

|| RANGES Grassland Simulation Model ||

Range Science Series No. 17

Bradley J. Gilbert

PROVIDED BY COLORADO STATE UNIVERSITY EXPERIMENT STATION

ABSTRACT

The purpose of the RANGES simulator is to link a rangeland ecosystem with the range management and range economic systems. With this goal in mind, the level of complexity of the model needs only to satisfy the input requirements of the management and economic models.

The model which meets these goals is composed of driving or exogenous variables, a soil water submodel, a producer section within a feedback loop containing a consumer section, and a market or economic section. The driving variables are mean daily temperature and daily precipitation either read from tapes of actual climatic data or generated hypothetically. The soil water submodel is primarily an evapotranspiration function based on soil characteristics such as wilting point and field capacity. The information from the soil water level coupled with the mean daily temperature is used to control plant growth. The forage consumed by livestock consists of green and dead plant material. The protein content of each forage component influences the computed livestock consumption rate, which in turn determines whether or not the animals are gaining weight. The market section of the model calculates an animal price vector of dependent normal random variables from an array of mean net prices and a variance-covariance matrix for net prices.

The plant growth response generated by the model is designed for information to management models such as forage standing crop and the variation of forage available to cattle. For example, the simulation model can be used interactively with a dynamic optimization model, supplying forage response to different levels of grazing intensity and climatic fluctuations. The simulation also supplies cattle weight gains for management purposes, and allows testing of management grazing regimes on simulated forage.

i

ACKNOWLEDGMENTS

Much of the original coding of this model was done by George S. Innis, Donald A. Jameson, and Richard Miskimins using modeling techniques programmed by Jon Gustafson. Data for various process expressions were provided by John Nunn, Richard Rice, M. J. Trlica, and Warren Whitman.

TABLE OF CONTENTS

P	age
Introduction	1
Concepts	2
Driving Variables	3
Soil Water Submodel.	5
Plant Growth Submodel.	8
Photosynthesis	8
	10
	10
	13
	13
Economic Simulation	20
Users Guide to the RANGES Grassland Simulation Model	23
Deck Structure	23
Data Section	26
	26
Output Control Directives	
PRINT Directives	27
FLOW Directives	27
PLOT Directives	27
User Supplied Data	28
Modifications	31
Additional output	31
Additional state variables and flows	31
Substituting Subroutines	31
	51
Case Examples	32
Example 1: Pawnee Grasslands	32
	35
Example 2: Pawnee Grasslands	
Example 3: Eastern Colorado Range Station	40
Example 4: Dickinson Site	40
Conclusions	43
	79
Literature Cited	47
Appendix #1: Variable Names, Definitions and Units	49
Appendix #2: FORTRAN Listing of the RANGES Grassland Simulation	
Model	61
Appendix #3: Example Subroutine Substitution	89
Appendix #4: RANGES Program Card	93
	0-
Appendix #5: Stochastic Precipitation Generator	95
Appendix #6: Coefficient Determination for Sine Function	
Weather Generation	115

INTRODUCTION

The western livestock grazing industry is very dependent upon certain biotic and abiotic processes occurring in the environment. Variations in the temperature and precipitation regime of an area cause unpredictable fluctuations in the resource response. However, given the source of the variation and the processes influencing resources, some questions concerning efficiency of grazing can be answered.

The study of ecological response to abiotic variables (variables that perturb the biotic system and cause variation) can be undertaken in several ways. The first, and most frequently utilized, is the measurement of ecological parameters in a field or laboratory study. This method attempts to reduce variation in an experiment by controlling certain physical parameters which affect the response of the dependent variable. Controlled experiments, however, usually entail removing the measured variable from its natural state.

A contemporary method, which relies on knowledge of the natural behavior of the system, involves adapting empirical information into a mathematical model. The result is a system which can be experimented with, while maintaining the interaction of the components. Field and laboratory experiments become costly and unmanageable quickly when dealing with a complex system. On the other hand, once a mathematical model has been developed, a multitude of experiments can be performed inexpensively. Although mathematical models are always less complete than the natural systems, they are capable of supplying information when only low resolution information is required to solve a problem.

Ecological data of the type generated by the model is of primary importance to grassland management. Given stochastic driving variables, plant production can be generated and the statistical properties of the resulting data can be used in management models.

A series of grassland ecological models were developed under the title RANGES. The objectives were to use available information regarding the growth of herbage in a grassland ecosystem as a function of the major driving variables which are taken to be temperature and rainfall. Forage response, together with the known response of various classes of domestic ruminants can then be used to determine animal weight gains for different stocking strategies and for different supplemental feeding strategies. The model can be used to investigate various marketing strategies and the effects of this entire collection of management practices on income.

These objectives delineate certain biotic, abiotic, and economic systems as being necessary in a model which addresses the objectives. A plant growth system is needed because of its influence on the livestock enterprise. With these data and acknowledging the natural variation inherent in herbage growth, the second system requirement becomes evident. Plant growth is determined by solar radiation, soil temperature, CO₂ availability, soil water and phenological stage; but air temperature and rainfall are assumed to be the most important driving variables.

The soil water system is the link between the abiotic and biotic systems. Although there is a strong correlation between rainfall and plant growth, there are important processes which limit the quantity of water available for plant growth and determine the persistence of available water in the soil layer.

The final model requirement, "investigating market strategies", necessitates information about market behavior, including random fluctuations which might be expected. The economic system also requires knowledge of costs involved in the enterprise to determine when marketing might be most profitable. Generality and simplicity are also necessary attributes of the model if it is to be used on varied grassland sites with minimal data input from a specific site.

CONCEPTS

Simulation techniques. There are two major schools of thought in mathematical modeling of dynamic systems. One revolves around systems of differential equations and their exact solutions, and the other is oreinted toward simulating the solution with systems of differential or difference equations. There are several practical and philosophical reasons for choosing simulation as the method for meeting the objectives of the RANGES model. The first and foremost

reason is the inability of analytic solution techniques to solve the system of equations which addresses the objectives of the model (Forrester, 1968). The conceptual difficulty in using instantaneous rates when they are physically unmeasurable (as $\lim_{\Delta t \to 0}$) also lends credibility to the use of simulation techniques (Innis, 1972). Systems are often viewed in difference terms; i.e. the flow from one state to another is measured over a finite time interval, which makes a scheme using difference equations more credible (Innis, 1972).

A system of difference equations is the basis of this model using an initial value solution technique. Thus, each successive state of the system is determined from the previous one, knowing only the flow definitions and the level of the state variable at the previous time step. The general form of the solution scheme is:

$$X(t + \Delta t) = X(t) + \Delta t * (\Sigma F_{J} - \Sigma F_{L})$$
(1)

where X(t) is the amount of material in, or the level of, the state variable X at time t, t is the current simulated time, Δt is the time step or increment of simulated time, ΣF_i is the sum of the flows into state variable X and ΣF_o is the sum of the flows out of state variable X.

Equation 1 states that the level of X after Δ t amount of time has elapsed will be the level of X at the previous time plus the sum of the flows into and out of X multiplied by the time step. Thus, interaction in the system can be easily represented by having the flows into or out of one level be dependent on another level. This conceptual framework was retained for all versions of the RANGES models whether they were coded in SIMCOMP or FORTRAN.

The model structure is composed of five principal parts: (i) abiotic driving variables, (ii) soil water submodel, (iii) producer section, (iv) consumer section, and (v) economic section (Fig. 1).

Driving Variables

Driving variables are those that are independent of the simulated system, but are necessary to initiate response by the system within some arbitrary boundary which is being modeled. In RANGES, the driving variables are daily precipitation and mean daily temperature. These variables are not mechanistically described in the model because they are not considered part of the feedback with the biotic subsystem.

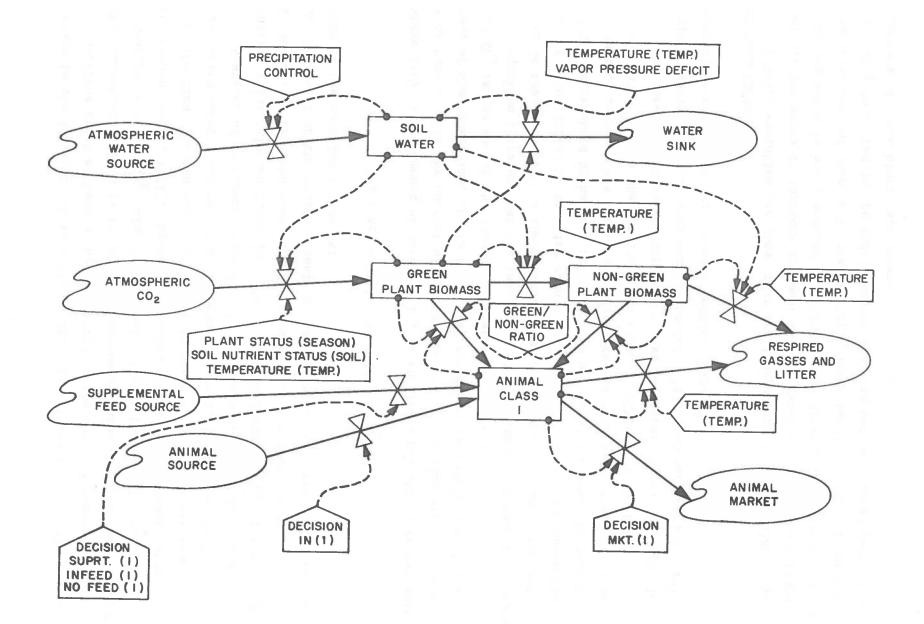


Fig. 1. Diagram for the RANGES grassland simulation model.

Daily precipitation can be obtained in three ways for use in RANGES: (1) from historical weather records read in from magnetic tape, (2) from a sine function representation of average historical data, or (3) from a stochastic precipitation generator. For debugging purposes, average historical precipitation data can be represented by a sine function with appropriate phase and amplitude shifts, such as the following equation for the Pawnee Grasslands (Fig. 2):

Daily Rainfall (inches) = $(14.*(SIN((IDAY-60)/365.*2\pi) + 1.))/365.$ (2)

where IDAY is the Julian day.

The average daily temperature has two possible forms for entry into the model: (1) historical data from magnetic tape, or (2) as a sine function representation of historical averages, such as the following for the Pawnee Grasslands (Fig. 3):

Average daily temperature (°F) = 30. * $(SIN((IDAY-120)/365.*2\pi) + 1.0) + 22.0$

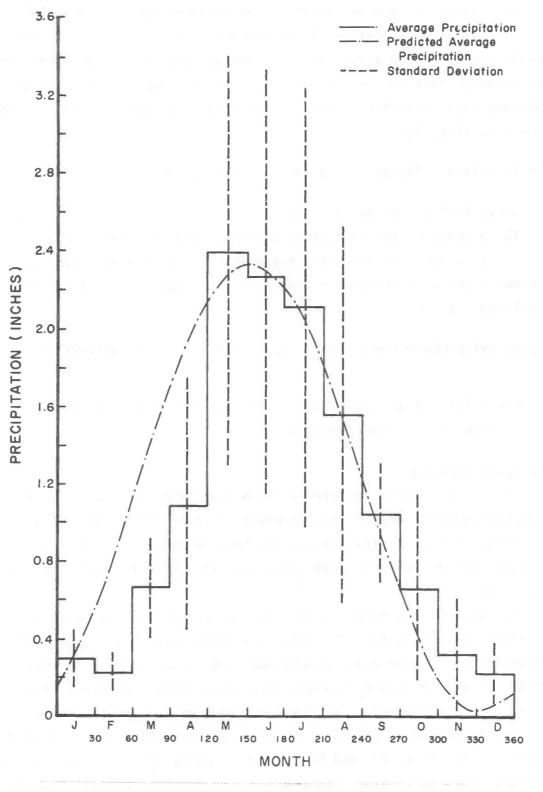
(3)

where IDAY is the Julian Day. (See Appendix 6 for derivation of coefficients for the sine function.)

Soil Water Submodel

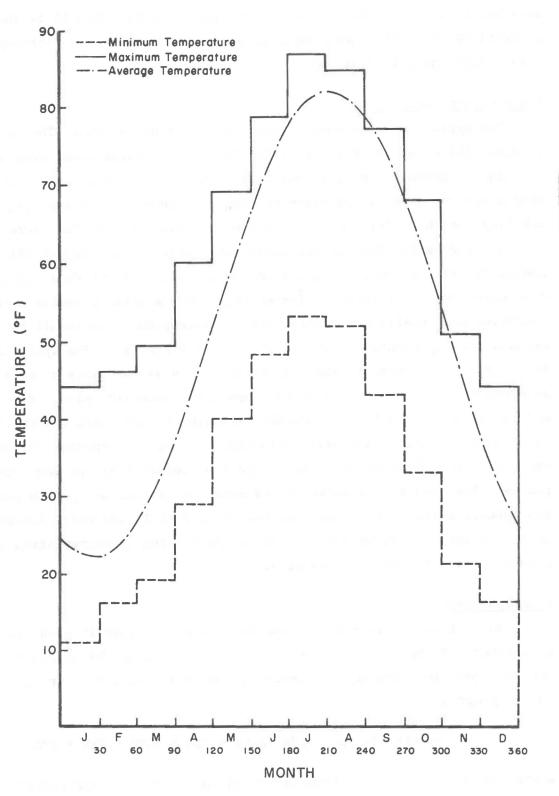
The driving variables perturb the system from its initial state via several processes. One of the processes, evapotranspiration (ET), is influenced by both precipitation and air temperature. Although other factors are important in the ET process, these two give results which meet the model objectives.

The soil water submodel takes temperature and precipitation as inputs and, after accounting for ET, yields available soil water. The soil water present influences the plant growth and is used as an index to moisture effects on such processes as plant death and decay. Potential ET is calculated as a function of temperature and is used to compute actual ET as a function of soil water present. However, above field capacity potential ET is used as the actual ET, and below wilting point ET is set equal to onehundredth of ET at wilting point, i.e. there is no more transpiration. When



PPT = 14.* (SIN((IDAY - 60)/365.* 2π) + 1)/12.

Fig. 2. Predicted monthly precipitation amounts using a sine function for the Pawnee National Grasslands. Averages and standard deviations are from Rasmussen et al., 1971.



Temperature = 30. * (SIN ((I DAY -120)/365. $*2\pi$) + 1.0) + 22.0

Fig. 3. Predicted daily temperature values using a sine function for the Pawnee National Grasslands. Maximum and minimum temperatures are from Rasmussen et al., 1971.

soil water exceeds total soil water holding capacity, then ET is increased proportionally to the excess soil water, which is a ploy to represent water runoff from the soil (Fig. 4).

Plant Growth Submodel

Two approaches were used in dealing with plant growth. The first used a regression equation of a particular form to represent plant growth as a function of temperature, soil water and live plant biomass. The second method used a more mechanistic approach in hopes of achieving a wider range of applicability for the model. Only the second approach is described here.

The regression formulation caused the model to be site specific. Its use at another site would require collecting plant growth data and determining new regression coefficients. Therefore, a more generally usable formulation requiring only easily obtainable data was attempted. The result is a more mechanistic representation of the plant growth process. The approach is to take basic photosynthesis and respiration rates as functions of soil water and temperature, and from these to compute the potential plant production per unit of live plant material. Plants were also divided into two categories that exhibit distinct photosynthesis and respiration responses to temperature and moisture. These two categories are cool season (C3) and warm season (C4) plants. The result is a better representation of seasonal growth patterns and species differences between sites. This formulation only requires information on abiotic parameters and soil water holding characteristics as input when the model is run on a new site.

Photosynthesis

The photosynthesis rate is assumed to depend primarily upon available soil water and average daily temperature. The form of the equation is the same for both cool and warm season plants with Q being equal to C or W, respectively in the program.

Photosynthesis rate
$$(g/m^2/day) = ETQ * ESMQ * QBM * QMAX$$
 (4)

where ETQ is the effect of temperature on photosynthesis (proportion); ESMQ is the effect of soil water on photosynthesis (proportion); QBM is aboveground green plant biomass (g/m^2) ; and QMAX is the maximum photosynthetic rate (grams photosynthate/gram live plant biomass/day).

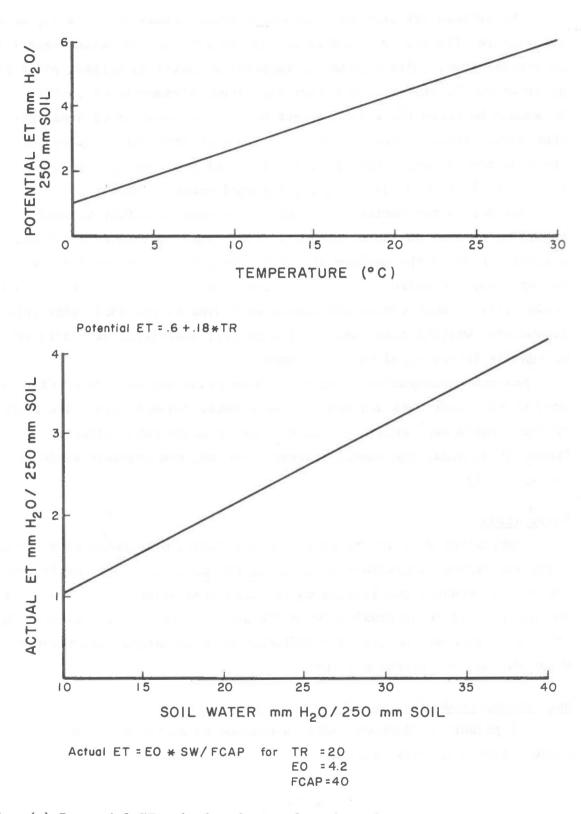


Fig. 4. (a) Potential ET calculated as a function of temperature.

(b) Actual ET calculated from potential ET and the amount of water present in the soil.

The temperature term is intended to embody radiation effects, as well as temperature effects. A parabola is used to give an increasing beneficial effect on photosynthesis with increasing temperature until an optimum point is reached, after which the temperature effect diminishes (Connor et al., 1974) (Fig. 5). It should be noted there is a slight shift in the curves to seemingly lower than normal temperatures, a result of using average daily temperature. Also, there is new evidence that western wheatgrass requires only very low temperatures for growth initiation (Joe Trlica, personal communication).

The soil water factor represents the impact of effective soil water in the top 250mm of soil on photosynthesis. Additional soil water increases the photosynthesis rate to the maximum at field capacity. Different functions were used to represent the differences between cool and warm season plants reported by Brown (1974). Both curves are scaled according to the soil-water relation parameters, wilting point and field capacity; therefore, site differences can be readily incorporated into the model.

Maximum photosynthesis rates for blue grama and western wheatgrass were used to represent warm and cool season plants, respectively. The highest reported rates were obtained from CO₂ gas exchange data collected at the Pawnee Grasslands, and should represent the maximum possible field rate (Brown, 1974).

Respiration

Respiration data was collected on the Pawnee Grasslands site for blue grama and western wheatgrass using a CO₂ gas exchange mechanism (Brown, 1974). The results enabled respiration to be calculated using an exponential function on the temperature interval between 0°C and 30°C (Fig. 6). Temperature was considered to have the dominant influence on respiration and, therefore, soil water was not considered a factor.

Net Primary Production and Aboveground Accumulation

Net primary production (NPP) is assumed to be the difference between photosynthesis and respiration:

$$NPP = P_{s} - R_{s}$$
(5)

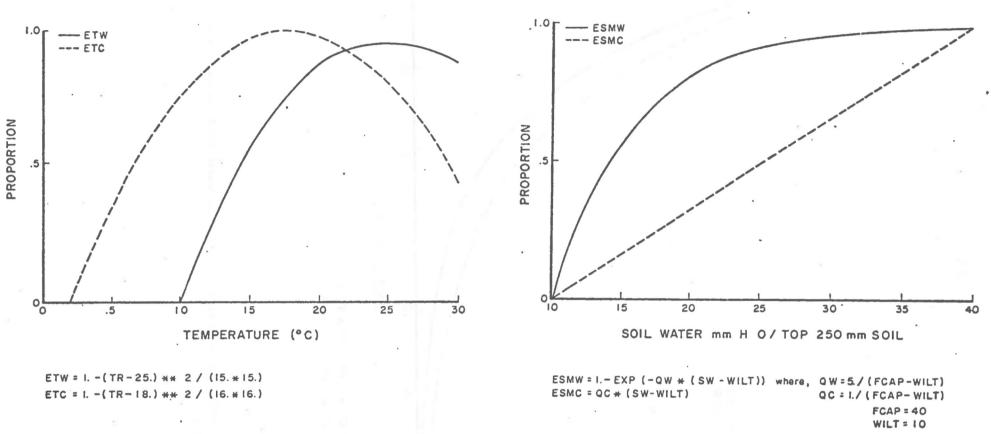
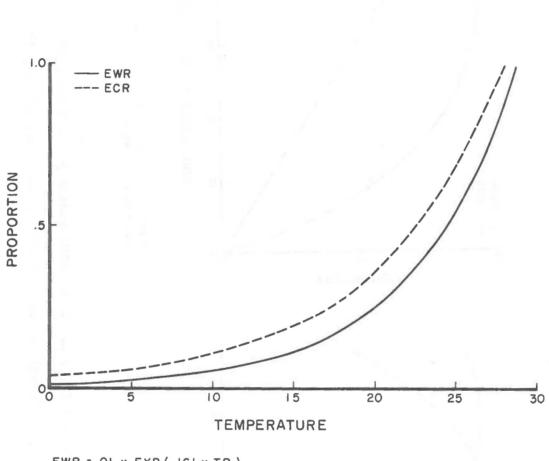


Fig. 5. (a) Effect of temperature on cool (ETC) and warm (ETW) season plant photosynthesis.

(b) Effect of soil water on cool (ESMC) and warm (ESMW) season plant photosynthesis.



EWR = .01 * EXP(.161 * TR) ECR = .03 * EXP(.126 * TR)

Fig. 6. Effect of temperature on cool (ECR) and warm (EWR) season plant respiration.

All of the carbon fixed by the plant shoots is not retained aboveground, however. Large quantities of the photoassimilate are translocated belowground to crowns and roots. There is evidence to suggest that as much as 60 to 88% of the photoassimilate is translocated (Singh and Coleman, 1974). Although seasonal variation undoubtedly exists as a function of plant phenology, the lumping of plant species in this model removes the importance of the variation. Therefore, a value of 70% was used to represent translocation.

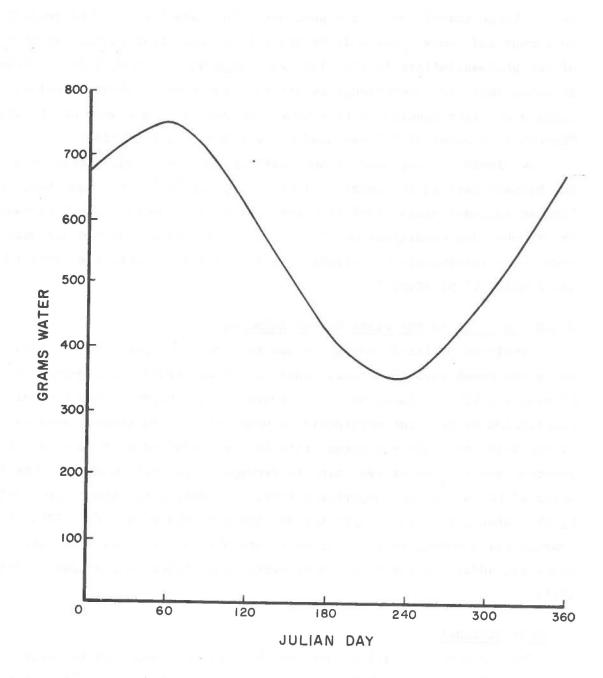
Two feedbacks are used to help give the photosynthesis equation stability. The biomass term in the equation itself helps stabilize the equation, and limited photosynthesis based on water availability helps prevent unreasonable growth when dry conditions occur. The grams of water required are varied over seasons to incorporate the effects of plant phenology into the model (McGinnies and Arnold, 1939) (Fig. 7).

Death and Decay in the Plant Growth Submodel

Providing realistic forage values requires incorporation of death and decay processes into the model. These processes result in non-green plant biomass and litter compartments. Non-green plant material can accumulate when respiration exceeds photosynthesis or when hot or cold temperatures kill plant tissue (Fig. 8). The non-green state is then modified as a function of temperature and soil water resulting in decomposition, represented by the flow of material to the litter compartment (Fig. 9). Both compartments are influenced by the intensity of cattle grazing simulated in the model (Fig. 10). Heavy grazing kills some live plant material and also knocks down non-green plant material, adding to the flows to non-green and litter, respectively (Whitman, 1974).

Consumer Submodel

The consumer submodel is part of the feedback loop with the producer submodel. Consumers exhibit control on plant growth as a function of the number of animals present. Control is attributed not only to grazing by consumers, but also to trampling loss. The model is designed to handle up to five livestock classes, e.g. steers, calves, cows, sheep, etc. All that is required for a consumer to be entered into the model is that the consumer



B = 550. + 200. * SIN((IDAY + 40)/365. * 6.28)

Fig. 7. Grams of water required to produce one gram dry weight of green plant biomass.

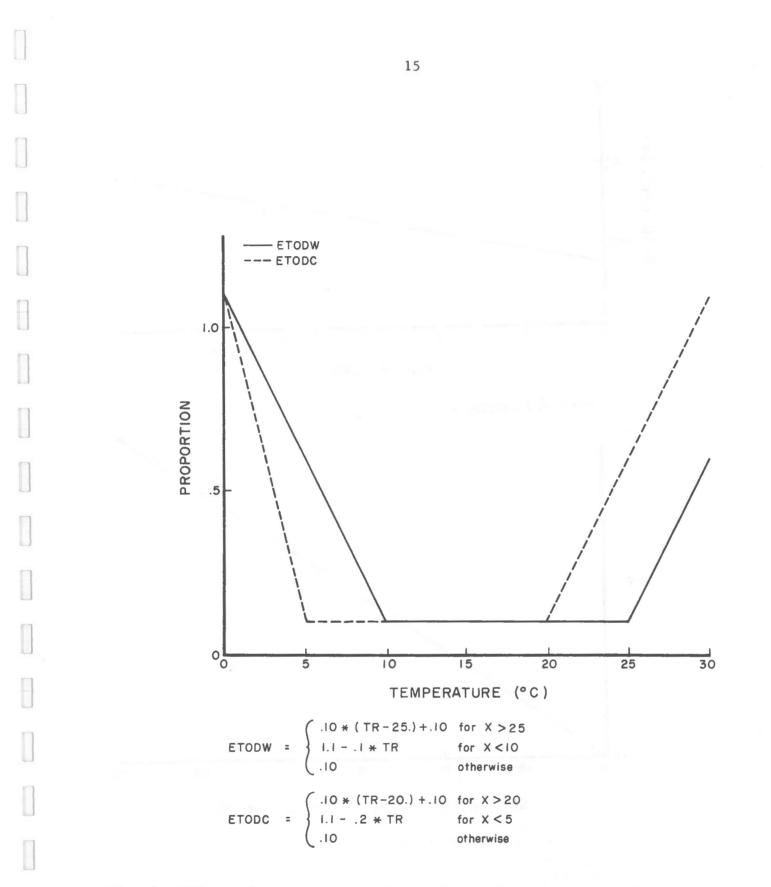
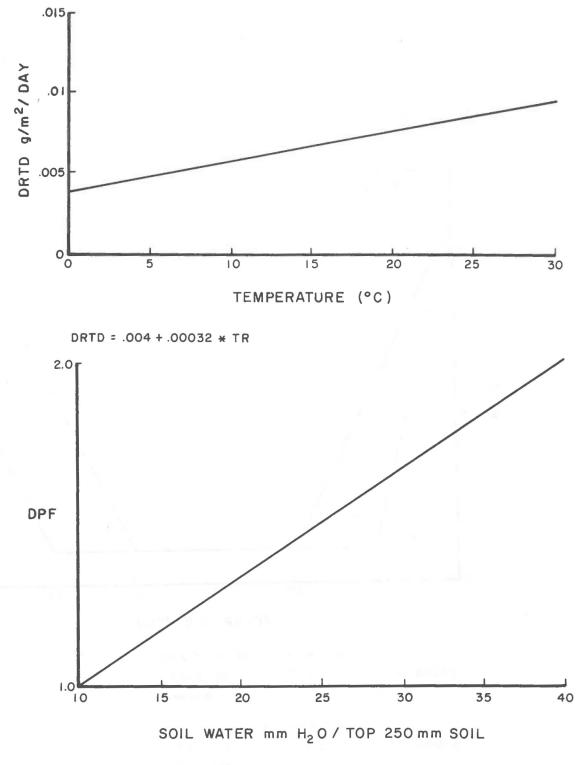


Fig. 8. Effect of temperature on death of cool (ETODC) and warm (ETODW) season plants.

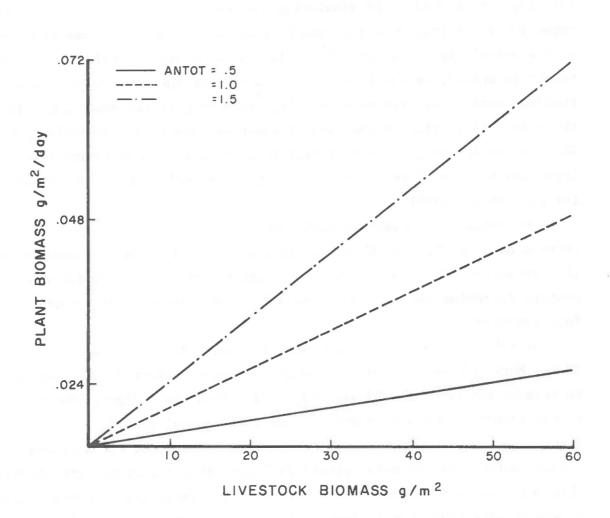


16

DPF = I. + (I. / (FCAP - WILT)) * (X(2I) - WILT) FCAP = 40 WILT = 10

Fig. 9. (a) Dry plant material flow rate from standing dead to litter as a function of temperature.

(b) Increased flow from standing dead to litter due to the presence of water.



Plant loss $(g/m^2/day) = .0008 * X(12) * ANTOT$

Fig. 10. Trampling loss from livestock for 3 grazing intensities.

be a ruminant herbivore and that certain pertinent information be known, i.e. number of individuals in the consumer class, average weight per individual, effect of supplemental nitrogen on forage consumption, etc.

The consumer's diet is composed of live and dead plant material. Ingestion rate is determined by simulating rumen activity (Rice et al., 1974). Rumen capacity is determined as a constant proportion (.16) of the metabolic size of the animal (kg liveweight ^{.75}). In all cases where a relationship to body weight is needed, metabolic size is used to make the model more general for ruminant consumers. The potential ingestion rate is the rumen capacity minus the rumen fill, which is composed of rumen dry matter and microbial protein. When available forage is not sufficient to meet livestock demand potential, ingestion is reduced as a function of the consumer's ability to utilize limited forage (Bement, 1969).

The dynamics of rumen fermentation are central to the herbivore intake subroutine. The biological assumptions are that the rumen microbes control the processing rate of foodstuffs and that the available carbohydrate and protein determine the quantity of microbial protein which is acting on the food ingested.

Microbial protein yield per 100g of digestible organic matter is about 8-20g (Hume, Morr and Somers, 1970). Dietary nitrogen content is used as an index to organic matter digestibility (Fig. 11). Thus, 100g digestible organic matter is equivalent to 430 kcal energy. Then if microbial protein yield is 8g/g digestible organic matter, .0186g of protein is produced per kcal digestible organic matter). Therefore, as nitrogen in the diet increases, digestibility increases and the growth of microbial protein increases. However, when dietary nitrogen content exceeds 1.8% microbial protein, growth is assumed to hold constant rather than to be continually increasing.

Rumen microbial protein content determines how much of the potentially digestible food will be digested. Four percent of ingested foods are digested per day for each gram of microbial protein per metabolic size as determined from digestion versus microbial population data. As microbial protein increases, the fermentation digestion rate increases, which increases the normal exit rate of digestible material, leaving more space in the rumen for increased ingestion. Also, higher fermentation rates add to the normal passage rate of material from the rumen to the intestine, again increasing potential intake.

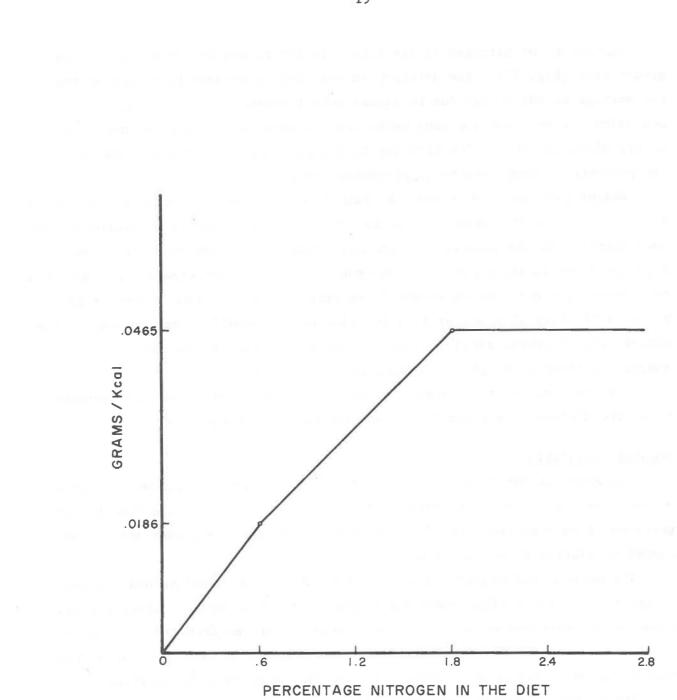


Fig. 11. Grams of microprotein resulting from each kcal of digestible organic matter as a function of percentage nitrogen in the diet.

The amount of nitrogen in the forage is determined by the relative plant growth rate (Fig. 12). The nitrogen in the diet is assumed to be higher than the average in the forage due to animal selectiveness. An approximate nitrogen value was obtained assuming herbivores consumed three times as much green as dry plant material. The nitrogen in the diet may also be increased by the presence of high protein supplemental feed.

Weight gain for each livestock class is determined by subtracting metabolic requirements from the energy resulting from digestion. Energy is obtained from food passed into the intestine, from microprotein, and from volatile fatty acids produced in the rumen. If this energy supply is not greater than metabolic requirements, a net loss in weight is experienced by the animal. Weight gain per kcal of digestible energy is a function of an animal's size compared to its mature size (Hedrick, 1968). Thus, the larger the animal, the greater the energy requirement per gram of weight gain (Fig. 13).

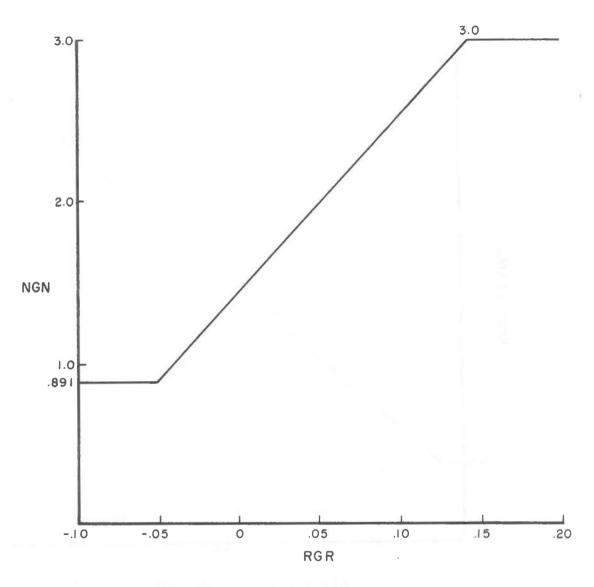
Livestock respiration, excretion and all other animal losses are assumed to be the difference between total food ingested and weight gain.

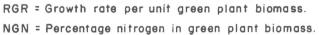
Economic Simulation

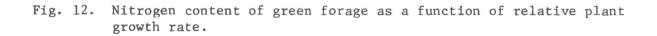
The economic submodel was developed to generate stochastic monthly market values from historical information. The value of livestock could then be determined if we know their initial cost and supplemental feed cost and assume rangeland grazing to be cost free.

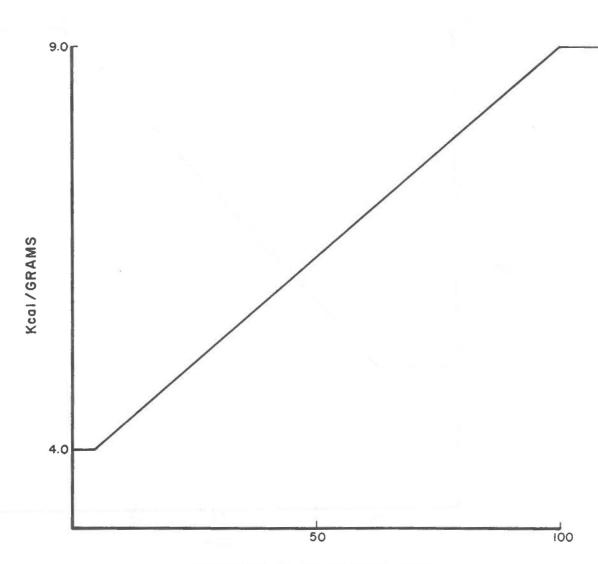
The method used requires a vector of monthly mean market values and the square root of the variance-covariance matrix of the dependent market values. A vector of dependent normal random variables is derived from this information using the methods of Naylor et al. (1966). The method entails generating the random variables from a multivariate normal distribution with covariance.

The economic submodel assumes a cattle enterprise for determining the base market value. Some transformation would be necessary for conversion to sheep or other livestock values. During the simulation, prices are determined on a monthly basis and compared with the cash value of each livestock class enabling the current values of each class to be determined.









PERCENTAGE OF MATURE SIZE

Fig. 13. kcal of available energy required to produce 1 gram of gain for animals at different stages of maturity.

USERS GUIDE TO THE RANGES GRASSLAND SIMULATION MODEL

RANGES is written in ANSI standard FORTRAN, and has been successfully exported to two other installations. A deck and this manual are available at cost from the Regional Systems Program, 325 Aylesworth Hall, Colorado State University, Fort Collins, Colorado, 80523.

The FORTRAN coded model is composed of two major parts. One is the biological simulation and the other is the simulation programming overhead (Fig. 14). In general, any simulation of interest could be substituted for the first section, while the second section would remain unchanged, except for the inputs.

Structural modifications to the model are described in a later section of this report, and give insight into the simulation method used. The model was developed in a modular fashion to allow changes and function substitutions. Subroutines describing biological functions such as plant growth (GROW), evapotranspiration (ET) and ruminant consumption (RUMEN) can be substituted directly for inadequate or obsolete functions, provided the subroutine parameter lists remain the same. A substitution for RUMEN illustrates this point (Appendix 3).

Deck Structure

There are three sections required for running this model: (i) control cards, (ii) FORTRAN program, and (iii) input data. The control cards required to run RANGES will vary from installation to installation. The control cards required at CSU are discussed below to give programmers at other installations an idea of the analogous commands to give other machines.

CARD NUMBER:

- 1 HI527, AFNRCSCT, T40, PR100, CM60000. RANGES.
- 2 FTN, L=0.
- 3 LGO.

4 - 7-8-9 (multipunched in column 1)

5 - (7 spaces) PROGRAM RANGES (INPUT, OUTPUT, TAPE1=INPUT, TAPE2=OUTPUT, TAPE3=PLTSV, TAPE4, TAPE5)

Card 1 identifies the job and sets the physical requirements and limits for the job. It contains the job sequence number (HI527), the charge account number (AFNRCSCT), the time limit (40 central processor seconds), printed page limit (100 pages), core limit (60 K), and an identification code (RANGES).

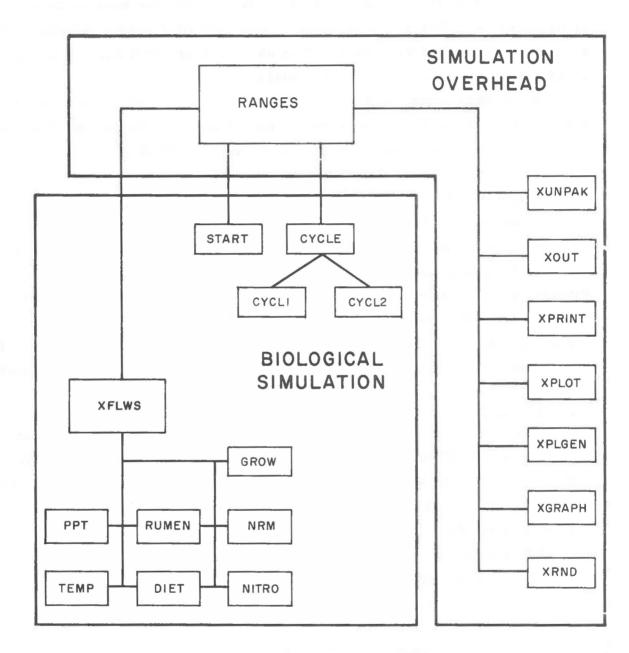


Fig. 14. RANGES program organization indicating modular design.

The model requires about 16 seconds to compile and about 16 seconds to execute for a one-year simulation with a one-day time step on a CDC 6400 computer. The number of printed pages varies according to the output requests and whether a program listing is obtained. The program listing alone requires about 60 pages including loader maps, and average printed simulation output is about 20 pages.

Card 2 (FTN,L=0.) loads the FORTRAN compiler into central memory and compiles the program. The L=0 parameter supresses the program listing to reduce printed output.

Card 3 (LGO.) loads the object code and executes the simulation with the results being printed on the system output device.

Card 4 (7-8-9) separates the control cards from the FORTRAN deck.

Card 5 (PROGRAM....) is the first card of the FORTRAN deck and specifies input and output devices. It is specific to CDC machines and is explained in Appendix 4.

The remainder of the FORTRAN deck (Appendix 2) follows Card 5 and is trailed by another separator card (7-8-9). The data section follows the end of record card (7-8-9) and can be composed of as many as four parts: (i) output control directives requesting variables to be printed and plotted, (ii) user supplied data cards which define the simulation input variables, (iii) data cards generated from the program PREGEN if stochastic precipitation is desired or sine function parameters for precipitation, and (iv) sine function parameters for temperature.

Different control cards are necessary if abiotic data is to be input from a magnetic tape; for example:

Job Card FTN,L=0. COPYCR,INPUT,DATA. REQUEST,EXFILE,HY,ID-D0245,READ. REWIND,EXFILE. COPYCF,EXFILE,DATA,1. REWIND,DATA. LGO,DATA. 7-8-9 The gist of these control cards is to copy the data following the program deck, onto a file called DATA. Secondly, request a magnetic tape with the necessary abiotic data on it and copy that information onto the file called DATA following the information from the previous copy. Finally, execute the program with file DATA attached as the input file. It should be noted that the input format in SUBROUTINE RECRD should correspond to the format of the abiotic tape. Also, specification of stochastic weather generation requires running a separate program (PREGEN) to obtain input parameters for the geographic point of interest (Appendix 5).

Data Section

The data section follows the end of record card (7-8-9), which trails the FORTRAN deck, and consists of two parts. The first part of the data section is a set of output control directives requesting variables to be printed and plotted. The second part consists of user-supplied data cards which define the input variables for the simulation. The last data card must be a blank trailer.

Output Control Directives

All output control directives contain the following data fields: COMMAND. nl,n2,n3,...,n14.

The command begins in column 1 and contains no embedded blanks. Legal commands are PRINT, FLOW, PLOT, and END. The integer constants n1 through n14 are right-justified in fields of five columns each, starting in column 11. Note that some fields may not be used by some commands.

The card columns of each of the fields are:

Field	Card Column		
COMMAND	1-10		
n1	11-15		
n2	16-20		
n3	21-25		
•	٠		
٠	•		
n14	76-80		

The output directives PRINT, FLOW, and PLOT may appear in any order before the END. card.

PRINT Directives

The state variables X(J), J=1,99, may be requested for tabular output by PRINT directives. Upon encountering a PRINT command in columns 1 through 6, the card is scanned in numeric fields of five columns each, starting in column 11 and ignoring blank fields. All constants encountered are interpreted as the indices of the state variables to be printed. Note that the constants must have values which range from one to 99. There is no limit to the number of PRINT cards which may be used. The time interval between printouts is the value of DTPR.

FLOW Directives

The current values for any of the flows between state variables may be requested by the command FLOW.

The command FLOW appears in columns 1 through 5. Successive pairs of numeric fields are then interpreted as the indices of the flows to be printed. For example:

FLOW. 11 12

would print out the flows between X(11) and X(12). As many FLOW cards as necessary may be included. If a flow is requested which does not exist, the flow is ignored. The interval between FLOW printouts is the value of DTFL.

PLOT Directives

A graph of a state variable over time may be requested by a PLOT command. The command PLOT. appears in columns 1 through 5. The first numeric field in columns 11 through 15 is interpreted as the total number of plots or graphs which are needed. The rest of this card is left blank. The cards which follow the PLOT card must be blank in columns 1 through 10. The number of these cards must equal the total number of plots desired, as each card generates a graph. The indices of the state variables which are to be plotted together appear on the same card in the first five numeric fields. There can be at most five variables plotted per graph and at most 20 graphs, i.e. at most, 20 cards after the PLOT card. Example:

In the above example, there will be two graphs. The first will plot state variable 80, and the second graph will plot state variables 21, 11 and 12.

The interval between plotted values is DTPL. This part of the data section must be terminated by an END. card beginning in column 1, whether or not tabular or graphical output is requested.

User Supplied Data

The second section contains initial values of variables for the simulation according to the following order and format (Note: the current state of the model requires DT = 1):

lst card:	TSTRT, TEND, DT, DTPR, DTFL, DTPL
	FORMAT (8F10.0)
2nd card:	X(10), X(11), X(12), X(21)
	FORMAT (8F10.0)
3rd card:	NCLAS, MODT, MODP, INYR, AREA, FCAP, THOLD,
	WILT, SOILA, SOILB, TINIT
	FORMAT (311, 12, 5X, 7F10.0)

Note: SOILA and SOILB are not presently used in the model but are available to the user for future development.

If NCLAS is zero, i.e. there are no herbivores present, then the following cards up to the weather data input are not read and should *not* be present in the data deck. Otherwise, there should be J=1, NCLAS cards following plus the price variance-covariance matrix and mean price vector.

> IN (J), MKT (J), INFD (J), NOFD (J), LVWT (J), SUPRT (J), SUPPR (J), STKNO (J), SUPN (J), MSZ (J) FORMAT (413,8X, 6F10.0)

The next 24 cards contain the animal class price variance-covariance matrix, six elements per card.

PROCOV(12, 12)

FORMAT(7X, 6(E10.4, 1X))

The user would probably want to use the supplied variance-covariance matrix as it represents several years data and then modify PRMN, the mean net price to correspond with the current market status.

Note: The matrix is actually read into a vector PRCOV(144) by columns and later passed to subroutine NRM as a matrix.

The next two cards contain the mean monthly animal price per hundredweight.

PRMN(12)

FORMAT(7X,6(E10.4,1X))

The last two card sets are read if MODP or MODT equals 2 or 3. MODP = 2: three sine function coefficients: H1, H2, H3 FORMAT(3F10.0)

MODP = 3: seven cards, the title and three sets of Fourier coefficients:

(NAME(I), I=1,3)
FORMAT(2A4,A2)
3 pairs:
NT,A1
FORMAT(I3,F10.0)
(A(I), B(I), I=1,NT)
FORMAT(12F6.4)

The first two cards have the number of terms and coefficients for the Fourier representation of the weekly averages of the probability of a dry day. The second two cards represent the probability of a dry day preceded by a dry day and the third pair represents the average weekly storm size.

