Water Resources Center
University of Delaware
Newark, Delaware

WATBUG: A FORTRAN IV Algorithm for Calculating the Climatic Water Budget

by

Cort J. Willmott

OWRT Project No. A-040-DEL
The Use of the Climatic Water Budget
in Water Resources Management and Control

December, 1977

REPORT NO. 1
OWRT PROJECT NO. A-040-DEL
THE USE OF THE CLIMATIC WATER BUDGET
IN WATER RESOURCES MANAGEMENT AND CONTROL

The work upon which this publication is based was supported in part by funds provided by the United States Department of the Interior as authorized under the Water Resources Research Act of 1964, Public Law 88-379.

University of Delaware Water Resources Center
Contribution Number 23

Also issued as
Volume XXX, No. 2, 1977
Publications in Climatology
C. W. Thornthwaite Associates
Laboratory of Climatology
Elmer, New Jersey

TABLE OF CONTENTS

I. INTRODUCTION

A FORTRAN IV computer program (WATBUG) is presented in order that it may aid in the calculation of climatic water budgets. The program is designed to be used for a variety of problems while requiring a minimum amount of input information. At the same time, every attempt has been made to make the code transparent and, as a result, it should be easy to modify when additional or alternative computations are desired. Many of the restrictions of past programs (e.g., Stone, et al., 1971) have been removed and the algorithm can calculate, for example, daily and/or monthly water budgets. Although the main purpose of this report is to describe the program, the significance and background of water budget analysis should, at least, be mentioned.

The far-reaching importance and history of water balance climatology is underscored by an extensive literature as well as by the dedication of researchers who have repeatedly performed arduous hand calculations in order to obtain those all-important estimates of evapotranspiration, soil moisture storage, runoff and deficit. A recent survey of the applications of the water budget in physical geography, for instance, indicates that, in addition to a rich history, its use is ongoing, if not expanding, in a variety of fields ranging from geomorphology to agriculture (Carter, et al., 1973). Even more recently, a detailed discussion of the nature and variety of techniques, as well as their uses in modern environmental analysis, has been compiled by Mather (1978). After reading through these, or any number of other papers, it seems clear that (1) the water budget has been and will continue to be a particularly important theme in climatology and (2) its betterment is a highly rewarding research endeavor.

Thornthwaite's (1948) approach to the water budget has often been singled out for criticism because (1) it is empirically based and, (2) it has been highly successful (Lee, 1978; Terjung, 1976). Although I strongly support arguments for more rigorous and/or systematic research in climatology, it can be said that many of the critics of empirical water budgets have misinterpreted the purpose and utility of regression--broadly defined. Researchers may correctly use morphological links and/or regression when (1) the data necessary for more rigorous analyses are lacking or not "realistically" obtainable and/or (2) the physical-biotic mechanisms that produce the desired answer are either well-known, unknown or unimportant. This is acceptable science, as most beginning texts in the sciences indicate. Lee (1978:135), however, believes "There is no adequate method of predicting evapotranspiration rates in the biosphere based on simple weather-element data." We may quibble about what constitutes "adequate" or problems of scale, but it is undeniable that empirically-based water budgets have been very successful at satisfying the only criterion on which they should be judged--accuracy (McGuinness and Bordne, 1972). The only real problem with such procedures is that they can "fool" an unsuspecting student or researcher into believing that they "explain" environmental processes or that they work equally well in all environments. These methods were never intended to be used for explanatory purposes and, as Thornthwaite (1961) would argue, they are merely temporary and useful only where they provide an answer where none better is available.

Until "better" methods and data are available, it is hoped that this program will contribute to the already extensive literature by lessening the laborious computations which have been traditionally associated with the climatic water budget.

Capabilities and Restrictions

Program WATBUG currently relies on the Thornthwaite (1948) method of calculating potential evapotranspiration (APE) although the subroutine where those estimates are made can be easily replaced by another, if desired. If the reader would like to see a comparison of techniques, it is suggested that McGuinness and Bordne (1972) or Mather (1978) be consulted. WATBUG has the following advantages over most previously published water budget programs (e.g., see Stone, et al., 1971):

1. Budgets can be computed on a monthly or daily basis.

2. The program can iterate over periods of record up to 40 years for monthly budgets or up to one year for daily budgets in order to "balance the budget". These limits can also be easily raised. This procedure is similar to that normally done by hand to obtain the initial soil moisture value.

3. Initial values of actual soil moisture and a station's heat index (see Thornthwaite and Mather, 1957) are not required as long as "balancing" (item 2 above) is performed on at least a year's data.

4. Following the balancing of the soil moisture budget, budgets may be calculated on a day-by-day or month-by-month basis.

5. No "look-up" tables (arrays from which values are interpolated) are needed as all relationships are explicitly specified. This tends to make computations more accurate and, as a result, WATBUG's computations may not always agree with those by performed by hand using tables which make discrete jumps at regular intervals.

6. The program can be easily modified as logical groups of computations have been segregated and appear as subroutines. Also, for ease of modification, WATBUG is extensively commented and the program logic has been kept simple--even, in a few places, at the expense of a more computationally-efficient code.

7. The required input is minimal, i.e., air temperature, precipitation and a few initial parameters.

8. The raw air temperature and precipitation information can be in a variety of units as WATBUG will make the translations. At the same time, the format by which the climatic data appears is flexible because it is specified by the user.

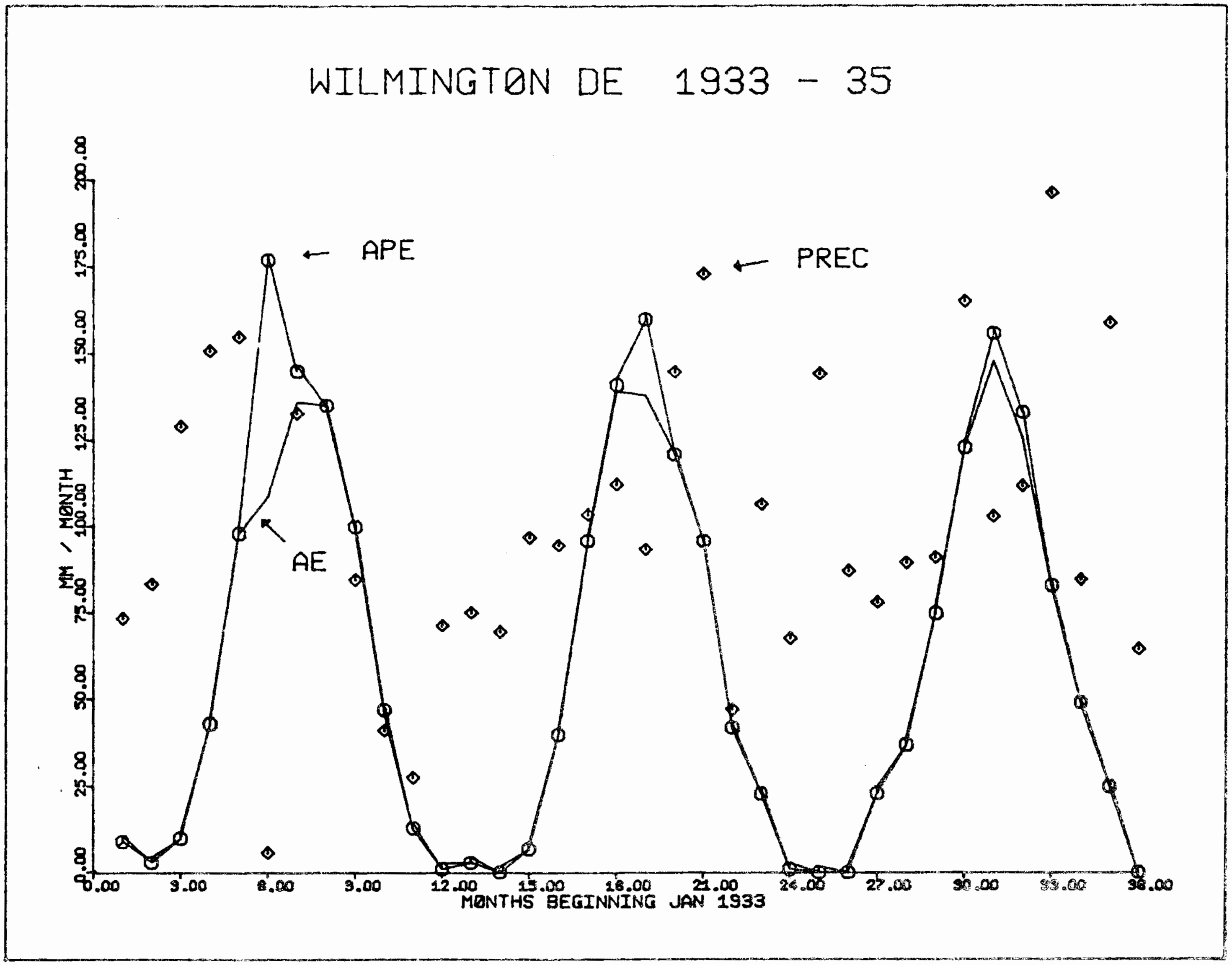


Figure 1. Exemplary monthly water budget at Wilmington, DE 1933 - 1935. The diamonds are precipitation magnitudes (PREC) while the curves represent potential evapotranspiration (APE) and actual evapotranspiration (AE). [Note: (1) The APE curve is distinguished from the AE curve by circles which are plotted at the data points; (2) PREC for August 1933 was not plotted because its value is 309 mm.]

9. Multiple budgets, i.e., on different stations and/or non-consecutive time series may be done in a single run.

The only requirement is that periods of time to be evaluated as a single budget should be consecutive. Because of its general nature, however, WATBUG does not perform a number of problem-specific computations.

The algorithm, for example, does not derive estimates of runoff as such functions are numerous and site specific. Moreover, only two soil moisture resistance functions (i.e., to evapotranspiration) are contained by WATBUG and both are single soil layer models. A multiple soil layer model was considered but without detailed knowledge of the vegetation cover and soil characteristics such precision would be unwarranted. Such procedures may be added to WATBUG with little difficulty by anyone familiar with simple programming and the water budget. These omissions notwithstanding, the program should aid both researchers and students in the computation of climatic water budgets.

Evaluation of the Program

WATBUG was tested on a variety of multi-annual monthly and daily data sets and all of the program's options were tried. One such monthly budget was plotted and is presented in figure 1. The results of each run were compared to hand computations made by J. R. Mather and WATBUG's answers were, in all cases, within a mm or two of the hand-computed values. The reasons for these slight differences are discussed elsewhere in this report. Because of the chance that possible problems were overlooked, users are encouraged to contact the author if errors are discovered.

II. PREPARATION OF INPUT

Program WATBUG only requires three control cards (records of at least 72 characters) for each station and/or new time period to be evaluated. Card one merely contains a 72-character problem label (anything you want, i.e., valid FORTRAN characters) in columns 1 through 72. These columns may be left blank or filled at the user's discretion but this card must appear as the first card in a new problem. On the second card, all the required initial parameters must be specified.

All information required on control card (record) two is summarized in table 1. Although many of the initial parameters on control card two need not be specified, others must. For this reason, reading table 1 carefully and in its entirety is prerequisite to a successful run. Computing jargon has been kept to a minimum so that most users who are familiar with the water budget can easily read table 1. Users should be somewhat careful in selecting the balancing period (N), because it can significantly affect the results if incorrect.

