Technical Report No. 298

STRUCTURE OF AN INSECT MODEL

Erik Framstad*
Natural Resource Ecology Laboratory
Colorado State University
Fort Collins, Colorado 80523

GRASSLAND BIOME
U.S. International Biological Program
February 1976

---

[*] Visiting scientist from Zoologisk Institutt, University of Oslo, Blindern, Oslo, 3, Norway.

TABLE OF CONTENTS

ABSTRACT

This is a description of the structure of an insect model. The model simulates the population and energy dynamics of seven major groups of insects. The processes handled in the model are predation, non-predatory mortality, recruitment of individuals, food selection and feeding, excretion, and respiration. In addition to the state variable giving biomass (in g C/m^2), density (in numbers/m^2) and average weight (in g C/individual) are given for each category of insects. No simulation results are given.

INTRODUCTION

Purpose

The below- and aboveground macroarthropods are among the more important groups of consumers in the grassland ecosystem. The evaluation of their impact on the producers should therefore be of interest in any total ecosystem study of the grasslands.

A total ecosystem model for the grasslands, ELM, has been developed at the Natural Resource Ecology Laboratory (Innis 1975*a*, Innis and Gustafson 1975; Parton 1975; Sauer 1975; Anway 1975; Rodell 1975; Hunt 1975; Reuss and Innis 1975; Cole, Innis, and Stewart 1975; Steinhorst, Hunt, and Innis 1975; Innis 1975*b*; Woodmansee 1975). However, at present macroarthropods are only represented by grasshoppers. The model described here is an attempt to go a step further. It is a simulation model for several major groups of insects and spiders and takes care of both the number and energy dynamics of each group. Eventually, this model will be run either as an integrated submodel in ELM or in conjunction with ELM.

Basic Approach

By reviewing the biomass data for macroarthropods it seemed that there were ten important groups to take into consideration: Araneida, Carabidae, Chrysomelidae-Curculionidae, Scarabaeidae, Tenebrionidae, aboveground Homoptera-Hemiptera, Margarodidae, Asilidae, Formicidae, and Orthoptera. (Margarodidae, Asilidae, and Formicidae have yet to be introduced into the model.)

Each of these groups is split up into 16 categories or life stages (eggs, larvae, diapausing individuals, adults, etc.). It is assumed that to represent each group by one particular life history is adequate and realistic within the framework of the model (See Appendix A). For each of these groups and categories several processes are handled. Individuals die from predation and from non-predatory causes, and they are transferred from one life stage to the next. They respire, feed, excrete, and produce litter through their feeding activity. In addition, average individual weights are calculated (in g C) and compared to an expected or "normal" weight for that particular life stage. This is to arrive at effects of starvation on survival and egg laying rate. In addition, a few abiotic variables are computed from abiotic data supplied by ELM.

The driving variables are supplied by the abiotic, producer, and mammalian consumer submodels of ELM (for list of variables see Appendix B).

Current Status

At this time only seven of the aforementioned ten groups have been included in the model.

Furthermore, little attempt has been made to "tune" the model, and most of the parameters have little or no basis in actual data. Parameter values were largely subjectively chosen to allow a successful execution of the model, that is, to utilize all the different pathways in the model. Since time was limiting, it was seen as more important to develop the structure of the model than to obtain a good literature base for the parameter values.

MODEL STRUCTURE

General Organization

The model is programmed in the simulation language SIMCOMP 3.0 (Gustafson and Innis 1973). Although SIMCOMP has its own basic structure, virtually all the important calculations take place in SIMCOMP's subroutine CYCL1. The program can therefore be read much as an ordinary FORTRAN program. SIMCOMP's source program is only used to create the appropriate variables for interconnections with ELM.

The overall organization of the model is as follows. First comes a block of coding which is computed once at each time step. This includes computation of a few abiotic variables, relative density, preference and rank for the food selection mechanism for herbivores, and the number of degree-days per day that will be experienced at the surface and at 7.5 cm soil depth.

Then follows a block of coding which is computed once every time step for each stage for each group of insects. Here the expected or "normal" weight and the actual average weight of an individual are computed. This is followed by a block that pertains to the energy needs of predators, the preference of predators for different prey, and, finally, the actual number of prey taken.

The rest of CYCL1 is made up of a block of coding for each group of insects which has been included in the model. In each of these blocks the following are computed for each stage: The number dying from non-predatory causes, the number transferred from one stage to the next, the total respiration, feeding, and excretion by all the individuals in the stage.

In SIMCOMP's subroutine START the values at the beginning of the simulation are set for some variables. In addition to this a number of functions and subroutines are supplied for use in different calculations in CYCL1.

The life cycles which were chosen as representative for the different groups were subjectively determined from field data and information in the literature. The length of a particular stage can be changed by changing parameter values (ETIME(I,J), ETIM1(I,J), ETIM2(I,J)). However, the particular sequence of stages, the length of the total life cycle, etc., can only be changed by changing the structure of the model.

The following life stages are recognized in the model: Eggs, active immatures (larvae, nymphs, or spiderlings), pupae, mature adults, and diapausing immatures and adults. Although there is a difference between active and diapausing eggs or pupae, this has not been considered in the model for other processes than respiration.

For subscripted variables that do not pertain to food selection, the first index refers to the life stage and the second to the particular group of insects. A description of the different life cycles can be found in Appendix A.

Processes

Below follows a description of the different processes and their implementation in the model. Where the implementation differs somewhat for the different groups of insects, these differences will be treated briefly at the end of each paragraph discussing a particular process. Definition of variables can be found in Appendix C. The flows affecting a particular life stage are illustrated in Fig. 1.

Computation of Degree-Days

The number of degree-days per day is computed in CYCL1 by calling up the function EDGDS with input parameters of maximum and minimum

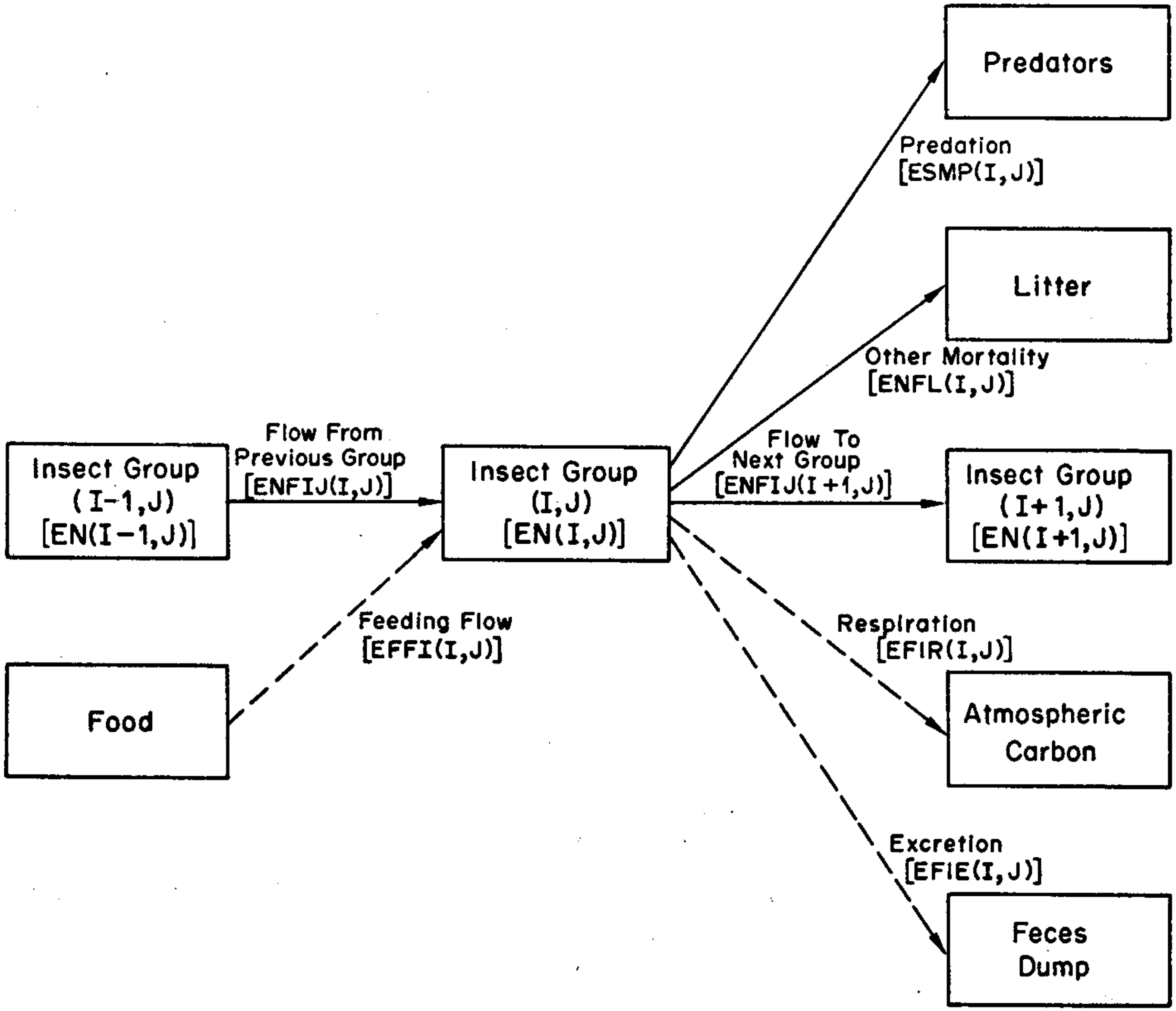


Fig. 1. Flows affecting a category (I,J) of insects. Flows of individuals occur along the solid lines and flows of carbon along both the solid and the stipled lines. (The symbols in brackets are the ones used in the coding; however, where flows of individuals occur, only the symbols pertaining to these flows are shown.)

temperature for the day and the developmental zero. The developmental zero is probably different from species to species and may be from life stage to life stage within one species. However, because of lack of information in the literature, no attempt has been made to distinguish between developmental zero values for the different groups. Instead, the number of degree-days is considered to be the same for all animals at the same level in the soil. Two values for degree-days are determined, one for animals at the surface and one for animals at a soil depth of approximately 7.5 cm.

The number of degree-days is used in the computation of several different variables. It is used in the function that computes the number of eggs laid per female (EEPF) and in the function for feeding of herbivores (EFOD). In addition the accumulated sum of degree-days is used in some of the transfers from one stage to another.

The function EDGDS simulates the temperature through the 24 hours by assuming that the maximum temperature occurs 8 hours after the minimum temperature, which occurs at zero hours, and that the temperature then drops off to the same minimum in the next 16 hours. The temperature between these points is described by part of a sine wave (Fig. 2). Note that it is unimportant for this purpose whether the minimum temperature actually occurs at midnight. The x-values for the intersection between the developmental zero and the temperature function (EX1 and EX2) is then determined by calling the function EASN. The appropriate number of degree-days is found by computing the area between the developmental zero and the temperature function.

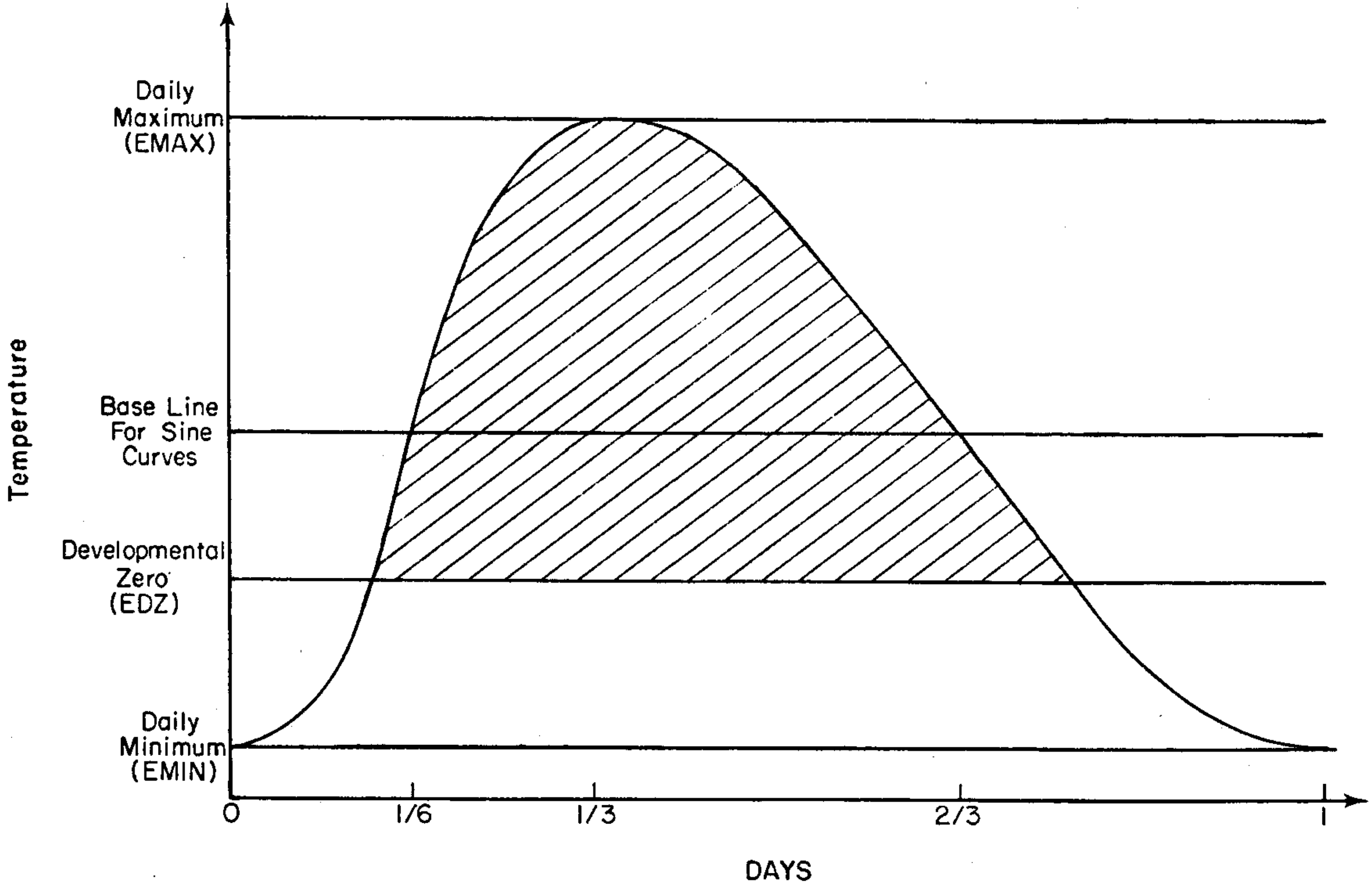


Fig. 2. The temperature curve over 24 hours as a function of time of day and daily maximum and minimum temperatures, simulated by two sine curves. Number of degree-days per day is equal to the barred area and is found by integration betweeen the developmental zero line and the temperature curve. (The symbols in parentheses are the variable names used in the coding.)

Food Selection

The mechanism for food selection for herbivores is based upon a conceptual model described by Ellis et al. (1975). The relative densities and ranks of different foods are established in the beginning of CYCL1. Determination of consumer food demand, selectivity, and actual amount eaten occurs in function EFOD which is called in the feeding part in the block of coding that is specific for each group.

Twenty different kinds of food, as given by the producer submodel in ELM, were considered. The relative densities of these foods were computed by normalizing the actual densities (in g C/m^2) between 0 and 1.

In order to save computer space and time a different set of indices is used for the herbivore consumers than for the insect categories in general (Appendix D).

For every consumer a preference index was subjectively assigned to each food. This had the value 10.0 for maximum preference and the value 0.0 for minimum preference. From this a relative preference index was created by dividing the assigned index by the index for the appropriate phenological stage as supplied by the producer submodel, and normalizing the values between 0 and 1. Finally a ranking of the different foods was achieved by multiplying the relative density and the relative preference index and normalizing the result between 0 and 1.

In the function EFOD the selectivity of a consumer is considered a function of its satiation. The satiation is computed as the ratio between the actual average weight and the expected weight for an average individual in a category of consumers. If this ratio is less than 0.8, the consumer is considered hungry and will eat according to availability rather than preference. If the ratio is greater than 1.0, the consumer is satiated and will eat completely according to preference. In between the selectivity follows an S-shaped curve.

Consumer food demand is a function of the difference between expected and actual weight, metabolic demand, and energy demand for reproduction. The consumer will tend to satisfy this demand by eating from each food category according to the ranking. However, this will be limited by the amount of food available and the time, measured as degree-days per day, available for feeding activity (see Fig. 3 and 4).

The food consumption by predators is handled rather differently. The predacious insect categories have been lumped into groups of predators (for a list of groups see Appendix D). For example, predator group 3 is Carabidae larvae and consists of EN(5,1) and EN(7,1). The total amount of food eaten by one predator group is computed by multiplying the number of prey taken by this group from an insect category by the prey's weight and summing over all insect categories. Food taken is then distributed among the insect groups in the particular predator group according to their weight. The computation of prey taken by each predator group is described in the part covering predation.

Excretion and Litter Production

Excretion is simply the amount of food ingested minus the amount assimilated by the insect. The amount assimilated is determined as the fraction EASK(I) of total food eaten.

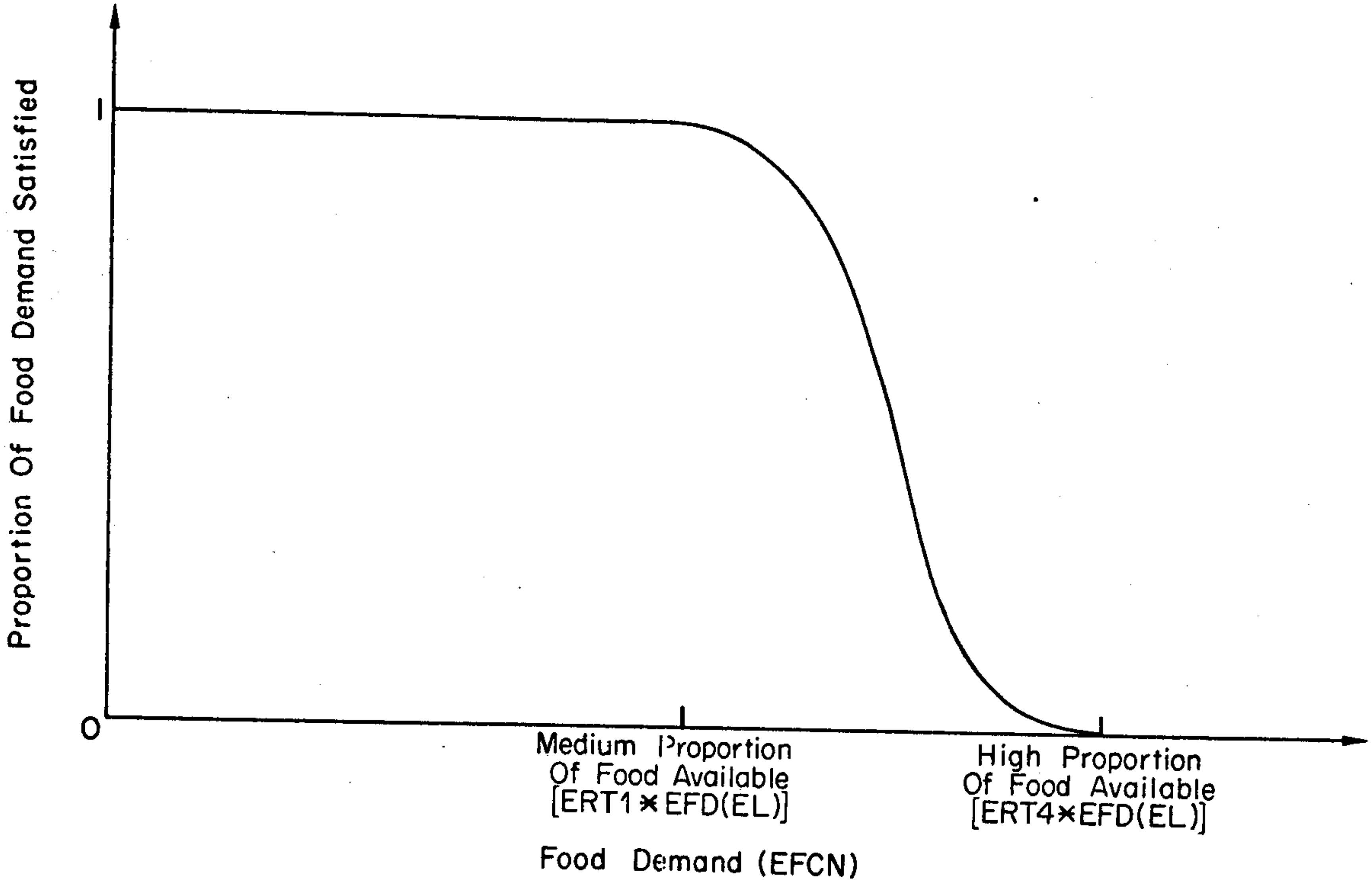


Fig. 3. Limitations on food consumption by amount of food available. When food demand exceeds a certain proportion of food available (ERT1), the actual proportion satisfied of the food demand is given by a sine function. When food demand exceeds the proportion ERT4 of food available, the actual food consumption will be zero.

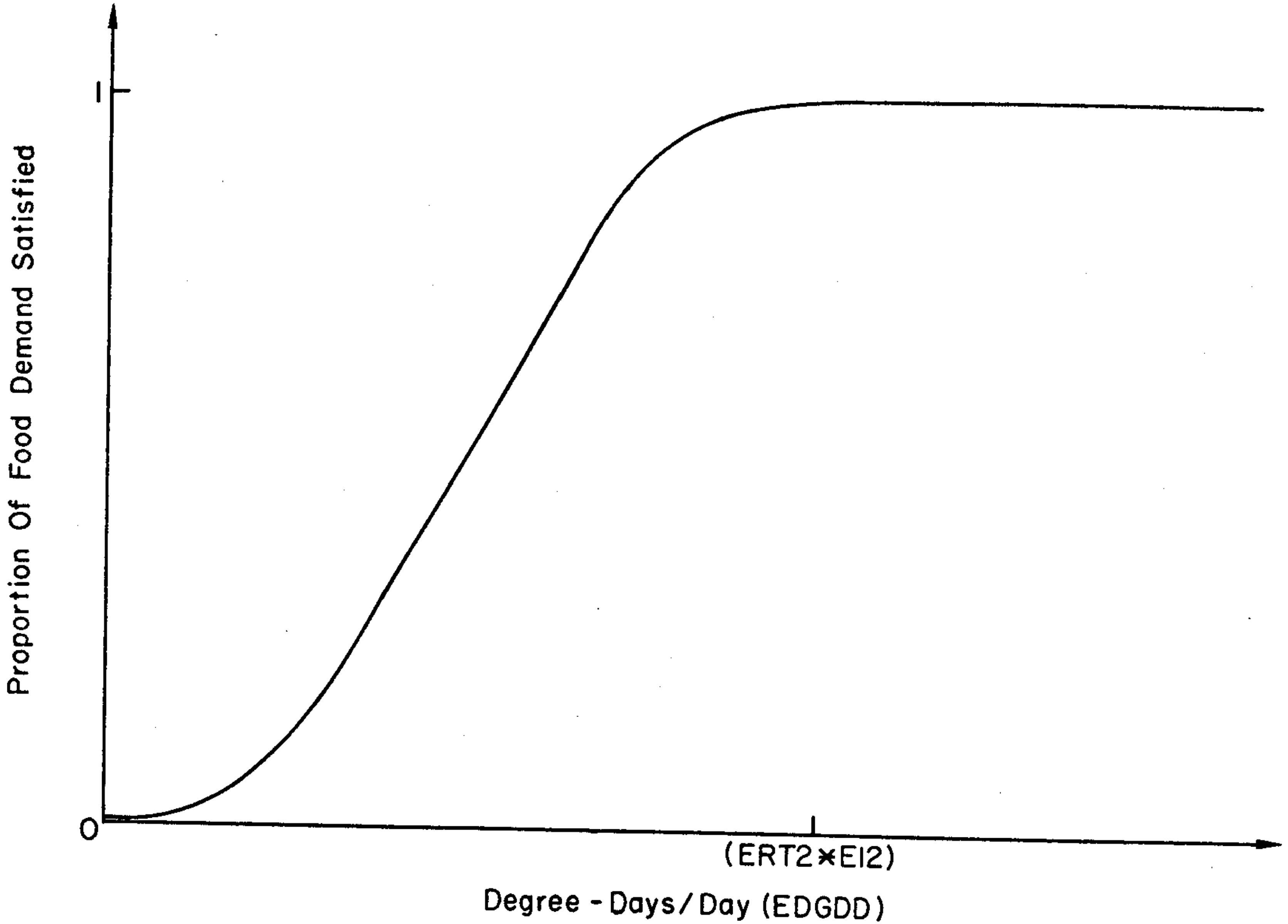


Fig. 4. Effect of degree-days on satisfaction of food demand. Degree-days is used to measure time available for feeding activity. Below a certain level (ERT2 * E12) the food demand can no longer be fully satisfied. The fraction of the food demand satisfied up to this level is given by a sine function.

It is assumed that insects will waste a certain amount of food while eating in addition to the food actually eaten. This is computed as a factor ECUT(I) times the food eaten.

Both for excretion and litter production it is further assumed that the particular factor is the same for all life stages within an insect group I. Stages that do not feed will, of course, not excrete or produce litter.

Respiration

The basal respiration for a stage of insects is computed by calling up function ERSPR. This basal respiration is considered to be the actual respiration of the stage for non-diapausing, inactive individuals (pupae and most egg stages). For active stages (non-diapausing larvae and adults) the ERSPR value is multiplied by an activity factor EACF. For diapausing stages it is divided by an inactivity factor EIAF.

The function ERSPR computes respiration of the whole insect group (I,J) as a function of individual average weight, maximum and minimum temperatures, and number of individuals in the particular stage. The daily maximum and minimum temperatures are used to generate the temperature at 12 different points during the day, by assuming that the daily temperature oscillation between these extremes can be described by two sine waves.

At each of these 12 points the temperature ETEMP is used together with the average weight EW(I,J) in the formula

```
ERESP = ERA * (10000. * EW(I,J)) ** ERB
            * EXP(EQ10 * (ETEMP - 20.0))
```

to compute the respiration per individual in ml 0_2/hr. The formula itself is from Van Hook (1971), however, the parameters ERA and ERB are taken from Hemmingsen (1960). Since Hemmingsen's equation is based on mg wet weight and the EW(I,J) is in g carbon, the EW(I,J) is multiplied by 10000 to correct for this (assuming 1 mg dry weight equals 0.4 mg carbon (H. W. Hunt, personal communication) and that 1 mg dry weight corresponds to 4 mg wet weight). The effect of temperature on respiration is handled by the last part of the equation. EQ10 corresponds to Q_{10} commonly used in texts on metabolism (EXP(10 * EQ10) = Q_{10}), and 20.0 is subtracted from ETEMP since the equation refers to a base temperature of 20°C.

The values of ERESP are summed up by ERSP for all the 12 points. This is then transformed into the total basal respiration for the stage (I,J), ERSPR, by multiplying by the number of individuals in the group (EN(I,J)) and the factor 0.00000067368. The last factor transforms the respiration from ml 0_2 consumed per 12 hours to g carbon produced per 24 hours. It is based on the following assumptions: 0.00048 cal/ml 0_2, 5.7 cal/mg dry weight, and 0.0004 g C/mg dry weight.

Predation

In principle, predation is handled much the same way as feeding by herbivores. However, since the predators are lumped in seven groups and since most of the predators are not dynamically represented in the model, some additional computations are necessary to compute the energy requirements of these predators. The computations pertaining to predation follow immediately after the weight computation parts in the model.

In order to obtain the energy requirements of the predator groups, the total actual and normal weights of the predator groups are needed as input in the function EPRED. For group 3 (Carabidae larvae) and group 4 (spiders) these weights are simply the sum of the corresponding weights of all individuals in the group. The respiration of the whole predator group is similarly derived by summing the respiration for the insect groups that constitute a group of predators. For spiders the cost of reproduction is also computed. The variables mentioned above are not available for the rest of the predator groups and a different approach is therefore necessary.

Biomass data of all these groups, except group 7, are provided by the parameters EPRD1, EPRD2, EPRD5, and EPRD6 (the values are in g C/m^2 and are loosely assessed from original data). For group 7 (small mammal) this information is provided by the consumer submodel in ELM. The part of the normal weight of these groups that has an impact on the insect populations is computed as the product of the biomass and the part of the diet which consists of arthropods (EPAD1, etc.). The actual weight of the groups are then constructed in such a way that function EPRED computes an energy requirement equal to what has been assessed for the particular groups based on their biomass (as given by EPRD1, etc.).

The calculations described above may seem rather redundant since the energy requirements could be derived more simply by multiplying the standing biomass of predators (EPRD1, etc.) by their corresponding energy requirement per day per unit biomass (EDU1, etc., in g C/day/g C biomass). It is done in order to force the calculations into the form described for groups 3 and 4. The reason for doing this is that eventually all the arthropod predators ought to be fully represented on the model, and then the approach described for groups 3 and 4 seems a reasonable one.

The next step in handling predation is to take care of the food selection of the predators. This corresponds closely to food selection for herbivores. The relative density of prey is computed from the biomass density (in g C/m^2) of a group (I,J) of prey. Then an assigned preference index (EPPRF(I,J,K)) of predator group K for prey group (I,J) is used to devise the corresponding relative rank (ERRNK(I,J,K)).

Finally the actual number of prey taken from an insect group (I,J) by a predator K is computed by calling up function EPRED. The input parameters are total predator weight (EWT), normal weight (ENWT), respiration (ERSP), energy cost of reproduction (ERPR), and degree-days per day (EDGDD). If the predator's preference index (EPPRF(I,J,K)) for a particular prey is zero, no prey will be taken from this group even if the predator shows no selective behavior. Otherwise, the predator's food demand, selectivity, and actual consumption of prey corresponds directly to the computations for herbivores. Similarly, the amount eaten is limited by the amount available and the time available for feeding, measured in degree-days. The vertebrate predators (groups 6 and 7) are not limited by the number of degree-days. Finally, the number of prey taken by a predator group is computed by dividing amount eaten (in g C) by the average weight of the prey group (in g C/individual).

Mortality Due to Non-predatory Causes

Apart from predation three causes of mortality are considered, mortality due to starvation, mortality due to unfavorable temperatures, and mortality due to unfavorable moisture conditions.

This is implemented in the model in the following way. The proportion of the population surviving starvation, ESFD, is computed from a comparison of the actual average weight of the stage to its normal or expected weight. When the average weight is below a factor EDW times the normal weight, at least some individuals in the group are starving. The fraction of the population in the group (I,J) surviving this is given by the equation below (see also Fig. 5).

$$\mathrm{ESFD} = 1.0 - (\mathrm{EW(I,J)} - \mathrm{EDW} * \mathrm{ENW(I,J)})^2/(\mathrm{EDW} * \mathrm{ENW(I,J)})^2$$

The implementation is the same for all groups and life stages.

