



NORTH DAKOTA
STATE UNIVERSITY

Research Report 100
February 1984

APR 27 1984

SERIALS DEPT.
LIBRARY

NORTH DAKOTA RESEARCH REPORT

Water Balance Irrigation Scheduling Based on Jensen-Haise Equation: Software for Apple II, II + , and IIE Computers

E.C. STEGMAN and D.A. COE*

*Professor and Student Computer Programmer, Agricultural Engineering Dept. North Dakota State University, Fargo, ND.



AGRICULTURAL EXPERIMENT STATION
NORTH DAKOTA STATE UNIVERSITY
FARGO, NORTH DAKOTA 58105

INTRODUCTION

Developments in applied methods for estimating crop water use (evapotranspiration) have made it possible to estimate root zone soil moisture deficits in irrigated fields with reasonable accuracy. A zero soil moisture deficit (SMD=0) implies that the root zone storage capacity (influenced by soil textures and rooting depths) is filled to "field capacity." Day-to-day crop water use produces a water deficit, whereas irrigations and rainfalls (when they occur) reduce the deficit. Irrigation scheduling implies that at some level of water deficit an irrigation system must be started and run until the prevailing deficit is reduced to lower levels if near maximum yield is the production goal.

The following material includes an algorithm for irrigation scheduling, source code with documentation (written in Applesoft Basic), and a users guide.

WATER BALANCE ALGORITHM

Water balance irrigation scheduling requires the estimation of prevailing root zone soil moisture levels. These levels are usually represented in the form of soil moisture deficits (SMD's). A balance equation can be written in the form:

$$\text{SMD}_i = \text{SMD}_{i-1} - P_i - \text{IR}_i + \text{ET}_i \quad (1)$$

Where SMD_i = root zone soil moisture deficit on day i (i.e., today); SMD_{i-1} = deficit on day $i-1$ (yesterday); P_i = effective (i.e., stored) precipitation on day i ; IR_i = net irrigation (i.e., gross amount minus evaporation and drift loss) on day i ; and ET_i = crop water use (evapotranspiration) on day i .

Applied usage of equation 1 begins in each season with initial measurements (on a chosen date) of prevailing soil moisture deficits. Usually these measurements are made with hand probe equipment and the use of guides (Lundstrom and Stegman, 1983) for judging how much water has been removed from the soil profile to the depth of expected root zone development. Due to soil-type variability, that is typically evident, it is necessary to key irrigation timing to the coarser textured soils in each field. Hence, the soil moisture balance in each field is usually computed for the soils that have the lower water holding capacities.

After the initial SMD is estimated this value is assigned to SMD_{i-1} in equation 1. The SMD on the following day (SMD_i) is estimated by subtracting today's effective precipitation (P_i) and today's net irrigation (IR_i). If rainfall plus irrigation exceeds the prevailing SMD_{i-1} , the excess is assumed to drain away quite rapidly. Thus, SMD_i is simply reduced to a value of zero on days when this condition occurs.

Crop water use is estimated (in this algorithm) by using the Jensen-Haise equation (1963) and a set of modifying coefficients. Potential evapotranspiration

is based on weather data; specifically, maximum and minimum daily air temperatures ($^{\circ}\text{F}$) and solar radiation (langleys).

In this algorithm, we used the unmodified Jensen-Haise equation of the form:

$$\text{ET}_p = (0.014T_m - 0.37)R_s / 1485.7 \quad (2)$$

Where, ET_p = potential evapotranspiration (in/day); T_m = daily maximum + minimum temperatures divided by two, ($^{\circ}\text{F}$); and R_s is divided by 1485.7 to convert solar radiation to equivalent evaporation units (in/day).

Daily crop ET is estimated as:

$$\text{ET} = K_c \cdot \text{ET}_p \quad (3)$$

Where, ET = estimate of daily evapotranspiration (in/day); and K_c = coefficient that modifies ET_p for effects of several factors. For the Jensen et. al (1971) scheduling model, K_c takes the form:

$$K_c = K_{co}K_a + K_s \quad (4)$$

Where, K_{co} = coefficient that modifies ET_p for plant growth stage and percent of land surface that is shaded by green cover; K_a = coefficient (ranging from zero to 1.0) that decreases ET as the root zone dries or SMD increases; and K_s = coefficient to increase crop ET when the soil surface is wet following rainfall and/or irrigation.

For the software in this report, K_{co} is based upon curves that were developed for North Dakota (Stegman et al. 1977). These curves are fitted by fourth order polynomials. Associated coefficients and their file names (CORN6, etc.) are given in Table 1. A program listing (CROP COEFFICIENTS) is given in Fig.

24. This program can be used to create new crop files or to change coefficients in existing files.

Table 1. Polynomial Coefficients (K_{co}) for crop curves. K_{co} is computed as a function of days post emergence (DPE)*

CORN6	ALFALFA6
C1 = .1814466119	C1 = .4507644773
C2 = 1.877271E-04	C2 = .0379602537
C3 = 7.004694E-04	C3 = -7.612063E-04
C4 = -9.3707E-06	C4 = 6.1893E-06
C5 = 3.12E-08	C5 = -1.76E-08
SUNFLR6	BARLEY6
C1 = -.1769272089	C1 = .1546199918
C2 = .0569635108	C2 = .0169164464
C2 = -8.756348E-04	C3 = 1.0181232E-03
C4 = 6.2417E-06	C4 = -2.59128E-05
C5 = -2.48E-08	C5 = 1.398E-08
WHEAT6	SBEAN6
C1 = .1449023485	C1 = .1747183800
C2 = .0146752708	C2 = 3.1039866E-03
C3 = 8.156267E-04	C3 = 5.743438E-04
C4 = -1.84894E-05	C4 = -7.7158E-06
C5 = 8.94E-08	C5 = 2.33E-08
PINTO6	
C1 = .2728081942	
C2 = -6.179509E-04	
C3 = 9.073189E-04	
C4 = -1.36184E-05	
C5 = 4.55E-08	

$$*K_{co} = C1 + C2 * DPE + C3 * DPE^2 + C4 * DPE^3 + C5 * DPE^4$$

For K_a computations we used the correction for limiting soil moisture conditions as given by:

$$K_a = 1 \text{ if } AW > 50\% \quad (5)$$

$$K_a = \frac{AW}{50} \text{ if } AW < 50\% \quad (6)$$

Where AW = percent available water remaining (100 when root zone is at field capacity).

The adjustment for wet surface soil conditions is limited to 3 days after rainfall or irrigation in periods when green ground cover is incomplete. (i.e., $K_{co} < 0.9$). Thus,

$$K_s = (0.9 - K_{co}) 0.8 \text{ on day of rain or irrigation} \quad (7)$$

$$K_s = (0.9 - K_{co}) 0.5 \text{ on first day after} \quad (8)$$

$$K_s = (0.9 - K_{co}) 0.3 \text{ on second day after} \quad (9)$$

$K_s = 0$ thereafter, until next rainfall or irrigation, or whenever $K_{co} > 0.9$.

USER'S GUIDE

The source code for this program was written to accommodate a maximum of 15 fields. If more fields are scheduled, it can be simply done by copying the program and associated files to another disk¹ and another 15 fields can be scheduled.

The program includes crop curve coefficients, (K_{co}) for seven crops. The crops that can be scheduled are identified by number in Fig. 3. Also, refer to Table 1 for the file names that are assigned to each crop.

To begin¹, the operator types **LOAD** Jensen-Haise. After the program is loaded into computer memory, the typing of **RUN** displays the main menu as in Fig. 1. The program is menu driven with four options. Typically the user first selects option 3 to initialize the **FIELDS11** file for each field. Thereafter, the user would usually select option 1 to compute weekly updates of the **SMD** level for each field.

Detailed explanation and example user responses are given in the following sections:

*****IRRIGATION SCHEDULING*****
BY THE JENSEN-HAISE METHOD
ALGORITHM BY E. STEGMAN
PROGRAMER D. COE
NDSU AG ENG. DEPT.

1 WEEKLY UPDATE PROGRAM
2 CHANGE SMD OR UPDATE DATE
3 INITIALIZE FIELD DATA FILES
4 EXIT THIS PROGRAM

SELECT ONE— >3

Fig. 1. The main menu

OPTION 3

Option 3 allows the user to initialize data files for each field that is scheduled. The first data input (Fig. 2) under this option is the total number of fields that are currently ready for initialization.

INITIALIZE FIELD DATA FILES

ENTER HIGHEST NEW FIELD#— >2

Fig. 2. The first input under menu choice 3.

¹This discussion assumes that the user has developed or has been provided with the scheduling program and associated files on a floppy disk. The software described herein is available from the senior author.

If, for example, fields 1 and 2 are to be initialized at the beginning of the irrigation season then a 2 would be entered as the highest field number. If later in the season fields 3-5 are initialized, then 5 would be entered. Note, data file dimensioning allows this program to schedule up to 15 different fields. More fields can be scheduled by simply making more copies of the program, therefore scheduling 15 fields per disk.

The program under Option 3 asks for added information about each field. The first item is the field #. At the first initialization we begin with #1 and then increment by one until the highest field number equals the number that was entered in response to the question in Fig. 2. Figure 3 shows the inputs that are needed to initialize each field. Also, see Table 3 for the structure of the FIELDS11 file.

```

INITIALIZE FIELD DATA FILES

ENTER FIELD #
(0 TO ESCAPE)—> 1

CROP NUMBERS
-----
1 = SUNFLOWERS
2 = CORN
3 = WHEAT
4 = PINTO BEANS
5 = SOYBEANS
6 = ALFALFA
7 = BARLEY

FIELD #1
ENTER CROP #—> 2
ENTER CROP EMERGENCE DATE (MONTH, DAY)—> 5, 21
ENTER THE PRESENT DATE (MONTH, DAY)—> 6, 1
ENTER IRRIGATION EFFICIENCY, IN PERCENT—> 85
ENTER TOTAL SOIL MOISTURE HOLDING CAP., IN INCHES—> 4
ENTER PRESENT SMD IN INCHES—> 1.8
ENTER 30% DEPLETION LEVEL IN INCHES—> 1.2
ENTER 50% DEPLETION LEVEL IN INCHES—> 2
ENTER IRRIGATIONS TO DATE IN INCHES—> 0
ENTER RAINFALL AMOUNT TO DATE IN INCHES—> 0
  
```

Fig. 3. Inputs that initialize the FIELDS11 file for each field.

Table 2. Glossary of Terms used in Computer Program

able	Description	Unit
A	Loop Counter	
AB	Truncated Crop Coefficient, KCO	Decimal
AKA	Coefficient That Adjusts ET for Effect of Root Zone Soil Moisture	Decimal
ASW\$	Answer, Yes or No	
AW%	Integer Representation for KCO	
C	Array Containing Polynomial Coefficients for KCO Computations	
DAE	Days Post Emergence	
DM	Days After May 1st	
DX	Number of Days in the Update Interval	
E	Actual ET as Influenced by Ground Cover and Soil Moisture Level	Inches

EFF	Irrigation Efficiency	Percent
EG%	Integer Representation of ET1	
EP	Potential Evapotranspiration	Inches
ER	Menu Option	
ET1	ET Component Associated with Soil Surface Wetness	Inches
F	Array For Field Data (see table 1)	
HH	Loop Counter	
I	Index for "F" Array	
ID	Loop Counter	
IR	Array for Irrigation Amounts	Inches
IRR	Irrigation Amount	Inches
IT	Total Irrigation to Date	Inches
J	Points to Start of Data for Each Field in Array "F"	
K	Highest Element of Array "F"	
KCO	Crop Coefficient Due to Degree of Green Ground Clover	Decimal
L(4)-L(9)	Length of Months, 4-9 in days	
M\$	Name of Crop Being Scheduled	
M1,D1	Emergence Date (also, old cut date for alfalfa)	
M2,D2	Starting Date of the Update Interval; M2 = Month, D2 = day	
M3,D3	Ending Date of Update Interval	
M4,D4	Current Date	
M5,D5	New Cut Date for Alfalfa	
M7,D7	Temporary Storage of Emergence Date	
MN	Minimum Temperature for Given Day	°F
MN	Menu Number	
MT	Maximum Temperature for Given Day	°F
N\$	File for Crop Curve Coefficients	
N	Total Number of Fields Scheduled	
NF	Number Given to Particular Fields	
NL, NH	Lowest and Highest Field Number	
P	Loop Counter	
P1	Array for Storing Rainfall & Irrigation	Inches
RA	Temporary Storage of Rainfall Amounts	Inches
RF	Rainfall Amount for Given Day	Inches
RT	Total Rainfall to Date	Inches
SA	Mean Daily Solar Radiation; Based on Climatic Data	Langley
SD%	Integer Representation of SMD	Inches
SL	Solar Radiation for Given Day	Langley
SMD	Root Zone Soil Moisture Deficit	Inches
SCMT	Root Zone Available Water Holding Capacity	Inches
SR	Daily Solar Radiation Converted to Inches of Water Equivalent	Inches
TA	Mean Daily Temperature	°F
TF	Mean Daily Air Temperature; Based on Climatic Data	°F
TKC	Coefficient Combining Effects of Ground Cover and Soil Moisture Level	Decimal
TW%	Integer Representation of KCO	
V	Lowest Element of Array "F"	