Balancing can be accomplished for any portion of the entire climatic record beginning with the first day or month, although periods of time which are not multiples of complete years should not be "balanced." If data remain beyond the "period of balancing," they can be budgeted on a day-by-day or month-by-month basis at the user's request, i.e., if NT is greater than N. Given a couple of further initial parameters (see ST(1) and HEAT in table 1) balancing can also be skipped if desired (N = 1), and all computations will be undertaken on a day-by-day or month-by-month basis. Once control card two has been successfully punched, the last control card (number three) can be set up.

Control card three specifies the format by which the raw air temperature and precipitation data are to be read. Any standard FORTRAN format is acceptable although no more than two observations (one on each variable) may be encoded on a single data card (record). The format statement may appear anywhere in columns 1-72 of control card three. WATBUG first reads an air temperature value and then a precipitation magnitude. This sequence is repeated over and over again until the entire data set is read. If on each data card, for example, you had encoded air temperature in columns 6-10 and precipitation in columns 16-20, with the decimals punched, and each card represented one time period (i.e., day or month), then the following formats could be used:

```
(5X,F5.0,5X,F5.0)

(2(5X,F5.0))

(2F10.0).
```

If the user is unfamiliar with FORTRAN format statements, nearly any beginning FORTRAN manual can be consulted. Control card three is followed by the raw

Table 1

Preparation of Control Card Number Two

Variable Name	Columns	Just.[1]	Description of the Parameter and Defaults[2]
N	1-5	R	Number of months or days over which soil moisture balancing is to occur. If N is left blank or equals 1 or 0, balancing does not take place and ST(1) and HEAT must be specified.
NT	6-10	R	Total number of months or days over which the water budget is to be calculated. NT should be greater than or equal to N. If it equals 0 or is left blank, it is set equal to N.
KD	11-15	R	Day of the month where the first calculations are to begin. KD should be less than or equal to the number of days in month KM. If left blank or zero in daily or day-by-day computations, it is set equal to the first day of the month, i.e., 1. When KD is left blank or set at zero and monthly or month-by-month computations are being made, KD will be set equal to a representative day for the middle of month KM, i.e., 15.
KM	16-20	R	First month of calculations. KM must be between zero and 12. When KM is left blank or set at zero, it is assumed to be 1, i.e., the first month of the year--January.
KY	21-25	R	First year of calculations. Include the last two digits of the year only. If KY is left blank, it is assumed to equal zero and the first year of computation will be 1900.
FC	26-30	N	Soil water holding capacity, or field capacity, of the top (only) soil layer in mm. FC must be specified. If not, it will be assumed to be zero.
SM	31-35	N	Determines which one of two resistance functions of soil moisture to evapotranspiration will be used. When SM is left blank or set at zero, the availability of soil moisture to evapotranspiration will decline linearly with the ratio of actual soil moisture to field capacity. Any other numeric designation will result in soil moisture being withdrawn at the maximum possible rate until the ratio of actual soil moisture to field capacity drops below 0.7 after which time a linear decline in availability is assumed (see Mather, 1974: 106, curves C and G).

Table 1 (Continued)

Variable Name	Columns	Just.[1]	Description of the Parameter and Defaults[2]
LAT	36-40	N	The station latitude in degrees.
DT	41-45	N	Determines whether the calculations are to be daily or monthly. If DT is left blank or set at zero, monthly or month-by-month computations are assumed. Any other numeric designation will result in daily or day-by-day budgeting.
TUNIT	46-50	N	Indicates the units of the raw air temperature data in order that they may be properly translated into degrees Celsius. 1.0 indicates that the raw data are in degrees Fahrenheit. 2.0 means Kelvin while any other numeric designation, or leaving TUNIT blank, indicates degrees Celsius.
PUNIT	51-55	N	Indicates the units of the raw precipitation data in order that they may be properly translated into mm. 1.0 indicates that the raw data are in cm, 2.0 means inches and 3.0 means hundreths of an inch. Leaving PUNIT blank, or giving it any other numeric designation, indicates mm.
ST(1)	56-60	N	Estimated soil moisture content (mm) of the top (only) soil layer just prior to the beginning of computations. ST(1) only needs to be specified when balancing is not done. Otherwise, it is computed during balancing.
HEAT	61-65	N	Estimated (Thornthwaite, 1948) heat index for the station. HEAT only needs to be specified when balancing is not done.
INDEX	66-70	N	Should be set greater than zero when calcuations for a subsequent station and/or time period are to follow the computations to which this control card refers. INDEX may be left blank or set at zero if only a single water budget is desired.

[1]"Just." refers to column justification. "R" indicates that the designated numerical value should be "right justified," i.e., placed as far to the right in the five-column field as possible. A decimal point should not be punched. "N" means that no justification is required, i.e., the number may appear anywhere in the proper five-column field; but, the decimal point should be punched.

[2]Each parameter must either be left blank or specified by a number. No letters or other non-numeric characters are acceptable.

air temperature and precipitation information; that is, your entire card data set for each station and/or new time period <u>must</u> contain control cards one, two and three (in that order) followed by the raw data (see Appendix 2 for examples).

When computations are daily or day-by-day and they include a leap year, a corrective action may be desired. If so, see the discussion at subroutine DATE.

III. INTERPRETING THE OUTPUT

Unlike Thornthwaite and Mather (1957), WATBUG's water budget results are formatted with the variables across the paper (columns) and time periods on the vertical dimension (rows). This minor alteration was made because (1) most of science uses this form of a data matrix and (2) it is more efficient--programmatically.

The program first writes the information contained on control card number one which can be useful in later identifying a particular problem or run (see Appendix 3 for sample results). Following this, WATBUG writes (1) the number of months (or days) over which balancing is to occur, (2) the total number of months (or days) to be evaluated, (3) the soil moisture (or field) capacity (mm) and (4) the latitude (degs). Each of these numbers is labelled for easy identification. WATBUG then proceeds to write and label the monthly or daily computations.

Monthly and daily budgets are formatted alike except that each case represents a month in the former and a day in the latter. The first variable is either the monthly designations (under the heading "MO") or the daily designations (labelled "DY"). In addition to the numeric time period labels--DY or MO--the year is specified at the very beginning of a new year's calculations. When daily budgets are being written, monthly labels are similarly printed at the beginning of each new month. Reading left to right across the output table, the following variables (with their associated labels in parens) appear:

1. air temperature (TEMP) in °C
2. unadjusted potential evapotranspiration (UPE) in mm
3. adjusted potential evapotranspiration (APE) in mm
4. precipitation (PREC) in mm
5. precipitation minus adjusted potential evapotranspiration (DIFF) in mm
6. soil moisture storage (ST) in mm
7. change in storage from the preceding day or month (DST) in mm
8. actual evapotranspiration (AE) in mm
9. soil moisture deficit (DEF) in mm
10. soil moisture surplus (SURP) in mm.

Users should note that regardless of the units of the raw input data, the results are given in whole mm's. At the end of each year (and month in daily budgets) totals of APE, PREC, AE, DEF and SURP are given.

Yearly totals are calculated from each January 1 (or the first case read for the first year) to either the end of that year (December 31) or the end of processing--whichever comes first. In either case, the totals are printed at the end of the calendar year at the bottom of the appropriate column. When daily or day-by-day budgets are being calculated, monthly totals are also calculated and written at the end of each month--in the appropriate columns. Again, the summing begins with the first day of the month (or first case read) and ends with the last day of the month or the end of processing--whichever comes first. At the end of computations, the total number of cases evaluated is printed and labelled in order that the user may check to see if the proper number of computations have been made.

When water budget computations are done by hand, or by programs which rely heavily on look-up tables, intermediate values are often rounded to whole numbers at each step in the computational sequence. As a result, rounding errors may accumulate. WATBUG, on the other hand, does not round during the computation of any of the budget terms thereby minimizing these errors. After the budget terms have been calculated, however, they are rounded to the nearest whole mm just prior to their being written onto paper. This was done (1) because accuracy beyond a whole mm is superfluous and (2) to be consistent with previous presentations of water budget results (Thornthwaite and Mather, 1957). As a result, a hand check of WATBUG's results will appear to show minor accounting errors. If, for example, WATBUG calculated an APE of 131.6 mm, with an associated PREC of 58.2 mm, then the difference (DIFF) would be

$$\text{DIFF} = \text{PREC} - \text{APE} = -73.4. \qquad \text{mm} \qquad (1)$$

The program would then print the rounded versions of these numbers. Judging from the output, therefore, the equation would be

$$58 - 132 = -73 \qquad \text{mm} \qquad (2)$$

which, according to most mathematics texts, is incorrect. Actually, however, the "correct" difference is closer to -73 than -74 (the answer derived by hand from the output values of PREC and APE). Rounding inconsistencies become even more apparent in monthly and yearly totals as they can accumulate. WATBUG's yearly totals, for example, could easily be dissimilar to totals calculated by hand from the output tables by 5, 10 or more mm. The reader is again cautioned, however, that WATBUG's values are probably more correct than their hand-produced counterparts.

Another apparent problem in interpreting the results occurs when daily budgets are calculated over a leap year and no corrective action is taken prior to running the program. Because WATBUG does not contain a leap year correction, users may either (1) delete February 29 from the input data or (2) be a bit careful in interpreting the results as the day labels will be one day ahead of their associated values following February 28 of a leap year. The former is probably the easiest corrective action, and it should have a minimal impact if budgets are calculated for time periods longer than a month or two. See the discussion of subroutine DATE if more details about leap year problems and corrections are desired.

IV. DESCRIPTION OF THE ALGORITHM

Methods of computation contained in program WATBUG are described in this section. The discussion is organized by subroutine, i.e., each subroutine is described in a separate subsection. Subroutines are presented in the general order that they are called by subroutine MAIN with the exception of the main program (figure 2). That is, the main program is described first followed by subroutines MAIN, DATE, MATHER, DAY, DIFF and so on.

Relationships appear in "quasi-FORTRAN" in order that the discussion and the appended source program (Appendix 1) are more easily comparable. At the same time, the exact form of an "equation" may differ slightly from its appended counterpart in order that this narrative may be more easily understood by readers without a strong background in computing. For users not at all familiar with FORTRAN equations, explanations of operators and/or procedures peculiar to FORTRAN are provided. This section is not recommended for the casual user but it should be helpful to those using the algorithm for research.

