SIMULATION REPORT CABO-TT No. 7

# MANUAL FOR GENERATION OF DAILY WEATHER DATA

Iwan Supit

Sim. Rep. CABO-TT No. 7

Wageningen, 1986

SIMULATION REPORTS CABO-TT

Reports published in this series

1. UNGAR, E. & H. van KEULEN:

   FORTRAN version of the simulation model ARID CROP. 1982, 39 pp.

2. CORDOVA, J., F.W.T. PENNING de VRIES & H.H. van LAAR:

   Modeling of crop production: evaluation of an international post graduate course held at IDEA, November 1982. 1983, 23 pp.

   Modelos matematics de produccion de cultivos: evaluacion del curso internacional realizado en IDEA, en Noviembre de 1982. 1983, 27 pp.

3. MARLETTO, V. & H. van KEULEN:

   Winter wheat experiments in The Netherlands and Italy analysed by the SUCROS model. 1984, 61 pp.

4. GENG, S., F.W.T. PENNING de VRIES & I. SUPIT:

   Analysis and simulation of wheather variables - part I: rain and wind in Wageningen. 1985, 55 pp.

5. GENG, S., F.W.T. PENNING de VRIES & I. SUPIT:

   Analysis and simulation of weather variables - part II: temperature and solar radiation. 1985, 74 pp.

6. BENSCHOP, M.:

   TUCROS, een simulatiemodel voor de tulpecultivar "Apeldoorn". 1985.

Simulation Reports CABO-TT

Simulation Reports is a series of supplementary information concerning simulation models in agriculture which have been published elsewhere. Knowledge of those publications will in general be a prerequisite for studying the supplementary information in these reports.

Simulation Reports may contain improvements of simulation models, new applications, or translations of the programs into other computer languages.

Simulation Reports are issued by CABO-TT and available on request.

Announcements of Reports in this series will be issued regularly. Addresses of those who are interested in the announcements will be put on the mailing list on request.

CABO-TT
Bornsesteeg 65
P.O. Box 14
6700 AA WAGENINGEN
The NETHERLANDS

CENTRUM VOOR AGROBIOLOGISCH ONDERZOEK (CABO)
Centre for Agrobiological Research

VAKGROEP THEORETISCHE TEELTKUNDE (TT), Landbouwhogeschool
Department of Theoretical Production Ecology, Agricultural University.

CONTENTS

ABSTRACT

The program SIMWTH is based on the Simulation Reports by Shu Geng etal (lit.2 and 3), which on their turn are based on the article of Richardson and Wright (lit.1). It combines the main subroutines TWGEN and TEST. Subroutine TWGEN derives the model parameters required for the weather data generation from a historical data set. The subroutine TEST generates the daily weather variables rain, maximum temperature, minimum temperature, solar radiation and windspeed. The humidity is not independently generated but is set equal to the saturated vapour pressure at the minimum temperature. The historical data set has to contain rainfall, solar radiation, windspeed, minimum and maximum temperature data.

In subroutine TWGEN a first order, two stage Markov chain has been used to describe the probabilities whether a day would be dry or wet. Estimates of these probalities are usually obtained from analysis of historical data. Furthermore it is assumed that the amount of rainfall on a wet day can be adequately described by a two parameter Gamma distribution function. In addition a Gamma function is also used to describe the windspeed. The other variables, solar radiation, maximum and minimum temperature are supposed to be conditional multivariate normal random variables conditioned on the precipitation. The mean and the standard deviation are described by Fourier series (lit.3).

Subroutine TEST uses the parameters, calculated in subroutine TWGEN, to generate the previous mentioned weather variables.

This program is developped on an IBM PC/AT with a mathematical co-processor 80287 and 512 KB Ram and is written in Fortran. Also a VAX version of this program exists.

## Introduction

The weather has a great impact on plant growth and development. Windspeed and humidity are important driving forces behind the evaporation. The radiation chemically trapped by plants in photosynthesis not only sustains the biomass of the plant itself, but also of all living creatures. Temperature controls the rate of enzymatic reactions, the solubility of gases in a cell, influence the availability and absorption of minerals and also influence the uptake of the percolated rainfall. Since all plants consist for 70%-90% of water it is clear that the precipitation is very important.

In order to study agro-ecological processes knowledge of the meteorological variables and their influence on plants is important. Also it is clear that sufficient weather data have to be available to study these agro-ecological processes. Usually observed data for one or more years. For yield stability testing much more data are required. When not enough data are available, methods have to be searched to solve this problem. One way of doing this is generating new data which are stochastically identical with the historical data. The program SIMWTH provides a stochastical method to generate such new weather variables. It analysizes historical data in a stochastical sense in order to formulate relations between the yesterday rain status and the today rain status. Also stochastical relations are formulated between the today rain status and the solar radiation, maximum and minimum temperature. With these relations one can easily generate new variables.

It has to be stressed that this program is not an explanatory model and that it does not give any physical explanation.

# 1. Theory

## 1.1. Introduction

The program SIMWTH is based on the model and program listing of Richardson and Wright (lit.1). Shu Geng etal. used this model for simulation runs with Wageningen data and with Los Banos (Philippines) data. The original program structure was changed and two seperate programs TWGEN and TEST were made. In a later stadium these two programs were combined to one program and new routines to simulate windspeed and calculate humidity were added. In the following text a brief review of Richardson's model and its extensions will be given.

## 1.2. Richardson's model

### 1.2.1. Temperature and solar radiation

Richardson (1981) presented a method to generate samples of daily precipitation, maximum and minimum temperature and solar radiation. Precipitation was assumed to follow a Markov chain - exponential model, while the temperature and radiation were supposed to be conditional multivariate normal random variables conditioned on the precipitation status of the day. Richardson took advantage of the existing seasonal variations and described the changes of the daily means and standard deviation by the first term of a Fourier serie. The form of the first term of a Fourier serie is

$$A_{ij} = C_{0j} + C_{1j} *\cos(2*pi*(i-q_j)/365) \quad (i=1,...,365) \quad (1)$$

where j represents the weather variable; j=1 is for the maximum temperature; j=2 is for the minimum temperature; j=3 is for the solar radiation. $A_{ij}$ is the daily mean or the standard deviation of the jth variable at the ith day, $C_{0j}$ = the periodical mean of $A_{ij}$, $C_{1j}$ is the amplitude and $q_j$ is the position of the maximum in days.

For non-tropical regions this model gives reasonable results. However for the tropical conditions an extended model has to be used. In this model a combination curve of two different cosine curves is used to describe the radiation and the temperature.

$$A_{ij} = C_{0j} + C_{1j} *\cos(2*pi*(i-I1)/Y) \quad (i=1,...,I1) \quad (2)$$

$$B_{ij} = D_{0j} + D_{1j} *\cos(2*pi*(i-I2)/Z) \quad (i=I1,...,I2)$$

$$A_{ij} = C_{0j} + C_{1j} *\cos(2*pi*(i-I2)/Y) \quad (i=I2,...,365)$$

where I1 and I2 are the calculated position of the maxima in days, Y=365-(I2-I1) and Z=I2-I1, $A_{ij}$ and $B_{ij}$ are the daily mean or standard deviation of the jth variable, $C_{0j}$ and $D_{ij}$ are the periodical mean of respectively $A_{ij}$ and $B_{ij}$, $C_{1j}$ and $D_{ij}$ are the amplitude.
For tropical conditions the position of the maxima, I1 and I2, of the temperature is in this program the same as the position of the maxima of the solar radiation. The position of the maxima of the solar radiation is the calculated position of the maxima of the potential radiation (subroutine LATITU).

The coefficients of the cosine curves can be estimated by a non-lineair least square procedure from historical data. The daily mean ($\bar{x}_{ij}$) and the standard deviation ($S_{ij}$) can be computed by the cosine function of equation (1) or (2) to generate daily values of $x_{ij}$, as is shown in the following equation:

$$x_{ij} = \bar{x}_{ij} + d_{ij} * S_{ij} \qquad (3)$$

where $d_{ij}$ is a residual component and is correlated with the other meteorological variables.
If we denote

$$d_{ij} = \begin{pmatrix} d_{i1} \\ d_{i2} \\ d_{i3} \end{pmatrix} \qquad (4)$$

the vector of residuals of maximum temperature, minimum temperature and solar radiation for day i and $e_i$ is a vector of three elements that are independently and normally distributed with zero mean and unit variance. Then $d_i$ can be expressed in the following equation:

$$d_i = A\, d_{i-1} + B\, e_i \qquad (6)$$

where

$$A = M_1 M_0^{-1} \quad , \quad B.B^t = M_0 - M_1 M_0^{-1} M_1^t \quad (5)$$

(superscript -1 and t are the inverse and transpose of the matrix)

and

$$M_0 = \begin{pmatrix} 1 & r12 & r13 \\ r21 & 1 & r23 \\ r31 & r32 & 1 \end{pmatrix} \qquad (7)$$

$$M_1 = \begin{pmatrix} r(11) & r(12) & r(13) \\ r(21) & r(22) & r(23) \\ r(31) & r(32) & r(33) \end{pmatrix}$$

rjk is the cross correlation between the j th and k th variable on the same day, r(jk) is the cross correlation coefficients between the variables j and k with respect variable k lagged 1 day with respect to variable j.

Through extensive analyses and tests of a large number of locations, Richardson (1982) showed that seasonal and spatial variations in the correlations coefficients were small, and therefore used the average values to define $M_0$ and $M_1$ matrices.

$$M_0 = \begin{pmatrix} 1 & 0.633 & 0.186 \\ 0.633 & 1 & -0.193 \\ 0.186 & -0.193 & 1 \end{pmatrix}$$

$$M_1 = \begin{pmatrix} 0.621 & 0.445 & 0.087 \\ 0.563 & 0.674 & -0.100 \\ 0.015 & -0.091 & 0.251 \end{pmatrix}$$

A and B matrices which are regression matrices can then be considered as constant matrices with the following elements:

$$A = \begin{pmatrix} 0.567 & 0.086 & -0.002 \\ 0.253 & 0.504 & -0.050 \\ -0.006 & -0.039 & 0.244 \end{pmatrix}$$

(These matrices are used in subroutine TWGEN to generate new sequences of residuals)

$$B = \begin{pmatrix} 0.881 & 0 & 0 \\ 0.328 & 0.647 & 0 \\ 0.238 & -0.341 & 0.873 \end{pmatrix}$$

Thus $d_i$ can be calculated from yesterday residuals, $d_{i-1}$, and the standard normal random deviates, $e_i$. Together with $X_{ij}$ and $S_{ij}$ generated from the fitted cosine curve, daily values of the three weather variables $X_{ij}$ can be generated from the equation (3)

The program SIMWTH is based on the program WGEN, developed by Richardson and Wright (lit.1). In this program, according to Richardson, a year is divided in 13 periods and the maximum temperature and radiation in the historical data set are divided in two groups. One observed on dry days and the other observed on wet days. For every period the mean, standard deviation and the coefficient of variation of the maximum temperature and radiation on wet and on dry days, is calculated. Also the mean, standard deviation and coefficient of variation of the minimum temperature is calculated. Through the results cosine curves (i.e. first term of a Fourier serie) are fitted and the coefficients of equation (1) or (2) are calculated. By expressing equation (3) in terms of the coefficients of variation, this equation becomes

$$X_{ij} = \overline{X}_{ij} (1 + d_{ij} C_{ij}) \qquad (8)$$

where $C_{ij}$ is the coefficient of variation for the i th day of the j th variable and is estimated from the fitted Fourier series. With equation (1) or (2) and (8) and knowledge about the rain status of a day, new maximum, minimum temperature and solar radiation can be generated.

In addition, several model parameters are used in generating temperature and solar radiation were substituted by constants. For instance, in equation (1) the variable q, the position of the maximum for temperature in the cosine curve is assumed to be 200 (day) for the northern hemisphere and 20 (day) for the southern hemisphere. For solar radiation the position of the maximum in the northern hemisphere is 172 (day) and 355 (day) in the southern hemisphere. These are reasonable assumptions since in the nothern hemisphere above the tropics the maximum solar radiation occurs at June 21 (day 172) and the maximum temperature occurs in the middle of July (around day 200). In the southern hemisphere beneath the tropics the maximum solar radiation occurs at December 21 (day 355) and the maximum temperature occurs around January 20 (day 20).

In subroutine TWGEN this generation method is programed.

### 1.2.2. Rainfall

A first order, two state, Markov chain assumes that a day can either be classified as dry or wet, and that the probability of the status of a given day, whether it is dry or wet, depends only on the status of the day before. A second order Markov chain is would require a dependence on the previous two days and so on. Below the first order Markov chain is described in detail.

Suppose P10 represents the probability that today is dry given that yesterday was wet, and P00 is the probability of today being dry given that yesterday was dry. Let i and j represent the status of yesterday and today which can be either 0 or 1. Then for given status of the t-th day and the (t-1) th-day, the variables fij are defined as,

f00(t) = yesterday dry and today dry
f01(t) = yesterday dry and today wet
f10(t) = yesterday wet and today dry
f11(t) = yesterday wet and today wey

and

$$P_{ij} = f_{ij}(t)/ n_i \ ; \text{ for } i = 0,1 \text{ and } j = 0,1.$$

The summation runs over a period of time for which the probabilities are estimated. And n0 is the total number of dry days, n1 is the total number of wet days in that period. It is clear that P11 = 1 - P10 and P01 = 1 - P00. In this program a day is defined as wet if the rainfall amount is greater then 0.0.

Thus for a given period of time, the proportions that today is dry or wet by given that yesterday was dry or wet are estimates of the conditional probabilities, Pij's. They are also referred to as transitional probalities.

Since it is well known that the rainfall pattern depends on seasonality in a year, the Markov chain can best be applied for each month separately. The reliablity of the estimates of the transitional probabilities depends on the homogeneity condition of the the rainfall and the number of available data.

After the estimates are obtained for a location, the simulation of the occurence of precipitation can be accomplished by comparing the computer generated random uniform deviates with the transitional probabilities. That is, a uniform random deviate, u, on the interval (0,1) is generated, and if u is less or equal to Pi0 then today is classified as a dry day, otherwise today is defined as a wet day.

Early studies of the amount of rainfall indicated that the frequency distribution amount tends to be reversely shaped (Neyman etal, 1969). That is: small amount rainfalls have higher probablities than the large amount rainfalls. This type distribution falls in the family of Gamma distributions. The probability density function of a Gamma variable, X, is of the following form,

$$P(X) = \frac{(X-r)^{a-1} \exp(-(X-r)/b)}{b^{a}\, T(a)} \qquad (9)$$

$a > 0, \quad b > 0 \text{ and } X > r$

Where T(a) is the conventional, mathematical Gamma function, a and b the shape parameter and scale parameter of the Gamma distribution and r is the variance. This distribution is also known as a type III of Pearson's system with three parameters; a, b and r. In case all X values are greater then zero which is the case for amount rainfall at wet days, r can be set equal to zero. So the distribution is reduced to a two parameter Gamma distribution which is,

$$P(X) = \frac{X^{a-1} \exp(-X/b)}{b^{a}\, T(a)} \qquad (10)$$

The estimation of the parameters a and b can be difficult when a is small which is the case when rainfall data are fitted. Estimates obtained from ordinary maximal likelihood and method of moments are not stable when a is less than one. Approximate maximal likelihood solutions were suggested by Das (1955), Greenwood and Durand (1960), and Buishand (1977). In this program the approximate method by Greenwood and Durand is used.

$$a = (0.5000876 + 0.1648852\,Y - 0.0544274\,Y^2)/\,Y \qquad (11)$$
$$\text{for } (0 \leq Y \leq 0.5772)$$

or

$$a = (8.898919 + 9.059950\,Y + 0.9775373\,Y2)/$$
$$Y\,(17.79728 + 11.968477\,Y + Y^2)$$
$$\text{for } (0.5772 \leq Y \leq 17)$$

where $Y = \mathrm{Ln}(\overline{X}/G)$; $\overline{X}$ = arithmetic mean; G = geometric mean.

$$b = \overline{X}/a$$

This estimation method is appropiate for $0 \leq a < 1$. After a and b estimates are obtained from the historical data one can simulated the amount of rainfall on a wet day. In the subroutine WGEN one can see the method which is used to calculate the simulated rainfall.

### 1.2.3. Windspeed and humidity

Richardson (1984) suggested to use a two parameter Gamma distribution to approximate the distribution of daily windspeed. For locations where windspeed is extremely variable and strong windspeeds over 10 m/s often occur, the Weibull distribution is expected to fit the windspeed data better. For locations where strong winds are rare, a Gamma distribution probably would fit the windspeed as well as the Weibull. As a matter of convenience, the same method to calculate the Gamma distribution parameters, a and b, for rainfall distibution is used for the calculation of these parameters for the windspeed distribution. In the program listing the routines GENWIN and GAMMAD are used for the simulation of windspeed. Further it is assumed that the windspeed does not depend on the condition of the precipitation.

The humidity is not simulated in this program since dewpoint temperature simulation models are not available, but is calculated as a direct function of the minimum temperature. It is assumed that a 100% relative humidity occurs in the early morning. This leads to too high values for the humidity in arid or semi arid regions. The empirical function given by Goudriaan (1977) is used to calculate the humidity. This function is,

humidity = 6.11*exp(17.47*(Tmin/Tmin+239))   (mbar)   (12)

## 1.3. Warning

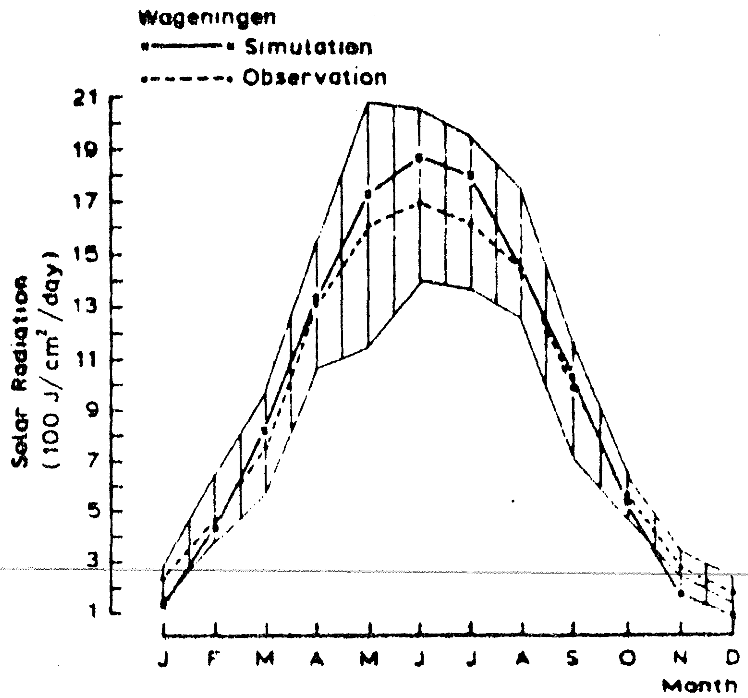
The validity of this program has to be tested for each location for which this program is used. One way to get an indication whether the results are reasonable is to use the generated data set as input and generate new weather variables. After a few repetitions one can compare the results with the original historical dataset. If the data remain stable and do not show divergence one can conclude the goodness of fit is reasonable.

Another way is to plot the historical data with the standard deviation, so one gets a range in which the value of the weather variable normally occurs. If the generated variables occur in the range one can conclude that the goodness of fit reasonable. Figure 1 shows the observed radiation, the generated radiation and the range of variation for Wageningen.

The humidity is a calculated variable and in this program a function of the minimum temperature. Since the dewpoint temperature can differ considerably one have to check the whether the humidity calculated with the historical and generated minimum temperature is reasonable.

As is mentioned in 1.2.1. the positions of the maximum in the cosine curve are not calculated from the historical data but are in this program calculated or set to a certain number. So it is possible that this can cause deviations.

Fig.1 Observed radiation, generated radiation and the range of variation for Wageningen.



## 2. About the program

### 2.1. The structure

In the following text the structure of the program will be shown by means of a few diagrams. At first the main program structure will be shown followed by a more detailed picture of the subroutines.

In the main program one can chose between a few combinations of subroutines. Diagram 1 shows the main structure of the program and its combinations.