Last card: MODT = 2 or 3: three sine function coefficients: H4, H5, H6 FORMAT(3F10.0)

The variables used above are defined in Appendix 1. An example set of user supplied data is given in Table 1. These inputs are representative of the Pawnee National Grasslands. .

5	Parameter	Input Value	Dimension	Description
System Control Parameters	TSTRT	0,	Day	Starting simulation time
	TEND	365.	Day	Ending simulation time
	DT	1.	Day	Simulation time step size
Abiotic Parameters	X(21)	20.	mm H ₂ 0/top 250 mm soil	Initial soil water level
	WILT	10.	mm H ₂ 0/top 250 mm soil	Soil wilting point
	FCAP	40.	mm H ₂ 0/top 250 mm soil	Soil field capacity
	THOLD	50.	mm H ₂ 0/top 250 mm soil	Total water holding capacit of soil
	MODT	2.	Control variable	Temperature generated by sine function
	MODP	2.	Control variable	Precipitation generated by a sine function
	AREA	320.	Acres	Area to be simulated
	н1	14.	Dimensionless .	Amplitude of precipitation sine wave divided by two
	H2	60.	Dimensionless	Displacement along time axi
	нз	0.	Dimensionless	Displacement along vertical axis
	н4	30.	Dimensionless	Amplitude of temperature sine wave divided by two
	H5	120.	Dimensionless	Displacement along time axi
	н6	22.	Dimensionless	Displacement along vertical axis
Biotic Parameters	X(10)	0.	g/m ²	Initial warm season plant biomass
	X(11)	0.	'g/m ²	Initial cool season plant biomass
	X(12)	20.	g/m ²	Initial standing dead plant biomass
	IN	0.	Day	Stocking day
	LVWT	400.	15	Initial average body weight for animal class
	MKT	365.	Day	Market day
	SUPRT	.001	Body weight proportion	Supplemental feed rate
	INFD	280.	Day	Begin supplemental feeding
	NOFD	365.	Day	End supplemental feeding
	SUPPR	.025	\$/1b	Supplemental feed cost
	STKNO	30.	Number	Number of animals
	SUPN .	25.	z	Supplemental feed nitrogen content
	MSZ	1200.	15	Mature body weight for animal class
	NCLAS	1.	Number	Number of livestock classes
Sconomic Input Parameters	PRMN	Matrix	\$/ewt	Mean net prices
	PRCOV	Matrix	\$/cwt	Price variance-covariance matrix

Modifications

Additional output. If additional output is needed, FORTRAN WRITE statements using logical unit number 2 or FORTRAN PRINT statements may be inserted in the program wherever they are desired. They must be accompanied by the appropriate FORMAT statements.

Additional state variables and flows. In order to add additional state variables, the variable XNST and the array XST(99) must be altered in the main program. XNST is the number of state variables used and XST(99) is a list of state variable indices.

For example, if the state variable 7 is to be added to the program, the value of XNST must be increased by one, and the number 7 entered at the end of the array XST(99).

In order to add additional flows to the program, the variable XNF and the array XFR must be altered in the main program. XNF is the number of flows and XFR is the flow reference table.

For example, if the flow from state variable 7 to state variable 39 is to be added, the value of XNF is increased by one and 739 is entered as the last entry in the array XFR.

In addition to the above changes, the FORTRAN code of the flow must be added in SUBROUTINE XFLWS as follows:

$$XN = XN + 1$$

$$F = 0.$$

$$.$$

$$.$$

$$FORTRAN CODE$$

$$.$$

$$.$$

$$XF(XN) = F$$

where F is the flow between 7 and 39.

Substituting Subroutines

RANGES was specifically designed to keep processes, as much as possible, within one routine. Thus, new subroutines can take the place of the original ones provided the subroutine argument list remains the same. Care must be taken that variables used in other parts of the program are not left undefined in a new subroutine. If additional variables are needed in a subroutine, some minor programming changes can be made to meet this end. This capability allows detailed site specific information to be readily entered into the model in functional form. An example of such a substitution is shown in Appendix 3.

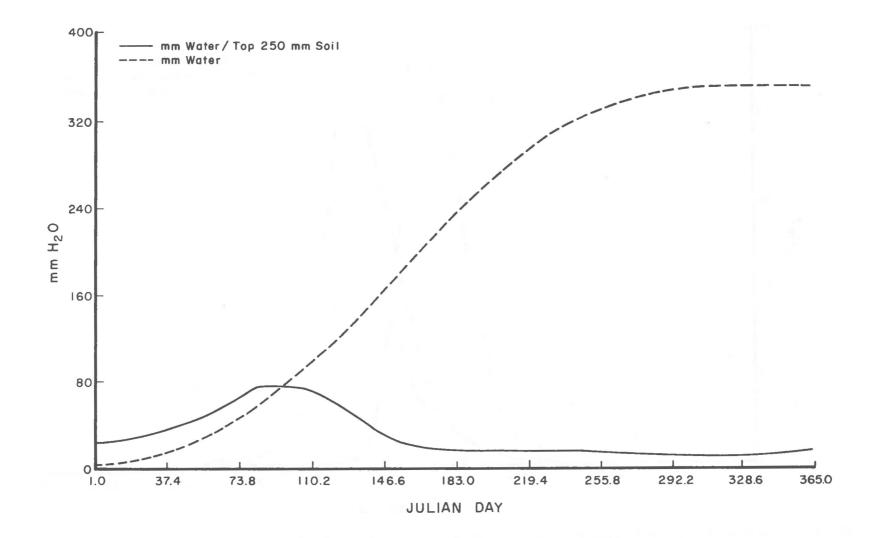
CASE EXAMPLES

Applicability of the model to a wide range of conditions is presented as a series of case examples run on three different experimental range sites. First, the whole model is exercised using Pawnee Grasslands input parameters. The Pawnee Site is simulated again, only without grazing, and the results are then compared to simulations of the Dickinson Site in North Dakota and the Eastern Colorado Range Station (ECRS) in Colorado. The only alterations required when running the model on any of these sites are new soil water parameters and appropriate driving variables for the site, i.e. temperature and precipitation data.

Example 1: Pawnee Grasslands

This example illustrates all of the model components, i.e. soil water, plant production, livestock grazing and economic considerations. The discussion of the model function should help when interpreting the other case examples. Input parameters used to generate these results are given in Table 1.

A simple sinusoidal function was used to generate daily precipitation and temperature values for a one-year run. Soil water rises sharply during the spring when there is precipitation and little evaporation to remove water from the soil. As soon as the temperature increases, soil water is lost both through evaporation and transpiration (Fig. 15). Above 5°C cool season plants initiate growth and above 10°C warm season plants begin growing (Fig. 16). A five-day growth initiation period is allowed for the plants, after which 70% of the photosynthate per day is assumed to be translocated to the roots. Plant production responds to soil water level and average daily air temperature. Growth continues, contingent on soil water availability, until 45 days have elapsed or a decrease in production is experienced. Either of these conditions causes senecence to begin and the accumulation of standing





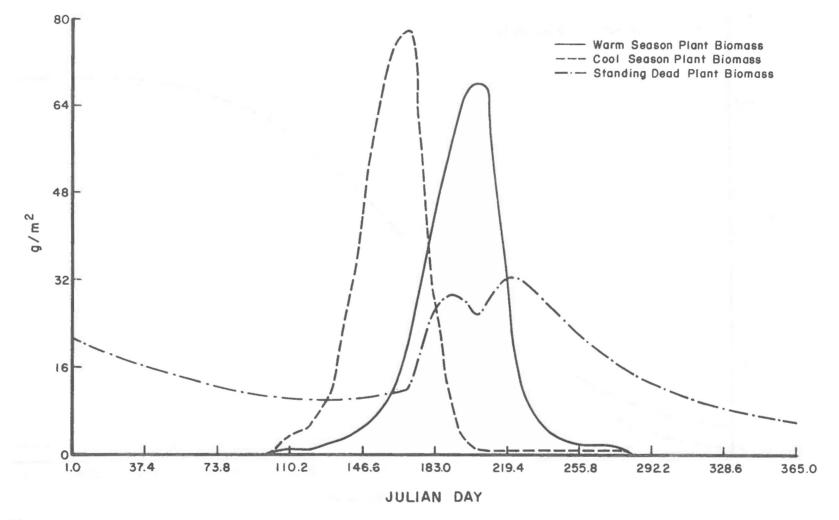


Fig. 16. Simulated cool and warm season plant production and standing dead plant biomass for the Pawnee National Grasslands.

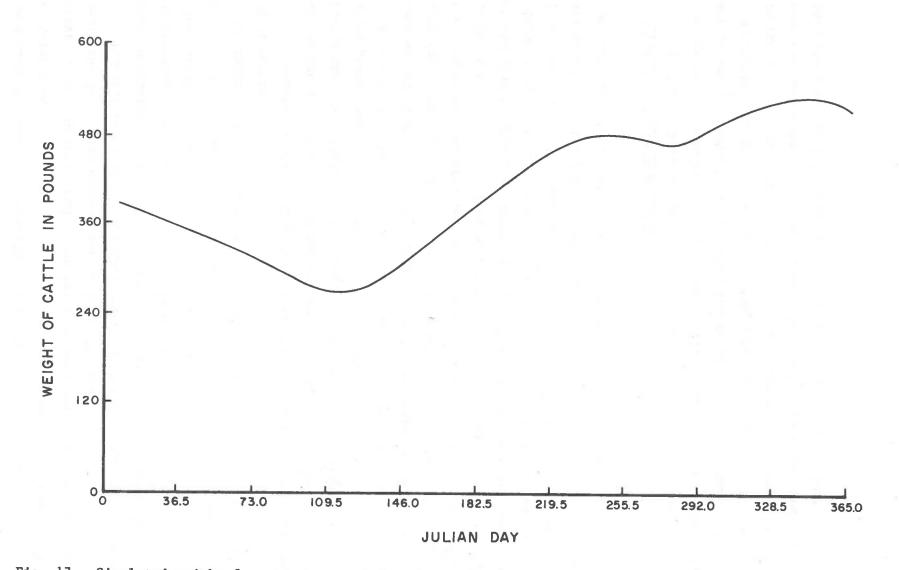
dead plant biomass which results. Early in the year, the previous year's standing dead plant biomass is decayed by the presence of spring moisture. After the phenological peak, standing dead plant material begins accumulation until the decay rate exceeds the accumulation rate.

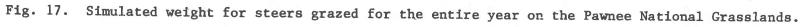
Temperature is used to regulate plant mortality with extremes of hot or cold increasing the death rate. Soil water is used as an index to the flow of standing dead plant material to litter in conjunction with temperature. Thus, standing dead plants decay rapidly in the spring when it is warm and wet, and in the fall, if precipitation occurs. Again, the presence of livestock contributes to litter accumulation as well as trampling loss to green plant material.

Simulated cattle grazing (30 steers averaging 400 lb each) was begun at day one and continued for the entire year (Fig. 17). Lack of available forage and low quality forage caused the cattle to lose weight rapidly until plant growth began. After this point they gained weight at a reasonable rate (1.6 lb/day) for the remainder of the growing season. At day 280 supplemental feed was given which kept the cattle from losing weight late in the season when forage was again at an inadequate level. Knowing the purchase price and the cost of supplemental feed allows the net cost of cattle to be determined, assuming no cost for grazing on private rangeland. Market value for each month of the year is generated stochastically (Fig. 18) in the model and allows net financial gain to be determined (Fig. 19). As many as five classes of ruminant livestock can be simulated per run; however, the market values presented here are based on cattle prices. Note that random number generators in different computers will result in different livestock prices and market values.

Example 2: Pawnee Grasslands

The Pawnee Site has sandy loam soils and a semiarid climate with an average yearly rainfall of approximately 11.7 inches. Warm season plants are generally predominant; however, the relatively large annual precipitation used in the simulation (14 inches) gave more cool season plant growth early in the season than warm season plants ever achieved (Fig. 20). This phenomenon has been observed at the Pawnee Site. Notice this run produced more than the previous run (Example 1) because livestock were absent.





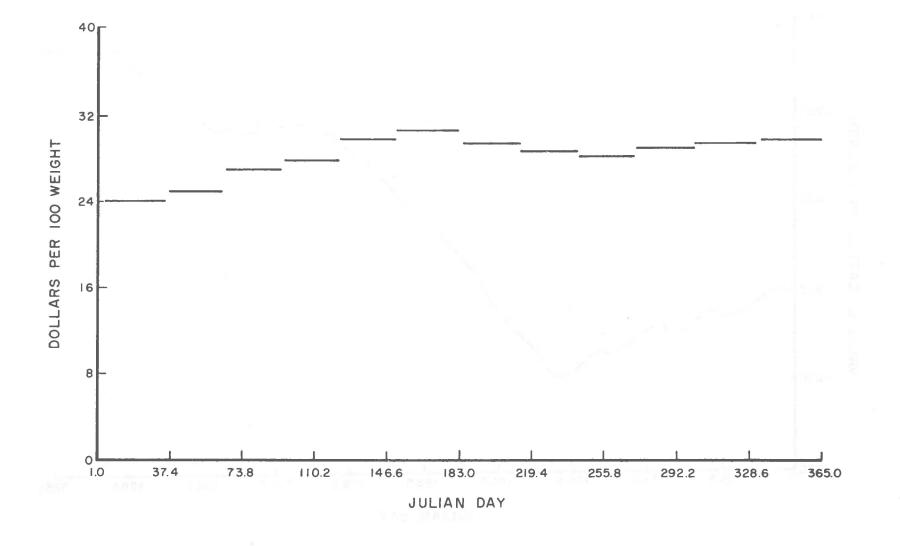


Fig. 18. Simulated stochastic market values for cattle.

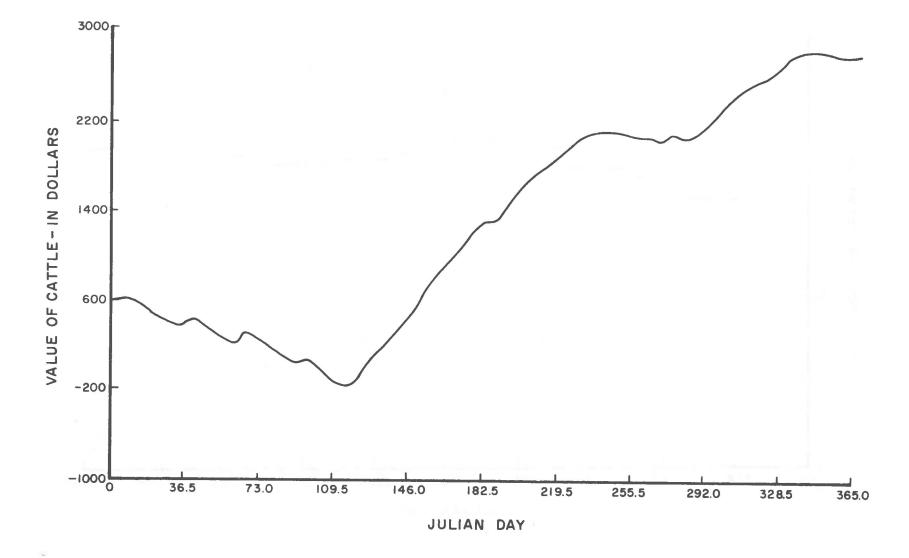


Fig. 19. Total market value of steers being grazed.

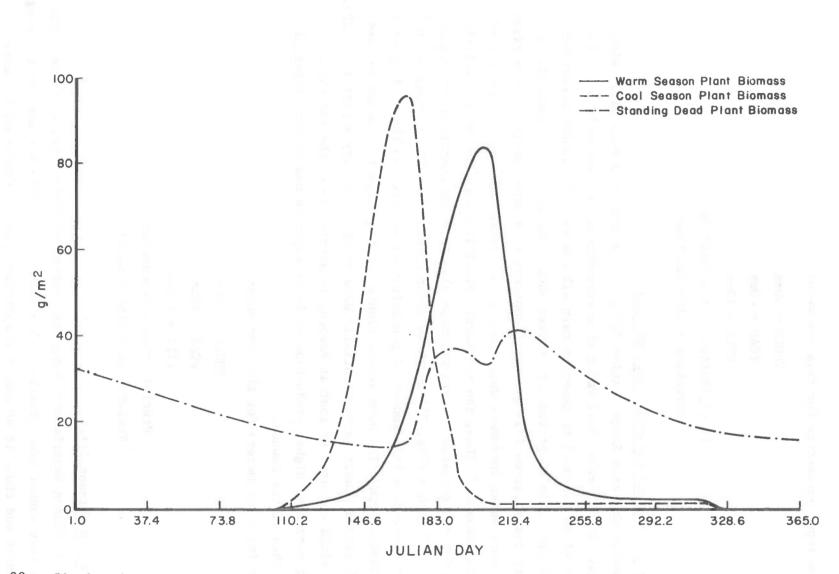


Fig. 20. Simulated cool and warm season plant production and standing dead plant biomass at the Pawnee National Grasslands without cattle grazing.

The input parameters for this run were:

THOLD = 50mm FCAP = 40mm WILT = 10mm Precipitation - sine function Temperature - sine function

Example 3: Eastern Colorado Range Station

Eastern Colorado Range Station (ECRS) is located on a sand hills site which gave it different soil water characteristics. A stochastic precipitation routine was used to generate rainfall events. The random occurrence of the events allowed periods of dryness when plant growth was reduced. The resulting regime gives less growth in contrast to a sine function which gives a daily soil water increment which only allows drying to occur at the end of the "rainy season." Thus, the stochastic precipitation gives more realistic results for the semiarid eastern Colorado climate. The precipitation regime yielded 15.7 inches (Fig. 21) which is about average for ECRS (average = 16.7 inches); however, a fair amount of precipitation occurred in the fall, giving late season growth. The warm season plants had the biggest response because the cool season plants were essentially dead after a hot, dry August (Fig. 22). Similar yield compared to that at Pawnee, is attributed to the stochastic weather; normally, higher production would be expected due to more rainfall at ECRS than at the Pawnee.

The input parameters for this run were:

THOLD = 35mm FCAP = 25mm WILT = 7.5mm Precipitation - stochastic Temperature - sine function

Example 4: Dickinson Site

The Dickinson Experimental Range is located in North Dakota. It receives slightly less annual precipitation (15.9 inches) than ECRS but has lower average temperatures and thus, lower evapotranspiration rates. Consequently, more

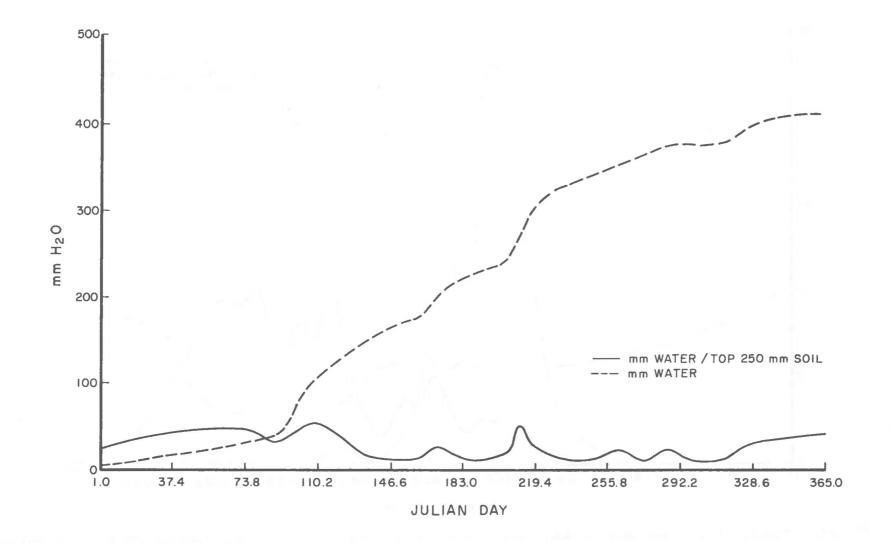


Fig. 21. Simulated stochastic precipitation and resulting soil water content at the Eastern Colorado Range Station.

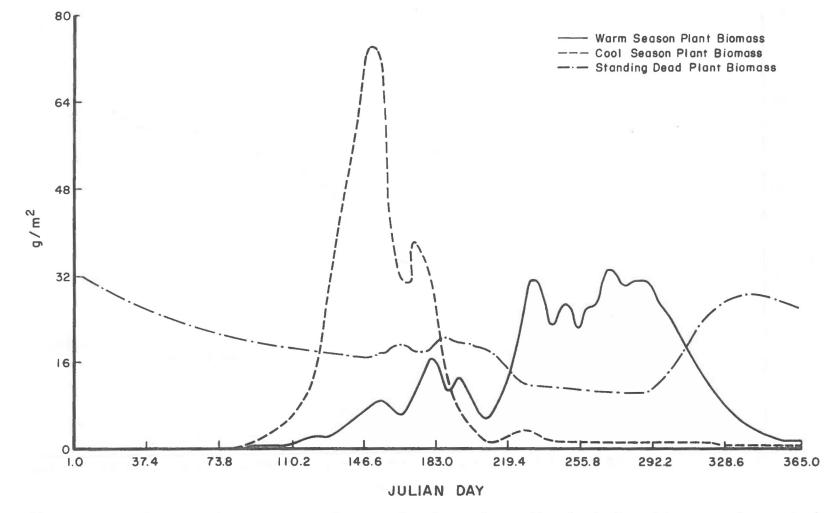


Fig.22. Simulated cool and warm season plant production and standing dead plant biomass under a stochastic precipitation regime without cattle grazing at the Eastern Colorado Range Station.

growth can be expected at the Dickinson Site. In fact, this is what happens both in the field and in the model results (Fig. 23). Another expectation would be an increased ratio of cool season to warm season plants, which is accurately simulated by the model. The soils at Dickinson are of the Flasher Vebar Complex.

Input parameters were:

THOLD = 38.5mm FCAP = 35mm WILT = 15mm

Precipitation - Fourier series representation of historical data (a sine function was not used because of multiple dips in the average precipitation curve) Temperature - sine function

CONCLUSIONS

The work reported here was based on a hypothesis regarding modeling ecosystems. The hypothesis is that complex ecosystem level models are, in general, too sophisticated to be readily usable by management personnel and that their heavy data requirements restrict the ease with which they can be used at different sites. Therefore, a management oriented model was proposed which would focus on the crucial mechanisms that were determined by ecosystem level modeling efforts in an attempt to represent a multitude of possible sites with only a minimal data requirement. The model developed from these hypotheses has been tried on three sites.

The submodel that is important in terms of representing a site adequately is the plant growth model. The livestock submodel will respond according to plant production so its variation is primarily a function of the plant model. The soil water submodel, on the other hand, is instrumental in simulating plant production. Therefore, the evapotranspiration and plant production functions must be very versatile and give reasonable results. The model was tested on three different sites for its ability to represent each site by changing only the specified input parameters. First attempts at running the

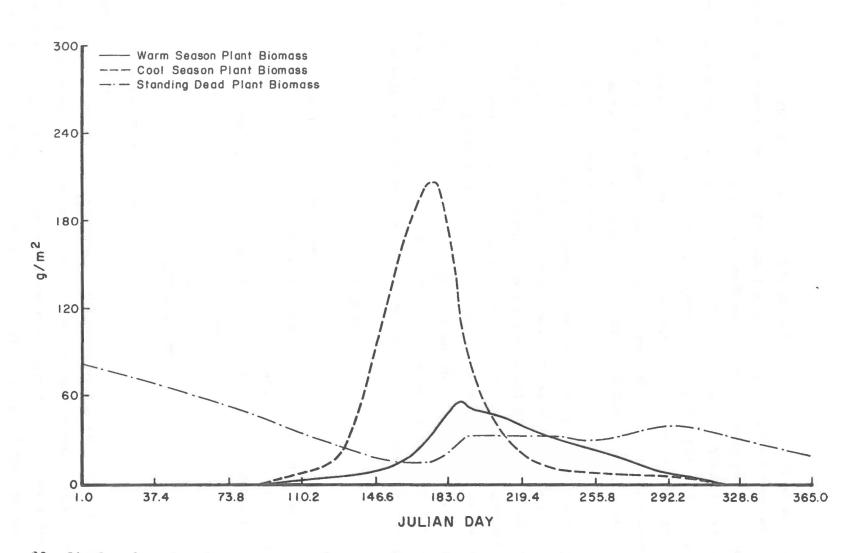


Fig. 23. Simulated cool and warm season plant production and standing dead plant biomass without cattle grazing at the Dickinson Experimental Range.

model at a new site using parameters tuned for the Pawnee site produced erroneous results. In order to adequately represent multiple sites compromises were made on some parameters. The resulting model gives reasonable results, but has some problems. For example, the simulated cool season plants exceed warm season plant production at the Pawnee site. This is not the usual case; however, under some conditions and on certain sites it does occur.

A second problem with the model is the timing of warm season growth. It begins growth at the appropriate time but often continues to produce well into the fall, a phenomenon not usually observed. Finally, the plant production at the Eastern Colorado Range Station appears a bit low considering the amount of rainfall received. However, given that the precipitation was stochastic, the effective moisture may not have been very high.

Such problems are to be expected when a model is built attempting to incorporate realism and generality at the expense of precision. The problems can usually be remedied by specifying some parameters for the particular site. But this should be done by a user and was not the intent of the modeling activity. Another possible solution to model inaccuracies is increased resolution in some of the mechanisms or additional mechanisms that may be important at a particular site.

With the results to date it would be impossible to state conclusively that a simple general model capable of handling a wide variety of sites can be built. On the other hand, it is not possible to say it cannot be done as this effort has shown. Rather it appears that with more knowledge about certain key mechanisms the results could be significantly improved on multiple sites and the hypothesis conclusively tested. A few of the important relationships that need further study follow.

Most general evapotranspiration functions are extremely complex and require many parameters that are not usually measured except at experimental sites. A general relationship with easily obtainable data inputs seems plausible with a semi-physical relationship such as evapotranspiration. Also some more general photosynthesis and respiration rate information would be helpful. Information currently available is species specific and it would be useful to know what factors are responsible for the observed variations in response trends.

If relationships such as the above can be developed then there is a good chance for simple general models; otherwise models will have to be site specific to be useful for management.

.

LITERATURE CITED

- Bement, R. E. 1969. Stocking rate guide for beef production on blue grama range. J. Range Manage. 22:83-86.
- Brown, L. W. 1974. Photosynthesis of two important grasses of the short grass prairie as affected by several ecological variables. Ph.D. Dissertation, Colo. State Univ., Ft. Collins. 191 p.
- Connor, D. J., L. F. Brown and M. J. Trlica. 1974. Plant cover, light interception and photosynthesis of shortgrass prairie - a functional model. Photosynthetica (unpubl.).
- Forrester, J. W. 1961. Industrial dynamics. MIT Press, Cambridge, Mass. 464 p.
- Forrester, J. W. 1968. Principles of systems. Wright-Allen Press, Inc., Cambridge, Mass.
- Hedrick, H. B. 1968. Bovine growth and composition. Miss. Univ. Agr. Exp. Sta. Bull. 928.
- Heermann, D. F., M. D. Finkner and E. A. Hiler. 1971. Probability of sequences of wet and dry days for eleven western states and Texas. Colo. State Univ. Exp. Sta., Ft. Collins. Tech. Bull. 117.
- Hume, I. D., R. J. Moir and M. Somers. 1970. Synthesis of microbial protein in the rumen. I. Influence of the level of nitrogen intake. Aust. J. Agr. Res. 21:283.
- Innis, G. S. 1972. The second derivative and population modeling: another view. Ecology 53(4):720-723.
- McGinnies, W. G. and J. F. Arnold. 1939. Relative water requirement of Arizona range plants. Ariz. Agr. Exp. Sta. Tech. Bull. 80:167-246.
- Naylor, T. H., J. L. Balintfy, D. S. Burdick and K. Chu. 1966. Computer simulation techniques. John Wiley and Sons, Inc., New York. 340 p.
- Rasmussen, J. L., G. Bertolin, and G. F. Almeyda. 1971. Grassland climatology of the Pawnee Grassland. US/IBP Tech. Rep. No. 127. Colo. State Univ., Ft. Collins. 79 p.
- Rice, R. W., J. G. Morris, B. T. Maeda and R. L. Baldwin. 1974. Simulation of animal functions in models of production systems: ruminants on the range. Federation of Am. Soc. for Experimental Biology. Proc., Vol. 33, No. 2.

- Singh, J. S. and D. C. Coleman. 1974. Evaluation of functional root biomass and translocation of photoassimilated carbon-14 in a shortgrass prairie ecosystem. In J. K. Marshall [ed.] The Belowground Ecosystem: A Synthesis of Plant Associated Processes. (unpubl.).
- Todorovic, P. and D. A. Woolhiser. 1971. Stochastic model of daily rainfall. Proc. of the USDA-IASPS Symp. on Statistical Hydrology, Tucson, Ariz.

Whitman, W. D. 1974. Standing dead component in mixed grass prairie at the Dickinson Site. (unpubl.).

APPENDIX 1

Variable names, definitions and units

	Name	Description	Unit
	A1	The first parameter of a Fourier series - the mean value over the time interval	variable
	A(I)	Array of Fourier sine coefficients for the terms remaining in the series I=1,2.	variable
	AEN	Available energy for gain	kcal/day
	AFAC	Grams microprotein growth per kcal digestible organic matter	g/kcal
	ANOS(I)	Actual number of animals of class I, I=1,5	numbers
	ANTOT	Total livestock numbers of all classes currently grazing	numbers
	AREA	Size of grazing area - input in acres converted to m * m	m * m
	ASZ	Proportion of mature size per livestock class	proportion
	AVAIL	Forage availability	proportion
	В	Grams of water needed to produce 1 gram of plant biomass	g H ₂ 0/g plant
	B(I)	Array of Fourier cosine coefficients - I=1,2	variable
	BASE(I)	Price paid for stock when they are put on the pasture. I=1,5	\$
	CAP	Total rumen capacity	kg/day
	CASH(I)	Amount invested in class I, I=1,5	\$
	CBM	Cool season green plant biomass	g/m ²
	СН	Chlorophyll in green forage	mg/g
	CMAX	Maximum cool season plant photosynthesis per day	g Ps/g/day
	CP	Proportion of warm season plants for total growth	proportion

a.

Name	Description	Unit
CRMAX	Maximum cool season plant respiration rate	g Rs/g/day
D	Percentage of green and dry forage that is digestible; second computation includes supplemental feed	proportion
DDAY	Julian day of the year, same as IDAY only a real variable	days
DG	Digestible component of all intake	kg
DIG	Total potential digestion, the sume of residual material digested from previous time step and current digested material	proportion
DIIT	Nitrogen content of animal diet	proportion
DMP	Microprotein growth	g/day
DMX	Green plant death loss	g lost/g/day
DOM	kcal contained in food eaten and digested	kcal/day
DPF	Effect of soil water on dead plant disappearance	proportion
DRDM	Change in rumen dry matter	kg/day
DRTD	Effect of temperature on disappearance of dead plant material	g lost/g/day
DS	Proportion of supplemental feed to total ingestion	proportion
DT	Solution time step for integration	days
DTFL	Time step between flow printouts	days
DTPL	Time step between plot value storage	days
DTPR	Time step between printouts	days
ECR	Effect of temperature on cool season respiration	proportion
EFAC	Proportion of volatile fatty acids to fermented energy	proportion
EO	Potential evapotranspiration	mm H ₂ 0/250mm soil

Descr	iption

Name

E

Unit

ESMC	Effect of soil water on cool season photosynthesis	proportion
ESMW	Effect of soil water on warm season photosynthesis	proportion
ET	Actual evapotranspiration	mm H ₂ 0/250mm soil
ETC	Effect of temperature on cool season photosynthesis	proportion
ETODC	Effect of temperature on green plant death for cool season plants	proportion
ETODW	Effect of temperature on green plant death for warm season plants	proportion
ETW	Effect of temperature on warm season photosynthesis	proportion
EVAP	Evapotranspiration	mm/m ² /day
EWR	Effect of temperature on warm season respiration	proportion
EXR	Exit rate - combination of digested intake and constant passage rate, constrained to be less than .95	proportion
EYR	The rate at which eaten food passes through the digestive tract	proportion
FCAP	Soil water field capacity	mm H ₂ 0/250mm soil
FCONS	Total forage consumption	g
FRDR	Fermentation digestion rate	proportion
G	Percentage of the forage that is green plant material	proportion
GAIN(I)	Animal weight gain by livestock class I, I=1,5	g/m ² /day
GAN	Animal weight gain	g/m ² /day
GC	Cool season plant growth returned to XFLWS from GROW	g/m ²
GFAC	kcal energy required for 1 gram of gain	kcal/gram
GN	Nitrogen content of green forage	proportion

Description

Name

Unit

Ĩ1

GNIN(I)	Intake of live forage by livestock class I, I≖1,5	g/day
GNN	Intake of live forage	g/day
GPLUSD	Total green and dry forage present	g/m ²
GR	Total cool and warm season growth used for con- straining growth to a rate compatible with ET	g/m ² /day
GROW	A FORTRAN function subroutine	
GW	Warm season plant growth returned to XFLWS from GROW	g/m ²
H1	Amplitude of sine wave divided by two for precipitation	inches
H2	Sine wave displacement along time axis for precipitation	days
НЗ	Sine wave displacement along vertical axis for precipitation	inches
Н4	Amplitude of sine wave divided by two for temperature	degrees
Н5	Sine wave displacement along time axis for temperature	days
Н6	Sine wave displacement along vertical axis for temperature	degrees
IDAY	Julian day of the year	days
IG	Ingestion rate	kg/day
IME	Time of simulation, same as TIME only an integer variable	days
IN(I)	Stocking day for livestock class I, I=1,5	day
INFD(I)	Beginning date of supplemental feed for livestock class I, I=1,5	day
INYR	First calendar year in which weather data is read from a tape (MODT = 1)	year
IYR	Year of simulation	year

Name	Description	Unit
JUDAY	Julian day read from weather tape	day
K	Counter which keeps track of days elapsed since growth initiation	days
KC	Counter to allow five days growth initiation for cool season plants	days
KW	Counter to allow five days growth initiation for warm season plants	days
KYR	Year of recorded data read from weather tape	years
LVWT(I)	Average individual animal weight. Input is in pounds, converted to grams in START. Total animal weight is found in each flow by multiplying by STKNO(I)/AREA	
LWT(I)	LVWT in pounds	pounds
MEN	Energy derived from microprotein	kcal/day
MIPRO	Microprotein level in the rumen	g
MIPROW	Microprotein per metabolic size	g/kg
MISS	Indicates when data is missing from weather tape	dimensionless
MKT(I)	Market day for livestock class I, I=1,5	day
MODP	Indicates method used for computing precipitation	dimensionless
	<pre>1 = read from a tape 2 = compute from sine function 3 = compute using stochastic generator</pre>	
MODT	Indicates method used for computing temperature	dimensionless
	<pre>1 = read from a tape 2 = compute from sine function 3 = compute from sine function</pre>	
MON	Month of the year	months
MPAS	Microprotein passage rate	g passed/g/day
MSZ	Mature size of livestock	g

Description

Name

Unit

N	Size of PRICE vector and PRCOV matrix	dimensionless
NAME	Station or ranch name for which weather parameters entered for stochastic generator	hollerith
NCLAS	The number of animal classes	numbers
NDIET	Percentage nitrogen in the diet, including supplement	proportion
NFOR	Percentage nitrogen in the forage	proportion
NFRDR	Proportion of intake digested per gram of microprotein, normal fermentation digestion rate	proportion
NGN	Nitrogen content of live forage	proportion
NITRO	A subroutine	
NOFD(I)	Ending date of supplemental feeding for livestock class I, I=1,5	days
NRA	l if no rain on IDAY, 2 if rain occurred; can be used to lower temperature when rain occurs	dimensionless
NRM	A subroutine	
NT	Number of cycles used in Fourier series, generally 2 if FREGEN (Appendix 5) data used	dimensionless
0	Percentage of forage that is dry plant material	proportion
OMIN(I)	Intake of standing dead forage by livestock class I, I=1,5	g/day
OMN	Intake of standing dead forage	g/day
Ρ	P(W/D) or P(W/W) depending on the occurrence of a storm the preceding day	proportion
PASFD	Passed food	kg/day
PEN	Energy obtained from passed food	kcal/day
PFL	Cumulative precipitation	mm
PH	Phosphorus content of forage	proportion

Name	Description	Unit
PHOS	Percentage phosphorus content in the diet	proportion
PP	Precipitation values read from weather tape	inches
PPT	A FORTRAN function subroutine	
PR	Protein content of live forage	proportion
PRC	Monthly livestock prices	\$/cwt
PRCOV(I)	Variance-covariance matrix for livestock class net prices: I, I=1,144	
PRICE(I)	Actual net price for livestock classes I, I=1,12 computed from PRMN(J) and PRCOV(I) for random normal deviates	\$/cwt
PRMN(I)	Mean monthly net price for livestock classes I=1,12	\$/cwt
PROT	Forage crude protein content	proportion
PT	Daily precipitation	mm
PW	Plant water	proportion
QO	Probability of a dry day preceded by a dry day; converted to probability of a wet day preceded by a dry day	probability
Q1	Probability of a dry day; converted to probability of dry day preceded by a wet day; converted to a probability of a wet day preceded by a wet day	probability
QC	Scaling parameter for ESMC, a function of FCAP and WILT	dimensionless
QW	Scaling parameter for ESMW, a function of FCAP and WILT	dimensionless
R	Uniformly distributed random variable	proportion
RDM	Rumen dry matter	kg
RDMD	Rumen dry matter digestible	kg/day
RFL	Rumen fill	kg/day

Description

Unit

Name

RG Relative growth rate of live forage proportion Relative growth rate of live forage RGR proportion RUMEN A subroutine S Dummy variable computed in SUBROUTINE SER, used for QO, Q1 and XLAM dimensionless SD Nitrogen content of dry forage proportion SG Supplemental feed ingested kg/day SITE Station name for weather tape hollerith SOILA(SLA) Input parameter for soil nutrient modifications SOILB(SLB) Input parameter for soil nutrient modifications STKNO(I) The number of animals in livestock class I, I = 1,5 numbers SUPL Proportion of supplemental feed to animal weight proportion SUPN Nitrogen content of supplemental feed proportion SUPP(I) Total price of supplemental feed for livestock class I, I=1,5 \$ SUPPR(I) Price of supplemental feed for livestock class I, I=1,5 \$/1Ъ SUPRT(I) Supplemental feed rate for livestock class I, I=1,5. Proportion of body weight proportion SUPWT(I) Total supplemental feed given to livestock class I, I=1,5 1bs. SW Soil water mm $H_20/250$ mm soil °C Т Mean 5 day temperature T1 Variable to save the last good maximum daily temperature value in case a missing value is encountered when reading a weather tape °F

Name	Description	Unit
Τ2	Variable to save the last good minimum daily temp- erature value in case a missing value is encountered when reading a weather tape	°F
TEMP	A FORTRAN function subroutine	
TEND	Ending time of simulation	days
THOLD	Total water holding capacity of soil	mm H ₂ 0/250mm soil
TINIT	Parameter used to initialize random number generator	odd number
TMAX	Maximum daily temperature read from a weather tape	°F
TMIN	Minimum daily temperature read from a weather tape	°F
TOTAL	<pre>Sum of VALUE(I), I=1,5. Each value taken at MKT(I), I=1,5</pre>	
TP	Same as TP(I) in subroutine RECRD - current average daily temperature	°F
TP(I)	Stack of the past five days mean temperature used to compute five day average	°C
TR	Mean daily temperature	°C
TSTRT	Starting time of simulation	day
UNDFD	Undigested food	kg/day
V	Vitamin A in live forage	mg/gm
VALUE(I)	Amount received for livestock class I, I=1,5 on market day	Ş
VFA	Energy from volatile fatty acids	kcal/day
VITA	Vitamin A in live forage	mg/gm
VL75	Metabolic size, based on kg body weight for average sized animal from livestock class I, I=1,5	kg
WBM	Warm season green plant biomass	g/m ²

Name	Description	Unit
WILT	Soil wilting point	mm H ₂ 0/250mm soil
WMAX	Maximum warm season plant photosynthesis per day	g Ps/g/day
WP	Percentage of total growth made up by warm season plants	proportion
WRMAX	Maximum warm season plant respiration per day	g Rs/g/day
X1011	Sum of the state variables X(10) and X(11) used to determine when phenological peak has been reached	g/m ²
X10T0	The amount of X(10) present at the previous time step	g/m ²
X11T0	The amount of X(11) present at the previous time step	g/m ²
XLAM	Daily lambda parameters generated by a Fourier series which when substituted into an experimental transformation yield storm size predictions	inches
ZZ(I)	Intermediate computation for price generation, I=1,12	dimensionless
	State Variables	
X(1)	Livestock class l	g/m ²
X(2)	Livestock class 2	g/m ²
X(3)	Livestock class 3	g/m ²
X(4)	Livestock class 4	g/m ²
X(5)	Livestock class 5	g/m ²
X(10)	Warm season green plant biomass	g/m ²
X(11)	Cool season green plant biomass	g/m ²
X(12)	Standing dead plant biomass	g/m ²
X(21)	Soil water	mm H ₂ 0/250mm soil

58

0

1

\$

X(31)	Atmospheric water source	mm
X(32)	Water sink	mm
X(34)	Respired gases and litter	g/m ²
X(J)	J=41, 45 VALUE(I), I=1,5	
X(J)	J=46, 50 BASE(I), I=1,5	
X(J)	J=51, 55 SUPP(I), I=1,5	
X(J)	J=56, 60 CASH(I), I=1,5	
X(J)	J=61, 65 SUPWT(I), I=1,5	
X(J)	J=66, 70 LWT(I), I=1,5	
X(71)	SEASN	
X(72)	Τ	
X(73)	GN	
X(74)	РН	
X(75)	PR	
X(76)	СН	
X(77)	V	
X(78)	PW	
X(79)	FCONS	
X(80)	PFL	
X(J)	J=81, 85 ANOS(I), I=1,5	

Function Subroutines

Name	Description	Argument List
ET	Calculates evapotranspiration	(SW, TR, FCAP, THOLD, WILT)
PPT	Calculates precipitation	(IDAY, MODP, TIME, TSTRT, NRA)
TEMP	Calculates temperature	(IDAY, MODT, NRA, TIME, TSTRT)
	Subroutines	
GROW	Calculates plant growth	(WBM, CBM, SW, TR, SLA, SLB, GW, GC, FCAP, WILT)
NITRO	Calculates nitrogen content of the forage	(RGR, NGN, PH, PROT, CH, VITA, PW)
NRM	Stochastic price generator	(PRICE, PRCOV, PRMN, N, ZZ)
RECRD	Read a weather tape	(TIME, TSTRT, IDAY, INYR, TP, P)
RUMEN	Calculate forage intake and animal gain	(LVWT, SUPN, SUPL, OMN, GNN, GAN, MSZ, RDM, RFL, RDMD, MIPRO, VL75)
SER	Generates a Fourier series given Fourier coefficients	(NT, A1, A, B, S)

APPENDIX #2

FORTRAN listing of the RANGES grassland simulation model.

PHUGHAM HANGES

TITNPUT + UUTPUT +PLISV + TAPES= INPUT + TAPE += UUTPUT + TAPE 3=PL TSV + TAPE4+

21AHE7)

č

С

C C

C

Č

С

C č

Ć

C C

C

C

c

C

С

c

c

c

c

C

Ċ

C

č

CCCC

Č CC

CCCCC

CCCC

CCCC

C С

č							C
č	XX XX	XX XX	33 33	XX XX	XX XX	XX	C
č	XX XX	XX XX	AX XK	AX XX	XX XX	XX XX	C
č	XXXX	XXXX	2222	****	XXXX	XXXX	C
č	*******	********	********	*******	*******	********	C
č	********	********	*********	*********	*******	*******	č
č	****	****	2222	****	XXXX	****	C
č	XX XX	XX XX	XX XX	XX XX	XX XX	XX XX	C
C	XX XX	XX XX	×× ××	XX XX	XX XX	XX XX	C
C							C
C							С
С	*****	*******	XX XX	XXXXXXXA	*****	*****	С
С	********	*******	XXX XX	*********	*******	******	C
С	XX XX	XX XX	XXX XX	XY XX	XX	XX	C
C	XX XX	XX XX	XX X XX	XX 1	XX	XX	C
С	XX XX	XX XX	XX XX XX	XX	хх	XX	C
С	*********	*******	XX X XX	XX XXXX	*****	XXXXXXXXX	С
С	*******	*******	XX X XX	XX XXXXX	*****	********	C
С	XX XX	XX XX	XX A XX	XX XX	ХХ	XX	С
С	XX XX	XX XX	XX XX XX	XX XX	XX	**	С
С	XX XX	XX XX	XX XXXX	XX XX	XX	XX	C
С	XX XX	XX XX	XX XXXX	******	******	********	С
С	XX XX	XX XX	XX XXX	XXXXXXXX	******	XXXXXXXX	C C
С							
С							С
С	XX XX	XX XX	XX XX	XX XX	XX XX	XX XX	c
С	XX XX	XX XX	XX XX	XX XX	XX XX	** **	C
С	****	XXXX	XXXX	****	XXXX	XXXX	С
С	******	********	*****	*******	*****	*********	C C
С	*******	*******	*****	********	******	*******	
C	****	XXXX	XXXX	XXXX	XXXX	XXXX	C
С	XX XA	XX XX	XX XX	XX XX	XX XX	XX XX	С
C	XX XX	XX XX	XX XX	XX XX	XX XX	XX XX	С
С							C

THIS PROGRAM WAS ADAPTED BY BRAD J. GILBERT FROM A MODEL DEVELOPED BY GEORGE S. INNIS AND RICHARD MISKIMINS. THE INPUT AND MUTPUT SUBBOUTINES WERE WHITTEN BY JON GUSTAFSON.

C

C

C

C

С

C

C

C

C

C

Ċ

C C

C č

Č

С

C č

С C

C

С

С

C

C

CCC

C

C

C

č

C C

C

C Ĉ

C

č

C C

PURPUSE -THE DEVELOPMENT OF A GRASSLAND ECOLOGICAL SIMULATION MODEL COMPOSED OF A PLANI GROWIN SYSTEM. SOLL WATER SYSTEM. ECONOMIC SYSTEM. AND A CONSUMER SYSTEM USED TO INVESTIGATE VARIOUS MARKETING STRATEGIES ALONG WITH CATTLE MANAGEMENT PHACTICES AND TO SHOW THE EFFECTS OF THIS ENTINE CULLECTION ON INCOME.

PROGRAM UNITS-(MAIN PROGRAM) - RANGES (SUBPROGRAM) - HLOCK DATA

(SUBHOUTINES) - CYCLE, CYCLI, CYCL2, GROW, NITRO, NRM, RECRD, RUMEN, SEP, START, AFLWS, XGRAPH, XOUT, XPLGEN, XPLOT, XPPINI, KEND, XUNPAK

(USEN DEFINED FUNCTIONS) - ET+ PPT, TEMP (CONTROL DATA UTILITY FUNCTIONS) - MANE (CONTROL DATA UTILITY SUPROUTINES) - PANSET

(EXTERNAL ANSI FUNCTIONS) - ALOGO CUSO EXPO SINO SORT (INTHINSIC ANSI FUNCTIONS) - ARSO AMAXIO AMINIO INTO MOD ABS FINDS THE ABSOLUTE VALUE OF A REAL ARGUMENT.

ALOG FINDS THE NATURAL LUGARITHM OF A REAL VAPIABLE. AMAXI CHOOSES THE LANGEST VALUE AMONG A LIST OF REAL ARGUMENTS. AS A RESULT THE NUMBER OF ARGUMENTS MUST RE TWO OR GREATER.

AMINI CHONSES THE SMALLEST VALUE AMONG A LIST OF REAL ARGUMENTS. AS A PESULT THE NUMBER OF APGUMENTS MUST BE TWO OH GHEATER.

