT,URBCST,RULCST
12 INTEGER OTHURB,OTHRUL,SRCE(210,28),TYPCOD(9,12),OPTVAL(8)
13 DIMENSION ITYP(210),NPOS(210),AUNIT(210,2),KTITLE(4,40),
14 C SURFEA(210,3),CTLOD(210,3),CTUSLE(3,30,5)
15 DIMENSION STMC TL(30,3),STMARA(30,3),COMCTL(30,3),
16 C OT1CTL(30,3),OT1ARA(30,3),OT2CTL(30,3),OT2ARA(30,3)
17 DIMENSION PFLOW(30),PCONC(30),UALS(30),UALC(30),W2UAL1(30),
18 C W2UAL2(30),TRNSMN(210)
19 INTEGER TLEUR1(12),TLEUR2(12),APOF(210),AREACN(4),UALCON(4)
20 DIMENSION USLER(30),USLEK(30),USLELS(30),USLEFC(30),W4UAL(30),
21 C USLEF(30),SOILS1(30),SOILS2(30),TTLER(30),RATIO(30)
22 DIMENSION PDIF(210,3),TCOST(210,3),PUCOST(210,3)
23 COMMON /WTRIO/ IN,IOUT,NSRCE,KTITLE,NTLE,NURB,NRUNCP,NRUCRP
24 COMMON /WTRTF/ LINPUT,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHURB,
25 C OTHRUL,LODRCP,TYPCOD,JIN,JOUT,L4UAL,KOUT
26 COMMON /WRKSH1/ SRCE,ITYP,NPOS,APOF,AUNIT,SURFEA,TLEUR1,TLEJR2
27 COMMON /WRKSH2/ PFLOW,PCONC,UALS,UALC,W2UAL1,W2UAL2
28 COMMON /WRKSH4/ USLEF,USLEK,USLELS,USLEFC,USLEF,SOILS1,
29 C SOILS2,TTLER,W4UAL,RATIO
30 COMMON /WRKSH6/ PDIF,TCOST,PUCOST,TRNSMN
31 COMMON /WTRCON/ UCTLOD,RCTLOD,URBCTL,RULCTL,UPRCNT,JXPCNT,RPRCNT,
32 C RXPCNT,OPTVAL,NOPT,URBCST,RULCST,AREACN,UALCON
33 DIMENSION CTLFLW(30,3),CTLCNC(30,3),PCT(8),APCT(4),UPCT(4)
34 DIMENSION CTLE(30,3),SOILC1(30,3),SOILC2(30,3),CTLER0(30,3),
35 C CUALW4(30,3)
36 COMMON /UBNCTL/ CTLFLW,CTLCNC,CTLOD,STMC TL,STMARA,COMCTL,
37 C COMARA,OT1CTL,OT1ARA,OT2CTL,OT2ARA
38 COMMON /REFCTL/ CTUSLE,CTLE,SOILC1,SOILC2,CTLER0,CUALW4
39 C *****
40 INTEGER OPTURB(5),OPTRUL(5),OPTOUT(8)
41 DATA OPTURB/'FLOW','CONC','AREA','UAL ','LOAD'/
42 DATA OPTRUL/'R ','K ','LS','C ','P '/
43 DATA IBLNK/' '//
44 DO 1 I=1,8
45 PCT(I)=0.0
46 1 OPTOUT(I)=IBLNK
47 C HEADING OUTPUT OF STAGE OF CONTROL
48 IER=2
49 IF (IS.GT.3) RETURN
50 IF (KEYWRD.EQ.1) WRITE (IOUT,2) IS
51 2 FORMAT(1H0,5X,'STAGE',12,' CONTROL STRATEGIES FOR URBAN SOURCES')
52 IF (KEYWRD.EQ.2) WRITE (IOUT,3) IS
53 3 FORMAT(1H0,5X,'STAGE',12,' CONTROL STRATEGIES FOR RURAL SOURCES')
54 C INITIALIZE
55 DO 4 I=1,4
56 APCT(I)=0.0
57 4 UPCT(I)=0.0
58 ISL=IS-1
59 CALL CTLINE (IS,KEYWRD,IER)

```

```

60      IF(IER.NE.0) RETURN
61      IF(KEYWRD.EQ.2) GO TO 199
62      C
63      C      URBAN CONTROL STRATEGIES
64      C
65      IER=3
66      NOUT=NOPT
67      IF(NOPT.EQ.0) NOUT=4
68      IF(LODURB .AND. NOPT.EQ.0) NOPT=1
69      IF(LODURB .AND. NOPT.EQ.0) OPTVAL(1)=5
70      DO 5 I=1,NOUT
71          IF(NOPT.EQ.0) OPTVAL(I)=I
72          K=OPTVAL(I)
73          OPTOUT(I)=OPTURB(K)
74      5 CONTINUE
75      IF(.NOT.UPRCNT) GO TO 20
76      IF(UXPCNT) GO TO 20
77      C      HERE TO ENTER PERCENT REDUCTIONS FOR SELECTED SOURCES
78      I=0
79      8 I=I+1
80      IF(I .GT. NOUT) GO TO 20
81      K=OPTVAL(I)
82      IF(K.EQ.3) GO TO 11
83      IF(K.EQ.4) GO TO 15
84      WRITE(IOUT,10) OPTOUT(I)
85      READ(IN,46) PCT(K)
86      GO TO 8
87      C      PROMPT FOR AREA % REDUCTIONS
88      11 IF(AREACN(1).EQ.0) GO TO 12
89      WRITE(IOUT,111)OPTURB(3)
90      READ(IN,46) APCT(1)
91      I=I+1
92      IF(I .GT. NOUT) GO TO 20
93      12 IF(AREACN(2).EQ.0) GO TO 13
94      WRITE(IOUT,112)OPTURB(3)
95      READ(IN,46) APCT(2)
96      I=I+1
97      IF(I .GT. NOUT) GO TO 20
98      13 IF(AREACN(3).EQ.0) GO TO 14
99      WRITE(IOUT,113)(TLEUR1(JNM),JNM=1,12),OPTURB(3)
100      READ(IN,46) APCT(3)
101      I=I+1
102      IF(I .GT. NOUT) GO TO 20
103      14 IF(AREACN(4).EQ.0) GO TO 8
104      WRITE(IOUT,113)(TLEUR2(JNM),JNM=1,12),OPTURB(3)
105      READ(IN,46) APCT(4)
106      I=I+1
107      IF(I .GT. NOUT) GO TO 20
108      111 FORMAT('ENTER PERCENT REDUCTIONS FOR NONPOINT STORM SEWER ',A4/
109      & ' IN DECIMAL FORM (E.G. ENTER 25% AS .25)')
110      112 FORMAT('ENTER PERCENT REDUCTIONS FOR NONPOINT COMBINED ',A4/
111      & ' IN DECIMAL FORM (E.G. ENTER 25% AS .25)')
112      113 FORMAT('ENTER PERCENT REDUCTIONS FOR ',12A1,1X,A4/
113      & ' IN DECIMAL FORM (E.G. ENTER 25% AS .25)')
114      GO TO 8
115      C      PROMPT FOR UAL % REDUCTIONS
116      15 IF(UALCON(1).EQ.0) GO TO 16
117      WRITE(IOUT,111)OPTURB(4)
118      READ(IN,46) UPCT(1)
119      I=I+1

```

```

120     IF(I.GT. NOU1) GO TO 20
121 16 IF(UALCON(2).EQ.0) GO TO 17
122     WRITE(IOUT,112)OPTUPB(4)
123     READ(IN,46) UPCT(2)
124     I=I+1
125     IF(I.GT. NOU1) GO TO 20
126 17 IF(UALCON(3).EQ.0) GO TO 18
127     WRITE(IOUT,113)(TLEUR1(JNM),JNM=1,12),OPTJRB(4)
128     READ(IN,46) UPCT(3)
129     I=I+1
130     IF(I.GT. NOU1) GO TO 20
131 18 IF(UALCON(4).EQ.0) GO TO 8
132     WRITE(IOUT,113)(TLEUR2(JNM),JNM=1,12),OPTURB(4)
133     READ(IN,46) UPCT(4)
134     GO TO 8
135 19 FORMAT('ENTER PERCENT REDUCTIONS FOR SOURCE ',A4/
136     & ' IN DECIMAL FORM (E.G. ENTER 25% AS .25)')
137 C     INITIALIZE
138 20 IP3=0
139     IP4=0
140     IP5=0
141     IP6=0
142     IP7=0
143     IOPT1=0
144     IOPT2=0
145     IOPT3=0
146     IOPT4=0
147     IOPT5=0
148     DO 22 I=1,NOPT
149         IF(OPTVAL(I).EQ.1) IOPT1=1
150         IF(OPTVAL(I).EQ.2) IOPT2=1
151         IF(OPTVAL(I).EQ.3) IOPT3=1
152         IF(OPTVAL(I).EQ.4) IOPT4=1
153         IF(OPTVAL(I).EQ.5) IOPT5=1
154 22 CONTINUE
155     IF(IOPT5.EQ.1 .AND. NOPT.GT.1) RETURN
156     IF(.NOT.UXPCT .AND. UPRCNT) GO TO 40
157 C     HERE FOR INPUT PROMPT OF SELECTED CONTROLS
158     IF(NOPT.GT.1) GO TO 25
159     WRITE(IOUT,23) OPTOUT(1)
160 23 FORMAT('ENTER CONTROL VALUES FOR ',A4,' PER SOURCE')
161     GO TO 40
162 25 IF(NOPT.GT.2) GO TO 30
163     IF((IOPT1.EQ.1.AND.IOPT2.EQ.1).OR.(IOPT3.EQ.1.AND.IOPT4.EQ.1))
164     & GO TO 27
165     WRITE(IOUT,26)(OPTOUT(I),I=1,2)
166 26 FORMAT('ENTER CONTROL VALUES FOR ',A4,' OR ',A4,' PER SOURCE')
167     GO TO 40
168 27 WRITE(IOUT,28)(OPTOUT(I),I=1,2)
169 28 FORMAT('ENTER CONTROL VALUES FOR ',A4,',',A4,' PER SOURCE')
170     GO TO 40
171 30 IF(NOPT.GT.3) GO TO 35
172     IF(IP3.EQ.1.AND.IOPT4.EQ.1) GO TO 33
173     WRITE(IOUT,32)(OPTOUT(I),I=1,3)
174 32 FORMAT('ENTER CONTROL VALUES FOR ',A4,',',A4,' OR ',A4,
175     & ' PER SOURCE')
176     GO TO 40
177 33 WRITE(IOUT,34)(OPTOUT(I),I=1,3)
178 34 FORMAT('ENTER CONTROL VALUES FOR ',A4,' OR ',A4,',',A4,
179     & ' PER SOURCE')

```

```

180      GO TO 40
181      35 WRITE(IOUT,37)(OPTOUT(I),I=1,4)
182      37 FORMAT('CENTER CONTROL VALUES FOR ',A4,', ',A4,', ' OR ',A4,', ',A4,
183      & ' PER SOURCE')
184      C      ACCEPT VALUES FOR CONTROLS
185      40 DO 95 I=1,NSRCE
186          IF(ITYP(I).LE.2) GO TO 95
187          IF(ITYP(I).LE.5.AND.IOPT1.EQ.0.AND.IOPT2.EQ.0.AND.IOPT5.EQ.0)
188              &      GO TO 95
189          IF(ITYP(I).GT.5.AND.IOPT3.EQ.0.AND.IOPT4.EQ.0.AND.IOPT5.EQ.0)
190              &      GO TO 95
191          IF(.NOT.UXPCNT .AND. UPRCNT) GO TO 44
192          IF(IOPT1.EQ.1.OR.IOPT2.EQ.1) GO TO 41
193          I6=I-1
194          I7=I-2
195          I8=I-3
196          I9=I-4
197          IF(ITYP(I).EQ.6) WRITE(IOUT,42)(SRCE(I6,J),J=1,28)
198          IF(ITYP(I).EQ.7) WRITE(IOUT,42)(SRCE(I7,J),J=1,28)
199          IF(ITYP(I).EQ.8) WRITE(IOUT,42)(SRCE(I8,J),J=1,28)
200          IF(ITYP(I).EQ.9) WRITE(IOUT,42)(SRCE(I9,J),J=1,28)
201      41 WRITE(IOUT,42)(SRCE(I,J),J=1,28)
202      42 FORMAT(1X,28A1)
203      44 IF(ITYP(I).GT.5) GO TO 65
204      C      HERE FOR CONTROLS FOR URBAN POINT SOURCES
205          IP3=IP3+1
206          IF(UXPCNT) GO TO 54
207          IF(IOPT5.EQ.1) GO TO 60
208          IF(UPRCNT) GO TO 56
209          IF(IOPT1.EQ.1 .AND. IOPT2.EQ.1) GO TO 52
210          IF(IOPT1.EQ.1) READ(IN,46) CTLFLW(IP3,IS)
211          IF(IOPT2.EQ.1) READ(IN,46) CTLCNC(IP3,IS)
212          GO TO 58
213      52 READ(IN,46) CTLFLW(IP3,IS),CTLCNC(IP3,IS)
214      46 FORMAT(2F10.0)
215          GO TO 58
216      C      HERE FOR PERCENT REDUCTION CALCULATIONS
217      54 IF(IOPT5.EQ.0) READ(IN,46) PCT(1),PCT(2)
218          IF(IOPT5.EQ.1) READ(IN,46) PCT(5)
219          IF(IOPT5.EQ.1) GO TO 60
220      56 IF(IOPT1.EQ.1) CTLFLW(IP3,IS)=CTLFLW(IP3,IS)-(CTLFLW(IP3,IS)*
221          &      PCT(1))
222          IF(IOPT2.EQ.1) CTLCNC(IP3,IS)=CTLCNC(IP3,IS)-(CTLCNC(IP3,IS)*
223          &      PCT(2))
224      58 CTLOD(I,IS)=CTLFLW(IP3,IS) * CTLCNC(IP3,IS) * 1382.0
225          GO TO 90
226      60 IF(UPRCNT) GO TO 64
227          READ(IN,46) CTLOD(I,IS)
228          GO TO 90
229      64 CTLOD(I,IS)=CTLOD(I,IS)-(CTLOD(I,IS)*PCT(5))
230          GO TO 90
231      C      HERE FOR CONTROLS FOR URBAN NON-POINT SOURCES
232      65 K=ITYP(I)-5
233          IF(APEACN(K).EQ.0 .AND. UALCON(K).EQ.0) GO TO 90
234          IF(UXPCNT) GO TO 76
235          IF(IOPT5.EQ.1) GO TO 85
236          IF(UPRCNT) GO TO 78
237          GO TO(67,69,71,73),K
238      C      HERE FOR NONPOINT CONTROLS FOR STORM SEWER
239      67 IP4=IP4+1

```