Diagram 1. Main routine and its combinations of subroutines

```
      main program (explanatory text and calling subroutines
                    combinations)

      combination 1)      LATITU
                          TWGEN
                          XSUM

      combination 2)      LATITU
                          TEST
                          XSUM

      combination 3)      LATITU
                          TWGEN
                          XSUM
                          TEST
                          XSUM
        end
```

The first combination only calculates the parameters. The input data consist of a historical data set, which is summarized in the subroutine XSUM. The second combination only simulates the weather variables and the input data set consists of the parameters calculated in the first combination. The weather variables are summarized in subroutine XSUM. The third combination combines these two combinations. It calculates the parameters followed by the calculation of the simulated weather variables.

Subroutine LATITU calculates the the daily maximum potential radiation which will occur on a certain place with a certain latitude. This subroutine also calculates the day(s) with the highest sun elevation. The input of this routine is the latitude (90- -90!)

Subroutine TWGEN consists of mainprogram and a few subroutine calls. Diagram 2 shows the structure of this subroutine.

Diagram 2: Subroutine TWGEN and its calling routines

```
  TWGEN               (main)

  call INPUT          (reading the historical data from the external
                       file and calculating the humidity with the
                       minimum temperature)
```

```
call MSD  call FOUR call FOUR2
                    (these three subroutines calculate the sta-
                     tistics of the solar radiation ,minimum
                     and maximum temperature, based on a 7-day
                     period and fit the cosine curve (eq.1)
                     to the results)

call  PPRN      (this subroutine calculates the shape and scale
                 parameters, alpha and beta, of the Gamma distri-
                 bution of rainfall and winspeed data. It also
                 calculates the first order, two stage Markov
                 parameters for the rainfall data)

return
end
```

The output of this routine is written on the external datafile PARAMET.DAT.

Subroutine TEST consists of a main program and a few subroutine calls. Diagram 3 shows the structure of this routine.

Diagram 3: Subroutine TEST and its calling routines

```
TEST                        (main and calculating the humidity)

call  WGEN   call COSF1/COSF2 call RANDN
                            (these subroutines generate
                             daily values of solar radiation,
                             rainfall maximum and minimum
                             temperature)

call GENWIN  call GAMMAD  call RANDN (these subroutines calcu-
                                      late the windspeed)

return
end
```

The simulated weather variables are written to two different external datafile types. One type consists of a set of yearly simulated datafiles, which are written in a CSMP format. These files are A1,A2,...,A20. The other type consists of a single datafile with all the simulation results in it. It also contains the input parameters, which are used for the simulation.

2.2. The input and instruction

After the name SIMWTH is typed the program starts with the possibilities between one can chose. Figure 2 shows the options in the way the are printed on the screen.

Fig.2

```
(1) INFORMATION AND INSTRUCTION
(2) CALCULATION OF THE PARAMETERS
(3) GENERATION OF THE WEATHER VARIABLES
(4) PARAMETER CALCULATION AND SIMULATION
(5) QUIT PROGRAM

    MAKE YOUR CHOICE _
```

At this moment you can only type the numbers 1 through 5. The first option prints brief information and instruction on the program on the screen. As is mentioned in 2.1. this option is part of the main program. The second option, which is combination 1 in 2.1., only calculates the parameters and uses a historical dataset as input. It starts with a few questions. Figure 3 shows these questions.

Fig.3

```
(2) CALCULATION OF THE PARAMETERS

    NAME OF INPUTFILE
    STARTING YEAR (optional for output)
    LATITUDE OF YOUR STATION
```

The second question is only optional, it is used for labelling the summary of the historical data. After the data are inserted the program asks whether the input is correct before it starts with the calculations. The inputfile has to contain the following data, maximum temperature, minimum temperature, rainfall, solar radiation and windspeed. The program only reads these data if they are written in the correct format. Further it only accepts whole years. A minimum of 2 years of data should be used. Figure 4 shows a part of the input reading routine so the user can adapt his dataset in such a way that it fits the read statements.

Fig.4

```
      READ(11,900,END=999) (TMAX(I,J),J=1,365)
      READ(11,901,END=999) (TMIN(I,J),J=1,365)
      READ(11,901,END=999) (RAIN(I,J),J=1,365)
      READ(11,905,END=999) (RAD(I,J),J=1,365
      READ(11,901,END=999) (WIND(I,J),J=1,365)
900   FORMAT(/////,37(9X,10F6.1,/))
901   FORMAT(//,37(9X,10F6.1,/))
905   FORMAT(//,37(9X,10F6.1,/))
```

The maximum and minimum temperature have to be in degrees Celsius (C), the rainfall in (mm), the radiation in (MJ/m2/day) and the windspeed in (m/s).

The third option, combination 2 in 2.1., only simulates the weather variables. This option also starts with a few questions. Figure 5 shows these questions as they are printed on the screen.

Fig.5

(3) GENERATION OF THE WEATHER VARIABLES

NAME OF INPUT FILE
LATITUDE OF YOUR STATION
HOW MANY YEARS HAVE TO BE SIMULATED

When these questions are answered the program asks whether everything is correct before it starts its calculation. The input file has to contain the data which are given in table 1 and table 2a and 2b and are calculated in the second option from the historical data. Table 2a has to contain the parameters for the non tropical regions, where only one cosine curve is used in the generation. It also can contain the parameters for the first cosine curve for the tropical regions. Table 2b contains the parameters for the second cosine curve for the tropical regions

Table 1. Input variables Markov chain and Gamma distribution

| | |
|---|---|
| PWW(I) | Probability of a wet day if previous day was wet |
| PWD(I) | Probability of a wet day if previous day was dry |
| ALPHA(I) | Gamma distribution shape parameter of the rainfall |
| BETA(I) | Gamma distribution scale parameter of the rainfall |
| ALPW(I) | Gamma distribution shape parameter of the windspeed |
| BETW(I) | Gamma distribution scale parameter of the windspeed |

(all inputs contain 12 monthly values)

Table 2a. Cosine curve input variables

Cosine curve coefficients of max temp. ( $^{o}$C)
PM(1) - mean of maximum temp. on dry days
PM(2) - amplitude of max. temp.on dry days
PM(3) - mean of coeff. of variation on dry days
PM(4) - amplitude of the coeff. of variation on dry days
PM(5) - mean of maximum temp. on wet days
PM(6) - amplitude of max. temp.on wet days
PM(7) - mean of coeff. of variation on wet days
PM(8) - amplitude of the coeff. of variation on wet day
Cosine curve coefficients of min. temp. on wet and dry days. ( $^{o}$C)
PM(9) - mean of min. temp.
PM(10) - amplitude of min. temp.
PM(11) - mean of coeff. of variation
PM(12) - amplitude of coeff. of variation
Cosine curve coefficents of radiation (MJ/M2/day)
PM(13) - mean of the radiaton on dry days
PM(14) - amplitude of the radiation on dry days
PM(15) - mean of the radiation on dry days
PM(16) - amplitude of the radiation on dry days
PM(17) - mean of the radiaton on wet days
PM(18) - amplitude of the radiation on wet days
PM(19) - mean of the radiation on wet days
PM(20) - amplitude of the radiation on wet days

Table 2b. Cosine curve input variables

Cosine curve coefficients of max temp. ($^{o}$C)
PM(21) - mean of maximum temp. on dry days
PM(22) - amplitude of max. temp.on dry days
PM(23) - mean of coeff. of variation on dry days
PM(24) - amplitude of the coeff. of variation on dry days
PM(25) - mean of maximum temp. on wet days
PM(26) - amplitude of max. temp.on wet days
PM(27) - mean of coeff. of variation on wet days
PM(28) - amplitude of the coeff. of variation on wet day
Cosine curve coefficients of min. temp. on wet and dry days. ($^{o}$C)
PM(29 - mean of min. temp.
PM(30) - amplitude of min. temp.
PM(31) - mean of coeff. of variation
PM(32) - amplitude of coeff. of variation
Cosine curve coefficents of radiation (MJ/M2/day)
PM(33) - mean of the radiaton on dry days
PM(34) - amplitude of the radiation on dry days
PM(35) - mean of the radiation on dry days
PM(36) - amplitude of the radiation on dry days
PM(37) - mean of the radiaton on wet days
PM(38) - amplitude of the radiation on wet days
PM(39) - mean of the radiation on wet days
PM(40) - amplitude of the radiation on wet days

Figure 6 shows a part of the input reading routine so the user can adapt his dataset in such a way that it fits the read statements.

Fig.6

```
      READ(20,60) (PWW(J),J=1,12)
      READ(20,60) (PWD(J),J=1,12)
      READ(20,60) (ALPHA(J),J=1,12)
      READ(20,60) (BETA(J),J=1,12)
      READ(20,60) (ALPW(J),J=1,12)
      READ(20,60) (BETW(J),J=1,12)
 60   FORMAT(10X,12F7.3)

      READ(20,100) PM(1),PM(2)
      READ(20,105) PM(3),PM(4)
      READ(20,105) PM(5),PM(6)
      READ(20,105) PM(7),PM(8)
      READ(20,105) PM(9),PM(10)
      READ(20,105) PM(11),PM(12)
      READ(20,105) PM(13),PM(14)
      READ(20,105) PM(15),PM(16)
      READ(20,105) PM(17),PM(18)
      READ(20,105) PM(19),PM(20)

 100  FORMAT(///,20X,F10.3,F10.3)
 105  FORMAT(20X,F10.3,F10.3)

      READ(20,300) PM(21),PM(22)
      READ(20,305) PM(23),PM(24)
      READ(20,305) PM(25),PM(26)
```

```
        READ(20,305) PM(27),PM(28)
        READ(20,305) PM(29),PM(30)
        READ(20,305) PM(31),PM(32)
        READ(20,305) PM(33),PM(34)
        READ(20,305) PM(35),PM(36)
        READ(20,305) PM(37),PM(38)
        READ(20,305) PM(39),PM(40)

 300    FORMAT(///,20X,F10.3,F10.3)
 305    FORMAT(20X,F10.3,F10.3)
```

The fourth option, combination 3 in 2.1., is a combination of the earlier mentioned options. The questions asked by the program are the same and the input file has to be the historical dataset, formated as is mentioned in the second option. This last option is most commonly used.

## 2.3. The output

The external datafile PARAMET.DAT contains all the parameters which are calculated in the subroutine TWGEN. These parameters are firstly the mean, standard deviation and the coefficient of variation, based on 7-day period, for the solar radiation maximum and minimum temperature. Since the program works with a 7-day period there are 52 periods in a year. Further it has to be mentioned that the mean, standard deviation and coefficient of variation of the maximum temperature are devided in a part calculated on dry days, and a part calculated on rainy days. Cosine curves (i.e. first term of a Fourier serie) are fitted to these statistics and the coefficients are also printed in this file. Secondly, the parameters which have to be used for the simulation are written in this file. These parameters are the transitional probabilities of the first order, two stage Markov chain, of the rainfall, the shape and scale parameters of the Gamma distribution of the rainfall and windspeed. The calculations are based on a monthly period, so every variable has 12 values. In table 1 of 2.2. the Markov chain parameters are shown. Table 2 in 2.2 shows the input variables for the cosine curve (eq.1).

The external datafile WEATHER.DAT contains the generated daily data of the complete simulation run. It also contains the input variables which are mentioned in table 1 and table 2.

The external files A1,...,A20 also contain the simulation results but in a CSMP format and on a yearly basis. These files can directly used in CSMP simulation programs. Figure 7 shows the beginning of such a file. The name of the historical data set,the time and date at which this file was created and the latitude of the location are added for indentification purposes.

Fig.7

```
 TITLE A1: GENERATED WEATHER DATA YEAR  1 * CREATED ON  5-11-1985
* TIME        16:37:18
```

```
* BASED ON FILE :WS.DAT
PARAM LAT=  52.

TABLE TMPHT(1-365)=...          (max.temp)
3.5,    5.1,    3.3,     .3,  -2.9,     .3,    -.3,    2.6,    2.0,    5.1,...
5.1,    1.4,    5.8,    5.6,    5.3,    4.0,    8.5,    1.4,    1.0,    3.1,...
```

The subroutine XSUM summarizes the input data in SUMMAR.DAT and the simulation results in SUMWEATH.DAT This routine summarizes the data on a monthly basis and also the yearly mean for the solar radiation, windspeed, humidity and the maximum and the minimum temperature. Also the total amount of rainfall and the number of rain days in a year are given. Further the standard variation, coefficient of variation and the standard error are calculated based on a monthly basis.

## 3. Literature

1) Johnson, N.L. and S. Kotz, 1970. Continuous univariate distributions -I. Houghton Mifflin Company. Boston

2) Matalas, N.C., 1967. Mathematical assesment of synthetic hydrology. Water Resources Res. 3(4):937-945

3) Richardson, C.W., 1981. Stochastic simulation of daily precipitation, temperature and solar radiation data. J. Hydrology 48:1-17

4) Richardson, C.W., 1982. Dependance structure of daily temperature and solar radiation. Trans. ASAE 25(3)735-739

5) Richardson, C.W., and Wright D.A. 1984. WGEN: A model for generating daily weather variables. U.S. Department of Agriculture, Agricultural Research Service, ARS-8.

6) Shu Geng, Frits Penning de Vries & Iwan Supit. Analysis and simulation of weather variables-part 1: Rain and Wind in Wageningen. Simulation Report CABO-TT No.4

7) Shu Geng, Frits Penning de Vries & Iwan Supit. Analysis and simulation of weather variables-part 2: Temperature and radiation Simulation Report CABO-TT No.5

8) Goudriaan, J., 1977. Crop micrometeorology: A simulation study. Sim. Monograph PUDOC, Wageningen, The Netherlands.