COS IS THE THIGUNUMETHIC COSINE WITH THE ARGUMENT IN MADIANS AND TYPE HEAL. EXP IS AN EXPONENTIAL FUNCTION WITH A REAL ARGUMENT.

INT IS A FUNCTION IMAT THUNCATES THE VALUE OF A REAL

VAPIABLE TO BECOME AN INTEGER VARIABLE. MOD COMPUTES THE REMAINDER FROM AN INTEGER DIVISION OF INTEGER ARGUMENTS.

PANE IS A RANDUM NUMBER GENERATOR WHICH RETURNS VALUES WANE IS A MANDUM NUMBER GENERATOR WHICH WEITHES VALUES UNIFORMETY DISTRIBUTED OVER THE INTERVAL REIMEEN ZERO AND ONE. INCLUDING ZERO HUT NOT INCLUDING ONE. THE ARGUMENT IS JUST A DUMMY WHICH ENDS UP HEING IGNORED. PANSET INITIALIZES A GENERATIVE VALUE FOR PANE. THE

ARGUMENT IS A HIT PAITERN WITH ATT ZERO SET TO 1 (FORCED OUD) AND BITS 59 THROUGH 46 SET TO 1717 OUTAL. SIN IS THE THIGONGMETRIC SINE WITH THE APGUMENT IN

RADIANS AND TYPE HEAL.

SORT CALCULATES THE SQUARE ROOT OF JIS HEAL ARGUMENT.

VADIANCE DEFINITIONS - SOME VARTABLE REFINITIONS APPEAR HELDE WHILE OTHERS ARE DEFINED IN THE OSERS GUIDE TO THE MANGES GRASSLAND SIMULATION NULFL. EXCEPT FOR SOME SUBSCRIPTS ALL THE VANTAMENS ARE DEPINED. - HEAL ANNAY CUNTAINING THE SAME VALUES AS THE DUM. AMPAY (A) IN THE SUMMOUTINE (APPLINT). - WEAL VARIABLE CONTAINING A TEMPUKANY STOPAGE FLOW THE AMOUNT OF MATENTAL PER UNIT TIME ENTERING ON LEAVING & LEVEL ON STATE VANIABLE. THE NE - INTEGER VARIABLE CONTAINTING & BLANK TO USE FOR CREATING THE PLOIS. - INTEGER VARIABLE CONTAINING & TEPPORARY VALUE REFORE ICHR PLACING INTO THE OUTPUT CHAMACTER STRING (IP). - INTEGER VARIABLE CONTAINING THE LMAPACTER (-) TO USE EUP CREATING THE PLOTS. TUASH - INTEGER VARIABLE CONTAINING THE LETTER (1) TO USE IF YEE FOR CREATING THE PLOTS. IF - INTEGER VARIABLE CONTAINING THE INDEX OF SOURCE COMPARTMENT IN THE SUPPORTINE (AUMPAN). INTEGER VARIABLE CONTAINING THE COMPUTER WORD INFU PASSED TO THE SUBBOUTINE (ZUMPAN) FOR PROCESSING. INTEGER ARRAY CONTAINING THE OUTPUT CHARACTER IP STHING WITH HLANKS AND GRAPHICAL REFERENCE LINES FOR THE PLOTTING OF STATE VARIABLES IN THE SUHHOUTINE (XGHAPH) . INTEGER VARIABLE CONTAINING & FLOW PRINT FLAG IN IP THE MAIN PRUGRAM (PANGES) AND THE SURROUTINE (XUNPAK) . - INTEGER VARIABLE CONTAINING THE CHAPACTER (=) IQUAL TO USE FOR CREATING THE PLUTS. ISTUP - INTEGER VARIABLE CONTAINING A FLAG TO SIGNAL THE LAST RECORD OF STURED PLOITING DATA. - INTEGER VAPIABLE CONTAINING THE INDEX OF II. DESTINATION COMPARTMENT IN THE SUBMOUTINE (XUNPAK). INTEGER VARIABLE CONTAINING THE WORD IN THE IV SURROUTINE LAUNPAK) TO BE UNPACKED. - INTEGER VARIABLE CONTAINING THE COMMAND VERBS INPUT BY THE CAPD READER FOR DUIPUT CONTROL. **IVERB** - INTEGER VARIABLE CONTAINING THE TRUNCATED VALUE FOR THE VARIABLE (ZI) USED IN SCALING THE Y AXIS 17 OF PLOTS GENERATED. - INTEGER VARIABLE CONTAINING AN INDEX OF SOURCE 11 CUMPARTMENT . 12 INTEGER VARIABLE CONTAINING AN INDEX OF DESTINATION COMPARTMENT. - INTEGER ARRAY CONTAINING THE POSSIBLE NUMERIC VALUES JCHAK REPRESENTING THE VARIARLES PLOTIED. - INTEGER VAPIABLE CONTAINING A SUBSCRIPT VALUE FOR JNT PRINTING THE & AXIS INDICES ON THE PLOTS GENEPATED. - INTEGEP VARIABLE CONTAINING & TEMPOPARY VALUE FROM JP THE OUTPUT CHARACTER STRING (IP). JI - INTEGER VARIABLE CONTAINING THE INITIAL SUBSCRIPT IN LISTING THE STATE VARIABLES FROM THE ARRAY (VAL) FOR EACH LINE. - INTEGER VARIABLE CONTAINING THE ENDING SUBSCRIPT J2 IN LISTING THE STATE VARIABLES FROM THE ARRAY (VAL) FOR FACH LINF. - INTEGER ARRAY CONTAINING THE POSSIBLE COMMAND KEY VERHS TO CHECK AGAINST THE VARIABLE (IVERB). - INTEGER VARIABLE CONTAINING A VENTICAL CONTROL KNT FACTOR FOR BUILDING THE ENAME OF THE GRAPH USED TO PLOT THE STATE VARIABLES. - INTEGER VARIABLE CONTAINING THE LOCATION POINTER LOC IN THE STATE VARIABLE ARRAY (7) FOR PLOTTING PURPOSES. INTEGER VARIABLE CONTAINING & TEMPOPARY STOPAGE м VALUE USED TO ADJUST THE VALUE OF ARRAYS (IN ANO MK1). NEL - INTEGER VARIABLE CONTAINING A SPECIAL FLAG TO DETERMINE FLOW PRINTOUTS BY THE TIME STEP RETWEEN THE FLOW PRINIOUTS. - INTEGER VARIABLE CONTAINING THE NUMPER OF STATE NKNT VARIAHLES PER LINE IN THE LISTING. - INTEGER VARIABLE CONTAINING THE NUMBER OF LINES MAKING A LIST OF STATE VARIABLES WITH NLINE FOUR VARIABLES PER LINE. - INTEGER VARIABLE CONTAINING THE NUMBER OF STATE VARIABLES TO BE PLOTTED. NN INTEGER VARIABLE CONTAINING & SPECIAL FLAG TO NPL DETERMINE PLOTTING TIME IN THE MAIN PROGRAM (RANGEA) OR THE PLOT NUMBER IN THE SUBROUTINE (XGRAPH). - INTEGEP VARIABLE CONTAINING & SPECIAL FLAG TO NPR DETERMINE TIME FOR PRINTOUTS. - INTEGER ARRAY CONTAINING THE INVICES OF THE NUM STATE VARIABLES REQUESTED FOR PRINTING ON THE PRINT OUTPUT CONTPOL CARD. NVAHS - INTEGER VARIABLE CONTAINING THE NUMBER OF VARIABLES PER PLOT. NVP - INTEGER VARIABLE CONTAINING THE NUMPER OF VARIABLES PER PLOT. - INTEGER MATRIX CONTAINING THE LUCATION OF FACH NXL OC VARIANLE IN THE LIST OF STATE VANIARIES FOR

63

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C.

C

C

C

C

C

c

C

C

C

C

C

C

C

C

C

C

Ĉ

C

C

C

C

C

C

C

Ĉ

C

C

C

C

C

C

C

1

С

С

C C

C C

C

C C

Ċ

c

C

c

C

c

С

С

c

Ċ

С

С

C

С

С

C C

С

C

c

č

¢

C C

č

C C C C

С

č

С

C

C

С

C C

C

C

C

¢

Ċ

С

č

С

Ç

C

C

С

С

CCC

C

C

č

C

С

C

С

С

¢

С

C

С

С

C

č

C

CC

C C

С

С

C

Ĉ

EACH PLUT.

NEPI - INTEGEN ANNAY CONTAINING A LIST OF STATE VANTABLES TO HE PLUTTED. - REAL VARIABLE CONTAINING EITHER CORPORT DAY WEATHER 14 INFORMATION OF PAST GOOD WEATHER DATA. - REAL VAPIANIE CONTAINING A TEMPUHARY VALUE ADDED TO OF SUMTRALIED FROM A STATE VAPIANCE. 0 - REAL VARIABLE CONTAINING THE FRUNCATED VALUE OF THE VARIABLE CONTAINING THE SUBJECT PURPOSES. - REAL VARIABLE CONTAINING THE EXTREME MAXIMUM HNVAR HN7MAX VALUE FOR THE GENERATED PLUTS. REAL VARIABLE CONTAINING THE EXTREME MINIMUM HNZMIN VALUE FOR THE GENERATED PLOTS. - REAL VARIABLE CONTAINING A SUMMAITON VALUE USED IN THE EXPANSION OF FOURIER SERIES. SUM TIME - HEAL VANIANLE CUNTAINING THE CUMMENT VALUE OF SIMULATED)IME. TIMEFT - REAL VARIABLE CONTAINING THE TIME VALUE USED AS A FLAG TO SIGNAL PENNING THE FLOWS. HEAL VAPLABLE COVIAIDING THE TIME VALUE USED AS A TIMPPL FLAG TO SIGNAL PLOTTING. TIMEPH - REAL VARIABLE CONTAINING THE TIME VALUE USED AS A FLAG TO SIGNAL PHINTING OF THE STATE VARIABLES. - HEAL VARIANLE CONTAINING THE GENERATIVE VALUE FOR TINII THE RANDOM NUMHER GENERATUR. - REAL VARIANCE CONTAINING THE TEMPORARY VALUE FOR THE NEXT TIME STEP USED IN PPINEING THE FLOWS. TM REAL ANNAY CUNTAINING THE WORKING STOPAGE VALUES VAL USED IN OUTPUT GENERATION. VAD - REAL VARIABLE CONTAINING THE RANGE OF THE MINIMUM AND MAXIMUM VALUES IN SCALING FOR THE PLOTS. REAL APPAY CONTAINING THE CURPENT VALUES OF FLOWS. XF XFU INTEGER ARRAY CONTAINING THE FLOW REFERENCE TABLE WHERE THE COMPARTMENTAL INDICES OF THE FLOWS ARE STORED ACCORDING TO THE FOLLOWING FORMULA IPP10000 FLOW PRINT FLAG. 11º100 INDEA OF SUURCE COMPARTMENT. 12 INDEA OF DESTIMATION COMPARTMENI. - REAL VANIABLE CONTAINING THE PEAL VALUE OF A XT SURSCHIPT (1) USED IN THE FAPANSION VARIANLE OF THE FOURTER SERIES. PEAL VAPIABLE CONTAINING THE INCREMENTAL VALUE HETWEEN INDICES OF THE X 4XIS ON THE PLOTS GENERATED. XINC XLINE - REAL APRAY CUNTAINING THE INDICES FOR THE X AXIS ON THE PLOTS GENERATED. XMAX - REAL VARIABLE CONTAINING A MAXIMUM VALUE FOR THE X AXIS UN THE PLUIS GENERATED. XMIN REAL VARIABLE CONTAINING A MINIMUM VALUE FOR THE X AXIS ON THE PLOIS GENERATED. INTEGER VARIABLE CONTAINING A COUNTER USED TO ENTER FLOWS INTO THE ARRAY (XF) IN THE SUBROUTINE XN (XFLWS) ON REAL VARIABLE CONTAINING THE REAL VALUE OF VARIABLE SUBSCRIPT (N) USED IN THE EXPANSION OF THE FOURTER SEMIES IN THE SUBROUTINE (SER) . XNF - INTEGER VARIABLE CONTAINING THE NUMBER OF FLOWS WITH A MAXIMUM OF 300. INTEGER MATRIX CONTAINING THE LUCATION OF EACH XNLOC VARTABLE IN THE LIST OF STATE VANIABLES (I.E. PL(K)) IN EACH PLOT. INTEGEP VARIABLE CONTAINING THE NUMBER OF STATE XNPL VARIABLES TO BE PLOTTED. XNPL I INTEGER VANIABLE CONTAINING THE NUMBER OF PLOTS TO BE GENERATED WITH & MAXIMUM UP 20. XNPR - INTEGER VARIABLE CONTAINING THE NUMBER OF STATE VARIABLES TO HE PRINTED. XNST INTEGER VARIABLE CONFAINING THE NUMBER OF STATE VARIABLES. XNVRS INTEGER ARRAY CONTAINING THE NUMBER OF VARIABLES PER PLOT WITH A MAXIMUM OF 5. XPL INTEGER ARRAY CONTAINING A LIST OF STATE VARIABLES TO ME PLUTTED. XPR INTEGER ARRAY CONTAINING A LIST OF STATE VARIABLES TO BE PRINIED. XST INTEGER ARRAY CONTAINING A LIST OF STATE VARIABLE INDICES. XIST HEAL VARIABLE CONTAINING A TEMPUHARY VALUE USED TO CALCULATE INDICES OF THE X AAIS ON THE PLOTS GENEPATED. YINC HEAL VANIABLE CONTAINING THE INCHEMENTAL VALUE HETWEEN INDICES OF THE Y AXIS ON THE PLOTS GENERATED. HEAL ARMAY CONTAINING THE INDICES FOR THE Y AXIS ON YLINE THE PLOTS GENERATED. YMAX - PEAL VARIANLE CONTAINING A MAXIMUM VALUE FOR THE Y AXIS ON THE PLOTS GENERATED. YMIN - HEAL VANIABLE CONTAINING A MINIMUM VALUE FOR THE Y AXIS UN THE PLUTS GENERATED. YIST REAL VARIABLE CONTAINING A TEMPURARY VALUE USED TO CALCULATE INDICES OF THE Y ARTS ON THE PLOTS

64

С

C C

c

c

C

C

С

С

c

č

C

C

č

С

C

C C C

č

CCCC

C

CCC

CCCC

Ĉ

c

CCC

С

CCCC

C

cc

C

CCC

C

CCC

C

C

c

CCCC

CCC

CCCC

C

CCC

CCC

č

C

CCC

CCCCCCCC

С

Z

z

GENERATED.

VANIABLE CONTAINING THE SCALING FACTOR FOR

SETTING THE PLOT VALUES IN THE SUBPOUTINE (XEND). REAL ARRAY CONTAINING THE STATE VARIABLES TO BE PLOTTED IN THE SUBPOUTINE (XELGEN).

PFAL

C

C

C

C

C

C

C

C

C

C

C

C

C

Ĉ

C C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

c

C

C

C

C

C 71 - FEAL VANIABLE CONTAINING THE LOCATION OF THE C PENTLING CHARACTER IN THE OUTPUT STPING ACCORDING C С TO THE VALUE OF THE DEPENDENT VANIABLE (77) REFORE C C C C SCAL ING. C 25 HEAL VARIABLE CONTAINING THE VALUE FOR THE INTEGER C С VARIANIE (17) USED IN SCALING THE Y AXIS OF PLOTS С C C C C C C C GENEWATED. - HEAL VANIABLE CONTAINING THE MAAIMUM GRAPHICAL ZMAX SCALING VALUE IN THE SUMPOUTINE (XEND). Ċ - REAL VARIABLE CONTAINING THE MINIMUM GRAPHICAL C ZMIN SCALING VALUE IN THE SUBROUTINE (XRND). - FEAL VAETABLE CONTAINING THE TRUNCATED VALUE FOR THE VAETABLE (2) IN THE SUBROUTINE (XRND). - REAL VAETABLE CONTAINING A TEMPUBARY VALUE STORING CC C ZHNI) C C C C C 22 C THE STATE VARIABLE TO BE PLOTTED. C С C С C č C C PROGRAM FILES-TAPES REFERENCED BY THE VARIABLE (U1) C PROGRAM FILLS - TAPPS REFERENCED BY THE ANTABLE CAPD IS EVUIVALENCED TO THE INPUT FILLS IN THIS CASE THE CAPD HEADERS TAPES REFERENCED BY THE VARIABLE (U2) TS FOUIVALENCED TO THE OUTPUT FILLS IN THIS CASE THE LINE PRINTER, TAPES REFERENCED BY THE VARIABLE (U3) IS EDUIVALENCED TO A DISK FILE THAT IS USED TO STORE PLOT GENERATION DATA. TAPES AND TAPES TO STORE PLOT GENERATION DATA. TAPES AND TAPES TO STORE PLOT GENERATION DATA. TAPES AND C С С С C C C Ċ C С TAPE / REFERENCED BY THE VARIABLE (LU) CAN BE FOUTVALENCED TO DISK FILES USED AS ALTERNATIVE STORAGE SPACE. ALL THESE CCC c VARIANLES ARE INITIALIZED IN THE HIUCK DATA SUMPROGRAM. C С Ċ С С С C FORMAT OF DATA ENTRIES ARE ALSU DESCHIBED IN THE C C USERS GUIDE TO THE RANGES GRASSLAND SIMULATION MODEL. Ċ С C Ĉ С C C C č REMARKS-ALL CODING WITHIN THIS PRUGRAM HAS REEN CONFURMED TO STANDARD ANSI FORTRAN EXCEPT THE FOLLOWING ITEMS. С Ċ CUPIE UNMED TO STANDARD ANSI FURTHAN EXCEPT THE FOLLOWING ITEMS THE MROGRAM CAND WHICH IS A PART OF CDC SCUPE 3.3 OPFRATING SYSTEM. SOME MIXED MODE OPERATIONS, AND THE COC UTILITY FUNCTION AND SUMPOUTINE. THE FUNCTION (RANE) IS REFERENCED IN THE SUBROUTINE (NRM) LINES IR AND 19. AND THE FUNCTION (PPT) LINES 91 AND 94. THE SUBROUTINE (RANSET) IS REFERENCED IN THE SUBROUTINE (START) LINE 66. С C C С Ċ ¢ Ĉ C C C C С RANGE001 INTEGER I.IUAY.INFO.INTR.IP.IYR.II.12.JUDAY.K.LU.MON RANGEOOZ INTEGER NFL+NPL+NPR+UT+UZ+U3+XFR+XNF+XNLOC+XNPL+XNPLT INTEGER XNPR+XNST+XNVR5+XFL+XPR+XST RANGE003 RANGE004 REAL OT+DTFL+DTPL+DTPR+U+TEND+11ME+TIMEFL+TIMEPL+TIMEPR RANGEOOS REAL TM+TSTRT+VAL (300) +X+XF+X10T0+X11T0 RANGEODA C RANGE007 COMMON / TO/ DTFL . DTPL . DTPH . TENU RANGEOOR COMMON /TT/ IDAY , INYN , IYR , JUDAY . K . MON RANGE009 COMMON /TX/ DT+TIME+ISIHT+X(99)+XNE+X10T0+X11T0 PANGE010 COMMON /X5Y5/ XF (300) + XFH (300) + XNLOC (20+5) + XNPL + XNPL + XNPR + XNST + XNRANGE011 1YK5 (20) + XPL (99) + XFR (99) + X5T (99) RANGE012 COMMON /XUNT/ LU.U1.U2.U3 RANGE013 С RANGE 014 С INITIALIZE SIMULATION CONTROL VARIABLES. RANGE015 RANGE 016 C DO 102 1=33+300 RANGE017 0t+1=L RANGEDIA IF (J.GT.99) GU TO 101 RANGE019 XST(J) = 0PANGE 020 101 XFR(I)=n RANGENZI 11EM=1-32 RANGE022 IF (ITEM.GT.99) GO TO 102 RANGE 023 X([TEM)=0.0 RANGE024 102 CONTINUL RANGE025 00 103 1=1,300 RANGE026 XF(1)=0.0 RANGE027 103 CONTINUE RANGE028 CALL XOUT RANGE029 RANGE030 C RANGE 031 С • • • • • • CHECK AND INITIALIZE MORE SIMULATION CONTROL VARIABLES. RANGE032 C RANGE 033 IF (TEND.LF.TSTPT) GO TO 115 RANGE034 IF (DT+LE+n+) GO TO 116 IF (LU+GT+N) REWIND LU RANGE 035 RANGE036 IIME=TSIRT RANGE037 IF (ANPR.GT.O.AND.OTPR.LE.O.) GO TO 117 RANGE038 NPREG PANGE039 TIMEPR=U. RANGE040 IF (ANPR.LF.0) GO TO 104 RANGE041 NPHEI RANGE042 TIMEPR=ISTRT RANGE043 104 NPL=0 PANGE044 TIMEPL=U. RANGE045 IF (XNPL.LF.0) GO TO 105 RANGE 046 NPLET RANGE047 TIMEPLEISTRT PANGE 04A DTPL=(IEND-TSTRT)/99. RANGE049

RANGE050

IF (OTPL.LT.DT) DTPL=DT

```
RANGEOSI
  105 MrL=U
                                                                              RANGE052
      TIMEFI = ...
      IF (DTFL+(F+0+) GO TO 106
                                                                              RANGEOST
      NIL=1
                                                                              RANGE 054
                                                                              PANGE 055
      TIMEFL # LIME
                                                                              HANGERSA
C
č....
          . . OUTPUT SIMULATION CHARACIENISTICS AND CONTROL VARIABLES
                                                                              RANGE057
                                                                              RANGEOSA
C
         AND INITIAL VALUES OF STATE VANIAHLES.
                                                                              RANGE 059
С
  106 WHITE (UZ+TIH) XNST+XNPR+XNPL+XNF
                                                                              RANGEODO
      WHITE (U2+119) TSTHT+TENU+UT+DIPR+DTPL+DTFL
                                                                              RANGEONI
      CALL XPRINT (VAL +XST+XNST)
                                                                              RANGE 062
                                                                              RANGE 063
      WRITE (U2.121)
                                                                              RANGE 064
C
č
  . . . . . . ENTER THE STMULATION LOOP.
                                                                              RANGEO65
С
                                                                              RANGEOGG
                                                                              RANGE 067
  107 CALL CYLLF
С
                                                                              RANGFOOR
                                                                              RANGE 059
С
  . . . . . GENERATE OUTPUT IF REQUESTED.
C
                                                                              RANGE 070
      IF (NPR+LE.D.OR.TIMFPR.GT.IIME) GO TO 10M
CALL XPMINT (VAL+XPR+XNPR)
                                                                              RANGE071
                                                                              RANGE072
       TIMEPH=IINFPR+DIPP
                                                                              PANGE073
  108 IF (NPL+LE_0+OH+TIMEPL+GT+TIME) GO TO 109
CALL XPLOT (VAL+0)
                                                                              RANGE 074
                                                                              RANGE075
      TIMEPL = IIMFPL + DTPL
                                                                              PANGE076
C
                                                                              RANGE 077
  . . . . . COMPUTE THE FLOWS.
c
c
                                                                              RANGE 78
                                                                              PANGE 079
                                                                              RANGEORO
  109 CALL XFLWS
         (NFL+LE+0+OR+TIMEFL+GT+TIME) GO TO 110
      IF
                                                                              RANGE OR1
      TM=TIME+DT
                                                                              RANGE 082
                                                                              PANGEDAS
C
  . . . . . UPDATE THE STATE VARIABLES AND PRINT FLOWS IF REQUESTED.
                                                                              PANGE084
C
č
                                                                              PANGEORS
  110 ×11T0=×(11)
                                                                              RANGEORA
      ×1010=×(10)
                                                                              RANGEOB7
      DO 111 1=1.XNF
                                                                              RANGEORA
          INFU=XFR(I)
                                                                              RANGEONS
          CALL XUMPAK (INFO.IP.II.12)
                                                                              RANGE090
          U=XF(I) OT
                                                                              RANGEN91
          x(11) = x(11) = 0
                                                                              RANGE 092
          X(15)=X(15)+0
                                                                              RANGE 193
          IF (NFL.LE.0.0R.TIMEFL.GT.TIME) GO 10 111
IF (IP.LE.0) GO TO 111
WRITE (12.120) I1.12.XF(I)
                                                                              RANGE 094
                                                                              RANGE 095
                                                                              RANGE 096
  111 CONTINUE.
                                                                              RANGE097
      IF (NFL+LF.0.0R.TIMEFL.GT.TIME) GO TO 112
                                                                              RANGE 098
      TIMEFL=1IMEFL+DTFL
                                                                              RANGED99
  112 TIME=TIME+DT
                                                                              RANGE100
      IF (IN1(TIME).GT.INT(TEND)) GO TU 113
                                                                              RANGEIDI
      GO TO 107
                                                                              RANGE102
C
                                                                              RANGE103
č
            . SIMULATION IS COMPLETE, PERFORM END PROCESSING AND PLOT
                                                                              RANGE104
  . . . .
С
         GENERATION IF PROVESTED.
                                                                              RANGE105
                                                                              RANGE 106
С
  113 IF (NPL+LE.0) 60 TO 114
CALL XPLOT (VAL+1)
                                                                              RANGE107
                                                                              RANGE 108
      CALL XPLGEN
                                                                              RANGE109
  114 STUP
                                                                              RANGE110
С
                                                                              RANGELLI
С
  • • • • • • GENERATE DIAGNOSTICS FUR ILLEGAL CONDITIONS.
                                                                              RANGE112
Ĉ
                                                                              RANGEI13
  115 WRITE (U2+122)
                                                                              RANGE114
      WRITE (U2+123) TSTRT+TEND
                                                                              RANGE115
      STOP
                                                                              RANGE116
  116 WRITE (U2+122)
                                                                              RANGE117
      WRITE (U2+124) DT
                                                                              RANGE118
      STOP
                                                                              RANGE119
  117 WRITE (U2.122)
                                                                              RANGE120
      WRITE (U2,125) DTPR
                                                                              RANGE121
      STOP
                                                                              RANGE122
C
                                                                              RANGE 123
C
  . . . . . FOPMATS USED IN THIS PROGRAM.
                                                                              RANGE 124
C
                                                                              RANGE 125
  1......+12.1/20X+20H0T......RANGE131
2.....+F12.1/20X+20HD1PL.....PANGE131
2.....+F12.1/20X+20HD1PL.....PANGE132
               ...FI2.1//)
                                                                              RANGE 133
  120 FORMAT (10X+1H(+12+1H++12+4H) = +F12.5)
                                                                              RANGE 134
  121 FORMAT (7M1HANGES,15%,1ANSIMULATION RESULTS//)
122 FORMAT (42H0000001LEGAL CONDITION - PARAMETER VALUES)
                                                                              RANGE135
                                                                              RANGE136
  123 FORMAT (13H0
                        TSTHT (+F12+1+12H) .GE. TENU(+F12+1+1H))
                                                                              PANGE 137
  124 FORMAT (9H0 DI (+F 12+1+9H) +LE+ 0+)
125 FORMAT (1H0+5X+3HHPRINT HEQUESIS ENCOUNTERED WHILE DTPR(+F12+1+9H)PANGE139
     1 .LE. 0.)
                                                                              RANGE 140
С
                                                                              RANGE141
      END
                                                                              RANGE142
```

	BLUCK DATA	ALKOT001
С		PLKPT002
с.	THIS SUPPHOGNAM (HEOCK DATA) ENTERS INITIAL VALUES INTO	ALKDI003
C C	VARIABLES IN THE LABELED COMMON PLOCKS PRIOR TO PROGRAM	RLKDT004
C	EXECUTION.	PLKDT005
С	TATE OF TO AN INTER TOM FILM AND TOWARD REPAYER.	BLKPT006 BLKPT007
	INTEGEN IRLNK & IPASH & FEYER & IN & INFD & TOUAL & JCHAR & KEY & LU INTEGEN MK T & MODP & MODT & NCLAS & MOED & CD & OD & VD & XEM & XNE	BLKDTOOR
	INIEGEN ANLUCONNELONALIONALIANENANIA	RLKPT009
	REAL ANUS + ANTOT + AREA + CH + CHAR + CHMAX + DITI + DT + FLUNS+ (ATN	BLKCT010
	HEAL GN+GNTN+LVWT+LWT+MSZ+UMIN+PFL+PH+PH+PRC+PT+PW	PLKDT011
	HEAL HUSSD.STKND.SOPL.SUPN.SUPPR.SUPPT.T.T.TIME TOTAL TP	ALKDT012
	HEAL THOISTHIGVOWMAXOWMMAXOXONE X10T00X11T0	PLKDT013
С		BLKCT014
	COMMON /CAT/ ANOS(5) +ANTOT + APEA + DIIT + FCONS+GAIN(5) + GNIN(5) + IN(5) +	IPLKDT015
	1NED (5) *EVWT (5) *EWT (5) *MKT (5) *MSZ (5) *NCEAS*NOED (5) *OMTN (5) *STKNO (5)	
	2+SUPL (5) +SUPN (5) +SUPPR (5) +SUPPT (5)	BLKDT017
	COMMON /HESH/ CMAX+CHMAX+WMAX+WMAX	ALKOTOIA
	COMMON /TAP/ MODP+MODT+PFI+PT+T+TP(6)+TR	ALKOTO19
	COMMON /TX/ DT+T1MF+TSTHT+X(99)+XHF+X1010+X11)0 COMMON /UX/ CH+GN+PH+PK+PKC+P++RG+SD+TUTAL+V	BLKDT020
	COMMON /XFIG/ KFY(5) +18LNK +10ASH+1FYFF+100AL+JCHAR(5)	BLKDT022
	COMMON /XSYS/ XF (300) +XFR(300) +XNL0C(20+5) +XNPL+XNPL+XNPR+XNST+X	
	1VRS(20) + XPL (99) + XPH (99) + X51 (99)	ALKOT024
	COMMON /XUNI/ LU+U1+U2+U3	BLKDT025
С		RLKDT026
с.	• • • • INITIALIZE SIMULATION CONTROL VARIABLES.	BLKDT027
С		ALKD1028
	DATA XNF/32/	BLKDT029
	DATA xFH(1)/3121/+xFH(2)/2132/+xFP(3)/3310/+AFR(4)/3311/	RLKDT030
	DATA XFM(5)/1012/+XFM(6)/1112/+XFM(7)/1234/+AFR(A)/1001/	PLKDT031
	DATA XEM (9)/1101/+XEM (10)/1201/+XEM (11)/3501/+XEM (12)/0134/ DATA XEM (13)/1002/+XEM (14)/1102/+XEM (15)/1202/+XEM (16)/3502/	REKDT032
	DATA XFH(17)/0234/+XFP(1A)/1003/+XFP(19)/1103/+XFP(20)/1203/	ALKOT034
	DATA XFK(21)/3503/.XFR(22)/0334/.XFR(23)/1004/.XFH(24)/1104/	PLKDT035
	DATA XFH(25)/1204/+XFH(26)/3504/+XFH(2/)/0434/+XFH(24)/1005/	HLKDT036
	DATA XFR(29)/1105/*XFR(30)/1205/*XFR(31)/3505/*XFR(32)/0534/	PLKDT037
	DATA XNST/AC/	PLKDT038
	DATA_X5T(1)/01/+XST(2)/02/+XST(3)/03/+XST(4)/04/+XST(5)/05/	8LK07-039
	DATA X51(6)/10/+X51(7)/11/+X51(8)/12/+X51(9)/21/+X51(10)/31/	PLKDT040
	DATA x51(11)/32/+x51(12)/33/+x51(13)/34/+x51(14)/35/+x51(15)/41/	PLKNT041
	DATA x51(16)/42/+x5T(17)/43/+x5T(18)/44/+x5T(19)/45/+x5T(20)/46/	RLKDT042
	DATA X51(21)/47/+X5T(22)/48/+X5T(23)/45/+X5T(24)/50/+X5T(25)/51/ DATA X51(26)/52/+X5T(27)/53/+X5T(28)/54/+X5T(29)/55/+X5T(30)/56/	RLKDT043
	DATA XSI (26)/52/+XST (32)/58/+XST (33)/54/+XST (34)/60/+XST (35)/61/	HLKDT045
	DA1A X51 (36) / 62/+X5T (37) / 63/+X5T (39) / 64/+X5T (39) / 65/+X5T (40) / 66/	PLADT046
	DATA \$51(41)/67/0x51(42)/68/0x51(43)/69/0x51(44)/70/0x51(45)/71/	BLKDT047
	DATA XSI (46)/72/+XST (47)/73/+XST (48)/74/+XST (49)/75/+XST (50)/76/	RLKDT048
	DATA X51 (51)/77/++ST (52)/78/+X51 (53)/79/+X51 (54)/80/++ST (55)/81/	PLKDT049
	DATA XSI(56)/H2/+XST(57)/H3/+XST(54)/H4/+XST(59)/H5/+XST(60)/H7/	BLKDT050
	DATA XST (61)/68/0XST (62)/49/	PLKPT051
	DATA XNPP/0/,XNPL/0/,XNPL1/0/,X1110/0./,X1010/0./	PLKDT052
	DATA KEY (1) /4HEND./.KEY (2) /4HPH1N/.KEY (3) /4HFLOW/.KEY (4) /4HPLOT/	PLKDT053
		BLKDT054
	DATA_JCHAP(1)/1H1/+JCHAR(2)/1H2/+JCHAR(3)/1H3/+JCHAP(4)/1H4/ DATA_JCHAR(5)/1H5/	PLKOTO55 PLKDTO56
	DATA IBLNK/IN / IDASH/IH-/ IFYEE/IHI/ IQUAL/IH=/	BLKDT057
С	menter and the second	BLKDT058
Ċ	• • • • INITIALIZE NON-PEAD VAHIARLES.	RLKDT059
C		PLKDT060
	DATA PRC/25.0/.+G/04/.PFL/0./.GN/1.0/.FCON5/0./.PH/.19/	BLKDT061
	DATA PH/6.25/.CH/0./.V/0./.PW/10./.DIIT/0./.1/0./.TOTAL/0./	RLKDT062
	DATA SU/.06/	BLKDT063
	DATA WMAX/.39/.CMAX/.41/	BLKDT064
	DATA WEMAX/.13/.CEMAX/.18/	RLKDT065
С	DATA U1/5/+U2/6/+U3/3/+LU/4/	BLKDT066
•	END	BLKDT068

		SUPPOUTINE CYCLE	CYCI E001
C	2		CYCLE002
0		• • • • THIS SUFRIUTINE (CYCLE) IS CALLED BY THE MAIN PROGRAM	CYCLE003
C		(HANGES) TO RECALCULATE THE VALUES OF TMPONTANT VARIABLES	CYCLE004
0	2	REFURE COMPUTATION OF FLOWS AT EACH TIME SLEP DUPING	CYCLE005
C		SIMULATION.	CYCI E006
C			CYCL E007
		INTEGER IDAY . IN . INFO . INY . IYR . J. JUDAY . K. MKT . MUDP	CYCLEOOR
		INTEGEN MONTOMONONCLASONUFUOXNE	CYCLE009
		REAL ANUS ANTOT ANFA HASE CASH CHADIIT AT A FOUNS GAIN	CYCLE010
		HEAL GNOGNINOLVWIOLWIOMSZOUNINOPHI. PHOPROPRCOPHCOV	CYCLE011
		HEAL PHICE +PHMN +PT +PW +K(+SU+STKNU+SUP) +SUPN +SUPP	CYCLE012
		HEAL SUPPH SUPHT SUPHT + TATTHE + 10TAL + TP + TH + TS INT + V	CYCLE013
		PEAL VALUE . X . X 1070 . X 1170 . 72	CYCLE014
C			CYCLE015
	-	COMMUNI /CAT/ ANOS(5) + ANTOT + AFFA + DIIT + FCONS+GAIN(5) + GNIN(5) + IN(5) +	
		INFU (5) +LVWT (5) +LWT (5) +MKT (5) +M52 (5) +NCLAS+NOFU (5) +OMIN (5) +STKND (5	
		2 • SUPL (5) • SUPN (5) • SUPPR (5) • SUPRT (5)	CYCLEOIA
		COMMON /FCON/ HASE (5) .CASH (5) .PHC(V(144) .PR1CE(12) .PHMN(12) .SUPP (
		1) • SUPWT (5) • VALUE (5) • 77 (12)	CYCLE020
		COMMON /TAP/ MODP.MUDT.PFL.PI.T. (P(K). 18	CYCLE021
		COMMON /TT/ IDAY . INYH . IYH . JUDAY .K . MON	CYCLE022
		COMMUN /TX/ UT.TIMF.TSTHT.X(99).XNF.X10T0.X1110	CYCLE023
		COMMON /UX/ CH+GN+PH+PHC+PHC+PH+KG+SD+TUTAL+V	CYCLE024
C	:		CYCLF025
		IF (NCLAS.LE.O) GO TO LOL	CYCLE026
0	2		CYCLE027
C		· · · · CHECK FOR FOULLITY OF LIME AND ISTRT.	CYCLE028
Ċ			CYCLE029
		JF (TIME LF. TSTHI) GO TO 101	CYCLE030
		CALL CYCL2	CYCLE031
	101	CALL CYCL1	CYCLE032
C			CYCLE033
C		TPANSFER VALUES OF SYSTEMS VARIABLES TO STATE VARIABLES	CYCLE034
C		FOR PLOT AND PRINT OUTPUTS.	CYCLE035
C	2		CYCLE036
		X (7I)=TH	CYCLE037
		X(72)=T .	CYCLE03R
		X (73)=GN	CYCLE039
		X (74) = Pn	CYCLE040
		X (75) =PK	CYCLE041
		入(76)=しい	CYCLE042
		X (77) = ∀	CYCLE043
		λ(78)≡P«	CYCLE044
		X (79)=FCONS	CYCLE045
		X(80)=PFL	CYCLE046
		X(87)=PRICF(MON)	CYCLE047
		X (88) = [UTAL	CYCLE048
		IF (NCLAS.LE.D) RETURN	CYCLE049
		00 102 J=1,1CLA5	CYCLE050
		X(J+40) = VALII(J)	CYCLE051
		X(J+45)=HASE(J)	CYCLE052
		X (J+50) = SUPP (J)	CYCLE053
		X (J+55)=CASH(J)	CYCL E054
		X(J00)=SUPWT(J)	CYCLE055
		X(J + 65) = LWT(J)	CYCLE056
		X (J+60) = ANOS (J)	CYCLE057
	102	CONTINUE	CYCLE058
		RETURN	CYCLE059
C	2		CYCLE060
		END	CYCLE061