```

240      IF (AREACN(K).EQ.1 .AND. UALCON(K).EQ.1) GO TO 68
241      IF (AREACN(K).EQ.1) READ(IN,46) STMARA(IP4,IS)
242      IF (UALCON(K).EQ.1) READ(IN,46) STMCTL(IP4,IS)
243      GO TO 280
244      68 READ(IN,46) STMARA(IP4,IS),STMCTL(IP4,IS)
245      GO TO 280
246      C      HERE FOR NONPOINT CONTROLS FOR COMBINED SEWER
247      69 IP5=IP5+1
248      IF (AREACN(K).EQ.1 .AND. UALCON(K).EQ.1) GO TO 70
249      IF (AREACN(K).EQ.1) READ(IN,46) COMARA(IP5,IS)
250      IF (UALCON(K).EQ.1) READ(IN,46) COMCTL(IP5,IS)
251      GO TO 282
252      70 READ(IN,46) COMARA(IP5,IS),COMCTL(IP5,IS)
253      GO TO 282
254      C      HERE FOR NONPOINT CONTROLS FOR OTHER SOURCE #1
255      71 IP6=IP6+1
256      IF (AREACN(K).EQ.1 .AND. UALCON(K).EQ.1) GO TO 72
257      IF (AREACN(K).EQ.1) READ(IN,46) OTIARA(IP6,IS)
258      IF (UALCON(K).EQ.1) READ(IN,46) OTICTL(IP6,IS)
259      GO TO 284
260      72 READ(IN,46) OTIARA(IP6,IS),OTICTL(IP6,IS)
261      GO TO 284
262      C      HERE FOR NONPOINT CONTROLS FOR OTHER SOURCE #2
263      73 IP7=IP7+1
264      IF (AREACN(K).EQ.1 .AND. UALCON(K).EQ.1) GO TO 74
265      IF (AREACN(K).EQ.1) READ(IN,46) OT2ARA(IP7,IS)
266      IF (UALCON(K).EQ.1) READ(IN,46) OT2CTL(IP7,IS)
267      GO TO 286
268      74 READ(IN,46) OT2ARA(IP7,IS),OT2CTL(IP7,IS)
269      GO TO 286
270      C      HERE FOR % REDUCTIONS WHICH DIFFER BETWEEN SOURCES
271      76 IF (IOPT5.EQ.1) READ(IN,46) PCT(5)
272      IF (IOPT5.EQ.1) GO TO 85
273      IF (AREACN(K).EQ.1 .AND. UALCON(K).EQ.1) GO TO 77
274      IF (AREACN(K).EQ.1) READ(IN,46) APCT(K)
275      IF (UALCON(K).EQ.1) READ(IN,46) UPCT(1)
276      GO TO 78
277      77 READ(IN,46) APCT(K),UPCT(K)
278      C      THE BELOW CONTROLS FOR % REDUCTIONS
279      78 GO TO (279,281,283,285),K
280      C      HERE FOR PERCENT REDUCTIONS IN CONTROLS FOR STORM SEWER
281      279 IP4=IP4+1
282      IF (AREACN(K).EQ.1) STMARA(IP4,IS)=STMARA(IP4,IS) -
283      & (STMARA(IP4,IS)*APCT(1))
284      IF (UALCON(K).EQ.1) STMCTL(IP4,IS)=STMCTL(IP4,IS) -
285      & (STMCTL(IP4,IS)*JPCT(1))
286      280 C LIND(1,IS)=STMARA(IP4,IS)*STMCTL(IP4,IS)
287      GO TO 90
288      C      HERE FOR PERCENT REDUCTIONS IN CONTROLS FOR COMBINED SEWER
289      281 IP5=IP5+1
290      IF (AREACN(K).EQ.1) COMARA(IP5,IS)=COMARA(IP5,IS) -
291      & (COMARA(IP5,IS)*APCT(2))
292      IF (UALCON(K).EQ.1) COMCTL(IP5,IS)=COMCTL(IP5,IS) -
293      & (COMCTL(IP5,IS)*JPCT(2))
294      282 C LIND(1,IS)=COMARA(IP5,IS)*COMCTL(IP5,IS)
295      GO TO 90
296      C      HERE FOR PERCENT REDUCTIONS IN CONTROLS FOR OTHER SOURCE #1
297      283 IP6=IP6+1
298      IF (AREACN(K).EQ.1) OTIARA(IP6,IS)=OTIARA(IP6,IS) -
299      & (OTIARA(IP6,IS)*APCT(3))

```

```

300      IF(UALCON(K).EQ.1) OT1CTL(IP6,IS)=OT1CTL(IP6,IS) -
301      & (OT1CTL(IP6,IS)*JPCT(3))
302      CTLOD(I,IS)=OT1ARA(IP6,IS)*OT1CTL(IP6,IS)
303      GO TO 90
304      C      HERE FOR PERCENT REDUCTIONS IN CONTROLS FOR OTHER SOURCE #2
305      285 IP7=IP7+1
306      IF(AREACN(K).EQ.1) OT2ARA(IP7,IS)=OT2ARA(IP7,IS) -
307      & (OT2ARA(IP7,IS)*APCT(4))
308      IF(UALCON(K).EQ.1) OT2CTL(IP7,IS)=OT2CTL(IP7,IS) -
309      & (OT2CTL(IP7,IS)*JPCT(4))
310      286 CTLOD(I,IS)=OT2ARA(IP7,IS)*OT2CTL(IP7,IS)
311      GO TO 90
312      85 IF(UPRCNT) GO TO 86
313      READ(IN,46) CTLOD(I,IS)
314      GO TO 90
315      96 CTLOD(I,IS)=CTLOD(I,IS)-(CTLOD(I,IS)*PCT(5))
316      C      CALCULATE DIFFERENCE IN LOADS BETWEEN STAGES
317      90 IF(IS.EQ.1) PDIF(I,1)=AUNIT(I,2)-CTLOD(I,1)
318      IF(IS.GT.1) PDIF(I,IS)=CTLOD(I,IS1)-CTLOD(I,IS)
319      95 CONTINUE
320      URBCTL=.TRUE.
321      KOUNTR=KOUNTR+1
322      IER=0
323      GO TO 999
324      C
325      C      RURAL CONTROL STRATEGIES
326      C
327      100 IER=3
328      IF(LQORCP .OR. L4JAL) GO TO 103
329      GO TO 105
330      103 IF(NOPT.GT.1) RETURN
331      IF(NOPT.EQ.0) NOPT=1
332      IF(LQOURB) OPTVAL(1)=8
333      IF(L4JAL) OPTVAL(1)=7
334      105 NOUT=NOPT
335      NUSLE=0
336      IF(NOPT.EQ.0) NOUT=5
337      DO 115 I=1,NOUT
338      IF(NOPT.EQ.0) OPTVAL(I)=I
339      K=OPTVAL(I)
340      IF(K.GT.5) GO TO 114
341      NUSLE=NUSLE+1
342      OPTOUT(I)=OPTROL(K)
343      GO TO 115
344      114 IF(K.EQ.6) OPTOUT(I)=IBLNK
345      IF(K.EQ.7) OPTOUT(I)=OPTURB(4)
346      IF(K.EQ.8) OPTOUT(I)=OPTURB(5)
347      115 CONTINUE
348      C      HERE TO ENTER PERCENT REDUCTIONS FOR SELECTED CONTROLS
349      IF(.NOT.RPRCNT) GO TO 125
350      IF(RXPCNT) GO TO 125
351      DO 120 I=1,NOUT
352      M=OPTVAL(I)
353      IF(M.NE.6) WRITE(1OUT,19) OPTOUT(I)
354      IF(M.EQ.6) WRITE(1OUT,117)
355      READ(IN,46) PCT(M)
356      120 CONTINUE
357      117 FORMAT('ENTER PERCENT REDUCTIONS FOR SOURCE EFFICIENCY RATIO')
358      125 K1=OPTVAL(1)
359      K2=OPTVAL(2)

```

```

360      K3=OPTVAL(3)
361      K4=OPTVAL(4)
362      IF(.NOT.PXPCT .AND. PPRCNT) GO TO 140
363  C      HERE FOR PROMPT FOR INPUT OF SELECTED CONTROLS
364      IT=0
365      IF(NUSLE.GT.0) WRITE(IOUT,127)(OPTOUT(1),I=1,NUSLE)
366  127  FORMAT('ENTER CONTROL VALUES PER SOURCE FOR ',4(A2,', '),A2)
367      IF(NUSLE.GT.0) GO TO 140
368  130  IT=IT+1
369      IF(IT.GT.NOUT) GO TO 200
370      IF(OPTVAL(IT).EQ.6) WRITE(IOUT,132)
371  132  FORMAT('ENTER CONTROL VALUES FOR EFFICIENCY RATIO PER SOURCE')
372      IF(OPTVAL(IT).NE.6) WRITE(IOUT,134) OPTOUT(IT)
373  134  FORMAT('ENTER CONTROL VALUES FOR ',A4,' PER SOURCE')
374  C      BEGIN CONTROLS
375  140  IP2=0
376      DO 195 I=1,NSRCE
377      IF(ITYP(I).NE.2) GO TO 195
378      IP2=IP2+1
379      IF(NUSLE.EQ.0) GO TO 170
380  C      HERE FOR INPUT OF USLE COEFFICIENTS
381      IF(RXPCT .OR. .NOT. RPRCNT) WRITE(IOUT,42)(SRCE(I,J),J=1,20)
382      IF(RXPCT) GO TO 150
383      IF(RPRCNT) GO TO 154
384      IF(NUSLE.EQ.1) READ(IN,155) CTUSLE(IS,IP2,K1)
385      IF(NUSLE.EQ.2) READ(IN,155) CTUSLE(IS,IP2,K1),CTUSLE(IS,IP2,K2)
386      IF(NUSLE.EQ.3) READ(IN,155) CTUSLE(IS,IP2,K1),CTUSLE(IS,IP2,K2),
387      CTUSLE(IS,IP2,K3)
388      IF(NUSLE.EQ.4) READ(IN,155) CTUSLE(IS,IP2,K1),CTUSLE(IS,IP2,K2),
389      CTUSLE(IS,IP2,K3),CTUSLE(IS,IP2,K4)
390      IF(NUSLE.EQ.5) READ(IN,155)(CTUSLE(IS,IP2,K0),K0=1,5)
391  155  FORMAT(5F10.0)
392      GO TO 160
393  C      HERE FOR PERCENT REDUCTION CALCULATIONS
394  150  IF(NUSLE.EQ.1) READ(IN,155) PCT(K1)
395      IF(NUSLE.EQ.2) READ(IN,155) PCT(K1),PCT(K2)
396      IF(NUSLE.EQ.3) READ(IN,155) PCT(K1),PCT(K2),PCT(K3)
397      IF(NUSLE.EQ.4) READ(IN,155) PCT(K1),PCT(K2),PCT(K3),PCT(K4)
398      IF(NUSLE.EQ.5) READ(IN,155)(PCT(K0),K0=1,5)
399  154  CTUSLE(IS,IP2,K1)=CTUSLE(IS,IP2,K1)-(CTUSLE(IS,IP2,K1)*PCT(K1))
400      IF(NUSLE.EQ.1) GO TO 160
401      CTUSLE(IS,IP2,K2)=CTUSLE(IS,IP2,K2)-(CTUSLE(IS,IP2,K2)*PCT(K2))
402      IF(NUSLE.EQ.2) GO TO 160
403      CTUSLE(IS,IP2,K3)=CTUSLE(IS,IP2,K3)-(CTUSLE(IS,IP2,K3)*PCT(K3))
404      IF(NUSLE.EQ.3) GO TO 160
405      CTUSLE(IS,IP2,K4)=CTUSLE(IS,IP2,K4)-(CTUSLE(IS,IP2,K4)*PCT(K4))
406      IF(NUSLE.EQ.4) GO TO 160
407      CTUSLE(IS,IP2,K5)=CTUSLE(IS,IP2,K5)-(CTUSLE(IS,IP2,K5)*PCT(K5))
408  C      CALCULATE SOIL LOSS
409  160  CRPLND=AUNIT(I,1)*100.0
410      SOILC1(IP2,IS)=CTUSLE(IS,IP2,1)
411      DO 165 J=2,5
412      SOILC1(IP2,IS)=SOILC1(IP2,IS)*CTUSLE(IS,IP2,J)
413  165  CONTINUE
414      SOILC2(IP2,IS)=SOILC1(IP2,IS) * 2.243
415      DIFRNC=CTLEND(IP2,IS)
416      CTLEND(IP2,IS)=SOILC2(IP2,IS) * CRPLND
417      DIFRNC=DIFRNC - CTLEND(IP2,IS)
418      CTLOD(I,IS)=CTLOD(I,IS)-(DIFRNC*RATIO(IP2)*CTLE(IP2,IS))
419      GO TO 190

```

