c./

A DAIRY CATTLE BREEDING AND MANAGEMENT COMPUTER SIMULATION PROGRAM FOR TEACHING AND RESEARCH

by

John Skinner

B.Sc., The University of British Columbia, 1978

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

(Department of Animal Science, Faculty of Agricultural Sciences)

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

April 1986

78

In presenting this thesis in partial fulfilment of the requirements for an advanved degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representitives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of ANIMAL SCIENCE

The University of British Columbia
2075 Westbrook Place
Vancouver, Canada
V6T 1W5

Date 28/04/86

ABSTRACT

The increased power and availability of computers has resulted in an increase in the value of simulation as a means of furthering our understanding of systems. Reducing the components and interactions of a system to mathematical models enables simulation to provide a clear basis of the system and this can be useful for teaching and research. Simulation is especially suited for studying genetic gain in dairy cattle because there are already reliable mathematical models available. Analyzing genetic gain in practice is difficult due to the numerous and diverse factors that affect it.

This project has provided a computer program that simulates the inheritance of the economically important traits in dairy cattle and includes interactions with the biological, management and economic factors which can affect genetic gain. It was designed primarily as a teaching tool for senior undergraduate students in animal genetics or dairy science, to heighten students interest and encourage them to think more deeply about the subject. Features were also included to make the simulation useful in research for stimulating and refining research objectives and for analyzing questions not easily tested in the field.

The program models the system at the cow level with critical management decisions made on a continuous basis and summaries and other management decisions on a calendar year basis. Mature equivalent milk production is given in Breed

Class Averages and the quota system of limiting milk production is used so that any small Canadian dairy population can be simulated. Parameters are provided and documented for simulating a Fraser Valley population.

Since a dairy population is an extremely complex system improvements and expansions to this simulation can be made. While some expansions and improvements are possible with existing information many would require more research to provide parameters and bases for models. Use and testing should reveal the changes that are the most beneficial and feasible.

The program has been written in a modularized form to more easily facilitate changes and additions. Full instructions, sample runs and documentation have been included to encourage knowledgeable use and expansion of the program.

TABLE OF CONTENTS

Chapter 3.

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I am grateful to Dr. R.G. Peterson for providing the inspiration and guidance that made this project possible. I would like to thank the members of my committee Drs. J.A. Shelford, K.M. Cheng, L.J. Fisher, R. Blair and C.C. Short for their criticisms and suggestions. I would also like to thank friends and family who provided encouragement and assistance, especially my wife for her help in preparing the final draft and her tolerance throughout.

INTRODUCTION

Simulation can be defined as applied systems analysis, with a system being defined in general terms as a set of objects together with the relationships between the objects and their attributes (Hall & Fagan, 1956). The aim of most scientific endeavor has been analytical, using deductive reasoning to understand complex situations by reducing them into smaller components (Joandet, 1975). With the concomitant radical increase in the volume of knowledge, intuitive thinking alone is often no longer able to take into account all of the relationships and interactions between the components. Systems analysis provides a structured approach to studying a system the results of which include: (1) revealing gaps and weaknesses in our knowledge of the system; (2) providing the basic structure of the system for teaching and application; (3) providing a clear perspective for planning further research; and (4) allowing preliminary testing of ideas and hypothesis.

The value of this approach has been greatly enhanced by the increasing availability and capabilities of computer facilities. Simulation using mathematical modelling techniques and computer programming provides a powerful tool for analyzing complex systems with the only limitation being the adequacy of the mathematical models in describing our knowledge.

Simulation has proven especially useful in Animal Science where experiments with large animals are expensive and can take many years. The growing availability of computers in Animal Science has resulted in a parallel increase in the use of computer simulation in teaching and research. Experiments for preliminary research and to aid in teaching can be performed on simulated populations cheaply and quickly. Most simulations to date, however, have had poor documentation or no documentation at all (Hocking, et al., 1983). This has undoubtedly resulted in much duplication of effort. It has been suggested (Hocking, 1983) that the documentation include: (1) a description of the program; (2) the objectives and procedures; (3) mathematical models used; (4) full instructions for use; and (5) examples of input and output. The list could be extended.

(1) Inclusion of a discussion of the models and the parameters used should reveal gaps and weaknesses in our knowledge of the system and hopefully stimulate research.

(2) Inclusion of flow charts and program documentation will greatly aid in modifying the program to keep pace with our advances in knowledge.

These features would allow programs to be fully utilized in a knowledgeable way by more people.

The aim of the current project was threefold.

(1) Design and build a comprehensive dairy cattle computer simulation program suitable for teaching and research, with special emphasis on the interactions between breeding, management and economics in a dairy enterprise.

(2) Provide full documentation and explanation of the program so that it can be of maximum benefit as a teaching tool for dairy cattle breeding and management courses and as a tool for investigating the complex relationships between breeding programs, management systems and economic returns to a dairy enterprise.

(3) Identify areas where the simulation can be improved by adding to the program, by collecting more data for better parameter estimation and by further research suggesting better models.

Due to the nature of this project it cannot be written up in a traditional format for a Masters Thesis. It has been organized into four chapters and extensive appendices. Chapter one outlines the system to be simulated, the aspects of the system that are of most importance for teaching and research, and the practical requirements of the program operation. Chapter two covers the philosophy of the modelling process and the details of the models used. Chapter three provides parameters for the models to simulate a Fraser Valley herd and discusses their adequacy. The fourth is a concluding chapter that brings together potential improvements in the simulation that are suggested in Chapters two and three. The appendices provide detailed instructions for using the program, annotated program input and output and the details of the program itself.

Chapter 1. OBJECTIVES

This chapter elaborates on the three main objectives of the simulation program.

(1) To make the program capable of simulating a population of Canadian dairy herds as comprehensively and realistically as is feasible.

(2) To tailor the program to be especially useful as a teaching aid for a senior undergraduate course in animal genetics.

(3) To maximize the programs usefulness to researchers in dairy breeding and management.

1.2: System Simulated

The system to be simulated is any Canadian dairy population. The size limits are given in sec. 2.1.1.1. Further limitations necessitated by conflicting goals and limitations of available models are discussed throughout chapter two. The major components and interactions within this system are discussed in this section.

1.2.1: General Description

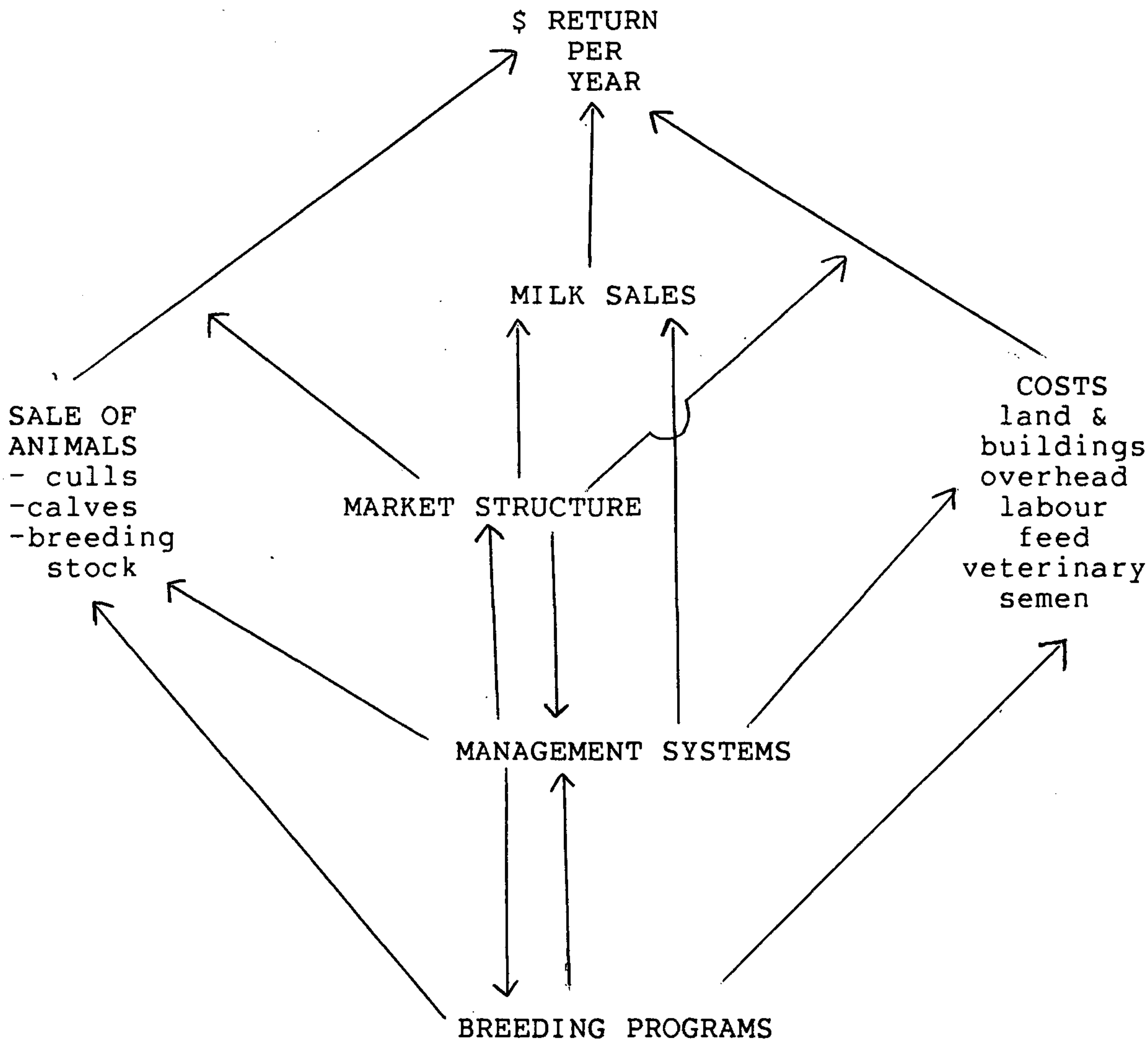
A Canadian dairy cattle population is composed primarily of herds between 30 and 200 cows each. Each herd is managed by an independent dairyman who depends on the operation as his sole or partial source of income. The prices of goods and services supplied by and required by a dairy operation are determined by free market forces with the exception of milk price. Milk in Canada is sold under control of a two price marketing structure where the Provincial Government sets the price for fluid milk and the Federal Government effectively sets the price for manufacturing milk by means of support programs. The unit price paid by both governments is dependent on whether the fat content of the milk is above or below 3.6 kg per hectolitre. The quantity of milk produced is regulated by allocating to dairy producers the rights to sell a specific amount of milk for fluid use and a separate quantity of milk for manufacturing purposes. In British Columbia approximately 35% of the total is allocated for manufacturing. Milk produced above the allotted amounts is subject to a penalty.

Many dairy operators also participate in a regional or national milk recording and breed improvement program. These programs collect cows production records and type classifications. Production records are projected to 305 days, adjusted for age and expressed in a standardized form which is known as the Breed Class Averages (BCA) in Canada. The BCA are then used to estimate producing abilities, breeding values of cows and to calculate sire proofs. Most animals are bred using artificial insemination (A.I.) with semen from bulls in the regional or national A.I. units. These AI sire proving programs which involve testing large numbers of daughters, account for most of the genetic gain in the population for milk production.

1.2.2: Major Interactions

Like any biologically based system a dairy cattle population involves multi-level and complex interactions. Fig. 1.1 attempts to summarize the major interactions at a macro level.

figure 1.1



The returns per year are determined by the operating costs, the income from milk and the income from animals sold. For this simulation it is assumed that, the driving force behind the dairy industry is the desire of milk producers to maximize their dollar return per year.

Operating costs can be subdivided into two major components: (1) fixed costs such as amortization of land, buildings, quota, animals and equipment; and (2) variable costs such as feed, labour, veterinary services, medication, semen and transportation of milk.

Income from animal sales in commercial herds is derived from animals sold for dairy purposes and the salvage value of culled animals. Cows culled for poor type, low milk yield or for fertility problems are usually sold for beef. Those removed for health reasons usually have some carcass value. Carcasses of cows which die may have a nominal value for dog food. Bull calves are sold at one week old to be raised for veal and can be a significant source of income if a reasonable price is received. Substantially more can be received for bull calves from top cows and bulls if they are deemed good enough to enter a young sire program in an A.I. unit. A substantial income can also be derived from selling animals for dairy purposes although this practice tends to decrease the productivity of the herd.

Both the quantity and quality of the milk produced can affect income. The quantity of milk produced is a product of the number of cows in the herd, the rolling herd average production and the calving interval. The limiting factor is either the size of the facilities, which limits the number of cows, or a quota restriction, which limits the amount of milk which can be sold. The quality or constituents of the milk can influence the unit price received. Usually a higher price is paid for extra percentages of fat.

1.2.2.1: Maximizing Returns

The major improvements in the dollar return within a given market structure are to be made by optimizing production per cow and optimizing calving interval through good management systems and breeding programs.

1.2.2.1.1: Management Systems

Management systems involve the decisions made to improve the animal's environment to take fuller advantage of their biological potential. The major gains are to be made by optimization of the feeding regime, the intensity of management and the culling policy.

The most energy efficient feeding regime for a cow involves feeding to maximize production so that a smaller proportion of the energy is used for maintenance of the cow and more is used to produce milk. However, feeding to maximize the energy efficiency of the cow may result in the use of higher priced feeds and/or in increases in labour and/or overhead costs. Extensive research has been done to find optimum feeding regimes and methods for various environments and market structures.

While labour costs can be minimized, lowering the level of management usually results in a decline in milk production per cow, a decline in the rate of heat detection resulting in a longer calving interval and an increase in the frequency of health problems. All of these factors increase costs and must be weighed against the cost of increased management intensity.

Culling practices have a major effect on the calving interval. Long calving intervals reduce the efficiency of the cow and this must be considered relative to the cost of replacement, expected production ability and the value of offspring.

1.2.2.1.2: Breeding Programs

Breeding programs are designed to increase the average biological potential of the cows in the herd through selection. Much research effort has been directed towards explaining the differences between individuals and developing efficient breeding strategies.

Since genetics has not progressed to the point where the specific effects of most genes in large mammals is understood, descriptive statistics have been employed to describe the animal's biological potential and the environmental effects. The phenotype (P) is made up of the genetic potential, (G) and the environmental effects (E) such that $P = G + E$. The genetic potential can be further subdivided based on how it is inherited. One portion behaves as though it is made up of genes which simply have a positive or negative (additive) effect on a particular trait, while the effects of the genes of the other portion seem to depend at least partially on what other genes are present either at the same locus (dominance or overdominance) or at other loci (epistasis). For production and type traits in dairy cattle most of the genes appear to have additive effects (Lasley, 1972) and the breeding schemes are designed to take advantage of this. The model of interest then becomes:

$$P = A + E$$

Where A is the additive effect and E is the environmental, dominance and epistatic genetic effects. This model becomes useful when it is applied to a population to describe the observed variability. The proportion of the variability that is due to additive genetic effects, the heritability, can then be calculated as a ratio of the variances, that due to additive genetic effects V_g divided by the total phenotypic variance V_p.

$$h^2 = V_g / V_p$$

The heritability is thus a measure of the potential to change a trait through selection. It can be measured by the regression of parent's phenotype on their offsprings or by correlations among sibs. It also provides a means of estimating the breeding value of an individual from its phenotype; $BV = h^2 \times (P - P)$. The estimate can be improved by including the phenotypes of close relatives. The estimated difference of an animal's offspring from the mean of the population is called the estimated transmitting ability (ETA) and is one half of the estimated breeding value (EBV).

1.2.2.1.2.1: Breeding Strategies

A breeding strategy seeks to maximize future income by maximizing genetic progress towards some defined goals. The expected progress per year (G) can be stated as a function of the heritability as (Lasley, 1972):

$$G = [h^2 \times SD_p \times i] / GI$$

where i is a measure of the selection intensity

SD_p is the phenotypic standard deviation of the trait

GI is the generation interval.

Thus the effectiveness of a breeding program depends on the traits selected for, the selection intensity and the generation interval. The goals of the program should depend on the measurable traits that have economic importance and genetic potential to change. If more than one trait is selected for they should be weighted in a selection index taking into consideration the economic value, the genetic variability and the phenotypic and genetic correlations. Under the current price structure the trait of most importance is milk production. As mentioned in section 1.2.2.1: above increasing the production of milk per cow allows the quota to be filled with fewer cows, thus lowering proportionally the maintenance costs. Milk yield is also highly variable and moderately heritable (Lasley, 1972), thus there is considerable potential for improvement.

The price paid for the milk is affected by the percentage of fat (sec. 1.2.1.), so increased milk fat production also results in higher returns which are partially offset by higher feed costs. Fat and protein are both moderately heritable (Wilcox et al., 1978) and have adequate variation for selection to be effective. Under the current market structure increases in the percentage of protein do not affect returns but do increase feed costs and thus in a selection index fat should be weighted positively and protein negatively. This is true despite the fact that fat and protein are positively correlated genetically and environmentally.

Type traits are also moderately heritable and increases in type score do increase the resale (Pearson and Miller, 1981) value of the breeding stock. Thus the importance of type score in a selection index is dependent on the proportion of a herd owners income that is derived from selling breeding stock.

Fertility and conception traits have low heritabilities and selection for these traits would give little improvement (Lasley, 1972).

The selection intensity on cows is low because most heifers are required to replace culled cows. With the extensive use of A.I., bulls are usually capable of inseminating 50,000 or more cows per year and the selection intensity on bulls can be very high (Lasley, 1972) .

The generation interval for dairy cattle is a minimum of three years. For bulls the generation interval is a minimum of six years because the traits of interest are sex limited and in order to identify the superior animals progeny testing must be used. The additional three years are required until their daughters have completed one lactation.

1.2.2.1.2.2: Actual Genetic Improvement

The genetic model has been validated by Freeman (1976) and others who have shown that estimated transmitting abilities are good predictors of daughter performance in small experimental populations. However, many factors contribute to reducing actual genetic progress from that predicted by the model (Van Vleck, 1977). The fact that many dairy operations do not participate in milk recording reduces the effective population size. The reluctance of many to use sires in an organized young sire program reduces the number of bulls that can be tested, effectively reducing the selection intensity. Inappropriate selection indexes, for example the over-weighting of type traits also reduces the selection intensity for traits which maximize dollar returns. Breeding programs which aim at maximizing future income often conflict with management strategies which try to maximize current income. For example methods for discounting future gains to arrive at an appropriate compromise between breeding and management strategies have been investigated (Pearson and Miller, 1981; Lin and Allaire, 1977; Gill and Allaire, 1976; Andrus and McGilliard, 1975).

Taking into account all the factors that can affect genetic gain and estimating actual gain is much more difficult than estimating potential gain. However, with the recent use of the best linear unbiased predictor (BLUP) method of estimating transmitting abilities proposed by Henderson (1966) it is possible to compare the average genetic value of sires over the past few years and measure genetic trends. These estimates confirm that the progress being made is considerably less than what is possible (Van Vleck, 1977).

1.3: Teaching

Despite the fact that one of the primary uses of computer simulation has been as a teaching aid the benefits are not clearly defined in the literature. Perhaps this is because teaching methodology in general is difficult to define (Taylor and Kauffman, 1983). Since simulation is based on a system analysis (presumably of the system being taught) it ensures that the structure of the system is central to the course. Bruner (1960) has effectively argued the value of emphasizing the structure of the subject in education. It makes the subject more comprehensible and more easily remembered and provides a base to which new learning can be added or related to.

Simulation can provide an opportunity for "hands on" or a problem solving type of learning which in many subjects is otherwise not practical or feasible. In surveying the literature on learning methods, Singer & Pease (1976) concluded that where subsequent recall and application are important, these learning methods are generally superior to guided instruction after the basics of the subject have been mastered. "Hands on" or problem solving type of learning will also encourage the students to acquire an inquiring mind (Smythe & Lovatt, 1979), increase students motivation and improve their attitude towards the course (Hocking et al., 1983).

The results of a survey of American and Canadian Agriculture Schools (Taylor and Kauffman, 1983) indicate there is potential for further growth in the use of simulation. The three most commonly mentioned weaknesses of the teaching programs were: (1) the need for more "hands-on" type courses; (2) the need for more exposure to computers; and (3) the need for more original thinking experiences.
Well structured computer simulations can address all of these concerns provided they are: (1) user friendly; (2) based on the system or principles being taught; (3) interactive and problem solving in nature; and (4) not too simplistic and in some way challenge the user to think more deeply.

1.3.1: Teaching Genetics

Computer simulation has been used in teaching animal genetics at least since 1961 when Heidhues and Henderson (1961) reported using computer generated herd records in teaching selection principles. The usages expanded greatly in the 1970's and 80's with the increasing availability of computers. Many programs are currently being used throughout Europe to teach various aspects of animal genetics (Hocking et al., 1983). A beef genetic simulation program (Willham, 1970) used in teaching over 2000 students at Iowa State University since 1969, now used at many other colleges and universities and even used with experienced cattle breeders has met with favorable responses from instructors and students

(Brackelsberg, 1978). A dairy cattle computer breeding simulation program was designed for teaching purposes by McGilliard and Edlund (1979) and has been used at various institutions with a generally favorable response.

Computer simulation programs have been especially useful in teaching animal genetics.

(1) They can be used to carry out experiments that would be too time consuming and costly with real herds.

(2) Simulated data can be generated that behaves according to the principles being taught but without additional unwanted variability that is often present in real data, obscuring the principles being studied.

(3) The true genetic potential of simulated animals can be known, making results clearer and easier to analyze than actual data.

1.3.1.1: Animal Genetics Course

The senior undergraduate animal genetics course at U.B.C., taught by Dr. R.G. Peterson has used computer simulation as a teaching tool for four years. Prerequisites are introductory courses in quantitative genetics and statistics. The chief aim of the course is to give the students a thorough understanding of selection programs and potential genetic gain from various programs, with emphasis on the economics of the program and implications on the total herd management. Five general areas are covered.

(1) Selection index theory and calculation of selection indexes.

(2) Designing breeding programs within given genetic, economic and management constraints.

(3) Methods of estimating potential genetic gain as outlined by Van Vleck (1978), by partitioning into the four sources suggested by Robinson and Rendal (1950): cows to produce cows; cows to produce young sires; bulls to produce cows and bulls to produce young sires.

(4) Practical problems in estimating genetic gain and evaluating biological, economic and management factors which can affect it.

(5) Estimating the heritability of the index from the response to selection and measuring the genetic and phenotypic correlations between index and milk, fat, protein and type.

1.3.1.1.1: Why Dairy Cattle?

Dairy cattle were chosen as the system on which the course was to be based for numerous reasons.

(1) They are the most economically important farm animals in British Columbia and many students will have had or are likely to have first hand experience with them.

(2) They involve a complex system with some interesting features, notably that the traits of interest are sex limited and breeding is done almost exclusively with A.I..

(3) Extensive quantitative information is available from milk recording and breed improvement programs.

(4) Selection has been based on more than one specific trait.

(5) The market structure is quite static, decreasing the need for taking diverse potential genetic goals into account.

(6) The market structure is unique and poses interesting questions.

(7) Many of the principles apply to other species.

1.3.1.1.2: Use of Simulation

In order to achieve the objectives of the course and give a deeper understanding of the subjects covered, a dairy cattle breeding simulation program is used in the following way. The students are asked to design and run a selection experiment with a pair of simulated dairy herds. Students first develop a hypothesis dealing with selection goals, breeding strategies, management practices and farm income. Then calculate an appropriate selection index and run a simulation of his / her model along with fellow students for a simulated fifteen years. In parallel, duplicates of all student herds are run as "controls" to provide a bases for comparison. At the end of the simulation the student is expected to do a full analysis of genetic gain.

(1) Test his / her hypothesis using their own herds, the "controls" and herds of other classmates who had similar or contrasting hypothesis.

(2) Estimate sources of the response to selection.

(3) Estimate heritabilities and correlations.

The Dairy Cattle Breeding Simulation Program of McGilliard and Edlund (1979) and a modified version were used in a similar way for two years. It generated cow herds and an

A.I. unit and for each simulated year it generated 305-day lactation records, dollar values for milk and final type scores and calculated estimated breeding values for each cow. Estimated transmitting abilities were calculated for A.I. bulls and calves were generated each year from specified matings.

The student response to the program was positive, however there were two general criticisms.

(1) Running the program required a fair amount of "busy work" which was of little educational benefit.

(2) Many aspects of the program were too simplistic for a senior undergraduate course.

From the instructor's point of view there were many shortcomings (Peterson, 1980).

(1) Selection indexes were not calculated.

(2) Each lactation was 305-days in length and was started and completed in the same year with the cow either completing a lactation or being open for a year. Thus neither reproduction nor economic parameters could be simulated adequately enough to evaluate the relationships between them and the response to selection.

(3) The control herds required an excessive amount of time to run.

(4) A considerable amount of computer programing was required to get final summaries in a form that allowed students access to data for analysis.

1.3.1.1.3: Objectives of the Simulation

The new dairy cattle breeding simulation program was written primarily to better meet the needs of this course.

(1) The program gives students an opportunity to customize their own selection index and breeding strategy. For example bulls can be selected based on whether they are young or proven, on their pedigree or proof, on their semen price and on their conception rate.

(2) Management options can be selected (sec. 2.2.3.5.1.) for decisions which have an economic effect and alter potential genetic gain.

(3) The simulation can be as realistic as is feasible. Production is simulated on a continuous basis allowing economic returns and reproduction to be included in the simulation (sec. 2.2.2.1. and 2.2.4.). At the end of each year it gives a detailed description of the performance and production of each cow as well as herd summaries (appendix B). Hopefully being exposed to the details, variability and complexity of the system will give the students insights into the problems and conflicts that exist.

(4) Unnecessary work is minimized because it is "user friendly" and provides available automated decision options. The program is interactive and leads the user through with questions, prompts and menus (appendix A1). It also detects unreasonable decisions entered and asks for plausible ones. Tedious decision making that is not essential to the management or breeding program can be

avoided by using automated options based on predefined management practices. For example, instead of having to specify the bull to use on each cow, a number of bulls can be selected to be mated at random to the cows (sec. 2.2.3.5.5.). Hopefully the time spent learning course material will be maximized and student frustration minimized.

(5) Groups of herds to be used as "controls" can be run quickly and at minimum cost. These herds are set up with management and breeding programs in the first simulated year and after this can be run automatically (appendix A). No printed output is generated but their summaries are accumulated to be used for the final analysis.

(6) Output from the simulation provides a comprehensive data base of genetic, production and economic summaries for all herds for all years and a statistical package to simplify the data manipulation (appendix B1). All herd summaries are stored and true breeding values are known so that the genetic gain and it's interaction with economic and management factors can be measured precisely. The statistical routines of the program are designed to detect differences in the rate of change over time or mean differences between herds or groups of herds. Output is both statistical and graphical for any of over a hundred summary variables (table IX). Hopefully this will encourage creative analysis of breeding programs.

(7) Computing costs of a simulation involving less than 100 herds and 20 years should not be excessive. In fact an

effort was made to keep down the costs of running the program and as a result some aspects of the simulation are somewhat simplistic. However, the program is written in a modular form to facilitate easier future expansion (appendices C & D). Most different simulated events are performed in a separate subroutine making it easier to add new events to expand the use of the program to include such things as embryo transplants or feeding regimes.

1.3.1.1.3.1: Intended Course Structure

The program could be useful for various courses. It would be compatible with a one term senior animal genetics course.

(1) Two or more weeks would be required for instruction and discussion on selection methods, management systems and formulation of hypotheses and calculation of selection indexes.

(2) Five or more weeks for the students to run the simulation program.

(3) At least three weeks would be needed for the students to adequately analyze trends and differences between herds. This requires access to a data base with genetic, production and economic summaries for all herds and years, and a tailor made statistical package for analysis.

An objective type midterm examination should be held early in the course to ensure that students know how to calculate a selection index, understand the breeding and management options available and have a worth-while and testable hypothesis. A major portion of the students grade for the course should be on the analysis and explanation of their results.

1.4: Research

Simulation has proved useful at all levels of animal research, at the metabolic pathway level to study energy efficiencies (Canolty and Koong, 1976), at the organ level to study rumen metabolism and digestive function (Baldwin et al., 1970), at the cow level to model animal growth (Brown et al., 1976), at the herd level to study reproduction and breeding management (Rounsaville, 1979; Bailie, 1982), at the population level to optimize sire sampling and progeny testing to maximize genetic gain (Hunt, 1974; Lane, 1973; Oltenacu, 1974). Many other studies are reported in the literature.

Koong & Baldwin (1978) in a summary of the literature have listed potential benefits of simulation in research. They include: (1) unification and summarization of data and concepts leading to increased understanding of a particular system; (2) increased effectiveness in utilization of existing information in the selection of experimental approaches; (3) reduction of conceptual difficulties encountered in analyses of systems too complex for intuitive resolution; and (4) formulation and testing of alternative hypotheses.

1.4.2: Simulation

This simulation has four features which should enhance its value for research.

(1) The simulation covers a broad range with the events simulated at the cow level, the user inputs at the herd level and the effects measured at the herd and population level (sec. 2.1.1.). It integrates the disciplines of dairy cattle management, reproduction, economics and breeding. Hopefully the program will aid researchers in understanding some of the complex interactions in the system.

(2) The option to run automated groups of herds (appendix A & B1) makes it feasible for researchers to ask questions about interactions between various herd management decisions and effects on the herd and population.

(3) The modularized design should facilitate customizing of the program for specific research purposes (appendix C & D).

(4) The program allows preliminary experiments to be carried out and fully analyzed within a few hours and at little cost (appendix A).

Some potential questions that the program could be used to research include: (1) The effects of breeding and management decisions on the sire proving program and conversely the value of proven sires in economic terms to the herd owner; (2) The effects of quota restrictions on genetic gain; (3) The economic merit of a trait under the quota system; and (4) The optimum use of young and proven sires in economic terms for the herd owner and in terms of genetic gain for the population.

Chapter 2. PHILOSOPHY AND DETAILS

Simulation requires evaluating the objectives and the information available to decide on the best level at which to model the system, to define the framework of the simulation and then to analyze the actions and interactions of the components in order to reduce these relationships to mathematical equations.

Biological based systems differ from most other systems in that they have a much greater degree of variability inherent in them because of the multiple levels involved. A typical hierarchy of possible levels might be:

Population
Herd
Animal
Organ
Tissue
Cell
Metabolic pathways

In general "as the level of description of a system is refined, precision is gained, but restraints due to lack of knowledge become more serious. As the level becomes higher, the relationships used between the variables become more empirical, and there is no biological explanation of the functions used to relate them" (Joandet 1975).

The relationships can have either a theoretical, empirical or intuitive basis as defined by Riggs (1963). A theoretical equation is biologically based and can usually give good predictions over a wide range. An empirical equation is one which is derived from fitting experimental data to describe a relationship between two or more variables. It describes the relationship over the range of the experimental data but cannot safely be applied beyond that range and does not explain the reasons for the relationship. Often, however, biological parameters can be found to explain an empirical relationship allowing the model to have broader application. An equation based on intuition can be employed to complete a simulation if no known biological or empirical relationships exist. This may limit but does not invalidate the model, as well the approximations can be improved with time by testing simulation results (Joandet 1975).

2.1: The Simulation

This dairy cattle simulation had to model the system described in section 1.1 and to conform to the requirements for teaching (sec. 1.3.) and research (sec. 1.4.). It had to be able to model the inheritance of the economically important traits and to extend this to include as many as possible of the factors that directly or indirectly affect genetic gain.

For modelling this system from a refined level such as the cell, there are good theoretical models for some biological functions but other functions are not well understood and there are no reliable empirical models. As well the complexity and cost of the simulation would be prohibitive given our current knowledge and computing facilities.

At the cow level there are extensive quantitative data which have given rise to good empirical models and some plausible biological explanations. These empirical models are reliable predictors over the measured ranges. Since the simulation was only intended to be used in those ranges the models do not impose serious restrictions.

2.1.1: Framework

In order to keep the cost of running the program reasonable certain restrictions on the simulation of a population were made.

(1) A maximum of 150 cows per herd.

(2) A maximum of 500 herds in a population.

(3) Only one A.I. unit with up to 500 bulls for a population.

(4) Only four genetically inherited traits, milk, fat and protein production, and overall type score.

(5) No provision is made for buying cows or quota.

(6) Prices of all services and commodities remain fixed for a given simulation.

(7) All herds are assumed to participate in a milk recording and breed improvement program.

Other restrictions due to limitations of specific models are discussed in section 2.2.

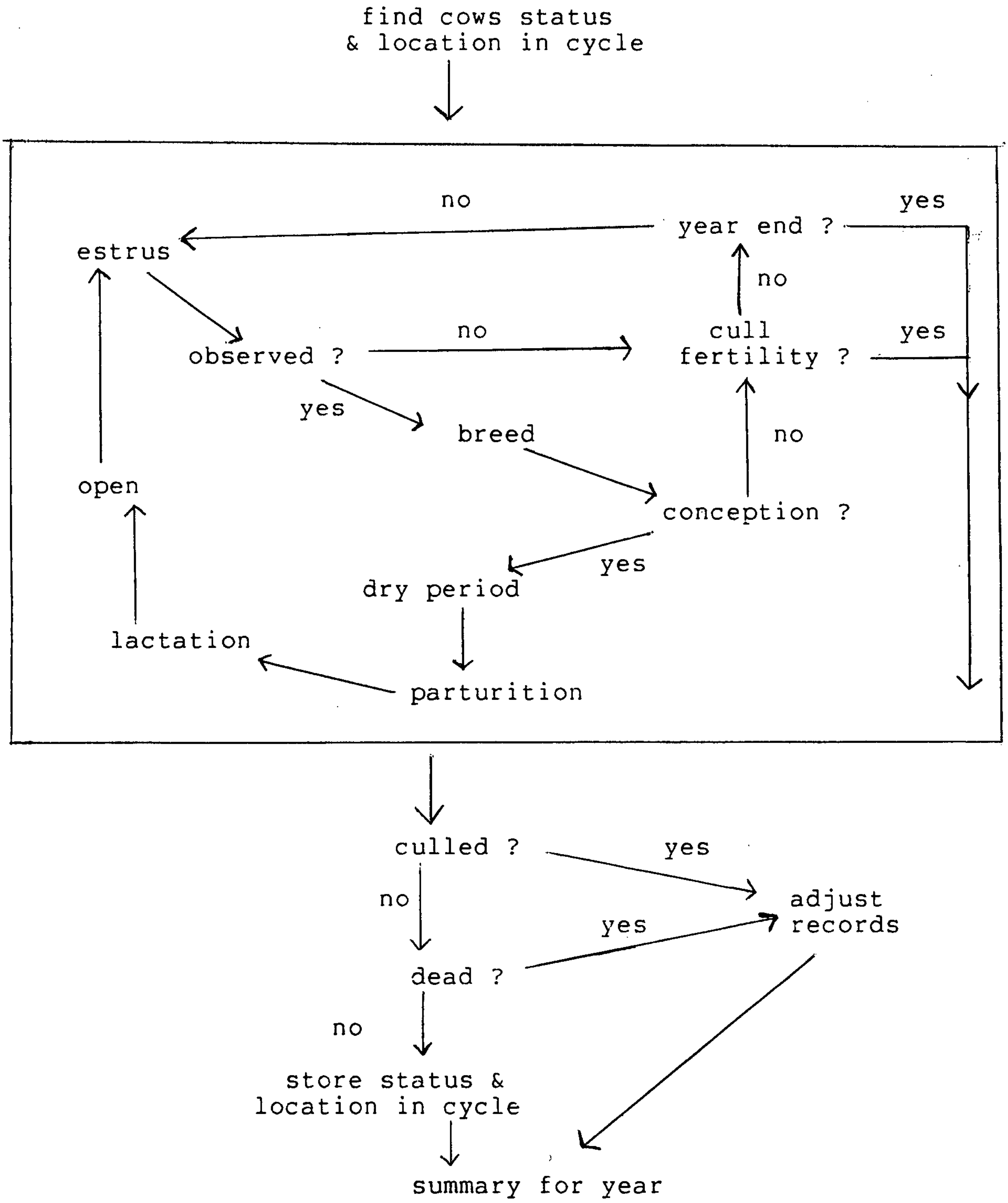
2.1.2: General Description

The program generates herds and an A.I. unit. Operation of the herd is then simulated one calendar year at a time with various options for the degree of management interaction. Bulls used can be selected and mated individually or automatically based on their estimated transmitting ability for the four traits, fertility, semen price and whether they are young or proven sires. Each herd can have a different selection index by which sires are chosen and cows are ranked for breeding and culling, a different level of management intensity and a different criterion for culling cows. Each year cows can be removed from the herd for any of three reasons.

health reasons	-	involuntarily
fertility problems	-	manager sets criteria
poor production or type	-	manager sets criteria and/or specific animals

Until a cow is removed from the herd a continuous cycle of parturition, open period and conception is simulated along with alternating between lactation and dry period, as flow charted in figure 2.1. Each year production from completed and partial lactations, feed costs and estimates of producing and transmitting ability are summarized for each cow along with herd averages, income and expenditures.

figure 2.1


find cows status
& location in cycle
no
yes
year end ?
estrus
no
observed ?
no
cull
fertility ?
yes
yes
breed
no
open
conception ?
yes
dry period
lactation
parturition
culled ?
yes
no
adjust
records
yes
dead ?
no
store status &
location in cycle
summary for year

The A.I. bulls are updated at the end of each year. The semen use, daughter records and good bull calves are inputted and semen prices, updated proofs and new young sires are outputted, ready for the next year.

Detailed records including the true breeding values of all animals in the herd and herd summaries are stored for analysis at the end of a simulation. Complete instructions for using the program and sample runs are given in appendicies A & B.

2.2: Simulated Events

The core of the simulation is the model for 305 day lactation records and yearly type scores. This has been expanded to simulate actual production on a continuous basis which allows models for other biological functions, economic factors and management to fit in.

2.2.1.1: 305-Day Mature Equivalent Production

The simulation uses a linear model (McGilliard and Edlund, 1979) to generate phenotypic values for the four traits, as follows:

$$P = u + G + E_c + E_{ct} + E_h + E_{ht}$$

where:

- P - phenotypic value for a trait for that lactation or year
- u - the population mean for the trait
- G - the true additive genotypic effect (or deviation)
- E_c - permanent cow environmental effect
- E_{ct} - temporary cow environmental effect
- E_h - permanent herd environmental effect
- E_{ht} - temporary herd environmental effect

In order to have the appropriate correlations between the traits, values for all four traits are generated together from the appropriate standard deviations and correlations. The environmental and genetic effects for production traits are generated in kilograms of mature equivalent milk production. With the exception of permanent herd effects they are generated from a "normal" distribution with a mean of zero. It is assumed that the variability is normally distributed since each effect is dependent on a large number of random factors.

Type scores are generated in type units and since cows in practice are only rescored if there is an improvement, if a phenotypic value for type score is generated that is lower than the previous score the type score is left unchanged.

The temporary herd effects are set at the beginning of each herd-year, the permanent cow and genetic effects at birth and the temporary cow effects at the beginning of each lactation. The stochastic component of the permanent herd effects is not included since one objective of the simulation is to compare differences between herds. Provisions are made to allow the permanent herd effects to vary with the management intensity of the herd (sec. 2.2.3.5.1.) by means of four predefined management levels.

Intuitive biological explanations can account for some of the variability of each of the effects, but little progress has been made in linking the effects quantitatively to the observed variability.

- Temporary herd effects are the differences in herds from one year to the next due to short term differences in management such as a new milker.
- Permanent herd effects are differences between herds that remain throughout the simulation. They are due to differences in herd management.
- Temporary cow effects are differences between records of the same cow due to the differences in treatment or chance, such as sickness or accident.
- Permanent cow effects are differences between cows in a herd due to anything from dominant or epistatic genetic effects to fetal environment or sickness which occur at an early age and have a lifelong effect.

The genetic correlations between the different traits are due to pleiotrophic gene action where genes have an effect on more than one trait. Environmental correlations arise from non genetic factors that affect more than one trait.

2.2.1.2: Adjustments

Two factors which are known to be correlated with production are the age of a cow and the degree to which she is inbred. A cow while still growing does not usually produce as much milk as she will once she reaches mature size. In order to estimate breeding values correction factors must be used to predict how much milk she will produce for a 305 day lactation as a mature animal (Lasley, 1972). In terms of the model used (sec. 2.2.1.1) the mean for the temporary cow environmental effects should be negative for young cows. The simulation adjusts the mature equivalent production to actual production by using the inverse of the correction. It allows for correction factors of the three production traits for 2, 3 and 4 year olds.

Inbred cows on average have lower potential production and genetic variability and a higher chance of death or fertility and health problems (McGilliard and Edlund, 1979; Dickerson, 1974). However, extensive checking for inbreeding would have substantially increased the computing costs of running the program, therefore it was treated quite superficially.

For this simulation in the automated mode sire daughter matings are avoided and for specified matings close inbreeding is discouraged. If sire daughter matings occurred the offspring are given reduced permanent cow environmental effects which in this model include dominant and over-dominant gene effects (sec. 2.2.1.). Other detrimental effects could be included (sec. 4.2.) if the intent was to study inbreeding and its interaction with economic returns.

2.2.1.3: Actual Production

Since the simulation is on a continuous basis for production and herd status is outputted once a year, it is necessary to be able to estimate a cow's production between any two points in her lactation. In order to decide when to dry a cow off it is also necessary to estimate her daily production on any given day.

In biological terms a lactation yield is the sum of the daily production which in turn is determined by the phenotypic characteristics of the animal interacting with the daily environment. However, since the simulation is based on a calender year and cow level it would have added little to its value to include daily fluctuations and indeed would have been difficult, given the little information available in the literature. Therefore an empirical relationship at the lactation level that best fits most lactations is suitable.

A rapid rise to a peak followed by a gradual decline is typical of a lactation curve and is best described as a gamma type curve. The function given by Wood (1967) to describe the curve appears to give the best fit of equations developed to date (Rowlands et al 1982). It relates average daily yield (Y) at week "n" to "a" a scale constant and "b" and "c" shape constants.

$$Yn = an^{b}e^{-cn}$$

The peak yield occurs at b/c weeks and is equal to $a(b/c)^{b}e^{-b}$. The persistency of the lactation can be given by $c^{-(b+1)}$ (Wood 1967), or more accurately by the second derivative of the original function and substituting "n" as 25 (Rowlands et al 1982).

In order to utilize this function in the simulation it is necessary to be able to estimate the constants for each lactation such that if an animal lactates for 305 days she produces a previously determined amount of milk and constituents.

Data provided by Keown (1984) which relates peak production to 305 day production showed close to a linear relationships within age groups, but cows tend to have higher peak yield and lower persistency as they go from the first to third lactations. Wood (1968 & 1970) observed significant differences in the shape of the lactation curves due to parity.

Therefore to solve for the lactation curve constants "a", "b", and "c" the simulation made the following assumptions:

(1) The 305 day production is linearly related to the peak production.

(2) That the slope and intercept of the regression in (1) varies only with parity and only until the third lactation.

(3) That the week of the peak in production varies only with parity and only to the third lactation.

Thus the lactation curve constants for a given cow can be solved for using the week of peak production as a constant for a given age and the 305-day production generated as described in section 2.1. The production between any two points in the lactation can be calculated by integration of Woods equation and using techniques of Chi-Leung Lau (1980) and Pike & Hill (1966) for evaluating the incomplete gamma integral (appendix D).

2.2.2: Reproduction

Reproduction involves generating newly simulated animals and modelling the reproductive process. The simulation of the reproductive process was intended primarily to try to include the major factors and interactions affecting calving interval. It could be modeled much more thoroughly than was done here (sec. 4.4.) if interactions between reproduction and genetic gain was to be studied in detail. Rounsaville (1978) and more recently Morant (1985) and Dijkhuizen (1985) have simulated reproduction in more detail. Numerous other papers have appeared on relationships between, reproduction, milk yield and management (Seykora, 1983; Reimers et al., 1985; Hansen et al., 1983) making possible even more comprehensive models. This simulation modeled the following factors without variability.