Table 3. Structure of "FIELDS11" File

Record No.	Variable
F(l)*	Field Number (NF)
F(l + 1)	Crop Number **
F(l + 2)	Month of Crop Emergence, M1
F(l + 3)	Day of Crop Emergence, D1
F(l + 4)	Irrigation Efficiency, EFF
F(l + 5)	Total Available Soil Moisture Holding Capacity, SCMT
F(l + 6)	Soil Moisture Deficit as of Last Update, SMD
F(l + 7)	Total Irrigation to Date, IT
F(l + 8)	Total Rainfall to Date, IR
F(l + 9)	Last Update Month, M2
F(l + 10)	Last Update Day, D2
F(l + 11)	30% Depletion Level
F(l + 12)	50% Depletion Level

*l is calculated as ((NF-1)*13) + 1

**Crop Numbers are:

Sunflowers	= 1
Corn	= 2
Wheat	= 3
Pinto Beans	= 4
Soybeans	= 5
Alfalfa	= 6
Barley	= 7

After the field number has been entered the program asks for the crop #. The number that corresponds to each crop that can be scheduled is displayed on the screen. In Fig. 3, crop number "2" was entered. A 2 indicates the crop is corn. The next input is the emergence date. This date is entered numerically as month,day. The date in Fig. 3 is May 21 or 5,21. The initialization or present date is entered in the same format. Irrigation efficiency is entered in percent units. Total soil moisture holding capacity should usually indicate the plant-available storage capacity for the expected root zone depth in the coarsest textured soil type in each field. This input has "inch" units. The present soil moisture deficit (SMD) should represent a current measurement or estimate that corresponds with the present date. A zero SMD indicates that the profile water content is at "field capacity." The 30 percent and 50 percent depletion levels (Fig. 3) represent approximate irrigation timing criteria; i.e., irrigations should begin when SMD falls between these levels of depletion. The "irrigation and rainfall to date" lines are usually initialized to zero. The program, thereafter, will keep running totals with each update.

After the data for a given field have been entered, the program prints the information back to the screen as shown in Fig. 4. If the data are correct and the user answers "Y," the program asks for the next field number.

```

FIELD#->1
CROP#->2
EMERGENCE DATE->5/21
PRESENT DATE->6/1
EFFICIENCY OF IRR->85
SOIL MOISTURE CAP->4
PRESENT SMD->1.8
30% DEPL.->1.2
50% DEPL.->2
IR TO DATE->0
RAIN TO DATE->0
ARE THESE VALUES CORRECT (Y/N)
?N
    
```

Fig. 4. Example of data as printed to CRT screen for checking.

If there is an error and the user enters "N" the program displays the data again as shown in Fig. 5 and asks which input is in error. In Fig. 5 the operator entered "7." The program then displays the current value and asks for the new value to be entered.

```

1 FIELD #->1
2 CROP #->2
3 EMERGENCE DATE->5/21
4 PRESENT DATE->6/1
5 EFFICIENCY OF IRR->85
6 SOIL MOISTURE CAP->4
7 PRESENT SMD->1.8
8 30% DEPL.->1.2
9 50% DEPL.->2
10 IR TO DATE->0
11 RAIN TO DATE->0
12 ALL CORRECTIONS ARE MADE

WHICH OF THE INPUTS IS IN ERROR 7

PRESENT SMD IS 1.8
ENTER NEW SMD 1.9
    
```

Fig. 5. Example of display for data correction

When a value is entered the information in Fig. 5 is re-displayed with the correction made. Again the program asks for the input in error. If all data are correct the operator should enter "12." This will move the program onto the next step.

Because the Jensen-Haise method includes estimates of the evaporation component that occurs in response to a recently wetted soil, the program initialization asks for the sum of irrigation and rainfall amounts (if any) for three consecutive days preceding the initialization date. Figure 6 shows the CRT display that asks for these data. If the operator

answers the first question with "N," the program enters a "0" for the data. If the operator answers "Y," then the program asks the user to enter the sum of the irrigation and rainfall amounts. These amounts should be entered starting with three days ago. In Fig. 3 the present date is 6,1. Thus, the first amount that should be entered is the sum for 5,30. Then enter that for 5,31 and the final entry is that

```

WAS THERE ANY RAINFALL OR IRRIGATION APPLIED IN
THE PAST THREE DAYS (Y/N) Y

ENTER THE SUM OF AMOUNTS OF IRRIGATION AND RAINFALL,
STARTING WITH THREE DAYS AGO.
?.5
?0
?.25

```

Fig. 6. Display from computer dealing with rainfall and/or irrigation amounts for the past three days.

for 6,1. Next, the program will return to the "Enter Field #" line in Fig. 3. The operator continues to initialize the data files for each field. Note: In Fig. 3 the present date must be the same for all fields that are initialized at this time.

After all fields (up to the number indicated in Fig. 2) have been initialized, the program returns to "Enter Field #" (see Fig. 3). The user then enters "99" to save all data on disk. After "99" is entered, it is necessary to enter (Fig. 7) the range in field numbers that are being initialized.

```

HIGHEST NEW FIELD IS INITIALIZED
(ENTER 99 TO SAVE DATA ON DISK)—>99
ALL DATA FOR FIELDS BETWEEN THE HIGH AND LOW FIELD #'S
WILL BE SAVED
ENTER THE HIGHEST NEW FIELD # TO BE SAVED—>12
ENTER THE BEGINNING FIELD # FOR THIS INITIALIZATION
OF FIELDS—>1
WRITING.....

```

Fig. 7. Field data to be stored on disk.

The program asks for the highest and lowest field numbers to be saved. In Fig. 7 (as an example) the highest number is "12." The lowest field number to be saved is "1." Thus, fields 1-12 are initialized. If there is a data gap between the high and low field # values, the program will not indicate an error but all data for noninitialized fields are set to zero. If this happens, it will be necessary to rerun Option 3 of the main menu (Fig. 1). After data initialization, the program returns to the main menu.

If the operator is adding more fields to the scheduling program at a later date, the lowest field number (Fig. 7) would **not** be "1." **Example:** Fields 1-12 were initialized at an earlier date. The operator has just initialized fields 13-15 by consecutively completing the questions in Fig. 3. Then the highest field number (Fig. 7) to be saved would be 15 and the lowest number would be 13. Also, note that the response in Fig. 2 should have been 15.

OPTION 1:

When the operator selects option 1 in the main menu, as shown in Fig. 1, the program enters routines that estimate the SMD update. For this update, the program first reads the disk file containing the information about field #1. Upon completion of the file reading, the program displays the last update and asks for the starting date of the new update interval (Fig. 8). When the starting date has been entered, the program checks to make sure that it is one day later than the last update date. If it is not, the

```

THE LAST UPDATE WAS AT 6/1
ENTER STARTING DATE OF THE UPDATE INTERVAL (M,D) 6,2
ENTER ENDING DATE OF THE UPDATE INTERVAL (M,D) 6,8

```

Fig. 8. The display of the last update date and the entering of the starting and ending date of the update interval.

program prints an error message and asks that the date be re-entered. Next, the program asks for the ending date of the interval. The update interval can be up to seven days long. If the interval length is over seven days, the program prints an error message and asks that the ending date be re-entered.

The program then asks for the weather data for each day of the update interval as shown in Fig. 9. Weather data for each day are entered in the format: rainfall (in inches), maximum temperature, minimum temperature (both in °F), and solar radiation (in langley's).

Each number is separated by a comma. After entering all of the weather data, the program displays the data back to the screen for error checking by the operator as shown in Fig. 10. If the operator enters "N" to the question, "Are these values correct?" the program then adds a number column as shown in Fig. 11. The user then enters the number that corresponds to the day with incorrect weather data. Then the operator re-enters the data for that day. The program re-displays the data as in Fig. 11 to allow a further correction of errors. If there are no more errors then a "0" is entered.

ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR 6/2
(RF,MT,MINT,SR) ?0,72,49,438

ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR 6/3
(RF,MT,MINT,SR) ?0,74,50,488

ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR 6/4
(RF,MT,MINT,SR) ?0,69,47,399

ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR 6/5
(RF,MT,MINT,SR) ?0,77,52,532

ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR 6/6
(RF,MT,MINT,SR) ?0,78,54,543

ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR 6/7
(RF,MT,MINT,SR) ?0.25,69,62,321

ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR 6/8
(RF,MT,MINT,SR) ?0,73,48,565

Fig. 9. The entering of the weather data for a 7 day update interval.

DATE	RAIN FALL	MAX TEMP	MIN TEMP	SOLAR RADIATION
6/2	0	72	49	438
6/3	0	74	50	488
6/4	0	69	47	399
6/5	0	77	52	532
6/6	0	78	54	543
6/7	0.25	69	62	321
6/8	0	73	48	565

ARE THESE VALUES CORRECT? (Y/N)N

Fig. 10. Example display of weather data for error checking by operator.

NOTE: This program assumes that the same weather data applies to all fields, therefore these data are entered only once for each update.

The program then asks if there were any irrigations on Field #1 (Fig. 12). If the operator answers "N" as in Fig. 12, the program moves on to estimate SMD as of the

WAS THERE ANY IRRIGATION ON FIELD #1 (Y/N)
?N

Fig. 12. Program asking about irrigation on Field #1 (Y/N)

NO.	DATE	RAIN-FALL	MAX TEMP	MIN TEMP	SOLAR RADIATION
1	6/2	0	72	49	438
2	6/3	0	74	50	488
3	6/4	0	69	47	399
4	6/5	0	77	52	532
5	6/6	0	78	54	543
6	6/7	0.25	69	52	321
7	6/8	0	73	48	565

ENTER THE NUMBER THAT CORRESPONDS TO THE DAY WITH INCORRECT WEATHER DATA.
(ENTER 0 TO END)2

ENTER THE RAINFALL, MAX TEMP, MIN TEMP, AND SOLAR RADIATION FOR THE DAY
IN ERROR ?.5,71,50,488.

Fig. 11. Display of weather data for error correction

end of the update interval. Example results of this calculation are shown in Fig. 13.

RESULTS FOR INTERVAL ENDING 6/8 FIELD #1:CORN	
THE LAST UPDATE 6/1	
	INCHES
SOIL MOISTURE TO DATE	1.55
IRRIGATION AMOUNT TO DATE	0.00
RAINFALL AMOUNT TO DATE	.75
SOIL MOISTURE HOLDING CAP	4.00
30% DEPLETION LEVEL	1.20
50% DEPLETION LEVEL	2.00
CROP COEFFICIENT—>.41	
SURFACE WETNESS COEFFICIENT—>.05	
DAYS AFTER EMERG.—>18	
ARE THESE RESULTS OK? (N TO REENTER) Y	

Fig. 13. Example results of the scheduling program for Field #1

The program output displays the current SMD estimate and the current status of several additional terms. The current SMD estimate can be compared with the soil moisture holding capacity (as previously initialized) and the associated 30 and 50 percent depletion levels. For corn the 50 percent depletion level defines a relatively safe depletion that can be allowed in the early vegetative (if root zone advances into moisture) and later seed-fill period. Irrigations should begin before root zone water depletion exceeds 30-40 percent in the tassel to blister kernel period. The "crop coefficient" and "days after emergence" data provide information that can be used to determine whether the crop is developing at an average rate. This is done by comparing the observed growth stage with the predicted stage (see crop curves in Stegman et al., 1976). If such a comparison indicates otherwise it may be necessary before the next SMD update to re-initialize the

emergence date. This adjustment can be used to create a better agreement between the observed and the predicted growth stage. If results are okay (Fig. 13) the program asks the operator if he/she would like a forecast of SMD for the next seven days (Fig. 14). If a "Y" answer is given, the program displays its prediction of the average temperature for the next seven days and then asks if this is an acceptable prediction. If on the basis of an advanced weather forecast the operator

(See Fig. 14 — bottom of page)

answers "N," the program asks for the entry of a better prediction. The same is true for the solar radiation term. When this is completed, the program estimates the advance water use rates and the resulting SMD for each day of the forecast period. This information is displayed as shown in Fig. 15. NOTE: This forecast assumes no rainfall will occur in the forecast period.

SMD FORECAST STARTING 6/8		
DAY	SMD	DAILY WATER USE
1	1.64	.09
2	1.73	.09
3	1.83	.10
4	1.93	.10
5	2.04	.11
6	2.15	.11
7	2.27	.12
30% DEPLETION LEVEL = 1.2		
50% DEPLETION LEVEL = 2		
PRESS RETURN TO CONTINUE.....		