Appendix A: Program listing SIMWTH

```
$DEBUG
C*******************************************************************
C       NAME: SIMWTH
C       DATE : FEBRUARY 12,1986
C
C       ORIGINAL SUBROUTINES TEST AND TWGEN PROGRAMMED BY RICHARDSON
C       MAIN PROGRAM AND ADAPTED SUBROUTINES BY IWAN SUPIT
C       THIS PROGRAM CAN RUN ON AN IBM PC/AT WITH A MATHEMATICAL
C       CO-PROCESSOR  80287 512 KB RAM
C
C       MAIN PROGRAM CAN ONLY RUN ON THE PREVIOUS MENTIONED MACHINES
C       SUBROUTINES CAN RUN ON OTHER MACHINES
C
C       LITERATURE:
C    1) RICHARDSON, C.W, AND D.A. WRIGHT, 1984 WGEN: A model for
C       generating daily weather variables. U.S. Depsrtment of Agri-
C       culture, Agricultural Research Service, ARS-8,83 p.
C    2) SHU GENG etal, 1985 Analysis and simualtion of weather
C       variables - part I. Simulation Reports CABO-TT No. 4
C    3) SHU GENG etal, 1985 Analysis and simualtion of weather
C       variables - part II. Simulation Reports CABO-TT No. 5
C*******************************************************************
       IMPLICIT INTEGER (I-N)
       IMPLICIT REAL (A-H,O-Z)
       COMMON /DAT1/ TMX(20,366)
       COMMON /DAT2/ TMN(20,366)
       COMMON /DAT3/ RN(20,366)
       COMMON /DAT4/ RD(20,366)
       COMMON /DAT11/ WN(20,366)
       COMMON /DAT14/HUM(20,366)
C*******************************************************************
C        TMX          MAXIMUM TEMPERATURE  (C)          INPUT
C        TMN          MINIMUM TEMPERATURE  (C)            "
C        RN           RAIN                 (MM)           "
C        RD           RADIATION            (MJ/M2/DAY)    "
C        WN           WINDSPEED            (M/S)          "
C        HUM          HUMIDITY (vap. pres.) (KPa)      CALCULATED
C*******************************************************************
       COMMON /DAT12/ ALPW(12),BETW(12),WIND(365)
       COMMON /DAT7/ PWD(12),PWW(12),ALPHA(12),BETA(12)
C*******************************************************************
C       WIND    SIMULATED WIND SPEED FOR ONE YEAR
C       PWW     PROBABILITY OF A WET DAY IF PREVIOUS DAY WAS WET
C       PWD     PROBABILITY OF A WET DAY IF PREVIOUS DAY WAS DRY
C       ALPHA   GAMMA DISTRIBUTION SHAPE PARAMETER OF THE RAIN PARAM.
C       BETA    GAMMA DISTRIBUTION SCALE PARAMETER OF THE RAIN PARAM.
C       ALPW    GAMMA DISTRIBUTION SHAPE PARAMETER OF THE WIND PARAM.
C       BETW    GAMMA DISTRIBUTION SCALE PARAMETER OF THE WIND PARAM.
C       TX      CONTAINS THE GENERATION PARAMETERS
C*******************************************************************
       COMMON /DAT13/ XN(20,366)
       COMMON /DAT5/ RTO(20,366)
       COMMON /DAT6/ TXM(366),TXS(366),TXM1(366),TXS1(366),TNM(366)
       COMMON /DAT8/ RAIN(366),TMIN(366),TMAX(366),RAD(366),TNS(366)
       COMMON /DAT9/ RMO(366),RSO(366),RM1(366),RS1(366),RC(366)
```

```
      COMMON /DAT10/ RCF(12)
C*********************************************************************
C     XN     DUMMY MATRIX FOR WINDSPEED AND RAIN
C     RTO    ACTUAL RADIATION/POTENTIAL RADIATION
C     RAIN   ONE YEAR SIMULATED RAINFALL              (MM)
C     TMAX   ONE YEAR SIMULATED MAXIMUM TEMP.         (F)
C     TMIN   ONE YEAR SIMULATED MINIMUM TEMP.         (F)
C     RAD    ONE YEAR SIMULATED RADIATION             (CAL/CM2/DAY)
C     RC     MAXIMUM POTENTIAL RADIATION              (CAL/CM2/DAY)
C     TXM    DAILY GENERATED MEAN OF MAX. TEMP. (DRY) (F)
C     TXM1   DAILY GENERATED MEAN OF MAX. TEMP. (WET) (F)
C     TNM    DAILY GENERATED MEAN OF MIN. TEMP.
C     RMO    DAILY GENERATED MEAN OF RAD. (DRY)        (CAL/CM2/DAY)
C     RM1    DAILY GENERATED MEAN OF RAD. (WET)        (CAL/CM2/DAY)
C     TXS    PRODUCT OF THE GENERATED DAILY MEAN AND COEF. OF VAR.
C            OF THE MAX. TEMP. (DRY)
C     TXS1   PRODUCT OF THE GENERATED DAILY MEAN AND COEF. OF VAR.
C            OF THE MAX. TEMP.  (WET)
C     TNS    PRODUCT OF THE GENERATED DAILY MEAN AND COEF. OF VAR.
C            OF THE MIN. TEMP.
C     RSO    PRODUCT OF THE GENERATED DAILY MEAN AND COEF. OF VAR.
C            OF THE RAD. (DRY)
C     RS1    PRODUCT OF THE GENERATED DAILY MEAN AND COEF. OF VAR.
C            OF THE RAD. (WET)
C**********************************************************************
      INTEGER*2 MH,MM,MYR,MMO,MDAY,MS,MHS
      CHARACTER*15 INPNAM
      CHARACTER CH,DH,XH
      CH=CHAR(27)
      XH=CHAR(7)
      CALL GETTIM(MH,MM,MS,MHS)
      CALL GETDAT(MYR,MMO,MDAY)
C********************************************************************
C     INPNAM=NAME OF INPUTFILE
C     DH= JUMP CONTROL FOR THE SCREEN TEXT
C     CH=ESC
C     XH=BEEP
C     GETTIM CALL TIME ROUTINE
C     GETDAT CALL DATE ROUTINE
C     MH = HOUR
C     MM = MINUTE
C     MS = SECOND
C     MHS= 1/100 SECOND
C     MYR= YEAR
C     MMO= MMO
C     MDAY=DAY
C*********************************************************************
C     TEXT AND SCREEN
 10   WRITE(*,'(1X,2A)')CH,'[2J'
      WRITE(*,'(1X,2A)')CH,'[49;49;13p'
      WRITE(*,'(1X,2A)')CH,'[50;50;13p'
      WRITE(*,'(1X,2A)')CH,'[51;51;13p'
      WRITE(*,'(1X,2A)')CH,'[52;52;13p'
      WRITE(*,'(1X,2A)')CH,'[53;53;13p'
      WRITE(*,'(///,4X,'' (1) INFORMATION AND INSTRUCTION '')')
```

```
      WRITE(*,'(4X,'' (2) CALCULATION OF THE PARAMETERS '')')
      WRITE(*,'(4X,'' (3) GENERATION OF THE WEATHER VARIABLES '')')
      WRITE(*,'(4X,'' (4) PARAMETER CALCULATION AND GENERATION'')')
      WRITE(*,'(4X,'' (5) QUIT PROGRAM '')')
      WRITE(*,'(//,8X,'' MAKE YOUR CHOICE '',)')
      READ(*,'(A1)') DH
      WRITE(*,'(1X,2A)')CH,'[49;49p'
      WRITE(*,'(1X,2A)')CH,'[50;50p'
      WRITE(*,'(1X,2A)')CH,'[51;51p'
      WRITE(*,'(1X,2A)')CH,'[52;52p'
      WRITE(*,'(1X,2A)')CH,'[53;53p'
      N=ICHAR(DH)
      I=N-48
      IF(N.LT.49 .OR. N.GT.53) GOTO 10
      IF(I .EQ. 1) THEN
      WRITE(*,'(1X,2A)')CH,'[66;66;13p'
      WRITE(*,'(1X,2A)')CH,'[70;70;13p'
      WRITE(*,'(1X,2A)')CH,'[77;77;13p'
      WRITE(*,'(1X,2A)')CH,'[98;98;13p'
      WRITE(*,'(1X,2A)')CH,'[102;102;13p'
      WRITE(*,'(1X,2A)')CH,'[109;109;13p'
      ELSE
      WRITE(*,'(1X,2A)')CH,'[66;66p'
      WRITE(*,'(1X,2A)')CH,'[70;70p'
      WRITE(*,'(1X,2A)')CH,'[77;77p'
      WRITE(*,'(1X,2A)')CH,'[98;98p'
      WRITE(*,'(1X,2A)')CH,'[102;102p'
      WRITE(*,'(1X,2A)')CH,'[109;109p'
      ENDIF
      WRITE(*,'(1X,2A)')CH,'[45;90;13p'
      GOTO(100,200,300,400,500)I
100   WRITE(*,'(1X,2A)')CH,'[2J'
      WRITE(*,'(4X,'' (1) INFORMATION AND INSTRUCTION ON:'')')
      WRITE(*,101)
101   FORMAT(/,5X,'[7m (2) CALCULATION OF THE PARAMETERS [0m')
      WRITE(*,105)
105   FORMAT(//,
    $ ' This program calculates the following parameters for the two',
    $ ' stage Markov Gamma Distribution Model:',/
    $ ' The transitional probabilities for the two stage Markov Model',
    $ ' and the shape and',/
    $ ' scale parameters, Alpha and Beta, for the Gamma',
    $ ' Distribution.',/
    $ ' Further the statistics of the maximum',
    $ ' temperature, minimum temperature',/
    $ ' and of the solar radiation based on a 7-day period',
    $ ' are calculated.',/
    $ ' One cosine curve is fitted for non tropical regions is',
    $ ' to the',/
    $ ' results of these statics. For tropical regions two curves',
    $ ' are fitted.',/
    $ ' These parameters are used for the weather generation',
    $ ' in option (3)',/
    $ ' and in option (4). Table 1 and Table 2a and 2b',
    $ ' show these parameters.',
```

```
     $  /////,' (F=FOREWARD, M=MENU)',2X,)
        READ(*,'(A1)') DH
        N=ICHAR(DH)
        IF(N .EQ. 66 ) THEN
           GOTO 100
        ELSEIF(N .EQ. 98) THEN
           GOTO 100
        ELSEIF(N .EQ. 70) THEN
           GOTO 110
        ELSEIF(N .EQ. 102) THEN
           GOTO 110
        ELSEIF(N .EQ. 77)THEN
           GOTO 10
        ELSEIF(N .EQ. 109) THEN
           GOTO 10
        ELSE
           GOTO 100
        END IF
110     WRITE(*,'(1X,2A)')CH,'[2J'
        WRITE(*,'(4X,'' (1) INFORMATION AND INSTRUCTION ON:'')')
        WRITE(*,101)
        WRITE(*,120)
120     FORMAT(//,' Table 1:',//
     $ ,3X,'PWW(I)   Probability of a wet day if previous day was wet',/
     $ ,3X,'PWD(I)   Probability of a wet day if previous day was dry',/
     $  ,3X,'ALPHA(I) Shape parameter of Gamma Distribution for rain',/
     $  ,3X,'BETA(I)  Scale parameter of Gamma Distribution for rain',/
     $  ,3X,'ALPW(I)  Shape parameter of Gamma Distribution for wind',/
     $  ,3X,'BETW(I)  Scale parameter of Gamma Distribution for wind',//
     $  ,' Table 1 shows the parameters used for the two stage',
     $  ,' Markov-Gamma Distribution',/
     $  ,' model. All parameters contain 12 monthly values.',////
     $  ,' (B=BACKWARD, F=FOREWARD, M=MENU)',2X,)
        READ(*,'(A1)') DH
        N=ICHAR(DH)
        IF(N .EQ. 66) THEN
           GOTO 100
        ELSEIF(N .EQ. 98) THEN
           GOTO 100
        ELSEIF(N .EQ. 70) THEN
           GOTO 130
        ELSEIF(N .EQ. 102) THEN
           GOTO 130
        ELSEIF(N .EQ. 77) THEN
           GOTO 10
        ELSEIF(N .EQ. 109) THEN
           GOTO 10
        ELSE
           GOTO 110
        END IF
130     WRITE(*,'(1X,2A)')CH,'[2J'
        WRITE(*,125)
125     FORMAT(/,' Table 2a: The Cosine coefficients for tropical and',
     $  ' non tropical regions',/
     $  '  Cosine coefficients of maximum temperature (TMAX)',/,
```

```
     $  3X,'PM(1)  Mean                         on dry days',/,
     $  3X,'PM(2)  Amplitude                    on dry days',/,
     $  3X,'PM(3)  Mean of the coef. of var.    on dry days',/,
     $  3X,'PM(4)  Amplitude of coef. of var.   on dry days',/,
     $  3X,'PM(5)  Mean                         on wet days',/,
     $  3X,'PM(6)  Amplitude                    on wet days',/,
     $  3X,'PM(7)  Mean of the coef. of var.    on wet days',/,
     $  3X,'PM(8)  Amplitude of coef. of var.   on wet days',/,
     $  '  Cosine coefficients of minimum temperature (TMIN)',/,
     $  3X,'PM(9)  Mean                         on wet or dry days',/,
     $  3X,'PM(10) Amplitude                    on wet or dry days',/,
     $  3X,'PM(11) Mean of coef. of var.        on wet or dry days',/,
     $  3X,'PM(12) Amplitude of coef. of var.   on wet or dry days')
        WRITE(*,126)
126     FORMAT('  Cosine coefficients of the radiation (RAD)',/,
     $  3X,'PM(13) Mean                         on dry days',/,
     $  3X,'PM(14) Amplitude                    on dry days',/,
     $  3X,'PM(15) Mean of the coef. of var.    on dry days',/
     $  3X,'PM(16) Amplitude of coef. of var.   on wet days',/
     $  3X,'PM(17) Mean                         on wet days',/,
     $  3X,'PM(18) Amplitude                    on wet days',/,
     $  3X,'PM(19) Mean of the coef. of var.    on wet days',/
     $  3X,'PM(20) Amplitude of coef. of var.   on dry days',/
     $  ' (B=BACKWARD, F=FOREWARD, M=MENU)',2X,)
        READ(*,'(A1)') DH
        N=ICHAR(DH)
        IF(N .EQ. 66) THEN
           GOTO 110
        ELSEIF(N .EQ. 98) THEN
           GOTO 110
        ELSEIF(N .EQ. 70) THEN
           GOTO 132
        ELSEIF(N. EQ. 102) THEN
           GOTO 132
        ELSEIF(N .EQ. 77) THEN
           GOTO 10
        ELSEIF(N .EQ. 109) THEN
           GOTO 10
        ELSE
           GOTO 130
        END IF
132     WRITE(*,'(1X,2A)')CH,'[2J'
        WRITE(*,127)
127     FORMAT(/,' Table 2b: The Cosine coefficients only for',
     $  ' tropical regions',/
     $  '  Cosine coefficients of maximum temperature (TMAX)',/,
     $  3X,'PM(21)  Mean                         on dry days',/,
     $  3X,'PM(22)  Amplitude                    on dry days',/,
     $  3X,'PM(23)  Mean of the coef. of var.    on dry days',/,
     $  3X,'PM(24)  Amplitude of coef. of var.   on dry days',/,
     $  3X,'PM(25)  Mean                         on wet days',/,
     $  3X,'PM(26)  Amplitude                    on wet days',/,
     $  3X,'PM(27)  Mean of the coef. of var.    on wet days',/,
     $  3X,'PM(28)  Amplitude of coef. of var.   on wet days',/,
     $  '  Cosine coefficients of minimum temperature (TMIN)',/,
```

```
     $  3X,'PM(29)  Mean                       on wet or dry days',/,
     $  3X,'PM(30) Amplitude                   on wet or dry days',/,
     $  3X,'PM(31) Mean of coef. of var.       on wet or dry days',/,
     $  3X,'PM(32) Amplitude of coef. of var. on wet or dry days')
        WRITE(*,128)
128     FORMAT('  Cosine coefficients of the radiation (RAD)',/,
     $  3X,'PM(33) Mean                       on dry days',/,
     $  3X,'PM(34) Amplitude                  on dry days',/,
     $  3X,'PM(35) Mean of the coef. of var.  on dry days',/
     $  3X,'PM(36) Amplitude of coef. of var. on wet days',/
     $  3X,'PM(37) Mean                       on wet days',/,
     $  3X,'PM(38) Amplitude                  on wet days',/,
     $  3X,'PM(39) Mean of the coef. of var.  on wet days',/
     $  3X,'PM(40) Amplitude of coef. of var. on dry days',/
     $  ' (B=BACKWARD, F=FOREWARD, M=MENU)',2X,)
        READ(*,'(A1)') DH
        N=ICHAR(DH)
        IF(N .EQ. 66) THEN
           GOTO 130
        ELSEIF(N .EQ. 98) THEN
           GOTO 130
        ELSEIF(N .EQ. 70) THEN
           GOTO 140
        ELSEIF(N. EQ. 102) THEN
           GOTO 140
        ELSEIF(N .EQ. 77) THEN
           GOTO 10
        ELSEIF(N .EQ. 109) THEN
           GOTO 10
        ELSE
           GOTO 132
        END IF
140     WRITE(*,'(1X,2A)')CH,'[2J'
        WRITE(*,'(4X,'' (1) INFORMATION AND INSTRUCTION ON:'')')
        WRITE(*,101)
        WRITE(*,135)
135     FORMAT(//,
     $  ' Option (2) stops after the calculation of the parameters.',
     $  ' In option (3) and (4)',/
     $  ' these parameters are used for the simulation of the weather',
     $  ' variables. The',/
     $  ' results of option (2) are printed on the external file',
     $  ' PARAMET.DAT. A summary of',
     $  ' the input data is printed on the external file SUMMAR.DAT.',
     $  ' Your input data',/
     $  ' file has to contain the following data TMAX, TMIN, RAIN',
     $  ' RAD and WIND. At least',/
     $  ' two years of data are needed. Your input data file has to be',
     $  ' read by the',/
     $  ' following FORTRAN statements:')
        WRITE(*,146)
146     FORMAT(/,
     $  ,6X,'READ(11,900,END=999) (TMAX(I,J),J=1,365)    * (C)',/
     $  ,6X,'READ(11,901,END=999) (TMIN(I,J),J=1,365)    * (C)',/
     $  ,6X,'READ(11,901,END=999) (RAIN(I,J),J=1,365)    * (mm)',/
```

```
     $  ,6X,'READ(11,905,END=999) (RAD(I,J),J=1,365)     * (MJ/M2/DAY)',/
     $  ,6X,'READ(11,901,END=999) (WIND(I,J),J=1,365)    * (m/s)',/,
     $  ' 900  FORMAT(/////,37(9X,10F6.1,/))',/
     $  ' 901  FORMAT(//,37(9X,F6.1,/))',/
     $  ' 905  FORMAT(//,37(9X,F6.1,/))',//
     $  ' I is the counter for the years. February 29 in leap years',
     $  ' is excluded.',//,
     $  ' (B=BACKWARD, F=FOREWARD, M=MENU)',2X,)
        READ(*,'(A1)') DH
        N=ICHAR(DH)
        IF(N .EQ. 66) THEN
           GOTO 130
        ELSEIF(N .EQ. 98) THEN
           GOTO 130
        ELSEIF(N .EQ. 70) THEN
           GOTO 150
        ELSEIF(N .EQ. 102) THEN
           GOTO 150
        ELSEIF(N .EQ. 77) THEN
           GOTO 10
        ELSEIF(N .EQ. 109) THEN
           GOTO 10
        ELSE
           GOTO 140