A similar survival factor is computed for the fraction of the population surviving temperature and moisture conditions (ESTMP and ESMST, respectively). Both these fractions are computed by linear interpolation between given points (see Fig. 6 and 7). For the computation of ESTMP four points are provided and for ESMST three points. The x-values of these points vary from life stage to life stage, assuming that inactive stages (eggs, pupae, diapause) are more resistant to extremes than active larvae and adults. The independent variable is different if the insect is below ground (assumed to be 7.5 cm below soil surface) or above ground. The variable for temperature above ground is mean soil temperature on the surface AVSTM(1), generated by the abiotic part of ELM. The variable below ground is the mean soil temperature at 7.5 cm soil depth, EAVTM, computed in the beginning of CYCL1. The variables

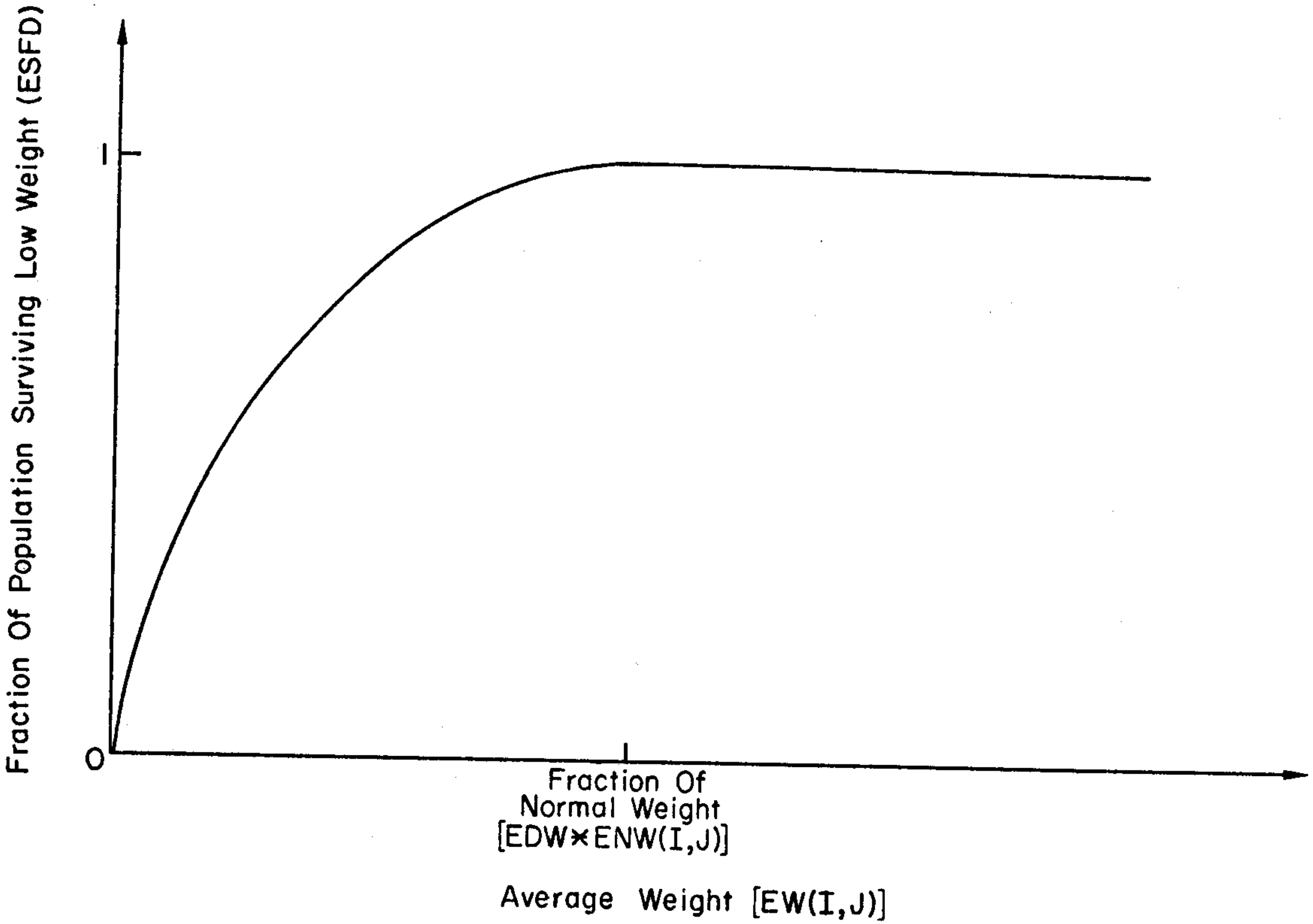


Fig. 5. Effect of starvation (measured as low average weight) on survival. Below a certain fraction of the normal weight (EDW) survival is affected by low average weight, implemented by a parabolic function.

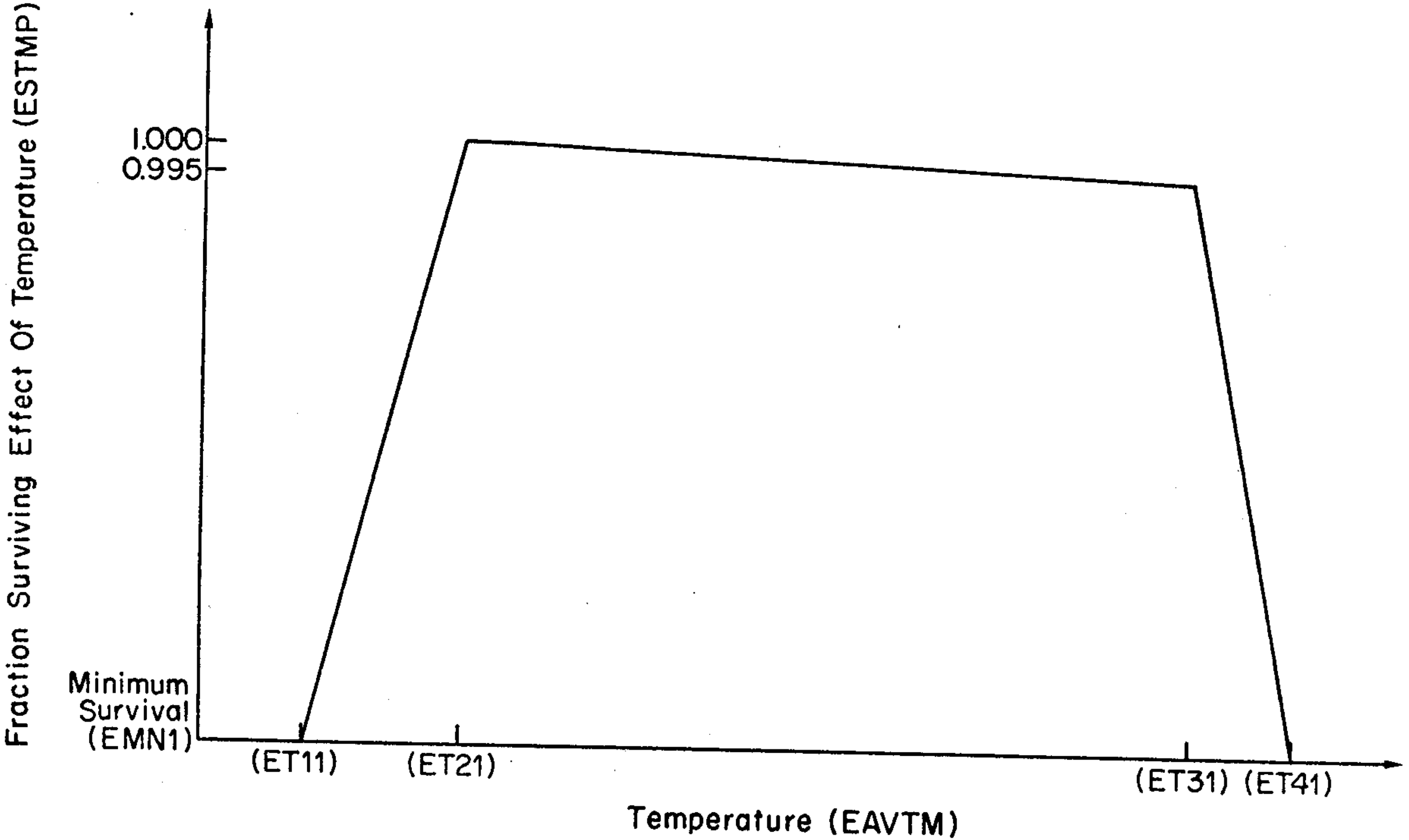


Fig. 6. Effect of temperature on survival. Both low and high temperatures are assumed to push the survival rate down to the minimum (EMN1). (The symbols shown are the ones used in the coding for inactive life stages below ground).

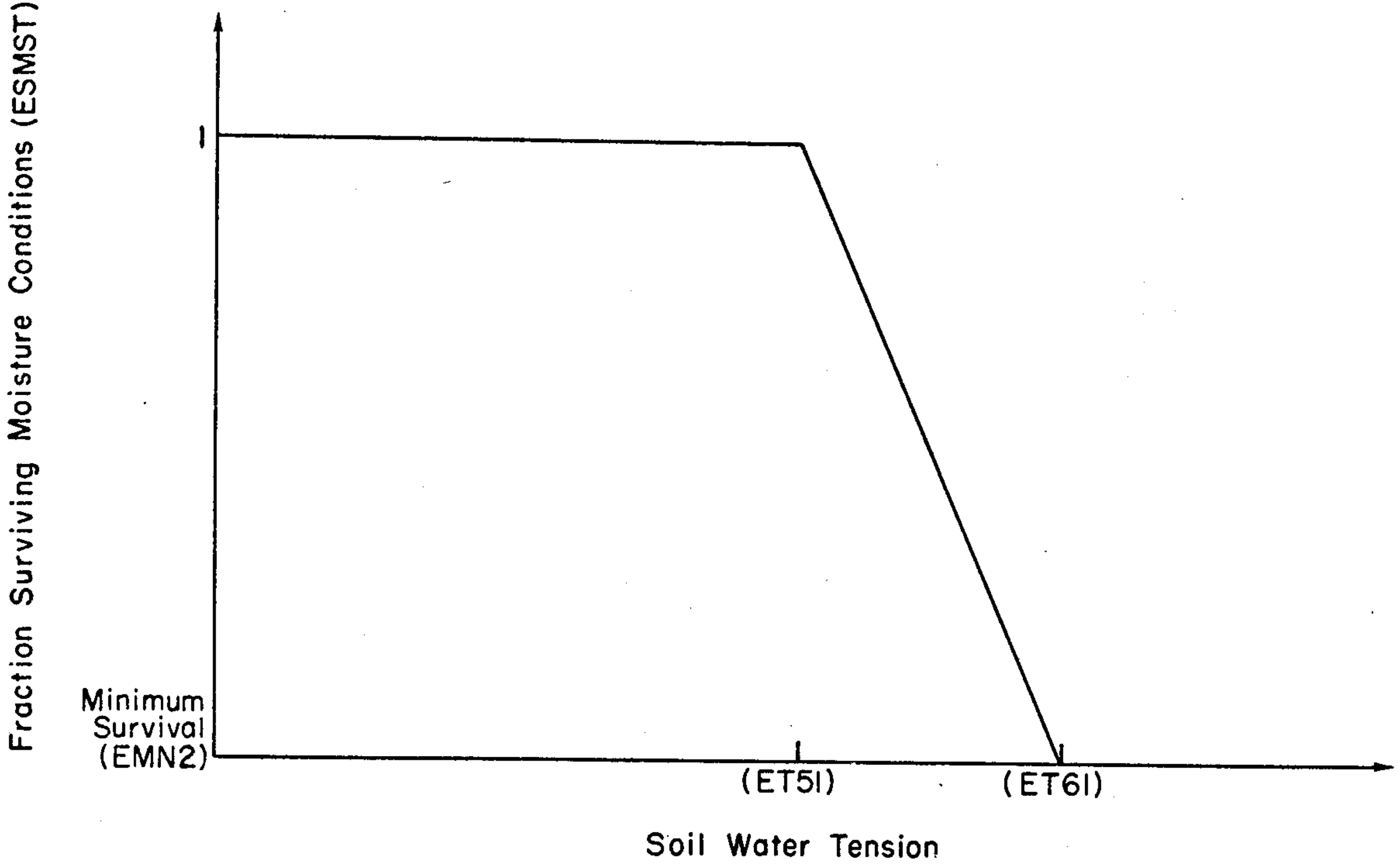


Fig. 7. Effect of moisture conditions on survival. Only dry conditions are assumed to be detrimental. (The symbols shown are the ones for inactive stages below ground. For stages above ground potential evapotranspiration is used instead of soil water tension as independent variable.)

for moisture conditions below and above ground are water tension between 4 and 15 cm, ATEN(3), and potential evapotranspiration, APEVA, both from the abiotic part of ELM.

The three survival factors are multiplied together and then subtracted from 1. This gives the non-predatory mortality rate of the population per day. To obtain the total non-predatory mortality of the insect group (I,J), ENFL(I,J), this factor is multiplied by the total number present in the group at the previous time step (EN(I,J)) and time step size (DT):

ENFL(I,J) = EN(I,J) * (1.0 - ESFD * ESMST * ESTMP) * DT

Recruitment of Individuals

The recruitment of insects into a group (I,J) is handled in three different ways in the model: (1) eggs produced by females, (2) transfer from inactive stages with rapid turnover (pupae and most egg stages), and (3) transfer from active stages (larvae and adults) and inactive stages with slow turnover (overwintering stages). The time when transfer can take place is limited between ETIM1(I,J) and ETIM2(I,J) for stage (I,J).

The recruitment into the first egg stage at each time step (ENFIJ(1,J)) is computed as the product of numbers of adults (EN(16,J)), proportion of females among adults of group I (EPPF(I)), egg production per female per day (EGGP), and time step size.

Egg production per female is computed in function EEPF as a maximum egg laying rate per female modified by temperature and nutrition of the female. The temperature is measured as degree-days, and the effect is expressed as the equation below (see also Fig. 8).

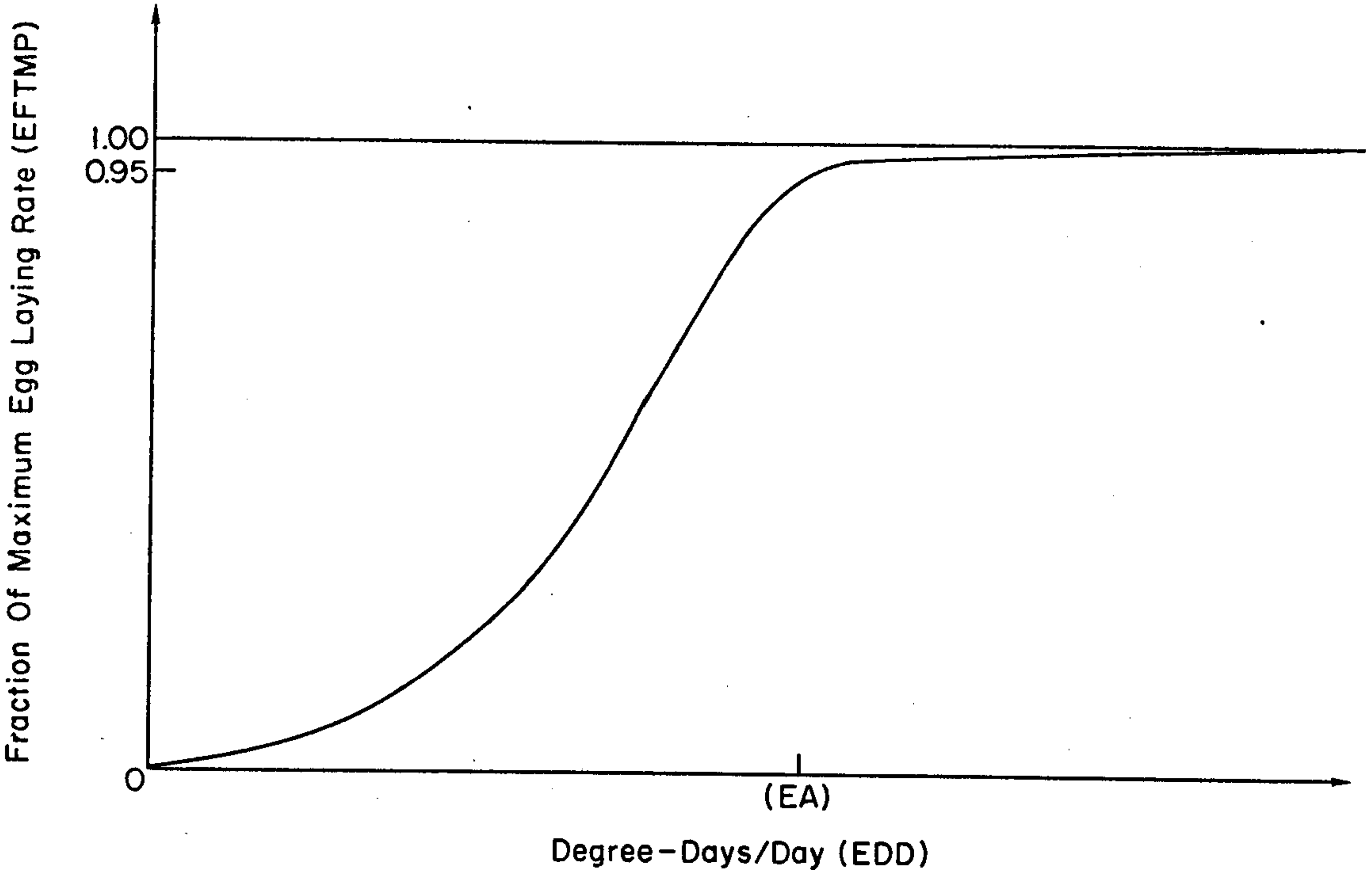


Fig. 8. Effect of temperature (measured as degree-days per day) on egg laying rate. When the number of degree-days is equal to a preset constant (EA) the egg laying rate is only 95% of the maximum possible rate. As the number of degree-days increases the rate approaches the maximum as an asymptote.

$$EFTMP = 1.0 - EXP(-2.99573 * EDD^2/EA^2)$$

where EFTMP is effect of temperature (EFTMP ϵ [0,1]), EDD is the number of degree-days per day, and EA is a contant number of degree-days per day. The constant -2.99573 is actually a parameter determining the form of the curve in Fig. 8. The value was chosen so that EFTMP=0.95 when EDD=EA.

In order to determine the effect of nutrition on egg production, the average weight of the females is compared to a preset minimum weight for egg production. An index of nutrition is computed as the difference between average and minimum weight divided by average weight (EDV=(EWT-EEWT)/EWT). EFNUT is then computed as the linear interpolation between the points (0.0, 0.0) and (0.2, 1.0) (see Fig. 9).

The transfer of individuals out of egg and pupal stages is mostly handled in an "all or nothing" manner. That is, until the previous stage has accumulated a certain number of degree-days (EDG(I-1,J)) no transfer into the stage EN(I,J) occurs. When the sum of accumulated degree-days (ESUM(I-1,J)) exceeds EDG(I-1,J), all the individuals in the stage (I-1,J) are transferred to the stage (I,J). Transfer out of first generation Hemiptera-Homoptera eggs and grasshopper eggs is handled as described below since these egg stages do not have the rapid dynamics characteristic of other egg and pupal stages in the model.

The transfer out of all other stages (i.e., active and diapause stages) is a gradual process as a function of degree-days per day. Some of these stages have to accumulate a certain number of degree-days before transfer out of this stage can occur. However, once this is achieved the transfer is handled in the same way as for the other stages. The stages, for which the accumulated degree-day mechanism was

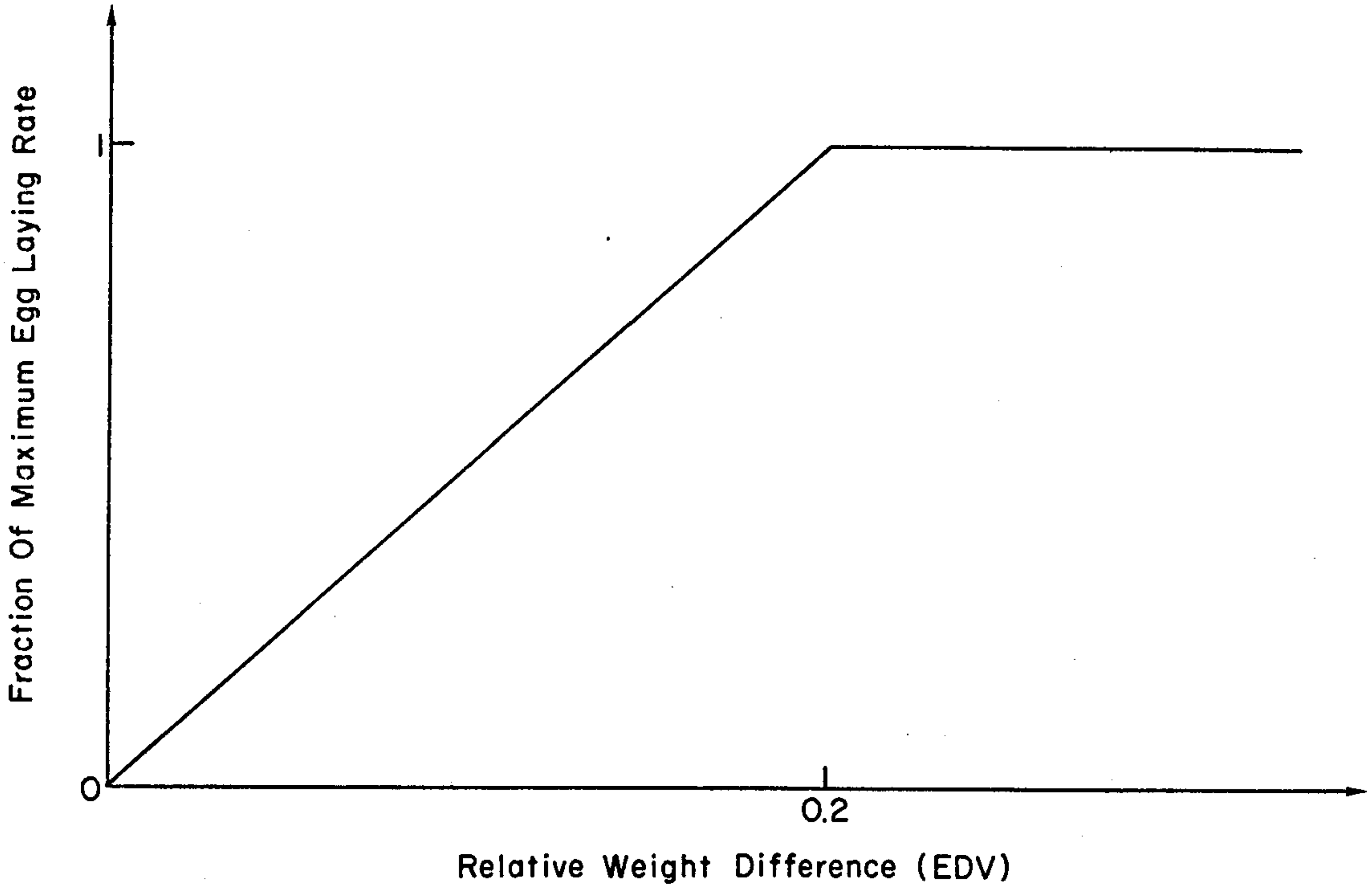


Fig. 9. Effect of nutrition (measured as average weight) on egg laying rate. The independent variable is the relative weight difference computed as the difference between the average weight and the minimum weight for reproduction divided by the average weight (EDV = (EWT - EEWT)/EWT).

not seen as appropriate have transfer either into or out of a diapausing stage. For these stages it is assumed that the development necessary to initiate diapause is automatically achieved and that the necessary number of cold days during diapause for normal development is automatically realized. The transfer into diapause is then simply anti-proportional to the number of degree-days, and the transfer out of diapause is a function of degree-days. The actual fraction to be transferred from a category is computed by the function ALINT2 as an interpolation between two points.

Finally, in spiders and grasshoppers the insects are divided according to sex before they reach the adult stage. In spiders the transfer from stage 10 goes to stages 11 and 14 and in grasshoppers from stage 1 to stages 2 and 9. The proportion going to the two stages is the same as the sexual ratio in adults. Otherwise, the transfer is as described above.

Computing Weights and Updating the Numbers in Each Category

A very important part of the calculations in CYCL1 is computation of normal and actual individual weights and the updating of the numbers in each insect category.

In order to assess effects of too low weight for individuals the normal or expected weight of an individual is generated to give a base for comparison with the actual average weight. In the model it is assumed that a life stage (I,J) will have the weight ENW1(I,J) until a time ETM1(I,J). Then it will increase to the weight ENW2(I,J) during the time ETIME(I,J). This increase is a linear function of time computed by function ALINT2. The time ETM1(I,J) is determined as the time when transfer into the stage (I,J) first takes place. Actually, only active larvae and adult female spiders show growth (i.e., weight increase), for the rest of the stages ENW2(I,J)=ENW1(I,J).

The updating of the numbers in an insect category (I,J) is calculated by adding the incoming individuals and subtracting the outgoing ones:

$$\begin{aligned}\mathrm{EN}(I,J)_{t+1} = \mathrm{EN}(I,J)_t + \mathrm{ENFIJ}(I,J)_t - \mathrm{ESMP}(I,J)_t \\ - \mathrm{ENFL}(I,J)_t - \mathrm{ENFIJ}(I+1,J)_t\end{aligned}$$

where EN(I,J) is the number of individuals in group (I,J)

ENFIJ(I,J) is the number coming into group (I,J)

ENFIJ(I+1,J) is the number going from group (I,J) into (I+1,J)

ESMP(I,J) is the number taken by predators from group (I,J)

ENFL(I,J) is the number in group (I,J) dying from other causes.

The subscript t refers to time t and t+1 to time t+1. There are a few exceptions to the equation above where the life cycles do not follow a simple pattern. For example, all adult categories have no outflow into a next life stage since eggs are new individuals and not simply old individuals pushed over into a new life stage. These categories are group (16,J) for all insects, group (13,2) for spiders, group (6,4) for Hemiptera-Homoptera, and group (8,7) for grasshoppers. In addition, a few categories will have two outflows to other life stages since they are split according to sex before the adult stage. Group (10,2) for spiders splits and goes to stages (11,2) and (14,2). Likewise for group (1,7) for grasshoppers which goes to (2,7) and (9,7).

Computing the new average weight for the individuals in a category (I,J) is very simple in principle. The new average weight for the individuals left in group (I,J) is equal to the old weight, plus what is eaten per individual, and minus what is lost through respiration and excretion.

$$EW(I,J)_{t+1} = EW(I,J)_t + (EFFI(I,J)_t - EFIE(I,J)_t - EFIR(I,J)_t)/EN(I,J)_t$$

where $EW(I,J)_{t+1}$ is the new average weight for the individuals left in group (I,J) at time t+1

$EW(I,J)_t$ is the old average weight for group (I,J)

$EFFI(I,J)_t$ is the amount eaten by group (I,J) at time t

$EFIE(I,J)_t$ is the amount excreted by group (I,J) at time t

$EFIR(I,J)_t$ is the amount respired by group (I,J) at time t

$EN(I,J)_t$ is the number of individuals in group (I,J) at time t

This new average weight will then have to be adjusted for the new individuals coming into the group (I,J). In the coding this is handled by first computing the new total weight of the group (I,J), then updating the numbers in the group, and finally computing the new average weight.

Total weight:

$$EW'(I,J)_{t+1} = ENFIJ(I,J)_t * EIW + (EN(I,J)_t - ENEG_t) * EW(I,J)_{t+1}$$

where $EW'(I,J)_{t+1}$ is total weight of group (I,J) at time t+1

EIW is average weight of new individuals

$EW(I,J)_{t+1}$ is average weight of old individuals left in group

$ENFIJ(I,J)_t$ is number of individuals entering group (I,J)

$EN(I,J)_t$ is number of individuals in group at time t

$ENEG_t$ is number of individuals removed from group during time step t.

New average weight for all the $EN(I,J)_{t+1}$ individuals in group (I,J):

$$EW(I,J)_{t+1} = EW'(I,J)_{t+1}/EN(I,J)_{t+1}$$

In the program itself certain additional calculations have to be carried out in order to solve problems such as division by zero, which might otherwise occur when all individuals in a group either die or proceed to the next life stage.

Flows of Carbon Between State Variables

As mentioned in the beginning, the main purpose of the flow section in the coding is to compute the flows of carbon between the state variables in the model and the ones in ELM. However, at the very beginning of the FLOW part a block of coding was put in to insure that the flow of individuals out of an insect category does not exceed the number present in this category. These computations could also have been performed at the very end of CYCL1.

The state variables for the insect groups in the program have no indices which correspond directly to the notation used for the rest of the variables. The reason is that SIMCOMP assumes one dimensional state variables and that the choice of indices was restricted by the other parts of ELM. The state variables (g C/m^2 in each insect category) are X(320-399) and X(520-551). For a definition of the variables see the end of Appendix C.

The flows of carbon within a main insect group (for example, Carabidae) and the appropriate compartments in ELM are depicted in Fig. 10. A detailed description follows below.

An insect category (I,J) will have two connections to other life stages in the same main group J. There will be a flow in from the

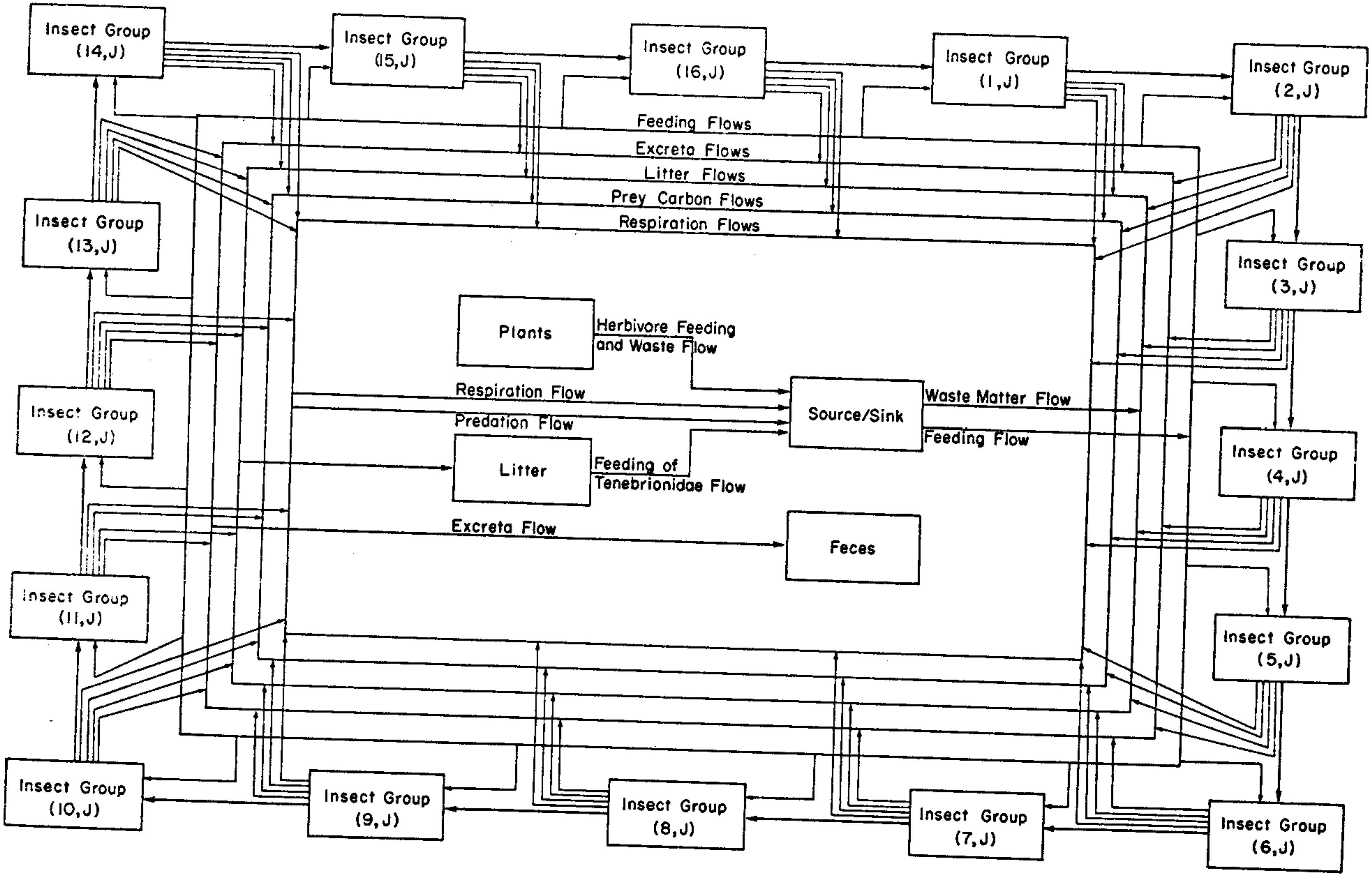


Fig. 10. Flow chart for flows of carbon from stage to stage within a main insect group and between these stages and the interactive compartments of ELM.

previous life stage computed as the number transferred times the average weight of these individuals (in the coding: ENFIJ(I,J) * EW(I-1,J)). There will also be a similar flow out to the next life stage (ENFIJ (I+1,J) * EW(I,J)). For adults this flow constitutes the flow of egg carbon to the first egg stage.

