

TABLE OF CONTENTS

	<u>Page</u>
Acknowledgements	i
Introduction	1
Operation of The Simulation	3
Player's Handout	9
Program Documentation	11
Flow Charts	20
Variable List	46

ACKNOWLEDGEMENTS

The design of a training simulation requires not only the expertise of a "game" builder but also the knowledge and wisdom of someone intimately familiar with subject of the simulation. In the case of "Thought For Food" that wisdom was supplied by Margaret Gorecki of CARE/Honduras. Without Ms. Gorecki's insight and willingness to impart that insight, "Thought For Food" could not have been completed. Thanks Marge!

I would also like to thank the people at the USAID mission in Honduras who contributed to this effort. Tom Hyslop took responsibility for a project initiated by his former colleague -- and made it all happen. George Moore helped me get started with a computer I had never seen before. Finally, Santiago Valladeres provided the Spanish for the computer dialogue with the participant.

I should not forget the role played by Paul Hartenberger -- Dr. Hyslop's former colleague who fought the bureaucracy to get a computer and to consummate my contract. Finally, I would like to thank David Nelson, my colleague in Tegucigalpa a year ago, whose vision of the food flow system suggested the need for the simulation and whose persuasiveness convinced us all to go ahead and build it.

THOUGHT FOR FOOD:
AN INTERACTIVE SIMULATION MODEL OF A FOOD DISTRIBUTION SYSTEM

Submitted to:

USAID/Honduras

Under Contract Number AID/SOD/PDC-C-0082/WO-6

By

Roy I. Miller

Community Systems Foundation
Ann Arbor, Michigan

March 15, 1980

INTRODUCTION

To understand "Thought For Food", one must return in time to November of 1978. A two person team from Community Systems Foundation in Ann Arbor, Michigan (David P. Nelson, Ph.D. and Roy I. Miller, Ph.D.) was in Tegucigalpa reviewing the information needs of the agencies running food supplementation programs in Honduras. The primary conclusion of the team was that the food flow system should be self-controlling. The people with responsibility for the various stages in the food flow system needed to learn how to use the information at their disposal to make proper decisions to manage the food flow in a timely and appropriate manner. The transfer of data to a central place (a computerized information system) for analysis leading to action by a central control person would be too time consuming to allow for corrective action in the time frame of the food flow.

Near the end of that consultancy, the CSF team members were chatting casually over lunch at poolside at the Maya hotel with Margaret Gorecki, the Director of Feeding Programs for CARE in Honduras. Dave Nelson spoke of his burgeoning enterprise to market micro-computers in Ecuador. Roy Miller spoke of one of his favorite applications of computers -- gaming-simulation. The notion to use computerized gaming-simulation to train the responsible people in the food flow system arose naturally as a consequence of this causal discussion. When the notion was suggested to Paul Hartenberger, of US/AID, Honduras, it was accepted as a good idea.

The world of government works slowly -- but it works. In February of 1980, Roy Miller returned to Tegucigalpa to implement a training simulation for people acting as distributors in the CARE school feeding program. In the interim, US/AID purchased a small computer (a PCC 2000). Although this computer was neither as small nor as portable as originally recommended, a decision was made to proceed with the simulation.

Thus, "Thought for Food" was conceived. It is an interactive simulation (some use the word "game") in which the computer simulates the behavior

of 10 different schools in a school district while a single participant (or small group) plays the role of regional supervisor in charge of food distribution. The schools are initially "making mistakes." Their storage for their supplementary foods may be inadequate or they may be giving out an incorrect ration of food. In each month of the school year, each school may or may not come to the distribution point to pick-up food; may or may not submit their reports; and may or may not upgrade their storage or correct their rations. These actions are simulated probabilistically inside the computer. However, the probabilities are set in response to the decisions of the participant playing the role of supervisor. The proper actions by the participant raise the probabilities -- but do not guarantee -- that the schools will operate properly.

The computer furnishes the participant with information similar to that which is available to a real supervisor of a distribution center. Using that information and his/her own ingenuity, the participant runs the distribution center; hopefully, deriving lessons about interpreting the information and acting to monitor the schools that will carry over into the real world.

This manual is prepared with two objectives. First, it is hoped that the manual will guide operators of the simulation in its use. Both the operation of the computer and the handling of the participant are discussed. Second, the manual documents the innards of the computer program in some detail. This documentation should enable local programmers to maintain and/or change the program as experience with its use suggest such changes.

OPERATION OF THE SIMULATION

There are two skills needed to operate the simulation. First, one must be able to perform the mechanical tasks of running the computer. Second, and more important, one must be adept at guiding the participant through the simulation so as to maximize his/her learning. The computer skills are easily learned. The teaching skills require sensitivity and a flare for inter-personal interaction best derived through experience. These skills are difficult to write about.

The Computer

The PCC 2000 is a micro-computer with keyboard input, a CRT tube for primary output, a printer for secondary output and two diskette drives for auxiliary storage. To run "Thought For Food", two diskettes are used. The first holds the BASIC interpreter (BASIC is the computer language) and the monitor system as well as the "Thought For Food" Program. The second contains the data files used by the program. To run the program

- 1 - Turn on both the computer and the printer (switches in back)
- 2 - Place the BASIC-PROGRAM diskette in the left hand diskette drive (the "0" drive)
- 3 - Place the DATA diskette in the right hand diskette drive (the "1" drive)
- 4 - After the machine finishes its testing routine (it should be done before you get the disks in -- it takes only a few seconds), close the diskette drive doors.
- 5 - Key in LØ and hit RETURN
- 6 - Enter the date and time in the exact format shown: for example
02/29/80
08:30:15

- 7 - Key in MOUNT Ø and hit return
- 8 - Key in LOAD "TFF" and hit RETURN
- 9 - Key in RUN and hit RETURN

After a brief self-initialization, the first "screen" with game instructions will appear.

If the computer is in use (already turned on) when you begin, remove the diskettes in the drives and, instead of powering-up again, simultaneously hit the ENABLE and RESET buttons under the diskette drives. This action initiates the machine self-test routine and clears it for additional use. Then proceed with step 2 as before.

The program cycles over 24 months. If you wish to terminate the program prematurely, hit both the CTRL key and the Letter "C" simultaneously. This action returns control of the computer to the monitor system. Then, key in UNLOAD and hit the return key. Remove the diskettes, place them in their protective jackets and turn off the machine. If the program terminates normally, it is unnecessary to key UNLOAD.

The operation of the printer (paper loading etc.) is described in the PCC-2000 manual which should be kept near the machine at all times.

There are three other computer actions that can be taken by the operator. The first occurs during the running of the program. When the player is asked to key in his decisions (1-7), a response of 99 transfers control of the program to a subroutine that enables the operator to look at the school data files. This may prove useful for review of history during the game or further explanation of what is happening. (At this time, both the prompts and formats are in English.)

Also, the operator may chose to reinitialize the starting data for either the schools or the distribution center.