```

420      C      HERE FOR INPUT OF EFFICIENCY RATIO
421      170 IF(OPTVAL(IT).NE.6) GO TO 193
422      IF(RPRCNT) GO TO 175
423      READ(IN,155) CTLE(IP2,IS)
424      GO TO 178
425      175 IF(.NOT.RXPCNT) GO TO 176
426      READ(IN,155) PCT(6)
427      176 CTLE(IP2,IS)=CTLE(IP2,IS)-(CTLE(IP2,IS)*PCT(6))
428      178 CRPLND=AUNIT(1,1)*100.0
429      IF(IS.EQ.1) DIFRNC=CTLE(IP2)-CTLE(IP2,IS)
430      IF(IS.NE.1) DIFRNC=CTLE(IP2,IS1)-CTLE(IP2,IS)
431      CTLOD(I,IS)=CTLOD(I,IS)-(DIFRNC*RATIO(IP2)*CTLE(IP2,IS))
432      GO TO 190
433      C      HERE FOR INPUT OF EITHER LOADS OR UAL
434      180 IF(OPTVAL(IT).EQ.8) GO TO 185
435      IF(RPRCNT) GO TO 182
436      READ(IN,155) CUALW4(IP2,IS)
437      GO TO 184
438      182 IF(RXPCNT) READ(IN,155) PCT(7)
439      CUALW4(IP2,IS)=CUALW4(IP2,IS)-(CUALW4(IP2,IS)*PCT(7))
440      184 CTLOD(I,IS)=AUNIT(1,1)*CUALW4(IP2,IS)
441      GO TO 190
442      185 IF(RPRCNT) GO TO 187
443      READ(IN,155) CTLOD(I,IS)
444      GO TO 190
445      187 IF(RXPCNT) READ(IN,155) PCT(8)
446      CTLOD(I,IS)=CTLOD(I,IS)-(CTLOD(I,IS)*PCT(8))
447      C      SCALE CONTROL LOADS & CALCULATE LOAD DIFFERENCES
448      190 CTLOD(I,IS)=CTLOD(I,IS)*1000.
449      IF(IS.EQ.1) PDIF(I,1)=AUNIT(1,2)-CTLOD(I,1)
450      IF(IS.GT.1) PDIF(I,IS)=CTLOD(I,IS1)-CTLOD(I,IS)
451      195 CONTINUE
452      IF(NUSLF.GT.0) IT=NUSLF
453      NUSLF=0
454      GO TO 130
455      C      DONE
456      200 RULCTL=.TRUE.
457      KOUNTR=KOUNTR+1
458      IFR=0
459      999 RETURN
460      END
461      C
462      > SUBROUTINE CTLINI(15,KEYWRD,IEP)
463      C
464      C      ROUTINE TO INITIALIZE CONTROL STRATEGY VALUES AT INITIAL STATE
465      C      OR PREVIOUS STAGE OF CONTROL S
466      C      IFR = 1 IF NO INPUT DATA EXISTS FOR KEYWRD SOURCE
467      C
468      C      *****
469      LOGICAL LINUT,LODRB,LODRNC,LODRCP,L4UAL,LIN1,LIN2,LIN3,LIN4
470      LOGICAL URBCTL,RULCTL,UCTLOD(3),RCTLOD(3),UPRCNT,UXPCNT,RPRCNT,
471      &      RXPCNT,URBCST,RULCST
472      INTEGER DTHURB,DTHRUJ,SRCE(210,28),TYPCOD(9,12),OPTVAL(8)
473      DIMENSION ITP(210),MPOS(210),AUNIT(210,21),KTITLE(4,40),
474      &      SURFA(210,3),CTLOD(210,3),CTUSLF(3,30,5)
475      DIMENSION PFLDW(30),PCONC(30),UALS(30),UALC(30),W2UAL1(30),
476      &      W2UAL2(30),TRNSMN(210)
477      DIMENSION STMCTL(3,3),STMARA(30,3),COMCTL(30,3),COMARA(30,3),
478      &      OT1CTL(30,3),OT1ARA(30,3),OT2CTL(30,3),OT2ARA(30,3)
479      INTEGER TLEUP1(12),TLEUP2(12),APOF(210),APFAC(4),UALCON(4)

```

```

480      DIMENSION USLEFR(30),USLEK(30),USLELS(30),USLEC(30),W4UAL(30),
481      &          USLEP(30),SOILS1(30),SOILS2(30),TTLEFR(30),RATIO(30)
482      DIMENSION PDIF(210,3),TCOST(210,3),PUCOST(210,3)
483      COMMON /WTR10/  IN,IOUT,NSRCE,KTTITLE,NTLE,NURB,NRUNCP,NRUCRP
484      COMMON /WTRTF/  LINPUT,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHURB,
485      &          OTHRUL,LODRCP,TPCOD,JIN,JOUT,L4UAL,KOUT
486      COMMON /WRKSH1/  SRCE,ITYP,NPOS,APOE,AUNIT,SURFFA,TLEUR1,TLEUR2
487      COMMON /WRKSH2/  PFLOW,PCONC,UALS,UALC,W2JAL1,W2UAL2
488      COMMON /WRKSH4/  USLEFR,USLEK,USLELS,USLEFC,USLEP,SOILS1,
489      &          SOILS2,TTLEFR,W4UAL,RATIO
490      COMMON /WRKSH6/  PDIF,TCOST,PUCOST,TRNSMN
491      COMMON /WTRCON/  UCTLOD,RCILOD,URBCIL,RULCTL,UPRCNT,UXPCNT,RPRCNT,
492      &          RXPCNT,OPTVAL,NOPT,URBCST,RULCST,AREACN,UALCON
493      DIMENSION CTLEFLW(30,3),CTLCNC(30,3)
494      DIMENSION CTLE(30,3),SOILC1(30,3),SOILC2(30,3),CTLER0(30,3),
495      &          CUALW4(30,3)
496      COMMON /URBCIL/  CTLEFLW,CTLCNC,CTLOD,STMCTL,STMARA,COMCTL,
497      &          COMARA,OT1CTL,OT1ARA,OT2CTL,OT2ARA
498      COMMON /RFDCTL/  CTUSLE,CTLE,SOILC1,SOILC2,CTLER0,CUALW4
499      C *****
500      IER=1
501      IS1=IS - 1
502      IF(KEYWRD.EQ.2) GO TO 20
503      C          HERE FOR URBAN CONTROLS
504      IF(NURB.EQ.0) RETURN
505      IP3=0
506      IP4=0
507      IP5=0
508      IP6=0
509      IP7=0
510      DO 10 I=1,NSRCE
511      IF(ITYP(I).LE.2) GO TO 10
512      IF(ITYP(I).GT.5) GO TO 5
513      C          HERE FOR POINT SOURCES
514      IP3=IP3+1
515      IF(IS.GT.1) GO TO 3
516      CTLEFLW(IP3,1)=PFLOW(IP3)
517      CTLCNC(IP3,1)=PCONC(IP3)
518      GO TO 9
519      3  CTLEFLW(IP3,IS)=CTLEFLW(IP3,IS1)
520      CTLCNC(IP3,IS)=CTLCNC(IP3,IS1)
521      GO TO 9
522      C          HERE FOR NON-POINT SOURCES
523      5  K=ITYP(I) - 5
524      GO TO(11,14,17,21),K
525      C          HERE FOR STORM SEWER
526      11 IP4=IP4+1
527      IF(IS.GT.1) GO TO 12
528      STMARA(IP4,1)=AUNIT(I,1)
529      STMCTL(IP4,1)=UALS(IP4)
530      GO TO 9
531      12 STMARA(IP4,IS)=STMARA(IP4,IS1)
532      STMCTL(IP4,IS)=STMCTL(IP4,IS1)
533      GO TO 9
534      C          HERE FOR COMBINED SEWER
535      14 IP5=IP5+1
536      IF(IS.GT.1) GO TO 15
537      COMARA(IP5,1)=AUNIT(I,1)
538      COMCTL(IP5,1)=UALC(IP5)
539      GO TO 9

```

```

540      15 COMARA(IP5,IS)=COMARA(IP5,IS1)
541      COMCTL(IP5,IS)=COMCTL(IP5,IS1)
542      GO TO 9
543      C      HERE FOR OTHER SOURCE #1
544      17 IP6=IP6+1
545      IF(IS.GT.1) GO TO 18
546      OTLARA(IP6,1)=AUNIT(1,1)
547      OTICTL(IP6,1)=W2UAL1(IP6)
548      GO TO 9
549      19 OTLARA(IP6,IS)=OTLARA(IP6,IS1)
550      OTICTL(IP6,IS)=OTICTL(IP6,IS1)
551      GO TO 9
552      C      HERE FOR OTHER SOURCE #2
553      21 IP7=IP7+1
554      IF(IS.GT.1) GO TO 22
555      OT2ARA(IP7,1)=AUNIT(1,1)
556      OT2CTL(IP7,1)=W2UAL2(IP7)
557      GO TO 9
558      22 OT2ARA(IP7,IS)=OT2ARA(IP7,IS1)
559      OT2CTL(IP7,IS)=OT2CTL(IP7,IS1)
560      C      CORRECT SCALING
561      9 IF(IS.EQ.1) CTLOD(1,1)=AUNIT(1,2)
562      IF(IS.GT.1) CTLOD(1,IS)=CTLOD(1,IS1)
563      10 CONTINUE
564      GO TO 50
565      C      HERE FOR RURAL CONTROLS
566      20 IF(NRUCRP.EQ.0) RETURN
567      IP2=0
568      DO 30 I=1,NSRCE
569      IF(ITYP(I).NE.2) GO TO 30
570      IP2=IP2+1
571      IF(IS.GT.1) GO TO 23
572      CTUSLF(1,IP2,1)=USLER(IP2)
573      CTUSLE(1,IP2,2)=USLEK(IP2)
574      CTUSLF(1,IP2,3)=USLELS(IP2)
575      CTUSLF(1,IP2,4)=USLEFC(IP2)
576      CTUSLF(1,IP2,5)=USLEP(IP2)
577      CTLERO(IP2,1)=TTLEP(IP2)
578      CTLE(IP2,1)=1.0
579      CUALW4(IP2,1)=W4JAL(IP2)
580      CTLOD(1,1)=AUNIT(1,2)/1000.
581      GO TO 30
582      23 DO 25 J=1,5
583      25 CTUSLE(IS,IP2,J)=CTUSLE(IS1,IP2,J)
584      CTLERO(IP2,IS)=CTLERO(IP2,IS1)
585      CTLE(IP2,IS)=CTLE(IP2,IS1)
586      CUALW4(IP2,IS)=CUALW4(IP2,IS1)
587      C
588      CTLOD(1,IS)=CTLOD(1,IS1)/1000.
589      30 CONTINUE
590      C      DONE
591      50 ICR=0
592      RETURN
593      END
594      C
595      > SUBROUTINE WRCST(IFLAG,KEYWRD,IS,KOUNTR,ICR)
596      C
597      C      ROUTINE TO ACCEPT INPUT OF COST OF CONTROL DATA
598      C      IFLAG = 1 IF COSTS DIFFER BETWEEN SOURCES
599      C      ICR = 'L' IF MORE THAN 3 STAGES OF COST DATA REQUESTED

```

```

600      = 2 IF INPUT DATA DOES NOT EXIST
601      KEYWRD = 1 FOR URBAN COSTS
602      C      = 2 FOR RURAL COSTS
603      C *****
604      LOGICAL LINPUT, LUDURB, LUDRNC, LUDRCP, L4UAL, LIN1, LIN2, LIN3, LIN4
605      LOGICAL URCTL, RULCTL, UCTLOD(3), RCTLOD(3), UPRCNT, UXPcnt, RPRCNT,
606      &      RXPCNT, URBCST, RULCST
607      INTEGER SPCE(210, 28), AREACN(4), UALCON(4), OPTVAL(8)
608      DIMENSION ITYP(210), NPOS(210), AUNIT(210, 2), KTITLE(4, 40),
609      &      SURFEA(210, 3)
610      INTEGER TLEUR1(12), TLEUR2(12), APOE(210)
611      DIMENSION POIF(210, 3), TCOST(210, 3), PUCOST(210, 3), TRNSMN(210)
612      COMMON /WTRTF/ LINPUT, LUDURB, LUDRNC, LIN1, LIN2, LIN3, LIN4, OTHURB,
613      &      OTHRUL, LUDRCP, TYPCND, JIN, JOUT, L4UAL, KOUT
614      COMMON /WTRIO/ IN, IOUT, NSRCE, KTITLE, NTLF, NURB, NRUNCP, NRUCRP
615      COMMON /WRKSH1/ SRCE, ITYP, NPOS, APOE, AUNIT, SURFEA, TLEUR1, TLEUR2
616      COMMON /WRKSH6/ POIF, TCOST, PUCOST, TRNSMN
617      COMMON /WTRCON/ UCTLOD, RCTLOD, URBCST, RULCTL, UPRCNT, UXPcnt, RPRCNT,
618      &      RXPCNT, OPTVAL, NOPT, URBCST, RULCST, AREACN, UALCON
619      C *****
620      IER=1
621      IF (IS.GT.3) RETURN
622      IER=2
623      IF (KEYWRD.EQ.1 .AND. NURB.EQ.0) RETURN
624      IF (KEYWRD.EQ.2 .AND. NRUCRP.EQ.0) RETURN
625      IF (KEYWRD.EQ.1) WRITE(IOUT, 5) IS
626      IF (KEYWRD.EQ.2) WRITE(IOUT, 6) IS
627      5 FORMAT('0', 5X, 'STAGE', 12, ' COSTS OF CONTROL FOR URBAN SOURCES')
628      6 FORMAT('0', 5X, 'STAGE', 12, ' COSTS OF CONTROL FOR RURAL SOURCES')
629      IF (IFLAG.EQ.1) GO TO 27
630      IF (KEYWRD.EQ.2) GO TO 20
631      C      HERE FOR URBAN COSTS FOR ALL SOURCES
632      L1=LENGTH(TLEUR1, 12)
633      L2=LENGTH(TLEUR2, 12)
634      IF (OTHURB.EQ.0) WRITE(IOUT, 10)
635      IF (OTHURB.EQ.1) WRITE(IOUT, 11) (TLEUR1(J), J=1, 12)
636      IF (OTHURB.EQ.2) WRITE(IOUT, 12) (TLEUR1(J), J=1, 12), (TLEUR2(J), J=1, 12)
637      10 FORMAT('CENTER PER UNIT COSTS FOR POINT, STORM, COMBINED SOURCES')
638      11 FORMAT('CENTER PER UNIT COSTS FOR POINT, STORM, COMBINED, ', 12A1,
639      &      ' SOURCES')
640      12 FORMAT('CENTER PER UNIT COSTS FOR POINT, STORM, COMBINED, ', 12A1,
641      &      ', ', 12A1, ' SOURCES')
642      C      INPUT OF COSTS
643      READ(IN, 15) COST1, COST2, COST3, COST4, COST5
644      15 FORMAT(5F10.0)
645      GO TO 30
646      C      HERE FOR RURAL COSTS FOR ALL SOURCES
647      20 WRITE(IOUT, 22)
648      22 FORMAT('CENTER PER UNIT COSTS FOR RURAL SOURCES')
649      READ(IN, 25) COST6
650      25 FORMAT(F10.0)
651      GO TO 30
652      C      HERE FOR COSTS WHICH VARY BETWEEN SOURCES
653      27 WRITE(IOUT, 28)
654      28 FORMAT('CENTER PER UNIT COSTS PER SOURCE')
655      30 DO 50 I=1, NSRCE
656      IF (ITYP(I).EQ.1) GO TO 50
657      IF (ITYP(I).EQ.2 .AND. KEYWRD.EQ.1) GO TO 50
658      IF (ITYP(I).NE.2 .AND. KEYWRD.EQ.2) GO TO 50
659      IF (IFLAG.EQ.0) GO TO 35

```