The losses suffered by the group (I,J) through predation are flowed to a source/sink rather than to the appropriate state variable for the predator. The reason is that the majority of the predators are not incorporated in the model, and, besides, this allows for the same representation of feeding flows for both predaceous and herbivorous insects. Sufficient detail in the predation process should still be achieved since the elements used to compute these flows are computed specifically for a predator-prey pair. The amount of carbon flowed from insect group (I,J) is computed as the number taken by all predators times their average weight (in the coding: ESMP(I,J) * EW(I,J)). In addition to this the amount of carbon produced by respiration by group (I,J) is also flowed into the source/sink (i.e., EFIR(I,J)). Individuals dying from non-predatory causes are moved to the compartment for litter (in the coding: ENFL(I,J) * EW(I,J)).

The flow of carbon into an insect group (I,J) through feeding (EFFI(I,J)) is taken from the source/sink. Another flow from the source/sink going to litter consists of carbon wasted during the feeding of the insects. This is the sum of the amount wasted by plant feeders and the amount wasted by Carabidae larvae and spiders. For the other predators no waste factor has been computed.

To balance these two flows out of the source/sink other flows from the food material (plants or prey) go into the source/sink. They consist of the amount actually ingested by the insect plus the amount wasted. For plants the amount flowed from a food category is the sum of the amount eaten and wasted by all consumers (in the coding: EFC(L,K) summed over all K for each L). An additional flow goes from litter to the source/sink. This is to balance off the amount eaten by adult Tenebrionidae (category 16,6) which is considered to be a litter feeder.

Finally, grams carbon excreted by an insect group is flowed to the feces compartment in ELM.

Note:

A word of caution, the FLOW part of the coding has not been tested in computer runs, and structural weaknesses and errors may therefore exist.

Minor changes in the data and in subroutine START are necessary to provide the connections between the initialized X variables and the EN variables. As these parts of the model are at present, these connections must be provided explicitly by the user. The reason is that the model was previously run without the X variables.

In the flow constituting dead animals to litter the flow ends in compartments for surface litter. A better approach would be to send dead animals below ground to the belowground litter compartment and only the aboveground dead animals to the present compartments.

LITERTURE CITED

Anway, J. C. 1975. A canonical mammalian model. US/IBP Grassland Biome Preprint No. 153. Colorado State Univ., Fort Collins. 82 p.

Cole, C. V., G. S. Innis, and J. W. B. Stewart. 1975. Simulation of phosphorus cycling in semiarid grasslands. US/IBP Grassland Biome Preprint No. 157. Colorado State Univ., Fort Collins. 61 p.

Ellis, J. E., J. A. Wiens, C. F. Rodell, and J. C. Anway. 1975. A conceptual model of diet selection as an ecosystem process. J. Theoret. Biol. (In press).

Gustafson, J. D., and G. S. Innis. 1973. SIMCOMP version 3.0 user's manual. US/IBP Grassland Biome Tech. Rep. No. 218. Colorado State Univ., Fort Collins. 149 p.

Hemmingsen, A. M. 1960. Metabolism in relation to body size. Rep. Steno Mem. Hosp. Nordisk Insulin Lab. 9:1-110.

Hunt, H. W. 1975. A simulation model for decomposition in grasslands. US/IBP Grassland Biome Preprint No. 155. Colorado State Univ., Fort Collins. 62 p.

Innis, G. S. 1975*a*. Rationale and procedures for building a simulation model for grasslands. US/IBP Grassland Biome Preprint No. 149. Colorado State Univ., Fort Collins. 18 p.

Innis, G. S. 1975*b*. Behavior and response of ELM: A simulation model for grasslands. US/IBP Grassland Biome Preprint No. 159. Colorado State Univ., Fort Collins. 51 p.

Innis, G. S., and J. D. Gustafson. 1975. Compartmental-flow/event-oriented simulation language:SIMCOMP 3.0. US/IBP Grassland Biome Preprint No. 150. Colorado State Univ., Fort Collins. 24 p.

Parton, W. J. 1975. Abiotic section of the ELM grassland ecosystem model. US/IBP Grassland Biome Preprint No. 151. Colorado State Univ., Fort Collins. 51 p.

Reuss, J. O., and G. S. Innis. 1975. A grassland nitrogen flow simulation model. US/IBP Grassland Biome Preprint No. 156. Colorado State Univ., Fort Collins. 41 p.

Rodell, C. F. 1975. A grasshopper model for a grassland ecosystem. US/IBP Grassland Biome Preprint No. 154. Colorado State Univ., Fort Collins. 73 p.

Sauer, R. H. 1975. A simulation model for grassland primary producer phenology and biomass dynamics. US/IBP Grassland Biome Preprint No. 152. Colorado State Univ., Fort Collins. 97 p.

Steinhorst, R. K., H. W. Hunt, and G. S. Innis. 1975. Sensitivity analyses of the ELM model. US/IBP Grassland Biome Preprint No. 158. Colorado State Univ., Fort Collins. 44 p.

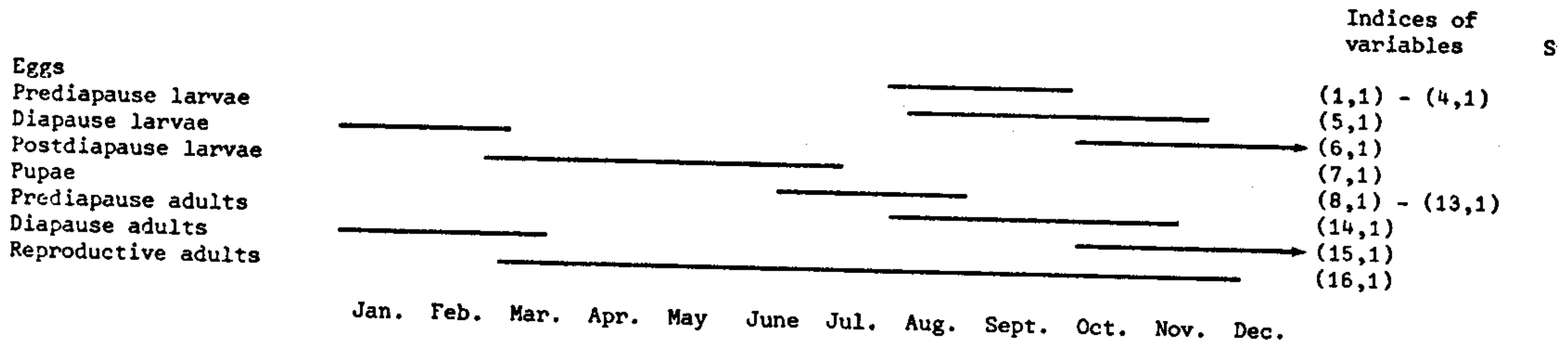
Van Hook, R. I., Jr. 1971. Energy and nutrient dynamics of spider and orthopteran populations in a grassland ecosystem. Ecol. Monogr. 41(1):1-26.

Woodmansee, R. G. 1975. Critique and analysis of the grassland ecosystem model ELM 73. US/IBP Grassland Biome Preprint No. 160. Colorado State Univ., Fort Collins. 52 p.

APPENDIX A

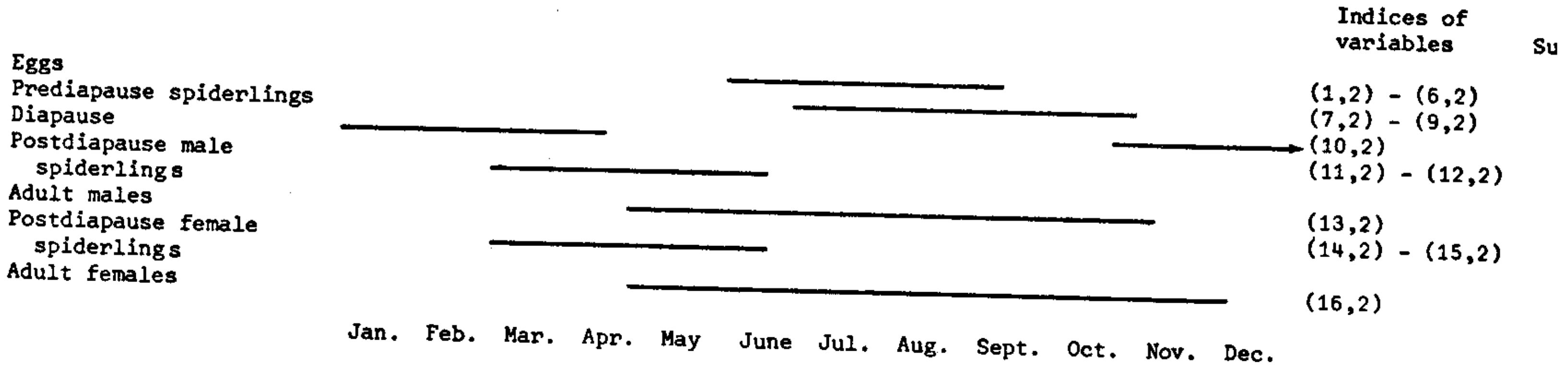
LIFE-STAGES AND APPROXIMATE TIMING OF LIFE-HISTORY EVENTS AS ASSUMED IN THE MODEL

Life cycle for the first group, Carabidae



2 years per generation, overwintering by larvae first winter and by adults second winter.

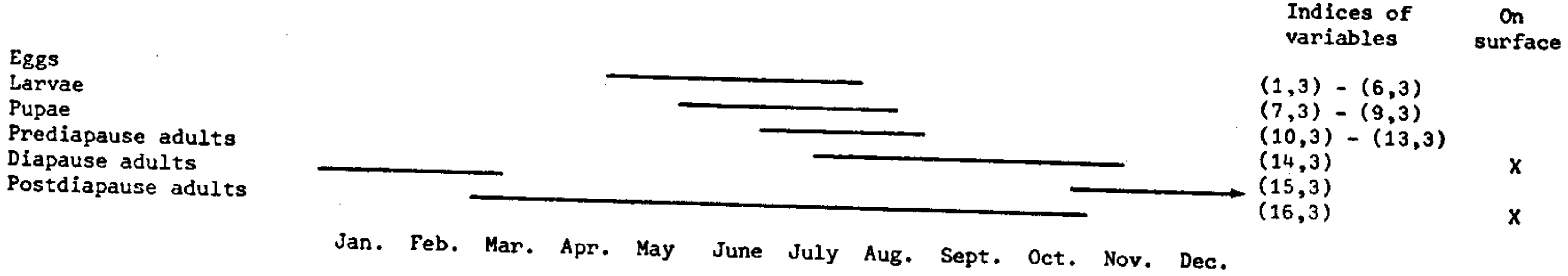
Life cycle for the second group, Araneida



1 year per generation, overwintering by third-fifth instar.

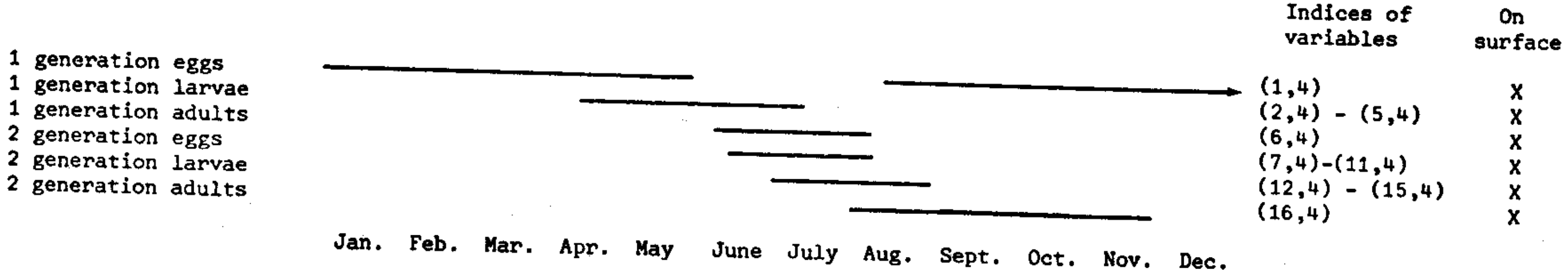
APPENDIX A (cont.)

Life cycle for the third group, Chrysomelidae-Curculionidae



1 year per generation, overwintering by adults.

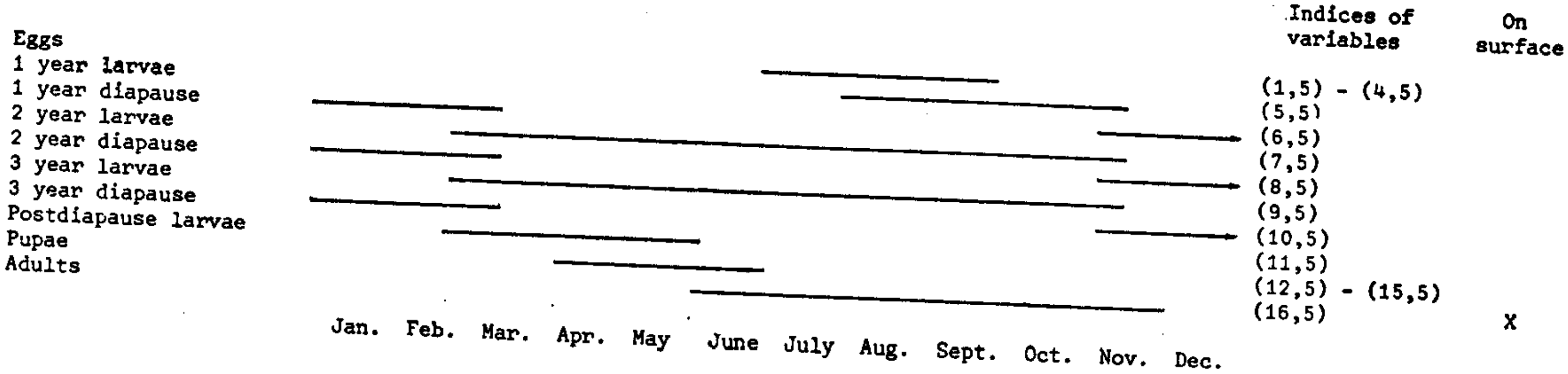
Life cycle for the fourth group, aboveground Hemiptera-Homoptera



2 generations per year, overwinter as first generation eggs.

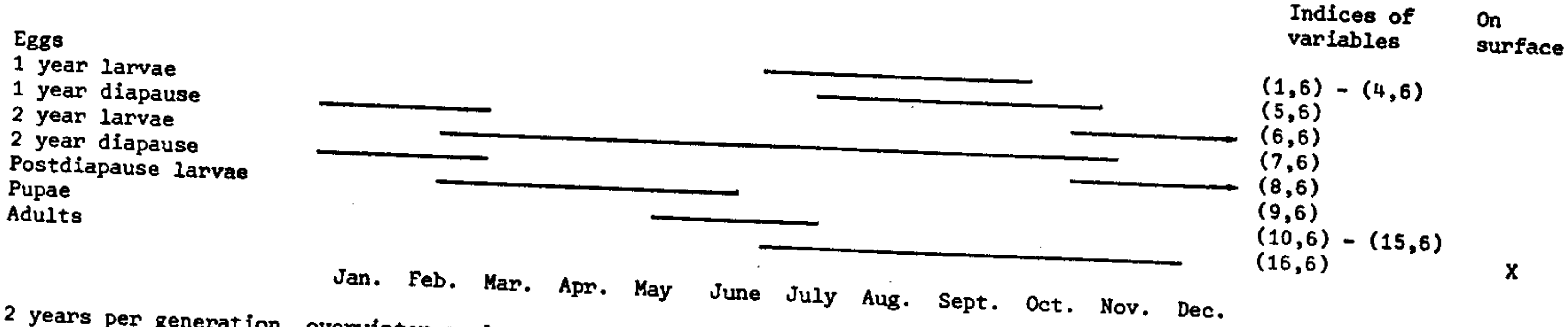
APPENDIX A (cont.)

Life cycle for the fifth group, Scarabaeidae



3 years per generation, overwinter as larvae.

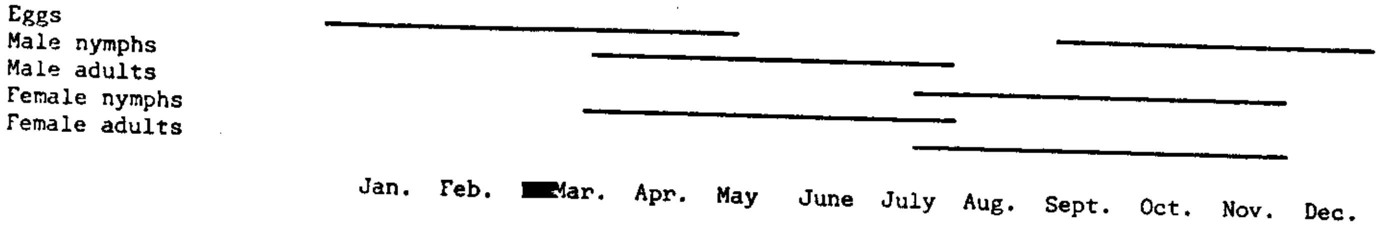
Life cycle for the sixth group, Tenebrionidae



2 years per generation, overwinter as larvae.

APPENDIX A (cont.)

Lif[redacted] cycle for the seventh group, Orthoptera



	Indices of variables	On surface
Eggs	(1,7)	
Male nymphs	(2,7) - (7,7)	X
Male adults	(8,7)	X
Female nymphs	(9,7) - (15,7)	X
Female adults	(16,7)	X

1 year per generation, overwinter as e[redacted]gs.

APPENDIX B

LIST OF DRIVING VARIABLES FOR THE MODEL

From the abiotic submodel of ELM:

APEVA - Potential evapotranspiration rate (cm water/day)

ATEN(3) - Water tension at 4-15 cm (negative bars)

AVMN - Average minimum air temperature for time step DT (°C at 2 m)

AVMX - Average maximum air temperature for time step DT (°C at 2 m)

AVSTM(1-3) - Average daily soil-temperature for time step DT (°C at 0, 15, 30 cm depth)

From the producer submodel of ELM:

PHEN (1-5) - Mean phenophase of producer 1-5 (g C/m^2/day)

X(200-204) - Live shoot state variables (g C/m^2)

X(210-214) - Crowns or storage state variables (g C/m^2)

X(230-234) - Seed state variables (g C/m^2)

X(240-244) - Live root state variables (g C/m^2)

From the mammalian consumer submodel of ELM:

X(303-305) - State variables for small mammals (g C/m^2)

APPENDIX C

DEFINITIONS OF VARIABLES AND PARAMETERS

The variables occurring in STORAGE or CYCL1 follow.

Variable name	Definition	Dimension
X(320-335)	Biomass in the different life stages of Carabidae	g C/m^2
X(336-351)	Biomass in the different life stages of Araneida	g C/m^2
X(352-367)	Biomass in the different life stages of Chrysomelidae/Curculionidae	g C/m^2
X(368-383)	Biomass in the different life stages of Homoptera/Hemiptera	g C/m^2
X(384-399)	Biomass in the different life stages of Scarabaeidae	g C/m^2
X(520-535)	Biomass in the different life stages of Tenebrionidae	g C/m^2
X(536-551)	Biomass in the different life stages of Orthoptera	g C/m^2
E12	Constant	Degree-days (°C)
E22	Variable expressing temperature effect on ESEF(I,J)	Prey found/predator/day
EA	Return variable from subroutine ESTEP (EA also occurs in other subprograms with a different meaning, see the respective subprograms)	Nondimensional
EACF	Factor adjusting respiration for inactive insects	Nondimensional
EASK(I)	Assimilation coefficient for insect group I, fraction of food ingested	Nondimensional
EAVTM	Average daily temperature at 7.5 cm soil depth	°C
EA0	Half temperature amplitude at 2 m	°C
EA1	Half temperature amplitude at surface	°C

APPENDIX C (cont.)

Variable name	Definition	Dimension
EA2	Half temperature amplitude at 7.5 cm depth	°C
EA3	Half temperature amplitude at 15 cm depth	°C
EB	Return variable from subroutine ESTEP (EB also occurs in other subprograms with a different meaning, see the respective subprograms)	Degree-days (°C)
ECUT(I)	Factor for food wasted in feeding, as proportion of food ingested by insect group I	Nondimensional
EDG(I,J)	Accumulated degree-days necessary for development into next stage, for insect group (I,J)	Degree-days (°C)
EDGDD	Temporary variable for degree-days	Degree-days (°C)
EDGDS	Function, degree-days computed from daily maximum and minimum temperatures and developmental zero	Degree-days (°C)
EDGD1	Number of degree-days per day experienced by an insect at the surface	Degree-days (°C)
EDGD2	Number of degree-days per day experienced by an insect at 7.5 cm soil depth	Degree-days (°C)
EDTM1	Temperature amplitude at surface	°C
EDTM2	Temperature amplitude at 7.5 cm soil depth	°C
EDTM3	Temperature amplitude at 15 cm soil depth	°C
EDUI	Temporary variable	Nondimensional
EDU1	Energy demand per unit biomass for insect predators	Nondimensional
EDU6	Energy demand per unit biomass for birds	Nondimensional
EDU7	Energy demand per unit biomass for small mammals	Nondimensional
EDW	Fraction of normal weight below which actual weight affects survival	Nondimensional
EDZ1	Developmental zero for surface insects	°C

APPENDIX C (cont.)

Variable name	Definition	Dimension
EDZ2	Developmental zero for insects at 7.5 cm depth	°C
EEPF	Function, computes number of eggs laid per female per day	Individuals/female/day
EFC(L,K)	Consumption of food type L by consumer K	g C/m^2
EFD(L)	Density of food type L	g C/m^2
EFFI(I,J)	Food intake by insect group (I,J)	g C/m^2
EFIE(I,J)	Excretion by insect group (I,J)	g C/m^2
EFIR(I,J)	Respiration by insect group (I,J)	g C/m^2
EFLG	Integer, flag required by ALINT2	Nondimensional
EFOD	Function, food eaten by herbivore insect group (I,J)	g C/m^2
EFOOD	Summing variable, food eaten by predacious insect group (I,J)	g C/m^2
EGGP	Number of eggs produced per female per day	Individuals/female/day
EHAF	Proportion of present stage going to the next stage in Hemiptera-Homoptera and Orthoptera	Nondimensional
EI	Integer, counting variable and index	Nondimensional
EIAF	Correction factor for respiration of diapausing insects	Nondimensional
EIW	Weight of individual on entering a stage	g C/individual
EJ	Integer, counting variable, and index	Nondimensional
EK	Integer, counting variable, and index	Nondimensional
EL	Integer, counting variable, and index	Nondimensional
EM(I)	Integer array relating index for herbivore (I) to that of species group (EM(I))	Nondimensional
EMER(I)	Maximum egg-laying rate per female in group (I)	Individuals/female/day

APPENDIX C (cont.)

Variable name	Definition	Dimension
EMN1	Minimum fraction surviving effect of temperature	Nondimensional
EMN2	Minimum fraction surviving effect of moisture	Nondimensional
EMW(I)	Minimum weight for reproduction by female of group (I)	g C/individual
EN(I,J)	Density of insects in group (I,J)	Individuals/m^2
ENEG	Number of insects at a stage (I,J) dying and proceeding to next stage	Individuals/m^2
ENFIJ(I,J)	Numbers proceeding into stage (I,J)	Individuals/m^2
ENFL(I,J)	Numbers dying from non-predatory causes	Individuals/m^2
ENFP(I,J,K)	Number of prey from group (I,J) taken by predator K	Individuals/m^2
ENW(I,J)	Expected or normal weight of an individual in group (I,J)	g C/individual
ENW1(I,J)	Expected average weight at start of existence of group (I,J)	g C/individual
ENW2(I,J)	Expected average weight at end of existence of group (I,J)	g C/individual
EPAD1	Proportion of arthropods in diet for *Onychomys ochrogaster*	Nondimensional
EPAD2	Proportion of arthropods in diet for *Peromyscus maniculatus*	Nondimensional
EPAD3	Proportion of arthropods in diet for *Spermophilus tridecemlineatus*	Nondimensional
EPAD5	Proportion of arthropods in diet for ants	Nondimensional
EPAD6	Proportion of arthropods in diet for birds	Nondimensional

APPENDIX C (cont.)

Variable name	Definition	Dimension
EPPF(I)	Proportion of productive females among adults in group (I)	Nondimensional
EPPRF(I,J,K)	Food preference of predator K for prey (I,J)	Nondimensional
EPRED	Function, number of prey taken by a predator group	Individuals
EPRD1	Biomass of Asilidae larvae	g C/m^2
EPRD2	Biomass of Asilidae adults	g C/m^2
EPRD5	Biomass of ants	g C/m^2
EPRD6	Biomass of birds	g C/m^2
EPRF(L,K)	Assigned preference index for food L, consumer K	Nondimensional
EQ10	Parameter representing Q_{10} value	1/°C
ERA	Constant used in ERSPR relating weight to respiration	µl O_2/mg/hour
ERB	Constant used in ERSPR relating weight to reproduction	Nondimensional
EREP	Energy cost of reproduction for reproductive stage	g C/m^2/day
EREP1	Energy cost of reproduction for first generation Hemiptera-Homoptera	g C/m^2/day
ERFD(L)	Relative density of food type L	Nondimensional
ERNK(L,K)	Relative rank of food L for consumer K	Nondimensional
ERPD(I,J)	Relative density of prey (I,J)	Nondimensional
ERPF(L,K)	Relative preference index for food L, consumer K	Nondimensional
ERPR	Energy cost of reproduction for group (I,J)	g C/m^2/day
ERRNK(I,J,K)	Rank of prey (I,J) for predator K	Nondimensional

APPENDIX C (cont.)

Variable name	Definition	Dimension
ERSPR	Function, respiration of group (I,J) (basal metabolism)	g C/m^2/day
ERSP1	Respiration for predator group 1	g C/time step
ERSP2	Respiration for predator group 2	g C/time step
ERSP3	Respiration for predator group 3	g C/time step
ERSP4	Respiration for predator group 4	g C/time step
ERSP5	Respiration for predator group 5	g C/time step
ERSP6	Respiration for predator group 6	g C/time step
ERSP7	Respiration for predator group 7	g C/time step
ERT1	Constant used in EFOD, fraction of food density	Nondimensional
ERT2	Constant used in EFOD, fraction of E12	Nondimensional
ERT4	Constant used in EFOD, fraction of food density	Nondimensional
ESFD	Fraction surviving below normal weight	Nondimensional
ESMF	Summing variable, density of all food types	g C/m^2
ESMP(I,J)	Total number taken by predators from group (I,J)	Individuals/m^2
ESMPF(K)	Summing variable, all preference indices for consumer K	Nondimensional
ESMRK(K)	Summing variable, all ranks for consumer K	Nondimensional
ESMST	Fraction surviving moisture conditions	Nondimensional
ESTEP	Subroutine name	
ESTMP	Fraction surviving temperature conditions	Nondimensional
ESUM(I,J)	Number of degree-days accumulated by group (I,J)	Degree-days (°C)

APPENDIX C (cont.)

Variable name	Definition	Dimension
ETIME(I,J)	Approximate duration of stage (I,J) for determination of normal weight	Days
ETIM1(I,J)	Time after which transfer into stage (I,J) can occur	Days
ETIM2(I,J)	Time until which transfer into stage (I,J) can occur	Days
ETMN1	Minimum daily temperature at the soil surface	°C
ETMN2	Minimum daily temperature at the 7.5 cm soil depth	°C
ETMN3	Minimum daily temperature at the 15.0 cm soil depth	°C
ETMX1	Maximum daily temperature at the soil surface	°C
ETMX2	Maximum daily temperature at 7.5 cm soil depth	°C
ETMX3	Maximum daily temperature at 15.0 cm soil depth	°C
ETM1(I,J)	Time for start of existence of group (I,J)	Days
ETM2	Time for end of existence of group	Days
ET11	Point on x-axis to determine ESTMP for inactive stages	°C
ET12	Point on x-axis to determine ESTMP for active stages	°C
ET21	Point on x-axis to determine ESTMP for inactive stages	°C
ET22	Point on x-axis to determine ESTMP for active stages	°C
ET31	Point on x-axis to determine ESTMP for inactive stages	°C

APPENDIX C (cont.)

Variable name	Definition	Dimension
ET32	Point on x-axis to determine ESTMP for active stages	°C
ET41	Point on x-axis to determine ESTMP for all stages	°C
ET51	Point on x-axis to determine ESMST for inactive stages below ground	-bars water pressure
ET52	Point on x-axis to determine ESMST for active stages below ground	-bars water pressure
ET53	Point on x-axis to determine ESMST for active stages above ground	cm of water/day
ET54	Point on x-axis to determine ESMST for inactive stages above ground	cm of water/day
ET61	Point on x-axis to determine ESMST for inactive stages below ground	-bars water pressure
ET62	Point on x-axis to determine ESMST for active stages below ground	-bars water pressure
ET63	Point on x-axis to determine ESMST for active stages above ground	cm of water/day
ET64	Point on x-axis to determine ESMST for inactive stages above ground	cm of water/day
EW(I,J)	Average actual weight for insects in group (I,J)	g C/individual
EWT1	Expected or normal weight of all group 1 predators	g C/individual
EWT2	Expected or normal weight of all group 2 predators	g C/individual
EWT3	Expected or normal weight of all group 3 predators	g C/individual
EWT4	Expected or normal weight of all group 4 predators	g C/individual

APPENDIX C (cont.)

Variable name	Definition	Dimension
EWT5	Expected or normal weight of all group 5 predators	g C/individual
EWT6	Expected or normal weight of all group 6 predators	g C/individual
EWT7	Expected or normal weight of all group 7 predators	g C/individual
EW1	Actual weight of all group 1 predators	g C/individual
EW2	Actual weight of all group 2 predators	g C/individual
EW3	Actual weight of all group 3 predators	g C/individual
EW4	Actual weight of all group 4 predators	g C/individual
EW5	Actual weight of all group 5 predators	g C/individual
EW6	Actual weight of all group 6 predators	g C/individual
EW7	Actual weight of all group 7 predators	g C/individual
EX1	Point on x-axis to determine rate of transfer into new stage	Degree-days (°C)
EX2	Point on x-axis to determine rate of transfer into new stage	Degree-days (°C)
EY1	Point on y-axis to determine rate of transfer into new stage	Nondimensional
EY2	Point on y-axis to determine rate of transfer into new stage	Nondimensional

APPENDIX C (cont.)