To alter the starting data for the schools, it is necessary to run a separate "stand-alone" program called "MAKE1." The data values for each school are stored in data statements in the program. The order in which the data are arranged for each school is:

- 1) School Number
- 2) Matriculation Two Years Previous
- 3) Matriculation Last Year (At the start of the program, a new Matriculation figure is computed for the year being run. This new figure is placed in this slot and the initial value is shifted to variable 2.
- 4) Storage condition
- 5) W.S.B. ration
- 6) Sharing Rate (.9 in all cases)
- 7) Nutritional Status
- 8) Attendance (always initialized to 0)
- 9) Adequacy Flags (always initialized to 0)
- 10) Stocks of W.S.B. on hand by age (eight entries)
- 11) Stocks of Milk on hand by age (eight entries)
- 12) Assorted Flags (always initialized to 0)

A change in any value can be made by altering the data statement and running "MAKE1". In BASIC, the easiest way to alter a statement is to reenter the entire line -- including line number -- after loading the program. Thus, to change a school-value, proceeding through steps 1 to 7 for running "TFF." Then

8 - Key in LOAD "MAKE1" and hit return

9 - Rekey the line with the value to be changed (including the line number)

10 - Key in RUN and hit return

To save the revised program, (which will destory the original!!!), key

11 - SAVE "MAKE1", Ø

At this point, one can LOAD "TFF" or stop by issuing the UNLOAD command.

To alter the starting data for the distribution center, the identical procedure is followed -- except that the program used is "MAKE3". The order of the data values is

- 1) Work units used (initilize to 0)
- 2) Stocks of W.S.B.
- 3) Stocks of Milk
- 4) Children Attending School
- 5) Number of Malnourished Children
- 6) Flag indicating WSB received (initialize to 0)
- 7) Flag indicating Milk received (initialize to 0)
- 8) Matriculation Last Year
- 9) Matrix of Play Decisions (initialize to 0)

Finally, it is possible for the operator to use "stand-alone" programs to look at the intermediate school files and distribution center files generated during a run. These two programs are "PRSF" and "PRDF." The same procedure for running "TFF" is followed except that either "PRSF" or "PRDF" is loaded in Step 8. Note, when running any of the auxiliary programs (or TFF) don't forget the quotation marks.

The Player

As noted earlier, interacting with the participant is more of an art than a learned skill. It is simultaneously desirable to feed the player hints and cues so that he will "learn" how to control the system while not "spoon feeding" him answers.

Let me illustrate with an example. Each school is subject to a "disaster" in which some of its food is destroyed. The probability of a "disaster" occurring is directly related to the quality of storage in the school. The ideal situation occurs when the player discovers for himself that some food has disappeared from a school -- a fact that should be immediately obvious from the monthly report. When he questions the "error" in the report, one should suggest only that he visit the school to see what happened. Upon his "visit", one can explain the "disaster" process and its relationship to storage condition. The less than ideal situation occurs when the player has to be told that a "disaster" has taken place. The worst situation is when he never finds out at all.

Each exercise must be run differently depending on the acumen and/or aggressiveness of the player. As the exercise draws to a close (or, in fact, at the close), the operator should help the participant sort out the important elements of his experience. The excitement and mystique of the computer may inhibit such reflective thought. Yet, the learning comes from understanding the model -- not from pushing the buttons. Lessons such as how to time requests that schools pick-up food or that periodic visits to each school should be made to monitor their rations and storage conditions must be clarified and accentuated. The act of "playing" illustrates the lessons -- the operator makes sure they are learned.

Because the exercise has a "two-year" cycle, one strategy might be to break for a few minutes at the end of year 1 and plan out a course of action for year 2. In this exercise, one expects the participant to foul up in year 1 -- it is all so new to him. A planning session at its close may help the participant make some order out of the process and enhance his/her ability to have a successful second year.

Although the exercise can be a one-person activity, it is often advantageous to have two people play the role of distribution center supervisor. The debate between the two reinforces the learning and may lead both participants to greater understanding.

Because each exercise is unique, it is difficult to prescribe a set of steps for the operator. We suggest that the player(s) be given the short player's handout in the next section of this manual before beginning. Be ready to respond to their questions as they arise. Know the model well so that your answers are correct. (To learn the model, read the documentation chapter of this report). Conduct a critique or discussion of the exercise at either its conclusion, between years, or both. Allow people to discuss their frustrations -- both with the exercise and the real world -- during these debriefings. A training exercise should not only teach, but should make people "feel good."

PLAYER'S HANDOUT

With the aid (or perhaps, magic) of a computer, it is possible to condense time -- to squeeze the events of two years into two hours. The technique for doing this is called "simulation." For the next hour or two, you will be part of one such simulation -- a replication of some of the circumstances which confront the people responsible for the distribution of food commodities received from a government run central warehouse for the purpose of dissemination to people at more distant points of consumption (for example, local schools).

A simulation condenses time by reproducing the outcomes of actions that span weeks of "real time" in short, discrete time intervals. In this particular simulation, you will make all of the decisions made by a regional distributor of government food commodities each month (each "round") of the simulation. In turn, the computer will simulate the reactions of both the schools and the central warehouse to your decisions. You then make a new set of decisions and the computer responds again -- and so on for 24 months or "rounds". To achieve this byplay, the simulation is interactive. You will have the opportunity to sit at the keyboard of a computer and "talk" to it. It is hoped that your participation in this exercise will enhance your understanding of the problems which confront the person representing the middle link in a food distribution chain -- a link which has no firm control over the actions and behaviors of the adjacent links.

In a moment, you will sit at the computer. It will provide you with information and ask you questions. The computer will pause at each question and wait patiently for your response. You enter your response by hitting the appropriate key on the keyboard and, then, hitting the oversized RETURN key on the lower right of the main keyboard. Please, do not be afraid that hitting a wrong key will confuse or break the computer. It is surprisingly tolerant of error and more durable than it appears.

To help you begin, let us tell you something about the district. It has 10 schools with matriculation as of "last" year as indicated in the following table:

Name	Students
1 - AGUA FRIA	42
2 - LAS MINAS	61
3 - EL GATO	31
4 - PINALEJO	86
5 - MORAZAN	49
6 - SAN JAUNCITO	27
7 - DULCE NOMBRE	78
8 - SANTA ANA	36
9 - LEMPIRA	29
10 - LAS FLORES	91

The school year starts in the middle of February. Food is supposed to be eaten in the schools in each of the eight months from March through October. During the months of February through October, you will be called upon to make decisions. In each of those months, your activities will be limited. You will start each month with 14 "work units" to expend. Each action on your part costs you some "work units."

The simulation starts slowly but by "April" you will be well into it. Enjoy yourself! Remember, the simulation operator is here to help you; however, you are the decision maker and you must act.

PROGRAM DOCUMENTATION

The "Thought For Food" program has two major components -- a component to process the distribution center activity (the decisions of the participant) and a component to simulate school behavior.

The distribution center component is divided into two modules. The first simulates the receipt of food commodities from the central warehouse. The second processes the participant decisions, in interactive mode, and sets "flags" used by the school component to make the schools responsive to participant decisions.

The school component has six modules: food pick-up, stock aging, feed the children, monthly report, update nutritional status and visit report.

The program documentation describes each module of the two components in some detail. Flow charts of the modules are included in a subsequent section. Finally a list of variable names and definitions is presented. Before describing the separate modules, a brief discussion of file management is included.

File Management

The "Thought For Food" program accesses 22 data files during a two year run; however, only three files are "active" (open, in computer jargon) at one time. Each file is a sequential file. The diskette containing the data files should be placed in the right hand disk drive (drive 1) of the computer.

Three of the files are linked to the distribution center component of the model. The primary active file is called "DISCEN." The file contains 13 records. (Conceptually, it is useful to think of a file as a collection lines called records. Each record has the same set of variables -- only the values of the variables differ from record to record.) The first record contains the initial values of all the variables saved during

the distribution center component. The second record contains the values attained by those same variables after the first month of processing. The third record contains the values after the second month -- and so on.

After the first 12 months (after the first year), the "DISCEN" file is renamed "DC.Y1." A new "DISCEN" is created for use in the second year. Like the original file, this "DISCEN" file adds a record with each month of processing.