The gestation length is assumed to be constant and is set for a given population. Although in practice there is variability associated with gestation length it's effect on calving interval is small relative to the variability in estrus detection and conception rates (Fisher et al., 1978; Smith, 1982).

The length of the estrous cycle also has some variability but for this simulation it is assumed to be constant for a population and abnormal cycles are not modeled.

The number of days post partum to first breed a cow is normally a management option. However, matings at less than 50 days increase the chance of an abnormal cycle (Ax, 1982). In the simulation the effects of days postpartum bred on reproduction are not modeled so the minimum days postpartum that a cow can be bred is set as a global parameter (sec. 3.4.2.).

Similarly a short dry period can have an affect on production and reproduction (Goodwill et al., 1984; Dias et al., 1982; Schneider et al., 1981). This also is not modeled so a global restriction is placed on the number of days prior to freshening when a cow must be dried off.

The reproductive simulation models heat detection, conception and the birth of a calf. It begins at the first day a cow is eligible to be bred at which time the day of the estrous cycle that she is in is generated at random. When she reaches estrus the chance of her conceiving is a product of a probability of heat detection and a probability of conception (fig. 2.1). This model assumes that the two probabilities are independent (sec. 4.4).

2.2.2.1: Estrus Detection

The model used to determine the probability of heat detection for a cow was:

$$P = M \, [C_e + Y \, (E_m + G_m)]$$

where:

C_e - cows factor (sec. 2.2.2.1.1)

M - factor depending on the management efficiency (sec. 2.2.3.5.1.)

Y - factor which relates potential milk yield to estrus detection rate

E_m - permanent cow deviation for milk yield

G_m - genetic deviation for milk yield

The largest improvements in reducing the calving interval can usually be made by improving estrus detection (Ax, 1982; Wilcox, 1978). The rate of estrus detection is largely due to the methods and efficiency of the management (Esslemont 1976 and Smith 1982). This simulation allowed for variation in heat detection rate with different management levels. It also allows the probability of estrus detection to vary linearly with potential milk production as Ax (1982) has reported that high producing cows have a higher proportion of silent heats, especially between 60 and 100 days.

2.2.2.2: Conception

The conception rate is determined by the fertility of the cow, the semen of the bull and perhaps some interaction. Since the nature of any interaction has not been well studied the simulation determines the probability of conception (C) simply as a product of a fertility factor associated with the cow (F) and one associated with the bull (Bf).

$$C = F \times Bf$$

The model used for the cow's fertility factor is:

$$F = C_f + (R \times N)$$

where:

R - factor to adjust for number of lactations

N - number lactations for cow

C_f - fertility factor generated at birth (sec. 2.2.2.1.1.)

The simulation allows the cows fertility to change with age as Whitmore (1974) has reported the cow fertility factor decreases for successive lactations.

The bulls fertility factor in the simulation is based only on a mean (u) and a random factor (e) from a gamma distribution.

$$Bf = u + e$$

Genetic contributions to fertility are presumed to be small and were not included in the model.

2.2.2.3: Birth of a Calf

The birth of a new calf is the biological event which makes genetic gain possible. It involves modelling of a number of different factors that are critical to the simulation.

(1) The sex of the animal is generated at random assuming an equal probability of male and female.

(2) True breeding values are generated from the mean breeding values of their parents and a random factor to account for sampling.

(3) If a female further factors are generated.

(a) A permanent environmental factors are generated (see lactation model, sec. 2.2.1.1.).

(b) Fertility and heat detection factors (C) are generated from a population mean (u) and a random factor (e) from a gamma distribution.

$$C = u + e$$

It is assumed that there is a permanent cow variability and that a gamma distribution approximates it.

2.2.3: Feed Consumption

The model for feed consumption does not take into account all the stochastic variability. It is rather a deterministic model, including only the variability that is due to changes in production or the reproductive status of the cow.

It assumes that yearly costs for a cow vary linearly with only five factors.

$$FC = (\text{DAYS DRY} \times MD) + (\text{DAYS LACTATING} \times ML) + (\text{kg CAR.} \times AC) + (\text{kg FAT} \times AF) + (\text{kg PROT.} \times AP)$$

where:

FC = feed costs for a year

MD = daily cost of maintaining a dry cow

ML = daily cost of maintaining a lactating cow

CAR. = carrier (water + minerals + lactose)

AC = additional cost per kg of carrier (assumes milk is 5% lactose and no cost associated with production of water and minerals (Hiller, 1979))

AF = additional cost per kg of fat

AP = additional cost per kg of protein

Additional costs for a high producing cow over a low producing cow are primarily feed costs to supply the energy to produce the extra milk constituents and health costs. The model allows for any costs that vary linearly with the production of milk or milk constituents, days lactating or days dry to be included.

2.2.3.1: Replacement Heifers

The cost of raising replacement heifers was modeled as follows:

$$HC = CF + CY$$

where:

HC = total costs in a calendar year

CF = (days year < 1 year old) x (daily cost for calves)

CY = (days year > 1 year old) x (daily cost for yearlings)

2.2.4: Health and Death

In the simulation it is assumed that the probability of having to cull a cow for health reasons (P_h) is correlated only with age, producing ability and type score.

$$P_h = A\ (N) + Y\ (Gm + Em) + T\ (Gt + Et)$$

where:

A = factor relating health to age

N = number of records

Y = factor relating health to milk producing ability

Gm = genetic deviation for milk

Em = environmental permanent cow deviation

T = factor relating health to type score

Gt = genetic deviation for type

Et = permanent cow environmental deviation for type

Deaths are also assumed to occur at random with the probability (P_d) varying only with age and producing ability.

$$P_d = A\ (N) + Y\ (Gm + Em)$$

where:

A = factor relating probability of death to age

N = number of records

Y = factor relating death to milk producing ability

Gm = genetic deviation for milk

Em = environmental permanent cow deviation

2.2.5: Management

In order for the simulation to be realistic it must provide management options similar to those available to herd operators.

It must also provide the same sort of information that a herd operator uses to base his decisions on. As well as the past performance and current status of the herd this information includes the projection of unfinished lactations, adjustments of lactations for age, estimation of a cows producing ability (EPA) and transmitting ability (ETA) and ETA's for A.I. sires.

2.2.5.1: BCA's

The breed class averages (BCA) being given, adjust lactations for age. They are calculated as a percentage of the population mean 305-day mature equivalent production (305-ME) plus a constant to scale the mean to a level comparable to actual populations.

$$BCA = (M_c / M_u * 100.) + K$$

where:

M_c Cows 305-ME

M_u Population average 305-ME

K Scaling constant for milk, fat, or protein

2.2.5.2: EPA's

Estimated producing ability (EPA) is an estimate of the cow's producing ability relative to herd-mates and was calculated using the method given by McGilliard (1979).

$$EPA = \frac{nr}{1 + (n-1)r} [\text{Avg.} - \text{Herd Avg.}]$$

Where n = number of records averaged

r = repeatability of the trait

2.2.5.3: ETA's

The estimated transmitting ability (ETA) of the cows or half the estimated breeding value (EBV) could be calculated using BLUP techniques but at a considerable increase in the cost of running the simulation. Conversely they could be simulated by adding a random error to the true breeding value. However, traditional methods of estimating transmitting abilities are computationally easy and will allow students the opportunity to verify values by hand.

The simulation calculates ETA's of cows and young animals using the method outlined by (Burnside, 1978) .

$$ETA = 1/2\ [EBV]$$

$$EBV = W\ [\text{Sire's ETA}] + W\ [\text{Dam's ETA}] + \frac{kN}{N + A}\ [\text{Cow's Ave. Dev.}]$$

Where
$$W = \frac{(1 - k)N + A}{N + A}$$

$$A = \frac{(1 - r)}{r - (1 - p)h^2}$$

$$k = \frac{p\ h^2}{r - (1 - p)h^2}$$

N = number of records on the cow

p = 1/2 if sire and dam information available

= 3/4 if only sire or dam information known

= 1 if neither sire nor dam known

h^2 = heritability of trait

r = repeatability of trait

2.2.5.4: Simulated Sire Proofs

Since genetic trends are expected in the simulated populations the sire evaluations require use of a "direct sire comparison" method of estimating transmitting ability. A BLUP analysis is expensive in terms of computer time, so the program uses a "direct sire comparison" based on the true breeding values and simulating the error of estimation. Daughter records are outputted in a format suitable for use by many BLUP programs if this method is desired.

In the simulation the error term to be added to the true breeding values depends on the environmental and genetic variances, the number of daughters and the number of herd-years with daughters.

$$ETA = b(Se + 1/2\ BV)$$

where:

BV = true breeding value

Se = simulated error is a random number from a normal distribution with a mean of zero and a standard deviation of "D".

b = factor which takes into account the heritability, the 1/4 of the genetic variability from the sire and the number of daughters. It regresses the estimate accordingly.

$$b = \frac{n}{n + \frac{4-h^2}{h^2}}$$

where:

h^2 = heritability

n = number of daughters

$$D^2 = 1/4\ V_g\ (1 - b) + 1/n\ (1/2\ V_g + V_{Ep} + V_{Et}) + 1/k\ (V_{Eh})$$

k = number of herds with daughters

V_g = genetic variance

V_{Ep} = permanent cow environmental variance

V_{Et} = temporary cow environmental variance

V_{Eh} = temporary herd environmental variance

The ETA's are calculated each year in which a sire has first lactation daughter records. A sires new ETA is the weighted (by number of daughter records) average of the new and old estimates. As the number of daughters and herds get large the standard error approaches zero and the ETA approaches the true breeding value.

2.2.5.5: Herd Options

The herd options include general management decisions and detailed options for ranking culling and mating animals.

2.2.5.5.1: Management Level

Four levels of management are offered as user options.

These can be set (sec. 3.1.1., 3.2.1.1. and 3.3.1.) to affect any or all of the following herd parameters; the fixed costs, the rate of estrus detection and/or the permanent herd environmental effects.

More intensively managed operations generally have higher operating costs but also higher herd production and more consistent estrus detection.

2.2.5.5.2: Minimum Daily Production

Two levels of minimum daily production are to be set, one for cows that are to be kept in the herd and another for cows to be culled. A pregnant cow will be dryed off earlier than these levels if it is necessary to ensure she has the minimum dry period (sec. 2.2.2.1.2).

2.2.5.5.3: Ranking Criteria

The ranking of cows is used to determine their culling order for low production, to determine how long an open cow is kept and to select top cows for special matings if matings are automated. The rankings are based on a selection index for which the weighting of each trait is chosen. The animal's value for each trait can be based on the current lactation, the estimated producing ability or the estimated transmitting ability.

2.2.5.5.3.1: Culling for Rank to Meet Quota

Low ranked cows are culled each year until production is within a set yearly quota.

If the herd does not use automated decisions an opportunity is given during the running of the program (appendix B1) to change these culling decisions. As well individual young animals can be selected for selling at the beginning of a simulated year.

2.2.5.5.4: Culling for Fertility

Herd policy with regards to how long to try and breed a cow must be set to one of three options.

(1) Set a maximum number of, days open and services for all animals.

(2) Same as above but with a set number of extra days and services for a group of top ranked animals.

(3) Same as (2) but with the extra days and services defined by a function of rank.

2.2.5.5.5: Matings

The options for mating schemes were set such that users could use realistic but general mating schemes or individual matings if desired and time permitted. The herd owner can use young sires, proven sires or both. Three options for selecting matings are offered for both young and proven sires.

(1) Individual matings are selected where the user must input a list of bulls and specify which bull to use on each cow. This method can be used to breed the whole herd but would be tedious. It is primarily intended to be used to make matings for potential A.I. sires.

(2) A list of bulls to use are entered along with the proportion to use each. These are then mated randomly in the proportions specified to all cows not mated individually. Sire daughter matings are avoided.

(3) Bulls are selected automatically. The user sets the minimum fertility, the maximum semen price and the method of selection. Selections can be made randomly or based on the top bulls as ranked by the herds selection index. Special matings to produce A.I. sire prospects can also be automated to breed a specified number of top ranked cows to a specified number of top ranked bulls.

2.2.6: Economic Factors

For this simulation all commodity prices are set at the beginning of a simulation and are constant throughout. (For a special use the prices could be changed between years but it would involve recompiling the program.)

(1) Fixed costs which were assumed to include all costs not otherwise accounted for, can be set to give realistic net earnings. Four management levels with four different fixed cost levels are included associated with different heat detection rates and production levels.

(2) Feed costs which include a cost per kg for carrier, fat, and protein and a price per day for lactating cows, dry cows, yearlings and calves.

(3) Other costs which include a base price for semen and the cost of transporting milk.

(4) Milk price which includes a base price for quota milk and excess milk, fat and protein differentials and the critical level of fat and protein for those differentials.

(5) Prices for culled cows based on the reason for removal, either production or type, fertility, health or death.

(6) Prices for other animals removed including new young sires, other bull calves, heifer calves, year old heifers and two year old heifers.

2.2.6.1: Semen Price

Semen price can effect management breeding decisions. The simulation assumed that the price depends primarily on supply and demand, the amount of semen a bull produces and how much is used and that the price increases exponentially as the available semen approaches zero. The model used is:

$$Ps = B + (U^2 / K)$$

where:

Ps	-	price of semen
B	-	base price
U	-	vials of semen used in a year
K	-	constant dependent on the average semen production and the population size

In order to prevent large fluctuations in price, the price is averaged with the previous year.

Chapter 3. PARAMETERS USED AND EVALUATION OF THEIR ADEQUACY

In order to model a population the program must be initialized with parameters for the modeled events. The simulation program assumes that the instructor or researcher initializing the program to simulate a specific population has an understanding of dairy cattle breeding and management and understands the parameters required. Initialized values that are unrealistic may cause unrealistic results or the program to fail (in contrast with unrealistic decision options made while running the program, sec. 1.3.1.1.3.(4)).

Ideally all parameters should be taken from accurate data on the population to be simulated and should be supported in the literature by values within which the estimated values fall. However, where data are lacking, values from the literature must be used and in cases where literature values are variable or nonexistent intuition must be employed. This chapter provides initialization values for simulating a Fraser Valley dairy cattle population, explains how they were arrived at and discusses their adequacy.

3.1.1: Mature Equivilant 305 Day Production and Type

The population means and the genetic and environmental standard deviations (table I) and covariances (table II) for 305-day production of milk, fat and protein are taken from

British Columbia Dairy Herd Improvement (BCDHI) data compiled in 1980 on 27,000 first lactation records (Peterson, 1980). The population means and the heritabilities (table IV) calculated from the variances all are within the ranges found in the literature (Wilcox et al., 1978).

The BCDHI data did not include type scores so the mean and standard deviations for type score were taken from the program of McGilliard and Edlund (1979).

The values for the four sets of permanent herd environmental effects corresponding to the four management levels were set so that higher levels resulted in slightly higher production (table III). The management choices for this population are assumed to primarily affect the hours of daily labour for the purposes of heat detection (sec 3.5.2.2). Since the effect these changes have on milk production are not well defined a small additional cost was assumed to result in a slight increase in herd production (table III).

3.1.2: Adjustments

To calculate actual 305 day production from mature equivalent production for younger animals the inverse of age correction factors used by the BCDHI in calculating BCA's (table III) was used.

Estimates in the literature of the effects of inbreeding on production are quite variable (Dickerson, 1974). McGilliard and Edlund (1978) suggest a value slightly less than one standard deviation of the permanent cow effects (table III).

table I

. GENERAL DESCRIPTION		PROGRAM VARIABLE NAME	INITIALIZED VALUE .
Population averages		AVG	
305-day yield	- milk		7200.0 kg
	- fat		260.0 kg
	- protein		230.0 kg
	type score		80.0
Standard Deviations			
- genetic		GSD	
	- milk		413.0
	- fat		18.2
	- protein		12.8
	- type		1.66
- permanent cow		PCESD	
	- milk		393.0
	- fat		12.6
	- protein		11.9
	- type		1.64
- temporary cow		TCESD	
	- milk		510.0
	- fat		18.6
	- protein		15.4
	- type		1.56
- temporary herd		THESD	
	- milk		255.0
	- fat		11.2
	- protein		7.7
	- type		1.13

3.1.3: Actual Production

The lactation curve parameters (table IV) have not been estimated in the Fraser Valley population. Although there are some estimates in the literature these parameters could be significantly variable between populations. However, for the current simulation the precise shape of the lactation curve is not critical (sec. 4.3.).

The parameters to define the relationship between peak and 305 day yield are calculated from data by Koong (1984).

table II

GENERAL DESCRIPTION		PROGRAM VARIABLE NAME	INITIALIZED VALUE
Correlations between traits			
environmental		ESIGMA	
	milk - milk		1.0
	milk - fat		0.83
	fat - fat		1.0
	milk - protein		0.96
	fat - protein		0.78
	protein - protein		1.0
	milk - type		0.2
	fat - type		0.2
	protein - type		0.2
	type - type		1.0
genetic		GSIGMA	
	milk - milk		1.0
	milk - fat		0.54
	fat - fat		1.0
	milk - protein		0.70
	fat - protein		0.81
	protein - protein		1.0
	milk - type		0.0
	fat - type		-0.15
	protein - type		0.0
	type - type		1.0

table III

GENERAL DESCRIPTION		PROGRAM VARIABLE NAME	INITIALIZED VALUE
Permanent herd effects for management levels		HELVL	
level 1	milk		150.0
	fat		6.6
	protein		4.5
	type		0.6
level 2	milk		50.0
	fat		2.2
	protein		1.5
	type		0.2
level 3	milk		-50.0
	fat		-2.2
	protein		-1.5
	type		-0.2
level 4	milk		-150.0
	fat		-6.6
	protein		-4.5
	type		-0.6
Age adjustments (fraction of mature equivalent)		AGEAJM	
	< 2 years - milk		0.7195
	< 2 years - fat		0.7154
	< 2 years - protein		0.7195
	2 - 3 years - milk		0.8368
	2 - 3 years - fat		0.8380
	2 - 3 years - protein		0.8368
	3 - 4 years - milk		0.9197
	3 - 4 years - fat		0.9243
	3 - 4 years - protein		0.9197
	> 4 years - milk		1.0
	> 4 years - fat		1.0
	> 4 years - protein		1.0
Penalty for sire daughter matings		PINB	
	milk		103.0
	fat		4.26
	protein		3.2
	type		0.41

table IV

GENERAL DESCRIPTION	PROGRAM VARIABLE NAME	INITIALIZED VALUE
Lactation curve		
peak - 305 day yield		
slopes	BSLP	
lactation 1		4.6395 E-3
lactation 2		4.8904 E-3
lactation 3		4.9499 E-3
intercepts	ACMT	
lactation 1		-2.83
lactation 2		-2.28
lactation 3		-2.7
weeks to peak	WKPK	
lactation 1		10.0
lactation 2		7.5
lactation 3		7.5
Other biological parameters		
gestation length (days)	GEST	283.0
estrous cycle (days)	DCYC	21.0
estrus detection rate		
mean potential	HDCM	0.75
standard deviation	HDCSD	0.01
effect milk	HDMLK	-2.0 E-5
management efficiency	HRDET	
level 1		1.00
level 2		0.85
level 3		0.65
level 4		0.45
fertility cows		
mean	FERTM	0.75
standard deviation	FERSD	0.01
effect records	FEREC	-0.015
death rates		
cows basic	DTRATE	9.0 E-3
effect milk	DTMLK	2.0 E-6
effect records	DTREC	1.2 E-2
yearlings	DTYLG	2.0 E-2
calves	DTCF	4.0 E-2
unborn calves	DTUB	1.0 E-2
health culling rate		
basic	HTL	1.9 E-2
effect milk	HTMLK	2.0 E-6
effect records	HTREC	2.9 E-2
effect type	HTYP	-2.0 E-3

The number of weeks until peak production for the different parities are estimates based on information in the literature. Rowlands and Lucey (1983) reported for a population of British Friesians where mature cows averaged 6300 kg 305-day milk that the number of weeks until peak production averaged 9.8, 6.8 and 7.1 weeks for first, second and third or greater lactations respectively. Earlier data (Wood, 1970) had lower values but the number of weeks until peak production decreased from the second to third lactations. Wood (1980) also showed that higher producing animals generally have later peaks. Therefore since the average production in the Fraser Valley is 7200 kilograms, values slightly higher than those reported have been used (table IV).

3.2.1: Reproduction

The length of gestation and of the estrous cycle (table IV) are average values for Holsteins in the literature. These are adequate for the current simulation (sec. 2.2.1.2.).

3.2.1.1: Visibility of Estrus

The values for estrus detection are mostly based on papers by Esslemont (1976) and Williamson et al (1972). They suggest that estrus detection rates range from 35 percent to 75 percent with the average about 55 percent. The parameters given in table IV for the model (sec. 2.2.2.1.2.1.) result in estrus detection rates that span the suggested range. Due to lack of information available the values for the stochastic variability (HDCSD) and the relation to milk yield (HDMLK) are based only on intuition.

3.2.1.2: Conception

The conception rate is the product of the cows fertility factor (table IV) and the bulls fertility factor (table VII). Smith (1982) suggest that conception rate for a herd can vary from 41 to 61 percent, while the mean conception rate is about 50 percent (Roundsaville, 1978). For this simulated population the conception rate was 52.5 percent for heifers and decreased 2 percent after each lactation.

3.2.2: Health and Death

Death rates (table VI) used Canadian record of performance (R.O.P.) data in 1975 summarized by Westell (1980) as no other data was found. He reported that the percent of cows being removed from the herd for death, sickness, bloat and poison were 0.98, 2.36, 3.71 for cows in their first, second and third or greater lactations respectively.

Removals for health reasons (table VI) for this simulated population were assumed to include those removed for udder breakdown, injury, mastitis, feet and legs, temperament, slow milking and other undefined reasons. In Westell's data the percent of removals for these reasons were 1.85, 4.77 and 7.90 for first, second and third or greater lactations. No other comparable data was found.

3.3: Economic

table V

GENERAL DESCRIPTION	PROGRAM VARIABLE NAME	INITIALIZED VALUE
Economic parameters		
base semen price	MPSEM	15
fixed costs	CFIX	
level 1		39935.
level 2		36015.
level 3		32825.
level 4		30000.
feed costs per:		
kg carrier	FDCAR	2.80 E-2
kg fat	FDFAT	1.2906
kg protein	FDPRO	0.6730
day lactating	FDAY	1.2823
day dry	FDRY	1.6880
day - yearling	FDYLG	1.0434
day - calf	FDCF	0.8211
cost transporting milk	CTRNP	1.34 / hl.
price quota milk	PQMLK	52.34
price excess milk	PEXMLK	38.18
fat differential	PFAT	4.80
critical fat/ hl.	CRFAT	3.60
protein differential	PPROT	0.0
critical protein / hl.	CRPROT	0.0
price for cow culled for:		
production	PCOWP	520.
fertility	PCOWF	520.
health	PCOWH	300.
death	PCOWD	0.0
price for:		
young sire	PYSP	1000.
bull calf	PCFB	50.
heifer calf	PCFH	100.
year old heifer	PCFO	300.
two year old heifer	PYLG	500.

3.3.1: Fixed Costs

Fixed costs (table V) were set to give realistic net incomes for herds and to give appropriate differences in labour costs. The differences in labour suggested by Rounsaville (1978) were used as a guideline. Level 4 assumes no time spent specifically for heat detection. The other levels 3 to 1 assume 1, 2 or 3 half hour or more periods in the day spent observing for estrus behavior. Presumably the periods are chosen from most to least convenient so costs per day of each period were assumed to be $5.00, $6.00 and $8.00. These costs were converted to a yearly basis. An additional $1000 associated with increasing production (sec. 3.3.1.) is added for each increase in level.

3.3.2: Feed and Maintenance Costs

Feed costs for maintenance and production (table V) are based on energy requirements and the price of feeds. The metabolizable energy (ME) required to produce milk constituents have been estimated (Hillers et al., 1979) to be 16.3, 8.5, 6.6 and 0.0 megacalories (Mcal) for fat, protein , lactose and minerals and water respectively. Carrier is assumed to be minerals, water and 5 percent lactose. The ME requirements for maintenance were taken from National Research Council tables (NRC, 1978). The ration was assumed to be 60 % alfalfa hay and 40 % concentrate, with the exception of dry cows being fed only alfalfa. Given the Mcal per kg (Shelford, 1982) and the current prices of alfalfa hay and dairy concentrate and assuming 10% feed wastage, a cost per

Mcal was calculated.

alfalfa hay	2.20 Mcal ME/kg	$16.12/100 kg	$0.07327/Mcal ME
dairy concentrate	2.95 Mcal ME/kg	$20.75/100 kg	$0.07037/Mcal ME
ration x 1.1 (10% waste)			$0.07931/Mcal ME

This was used in calculating the costs of the energy required for maintenance and production.

3.3.2.1: Yearlings and Calves

The model for the costs of rearing heifers (table V) is a simplistic one consisting only of an average daily cost for calves and yearlings. These values are calculated from ME requirements (NRC 1978) and the cost per Mcal of ME for the ration above.

3.3.3: Milk Price

The prices for quota milk, excess milk and fat differential (table V) are the prices in the Fraser Valley as of November 1985. No differential is paid for protein.

Transportation costs are the current costs of shipping milk in dollars per hectoliter

3.3.4: Price for Sold Animals

Prices for animals sold approximate current beef prices in the Fraser Valley.

3.4: Additional Management Information

table VI

. GENERAL DESCRIPTION	PROGRAM VARIABLE NAME	INITIALIZED VALUE .
Management aid parameters		
heritabilities	HERT	
milk		0.26
fat		0.34
protein		0.27
type		0.30
repeatabilities	REP	
milk		0.50
fat		0.51
protein		0.51
type		0.59
minimum days:		
to first mating;	BRDMIN	50.0
dry;	DRYMIN	50.0
to project lactation;	DPROJ	90.0
breed heifers (+730).	DFBRD	-120.0
deviation of base BCA's	IAJBCA	
milk		37
fat		36
protein		35
type		0

3.4.1.1: Projection of Lactation Record

Lactations that are in progress more than 90 days (table VI) are projected to 305 days as is currently done for Fraser Valley herds which participate in the BCDHI program. In the simulation the actual 305 day production records generated previously (sec. 2.2.1.1) are printed out for herds with printed output.

3.4.1.2: Adjustment for BCA Units

The adjustment to give realistic BCA units (table VI) are taken from the BCDHI data (1980).

3.4.1.3: Repeatabilities

The repeatabilities (table VI) used are supported in the literature (Wilcox et al., 1978). They are used in calculating EPA's.

3.4.1.4: Heritabilities

The heritabilities are used in calculating ETA's and were taken from the BCDHI data (1980) and are in agreement with values in the literature (Wilcox et al., 1978).

3.4.2: Population Decisions

Some management parameters are held constant for the population because of limitations in the simulation models (sec. 4.4).

3.4.2.1: Minimum Days Post Partum to Breed

The first day post partum that an animal was eligible to breed (table VI) was based on recommendations in the literature. Traditionally this has been 60 days, however, current recommendations are between 45 and 55 days (Smith, 1982). Smith (1982) also reports that studies using progesterone analysis or television cameras have demonstrated that 85 to 90 percent of cows are cycling normally by 50 days post partum.

3.4.2.2: Minimum Dry Period

The minimum dry period (table VI) is set at a level that does not seriously affect production in the next lactation as this effect was not simulated. The optimum dry period depends on the condition of the animal and can range from 23 to 76 days (Dias and Allaire, 1982). Recommendations for herd policy have been as low as 40 days (Ax, 1982), however, 50 days is more widely accepted.

3.5: Setup

The program to generate the herds and A.I. unit (SETUP Appendix D.1) requires that the initial herd decision options be specified.

table VII

GENERAL DESCRIPTION	PROGRAM VARIABLE NAME	INITIALIZED VALUE
Herd setup parameters		
management decision options	PARM	
herd number (set to zero)		0.0
code (set to zero)		0.0
initial year		-2.0
operating mode. seed		2.2
select criteria. management level		2.2
index weights		
milk		0.5
fat		0.5
protein		0.0
type		0.0
cull fertility		
days open		200.0
services		5.0
# cows to get extra		0.0
exponent rank		0.0
extra days		0.0
extra services		0.0
daily production to dry off		7.0
daily production to cull		12.0
# proven sires . method selection		10.2
max semen price . min fertility		100.2
# young sires . method selection		3.3
% young sires . # bulls-young bulls		30.02
# cows-bulls . method selection		10.2
Quota		1500.
Excess		500.
number herds (always set to 1)		1.0
AI program specific parameters		
bull fertility factor		
mean	BFM	0.7
standard deviation	BFSD	0.1
years before used	UYR	1.0
age removed	DYR	15.0
cows in the population	TCOW	750.0

3.5.1: Decision Options

3.5.1.1: Initial Year

The number of years that are to be run before summaries are saved for analysis is specified by starting the simulation at a negative year. Summaries are not stored until year one.

3.5.1.2: Operating Mode

Two operating modes are provided.

(1) Full output mode allows more specific user options and prints a hard copy of the herds year end status.

(2) Automated mode gives no printed output.

The automated mode is primarily for running "control" herds or for using the program for research. However, when generating the initial population all herds should be set to automated mode and run for two years to allow bulls to get proofs.

The seed to initialize the random number generators can be: (1) user specified; or (2) automatically selected from the time of day clock. The user specified seed is used only to duplicate a run if problems arise.

3.5.1.3: Management Level

The management level can be set at levels 1 to 4 for excellent to poor management respectively. More intensively managed operations usually have higher operating costs (sec. 3.3.1.) but also higher production (sec. 3.1.1.) and more consistent estrus detection (sec. 3.2.2.1.). Generally the herds should be set up with a moderate level of management.

3.5.1.4: Selection Criteria

Selection and culling of animals may be based on one of three criteria: (1) the current lactation; (2) the EPA; or (3) the ETA.

3.5.1.5: Selection Index

The weightings on each trait for the selection index should be set to sum to 1. (During the running of the program values entered are automatically adjusted to sum to 1)

3.5.1.6: Minimum Daily Production

At a certain level of production it is no longer economical to continue milking a cow. This level of daily production is set independently for cows that are pregnant and cows that are to be culled.

3.5.1.7: Culling for Fertility

Three options are provided for determining the maximum days or services a cow is allowed to be open before she is culled.

(1) Set constants for all animals.
(2) Set two levels one for a specified number of top cows and another for the rest.
(3) Allow the maximums to be determined as a function of the cows rank.

Cows are not ranked when the herds are set up and so the days and services should be set to constants initially. They should not be set too low in order for herd sizes to be maintained.

3.5.1.8: Breeding Scheme

The breeding scheme involves setting the number of proven bulls, their maximum semen price and minimum fertility, the number of young sires and the proportion to use them and the number of top cows to mate to a number of top bulls to produce potential young sires. Three methods are provided for selecting the animals in each group.

(1) Individual selection (not in automated herd).

(2) Selection by index rank.

(3) Randomly selection.

When the herds are set up all selections should be made randomly to ensure that all bulls get proofs.

3.5.2: Quota

The quota for all herds in the simulation must also be set when the herds are generated. It is assigned based on the approximate size and average production of the herds to be simulated. For testing only about 30 cow herds were desired.

3.6: A.I. Unit

To update AI bulls a few more parameters need to be defined (table VII).

3.6.1: Bull Fertility

The relative contribution of the bull and the cow to conception is unknown. The important criteria is the product of the two fertility factors, the conception rate (sec 3.2.1.2.) .

3.6.2: Ages

The age at which a bull can first be used as a young sire is usually one year. He is then not used until he obtains a proof. In order to limit the size of an AI unit the bulls can be removed at a certain age.

3.6.3: Number of Cows

This is a rough estimate of the number of cows in the population which in this case is 24 herds times about 30 cows per herd. It is used to weight the bulls relative semen use for calculating the price (2.2.4.1.).

3.6.4: Semen Cost

The base semen price is the minimum cost in the Fraser Valley. The weighting factor depends on the approximate size of the population to be simulated.

Chapter 4. POTENTIAL IMPROVEMENTS

Any simulation of a biological system is never complete because the models used can always be refined to a more complex level. However refinements may not necessarily be better at the present time. Three considerations should be made.

(1) Are the refinements beneficial to the goals of the simulation?

(2) Are the available computing facilities capable of handling the increased complexity?

(3) Is there sufficient information to model the system at a more refined level?

Testing and use of this simulation should reveal many desirable improvements. More research and recording of information in the field will make available more accurate parameters and give a basis for more detailed and reliable models.

This chapter discusses some potential improvements that have been suggested during the development write up and preliminary testing (table VIII and Appendix B1) of the program.

4.1: 305-Day Production and Type Score

The model used for generating 305-day lactation records and type scores (sec. 2.2.1.1.) is an empirical one, the basis of which has been validated by Freeman (1976). In its current form most of the components of the model are stochastic parameters which model accurately but do not explain the variability.

The exception is the model proposed for the permanent herd effects, which allows the permanent herd effects to be related to up to four specific changes in management. This model is an intuitive one but it attempts to explain some of the variability between herds. It could for example be used to compare different types of housing, feeding practices or climatic conditions provided their effects on production and type scores have been measured. Further research is needed to identify in a quantitative fashion the effects of all management practices on economically important traits. This should also provide the basis for more complex and accurate models.

Further research is also needed to find quantitative relationships between other management or biological factors (sec. 2.2.2.) and all of the stochastic parameters in the model to allow a more comprehensive simulation. e.g. Health problems and the temporary cow environmental effects.

The model used for 305-day lactation record and type could be expanded to include other traits such as different type scores if it was desired and estimations of the variability and correlations were available.

4.2: Adjustments

Corrections for age could be more accurate if a linear function that best fits the observed values was used rather than the defined factor for each age (sec 2.2.1.2).

The model for correcting production and type for inbreeding (sec. 2.2.1.2) could be extended to include known effects of inbreeding depression such as increased probabilities of health problems and death. It could also be extended to include all levels of inbreeding. However, extensive checking for inbreeding would result in the program requiring considerably more computer memory and a longer execution time.

4.3: Lactation Curve

The reasons for including the lactation curve in the simulation was to give an estimate of production for lactations that are not 305 days in length and for a portion of a lactation interrupted by a year end. For the current purposes of the simulation the parameters and model used are adequate, improvements would have a small effect on the results (sec. 2.2.1.3).

However, if the simulation was expanded to include season effects or to model management or feeding in more detail the lactation curve should be estimated more precisely.

(1) Parameters relating age and 305 day production to the week of peak production and actual peak production need to be estimated for the specific population being simulated.

(2) Wood (1977) has demonstrated that his equation can also be used to give the production between two points for fat and protein. This would require that the relationships in (1) above be estimated independently for fat and protein.

(3) Many other factors have been shown to affect the shape of the lactation curve such as pregnancy (Bar-Anan, 1981), length of dry period, sickness or injury and feeding. If these factors were to be taken into account they would explain part of the temporary cow variability (sec. 2.2.1.) and the appropriate reduction in the stochastic variability would have to be estimated or adjustments made to the lactation curve.

4.4: Reproduction

The reproductive process was included in the simulation primarily to model the calving interval and its interaction with management and with the level of production. The adequacy of the model (sec. 2.2.2.1.2) can be measured by comparing calving intervals with values reported in the literature. Smith (1982) provided data showing that improving heat detection rates from 35 to 75 percent shortens the calving interval by 19 days on average. These rates of heat detection are comparable to management levels 1 and 4 of the simulation with the parameters supplied (table IV). A preliminary test run of the simulation (table VIII) gave average differences in calving intervals between levels 1 and 4 of 26 days.

The first three culling policies in the test run (table VIII) are the same as those used by Rounsaville (1978) in a reproductive simulation in which he obtained comparable calving intervals. Culling policies 4 to 6 indicate that with intensive culling 365 day calving intervals should be possible. Further use and fine tuning of the fertility and estrus detection parameters used should reveal more fully the adequacy of the models used.

table VIII

TEST RUN MEANS[1]

Calving interval		
- all herds		376.986 days
- management level	1	365.244 days
	2	371.018 days
	3	379.749 days
	4	391.936 days
- culling policies		
	1	389.450 days
	2	386.805 days
	3	382.549 days
	4	363.459 days
	5	369.574 days
	6	370.082 days

[1]Results are from the test run (appendix B1) with 24 herds of 30 cows each, simulated for 7 years using parameters from Chapter three. (Conception rate averaged 0.505 and all four management levels were used.)

Improvements could be made with the current model if more estimates of stochastic parameters linking additional herd management costs to improvements in estrus detection and fertility were available.

A more detailed modelling of the reproductive process could also improve the simulation.

(1) Embryonic losses could be included.

(2) Abnormal cycles together with their probability could be modeled.

(3) The variability of fertility and estrus detection rate within lactations have significant effects and should be modeled if more detailed herd management is included in the simulation.

(4) Breeding by embryo transplant is a possible addition to the reproductive simulation that would be useful for teaching and research.

4.5: Feeding

Feed consumption was included primarily as a basis for calculating a selection index and to give an estimate of costs. For these purposes the models used (sec. 2.2.2.2) are adequate, however, some simple improvements could be made.

(1) A ration which has a roughage to concentrate ratio that varies with the level of production as higher producing cows are normally fed a higher proportion of more expensive concentrates in order to meet their energy requirements.

(2) The energy required for growth of young cows in their dry period should be included.

(3) The daily costs of young animals could be modeled more accurately as a function of age rather than the arbitrary distinction between 1 and 2 year olds.

Expansion of the simulation to allow it to be used in teaching or research of nutrition or feeding management is possible.

(1) An additional subroutine could be added to the simulation to allow least cost ration formulation prior to simulating the year. This would be useful for studying the cost effectiveness of least cost ration formulation.

(2) A more comprehensive simulation could involve modelling the effects of the ration content and feeding practices on the lactation curve and milk yield if appropriate models are available.

(3) The variability in body weight could be modeled as it has a significant relationship with feed consumption (Wilcox et al., 1978). This would allow feed costs to be tied to body weight.

4.6: Health and Death

An omission that would be simple to add is to allow for interactions between management level and the rate of health problems and deaths.

A further expansion of the simulation would be to model the interactions between health problems and the shape of the lactation curve.

4.6: Economics

All prices except semen are modeled as fixed prices (sec. 2.2.4) which is adequate since other market factors affect these prices more than dairy operators decisions. The model for semen prices is only an intuitive one and probably should be improved. Logical expansions of the simulation would be to allow the buying and selling of cows for dairy purposes and the buying and selling of quota. These factors would require a model for the interaction of prices with supply and demand and perhaps other factors.

4.7: Management

The major short coming of the program from a management point of view is the decision of when to cull a cow for fertility. Realistically it should be partially based on whether the herd is over or under quota. The current model (sec. 2.2.3.5.4) allows the decision to vary only with the rank of the cow in the herd. This causes severe problems when using low levels of heat detection and intensive culling for fertility (appendix B). It can result in the herd size becoming severely reduced and quota not being met. One solution is to place limits on the fertility culling specifications allowed. A much more satisfactory solution would be to postpone final culling decisions on open cows in the simulation until herd production is estimated (appendix C), which is when other voluntary culling decisions are made. This would allow the days open to be used in combination with the cows index score in making a culling decision.

For optimum management the days to first breeding and minimum days dry (sec. 2.2.3.5 & 3.4.2) should be set as a function of a cows production and perhaps other factors rather than as a fixed value for all animals. If these decisions were to be included as management options the effects of pregnancy and the length of the dry period on production and the lactation curve would have to be modeled.

Other management decision options that could be added have been covered previously.

(1) Breeding via embryo transplants (sec. 4.4).

(2) Least cost ration formulation (sec. 4.5).

(3) Buy and sell cows (4.7).

(4) Buy and sell quota (4.7).

4.8: Practical

Practical considerations that would make the program easier to use, more flexible or faster to run will become apparent as it is used. Two factors have already arisen.

(1) The calculation of lactation factors could more efficiently be done with breed class averages in integers rather than the current calculations in mature equivalents and real numbers.

(2) If the program could be adapted to run in a reasonable time on a microcomputer the cost of running the program would be greatly reduced. This would facilitate more extensive use of and allow for more expansion of the program.

SUMMARY

A computer program has been developed that is capable of simulating a population of herds of dairy cattle. It can simulate any small population of dairy cattle in Canada with respect to production and type score, reproduction, management decisions and overall economics with summaries at the cow and herd levels on a calendar year basis. It should be useful as a teaching tool for senior undergraduate students in dairy science or animal genetics and as an aid in researching the complex interactions between breeding programs, management systems, biological factors and economics. The program is "user friendly" and allows breeding and management decisions to be automated to reduce unnecessary tedious work. It also is written in a modular form to facilitate modification and expansion.

The primary model for the inheritance of the four traits, milk, fat and protein production and overall type score is an empirical one that has been validated in commercial and research herds over current biological ranges. Accepted theoretical models were used for other components of the system. Due to a lack of documented mathematical models for many of the interactions some models based partly on intuition had to be used to complete the system.

Parameters used are recorded data from the population being simulated or values from the literature where they are

available.

Improvements to the simulation can be made by accurate measurements in the field to give appropriate parameters and more detailed empirical models. Other improvements could be made by expanding the simulation to include other aspects of dairy herd management.

BIBLIOGRAPHY

1: Andrus, D.F. and McGilliard, L.D. 1975. Selection of dairy cattle for overall excellence. J. Dairy Sci. 58: 1876-1879.

2: Ax, R.L. 1982. Shortening the calving interval. Dairy Herd Management. 19(3): 58-60.

3: Bailie, J.H. 1982. The influence of breeding management efficiency on dairy herd performance. Anim. Prod. 34: 315-323.

4: Baldwin, R.L. and Smith N.E. 1971. Intermediary aspects and tissue interactions of ruminant fat metabolism. J. Dairy Sci. 54: 583-589.

5: Brown, J.E., Fitzhugh, H.A. Jr. and Cartwright, T.C. 1976. A comparison of nonlinear models for describing weight age relationships in cattle. J. Anim. Sci. 42: 810-816.

6: Bar-Anan, R. and Genizi, A. 1981. The effects of lactaion, pregnancy and calendar month on milk records. Anim. Prod. 33: 281-290.

7: Black, J.R. and Fox, D.G. 1978. Computer applications in extension education. Symposium on the use of the computer in animal science teaching, research and extention. 1976. A.S.A.S. Illinois. pp. 20-37.

8: Boulding, K.E. 1956. Genèral systems theory. The skeleton of science. Management Sci. 2: 197-203.

9: Brackelsberg, P.O. 1978. The computer in animal science teaching experiences with a beef genetic simulation. Symposium on the use of the computer in animal science teaching, research and extention. A.S.A.S. Illinois. pp. 2-8.

10: Broster, W.H., Broster, V.J., Clements, A.J., Smith, T. 1981. The relationship between yield of milk solids of dairy cows and response to change in plane of nutritions. J. agric. Sci., Camb. 97: 643-647.

11: Bruner, J.S. 1960. The Process of Education, Harvard University Press. pp. 37-40.

12: Canolty, N.J. and Koong, L.J. 1976. Utilization of energy for maintenance and for fat and lean gains by mice selected for rapid post weaning growth rate. J. Nutr. 106: 1202.

13: Chi-Leung Lau. 1980. Algorithim AS 147. A simple series for the incomplete gamma integral. Applied Statistics 29, 113-114.

14: Dhanoa, M.S. 1981. A note on an alternative form of the lactation model of Wood. Anim. Prod. 32, 349-351.

15: Dhanoa, M.S. and Le Du, Y.L.P. 1982. A partial adjustment model to describe the lactation curve of a dairy cow. Anim. Prod. 34: 243-247.

16: Dias, F.M., Allaire, F.R. 1982. Dry period to maximize milk production over two consecutive lactations. J. Dairy Sci. 65: 136-145.