        END IF
150     WRITE(*,'(1X,2A)')CH,'[2J'
        WRITE(*,'(4X,'' (1) INFORMATION AND INSTRUCTION ON:'')')
        WRITE(*,151)
151     FORMAT(/,5X,'[7m (3) GENERATION OF THE WEATHER VARIABLES [0m')
        WRITE(*,152)
152     FORMAT(/,
     $  ' This program generates weather variables on a daily basis',
     $  ' and the',/
     $  ' results are printed out to the external file WEATHER.DAT',
     $  ' and SUMWEATH.DAT.',/
     $  ' Your inputfile has to contain the earlier mentioned',
     $  ' transitional probabilities,',/
     $  ' shape and scale parameters and the Fourier coefficients.',
     $  ' These data have',/
     $  ' to be read by the following FORTRAN statements:',//
     $  ,6X,'READ(20,60) (PWW(I),I=1,12)',/
     $  ,6X,'READ(20,60) (PWD(I),I=1,12)',/
     $  ,6X,'READ(20,60) (ALPHA(I),I=1,12)',/
     $  ,6X,'READ(20,60) (BETA(I),I=1,12)',/
     $  ,6X,'READ(20,60) (ALPW(I),I=1,12)',/
     $  ,6X,'READ(20,60) (BETW(I),I=1,12)',/
     $  ' 60   FORMAT(10X,12(F7.3))',///
     $  ' (B=BACKWARD, F=FOREWARD, M=MENU)',2X,)
        READ(*,'(A1)') DH
        N=ICHAR(DH)
        IF(N .EQ. 66) THEN
           GOTO 140
        ELSEIF(N .EQ. 98) THEN
           GOTO 140
        ELSEIF(N .EQ. 70) THEN
```

```
          GOTO 153
       ELSEIF(N .EQ. 102) THEN
          GOTO 153
       ELSEIF(N .EQ. 77) THEN
          GOTO 10
       ELSEIF(N .EQ. 109) THEN
          GOTO 10
       ELSE
          GOTO 150
       END IF
153    WRITE(*,'(1X,2A)')CH,'[2J'
       WRITE(*,'(4X,'' (1) INFORMATION AND INSTRUCTION ON:'')')
       WRITE(*,151)
       WRITE(*,154)
154    FORMAT(/,' Cosine parameters for non tropical regions or',
     $ ' parameters for',/
     $ ,'the first cosine for tropical regions',//
     $ ,8X,'READ(20,100) PM(1),PM(2)',/
     $ ,8X,'READ(20,105) PM(3),PM(4)',/
     $ ,8X,'READ(20,110) PM(5),PM(6)',/
     $ ,8X,'READ(20,115) PM(7),PM(8)',/
     $ ,8X,'READ(20,120) PM(9),PM(10)',/
     $ ,8X,'READ(20,125) PM(11),PM(12)',/
     $ ,8X,'READ(20,130) PM(13),PM(14)',/
     $ ,8X,'READ(20,135) PM(15),PM(16)',/
     $ ,8X,'READ(20,140) PM(17),PM(18)',/
     $ ,8X,'READ(20,145) PM(19),PM(20)',//
     $ ,' 100     FORMAT(///,20X,F10.3,F10.3)',/
     $ ,' 105     FORMAT(20X,F10.3,F10.3)',//
     $ ' (B=BACKWARD, F=FOREWARD, M=MENU)',2X,)
       READ(*,'(A1)') DH
       N=ICHAR(DH)
       IF(N .EQ. 66) THEN
          GOTO 150
       ELSEIF(N .EQ. 98) THEN
          GOTO 150
       ELSEIF(N .EQ. 70) THEN
          GOTO 155
       ELSEIF(N .EQ. 102) THEN
          GOTO 155
       ELSEIF(N .EQ. 77) THEN
          GOTO 10
       ELSEIF(N .EQ. 109) THEN
          GOTO 10
       ELSE
          GOTO 153
       END IF
155    WRITE(*,'(1X,2A)')CH,'[2J'
       WRITE(*,'(4X,'' (1) INFORMATION AND INSTRUCTION ON:'')')
       WRITE(*,151)
       WRITE(*,156)
156    FORMAT(/,' Parameters of the second cosine only for tropical',
     $ ' regions',//
     $ ,8X,'READ(20,100) PM(21),PM(22)',/
     $ ,8X,'READ(20,105) PM(23),PM(24)',/
```

```
     $    ,8X,'READ(20,110) PM(25),PM(26)',/
     $    ,8X,'READ(20,115) PM(27),PM(28)',/
     $    ,8X,'READ(20,120) PM(29),PM(30)',/
     $    ,8X,'READ(20,125) PM(31),PM(32)',/
     $    ,8X,'READ(20,130) PM(33),PM(34)',/
     $    ,8X,'READ(20,135) PM(35),PM(36)',/
     $    ,8X,'READ(20,140) PM(37),PM(38)',/
     $    ,8X,'READ(20,145) PM(39),PM(40)',//
     $    ,' 100     FORMAT(///,20X,F10.3,F10.3)',/
     $    ,' 105     FORMAT(20X,F10.3,F10.3)',///
     $    ' Input tables for CSMP programs are printed on external',
     $    ' files A1,..,A20.',/////
     $    ' (B=BACKWARD, F=FOREWARD, M=MENU)',2X,)
          READ(*,'(A1)') DH
          N=ICHAR(DH)
          IF(N .EQ. 66) THEN
             GOTO 153
          ELSEIF(N .EQ. 98) THEN
             GOTO 153
          ELSEIF(N .EQ. 70) THEN
             GOTO 170
          ELSEIF(N .EQ. 102) THEN
             GOTO 170
          ELSEIF(N .EQ. 77) THEN
             GOTO 10
          ELSEIF(N .EQ. 109) THEN
             GOTO 10
          ELSE
             GOTO 155
          END IF
170       WRITE(*,'(1X,2A)')CH,'[2J'
          WRITE(*,'(4X,'' (1) INFORMATION AND INSTRUCTION ON:'')')
          WRITE(*,171)
171       FORMAT(/,5X,'[7m (4) PARAMETER CALCULATION AND GENERATION[0m')
          WRITE(*,172)
172       FORMAT(//,
     $    ' This program combines option (2) and (3). Your input file',
     $    ' has to be',/
     $    ' formated like the earlier mentioned input file for',
     $    ' option (2).',/
     $    ' The results of the parameter calcualtion are printed on',
     $    ' the external',/
     $    ' outputfiles PARAMET.DAT and SUMMAR.DAT. The results of the',
     $    ' simulation part',/
     $    ' are printed on the external output files WEATHER.DAT and',
     $    ' SUMWEATH.DAT',/
     $    ' Also input tables for CSMP programs are printed on external',
     $    ' files A1,..,A20.',//
     $    ' (B=BACKWARD, M=MENU)',2X,)
          READ(*,'(A1)') DH
          N=ICHAR(DH)
          IF(N .EQ. 66) THEN
             GOTO 155
          ELSEIF(N .EQ. 98) THEN
             GOTO 155
```

```
      ELSEIF(N .EQ. 77) THEN
         GOTO 10
      ELSEIF(N .EQ. 109) THEN
         GOTO 10
      ELSE
         GOTO 170
      END IF
      GOTO 500
C     ******************************************************
 200  WRITE(*,'(1X,2A)')CH,'[2J'
      WRITE(*,205)
 205  FORMAT(4X,'[7m','(2) CALCULATION OF THE PARAMETERS','[0m',/)
      WRITE(*,210)
 210  FORMAT(' [7m',' NAME OF INPUT FILE                         ',/
     $ '  STARTING YEAR (optional for output)  ',/
     $ '  LATITUDE OF YOUR STATION              ','[0m')
C     INPUT SQUARE
      WRITE(*,'(1X,2A)')CH,'[3;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
      WRITE(*,'(1X,2A)')CH,'[4;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
      WRITE(*,'(1X,2A)')CH,'[5;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
C     READ DATA FROM SCREEN
      WRITE(*,'(1X,2A)')CH,'[3;43H'
      READ(*,'(A15)') INPNAM
 207  WRITE(*,'(1X,2A)')CH,'[4;43H'
      READ(*,'(I8)',ERR=220) LSTART
 211  WRITE(*,'(1X,2A)')CH,'[5;43H'
      READ(*,'(F5.0)',ERR=224) ALAT
C     POSITION CURSOR, SET Y AND N KEY WITH RETURN
 212  WRITE(*,'(1X,2A)')CH,'[8;2H'
      WRITE(*,'(1X,2A)')CH,'[89;89;13p'
      WRITE(*,'(1X,2A)')CH,'[78;78;13p'
      WRITE(*,'(1X,2A)')CH,'[121;121;13p'
      WRITE(*,'(1X,2A)')CH,'[110;110;13p'
      WRITE(*,215)
      READ(*,'(A1)') DH
C     RESET Y AND N KEY WITHOUT RETURN
      WRITE(*,'(1X,2A)')CH,'[89;89p'
      WRITE(*,'(1X,2A)')CH,'[78;78p'
      WRITE(*,'(1X,2A)')CH,'[121;121p'
      WRITE(*,'(1X,2A)')CH,'[110;110p'
      N=ICHAR(DH)
C     JUMP CONTROL
      IF(N. EQ. 89) THEN
         GOTO 225
      ELSEIF(N .EQ. 121) THEN
         GOTO 225
      ELSEIF(N. EQ. 78) THEN
          GOTO 200
      ELSEIF(N .EQ. 110) THEN
          GOTO 200
      ELSE
```

```
            GOTO 212
         ENDIF
C******* Error in input routine********
 220     WRITE(*,'(1X,2A)')XH
         WRITE(*,'(1X,2A)')CH,'[4;41H'
         WRITE(*,'('' [7m'',''              '',''[0m'')')
         GOTO 207
 224     WRITE(*,'(1X,2A)')XH
         WRITE(*,'(1X,2A)')CH,'[5;41H'
         WRITE(*,'('' [7m'',''              '',''[0m'')')
         GOTO 211
C*********Start of the calculation*******************************
 225     WRITE(*,'(1X,2A)')CH,'[2J'
         WRITE(*,'(1X,2A)')CH,'[10;35H'
         WRITE(*,'('' [5m'',''COMPUTING'',''[0m'')')
         CALL LATITU(ALAT,I1,I2)
         CALL TWGEN (ALAT,I1,I2,NYR1,INPNAM,MDAY,MMO,MYR,MH,MM,MS)
         CALL XSUM(NYR1,LSTART,0,MDAY,MMO,MYR,MH,MM,MS,INPNAM)
         WRITE(*,'(1X,2A)')CH,'[2J'
         GOTO 500
C*****************************************************************
 300     WRITE(*,'(1X,2A)')CH,'[2J'
         WRITE(*,305)
 305     FORMAT(4X,'[7m','(3) GENERATION OF THE WEATHER VARIABLES',
     $   '[0m',/)
         WRITE(*,310)
 310     FORMAT(' [7m',' NAME OF INPUT FILE                        ',/
     $   '  LATITUDE OF YOUR STATION              ',/
     $   '  HOW MANY YEARS HAVE TO BE GENERATED ',1X,'[0m')
C        INPUT SQUARE
         WRITE(*,'(1X,2A)')CH,'[3;41H'
         WRITE(*,'('' [7m'',''              '',''[0m'')')
         WRITE(*,'(1X,2A)')CH,'[4;41H'
         WRITE(*,'('' [7m'',''              '',''[0m'')')
         WRITE(*,'(1X,2A)')CH,'[5;41H'
         WRITE(*,'('' [7m'',''              '',''[0m'')')
C        READ DATA FROM SCREEN
         WRITE(*,'(1X,2A)')CH,'[3;43H'
         READ(*,'(A15)') INPNAM
 309     WRITE(*,'(1X,2A)')CH,'[4;43H'
         READ(*,'(F5.0)',ERR=320) ALAT
 311     WRITE(*,'(1X,2A)')CH,'[5;43H'
         READ(*,'(I8)',ERR=322) NYR2
C        POSITION CURSOR, SET Y AND N KEY WITH RETURN
 312     WRITE(*,'(1X,2A)')CH,'[8;2H'
         WRITE(*,'(1X,2A)')CH,'[89;89;13p'
         WRITE(*,'(1X,2A)')CH,'[78;78;13p'
         WRITE(*,'(1X,2A)')CH,'[121;121;13p'
         WRITE(*,'(1X,2A)')CH,'[110;110;13p'
         WRITE(*,315)
 315     FORMAT(///,1X,'[7m',' EVERYTHING CORRECT? (Y/N)','[0m',2X,)
         READ(*,'(A1)') DH
C        RESET Y AND N KEY WITHOUT RETURN
         WRITE(*,'(1X,2A)')CH,'[89;89p'
         WRITE(*,'(1X,2A)')CH,'[78;78p'
```

```
      WRITE(*,'(1X,2A)')CH,'[121;121p'
      WRITE(*,'(1X,2A)')CH,'[110;110p'
      N=ICHAR(DH)
C     JUMP CONTROL
      IF(N. EQ. 89) THEN
         GOTO 325
      ELSEIF(N .EQ. 121) THEN
         GOTO 325
      ELSEIF(N. EQ. 78) THEN
          GOTO 300
      ELSEIF(N .EQ. 110) THEN
          GOTO 300
      ELSE
          GOTO 312
      ENDIF
C******* Error in input routine********
 320  WRITE(*,'(1X,2A)')XH
      WRITE(*,'(1X,2A)')CH,'[4;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
      GOTO 309
 322  WRITE(*,'(1X,2A)')XH
      WRITE(*,'(1X,2A)')CH,'[5;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
      GOTO 311
C*********Start of the simulation*********************************
 325  WRITE(*,'(1X,2A)')CH,'[2J'
      WRITE(*,'(1X,2A)')CH,'[10;35H'
      WRITE(*,'('' [5m'',''COMPUTING'',''[0m'')')
      CALL LATITU(ALAT,I1,I2)
      CALL TEST(0,NYR2,INPNAM,ALAT,MDAY,MMO,MYR,MH,MM,MS,I1,I2)
      CALL XSUM(NYR2,1,1,MDAY,MMO,MYR,MH,MM,MS,INPNAM)
      WRITE(*,'(1X,2A)')CH,'[2J'
      GOTO 500
C*****************************************************************
 400  WRITE(*,'(1X,2A)')CH,'[2J'
      WRITE(*,405)
 405  FORMAT(4X,'[7m','(4) PARAMETER CALCULATION AND GENERATION',
     $ '[0m',/)
      WRITE(*,410)
 410  FORMAT(' [7m',' NAME OF INPUT FILE                        ',/
     $ '  STARTING YEAR (optional for output)  ',/
     $ '  LATITUDE OF YOUR STATION             ',/
     $ '  HOW MANY YEARS HAVE TO BE GENERATED ',1X,'[0m')
C     INPUT SQUARE
      WRITE(*,'(1X,2A)')CH,'[3;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
      WRITE(*,'(1X,2A)')CH,'[4;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
      WRITE(*,'(1X,2A)')CH,'[5;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
      WRITE(*,'(1X,2A)')CH,'[6;41H'
      WRITE(*,'('' [7m'',''             '',''[0m'')')
C     READ DATA FROM SCREEN
      WRITE(*,'(1X,2A)')CH,'[3;43H'
      READ(*,'(A15)') INPNAM
```

```
 406    WRITE(*,'(1X,2A)')CH,'[4;43H'
        READ(*,'(I8)',ERR=420) LSTART
 409    WRITE(*,'(1X,2A)')CH,'[5;43H'
        READ(*,'(F5.0)',ERR=424) ALAT
 411    WRITE(*,'(1X,2A)')CH,'[6;43H'
        READ(*,'(I8)',ERR=426) NYR2
C       POSITION CURSOR, SET Y AND N KEY WITH RETURN
 412    WRITE(*,'(1X,2A)')CH,'[9;2H'
        WRITE(*,'(1X,2A)')CH,'[89;89;13p'
        WRITE(*,'(1X,2A)')CH,'[78;78;13p'
        WRITE(*,'(1X,2A)')CH,'[121;121;13p'
        WRITE(*,'(1X,2A)')CH,'[110;110;13p'
        WRITE(*,415)
 415    FORMAT(///,1X,'[7m',' EVERYTHING CORRECT? (Y/N)','[0m',2X,)
        READ(*,'(A1)') DH
C       RESET Y AND N KEY WITHOUT RETURN
        WRITE(*,'(1X,2A)')CH,'[89;89p'
        WRITE(*,'(1X,2A)')CH,'[78;78p'
        WRITE(*,'(1X,2A)')CH,'[121;121p'
        WRITE(*,'(1X,2A)')CH,'[110;110p'
        N=ICHAR(DH)
C       JUMP CONTROL
        IF(N. EQ. 89) THEN
           GOTO 425
        ELSEIF(N .EQ. 121) THEN
           GOTO 425
        ELSEIF(N. EQ. 78) THEN
            GOTO 400
        ELSEIF(N .EQ. 110) THEN
            GOTO 400
        ELSE
            GOTO 412
        ENDIF
C******* Error in input routine********
 420    WRITE(*,'(1X,2A)')XH
        WRITE(*,'(1X,2A)')CH,'[4;41H'
        WRITE(*,'('' [7m'',''                '',''[0m'')')
        GOTO 406
 424    WRITE(*,'(1X,2A)')XH
        WRITE(*,'(1X,2A)')CH,'[5;41H'
        WRITE(*,'('' [7m'',''                '',''[0m'')')
        GOTO 409
 426    WRITE(*,'(1X,2A)')XH
        WRITE(*,'(1X,2A)')CH,'[6;41H'
        WRITE(*,'('' [7m'',''                '',''[0m'')')
        GOTO 411
C*********Start of the calcualtion and generation*****
 425    WRITE(*,'(1X,2A)')CH,'[2J'
        WRITE(*,'(1X,2A)')CH,'[10;35H'
        WRITE(*,'('' [5m'',''COMPUTING'',''[0m'')')
        CALL LATITU(ALAT,I1,I2)
        CALL TWGEN(ALAT,I1,I2,NYR1,INPNAM,MDAY,MMO,MYR,MH,MM,MS)
        CALL XSUM(NYR1,LSTART,0,MDAY,MMO,MYR,MH,MM,MS,INPNAM)
        CALL TEST(1,NYR2,INPNAM,ALAT,MDAY,MMO,MYR,MH,MM,MS,I1,I2)
        CALL XSUM(NYR2,1,1,MDAY,MMO,MYR,MH,MM,MS,INPNAM)
```

```
      WRITE(*,'(1X,2A)')CH,'[2J'
 500  WRITE(*,'(1X,2A)')XH
      WRITE(*,'(1X,2A)')CH,'[45;45p'
      END
      SUBROUTINE TEST(NTEST,NYRS,INPNAM,ALAT,MDAY,MMO,MYR,MH,MM,MS,
     $ I1,I2)
C**********************************************************************
C     MAIN GENERATION SUBROUTINE
C     PROGRAMMED BY I. SUPIT (HOPELESS PRODUCTIONS)
C     JANUARY 1986
C
C     CREATES DAILY VALUES FOR TMAX,TMIN,RAD,RAIN AND WIND FOR
C     NYRS (=NUBER OF YEARS). THE INPUT CAN COME FROM SUBROUTINE
C     TWGEN OR FROM AN EXTERNAL DATA FILE WHICH CONTAINS THE
C     PARAMETERS. THE RESULTS OF THE GENERATION ARE SUMMARIZED
C     IN THE EXTERNAL DATAFILE SUMWEATH.DAT. THE COMPLETE
C     SIMULATED DATASET IS PRINTED IN WEATHER.DAT.
C
C     NTEST =0 TEST PARAMETER: READ PARAMETERS FROM EXTERNAL FILE
C              INPNAM
C     NTEST =1 TEST PARAMETER: READ PARAMETERS FROM INTERNAL FILE
C     NYRS     NUMBER OF YEARS WHICH HAVE TO BE SIMULATED
C     ALAT     LATITUDE
C     MDAY,MMO,MYR,MH,MM,MS  = DAY,MONTH,YEAR,HOUR,MINUTE AND SECOND
C**********************************************************************
      IMPLICIT REAL(A-H,O-Z)
      IMPLICIT INTEGER(I-N)
      COMMON /DAT1/ TMX(20,366)
      COMMON /DAT2/ TMN(20,366)
      COMMON /DAT3/ RN(20,366)
      COMMON /DAT4/ RD(20,366)
      COMMON /DAT5/ RTO(20,366)
      COMMON /DAT14/ HUM(20,366)
      COMMON /DAT6/ TXM(366),TXS(366),TXM1(366),TXS1(366),TNM(366)
      COMMON /DAT7/ PWD(12),PWW(12),ALPHA(12),BETA(12)
      COMMON /DAT8/ RAIN(366),TMIN(366),TMAX(366),RAD(366),TNS(366)
      COMMON /DAT9/ RMO(366),RSO(366),RM1(366),RS1(366),RC(366)
      COMMON /DAT10/ RCF(12)
      COMMON /DAT12/ ALPW(12),BETW(12),WIND(365)
      COMMON /DAT11/ WN(20,366)
      COMMON /PARAM/ PM(40),JCT
      INTEGER*2 MH,MM,MYR,MMO,MDAY,MS,MHS
      CHARACTER*15 INPNAM
      CHARACTER*5 DATE(12)
      CHARACTER*30 NAME
      CHARACTER*15 FNAME(20)
      CHARACTER CX
      DIMENSION NI(12),NII(12)
      DIMENSION PW(12),RG(12),RATIO(366)
      DIMENSION ZTMAX(366),ZTMIN(366),ZRAD(366)
C**********************************************************************
