THE USE OF A DYNAMIC DIGRAPH STRUCTURE IN A POPULATION SIMULATION MODEL FOR GRAIN SORGHUM

By

JESS WALTER CURRY, JR.

B.S. Oklahoma State University

Stillwater, 1970

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Computer Science

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1977

Approved by:

____

Major Professor

ILLEGIBLE DOCUMENT

THE FOLLOWING DOCUMENT(S) IS OF POOR LEGIBILITY IN THE ORIGINAL

THIS IS THE BEST COPY AVAILABLE

ACKNOWLEDGEMENTS

The project detailed in this report was funded by the Kansas State University Agricultural Experimental Station, Floyd W. Smith, Director.

I would like to thank Dr. Richard L. Vanderlip who spent many hours directing this project and Dr. Linda G. Shapiro who advised and directed me throughout.

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

The Use of a Dynamic Digraph Structure in a Population Simulation Model for Grain Sorghum

I. INTRODUCTION

Sorghum (often referred to as milo) has, through successful research efforts, grown to be one of the leading world crops both in production area and in food value. As of 1969, sorghum ranked fifth in acreage of crops in the world, being exceeded only by wheat, rice, corn, and barley. Perhaps of greater importance is the fact that it is a major food crop in the most heavily populated regions of the world. The three top producers of sorghum in 1970 were the United States, Mainland China, and India.

Because of its resistance to drought and heat, sorghum has been a prominent crop in the southern half of the Great Plains area since the early 19th Century. It was a highly successful livestock food for the early pioneer farmers especially in the semi-arid regions. Also, due primarily to research, sorghum production grew seven-fold between 1940 and 1968 alone, and as a result, it has become an important cash crop for the American farmer. The development of early maturing varieties allowed expansion to the higher altitudes to the west and cooler areas to the north. Shorter varieties allowed automation of harvesting while development of insect and disease resistant hybrids have increased per acre production considerably.

Kansas ranks second only to Texas in total sorghum production and has played a key role in the continued improvement of sorghum. Significant contributions have been made and are presently being made by the agricultural experiment stations of Texas and Kansas. One such contribution was the development in 1975 of the first computer simulation model for grain sorghum by Drs. G. F. Arkin and J. T. Ritchie of the Blacklands Conservation Research Center, Temple, Texas and Dr. R. L. Vanderlip of the Agronomy Department at Kansas State University <1>. This model was developed primarily as a tool for improving management decisions for increased sorghum crop production.

The Arkin, Ritchie, Vanderlip model was based upon simulating a single sorghum plant from emergence to maturity under the weather and soil conditions given for the sorghum field under consideration. A per acre production figure was derived by multiplying the production of the single plant by the number of sorghum plants per acre. This model, which will be referred to as the Single Plant Model, was a fairly sophisticated model with a good degree of accuracy. However, a detailed study of the Single Plant Model showed a complete lack of program structuring and internal documentation. More importantly, the study raised the question as to the accuracy of modeling an entire field based upon the output of one single plant. For example, in the original simulation model, no consideration was given to field variables such as soil variability, seed distribution, emergence date variance, and plant leaf area and tillering.

The current project was instigated to design a field level simulation model which: 1) would have a modular structure, 2) would have sufficient internal documentation, and 3) would expand the model developed by Drs. Arkin, Ritchie, and Vanderlip to include simulation of the variability within a field of sorghum. The new simulation model developed during this project will be referred to as the Field Level Model. The basis of this model is a data structure representing a directed graph (digraph). Each node in the data structure will represent a portion of the field which has some significant difference from the rest of the model. The digraph is the result of plotting the nodes from day to day as they diverge and converge (see figure 3). For example, if half of the field holds water very well while the remainder can not, then the model would be split (the graph will diverge) into two equal parts (or nodes) each of which will be simulated based upon its separate characteristics. The nodes will also be allowed to merge (or converge) if the difference between the two parts is eliminated. If in the above example it rains enough to allow sufficient water for the entire field then the nodes would converge. The converging and diverging of nodes will be based upon internal policies programmed into the model at a later date. It is the objective of the current project to develop the data structure and prove that it is applicable and useful.

II. The Single Plant Model

It is difficult to present the workings of the Single Plant Model because of the initial lack of modularity and program structure. To overcome this problem the model will be presented in a more logical format with proper modularization and program flow. This represents the state of the model after it was reorganized for use in the Field Level Model but before any new modules were added.

Figure 1. gives a high level diagram for the Single Plant Model showing the various subroutines in the order that they are called. Figure 2. shows a hierarchy diagram of the model before modularization was undertaken. The main routine initializes the critical variables to zero and reads the remaining constant variables from cards. These variables represent plant data, soil data, planting data and location data, all of which remain unchanged throughout the simulation. Table 1 gives a breakdown of the information used in each data type.

After the constant data has been read, up to one year's climatic data is read. This data gives the maximum and minumum temperatures, solar radiation, and rainfall. As each day's climatic data is read, the day's evaporation is calculated by the subroutine EVAP. This value is subtracted from the available water content of the soil by a second subroutine called SOLWAT. SOLWAT also adds rainfall amounts to the soil contents when applicable and then calculates the moisture limiting factor for photosynthesis. Both of these

TABLE

Plant Data

Leaf number -- total number of leaves produced

Leaf area -- maximum area of each individual leaf, cm2

Planting Data

Planting date

Plant population

Row width

Row direction

Climatic Data (daily from planting to maturity)

Maximum temperature, C

Minimum temperature, C

Solar radiation, langleys per day

Rainfall, cm

Location Data

Extractable soil water capacity, cm

Initial extractable soil water content, cm

Latitude

Table 1. Input data required.

The Single Plant Model
(After Modularization)

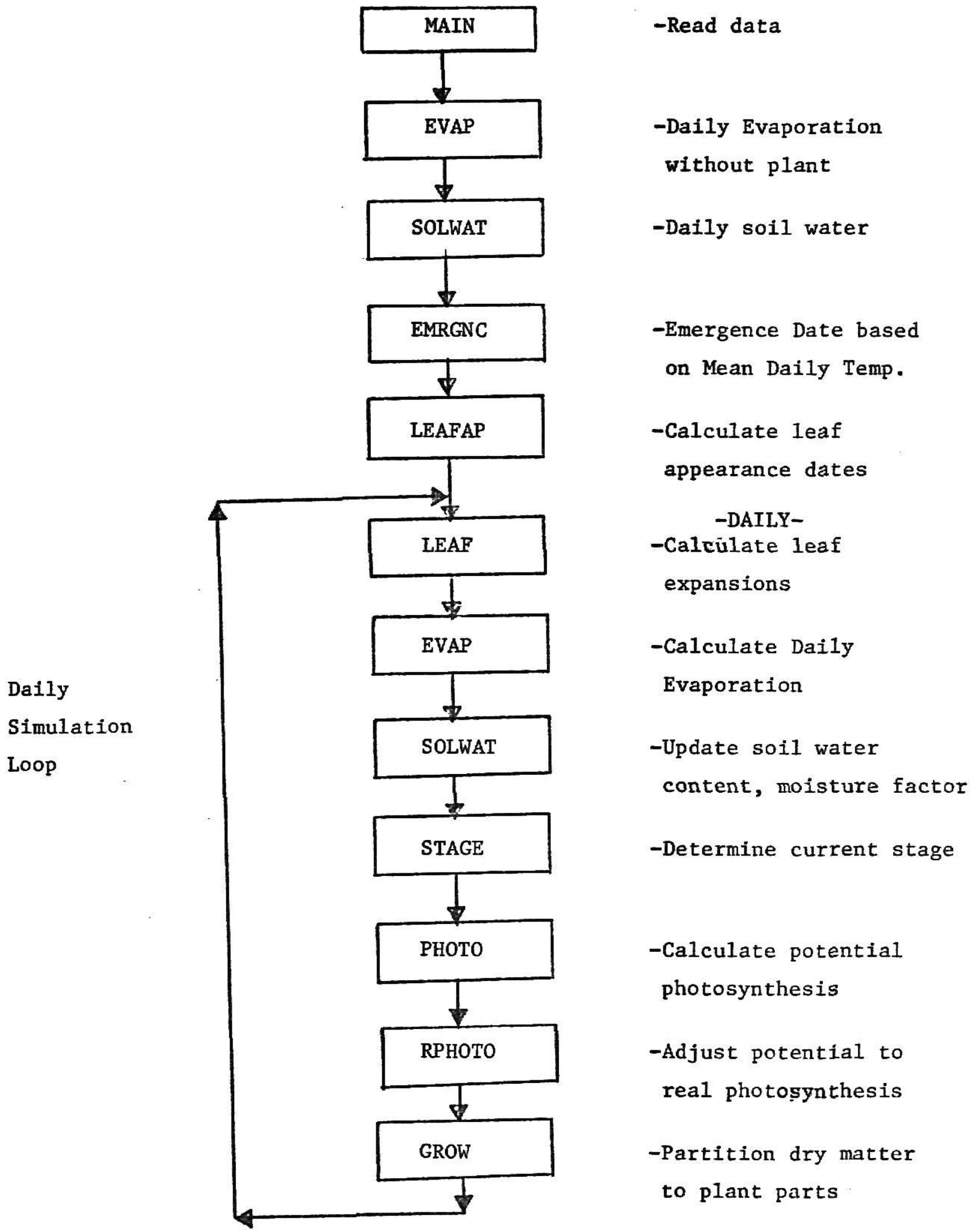

FIGURE 1.

Single Plant Model's Hierarchy Diagram*

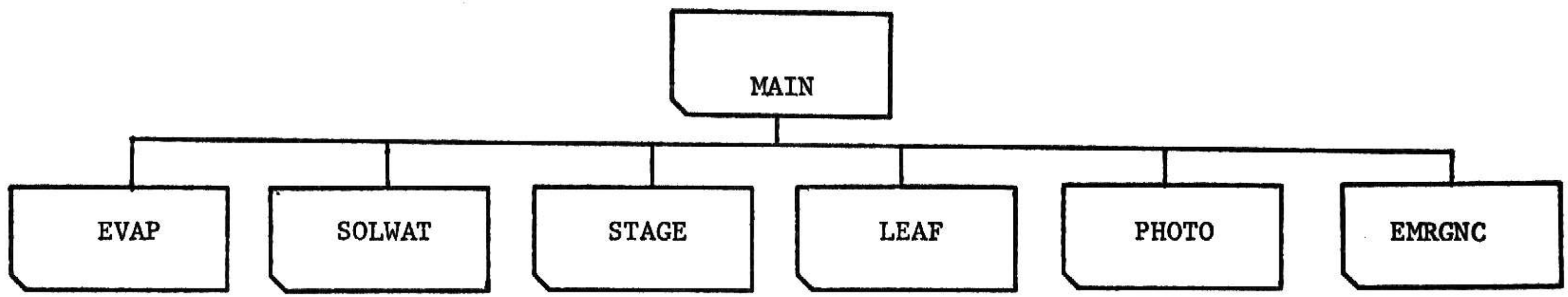

FIGURE 2.

subroutines will be used again when growth simulation begins. After all the climatic data has been read, the emergence module (EMRGNC) is called to calculate emergence date of the single plant based upon mean daily temperature.

Each sorghum hybrid has a maximum number of leaves which will develop and a maximum leaf area for each leaf. The model is heavily based upon the amount of light intercepted by the plant. Therefore, the leaf area per plant must be known. Field and phytotron studies have shown that the rate leaves appear on grain sorghum is directly related to mean daily temperature. The next step is to call the leaf appearance module to calculate the date each leaf appears (subroutine LEAFAP) based on mean daily temperature.

These first three stages are completed in preparation for the actual daily growth simulation. A large loop is established beginning on the date of emergence and ending either at plant maturity or the last date climatic data was supplied, whichever comes first. Within this loop there are seven stages or modules (1).

The first stage calls the module LEAF which calculates the expansion of each leaf and adds this amount to the area of that particular leaf. This figure will then be used to

----

(1) This loop with some modifications will later be referred to as the Daily Simulation Control Routine or DSCR.

calculate the amount of light intercepted by the plant. Leaf senescence (death due to aging) is accounted for by subtracting the maximum leaf area of the first leaf from the total leaf area at the time the twelfth leaf has reached its maximum area. This process then continues for each successive leaf which appears after the twelfth leaf. Once all the leaves have appeared the modeling of senescence continues by subtracting 0.4 percent of the total leaf area from the total area each day until maturity.

In the second and third stages, the EVAP and SOLWAT routines are called again to give the daily water evaporation and to update the soil water content, respectively. During initialization these routines were called to establish the soil condition for each day under the assumption that no plant was present. This data is now used as input on a daily basis and new figures are calculated based upon the presence of the sorghum plant. Although this seems to be a fine point on the differences between the initialization calls to EVAP and SOLWAT and the simulation calls, it is an important point. This is especially true for SOLWAT which takes the evaporation figures from EVAP and separates them into soil evaporation and plant evaporation both of which are taken from the soil content figure. The division is important from two standpoints; first in the initial calls no plant evaporation occurs, thus affecting the soil content in a different manner, and secondly in the simulation calls the moisture limiting factor for photosynthesis which is derived in SOLWAT is a major factor on plant growth.

The fourth stage is for the simulation model to determine the stage of plant development. There are three important stages in the development of grain sorghum and the model must identify the current stage in order to apply certain important growth factors. The subroutine STAGE does this based upon the date that certain leaves reach their maximum leaf area which is an output from the LEAF routine. STAGE sets a common variable representing the plants development stage which is then referenced by the final three stages.

Stage five of the plant simulation is the most sophisticated and technically based portion of the model. In this stage the PHOTO subroutine is called to calculate the potential photosynthesis of the plant under each day's conditions. Since light interception is the most critical element in the model, the shading of leaves both from the plant itself and from neighboring plants becomes a critical factor. Complicating the problem is the fact that shading within the plant's canopy is dynamic, that is, it changes with the sun's altitude and azimuth and with plant's size. To account for these dynamic interactions the PHOTO routine applies a mathematical model developed by Arkin and Ritchie <2> to simulate the light interception in the grain sorghum plant canopy.

Using this model, PHOTO estimates the net potential photosynthesis in terms of the amount of carbon dioxide fixed during daylight hours under certain common conditions. The calculations are made and summed for each hour of the daylight portion of the day. The result is a daily figure representing potential photosysthesis.

In stage six the potential photosynthesis is passed to the RPHOTO routine where a series of efficiency functions are applied to the data. These functions modify the potential photosynthesis to reflect a more realistic value based upon non-optimum temperatures and soil water conditions. There is an efficiency parameter for each environmental constraint on the photosynthetic rate. Examples of these would be night-time respiration loss, temperature conditions and soil water conditions. As an example of their effect, all photosynthesis stops below the temperature of minus five degrees C; thus all potential photosynthesis calculated below that temperature would be subtracted to give a zero rate for each hour the temperature was below minus five degrees C. Similarly, reduction in photosynthesis due to limiting soil moisture has been shown to be proportionate to the reduction in plant evaporation resulting from the limited water availability <7>. There is a threshold of extractable soil moisture below which plant evaporation is affected. This threshold is dependent upon the particular soil and enters the model as a single soil water holding capacity variable. The extractable soil water is determined daily by the model using a modified soil water balance routine developed by Ritchie <6>. Once the water available has been reduced to 25% of the soil holding capability an effect upon net photosynthesis is seen.

The seventh and final stage of the Single Plant Model is the GROW routine. It is within this routine that all dry matter is partitioned between the various plant parts (root, leaf,culm, head, and grain). The basis of the distribution is the stage of development which is an output from the STAGE routine. The plant's growth is broken into three major phases and each phase gives a different dry matter distribution. Each part's weight is kept on an acculmulative basis from day to day until maturity.

After the GROW routine has calculated the day's growth factors and they have been added to the plant's development, the model loops back to stage one and begins simulating the next day's growth. Final maturity is based upon 1.6 times the number of days to half-bloom (anthesis) which, in turn, is based upon leaf development.

In summary, the model goes through seven stages each day with each stage applying critical factors progressively until the final stage where the day's growth estimates are partitioned into the various plant parts. The plant's maturity date is calculated as a function of leaf development. These steps are summarized in Figures 1 and 2.

III. OBJECTIVES AND APPROACHES

The following are the four main objectives of this project.

1. To add internal documentation to the main routines of the Single Plant Model.

2. To modularize the model into logical units in preparation for later updates and replacements.

3. To develop an acceptable approach to extending the Single Plant Model to a field population model.

4. To prove the validity of the approach by implementing divergence due to variations in emergence dates.

The first two objectives were to represent only about five percent of the project effort. The development of the field model was expected to account for approximately 40% of the project effort, while the remaining 55% was dedicated to converting the Single Plant Model into a field model in accordance with objective four.

A limited literature search was undertaken to study some of the more widely used plant simulation models. Included in the study was a cotton model developed at

Missippippi State <5>, a vegetative plant growth model <8>, a sugar beet model <4>, and a distribution model of organism Development times <3>. Although several of the models studied presented alternate approaches, the result of the study was that no new approach would be undertaken and the project would be oriented toward the Single Plant Model because of the large amount of local time and effort already committed to the Single Plant Model and the successes seen in its development.

Because of the already large size of the program that simulates the Single Plant Model (86K-bytes), the problem was to develop an approach which would not greatly increase the model's core requirements or run time.

The first serious approach considered was to partition the sorghum field into a variable number of segments, each of which would represent a specified combination of field variables. For example, a field consisting of two basic soil types, 60% of one and 40% of another would be split into two partitions representing 60% and 40% of the field, respectively. These partitions would then be sub-partitioned based upon a second variable, for example, there might be three separate emergence dates representing 30% emergence on day one, 40% on day two, and 30% on day three (2). This results in six partitions representing 18%,

----

(2) That is 30% of all the plants which will emerge do so on the same day the first plant emerges while 40% emerge on the next day and all remaining plants emerge on day three.

12%, 24%, 16%, 18%, and 12% of the field. The process would continue for each field variable appliciable to the model. The Single Plant Model would then be called to simulate each partition and a total field production would be estimated by weighting each partition based upon the percent of the entire field it represented.

There were several major difficulties with this approach. The number of simulations, and thus the run time required, appeared to grow exponentially with the number of field variables added to the model. Secondly, several of the crucial field variables do not become critical until later into the simulation itself. Since these variables could not be ignored, a modification was considered which would allow the model to dynamically add additional partitions during the simulation of any partition. These partitions would then be added to the list of partitions to be simulated. Although, the new approach cut down the initial number of partitions required, it allowed each partition to create new partitions which in turn could create still more partitions, etc. The approach was theorically desirable but since there appeared to be no limit to the number of partitions which could be created, it was not considered realistically applicable.

An alternative solution was to apply the same theoretical approach, but in a more manageable structure. The idea was to build a dynamic tree structure which would allow the concurrent simulation of the various partitions on a daily basis. The tree would grow dynamically each day with each leaf (3) representing a field partition. When a new field variable becomes critical at one of the leaves then that leaf would diverge into one or more additional leaves in preparation for the next day's simulation.

At first, a tree search routine was considered for limiting the number of nodes. With this approach partitioning or branching would be unlimited. At simulation time each branch would be searched to see if its simulation would be critical based upon policies which would have to be written into the program. Figure 4. demonstrates how this approach would have worked. In the beginning each branch is considered and only those nodes considered critical would be simulated (nodes 1,2,4, and 6) . On the next day only those branches and their potential diverging (sons) branches would be considered. At day two nodes 1, 2, 7, 8, and 6 are simulated. This procedure would continue until all paths reach maturity.

Theoretically this approach also seemed acceptable because there are several known algorithms for tree searching and dynamic divergence would be allowed. However, some major difficulties arose. First allowing the creation of branches which may never be used increases storage requirements unnecessarily. Even if such unused branches were ignored and never stored in memory there would be no

---

(3) The word 'leaf' or 'leaves' here represents leaves in the tree structure not leaves on the Sorghum plant.

limitations on the number of branch nodes which would have to be kept. The second major problem was that tree searching is a very time consuming process. It was felt that as much time would be used searching as simulating and if possible this should be eliminated. Therefore a modification was considered which would limit the number of partitions the tree structure would be allowed to diverge into by requiring that any new partition represent five percent or more of the entire field. Once a leaf matured its daily simulation would be discontinued, but the remaining leaves would be simulated on a daily basis until all leaves reached maturity.

This approach seemed to overcome all of the major objections to the previous alternatives considered. The maximum number of leaves which would have to be simulated on any given day would be twenty (4). This limited run time to an acceptable level. Secondly, the critical variables would be able to enter the system when they became critical without requiring any form of catch-up simulation time. However, if a variable which became critical represented less than 5% of the field, then it would not be considered crucial to the model as a whole and divergence would not be allowed. The only remaining concern was the amount of core that would be required to simultaneously simulate twenty plants. Since this approach had generated a considerable

----

(4) There are at most twenty nodes representing 5% or more of a field. However, as will be seen later a twenty-first node is used for data accumulation after maturity.

amount of excitement, this concern was ruled secondary to the potential of the proposed data structure. It was hoped that the tree structure would lend itself to better memory management and that the core requirements could be reduced in the Single Plant Model itself.