17: Dickerson, G.E. 1974. Inbreeding and heterosis in animals. Proceedings of the Animal Breeding and Genetics Symposium in Honor of J.L. Lush. ASAS and ADSA. Champaign, III.

18: Dijkhuizen, A.A., Hibma, Sj., Renkema, J.A. 1985. A stochastic model for the simulation of mangement decisions in dairy herds, with special reference to reproductive performance. Netherlands Journal of Agricultural Science, 33: 59-61.

19: Ducker, M.J., and Morant, S.V. 1984. Observations on the relationships between the nutrition, milk yield, live weight and reproduction performance of dairy cows. Anim. Prod., 38: 9-14.

20: Esslemont, R.J. 1976. Oestrus behaviour in a herd of dairy cows. Vet. Rec. 99: 472.

21: Fisher, L.J. and Williams, C.J. 1978. Effects of Environmental factors and fetal and maternal genotypes on gestation length and birth weight of Holstein Calves. J. Dairy Sci. 61: 1462-1467.

22: Forrester, J.W. 1968. Principles of Systems. Wright-Allen Press Inc., Cambridge, Mass. pp. 2-24.

23: France, J. and Dhanoa, M.S. 1984. Short Note on estimating lactation yield. J. agric. Sci., Camb. 103, 245-247.

24: Freeman, A.E., Gaunt, S.N., Damon R.A. and Bean B.H. 1976. Heritability and repeatability of fertility of dairy sires. J. Dairy Sci. 59: 1502.

25: Goodwill, R.E., Berger, P.J. and Freeman, A.E. 1984. Effects of previous days open, previous days dry and present days open on milk production in Holsteins. J. Anim. Sci. 59: 68.

26: Gill, G.S. and Allaire, F.R. 1976. Genetic and phenotypic parameters for a profit function and selection method for optimizing profit in dairy cattle. J. Dairy Sci. 59: 1325-1333.

27: Goeke, C.L. and McGilliard, L.D. 1978. Simulated dairy breeding breeding programs. Mimeo. Michigan State University, East Lansing.

28: Golden, H.J., 1977. Mathamatical Modeling of Biological Systems, John Wiley, New York.

29: Grossman, M. and Walter D. 1978. Teaching with interactive compute capabilities (plato: Computer based education for animal breeding). J. Dairy Sci. 61: 1308-1311.

30: Hall, A.D., and Fagen, R.E. 1956. Definition of systems. General Systems 1: 18.

31: Hansen, L.B., Freeman, A.E. and Berger, P.J. 1983. Yield and fertility relationships in dairy cattle. J. Dairy Sci. 66: 293-305.

32: Harvey, W.R. 1969. Procedures used in the generation of swine populations on a computer for teaching selection principles. Mimeo. Ohio State University, Columbus.

33: Henderson, C.R. 1966. A sire evaluation method which accounts for unknown genetic and environmental trends, herd differences, season, age effects, and differential culling. Proc. of Symp. on Estimating Breeding Values of Dairy Sires and Cows. Washington, D.C.

34: Heidhues, T. and Henderson, C.R. 1961. Teaching selection principles with herd records generated by an electonic computer. J. Anim. Sci. 20: 659-667.

35: Hillers, J.K., Young, J.W., Freeman, A.E., and Dommerholt, J. 1979. Effects of milk composition and production on the feed costs of producing milk. J. Dairy Sci. 62: 1662-1664.

36: Hocking, P.M., Foulley, J.L., Petersen, P.H. , Schulte, C. and Zarnecki, A. 1983. Computer programs for teaching animal breeding and genetics. Lives. Prod. Sci. 10: 589-599.

37: Hunt, M.S., Burnside, E.B., Freeman, M.G. and Wilton, J.W. 1974. Genetic gain from sire-sampling and proving programs vary in different A.I. population sizes. J. Dairy Sci., 57: 251-260.

38: Joandet, G.E. and T.C. Cartwright. 1975. Modeling beef Production systems. J. Anim. Sci. 41: 1238-1246.

39: Kennedy, B.W. 1984. Selection limits: Have they been reached with the dairy cow? Can. J. Anim. Sci., 64: 207-215.

40: Keown, J.F., 1984. Peak milk is key to entire lactation yield. Hoards Dairyman, September 10, v. 129(17): 1041.

41: Koong, L.J., Baldwin, R.L., and Ulyatt, M.J. 1978. The application of systems analysis of mathematical modeling techniques to animal science research. Symposium on the use of the computer in animal science teaching, research and extention. A.S.A.S. Illinois. pp. 9-19.

42: Lasley, J.F., 1972. Genetics of Livestock Improvement. Prentice-Hall, Inc. Englewood Cliffs, N.J. pp. 380-410.

43: Lane, W.G., Burnside, E.B., Freeman, M.G., Wilton, J.W. and Driver, H.C. 1973. Economics of progeny testing programs for dairy cattle. J. Dairy Sci., 56: 675-682.

44: Lin, C.Y. and Allaire, R.F. 1977. Relative efficiency of selection methods for profit in dairy cows. J. Dairy Sci. 60: 1970-1978.

45: Mao, I.L. 1978. Teaching methods in amimal breeding: Teaching with self-instructional modules. J. Dairy Sci. 61: 1298-1302.

46: Matsoukas, J. and Fairchild, T.P. 1975. Effects of various factors on reproductive efficiency. J. Dairy Sci. 58: 540-547.

47: McGilliard, M.L. and Edlund, D. 1979. Dairy cattle breeding simulation program. Mimeograph. Virginia Polytechnic and State University, Blacksburg, Virginia, 24061, U.S.A.

48: Morant, S.V. 1985. A stochastic model of the reproductive performance of dairy herds. J. agric. Sci., Camb. 104: 505-512.

49: National Research Council. 1978. Nutrient requirements of dairy cattle. 5th rev. ed. Nat. Res. Council, Washington D.C.

50: Oltenacu, P.A. and Young, C.W. 1974. Genetic and financial considerations of progeny testing programs in an A.I. dairy cattle population. J. Dairy Sci., 57: 1245.

51: Pearson, R.E., and Miller, R.H. 1981. Our Industry Today: Economic definition of total performance, breeding goals, and breeding values for dairy cattle. J. Dairy Sci. 64: 857-869.

52: Peterson, R.G. 1980. British Columbia Dairy Herd Improvement data. Personal communication.

53: Pike, M.C. and Hill, I.D. 1966. Algorithm 291. Logarithm of gamma function. Comunications of the association for Computing machinery. 9: 684.

54: Pirchner, F. 1978. Teaching animal breeding in Europe. Symposium: Teaching animal breeding. J. Dairy Sci. 61: 1292-1297.

55: Reimers, T.J., Smith, R.D., Newman, S.K. 1985. Management factors affecting reproductive performance of dairy cows in the Northeastern United States. J. Dairy Sci. 68: 963-972.

56: Rounsavile, T.R., Oltenacu, P.A., Milligan, R.A., and Foote, R.H. 1979. Effects of heat detection, conception rate, and culling policy on reproductive performance in dairy herds. J. Dairy Sci. 62: 1435-1442.

57: Robertson, A. and Rendal, J.M. 1950. The use of progeny testing with artificial insemination in dairy cattle. J. Genetics 50: 21-30.

58: Rowlands, G.J., Lucey S. and Russell A.M. 1982. A comparison of different models of the lactation curve in dairy cattle. Anim. Prod. 35: 135-144.

59: Schneider, F., Shelford, J.A., Peterson, R.G. and Fisher, L.J. 1981. Effects of early and late breeding of dairy cows on reproduction and production in current and subsequent lactation. J. Dairy Sci. 64: 1996-2002.

60: Seykora, A.J., and McDaniel, B.T. 1983. Heritabilities and correlations of lactation yields and fertility for holsteins. J. Dairy Sci., 66: 1486-1493.

61: Shelford, J.A. 1982. Energy content of local feedstuffs. Personal communication.

62: Singer, R.N. and Pease, D. 1976. A comparison of discovery learning and guided instructional strategies on motor skill learning, retention and transfere, Research Quarterly. 47: 788-796.

63: Smith, R.D. 1982. Catching the cycling cow. Dairy Herd Management. 19(3): 8-11.

64: Smythe, R. and Lovatt, K.F. 1979. Application of the computer in biology teaching: computer assisted and computer managed learning. J. of Biol. Educ., 13: 207-220.

65: Taylor, R.E., and Kauffman, R.G. 1983. Teaching animal science: Changes and challenges. J. Anim. Sci., Suppl. 2, 57: 171-196.

66: Thomas, R. and William, R.L. 1976. The effects of model fidelity and competition in an animal selection simulation on professional breeder's attitude toward the simulation. Iowa State J. of Res., 50: 363-370.

67: Van Vleck, L.D. 1977. Theoretical and actual genetic progress in dairy cattle. Proc. Int. Conf. Quant. Genet. 16-21 Aug. 1976. Iowa State University, Ames, Iowa. pp. 543-567.

68: Vesley, J.A., Mcallister, A.J., Lee, A.J., Batra, T.R., Darisse, J.F.P., Roy, G.L., and Winter, K.A. 1983. Evaluation of cow reproduction in the pureline foundation phase of the Canadian National Dairy Cattle Breeding Project. J. Dairy Sci., 66: 867-873.

69: Westell, R. 1980. Dairy herd disposal reasons. MSc. thesis, University of Guelph.

70: White, J.M., McGilliard, M.L. and Vinson, W.E. 1978. Teaching with computer simulated herds. J. Dairy Sci. 61: 1314-1317.

71: Whitmore, J.L., Tyler, W.J., Cassida, L.E. 1974. Effects of early postpartum breeding in dairy cattle. J. Anim. Sci. 38: 2.

72: Wilcox, C.J., Van Horn, H.H., Harris, B., Jr., Head, H.H., Marshall, S.P., Thatcher, W.W., Webb and D.W., Wing, J.M. 1978. Large Dairy Herd Management. University Presses of Florida, Gainesville. pp. 13-51.

73: Willham, R.L. 1970. Beef Genetic Simulation Program. Copyright Iowa State University, Ames.

74: Williamson, N.B., Morris, R.S., Blood, D.C. and Cannon, C.M. 1972. A study of oestrus behavior and oestrus detection methods in a large commercial dairy herd. R1. The relative efficiency of oestrus detection. Vet. Rec. 91: 50.

75: Wood, P.D.P. 1967. Algabraic model of the lactation curve in Cattle. Nature, London. 216: 164-165.

76: Wood, P.D.P. 1968. Factors affecting percistency of lactation in cattle. Nature, London. 218: 894.

77: Wood, P.D.P. 1969. Factors affecting the shape of the lactation curve in cattle. Anim. Prod. 11: 307-316.

78: Wood, P.D.P. 1970. A note on repeatability of parameters of the lactation curve in cattle. Anim. Prod. 12: 535-538.

79: Wood, P.D.P. 1976. Algebraic models of the lactation curves for milk, fat, and protein production with estimates of seasonal variation. Anim. Prod. 22: 35-40.

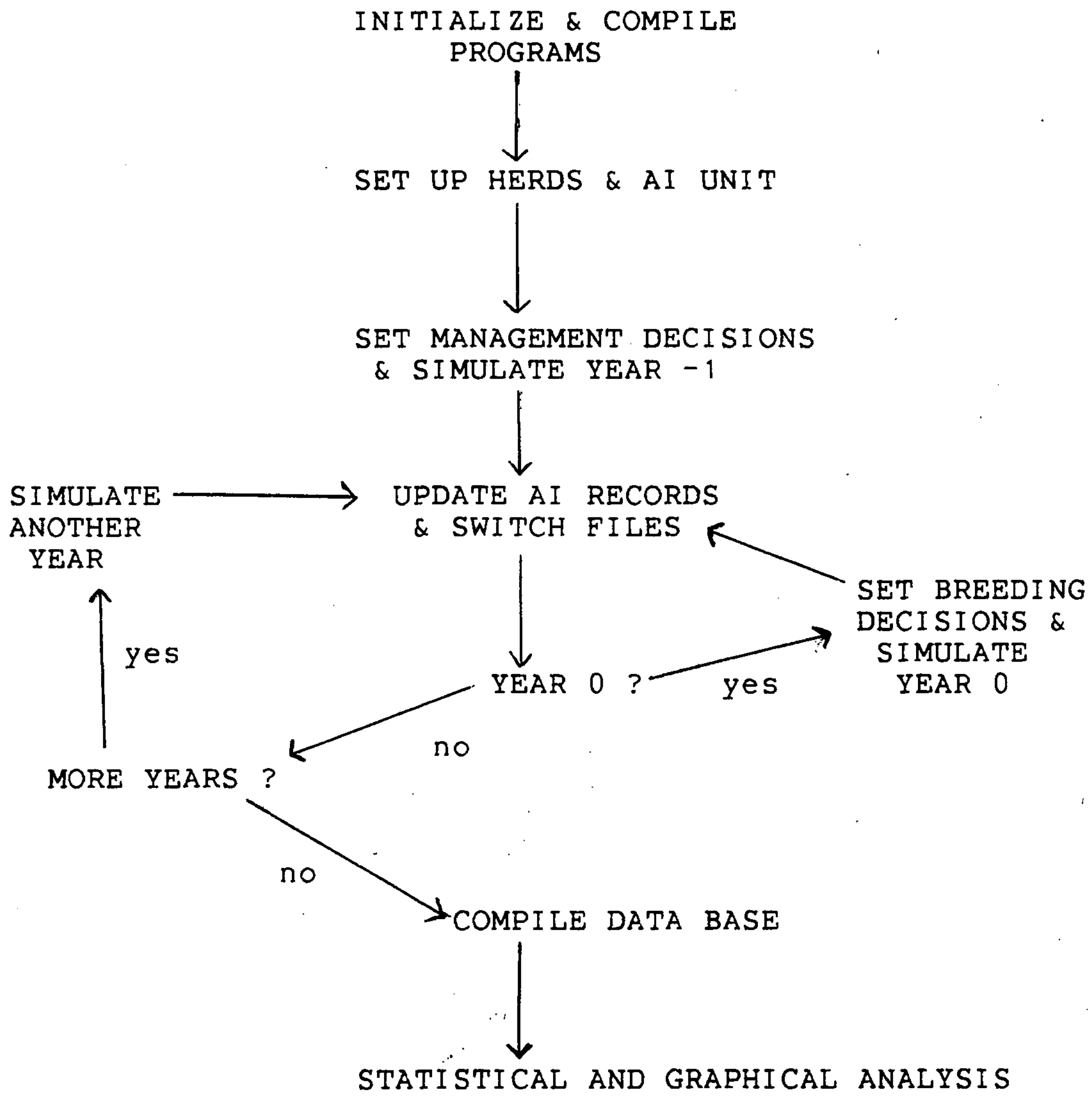
80: Wood, P.D.P. 1977. The biometry of lactation. J. agric. Sci. Camb. 88: 333-339.

81: Wood, P.D.P. 1980. A note on the lactation curves of some high-yielding British Friesian cows. Anim. Prod. 30: 299-302.

Appendix A. FULL INSTRUCTIONS

This appendix gives complete instructions for running a simulation for teaching or research. The basic steps are outlined in figure 5.1 below.

Figure 5.1



All the programs in this package are designed to be run from a terminal by entering "RUN PROGRAM NAME". The user is prompted for all input decisions in one of three formats.

(1) A list of numbered options followed by "OPTION ?" where the user is to input the number of the option desired.

(2) A single value asked for by "?" where the user is to input one value.

(3) A number of values are asked for by "ENTER" where the user is to input more than one value in a line.

NOTE : The programs accept input in "semi-free format" so that each value entered MUST be followed by a comma.

For complete sample input and output see appendix B1.

1: INITIALIZE AND COMPILE ALL PROGRAMS

The simulation package includes six programs.

SETUP

SIM.YEAR

AI.UPD

AIVIEW

CR.DBASE

STAT.ANAL

(see Appendix C for flow charts and Appendix D for full listings)

1.1: Initialize all programs

Parameters that describe the population to be simulated (chapter 3) should be used to initialize all programs. The variable names (tables I - VII) can be found in the REAL or

INTEGER statements in the beginning of each program (Appendix D). The appropriate values should be entered between the slashes after the variable name.

1.2: Compile all programs

The programs should be compiled using the Fortran compiler with the highest optimization level.

2: SET UP HERDS AND AI UNIT

2.1: Create files

A.I.S - for updated AI bull records

A.I.P - for year old AI records

DREC - first lactation records

BUSE - bull use and conception records

YSIRE - potential young sires

SUMS - to store yearly summaries

CODES - to store herd codes

CHECK.RUN - to check if herds have been run

G*.SAV G*.IN G*.OUT - automated groups of herds status

* = 100, 200, 300, or 400

H*.SAV H*.IN H*.OUT - student herds * = 1 - 100

* .SAV - past year status

* .IN - current year status

* .OUT - end of year status

H*-* - Batch file for printing students full outputs

2.2: Run SETUP

This program prompts the user for the herd size, AI unit size, number of herds and the number of students (Appendix

B1). It generates all cows and AI bulls and fills the files A.I.S, G*.IN, H*.IN and CODES.

3: SIMULATE A YEAR

This only involves running the program SIM.YEAR (appendix B1)

The program reads the herd status from the herd .IN file and the AI bulls from A.I.S . It outputs first lactation records to DREC, bull use and conceptions to BUSE, selected bull calves to YSIRE, year end summaries to SUMS and herd status to the herd .OUT file. If desired it also prints a detailed year end summary for the herd and full year performance for all cows (appendix B2).

3.1: Initial years

For two preparatory years "-1" and "0" (can set more, table VII) both student and group herds are run in a semi-automated (option 1 - NO printed copy) mode to allow basic breeding and management options to be set and to allow students to become familiar with the program.

(1) The user is prompted for a herd number and code which ensures against accidental running of the wrong herd. Student herds are numbered 1-100 and group herds 101, 201, 301 and 401. Codes can be found in the file CODES.

(2) The main option menu is displayed. For year "-1" only options 2-5 should be changed. When option 7 (CONTINUE) is entered the program executes with no further menus or prompts.

(3) Before running year "0" management and breeding policies should be decided upon and major decision options set. Student herds should set option 1 to "output printed copy".

3.2: Later years

Summaries of all herds for years 1 and on are saved for the final analysis.

(1) Control herds should be run in the fully automated mode. When prompted for herd and code, the herd and code are entered followed by a "T". All menus are then bypassed and no printed output is generated.

(2) Student herds should be run with the full output option (see appendix B1 for a sample run).

(a) The main menu is displayed to allow options to be changed or fine tuned.

(b) A sub-menu is displayed to allow culling or mating of specific animals.

(c) Cows to be culled for low index score are listed on the screen to allow changes to be made before the year ends.

(d) Herd summaries for the year are immediately listed on the screen for inspection.

(e) The details of the herds performance for the year and the herd summaries are outputted to the H*-* batch file for printing (appendix B2).

4: <u>UPDATE AI AND SWITCH FILES</u>

After all herds have completed a year the AI file must be updated with the information on the bull use and his dauaghters first lactation records and the new young bulls. The updated AI bulls can then be looked at using the program AIVIEW. Herd files must also be switched before running the next year.

4.1: AI UPDATE

(1) Files DREC and BUSE must be sorted by bull number and the current AI file (A.I.S) must be copied to the old AI file (A.I.P). These operations can be simplified by using a batch file.

(2) Run the program AI.UPD . This program prompts the user for the number of new young sires to select and an index by which to choose them (appendix B1). The program automatically reads in the potential young sires from all herds (YSIRE), the new first lactation records (DREC), the number of vials of semen used and the number of conceptions for each bull (BUSE). It then calculates new estimates of transmitting ability and fertility and a new semen price for each bull and adds new young bulls. The updated AI bulls are then written to the A.I.S file for the next year.

4.2: AIVIEW

Students can look at the bulls available to prepare for the simulation of the next year by running the program AIVIEW. This program prompts the user for their own selection index (appendix B1). It then calculates the index score for all sires and lists them on the screen in order of index score. It also includes their ETA's for the four traits and the number of daughters on which it is based, their sire and maternal grandsire, their estimated conception rate and their semen price.

4.3: SWITCH FILES

The herd files must be switched before simulating the next year. A "BATCH" file can be set up to copy **.IN files to temporary save files (**.SAV) and **.OUT files to **.IN files and then to empty **.OUT files. The files YSIRE, DREC and BUSE should also be emptied.

5: COMPILE DATA BASE

The program CR.DBASE copies the herd-year summaries in SUMS and the list of the information stored to new files which can be accessed quickly for analysis.

(1) Before compiling the program check that the program variable TSUM is dimensioned with the number of years and number of herds in the simulation run.

(2) Create files SUMMARIES and VARIABLES to receive the new information and run CR.DBASE .

6: ANALYSIS

Run the program STAT.ANAL . This interactive program does the final data summary and statistical analysis. It is designed to allow the user to group herds as treatment groups and test for differences in the means or the changes over time (slopes) for the summary variables (table IX). It does an analysis of covariance with years as the covariate, tests for differences in means or slopes within groups (test if herds should be grouped together) and between groups and does a Student Knewman Keuls test (SNK) where significant differences in slopes or means are found.

The program also allows limited arithmetic operations to be performed on the on the summary variables (table IV) for more extensive analysis.

The program reads variable names and numbers from the file VARIABLES and data from SUMMARIES. Titles and specifications are entered interactively. The statistical analysis is written to a specified herds file and all data used is written to a file -DAT.

(2) Further analysis can be done using the data in -DAT with other statistical or graphics programs. These analysis can be done more economically by first down loading -DAT to a microcomputer.

Table IX

1. NO. CMPLT. LACT. BEF. CULL
2. BC HERD AVG. MILK (KG)
3. BC HERD AVG. FAT (KG)
4. BC HERD AVG. PROTEIN (KG)
5. BC AVG. BCA MILK
6. BC AVG. BCA FAT
7. BC AVG. BCA PROTEIN
8. BC AVG. TYPE SCORE
9. BC AVG. AGE
10. BC AVG. DAYS MILKED
11. BC AVG. % FAT
12. BC AVG. % PROTEIN
13. NO. CMPLT. LACT. AFTER CULL
14. AC HERD AVG. MILK(KG)
15. AC HERD AVG. FAT (KG)
16. AC HERD AVG. PROTEIN (KG)
17. AC AVG. BCA MILK
18. AC AVG. BCA FAT
19. AC AVG. BCA PROTIEN
20. AC AVG. BCA TYPE SCORE
21. AC AVG. AGE
22. AC AVG. DAYS MILKED
23. AC AVG. % FAT
24. AC AVG. % PROTEIN
25. CALVING INTERVAL
26. CONCEPTION RATE
27. TOTAL HECTOLITRES MILK
28. FAT TEST
29. PROTEIN TEST
30. TOTAL QUOTA MILK (HL)
31. PRICE QUOTA MILK ($/HL)
32. PAID QUOTA MILK ($)
33. TOTAL EXCESS MILK (HL)
34. PRICE EXCESS MILK ($/HL)
35. PAID EXCESS MILK ($)
36. TOTAL SURPLUS MILK (HL)
37. FIXED COSTS
38. FEED COST FOR COWS
39. FEED COST FOR YEARLINGS
40. FEED COST FOR CALVES

55. NO. LIVE COWS
56. LIVE COWS B.V MILK
57. LIVE COWS B.V. FAT
58. LIVE COWS B.V. PROTEIN
59. LIVE COWS B.V. TYPE
60. NO. BRED HEIFERS
61. BRED HEIF. B.V. MILK
62. BRED HEIF. B.V. FAT
63. BRED HEIF. B.V. PROTEIN
64. BRED HEIF. B.V. TYPE
65. NO. FERTILITY CULLS
66. FERT. CULL B.V. MILK
67. FERT. CULL B.V. FAT
68. FERT. CULL B.V. PROTEIN
69. FERT. CULL B.V. TYPE
70. NO. CULLED BY LAST LACT.
71. LACT. CULL B.V. MILK
72. LACT. CULL B.V. FAT
73. LACT. CULL B.V. PROTEIN
74. LACT. CULL B.V. TYPE
75. NO. CULLED BY EPA
76. EPA CULL B.V. MILK
77. EPA CULL B.V. FAT
78. EPA CULL B.V. PROTEIN
79. EPA CULL B.V. TYPE
80. NO. CULLED BY ETA
81. ETA CULL B.V. MILK
82. ETA CULL B.V. FAT
83. ETA CULL B.V. PROTEIN
84. ETA CULL B.V. TYPE
85. NO. CULLED FOR HEALTH
86. HTH. CULL B.V. MILK
87. HTH. CULL B.V. FAT
88. HTH. CULL B.V. PROTEIN
89. HTH. CULL B.V. TYPE
90. NO. DEAD COWS
91. DEAD COWS B.V. MILK
92. DEAD COWS B.V. FAT
93. DEAD COWS B.V. PROTEIN
94. DEAD COWS B.V. TYPE

109. DEAD YLGS. B.V. TYPE
110. NO. LIVE HEIFER CALVES
111. LIVE CALVES B.V. MILK
112. LIVE CALVES B.V. FAT
113. LIVE CALVES B.V. PROTEIN
114. LIVE CALVES B.V. TYPE
115. NO. SOLD HEIFER CALVES
116. SOLD H. CF. B.V. MILK
117. SOLD H. CF. B.V. FAT
118. SOLD H. CF. B.V. PROTEIN
119. SOLD H. CF. B.V. TYPE
120. NO. SOLD BULL CALVES
121. SOLD B. CF. B.V. MILK
122. SOLD B. CF. B.V. FAT
123. SOLD B. CF. B.V. PROTEIN
124. SOLD B. CF. B.V. TYPE
125. NO. DEAD CALVES
126. DEAD CALVES B.V. MILK
127. DEAD CALVES B.V. FAT
128. DEAD CALVES B.V. PROTEIN
129. DEAD CALVES B.V. TYPE
130. NO. A.I. PROSPECTS
131. PROSP. A.I. B.V. MILK
132. PROSP. A.I. B.V. FAT
133. PROSP. A.I. B.V. PROTEIN
134. PROSP. A.I. B.V. TYPE
135. PROVEN SIRES B.V. MILK
136. PROVEN SIRES B.V. FAT
137. PROVEN SIRES B.V. PROTEIN
138. PROVEN SIRES B.V. TYPE
139. PROVEN SIRES NO. VIALS USED
140. PROVEN SIRES NO. CONCEPTIONS
141. PROVEN SIRES FERTILITY
142. PROVEN SIRES SEMEN PRICE
143. UNPROVEN SIRES B.V. MILK
144. UNPROVEN SIRES B.V. FAT
145. UNPROVEN SIRES B.V. PROTEIN
146. UNPROVEN SIRES B.V. TYPE
147. UNPROVEN SIRES NO. VIALS
148. UNPROVEN SIRES NO. CONCEPT.

Table IX continued

41. SEMEN COSTS
42. COST SHIPPING MILK
43. TOTAL EXPENSES
44. SOLD - PRODUCTION ($)
45. SOLD - FERTILITY ($)
46. SOLD - HEALTH ($)
47. SOLD - DEAD COWS ($)
48. SOLD - YEARLINGS ($)
49. SOLD - 1 YEAR CALVES ($)
50. SOLD - HEIFER CALVES ($)
51. SOLD - BULL CALVES ($)
52. SOLD - A.I. STUD ($)
53. TOTAL INCOME
54. NET INCOME
95. NO. LIVE YEARLINGS
96. LIVE YLGS. B.V. MILK
97. LIVE YLGS. B.V. FAT
98. LIVE YLGS. B.V. PROTEIN
99. LIVE YLGS. B.V. TYPE
100. NO. SOLD YEARLINGS
101. SOLD YLGS. B.V. MILK
102. SOLD YLGS. B.V. FAT
103. SOLD YLGS. B.V. PROTEIN
104. SOLD YLGS. B.V. TYPE
105. NO. DEAD YEARLINGS
106. DEAD YLGS. B.V. MILK
107. DEAD YLGS. B.V. FAT
108. DEAD YLGS. B.V. PROTEIN
149. UNPROVEN SIRES FERTILITY
150. UNPROVEN SIRES SEMEN PRICE
151. YOUNG SIRES B.V. MILK
152. YOUNG SIRES B.V. FAT
153. YOUNG SIRES B.V. PROTEIN
154. YOUNG SIRES B.V. TYPE
155. YOUNG SIRES NO. VIALS
156. YOUNG SIRES NO. CONCEPTIONS
157. YOUNG SIRES FERTILITY
158. YOUNG SIRES SEMEN PRICE
159. SELECTION INDEX WT. MILK
160. SELECTION INDEX WT. FAT
161. SELECTION INDEX WT. PROTEIN
162. SELECTION INDEX WT. TYPE

Appendix B1 SAMPLE RUN

This appendix gives the input and output to the screen for sample interactive runs for all programs.

Program output - CAPITAL LETTERS

User input - **BOLDFACE**

Annotations - *italics*

1: PROGRAM "SETUP"

```
#RUN SETUP
#Execution begins
   ENTER THE NUMBER OF STUDENTS AND THE NUMBER OF HERDS EACH
   (MAXIMUM 100 STUDENT HERDS)
   THE NUMBER OF CONTROL GROUPS (MAXIMUM 4),
   THE NUMBER OF COWS PER HERD (MAXIMUM 150)
   THE NUMBER OF YOUNG BULLS TO ADD EACH YEAR (AND AN INTEGER SEED
                                                   - OPTIONAL)
 12,1,1,30,3,
 ENTER THE NUMBER OF COWS OF EACH AGE 2-7
 9,7,5,4,3,2,
        12 STUDENTS      1 HERDS/STUDENT    30 COWS/HERD
         1 REPLICATE GROUPS      3 BULLS       929 SEED

 AGE DISTRIBUTION
       2        3        4        5        6        7

       9        7        5        4        3        2
 ENTER "T" IF ERROR OR RETURN IF OK
 RETURN
#Execution terminated
```

The herds and AI unit are now setup ready to run the simulation

2: PROGRAM "SIM.YEAR" FULL OUTPUT MODE

```
#RUN SIM.YEAR
#Execution begins
                        DAIRY CATTLE BREEDING SIMULATION
                        --------------------------------


  YOU WILL BE PROMPTED FOR DECISIONS

  - ENTER  ALL VALUES ON ONE LINE (EXCEPT WHEN ENTERING ANIMALS)
  - WHEN ENTERING ANIMALS START A NEW LINE FOR EACH ANIMAL
  - EACH VALUE ENTERED MUST BE FOLLOWED BY A COMMA !!

HERD NUMBER AND CODE ?
11,500,

      HERD  11     SIMULATION OF YEAR   9

     MANAGEMENT DECISIONS SELECTED          MAIN MENU

  1   OUTPUT PRINTED COPY OF HERDS ? NO
  2   MANAGEMENT LEVEL OF  1
        RANKING AND CULLING BASED ON:  ETA
  3   WEIGHTINGS FOR THE SELECTION INDEX :
        MILK   0.50   FAT   0.50   PROTEIN   0.0    TYPE   0.0
  4   NUMBER OF DAYS OPEN OR SERVICES :
        130. DAYS OR  4. SERVICES WITH EXTRA FOR THE TOP 12. COWS
        BASED ON AN EXPONENT  1.000  AN ADJUSTMENT FOR DAYS  10.00
        AND AN ADJUSTMENT FOR SERVICES  0.300
  5   MINIMUM DAILY MILK PRODUCTION:
        AT  7.0 KG/DAY A COW IS DRIED OFF
        AT 12.0 KG/DAY A CULL COW IS SOLD
  6   MATINGS-   70% TO   5 PROVEN SIRES SELECTED BY RANK
         30% TO   3 YOUNG SIRES SELECTED RANDOMLY
        SPECIAL MATE  10  COWS SELECTED BY RANK
        MAXIMUM SEMEN PRICE $  500/VIAL    MINIMUM CONCEPTION  20%
  7   CONTINUE
OPTION ?

   ENTERING  1-6 PRODUCES SUB-MENUS TO ALLOW OPTIONS TO BE CHANGED


1,                        option 1 sub-menu

  THE OUTPUT TO BE
  1   FULL PRINTED OUTPUT
  2   ONLY SUMMARIES STORED
OPTION ?
1,
  THE RANDOM NUMBER GENERATORS TO BE INITIALIZED WITH
  1   A NUMBER TO BE SPECIFIED
  2   A RANDOM NUMBER
OPTION ?
1,

2,                        option 2 sub-menu

MANAGEMENT LEVEL ?
1,
```

```
  CULLING DECISIONS TO BE BASED ON
   1   CURRENT LACTATION
   2   ESTIMATED PRODUCING ABILITY
   3   ESTIMATED TRANSMITTING ABILITY
OPTION ?
3,
```

3, *option 3 sub-menu*

```
  ENTER THE SELECTION INDEX WEIGHTS FOR
  MILK, FAT, PROTEIN, AND TYPE
5,5,0,0
```

4, *option 4 sub-menu*

```
  ENTER THE MINIMUM NUMBER OF DAYS AND SERVICES
  AND THE NUMBER OF TOP COWS TO KEEP LONGER
130,4,12,
```
130 days, 4 services, 12 cows

```
  PRESS RETURN FOR TWO LEVELS OR
  ENTER THE EXPONENT FOR AN EXPONENTIAL RELATIONSHIP
1,
```
gives linear relationship

```
ADJUSTMENT FOR DAYS ?
10,
```
e.g. 10 extra for 11th ranked and 110 extra for top cow

```
ADJUSTMENT FOR SERVICES ?
0.3,
```
1 extra service for top 8 cows, 2 for top 5 and 3 for top 2

5, *option 5 sub-menu*

```
DAILY MILK PRODUCTION AT WHICH TO DRY OFF A COW ?
7.,
DAILY MILK PRODUCTION AT WHICH TO SELL A COW SELECTED FOR CULLING ?
12.,
```

6, *option 6 sub-menu*

```
NUMBER OF PROVEN SIRES ?
5,
  ENTER THE METHOD OF SELECTION
  1   INDIVIDUALLY
  2   BY RANK
  3   RANDOMLY
OPTION ?
2,
MAXIMUM SEMEN PRICE ?
500,
```
dollars

```
MINIMUM CONCEPTION RATE ?
20,
```
percent

```
PERCENT OF MATINGS TO YOUNG SIRES ?
30,
NUMBER OF YOUNG BULLS ?
3,
  ENTER THE METHOD OF SELECTION
```
(for young bulls)

```
  1   BY RANK
  2   RANDOMLY
OPTION ?
2,
```

```
NUMBER OF SPECIAL MATINGS ?          for potential young sires
10,
  ENTER THE METHOD OF SELECTION
  1   INDIVIDUALLY
  2   BY RANK
OPTION ?
2,
NUMBER OF TOP BULLS TO USE FOR SPECIAL MATINGS ?
2,

7,   option 7   CONTINUE


    PROGRAM CONTINUES

INTEGER SEED ?  prompts for seed if selected in option 1
875,

        special selections menu for full output herds

    OTHER OPTIONS FOR SPECIFIC ANIMALS
  1   MAKE SPECIFIC MATINGS
  2   SELL WEEK OLD CALVES
  3   SELL YEAR OLD CALVES
  4   SELL YEARLINGS
  5   CONTINUE
OPTION ?   options 1-4 give sub-menus and prompts


1,                           option 1

  BULLS AVAILABLE     bulls selected to be used
    9039  9036  9002  9006  9023  9049  9049  9049
NUMBER OF INDIVIDUAL MATINGS ?
2,
  ENTER    2 COWS EACH FOLLOWED BY A BULL
START A NEW LINE FOR EACH ANIMAL !
1,9039,
2,9006,
        1      9039             check if entries correct
        2      9006
PRESS RETURN IF OK.  ENTER "T" TO RE-ENTER
 RETURN


2,                            option 2
  BULLS AVAILABLE
    9039  9036  9002  9006  9023  9049  9049  9049
NUMBER OF NEW CALVES TO BE CULLED ?
1,
  ENTER    1 COWS
START A NEW LINE FOR EACH ANIMAL !
3,
        3
PRESS RETURN IF OK.  ENTER "T" TO RE-ENTER
 RETURN
  ENTER A CHEAP BULL TO USE ON THESE COWS
9049,
```

```
3,                              option 3
NUMBER OF OLD CALVES TO BE CULLED ?
2,
  ENTER   2 YEAR OLD CALVES
START A NEW LINE FOR EACH ANIMAL !
101,
102,
      101
      102
PRESS RETURN IF OK.  ENTER "T" TO RE-ENTER
 RETURN


4,                                option 4
NUMBER OF YEARLINGS TO BE CULLED ?
1,
  ENTER   1 YEARLINGS
START A NEW LINE FOR EACH ANIMAL !
201,
      201
PRESS RETURN IF OK.  ENTER "T" TO RE-ENTER
 RETURN

5,      CONTINUE
```

2.1.1. Culling cows

In full output mode the cows queued to be culled are listed and an opportunity to change them is given

COWS TO BE CULLED

LOC	COW	CULL DAY	INDEX SCORES LACT	EPA	ETA	MAX PROD	ACT PROD	STATUS
1.	1112.	79.	-19.	-11.	-4.	6179.	895.	ETA
2.	1213.	137.	-19.	-8.	-3.	6979.	1884.	ETA
3.	1313.	142.	3.	-3.	-2.	7177.	2176.	ETA
4.	1513.	161.	-7.	-7.	-2.	5411.	3178.	ETA
5.	1608.	59.	-21.	-9.	-2.	6500.	926.	ETA
6.	1412.	0.	-21.	-8.	-2.	6162.	6162.	KEEP
7.	1206.	0.	-7.	-3.	-2.	6607.	6607.	KEEP
8.	1510.	0.	-13.	-9.	-2.	6914.	6914.	KEEP
9.	1312.	190.	-7.	-6.	-2.	3110.	3110.	FERT
10.	1615.	0.	0.	0.	-1.	5176.	5176.	KEEP
11.	1607.	0.	-20.	-9.	-1.	6003.	6003.	KEEP
12.	1707.	0.	0.	0.	-1.	3.	3.	KEEP

***WHERE*:**

LOC - *cows location number in cull list*
COW - *cows I.D. number*
CULL DAY - *if positive it is the day of the year culled*
- *if zero the cow is not to be culled*
- *if negative it is the day in the next year that the cows production will have declined to the specified daily production to allow culling*

LACT - *deviation from the population average for the index score calculated from the BCA's and type score in the last lactation*
EPA - *the index score calculated from the cows EPA's*
ETA - *the index score calculated from the cows ETA's*

MAX PROD - *years production for a cow if not culled for production*
ACT PROD - *years production if no changes are made in status*
STATUS - *status of a cow at year end*
KEEP *cow kept in the herd*
FERT *cow culled for fertility (can't keep)*
HLTH *cow culled for health (can't keep)*
DEAD *cow is dead*
BCA *culled for low rank* "LACT"
EPA *culled for low rank* "EPA"
ETA *culled for low rank* "ETA"

```
    YOUR FLUID QUOTA IS  1500. HL  PLUS EXCESS OF   500.
     THIS YEARS PRODUCTION WILL BE  1967.8 HECTOLITRES
  ENTER - 1  TO PRINT A LONGER LIST
        - 2  TO CULL MORE COWS
        - 3  TO RESTORE SELECTIVELY CULLED COWS
        - 4  TO CONTINUE
OPTION ?
```

1, *option 1*

HOW MANY MORE COWS DO YOU WANT PRINTED ?