C     NI   = JULIAN DATE OF THE END OF EACH MONTH OF NORMAL YEARS
C     NII  = JULIAN DATE OF THE END OF EACH MONTH OF LEAP YEARS
C     PW    = PROBABILITY OF A WET DAY
C     RG    = MONTHLY RAINFALL CORRECTION FACTOR
```

```
C       RATIO = RATIO ACTUAL RADIATION/POTENTIAL RADIATION FOR 1 YEAR
C       ZTMAX = ONE YEAR SIMULATED TMAX (C)
C       ZTMIN = ONE YEAR SIMULATED TMIN (C)
C       ZRAD  = ONE YEAR OF SIMULATED RAD (J/CM2/DAY)
C*********************************************************************
        DATA NI/31,59,90,120,151,181,212,243,273,304,334,365/
        DATA NII/31,60,91,121,152,182,213,244,274,305,335,366/
        DATA DATE/ 'JAN.','FEB.','MAR.','APR','MAY','JUN.','JUL.',
     $          'AUG.','SEP.','OCT.','NOV.','DEC.'/
        DATA FNAME/'A1','A2','A3','A4','A5','A6','A7','A8','A9','A10',
     $  'A11','A12','A13','A14','A15','A16','A17','A18','A19','A20'/
C**********************************************************************
C*           KGEN - GENERATION OPTION CODE
C*                  IF KGEN = 1,RAIN, MAX TEMP, MIN TEMP, AND
C*                  SOLAR RADIATION WILL BE GENERATED
C*                  IF KGEN = 2 OBSERVED RAIN WILL BE USED AND
C*                  MAX TEMP, MIN TEMP, SOLAR RADIATION WILL
C*                  BE GENERATED
C*                  (IN THIS SUBROUTINE ONLY KGEN=1 WILL BE USED)
C*           KTCF - TEMP. CORRECTION FACTOR OPTION CODE
C*                  IF KTCF = 0 NO TEMP CORRECTION WILL BE MADE
C*                  IF KTCF = 2 GENERATED MAX TEMP AND
C*                  MIN TEMP. WILL BE CORRECTED BASED ON
C*                  OBSERVED MEAN MONTHLY MAX AND MIN TEMP
C*                  IF KTCF = 1 GENERATED MAX TEMP AND MIN TEMP
C*                  WILL BE CORREATED BASED ON OBSERVED MEAN
C*                  MONTHLY TEMP
C*                  (IN THIS SUBROUTINE ONLY KTCF=0 WILL BE USED)
C*           KRCF - RAIN CORRECTION FACTOR OPTION CODE
C*                  IF KRCF = 1 GENERATED RAIN WILL BE CORRECTED
C*                  BASED ON OBSERVED MEAN MONTHLY RAIN
C*                  IF KRCF = 0 NO RAIN CORRECTION WILL BE MADE
C*                  (IN THIS SUBROUTINE ONLY KRCF=0 WILL BE USED)
C******************************************************************
        KGEN=1
        KTCF=0
        KRCF=0
        CX=CHAR(27)
C****************************************************************************
C                    INPUT VARIABLES
C
C       PWW(I)     PROBABILITY OF A WET DAY IF PREVIOUS DAY WAS WET
C       PWD(I)     PROBABILITY OF A WET DAY IF PREVIOUS DAY WAS DRY
C       ALPHA(I)   GAMMA DISTRIBUTION SHAPE PARAMETER OF THE RAIN PARAM.
C       BETA(I)    GAMMA DISTRIBUTION SCALE PARAMETER OF THE RAIN PARAM.
C       ALPW(I)    GAMMA DISTRIBUTION SHAPE PARAMETER OF THE WIND PARAM.
C       BETW(I)    GAMMA DISTRIBUTION SCALE PARAMETER OF THE WIND PARAM.
C       ALL INPUTS CONTAIN 12 MONTHLY VALUES
C**********************************************************************
C*                   ---- FIRST COSINE ----
C*      COSINE COEFFICIENTS OF MAX TEMP ON DRY DAYS
C*      PM(1)  - MEAN OF TMAX - DRY
C*      PM(2)  - AMPLITUDE OF TMAX - WET OR DRY
C*      PM(3)  - MEAN OF COEF. OF VAR. OF TMAX - WET OR DRY
C*      PM(4)  - AMPLITUDE OF COEF.OF VAR. OF TMAX - WET OR DRY
```

```
C*      COSINE COEFFICIENTS OF MAX TEMP ON WET DAYS
C*      PM(5)  - MEAN OF TMAX - WET
C*      PM(6)  -AMPLITUDE OF TMAX - DRY
C*      PM(7)  -  MEAN OF THE COEF. OF VAR. OF TMAX - WET
C*      PM(8)  - AMPLITUDE OF THE COEF. OF VAR. OF TMAX - WET
C*      COSINE COEFFICIENTS OF MIN TEMP
C*      PM(9)  - MEAN OF TMIN - WET OR DRY
C*      PM(10) - AMPLITUDE OF TMIN - WET OR DRY
C*      PM(11) - MEAN OF COEF. OF VAR. OF TMIN - WET OR DRY
C*      PM(12) - AMPLITUDE OF COEF. OF VAR. OF TMIN - WET OR DRY
C*      COSINE COEFFICIENTS OF RAD ON DRY DAYS
C*      PM(13) - MEAN OF RAD - DRY
C*      PM(14) - AMPLITUDE OF RAD -  DRY
C*      PM(15) - MEAN OF THE COEF. OF VAR. OF RAD -DRY
C*      PM(16) - AMPLTUDE OF THE COEF. OF VAR. OF RAD -DRY
C*      COSINE COEFFICIENTS OF RAD ON WET DAYS
C*      PM(17) - MEAN OF RAD - WET
C*      PM(18) - AMPLITUDE OF RAD -  WET
C*      PM(19) - MEAN OF THE COEF. OF VAR. OF RAD - WET
C*      PM(20) - AMPLTUDE OF THE COEF. OF VAR. OF RAD - WET
C***********************************************************************
C*              ---- SECOND COSINE ----
C*      COSINE COEFFICIENTS OF MAX TEMP ON DRY DAYS
C*      PM(21)  - MEAN OF TMAX - DRY
C*      PM(22)  - AMPLITUDE OF TMAX - WET OR DRY
C*      PM(23)  - MEAN OF COEF. OF VAR. OF TMAX - WET OR DRY
C*      PM(24)  - AMPLITUDE OF COEF.OF VAR. OF TMAX - WET OR DRY
C*      COSINE COEFFICIENTS OF MAX TEMP ON WET DAYS
C*      PM(25)  - MEAN OF TMAX - WET
C*      PM(26)  -AMPLITUDE OF TMAX - DRY
C*      PM(27)  -  MEAN OF THE COEF. OF VAR. OF TMAX - WET
C*      PM(28)  - AMPLITUDE OF THE COEF. OF VAR. OF TMAX - WET
C*      COSINE COEFFICIENTS OF MIN TEMP
C*      PM(29)  - MEAN OF TMIN - WET OR DRY
C*      PM(30) - AMPLITUDE OF TMIN - WET OR DRY
C*      PM(31) - MEAN OF COEF. OF VAR. OF TMIN - WET OR DRY
C*      PM(32) - AMPLITUDE OF COEF. OF VAR. OF TMIN - WET OR DRY
C*      COSINE COEFFICIENTS OF RAD ON DRY DAYS
C*      PM(33) - MEAN OF RAD - DRY
C*      PM(34) - AMPLITUDE OF RAD -  DRY
C*      PM(35) - MEAN OF THE COEF. OF VAR. OF RAD -DRY
C*      PM(36) - AMPLTUDE OF THE COEF. OF VAR. OF RAD -DRY
C*      COSINE COEFFICIENTS OF RAD ON WET DAYS
C*      PM(37) - MEAN OF RAD - WET
C*      PM(38) - AMPLITUDE OF RAD -  WET
C*      PM(39) - MEAN OF THE COEF. OF VAR. OF RAD - WET
C*      PM(40) - AMPLTUDE OF THE COEF. OF VAR. OF RAD - WET
C***********************************************************************
C       READ IN PARAMETERS FROM UNIT 20
        IF(NTEST. EQ. 0) THEN
        OPEN(20,STATUS='OLD',FILE=INPNAM)
        READ(20,60) (PWW(J),J=1,12)
        READ(20,60) (PWD(J),J=1,12)
        READ(20,60) (ALPHA(J),J=1,12)
        READ(20,60) (BETA(J),J=1,12)
```

```
      READ(20,60) (ALPW(J),J=1,12)
      READ(20,60) (BETW(J),J=1,12)
 60   FORMAT(10X,12F7.3)
      IF (ALAT. GE. 23.5 .OR. ALAT .LE. -23.5) THEN
      READ(20,100) PM(1),PM(2)
      READ(20,105) PM(3),PM(4)
      READ(20,105) PM(5),PM(6)
      READ(20,105) PM(7),PM(8)
      READ(20,105) PM(9),PM(10)
      READ(20,105) PM(11),PM(12)
      READ(20,105) PM(13),PM(14)
      READ(20,105) PM(15),PM(16)
      READ(20,105) PM(17),PM(18)
      READ(20,105) PM(19),PM(20)
 100  FORMAT(///,20X,F10.3,F10.3)
 105  FORMAT(20X,F10.3,F10.3)
      ELSE
      READ(20,200) PM(1),PM(2)
      READ(20,205) PM(3),PM(4)
      READ(20,205) PM(5),PM(6)
      READ(20,205) PM(7),PM(8)
      READ(20,205) PM(9),PM(10)
      READ(20,205) PM(11),PM(12)
      READ(20,205) PM(13),PM(14)
      READ(20,205) PM(15),PM(16)
      READ(20,205) PM(17),PM(18)
      READ(20,205) PM(19),PM(20)
 200  FORMAT(///,20X,F10.3,F10.3)
 205  FORMAT(20X,F10.3,F10.3)
      READ(20,300) PM(21),PM(22)
      READ(20,305) PM(23),PM(24)
      READ(20,305) PM(25),PM(26)
      READ(20,305) PM(27),PM(28)
      READ(20,305) PM(29),PM(30)
      READ(20,305) PM(31),PM(32)
      READ(20,305) PM(33),PM(34)
      READ(20,305) PM(35),PM(36)
      READ(20,305) PM(37),PM(38)
      READ(20,305) PM(39),PM(40)
 300  FORMAT(///,20X,F10.3,F10.3)
 305  FORMAT(20X,F10.3,F10.3)
      END IF
      CLOSE(20)
      ENDIF
      OPEN(20,STATUS='NEW',FILE='WEATHER.DAT')
      WRITE(20,811) MDAY,MMO,MYR
      WRITE(20,822) MH,MM,MS
      WRITE(20,833) INPNAM
 811  FORMAT(/,' RESULTS OF THE GENERATION',
     $ /,' CREATED ON ',I2,'-',I2,'-',I4)
 822  FORMAT(' TIME       ',I2,':',I2,':',I2)
 833  FORMAT(' BASED ON: ',A15,//)
C*****************************************************************
C     GENERATION OF THE PARAMETERS OF
      "C            _"C
```

```
C       X  = X  (1 + d  C  )
C        ij   ij      ij ij
C       WITH THE COSINE SERIE
C
C       XCR1  DAILY GENERATED COEF. OF VAR. FOR MAX. TEMP. ON WET
C             OR DRY DAYS
C       XCR2  DAILY GENERATED COEF. OF VAR. FOR MIN. TEMP.
C       XCR3  DAILY GENERATED COEF. OF VAR. FOR RAD. ON DRY DAYS
C       XCR4  DAILY GENERATED COEF. OF VAR. FOR RAD. ON WET DAYS
C*******************************************************************
        IF(ALAT .GE. 23.5 .OR. ALAT .LE. -23.5) THEN
          IF(ALAT .GT. 0.) THEN
            S1=200.
          ELSE
            S1=20.
          END IF
          S2=FLOAT(I1)
          CALL COSF1(1,365,0.0172,S1,S2)
          ELSE
              X1=365.-FLOAT(I2-I1)
              Y1=FLOAT(I2-I1)
              Z1=365.-FLOAT(I2-I1)
              QPI=6.28318/X1
              S1=FLOAT(I1)
              CALL COSF1(1,I1,QPI,S1,S1)
              QPI=6.28318/Y1
              S2=FLOAT(I2)
              CALL COSF2(I1,I2,QPI,S2,S2)
              QPI=6.28318/Z1
              S2=FLOAT(I2)
              CALL COSF1(I2,365,QPI,S2,S2)
        END IF
      DO 22 IM=1,12
        RCF(IM) = 1.0
        PW(IM) = PWD(IM)/(1. -PWW(IM)+PWD(IM))
 22   CONTINUE
C       CALCULATE MONTHLY RAINFALL CORRECTION FACTOR
        ZN=1/31
        RG(IM) = ALPHA(IM)*BETA(IM)*ZN*PW(IM)
C************************************************************
C       IDA = COUNTER FOR THE DAYS IN A MONTH
C       IM  = COUNTER FOR THE MONTHS
C       IDAYS = NUMBER OF DAYS IN A YEAR
C************************************************************
      DO 40 I = 1,NYRS
        IYR = I
        IDAYS = 365
C       TEST IF I IS A LEAP YEAR
        IFLG = MOD(IYR,4)
        IF(IFLG .EQ. 0) IDAYS = 366
C********************************************
C       CALLING THE SUMULATION ROUTINES
C********************************************
        CALL WGEN(KGEN,IDAYS,NI,NII)
        CALL GENWIN(NI)
```

```
      IM = 1
      IDA = 0
      WRITE(20,851)
851   FORMAT(//,' SIMULATED DAILY WEATHER VALUES',//)
      WRITE(20,852)
852   FORMAT(' MONTH',2X,'DATE',2X,'YEAR',2X,'JUL.DATE',2X,'RAINFALL',
     $ 2X,'MAX TEMP',2X,'MIN TEMP',2X,'SOLAR RAD',5X,'RATIO',4X,
     $ 'WINDSP.',4X,'HUM',/
     $ ,31X,' (MM)',5X,'(C)',7X,'(C)',6X,'(MJ/M2/DAY)',12X,'(M/S)',
     $ 5X,'(KPa)',)
C     DETERMINE COUNTERS
      DO 30 J=1,IDAYS
      IDA = IDA + 1
      IF(IDAYS .EQ. 366) THEN
         IF(J .GT. NII(IM)) THEN
            IM = IM + 1
            IDA = 1
         ENDIF
      ELSE
         IF(J .GT.NI(IM)) THEN
            IM = IM + 1
            IDA = 1
         ENDIF
      ENDIF
C*******************************************************************
C    THE FOLLOWING STATEMENTS WRITES DAILY GENERATED WEATHER ON AN
C    EXTERNAL FILE (UNIT 20).
C*******************************************************************
C     WRITING THE YEARLY GENERATION RESULTS TO A IYR MATRIX
      RN(IYR,J)=RAIN(J)
      TMX(IYR,J)=TMAX(J)
      TMN(IYR,J)=TMIN(J)
      RD(IYR,J)=RAD(J)
      RTO(IYR,J)=RAD(J)/RC(J)
      WN(IYR,J)=WIND(J)
C*******************************************************************
C     CALCULATING VAP. PRESS. BASED ON THE MINIMUM TEMPERATURE.
C     SHOULD BE CALCULATED WITH THE DEWPOINT TEMPERATURE
C*******************************************************************
    HUM(IYR,J)=6.11*EXP(17.47*TMIN(J)/(TMIN(J)+239.))*0.1
    WRITE(20,288)IM,IDA,IYR,J,RAIN(J),TMAX(J),TMIN(J),RAD(J)
   $,RATIO(J),WIND(J),HUM(IYR,J)
800 CONTINUE
C*******************************************************************
C    THE FOLLOWING STATEMENTS WRITES DAILY GENERATED WEATHER ON AN
C    EXTERNAL FILE (UNIT 20).
C*******************************************************************
288 FORMAT(2X,I2,5X,I2,4X,I2,5X,I3,6X,F5.1,3X,F5.1,5X,F5.1,8X
   $,F5.2,6X,F4.2,6X,F4.1,7X,F4.2)
30  CONTINUE
40  CONTINUE
      CLOSE(20)

C*****************************************
C     CREATING INPUT TABLES FOR CSMP
```

```
C****************************************
      DO 930 I=1,NYRS
      NAME=FNAME(I)
      OPEN(15,STATUS='NEW',FILE=NAME)
      WRITE(15,712) NAME,I,MDAY,MMO,MYR
      WRITE(15,722) MH,MM,MS
      WRITE(15,723) INPNAM,ALAT
 712  FORMAT(' TITLE ',A2,': SIMULATED WEATHER DATA YEAR ',I2,1X,
     $ '* CREATED ON ',I2,'-',I2,'-',I4)
 722  FORMAT('* TIME         ',I2,':',I2,':',I2)
 723  FORMAT('* BASED ON FILE :',A10,/,'PARAM LAT=',F5.0)
      WRITE(15,124) (TMX(I,J),J=1,365)
 124  FORMAT(/,' TABLE TMPHT(1-365)=...',/
     $ ,37(10(F6.1,','),'...',/))
      WRITE(15,925) (TMN(I,J),J=1,365)
 925  FORMAT(/,' TABLE TMPLT(1-365)=...',/
     $ ,37(10(F6.1,','),'...',/))
      WRITE(15,126) (RD(I,J),J=1,365)
 126  FORMAT(/,' TABLE RDTMT(1-365)=...',/
     $ ,37(10(F6.1,','),'...',/))
      WRITE(15,127) (RN(I,J),J=1,365)
 127  FORMAT(/,' TABLE RAINT(1-365)=...',/
     $ ,37(10(F6.1,','),'...',/))
      WRITE(15,128) (WN(I,J),J=1,365)
 128  FORMAT(/,' TABLE WDST(1-365)=...',/
     $ ,37(10(F6.1,','),'...',/))
      WRITE(15,133) (HUM(I,J),J=1,365)
 133  FORMAT(/,' TABLE HUAD(1-365)=...',/
     $ ,37(10(F6.2,','),'...',/))
      CLOSE(15)
 930  CONTINUE
        RETURN
        END
      SUBROUTINE COSF1(L1,L2,QPI,S1,S2)
      COMMON /DAT6/ TXM(366),TXS(366),TXM1(366),TXS1(366),TNM(366)
      COMMON /DAT8/ RAIN(366),TMIN(366),TMAX(366),RAD(366),TNS(366)
      COMMON /DAT9/ RMO(366),RSO(366),RM1(366),RS1(366),RC(366)
      COMMON /PARAM/ PM(40),JCT
      DO 11 J = L1,L2
           XJ = FLOAT(J)
           DT = COS(QPI*(XJ-S1))
           DR = COS(QPI*(XJ-S2))
           TXM(J)  = PM(1)+PM(2)*DT
           XCR1    = PM(3)+PM(4)*DT
           IF(XCR1 .LT. 0.0) XCR1=0.06
           TXS(J)  = TXM(J)*XCR1
           TXM1(J) = PM(5)+PM(6)*DT
           XCR1W   = PM(7)+PM(8)*DT
           IF(XCR2 .LT. 0.0) XCR2=0.06
           TXS1(J) = TXM1(J)*XCR2
           TNM(J)  = PM(9) + PM(10)*DT
           XCR2    = PM(11) + PM(12)*DT
           IF(XCR3 .LT. 0.0) XCR3=0.06
           TNS(J)  = TNM(J)*XCR3
           RMO(J)  = PM(13) + PM(14)*DR
```

```
            XCR3     = PM(15) + PM(16)*DR
            IF(XCR4 .LT. 0.0)XCR4=0.06
            RSO(J)   = RMO(J)*XCR4
            RM1(J)   = PM(17) + PM(18)*DR
            XCR4     = PM(19) + PM(20)*DR
            IF(XCR5 .LT. 0.0) XCR5=0.06
            RS1(J)   = RM1(J)*XCR5
 11       CONTINUE
          RETURN
          END
        SUBROUTINE COSF2(L1,L2,QPI,S1,S2)
        COMMON /DAT6/ TXM(366),TXS(366),TXM1(366),TXS1(366),TNM(366)
        COMMON /DAT8/ RAIN(366),TMIN(366),TMAX(366),RAD(366),TNS(366)
        COMMON /DAT9/ RMO(366),RSO(366),RM1(366),RS1(366),RC(366)
        COMMON /PARAM/ PM(40),JCT
        DO 11 J = L1,L2
            XJ = FLOAT(J)
            DT = COS(QPI*(XJ-S1))
            DR = COS(QPI*(XJ-S2))
            TXM(J)   = PM(21)+PM(22)*DT
            XCR1     = PM(23)+PM(24)*DT
            IF(XCR1 .LT. 0.0) XCR1=0.06
            TXS(J)   = TXM(J)*XCR1
            TXM1(J) = PM(25)+PM(26)*DT
            XCR1W    = PM(27)+PM(28)*DT
            IF(XCR2 .LT. 0.0) XCR2=0.06
            TXS1(J) = TXM1(J)*XCR2
            TNM(J)   = PM(29) + PM(30)*DT
            XCR2     = PM(31) + PM(32)*DT
            IF(XCR3 .LT. 0.0) XCR3=0.06
            TNS(J)   = TNM(J)*XCR3
            RMO(J)   = PM(33) + PM(34)*DR
            XCR3     = PM(35) + PM(36)*DR
            IF(XCR4 .LT. 0.0) XCR4=0.06
            RSO(J)   = RMO(J)*XCR4
            RM1(J)   = PM(37) + PM(38)*DR
            XCR4     = PM(39) + PM(40)*DR