Although the tree structure approach had several desirable characteristics, one additional modification was made to take advantage of the fact that with grain sorghum, it is quite common for a field variable which has become critical and has caused a slow-down in the plant's growth, to be eliminated before it causes too much damage. In this case, the plant could 'catch up' to its normal growth level. For example, a lack of available soil water in one area of the field might cause a slow-down in the growth rate in that partition (simulated by a divergence at that point). However, if rain comes, the plants in the partition may speed up growth until they reach the growth level of the rest of the field. To allow for this possibility the proposed data structure was modified from a tree structure with only divergence to a directed graph structure with divergence and convergence.

Convergence in the model is a unique concept that is expected to increase the accuracy of the model. Once a node matures it will be possible to merge the associated data and to free the node space for other uses. This concept of convergence add tremendous potential to the model from the stand point of the Agronomist, and it is perhaps as unique as the proposed data structure itself.

The final conclusion was that a graph structure which allowed dynamic divergence and convergence of nodes would best serve the objectives of the project.

IV. THE PROPOSED DATA STRUCTURE

The core of the Field Level Model is its data structure. Figure 3 illustrates the gradual development of the model's digraph structure. The Source Node represents the field prior to entering the simulation model. The Sink Node at the bottom of the graph is a special node, where all simulation nodes are merged at maturity. It is represented in the program as the twenty-first element of each of the dimensioned variables in the NODBLK COMMON block. Node 1 represents the field as initialized from the reading of global and field variables; that is, one full set of data is read in and stored in the first of the twenty available nodes. At this point the weighting factor for the node is 100%. Immediately following the input of both the global and field data, the emergence routine is called to calculate the days and percent per day that emergence occurs. The Emergence Routine in turn calls the CREATE routine for each additional day emergence occurs. Each call to the CREATE routine represents a divergence of the node presently being processed.

For the example shown in Figure 3, four emergence days are shown (June 1,3,4, and 5). Therefore, CREATE must be called three times. The weighting factor for each node is calculated by the Emergence Routine; for our example, 15%, 35%, 35%, and 15% respectively. (Note that node one simply has had its emergence date established and its weighting factor changed.) After the necessary divergence has occurred

The Digraph Structure

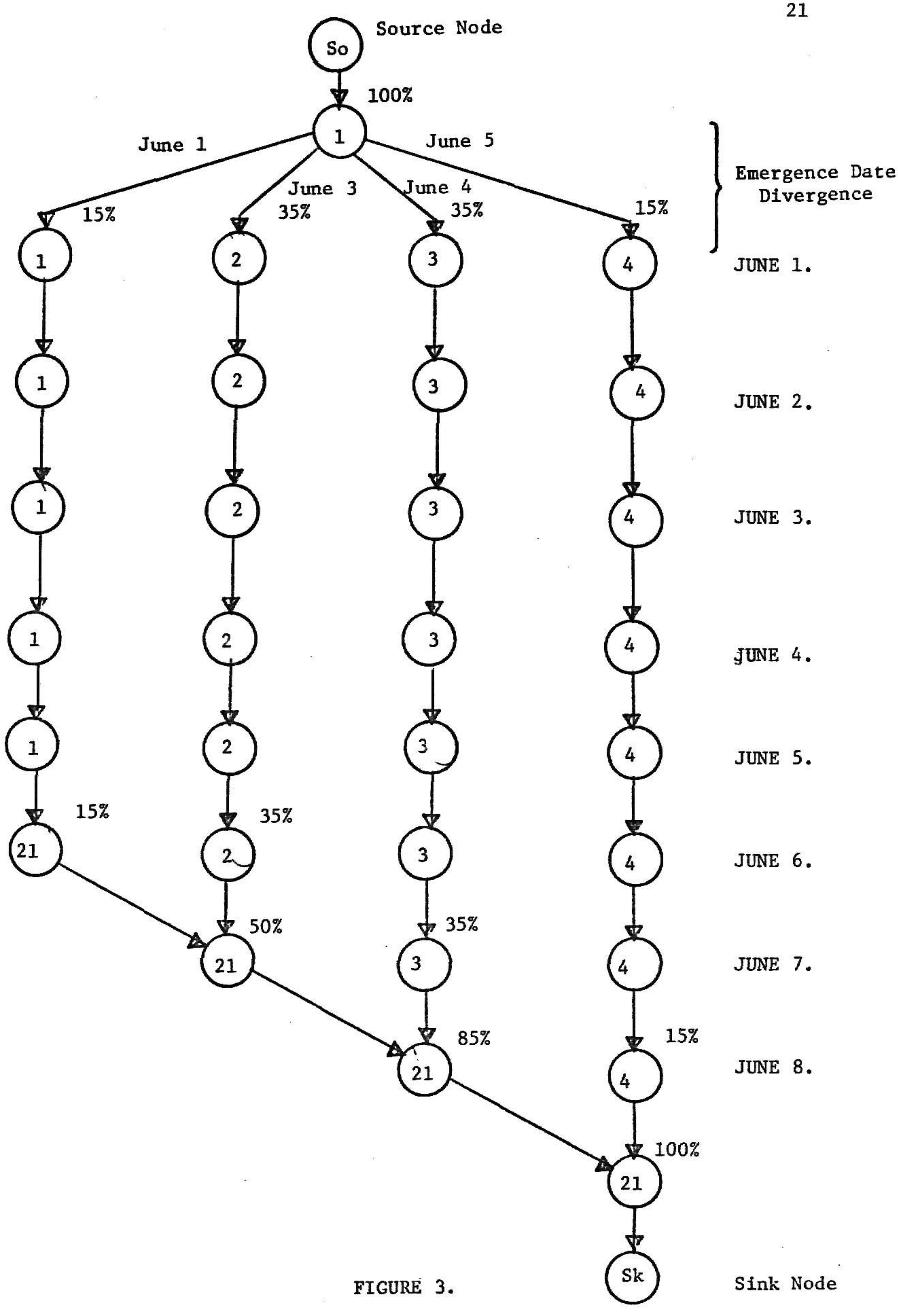

FIGURE 3.

Tree Search Approach

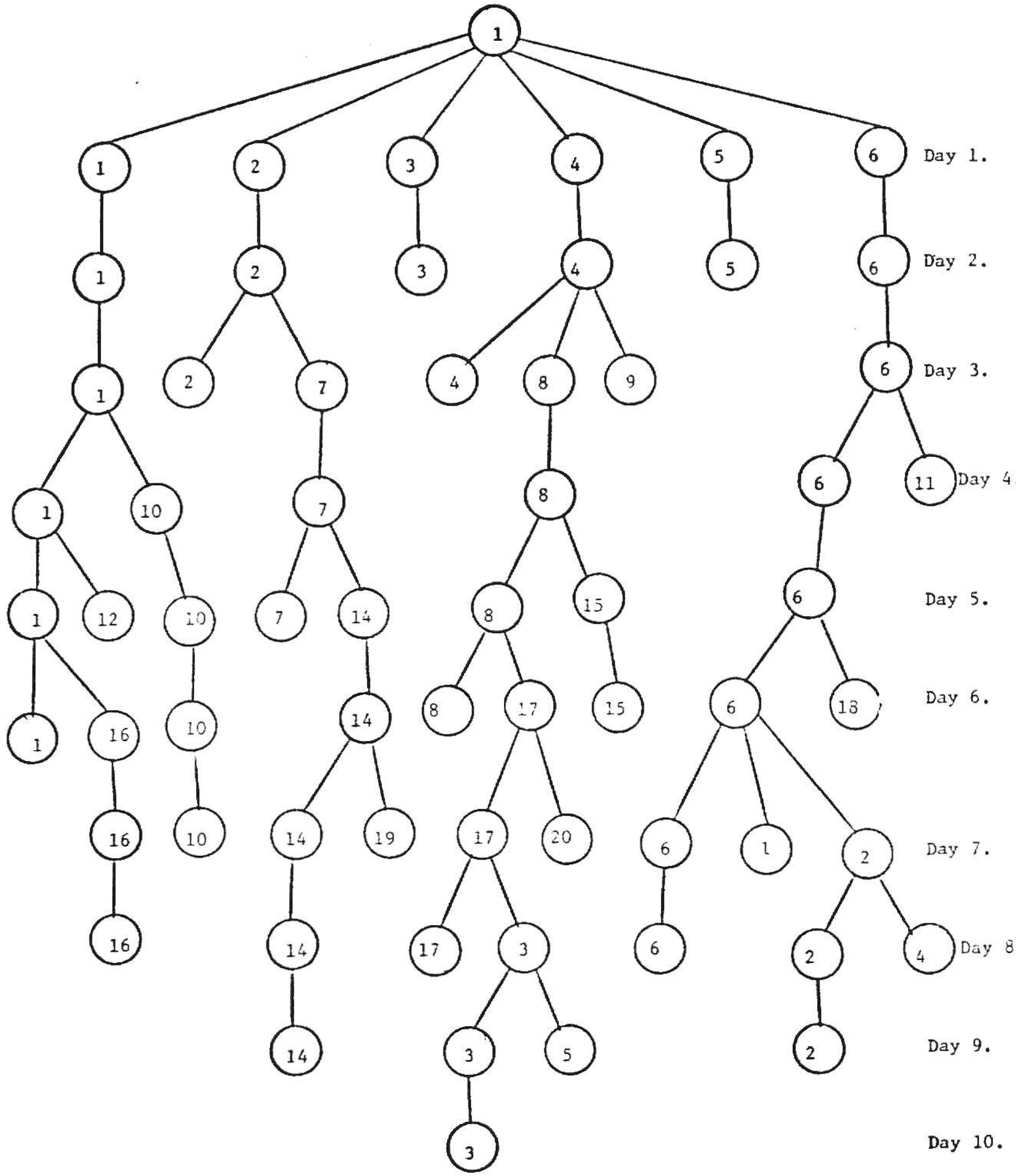

FIGURE 4.

the main routine enters the Daily Simulation Control Routine. At this point the earliest emergence date is located and the daily simulation begins (in the above example June 1).

To simulate a node for a particular day, the Single Plant Model routine (SPMDOL) is called, which in turn follows the previously described procedure for the Single Plant Model (see section II). Once the simulation is completed control passes back to the Daily Simulation Control Routine in the main routine which checks to see if that node has reached maturity. If so, a flag is set which will allow that particular branch to be bypassed from the next day on. The control routine then uses the memory management scheme illustrated in Figure 5 (and described in the next section) to scan each of the remaining active nodes to see if its emergence date has been reached. If it has, then a check is made as to whether or not the node has already reached maturity. If the emergence date has not been reached or if maturity has already occurred then the node is bypassed. Otherwise, the SPMODL is called for each active node. This process continues until all the nodes in the INUSE list (5) have been tested. At that point the date counters are incremented to the next day and the procedure is repeated.

In the example in Figure 3, the SPMODL routine is

----

(5) The INUSE list will be covered in the next section.

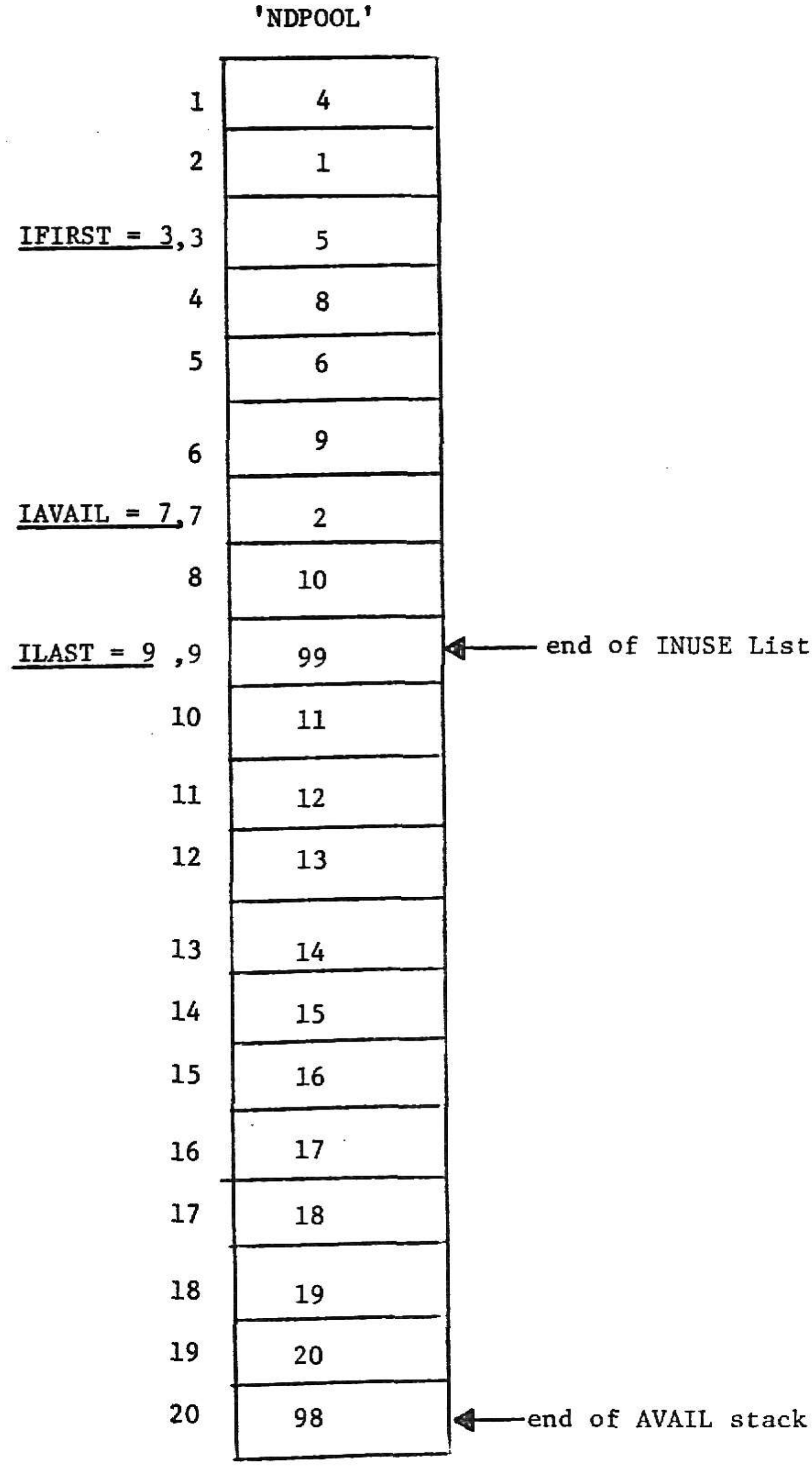
'NDPOOL'
1 4
2 1
IFIRST = 3, 3 5
4 8
5 6
6 9
IAVAIL = 7, 7 2
8 10
ILAST = 9 , 9 99
end of INUSE List
10 11
11 12
12 13
13 14
14 15
15 16
16 17
17 18
18 19
19 20
20 98
end of AVAIL stack

FIGURE 5.

called once for June 1 and once for June 2, both to simulate the 15% of the field represented by node 1. (The yellow nodes represent those nodes which are simulated each day.) On June 3 the SPMODL is called twice; once for node 1 and once for node 2. This process continues with each node entering the system as its emergence date is reached until June 5th where node 1 matures. At this point node 1 is merged into the Sink Node by the MERGE routine. Node 1's space is released by the FRNODE (free node) routine which is called by the MERGE routine and the daily simulation continues with one less node. Finally on June 9th the last node has reached maturity and is merged into the Sink Node (6). The daily simulation control routine senses this condition and passes control back to the main routine where the results are printed from the Sink Node. The field simulation is completed. The main routine next tries to read a new set of global and field data for a totally new field simulation. If no new data is available the program stops.

It is important, at this point, to note that the only convergence which actually occurs in the model as it is, is the converging of nodes 1, 2, 3, and 4 into the Sink Node (node 21). Objective four of this project stipulated that the emergence date variable would be used to test the proposed data structure. This stemmed from the fact that it is the emergence date variable which actually causes the

----

(6) Obviously this example is not realistic.

first change in the various field plants. This change occurs prior to the model's entrance into the Daily Simulation Control Routine. The unfortunate aspect of using the emergence date for a test divergence is that, to date, no relationship has been developed for determining when various plants with different emergence dates can be considered similiar enough in growth to allow the model to converge. This question is still being researched. As a result, with the test data used, the only convergence that occurs is the merging of various nodes into the Sink Node. Section VII on Proposed Extentions will cover this problem in greater detail.

V. MEMORY MANAGEMENT SCHEME

The one major drawback to the proposed data structure was the unknown factor of memory space requirements. This problem was complicated by the fact that the project was limited to a FORTRAN implementation (a stipulation set by the Agronomy Department) which has no instrument for allowing dynamic allocation and deallocation of memory. Because the Field Model would be creating and freeing nodes dynamically during the simulation process, it became necessary to establish some form of memory management to get freed nodes recycled back into the system. To accomplish this, a simple memory management scheme was established using an array of twenty pointers (called NDPOOL) and three special pointers (IAVAIL, IFIRST, and ILAST). The scheme is based upon two data structures; a queue called INUSE and a stack called AVAIL.

The INUSE queue is a linked list used to store pointers to the nodes which are presently in use by the model. IFIRST is a pointer to the node which has been in the queue the longest (called the bottom of the queue) and ILAST points to the last node added to the queue (the top of the queue). When new nodes are used ILAST is set to point to the new node and the node ILAST originally pointed to is set to point to the new node. However, contrary to the normal use of queues, IFIRST is not necessarially the next node to be taken off the queue. However, IFIRST is still of value as the node at which each new day's simulation begins.

The second data structure is the AVAIL stack which is a linked list of nodes which are not in use and which can be used by the model at any time. Access to AVAIL is always through the IAVAIL pointer which always points to the next available node. Since every node will be in one and only one of these two structures, one array can be used to store both structures. NDPOOL serves this function. Figure 5 shows how the INUSE queue and an AVAIL stack are both stored in NDPOOL. In Figure 5, the INUSE queue consists of nodes 3,5,6, and 9, respectively, and the rest of the nodes are on the AVAIL stack.

The first call to CREATE is made prior to reading in the field variables. This sets up the memory management pointers so that the first node is available for reading the field variables. At that point IFIRST is set to one which is the pointer into the INUSE list. NDPOOL(1) is set to 99 representing the end of the INUSE list (ie. there is only one node in use). ILAST is also set to one to point to the last node presently in use. This pointer will be used to add new nodes to the INUSE queue. The remaining NDPOOL elements are initialized to point to the next sequential element (for example, NDPOOL(4)=5, NDPOOL(5)=6, etc.) and the last NDPOOL element (NDPOOL(20)) is set to 98 to represent the end of the AVAIL stack. IAVAIL is set to 2 which is the first available node. This linked list is used as a stack with IAVAIL serving as the pointer for adding to or taking nodes from the AVAIL stack.

All other calls to the CREATE routine add an available node (the node pointed to by IAVAIL) to the end of the INUSE queue. IAVAIL is updated to the next available node. In addition, CREATE copies all node data from the node presently being processed to the newly created node, making no modifications to this data. The necessary modifications such as new weighting factors or emergence dates must be made by the routine calling CREATE after the new node has been created.

The CREATE routine does boundary checking to verify that nodes are available prior to allocating them. This condition should never occur, since nodes of less than 5% are not allowed. However, if it does occur CREATE will print an error message and force an immediate end to the program.

VI. CONVERSION AND TESTING

Once the design of the Field Model was completed the conversion of the Single Plant Model to the Field Model was planned in three phases. The first phase was to modularize the Single Plant Model making it a Daily Simulation Routine (called SPMODL). This routine is called by a special section of the main routine which has been referred to as the Daily Simulation Control Routine (see Figure 3). In addition, a Block Data subroutine was added for load time initialization of most variables and an INITLZ routine was written to give run time initialization of the same variables. The INITLZ routine will be used for multiple simulation runs.

Implementing the daily simulation model involved a re-arrangement of the initial Single Plant Model which already had a daily simulation procedure imbedded within it. The procedure had to be extracted and made into a separate subroutine which would be called once for each node to be simulated for each day. The idea was to test the new structure with the same set of data as the original Single Plant Model. However, the debugging processes turned up several inherent program bugs which existed in the original Single Plant Model. Once these problems were corrected the two models corresponded and phase one was considered accomplished.