Below are listed the variables only occurring in function EASN.

Variable name	Definition	Dimension
EASN	General arc sine function	Nondimensional
EA	Half amplitude of sine wave	Nondimensional
EB	Wavelength of sine wave	Nondimensional
EC	Parameter moving sine wave on x-axis	Nondimensional
EE	Distance from x-axis to half amplitude line	Nondimensional
EY	y-value of point whose x-value is to be found	Nondimensional

APPENDIX C (cont.)

Below are listed the variables only occurring in function EDGDS.

Variable name	Definition	Dimension
EDGDS	Function computing number of degree-days	Degree-days (°C)
E1	Area between developmental zero line and sine wave 1	Degree-days (°C)
E2	Area between developmental zero line and sine wave 2	Degree-days (°C)
EA	Half the temperature amplitude	°C
EB1	Wave length of sine wave 1	Days
EB2	Wave length of sine wave 2	Days
EC1	Parameter moving sine wave 1 along x-axis	Days
EC2	Parameter moving sine wave 2 along x-axis	Days
ED	Distance from x-axis to develop mental zero line	°C
EDZ	Developmental zero	°C
EE	Distance from x-axis to half amplitude line	°C
EMAX	Maximum daily temperature	°C
EMIN	Minimum daily temperature	°C
EX1	x-value for intersection between developmental zero line and sine wave 1	Days
EX2	x-value for intersection between developmental zero line and sine wave 2	Days

APPENDIX C (cont.)

Below are listed the variables only occurring in function EEPF.

Variable name	Definition	Dimension
EEPF	Function computing number of eggs laid per female	Individuals/female/day
EA	Constant, equals E12	Degree-days (°C)
EDD	Degree-days experienced by insect per day	Degree-days/day
EDV	Relative difference between actual weight and minimum weight for reproduction	Nondimensional
EEWT	Minimum weight for reproduction	g C/individual
EFNUT	Effect of nutrition on egg production	Nondimensional
EFTMP	Effect of temperature on egg production	Nondimensional
EMX	Maximum number of eggs laid per female	Individuals/female/day
EWT	Actual average weight of females	g C/individual

APPENDIX C (cont.)

Below are listed the variables only occurring in function EFOD.

Variable name	Definition	Dimension
EFOD	Total amount of food eaten by insect group (I,J)	g C/m^2
ECFD	Energy demand of insect group (I,J)	g C/m^2
EDGDD	Degree-days experienced by insect group (I,J)	Degree-days/day
EFCN	Food consumed by insect group (I,J) from food L	g C/m^2
ERAT	Ratio of actual to normal average weight	Nondimensional
ERPR	Energy cost of reproduction for group (I,J)	g C/m^2
ESEL	Selectivity index for consumer	Nondimensional

APPENDIX C (cont.)

Below are listed the variables which only occur in function EPRED.

Variable name	Definition	Dimension
EPRED	Number of prey (I,J) taken by predator K	Individual/m^2
ECFD	Consumer food demand	g C/m^2
EDGDD	Degree-days experienced by predator K	Degree-days/day
EFCN	Total consumption of prey (I,J) by predator K	g C/m^2
ENWT	Total expected weight of predator group K	g C/m^2
ERAT	Ratio of EWT to ENWT	Nondimensional
ERPR	Total reproductive cost of predator group K	g C/m^2
ERSP	Total respiration of predator group K	g C/m^2
ESEL	Selectivity of predator	Nondimensional
EWT	Total weight of predator group K	g C/m^2

APPENDIX C (cont.)

Below are listed the variables which only occur in function ERSPR.

Variable name	Definition	Dimension
ERSPR	Respiration of group (I,J) (basal metabolism)	g C/m^2
EEA	Half temperature amplitude	°C
EEB	Wave length of sine wave in temperature simulation	Days
EEC	Parameter moving sine wave along x-axis	Days
EEE	Distance from x-axis to half amplitude line	°C
EEX	Point on x-axis where ETEMP is to be computed	Days
ERESP	Respiration of individual at ETEMP	µl O_2/hour
ERSP	Summing variable for respiration	µl O_2/12 hours
ETEMP	Temperature generated from maximum and minimum temperatures by sine curves	°C
ETMN	Daily minimum temperature	°C
ETMX	Daily maximum temperature	°C

APPENDIX C (cont.)

Below are listed the variables only occurring in subroutine ESTEP.

Variable name	Definition	Dimension
ESTEP	Subroutine name, computes transfer to next stage	
EA	Return parameter (0 if no transfer, 1 if full transfer)	Nondimensional
EB	Return parameter, rest of accumulated degree-days	Degree-days (°C)
EDGG	Required accumulated degree-days for transfer	Degree-days (°C)
ESM	Accumulated degree-days	Degree-days (°C)

APPENDIX D

CATEGORIES OF HERBIVORES AND PREDATORS

Number	Category of Herbivore
1	Carabidae, prediapause adult [(14,1)]
2	Carabidae, postdiapause adult [(16,1)]
3	Chrysomelidae-Curculionidae larvae [(7,3)-(9,3)]
4	Chrysomelidae-Curculionidae adults [(14,3),(16,3)]
5	Hemiptera-Homoptera [(2,4)-(6,4),(12,4)-(16,4)]
6	Scarabaeidae larvae [(5,5),(7,5),(9,5),(11,5)]
7	Scarabaeidae adults [(16,5)]
8	Tenebrionidae larvae [(5,6),(7,6),(9,6)]
9	Orthoptera [(2,7)-(16,7)]

The variables typically have the form Var(I,J) where I refers to the food category and J to the herbivore category.

APPENDIX D (cont.)

Number	Category of Predator
1	Asilidae larvae
2	Asilidae adults
3	Carabidae larvae [(5,1),(7,1)]
4	Aranea [(7,2)-(9,2),(11,2)-(16,2)]
5	Formicidae
6	Birds
7	Small mammals [X(303)-X(305)]

The variables typically have the form Var K(I,J) where K refers to the predator group and (I,J) refers to the insect category that is preyed upon.

APPENDIX E

COMPUTER LISTING

Note:

The statements numbered ERIC 16 and 17 pertain to input variables from other submodels of ELM. Those numbered ERIC 18, 19, and 20 pertain to either temporary variables used to read input variables from a file or variables defined in other submodels used for interaction with these submodels.

The comment statements used to define variables are not entirely up to date, the reader should refer to Appendix C. The statements ERIC 448 to ERIC 471 were used to execute the model at 2-day time steps from input listed on file at 1-day time steps.