Fig. 15. Results from a seven day forecast

When the operator presses "return" to continue, the program begins the update process for field #2 as shown in Fig. 16. If the operator answers "Y," the

WRITING.....
DO YOU WISH TO FORECAST SMD FOR THE NEXT 7 DAYS FOR FIELD #1?
?Y
IS 64.01 A GOOD PREDICTION FOR THE AVE TEMP FOR THE NEXT 7 DAYS (Y/N)
?Y
IS 594 A GOOD PREDICTION FOR SOLAR RADIATION FOR THE NEXT 7 DAYS (Y/N)
?Y

Fig. 14. The program asking the operator whether he/she would like to forecast SMD. Also, given are program predictions for average temperature and solar radiation.

program asks for the entry of irrigation dates and amounts as shown in Fig. 17. The operator

WAS THERE ANY IRRIGATION ON FIELD #2 (Y/N)
 ?Y

Fig. 16. Were irrigation amounts applied to the next field?

enters the date and irrigation amounts in the format "month, day, amount." The amount is in inches. When all data (falling within the seven day update period) have been entered for the given field, the operator enters "99,99,99" to exit this routine (Fig. 17).

(See Fig. 17 — bottom of page)

The program then lists the dates and irrigation amounts for the update interval (Fig. 18). If these values are not correct("N") the program returns to the line in

DATE	IRR. AMOUNT
6/2	0
6/3	0
6/4	1.00
6/5	0
6/6	0
6/7	0
6/8	0
ARE THESE VALUES CORRECT? (Y/N) Y	

Fig. 18. Display of irrigation dates and amounts.

Fig. 16. Hence, all irrigation data in the update interval must be re-entered. The program then estimates the SMD for the second field.

RESULTS FOR INTERVAL ENDING 6/8 FIELD #2: SUNFLOWERS	
THE LAST UPDATE 6/1	
	INCHES
SOIL WATER DEFICIT TO DATE	0.14
IRRIGATION AMOUNT TO DATE	1.00
RAINFALL AMOUNT TO DATE	.75
SOIL MOISTURE HOLDING CAP.	4.50
30% DEPLETION LEVEL	1.35
50% DEPLETION LEVEL	2.25
CROP COEFFICIENT— >.50	
SOIL WETTING FACTOR— >0	
DAYS AFTER EMERG.— >17	
ARE THESE RESULTS OK (N TO REENTER) Y	

Fig. 19. The results for field 2 update.

After listing these results, the program asks if they are okay. If the operator enters "N" the program asks if the error is in the irrigation (I) or rainfall (R) amounts. If an "I" is entered the program returns to the line in Fig. 16. After irrigations are re-entered the results (like Fig. 19) are recalculated. If an "R" had been entered the program would return to the routine for weather data correction (Fig. 11). **Note:** If the operator makes changes in weather data after the SMD's for some fields have been estimated, the output for the earlier fields would likely be in error. (Also, see appended comment #4). The earlier fields are not automatically rescheduled. Rather, the program proceeds to ask for irrigation amounts and recalculates the SMD for the field in question.

Following each listing of SMD updates (Fig. 13 and 19) and associated output, the operator is given the option to request a SMD forecast. The program continues this scheduling process until all fields (in consecutive sequence) have been updated. Then it returns to the main menu.

ENTER THE DATE OF IRRIGATION AND AMOUNT OF IRR(M,D,IRR) (ENTER 99, 99,99 TO EXIT) 6,4,1.0

ENTER THE DATE OF IRRIGATION AND AMOUNT OF IRR(M,D,IRR) (ENTER 99, 99,99 TO EXIT) 99,99,99

Fig. 17. Entry of irrigation dates and amounts for field being scheduled

OPTION 2

Option 2 in the main menu (Fig. 1) allows the operator to examine the SMD estimate as of the last update. This option allows the user to adjust SMD's to values that are based on in-the-field measurements. Figure 20 illustrates the program display when option 2 is selected. The operator is asked to enter the appropriate field number to be examined. The program then displays the information as shown in Fig. 21. To make a change the operator enters "Y." The program then displays the message given in Fig. 21

```
EXAMINE SMD, TOTAL IR AND RF; CHANGE SMD
ENTER FIELD # (0 TO ESCAPE)—>2
```

Fig. 20. The first step in examining SMD

which asks for the new SMD and update date.

```
THE LAST UPDATE WAS ON 6/8
PRESENT SMD=.14 INCHES

DO YOU WISH TO CHANGE SMD OR LAST UPDATE
(Y/N) Y
```

Fig. 21. The information on field #2 from menu choice #2.

```
ENTER NEW SMD IN INCHES—>0
ENTER UPDATE DATE(M,D)—>6/8
```

Fig. 22. Option 2 display when changes are made to SMD and/or the update date.

If the operator wishes to only change one of the values, then the old value is re-entered for the other. If the update date is changed the program asks that the sum of rainfall and irrigation amounts be re-initialized for the past three days as in Fig. 6. Also, **note**, that if an update date is changed, then all dates (for every field) must be changed. The last update (Fig. 8) must be common for all fields.

OPTION 4

The final menu choice will exit the program.

APPENDED COMMENTS

(1) This program assumes that temperatures and solar radiation data can be suitably obtained such that a given data set applies to all fields that are initialized on a given disk. To accommodate the scheduling of widely scattered fields it may be necessary to use different weather sites. The user must then decide which fields are best served by each particular weather site and then group his fields accordingly on each disk.

(2) Option 3 (initialization of fields) was written to allow the addition of more fields over time up to a total of 15 on a particular disk. The program does not permit an easy method for deletion of fields. For example, if the irrigation season has ended for a "wheat" field, the simplest approach is to continue updating this field as though the wheat is still growing. The alternate approach would be to use option 3 to reinitialize all remaining fields (dropping out appropriate fields) and renumbering them consecutively. Again, all initial or present SMD input must be common to the same initial or present date.

(3) The program, we feel, is reasonably error trapped. Problems can be kept to a minimum by careful entry of required inputs. If an input error causes the program to **hang** or **break** at some line number it is usually necessary to punch RESET. Then enter RUN to get back to the main menu. At this point the user must decide whether he can again continue into the appropriate option or whether re-initializations would be required.

(4) The program assumes that the same weather data (Fig. 9) applies to all fields that are initialized on a given disk. If rainfall is quite variable from one field to another it may be best to enter rainfalls as "zero amounts" in the weather file (Fig. 9) and then to enter rainfalls as though they were irrigations (Fig. 17). Thus, rainfalls and irrigation amounts would be field specific.

SOFTWARE LISTING AND DOCUMENTATION

This program is written in Applesoft Basic. Figure 23 gives an abbreviated flow chart for each of the menu options. These charts identify the principal subroutines and branches. A listing of variables and associated descriptions is given in Table 2. In the following discussion the program listing has been broken into segments for documentation purposes. The first segment, lines 10-40, initializes the Disk Operating System (DOS) commands for Applesoft Basic.

```
10 REM IR SCHEDULE--BY JENSEN-HAISE METHOD
20 D$=CHR$(4)
30 PRINT D$;"NOMON C,I,O"
40 DIM F(220)
```

Line 20 defines D\$ as control-D (ASCII character 4). This command facilitates read and write operations to disk files. Line 30 suppresses the display of DOS commands to a video monitor. Line 40 dimensions an array "F" that belongs to a "scratchpad" file named "FIELDS11." Its structure is illustrated in Table 1.

```

50 REM PRINT MENU
60 HOME
70 INVERSE
80 HTAB 5
90 PRINT "
100 VTAB 2: HTAB 5
110 PRINT "**** IRRIGATION SCHEDULING ****"
120 VTAB 3: HTAB 5
130 PRINT "
140 VTAB 4: HTAB 5
150 PRINT "BY JENSEN-HAISE METHOD"
160 VTAB 5: HTAB 5
170 PRINT "
180 NORMAL
190 PRINT
200 PRINT TAB( 9)"ALGORITHM BY E. STEGMAN"
210 PRINT : PRINT TAB( 11)"PROGRAMER D. COE"
220 PRINT : PRINT TAB( 10)"NDSU AG. ENG. DEPT."

```

Lines 50-220 print the program header. **HOME** clears the screen and moves the cursor to the upper left screen position. **INVERSE** sets the computer to print black letters on a white background. **VTAB 2** moves the cursor vertically to the line number that follows the command. A colon separates multiple instructions on one program line. **HTAB** moves the cursor horizontally to the column number indicated after the command. **NORMAL** returns the print mode back to white on black. The **TAB** command works the same as **HTAB** except it must be in a print statement.

```

230 VTAB 12: PRINT "....."
240 VTAB 13:
250 PRINT "!"
260 PRINT "!"          1 WEEKLY UPDATE PROGRAM      "!"
270 PRINT "!"
280 PRINT "!"          2 CHANGE SMD OR UPDATE FILES  "!"
290 PRINT "!"
300 PRINT "!"          3 INITIALIZE FIELD DATA FILES "!"
310 PRINT "!"
320 PRINT "!"          4 EXIT THIS PROGRAM           "!"
330 PRINT "!"
340 PRINT "....."
350 HTAB 12: VTAB 24
360 INPUT "SELECT ONE— >";MN

```

Lines 230-360 print the main menu and ask the operator for a selection, MN.

```

370 HOME
380 IF MN < 1 OR MN > 4 THEN 60
390 ON MN GOSUB 420,3680,1850,410
400 PRINT "END OF PROGRAM": END

```

Lines 370-390 check the selection that was made and determine whether it is a correct entry. **ON MN GOSUB** branches to subroutines that begin at the specified line numbers. The particular line number is determined by the **MN** selection. Line 380 causes the program to go to line 60 which will print the header and main menu again. The **END OF PROGRAM** statement occurs at line 400.

The main program for computing current SMD estimates begins at line 430.

```

420 REM MAIN PROGRAM
430 REM L IS THE LENGTH OF THE MONTHS OF APRIL-SEPT
440 L(4)=30:L(5)=31:L(6)=30
450 L(7)=31:L(8)=31:L(9)=30

```

Lines 430-450 establish an array "L" containing the number of days in each month from April (month 4) thru September (month 9).

```

460 NL=1:NH=0
470 GOSUB 2570: REM READ THE FIELD DATA

```

The set values for NL and NH in line 460 allow the subroutine at 2570 to read-in the data for all fields that can be scheduled following a particular initialization of the "F" array in the FIELDS11 file.

```

480 HOME
490 PRINT
500 PRINT "THE LAST UPDATE WAS AT" F(10)"/"F(11)
510 PRINT
520 INPUT "ENTER STARTING DATE OF THE UPDATE INTERVAL
(M,D)";M2,D2
530 IF F(10)=M2 and F(11)+1=D2 THEN 560
540 IF F(11)=L(F(10)) AND D2=1 THEN 560
550 HOME : PRINT : PRINT "ERROR IN NEW DATE. DATE NOT
ONE DAY LATER THEN LAST UPDATE.": GOTO 490
560 PRINT
570 INPUT "ENTER ENDING DATE OF THE UPDATE INTERVAL
(M,D)";M3,D3
580 IF M2=M3 AND D3 < D2 + 7 THEN 620
590 IF M2 + 1 = M3 AND (L(M2) - D2) + D3 <= 7 THEN 610
600 PRINT : PRINT "DATE ERROR INTERVAL OVER 7 DAYS LONG
PLEASE REENTER": GOTO 560
610 DX=(L(M2)-D2)+D3: GOTO 630
620 DX=D3-D2

```

Line 500 prints to a video monitor the ending date (month/day) of the last update interval. Thus, in line 520 the operator enters the starting date of the new update interval, which must be one day later than the end of the last update date. An error message with return to line 490 occurs if the update date is not one day later. The operator enters the ending date (line 570) of the new update date interval, which is checked to make sure that the new interval is not over seven days long. If too long, an error message occurs with a return to line 560 for re-entry of an ending date that falls within 7 days of the last ending date. Lines 610-620 determine interval length (DX) in days.

The weather data are printed (lines 690-920) on the monitor so the operator can check it. Lines 690-780 print the heading for each column of data. Lines 810-860 are used when a change of month occurs during the update interval and lines 870-910 apply when the data fall within the same month.

```
930 PRINT : INPUT "ARE THESE VALUES CORRECT? (Y/N)";ASW$
940 IF ASW$="N" THEN GOSUB 5780
```

```
630 HOME
640 FOR ID=0 TO DX
650 IF L(M2)>=D2+ID THEN 670
660 PRINT : PRINT "ENTER RAINFALL, MAX TEMP, MIN TEMP, AND SOLAR RADIATION FOR "M3"/"D2-L(M2)+ID"
(RF,MT,MINT,SR)": INPUT RF(ID),MT(ID),MNT(ID),SL(ID): GOTO 680
670 PRINT : PRINT "ENTER RAINFALL, MAX TEMP, MIN TEMP AND SOLAR RADIATION FOR "M2"/"D2+ID"
(RF,MT,MINT,SR)": INPUT RF(ID),MT(ID),MNT(ID),SL(ID)
680 NEXT ID
```

The operator enters (lines 630-680) daily weather data for the update interval. Each entry is in the sequence: rainfall, max. temperature, min. temperature, and solar radiation. The respective units are inches, °F, and langleys. Line 660 applies when the update starts in one month and ends in another. Line 670 is used when the update interval is all within the same month.