```

660      WRITE(IOUT,32)(SRCE(I,J),J=1,28)
661      FORMAT(1X,28A1)
662      READ(IV,25) PUCOST(I,IS)
663      GO TO 40
664      35 IF(ITYP(I).EQ.2) PUCOST(I,IS)=COST0
665         IF(ITYP(I).GE.3.AND.ITYP(I).LE.5) PUCOST(I,IS)=COST1
666         IF(ITYP(I).EQ.6) PUCOST(I,IS)=COST2
667         IF(ITYP(I).EQ.7) PUCOST(I,IS)=COST3
668         IF(ITYP(I).EQ.8) PUCOST(I,IS)=COST4
669         IF(ITYP(I).EQ.9) PUCOST(I,IS)=COST5
670      40 TCOST(I,IS)=PUCOST(I,IS)*AUNIT(I,I)
671      50 CONTINUE
672      C      DONE
673      IF(KEYWRD.EQ.1) URCST=.TRUE.
674      IF(KEYWRD.EQ.2) RUCST=.TRUE.
675      KOUNTR=KOUNTR+1
676      IER=0
677      RETURN
678      END
679      C
680      > SUBROUTINE WTRTRN(KPOS)
681      C
682      C      ROUTINE TO ALTER EFFECTIVE TRANSMISSION RATES
683      C
684      LOGICAL LINPUB,LODURB,LODRNC,LODRCP,L4JAL,LIN1,LIN2,LIN3,LIN4
685      INTEGER OTHURB,OTHRUL,SRCE(210,28),TYPCOD(9,12)
686      INTEGER TLEUR1(12),TLEUR2(12),APOE(210)
687      DIMENSION AUNIT(210,2),SURFEA(210,3),TRNSMN(210),PDIF(210,3),
688      &          TCOST(210,3),PUCOST(210,3),ITYP(210),NPOS(210),
689      &          KTITLE(4,40)
690      COMMON /WTRTF/ LINPUB,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHURB,
691      &          OTHRUL,LODRCP,TYPCOD,JIN,JOUT,L4UAL,KOUT,LANK
692      COMMON /WRKSH1/ SRCE,ITYP,NPOS,APOE,AUNIT,SURFEA,TLEUR1,TLEUR2
693      COMMON /WRKSH6/ PDIF,TCOST,PUCOST,TRNSMN
694      COMMON /WTRIO/ IN,IOUT,NSRCE,KTITLE,NTLE,NURB,NRUNCP,NRUCRP
695      C
696      WRITE(IOUT,5)
697      5 FORMAT('ENTER EFFECTIVE TRANSMISSION PER SOURCE')
698      IF(KPOS.EQ.0) GO TO 50
699      K1=KPOS
700      M1=ITYP(KPOS)
701      IF(M1.EQ.6) K1=KPOS-5
702      IF(M1.EQ.7) K1=KPOS-6
703      IF(M1.EQ.8) K1=KPOS-7
704      IF(M1.EQ.9) K1=KPOS-8
705      C
706      WRITE(IOUT,10)(SRCE(K1,J),J=1,28),(TYPCOD(M1,J),J=1,12)
707      10 FORMAT(1X,40A1)
708      READ(IN,15) TRNSMN(KPOS)
709      15 FORMAT(F10.0)
710      GO TO 99
711      C
712      C      HERE IF KPOS = 0 I.E. PROMPT FOR INPUT FOR ALL SOURCES
713      C
714      50 DO 60 I=1,NSRCE
715         M1=ITYP(I)
716         WRITE(IOUT,10)(SRCE(I,J),J=1,28),(TYPCOD(M1,J),J=1,12)
717         READ(IN,15) TRNSMN(I)
718      60 CONTINUE
719      C

```

```

720      99 RETURN
721      END
722  C
723  > SUBROUTINE WTRANK(IS)
724  C
725  C      ROUTINE TO RANK CONTROL STRATEGIES BY COST EFFECTIVENESS
726  C
727  C *****
728      DIMENSION CPUTMP(540),KERTMP(540)
729      LOGICAL LINPUT,LODURB,LODRNC,LODRCP,L4UAL,LIN1,LIN2,LIN3,LIN4
730      LOGICAL URBCTL,RULCTL,UCTLDD(3),RCTLDD(3),UPRCNT,UXPCNT,RPRCNT,
731      &      RXPCNT,URBCST,RULCST,LRANK
732      INTEGER OTHURB,OTHRUL,SRCE(210,28),TYPCOD(9,12),OPTVAL(8)
733      DIMENSION ITYP(210),NPOS(210),AUNIT(210,2),KTITLE(4,40),
734      &      SURFFA(210,3),CTLDD(210,3),CTUSLE(3,30,5)
735      DIMENSION STMCTL(30,3),STMARA(30,3),COMCTL(30,3),COMARA(30,3),
736      &      OT1CTL(30,3),OT1ARA(30,3),OT2CTL(30,3),OT2ARA(30,3)
737      DIMENSION PFLOW(30),PCONC(30),UALS(30),JALC(30),W2UAL1(30),
738      &      W2UAL2(30),TRNSMN(210)
739      INTEGER TLFURL(12),TLFUR2(12),TLERU1(12),TLERU2(12),TLERU3(12),
740      &      APDF(210),APEACN(4),UALCON(4)
741      DIMENSION GRASA(30),UALG(30),WLNDAA(30),UALW(30),W3OTA1(30),
742      &      W3UAL1(30),W3UAL2(30),W3UAL3(30),W3OTA2(30),W3OTA3(30)
743      REAL LODRNI(30),LODRN2(30),LOD1W3(30),LOD2W3(30),LOD3W3(30)
744      DIMENSION USLER(30),USLEK(30),USLELS(30),USLEC(30),W4UAL(30),
745      &      USLEP(30),SOILS1(30),SOILS2(30),TTLER(30),RATIO(30)
746      DIMENSION PDIF(210,3),TCOST(210,3),PUCOST(210,3)
747      DIMENSION PMOUTH(210),DIEMTH(210,3),CPJREM(210,3),KERANK(210,3)
748      COMMON /WTRIC/ IN,IOUT,NSRCE,KITLLE,NTLE,NURB,NRUNCP,NRUCRP
749      COMMON /WTRIF/ LINPUT,LODURB,LODRNC,LIN1,LIN2,LIN3,LIN4,OTHURB,
750      &      OTHRUL,LODRCP,TYPCOD,JIN,JOIT,L4UAL,KOUT,LRANK
751      COMMON /WRKSH1/ SRCE,ITYP,NPOS,APDF,AUNIT,SURFFA,TLFURL,TLFUR2
752      COMMON /WRKSH2/ PFLOW,PCONC,UALS,UALC,W2UAL1,W2UAL2
753      COMMON /WRKSH3/ GRASA,UALG,WLNDAA,UALW,LODRNI,LODRN2,
754      &      W3OTA1,W3UAL1,W3UAL2,W3UAL3,W3OTA2,W3OTA3,
755      &      TLERU1,TLFUR2,TLERU3,LOD1W3,LOD2W3,LOD3W3
756      COMMON /WRKSH4/ USLER,USLEK,USLELS,USLEC,USLEP,SOILS1,
757      &      SOILS2,TTLER,W4UAL,RATIO
758      COMMON /WRKSH6/ PDIF,TCOST,PUCOST,TRNSMN,PMOUTH,DIEMTH,CPJREM,
759      &      KERANK
760      COMMON /WTRCON/ UCTLDD,RCTLDD,URBCTL,RULCTL,UPRCNT,UXPCNT,RPRCNT,
761      &      RXPCNT,OPTVAL,NOPT,URBCST,RULCST,APDFACN,UALCON
762      DIMENSION CTLFLW(30,3),CTLCNC(30,3)
763      DIMENSION CTLE(30,3),SOILC1(30,3),SOILC2(30,3),CTLER0(30,3),
764      &      CUALW4(30,3)
765      DIMENSION TOTCST(3),TOTREM(3),KNDX(540),FINCST(540),FINRFM(540),
766      &      ISFIN(540)
767      INTEGER FINSRC(540,43)
768      COMMON /WRKSH8/ TOTCST,TOTREM,TOTLDD,TOTMTH,KNDX,FINSRC,FINCST,
769      &      FINRFM,ISFIN,NUM
770      COMMON /URBCTL/ CTLFLW,CTLCNC,CTLDD,STMCTL,STMARA,COMCTL,
771      &      COMARA,OT1CTL,OT1ARA,OT2CTL,OT2ARA
772      COMMON /RULCTL/ CTUSLE,CTLE,SOILC1,SOILC2,CTLFRO,CJALW4
773  C *****
774      DATA IRANK,IOSH/' ','-' /
775      TOTLDD=0.0
776      TOTMTH=0.0
777      DO 5 I=1,3
778      TOTCST(I)=0.0
779      5 TOTREM(I)=0.0

```

```

780      DO 7 I=1,540
781      CPUTMP(I)=0.0
782      7 KERTMP(I)=0
783      C
784      DO 10 I=1,NSRCE
785      PMOUTH(I)=TRNSMN(I)*AUNIT(I,2)
786      TOTMTH=TOTMTH+PMOUTH(I)
787      TOTLCO=TOTLCO+AUNIT(I,2)
788      IF(IITYP(I).EQ.1 .OR. AUNIT(I,2).EQ.0.0) GO TO 10
789      DO 8 J=1,IS
790      DIFMTH(I,J)=POIF(I,J)*TRNSMN(I)
791      IF(DIFMTH(I,J).EQ.0.0) GO TO 8
792      CPUREM(I,J)=TCOST(I,J)/DIFMTH(I,J)
793      TOTCST(J)=TOTCST(J)+TCOST(I,J)
794      TOTREM(J)=TOTREM(J)+DIFMTH(I,J)
795      8 CONTINUE
796      10 CONTINUE
797      C
798      K=0
799      DO 20 J=1,IS
800      DO 18 I=1,NSRCE
801      IF(IITYP(I).EQ.1 .OR. AUNIT(I,2).EQ.0.0) GO TO 18
802      K=K+1
803      CPUTMP(K)=CPUREM(I,J)
804      KNDX(K)=K
805      FINREM(K)=DIFMTH(I,J)
806      FINCST(K)=TCOST(I,J)
807      ISFIN(K)=J
808      L1=IITYP(I)
809      I1=0
810      IF(L1.EQ.2) I1=2
811      IF(L1.EQ.3) I1=1
812      IF(L1.EQ.4) I1=2
813      IF(L1.EQ.6) I1=3
814      IF(L1.EQ.7) I1=2
815      IF=12-(2*I1)
816      IF(L1.EQ.3) IE=IE-1
817      IF(L1.EQ.6) IF=5
818      DO 12 I2=1,IE
819      12 FINSRC(K,I2)=TYPCOD(L1,I2+I1)
820      FINSRC(K,IE+1)=IBLNK
821      FINSRC(K,IE+2)=IDSH
822      FINSRC(K,IE+3)=IBLNK
823      L2=0
824      IR=IE+4
825      IE=IB+27
826      I3=1
827      IF(L1.EQ.6) I3=I-1
828      IF(L1.EQ.7) I3=I-2
829      IF(L1.EQ.8) I3=I-3
830      IF(L1.EQ.9) I3=I-4
831      DO 14 I2=IB,IE
832      L2=L2+1
833      14 FINSRC(K,I2)=SRCE(I3,L2)
834      C
835      18 CONTINUE
836      20 CONTINUE
837      C      RANK USING SHELL SORT
838      IGAP=K/2
839      22 IF(IGAP.LE.6) GO TO 38

```