Finally, there is an initial file "INITDC" with the initial values of the variables preserved in "DISCEN." At the very beginning of the program, "INITDC" is copied into "DISCEN" prior to the first month of processing.

The remaining files are linked to the school simulation component. Because the feeding program is in operation for only eight months, there are 9 files per year containing school information. The first contains the initial configuration; the second, the status after one month, etc. Each file contains 10 records -- one for each school.

At the end of the first year, the nine files -- "SCH0", "SCH1", ..., "SCH8" are renamed "SCH0Y1", "SCH1Y1", ..., "SCH8Y1." The contents of "SCH8Y1" are transferred to "SCH0" for the start of the second year. Note, however, the commodities are aged for four months during this transfer. During the second year, the files "SCH1", ..., "SCH8" are rebuilt with the results of the second year processing.

Finally, there is an initial file, "INITSC" with the initial values for all the variables preserved for each school. At the very beginning of the program, "INITSC" is copied into "SCH0" prior to the first months processing.

Thus, we have 22 files -- 2 initialization files, one active distribution center, one historical distribution file, 9 school files for the first year and 9 for the second. At any one time, only two school files are

open -- the one with the status of the schools before a month begins and the one which will receive the new status as the month ends.

Distribution Center Component

The distribution center component is invoked once each month; that is, each module is called during each of the 24 months of the simulation.

A. Shipment From Warehouse Module

In the ideal year, the distribution center receives its stocks of W.S.B. and MILK from the central warehouse twice -- in February and again in June. In practice, the distribution centers often receive their commodities in a more staggered, often delayed, mode. The nature and degree of delay is a function of the vagaries of international food exports and Ministry controlled trucking schedules.

To simulate these vagaries, the warehouse component brings food to the distribution center at random each month during a school semester until all food commodities have been received. In each month, none, half or all of each commodity may be received.

If none of a given commodity has been received, the probability is .25 that none will be, .25 that half will be and .50 that all will be received. If half has already been received, the balance will arrive with probability .25.

Note, the amount shipped is predicated on last year's matriculation figures and, therefore, may be inadequate.

B. Participant Decision Module

The Participant Decision module is the interface between the simulation and the participant. During this segment of the program, the

participant receives his/her monthly summary (stocks on hand, children fed adequately, etc.) and makes his/her choices for the month. The accounting for the work units is done as decisions are made. Finally, elements in a matrix of flags are set to 1 (they are 0 at the start) when a decision to take action is made.

School Simulation

The School Simulation is invoked 10 times (once for each school) in each of the months from March through October. Except for the responses to a school visit by the participant, the School Simulation responds to the set of participant decisions made in the prior month. For example, a request that a particular school pick-up its food made in March triggers such an action by the school in April.

A. Food Pick-up Module

A decision by the player to request that schools come to the distribution point triggers a response by the school. The standard response is that the school picks-up food with probability .7 in the first month after the request and .4 if it is more than one month after the request. The amount picked-up is calculated on the basis of last-year's matriculation; therefore, it may be incorrect. If the player "Fed the Pigs", one less bag than required is picked-up -- the bag that goes to the player's pigs.

With only two exceptions, a school comes to the distribution point in only one month of the four month semester. The exceptions are:

- 1) if the player visits the school in one month and issues a second request for a pick-up in the next month (Note, unlike the standard response, this sequence triggers a pick-up in the same month as the request)
- 2) A second pick-up was requested and, due to a shortage at the distribution point, the first pick-up was deficient.

B. Age Stock Module

A certain amount of food stored at each school spoils every month. The rate of spoilage depends on the conditions under which the food is stored. Also, larger amounts of food are destroyed by "disasters" which occur with higher probability under worse storage conditions.

The first step in the aging module is the upgrading or downgrading of the storage facility. There are three classes of storage -- poor, medium and good. If no visit is made to the school, the storage condition is downgraded randomly with probability .05. If a visit is made, the probability of upgrading storage condition by one category is .5. The quality of storage never changes by more than one category during any one month.

The stock is then aged. One percent per month is destroyed, if storage is good; 3 percent destroyed, if storage is fair; and 5 percent destroyed if storage is poor.*

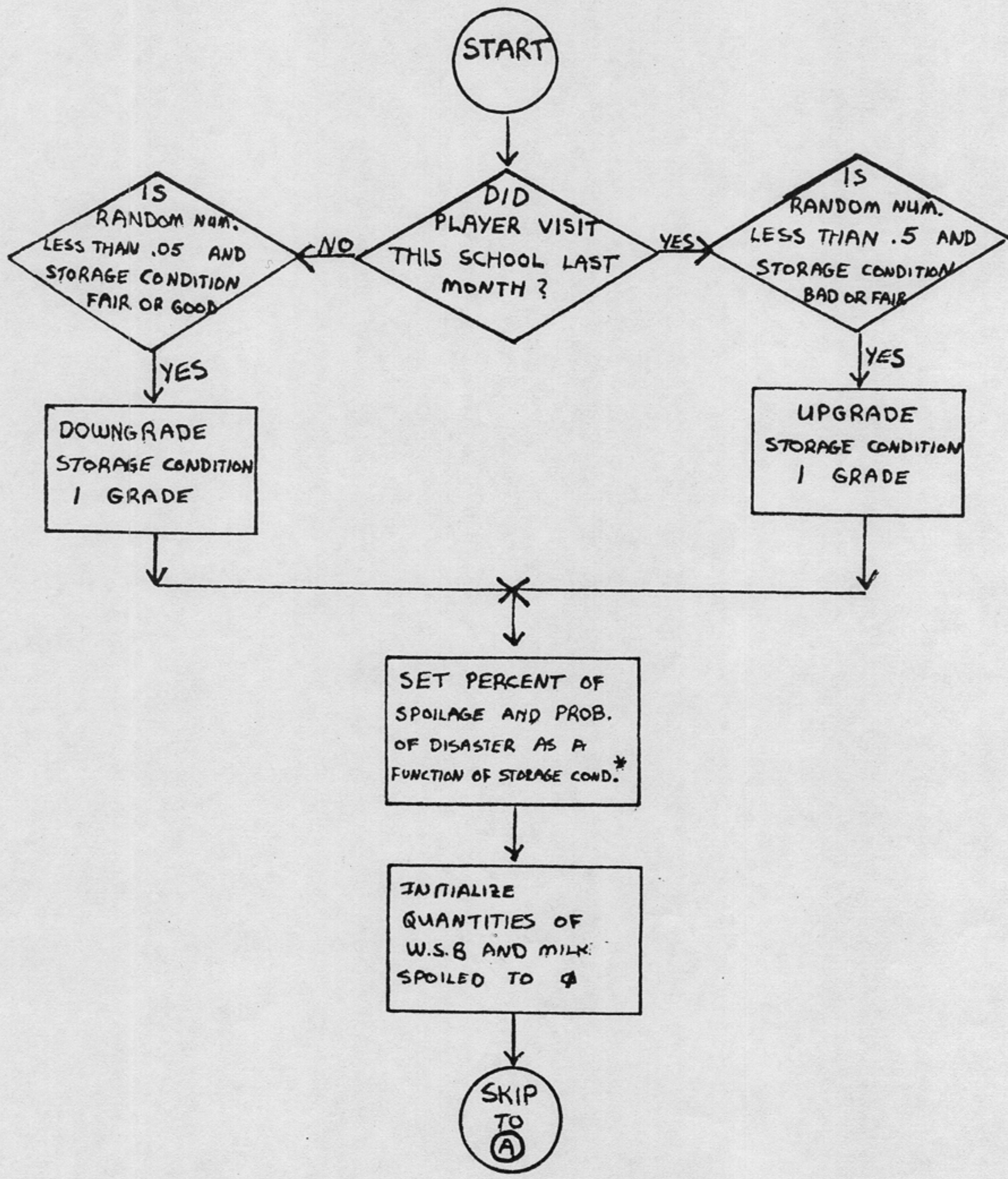
In each month, there is the possibility that an additional 25 percent of the stock is destroyed. If the storage is good, this happens 1 percent of the time, if storage is fair, disaster strikes 3 percent of the time and, finally, if storage is poor, disaster strikes with probability .05.