If, upon inspection the weather data are incorrect, a "N" input from the keyboard in line 940 calls a subroutine (at line 5780) that allows changes to be made without re-entry of all data.

```
690 HOME : PRINT "DATE";
700 HTAB 8: PRINT "RAIN.";
710 HTAB 16: PRINT "MAX";
720 HTAB 22: PRINT "MIN";
730 HTAB 29: PRINT "SOLAR";
740 HTAB 8: PRINT "FALL";
750 HTAB 16: PRINT "TEMP";
760 HTAB 22: PRINT "TEMP";
770 HTAB 29: PRINT "RADIATION"
780 PRINT "....."
790 FOR ID=0 TO DX
800 IF L(M2)>=D2+ID THEN GOTO 870
810 PRINT M3"/"D2-L(M2)+ID;
820 HTAB 10: PRINT RF(ID);
830 HTAB 18: PRINT MT(ID);
840 HTAB 24: PRINT MNT(ID);
850 HTAB 32: PRINT SL(ID)
860 GOTO 920
870 PRINT M2"/"D2+ID;
880 HTAB 10: PRINT RF(ID);
890 HTAB 18: PRINT MT(ID);
900 HTAB 24: PRINT MNT(ID);
910 HTAB 32: PRINT SL(ID)
920 NEXT ID
```

```
950 FOR J=1 TO N
960 I=((J-1)*13)+1
970 HOME : PRINT : PRINT
980 PRINT "WAS THERE ANY IRRIGATION ON FIELD # "F(I)" (Y/N)"
990 INPUT ASW$
1000 IF ASW$="Y" THEN GOSUB 2940
1010 GOSUB 2710: REM FIND NUMBER OF DAYS AFTER EMERGE.
1020 X=FRE (0)
```

A loop is begun in line 950 to update the root zone water balance estimates (SMD) for all fields (1 thru N). This loop ends at line 1530. Line 960 sets up a pointer for the "F" array that identifies thirteen data records for each field (see table 1). Next, (line 980) the program asks whether irrigations were applied to field F(I). If yes (Y) a GOSUB takes place to line 2940. Following the entry of irrigations (2940-3230), a GOSUB is executed to line 2710. This subroutine (2710-3820) computes the number of days since crop emergence for the crop in field F(I).

```
1030 REM FIND THE CROP THAT IS PLANTED IN THIS FIELD
1040 IF F(I+1)=1 THEN N$="SUNFLR6":M$="SUNFLOWERS"
1050 IF F(I+1)=2 THEN N$="CORN6":M$="CORN"
1060 IF F(I+1)=3 THEN N$="WHEAT6":M$="WHEAT"
1070 IF F(I+1)=4 THEN N$="PINTO6":M$="PINTO BEANS"
1080 IF F(I+1)=5 THEN N$="SBEAN6":M$="SOYBEANS"
1090 IF F(I+1)=6 THEN M$="ALFALFA": GOSUB 1580
1100 IF F(I+1)=7 THEN N$="BARLEY6":M$="BARLEY"
1110 IF M$="ALFALFA" THEN 1150
```

Lines 1030-1110 determine what crop was planted in field F(I). Then a subroutine (**GOSUB 2840**) determines crop curve coefficients for each day (ID = 0 TO DX) in the update interval. If the crop is alfalfa, a **GOSUB** to line 1580 determines whether a new cutting date has occurred. Subsequently, days after cutting (if within 20 days) or days after May 1 are used to compute the crop coefficient for alfalfa. **GOSUB 4050** (line 1150) encompassing lines 4050-4930, computes the SMD updates.

```

1120 FOR ID = 0 TO DX
1130 GOSUB 2840: REM FIND CROP CURVE DATA
1140 NEXT ID
1150 GOSUB 4050: REM CALCULATE SMD
1160 HOME
1170 PRINT
1180 PRINT
1190 PRINT "RESULTS FOR INTERVAL ENDING" M3"/"D3
1200 PRINT "FIELD #" F(I)"/"M$
1210 PRINT
1220 PRINT "THE LAST UPDATE" F(I + 9)"/"F(I + 10)
1230 PRINT "....."
1240 PRINT TAB( 30)"INCHES"
1250 PRINT TAB( 30)"....."
1280 PRINT
1270 PRINT "SOIL WATER DEFICIT TO DATE" TAB( 32)SMD
1280 PRINT
1290 PRINT "IRRIGATION AMOUNT TO DATE" TAB( 32)IT
1300 PRINT "RAINFALL AMOUNT TO DATE" TAB( 32)RT
1310 PRINT "SOIL MOISTURE HOLDING CAP." TAB( 32)F(I + 5)
1320 PRINT
1330 PRINT "30% DEPLETION LEVEL" TAB( 32)F(I + 11)
1340 PRINT "50% DEPLETION LEVEL" TAB( 32)F(I + 12)
1350 PRINT "CROP COEFFICIENT—>"AB
1360 PRINT "SURFACE WETNESS COEFF.—>"ET1
1370 PRINT "DAYS AFTER EMERG.—>"DAE + DX
1380 PRINT "....."
1390 PRINT

```

Lines 1160-1390 output results for the end of the update interval. Output to a printer requires insertion of lines 1175 and 1395. Line 1175 reads PR#1 and line 1395 reads PR#0.

```

1400 INPUT "ARE THESE RESULTS OK? (N TO REENTER)";ASW$
1410 IF ASW$ = "N" THEN GOTO 1550

```

Lines 1400-1410 ask the operator if the results are correct. A no (N) branches to line 1550 where errors in either rainfall or irrigation inputs can be corrected. The SMD update routine is then re-run.

```

1420 NL = F(I):NH = F(I)
1430 F(I + 9) = M3
1440 F(I + 10) = D3
1450 F(I + 6) = SMD:F(I + 7) = IT:F(I + 8) = RT
1460 GOSUB 2440

```

When results are accepted (in line 1410), lines 1420-1460 update the FIELDS11 file with the data needed for the next program run. Line 1420 sets NL and NH = F(I) so that the data for one field are saved. **GOSUB 2440** saves these data on disk.

```

1470 PRINT "DO YOU WISH TO FORECAST SMD FOR
THE NEXT 7 DAYS FOR FIELD #"F(I)""?"

```

```

1480 INPUT ASW$
1490 IF ASW$ = "Y" THEN GOSUB 3240
1500 FOR ID = 0 TO DX
1510 IR(ID) = 0
1520 NEXT ID
1530 NEXT J
1535 IF J > N THEN 60
1540 RETURN

```

Subsequent to an SMD update the operator is offered an opportunity (line 1470) to forecast the probable SMD development for the next seven days. A yes (Y) in 1490 calls (**GOSUB 3240**) the subroutine at lines 3240-3670 to compute the SMD forecasts. The return to lines 1400-1420 zeros the IR (Irrigation Amounts) array before "J" is incremented (1540) to begin the SMD update for the next field. The main program for SMD updates ends at line 1540. The **RETURN** here returns program execution back to the main menu.

The subsequent program components are generally subroutines that serve the main program.

```

1550 INPUT "WAS ERROR IN RAINFALL OR IRRIGATION
AMOUNTS (R,I)";ASW$
1560 IF ASW$ = "R" THEN GOSUB 5780
1570 GOTO 970

```

Lines 1550-1570 allow the user to make changes in the irrigation data or rainfall data if an error is noted when the information is printed to the screen but before these data are saved on the disk. Branching is based on the answer to this question.

```

1580 REM ROUTINE FOR ALFALFA CROP CURVE
1590 REM THE OLD CUTTING DATE IS M1,D1;NEW CUTTING DATE, IF ANY, IS M5,D5; CURRENT DATE IS M4,D4.
1600 HOME
1610 VTAB 3
1620 PRINT "THE LAST CUTTING DATE WAS: "F(I + 2)"/"F(I + 3)
1630 PRINT : INPUT "DO YOU WISH TO ENTER NEW CUTTING DATE?";ASW$
1640 IF ASW$ = "N" THEN M5 = F(I + 2);D5 = F(I + 3): GOTO 1670
1650 PRINT : INPUT "ENTER THE CUTTING DATE (M,D)";M5,D5
1660 IF M5 = 5 and D5 = 1 THEN 1770
1670 FOR ID = 0 TO DX
1680 IF L(M2) >= D2 + ID THEN M4 = M2;D4 = D2 + ID: GOTO 1700
1690 M4 = M3;D4 = D2 - L(M2) + ID
1700 IF M4 < M5 THEN 1770
1710 IF M4 = M5 AND D4 < D5 THEN 1770
1720 F(I + 2) = M5;F(I + 3) = D5
1730 GOSUB 2710
1740 IF DAE + ID > 20 THEN 1770
1750 KCO(ID) = 0.5 + (0.5*(DAE + ID) / 20)
1760 GOTO 1820
1770 F(I + 2) = 5;F(I + 3) = 1
1780 GOSUB 2710
1790 N$ = "ALFALFA6"
1800 GOSUB 2840
1810 F(I + 2) = M5;F(I + 3) = D5
1820 NEXT ID
1830 N$ = "ALFALFA6"
1840 RETURN

```

Lines 1580-1840 encompass the subroutine that computes the crop curve data for alfalfa. Lines 1620-1630 print out the last cutting date and ask if the user would like to enter a new cutting date. If the answer is "NO," line 1640 sets the new cutting date equal to the old cutting date (i.e., no change). Line 1680 checks to see if the end of the current month has been reached. If so, line 1690 increases the variable M4 to the next month. Lines 1700-1710 compare the current date with the new cutting date. If it is earlier, a branch is made to lines 1770-1820 to determine the coefficient (KCO) for today's date. If the current date is later than the new cutting date, line 1730 branches to the subroutine at line 2170 to determine the number of days since the last cutting date. Line 1740 checks to see if the current date exceeds the last cutting date by 20 days. If so, the program branches to line 1770 and the appropriate subroutine at lines 4700 and 2830 to compute the current "crop coefficient."

If the cutting date and present date are less than 20 days apart, then the crop curve is based on a straight line function as given in line 1750.

The subroutine for menu option 3 (INITIALIZE FIELD DATA FILES) begins at line 1850. Lines 1850-2430 initialize the "F" array in the FIELDS11 file for each field to be scheduled. Lines 1880-1890 ask the operator to enter the total number of fields to be

```

1850 REM SUBROUTINE TO INITIALIZE FIELD DATA FILES
1860 PRINT "INITIALIZE FIELD DATA FILES"
1870 NF = 0
1880 INPUT "ENTER HIGHEST NEW FIELD # - >";N
1890 IF N < 1 OR N > 15 THEN 1880
1900 HOME
1910 PRINT "INITIALIZE FIELD DATA FILES"
1920 PRINT
1921 IF NF < N GOTO 1930
1922 NF = N
1923 PRINT "HIGHEST NEW FIELD IS INITIALIZED"
1924 INPUT "(ENTER 99 TO SAVE DATA ON DISK) - >";NF
1925 GOTO 1950
1930 PRINT "ENTER FIELD #,": INPUT "(0 TO ESCAPE) - >";NF
1940 IF NF < 1 OR NF > 15 AND NF <> 99 THEN RETURN
1950 IF NF = 99 THEN 2380
1960 I = ((NF - 1) * 13) + 1
1970 F(I) = NF
1980 PRINT