The second phase was to completely reorganize and restructure the model's data structure. The Single Plant Model used six COMMON blocks, primarily for parameter passage. Since only parts of any particular block were used by each subroutine and since each subroutine had at least two or three COMMON blocks, this structure was overly complex. Its initial value in straightforward parameter passage had long since been lost in the numerious revisions and updates. As a result, phase two established two COMMON blocks. One block , NODBLK, is for the data relative to each node or leaf (field data) and the second block, COMBLK, is for the various other types of global data. NODBLK includes such items as emergence date, weighting percent, leaf area and leaf appearance dates. COMBLK has various independent variables such as those used in the memory management scheme, plus climatic data and daily light interception figures. In addition to using the two COMMON blocks, normal parameter passing via call statements were established. Again the procedure was to get the model to the point that it would correctly simulate one plant (node). Once the results matched those of the original Single Plant Model phase three, the hardest and most time consuming phase, was initiated.

The third and final phase of conversion involved modifying the new model to take advantage of the proposed digraph structure. The memory management scheme had to be implemented to create and free nodes and to keep track of all nodes in use and available. Divergence due to various emergence dates were forced at several locations within the main routine in an effort to test the model's operation and to assure that all variables had been properly placed with respect to the new COMMON blocks. As set out in the objectives, the Emergence Data Routine was used as a policy for several divergences wherein several emergence dates and maturity dates could be used to test the model's accuracy and sensitivity.

Table 2 shows a summary of a test which forced three emergence divergences. Node 1 represented the test used on the Single Plant Model while node 2 had a forced divergence of two days prior to node 1 and node 3 was given an emergence date three days after node 1. The dates are given in Table 2 as Julian days from January 1. The second entry gives the anthesis or half-bloom point. As is seen in the table all three nodes are converging toward each other. Node 2 is only one day behind node 1 and node 3 is now only two days ahead of node 1 by half bloom. By maturity node 2 was still one day behind while node 1 had caught node 3 with both maturing on Julian day 211. Of course, the particulars of weather was a factor in the test. However, the results were reviewed and found to be in agreement with expected real plant responses.

TABLE

	Node 1	Node 2	Node 3
Emergence Date	0	+3	-2
Half Bloom	210	211	208
Maturity	242	242	240

Table 2. Summary of Field Model test results.

VII. PROPOSED EXTENSIONS

As pointed out in the objectives, 95% of the project was to be oriented toward developing an approach for expanding the Single Plant Model to a field population model and then proving the approach is applicable by expanding the Single Plant Model to include the emergence date field variable. Obviously the emergence date field variable is only one of a number of potential field variables. The consequence of this is that the Field Level Model implemented by the project represents a very limited and narrow application of the proposed approach. This section will discuss how the implemented Field Level Model can be expanded to reach its full potential.

Among those variables which presently are being considered for the Field Level Model are tillering, available soil water, soil type, available nitrogen, hail damage, and seed distribution. It will not take many of these additional field variables to make the digraph structure very dynamic. Figure 6 exemplifies this fact by showing only two additional field variables; water availability and hail damage. It is easy to see the dynamic effects only three field variables could have on the digraph structure. The present Field Level Model was designed to allow easy inclusion of new field variables. There are, however, two new control modules which would have to be added. A Convergence Policy Control Routine (CPCR) and a Divergence Policy Control Routine (DPCR) would have to be included to control the various sets of convergence and divergence policies. Figure 7 shows the proposed extended model's hierarch diagram. The red routines are the proposed extension routines. This proposed extension includes extracting the Daily Simulation Control Routine (DSCR), adding the Convergence Policy Control Routine, the Divergence Policy Control Routine, and the various Convergence and Divergence routines.

The Field Level Model is prepared for these extensions. The Divergence Policy Control Routine should be called prior to beginning a new day's simulation, and the Convergence Policy Control Routine should be called at the end of each day's simulation. Comment cards have been placed in the proper place in the program to indicate where these calls should occur. The objective in this calling arrangement is to allow the Divergence Policy Control Routine to scan the active field partitions (nodes) checking the condition of each node with the various divergence schemes available to it. If a candidate for divergence is found then the routine calls the proper divergence routine which does a detailed analysis of the field partition and, if necessary, calls CREATE to force a divergence. The control then passes back to the Divergence Policy Control Routine where statistical data is saved for later displaying and the rest of the nodes are scanned in the same manner. After all nodes have been scanned by the DPCR, the Daily Simulation Control Routine would be called to begin the simulation process for that day. After all active nodes

Sample Digraph Using Water, Hail, and Emergence

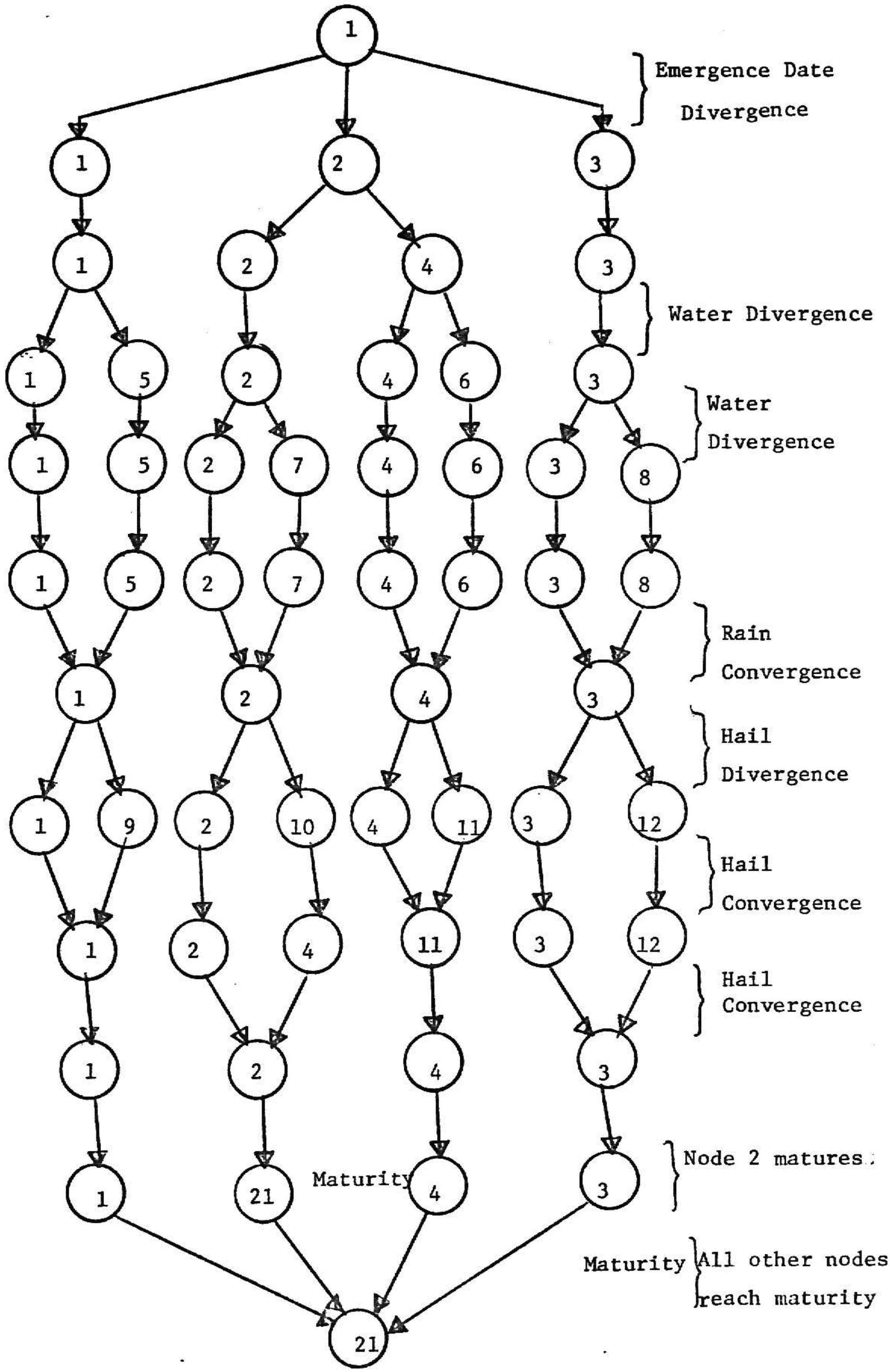

FIGURE 6.

Field Level Model Hierarchy Diagram
with Proposed Extensions

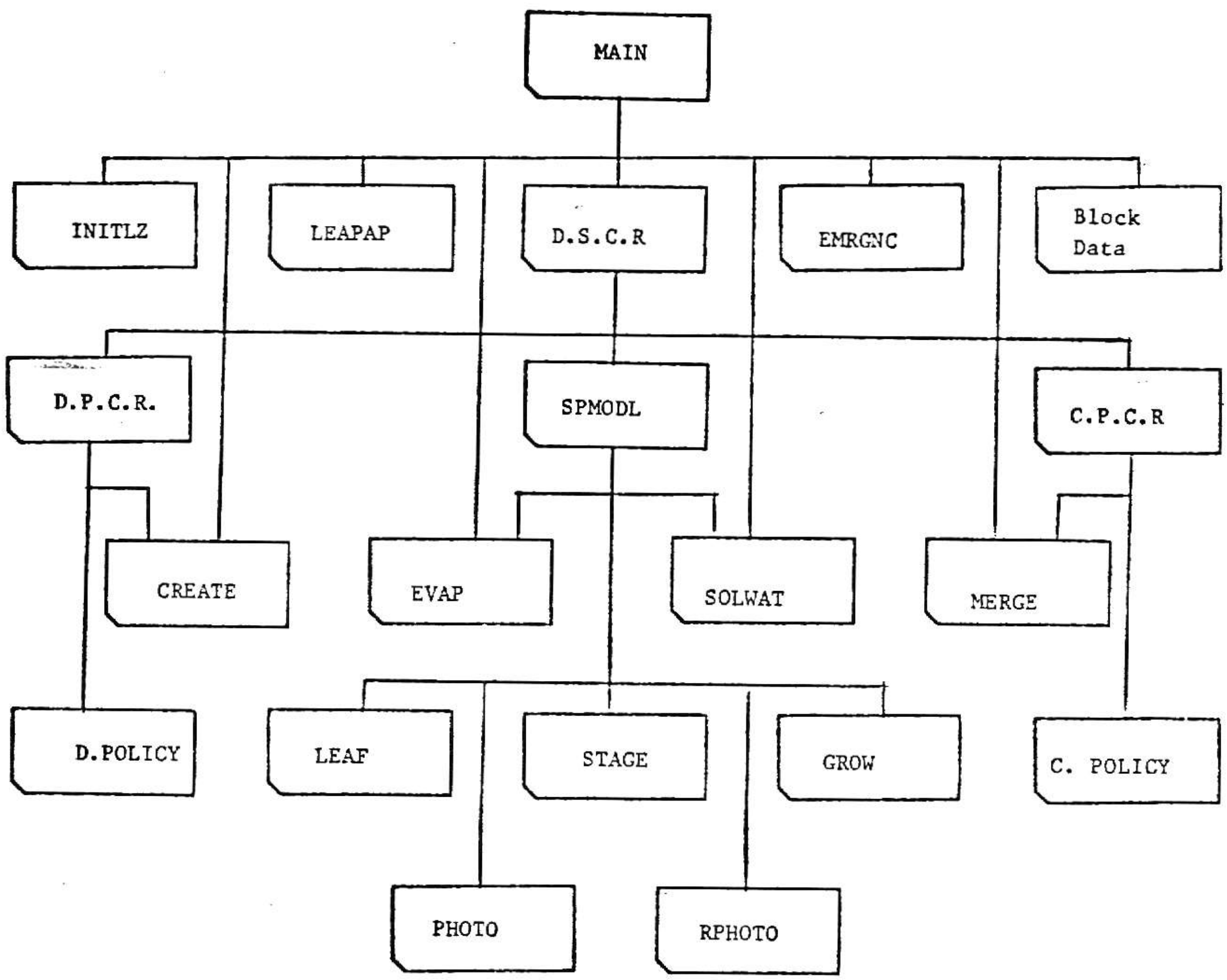

FIGURE 7.

have been simulated control would pass to the Convergence Policy Control Routine which would scan the INUSE list for any conditions which might warrent convergence. The MERGE routine would be called when necessary.

Convergence is allowed only with respect to an earlier devergence. This is to say that two field partitions can not merge simply because they look similar to the model. A convergence occurs, as explain earlier, when a critical field variable which caused a divergence earlier has been sufficiently effected to allow the plant to overcome its growth loss. At this point, and only at this point will convergence be allowed.

In order to facilate this restriction a shared data base will be required between the DPCR and CPCR routines. The DPCR will add new data to the shared data base while the CPCR will access the data base to find candidates for convergence. After merging nodes, the CPCR will be allowed to update the shared data base.

Since the type and form of the information stored in the shared data base is totally dependent upon the convergence policies implemented, it was not a concern of this project to design the format of the shared data base. Again it should be pointed out that this particular structure was not required in the project model because no convergence policies , per se, were required.

VIII. RESULTS AND CONCLUSIONS

The one major concern of the project was memory requirements. Efforts to reduce the memory requirements of the Single Plant Model were successfull in reducing execution requirements from 86K-bytes to 54K-bytes; a 37.2% reduction. The extension of the model raised the requirements to 76K-bytes. This resulted in a net improvement of 11.7% which is much better than was expected.

The field model was not expected to improve run time. In fact, initial estimates were that run time for the field model would increase directly with the number of nodes which had to be simulated. To test the model the Single Plant Model was run for one plant and using the same data, a field model was run forcing three divergences at the beginning. The anticipated run time was expected to be three times longer for the field model. The actual results were 2.7 times longer or slightly better than expected. This does, however, represent a serious consideration which must be taken into account when adding diverging policies; that is, that each daily simulation will take the same relative CPU time and too many diverging nodes can quickly cause drastic increases in run time.

Even under this constraint the proposed model structure offers a new and unique approach to crop population simulation. The feasibility of extending the model to a large number of field variables has been shown. Plans are now being made and funding sought to expand the Field Level Model into four areas: hail damage, a better emergence model, nutrient effects, and tillering.

The conclusion drawn is that the proposed digraph structure is a good and viable approach to extending the Single Plant Model.

REFERENCES

1. Arkin, G. F. , R. L. Vanderlip and J. T. Ritchie. 1976. A Dynamic Grain Sorghum Growth Model. TRANSACTIONS of the ASAE 14:622-630.

2. Arkin,G. F. and J.T. Ritchie. 1973. A Simplified Model for Calculating Light Interception by a Sorghum Canopy. Agronomy Abstracts, American Society of Agronomy, Madison, WI. Page 162.

3. Sharpe, P.J.H, G.L. Curry, D.W.DeMichele, and C.L. Cole. Unpublished. Distribution Model of Organism Development Times.

4. Fick,G.W., W.A. Williams, and R.S. Loomis. 1973. Computer Simulation of Dry Matter Distribution During Sugar Beet Growth. Crop Science. 13:413-417.

5. McKenion, J.M., J.W. Jones, and J.D. Hesketh. 1974. Analysis of SIMCOT: Photosythesis and Growth. Beltwide Cotton Production Research Conf. Proceedings.

6. Ritchie, J.T. 1972. Model for Prediction Evaporation From a Row Crop with Incomplete Cover. Water Resources Res. 8:1204-1213.

7. Ritchie,J.T. 1973. Influence of soil Water Status and Meteorological Conditions on Evaporation from a Corn Canopy. Agron. J. 65:893-897.

8. Thornley,J.H.M. 1971. A Model to Describe the Partitioning of Photosynthate during Vegetative Plant Growth. Ann. Bot. 33:419-30.