The list of parameters used is not entirely up to date and part of it is redundant.

```
C***********************************************************************ERIC
C                                                                       ERIC
C***********************************************************************ERIC
STORAGE.EN(16. 7)                                                       ERIC
STORAGE.ESUM(16. 7),EDG(16, 7),EDZ1,EDZ2                                ERIC
STORAGE.ENFIJ(17. 7).ENFL(16. 7).EFFI(16. 7).EFIE(16, 7),EFIR(16, 7)    ERIC
STORAGE.EW(16.7)                                                        ERIC
STORAGE.ETM1(16. 7).ETIME(16, 7).ENW(16, 7).ENW1(16, 7).ENW2(16, 7)     ERIC
STORAGE.ENFP(16.7.7).ESMP(16.7).ERPU(16,7).EPPRF(16,7,7),ERRNK(16,7,7)  ERIC
STORAGE.EPRD1,EPRD2.EPRD5.EPRD6.EDU1.EDU6,EDU7                          ERIC
STORAGE.EPAD1.EPAD2.EPAD3.EPAD5,EPAD6                                   ERIC
STORAGE.ET11.ET12.ET21.ET22,ET31.ET32.ET41,ET51,ET52,ET53,ET61,ET62,ET63ERIC
STORAGE.EMN1.EMN2,ET54.ET64                                             ERIC
STORAGE.ETIM1(16. 7).ETIM2(16. 7),EDW,EPPF( 7),EMER( 7),EMW( 7)         ERIC
STORAGE.EDAY,APEVA.ATEN(4),ARAIN,AVMX,AVMN,AVSTM(3),PGROS(5),PRESP(5)   ERIC
STORAGE.PHEN(5)                                                         ERIC
STORAGE.EDA1,EDA2.APEV1,APEV2.ATE1(4).ATE2(4),ARAI1,ARAI2,AVN1,AVN2,AVX1ERIC
STORAGE.AVX2.AVST1(3).AVST2(3),PGRO1(5),PGRO2(5),PRES1(5),PRES2(5)      ERIC
STORAGE.PHE1(5).PHE2(5),Z1(28),Z2(28),DLABF.DLABG,DLABI,DLASE,DLART(1)  ERIC
STORAGE.EFD(20).ERFD(20),ERPF(20, 9).EPRF(20, 9),ESMPF( 9),ESMRK( 9)    ERIC
STORAGE.ERNK(20.9).EFC(20,9),ECUT(7).EASK(7)                            ERIC
STORAGE.EACF.EIAF.EQ10,ERA.ERH,ERT1.ERT2.ERT3,ERT4,ERT5,E12             ERIC
INTEGER.EI.EJ.EK.EL.EM(9).EFLG                                          ERIC
C***DEFINITION OF VARIABLES ********************************************ERIC
C***  NOTE. NUMBER OF INSECTS REFERS TO NUMBER OF INDIVIDUALS PER SQUAR ERIC
C***           METER UNLESS OTHERWISE SPECIFICALLY STATED ***************ERIC
C***VARIABLES USED IN SUBROUTINE CYCL1 *********************************ERIC
C***  EA           - RETURN VARIABEL FROM SUBROUTINE ESTEP (VALUES 1 OR  ERIC
C***  EACF         - FACTOR ADJUSTING RESPIRATION FOR ACTIVE INSECTS *** ERIC
C***  EASK(I)      - ASSIMILATION COEFFICIENT FOR INSECT GROUP I, AS FRA ERIC
C***                 TOTAL FOOD INGESTED ******************************* ERIC
C***  EAVTM        - AVERAGE TEMPERATURE AT 7.5CM DEPTH **************** ERIC
C***  EA0          - (AVMX-AVMN)/2.                                      ERIC
C***  EA1          - EDTM1/2.                                            ERIC
C***  EA2          - EDTM2/2.                                            ERIC
C***  EA3          - EDTM3/2.                                            ERIC
C***  EB           - RETURN VARIABEL FROM SUBROUTINE ESTEP, DEGREE-DAYS  ERIC
C***  ECUT(I)      - FRACTION OF FOOD EATEN THAT IS SIMULTANEOUSLY WASTE ERIC
C***                 INDIVIDUAL OF GROUP I ( IN GRAMS CARBON ) ********* ERIC
C***  EDG(I.J)     - NUMBER OF DEGREE-DAYS PER DAY NECESSARY FOR DEVELOP ERIC
C***                 INTO STAGE I+1.J ********************************** ERIC
C***  EDGDD        - DUMMY VARIABEL (EQUAL TO EDGD1 OR EDGD2) ********** ERIC
C***  EDGDS        - FUNCTION TO DETERMINE DEGREE-DAYS FROM MAX AND MIN  ERIC
C***                 AND DEVELOPEMENTAL ZERO *************************** ERIC
C***  EDGD1        - NUMBER OF DEGREE-DAYS EXPERIENCED BY INSECTS AT SUR ERIC
C***  EDGD2        - NUMBER OF DEGREE-DAYS EXPERIENCED BY INSECTS AT 7.5 ERIC
C***  EDTM1        - DIFFERENCE BETWEEN MAX AND MIN TEMPERATURE AT SURFA ERIC
C***  EDTM2        - DIFFERENCE BETWEEN MAX AND MIN TEMP. AT 7.5CM DEPTH ERIC
C***  EDTM3        - DIFFERENCE BETWEEN MAX AND MIN TEMP. AT 10CM DEPTH  ERIC
C***  EDU1         - ENERGY DEMAND PER UNIT BIOMASS FOR INSECT PREDATORS ERIC
C***  EDU6         - ENERGY DEMAND PER UNIT BIOMASS (IN G CB) FOR BIRDS  ERIC
C***  EDU7         - ENERGY DEMAND PER UNIT BIOMASS FOR SMALL MAMMALS ** ERIC
C***  EDW          - FRACTION OF WEIGHT BELOW WHICH WEIGHT AFFECTS SURVI ERIC
C***  EDZ1         - INPUT PARAMETER, DEVELOPMENTAL ZERO FOR SURFACE INS ERIC
C***  EDZ2         - INPUT PARAMETER, DEVELOPMENTAL ZERO FOR INSECTS AT  ERIC
C***  EEPF         - FUNCTION TO COMPUTE NUMBER OF EGGS LAID PER FEMALE  ERIC
C***                 PARAMETERS- EMER(1),E12.EDGD(1 OR 2),EW(16,I),EMW(I ERIC
C***  EFC(L.K)     - CONSUMPTION OF FOOD L BY CONSUMER GROUP K (G CB) ** ERIC
C***  EFD(L)       - FOOD DENSITY AS GRAMS PLANT CARBON PER SQUARE METER ERIC
C***                 FOOD CATEGORY L *********************************** ERIC
C***  EFFI(I.J)    - FOOD INTAKE BY INSECT GROUP (I,J) IN GRAMS CARBON * ERIC
```

```
C***  EFIE(I.J)    - EXCRETION BY GROUP (I.J) IN GRAMS CARBON ********** ERIC
C***  EFIR(I.J)    - RESPIRATION BY GROUP (I.J) IN GRAMS CARBON ******** ERIC
C***  EFLG         - FLAG. RETURN PARAMETER IN FUNCTION ALINT2 ********* ERIC
C***  EFOOD        - SUMMING VARIABEL FOR FOOD EATEN BY PREDATORS ****** ERIC
C***  EGGP         - NUMBER OF EGGS PRODUCED PER FEMALE PER TIME STEP ** ERIC
C***  EHAF         - DUMMY VARIABEL USED IN TRANSFER FOR HEMI-HOMO AND G ERIC
C***  EI           - INTEGER. COUNTER ********************************** ERIC
C***  EIAF         - FACTOR ADJUSTING RESPIRATION FOR DIAPAUSING STAGES  ERIC
C***  EIW          - WEIGHT OF INDIVIDUAL ENTERING A STAGE ************* ERIC
C***  EJ           - INTEGER. COUNTER ********************************** ERIC
C***  EK           - INTEGER, COUNTER ********************************** ERIC
C***  EL           - INTEGER. COUNTER ********************************** ERIC
C***  EMER(I)      - MAXIMUM EGG-LAYING RATE PER FEMALE OF GROUP I PER T ERIC
C***  EMN1         - MINIMUM FRACTION SURVIVING EFFECT OF TEMPERATURE ** ERIC
C***  EMN2         - MINIMUM FRACTION SURVIVNG EFFECT OF MOISTURE OR EVA ERIC
C***  EMW(I)       - MINIMUM WEIGHT FOR REPRODUCTION BY FEMALE OF GROUP  ERIC
C***  EN(I.J)      - NUMBER OF INDIVIDUALS IN INSECT GROUP J AND CATEGOR ERIC
C***  ENEG         - NUMBERS GOING OUT FROM EN(I.J) ******************** ERIC
C***  ENFIJ(I.J)   - NUMBERS PER SQ.M FLOWING FROM EN(I-1,J) TO EN(I.J)  ERIC
C***  ENFL(I.J)    - NUMBER OF INSECTS IN GROUP J, CATEGORY I DYING FROM ERIC
C***                 PREDATORY CAUSES ********************************** ERIC
C***  ENFP(I.J.K)  - NUMBER OF PREY FROM GROUP I.J TAKEN BY PREDATOR K * ERIC
C***  ENW(I.J)     - EXPECTED WEIGHT OF AN INDIVIDUAL IN I,J *********** ERIC
C***  ENW1(I.J)    - EXPECTED WEIGHT AT START OF EXISTENCE OF INDIVIDUAL ERIC
C***  ENW2(I.J)    - EXPECTED WEIGHT AT END OF EXISTENCE OF INDIVIDUAL * ERIC
C***  EPAD1        - PROPORTION ARTHROPODS IN DIET FOR ONYCHOMYS LEUCOGA ERIC
C***  EPAD2        - PROPORTION ARTHROPODS IN DIET FOR PEROMYSCUS MANICU ERIC
C***  EPAD3        - PROPORTION ARTHROPODS IN DIET FOR SPERMOPHILUS TRID ERIC
C***  EPAD5        - PROPORTION ARTHROPODS IN DIET FOR ANTS ************ ERIC
C***  EPAD6        - PROPORTION ARTHROPODS IN DIET FOR BIRDS *********** ERIC
C***  EPPF(I)      - PROPORTION OF PRODUCTIVE FEMALES OF ADULT POPULATIO ERIC
C***                 INSECT GROUP I ************************************ ERIC
C***  EPPRF(I.J.K)- FOOD PREFERENCE OF PREDATOR K FOR PREY I,J ******** ERIC
C***  EPRD1        - TOTAL BIOMASS OF PREDATOR 1 IN GRAMS CARBON PER SQ. ERIC
C***  EPRD2        - TOTAL BIOMASS OF PREDATOR 2 IN GRAMS CARBON PER SQ. ERIC
C***  EPRD5        - TOTAL BIOMASS OF PREDATOR 5 IN GRAMS CARBON PER SQ. ERIC
C***  EPRD6        - TOTAL BIOMASS OF PREDATOR 6 IN GRAMS CARBON PER SQ. ERIC
C***  EPRF(L.K)    - ASSIGNED PREFERENCE INDEX FOR FOOD L AND CONSUMER K ERIC
C***  ERA          - CONSTANT USED IN FUNCTION ERSPR ******************* ERIC
C***  ERB          - CONSTANT USED IN FUNCTION ERSPR ******************* ERIC
C***  EREP         - ENERGY COST OF EGG PRODUCTION FOR REPRODUCTIVE STAG ERIC
C***  EREP1        - SAME AS EREP BUT FOR HEMI-HOMOS FIRST GENERATION ** ERIC
C***  ERFD(L)      - RELATIVE FOOD DENSITY FOR FOOD CATEGORY L. NORMALIZ ERIC
C***  ERNK(L.K)    - RELATIVE RANK OF FOOD L FOR CONSUMER K , NORMALIZED ERIC
C***  ERPD(I.J)    - RELATIVE DENSITY,OF PREY I.J ********************** ERIC
C***  ERPF(L.K)    - RELATIVE PREFERENCE INDEX FOR FOOD L AND CONSUMER K ERIC
C***  ERPR         - DUMMY VARIABEL (EQUAL TO EREP OR 0.) ************** ERIC
C***  ERPNK(I.J.K)- RANK OF PREY I,J FOR PREDATOR K ******************* ERIC
C***  ERT1         - CONSTANT USED IN FUNCTION EFOD ******************** ERIC
C***  ERT2         - CONSTANT USED IN FUNCTION EFOD ******************** ERIC
C***  ERT3         - PARAMETER USED IN EPRED *************************** ERIC
C***  ESFD         - FRACTION SURVIVING LACK OF FOOD ******************* ERIC
C***  ESMF         - SUMMING VARIABLE. SUM OF ALL FOOD CATEGORIES ****** ERIC
C***  ESMP(I.J)    - TOTAL NUMBERS TAKEN BY PREDATORS FROM GROUP J, CATE ERIC
C***  ESMPF(K)     - SUM OF PREFERENCES FOR CONSUMER K ***************** ERIC
C***  ESMRK(K)     - SUM OF RANKS FOR CONSUMER K *********************** ERIC
C***  ESMST        - FRACTION SURVIVING MOISTURE CONDITIONS ************ ERIC
C***  ESPR1        - SUM OF WEIGHTS OF ALL POTENTIAL PREY ************** ERIC
C***  ESPR2        - SUM OF WEIGHTS OF ALL POTENTIAL PREY BELOW GROUND * ERIC
C***  ESPR3        - SUM OF WEIGHTS OF ALL POTENTIAL PREY ABOVE GROUND * ERIC
C***  ESTMP        - FRACTION SURVIVING TEMPERATURE CONDITIONS ********* ERIC
C***  ESUM(I.J)    - NUMBER OF DEGREE-DAYS ACCUMULATED BY GROUP J. CATEG ERIC
C***  ETIME(I.J)   - LENGTH OF EXISTENCE OF INDIVIDUAL (ETM2(I,J)-ETM1(I ERIC
C***  ETIM1(I.J)   - TIME AFTER WHICH ENFIJ(I.J) FLOW CAN OCCUR ******** ERIC
C***  ETIM2(I.J)   - TIME UNTIL WHICH ENFIJ(I.J) FLOW CAN OCCUR ******** ERIC
C***  ETMN1        - MINIMUM TEMPERATURE AT THE SOIL SURFACE *********** ERIC
C***  ETMN2        - MINIMUM TEMPERATURE AT 7.5CM SOIL DEPTH *********** ERIC
C***  ETMN3        - MINIMUM TEMPERATURE AT 10CM SOIL DEPTH ************ ERIC
C***  ETMX1        - MAXIMUM TEMPERATURE AT THE SOIL SURFACE *********** ERIC
C***  ETMX2        - MAXIMUM TEMPERATURE AT 7.5CM SOIL DEPTH *********** ERIC
```

```
C***   ETMX3        - MAXIMUM TEMPERATURE AT 10CM SOIL DEPTH ************ ERIC
C***   ETM1(I,J)    - TIME FOR START OF EXISTENCE OF GROUP J, CATEGORY I  ERIC
C***   ETM2         - TIME FOR END OF EXISTENCE OF GROUP **************** ERIC
C***   ET11         - FIRST POINT ON X-AXIS TO DETERMINE TEMPERATURE EFFE ERIC
C***                  SURVIVAL FOR EGGS, PUPAE, AND DIAPAUSING INDIVIDUAL ERIC
C***   ET12         - FIRST POINT ON X-AXIS TO DETERMINE EFFECT OF TEMPER ERIC
C***                  SURVIVAL FOR ACTIVE LARVAE AND ADULTS ************* ERIC
C***   ET21         - SECOND POINT ON X-AXIS TO DETERMINE TEMPERATURE EFF ERIC
C***                  SURVIVAL OF EGGS. PUPAE. AND DIAPAUSING INDIVIDUALS ERIC
C***   ET22         - SECOND POINT ON X-AXIS TO DETERMINE TEMPERATURE EFF ERIC
C***                  SURVIVAL OF ACTIVE LARVAE AND ADULTS ************** ERIC
C***   ET31         - THIRD POINT ON X-AXIS TO DETERMINE TEMPERATURE EFFE ERIC
C***                  SURVIVAL OF EGGS. PUPAE. AND DIAPAUSING INDIVIDUALS ERIC
C***   ET32         - THIRD POINT ON X-AXIS TO DETERMINE TEMPERATURE EFFE ERIC
C***                  SURVIVAL OF ACTIVE LARVAE AND ADULTS ************** ERIC
C***   ET41         - FOURTH POINT ON X-AXIS TO DETERMINE TEMPERATURE EFF ERIC
C***                  ON SURVIVAL *************************************** ERIC
C***   ET51         - FIRST POINT ON X-AXIS TO DETERMINE EFFECT OF SOIL M ERIC
C***                  ON SURVIVAL OF EGGS, PUPAE, AND DIAPAUSING INDIVIDU ERIC
C***   ET52         - FIRST POINT ON X-AXIS TO DETERMINE EFECT OF SOIL MO ERIC
C***                  ON SURVIVAL OF ACTIVE LARVAE AND ADULTS *********** ERIC
C***   ET53         - FIRST POINT ON X-AXIS TO DETERMINE EFFECT OF POTENT ERIC
C***                  EVAPOTRANSPIRATION ON SURVIVAL OF ACTIVE LARVAE AND ERIC
C***   ET54         - FIRST POINT ON X-AXIS TO DETERMINE EFFECT OF POTENT ERIC
C***                  EVAPOTRANSPIRATION ON SURVIVAL OF EGGS, PUPAE,AND D ERIC
C***   ET61         - SECOND POINT ON X-AXIS TO DETERMINE EFFECT OF SOIL  ERIC
C***                  ON SURVIVAL OF EGGS, PUPAE, AND DIAPAUSING INDIVIDU ERIC
C***   ET62         - SECOND POINT ON X-AXIS TO DETERMINE EFFECT OF SOIL  ERIC
C***                  ON SURVIVAL OF ACTIVE LARVAE AND ADULTS *********** ERIC
C***   ET63         - SECOND POINT ON X-AXIS TO DETERMINE EFFECT OF POTEN ERIC
C***                  EVAPOTRANSPIRATION ON SURVIVAL OF ACTIVE LARVAE AND ERIC
C***   ET64         - SECOND POINT ON X-AXIS TO DETERMINE EFFECT OF POTEN ERIC
C***                  EVAPOTRANSPIRATION ON SURVIVAL OF EGGS. PUPAE AND D ERIC
C***   EW(I.J)      - WEIGHT IN GRAMS CARBON OF INDIVIDUAL IN GROUP J, CA ERIC
C***   EWT1         - NORMAL WEIGHT OF ALL GROUP 1 PREDATORS ************ ERIC
C***   EWT2         - NORMAL WEIGHT OF ALL GROUP 2 PREDATORS ************ ERIC
C***   EWT3         - NORMAL WEIGHT OF ALL GROUP 3 PREDATORS ************ ERIC
C***   EWT4         - NORMAL WEIGHT OF ALL GROUP 4 PREDATORS ************ ERIC
C***   EWT5         - NORMAL WEIGHT OF ALL GROUP 5 PREDATORS ************ ERIC
C***   EWT6         - NORMAL WEIGHT OF ALL GROUP 6 PREDATORS ************ ERIC
C***   EWT7         - NORMAL WEIGHT OF ALL GROUP 7 PREDATORS ************ ERIC
C***   EW1          - ACTUAL WEIGHT OF ALL GROUP 1 PREDATORS ************ ERIC
C***   EW2          - ACTUAL WEIGHT OF ALL GROUP 2 PREDATORS ************ ERIC
C***   EW3          - ACTUAL WEIGHT OF ALL GROUP 3 PREDATORS ************ ERIC
C***   EW4          - ACTUAL WEIGHT OF ALL GROUP 4 PREDATORS ************ ERIC
C***   EW5          - ACTUAL WEIGHT OF ALL GROUP 5 PREDATORS ************ ERIC
C***   EW6          - ACTUAL WEIGHT OF ALL GROUP 6 PREDATORS ************ ERIC
C***   EW7          - ACTUAL WEIGHT OF ALL GROUP 7 PREDATORS ************ ERIC
C***   EX1          - FIRST POINT ON X-AXIS IN ALINT2 TRANSFER FUNCTION * ERIC
C***   EX2          - SECOND POINT ON X-AXIS IN ALINT2 TRANSFER FUNCTION  ERIC
C***   EY1          - FIRST POINT ON Y-AXIS IN ALINT2 TRANSFER FUNCTION * ERIC
C***   EY2          - SECOND POINT ON Y-AXIS IN ALINT2 TRANSFER FUNCTION  ERIC
C***   E12          - CONSTANT . DEGREE-DAYS. USED IN RELATION TO EDGD ** ERIC
C***   E22          - VARIABLE EXPRESSING TEMPERATURE EFFECT ON ESEF(I,J) ERIC
C***BELOW ARE LISTED THE VARIABLES WHICH ONLY OCCUR IN SUBPROGRAMS ***** ERIC
C***VARIABLES IN FUNCTION EASN ***************************************** ERIC
C***EASN GIVES THE X-VALUE FOR A POINT ON A SINE WAVE WHEN Y IS GIVEN ** ERIC
C***   EA           - FORMAL PARAMETER, HALF AMPLITUDE OF SINE WAVE ***** ERIC
C***   EB           - FORMAL PARAMETER. WAVE LENGTH OF SINE WAVE ******** ERIC
C***   EC           - FORMAL PARAMETER. PARAMETER FOR MOVING WAVE ALONG X ERIC
C***   EE           - FORMAL PARAMETER. DISTANCE FROM X-AXIS TO HALF AMPL ERIC
C***   EY           - FORMAL PARAMETER, Y-VALUE OF POINT ON SINE WAVE *** ERIC
C***VARIABLES IN FUNCTION EDGDS **************************************** ERIC
C***   EA           - HALF THE TEMPERATURE AMPLITUDE ******************** ERIC
C***   EB1          - WAVE LENGTH OF SINE WAVE 1 ************************ ERIC
C***   EB2          - WAVE LENGTH OF SINE WAVE 2 ************************ ERIC
C***   EC1          - PARAMETER MOVING SINE WAVE 1 ALONG X-AXIS ********* ERIC
```

```
C***   EC2         - PARAMETER MOVING SINE WAVE 2 ALONG X-AXIS ********* ERIC
C***   ED          - DISTANCE FROM X-AXIS TO DEVELOPMENTAL ZERO LINE *** ERIC
C***   EDZ         - FORMAL PARAMETER, DEVELOPMENTAL ZERO ************** ERIC
C***   EE          - DISTANCE FROM X-AXIS TO HALF AMPLITUDE LINE ******* ERIC
C***   EMAX        - FORMAL PARAMETER, MAXIMUM TEMPERATURE DURING THE DA ERIC
C***   EMIN        - FORMAL PARAMETER, MINIMUM TEMPERATURE DURING THE DA ERIC
C***   EX1         - X-VALUE FOR INTERSECTION DEVELOP ZERO AND SINE WAVE ERIC
C***   EX2         - X-VALUE FOR INTERSECTION DEVELOP ZERO AND SINE WAVE ERIC
C***   E1          - AREA BETWEEN DEVELOPMENTAL ZERO LINE AND SINE WAVE  ERIC
C***   E2          - AREA BETWEEN DEVELOPMENTAL ZERO LINE AND SINE WAVE  ERIC
C***VARIABLES IN FUNCTION EEPF ****************************************** ERIC
C***EEPF COMPUTES THE NUMBER OF EGGS PRODUCED PER FEMALE PER DAY ******* ERIC
C***   EA          - FORMAL PARAMETER, EQUALS E12 *********************** ERIC
C***   EDD         - FORMAL PARAMETER, DEGREE-DAYS PER DAY ************** ERIC
C***   EDV         - RELATIVE DIFFERENCE BETWEEN ACTUAL AND MINIMUM WEIG ERIC
C***   EEWT        - FORMAL PARAMETER, MINIMUM WEIGHT FOR REPRODUCTION * ERIC
C***   EFNUT       - EFFECT OF NUTRITION ON EGG PRODUCTION ************* ERIC
C***   EFTMP       - EFFECT OF TEMPERATURE ON EGG PRODUCTION *********** ERIC
C***   EMX         - FORMAL PARAMETER, MAXIMUM NUMBER OF EGGS /FEMALE/DA ERIC
C***   EWT         - FORMAL PARAMETER, ACTUAL AVERAGE WEIGHT *********** ERIC
C***VARIABLES IN FUNCTION EFOD ****************************************** ERIC
C***EFOD GIVES THE AMOUNT EATEN BY HERBIVORE GROUP (I,J) (IN GRAMS CARBO ERIC
C***   ECFD        - FOOD DEMAND OF CONSUMER (IN GRAMS CARBON) ********* ERIC
C***   EDGDD       - FORMAL PARAMETER, DEGREE-DAYS PER DAY ************** ERIC
C***   EFCN        - FOOD CONSUMED BY INSECT GROUP (I,J) FROM FOOD GROUP ERIC
C***   ERAT        - RATIO OF ACTUAL TO NORMAL WEIGHT ****************** ERIC
C***   ERPR        - FORMAL PARAMETER, ENERGY COST OF REPRODUCTION ***** ERIC
C***   ESEL        - SELECTIVITY INDEX ********************************** ERIC
C***VARIABLES IN FUNCTION EPRED ***************************************** ERIC
C***EPRED COMPUTES THE NUMBER OF INSECTS IN GROUP (I,J) TAKEN BY PREDATO ERIC
C***   ECFD        - ENERGY DEMAND OF PREDATOR IN GRAMS CARBON ********* ERIC
C***   EDGDD       - DEGREE-DAYS PER DAY ******************************** ERIC
C***   EFCN        - FOOD CONSUMPTION IN GRAMS CARBON ****************** ERIC
C***   ENWT        - FORMAL PARAMETER, NORMAL WEIGHT OF PREDATORS ****** ERIC
C***   ERSP        - FORMAL PARAMETER, RESPIRATION OF PREDATORS ******** ERIC
C***   EWT         - FORMAL PARAMETER, ACTUAL WEIGHT OF PREDATORS ****** ERIC
C***   ERAT        - RATIO OF ACTAUL TO NORMAL WEIGHT ****************** ERIC
C***   ERPR        - ENERGY COST OF REPRODUCTION ********************** ERIC
C***   ESEL        - SELECTIVITY INDEX ********************************** ERIC
C***VARIABLES IN FUNCTION ERSPR ***************************************** ERIC
C***ERSPR COMPUTES THE RESPIRATION OF GROUP(I,J) AS FUNCTION OF WEIGHT , ERIC
C***   EEA         - HALF THE TEMPERATURE AMPLITUDE ******************** ERIC
C***   EEB         - WAVE LENGTH OF SINE WAVE ************************** ERIC
C***   EEC         - PARAMETER MOVING SINE WAVE ON X-AXIS ************** ERIC
C***   EEE         - DISTANCE FROM X-AXIS TO HALF AMPLITUDE LINE ******* ERIC
C***   EEX         - POINT ON X-AXIS WHERE ETEMP IS TO BE COMPUTED ***** ERIC
C***   EQ10        - PARAMETER REPRESENTING Q10 VALUE ****************** ERIC
C***   ERA         - PARAMETER TO COMPUTE RESPIRATION AS FUNCTION OF WEI ERIC
C***   ERB         - PARAMETER TO COMPUTE RESPIRATION AS FUNCTION OF WEI ERIC
C***   ERESP       - RESPIRATION OF INDIVIDUAL AT ETEMP **************** ERIC
C***   ERSP        - SUMMING VARIABEL FOR RESPIRATION ****************** ERIC
C***   ETEMP       - TEMPERATURE GENERATED FROM MAX AND MIN TEMPERATURES ERIC
C***   ETMN        - FORMAL PARAMETER, MINIMUM TEMPERATURE OF THE DAY ** ERIC
C***   ETMX        - FORMAL PARAMETER, MAXIMUM TEMPERATURE OF THE DAY ** ERIC
C***VARIABLES IN SUBROUTINE ESTEP *************************************** ERIC
C***ESTEP DETERMINES WHETHER ALL OR NOTHING IS FLOWED IN TRANSFER ****** ERIC
C***   EA          - RETURN PARAMETER, (0 IF NO FLOW, 1 IF FULL FLOW) ** ERIC
C***   EB          - RETURN PARAMETER, REST OF ACCUMULATED DEGREE-DAYS * ERIC
C***   EDGG        - FORMAL PARAMETER, REQUIRED ACCUMULATED DEGREE-DAYS* ERIC
C***   ESM         - FORMAL PARAMETER, ACCUMULATED DEGREE-DAYS ********* ERIC
C********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
```

```
      FUNCTION EDGDS(EMAX,EMIN,EDZ)                                     ERIC
C---THIS FUNCTION COMPUTES THE NUMBER OF DEGREE-DAYS PER DAY AS A FUNCTI ERIC
C---MAXIMUM AND MINIMUM TEMPERATURE AND THE DEVELOPEMENTAL ZERO ******** ERIC
      EDGDS=0.                                                          ERIC
      IF(EDZ.GE.EMAX) RETURN                                            ERIC
      EA=(EMAX-EMIN)/2.                                                 ERIC
      EB1=2./3.                                                         ERIC
      EB2=4./3.                                                         ERIC
      EC1=1./6.                                                         ERIC
      EC2=0.                                                            ERIC
      EX1=0.                                                            ERIC
      EX2=1.                                                            ERIC
      IF(EDZ.LE.EMIN) GO TO 1                                           ERIC
      EE=EA+EMIN                                                        ERIC
      EX1=EASN(EDZ,EA,EB1,EC1,EE)                                       ERIC
      EX2=2./3.-EASN(EDZ,EA,EB2,EC2,EE)                                 ERIC
    1 CONTINUE                                                          ERIC
      ED=EA+EMIN-EDZ                                                    ERIC
      E1=EA*EB1*(COS(6.2832*(EX1-EC1)/EB1)-COS(6.2832*(1./3.-EC1)/EB1))/ ERIC
     /6.2832+ED*(1./3.-EX1)                                             ERIC
      E2=EA*EB2*(COS(6.2832*(1./3.-EC2)/EB2)-COS(6.2832*(EX2-EC2)/EB2))/ ERIC
     /6.2832+ED*(EX2-1./3.)                                             ERIC
      EDGDS=E1+E2                                                       ERIC
      RETURN                                                            ERIC
      END                                                               ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      FUNCTION EASN(EY,EA,EB,EC,EE)                                     ERIC
C---THIS FUNCTION GIVES THE X-VALUE FOR A SINE FUNCTION WHEN Y IS INPUT ERIC
      EASN=EB*ASIN((EY-EE)/EA)/6.2832+EC                                ERIC
      RETURN                                                            ERIC
      END                                                               ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      SUBROUTINE ESTEP(ESM,EDGG,EA,EB)                                  ERIC
C***THIS SUBROUTINE COMPUTES A STEP FUNCTION FOR USE IN CATEGORY TRANSFE ERIC
      EA=0.                                                             ERIC
      EB=ESM                                                            ERIC
      IF(ESM.LT.EDGG) RETURN                                            ERIC
      EA=1.                                                             ERIC
      EB=ESM-EDGG                                                       ERIC
      IF(EB.LT.0.001) EB=0.                                             ERIC
      RETURN                                                            ERIC
      END                                                               ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      FUNCTION EEPF(EMX,EA,EDD,EWT,EEWT)                                ERIC
C***THIS FUNCTION COMPUTES THE NUMBER OF EGGS PRODUCED PER FEMALE PER DA ERIC
      EEPF=0.                                                           ERIC
      IF(EDD.LE.0.) RETURN                                              ERIC
      EDV=(EWT-EEWT)/EWT                                                ERIC
      IF(EDV.LE.0.) RETURN                                              ERIC
      EFTMP=1.-1./EXP(2.99573*EDD*EDD/(EA*EA))                          ERIC
      EFNUT=ALINT2(EDV,EFLG,0.,0.,0.2,1.)                               ERIC
      EEPF=EMX*EFTMP*EFNUT                                              ERIC
      RETURN                                                            ERIC
      END                                                               ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      FUNCTION ERSPR(ETMX,ETMN)                                         ERIC
```

```
C***THIS FUNCTION COMPUTES THE RESPIRATION AS A FUNCTION OF AMBIENT TEMP ERIC
C***(FROM ETMX AND ETMN) AND WEIGHT (EW(EI,EJ)) ************************** ERIC
C***THE VALUE IS TRANSFORMED FROM MIKROLITER OXYGEN PER HOUR TO GRAMS CA ERIC
C***PER 24 HOURS BY THE FACTOR 0.00000067368 **************************** ERIC
      ERSP=0.                                                            ERIC
      EEA=(ETMX-ETMN)/2.                                                 ERIC
      EEE=EEA+ETMN                                                       ERIC
      DO 4034 EJ=1.12                                                    ERIC
      EEX=(EJ-1)*2.                                                      ERIC
      EEB=16.                                                            ERIC
      EEC=4.                                                             ERIC
      IF(EJ.LE.5) GO TO 4030                                             ERIC
      EEB=32.                                                            ERIC
      EEC=0.                                                             ERIC
 4030 CONTINUE                                                           ERIC
      ETEMP=EEA*SIN(6.2832*(EEX-EEC)/EEB)+EEE                            ERIC
      IF(ETEMP.NE.20.) GO TO 4031                                        ERIC
      ERESP=ERA*(10000.*EW(EI.EK))**ERB                                  ERIC
      GO TO 4033                                                         ERIC
 4031 CONTINUE                                                           ERIC
      IF(ETEMP.LT.20.) GO TO 4032                                        ERIC
      ERESP=ERA*(10000.*EW(EI.EK))**ERB*EXP(EQ10  *(ETEMP-20.))          ERIC
      GO TO 4033                                                         ERIC
 4032 CONTINUE                                                           ERIC
      ERESP=ERA*(10000.*EW(EI.EK))**ERB/EXP(EQ10  *(20.-ETEMP))          ERIC
      IF(ETEMP.LE.-2.)  ERESP=0.                                         ERIC
 4033 CONTINUE                                                           ERIC
      ERSP=ERSP+ERESP                                                    ERIC
 4034 CONTINUE                                                           ERIC
      ERSPR=ERSP*EN(EI.EK)*0.00000067368                                 ERIC
      RETURN                                                             ERIC
      END                                                                ERIC
C*********************************************************************** ERIC
C                                                                      * ERIC
C*********************************************************************** ERIC
      FUNCTION EFOD(ERPR.EDGDD)                                        * ERIC
C***THIS FUNCTION COMPUTES THE FOOD INTAKE BY HERBIVORES                  ERIC
      EFOD=0.                                              ************** ERIC
C***SELECTIVITY OF CONSUMER AS FUNCTION OF SATIATION                      ERIC
      ERAT=EW(EI.EK)/ENW(EI.EK)                       ******************* ERIC
      ESEL=0.5*SIN(3.14159*(ERAT-0.9)/0.2)+0.5                           ERIC
      IF(ERAT.GE.1.) ESEL=1.                                             ERIC
      IF(ERAT.LE.0.8) ESEL=0.                                            ERIC
C***FOOD DEMAND OF CONSUMER ECFD                                           ERIC
      ECFD=(1.3*ENW(EI.EK)-EW(EI.EK))*EN(EI.EK)+EFIR(EI,EK)+ERPR  ******* ERIC
      IF(ECFD.LE.0.) ECFD=0.                                             ERIC
C***TOTAL FOOD CONSUMPTION BY CONSUMER J FROM FOOD GROUP I                ERIC
      DO 4021 EL=1.20                                      ************* ERIC
      IF(ERNK(EL.EJ).EQ.0.) GO TO 4021                                   ERIC
      EFCN=(ESEL*ECFD*ERNK(EL.EJ)+(1.-ESEL)*ECFD*ERFD(EL))*(1.+ECUT(EK)) ERIC
      IF(EFCN.GT.ERT1*EFD(EL)) EFCN=EFCN*(0.5*SIN(3.14159*(EFCN/EFD(EL)+ ERIC
     *0.5*ERT4-1.5*ERT1)/(ERT4-ERT1))+0.5)                               ERIC
      IF(EFCN.GT.ERT4*EFD(EL)) EFCN=0.                                   ERIC
      IF(EDGDD.LT.ERT2*E12) EFCN=EFCN*(0.5*SIN(3.14159*(EDGDD-0.5*E12*ER ERIC
     RT2)/(ERT2*E12))+0.5)                                               ERIC
      EFOD=EFOD+EFCN/(1.+ECUT(EK))                                       ERIC
      EFC(EL.EJ)=EFC(EL.EJ)+EFCN                                         ERIC
 4021 CONTINUE                                                           ERIC
      RETURN                                                             ERIC
      END                                                                ERIC
C*********************************************************************** ERIC
C                                                                      * ERIC
C*********************************************************************** ERIC
      FUNCTION EPRED(EWT,ENWT,ERSP,ERPR,EDGDD)                         * ERIC
C***THIS FUNCTION COMPUTES FOOD DEMAND AND SELECTIVITY OF A PREDATOR GRO ERIC
C***RETURNS WITH THE NUMBER OF PREY TAKEN FROM A GROUP OF PREY ********* ERIC
      EPRED=0.                                                           ERIC
      IF(EPPRF(EI,EJ.EK).LE.0.) RETURN                                   ERIC
      IF(EWT.EQ.0.) RETURN                                               ERIC
C***ENERGY DEMAND OF PREDATOR                                              ERIC
      ECFD=1.3*ENWT-EWT+ERSP+ERPR           ***************************** ERIC
      IF(ECFD.LT.0.) ECFD=0.                                             ERIC
C***SELECTIVITY                                                           ERIC
                                            ***************************** ERIC
```

```
      ERAT=EWT/ENWT
      ESEL=0.5*SIN(3.14159*(ERAT-0.9)/0.2)+0.5
C***TOTAL FOOD CONSUMPTION FROM PREY (I,J) *****************************
      EFCN=ESEL*ECFD*ERRNK(EI.EJ.EK)+(1.-ESEL)*ECFD*ERPD(EI.EJ)
C***MODIFICATIONS OF EFCN BY PREY AND FEEDING ACTIVITY AVAILABLE *******
      IF(EDGDD.LT.ERT2*E12) EFCN=EFCN*(0.5*SIN(3.14159*(EDGDD-0.5*E12*ER
     RT2)/(ERT2*E12))+0.5)
C***NUMBER OF PREY TAKEN ***********************************************
      EPPED=EFCN/EW(EI.EJ)
      RETURN
      END
C***********************************************************************
C **********************************************************************
C***********************************************************************
      SUBROUTINE START
      READ(7) DAY
      READ(7) DAY
      DO 4 EJ=1.7
      DO 4 EI=1.16
      ESUM(EI.EJ)=0.
      ESMP(EI.EJ)=0.
      ENFL(EI.EJ)=0.
      EFIR(EI.EJ)=0.
      EFFI(EI.EJ)=0.
      EFIE(EI.EJ)=0.
      ENFIJ(EI.EJ)=0.
      ENFIJ(17.EJ)=0.
      EW(EI.EJ)=0.
      IF(EJ.LE.5) X(303+16*EJ+EI)=EN(EI,EJ)*ENW1(EI,EJ)
      IF(EJ.GT.5) X(503+16*(EJ-5)+EI)=EN(EI,EJ)*ENW1(EI,EJ)
    4 CONTINUE
      EW(6.1)=ENW1(6.1)
      EW(15.1)=ENW1(15.1)
      EW(10.2)=ENW1(10.2)
      EW(15.3)=ENW1(15.3)
      EW(1.4)=ENW1(1.4)
      EW(6.5)=ENW1(6.5)
      EW(8.5)=ENW1(8.5)
      EW(10.5)=ENW1(10.5)
      EW(6.6)=ENW1(6.6)
      EW(8.6)=ENW1(8.6)
      EW(1.7)=ENW1(1.7)
      RETURN
      END
C***********************************************************************
C **********************************************************************
C***********************************************************************
      SUBROUTINE CYCL1
      READ(7) EDA1,APEV1.(ATE1(J),J=1,4),ARAI1,AVN1,AVX1,(AVST1(J),J=1,3
     )).(PGRO1(J),J=1,5),(PRES1(J),J=1,5),(PHE1(J).J=1,5),(Z1(J),J=1,28)
      READ(7) EDA2,APEV2.(ATE2(J).J=1.4),ARAI2.AVN2.AVX2,(AVST2(J),J=1,3
     )).(PGRO2(J).J=1.5).(PRES2(J),J=1,5),(PHE2(J),J=1,5),(Z2(J),J=1,28)
      EDAY=(EDA1+EDA2)/2.
      APEVA=(APEV1+APEV2)/2.
      DO 50 I=1.4
      ATEN(I)=(ATE1(I)+ATE2(I))/2.
   50 CONTINUE
      ARAIN=(ARAI1+ARAI2)/2.
      AVMN=(AVN1+AVN2)/2.
      AVMX=(AVX1+AVX2)/2.
      DO 51 I=1.3
      X(302+I)=(Z1(25+I)+Z2(25+I))/2.
      AVSTM(I)=(AVST1(I)+AVST2(I))/2.
   51 CONTINUE
      DO 52 I=1.5
      PHEN(I)=(PHE1(I)+PHE2(I))/2.
      X(199+I)=(Z1(I)+Z2(I))/2.
      X(209+I)=(Z1(5+I)+Z2(5+I))/2.
      X(219+I)=(Z1(10+I)+Z2(10+I))/2.
      X(229+I)=(Z1(15+I)+Z2(15+I))/2.
```

```
        X(239+I)=(Z1(20+I)+Z2(20+I))/2.
   52 CONTINUE                                                          ERIC
C***THIS PART GIVES THE APPROPRIATE TEMPERATURES FROM ABIOTIC INPUT **** ERIC
C***INPUT=MAX AND MIN TEMPERATURES AT 200 CM ABOVE GROUND AND AVERAGE   ERIC
C***TEMPERATURES AT DEPTHS OF 0, 15, AND 30 CMS BELOW GROUND *********** ERIC
      EDTM1=(AVMX-AVMN)*1.5                                             ERIC
      EDTM2=EDTM1*0.527515                                              ERIC
      EDTM3=EDTM1*0.278272                                              ERIC
      EA0=(AVMX-AVMN)/2.                                                ERIC
      EA1=EDTM1/2.                                                      ERIC
      EA2=EDTM2/2.                                                      ERIC
      EA3=EDTM3/2.                                                      ERIC
      EAVTM=0.3*AVSTM(1)+0.7*AVSTM(2)                                   ERIC
      ETMN1=AVSTM(1)-EA1                                                ERIC
      ETMX1=AVSTM(1)+EA1                                                ERIC
      ETMN2=EAVTM-EA2                                                   ERIC
      ETMX2=EAVTM+EA2                                                   ERIC
      ETMN3=AVSTM(2)-EA3                                                ERIC
      ETMX3=AVSTM(2)+EA3                                                ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
C***THIS PART TAKES CARE OF COMPUTATIONS THAT ARE COMMON FOR ALL GROUPS  ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
      EGGP=0.                                                           ERIC
      DO 420 EJ=1,9                                                     ERIC
      DO 420 EI=1,20                                                    ERIC
      EFC(EI,EJ)=0.                                                     ERIC
  420 CONTINUE                                                          ERIC
C***THIS PART COMPUTES THE FOOD PREFERENCES OF THE PLANT FEEDERS ******* ERIC
C***ABSOLUTE FOOD DENSITY EFD(I) AND RELATIVE FOOD DENSITY ERFD(I) ***** ERIC
      DO 4011 EK=1,5                                                    ERIC
      EFD(EK)=X(199+EK)                                                 ERIC
      EFD(EK+5)=X(209+EK)                                               ERIC