```

scheduled at the time that the program is initialized. This program is written to schedule up to 15 fields. However, more fields can be scheduled by simply making more copies of this program on separate disks. Line 1890 checks to see that this limit is not exceeded. Line 1930 asks the operator to enter the number of fields that are to be initialized at this time. Line 1940 again checks to see that the field number is within the 15 limit and whether the data are to be saved or if the user wishes to escape back to the main menu. Line 1960 computes a "pointer" for establishing 13 records in the FIELDS11 file (see table 1) for each scheduled field.

```

1990 PRINT "CROP NUMBERS"
2000 PRINT "....."
2010 PRINT "1 = SUNFLOWERS"
2020 PRINT "2 = CORN"
2030 PRINT "3 = WHEAT"
2040 PRINT "4 = PINTO BEANS"
2050 PRINT "5 = SOYBEANS"
2060 PRINT "6 = ALFALFA"
2070 PRINT "7 = BARLEY"
2080 PRINT
2090 PRINT "FIELD #" NF
2100 PRINT : INPUT "ENTER CROP #—>";F(I + 1)
2110 IF F(I + 1) = 6 THEN PRINT : INPUT "ENTER ALFALFA CUT DATE (M,D) OR 5,1 TO START—>";F(I + 2),F(I + 3): GOTO 2130
2120 PRINT : INPUT "ENTER CROP EMERGENCE DATE (MONTH,DAY)—>";F(I + 2), F(I + 3)
2130 PRINT : INPUT "ENTER THE PRESENT DATE (MONTH,DAY)—>";F(I + 9),F(I + 10)
2140 PRINT : INPUT "ENTER IRRIGATION EFFICIENCY, IN PERCENT—>";F(I + 4)
2150 PRINT : INPUT "ENTER TOTAL SOIL MOISTURE HOLDING CAP., IN INCHES—>";F(I + 5)
2160 PRINT : INPUT "ENTER PRESENT LEVEL SMD IN INCHES—>";F(I + 6)
2170 PRINT : INPUT "ENTER 30% DEPLETION LEVEL IN INCHES—>";F(I + 11)
2180 PRINT : INPUT "ENTER 50% DEPLETION LEVEL IN INCHES—>";F(I + 12)
2190 PRINT : INPUT "ENTER IRRIGATIONS TO DATE IN INCHES—>";F(I + 7)
2200 PRINT : INPUT "ENTER RAINFALL AMOUNT TO DATE IN INCHES—>";F(I + 8)

```

Lines 1990-2080 assign specific numbers to each crop. Crop number is entered in position F(I + 1) in line 2100. Line 2110 checks to see if the crop is alfalfa. If it is, it asks for the last cutting date or May 1st to be entered. Lines 2120-2200 ask for additional data to be entered such as crop emergence date, present date, irrigation efficiency, total root zone available water holding capacity, present SMD estimate, 30% and 50% depletion levels, irrigations to date, and rainfall to date. The rainfall and irrigation amounts would typically be summed from May 1.

Lines 2210-2330 echo back these data for checking of errors. If these data are correct ("Y" response) the program branches (GOSUB 4940) to 4940 for input of rainfall and irrigation amounts (if any) for the preceding three days (explained later). If operator response is no ("N") in line 2350, a branch to a subroutine at 5160 makes possible the correction of any one or more inputs without total re-entry of all others. After inputs for each field have been entered, checked, and corrected, this routine branches back to 1900 until all data have been entered. A "99" is entered to branch to line 2380. The operator then enters the highest field number (NH)

```

2210 REM ECHO RESULTS FOR CHECK
2220 PRINT : PRINT "FIELD #—>"F(I)
2230 PRINT "CROP #—>"F(I + 1)
2240 PRINT "EMERGENCE DATE—>"F(I + 2)"/"F(I + 3)
2250 PRINT "PRESENT DATE—> "F(I + 9)"/"F(I + 10)
2260 PRINT "EFFICIENCY OF IRR—>"F(I + 4)
2270 PRINT "SOIL MOISTURE CAP—>"F(I + 5)
2280 PRINT "PRESENT SMD—>"F(I + 6)
2290 PRINT "30% DEPL.—>"F(I + 11)
2300 PRINT "50% DEPL.—>"F(I + 12)
2310 PRINT "IR TO DATE—>"F(I + 7)
2320 PRINT "RAIN TO DATE—>"F(I + 8)
2330 PRINT
2340 PRINT TAB( 10)"ARE THESE VALUES
CORRECT (Y/N)": INPUT ASW$
2350 IF ASW$ = "N" THEN GOSUB 5160
2360 GOSUB 4940
2370 GOTO 1900

```

```

2380 REM WRITE TO DISK
2390 PRINT : PRINT "ALL DATA FOR FIELDS BETWEEN
THE HIGH AND LOW FIELD #'S WILL BE SAVED"
2400 PRINT : INPUT "ENTER HIGHEST FIELD # TO BE
SAVED—>";NH
2410 PRINT : INPUT "ENTER LOWEST FIELD # TO BE
SAVED—>";NL
2420 GOSUB 2440
2430 RETURN

```

and the lowest number (NL) to be saved. **NOTE:** If five fields were initialized at the start of the irrigation scheduling season, NL would equal 1 and NH = 5. If an additional 5 fields were initialized later, NL would equal 6 and NH = 10. Thus, the NL and NH values represent the range in fields that are initialized on a specific date. The reason for this is so the operator

does not have to reinitialize all data to add more fields to the scheduling program later. However, all fields must be initialized in consecutive order and the present SMD values must reflect a common date.

```

2440 REM SUBROUTINE TO WRITE DATA TO DISK
2450 V = ((NL-1)*13) + 1
2460 K = ((NH-1)*13) + 13
2470 VTAB 24: PRINT TAB( 10)"WRITING..."
2480 PRINT D$;"OPEN FIELDS11,L10"
2490 FOR P = V TO K
2500 PRINT D$;"WRITE FIELDS11,R";P
2510 PRINT F(P)
2520 NEXT P
2530 PRINT D$;"WRITE FIELDS11,R210"
2540 PRINT N
2550 PRINT D$;"CLOSE FIELDS11"
2560 RETURN

```

Lines 2440-2560 define the subroutine that initializes data to a disk file. Lines 2450-2460 set the high and low end of the "F" array in the FIELDS11 file. **Note:** If only one field is scheduled, NL = NH = 1 and 13 records would be written to the disk file. Line 2480 opens the FIELDS11 file with record length of 10. Line 2490 sets up a loop to write the contents of the "F" array to this disk file. Line 2500 writes the value of F(P) to the FIELDS11 file. The R indicates this is a random-access file. Lines 2530-2540 write the total number (N) of fields (N is defined in line 1880) being scheduled to position 210 in the FIELDS11 file. Therefore, should the dimension of the "F" array (line 40) be increased to accommodate more fields, the position for N would also need revision. Line 2550 closes the FIELDS11 file. **RETURN** at 2560 branches back to **RETURN** at 2430 which in turn branches back to line 400 and back to the main menu (260-320).

```

2570 REM SUBROUTINE TO OUTPUT FIELD DATA
FROM DISK
2580 REM IF THE DATA FOR ONE FIELD IS TO BE
READ THEN NL = NH = FIELD#
2590 V = ((NL-1)*13) + 1
2600 K = ((NH-1)*13) + 13
2610 PRINT D$;"OPEN FIELDS11,L10"
2620 PRINT D$;"READ FIELDS11,R210"
2630 INPUT N
2640 IF K = 0 THEN K = ((N-1)*13) + 13
2650 FOR P = V TO K
2660 PRINT D$;"READ FIELDS11,R";P
2670 INPUT F(P)
2680 NEXT P
2690 PRINT D$;"CLOSE FIELDS11"
2700 RETURN

```

The branch to the subroutine at 2570 originates at line 470. This routine provides data to the video from which the operator begins his next scheduling update. Lines 2590-2600 determine the number of records that will be read. Line 2600 opens the FIELDS11 disk file. Line 2610 indicates that the value for N will be read from record number 210 (see subroutine at 1850). If NH=0 (see line 460) then K=0. This indicates that all data for the "F" array are to be read. Therefore, K in line 2640 is set equal to ((N-1)*13) + 13. The return at 2700 branches back to 480.

```

2710 REM FIND THE NUMBER OF DAYS AFTER
EMERGE.
2720 M1 = F(I + 2):D1 = F(I + 3)
2730 REM M2 IS THE STARTING MONTH OF THE
INTERVAL
2740 REM D2 IS THE STARTING DAY OF THE INTERVAL
2750 IF M1 = M2 THEN DAE = D2-D1: GOTO 2830: REM
DAE IS DAYS AFTER EMERGE.
2760 IF M2-M1 = 1 THEN DAE = D2 + L(M1)-D1: GOTO 2830
2770 DAE = 0
2780 FOR K = M1 + 1 TO M2-1
2790 DAE = DAE + L(K)
2800 NEXT K
2810 DAE = D2 + L(M1)-D1 + DAE
2820 DAE = DAE-1
2830 RETURN

```

The branch to line 2710 occurs at line 1010. Lines 2710-2830 define the subroutine for finding the days after emergence for each crop. The emergence date (month,day) is M1, D1. M2, D2 defines the beginning date of the update interval. If the month of emergence and month of update are the same, DAE (days after emergence) equals D2-D1. If M2 and M1 are one month apart then DAE equals the starting date of update interval plus the number of days since emergence in the month of emergence (line 2760). If the first two cases are incorrect, the program counts up the days in each month between emergence date and present month and then adds the days to the present date in the present month.

```

2840 REM FIND THE CROP CURVE DATA
2850 DE = DAE + ID
2860 PRINT D$;"OPEN";N$";L15"
2870 FOR A = 1 TO 5
2880 PRINT D$;"READ";N$";R"A
2890 INPUT C(A)
2900 NEXT A
2910 PRINT D$;"CLOSE";N$
2920 KCO(ID) = C(1) + C(2)*DE + C(3)*(DE ^ 2) + C(4)*(DE ^ 3)
+ C(5)*(DE ^ 4)
2930 RETURN

```


The branch to line 2840 occurs at line 1880. Lines 2840-2930 define the subroutine that determines crop coefficient values for each day of the update interval. The particular file names for each crop are indicated in lines 1040-1110.

```

2940 REM SUB TO INPUT IRRIGATION AMOUNTS
2950 FOR ID = 0 TO DX
2960 IR(ID) = 0
2970 NEXT ID
2980 HOME
2990 PRINT : PRINT
3000 PRINT : INPUT "ENTER THE DATE OF IRRIGATION
AND AMOUNT OF IRR (M,D,IRR) (ENTER 99,99,999
to EXIT)"; M4,D4,IRR
3010 IF M2 = M3 AND M2 = M4 THEN 3060
3020 IF M2 = M4 THEN 3080
3030 IF M3 = M4 THEN 3100
3040 IF M4 = 99 THEN 3120
3050 GOTO 3110
3060 IF D2 <= D4 <= D3 THEN IR(D4-D2) = IRR:
GOTO 2990
3070 GOTO 3110
3080 IF D2 <= D4 <= L(M2) THEN IR(D4-D2) = IRR:
GOTO 2990
3090 GOTO 3110
3100 IF D4 <= D3 THEN IR(DX-(D3-D4)) = IRR: GOTO 2990
3110 PRINT "ERROR IN THE DATE OF IRRIGATION,
PLEASE REENTER": PRINT : GOTO 2990
3120 HOME
3130 PRINT "DATE" TAB( 10)"IRR. AMOUNT"
3140 FOR ID = 0 TO DX
3150 IF L(M2) >= D2 + ID THEN 3180
3160 HTAB 4: PRINT M3"/"D2-L(M2) + ID;
3170 HTAB 15: PRINT IR(ID): GOTO 3200
3180 HTAB 4: PRINT M2"/"D2 + ID;
3190 HTAB 15: PRINT IR(ID)
3200 NEXT ID
3210 PRINT : INPUT "ARE THESE VALUES CORRECT?
(Y/N)";ASW$
3220 IF ASW$ = "N" THEN 2950
3230 RETURN

```

The GOSUB call for irrigation input occurs at line 1000. The associated subroutine begins at line 2940. Lines 3000-3110 ask for input of each irrigation date, (M,D) and irrigation amount (IRR). A check is made to determine whether each irrigation date falls within the update interval. If so, each irrigation amount is properly placed in the array for irrigation amounts. If the month is input as 99, the lines 3120-3200 output the irrigation amounts and dates must be re-entered.

```

3240 REM TO FORECAST THE SMD
3250 M7 = F(I + 2):D7 = F(I + 3):F(I + 2) = 5:F(I + 3) = 1
3260 GOSUB 2710: REM THESE CHANGES ARE MADE
TO FIND THE NUMBER OF DAYS AFTER MAY FIRST
3270 F(I + 2) = M7:F(I + 3) = D7:DM = DAE + 3 + ID
3280 GOSUB 2710
3290 DAE = DAE + ID

```

The branch to line 3240 originates at line 1490 if the operator decides to compute forecasted SMD's for the next seven days. Line 3250 assigns month and day of crop emergence to temporary storage at M7 and D7. A branch is then made to the subroutine at 2710 to find the days accumulated since May 1. Then DM equals the days from May 1 to the middle of the forecast interval. This variable is subsequently used to predict average daily temperature and average daily solar radiation in the next seven days of the forecast period.