C. Feed the Children Module

The first step in determining the amount of food reaching the children is the resetting of the ration. Unless the school is visited or the supervisor at the distribution center (the participant) requests that the school correct its ration, no change is made. If a visit is made, the ration is corrected with probability .7. A request for a change causes a change with probability .4. The only change in

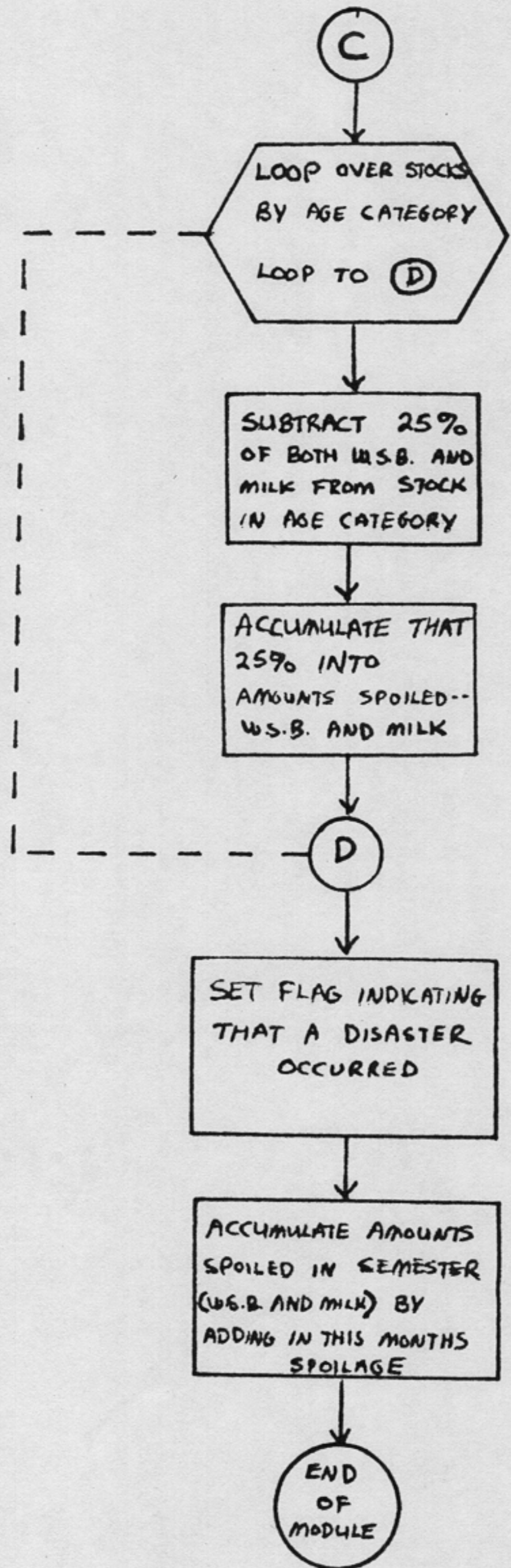
*The original intent to age the stock at a differential rate depending on how "old" it is will be implemented in a later version.