1, *outputs list with 1 more cow*

2, *option 2*

NUMBER OF COWS TO CHANGE ?
2,
COW LOCATION NUMBER ?
13,
COW LOCATION NUMBER ?
6,

the two cows are culled if possible
cow 1412 at location 6 is now culled
cow 1613 at location 13 has too high daily production until the 4th day of the next year

COWS TO BE CULLED

LOC	COW	CULL DAY	INDEX SCORES LACT	EPA	ETA	MAX PROD	ACT PROD	STATUS
1.	1112.	79.	-19.	-11.	-4.	6179.	895.	ETA
6.	1412.	171.	-21.	-8.	-2.	6162.	3387.	ETA
13.	1613.	-4.	0.	0.	-1.	4689.	4689.	KEEP

```
    YOUR FLUID QUOTA IS  1500. HL  PLUS EXCESS OF   500.
     THIS YEARS PRODUCTION WILL BE  1940.9 HECTOLITRES
  ENTER - 1  TO PRINT A LONGER LIST
        - 2  TO CULL MORE COWS
        - 3  TO RESTORE SELECTIVELY CULLED COWS
        - 4  TO CONTINUE
OPTION ?
```

3, *option 3*

NUMBER OF COWS TO CHANGE ?
1,
COW LOCATION NUMBER ?
6,

cow number 1412 is now reinstated and the list is outputted again

4,   *option 4*

the simulation now continues by outputting the end of year herd summary on the screen and all herd information to a file for printing (appendix B2)

2.2: SEMI-AUTOMATED MODE

```
 R SIM.YEAR
#Execution begins
                        DAIRY CATTLE BREEDING SIMULATION
                        --------------------------------

   YOU WILL BE PROMPTED FOR DECISIONS

   - ENTER  ALL VALUES ON ONE LINE (EXCEPT WHEN ENTERING ANIMALS)
   - WHEN ENTERING ANIMALS START A NEW LINE FOR EACH ANIMAL
   - EACH VALUE ENTERED MUST BE FOLLOWED BY A COMMA !!

 HERD NUMBER AND CODE ?
 11,500,

     HERD  11     SIMULATION OF YEAR   9
                                                  MAIN MENU SELECTIONS
      MANAGEMENT DECISIONS SELECTED

    1   OUTPUT PRINTED COPY OF HERDS ? YES
    2   MANAGEMENT LEVEL OF  1
          RANKING AND CULLING BASED ON:  ETA
    3   WEIGHTINGS FOR THE SELECTION INDEX :
          MILK   0.50    FAT    0.50   PROTEIN    0.0    TYPE    0.0
    4   NUMBER OF DAYS OPEN OR SERVICES :
          130. DAYS OR  4. SERVICES WITH EXTRA FOR THE TOP 12. COWS
          BASED ON AN EXPONENT  1.000  AN ADJUSTMENT FOR DAYS  10.00
          AND AN ADJUSTMENT FOR SERVICES  0.300
    5   MINIMUM DAILY MILK PRODUCTION:
          AT  7.0 KG/DAY A COW IS DRIED OFF
          AT 12.0 KG/DAY A CULL COW IS SOLD
    6   MATINGS-    70% TO    5 PROVEN SIRES SELECTED BY RANK
            30% TO    3 YOUNG SIRES SELECTED RANDOMLY
          SPECIAL MATE  10  COWS SELECTED BY RANK
          MAXIMUM SEMEN PRICE $  500/VIAL    MINIMUM CONCEPTION  20%
    7   CONTINUE
 OPTION ?
```

SEMI-AUTOMATED MODE IS SET BY ENSURING OPTION 1 IS SET TO "ONLY SUMMARIES STORED"

1,   *option 1 sub-menu*

```
   THE OUTPUT TO BE
    1   FULL PRINTED OUTPUT
    2   ONLY SUMMARIES STORED
 OPTION ?
 2,
```

allows all other menus to be bypassed

```
  THE RANDOM NUMBER GENERATORS TO BE INITIALIZED WITH
  1   A NUMBER TO BE SPECIFIED
  2   A RANDOM NUMBER
OPTION ?
2,
```

WHEN OPTION 7 IS ENTERED THE PROGRAM COMPLETES EXECUTION

```
OPTION ?
7,   CONTINUE
                HERD    11       YEAR    7        THE SEED WAS    182
                ---------------------------

NO. LAC   CF INT    CONC    SV/CF  NLCW    NBHF    HL MLK  NPRCUL  NFCUL
    37    363.9   0.495   1.545    33       6    2108.      15       5
#Execution terminated
```

2.3: FULLY AUTOMATED MODE

```
#R SIM.YEAR
#Execution begins
                      DAIRY CATTLE BREEDING SIMULATION
                      --------------------------------

  YOU WILL BE PROMPTED FOR DECISIONS

  - ENTER  ALL VALUES ON ONE LINE (EXCEPT WHEN ENTERING ANIMALS)
  - WHEN ENTERING ANIMALS START A NEW LINE FOR EACH ANIMAL
  - EACH VALUE ENTERED MUST BE FOLLOWED BY A COMMA !!

HERD NUMBER AND CODE ?
```

ENTERING A "T" AFTER THE CODE ALLOWS ALL MENUS TO BE BYPASSED

```
11,500,T
:
                HERD    11       YEAR    7        THE SEED WAS    182
                ---------------------------

NO. LAC   CF INT    CONC    SV/CF  NLCW    NBHF    HL MLK  NPRCUL  NFCUL
    37    363.9   0.495   1.545    33       6    2108.      15       5
#Execution terminated
```

3: *YEAR END UPDATE*

#ED CHECK.RUN *To check if all student herds have been run*

```
                      year       herd
:P /F
:     0.001             7           1
:     0.002             7           2
:       .               7           .
:       .               7           .
:     0.012             7          12
:STOP
```

RUN BATCH FILE TO SWITCH HERD AND AI FILES

AND TO SORT DREC *and* BUSE

```
#R  BATCH SCARDS BUPD
#Execution begins
=The batch signon record is:
=$SIG KINN 'SORT FOR AIUPD' DELIVERY=FOR.
=Enter password for KINN.
=
=*BATCH* assigned job number 197718
=*BATCH* RM197718 released  COMMENT="SORT FOR AIUPD" DELIVERY=FOR.
-KINN:RM197718 is executing.
#Execution terminated

#C CHECK.RUN(1,1)
>FILES READY TO RUN "AIUPD"
```

3.2: *PROGRAM "AIUPD"*

```
#RUN AIUPD
#Execution begins
 ENTER THE CURRENT YEAR (NEXT)
 8,

 ENTER THE NUMBER OF BULLS TO ADD, THE MINIMUM INDEX,
 THE MINIMUM ETA FOR TYPE AND THE DAMS MINIMUM ETA
 FOR TYPE
 3,    three bulls and minimum index and type of "0"

 ENTER SELECTION INDEX WEIGHTINGS FOR
 MILK, FAT, PROTEIN AND TYPE
 5,5,
#Execution terminated
```

3.2: *PROGRAM "AIVIEW"*

```
#RUN AIVIEW
#Execution begins
 ENTER INDEX WEIGHTINGS FOR MILK, FAT, PROTEIN AND TYPE
  THEY MUST NOT SUM TO "0"
 5,5,

 ENTER:  "Y" FOR YOUNG SIRES
     OR "P" FOR PROVEN SIRES
     OR RETURN TO STOP
 P

 HOW MANY SIRES DO YOU WANT LISTED ?
 MAXIMUM  39
 39,
```

BULL	SIRE	MGS	MILK	FAT	ETA PROT	TYPE	INDEX	CONCEP- TION	NO. DAU.	NO. HERDS	SEMEN PRICE
9039.	9002.	0.	5.1	9.9	3.1	-0.6	7.5	0.56	20.	13.	23.51
9036.	9002.	0.	5.7	6.3	1.7	-0.4	6.0	0.48	20.	14.	234.78
9002.	0.	0.	3.9	7.0	3.9	-1.7	5.5	0.54	326.	109.	105.55
9032.	9020.	0.	-3.5	-4.4	-2.1	0.1	-3.9	0.50	33.	19.	16.34
9020.	0.	0.	-3.8	-4.8	-1.3	1.2	-4.3	0.53	115.	41.	15.69

```
ENTER:  "Y" FOR YOUNG SIRES
    OR "P" FOR PROVEN SIRES
    OR RETURN TO STOP
Y
```

					ETA			CONCEP-	NO.	NO.	SEMEN
BULL	SIRE	MGS	MILK	FAT	PROT	TYPE	INDEX	TION	DAU.	HERDS	PRICE
9040.	9006.	0.	5.1	5.2	2.4	0.3	5.1				15.00
9041.	9002.	0.	2.4	4.2	2.1	0.1	3.3				15.00
9042.	9019.	0.	2.0	4.4	2.6	0.3	3.2				15.00
9051.	9006.	0.	4.7	5.0	2.2	0.2	4.8				15.00

```
ENTER:  "Y" FOR YOUNG SIRES
    OR "P" FOR PROVEN SIRES
    OR RETURN TO STOP

#Execution terminated
```

3.4: *EMPTY FILES FOR NEXT RUN*

```
#EMPTY YSIRE OK
 File "YSIRE" has been emptied.
#EMPTY DREC OK
 File "DREC" has been emptied.
#EMPTY BUSE OK
```

AFTER THE NUMBER OF DESIRED YEARS ARE SIMULATED

4: FINAL SIMULATION SUMMARIES

```
#RUN CRDBASE
#Execution begins
ENTER  NUMBER OF: YEARS; HERDS/GROUP; GROUPS; AND 1 IF NO STUDENT HERDS
 7,12,2,
#Execution terminated
```

THE SUMMARIES ARE NOW READY TO RUN THE STATISTICAL ANALYSIS PROGRAM

4.2: *Treatments for this Simulation*

This sample run had 12 single herds and a duplicate group of 12 herds. The simulation was run for 7 years with the following options selected:

OPTIONS	*HERDS*
Management level 1	*1, 3, 5, 7, 9, 11*
Management level 2	*101, 103, 105, 107, 109, 111*
Management level 3	*102, 104, 106, 108, 110, 112*
Management level 4	*2, 4, 6, 8, 10, 12*
Cull fertility - 5 services or 305 days	*1, 2, 101, 102*
Cull fertility - 4 services or 305 days	*3, 4, 103, 104*
Cull fertility - 3 services or 305 days	*5, 6, 105, 106*
Cull fertility - 3 services or 120 days *top 10 cows - 5 services or 150 days*	*7, 8, 107, 108*
Cull fertility - 3 services or 140 days *top 10 cows - 5 services or 200 days*	*9, 10, 109, 110*
Cull fertility - 3 services or 130 days *top 12 cows kept longer as a function of rank* *days = 130 + 10 x (12 - rank)* *services = 3 + 0.3 x (12 - rank)*	*11, 12, 111, 112*
Bred to top 5 bulls with *semen price of less than 500.*	*1 - 12*
Bred to top 10 bulls with *semen price of less than 100.*	*101 - 112*

6: PROGRAM "STAT.ANAL"

```
R STAT.ANAL
Execution begins
ENTER YOUR HERD NUMBER
11,   Used only to direct output to student file
ENTER THE NUMBER OF GROUPS
4,    Up to 10 groups of herds

ENTER THE FIRST YEAR AND LAST YEAR
1,7,  consecutive years only

ENTER THE TITLE FOR THIS RUN
( LABEL WILL BE TRUNCATED TO 75 CHARACTERS )
MANAGEMENT LEVELS
```

```
ENTER GROUP   1  LABEL
GROUP NO., HERDS AND TREATMENT
( LABEL WILL BE TRUNCATED TO 52 CHARACTERS )
LEVEL 1

ENTER THE NUMBER OF HERDS IN THIS GROUP
6,  Up to 30 herds per group

ENTER THE  6 HERDS
1,3,5,7,9,11,

ENTER GROUP   2  LABEL
GROUP NO., HERDS AND TREATMENT
( LABEL WILL BE TRUNCATED TO 52 CHARACTERS )
LEVEL 2

ENTER THE NUMBER OF HERDS IN THIS GROUP
6,

ENTER THE  6 HERDS
101,103,105,107,109,111,

ENTER GROUP   3  LABEL
GROUP NO., HERDS AND TREATMENT
( LABEL WILL BE TRUNCATED TO 52 CHARACTERS )
LEVEL 3

ENTER THE NUMBER OF HERDS IN THIS GROUP
6,

ENTER THE  6 HERDS
102,104,106,108,110,112,

ENTER GROUP   4  LABEL
GROUP NO., HERDS AND TREATMENT
( LABEL WILL BE TRUNCATED TO 52 CHARACTERS )
LEVEL 4

ENTER THE NUMBER OF HERDS IN THIS GROUP
6,

ENTER THE  6 HERDS
2,4,6,8,10,12,

VARIABLE  #  1
1  SIMPLE VARIABLE
2  COMBINE BREEDING VALUES
3  OTHER CALCULATIONS
1,      A simple variable from table XI

ENTER A VARIABLE NUMBER
 OR PRESS RETURN FOR NO MORE VARIABLES
25,

THE VARIABLE IS:    CALVING INTERVAL    variable  25
PRESS RETURN IF OK... OR ENTER "T" TO RETRY.
 RETURN

PRESS RETURN FOR ANOTHER VARIABLE OR ENTER "T" TO STOP
<RETURN>
```

VARIABLE # 2
1 SIMPLE VARIABLE
2 COMBINE BREEDING VALUES
3 OTHER CALCULATIONS
2, *Allows weighted averages of breeding values for a trait to be combined as a new variable*

ENTER THE TRAIT OF INTREST
1 MILK BCA
2 FAT BCA
3 PROTEIN BCA
4 TYPE SCORE
5 INDEX SCORE
5,
ENTER THE SELECTION INDEX WEIGHTS
OR PRESS RETURN TO USE VARIABLES 159 - 162 *Index used in the herd*
9.3,6.2,-1,0,
ENTER THE NUMBER OF ANIMAL GROUPS TO COMBINE
1,

FOR THE ANIMAL GROUP NUMBER 1
ENTER THE VARIABLE FOR NUMBER OF ANIMALS OR CONCEPTIONS IN THE GROUP
55, *Use only variable numbers ending in 0 or 5 between 55-130 or 140, 148 or 156.*

THE VARIABLE IS: NO. LIVE COWS
PRESS RETURN IF OK... OR ENTER "T" TO RETRY.
RETURN

ENTER THE VARIABLE NAME
(LABEL WILL BE TRUNCATED TO 28 CHARACTERS)
BV LIVE COWS ECON. INDEX

PRESS RETURN FOR ANOTHER VARIABLE OR ENTER "T" TO STOP
RETURN

VARIABLE 3
1 SIMPLE VARIABLE
2 COMBINE BREEDING VALUES
3 OTHER CALCULATIONS
3, *Allows other calculations to create new variables from those* in table *XI*

ENTER A VARIABLE NUMBER
OR PRESS RETURN FOR NO MORE VARIABLES
54,

THE VARIABLE IS: NET INCOME
PRESS RETURN IF OK... OR ENTER "T" TO RETRY.
<RETURN>

WEIGHT FOR NET INCOME *Multiply 54 by another variable or number*
PRESS RETURN FOR NO WEIGHT OR ENTER A VARIABLE NUMBER
OR ENTER A WEIGHT FOLLOWED BY A "T"
RETURN

ENTER A VARIABLE NUMBER
OR PRESS RETURN FOR NO MORE VARIABLES
44, *Variable to be added to 54*

```
THE VARIABLE IS:    SOLD - PRODUCTION ($)
PRESS RETURN IF OK... OR ENTER "T" TO RETRY.
 RETURN

WEIGHT FOR   SOLD - PRODUCTION ( )
  PRESS RETURN FOR NO WEIGHT   OR ENTER A VARIABLE NUMBER
   OR ENTER A WEIGHT FOLLOWED BY A "T"
-1,T      Multiply by -1 gives subtraction

ENTER A VARIABLE NUMBER
 OR PRESS RETURN FOR NO MORE VARIABLES
 RETURN    Numerator complete

ENTER "T" IF CALCULATIONS COMPLETE
OR PRESS RETURN IF DENOMINATOR REQUIRED
T      No denominator

ENTER THE VARIABLE NAME
( LABEL WILL BE TRUNCATED TO 28 CHARACTERS )
NET INC. - INC. PROD. CULLS

PRESS RETURN FOR ANOTHER VARIABLE OR ENTER "T" TO STOP
T
#Execution terminated
```

Output is directed to files as follows:
(see appendix B)

H?-?	Analysis of covariance, mean and slope tests both within and between groups (?'s are consecutive numbers one of which is entered for the herd number.)
-DAT	Data used for analysis

Appendix B21

ANNOTATATED SAMPLE HERD OUTPUT FROM A "SIM.YEAR" RUN WITH THE FULL HERD OUTPUT OPTION

COMPUTER OUTPUT IS ILLUSTRATED IN **BOLD FACE CHARACTERS**

ANNOTATIONS ARE *LIGHTER ITALICS* .

HERD 11 YEAR 9 **THE SEED WAS 875** *to repeat a herd year*

2 MANAGEMENT LEVEL OF 1
RANKING AND CULLING BASED ON: ETA
3 WEIGHTINGS FOR THE SELECTION INDEX :
MILK 0.50 FAT 0.50 PROTEIN 0.0 TYPE 0.0
4 NUMBER OF DAYS OPEN OR SERVICES :
130. DAYS OR 4. SERVICES WITH EXTRA FOR THE TOP 12. COWS
BASED ON AN EXPONENT 1.000 AN ADJUSTMENT FOR DAYS 10.00
AND AN ADJUSTMENT FOR SERVICES 0.300
5 MINIMUM DAILY MILK PRODUCTION:
AT 7.0 KG/DAY A COW IS DRIED OFF
AT 12.0 KG/DAY A CULL COW IS SOLD
6 MATINGS- 70 TO 5 PROVEN SIRES SELECTED BY RANK
30 TO 3 YOUNG SIRES SELECTED RANDOMLY
SPECIAL MATE 10 COWS SELECTED BY RANK
MAXIMUM SEMEN PRICE 500/VIAL MINIMUM CONCEPTION 20

final selections made

BULLS USED, THEIR PROPORTION OF USE AND THEIR INDEX
(*) PRELIMINARY PROOF ONLY (**) PEDIGREE ESTIMATE ONLY

9039	**9036**	**9002**	**9006**	**9023**	**9049**	**9049**	**9049**	*bull number*
15	**15**	**15**	**15**	**15**	**10**	**10**	**10**	*relative use*
*****					******	******	******	*reliability of estimate*
7.5	**6.0**	**5.5**	**4.1**	**4.1**	**4.9**	**4.9**	**4.9**	*index scores*

```
                        CURRENT PRODUCTION                           MEAN BCA          EPA                      ETA
                  ---------------------------- FEED TYP NO  ----------   -----------------         -----------------
COW SIRE DAM AGE STAT DAY FRESH DAYS   MILK   FAT   PROT COST  SC RC  MLK FAT PRO  MILK FAT  PROT TYPE  IND  MILK FAT  PROT TYPE  IND

              reproductive                                  years       #                                    own
    identity    status - Bred,Open or culled                feed   completed             EPA's              index        ETA's        own
     numbers              culled for:                       cost   lactations                                                         index
                            FERT-fertility
 455 9003    0 15 BRED 227  BCA-last lactation             1168. 86  9  146 147 145  11.1 12.7 10.9  3.6 11.9  2.0  2.7  2.2  0.8  2.4
                            EPA's or ETA's

        LACTATION  # 9    82/ 8  32   290.  10.1   9.5  COMPLETE RECORD 315 DAYS MILKED  7221. 252.6 235.4   136 132 136
                  parity        days                     total days in production                            BCA values
                  number      producing                  for this complete lactation                         for this
                              in current                                                                     lactation
                                 year

        LACTATION  #10   123/ 9 242  6784. 244.5 212.7    PROJECTED TO 305 DAY           7500. 270.3 235.1   141 140 137
                                                                                          total production and BCA's
                                                                                          if lactation continues to
                                                                                               305-days

 708 9008     0 12 BRED 307                                1260. 85  9  146 148 143  11.7 13.9  9.2  0.9 12.8  2.2  3.2  1.8  0.2  2.7
             age     day                                    type      average
                     culled                                 score     BCA's
                   or serviced                                       all lactations

        LACTATION  # 9   143/ 8  96  1154.  41.5  37.4  COMPLETE RECORD 318 DAYS MILKED  7784. 279.7 252.0   144 142 143
                        day and         production in                                   total lactation
                        year of         current year                                     yield milk, fat,
                       parturition       milk,fat,protein                                  protein

        LACTATION  #10   146/ 9 219  6958. 257.4 221.0    PROJECTED TO 305 DAY           8104. 299.8 257.4   150 151 147
 804 9013     0 11 BRED 248                                1255. 84  8  143 145 143   8.5 11.2  9.0  2.0  9.9  1.4  2.3  1.7  0.4  1.8
        LACTATION  # 8   144/ 8  92  1095.  40.0  35.2  COMPLETE RECORD 313 DAYS MILKED  7517. 274.3 241.3   140 141 139
        LACTATION  # 9   142/ 9 223  6881. 259.0 221.8    PROJECTED TO 305 DAY           7930. 298.5 255.6   147 151 146
1112 9022  813  8 ETA   79                                  181. 81  6  124 118 120  -8.4-12.7-11.0 -0.9-10.6 -2.4 -4.9 -3.1 -0.5 -3.7
        LACTATION  # 6   156/ 8  79   895.  28.9  28.6  COMPLETE RECORD 288 DAYS MILKED  5993. 193.8 191.5   118 109 116
1206 9030  577  7 BRED 267                                 1115. 82  5  129 129 130  -3.2 -2.9 -2.1 -0.7 -3.0 -1.6 -2.1 -0.8  0.0 -1.8
        LACTATION  # 5   205/ 8 147  2035.  71.7  68.0  COMPLETE RECORD 307 DAYS MILKED  6559. 231.0 219.0   128 125 130
        LACTATION  # 6   200/ 9 165  4572. 161.8 156.9    PROJECTED TO 305 DAY           6467. 228.9 221.8   127 124 131
1213 9029  901  7 ETA  137                                  343. 79  5  126 122 119  -5.9 -9.1-11.2 -3.1 -7.5 -2.4 -4.6 -3.5  0.0 -3.5
        LACTATION  # 5   211/ 8 137  1884.  61.2  57.3  COMPLETE RECORD 291 DAYS MILKED  6002. 194.9 182.5   119 110 113
1302 9006  573  6 BRED 345                                 1380. 83  4  152 160 151  14.9 22.1 14.9 -1.2 18.5  2.1  5.8  3.2  0.0  3.9
        LACTATION  # 4   257/ 8 175  4085. 160.0 134.9  COMPLETE RECORD 283 DAYS MILKED  8635. 338.1 285.1   154 162 156
        LACTATION  # 5   225/ 9 140  4981. 191.5 166.1    PROJECTED TO 305 DAY           7941. 305.3 264.8   147 153 150
1312 9018 1010  6 FERT 190                                  529. 77  4  127 125 122  -4.8 -6.4 -8.8 -3.8 -5.6 -1.3 -2.0 -1.7  0.0 -1.6
        LACTATION  # 4   247/ 8 190  3110. 107.4  97.2  COMPLETE RECORD 308 DAYS MILKED  6631. 229.0 207.2   129 124 125
```

```
                               CURRENT PRODUCTION                         MEAN BCA          EPA                      ETA
                         ---------------------------- FEED TYP NO  ----------   -----------------         -----------------
COW SIRE DAM AGE STAT DAY FRESH DAYS   MILK  FAT  PROT COST  SC RC  MLK FAT PRO  MILK FAT  PROT TYPE  IND  MILK FAT  PROT TYPE  IND

1313 9002 1012  6 ETA  142                               386. 82  4  130 128 129  -2.5 -4.1 -3.2  0.5 -3.3  -1.9 -2.7 -1.6  0.0 -2.3
         LACTATION  # 4   204/ 8 142  2176.  77.5  69.9  COMPLETE RECORD 303 DAYS MILKED   7247. 258.2 232.8    137 135 136
1411 9018 1005  5 OPEN 320                              1378. 77  3  137 138 131   2.9  4.2 -1.4 -4.1  3.5   1.2  1.8  0.1 -1.0  1.5
         LACTATION  # 3   177/ 8 120  1498.  53.9  45.6  COMPLETE RECORD 308 DAYS MILKED   6625. 238.6 201.6    137 135 130
         LACTATION  # 4   176/ 9 189  7682. 306.9 225.4     PROJECTED TO 305 DAY          9733. 388.8 285.6    172 186 159
1412 9035 1007  5 BRED 310                              1078. 86  3  124 120 127  -6.9 -9.6 -4.6 -0.5 -8.2  -1.4 -2.6 -0.8  0.0 -2.0
         LACTATION  # 3   263/ 8 185  2763.  92.0  93.6  COMPLETE RECORD 287 DAYS MILKED   5349. 178.2 181.2    116 108 118
         LACTATION  # 4   256/ 9 109  3399. 122.7 119.9     PROJECTED TO 305 DAY          6836. 246.7 241.1    132 131 140
1414 9035 1010  5 DEAD  73                               268. 85  3  136 134 134   1.7  1.2  0.7  0.2  1.4   0.2  0.4  0.5  0.0  0.3
         LACTATION  # 3   246/ 8  73  1825.  66.8  60.0  COMPLETE RECORD 192 DAYS MILKED   7287. 266.6 239.4    124 124 124
1416 9035 1104  5 BRED 360                              1084. 81  3  144 144 142   8.4  9.1  7.0 -0.9  8.8   1.2  2.2  1.4  0.0  1.7
         LACTATION  # 3   238/ 8 183  2939. 105.1  94.3  COMPLETE RECORD 310 DAYS MILKED   6812. 243.5 218.7    139 137 138
         LACTATION  # 4   269/ 9  96  3251. 116.7 104.2     PROJECTED TO 305 DAY          7374. 264.8 236.4    139 138 138
1423 9037 1208  5 OPEN 298                              1229. 81  2  140 143 140   4.6  7.2  5.0 -1.3  5.9   0.3  1.2  0.5  0.0  0.8
         LACTATION  # 2   162/ 8  95  1068.  39.6  34.6  COMPLETE RECORD 298 DAYS MILKED   6037. 224.0 195.5    136 138 136
         LACTATION  # 3   178/ 9 187  6284. 250.8 204.7     PROJECTED TO 305 DAY          8062. 321.7 262.6    159 170 159
1501 9006  455  4 BRED 339                              1205. 85  2  141 144 142   5.2  7.8  6.3  2.6  6.5   0.8  2.8  2.4  0.0  1.8
         LACTATION  # 2   229/ 8 170  2584.  95.5  84.2  COMPLETE RECORD 306 DAYS MILKED   6550. 242.1 213.4    146 147 146
         LACTATION  # 3   227/ 9 138  4781. 183.2 157.1     PROJECTED TO 305 DAY          7724. 296.0 253.7    154 159 155
1510 9002 1007  4 BRED 209                              1150. 78  2  121 119 120  -8.5 -9.6 -8.8 -2.5 -9.0  -1.4 -2.2 -1.5 -0.4 -1.8
         LACTATION  # 2   182/ 8 100  1127.  39.3  36.1  COMPLETE RECORD 283 DAYS MILKED   5260. 183.6 168.7    122 118 120
         LACTATION  # 3   152/ 9 213  5787. 219.0 183.3     PROJECTED TO 305 DAY          6877. 260.3 217.8    141 144 138
1511 9023 1010  4 BRED 283                              1074. 80  2  129 127 126  -3.3 -3.8 -4.6 -1.0 -3.5  -0.7 -0.7 -0.5  0.0 -0.7
         LACTATION  # 2   224/ 8 144  1897.  64.3  61.5  COMPLETE RECORD 285 DAYS MILKED   5335. 180.8 172.9    123 117 122
         LACTATION  # 3   208/ 9 157  4333. 154.7 138.0     PROJECTED TO 305 DAY          6383. 227.9 203.2    133 131 131
1513 9023 1205  4 ETA  161                               506. 82  2  124 122 124  -6.6 -7.6 -5.9 -1.4 -7.1  -1.7 -2.8 -1.5 -0.0 -2.2
         LACTATION  # 2   246/ 8 161  3178. 112.5 105.2  COMPLETE RECORD 280 DAYS MILKED   5576. 197.4 184.6    151 148 153
1514 9017 1207  4 BRED 302                              1143. 79  2  144 150 141   7.7 12.5  6.0 -1.6 10.1   2.4  4.6  2.0 -0.5  3.5
         LACTATION  # 2   206/ 8 157  2298.  89.1  72.2  COMPLETE RECORD 316 DAYS MILKED   7239. 280.7 227.3    156 163 152
         LACTATION  # 3   244/ 9 121  4259. 166.0 134.7     PROJECTED TO 305 DAY          7718. 300.8 244.2    154 161 150
1516 9002 1213  4 BRED  89                              1281. 84  2  158 167 158  16.8 23.8 17.9  0.6 20.3   2.6  5.0  2.5  0.1  3.8
         LACTATION  # 2    24/ 9 298  7466. 296.0 246.7  COMPLETE RECORD 298 DAYS MILKED   7532. 298.6 248.9    161 172 163
1518 9017 1215  4 BRED 109                              1180. 86  2  148 152 151  10.2 13.4 12.9  2.2 11.8   2.1  3.7  3.2  0.8  2.9
         LACTATION  # 2    43/ 9 299  6881. 262.6 232.2  COMPLETE RECORD 299 DAYS MILKED   6934. 264.7 234.0    151 157 156
1520 9006 1305  4 BRED 257                              1099. 83  1  130 131 129  -1.9 -1.1 -2.1  0.5 -1.5  -0.7  0.7  0.6  0.0  0.0
         LACTATION  # 1   203/ 8 128  1880.  69.1  60.9  COMPLETE RECORD 290 DAYS MILKED   5744. 211.1 185.9    128 130 128
         LACTATION  # 2   178/ 9 187  4604. 164.4 148.0     PROJECTED TO 305 DAY          5971. 213.2 192.0    136 134 135
1601 9017  804  3 OPEN   0                              1047. 82  1  124 125 125  -5.1 -4.3 -4.4  0.2 -4.7  -0.6 -0.5 -0.3  0.2 -0.6
         LACTATION  # 1     4/ 9 283  5140. 188.9 170.6  COMPLETE RECORD 283 DAYS MILKED   5340. 196.3 177.3    121 122 123
         LACTATION  # 2   337/ 9  28   673.  25.5  21.6
1602 9041  905  3 BRED 327                              1195. 79  1  131 134 130  -1.1  0.8 -1.7 -1.4 -0.2  -0.7 -0.2 -0.7  0.0 -0.4
         LACTATION  # 1   308/ 8 227  4511. 169.6 144.5  COMPLETE RECORD 284 DAYS MILKED   5902. 221.8 189.1    130 134 128
         LACTATION  # 2   277/ 9  88  2817. 108.1  89.5
1604 9002 1010  3 BRED 351                              1093. 82  1  138 138 139   2.3  3.2  3.6  0.3  2.8   0.6  1.6  1.5  0.0  1.1
         LACTATION  # 1   244/ 8 191  2745. 100.6  91.5  COMPLETE RECORD 312 DAYS MILKED   5274. 193.2 175.7    138 139 140
         LACTATION  # 2   241/ 9 124  3531. 133.4 114.9     PROJECTED TO 305 DAY          6343. 239.7 206.4    142 146 142
```

```
                                    CURRENT PRODUCTION                          MEAN BCA         EPA                        ETA
                                 ------------------------------ FEED TYP NO   ----------    ------------------         ------------------
COW SIRE DAM AGE STAT DAY FRESH DAYS    MILK   FAT  PROT COST  SC RC  MLK FAT PRO  MILK FAT  PROT TYPE  IND  MILK FAT  PROT TYPE  IND

1605 9023 1112  3 BRED 132                               1003. 82  1   139 139 141   3.0  3.7  4.6  0.3  3.4  -0.4 -0.8 -0.2 -0.1 -0.6
         LACTATION  # 1    43/ 9 317  5394. 196.5 180.0  COMPLETE RECORD 317 DAYS MILKED   5394. 196.5 180.0      139 140 142
1607 9041 1207  3 BRED 293                               1047. 81  1   118 116 115  -8.3 -9.3-10.0 -1.9 -8.8  -0.8 -1.0 -1.3 -0.5 -0.9
         LACTATION  # 1   264/ 8 182  2744.  96.4  85.6  COMPLETE RECORD 283 DAYS MILKED   4877. 171.3 152.1      114 112 111
         LACTATION  # 2   232/ 9 133  3258. 115.1 105.9     PROJECTED TO 305 DAY          5530. 195.3 179.8      129 126 128
1608 9019 1304  3 ETA   59                                161. 83  1   118 113 118  -8.3-10.7 -7.9  0.4 -9.5  -1.4 -2.6 -1.3  0.1 -2.0
         LACTATION  # 1   215/ 8  59   926.  31.3  30.6  COMPLETE RECORD 209 DAYS MILKED   4806. 162.5 158.9       93  89  93
1610 9023 1311  3 BRED 274                               1182. 81  1   126 126 125  -3.8 -3.5 -4.5 -0.6 -3.7  -0.5 -0.3 -0.3  0.0 -0.4
         LACTATION  # 1   241/ 8 161  2567.  93.1  82.1  COMPLETE RECORD 285 DAYS MILKED   5512. 200.0 176.4      124 124 122
         LACTATION  # 2   211/ 9 154  4680. 177.5 148.9     PROJECTED TO 305 DAY          6952. 263.6 221.2      152 157 150
1611 9002 1209  3 HLTH  21                                 35. 80  0     0   0   0   0.3  0.3  0.2 -0.2  0.3   0.5  1.0  0.8  0.3  0.7
1613 9044 1403  3 BRED 212                                970. 78  0     0   0   0   0.3  0.3  0.2 -0.2  0.3  -0.8 -0.6 -0.0  0.0 -0.7
         LACTATION  # 1   136/ 9 229  4689. 174.5 147.6     PROJECTED TO 305 DAY          5532. 205.8 174.2      144 147 140
1615 9006 1412  3 OPEN 230                                995. 79  0     0   0   0   0.3  0.3  0.2 -0.2  0.3  -1.5 -0.6  0.4  0.0 -1.1
         LACTATION  # 1   104/ 9 261  5176. 187.8 160.0     PROJECTED TO 305 DAY          5623. 204.0 173.8      146 146 140
1616 9023 1416  3 BRED 105                               1007. 78  1   140 141 137   3.4  4.4  2.5 -1.6  3.9   1.1  1.9  1.0 -0.3  1.5
         LACTATION  # 1    53/ 9 285  5338. 195.8 169.6  COMPLETE RECORD 285 DAYS MILKED   5523. 202.6 175.4      140 141 137
1617 9044 1419  3 BRED 278                                888. 78  0     0   0   0   0.3  0.3  0.2 -0.2  0.3   0.0  0.1 -0.1 -0.3  0.1
         LACTATION  # 1   171/ 9 194  3818. 133.5 116.0     PROJECTED TO 305 DAY          5111. 178.7 155.2      136 132 129
1618 9019 1420  3 BRED 184                                999. 79  1   141 140 138   3.8  4.0  2.9 -1.2  3.9  -0.0 -0.3  0.0 -0.3 -0.2
         LACTATION  # 1    59/ 9 306  5392. 194.3 171.3        RECORD TO 306 DAYS         5392.  194.  171.      141 140 138
1701 9044  697  2 OPEN   0                                705. 81  0     0   0   0   0.3  0.3  0.2 -0.2  0.3   0.2  0.3 -0.0 -0.1  0.3
         LACTATION  # 1   307/ 9  58  1165.  44.1  37.7
1702 9006  708  2 BRED 347                                734. 78  0     0   0   0   0.3  0.3  0.2 -0.2  0.3   0.2  2.2  1.7  0.0  1.2
         LACTATION  # 1   278/ 9  87  1630.  58.6  51.6
1703 9006  804  2 OPEN   0                                664. 78  0     0   0   0   0.3  0.3  0.2 -0.2  0.3  -0.2  1.7  1.6  0.0  0.8
         LACTATION  # 1   326/ 9  39   669.  25.1  21.4
1704 9002  905  2 FERT  96                                162.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0  -0.5 -0.4 -0.3  0.3  0.0
1705 9002 1010  2 BRED 273                                891. 77  0     0   0   0   0.3  0.3  0.2 -0.2  0.3   0.3  1.2  0.9  0.0  0.7
         LACTATION  # 1   205/ 9 160  3587. 133.1 110.1     PROJECTED TO 305 DAY          5616. 208.5 172.4      145 148 139
1706 9002 1201  2 FERT -50                                -86.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0   0.2  0.8  1.1  0.0  0.0
1707 9019 1206  2 OPEN   0                                616. 84  0     0   0   0   0.3  0.3  0.2 -0.2  0.3  -0.8 -1.0 -0.4  0.0 -0.9
         LACTATION  # 1   364/ 9   1     3.   0.1   0.1
1708 9028 1411  2 BRED 112                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0   0.6  0.9  0.0 -0.5  0.0
1709 9023 1414  2 BRED 110                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0   0.1  0.2  0.2  0.0  0.0
1710 9028 1416  2 BRED 152                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0   0.6  1.1  0.7  0.0  0.0
1711 9043 1421  2 OPEN   0                                681. 77  0     0   0   0   0.3  0.3  0.2 -0.2  0.3   0.2  0.6  0.6  0.0  0.4
         LACTATION  # 1   323/ 9  42   839.  32.9  26.2
1712 9002 1424  2 BRED 358                                740. 80  0     0   0   0   0.3  0.3  0.2 -0.2  0.3  -0.8 -0.5 -0.2  0.0 -0.7
         LACTATION  # 1   292/ 9  73  1614.  60.2  50.3
1713 9044 1426  2 BRED 348                                724. 79  0     0   0   0   0.3  0.3  0.2 -0.2  0.3   0.1  0.2  0.3  0.2  0.1
         LACTATION  # 1   286/ 9  79  1487.  52.9  48.0
1714 9036 1201  2 BRED 284                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0  -1.9 -0.2  1.2  0.0  0.0
1715 9047 1503  2 BRED 248                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0  -0.1  0.4  0.6  0.0  0.0
1716 9002 1507  2 BRED 139                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0   0.3  1.3  1.1  0.0  0.0
1717 9023 1509  2 BRED 236                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0  -1.0 -1.4 -0.9  0.0  0.0
1719 9006 1511  2 BRED 152                                616.  0  0     0   0   0   0.0  0.0  0.0  0.0  0.0  -1.2  0.3  0.6  0.0  0.0
```

YEARLINGS

I.D. numbers

HEIFER	SIRE	DAM	DATE *day 1st birthday*	ETA MILK	ETA FAT	ETA PROT	ETA TYPE	ETA INDEX	FEED COSTS	STAT *dead ?*
1801.	9036.	708.	143.	-1.1	0.5	1.0	0.0	-0.3	348.94	
1802.	9036.	905.	177.	-2.6	-1.4	-0.2	0.0	-2.0	341.42	
1803.	9002.	1209.	14.	0.5	1.0	0.8	0.3	0.8	377.59	
1804.	9036.	1312.	247.	-2.8	-2.0	-0.8	0.0	-2.4	325.79	
1805.	9018.	1313.	204.	-1.0	-1.3	-0.8	0.0	-1.2	335.39	
1806.	9036.	1411.	177.	-1.6	-0.1	0.1	0.0	-0.9	341.42	
1807.	9048.	1412.	263.	-0.7	-1.3	-0.4	0.0	-1.0	322.34	
1808.	9048.	1414.	246.	0.1	0.2	0.2	0.0	0.1	326.01	
1809.	9023.	1423.	162.	0.1	0.6	0.2	0.0	0.4	344.75	
1810.	9048.	1501.	229.	0.4	1.4	1.2	0.0	0.9	329.88	
1811.	9002.	1511.	224.	-0.3	-0.3	-0.3	0.0	-0.3	330.90	
1812.	9018.	1513.	246.	-0.9	-1.4	-0.8	-0.0	-1.1	326.11	
1813.	9023.	1514.	206.	1.2	2.3	1.0	-0.2	1.8	334.98	
1814.	9023.	1516.	26.	1.3	2.5	1.3	0.1	1.9	374.93	
1815.	9047.	1518.	35.	1.1	1.9	1.6	0.4	1.5	372.84	
1816.	9048.	1519.	114.	0.1	0.4	0.5	0.0	0.3	355.40	
1817.	9048.	1520.	203.	0.0	0.0	0.0	0.0	0.0	158.72	DEAD
1818.	9018.	1522.	57.	-0.1	-0.1	-0.1	0.0	-0.1	368.01	
1819.	9049.	1604.	244.	0.3	0.8	0.7	0.0	0.6	326.46	
1820.	9036.	1610.	241.	-2.4	-1.2	-0.1	0.0	-1.8	327.22	

CALVES

I.D. numbers

CALF	SIRE	DAM	*birth* DATE	ETA MILK	ETA FAT	ETA PROT	ETA TYPE	ETA INDEX	FEED COSTS	SEX	STATUS	
1901.	9036.	455.	123.	-1.2	0.3	1.2	0.0	-0.4	198.71	HEIF		*sold as*
1.	9039.	708.	146.	1.1	1.6	0.9	0.1	1.3	0.0	BULL	S AI	*potential A.I. bull*
1902.	9039.	804.	142.	0.7	1.1	0.9	0.2	0.9	183.32	HEIF		
1903.	9006.	1206.	200.	-1.7	-0.4	0.4	0.0	-1.0	135.48	HEIF		
2.	9036.	1302.	225.	-1.1	1.8	1.7	0.0	0.4	0.0	BULL	S AI	
0.	9006.	1411.	176.	-0.3	1.6	0.8	0.0	0.0	0.0	BULL	SOLD	
1904.	9036.	1412.	256.	-2.9	-2.3	-0.3	0.0	-2.6	89.50	HEIF		
1905.	9036.	1416.	269.	-1.6	0.0	0.8	0.0	-0.8	78.83	HEIF		
0.	9039.	1423.	178.	0.1	0.6	0.2	0.0	0.0	0.0	BULL	SOLD	
1906.	9036.	1501.	227.	-1.8	0.4	1.3	0.0	-0.7	113.31	HEIF		
1907.	9049.	1510.	152.	-0.7	-1.1	-0.8	-0.2	-0.9	174.89	HEIF		
0.	9002.	1511.	208.	-0.3	-0.3	-0.3	0.0	0.0	0.0	BULL	SOLD	
1908.	9039.	1514.	244.	1.2	2.3	1.0	-0.2	1.8	99.35	HEIF		
0.	9049.	1516.	24.	1.3	2.5	1.3	0.1	0.0	0.0	BULL	SOLD	
1909.	9036.	1518.	43.	-1.1	0.8	1.7	0.0	-0.2	264.08	HEIF		

CALVES

CALF	SIRE	DAM	DATE	ETA MILK	ETA FAT	ETA PROT	ETA TYPE	ETA INDEX	FEED COSTS	SEX	STATUS
1910.	9049.	1520.	178.	-0.3	0.3	0.3	0.0	0.0	153.55	HEIF	
0.	9002.	1601.	4.	-0.3	-0.3	-0.2	0.1	0.0	0.0	BULL	SOLD
1911.	9050.	1604.	241.	0.3	0.8	0.7	0.0	0.6	101.47	HEIF	
0.	9006.	1605.	43.	-1.1	0.3	0.7	0.0	0.0	0.0	BULL	SOLD
1912.	9049.	1607.	232.	-0.4	-0.5	-0.6	-0.3	-0.5	109.21	HEIF	
1913.	9006.	1610.	211.	-1.1	0.5	0.7	0.0	-0.3	126.45	HEIF	
1914.	9049.	1613.	136.	-0.4	-0.3	-0.0	0.0	-0.4	188.03	HEIF	
1915.	9039.	1615.	104.	-0.7	-0.3	0.2	0.0	-0.5	214.31	HEIF	
1916.	9049.	1616.	53.	0.6	1.0	0.5	-0.2	0.8	256.18	HEIF	
0.	9002.	1617.	171.	0.0	0.0	-0.0	-0.2	0.0	0.0	BULL	SOLD
0.	9036.	1618.	59.	-2.2	-1.2	0.1	0.0	0.0	0.0	BULL	SOLD
0.	9039.	1601.	337.	-0.3	-0.3	-0.2	0.1	0.0	0.0	BULL	SOLD
0.	9006.	1602.	277.	-1.2	0.6	0.5	0.0	0.0	0.0	BULL	SOLD
0.	9049.	1701.	307.	0.1	0.2	-0.0	-0.1	0.0	0.0	BULL	SOLD
0.	9002.	1702.	278.	0.1	1.1	0.8	0.0	0.0	0.0	BULL	SOLD
0.	9049.	1703.	326.	-0.1	0.9	0.8	0.0	0.0	0.0	BULL	SOLD
0.	9023.	1705.	205.	0.1	0.6	0.5	0.0	0.0	0.0	BULL	SOLD
1917.	9023.	1707.	364.	-0.4	-0.5	-0.2	0.0	-0.4	0.82	HEIF	
1918.	9049.	1711.	323.	0.1	0.3	0.3	0.0	0.2	34.49	HEIF	
1919.	9039.	1712.	292.	-0.4	-0.3	-0.1	0.0	-0.3	59.94	HEIF	
1920.	9002.	1713.	286.	0.0	0.1	0.2	0.1	0.1	64.55	HEIF	

ROLLING HERD AVERAGES

Average production in 305 days or less (for cows dried off before 305 days) of all lactations completed or reaching 305-days in the current year

BEFORE CULLING

records as they would have been if no cows were culled for low production or type

NO. REC.	MILK	FAT	PROTEIN
31	5955.	215.0	192.7

average AGE *at parturition*	*average* DAYS MILK*ed*	*percent* FAT	PROTEIN
4.6	292.3	3.61	3.24

BCA *milk*	BCA *fat*	BCA *protein*	TYPE *score*
133.	133.	133.	82.

AFTER CULLING

some records are usually shorter
a few can be missed

NO. REC.	MILK	FAT	PROTEIN
31	5948.	214.8	192.5

AGE	DAYS MILK	FAT	PROTEIN
4.6	289.2	3.61	3.24

BCA			TYPE
133.	132.	132.	82.