APPENDIX A

Original Main Routine

```
      COMMON/BLK1/GRO(366,20),XMAX(20),PDAYS(20),ROSPZ,PAREA,TEMPCO,     CAR00180
     1DAYPFO,LAT,WYE,BO,R2                                               CAR00190
      COMMON/BLK2/TEMPMX(366),TEMPMN(366),TEMP(366),SPROUT,FMRGDA        CAR00200
      COMMON/BLK3/DLAI(366),SOLRAD(366),RAIN(366),SW,WATSCO,EOS,EO,UL,PLCAR00210
     1L                                                                  CAR00220
      COMMON/BLK4/RCOUNT(20),ACDAYS(20),N,IDAY3,IDAY6,IDAY9,ISTAGE,DLA   CAR00230
      COMMON/BLK5/F,WR,WL,WC,WH,WG,DRIWT,DWG,DWH,DWC,DWL,DWR             CAR00240
      DIMENSION XDAYS(21),SXDIN(20),TITLE(20),ICOR(12),FMT(20)           CAR00250
      DATA HO,TOTWT/2*0./
      DATA ICOR/1,-1,0,0,1,1,2,3,3,4,4,5/
      REAL LAT                                                           CAR00290
   98 READ(5,1301,END=100) TITLE
 1301 FORMAT(20A4)                                                       CAR00310
      READ(5,1302) KI, N, MO,ND,IYR,ROSPZ,P,LAT,SW,UL,IFREQ
 1302 FORMAT (5I4,F5.1,F10.0,3F5.2,T77,I4)
      READ(5,1000) (XMAX(J),J=1,N)                                       CAR00340
 1000 FORMAT(10F7.2)                                                     CAR00350
      IF (MOD(IYR , 4)) 1563,1564,1563                                   CAR00360
 1564 LPY =1                                                             CAR00370
      GO TO 1569                                                         CAR00380
 1563 LPY =0                                                             CAR00390
 1569 ICDAY=30*(MO-1)+ICOR(MO-1)+ND+LPY                                  CAR00400
      IMAX=ICDAY+KI-1                                                    CAR00410
      DO 5 J=1,N
      PDAYS(J)=0.
      RCOUNT(J)=0.
      ACDAYS(J)=0.
      DO 6 I=ICDAY,IMAX
      GRO(I,J)=0.0                                                       CAR00510
    6 CONTINUE
    5 CONTINUE                                                           CAR00520
      WR=0.
      WL=0.
      WC=0.
```

```
      WH=0.
      WG=0.
      ISTAGE=0
      IDAY3=0
      IDAY6=0
      IDAY9=0
      FMRGDA=0
      WRITE(6,1571) TITLE,ICDAY,MO,ND,IYR                               CAR00420
 1571 FORMAT(1H1,20A4//,5X,'PLANTING DATE=',I4,6X,'CALENDAR DAY=',3I6)  CAR00430
      IF(KI.EQ.0) GO TO 7
      READ(5,1301) FMT                                                  CAR00440
      DO 4 K=1,KI                                                       CAR00460
      READ(5,FMT) (I,TEMPMX(I),TEMPMN(I),SOLRAD(I),RAIN(I))
      AVTEMP=(TEMPMX(I)+TEMPMN(I))/2.                                   CAR00470
      TEMP(I)=(AVTEMP-32.)*5./9.                                        CAR00480
      RAIN(I)=RAIN(I)*2.54                                              CAR00490
    4 CONTINUE                                                          CAR00530
    7 CONTINUE
      CALL EMRGNC (ICDAY)                                               CAR00540
      I = SPROUT- (1.-FMRGDA) + .5                                      CAR00550
      ISPRT=SPROUT+.5                                                   CAR00560
      XDAYS(1)=0.0                                                      CAR00570
      XDAYS(2)=FMRGDA                                                   CAR00580
 1104 DO 102 J=1,N                                                      CAR00590
      IF(J.NE.1) GO TO 1085                                             CAR00600
      RCOUNT(J)=SPROUT                                                  CAR00610
      GO TO 1084                                                        CAR00620
 1085 SXDIN(J)=0.0                                                      CAR00630
      PDAYS(J-1)=XDAYS(J)                                               CAR00640
      IF(XDAYS(2).GT.0.0) DAYS=SPROUT                                   CAR00650
      IF(XDAYS(J).GT.0.) GO TO 13                                       CAR00660
  103 I=I+1                                                             CAR00670
      DAYS=I                                                            CAR00680
      IF(DAYS.GT.IMAX) GO TO 104
   13 AVTEMP=TEMP(I)                                                    CAR00690
      IF(AVTEMP.GT.22) AVTEMP=22.                                       CAR00700
      IF(J.LT.6) GO TO 101                                              CAR00710
      IF(AVTEMP.LT.21.) GO TO 108                                       CAR00720
```

```
      IF(TEMP(I).GT.30.) AVTEMP=30.                                     CAR00730
      DIN=2.957-.0562*(AVTEMP-21)                                       CAR00740
      GO TO 18                                                          CAR00750
  108 DIN=2.9+.0567*(AVTEMP-22.)**2                                     CAR00760
      GO TO 18                                                          CAR00770
  101 DIN=2.8+.0292*((AVTEMP-22.)**2)                                   CAR00780
   18 IF(XDAYS(J).EQ.0.) GO TO 107                                      CAR00790
      XDIN=(1./DIN)*XDAYS(J)                                            CAR00800
      XDAYS(J)=0.0                                                      CAR00810
      GO TO 14                                                          CAR00820
  107 XDIN=1./DIN                                                       CAR00830
   14 SXDIN(J)=XDIN+SXDIN(J)                                            CAR00840
      IF (1.0-SXDIN(J)) 106,105,103                                     CAR00850
  105 RCOUNT(J)=DAYS                                                    CAR00860
      XDAYS(J)=0.0                                                      CAR00870
      GO TO 1108                                                        CAR00880
  106 XDAYS(J)= (SXDIN(J)-1.0)*DIN                                      CAR00890
      RCOUNT(J)= DAYS-XDAYS(J)                                          CAR00900
 1108 XDAYS(J+1)=XDAYS(J)                                               CAR00910
      IF (J.EQ.N) PDAYS(J)=XDAYS(J)                                     CAR00920
 1084 WRITE(6,19) J,RCOUNT(J)                                           CAR00930
   19 FORMAT(5X, 'DAY LEAF',I3,' APPEARS = ',F7.2)                      CAR00940
  102 CONTINUE                                                          CAR00950
  104 CONTINUE
      PLL=UL*.25                                                        CAR00960
      PAREA=100000000/P                                                 CAR00970
      R2=PAREA/ROSPZ                                                    CAR00980
      S=R2/ROSPZ                                                        CAR00990
      WYE=0.404*ALOG(S)+1.49                                            CAR01000
      BO=EXP(-0.384*WYE)                                                CAR01010
      WRITE(6,22) P,ROSPZ,R2,PAREA                                      CAR01020
   22 FORMAT(////' PLANTS PER HECTARE=',1X,F10.0,/,' ROW SPACING=',1X,
     1F6.2/,' PLANT SPACING=',1X,F6.1,/,' AREA PER PLANT=',1X,F8.2)     CAR01040
      WRITE(6,1001)                                                     CAR01050
 1001 FORMAT(////4X,'JULIAN DAY',3X,'STAGE',2X,'ROOT WT',3X,'LEAF WT'
     1,3X,'CULM WT',3X,'HEAD WT',3X,'GRAIN WT',2X,'TOTAL WT',3X,'LA,CM',CAR01070
     27X,'LAI',3X,'TCO',2X,'WCO',3X,'SW')
      DO 200 I=ISPRT,IMAX                                               CAR01090
```

```
      CALL LEAF (I)                                                      CAR01100
      CALL EVAP (I)                                                      CAR01110
      CALL SOLWAT (I)                                                    CAR01120
      CALL STAGE (I)                                                     CAR01130
      CALL PHOTO (I)                                                     CAR01140
      IF (TEMP(I).LT.5.0.OR.TEMP(I).GE.45.0) TEMPCO=0.                   CAR01150
      IF (TEMP(I).GE.25.0.AND.TEMP(I).LE.40.0) TEMPCO=1.                 CAR01160
      IF(TEMP(I).GE.5.0.AND.TEMP(I).LE.25.) TEMPCO=.05*TEMP(I)-.25       CAR01170
      IF (TEMP(I).GT.40.0.AND.TEMP(I).LT.45.) TEMPCO=-.2*TEMP(I)+9.      CAR01180
      TOFOTO=TEMPCO*WATSCO*DAY PFO                                       CAR01190
      TEMPN=(TEMPMN(I)+TEMP(I))/2.                                       CAR01200
      RESPG=.14*(.65+.011*TEMPN)                                         CAR01210
      RESPM=.0054*(.044+.0019*TEMPN+.001*TEMPN*TEMPN)                    CAR01220
      RESPN=RESPG*TOFOTO+RESPM*(TOTWT+WR)*1.43                           CAR01230
      TOFOTO=TOFOTO-RESPN                                                CAR01240
      IF(TOFOTO.EQ.0.) WRITE(6,1003) TEMPCO, WATSCO, DAYPFO              CAR01250
 1003 FORMAT(1X,'TEMPCO=',F4.2,5X,'WATSCO=',F4.2,5X,'DAYPFO=',F5.1)      CAR01260
      DRIWT=TOFOTO*(PAREA/100.)*(12./44.)*(1./.4)*.001                   CAR01270
      CALL GROW (I)                                                      CAR01280
      TOTWT=WL+WC+WH+WG                                                  CAR01290
      IPRINT=I-ISPRT                                                     CAR01300
      IF(ISTAGE.EQ.5) GO TO 1006                                         CAR01310
      IF(I.NE.IDAY6) GO TO 1007                                          CAR01320
      WRITE(6,1005) I                                                    CAR01330
 1005 FORMAT(1X,'ANTHESIS OCCURRED ON JULIAN DAY',I4)                    CAR01340
 1007 IF(MOD(IPRINT,IFREQ)) 200,1006,200
 1006 WRITE(6,2000)I,IPRINT,ISTAGE,WR,WL,WC,WH,WG,TOTWT,DLA,DLAI(I),     CAR01360
     1TEMPCO,WATSCO,SW                                                   CAR01370
 2000 FORMAT(6X,I3,'(',I3,')',5X,I2,8F10.3,2F5.1,2F6.1)                  CAR01380
      IF(ISTAGE.EQ.5) GO TO 201                                          CAR01390
  200 CONTINUE                                                           CAR01400
      IF(ISTAGE.EQ.5) GO TO 201                                          CAR01410
      WRITE(6,1004) IDAY9                                                CAR01420
 1004 FORMAT(/1X,'PHYSIOLOGICAL MATURITY NOT REACHED.'/1X,'PREDICTED ON  CAR01430
     1 JULIAN DAY',2X,I4///)                                             CAR01440
       GO TO 99                                                          CAR01450
  201 WRITE(6,1002) I                                                    CAR01460
 1002 FORMAT(/1X,'PHYSIOLOGICAL MATURITY OCCURRED ON JULIAN DAY',I4///)  CAR01470
```

```
 99 CONTINUE
    GO TO 98
100 STOP
    END
```