Main Program

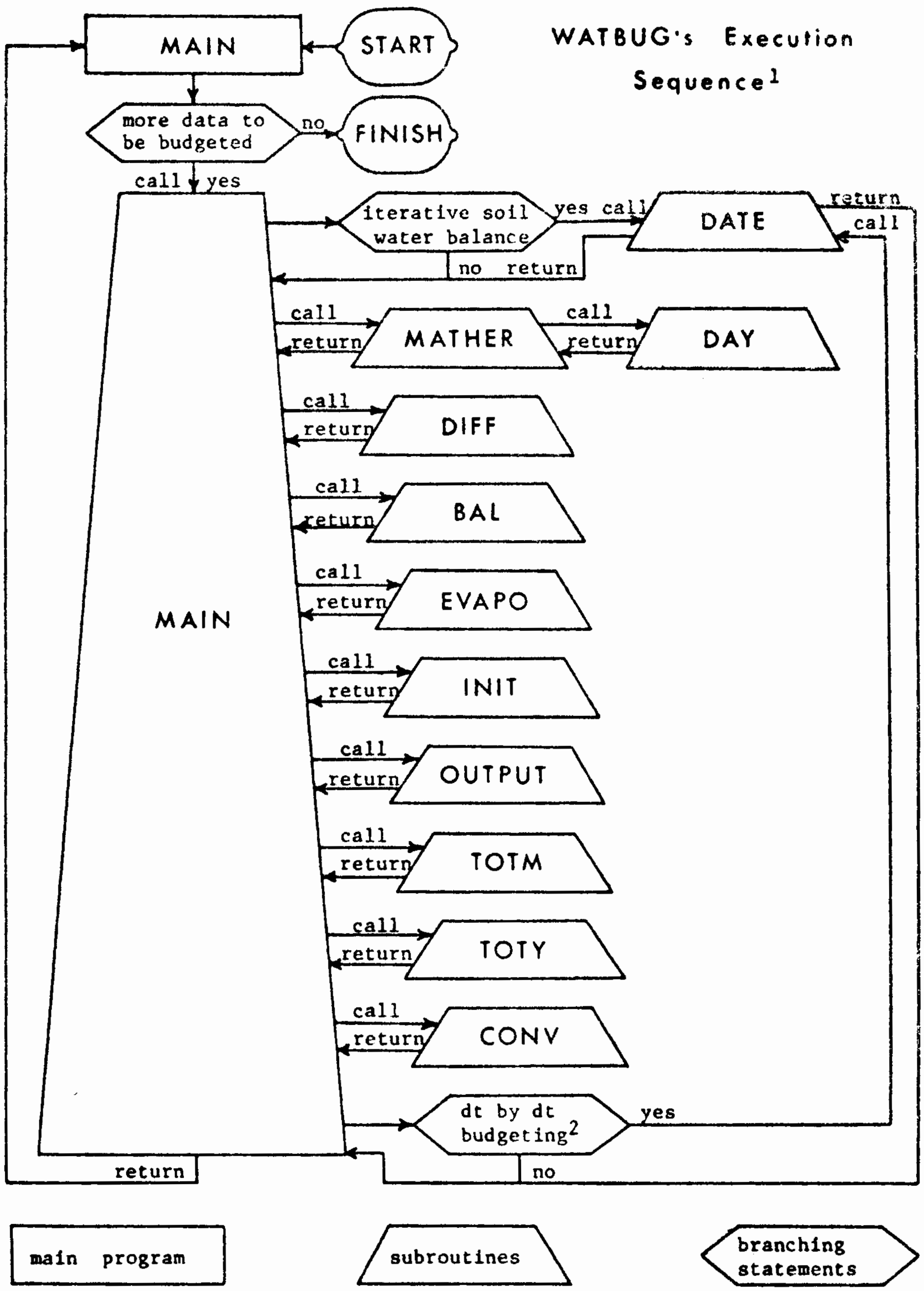
The main program performs no calculations but rather serves to (1) give initial dimensions to all arrays, (2) read the initial constants and semi-constants, (3) set the default options, (4) specify execution time array sizes, and (5) test whether or not calculations for more than a single station are to occur. Normally, the main program would control the sequence in which subroutines and functions are called; however, in this case, that task is relegated to subroutine MAIN.

The reason for this structure is that the size of the arrays used by the subroutines can be reduced from their initialized size by setting their dimensions equal to an argument which is specified in the main program and then transferred into the appropriate subroutines. This tends to lessen computational time and expense. All significant arrays are dimensioned in this fashion as M (which equals N + 1) where N is the number of months or days over which soil moisture balancing is to occur. These matrices have been given a maximum dimension of 481 which allows for a balancing of one year for daily budgets or 40 years for monthly budgets. If balancing over longer periods of time is required the initial dimensions of 481 will have to be increased.

Subroutine MAIN

Once all the initial constants and semi-constants have been initialized or read by the main program, they are transferred into subroutine MAIN which controls the sequence of calculations. In addition, subroutine MAIN (1) reads the necessary climatic information (air temperature and precipitation) and their format, (2) makes the appropriate unit translations to degrees Celsius and mm, (3) performs summations of daily and monthly values, (4) formats and writes the results onto file 6 (the line printer) and (5) keeps track of how many days or months have been processed. Subprogram MAIN is divided into four major sections. Each section controls a specific type of

Figure 2.



1. see text for descriptions of the subroutines and main program.
2. dt refers to month or day.

calculation sequence, i.e., monthly balancing, daily balancing, month-by-month calculations, or day-by-day calculations.

When the argument DT (time period index) is equal to zero and N (number of time periods over which balancing is to occur) is greater than one, monthly balancing of the soil water regime, N time periods long, is performed by iteration. (Note: N should be a multiple of 12 for monthly balancing to be legitimate.) These calculations begin at sequence number 167. When the balancing is complete and the results have been written, N is compared to NT (the total number of time periods over which the water budget is to be calculated). If NT is greater than N, subroutine MAIN transfers control to statement number 130 (sequence number 264) which represents the beginning of month-by-month calculations which are then performed on the remaining NT - N months. If N is not specified or is equal to one, balancing by iteration is skipped and month-by-month calculations are made exclusively. Following this, control is returned to the main program which either begins calculations for a new station or terminates processing.

When DT is not equal to zero, daily calculations are assumed. If N is greater than one, at the same time, subroutine MAIN undertakes daily balancing of the soil moisture budget for a period of N days beginning at statement number 60 (sequence number 207). (Note: N should be a multiple of 365 if daily balancing by iteration is specified.) Once N days have been balanced, N is compared with NT. When N is equal to NT control is returned to the main program. Otherwise, the subroutine goes to statement 190 (sequence number 340) in order to do day-by-day calculations for the residual NT - N days. If N equals one, all iteration is bypassed and calculations for NT days are done on a day-by-day basis beginning at statement 190. In either case, when a total of NT calculations have been made, control is transferred back to the main program.

Regardless of which of the four types of calculations are made, the sequence in which other subroutines are called by subroutine MAIN is essentially the same--with one minor exception--subroutine DATE. The sequence is subroutine DATE, MATHER, DAY (which is called by MATHER not MAIN), DIFF, BAL, EVAPO, INIT, OUTPUT, TOTM, TOTY, and CONV (figure 2). When monthly or daily balancing is being performed, DATE is called first, i.e., before MATHER, whereas when month-by-month or day-by-day calculations are made DATE is the last subroutine called. Each of these subroutines are described in the ensuing pages and, again, they are discussed in the order that they are called by subroutine MAIN.

Subroutine DATE

Subroutine DATE generates the day and month designations that will appear on the output. The subroutine requires the number of days over which balancing is to occur (N), the maximum array size for climatic variables (M), the initial or previous day designation (KD), the initial or previous month designation (KM), the time period index (DT) and the array DAYS which gives the number of days in each month.

It should be noted that no correction for leap year is made; therefore, users ought to be careful in interpreting daily or day-by-day output where a calculation for February 29 has been made. In such cases, the day designations on the output will be a day ahead of the day to which the day's water budget corresponds, i.e., for days following February 28. At the same time, daylength calculations made by subroutine DAY will be slightly incorrect although the error is insignificant. When daily balancing is performed on a leap year, N should be incremented by one in order to account for the 366-day year. The alternative to suffering the above is, of course, to remove February 29 and accept the small error introduced by that action. Regardless of which way the problem is handled the water budget values will be little influenced.

Depending on the type of calculations being made, DATE returns either the designations for the next day (DY(N) and KD) or month (MON(N) and KM) or, if balancing is occurring, an array of daily (DY) or monthly (MON) designations N values long. A couple of examples should illustrate the subprogram's function. If, for example, DATE were called just prior to the balancing of a daily water budget over a year, it would most likely receive the necessary arguments: N = 365, M = 366, KD = 1, KM = 1 and DT = 1. It would then return the arrays DY and MON each containing 365 day and month labels, respectively. On the other hand, if DATE were called just after a day-by-day calculation had been made for December 18, for instance, it would return values of KD = 19, KM = 12, DY(N) = 19 and MON(N) = 12 which would be used to label the next day's budget values.

When monthly calculations are made the day is held constant at 15--a representative day for the month. This value is important in that it is used in obtaining a daylength correction for the month via subroutines DAY and MATHER. When balancing is desired subroutine MATHER is called next in order to obtain potential evapotranspiration.

Subroutine MATHER

Subprogram MATHER is the hub of the algorithm as it calculates daily or monthly potential evapotranspiration according to the well-known Thornthwaite (1948) methodology. Since this discussion is presented to describe the procedures and use of the program, the author's choice and the accuracy of the Thornthwaite approach will not be examined as that has been done many times before (e.g., see McGuinness and Bordne, 1972). Suffice it to say, the approach has proven to be highly accurate in deriving monthly water budgets while requiring only a minimum amount of input information, i.e., air temperature and precipitation. The Thornthwaite method is less accurate in deriving daily potential evapotranspiration although such calculations can be useful in examining general within-month trends. Individual daily values should not be considered accurate, however. The subroutine requires a number of input parameters.

In particular, subroutine MATHER requires N, M, an array containing the air temperature data (T) in °C, the array DAYS, latitude (LAT), daylength (DL), as well as the day and month designations KD, KM, DT, MON and DY. When soil moisture balancing does not occur (i.e., N = 1), MATHER also requires an estimate of the station's heat index (HEAT). Using these variables, the subroutine calculates and returns to subroutine MAIN: an array of monthly heat indices (H), HEAT (Note: H and HEAT are only calculated during balancing), an empirical coefficient (A), an array of unadjusted potential evapotranspiration values (PE), and finally an array of adjusted potential evapotranspiration values (APE). When soil moisture balancing occurs, these arrays contain N climatic values. Otherwise, they are single climatic-valued arrays although it should be remembered that their actual size is N + 1. Calculations begin with the heat index when balancing is to be done.

Consistent with the Thornthwaite approach, a station's heat index is obtained from

$$\text{HEAT} = (12.0 \;/\; \text{XN}) \sum_{I=1}^{N} (\text{T(I)} \;/\; 5.0) ** 1.514 \qquad (3)$$

where T(I) is the mean daily or monthly temperature and XN (XN = N) is the number of days or months over which balancing (if specified) is to occur. The double star "**" is a FORTRAN operator indicating that the quantity to the left of the stars is to be raised to the power at the right of the stars. A single star "*", on the other hand, is the FORTRAN operator which specifies multiplication. Once again, when N refers to days, it should be a multiple of 365 or, if the time unit is months, N should be a multiple of 12 as HEAT is not defined for periods other than whole years. When computations are to be made on a day-by-day or month-by-month basis, HEAT cannot be correctly calculated and, therefore, must be supplied by the user. In such cases, the above-described calculation or HEAT will be circumvented. An empirically-derived exponent is next defined as

```
A = 6.75 / 10.0 ** 7.0 * HEAT ** 3.0

    - 7.71 / 10.0 ** 5.0 * HEAT ** 2.0

    + 1.79 / 10.0 ** 2.0 * HEAT + 0.49 .                          (4)
```

Unadjusted potential evapotranspiration is subsequently calculated as a function of T(I), HEAT, and A. Its form is

```
PE(I) = 16.0 * (10.0 * T(I) / HEAT) ** A .                 mm     (5)
```

The reader should be aware that units specified as "mm" can be either mm/day or mm/month depending upon the mode of analysis, i.e., daily or monthly. When T(I) $\geq$ 26.5, PE(I) is not estimated from the above but rather it becomes

```
PE(I) = - 415.85 + 32.24 * T(I) - 0.43 * T(I) ** 2.0       mm     (6)
```

where the above relationship was developed from, and explains virtually all the variance in, Thornthwaite's (1948: 94) correction table. When daily computations are made, PE(I) is divided by 30.