CALVING INTERVAL 364.0
actually days open plus gestation length for cows conceiving in the year

CONCEPTION RATE 0.545
all services to cows conceiving or culled in the year

LIVE ANIMALS AT YEAR END *heifers*

35 COWS	8 BRED HEIFERS	19 YEARLINGS	20 CALVES

ECONOMIC SUMMARY

total production — 1967.78 HECTOLITRES OF MILK
kilograms / hectolitre — 3.82 FAT TEST 3.32 PROTEIN TEST

EXPENDITURES

FIXED COSTS	39935.00
fixed plus additional management	
FEED COSTS	
- COWS	31375.47
-YEARLINGS	6669.05
- CALVES	2646.46
SEMEN COSTS	9361.31
MILK SHIPPING COST	2636.83
TOTAL EXPENDITURES	92623.94

INCOME

1500.00 HL QUOTA MILK AT 53.38/HL	80073.88
467.78 HL EXCESS MILK AT 39.22/HL	18347.73
0.0 HL SURPLUS MILK AT .000/HL	0.00
SOLD ANIMALS	
5 COWS (PRODUCTION)	2600.00
3 COWS (FERTILITY)	1560.00
1 COWS (HEALTH)	300.00
1 COWS DEAD	0.0
0 YEARLINGS	0.0
0 OLD CALVES	0.0
14 BULL CALVES	800.00
0 HEIFER CALVES	0.0
0 SELECTED YOUNG SIRES	0.0
TOTAL INCOME	103681.56

TOTAL PROFIT 11057.63

Appendix B22 <u>ANNOTATED PRINTED OUTPUT FROM "STAT.ANAL"</u>

(FOR SAME RUN AS IN APPENDIX B1)

Annotations in *italics*

1: *<u>OUTPUT FROM FILE -STAT</u>*

<u>ANALYSIS OF COVARIANCE AND SLOPE TEST WITH YEARS (1-7) AS THE COVARIATE</u>

<u>MANAGEMENT LEVELS</u>

VARIABLE # 1 **<u>CALVING INTERVAL</u>**

GROUP 1
ANALYSIS OF COVARIANCE (42 OBSERVATIONS)

test if significant differences between herds in the treatment group

SOURCE	DF	SUM SQ	MEAN SQ	F-VALUE	PROB
TOTAL	41	1119.0			
SLOPES	5	27.637	5.527	0.2889	0.915
- slopes different ?					*not significant*
ERROR 1	30	574.06	19.14		
- for testing slopes					
MEANS	5	517.19	103.4	6.017	0.406E-03
- means different ?				*significant alpha .05*	
COMMON SLOPE	1	0.11841	0.1184	0.6888E-02	0.934
- common slope different from 0.0					*not significant*
ERROR 2	35	601.69	17.19		
- for testing means					

<u>STUDENT NEWMAN KUELS TEST - HERD MEAN 'S</u>

ALPHA=0.05 AND 35 DF

THERE ARE 2 HOMOGENOUS SUBSETS

(1, 3,
(5, 9, 11, 7,

HERD	MEAN	S.E.	SLOPE	S.E.	INTERCEPT
COMMON	365.244	0.836	0.027	0.320	365.137
1	370.251	1.567	0.293	0.827	369.078
3	369.721	1.567	-0.273	0.827	370.812
5	364.686	1.567	0.202	0.827	363.880
7	361.223	1.567	-0.360	0.827	362.663
9	362.846	1.567	0.704	0.827	360.031
11	362.735	1.567	-0.407	0.827	364.362

GROUP 2
ANALYSIS OF COVARIANCE (42 OBSERVATIONS)

SOURCE	DF	SUM SQ	MEAN SQ	F-VALUE	PROB
TOTAL	41	3536.0			
SLOPES	5	127.70	25.54	1.323	0.281
ERROR 1	30	578.97	19.30		
MEANS	5	2824.3	564.9	27.98	0.264E-10
COMMON SLOPE	1	5.0088	5.009	0.2481	0.622
ERROR 2	35	706.67	20.19		

STUDENT NEWMAN KUELS TEST - HERD MEAN 'S

ALPHA=0.05 AND 35 DF

THERE ARE 4 HOMOGENOUS SUBSETS

(101, 103,
(105,
(109, 111,
(107,

HERD	MEAN	S.E.	SLOPE	S.E.	INTERCEPT
COMMON	371.018	1.486	-0.173	0.347	371.708
101	383.245	1.698	-1.338	0.830	388.596
103	379.380	1.698	0.039	0.830	379.225
105	371.919	1.698	-1.386	0.830	377.464
107	359.574	1.698	0.275	0.830	358.472
109	366.702	1.698	0.621	0.830	364.219
111	365.289	1.698	0.752	0.830	362.280

GROUP 3
ANALYSIS OF COVARIANCE (42 OBSERVATIONS)

SOURCE	DF	SUM SQ	MEAN SQ	F-VALUE	PROB
TOTAL	41	4838.0			
SLOPES	5	70.204	14.04	0.5676	0.724
ERROR 1	30	742.16	24.74		
MEANS	5	3973.2	794.6	34.24	0.156E-11
COMMON SLOPE	1	52.391	52.39	2.257	0.142
ERROR 2	35	812.36	23.21		

STUDENT NEWMAN KUELS TEST - HERD MEAN 'S

ALPHA=0.05 AND 35 DF

THERE ARE 3 HOMOGENOUS SUBSETS

(102,
(106, 104,
(110, 112, 108,

HERD	MEAN	S.E.	SLOPE	S.E.	INTERCEPT
COMMON	379.749	1.729	0.558	0.372	377.515
102	394.637	1.821	0.661	0.940	391.993
104	383.377	1.821	0.261	0.940	382.331
106	388.329	1.821	1.277	0.940	383.222
108	367.928	1.821	1.413	0.940	362.276
110	373.234	1.821	-0.457	0.940	375.062
112	370.992	1.821	0.195	0.940	370.210

GROUP 4
ANALYSIS OF COVARIANCE (42 OBSERVATIONS)

SOURCE	DF	SUM SQ	MEAN SQ	F-VALUE	PROB
TOTAL	41	17545.			
SLOPES	5	421.60	84.32	1.478	0.226
ERROR 1	30	1710.9	57.03		
MEANS	5	14811.	2962.	48.62	0.110E-13
COMMON SLOPE	1	601.22	601.2	9.867	0.341E-02
ERROR 2	35	2132.5	60.93		

STUDENT NEWMAN KUELS TEST - HERD MEAN 'S

ALPHA=0.05 AND 35 DF

THERE ARE 3 HOMOGENOUS SUBSETS

(4, 2, 6,
(12, 10,
(8,

HERD	MEAN	S.E.	SLOPE	S.E.	INTERCEPT
COMMON	391.936	3.254	1.892	0.602	384.368
2	409.667	2.950	2.165	1.427	401.008
4	414.743	2.950	0.630	1.427	412.224
6	405.264	2.950	3.999	1.427	389.269
8	365.114	2.950	0.080	1.427	364.794
10	375.516	2.950	0.590	1.427	373.154
12	381.312	2.950	3.886	1.427	365.769

FINAL - OVERALL

test if significant differences between groups

ANALYSIS OF COVARIANCE (168 OBSERVATIONS)

SOURCE	DF	SUM SQ	MEAN SQ	F-VALUE	PROB
TOTAL	167	44032.			
SLOPES	3	435.78	145.3	0.8811	0.452
ERROR 1	160	26379.	164.9		
MEANS	3	16994.	5665.	34.43	0.580E-16
COMMON SLOPE	1	222.95	222.9	1.355	0.246
ERROR 2	163	26815.	164.5		

STUDENT NEWMAN KUELS TEST - GROUP MEAN 'S

ALPHA=0.05 AND 163 DF

THERE ARE 4 HOMOGENOUS SUBSETS

(4,
(3,
(2,
(1,

GROUP	MEAN	S.E.	SLOPE	S.E.	INTERCEPT
COMMON	376.986	1.261	0.576	0.495	374.682
1	365.244	1.979	0.027	0.991	365.137
2	371.018	1.979	-0.173	0.991	371.708
3	379.749	1.979	0.558	0.991	377.515
4	391.936	1.979	1.892	0.991	384.368

VARIABLE # 2 NET INCOME

\- .
\- .
\- .
\- . *SAME ANALYSIS AS VARIABLE #1*
\- .

ANALYSIS COMPLETE

2: <u>OUTPUT FROM FILE -PLOT</u>

ALL GROUPS AND COMMON SLOPE

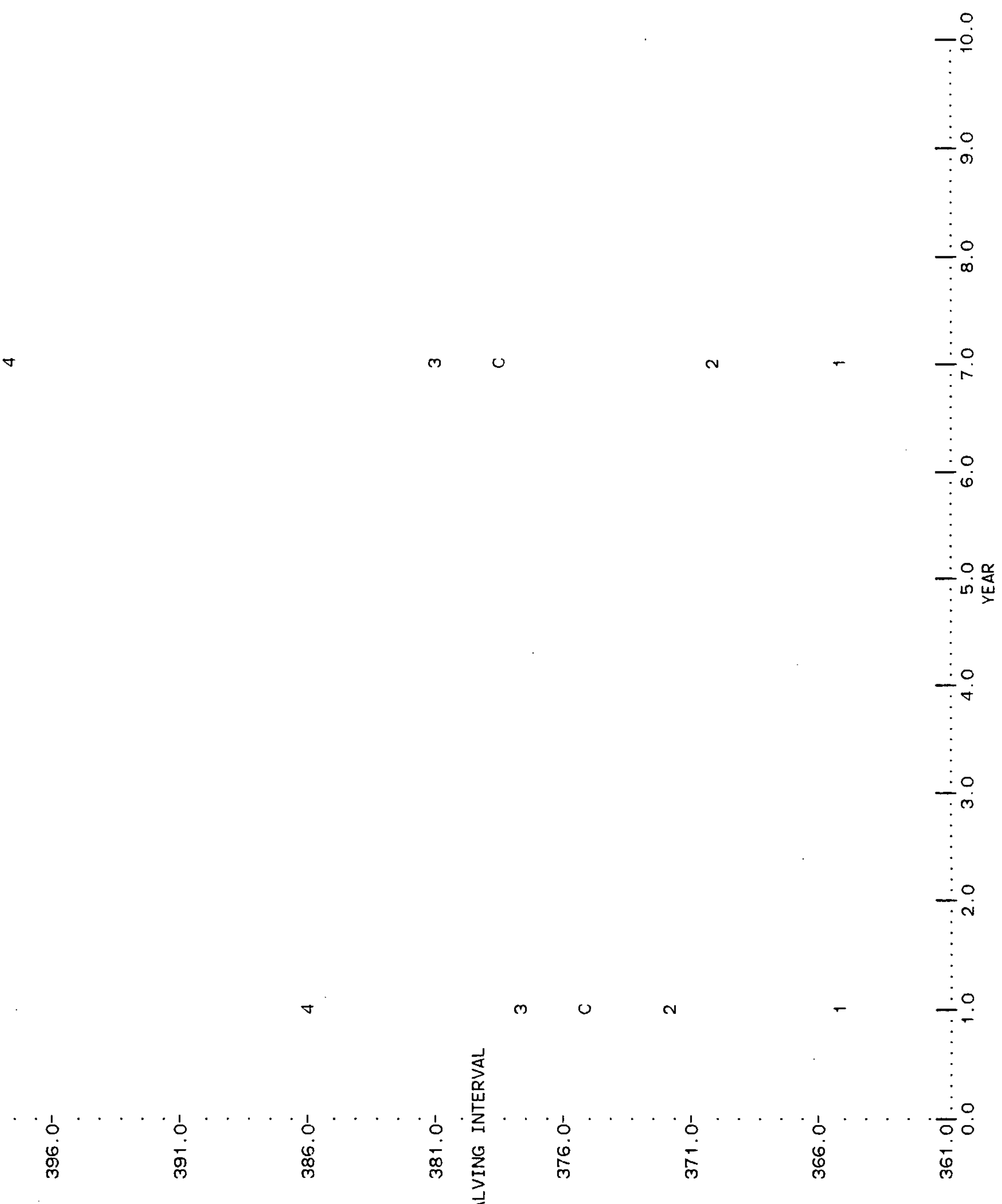
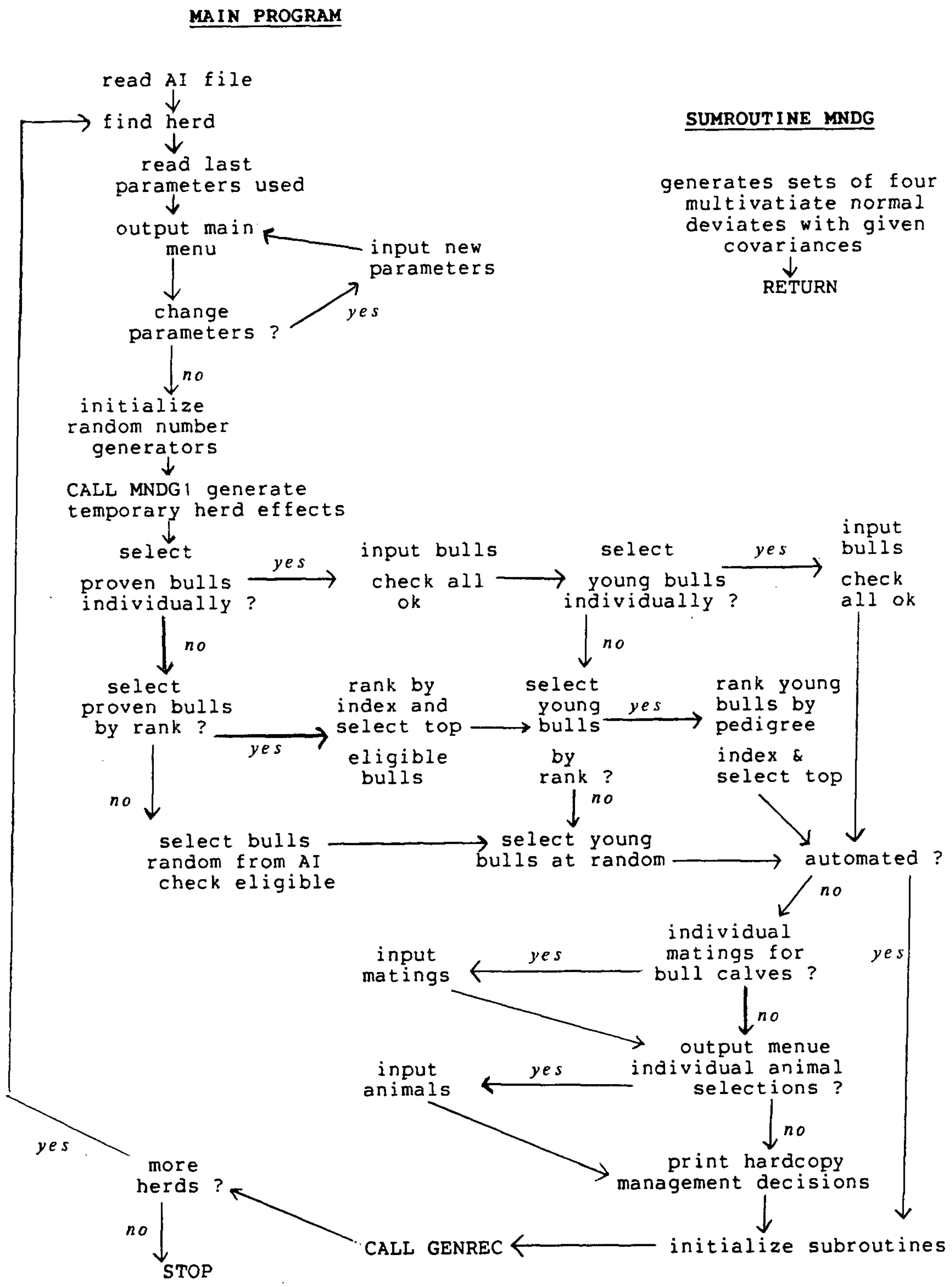
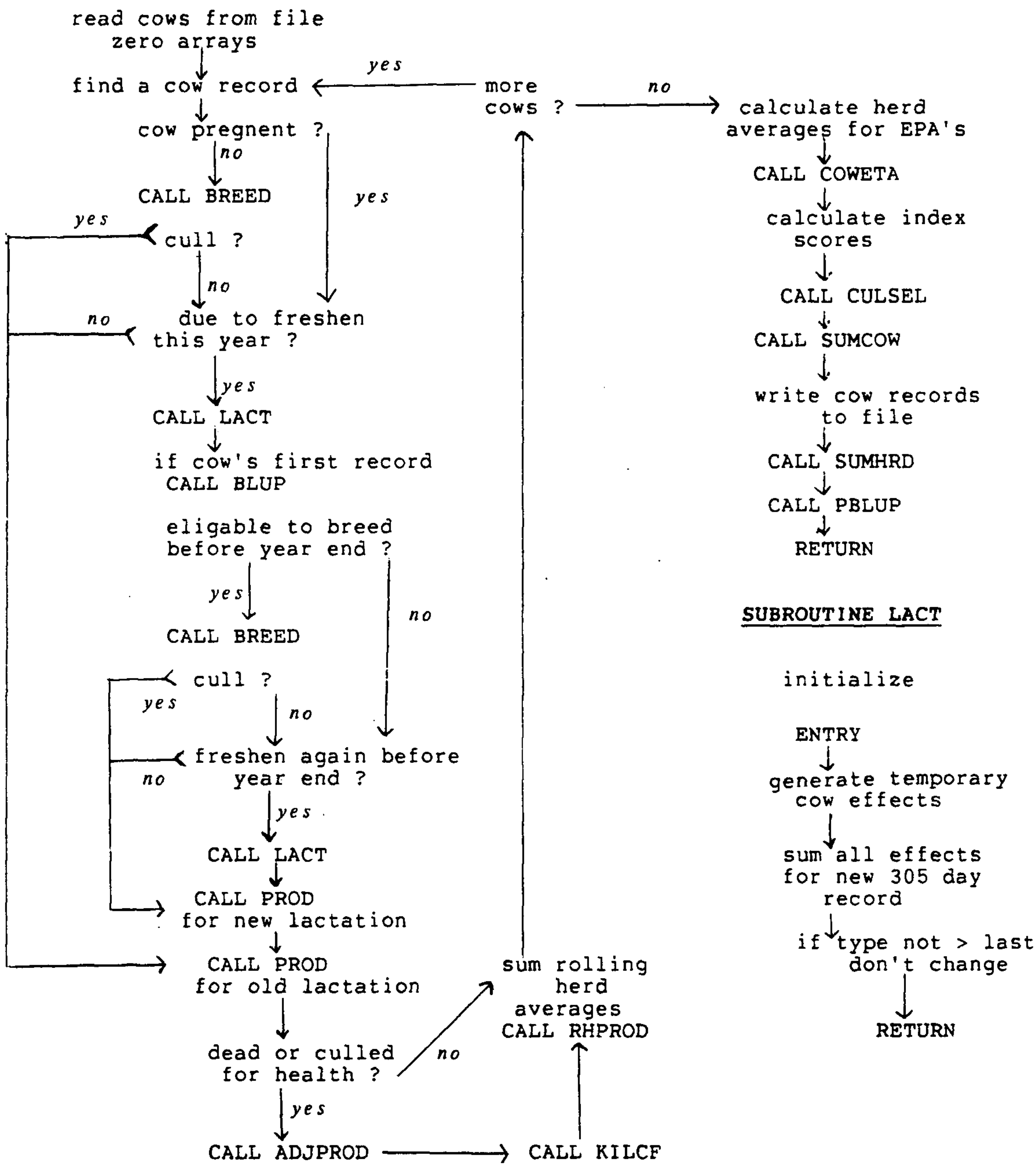


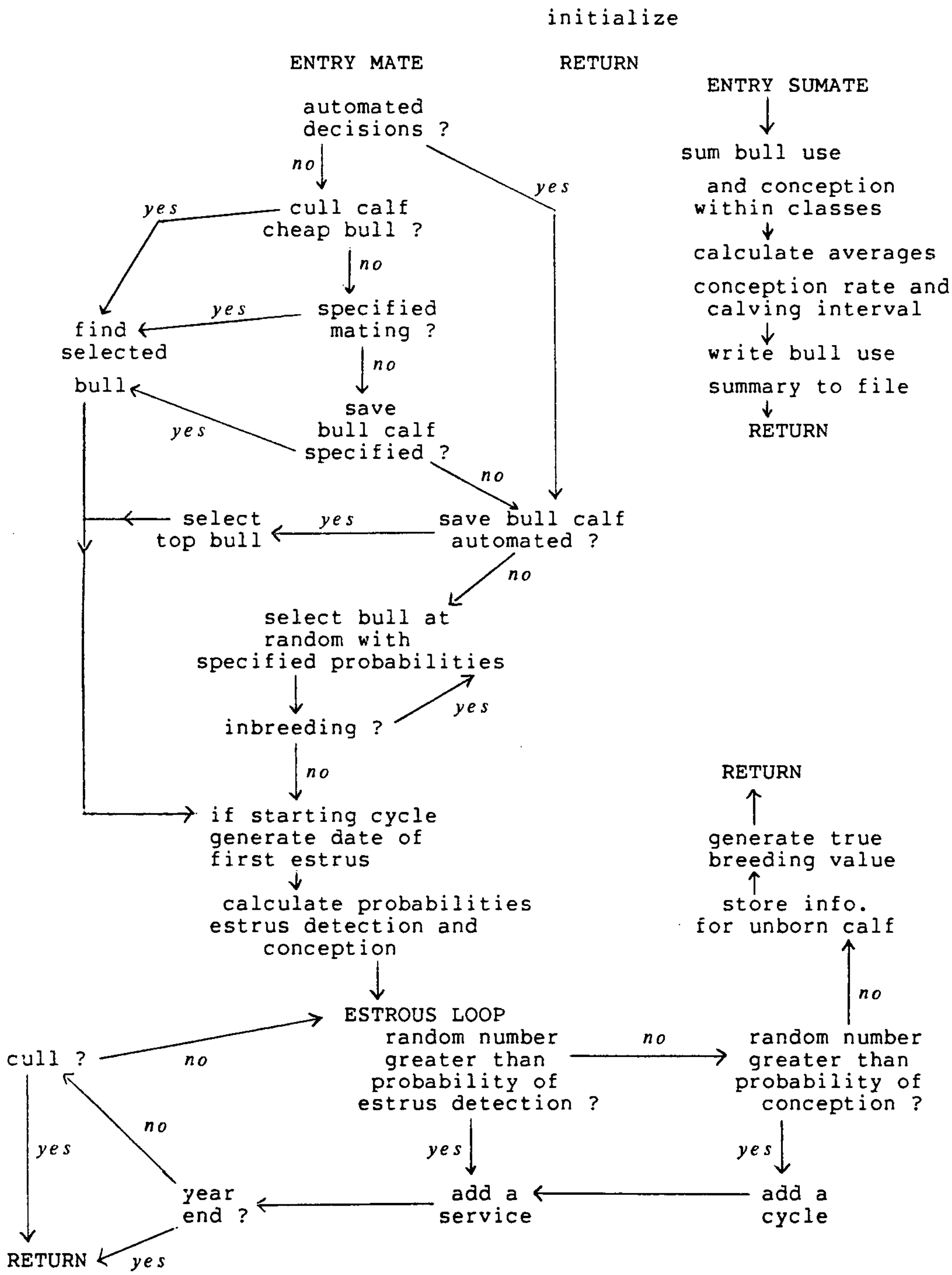
Figure C.1

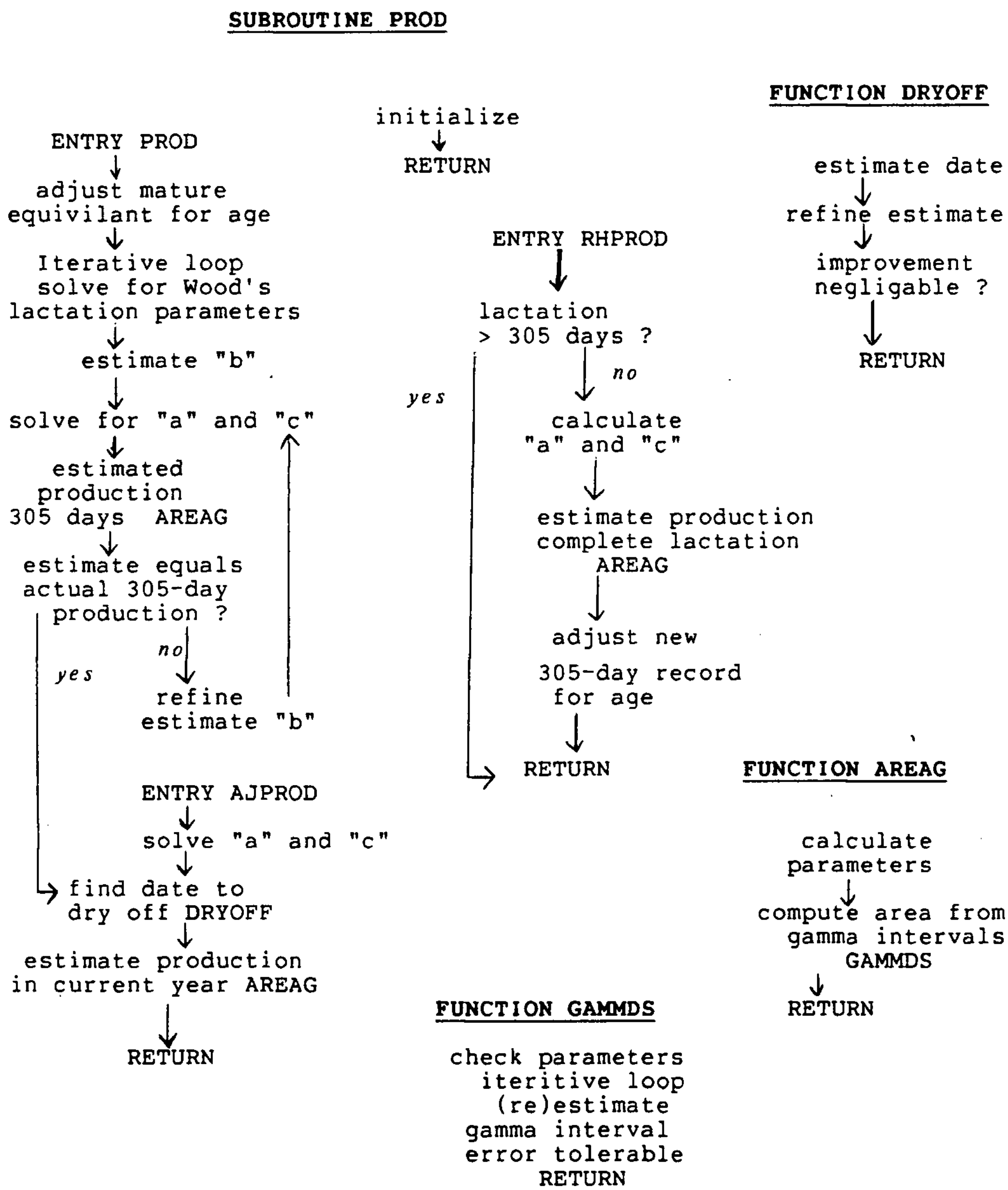

MAIN PROGRAM
read AI file
find herd
read last parameters used
output main menu
input new parameters
change parameters ?
yes
no
initialize random number generators
CALL MNDG1 generate temporary herd effects
select proven bulls individually ?
yes
input bulls check all ok
select young bulls individually ?
yes
input bulls check all ok
no
select proven bulls by rank ?
yes
rank by index and select top eligible bulls
select young bulls by rank ?
yes
rank young bulls by pedigree index & select top
no
select bulls random from AI check eligible
select young bulls at random
automated ?
no
yes
individual matings for bull calves ?
yes
input matings
no
output menue individual animal selections ?
yes
input animals
no
print hardcopy management decisions
initialize subroutines
CALL GENREC
more herds ?
yes
no
STOP
SUMROUTINE MNDG
generates sets of four multivatiate normal deviates with given covariances
RETURN

SUBROUTINE GENREC



SUBROUTINE BREED




SUBROUTINE PROD
ENTRY PROD
adjust mature equivilant for age
Iterative loop solve for Wood's lactation parameters
estimate "b"
solve for "a" and "c"
estimated production 305 days AREAG
estimate equals actual 305-day production ?
yes
no
refine estimate "b"
ENTRY AJPROD
solve "a" and "c"
find date to dry off DRYOFF
estimate production in current year AREAG
RETURN
initialize
RETURN
ENTRY RHPROD
lactation > 305 days ?
yes
no
calculate "a" and "c"
estimate production complete lactation AREAG
adjust new 305-day record for age
RETURN
FUNCTION DRYOFF
estimate date
refine estimate
improvement negligable ?
RETURN
FUNCTION AREAG
calculate parameters
compute area from gamma intervals GAMMDS
RETURN
FUNCTION GAMMDS
check parameters
iteritive loop
(re)estimate
gamma interval
error tolerable
RETURN

SUBROUTINE ETA

initialize
↓
RETURN

ENTRY COWETA
↓
adjust cows deviations for herd and population mean
↓
calculate EPA's
↓
ENTRY CLFETA
↓
ENTRY YNGETA
↓
find dam if alive update dams ETA
↓
update sire's ETA
↓
calculate animal's ETA
↓
RETURN

SUBROUTINE SUMCOW

add the animal's true BV to the total for the appropriate animal class
↓
find lactation dates
↓
calculate BCA's
↓
CALL WRTLIN
↓
RETURN

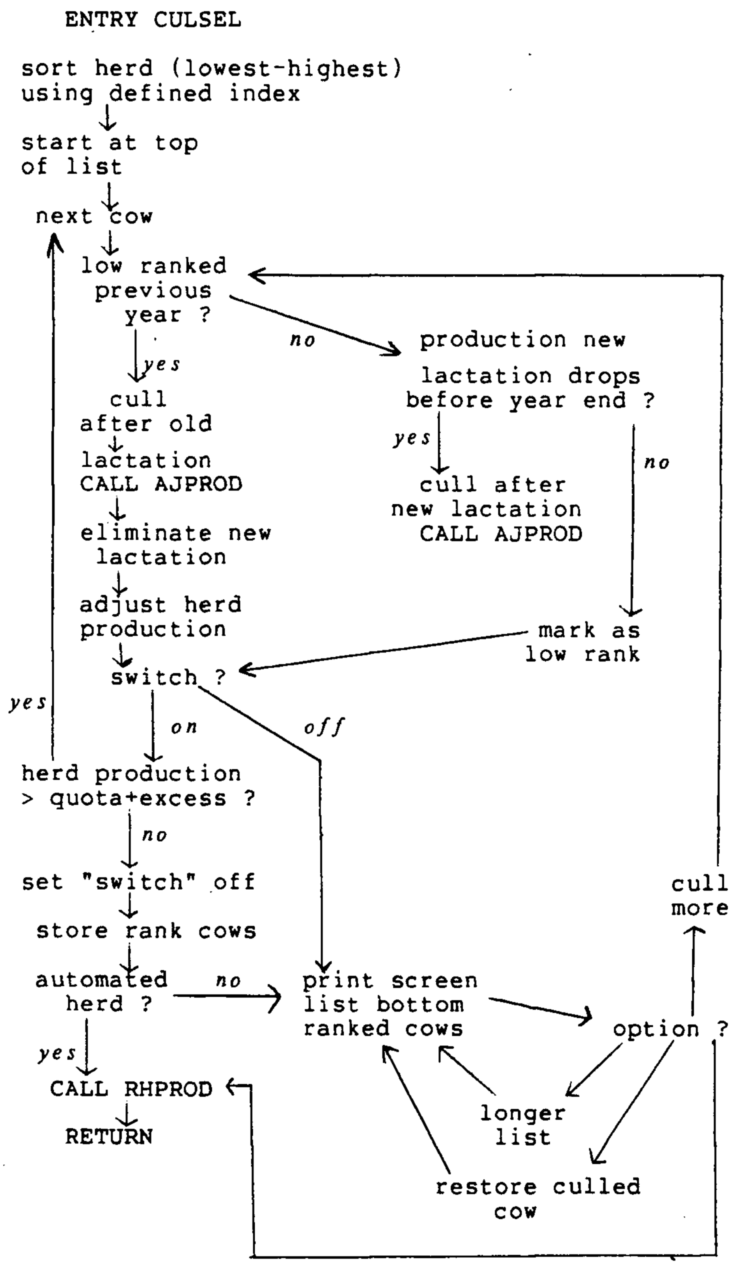
SUBROUTINE SUMHRD

calculate milk price and income
↓
CALL YNGOUT
↓
CALL SUMATE
↓
calculate income from culled animals
↓
calculate average BV's for each class of animal
↓
calculate rolling herd averages
↓
write herd and summaries to files
↓
automated herd ? (*yes* → RETURN)
↓ *no*
write production and economic summaries to printer and screen
↓
RETURN