APPENDIX B

FIELD MODEL SOURCE LISTING

```
$JOB           JWC,PAGES=20,TIME=(1,0),NOLIST
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
       COMMON/SPECAL/ISPRT(20)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
      DATA HO,TOTWT/2*0./
      DIMENSION  TITLE(20),ICOR(12),FMT(20),ACTW(366)
      DATA ICOR/1,-1,0,0,1,1,2,3,3,4,4,5/
      REAL LAT
 1571 FORMAT(1H1,20A4//,5X,'PLANTING DATE=',I4,6X,'CALENDAR DAY=',3I6)   CAR00430
 1000 FORMAT(10F7.2)                                                      CAR00350
1302  FORMAT (5I4,F5.1,F10.0,3F5.2,T77,I4)
 1301 FORMAT(20A4)                                                        CAR00310
 1002 FORMAT(/1X,'PHYSIOLOGICAL MATURITY OCCURRED ON JULIAN DAY',I4///)  CAR01470
 1004 FORMAT(/1X,'PHYSIOLOGICAL MATURITY NOT REACHED.'/1X,'PREDICTED ON CAR01430
     1 JULIAN DAY',2X,I4///)                                              CAR01440
 2000 FORMAT(6X,I3,'(',I3,')',5X,I2,8F10.3,2F5.1,F6.1,3X,I2)              CAR01380
 1005 FORMAT(1X,'ANTHESIS OCCURRED ON JULIAN DAY',I4)                     CAR01340
CC
CC      THE FIRST PASS THROUGH- ALL VARIBLES ARE INITILIZED BY
CC       THE BLOCK DATA ROUTINE THEREFORE THE CALL INITLZ IS
CC      BYPASSED.  ALL OTHER PASSES CALL INITLZ TO INITILIZE
CC      NECCESSARY VARIABLES.
CC
CC
       DO 21 I=1,20
```

```
       IDAY(I)=0
      DO 24 K=1,20
      DO 23 J=1,2
   23 GRO(I,J,K)=0.
   24 CONTINUE
   21 CONTINUE
      NODE=0
            CALL CREATE
            PERCNT(NODE)=.20
      LOOP=0
   97 IF(LOOP.NE.0)CALL INITLZ
      LOOP=1
CC         -------- READ DATA ROUTINE --------
CC
   98 READ(5,1301,END=100) TITLE
C
      READ(5,1302) KI, N, MO,ND,IYR,ROSPZ,P,LAT,SW(NODE),UL(NODE),IFREQ
       IFREQ=1
      READ(5,1000) (XMAX(NODE,J),J=1,N)                                 CAR00340
C
C             CONVERT JULIAN DATES TO CALENDAR DATES
C
      IF (MOD(IYR , 4)) 1563,1564,1563                                  CAR00360
1564  LPY =1                                                            CAR00370
      GO TO 1569                                                        CAR00380
1563  LPY =0                                                            CAR00390
 1569 ICDAY=30*(MO-1)+ICOR(MO-1)+ND+LPY                                 CAR00400
CC
CC     ------ PRINT HEADINGS AND TITLE ------
CC
      WRITE(6,1571) TITLE,ICDAY,MO,ND,IYR                               CAR00420
CC
C
C             READ RAIN AND TEMP DATA AND CONVERT TO METRIC
C
      IF(KI.EQ.0) GO TO 7
      IMAX=ICDAY+KI-1                                                   CAR00410
      READ(5,1301) FMT                                                  CAR00440
```

```
      DO 4 K=1,KI                                                        CAR00460
      READ(5,FMT) I,TEMPMX(I),TEMPMN(I),SOLRAD(I),RAIN(I)
      TEMPMX(I)=(TEMPMX(I)-32.)*5./9.
      TEMPMN(I)=(TEMPMN(I)-32.)*5./9.
      TEMP(I)=(TEMPMX(I) + TEMPMN(I))/2.
      RAIN(I)=RAIN(I)*2.54                                               CAR00490
CC
C       CALCULATE EVAPORATION AND SOILWATER CONTENT WITHOUT PLANT'S
C       PRESENCE IN THE SOIL, THEN FIND EMERGENCE DATE
C
      CALL EVAP (I)
      CALL SOLWAT (I)
      ACTW(I)=SW(NODE)
    4 CONTINUE                                                           CAR00530
    7 CONTINUE
      CALL EMRGNC (ICDAY)                                                CAR00540
CC
CC          ISPRT(NODE) = DAY NUMBER OF EMERGANCE FOR THE 'NODE'TH GROUP
CC
C     TEST CALL
          CALL CREATE
          PERCNT(ILAST)=.40
         SPROUT(ILAST)=SPROUT(NODE)+3.
         CALL CREATE
          SPROUT(ILAST)=SPROUT(NODE)-2.
          PERCNT(ILAST)=.40
          NODE=IFIRST
         IBEGIN=366
C
C                 CALCULATE LEAF APPEARENCE DATES
C                   FOR ALL INITIL NODES
C                 PRINT THESE DAT A TIMEES FIVE NODES AT
C                ALSO FIND THE SMALLEST EMERGENCE DATE FOR
C                     BEGINNING THE SIMULATION LOOP FOLLOWING
C
   25  I=SPROUT(NODE)-(1.-FMRGDA) + .5
       ISPRT(NODE)=SPROUT(NODE)+ .5
      CALL LEAFAP (J,I,IMAX)
```

```
      IF(IBEGIN.GE.ISPRT(NODE))IBEGIN=ISPRT(NODE)
      PLL(NODE)=UL(NODE)*.25
      NODE=NDPOOL(NODE)
      IF(NODE.NE.99)GO TO 25
      NODE=IFIRST
CC
C             CALCULATE PLANT SPACING DATA
C
      PAREA=100000000/P                                              CAR00970
      R2=PAREA/ROSPZ                                                 CAR00980
      S=R2/ROSPZ                                                     CAR00990
      WYE=0.404*ALOG(S)+1.49                                         CAR01000
      BO=EXP(-0.384*WYE)                                             CAR01010
      WRITE(6,22) P,ROSPZ,R2,PAREA                                   CAR01020
   22 FORMAT(////' PLANTS PER HECTARE=',1X,F10.0,/,' ROW SPACING=',1X,
     1F6.2/,' PLANT SPACING=',1X,F6.1,/,' AREA PER PLANT=',1X,F8.2)  CAR01040
      WRITE(6,1001)                                                  CAR01050
 1001 FORMAT(////4X,'JULIAN DAY',3X,'STAGE',2X,'ROOT WT',3X,'LEAF WT'
     1,3X,'CULM WT',3X,'HEAD WT',3X,'GRAIN WT',2X,'TOTAL WT',3X,'LA,CM',CAR01070
     27X,'LAI',3X,'TCO',2X,'WCO',3X,'SW')
C
C             DAILY GROWTH SIMULATION
C
      DO 200 I=IBEGIN,IMAX                                           CAR01090
              NODE=IFIRST
   10         CONTINUE
C
C        INSERT DIVERGENCE POLICY CONTROL ROUTINE HERE
C
              IF (MATURE(NODE).EQ.1) GO TO 11
CC
CC          CALL THE SINGLE PLANT MODLE TO CALCULATE DAILY GROWTH
CC
      IF(I.LE.ISPRT(NODE))GO TO 11
      IF(I.EQ.ISPRT(NODE))SW(NODE)=ACTW(I)
      CALL SPMODL (I, TOTWT)
CC
CC           CHECK TO SEE IF REQUESTED PRINT POINT HAS
```

```
CC            BEEN REACHED,   IF SO PRINT,   ELSE CCONTINUE
CC
C
C       IPRINT IS THE NUMBER OF DAYS OF THE NODE
C              FROM EMERGENCE
C
      IPRINT(NODE)=I-ISPRT(NODE)                                        CAR01300
      IF(ISTAGE(NODE).EQ.5) GO TO 1006                                  CAR01310
      IF(I.NE.IDAY6(NODE)) GO TO 1007                                   CAR01320
      WRITE(6,1005) I                                                   CAR01330
 1007 IF(MOD(IPRINT(NODE),IFREQ)) 11,1006,11
 1006 WRITE(6,2000)I,IPRINT(NODE),ISTAGE(NODE),WR(NODE),WL(NODE),       CAR01360
     1 WC(NODE),WH(NODE),WG(NODE),TOTWT,DLA(NODE),DLAI(I),              CAR01370
     2 TEMPCO,WATSCO,SW(NODE),NODE
C
C        IF NODE IS MATURE ALREADY THEN MATURE(NODE)=1
C
C     IF(ISTAGE(NODE).EQ.5) GO TO 201
              IF(ISTAGE(NODE).NE.5) GO TO 11
              WRITE (6,1002) I
             MATURE(NODE)=1
   11         CONTINUE
C
C       INSERT CONVERGENCE CONTROL ROUTINE HERE
C
              NODE=NDPOOL(NODE)
              IF(NODE.NE.99) GO TO 10
  200 CONTINUE                                                          CAR01400
              NODE=IFIRST
   95         CONTINUE
      IF (MATURE(NODE).NE.1) GO TO 96
       CALL MERGE (NODE,21)
       GO TO 99
   96 CONTINUE
      IF(ISTAGE(NODE).EQ.5) GO TO 201
      WRITE(6,1004) IDAY9(NODE)                                         CAR01420
       GO TO 99                                                         CAR01450
  201 WRITE(6,1002) I                                                   CAR01460
```

```
   99 CONTINUE
              NODE=NDPOOL(NODE)
      IF(NODE.NE.99) GO TO 95
C
C     FINAL NODE HAS MATURED--PRINT SINK NODE VALUES
C
       FLDWT=WL(21)+WC(21)+WH(21)+WG(21)
       WRITE(6,1003)
 1003 FORMAT (3X,'FIELD MATURITY REACHED')
       WRITE (6,1008) WL(21),WC(21),WH(21),WG(21),FLDWT
 1008 FORMAT (3X,'LEAF= ',F10.3,3X,'CLUM= ',F10.3,3X,'HEAD= ',F10.3,3X
     1,'GRAIN= ',F10.3,3X,3X,'FINAL TOTAL WT = ',F10.3)
      GO TO 97
  100 STOP
      END
      BLOCK DATA
       COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
       COMMON/SPECAL/ISPRT(20)
      COMMON/COMBLK/ROSPZ,PAREA,N,                                      CARO
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,CRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
       DATA DLAI,UL,SW,PLL,SUMES1,SUMES2,ISPRT/366*0.,100*0.,20*0/
      DATA PDAYS,XMAX/800*0./,RCOUNT/400*0./,ACDAYS/400*0./
       DATA SPROUT/20*0./,DLA/20*0./
      DATA WL,WC,WH,WG,SGDD,PMGDD,FMRGDA,WR,ISTAGE,
     1 IDAY3,IDAY6,IDAY9,IFREQ,IDAY/108*0.,80*0,5,20*0/
      DATA IAVAIL,IFIRST,NDPOOL,MATURE/1,99,40*0/
      DATA NODE,JDAY,ICOMP/41*0/,PL,YDLA/40*0./
       DATA PERCNT/20*0.,100./
      END
              SUBROUTINE CREATE
```

```
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/SPECAL/ISPRT(20)
      COMMON/COMBLK/ROSPZ,PAREA,N,                                      CARO
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
C
C      IF NODE=0 THEN INITILIZE MEMORY MANAGEMENT POINTERS
C
      IF (NODE.NE.0)GO TO 5
       NODE=1
      IFIRST=1
       IAVAIL=2
       NDPOOL(1)=99
      ILAST=1
      DO 20 I=2,19
   20 NDPOOL(I)=I+1
      NDPOOL(20)=99
       RETURN
    5 IF (IAVAIL.NE.99) GO TO 10
  100  FORMAT(1X,'ERROR--NODE OVERFLOW,  IN CREATE SUBROUTINE')
        WRITE(5,100)
       STOP
C
C     COPY DATA FROM PRESENT NODE TO NEXT AVAIL NODE
C
   10 DO 15 I=1,20
      GRO(IAVAIL,1,I)=GRO(NODE,1,I)
      PDAYS(IAVAIL,I)=PDAYS(NODE,I)
      GRO(IAVAIL,2,I)=GRO(NODE,2,I)
      XMAX(IAVAIL,I)=XMAX(NODE,I)
      RCOUNT(IAVAIL,I)=RCOUNT(NODE,I)
```

```
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMCN/COMBLK/F,CRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
C
C     THIS ROUTINE MERGES THE FRCM-NODE (NODEF)
C     INTO THE TO-NODE (NODET) WHERE PERCENT (NODET)
C     IS GT PERCENT (NODEF)
C
C     - AS IMPLEMENTED ONLY DRY MATTER OF PLANT PARTS
C      AND PERCENT ARE MERGED
C     -WHEN CONVERGENCE POLICIES ARE ADDED THIS ROUTINE
C      MUST BE EXPANDED
C
      IF(NODET.EQ.21)GO TO 50
      IF (PERCNT (NODET).GE.PERCNT(NODEF))GO TO 50
      NDSAVE = NODET
      NODET = NODEF
      NODEF = NDSAVE
   50 CONTINUE
C
C     COMBINE PLANT PARTS AND PERCENT
C
      WR(NODET) = WR (NODET)+WR(NODEF)
      WL(NODET) = WL (NODET)+WL(NODEF)
      WC(NODET) = WC (NODET)+WC(NODEF)
      WH(NODET) = WH (NODET)+WH(NODEF)
      WG(NODET) = WG(NODET)+WG(NODEF)
      PERCNT(NODET)=PERCNT(NODET)+PERCNT(NODEF)
C     CALL FRNODE (NODEF)
       WRITE(6,1000) NODEF,NODET
       WRITE(6,1005)
```

```
      ISPRT(IAVAIL)=ISPRT(NODE)
      ACDAYS(IAVAIL,I)=ACDAYS(NODE,I)
   15 CONTINUE
      IDAY(IAVAIL)=IDAY(NODE)
      PL(IAVAIL)=PL(NODE)
      YDLA(IAVAIL)=YDLA(NODE)
      JDAY(IAVAIL)=JDAY(NODE)
      ICOMP(IAVAIL)=ICOMP(NODE)
      MATURE(IAVAIL)=MATURE(NODE)
      SPROUT(IAVAIL)=SPROUT(NODE)
      SW(IAVAIL)=SW(NODE)
      UL(IAVAIL)=UL(NODE)
      PLL(IAVAIL)=PLL(NODE)
      IDAY3(IAVAIL)=IDAY3(NODE)
      IDAY6(IAVAIL)=IDAY6(NODE)
      IDAY9(IAVAIL)=IDAY9(NODE)
      ISTAGE(IAVAIL)=ISTAGE(NODE)
      DLA(IAVAIL)=DLA(NODE)
      WR(IAVAIL)=WR(NODE)
      WL(IAVAIL)=WL(NODE)
      WC(IAVAIL)=WC(NODE)
      WH(IAVAIL)=WH(NODE)
      WG(IAVAIL)=WG(NODE)
      SUMES1(IAVAIL)=SUMES1(NODE)
      SUMES2(IAVAIL)=SUMES2(NODE)
      T(IAVAIL)=T(NODE)
C
C       UPDATE NODE POOL ARRAY NDPOOL
C
     ITEMPX=NDPOOL(IAVAIL)
       NDPOOL(IAVAIL)=99
       NDPOOL(ILAST)=IAVAIL
         ILAST=IAVAIL
       IAVAIL=ITEMPX
       RETURN
              END
      SUBROUTINE MERGE (NODEF,NODET)
     COMMON/NODBLK/NODX,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
```

```
 1005  FORMAT (1H1)
 1000  FORMAT (2X,'NODE ',I2,1X,'MERGED INTO NODE ',I2)
      RETURN
      END
       SUBROUTINE FRNODE (NODE)
      COMMON/NODBLK/NODX,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
C     FIND FDPOOL NODE POINTING TO NODE TO BE DELETED-ISAVE
C     FIND FDPOOL NODE POINTING TO ILAST NODE- LSAVE
C
       ISAVE=0
       LSAVE=0
       DO 50 I=1,20
       IF (NDPOOL(I).EQ.NODE) ISAVE=I
       IF (NDPOOL(I).EQ.ILAST) LSAVE=I
   50 CONTINUE
C
C     IF NODE TO BE DELETE IS IFIRST THEN UPDATE IFIRST
C     UPDATE NDPOOL NODE POINTING TO NODE TO BE DELETED
C     UPDATE IAVAIL TO NEW FREED NODE
       IF(NODE.EQ.ILAST) ILAST=LSAVE
       IF (NODE.EQ.IFIRST) IFIRST=NDPOOL (IFIRST)
       IF (ISAVE.NE.0) NDPOOL(ISAVE)=NDPOOL (NODE)
       NDPOOL(NODE) = IAVAIL
       IAVAIL = NODE
       RETURN
       END
      SUBROUTINE SPMODL (I, TOTWT)
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
```

```
     1PINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,                                      CAR00180
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2                                       CAR00190
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA            CAR00200
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO       CAR00210
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC                                 CAR00240
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20))
      REAL LAT                                                          CAR00290
      CALL LEAF (I)                                                     CAR01100
      CALL EVAP (I)
      CALL SOLWAT (I)                                                   CAR01120
      CALL STAGE (I,N)                                                  CAR01130
      CALL PHOTO (I)                                                    CAR01140
      CALL RPHOTO(I,TOTWT,TOFOTO)
      IF(TOFOTO.EQ.0.) WRITE(6,1003) TEMPCO, WATSCO, DAYPFO             CAR01250
 1003 FORMAT(1X,'TEMPCO=',F4.2,5X,'WATSCO=',F4.2,5X,'DAYPFO=',F5.1)     CAR01260
      DRIWT=TOFOTO*(PAREA/100.)*(12./44.)*(1./.4)*.001                  CAR01270
      CALL GROW (I)                                                     CAR01280
      TOTWT=WL(NODE)+WC(NODE)+WH(NODE)+WG(NODE)                         CAR01290
      RETURN
      END
      SUBROUTINE INITLZ
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
       COMMON/SPECAL/ISPRT(20)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
       COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,CRIWT,CWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
      DO 10 NODE=1,20
```

```
      DO 5 J=1,20
      RCOUNT(NODE,J)=0.
       PDAYS(NODE,J)=0.
       ISPRT(J)=0.
      XMAX(NODE,J)=0.
      ACDAYS(NODE,J)=0.
       XDAYS(NODE,J)=0.
    5 CONTINUE                                                          CAR00520
                MATURE(NODE)=0
      ISTAGE(NODE)=0
      WL(NODE)=0.
      WC(NODE)=0.
      WH(NODE)=0.
       WG(NODE)=0.
      WR(NODE)=0.
      IDAY(NODE)=0
      IDAY3(NODE)=0
      IDAY6(NODE)=0
      IDAY9(NCDE)=0
      DO 6 ITEMPX=1,20
      DO 7 I=1,2
    7 GRO(NODE,I,ITEMPX)=0.
    6 CONTINUE
       ICCMP(NODE)=0
       JDAY(NODE)=0
       PL(NODE)=0.
       YDLA(NODE)=0.
          NDPOOL(NODE)=NODE + 1
       PLL(NODE)=0.
       UL(NODE)=0.
       SPROUT(NODE)=0.
       DLA(NODE)=0.
       SW(NODE)=0.
       SUMES1(NODE)=0.
       SUMES2(NODE)=0.
   10 CONTINUE
          NDPOOL(20)=99
          IFIRST=0
```

```
          IAVAIL=1
      SGDD=0
      PMGDD=0
      IFREQ=5.
      NODE=0
       DO 15 I=1,366
       DLAI(I)=0.
   15 CONTINUE
      FMRGDA=0
      RETURN
      END
      SUBROUTINE EMRGNC (ICDAY)
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
      SMRGDA=0.
      DO 2999 I=ICDAY,366                                               CAR03730
      AVSLTP=TEMP(I)                                                    CAR03740
      IF(AVSLTP.GT.21.4) AVSLTP=21.4                                    CAR03750
      EMRGDA=1./(-1.05*AVSLTP+26.6)                                     CAR03760
      SMRGDA=SMRGDA+EMRGDA                                              CAR03770
      IF(SMRGDA-1.)2999,2997,2996                                       CAR03780
 2999 CONTINUE                                                          CAR03790
 2996 FMRGDA=SMRGDA-1.                                                  CAR03800
      SPROUT(NODE)=I-FMRGDA                                             CAR03810
      GO TO 2994                                                        CAR03820
 2997 SPROUT(NODE)=SMRGDA+ICDAY                                         CAR03830
 2994 RETURN                                                            CAR03840
      END                                                               CAR03850
      SUBROUTINE RPHOTO (I,TOTWT,TOFOTO)
```

```
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
CC
CC       THIS ROUTINE REDUCES POTENTIAL PHOTOSYNTHESIS TO
CC       ACTUAL PHOTOSYNTHESIS.  THE DIFFERENCE IS DUE TO
CC       THE LIMITING TEMP. AND WATER.  THIS ROUTINE ALSO
CC       SUBTRACTS THE RESPIRATION LOSS.
CC
      IF (TEMP(I).LT.5.0.OR.TEMP(I).GE.45.0) TEMPCO=0.              CAR01150
      IF (TEMP(I).GE.25.0.AND.TEMP(I).LE.40.0) TEMPCO=1.            CAR01160
      IF(TEMP(I).GE.5.0.AND.TEMP(I).LE.25.) TEMPCO=.05*TEMP(I)-.25  CAR01170
      IF (TEMP(I).GT.40.0.AND.TEMP(I).LT.45.) TEMPCO=-.2*TEMP(I)+9. CAR01180
      TOFOTO=TEMPCO*WATSCO*DAY PFO                                  CAR01190
      TEMPN=(TEMPMN(I)+TEMP(I))/2.                                  CAR01200
      RESPG=.14*(.65+.011*TEMPN)                                    CAR01210
      RESPM=.0054*(.044+.0019*TEMPN+.001*TEMPN*TEMPN)               CAR01220
      RESPN=RESPG*TOFOTO+RESPM*(TOTWT+WR(NODE))*1.43                CAR01230
      TOFOTO=TOFOTO-RESPN                                           CAR01240
      RETURN
      END
       SUBROUTINE LEAFAP (J,I,IMAX)
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
```

```
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/CCMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
      DIMENSION SXDIN(20)
      XDAYS(NODE,1)=0.0                                                 CAR00570
      XDAYS(NODE,2)=FMRGDA                                              CAR00580
 1104 DO 102 J=1,N                                                      CAR00590
      IF(J.NE.1) GO TO 1085                                             CAR00600
      RCOUNT(NODE,J)=SPROUT(NODE)                                       CAR00610
      GO TO 1084                                                        CAR00620
 1085 SXDIN(J)=0.0                                                      CAR00630
      PDAYS(NODE,J-1)=XDAYS(NODE,J)                                     CAR00640
      IF(XDAYS(NODE,2).GT.0.0) DAYS=SPROUT(NODE)                        CAR00650
      IF(XDAYS(NODE,J).GT.0.) GO TO 13                                  CAR00660
  103 I=I+1                                                             CAR00670
      DAYS=I                                                            CAR00680
      IF(DAYS.GT.IMAX) GO TO 104
   13 AVTEMP=TEMP(I)                                                    CAR00690
      IF(J.LT.6) GO TO 101                                              CAR00710
      IF(AVTEMP.LT.21.) GO TO 1C8                                       CAR00720
      IF(TEMP(I).GT.30.) AVTEMP=30.                                     CAR00730
      DIN=2.957-.0562*(AVTEMP-21)                                       CAR00740
      GO TO 18                                                          CAR00750
  108 DIN=2.9+.0567*(AVTEMP-22.)**2                                     CAR00760
      GO TO 18                                                          CAR00770
  101 IF(AVTEMP.GT.22) AVTEMP=22.
      DIN=2.8+.0292*((AVTEMP-22.)**2)
   18 IF(XDAYS(NODE,J).EQ.0.) GO TO 107                                 CAR00790
      XDIN=(1./DIN)*XDAYS(NODE,J)                                       CAR00800
      XDAYS(NODE,J)=0.0                                                 CAR00810
      GO TO 14                                                          CAR00820
  107 XDIN=1./DIN                                                       CAR00830
   14 SXDIN(J)=XDIN+SXDIN(J)                                            CAR00840
      IF (1.0-SXDIN(J)) 106,105,103                                     CAR00850
  105 RCOUNT(NODE,J)=DAYS                                               CAR00860
      XDAYS(NODE,J)=0.0                                                 CAR00870
      GO TO 1108                                                        CAR00880
  106 XDAYS(NODE,J)= (SXDIN(J)-1.0)*DIN                                 CAR00890
```

```
      RCOUNT(NODE,J)= DAYS-XDAYS(NODE,J)                                CAR00900
 1108 XDAYS(NODE,J+1)=XDAYS(NODE,J)                                     CAR00910
      IF (J.EQ.N) PDAYS(NODE,J)=XDAYS(NODE,J)                           CAR00920
 1084 WRITE(6,19) J,RCOUNT(NODE,J)                                      CAR00930
   19 FORMAT(5X, 'DAY LEAF',I3,' APPEARS = ',F7.2)                      CAR00940
  102 CONTINUE                                                          CAR00950
  104 CONTINUE
      RETURN
      END
      SUBROUTINE LEAF (I)                                               CAR01500
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
      DIMENSION SLA(20)                                                 CAR01570
      DO 108 J=1,N                                                      CAR01590
      IF(TEMP(I).GT.12.) GO TO 101                                      CAR01600
      GRO(NODE,2,J)=GRO(NODE,1,J)                                       CAR01610
      GO TO 108                                                         CAR01620
  101 CONTINUE                                                          CAR01630
      DAYS=I                                                            CAR01640
      IF(RCOUNT(NODE,J)-DAYS)110,111,109                                CAR01650
  110 CONTINUE                                                          CAR01660
      IF(GRO(NODE,1,J).GT.0.) GO TO 111                                 CAR01670
      GRO(NODE,2,J)=(5.1*(TEMP(I)-12.)*PDAYS(NODE,J))                   CAR01680
      GO TO 131                                                         CAR01690
  111 CONTINUE                                                          CAR01700
      IF(ACDAYS(NODE,J).GT.0.) GRO(NODE,2,J)=GRO(NODE,1,J)              CAR01710
      IF(ACDAYS(NODE,N).GT.0.) GRO(NODE,2,J)=GRO(NODE,2,J)*.996         CAR01720
      IF(ACDAYS(NODE,J).GT.0.) GO TO 108                                CAR01730
  122 GRO(NODE,2,J)=5.1*(TEMP(I)-12.)
```

```
      IF(GRO(NODE,1,J).GT.0.) GO TO 121                                 CAR01750
      GO TO 131                                                         CAR01760
  121 GRO(NODE,2,J)=(GRO(NODE,2,J)+GRO(NODE,1,J))                       CAR01770
  131 IF(GRO(NODE,2,J)-XMAX(NODE,J)) 108,208,1280                       CAR01780
 1280 IF(ACDAYS(NODE,J).GT.0.) GO TO 108                                CAR01790
      IF(DAYS-RCOUNT(NODE,J).GT.1.0)GO TO 1281                          CAR01800
      FGRO=XMAX(NODE,J)/GRO(NODE,2,J)                                   CAR01810
      ACDAYS(NODE,J)=FGRO+RCOUNT(NODE,J)                                CAR01820
      GO TO 209                                                         CAR01830
 1281 XXMAX=GRO(NODE,2,J)-XMAX(NCDE,J)                                  CAR01840
      DGRO=GRO(NODE,2,J)-GRO(NODE,1,J)                                  CAR01850
      FGRO=(1./DGRO)*XXMAX                                              CAR01860
      ACDAYS(NODE,J)=DAYS-FGRO                                          CAR01870
      GO TO 209                                                         CAR01880
  208 ACDAYS(NODE,J)=DAYS                                               CAR01890
  209 WRITE(6,210)J,ACDAYS(NODE,J),XMAX(NODE,J),NODE                    CAR01900
  210 FORMAT(' LEAF NO.=',1X,I2,5X,'DAY MAX LA ATTAINED=',F6.2,5X,      CAR01910
     1'MAX LA=',F6.2,3X,I2)                                             CAR01920
      GRO(NODE,2,J)=XMAX(NODE,J)                                        CAR01930
      IF(J.LE.11) GO TO 104                                             CAR01940
      GRO(NODE,2,J-11)=0.0                                              CAR01950
  104 IF(J-5) 108,107,106                                               CAR01960
  107 IDAY3(NODE)=(ACDAYS(NODE,5)+RCOUNT(NODE,N))/2.                    CAR01970
  106 IF(J-N) 108,105,108                                               CAR01980
  105 IDAY6(NODE)=ACDAYS(NODE,N)+((RCOUNT(NODE,N)-IDAY3(NODE))*.86)     CAR01990
      IDAY9(NODE)=(IDAY6(NODE)-RCOUNT(NODE,1))*1.6+RCOUNT(NODE,1)       CAR02000
  108 CONTINUE                                                          CAR02010
  109 CONTINUE
      DO 1112 J=1,N                                                     CAR02020
      IF(J.GT.1) GO TO 1100                                             CAR02030
      SLA(J)=GRO(NODE,2,J)                                              CAR02040
      GO TO 1112                                                        CAR02050
 1100 SLA(J)=SLA(J-1)+GRO(NODE,2,J)                                     CAR02060
 1112 CONTINUE                                                          CAR02070
 1500 DLA(NODE)=SLA(N)                                                  CAR02080
      DLAI(I)=DLA(NODE)/PAREA                                           CAR02090
      DO 54 J=1,20
      GRO(NODE,1,J)=GRO(NODE,2,J)
```

```
54 GRO(NODE,2,J)=0.
   RETURN                                                             CAR02100
   END                                                                CAR02110
   SUBROUTINE EVAP (I)                                                CAR03860
   COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
  1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
  2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
  3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
  4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
   COMMON/COMBLK/ROSPZ,PAREA,N,
  1TEMPCO,DAYPFO,LAT,WYE,BO,R2
   COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
   COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
   COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
   COMMON/COMBLK/SGDD,PMGDD, IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
   DATA GAMMA/.68/                                                    CAR03900
   TK=TEMP(I)+273.                                                    CAR03910
   DELTA = (EXP(21.255- 5304./TK))*(5304./(TK**2))                    CAR03920
   D=DELTA/GAMMA                                                      CAR03930
   ALBEDO=.3367-.1867*EXP(-.6*DLAI(I))                                CAR03940
   RO=520+193*SIN(.0172*(I-80))                                       CAR03950
   IF(SOLRAD(I).GT.RO)  SOLRAD(I)=RO                                  CAR03960
   R4=1.-.261*EXP(-7.77E-04*TEMP(I)**2)                               CAR03970
   R6=(R4-.96)*1.17E-07*TK**4*(.2+.8*(SOLRAD(I)/RO))                  CAR03980
   H=(1.-ALBEDO)*SOLRAD(I)+R6                                         CAR03990
   HO = H/583.                                                        CAR04000
   EO=1.28*DELTA/(DELTA+GAMMA)*HO                                     CAR04010
   IF (DLAI(I) .LT. 0.5) EOS = EO                                     CAR04020
   IF (DLAI(I) .LT. 0.5) GO TO 41                                     CAR04030
   HOS = HO * EXP (-0.398*DLAI(I))                                    CAR04040
   EOS = (D*HOS)/(D+1.)                                               CAR04050
41 RETURN                                                             CAR04060
   END                                                                CAR04070
   SUBROUTINE SOLWAT (I)                                              CAR04080
   COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
  1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
  2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
  3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
```

```
 4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
  COMMON/COMBLK/ROSPZ,PAREA,N,
 1TEMPCO,DAYPFO,LAT,WYE,BO,R2
  COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
  COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
  COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
  COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
  DATA U,CONA/.6,.35/                                               CAR04110
  PRECIP=RAIN(I)                                                    CAR04120
  WATSCO=1.                                                         CAR04130
  IF(SUMES1(NODE) - U) 1,2,2                                        CAR04140
1 IF (PRECIP - SUMES1(NODE)) 3, 4,4                                 CAR04150
3 SUMES1(NODE) = SUMES1(NODE)-PRECIP                                CAR04160
  GO TO 5                                                           CAR04170
4 SUMES1(NODE) = 0.                                                 CAR04180
5 SUMES1(NODE) = SUMES1(NODE) + EOS                                 CAR04190
  IF(SUMES1(NODE) - U) 6,6,7                                        CAR04200
6 ES = EOS                                                          CAR04210
  GO TO 24                                                          CAR04220
7 ES = EOS - 0.4*(SUMES1(NODE) - U)                                 CAR04230
  SUMES2(NODE) = 0.6*(SUMES1(NODE) - U)                             CAR04240
  T(NODE) = (SUMES2(NODE)/CONA)**2                                  CAR04250
  GO TO 24                                                          CAR04260
2 IF(PRECIP - SUMES2(NODE)) 9,8,8                                   CAR04270
8 PRECIP=PRECIP-SUMES2(NODE)                                        CAR04280
  SUMES1(NODE) = U - PRECIP                                         CAR04290
  T(NODE) = 0.                                                      CAR04300
  IF(PRECIP-U) 5,5,4                                                CAR04310
9 T(NODE)=T(NODE)+1.                                                CAR04320
  ES = CONA*T(NODE)**0.5 - SUMES2(NODE)                             CAR04330
  IF(PRECIP .GT. 0.) GO TO 10                                       CAR04340
  IF(ES .GT. EOS) ES = EOS                                          CAR04350
  GO TO 11                                                          CAR04360
10 ESX=0.8*PRECIP                                                   CAR04370
  IF(ESX .LE.ES) ESX = ES+PRECIP                                    CAR04380
  IF(ESX .GT. EOS) ESX = EOS                                        CAR04390
  ES = ESX                                                          CAR04400
11 SUMES2(NODE) = SUMES2(NODE) + ES - PRECIP                        CAR04410
```

```
      T(NODE) = (SUMES2(NODE)/CONA) **2                                 CAR04420
   24 IF(ES .LT. 0.) ES = 0.                                            CAR04430
      IF(DLAI(I) .GT. 3.0) GO TO 26                                     CAR04440
      EP=.53*DLAI(I)**.5*EO                                             CAR04450
      IF (EP .LT. 0.) EP = 0.                                           CAR04460
      GO TO 27                                                          CAR04470
   26 EP = EC - ES                                                      CAR04480
   27 IF(EP .LT. 0.) EP = 0.                                            CAR04490
   25 ET=ES+EP                                                          CAR04500
      IF(EO-ET) 39,41,41                                                CAR04510
   39 ET=EO                                                             CAR04520
      EP = ET - ES                                                      CAR04530
   41 CONTINUE                                                          CAR04540
      IF(SW(NODE).GT.UL(NODE)) SW(NODE)=UL(NODE)                        CAR04550
      ALLEO=PLL(NODE)                                                   CAR04560
      IF(DLAI(I) - 1.) 21,21,28                                         CAR04570
   28 IF(DLAI(I).LT.DLAI(I-1)) GO TO 23                                 CAR04580
      IF(DLAI(I)-2.7) 22,22,23                                          CAR04590
   21 ALLEO=.8*UL(NODE)-.1*UL(NODE)*DLAI(I)                             CAR04600
      GO TO 23                                                          CAR04610
   22 ALLEO=.59*PLL(NODE)*DLAI(I)-.44*UL(NODE)*DLAI(I)+1.19*            CAR04620
     1 UL(NODE)-.59*PLL(NODE)
   23 IF(ALLEO-SW(NODE)) 30,30,29                                       CAR04630
   29 WATSCO=SW(NODE)/ALLEO                                             CAR04640
      EP=EP*WATSCO                                                      CAR04650
   30 ET=ES+EP                                                          CAR04660
      SW(NODE)=SW(NODE)-ET+RAIN(I)                                      CAR04670
      IF(SW(NODE).LT.0.0) SW(NODE)=0.0                                  CAR04680
      IF(SW(NODE).GT.UL(NODE)) SW(NODE)=UL(NODE)                        CAR04690
   14 CONTINUE                                                          CAR04700
      RETURN                                                            CAR04710
      END                                                               CAR04720
      SUBROUTINE STAGE (I,N)                                            CAR04730
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
```

```
      IF(ISTAGE(NODE).EQ.5) GO TO 199                                   CAR04750
      IF(ACDAYS(NODE,5).EQ.0.) GO TO 100                                CAR04760
      IF(I.GT.IDAY3(NODE)) GO TO 101                                    CAR04770
  100 ISTAGE(NODE)=1                                                    CAR04780
      GO TO 199                                                         CAR04790
  101 IF(ACDAYS(NODE,N).GT.0.) GO TO 102                                CAR04800
      ISTAGE(NODE)=2                                                    CAR04810
      GO TO 199                                                         CAR04820
  102 IF(ACDAYS(NODE,N).NE.0.0.AND.I.GE.IDAY6(NODE)) GO TO 103          CAR04830
      ISTAGE(NODE)=3                                                    CAR04840
      GO TO 199                                                         CAR04850
  103 ISTAGE(NODE)=4                                                    CAR04860
      IF(I.EQ.IDAY9(NODE)) ISTAGE(NODE)=5                               CAR04870
  199 CONTINUE                                                          CAR04880
      RETURN                                                            CAR04890
      END                                                               CAR04900
      SUBROUTINE GROW (I)                                               CAR04910
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
      IF(I-RCOUNT(NODE,1).LT.1.)PL(NODE)=1.
      IF(I-RCOUNT(NODE,1).LT.1.)YDLA(NODE)=0.
      IF(DLA(NODE).LE.0.) GO TO 5                                       CAR04960
      IF(DRIWT.LE.0.0) GO TO 5                                          CAR04970
      DDLA=DLA(NODE)-YDLA(NODE)                                         CAR04980
      SLW=YDLA(NODE)/(WL(NODE)+.00001)                                  CAR05000
      IF(WL(NODE).LT..1) SLW=200.                                       CAR04990
      ITEMPX=ISTAGE(NODE)
      GO TO (1,2,3,4,4), ITEMPX                                         CAR05020
    1 IF(SLW.GT.200.) SLW=200.                                          CAR05030
```

```
      IF(SLW.LT.160.) SLW=160.                                          CAR05040
      PDWL=DDLA/SLW                                                     CAR05050
      IF(DDLA.EQ.0.) GO TO 100                                          CAR05060
      REQ=PDWL*1.25                                                     CAR05070
      F=DRIWT/REQ                                                       CAR05080
      IF(F.GT.1.25) F=1.25                                              CAR05090
      PDWL=PDWL*F                                                       CAR05100
  100 DWL=PDWL                                                          CAR05110
      DWR=DRIWT-DWL                                                     CAR05120
      GO TO 600                                                         CAR05130
    2 IF(I.EQ.IDAY3(NODE))IDAY(NODE)=0
      IDAY(NODE)=IDAY(NODE)+1                                           CAR05140
      IF(SLW.GT.160.) SLW=160.                                          CAR05150
      IF(SLW.LT.140.) SLW=140.                                          CAR05160
  204 PDWL=DDLA/SLW                                                     CAR05170
      IF(IDAY(NODE).LE.10) GO TO 200                                    CAR05180
      REQ=PDWL*1.5                                                      CAR05190
      F=DRIWT/REQ                                                       CAR05200
      IF(F.GT.2.) F=2.                                                  CAR05210
      PDWL=PDWL*F                                                       CAR05220
      DWL=PDWL                                                          CAR05230
      IF(ISTAGE(NODE).NE.3) GO TO 202                                   CAR05240
    3 PL(NODE)=PL(NODE)-.1                                              CAR05250
      IF(PL(NODE).LT.0.) PL(NODE)=0.                                    CAR05260
      DWL=WL(NODE)/(I-IDAY3(NODE))*PL(NODE)                             CAR05270
  202 REM=DRIWT-DWL                                                     CAR05280
      DWR=REM*.20                                                       CAR05290
      DWC=REM*.45                                                       CAR05300
      DWH=REM*.35                                                       CAR05310
      GO TO 602                                                         CAR05320
  200 REQ=PDWL*2.                                                       CAR05330
      F=DRIWT/REQ                                                       CAR05340
      IF(F.GT.2.) F=2.                                                  CAR05350
      PDWL=PDWL*F                                                       CAR05360
  205 DWL=PDWL                                                          CAR05370
      REM=DRIWT-DWL                                                     CAR05380
      IF(REM.GE.DRIWT*.20) GO TO 203                                    CAR05390
      DWR=REM*.4                                                        CAR05400
```

```
      GO TO 206                                                         CAR05410
  203 DWR=DRIWT*.20                                                     CAR05420
  206 DWC=DRIWT-DWL-DWR                                                 CAR05430
      GO TO 601                                                         CAR05440
    4 IF(I.EQ.IDAY6(NODE))JDAY(NODE)=0
      JDAY(NODE)=JDAY(NODE)+1                                           CAR05450
      ICOMP(NODE)=.1*(IDAY6(NODE)-RCOUNT(NODE,1))+.5                    CAR05460
      IF(JDAY(NODE).GT.ICOMP(NODE)) GO TO 402                           CAR05470
      DWH=DRIWT*.1                                                      CAR05480
      DWR=WC(NODE)*.005                                                 CAR05490
      DWC=DRIWT-DWH-DWR                                                 CAR05500
      DWL=0.                                                            CAR05510
      GO TO 602                                                         CAR05520
  402 RES=WC(NODE)*.025                                                 CAR05530
      IF(JDAY(NODE).LE.(IDAY9(NODE)-ACDAYS(NODE,N)+10)*.5) GO TO 400    CAR05540
      RES=-.5*RES                                                       CAR05550
      DWR=0.                                                            CAR05560
      GO TO 401                                                         CAR05570
  400 DWR=WC(NODE)*.005                                                 CAR05580
  401 DWG=DRIWT+RES-DWR                                                 CAR05590
      DWC=-RES                                                          CAR05600
      DWH=0.                                                            CAR05610
      DWL=0.                                                            CAR05620
      WG(NODE)=WG(NODE)+DWG                                             CAR05630
  602 WH(NODE)=WH(NODE)+DWH                                             CAR05640
  601 WC(NODE)=WC(NODE)+DWC                                             CAR05650
  600 WL(NODE)=WL(NODE)+DWL                                             CAR05660
      WR(NODE)=WR(NODE)+DWR                                             CAR05670
      YDLA(NODE)=DLA(NODE)                                              CAR05680
    5 CONTINUE                                                          CAR05690
      RETURN                                                            CAR05700
      END                                                               CAR05710
      SUBROUTINE PHOTO (I)                                              CAR02120
      COMMON/NODBLK/NODE,IDAY(20),PL(20),YDLA(20),JDAY(20),ICOMP(20),IP
     1RINT(20),SUMES1(20),SUMES2(20),T(20),PDAYS(20,20),PERCNT(21),XDAYS
     2(20,21),GRO(20,2,20),XMAX(20,20),SPROUT(20),SW(20),UL(20),PLL(20)
     3,RCOUNT(20,20),ACDAYS(20,20),IDAY3(20),IDAY6(20),IDAY9(20)
     4,ISTAGE(20),DLA(20),WR(21),WL(21),WC(21),WH(21),WG(21)
```

```
      COMMON/COMBLK/ROSPZ,PAREA,N,
     1TEMPCO,DAYPFO,LAT,WYE,BO,R2
      COMMON/COMBLK/TEMPMX(366),TEMPMN(366),TEMP(366),FMRGDA
      COMMON/COMBLK/DLAI(366),SOLRAD(366),RAIN(366),WATSCO,EOS,EO
      COMMON/COMBLK/F,DRIWT,DWG,DWH,DWC
      COMMON/COMBLK/SGDD,PMGDD,IAVAIL,IFIRST,ILAST,NDPOOL(20),MATURE(20)
      INTEGER HOURAN                                                    CAR02190
      REAL LAT,L,KAPPA,LL,LLPP,K,LITRAN,INTPAR,LLAC,LLIN                CAR02200
      IHR=0                                                             CAR02210
      DAYPFO=0.                                                         CAR02220
      RHO=90.                                                           CAR02230
      DECDAY=I+100                                                      CAR02240
      DECLIN=SIN((DECDAY*360./365.)/180.*3.141593)*(-23.47)             CAR02250
      SUNRIS=12.-ARCOS(-TAN(LAT/180.*3.141593)*TAN(DECLIN/180.*         CAR02260
     *3.141593))/0.2618                                                 CAR02270
  100 LST=IHR+IFIX(SUNRIS)+1                                            CAR02280
      HOURAN=15*IABS(12-LST)                                            CAR02290
      TERM=SIN(LAT/180.*3.141593)*SIN(DECLIN/180.*3.141593)             CAR02300
      SINBET=COS(LAT/180.*3.141593)*COS(HOURAN/180.*3.141593)*          CAR02310
     *COS(DECLIN/180.*3.141593)+TERM                                    CAR02320
      BETA=ARSIN(SINBET)/3.141593*180.                                  CAR02330
      TERM1=COS(LAT/180.*3.141593)*SIN(DECLIN/180.*3.141593)            CAR02340
      TERM2=COS(DECLIN/180.*3.141593)*SIN(LAT/180.*3.141593)*           CAR02350
     *COS(HOURAN/180.*3.141593)                                         CAR02360
      COSGAM=1./COS(BETA/180.*3.141593)*(TERM1-TERM2)                   CAR02370
      COSTST=ABS(COSGAM)                                                CAR02380
      IF(COSGAM.LT.-1.) GO TO 101                                       CAR02390
      IF(COSTST.LE.1.) GO TO 102                                        CAR02400
      COSGAM=1.                                                         CAR02410
      GO TO 102                                                         CAR02420
  101 COSGAM=-1.                                                        CAR02430
  102 GAMMA=ARCOS(COSGAM)/3.141593*180.                                 CAR02440
      IF(LST.LE.12) GO TO 103                                           CAR02450
      GAMMA=360.-GAMMA                                                  CAR02460
  103 PHI=GAMMA-RHO                                                     CAR02470
      THETA=BETA                                                        CAR02480
      IF(PHI.LE.-90.) PHI=PHI+180.                                      CAR02490
      IF(PHI.LE.0..AND.PHI.GT.-90.) PHI=ABS(PHI)                        CAR02500
```

```
      IF(PHI.LE.90..AND.PHI.GT.0.) PHI=PHI
      IF(PHI.LE.180..AND.PHI.GT.90.) PHI=180.-PHI
      IF(PHI.LE.270..AND.PHI.GT.180.) PHI=PHI-180.
      IF(PHI.GT.270.) PHI=360.-PHI
      IF(DLA(NODE).LT.700.) GO TO 125
       DLATMP=DLA(NODE)
      L=ALOG10(DLATMP)*62.5-136.25
      GO TO 126
  125 DLATMP=DLA(NODE)
      L=ALOG10(DLATMP)*23.-18.
      IF(L.LE.0.) L=.5
  126 SAC=0.
      AS=0.
      AR=0.
      IF(THETA.EQ.0.) GO TO 116
      IF(THETA.EQ.90.) GO TO 116
      IF(PHI.EQ.0.) GO TO 111
      IF(L.GE.R2) GO TO 111
      X=L/TAN(THETA/180.*3.141593)*SIN(PHI/180.*3.141593)
      Y=R2-L
      IF(X.LE.Y) GO TO 110
      X=(R2-L)/L
      Y=TAN(PHI/180.*3.141593)
      IF(X.GE.Y) GO TO 110
      AS=((R2-L)**2)*TAN(THETA/180.*3.141593)/TAN(PHI/180.*3.141593 )
      GO TO 111
  110 AS=L*L
  111 CONTINUE
      IF(PHI.EQ.90.) GO TO 113
      IF(L.GE.ROSPZ) GO TO 113
      X=L/TAN(THETA/180.*3.141593)*COS(PHI/180.*3.141593)
      Y=ROSPZ-L
      IF(X.LE.Y) GO TO 112
      X=L/(ROSPZ-L)
      Y=TAN(PHI/180.*3.141593)
      IF(X.LE.Y) GO TO 112
      AR=((ROSPZ-L)**2)*TAN(THETA/180.*3.141593)*TAN(PHI/180.*3.141593)
      GO TO 113
```

```
112 AR=L*L                                                                     CAR02870
113 CONTINUE                                                                   CAR02880
    X=TAN(THETA/180.*3.141593)                                                 CAR02890
    IF(L.GE.R2) GO TO 114                                                      CAR02900
    YY=SQRT(((ROSPZ-L)**2)+((R2-L)**2))                                        CAR02910
    Y=L/YY                                                                     CAR02920
    IF(X.GE.Y) GO TO 116                                                       CAR02930
    IF(PHI.EQ.0.) GO TO 114                                                    CAR02940
    X=TAN(PHI/180.*3.141593)                                                   CAR02950
    Y=(R2-L)/(ROSPZ-L)                                                         CAR02960
    IF(X.GE.Y) GO TO 114                                                       CAR02970
    SAC=((R2-L)/TAN(PHI/180.*3.141593)-(ROSPZ-L))*(L/TAN(THETA/                CAR02980
   *180.*3.141593)-YY)*TAN(THETA/180.*3.141593)                                CAR02990
    GO TO 116                                                                  CAR03000
114 IF(PHI.EQ.90.) GO TO 116                                                   CAR03010
    IF(L.GE.ROSPZ) GO TO 116                                                   CAR03020
    IF(L.LE.R2) GO TO 115                                                      CAR03030
    AR=L*TAN(THETA/180.*3.141593)*(ROSPZ-L)*COS(PHI/180.*3.141593)             CAR03040
    X=ROSPZ-L                                                                  CAR03050
    Y=(L/TAN(THETA/180.*3.141593))*COS(PHI/180.*3.141593)                      CAR03060
    IF(X.LE.Y) GO TO 116                                                       CAR03070
    AR=L*L                                                                     CAR03080
    GO TO 116                                                                  CAR03090
115 DOWN=((ROSPZ-L)*TAN(PHI/180.*3.141593)-(R2-L))                             CAR03100
    IF(DOWN.GT.L)DOWN=L                                                        CAR03110
    SAC=DOWN*(L/TAN(THETA/180.*3.141593)-YY)*TAN(THETA/180.*3.141593)          CAR03120
116 IF(L.GE.ROSPZ) GO TO 127                                                   CAR03130
    DLATMP=DLA(NODE)
    ETA=SIN(THETA/180.*3.141593)*(-16.67*ALOG10(DLATMP)+112.08)*.01            CAR03140
    GO TO 128                                                                  CAR03150
127 DLATMP=DLA(NODE)
    ETA=(-16.67*ALOG10(DLATMP)+112.08)*.01                                     CAR03160
128 KAPPA=.5*COS(THETA/180.*3.141593)                                          CAR03170
    PA=(((AS+AR-SAC)/(L**2))*KAPPA+ETA)*DLA(NODE)                              CAR03180
117 LL=SQRT(PA)                                                                CAR03190
    IF(LL.LE.R2) GO TO 121                                                     CAR03200
    NP=2*IFIX(LL/2./R2)                                                        CAR03210
    NPR=2*IFIX(LL/2./ROSPZ)                                                    CAR03220
```

```
      IF(NPR.GT.0) GO TO 122                                            CAR03230
      IF(LL.GT.ROSPZ) GO TO 119                                         CAR03240
      IF(NP.EQ.0) GO TO 118                                             CAR03250
      WBD=LL-R2*FLOAT(NP)                                               CAR03260
      A=(NP+1)*(WBD*LL)*(1./FLOAT(NP+1))+NP*(R2-WBD)*LL/FLOAT(NP)       CAR03270
      GO TO 123                                                         CAR03280
  118 A=LL*LL-LL*(LL-R2)                                                CAR03290
      GO TO 123                                                         CAR03300
  119 IF(NP.EQ.0) GO TO 120                                             CAR03310
      WBD=LL-R2*FLOAT(NP)                                               CAR03320
      LLPP=2.*(LL-ROSPZ)                                                CAR03330
      XNP=NP                                                            CAR03340
      TERM=XNP*(R2-WBD)*(LL-LLPP)/XNP+XNP*(R2-WBD)*LLPP/(1.+2.*XNP)     CAR03350
      A=(XNP+1.)*(WBD*(LL-LLPP))/(XNP+1.)+(XNP+1.)*(WBD*LLPP)/          CAR03360
     *(3.+2.*XNP)+TERM                                                  CAR03370
      GO TO 123                                                         CAR03380
  120 A=PA-((2.*(LL-R2))*(LL-2.*(LL-ROSPZ))+(LL-R2)*(LL-2.*(LL-ROSPZ))+ CAR03390
     *(LL-R2)*(LL-ROSPZ))                                               CAR03400
      GO TO 123                                                         CAR03410
  121 A=PA                                                              CAR03420
      GO TO 123                                                         CAR03430
  122 WBD=LL-R2*XNP                                                     CAR03440
      WBL=R2-WBD                                                        CAR03450
      XNPR=NPR                                                          CAR03460
      WBDR=LL-XNPR*ROSPZ                                                CAR03470
      WBLR=ROSPZ-WBDR                                                   CAR03480
      TERM1=(XNP+1.)*(XNPR+1.)*WBDR*WBD/(XNP+XNPR+2.)+                  CAR03490
     *XNP*XNPR*WBLR*WBL/(XNP+XNPR)                                      CAR03500
      A=XNPR*(XNP+1.)*WBLR*WBD/(XNP+XNPR+1.)+XNP*(XNPR+1.)*WBDR*WBL/    CAR03510
     *(XNP+XNPR+1.)+TERM1                                               CAR03520
  123 ARATIO=A/PA                                                       CAR03530
      K=.11+.680*ARATIO                                                 CAR03540
  124 LITRAN=EXP(-K*DLAI(I))*100.                                       CAR03550
      EFFLAI=A/PAREA                                                    CAR03560
      DAYLN=120.*(12.-SUNRIS)                                           CAR03570
      SUNTIM=FLOAT(IFIX(SUNRIS)+1)-SUNRIS                               CAR03580
      TIME=60.*(SUNTIM+FLOAT(IHR))                                      CAR03590
      SOLARH=1.5708*SOLRAD(I)/DAYLN*SIN(TIME*180./DAYLN/180.*3.141593)  CAR03600
```

```
      INTPAR=.121*SOLARH*(1.-LITRAN/100.)*60.                           CAR03610
      POFO=31.35*INTPAR**.5-13.32                                       CAR03620
      IF(POFO.LT.1.) POFO=INTPAR*4.8                                    CAR03630
      DAYPFO=DAYPFO+POFO                                                CAR03640
      IHR=IHR+1                                                         CAR03650
      LIMIT=2*(11-IFIX(SUNRIS))                                         CAR03660
      IF(IHR.LE.LIMIT) GO TO 100                                        CAR03670
      RETURN                                                            CAR03680
      END                                                               CAR03690
```