SCHOOL SIMULATION STOCK AGING MODULE

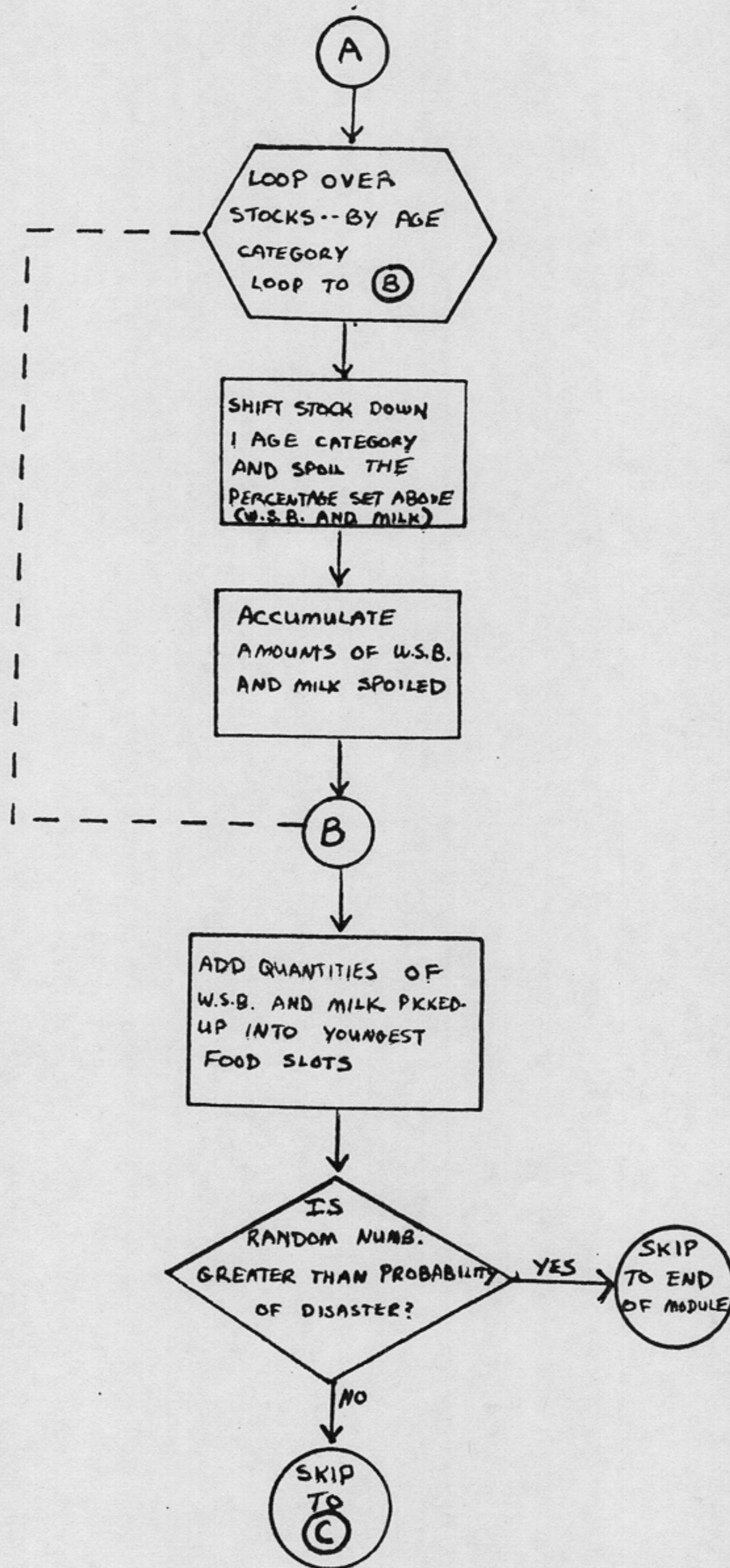


* STORAGE CONDITION	PCT. SPOILAGE	PROB. DISASTER
POOR	.01	.01
FAIR	.03	.05
GOOD	.05	.1

STOCK AGING
MODULE ③

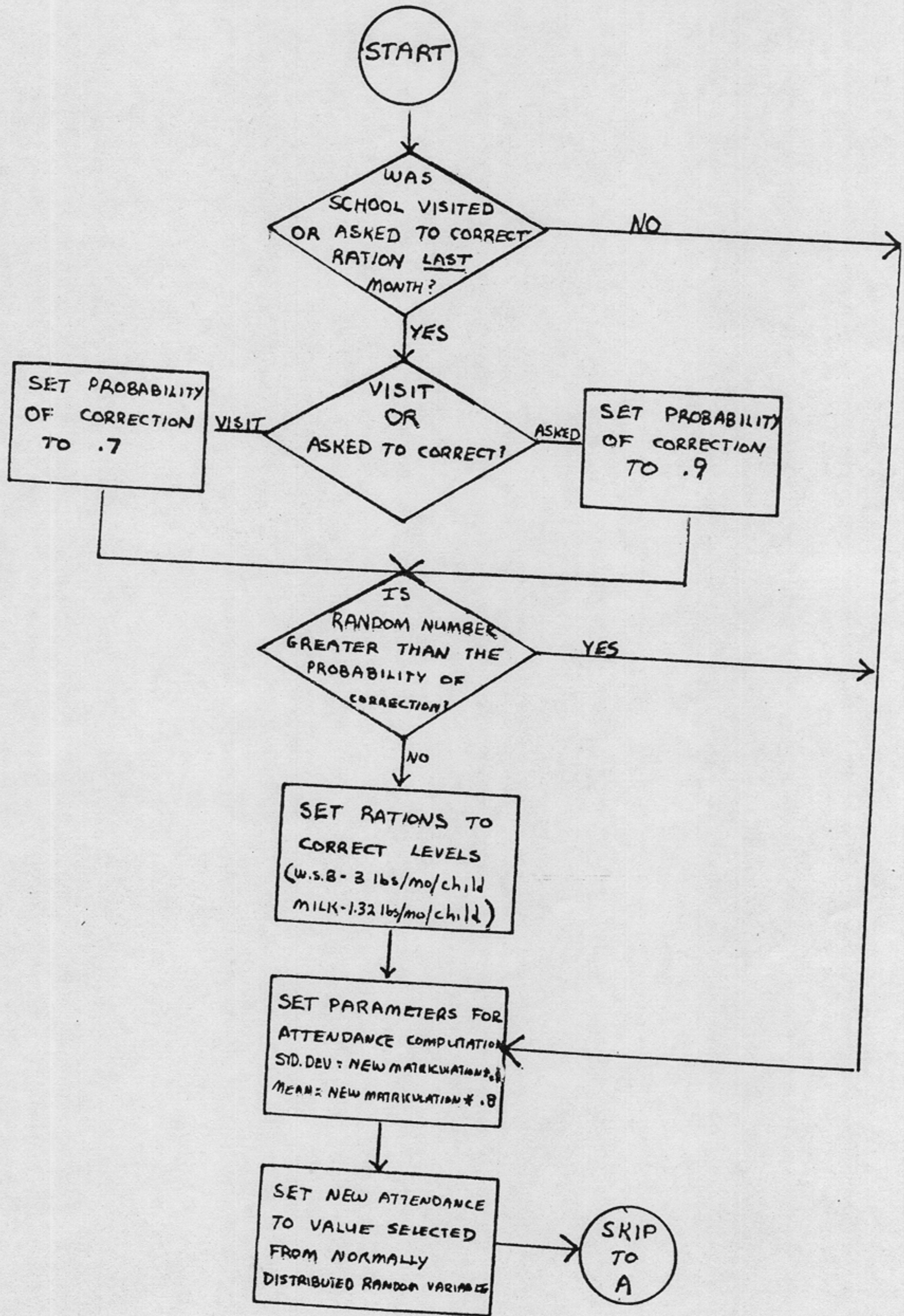


STOCK AGING
MODULE ②

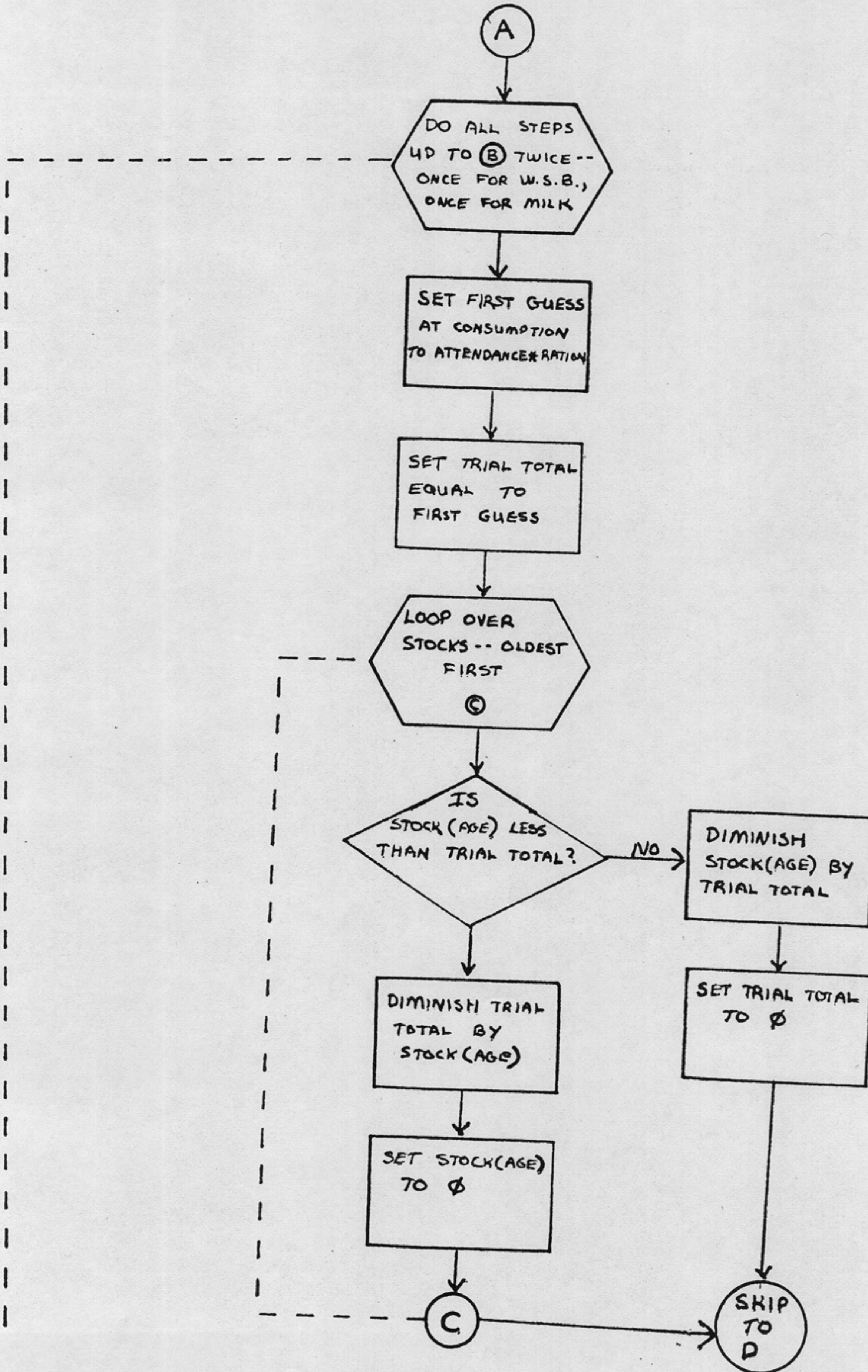


SCHOOL SIMULATION

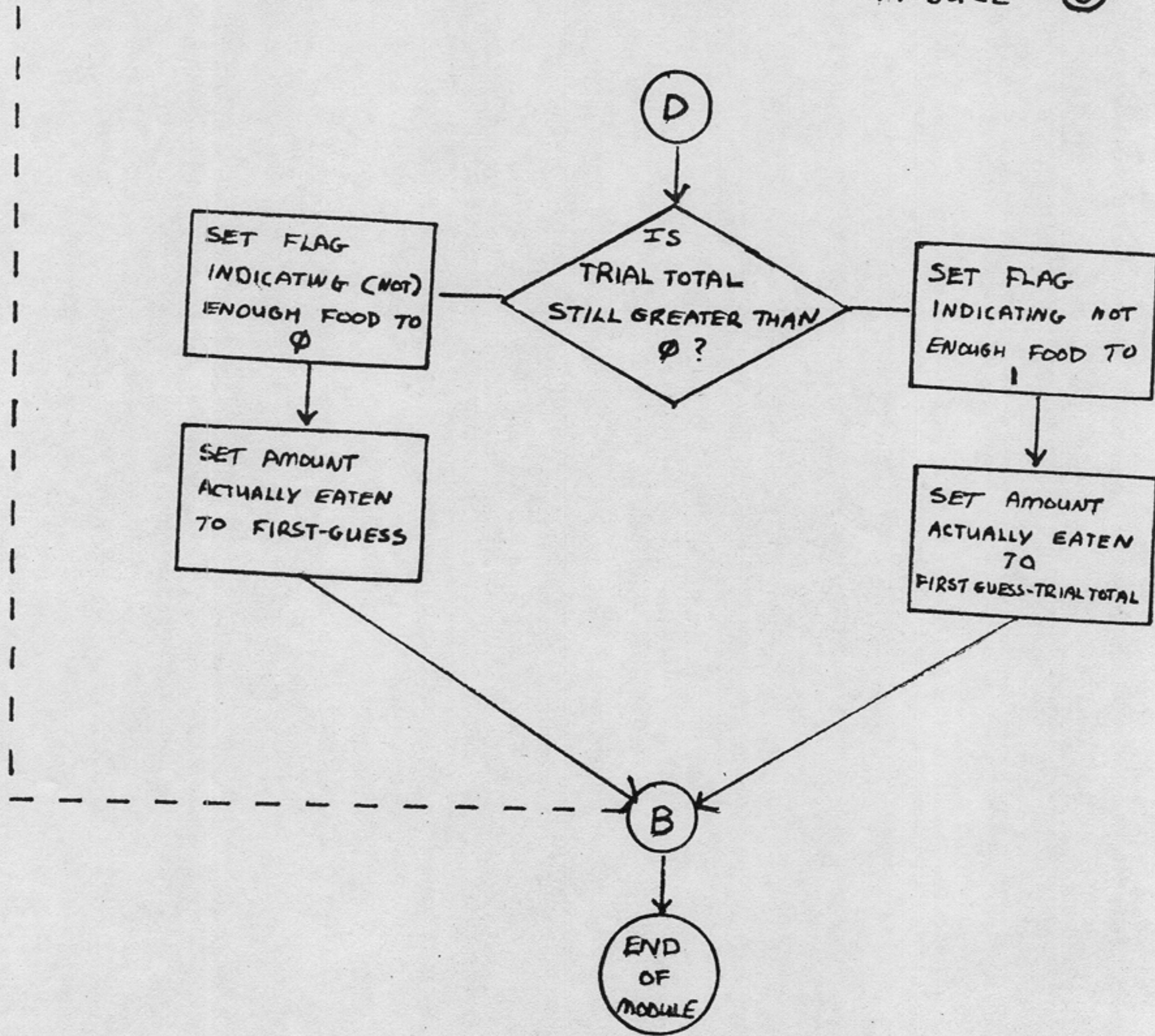
FEED THE CHILDREN MODULE



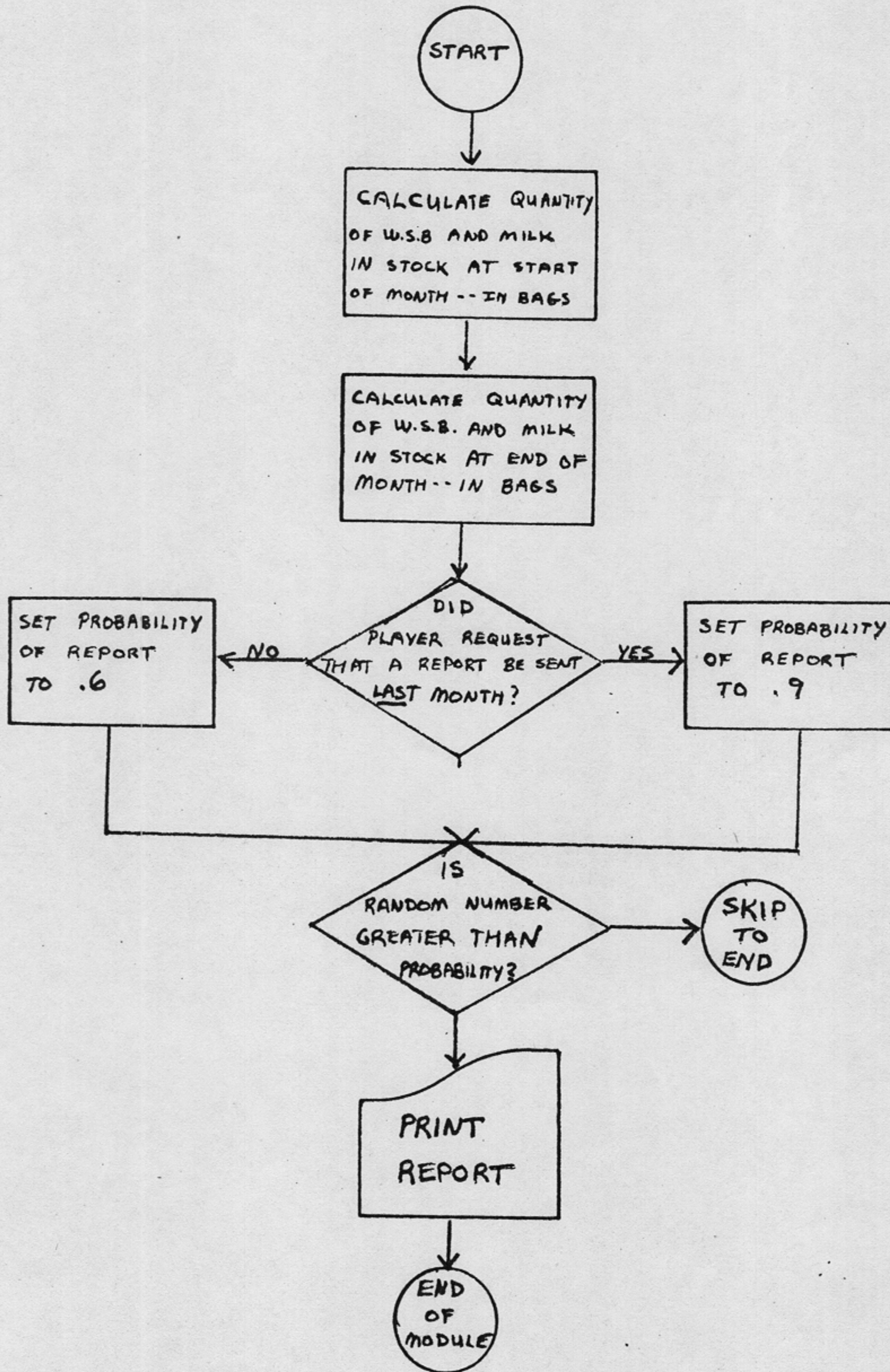
FEED THE CHILDREN
MODULE ②



FEED THE CHILDREN
MODULE ③

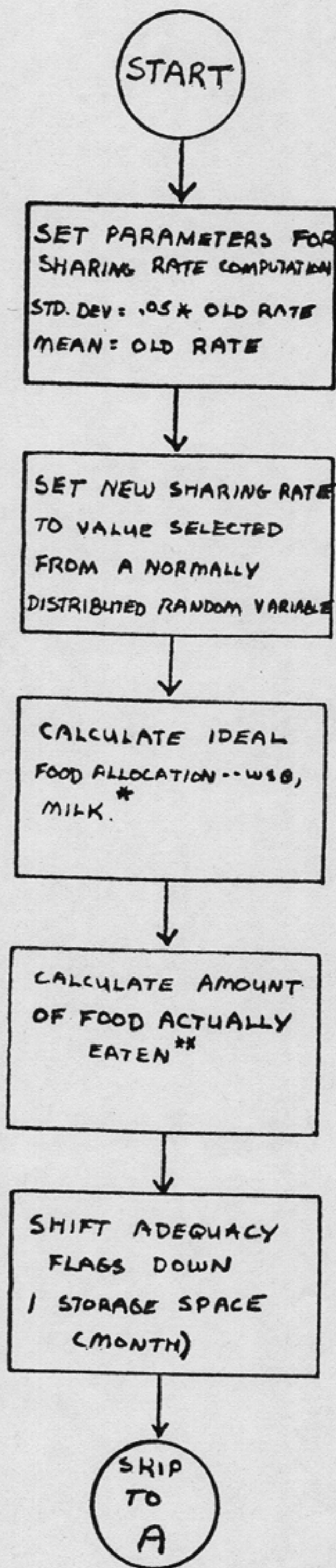


SCHOOL SIMULATION
MONTHLY REPORT MODULE



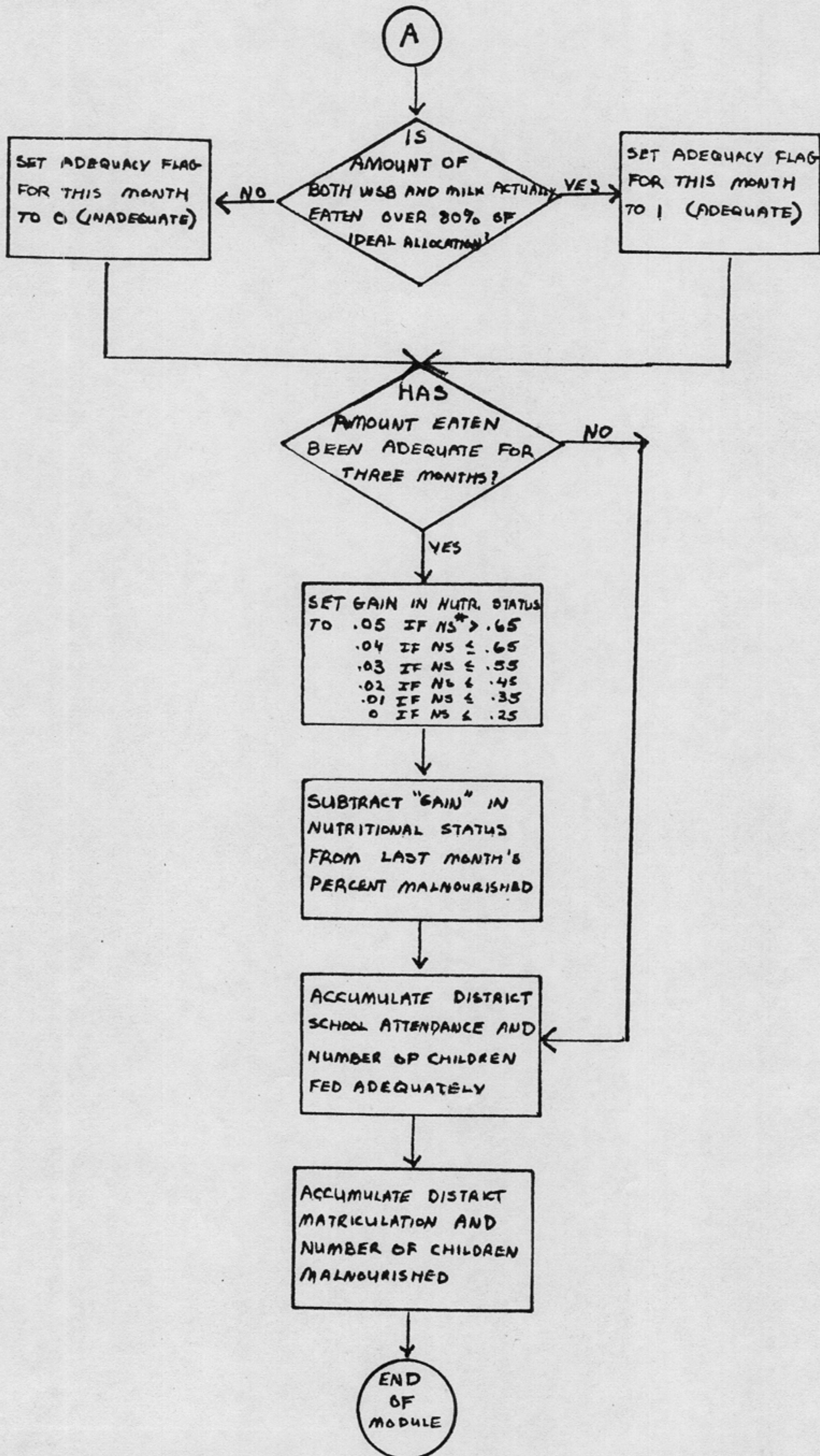
SCHOOL SIMULATION

UPDATE NUTRITIONAL STATUS MODULE



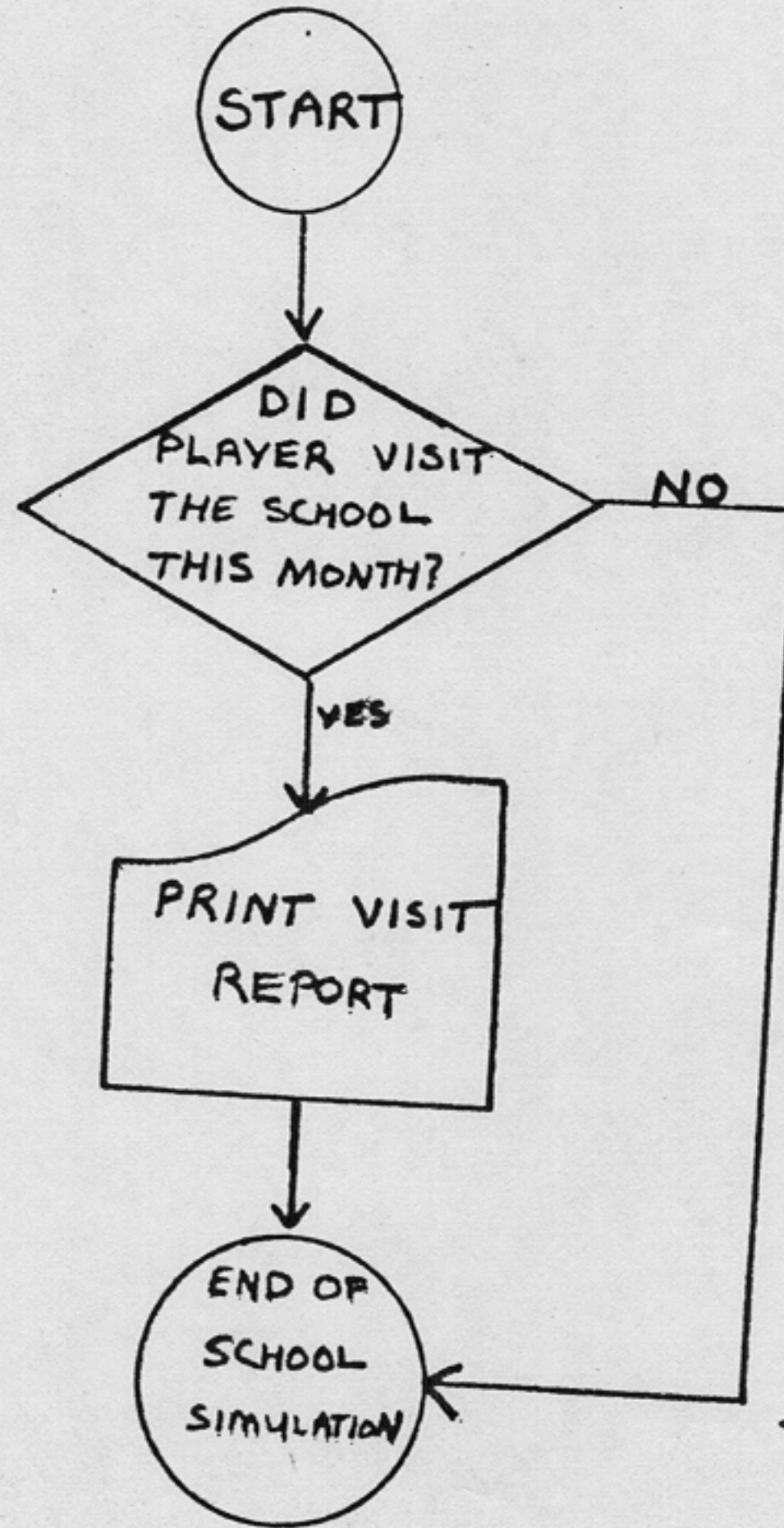
* IDEAL ALLOCATION = ATTENDANCE * IDEAL RATION

** FOOD EATEN = CONSUMPTION (SET IN FEED MODULE * SHARING RATE



* NS IS LAST MONTH'S PERCENT MALNOURISHED

SCHOOL SIMULATION VISIT REPORT MODULE



VARIABLE LIST
STORED IN DISTRIBUTION CENTER FILE

Variable(s)	Description
W1	Work Units available to player
D1, D2	Quantities of W.S.B.(D1) and MILK(D2) in district warehouse--in bags
D3	Number of students matriculated in district
D4	Number of malnourished students in district
D5, D6	Indicator of receipt of W.S.B.(D5) and MILK(D6) during a semester: 0=NONE, 1=HALF, 2=ALL
D7	Number of students attending school in district in a given month