```

840      IIGAP=IGAP+1
841      DO 30 I=IIGAP,K
842          J=I - IGAP
843      25  IF(J .LE. 0) GO TO 30
844          JG=J + IGAP
845          IF(CPUTMP(J) .LE. CPUTMP(JG)) GO TO 28
846      C      EXCHANGE PLACES IN TEMPORARY DATA ARRAY
847          AK=CPUTMP(J)
848          CPUTMP(J)=CPUTMP(JG)
849          CPUTMP(JG)=AK
850      C      EXCHANGE PLACES OF INDEXES
851          IK=KNDX(J)
852          KNDX(J)=KNDX(JG)
853          KNDX(JG)=IK
854      28  J=J - IGAP
855          GO TO 25
856      30  CONTINUE
857          IGAP=IGAP/2
858          GO TO 22
859      38  CONTINUE
860      C      DONE WITH SORT
861          DO 40 I=1,K
862              M=KNDX(I)
863              KERTMP(M)=I
864      40  CONTINUE
865          K2=0
866          DO 60 J=1,IS
867              DO 50 I=1,NSRCE
868                  IF(ITYP(I).EQ.1 .OR. AUNIT(I,2).EQ.3.0) GO TO 50
869                  K2=K2+1
870                  KERANK(I,J)=KERTMP(K2)
871      50  CONTINUE
872      60  CONTINUE
873          LRANK=.TRUE.
874          NUM=K
875          RETURN
876      END

```

END OF FILE

Data Files

The example data file listings that follow were generated from terminal sessions similar to that included in this manual. It is important to note that the data file structure can change for some parameters depending upon the commands and key words utilized in a particular run. The data files are compatible with the NETWORK program.

32 0 0 3 0 0 0 0
SANDUSKY RIVER

LAKE FRIE

LOSS CREEK		1A	36.9700	368.6997					
11.8300	10.0000		25.0400	10.0000	0.0	0.0	0.0	0.0	0.0
LOSS CREEK		2A	189.8400	18.0678					
125.0000	0.3500		0.4020	0.2330	1.0000	0.00010355			
BUCHUS CITY		3A	13500.0000	13820.0000					
2.5000	4.0000								
STORM		6A	1.3000	325.0000					
250.0000									
COMBINED		7A	7.2000	6479.9961					
900.0000									
BROKEN SWORD		1B	21.7600	217.6000					
3.5900	10.0000		18.1700	10.0000	0.0	0.0	0.0	0.0	0.0
BROKEN SWORD		2B	187.1900	20.4012					
125.0000	0.3800		0.4240	0.2330	1.0000	0.00010355			
UPPER SANDUSKY RIVER		1B	47.1900	471.8999					
12.8100	10.0000		34.3800	10.0000	0.0	0.0	0.0	0.0	0.0
UPPER SANDUSKY RIVER		2B	274.6001	36.3436					
130.0000	0.4200		0.4260	0.2450	1.0000	0.00010355			
UPPER SANDUSKY CITY		3B	6250.0000	8292.0000					
1.5000	4.0000								
STORM		6B	0.0	0.0					
0.0									
COMBINED		7B	9.7000	2174.9998					
250.0000									

TYMOCHTEE CREEK		1C	56.1700	697.2998					
9.0400	25.0000		47.1300	10.0000	0.0	0.0	0.0	0.0	0.0
TYMOCHTEE CREEK		2C	528.3999	54.1723					
138.0000	0.3200		0.3570	0.2800	1.0000	0.00010355			
MIDDLE SANDUSKY		1C	85.2300	852.2998					
17.2700	10.0000		67.9600	10.0000	0.0	0.0	0.0	0.0	0.0
MIDDLE SANDUSKY		2C	542.3601	65.4346					
138.0000	0.3800		0.3810	0.2600	1.0000	0.00010355			
HONEY CREEK		1D	47.2400	472.3999					
3.1700	10.0000		44.0700	10.0000	0.0	0.0	0.0	0.0	0.0
HONEY CREEK		2D	343.5500	27.9635					
125.0000	0.3500		0.3380	0.2370	1.0000	0.00010355			
TIFFIN CITY		3D	26000.0000	17689.5977					
3.2000	4.0000								
STORM		6D	7.0000	1750.0000					
250.0000									
COMBINED		7D	10.4000	10399.9961					
1000.0000									
WOLF CREEK		1E	30.4600	304.5999					
7.5700	10.0000		22.8900	10.0000	0.0	0.0	0.0	0.0	0.0
WOLF CREEK		2E	343.1101	17.5262					
125.0000	0.2900		0.2560	0.2370	1.0000	0.00010355			
ROCK CREEK		1E	68.6100	686.0999					
12.1400	10.0000		56.4700	10.0000	0.0	0.0	0.0	0.0	0.0
ROCK CREEK		2E	409.8999	40.9451					
125.0000	0.3400		0.4270	0.2370	1.0000	0.00010355			
FREMONT CITY		3E	19700.0000	28192.7891					
5.1000	4.0000								
STORM		6E	1.4000	349.9998					
250.0000									
COMBINED		7E	12.3000	12300.0000					

1000.0000									
LOWER SANDUSKY (SOIL 1)	1F	84.5400	995.3999						
10.0000	25.0000	74.5400	10.0000	0.0	0.0	0.0	0.0	0.0	
LOWER SANDUSKY (SOIL 1)	2F	243.5500	24.8614						
125.0000	0.3200	0.4100	0.2680	1.0000	0.00010355				
LOWER SANDUSKY (SOIL 2)	1F	8.2500	82.5000						
8.2500	10.0000	0.0	10.0000	0.0	0.0	0.0	0.0	0.0	
LOWER SANDUSKY (SOIL 2)	2F	100.0000	8.3611						
125.0000	0.2800	0.4340	0.2370	1.0000	0.00010355				
####									
1	1	1	2						
8374.2578									
125.0000	0.3500	0.4020	0.1080	1.0000	1.8994	4.2605	80880.5625	1.00	
3455.0000									
2.5000	1.0000	1.3000	187.5000	7.2000	846.0000	0.0	0.0	0.0	
243.7500									
1.5000	1.0000	0.0	-0.0	8.7000	235.0000	0.0	0.0	0.0	
6091.1992									
3.2000	1.0000	7.0000	187.5000	10.4000	940.0000	0.0	0.0	0.0	
8667.7148									
125.0000	0.3800	0.4240	0.0990	1.0000	1.9939	4.4722	83715.4375	1.00	
15278.1523									
130.0000	0.4200	0.4260	0.1030	1.0000	2.3957	5.3736	147559.9375	1.00	
2073.0000									
5.1000	1.0000	1.4000	187.5000	12.3000	940.0000	0.0	0.0	0.0	
0.0									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2044.4998									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
20893.4922									
138.0000	0.3200	0.3570	0.1080	1.0000	1.7026	3.8190	201795.9375	1.00	
27682.0977									
138.0000	0.3800	0.3810	0.1100	1.0000	2.1978	4.9296	267360.1875	1.00	
11916.1523									
125.0000	0.3500	0.3380	0.1010	1.0000	1.4935	3.3500	115089.2500	1.00	
4422.3784									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1312.5000									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
9775.9961									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
7986.1250									
125.0000	0.2900	0.2560	0.1080	1.0000	1.0022	2.2480	77131.8125	1.00	
18139.1602									
125.0000	0.3400	0.4270	0.1050	1.0000	1.9055	4.2740	175191.3125	1.00	
7048.1992									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
262.4998									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
11562.0000									
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
10760.2227									
125.0000	0.3200	0.4100	0.1160	1.0000	1.9024	4.2671	103924.6875	1.00	
3492.3647									
125.0000	0.2800	0.4340	0.0990	1.0000	1.5038	3.3730	33730.4180	1.00	
2635.7368									
125.0000	0.3500	0.4020	0.0340	1.0000	0.5980	1.3413	25462.3984	1.00	
1727.5000									
2.5000	0.5000	1.3000	140.6250	7.2000	634.5000	0.0	0.0	0.0	
182.8125									

1.5000	0.5000	0.0	-0.0	8.7000	176.2500	0.0	0.0	0.0
4568.3984	0.5000	7.0000	140.6250	10.4000	705.0000	0.0	0.0	0.0
3.2000								
3063.6863	0.3800	0.4240	0.0350	1.0000	0.7049	1.5811	29596.3867	1.00
125.0000								
5042.1523	0.4200	0.4260	0.0340	1.0000	0.7908	1.7738	48709.1328	1.00
130.0000								
1036.5000	0.5000	1.4000	140.6250	12.3000	705.0000	0.0	0.0	0.0
5.1000								
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
1533.3748	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
6188.9102	0.3200	0.3570	0.0320	1.0000	0.5045	1.1316	59791.3789	1.00
138.0000								
8805.8438	0.3800	0.3810	0.0350	1.0000	0.6993	1.5685	85069.1250	1.00
138.0000								
4010.4980	0.3500	0.3380	0.0340	1.0000	0.5028	1.1277	38742.9492	1.00
125.0000								
2211.1990	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
984.3750	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
7331.9961	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
2365.6472	0.2900	0.2560	0.0320	1.0000	0.2970	0.6661	22853.9023	1.00
125.0000								
5699.5742	0.3400	0.4270	0.0330	1.0000	0.5989	1.3433	55060.1250	1.00
125.0000								
3524.1001	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
196.8749	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
8671.5000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0								
3338.5676	0.3200	0.4100	0.0360	1.0000	0.5904	1.3243	32252.4844	1.00
125.0000								
1163.8401	0.2800	0.4340	0.0330	1.0000	0.5013	1.1243	11243.4648	1.00
125.0000								

\$\$\$

42	420999.0625	420999.0625				
234369.9375	107197.4375	720133.8750	10760062.0000			
1.0000	368.6997					
0.0	0.0	0.0	0.0	0		
0.0	0.0	0.0	0.0	0		
1.0000	18767.7773					
9693.5195	65.0000	12339.5977	9693.5195	1.2730	7	
5738.5508	3000.0000	569519.9375	5738.5508	99.2445	23	
1.0000	13820.0000					
10365.0000	2.4000	32399.9922	10365.0000	3.1259	13	
1727.5000	3.6000	48600.0039	1727.5000	28.1331	17	
1.0000	325.0000					
81.2500	7400.0000	9620.0000	81.2500	118.4000	31	
60.9375	8600.0000	11180.0000	60.9375	183.4666	38	
1.0000	6479.9961					
388.7960	7400.0000	53279.9961	388.7960	137.0391	36	
1522.8008	24700.0000	177939.9375	1522.8008	116.7848	29	
1.0000	217.6000					
0.0	0.0	0.0	0.0	0.0	0	
0.0	0.0	0.0	0.0	0.0	0	

1.0000	20401.1836				
11733.4688	65.0000	12167.3477	11733.4688	1.0370	4
5604.0273	3000.0000	561570.0000	5604.0273	100.2083	24
1.0000	471.8999				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	36343.5820				
21065.4297	65.0000	17949.0039	21065.4297	7.8473	1
10236.0000	3000.0000	823800.2500	10236.0000	80.4807	17
1.0000	8292.0000				
6219.0000	2.4000	14999.9961	6219.0000	2.4120	12
1036.5000	3.6000	22500.0000	1036.5000	21.7077	16
1.0000	0.0				
0.0	7400.0000	0.0	0.0	0.0	0
0.0	8600.0000	0.0	0.0	0.0	0
1.0000	2174.9998				
130.5000	7400.0000	64379.9961	130.5000	493.3333	42
511.1250	24700.0000	214889.9375	511.1250	420.4253	41
1.0000	697.2998				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	54172.2852				
33278.7930	65.0000	34345.9922	33278.7930	1.0321	3
14704.5820	3000.0000	1585199.0000	14704.5820	107.8031	27
1.0000	852.2998				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	65434.5820				
37752.4844	65.0000	35253.4063	37752.4844	0.9338	2
18876.2539	3000.0000	1627080.0000	18876.2539	86.1972	20
1.0000	472.3999				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	27963.4844				
16047.3320	65.0000	22330.7500	16047.3320	1.3916	9
7905.6523	3000.0000	1030650.1250	7905.6523	130.3688	35
1.0000	17689.5977				
13267.1992	2.4000	62399.9883	13267.1992	4.7033	14
2211.1995	3.6000	93600.0000	2211.1995	42.3300	18
1.0000	1750.0000				
437.5000	7400.0000	51800.0000	437.5000	118.4000	30
328.1250	8500.0000	60200.0000	328.1250	183.4667	39
1.0000	10399.9961				
624.0000	7400.0000	76959.9375	624.0000	123.3332	32
2444.0000	24700.0000	256879.9375	2444.0000	105.1064	25
1.0000	304.5999				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	17526.1836				
9540.0586	65.0000	22302.1563	9540.0586	2.3377	11
5620.4766	3000.0000	1029330.3125	5620.4766	183.1393	37
1.0000	686.0999				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	40945.0820				
22805.9219	65.0000	26643.4922	22805.9219	1.1683	6
12439.5859	3000.0000	1229699.0000	12439.5859	98.8537	22
1.0000	28192.7891				
21144.5898	2.4000	47351.9922	21144.5898	2.2394	10
3524.0000	3.6000	71028.0000	3524.0000	20.1549	15

1.0000	349.9998				
87.5000	7400.0000	10359.9961	87.5000	118.3999	29
65.6248	8600.0000	12039.9961	65.6248	183.4671	47
1.0000	12300.0000				
738.0000	7400.0000	91020.0000	738.0000	123.3333	33
2890.5000	24700.0000	303810.0000	2890.5000	105.1064	26
1.0000	995.3999				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	24861.3867				
14101.1641	65.0000	15830.7500	14101.1641	1.1227	5
7421.6523	3000.0000	730650.0000	7421.6523	98.4484	21
1.0000	82.5000				
0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0
1.0000	8361.0938				
4868.7266	65.0000	6500.0000	4868.7266	1.3351	8
2328.5247	3000.0000	300000.0000	2328.5247	128.8359	34
CROPLAND - LOSS CREEK		6	1	9693.5195	12339.5977
MUNICIPAL - BUCYRUS CITY		10	1	10365.0000	32399.9922
STORM - BUCYRUS CITY		9	1	81.2500	9620.0000
COMBINED - BUCYRUS CITY		5	1	388.7969	53279.9961
CROPLAND - BROKEN SWORD		20	1	11733.4688	12167.3477
CROPLAND - UPPER SANDUSKY RIVER		16	1	21065.4297	17849.0039
MUNICIPAL - UPPER SANDUSKY CITY		1	1	6219.0000	14999.9961
COMBINED - UPPER SANDUSKY CITY		21	1	130.5000	64379.9961
CROPLAND - TYMOCHTEE CREEK		11	1	33278.7930	34345.9922
CROPLAND - MIDDLE SANDUSKY		17	1	37752.4844	35253.4063
CROPLAND - HONEY CREEK		15	1	16047.3320	22330.7500
MUNICIPAL - TIFFIN CITY		7	1	13267.1992	62399.9883
STORM - TIFFIN CITY		2	1	437.5000	51800.0000
COMBINED - TIFFIN CITY		12	1	624.0000	76959.9375
CROPLAND - WOLF CREEK		38	1	9540.0586	22302.1563
CROPLAND - ROCK CREEK		28	1	22805.9219	26643.4922
MUNICIPAL - FREMONT CITY		23	1	21144.5898	47351.9922
STORM - FREMONT CITY		33	1	87.5000	10359.9961
COMBINED - FREMONT CITY		27	1	738.0000	91020.0000
CROPLAND - LOWER SANDUSKY (SOIL 1)		31	1	14101.1641	15830.7500
CROPLAND - LOWER SANDUSKY (SOIL 2)		41	1	4868.7266	6500.0000
CROPLAND - LOSS CREEK		37	2	5738.5508	569519.9375
MUNICIPAL - BUCYRUS CITY		22	2	1727.5000	48600.0039
STORM - BUCYRUS CITY		26	2	60.9375	11180.0000
COMBINED - BUCYRUS CITY		35	2	1522.8008	177839.9375
CROPLAND - BROKEN SWORD		40	2	5604.0273	561570.0000
CROPLAND - UPPER SANDUSKY RIVER		30	2	10236.0000	823800.2500
MUNICIPAL - UPPER SANDUSKY CITY		25	2	1036.5000	22500.0000
COMBINED - UPPER SANDUSKY CITY		18	2	511.1250	214889.9375
CROPLAND - TYMOCHTEE CREEK		13	2	14704.5820	1585199.0000
CROPLAND - MIDDLE SANDUSKY		3	2	18876.2539	1627080.0000
CROPLAND - HONEY CREEK		14	2	7905.6523	1030650.1250
MUNICIPAL - TIFFIN CITY		19	2	2211.1995	93600.0000
STORM - TIFFIN CITY		42	2	328.1250	60200.0000
COMBINED - TIFFIN CITY		32	2	2444.0000	256879.9375
CROPLAND - WOLF CREEK		4	2	5620.4766	1229330.3125
CROPLAND - ROCK CREEK		36	2	12439.5859	1229699.0000
MUNICIPAL - FREMONT CITY		24	2	3524.0991	71028.0000
STORM - FREMONT CITY		34	2	65.6248	12039.9961
COMBINED - FREMONT CITY		39	2	2890.5000	303810.0000
CROPLAND - LOWER SANDUSKY (SOIL 1)		29	2	7421.6523	730650.0000
CROPLAND - LOWER SANDUSKY (SOIL 2)		8	2	2328.5247	300000.0000

SECTION 2 - NETWORK PROGRAM

Commands

Table 2 includes a list of the commands utilized by the program NETWORK. This program simply reads in stored data from Worksheet 7 for up to 7 river basins, performs the cost effectiveness ranking, and outputs one master Worksheet 8. This program is designed to support the network concept discussed in Chapter 4 of the Handbook.

TABLE 2

Legal Commands for NETWORK Program

NAME(n) basin name

Allows for the input of basin names. "n" is the basin number, a positive non-zero integer less than or equal to 7. "basin name" is a maximum 20 character label for the sub-basin.

READ $i_1, i_2, \dots, i_7$

Causes the input of sub-basin data. "i" is the input device number of a WATERSHED data file. Multiple device numbers, separated by commas are accepted.

RANK(m)

Causes the ranking and output of the basin pollutant loads. "m" is any optional positive, non-zero output file number operating just like the WORKSHEET(m) command of WATERSHED.

STOP

Terminates the program.

Sample Terminal Session

In this session the computer prompts are marked by a triangle. It should be noted that the READ command is used to designate the file numbers that contain the results from the main WATERSHED program for the various river basins in the network. There must be a unique NAME command for each file number that follows the READ command.

RUN BASIN 9=-WATER.ONE 9=WATER.DAT 3=-TEMPFIL T=5
▷ #EXECUTION BEGINS

▷ WATERSHED MULTIPLE BASINS PROGRAM

▷ ENTER COMMAND
NAME(1)SANDUSKY ONE STAGE

▷ ENTER COMMAND
NAME(2)SANDUSKY TWO STAGE

▷ ENTER COMMAND
READ 8,9

▷ ENTER COMMAND
RANK(3)

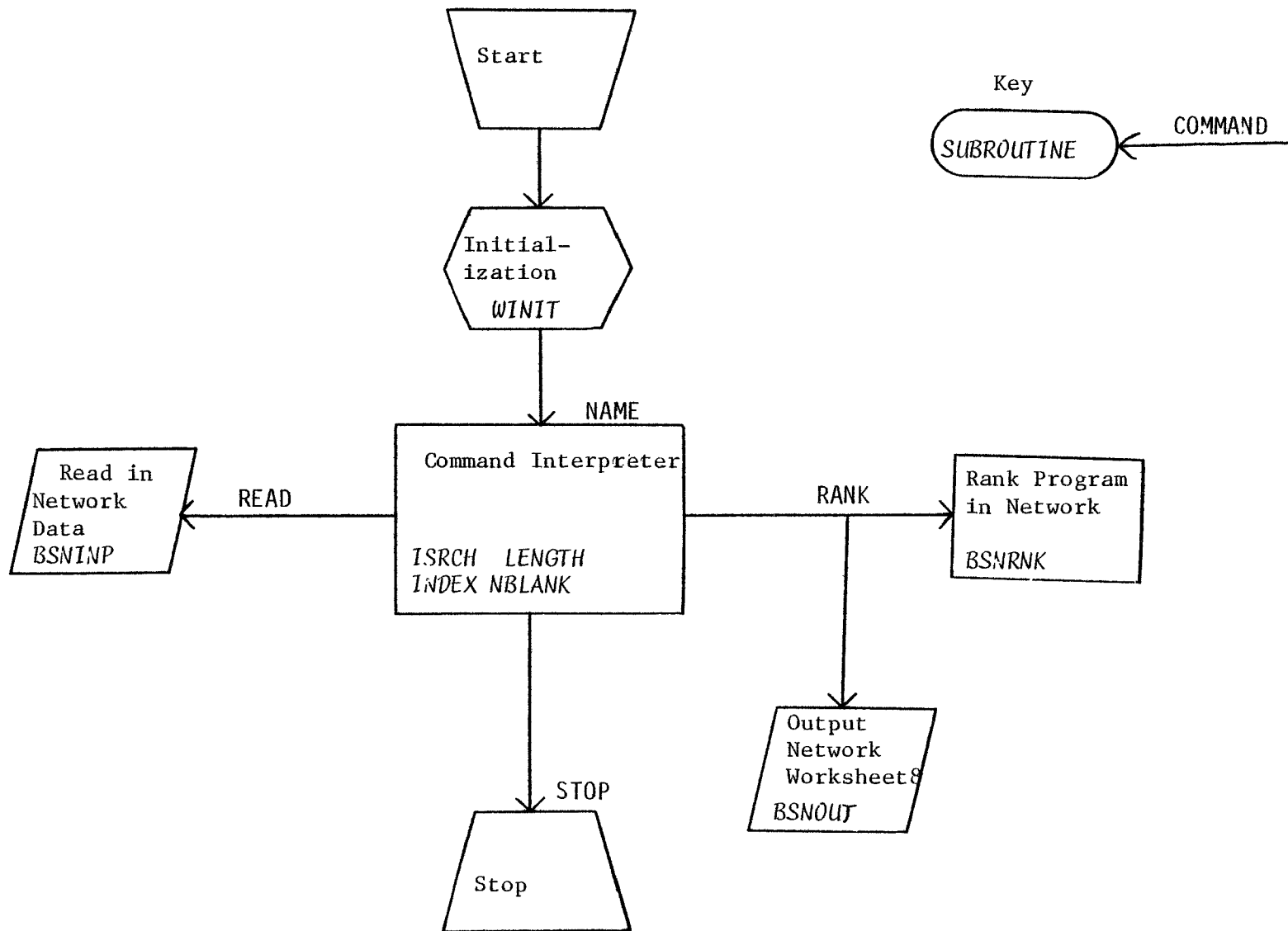
▷ ENTER COMMAND
STOP
▷ #EXECUTION TERMINATED

Computer Flow Chart and Listings

- Figure 3 contains a flow chart from the NETWORK program which includes the commands and subroutine names. The following program listing is documented to describe the workings of the various sections of the program. The command sections and subrouting are underlined and marked by a bracket.
- This entire program is in one source file called NTWRK.

Figure 3

Network Program Flow Chart



NETWORK PROGRAM