            IF(XCR5 .LT. 0.0) XCR5 = 0.06
            RS1(J)   = RM1(J)*XCR5
 11       CONTINUE
          RETURN
          END
      SUBROUTINE WGEN(KGEN,IDAYS,NI,NII)
C*****************************************************************
C     THE FOLLOWING SUBROUTINE GENERATES DAILY WEATHER DATA FOR
C     ONE YEAR.
C     KGEN = GENERATION OPTION CODE (SEE SUBROUTINE TEST)
C     IDAYS= NUMBER OF DAYS IN A YEAR
C     NI    = JULIAN DATE OF THE END OF EACH MONTH OF NORMAL YEARS
C     NII   = JULIAN DATE OF THE END OF EACH MONTH OF LEAP YEARS
C*****************************************************************
        COMMON /DAT6/ TXM(366),TXS(366),TXM1(366),TXS1(366),TNM(366)
        COMMON /DAT7/ PWD(12),PWW(12),ALPHA(12),BETA(12)
        COMMON /DAT8/ RAIN(366),TMIN(366),TMAX(366),RAD(366),TNS(366)
        COMMON /DAT9/ RMO(366),RSO(366),RM1(366),RS1(366),RC(366)
```

```
        COMMON /DAT10/ RCF(12)
        DIMENSION A(3,3),B(3,3),XIM1(3),E(3),R(3),X(3),RR(3)
        DIMENSION TCF(12),NI(12),NII(12)
      DATA A/0.567,0.253,-0.006,0.086,0.504,-0.039,-0.002,
     *-0.050,0.244/
      DATA B/0.781,0.328,0.238,0.0,0.637,-0.341,0.0,0.0,0.873/
      DATA XIM1/0.,0.,0./
      DATA IX/9398039/
      DATA IP/0/
C****************************************************************
C       A AND B ARE REGRESSION MATRICES, E IS MATRIX WITH THE
C       RANDOM VARIABLES
C****************************************************************
        IM = 1
        DO 50 IDAY=1,IDAYS
        IF(IDAYS .EQ. 366) THEN
           IF(IDAY .GT. NII(IM)) THEN
              IM = IM + 1
           ENDIF
        ELSE
           IF(IDAY .GT.NI(IM)) THEN
              IM = IM + 1
           ENDIF
        ENDIF
           IF(KGEN .EQ. 2) GO TO 15
C********************************************************
C     DETERMINE WET OR DRY DAY USING MARKOV CHAIN MODEL
C********************************************************
           CALL RANDN(RN)
           IF(IP-0) 7,7,10
 7         IF(RN - PWD(IM  ))11,11,8
 8         IP = 0
           RAIN(IDAY) = 0.
           GOTO 18
 10        IF(RN-PWW(IM  ))11,11,8
 11        IP = 1
C*********************************************************
C     DETERMINE RAINFALL AMOUNT FOR WET DAYS USING GAMMA
C     DISTRIBUTION
C*********************************************************
        AA = 1./ALPHA(IM)
        AB = 1./(1.-ALPHA(IM))
        TR1 = EXP(-18.42/AA)
        TR2 = EXP(-18.42/AB)
        SUM = 0.
        SUM2 = 0.
 12   CALL RANDN(RN1)
      CALL RANDN(RN2)
        IF((RN1-TR1) .LE. 0.) THEN
           S1 = 0.
        ELSE
           S1 = RN1**AA
        END IF
        IF((RN2-TR2) .LE. 0.) THEN
           S2 = 0.
```

```
         ELSE
            S2 = RN2**AB
         END IF
         S12 = S1 + S2
         IF(S12-1.) 13,13,12
 13      Z = S1/S12
       CALL RANDN(RN3)
         RAIN(IDAY) = -Z*ALOG(RN3)*BETA(IM)*RCF(IM)
C**********************************************
C      RAIN(IDAY) IS GENERATED RAINFALL FOR IDAY
C**********************************************
 15      IF(RAIN(IDAY) .LE. 0.) THEN
            IP = 0
         ELSE
            IP = 1
         END IF
C        GENERATE TMAX,TMIN, AND RAD FOR IDAY
 18      IF((IP-1) .LT. 0.) THEN
            RM   = RMO(IDAY)
            RS   = RSO(IDAY)
            TXXM = TXM(IDAY)
            TXXS = TXS(IDAY)
         ELSE
           RM   = RM1(IDAY)
           RS   = RS1(IDAY)
           TXXM = TXM1(IDAY)
           TXXS = TXS1(IDAY)
         END IF
       DO 30 K = 1,3
 131     AA = 0.
       CALL RANDN(RN1)
       CALL RANDN(RN2)
         V = SQRT(-2.*ALOG(RN1))*COS(6.283185*RN2)
         IF(ABS(V) .GT. 2.5) GO TO 131
         E(K) = V
 30    CONTINUE
       DO 31 I = 1,3
          R(I) = 0.
          RR(I) = 0.
 31    CONTINUE
         DO 32 I = 1,3
         DO 32 J = 1,3
            R(I) = R(I)+B(I,J)*E(J)
            RR(I) = RR(I) + A(I,J)*XIM1(J)
 32      CONTINUE
         DO 37 K = 1,3
            X(K) = R(K) + RR(K)
            XIM1(K) = X(K)
 37      CONTINUE
         TMAX(IDAY) = X(1) * TXXS + TXXM
         TMIN(IDAY) = X(2)*TNS(IDAY)+TNM(IDAY)
         IF(TMIN(IDAY) .GT. TMAX(IDAY)) THEN
            TMM = TMAX(IDAY)
            TMAX(IDAY) = TMIN(IDAY)
            TMIN(IDAY) = TMM
```

```
        ENDIF
C********************************************
C     TMAX(IDAY) IS GENERATED TMAX FOR IDAY
C     TMIN(IDAY) IS GENERATED TMIN FOR IDAY
C     RAD(IDAY) IS GENERATED RAD FOR IDAY
C********************************************
        RAD(IDAY) = X(3)*RS+RM
        RMIN = 0.05*RC(IDAY)
        IF(RAD(IDAY) .LT. RMIN) RAD(IDAY) = RMIN
        IF (RAD(IDAY) .GT. RC(IDAY)) RAD(IDAY) = RC(IDAY)
 50   CONTINUE
      RETURN
      END
      SUBROUTINE RANDN(YFL)
C***********************************************************
C     THE FOLLOWING SUBROUTINE GENERATES A UNIFORM RANDOM
C     NUMBER ON THE INTERVAL 0 - 1
C***********************************************************
      DIMENSION K(4)
      INTEGER*2 MH,MM,MS,MHS
      DATA K/2510,7692,2456,3765/
C************************************************************
C     IF CALL GETTIM AND THE DO 10 LOOP ARE OMMITED THEN
C     THE SEEDS ARE ALWAYS THE SAME. SO EVERY TIME THE PRO-
C     GRAM GIVE THE SAME RESULTS
C************************************************************
C     CALL GETTIM(MH,MM,MS,MHS)
C
C     DO 10 I=1,4
C        K(I)=K(I)+MM*MS-MH+MHS
C10   CONTINUE
      K(4) = 3*K(4)+K(2)
      K(3) = 3*K(3)+K(1)
      K(2)=3*K(2)
      K(1)=3*K(1)
      I=K(1)/1000
      K(1)=K(1)-I*1000
      K(2)=K(2) + I
      I = K(2)/100
      K(2)=K(2)-100*I
      K(3) = K(3)+I
      I = K(3)/1000
      K(3)=K(3)-I*1000
      K(4)=K(4)+I
      I = K(4)/100
      K(4)=K(4)-100*I
      YFL=(((FLOAT(K(1))*.001+FLOAT(K(2)))*.01+FLOAT(K(3)))*.001
     *+FLOAT(K(4)))*.01
      YFL=ABS(YFL)
      RETURN
      END
        SUBROUTINE GENWIN(NI)
C*******************************************************************
C       GENERATION OF ONE YEAR OF WINDSPEED DATA
C       MINIMUM WINDSPEED WMIN=0.2
```

```
C       MAXIMUM WINDSPEED WMAX=9.5
C       THIS LIMITS ARE CHOSEN BECAUSE THE EFFECTS ON THE PLANT
C       GROWTH ARE MINIMAL WHEN THE WIND EXCEEDS THESE BOUNDARIES
C**********************************************************************
        COMMON /DAT12/ ALPW(12),BETW(12),WIND(365)
        DIMENSION NI(12)
C*************************************************************
C       LBEGIN = JULIAN DATE OF THE BEGINNING OF A MONTH
C       LEND   = JULIAN DATE OF THE END OF A MONTH
C       ALPH   = ALPHA
C       BET    = BETA
C**********************************************************
        WMIN=0.2
        WMAX=9.5
        DO 211 I=1,365
           WIND(I)=0.0
 211    CONTINUE
        DO 218 L=1,12
           ALPH=ALPW(L)
           BET=BETW(L)
           IF(L.GT.1) THEN
              LL=L-1
              LBEGIN=NI(LL)
           ELSE
              LL=1
              LBEGIN=1
           ENDIF
           LEND=NI(L)
        DO 212 I=LBEGIN,LEND
 213    WIND(I)=(AINT((GAMMAD(ALPH,BET)+0.005)*100.))/100.
        IF(WIND(I).LT.WMIN) GOTO 213
        IF(WIND(I).GT.WMAX) GOTO 213
 212    CONTINUE
 218    CONTINUE
        RETURN
        END
        FUNCTION  GAMMAD(ALPH,BET)
C*************************************************************
C       METHOD OF JOHN K
C*************************************************************
        Z=0.0
        K=ALPH
        F=K
        IF (K) 303,303,301
 301    PROD=1.0
        DO 302 L=1,K
           CALL RANDN(U)
 302       PROD=PROD*U
        Z=-ALOG(PROD)
 303    D=ALPH-F
        IF (D) 308,308,304
 304    A=1.0/D
        B=1.0/(1.0-D)
        L=1
```

```
 305      CALL RANDN(U)
        UA=-50/ALOG10(U)
        X=0.
        IF (A.LT.UA) X=U**A
        CALL RANDN(U)
        UB=-50/ALOG10(U)
        Y=X
        IF (B.LT.UB) Y=U**B+X
        IF (Y-1.0) 307,307,306
 306    L=L+2
        GOTO 305
 307    W=X/Y
        CALL RANDN(U)
        Y=-ALOG(U)
        GAMMAD=(Z+W*Y)*BET
        RETURN
 308    GAMMAD=Z*BET
        RETURN
        END
        SUBROUTINE LATITU(ALAT,I1,I2)
C*******************************************************************
C       CALCULATE MAXIMUM SOLAR RADIATION FOR EACH DAY  FOR
C       LATITUDE = ALAT
C*******************************************************************
        COMMON /DAT9/ RMO(366),RSO(366),RM1(366),RS1(366),RC(366)
        DIMENSION B(366),DMAX(2),IMAX(2)
        PI=3.1415926
        DO 6 I = 1,366
C       CALCULATION SUNHEIGHT AND RADIATION
C       RADIATION MJ/M2/DAY
           DEC    =-23.4*COS(2.*PI*(I+10.)/365.)
           COSLD =COS(DEC*PI/180.)*COS(ALAT*PI/180.)
           SINLD =SIN(DEC*PI/180.)*SIN(ALAT*PI/180.)
           SINB  =SINLD+COSLD
           B(I)=ASIN(SINB)*180./PI
           DLA    =12.*(PI+2.*ASIN(SINLD/COSLD))/PI
           DLE    =12.*(PI+2.*ASIN((-SIN(8.*PI/180.)+SINLD)/COSLD))/PI
           DLP    =12.*(PI+2.*ASIN((-SIN(-4.*PI/180.)+SINLD)/COSLD))/PI
        RDN =3600.*(SINLD*DLA+24./PI*COSLD*SQRT(1.-(SINLD/COSLD)**2))
           RDTC   =1370.*RDN*EXP(-0.1/(RDN/(DLA*3600.)))
           RC(I) = RDTC * 1.0 * 0.0001 * 0.01
 6      CONTINUE
        IM=1
        DMAX(1)=0.
        DMAX(2)=0.
        ITEST=0
        DEG=0.
        IF (ALAT .LT. 23.5 .AND. ALAT .GT. -23.5) THEN
        DO 10 I=1,365
          XT=DEG-B(I)
        IF (ITEST .EQ. 0) THEN
           IF (B(I) .GT. DMAX(1) .AND. XT .LT. 0.) THEN
             DMAX(1)=B(I)
              IMAX(1)=I
           END IF
```

```
      END IF
      IF (XT .GT.0) THEN
          ITEST=1
      END IF
      IF (ITEST .EQ. 1) THEN
         IF (B(I) .GT. DMAX(2) .AND. XT .LT. 0.) THEN
            DMAX(2)=B(I)
            IMAX(2)=I
         END IF
      END IF
      DEG=B(I)
 10   CONTINUE
        I1=IMAX(1)
        I2=IMAX(2)
      ELSEIF(ALAT .GE. 23.5) THEN
        I1=172
        I2=0
      ELSEIF(ALAT .LE. -23.5) THEN
        I1=355
        I2=0
      END IF
      RETURN
      END
      SUBROUTINE TWGEN(ALAT,I1,I2,NYRS,INPN,MDAY,MMO,MYR,MH,MM,MS)
C**********************************************************************
C     ORIGINAL PROGRAM BY RICHARDSON
C     REVISED BY I. SUPIT
C
C     THIS SUBROUTINE ESTITIMATES THE PARAMETERS WHICH ARE USED
C     IN THE GENERATION SUBROUTINE TEST.
C     THE INPUT DATA FILES CONSIST OF ACTUAL DATA. ALSO A
C     SUMMARY OF THE ACTUAL DATA IS GIVEN IN OUTPUT FILE 'SUMMAR.DAT'.
C     THE ESTIMATED PARAMETERS ARE GIVEN IN OUTPUT FILE 'OUTWGEN.DAT'.
C
C     NYRS = NUMBER OF AVALAILABLE YEARS IN DATA EXTERNAL DATA FILE
C            INPNAM
C     LSTART= STARTING YEAR OF DATA FILE (ONLY OPTINAL FOR PRINT OUT)
C     MDAY,MMO,MYR,MH,MM,MS  = DAY,MONTH,YEAR,HOUR,MINUTE AND SECOND
C**********************************************************************
      IMPLICIT REAL (A-H,O-Z)
      IMPLICIT INTEGER (I-N)
      COMMON /DAT1/ TMX(20,366)
      COMMON /DAT2/ TMN(20,366)
      COMMON /DAT3/ RN(20,366)
      COMMON /DAT4/ RD(20,366)
      COMMON /DAT5/ RTO(20,366)
      COMMON /DAT11/ WN(20,366)
      COMMON /DAT7/ PWD(12),PWW(12),ALPHA(12),BETA(12)
      COMMON /DAT9/ RMO(366),RSO(366),RM1(366),RS1(366),RC(366)
      COMMON /DAT12/ ALPW(12),BETW(12),WIND(365)
      COMMON /PARAM/ PM(40),JCT
      INTEGER*2 MH,MM,MYR,MMO,MDAY,MS,MHS
      DIMENSION A(40)
      CHARACTER CX
      CHARACTER*15 INPN
```

```
      CX=CHAR(27)
      JCT=1
      NYRS=1
      OPEN(11,STATUS='OLD',FILE=INPN)
 7       CALL INPUT(NYRS,MT)
      IF(MT.EQ.0) THEN
         NYRS=NYRS+1
         GOTO 7
      ELSE
         NYRS=NYRS-1
      ENDIF
      CLOSE(11)
      DO 4 I=1,20
         PM(I)=0.
 4    CONTINUE
      DO 8 I=1,NYRS
      DO 8 J=1,365
         IF(RD(I,J).GT. RC(J)) RD(I,J)=RC(J)
         RTO(I,J)=RD(I,J)/RC(J)
 8    CONTINUE
C*******************************************************
C               SUMMARY TABLES
C*******************************************************
      OPEN(20,STATUS='NEW',FILE='PARAMET.DAT')
      WRITE(20,10)
      WRITE(20,15) MDAY,MMO,MYR
      WRITE(20,20) MH,MM,MS
      WRITE(20,25) INPN
 10   FORMAT(' CALCULATED PARAMETERS ')
 15   FORMAT(' CREATED ON ',I2,'-',I2,'-',I4)
 20   FORMAT(' TIME        ',I2,':',I2,':',I2)
 25   FORMAT(' BASED ON FILE :',A15,//)
      WRITE(20,30)
 30   FORMAT(//5X,' MAXIMUM TEMPERATURE',/)
C***** CALCULATE TMX PARAMETERS   ******
      CALL MSD(ALAT,I1,I2,NYRS,TMX,1)
      WRITE(20,35)
 35   FORMAT(' 1'///5X,' MINIMUM TEMPERATURE',/)
C***** CALCULATE TMN PARAMETERS   *****
      CALL MSD(ALAT,I1,I2,NYRS,TMN,2)
      WRITE(20,40)
 40   FORMAT(' 1'///,5X,' SOLAR RADIATION',/)
C***** CALCULATE RD PARAMETERS   *****
      CALL MSD(ALAT,I1,I2,NYRS,RD,3)
      WRITE(20,45)
 45   FORMAT(' 1',///,5X,' PRECIPITATION')
C***** CALCULATE RAINFALL PARAMETERS  *****
      CALL PPRN(1,NYRS)
C***** CALCULATE WINDSPEED PARAMETERS  *****
      WRITE(20,50)
 50   FORMAT(' 1',///,5X,' WINDSPEED')
      CALL PPRN(2,NYRS)
C--AND SIGNS CHANGED ON AMPLITUDES
      WRITE(20,52)
 52   FORMAT(//,20X,' INPUT CARDS FOR THE WEATHER GENERATOR ARE AS
```

```
     $   FOLLOWS--------------'////)
         WRITE(20,54)
 54      FORMAT(///)
         WRITE(20,56) (PWW(J),J=1,12)
         WRITE(20,58) (PWD(J),J=1,12)
         WRITE(20,60) (ALPHA(J),J=1,12)
         WRITE(20,62) (BETA(J),J=1,12)
         WRITE(20,64) (ALPW(J),J=1,12)
         WRITE(20,66) (BETW(J),J=1,12)
 56      FORMAT(1X,'P(W/W)   ',12F7.3)
 58      FORMAT(1X,'P(W/D)   ',12F7.3)
 60      FORMAT(1X,'ALPHA    ',12F7.3)
 62      FORMAT(1X,'BETA     ',12F7.3)
 64      FORMAT(1X,'ALPWI    ',12F7.3)
 66      FORMAT(1X,'BETWI    ',12F7.3)
         IF (ALAT. GE. 23.5 .OR. ALAT .LE. -23.5) THEN
         WRITE(20,95)
         WRITE(20,100) PM(1),PM(2)
         WRITE(20,105) PM(3),PM(4)
         WRITE(20,110) PM(5),PM(6)
         WRITE(20,115) PM(7),PM(8)
         WRITE(20,120) PM(9),PM(10)
         WRITE(20,125) PM(11),PM(12)
         WRITE(20,130) PM(13),PM(14)
         WRITE(20,135) PM(15),PM(16)
         WRITE(20,140) PM(17),PM(18)
         WRITE(20,145) PM(19),PM(20)
 95      FORMAT(//,' PARAM. COS FUNCTION',4X,' MEAN',5X,' AMPL.')
 100     FORMAT(' MAX.TEMP. DRY  MEAN',F10.3,F10.3)
 105     FORMAT(' "    "     "   C.V.',F10.3,F10.3)
 110     FORMAT(' "    "     WET MEAN',F10.3,F10.3)
 115     FORMAT(' "    "      "  C.V.',F10.3,F10.3)
 120     FORMAT(' MIN.TEMP.      MEAN',F10.3,F10.3)
 125     FORMAT(' "    "         C.V.',F10.3,F10.3)
 130     FORMAT(' RADIATION  DRY MEAN',F10.3,F10.3)
 135     FORMAT(' "    "      "  C.V.',F10.3,F10.3)
 140     FORMAT(' "    "     WET MEAN',F10.3,F10.3)
 145     FORMAT(' "    "      "  C.V.',F10.3,F10.3)
         ELSE
         DO 146 I=1,40
            A(I)=PM(I)
 146     CONTINUE
C        ---FIRST COSINE---
         PM(1)=A(1)
         PM(2)=A(2)
         PM(3)=A(3)
         PM(4)=A(4)
         PM(5)=A(9)
         PM(6)=A(10)
         PM(7)=A(11)
         PM(8)=A(12)
         PM(9)=A(17)
         PM(10)=A(18)
         PM(11)=A(19)
         PM(12)=A(20)
```

```
      PM(13)=A(25)
      PM(14)=A(26)
      PM(15)=A(27)
      PM(16)=A(28)
      PM(17)=A(33)
      PM(18)=A(34)
      PM(19)=A(35)
      PM(20)=A(36)
C     ---SECOND COSINE---
      PM(21)=A(5)
      PM(22)=A(6)
      PM(23)=A(7)
      PM(24)=A(8)
      PM(25)=A(13)