D8	Number of students receiving an adequate ration in district in a given month
NF(10,5), MF(10,5)	Matrix of indicators for storing player decisions--NF holds last month's decisions, MF holds this month's decisions. (Only this month's decisions are in file)

STORED IN SCHOOL FILE

Variable(s)	Description
SN	School number
S1	Matriculation last year
S2	Matriculation this year
S3	Storage conditions in school
C1, C2	Ration in effect in school--W.S.B.(C1) and MILK(C2)
SR	Sharing rate in school
NS	Nutritional status (percent malnourished) in school
S4	Attendance
AA(3)	Array of adequacy of feeding--each space corresponds to a month: current, last, two months ago
X1(8), X2(8)	Stock of W.S.B.(X1) and MILK(X2) on hand by age of stock
F1	Flag indicating whether food has been picked-up from district warehouse in semester
F2	Flag indicating whether player requested a pick-up in semester
F3	Flag indicating whether player fed the pigs in this semester
F4, F5	Flag indicating whether sufficient quantities of W.S.B.(F4) and MILK (F5) were picked-up during visit to district warehouse
F6	Flag indicating whether player visited school last month

Variable(s)	Description
F7	Flag indicating whether disaster hit stocks in this semester
F8	Flag indicating whether player asked for a correction of rations last month