      EFD(EK+10)=X(229+EK)                                              ERIC
      EFD(EK+15)=X(239+EK)                                              ERIC
 4011 CONTINUE                                                          ERIC
      ESMF=EFD(1)                                                       ERIC
      DO 4012 EK=2,20                                                   ERIC
      ESMF=ESMF+EFD(EK)                                                 ERIC
 4012 CONTINUE                                                          ERIC
      DO 4013 EK=1,20                                                   ERIC
      ERFD(EK)=EFD(EK)/ESMF                                             ERIC
 4013 CONTINUE                                                          ERIC
C***ASSIGNED PREFERENCE INDEX EPRF(I,J), ABSOLUTE PREFERENCE ERNK(I,J) A ERIC
C***RELATIVE PREFERENCE ERPF(I,J) FOR FOOD GROUP I AND CONSUMER J ****** ERIC
      DO 4017 EJ=1,9                                                    ERIC
      DO 4014 EI=1,4                                                    ERIC
      EL=5*EI                                                           ERIC
      ERNK(EL-4,EJ)=EPRF(EL-4,EJ)/PHEN(1)                               ERIC
      ERNK(EL-3,EJ)=EPRF(EL-3,EJ)/PHEN(2)                               ERIC
      ERNK(EL-2,EJ)=EPRF(EL-2,EJ)/PHEN(3)                               ERIC
      ERNK(EL-1,EJ)=EPRF(EL-1,EJ)/PHEN(4)                               ERIC
      ERNK(EL,EJ)=EPRF(EL,EJ)/PHEN(5)                                   ERIC
 4014 CONTINUE                                                          ERIC
      ESMPF(EJ)=ERNK(1,EJ)                                              ERIC
      DO 4015 EI=2,20                                                   ERIC
      ESMPF(EJ)=ESMPF(EJ)+ERNK(EI,EJ)                                   ERIC
 4015 CONTINUE                                                          ERIC
      DO 4016 EI=1,20                                                   ERIC
      ERPF(EI,EJ)=ERNK(EI,EJ)/ESMPF(EJ)                                 ERIC
 4016 CONTINUE                                                          ERIC
 4017 CONTINUE                                                          ERIC
C***RANKING OF FOODS FOR A CONSUMER ************************************ ERIC
C***ERNK(I,J) WILL HEREAFTER STAND FOR THE RANK OF FOOD I WITH RESPECT T ERIC
C***CONSUMER J ********************************************************* ERIC
      DO 4020 EJ=1,9                                                    ERIC
      ESMRK(EJ)=ERPF(1,EJ)*ERFD(1)                                      ERIC
      DO 4018 EI=2,20                                                   ERIC
      ESMRK(EJ)=ESMRK(EJ)+ERPF(EI,EJ)*ERFD(EI)                          ERIC
 4018 CONTINUE                                                          ERIC
      DO 4019 EI=1,20                                                   ERIC
      ERNK(EI,EJ)=  ERPF(EI,EJ)*ERFD(EI)/ESMRK(EJ)                      ERIC
```

```
 4019 CONTINUE
 4020 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***THIS PART COMPUTES NUMBER OF DEGREE-DAYS PER DAY FOR EACH CATEGORY * ERIC
      EDGD1=EDGDS(ETMX1,ETMN1,EDZ1)                                     ERIC
      EDGD2=EDGDS(ETMX2,ETMN2,EDZ2)                                     ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      DO 4005 EJ=1,7                                                    ERIC
      DO 4005 EI=1,16                                                   ERIC
C***THIS PART COMPUTES THE NORMAL WEIGHT ENW(I,J) OF THIS TIME STEP **** ERIC
      IF(ETM1(EI,EJ).LT.370.) GO TO 41                                  ERIC
      IF(ENFIJ(EI,EJ).GT.0.) ETM1(EI,EJ)=TIME                           ERIC
   41 CONTINUE                                                          ERIC
      ETM2=ETM1(EI,EJ)+ETIME(EI,EJ)                                     ERIC
      ENW(EI,EJ)=ENW1(EI,EJ)                                            ERIC
      IF(ETM1(EI,EJ).GE.370.) GO TO 42                                  ERIC
      ENW(EI,EJ)=ALINT2(TIME,EFLG,ETM1(EI,EJ),ENW1(EI,EJ),ETM2,ENW2(EI,E ERIC
     EJ))                                                               ERIC
      IF(TIME.GT.ETM2) ENW(EI,EJ)=ENW2(EI,EJ)                           ERIC
   42 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***THIS PART COMPUTES THE ACTUAL AVERAGE WEIGHT EW(I,J) OF EACH CATEGOR ERIC
      EIW=ENW1(1,EJ)                                                    ERIC
      IF(EI.NE.1) EIW=EW(EI-1,EJ)                                       ERIC
      IF(EI.EQ.14.AND.EJ.EQ.2) EIW=EW(10,2)                             ERIC
      IF(EI.EQ.7.AND.EJ.EQ.4) EIW=ENW1(7,4)                             ERIC
      IF(EI.EQ.9.AND.EJ.EQ.7) EIW=EW(1,7)                               ERIC
      ENEG=ESMP(EI,EJ)+ENFL(EI,EJ)                                      ERIC
      IF(EI.EQ.16) GO TO 440                                            ERIC
      IF((EI.EQ.13.AND.EJ.EQ.2).OR.(EI.EQ.6.AND.EJ.EQ.4).OR.(EI.EQ.8.AND ERIC
     ..EJ.EQ.7)) GO TO 440                                              ERIC
      ENEG=ENEG+ENFIJ(EI+1,EJ)                                          ERIC
      IF(EI.EQ.10.AND.EJ.EQ.2) ENEG=ENEG+ENFIJ(14,2)                    ERIC
      IF(EI.EQ.1.AND.EJ.EQ.7) ENEG=ENEG+ENFIJ(9,7)                      ERIC
  440 CONTINUE                                                          ERIC
      IF(ENEG.LT.0.0000000001) ENEG=0.                                  ERIC
      IF(EN(EI,EJ)+ENFIJ(EI,EJ).GT.1.0000000001*ENEG) GO TO 4002        ERIC
      EN(EI,EJ)=0.                                                      ERIC
      EW(EI,EJ)=0.                                                      ERIC
      GO TO 4008                                                        ERIC
 4002 CONTINUE                                                          ERIC
      IF(EN(EI,EJ).GT.1.0000000001*ENEG) GO TO 4003                     ERIC
      EN(EI,EJ)=ENFIJ(EI,EJ)-(ENEG-EN(EI,EJ))                           ERIC
      EW(EI,EJ)=EIW                                                     ERIC
      IF(EN(EI,EJ).GE.0.00001) GO TO 4008                               ERIC
      EN(EI,EJ)=0.                                                      ERIC
      EW(EI,EJ)=0.                                                      ERIC
      GO TO 4008                                                        ERIC
 4003 CONTINUE                                                          ERIC
      EW(EI,EJ)=EW(EI,EJ)+(EFFI(EI,EJ)-EFIE(EI,EJ)-EFIR(EI,EJ))/EN(EI,EJ ERIC
     ))                                                                 ERIC
      IF(EW(EI,EJ).LT.ENW(EI,EJ)) EW(EI,EJ)=EW(EI,EJ)+(ENW(EI,EJ)-EW(EI, ERIC
     ,EJ))*ENFL(EI,EJ)/EN(EI,EJ)                                        ERIC
      EW(EI,EJ)=ENFIJ(EI,EJ)*EIW+(EN(EI,EJ)-ENEG)*EW(EI,EJ)             ERIC
      EN(EI,EJ)=EN(EI,EJ)+ENFIJ(EI,EJ)-ENEG                             ERIC
      EW(EI,EJ)=EW(EI,EJ)/EN(EI,EJ)                                     ERIC
 4008 CONTINUE                                                          ERIC
      IF(EW(EI,EJ).NE.0.AND.EW(EI,EJ).LT.0.000008001) WRITE(6,40000) EW( ERIC
     (EI,EJ),TIME,EI,EJ                                                 ERIC
40000 FORMAT(10X,F17.10,F8.2,2I4)                                       ERIC
      IF(EW(EI,EJ).GT.0.) GO TO 4005                                    ERIC
      EW(EI,EJ)=0.                                                      ERIC
                                                                        ERIC
```

```
      EN(EI,EJ)=0.
 4005 CONTINUE                                                          ERIC
C***********************************************************************ERIC
C                                                                       ERIC
C***********************************************************************ERIC
C***VARIABLES TO COMPUTE FOOD REQUIREMENT OF PREDATORS ******************ERIC
C***EW1, EW2, ETC. AND EWT1, EWT2, ETC. STAND FOR TOTAL ACTUAL AND NORMA ERIC
C***RESPECTIVELY FOR PREDATOR GROUPS 1, 2, ETC. *************************ERIC
      ERSP3=EFIR(5,1)+EFIR(7,1)                                         ERIC
      EW3=EW(5,1)*EN(5,1)+EW(7,1)*EN(7,1)                               ERIC
      EWT3=ENW(5,1)*EN(5,1)+ENW(7,1)*EN(7,1)                            ERIC
      EW4=0.                                                            ERIC
      EWT4=0.                                                           ERIC
      ERSP4=0.                                                          ERIC
      DO 4000 EI=7,9                                                    ERIC
      ERSP4=ERSP4+EFIR(EI,2)                                            ERIC
      EW4=EW4+EW(EI,2)*EN(EI,2)                                         ERIC
      EWT4=EWT4+ENW(EI,2)*EN(EI,2)                                      ERIC
 4000 CONTINUE                                                          ERIC
      DO 4009 EI=11,16                                                  ERIC
      ERSP4=ERSP4+EFIR(EI,2)                                            ERIC
      EW4=EW4+EW(EI,2)*EN(EI,2)                                         ERIC
      EWT4=EWT4+ENW(EI,2)*EN(EI,2)                                      ERIC
 4009 CONTINUE                                                          ERIC
      ERSP1=ERSP2=ERSP5=ERSP6=ERSP7=0.                                  ERIC
      EWT1=EPRD1                                                        ERIC
      EWT2=EPRD2                                                        ERIC
      EWT5=EPRD5*EPAD5                                                  ERIC
      EWT6=EPRD6*EPAD6                                                  ERIC
      EWT7=X(303)*EPAD1+X(304)*EPAD2+X(305)*EPAD3                       ERIC
      EDUI=EDU1                                                         ERIC
      IF(TIME.GT.150.AND.TIME.LT.250.) EDUI=0.                          ERIC
      EW1=EWT1*(1.3-EDUI)                                               ERIC
      EW2=EWT2*(1.3-EDUI)                                               ERIC
      EW5=EWT5*(1.3-EDUI)                                               ERIC
      EW6=EWT6*(1.3-EDU6)                                               ERIC
      EW7=EWT7*(1.3-EDU7)                                               ERIC
C***THIS PART COMPUTES FOOD PREFERENCES FOR PREDATORS *******************ERIC
C***COMPUTING RELATIVE PREY DENSITY (ERPD(I,J)) (BASED ON G CB/SQ.M) *** ERIC
      ESMF=0.                                                           ERIC
      DO 450 EJ=1,7                                                     ERIC
      DO 450 EI=1,16                                                    ERIC
      ESMF=ESMF+EN(EI,EJ)*EW(EI,EJ)                                     ERIC
  450 CONTINUE                                                          ERIC
      IF(ESMF.EQ.0.) GO TO 452                                          ERIC
      DO 451 EJ=1,7                                                     ERIC
      DO 451 EI=1,16                                                    ERIC
      ERPD(EI,EJ)=EN(EI,EJ)*EW(EI,EJ)/ESMF                              ERIC
  451 CONTINUE                                                          ERIC
  452 CONTINUE                                                          ERIC
C***COMPUTING RELATIVE RANK (ERRNK(I,J,K)) OF PREY (I,J) FOR PREDATOR K ERIC
C***EPPRF(I,J,K) IS THE ASSIGNED PREFERENCE INDEX ***********************ERIC
      DO 456 EK=1,7                                                     ERIC
      ESMPF(EK)=0.                                                      ERIC
      DO 453 EJ=1,7                                                     ERIC
      DO 453 EI=1,16                                                    ERIC
      ESMPF(EK)=ESMPF(EK)+EPPRF(EI,EJ,EK)                               ERIC
  453 CONTINUE                                                          ERIC
      ESMRK(EK)=0.                                                      ERIC
      DO 454 EJ=1,7                                                     ERIC
      DO 454 EI=1,16                                                    ERIC
      ESMRK(EK)=ESMRK(EK)+ERPD(EI,EJ)*EPPRF(EI,EJ,EK)/ESMPF(EK)         ERIC
```

```
  454 CONTINUE
      DO 455 EJ=1.7
      DO 455 EI=1.16
      ERRNK(EI.EJ.EK)=(ERPD(EI,EJ)*EPPRF(EI.EJ.EK)/ESMPF(EK))/ESMRK(EK)
  455 CONTINUE
  456 CONTINUE
C***THIS PART COMPUTES NUMBER OF PREY TAKEN FROM INSECT GROUP I.J BY PRE
C***NULLIFYING ENFP(I.J.K) ******************************************
      DO 457 EK=1.7
      DO 457 EJ=1.7
      DO 457 EI=1.16
      ENFP(EI.EJ.EK)=0.
      ESMP(EI.EJ)=0.
  457 CONTINUE
C***COMPUTING PREY TAKEN ********************************************
      DO 461 EJ=1.7
      DO 461 EI=1.16
      EPPP=0.
      IF(EW(EI.EJ).LE.0.) GO TO 460
      EK=1
      ENFP(EI.EJ.1)=EPRED(EW1.EWT1.ERSP1,ERPR,EDGD2)*DT
      EK=2
      ENFP(EI.EJ.2)=EPRED(EW2,EWT2,ERSP2,ERPR,EDGD1)*DT
      EK=3
      ENFP(EI.EJ.3)=EPRED(EW3.EWT3.ERSP3,ERPR,EDGD2)*(1.+ECUT(3))*DT
      EK=4
      EPPP=ENFIJ(1.2)*EW(1.2)
      ENFP(EI,EJ.4)=EPRED(EW4.EWT4.ERSP4.ERPR,EDGD1)*(1.+ECUT(4))*DT
      EK=5
      ERPR=0.
      ENFP(EI.EJ.5)=EPRED(EW5.EWT5,ERSP5,ERPR,EDGD2)*DT
      EK=6
      ENFP(EI.EJ.6)=EPRED(EW6.EWT6,ERSP6.ERPR,E12)*DT
      EK=7
      ENFP(EI.EJ.7)=EPRED(EW7,EWT7.ERSP7.ERPR.E12)*DT
C***SUMMING ALL PREY TAKEN FROM INSECT GROUP (I,J) ********************
      ESMP(EI.EJ)=0.
      DO 458 EK=1.7
      ESMP(EI.EJ)=ESMP(EI.EJ)+ENFP(EI.EJ.EK)
  458 CONTINUE
      IF(ESMP(EI.EJ).LE.ERT3*EN(EI,EJ)) GO TO 460
      IF(ESMP(EI.EJ).LT.ERT5*EN(EI,EJ)) GO TO 462
      ESMP(EI,EJ)=0.
      DO 463 EK=1.7
      ENFP(EI.EJ.EK)=0.
  463 CONTINUE
      GO TO 460
  462 CONTINUE
      ESMPR=ESMP(EI,EJ)*(0.5*SIN(3.14159*(ESMP(EI,EJ)/EN(EI,EJ)+0.5*ERT5
     5-1.5*ERT3)/(ERT5-ERT3))+0.5)
      DO 459 EK=1.7
      ENFP(EI.EJ,EK)=ENFP(EI,EJ,EK)*ESMPR/ESMP(EI,EJ)
  459 CONTINUE
      ESMP(EI.EJ)=ESMPR
  460 CONTINUE
  461 CONTINUE
C**********************************************************************
C**********************************************************************
C***BEGINNING OF THE SPECIFIC COMPUTATIONS FOR CARABIDES ***************
C**********************************************************************
C**********************************************************************
      EREP=0.
      DO 405 EI=1.16
      ESUM(EI.1)=ESUM(EI.1)+EDGD2*DT
C***THIS PART COMPUTES THE NUMBER DYING FROM NON-PREDATORY CAUSES ******
      ENFL(EI.1)=0.
      IF(EN(EI.1).LE.0.) GO TO 402
      IF(ESMP(EI.1).EQ.EN(EI.1)) GO TO 402
      ESOIT=EAVTM
      ESOIM=ATEN(3)
      ESFD=1.
```

```
      IF(EW(EI,1).GT.EDW      *ENW(EI,1)) GO TO 48                      ERIC
      ESFD=1.-(EW(EI,1)-EDW*ENW(EI,1))*(EW(EI,1)-EDW*ENW(EI,1))/(EDW*EDW ERIC
     **ENW(EI,1)*ENW(EI,1))                                             ERIC
   48 CONTINUE                                                          ERIC
      IF(EI.EQ.5.OR.EI.EQ.7.OR.EI.EW.14.OR.EI.EQ.16) GO TO 47           ERIC
      ESTMP=ALINT2(ESOIT,EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1)   ERIC
      ESMST=ALINT2(ESOIM,EFLG,ET51,1.,ET61,EMN2)                         ERIC
      GO TO 400                                                         ERIC
   47 CONTINUE                                                          ERIC
      IF(EI.EQ.16) GO TO 49                                             ERIC
      ESTMP=ALINT2(ESOIT,EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1)   ERIC
      ESMST=ALINT2(ESOIM,EFLG,ET52,1.,ET62,EMN2)                        ERIC
      GO TO 400                                                         ERIC
   49 CONTINUE                                                          ERIC
      ESTMP=ALINT2(AVSTM(1),EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1) ERIC
      ESMST=ALINT2(APEVA,EFLG,ET53,1.,ET63,EMN2)                        ERIC
  400 CONTINUE                                                          ERIC
      ENFL(EI,1)=EN(EI,1)*(1.-ESTMP*ESMST*ESFD)*DT                      ERIC
      IF(ENFL(EI,1).GT.EN(EI,1)-ESMP(EI,1)) ENFL(EI,1)=EN(EI,1)-ESMP(EI, ERIC
     .1)                                                                ERIC
  402 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***THIS PART COMPUTES THE FLOW FROM THE PREVIOUS CATEGORY TO THE PRESEN ERIC
      EK=EI-1                                                           ERIC
      IF(EI.EQ.1) EK=16                                                 ERIC
      ENFIJ(EI,1)=0.                                                    ERIC
      IF(EN(EK,1).LE.0.) ESUM(EK,1)=0.                                  ERIC
      IF(EN(EK,1).LE.0.) GO TO 405                                      ERIC
      IF(TIME.LE.ETIM1(EI,1).OR.TIME.GE.ETIM2(EI,1)) GO TO 405          ERIC
      IF(EI.NE.1) GO TO 403                                             ERIC
      EGGP=EEPF(EMER(1),E12,EDGD1,EW(EK,1),EMW(1))                      ERIC
      ENFIJ(EI,1)=EN(EK,1)*EPPF(1)*EGGP*DT                              ERIC
      EPEP=ENFIJ(EI,1)*EW(1,1)                                          ERIC
      GO TO 405                                                         ERIC
  403 CONTINUE                                                          ERIC
      IF((EI.GE.6.AND.EI.LE.8).OR.EI.GE.15) GO TO 404                   ERIC
      CALL ESTEP(ESUM(EK,1),EDG(EK,1),EA,EB)                            ERIC
      ESUM(EK,1)=EB                                                     ERIC
      ENFIJ(EI,1)=EN(EK,1)*EA                                           ERIC
      GO TO 4007                                                        ERIC
  404 CONTINUE                                                          ERIC
      EX1=0.                                                            ERIC
      EX2=E12                                                           ERIC
      EY1=0.                                                            ERIC
      EY2=0.15                                                          ERIC
      IF(EI.NE.8) GO TO 4006                                            ERIC
      IF(ESUM(7,1).LT.EDG(7,1)) GO TO 4007                              ERIC
      EX1=E12/2.                                                        ERIC
 4006 CONTINUE                                                          ERIC
      IF(EI.NE.6.AND.EI.NE.15) GO TO 4004                               ERIC
      EY1=0.2                                                           ERIC
      EY2=0.                                                            ERIC
 4004 CONTINUE                                                          ERIC
      ENFIJ(EI,1)=EN(EK,1)*ALINT2(EDGD2,EFLG,EX1,EY1,EX2,EY2)*DT        ERIC
 4007 CONTINUE                                                          ERIC
  405 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      DO 4039 EI=1,16                                                   ERIC
C***THIS PART COMPUTES THE RESPIRATION AS A FUNCTION OF WEIGHT, AMBIENT  ERIC
C***TEMPERATURE, AND ACTIVITY ****************************************** ERIC
      EFFI(EI,1)=0.                                                     ERIC
      EFIR(EI,1)=0.                                                     ERIC
      EK=1                                                              ERIC
      IF(EN(EI,1).LE.0.) GO TO 4024                                     ERIC
      IF(EW(EI,1).LE.0.) GO TO 4024                                     ERIC
      IF(EI.EQ.16) GO TO 4035                                           ERIC
      EFIR(EI,1)=ERSPR(ETMX2,ETMN2)*DT                                  ERIC
      IF(EI.EQ.6.OR.EI.EQ.15) EFIR(EI,1)=EFIR(EI,1)/EIAF                ERIC
      IF(EI.EQ.5.OR.EI.EQ.7.OR.EI.EQ.14) EFIR(EI,1)=EFIR(EI,1)*EACF     ERIC
```

```
      GO TO 4036                                                         ERIC
 4035 CONTINUE                                                           ERIC
      EFIR(EI,1)=ERSPR(ETMX1,ETMN1)*EACF*DT                              ERIC
 4036 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
C***FOOD CONSUMPTION BY PLANT FEEDERS ********************************** ERIC
      IF(EI.NE.14.AND.EI.NE.16) GO TO 4022                               ERIC
      ERPR=EREP                                                          ERIC
      EK=1                                                               ERIC
      EJ=2                                                               ERIC
      EDGDD=EDGD1                                                        ERIC
      IF(EI.EQ.16) GO TO 4025                                            ERIC
      EJ=1                                                               ERIC
      ERPR=0.                                                            ERIC
      EDGDD=EDGD2                                                        ERIC
 4025 CONTINUE                                                           ERIC
      EFFI(EI,1)=EFOD(ERPR,EDGDD)*DT                                     ERIC
 4022 CONTINUE                                                           ERIC
      IF(EI.NE.5.AND.EI.NE.7) GO TO 4024                                 ERIC
C***FOOD CONSUMPTION BY PREDATORS ************************************** ERIC
      EFOOD=0.                                                           ERIC
      DO 4023 EK=1,16                                                    ERIC
      DO 4023 EJ=1,7                                                     ERIC
      EFOOD=EFOOD+ENFP(EK,EJ,3)/(1.+ECUT(3))*EW(EK,EJ)                   ERIC
 4023 CONTINUE                                                           ERIC
      EFFI(EI,1)=EFOOD*EN(EI,1)*EW(EI,1)/(EN(5,1)*EW(5,1)+EN(7,1)*EW(7,1 ERIC
     )))                                                                 ERIC
 4024 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
C***EXCRETION COMPUTED AS A FRACTION OF FOOD INGESTED ****************** ERIC
      EFIE(EI,1)=EFFI(EI,1)*(1.-EASK(1))                                 ERIC
 4039 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
C***BEGINNING OF THE SPECIFIC COMPUTATIONS FOR SPIDERS ***************** ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
      EREP=0.                                                            ERIC
      DO 4124 EI=1,16                                                    ERIC
      ESUM(EI,2)=ESUM(EI,2)+EDGD1*DT                                     ERIC
C***NUMBER OF SPIDERS DYING FROM NON-PREDATORY CAUSES ***************** ERIC
      ENFL(EI,2)=0.                                                      ERIC
      IF(EN(EI,2).LE.0.) GO TO 4117                                      ERIC
      IF(ESMP(EI,2).EQ.EN(EI,2)) GO TO 4117                              ERIC
      ESFD=1.                                                            ERIC
      IF(EW(EI,2).GT.EDW*ENW(EI,2)) GO TO 4113                           ERIC
      ESFD=1.-(EW(EI,2)-EDW*ENW(EI,2))*(EW(EI,2)-EDW*ENW(EI,2))/(EDW*EDW ERIC
     **ENW(EI,2)*ENW(EI,2))                                              ERIC
 4113 CONTINUE                                                           ERIC
      IF(EI.LE.6.OR.EI.EQ.10) GO TO 4114                                 ERIC
      ESTMP=ALINT2(AVSTM(1),EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1) ERIC
      ESMST=ALINT2(APEVA,EFLG,ET53,1.,ET63,EMN2)                         ERIC
      GO TO 4116                                                         ERIC
 4114 CONTINUE                                                           ERIC
      IF(EI.EQ.10) GO TO 4115                                            ERIC
      ESTMP=ALINT2(AVSTM(1),EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1) ERIC
      ESMST=1.                                                           ERIC
      GO TO 4116                                                         ERIC
 4115 CONTINUE                                                           ERIC
      ESTMP=ALINT2(EAVTM,EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1)    ERIC
      ESMST=ALINT2(ATEN(3),EFLG,ET51,1.,ET61,EMN2)                       ERIC
 4116 CONTINUE                                                           ERIC
      ENFL(EI,2)=EN(EI,2)*(1.-ESFD*ESTMP*ESMST)*DT                       ERIC
      IF(ENFL(EI,2).GT.EN(EI,2)-ESMP(EI,2)) ENFL(EI,2)=EN(EI,2)-ESMP(EI, ERIC
     ,2)                                                                 ERIC
 4117 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
C***TRANSFER FROM THE PREVIOUS CATEGORY OF SPIDERS TO THE PRESENT ****** ERIC
```

```
      EK=EI-1                                                           ERIC
      IF(EI.EQ.1) EK=16                                                 ERIC
      IF(EI.EQ.14) EK=10                                                ERIC
      ENFIJ(EI,2)=0.                                                    ERIC
      IF(EN(EK,2).LE.0.) ESUM(EK,2)=0.                                  ERIC
      IF(EN(EK,2).LE.0.) GO TO 4124                                     ERIC
      IF(TIME.LE.ETIM1(EI,2).OR.TIME.GE.ETIM2(EI,2)) GO TO 4124         ERIC
      IF(EI.NE.1) GO TO 4118                                            ERIC
      EGGP=EEPF(EMER(2),E12,EDGD1,EW(EK,2),EMW(2))                      ERIC
      ENFIJ(EI,2)=EN(EK,2)*EGGP*DT                                      ERIC
      GO TO 4124                                                        ERIC
 4118 CONTINUE                                                          ERIC
      IF(EI.GE.8) GO TO 4119                                            ERIC
      CALL ESTEP(ESUM(EK,2),EDG(EK,2),EA,EB)                            ERIC
      ESUM(EK,2)=EB                                                     ERIC
      ENFIJ(EI,2)=EN(EK,2)*EA                                           ERIC
      GO TO 4123                                                        ERIC
 4119 CONTINUE                                                          ERIC
      EDGDD=EDGD1                                                       ERIC
      EHAF=1.                                                           ERIC
      IF(EI.NE.11.AND.EI.NE.14) GO TO 4120                              ERIC
      EHAF=.5                                                           ERIC
      EDGDD=EDGD2                                                       ERIC
 4120 CONTINUE                                                          ERIC
      EX1=0.                                                            ERIC
      EX2=E12                                                           ERIC
      EY1=0.                                                            ERIC
      EY2=0.15                                                          ERIC
      IF(EI.NE.10) GO TO 4121                                           ERIC
      EY1=0.2                                                           ERIC
      EY2=0.                                                            ERIC
 4121 CONTINUE                                                          ERIC
      IF(EI.EQ.10.OR.EI.EQ.11.OR.EI.EQ.14) GO TO 4122                   ERIC
      IF(ESUM(EK,2).LT.EDG(EK,2)) GO TO 4123                            ERIC
 4122 CONTINUE                                                          ERIC
      ENFIJ(EI,2)=EN(EK,2)*ALINT2(EDGDD,EFLG,EX1,EY1,EX2,EY2)*DT*EHAF   ERIC
 4123 CONTINUE                                                          ERIC
 4124 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      DO 4130 EI=1,16                                                   ERIC
C***SPIDER RESPIRATION AS FUNCTION OF WEIGHT, AMBIENT TEMPERATURE AND AC ERIC
      EFFI(EI,2)=0.                                                     ERIC
      EFIR(EI,2)=0.                                                     ERIC
      EK=2                                                              ERIC
      IF(EN(EI,2).LE.0.) GO TO 4126                                     ERIC
      IF(EW(EI,2).LE.0.) GO TO 4126                                     ERIC
      EFIR(EI,2)=ERSPR(ETMX1,ETMN1)*EACF*DT                             ERIC
      IF(EI.LE.6) EFIR(EI,2)=EFIR(EI,2)/EACF                            ERIC
      IF(EI.EQ.10) EFIR(EI,2)=ERSPR(ETMX2,ETMN2)/EIAF*DT                ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***FOOD CONSUMPTION SPIDERS ******************************************* ERIC
      IF(EI.LE.6.OR.EI.EQ.10) GO TO 4126                                ERIC
      EFOOD=0.                                                          ERIC
      DO 4125 EK=1,16                                                   ERIC
      DO 4125 EJ=1,7                                                    ERIC
      EFOOD=EFOOD+ENFP(EK,EJ,4)/(1.+ECUT(4))*EW(EK,EJ)                  ERIC
 4125 CONTINUE                                                          ERIC
      EFFI(EI,2)=EFOOD*EN(EI,2)*EW(EI,2)/EW4                            ERIC
 4126 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***SPIDERS EXCRETION ( AS FRACTION OF FOOD INGESTED ) ***************** ERIC
      EFIE(EI,2)=EFFI(EI,2)*(1.-EASK(2))                                ERIC
 4130 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
C***BEGINNING OF SPESIFIC COMPUTATIONS FOR CHRYS/CURC ****************** ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
```

```
      EREP=0.
      DO 4144 EI=1,16                                                    ERIC
      ESUM(EI,3)=ESUM(EI,3)+EDGD2*DT                                     ERIC
C***NUMBER OF CHRYS/CURC DYING FROM NON-PREDATORY CAUSES *************** ERIC
      ENFL(EI,3)=0.                                                      ERIC
      IF(EN(FI,3).LE.0.) GO TO 4138                                      ERIC
      IF(EN(EI,3).EQ.ESMP(EI,3)) GO TO 4138                              ERIC
      ESFD=1.                                                            ERIC
      IF(EW(EI,3).GT.EDW*ENW(EI,3)) GO TO 4134                           ERIC
      ESFD=1.-(EW(EI,3)-EDW*ENW(EI,3))*(EW(EI,3)-EDW*ENW(EI,3))/(EDW*EDW ERIC
     **ENW(EI,3)*FNW(EI,3))                                              ERIC
 4134 CONTINUE                                                           ERIC
      IF((EI.GE.7.AND.EI.LE.9).OR.EI.EQ.14.OR.EI.EQ.16) GO TO 4135       ERIC
      ESTMP=ALINT2(EAVTM,EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1)    ERIC
      ESMST=ALINT2(ATEN(3),EFLG,ET51,1.,ET61,EMN2)                       ERIC
      GO TO 4137                                                         ERIC
 4135 CONTINUE                                                           ERIC
      IF(EI.EQ.14.OR.EI.EQ.16) GO TO 4136                                ERIC
      ESTMP=ALINT2(EAVTM,EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1)    ERIC
      ESMST=ALINT2(ATEN(3),EFLG,ET52,1.,ET62,EMN2)                       ERIC
      GO TO 4137                                                         ERIC
 4136 CONTINUE                                                           ERIC
      ESTMP=ALINT2(AVSTM(1),EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1) ERIC
      ESMST=ALINT2(APEVA,EFLG,ET53,1.,ET63,EMN2)                         ERIC
 4137 CONTINUE                                                           ERIC
      ENFL(EI,3)=EN(EI,3)*(1.-ESFD*ESTMP*ESMST)*DT                       ERIC
      IF(ENFL(EI,3).GT.EN(EI,3)-ESMP(EI,3)) ENFL(EI,3)=EN(EI,3)-ESMP(EI, ERIC
     .3)                                                                 ERIC
 4138 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
C***TRANSFER FROM THE PREVIOUS CATEGORY OF CHRYS/CURC TO THE PRESENT *** ERIC
      EK=EI-1                                                            ERIC
      IF(EI.EQ.1) EK=16                                                  ERIC
      ENFIJ(EI,3)=0.                                                     ERIC
      IF(EN(EK,3).LE.0.) ESUM(EK,3)=0.                                   ERIC
      IF(EN(EK,3).LE.0.) GO TO 4144                                      ERIC
      IF(TIME.LE.ETIM1(EI,3).OR.TIME.GE.ETIM2(EI,3)) GO TO 4144          ERIC
      IF(EI.NE.1) GO TO 4139                                             ERIC
      EGGP=EEPF(EMER(3),E12,EDGD1,EW(EK,3),EMW(3))                       ERIC
      ENFIJ(EI,3)=EN(EK,3)*EPPF(3)*EGGP*DT                               ERIC
      EREP=ENFIJ(EI,3)*EW(1,3)                                           ERIC
      GO TO 4144                                                         ERIC
 4139 CONTINUE                                                           ERIC
      IF((EI.GE.8.AND.EI.LE.10).OR.EI.GE.15) GO TO 4140                  ERIC
      CALL ESTEP(ESUM(EK,3),EDG(EK,3),EA,EB)                             ERIC
      ESUM(EK,3)=EB                                                      ERIC
      ENFIJ(FI,3)=EN(EK,3)*EA                                            ERIC
      GO TO 4143                                                         ERIC
 4140 CONTINUE                                                           ERIC
      EX1=0.                                                             ERIC
      EX2=E12                                                            ERIC
      EY1=0.                                                             ERIC
      EY2=0.15                                                           ERIC
      IF(EI.GE.15) GO TO 4141                                            ERIC
      IF(ESUM(EK,3).LT.EDG(EK,3)) GO TO 4143                             ERIC
 4141 CONTINUE                                                           ERIC
      IF(EI.NE.15) GO TO 4142                                            ERIC
      EY1=0.2                                                            ERIC
      EY2=0.                                                             ERIC
 4142 CONTINUE                                                           ERIC
      ENFIJ(FI,3)=EN(EK,3)*ALINT2(EDGD2,EFLG,EX1,EY1,EX2,EY2)*DT         ERIC
 4143 CONTINUE                                                           ERIC
 4144 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
      DO 4147 EI=1,16                                                    ERIC
C***THIS PART COMPUTES RESPIRATION FOR AN INDIVIDUAL CHRYS/CURC AS FUNCT ERIC
C***WEIGHT, AMBIENT TEMPERATURE AND ACTIVITY *************************** ERIC
      EFIP(EI,3)=0.                                                      ERIC
      EFFI(EI,3)=0.                                                      ERIC
      EK=3                                                               ERIC
                                                                         ERIC
```

```
      IF(EN(EI,3).LE.0.) GO TO 4146
      IF(EW(EI.3).LE.0.) GO TO 4146
      EFIP(EI.3)=ERSPR(ETMX2.ETMN2)*DT
      IF(EI.GE.7.AND.EI.LE.9) EFIR(EI,3)=EFIR(EI,3)*EACF
      IF(EI.EQ.15) EFIR(EI,3)=EFIR(EI.3)/EIAF
      IF(EI.EQ.14.OR.EI.EQ.16) EFIR(EI,3)=ERSPR(ETMX1,ETMN1)*EACF*DT
C***********************************************************************
C
C***********************************************************************
C***FOOD INTAKE BY CHRYS/CURC ******************************************