• • • • • TAKE ANIMALS OFF PASTURE. CYCL 12 X(1)=0.0 CYCL 13 CONTINUE CYCL CYCL 14 1F (X(1u).(T+0.) X(10)=0. CYCL CYCL 15 (X(11).(E.0.0) X(1))=0.0 CYCL CYCL 16 (X(11).(E.0.0) X(1))=0.0 CYCL CYCL 17 (X(11).(E.0.0) X(1))=0.0 CYCL CYCL 18 (X(21).(E.0.0) X(21)=0.0 CYCL CYCL NETURN CYCL CYCL 540 CYCL CYCL		69	
, This summarity: (T(1)) (C CALC) (C CALC) </td <td></td> <td></td> <td></td>			
(1111)10.10.10.10.10.10.10.10.10.10.10.10.10.1			
Differentiation and the State (See a first First And Andrew Stephenia) CYTE Differentiation and the state (See a first And Andrew Stephenia) CYTE Differentiation and the state (See a first Andrew Stephenia) CYTE Differentiation and the state (See a first Andrew Stephenia) CYTE Differentiation and the state (See a first Andrew Stephenia) CYTE Differentiation and the state (See a first Andrew Stephenia) CYTE Common Andrew Stephenia CYTE Differentiation and the state (See a first Andrew Stephenia) CYTE Common Andrew Stephenia	•		
INTEGER 1.110000100000000000000000000000000000			
Introductions.A			
M AL ANOSA AND DE ANAS - LESS (11) TETTE VAP / LEARS (CONS CYCL P AL CALVACIANE VALUE (11 SAVE (110) CALVE (110) CONSTANT) CYCL P AL CALVACIANE VALUE (11 SAVE (110) CALVE (110) CALVE (111) CYCL M AL SUPERIAL (110)			
PF AL ALINGUNINUMILWIGT.MY, DETARATION AND ALTARATION AND ALTARA			
MALA LAVELIA (MULTIAL (MULTIAL (MULTIAL) (MULTIAL) (MULTIAL) CYCC WHALA LAVELID (MULTIAL) CYCC CUMMUN / CAT/ ANDY(S) AND (S) AND (S) AND (S) (G) (S) (G) (S) (S) (S) (S) (S) (S) (S) (S) (S) (S		PEAL GAIN+GNIN+LVWI+LWI+MSZ+OMIN+FEL+PRCOV+PRICE+PRMN	CYCI
HEAL AAADTOUALTIDA/2 CYCC CUMMUD /CATZ ADDUCGIANTIDA ADDUCAS, GALD(S), GALD(S), IGN(S), IAN(S), ICN(S), ICN(S)			
CUMMUN /CAT/ ANUN(5) +ANINTANPA HITT.FCUNS(GAIN(5), GNIN(5), IN(5), IN			
CUMMUN /CAT/ AND/CS AND/CS AND/CS AND/CS AND CS AND (CS AND (C			
2-SUPPL (5) SUPPL(5) SUPPL(5) SUPPL(5) CTC CDMMUN / FORM / ASS (5) CANIGS JPUELDY (144) JPD(2) (12) SUPPL(5) CYCL CYCL CDMMUN / TAP / LAND / MULTAPL (14) TIT (14) (14) JPD (2) (12) SUPPL(5) SUPPL(5) CYCL CYCL CDMMUN / TAP / LAND / MULTAPL (14) TIT (14) (14) JPD (2) (12) SUPPL(5) SUPPL(5) CYCL CYCL CDMMUN / TAP / LAND / MULTAPL (14) TIT (14) (14) JPD (2) (12) SUPPL(5) SUPPL(5) CYCL CYCL CDMMUN / TAP / LAND / MULTAPL (14) TIT (14) (14) JPD (2) (12) SUPPL(5) SU			
CUMMUM AFCONZ NEASI (5) CASINGS AMACON (124) AMP[Cc (12), PMMM[12] +SUDP[Cc (12), PMM[12] +SUDP[CC (12			-
11.5.10001153.vx1101(5).7/112) CYCL COMMUN /TAP/ MUP-MOTI-FLIPTI-LPT.11.20(1).TM CYCL COMMUN /TAP/ MUP-MOTI-FLIPTI-LPT.11.20(0).TM CYCL TOTAL-0. CYCL TOTAL-0. CYCL TOTAL-0. CYCL TOTAL-0. CYCL TOTAT-10.5. CYCL TOTAT-10.5. CYCL TOTAT-11.4. CYCL MONEIDAT/30.1667-1.0 CYCL F(10AT.6			
CUMMON / TAP/ BUDD-SUDD-STLIPT.1.1P(B).IN CYCL CUMMON / TA/ UTT / INAY.INAY.HATA.S.MOM CYCL CUMMON / TA/ UTT./ INAY.INAY.HATA.S.MOM CYCL CUMMON / TA/ UTT./ INAY.INAY.HATA.S.MOM CYCL CUMMON / TA/ UTT./ INAY.INAY.HATA.S.MOM CYCL INATELED. CYCL INAYELIME-TINE365. CYCL INAYELIME-TINES6. CYCL INAYELIME-TINE365. CYCL INAYELIME-TINE36. CYCL INAYELIME-TINE36. CYCL INAYELIME-TINE36. CYCL INAYELIME. CYCL INAYELIME. CYCL GO TO 104 CYCL IF (IND1-2) 107.105. CYCL IF (IND34400TANAATHADTANDAT			
CUMMUN /TI/ 104:11104:11104:0104:A.MON CUMMUN /TI/ 104:11104:15114:A.195:A.AF6.4:U10.4:1110 CYCL CYCL T01AL=0.			
CUMMUN /IX/ DITTINF.ISTHI.A.(96).ANF.XIUI0.XIII0 CYCL CALCULATE YFAH AND DAY OF YFAP UF SIMULATION. CYCL TOTAL=0. CYCL TOTAL=0. CYCL THATIYE.JTHF.ITH.A.104.DAY OF YFAP UF SIMULATION. CYCL TOTAL=0. CYCL THATIYE.JTHF.ITH.A.104.DAY OF THE YEAR. CYCL INATATIME.TIME.THM.A.104.DAY OF THE YEAR. CYCL MODELDAT.ZO.LAFF.MONTH OF THE YEAR. CYCL MODELDAT.ZO.LAFF.MONTH.AFTH.AFTH.TH.THEY OF TAND THEMPELDAT. CYCL CALL RELEMD (THRE.STMT.LUNAT.INYM.TP(1).PT) CYCL CALL RELEMD (THRE.STMT.LUNAT.INYM.TP(1).PT) CYCL CALL RELEMD TATA READ.FROMTHATINE ATTRE. CYCL CALL RELEMD TATA READ.FROMTHATINE ATTRE. CYCL CALL RELEMD TIME.THET.INTT.NANT.THEF.INTT.INTT.THET.INTT.INTT.THET.TOTATTRE.CYCL CYCL </td <td></td> <td></td> <td></td>			
TOTAL=0. CYCC TYM=T1P(Z)SN. CYCC TIMATETIFF-TIME365. CYCC TIMATETIFF-TIME365. CYCC TWHETIMETTAL.NESS. CYCC			
TOTALED. CYCL TYMETIME/JANA. CYCL TYMETIME/JANA. CYCL TYMETIME/JANA. CYCL TYMETIME/JANA. CYCL TYMETIME/JANA. CYCL TYMETIME/JANA. CYCL MONEIDAT/JA.1667+1.0 CYCL TF (MON.GT.17) MUNET2 CYCL MEDTO.5. CYCL TI (DAT.GT.M) GO TO 101 CYCL TF (HOD.5.1) CYCL TF (HOD.7.1.0) CYCL TF (HOD.7.1.0) CYCL CALL RELEWD (TIME.FISTHT.1DAY.NYNH.TP(1)*PT) CYCL GO 10 104 CYCL CALL RELEWD (TIME.FISTHT.1DAY.NYNH.TP(1)*PT) CYCL GO 10 104 STOCHASTICALEY IN THE FINCTIONS (PPT AND TEMP). CYCL STACA. CYCL GO 10 104 STOCHASTICALEY IN THE FINCTIONS (PPT AND TEMP.CYCL STACA. (TP). CYCL STACA. (TP). CYCL TE0.0 CYCL D0 105 1=1.5 CYCL JF-10 CYCL JF (JJ)=TP((J). CYCL TF (JJ)=TP((J). CYCL TF (JJ)=TP((J).	•	CALCULATE YEAR AND DAY OF YEAP OF SIMULATION.	CYCI
11W#=1[ME/3%3. CYCL 11W#=1[M#+1] CYCL 11W#=1[M#+1] CYCL WON=1DAT_JOLAFT+MATHOF IME YEAR. CYCL WON=1DAT_JOLAFT+1.01 CYCL WON=1DAT_JOLAFT+1.01 CYCL If (MON_GT,12) MUN=12 CYCL WINT GTO TO 101 PFL=00. CYCL NIF (MODI-2) 102.103.103 CYCL CYCL CYCL GO 10 104 CYCL WEATHER DATA CALCULATED HY FITHEW & SINF FUNCTION OF CYCL WEATHER DATA CALCULATED HY FITHEW A SINF FUNCTION OF CYCL WEATHER DATA CALCULATED HY FITHEW A SINF FUNCTION OF CYCL WEATHER DATA CALCULATED HY FITHEW A SINF FUNCTION OF CYCL COMPORT SICALLY IN THE FINCTIONS (PPT AND TEMP1. CYCL COMPORT SDAY TEMPEHATURE AVERAGE (T) FROM TEMPERATURE CYCL			
IDATEINF-ITWEBSS. ITWEINF: ITWEINF: WONEIDAT/30.1667+1:0 IF (HON.GT.17) MUNEI2 MONEIDAT/30.1667+1:0 IF (HON.GT.17) MUNEI2 MEDT0.5 IF (IDAT.GT.H) GO TO 101 FFLEG. KEG OT CL KEG OT CL KEG NOT CL KEG OT CL KEG NOT CL KEG NOT CL KEG OT CL KEG NOT CL KEG KEG NOT CL KEG NOT CL KEG NOT CL KEG NOT CL KEG NOT CL KEG KEG KEG NOT CL KEG KEG KEG KEG KEG KEG KEG KEG			
1YH=1YH+1 CYCL • • • • • CALCULATE MONTH OF THE YEAR. CYCL MON=1DAT/30.1667+1.0 CYCL IF (MON.GT.12) MUN=12 CYCL MOD-10.5 CYCL K=0 CYCL K=1 F(MOD-2) 102.103.103 CYCL CYCL * * * * WEATHER DATA READ FROM A TAPE. CYCL CALL RELCHD (TIME.ISTENT.INDAY.INYH.IP(1).PT) CYCL GENEMATION STOCHASTICALLY IN THE FUNCTIONS (PPT AND TEMP). CYCL GENEMATION STOCHASTICALLY IN THE FUNCTIONS (PPT AND TEMP). CYCL TP(1)=TEMP(10AY.MODT.INKA.TIME.ISTET) CYCL TP(1)=TEMP(10AY.MODT.INKA.TIME.ISTET) CYCL TF0.0 CYCL CYCL D0 105 1=1.5 CYCL J=7-1 CYCL J=7-1 CYCL MODAT SOF ANIMALS ON PASTUPF (XTI). ACTUME MUMER OF CYCL ANIMALS UN PASTUPF (AUNT). ANIMALS PFW ACHE (ANIMARE OF CYCL ANIMALS OF ANIMALS ON P			
• • • • CALCULATE MONTH OF THE YEAR. CYCL MONEIDAT/30.166741.0 CYCL IF (MON.GT.12) MUNE12 CYCL MODEIDAT/30.166741.0 CYCL IF (MON.GT.12) MUNE12 CYCL MODEIDAT/30.166741.0 CYCL MEDIOA. CYCL MEDIOA. CYCL MEDIOA. CYCL K=0 CYCL MEDIOA. CYCL K=0 CYCL K=0 CYCL C CYCL K=0 CYCL C CYCL C CYCL C CYCL C CYCL C CYCL C CYCL GO 10 104 CYCL C GENEMATION STOCHASTICALLY FNINH, TRY (1) +FT CYCL CYCL GENEMATION STOCHASTICALLY FNINH, TRY (1) +FT CYCL CYCL SEGNEMATION STOCHASTICALLY FNINH, TRY (1) +FT CYCL CYCL SEGNEMATION STOCHASTICALLY FNINH, TRY (1) +FT <			CYCI
MON=1DAT/30.1667+1.0 CYCL IF (MON,6T,12) MUN=12 CYCL H=D160.5 CYCL IF (1DAT.6T,*) GO TO 101 CYCL PFL=0. CYCL K=0 CYCL NIF (MOD)-2) 102+103+103 CYCL CALL RELEAD (TIME.1STHT.1DAY.1NYH.1P(1)+PT) CYCL GO 10 104 CYCL ************************************		CALOR ATE MONTH OF THE MENO	
MONEIDAY230.166741.0 CYCL IF (MON.GT.12) MUNE12 CYCL MEDT40.5 CYCL CALL MELCH0 (TIME+ISTHT+IDAY-INYH-IP(1)*PT) CYCL CGUAL MELCH0 (TIME+ISTHT-ISTHT) CYCL GENERATION STOCHASTICALLY IN THE FINCTIONS (PPT AND TEMPE) CYCL JEPHT (IDAY-MODT+IME_ISTHT.INPA) CYCL SIACK (TH). CYCL		CALCULATE MONTH OF THE YEAR.	
IF (MON.GT.17) MUNE12 CYCL HEDT0.5 CYCL IF (1DAT.GT.M) GD TD 101 CYCL PFLE0. CYCL NI F (MODI-2) 102+103+103 CYCL CYCL CYCL NI F (MODI-2) 102+103+103 CYCL CYCL STACA (TP). CYCL CYCL </td <td></td> <td>MON=10A1/30.1667+1.0</td> <td></td>		MON=10A1/30.1667+1.0	
IF (IDAY.GT.M) GO TO 101 CYCL FE_0 CYCL NI IF (MODI-2) 102+103+103 CYCL OL IF (MODI-2) 102+103+104+1104+1104+1104+1104+104+104+104 CYCL OL OL UA CYCL OL OL UA CYCL OL OL UA CYCL • • • • WEATHER DATA CALCULATED HY FITHEN A SINF FUNCTION OP CYCL OL OL UA CYCL • • • • • WEATHER DATA CALCULATED HY FINCTIONS (PPT AND TEMPE). CYCL OL OL UA CYCL CYCL • • • • • WEATHER DATA CALCULATED HY FINCTIONS (PPT AND TEMPE). CYCL OL OL UA CYCL CYCL • • • • • COMPUTE 5 DAY TEMPEHATURE AVERAGE (T) FROM TEMPERATURE CYCL • • • • COMPUTE 5 DAY TEMPEHATURE AVERAGE (T) FROM TEMPERATURE CYCL • • • • • ESTABLISH A LIVESTOCK SCHEDULE WITH THE VARIABLES OF CYCL CYCL • • • • • • ESTABLISH A LIVESTOCK SCHEDULE WITH THE VARIABLES OF CYCL CYCL • • • • • • • ESTABLISH A LIVESTOCK SCHEDULE WITH THE VARIABLES OF CYCL CYCL		1F (MON.GT.17) MUN=12	CYCL
PFLE0. CYCL K=0 CYCL N: IF (MODI-2) 102+103+103 CYCC			
K#0 CYCL n1 IF (MODI-2) 102+103+103 CYCL 01 IF (MODI-2) 102+103+103 CYCL 02 ALL RELMD (TIME+TSTMT+1DAT+1NYH+TP(1)+PT) CYCL 02 CALL RELMD (TIME+TSTMT+1DAT+1NYH+TP(1)+PT) CYCL 03 PT=PPT(IDAT+MODP-TIME+TSTMT+NNA) CYCL 03 PT=PPT(IDAT+MODP+TIME+TSTMT+NNA) CYCL 04 THETP(1) CYCL 154 THETP(1) CYCL 154 THETP(1) CYCL 155 T=T+TP(1) CYCL 166 T=T*2 CYCL 176 (10AY-S) 107+107+106 CYCL 177 (10AT+MAGS AN PASTUPF (ANDS(1))+ AND ON MANKET DAY+ CASH CYCL 176 (NCLAS_LE.0) GO TO 114 CYCL 177 (ANTOT=0. CYCL			
<pre>CYCL CYCL CYCL CYCL CYCL CYCL CYCL CYCL</pre>		K=0	
	10		
D2 CALL RELMD (TIME+ISTMT+1DAY+INYH+TP(1)+PT) CYCL G0 10 104 CYCL CYCL CYCL CYCL G0 10 104 CYCL CYCL CYCL G0 10 104 CYCL CYCL CYCL CYCL G1 104 CYCL CYCL CYCL CYCL CYCL CYCL CYCL CYCL			
D2 CALL RECEPD (TIME TSTNT.1DAY.INYH.TP(1):PT) G0 TO 104 CYCL G0 TO 104 CYCL G0 TO 104 CYCL G0 TO 104 CYCL G0 TO 104 CYCL G0 TO 104 CYCL G0 TO TOTAL TOTAL TOTAL TOTAL THE FUNCTION OP CYCL G0 TOTAL TOTAL TOTAL TOTAL TOTAL TOTAL TOTAL TO G0 TOTAL TOTAL TOTAL TOTAL TOTAL TOTAL TO TP(1)=TEPT(IDAY.MODT.TWE.TSTNT.NNA) CYCL TP(1)=TEPT(IDAY.MODT.TWE.TSTNT.NNA) CYCL TP(1)=TEPT(IDAY.MODT.TWE.TSTNT.NNA) CYCL TP(1)=TEPT(IDAY.MODT.TWE.TSTNT.NTE.TSTRT) CYCL TOTAL TOTAL TOTAL TO TTOTAL TOTAL TOTAL TO TTOTAL TOTAL TOTAL TO CYCL TOTAL MASS OF ANIMALS ON PASTUPF (T) FROM TEMPERATURE CYCL TTOTAL MASS OF ANIMALS ON PASTUPF (T).CYCL AVERAGE ANIMALS ON PASTUPF (T). CYCL AVERAGE ANIMALS ON PASTUPF (T). ACTUAL MUMMER OF CYCL AVERAGE ANIMALS ON PASTUPF (T). ACTUAL MUMMER OF CYCL AVERAGE ANIMALS ON PASTUPF (T). ACTUAL MUMMER OF CYCL AVERAGE ANIMALS ON PASTUPF (T). AND ON MAKKET DAY. CASH AVERAGE ANIMALS ON PASTUPF (T). AND ON MAKKET DAY. CASH CYCL AVERAGE ANIMALS OF ANY CLASS ARE ON PASTUF. CYCL IF (IDAY-MKT(I)) 108.111.112 CYCL IF (IDAY		, a a a esentence trate tisedΩ FISHER R. ERES.g.	
<pre>cvcl cvcl generation Stochastically in the functions (PPT and Temp). cvcl generation Stochastically in the functions (PPT and Temp). cvcl generation Stochastically in the functions (PPT and Temp). cvcl trep(l)=lemp(loay+modt+nume+tstr) cvcl trep(l)=lemp(loay+modt+nume+tstr) cvcl stack (tP). cvcl stack (tP). cvcl stack (tP). cvcl generation trep() feld=trep() cvcl stack (tP). cvcl stack (tP). cvcl stack (tP). cvcl tre0. cvcl generation cvcl stack (tP). cvcl tre0. cvcl generation cvcl stack (tP). cvcl stack (tP). cvcl tre0. cvcl stack (tP). cvcl stack (tP). cvcl tre0. cvcl stack (tP). cvcl stack (tP). cvcl tre0. cvcl tre0. cvcl tre1. cvcl stack (tP). cvcl tre0. cvcc tre0. cvcl tre0. cvcvcl tre0. cvcl tre0. cvcl tr</pre>	0		CYCL
 * * * * * * * * * * * * * * * * * * *			
GENEMATION STOCHASTICALLY IN THE FUNCTIONS (PPT AND TEMP). CYCL 013 PTEPPT(I)AX+MODP+TIME+TSTPTIANA) CYCL 023 PTEPPT(I)AX+MODT+NKA+TIME+TSTRT) CYCL 04 THETP(I) CYCL 04 THETP(I) CYCL 05 TE0+0 COMPUTE 5 DAY TEMPEHATURE AVERAGE (T) FROM TEMPERATURE CYCL 05 TE0+0 CYCL CYCL 05 TE1+0 CYCL CYCL 05 TE1+0(J-1) CYCL CYCL 05 TE1+0(J-1) CYCL CYCL 06 TE1+0(J-1) CYCL CYCL 06 TE1+0(J-1) CYCL CYCL 06 TE1+0(J-1) CYCL CYCL 06 TE1+0(J-1) CYCL CYCL 07 TATATA LIVESTUCK SCHFDULF WITH THE VAPTAPLEFS OF CYCL 07 TUTAL MASS OF ANTMALS ON PASTUPF (X(T))+ ACTUAL MUMHER OF CYCL 06 TATATA LIVESTUCK SCHFDULF WITH THE VAPTAPLEFS OF CYCL 07 ANTOT=0. CYCL CYCL CYCL 01 TEALESTUP AND ON MARKET D			
03 PT=PPT(10AY,MODP,TIME,ISIPT,NRA) CYCL 17 P(1)=LEMP(IDAY,MODT,NRA,TIMF,ISTRT) CYCL 04 TH=TP(1) CYCL 05 FIFT CYCL 06 IN=TP(1) CYCL 07 SIACA (TP). CYCL 06 IN=TP CYCL 07 SIACA (TP). CYCL 07 T=0.0 CYCL 00 105 L=1.5 CYCL 05 T=1.1 CYCL 16 T=T+1P(J) CYCL 05 T=T+1P(J) CYCL 17 IDAT-5) 107+107+106 06 T=T*2 CYCL 10 TAL MASS OF ANIMALS ON PASTUPF (x(1)). ACTUAL MUMHER OF CYCL 104 MANAGE ANIMAL WFIGHT (LVWT(1)). ACTUAL MUMHER DAY, CASH CYCL AMPAGE ANIMAL WFIGHT (LVWT(1)). ANU ON MARKET DAY, CASH CYCL AMOUNT PECEIVFD TO THE DATE. CYCL CYCL 10 ANTOT=0. CYCL CYCL 11 TOTA=SILE.0) GO TO 112 CYCL 11 TOTA=FRMINE 1F ANIMALS OF ANY CLASS	*		
TP(1)=1L+P(IDAY+MODT+NKA+TIMF+ISTRT) CYCL DAT THETP(1) CYCL OAT THETP(1) CYCL CYCL CYCL SIACN (TP). CYCL T=0.0 CYCL D0 105 1=1.5 CYCL J=7-1 CYCL TP(J)=TP(J-1) CYCL CYCL CYCL TF(J)=TP(J-1) CYCL CYCL CYCL D5 T=T*FP(J) CYCL CYCL CYCL ANTMALS OF ANIMALS ON PASTUPF (X(1)). ACTUAL MUMHER OF CYCL ANTMAL WASS OF ANIMALS ON PASTUPF (X(1)). ACTUAL MUMHER OF CYCL ANTMAL WASS OF ANIMALS ON PASTUPF (X(1)). ACTUAL MUMHER OF CYCL ANONN PACLENT (ANGS(1)). TOTAL ANIMALS PACHE (ANTOT). CYCL ANOUNT PRECEIVED TO THE DATE. CYCL CYCL ANTOT=0. CYCL IF (NCLAS_LE.0) GO TO 114 CYCL D0 113 1=1.NCLAS CYCL ANDOY CYCL CYCL ANOT=0. CYCL CYCL IF (IDAY-MKT(I)) 10K+111+112 CYCL CYCL IF (IDAY-MKT(I)) 10K+111+112 CYCL			
D4 TR=TP(1) CYCL SIACA (TP). CYCL SIACA (TP). T=0.0 D0 105 1=1.5 J=7-1 TP(J)=TP(J=1) CYCL CYCL CYCL D0 105 1=1.5 (CYCL D0 105 1=1.5 CYCL CYCL D1 TP(J)=TP(J=1) CYCL CYC	10		
<pre> *********************************</pre>	104		
STACN (TP). CYCL T=0.0 CYCL DO 105 1=1,5 CYCL J=7-1 CYCL TP(J)=TP(J-1) CYCL CYCL CYCL TF(IDAY-5) 107+107+106 CYCL CYCL CYCL ANTMALS ON PASTUGE (ANOS(1)). CYCL ANTOT=0. CYCL IF (NCLAS, LE.0) GO TO 114 CYCL OO 115 1=1. CYCL ANOS(1)=0.0 CYCL IF (IDAY-MKT(1)) 10+111+112 CYCL OB IF (IDAY-MKT(1)) 10+11+112 CYCL OD ANS(1)=STKNO(1) CYCL ANDS(1)=STKNO(1) CYCL ANTUEANIOT-ANOS(1)/ARFA CYCL GO TU 113 CYCL 11 TOTAL=TOTAL=VALUE(1) CYCL I2 X(1)=0.0			
T=0.0 CYCL D0 105 L=1+5 CYCL J=7-1 CYCL TP(J)=TP(J=1) CYCL 05 T=1*F1J CYCL 05 T=1*-2 CYCL 1F (IDAY-5) 107+107+106 CYCL 05 T=1*-2 CYCL ANIMALS ON PASTUPF (ANOS(1)), TOTAL ANIMALS PFP ACHE (ANTOT), CYCL CYCL ANIMALS ON PASTUPF (ANOS(1)), TOTAL ANIMALS PFP ACHE (ANTOT), CYCL CYCL AMOUNT RECEIVED TO THE DATE. CYCL 07 ANTOT=0. CYCL CYCL 1F (ICAS+LE.0) 60 TO 114 CYCL CYCL 00 113 1=1+NCLAS CYCL CYCL ANOS(1)=0.0 GO TO 112 CYCL IF (IDAY-IN(1)) 10+LF.0.0) GO TO 112 CYCL CYCL IF (IDAY-IN(1)) 10+LF.0.0) GO TO 112 CYCL CYCL IF (IDAY-IN(1)) 11+11+112 CYCL CYCL IF (IDAY-IN(1)) 11+10+10+110 CYCL CYCL IF (IDAY-IN(1)) 11+10+10+11+112 CYCL CYCL IF (IDAY-IN(1)) 11+10+10+10+10 CYCL CYCL IF (IDAY-IN(1)) 11+10+10+10+10 CYCL CYCL IF (ATUA+ANTOT+ANOT)) 10+ANFA CYCL	٠		
T=0.0 CYCL DO 105 1=1+5 CYCL J=7-1 CYCL TP(J)=TP(J-1) CYCL ST=T+FUJ CYCL IF (IDAY-5) 107+107+106 CYCL CYCL CYCL MIMALS ON PASTUPF (ANDS(1))+ ACTUAL MUMHER OF CYCL ANIMALS ON PASTUPF (ANDS(1))+ TOTAL ANIMALS PFP ACPE (ANTOT)+ CYCL AVEHAGE ANIMAL WFIGHT (LVWT(11)+ AND ON MARKET DAY+ CASH CYCL ANOUNT PECELVED TO THE DATE. CYCL O7 ANTOT=0. CYCL IF (NCLAS_LE*0) GO TO 114 CYCL CYCL CYCL AMOUNT PECELVED TO THE DATE. CYCL CYCL CYCL ANTOT=0. CYCL IF (NCLAS_LE*0) GO TO 114 CYCL CYCL CYCL ANOS(1)=0.0 GO TO 112 CYCL CYCL IF (IDAY-MKT(1)) 10K+11+112 CYCL CYCL GY IF (IDAY-MKT(1)) 10K+11+112 CYCL CYCL IF (IDAY-MKT(1)) 10K+11+112 CYCL CYCL ANOS(1)=STKNO(1) CYCL ANOS(1)=CXNNO(1)*A		STALK (TP).	
D0 105 1=1+5 CYCL J=7-1 CYCL CYCL TP(J)=TP(J=1) CYCL D5 T=T*TP(J) CYCL D6 T=T*P(J) CYCL CYCL CYCL CYCL ANIMALS ON PASTUPF (ANOS(1))+ TOTAL ANIMALS PFP ACPE (ANTOT)+ CYCL CYCL AMOUNT RECEIVED TO THE DATE. CYCL D7 ANTOT=0. CYCL IF (ICLAS_LE.0) GO TO 114 CYCL OO 113 1=1+NCLAS CYCL ANOS(1)=0.0 GO TO 112 IF (IDAY-MKT(I)) 10K+11+112 CYCL OB X[1]=LVWT(1) + STKNO(1) + ANEA CYCL G0 IU 13 CYCL IF (IDAY-MKT(I)) 10K+11+112 CYCL G0 CYCL CYCL IF (IDAY-MKT(I)) 10K+11+112 CYCL GO IU 113 CYCL </td <td></td> <td>T = U • O</td> <td></td>		T = U • O	
TP(J)=TP(J-1) CYCL 05 T=T*TP(J) CYCL 1F (IDAY-5) 107*107*106 CYCL 06 T=T*.2 CYCL CYCL 107 TOTAL MASS OF ANIMALS ON PASTUPF (X(I))* ACTUAL NUMHER OF CYCL ANIMALS ON PASTUPF (XNOS(I))* TOTAL ANIMALS PEP ACHE (ANTOT)* CYCL ANIMALS NO PASTUPF (XNOS(I))* ANU ON MANKET DAY* CASH CYCL ANTOT=0* CYCL CYCL 1F (IDAY-5) GO TO 114 CYCL 00 13 1=1*NCLAS CYCL ANOS(I)=0.0 GO TO 112 CYCL CYCL • • • DETERMINE IF ANIMALS OF ANY CLASS ARE ON PASTURF* CYCL CYCL • • • • DETERMINE IF ANIMALS OF ANY CLASS ARE ON PASTURF* CYCL CYCL 1F (IDAYIN(I)) 112*105*110 CYCL CYCL 08 IF (IDAYIN(I)) 112*105*110 CYCL 10 ANOS(I)=STKNO(I) ANEA CYCL CYCL 10 ANOS(I)=STKNO(I) CYCL CYCL CYCL 11 TOTAL=TOTAL=YALUF(I) CYCL CYCL CYCL <			CYCL
05 T=T+TP[J] CYCL 1F (IDAY-5) 107+107+106 CYCL 06 I=T**.2 CYCL CYCL * * * * ESTABLISH & LIVESTOCK SCHEDULF WITH THE VAPTAPLES OF CYCL CYCL TOTAL MASS OF ANIMALS ON PASTUPF (X(I)) * ACTUAL MUMMER OF CYCL CYCL ANIMALS ON PASTUPF (ANOS(1)) * TOTAL ANIMALS PEP ACHE (ANTOT) * CYCL AVERAGE ANIMAL WEIGHT (LVWT(I)) * ADVU ON MARKET DAY* CASH CYCL AVERAGE ANIMAL WEIGHT (LVWT(I)) * ADVU ON MARKET DAY* CASH CYCL CYCL AVERAGE ANIMAL WEIGHT (LVWT(I)) * ADVU ON MARKET DAY* CASH CYCL ANTOT=0* CYCL CYCL 1F (NCLAS*LE*0) GO TO 114 CYCL CYCL 00 13 T=1*NCLAS CYCL CYCL ANTOT=0* CYCL CYCL CYCL 1F (IDAY-MKT(I)) LE*0*0*010 CYCL CYCL 01 13 T=1*NCLAS CYCL CYCL 1F (IDAY-MKT(I)) LE*0*0*010 CYCL CYCL 01 13 TE*0*0*010 CYCL CYCL 1F (IDAY-MKT(I)) LE*0*0*010 CYCL CYCL 1F (IDAY-MKT(I)) LE*0*0*010 <td></td> <td></td> <td></td>			
<pre>1F (IDAY-5) 107+107+106 06 T=T*+2 CYCL CYCL CYCL TUTAL MASS OF ANIMALS ON PASTUPF (X(I))+ ACTUAL MUMHER OF CYCL ANIMALS UN PASTUPF (ANDS(I))+ TOTAL ANIMALS PFP ACRE (ANTOT)+ CYCL AVEMAGE ANIMAL WEIGHT (LVWT(I))+ ANU ON MANKET DAY+ CASH CYCL CYCL OT ANTOT=0. IF (NCLAS+LE+0) GO TO 114 OO 113 I=1+NCLAS ANOS(I)=0+0 IF (NCLAS+LE+0) GO TO 112 * * * * DETERMINE IF ANIMALS OF ANY CLASS ARE ON PASTURF. CYCL IF (IDAY-MKT(I)) 10+11+112 CYCL GO IV 113 IF (IDAY-MKT(I)) 10+11+112 CYCL GO IV 113 IF (IDAY-MKT(I)) 10+0+110 OS X(I)==5TKNO(I)/AREA GO IV 113 CYCL ANTUI=ANIOT+ANOS(I)/AREA CYCL CYCL CYCL GO TV 113 CYCL ANTUI=ANIOT+ANOS(I)/AREA CYCL CYCL CYCL CYCL CYCL CYCL CYCL CYC</pre>	10		
06 T=T*.2 CYCL 07 T=T*.2 CYCL 07 T=T*.2 CYCL 07 TUTAL MASS OF ANIMALS ON PASTUPF (x(1)), ACTUAL MUMHER OF CYCL ANIMALS ON PASTUPF (ANOS(1)), TOTAL ANIMALS PFP ACHE (ANTOT), CYCL ANTOTOT, CYCL ANUMALS ON PASTUPF (ANOS(1)), AND ON MARKET DAY, CASH CYCL AVERAAGE ANIMAL WFIGHT (LVWT(1)), AND ON MARKET DAY, CASH CYCL AMOUNT RECEIVED TO THE DATE. CYCL 07 ANTOT=0. CYCL CYCL 1F (NCLAS,LE.0) GO TO 114 CYCL CYCL 00 113 1=1,NCLAS CYCL CYCL ANOS(1)=0.0 CYCL CYCL 1F (IDAY-MKT(1)) LF.0.0) GO TO 112 CYCL CYCL 00 113 1=0.0 CYCL CYCL 014 IF (IDAY-MKT(1)) 108+111+112 CYCL CYCL 015 IF (IDAY-MKT(1)) 108+111+112 CYCL CYCL 016 IF (IDAY-MKT(1)) 108+111+112 CYCL CYCL 02 X(1)=LVWT(1)*STKNO(1)/AREA CYCL CYCL 03 IF (IDAY-MKT(1)) 112+105+110 CYCL CYCL 04 IF (IDAY-MKT(1)) 112+105+110 CYCL CYCL 05 II = STKNO(1) CYCL CYCL <tr< td=""><td></td><td></td><td></td></tr<>			
<pre>* * * * * ESTABLISH & LIVESTOCK SCHFPULE WITH THE VAPIAPLES OF TOTAL MASS OF ANIMALS ON PASTUPF (X(I)) * ACTUAL MUMHER OF CYCL ANIMALS UN PASTURF (ANOS(I)) * TOTAI ANIMALS PFM ACHE (ANTOT) * CYCL AVEMAGF ANIMAL WFIGHT (LVWT(I)) * ANU ON MARKET DAY, CASH AMOUNT RECEIVED TO THE DATE. CYCL OT ANTOT=0. IF (NCLAS_LE.0) GO TO 114 OO 113 1=1,NCLAS ANOS(I)=0.0 IF (STKNU(I).LF.0.0) GO TO 112 * * * * DETERMINE 1F ANIMALS OF ANY CLASS ARL ON PASTURF. CYCL IF (IDAY-MKT(I)) 108*111*112 CYCL OB IF (IDAY-IN(I)) 112*109*110 CYCL GU IU 113 CYCL GU IU 113 CYCL ANTUI=ANIOT*ANOS(I)/AREA GO TU 113 CYCL * * * * TAKE ANIMALS OFF PASTURE. CYCL CYCL CYCL CYCL CYCL CYCL CYCL CYC</pre>	10		CYCL
TOTAL MASS OF ANIMALS ON PASTUPF (x(I)). ACTUAL MUMBER OF CYCL ANIMALS UN PASTURF (ANOS(I)). TOTAL ANIMALS PFM ACHE (ANTOT). CYCL AVEMAGE ANIMAL WEIGHT (LVWT(I)). ANU ON MARKET DAY. CASH CYCL AMOUNT RECEIVED TO THE DATE. CYCL 07 ANTOT=0. CYCL IF (NCLAS.LE.0) GO TO 114 CYCL 00 113 1=1.NCLAS CYCL ANOS(I)=0.0 CYCL 1F (STRNU(I).LF.0.0) GO TO 112 CYCL * * * DETERMINE IF ANIMALS OF ANY CLASS ARE ON PASTURF. CYCL CYCL CYCL 08 IF (IDAY=MKT(T)) 10H+1)1+112 CYCL 09 X(I)=CVWI(I)*STKNO(I)/AREA CYCL GO IU 113 CYCL ANOS(I)=STKNO(I) CYCL ANDS(I)=STKNO(I) CYCL II CYCL II TOTAL=TOTAL*VALUE(I) II CYCL CYCL <		CETABLICS & LINCETON POURDLE STOL FOR MORTON PO OF	
ANIMALS ON PASTURF (ANOS(1)), TOTAL ANIMALS PFM ACHE (ANTOT), CYCL AVERAGE ANIMAL WEIGHT (LVWT(I)), AND ON MARKET DAY, CASH AMOUNT RECEIVED TO THE DATE. O7 ANTOT=0. IF (NCLAS,LE.0) GO TO 114 O0 113 1=1.NCLAS ANOS(1)=0.0 IF (STKND(1)).LF.0.0) GO TO 112 • • • • DETERMINE 1F ANIMALS OF ANY CLASS ARE ON PASTURF. CYCL IF (IDAY-MKT(I)) 108.111.112 O8 IF (IDAY-NKI) 112.109.110 O9 X(1)=LVWT(I)*STKND(I)/AREA GO 10 113 O1 ANTOL=ANOS(1)/STKND(I)/AREA GO TO 113 CYCL • • • • TAKE ANIMALS OFF PASTURE. CYCL 2 X(1)=0.0 CYCL 2 X(1)=0.0 CYCL	0		
AVFRAGE ANIMAL WFIGHT (LVWT(I))* AND ON MARKET DAY* CASH AMOUNT RECEIVED TO THE DATE* CYCL CYCL CYCL CYCL CYCL CYCL CYCL O7 ANTOT=0* CYCL CYCL CYCL CYCL O7 ANTOT=0* CYCL CYCL CYCL CYCL CYCL CYCL CYCL CYCL			
07 ANTOT=0. CYCL IF (NCLAS.LE.0) GO TO 114 CYCL D0 113 1=1.NCLAS CYCL ANOS(I)=0.0 CYCL 1F (STKNU(I).LF.0.0) GO TO 112 CYCL • • • • DETERMINE 1F ANIMALS OF ANY CLASS ARL ON PASTURF. CYCL • • • • DETERMINE 1F ANIMALS OF ANY CLASS ARL ON PASTURF. CYCL 01 IF (IDAY-MKT(I)) 10H.111.112 CYCL 02 X(I)=UVWT(I)*STKNU(I)/AREA CYCL 03 X(I)=UVWT(I)*STKNU(I)/AREA CYCL 04 IF (IDAY-IN(I)) 112.104.110 CYCL 05 X(I)=UVWT(I)*STKNU(I)/AREA CYCL 05 X(I)=STKNU(I) CYCL 06 IU 113 CYCL 10 ANDS(I)=STKNU(I) CYCL ANTU!=ANIOT+ANOS(I)/AREA CYCL 11 TOTAL=TOTAL+VALUF(I) CYCL 12 X(I)=STKNU(I)*AREA CYCL 13 COTU 113 CYCL 14 IF (X(IU)=KONS(I)/AREA CYCL 15 COTU 113 CYCL 14 IF (X(IU)=KONS(I)/AREA CYCL 15 COTU 13		AVENAGE ANIMAL WEIGHT (LVWT(I)). AND ON MARKET DAY. CASH	CYCL
07 ANTOT=0. CYCL IF (NCLAS_LE.0) GO TO 114 CYCL 00 13 1=1.NCLAS CYCL ANOS(I)=0.0 CYCL CYCL 1F (STKNU(I).LF.0.0) GO TO 112 CYCL • • • • DETERMINE 1F ANIMALS OF ANY CLASS ARE ON PASTURF. CYCL 09 (I)A (I)A (I) (I)A (I) (I)A (I)A (I)A (I			
<pre>IF (NCLAS.LE.0) GO TO 114 OO 113 1=1.NCLAS ANOS(I)=0.0 IF (STKNU(I).LF.0.0) GO TO 112 CYCL</pre>	0		
D0 113 1=1+NCLAS CYCL ANDS(I)=0.0 GYCL CYCL 1F (STKNU(I).LF.0.0) GO TO 112 CYCL • • • • DETERMINE 1F ANIMALS OF ANY CLASS ARL ON PASTURF. CYCL 01 1F (IDAY-MKT(I)) 108+111+112 CYCL 02 1F (IDAY-MKT(I)) 112+109+110 CYCL 03 IF (IDAY-IN(I)) 112+109+110 CYCL 04 IF (IDAY-IN(I)) 112+109+110 CYCL 05 X(I)=UVWI(I)=STKNU(I)/AREA CYCL 05 X(I)=IVWI(I)=STKNU(I)/AREA CYCL 05 ANDS(I)=STKNU(I)/AREA CYCL 06 UUI13 CYCL CYCL 10 ANOS(I)=STKNU(I)/AREA CYCL CYCL 06 UUI13 CYCL CYCL 11 TOTAL=TOTAL=VALUE(I) CYCL CYCL 12 X(I)=0.0 CYCL CYCL 13 CONTINUE CYCL CYCL 14 IF (X(IU).LF.0.0) X(10)=0. CYCL CYCL 15 CONTINUE CYCL CYCL CYCL			
1F (STKNOUT).LF.0.0) GO TO 112 CYCL 1F (STKNOUT).LF.0.0) GO TO 112 CYCL 1F (IDAY-MKT(I)) 108.111.112 CYCL 1F (IDAY-MKT(I)) 108.111.112 CYCL 08 IF (IDAY-IN(I)) 112.109.110 CYCL 09 X(I)=LVWT(I).95TKNU(I)/AREA CYCL 09 X(I)=LVWT(I).95TKNU(I)/AREA CYCL 10 ANOS(I)=STKNU(I)/AREA CYCL 60 IU 113 CYCL 10 ANOS(I)=STKNU(I)/AREA CYCL 60 IU 113 CYCL 11 TOTAL=TOTAL.*VALUE(I) CYCL 12 X(1)=0.0 CYCL 13 CONTINUE CYCL 14 IF (X(LU).LT.0.) X(10)=0. CYCL 15 CONTINUE CYCL 16 IF (X(LU).LT.0.) X(10)=0. CYCL 17 CYCL CYCL 18 CONTINUE CYCL 19 IF (X(12).LF.0.0) X(11)=0. CYCL 19 IF (X(12).LF.0.0) X(12)=0.0 CYCL 11 IF (X(12).LF.0.0) X(12)=0.0 CYCL 11 IF (X(12).LF.0.0) X(12)=0.0 CYCL <		00 113 1=1.NCLAS	CYCL
<pre>CYCL</pre>			
 • • • • DETERMINE IF ANIMALS OF ANY CLASS ARE ON PASTURF. CYCL IF (IDAY-MKT(I)) 108+111+112 CYCL ANDS(I)=STKHO(I) ANDS(I)=ANEA CYCL ANTUL=ANIOT+ANOS(I)/ARFA CYCL ANTUL=ANIOT+ANOS(I)/ARFA CYCL ANTUL=ANIOT+ANOS(I)/ARFA CYCL CYCL			
IF (IDAY-MKT(I)) 108+111+112 CYCL 08 IF (IDAY-IN(I)) 112+105+110 CYCL 09 X(I)=LVWT(I)*5TKN0(I)/AREA CYCL 09 X(I)=LVWT(I)*5TKN0(I)/AREA CYCL 010 ANOS(I)=STKN0(I) CYCL ANTU[=ANIOT*ANOS(I)/AREA CYCL LVWI(I)=x(I)/STKN0(J)*AREA CYCL GO TU 113 CYCL 11 TOTAL=TOTAL*VALUE(I) CYCL * * * * TAKF ANIMALS OFF PASTURE* CYCL 12 X(1)=0.0 CYCL 13 CONTINUE CYCL 14 IF (X(12), (I+0)*X(10)=0. CYCL 15 X(1)=0.0) CYCL 16 IF (X(12).L(*.0.0) X(1))=0. CYCL 17 IF (X(12).L(*.0.0) X(1))=0. CYCL 18 K(1).L(*.0.0) X(1))=0. CYCL 19 CON) X(1))=0.0 CYCL 19 CNO) X(1))=0.0 CYCL 19 CNO) X(1))=0.0 CYCL 19 K(10).L(*.0.0) X(1))=0.0 CYCL 19 CNO) X(1))=0.0 CYCL 19 CNO) X(2))=0.0 CYCL			
08 IF (IDAY-IN(I)) 112+109+110 CYCL 09 X(I)=LVWT(I)=STKNU(I)/AREA CYCL GU UU 113 CYCL 10 ANOS(T)=STKNU(I) ANTUL=ANIOT+ANOS(I)/AREA CYCL LVWI(I)=X(I)/STKNU(I)*AREA CYCL GO TU 113 CYCL I1 TOTAL=TOTAL*VALUE(I) CYCL * * * TAKF ANIMALS OFF PASTURE* CYCL I2 X(I)=0.0 CYCL I4 IF (X(LU)*(T*0*) X(10)=0* CYCL IF (X(I1)*LE*0.0) X(11)=0.0 CYCL IF (X(12)*LE*0.0) X(11)=0.0 CYCL IF (X(12)*LE*0.0) X(12)=0.0 CYCL IF (X(12)*LE*0.0) X(12)=0.0 CYCL RETURN CYCL ShD CYCL			CYCL
09 X(T)=LVWT(T)+STKNU(T)/AKEA CYCL G0 10 113 CYCL 10 ANDS(T)=STKNU(T) CYCL CYCL ANTUT=ANIOT+ANOS(T)/AREA CYCL CYCL LVWT(T)=X(T)/STKNU(T)*AREA CYCL CYCL GO TU 113 CYCL CYCL 11 TOTAL=TOTAL*VALUE(T) CYCL * * * * TAKF ANIMALS OFF PASTURE* CYCL 12 X(T)=0.0 CYCL 13 CONTINUE CYCL 14 F(X(LU),(T+D)*X(10)=0. CYCL 1F(X(LU),(T+D)*X(10)=0. CYCL 1F(X(LU))*(E+0.0) CYCL 1F(X(LU)*(E+0.0) CYCL 1F(X(LU)*(E+0.0) CYCL 1F(X(LU)*(E+0.0) CYCL 1F(X(LU)*(E+0.0) CYCL 1F(X(LU)*(E+0.0) CYCL 1F(X(LU)*(E+0.0) </td <td>0</td> <td></td> <td></td>	0		
G0 10 113 CYCL 10 ANOS(T)=STKHO(T) CYCL ANTU=ANIOT+ANOS(T)/ARFA CYCL LVWT(T)=X(T)/STKNO(T)*ARFA CYCL G0 TU 113 CYCL 11 TOTAL=TOTAL*VALUE(T) CYCL * * * * TAKF ANIMALS OFF PASTURE* CYCL 12 X(T)=0.0 CYCL 13 CONTINUE CYCL 14 1F (X(LU)+(T+0+) X(T0)=0+ CYCL 15 CONTINUE CYCL 14 1F (X(LU)+(T+0+) X(T0)=0+ CYCL 15 (ANTINUE CYCL 16 (X(LU)+(T+0+) X(T0)=0+ CYCL 17 (X(L1)+(E+0+0) X(T0)=0+ CYCL 18 (X(L1)+(E+0+0) X(T1)=0+0+ CYCL 19 (X(L1)+(E+0+0) X(T0)=0+0+ CYCL 11 (X(L1)+(E+0+0) X(T0)=0+0+ CYCL 12 (X(L1)+(E+0+0) X(T0)=0+0+ CYCL 13 (X(L1)+(E+0+0) X(T0)=0+0+ CYCL 14 (X(L1)+(E+0+0) X(T0)=0+0+ CYCL			
10 ANOS(T)=STKNO(T) CYCL ANTU[=ANIOT+ANOS(1)/ARFA CYCL LWWI(I)=x(1)/STKNO(J)*ARFA CYCL GO TU 113 CYCL 11 TOTAL=TOTAL+VALUE(T) CYCL • • • • TAKF ANIMALS OFF PASTURE. CYCL 12 X(1)=0.0 CYCL 13 CONTINUE CYCL 14 F(X(LU)+(T+0)) X(10)=0. CYCL 15 CNTINUE CYCL 16 T(X(LU)+(T+0)) X(10)=0. CYCL 17 F(X(11)+(E+0+0) X(10)=0. CYCL 18 CHI)+(E+0+0) X(10)=0. CYCL 19 F(X(12)+(E+0+0) X(11)=0.0 CYCL 19 F(X(12)+(E+0+0) X(11)=0.0 CYCL 19 F(X(12)+(E+0+0) X(12)=0.0 CYCL 19 F(X(12)+(E+0+0) X(12)=0.0 CYCL 11 F(X(12)+(E+0+0) X(12)=0.0 CYCL 11 F(X(12)+(E+0+0) X(12)=0.0 CYCL 11 F(X(2)+(E+0+0) X(2)=0.0 CYCL 11 F(X(2)+(E+0+0) X(2)=0.0 CYCL 12 F(X(2)+(E+0+0) X(2)=0.0 CYCL 14 F(GU 1U 113	
LVWI(I)=X(1)/STKNO(I)*AREA GO TU 113 II TOTAL=TOTAL+VALUE(I) * * * * TAKE ANIMALS OFF PASTURE. CYCL * * * * TAKE ANIMALS OFF PASTURE. CYCL	1	ANO5(T)=STKHO(T)	CYCL
GO TU 113 CYCL 11 TOTAL=TOTAL+VALUE(I) CYCL * * * * TAKF ANIMALS OFF PASTURE. CYCL 12 X(1)=0.0 CYCL 13 CONTINUE CYCL 14 IF (X(1U).(T.*0.) X(10)=0. CYCL 15 CONTINUE CYCL 14 IF (X(1U).(T.*0.) X(10)=0. CYCL 15 (X(11).(E.0.0) X(11)=0.0 CYCL 16 (X(12).(E.0.0) X(12)=0.0 CYCL 17 (X(21).(E.0.0) X(12)=0.0 CYCL 18 (X(21).(E.0.0) X(21)=0.0 CYCL 19 (X(21).(E.0.0) X(21)=0.0 CYCL 11 (X(21).(E.0.0) X(21)=0.0 CYCL 12 (X(21).(E.0.0) X(21)=0.0 CYCL 13 (X(21).(E.0.0) X(21)=0.0 CYCL 14 (X(21).(E.0.0) X(21)=0.0 CYCL 15 (X(21).(E.0.0) X(21)=0.0 CYCL 15 (X(21).(E.0.0) X(21)=0.0 CYCL 14 (X(21).(E.0.0) X(21)=0.0 CYCL 15 (X(21).(E.0.0) X(21)=0.0 CYCL 15 (X(21).(E.0.0) X(21)=0.0 CYCL 16 (X(21).(E.0.0) X(21)=0.0 CYCL 17 (X(21).(E.0.0) X(21)=0.0 CYCL 18 (X(21).(E.0.0) X(21)=0.0 CYCL 17 (X(21).(E.0.0) X(21)=0		A MART P R A MART A MART A MART A MART A MART	
11 TOTAL=TOTAL+VALUE(I) CYCL • • • • TAKE ANIMALS OFF PASTURE. CYCL 12 X(1)=0.0 CYCL 13 CONTINUE CYCL 14 F(X(1U), (T+0.)) X(10)=0. CYCL 15 (X(11), (T+0.)) X(10)=0. CYCL 16 17 (X(12), (T+0.0) X(11)=0.0 CYCL 17 (X(12), (T+0.0) X(11)=0.0 CYCL CYCL 18 (X(12), (T+0.0) X(11)=0.0 CYCL CYCL 19 (X(12), (T+0.0) X(12)=0.0 CYCL CYCL 19 (X(12), (T+0.0) X(12)=0.0 CYCL CYCL 19 (X(12), (T+0.0) X(21)=0.0 CYCL CYCL 19 (T+0.0) X(21)=0.0 CYCL CYCL 19 (T+0.0) X(21)=0.0 CYCL CYCL 10 (T+0.0) X(21)=0.0 CYCL CYCL		60 TU 113	
• • • • • TAKE ANIMALS OFF PASTURE. CYCL 12 X(1)=0.0 CYCL 13 CONTINUE CYCL 14 1F (X(1u).(T*0.) X(10)=0. CYCL 15 (X(11).(E.0.0) X(1))=0.0 CYCL 16 14(12).(E.0.0) X(1)=0.0 CYCL 17 (X(11).(E.0.0) X(1))=0.0 CYCL 18 (X(21).(E.0.0) X(21)=0.0 CYCL 19 CYCL CYCL 19 CYCL CYCL 10 CYCL CYCL	11		
12 X(1)=0.0 CYCL 13 CONTJNUE CYCL 14 1F (X(1))=(1,0.0) X(10)=0.0 15 (X(1))=(1,0.0) X(1))=0.0 CYCL 16 1F (X(1))=(1,0.0) X(1))=0.0 CYCL 17 (X(2))=(1,0.0) X(1))=0.0 CYCL CYCL 18 (X(2))=(1,0.0) X(2))=0.0 CYCL CYCL NETURN CYCL CYCL CYCL			CYCL
12 X(1)=0.0 CYCL 13 CONTINUE CYCL 14 F(X(10)+(1+0+) X(10)=0+ CYCL 1F (X(11)+L+0+0) X(10)=0+ CYCL 1F (X(12)+L+0+0) X(10)=0+ CYCL 1F (X(12)+L+0+0) X(12)=0+0 CYCL 1F (X(21)+L+0+0) X(21)=0+0 CYCL RETURN CYCL	•		
14 1F (X(1U).(T.0.) X(10)=0. CYCL IF (X(1).(E.0.0) X(1))=0.0 CYCL CYCL IF (X(21).(E.0.0) X(1))=0.0 CYCL CYCL IF (X(21).(E.0.0) X(2))=0.0 CYCL CYCL RETURN CYCL CYCL CYCL	113	5 X(1)=0.0	
IF (X(11),LE,0,0) X(1))=0.0 CYCI IF (X(12),LE,0,0) X(1))=0.0 CYCI IF (X(21),LE,0,0) X(2))=0.0 CYCI IF (X(21),LE,0,0) X(2))=0.0 CYCI ShO CYCI CYCI		15 dealers and a sector of	
IF (x(12), 1, E, 0, 0) x(12) = 0, 0 CYCI IF (x(21), 1, E, 0, 0) x(21) = 0, 0 CYCI RETURN CYCI SND CYCI	+ 1 '	The should be a set of the set of	
IF (X(21),(E,0,0), X(21)=0,0) CYCI RETURN CYCI SND CYCI		1F (x(12), 1E = 0, 0) x(12) = 0.0	
CYCI		$IF (x(21)) = (E_0 0_0) x(21) = 0_0$	CYCI
E MD			
		END	CYCL