```

1  C
2  C      MAIN PROGRAM FOR MULTIPLE WATERSHED BASINS
3  C      PROGRAM ACCEPTS BASIN NAMES, READS INPUT DATA FROM
4  C      A WATERSHED DATA FILE AND OUTPUTS RANKED COST-
5  C      EFFECTIVENESS RATIOS AMONG ALL BASINS
6  C
7  C *****
8  DIMENSION AUNIT(7,210),TRNSMN(7,210),PMOUTH(7,210),TCOST(7,210,3),
9  & DIFMTH(7,210,3),CPUREM(7,210,3),FINREM(3780),FINCST(3780)
10 INTEGER SRCE(7,210,281),ITYP(7,210),ISTAGE(7),KERANK(7,210,3),
11 & FINSRC(3780,43),ISFIN(3780),KNDX(3780),NAMBSN(7,20),
12 & KTRANK(7,210,3),NSRCE(7),BASIN(3780,20)
13 COMMON /WTRIO/ IN,IOU
14 COMMON /WTRINP/ AUNIT,TRNSMN,PMOUTH,TCOST,DIFMTH,CPUREM,NSRCE
15 COMMON /WTRIN2/ SRCE,ITYP,ISTAGE,KERANK,NAMBSN,BASIN
16 COMMON /WTROUT/ FINREM,FINCST,FINSRC,ISFIN,KNDX,KTRANK,BSNTL,NUM
17 C *****
18 INTEGER CMD(30),BLNK,NMBR(9),LPAREN,RPAREN,COMA
19 DATA NMBR/'1','2','3','4','5','6','7','8','9'/
20 DATA BLNK,LPAREN,RPAREN,COMA/' ','(','(',')','(',')'/
21 C
22 NBASN = 0
23 KOUNTR = 1
24 CALL WTRIT
25 WRITE(IOU,2)
26 2 FORMAT(' ',10X,'WATERSHED MULTIPLE BASINS PROGRAM')
27 DO 12 I=1,30
28 12 CMD(I)=BLNK
29 C      READ COMMAND LINE
30 WRITE(IOU,15)
31 15 FORMAT(' -ENTER COMMAND')
32 READ(IN,16) (CMD(I),I=1,30)
33 16 FORMAT(30A1)
34 C      FIND COMMAND NUMBER, LENGTH OF COMMAND, POSITION OF 1ST BLANK SPACE
35 NUMCMD = 1SRCH(CMD,LEN,IPOS)
36 IF(NUMCMD .LE. 0) GO TO 800
37 C
38 C      NUMCMD      COMMAND
39 C      1           NAME
40 C      2           RANK
41 C      3           READ
42 C      4           STOP
43 C
44 GO TO(100,200,300,900),NUMCMD
45 C
46 C      NAME COMMAND
47 C
48 100 IPOS1=INDEX(CMD,LPAREN,3)
49 IF(IPOS1 .EQ. -1) GO TO 810
50 IPOS1=IPOS1+1
51 IPOS1=NBLANK(CMD,IPOS1)
52 IF(IPOS1 .EQ. -1) GO TO 810
53 DO 105 J=1,7
54 M=J
55 IF(CMD(IPOS1) .EQ. NMBR(J)) GO TO 110
56 105 CONTINUE
57 GO TO 810
58 110 IPOS1=INDEX(CMD,RPAREN,IPOS1)
59 IF(IPOS1 .EQ. -1) GO TO 810

```

```

60      IREG=IPOS1 + 1
61      IFND=IBEG + 19
62      K = 0
63      DO 115 J=IREG,IFND
64          K=K + 1
65      115 NAMBSN(M,K)=CMD(J)
66      GO TO 10
67  C
68  C      RANK COMMAND
69  C
70      200 IOCHNL = IOUT
71          IPOS1=INDEX(CMD,LPAREN,3)
72          IF(IPOS1 .EQ. -1) GO TO 220
73          IPOS1=IPOS1 + 1
74          IPOS1=NBLANK(CMD,IPOS1)
75          IF(IPOS1 .EQ. -1) GO TO 810
76          DO 205 J=1,9
77              IOCHNL = J
78              IF(CMD(IPOS1) .EQ. NMRR(J)) GO TO 210
79      205 CONTINUE
80          GO TO 810
81      210 IPOS1=INDEX(CMD,RPAREN,IPOS1)
82          IF(IPOS1 .EQ. -1) GO TO 810
83      220 CALL BSNRNK(NBASN)
84          CALL BSNOUT(IOCHNL,NBASN)
85          GO TO 10
86  C
87  C      READ COMMAND
88  C
89      300 IF(IEN .LE. IPOS) GO TO 820
90          IPOS1=NBLANK(CMD,IPOS)
91      302 DO 305 J=1,9
92          IOCHNL = J
93          IF(CMD(IPOS1) .EQ. NMRR(J)) GO TO 310
94      305 CONTINUE
95          GO TO 810
96      310 NBASN = NBASN + 1
97          IF(NBASN .GT. 7) GO TO 830
98          CALL BSNINP(IOCHNL,NBASN,KOUNTR)
99          IPOS1 = IPOS1 + 1
100         IF(IEN .LE. IPOS1) GO TO 10
101         IPOS1=NBLANK(CMD,IPOS1)
102         IF(CMD(IPOS1) .NE. COMA) GO TO 810
103         IPOS1 = IPOS1 + 1
104         IPOS1=NBLANK(CMD,IPOS1)
105         IF(IPOS1 .EQ. -1) GO TO 810
106         GO TO 302
107  C
108  C      ERRORS
109  C
110      800 WRITE(IOUT,802)
111      802 FORMAT('OILLEGAL COMMAND')
112          GO TO 10
113      810 WRITE(IOUT,812)
114      812 FORMAT('OILLEGAL COMMAND ARGUMENT')
115          GO TO 10
116      820 WRITE(IOUT,822)
117      822 FORMAT('OMISSING FILE NUMBER FOR INPUT DATA')
118          GO TO 10
119      830 WRITE(IOUT,832)

```