      ERPR=0.
      IF(EI.EQ.16) ERPR=EREP
      EJ=4
      EDGDD=EDGD1
      IF(EI.EQ.14.OR.EI.EQ.16) GO TO 4145
      IF(EI.LT.7.OR.EI.GT.9) GO TO 4146
      EJ=3
      EDGDD=EDGD2
 4145 CONTINUE
      EFFI(EI.3)=EFOD(ERPR,EDGDD)*DT
 4146 CONTINUE
C***********************************************************************
C
C***********************************************************************
C***EXCRETION BY CHRYS/CURC ********************************************
      EFIE(EI.3)=EFFI(EI.3)*(1.-EASK(3))
 4147 CONTINUE
C***********************************************************************
C***********************************************************************
C***BEGINNING OF THE SPECIFIC COMPUTATIONS FOR THE HEMIPTERA/HOMOPTERA G
C***********************************************************************
C***********************************************************************
      EREP=0.
      EREP1=0.
      DO 4156 EI=1.16
C***NUMBER OF HEMI/HOMO DYING FROM NON-PREDATORY CAUSES ****************
      IF(EN(EI.4).LE.0.) GO TO 4152
      IF(EN(EI.4).EQ.ESMP(EI.4)) GO TO 4152
      ESFD=1.
      IF(EW(EI.4).GT.EDW*ENW(EI,4)) GO TO 4149
      ESFD=1.-(EW(EI.4)-EDW*ENW(EI,4))*(EW(EI,4)-EDW*ENW(EI,4))/(EDW*EDW
     **ENW(EI,4)*ENW(EI,4))
 4149 CONTINUE
      IF(EI.EQ.1.OR.(EI.GE.7.AND.EI.LE.11)) GO TO 4150
      ESTMP=ALINT2(AVSTM(1).EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1)
      ESMST=ALINT2(APEVA,EFLG,ET53,1.,ET63.EMN2)
      GO TO 4151
 4150 CONTINUE
      ESTMP=ALINT2(AVSTM(1).EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1)
      ESMST=ALINT2(APEVA,EFLG,ET54.1.,ET64.EMN2)
 4151 CONTINUE
      ENFL(EI,4)=EN(EI.4)*(1.-ESFD*ESTMP*ESMST)*DT
      IF(ENFL(EI.4).GT.EN(EI.4)-ESMP(EI,4)) ENFL(EI,4)=EN(EI,4)-ESMP(EI,
     .4)
 4152 CONTINUE
C***********************************************************************
C
C***********************************************************************
C***TRANSFER FROM THE PREVIOUS CATEGORY OF HEMI/HOMO TO THE PRESENT ****
      ENFIJ(EI,4)=0.
      EK=EI-1
      IF(EI.EQ.1) EK=16
      IF(EN(EK.4).LE.0.) ESUM(EK.4)=0.
      IF(EN(EK.4).LE.0.) GO TO 4156
      IF(TIME.LE.ETIM1(EI.4).OR.TIME.GE.ETIM2(EI,4)) GO TO 4156
      IF(EI.NE.1.AND.EI.NE.7) GO TO 4153
      EGGP=EEPF(EMER(4),E12.EDGD1,EW(EK,4),EMW(4))
      ENFIJ(EI.4)=EN(EK,4)*EPPF(4)*EGGP*DT
      EREP1=ENFIJ(EI.4)*EW(1.4)
      IF(EI.EQ.7) EREP=ENFIJ(EI,4)*EW(7.4)
      GO TO 4156
```

```
 4153 CONTINUE
      IF(EI.LT.8.OR.EI.GT.12) GO TO 4154
      CALL ESTEP(ESUM(EK,4),EDG(EK,4),EA,EB)
      ESUM(EK,4)=EB
      ENFIJ(EI,4)=EN(EK,4)*EA
      GO TO 4155
 4154 CONTINUE
      EX1=0.
      EX2=E12
      EY1=0.
      EY2=0.1
      IF(ESUM(EK,4).LT.EDG(EK,4)) GO TO 4156
      ENFIJ(EI,4)=EN(EK,4)*ALINT2(EDGD1,EFLG,EX1,EY1,EX2,EY2)*DT
 4155 CONTINUE
 4156 CONTINUE
C***********************************************************************
C
C***********************************************************************
      DO 4158 EI=1,16
C***HEMI/HOMO RESPIRATION AS FUNCTION OF WEIGHT, AMBIENT TEMPERATURE,
C*** AND ACTIVITY ******************************************************
      EFIR(EI,4)=0.
      EFFI(EI,4)=0.
      EK=4
      IF(EN(EI,4).LE.0.) GO TO 4157
      IF(EW(EI,4).LE.0.) GO TO 4157
      EFIR(EI,4)=ERSPR(ETMX1,ETMN1)*DT
      IF(EI.EQ.1) EFIR(EI,4)=EFIR(EI,4)/EIAF
      IF((EI.GE.2.AND.EI.LE.6).OR.EI.GE.12) EFIR(EI,4)=EFIR(EI,4)*EACF
C***********************************************************************
C
C***********************************************************************
C***FEEDING BY HEMI/HOMO ***********************************************
      IF(EI.EQ.1.OR.(EI.GE.7.AND.EI.LE.11)) GO TO 4157
      ERPR=EREP
      EJ=5
      EDGDD=EDGD1
      IF(EI.EQ.16) ERPR=EREP1
      IF(EI.NE.6.AND.EI.NE.16) ERPR=0.
      EFFI(EI,4)=EFOD(ERPR,EDGDD)*DT
 4157 CONTINUE
C***********************************************************************
C
C***********************************************************************
C***EXCRETION BY HEMI/HOMO *********************************************
      EFIE(EI,4)=EFFI(EI,4)*(1.-EASK(4))
 4158 CONTINUE
C***********************************************************************
C***********************************************************************
C***BEGINNING OF THE SPECIFIC COMPUTATIONS FOR SCARABAEIDAE ************
C***********************************************************************
C***********************************************************************
      EREP=0.
      DO  4171 EI=1,16
      ESUM(EI,5)=ESUM(EI,5)+EDGD2*DT
C***NUMBER OF SCARABAEIDAE DYING FROM NON-PREDATORY CAUSES *************
      IF(EN(EI,5).LE.0.) GO TO 4165
      IF(EN(EI,5).EQ.ESMP(EI,5)) GO TO 4165
      ESFD=1.
      IF(EW(EI,5).GT.EDW*ENW(EI,5)) GO TO 4161
      ESFD=1.-(EW(EI,5)-EDW*ENW(EI,5))*(EW(EI,5)-EDW*ENW(EI,5))/(EDW*EDW
     **ENW(EI,5)*ENW(EI,5))
 4161 CONTINUE
      IF(EI.NE.16) GO TO 4162
      ESTMP=ALINT2(AVSTM(1),EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1)
      ESMST=ALINT2(APEVA,EFLG,ET53,1.,ET63,EMN2)
      GO TO 4164
 4162 CONTINUE
      IF(EI.EQ.5.OR.EI.EQ.7.OR.EI.EQ.9.OR.EI.EQ.11) GO TO 4163
      ESTMP=ALINT2(EAVTM,EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1)
      ESMST=ALINT2(ATEN(3),EFLG,ET51,1.,ET61,EMN2)
```

```
      GO TO 4164
 4163 CONTINUE                                                           ERIC
      ESTMP=ALINT2(EAVTM,EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1)    ERIC
      ESMST=ALINT2(ATEN(3),EFLG,ET52,1.,ET62,EMN2)                       ERIC
 4164 CONTINUE                                                           ERIC
      ENFL(EI,5)=EN(EI,5)*(1.-ESFD*ESTMP*ESMST)*DT                       ERIC
      IF(ENFL(EI,5).GT.EN(EI,5)-ESMP(EI,5)) ENFL(EI,5)=EN(EI,5)-ESMP(EI, ERIC
     *5)                                                                 ERIC
 4165 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
C***TRANSFER FROM THE PREVIOUS CATEGORY OF SCARABAEIDAE TO THE PRESENT * ERIC
      EK=EI-1                                                            ERIC
      IF(EI.EQ.1) EK=16                                                  ERIC
      ENFIJ(EI,5)=0.                                                     ERIC
      IF(EN(EK,5).LE.0.) ESUM(EK,5)=0.                                   ERIC
      IF(EN(EK,5).LE.0.) GO TO 4171                                      ERIC
      IF(TIME.LE.ETIM1(EI,5).OR.TIME.GE.ETIM2(EI,5)) GO TO 4171          ERIC
      IF(EI.NE.1) GO TO 4166                                             ERIC
      EGGP=EEPF(EMER(5),E12,EDGD1,EW(EK,5),EMW(5))                       ERIC
      ENFIJ(EI,5)=EN(EK,5)*EPPF(5)*EGGP*DT                               ERIC
      EREP=ENFIJ(EI,5)*EW(1,5)                                           ERIC
      GO TO 4171                                                         ERIC
 4166 CONTINUE                                                           ERIC
      IF(EI.GE.6.AND.EI.LE.12) GO TO 4167                                ERIC
      CALL ESTEP(ESUM(EK,5),EDG(EK,5),EA,EB)                             ERIC
      ESUM(EK,5)=EB                                                      ERIC
      ENFIJ(EI,5)=EN(EK,5)*EA                                            ERIC
      GO TO 4170                                                         ERIC
 4167 CONTINUE                                                           ERIC
      EX1=0.                                                             ERIC
      EX2=E12                                                            ERIC
      EY1=0.                                                             ERIC
      EY2=0.15                                                           ERIC
      IF(EI.NE.12) GO TO 4168                                            ERIC
      IF(ESUM(EK,5).LT.EDG(EK,5)) GO TO 4170                             ERIC
 4168 CONTINUE                                                           ERIC
      IF(EI.NE.6.AND.EI.NE.8.AND.EI.NE.10) GO TO 4169                    ERIC
      EY1=0.2                                                            ERIC
      EY2=0.                                                             ERIC
 4169 CONTINUE                                                           ERIC
      ENFIJ(EI,5)=EN(EK,5)*ALINT2(EDGD2,EFLG,EX1,EY1,EX2,EY2)*DT         ERIC
 4170 CONTINUE                                                           ERIC
 4171 CONTINUE                                                           ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
      DO 4174 EI=1,16                                                    ERIC
C***COMPUTING RESPIRATION FOR SCARABAEIDAE AS FUNCTION OF WEIGHT, AMBIEN ERIC
C*** TEMPERATURE, AND ACTIVITY ***************************************** ERIC
      EFFI(EI,5)=0.                                                      ERIC
      EFIR(EI,5)=0.                                                      ERIC
      EK=5                                                               ERIC
      IF(EN(EI,5).LE.0.) GO TO 4173                                      ERIC
      IF(EW(EI,5).LE.0.) GO TO 4173                                      ERIC
      EFIR(EI,5)=ERSPR(ETMX2,ETMN2)*DT                                   ERIC
      IF(EI.EQ.6.OR.EI.EQ.8.OR.EI.EQ.10) EFIR(EI,5)=EFIR(EI,5)/EIAF      ERIC
      IF(EI.EQ.5.OR.EI.EQ.7.OR.EI.EQ.9.OR.EI.EQ.11) EFIR(EI,5)=EFIR(EI,5 ERIC
     ))*EACF                                                             ERIC
      IF(EI.EQ.16) EFIR(EI,5)=ERSPR(ETMX1,ETMN1)*EACF*DT                 ERIC
      IF(EI.EQ.16) EFIR(EI,5)=ERSPR(ETMX1,ETMN1)*DT                      ERIC
C*********************************************************************** ERIC
C                                                                        ERIC
C*********************************************************************** ERIC
C***FEEDING BY SCARABAEIDAE ******************************************** ERIC
      IF(EI.NE.5.AND.EI.NE.7.AND.EI.NE.9.AND.EI.NE.11.AND.EI.NE.16) GO   ERIC
     *TO 4173                                                            ERIC
      ERPR=EREP                                                          ERIC
      EJ=7                                                               ERIC
```

```
      EDGDD=EDGD1                                                       ERIC
      IF(EI.EQ.16) GO TO 4172                                           ERIC
      EJ=6                                                              ERIC
      ERPR=0.                                                           ERIC
      EDGDD=EDGD2                                                       ERIC
 4172 CONTINUE                                                          ERIC
      EFFI(EI,5)=EFOU(ERPR,EDGDD)*DT                                    ERIC
 4173 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***EXCRETION BY SCARABAEIDAE ****************************************** ERIC
      EFFI(EI,5)=EFFI(EI,5)*(1.-EASK(5))                                ERIC
 4174 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
C***BEGINNING OF SPECIFIC COMPUTATIONS FOR TENEBRIONIDAE *************** ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
      DO 4185 EI=1,16                                                   ERIC
      ESUM(EI,6)=ESUM(EI,6)+EDGD2*DT                                    ERIC
C***NUMBER OF TENEBRIONIDAE DYING FROM NON-PREDATORY CAUSES ************ ERIC
      IF(EN(EI,6).LE.0.) GO TO 4180                                     ERIC
      IF(EN(EI,6).LE.ESMP(EI,6)) GO TO 4180                             ERIC
      ESFD=1.                                                           ERIC
      IF(EW(EI,6).GT.EDW*ENW(EI,6)) GO TO 4177                          ERIC
      ESFD=1.-(EW(EI,6)-EDW*ENW(EI,6))*(EW(EI,6)-EDW*ENW(EI,6))/(EDW*EDW ERIC
     **ENW(EI,6)*ENW(EI,6))                                             ERIC
 4177 CONTINUE                                                          ERIC
      IF(EI.EQ.5.OR.EI.EQ.7.OR.EI.EQ.9.OR.EI.EQ.16) GO TO 4178          ERIC
      ESTMP=ALINT2(EAVTM,EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1)   ERIC
      ESMST=ALINT2(ATEN(3),EFLG,ET51,1.,ET61,EMN2)                      ERIC
      GO TO 4179                                                        ERIC
 4178 CONTINUE                                                          ERIC
      IF(EI.EQ.16) GO TO 415                                            ERIC
      ESTMP=ALINT2(EAVTM,EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1)   ERIC
      ESMST=ALINT2(ATEN(3),EFLG,ET52,1.,ET62,EMN2)                      ERIC
      GO TO 4179                                                        ERIC
 415  CONTINUE                                                          ERIC
      ESTMP=ALINT2(AVSTM(1),EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1) ERIC
      ESMST=ALINT2(APEVA,EFLG,ET53,1.,ET63,EMN2)                        ERIC
 4179 CONTINUE                                                          ERIC
      ENFL(EI,6)=EN(EI,6)*(1.-ESFD*ESTMP*ESMST)*DT                      ERIC
      IF(ENFL(EI,6).GT.EN(EI,6)-ESMP(EI,6)) ENFL(EI,6)=EN(EI,6)-ESMP(EI, ERIC
     *6)                                                                ERIC
      IF(EI.EQ.6.OR.EI.EQ.8.OR.EI.EQ.16) GO TO 4180                     ERIC
      EK=EI+1                                                           ERIC
      IF(TIME.GT.ETIM2(EK,6)) ENFL(EI,6)=EN(EI,6)-ESMP(EI,6)            ERIC
 4180 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***TRANSFER FROM THE PREVIOUS CATEGORY OF TENEBRIONIDAE TO THE PRESENT  ERIC
      ENFIJ(EI,6)=0.                                                    ERIC
      EK=EI-1                                                           ERIC
      IF(EI.EQ.1) EK=16                                                 ERIC
      IF(EN(EK,6).LE.0.) ESUM(EK,6)=0.                                  ERIC
      IF(EN(EK,6).LE.0.) GO TO 4185                                     ERIC
      IF(TIME.LE.ETIM1(EI,6).OR.TIME.GE.ETIM2(EI,6)) GO TO 4185         ERIC
      IF(EI.NE.1) GO TO 4181                                            ERIC
      EGGP=EEPF(EMER(6),E12,EDGD1,EW(EK,6),EMW(6))                      ERIC
      ENFIJ(EI,6)=EN(EK,6)*EPPF(6)*EGGP*DT                              ERIC
      EREP=ENFIJ(EI,6)*EW(1,6)                                          ERIC
      GO TO 4185                                                        ERIC
 4181 CONTINUE                                                          ERIC
      IF(EI.GE.6.AND.EI.LE.10) GO TO 4182                               ERIC
      CALL ESTEP(ESUM(EK,6),EDG(EK,6),EA,EB)                            ERIC
      ESUM(EK,6)=EB                                                     ERIC
      ENFIJ(EI,6)=EN(EK,6)*EA                                           ERIC
      GO TO 4185                                                        ERIC
```

```
 4182 CONTINUE
      EX1=0.
      EX2=E12
      EY1=0.
      EY2=0.1
      IF(EI.NE.10) GO TO 4183
      IF(ESUM(EK,6).LT.EDG(EK,6)) GO TO 4185
 4183 CONTINUE
      IF(EI.NE.6.AND.EI.NE.8) GO TO 4184
      EY1=0.2
      EY2=0.
 4184 CONTINUE
      ENFIJ(EI,6)=EN(EK,6)*ALINT2(EDGD2,EFLG,EX1,EY1,EX2,EY2)*DT
 4185 CONTINUE
C***********************************************************************
C
C**********************************************************************,
      DO 4188 EI=1,16
C***RESPIRATION OF TENEBRIONIDAE AS FUNCTION OF WEIGHT, AMBIENT TEMPERAT
C*** AND ACTIVITY ******************************************************
      EFFI(EI,6)=0.
      EFIR(EI,6)=0.
      EK=6
      IF(EN(EI,6).LE.0.) GO TO 4187
      IF(EW(EI,6).LE.0.) GO TO 4187
      EFIR(EI,6)=ERSPR(ETMX2,ETMN2)*DT
      IF(EI.EQ.6.OR.EI.EQ.8) EFIR(EI,6)=EFIR(EI,6)/EIAF
      IF(EI.EQ.5.OR.EI.EQ.7.OR.EI.EQ.9) EFIR(EI,6)=EFIR(EI,6)*EACF
      IF(EI.EQ.16) EFIR(EI,6)=ERSPR(ETMX1,ETMN1)*EACF*DT
C***********************************************************************
C
C***********************************************************************
C***FOOD CONSUMPTION BY TENEBRIONIDAE **********************************
      IF(EI.NE.5.AND.EI.NE.7.AND.EI.NE.9) GO TO 4186
      ERPR=0.
      EJ=8
      EFFI(EI,6)=EFOD(ERPR,EDGD2)*DT
 4186 CONTINUE
      IF(EI.NE.16) GO TO 4187
      ERPR=EREP
      EFFI(16,6)=EN(16,6)*(1.3*ENW(16,6)-EW(16,6))+ERPR+EFIR(16,6)
      IF(EFFI(16,6).LT.0.) EFFI(16,6)=0.
 4187 CONTINUE
C***********************************************************************
C
C***********************************************************************
C***EXCRETION AS FRACTION OF FOOD INGESTED *****************************
      EFIE(EI,6)=EFFI(EI,6)*(1.-EASK(6))
 4188 CONTINUE
C***********************************************************************
C***********************************************************************
C***BEGINNING OF THE SPECIFIC COMPUTIONS FOR GRASSHOPPERS **************
C***********************************************************************
C***********************************************************************
      EREP=0.
      DO 4197 EI=1,16
      ESUM(EI,7)=ESUM(EI,7)+EDGD1*DT
C***NUMBER OF GRASSHOPPERS DYING FROM NON-PREDATORY CAUSES *************
      IF(EN(EI,7).LE.0.) GO TO 4195
      IF(EN(EI,7).LE.ESMP(EI,7)) GO TO 4195
      ESFD=1.
      IF(EW(EI,7).GT.EDW*ENW(EI,7)) GO TO 4192
      ESFD=1.-(EW(EI,7)-EDW*ENW(EI,7))*(EW(EI,7)-EDW*ENW(EI,7))/(EDW*EDW
     **ENW(EI,7)*ENW(EI,7))
 4192 CONTINUE
      IF(EI.EQ.1) GO TO 4193
      ESTMP=ALINT2(AVSTM(1),EFLG,ET12,EMN1,ET22,1.,ET32,0.995,ET41,EMN1)
      ESMST=ALINT2(APEVA,EFLG,ET53,1.,ET63,EMN2)
      GO TO 4194
 4193 CONTINUE
      ESTMP=ALINT2(EAVTM,EFLG,ET11,EMN1,ET21,1.,ET31,0.995,ET41,EMN1)
      ESMST=ALINT2(ATEN(3),EFLG,ET51,1.,ET61,EMN2)
 4194 CONTINUE
      ENFL(EI,7)=EN(EI,7)*(1.-ESFD*ESTMP*ESMST)*DT
```

```
      IF(ENFL(EI,7).GT.EN(EI,7)-ESMP(EI,7)) ENFL(EI,7)=EN(EI,7)-ESMP(EI, ERIC
     .7)                                                                ERIC
 4195 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***TRANSFER FROM THE PREVIOUS CATEGORY OF GRASSHOPPERS TO THE PRESENT * ERIC
      ENFIJ(EI,7)=0.                                                    ERIC
      EK=EI-1                                                           ERIC
      IF(EI.EQ.1) EK=16                                                 ERIC
      IF(EI.EQ.9) EK=1                                                  ERIC
      IF(EN(EK,7).LE.0.) ESUM(EK,7)=0.                                  ERIC
      IF(EN(EK,7).LE.0.) GO TO 4197                                     ERIC
      IF(TIME.LE.ETIM1(EI,7).OR.TIME.GE.ETIM2(EI,7)) GO TO 4197         ERIC
      IF(EI.NE.1) GO TO 4196                                            ERIC
      EGGP=EFPF(EMER(7),E12,EDGD1,EW(EK,7),EMW(7))                      ERIC
      ENFIJ(EI,7)=EN(EK,7)*EGGP*DT                                      ERIC
      EREP=ENFIJ(EI,7)*EW(1,7)                                          ERIC
      GO TO 4197                                                        ERIC
 4196 CONTINUE                                                          ERIC
      IF(ESUM(EK,7).LT.EDG(EK,7)) GO TO 4197                            ERIC
      EHAF=1.                                                           ERIC
      IF(EI.EQ.9) EHAF=EPPF(7)                                          ERIC
      IF(EI.EQ.2) EHAF=1.-EPPF(7)                                       ERIC
      EX1=0.                                                            ERIC
      EX2=E12                                                           ERIC
      EY1=0.                                                            ERIC
      EY2=0.15                                                          ERIC
      ENFIJ(EI,7)=EN(EK,7)*ALINT2(EDGD2,EFLG,EX1,EY1,EX2,EY2)*EHAF*DT   ERIC
 4197 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***RESPIRATION OF GRASSHOPPERS AS FUNCTION OF WEIGHT, AMBIENT TEMPERATU ERIC
C*** AND ACTIVITY ****************************************************** ERIC
      DO 4199 EI=1,16                                                   ERIC
      EFIR(EI,7)=0.                                                     ERIC
      EFFI(EI,7)=0.                                                     ERIC
      EK=7                                                              ERIC
      IF(EN(EI,7).LE.0.) GO TO 4198                                     ERIC
      IF(EW(EI,7).LE.0.) GO TO 4198                                     ERIC
      EFIR(EI,7)=ERSPR(ETMX1,ETMN1)*EACF*DT                             ERIC
      IF(EI.EQ.1) EFIR(EI,7)=ERSPR(ETMX2,ETMN2)/EIAF*DT                 ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***FEEDING BY GRASSHOPPERS ******************************************** ERIC
      IF(EI.EQ.1) GO TO 4198                                            ERIC
      EJ=9                                                              ERIC
      ERPR=0.                                                           ERIC
      IF(EI.EQ.16) ERPR=EREP                                            ERIC
      EFFI(EI,7)=EFOD(ERPR,EDGD1)*DT                                    ERIC
 4198 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C***EXCRETION BY TENEBRIONIDAE ***************************************** ERIC
      EFIE(EI,7)=EFFI(EI,7)*(1.-EASK(7))                                ERIC
 4199 CONTINUE                                                          ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
      RETURN                                                            ERIC
      END                                                               ERIC
C*********************************************************************** ERIC
C                                                                       ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
C*********************************************************************** ERIC
```

```
C***NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOTE-NOT ERIC
C***THE BLOCK OF CODING FOLLOWING BELOW WHICH DESCRIBES THE FLOWS , HAS  ERIC
C***TESTED BY COMPUTER RUNS BY ERIK FRAMSTAD (DATE JULY 14, 1975) ****** ERIC
C*********************************************************************** ERIC
(281-1).                                                                 ERIC
C***COMPUTING FLOW FROM LITTER COMPARTMENT TO SOURCE/SINK ************** ERIC
      ESM=X(280)+X(281)                                                  ERIC
      FLOW=EFFI(16,6)*X(281)/ESM                                         ERIC
(280-1).                                                                 ERIC
      FLOW=EFFI(16,6)*X(280)/ESM                                         ERIC
C***CHECKING THAT OUT-FLOWS DO NOT EXCEED AMOUNT IN CATEGORY *********** ERIC
      DO 4200 EJ=1,7                                                     ERIC
      DO 4200 EI=2,16                                                    ERIC
      EK=EI-1                                                            ERIC
      IF(EJ.EQ.2.AND.(EI.EQ.11.OR.EI.EQ.14)) GO TO 4200                  ERIC
      IF(EJ.EQ.4.AND.EI.EQ.7) GO TO 4200                                 ERIC
      IF(EJ.EQ.7.AND.(EI.EQ.2.OR.EI.EQ.9)) GO TO 4200                    ERIC
      IF(EN(EK,EJ)-ESMP(EK,EJ)-ENFL(EK,EJ).LT.ENFIJ(EI,EJ)) ENFIJ(EI,EJ) ERIC
     ==EN(EK,EJ)-ESMP(EK,EJ)-ENFL(EK,EJ)                                 ERIC
 4200 CONTINUE                                                           ERIC
      IF(EN(10,2)-ESMP(10,2)-ENFL(10,2).GE.ENFIJ(11,2)+ENFIJ(14,2)) GO   ERIC
     *TO 4201                                                            ERIC
      ENFIJ(11,2)=ENFIJ(11,2)*(EN(10,2)-ESMP(10,2)-ENFL(10,2))/(ENFIJ(11 ERIC
     .,2)+ENFIJ(14,2))                                                   ERIC
      ENFIJ(14,2)=ENFIJ(14,2)*(EN(10,2)-ESMP(10,2)-ENFL(10,2))/(ENFIJ(11 ERIC
     .,2)+ENFIJ(14,2))                                                   ERIC
 4201 CONTINUE                                                           ERIC
      IF(EN(1,7)-ESMP(1,7)-ENFL(1,7).GE.ENFIJ(2,7)+ENFIJ(9,7)) GO TO     ERIC
     *4202                                                               ERIC
      ENFIJ(2,7)=ENFIJ(2,7)*(EN(1,7)-ESMP(1,7)-ENFL(1,7))/(ENFIJ(2,7)+EN ERIC
     NFIJ(9,7))                                                          ERIC
      ENFIJ(9,7)=ENFIJ(9,7)*(EN(1,7)-ESMP(1,7)-ENFL(1,7))/(ENFIJ(2,7)+EN ERIC
     NFIJ(9,7))                                                          ERIC
 4202 CONTINUE                                                           ERIC
C***FLOWS FROM PLANT CATEGORIES TO SOURCE/SINK ************************* ERIC
C***FLOWS FROM LIVE SHOOTS ********************************************* ERIC
(EI=200,204-1).                                                          ERIC
      ESM=0.                                                             ERIC
      DO 4203 EK=1,9                                                     ERIC
      ESM=ESM+EFC(EI-199,EK)                                             ERIC
 4203 CONTINUE                                                           ERIC
      FLOW=ESM                                                           ERIC
C***FLOWS FROM CROWNS OR STORAGE ORGANS ******************************** ERIC
(EI=210,214-1).                                                          ERIC
      ESM=0.                                                             ERIC
      DO 4204 EK=1,9                                                     ERIC
      ESM=ESM+EFC(EI-204,EK)                                             ERIC
 4204 CONTINUE                                                           ERIC
      FLOW=ESM                                                           ERIC
C***FLOWS FROM SEEDS *************************************************** ERIC
(EI=230,234-1).                                                          ERIC
      ESM=0.                                                             ERIC
      DO 4205 EK=1,9                                                     ERIC
      ESM=ESM+EFC(EI-219,EK)                                             ERIC
 4205 CONTINUE                                                           ERIC
      FLOW=ESM                                                           ERIC
C***FLOWS FROM LIVE ROOTS ********************************************** ERIC
(EI=240,244-1).                                                          ERIC
      ESM=0.                                                             ERIC
      DO 4206 EK=1,9                                                     ERIC
      ESM=ESM+EFC(EI-224,EK)                                             ERIC
 4206 CONTINUE                                                           ERIC
      FLOW=ESM                                                           ERIC
C***FLOW FROM SOURCE/SINK TO LITTER ************************************ ERIC
(1-280).                                                                 ERIC
      ESM=0.                                                             ERIC
      ESM1=0.                                                            ERIC
      ESM2=0.                                                            ERIC
      ESM3=0.                                                            ERIC
C***FLOW OF ANIMAL MATTER ********************************************** ERIC
      DO 4208 EJ=1,7                                                     ERIC
      DO 4208 EI=1,16                                                    ERIC
```

```
      ESM=ESM+ENFP(EI.EJ.3)*ECUT(3)/(1.+ECUT(3))+ENFP(EI.EJ.4)*ECUT(4)/( ERIC
     (1.+ECUT(4))                                                       ERIC
 4208 CONTINUE                                                          ERIC
C***FLOW OF PLANT MATTER ********************************************** ERIC
      DO 4207 EK=1.9                                                    ERIC
      DO 4207 EI=1.10                                                   ERIC
      ESM1=ESM1+EFC(EI.EK)*ECUT(EM(EK))/(1.+ECUT(EM(EK)))               ERIC
 4207 CONTINUE                                                          ERIC
      DO 4209 EK=1.9                                                    ERIC
      DO 4209 EI=11.15                                                  ERIC
      ESM2=ESM2+EFC(EI.EK)*ECUT(EM(EK))/(1.+ECUT(EM(EK)))               ERIC
 4209 CONTINUE                                                          ERIC
      DO 4210 EK=1.9                                                    ERIC
      DO 4210 EI=16.20                                                  ERIC
      ESM3=ESM3+EFC(EI.EK)*ECUT(EM(EK))/(1.+ECUT(EM(EK)))               ERIC
 4210 CONTINUE                                                          ERIC
      FLOW=(1.-DLABG)*ESM+(1.-DLABI)*ESM1+(1.-DLASE)*ESM2+(1.-DLART(1))* ERIC
     *ESM3                                                              ERIC
(1-281).                                                                ERIC
      FLOW=DLABG*ESM+DLABI*ESM1+DLASE*ESM2+DLART(1)*ESM3                ERIC
C***FLOW OF EXCRETORY PRODUCTS FROM A GROUP (I.J) OF INSECTS TO FAECES D ERIC
(EK=320.399-499).                                                       ERIC
      EJ=(EK-304)/16                                                    ERIC
      EI=EK-303-EJ*16                                                   ERIC
      FLOW=DLABF*EFIE(EI.EJ)                                            ERIC
(EK=520.551-499).                                                       ERIC
      EJ=(EK-42 )/16                                                    ERIC
      EI=EK-423-EJ*16                                                   ERIC
      FLOW=DLABF*EFIE(EI.EJ)                                            ERIC
(EK=320.399-498).                                                       ERIC
      EJ=(EK-304)/16                                                    ERIC
      EI=EK-303-EJ*16                                                   ERIC
      FLOW=(1.-DLABF)*EFIE(EI.EJ)                                       ERIC
(EK=520.551-498).                                                       ERIC
      EJ=(EK-424)/16                                                    ERIC
      EI=EK-423-EJ*16                                                   ERIC
      FLOW=(1.-DLABF)*EFIE(EI.EJ)                                       ERIC
C***FLOW OF DEAD INSECT BIOMASS FROM GROUP (I.J) TO LITTER ************* ERIC
(EK=320.399-280).                                                       ERIC
      EJ=(EK-304)/16                                                    ERIC
      EI=EK-303-EJ*16                                                   ERIC
      FLOW=(1.-DLABG)*ENFL(EI.EJ)*EW(EI.EJ)                             ERIC
(EK=320.399-281).                                                       ERIC
      EJ=(EK-304)/16                                                    ERIC
      EI=EK-303-EJ*16                                                   ERIC
      FLOW=DLABG*ENFL(EI.EJ)*EW(EI.EJ)                                  ERIC
(EK=520.551-280).                                                       ERIC
      EJ=(EK-424)/16                                                    ERIC
      EI=EK-423-EJ*16                                                   ERIC
      FLOW=(1.-DLABG)*ENFL(EI.EJ)*EW(EI.EJ)                             ERIC
(EK=520.551-281).                                                       ERIC
      EJ=(EK-424)/16                                                    ERIC
      EI=EK-423-EJ*16                                                   ERIC
      FLOW=DLABG*ENFL(EI.EJ)*EW(EI.EJ)                                  ERIC
C***FLOW OF INSECT BIOMASS EATEN BY PREDATORS AND GRAMS CARBON RESPIRED  ERIC
C***GROUP (I.J) TO SOURCE/SINK ****************************************** ERIC
(EK=320.399-1).                                                         ERIC
      EJ=(EK-304)/16                                                    ERIC
      EI=EK-303-EJ*16                                                   ERIC
      FLOW=EFIR(EI.EJ)+ESMP(EI.EJ)*EW(EI.EJ)                            ERIC
(EK=520.551-1).                                                         ERIC
      EJ=(EK-424)/16                                                    ERIC
      EI=EK-423-EJ*16                                                   ERIC
      FLOW=EFIR(EI.EJ)+ESMP(EI.EJ)*EW(EI.EJ)                            ERIC
C***FEEDING FLOW FROM SOURCE/SINK TO INSECT GROUP (I.J) **************** ERIC
(1-EK=320.399).                                                         ERIC
      EJ=(EK-304)/16                                                    ERIC
      EI=EK-303-EJ*16                                                   ERIC
      FLOW=EFFI(EI.EJ)                                                  ERIC
(1-EK=520.551).                                                         ERIC
      EJ=(EK-424)/16                                                    ERIC
      EI=EK-423-EJ*16                                                   ERIC
      FLOW=EFFI(EI.EJ)                                                  ERIC
```

```
C***FLOW OF INSECT BIOMASS FROM GROUP (I-1,J) TO GROUP(I,J) ************
C***FLOW FOR CARABIDAE *************************************************
(335-320).
      FLOW=ENFIJ(1,1)*ENW1(1,1)
(EK=320,334-EL=1*EK+1).
      EI=EL-319
      FLOW=ENFIJ(EI,1)*EW(EI-1,1)
C***FLOW FOR ARANEIDA **************************************************
(351-336).
      FLOW=ENFIJ(1,2)*ENW1(1,2)
(EK=336,347-EL=1*EK+1).
      EI=EL-335
      FLOW=ENFIJ(EI,2)*EW(EI-1,2)
(345-349).
      FLOW=ENFIJ(14,2)*EW(10,2)
(EK=349,350-EL=1*EK+1).
      EI=EL-335
      FLOW=ENFIJ(EI,2)*EW(EI-1,2)
C***FLOW FOR CHRYSOMELIDAE/CURCULIONIDAE *******************************
(367-352).
      FLOW=ENFIJ(1,3)*ENW1(1,3)
(EK=352,366-EL=1*EK+1).
      EI=EL-351
      FLOW=ENFIJ(EI,3)*EW(EI-1,3)
C***FLOW FOR HEMIPTERA/HOMOPTERA ***************************************
(383-368).
      FLOW=ENFIJ(1,4)*ENW1(1,4)
(EK=368,382-EL=1*EK+1).
      EI=EL-367
      FLOW=ENFIJ(EI,4)*EW(EI-1,4)
      IF(EI.EQ.7) FLOW=ENFIJ(7,4)*ENW1(7,4)
C***FLOW FOR SCARABAEIDAE **********************************************
(399-384).
      FLOW=ENFIJ(1,5)*ENW1(1,5)
(EK=384,398-EL=1*EK+1).
      EI=EL-383
      FLOW=ENFIJ(EI,5)*EW(EI-1,5)
C***FLOW FOR TENEBRIONIDAE *********************************************
(535-520).
      FLOW=ENFIJ(1,6)*ENW1(1,6)
(EK=520,534-EL=1*EK+1).
      EI=EL-519
      FLOW=ENFIJ(EI,6)*EW(EI-1,6)
C***FLOW FOR ORTHOPTERA ************************************************
(551-536).
      FLOW=ENFIJ(1,7)*ENW1(1,7)
(EK=536,542-EL=1*EK+1).
      EI=EL-535
      FLOW=ENFIJ(EI,7)*EW(EI-1,7)
(536-544).
      FLOW=ENFIJ(9,7)*EW(1,7)
(EK=544,551-EL=1*EK+1).
      EI=EL-535
      FLOW=ENFIJ(EI,7)*EW(EI-1,7)
C************* OG DET VAR DET ******************************************
C***********************************************************************
C***********************************************************************
C***********************************************************************
```