		SUBRUITINE CYCLS	CAC(5001
С			CACI 5005
С		• • • • THIS SUBROUTINE (CYCL2) IS CALLED BY THE SUBROUTINE	CACI 5003
С		(CYCLE) TO CALCULATE AUXILLARY FINANCIAL VARIABLES SHOWING	CYCL 2004
С		STATUS OF THE LIVESTOCK ENTERPRISE BY THE USE OF MONTHLY	CYCL 2005
c		PHICE GENERATOR (NRM).	CYCL 2006
C			CYCL 2007
-		INTEGEN INTDAY . IN. INFR. INTR. IYR. JUDAY. N. M. MKI . MON	CYCL 2008
		INTEGEN NCLAS NUFT XNF	CYCL 2009
		REAL ANUS + ANTOT + AFE A + HASE + CASH + DIIT + DT + F CUNS + GAIN + GNIN	CACT 5010
		REAL LYWT . LWT . MSZ . UMIN . PHCOV . PHICE . PHMN . STENU . SUPL . SUPN	CYCI 2011
		REAL SUPP.SUPPR.SUPPT.SUPPT.TIME.ISTRT.VALUE.A.XIOTO	CYCI 2012
		HEAL XIITO.ZZ	CYCI 2013
С			CYCL 2014
		COMMON /CAT/ ANOS(S) +ANTOT + ARE A + DIIT + FCUNS + GAIN(S) + GNIN(S) + IN(S) + I	
		INFD (5) +LVWT (5) +LWT (5) +MKT (5) +MSZ (5) +NCLAS+NOFU (5) +OMIN (5) +STKNO (5)	
		2+SUPL (5) + SUPPH (5) + SUPPH (5) + SUPPH (5)	CYCL2017
		COMMON /ECON/ HASE (5) + CASH (5) + PRCOV (144) + PRICE (12) + PRMN(12) + SUPP (*	
	1	[] • SUPWI (5) • VALUE (5) • 77 (12)	CYCL 2019
		COMMUN /TT/ 1DAY+INYH+IYH+JUDAY+K+MON	CYCI 2020
		COMMON $/TX/UT+THE_T5THT+X(99)+ANE+X1UT0+X1110$	CYCI 2021
С			CYCI 2022
		M=UI+0.5	CYCI 2023
		1F (1DAT.GT.M) GO TO 101	CYCL 2024
		CALL NEM (PRICE PRCOV PRMN + 12 + 27)	CYCL 2025
	101	1F (PH) LF (MUN) + LF + [+0] PP1CE (MUN) = 1 + 0	CYCL 2026
	101	IF (NCLAS.(F.0) RETURN	CYCI 2027
С		IT INCLASSIFSUI RETURN	CYCL 2027
č		WHEN LIVESTOCK ARE PUT OUT TO PASTURE (IDAY=IN(1)) THE	CYCL 2029
č		PRICE OF CATTLE IS UTILIZED AS A BASE PPICE FOR THE CLASS I.	
c		CALCULATIONS ARE MADE FOR TOTAL AMOUNT PAIL FOR	CYCL 2030
č		SUPPLEMENTAL FEED (SUPP(1)) BY MULTIPLYING AMOUNT OF FEED	CYCL 2031
č		(SUP #T(1)) AND PRICE OF FFED (SUPPR(1)). ALSO CALCULATIONS	CYCL 2032
č		ARE MADE FOR AMOUNT INVESTED (CASH(1)). AVERAGE WEIGHT	CYCL2033
0000		(LWT(I)) AND POTENTIAL PROFITO IF ANIMALS SOLD NOW (VALUE(I)).	CYCI 2034
č		CENTER AND FORTHAL PROFILE IF ANTALS SUCH NOW TRADECING.	
6		DO 103 1=1.NCLAS	CYCI 2036
		IF (IDAY.EQ.IN(I)) BASE([)=PRICE(MON)*LVWT(I)*STKN0(1)/45400.0	CYCL 2037
		SUPP(1)=SUPWT(1)*SUPPr(1)	
		CASH(I) = HASE(I) + SUPP(I)	CYCL 2039
		LwT(1)=0.0	CYCL 2040
		IF (INT(STKNO(1)).LE.0) GO TO 102	CYCI 2041
		IF (A(I).LT.0.01) GU 10 102	CYCL2042
		LbT(1) = x(1) = AAFA/(454.0 = STRN(1))	CYCL2043
		VALUE (I)=PRICE (MUN)*LVWI(I)*SIKNO(I)/45400+0-CASH(I)	CYCL 2044
		GO TU 103	CYCL 2045
	102	VALUE (I)=0.0	CYCL 2046
	100	X(I)=0.0	CYCL 2047
	103	CONTINUE	CYCL2048
	103	RETURN	CYCL 2049
С		IT FAILING	CYCL 2050
-		END	CYCL 2051
			CICLEUSE

				REAL FUNCTION ET(SHOTHOFCAPOTHULDOWILT)	ET	001
	C C			• • • • THIS FUNCTION (FT) IS CALLED BY THE SURGOUTINE (XELWS)	ET	002 003
	C C			TO CALCULATE EVAPOTRANSPIRATION USING MEAN AIR TEMPERATURE (TR) AS THE DETERMINANT.	FT	004
	č			HEAL FOFFCAP SWOTHOLDO IN SHOULT	FT	006
	С				FT	00A
				ET≖0.0 16 (TP.LF.0.1) GO TO 102	FT	009
				£()=.6.4.1μΦΤκ 1F (FCAP.LF.0.0) 60 TO 101	ET	011
	C C			• • • • IF SOIL WATER (Sw) IS LESS THAN FIFLU CAPACITY (FCAP).	FT ET	013
	С		•	ACTUAL EVAPOIRANSPINATION (ET) IS PROPORTIONAL TO THE RELATIVE	ET	015
	C C			WATER CONTENT OF THE SUIL (FO).	ET.	016
	С			ET=EU+S#/FCAP	FT ET	018
	C C			 IF SULL WATER (SW) IS GREATER THAN OR FOUND TO FIELD CAPACITY (FCAP), ACTUAL EVENDERANSPLEATION (ET) IS TAKEN TO BE 	ET FT	020
	C C			THE SAME AS POTENTIAL EVAPOIRANSFIRATION (LU).	FT FT	022
	c	1	01	IF (SW+GF+FCAP) ET=FO	ET	024
	С			• • • IF SOIL WATER (SW) EXCEEDS TOTAL WATER HOLDING CAPACITY	ET	026
	c c			OF SUIL (ED) + THE EXCESS IS SET EQUAL TO EI.	ET	027 028
		1	02	1F (Sw.GT.THOLD) ET≃((2.*Sw-THOLD)/SW)*ET 1F (ET.LE.0.0) ET=0.0	ET.	029 030
	C C			• • • • IF SOIL WATER (SW) IS LESS THAN OR EVUAL TO THE MINIMUM	FT	031
	C			WATER NEEDS OF THE PLANIS (WILT) . FT IS SET EDUAL TO 1/100 OF ET.	ET	033
	č			IF (SW+LE,WILT) ET=+01*ET	ET	035
				RETURN	ET	037
	С			END	ET	03A 039
				SUMHOUTINE GHOW (WHM+CHM+SH+TR+SLA+SLA+GH+GC+FCAP+WTLT+K)	GROW	001
	C C			• • • THIS SUBROUTINE (GROW) IS CALLED BY THE SUBROUTINE	GROW	
	C C			(XFL#S) TO CALCULATE COUL AND WARM SPASON PLANT GROWTH.	GROW	004
	1			INTEGER K. K. K. C. K. W. KFAL CHM + CMAX + CK+ FSML + ESMW + FTC + EIW + FWK + FCAP	GROW	006
	~			PEAL GC+GW+UC+SLA+SLH+SW+TR+WHM+WILT+WMAX+WRMAX	GROW	008
	С			CUMMUN THESPT CMAX. CHMAX. WMAX. WRAX	GROW	
	C C			• • • • INITIALIZE GROWTH DAYS COUNTER AND CUMPUTE SCALING	GROW GROW	
	C C			HASES ON SULL-WATER PARAMETERS IF FIRST CALL TO THE SUMMUUTINE.	GRDW GROW	
	С			1F (K.L.L.O) GO TO 101	GROW	015
		14	. 1	GU TU 102 Gw=5.0/(FCAP-w1LT)	GROW	017
				QC=1./(FCAP-WILT)	GROW	019
				Kw=1 NC=1	GROW	
	C C			• • • • CALCULATIONS ARE MADE ON THE EFFECT OF SOIL WATER AND	GROW	
	C C			TEMPERATURE ON WARM SEASON GROWTH, AND NET PHOTOSYNTHESIS (GW) BY SURTRACTING RESPIRATION FROM PHOTUSYNTHESIS, IF	GROW	024
	C C			FIVE DAYS GROWTH INITIATION IS PAST. THEN SEVENTY PERCENT	GHOW	026
	c			OF PHOTOSYNTHATE IS TRANSLOCATED TO THE RUUTS.	GROW	850
		10	2(£SMw≈1EXP(-Qw*(Sw-w1L7))	GROW	
				IF (ESMw.LT.0.) FSMw=0. ETw=1(TR-25.)002/(15.015.)	GROW	
				IF (ETW+LT+0+) FTW=C+ EWR=+01*EXP(+161*TP)	GROW	033
				GW=ESMW ⁰ FTW ⁰ WHMAWHAX-FWR ⁰ WHM ⁰ WHMAX IF (GW) 104+103	GROW	035
		10	3	Kw=Kw+1 IF (Kw₀GT₀5) Gw=Gw⊕₀3	GRON	037
	C				GROW GROW	039
	C	٠	•	• • • • CALCULATIONS ARE MADE THE SAME WAY FUR COOL SEASON PLANIS AS FUR WARM SEASUN PLANIS ARUVE.	GROW	
	С	10)4	IF (CHM+LE+0+) GO TO 106	GROW	
				ESMC=OC*(Sw+W1LT) IF (ESMC=LT+0+) FSMC=0+	GROW	
				1F (ESMC.67.1.) FSMC=1. ETC=1.~(TP-1A.)**2/(16.*16.)	GROW	046
				IF (ETC.LT.U.) FTC=0. ECR=.03°FXP(.126°TR)	GROW	04R
				GC=LSMC°ETC°CHM°CMAX-LCP°CBM°CHMAX	GROW GROW	050
		10	5	IF (GC) 106,106,105 KC=KC+1	GR0₩ GR0₩	052
		10	6	IF (KC.GT.5) GC=GC+.3 RETURN	GROW	
1	С			END	GROW	-

SURROUTINE NITRO (RGR.NGN.PH.PHOI.CH.VITA.PW) NITHDOOL NITRO002 С . THIS SUBROUTINE (NITRO) IS CALLED BY THE SUBROUTINE С NITPO003 (AFLWS) TO COMPUTE PLANT NUTHIENTS IN GREEN FORAGE FROM KNOWN PELATIONS WITH NITHOGEN. NUTHIENT CUNTENTS OTHER NITPORO4 c NITPOROS C C C C THAN NITHOGEN ARE NITHOGEN BASED. THE NGN/RGP RELATIONSHIP NITROOOG IS FROM UNESK 1972. NITE0007 NITHCOOR HEAL CH+NGN+PH+PROT+PH+RGH+VITA NITPODOS С NITROOLO N(-N=1-0 NITRO011 С NITROOIS c HELATIONSHIP DEVELOPED BETWEEN RELATIVE GROWTH RATE OF NITRD013 FURAGE (HGR) AND NITHUGEN CONTENT OF FORAGE (NGN). NITROA14 Ċ NITROALS NITPOOLS IF (RGR+11-0.05) NGN=0.891 IF (HGH.GT.0.142) NGN=3.0 NITRO017 IF (HGR.GF.-0.05.AND.HGH.LE.0.142) NGN=10.9*HGR+1.436 NITPOOLA PH=NGN+0.19 NITRO019 PHUT=NUN+6.25 NITRO020 CH=2.0*NGN-1.0 NITHOOZI VITA=CHª0.5 S2004TIN PW=30.0*NGN NITRO023 NITRON24 RETURN С NITHD025 END NITRO026 SUBROUTINE NRM (PRICE+PRCOV+PRMN+N+ZZ) NRM 001 c NRM 002 • THIS SUBROUTINE (NRM) IS CALLED BY THE SUBROUTINE (CYCLP) AS A PRICE GENERATOR CALCULATING A VECTOR DE NRM 003 . CCCC NRM 004 DEPENDENT NORMAL RANDOM VARIABLES FROM THE SQUARE ROOT NRM 005 MATRIX (PRCOV), WHERE PRCOV = PRCOV = V. IS THE COVARIANCE MATRIX, THE MATRIX (PRCOV) AND THE APRAY (PRMN) APE NRM 006 NRM 007 CC DISCUSSED IN THE METHODS OF NAYLOR, BALINIFY, HURDICK. NRM 008 AND CHU, 1966. NRM 009 C NPM 010 INTEGER I.J.N NRM 011 REAL PRCOV (N+N) +PHICE (N) +PHNN (N) +R1+R2+ZZ (N) NRM 012 C NRM 013 DO 101 1=1.N NRM 014 С NDM 015 Ĉ GENERATE RANDOM FLUCTUATIONS. NRM . . 016 C NRM 017 R1=RANE (1.1) NRM 01A R2=HANF (1.1) NRM 019 101 ZZ(1)=SIN(6.283185*R1)*SORT(-2.0*ALOG(K2)) NRM 020 00 102 1=1.N PRICE(I)=0.0 NRM 021 NRM 022 00 102 J=1.N NRM 023 С NRM 024 CC CALCULATE DEVIATIONS USING COVARIANCE MATRIX. NRM 025 NRM 026 102 PRICE(1)=PRICE(1)+Z7(J)+PRCDV(1,J) NDM 027 00 103 1=1.N NRM 028 CC NRM 029 CALCULATE PRICE FROM DEVIATIONS AND MEAN PRICE. NRM 030 C NRM 031 103 PRICE(I)=PRICE(I)+PRMN(1) NRM 032 RETURN NRM 033 С NRM 034 END NRM 035

				NEAL FUNCTION PPI((NAY+MUNIPFILME+15TH)+000)	PP1	001
	Ľ			a sea and an the second of a second	PDT	002
	C C		٠	I CALCULATE PHYCTPLIATION. DE ALGUNITHE REQUIES A WET PAY	144	004
				PHECENED BY A DAY DAY AND A WET DAY PREFEDED BY A WET DAY.	THH	005
	C C C			THUS THE CONVENSION IS MADE BY TANTING THE COMPLEMENT OF THE	PPT	006
	C C			INPUT PROBABILITIES.	PPT	607
	6			INTEGER INTIAYOR OLUS MULTIPONAME (3) ONRAON COLLOUD	PPT	800 800
				HEAL A (4) + A1 + H (4) + M + AY + H + H + H + H + H + H + H + H + H +	PPT	010
	_			HEAL HOIIMFOISTHIOXLAM (365)	PPT	011
	C			COMMON /XUNT/ LU+01+02+03	PPT	012
	С				PPT	014
	С			• • • • CHFCK FOR EQUALITY OF TIME AND ISTRT.	PPT	015
	С			IF (11ME.GT.TSTRT) 60 10 104	PPT	016
				IF (MODH.NF.3) GO TU 103	PPT	018
-	С				PPT	019
1	C C	٠		 PEAD FOURTHE COFFETCIENTS REPRESENTING PRECIPITATION PARAMETERS 1F TIME = START AND STOCHASTIC WEATHER WANTED. 	PPT	020
ì	č			FARANCIERS IF IT THE - START WIN STUDIATING MEALDER PARTEIN	PPT	021
				READ (U1 + 117) (EAME(1) + 1 = 1 + 3)	PPT	023
	~			WRITE (U2+118) (NAME(1)+1=1+3)	PPT	024
	C			· · · PEAD COFFFICIENT FOR DO.	PPT	025
	č				PPT	027
				HEAD (U1+114) NT+A1	PPT	028
				<pre>kEAD (Ul+)?0) (A(1)+H(1)+L=1+N1) wkltE (U2+122)</pre>	PPT	029
				WHITE (U2.121) NT.A1. (A(I) + (I) + (I) + = 1 + NT)	PPT	031
	_			CALL SEM (NT+A1+A+H+UO)	PPT	032
	C C			· · · · READ CULFFICIENT FUR UI.	PPT	033 034
	č		•	a a a wend connected non-art	PPT	035
				KEAD (U1.114) NT.A1	PPT	036
				READ (U1+120) (A(I)+P(I)+I=1+NT) WRITE (U2+123)	PPT	037
				WRITE $(U2 \circ 121)$ NT $\circ A1 \circ (A(1) \circ H(1) \circ I = 1 \circ NT)$	PPT	039
				CALL SEM (NTOAloAoHoUI)	PPT	040
	C C			CALCULATION OF BROWARD TIN JOBY RESCARED BY HET DAYN	PPT	041
	c	۰	•	• • • CALCULATION OF PROHABILITY (DRY PRECEDED MY WET DAY) FROM PROBABILITY (DRY DAY) AND PROBABILITY (DRY PRECEDED BY	PPT	042
	C C			DRY) WHICH ARE DERIVED FROM DATA.	PPT	944
1	С			$\Delta 1 (1) = (01 (1) = (01 (2) (0) (0) (0) (1) (1) (1) = (01 (2) (2))$	PPT	045
				Q1(1) = (U1(1) - U1(3(5))) + U1(1) - U1(3(5)) D0 101 N=2,3(5)	PPT	046
		10	1	$Q_1(K) = (Q_1(K) - Q_1(K-1) \circ Q_0(K)) / (1 - Q_1(K-1))$	PPT	048
	~			DO 102 M=1,365	PPT	049
	C C			DATA DRY/DRY CONVERT TO WET/DRY.	PPT	050
	C		-		PPT	052
	~			$Q_0(K) = 1_0 - Q_0(K)$	PPT	053
	C C			• • • • DATA DHY/WET CONVERT TO WET/WEI.	PPT	054
	č				PPT	056
				$Q_1(\kappa) = 1, -Q_1(\kappa)$	PPT	057
	С	10	50	CONTINUE	PPT	05A 059
	č			• • • • PEAD COEFFICIENT FOR RAINFALL DISTRIBUTION.	PPT	060
(C				PPT	061
				READ (U1+119) NT+A(READ (U1+120) (A(I)+B(I)+1=1+NT)	PPT	062
				WRITE (U2.124)	PPT	064
				WRITE (U2+121) NT+A1+(A(I)+B(I)+I=1+NT)	PPT	065
				CALL SEK (NT+A)+A+H+XLAM) P=00(1)	PPT	066
		10)3	1F (MODP.NE.2) GO TO 104	PPT	068
	C				PPT	069
	C C	۰	٠	 READ IN PARAMETERS FOR PRECIPITATION CURVE. THE WAVE- LENGIH IS CONSIDERED TO BE 365 DAYS (.0172 = I*PI/365.) 	PPT	070
	č			and a second provide the second	PPT	072
				READ (U1+115) H1+H2+H3	PPT	073
		10)4	WRITE (U2:116) H1:H2:H3 IF (IDAY:E0:0) (U TO 114	PPT PPT	074
		- 1	1	IF (MOUP-2) 106+105+107	PPT	075
	C C			APARDATE AND ADDATALLIAN ACTUAL COMPANY	PPT	077
	C	۰	•	• • • GENERATE DAILY PRECIPITATION USING A SINE FUNCTION.	PPT PPT	078 079
	~	10)5	DDAY=IUAY	PPT	080
				PPT=H1*(SIN((DDAY-H2)*.0172)+1.)+H3	PPT	081
				PPT=PPT/365. IF (PPT.LE.U.) PPT=0.	PPT	082
				GO TO 113	PPT	084
	С	10	6	RETURN	PPT	085
- (C			• • • • GENERATE DAILY PRECIPITATION STOCHASTICALLY USING A	PPT	086
(С	-		MARKUV CHAIN FOR DETERMINING WHEN & STORM UCCIWS AND AN	PPT	087
	Ĉ			EXPONENTIALLY DISTRIBUTED EVENT SIZE GIVEN & STORM DID OCCUR.	PPT	089
1	•	10	7	REPARF (U)	PPT	090
				IF (H-P) 108+108+109	PPT	092
		10	A (NRA=2 Hafanf (U)	PPT	093
				PPT==ALUG(R)=XLAV(I()AY)	PPT	094
			-	GU TO 110	PPT	096
		10	9	NHA=1 PPT=0.	PPT	097
-	C				PPT	09A
					10. Con 15	

C	• •	• • • • ESTABLISH NEW CONDITIONAL PROBABILITIES.	PPT	100
C			PPT	101
	110	GO TO (111,112), NHA	PPT	105
	111	P=Q0(IDAY)	PPT	103
		GO TO 113	PPT	104
	112	P=01(JUAY)	PPT	105
С			PPT	106
C		• • • • CUNVERT INCHES PRECIPITATION TO MM.	PPT	107
C			PPT	108
	113	PP1=PP1=25.4	PPT	109
		RETURN	PPT	110
	114	₽₽1=0.	PPT	111
		KE TURN	PPT	112
C			PPT	113
С		• • • • FOPMATS USED IN THIS SUBRDUTINE.	PPT	114
С			PPT	115
	115	FORMAT (3F10.0)	PPT	116
	116	FORMAT (27HOPRECIP SINE FCN CUEF H1 = +10.4+5X+5HH2 = +F10.4+5X+	SPPT	117
		1HH3 = +F10.4)	PPT	118
	117	FORMAT (2A4+A2)	PPT	119
	118	FORMAT (1H0+204+02)	PPT	120
	119	FURMAT (13,F10.0)	PPT	121
	150	FURMAT (12F6.4)	PPT	122
	121	FORMAT (1H +5HNT = +12+2X+5HA1 = +F10,4+2X+18H(A(I)+B(I)+I=1+NT)+	IPPT	123
		$10(F_{6,4,4,2})$	PPT	124
	122	FORMAT (HHADA COFF)	PPT	125
	123	FORMAT (PHNG) COFF)	PPT	126
	124	FORMAT (11HOLAHDA CDEF)	PPT	127
C			PPT	128
		END	PPT	129
		Me active sa		4 44 1

SUBROUTINE RECRD (TIME+ISIRT+]DAT+INTP+IP+P) RF CRD001 RECHDOOS С С . . THIS SUPROUTINE (RECRD) IS CALLED BY THE SUPROUTINE RECEDO03 (CYCLI) TO READ WEATHER DATA RECORDS FROM A TAPE. C RECHDOO4 č RECHDOOS INTEGER IDAY+INYR+JUBAY+KYR+LU+MISS+U1+U2+U3 RECHOOOS REAL POPPOPIOSITESTIMESTMANOMINOPOTSIRTOTICS RECHDOOT С PECHDOOA COMMUN /XHINT/ LU.U1.02.013 RECROOOS PECHDUJU С CHECK FOR FOULLITY OF TIME AND ISTRI-PECPDOIL С С RECRO015 1F (TIME.GT.TSTRT) GO TO 101 RECPD013 C RECED014 С . VALUES PI. TI. TZ ARE INITIALIZED TO FORM A TABLE IN RECROO15 . WHICH THE VALUES OF THE PREVIOUS -GUOD DATA- ANE STORED. С RECHO016 С RECPD017 $P_{1=0}$ RECRO018 RECPD019 TI=0. 12=0. RECRD020 M155=0 RECRONSI RECHD022 RECHD023 RECKD024 RECRD025 С FIND REGINNING OF SIMULATION RUN. RECEDUS4 С RECHD027 102 1F (KYR+LT. INYR) GO TO 101 RECEDO28 103 MISS=0 RECHD029 IF (IDAY.E0.0) 60 TO 106 IF (IDAY.E0.1.400.JUDAY.E0.366) 60 TO 101 RECPD030 RECRD031 1F (1NT(TIME).EU.INT(TSTRT+1.)) WRITE (U2.110) KYP.JUDAY.IDAY RECRD032 С RECRDA33 CC DATA RECORD FOUND FOR IDAY BY THREE VETTONS# RECHD034 • • • DATA PECOND FOUND FOR IDAY BY INREF OFTIDAS# JUDAY •II• IDAY THEN NEW DATA CAPD IS READW JUDAY •GT• IDAY THEN NEW DATA CAPD IS READW JUDAY •GT• IDAY THEN DETERMINE IF ANY MISSING INFORMATION ON THE RECORDS. TI•)?• AND P] CONTAIN EITHEP CURRENT DAY WEATHER INFORMATION OF PAST GOOD DATA. IF ACTUAL DATA FOR THE DAY IS MISSING. THE WEATHER DATA FROM THE PMFVIDUS =GOOD DATA- DAY IS USED. RECPN035 C C PECP0036 RECR0037 С RECROAJA RECPD039 CCC RECRO040 RECPD041 С PECPD042 IF (JUDAY-TUAY) 101+105+104 RECPD043 104 M155=1 GU TU 106 RECPD044 RECRD045 105 IF (INF(TMAX).EQ.-0) TMAX=T1 RECED046 1 F (INT(TM1N).EQ.-0) TM1N=T2 RECPD047 IF (PP+LT+0+) PP=P1 RECHD048 C RECRD049 č RESET DATA TABLE. RECED050 С RECRD051 T1=TMAA RFCRD052 T2=IMIN RECPD053 P1=PH RECPD054 106 TP=(11+12)/2. RECRDASS С RECPONS6 . . CONVERSION FROM FARKEDHEIT TO CENTIGRADE OF TP AND С RECPD057 . č CUNVERSION FROM INCHES TO HM OF P. RECPD058 č RECEDA59 TP=(TP-32.)*.55555 RECPOORO P=P1 RECRD061 P=P+25.4 RECPDOE2 RETURN RECPD063 107 WHITE (U2.109) KYR.JUDAY.IUAY RECKD064 RETURN RECRD065 C RECPDO66 С • • • • • • FOPMATS USED IN THIS SUPROUTINE. RECRD067 C RECPD068 108 FORMAT (A4.9X.12.13.1X.F4.0.1X.F4.0.1X.F5.2) RECPD069 109 FORMAT (1x.317) RECRD070 110 FORMAT (10H KYH = +17+5X+8HJUDAY = +17+5X+7HIDAY = +17) RECPO071 С RECRD072 END RECRD073

SULKOUTINE RUMEN (LVWT+SUPN+SUPN+SUPN+GAN+GAN+MSZ+ROM+REL+RDMD+MTPRHUMEN001 RUME NO02 10.VL/5) С HUME NOO 3 . . THIS SUPROUTINE (HUMEN) IS CALLED BY THE SUPROUTINE Ċ RUME NO04 . . С CAFLES) TO DETERMINE FORAGE INTAKE AND ANIMAL GAIN AS A PUME NO 05 FUNCTION OF NETHOGEN CONTENT IN THE FORAGE. C PLIME NOOG C RUME NO 07 RUMENOOR REAL ALMOAFACOASZOAVAILOCAPODOGODIGODMPODMOUNDM HEAL DSOFFACOEXHOLYHOFRDROGOGANOGFACOGNNOGPLUSDOTG RUMENO09 REAL LUNTOMENOMIPHOONIPHONOPASOMSTONDILTONFUNONFROR RUME NO10 REAL DOUMNOPASEDOPENORDNOFUNDORELOSGOSUPLOSUPN HUMENO11 REAL UNDED. VEA. VL 75 NUME NO 12 С RUMEN013 COMMUN /COW/ AVAIL .FYH.G.GPLUSD.MPAS.NEOR.NEKUR.O RUME NO14 C RUME NO 15 COMPUTATION OF INCESTION HASED ON THE AMOUNT AVAILABLE. AND THE DIGESTION FACTORS. č PUMEN016 С RUME NO17 Ċ RUMENOIN CAP=.16"VL75 1G=CAP=*FL RUMEN019 RUMEN020 1F (1G.LT.n.) 1G=0. RUMEN021 IG=AVAIL*16 RUMF NO 22 D=00.45+60.7 RUMEN023 SG=SUPL *LVWT*.001 RUME NO 24 ()S=0. RIJME NO 25 IF (SG+1G) 102+102+101 RUME NO 26 101 D5=56/(56+16) D=D*(1+-D5)++75*05 RUME NO 27 RUNFN02A 102 NOIE I=NFOR*(1.-05)+SUPN*05 RUME NO29 DG=U* (16+56) RUMEN030 DUM=DG#4300. RUME NO 31 IF (NDILT.LT.0.) AFAC=0. RUME NO32 (NDIET.LI...6.AND.NDIET.GE.D.) AFAC=.031*NULET IF RUMENO33 (ND1LT.GL...6. AND.ND1ET.LT.1.8) AFAC=.023* (ND1FT-.61+.0186 TF RUME NO 34 IF (NDILT.GL.1.H) AFAC=.0465 DMP=DOM®AFAC RUME NO 35 PUME NO 36 MIPHO=MIPRO+DMP-MPAS+MIPRO RUMEN037 MIPHOW=MIPHO/VL75 RUMENO3A **FRDRENFRORMIPPOW** RUME NO 39 FIRSFRUKOFYR RUME NO40 IF (EXH.GT.. 95) EXHE.45 RUMEN041 DIG=RDMU+DG RUMEN042 UNDFD=IG+(1.-D)+ROM-HDMD PASFD=(1.0-FRDR)+EYR+DG+EYR+HNDFD RUMF N043 RUME NO44 C RUMENOAS Ċ COMPUTATIONS MADE FOR THE RUMEN FACTURS. RUME NO46 RUME NO47 DRDM=1G-EXR+DIG-PASED RUMF NA4A RDM=HUM+DHDM+SUPL+LVWT+.001 PLIME NOAQ IF (HDM.LT.O.) HDMEO. RUME NOSO RDMDEFAHAROM RUMEN051 IF (HDMU+LT+0+) PDMD=0+ RFL=RDM+MIPHU+.001 RUMENOSZ RUME NOS3 CCC RUMEN054 COMPUTATIONS MADE FOR ALL THE ENERGY RELATED VARIABLES. PUME NO55 RUME NOS6 EFAC=. 747-. 15* (ND1FT-.6) RUME NO57 IF (NDILT.LT.A.G. EFAC=.747 IF (NDILT.G. 1.4) EFAC=.541 VFA=FRUM@DOM@EFAC RUMENOSA RUME NO 59 RUME NO60 PEN=. 25*PASFU#4300. RUMEN061 HENE . 65"MIPHOP4. JOHAS RUME NO62 AENEVIA+PEN+MEN-125.+VL75 RUME NO63 AS2=LV#1/MS2 RUME NO64 IF (AS2+LT++05) GFAC=4+ IF (AS2+GF++05+AND+AS2+LT+1+0) GFAC=3+74+5+26*AS2 RUME NO 65 RUMENOAA IF (A54.GF.1.0) GFAC=9. RUMFN067 GANBAL W/GFAC PUME NO68 OMNEU-10+100. RUME NO69 GNN=G+10+1000. RUMEN070 RETURN RUME NO 71 C RUME NO72 END RUMF NO73 SFH SUBMOLITINE SER (NT.A) . A.R. S) 001 C SER 002 CCCC THIS SUPPOUTINE ISEN) IS CALLED BY THE FUNCTION (PPT) SFR 003 TO UDTAIN DAILY PARAMELER VALUES FOR EXPANSION OF FOURIER SER 004 SFH SEFILS. 005 SFR 006 INTEGER INNINT SFR 007 HEAL ALLOD +A1+H(10) +5(365)+50M+X1+XN SFR 00A C SER 009 00 102 1=1.365 SFR 010 ALS+LOAT(1) SFR 011 SUMEU. SER 012 00 101 N=1.NT SFR 013 ANEFLUAT (N) SFR 014 SUM=500++A(N)+CO5(+01725+X1+XN)+P(N)+SIN(+01725+X1+XN) 101 SFR 015 S(I)=A1+SIM 102 SER 016 HETUKN SER 017 C SER 018 END SFR 019

77 STAPTOO1 SUNPOULINE START THIS SUBJUCTIVE (START) IS CALLED BY THE MAIN PROGRAM. (RAGOES) TO READ USEN DEFILED PETTAL VALUES ON TO INTELATIZE STAR ATTON CONTROL VARIABLES. TNTEGER TOTIAYOTASTANTASTAYNOTAROJAHOJOJUNAYOKOLUOMOMKT INTEGER MORP , MONT , MON , NCLAS , NUED , 117 , 112 , 13 , XNF HEAL ANUS .ANTOT . AHEA. HASE . CASH. CH. DIT. DT. DT.L. DTPL HEAL DIFNOFVAPOFCAPOFCUNSONATING GAINOLVWTOLWTOMSZ HEAL OHINOPHI . PHOENOPHCOPHCOVOPHILE OPHMNOPTOPWONG REAL SUSSOILA, SUTER, STAND, SUPE, SUPA, SUPP, SUPPA, SUPPA HEAL SUPWENT TOTENDATION DATIMENTIMITATOTAL TPAINATSTRE MFAL VOVALIN OWILLOGATIONALITO.77 CUMMON /CAT/ ANUS(S),ANIUI.APEA.0111.FCUMS.GAIN(S).GNIN(S).IN(S).TSTARTOL6 INFU(5) +LVWT(5) +L #T(5) +MAT(5) +MS7(5) +MCLAS+MOPU(5) +OMIN(5) +STAND(5) STAPT017 2.50PL(5).50PN(5).50PPP(5).50PR1(5) COMMON VECONV HASE (5) . CASH (5) . PRCOV (144) . PHICE (12) . PRMN (12) . SUPP (SSTARTO19 1) • SUPWT (5) • VALUE (5) • 77 (12) COMMON 2102 DIFL • 01PL • 01PP • 1END START020 COMMON ZEAZ MODE + CAP+SOILA-SOILA-THULD+WILL COMMON ZEAZ MODE + CAP+SOILA-SOILA-THULD+WILL COMMON ZEAZ MODE + CAP+SOILA-SOILA-THULD+WILL COMMON ZEAZ MODE + CARADON COMMON ZEAZ DI+TIME+TSIR1+2(94)+20F221002110 COMMON /UX/ CH.GN.PH.PHC.PHC.PHC.FW.RG.SD.TUTAL.V COMMON /XUNI/ LU-U1-U/+U3 SET ALL NON-PEAD ARRAYS TO ZEPU. DO 102 1=1.12 Fk1Ct(1)=0.0 1F (1.GT.6) GU TO 101 TP(1)=0.0 1F (1.6T.5) GO TO 102 ANO5(1)=0.0 SUPP(1)=0.0 CASH(1)=0.0 GAIM(1)=0.0 BASE (1)=0.0 GN1N(1)=0.0 LWT(1)=().() 0.0=([) VIIMO SUPL(1)=0.0 VALUE (1) = 0.0 SUPw1(1)=0.0 102 CONTINUE READ DATA FROM DATA SECTION. READ (UI.105) ISTRT.TEND.UT.DTPP.DTFL.UTPL READ (U1+105) X(10)+X(11)+X(12)+X(2)) READ (U1.107) NCLAS.MUDT.MUDP.INYH.ARFA.FCAP.THOLD.VILT.SOILA.SOILSTARTA53 18.TINIT 1F (NCLAS.LE.0) 60 TO 103 READ (UI+10B) (IN(J)+MKT(J)+TNFD(J)+NOFD(J)+LVWT(J)+SUPRT(J)+SUPPRSTART056 1 (J) + STKNU (J) + SUPN (J) + M52 (J) + J=1 + NCL AS) READ (UI+106) (PRCOV(I)+1=1+)44) READ (U1+106) (PRNN(1)+1=1+)2) . . AREA IS INPUT IN ACRES AND CONVERTED TO SQUAPE METERS THEN THE RANDOM NUMBER GENERATOR IS INITIALIZED.

C C С 103 AREA=AHLA+4047. IF (ARLA.LF.0.0) AREA=1.0 CALL RANSET (TINIT) C С SET OEFAULT VALUES. С IF (TENU.LF.0.) TEND=365. IF (DT.LF.N.) DT=1. IF (NCLAS.GT.5) NCLAS=5 IF INCLAS.LE.0) RETURN DO 104 J=1.NCLAS C C • • • • • • ADJUST VALUE OF 1N(J) AND MKT(J). C M=(IN(J)-TSTRT)/DT++5 IN(J)=ISTHT+MOOT M=(MAT(J)=TSTPT)/DT+.5 MKT(J)=TSIRI+MO()T С С LVWT IS CONVERTED TO GRAMS AT THIS PUINT. M57(J)=M57(J)+454. LVW1(J)=LVWT(J)*454.0 IF (S1KN0(J).LF.0.0) LVWT(J)=0.0 104 CUNTINUE RETURN С C FORMATS USED IN THIS SUBROUTINE. C 105 FORMAT (HEID.0) 106 FORMAT ((7x+6(F10+4+1x))) 107 FURMAT (311+12+5X+7F10+0)

108 FURMAT (413+8X+6F10.0)

С

END

STAFT007 STANTODE

STANTI ...

STALTOOS STANTONA

STANTOOT

STAFTOOR

STANTONS

STANT010

STANTOIL

STARTC12

STAFT013

STANTOIA

STARTO15

STAFTCIA

STAPT021

START022 STAPT023 STARI024 START025

STAPT026

START027

STAPT028

STAP1029

STAPT030

STAHT031

START032

STAPT033

STARTC34

START035

STAPT036

START037

STAPTO38

STAPT039

STAPT040

START041

STAFT042

STAFT043

START044

START045

STANTOAS

STAFT047

STAPT048

START049

START050

START051

STAFT052

START054

STAPIDS5

START057

STARTOSA START059 START060

STAFT061

START062

START063

STAPT064

STAPT065 START066

STARTO67

START068

START069

STAPT070

START071 STAPI072

STAPT073

STARI074

STAPT075

START076

START077

START078

STAPT079

STARTORO

STAPTORI

STARIO82

STARTOR3 START084

STARTOH5

STARTORS

STARTOR7

STANTORR

STARTORS

START090

STAPTORI

STAPT092

STAPT093

START094

STAPTOPS

STAPT096

STAUT097

STARTOPA

C

С С

C

C

C

C

č

С

С

C

HEAL FUNCTION TEMP (IDAY . MOUT . NINA . FIME . ISTNT) TEMP 001 CCCC TEMP 002 • • • THIS FUNCTION (TEMP) IS CALLED BY THE SUPPORTINE (CYCL) TO CALCULATE TEMPERATURE. A TEPPEPATURE SINE FUNCTION CAN BE EAST(Y DENIVED FOR A LOCATION IF THE AVERAGE YEARLY TEMPEMATURE REGIME FOR A NEARBY LOCATION IS KNOWN, NRA IS A VANIARLE ALL(WING TEMPERATURE TO BE REDUCED WHEN RAIN UCCURS. IF SOME REGIME OTHER THAN SINE IS USED. TEMP 003 TEMP 004 TEMP 005 č TEMP 006 CCC TEMP 007 TEMP COR TEMP 009 TEMP 010 INTEGER IDAY+LU+MUNT+NHA+U1+12+13 HEAL DUAY . HA . HS . HA . TIME . TSINT TEMP 011 TEMP 012 C TEMP 013 COMMON /XUNT/ LU.U1.U2.U3 С TEMP 014 IF (MODT-2) 101+102+102 TEMP 015 101 HETURN TEMP 016 TEMP 017 CCC CHECK FOR FOUALITY OF 11ME AND TSTPT. TEMP 018 TEMP 019 102 IF (TIME.GT.TSTHT) GO TO 163 TEMP 020 С TEMP 021 • • FAU IN PARAMETERS FOR TEMPERATURE CURVE. The wavelength is cunsidered to be 345 have (.0172 = CCCC TEMP 022 . TEMP 023 (2+1/365.). TEMP 024 TEMP 025 HEAD (UI+165) H4+H5+H6 WHITE (U2+106) H4+H5+H6 103 IF (IUHY+E0+0) G0 T0 164 TEMP 026 TEMP 027 TEMP 02A DOAY=IDAY TEMP 029 TEMP=H4* (SIN((DDAY=H5)*+0172)+1,)+H6 TEMP 030 TEMP=(1LMP-32.)*.55555 TEMP 031 RETURN TEMP 032 104 TEMP=0. TEMP 033 HETUKN TEMP 034 С TEMP 035 C FORMATS USED IN THIS SUPHOUTINE. TEMP 036 **TEMP 037** 105 FORMAT (3F10.0) TEMP 039 106 FURMAT (25HOTEMP SINE FCN COFF H1 = .F10.4.54.5HH5 = .F10.4.5X.5HHTEMP 039 16 = +F1V.4) TEMP 040 С TEMP 041 END TEMP 042

с			SUMMULTINE AFLWS	XFLWSO
č	•		THIS SUPROUTINE (XELWS) IS CALLED BY THE MAIN PROGRAM	XFL WSO
С			(WARGES) TO COMPUTE SHE VALUES OF THE FLOWS AND STORE THE	KEL WSO
C C			VALUES IN THE COMPUTED FLOW STACK.	XFLWS0 XFLWS0
			INTEGEN INTOXYNIMENINA INFENINYRNIYRN IYRN HUUAYNK OMNT	XFLWSO
			INTEGER MODP+MODIT+MON+NCLAS+NOFD+XFF+XN+XNF+ANLOC	XFLWSO
			ΙΝΤΕΟΕΝ ΙΧΝΡΕΙ «ΧΝΡΕΙ «ΧΝΡΗ «ΧΝΕΙ» «ΧΝΥΚS «ΧΡΙ «ΧΗΡ«ΧΟΕ ΚΕΔΕΙ ΙΔΝΟS«ΛΝΙΟΙ» ΔΗ Δ.«ΝΥΔΙΕ» ««ΗΔΕΕ» COSH«CH«CP» υΠΔΥ	XFLWS0 XFLWS0
			REAL DELT. DMX. OFF. OPID. DI. DIFL. OTFL. OTFR. FTOUL. ETODW	XELWSO
			PEAL EVAPORTHORCAPORCONSOFLOWOLOUNTHORCOGNO GNIN	XFLWSO
			HEAL GPLUSDOGHOGOOLVATOLWTON PPROGONOXOPPTCFOPMONOPT	XFLWSO
			KEAL PW+RDM(5)+RDMD(5)+REL(5)+RG+SD+SO1LA+SO1LH+STKNO	XFLWSO
			REAL SUPLISSIPPISSIPPISSUPPISSUPPISSUPPITS ISTENDS THUND	XELWSO
			REAL TIME TOTAL THAT HAS INTO A VOLUFAVI / SAWTLAWPAX	XFLWSO
С			WEAL XF #X1010#X1011#X1110#22	XFLWSO
			CUMMUN /CAT/ ANUS(5) + ANIUI + AHEA + DITT+FLUNS+GAIN(5) + GNIN(5) + IN(5) +	
			NED(5) +LVWT(5) +LWT(5) +MKT(5) +MKT(5) +NCLAS+NOFU(5) +OMIN(5) +STKNO(5)	
			2950PL(5)+S0PN(5)+S0PPR(5)+S0PPT(5) COMMON_ZCOWZ_AVATL+FYR+0+6PLUSD+MPAS+NEOR+NEKUR+0	XFLWS0 XFLWS0
			COMMON /FCON/ HASE (5) + CASH (5) + PRCOV (144) + PPILE (17) + PRMN (12) + SUPP (
		1) + 50Pw1 (5) + VALUF (5) + 77 (12)	XFLWSO
			COMMON 2102 DTEL +DTPL +DTPE +TEND COMMON 2PENIZ EVAP+FCAP+SOILA+SOILA+THULD+VILT	XFLWSO
			COMMON /IAP/ MODP.MODI. + PFL + PT+1 + IP(6) + IN	XFLWSO
			COMMUN /TT/ 10AY+INYH+1YH+JUDAY+K+MON	XFLWS
			COMMON /TX/ DISTIMESTSINISX(94)SANESXIUTOSXIIIO COMMON /UX/ CHSGNSPHSPHSPHCPPHSPGSDSTUTALSV	XFLWSA
			COMMON /XSYS/ XF (300) + XF (300) + XNLOC (20+5) + XNPL + XNPL T+ XNPR + XNST + XI	
		1	VRS(2C) + XPL (99) + XPP (94) + XST (99)	XFLWSO
C			×N≈1	XFLWS0 XFLWS0
с			AN • I	XFLWSO
Ç	•		• • • • PROCESS = PRECIPITATION.	XFLWSO
С			FLOW-DT	XFLWSO
			FLOW=PT 1F (FLOW=LT=0.) FLOW=0.	XFLWS0 XFLWS0
			PFL=PFL+FLOW+DT	XFLWSP
			XF (XN)=+LOW	XFLN50
с			XN=XN +]	XFLW50 XFLW50
č			• • • • PROCESS = EVAPUIRANSPIRATION.	XELWSO
С				XELWSO
			EVAP=E1(X(21) +TK+FCAP+THOLD+WILT) FLOW=FVAP	XFLWSO
			IF (FLUW.LT.O.) FLOW=0.	XFLWS0
			XF (XN) =FLOW	XFL,WS0
С			XN=XN+1	XELWSO
č			· · · · PROCESS = NET ACCUMULATION OF GREEN FURAGE.	XFLWSO
С			CALCULATION OF PLANT USE OF ATHOSPHENIC CV2 INCLUDING BOTH	XFLWSO
C C			GPOWIH AND LOSS TERMS WHICH MAYPE IESS THAN ZERO.	XFLWSO
•			Gw=0.	XFLWS0 XFLWS0
			GC=0.	XFLWS
С			$X1011 = \lambda(10) + \lambda(11)$	XFLWSO
č			WHEN DAILY TEMPERATURES ARE BELOW TO DEGREES CENTIGRADE.	XFLWS0
C			THERE IS NO GROWTH OF FORAGE.	XELWS
С				XFLWSO
С			IF (T.LT.5.) GO TO 104	XFLWS
C	•		ONE GRAM GROWTH INITIATION FOR COOL SEASON IF TEMP .GE.	XFLWSO
C			5. MARM SEASON IF TEMP .GF. 10.	XFLASO
C			IF (T.GE.5AN().X(II).LE00001) X(II)=I.	XFLWS
			$IF = (T_{0}G_{1}, 10, 0, 0, 0), X(10), UE = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, $	XELWSO
			CALL GRUW (X(10) +X(11) +X(21) +TR+SOILA+SOILA+SOILR+GH+GC+FCAP+WILT+K)	XFLWSO
С			K=K+1	XFLWS0
č			• • • • ONLY CHECK GROWTH IF AT LEAST UNE COMPONENT HAD NET	XFLWSO
C			GROWTH OVER THE LAST DT.	XFLWSO
C			IF (GW+LT+0++AND+GC+LT+0+) GO TO 104	XFLWSO
			EVAP=EVAP=1000.	XFLWSO
			GH=0.	XFLWSO
			DDAY=1DAY+40 A=550.*200.*514((DDAy)/365.*6.28)	XFLWSO
			IF (6w+6T_0_+AND+6C_6T_0) 60 TO 103	XFLWS0 XFLWS0
			1F (Gw) 101+101+102	XFLWSO
C			DECOMPLIES HADM AND CODE DESERVE COMPLEX SERVICES	XFLWSO
C	•	•	 PECOMPUTE WARM AND COOL SEASON GROWTH ASSUMING GROWTH WAS MEDUCED. 	XFLWSO
č				XFLWSO
	10	1	1F (GC®B.GT.F.VAP) GC=FVAP/B	XELWSO
			GR=GC GO TO 104	XFLWSO
	10	2		XFLWS0
			GH=Gs	XEL WSO
		2	GO TO 104 GH#Gh+GC	XFLWS0 XFLWS0
	10			
	10	3		AT CHOIL