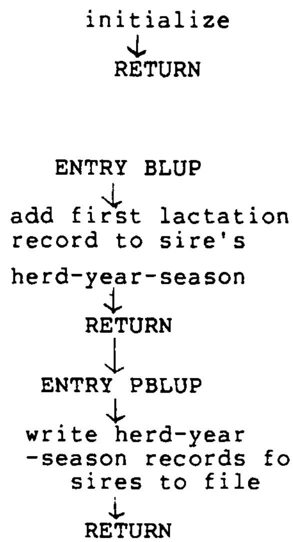
SUBROUTINE WRTLIN

write to printer cows preformance and record in appropriate format

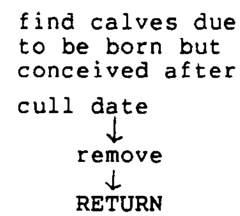
SUBROUTINE SELEC



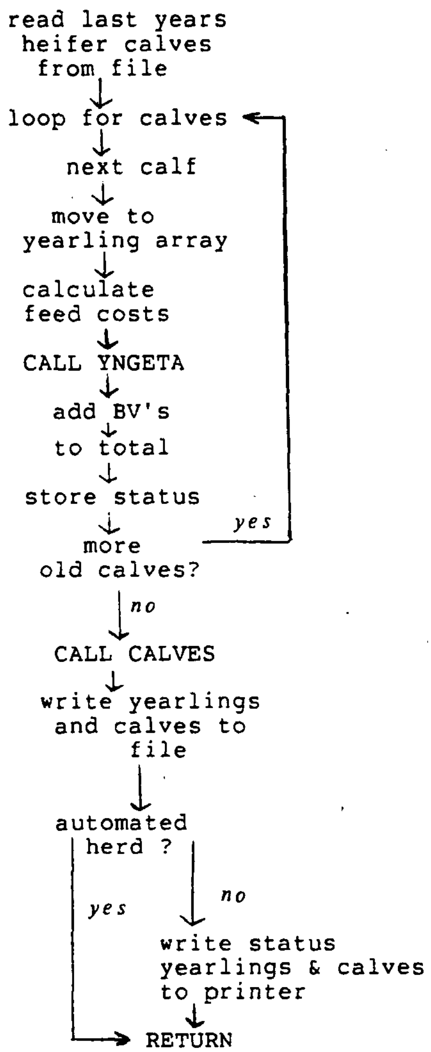
SUBROUTINE BLUP



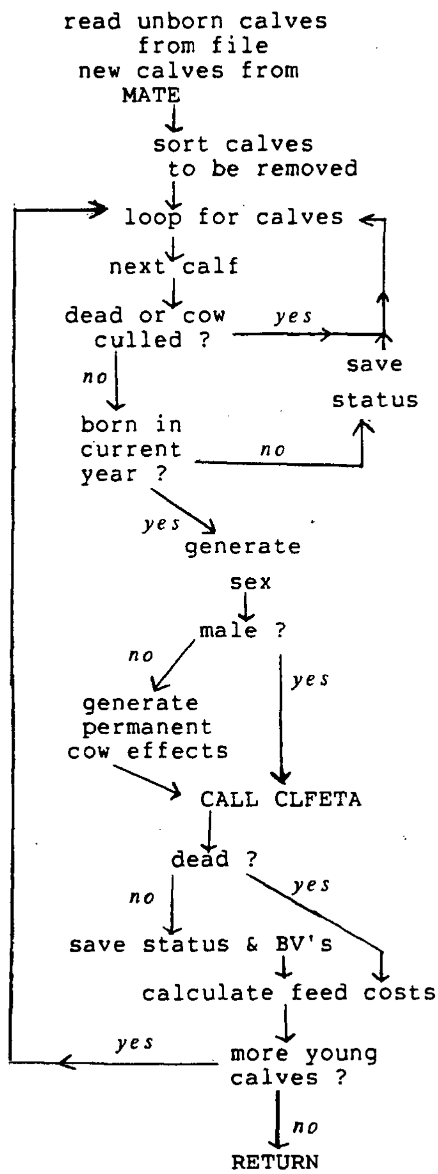
SUBROUTINE KILCF



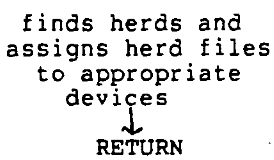
SUBROUTINE YOUNG



SUBROUTINE CALVES



SUBROUTINE FILES



```
C   *****************************************************************
C   *                                                               *
C   *                  Program -   SETUP                            *
C   *                                                               *
C   *      Sets up herds and A.I. unit ready to run year -1.        *
C   *                                                               *
C   *          Population size and age distribution are set         *
C   *     interactively when the program is run.  All other parameters *
C   *     are determined at compile time, so check all initialized  *
C   *     parameters before compiling.                              *
C   *                                                               *
C   *****************************************************************
C   *
C   *
C   *
      DIMENSION AI(30,500), HRDCOW(40,150), HRDYLG(18,75),
     1          CALF(18,75), UCALF(8,150),
     2          RNE(4), RNG(4), EWK(4),GWK(4), IDIST(6),
     3          IAGE(150), BAGE(500)
C   *
C   *    Parameters needed to generate genetic and environmental    *
C   *    effects for production traits                              *
C   *
      REAL*4 ESIGMA(10)/1., .83, 1., .96, .78, 1., .2, .2, .2, 1./,
     1       GSIGMA(10)/1., .54, 1., .7, .81, 1., 0., -.15, 0., 1./,
     2       GSD(4)/413., 18.2, 12.8, 1.66/, PCESD(4)/393., 12.6,
     3       11.9, 1.64/
C   *
C   *    Other biological parameters                                *
C   *
      REAL *4  BFSD/0.02/, BFM/0.7/, HDCSD/0.01/, HDCM/0.75/,
     1         FERSD/0.01/, FERTM/0.75/,
     2         DAYR/365./, UYR/1.0/
C   *
C   *    Management decision options                                *
C   *
      REAL *4  PARM(25)/0.0, 0.0, -2., 2.2, 3.2, 0.5, 0.5, 0.0, 0.0,
     1        305., 5., 0.0, 0.0, 0.0, 0.0, 7., 12., 10.2, 100.2, 3.3,
     2        30.02, 10.2, 1500., 500., 1./
C   *
      LOGICAL*1 TRUE /T/, FALSE /F/, CONT /F/, FINISH
C   *
C   *   READ PARAMETERS                                             *
C   *
   10 WRITE (6,280)
      READ (5,290,ERR=10) NSH, NGSH, NGGH, NCHD, NBULLS, ISEED
      IF (NBULLS .LE. 0) GO TO 10
      IF (NSH .LE. 0) GO TO 10
      IF (NCHD .LE. 0) GO TO 10
      I = IRAND(0)
      IF (ISEED .LE. 0) ISEED = IRAND(1000)
      GO TO 30
   20 WRITE (6,340) NCHD
   30 WRITE (6,300)
      READ (5,290) IDIST
      NT = 0
      DO 40 I = 1, 6
        NT = NT + IDIST(I)
   40 CONTINUE
      IF (NT .NE. NCHD) GO TO 20
      WRITE (6,320) NSH, NGSH, NCHD, NGGH, NBULLS, ISEED, IDIST
      READ (5,330) CONT
```

```
      IF (CONT) GO TO 10
C
C  *    set cow ages for selected distribution                               *
C
      IT = 0
      DO 60 I = 1, 6
        K = 7 - I
        N = IDIST(K)
        IDIST(K) = 0.
        IF (N .LE. 0) GO TO 60
        DO 50 J = 1, N
          IT = IT + 1
          IAGE(IT) = I * 100
   50   CONTINUE
   60 CONTINUE
      IF (NCHD .GT. 150) GO TO 10
      NTSH = NSH * NGSH
      IF (NTSH .GT. 100) GO TO 10
C
C  INITIALIZE RANDOM NUMBER GENERATORS
C
      SEED = ISEED
      I = IRAND(-ISEED)
      F = RAND(SEED)
      FN = RANDN(SEED)
      CALL MNDG(ISEED, ESIGMA, RNE, EWK)
      CALL MNDG(ISEED, GSIGMA, RNG, GWK)
C  *
C  *    Set numbers and ages of bulls                                         *
C  *
      NOAI = 2 * NBULLS
      NUAI = 3 * NOAI
      NEAI = NUAI + NBULLS
      NPAI = NEAI + UYR * NBULLS
      NTAI = NPAI + NBULLS
      IOA = -1 * NTAI / NBULLS
      NWY = 0
      DO 67 I = 1, NTAI
        IF (NWY .LT. NBULLS) GO TO 66
        NWY = 0
        IOA = IOA + 1
   66   NWY = NWY + 1
        BAGE(I) = IOA + FRAND(0.0)
   67 CONTINUE
      CALL ISORT (BAGE, 1, 500, 1, NTAI, 1, 3, 0)
C  *                                                                          *
C  *      Generate A.I. Sires                                                 *
C  *                                                                          *
      DO 100 I = 1, NTAI
        CALL MNDG1 (ISEED, GSIGMA, RNG, GWK)
        DO 70 J = 1, 4
          AI(J+1,I) = 0.0
          AI(J+10,I) = 0.0
          AI(J+18,I) = 0.0
          AI(J+18,I) = 0.0
          AI(J+22,I) = 0.0
          AI(J+26,I) = 0.0
          AI(J+14,I) = RNG(J) * GSD(J)
   70   CONTINUE
        AI(1,I) = 9000 + I
   80   FERTB = BFM + 1 - EXP (BFSD * FRANDN(0.0))
        IF (FERTB .GT. 1.) GO TO 80
```

```
        AI(6,I) = FERTB
        AI(7,I) = 15.
        AI(8,I) = BAGE(I)
        AI(9,I) = 0.
        AI(10,I) = 0.0
        AI(11,I) = 10.
        AI(12,I) = 10. * FERTB * (FERTM + FRANDN(0.0) * FERSD / 10.)
  100 CONTINUE
      CALL FTNCMD('ASSIGN 4=KINN:AI.U;')
      NB = 0
      WRITE (4) NB, NB, NUAI, NEAI, NPAI, NTAI
      CALL WRTMAT(AI, 30, NTAI, 4)
  105 CALL FTNCMD('ASSIGN 9=KINN:CODES;')
C
C  *   Generate cow and unborn calf                                        *
C
      NCG = NCHD / 2
      NCOW = 2 * NCG
      VMC = SQRT (0.5)
      DO 270 L = 1, NTSH, NGSH
        INC = 0
        DO 220 I = 1, NCG
          N = I + INC
          ASSIGN 148 TO ISWIT
  110 M = IAGE(N) / 100
      IDIST(M) = IDIST(M) + 1
      IBL = IRAND (NOAI)
          HRDCOW(1,N) = IAGE(N) + IDIST(M)
          HRDCOW(2,N) = 9000 + IBL
          HRDCOW(3,N) = 0.0
          HRDCOW(4,N) = IRAND(365)
          HRDCOW(5,N) = 1.
          HRDCOW(6,N) = 0.0
          HRDCOW(7,N) = 1.0
          HRDCOW(8,N) = 0.0
          HRDCOW(9,N) = HRDCOW(4,N) + DAYR
          HRDCOW(10,N) = 1.0
  120     HRDCOW(39,N) = HDCM + 1 - EXP (HDCSD * FRANDN(0.0))
          IF (HRDCOW(39,N) .GT. 1.) GO TO 120
  130     HRDCOW(40,N) = FERTM + 1 - EXP (FERSD * FRANDN(0.0))
          IF (HRDCOW(40,N) .GT. 1.) GO TO 130
          CALL MNDG1(ISEED, GSIGMA, RNG, GWK)
          CALL MNDG1(ISEED, ESIGMA, RNE, EWK)
          DO 140 J = 1, 4
            HRDCOW(J + 10,N) = RNE(J) * PCESD(J)
            HRDCOW(J + 14,N) = RNG(J) * GSD(J) * VMC + 0.25 *
     1       AI(J+14,IBL)
            HRDCOW(J + 18,N) = 0.0
            HRDCOW(J + 22,N) = 0.0
            HRDCOW(J + 26,N) = 0.0
            HRDCOW(J + 30,N) = 0.0
            HRDCOW(J + 34,N) = 0.0
  140     CONTINUE
          IST = 2
          BYA = FRAND(0.0)
          IF (BYA .LT. 0.3) IST = 1
          UCALF(1,N) = IST
          IBL = IRAND (NBULLS) + 5 * NBULLS
          UCALF(2,N) = 9000 + IBL
          UCALF(3,N) = HRDCOW(1,N)
          UCALF(4,N) = HRDCOW(9,N)
          CALL MNDG1(ISEED, GSIGMA, RNG, GWK)
```

```
      DO 145 J = 1, 4
        UCALF(J+4,N) = RNG(J) * GSD(J) * VMC + 0.5 *
     1    (HRDCOW(J+14,N) + AI(J+14,IBL))
145   CONTINUE
      GO TO ISWIT, (150, 148)
148   ASSIGN 150 TO ISWIT
      INC = INC + 1
        N = N + 1
      IF (N .GT. NCOW) GO TO 150
        GO TO 110
C *
C *      GENERATE YEARLING                                              *
C *
150     HRDYLG(1,I) = 700 + I
        IBL = IRAND (NOAI) + NOAI
        HRDYLG(2,I) = 9000 + IBL
        HRDYLG(3,I) = 0.0
        HRDYLG(4,I) = FRAND(0.0) * 365.
160     HRDYLG(5,I) = HDCM + 1 - EXP (HDCSD * FRANDN(0.0))
        IF (HRDYLG(5,I) .GT. 1.) GO TO 160
170     HRDYLG(6,I) = FERTM + 1 - EXP (FERSD * FRANDN(0.0))
        IF (HRDYLG(6,I) .GT. 1.) GO TO 170
        CALL MNDG1(ISEED, GSIGMA, RNG, GWK)
        CALL MNDG1(ISEED, ESIGMA, RNE, EWK)
        DO 180 J = 1, 4
          HRDYLG(J + 6,I) = RNE(J) * PCESD(J)
          HRDYLG(J + 10,I) = RNG(J) * GSD(J) * VMC + 0.25 *
     1    AI(J+14,IBL)
          HRDYLG(J + 14,I) = 0.0
180     CONTINUE
C *
C *     GENERATE CALF                                                   *
C *
        CALF(1,I) = 800 + I
        IBL = IRAND (NBULLS) + 4 * NBULLS
        CALF(2,I) = 9000 + IBL
        CALF(3,I) = 0.0
        CALF(4,I) = FRAND(0.0) * 365.
190     CALF(5,I) = HDCM + 1 - EXP (HDCSD * FRANDN(0.0))
        IF (CALF(5,I) .GT. 1.) GO TO 190
200     CALF(6,I) = FERTM + 1 - EXP (FERSD * FRANDN(0.0))
        CALL MNDG1(ISEED, GSIGMA, RNG, GWK)
        CALL MNDG1(ISEED, ESIGMA, RNE, EWK)
        DO 210 J = 1, 4
          CALF(J + 6,I) = RNE(J) * PCESD(J)
          CALF(J + 10,I) = RNG(J) * GSD(J) * VMC + 0.25 *
     1    AI(J+14,IBL)
          CALF(J + 14,I) = 0.0
210     CONTINUE
220   CONTINUE
      IF (NGSH .LT. 1) GO TO 240
C *
C *   WRITE THE STUDENTS HERDS UNFORMATTED IN THE APPROPRIATE FILES    *
C *
      DO 230 I = 1, NGSH
        N = I - 1
        IHRD = L + N
        PARM(1) = IHRD
        PARM(2) = 500
        PARM(25) = 1.
        WRITE (9'IHRD,310) IHRD, PARM(2)
        CALL FILE(IHRD)
```

```
          WRITE (12) PARM
          WRITE (12) NCOW
          CALL WRTMAT(HRDCOW, 40, NCOW, 12)
          WRITE (12) NCG
          CALL WRTMAT(HRDYLG, 18, NCG, 12)
          WRITE (12) NCG, NCOW
          CALL WRTMAT(CALF, 18, NCG, 12)
          CALL WRTMAT(UCALF, 8, NCOW, 12)
  230   CONTINUE
  240   IF (NGGH .LT. 1) GO TO 270
C  *
C  *    WRITE THE CONTROL HERDS UNFORMATTED IN THE APPROPRIATE         *
C  *         GROUP FILE                                                *
C  *
        DO 260 I = 1, NGGH
          IHRD = I * 100 + L
          CALL FILED(IHRD)
          PARM(1) = IHRD
          PARM(2) = 500
          WRITE (9,310) IHRD, PARM(2)
          PARM(25) = NSH
          WRITE (12) PARM
  250     WRITE (12) NCOW
          CALL WRTMAT(HRDCOW, 40, NCOW, 12)
          WRITE (12) NCG
          CALL WRTMAT(HRDYLG, 18, NCG, 12)
          WRITE (12) NCG, NCOW
          CALL WRTMAT(CALF, 18, NCG, 12)
          CALL WRTMAT(UCALF, 8, NCOW, 12)
  260   CONTINUE
  270 CONTINUE
  280 FORMAT ('    ENTER THE NUMBER OF STUDENTS AND THE NUMBER',
     1        ' OF HERDS EACH', /,
     *        '       (MAXIMUM 100 STUDENT HERDS)', /,
     2        '    THE NUMBER OF CONTROL GROUPS (MAXIMUM 4),   ', /,
     3        'THE NUMBER OF COWS PER HERD (MAXIMUM 150)', / ,
     4        '    THE NUMBER OF YOUNG BULLS TO ADD EACH YEAR (AND AN',
     5        ' INTEGER SEED - OPTIONAL)')
  290 FORMAT (10I10)
  300 FORMAT (' ENTER THE NUMBER OF COWS OF EACH AGE 2-7')
  310 FORMAT (I10, F10.0)
  320 FORMAT (I10, ' STUDENTS', I6, ' HERDS/STUDENT', I6, ' COWS/HERD',
     1        /, I10, ' REPLICATE GROUPS', I6, ' BULLS', I10, ' SEED',
     2        //, ' AGE DISTRIBUTION', /, 7X, '2', 7X, '3', 7X, '4', 7X,
     3        '5', 7X, '6', 7X, '7', //, 6I8, /,
     4        'ENTER "T" IF ERROR OR RETURN IF OK')
  330 FORMAT (L1)
  340 FORMAT ('ERROR - COWS SHOULD ADD TO', I5)
      STOP
      END


      SUBROUTINE WRTMAT(RMAT, ICOL, IROW, INP)
C  ********************************************************************
C  *    This subroutine writes and reads two dimensional matrixes.    *
C  ********************************************************************
      DIMENSION RMAT(ICOL,IROW)
      WRITE (INP) RMAT
      RETURN
```

```
C  *
      ENTRY REAMAT (RMAT, ICOL, IROW, INP)
      READ (INP) RMAT
      RETURN
      END



      FUNCTION RANDT(ISEED)
C     --------------
C         THIS FUNCTION GENERATES UNIFORM (0,1) RANDOM NUMBERS
C
      DOUBLE PRECISION Z, DN1MOD, DN1

      DATA DN1MOD /2147483647.D0/, DN1 /Z3920000000000000/
      Z = ISEED
      Z = DMOD(16807.D0*Z,DN1MOD)
      RANDT = Z * DN1
      ISEED = Z
      RETURN
      END



      FUNCTION RN(IX)
C
C         This function generates pairs of normal (0,1) random deviates,
C         using a modification of the box-mueller method.
C
      DATA I /1/
      IF (I .NE. 1) GO TO 30
      I = 2
   10 U = 2. * RANDT(IX) - 1.
      V = 2. * RANDT(IX) - 1.
      W = U * U + V * V
      IF (W - 1.) 20, 20, 10
   20 W = SQRT(-2.*ALOG(W)/W)
      RN = U * W
      RETURN
   30 I = 1
      RETURN
      END



      SUBROUTINE DCSIG(SIGMA, UL, A, B)
C     ----------------
C   C
C         This subroutine decomposes the symmetric matrix of variances-
C         covariances into its factor (square root).  DCSIG is called by
C         MNDG for generating multivariate normal deviates.
C
      DIMENSION SIGMA(1), UL(1)
      DATA ZERO, ONE, FOUR, SIXTN, SIXTH /0.0, 1., 4., 16., .0625/
      A = ONE
      B = ZERO
      R1N = ONE / (4*SIXTN)
      IP = 1
      DO 90 I = 1, 4
```

```
      IQ = IP
      IR = 1
      DO 80 J = 1, I
        X = SIGMA(IP)
        IF (J .EQ. 1) GO TO 20
        DO 10 L = IQ, IP1
          X = X - UL(L) * UL(IR)
          IR = IR + 1
 10     CONTINUE
 20     IF (I .NE. J) GO TO 60
        A = A * X
        IF (SIGMA(IP) + X*R1N .LE. SIGMA(IP)) GO TO 100
 30     IF (ABS(A) .LE. ONE) GO TO 40
        A = A * SIXTH
        B = B + FOUR
        GO TO 30
 40     IF (ABS(A) .GE. SIXTH) GO TO 50

        A = A * SIXTN
        B = B - FOUR
        GO TO 40
 50     UL(IP) = ONE / SQRT(X)
        GO TO 70
 60     UL(IP) = X * UL(IR)
 70     IP1 = IP
        IP = IP + 1
        IR = IR + 1
 80   CONTINUE
 90 CONTINUE
    GO TO 120
100 WRITE (6,110)
110 FORMAT ('1', 'TROUBLE')
120 RETURN
    END



    SUBROUTINE MNDG(ISEED, SIGMA, RVEC, WKVEC)
C   ---------------
C
C       This subroutine generates sets of four multivariate normal
C       deviates, distributed with zero mean and covariances matrix
C       sigma.  To produce genetically correlated normal deviates,
C       GSIGMA is entered for SIGMA, for environmentally correlated
C       normal deviates, ESIGMA is entered.
C
    DIMENSION SIGMA(1), RVEC(4), WKVEC(4)
    CALL DCSIG(SIGMA, SIGMA, A, B)
    L = 0
    DO 10 I = 1, 4
      L = L + I
 10 SIGMA(L) = 1.0 / SIGMA(L)
    GO TO 20
    ENTRY MNDG1(ISEED,SIGMA,RVEC,WKVEC)
 20 DO 30 I = 1, 4
 30 WKVEC(I) = RN(ISEED)
    L = 1
    DO 50 II = 1, 4
      RVEC(II) = 0.0
      DO 40 I = 1, II
        RVEC(II) = RVEC(II) + DBLE(WKVEC(I)) * DBLE(SIGMA(L))
```

```
40   L = L + 1
50 CONTINUE
   RETURN
   END


   SUBROUTINE FILE(IHRD)
C
C   This subroutine finds the file for student herds
C
   GO TO (10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130,
  1140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260,
  2270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390,
  3400, 410, 420, 430, 440, 450, 460, 470, 480), IHRD
   GO TO 1060
10 CALL FTNCMD('ASSIGN 12=KINN:H1IN;')
   RETURN
20 CALL FTNCMD('ASSIGN 12=KINN:H2IN;')
   RETURN
30 CALL FTNCMD('ASSIGN 12=KINN:H3IN;')
   RETURN
40 CALL FTNCMD('ASSIGN 12=KINN:H4IN;')
   RETURN
50 CALL FTNCMD('ASSIGN 12=KINN:H5IN;')
   RETURN
60 CALL FTNCMD('ASSIGN 12=KINN:H6IN;')
   RETURN
70 CALL FTNCMD ('ASSGN 12=KINN:H7IN;')
   RETURN
80 CALL FTNCMD ('ASSIGN 12=KINN:H8IN;')
   RETURN
90 CALL FTNCMD ('ASSIGN 12=KINN:H9IN;')
   RETURN
100 CALL FTNCMD ('ASSIGN 12=KINN:H10IN;')
   RETURN
110 CALL FTNCMD ('ASSIGN 12=KINN:H11IN;')
   RETURN
120 CALL FTNCMD ('ASSIGN 12=KINN:H12IN;')
   RETURN
130 CALL FTNCMD ('ASSIGN 12=KINN:H13IN;')
   RETURN
140 CALL FTNCMD ('ASSIGN 12=KINN:H14IN;')
   RETURN
150 CALL FTNCMD ('ASSIGN 12=KINN:H15IN;')
   RETURN
160 CALL FTNCMD ('ASSIGN 12=KINN:H16IN;')
   RETURN
170 CALL FTNCMD('ASSIGN 12=KINN:H17IN;')
   RETURN
180 CALL FTNCMD('ASSIGN 12=KINN:H18IN;')
   RETURN
190 CALL FTNCMD('ASSIGN 12=KINN:H19IN;')
   RETURN
200 CALL FTNCMD('ASSIGN 12=KINN:H20IN;')
   RETURN
210 CALL FTNCMD('ASSIGN 12=KINN:H21IN;')
   RETURN
220 CALL FTNCMD('ASSIGN 12=KINN:H22IN;')
   RETURN
230 CALL FTNCMD('ASSIGN 12=KINN:H23IN;')
   RETURN
```

```
240 CALL FTNCMD('ASSIGN 12=KINN:H24IN;')
    RETURN
250 CALL FTNCMD('ASSIGN 12=KINN:H25IN;')
    RETURN
260 CALL FTNCMD('ASSIGN 12=KINN:H26IN;')
    RETURN
270 CALL FTNCMD('ASSIGN 12=KINN:H27IN;')
    RETURN
280 CALL FTNCMD('ASSIGN 12=KINN:H28IN;')
    RETURN
290 CALL FTNCMD('ASSIGN 12=KINN:H29IN;')
    RETURN
300 CALL FTNCMD('ASSIGN 12=KINN:H30IN;')
    RETURN
310 CALL FTNCMD('ASSIGN 12=KINN:H31IN;')
    RETURN
320 CALL FTNCMD('ASSIGN 12=KINN:H32IN;')
    RETURN
330 CALL FTNCMD('ASSIGN 12=KINN:H33IN;')
    RETURN
340 CALL FTNCMD('ASSIGN 12=KINN:H34IN;')
    RETURN
350 CALL FTNCMD('ASSIGN 12=KINN:H35IN;')
    RETURN
360 CALL FTNCMD('ASSIGN 12=KINN:H36IN;')
    RETURN
370 CALL FTNCMD('ASSIGN 12=KINN:H37IN;')
    RETURN
380 CALL FTNCMD('ASSIGN 12=KINN:H38IN;')
    RETURN
390 CALL FTNCMD('ASSIGN 12=KINN:H39IN;')
    RETURN
400 CALL FTNCMD('ASSIGN 12=KINN:H40IN;')
    RETURN
410 CALL FTNCMD('ASSIGN 12=KINN:H41IN;')
    RETURN
420 CALL FTNCMD('ASSIGN 12=KINN:H42IN;')
    RETURN
430 CALL FTNCMD('ASSIGN 12=KINN:H43IN;')
    RETURN
440 CALL FTNCMD('ASSIGN 12=KINN:H44IN;')
    RETURN
450 CALL FTNCMD('ASSIGN 12=KINN:H45IN;')
    RETURN
460 CALL FTNCMD('ASSIGN 12=KINN:H46IN;')
    RETURN
470 CALL FTNCMD('ASSIGN 12=KINN:H47IN;')
    RETURN
480 CALL FTNCMD('ASSIGN 12=KINN:H48IN;')
    RETURN
490 CALL FTNCMD('ASSIGN 12=KINN:H49IN;')
    RETURN
500 CALL FTNCMD('ASSIGN 12=KINN:H50IN;')
    RETURN
510 CALL FTNCMD('ASSIGN 12=KINN:H51IN;')
    RETURN
520 CALL FTNCMD('ASSIGN 12=KINN:H52IN;')
    RETURN
530 CALL FTNCMD('ASSIGN 12=KINN:H53IN;')
    RETURN
540 CALL FTNCMD('ASSIGN 12=KINN:H54IN;')
    RETURN
```

```
550 CALL FTNCMD('ASSIGN 12=KINN:H55IN;')
    RETURN
560 CALL FTNCMD('ASSIGN 12=KINN:H56IN;')
    RETURN
570 CALL FTNCMD('ASSIGN 12=KINN:H57IN;')
    RETURN
580 CALL FTNCMD('ASSIGN 12=KINN:H58IN;')
    RETURN
590 CALL FTNCMD('ASSIGN 12=KINN:H59IN;')
    RETURN
600 CALL FTNCMD('ASSIGN 12=KINN:H60IN;')
    RETURN
610 CALL FTNCMD('ASSIGN 12=KINN:H61IN;')
    RETURN
620 CALL FTNCMD('ASSIGN 12=KINN:H62IN;')
    RETURN
630 CALL FTNCMD('ASSIGN 12=KINN:H63IN;')
    RETURN
640 CALL FTNCMD('ASSIGN 12=KINN:H64IN;')
    RETURN
650 CALL FTNCMD('ASSIGN 12=KINN:H65IN;')
    RETURN
660 CALL FTNCMD('ASSIGN 12=KINN:H66IN;')
    RETURN
670 CALL FTNCMD('ASSIGN 12=KINN:H67IN;')
    RETURN
680 CALL FTNCMD('ASSIGN 12=KINN:H68IN;')
    RETURN
690 CALL FTNCMD('ASSIGN 12=KINN:H69IN;')
    RETURN
700 CALL FTNCMD('ASSIGN 12=KINN:H70IN;')
    RETURN
710 CALL FTNCMD('ASSIGN 12=KINN:H71IN;')
    RETURN
720 CALL FTNCMD('ASSIGN 12=KINN:H72IN;')
    RETURN
730 CALL FTNCMD('ASSIGN 12=KINN:H73IN;')
    RETURN
740 CALL FTNCMD('ASSIGN 12=KINN:H74IN;')
    RETURN
750 CALL FTNCMD('ASSIGN 12=KINN:H75IN;')
    RETURN
760 CALL FTNCMD('ASSIGN 12=KINN:H76IN;')
    RETURN
770 CALL FTNCMD('ASSIGN 12=KINN:H77IN;')
    RETURN
780 CALL FTNCMD('ASSIGN 12=KINN:H78IN;')
    RETURN
790 CALL FTNCMD('ASSIGN 12=KINN:H79IN;')
    RETURN
800 CALL FTNCMD('ASSIGN 12=KINN:H80IN;')
    RETURN
810 CALL FTNCMD('ASSIGN 12=KINN:H81IN;')
    RETURN
820 CALL FTNCMD('ASSIGN 12=KINN:H82IN;')
    RETURN
830 CALL FTNCMD('ASSIGN 12=KINN:H83IN;')
    RETURN
840 CALL FTNCMD('ASSIGN 12=KINN:H84IN;')
    RETURN
850 CALL FTNCMD('ASSIGN 12=KINN:H85IN;')
    RETURN
```

```
  860 CALL FTNCMD('ASSIGN 12=KINN:H86IN;')
      RETURN
  870 CALL FTNCMD('ASSIGN 12=KINN:H87IN;')
      RETURN
  880 CALL FTNCMD('ASSIGN 12=KINN:H88IN;')
      RETURN
  890 CALL FTNCMD('ASSIGN 12=KINN:H89IN;')
      RETURN
  900 CALL FTNCMD('ASSIGN 12=KINN:H90IN;')
      RETURN
  910 CALL FTNCMD('ASSIGN 12=KINN:H91IN;')
      RETURN
  920 CALL FTNCMD('ASSIGN 12=KINN:H92IN;')
      RETURN
  930 CALL FTNCMD('ASSIGN 12=KINN:H93IN;')
      RETURN
  940 CALL FTNCMD('ASSIGN 12=KINN:H94IN;')
      RETURN
  950 CALL FTNCMD('ASSIGN 12=KINN:H95IN;')
      RETURN
  960 CALL FTNCMD('ASSIGN 12=KINN:H96IN;')
      RETURN
  970 CALL FTNCMD('ASSIGN 12=KINN:H97IN;')
      RETURN
  980 CALL FTNCMD('ASSIGN 12=KINN:H98IN;')
      RETURN
  990 CALL FTNCMD('ASSIGN 12=KINN:H99IN;')
      RETURN
 1000 CALL FTNCMD('ASSIGN 12=KINN:H100IN;')
      RETURN
      ENTRY FILED(IHRD)
C
C   FINDS THE FILE FOR EACH GROUP OF CONTROL HERDS
C
 1010 I = IHRD / 100
      GO TO (1020, 1030, 1040, 1050), I
      GO TO 1060
 1020 CALL FTNCMD('ASSIGN 12=KINN:G100IN(LAST+1);')
      RETURN
 1030 CALL FTNCMD('ASSIGN 12=KINN:G200IN(LAST+1);')
      RETURN
 1040 CALL FTNCMD('ASSIGN 12=KINN:G300IN(LAST+1);')
      RETURN
 1050 CALL FTNCMD('ASSIGN 12=KINN:G400IN(LAST+1);')
      RETURN
 1060 WRITE (6,1070)
 1070 FORMAT ('  ERROR EXTRA HERDS')
      STOP
      END
```

```
C  *******************************************************************
C  *                                                                 *
C  *                  DAIRY CATTLE BREEDING SIMULATION               *
C  *                                                                 *
C  *                                                                 *
C  *
      DIMENSION AI(30,500), PARM(25), SINDEX(4), OPNMAX(6),
     1          BULREC(13,20), IBULLS(20), IPROP(20), LSLAI(500),
     2          RNE(4), HENV(4), ISPMAT(10), ASCOR(2,500),
     3          ISPBUL(10), MCOW(100), MBUL(100), MNCCUL(50),
     4          MCFCUL(50), MYLCUL(50), LBUF(2,100), ISTAT(20),
     5          RNG(4), DUM(3), DUMY(4), IOP(20)
C  *
      COMMON ISEED, EWK(4), GWK(4)
C  *
      LOGICAL*1 AUT(3)/F, T, T/, FALSE /F/, TRUE /T/, AUTFUL /F/,
     1         AUTO, CONT, FIN
C  *
      INTEGER IAJBCA(4)/37, 36, 35, 0/,  MPSEM/15/,
     1         MAUT(2)/'YES', 'NO '/, OPT(9)/'CUR.', ' LAC', 'T.  ',
     1         ' EPA', '    ', '    ', ' ETA', '    ', '    '/,
     2         BLK /'    '/, AST1 /'  * '/, AST2 /' ** '/, NHM /0/,
     3         NCFCUL /0/, NOCCUL /0/, NYLCUL /0/
C  *
C  *     Parameters to generate 305-day lactation records             *
C  *
      REAL *4  AVG(4)/7200., 260., 230., 80./, GSD(4)/413., 18.2,
     1     12.8, 1.66/, PCESD(4)/393., 12.6, 11.9, 1.64/, TCESD(4)
     2     /510., 18.6, 15.4, 1.56/, THESD(4)/255., 11.2, 7.7, 1.13/,
     3     ESIGMA(10)/1., 0.83, 1., 0.96, 0.78, 1., 0.2, 0.2, 0.2, 1./,
     4     GSIGMA(10)/1., 0.54, 1., 0.7, 0.81, 1., 0.0, -0.15, 0.0,
     5     1./, HELVL(4,4)/150., 6.6, 4.5, 0.6, 50., 2.2, 1.5, 0.2,
     6     -50., -2.2, -1.5, -.2, -150., -6.6, -4.5, -0.6/,
     7     AGEAJM(3,4)/.7195, .7154, .7195, .8368, .8380, .8368,
     8     .9197, .9243, .9197, 1., 1., 1./, PINB(4)/103.,
     5     4.26, 3.2, 0.41/
C  *
C  *     Lactation curve parameters and other biological parameters *
C  *
      REAL *4  ACMT(3)/-2.83, -2.28, -2.7/, BSLP(3)/4.6395E-3,
     1     4.8904E-3, 4.9499E-3/, WKPK(3)/10., 7.5, 7.5/, GEST/283./,
     2     DCYC/21./, HDCSD /0.01/, HDCM /0.75/, HDMLK/-2.E-5/,
     3     HRDET(4)/1.0, 0.85, 0.65, 0.45/, FERTM /0.75/,
     4     FERSD /0.01/, FEREC/-.02/,  DTRATE /0.009/, DTMLK/2.E-6/,
     5     DTREC /1.2E-2/, DTYLG /0.02/, DTCF /0.04/, DTUB /0.01/,
     6     HTL /0.019/, HTMLK /2.E-6/, HTREC /0.029/, HTYP /-2.E-3/
C  *
C  *     Economic parameters                                          *
C  *
      REAL *4  CFIX(4)/39935., 36015., 32825., 30000./, CTRNP /1.34/,
     1     FDCAR /2.801E-2/, FDFAT /1.2906/, FDPRO /.6730/,
     2     FDAY/1.2823/, FDRY /1.6880/, FDYLG /1.043/, FDCF /.8211/,
     3     CRFAT /3.6/, CRPROT /0.0/, PQMLK/52.34/,
     4     PEXMLK /38.18/, PFAT /4.80/, PPROT /0.0/,
     5     PCOWP /520./, PCOWF /520./, PCOWH /300./, PCOWD /0.0/, PYLG
     6     /500./, PCFO /300./, PCFH /100./, PCFB /50./, PYSP /1000./
C  *
C  *     Management parameters                                        *
C  *
      REAL *4  HERT(4)/.26, .34, .27, .30/, REP(4)/.50, .51, .51,
     1     .59/, DPROJ/90./, DRYMIN/50./, DAYR/365./,
     2     BRDMIN /50./, DFBRD/-120./
```

```
C  *
      COMPLEX*16 NAUT(3) /'INDIVIDUALLY    ', 'BY RANK ',
     1           'RANDOMLY  '/
C
C***     Find herd and read previous management decisions and AI     ***
C
      WRITE (6,1190)
      READ (5,1210) IHRD, CODE, AUTFUL
      CALL FILES(IHRD)
      READ (4) NDAI, NOAI, NUAI, NEAI, NPAI, NTAI
      IBN = NDAI + 9000
      BBNO = IBN
      NYAI = NEAI - NUAI
      NHRL = 0
      CALL REAMAT(AI, 30, NTAI, 4)
C  *
C  *    Last parameters used                                           *
C  *
   10 READ (2) PARM
      IF (CODE .NE. PARM(2)) GO TO 1180
      IYR = PARM(3) + 1
      PARM(3) = IYR
      IAUT = PARM(4)
      ISDA = (PARM(4) - IAUT) * 10. + 0.5
      IOPT = PARM(5)
      LEV = (PARM(5) - IOPT) * 10. + 0.5
      SINDEX(1) = PARM(6)
      SINDEX(2) = PARM(7)
      SINDEX(3) = PARM(8)
      SINDEX(4) = PARM(9)
      OPNMAX(1) = PARM(10)
      OPNMAX(2) = PARM(11)
      OPNMAX(3) = PARM(12)
      OPNMAX(4) = PARM(13)
      OPNMAX(5) = PARM(14)
      OPNMAX(6) = PARM(15)
      DRYPRD = PARM(16)
      PRDCUL = PARM(17)
      NOBL = PARM(18)
      IBLA = (PARM(18) - NOBL) * 10. + 0.5
      MSEMP = PARM(19)
      SEMP = MSEMP
      CON = PARM(19) - SEMP
      MCON = CON * 100. + 0.5
      NYBL = PARM(20)
      IYBA =(PARM(20) - NYBL) * 10. + 0.5
      MPYS = PARM(21)
      MPOB = 100 - MPYS
      NTBS = (PARM(21) - MPYS) * 100. + 0.5
      NSPM = PARM(22)
      ISPA = (PARM(22) - NSPM) * 10. + 0.5
      QUOTA = PARM(23)
      EXCES = PARM(24)
      NHRD = PARM(25)
      NBULS = NOBL + NYBL
   12 IF (NOBL .LE. NOAI) GO TO 15
      NEXA = NOBL - NOAI
      NOBL = NOAI
      NYBL = NYBL + NEXA
   15 IF (NYAI .GT. 0) GO TO 18
      NOBL = NOBL + NYBL
      NYBL = 0
```

```
   18 IF (NYBL .LE. 0) MPYS = 0
      IF (NOBL .LE. 0) MPOB = 0
      IF (AUTFUL) GO TO 490
   20 WRITE (6,1220) IHRD, IYR, MAUT(IAUT)
      NOP = (IOPT - 1) * 3 + 1
      LOP = NOP + 2
C  *
C  *              Main Menu                                            *
C  *
      WRITE (6,1230) LEV, (OPT(L),L=NOP,LOP), SINDEX, OPNMAX, DRYPRD,
     1  PRDCUL, MPOB, NOBL, NAUT(IBLA), MPYS, NYBL, NAUT(IYBA),
     2  NSPM, NAUT(ISPA)
      IF (IBLA .GT. 1) WRITE (6, 1235) MSEMP, MCON
      WRITE (6,1240)
C  *
C***           CHANGE MANAGEMENT DECISIONS ?                         ***
C  *
   30 WRITE (6,1200)
      READ (5,1210) ICH
   40 GO TO (460, 430, 400, 290, 380, 60, 500), ICH
      GO TO 20
C  *
C***     Input new breeding strategy                                   *
C  *
   50 WRITE (6,1250) MIN, MAX
   60 WRITE (6,1280)
      MAX = 20
      MIN = 0
      READ (5,1690,ERR=50) NOBL
      IF (NOBL .GT. MAX .OR. NOBL .LT. MIN) GO TO 50
      IBLA = 1
      IYBA = 3
      IF (NOBL .EQ. 0) GO TO 150
      GO TO 80
   70 WRITE (6,1250) MIN, MAX
   80 WRITE (6,1290) NAUT
      WRITE (6,1200)
      MIN = 1
      MAX = 3
      READ (5,1690,ERR=70) IBLA
      GO TO (140, 100, 100), IBLA
      GO TO 70
   90 WRITE (6,1250) MIN, MAX
  100 WRITE (6,1320)
      MIN = MPSEM
      MAX = 9999
      READ (5,1410,ERR=90) SEMP
      MSEMP = SEMP
      IF (MSEMP .GT. MAX .OR. MSEMP .LT. MIN) GO TO 90
      GO TO 120
  110 WRITE (6,1250) MIN, MAX
  120 WRITE (6,1330)
      MIN = 10
      MAX = 90
      READ (5,1410,ERR=110) CON
      MCON = CON + .5
      CON = CON / 100.
      IF (MCON .GT. MAX .OR. MCON .LT. MIN) GO TO 110
      GO TO 140
  130 WRITE (6,1250) MIN, MAX
  140 WRITE (6,1310)
      MIN = 0
```

```
      MAX = 100 - NOBL
      READ (5,1410,ERR=130) PYS
      MPYS = PYS + .5
      IF (MPYS .GT. MAX .OR. MPYS .LT. MIN) GO TO 130
      IF (MPYS .EQ. 0) GO TO 180
      GO TO 170
  150 MPYS = 100
      PYS = 100.
      GO TO 170
  160 WRITE (6,1250) MIN, MAX
  170 WRITE (6,1340)
      MIN = 1
      MAX = 20 - NOBL
      READ (5,1690,ERR=160) NYBL
      IF (NYBL .GT. MAX .OR. NYBL .LT. MIN) GO TO 160
      GO TO 205
  200 WRITE (6,1250) MIN, MAX
  205 GO TO (210, 212, 190), IBLA
  210 WRITE (6,1290) NAUT
      WRITE (6,1200)
      MAX = 3
      READ (5,1690,ERR=200) IYBA
      IF (IYBA .GT. MAX .OR. IYBA .LT. MIN) GO TO 200
      GO TO 190
  212 WRITE (6, 1300) NAUT(2), NAUT(3)
      WRITE (6,1200)
      READ (5,1690,ERR=200) IYBA
      IYBA = IYBA + 1
      MIN = 2
      IF (IYBA .GT. MAX .OR. IYBA .LT. MIN) GO TO 200
      GO TO 190
  180 NYBL = 0
  190 MIN = 1
      MAX = 20
      NBULS = NOBL + NYBL
      IF (NBULS .GT. MAX .OR. NBULS .LT. MIN) GO TO 50
      GO TO 230
  220 WRITE (6,1250) MIN, MAX
  230 WRITE (6,1350)
      MAX = 10
      MIN = 0
      READ (5,1690,ERR=220) NSPM
      IF (NSPM .GT. MAX .OR. NSPM .LT. MIN) GO TO 220
      IF (NSPM .EQ. 0) GO TO 255
      GO TO 250
  240 WRITE (6,1250) MIN, MAX
  250 WRITE (6,1300) NAUT(1), NAUT(2)
      WRITE (6,1200)
      MIN = 1
      MAX = 2
      READ (5,1690,ERR=240) ISPA
      IF (ISPA .GT. MAX .OR. ISPA .LT. MIN) GO TO 240
  255 PARM(18) = NOBL + IBLA / 10.
      PARM(19) = MSEMP + CON
      PARM(20) = NYBL + IYBA / 10.
      PARM(21) = MPYS
      PARM(22) = NSPM + ISPA / 10.
      MPOB = 100 - MPYS
      IF ( .NOT. AUT(ISPA)) GO TO 12
      GO TO 270
  260 WRITE (6,1250) MIN, MAX
  270 WRITE (6,1360)
```

```
      READ (5,1210,ERR=50) NTBS
      MIN = 1
      MAX = NBULS
      IF (NTBS .GT. MAX .OR. NTBS .LT. MIN) GO TO 260
      PARM(21) = PARM(21) + NTBS / 100.
      GO TO 12
C  *
C***     Input new days open and number of services                 *
C  *
  280 WRITE (6,1250) MIN, MAX, IER
  290 WRITE (6,1370)
      READ (5,1410,ERR=290) OPNMAX
      OMIN = BRDMIN + DCYC
      MIN = OMIN
      MAX = 1000
      IER = OPNMAX(1)
      IF (OPNMAX(1) .LT. OMIN .OR. OPNMAX(1) .GT. 1000.) GO TO 280
      MAX = 100
      MIN = 1
      IER = OPNMAX(2)
      IF (OPNMAX(2) .GT. 100. .OR. OPNMAX(2) .LT. 1.) GO TO 280
      IF (OPNMAX(3) .LE. 0.) GO TO 350
  300 WRITE (6,1380)
      READ (5,1410,ERR=300) OPNMAX(4)
      GO TO 320
  310 WRITE (6,1250) MIN, MAX
  320 WRITE (6,1390)
      MAX = 1000
      MIN = 0
      READ (5,1410,ERR=310) OPNMAX(5)
      IF (OPNMAX(5) .LT. 0.) GO TO 310
      GO TO 340
  330 WRITE (6,1250) MIN, MAX
  340 WRITE (6,1400)
      READ (5,1410,ERR=340) OPNMAX(6)
      IF (OPNMAX(6) .LT. 0.) GO TO 330
  350 DO 360 J = 1, 6
  360 PARM(J + 9) = OPNMAX(J)
      GO TO 20
C  *
C***     Input new minimum daily production                         *
C  *
  370 WRITE (6,1270)
  380 WRITE (6,1420)
      READ (5,1410,ERR=370) DRYPRD
      WRITE (6,1430)
      READ (5,1410,ERR=370) PRDCUL
      IF (DRYPRD .LE. 0. .OR. PRDCUL .LE. 0.) GO TO 370
      PARM(16) = DRYPRD
      PARM(17) = PRDCUL
      GO TO 20
C  *
C***     Input new selection index                                  *
C  *
  390 WRITE (6,1260)
  400 WRITE (6,1440)
      READ (5,1410,ERR=390) SINDEX
      SX = ABS(SINDEX(1) + SINDEX(2) + SINDEX(3) + SINDEX(4))
      IF (SX .EQ. 0.0) GO TO 390
      DO 410 J = 1, 4
        SINDEX(J) = SINDEX(J) / SX
  410 PARM(J + 5) = SINDEX(J)
```

```
      GO TO 20
C  *
C***    New management level and basis for selection                  *
C  *
  420 WRITE (6,1250) MIN, MAX
  430 WRITE (6,1450)
      READ (5,1690,ERR=430) LEV
      MAX = 4
      MIN = 1
      IF (LEV .LT. 1 .OR. LEV .GT. 4) GO TO 420
      GO TO 450
  440 WRITE (6,1250) MIN, MAX
  450 WRITE (6,1460)
      WRITE (6,1200)
      READ (5,1690,ERR=450) IOPT
      MAX = 3
      IF (IOPT .GT. 3 .OR. IOPT .LT. 1) GO TO 440
      PARM(5) = IOPT + LEV / 10.
      GO TO 20
C  *
C***    Automated decisions and summary output only ?               ***
C  *
  460 WRITE (6,1480)
      WRITE (6,1200)
      READ (5,1210,ERR=460) IAUT
      IF (IAUT .GT. 2 .OR. IAUT .LT. 1) GO TO 460
  470 WRITE (6,1470)
      WRITE (6,1200)
      READ (5,1210,ERR=470) ISDA
      IF (ISDA .GT. 2 .OR. ISDA .LT. 1) GO TO 470
      PARM(4) = IAUT + ISDA / 10.
      IF ( .NOT. AUT(IAUT)) GO TO 20