```

120      832 FORMAT('OERROR - MAXIMUM IS 7 BASINS')
121      GO TO 10
122      C
123      C          THE END
124      C
125      900 CONTINUE
126      STOP
127      END
128      C
129      FUNCTION ISRCHCMD,LEN,IPOS)
130      C
131      C          WTRSHD FUNCTION TO DETERMINE COMMAND NUMBER, LENGTH OF COMMAND,
132      C          AND POSITION OF FIRST SPACE FOLLOWING THE COMMAND
133      C
134      INTEGER CMD(30),COMMAND(4,3),BLNK
135      DATA COMMAND
136      & /'N','R','R','S',
137      &  'A','A','E','T',
138      &  'M','N','A','O'/
139      DATA BLNK/' '/
140      ISRCH=0
141      DO 10 I=1,4
142      K=I
143      DO 5 J=1,3
144      IF (CMD(J).NE.COMMAND(I,J)) GO TO 10
145      IFOUND=J
146      5 CONTINUE
147      IF (IFOUND.EQ.3) GO TO 35
148      10 CONTINUE
149      RETURN
150      C
151      35 ISRCH=K
152      LEN=LENGTH(CMD,30)
153      IPOS=INDEX(CMD,BLNK,1)
154      RETURN
155      END
156      C
157      FUNCTION LENGTH(CMD,N)
158      C
159      C          FUNCTION TO FIND THE LENGTH OF CMD WITH DIMENSION N
160      C
161      C          INTEGER CMD(N),BLNK
162      C          DATA BLNK/' '/
163      C          K=N+1
164      C          DO 10 I=1,N
165      C          K=K-1
166      C          IF (CMD(K).NE.BLNK) GO TO 15
167      C          10 CONTINUE
168      C          LENGTH=-1
169      C          RETURN
170      C
171      15 LENGTH=K
172      RETURN
173      END
174      C
175      FUNCTION INDEX(CMD,ISTVAL,IBEG)
176      C
177      C          FUNCTION TO LOCATE ISTVAL IN CMD BEGINNING SEARCH AT IBEG
178      C
179      INTEGER CMD(30),ISTVAL

```

```

180      I=IREG-1
181      10 I=I+1
182      IF(I.GT.30) GO TO 15
183      IF(CMD(I).NE.TSTVAL) GO TO 10
184      INDEX=I
185      RETURN
186  C
187      15 INDEX=-1
188      RETURN
189      END
190  C
191  > FUNCTION NBLANK(CMD, IBEG)
192  C
193  C      FUNCTION TO FIND THE FIRST NON-BLANK CHARACTER
194  C      IN CMD BEGINNING SEARCH AT IBEG
195  C
196      INTEGER CMD(30),BLNK
197      DATA BLNK/' '/
198      I=IREG-1
199      10 I=I+1
200      IF(I.GT.30) GO TO 15
201      IF(CMD(I).EQ.BLNK) GO TO 10
202      NBLANK=I
203      RETURN
204  C
205      15 NBLANK=-1
206      RETURN
207      END
208  C
209  > SUBROUTINE WINIT
210  C
211  C      ROUTINE TO INITIALIZE ALL ARRAYS AND VARIABLES IN COMMON
212  C
213  C      *****
214      DIMENSION AUNIT(7,210),TRNSMN(7,210),PMOUTH(7,210),TCOST(7,210,3),
215      C      DIFMTH(7,210,3),CPUREM(7,210,3),FINREM(3780),FINCST(3780)
216      INTEGER SRCE(7,210,281),ITYP(7,210),ISTAGE(7),KERANK(7,210,3),
217      C      FINSRC(3780,43),ISFIN(3780),KNDX(3780),NAMBSN(7,20),
218      C      KTRANK(7,210,3),NSRCE(7),BASIN(3780,20)
219      COMMON /WTRIO/ IN,IOU
220      COMMON /WTRINP/ AUNIT,TRNSMN,PMOUTH,TCOST,DIFMTH,CPUREM,NSRCE
221      COMMON /WTRIN2/ SRCE,ITYP,ISTAGE,KERANK,NAMBSN,BASIN
222      COMMON /WTROUT/ FINREM,FINCST,FINSRC,ISFIN,KNDX,KTRANK,BSENTL,NUM
223  C      *****
224      INTEGER ONE(3),TWO(3),THREE(5),FOUR(4),FIVE(4),SIX(3),SEVEN(5),BLNK
225      DATA ONE,TWO,THREE,FOUR,FIVE,SIX,SEVEN,BLNK
226      &'O','N','C','T','W','O','T','H','R','E','E','F','O','U','R','F','I','V','E','S','I','X','S','E','V','E','N','B','L','N','K'
227      &'I','V','F','S','I','X','S','F','V','E','N',' '/
228  C
229      IN=5
230      IOU=5
231      DO 10 I=1,7
232      DO 5 J=1,20
233      5 NAMBSN(I,J)=BLNK
234      10 CONTINUE
235      DO 15 I=3,7
236      J=1-2
237      IF(J.GT.3) GO TO 12
238      NAMBSN(1,I)=ONE(J)
239      NAMBSN(2,I)=TWO(J)

```

```

240      NAMBSN(6,1)=SIX(J)
241      IF(J.GT.4) GO TO 14
242      NAMBSN(4,1)=FOUR(J)
243      NAMBSN(5,1)=FIVE(J)
244      IF(J.EQ.3) NAMBSN(3,1)=THREE(J)
245      NAMBSN(7,1)=SEVEN(J)
246      15 CONTINUE
247      BSNTL=0.0
248      NUM=0
249      M=0
250      DO 20 I=1,7
251          ISTATE(I)=0
252          NSRCE(I)=0
253          DO 25 J=1,210
254              M=M+1
255              AUNIT(I,J)=0.0
256              TRNSMN(I,J)=0.0
257              PMOUTH(I,J)=0.0
258              ITYP(I,J)=0
259              DO 17 L=1,28
260                  17 SRCE(I,J,L)=BLNK
261                  DO 19 K=1,3
262                      TCOST(I,J,K)=0.0
263                      DIFMTH(I,J,K)=0.0
264                      CPUREM(I,J,K)=0.0
265                      KFRANK(I,J,K)=0
266                  19 KTRANK(I,J,K)=0
267                      FINREM(M)=0.0
268                      FINCST(M)=0.0
269                      KNDX(M)=0
270                      ISFIN(M)=0
271                      DO 22 L=1,43
272                          22 FINSRC(M,L)=BLNK
273                  25 CONTINUE
274              30 CONTINUE
275      C
276      RETURN
277      END
278      C
279      C
280      C SUBROUTINE BASRANK(NBASN)
281      C
282      C     ROUTINE TO RANK BASINS FOR COST-EFFECTIVENESS
283      C
284      C *****
285      C     DIMENSION AUNIT(7,210),TRNSMN(7,210),PMOUTH(7,210),TCOST(7,210,3),
286      C     & DIFMTH(7,210,3),CPUREM(7,210,3),FINREM(3780),FINCST(3780)
287      C     INTEGER SRCE(7,210,28),ITYP(7,210),ISTAGE(7),KFRANK(7,210,3),
288      C     & FINSRC(3780,43),ISFIN(3780),KNDX(3780),NAMBSN(7,20),
289      C     & KTRANK(7,210,3),NSRCE(7),BASIN(3780,20)
290      C     COMMON /WTRID/ IN,IOU
291      C     COMMON /WTRINP/ AUNIT,TRNSMN,PMOUTH,TCOST,DIFMTH,CPUREM,NSRCE
292      C     COMMON /WTRIN2/ SRCE,ITYP,ISTAGE,KFRANK,NAMBSN,BASIN
293      C     COMMON /WTROUT/ FINREM,FINCST,FINSRC,ISFIN,KNDX,KTRANK,BSNTL,NUM
294      C *****
295      C     DIMENSION CPUTMP(3780),KERTMP(3780)
296      C
297      DO 5 I=1,3780
298          CPUTMP(I)=0.0
299          5 KERTMP(I)=0

```

```

300      K = 0
301      DO 40 I=1,NBASN
302          IS=ISTAGE(I)
303          NS=NSPCE(I)
304          DO 30 J=1,IS
305              DO 20 I=1,NS
306                  IF(IITYP(L,I).EQ.1 .OR. AUNIT(L,I).EQ.3.0) GO TO 20
307                  K=K+1
308                  CPUTMP(K)=CPUREM(L,I,J)
309                  DO 10 I1=1,20
310                      BASIN(K,I1)=NAMBSN(L,I1)
311              20 CONTINUE
312          30 CONTINUE
313      40 CONTINUE
314      C      RANK USING SHELL SORT
315          IGAP=K/2
316      50 IF(IGAP.LE.0) GO TO 70
317          IIGAP=IGAP + 1
318          DO 65 I=IIGAP,K
319              J=I - IGAP
320      55 IF(J.LE.0) GO TO 65
321              JG=J + IGAP
322              IF(CPUTMP(J) .LE. CPUTMP(JG)) GO TO 60
323      C      EXCHANGE PLACES IN TEMPORARY DATA ARRAY
324              AK=CPUTMP(J)
325              CPUTMP(J)=CPUTMP(JG)
326              CPUTMP(JG)=AK
327      C      EXCHANGE PLACES OF INDEXES
328              IK=KNDX(J)
329              KNDX(J)=KNDX(JG)
330              KNDX(JG)=IK
331      60 J=J - IGAP
332              GO TO 55
333      65 CONTINUE
334          IGAP=IGAP/2
335          GO TO 50
336      70 CONTINUE
337      C      DONE WITH SORT
338          DO 80 I=1,K
339              M=KNDX(I)
340      80 KERTMP(M)=I
341          K2=0
342          DO 110 L=1,NBASN
343              IS=ISTAGE(L)
344              NS=NSRCE(L)
345              DO 100 J=1,IS
346                  DO 90 I=1,NS
347                      IF(IITYP(L,I).EQ.1 .OR. AUNIT(L,I).EQ.3.0) GO TO 90
348                      K2=K2+1
349                      KTRANK(L,I,J)=KERTMP(K2)
350      90 CONTINUE
351      100 CONTINUE
352      110 CONTINUE
353      NUM=K
354      RETURN
355      END
356      C
357      C
358      > SUBROUTINE BSNINP(JN,N,INDX)
359      C

```

```

360 C ROUTINE TO INPUT CONTROL STRATEGY DATA PER BASIN
361 C
362 C *****
363 DIMENSION AUNIT(7,210),TPNSMN(7,210),PMOUTH(7,210),TCOST(7,210,3),
364 C DIFMTH(7,210,3),CPUREM(7,210,3),FINREM(3780),FINCST(3780)
365 INTEGER SRCF(7,210,28),ITYP(7,210),ISTAGE(7),KERANK(7,210,3),
366 C FINSRC(3780,43),ISFIN(3780),KNDX(3780),NAMBSN(7,20),
367 C KTRANK(7,210,3),NSRCE(7),BASIN(3780,20)
368 COMMON /WTRIO/ IN,OUT
369 COMMON /WTRINP/ AUNIT,TRNSMN,PMOUTH,TCOST,DIFMTH,CPUREM,NSRCE
370 COMMON /WTRIN2/ SRCF,ITYP,ISTAGE,KERANK,NAMBSN,BASIN
371 COMMON /WTROUT/ FINREM,FINCST,FINSRC,ISFIN,KNDX,KTRANK,BSNTL,NUM
372 C *****
373 DATA ALPHA1,ALPHA2/'####','$$$$'/
374 C
375 READ(JN,5,ERR=800) NSRCE(N),IT1,IT2,IT3
376 5 FORMAT(4I5)
377 IFND=NSRCE(N)
378 IF(IT3.EQ.0) GO TO 15
379 DO 10 I=1,IT3
380 10 READ(JN,7) ALPHA
381 7 FORMAT(A4)
382 15 IF(IT1.GE.1) READ(JN,7) ALPHA
383 IF(IT1.EQ.2) READ(JN,7) ALPHA
384 IF(IT2.GE.1) READ(JN,7) ALPHA
385 IF(IT2.GE.2) READ(JN,7) ALPHA
386 IF(IT2.EQ.3) READ(JN,7) ALPHA
387 C
388 DO 30 I=1,IFND
389 READ(JN,20,ERR=810)(SRCF(N,I,J),J=1,28),ITYP(N,I),ALPHA,TEMP,AUNIT(N,I)
390 20 FORMAT(28A1,I3,A1,2F15.4)
391 IF(ITYP(N,I).EQ.2) AUNIT(N,I)=AUNIT(N,I)*1000.
392 30 READ(JN,7) ALPHA
393 C
394 40 READ(JN,7,END=780) ALPTST
395 IF(ALPTST.NE.ALPHA1) GO TO 40
396 READ(JN,5) IT1,IT2,IT3,ISTAGE(N)
397 IF(ISTAGE(N).EQ.0.OR.IT3.EQ.0) GO TO 830
398 45 READ(JN,7,END=780) ALPTST
399 IF(ALPTST.NE.ALPHA2) GO TO 45
400 READ(JN,50) NUM,TOTLOD,TOTMTH
401 50 FORMAT(15,2F15.4)
402 READ(JN,7) ALPHA
403 IFND2=ISTAGE(N)
404 DO 65 I=1,IFND
405 READ(JN,67) TRNSMN(N,I),PMOUTH(N,I)
406 DO 50 J=1,IFND2
407 60 READ(JN,68) PDIF,PUCOST,TCOST(N,I,J),DIFMTH(N,I,J),CPUREM(N,I,J),
408 C KERANK(N,I,J)
409 65 CONTINUE
410 67 FORMAT(2F15.4)
411 69 FORMAT(5F15.4,I5)
412 BSNTL=BSNTL+TOTLOD
413 70 READ(JN,72,END=75)(FINSRC(INDX,J),J=1,43),IALP,ISFIN(INDX),
414 C FINREM(INDX),FINCST(INDX)
415 KNDX(INDX)=INDX
416 INDX=INDX+1
417 GO TO 70
418 72 FORMAT(43A1,2I5,2F15.4)
419 C

```