E

```
С
                                                                                     RELUSION
č
  . . . . . . CUMPUTE PERCENT OF TOTAL WARM AND COUL SEASON PLANTS
                                                                                     XEL #50.96
С
          COMPRESE.
                                                                                     XEL SCOT
                                                                                     AFTASAGA
С
       #Palis/IN
                                                                                     XFL $5099
       CHEGC/GH
                                                                                     XFL#SION
С
                                                                                     XFL $ 5101
С
              . IF AMOUNT OF WATER EVAPOTRANSPIRED IS LESS THAN AMOUNT
  . . . .
                                                                                     XFLWS102
С
          USED TO PHODUCE FLOW OF HIGMASS. REDUCE FLOW TO MAXIMUM
                                                                                     XFL#5103
č
           PUSSIALF.
                                                                                     XFL+5104
C
                                                                                     XFL #5105
       IF (GHOD.GT.FVAP) GHEFVAP/H
                                                                                     XFLWS104
       GWERHAUK
                                                                                     XFL #5107
       GC=LPOGH
                                                                                     XFL#S10A
С
                                                                                     XFL#5109
C
C
  XFLWS110
                                                                                     AFL#S111
  104 FLUW=(.#
                                                                                     XELWS112
       XF (AN) =FLOW
                                                                                     XFLWS113
       AN=AN+1
                                                                                     XFLWS114
       FLU#=GC
                                                                                     XFLWS115
       XF (XN)=FLOW
                                                                                     XFLVS116
       XN=AN+]
                                                                                     XFLWS117
       R(=0.
                                                                                     XFLWS11A
       IF (A1011+LT++00001) GO TO 105
                                                                                     XFLWS119
       FG=6F/X1011
                                                                                     XFL #5120
  105 CALL NITHO (RG+GN+PH+PH+CH+V+PH)
                                                                                     XFLWS121
С
                                                                                     XFLWS127
C
  . . . . . PHOCESS = PLANT SENESCENCE. DETING. AND FREEZING.
                                                                                     XELWS123
C
                                                                                     XFLWS124
       DMA=.05
                                                                                     XFLWS125
       ETODw=0.
                                                                                     XFLWS126
С
                                                                                     XFLWS127
      • • • • IF K IS LESS THAN 45 OR THE GROWTH RATE IS LESS THAN AT
THE PREVIOUS TIME STEP THEN NO DEATH.
C
C
  . .
                                                                                     XFLWS12A
                                                                                     XFLWS129
C
                                                                                     XFLWS130
       IF (K.L1.45.0R.X(10).GT.X10T0) GO TO 106
                                                                                     XFLWS131
c
                                                                                     XFLWS137
  . . . . . WARM SEASON DRYING.
                                                                                     XFLWSI33
С
                                                                                     XFLWS134
       ETUDw=+1
                                                                                     XFLVS135
       IF (TR.GT.25.) ETOD#=.10*(TH-25.)+.10
                                                                                     XFLNS136
        1F (TH+LT+10+) ETCOW=1+1+++1+14
                                                                                     XFLVS137
C
                                                                                     XFLWS13A
C
  . . . . . SECOND TERM REPRESENTS TRAMPLING LOSS.
                                                                                     XFLWS139
C
                                                                                     XFLWS140
  106 FLOW=EIUDWOX(10) ONMX+.0004+X(10) CANTOT
                                                                                     XFLWS141
       IF (FLUA.LT.O.) FLOW=0.
XF(XN)=FLOW
                                                                                     XFLVS142
                                                                                     XFLWS143
       XN=XN+1
                                                                                     XFLVS144
CCC
                                                                                     XELWS145
  . . . . . COOL SEASON DRYING.
                                                                                     XFLWS146
                                                                                     XFLVS147
       ETOUC=0.
                                                                                     XFLVS148
       IF (K.LI.45.0R.X(11).GT.X11T0) GO TO 107
                                                                                     XELWS149
       ET00C=.1
                                                                                     XFLWS1S0
       IF (TR.GT.20.1 ETODC=.10*(TR-20.)+.10
IF (TP.LT.5.) FTODC=1.1-.2*TR
                                                                                     XFLWS151
                                                                                     XFLWS152
  107 FLOW=EIUDCOA(11) 004X+.00040X(11) 0ANTOT
                                                                                     XEL WS153
       IF (FLUW.LT.O.) FLOWED.
                                                                                     XFLWS154
       XF (XN) =FLOW
                                                                                     XFLWS155
       XN=XN+1
                                                                                     XFLVS156
c
c
                                                                                     XELVS157
          • • • PROCESS = DISAPPEAHANCE OF STANDING DEAD FORAGE.
DISAPPEARANCE OF DEAD FURAGE IS CONTROLLED AS A FUNCTION
OF THE MEAN DAILY TEMPERATURE. SOIL MOISTURE AND TRAMPLING
  . . . .
                                                                                     XFLWS158
CCCC
                                                                                     XFLWS159
                                                                                     XFLWS160
          BY LIVESTOCK.
                                                                                     XELWS161
C
                                                                                     XFLWS167
       DRT0=.004+.00032*TR
                                                                                     XFLWS163
       IF (IR.LT.0.) DRTD=.004
DPF=1.0*(1./(FCAP=WILT))*(X(21)=WILT)
                                                                                     XFLWS164
       DPF=1.0-(1.) (F CAP ) (DF=2.)

IF (X(21).c1.FCAP) (DF=2.)

IF (X(21).c1.wILT) DPF=1.

FLOW=DRID@DFF*X(12)*.0005*X(12)*ANTOT
                                                                                    XFLWS165
                                                                                     XFLWS166
                                                                                     XFLWS167
                                                                                    XFLWS16P
       IF (FLOW.LT.0.) FLOW=0.
                                                                                    XELWS169
       XF (XN) =+ LOW
                                                                                     XFLVS170
C
                                                                                    XFLWS171
CC
  • • • • • • LIVESTOCK LOOP.
                                                                                    XFLWS172
                                                                                    XFLWS173
       IF (NCLAS.LE.O) RETURN
                                                                                    XFLWS174
       IF (X1011.GT..00001) GO TO 108
                                                                                    XFLWS175
       WP=0.
                                                                                    XFLWS176
       CP=0.
                                                                                     XFLWS177
  GO TO 109
108 P=X(10)/X1011
                                                                                    XFLWS178
                                                                                    XFLWS179
       CP=1.-WP
                                                                                    XFLWS180
CCC
                                                                                    XFLWS181
  . . . . . COMPUTE TOTAL FORAGE PRESENT.
                                                                                    XFLWS182
  109 GPLUS()=X1011+X(12)
IF (GPLUSD) 110+110+111
110 NFUH=0.
                                                                                    XFLWS183
                                                                                    XFLWS184
                                                                                    XFLWS185
                                                                                    XFLWS186
       AVAIL=0.
                                                                                    XFLWS187
       GO TO 112
                                                                                    XFLWSIAA
C
                                                                                    XFLWS189
C . . . . . COMPUTE PERCENTAGE GREEN AND DRY COMPRISE.
```

XFLWS190

~				XFLB
С	111	110	\$•**10]{/(\$•**11]}**([?))	XFLB
		()#	1 e=(-	AP 1. 1
~		111	0K=(++0K+0+5)	XFL
C			· · · COMPUTE FURAGE AVAILABLE FOR CUNSUMP(10N.	XFL
č				XFL
			AIL = 1 + 0 - F XP (- + 167*(++ U50)	XFL
	112	00	117 L=IONCLAS XN=XNoJ	XFL
С			Viterus 1	XFLI
			• • • PENCESS = INTAKE OF GREEN FORAGE.	XFL
С			FLON=GHIN(1) #WF#STKNU(1)/AREA	XFLI
			$1F (FLOW_{\bullet}LT_{\bullet}O_{\bullet}) F1(W=0_{\bullet}$	XFL
			FLONS=FCONS+FLOW#OT	XFL
			XF(XN) = F(Ow)	XFL
			XN=XN+) FL()==(NTN(T)+CP+STKN()(T)/AKFA	XFL
			$1F (FLOW_{\bullet}LT_{\bullet}O_{\bullet}) FLOW=0_{\bullet}$	XFL
			FCONS+FCONS+FLOW+OT	XFL
			XF (XN) ≈FLOw	XFL
с			XN=Xtx+1	XFL
C			• • • PROCESS = INTAKE OF STANDING DEAD FORAGE.	XFL
С			EL AUTOURU (TARETZANIA) A ANTA	XFL
			FLOW=UMIN(T)PSTKNU(1)ZAREA IF (FLOW=UT=0=) FLOW=0=	XFL XFL
			FLONS=FCUNS+FLOW®DT	XFL
			AF (X++)=FLOW	XFL
с			XN=XN+1	XFL
č			PROCESS = INTAKE OF SUPPLEMENTAL FEED.	XFL
č				XFL
				XFL
С			FLOW=0.	XFL
C			DUPING SUPPLEMENTAL FELO PERIOD. WEIGHT GAIN DUE TO	XFL
C			SUPPLEMENTAL FEEDING IS DETERMINED BY FEED HATE AND BODY MASS.	
С			IF (1DAY.GF.TNED(T).AND.IDAY.LE.NOFU(I)) *LOW=SUPRT(T)*X(I)	XFL
С			IL CEDEFED, FINEDCED FRADELINE FEE FRADE CLEAR CONSCENT CLEAR FEE	XFL
С			DURING SUPPLEMENTAL FEED PEPIDU, CALCULATE PROPORTION OF	XFL
C			WEIGHT GAINED FROM SUPPLEMENTAL FFFUING TU DATF.	XFL
С			IF (IDAY.(F.INFD(I).AND.IDAY.LE.NOFU(I)) SUPL(I)=SUPRT(I)	XFL
			IF (+L()w.LT.0.) FL()w=0.	XFL
			SUPwi(1) = SUPwT(1) + FLOW + 1T + APFA/454.0	XFL
			XF(X(4)=FLOw XN=XN+1	XFL
С				XFL
С			• • • PROCESS = TOTAL OF ANIMAL LOSSES.	XFL
c			IF THE DAY IS OUTSIDE THE GRAZING PERIOD, INTAKE OF FORAGE	XFL
č			AND WEIGHT GAIN ARE SET EQUAL TO 0.	XFL
			IF (104Y.LT.IN(1).OR.104Y.GT.MKT(1)) GO TO 114	XFL
C C			TAITTEALTE CUMMATION MAUTADI CO	XFL
č		• •	• • • INITIALIZE SUMMATION VARIABLES•	XFL
-			IMF=1IMF+.5	XFL
			VL75=(.n01°LVwT(1))**.75	XFL
			IF (IMF.NE.IN(I)) GO TO 113 HDM(I)=.06#VL75	XFL
			KFL(1)=RijM(1)	XFL
			KDMU(1)=KDM(I)	XFL
~			MIPKU(I)=200.	XFL
			• • • INITIALIZE CONSTANIS.	XFL
				XFL
			MPAS=.52	XFL
			NFRUx=04 EYR=025	XFL
с				XFL
	•	•	• • • IF THERE ARE NU ANIMALS. INTAKE OF FURAGE AND WEIGHT	XFL
CC			GAIN ARE SET TO ZERO.	XFL
C	11:	3	IF (INT(STKNO(I)).GT.0) GO TO 115	XFL
	114		Om144(I)=0.0	XFL
			GAIN(1)=0.0 $GNIN(1)=0.0$	XFL
			GNIN(1)=0.0 GU TU 116	XFL
	119		CALL RUMEN (LVWT(I) .SUPN(I) .SUPL(I) .OMIN(I) .GNIN(I) .GAIN(I) .MS	
~		1	(1] • KOM(1) • KFL(1) • KOMD(1) • MIPKO(1) • VL 75)	XFL
C C			• • • ANIMAL LOSSES ARE THE DIFFERENCE RETWEEN TOTAL FOOD	XFL
č			INGESTED AND WEIGHT GAINED.	XFL
C				XFL
	110	5	FLOW= (GNIN(I) + OMIN(I) - GAIN(I)) #STKNU(I) / AKEA	XFL
			IF (FLOW→LT→O→) FLOW=O→ XF(XN)=FL()₩	XFL
	117	7 0.0	INTINUE CARLEND AND A COMPANY A COMPANY AND A	XFL
				XFL
			TURN	XFL
		EN	D	XFL
		6.11		

SUBRUELING AGEAPH CHELENVERNALUCERTINE VLINE +ZEVMINEVMAXENNENAPL) RGHPH001 XI-HE-H002 C . THIS SUBROUTLEE (ASHAPH) IS CALLED BY THE SUBROUTINE C C XGHEHOO3 (APLOEN) TO GENERATE AND OUTPUT ONE PRINTEN PLUT AT FACH KI-PHAGCA c XINHHOC'S CALL. С XGH HOOK INTEGER 1. THE NE. ICHE. TOASH. IF YEE . IP () OUT . TOUAL . ISTOP. 17 XIIRPHO07 INTEGEN JOURNAROUNTOUPONEYONNTOLOCOLUONNONPLONVA XURPHOOR INTEGER NX1 0C (20.5) .NXPL (14) .01. .112.113 **POOH449X** XGRPH010 REAL TIME + XLINE (11) + YLINE (6) + YMAX + YMIN + 2 (100) + 21 + 7 J + 77 С XGRPH011 COMMON /XFIG/ KFY(5)+1HLNK+1DASH+1FYFF+1QUAL+JCHAP(5) XGHPH012 CUMMON /XUNT/ LU+U1+U2+U3 XGRPH013 XGHPH014 С KNI=() XGHPH015 XGRPH016 JH1=0 10 101 1=1. NVR XGRPH017 J=NXLOC(NPL+1) XGRPHO18 101 1P(1)=MAPL(J) XINHHHUJO WHITE (02+110) NPL+(1+1P(1)+1=1+NVP) XGRPH020 XGHPH021 WHITE (U2.111) (YLINE(1).I=1.6) С XGRPH022 C • • FACH PASS THROUGH THE FOLLOWING EXPLICIT LOOP (STATEMENT 102-10 STATEMENT 108) GENERATES ONE LINE VE THE PRINTED PLOT XGRPH023 . . . XGRPH024 Ċ UN INF OUTPUT. XGHPH025 C READ IN ONE TIME STEP OF PLOTTING DATA. KGRPH026 С XGPPHG27 102 READ (U.J.) ISTOP+TIMF+(/(1)+1=1+NN) 1F (ISTOP+GE+1) GO TO 109 XGELHUZA KCHAHUSA С XGHPH030 С . . INITIALIZE THE OUTPUT CHARACTER STRING (TP) TO CONTAIN XGHPH031 C HLANNS AND GRAPHICAL REFERENCE LINES. XGRPH0.32 Ċ XGRPH033 KN[=KN]+] XGRPH034 ICHH=IHLNK XGRPH035 IF (MUD (FNT+10) .NE+0) GO TO 103 XGRPH036 1F (KN1+EQ.100) GO TO 103 XGRPH037 ICHH=IUASH XGRPH03A 103 DO 104 1=1.100 XGRPH039 XGRPH040 IP(1)=1CHR IF (MOD(1+20).NE.0) GO IO 104 XGRPH041 IF (1.FO.100) GO TO 104 XGRPH042 IP(1)=IEYEF XGRPH043 XGRPH044 104 CONTINUE С XGRPH045 CC . INSERT PLOTTING CHARACTERS INTO STRING WHICH REPRESENTS XGRPH046 . . THE PLOTTED VARIABLE. XGRPH047 C XGHPH048 DO 107 1=1.NVR XGRPH049 LOC=NALOC(NPL,1) XGRPH050 27=2(LUC) XGRPH051 c XGRPH052 . DETERMINE LOCATION OF PLOTTING CHARACIER IN STRING XGRPH053 C ACCURDING TO VALUE OF DEPENDENT VARIARLE (22) AND SCALING XGRFH054 C C PARAMETERS. XGRPHOSS XGRPH056 12=100 XGRPH057 С XGRPH05A С CHECK FOR THE EQUALITY OF YMAX AND YMIN. XGRPH059 C XGRPH060 IF (ARS(YMAX-YMIN).LT.1.0E-20) GO TU 105 XGRPHO61 Z1=1++99+*(ZZ-YMIN)/(YMAX-YMIN) XGRPH062 IZ=ZI XGRPH063 ZJ=12 XGRPH064 IF ((Z1-ZJ).GE.0.5) 1Z=1Z+1 IF (1Z.GT.100) IZ=100 XGRPH065 XGRPH066 IF (IZ.LT.1) IZ=1 XGHPH067 c XGRPH06A STORE PLOTTING CHARACTER IN STRING. XGRPH069 C XGRPH070 105 JP=1P(17) XGRPH071 ICHR=JCHAR(I) XGRPH072 IF (JP.FO.IHLNK) GO TO 106 XGRPH073 IF (JP.EQ. 10ASH) GO TO 106 XGRPH074 1 F (JP.EU. 1EYEF) GO TO 106 XGRPHN 75 IF (JP.FU.ICHR) GO TO 107 XGRPH076 ICHH=1QUAL XGRPH077 IP(12)=ICHR 106 XGRPH07A 107 CONTINUE XGRPH079 C XGRPHOAD C OUTPUT CHARACTER STRING. . . XGRPH081 C XGRPHOR2 1F (KNT+E0.1+0R+MOD(KNT+10)+E0.0) GO TU 108 XGRPHAR3 WRITE (U2.112) (IP(1).1=1.1001 GO TU 102 XGRPH084 XGRPHORS 108 JNT=JNT+I **XGRPHORA** WRITE (U2+113) XLINE(JNT)+(1P(1)+1=1+100) GO TO 102 XGRPH087 XGHPHOAA 109 WHITE 102.1141 *GRPHAN9 RETURN XGRPH090 C XGRPH091 C FORMATS USED IN THIS SUBROUTINE. XGRPH092 C XGRPH093 110 FORMAT (1H1+11H PLOT NO. +12//201+5(11+5H = 1(+12+1H)+4X)) XGRPH094 111 FORMAT (////13x+F12.1+5(8/+F12.1)/20x+1H++)RA+1H++4(19x+1H+)/20x+1XGRPH095 100(1HH)) XGRPHOSA 112 FORMAT (19X+IHH+I00A1+IHH) XGRPHA97 113 FORMAT (SA+F12+1+3H +H+100A1+1HH) XGRPH09A 114 FORMAT (20X+100(1HH)) XGRPH099 C KGRPH100 END IGRPH101

SUBNULTINE AUUT XOUT 001 LOUT 002 С * * THIS SUPROUTINE (XUUT) IS CALLED BY THE MAIN PROGRAM (KANGES) TO REAU THE DUTPUT CONTROL COMMANUS. FACH CARD INPUT IS SCANNED FOR A COMMAND VERR IN THE FIRST FOUR ¢ XOUT 003 C XOUT 004 C XOUT 005 cc COLUMNS IN ALPHANUMENIC MODE AND LEFT JUSTIFIED. XOUT DOG LIPON COLORIS IN ALPHANINEERIC MODE AND LEFT JOSTIFIEM, DOON RECONTION OF A COMMAND VERH THE PEMAINING COLUMNS ANE ACCEFTED IN INTEGER MODE. IN FIFEDS OF FIVE DIGITS AND RIGHT, JUSTIFIED, ON THE PLOT CAPD (HE FIRST NUMERIC FIELD (5 TO CONTAIN THE NUMER OF PLOTS (ANPLT) REQUESTED. XOUT 007 C XOUT DOA c X0UT 009 č XOUT DIO XOUT 011 C INTEGER TOTHLAKOTASHOLEYEEOINEOOIPOTAUALOTVERROTIOTZ X0UT 012 $\begin{array}{l} INTEGER \quad J_{0,J}CHAR_{0}K_{0}KEY_{0}L_{0}L_{0}UHIM(14)_{0}UI_{0$ XOUT OIS XOUT 016 INTEGEN XPR.XST XOUT 015 XOUT 016 HEAL NTOTIMEOTSTRTOKOKITO C XOUT 017 CUMMON /TX/ DT.TIMF.ISTHT.X(99).XNF.X1010.X1110 XOUT OIS COMMON /XF16/ KFY(5)+JHLVK+IDASH+IFYFF+IQUAL+JCHAR(5) XOUT DI9 COMMON /X5Y5/ XF(300)+XF+(300)+XNL0C(20+5)+XNPL+XNPLT+XNPK+XNST+XNXOUT D20 IVPS(20)+XPL(99)+XPF(99)+XST(99) XOUT 021 COMMON /XUN1/ LU+U1+U2+U3 XOUT 022 С XOUT 023 101 READ (01+123) IVERP+ (NUM(1)+1=1+14) 10UT 024 DO 102 1=1.5 IF (IVERH.EQ.KEY(1)) GO 30 103 XOUT 025 XOUT 026 102 CUNTINUE XOUT 027 GU TO 119 XOUT 02A C X001 029 č A COMMAND VERB HAS HEEN ENCOUNTERED. XOUT 030 С XOUT 031 103 GU TU (104+105+109+112+119) + 1 XOUT 032 C XOUT 033 č END. XOUT 034 č XOUT 035 104 RETURN XOUT 036 C XOUT 037 C . PRINT. XOUT 038 č STORE THE INDICES OF THE STATE VARIABLES REQUESTED FOR PRINTING (NUM(1)) IN THE PRINT STACK (APR(ANPR)). XOUT 039 С XOUT 040 С XOUT 041 105 00 108 1=1.14 X0UT 042 С XOUT 043 C . INDICES .LF. D ARE ASSUMED A PLANK AND APE IGNORED. XOUT 044 C C INDICES .GT. 99 PRODUCE A DIAGNOSTIC. REPETITIOUS REQUESTS XOUT 045 ARE IGNORED. XOUT 046 č XOUT 047 IF (NUM(1).LE.0) GO TO IOA XOUT 048 IF (NUM(1).GT.99) GO TO 120 1F (ANPP.LE.0) GO TO 107 XOUT 049 XOUT 050 00 106 J=1 . XNPH XOUT 051 (NUM(1).E0.XPH(J)) 60 TO 108 XOUT 052 106 CONTINUE XOUT 053 107 ANPH=XUPH+1 XOUT 054 XPR (ANPR) =NUM (1) XOUT 055 108 CONTINUE XOUT 056 GO TO 101 XUUT 057 C XOUT 058 С FLOW. XOUT 059 . č SET THE FLOW PRINT FLAG FOR THE K-TH FLOW IN THE FLOW XOUT 060 REFERENCE TABLE (XFR(K)) FOR EACH PAIR OF INDICES IN THE RANGE 1 THROUGH 99 (NUM(I) AND NUM (J)) FUR WHICH A CURRESPONDING ENTRY EXISTS IN THE FLOW REFERENCE TABLE. CCC XOUT 061 XOUT 062 XOUT 063 C XOUT 064 109 00 111 1=1.13.2 XOUT 065 J=I•I XOUT 066 IF (NUM(I).LE.0.AND.NUM(J).LE.0) GO TO 111 IF (NUM(I).LE.0.AND.NUM(J).GI.99) GO TO 120 IF (NUM(I).LE.0.AND.NUM(J).GI.99) GO TO 120 XOUT 067 XOUT 068 XOUT 069 00 110 K=1+XNF XOUT 070 INFO=XFH(K) XOUT 071 CALL XUNPAK (INFU. IP. II. 12) XOUT 072 IF (NUM(1).NF.II.0R.NUM(J).NE.I2) GO TO 110 IF (IP.E4.I) GO TO 111 XOUT 073 XOUT 074 AFR(K)=XFR(K)+10000 XOUT 075 GU TO 111 XOUT 076 CONTINUE 110 XOUT 077 111 CONTINUE XOUT 078 GO TO LUI XOUT 079 С XOUT 080 . PLOT. С XOUT 061 THE NEXT (XNPLT) CARDS ARE READ. ONE CARD PER PLOT. WITH THE FIRST ETVE NUMERIC FIELDS INTERPRETED AS THE INDICES OF UP TO FIVE STALE VARIABLES TO APPEAR IN EACH PLUT. CC XOUT 082 XOUT 083 č XOUT ANA С XOUT CAS 112 XNPLIENUM (1) XOUT 086 IF (INPLT.LE.O. GP. INPLT. GT. 20) GU TO 121 XOUT 087 DO 116 LE1.XNPLT XOUT ORA NEAU (11.123) IVI HH. (NUM (J) . J=1.14) YOUT ARS 00 113 .j=1+3 XOUT 090 11 (IVEPH. EQ. KEY(J)) GO TO 122 XOUT 091

S6U LOOX

113

CUNTINUE

		XAVN2(1)=0	XOUT	093
		00 117 J#1+5	XOUT	
		IF (FUM(J)_LF_0) (50 TO 117	XOUT	
		1 (HUM(J).61.99) GU TU 120	XOUT	
		XNVRS(1) = XNVRS(1) + 1	XOUT	
		K=XNVHS(1)	XOUT	
		1r (XNPL.LF.0) GO TO 115	XOUT	
		UU 114 L=1.XNPL	XOUT	
		1F (NUM (J) . EQ. XPL (L)) GO TO 116	XOUT	
	114	CUNTINUF	XOUT	
	115	ANPL=XNPL+1	XOUT	
	•	APL(XNPL) = NUM(J)	XOUT	
	116	XNLOC(Isk)=XNPL	XOUT	
	117	CONTINUE	XOUT	
	118	CONTINUE	XOUT	-
		60 10 101	XOUT	
С			XOUT	109
C		• • • • IF FRRORS OCCUPED GENERATE A DIAGNOSLIC.	XOUT	
C			XOUT	
	119	$WRITE$ (U2,124) $IVFRP_{0}(N(IM(I), I=1, 14))$	XOUT	112
		WHITE (U2+125)	XOUT	113
		STOP	XOUT	114
	120	WRITE (U2+124) IVEPA+ (NUM(I)+1=1+14)	XOUT	115
		wklTt (U2,126)	XOUT	116
		STOP	XOUT	117
	121	WRITE (U2+124) IVERB+(NUM(I)+1=1+14)	XOUT	118
		WRITE (U2,127)	XOUT	119
		S10P	XOUT	150
	155	#FITE (U2+124) IVERR+(NUM(I)+1=1+14)	XOUT	151
		wRITE (U2+128)	XOUT	155
		STOP	XOUT	123
С			XOUT	124
C		• • • • FORMATS USED IN THIS SUBROUTINE.	LOUT	125
C			XOUT	159
		FORMAT (A4.6X.1415)	XOUT	127
		FORMAT (33H0++++ERPOR IN DATA SECTION INPUT/10X+A4+6X+1415)	XOUT	
		FORMAT (25H ILLEGAL COMMAND VERR)	XOUT	129
		FORMAT (43H STATE VARIABLE INDEX .LE. O UH .GT. 99)	XOUT	
		FORMAT (45H NO. OF PLOTS REQUESTED .LE. U OR .GT. 20)	XOUT	
		FORMAT 122H COMMAND VERH ENCOUNTERED WHILE PROCESSING PLOT R		
		LOUEST, CHECK FOR NO. OF PLOTS REQUESTED .NF. NO. OF SUBSEQUENT CA		
~	2	2D51	XOUT	
C		5ND	XOUT	
		END	XOUT	136

2		SUBROUTINE APLICEN	PLGN	
•		THE COMMENTAL CONCERNMENTS OF THE MAIN DUCCAM	PL GN	
		• • • THIS SUMROUTINE (APLGEN) IS CALLED BY THE MAIN PROGRAM		
		(HANGES) TO GENERATE THE REQUESTED PLOTS UNDER CONTROL OF	PLGN	
		THE PLOT GENERATION VARIABLES. THE INFORMATION FOR PLOTTING	PLGN	
		IS UN THE TEMPOHANY MASS STORAGE OFFICE (U3).	PLGN	
			FLGN	
		INTEGER I + I I + I STOP + LOC + LU + NVARS + UI + 117 + U3 + XFR + ANF + XNLOC	PLGN	-
		INTEGER XNPLOXNPLTOXNPPOXNSTOXNPSOXPLOXPROXSI	PLGN	00
		HEAL DIVIINEVISIALVXVXEVALUCVXI INF (11) VXMAXVXVINVX1ST	PLGN	01
		REAL X1010+X11T0+Y1NC+YLINE(6)+YMAX+YMIN+Y151+Z(100)+ZZ	PLGN	0
			PLGN	
		COMMON /TX/ OT+TIME+ISTRT+X(99)+XHF+X10T0+X11T0	PLGN	
		COMMON /XSY5/ XF (300) + XFR (300) + XNI OC (20+5) + XNPL + XNPP + XNST + XI		
	-	1VH5(20)+XPL(99)+XPR(99)+X5T(99)	PLGN	
		COMMON //UNI/ LU-U)-U2-U3	PLGN	
		CONNOR //041/ E010/102103		
			PLGM	
		DO 106 II=I+XNPLT	PLGN	
		XmIN=1.F30	PL GN	
		XMDX=-XM1N	PLGN	
		AWIN=)*E30	PLGN	0
		YMAX=~YM]N	PLGN	0;
			PLGN	
		SEARCH THE PLOT DATA, DETERMINING MAA AND MIN VALUES FOR	PLGN	0
		SCALL.	PLGN	
			PLGN	
		REWIND 113	PLGM	
	101			
1	101	READ (U3) ISTOP TIME ((2(1)) I=I, XNP())	PLGN	
		IF (ISTOP.GF.1) GU 10 103	PLGN	
		XMIN=AMTN1 (XMIN+TIME)	PLGN	0
		XMBX=AMBXI(XMAX+IIME)	PLGN	0
		NVAKS=XNVKS(11)	PLGN	0
		DO 102 J=1,NVARS	PLGN	0
		$LUC=XNLOC(II \bullet I)$	PLGN	0
		Z Z = Z (L UC)	PLGN	
		YMIN=AMIN1 (YMIN+ZZ)	PLGN	
	102		PLGN	
	I VE	G0 10 101	PLGN	
		66 16 101	PLON	- 11,
			- C	
			PLGN	
	• •	• • • • CALCULATE GRAPHICAL SCALING VALUES.	PLGN	04
1	••		PLGN	04
1	••• 103	CALL XRND (YMIN+YMAX+YMIN+YMAX)	PLGN	04
1	••• 103	CALL XRAD (YMIN&YMAX&YMIN&YMAX) XLINC(1)=XMIN	PLGN	0
1	103	CALL XPND (YM1N+YMAX+YM1N+YMAX) XLINL(1)=XM1N XLINL(11)=XMAX	PLGN PLGN PLGN	0000
1	103	CALL XRAD (YMIN&YMAX&YMIN&YMAX) XLINC(1)=XMIN	PLGN PLGN PLGN PLGN	
1	•••	CALL XPND (YM1N+YMAX+YM1N+YMAX) XLINL(1)=XM1N XLINL(11)=XMAX	PLGN PLGN PLGN PLGN PLGN	0.000
1	103	CALL XHAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XM1N XLINE(11)=XMAX XINC=(XMAX-XM1N)/10* XIST=XM1N	PLGN PLGN PLGN PLGN PLGN PLGN	000000
1	103	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINL(1)=XMIN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DU 104 (=2*10	PLGN PLGN PLGN PLGN PLGN PLGN PLGN	000000
		CALL XMAD (YMIN*YMAX*YMIN*YMAX) XLINL(1)=XMIN XLINL(11)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DU 104 1=2*10 XIST=x1S1*XINC	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	0000000
	103	CALL XMAD (YMIN,YMAX,YMIN,YMAX) XLINL(1)=XMAN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10, XIST=XMIN DU 104 T=2,10 XIST=XIST+XINC XLINL(T)=XIST	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
		CALL XMAD (YMIN,YMAX,YMIN,YMAX) XLINL(1)=XMAN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10, XIST=XMIN DU 104 T=2,10 XIST=XIST+XINC XLINL(1)=XIST YLINL(1)=YMIN	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
		CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINL(1)=XMIN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DO 104 1=2*10 XIST=X1ST*XINC XLINL(1)=X1ST YLINL(1)=YMAX	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
		CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINL(1)=XMIN XLINL(11)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DU 104 T=2*10 XIST=XIST*XINC XLINL(1)=XIST YLINL(1)=YMAX YINC=(YMAX-YMIN)/5*	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	000000000000000000000000000000000000000
		CALL XMAD (YMIN,YMAX,YMIN,YMAX) XLINL(1)=XMAN XLINL(1)=XMAN XINC=(XMAX-XMIN)/10, XIST=XMIN DU 104 T=2,10 XIST=XIST+XINC XLINL(1)=XIST YLINL(1)=YMIN YLINL(6)=YMAA YINC=(YMAX-YMIN)/5, YIST=YMIN	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	000000000000000000000000000000000000000
		CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINL(1)=XMIN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DU 104 [=2*10 XIST=X1ST*XINC XLINL(1)=XIST YLINL(1)=YMIN YLINL(6)=YMAX YINC=(YMAX-YMIN)/5* YIST=YMIN OU 105 [=2*5	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
		CALL XMAD (YMIN,YMAX,YMIN,YMAX) XLINL(1)=XMAN XLINL(1)=XMAN XINC=(XMAX-XMIN)/10, XIST=XMIN DU 104 T=2,10 XIST=XIST+XINC XLINL(1)=XIST YLINL(1)=YMIN YLINL(6)=YMAA YINC=(YMAX-YMIN)/5, YIST=YMIN	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1		CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINL(1)=XMIN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DU 104 [=2*10 XIST=X1ST*XINC XLINL(1)=XIST YLINL(1)=YMIN YLINL(6)=YMAX YINC=(YMAX-YMIN)/5* YIST=YMIN OU 105 [=2*5	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XMIN XLINE(11)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DO 104 1=2*10 XIST=X1ST*XINC XLINE(1)=X1ST YLINE(1)=YMAX YINC=(YMAX-YMIN)/5* YIST=Y1ST*YINC DO 105 1=2*5 YIST=Y1ST*YINC	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XMIN XLINE(1)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DU 104 [=2*10 XIST=X1ST*XINC XLINE(1)=XIST YLINE(1)=YMIN YLINE(1)=YMIN YLINE(6)=YMAX YINC=(YMAX-YMIN)/5* YIST=Y1ST*YINC YLINE(1)=YIST	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XMIN XLINE(11)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DO 104 1=2*10 XIST=X1ST*XINC XLINE(1)=X1ST YLINE(1)=YMAX YINC=(YMAX-YMIN)/5* YIST=Y1ST*YINC DO 105 1=2*5 YIST=Y1ST*YINC	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XMAD (YMIN,YMAX,YMIN,YMAX) XLINL(1)=XMIN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10. XIST=XMIN DU 104 T=2,10 XIST=x1ST+XINC XLINL(1)=XMIN YLINL(1)=YMIN YLINL(1)=YMAX YINC=(YMAX-YMIN)/5. YIST=YIST+YINC YLINL(1)=YIST • • • • GENERATE THE PLOT.	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XMMD (YMIN,YMAX,YMIN,YMAX) XLINL(1)=XMIN XLINL(1)=XMAX XINC=(XMAX-XMIN)/10. XIST=XMTN DU 104 T=2,10 XIST=X1ST *XINC XLINL(1)=X1ST YLINL(1)=YMIN YLINL(1)=YMAX YINC=(YMAX-YMIN)/5. YIST=YMIN OD 105 T=2,5 YIST=YIST*YINC YLINL(1)=YIST • • • • GENEHATE THE PLOT. HEWIND U3	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XMIN XLINE(1)=XMAX XINC=(XMAX-XMIN)/10* XINC=(XMAX-XMIN)/10* XIST=XMIN DO 104 [=2*10 XIST=X1SI*XINC XLINE(1)=XIST YLINE(1)=XIST YLINE(1)=YMIN YLINE(1)=YMIN YLINE(6)=YMAX YINC=(YMAX-YMIN)/5* YIST=YMIN OO 105 [=2*5 YIST=Y1ST*YINC YLINE(1)=YIST * * * GFNEHATE THE PLOT* KEWIND U3 CALL XGPAPH (11*NVAPS*XNLOC*XLINE*YLINE*Z*YMIN*YMAX*XNPL*XPL)	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XMIN XLINE(1)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DO 104 1=2*10 XIST=X1ST*XINC XLINE(1)=X1ST YLINE(1)=YMAN YLINE(1)=YMAN YINC=(YMAX-YMIN)/5* YIST=YMIN OO 105 1=2*5 YIST=Y1ST*YINC YLINE(1)=YIST * * * * 6FNEHATE THE PLOT. MEWIND U3 CALL XGPAPH (11*NVAR5*XNLOC*XLINE*YLINE*Z*YMIN*YMAX*XNPL*XPL) CONTINUE	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XMIN XLINE(1)=XMAX XINC=(XMAX-XMIN)/10* XINC=(XMAX-XMIN)/10* XIST=XMIN DO 104 [=2*10 XIST=X1SI*XINC XLINE(1)=XIST YLINE(1)=XIST YLINE(1)=YMIN YLINE(1)=YMIN YLINE(6)=YMAX YINC=(YMAX-YMIN)/5* YIST=YMIN OO 105 [=2*5 YIST=Y1ST*YINC YLINE(1)=YIST * * * GFNEHATE THE PLOT* KEWIND U3 CALL XGPAPH (11*NVAPS*XNLOC*XLINE*YLINE*Z*YMIN*YMAX*XNPL*XPL)	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	
1	104	CALL XPAD (YMIN*YMAX*YMIN*YMAX) XLINE(1)=XMIN XLINE(1)=XMAX XINC=(XMAX-XMIN)/10* XIST=XMIN DO 104 1=2*10 XIST=X1ST*XINC XLINE(1)=X1ST YLINE(1)=YMAN YLINE(1)=YMAN YINC=(YMAX-YMIN)/5* YIST=YMIN OO 105 1=2*5 YIST=Y1ST*YINC YLINE(1)=YIST * * * * 6FNEHATE THE PLOT. MEWIND U3 CALL XGPAPH (11*NVAR5*XNLOC*XLINE*YLINE*Z*YMIN*YMAX*XNPL*XPL) CONTINUE	PLGN PLGN PLGN PLGN PLGN PLGN PLGN PLGN	

```
SUBROUFINE XPLOT (VAL +1510P)
                                                                                   XPLOTOO1
                                                                                   XPL OT 002
С
č
              . THIS SUBROUTINE (APLOT) IS CALLED BY THE MAIN PROGRAM
                                                                                   XPLOTO03
  . .
          (MANUES) TO GENERATE UNE RECORD OF PLOT VANTARLE VALUES ON
                                                                                   XPI OTO04
CCCC
          MASS STOKAGE OFVISE (U3) AT FACH CALL. THE FORMAT OF FAC
HECURD IS ISTOP (ISTOPE) IS FINAL ISTOPED IS OTHERWISE).
                                                        THE FORMAT OF FACH
                                                                                   XPL OTOOS
                                                                                   XPI OTOOM
          TIME + AND VALUED. THE NUMBER OF RECORDS 15 XMPL+7. STORAL
OF THE STATE VANTABLES TO BE SAVED FOR PLUTTING ARE IN THE
                                                                       STURAGE
                                                                                   XPL OTOO7
c
                                                                                   XPLOTOOR
                                                                                   XPLOT009
          AMHAY (VAL).
С
                                                                                   XPLCT010
       INTEGER I.ISTOP.J.LU.UI.UP.U.S.XFR.XNF.XNLUC.ANPL
                                                                                   XPLCT011
       INTEGER XNPLT.XNPP.XNST.XNVPS.XPL.XPR.AST
                                                                                   XPI OTO12
       NEAL DIVIJME . ISINT . VAL ( 300) . X . XF . XIOTO . A) ITO
                                                                                   XPL DID13
С
                                                                                   XPLOT014
       COMMON /TX/ DT.TTHE.TSTHI.X(94).XNE.X1010.X1110
                                                                                   XPLOT015
       COMMON /XSYS/ XF (300) + AF ( 300) + XNLOC (20+5) + XNPL + XNPL + XNPP + XNST + XNXPLOT016
      1VH5(20) + XPL (99) + XPH (99) + X51 (99)
                                                                                   XPL01017
                                                                                   XPI OTOLS
       COMMON /XUNT/ LU-U1-U2+U3
                                                                                   XPLOT019
C
       (0 101 1=1. ANPL
                                                                                   XPLOT020
          J=XPL(1)
                                                                                   XPI 01021
  101 VAL(I)=A(J)
                                                                                   XPL OT 022
       WHITE (U3) ISTOP .TIME . (VAL (1) .I=1.XNPL)
                                                                                   XPLOT023
                                                                                   XPLOT024
       METURN
                                                                                   XPLOTO25
C
       END
                                                                                   XPLOT026
       SUBROUTINE XPRINT (VAL. XPR. XNPR)
                                                                                   XPRNT001
С
                                                                                   XPRN TOO2
              . THIS SUBROUTINE (APRINT) IS CALLED BY THE MAIN PROGRAM
                                                                                   XPHNT003
CCC
  . . . .
          (RANGES) TO PRODUCE PRINTED OUTPUT OF STATE VARIABLE NAMES
AND VALUES, FOUR STATE VAPIABLES PER LINE. STORAGE OF THE
STATE VARIABLES TO BE PRINTED ARE IN THE ARRAY (VAL).
                                                                                   XPRNT004
                                                                                   XPRNT005
Ċ
                                                                                   XPHNT006
С
                                                                                   XPRNT007
       INTEGER XFR (99)
                                                                                    XPRNTOOR
                                                                                   XPRNT009
       HEAL DI.TIME.TSTRT.VAL (300) .X.X10T0.X11T0
                                                                                   XPRNT010
С
                                                                                   XPRNT011
       CUMMON /TX/ DT+TIPE+TSTRT+X(99)+XNF+X10T0+X1110
                                                                                    XPRNT012
       COMMON /XUNT/ LU+U1+U2+U3
                                                                                   XPHNT013
С
                                                                                   XPRNT014
       00 101 1=1.XNPR
                                                                                   XPRNT015
           J=XPH(I)
                                                                                   XPRNT016
  101 VAL(1)=>(J)
                                                                                   XPRNT017
С
                                                                                   XPRNT018
cc
  . . . . . CHECK FOR EQUALITY OF TIME AND ISTRT.
                                                                                   XPRNT019
                                                                                   XPRNT020
       IF (TIME.LE.TSTRT) GO TO 102
                                                                                    XPHNT021
  WRITE (U2+110) TIME
102 NLINE=XIPR/4+1
                                                                                   XPRNT022
                                                                                   XPRNT023
       NKNT=MUD (XMPR.4)
                                                                                   XPRNT024
       1F (NENI.NF.0) GO TO 103
                                                                                   XPHNT025
       NLINE=NLINE-1
                                                                                   XPRNT026
       NKNT=4
                                                                                   XPRNT027
   103 J1=1
                                                                                   XPRNT028
       DO 109 1=1.NLINE
                                                                                    XPRNT029
           IF (1.EO.NLINE) GO TO 104
                                                                                   XPRNT030
           J2=J1+3
                                                                                   XPRNT031
           WRITE (U2+111) (XPR(J)+VAL(J)+J=J1+J2)
                                                                                   XPRNT032
          IF (LU.GT.0) WPITE (LU.115) (XPR(J).VAL(J).J=J1.J2)
GO TU 104
                                                                                   XPRNT033
                                                                                   XPRNT034
           J2=J1+NKNT-1
  104
                                                                                   XPRNT035
           GO TU (105.106.107.108) . NKNT
                                                                                   XPRNT036
  105
           WRITE (U2+112) (XPR(J)+VAL(J)+J=J1+J2)
                                                                                   XPRNT037
           IF (LU.GT.0) WRITE (LU.115) (XPR(J).VAL(J).J=J1.J2)
                                                                                   XPRNT038
           GU 1U 109
                                                                                   XPPNT039
           WRITE (12.113) (XPR(J).VAL(J).J=J1.J2)
   106
                                                                                   XPRNT040
           IF (LU.GT.0) WRITE (LU.115) (XPR(J).VAL(J).J=J1.J2)
                                                                                   XPRNT041
           GO 10 104
                                                                                   XPRNT042
           WRITE (U2+114) (XPR(J)+VAL(J)+J=J1+J2)
  107
                                                                                   XPRN1043
             (LU.GT.0) WRITE (LU.115) (XPR(J) + VAL(J) + J=J1+J2)
           IF
                                                                                   XPRNT044
           GO TU 109
                                                                                   XPRN1045
  108
          WRITE (U2+111) (XPR(J)+VAL(J)+J=J1+J2)
                                                                                   XPRN1046
          IF (LU.GT.0) WRITE (LU.115) (XPR(J).VAL(J).J=J1.J2)
                                                                                   XPRNT047
   109 J1=J1+4
                                                                                   XPHNT048
       PETURN
                                                                                   XPRNT049
c
                                                                                   XPHNT050
  . . . . . FORMATS USED IN THIS SUBROUTINE.
                                                                                   XPHNT051
C
                                                                                   XPRNT052
  110 FORMAT (HHOTIME = +F12-1)
                                                                                   XPRNT053
  111 FOHMAT (10x+4(2HX(+12+4H) = +F)2+2+5X))
                                                                                   XPRNT054
  112 FORMAT (10x+2HX(+12+4H) = +F12+2+5X)
113 FORMAT (10x+2(2HX(+12+4H) = +F12+2+5X)
                                                                                   XPRNT055
                                                                                   XPHNT056
   114 FURMAT (10X+3(2HX(+12+4H) = +F12+2+5K))
                                                                                   XPRNT057
  115 FORMAT (1H0+12+E12+5)
                                                                                   XPHNT05A
С
                                                                                   XPHNT059
       END
                                                                                   XPRNT060
```

		SURPOUTLIE AND (7MTN+7MAX+R17K)N+R17MAA)	TRNN	001
C			REPAIN	005
C		• • • DOLD STOCHARTING (YMAL) 12 CALLED HA THE STOPPOLITINE	XHND	003
C		(XPLGER) TO DETERMINE THE APPROPRIATE SCALING FOR A PLOT	XHND	004
C		OF A FUNCTION PHONE VALUES RANGE FROM ZMIN TO ZMAX. HNZMIN	KHND	005
С		AND POZMAN ARE THE EXTREME VALUES OF THE GRAPH.	XNNP	106
С			ARND	057
		INTEGER I	XRND	900
		KEAL HUVAR . KN/MAX . KN/HIU . VAK . / . /MAX . / MIN . / NU . //	XHND	009
C			XRND	010
С		• • • THE CASE WHERE /MIN = ZMAX IS INFATED SEPARATELY.	XRND	011
С			XRND	012
		IF 1(AH5(2MIN-ZMAX)).G[.1.0F-30) GO TO 107	XUND	013
		IF (ARS(/MAX).GT.1.0) 60 TO 101	XRND	014
		WN/MIN==I.	XHND	015
		KN/MAX=1.	XRND	016
		RETURN	XHND	017
С			XRND	
С		• • • SCALE Z UNTIL THE FIRST SIGNIFICANT DIGIT IS IN THE	XRND	019
С		THOUSANDS PLACE AND HOUND AT THE DECIMAL PLACE.	XRND	020
С			XRND	
	101	ζ=ζ14Αμ	XRND	022
		1=0	XRND	023
	102	1F (7.0E.1000.) GO TO 103	XHND	024
		∠=∠°I0.	XRND	025
		1 = 1 - 1	XRND	
		G0 10 102	XRND	
	103	1F (2.L).10000) 60 TO 104	XRND	028
		2=2/10.	XRND	029
		1=1+1	XRND	
		GO TU 103	XHND	031
	104	2=1N1(2+.5)	XRNC	
С			XRND	033
C	• •	OFTERMINE THE NUMBER OF SIGNIFICANT UIGITS IN Z. TRUNCATE		
С		THE LAST UNE. AND USE THIS NUMMER AS A RASIS FOR SETTING THE	XRND	035
С		GRAPH VALUES.	XRND	
С			XRND	0.37
		Z = 2/10.	XRND	
		1=1+1	XRND	
	105	2 RND=IN1(7)	XRND	
С			XRND	
С		CHECK FOR THE EQUALITY OF ZEND AND Z.	XRND	
С			XRND	
		IF ((AHS(ZRND-Z)).GT.1.0E-30) GU TO 106	XRND	
		7=2/10.	XRND	
		1 = 1 + 1	XRND	-
		GO TO 105	XRND	
	106	IF (Z. GE. O.) RH7MIN=/PNI)-I.	XRND	
		1F (2.L1.0.) RN2MIN=7RND-2.	XRND	
		HNZMAX=HNZMIN+3.	XRNP	
С			XRND	
		RESTORE THE NUMBERS TO THE ORIGINAL MAGNITURE.	XRND	
C			XRND	
		RN2MIN=HN7WIN+10.0+1	XRND	
		KNZMAX=HN7MAX#10.**I	XRND	
		RETURN	XRND	
С			XRND	
С	• •	• • • IN THE GENERAL CASE THE DIFFERENCE, ZMAX-ZMIN, IS	XRND	058
С		TRUNCATED TO THE FIRST SIGNIFICANT DIGIT AND ENLARGED IF	XRND	
С		NECESSARY TO ENCOMPASS THE ENTIRE RANGE . 2MIN-ZMAX.	XRND	
C			XRND	
	107	VAR=ZMAX-ZMIN	XRND	
		1=0	XRND	
	108	IF (VAN.GE.1.) GO TO 109	XRND	
		VAH=VAH®10.	XRND	
		1=1-1	XRND	
		GO TO LUA	XRND	
	109	1F (VAR.LT.10.) GO TO 110	XRND	
		VAR=VAR/10.	XRND	
		I=I + I	XRND	070
		GO TO 109	XRNP	
	110	RNVAH=INT (VAR)	XRND	

E

```
XRND 073
CCC
  . . . . . CHECK FUR THE EQUALITY OF RNVAN AND VAR.
                                                                                                          XRND 074
                                                                                                          XRND 075
         IF ((AHS(RNVAR-VAR)).LE.1.0E-30! GO TO 111
                                                                                                          XRND 076
         IF (VAR.GT.O.) HNVAREPNVAH+1.
                                                                                                          XRND 077
         IF (VAH.LT.O.) HNVAR=HNVAH-1.
                                                                                                          XRND 078
C
C
C
                                                                                                          XRND 079
            • • • TRUNCATE ZMIN AT THE SAME DECIMAL PLACE AS THE
DIFFLMENCE, ZMAX-ZMIN, WAS IPUNCATED AND LUWER THIS VALUE IF
NECESSARY TO INSUME THAT IT IS LESS THAN ZMIN. THIS VALUE IS
USED FOR RNZMIN AND THE TPUNCATED DIFFLMENCE, RNVAR, IS ADDED
TD OBTAIN RNZMAX (HNVAM IS ENLARGED IF NECESSARY TO INSURE
INCLUSION OF THE ENTIRE INTERVAL).
                                                                                                          XRND ORA
   .
                                                                                                          XRND 081
                                                                                                          XRND 082
0000
                                                                                                          XRND 083
                                                                                                          XRND 084
                                                                                                          XRND 085
č
                                                                                                          XRND 086
  111 Z=2MIN+10.++(-1)
                                                                                                          XRND 087
        ZZ=ZMAX*10.**(-1)
ZRNU=1N1(7)
                                                                                                          XRND 088
                                                                                                          XRND 089
         IF (VAN+LT.0.) 60 TO 112
                                                                                                          XRND 090
        IF (2.6C.0.) PN/MIN=7HNI)
IF (7.1.1.0.) PN/MIN=7HNI)=1.
IF (HN/MIN+PNVAF.LT.22) RNVAR=HNVAR+1.
                                                                                                          XRND 091
                                                                                                          XRND 092
                                                                                                          XRND 093
         60 10 113
                                                                                                          XHND 094
   112 1F (2.GI.O.) HNZMIN=/HND+1.
                                                                                                          XRND 095
         IF (2.LE.D.) KNZMINEZHNU
IF (KNZMIN+RNVAH-GT.Z7) RNVAHEKNVAR-1.
                                                                                                          XRND 096.
                                                                                                          XRND 097
  113 RNZMAX=HNZMIN+RNVA4
                                                                                                          XRND 09A
С
                                                                                                          XRND 099
C
  . . . . . MESTORE THE NUMBERS TO THE ORIGINAL MAGNITURE.
                                                                                                          XRUD IOO
                                                                                                          XRHD 101
         PNZMIN=FN7MIN=I0.=+1
                                                                                                          XPND 102
         WN/MAX=HN/MAX*10.**1
                                                                                                          XRND 103
         HE TUNH
                                                                                                          XRND 104
С
                                                                                                          XRND 105
        1 413
                                                                                                          THUE 196
```

	SUBROUTINE XUNPAK (1V+1P+1F+1T)	XUNPKOOI
С		XUNPK002
С	THIS SUBPOUTINE (XUNPAK) IS CALLED BY THE MAIN PROGRAM	XUNPK003
С	(RANGES) AND THE SUBROUTINE (XOUT) TO UNPACK INFORMATION	XUNPK004
C	STORED IN THE FLOW REFERENCE TABLES.	XUNPK005
С	SOME VARIABLE DEFINITIONS	XUNPK006
C	IV - INTEGER VANIANLE CONTAINING THE WORD TO BE	XUNPK007
C	UNPACKED.	XUNPKOOR
C	IF - INTEGER VARIABLE CONTAINING THE INDEX OF SOURCE	XUNPK009
č	COMPACTMENT.	XUNPK010
C	IT - INTEGER VARIABLE CONTAINING THE INDEX OF	XUNPK011
Ċ	DESTINATION COMPANTMENT.	XUNPK012
č		XUNPK013
•	INTEGEN I.IF.IP.IT.IV	XUNPK014
С	THEFT TATIATALA	
~	1P=1V/10000	XUNPK015
	1=1V-1P*10000	XUNPKOI6
		XUNPK017
	IF=1/100	XUNPK018
	IT=I-IF*100	XUNPK019
	RETURN	· XUNPK020
С		XUNPK021
	END	XUNPK022

APPENDIX #3

Example Subroutine Substitution

SUBROUTINE RUMEN will be used to explain the procedure for substituting new or revised subroutines for the original. An early version of RUMEN can be substituted into the deck for the current RUMEN routine. Notice that the substituted version of RUMEN has a shorter argument list than the current routine. By shortening the argument list in the calling routine the substitution is correctly made. RUMEN is called near the end of SUBROUTINE XFLWS. The switch is complete by dropping variables TIME, IN, MSZ, RDM, RFL, RDMD, MIPRO and VL75. Conversely, if more or different variables are needed from other routines the argument list of the call and the subroutine can be lengthened. The correct form of the calling statement and the substituted subroutine follow.