Field Model Sample Output

```
MANHATTAN JUNE 1 1966 DATA RS610

 PLANTING DATE= 152       CALENDAR DAY=     6     1  1966
 DAY LEAF  1 APPEARS =  155.87
 DAY LEAF  2 APPEARS =  157.68
 DAY LEAF  3 APPEARS =  160.56
 DAY LEAF  4 APPEARS =  163.77
 DAY LEAF  5 APPEARS =  166.71
 DAY LEAF  6 APPEARS =  170.10
 DAY LEAF  7 APPEARS =  172.86
 DAY LEAF  8 APPEARS =  175.47
 DAY LEAF  9 APPEARS =  178.07
 DAY LEAF 10 APPEARS =  180.58
 DAY LEAF 11 APPEARS =  183.15
 DAY LEAF 12 APPEARS =  185.67
 DAY LEAF 13 APPEARS =  188.22
 DAY LEAF 14 APPEARS =  190.67
 DAY LEAF 15 APPEARS =  193.12
 DAY LEAF 16 APPEARS =  195.57
 DAY LEAF  1 APPEARS =  158.87
 DAY LEAF  2 APPEARS =  161.15
 DAY LEAF  3 APPEARS =  163.95
 DAY LEAF  4 APPEARS =  166.91
 DAY LEAF  5 APPEARS =  170.01
 DAY LEAF  6 APPEARS =  172.76
 DAY LEAF  7 APPEARS =  175.39
 DAY LEAF  8 APPEARS =  177.95
 DAY LEAF  9 APPEARS =  180.49
 DAY LEAF 10 APPEARS =  183.06
 DAY LEAF 11 APPEARS =  185.59
 DAY LEAF 12 APPEARS =  188.14
 DAY LEAF 13 APPEARS =  190.59
 DAY LEAF 14 APPEARS =  193.04
 DAY LEAF 15 APPEARS =  195.49
 DAY LEAF 16 APPEARS =  198.04
 DAY LEAF  1 APPEARS =  153.87
 DAY LEAF  2 APPEARS =  155.67
 DAY LEAF  3 APPEARS =  158.50
 DAY LEAF  4 APPEARS =  161.77
 DAY LEAF  5 APPEARS =  164.60
 DAY LEAF  6 APPEARS =  168.06
 DAY LEAF  7 APPEARS =  171.00
 DAY LEAF  8 APPEARS =  173.78
 DAY LEAF  9 APPEARS =  176.33
 DAY LEAF 10 APPEARS =  178.92
 DAY LEAF 11 APPEARS =  181.45
 DAY LEAF 12 APPEARS =  184.01
 DAY LEAF 13 APPEARS =  186.51
 DAY LEAF 14 APPEARS =  189.05
 DAY LEAF 15 APPEARS =  191.50
 DAY LEAF 16 APPEARS =  193.95


PLANTS PER HECTARE=    125000.
ROW SPACING=  76.20
PLANT SPACING=    10.5
AREA PER PLANT=   800.00
```