APPENDIX F

PARAMETER VALUES

```
TSTRT=1.$
TEND=364.$                                                          DATA       2
DTFL=5.$                                                            DATA       3
DT=2.$                                                              DATA       4
DTPL=2.$                                                            DATA       5
DTPR=80.$                                                           DATA       6
X(1)=999*0.$                                                        DATA       7
X(1)=16*0.$                                                         DATA       8
X(6)=250.$                                                          DATA       9
X(15)=240.$                                                         DATA      10
X(17)=0.$                                                           DATA      11
X(18)=0.$                                                           DATA      12
EACF=3.$                                                            DATA      13
FASK(1)=0.7$                                                        DATA      14
                                                                    DATA      15
FASK(2)=0.9$                                                        DATA      16
FASK(3)=0.85$                                                       DATA      17
FASK(4)=0.85$                                                       DATA      18
FASK(5)=0.7$                                                        DATA      19
FASK(6)=0.65$                                                       DATA      20
FASK(7)=0.5$                                                        DATA      21
ECUT(1)=0.2$                                                        DATA      22
FCUT(2)=0.5$                                                        DATA      23
FCUT(3)=0.05$                                                       DATA      24
ECUT(4)=1.$                                                         DATA      25
FCUT(5)=0.3$                                                        DATA      26
FCUT(6)=0.5$                                                        DATA      27
ECUT(7)=3.$                                                         DATA      28
FDG(1.1)=4*60.$                                                     DATA      29
FDG(5.1)=3*900.$                                                    DATA      30
FDG(8.1)=6*105.$                                                    DATA      31
FDG(14.1)=3*0.$                                                     DATA      32
FDG(1.2)=6*50.$                                                     DATA      33
FDG(7.2)=4*90.$                                                     DATA      34
FDG(11.2)=6*120.$                                                   DATA      35
FDG(1.3)=6*40.$                                                     DATA      36
FDG(7.3)=3*110.$                                                    DATA      37
                                                                    DATA      38
FDG(14.3)=3*1000.$                                                  DATA      39
FDG(1.4)=150.$                                                      DATA      40
FDG(2.4)=15*120.$                                                   DATA      41
FDG(7.4)=5*50.$                                                     DATA      42
FDG(12.4)=5*100.$                                                   DATA      43
FDG(1.5)=16*90.$                                                    DATA      44
FDG(1.6)=8*50.$                                                     DATA      45
FDG(9.6)=500.$                                                      DATA      46
FDG(10.6)=7*60.$                                                    DATA      47
FDG(1.7)=250.$                                                      DATA      48
FDG(2.7)=14*150.$                                                   DATA      49
FDG(16.7)=300.$                                                     DATA      50
FDU1=0.4$                                                           DATA      51
FDU6=0.25$                                                          DATA      52
FDU7=0.15$                                                          DATA      53
FDW=0.7$                                                            DATA      54
FDZ1=10.$                                                           DATA      55
FDZ2=7.$                                                            DATA      56
FIAF=20.$                                                           DATA      57
FM(1)=1$                                                            DATA      58
FM(2)=1$                                                            DATA      59
FM(3)=3$                                                            DATA      60
FM(4)=3$                                                            DATA      61
FM(5)=4$                                                            DATA      62
FM(6)=5$                                                            DATA      63
FM(7)=5$                                                            DATA      64
FM(8)=6$                                                            DATA      65
```

```
EM(9)=7$                                                         DATA     66
EMFH(1)=7*2.$                                                    DATA     67
EMN1=0.90$                                                       DATA     68
EMN2=0.95$                                                       DATA     69
EMW(1)=0.00384$                                                  DATA     70
EMW(2)=0.0016$                                                   DATA     71
EMW(3)=0.000336$                                                 DATA     72
EMW(4)=0.00016$                                                  DATA     73
EMW(5)=0.0009$                                                   DATA     74
EMW(6)=0.00128$                                                  DATA     75
EMW(7)=0.019$                                                    DATA     76
EN(1.1)=16*0.$                                                   DATA     77
EN(6.1)=5.$                                                      DATA     78
EN(15.1)=30.$                                                    DATA     79
EN(1.2)=16*0.$                                                   DATA     80
EN(10.2)=2.$                                                     DATA     81
EN(1.3)=16*0.$                                                   DATA     82
EN(15.3)=800.$                                                   DATA     83
EN(1.4)=2000.$                                                   DATA     84
EN(2.4)=15*0.$                                                   DATA     85
      EN(1.5)=16*0.$                                             DATA     86
EN(6.5)=800.$                                                    DATA     87
EN(8.5)=600.$                                                    DATA     88
EN(10.5)=400.$                                                   DATA     89
EN(1.6)=16*0.$                                                   DATA     90
EN(6.6)=600.$                                                    DATA     91
EN(8.6)=400.$                                                    DATA     92
EN(1.7)=1000.$                                                   DATA     93
EN(2.7)=15*0.$                                                   DATA     94
ENFIJ(1.17)=0.$                                                  DATA     95
ENW1(1.1)=5*0.0005$                                              DATA     96
ENW1(6.1)=2*0.002$                                               DATA     97
ENW1(8.1)=0.0056$                                                DATA     98
ENW1(9.1)=0.0054$                                                DATA     99
ENW1(10.1)=0.0052$                                               DATA    100
ENW1(11.1)=0.005$                                                DATA    101
ENW1(12.1)=0.0046$                                               DATA    102
ENW1(13.1)=0.0046$                                               DATA    103
ENW1(14.1)=0.004$                                                DATA    104
ENW1(15.1)=2*0.0048$                                             DATA    105
ENW1(1.2)=7*0.00002$                                             DATA    106
ENW1(8.2)=0.00015$                                               DATA    107
ENW1(9.2)=0.00028$                                               DATA    108
ENW1(10.2)=0.0004$                                               DATA    109
ENW1(11.2)=0.0003$                                               DATA    110
ENW1(12.2)=0.0008$                                               DATA    111
ENW1(13.2)=0.0013$                                               DATA    112
ENW1(14.2)=0.0003$                                               DATA    113
ENW1(15.2)=0.0011$                                               DATA    114
ENW1(16.2)=0.002$                                                DATA    115
ENW1(1.3)=7*0.00002$                                             DATA    116
ENW1(8.3)=0.00008$                                               DATA    117
ENW1(9.3)=0.00014$                                               DATA    118
ENW1(10.3)=5*0.0002$                                             DATA    119
ENW1(15.3)=2*0.00042$                                            DATA    120
ENW1(1.4)=2*0.00001$                                             DATA    121
ENW1(3.4)=0.00005$                                               DATA    122
ENW1(4.4)=0.0001$                                                DATA    123
ENW1(5.4)=0.00015$                                               DATA    124
ENW1(6.4)=0.0002$                                                DATA    125
ENW1(7.4)=6*0.00001$                                             DATA    126
ENW1(13.4)=0.00005$                                              DATA    127
ENW1(14.4)=0.0001$                                               DATA    128
ENW1(15.4)=0.00015$                                              DATA    129
ENW1(16.4)=0.0002$                                               DATA    130
ENW1(1.5)=5*0.00005$                                             DATA    131
ENW1(6.5)=2*0.0002$                                              DATA    132
ENW1(8.5)=2*0.0004$                                              DATA    133
ENW1(10.5)=2*0.0007$                                             DATA    134
ENW1(12.5)=5*0.001$                                              DATA    135
```

```
FNW1(1,6)=5*0.0001$                                 DATA     136
FNW1(6,6)=2*0.0006$                                 DATA     137
FNW1(8,6)=2*0.0011$                                 DATA     138
FNW1(10,6)=7*0.0016$                                DATA     139
FNW1(1,7)=2*0.0012$                                 DATA     140
FNW1(3,7)=0.0032$                                   DATA     141
FNW1(4,7)=0.0053$                                   DATA     142
FNW1(5,7)=0.0074$                                   DATA     143
FNW1(6,7)=0.0096$                                   DATA     144
FNW1(7,7)=0.0118$                                   DATA     145
FNW1(8,7)=0.014$                                    DATA     146
FNW1(9,7)=0.0012$                                   DATA     147
FNW1(10,7)=0.0042$                                  DATA     148
FNW1(11,7)=0.0073$                                  DATA     149
FNW1(12,7)=0.0105$                                  DATA     150
FNW1(13,7)=0.0138$                                  DATA     151
FNW1(14,7)=0.0171$                                  DATA     152
FNW1(15,7)=0.0205$                                  DATA     153
FNW1(16,7)=0.024$                                   DATA     154
FNW2(1,1)=4*0.0005$                                 DATA     155
FNW2(5,1)=2*0.002$                                  DATA     156
FNW2(7,1)=0.0056$                                   DATA     157
FNW2(8,1)=0.0054$                                   DATA     158
FNW2(9,1)=0.0052$                                   DATA     159
FNW2(10,1)=0.005$                                   DATA     160
FNW2(11,1)=0.0048$                                  DATA     161
FNW2(12,1)=0.0046$                                  DATA     162
FNW2(13,1)=0.0044$                                  DATA     163
FNW2(14,1)=3*0.0048$                                DATA     164
FNW2(1,2)=6*0.00002$                                DATA     165
FNW2(7,2)=0.00015$                                  DATA     166
FNW2(8,2)=0.00028$                                  DATA     167
FNW2(9,2)=0.0004$                                   DATA     168
FNW2(10,2)=0.0003$                                  DATA     169
FNW2(11,2)=0.0008$                                  DATA     170
FNW2(12,2)=0.0013$                                  DATA     171
FNW2(13,2)=0.0013$                                  DATA     172
FNW2(14,2)=0.0011$                                  DATA     173
FNW2(15,2)=0.002$                                   DATA     174
FNW2(16,2)=0.0025$                                  DATA     175
FNW2(1,3)=6*0.00002$                                DATA     176
FNW2(7,3)=0.00008$                                  DATA     177
FNW2(8,3)=0.00014$                                  DATA     178
FNW2(9,3)=5*0.0002$                                 DATA     179
FNW2(14,3)=3*0.00042$                               DATA     180
FNW2(1,4)=0.00001$                                  DATA     181
FNW2(2,4)=0.00005$                                  DATA     182
FNW2(3,4)=0.0001$                                   DATA     183
FNW2(4,4)=0.00015$                                  DATA     184
FNW2(5,4)=2*0.0002$                                 DATA     185
FNW2(7,4)=5*0.00001$                                DATA     186
FNW2(12,4)=0.00005$                                 DATA     187
FNW2(13,4)=0.0001$                                  DATA     188
FNW2(14,4)=0.00015$                                 DATA     189
FNW2(15,4)=2*0.0002$                                DATA     190
FNW2(1,5)=4*0.00005$                                DATA     191
FNW2(5,5)=2*0.0002$                                 DATA     192
FNW2(7,5)=2*0.0004$                                 DATA     193
FNW2(9,5)=2*0.0007$                                 DATA     194
FNW2(11,5)=5*0.001$                                 DATA     195
FNW2(16,5)=0.0012$                                  DATA     196
FNW2(1,6)=4*0.0001$                                 DATA     197
FNW2(5,6)=2*0.0006$                                 DATA     198
FNW2(7,6)=2*0.0011$                                 DATA     199
FNW2(9,6)=8*0.0016$                                 DATA     200
```

```
FNW2(1,7)=0.0012$                                                   DATA      201
FNW2(2,7)=0.0032$                                                   DATA      202
FNW2(3,7)=0.0053$                                                   DATA      203
FNW2(4,7)=0.0074$                                                   DATA      204
FNW2(5,7)=0.0096$                                                   DATA      205
FNW2(6,7)=0.0116$                                                   DATA      206
FNW2(7,7)=2*0.014$                                                  DATA      207
FNW2(9,7)=0.0042$                                                   DATA      208
FNW2(10,7)=0.0073$                                                  DATA      209
FNW2(11,7)=0.0105$                                                  DATA      210
FNW2(12,7)=0.0138$                                                  DATA      211
FNW2(13,7)=0.0171$                                                  DATA      212
FNW2(14,7)=0.0205$                                                  DATA      213
FNW2(15,7)=2*0.024$                                                 DATA      214
FPAD1=1.$                                                           DATA      215
FPAD2=0.7$                                                          DATA      216
FPAD3=0.45$                                                         DATA      217
FPAD5=0.2$                                                          DATA      218
FPAD6=0.7$                                                          DATA      219
FPPF(1)=7*0.5$                                                      DATA      220
FPPRF(1,1,1)=4*8.$                                                  DATA      221
FPPRF(5,1,1)=1.$                                                    DATA      222
FPPRF(6,1,1)=10.$                                                   DATA      223
FPPRF(7,1,1)=1.$                                                    DATA      224
FPPRF(8,1,1)=8*10.$                                                 DATA      225
FPPRF(14,1,1)=4.$                                                   DATA      226
FPPRF(16,1,1)=1.$                                                   DATA      227
FPPRF(1,2,1)=16*0.$                                                 DATA      228
FPPRF(10,2,1)=10.$                                                  DATA      229
FPPRF(1,3,1)=6*9.$                                                  DATA      230
FPPRF(7,3,1)=3*10.$                                                 DATA      231
FPPRF(10,3,1)=6*9.$                                                 DATA      232
FPPRF(14,3,1)=0.$                                                   DATA      233
FPPRF(16,3,1)=0.$                                                   DATA      234
FPPRF(1,4,1)=16*0.$                                                 DATA      235
FPPRF(1,5,1)=4*8.$                                                  DATA      236
FPPRF(5,5,1)=4.$                                                    DATA      237
FPPRF(6,5,1)=10*10.$                                                DATA      238
FPPRF(7,5,1)=4.$                                                    DATA      239
FPPRF(9,5,1)=4.$                                                    DATA      240
FPPRF(11,5,1)=4.$                                                   DATA      241
FPPRF(16,5,1)=1.$                                                   DATA      242
FPPRF(1,6,1)=4*8.$                                                  DATA      243
FPPRF(5,6,1)=4.$                                                    DATA      244
FPPRF(6,6,1)=10*10.$                                                DATA      245
FPPRF(7,6,1)=4.$                                                    DATA      246
FPPRF(9,6,1)=4.$                                                    DATA      247
FPPRF(16,6,1)=1.$                                                   DATA      248
FPPRF(1,7,1)=10.$                                                   DATA      249
FPPRF(2,7,1)=15*0.$                                                 DATA      250
FPPRF(1,1,2)=15*0.$                                                 DATA      251
FPPRF(14,1,2)=6.$                                                   DATA      252
FPPRF(16,1,2)=6.$                                                   DATA      253
FPPRF(1,2,2)=6*1.$                                                  DATA      254
FPPRF(7,2,2)=3*4.$                                                  DATA      255
FPPRF(10,2,2)=0.$                                                   DATA      256
FPPRF(11,2,2)=6*2.$                                                 DATA      257
FPPRF(1,3,2)=15*0.$                                                 DATA      258
FPPRF(14,3,2)=8.$                                                   DATA      259
FPPRF(16,3,2)=8.$                                                   DATA      260
FPPRF(1,4,2)=0.$                                                    DATA      261
FPPRF(2,4,2)=15*8.$                                                 DATA      262
FPPRF(7,4,2)=5*0.$                                                  DATA      263
FPPRF(1,5,2)=15*0.$                                                 DATA      264
FPPRF(16,5,2)=7.$                                                   DATA      265
FPPRF(1,6,2)=15*0.$                                                 DATA      266
FPPRF(16,6,2)=7.$                                                   DATA      267
FPPRF(1,7,2)=0.$                                                    DATA      268
FPPRF(2,7,2)=15*10.$                                                DATA      269
FPPRF(1,1,3)=4*8.$                                                  DATA      270
```

```
FPPHF(5,1,3)=1.$                                                        DATA     271
FPPHF(6,1,3)=10.$                                                       DATA     272
FPPHF(7,1,3)=1.$                                                        DATA     273
FPPHF(8,1,3)=8*10.$                                                     DATA     274
FPPHF(14,1,3)=4.$                                                       DATA     275
FPPHF(16,1,3)=1.$                                                       DATA     276
FPPHF(1,2,3)=16*0.$                                                     DATA     277
FPPHF(10,2,3)=10.$                                                      DATA     278
FPPHF(1,3,3)=6*8.$                                                      DATA     279
FPPHF(7,3,3)=3*10.$                                                     DATA     280
FPPHF(10,3,3)=6*9.$                                                     DATA     281
FPPHF(14,3,3)=0.$                                                       DATA     282
FPPHF(16,3,3)=0.$                                                       DATA     283
FPPHF(1,4,3)=16*0.$                                                     DATA     284
FPPHF(1,5,3)=4*8.$                                                      DATA     285
FPPHF(5,5,3)=4.$                                                        DATA     286
FPPHF(6,5,3)=10*10.$                                                    DATA     287
FPPHF(7,5,3)=4.$                                                        DATA     288
FPPHF(9,5,3)=4.$                                                        DATA     289
FPPHF(11,5,3)=4.$                                                       DATA     290
FPPHF(16,5,3)=1.$                                                       DATA     291
FPPHF(1,6,3)=4*8.$                                                      DATA     292
FPPHF(5,6,3)=4.$                                                        DATA     293
FPPHF(6,6,3)=10*10.$                                                    DATA     294
FPPHF(7,6,3)=4.$                                                        DATA     295
FPPHF(9,6,3)=4.$                                                        DATA     296
FPPHF(16,6,3)=1.$                                                       DATA     297
FPPHF(1,7,3)=10.$                                                       DATA     298
FPPHF(2,7,3)=15*0.$                                                     DATA     299
FPPHF(1,1,4)=15*0.$                                                     DATA     300
FPPHF(14,1,4)=8.$                                                       DATA     301
FPPHF(16,1,4)=8.$                                                       DATA     302
FPPHF(1,2,4)=6*1.$                                                      DATA     303
FPPHF(7,2,4)=3*4.$                                                      DATA     304
FPPHF(10,2,4)=0.$                                                       DATA     305
FPPHF(11,2,4)=3*3.$                                                     DATA     306
FPPHF(14,2,4)=3*2.$                                                     DATA     307
FPPHF(1,3,4)=15*0.$                                                     DATA     308
FPPHF(14,3,4)=10.$                                                      DATA     309
FPPHF(16,3,4)=10.$                                                      DATA     310
FPPHF(1,4,4)=1.$                                                        DATA     311
FPPHF(2,4,4)=15*8.$                                                     DATA     312
FPPHF(7,4,4)=5*1.$                                                      DATA     313
FPPHF(1,5,4)=15*0.$                                                     DATA     314
FPPHF(16,5,4)=8.$                                                       DATA     315
FPPHF(1,6,4)=15*0.$                                                     DATA     316
FPPHF(16,6,4)=8.$                                                       DATA     317
FPPHF(1,7,4)=0.$                                                        DATA     318
FPPHF(2,7,4)=15*10.$                                                    DATA     319
FPPHF(1,1,5)=4*10.$                                                     DATA     320
FPPHF(5,1,5)=0.$                                                        DATA     321
FPPHF(6,1,5)=8*3.$                                                      DATA     322
FPPHF(7,1,5)=0.$                                                        DATA     323
FPPHF(14,1,5)=1.$                                                       DATA     324
FPPHF(15,1,5)=2.$                                                       DATA     325
FPPHF(16,1,5)=0.$                                                       DATA     326
FPPHF(1,2,5)=9*1.$                                                      DATA     327
FPPHF(10,2,5)=3.$                                                       DATA     328
FPPHF(11,2,5)=6*0.$                                                     DATA     329
FPPHF(1,3,5)=6*10.$                                                     DATA     330
FPPHF(7,3,5)=3*2.$                                                      DATA     331
FPPHF(10,3,5)=6*4.$                                                     DATA     332
FPPHF(14,3,5)=1.$                                                       DATA     333
FPPHF(16,3,5)=1.$                                                       DATA     334
FPPHF(1,4,5)=3.$                                                        DATA     335
FPPHF(2,4,5)=15*2.$                                                     DATA     336
FPPHF(7,4,5)=5*3.$                                                      DATA     337
FPPHF(1,5,5)=4*10.$                                                     DATA     338
FPPHF(5,5,5)=1.$                                                        DATA     339
FPPHF(6,5,5)=10*3.$                                                     DATA     340
```

```
FPPHF(7,5,5)=1.$                    DATA   341
FPPHF(9,5,5)=1.$                    DATA   342
FPPHF(11,5,5)=1.$                   DATA   343
FPPHF(16,5,5)=0.$                   DATA   344
FPPHF(1,6,5)=4*10.$                 DATA   345
FPPHF(5,6,5)=1.$                    DATA   346
FPPHF(6,6,5)=10*3.$                 DATA   347
FPPHF(7,6,5)=1.$                    DATA   348
FPPHF(9,6,5)=1.$                    DATA   349
FPPHF(16,6,5)=0.$                   DATA   350
FPPHF(1,7,5)=10.$                   DATA   351
FPPHF(2,7,5)=15*0.$                 DATA   352
FPPHF(1,1,6)=4*1.$                  DATA   353
FPPHF(5,1,6)=4.$                    DATA   354
FPPHF(6,1,6)=10*2.$                 DATA   355
FPPHF(7,1,6)=4.$                    DATA   356
FPPHF(14,1,6)=4.$                   DATA   357
FPPHF(16,1,6)=10.$                  DATA   358
FPPHF(1,2,6)=16*6.$                 DATA   359
FPPHF(1,3,6)=6*1.$                  DATA   360
FPPHF(7,3,6)=3*2.$                  DATA   361
FPPHF(10,3,6)=4*1.$                 DATA   362
FPPHF(14,3,6)=10.$                  DATA   363
FPPHF(15,3,6)=1.$                   DATA   364
FPPHF(16,3,6)=10.$                  DATA   365
FPPHF(1,4,6)=3.$                    DATA   366
FPPHF(2,4,6)=15*10.$                DATA   367
FPPHF(7,4,6)=5*3.$                  DATA   368
FPPHF(1,5,6)=4*1.$                  DATA   369
FPPHF(5,5,6)=4.$                    DATA   370
FPPHF(6,5,6)=10*2.$                 DATA   371
FPPHF(7,5,6)=4.$                    DATA   372
FPPHF(9,5,6)=4.$                    DATA   373
FPPHF(11,5,6)=4.$                   DATA   374
FPPHF(16,5,6)=10.$                  DATA   375
FPPHF(1,6,6)=4*1.$                  DATA   376
FPPHF(5,6,6)=4.$                    DATA   377
FPPHF(6,6,6)=10*2.$                 DATA   378
FPPHF(7,6,6)=4.$                    DATA   379
FPPHF(9,6,6)=4.$                    DATA   380
FPPHF(16,6,6)=10.$                  DATA   381
FPPHF(1,7,6)=1.$                    DATA   382
FPPHF(2,7,6)=15*10.$                DATA   383
FPPHF(1,1,7)=4*1.$                  DATA   384
FPPHF(5,1,7)=4.$                    DATA   385
FPPHF(6,1,7)=10*2.$                 DATA   386
FPPHF(7,1,7)=4.$                    DATA   387
FPPHF(14,1,7)=4.$                   DATA   388
FPPHF(16,1,7)=10.$                  DATA   389
FPPHF(1,2,7)=16*6.$                 DATA   390
FPPHF(1,3,7)=6*1.$                  DATA   391
FPPHF(7,3,7)=3*2.$                  DATA   392
FPPHF(10,3,7)=4*1.$                 DATA   393
FPPHF(14,3,7)=10.$                  DATA   394
FPPHF(15,3,7)=1.$                   DATA   395
FPPHF(16,3,7)=10.$                  DATA   396
FPPHF(1,4,7)=3.$                    DATA   397
FPPHF(2,4,7)=15*10.$                DATA   398
FPPHF(7,4,7)=5*3.$                  DATA   399
FPPHF(1,5,7)=4*1.$                  DATA   400
FPPHF(5,5,7)=4.$                    DATA   401
FPPHF(6,5,7)=10*2.$                 DATA   402
FPPHF(7,5,7)=4.$                    DATA   403
FPPHF(9,5,7)=4.$                    DATA   404
FPPHF(11,5,7)=4.$                   DATA   405
FPPHF(16,5,7)=10.$                  DATA   406
FPPHF(1,6,7)=4*1.$                  DATA   407
FPPHF(5,6,7)=4.$                    DATA   408
FPPHF(6,6,7)=10*2.$                 DATA   409
FPPHF(7,6,7)=4.$                    DATA   410
```

```
FPPRF(9,6,7)=4.$                                                  DATA    411
FPPRF(16,6,7)=10.$                                                DATA    412
FPPRF(1,7,7)=1.$                                                  DATA    413
FPPRF(2,7,7)=15*10.$                                              DATA    414
FPRD1=0.03$                                                       DATA    415
FPRD2=0.0012$                                                     DATA    416
FPRD5=0.012$                                                      DATA    417
FPRD6=0.0005$                                                     DATA    418
FPRF(1,1)=3*1.$                                                   DATA    419
FPRF(4,1)=12*0.$                                                  DATA    420
FPRF(16,1)=3*10.$                                                 DATA    421
FPRF(19,1)=5.$                                                    DATA    422
FPRF(20,1)=7.$                                                    DATA    423
FPRF(1,2)=10.$                                                    DATA    424
FPRF(2,2)=8.$                                                     DATA    425
FPRF(3,2)=9.$                                                     DATA    426
FPRF(4,2)=7.$                                                     DATA    427
FPRF(5,2)=1.$                                                     DATA    428
FPRF(6,2)=5.$                                                     DATA    429
FPRF(7,2)=4.$                                                     DATA    430
FPRF(8,2)=6.$                                                     DATA    431
FPRF(9,2)=7*0.$                                                   DATA    432
FPRF(16,2)=3*2.$                                                  DATA    433
FPRF(19,2)=0.$                                                    DATA    434
FPRF(20,2)=1.$                                                    DATA    435
FPRF(1,3)=15*0.$                                                  DATA    436
FPRF(16,3)=10.$                                                   DATA    437
FPRF(17,3)=2*9.$                                                  DATA    438
FPRF(19,3)=2*5.$                                                  DATA    439
FPRF(1,4)=10.$                                                    DATA    440
FPRF(2,4)=2*9.$                                                   DATA    441
FPRF(4,4)=2*5.$                                                   DATA    442
FPRF(6,4)=3*8.$                                                   DATA    443
FPRF(9,4)=2*3.$                                                   DATA    444
FPRF(11,4)=10*0.$                                                 DATA    445
FPRF(1,5)=7.$                                                     DATA    446
FPRF(2,5)=9.$                                                     DATA    447
FPRF(3,5)=10.$                                                    DATA    448
FPRF(4,5)=5.$                                                     DATA    449
FPRF(5,5)=6.$                                                     DATA    450
FPRF(6,5)=2*8.$                                                   DATA    451
FPRF(8,5)=1.$                                                     DATA    452
FPRF(9,5)=12*0.$                                                  DATA    453
FPRF(1,6)=15*0.$                                                  DATA    454
FPRF(16,6)=8.$                                                    DATA    455
FPRF(17,6)=9.$                                                    DATA    456
FPRF(18,6)=10.$                                                   DATA    457
FPRF(19,6)=8.$                                                    DATA    458
FPRF(20,6)=5.$                                                    DATA    459
FPRF(1,7)=7.$                                                     DATA    460
FPRF(2,7)=8.$                                                     DATA    461
FPRF(3,7)=10.$                                                    DATA    462
FPRF(4,7)=8.$                                                     DATA    463
FPRF(5,7)=6*3.$                                                   DATA    464
FPRF(11,7)=5*0.$                                                  DATA    465
FPRF(16,7)=5*4.$                                                  DATA    466
FPRF(1,8)=4*3.$                                                   DATA    467
FPRF(5,8)=4*1.$                                                   DATA    468
FPRF(9,8)=7*0.$                                                   DATA    469
FPRF(16,8)=5*8.$                                                  DATA    470
FPRF(1,9)=7.$                                                     DATA    471
FPRF(2,9)=8.$                                                     DATA    472
FPRF(3,9)=10.$                                                    DATA    473
FPRF(4,9)=9.$                                                     DATA    474
FPRF(5,9)=2.$                                                     DATA    475
FPRF(6,9)=9.$                                                     DATA    476
FPRF(7,9)=6.$                                                     DATA    477
FPRF(8,9)=8.$                                                     DATA    478
FPRF(9,9)=12*0.$                                                  DATA    479
FQ10=0.08329$                                                     DATA    480
FRA=0.8$                                                          DATA    481
FRB=0.75$                                                         DATA    482
FRT1=0.4$                                                         DATA    483
```

```
FRT2=0.3$                                                DATA     484
FRT3=0.5$                                                DATA     485
FRT4=0.9$                                                DATA     486
FRT5=0.9$                                                DATA     487
FTIME(1.1)=4*4.$                                         DATA     488
FTIME(5.1)=70.$                                          DATA     489
FTIME(6.1)=110.$                                         DATA     490
FTIME(7.1)=80.$                                          DATA     491
FTIME(8.1)=6*5.$                                         DATA     492
FTIME(14.1)=100.$                                        DATA     493
FTIME(15.1)=110.$                                        DATA     494
FTIME(16.1)=300.$                                        DATA     495
FTIME(1.2)=6*50.$                                        DATA     496
FTIME(7.2)=3*80.$                                        DATA     497
FTIME(10.2)=200.$                                        DATA     498
FTIME(11.2)=6*40.$                                       DATA     499
FTIME(13.2)=20.$                                         DATA     500
FTIME(16.2)=20.$                                         DATA     501
FTIME(1.3)=9*10.$                                        DATA     502
FTIME(10.3)=4*6.$                                        DATA     503
FTIME(14.3)=90.$                                         DATA     504
FTIME(15.3)=120.$                                        DATA     505
FTIME(16.3)=60.$                                         DATA     506
FTIME(1.4)=60.$                                          DATA     507
FTIME(2.4)=4*12.$                                        DATA     508
FTIME(6.4)=30.$                                          DATA     509
FTIME(7.4)=5*6.$                                         DATA     510
FTIME(12.4)=4*12.$                                       DATA     511
FTIME(16.4)=30.$                                         DATA     512
FTIME(1.5)=4*7.$                                         DATA     513
FTIME(5.5)=60.$                                          DATA     514
FTIME(6.5)=5*180.$                                       DATA     515
FTIME(11.5)=70.$                                         DATA     516
FTIME(12.5)=4*5.$                                        DATA     517
FTIME(16.5)=30.$                                         DATA     518
FTIME(1.6)=4*4.$                                         DATA     519
FTIME(5.6)=40.$                                          DATA     520
FTIME(6.6)=3*120.$                                       DATA     521
FTIME(9.6)=70.$                                          DATA     522
FTIME(10.6)=7*5.$                                        DATA     523
FTIME(1.7)=8*15.$                                        DATA     524
FTIME(9.7)=8*14.$                                        DATA     525
FTIM1(1.1)=5*225.$                                       DATA     526
FTIM1(6.1)=290.$                                         DATA     527
FTIM1(7.1)=50.$                                          DATA     528
FTIM1(8.1)=9*60.$                                        DATA     529
FTIM1(15.1)=270.$                                        DATA     530
FTIM1(1.2)=10*135.$                                      DATA     531
FTIM1(10.2)=250.$                                        DATA     532
FTIM1(11.2)=6*60.$                                       DATA     533
FTIM1(13.2)=90.$                                         DATA     534
FTIM1(16.2)=90.$                                         DATA     535
FTIM1(1.3)=14*105.$                                      DATA     536
FTIM1(15.3)=305.$                                        DATA     537
FTIM1(16.3)=50.$                                         DATA     538
FTIM1(1.4)=16*100.$                                      DATA     539
FTIM1(1.5)=5*180.$                                       DATA     540
FTIM1(6.5)=5*300.$                                       DATA     541
FTIM1(7.5)=50.$                                          DATA     542
FTIM1(9.5)=50.$                                          DATA     543
FTIM1(11.5)=50.$                                         DATA     544
FTIM1(12.5)=5*100.$                                      DATA     545
FTIM1(1.6)=5*170.$                                       DATA     546
FTIM1(6.6)=3*300.$                                       DATA     547
FTIM1(7.6)=50.$                                          DATA     548
FTIM1(9.6)=50.$                                          DATA     549
FTIM1(10.6)=7*110.$                                      DATA     550
FTIM1(1.7)=230.$                                         DATA     551
FTIM1(2.7)=15*70.$                                       DATA     552
FTIM2(1.1)=270.$                                         DATA     553
FTIM2(2.1)=288.$                                         DATA     554
```

```
FTIM2(3,1)=289.$
FTIM2(4,1)=290.$                                              DATA     555
FTIM2(5,1)=2*365.$                                            DATA     556
FTIM2(7,1)=8*275.$                                            DATA     557
FTIM2(14,1)=310.$                                             DATA     558
FTIM2(15,1)=365.$                                             DATA     559
FTIM2(16,1)=200.$                                             DATA     560
FTIM2(1,2)=6*260.$                                            DATA     561
FTIM2(1,2)=230.$                                              DATA     562
FTIM2(7,2)=362.$                                              DATA     563
FTIM2(8,2)=363.$                                              DATA     564
FTIM2(9,2)=364.$                                              DATA     565
FTIM2(10,2)=365.$                                             DATA     566
FTIM2(11,2)=6*160.$                                           DATA     567
FTIM2(13,2)=200.$                                             DATA     568
FTIM2(16,2)=200.$                                             DATA     569
FTIM2(1,3)=14*250.$                                           DATA     570
FTIM2(1,3)=215.$                                              DATA     571
FTIM2(15,3)=365.$                                             DATA     572
FTIM2(16,3)=130.$                                             DATA     573
FTIM2(1,4)=315.$                                              DATA     574
FTIM2(2,4)=15*240.$                                           DATA     575
FTIM2(1,5)=5*290.$                                            DATA     576
FTIM2(1,5)=240.$                                              DATA     577
FTIM2(6,5)=5*365.$                                            DATA     578
FTIM2(7,5)=120.$                                              DATA     579
FTIM2(9,5)=120.$                                              DATA     580
FTIM2(11,5)=120.$                                             DATA     581
FTIM2(12,5)=5*210.$                                           DATA     582
FTIM2(1,6)=5*280.$                                            DATA     583
FTIM2(6,6)=3*365.$                                            DATA     584
FTIM2(7,6)=150.$                                              DATA     585
FTIM2(9,6)=150.$                                              DATA     586
FTIM2(10,6)=7*240.$                                           DATA     587
FTIM2(1,7)=330.$                                              DATA     588
FTIM2(2,7)=180.$                                              DATA     589
FTIM2(3,7)=14*260.$                                           DATA     590
FTIM2(9,7)=180.$                                              DATA     591
FTIM1(1,1)=112*370.$                                          DATA     592
FT11=-30.$                                                    DATA     593
FT12=-15.$                                                    DATA     594
FT21=-10.$                                                    DATA     595
FT22=10.$                                                     DATA     596
FT31=40.$                                                     DATA     597
FT32=35.$                                                     DATA     598
FT41=46.$                                                     DATA     599
FT51=45.$                                                     DATA     600
FT52=30.$                                                     DATA     601
FT53=0.62$                                                    DATA     602
FT54=0.7$                                                     DATA     603
FT61=65.$                                                     DATA     604
FT62=50.$                                                     DATA     605
FT63=0.8$                                                     DATA     606
FT64=0.9$                                                     DATA     607
F12=16.$                                                      DATA     608
                                                              DATA     609
```