Lines 3300-3390 estimate daily solar radiation and daily average temperature from polynomial relationships that fit long term weather data for eastern North Dakota. The predictions for solar radiation and average temperature are output to the screen for check by the operator. At this point the operator may keep this prediction or input his/her own prediction of these weather parameters. These inputs can come from actual advance weather forecasts by the U.S. Weather Bureau or other forecasting agencies.

```

3300 TF = 51.661453 + 0.138765156*DM + .006914265*DM ^ 2 - 0.000086250*DM ^ 3 + .00000023*DM ^ 4
3310 SA = 538.828613 - 1.19905948*DM + 0.109132409*DM ^ 2 - 0.001387924*DM ^ 3 + 0.000004382*DM ^ 4
3320 TF% = TF*100.0 + .5:TF = TF%/100.0:SA% = SA + .5: SA = SA%
3330 HOME : PRINT "IS "TF" A GOOD PREDICTION FOR THE AVE TEMP FOR THE NEXT 7 DAYS (Y/N)": INPUT ASW$
3340 IF ASW$ = "Y" THEN 3360
3350 PRINT : INPUT "ENTER YOUR PREDICTION FOR THE AVE TEMP FOR THE NEXT 7 DAYS";TF
3360 PRINT : PRINT "IS "SA" A GOOD PREDICTION FOR SOLAR RADIATION FOR THE NEXT 7 DAYS. (Y/N)": INPUT ASW$
3370 IF ASW$ = "Y" THEN 3390
3380 PRINT : INPUT "ENTER YOUR PREDICTION FOR SOLAR RADIATION FOR THE NEXT 7 DAYS ";SA
3390 HOME

```

```

3400 FOR HH = 0 TO DX
3410 RA(HH) = RF(HH):AC(HH) = MT(HH):AD(HH) =
      MNT(HH):AE(HH) = SL(HH)
3420 NEXT HH

```

Lines 3400-3420 save the current weather data in temporary storage during the forecast procedure.

```

3430 PRINT
3440 PRINT TAB( 5)"SMD FORECAST STARTING
      "M3"/"D3
3450 PRINT "....."
3460 PRINT "DAY" TAB( 10)"SMD" TAB( 17)"DAILY
      WATER USE"
3470 PRINT "....." TAB( 10)"....." TAB( 17)"....."
3480 PRINT

```

Lines 3440-3480 print the header for the forecast output (SMD) of the program.

```

3490 FOR ID = 0 TO 6
3500 MT(ID) = TF:MNT(ID) = TF:IR(ID) = 0:RF(ID) =
      0:SL(ID) = SA
3510 GOSUB 2840
3520 GOSUB 4330
3530 E% = E*100.00 + 0.5:E = E% / 100
3540 PRINT ""ID + 1;
3550 HTAB 10: PRINT SMD;
3560 HTAB 23: PRINT E
3570 NEXT ID

```

Line 3490 increments time for subsequent use in the subroutine at 2840 to calculate advance crop curve coefficients. Line 3500 sets max. and min. temperature and solar radiation to predicted average values for the entire forecast period. Rainfall (RF) and irrigation amounts (IR) are assumed to be zero in the forecast period. The forecast SMD's are computed in the subroutine at line 4330. Line 3530 truncates the forecasted evapotranspiration (E) values. Lines 3540-3560 print the forecast data.

```

3580 PRINT
3590 PRINT "30% DEPLETION LEVEL = "F(I + 11)
3600 PRINT "50% DEPLETION LEVEL = "F(I + 12)
3610 PRINT
3620 VTAB 24
3630 INPUT "PRESS RETURN TO CONTINUE...";ASW$
3640 FOR HH = 0 TO DX
3650 RF(HH) = RA(HH):MT(HH) = AC(HH):MNT(HH) =
      AD(HH):SL(HH) = AE(HH)
3660 NEXT HH
3670 RETURN

```

Lines 3580-3600 print the 30% and 50% allowable depletion levels and then allow a pause for user review of these data. Lines 3640-3670 set the current weather data back to its original storage areas. If the user wishes to have the forecast data printed to the printer, the lines 3415 and 3605 should be inserted at the appropriate locations.

```

3415 PRINT D$ "PR#1"
3605 PRINT D$ "PR#0"

```

```

3680 REM SUBROUTINE TO CHANGE SMD AND
      DISPLAY TOTAL IR AND RF
3681 PRINT D$;"OPEN FIELDS11,L10"
3682 PRINT D$;"READ FIELDS11,R210"
3683 INPUT N
3684 PRINT D$;"CLOSE FIELDS11"
3690 HOME
3700 PRINT "EXAMINE SMD, TOTAL IR AND RF;
      CHANGE SMD"
3710 PRINT : INPUT "ENTER FIELD #
      (0 TO ESCAPE)—>";NF
3720 IF NF < 1 OR NF > N THEN RETURN
3730 J = ((NF-1)*13) + 1
3740 PRINT D$;"OPEN FIELDS11,L10"
3750 FOR I = J + 6 TO J + 10
3760 PRINT D$;"READ FIELDS11,R";I
3770 INPUT F(I)
3780 NEXT I
3790 PRINT D$;"CLOSE FIELDS11"
3800 M1 = F(J + 9): D1 = F(J + 10)
3810 HOME : PRINT
3820 PRINT "THE LAST UPDATE WAS AT
      "F(J + 9)"/"F(J + 10)
3830 PRINT "PRESENT SMD = "F(J + 6)"INCHES"
3840 PRINT
3890 INPUT "DO YOU WISH TO CHANGE SMD OR
      LAST UPDATE DATE ? (Y/N)";ASW$
3900 IF ASW$ = "N" THEN 3690
3910 PRINT "IF YOU WISH ONLY TO CHANGE ONE
      ENTER OLD VALUE FOR OTHER."
3920 PRINT
3930 INPUT "ENTER NEW SMD IN
      INCHES—> ";F(J + 6)
3940 INPUT "ENTER UPDATE DATE (M,D)—> ";
      F(J + 9),F(J + 10)

```

Continued on page 19

Continued from page 18

```
3950 PRINT
3960 PRINT D$;"OPEN FIELDS11,L10"
3970 FOR I = J + 6 TO J + 10
3980 PRINT D$;"WRITE FIELDS11,R";I
3990 PRINT F(I)
4000 NEXT I
4010 PRINT D$;"CLOSE FIELDS11"
4020 IF M1 = F(J + 9) AND D1 = F(J + 10) THEN 4040
4030 GOSUB 4940
4040 GOTO 3690
```

Option 2 of the main menu (line 280: CHANGE SMD OR UPDATE DATE) accesses subroutine beginning 3680. This subroutine allows the user to check and/change the most recent SMD estimate to an updated SMD value that is based upon actual field measurements. Lines 3710-3720 ask for the field number that is to be checked. This number is checked to see that it is within the 15 field limit. If the number is not, the program returns to the main menu.

Line 3730 sets the "F" array pointer. Lines 3740-3790 read the necessary data from the FIELDS11 disk file. Lines 3810-3890 print the present value for SMD and the last update date. Then the operator is asked if he/she would like to change SMD and/or last update date.

Lines 3900-3950 ask for the new value for SMD and the associated date. If only one of these is to be changed, the operator must also enter the current value for the other. Lines 3960-4000 write these new values back to the disk file. Lines 4020-4030 check to see if the update date was changed. If so, the program branches to the subroutine at line 4940 to allow an update of the weather file (rainfall and irrigation only) for the past three days. **NOTE:** If the update date is changed, then all other fields must be updated (SMD and associated date) to a common date. Then at the next selection of menu option 2, all updates begin from a common date.

The WEEKLY SMD UPDATE (menu option 1) begins with operator inputs of weather data and irrigation amounts. Then crop coefficients are calculated and the program branches (at line 1150) to subroutines (lines 4050-4920) for SMD updates. The subroutine at 4050 sets the values for EFF, SMD, SCMT, RT, IT to their last update value for the field being updated. Line 4120 refers to the file "WEATHER" which is used for storage of rainfall and irrigation amounts for three days prior to the present date. These data are needed for calculating the evapotranspiration component that results whenever the soil surface is wetted by rainfall or irrigation in the time period from crop emergence to full ground cover. Line 4130 sets up a pointer for reading rainfalls and irrigations as previously stored for each field. Field 1 uses positions 1,2,3, field 2 uses positions 4,5,6; etc. The "WEATHER" file is initialized by

```
4050 REM CALCULATE THE SMD
4060 REM SMD OF LAST UPDATE
4070 EFF = F(I + 4): REM EFF IS THE EFFICIENCY OF IRRIGATION SYSTEM IN PERCENT
4080 SMD = F(I + 6): REM SMD OF THE LAST UPDATE
4090 SCMT = F(I + 5): REM THE TOTAL AVAILABLE SOIL MOISTURE CAPACITY
4100 RT = F(I + 8): REM THE TOTAL RAINFALL TO DATE
4110 IT = F(I + 7): REM THE TOTAL IRRIGATION TO DATE
4120 REM FIND RAINFALL PLUS IRRIGATION FOR THE LAST THREE DAYS
4130 V = ((F(I)-1)*3) + 1
4140 PRINT D$; "OPEN WEATHER, L15"
4150 FOR A = 0 TO 2
4160 PRINT D$; "READ WEATHER,R"; A + V
4170 INPUT P1(A)
4180 NEXT A
4190 PRINT D$; "CLOSE WEATHER"
4200 REM
4210 REM FIND THE SMD FOR EACH DAY OF THE UPDATE INTERVAL
4220 FOR ID = 0 TO DX
4230 GOSUB 4330
4240 NEXT ID
4250 REM PLACE THE WEATHER DATA IN DISK FILE
4260 PRINT D$; "OPEN WEATHER, L15"
4270 FOR A = 0 TO 2
4280 PRINT D$; "WRITE WEATHER,R" A + V
4290 PRINT P1(A)
4300 NEXT A
4310 PRINT D$; "CLOSE WEATHER"
4320 RETURN
```

the subroutine at lines 4940-5150. The loop at 4120-4150 reads the rainfall and irrigation amounts for the previous 3 days into an array "P1." The loop at lines 4190-4210 with a branch to the subroutine at 4300 updates SMD for each day in the update interval. Lines 4220-4280 are executed subsequent to the return at line 4920. Thus, this program segment updates the P1 array in the "WEATHER" file.

```

4330 REM THE FINAL CALCULATION OF SMD
4340 SMD = SMD-RF(ID)-IR(ID)*EFF / 100
4345 P1(3) = RF(ID) + IR(ID)
4350 IF SMD < 0 THEN SMD = 0
4360 FC = (SCMT-SMD) / SCMT: REM FIND THE FIELD CAP.
      FOR THAT DATE
4370 FC = FC*100.00
4380 ET1 = 0
4390 REM AKA, SOIL MOISTURE COEFFICIENT
4400 IF FC <= 50 THEN AKA = (1 / 50)*FC: GOTO 4420

```

Line 4340 computes the SMD for day ID as changed by rainfall and irrigation amounts. Line 4345 sums irrigation and rainfall amounts for day ID and places this sum into the P1 array. If rainfall and irrigation reduce SMD to less than zero, SMD is set equal to zero. Lines 4360-4370 determine the percent available water remaining in the root zone. ET in this model is estimated as $ET = (KCO * AKA) * EP + ET1$. ET1 represents the ET component from the soil surface when it is wet due to rainfall or irrigation and when ground cover is incomplete; i.e.; $KCO \leq 0.9$. Line 4380 initializes ET1 to zero. The AKA coefficient adjusts potential evapotranspiration (EP) for effects of root zone moisture level. AKA is computed in lines 4400-4410. The crop coefficient term (KCO) is computed from crop curves that reflect degree of surface ground cover (see line 2920). The KCO term is rounded off to two decimal places and stored at AB in lines 4420-4430.