G1, G2	Flags indicating whether students ate enough W.S.B.(G1) and MILK(G2) this month
G3	Flag indicating whether player requested a report last month
G5, G6	Amount of W.S.B.(G5) and MILK(G6) spoiled during the semester--in pounds
G7, G8	Running total of amount of W.S.B.(G7) and MILK(G8) picked-up at district during this semester

OTHERS IN APPROXIMATE ORDER OF APPEARANCE

Variable(s)	Description
R	Argument for random number generator
YC	Year counter
M1	Month number (counter of main program loop)
AM	Intermediate indicator of amount of a commodity to come to district warehouse in a month: 0=NONE, 1=HALF, 2=ALL
LBS	Quantity of a commodity that should arrive at the warehouse--in pounds
LC	Quantity of a commodity that should arrive at the warehouse--in bags
P9	Counter for number of visits in a month
M 8	Name of current month
P1, P2, P3	Working variables--used repeatedly throughout the program for intermediate calculations
CH	Number of player's choice--in decision loop
Z5	Number of school chosen by player--in decision loop
TU	Variables assigned "cost" of each task chose by player
FL 8	File name school file--input
GL 8	File name for school file--output
N1	School number (index for school loop)
M	Month-2
SD	Standard Deviation in a computation to get a value from a normally distributed random variable