JULIAN DAY	STAGE	ROOT WT	LEAF WT	CULM WT	HEAD WT	GRAIN WT	TOTAL WT	LA,CM	LAI	TCO	WCO	SW	
LEAF NO.= 1	DAY MAX	LA ATTAINED=154.12		MAX LA=	1.00	3							
155(1)	1	0.000	0.001	0.000	0.000	0.000	0.001	1.000	0.001	1.0	1.0	15.2	3
LEAF NO.= 2	DAY MAX	LA ATTAINED=155.76		MAX LA=	2.00	3							
156(2)	1	0.001	0.003	0.000	0.000	0.000	0.003	3.000	0.004	1.0	1.0	15.2	3
LEAF NO.= 1	DAY MAX	LA ATTAINED=156.16		MAX LA=	1.00	1							
157(1)	1	0.000	0.001	0.000	0.000	0.000	0.001	1.000	0.001	0.8	1.0	15.2	1
157(3)	1	0.005	0.003	0.000	0.000	0.000	0.003	3.000	0.004	0.8	1.0	15.1	3
LEAF NO.= 2	DAY MAX	LA ATTAINED=157.82		MAX LA=	2.00	1							
158(2)	1	0.001	0.003	0.000	0.000	0.000	0.003	3.000	0.004	0.8	1.0	15.2	1
158(4)	1	0.008	0.003	0.000	0.000	0.000	0.003	3.000	0.004	0.8	1.0	15.1	3
159(3)	1	0.001	0.003	0.000	0.000	0.000	0.003	3.000	0.004	0.8	1.0	16.0	1
LEAF NO.= 3	DAY MAX	LA ATTAINED=158.68		MAX LA=	4.00	3							
159(5)	1	0.009	0.004	0.000	0.000	0.000	0.004	7.000	0.009	0.8	1.0	15.9	3
160(4)	1	0.004	0.003	0.000	0.000	0.000	0.003	3.000	0.004	0.6	1.0	16.1	1
LEAF NO.= 1	DAY MAX	LA ATTAINED=159.34		MAX LA=	1.00	2							
160(1)	1	0.000	0.001	0.000	0.000	0.000	0.001	1.000	0.001	0.6	1.0	15.3	2
160(6)	1	0.015	0.004	0.000	0.000	0.000	0.004	7.000	0.009	0.6	1.0	16.0	3
161(5)	1	0.005	0.007	0.000	0.000	0.000	0.007	4.068	0.005	0.6	1.0	15.9	1
161(2)	1	0.001	0.001	0.000	0.000	0.000	0.001	1.000	0.001	0.6	1.0	15.3	2
161(7)	1	0.022	0.004	0.000	0.000	0.000	0.004	7.000	0.009	0.6	1.0	15.8	3
LEAF NO.= 3	DAY MAX	LA ATTAINED=161.04		MAX LA=	4.00	1							
162(6)	1	0.007	0.015	0.000	0.000	0.000	0.015	7.000	0.009	1.0	1.0	15.8	1
LEAF NO.= 2	DAY MAX	LA ATTAINED=161.18		MAX LA=	2.00	2							
162(3)	1	0.002	0.004	0.000	0.000	0.000	0.004	3.000	0.004	1.0	1.0	15.2	2
LEAF NO.= 4	DAY MAX	LA ATTAINED=162.20		MAX LA=	7.00	3							
162(8)	1	0.026	0.020	0.000	0.000	0.000	0.020	14.000	0.017	1.0	1.0	15.7	3
163(7)	1	0.016	0.015	0.000	0.000	0.000	0.015	7.000	0.009	1.0	1.0	15.8	1
163(4)	1	0.006	0.004	0.000	0.000	0.000	0.004	3.000	0.004	1.0	1.0	15.2	2
163(9)	1	0.043	0.020	0.000	0.000	0.000	0.020	14.000	0.017	1.0	1.0	15.7	3
LEAF NO.= 4	DAY MAX	LA ATTAINED=164.30		MAX LA=	7.00	1							
164(8)	1	0.019	0.030	0.000	0.000	0.000	0.030	14.000	0.017	0.9	1.0	15.9	1
164(5)	1	0.007	0.010	0.000	0.000	0.000	0.010	5.790	0.007	0.9	1.0	15.3	2
164(10)	1	0.062	0.020	0.000	0.000	0.000	0.020	14.000	0.017	0.9	1.0	15.7	3
165(9)	1	0.031	0.030	0.000	0.000	0.000	0.030	14.000	0.017	0.7	1.0	15.7	1
LEAF NO.= 3	DAY MAX	LA ATTAINED=164.03		MAX LA=	4.00	2							
165(6)	1	0.008	0.015	0.000	0.000	0.000	0.015	7.000	0.009	0.7	1.0	15.2	2
LEAF NO.= 5	DAY MAX	LA ATTAINED=165.35		MAX LA=	12.00	3							
165(11)	1	0.066	0.037	0.000	0.000	0.000	0.037	26.000	0.032	0.7	1.0	15.6	3
166(10)	1	0.040	0.030	0.000	0.000	0.000	0.030	14.000	0.017	0.8	1.0	16.3	1
166(7)	1	0.013	0.015	0.000	0.000	0.000	0.015	7.000	0.009	0.8	1.0	15.7	2
166(12)	1	0.084	0.037	0.000	0.000	0.000	0.037	26.000	0.032	0.8	1.0	16.1	3
167(11)	1	0.045	0.049	0.000	0.000	0.000	0.049	23.860	0.030	0.7	1.0	16.1	1
167(8)	1	0.015	0.023	0.000	0.000	0.000	0.023	9.915	0.012	0.7	1.0	15.7	2
167(13)	1	0.111	0.037	0.000	0.000	0.000	0.037	26.000	0.032	0.7	1.0	16.1	3
LEAF NO.= 5	DAY MAX	LA ATTAINED=167.08		MAX LA=	12.00	1							
168(12)	1	0.048	0.059	0.000	0.000	0.000	0.059	26.000	0.032	0.6	1.0	15.9	1
LEAF NO.= 4	DAY MAX	LA ATTAINED=167.15		MAX LA=	7.00	2							
168(9)	1	0.017	0.029	0.000	0.000	0.000	0.029	14.000	0.017	0.6	1.0	15.6	2
168(14)	1	0.123	0.037	0.000	0.000	0.000	0.037	26.000	0.032	0.6	1.0	15.9	3
169(13)	1	0.079	0.059	0.000	0.000	0.000	0.059	26.000	0.032	0.7	1.0	15.8	1
169(10)	1	0.034	0.029	0.000	0.000	0.000	0.029	14.000	0.017	0.7	1.0	15.5	2
LEAF NO.= 6	DAY MAX	LA ATTAINED=168.91		MAX LA=	28.00	3							
169(15)	1	0.159	0.180	0.000	0.000	0.000	0.180	54.000	0.067	0.7	1.0	15.7	3
170(14)	1	0.121	0.059	0.000	0.000	0.000	0.059	26.000	0.032	0.9	1.0	15.6	1
170(11)	1	0.055	0.029	0.000	0.000	0.000	0.029	14.000	0.017	0.9	1.0	15.4	2
170(16)	1	0.393	0.180	0.000	0.000	0.000	0.180	54.000	0.067	0.9	1.0	15.5	3

LEAF NO.= 6	DAY MAX LA ATTAINED=170.53			MAX LA= 28.00		1							
171(15)	1	0.201	0.234	0.000	0.000	0.000	0.234	54.000	0.067	1.0	1.0	15.4	1
LEAF NO.= 5	DAY MAX LA ATTAINED=170.18			MAX LA= 12.00		2							
171(12)	1	0.065	0.064	0.000	0.000	0.000	0.064	26.000	0.032	1.0	1.0	15.3	2
171(17)	1	0.649	0.180	0.000	0.000	0.000	0.180	54.000	0.067	1.0	1.0	15.3	3
172(16)	1	0.431	0.234	0.000	0.000	0.000	0.234	54.000	0.067	1.0	1.0	15.2	1
172(13)	1	0.106	0.064	0.000	0.000	0.000	0.064	26.000	0.032	1.0	1.0	15.1	2
172(18)	1	0.902	0.566	0.000	0.000	0.000	0.566	115.743	0.145	1.0	1.0	15.1	3
173(17)	1	0.704	0.286	0.000	0.000	0.000	0.286	62.280	0.078	0.9	1.0	15.1	1
173(14)	1	0.166	0.150	0.000	0.000	0.000	0.150	39.652	0.050	0.9	1.0	15.0	2
LEAF NO.= 7	DAY MAX LA ATTAINED=172.39			MAX LA= 84.00		3							
173(19)	1	1.540	0.705	0.000	0.000	0.000	0.705	138.000	0.172	0.9	1.0	14.9	3
174(18)	1	1.020	0.718	0.000	0.000	0.000	0.718	131.413	0.164	1.0	1.0	14.8	1
LEAF NO.= 6	DAY MAX LA ATTAINED=173.21			MAX LA= 28.00		2							
174(15)	1	0.319	0.239	0.000	0.000	0.000	0.239	54.000	0.067	1.0	1.0	14.9	2
174(20)	1	2.270	0.803	0.000	0.000	0.000	0.803	153.343	0.192	1.0	1.0	14.6	3
LEAF NO.= 7	DAY MAX LA ATTAINED=174.08			MAX LA= 84.00		1							
175(19)	1	1.730	0.763	0.000	0.000	0.000	0.763	138.000	0.172	1.0	1.0	14.6	1
175(16)	1	0.553	0.239	0.000	0.000	0.000	0.239	54.000	0.067	1.0	1.0	14.7	2
175(21)	1	2.791	1.330	0.000	0.000	0.000	1.330	233.809	0.292	1.0	1.0	14.4	3
176(20)	1	2.363	1.037	0.000	0.000	0.000	1.037	184.809	0.231	1.0	1.0	14.4	1
176(17)	1	0.858	0.577	0.000	0.000	0.000	0.577	108.072	0.135	1.0	1.0	14.5	2
LEAF NO.= 8	DAY MAX LA ATTAINED=175.54			MAX LA=144.00		3							
176(22)	1	3.656	1.673	0.000	0.000	0.000	1.673	282.000	0.352	1.0	1.0	14.1	3
177(21)	1	2.602	1.637	0.000	0.000	0.000	1.637	259.609	0.325	1.0	1.0	14.2	1
LEAF NO.= 7	DAY MAX LA ATTAINED=176.40			MAX LA= 84.00		2							
177(18)	1	1.154	0.777	0.000	0.000	0.000	0.777	138.000	0.172	1.0	1.0	14.4	2
177(23)	1	4.202	2.042	0.000	0.000	0.000	2.042	331.771	0.415	1.0	1.0	13.9	3
LEAF NO.= 8	DAY MAX LA ATTAINED=177.29			MAX LA=144.00		1							
178(22)	1	2.933	1.811	0.000	0.000	0.000	1.811	282.000	0.352	1.0	1.0	14.7	1
178(19)	1	1.409	0.784	0.000	0.000	0.000	0.784	139.017	0.174	1.0	1.0	14.9	2
178(24)	1	4.335	2.575	0.000	0.000	0.000	2.575	407.988	0.510	1.0	1.0	14.4	3
179(23)	1	3.593	2.448	0.000	0.000	0.000	2.448	363.470	0.454	1.0	1.0	14.5	1
179(20)	1	1.800	1.402	0.000	0.000	0.000	1.402	226.567	0.283	1.0	1.0	14.7	2
179(25)	1	5.114	3.317	0.000	0.000	0.000	3.317	502.918	0.629	1.0	1.0	14.0	3
180(24)	1	4.419	3.154	0.000	0.000	0.000	3.154	453.853	0.567	1.0	1.0	14.1	1
LEAF NO.= 8	DAY MAX LA ATTAINED=179.61			MAX LA=144.00		2							
180(21)	1	2.580	1.831	0.000	0.000	0.000	1.831	282.000	0.352	1.0	1.0	14.4	2
LEAF NO.= 9	DAY MAX LA ATTAINED=179.12			MAX LA=224.00		3							
180(26)	2	5.463	4.190	0.524	0.000	0.000	4.715	603.764	0.755	1.0	1.0	13.6	3
LEAF NO.= 9	DAY MAX LA ATTAINED=180.66			MAX LA=224.00		1							
181(25)	1	5.417	3.824	0.000	0.000	0.000	3.824	539.546	0.674	1.0	1.0	13.7	1
181(22)	1	3.564	2.145	0.000	0.000	0.000	2.145	322.284	0.403	1.0	1.0	14.1	2
181(27)	2	5.833	5.116	1.079	0.000	0.000	6.195	682.814	0.854	1.0	1.0	13.2	3
182(26)	2	5.762	4.686	0.517	0.000	0.000	5.203	618.596	0.773	1.0	1.0	13.3	1
182(23)	1	4.354	2.763	0.000	0.000	0.000	2.763	401.334	0.502	1.0	1.0	13.8	2
182(28)	2	6.217	6.074	1.654	0.000	0.000	7.729	805.057	1.006	1.0	1.0	12.8	3
183(27)	2	6.123	5.590	1.060	0.000	0.000	6.649	701.896	0.877	1.0	1.0	12.9	1
183(24)	1	5.255	3.414	0.000	0.000	0.000	3.414	464.634	0.606	1.0	1.0	13.5	2
LEAF NO.= 10	DAY MAX LA ATTAINED=182.48			MAX LA=296.00		3							
183(29)	2	6.727	7.101	2.271	0.000	0.000	9.372	928.492	1.161	1.0	1.0	12.3	3
184(28)	2	6.530	6.606	1.670	0.000	0.000	8.276	853.615	1.067	1.0	1.0	12.5	1
LEAF NO.= 9	DAY MAX LA ATTAINED=183.26			MAX LA=224.00		2							
184(25)	1	6.226	4.180	0.000	0.000	0.000	4.180	582.730	0.728	1.0	1.0	13.1	2
184(30)	2	7.065	8.196	2.928	0.000	0.000	11.124	1010.375	1.263	1.0	1.0	11.9	3
LEAF NO.= 10	DAY MAX LA ATTAINED=184.22			MAX LA=296.00		1							
185(29)	2	6.956	7.673	2.310	0.000	0.000	9.982	955.135	1.194	1.0	1.0	12.1	1
185(26)	2	6.594	5.099	0.551	0.000	0.000	5.651	666.030	0.833	1.0	1.0	12.7	2
185(31)	2	7.531	9.360	3.626	0.000	0.000	12.985	1175.772	1.470	1.0	1.0	11.4	3
185(30)	2	7.407	8.799	2.986	0.000	0.000	11.785	1092.143	1.365	1.0	1.0	11.6	1

```
          186( 27)     2      6.990      6.089      1.145      0.300     0.000      7.234   811.352     1.014  1.0  1.0  12.2   2
LEAF NO.= 11          DAY MAX LA ATTAINED=185.66         MAX LA=360.00     3
          186( 32)     2      8.020     10.584      4.360      0.000     0.000     14.944  1347.230     1.684  1.0  1.0  10.8   3
          187( 31)     2      7.889     10.005      3.709      0.000     0.000     13.715  1272.910     1.591  1.0  1.0  11.0   1
LEAF NO.= 10          DAY MAX LA ATTAINED=186.36         MAX LA=296.00     2
          187( 28)     2      7.414      7.149      1.781      0.000     0.000      8.930   934.572     1.168  1.0  1.0  11.8   2
          187( 33)     2      8.533     11.866      5.129      0.000     0.000     16.995  1482.342     1.853  1.0  1.0  10.3   3
LEAF NO.= 11          DAY MAX LA ATTAINED=187.19         MAX LA=360.00     1
          188( 32)     2      8.386     11.204      4.497      0.000     0.000     15.701  1356.808     1.696  1.0  1.0  11.5   1
          188( 29)     2      7.854      8.157      2.535      0.000     0.000     10.692  1005.122     1.256  1.0  1.0  12.2   2
          188( 34)     2      9.066     13.197      5.928      0.000     0.000     19.125  162[illegible].442     2.029  1.0  1.0  10.7   3
          189( 33)     2      8.888     12.459      5.250      0.000     0.000     17.708  1520.554     1.901  1.0  1.0  10.8   1
          189( 30)     2      8.307      9.270      3.215      0.000     0.000     12.504  1176.268     1.470  1.0  1.0  11.7   2
LEAF NO.= 12          DAY MAX LA ATTAINED=188.63         MAX LA=404.00     3
          189( 35)     2      9.595     14.521      6.722      0.000     0.000     21.243  1772.079     2.215  1.0  1.0  10.1   3
          190( 34)     2      9.402     13.744      6.021      0.000     0.000     19.765  1729.654     2.162  1.0  1.0  10.2   1
LEAF NO.= 11          DAY MAX LA ATTAINED=189.62         MAX LA=360.00     2
          190( 31)     2      8.774     10.458      3.915      0.000     0.000     14.373  1345.896     1.682  1.0  1.0  11.1   2
          190( 36)     2      9.775     16.317      7.127      0.314     0.000     23.758  1989.918     2.487  1.0  1.0   9.5   3
LEAF NO.= 12          DAY MAX LA ATTAINED=190.12         MAX LA=404.00     1
          191( 35)     2      9.945     15.101      6.835      0.000     0.000     21.936  1878.924     2.349  1.0  1.0   9.6   1
          191( 32)     2      9.275     11.710      4.667      0.000     0.000     16.377  1491.637     1.865  1.0  1.0  10.5   2
          191( 37)     2      9.965     18.218      7.554      0.647     0.000     26.419  2196.185     2.745  1.0  1.0   8.8   3
          192( 36)     2     10.130     16.949      7.251      0.323     0.000     24.523  2098.622     2.623  1.0  1.0   9.0   1
          192( 33)     2      9.792     13.002      5.442      0.000     0.000     18.445  1697.904     2.122  1.0  1.0  10.0   2
LEAF NO.= 13          DAY MAX LA ATTAINED=191.05         MAX LA=420.00     3
          192( 38)     2     10.156     20.131      7.985      0.982     0.000     29.098  2354.344     2.943  1.0  1.0   8.2   3
LEAF NO.= 13          DAY MAX LA ATTAINED=192.33         MAX LA=420.00     1
          193( 37)     2     10.320     18.856      7.680      0.657     0.000     27.194  2248.326     2.810  1.0  1.0   8.3   1
LEAF NO.= 12          DAY MAX LA ATTAINED=192.12         MAX LA=404.00     2
          193( 34)     2     10.328     14.342      6.246      0.000     0.000     20.587  1825.208     2.282  1.0  1.0   9.3   2
LEAF NO.= 14          DAY MAX LA ATTAINED=192.82         MAX LA=400.00     3
          193( 39)     2     10.443     22.003      8.406      1.309     0.000     31.718  2596.673     3.246  1.0  1.0   7.5   3
          194( 38)     2     10.518     20.330      8.124      1.003     0.000     29.956  2484.368     3.105  1.0  1.0   7.6   1
          194( 35)     2     10.887     15.739      7.084      0.000     0.000     22.823  2049.873     2.562  1.0  1.0   8.7   2
          194( 40)     2     10.535     23.917      8.837      1.644     0.000     34.398  2763.407     3.454  1.0  1.0   6.8   3
LEAF NO.= 14          DAY MAX LA ATTAINED=194.37         MAX LA=400.00     1
          195( 39)     2     10.703     22.679      8.540      1.326     0.000     32.545  2631.307     3.289  1.0  1.0   7.0   1
LEAF NO.= 13          DAY MAX LA ATTAINED=194.48         MAX LA=420.00     2
          195( 36)     2     11.074     17.608      7.505      0.327     0.000     25.439  2210.498     2.763  1.0  1.0   8.1   2
LEAF NO.= 15          DAY MAX LA ATTAINED=194.24         MAX LA=300.00     3
          195( 41)     2     10.735     25.768      9.286      1.994     0.000     37.048  2892.986     3.616  1.0  1.0   6.2   3
          196( 40)     2     10.800     23.652      8.759      1.497     0.000     33.908  2768.567     3.461  1.0  1.0   6.7   1
          196( 37)     2     11.173     18.598      7.728      0.500     0.000     26.827  2430.510     3.038  1.0  1.0   7.8   2
LEAF NO.= 16          DAY MAX LA ATTAINED=195.69         MAX LA=180.00     3
          196( 42)     3     11.033     25.768      9.958      2.517     0.000     38.243  2967.059     3.709  1.0  1.0   6.0   3
LEAF NO.= 15          DAY MAX LA ATTAINED=196.03         MAX LA=300.00     1
          197( 41)     2     11.099     24.649      9.433      2.020     0.000     36.102  2838.294     3.548  1.0  1.0   6.3   1
          197( 38)     2     11.334     20.214      8.091      0.783     0.000     29.089  2580.110     3.225  1.0  1.0   7.4   2
          197( 43)     3     11.539     25.768     11.096      3.402     0.000     40.266  2985.317     3.732  1.0  1.0   5.5   3
LEAF NO.= 16          DAY MAX LA ATTAINED=197.77         MAX LA=180.00     1
          198( 42)     3     11.644     24.649     10.659      2.974     0.000     38.282  2911.175     3.639  1.0  1.0   5.8   1
LEAF NO.= 14          DAY MAX LA ATTAINED=197.22         MAX LA=400.00     2
          198( 39)     2     11.583     21.631      8.651      1.219     0.000     31.501  2679.298     3.349  1.0  1.0   6.9   2
          198( 44)     3     12.087     25.768     12.330      4.361     0.000     42.459  2973.375     3.717  1.0  1.0   5.0   3
          199( 43)     3     12.209     24.649     11.929      3.962     0.000     40.540  2933.417     3.667  1.0  1.0   5.1   1
LEAF NO.= 15          DAY MAX LA ATTAINED=198.80         MAX LA=300.00     2
          199( 40)     2     11.771     23.504      9.073      1.547     0.000     34.123  2881.886     3.602  1.0  1.0   6.2   2
          199( 45)     3     12.653     25.768     13.603      5.352     0.000     44.723  2961.480     3.702  1.0  1.0   4.4   3
          200( 44)     3     12.661     24.649     12.947      4.754     0.000     42.349  2921.683     3.652  1.0  1.0   4.7   1
```