Following this, PE(I) is adjusted for variable day and month lengths. That is, adjusted potential evapotranspiration (APE(I)) is calculated as

APE(I) = PE(I) * (DAYS(KM + 1) / 30.0) * (DL / 12.0)   mm   (7)

where DAYS(KM + 1) is the number of days in month KM and DL is the daylength (hours). Daylength calculations are made by subroutine DAY (discussed next) while DAYS(KM + 1) is selected from the array DAYS. Again, when soil moisture balancing is being done the output arrays PE and APE are filled with N values. Otherwise, single climatic values are returned to subroutine MAIN.

Subroutine DAY

This subprogram estimates both the solar declination (DECD) and daylength (DL) although the former is not used again. Required input includes: the array DAYS, LAT, KD, KM and DT.

Although the approach taken is quite simple, as the anomalies of time are not considered, the maximum error possible in length of day estimates for mid-latitude locations is on the order of 10 to 15 minutes. Most estimates, however, are only off by a few minutes. Users are again reminded that no correction is made for leap year.

The first step is to calculate the number of days since January 1 and this value is stored as "SUM." SUM is then used to get the number of days since the last vernal equinox (DAYL). Declination (DECD) is then calculated from

DECD = 23.45 * SIN(DAYL / 365.0 * 6.2832)   deg   (8)

which was found to be a very good approximation to more detailed calculations based upon Kepler's law (Vowinckel and Orvig, 1972). In FORTRAN, trigonometric functions of a quantity or function X are expressed, for example, as SIN(X) which is equivalent to sin X. Some common ones are SIN(X), COS(X), ARCOS(X) and ATAN(X) for the tangent of X. Once DECD has been calculated and converted to radians (DECR), daylength can be calculated (Sellers, 1965).

When the sun is on the horizon the cosine of the zenith angle (CZ) should approach zero. Here, however, it is set slightly greater than zero in order to adjust the solar geometric equations which refer to the center of the solar disc. If this modification were not made, the cosine law would predict sunset when one-half the disc is still above the horizon. In general,

CZ = SIN(DECR) * SIN(ALAT) + COS(DECR) * COS(ALAT) * COS(H)   (9)

where ALAT is the latitude in radians, and H is the hour angle in radians. Since CZ is known at sunset and sunrise, H can be solved for by

H = ARCOS((CZ - SIN(DECR) * SIN(ALAT))/

(COS(DECR) * COS(ALAT))) .   rad   (10)

After H is calculated, it is translated into hours, i.e., daylength (DL) becomes

$$DL = 24.0 * H / 3.1416 . \qquad hr \quad (11)$$

Subroutine DAY then returns to subroutine MAIN with a value for DECD and DL.

Subroutine DIFF

Subroutine DIFF calculates the difference (D(I)) between adjusted evapotranspiration (APE(I)) and precipitation (P(I)) as well as the deficit (DEF(I)). The required input includes the precipitation and adjusted evapotranspiration arrays (P and APE) as well as their dimensions (M) and looping limit (N). The calculation is

$$D(I) = P(I) - APE(I). \qquad mm \quad (12)$$

When D(I) is less than zero, DEF(I) is set equal to D(I). Otherwise, DEF(I) equals zero. Subprogram DIFF then returns N new values of D(I) and DEF(I) to subroutine MAIN.

Subroutine BAL

Subroutine BAL is extremely important since it (1) iteratively balances the soil moisture budget for N months or days and/or (2) calculates month-by-month or day-by-day removal or addition of soil moisture. Required input includes: N, M, the array of differences between precipitation and adjusted potential evapotranspiration (D), the soil moisture field capacity (FC), an index which specifies which one of two soil moisture resistance (to evapotranspiration) functions is to be used (SM), DT, the array DAYS and KM. When month-by-month or day-by-day calculations are to be made without any previous balancing, an initial soil moisture storage value (ST(1)) must be included among the input values. The subroutine then calculates, and returns to subprogram MAIN, an array of soil moisture storage values (ST) as well as arrays of soil moisture surplus (SUR) and the difference in soil moisture storage between present and previous months/days (DST). Calculations are made for N time periods and all terms are in mm per time period. BAL begins with a test in order to determine whether balancing is to be performed or not.

When monthly or daily balancing is to take place, N will be greater than one, i.e., a multiple of 12 or 365, and BAL will begin its balancing computations by setting initial values. Conversely, if N equals one, month-by-month or day-by-day calculations are assumed and BAL will only make calculations for one time period each time it is called by subroutine MAIN. Although all the mathematical relationships are exactly the same whether balancing occurs or not, the balancing operations are fundamentally different; that is, balancing requires that soil moisture at the beginning of a balancing period must be equivalent to soil moisture at the end of the balancing period. In other words, the algorithm assumes that there will be no significant (net) increase or decrease in soil water over N time periods of computation. In order to accomplish this, a hypothetical time period (N + 1) is used.

On the first pass through the iteration procedures, soil moisture storage terms are continually adjusted until

$$ABS(ST(N + 1) + DST(1) - ST(1)) < 1.0. \quad mm \quad (13)$$

ABS(X) is a FORTRAN function of X equivalent to $|X|$. During the first set of iterations, however, DST(1) is equal to zero and so it has no impact. After the above relationship has been satisfied once, ST(1) is re-computed, beginning at statement 90 (sequence number 747), and DST(1) then becomes

$$DST(1) = ST(1) - ST(N + 1) \quad mm \quad (14)$$

which may no longer be zero. The soil moisture budget is then re-calculated over the N time periods until relationship (13) is again satisfied with DST(1) $\neq$ 0.0. After the second set of iterations, i.e., when the soil moisture budget has been "balanced," subroutine BAL returns to subroutine MAIN. Whether or not balancing is done, the ensuing computations are made.

Monthly calculations for the removal of soil moisture are made on an approximate day-by-day basis, i.e., assuming 30 days in a month. Soil moisture storage for each day (I) in the 30-day month is derived as

$$ST(I) = ST(I - 1) + D(I) * RATIO / 30.0 \quad mm/day \quad (15)$$

where RATIO = ST(I - 1) / FC. RATIO is the "normal" resistance of soil moisture to evapotranspiration used by Thornthwaite and Mather (Mather, 1974: 106. See curve C). Alternatively, when RATIO is greater than or equal to 0.7 <u>and</u> SM (set by the user) is greater than zero, ST(I) will be obtained from

$$ST(I) = ST(I - 1) + D(I) / 30.0. \quad mm/day \quad (16)$$

For a monthly or month-by-month budget the above computations for ST(I) are repeated 30 times. The last value of ST(I) is taken to be the new soil moisture for the month which is also subscripted "I". Daily or day-by-day computations are dissimilar in that only one computation is made for each day. The equations used, however, are identical to the above with the exception that the division by 30.0 is not made. In the event that D(I) is greater than zero (equation (12)), the removal steps are skipped and, beginning at statement 50 (sequence number 722), soil moisture is incremented by

$$ST(I) = ST(I - 1) + D(I). \quad mm \quad (17)$$

If, as a result, ST(I) is greater than or equal to FC, surplus is first calculated as

$$SUR(I) = ST(I) - FC \quad mm \quad (18)$$

and then ST(I) is set equal to FC. On the other hand, SUR(I) is set at zero when ST(I) is less than FC. Following these, a final calculation is made for DST(I) such that

$$DST(I) = ST(I) - ST(I - 1). \quad mm \quad (19)$$

It should be noted that unreasonable values of ST(I), i.e., less than 1.0, are set at 1.0, while SUR(I) is set at zero when D(I) is less than or equal to zero. Once the above computations are made, subroutine BAL either (1) returns to subprogram MAIN or (2) begins a new pass through the above relationships in order to balance the soil moisture budget.

Subroutine EVAPO

Subroutine EVAPO calculates the actual evapotranspiration and associated water deficit. The input used by EVAPO includes: N, M and the arrays D, APE, P, and DST. From these EVAPO generates the actual evapotranspiration array (AE) and a deficit array (DEF).

When D(I) is greater than or equal to zero, AE(I) is set equal to APE(I). Otherwise, AE(I) becomes

$$AE(I) = P(I) + ABS(DST(I)). \qquad \text{mm} \qquad (20)$$

And last, the deficit is calculated as

$$DEF(I) = APE(I) - AE(I) \qquad \text{mm} \qquad (21)$$

Subroutine INIT

Subroutine INIT merely re-sets (initializes) all the N values of any array (SUM) equal to zero. Its function, in this context, is to clear those arrays which are being used to keep track of monthly or yearly totals of APE, P, AE, DEF and SUR. Once an array has been initialized, INIT returns to subroutine MAIN.

Subroutine OUTPUT

Subprogram OUTPUT is used to fill specified elements of a single output array (OUT) with the values of each water budget variable. The values of OUT (after a minor modification to be described in the section on subroutine CONV) are then written onto paper. In this case, elements 2 through 10 of OUT are replaced with PE(L), APE(L), P(L), D(L), ST(L), DST(L), AE(L), DEF(L) and SUR(L) where L can refer to either of the subscripts I or N used by subprogram MAIN. Subroutine OUTPUT then returns to subroutine MAIN.

Subroutines TOTM and TOTY

Subroutines TOTM and TOTY are identical in form. The former is used to keep a running total of APE, P, AE, DEF and SUR over the month (for daily or day-by-day computations only). As these values are contained in elements 3, 4, 8, 9 and 10 of the array OUT (specified by the array IND), only OUT, IND and the array dimensions N and NN are required as input. The totals are stored in the array SUM. Subroutine TOTY performs an identical function on a yearly basis (for daily or monthly balancing and/or day-by-day or month-by-month computations). Once the appropriate elements of the array SUM have been incremented, these subprograms return to subroutine MAIN. At the end of a month or year, the array SUM is initialized with zeros by subroutine INIT.

Subroutine CONV

Subroutine CONV rounds off (converts) values of the output array (OUT), all except air temperature, to the nearest whole number before writing them. CONV requires an array (X) of dimension NUM and it rounds those elements of X from element MIN through element MAX. It should be understood that the array X is actually the array OUT. The rounded values are not used in computations but they are consistent and comparable with calculations done from tables by hand. After the specified values of X have been rounded to the nearest whole number, subroutine CONV returns to subroutine MAIN.

SELECTED REFERENCES

Carter, D.B., et al. (1973): Applications of the Water Budget in Physical Geography: An Annotated Bibliography, Publications in Climatology, Laboratory of Climatology, Vol. XXVI, No. 3.

Lee, R. (1978): Forest Microclimatology, New York: Columbia Univ. Press.

Mather, J.R. (1974): Climatology: Fundamentals and Applications, New York: McGraw-Hill Book Comp.