IN SUBROUTINE XFLWS: CALL RUMEN (GREEN, DRY, NGN, NFOR, LVWT, SUPN, SUPL, OMN, GAN, GNN) The substituted RUMEN routine:

SUBROUTINE RUMEN (GREEN, DRY, NGN, NFOR, LVWT, SUPN, SUPL, OMN, GAN, GNN) RUMEN002CALCULATES INTAKE AND GAIN FOR RUMINAN) ANIMALS AS A RUMEN003 FUNCTION OF CONTENT OF FORAGE RUMFN004SUBROUTINE SUITABLE FOR NON-REPRODUCTIVE GROWING ANIMALS. RUMEN005 BUT IS NOT DESIGNED FOR MATURE AND PREGNANT ANIMALS. RUMFN006 RUMEN007 REAL INTAK .NFOR .NDIET .NGN .NSU .LVWT RUMEN009 INPUTS RUMEN010 RECRD032 GREEN=AMOUNT OF LIVE FORAGE GRAMS PER METER SQ RUMEN011 DRY =AMOUNT OF STANDING DEAD FORAGE GRAMS PER METER SQ RUMEN012 NGN = NITROGEN CONTENT OF GREEN FORAGE (PERCENT) RUMEN013 NFOR = AVERAGE NITROGEN CONTENT OF FORAGE SELECTED (PERCENT) RUMEN014 NFOR = DIET VALUES FROM FUNCTION DIET RUMEN015 LVWT= ANIMAL WEIGHT IN GRAMS SUPL = PROPORTION OF LVWT GIVEN AS SUPPLEMENTAL FEED. DIMEMSION-- 1/TIME RUMEN018 SUPN = NITROGEN CONTENT OF SUPPLEMENTAL FEED(PERCENT) RUMEN019 OUTPUTS RUMEN020 GNN = INTAKE OF LIVE FORAGE (GRAMS) RUMEN021 INTAKE OF STANDING DEAU FORAGE (GRAMS) UNCORRECTED FOR OMN =RUMEN022 AMOUNT OF SUPPLEMENTAL FEED SUPPLIED RUMEN023 = ANIMAL GAIN PER DAY (GRAMS) GAN RUMEN024 NITROGEN CONTENT OF STANDING DEAD HERBAGE ASSUMED TO BE 1.0 PCT RUMEN025 DATA FOR SUBROUTINE FROM RICE 1972 RUMEN026 COMMENTS AND VERIFICATION BY COOK 1972 RUMEN027 RUMEN028 IF (LVWT.LE.0.0) GO TO 105 RUMEN030 AVAIL TERM ACCORDING TO BEMENT 1969 RUMEN031 CORRECTION OF INTAKE FOR FORAGE AVAILABILITY RUMEN032 RUMEN033 AVAIL=1.0-EXP(-00.167*(GREEN+DRY))+SUPL/0.02 IF (AVAIL.GT.1.0) AVAIL=1.0 RUMEN035 NSD=1.0 RUMEN036 IF (NFOR.LE.0.4) NFOR=0.4 RUMEN037 NDIET=NFOR RUMEN038 FOREF=0.02 RUMEN039 0=0.0 RUMEN040 RUMEN041BW75 IS THE STANDARD BODY WT IN KILUGRAMS TO THE 3/4 POWER RUMEN042 RUMEN043 8w75=(0.001*LVwT)**0.75 K=1 RUMEN045 RUMEN046 COMPUTE INTAKE AS A FUNCTION OF NITROGEN CONTENT OF DIET RUMEN047CALCULATE INTAKE PER KILOGRAM OF METABOLIC BODY WEIGHT RUMEN048 RUMEN049 101 BWINT= (50.0*NDIET-20.0) *AVAIL RUMEN050 INTAK=BWINT*BW75 RUMEN051 IF (SUPL.LE.0.0) GO TO 102 RUMEN052 RUMEN053IF SUPPLEMENTAL FEED, CALCULATE INTAKE RUMEN054 RUMEN055 FORIT=INTAK/LVWT-SUPL IF ((FORIT+SUPL).LE.0.0) GO TO 102 RUMEN057 RUMEN058 ADJUST NITROGEN OF DIET BY SUPPLEMENT RUMEN059 •••••CALCULATE NITROGEN IN DIET RUMEN060 RUMEN061 NOTET= INFOR*FORTT+SUPN*SUPL 1/ (FORTT+SUPL 1 RIMEN062

90

С

C

С

С

C

С

С

С

С

С

С

С

С

С

С

С

С

С

C C C

C C

С

С

С

С

С

С

С

С

С

С

С

С

С

С

С

С

С

С

С

С

C

C

С

C

C.

С RUMEN063COMPARE INTAKE TO REFERENCE RUMEN064 С С RUMEN065 TEST=FOREF/FORIT RUMEN066 IF (TEST.GT.0.98.AND.TEST.LT.1.02) GO TO 102 RUMEN067 K=K+1 RUMEN068 С RUMEN069 AT MOST 20 ITERATIONS ARE ALLOWED С RUMEN070 С RUMEN071 IF (K.GT.20) GO TO 102 RUMEN072 С RUMENU73 СIF ERROR IS GREATER THAN 2 PERCENT, ESTABLISH NEW REFERENCE RUMEN074 С AND RECOMPUTE INTAKE RUMENU75 C RUMEN076 FOREF=FORIT RUMEN077 GO TO 101 RUMEN078 102 INTAK=INTAK-SUPL*LVWT IF (NGN.GT.NSD.AND.NGN.GE.NFUR.AND.NFOR.GT.NSD) GO TO 103 RUMENOBO IF (GREEN+ DRY) 98,98,99 98 INTAK=0. GO TO 104 99 O=DRY/(GREEN+DRY) GO TO 104 RUMEN084 **103 CONTINUE** RUMENU35 С RUMEN086 PROPORTION OF LIVE (G) AND DEAD (O) FORAGE COMPUTED BY EQUATIONS С RUMEN087 С 0 = 1.0 - 6RUMEN088 С O(NSD) + G(NGN) = NFORRUMENO89 C: THEREFORE RUMEN090 С RUMEN091 O= (NGN-NFOR) / (NGN-NSD) RUMEN092 104 G=1.0-0 RUMEN093 С RUMFN094CALCULATE DEAD FORAGE INTAKE C RUMEN095 С RUMENU96 OMN=0*INTAK RUMENU 97 С RUMEN098CALCULATE LIVE FORAGE INTAKE C. RUMEN099 С RUMEN100 GNN=G*INTAK RUMEN101 С RUMEN102 THE CORRECT FORM OF THE EQUATION FOR GAIN WILL RESEMBLE С RUMEN103 С GAN = (0.35 * BWINT - 14.0) * BW75 RUMEN104 С THE FOLLOWING EQUATION IS A DUMMY PENDING DEBUG OF THE ABOVE RUMEN105 С RUMEN106 GAN=0.005*LVWT*G + 0.25*SUPL 1'05 CONTINUE RUMEN108 RETURN RUMEN109 С RUMEN110 END RUMEN111

APPENDIX #4

PROGRAM RANGES (INPUT, OUTPUT, PLTSV, TAPE5 = INPUT, TAPE6 = OUTPUT, TAPE3 = PLTSV, TAPE4, TAPE7)

We call this card a "program" card. It specifies the input, output and scratch tape devices. There is a one to one correspondence between the first three parameters and the following three equivalences. The total meaning is that TAPE5 corresponds to the system input device, TAPE6 corresponds to the system output device and TAPE3 is a user declared storage file, in this case the information to be plotted on the line printer is stored on file PLTSV. TAPE4 and TAPE7 are scratch files used for saving any information that may be desired. Specific WRITE statements on these devices are accounted for in SUBROUTINE XPRINT, where LU is defined to be 4 or 7 in the block data subprogram, then the state variables specified in the data section will be written on one of these respective devices. The information can subsequently be used in any manner, i.e. printed, punched, etc.

The meaning of the tape numbers should become clear when considered in conjunction with READ and WRITE statements. For example, READ (U1, 105) means read from TAPE5 which in this program corresponds to the system input device. Similarly, WRITE (U2, 106) will write on the system output device declared TAPE6 on the program card.

Variables Ul, U2 and U3 are initialized in a DATA statement at about line 66 of the block data subprogram. Then when the program says READ (Ul, 105) a read from TAPE5 = INPUT is implied because Ul was initialized to 5. Also, note these values can be readily changed in the DATA statement to any unit number that is convenient for your installation. For example, if your card reader is designated as unit 1 then just present DATA Ul/1/ and all subsequent READS will be from the card reader.

boli

APPENDIX #5

Stochastic Precipitation Generator

A stochastic precipitation algorithm requiring only three parameters for generating a daily precipitation regime has been developed by Todorovic and Woolhiser (1971) and is used in the RANGES model. The three parameters required by the algorithm are (i) probability of a dry day preceded by a dry day, P(D/D), (ii) probability of a dry day, P(D), and (iii) the average storm size, SS. The data values used for these parameters are average weekly values for the year and can be represented succinctly using a Fourier series for each parameter for the year interval. Five Fourier coefficients for each parameter are read by FUNCTION PPT and then expanded by SUBROUTINE SER into daily values for the year. This apparent complexity has been incorporated to allow RANGES to be used at most grassland sites in the western United States. The algorithm used to determine whether a storm occurs on any given day of the year uses the probability of a wet day preceded by a dry day, P(W/D), and the probability of a wet day preceded by a wet day, P(W/W). These can be obtained from the input, P(D/D) and P(D), by the methods of Heermann et al. (1971). A uniformly distributed random variable is used to determine if the conditional probability is high enough to generate a storm, based on whether a storm occurred the previous day. If a storm occurs, another uniform random variable is used in conjunction with the storm size parameter to generate the event size from an exponential distribution.

Algorithm for generating daily precipitation events.

- (i) Read in Fourier coefficients for the three precipitation parameters, P(D/D), P(D) and SS.
- (ii) Expand into a series representation of the daily values for the year.

 $P(D) = a_0 + a_1 \cos(\pi \cdot day) + b_1 \sin(\pi \cdot day)$

+ $a_2 \cos(2\pi \cdot day) + b_2 \sin(2\pi \cdot day)$

where a_0 is the mean response of the parameter over the year and a_1 , b_1 , a_2 , and b_2 are the remaining Fourier regression coefficients.

(111) Convert P(D/D) and P(D) to P(W/D) and P(W/W). Let t = the current day and t-1 = the preceding day.

$$P(W_{t}/D_{t-1}) = 1 - P(D_{t}/D_{t-1})$$

$$P(D_{t}/W_{t-1}) = P(D_{t}) - P(D_{t-1}) \cdot P(D_{t}/D_{t-1})$$

$$P(W_{t}/W_{t-1}) = 1 - P(D_{t}/W_{t-1})$$

- (iv) Initialize simulation to begin with no storm.
- (v) Generate a uniformly distributed random variable.
- (vi) Test to see if the random number (r₁) is less than the transition probability (P).
- (vii) If $r_1 \leq P$, then generate a storm, otherwise set precipitation for the day to zero and set the transition probability to be tested to P(W/D).
- (viii) If a storm occurs, r₁ > P, generate another uniform random number and compute the storm size from an exponential distribution.

$$SS = -\log r_2/\lambda$$

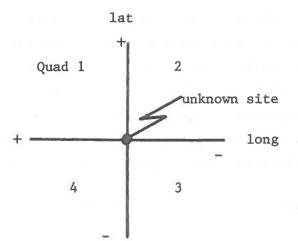
where

 r_2 = uniform random variable λ = 1/mean weekly storm size.

The Fourier coefficients required by this algorithm can be generated for most grassland sites in the western United States. A computer program, PREGEN, has been written which uses as input latitude, longitude and elevation to generate the necessary Fourier coefficients for input into RANGES. PREGEN checks an unknown site to see if it lies in the area of representativeness of a known weather station as reported by Heermann et al. (1971). If it does, then the Fourier coefficients are generated from the data for the known weather station. Otherwise, a linear interpolation is performed on surrounding known stations to determine a weighted average of their data, from which Fourier coefficients are generated. The average is weighted because elevation is more highly correlated with precipitation than proximity. Therefore, twice as much influence was ascribed to elevation similarity than to horizontal proximity.

Algorithm for interpolating between known weather stations for an unknown site.

- (i) Check to see if unknown site lies in area of representativeness of any known stations. If it does use the parameters for the known station unless the user specifies an interpolation is to be performed.
- (ii) If a site does not lie in the area of representativeness of any known station then set a conceptual coordinate system on the unknown site with the vertex at the site.



- (iii) Now find the minimum distance station within each quadrant within a radius of 120 miles.
- (iv) Calculate the weighting factors for, at most, four stations surrounding the unknown station. Note - there may be fewer than four stations.
 - a) Find the similarity index (SID) for station distances.

$$SID_i = 2w/(a+b_i)$$

where i = quadrants 1, 2, 3, 4;

- a = the minimum distance for the four quadrants;
- b, = the distance of the station in quadrant i; and
- w = a. In general, w is the minimum of a and b.

- b) Normalize the similarity indexes (SID).
- c) Find similarity index for station elevations (SIE).
- d) Normalize SIE.
- e) Compute weighting factor

WAI = $(2 \cdot SIE + SID)/3$

- f) Normalize WAI.
- (v) Generate Fourier series for the year for the three parameters P(D/D), P(D) and SS for, at most, four stations.
- (vi) Multiply the resulting daily values by the weighting factor.
- (vii) Sum the weighted daily values for each parameter to obtain the weighted average for the unknown site.
- (viii) Find new Fourier coefficients for the result and print them for use in RANGES.
 - (ix) If the user specifies rain generation from PREGEN, perform the rain generation algorithm as described for RANGES.

PREGEN users guide. The total PREGEN package consists of a FORTRAN deck and a data deck. The data deck has a card with user specified input parameters and a weather station card file supplied with the program. The user input card allows various PREGEN options to be chosen. All variables on the first data card are integer and should be right justified in the defined field.

	8	
Columns	Variable Name	Description
1–5	ILAT	Latitude of site to be interpolated
6-10	ILON	Longitude of site to be interpolated
11-15	ILEV	Elevation of site to be interpolated
16	IS	IS=1, print out name, latitude, longitude and elevation of stations searched
19	IP	IP=1, generate precipitation regime for interpolated site
22-2 4	IY	IY=n, number of years precipitation to be generated where $n < 999$

25	IO	IO=1, daily precipitation amounts printed, otherwise yearly
28	IZ	IZ=1, interpolation is desired even though the un- known may lie in the area of representativeness of a known station.
31	IT	IT=1, ignore interpolation routine, instead read Fourier coefficients from input and generate pre- cipitation amounts with IY and IO used as output criteria. This option implies that the weather station data file should not be read and should be replaced by seven cards with the following format:
lst card		TITLE FORMAT (2A4,A2)
		TITLE = name of the site for which precipitation is being generated
2nd card		AA(4) FORMAT (3X,F10.0)
		AA(4) mean response of $P(D/D)$ parameter
3rd card		AA(5),BB(5),AA(6),BB(6) FORMAT (4F10.0)
		AA(I), BB(I), I=5,6 are Fourier regression coefficients for $P(D/D)$
4th card		AA(1) FORMAT (3X,F10.0)
		AA(1) mean response of P(D) parameter
5th card		AA(2),BB(2),AA(3),BB(3) FORMAT (4F10.0)
		AA(I),BB(I), I=2,3 are Fourier regression coefficients for P(D)
6th card		AA(7) FORMAT(3X,F10.0)
		AA(7) mean response of SS parameter
7th card		AA(8),BB(8),AA(9),BB(9) FORMAT (4F10.0)
		AA(I),BB(I), I=8,9 are Fourier regression coefficients for SS

The second part of the data deck consists of 253 weather station records with the following information punched on two cards per station. A blank card will be the terminator for this section.

Columns	Variable Name	Description	
lst card			
1-10	NAME	Weather station name	
13-18	no variable	Average yearly rainfall	
20-22	NN	Representative miles north for station	
24-26	NS	Representative miles south for station	
28-30	NE	Representative miles east for station	
32-34	NW	Representative miles west for station	
37-40	LEV	Weather station elevation, feet	
43-47	LAT	Weather station latitude, degrees	
50-54	LON	Weather station longitude, degrees	
2nd card			
1-5	SWDT(1)	Mean response of parameter P(D)	
6-10	SWDT(2)	Fourier coefficients for P(D)	
11-15	SWDT(3)	Fourier coefficients for P(D)	
16-20	SWDT(4)	Fourier coefficients for P(D)	
21-25	SWDT(5)	Fourier coefficients for P(D)	
26-30	SWDT(6)	Mean response of parameter $P(D/D)$	
•	•	Fourier coefficients	
•	•	Fourier coefficients	
•	٠	Fourier coefficients	
46-50	SWDT(10)	Fourier coefficients	
51-55	SWDT(11)	Mean response of parameter SS	
٠	0	Fourier coefficients	
٠	•	Fourier coefficients	
٠	•	Fourier coefficients	
71-75	SWDT(15)	Fourier coefficients	

PREGEN requires the following resources on a CDC 6400 computer; about 35k, 7 seconds compilation time and 7 seconds execution time for an interpolation run with 10 years simulated daily rainfall with yearly summaries only.

PHULLAM PHELEN

C

CCCC

00000000

CCCC

000000000000

CCCCCCCC

A

AT

ICINPUTOUITPUTOTAPES=INPUTOTAFE6=UNTPUT)

C						
	.ccccccccrrr	concorcerer (.conconcon	contractor	criticit	<u></u>
C						C
C						C
С	XX XX	XX XX	XX XX	XX XX	XX XX	XX XX C
С	** **	XX XX	XX XX	** **		хх хх С
С	****	* * * *	XXXX	XXXX	XXXX	XXXX C
C	*********	* ********	XAXXAXXAXX	XYAXXXXXAA	*****	XXXXXXXXXX C
С	*********	*********	*****	********	********	XXXXXXXXXX C
C	XXXX	* * * *	****	****	FXXX	XXXX C
С	XX XX	XX XX	AX XX	XX XX	да хх	XX XX C
С	XX XX	XX XX	XX XX	XX XX	XX XX	XX XX C
С						С
С						C
C	********	*********	*********	*******	******	XX XX C
C	********	*******	*******	*********	*****	XXX XX C
С	XX X1	K XX XX	**	XX XX	X X	XXX XX C
C	×x ×:	K XX XX	XX	XX	XX	XX X XX C
Ċ	XX X	K XX XX	XX	XX	×х	XX XX XX C
C	******	*********	*****	XX XXXA	XXXXXX	XX X XX C
C	********	********	*****	XX XXXAA	XXXXX	YX X XX C
č	A X	XX XX	x X	XX XX	XX	XX X XX C
c	XX	XX XX	X 4	XX AA	XX	XX XX XX C
č	XX	XX XX	XX	XX AA	XX	XX XXXX C
č	XX	XX XX	*********	XXXXXXXXXAA	*******	XX XXXX C
C C C	XX	XX XX		XXXXXXXX	*******	XX XXX C
č						c
č						c
č	XX XX	XX XX	** **	XX XX	XX XX	XX XX C
č	XX XX		XX XX	XX XX	XX XX	XX XX C
č	XXXX		XXXX	XXXX	XXXX	XXXX C
č	*******					
č	XXXXXXXXXXX		********	******		XXXXXXXXXXX C
č	XXXX	****	XXXX	XXXX	XXXX	XXXX C
č	XX XA			AK XX		
č	XX XX		XX XX			XX XX C
č	NG N					
•						6

C

C С

С c

C

С

С C

C С

C

C C С

CCC

Ċ CCC С

Č

С C

C

č C C

č

č CCCC

C

c

c

C С

C

C C

C C

THIS PROGRAM WAS WHITTEN BY BRAU J. GILBERT FOR USE WITH THE COMPUTER PROGRAM (RANGES) .

TO DETERMINE THE FOURIER CUEFFICIENTS BY PUPPOSE-MERELY INPUTING THE LATITUDE. LONGTINDE. AND FLEVATION OF ANY SITE IN THE WESTERN UNITED STATES FOR INPUT INTO THE PROGRAM (HANGES) .

PROGRAM UNITS-(MAIN PROGRAM) - PREGEN

(SUMMOUTINES) - HLOCK DATA (SUMMOUTINES) - APPLI: ULADAY, FORTI: MINUM, SEARCH, SER WAVGI: WAVG2

WAVG1. WAVG2 (USEN DFFINED FUNCTIONS) = ICUN (CONIROL DATA UTILITY FUNCTIONS) = MANF (EXTERNAL ANSI FUNCTIONS) = ALOG. CUS. SIN. SORT (INTRINSIC ANSI FUNCTIONS) = AHS. AMINI. FLOAT. IARS AHS CALCULATES THE ARSOLUTE VALUE FOR A REAL ARGUMENT. ALOG FINDS THE NATURAL LOGAPITHM OF A REAL VARIABLE. AMINI CHOOSES THE SMALLEST VALUE AMONG A LIST OF REAL ARGUMENTS. AS A RESULT THE NUMBER OF ARGUMENTS MUST BE TWO OF GREATER.

OR GREATER. COS IS THE TRIGONOMETRIC COSINE WITH THE ARGUMENT IN RADIANS AND TYPE REAL.

FLOAT IS A CONVERSION PRUCESS GIVEN AN INTEGER ARGUMENT

TO A REAL VAPIANLE. IARS FINDS THE AUSOLUTE VALUE FOR AN INTEGEP ARGUMENT. PANE IS A RANDOM NUMBER GENERATOR WHICH RETURNS VALUES UNIFURMLY DISTRIBUTED OVER THE INTERVAL BETWEEN ZERO AND ONE. INCLUDING ZERO BUT NOT INCLUDING ONE. A DUMMY WHICH ENDS UP HEING IGNORED. THE ARGUMENT IS JUST

SIN IS THE TRIGONOMETRIC SINE WITH THE APGUMENT IN RADIANS AND TYPE HEAL.

SORT CALCULATES THE SQUARE HONT OF ITS REAL ARGUMENT.

VARIABLE DEFINITIONS- SOME VARIABLE DEFINITIONS APPEAR HELUW WHILF OTHERS ARE DIFINID IN THE USERS GUIDE TO THE RANGES GRASSLAND SIMULATION MODEL (APPENDIX 5). EXCEPT FOR SOME SURSCRIPTS ALL DIF VARIABLES ARE DEFINED. A - REAL ARRAY CUNTAINING THE COSINE COEFFICIENTS FOOD THE CONTAINING THE COSINE COEFFICIENTS

- FOR THE FOUPIER SEPIES.
- REAL VARIABLE CONTAINING A TESTING VALUE ON ELEVATION SIMILARITY FOR TWO STALIONS EQUIDISTANT IN THE SUBHOUTINE (MINUM) .
 - REAL VARIANLE CONTAINTING THE SUMMATTON VALUE USED TO NUPMALIZE THE ARRAY (AV).

- HEAL ARMAN CONTAINING THE AVERAGE BETCHTING FACTORS. AV · HEAL VARIANTE CONTAINING THE REMINER GENERATION A I PANANE LER FOR A WIADEANT. н WHAL ANHAY CONTAINING THE STRE LUFFFICTERTS FUR THE EDUKIER SEFIES. REAL VANIANTE CONTAINT A TESTING VALUE ON H ELEVATION SIMILABILY FOR TWO STATIONS FOUIDISTANT IN THE SUBROUTINE (MINUM). - NEAL VANIANLE CONTAINING THE SUMMATTON VALUE USED TO NUMALIZE THE ARRAY (OPSU). OT INTEGEN VARIABLE CONTAINING THE EAST LONGITUDE LLON OF AFFA OF APPLICANILITY. - INTEGER VARIANIE CONTAINING HEANN CHARACTERS FOF FOR A PHIL OF FILL CHECK. - REAL VAMIANLE CONTINING THE SUMMATION VALUE USED TO NORMALIZE THE ARDAY (V). ET TER INTEGER VANTABLE CONTAINING AN ERHOR CODE FROM THE SUMMUTINE (FONTI). INTEGEN VARIANLE CONTAINING THE WHOLE NUMBER IT INTEGER VARIABLE (IVAR).
 INTEGER VARIABLE (IVAR).
 INTEGER VARIABLE (OVITATING THE FINAL FRACTIONAL PART OF THE VARIABLE (IVAR).
 INTEGER VARIABLE CONTAINING THE MINUTES OF DEGREES. 1 V IVAN - INTEGER VARIABLE CONTAINING THE NUMBER OF STATIONS WITHIN & TWO DEGREE HADINS IN FACH DUADHANT IN THE SUHROUTIME (MINUM) . KK INTEGER ARRAY CONTAINING THE NUMBER OF STATIONS WITHIN A IND DEGREE HAPINS IN EACH DUADHANT. - INTEGER VAPIABLE CONTAINING THE LATITUDE FACTOR FOR CALCULATING THE VAPIABLE (RSV). LLAT INTEGER ARRAY CONTAINING THE NUMBER OF STORMS LT IN ONE YEAR STUFFU BY WEEKS. INTEGER VARIABLE CONTAINING THE MAXIMUM ORDER м OF HARMONICS TO HE FITTED. INTEGER VARIABLE CONTAINING THE SIZE OF THE N INTERVAL USED TO DETERMINE THE FOURTER CUEFFICIENIS. NLAT INTEGER VARIABLE CONTAINING THE NORTH LATITUDE OF AREA OF APPLICABILITY. INTEGER VARIABLE CONTAINING A CUNSTANT FACTOR IN GENERATING THE FOURTER SENTES OF DATLY VALUES.
 INTEGER VARIABLE CONTAINING AN INDICATOR OF NT NK RAIN ON & CENTAIN DAY OR NO RAIN. P - REAL VARIABLE CONTAINING & CEPTAIN DAYS PROBABILITY FOR DAY DAY PRECEDED BY A DAY. REAL VARIABLE CONTAINING THE CUMULATIVE YEARLY PH PRECIPITATION. ULAT - PEAL APRAY CONTAINING THE MINIMUM LATITURES FOR EACH QUADRANT. QLEV REAL ANNAY CONTAINING THE MINIMUM ELEVATIONS FOR EACH QUADRANT. QLON REAL APHAY CONTAINING THE MINIMUM LONGITUDES FOR EACH QUADRANT. REAL VAMIABLE CUNTAINING THE MINIMUM DISTANCE FROM THE UNKNOWN STATION ID A KNOWN STATION IN OMIND ANY OF THE QUADRANTS. REAL VARIABLE CONTAINING THE MINIMUM ELEVATION FHOM THE UNKNOWN STATION TO A KNOWN STATION IN OMINL ANY OF THE QUADRANTS. INTEGEN MAIRIX CONTAINING THE MINIMUM STATION UNAM NAMES FOR EACH QUADRANT. OP - REAL VARIABLE CONTAINING THE YEARLY AVERAGE FOR PRECIPITATION ORSU - REAL APRAY CONTAINING THE STATION DISTANCES FOR EACH QUADHANT. OSVOI REAL APPRAY CONTAINING THE WEATHER GENERATION PANAMETEPS FOR EACH QUADPANT. REAL ARKAY CUNTAINING THE DAILY PPORABILITIES OF A DRY DAY PRECEDED RY A WET DAY. OT QUAD - REAL ARRAY CONTAINING FOURIER PARAMETERS FOR ONE QUADRANT IN THE SUBROUTINE (SFR). REAL ARHAY CONTAINING THE DAILY PROBABILITIES OF A DRY DAY PRECEDED BY A DRY DAY. 00 REAL ARRAY CONTAINING THE DAILY PROBABILITIES 91 OF A DRY DAY - UNCONDITIONAL. REAL VARIABLE CONTAINING A RANDUM NUMBER FROM THE RANDOM NUMBER GENERATUR USED TO DETERMINE IF A CONDITIONAL PROHABILITY IS HIGH ENOUGH OF USED TO GENERATE A EVENT SIZE. REAL VAHIABLE CONTAINING THE STATION DISTANCE RSO FROM A HADIUS OF INTERIST.
 REAL MATHIX CONTAINING THE FOURLER SERIES EXPANSION FOP PAPAMETERS OF FOUR OHADRAMTS. S INTEGER VANIABLE CONTAINING THE SOUTH LATITUDE SLAT INTEGER VARIABLE CONTAINING THE SOUTH LATITUDE OF AREA OF APPLICABILITY.
 REAL AMPAY CONTAINING THE SIZE OF WEEKLY STORMS.
 REAL VARIABLE CONTAINING A SUMMATION VALUE USED IN THE EXPANSION OF FOURTER SERIES.
 INTEGER ARRAY CONTAINING A HEADING FOR THE FOURTER COFFICIENTS FOR FID/D) + P(D) + AMD ETODM CIFE. ST7 SUN T STORM SIZE. TMP HEAL VARIABLE CONTAINING THE FRACTIONAL PART OF THE VARIANIL (IVAH). 15 - REAL ARRAY CONTAINING THE WEIGHIED MEAN FOURIER SEMIES EXPANSION.

REAL APPAY CONTAINING THE STATION ELEVATIONS

FOR EACH QUADHANI.

102

C

C

cc

c

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

Ċ

C

C

C

C

Ć

C

C

C

C

C

C

Ċ

C

C

Č

CC

C

C

C

C

C

C

C

C

C

C

C

C

c

C

C

C

C

C

C

C

C

C

C

C

C

CC

C

C

C

Ċ

C

C

č

C

č

C

CCCC

C

C

č

c

C

č

č

C

č

C

C

č

C

C

C

C

C

č

C

C

C

č

C

č

C

č

CC

CC

CCCC

CC

č

CC

Ć

č

CCCC

C

CC

0000

CC

Č

č

Č

CCCC

Č

č

CC

000000

CCCCCCCC

C

Ĉ

- INTEGER VARIABLE CONTAINING THE WEST LONGITUDE C wt chi OF AFFA OF APPLICABLE ITY. c c ¢ - HEAL ARRAY CONTAINING THE AMOUNT OF PRECIPITATION C a M WEEKLY IN A YEARS TIPE. C C HEAL VANIABLE CONTAINING THE REAL VALUE OF THE ¢ XI C VARIABLE SUBSCRIPT (1) USED IN THE FAPANSION OF Ċ C C THE FOURIER SERIES. C - PEAL ARRAY CONTAINING THE AVERAGE WEEKLY STORM SIZE REPRESENTED BY THE RECTURDED OF THE MEAN XL AM C č C C DALLY PRECIPITATION ON RAINY DAYS. C C VANIABLE CONTAINING THE LATITUDE OF THE C ALAT - HEAL MINIMUM STATION IN THE SUBROUTINE (MINUM). - REAL VARIABLE CONTAINING THE FLEVATION OF THE CCCC C ALF V C MINIMUM STATION IN THE SUCHDUTINE (MINUM). C Ċ - HEAL VARIABLE CONTAINING THE LONGTIONE OF THE AL ON С MINIMUM STATION IN THE SUBROUTINE (MINIM). - REAL VARIABLE CONTAINING THE PEAL VALUE OF THE VARIABLE SUBSCRIPT (N) USED IN THE EXPANSION OF С C C C XN C THE FOURIER SERIES. С - INFEGER VARIABLE CONTAINING THE FIRST FOUR CHARACTERS FOR THE NAME OF THE MINIMUM STATION С XN6M1 С С IN THE SUBRULLINE (MINUM) . - INTEGER VARIABLE CONTAINING THE SECOND FOUR C C č ANAME CHARACTERS FOR THE NAME OF THE MINIMUM STATION C IN THE SUBROUTINE (MINIM). - INTEGER VANTABLE CONTAINING THE LAST TWO CHARACTERS FOR THE NAME OF THE MINIMUM STATION IN THE SUBROUTINE (MINUM). С C C ANAMS С C С - REAL VARIABLE CONTAINING THE MINIMUM STATION DISTANCE FRUM A RADIUS OF INTEREST. С ARSU C С 21 - REAL VARIABLE CONTAINING THE PRECIPITATION FOR С č A JULIAN DAY. C С CCC PROGRAM FILES-TAPES REFERENCED BY THE VARIABLE (READU) С IS EQUIVALENCED TO THE INPUT FILE, IN THIS CASE THE CAND С C NEAUER. TAPES REFERENCED BY THE VANIABLE (WRITEU) IS EQUIVALENCED TO THE OUTPUT FILLS IN THIS CASE THE LINE PHINIER. THESE TWO VARIABLES ARE INITIALIZED IN THE BLOCK С C C С C Ċ č DATA SURPHOGRAM. С С C C FORMAT OF DATA ENTRIES ARE ALSU DESCRIPED IN THE USERS. С GUIDE TO THE HANGES GRASSLAND STMULATION MUDEL (APPENDIX 5). C С С C C HEMARKS-ALL CODING WITHIN THIS PRUGRAM HAS REEN C CONFURMED TO STANDARD ANSI FORTRAN EXCEPT THE FOLLOWING С C С ITEMS. THE PROGRAM CARD WHICH IS A PART OF SCOPE 3.3 OPERATING ITEMS, THE PROGRAM CARD WHICH IS A PART OF SCOPE 3.3 OPERATING SYSTEM, SOME MIXED MODE UPERATIONS, AND THE COC PANDOM NUMBER UTILITY FUNCTION (HANF) REFERENCED IN THE SURPOUTINE (HEADAY) LINES 97 AND 113, NOTE THAT USE OF THE FUNCTION (RANF) GIVES DIFFERENT OUTPUT FROM THE SURROUTINE (HEADAY) FOR DIFFERENT SYSTEMS AND OIFFERENT RUNS. SURROUTINE (FORIT) WAS EXTRACTED FROM THE TRM SCIENTIFIC SURROUTINE PACKAGE WITH THE ADDITION OF TWO WRITE STATEMENTS. C č С C C C C č C C INTEGER I.I.J.ILAT.ILEV.ILON.IU.IP.IS.II.IY.IZ.II.I2 PRGENOOI INTEGER 13.14.J.K.K.M.MJ.MOR.NE.NN.NS.NW.QNAM PRGEN002 INTEGER READU WRITEU PRGENOOS REAL A+AA+AV+H+HH+QLAT+QLEV+GLON+URSG+GSWOT+HSG+S+TS PRGEN004 С PRGE NO05 COMMON /FILES/ PEADU+WRITEU PRGENOOA COMMON /VIPS/ A(3) +AA(9) +AV(5) +A(3) +BA(9) +ILAI +II FV+ILON + IO + IP + IS +PRGF N007 11T+1Y+1Z+KK (5)+NF+NN+NS+NW+OLAT (5)+OLEV (5)+QLUN (5)+ONAM (5+3)+DR5Q (PRGENODA 25) + QSWU1 (75) + RSU+ S (365+4) + TS (365) PRGENOOS C PRGEN010 READ (READUATE) ILATAILONAILEVAISAIPAIYAINAIZAIT PRGEN011 IF (IT-NE-1) GO TO 101 PPGEN012 С PRGEN013 С . FOURIER COEFFICIENTS READ AS INPUT AND PRECIPITATION PRGENOIA С AMOUNTS GENERATED. PRGENOIS Ċ PRGEN016 CALL BEADAY GO TU 110 PRGENO17 PRGENOIA С PRGFN019 С • • • • • INTERPOLATION PROCESS FOR THE FOURIER COFFFICIENTS. PRGEN020 С PRGENO21 101 DO 102 1=1.5 PRGEN022 QLA1(1)=0. PRGEN023 QL(N(I)=0.PRGEN024 QLEV(I)=0. PRGEN025 102 OR50(1)=0. PRGEN026 00 103 J=1+75 PRGFN027 QSWUI (J)=0. PRGE NOZA 103 CONTINUE PRGENO29 CCC PRGE NO 30 CONVERT MINUTES OF DEGREES TO FRACTIONS OF DEGREES. . PPGF NO31 PRGEN032 ILAI=ICUN(ILAT) PPGF NO33 ILON=ICUN(ILON) PRGEN034 WRITE (#WITEU+112) WRITE (#WITEU+113) 1.41+ILON+ILEV

PRGEN035 PRGEN036

C PROF NO 37 . SEARCH TAPE AND FIND AT MOST & STATIONS IN QUADRANTS PHOL NO 3A C PHGE NA 39 C SURRUUNDING USER LOCATION. C DUGE NALA CALL SEARCH PRGEN041 С PHGENA42 Ċ . . IF UNKNOWN LIES IN AREA OF APPLICAPILITY OF A KNOWN PHGE NO43 STATION WHITE MESSAGE AND CHECK FOR CHEFFICIENT USAGE. PHGE NALL C PRGEN045 1.6 (KK(5)) 10H+10H+104 PRGE NO46 104 #HITE (ANITEU.114) (UNAM(5.0).J=1.3).0LAT(5).ULOM(5).QLEV(5) PRGENA47 ##11t (##17EU+122) 00 105 1=61+71+5 DOGL NOLD PHGE NA49 11=1+1 PRGENOSO 12=1+2 PRGEN051 13=1+3 PRGENOS2 14=1 •4 PHGE NOS3 WHILE (WHITEU-123) USWDT(1)+USWDT(11)+DSWDT(13)+DSWDT(12)+DSWDTPRGEN054 (14) PRGE NOSS ĩ 105 CUNTINUE PRGENOSO IF (1Z.LU.1) 60 TO 108 PHGE NOST K=U PRGENOSA 00 107 1=1./.3 PPGL NASO A+ I=M PRGEN060 M60=1+60 PRGENOGI AA(1)=05#07(M60) PRGEN062 HH(I)=0. PHGEN063 5.1=L 6v1 00 PRGE NO64 1J=1+J PRGENO65 MJ=M+J+60 PRGENDAR AA(IJ)=USWDT(MJ) PRGEND67 MJ=MJ+2 PRGENOGA HH (IJ) = USWDT (MJ) PRGENO69 PRGEN070 106 CONTINUE K=K+5 PRGEN071 107 CONTINUE PRGENA72 CALL HEADAY PRGE NO73 STUP PRGEN074 IOR WHITE (WHITEU.115) PRGEN075 WRITE (WRITEU:116) PRGEN076 DU 109 1=1.4 PHGEN077 WHITE (WHITEU.117) I. (ONAM(I.J). J=1.3). OLAT(I). OLON(I). OLFV(I) PHOFMO78 109 CONTINUE PRGEN079 WRITE (WRITEU+11A) PRGENOBO WRITE (WRITEU+)19) (KK(I)+1=1+4) PRGENOAL С PRGENOR2 C • • • • • • COMPUTE WEIGHTING FACTURS. PRGENO83 PRGE NOR4 CALL WAVGT PRGENORS WRITE (WRITEU+120) PRGENORG WRITE (WRITEU.121) (AV(I).1=).4) PRGENOR7 C PRGENOAA C . COMPUTE NEW FOURIER COEFFICIENTS TO PASS TO RANGES PRGENOAS C SIMULATION. PRGEN090 С PRGEN091 CALL WAVES PRGEN092 IF (1P.LO.1) CALL READAY PRGENOGS 110 STOP PRGE NO94 C C PRGE NO95 FORMATS USED IN THIS PROGRAM. PRGEN096 C PRGEN097 111 FORMAT (315+613) PRGE NO 98

 111 FORMAT (315:613)
 PRGEN098

 112 FORMAT (1H1:20X:39HC00HDINATES OF POINT TO BE INTERPOLATED:52H ALLPRGEN099
 PRGEN098

 1 LAT AND LON IN DEGREES AND HUNDRETHS OF DEGREES)
 PRGEN009

 113 FORMAT (1H0:20X:6HLAT = +110:5X;6HLON = +110:5X;7HELEV = +110/)
 PRGEN00

 114 FORMAT (1H0:40HUNKNOWN LIES)N AREA OF APPLICABLITY OF STATION *2PRGEN103
 PRGEN103

 115 FORMAT (1H0:40HUNKNOWN LIES)N AREA OF APPLICABLITY OF STATION *2PRGEN103
 PRGEN104

 115 FORMAT (1H0:40X:33HMINIMUM STATIONS IN EACH WUADPANT)
 PRGEN104

 116 FORMAT (140+12X+440)AD:12X+440AAC+17X+34LAT+17X+34LON+16X+44ELEV) PRGENIOS 117 FORMAT (14X+11+10X+2A4+A2+3()0X+F10+4)) PRGENIOS 118 FORMAT (140+10X+7640044) F STATIONS WITHIN 2 OEGREE RADIUS (APPRPRGENIO7 PHGEN105 PRGENIOR 10%. 120 MI.) IN EACH QUADHANT) PRGENIOR 119 FORMAT (1H0.2HKK.4(5x.110)) 120 FORMAT ()H ///IRH WFIGHTING FACTORS) 121 FORMAT (1H0.4(107.F10.4)) PRGEN109 PRGENI10 PRGENITI 122 FORMAT (1/74H FOURTER COEFFICIENTS FOR CLIMATIC PARAMETERS.P (D/D) .PRGENIL? 1 P(D) . AND STURM SIZE/17X.2HAU.1BX.2HAI.1BX.CHBI.1AX.2HA2.1BX.2HB2PRGENII3 Z) PRGEN114 123 FORMAT (1H0.5(10X.F10.4)) PRGENIIS C PRGENIIG END PRGENI17

	SUHROUTINE APPLI (LAT.LON.LEV.NAME.SWDT)	APPL 1001
С		APPLI002
C	 • • • • THIS SURROUTINE (APPLI) IS CALLED BY THE SUPROUTINE	APPLI003
C	(SEARCH) TO DETERMINE IF AN UNKNOWN STATION LIFS WITHIN AN	APPL 1004
C	AREA OF APPLICATILITY (ACCORDING TO HEEPMAN ET. AL. AND	APPLIOOS
č	GIFFUPD ET AL.). IF IT LIES IN MORE THAN UNE REGION THE	APPL TOOS
č	CLOSEST STATION IS CHOSEN.	APPLI007
č	officers, publiculty supplies	APPLICOR
	INTEGEN FLONAILATAILEVALUNAIOAIPAISAITAIYAIZAKKALAT	APPL 1009
	INTEGEN LEVOLUNONAME (3) ONE ONLATONNONSONDOQUADOSLAT	APPL 1010
	INTEGER WIDN	APP1 1011
	PEAL ACAAAAAVARAHRADIATAULEVADLONADPSQAUSWDTAMSD	APPL 1012
	REAL SASWOT (15) ATS	APPL IOI3
С	MEME 3434(1)(13/4/3	APPL 1014
	COMMON /VIPS/ A(3) + AA(4) + AV(5) + H(3) + HA(9) + ILAI + ILFV+ ILON+ IO+ IP+ IS	
	111+14+12+FK (5) +NF+NN+NS+N++0LA1 (5) +DLFV (5) +QLUN(5) +QNAM (5+3) +QRSG	
	25) • 45 w 1 (75) • 45 (365 • 4) •) 5 (365)	APPL 1017
С	100111111111111111111111111111111111111	APPL TOTA
c	DETERMINE LATITUDE AND LONGITUUE OF AREA OF APPLICATIONTY	
c	Drivering Latitude and Longinue of any a of articlarities	APPLI020
C	ELUN=LUN-NF/59.*100	APPL 1020
	WLUN=LUI+NW/59.0100	APPL 1023
	NLAT=LAI+NN/78.*100	APPL 1023
	SLAT=LAI=NS/78.º100	APPL 1024
C		APPL 1025
C	• • • • SEE IF UNKNOWN STATION IN AREA OF APPLICABILITY.	APPL 1026
C		APPL1027
	1F (ILUN.LT.ELON.OF.ILUN.GT.WUN) GO TU IOI	APPI 1028
	IF (ILAI.LT.SLAT.OR.ILAI.GT.NLAT) GO TU 101	APPL 1029
	CALL MINUM (KK(5)+LAT+LUN+LEV+ILEV+DLAT(5)+OLUN(5)+DLEV(5)+OSWDT(
	11) • 5WD I • H 50 • QRSU (5) • NAME • UNAM (5 • 1) • QNAM (5 • 7) • UNAM (5 • 3))	APPL 1031
÷	L RETURN	APPLI032
C		APPL 1033
	END	APPL1034