EX	Mean in a computation to get a value from a normally distributed random variable
T	Variable used in computing running totals
Y1, Y2	Quantity of W.S.B.(Y1) and MILK(Y2) in school at start of a month--in pounds

Variable(s)	Description
V1, V2	Quantities of W.S.B.(V1) and MILK(V2) picked-up from district warehouse in a month--in bags
A1, A2	Quantities of W.S.B.(A1) and MILK(A2) that should be picked-up for the semester--in pounds
A7,A8	Quantities of W.S.B.(A7) and MILK(A8) that should be picked-up for the semester--in bags
RW, RM	Quantities of W.S.B.(RW) and MILK(RM) spoiled during a month
TT	Trial total (used in feeding children)
K1, K2	Quantities of W.S.B.(K1) and MILK(K2) in school at beginning of month--in bags
Y3, Y4	Quantities of W.S.B.(Y3) and MILK(Y4) in school at end of month (after feeding and spoilage) - in bags
K3, K4	Quantities of W.S.B.(K3) and MILK(K4) eaten during a month--in bags
E1, E2	Quantities of W.S.B.(E1) and MILK(E2) actually eaten during a month (consumption less sharing)
A1, A2 (reused)	Quantities of W.S.B.(A1) and MILK(A2) which should be eaten during a month--in pounds
NA§	School Name
T5, T6	Quantities of W.S.B.(T5) and MILK(T6) spoiled in a month--in bags
K5, K6	Quantities of W.S.B.(K1) and MILK(K2) which should be consumed in a month--in bags