  480 WRITE (6,1490)
      READ (5,1210,ERR=480) NHRD
      IF (NHRD .LT. 1) GO TO 480
      PARM(25) = NHRD
      GO TO 20
C  *
C  *         INITIALIZE RANDOM NUMBER GENERATORS                       *
C  *
  490 IF ( .NOT. AUT(IAUT)) GO TO 1180
  500 IF (NHRL .GT. 0) GO TO 530
      IF ( .NOT. AUT(ISDA)) GO TO 510
      INIT = IRAND(0)
      INIT = IRAND(1000)
      GO TO 520
  510 WRITE (6,1500)
      READ (5,1210) INIT
      IF (INIT .LE. 0) GO TO 510
  520 I = IRAND(-INIT)
      ISEED = INIT
      SINIT = ISEED
      UNIF = RAND(SINIT)
      SNORM = RANDN(SINIT)
      CALL MNDG(ISEED, GSIGMA, RNG, GWK)
      CALL MNDG(ISEED, ESIGMA, RNE, EWK)
C  *
C  *   Set temporary herd effects                                      *
C  *
  530 CALL MNDG1(ISEED, ESIGMA, RNE, EWK)
      HENV(1) = THESD(1) * RNE(1)
      HENV(2) = THESD(2) * RNE(2)
```

```
      HENV(3) = THESD(3) * RNE(3)
      HENV(4) = THESD(4) * RNE(4)
C  *
C  *    Check if any of own bulls selected for A.I.                   *
C  *
      NYSR = 0
      HRD = IHRD
      NFYS = NPAI + 1
      IF (NFYS .GT. NTAI) GO TO 545
      DO 540 I = NFYS, NTAI
        IF (AI(5,I) .EQ. HRD) NYSR = NYSR + 1
  540 CONTINUE
  545 IFIN = 1
  550 IF (IYBA .EQ. 1) GO TO 560
      IF (NOBL .EQ. 0) GO TO 720
      IF (IBLA .GT. 1) GO TO 620
      NB = NOBL
      GO TO 580
  560 NB = NBULS
      GO TO 580
C
C***     SELECT BULLS INDIVIDUALLY                                    ***
C
  565 WRITE (6, 1515)
      GO TO 580
  570 WRITE (6,1510) IBULLS(J)
      GO TO 586
  580 WRITE (6,1520) NB
      J = 0
  585 J = J + 1
  586 READ (5, 1690,ERR=570) IBULLS(J), IPROP(J), NCS
  588 LCBUL = IBULLS(J) - IBN
      IF (LCBUL .LT. 1 .OR. LCBUL .GT. NEAI) GO TO 570
      DO 590 K = 1, 7
  590 BULREC(K,J) = AI(K,LCBUL)
      BULREC(8,J) = AI(13,LCBUL)
      DO 600 K = 9, 12
  600 BULREC(K,J) = AI(K + 6,LCBUL)
      BULREC(13,J) = AI(27,LCBUL) * SINDEX(1) + AI(28,LCBUL) * SINDEX
     1     (2) + AI(29,LCBUL) * SINDEX(3) + AI(30,LCBUL) * SINDEX(4)
  605 IF (J .EQ. NB) GO TO 610
      IF (NCS .LE. 1) GO TO 585
      NCS = NCS - 1
      J = J + 1
      IBULLS(J) = IBULLS(J-1) + 1
      IPROP(J) = IPROP(J-1)
      GO TO 588
C  *
C  *         All ok ?                                                 *
C  *
  610 CALL PRINTL (IBULLS, NB, 6, 0)
      CALL PRINTL (IPROP, NB, 6, 0)
      WRITE (6,1530)
      READ (5,1540,ERR=580) CONT
      IF (CONT) GO TO 580
      IF (J .LT. 2) GO TO 616
      DO 615 I = 2, J
  615 IPROP(I) = IPROP(I) + IPROP(I - 1)
  616 IF (NBULS .GT. NB) GO TO 720
      IF (IPROP(NB) .LE. 0) GO TO 565
      GO TO 860
C  *
```

```
C  *     Find proven bulls that meet minimum criteria                  *
C  *
  620 NSTB = 0
      DO 630 I = 1, NOAI
        IF (AI(7,I) .GT. SEMP) GO TO 630
        IF (AI(11,I) .LE. 0.0) GO TO 630
        FERT = AI(12,I) / AI(11,I)
        IF (FERT .LT. CON) GO TO 630
        NSTB = NSTB + 1
        LSLAI(NSTB) = I
  630 CONTINUE
      NFT = 1
      NLT = NOBL
      IF (IBLA .GT. 2) GO TO 740
      IF (NSTB .GE. NOBL) GO TO 640
      NYBL = NYBL + NOBL - NSTB
      NOBL = NSTB
      NLT = NOBL
      IF (NOBL .LE. 0) GO TO 720
C  *
C***     SELECT BUllS BY INDEX RANK                                   ***
C  *
  640 DO 650 I = NFT, NSTB
        LAI = LSLAI(I)
        ASCOR(2,I) = AI(27,LAI) * SINDEX(1) + AI(28,LAI)
     1     *SINDEX(2) + AI(29,LAI) * SINDEX(3) + AI(30,LAI)
     2     *SINDEX(4)
        ASCOR(1,I) = LAI
  650 CONTINUE
      CALL ISORT (ASCOR, 2, 500, NFT, NSTB, 2, 3, -1)
      DO 655 I = NFT, NLT
        LSLAI(I) = ASCOR(1,I)
        BULREC(1,I) = LSLAI(I)
        BULREC(13,I) = ASCOR(2,I)
  655 CONTINUE
      IF (NBULS .LE. NLT) GO TO 760
C  *
C  *      Store locations of young bulls                               *
C  *
  680 IF (NYBL .LE. 0) GO TO 760
  720 NSTB = NOBL
      NFAI = NUAI + 1
      DO 730 I = NFAI, NEAI
        NSTB = NSTB + 1
        LSLAI(NSTB) = I
  730 CONTINUE
      NFT = NOBL + 1
      NLT = NBULS
      IFIN = 2
      IF (NSTB .GE. NLT) GO TO 735
      NFBS = NSTB + 1
      NYAI = NSTB - NOBL
      DO 732 I = NFBS, NLT
        NSTB = NSTB + 1
        LOC = IRAND (NYAI) + NOBL
        LSLAI(NSTB) = LSLAI(LOC)
  732 CONTINUE
  735 IF (IYBA .EQ. 2) GO TO 640
C***
C***     SELECT BULLS AT RANDOM                                       ***
C***
  740 NOS = NFT - 1
```

```
      NRNG = NSTB - NOS
  745 DO 750 I = NFT, NLT
        K = IRAND (NRNG) + NOS
        LAI = LSLAI(K)
        BULREC(1,I) = LAI
        BULREC(13,I) = AI(27,LAI) * SINDEX(1) + AI(28,LAI)
     1      *SINDEX(2) + AI(29,LAI) * SINDEX(3) + AI(30,LAI)
     2      *SINDEX(4)
  750 CONTINUE
      DO 755 I = NFT, NLT
        LSLAI(I) = BULREC(1,I)
  755 CONTINUE
      IF (NLT .LT. NBULS) GO TO 720
C  *
C  *    Fill in bulls information and proportions to use             *
C  *
  760 IF (NOBL .LE. 0) GO TO 790
      IF (IBLA .LE. 1) GO TO 780
      IPOB = MPOB / NOBL + 1
      KPOB = 0
      DO 770 I = 1, NOBL
        KPOB = KPOB + IPOB
        LOC = LSLAI(I)
        IPROP(I) = KPOB
        DO 763 J = 1, 7
          BULREC(J,I) = AI(J,LOC)
  763   CONTINUE
        BULREC(8,I) = AI(13,LOC)
        DO 765 J = 9, 12
          BULREC(J,I) = AI(J+6,LOC)
  765   CONTINUE
        IBULLS(I) = BULREC(1,I)
  770 CONTINUE
  780 IF (NYBL .LE. 0) GO TO 860
      NM = NOBL + 1
      KPOB = IPROP(NOBL)
      IPOB = (MPYS * KPOB / MPOB) / NYBL
      GO TO 800
  790 IPOB = MPYS / NYBL + 1
      KPOB = 0
      NM = 1
  800 DO 830 I = NM, NBULS
        KPOB = KPOB + IPOB
        IPROP(I) = KPOB
        LOC = LSLAI(I)
        DO 810 J = 1, 7
          BULREC(J,I) = AI(J,LOC)
  810   CONTINUE
        BULREC(8,I) = AI(13,LOC)
        DO 820 J = 9, 12
          BULREC(J,I) = AI(J+6,LOC)
  820   CONTINUE
        IBULLS(I) = BULREC(1,I)
  830 CONTINUE
  860 IF (AUT(IAUT)) GO TO 1060
      IF (AUT(ISPA) .OR. NSPM .LE. 0) GO TO 920
C
C***     Select individual matings for bull calves                  ***
C
      IF ( .NOT. AUT(IBLA)) GO TO 880
  870 WRITE (6,1570)
      CALL PRINTL(IBULLS, NBULS, 6, 10)
```

```
  880 WRITE (6,1560) NSPM
      WRITE (6,1550)
      CALL REARY2(NSPM, ISPMAT, ISPBUL, LBUF, CONT)
      DO 910 J = 1, NSPM
  890   DO 900 K = 1, NBULS
          IF (ISPBUL(J) .EQ. IBULLS(K)) GO TO 910
  900   CONTINUE
        WRITE (6,1580) ISPBUL(J), ISPMAT(J)
        READ (5,1590) ISPBUL(J)
        GO TO 890
  910 ISPBUL(J) = K
      IF (CONT) GO TO 870
      WRITE (6,1530)
      READ (5,1540,ERR=870) CONT
      IF (CONT) GO TO 870
  920 WRITE (6,1600)
C
C***     OTHER OPTIONAL SPECIFIED MATINGS AND CULLS                    ***
C
      READ (5,1210,ERR=920) ICOPT
  930 GO TO (940, 990, 1040, 1050, 1070), ICOPT
  940 IF ( .NOT. AUT(IBLA)) GO TO 950
      WRITE (6,1570)
      CALL PRINTL(IBULLS, NBULS, 6, 10)
  950 WRITE (6,1610)
      READ (5,1210,ERR=950) NHM
      IF (NHM .LE. 0) GO TO 920
      IF (NHM .GT. 100) GO TO 950
      WRITE (6,1560) NHM
      WRITE (6,1550)
      CALL REARY2(NHM, MCOW, MBUL, LBUF, CONT)
      IF (CONT) GO TO 950
      DO 980 J = 1, NHM
  960   DO 970 K = 1, NBULS
          IF (MBUL(J) .EQ. IBULLS(K)) GO TO 980
  970   CONTINUE
        WRITE (6,1570)
        CALL PRINTL(IBULLS, NBULS, 6, 10)
        WRITE (6,1580) MBUL(J), MCOW(J)
        READ (5,1590) MBUL(J)
        GO TO 960
  980 MBUL(J) = K
      WRITE (6,1530)
      READ (5,1540,ERR=950) CONT
      IF (CONT) GO TO 950
      GO TO 920
  990 IF ( .NOT. AUT(IBLA)) GO TO 1000
      WRITE (6,1570)
      CALL PRINTL(IBULLS, NBULS, 6, 10)
 1000 WRITE (6,1620)
      READ (5,1210,ERR=1000) NCFCUL
      IF (NCFCUL .LE. 0) GO TO 1000
      WRITE (6,1630) NCFCUL
      WRITE (6,1550)
      CALL REARRY(NCFCUL, MNCCUL, CONT)
      IF (CONT) GO TO 1000
      WRITE (6,1530)
      READ (5,1540,ERR=1000) CONT
      IF (CONT) GO TO 1000
 1010 WRITE (6,1640)
      READ (5,1210,ERR=1010) ICHPB
      DO 1020 J = 1, NBULS
```

```
        IF (IBULLS(J) .EQ. ICHPB) GO TO 1030
 1020 CONTINUE
      WRITE (6,1580) ICHPB
      GO TO 1010
 1030 ICHPB = J
      GO TO 920
 1040 WRITE (6,1650)
      READ (5,1210,ERR=1040) NOCCUL
      IF (NOCCUL .LE. 0) GO TO 1040
      WRITE (6,1660) NOCCUL
      WRITE (6,1550)
      CALL REARRY(NOCCUL, MCFCUL, CONT)
      IF (CONT) GO TO 1040
      WRITE (6,1530)
      READ (5,1540,ERR=1040) CONT
      IF (CONT) GO TO 1040
      GO TO 920
 1050 WRITE (6,1670)
      READ (5,1210,ERR=1050) NYLCUL
      IF (NYLCUL .LE. 0) GO TO 1050
      WRITE (6,1680) NYLCUL
      WRITE (6,1550)
      CALL REARRY(NYLCUL, MYLCUL, CONT)
      IF (CONT) GO TO 1050
      WRITE (6,1530)
      READ (5,1540,ERR=1050) CONT
      IF (CONT) GO TO 1050
      GO TO 920
 1060 WRITE (6,1700) IHRD, IYR, INIT
C
C   INITIALIZE SUBROUTINES
C
 1070 CALL BREED(BULREC, IBULLS, IPROP, ISPMAT, ISPBUL, MCOW, MBUL,
     1     MNCCUL, NBULS, NSPM, ISPA, NTBS, NHM, NCFCUL, ICHPB,
     2     AUT(IAUT), GSD, GSIGMA, BRDMIN, HRDET(LEV), DCYC, GEST,
     3     DAYR)
      CALL INLAC (HELVL(1,LEV), HENV, ESIGMA, TCESD, AVG)
      CALL INPROD(DRYPRD, DRYMIN, DAYR, DPROJ, AGEAJM, FDCAR, FDFAT,
     1     FDPRO, FDAY, ACMT, BSLP, WKPK, DUM, DUM, DUMY, 3, 4)
      CALL INETA(AI, SINDEX, AVG, IAJBCA, HERT, REP, DAYR, NOAI, BBNO,
     1     DPROJ, DUMY, DUMY, DUMY, DUMY, DUMY, DUMY, DUMY, DUMY, DUMY,
     2     4)
      CALL SELEC(IOPT, AUT(IAUT), QUOTA, EXCES, DPROJ, DAYR, FDRY,
     1     PRDCUL)
      CALL YOUNG(SINDEX, MCFCUL, NOCCUL, IYR, IHRD, DAYR, DTYLG, DTCF,
     1     FDYLG, FDCF, AUT(IAUT))
      CALL CALVES(ESIGMA, PCESD, PINB, SINDEX, FDCF, FERSD, FERTM,
     1     HDCSD, HDCM, DTCF, DTUB, IHRD, DAYR, IYR, AUT(IAUT))
      CALL SUMOUT(QUOTA, PQMLK, EXCES, PEXMLK, PFAT, CRFAT, PPROT,
     1     CRPROT, PCOWP, PCOWF, PCOWH, PCOWD, PYLG, PCFO, PCFH, PCFB,
     2     PYSP, CFIX(LEV), CTRNP, NYSR, AVG, IAJBCA, AUT(IAUT), AUTFUL)
      IF (AUT(IAUT)) GO TO 1170
C
C*** Output hard copy of all management decisions for the current year *
C
      WRITE (10,1700) IHRD, IYR, INIT
      WRITE (10,1230) LEV, (OPT(L),L=NOP,LOP), SINDEX, OPNMAX, DRYPRD,
     1 PRDCUL, MPOB, NOBL, NAUT(IBLA), MPYS, NYBL, NAUT(IYBA),
     2 NSPM, NAUT(ISPA)
      IF (IBLA .GT. 1) WRITE (10, 1235) MSEMP, MCON
      IP = 0
      DO 1110 J = 1, NBULS
```

```
        IOP(J) = IPROP(J) - IP
        IP = IPROP(J)
        K = BULREC(8,J) / 20. + 1.99
        IF (K - 2) 1080, 1090, 1100
 1080   ISTAT(J) = AST2
        GO TO 1110
 1090   ISTAT(J) = AST1
        GO TO 1110
 1100   ISTAT(J) = BLK
 1110 CONTINUE
      WRITE (10,1710)
      CALL PRINTL(IBULLS, NBULS, 10, 10)
      CALL PRINTL(IOP, NBULS, 10, 10)
      CALL PRINTL(ISTAT, NBULS, 10, 20)
      WRITE (10,1720) (BULREC(13,J),J=1,NBULS)
      LINE = 33
 1120 IF (NSPM .LE. 0 .OR. AUT(ISPA)) GO TO 1130
      WRITE (10,1730)
      CALL PRINTL(ISPMAT, NSPM, 10, 10)
      CALL PRINTL(ISPBUL, NSPM, 10, 10)
      LINE = LINE + 3
 1130 IF (NHM .LE. 0) GO TO 1140
      WRITE (10,1740)
      CALL PRINTL(MCOW, NHM, 10, 10)
      CALL PRINTL(MBUL, NHM, 10, 10)
      LINE = LINE + 3
 1140 IF (NCFCUL .LE. 0) GO TO 1150
      WRITE (10,1750) ICHPB
      CALL PRINTL(MNCCUL, NCFCUL, 10, 10)
      LINE = LINE + 2
 1150 IF (NOCCUL .LE. 0) GO TO 1160
      WRITE (10,1760)
      CALL PRINTL(MCFCUL, NOCCUL, 10, 10)
      LINE = LINE + 2
 1160 IF (NYLCUL .LE. 0) GO TO 1170
      WRITE (10,1770)
      CALL PRINTL(MYLCUL, NYLCUL, 10, 10)
      LINE = LINE + 2
 1170 WRITE (12) PARM
C  *
C  *          Simulate herd year and check if more herds              *
C  *
      CALL GENREC(OPNMAX, MYLCUL, SINDEX, DRYPRD, IHRD, IYR, NYLCUL,
     1     IOPT, AUT(IAUT), DPROD, DAYR, DTRATE, DTMLK, DTREC, IAJBCA,
     2     AVG, HRDET(LEV), HDMLK, FEREC, DCYC, DPROJ, BRDMIN, DFBRD,
     3     FDRY, GEST, HTL, HTMLK, HTREC, HTYP, LINE)
      NHRL = NHRL + 1
      IHRD = IHRD + 1
      WRITE (7'IHRD, 1690) IYR, IHRD
      IF (NHRD-NHRL .GT. 0) GO TO 10
      GO TO 1790
 1180 WRITE (6,1780)
 1190 FORMAT (20X, 'DAIRY CATTLE BREEDING SIMULATION', /, 20X, 32('-'),
     1       ////, 3X, 'YOU WILL BE PROMPTED FOR DECISIONS', //, 3X,
     2       '- ENTER  ALL VALUES ON ONE LINE (EXCEPT WHEN ENTERING',
     3       ' ANIMALS)', /, 3X, '- WHEN ENTERING ANIMALS START A NEW ',
     4       'LINE FOR EACH ANIMAL', /, 3X, '- EACH VALUE ENTERED MUST',
     5       ' BE FOLLOWED BY A COMMA !!', ///, 'HERD NUMBER AND CODE ?'
     6       )
 1200 FORMAT ('OPTION ?')
 1210 FORMAT (I12, F12.0, L1)
 1220 FORMAT (/, '     HERD', I4, '     SIMULATION OF YEAR', I4, //, 6X,
```

```
     1        'MANAGEMENT DECISIONS SELECTED', //,
     2        '   1   OUTPUT PRINTED COPY OF HERDS ? ', A3)
1230 FORMAT ('   2   MANAGEMENT LEVEL OF', I3, /, 9X,
     1        'RANKING AND CULLING BASED ON: ', 3A4, /,
     2        '   3   WEIGHTINGS FOR THE SELECTION INDEX :', /, 9X,
     3        'MILK', F7.2, '    FAT', F7.2, '    PROTEIN', F7.2,
     4        '    TYPE', F7.2, /,
     5        '   4   NUMBER OF DAYS OPEN OR SERVICES :', /, 9X, F4.0,
     6        ' DAYS OR', F4.0, ' SERVICES WITH EXTRA', ' FOR THE TOP',
     7        F4.0, ' COWS', /, 9X, 'BASED ON AN ', 'EXPONENT', F7.3,
     8        '  AN ADJUSTMENT FOR DAYS', F7.2, /, 9X,
     9        'AND AN ADJUSTMENT FOR SERVICES', F7.3, /,
     *        '   5   MINIMUM DAILY MILK PRODUCTION:', /, 9X, 'AT', F5.1,
     1        ' KG/DAY A COW IS DRIED OFF', /, 9X, 'AT', F5.1,
     2        ' KG/DAY A CULL COW IS SOLD', /, '    6   MATINGS-', I5,
     3        '% TO', I4, ' PROVEN SIRES SELECTED ', 2A8, /, 9X, I3,
     4        '% TO', I4, ' YOUNG SIRES SELECTED ', 2A8, /,
     5        9X, 'SPECIAL MATE', I4, '  COWS SELECTED ', 2A8)
1235 FORMAT ( 9X, 'MAXIMUM SEMEN PRICE $', I5, '/VIAL', 4X,
     1        'MINIMUM CONCEPTION', I4, '%')
1240 FORMAT ('   7   CONTINUE')
1250 FORMAT (' ERROR - MINIMUM VALUE IS', I4, ' MAXIMUM IS', I5,
     1        ' VALUE', I4, '?')
1260 FORMAT ('    ERROR WEIGHTS SUM TO "0"')
1270 FORMAT ('    ERROR PRODUCTION MUST BE GREATER THAN  "0"')
1280 FORMAT ('NUMBER OF PROVEN SIRES ?')
1290 FORMAT ('    ENTER THE METHOD OF SELECTION', /, '    1    ', 2A8, /,
     1        '    2    ', 2A8, /, '    3    ', 2A8)
1300 FORMAT ('    ENTER THE METHOD OF SELECTION', /, '    1    ', 2A8, /,
     1        '    2    ', 2A8)
1310 FORMAT ('PERCENT OF MATINGS TO YOUNG SIRES ?')
1320 FORMAT ('MAXIMUM SEMEN PRICE ?')
1330 FORMAT ('MINIMUM CONCEPTION RATE ?')
1340 FORMAT ('NUMBER OF YOUNG BULLS ?')
1350 FORMAT ('NUMBER OF SPECIAL MATINGS ?')
1360 FORMAT ('NUMBER OF TOP BULLS TO USE FOR SPECIAL MATINGS ?')
1370 FORMAT ('    ENTER THE MINIMUM NUMBER OF DAYS AND SERVICES', /,
     1        '   AND THE NUMBER OF TOP COWS TO KEEP LONGER')
1380 FORMAT ('    PRESS RETURN FOR TWO LEVELS OR', /,
     1     '   ENTER THE EXPONENT FOR AN EXPONENTIAL RELATIONSHIP')
1390 FORMAT ('ADJUSTMENT FOR DAYS ?')
1400 FORMAT ('ADJUSTMENT FOR SERVICES ?')
1410 FORMAT (6F10.2)
1420 FORMAT ('DAILY MILK PRODUCTION AT WHICH TO DRY OFF A COW ?')
1430 FORMAT ('DAILY MILK PRODUCTION AT WHICH TO SELL A COW ',
     1        'SELECTED FOR CULLING ?')
1440 FORMAT ('    ENTER THE SELECTION INDEX WIEGHTS FOR', /,
     1        '   MILK, FAT, PROTEIN, AND TYPE')
1450 FORMAT ('MANAGEMENT LEVEL ?')
1460 FORMAT ('    CULLING DECISIONS TO BE BASED ON', /,
     1        '      1   CURRENT LACTATION', /,
     2        '      2   ESTIMATED PRODUCING ABILITY', /,
     3        '      3   ESTIMATED TRANSMITTING ABILITY')
1470 FORMAT ('    THE RANDOM NUMBER GENERATORS TO BE INITIALIZED WITH',
     1        /, '    1   A NUMBER TO BE SPECIFIED', /, '    2   A RANDOM',
     2        ' NUMBER')
1480 FORMAT ('    THE OUTPUT TO BE', /, '    1   FULL PRINTED OUTPUT', /,
     1        '    2   ONLY SUMMARIES STORED')
1490 FORMAT ('NUMBER IF HERDS IN THIS GROUP ?')
1500 FORMAT ('INTEGER SEED ?')
1510 FORMAT ('    BULL', I5, ' DOES NOT EXIST')
1515 FORMAT ('AT LEAST ONE BULL MUST BE BRED TO A PROPORTION > 0')
```

```
 1520 FORMAT ('  ENTER', I4, ' BULLS EACH ON A SEPARATE LINE !', /,
    1 ' BULL NUMBER, THE PROPORTION TO USE HIM AND PRESS RETURN')
 1530 FORMAT ('PRESS RETURN IF OK.  ENTER "T" TO RE-ENTER')
 1540 FORMAT (L1)
 1550 FORMAT ('START A NEW LINE FOR EACH ANIMAL !')
 1560 FORMAT ('   ENTER', I4, ' COWS EACH FOLLOWED BY A BULL')
 1570 FORMAT ('   BULLS AVAILABLE')
 1580 FORMAT ('   BULL', I5, ' NOT AVAILABLE', /, '   CHOOSE ANOTHER',
    1       ' BULL TO BREED TO', I5)
 1590 FORMAT (I10)
 1600 FORMAT (5X, 'OTHER OPTIONS FOR SPECIFIC ANIMALS', /,
    1       '   1   MAKE SPECIFIC MATINGS', /,
    2       '   2   SELL WEEK OLD CALVES', /,
    3       '   3   SELL YEAR OLD CALVES', /, '   4   SELL YEARLINGS',
    4       /, '   5   CONTINUE', /, 'OPTION ?')
 1610 FORMAT ('NUMBER OF INDIVIDUAL MATINGS ?')
 1620 FORMAT ('NUMBER OF NEW CALVES TO BE CULLED ?')
 1630 FORMAT ('   ENTER', I4, ' COWS')
 1640 FORMAT ('   ENTER A CHEAP BULL TO USE ON THESE COWS')
 1650 FORMAT ('NUMBER OF OLD CALVES TO BE CULLED ?')
 1660 FORMAT ('   ENTER', I4, ' YEAR OLD CALVES')
 1670 FORMAT ('NUMBER OF YEARLINGS TO BE CULLED ?')
 1680 FORMAT ('   ENTER', I4, ' YEARLINGS')
 1690 FORMAT (6I10)
 1700 FORMAT (':', //, 30X, 'HERD', I5, 6X, 'YEAR', I5, 8X,
    1       'THE SEED WAS', I7, /, '+', 29X, 24('_'), /)
 1710 FORMAT (//, '   BULLS USED, THEIR PROPORTION OF USE AND',
    1       ' THEIR INDEX', /, 10X, '(*)  PRELIMINARY PROOF ONLY', 10X,
    2       '(**)  PEDIGREE ESTIMATE ONLY', /)
 1720 FORMAT (3X, 20F6.1, //)
 1730 FORMAT ('   SPECIAL MATINGS')
 1740 FORMAT ('   OTHER HAND MATINGS')
 1750 FORMAT ('   COWS TO BE BRED TO', I5, ' AND TO HAVE',
    1       ' THE CALVES SOLD')
 1760 FORMAT ('   CALVES TO BE CULLED')
 1770 FORMAT ('   YEARLINGS TO BE CULLED')
 1780 FORMAT ('   INCORRECT CODE')
 1790 STOP
      END


C   ****************************************************************
C   *                                                              *
C   *  This subroutine generates cows records for all cows and     *
C   *   yearlings.                                                 *
C   ****************************************************************
C
      SUBROUTINE GENREC(OPNMAX, MYLCUL, SINDEX, DRYPRD, IHRD, IYR,
     1           NYLCUL, ISEL, AUTO, DPROD, DAYR, DTRATE, DTMLK, DTREC,
     2           IAJBCA, AVG, HRDET, HDMLK, FEREC, DCYC, DPROJ, BRDMIN,
     3           DFBRD, DRYFD, GEST, HTL, HTMLK, HTREC, HTYP, LINE)
C
      DIMENSION HRDCOW(40,250), HRDYLG(18,100), AGE(250), OLDAT(250),
     1          OSTART(250), OFIN(250), OSTOP(250), DSTART(250),
     2          DFIN(250), DSTOP(250), OYRLAC(4,250), OLDLAC(4,250),
     3          DYRLAC(4,250), ETA(4,250), EPA(4,250), OAPRD(3,250),
     4          DAPRD(3,250), OUTDAT(250), PBCA(4), TAPRD(4), HBCA(4),
     5          HPA(4), HTA(4), SCUR(4), DETA(4), OPNMAX(6), SINDEX(4),
     6          TRPRD(3), MCFCUL(50), MYLCUL(50), DBCA(4), IAJBCA(4),
     7          AVG(4), SCORE(3,250), CURBCA(4,250), COWS(250),
     8          TACPRD(3), PACBCA(4), OLDRB(250), DUEPK(250),
```

```
     9          OLDPK(250), RPRD(3), DFER(2)
C
      LOGICAL*1 OLCOW, CULL /F/, FALSE /F/, TRUE /T/, FINISH, AUTO
   10 CALL INBLUP(IAJBCA, AVG)
      CALL INKIL(GEST)
      DO 20 I = 1, 3
        TAPRD(I) = 0.0
        PBCA(I) = 0.0
   20 TRPRD(I) = 0.0
      TAPRD(4) = 0.0
      PBCA(4) = 0.0
      TDP = 0.0
      NCLAC = 0
      NPCUL = 0
      NFCUL = 0
      NHCUL = 0
      NDCUL = 0
      NYCUL = 0
      NYC = 1
      NRLAC = 0
      TAGE = 0.0
      FINISH = FALSE
      OLCOW = TRUE
      NOPNEX = OPNMAX(3) + 1.
C
      NHYS = IYR * 500 + IHRD
      READ (2) NOC
      IF (NOC .EQ. 0) GO TO 440)
C  *
C  *    Read in old cows                                               *
C  *
      CALL REAMAT(HRDCOW, 40, NOC, 2)
      NCOW = 0
      NC = NOC
   30 DO 430 I = 1, NC
        NCOW = NCOW + 1
        IF (OLCOW) GO TO 70
C  *
C  *      Initialize yearling record                                   *
C  *
   40   DO 50 J = 1, 4
          HRDCOW(J,NCOW) = HRDYLG(J,I)
          HRDCOW(J + 10,NCOW) = HRDYLG(J + 6,I)
          HRDCOW(J + 14,NCOW) = HRDYLG(J + 10,I)
          HRDCOW(J + 34,NCOW) = HRDYLG(J + 14,I)
          HRDCOW(J + 18,NCOW) = 0.0
          HRDCOW(J + 22,NCOW) = 0.0
          HRDCOW(J + 26,NCOW) = 0.0
          HRDCOW(J + 30,NCOW) = 0.0
   50   CONTINUE
          HRDCOW(5,NCOW) = 150.
        BRLAST = 1.0
        HRDCOW(39,NCOW) = HRDYLG(5,I)
        HRDCOW(40,NCOW) = HRDYLG(6,I)
        NREC = 0
        ISTAT = 1
        DUEDAT = HRDYLG(4,I) - BRDMIN + DFBRD
        FUTDAT = -1.
        DUERB = 0.
   60   IF (NYLCUL .LE. 0) GO TO 80
        ICOW = HRDCOW(1,NCOW)
        IF (ICOW .LT. MYLCUL(NYC)) GO TO 80
```

```
      NYC = NYC + 1
      IF (NYC .GT. NYLCUL) NYLCUL = 0
      IF (ICOW .NE. MYLCUL(NYC - 1)) GO TO 60
      NYCUL = NYCUL + 1
      ISTAT = 7
      GO TO 80
C  *
C  *     Find a cows old record and set status                          *
C  *
   70 NREC = HRDCOW(6,NCOW)
      ISTAT = HRDCOW(7,NCOW)
      DUEDAT = HRDCOW(8,NCOW) - DAYR
      FUTDAT = HRDCOW(9,NCOW)
      BRLAST = HRDCOW(10,NCOW)
      DUERB = 10. * (HRDCOW(6,NCOW) - NREC) + HRDCOW(7,NCOW) - ISTAT
   80 RANK = HRDCOW(5,NCOW)
      IRANK = RANK
      COWS(NCOW) = HRDCOW(1,NCOW)
      IAGE = IYR + 10 - IFIX(HRDCOW(1,NCOW)/100.)
      AGE(NCOW) = FLOAT(IAGE) - HRDCOW(4,NCOW) / DAYR + 1.
      IF (HRDCOW(1,NCOW) .GT. 10000. .OR. HRDCOW(1,NCOW) .LT. 1.)
     1   GO TO 410
      OSTART(NCOW) = 0.0
      DSTART(NCOW) = 0.0
      OSTOP(NCOW) = 0.0
      DSTOP(NCOW) = 0.0
      OFIN(NCOW) = 0.0
      DFIN(NCOW) = 0.0
      DO 90 J = 1, 3
        OYRLAC(J,NCOW) = 0.0
        DYRLAC(J,NCOW) = 0.0
        OAPRD(J,NCOW) = 0.0
        DAPRD(J,NCOW) = 0.0
   90 CONTINUE
      OYRLAC(4,NCOW) = 0.0
      DYRLAC(4,NCOW) = 0.0
      DEVMLK = (HRDCOW(11,NCOW) + HRDCOW(15,NCOW)) / AVG(1)
      DFER(1) = HRDCOW(39,NCOW) + HDMLK * DEVMLK
      DFER(2) = HRDCOW(40,NCOW) + FEREC * NREC
      DOPMAX = OPNMAX(1)
      ISEMAX = OPNMAX(2)
      IF (NOPNEX .LE. 1) GO TO 100
      EXOP = NOPNEX - RANK
      IF (EXOP .LE. 0.0) GO TO 100
      OX = EXOP ** OPNMAX(4)
      DOPMAX = DOPMAX + OX * OPNMAX(5)
      ISEMAX = ISEMAX + (OX * OPNMAX(6) + 0.5)
  100 IF (ISTAT .GT. 2) GO TO 200
C  *
C  *     Pregnant ?                                                     *
C  *
      IF (FUTDAT .GE. 0.0) GO TO 110
      CALL MATE(HRDCOW(15,NCOW), HRDCOW(1,NCOW), IRANK,
     1     DFER, DUEDAT, FUTDAT, BRLAST, ISEMAX, DOPMAX,
     2     CULL)
      IF (CULL) GO TO 220
      GO TO 120
  110 FUTDAT = FUTDAT - DAYR
C  *
C  *     New lactation ?                                                *
C  *
  120 IF (FUTDAT .GT. DAYR .OR. FUTDAT .LT. 0.0) GO TO 170
```

```
      CALL LACT(OLDLAC(1,NCOW), HRDCOW(19,NCOW), HRDCOW(11,NCOW),
     1     HRDCOW(15,NCOW), OLDRB(NCOW), DUERB, OLDAT(NCOW), DUEDAT,
     2     FUTDAT)
      IF (ISTAT .LT. 2) CALL BLUP(HRDCOW(2,NCOW), IHRD, IYR,
     1     HRDCOW(19,NCOW))
  130 FBR = DUEDAT + BRDMIN
C  *
C  *      Breed again ?                                              *
C  *
      IF (FBR .GT. DAYR) GO TO 140
      CALL MATE(HRDCOW(15,NCOW), HRDCOW(1,NCOW), IRANK,
     1     DFER, DUEDAT, FUTDAT, BRLAST, ISEMAX, DOPMAX,
     2     CULL)
      IF (CULL) GO TO 210
C  *
C  *      Another lactation ?                                        *
C  *
      IF (FUTDAT .GT. DAYR .OR. FUTDAT .LT. 0.0) GO TO 140
      CALL LACT(OLDLAC(1,NCOW), HRDCOW(19,NCOW), HRDCOW(11,NCOW),
     1     HRDCOW(15,NCOW), OLDRB(NCOW), DUERB, OLDAT(NCOW), DUEDAT,
     2     FUTDAT)
C
C  *       ACTUAL PRODUCTION                                         *
C
      GO TO 150
  140 IF (ISTAT .LT. 2) GO TO 160
      OSTART(NCOW) = ABS(OLDAT(NCOW))
  150 CALL PROD(AGE(NCOW), OLDRB(NCOW), OLDPK(NCOW), OLDAT(NCOW),
     1     DUEDAT, OSTART(NCOW), OSTOP(NCOW), OFIN(NCOW),
     2     OLDLAC(1,NCOW), OYRLAC(1,NCOW), OAPRD(1,NCOW), DRYPRD, 3,
     3     4)
  160 CALL PROD(AGE(NCOW), DUERB, DUEPK(NCOW), DUEDAT, FUTDAT,
     1     DSTART(NCOW), DSTOP(NCOW), DFIN(NCOW), HRDCOW(19,NCOW),
     2     DYRLAC(1,NCOW), DAPRD(1,NCOW), DRYPRD, 3, 4)
      ISTAT = 2
      GO TO 260
C
C  *      Cows with no new lactation                                 *
C
  170 IF (ISTAT .LT. 2) GO TO 180
      DSTART(NCOW) = ABS(DUEDAT)
      CALL PROD(AGE(NCOW), DUERB, DUEPK(NCOW), DUEDAT, FUTDAT,
     1     DSTART(NCOW), DSTOP(NCOW), DFIN(NCOW), HRDCOW(19,NCOW),
     2     DYRLAC(1,NCOW), DAPRD(1,NCOW), DRYPRD, 3, 4)
  180 OLDLAC(1,NCOW) = 0.0
      OLDLAC(2,NCOW) = 0.0
      OLDLAC(3,NCOW) = 0.0
      OLDLAC(4,NCOW) = 0.0
      OLDAT(NCOW) = 0.0
      GO TO 260
  190 DFIN(NCOW) = HRDCOW(4,NCOW)
      GO TO 180
C  *
C  *     Dry off fertility culls from last year                      *
C  *
  200 IF (ISTAT .GT. 3) GO TO 190
      ISTAT = 4
      NFCUL = NFCUL + 1
      OLDAT(NCOW) = 0.0
      DSTART(NCOW) = ABS(DUEDAT)
      CALL PROD(AGE(NCOW), DUERB, DUEPK(NCOW), DUEDAT, FUTDAT,
     1     DSTART(NCOW), DSTOP(NCOW), DFIN(NCOW), HRDCOW(19,NCOW),
```

```
     2       DYRLAC(1,NCOW), DAPRD(1,NCOW), DRYPRD, 3, 4)
        GO TO 180
C
C  *        New fertility culls                                   *
C  *
  210   IF (ISTAT .LT. 2) GO TO 240
        OSTART(NCOW) = ABS(OLDAT(NCOW))
        CALL PROD(AGE(NCOW), OLDRB(NCOW), OLDPK(NCOW), OLDAT(NCOW),
     1       DUEDAT, OSTART(NCOW), OSTOP(NCOW), OFIN(NCOW),
     2       OLDLAC(1,NCOW), OYRLAC(1,NCOW), OAPRD(1,NCOW), DRYPRD, 3,
     3       4)
        GO TO 240
  220   OLDAT(NCOW) = 0.0
        OLDLAC(1,NCOW) = 0.0
        OLDLAC(2,NCOW) = 0.0
        OLDLAC(3,NCOW) = 0.0
        OLDLAC(4,NCOW) = 0.0
        IF (ISTAT .GT. 1) GO TO 230
        ISTAT = 4
        NFCUL = NFCUL + 1
        DFIN(NCOW) = BRLAST
        GO TO 260
  230   DSTART(NCOW) = ABS(DUEDAT)
  240   CALL PROD(AGE(NCOW), DUERB, DUEPK(NCOW), DUEDAT, FUTDAT,
     1       DSTART(NCOW), DSTOP(NCOW), DFIN(NCOW), HRDCOW(19,NCOW),
     2       DYRLAC(1,NCOW), DAPRD(1,NCOW), DRYPRD, 3, 4)
        ISTAT = 3
        IF (DFIN(NCOW) .EQ. DAYR) GO TO 260
  250   ISTAT = 4
        NFCUL = NFCUL + 1
        IF (DFIN(NCOW) .GT. BRLAST) GO TO 260
        DFIN(NCOW) = BRLAST
        DSTOP(NCOW) = DFIN(NCOW) - DUEDAT
        CALL AJPROD(DSTART(NCOW), DSTOP(NCOW), DUERB, DUEPK(NCOW),
     1       DUEDAT, DAPRD(1,NCOW), DYRLAC(1,NCOW), DRYPRD, 0., 3, 4)
C
C  ***        RANDOM DEATH AND HEALTH LOSSES                        ***
C
  260   HTPROB = HTL + HTMLK * DEVMLK + HTYP * (HRDCOW(14,NCOW) +
     1  HRDCOW(18,NCOW))  + HTREC * NREC
        HTRAN = FRAND(0.)
        IF (HTRAN .GE. HTPROB) GO TO 270
        DEDAT = FRAND(0.) * 365.
        IF (DEDAT .GT. DFIN(NCOW) .AND. ISTAT .GT. 3) GO TO 270
        ISTAT = 8
        NHCUL = NHCUL + 1
        GO TO 280
  270   DTPROB = DTRATE + DTMLK * DEVMLK + DTREC * NREC
        DTRAN = FRAND(0.)
        IF (DTRAN .GE. DTPROB) GO TO 350
        DEDAT = FRAND(0.) * DAYR
        IF (DEDAT .GT. DFIN(NCOW) .AND. ISTAT .GT. 3) GO TO 350
        ISTAT = 9
        NDCUL = NDCUL + 1
  280   IF (HRDCOW(19,NCOW) .LE. 0.) GO TO 340
        IF (DEDAT .GT. DFIN(NCOW)) GO TO 340
        IF (DEDAT .GT. DUEDAT) GO TO 320
        IF (DEDAT .GT. OFIN(NCOW)) GO TO 310
        IF (DEDAT .GT. OLDAT(NCOW)) GO TO 290
        OSTOP(NCOW) = 0.0
        OFIN(NCOW) = DEDAT
        GO TO 300
```

```
  290    OSTOP(NCOW) = OSTOP(NCOW) - (OFIN(NCOW) - DEDAT)
  300    CALL AJPROD(OSTART(NCOW), OSTOP(NCOW), OLDRB(NCOW), OLDPK(NCOW),
     1        OLDAT(NCOW), OAPRD(1,NCOW), OYRLAC(1,NCOW), DRYPRD, 0., 3,
     2        4)
  310    DSTOP(NCOW) = 0.0
         DFIN(NCOW) = DEDAT
         GO TO 330
  320    DSTOP(NCOW) = DSTOP(NCOW) - (DFIN(NCOW) - DEDAT)
         DFIN(NCOW) = DEDAT
  330    CALL AJPROD(DSTART(NCOW), DSTOP(NCOW), DUERB, DUEPK(NCOW),
     1        DUEDAT, DAPRD(1,NCOW), DYRLAC(1,NCOW), DRYPRD, 0., 3, 4)
         CALL KILCF(HRDCOW(1,NCOW), OLDAT(NCOW), DUEDAT, FUTDAT, DEDAT)
         GO TO 350
  340    DFIN(NCOW) = DEDAT
         IF (FUTDAT .GT. 0.) CALL KILCF(HRDCOW(1,NCOW), OLDAT(NCOW),
     1        DUEDAT, FUTDAT, DEDAT)
C
C  ***        SUMS FOR ROLLING HERD AVERAGES                          ***
C
  350    IF (OFIN(NCOW) .LE. 0.0) GO TO 370
         IF (OSTOP(NCOW) .LT. DPROJ) GO TO 370
         IF (OSTART(NCOW) .GE. 305.) GO TO 370
         NRLAC = NRLAC + 1
         NREC = NREC + 1
         CALL RHPROD(AGE(NCOW), OLDRB(NCOW), OLDPK(NCOW), OLDAT(NCOW),
     1        OSTOP(NCOW), STOP, RPRD, OLDLAC(1,NCOW), 3, 4)
         TDP = TDP + STOP
         TAGE = TAGE + AGE(NCOW) + OLDAT(NCOW) / DAYR - 1.
         DO 360 J = 1, 3
           PBCA(J) = OLDLAC(J,NCOW) + PBCA(J)
  360    TRPRD(J) = RPRD(J) + TRPRD(J)
         PBCA(4) = OLDLAC(4,NCOW) + PBCA(4)
  370    IF (DFIN(NCOW) .LE. 0.0) GO TO 390
         IF (DSTOP(NCOW) .LT. DPROJ) GO TO 390
         IF (DFIN(NCOW) .EQ. DAYR .AND. DSTOP(NCOW) .LT. 305.)
     1      GO TO 390
         IF (DSTART(NCOW) .GE. 305.) GO TO 390
         NRLAC = NRLAC + 1
         NREC = NREC + 1
         CALL RHPROD(AGE(NCOW), DUERB, DUEPK(NCOW), DUEDAT, DSTOP(NCOW),
     1        STOP, RPRD, HRDCOW(19,NCOW), 3, 4)
         TDP = TDP + STOP
         TAGE = TAGE + AGE(NCOW) + DUEDAT / DAYR - 1.
         DO 380 J = 1, 3
           PBCA(J) = HRDCOW(J + 18,NCOW) + PBCA(J)
  380    TRPRD(J) = RPRD(J) + TRPRD(J)
         PBCA(4) = HRDCOW(22,NCOW) + PBCA(4)
  390    DO 400 J = 1, 4
  400    TAPRD(J) = TAPRD(J) + OYRLAC(J,NCOW) + DYRLAC(J,NCOW)
         CREC = AINT(DUERB)
         HRDCOW(5,NCOW) = RANK
         HRDCOW(6,NCOW) = NREC + CREC / 10.
         HRDCOW(7,NCOW) = ISTAT + DUERB - CREC
         HRDCOW(8,NCOW) = DUEDAT
         HRDCOW(9,NCOW) = FUTDAT
         HRDCOW(10,NCOW) = BRLAST
         GO TO 430
  410    WRITE (6,420)
  420    FORMAT ('   PROBLEMS COW LOST')
         NCOW = NCOW - 1
  430 CONTINUE
      IF (FINISH) GO TO 450
```

```
C  *
C  *      Read in last years yearlings                               *
C  *
      READ (2) NYLG
      CALL REAMAT(HRDYLG, 18, NYLG, 2)
  440 NC = NYLG
      FINISH = TRUE
      OLCOW = FALSE
      GO TO 30
C  *
C  *      END COW LOOP                                               *
C  *
C  *        Herd averages                                            *
C  *
  450 NR = NRLAC
      IF (NRLAC .LE. 0) NR = 1
      TR = NR + 0.00001
      DO 460 J = 1, 4
        HPA(J) = PBCA(J) / NR
        HTA(J) = AVG(J) + ((TR - 1.)/(NR + 1)) * (HPA(J) - AVG(J))
        PACBCA(J) = PBCA(J)
  460 CONTINUE
      TACPRD(1) = TRPRD(1)
      TACPRD(2) = TRPRD(2)
      TACPRD(3) = TRPRD(3)
      ACAGE = TAGE
      ACTDP = TDP
      NACLAC = NRLAC
      CALL ALETA(COWS, HPA, HTA, 0)
C  *
C  *    Update cows EPA's and ETA's and index scores                 *
C  *
      DO 480 I = 1, NCOW
        CALL COWETA(ETA, ETA(1,I), EPA(1,I), CURBCA(1,I), HRDCOW(35,I),
     1       HRDCOW(19,I), OLDLAC(1,I), HRDCOW(23,I), HRDCOW(27,I),
     2       HRDCOW(31,I), HRDCOW(2,I), HRDCOW(3,I), HRDCOW(6,I),
     3       OSTART(I), OSTOP(I), DSTART(I), DSTOP(I), OFIN(I), DFIN(I),
     4       I, 4)
        IF (HRDCOW(6,I) .LE. 0.0) GO TO 470
        SCORE(1,I) = CURBCA(1,I) * SINDEX(1) + CURBCA(2,I) * SINDEX(2) +
     1   CURBCA(3,I) * SINDEX(3) + CURBCA(4,I) * SINDEX(4)
        SCORE(2,I) = EPA(1,I) * SINDEX(1) + EPA(2,I) * SINDEX(2) + EPA(