Mather, J.R. (1978): The Climatic Water Budget in Environmental Analysis, Lexington, Mass.: Lexington Books.

McGuinness, J.L. and E.F. Bordne (1972): A Comparison of Lysimeter-Derived Potential Evapotranspiration with Computed Values, U.S. Dept. of Agriculture Tech. Bull. No. 1452, Washington, D.C.

Sellers, W.D. (1965): Physical Climatology, Chicago: Univ. of Chicago Press.

Stone, L.A., J.C. Albrecht and G.A. Yoshioka (1971): Computer Programs for the Climatic Water Balance, Publications in Climatology, Laboratory of Climatology, Vol. XXIV, No. 3.

Terjung, W.H. (1976): Climatology for Geographers, Annals A.A.G., 66:199-222.

Thornthwaite, C.W. (1948): An Approach Toward a Rational Classification of Climate, Geog. Rev. 38:55-94.

Thornthwaite, C.W. (1961): The Task Ahead, Annals A.A.G. 51:345-356.

Thornthwaite, C.W. and J.R. Mather (1957): Instructions and Tables for Computing Potential Evapotranspiration and the Water Balance, Publications in Climatology, Laboratory of Climatology, Vol. X, No. 3.

Vowinckel, E and S. Orvig (1972): EBBA: An Energy Budget Programme, Publications in Meteorology, No. 105, McGill University, Montreal.