      PM(26)=A(14)
      PM(27)=A(15)
      PM(28)=A(16)
      PM(29)=A(21)
      PM(30)=A(22)
      PM(31)=A(23)
      PM(32)=A(24)
      PM(33)=A(29)
      PM(34)=A(30)
      PM(35)=A(31)
      PM(36)=A(32)
      PM(37)=A(37)
      PM(38)=A(38)
      PM(39)=A(39)
      PM(40)=A(40)
      WRITE(20,195)
      WRITE(20,200) PM(1),PM(2)
      WRITE(20,205) PM(3),PM(4)
      WRITE(20,210) PM(5),PM(6)
      WRITE(20,215) PM(7),PM(8)
      WRITE(20,220) PM(9),PM(10)
      WRITE(20,225) PM(11),PM(12)
      WRITE(20,230) PM(13),PM(14)
      WRITE(20,235) PM(15),PM(16)
      WRITE(20,240) PM(17),PM(18)
      WRITE(20,245) PM(19),PM(20)
195   FORMAT(//,' PARAM COS FUNCTION(1)',2X,' MEAN',6X,' AMPL.')
200   FORMAT(' MAX.TEMP. DRY  MEAN',F10.3,F10.3)
205   FORMAT(' "    "      "  C.V.',F10.3,F10.3)
210   FORMAT(' "    "     WET MEAN',F10.3,F10.3)
215   FORMAT(' "    "       " C.V.',F10.3,F10.3)
220   FORMAT(' MIN.TEMP.      MEAN',F10.3,F10.3)
225   FORMAT(' "    "         C.V.',F10.3,F10.3)
230   FORMAT(' RADIATION  DRY MEAN',F10.3,F10.3)
235   FORMAT(' "    "       " C.V.',F10.3,F10.3)
240   FORMAT(' "    "     WET MEAN',F10.3,F10.3)
245   FORMAT(' "    "       " C.V.',F10.3,F10.3)
      WRITE(20,295)
      WRITE(20,300) PM(21),PM(22)
      WRITE(20,305) PM(23),PM(24)
      WRITE(20,310) PM(25),PM(26)
```

```
        WRITE(20,315) PM(27),PM(28)
        WRITE(20,320) PM(29),PM(30)
        WRITE(20,325) PM(31),PM(32)
        WRITE(20,330) PM(33),PM(34)
        WRITE(20,335) PM(35),PM(36)
        WRITE(20,340) PM(37),PM(38)
        WRITE(20,345) PM(39),PM(40)
 295    FORMAT(//,' PARAM COS FUNCTION(2)',2X,' MEAN',6X,' AMPL.')
 300    FORMAT(' MAX.TEMP. DRY  MEAN',F10.3,F10.3)
 305    FORMAT(' "     "      "   C.V.',F10.3,F10.3)
 310    FORMAT(' "     "     WET MEAN',F10.3,F10.3)
 315    FORMAT(' "     "       "  C.V.',F10.3,F10.3)
 320    FORMAT(' MIN.TEMP.       MEAN',F10.3,F10.3)
 325    FORMAT(' "     "         C.V.',F10.3,F10.3)
 330    FORMAT(' RADIATION  DRY MEAN',F10.3,F10.3)
 335    FORMAT(' "     "       "  C.V.',F10.3,F10.3)
 340    FORMAT(' "     "     WET MEAN',F10.3,F10.3)
 345    FORMAT(' "     "       "  C.V.',F10.3,F10.3)
        END IF
        RETURN
        END
        SUBROUTINE FOUR(ALAT,I1,I2,XM,SD,CV)
C*******************************************************************
C       THIS SUBROUTINE CALCULATES THE FOURIER COEFFICIENTS
C       BASED ON A 28-DAY PERIOD. INPUT DATA ARE ID,XM,SD AND CV.
C       OUTPUT IS XDATA.
C       H1 AND XM = ARRAYS OF THE MEAN VALUES
C       H2 AND SD = ARRAYS OF THE STANDARD DEVIATIONS
C       H3 AND CV = ARRAYS OF THE COEFFIECIENTS OF VARIATION
C       XM,SD,CV ARE BASED ON A 28-DAY PERIOD.
C*******************************************************************
        COMMON /HELP/ H1(54),H2(54),H3(54)
        COMMON /PARAM/ PM(40),JCT
        DIMENSION XM(52),SD(52),CV(52)
C*******************************************************************
C       FOURIER COEFFICIENTS  OF THE MEAN VALUES
C       XBAR = MEAN        (based on (C) and (MJ/M2/DAY)
C       C     = AMPLITUDE        "    "   "    "     "
C       T      = PHASE
C       FOURIER COEFFICIENTS OF THE STD. DEVIATION
C       XBAR1 = MEAN        (based on (C) and (MJ/M2/DAY)
C       C1     = AMPLITUDE       "    "   "    "     "
C       T1      = PHASE
C       FOURIER COEFFICIENTS OF THE COEFF. OF VAR.
C       XBAR2  = MEAN
C       C2      = AMPLITUDE
C       T2      = PHASE
C*********************************************************************
        DO 90 I=1,52
           WRITE(20,80) I,XM(I),SD(I),CV(I)
 80        FORMAT(30X,I10,3F10.2)
 90     CONTINUE
        X1=FLOAT(I1)/7.
        X2=FLOAT(I2)/7.
        L1=NINT(X1)
```

```
      L2=NINT(X2)
C*****************************************************************
C     CHANGING SEQUENCE IN THE ARRAYS OF MEAN, SD AND CV
C     IN ORDER TO SHIFT THE PHASE 180 DEGREES AND CHANGE THE
C     SIGN OF THE AMPLITUDE.
C*****************************************************************
      IF (ALAT .LT. 23.5 .AND. ALAT .GT. -23.5) THEN
         L=1
         DO 100 I=L2,52
            H1(L)=XM(I)
            H2(L)=SD(I)
            H3(L)=CV(I)
            L = L + 1
 100     CONTINUE
         DO 110 I=1,L1
            H1(L)=XM(I)
            H2(L)=SD(I)
            H3(L)=CV(I)
            L = L + 1
 110     CONTINUE
         DO 115 I=L1,L2
            H1(L)=XM(I)
            H2(L)=SD(I)
            H3(L)=CV(I)
             L = L + 1
 115     CONTINUE
      ELSE
         L = 1
         DO 120 I=L1,52
            H1(L)=XM(I)
            H2(L)=SD(I)
            H3(L)=CV(I)
            L = L + 1
 120     CONTINUE
         DO 125 I=1,L1-1
            H1(L)=XM(I)
            H2(L)=SD(I)
            H3(L)=CV(I)
            L = L + 1
 125     CONTINUE
      END IF
      IF (ALAT .GE. 23.5 .OR. ALAT .LE. -23.5) THEN
         LB = 1
         LE =52
         Z1 =52.
         WRITE(20,130)
 130     FORMAT(/,' COSINUS PARAMETERS')
         CALL FOUR2(LB,LE,Z1)
      ELSE
         LB=1
         LE=52-(L2-L1)+1
         M1=LE
         Z1 =FLOAT(M1)
         WRITE(20,135)
 135     FORMAT(/,' FIRST COSINE PARAMETERS')
```

```
          CALL FOUR2(LB,LE,Z1)
          LB=LE+1
          LE=LB+L2-L1
          M1=L2-L1+1
          Z1 =FLOAT(M1)
          WRITE(20,140)
140       FORMAT(/,' SECOND COSINE PARAMETERS')
          CALL FOUR2(LB,LE,Z1)
       END IF
       RETURN
       END
       SUBROUTINE FOUR2(LB,LE,Z1)
       COMMON /HELP/ H1(54),H2(54),H3(54)
       COMMON /PARAM/ PM(40),JCT
       S=0.
       S1=0.
       S2=0.
       DO 10 I=LB,LE
          S=S+H1(I)
          S1=S1+H2(I)
          S2=S2+H3(I)
10     CONTINUE
       XBAR=S/Z1
       XBAR1=S1/Z1
       XBAR2=S2/Z1
       SUMA=0.
       SUMB=0.
       SUMA1=0.
       SUMA2=0.
       SUMB1=0.
       SUMB2=0.
       XK=1
       DO 15 K=LB,LE
          SUMA=SUMA+(H1(K)-XBAR)*COS(6.2832*XK/Z1)
          SUMA1=SUMA1+(H2(K)-XBAR1)*COS(6.2832*XK/Z1)
          SUMA2=SUMA2+(H3(K)-XBAR2)*COS(6.2832*XK/Z1)
          SUMB=SUMB+(H1(K)-XBAR)*SIN(6.2832*XK/Z1)
          SUMB1=SUMB1+(H2(K)-XBAR1)*SIN(6.2832*XK/Z1)
          SUMB2=SUMB2+(H3(K)-XBAR2)*SIN(6.2832*XK/Z1)
          XK=XK+1
15     CONTINUE
       A=SUMA*(2./Z1)
       A1=SUMA1*(2./Z1)
       A2=SUMA2*(2./Z1)
       B=SUMB*(2./Z1)
       B1=SUMB1*(2./Z1)
       B2=SUMB2*(2./Z1)
       T=ATAN(-B/A)
       T1=ATAN(-B1/A1)
       T2=ATAN(-B2/A2)
       C=A/COS(T)
       C1=A1/COS(T1)
       C2=A2/COS(T2)
       WRITE(20,20)
20     FORMAT(/,15X,' FOURIER COEFFICIENTS--MEAN')
```

```
      WRITE(20,25) XBAR,C,T
 25   FORMAT(15X,' MEAN =',F10.4,5X,' AMPLITUDE =',F10.4,5X,
     $ ' PHASE =',F10.4)
      PM(JCT)=XBAR
      JCT=JCT+1
      PM(JCT)=C
      JCT=JCT+1
      WRITE(20,30)
 30   FORMAT(/,15X,' FOURIER COEFFICIENTS--STD. DEV.')
      WRITE(20,25) XBAR1,C1,T1
      WRITE(20,35)
 35   FORMAT(/,15X,' FOURIERS COEFFICIENTS--CV')
      WRITE(20,25) XBAR2,C2,T2
      PM(JCT)=XBAR2
      JCT=JCT+1
      PM(JCT)=C2
      JCT=JCT+1
      RETURN
      END
      SUBROUTINE MSD(ALAT,I1,I2,NYRS,W,ID)
C*********************************************************************
C     THE FOLLOWING SUBROUTINE CALCULATES THE STATISTICS OF TMX, TMN,
C     TMN AND RD BY 28-DAY PERIOD OF THE YEAR AND FITS A FOURIER
C     SERIE TO THE RESULTS.
C     NYRS  = NUMBER OF AVALAILABLE YEARS IN DATA EXTERNAL DATA FILE
C     W     = DUMMY MATRIX FOR TMX,TMN AND RD
C     ID    = 1 TEST PARAMETER IF TMAX IS USED
C     ID    = 2 TEST PARAMETER IF TMIN IS USED
C     ID    = 3 TEST PARAMETER IF RAD IS USED
C*********************************************************************
      COMMON /DAT3/ RN(20,366)
      DIMENSION W(20,366),XM(52),XM1(52),SD(52), SD1(52)
      DIMENSION CX(52), CX1(52)
C********************************************************************
C     IN TEMPERATE CLIMATES THE COEFFICIENT OF VAR. FOR THE
C     MINIMUM TEMPERATURE TENDS TO BE SOMETHING OVER ZERO
C     BECAUSE THE MEAN TEMPARATURE IS ABOUT ZERO DEG. (C)
C     TO AVOID THIS THE MIN. TEMP. IS CONVERTED TO DEG. (K)
C********************************************************************
      IF (ID .EQ. 2) THEN
      DO 10 I=1,20
      DO 10 J=1,366
         W(I,J)=W(I,J)+273.
 10   CONTINUE
      END IF
      DO 20 I = 1, 52
         NF = I*7
         NI=NF-6
         XN = 0.
         XN1 = 0.
         SUM = 0.
         SUM1 = 0.
         SS = 0.
         SS1 = 0.
      DO 15 JD=NI,NF
```

```
      DO 15 JY = 1,NYRS
         IF(ID  .EQ.  2) THEN
            XN = XN + 1.
            SUM = SUM+W(JY,JD)
            SS = SS + (W(JY,JD)*W(JY,JD))
         ELSE
            IF(RN(JY,JD) .LE. 0.) THEN
               XN = XN + 1.
               SUM = SUM+W(JY,JD)
               SS = SS + (W(JY,JD)*W(JY,JD))
            ELSE
               XN1=XN1 + 1.
               SUM1 = SUM1 + W(JY,JD)
               SS1=SS1+(W(JY,JD)*W(JY,JD))
            END IF
         END IF
15    CONTINUE
      IF(XN  .LE.  2.  ) THEN
         XM(I)=0.0
         SD(I)=0.0
         CX(I)=0.0
      ELSE
         XM(I) = SUM / XN
         SD(I) = SQRT((SS-SUM*SUM/XN)/(XN-1.))
         IF(XM(I)  .LT.  0.001) XM(I) = 0.001
         CX(I) = SD(I)  / XM(I)
      ENDIF
      IF(ID  .EQ.  2) GO TO 20
      IF(XN1 .LE.  2.) THEN
         XM1(I)=0.0
         SD1(I)=0.0
         CX1(I)=0.0
      ELSE
         XM1(I) = SUM1 / XN1
         SD1(I) =SQRT((SS1-SUM1*SUM1/XN1)/(XN1-1.))
         IF(XM1(I)  .LT.  0.001) XM1(I) = 0.001
         CX1(I)=SD1(I)/XM1(I)
      ENDIF
20    CONTINUE
      IF(ID  .EQ.  2) THEN
         DO 25 I=1,52
            XM(I)=XM(I)-273.
25       CONTINUE
         WRITE(20,30)
30       FORMAT(10X,'WET AND DRY DAYS')
         CALL FOUR(ALAT,I1,I2,XM,SD,CX)
      ELSE
         WRITE(20,35)
35       FORMAT(10X,'DRY DAYS')
         CALL FOUR(ALAT,I1,I2,XM,SD,CX)
         WRITE(20,40)
40       FORMAT(/,10X,'WET DAYS')
         CALL FOUR(ALAT,I1,I2,XM1,SD1,CX1)
      END IF
      IF (ID .EQ. 2) THEN
```

```
      DO 50 I=1,20
      DO 50 J=1,366
         W(I,J)=W(I,J)-273.
 50   CONTINUE
      END IF
      RETURN
      END
      SUBROUTINE PPRN(IPAR,NYR)
C******************************************************************
C     THIS SUBROUTINE CALCULATES THE RAINFALL GENERATION AND WIND-
C     SPEED PARAMETERS USING THE MARKOV CHAIN-GAMMA MODEL
C     IPAR = 1 TEST PARAMETER IF RAINFALL DATA HAVE TO BE USED
C     IPAR = 2 TEST PARAMETER IF WINDSPEED DATA HAVE TO BE USED
C     NYR  = NUMBER OF AVALAILABLE YEARS IN DATA EXTERNAL DATA FILE
C******************************************************************
      COMMON /DAT3/ RN(20,366)
      COMMON /DAT11/ WN(20,366)
      COMMON /DAT13/ XN(20,366)
      COMMON /DAT7/ PWD(12),PWW(12),ALPHA(12),BETA(12)
      COMMON /DAT12/ ALPW(12),BETW(12),WIND(365)
      DIMENSION NWD(12),NDD(12),NDW(12), NWW(12)
      DIMENSION SUM(12),SUM2(12),SUM3(12)
      DIMENSION SL(12),RBAR(12)
      DIMENSION NW(12),IC(12),SUML(12)
      DIMENSION RLBAR(12)
      DIMENSION PPPW(12),ND(12),LB(12),LE(12)
C*******************************************************************
C     NWD = NUMBER OF WET DAYS IF PREVIOUS DAY WAS DRY
C     NDD = NUMBER OF DRY DAYS IF PREVIOUS DAY WAS DRY
C     NDW = NUMBER OF DRY DAYS IF PREVIOUS DAY WAS WET
C     NWW = NUMBER OF WET DAYS IF PREVIOUS DAY WAS WET
C     NW  = NUMBER OF WET DAYS
C     ND  = NUMBER OF DRY DAYS
C     SUM = COUNTER FOR THE AMOUNT OF RAIN
C     SUM2= QUADRATIC COUNTER FOR THE AMOUNT OF RAIN
C     SUM3= THIRD POWER COUNTER FOR THEAMOUNT OF RAIN
C     SUML= LOGARITHMIC COUNTER FOR THE AMOUNT OF RAIN
C     SL  = LOGARITHMIC COUNTER FOR THE AMOUNT OF RAIN
C     RBAR= MEAN AMOUNT OF RAIN ON A WET DAY
C     RLBAR= LOGARITHMIC MEAN OF RAIN AMOUNT ON A WET DAY
C     LB  = JULIAN DATE OF THE BEGINNING OF EACH MONTH OF NORMAL YEARS
C     LE  = JULIAN DATE OF THE END OF EACH MONTH OF NORMAL YEARS
C     IC  = STATUS COUNTER FOR CHARACTER ARRAY A(2)
C*******************************************************************
      CHARACTER *36 A(2)
      CHARACTER *5 DATE(12)
      DATA DATE/'JAN.','FEB.','MAR.','APR.','MAY.','JUN.',
     $ 'JUL.','AUG.','SEP.','OCT.','NOV.','DEC.'/
      DATA LB/1,32,60,91,121,152,182,213,244,274,305,335/
      DATA LE/31,59,90,120,151,181,212,243,273,304,334,365/
      DATA A(1) /'                                    '/
      DATA A(2) /'NOT ENOUGH DATA TO DEFINE PARAMETERS'/
C     TEST IF RAINFALL OR WINDSPEED DATA HAVE TO BE USED
      IF(IPAR.EQ.1) THEN
        DO 35 I=1,NYR
```

```
          DO 35 J=1,365
               XN(I,J)=RN(I,J)
 35       CONTINUE
        ELSEIF(IPAR.EQ.2) THEN
          DO 36 I=1,NYR
          DO 36 J=1,365
               XN(I,J)=WN(I,J)
 36       CONTINUE
        ENDIF
C       INITIALIZE
        DO 10 I =1, 12
           ND(I) =0
           PPPW(I) =0.
           NWD(I) = 0
           NWW(I) = 0
           NDD(I) =0
           NDW(I) =0
           NW(I) =0
           SL(I) = 0.
           SUML(I) = 0.
           SUM(I) =0.
           SUM2(I) = 0.
           SUM3(I) = 0.
 10     CONTINUE
C*****************************************************************
C       RIM1 = RAIN AMOUNT OF THE PREVOUS DAY
C       XRN  = RAIN AMOUNT OR WINDSPEED
C       XXND = NUMBER OF DRY DAYS
C       YYNW = NUMBER OF WET DAYS
C       XNW  = NUMBER OF WET DAYS
C       XNWW = NUMBER OF WET DAYS IF PREVIOUS DAY WAS WET
C       XNWD = NUMBER OF WET DAYS IF PREVIOUS DAY WAS WET
C       MO   = MONTH NUMBER
C*****************************************************************
        RIM1 = 0.
        DO 20 J = 1,NYR
           MO=1
        DO 30 K = 1,365
           IF(K  .GE.  LB(MO) .AND.  K  .LE.  LE(MO)) THEN
              MO=MO
           ELSE
              MO=MO+1
           ENDIF
        XRN=XN(J,K)
        IF(XRN  .GT.  0.00) THEN
           NW(MO)=NW(MO)+1
        ENDIF
        IF(IPAR.EQ.2) GOTO 11
        ND(MO)=ND(MO)+1
        IF(XRN) 5,5,3
 3      IF(RIM1.LE. 0.) THEN
           NWD(MO)=NWD(MO)+1
        ELSE
           NWW(MO)=NWW(MO)+1
        END IF
```

```
 11      SUML(MO)=SUML(MO)+ALOG(XRN)
         SUM(MO)=SUM(MO)+XRN
         SUM2(MO)=SUM2(MO) + XRN * XRN
         SUM3(MO)=SUM3(MO) +XRN*XRN*XRN
         SL(MO) = SL(MO)+ALOG(XRN)
         GO TO 9
 5       IF(RIM1 .LE. 0.) THEN
           NDD(MO)=NDD(MO)+1
         ELSE
           NDW(MO)=NDW(MO)+1
         END IF
 9       RIM1 = XRN
 30      CONTINUE
 20      CONTINUE
         DO 120 I = 1, 12
         IF(IPAR.EQ.2) GOTO 12
            XXND=ND(I)
            YYNW=NW(I)
            PPPW(I) = YYNW/XXND
            III=1
C        TEST IF ENOUGH DATA AVAILABLE
         IF(NW(I)  .LT.  3) III=2
            IC(I) = III
         IF(NW(I)  .LT.  3) GOTO 120
C        CALCULATION OF PWW AND PWD
            XNWW=NWW(I)
            XNWD=NWD(I)
            XXNW=NWW(I)+NDW(I)
            XND=NDD(I)+NWD(I)
            PWW(I)=XNWW/XXNW
            PWD(I)=XNWD/XND
C        CALCULATION O ALPHA AND BETA
 12         XNW=NW(I)
            RBAR(I)=SUM(I)/XNW
            RLBAR(I)=SUML(I)/XNW
            Y=ALOG(RBAR(I))-RLBAR(I)
            ANUM=8.898919+9.05995*Y+0.9775373*Y*Y
            ADOM=Y*(17.79728+11.968477*Y+Y*Y)
            ALPHA2=ANUM/ADOM
         IF(IPAR.EQ.1) THEN
            IF(ALPHA2  .GE.  1.0) THEN