The decimal forms of KCO and AKA are multiplied in line 4440 and then represented by the term TKC. Lines 4460-4470 determine the average temperature and convert solar radiation from calories (the usual input) to inches of evaporated water equivalent. Potential evapotranspiration is calculated in line 4480. Evapotranspiration (ignoring the surface wetting effect) is estimated in line 4490. Line 4510 checks to determine if the crop ground cover is complete or whether it has developed to where soil surface wetness need not be considered; i.e., KCO (ID), as defined above, is greater than 0.9.

```

4480 EP = (0.014*TA-0.37)*SR
4490 E = TKC*EP
4500 REM ET ADJUSTMENT FOR SOIL SURFACE
      WETNESS IN PERIOD OF INCOMPLETE COVER
4510 IF KCO(ID) >= 0.9 THEN ET1 = 0: GOTO 4810
4520 IF P1(2) < 0 THEN 4680
4530 ET1 = 0.8*(.9-KCO(ID))*EP
4540 P1(2) = P1(2)-ET1
4550 IF P1(2) > 0 THEN 4800
4560 P1(1) = P1(1) + P1(2)
4570 P1(2) = 0
4580 IF P1(1) >= 0 THEN 4800
4590 P1(0) = P1(0) + P1(1)
4600 P1(1) = 0
4610 IF P1(0) >= 0 THEN 4800
4650 ET1 = ET1 + P1(0)
4660 P1(0) = 0
4670 GOTO 4800
4680 IF P1(1) <= 0 THEN 4720
4690 ET1 = 0.5*(0.9-KCO(ID))*EP
4700 P1(1) = P1(1)-ET1
4710 GOTO 4580
4720 IF P1(0) <= 0 THEN 4800
4730 ET1 = 0.3*(0.9-KCO(ID))*EP
4740 P1(0) = P1(0)-ET1
4750 GOTO 4610
4800 IF ET1 < 0 THEN ET1 = 0

```

Lines 4520-4800 compute the effect of surface wetness on daily evapotranspiration with the result expressed as ET1.

```

4410 AKA = 1
4420 AW% = KCO(ID)*100.00 + 0.5
4430 AB = AW% / 100
4440 TKC = KCO(ID)*AKA: REM TOTAL CROP COEFFICIENT IS THEN BASED ON
      CROP COEFFICIENT AND ADJUSTMENT FOR SOIL MOISTURE
4450 REM FIND THE POTENTIAL EVAPOTRANSPIRATION FOR THE PRESENT DAY
4460 TA = (MT(ID) + MNT(ID)) / 2: REM THE AVE. TEMP
4470 SR = SL(ID) / 1485.7: REM CONVERSION OF SOLAR RADIATION IN CALORIES TO INCHES OF WATER EQUIVALENT

```

```

4810 E = E + ET1
4820 SMD = SMD + E
4830 IF SMD >= SCMT THEN SMD = SCMT
4840 SD% = SMD * 100.0 + 0.5
4850 SMD = SD% / 100

```

Lines 4810-4820 calculate the total evapotranspiration and consequent SMD update as of the current day, ID. Line 4830 verifies that the current SMD is not greater than the total root zone available water holding capacity, SCMT. Lines 4840-4850 round off the SMD update to two decimal places.

```

4860 FOR A = 0 TO 2
4870 P1(A) = P1(A + 1)
4880 NEXT A
4890 RT = RT + RF(ID)
4900 IT = IT + IR(ID)
4910 TW% = KCO * 100.0 + 0.5 : KCO = TW% / 100.0
4920 EG% = ET1 * 100.0 + .5 : ET1 = EG% / 100.0
4930 RETURN

```

Lines 4860-4880 adjust the P1 array to reflect values for only the past three days. Lines 4890-4990 sum the rainfall and irrigation amounts to the current running totals and lines 4900-4910 round off the printed values to the screen to two decimal places.

The subroutine at 4940 initializes or updates the P1 array that is used for computing the soil surface wetness factor. This routine is used only (see branch at line 4030) when menu option No. 2 (line 280) is selected. If irrigation and rainfall amounts equal zero in the past three days the routine sets the P1 array to 0 for that field. "J" is the file pointer that is computed in line 3730.

5160 REM SUBROUTINE TO CORRECT INITIALIZING ERRORS

```

5170 HOME
5180 PRINT : PRINT "1 FIELD #—>"F(I)
5190 PRINT "2 CROP#—>"F(I + 1)
5200 PRINT "3 EMERGENCE DATE—>"F(I + 2)"F(I + 3)
5210 PRINT "4 PRESENT DATE—>"F(I + 9)"F(I + 10)
5220 PRINT "5 EFFICIENCY OF IRR—>"F(I + 4)
5230 PRINT "6 SOIL MOISTURE CAP—>"F(I + 5)
5240 PRINT "7 PRESENT SMD—>"F(I + 6)
5250 PRINT "8 30% DEPL.—>"F(I + 11)
5260 PRINT "9 50% DEPL.—>"F(I + 12)
5270 PRINT "10 IR TO DATE—>"F(I + 7)
5280 PRINT "11 RAIN TO DATE—>"F(I + 8)
5290 PRINT "12 ALL CORRECTIONS ARE MADE"
5300 PRINT
5310 INPUT "WHICH OF THE INPUTS IS IN ERROR ";ER
5320 PRINT
5330 IF ER < 1 OR ER > 12 THEN PRINT "ERROR IN
SELECTION PLEASE REENTER": PRINT :
GOTO 5310
5340 ON ER GOTO 5350,5380,5500,5530,5560,5590,5620,
5650,5680,5710,5740,5770,6580

```

Continued on page 22

```

4940 REM ROUTINE TO UPDATE RAINFALL AND IRRIGATION AMOUNTS FOR THE PAST THREE DAYS
4950 HOME : PRINT : INPUT "WAS THERE ANY RAINFALL OR IRRIGATION APPLIED IN THE PAST THREE DAYS (Y/N)";
ASW$
4960 IF ASW$ = "N" THEN 5090
4970 PRINT : PRINT "ENTER THE SUM OF RAINFALL AND IRRIGATION AMOUNTS STARTING THREE DAYS AGO"
4980 V = ((NF - 1) * 3) + 1
4990 FOR A = 0 TO 2
5000 INPUT P1(A)
5010 NEXT A
5020 PRINT D$; "OPEN WEATHER,L15"
5030 FOR A = 0 TO 2
5040 PRINT D$; "WRITE WEATHER,R" A + V
5050 PRINT P1(A + 1)
5060 NEXT A
5070 PRINT D$; "CLOSE WEATHER"
5080 RETURN
5090 PRINT D$; "OPEN WEATHER,L15"
5100 FOR A = 0 TO 2
5110 PRINT D$; "WRITE WEATHER,R" A + V
5120 PRINT 0
5130 NEXT A
5140 PRINT D$; "CLOSE WEATHER"
5150 RETURN

```

Continued from page 21

```
5350 PRINT "FIELD # IS "F(I)
5360 INPUT "ENTER CORRECT FIELD #";F(I)
5370 GOTO 5170
5380 PRINT "FIELD # IS "F(I + 1)
5390 PRINT "CROP NUMBERS"
5400 PRINT "-----"
5410 PRINT "1 = SUNFLOWERS"
5420 PRINT "2 = CORN"
5430 PRINT "3 = WHEAT"
5440 PRINT "4 = PINTO BEANS"
5450 PRINT "5 = SOYBEANS"
5460 PRINT "6 = ALFALFA"
5470 PRINT "7 = BARLEY"
5480 INPUT "ENTER NEW CROP #";F(I + 1)
5490 GOTO 5170
5500 PRINT "EMERGENCE DATE IS "F(I + 2)"/"F(I + 3)
5510 INPUT "NEW EMERGENCE DATE (M,D) ";F(I + 2),
    F(I + 3)
5520 GOTO 5170
5530 PRINT "PRESENT DATE IS "F(I + 9)"/"F(I + 10)
5540 INPUT "ENTER NEW PRESENT DATE (M,D) ";
    F(I + 9),F(I + 10)
5550 GOTO 5170
5560 PRINT "EFFICIENCY OF IRR. IS "F(I + 4)
5570 INPUT "ENTER NEW EFFICIENCY OF IRR, ";F(I + 4)
5580 GOTO 5170
5590 PRINT "SOIL MOISTURE CAP. IS "F(I + 5)
5600 INPUT "ENTER NEW SOIL MOISTURE CAP. ";F(I + 5)
5610 GOTO 5170
5620 PRINT "PRESENT SMD IS "F(I + 6)
5630 INPUT "ENTER NEW SMD";F(I + 6)
5640 GOTO 5170
5650 PRINT "30% DEPL. IS "F(I + 11)
5660 INPUT "NEW 30% DEPL. ";F(I + 11)
5670 GOTO 5170
5680 PRINT "50% DEPL IS "F(I + 12)
5690 INPUT "NEW 50% DEPL.";F(I + 12)
5700 GOTO 5170
5710 PRINT "IR TO DATE IS "F(I + 7)
5720 INPUT "NEW IR TO DATE ";F(I + 7)
5730 GOTO 5170
5740 PRINT "RAINFALL TO DATE IS "F(I + 8)
5750 INPUT "NEW RAINFALL TO DATE ";F(I + 8)
5760 GOTO 5170
5770 RETURN
```

Menu option 3 initializes the FIELDS11 file for each field. These data are echoed back (lines 2210-2330). If incorrect a no (N) at line 2350 branches the program to the subroutine at line 5160 for error corrections. The subroutine at 5160-5750 prints out the data as was input to the "F" array for a given field. The user inputs the line number of the data in error and the program then displays the old value and asks for the input of the corrected value. Then all the data are re-displayed. If all data are correct the user selects option number "12" to exit this routine.

Similarly, after the weather data for a weekly update have been entered, these data are echoed back

5780 REM SUBROUTINE FOR ERROR TRAPPING WEATHER DATA

```
5790 HOME
5800 PRINT "NO.";
5810 HTAB 4: PRINT "DATE";
5820 HTAB 10: PRINT "RAIN";
5830 HTAB 18: PRINT "MAX";
5840 HTAB 24: PRINT "MIN";
5850 HTAB 30: PRINT "SOLAR"
5860 HTAB 10: PRINT "FALL";
5870 HTAB 18: PRINT "TEMP";
5880 HTAB 24: PRINT "TEMP";
5890 HTAB 30: PRINT "RADIATION"
5900 PRINT "-----"
5910 FOR ID = 0 TO DX
5920 IF L(M2) >= D2 + ID THEN 6000
5930 PRINT ID + 1;
5940 HTAB 4: PRINT M3"/"D2-L(M2) + ID;
5950 HTAB 10: PRINT RF(ID);
5960 HTAB 18: PRINT MT(ID);
5970 HTAB 24: PRINT MNT(ID);
5980 HTAB 30: PRINT SL(ID);
5990 GOTO 6060
6000 PRINT ID + 1;
6010 HTAB 4: PRINT M2"/"D2 + ID;
6020 HTAB 10: PRINT RF(ID);
6030 HTAB 18: PRINT MT(ID);
6040 HTAB 24: PRINT MNT(ID);
6050 HTAB 30: PRINT SL(ID)
6060 NEXT ID
6070 PRINT
6080 INPUT "ENTER THE NUMBER THAT
    CORRESPONDS TO THE DAY WITH INCORRECT
    WEATHER DATA. (ENTER 0 TO END)";ER
6090 IF ER = 0 THEN RETURN
6100 PRINT
6110 PRINT "ENTER THE RAINFALL, MAX TEMP, MIN
    TEMP, AND SOLAR RADIATION FOR THE DAY
    IN ERROR"
6120 INPUT RF(ER = 1),MT(ER-1), MNT(ER-1),SL(ER-1)
6130 GOTO 5780
```

for error deletion. A no (N) at line 940 branched the program to line 5780 to facilitate error corrections. The subroutine at 5780-6130 displays the weather data and asks which data are in error. The operator re-enters the correct data.

REFERENCES

- Lundstrom, D.R. and E.C. Stegman. 1983. Irrigation scheduling by the checkbook method. AE Circular 792. N. Dak. Coop Ext Serv., N. Dak. State Univ. Fargo, ND, 14 p.
- Jensen, M.E. and H.R. Haise. 1963. Estimating evapotranspiration from solar radiation. Proc. Am. Soc. Civ. Engr., J. Irrig. and Drain Div. 89:15-41.
- Jensen, M.E., J.L. Wright and B. J. Pratt. 1971. Estimating soil moisture depletion from climate, crop, and soil data. Trans. of ASAE 14(5): 954-959.
- Stegman, E.C., A. Bauer, J.C. Zubriski, and J. Bauder. 1977. Crop curves for water balance irrigation scheduling in S.E. North Dakota. Res. Rpt. No. 66 Agric. Expt. Stat. N. Dak. State Univ. Fargo, ND, 11 p.