APPENDIX 1

PROGRAM WATBUG

```
      DIMENSION T(481),P(481),PE(481),APE(481),D(481),MON(481),LABEL(18)
     1          ,DAYS(13),H(481),AE(481),ST(481),DST(481),DEF(481),
     2          SUR(481),DY(481)
      DATA DAYS/0.0,31.0,28.0,31.0,30.0,31.0,30.0,31.0,31.0,30.0,31.0,
     1     30.0,31.0/
      REAL LAT,INDEX
C
C     *******************************************************************
C     *                                                                 *
C     *  THIS ALGORITHM WAS DEVELOPED BY C. WILLMOTT AT THE DEPARTMENT  *
C     *  OF GEOGRAPHY, UNIVERSITY OF DELAWARE IN 1978 IN ORDER TO       *
C     *  FACILITATE THE CALCULATION OF CLIMATIC WATER BUDGETS.  A       *
C     *  MINIMUM AMOUNT OF DATA (I.E., AIR TEMPERATURE, PRECIPITATION   *
C     *  AND A FEW INITIAL PARAMETERS) AND NO "LOOK-UP" TABLES ARE      *
C     *  REQUIRED AS ALL RELATIONSHIPS ARE EXPLICITLY SPECIFIED.  THE   *
C     *  PROGRAM WAS REFINED ON A BURROUGHS' B7700 ALTHOUGH STANDARD    *
C     *  (ANSI COMPATIBLE) FORTRAN WAS USED. IT SHOULD, THEREFORE, RUN *
C     *  WITH FEW OR NO MODIFICATIONS ON MOST MODERATE TO LARGE SIZED   *
C     *  MACHINES.  IF PROBLEMS ARE ENCOUNTERED, HOWEVER, USERS ARE     *
C     *  URGED TO CONTACT THE AUTHOR.                                   *
C     *                                                                 *
C     *******************************************************************
C
C     INITIAL PARAMETERS:
C
C     "LABEL" - 72 CHARACTER ALPHANUMERIC PROBLEM TITLE.
C
C     "N"    NUMBER OF MONTHS OR DAYS OVER WHICH SOIL MOISTURE
C            BALANCING IS TO OCCUR. IF N EQUALS ONE BALANCING
C            DOES NOT OCCUR AND ST(1) MUST BE SPECIFIED.
C
C     "NT" - TOTAL NUMBER OF MONTHS OR DAYS OVER WHICH THE WATER BUDGET
C             IS TO BE CALCULATED.
C
C     "KD" - THE DAY OF THE MONTH WHERE THE FIRST CALCULATIONS
C             ARE TO BEGIN. KD MUST BE LESS THAN OR EQUAL
```

```
C          TO THE NUMBER OF DAYS IN MONTH KM.
C
C   "KM" - THE FIRST MONTH OF CALCULATIONS.
C          KM MUST BE BETWEEN ZERO AND 12.
C
C   "KY" - THE FIRST YEAR OF CALCULATIONS. LAST TWO DIGITS ONLY.
C
C   "FC" - SOIL WATER HOLDING OR FIELD CAPACITY OF THE TOP (ONLY)
C          SOIL LAYER IN MM.
C
C   "SM" - DETERMINES A RESISTANCE FUNCTION OF SOIL WATER TO REMOVAL
C          BY EVAPOTRANSPIRATION. BLANK OR ZERO INDICATES THAT THE
C          AVAILABILITY OF SOIL MOISTURE TO EVAPOTRANSPIRATION WILL
C          DECLINE LINEARLY WITH THE RATIO OF ACTUAL TO POTENTIAL
C          MAXIMUM SOIL MOISTURE. ANY OTHER NUMERIC DESIGNATION
C          WILL RESULT IN AN ALTERNATIVE PROCEDURE WHERE MOISTURE IS
C          WITHDRAWN AT THE MAXIMUM RATE UNTIL THE ACTUAL/POTENTIAL
C          RATIO DROPS BELOW 0.7 AT WHICH TIME A LINEAR DECLINE IN
C          AVAILABILITY IS ASSUMED (SEE MATHER, 1974: 106 - CURVES
C          C AND G).
C
C   "LAT" - THE LATITUDE IN DEGREES.
C
C   "DT" - TIME DIFFERENTIAL. BLANK OR ZERO INDICATES MONTHLY
C          CALCULATIONS. ANY OTHER NUMBER CAUSES DAILY CALCULATIONS.
C
C   "TUNIT" - DESIGNATES THE UNITS OF AIR TEMPERATURE. 1.0 MEANS
C             THE RAW TEMPERATURE DATA ARE IN DEGREES FAHRENHEIT.
C             2.0 MEANS DEGREES KELVIN. ANY OTHER NUMERIC
C             DESIGNATION OR A BLANK MEANS DEGREES CELSIUS.
C
C   "PUNIT" - UNITS OF PRECIPITATION. 1.0 MEANS THE RAW DATA
C             ARE IN CM. 2.0 MEANS INCHES. 3.0 MEANS HUNDRETHS
C             OF AN INCH. OTHER DESIGNATIONS OR BLANKS MEANS MM.
C
C   "ST(1)"   ESTIMATED SOIL MOISTURE CONTENT OF THE TOP SOIL
```

```
C                 LAYER JUST PRIOR TO THE BEGINNING OF CALCULATIONS.
C                 ST(1) ONLY NEEDS TO BE SPECIFIED WHEN BALANCING
C                 IS NOT TO BE DONE (SEE NOTE BELOW). ST(1) IS IN MM.
C
C     "HEAT" - ESTIMATED HEAT INDEX. IT NEEDS TO BE SPECIFIED
C              ONLY WHEN SOIL MOISTURE BALANCING DOES NOT
C              OCCUR (NOTE: BALANCING SHOULD ONLY BE SPECIFIED FOR
C              PERIODS CONTAINING ONE OR MORE COMPLETE YEARS OF DATA).
C
C     "INDEX" - SHOULD BE SET GREATER THAN ZERO WHEN CALCULATIONS
C               FOR A SUBSEQUENT STATION ARE TO FOLLOW THESE.(NOTE:
C               CONTROL PARAMETERS AND DATA MUST BE INCLUDED
C               SEQUENTIALLY IN THE INPUT DATA SET FOR EACH STATION
C               THAT IS TO BE EVALUATED).
C
C     READ INITIAL PARAMETERS:
C
   10 READ(5,1000,END=20,ERR=20) LABEL,N,NT,KD,KM,KY,FC,SM,LAT,DT,
     1                           TUNIT,PUNIT,ST(1),HEAT,INDEX
      NNN= NNN + 1
C
C     ASSUMED PARAMETER VALUES, I.E., WHEN THEY ARE NOT SPECIFIED.
C
      IF (N.EQ.0) N= 1
      IF (NT.EQ.0) NT= N
      IF (KD.EQ.0.AND.DT.NE.0.0) KD= 1
      IF (KD.EQ.0.AND.DT.EQ.0.0) KD= 15
      IF (KM.EQ.0) KM= 1
C
C     SET THE ARRAY SIZES FOR CALCULATING A SOIL WATER BALANCE.
C
      M= N + 1
C
C     CALL THE MAIN SUBPROGRAM WHICH CONTROLS ALL CALCULATIONS.
C
      CALL MAIN(N,NT,M,FC,LAT,KD,KM,KY,DT,UY,HEAT,SM,T,P,PE,APE,D,DA,S,
```

```
     1             H,AE,ST,DST,DEF,SUR,MON,LABEL,TUNIT,PUNIT)
C
C     TEST TO SEE IF SUBSEQUENT STATIONS ARE TO BE EVALUATED.
C
      IF (INDEX.GT.0.0) GO TO 10
C
   20 CONTINUE
      STOP
 1000 FORMAT (18A4,/,5I5,9F5.0)
      END
C*********************************************************************
      SUBROUTINE MAIN(N,NT,M,FC,LAT,KD,KM,KY,DT,DY,HEAT,SM,T,P,PE,APE,D
     1         ,DAYS,H,AE,ST,DST,DEF,SUR,MON,LABEL,TUNIT,PUNIT)
      REAL LAT,T(M),P(M),PE(M),APE(M),D(M),DAYS(13),H(M),DY(M),AE(M),
     1     ST(M),DST(M),DEF(M),SUR(M),MON(M),FMT(18),LABEL(18)
      DIMENSION OUT(10),SUMM(5),SUMY(5)
      INTEGER IND(5)/3,4,8,9,10/
C
C     READ THE DATA FORMAT (FMT).
C
      READ(5,1000) FMT
      NNN= 0
C
C     UNIT CORRECTION FACTORS.
C
      C1= 1.0
      C2= 1.0
      FK= 0.0
      IF (TUNIT.EQ.1.0) FK= 32.0
      IF (TUNIT.EQ.1.0) C1= 5.0 / 9.0
      IF (TUNIT.EQ.2.0) FK= 273.16
      IF (PUNIT.EQ.1.0) C2= 10.0
      IF (PUNIT.EQ.2.0) C2= 25.4
      IF (PUNIT.EQ.3.0) C2= 0.254
C
C     READ AIR TEMPERATURE AND PRECIPITATION DATA.
```

```
C
      DO 10 I= 1,N
        READ(5,FMT,END=290,ERR=280) T(I),P(I)
        NNN= NNN + 1
C
C     UNIT TRANSLATIONS.
C
        T(I)= C1 * (T(I) - FK)
        P(I)= C2 * P(I)
C
   10 CONTINUE
C
      KY= KY + 1900
C
C     TEST FOR DAILY, MONTHLY, DAY BY DAY OR MONTH BY MONTH BUDGETING.
C
      IF (DT.NE.0.0.AND.N.EQ.1) GO TO 70
      IF (DT.EQ.0.0.AND.N.EQ.1) GO TO 20
      IF (DT.NE.0.0) GO TO 60
C***
C***  HERE FOR MONTHLY BALANCING.
C***
      CALL DATE(N,M,KD,KM,DY,MON,DT,DAYS)
      CALL MATHER(N,M,H,T,HEAT,A,PE,APE,DAYS,LAT,DL,KD,KM,DT,MON,DY)
      CALL DIFF(N,M,P,APE,D,DEF)
      CALL BAL(N,M,ST,D,FC,SM,SUR,DST,DT,DAYS,KM)
      CALL EVAPO(N,M,D,AE,APE,P,DST,DEF)
C
C     WRITE MONTHLY INPUT DATA AND RESULTS.
C
   20 CONTINUE
C
      WRITE(6,1010) LABEL
      WRITE(6,1020) N,NT,FC,LAT
C
      IF (N.EQ.1) GO TO 130
```

```
C
      I= 0
      GO TO 40
   30 CONTINUE
      CALL CONV(SUMY,5,1,5)
      WRITE(6,1030) SUMY
   40 CALL INIT(SUMY,5)
      WRITE(6,1040) KY
C
      KY= KY + 1
      WRITE(6,1050)
   50 I= I + 1
C
C     ROUND OFF TO NEAREST WHOLE NUMBER AND GET TOTALS BEFORE WRITING.
C
      CALL OUTPUT(PE,APE,P,D,ST,DST,AE,DEF,SUR,M,OUT,I)
      CALL TOTY(OUT,10,IND,SUMY,5)
      CALL CONV(OUT,10,2,10)
      OUT(1)= T(I)
C
      WRITE(6,1060) MON(I),OUT
      IF (I.LT.N.AND.MON(I).EQ.12) GO TO 30
      IF (I.EQ.N.AND.NT.GT.N) GO TO 130
      IF (I.EQ.N) GO TO 290
      GO TO 50
C
   60 CONTINUE
C***
C***  HERE FOR DAILY BALANCING.
C***
      CALL DATE(N,M,KD,KM,DY,MON,DT,DAYS)
      CALL MATHER(N,M,H,T,HEAT,A,PE,APE,DAYS,LAT,DL,KD,KM,DT,MON,DY)
      CALL DIFF(N,M,P,APE,D,DEF)
      CALL BAL(N,M,ST,D,FC,SM,SUR,DST,DT,DAYS,KM)
      CALL EVAPO(N,M,D,AE,APE,P,DST,DEF)
C
```

```
C     WRITE DAILY INPUT DATA AND RESULTS.
C
   70 CONTINUE
C
      WRITE(6,1010) LABEL
      WRITE(6,1070) N,NT,FC,LAT
C
      IF (N.EQ.1) GO TO 190
C
      I= 0
      GO TO 90
   80 CONTINUE
      CALL CONV(SUMM,5,1,5)
      CALL CONV(SUMY,5,1,5)
      WRITE(6,1080) SUMM
      WRITE(6,1030) SUMY
   90 CALL INIT(SUMY,5)
      WRITE(6,1040) KY
      KY= KY + 1
      GO TO 110
  100 CONTINUE
      CALL CONV(SUMM,5,1,5)
      WRITE(6,1080) SUMM
  110 CALL INIT(SUMM,5)
      WRITE(6,1090) MON(I+1)
      WRITE(6,1100)
  120 I= I + 1
      KD= DY(I)
      KM= MON(I)
C
C     ROUND OFF TO NEAREST WHOLE NUMBERS AND GET TOTALS BEFORE WRITING.
C
      CALL OUTPUT(PE,APE,P,D,ST,DST,AE,DEF,SUR,M,OUT,I)
      CALL TOTM(OUT,10,IND,SUMM,5)
      CALL TOTY(OUT,10,IND,SUMY,5)
      CALL CONV(OUT,10,2,10)
```

```
      OUT(1)= T(I)
C
      WRITE(6,1060) KD,OUT
      IF (I.LT.N.AND.MON(I).EQ.12.AND.DY(I).EQ.DAYS(KM+1)) GO TO 80
      IF (I.LT.N.AND.DY(I).EQ.DAYS(KM+1)) GO TO 100
      IF (I.EQ.N.AND.NT.LE.N) GO TO 290
      IF (I.EQ.N) GO TO 190
      GO TO 120
C***
C***  HERE FOR MONTH BY MONTH CALCULATIONS.
C***
  130 CONTINUE
      N1= 0
      IF (N.GT.1) N1= 1
C
C     GET THE INITIAL SOIL MOISTURE, MONTH AND DAY.
C
      IF (N.GT.1) ST(1)= ST(N)
      KD= 15
      IF (N.GT.1) KM= MON(N) + 1
      IF (KM.GE.13) KM= 1
      DY(1)= KD
      MON(1)= KM
C
C     SET INITIAL PARAMETERS.
C
      NN= NT - N
      IF (N.EQ.1) NN= NT
      IF (NNN.EQ.1) NNN= 0
      N= 1
      LL= 0
      M= N + 1
C
C     TEST FOR APPROPRIATE LABELS.
C
      IF (NNN.EQ.0) GO TO 150
```

```
      IF (KM.GT.1.AND.N1.EQ.1) GO TO 170
      IF (KM.GT.1) GO TO 160
C
  140 CONTINUE
C
C     WRITE LABELS, YEAR AND LAST YEAR'S TOTALS.
C
      CALL CONV(SUMY,5,1,5)
      WRITE(6,1030) SUMY
  150 CALL INIT(SUMY,5)
      WRITE(6,1040) KY
      KY= KY + 1
  160 CONTINUE
      WRITE(6,1050)
C
C     READ INPUT DATA AND CALL BUDGET SUBROUTINES.
C
  170 LL= LL + 1
      IF (NNN.EQ.0) GO TO 180
      READ(5,FMT,END=290,ERR=280) T(N),P(N)
      T(N)= C1 * (T(N) - FK)
      P(N)= C2 * P(N)
  180 NNN= NNN + 1
C
      CALL MATHER(N,M,H,T,HEAT,A,PE,APE,DAYS,LAT,DL,KD,KM,DT,MON,DY)
      CALL DIFF(N,M,P,APE,D,DEF)
      D(N+1)= D(N)
      CALL BAL(N,M,ST,D,FC,SM,SUR,DST,DT,DAYS,KM)
      ST(N)= ST(N+1)
      SUR(N)= SUR(N+1)
      DST(N)= DST(N+1)
      CALL EVAPO(N,M,D,AE,APE,P,DST,DEF)
C
C     ROUND OFF TO NEAREST WHOLE NUMBER AND GET TOTALS BEFORE WRITING.
C
      CALL OUTPUT(PE,APE,P,D,ST,DST,AE,DEF,SUR,M,OUT,N)
```

```
      CALL TOTY(OUT,10,IND,SUMY,5)
      CALL CONV(OUT,10,2,10)
      OUT(1)= T(N)
C
C     WRITE RESULTS AND GET THE NEXT MONTH.
C
      WRITE(6,1060) KM,OUT
      CALL DATE(N,M,KD,KM,DY,MON,DT,DAYS)
C
      IF (NN.EQ.LL) GO TO 290
      IF (KM.EQ.1) GO TO 140
      GO TO 170
C***
C***  HERE FOR DAY BY DAY CALCULATIONS.
C***
  190 CONTINUE
      N1= 0
      IF (N.GT.1) N1= 1
C
C     GET THE INITIAL SOIL MOISTURE, MONTH AND DAY.
C
      IF (N.GT.1) ST(1)= ST(N)
      IF (N.GT.1) KD= DY(N) + 1
      IF (N.GT.1) KM= MON(N)
      IF (KD.GT.DAYS(KM+1)) GO TO 200
      DY(1)= KD
      MON(1)= KM
      GO TO 210
  200 CONTINUE
      KM= KM + 1
      KD= 1
      IF (KM.GE.13) KM= 1
      DY(1)= KD
      MON(1)= KM
  210 CONTINUE
C
```

```
C     INITIALIZE PARAMETERS.
C
      NN= NT - N
      IF (N.EQ.1) NN= NT
      IF (NNN.EQ.1) NNN= 0
      N= 1
      L= 0
      M= N + 1
C
C     TEST FOR APPROPRIATE LABELS.
C
      IF (NNN.EQ.0) GO TO 230
      IF (KD.NE.1.AND.KM.NE.1.AND.N1.EQ.1) GO TO 260
      IF (KD.NE.1.OR.KM.NE.1) GO TO 240
C
  220 CONTINUE
C
C     WRITE LABELS,THE YEAR AND MONTH AND LAST YEAR'S OR MONTH'S TOTALS.
C
      CALL CONV(SUMM,5,1,5)
      CALL CONV(SUMY,5,1,5)
      WRITE(6,1080) SUMM
      WRITE(6,1030) SUMY
  230 CALL INIT(SUMY,5)
      WRITE(6,1040) KY
      KY= KY + 1
      GO TO 250
  240 CONTINUE
      CALL CONV(SUMM,5,1,5)
      WRITE(6,1080) SUMM
  250 CALL INIT(SUMM,5)
      WRITE(6,1090) KM
      WRITE(6,1100)
C
C     READ INPUT DATA AND CALL BUDGET SUBROUTINES.
C
```

```
  260 L= L + 1
      IF (NNN.EQ.0) GO TO 270
      READ(5,FMT,END=290,ERR=280) T(N),P(N)
      T(N)= C1 * (T(N) - FK)
      P(N)= C2 * P(N)
  270 NNN= NNN + 1
C
      CALL MATHER(N,M,H,T,HEAT,A,PE,APE,DAYS,LAT,DL,KD,KM,DT,MON,DY)
      CALL DIFF(N,M,P,APE,D,DEF)
      D(N+1)= D(N)