              ALPHA2=0.998
            ENDIF
            BETA2=RBAR(I)/ALPHA2
            ALPHA(I)=ALPHA2
            BETA(I)=BETA2
         ELSEIF(IPAR.EQ.2) THEN
            BETA2=RBAR(I)/ALPHA2
            ALPW(I)=ALPHA2
            BETW(I)=BETA2
         ENDIF
 120     CONTINUE
         IF(IPAR.EQ.1) THEN
         WRITE(20,201)
 201     FORMAT(///,8X,'--MARKOV CHAIN--',16X,'-GAMMA DIST-',/,
```

```
     $  1X,' MONTH  P(W/W)     P(W/D)',11X,' ALPHA    BETA',/)
        DO 130 I = 1, 12
        WRITE(20,202) DATE(I),PWW(I),PWD(I),ALPHA(I),BETA(I),A(IC(I))
 202    FORMAT(1X,A4,F8.3,F10.3,11X,F11.3,F7.3,5X,A36)
 130    CONTINUE
        ELSEIF(IPAR.EQ.2) THEN
        WRITE(20,243)
 243    FORMAT(///,8X,'-GAMMA DIST-',/,
     $  1X,' MONTH  ',11X,' ALPWI    BETWI',/)
        DO 131 I = 1, 12
        WRITE(20,252) DATE(I),ALPW(I),BETW(I),A(IC(I))
 252    FORMAT(1X,A4,11X,F11.3,F7.3,5X,A36)
 131    CONTINUE
        ENDIF
        RETURN
        END

        SUBROUTINE INPUT(I,MT)
C*****************************************************************
C       THIS SUBROUTINE READS THE DATA OF THE INPUTFILE
C       I=YEAR NUMBER
C       MT = 1 TEST PARAMETER IF END OF FILE IS REACHED
C       MT = 0 TEST PARAMETER IF END OF FILE IS NOT YET REACHED
C*****************************************************************
        IMPLICIT REAL (A-Z)
        INTEGER I,J
        COMMON /DAT1/ TMX(20,366)
        COMMON /DAT2/ TMN(20,366)
        COMMON /DAT3/ RN(20,366)
        COMMON /DAT4/ RD(20,366)
        COMMON /DAT5/ RTO(20,366)
        COMMON /DAT11/ WN(20,366)
        COMMON /DAT14/ HUM(20,366)
        READ (11,900,END=999) (TMX(I,J),J=1,365)
        READ (11,901,END=999) (TMN(I,J),J=1,365)
        READ (11,901,END=999) (RN(I,J),J=1,365)
        READ (11,905,END=999) (RD(I,J),J=1,365)
        READ (11,901,END=999) (WN(I,J),J=1,365)
 900    FORMAT(/////,37(9X,10F6.1,/))
 901    FORMAT(//,37(9X,10F6.1,/))
 905    FORMAT(//,37(9X,10F6.3,/))
        DO 902 J=1,365
           RTO(I,J)=0.0
           HUM(I,J)=6.11*EXP(17.47*TMN(I,J)/(TMN(I,J)+239.))*0.1
 902    CONTINUE
        GOTO 888
 999    MT=1
 888    RETURN
        END
        SUBROUTINE XSUM(NYRS,LSTART,LTEST,MDAY,MMO,MYR,MH,MM,MS,XNAM)
C*******************************************************************
C       THIS SUBROUTINE CALCULATES THE AVERAGE,MEAN,STANDARD DEVIATION,
C       STANDARD ERROR AND THE COEFFICIENTS OF VARIATION OF THE ACTUAL
C       AND OF THE SIMULATED MAX. TEMP., MIN. TEMP., SOLAR RAD., RAIN
C       AND  WINDSPEED DATA
```

```
C
C       NYRS   = NUMBER OF YEARS OF ACTUAL OR SIMULATED DATA WHICH HAVE
C                TO BE SUMMARIZED
C       LSTART= STARTING YEAR USED FOR THE PRINTOUT OF THE SUMMARIZED
C                ACTUAL DATA
C       LTEST = 0 TEST PARAMETER IF ACTUAL DATA HAVE TO BE USED
C       LTEST = 1 TEST PARAMETER IF SIMULARED DATA HVE TO BE USED
C       MDAY,MMO,MYR,MH,MM,MS   = DAY,MONTH,YEAR,HOUR,MINUTE AND SECOND
C       XNAM   = NAME OF THE EXTERNAL INPUT DATAFILE
C*********************************************************************
        COMMON /DAT1/ TMX(20,366)
        COMMON /DAT2/ TMN(20,366)
        COMMON /DAT3/ RN(20,366)
        COMMON /DAT4/ RD(20,366)
        COMMON /DAT5/ RTO(20,366)
        COMMON /DAT11/ WN(20,366)
        COMMON /DAT14/HUM(20,366)
        INTEGER*2 MH,MM,MYR,MMO,MDAY,MS,MHS
        CHARACTER*30 TITLE
        CHARACTER*17 TITL2
        CHARACTER*15 INPNAM,XNAM
        CHARACTER*5 DATE(12)
        DIMENSION CRRT(12)
        DIMENSION ATRT(12),SDRT(20,12),SRRT(12)
        DIMENSION LANW(12),ATRAIN(12),ATTMIN(12),ATTMAX(12),ATRAD(12)
        DIMENSION LBMON(12),LEMON(12),SDRN(20,12),SDTX(20,12)
        DIMENSION NW(12),SDR(20,12),SDWET(20,12),SDTN(20,12)
        DIMENSION SRQWT(12),SRQRN(12),SRDTN(12),SRDTX(12),SRRD(12)
        DIMENSION CRQWT(12),CRQRN(12),CRDTN(12),CRDTX(12),CRRD(12)
        DIMENSION QWT(12),QDRN(12),QDTN(12),QDTX(12),QRD(12),QRTO(12)
        DIMENSION ATWIN(12),SDWIN(20,12),QWND(12),SRWIN(12),CRWIN(12)
        DIMENSION ATHUM(12),SDHUM(20,12),SRHUM(12),CRHUM(12),QHM(12)
        DATA DATE/'JAN.','FEB.','MAR.','APR.','MAY.','JUN.',
     $  'JUL.','AUG.','SEP.','OCT.','NOV.','DEC.'/
        DATA LBMON/1,32,60,91,121,152,182,213,244,274,305,335/
        DATA LEMON/31,59,90,120,151,181,212,243,273,304,334,365/
        IF(LTEST .EQ. 0) THEN
           INPNAM='SUMMAR.DAT'
           TITLE=' SUMMARY OF THE ACTUAL DATA'
           TITL2=' BASED ON FILE: '
        ELSEIF(LTEST. EQ. 1) THEN
           INPNAM='SUMWEATH.DAT'
           TITLE=' SUMMARY OF THE SIMULATED DATA'
           TITL2=' CALCULATED WITH '
        ENDIF
        OPEN(11,STATUS='NEW',FILE=INPNAM)
C       INITIALIZE COUNTERS
        DO 871 L=1,12
           ATRAIN(L)=0.
           ATTMIN(L)=0.
           ATTMAX(L)=0.
           ATRAD(L)=0.
           ATRT(L)=0.
           ATWIN(L)=0.
           LANW(L)=0
```

```
         ATHUM(L)=0.
 871    CONTINUE
C       INITIALIZE COUNTERS
        DO 872 I=1,NYRS
        DO 872 L=1,12
           SDRN(I,L)=0.
           SDTX(I,L)=0.
           SDTN(I,L)=0.
           SDR(I,L)=0.
           SDWET(I,L)=0.
           SDRT(I,L)=0.
           SDWIN(I,L)=0.
           SDHUM(I,L)=0.
 872    CONTINUE
C       INITIALIZE COUNTERS
           STRAIN=0.
           STMIN=0.
           STMAX=0.
           STRAD=0.
           STWIN=0.
           LWET=0
           STRT=0.
           STHUM=0.
        WRITE(11,882) TITLE
 882    FORMAT(A30)
        WRITE(11,10) MDAY,MMO,MYR
        WRITE(11,22) MH,MM,MS
        WRITE(11,33) TITL2,XNAM
 10     FORMAT(/,' CREATED ON ',I2,'-',I2,'-',I4)
 22     FORMAT(' TIME         ',I2,':',I2,':',I2)
 33     FORMAT(A17,A15)
        DO 20 I = 1,NYRS
C          INITIALIZE COUNTERS
           PYTMAX=0.
           PYTMIN=0.
           PYRAD=0.
           NYWET=0
           RYR=0.
           PYRT=0.
           PYWIN=0.
           PYHUM=0.
           KYEAR=LSTART+I-1
           WRITE(11,201) KYEAR
 201    FORMAT(//,5X,'SUMMARY FOR YEAR',I5,/)
           WRITE(11,202)
 202    FORMAT(' MONTH',2X,'WET DAYS',2X,'RAINFAL',2X,'MAX TEMP',2X
     $  'MIN TEMP',2X,'SOLAR RAD',4X,'RATIO',7X,'WIND',5X,'HUMIDITY',/
     $  ,18X,' (MM)',5X,'(C)',7X,'(C)',6X,'(MJ/M2/DAY)',13X,'(M/S)',5X,
     $  '(KPa)')
        DO 25 K=1,12
C          INITIALIZE COUNTERS
           NW(K)=0
           QRAIN=0.
           QTMIN=0.
           QTMAX=0.
```

```
            QRAD=0.
            QRT=0.
            QWIN=0.
            QHUM=0.
         DO 30 J = LBMON(K),LEMON(K)
C        COUNTING NUMBER OF WET DAYS
         IF(RN(I,J).GT.0.0) THEN
            NW(K)=NW(K)+1
            LANW(K)=LANW(K)+1
            LWET=LWET+1
         END IF
C           MONTHLY TOTAL
            QRAIN=QRAIN+RN(I,J)
            QTMIN=QTMIN+TMN(I,J)
            QTMAX=QTMAX+TMX(I,J)
            QRAD=QRAD+RD(I,J)
            QRT=QRT+RTO(I,J)
            QWIN=QWIN+WN(I,J)
            QHUM=QHUM+HUM(I,J)
C           NYRS TOTAL FOR EVERY MONTH
            ATRAIN(K)= ATRAIN(K)+RN(I,J)
            ATTMIN(K)=ATTMIN(K)+TMN(I,J)
            ATTMAX(K)=ATTMAX(K)+ TMX(I,J)
            ATRAD(K)=ATRAD(K)+RD(I,J)
            ATRT(K)=ATRT(K)+RTO(I,J)
            ATWIN(K)=ATWIN(K)+WN(I,J)
            ATHUM(K)=ATHUM(K)+HUM(I,J)
 30      CONTINUE
C           TOTAL NUMBER OF WET DAYS IN A YEAR
            NYWET=NYWET+NW(K)
C           TOTAL AMOUNT OF RAIN IN A YEAR
            RYR=RYR+QRAIN
C           MONTHLY AVERAGE
            XYZ=1/(FLOAT(LEMON(K)-LBMON(K))+1)
            AVTMIN=QTMIN*XYZ
            AVTMAX=QTMAX*XYZ
            AVRAD=QRAD*XYZ
            AVRT=QRT*XYZ
            AVWIN=QWIN*XYZ
            AVHUM=QHUM*XYZ
C           AVRAD=AVRAD*4.184*0.01
C           MONTHLY TOTAL
            SDRN(I,K)=QRAIN
            SDTX(I,K)=AVTMAX
            SDTN(I,K)=AVTMIN
            SDR(I,K)=AVRAD
            SDWET(I,K)=NW(K)
            SDRT(I,K)=AVRT
            SDWIN(I,K)=AVWIN
            SDHUM(I,K)=AVHUM
         WRITE(11,203) DATE(K),NW(K),QRAIN,AVTMAX,AVTMIN,AVRAD,
     $   AVRT,AVWIN,AVHUM
 203     FORMAT(A5,3X,I4,5X,F6.1,3X,F5.1,5X,F5.1,6X,F6.3,5X,F6.2,5X,F6.1,
     $   5X,F6.2)
C           YEARLY TOTAL
```

```
            PYTMAX=PYTMAX+AVTMAX
            PYTMIN=PYTMIN+AVTMIN
            PYRAD=PYRAD+AVRAD
            PYRT=PYRT+AVRT
            PYWIN=PYWIN+AVWIN
            PYHUM=PYHUM+AVHUM
 25      CONTINUE
C           YEARLY MEAN
            PYTMAX=PYTMAX/12.
            PYTMIN=PYTMIN/12.
            PYRAD=PYRAD/12.
            PYRT=PYRT/12.
            PYWIN=PYWIN/12.
            PYHUM=PYHUM/12

         WRITE(11,205) NYWET,RYR
 205     FORMAT(/' TOTAL',2X,I3,5X,F7.1,6X,'_',8X,'_',11X,'_',
     $    8X,'_',8X,'_',9X,'_')
         WRITE(11,350) PYTMAX,PYTMIN,PYRAD,PYRT,PYWIN,PYHUM
 350     FORMAT(' AVER.',3X,'_',10X,'_',3X,F5.1,5X,F5.1,5X,F7.3,
     $   5X,F6.2,5X,F6.1,5X,F6.1,5X,F6.2)
 20      CONTINUE
         WRITE(11,861) NYRS
 861     FORMAT(//,5X,'OVERALL SUMMARY FOR',I3,' YEARS',/,5X,
     $   '(AVERAGED OVER THE YEARS)')
         WRITE(11,862)
 862     FORMAT(' MONTH',2X,'WET DAYS',2X,'RAINFAL',2X,'MAX TEMP',2X
     $   'MIN TEMP',2X,'SOLAR RAD',4X,'RATIO',7X,'WIND',5X,'HUMIDITY',/
     $   ,18X,' (MM)',5X,'(C)',7X,'(C)',6X,'(MJ/M2/DAY)',13X,'(M/S)',
     $   5X,'(KPa)')
C        TOTAL OVER NYRS YEARS
         DO 863 K=1,12
            STRAIN=STRAIN+ATRAIN(K)
            STMIN=STMIN+ATTMIN(K)
            STMAX=STMAX+ATTMAX(K)
            STRAD=STRAD+ATRAD(K)
            STRT=STRT+ATRT(K)
            STWIN=STWIN+ATWIN(K)
            STHUM=STHUM+ATHUM(K)
C           MONTHLY AVERAGE OVER NYRS YEARS
            XYZ= 1/(FLOAT(NYRS)*(FLOAT(LEMON(K)-LBMON(K))+1))
            ATTMIN(K)=ATTMIN(K)*XYZ
            ATTMAX(K)=ATTMAX(K)*XYZ
            ATRAD(K)=ATRAD(K)*XYZ
            ATRT(K)=ATRT(K)*XYZ
            ATWIN(K)=ATWIN(K)*XYZ
            ATHUM(K)=ATHUM(K)*XYZ
C           AVERAGE NUMBER OF WET DAYS IN A MONTH OVER NYRS YEARS
            LANW(K)=LANW(K)/FLOAT(NYRS)
C           AVERAGE AMOUNT OF RAIN ON A MONTH OVER NYRS YEARS
            ATRAIN(K)=ATRAIN(K)/FLOAT(NYRS)
         WRITE(11,864) DATE(K),LANW(K),ATRAIN(K),ATTMAX(K),
     $   ATTMIN(K),ATRAD(K),ATRT(K),ATWIN(K),ATHUM(K)
 864     FORMAT(A5,3X,I4,6X,F5.1,3X,F5.1,5X,F5.1,6X,F6.2,5X,F6.2,5X,F6.1,)
     $   5X,F6.2)
```

```
 863    CONTINUE
C          DAILY AVERAGE OVER NYRS YEARS
           STMIN=STMIN/(365*FLOAT(NYRS))
           STMAX=STMAX/(365*FLOAT(NYRS))
           STRAD=STRAD/(365*FLOAT(NYRS))
           STRT=STRT/(365*FLOAT(NYRS))
           STWIN=STWIN/(365*FLOAT(NYRS))
           STHUM=STHUM/(365*FLOAT(NYRS))
C          AVERAGE NUMBER OF WET DAYS OVER NYRS YEARS
           LWET=LWET/NYRS
C          AVERAGE YEARLY TOTAL RAIN
           STRAIN=STRAIN/NYRS

        WRITE(11,865) LWET,STRAIN
 865    FORMAT(/,' TOTAL',1X,I5,4X,F7.1,6X,'_',8X,'_',11X,'_',
     $  8X,'_',8X,'_',9X,'_')
        WRITE(11,866) STMAX,STMIN,STRAD,STRT,STWIN,STHUM
 866    FORMAT(' AVER.',3X,'_',10X,'_',3X,F5.1,5X,F5.1,6X,F6.1,
     $  5X,F6.2,5X,F6.1,5X,F6.1)

        WRITE(11,867)
 867    FORMAT(///,' SUMMARY OF STANDARD DEVIATIONS'//)
        WRITE(11,876)
 876    FORMAT(' MONTH',2X,'WET DAYS',2X,'RAINFAL',2X,'MAX TEMP',2X
     $  'MIN TEMP',2X,'SOLAR RAD',5X,'RATIO',8X,'WSPEED',5X,'HUMIDITY')
        DO 873 K=1,12
C          INITIALIZE COUNTERS
           DRN=0.
           DTX=0.
           DTN=0.
           DWET=0.
           DRAD=0.
           DRT=0.
           DWIN=0.
           DHUM=0.
           SQRN=0.
           SQDTX=0.
           SQDTN=0.
           SQRAD=0.
           SQWET=0.
           SXRTO=0.
           SQWIN=0.
           SQHUM=0.
         DO 874 I=1,NYRS
           DRN=DRN+SDRN(I,K)
           DTX=DTX+SDTX(I,K)
           DTN=DTN+SDTN(I,K)
           DWET=DWET+SDWET(I,K)
           DRAD=DRAD+SDR(I,K)
           DRT=DRT+SDRT(I,K)
           DWIN=DWIN+SDWIN(I,K)
           DHUM=DHUM+SDHUM(I,K)
 874    CONTINUE
           QYZ=1./FLOAT(NYRS)
           DRNM=DRN*QYZ
```

```
         DTXM=DTX*QYZ
         DTNM=DTN*QYZ
         DWETM=DWET*QYZ
         DRADM=DRAD*QYZ
         DRTM=DRT*QYZ
         DWINM=DWIN*QYZ
         DHUMM=DHUM*QYZ

      DO 889 I=1,NYRS
         SQRN=SQRN+(SDRN(I,K)-DRNM)*(SDRN(I,K)-DRNM)
         SQDTX=SQDTX+(SDTX(I,K)-DTXM)*(SDTX(I,K)-DTXM)
         SQDTN=SQDTN+(SDTN(I,K)-DTNM)*(SDTN(I,K)-DTNM)
         SQRAD=SQRAD+(SDR(I,K)-DRADM)*(SDR(I,K)-DRADM)
         SQWET=SQWET+(SDWET(I,K)-DWETM)*(SDWET(I,K)-DWETM)
         SXRTO=SXRTO+(SDRT(I,K)-DRTM)*(SDRT(I,K)-DRTM)
         SQWIN=SQWIN+(SDWIN(I,K)-DWINM)*(SDWIN(I,K)-DWINM)
         SQHUM=SQHUM+(SDHUM(I,K)-DHUMM)*(SDHUM(I,K)-DHUMM)
 889  CONTINUE

C        STANDARD DEVIATION
         QDRN(K)=SQRT(SQRN)/FLOAT(NYRS-1)
         QDTX(K)=SQRT(SQDTX)/FLOAT(NYRS-1)
         QDTN(K)=SQRT(SQDTN)/FLOAT(NYRS-1)
         QRD(K)=SQRT(SQRAD)/FLOAT(NYRS-1)
         QWT(K)=SQRT(SQWET)/FLOAT(NYRS-1)
         QRTO(K)=SQRT(SXRTO)/FLOAT(NYRS-1)
         QWND(K)=SQRT(SQWIN)/FLOAT(NYRS-1)
         QHM(K)=SQRT(SQHUM)/FLOAT(NYRS-1)
C        STANDARD ERRORS
         SRQRN(K)=QDRN(K)/SQRT(FLOAT(NYRS))
         SRDTX(K)=QDTX(K)/SQRT(FLOAT(NYRS))
         SRDTN(K)=QDTN(K)/SQRT(FLOAT(NYRS))
         SRRD(K)=QRD(K)/SQRT(FLOAT(NYRS))
         SRQWT(K)=QWT(K)/SQRT(FLOAT(NYRS))
         SRRT(K)=QRTO(K)/SQRT(FLOAT(NYRS))
         SRWIN(K)=QWND(K)/SQRT(FLOAT(NYRS))
         SRHUM(K)=QHM(K)/SQRT(FLOAT(NYRS))
C        COEFFICIENTS OF VARIATION
         CRQRN(K)=QDRN(K)/ATRAIN(K)
         CRDTX(K)=QDTX(K)/ATTMAX(K)
         CRDTN(K)=QDTN(K)/ATTMIN(K)
         CRRD(K)=QRD(K)/ATRAD(K)
         CRQWT(K)=QWT(K)/LANW(K)
         CRRT(K)=QRTO(K)/ATRT(K)
         CRWIN(K)=QWND(K)/ATWIN(K)
         CRHUM(K)=QHM(K)/ATHUM(K)

      WRITE(11,877) DATE(K),QWT(K),QDRN(K),QDTX(K),QDTN(K),QRD(K),
     $ QRTO(K),QWND(K),QHM(K)
 877  FORMAT(A5,1X,F6.2,4X,F8.2,3X,F5.2,5X,F5.2,6X,F8.3,5X,F6.2,5X,
     $ F6.2,5X,F6.2)
 873  CONTINUE
      WRITE(11,879)
 879  FORMAT(///,' SUMMARY OF STANDARD ERRORS'//)
      WRITE(11,876)
```

```
      DO 878 K=1,12
      WRITE(11,877) DATE(K),SRQWT(K),SRQRN(K),SRDTX(K)
     $ ,SRDTN(K),SRRD(K),SRRT(K),SRWIN(K),SRHUM(K)
878   CONTINUE

      WRITE(11,880)
880   FORMAT(///,' SUMMARY OF COEFFICIENTS OF VARIATION'//)
      WRITE(11,876)
      DO 881 K=1,12
      WRITE(11,877) DATE(K),CRQWT(K),CRQRN(K),CRDTX(K)
     $ ,CRDTN(K),CRRD(K),CRRT(K),CRWIN(K),CRHUM(K)
881   CONTINUE
      CLOSE(11)
      RETURN
      END
```

Appendix B

There are a few differences between the VAX version and the IBM version of the program SIMWTH. Both versions are written in Fortran 77.

The VAX version has to be started by typing RUN SIMWTH. The IBM version starts when only SIMWTH is typed. Further the procedures are the same. Since the timer in both computers is different, the procedure to create seed number for the random generator is different. In the following part the differences in subroutine RANDN are given. K(I) is the array with fixed seed numbers.

VAX version

```
      CHARACTER*8 HMS
      CALL TIMER(HMS)
      READ(HMS,5) ISEC
  5   FORMAT(6X,I2)
      DO I=1,4
         K(I)=K(I)+ISEC
      END DO
```

IBM version

```
      INTEGER*2 MH,MM,MS,MHS
      CALL GETTIM(MH,MM,MS,MHS)

      DO 10 1=1,4
         K(I)=K(I)+MM*MS-MH+MHS
 10   CONTINUE
```