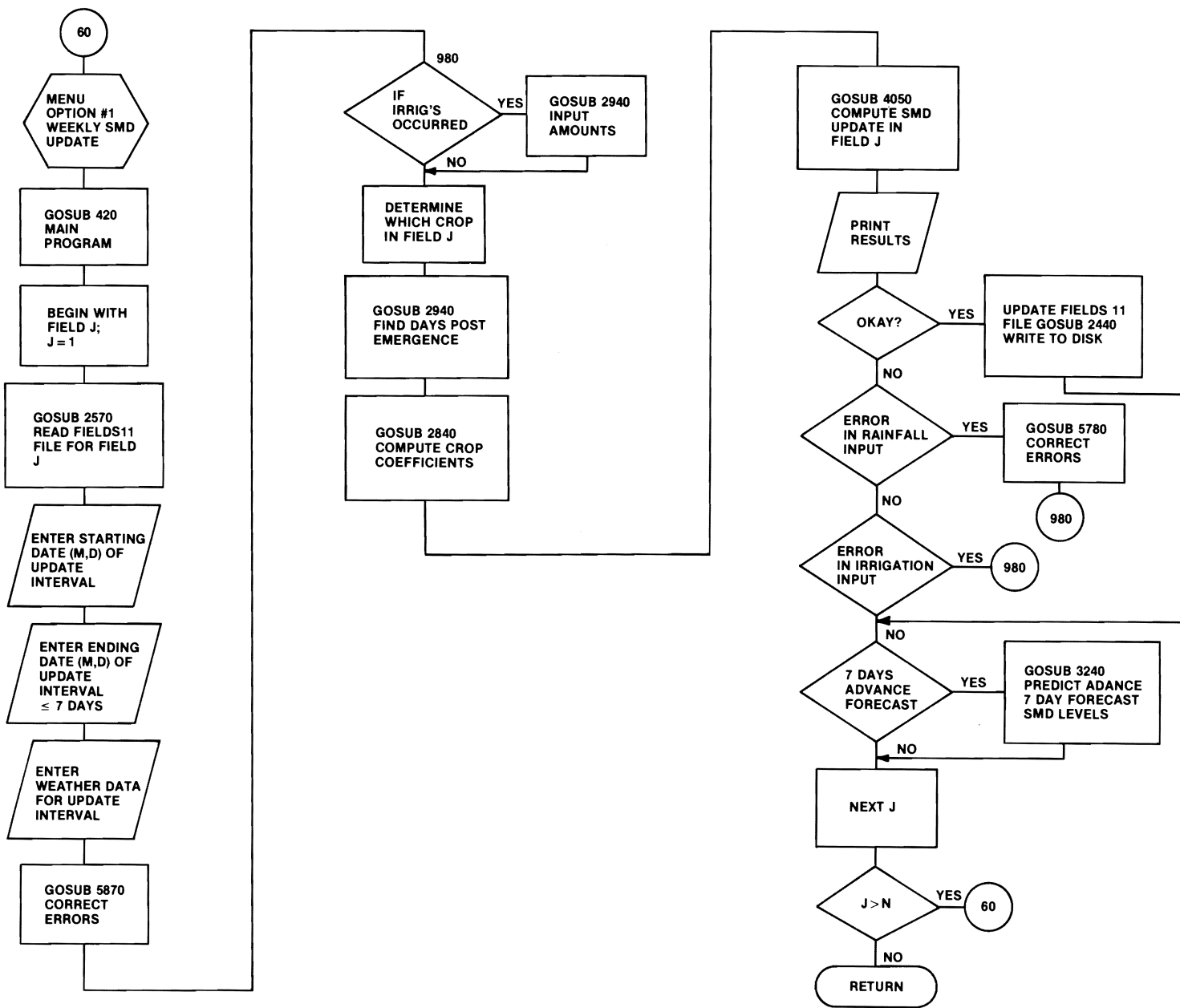


Fig. 23a. Flow chart for menu option #1.

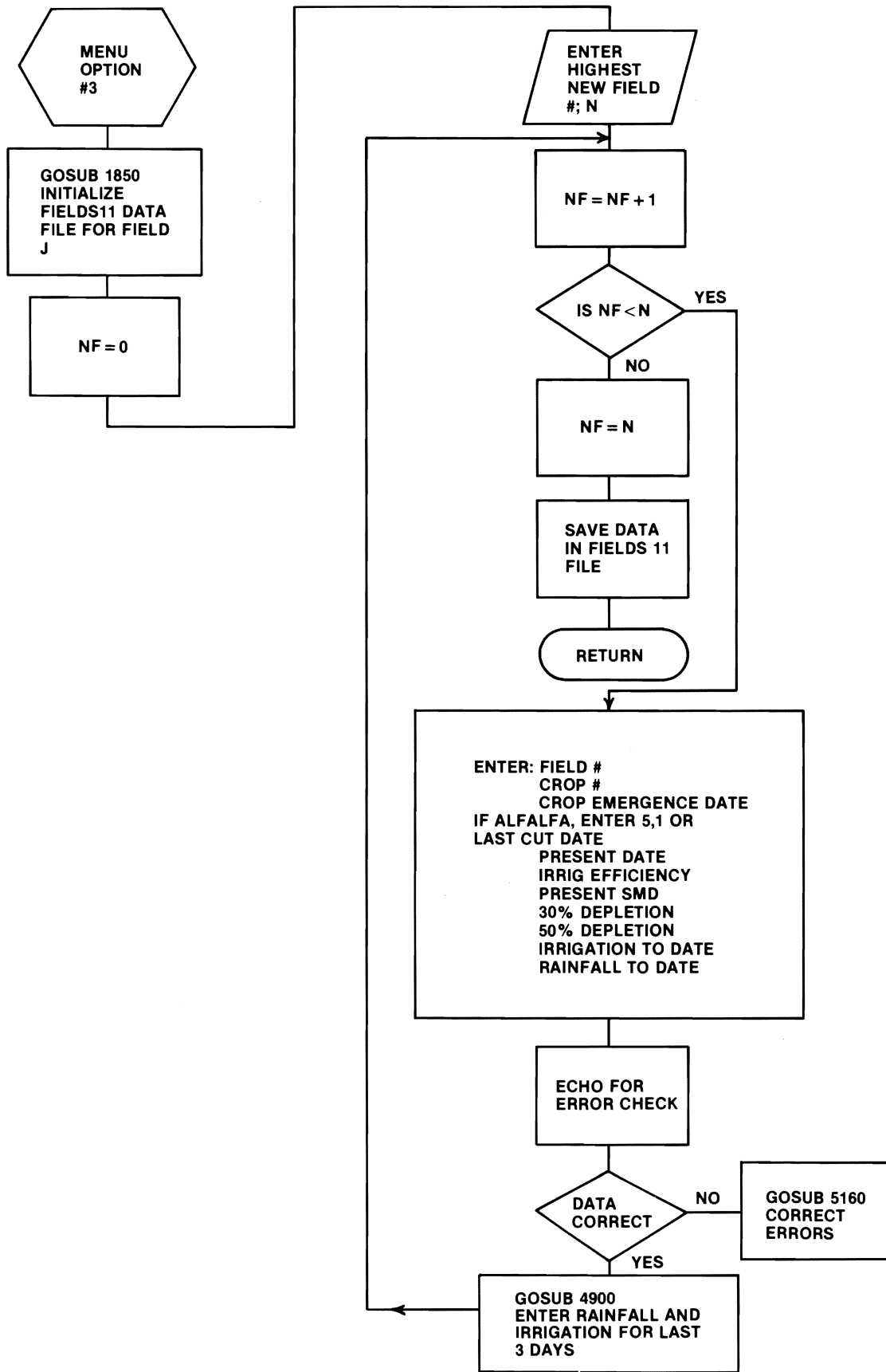


Fig. 23c. Flow chart for menu option #3.

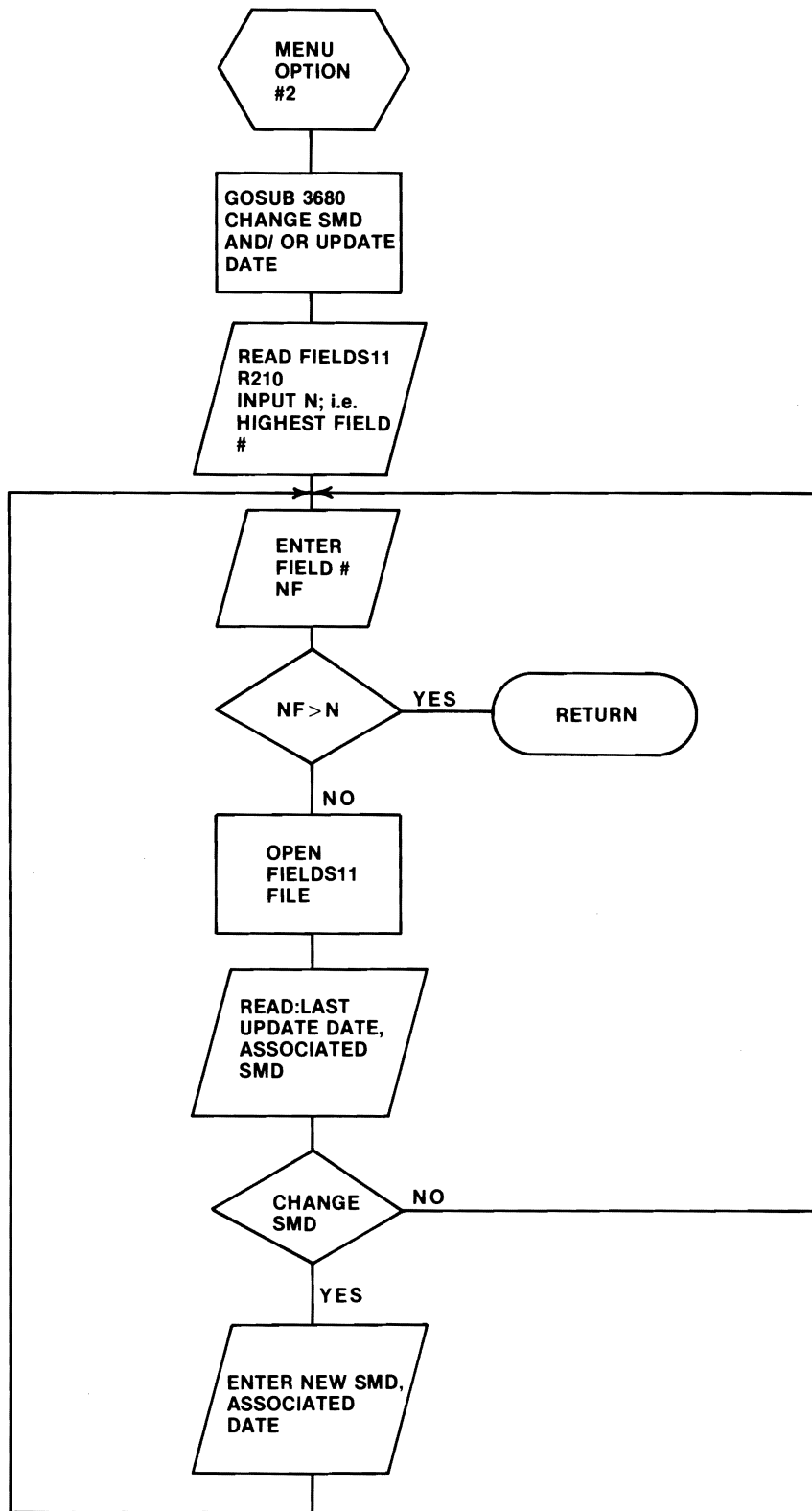


Fig. 23b. Flow chart for menu option #2.

```

10 REM PROGRAM NAME IS: CROP COEFFICIENTS
14 D$ = CHR$(4)
15 PRINT D$;"NOMON C,I,O"
16 DIM W(100)
17 INPUT "DO YOU WANT TO INPUT A NEW CROP?—>";ASW$
18 IF ASW$ = "Y" THEN 300
19 INPUT "DO YOU WISH TO DISPLAY OR CHANGE EXISTING COEFFICIENTS?—>";ASW$
20 IF ASW$ = "Y" THEN 22
21 END : HOME
22 INPUT "WHAT FILE NAME?—>";N$
25 N = 5
30 PRINT D$;"OPEN";N$",L15"
40 FOR I = 1 TO N
50 PRINT D$;"READ";N$",R";I
55 INPUT W(I)
60 NEXT I
70 PRINT D$;"CLOSE";N$
72 PR#1
74 PRINT : PRINT : PRINT
75 PRINT TAB( 15)N$
80 FOR I = 1 TO N
90 PRINT TAB( 10)"C"|"—>"W(I)
100 NEXT
102 PR# 0
105 PRINT "ENTER 0 TO ESCAPE"
110 INPUT "COEFF. NO. TO BE CHANGED C";I
120 IF I = 0 THEN 21
130 PRINT W(I)
140 INPUT "ENTER NEW VALUE—> ";V
150 W(I) = V
200 PRINT D$;"OPEN";N$",L15"
210 PRINT D$;"WRITE";N$",R";I
215 PRINT W(I)
220 PRINT D$;"CLOSE";N$
230 GOTO 105
300 INPUT "ENTER # OF POLYNOMIAL COEFFICIENTS ";N
310 FOR I = 1 TO N
320 PRINT "ENTRY C";I;
330 INPUT W(I)
340 NEXT
350 INPUT "NEW FILE NAME—>";N$
360 PRINT D$;"OPEN "N$",L15"
370 FOR I = 1 TO N
380 PRINT D$;"WRITE";N$",R";I
390 PRINT W(I)
400 NEXT I
410 PRINT D$;"CLOSE";N$
420 GOTO 19

```

Fig. 24. Program for creating and/or changing crop coefficient files.

North Dakota State University and the program authors shall have no liability or responsibility to any person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by this software, including, but not limited to any interruption of service, loss of business or anticipatory profits or consequential damages resulting from the use or operation of this software.