      CALL BAL(N,M,ST,D,FC,SM,SUR,DST,DT,DAYS,KM)
      ST(N)= ST(N+1)
      SUR(N)= SUR(N+1)
      DST(N)= DST(N+1)
      CALL EVAPO(N,M,D,AE,APE,P,DST,DEF)
C
C     ROUND OFF TO NEAREST WHOLE NUMBER AND GET TOTALS BEFORE WRITING.
C
      CALL OUTPUT(PE,APE,P,D,ST,DST,AE,DEF,SUR,M,OUT,N)
      CALL TOTM(OUT,10,IND,SUMM,5)
      CALL TOTY(OUT,10,IND,SUMY,5)
      CALL CONV(OUT,10,2,10)
      OUT(1)= T(N)
C
C     WRITE RESULTS AND GET A NEW DATE.
C
      WRITE(6,1060) KD,OUT
      CALL DATE(N,M,KD,KM,DY,MON,DT,DAYS)
C
      IF (NN.EQ.L) GO TO 290
      IF (KM.EQ.1.AND.KD.EQ.1) GO TO 220
      IF (KD.EQ.1) GO TO 240
      GO TO 260
C
C     WRITE FINAL MESSAGES AND TOTALS.
C
```

```
 280 CONTINUE
     WRITE(6,1110) NNN + 1
     GO TO 300
 290 CONTINUE
     IF (DT.NE.0.0) WRITE(6,1080) SUMM
     WRITE(6,1030) SUMY
     WRITE(6,1120) NNN
 300 CONTINUE
     WRITE(6,1130)
     RETURN
1000 FORMAT (18A4)
1010 FORMAT ('0',18A4)
1020 FORMAT (//,'  NO. OF MONTHS OVER WHICH BALANCING OCCURS IS ',I5,//
    *,'  TOTAL NO. OF MONTHS EVALUATED IS',I5,//,
    *'  SOIL MOISTURE CAPACITY IS ',F5.1,' MM ',//,
    *'  LATITUDE IS ',F4.1)

1030 FORMAT (/,'  YEARLY TOTALS',3X,2I6,18X,3I6)
1040 FORMAT (///,'  YEAR IS ',I4)
1050 FORMAT (//,'  MO     TEMP    UPE    APE   PREC  DIFF     ST    DST     AE
    *   DEF   SURP',/)
1060 FORMAT (I4,2X,F6.1,9I6)
1070 FORMAT (//,'  NO. OF DAYS OVER WHICH BALANCING OCCURS IS ',I5,//,
    *'  TOTAL NO. OF DAYS EVALUATED IS ',I5,//,
    *'  SOIL MOISTURE CAPACITY IS ',F5.1,' MM ',//,
    *'  LATITUDE IS ',F4.1)
1080 FORMAT (/,'  MONTHLY TOTALS',2X,2I6,18X,3I6)
1090 FORMAT (//,'  MONTH IS ',I2)
1100 FORMAT (//,'  DY     TEMP    UPE    APE   PREC  DIFF     ST    DST     AE
    *   DEF   SURP',/)
1110 FORMAT (///,'  ERROR ENCOUNTERED IN THE DATA AT RECORD ',I5)
1120 FORMAT (///,'  PROCESSING TERMINATED AFTER RECORD ',I5)
1130 FORMAT ('1')
     END
C**********************************************************************
     SUBROUTINE DATE(N,M,KD,KM,DY,MON,DT,DAYS)
```

```
      REAL DY(M),MON(M),DAYS(13)
C
C     GENERATE DAY AND MONTH DESIGNATIONS.
C
C     TEST FOR MONTHLY, DAILY, MONTH BY MONTH OR DAY BY DAY
C     CALCULATIONS.
C
      IF (DT.NE.0.0.AND.N.EQ.1) GO TO 60
      IF (DT.EQ.0.0.AND.N.EQ.1) GO TO 50
      IF (DT.NE.0.0) GO TO 20
C
C     MONTHLY CALCULATIONS.
C
      KD= 15
      KM= KM - 1
      IF (KM.LE.0) KM= 0
      DO 10 I= 1,N
        KM= KM + 1
        IF (KM.GE.13) KM= 1
        MON(I)= KM
        DY(I)= KD
   10 CONTINUE
      GO TO 80
C
C     DAILY CALCULATIONS.
C
   20 CONTINUE
      K= 1
      KD= KD - 1
      IF (KD.LE.0) KD= 0
      IF (KD.GT.0) K= KD
      KM= KM - 1
      IF (KM.LE.0) KM= 0
      J= 0
   30 KM= KM + 1
      IF (KD.GT.DAYS(KM+1)) KD= 1
```

```
      IF (KM.GE.13) KM= 1
      DO 40 I= K,DAYS(KM+1)
        J= J + 1
        MON(J)= KM
        DY(J)= I
   40 CONTINUE
      IF (J.GE.N) GO TO 80
      K= 1
      GO TO 30
C
C     MONTH BY MONTH CALCULATIONS.
C
   50 CONTINUE
      KD= 15
      KM= KM + 1
      IF (KM.GE.13) KM= 1
      DY(N)= KD
      MON(N)= KM
      GO TO 80
C
C     DAY BY DAY CALCULATIONS.
C
   60 CONTINUE
      KD= KD + 1
      DY(N)= KD
      MON(N)= KM
      IF (KD.GT.DAYS(KM+1)) GO TO 70
      GO TO 80
   70 CONTINUE
      KM= KM + 1
      KD= 1
      IF (KM.GE.13) KM= 1
      DY(N)= KD
      MON(N)= KM
   80 CONTINUE
      RETURN
```

```
      END
C***********************************************************************
      SUBROUTINE MATHER(N,M,H,T,HEAT,A,PE,APE,DAYS,LAT,DL,KD,KM,DT,MON,
     1       DY)
      REAL LAT,H(M),T(M),PE(M),APE(M),DAYS(13),MON(M),DY(M)
C
C     CALCULATE POTENTIAL EVAPOTRANSPIRATION.
C
C     WHEN LAT IS GREATER THAN 50 DEGS, THE DAYLENGTH CORRECTION
C     REMAINS EQUAL TO THAT FOR 50 DEGS. ALAT IS, THEREFORE,
C     USED AS THE ARGUMENT FOR SUBROUTINE DAY.
C
      ALAT= LAT
      IF (ALAT.GE.50.0) ALAT= 50.0
C
C     CALCULATE THE HEAT INDEX DURING BALANCING ON THE FIRST CALL
C     OF MATHER. ON THE SECOND CALL, GO DIRECTLY TO "PE" CALCULATIONS.
C
      IF (N.LT.12) GO TO 40
      IF (N.LT.365.AND.DT.NE.0.0) GO TO 40
C
      XN= N
      HEAT= 0.0
      DO 30 I= 1,N
        IF (T(I).LE.0.0) GO TO 10
        H(I)= (T(I) / 5.0) ** 1.514
        GO TO 20
   10   H(I)= 0.0
   20   CONTINUE
        HEAT= HEAT + H(I)
   30 CONTINUE
C
C     ADJUST "HEAT" FOR BUDGETS GREATER THAN
C     A YEAR.
C
      HEAT= HEAT * 12.0 / XN
```

```
C
C     NOTE: "A" IS AN EMPIRICALLY
C     DERIVED EXPONENT BASED UPON "HEAT".
C
   40 CONTINUE
      A= 6.75 / 10.0 ** 7.0 * HEAT ** 3.0-7.71 / 10.0 ** 5.0 * HEAT **
     1   2.0+1.79 / 10.0 ** 2.0 * HEAT+0.49
C
C     GET INITIAL MONTHLY PE, I.E. BASED UPON 30 DAYS IN
C     A MONTH AND 12HOURS IN A DAY.
C     NOTE: PE(I) AND APE(I) ARE CALCULATED IN MM / MONTH
C     OR DAY.
C
      DO 70 I= 1,N
        IF (T(I).LE.0.0) GO TO 50
        PE(I)= 16.0 * (10.0 * T(I) / HEAT) ** A
C
C     CORRECT FOR TEMPERATURES GREATER THAN 26.5 DEG C.
C     SEE THORNTHWAITE (1948) FOR EXPLANATION.
C
        IF (T(I).GE.26.5) PE(I)= (-41.58547 + 3.22441 * T(I)-0.04325 *
     1                          T(I) ** 2.0) * 10.0
C
C     CORRECT PE(I) FOR DAILY CALCULATIONS.
C
        IF (DT.NE.0.0) PE(I)= PE(I) / 30.0
        GO TO 60
   50   PE(I)= 0.0
   60   CONTINUE
        KD= DY(I)
        KM= MON(I)
C
C     ADJUST PE FOR DAYLENGTH AND THE NUMBER OF DAYS IN A MONTH.
C
        CALL DAY(DAYS,ALAT,KD,KM,DT,DECD,DL)
        APE(I)= PE(I) * (DAYS(KM + 1) / 30.0) * (DL / 12.0)
```

```
   70 CONTINUE
      RETURN
      END
C*********************************************************************
      SUBROUTINE DAY(DAYS,LAT,KD,KM,DT,DECD,DL)
      REAL DAYS(13)
      REAL LAT
C
C     CALCULATE THE NUMBER OF HOURS IN A
C     DAY AND THE SOLAR DECLINATION ASSOCIATED WITH THAT
C     DAY. THE INPUT REQUIRED INCLUDES: THE MONTH (KM),
C     THE DAY (KD) AND THE LATITUDE (LAT).
C
      X= 0.0
      DO 10 I= 1,KM
        X= X + DAYS(I)
   10 CONTINUE
      SUM= X + KD
C
C     GET THE NUMBER OF DAYS SINCE THE VERNAL EQUINOX (MARCH 21).
C
      DAYL= SUM - 80.0
      IF (DAYL.LE.0.0) DAYL= 285.0 + SUM
C
C     CALCULATE THE DECLINATION.
C
      DECD= 23.45 * SIN(DAYL / 365.0 * 6.2832)
      DECR= DECD * 0.017453
C
C     CALCULATE THE NUMBER OF HOURS OF DAYLIGHT CORRESPONDING
C     TO DAY KD AND MONTH KM (SEE SELLERS, 1965).
C
      CZ= COS(1.5708 + 0.01745 * (100.0 / 60.0))
      ALAT= LAT * 0.017453
      XX= COS(DECR) * COS(ALAT)
      IF (XX.LE.0.0) GO TO 20
```

```
      CSH= (CZ - SIN(DECR) * SIN(ALAT)) / XX
      H= ARCOS(CSH)
      DL= 24.0 * H / 3.1416
      GO TO 30
C
C     ERROR MESSAGE - DIVIDE BY ZERO OR LESS.
C
   20 WRITE(6,1000)
C
   30 CONTINUE
      RETURN
 1000 FORMAT ('0',' ERROR - DIVIDE BY ZERO OR LESS - LAT. ',//,
     *'  OR THE DECLINATION IS PROBABLY INCORRECT ')
      END
C************************************************************************
      SUBROUTINE DIFF(N,M,P,APE,D,DEF)
      REAL P(M),APE(M),D(M),DEF(M)
C
C     COMPARE APE(I) WITH PRECIPITATION (P(I)).
C
      DO 10 I= 1,N
        D(I)= P(I) - APE(I)
        DEF(I)= 0.0
        IF (D(I).LT.0.0) DEF(I)= D(I)
   10 CONTINUE
      RETURN
      END
C************************************************************************
      SUBROUTINE BAL(N,M,ST,D,FC,SM,SUR,DST,DT,DAYS,KM)
      REAL ST(M),D(M),SUR(M),DST(M),DAYS(13)
C
C     ITERATE FOR SOIL MOISTURE TERMS THAT BALANCE
C     THE WATER BUDGET ON THE FIRST CALL OF BAL. ON A SECOND CALL
C     AND/OR WHEN N IS ONE, BAL DOES MONTH BY MONTH OR DAY BY
C     DAY SOIL MOISTURE CALCULATIONS.
C
```

```
      IF (N.EQ.1) GO TO 10
C
      ST(N + 1)= 0.0
      ST(1)= 300.0
      DST(1)= 0.0
      K= 0
      Z= 0.0
   10 CONTINUE
      DO 80 I= 2,N + 1
        IF (D(I).GE.0.0) GO TO 50
C
C     TEST FOR MONTHLY OR DAILY WITHDRAWAL.
C
        IF (DT.NE.0.0) GO TO 30
C
C     WITHDRAWAL FOR MONTHLY BUDGETS (NOTE: THIS IS DONE ON
C     AN APPROXIMATE DAY BY DAY BASIS).
C
        X1= ST(I - 1)
        DO 20 J= 1,30
          RATIO= ST(I - 1) / FC
          ST(I)= ST(I - 1) + D(I) * RATIO / 30.0
          IF (RATIO.GE.0.7.AND.SM.GT.0.0) ST(I)= ST(I - 1) + D(I) / 30.0
          ST(I - 1)= ST(I)
   20   CONTINUE
        ST(I - 1)= X1
        GO TO 40
   30   CONTINUE
C
C     WITHDRAWAL FOR DAILY BUDGETS.
C
        ST(I)= ST(I-1) + ST(I-1) / FC * D(I)
        IF ((ST(I-1) / FC).GE.0.7.AND.SM.GT.0.0) ST(I)= ST(I-1) + D(I)
C
   40   CONTINUE
        IF (ST(I).LE.1.0) ST(I)= 1.0
```

```
      GO TO 70
 50   CONTINUE
      ST(I)= ST(I-1) + D(I)
      IF (ST(I).GE.FC) GO TO 60
      SUR(I)= 0.0
      GO TO 70
 60   SUR(I)= ST(I) - FC
      ST(I)= FC
 70   CONTINUE
      DST(I)= ST(I) - ST(I-1)
      IF (D(I).LE.0.0) SUR(I)= 0.0
 80 CONTINUE
C
    IF (N.EQ.1) GO TO 160
C
    K= K + 1
C
C   TESTS FOR BALANCES
C
    IF (K.GT.50) GO TO 160
    XX= ABS(ST(N + 1) + DST(1) - ST(1))
    IF (XX.LT.1.0.AND.Z.EQ.1.0) GO TO 160
    IF (XX.LT.1.0) GO TO 90
    ST(1)= ST(N + 1) + DST(1)
    GO TO 10
C
 90 CONTINUE
    IF (D(1).GE.0.0) GO TO 130
    IF (DT.NE.0.0) GO TO 110
C
C   BALANCE FOR THE FIRST MONTH
C
    X2= ST(N + 1)
    DO 100 L= 1,30
      RATIO= ST(N + 1) / FC
      ST(1)= ST(N + 1) + D(1) * RATIO / 30.0
```

```
      IF (RATIO.GE.0.7.AND.SM.GT.0.0) ST(1)= ST(N + 1) + D(1) / 30.0
        ST(N + 1)= ST(1)
  100 CONTINUE
      ST(N + 1)= X2
      GO TO 120
  110 CONTINUE
C
C     BALANCE FOR THE FIRST DAY
C
      ST(1)= ST(N + 1) + ST(N + 1) / FC * D(1)
      IF ((ST(N + 1) / FC).GE.0.7.AND.SM.GT.0.0) ST(1)= ST(N + 1) + D(1)
C
  120 CONTINUE
      IF (ST(1).LE.1.0) ST(1)= 1.0
      GO TO 150
  130 CONTINUE
      ST(1)= ST(N + 1) + D(1)
      IF (ST(1).GE.FC) GO TO 140
      SUR(1)= 0.0
      GO TO 150
  140 SUR(1)= ST(1) - FC
      ST(1)= FC
  150 CONTINUE
      DST(1)= ST(1) - ST(N + 1)
      IF (D(1).LE.0.0) SUR(1)= 0.0
      Z= 1.0
      GO TO 10
C
  160 CONTINUE
C
      RETURN
      END
C*************************************************************************
      SUBROUTINE EVAPO(N,M,D,AE,APE,P,DST,DEF)
      REAL D(M),AE(M),APE(M),P(M),DST(M),DEF(M)
C
```

```
C       CALCULATE ACTUAL EVAPOTRANSPIRATION AND DEFICIT.
C
        DO 10 I= 1,N
          AE(I)= APE(I)
          IF (D(I).LT.0.0) AE(I)= P(I) + ABS(DST(I))
          DEF(I)= APE(I) - AE(I)
   10   CONTINUE
        RETURN
        END
C***********************************************************************
        SUBROUTINE INIT(SUM,N)
        DIMENSION SUM(N)
C
C       INITIALIZE ARRAY "SUM" WITH ZEROS.
C
        I= 0
   10   I= I + 1
        SUM(I)= 0.0
        IF (I.LT.N) GO TO 10
        RETURN
        END
C***********************************************************************
        SUBROUTINE OUTPUT(PE,APE,P,D,ST,DST,AE,DEF,SUR,M,OUT,L)
        DIMENSION PE(M),APE(M),P(M),D(M),ST(M),DST(M),AE(M),DEF(M),SUR(M),
     1            OUT(10)
C
C       FILL THE OUTPUT ARRAY "OUT".
C
        OUT(2)= PE(L)
        OUT(3)= APE(L)
        OUT(4)= P(L)
        OUT(5)= D(L)
        OUT(6)= ST(L)
        OUT(7)= DST(L)
        OUT(8)= AE(L)
        OUT(9)= DEF(L)
```

```
      OUT(10)= SUR(L)
      RETURN
      END
C***********************************************************************
      SUBROUTINE TOTM(X,N,IND,SUM,NN)
      DIMENSION SUM(NN),IND(NN),X(N)
C
C     SUM VALUES OF "X" SPECIFIED BY "IND" OVER THE MONTH.
C
      I= 0
   10 I= I + 1
      J= IND(I)
      SUM(I)= SUM(I) + X(J)
      IF (I.LT.NN) GO TO 10
      RETURN
      END
C***********************************************************************
      SUBROUTINE TOTY(X,N,IND,SUM,NN)
      DIMENSION SUM(NN),IND(NN),X(N)
C
C     SUM VALUES OF "X" SPECIFIED BY "IND" OVER THE YEAR.
C
      I= 0
   10 I= I + 1
      J= IND(I)
      SUM(I)= SUM(I) + X(J)
      IF (I.LT.NN) GO TO 10
      RETURN
      END
C***********************************************************************
      SUBROUTINE CONV(X,NUM,MIN,MAX)
      DIMENSION X(NUM)
C
C     ROUND X(I) OFF TO NEAREST WHOLE NUMBER
C
      I= MIN - 1
```

```
10 I= I + 1
   IF (X(I).EQ.0.0) GO TO 20
   Y= ABS(X(I))
   J= X(I) / Y
   K= Y + 0.5
   X(I)= K * J
20 IF (I.LT.MAX) GO TO 10
   RETURN
   END
```