```

420      800 WRITE(IOUT,800)
421      902 FORMAT('OERROR IN INPUT OF INITIAL VALUES OF WATERSHED DATA FILE')
422      GO TO 75
423      810 WRITE(IOUT,810)
424      812 FORMAT('OERROR IN INPUT OF SOURCE ID DATA')
425      GO TO 75
426      780 WRITE(IOUT,780)
427      782 FORMAT('OERROR -- CONTROL DATA NOT FOUND')
428      GO TO 75
429      820 WRITE(IOUT,820)
430      822 FORMAT('OERROR IN INPUT OF CONTROL DATA')
431      GO TO 75
432      830 WRITE(IOUT,830)
433      832 FORMAT('OZERO STAGES OF CONTROL IN WATERSHED DATA FILE')
434
435      C
436      75 CONTINUE
437      RETURN
438      END
439
440      C
441      C      SUBROUTINE BSNOUT(M,NBASN)
442      C
443      C      ROUTINE TO OUTPUT WORKSHEET SUMMARY OF BASIN RANKING
444      C
445      C      *****
446      C      DIMENSION AUNIT(7,210),TRNSMN(7,210),PMOUTH(7,210),TCOST(7,210,3),
447      C      &      DIFMTH(7,210,3),CPUREM(7,210,3),FINREM(3780),FINCST(3780)
448      C      INTEGER SRCE(7,210,28),ITYP(7,210),ISTAGE(7),KERANK(7,210,3),
449      C      &      FINSRC(3780,43),ISFIN(3780),KNDX(3780),NAMBSN(7,20),
450      C      &      KTRANK(7,210,3),NSRCE(7),BASIN(3780,20)
451      C      COMMON /WTRIO/ IN,IOUT
452      C      COMMON /WTRINP/ AUNIT,TRNSMN,PMOUTH,TCOST,DIFMTH,CPUREM,NSRCE
453      C      COMMON /WTRIN2/ SRCE,ITYP,ISTAGE,KERANK,NAMBSN,BASIN
454      C      COMMON /WTROUT/ FINREM,FINCST,FINSRC,ISFIN,KNDX,KTRANK,BSNTL,NUM
455      C      *****
456      C      DATA IPG,ITRM/'1','2'/
457      C      IFF=ITRM
458      C      IF(M.NE.IOUT) IFF=IPG
459      C      SUM1=0.0
460      C      SUM2=0.0
461      C      WRITE(M,10) IFF
462      C      10 FORMAT(1,56X,'SUMMARY OF PROGRAMS'/56X,'TOTAL POLLUTANT LOADS'//
463      C      &      61X,65(' '),66(' ')/62X,62X,'SUM OF'/80X,'LOAD',5X,'SUM OF LOAD',
464      C      &      613X,'COST OF REDUCTION'/78X,2('REDUCTION ','PERCENT',3X,
465      C      &      6'REDUCTION COSTS'/6X,'SOURCE',34X,'RANK',6X,'BASIN',10X,'STAGE',
466      C      &      62(3X,'(KG/YR) '), 'REDUCTION ($/YR)',5X,'($/YR)'/
467      C      &      61X,65(' '),66(' '))
468      C
469      C      DO 30 I=1,NUM
470      C      J=KNDX(I)
471      C      SUM1=SUM1+FINREM(J)
472      C      SUM2=SUM2+FINCST(J)
473      C      PCTRED=(SUM1/BSNTL)*100.0
474      C      WRITE(M,25) (FINSRC(J,K),K=1,43),I,(BASIN(J,K),K=1,20),ISFIN(J),
475      C      &      FINREM(J),SUM1,PCTRED,FINCST(J),SUM2
476      C      25 FORMAT(1X,43A1,15,2X,20A1,13,2F12.0,F10.1,F12.0,F11.0)
477      C      30 CONTINUE
478      C      WRITE(M,35)
479      C      35 FORMAT(1X,65(' '),66(' '))
480      C      RETURN
481      C      END

```

Program Output

The worksheet 8 that follows is the output from the sample terminal session that has been compiled for example purposes only.

SUMMARY OF PROGRAMS
TOTAL POLLUTANT LOADS

WORKSHEET 8

SOURCE	RANK	BASIN	STAGE	LOAD REDUCTION (K3/YR)	SUM OF LOAD REDUCTION (K3/YR)	PERCENT REDUCTION	COST OF REDUCTION (\$/YR)	SUM OF REDUCTION COSTS (\$/YR)
CROPLAND - UPPER SANDUSKY RIVER	1	SANDUSKY TWO STAGE	1	21065.	21065.	2.5	17849.	17849.
CROPLAND - MIDDLE SANDUSKY	2	SANDUSKY TWO STAGE	1	37752.	58818.	7.0	35253.	53192.
CROPLAND - TYNOCHTEE CREEK	3	SANDUSKY TWO STAGE	1	33279.	92097.	10.9	34346.	87448.
CROPLAND - BROKEN SWORD	4	SANDUSKY TWO STAGE	1	11733.	103830.	12.3	12167.	99616.
CROPLAND - LOWER SANDUSKY (SOIL 1)	5	SANDUSKY TWO STAGE	1	14101.	117931.	14.0	15831.	115446.
CROPLAND - ROCK CREEK	6	SANDUSKY TWO STAGE	1	22806.	140737.	16.7	26643.	142090.
CROPLAND - LOSS CREEK	7	SANDUSKY TWO STAGE	1	9694.	153431.	17.9	12340.	154429.
CROPLAND - LOWER SANDUSKY (SOIL 2)	8	SANDUSKY TWO STAGE	1	4869.	155299.	18.4	6500.	160929.
CROPLAND - HONEY CREEK	9	SANDUSKY TWO STAGE	1	16047.	171347.	23.3	22331.	183260.
CROPLAND - LOWER SANDUSKY (SOIL 2)	10	SANDUSKY ONE STAGE	1	3882.	175229.	23.8	6530.	189760.
MUNICIPAL - FREMONT CITY	11	SANDUSKY TWO STAGE	1	21145.	196373.	23.3	47352.	237112.
CROPLAND - WOLF CREEK	12	SANDUSKY TWO STAGE	1	9540.	205913.	24.5	22332.	259414.
MUNICIPAL - UPPER SANDUSKY CITY	13	SANDUSKY TWO STAGE	1	6219.	212132.	25.2	15000.	274414.
CROPLAND - UPPER SANDUSKY RIVER	14	SANDUSKY ONE STAGE	1	5923.	219355.	26.0	17849.	292263.
MUNICIPAL - BUCYRUS CITY	15	SANDUSKY TWO STAGE	1	10365.	229420.	27.2	32400.	324663.
MUNICIPAL - FREMONT CITY	16	SANDUSKY ONE STAGE	1	14096.	243517.	28.9	47352.	372015.
CROPLAND - LOSS CREEK	17	SANDUSKY ONE STAGE	1	3614.	247130.	29.4	12340.	384355.
MUNICIPAL - UPPER SANDUSKY CITY	18	SANDUSKY ONE STAGE	1	4146.	251276.	29.8	15030.	399355.
CROPLAND - BROKEN SWORD	19	SANDUSKY ONE STAGE	1	3221.	254498.	30.2	12167.	411522.
CROPLAND - HONEY CREEK	20	SANDUSKY ONE STAGE	1	5593.	260091.	30.9	22331.	433853.
MUNICIPAL - BUCYRUS CITY	21	SANDUSKY ONE STAGE	1	5910.	267001.	31.7	32400.	466253.
MUNICIPAL - TIFFIN CITY	22	SANDUSKY TWO STAGE	1	13267.	280268.	33.3	62400.	528653.
CROPLAND - LOWER SANDUSKY (SOIL 1)	23	SANDUSKY ONE STAGE	1	3109.	283376.	33.7	15831.	544483.
CROPLAND - ROCK CREEK	24	SANDUSKY ONE STAGE	1	4817.	288193.	34.2	26643.	571127.
CROPLAND - TYNOCHTEE CREEK	25	SANDUSKY ONE STAGE	1	5079.	293272.	34.8	34346.	605473.
CROPLAND - MIDDLE SANDUSKY	26	SANDUSKY ONE STAGE	1	5166.	298438.	35.4	35253.	640726.
MUNICIPAL - TIFFIN CITY	27	SANDUSKY ONE STAGE	1	8845.	307283.	36.5	62400.	703126.
CROPLAND - WOLF CREEK	28	SANDUSKY ONE STAGE	1	2418.	309709.	36.8	22332.	725428.
MUNICIPAL - FREMONT CITY	29	SANDUSKY TWO STAGE	2	3524.	313224.	37.2	71028.	796456.
MUNICIPAL - UPPER SANDUSKY CITY	30	SANDUSKY TWO STAGE	2	1037.	314261.	37.3	22530.	818956.
MUNICIPAL - BUCYRUS CITY	31	SANDUSKY TWO STAGE	2	1729.	315988.	37.5	48630.	867556.
MUNICIPAL - TIFFIN CITY	32	SANDUSKY TWO STAGE	2	2211.	318200.	37.8	93600.	961156.
CROPLAND - UPPER SANDUSKY RIVER	33	SANDUSKY TWO STAGE	2	10236.	328436.	39.3	82380.	1784956.
CROPLAND - MIDDLE SANDUSKY	34	SANDUSKY TWO STAGE	2	18876.	347312.	41.2	162708.	3412036.
CROPLAND - LOWER SANDUSKY (SOIL 1)	35	SANDUSKY TWO STAGE	2	7422.	354733.	42.1	730650.	4142686.
CROPLAND - ROCK CREEK	36	SANDUSKY TWO STAGE	2	12440.	367173.	43.6	1229699.	5372385.
CROPLAND - LOSS CREEK	37	SANDUSKY TWO STAGE	2	5739.	372911.	44.3	569520.	5941904.
CROPLAND - BROKEN SWORD	38	SANDUSKY TWO STAGE	2	5604.	378515.	45.3	561570.	6503474.
COMBINED - TIFFIN CITY	39	SANDUSKY TWO STAGE	2	2444.	380959.	45.2	256880.	6760353.
COMBINED - FREMONT CITY	40	SANDUSKY TWO STAGE	2	2891.	383850.	45.6	303810.	7064163.
CROPLAND - TYNOCHTEE CREEK	41	SANDUSKY TWO STAGE	2	14705.	398355.	47.3	1585199.	8649362.
COMBINED - BUCYRUS CITY	42	SANDUSKY TWO STAGE	2	1523.	400077.	47.5	177840.	8827201.
STORM - FREMONT CITY	43	SANDUSKY TWO STAGE	1	89.	401165.	47.5	13360.	8837560.
STORM - BUCYRUS CITY	44	SANDUSKY TWO STAGE	1	41.	403246.	47.5	9620.	8847180.
STORM - TIFFIN CITY	45	SANDUSKY TWO STAGE	1	439.	403684.	47.6	51800.	8898980.
COMBINED - TIFFIN CITY	46	SANDUSKY TWO STAGE	1	624.	401309.	47.7	76960.	8975939.
COMBINED - FREMONT CITY	47	SANDUSKY TWO STAGE	1	739.	402046.	47.7	91020.	9066959.
CROPLAND - LOWER SANDUSKY (SOIL 2)	48	SANDUSKY TWO STAGE	2	2329.	404374.	48.0	300000.	9366959.
CROPLAND - HONEY CREEK	49	SANDUSKY TWO STAGE	2	7906.	412280.	49.3	133650.	10397609.
COMBINED - BUCYRUS CITY	50	SANDUSKY TWO STAGE	1	399.	412668.	49.0	53280.	10450880.
COMBINED - TIFFIN CITY	51	SANDUSKY ONE STAGE	1	520.	413188.	49.1	76960.	10527847.
COMBINED - FREMONT CITY	52	SANDUSKY ONE STAGE	1	615.	413803.	49.1	91020.	10618867.

COMBINED - DUCYRUS CITY	53	SANDUSKY ONE STAGE	1	324.	414127.	49.2	532007	10672146.
CROPLAND - WOLF CREEK	54	SANDUSKY TWO STAGE	2	5620.	419749.	49.9	1029330.	11701476.
STORM - DUCYRUS CITY	55	SANDUSKY TWO STAGE	2	61.	419809.	49.9	11180.	11712656.
STORM - TIFFIN CITY	56	SANDUSKY TWO STAGE	2	328.	420137.	49.9	60200.	11772856.
STORM - FREMONT CITY	57	SANDUSKY TWO STAGE	2	66.	420232.	49.9	12040.	11784895.
STORM - FREMONT CITY	58	SANDUSKY ONE STAGE	1	35.	420237.	49.9	10360.	11795254.
STORM - DUCYRUS CITY	59	SANDUSKY ONE STAGE	1	33.	420270.	49.9	9620.	11804874.
STORM - TIFFIN CITY	60	SANDUSKY ONE STAGE	1	175.	420445.	49.9	51800.	11856674.
COMBINED - UPPER SANDUSKY CITY	61	SANDUSKY TWO STAGE	2	511.	420956.	50.0	214890.	12071563.
COMBINED - UPPER SANDUSKY CITY	62	SANDUSKY TWO STAGE	1	131.	421087.	50.0	64380.	12135942.
COMBINED - UPPER SANDUSKY CITY	63	SANDUSKY ONE STAGE	1	109.	421195.	50.0	64380.	12200321.