SUBMULTINE PEADAY READYOOT С RF ADYOO? Č THIS SUPPORTINE (HEADAY) IS CALLED BY THE MAIN PROGRAM HE ADYOO T CC (PHEGEN) AND THE SUMMOUTINE (SEARCH) TO SIMULATE DATLY READY004 PRELIPITATION. READYOOS č READYOOA INIFGER INTLATOILEVALUATION INATAINATAANATAAN BEADY007 INTEGEN NKOLOET (52) OMONEONNONKONSONWODE ANUOT (9) INTEGEN IITLE (3) OMUIIEU READYOOR RF ALLYODS HEAL ASAASAVSHSHHSPSPHSULATOULEVSULONSUNAMOUPSOHSU READYOLD HEAL USWUT . UI (165) . U. (365) . 41 (365) . P. + 50 . 5. 512 (52) . TS HEADYOLL REAL WM(57) + XLAM(365) + /1 HEADY012 C READYOL3 CUMMON /FILES/ HEADU. WHITEU READY014 (DMMUN /VIPS/ 4(3) +A4(4) +AV(5) +H(3) +H(4) +TLAT +ILEV ILON IO + IP + IS - REALYOIS 111+1Y+12+KK (5)+NF+N/2+NS+NK+ULAT (5)+OLEV (5)+OLUN (5)+ONAM (5+3)+ORSU (HEADYOTA 25) + 45W11 (75) + P50+ S (365+4) + 15 (365) READY017 C READYOLA READYOLS DATA T(1)/4HP(0)/.T(2)/4H 1.1(3)/2H / UATA T (4) /4HP (0//+1(5)/4HD) /+T(6)/2H READY020 DATA T1/)/4HSTUR/+T(8)/4HM S1/+T(9)/2HZE/ READY021 C BEADY022 K = 0READY023 IF (11.44F.1) 60 TO 102 HEAD (HEAD(+115) (TITLE(1)+1=1.3) BEALY024 BEADY025 WRITE (#RITEU+116) (TITLE(1)+I=1+3) BEADY026 С READY027 . READ FOURIER COEFFICIENTS FOR NO.01.55. PARAMETERS C READYOZR HEAU IN FOLLOWING ONDER. P(U/U) .P(D) .SS. C BEADY029 C BEADY030 DO 101 L=1.7.3 BEADY031 K=K+1 READY032 K1=K+1 READYONS K2=K+2 READY034 С PEADY035 REORDER AA AND BE TO BE COMPATIBLE WITH PANGES AND PREGEN, REALYO36 C č READY037 1=7 READYO3A IF (L.ED.1) I=4 IF (L.ED.4) I=1 READY039 READY040 READ (READU.117) AA(I) READY041 WRITE (WRITEU.114) T(K).T(K1).T(K2).AA(1) READY042 J≓1 READY043 IJ=I+J READYOLL 111=11+1 READY045 READ (READU+119) AA(1J)+HB(IJ)+AA(IJ)+RB(1J) WRIIE (WRITEU+120) AA(1J)+BB(IJ)+AA(IJ)+BB(IJ) READY046 BEADY047 101 CONTINUE READY048 102 CALL SER (2. AA(1). AA(2). 86(2).01) BEADY049 CALL SER (2+AA(4)+HR(5)+HR(5)+(0)) CALL SER (2+AA(7)+AA(6)+BR(8)+XLAM) BEADY050 BEADYOSI OT(1)=(41(1)-01(365)*00(1))/(1.-01(365)) BEADY052 CCC BEADY053 • • • • • CALCULATE P(D/W) FROM P(D) AND P(O/D) WHICH WFRE DERIVED FROM DATA. HEADY054 READY055 Č READYOSA 00 103 M=2,365 BEADY057 $103 \ Q1(K) = (Q1(K) - Q1(K-1) + 00(K)) / (1 - Q1(K-1))$ BEADY058 С BEADY059 C DATA DRY/DRY SU CONVERT TO WET/ORY. BEADY060 С BEADY061 00 104 N=1.365 READY062 Q0(K)=1.-00(K) BEADYN63 С BEADY064 C DATA DRY/WET CONVERT TO WET/WET. BEADY065 BEADY066 Q1(K)=1.-QT(K) BEADYN67 104 CONTINUE READY068 C BEADY069 Č • • • • • • INITIALIZE. BEADY070 Č BEADY071 DO 105 M=1.365 BEADY072 C C BEADY073 . . XLAM IS IN FACT THE RECIPROCAL OF THE MEAN DAILY PEADY074 Ċ PRECIPITATION ON PAINY DAYS. BEADY075 C BEADY076 $XLAM(M) = 1 \cdot / XLAM(M)$ READY077 105 CONTINUE READY078 P=00(1) BEADY079 J=1 BEADYORO DO 106 M=1.52 READYORI LT (M) =0 BEADY082 106 WM (M)=0. BEADY083 UP=0. BEADY084 С READYORS С YFAH LOOP. READYORG C BEADY087 00 112 J=1.1Y READYOAA PR=0. BE ALLYORS ¢ BEADY090 C PAY LOOP. READYDOI C READY092 00 111 1=1.365 PEADY093 С BF ALLYO 96 C GENERATE UNIFORM MANDOM VARIABLE. BEADY095

READY096 С AE ADY097 H=HANF (1.0) READYNAR IF (P.1F.P) GO TO 107 HE ADYONG C HEADY100 C NHELE THEN NO HAIN OF DAY I. . С HEADYTO1 RF ADY 102 NH=1 PFADY103 /1=0. RF ADVIOS OU TO IOH С READY105 NRECT THEN PAIN OCCUPRED. READYIOS C C HFADY107 READYIOR 107 NK=2 C C PEADYING TRANSFORM UNIFORM RANDUM VARIABLE (R) . TO EXPONENTIAL READVIIO С DISTHINUTION WITH PARAMETER LAMADA. READYIII READY112 C H=HANF (1.0) READY113 READY114 21=-ALOG(#)/XLAM(1) Iw = (1 - 1)/7 + 1READY115 BEADY116 1+ (IW.GE.53) GO TO 108 BEADY117 C С . COMPUTE FREQUENCY OF STORMS AND AMOUNTS OF READYIIR C PHECIPITATION. RFADY119 C BEADY120 Li(IW)=LT(IW)+1 READYIZI wri(IW)=WM(IW)+ZI READY122 READY123 C С CUMULATIVE YEAHLY HAINFALL. READY124 . . C BEADY125 1F (10.F0.1) WRITE (WRITEU+121) 1+71 BEADY126 108 READY127 PH=PR+Z1 IF (NR-1) 109+109+110 BEADY128 109 P=QO(1)READY129 60 TO 111 READY130 P=01(1) 110 READY131 111 CUNIINUE READY132 WHITE (WRITEU+122) J+PR UP=UP+PR BEADY133 READY134 112 CONTINUE BEADY135 UP=UP/IT READY136 WRITE (WRITEU+123) IY+0P READY137 WRITE (WPITEU+124) BEADY138 00 114 L=1.52 BEADY139 IF (LT(L).NE.D) GO TO 113 BEADY140 ST7(L)=0. BEADYI4I GO 10 114 STZ(L)=WM(L)/LT(L) READY142 BEAOY143 113 114 WM(L)=WM(L)/IY BEADY144 WRITE (#HITEU+125) (L+LT(L)+WH(L)+STZ(L)+L=1+52) BEADY145 RETURN BEADY146 C READY147 FORMATS USED IN THIS SUBROUTINE. BEADY148 С С BEADY149 115 FORMAT (244.42) BEADY150 116 FORMAT (1H1+10X+2A4+A2) BEADY151 117 FORMAT (3X.F10.0) BEADY152 118 FORMAT (1H0+10X+18HFOURIER COFF FOR +2A4+A2+10X+5HA1 = +F10+4) READY153 119 FORMAT (4F10.0) READY154 120 FORMAT (12(4X.F6.4)) READY155 121 FORMAT (10X.11HJULIAN DAY .13.2X.9HPRFCIP = .FI0.4) READY156 122 FURMAT (1H0+10X+20HCUM YEAHLY PHECIP YK+13+3H = +F10+4) BEADY 157 123 FORMAT (IHO.10X.19HYFARLY AVEMAGE FOR .13.1X. HYFARS = .FI0.4) READYISA 124 FORMAT (1H1+10X+SHWK NU+10X+9HNO STORMS+10X+16HMFAN WKLY PRECIP+10READY159 1X . BHSTORM SZ) BEAOY160 125 FORMAT (11X+15+12X+15+12X+F16+4+10X+FA+4) BEADY161 С READY162 END READY163 HLUCK DATA BLKDT001 С BLKDT002 С . . THIS SUMPROGRAM (ELUCK DATA) ENTERS MATA VALUES INTO BLKDT003 C LAHELED COMMON BLOCKS PRICE TO PROGRAM EXECUTION. ALKOT004 C BLKDT005 INTEGER ILATOILEVOILONOIDOIPOISOITOIYOIZOKKONEONN BLKDTOOK INTEGEN NSONWOUNAMOHFAULIOWHITEL HI KDT007 REAL A+AA+AV+R+HH+OLAT+QLEV+OLON+ORSO+USWDT+HSO+S+TS ALKOTOOA C BLKDT009 COMMON /FILES/ READU+WRITEU ALKDT010 COMMON /VIPS/ A(3) . AA(4) . AV(5) . H(3) . HB(9) . TLAT . TLEV. TLON. TO. IP. IS. BLKDTOIT 111.1Y.12.KK(5).NF.NN.NS.NW.ULAT(5).OLFV(5).QLUN(5).QNAM(5.3).QRSQ(ALKDT012 25) + Q5WD1 (75) + HSU+ S (365+4) + 15 (365) **BLKOTOI3** C ALKDTOI4 DATA ONAM (1+1) /4H 1.0NAM(1.2)/4H 1+0NAM(1+3)/2H / BLKDT015 DATA ONAM (2,1)/4H 1.0NAM (2.2)/4H 1.0NAM (2.3) /2H / BLKDT016 UATA ONAM (3.1)/44 1.01NAM (3.2)/4H 1. UNAM (3.3) /2H / BLKDT017 DATA UNAM (4+1)/4H 1. ONAN (4.2) /4H 1.UNAM (4.3) /2H ALKDT018 DATA QHAM (5.1)/4H 1.UNAM(5.2)/4H 1. UNAM (5.3) /2H BLKDT019 DATA PLADU/5/ WRITEU/6/ BLKDT020

BLKDT021

BLKDT022

С

END

	SURRULTINE FURIT (FNIONOMODOMOTIN)	FORITOOI
С		FONITO02
С	 • • • THE STREET STREET FEILETER (FILE) IN CALLED BY LOD STREETTINE	FOWITCOR
С	(WAVOR) 11 SAN TAREN CONFECTLY FROM THE IOM SCIENTIFIC	FUNITOON
С	SUBBUTINE PACKAGE .	FORITODS
C		FORTTOOR
C	SUBRUULENE FOR 11	FORITOU7
С		FORITOOR
С	PURPOSE	FORITADO
С	FOUMLER ANALYSIS OF A PERIODICALLY LANDLALED FUNCTION.	FUNITOIO
С	CUMPUTES THE CONFETCIENTS OF THE DESTRED NUMBER OF TERMS	FORITOLI
С	IN THE FULL IFR SENTES F (A) = A (O) + SUM (A (K) CUS A X + H (K) STA KX)	SINTINO?
С	WHIFH RELOPED A TIMPHUALMAIF A UTVEN STI OF	FORITO13
С	PERIVALLY TANULATED VALUES OF A FUNCTION.	FORITOIA
C		FORITOIS
С	USAGE	FORITOIA
С	CALL FURIT(FNT+N+M+A+H+IE+)	FORITO17
С		FORITOIA
Ċ	DESCRIPTION OF PARAMETERS	FORITOIS
C	ENT-VECTOR OF TARULATED FUNCTION VALUES OF LENGTH 2N+1	FORTTO20
С	N -UFFINES THE INTERVAL SUCH THAT 2NOT PUINTS ARE TAKEN	FONITO21
C	UVER THE INTERVAL (U. 201). THE SPACINU IS THUS 201/20+1	FORITOZZ
C	M -MAXIMUM ORLER OF HARMONICS TO BE FITTED	FORTTO23
C	A -NESULTANT VECTOR OF FURNITH COSINE COLFFICIENTS OF	FOHITO24
C	LENGIH M+1	FORTTO25
С	A SUH DO A SUH LOAD A SUM M	FORITO26
C	H - MESULTANT VECTOR OF FOURTER SINE COFFFICIENTS OF	FORITO27
C	LENGTH M+1	FORITO2A
CC	d Sub Oo R Sub loose R Sub M	FOHIT029
С	IER-RESULTANT ERROR CODE WHERE	FORITOSO
C	ILK=0 NO FRROP	FOHITOJI
C	ILH=1 N NUT GREATEN OR EQUAL TO M	FORITO32
С	ILH=2 M LESS THAN ()	FORITO33
С		FORTTO34
С	REMARKS	FORITOSS
С	M MUST RE GREATER THAN OF EQUAL TO ZERO	FORIT036
С	N MUST RE GREATEN THAN OR FOULT TO M	FORIT037
С	THE FIRST ELEMENT OF VECTOR & IS ZERO IN ALL CASES	FORITOJA
С	68	FORIT039
С	SUBROUTINES AND FUNCTION SURPROGRAMS REQUIRED	FORIT040
С	NUNE	FORITO41
C		FORIT042
С	METHOD .	FORITO43
С	USES PECURSIVE TECHNIQUE DESCRIPTD IN A. RALSTON. H. WILF.	FORIT044
С	MATHEMATICAL METHODS FOR DIGITAL COMPUTERS . JOHN WILFY	FORTTO45
c	AND SONS . NEW YORK . 1960 . CHAPTER 24. THE METHOD OF INDEXING	FORTTOAS
С	THRUUGH THE PROCEDURE HAS HEEN MODIFIED TO SIMPLIFY THE	FORIT047
С	CUMPUTATION.	FORITO48
С		FORTTOAS
С		FORITOSO
	INTEGEN READUNARITEU	FORITOSI
	DIMENSIUN A(3) + H(3) + HNT(365)	FORITO52
	COMMUN /FILES/ READU WHITEU	FORIT053

1

[

 2		FORIT054
	• • • • CHFCK FOH ERHOH PARAMETEH ERHORS.	FORITOSS
 C		FORITOSA
	1ER=0	FORI1057
	1F (M) 101+102+102	FORITOSA
101	164=2	FORIT059
	WRITE (WHITEU+110)	FORIT060
	RETURN	FORIT061
102	IF (M~N) 104+104+103	FORIT062
103	16R=1	FORIT063
	WRITE (WRITEU+111)	FOR1T064
	HETURN	FORITO65
 C		FOPITO66
Č	• • • COMPUTE AND PRESET CONSTANTS.	FORITO67
c		FORITOGR
	AN≏N	FORITO69
104	COEF=2.U/(2.0*AN+).0)	FORIT070
	CUNST=3+141593°COEF	FORIT071
	S1=SIN (LONST)	FORITO72
	C1=CUS(LONST)	
		FOR1T073
	C=1.0	FORIT074
	S=0.0	FORIT075
		FORIT076
	FNTZ=FNI(1)	FORIT077
105	U2=0.0	FORITORA
	U1=0.0	FORIT079
	I=2*N+1	FORITORO
C		FORITOPI
с	• • • • FORM FOURIER CUEFFICIENTS RECHESIVELY.	FORITORS
 C		FORITORS
106	U0=FNT(1)+2+0*C*U1=U2	FORITOR4
	02=01	FORITOBS
	U1=U0	FORITOPA
) = I - 1	FORITOR7
	IF (1-1) 107+107+106	FORITOPA
107	A(J) = CUt f + (FNTZ + C + U) = UZ)	FORITORS
	£ (J) ≂C() ∈ F ≤ S ≤ U]	FORITOSO
	1F (J=(M+1)) 108+109+109	FORITOPI
108	0=C1+C-51+5	FORITO92
	S#C1#5+51*C	FORIT093
	C=U	FORIT094
	(*) { *}	FORITOSS
	GO TO 105	
100	A(1)=A(1)=n.5	FOR11096
104	RETURN	FORITO97
 0	REIDEN	FORITOPA
P		FOR11099
	• • • • FORMATS USED IN THIS SUBPOUTINE.	FORITION
 -		FORITIOI
110	FORMAT (90HUIN FORIT - M MUST HE GREATEN THAN OF EQUAL TO ZERD.	
	TEASE CHECK SITUATION NOT SATISFIED.)	FORIT103
111	FURMAT GATHUIN FORTE - N HUST HE GREATEN THAN OR FOUAL TO M. PLE	
	1E CHECK SITUATION NOT SATISFIED,)	FOR11165
 C		FORITISS
	E feta	FOH1117

FUNCTION ICON(IVAR)	ICON 001
C	ICON 002
C THIS FUNCTION (ICON) IS CALLED BY THE MAIN PROGRAM	ICON 003
C (PREGEN) AND THE SUBROUTINE (SEARCH) TO CUNVERT MINUTES OF	ICON 004
C DEGREES TO FRACTIONS OF DEGREES.	ICON 005
C	ICON 006
INTEGER IT, IV, IVAR	ICON 007
REAL TMP	ICON DOR
c	ICON 009
IT=1VAH/100	ICON 010
1T=IT+100	ICON 011
TMP=1VAK-IT	ICON 012
IV=(THP/60,)*100,	ICON 013
1CON=II+IV	ICON 014
RETURN	ICON 015
c	ICON 016
END	1CON 017

.

			SUHROUTINE MINUM (K.LAT.LUN.LEV.ILEV.XLAT.XLUN.XLEV.QSWDT.SWDT.RS	QM1NUM001
		1	+ 1454 + NAMF + 1NAM] + 1NAM2 + 1NAM3)	MINUMOOZ
	C			MINUM003
	C		• • • THIS SUBROUTINE (MINUM) IS CALLED BY THE SUBROUTINES	MINUM004
	C		(SFANCH AND APPEL) TO FIND THE CLOSEST STATION TO THE POINT	MINUMOOS
	C		OF INTEREST. IT IS CALLED FOR FACH QUARPANT.	MINUMOOR
	C			MINUM007
			INTEGER ILEVOKOLAIOLEVOLONOMOMIONZOMZOMZONZONAME(3)	MINUMOOA
			INTEGER XNAMI, XNAM2, XNAM3	MINUM009
			REAL A+B+QSWDT(15)+RSQ+SWDT(15)+XLAT+XLEV+XLUN+XRSQ	MINUMO10
	C			MINUM011
			IF (K.EW.0) GO TO 102	MINUM012
			IF (HSU-XHSU) 102+101+104	MINUM013
	C			MINUM014
	C		• • • • IF TWO STATIONS ARE FOULDISTANT THEN THE CLOSEST DEPENDS	MINUM015
	C		ON ELEVATION SIMILARITY.	MINUM016
	C			MINUMO17
		101	$A = AUS(I \perp Oat(1 \perp Fv) - x \perp Fv)$	MINUMOIA
			B=1A8S(1LFV-LEV)	MINUM019
			IF (A-0) 104+104+102	MINUMO20
	C			MINUM021
	C		• • • • SAVE THE LATITUDE. LONGITUDE, ELEVATION AND FOURIER	S20MINIM
	C		PARAMETERS OF THE MINIMUM STATION.	ESOMUNIM
	C			MINUM024
		102	XLATELPI	MINUM025
			XLUN=LUN	MINUM026
			XLEV=LEV	MINUM027
			XRS0=PSu	MINUM028
			XNAM]=NAME(1)	MINUM029
			XNAM2=NAME (2)	MINUM030
			ANAM3=NAME (3)	MINUM031
			K=K +]	M1NUM032
	C			MINUM033
	C		• • • PEORDER THE APRAY (QSWDT) TO SATISFY THE REQUIREMENTS	MINUM034
	C		FOR THE SUBROUTINE (SER).	MINUM035
	C			MINUM036
			00 103 M=1+11+5	MINUM037
			M]=M+]	MINUM038
			M2=M+2	MINUMA39
			M3=M+3	MINUM040
				MINUM041
			QSWD1(M)=SWDT(M)+.001	MINUM042
			QSWDF(M1)=SVDT(M1)+.001	MINUM043
			QSWD1(M2)=SWD1(M3) <.001	MINUMA44
			QSWD1(M3)=SWD1(M2)*.001	MINUM045
		102	QSWD1(M4)=SWD1(M4)*.001 CONTINUE	MINUM046
			RETURN	MINUM047
	C	104	PC I VAR	MINUM048
	¥.		END	MINUM049
			5 TV	MINUM050

SUBBOUT INF. SEARCH SENCHIO1 SENCHOO2 C SENCHOO3 . THIS SUBSIDITINE (SEARCH) IS CALLED BY THE MAIN PROGRAM С . . (PELGEN) TO SEARCH THE DATA FILE TO SEE IF AN UNRANOWS STATION LIES IN THE WEA US APPLICAMILITY OF ANY KNOWN STATIONS AND SENCHOO4 C cc SENCHOOS WHICH STATIONS AND THE CLOSEST IN FACH OF FOUR QUADRANTS. SERCHOOM C SERCHOO7 INTEGER FOF+1+1J+1LAT+1LEV+1LUH+10+TP+15+1T+17+17 SERCHOOR INIFORM II. 12+12+14+J+K+K+LAT+LEV+LAI+LON+M+MJ SERCHOOS INIF GER NAME (3) . NE . NILONS . NW. UNAM. READLIG #RITEU SERCHO10 SERCH011 REAL ASAASAVSHSHRSDI AT OLEVSOLUNSURSUSWOTSHSD MEAL S+SWOT(15)+15 SERCHO12 C SERCH013 COMMON /FILES/ READU SERCHO14 COMMON /VIPS/ A(3) +AA(4) +AV(5) +H(3) +HA(4) +TLAI+11 FV+TI ON+TO+IF+IS+SERCHAIS 111+14+14+KK (5)+NF+NU+NS+NF+ULAT (5)+OLFV (5)+OLAN (5)+ONAM (5+3)+ORSO (SERCHOIA 25) + US&U1 (75) + RSU+ S (365+4) + 15 (365) SERCH017 SENCHOIA С UATA FUT /4H SENCHOIG SENCH020 С 00 101 1=1.5 SERCH021 SERCH022 101 KK(I)=0 IF (15.LO.1) WRITE (WRITEU+121) SERCH023 SENCH024 C C . READ LATITUDE. LONGITUDE. FLEVATION AND WEATHER SENCH025 SENCH026 PARAMETERS FOR UNE STATION. C С SERCH027 SEPCHAZA 102 HEAD (MEADINO122) (NAME (1) .1=1.3) ONNONSONFONWOLFVOLATOLON 1F (NAME (1) . EQ. EOF) GO TO 112 SERCH029 WEAD (HEADER-123) (SWIT(J) + J=1+15) SERCH030 LAT=1CUN(LAT) SERCH031 LON=1CUN (LON) SEPCH032 IF (LAI-EQ.ILAI-AND-LON-EQ.TLUN) GO TO 113 SERCH033 1F (TS.NE.1) 60 TO 103 SERCH034 WRITE (#PITEU+124) (NAME(I)+I=1+3)+LAT+LON+LEV SERCH035 SERCH036 C С . DETERMINE IF THE STATION LIES WITHIN THE RADIUS OF SERCH037 INTEREST. ON THE AVERAGE 59 MILES = 1 PEORFE LONGITUDE AND 1 LAIITHDE DEGREFF = 1.32 AVERAGE LONGITHDE DEGREFS. C SERCH038 C SERCH039 C SERCH040 103 LLAT=(LAT-1LAT) 91. 32 SERCH041 KSU=L[A1 ** 2+ (L()N-1L(IN) **2 SERCH042 RSU=SOR1 (RSU) SERCH043 С SERCH044 Ĉ . . DETERMINE IF STATION IS WITHIN 2 DEGREES (118 MI.) SEPCH045 PADIUS. LAT. LON READ IN 15 FORMAT. THUS THEY ARE TOO BIG HY A FACTOR OF 100. С SENCH046 c SERCH047 SERCH048 1F (K50-200.) 104.104.102 SENCH049 SERCH050 CCCC . . DETERMINE IF UNKNOWN IN AREA OF APPLICABILITY OF KNOWN SERCH051 STATION. SERCH052 С SERCH053 104 1F (SWDT(1) .NE.0..AND.SWDT(6) .NE.0..AND.SWDT(11) .NE.0.) GO TO 105 SERCH054 WRITE (#RITFU+120) (NAME(1)+1=1+3) GO TO 102 SEPCH055 SERCH056 105 CALL APPLI (LAT+LON+LEV+NAME+SWDT) SERCH057 SEHCH058 DETERMINE WHICH QUADRANT IT IS IN AND IF IT IS THE SERCH059 MINIMUM DISTANCE STATION IN THAT QUADRANT SO FAR. SERCHORD ē SERCH061 IF (LAT-1LAT) 106+106+108 SERCH062 106 IF (LON-ILON) 107+107+111 SERCH063 107 CALL MINUM (KK (3)+LAT+LON+LEV+ILEV+OLAT (3)+OLUN (3)+OLFV (3)+OSWOT (3SERCH064 11) + SHDT + RSQ + QKSQ (3) + NAME + (NAM (3+1) + QNAM (3+2) + UNAM (3+3)) SERCH065 GU TU 102 SERCH066 108 IF (LON-ILON) 109+109+110 SERCH067 109 CALL MINUM (KK(2)+LAT+LOH+LEV+1LEV+0LA1(2)+0LUN(2)+0LFV(2)+0SWDT(1SERCH068 16) • SWDI • HSQ • QHSQ (2) • NAME • QNAM (2 • 1) • QNAM (2 • 2) • UNAM (2 • 3)) SERCH069 60 TU 102 SERCH070 110 CALL MINUM (KK(1)+LAT+LUN+LEV+ILEV+QLA1(1)+QLUN(1)+QLEV(1)+QSWDT(1SERCH071 1) . SWUT . M SQ . ORSQ (1) . NAME . UNAM (1.1) . ONAM (1.2) . UNAM (1.3)) SFRCH072 SERCH073 GO TO 102 111 CALL MINUM (KK(4)+LAT+LON+LEV+TLEV+OLAT(4)+OLON(4)+OLEV(4)+OSWDT(4SERCH074 10) . 5HDI . RSD . UNSD (4) . NAME . UNAM (4.1) . ONAM (4.7) . UNAM (4.3)) SERCH075 GO TU 102 SERCH076 112 RETURN SERCH077 113 WHITE (WHITEU.125) (NAME(1).1=1.3) SERCH078 IF (SWUI(1).NF.0..AND.SWUT(6).NE.0..AND.SWDT(11).NF.0.1 GO TO 114 SERCH079 WRITE (WRITEU+119) SERCHORD GO TU TU2 114 WRITE (WRITEU+126) SERCHOR1 SERCHOR2 DO 115 1=1.15 SEPCHOR3 115 SWUT(1)=SWD1(1)*.001 SERCHOR4 DO 116 l=1+11+5 SENCHORS Il=l+1 SEPCHORA 12=1+2 SERCHON7 13=1+3 SERCHOAA 14=1+4 SERCHOR9 WHITE (WHITEU+127) SWDT(1)+SWDT(11)+SWDT(12)+SWDT(13)+SWDT(14) SEPCH090 116 CUNTINUE SEPCH091 IF (1P+NE+1) HETUEN SEPCH092 K=0 SERCH093 DO 118 1=1+7+3 SEHCH094 M=1+K SENCH095 AA(1)=SWDT(M) SERCH096

SERCH097 HH(1)=0. SEHCH09A Sel=L 711 00 L+L=L1 SEHCH099 SERCH100 MJ=M+J SENCH101 AA(1.J)=SWDT(4J) SERCH102 MJ=MJ+1 HB(1J)=SWI)T(HJ) SERCH103 M=M+1 SERCH104 SERCH105 117 CUNTINUE SERCH106 K=K+5 SERCH107 118 CONTINUE SERCH108 CALL HEADAY RETURN SEPCH109 SERCH110 CCC SERCHIII FORMATS USED IN THIS SUBROUTINE. SERCH112 119 FORMAT (1H0,111HTH1S STATION POFS NOT HAVE A COMPLETE DATA SET - TSERCH12 1HE PROGRAM WILL ATTEMPT TO USE ANOTHER STATION OF INTERPOLATE) SERCH14 120 FORMAT (10x,9HSTATION +244,42,86H1S WITHIN 1WO PEGPEE RADIUS, BUTSERCH15 1 DOES NUT HAVE A COMPLETE DATA SET - THEREFORE IGNORED) SERCH16 121 FORMAT (1H0,20x,17HSTATIONS SEAPCHED) SERCH16 SERCH17 122 FORMAT (224+A2+9X+4(13+1X)+1X+14+2(2X+15)) 123 FORMAT (15F5+0) SERCH118 SFRCH119 124 FORMAT (5x+244+42+5x+6HLAT = +110+5x+6HLON = +110+5X+7HELEV = +1105EHCH120 SEPCH121 1) 125 FORMAT (1H0+244+42+47H IS A WEATHER STATION AT THIS LAT. LON AND FSERCH122 ILEV) SERCH123 126 FORMAT 1//75H FOURIER COEFFICIENTS FOR CLIMATIC PARAMETERS, P(D/D)SERCH124 1. D(D) . AND STUHM SIZE/17X.2HAU.1AX.2HAI.1AX.2HAI.1AX.2HBI.18X.2HA2.1AX.2HHSERCH125 22) SERCH126 SERCH127 127 FORMAT (1H0.5(10X.F10.4)) SERCH128 С SENCH129 END

~		SUBROUTINE SER (NT+A1+A+B+QUA())	SER	001
C			SER	002
C		THIS SUBROUTINE (SER) IS CALLED BY THE SUBROUTINES	SER	003
C		(WAVG2 AND HEADAY) TO GENERATE & FOURTER SERIES REPRESENTATION	SFR	004
C		OF THE DAILY VALUES FOR A GIVEN PARAMETER UVER A YEARS TIME.	SER	005
С			5ER	006
		INTEGER I.N.NT	SER	007
		REAL A(3)+A1+B(3)+QUA()(365)+SUM+X1+XN	SER	008
C			SER	009
		00 102 I=1,365	SER	010
		XI=FLOAT(I)	SER	011
		SUM=0.	SER	012
		DO 101 N=1.NT	SER	013
		AN=FLOAT(N)	SER	014
	101	5UM=5UM+A(N)*CO5(+01725*X1*XN)+B(N)*SIN(+01725*X1*XN)	SER	015
	105	QUAD(I)=A)+SUM	SFR	016
		RETURN	SER	017
С			SER	018
		END	SER	019

CUUDIOUZING 000

SURRUHTINE WAVEL WAVG1001 C C HAV61002 . . THIS SUPROLITINE (WAVEL) IS CALLED BY THE MAIN PROCRAM. WAVG1003 č (MUELED) TO COMPUTE THE WEIGHTING FACTORS FOR INTERPOLATION. WAVG1004 č WAV61005 INTEGLE INTLATAILEVALUNATUATEAISAITAITAITAKEANEANN WAVG1006 INTEGER NSONWORMAN OF ALLOWHITEL REAL AGAAGATONVOROHIGETOETOCIATOLEVOLONOMIND WAVG1007 WAVG1008 REAL UMINI . UHSU . USWIT . HSU . 5. TS . V (4) WAVG1009 С WAVG1010 CUMMON 2F11ES2 READUSWEITEU WAVGIOII CUMMUN 2V1PS2 A(3) + A4(9) + A4(5) + H(3) + HF(9) + ILAI + 1 + EV+ ILON+IO+1P+IS+WAVGIO12 111.11.11.12.KK (5) .NF .NH. N5. NK. 01 AT (5) .0LFV (5) .0LUN (5) .0NAM (5.3) .0250 (WAVG1013 25) +05w01 (75) +650+5 (365+4) +TS (365) WAVG1014 WAVG1015 С DT = 0WAVG1016 AI=0. WAVG1017 t T=U. WAVG1018 WAVG1019 С С SUM DISTANCES. WAVG1020 C WAVG1021 QMIND=0. WAVG1022 00 101 1=1.4 WAVG1023 IF (WKSD(I)*FQ*n*) GO TO 101 IF (WMINI*Eu*n*) UMIND=URSU(I)WAVG1024 WAVG1025 UNTIOD=AMIN1 (UN1UD+0050(1)) WAVG1026 101 CONTINUE WAVG1027 NO 103 1=1.4 WAVG1028 IF (WRSQ(1).FQ.0.) GO TO 102 WAVG1029 C WAVG1030 C . COMPUTE & SIMILARITY INDEX (DT) USING THE CLOSEST WAVG1031 STATION FROM ALL FOUR GUADRANTS AS THE RASTS. SIMILARITY INDEA = 2W/(A+P) where w is the minimum of A and B. IN this C WAVG1032 C WAVG1033 CASE . THE MINIMUM DISTANCE STATION FROM ALL FOUR QUADRANTS. C WAVG1034 С WAVG1035 QKSU(1) =2.00M1ND/(OM1ND+0850(1)) WAVG1036 DT=DI+ORSO(1) WAVG1037 102 V(I)=0. WAVG1038 103 CONTINUE WAVG1039 IF (DT) 104+104+105 WAVG1040 104 WRITE (WRITEU+115) WAVG1041 RETURN WAVG1042 WAVG1043 C C FIND RFLATIVE DISTANCES. WAVG1044 С WAVG1045 105 00 106 1=1.4 WAVG1046 106 ORSU(1)=0RSU(1)/01 WAVG1047 С WAVG1048 С SUM ELEVATIONS. NOTE - A ZERO VALUE FOR ELEVATION WAVG1049 . . INDICATES NO STATION PRESENT IN THAT QUADMANT. NOT AN ELEVATION OF ZERO. C WAVG1050 WAVG1051 č WAVG1052 QMINL=0. WAVG1053 DO 109 1=1.4 WAVG1054 IF (ULFV(I).E0.0.) GO TO 109 WAVG1055 V(I) = ABS(OLEV(I) - FLOAT(ILEV))WAVG1056 (V(I)) 107+107+108 IF WAVG1057 107 V(1)=1. WAVG1058 QMINL=1 WAVG1059 GO 10 109 1F (GMINL.FO.O.) GMINL=V(]) WAVG1060 108 WAVG1061 QMINL=AMINI (QMINL +V(1)) WAVG1062 109 CONTINUE WAVG1063 00 110 I=1.4 WAVG1064 C WAVG1065 С COMPUTE SIMILARITY INDEX FOR ELEVATIONS (FTL. WAVG1066 C WAVG1067 1F (V(I).EQ.0.) GO TO 110 WAVG1068 A(I)=5*#0WINF\(OwINF+A(I)) WAVG1069 ET=ET+V(11 WAVG1070 110 CONTINUE WAVG1071 С WAVG1072 C FIND RELATIVE SIMILARITY IN ELEVATIONS. WAVG1073 Ċ WAVG1074 00 111 1=1.4 WAVG1075 111 V(I)=V(1)/FT WAVG1076 С WAVG1077 C FIND AVERAGE WEIGHTING FACTOR (AV) - AVEPAGE OF TWO TIMES WAVGIOTA Ċ ELEVATION FACTOR AND THE DISTANCE FACTOR (USRO). WAVG1079 С WAVG1080 DO 112 1=1.4 WAVG1081 112 AV(1)=(c.*V(1)+0PSO(1))/3. WAVG1082 С WAVG1083 С NORMALIZE AVERAGE WEIGHTING FACTOR (AV). WAVG1084 C WAVG1085 DU 113 1=1.4 WAVGIORA 113 AT=AT+AV(1) WAVG1087 DO 114 1=1.4 WAVG1088 114 AV(I)=AV(1)/AT WAVG1089 RETURN WAVG1090 C WAVG1091 C FORMATS USED IN THIS SUMMOUTINE. WAVG1092 C WAV61093 115 FORMAT (1H0+47HNO STATIONS WITHIN TWO DEGREE HADIUS OF UNKNOWN) WAVG1094 C WAVG1095 FAD

SUHRUUIINE WAVG2 WAV(-2001 WAV62002 C. . • THIS SUMMOUTINE (WAVG2) IS CALLED BY THE MAIN PROGRAM WAVG2003 (PREDEN) TO GENERATE DALLY VALUES FOR P(D/D) + P(D) + AND SS WAVG2004 AS WELL AS FINDING THE WEIGHTED AVEHAGE AND FINALLY RECOMPUTES WAVG2005 CCCC . FOURLEP COFFFICIENTS. WAV62006 WAVG2007 Ċ WAVG200R INTEGER TA. IH. ILH. IJ. ILAT. ILFV. TLON. TO. IP. TS. IT. TY INTEGER 17. J.J. J. KOKK OLOLJOMONT ONNONSONI ONWORNAM WAVG2009 INTEGER HEADU. WETTEU WAVG2010 HEAL A.A.A.AV.AI.R.HR.OLAT.OLFV.QLON.QRSU.QSWUI.RSO WAVG2011 WAV62012 HEAL SOIS WAVG2013 С COMMON /FILES/ READU.WRITEU WAVC 2014 COMMOD: /VIPS/ A(3) +AA(9) +AV(5) +H(3) +H(3) +ILAI+II FV+II ON+IO+TP+I5+WAVG2015 111+IY+IZ+KK(5) +NF+NH+NS+NW+ULAI(5) +OLFV(5) +OLUN(5) +ONAM(5+3) +ORSO(WAVG2015 25) + 45401 (75) + 854+5 (365+4) + 15 (365) WAVG2017 WAVG2018 С WAVG2019 L=1 WAV6-2020 NT=2 С WAVG2021 č . LOOP FOR GENERATING THE THREE PARAMETERS. P(D/D). P(D). WAVG2022 С AND 55. WAV62023 WAVG2024 C DU 105 1J=1+11+5 WAVG2025 00 101 K=1+365 WAVG2026 101 15(K)=0. WAVG2027 С WAVG202A LOOP FOR THE FOUR QUADHANTS. č WAVG2029 C WAVG2030 00 103 J=1+4 1JJ=(J-1)*15+IJ WAVG2031 WAVG2032 A1=USWDT(IJJ) WAVG2033 1A=150(J=1)+1J+1 WAV62034 19=14+5 WAV62635 C ¥±VG2036 č GENERATE PAPAMETER SERIES FOR EACH QUADRANT. WAVG2037 . C WAVG203A CALL SER (NT.A1.05.01(IA).USENT(IR).S(1.J)) WAVG2039 C WAVG2040 č . SUM WEIGHTED DAILY PARAMETER VALUES UVER THE FOUR WAVG2041 . C QUADHANTS. WAVG2042 C WAVG2043 DU 102 K=1.365 WAVG2044 TS(K)=TS(K)+S(K,J)+AV(J) WAVG2045 102 CUNTINUE WAVG2046 103 CONTINUE WAV62047 C WAVG2048 č WAVG2049 . GENERATE NEW FOURIER COEFFICIENT FOR ITH PARAMETER IN SUANUUTINE (FORIT). C WAVG2050 WAVG2051 N=182 WAVG2052 M=2 WAV62053 CALL FORIT (TS.N.M.A.B.IER) WAVG2054 C WAVG2055 . . . WRITE TO OUTPUT. С . WAVG2056 č WAVG2057 1F (1J.FQ.1) WRITE (WRITEU.106) WAVG2058 WRILE (WRITEU.107) A(1). (A(J).B(J).J=2.3) WAVG2059 AA(L)=A(1) WAVG2060 BB(L)=0. WAVG2061 DO 104 J=1.2 WAVG2062 LJ=L+J WAVG2063 JI=J+1 WAVG2064 AA(LJ)=A(J]) WAVG2065 88(LJ)=8(J1) 104 WAVG2066 1 =1 + 4 WAVG2067 105 CONTINUE WAVG206A RETURN WAVG2069 С WAV62070 C C FORMATS USED IN THIS SUBROUTINE. WAVG2071 WAVG2072 106 FORMAT (775H FOURTER COEFFICIENTS FOR CLIMATIC PARAMETERS, PID/D)WAVG2073 1. P(D). AND STORM SIJE/17X+2HA0+16X+2HA1+18X+2HB1+18X+2HA2+18X+2HPWAVG2074 22) ¥AVG2075 107 FORMAT (1H0+5(10X+F10.4)) WAVG2076 С WAVG2077 END WAVG207A

APPENDIX #6

Coefficient Determination for Sine Function Weather Generation

An algorithm follows which enables a user to determine the input parameters for the RANGES model which will determine the coefficients for either the temperature or precipitation sine functions.

- (i) Compute average monthly values for either temperature or precipitation (Fig. 1).
- (ii) Plot the monthly values for a year. The result will probably give data that has the form of a sine function. If not, then this method may not be applicable.
- (iii) Find the difference between the high and low average monthly values(e.g. 2.8 .2 = 2.6). This is twice the amplitude of the sine wave.Divide the difference by two to find the amplitude H1(H1 = 2.6/2 = 1.3).
- (iv) Add the low average monthly value to the amplitude and construct a horizontal line (L) at this height (2.6/2 + .2 1.5).
- (v) Find the point where the sine wave crosses line L going up from left to right, and determine the number (H2) of the day of the year (Julian day) where this occurs (H2 = 110 days).
- (vi) Set H3 equal to the low average monthly value (H3 = .2).
- (vii) These three parameters will be read into the model and take their corresponding places in the following equation:

Y = H1 * (SIN ((DDAY - H2) * .0172) + 1.) + H3

where in the example the expanded equation is:

PPT = 1.3 * (SIN ((DDAY - 110) * .0172) + 1.) + .2

The monthly values predicted by this equation are represented by o in Fig. 1.

If the predicted monthly values are summed, they may over or underestimate the average yearly precipitation. Therefore, it may be necessary to increase or decrease the amplitude of the sine wave to obtain a more realistic yearly average. In our example, precipitation is overestimated, so the following procedure is used.

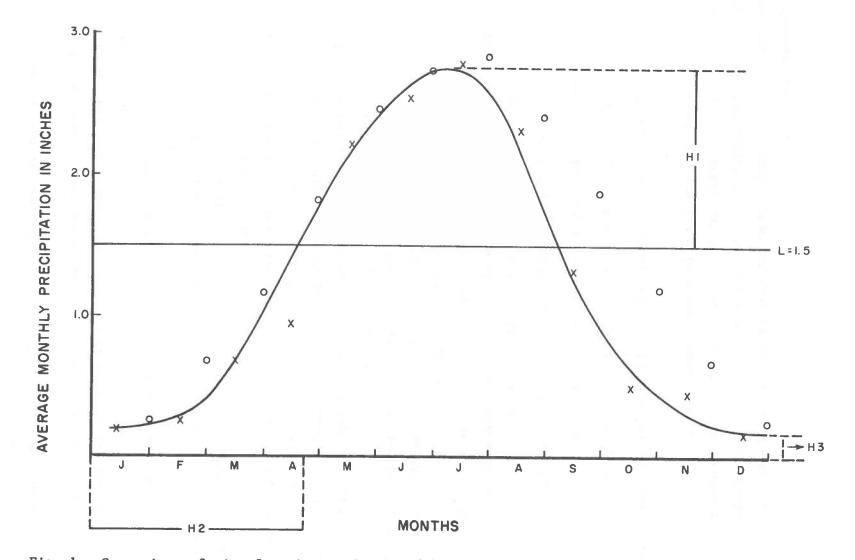


Fig. 1. Comparison of sine function prediction (o) and actual (x) monthly precipitation amounts for the Pawnee Site.

The steps to improve sine function estimate of average yearly precipitation are:

- (i) Determine a new axis based on a lower maximum value (L = (2.6 .2)/2 + .2 = 1.4).
- (ii) Find the new amplitude (H1 = 2.4/2 = 1.2).
- (iii) Substitute the new H1 into the sine function and leave H2, H3 as before.
- (iv) Check the new estimate (Fig. 2).

Again, the yearly estimate is too high. Therefore, another iteration is required to converge on an adequate function. Therefore, determine a new axis (L = (2.2 - .2)/2 + .2 = 1.2). Find a new H1, H1 = 2.0/2 = 1.0), and recompute the estimates. Now, the yearly average of the sine function is 14.6" compared with 14.9" computed from the data (Fig. 3). It is assumed close enough, although there is some underestimation in the summer and overestimation in the fall.

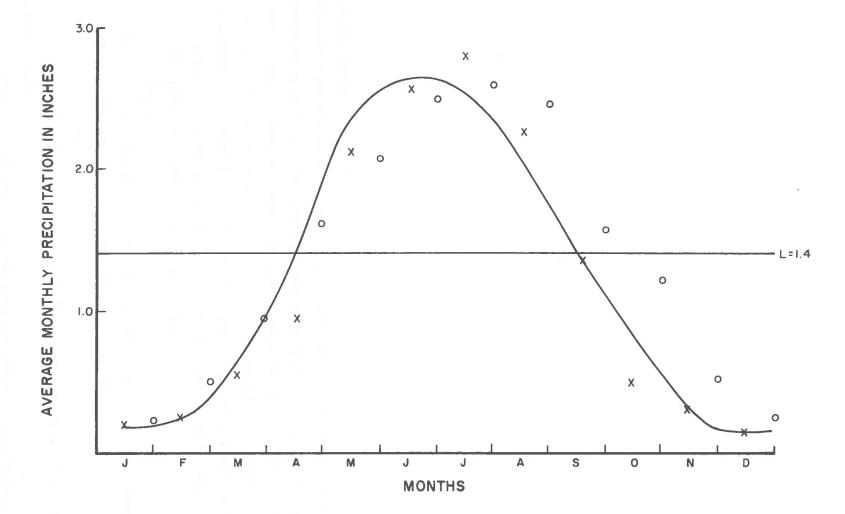


Fig. 2. Comparison of sine function prediction (o) and actual (x) monthly precipitation amounts for the Pawnee Site.

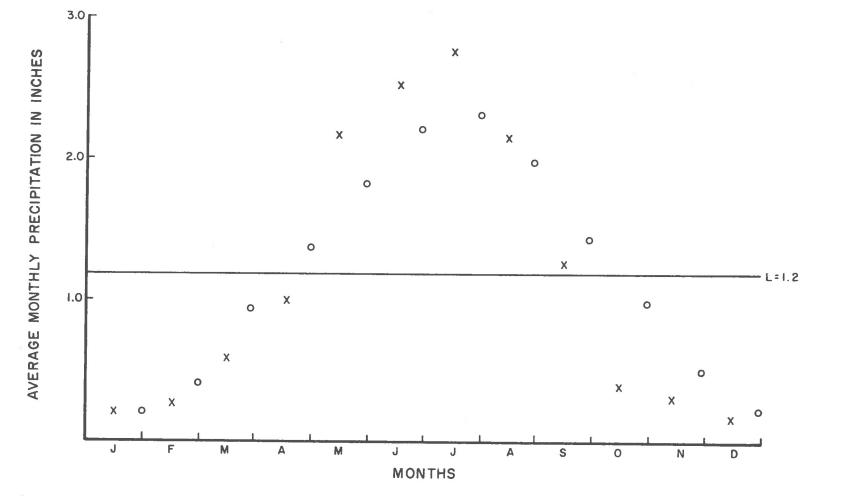


Fig. 3. Comparison of sine function prediction (o) and actual (x) monthly precipitation amounts for the Pawnee Site.