APPENDIX 2

SAMPLE INPUT

```
SAMPLE MONTHLY WATER BUDGET ONE YEAR LONG. DATA FROM SEABROOK, N.J..
  12   12     1     1    77 300.0      40.0
(2F10.5)
      0.9       87.0
      1.2       93.0
      5.9      102.0
     11.3       88.0
     17.5       92.0
     22.3       91.0
     24.7      112.0
     23.7      113.0
     20.2       82.0
     14.0       85.0
      7.6       70.0
      2.3       93.0
```

```
SAMPLE DAILY WATER BUDGET ONE MONTH LONG. DATA FROM SEABROOK, N. H.
   1    30     1     6    53300.0         40.0  1.0              300.0 58.2
(2F10.0)
      11.7      11.0
      12.8       1.0
      17.8       0.0
      18.9       0.0
      23.3       0.0
      25.0      15.0
      25.6       1.0
      22.8       0.0
      22.2       0.0
      23.9       0.0
      20.0       0.0
      17.2       0.0
      18.9       1.0
      16.7      14.0
      13.9       0.0
      17.8       0.0
      20.0       0.0
      18.9       0.0
      21.1       0.0
      24.4       0.0
      26.1       0.0
      26.7       0.0
      25.0       0.0
      22.2       0.0
      20.0       0.0
      22.8       0.0
      27.2       0.0
      26.7       0.0
      26.1       2.0
      25.6       1.0
```

APPENDIX 3

SAMPLE OUTPUT

SAMPLE MONTHLY WATER BUDGET ONE YEAR LONG. DATA FROM SEABROOK, N.J..

NO. OF MONTHS OVER WHICH BALANCING OCCURS IS 12

TOTAL NO. OF MONTHS EVALUATED IS 12

SOIL MOISTURE CAPACITY IS 300.0 MM

LATITUDE IS 40.0

YEAR IS 1977

MO	TEMP	UPE	APE	PREC	DIFF	ST	DST	AE	DEF	SURP
1	0.9	1	1	87	86	300	0	1	0	86
2	1.2	2	1	93	92	300	0	1	0	92
3	5.9	16	17	102	85	300	0	17	0	85
4	11.3	41	45	88	43	300	0	45	0	43
5	17.5	75	94	92	-2	298	-2	94	0	0
6	22.3	105	133	91	-42	259	-39	130	3	0
7	24.7	122	156	112	-44	223	-35	147	9	0
8	23.7	115	137	113	-24	206	-17	130	7	0
9	20.2	92	96	82	-14	197	-9	91	4	0
10	14.0	55	53	85	32	229	32	53	0	0
11	7.6	23	19	70	51	280	51	19	0	0
12	2.3	4	4	93	89	300	20	4	0	69
YEARLY TOTALS			756	1108				734	23	374

PROCESSING TERMINATED AFTER RECORD 12

SAMPLE DAILY WATER BUDGET ONE MONTH LONG. DATA FROM SEABROOK, N.J..

NO. OF DAYS OVER WHICH BALANCING OCCURS IS 1

TOTAL NO. OF DAYS EVALUATED IS 30

SOIL MOISTURE CAPACITY IS 300.0 MM

LATITUDE IS 40.0

YEAR IS 1953

MONTH IS 6

DY	TEMP	UPE	APE	PREC	DIFF	ST	DST	AE	DEF	SURP
1	11.7	1	2	11	9	300	0	2	0	9
2	12.8	2	2	1	-1	299	-1	2	0	0
3	17.8	3	3	0	-3	296	-3	3	0	0
4	18.9	3	3	0	-3	292	-3	3	0	0
5	23.3	4	5	0	-5	288	-5	5	0	0
6	25.0	4	5	15	10	298	10	5	0	0
7	25.6	4	5	1	-4	293	-4	5	0	0
8	22.8	4	5	0	-5	289	-4	4	0	0
9	22.2	3	4	0	-4	285	-4	4	0	0
10	23.9	4	5	0	-5	280	-5	5	0	0
11	20.0	3	4	0	-4	276	-4	4	0	0
12	17.2	2	3	0	-3	274	-3	3	0	0
13	18.9	3	4	1	-3	271	-2	3	0	0
14	16.7	2	3	14	11	282	11	3	0	0
15	13.9	2	2	0	-2	280	-2	2	0	0

16	17.8	3	3	0	-3	277	-3	3	0	0
17	20.0	3	4	0	-4	274	-4	4	0	0
18	18.9	3	4	0	-4	270	-3	3	0	0
19	21.1	3	4	0	-4	267	-4	4	0	0
20	24.4	4	5	0	-5	262	-4	4	1	0
21	26.1	4	6	0	-6	257	-5	5	1	0
22	26.7	5	6	0	-6	252	-5	5	1	0
23	25.0	4	5	0	-5	248	-4	4	1	0
24	22.2	3	4	0	-4	244	-4	4	1	0
25	20.0	3	4	0	-4	241	-3	3	1	0
26	22.8	4	5	0	-5	238	-4	4	1	0
27	27.2	5	6	0	-6	233	-5	5	1	0
28	26.7	5	6	0	-6	228	-4	4	1	0
29	26.1	4	6	2	-4	226	-3	5	1	0
30	25.6	4	5	1	-4	222	-3	4	1	0
MONTHLY TOTALS			127	46				114	13	9
YEARLY TOTALS			127	46				114	13	9

PROCESSING TERMINATED AFTER RECORD 30