LEAF NO.= 16 DAY MAX LA ATTAINED=199.73 MAX LA=180.00 2

200(41)	3	12.224	23.504	10.093	2.340	0.000	35.937	2947.089	3.684	1.0	1.0	5.8	2
200(46)	3	13.107	25.768	14.624	6.146	0.000	46.538	2949.634	3.687	1.0	1.0	3.9	3
201(45)	3	13.158	24.649	14.065	5.623	0.000	44.337	2909.995	3.637	1.0	1.0	4.3	1
201(42)	3	12.726	23.504	11.222	3.218	C.000	37.944	3011.000	3.764	1.0	1.0	5.4	2
201(47)	3	13.568	25.768	15.662	6.953	0.000	48.383	2937.834	3.672	1.0	0.9	3.5	3
202(46)	3	13.398	24.649	14.605	6.043	0.000	45.297	2898.354	3.623	0.9	1.0	4.1	1
202(43)	3	12.968	23.504	11.768	3.643	0.000	38.915	2998.956	3.749	0.9	1.0	5.2	2
202(48)	3	13.768	25.768	16.111	7.302	0.000	49.182	2926.082	3.658	0.9	0.8	3.4	3
203(47)	3	13.767	24.649	15.434	6.688	0.000	46.771	2886.760	3.608	1.0	1.0	3.9	1
203(44)	3	13.352	23.504	12.631	4.314	0.000	40.449	2986.559	3.734	1.0	1.0	4.9	2
203(49)	3	14.073	25.768	16.797	7.836	0.000	50.400	2914.377	3.643	1.0	0.8	3.2	3
204(48)	3	14.177	24.649	16.357	7.405	0.000	48.411	2875.212	3.594	1.0	0.9	3.5	1
204(45)	3	13.808	23.504	13.557	5.112	0.000	42.274	2975.010	3.719	1.0	1.0	4.6	2
204(50)	3	14.411	25.768	17.558	8.427	0.000	51.753	2902.719	3.628	1.0	0.7	2.9	3
205(49)	3	14.622	24.649	17.360	8.186	0.000	50.194	2863.711	3.580	1.0	0.8	3.1	1
205(46)	3	14.354	23.504	14.884	6.067	0.000	44.455	2963.109	3.704	1.0	1.0	4.1	2
205(51)	3	14.773	25.768	18.383	9.069	0.000	53.220	2891.107	3.514	1.0	0.7	2.5	3
206(50)	3	14.567	24.649	18.135	8.789	0.000	51.572	2852.255	3.565	1.0	0.7	2.8	1
206(47)	3	14.811	23.504	15.914	6.868	0.000	46.236	2951.256	3.689	1.0	1.0	3.6	2
206(52)	3	15.060	25.768	19.019	9.564	0.000	54.351	2879.541	3.599	1.0	0.6	2.2	3
207(51)	3	15.303	24.649	18.691	9.377	0.000	52.916	2840.845	3.551	1.0	0.6	2.4	1
207(48)	3	15.259	23.504	16.922	7.651	0.000	48.077	2939.450	3.674	1.0	0.9	3.2	2
207(53)	3	15.325	25.768	19.637	10.045	0.000	55.450	2868.022	3.585	1.0	0.5	1.9	3
208(52)	3	15.565	24.649	19.480	9.835	0.000	53.963	2829.481	3.537	1.0	0.6	6.7	1
208(49)	3	15.609	23.504	17.710	8.265	0.000	49.479	2927.692	3.660	1.0	0.7	7.4	2
ANTHESIS OCCURRED ON JULIAN DAY 208													
209(54)	4	15.433	25.768	20.498	10.151	0.000	56.417	2856.549	3.571	1.0	0.5	6.3	3
209(53)	3	16.037	24.649	20.544	10.662	0.000	55.854	2818.162	3.523	1.0	1.0	6.3	1
209(50)	3	16.087	23.504	18.784	9.100	0.000	51.389	2915.980	3.645	1.0	1.0	7.0	2
209(55)	4	15.536	25.768	22.529	10.388	0.000	58.685	2845.122	3.556	1.0	1.0	5.9	3
ANTHESIS OCCURRED ON JULIAN DAY 210													
210(54)	4	16.140	24.649	22.815	10.926	0.000	58.389	2806.888	3.509	1.0	1.0	6.2	1
210(51)	3	16.619	23.504	19.983	10.032	0.000	53.519	2904.315	3.630	1.0	1.0	6.9	2
210(56)	4	15.648	25.768	24.797	10.653	0.000	61.218	2833.741	3.542	1.0	1.0	5.8	3
211(55)	4	16.254	24.649	25.168	11.202	0.000	61.039	2795.660	3.495	1.0	1.0	5.7	1
ANTHESIS OCCURRED ON JULIAN DAY 211													
211(52)	4	16.719	23.504	22.393	10.311	0.000	56.208	2892.697	3.616	1.0	1.0	6.4	2
211(57)	4	15.772	25.768	27.167	10.930	0.000	63.865	2822.406	3.528	1.0	1.0	5.3	3
212(56)	4	16.380	24.649	27.630	11.487	0.000	63.766	2764.477	3.481	1.0	1.0	5.1	1
212(53)	4	16.831	23.504	24.872	10.599	0.000	58.975	2881.125	3.601	1.0	1.0	5.9	2
212(58)	4	15.908	25.768	29.605	11.216	C.000	66.589	2811.115	3.514	1.0	1.0	4.7	3
213(57)	4	16.518	24.649	29.145	11.671	0.000	65.465	2773.338	3.467	1.0	1.0	4.8	1
213(54)	4	16.956	23.504	26.418	10.784	0.000	60.707	2869.600	3.587	1.0	1.0	5.6	2
213(59)	4	16.056	25.768	28.865	11.216	2.435	68.284	2799.870	3.500	1.0	1.0	4.4	3
214(58)	4	16.664	24.649	31.481	11.947	0.000	68.077	2762.244	3.453	1.0	1.0	4.3	1
214(55)	4	17.088	23.504	28.791	11.063	0.000	63.358	2858.121	3.573	1.0	1.0	5.0	2
214(60)	4	16.201	25.768	28.143	11.216	5.777	70.905	2788.670	3.486	1.0	1.0	3.9	3
215(59)	4	16.821	24.649	30.694	11.947	2.987	70.277	2751.194	3.439	0.8	1.0	3.8	1
215(56)	4	17.232	23.504	33.788	11.301	0.000	65.593	2846.688	3.558	0.8	1.0	4.6	2
215(61)	4	16.341	25.768	27.440	11.216	8.503	72.927	2777.514	3.472	0.8	0.9	3.4	3
216(60)	4	16.975	24.649	29.927	11.947	5.577	72.100	2740.189	3.425	0.9	0.9	3.5	1
216(57)	4	17.386	23.504	30.019	11.301	2.833	67.656	2835.300	3.544	0.9	1.0	4.2	2
216(62)	4	16.478	25.768	26.754	11.216	10.838	74.576	2766.403	3.458	0.9	0.8	3.1	3
217(61)	4	17.124	24.649	29.179	11.947	8.064	73.859	2729.227	3.412	1.0	0.8	3.1	1
217(58)	4	17.536	23.504	29.268	11.301	5.750	69.823	2823.958	3.530	1.0	1.0	3.8	2
217(63)	4	16.612	25.768	26.085	11.216	13.095	76.164	2755.337	3.444	1.0	0.7	2.8	3
218(62)	4	17.270	24.649	28.449	11.947	10.537	75.581	2716.309	3.398	1.0	0.7	2.8	1
218(59)	4	17.682	23.504	28.536	11.301	8.613	71.954	2812.662	3.516	1.0	0.9	3.4	2
218(64)	4	16.743	25.768	25.433	11.216	15.297	77.714	2744.314	3.430	1.0	0.7	2.5	3

219(63)	4	17.413	24.649	27.738	11.947	12.712	77.046	2707.435	3.384	1.0	0.7	2.5	1
219(60)	4	17.825	23.504	27.823	11.301	11.155	73.783	2801.410	3.502	1.0	0.8	3.0	2
219(65)	4	16.870	25.768	24.797	11.216	17.246	79.027	2733.336	3.417	1.0	0.6	2.2	3
220(64)	4	17.551	24.649	27.045	11.947	14.174	77.814	2696.604	3.371	0.8	0.6	2.3	1
220(61)	4	17.964	23.504	27.127	11.301	12.833	74.765	2790.204	3.488	0.8	0.7	2.8	2
220(66)	4	16.994	25.768	24.177	11.216	18.551	79.712	2722.402	3.403	0.8	0.5	2.0	3
221(65)	4	17.687	24.649	26.369	11.947	14.997	77.961	2685.817	3.357	0.7	0.5	2.3	1
221(62)	4	18.059	23.504	26.449	11.301	13.728	74.982	2779.042	3.474	0.7	0.7	2.8	2
221(67)	4	17.115	25.768	23.573	11.216	19.286	79.843	2711.511	3.389	0.7	0.5	2.0	3
222(66)	4	17.818	24.649	25.709	11.947	16.330	78.634	2675.073	3.344	0.8	0.5	5.3	1
222(63)	4	18.232	23.504	25.788	11.301	15.258	75.851	2767.925	3.460	0.8	0.7	5.8	2
222(68)	4	17.232	25.768	22.983	11.216	20.475	80.442	2700.664	3.376	0.8	0.5	5.1	3
223(67)	4	17.947	24.649	25.067	11.947	18.849	80.510	2664.371	3.330	0.7	1.0	4.9	1
223(64)	4	18.361	23.504	25.143	11.301	17.797	77.745	2756.852	3.446	0.7	1.0	5.4	2
223(69)	4	17.347	25.768	22.409	11.216	22.944	82.336	2689.861	3.362	0.7	1.0	4.7	3
224(68)	4	18.072	24.649	24.440	11.947	20.958	81.994	2653.713	3.317	0.9	1.0	4.7	1
224(65)	4	18.486	23.504	24.515	11.301	19.926	79.245	2745.824	3.432	0.9	1.0	5.2	2
224(70)	4	17.459	25.758	21.848	11.216	25.006	83.838	2679.100	3.349	0.9	1.0	4.4	3
225(69)	4	18.194	24.649	23.829	11.947	23.004	83.428	2643.097	3.304	0.9	1.0	4.5	1
225(66)	4	18.609	23.504	23.902	11.301	21.990	80.696	2734.840	3.419	0.9	1.0	5.0	2
225(71)	4	17.569	25.768	21.302	11.216	27.004	85.291	2668.383	3.335	0.9	1.0	4.2	3
226(70)	4	18.314	24.649	23.233	11.947	25.989	85.817	2632.523	3.291	0.9	1.0	4.0	1
226(67)	4	18.728	23.504	23.304	11.301	25.000	83.109	2723.899	3.405	0.9	1.0	4.5	2
226(72)	4	17.675	25.768	20.770	11.216	29.923	87.676	2657.709	3.322	0.9	1.0	3.7	3
227(71)	4	18.430	24.649	22.652	11.947	28.621	87.869	2621.993	3.277	1.0	0.9	3.6	1
227(68)	4	18.845	23.504	22.722	11.301	27.808	85.335	2713.003	3.391	1.0	1.0	4.1	2
227(73)	4	17.779	25.768	20.250	11.216	32.382	89.617	2647.077	3.309	1.0	0.9	3.4	3
228(72)	4	18.543	24.649	22.086	11.947	31.329	90.011	2611.504	3.264	1.0	0.8	3.1	1
228(69)	4	18.958	23.504	22.154	11.301	30.846	87.804	2702.149	3.378	1.0	1.0	3.6	2
228(74)	4	17.890	25.768	19.744	11.216	34.911	91.639	2636.488	3.296	1.0	0.8	3.0	3
229(73)	4	18.653	24.649	21.534	11.947	33.417	91.547	2601.057	3.251	1.0	0.7	3.6	1
229(70)	4	19.069	23.504	21.600	11.301	33.185	89.539	2691.340	3.364	1.0	0.8	4.0	2
229(75)	4	17.979	25.768	19.251	11.216	36.653	93.088	2625.941	3.282	1.0	0.7	3.4	3
230(74)	4	18.761	24.649	20.995	11.947	35.250	92.841	2570.652	3.238	1.0	0.9	3.6	1
230(71)	4	19.177	23.504	21.060	11.301	35.194	91.058	2680.574	3.351	1.0	0.9	4.0	2
230(76)	4	18.075	25.768	18.769	11.216	38.574	94.328	2615.437	3.269	1.0	0.8	3.5	3
231(75)	4	18.866	24.649	20.471	11.947	37.242	94.308	2580.289	3.225	0.9	0.9	3.4	1
231(72)	4	19.282	23.504	20.533	11.301	37.374	92.712	2669.851	3.337	0.9	1.0	3.8	2
231(77)	4	18.169	25.758	18.300	11.216	40.451	95.736	2604.974	3.256	0.9	0.8	3.2	3
232(76)	4	18.968	24.649	19.959	11.947	39.357	95.911	2569.967	3.212	1.0	0.8	3.1	1
232(73)	4	19.385	23.504	20.020	11.301	39.655	94.520	2659.171	3.324	1.0	0.9	3.4	2
232(78)	4	18.261	25.768	17.842	11.216	42.445	97.271	2594.554	3.243	1.0	0.8	2.9	3
233(77)	4	19.068	24.649	19.460	11.947	40.779	96.834	2559.687	3.200	1.0	0.7	7.7	1
233(74)	4	19.485	23.504	19.519	11.301	41.248	95.572	2648.533	3.311	1.0	0.8	8.0	2
233(79)	4	18.350	25.768	17.396	11.216	43.777	98.158	2584.175	3.230	1.0	0.7	7.6	3
234(78)	4	19.165	24.649	18.973	11.947	42.946	98.515	2549.447	3.187	0.7	1.0	7.3	1
234(75)	4	19.583	23.504	19.031	11.301	43.433	97.269	2637.938	3.297	0.7	1.0	7.6	2
234(80)	4	18.437	25.768	16.962	11.216	45.908	99.853	2573.838	3.217	0.7	1.0	7.1	3
235(79)	4	19.260	24.649	18.499	11.947	44.958	100.052	2539.249	3.174	0.6	1.0	6.9	1
235(76)	4	19.678	23.504	18.556	11.301	45.462	98.822	2627.386	3.284	0.6	1.0	7.3	2
235(81)	4	18.437	25.768	17.174	11.216	47.333	101.490	2563.542	3.204	0.6	1.0	6.8	3
236(80)	4	19.353	24.649	18.037	11.947	46.656	101.289	2529.091	3.161	0.7	1.0	6.7	1
236(77)	4	19.771	23.504	18.092	11.301	47.176	100.073	2616.876	3.271	0.7	1.0	7.0	2
236(82)	4	18.437	25.768	17.388	11.216	48.451	102.823	2553.287	3.192	0.7	1.0	6.5	3
237(81)	4	19.353	24.649	18.262	11.947	48.578	103.435	2518.974	3.149	0.8	1.0	6.3	1
237(78)	4	19.771	23.504	18.318	11.301	49.118	102.240	2606.408	3.258	0.8	1.0	6.6	2
237(83)	4	18.437	25.768	17.606	11.216	50.386	104.976	2543.073	3.179	0.8	1.0	6.1	3
238(82)	4	19.353	24.649	18.490	11.947	50.523	105.619	2508.898	3.136	0.8	1.0	5.8	1
238(79)	4	19.771	23.504	18.547	11.301	51.084	104.435	2595.982	3.245	0.8	1.0	6.2	2
238(84)	4	18.437	25.768	17.826	11.216	52.346	107.155	2532.900	3.166	0.8	1.0	5.7	3

239(83)	4	19.353	24.649	18.721	11.947	52.856	108.173	2498.862	3.124	0.9	1.0	5.4	1
239(80)	4	19.771	23.504	18.779	11.301	53.265	106.848	2585.598	3.232	0.9	1.0	5.7	2
239(85)	4	18.437	25.768	18.048	11.216	54.519	109.551	2522.768	3.153	0.9	1.0	5.2	3
240(84)	4	19.353	24.649	18.955	11.947	55.115	110.666	2488.865	3.111	1.0	1.0	5.0	1
240(81)	4	19.771	23.504	19.013	11.301	55.383	109.201	2575.254	3.219	1.0	1.0	5.3	2
240(86)	5	18.437	25.768	18.274	11.216	56.630	111.888	2512.676	3.141	1.0	1.0	4.8	3

PHYSIOLOGICAL MATURITY OCCURRED ON JULIAN DAY 240

241(85)	4	19.353	24.649	19.192	11.947	57.096	112.884	2478.909	3.099	1.0	1.0	4.6	1
241(82)	4	19.771	23.504	19.251	11.301	57.234	111.290	2564.953	3.206	1.0	1.0	5.0	2
242(86)	5	19.353	24.649	19.432	11.947	59.337	115.365	2468.993	3.086	1.0	1.0	4.2	1

PHYSIOLOGICAL MATURITY OCCURRED ON JULIAN DAY 242

242(83)	5	19.771	23.504	19.492	11.301	59.332	113.628	2554.692	3.193	1.0	1.0	4.6	2

PHYSIOLOGICAL MATURITY OCCURRED ON JULIAN DAY 242

NODE 1 MERGED INTO NODE 21
NODE 2 MERGED INTO NODE 21
NODE 3 MERGED INTO NODE 21
FIELD MATURITY REACHED
LEAF= 73.921 CLUM= 57.198 HEAD= 34.464 GRAIN= 175.298 FINAL TOTAL WT = 340.881

CORE USAGE OBJECT CODE= 58168 BYTES,ARRAY AREA= 24180 BYTES,TOTAL AREA AVAILABLE= 174176 BYTES

DIAGNOSTICS NUMBER OF ERRORS= 0, NUMBER OF WARNINGS= 0, NUMBER OF EXTENSIONS= 0

COMPILE TIME= 2.73 SEC,EXECUTION TIME= 30.38 SEC, WATFIV - JUL 1973 V1L4 9.58.06 TUESDAY 28 DEC 76

THE USE OF A DYNAMIC DIGRAPH STRUCTURE IN A POPULATION SIMULATION MODEL FOR GRAIN SORGHUM

By

JESS WALTER CURRY, JR.

B.S. Oklahoma State University

Stillwater, 1970

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Computer Science

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1977

ABSTRACT

This paper describes a model for simulating the growth of a field of grain sorghum. The model consists of a digraph structure, each node representing a logical partition of the field based on variables such as available water, emergence dates, or soil types. The structure is dynamic in that each node can diverge if conditions in different parts of the field become different. Equally important, nodes can converge if conditions change again in such a way that 1) the plants in several different partitions reach the same stage of development and 2) these partitions currently have the same conditions. The model may be applied to any crop simulation system where discrete simulation techniques are used to simulate a single "average" plant.