     1  3,I) * SINDEX(3) + EPA(4,I) * SINDEX(4)
        SCORE(3,I) = ETA(1,I) * SINDEX(1) + ETA(2,I) * SINDEX(2) + ETA(
     1  3,I) * SINDEX(3) + ETA(4,I) * SINDEX(4)
        GO TO 480
  470   SCORE(1,I) = 0.0
        SCORE(2,I) = 0.0
        SCORE(3,I) = 0.0
  480 CONTINUE
  490 CONTINUE
      CALL CULSEL(HRDCOW, AGE, OLDAT, OLDRB, SCORE, OFIN, DFIN, OSTART,
     1     DSTART, OSTOP, DSTOP, OYRLAC, DYRLAC, OAPRD, DAPRD, OLDLAC,
     2     OLDPK, DUEPK, TAPRD, TACPRD, PACBCA, ACAGE, ACTDP, NACLAC,
     3     DRYPRD, NPCUL, NCOW)
```

```
C  *
C  *    Output cows records                                                *
C  *
      CALL SUMCOW(HRDCOW, OYRLAC, DYRLAC, OAPRD, DAPRD, OLDLAC, EPA,
     1     ETA, SCORE, AVG, IAJBCA, OSTART, DSTART, OSTOP, DSTOP, OFIN,
     2     DFIN, OLDAT, AGE, NCOW, IYR, DPROJ, DRYFD, DAYR, GEST, LINE,
     3     AUTO)
      WRITE (12) NCOW
      IF (NCOW .LE. 0) GO TO 500
      CALL WRTMAT(HRDCOW, 40, NCOW, 12)

  500 CALL SUMHRD(TRPRD, PBCA, TAGE, TDP, NRLAC, TACPRD, PACBCA, ACAGE,
     1     ACTDP, NACLAC, TAPRD, SINDEX, NYCUL, NCOW, IHRD, IYR)
      CALL PBLUP
      RETURN
      END
      SUBROUTINE BREED(BULREC, IBULLS, IPROP, ISPMAT, ISPBUL, COWMAT,
     1           MATBUL, CULCAF, NBULS, NSPM, ISPA, NBS, NHM, NCFCUL,
     2           CHEAPB, AUTO, GSD, GSIGMA, BRDMIN, HRDET, FCYCL,
     3           GEST, DAYR)
C  ********************************************************************
C  *                                                                  *
C  *  This subroutine finds a bull to mate the cow.  If               *
C  *  the cow is to be special mated, BULL is given the appropriate   *
C  *  value in MATBUL.  If the resultant calf is to be culled BULL is *
C  *  assigned the specified bull in IBULLS.  Otherwise the BULL is   *
C  *  selected at random from IBULLS with the chance of each being    *
C  *  selected equal to the proportion indicated in IPROP.  If        *
C  *  possible sire daughter matings are avoided.  If the cow         *
C  *  conceives befoe the end of the year and befor OPNMAX a new      *
C  *  Futdat and young calf are generated.                            *
C  ********************************************************************
C
      IMPLICIT INTEGER(C)
      COMMON ISEED, EWK(4), GWK(4) /YNG/ YGCALF(8,200), NYCLF
      DIMENSION BULREC(13,20), ISPMAT(10), ISPBUL(10), IBULLS(20),
     1          IPROP(20), COWMAT(100), MATBUL(100), CULCAF(20),
     2          BV(4), ETA(4), GSIGMA(4), GSD(4), SAC(2), FERC(2),
     3          NBSV(20), NBSSV(20), RNG(4), IBUSE(5)
      REAL *4  BVPS(4), PPS(4), BVUS(4), PUS(4), BVYS(4), PYS(4)
C
      LOGICAL*1 TRUE /T/, FALSE /F/, CULL, INB, SPBR, AUTO
C
      ENTRY REBRD
      DO 10 I = 1, NBULS
        NBSV(I) = 0
   10 NBSSV(I) = 0
      DO 15 I = 1, 4
        BVPS(I) = 0.
        PPS(I) = 0.
        BVUS(I) = 0.
        PUS(I) = 0.
        BVYS(I) = 0.
        PYS(I) = 0.
   15 CONTINUE
      ICPS = 0
      ICUS = 0
      ICYS = 0
      NYCLF = 0
      NSVK = 0
      VMC = SQRT(0.5)
      TCINT = 0.0
```

```
      NCHB = NCFCUL
      NHMB = NHM
      NSPMB = 1
      RETURN
      ENTRY MATE(BV,SAC,IRANK,FERC,DUEDAT,FUTDAT,BRLAST,ISEMAX,OPNMAX,
     1      CULL)
      COW = SAC(1)
      ISIRE = SAC(2)
      CULL = FALSE
      CHECK = -2
      INB = FALSE
      IF (AUTO) GO TO 80
   20 IF (NCHB .LE. 0) GO TO 30
C
C   MATINGS TO A CHEAP BULL
C
      IF (COW .LT. CULCAF(NCHB)) GO TO 30
      NCHB = NCHB - 1
      IF (COW .NE. CULCAF(NCHB)) GO TO 20
      LCBUL = CHEAPB
      BULL = IBULLS(LCBUL)
      CLFKEP = 3
      GO TO 220
   30 IF (NHMB .LE. 0) GO TO 50
C
C   HAND MATINGS
C
      IF (COW .LT. COWMAT(NHMB)) GO TO 50
      IF (COW .EQ. COWMAT(NHMB)) GO TO 40
      NHMB = NHMB - 1
      GO TO 30
   40 CLFKEP = 2
      LCBUL = MATBUL(NHMB)
      BULL = IBULLS(LCBUL)
      NHMB = NHMB - 1
      IF (IBULLS(LCBUL) .EQ. ISIRE) INB = TRUE
      GO TO 220
   50 IF (ISPA .GT. 1) GO TO 80
C
C   SPECIAL MATINGS
C
   60 IF (NSPMB .GT. NSPM) GO TO 110
      IF (COW .LT. ISPMAT(NSPMB)) GO TO 110
      IF (COW .EQ. ISPMAT(NSPMB)) GO TO 70
      NSPMB = NSPMB + 1
      GO TO 60
   70 LCBUL = ISPBUL(NSPMB)
      CLFKEP = 1
      BULL = IBULLS(LCBUL)
      IF (IBULLS(LCBUL) .EQ. ISIRE) INB = TRUE
      NSPMB = NSPMB + 1
      GO TO 220
   80 IF (IRANK .GT. NSPM) GO TO 110
C
C   SPECIAL MATINGS BY RANK
C
      CLFKEP = 1
   90 LCBUL = IRAND(NBS)
      BULL = IBULLS(LCBUL)
      IF (IBULLS(LCBUL) .NE. ISIRE) GO TO 220
      DO 100 J = 1, NBS
        IF (IBULLS(J) .NE. ISIRE) GO TO 90
```

```
  100 CONTINUE
      INB = TRUE
      GO TO 220
C
C   RANDOM MATINGS
C
  110 J = (NBULS/2) + 1
      CLFKEP = 2
  120 NRAN = IRAND(IPROP(NBULS))
      IF (NRAN .LE. IPROP(J)) GO TO 130
      K = J + 1
      IF (NRAN - IPROP(K)) 170, 170, 150
  130 CONTINUE
      DO 140 K = 1, J
        IF (NRAN .LE. IPROP(K)) GO TO 170
  140 CONTINUE
  150 J = J + 2
      DO 160 K = J, NBULS
        IF (NRAN .LE. IPROP(K)) GO TO 170
  160 CONTINUE
  170 BULL = IBULLS(K)
      LCBUL = K
      IF (IBULLS(LCBUL) .NE. ISIRE) GO TO 220
C
C***     Selects up to three bulls at random                          ***
C
      CHECK = CHECK + 1
      IF (CHECK) 180, 110, 210
C
C   CHECK IF INBREEDING AVOIDABLE
C
  180 IPL = 0
      DO 190 I = 1, NBULS
        IPS = IPROP(I) - IPL
        IF (IBULLS(I) .NE. ISIRE .AND. IPS .GT. 0) GO TO 110
        IPL = IPROP(I)
  190 CONTINUE
  200 INB = TRUE
      GO TO 220
C
C   ASSIGN TO THE FIRST BULL THAT IS NOT RELATED
C
  210 BULL = IBULLS(I)
      LCBUL = I
  220 CONTINUE
C
C   BREEDING
C
      IF (BRLAST .LT. 0.0) GO TO 230
      NSVC = 0
      DAYCYC = FRAND(0.) * FCYCL
      OPNDAY = BRDMIN + DAYCYC
      GO TO 240
  230 DLB = AINT(BRLAST)
      OPNDAY = DLB + FCYCL - DUEDAT
      NSVC = (DLB - BRLAST) * 100. + 0.5
      BRLAST = 1.0
      IF (OPNDAY .GT. OPNMAX) GO TO 270
  240 FERB = BULREC(6,LCBUL)
      ISTP = (OPNMAX - OPNDAY) / FCYCL + 1.
      OPNDAT = DUEDAT + OPNDAY
      IF (OPNDAT .GT. DAYR) RETURN
```

```
      MYRST = (DAYR - OPNDAT) / FCYCL + 1.
      IF (MYRST .LT. ISTP) ISTP = MYRST
      HDET = HRDET * FERC(1)
      FERT = FERB * FERC(2)
C***       Loop for estrous cycle, stops if:  conception;  culled;    **
C***                                          or end of year;          **
      DO 250 I = 1, ISTP
        DPROB = FRAND(0.)
        IF (DPROB .GE. HDET) GO TO 250
        OPROB = FRAND(0.)
        NSVC = NSVC + 1
        NBSV(LCBUL) = NBSV(LCBUL) + 1
        IF (OPROB .LT. FERT) GO TO 280
        IF (NSVC .GE. ISEMAX) GO TO 270
  250 OPNDAY = OPNDAY + FCYCL
      IF (OPNDAY .GT. OPNMAX) GO TO 270
  260 BRLAST = AINT(DUEDAT + OPNDAY - FCYCL - DAYR - 0.5) - FLOAT(NSVC)
     1 / 100.
      RETURN
  270 CULL = TRUE
      BRLAST = DUEDAT + OPNDAY - FCYCL
      RETURN
C
C   GENERATE CALF
C
  280 CINT = OPNDAY + GEST
      FUTDAT = DUEDAT + CINT
      IF (FUTDAT .LT. 0.) GO TO 270
      NBSSV(LCBUL) = NBSSV(LCBUL) + 1
      NSVK = NSVK + NSVC
      TCINT = TCINT + CINT
      NYCLF = NYCLF + 1
      YGCALF(2,NYCLF) = BULL
      YGCALF(3,NYCLF) = COW
      YGCALF(4,NYCLF) = FUTDAT
      YGCALF(1,NYCLF) = CLFKEP + ISIRE / 10000
  290 CALL MNDG1(ISEED, GSIGMA, RNG, GWK)
      DO 300 J = 1, 4
  300 YGCALF(J + 4,NYCLF) = 0.5 * (BV(J) + BULREC(J + 8,LCBUL)) + VMC *
     1RNG(J) * GSD(J)
      IF ( .NOT. INB) GO TO 320
      YGCALF(2,NYCLF) = BULL + 0.1
  320 RETURN
      ENTRY SUMATE(BCON,SVCF,HCINT,SEMC, BVPS, PPS, BVUS, PUS, BVYS,
     1      PYS, IHRD, IYR)
      NSV = 0
      NCON = 0
      SEMC = 0.0
      DO 330 I = 1, NBULS
        NBC = NBSSV(I)
        NSV = NSV + NBSV(I)
        NCON = NCON + NBC
        IBUSE(1) = IHRD
        IBUSE(2) = IYR
        IBUSE(3) = IBULLS(I)
        IBUSE(4) = NBSV(I)
        IBUSE(5) = NBC
        SEMC = SEMC + IBUSE(4) * BULREC(7,I)
        WRITE (14) IBUSE
        IF (BULREC(I,8) .LT. 20.) GO TO 323
        DO 321 J = 1, 4
          BVPS(J) = BVPS(J) + BULREC(J+8,I) * NBC
```

```
321   CONTINUE
      PPS(1) = PPS(1) + NBSV(I)
      ICPS = ICPS + NBC
      PPS(3) = PPS(3) + BULREC(6,I) * NBC
      PPS(4) = PPS(4) + BULREC(7,I) * NBC
      GO TO 330
323   IF (BULREC(I,8) .LT. 1.) GO TO 327
      DO 324 J = 1, 4
        BVUS(J) = BVUS(J) + BULREC(J+8,I) * NBC
324   CONTINUE
      PUS(1) = PUS(1) + NBSV(1)
      ICUS = ICUS + NBC
      PUS(3) = PUS(3) + BULREC(6,I) * NBC
      PUS(4) = PUS(4) + BULREC(7,I) * NBC
      GO TO 330
327   DO 328 J = 1, 4
        BVYS(J) = BVYS(J) + BULREC(J+8,I) * NBC
328   CONTINUE
      PYS(1) = PYS(1) + NBSV(1)
      ICYS = ICYS + NBC
      PYS(3) = PYS(3) + BULREC(6,I) * NBC
      PYS(4) = PYS(4) + BULREC(7,I) * NBC
330 CONTINUE
    IF (ICPS .LE. 0) GO TO 340
     DO 335 J = 1, 4
335  BVPS(J) = BVPS(J) / ICPS
     PPS(2) = ICPS
     PPS(3) = PPS(3) / ICPS
     PPS(4) = PPS(4) / ICPS
340 IF (ICUS .LE. 0) GO TO 360
     DO 350 J = 1, 4
350  BVUS(J) = BVUS(J) / ICUS
     PUS(2) = ICUS
     PUS(3) = PUS(3) / ICUS
     PUS(4) = PUS(4) / ICUS
360 IF (ICYS .LE. 0) GO TO 380
     DO 370 J = 1, 4
370  BVYS(J) = BVYS(J) / ICYS
     PYS(2) = ICYS
     PYS(3) = PYS(3) / ICYS
     PYS(4) = PYS(4) / ICYS
380 CONTINUE
    SVCF = 0.
    HCINT = 0
    BCON = 0.
    IF (NYCLF .LE. 0) GO TO 390
    TLCF = NYCLF
    SVCF = FLOAT (NSVK) / TLCF
    HCINT = TCINT / TLCF
390 BCON = FLOAT (NCON) / (FLOAT (NSV) + 1.E-6)
    RETURN
    END


    SUBROUTINE INLAC(PHENV, HENV, ESIGMA, TCESD, AVG)
C  *******************************************************************
C  *                                                                 *
C  *     This subroutine generates a new lactation for DUELAC and    *
C  *  moves the calving date up.                                     *
C  *******************************************************************
```

```
      COMMON ISEED, EWK(4), GWK(4)
      DIMENSION DPROD(4), OLDLAC(4), DUELAC(4), PECOW(4), BV(4),
     1          TCESD(4), AVG(4), RNE(4), ESIGMA(4), HENV(4), PHENV(4)
      ENTRY RELAC
      CINT = 0.0
      NBORN = 0
      RETURN
      ENTRY LACT(OLDLAC,DUELAC,PECOW,BV,OLDRB,DUERB,OLDAT,DUEDAT,FUTDAT)
      DO 10 I = 1, 4
        OLDLAC(I) = DUELAC(I)
   10 CONTINUE
      CALL MNDG1(ISEED, ESIGMA, RNE, EWK)
      DO 20 I = 1, 3
        DPROD(I) = RNE(I) * TCESD(I) + HENV(I) + PHENV(I) + PECOW(I) +
     1     BV(I)
   20 DUELAC(I) = AVG(I) + DPROD(I)
      DPROD(4) = RNE(4) * TCESD(4) + HENV(4) + PHENV(4) + PECOW(4) +
     1     BV(4)
      DUELAC(4) = AVG(4) + DPROD(4)
      IF (DUELAC(4) .LT. OLDLAC(4)) DUELAC(4) = OLDLAC(4)
      DPROD(4) = DUELAC(4) - AVG(4)
      OLDRB = DUERB
      IF (DUERB .LT. 3.) DUERB = DUERB + 1.
      OLDAT = DUEDAT
      DUEDAT = FUTDAT
      FUTDAT = -1.0
      IF (OLDLAC(1) .LE. 0.0) RETURN
      CINT = CINT + DUEDAT - OLDAT
      NBORN = NBORN + 1
      RETURN
      ENTRY SUMLAC(CINT,NBORN)
      RETURN
      END


      SUBROUTINE INPROD(DPMIN, DRYMIN, DAYR, DPROJ, AJUSM, FDCAR, FDFAT,
     1          FDPRO, FDAY, ATCP, BSLP, WEKPK, TLPROD, TPROD, PPROD,
     2          M, N)
C ********************************************************************
C *   This subroutine adjusts the M. E. 305 lactation for age and    *
C *  then calculates the portion that is produced in the current year*
C ********************************************************************
      REAL*4 AJUSM(3,4), TLPROD(M), PPROD(N), TPROD(M), ATCP(3),
     1     BSLP(3), WEKPK(3)

      DPROJ = 90.
      RETURN
      ENTRY PROD(AGE,RB,PEAK,FRSH,FUTDAT,DSTART,DSTOP,DFIN,TLPROD,PPROD,
     1      TPROD,DPMIN,M,N)
C***     Age adjustment                                               ***
      IPAGE = AGE + FRSH / DAYR - 2.
   10 DO 20 I = 1, 3
   20 TPROD(I) = TLPROD(I)
      IF (IPAGE .GT. 3) IPAGE = 4
   30 DO 40 I = 1, 3
        TPROD(I) = TPROD(I) * AJUSM(I,IPAGE)
   40 CONTINUE
      IREC = RB
      PEAK = TPROD(1) * BSLP(IREC) + ATCP(IREC)
      WKPK = WEKPK(IREC)
C
```

```
C***     Find values for Wood's equation by estimating B and          ***
C***     solving for A and C until within a unit of M. E. milk.       ***
C
      STPMAX = 100000.
      IF (DSTART .GT. 0.) GO TO 80
      PB = 0.
      PPRD = 305. * PEAK
      WDB = .999
      DO 60 I = 1, 20
        WDA = PEAK * EXP(WDB) / WKPK ** WDB
        WDC = WDB / WKPK
        TAR = AREAG (WDA, WDB, WDC, 0.0, 305.0)
        DIFP = TPROD(1) - TAR
        IF (ABS(DIFP) .LE. 1.) GO TO 70
        DPP = PPRD - TAR
        DPB = PB - WDB
        DPUB = ABS(DPP/DPB)
        TPB = WDB
        WDB = WDB - DIFP / DPUB
        IF (ABS(WDB - TPB) .GT. ABS(WDB - PB)) GO TO 60
        PB = TPB
        PPRD = TAR
   60 CONTINUE
      WRITE (6,240)
   70 RB = IREC + WDB
      GO TO 90
      ENTRY AJPROD(DSTART,DSTOP,RB,PEAK,FRSH,TPROD,PPROD,DPMIN,STPMAX,M,
     1       N)
   80 IREC = RB
      WDB = RB - IREC
      WDA = PEAK * EXP(WDB) / WKPK ** WDB
      WDC = WDB / WKPK
      IF (STPMAX .LE. 0.) GO TO 110
C
C***     Calculate production for the current portion of the lactation**
C
   90 DSTOP = DRYOFF(DPMIN,WKPK,PEAK,WDA,WDB) * 7.
      DSTOP = AMAX1(DSTOP,DPROJ)
      IF (STPMAX .GE. 100000.) GO TO 100
      IF (DSTOP .GE. STPMAX) RETURN
      IF (DSTOP .LE. DSTART) GO TO 190
      GO TO 110
  100 DFIN = DAYR
      IF (FUTDAT .GE. 0.) DFIN = AMIN1(DFIN,FUTDAT - DRYMIN)
      DSTOP = AMIN1(DSTOP,DFIN - FRSH)
  110 DFIN = DSTOP + FRSH
      IF (DFIN .LE. 0.0) GO TO 200
  170 PPROD(1) = AREAG (WDA, WDB, WDC, DSTART, DSTOP)
      FLAC = PPROD(1) / TPROD(1)
      PPROD(2) = FLAC * TPROD(2)
      PPROD(3) = FLAC * TPROD(3)
      DPRD = DSTOP - DSTART
      PPROD(4) = DPRD * FDAY + PPROD(2) * FDFAT + PPROD(3) * FDPRO + (
     1PPROD(1) - PPROD(2) - PPROD(3)) * FDCAR
      IF (DSTOP .LT. 305. .AND. DFIN .EQ. DAYR) RETURN
      ART = (PEAK - ATCP(IREC)) / BSLP(IREC) + AREAG (WDA,WDB,WDC,
     1     305.0, DSTOP)
      FLAC = ART / TPROD(1)
      TPROD(1) = ART
      TPROD(2) = FLAC * TPROD(2)
      TPROD(3) = FLAC * TPROD(3)
      RETURN
```

```
  190 DFIN = 1.0
      DSTOP = DFIN - FRSH
  200 DO 210 I = 1, 4
        PPROD(I) = 0.0
  210 CONTINUE
      RETURN
C
C***     Production for 305 day or less for rolling herd averages     ***
C
      ENTRY RHPROD(AGE,RB,PEAK,FRSH,DSTOP,STOP,TPROD,TLPROD,M,N)
      STOP = DSTOP
      IPAGE = AGE + FRSH / DAYR - 2.
      IF (IPAGE .GT. 3) IPAGE = 4
      TPROD(1) = TLPROD(1) * AJUSM(1,IPAGE)
      TPROD(2) = TLPROD(2) * AJUSM(2,IPAGE)
      TPROD(3) = TLPROD(3) * AJUSM(3,IPAGE)
      IF (STOP .GE. 305.) GO TO 230
      IREC = RB
      WDB = RB - IREC
      PEAK = TPROD(1) * BSLP(IREC) + ATCP(IREC)
      WDA = PEAK * EXP(WDB) / WKPK ** WDB
      WDC = WDB / WKPK
      RPROD = AREAG (WDA, WDB, WDC, 0.0, DSTOP)
  220 FLAC = RPROD / TPROD(1)
      TPROD(1) = RPROD
      TPROD(2) = FLAC * TPROD(2)
      TPROD(3) = FLAC * TPROD(3)
      TLPROD(1) = TPROD(1) / AJUSM(1,IPAGE)
      TLPROD(2) = TPROD(2) / AJUSM(2,IPAGE)
      TLPROD(3) = TPROD(3) / AJUSM(3,IPAGE)
      RETURN
  230 STOP = 305.
  240 FORMAT ('   PROBLEMS WITH LACTATION CURVE')
      RETURN
      END


      FUNCTION AREAG (A, B, C, TF, TL)
C
C***     This function calculates the area under Wood's lactation     ***
C***     curve from TF to TL using the GAMMDS function                ***
C***                                                                  ***
C
      AREAG = 0.0
      IF (TF .GE. TL) RETURN
      B1 = B + 1.
      C1 = C / 7.
      A1 = (A / 7.0**B1) * 7.0 / C1**B1
      T1 = TF * C1
      T2 = TL * C1
      AREAG = A1 * (GAMMDS(T2, B1, IC) - GAMMDS(T1, B1, IC))
      RETURN
      END


      FUNCTION GAMMDS (Y, P, IFAULT)
C
C     CHI-LEUNG LAU 1980      APPLIED STATISTICS VOL. 29, NO. 1
C
C                  INCOMPLETE GAMMA INTERVAL
C
```

```
      DATA E /1.0E-3/
C
      IFAULT = 1
      GAMMDS = 0.0
      IF (Y .LE. 0.0 .OR. P .LE. 0.0) RETURN
      IFAULT = 2
C
      F = EXP(P * ALOG(Y) - ALGAMA(P+1.0) - Y)
      IF (F .EQ. 0.0) RETURN
      IFAULT = 0
C
      C = 1.0
      GAMMDS = 1.0
      A = P
    1 A = A + 1.0
      C = C * Y / A
      GAMMDS = GAMMDS + C
      IF (C / GAMMDS .GT. E) GO TO 1
      GAMMDS = GAMMDS * F
      RETURN
      END



      FUNCTION DRYOFF(PCUL, WKPK, PEAK, A, B)
      IF (PCUL .GE. PEAK) GO TO 30
      XK = WKPK / B * ABS(ALOG(PCUL) - ALOG(A))
      T = AMAX1(XK,WKPK)
      IF (PCUL .GT. A) XK = -XK
      DO 10 I = 1, 20
        DRYOFF = WKPK * ALOG(T) + XK
        DIF = ABS(T - DRYOFF)
        IF (DIF .LT. 0.1) RETURN
        T = DRYOFF
   10 CONTINUE
      WRITE (6,20)
   20 FORMAT ('   PROBLEMS WITH DAY DRY')
   30 DRYOFF = WKPK
      RETURN
      END



      SUBROUTINE INETA(AI, SINDEX, AVG, IAJBCA, HER, REP, DAYR, NOAI,
     1           BBNO, DPROJ, DETA, ETA, EPA, SCUR, SUMDEV, SAJDEV,
     2           SETDEV, OLDLAC, DUELAC, N)
C ***************************************************************
C  *
C *
C  *  This subroutine calculates the eta's for cows, using their
C  *
C  *  records, their dams eat'a and their sires eat's.
C  *
C *                                                             *
C ***************************************************************
      DIMENSION REP(4), HER(4), FA(4,3), FK(4,3), SETA(4), DETA(N),
     1          ETA(N), SDEV(4), AVG(4), FDEN(3), ADEV(4), EPA(N),
     2          PDEV(4), SCUR(N), IAJBCA(4), SAJDEV(N), SETDEV(N),
```

```
     3           AETA(4,250), AI(27,500), COWS(250), OLDLAC(N),
     4           DUELAC(N), HPA(4), HTA(4), SINDEX(4), SUMDEV(N)
      DO 20 J = 1, 4
        P = 1.
        R = 1. - REP(J)
        DO 10 K = 1, 3
          FDEN(K) = REP(J) - (1 - P) * HER(J)
          FA(J,K) = R / FDEN(K)
          FK(J,K) = P * HER(J) / FDEN(K)
   10   P = P - 0.25
   20 CONTINUE
      ENTRY ALETA(COWS,HPA,HTA,NT)
      RETURN
      ENTRY COWETA(AETA,ETA,EPA,SCUR,DETA,DUELAC,OLDLAC,SUMDEV,SAJDEV,
     1      SETDEV,SIRE,DAM,REC,OSTART,OSTOP,DSTART,DSTOP,OFIN,DFIN,
     2      NCOW,N)
      IF (OFIN .LE. 0.0) GO TO 40
      IF (OSTOP .LT. DPROJ) GO TO 40
      IF (OSTART .GE. 305.) GO TO 40
      DO 30 J = 1, 4
        TCUR = OLDLAC(J)
        SUMDEV(J) = SUMDEV(J) + TCUR - AVG(J)
        SAJDEV(J) = SAJDEV(J) + TCUR - HPA(J)
   30 SETDEV(J) = SETDEV(J) + TCUR - HTA(J)
   40 IF (DFIN .LE. 0.0) GO TO 80
      IF (DSTOP .LT. DPROJ) GO TO 80
      IF (DFIN .EQ. DAYR .AND. DSTOP .LT. 305.) GO TO 60
      IF (DSTART .GE. 305.) GO TO 60
      DO 50 J = 1, 4
        TCUR = DUELAC(J)
        SCUR(J) = (TCUR - HPA(J)) / AVG(J) * 100.
        SUMDEV(J) = SUMDEV(J) + TCUR - AVG(J)
        SAJDEV(J) = SAJDEV(J) + TCUR - HPA(J)
   50 SETDEV(J) = SETDEV(J) + TCUR - HTA(J)
      GO TO 120
   60 DO 70 J = 1, 4
   70 SCUR(J) = (DUELAC(J) - HPA(J)) / AVG(J) * 100.
   80 IF (OFIN .LE. 0.0) GO TO 100
      DO 90 J = 1, 4
   90 SCUR(J) = (OLDLAC(J) - HPA(J)) / AVG(J) * 100.
      GO TO 120
  100 DO 110 J = 1, 4
        EPA(J) = 0.0
  110 SCUR(J) = 0.0
  120 IF (REC .LE. 0.0) GO TO 150
      DO 130 J = 1, 4
        EPA(J) = (SUMDEV(J) / REC - HPA(J) + AVG(J)) * 100. / AVG(J)
     1  * (REP(J) * REC / (1 + (REC - 1)*REP(J)))
  130 ADEV(J) = SETDEV(J) / REC / AVG(J) * 100.
      INF = 1
      GO TO 170
      ENTRY CLFETA(SIRE,DAM,ETA,DETA,N)
      DO 140 J = 1, 4
  140 DETA(J) = 0.0
      ENTRY YNGETA(SIRE,DAM,ETA,DETA,N)
  150 INF = 0
      DO 160 J = 1, 4
  160 ADEV(J) = 0.0
  170 IF (DAM .EQ. 0.) GO TO 250
C
C***     Find the dam                                                ***
```

```
C
      MIN = 1
      MAX = NCOW
      NT = NT + 1
      IF (NT .GT. MAX) NT = MIN
      IF (DAM - COWS(NT)) 200, 220, 180
  180 NT = NT + 1
      IF (DAM - COWS(NT)) 240, 220, 190
  190 IF (MIN .GE. MAX) GO TO 240
      MIN = NT + 1
      GO TO 210
  200 IF (MIN .GE. MAX) GO TO 240
      MAX = NT - 1
  210 NT = MIN + (MAX - MIN) / 2
      IF (DAM - COWS(NT)) 200, 220, 190
  220 DO 230 J = 1, 4
  230 DETA(J) = AETA(J,NT)
  240 IF (DETA(1) .EQ. 0.0) GO TO 270
      INF = INF + 1
      GO TO 270
  250 DO 260 J = 1, 4
  260 DETA(J) = 0.0
  270 IF (SIRE .LE. BBNO .OR. SIRE .GT. BBNO + NOAI) GO TO 290
      LOCS = SIRE - BBNO
      IF (AI(9,LOCS) .LT. 50.) GO TO 290
      DO 280 J = 1, 4
  280 SETA(J) = AI(J + 12,LOCS)
      INF = INF + 1
      GO TO 320
  290 DO 300 J = 1, 4
  300 SETA(J) = 0.0
  310 IF (INF .EQ. 0) GO TO 340
  320 DO 330 J = 1, 4
         BKR = FK(J,INF) * REC
         DEN = FA(J,INF) + REC
         FNUM = DEN - BKR
         ETA(J) = 0.5 * (FNUM*(SETA(J) + DETA(J)) + BKR*ADEV(J)) / DEN
  330 CONTINUE
      RETURN
  340 DO 350 J = 1, 4
  350 ETA(J) = 0.0
  360 CONTINUE
      RETURN
      END


      SUBROUTINE SELEC(ISEL, AUTO, QTA, XCES, DPROJ, DAYR, FDRY, PRDCUL)
C   ******************************************************************
C   *                                                                *
C   *  This subroutine selects the cows to be culled for low production*
C   *  or type score                                                 *
C   *                                                                *
C   ******************************************************************
      DIMENSION OLDAT(250), SCORE(3,250), OFIN(250), DFIN(250),
     1          OSTART(250), DSTART(250), OSTOP(250), DSTOP(250),
     2          OYRLAC(4,250), DYRLAC(4,250), AGE(250), CULIST(23,250),
     3          FCUL(9,250), HRDCOW(40,250), CID(9), TPROD(4),
     4          OAPRD(3,250), DAPRD(3,250), TRPRD(3), PBCA(4),
     5          OLDRB(250), OLDLAC(4,250), DUEPK(250), OLDPK(250),
     6          RPRD(3), DCUL(250) ,CNXT(250)
      LOGICAL*1 FALSE /F/, TRUE /T/, AUTO, SPSEL
      REAL*4 CID /'HEIF', 'KEEP', 'KEEP', 'FERT', ' BCA', ' EPA',
```

```
     1       ' ETA', 'HLTH', 'DEAD'/
      RETURN
      ENTRY CULSEL(HRDCOW,AGE,OLDAT,OLDRB,SCORE,OFIN,DFIN,OSTART,DSTART,
     1       OSTOP,DSTOP,OYRLAC,DYRLAC,OAPRD,DAPRD,OLDLAC,OLDPK,DUEPK,
     2       TPROD,TRPRD,PBCA,TAGE,TDP,NRLAC,DAYDRY,NVC,NCOW)
      PRODM = TPROD(1) / 103.2
      QUOTA = QTA + XCES
      SPSEL = FALSE
      NVC = 0
      NOC = NCOW
      NMC = 0
      NNC = 0
      CSEL = ISEL + 4
      DO 20 I = 1, NOC
        IF (HRDCOW(6,I) .LE. 0.) GO TO 20
        NMC = NMC + 1
        CULIST(1,NMC) = I
        CULIST(2,NMC) = HRDCOW(1,I)
        CULIST(3,NMC) = SCORE(ISEL,I)
        CULIST(4,NMC) = HRDCOW(7,I)
        CULIST(5,NMC) = DSTART(I)
        CULIST(6,NMC) = OFIN(I)
        CULIST(7,NMC) = DFIN(I)
        CULIST(8,NMC) = OSTOP(I)
        CULIST(9,NMC) = DSTOP(I)
        DO 10 J = 1, 3
          CULIST(J + 9,NMC) = OYRLAC(J,I)
          CULIST(J + 13,NMC) = DYRLAC(J,I)
          CULIST(J + 17,NMC) = OAPRD(J,I)
10      CULIST(J + 20,NMC) = DAPRD(J,I)
        CULIST(13,NMC) = OYRLAC(4,I)
        CULIST(17,NMC) = DYRLAC(4,I)
        DCUL(NMC) = 0.
        CNXT(I) = HRDCOW(5,I) - AINT (HRDCOW(5,I))
20 CONTINUE
      CALL ISORT(CULIST, 23, NMC, 1, NMC, 3, 3, 0)
      IC = 0
      IR = NMC + 1
      MCP = 0.8 * NMC + 1
30 IC = IC + 1
      IF (IC .EQ. MCP) GO TO 190
40 IA = CULIST(1,IC)
      NCUL = 10. * CNXT(IC) + .5
      CNXT(I) = 0.8
      IF (DSTOP(IA) .GT. DPROJ .AND. NCUL .LT. 5) GO TO 70
      IF (OSTOP(IA) .LE. OSTART(IA) .OR. OFIN(IA) .LE. 0.)
     1   GO TO 45
      STOP = OSTOP(IA)
      CALL AJPROD(OSTART(IA), STOP, OLDRB(IA), OLDPK(IA), OLDAT(IA),
     1     OAPRD(1,IA), OYRLAC(1,IA), PRDCUL, OSTOP(IA), 3, 4)
      IF (STOP .GE. OSTOP(IA) .AND. HRDCOW(7,IA) .GE. 4.) GO TO 140
      IF (STOP .GE. OSTOP(IA)) GO TO 50
      OSTOP(IA) = STOP
      OFIN(IA) = OSTOP(IA) + OLDAT(IA)
      GO TO 50
45 IF (NCUL .LT. 5) GO TO 70
50 HRDCOW(7,IA) = CSEL + HRDCOW(7,IA) - AINT(HRDCOW(7,IA))
      DO 60 J = 1, 4
        TPROD(J) = TPROD(J) - CULIST(J + 9,IC) - DYRLAC(J,IA) + OYRLAC(
     1  J,IA)
60 DYRLAC(J,IA) = 0.0
      IF (OFIN(IA) .LT. 0.0) OFIN(IA) = 0.0
```

```
      DFIN(IA) = OFIN(IA)
      DSTART(IA) = 0.0
      DSTOP(IA) = 0.0
      GO TO 120
   70 B = HRDCOW(7,IA) - AINT(HRDCOW(7,IA))
      REC = 10. * (HRDCOW(6,IA) - AINT(HRDCOW(6,IA)))
      DUERB = REC + B
      STOP = DSTOP(IA)
      IF (DSTOP(IA) .LE. DSTART(IA) .OR. DFIN(IA) .LE. 0.)
     1   GO TO 140
   80 CALL AJPROD(DSTART(IA), STOP, DUERB, DUEPK(IA), HRDCOW(8,IA),
     1      DAPRD(1,IA), DYRLAC(1,IA), PRDCUL, DSTOP(IA), 3, 4)
      IF (DSTOP(IA) .LE. STOP) GO TO 150
      HRDCOW(7,IA) = CSEL + B
   90 DFIN(IA) = STOP + HRDCOW(8,IA)
  100 DSTOP(IA) = STOP
      DO 110 J = 1, 4
  110 TPROD(J) = TPROD(J) - CULIST(J + 13,IC) + DYRLAC(J,IA)
  120 PRODM = TPROD(1) / 103.2
      CALL KILCF(HRDCOW(1,IA), OLDAT(IA), HRDCOW(8,IA), HRDCOW(9,IA),
     1      DFIN(IA))
  130 NVC = NVC + 1
  140 DCUL(IC) = DFIN(IA)
      IF (DCUL(IC) .EQ. 0.) DCUL(IC) = 1.
      GO TO 170
  150 IF (HRDCOW(7,IA) .GE. 4.) GO TO 140
      IF (DFIN(IA) .EQ. DAYR) GO TO 160
      HRDCOW(7,IA) = CSEL + B
      GO TO 130
  160 NNC = NNC + 1
      DCUL(IC) = -1 * (STOP + HRDCOW(8,IA) - DAYR)
  170 IF (SPSEL) GO TO 280
      IF (PRODM .GT. QUOTA) GO TO 30
      IC = IC + 1
      IF (IC .GE. MCP .OR. NNC .GE. NVC) GO TO 190
      MCP = MIN0(MCP,IC + NVC - NNC)
      DO 180 J = IC, MCP
        IA = CULIST(1,J)
        HRDCOW(5,IA) = AINT(HRDCOW(5,IA)) + .2
  180 CONTINUE
      IC = MCP + 1
  190 JMC = NMC + 1
      IR = 0
      DO 200 J = 1, NMC
        L = JMC - J
        IA = CULIST(1,L)
        IF (HRDCOW(7,IA) .LT. 4.) IR = IR + 1
        HRDCOW(5,IA) = IR + CNXT(IA)
  200 CONTINUE
  210 IF (AUTO) GO TO 370
      NCWL = IC
      NS = 1
  220 DO 230 I = NS, NCWL
        JF = CULIST(1,I)
        FCUL(1,I) = I
        FCUL(2,I) = HRDCOW(1,JF)
        FCUL(3,I) = DCUL(I)
        FCUL(4,I) = SCORE(1,JF)
        FCUL(5,I) = SCORE(2,JF)
        FCUL(6,I) = SCORE(3,JF)
        FCUL(7,I) = CULIST(10,I) + CULIST(14,I)
        FCUL(8,I) = OYRLAC(1,JF) + DYRLAC(1,JF)
```

```
        ISET = HRDCOW(7,JF)
        FCUL(9,I) = CID(ISET)
230 CONTINUE
240 WRITE (6,510)
    NOUT = NCWL - NS + 1
    CALL WRCUL(FCUL, 9, NOUT)
    SPSEL = TRUE
    WRITE (6,530) QTA, XCES, PRODM
    WRITE (6,540)
    READ (5,550) IOPT
    GO TO (250, 270, 320, 370), IOPT
    WRITE (6,580)
250 WRITE (6,590)
    READ (5,550) NEX
    NW = NCWL + NEX
    IF (NW .LE. NMC) GO TO 260
    WRITE (6,600) NMC
    NW = NMC
260 NCWL = NW
    GO TO 220
270 WRITE (6,560)
    READ (5,550) NCH
280 IF (NCH .LE. 0) GO TO 220
    NCH = NCH - 1
    GO TO 300
290 WRITE (6,580)
300 WRITE (6,570)
310 READ (5,550) IC
    IF (IC .GT. NMC .OR. IC .LT. 1) GO TO 290
    IF (CULIST(4,IC) .GT. 3.) GO TO 280
    GO TO 40
320 WRITE (6,560)
    READ (5,550) NRC
330 IF (NRC .LE. 0) GO TO 220
    NRC = NRC - 1
    GO TO 350
340 WRITE (6,580)
350 WRITE (6,570)
    READ (5,550) NIC
    IF (IC .GT. NMC .OR. IC .LT. 1) GO TO 340
    IA = CULIST(1,NIC)
    IF (AINT(HRDCOW(7,IA)) .NE. CSEL) GO TO 330
    NVC = NVC - 1
    DO 360 J = 1, 3
      TPROD(J) = TPROD(J) - DYRLAC(J,IA) - OYRLAC(J,IA) + CULIST(J +
   1  9,NIC) + CULIST(J + 13,NIC)
      OYRLAC(J,IA) = CULIST(J + 9,NIC)
      DYRLAC(J,IA) = CULIST(J + 13,NIC)
      OAPRD(J,IA) = CULIST(J + 17,NIC)
360 DAPRD(J,IA) = CULIST(J + 20,NIC)
    TPROD(4) = TPROD(4) - DYRLAC(4,IA) - OYRLAC(4,IA) + CULIST(13,NIC)
   1 + CULIST(17,NIC)
    OYRLAC(4,IA) = CULIST(13,NIC)
    DYRLAC(4,IA) = CULIST(17,NIC)
    HRDCOW(7,IA) = CULIST(4,NIC)
    CALL REVCF(HRDCOW(1,IA), OLDAT(IA), HRDCOW(8,IA), HRDCOW(9,IA),
   1     DFIN(IA))
    DSTART(IA) = CULIST(5,NIC)
    OFIN(IA) = CULIST(6,NIC)
    DFIN(IA) = CULIST(7,NIC)
    OSTOP(IA) = CULIST(8,NIC)
    DSTOP(IA) = CULIST(9,NIC)
```

```
      DCUL(NIC) = 0.
      PRODM = TPROD(1) / 103.2
      GO TO 330
C
C***    Adjust rolling herd averages                                    ***
C
  370 IF (NVC .LE. 0) RETURN
      IC = 0
      DO 490 I = 1, NVC
  380   IC = IC + 1
        IF (IC .GT. NMC) GO TO 500
        IA = CULIST(1,IC)
        IF (AINT(HRDCOW(7,IA)) .NE. CSEL) GO TO 380
        IF (OSTOP(IA) .EQ. CULIST(8,IC) .OR. OSTOP(IA) .GT. 305.)
     1     GO TO 440
        CALL RHPROD(AGE(IA), OLDRB(IA), OLDPK(IA), OLDAT(IA),
     1       CULIST(8,IC), STOPO, RPRD, OLDLAC(1,IA), 3, 4)
        DO 390 J = 1, 3
          PBCA(J) = PBCA(J) - OLDLAC(J,IA)
          TRPRD(J) = TRPRD(J) - RPRD(J)
  390   CONTINUE
        CALL RHPROD(AGE(IA), OLDRB(IA), OLDPK(IA), OLDAT(IA), OSTOP(IA),
     1       STOPN, RPRD, OLDLAC(1,IA), 3, 4)
        DO 400 J = 1, 3
          PBCA(J) = PBCA(J) + OLDLAC(J,IA)
          TRPRD(J) = TRPRD(J) + RPRD(J)
  400   CONTINUE
        TDP = TDP - STOPO + STOPN
  410   IF (CULIST(9,IC) .LT. 305. .AND. CULIST(7,IC) .EQ. DAYR)
     1     GO TO 490
        IF (HRDCOW(19,IA) .LE. 0.) GO TO 490
  420   REC = 10. * (HRDCOW(6,IA) - AINT(HRDCOW(6,IA)))
        DUERB = REC + HRDCOW(7,IA) - AINT(HRDCOW(7,IA))
        CALL RHPROD(AGE(IA), DUERB, DUEPK(IA), HRDCOW(8,IA),
     1       CULIST(9,IC), STOPO, RPRD, HRDCOW(19,IA), 3, 4)
        DO 430 J = 1, 3
          PBCA(J) = PBCA(J) - HRDCOW(J + 18,IA)
          TRPRD(J) = TRPRD(J) - RPRD(J)
  430   CONTINUE
        PBCA(4) = PBCA(4) - HRDCOW(22,IA)
        TDP = TDP - STOPO
        TAGE = TAGE - AGE(IA) - HRDCOW(8,IA) / DAYR + 1.
        NRLAC = NRLAC - 1
        HRDCOW(6,IA) = HRDCOW(6,IA) - 1.
        GO TO 490
  440   IF (DSTOP(IA) .LE. 0.) GO TO 410
        IF (DSTOP(IA) .GT. 305.) GO TO 490
        REC = 10. * (HRDCOW(6,IA) - AINT(HRDCOW(6,IA)))
        DUERB = REC + HRDCOW(7,IA) - AINT(HRDCOW(7,IA))
        IF (CULIST(7,IC) .EQ. DAYR .AND. CULIST(9,IC) .LT. 305.)
     1     GO TO 470
        CALL RHPROD(AGE(IA), DUERB, DUEPK(IA), HRDCOW(8,IA),
     1       CULIST(9,IC), STOPO, RPRD, HRDCOW(19,IA), 3, 4)
        DO 450 J = 1, 3
          PBCA(J) = PBCA(J) - HRDCOW(J + 18,IA)
          TRPRD(J) = TRPRD(J) - RPRD(J)
  450   CONTINUE
        CALL RHPROD(AGE(IA), DUERB, DUEPK(IA), HRDCOW(8,IA), DSTOP(IA),
     1       STOPN, RPRD, HRDCOW(19,IA), 3, 4)
        DO 460 J = 1, 3
          PBCA(J) = PBCA(J) + HRDCOW(J + 18,IA)
          TRPRD(J) = TRPRD(J) + RPRD(J)
```

```
460   CONTINUE
      TDP = TDP - STOPO + STOPN
      GO TO 490
470   CALL RHPROD(AGE(IA), DUERB, DUEPK(IA), HRDCOW(8,IA), DSTOP(IA),
   1       STOPN, RPRD, HRDCOW(19,IA), 3, 4)
      DO 480 J = 1, 3
        PBCA(J) = PBCA(J) + HRDCOW(J + 18,IA)
        TRPRD(J) = TRPRD(J) + RPRD(J)
480   CONTINUE
      PBCA(4) = PBCA(4) + HRDCOW(22,IA)
      TDP = TDP + STOPN
      TAGE = TAGE + AGE(IA) + HRDCOW(8,IA) / DAYR - 1.
      HRDCOW(6,IA) = HRDCOW(6,IA) + 1.
      NRLAC = NRLAC + 1
490 CONTINUE
    RETURN
500 WRITE (6,610)
510 FORMAT (25X, 'COWS TO BE CULLED', //, 17X, 'CULL', 6X,
   1       'INDEX SCORES', 7X, 'MAX', 4X, 'ACT', /, 4X, 'LOC', 4X,
   2       'COW', 4X, 'DAY', 3X, 'LACT', 4X, 'EPA', 4X, 'ETA', 3X,
   3       'PROD   PROD  STATUS')
520 FORMAT (8F7.0, 2X, A4)
530 FORMAT (5X, 'YOUR FLUID QUOTA IS', F7.0, ' HL  PLUS EXCESS OF',
   1       F7.0, /, 6X, 'THIS YEARS PRODUCTION WILL BE', F8.1,
   2       ' HECTOLITRES')
540 FORMAT ('   ENTER - 1  TO PRINT A LONGER LIST'/,
   1       '          - 2  TO CULL MORE COWS'/,
   2       '          - 3  TO RESTORE SELECTIVELY CULLED COWS'/,
   3       '          - 4  TO CONTINUE', /, 'OPTION ?')
550 FORMAT (6I10)
560 FORMAT ('NUMBER OF COWS TO CHANGE ?')
570 FORMAT ('COW LOCATION NUMBER ?')
580 FORMAT ('   INVALID ENTRY')
590 FORMAT ('HOW MANY MORE COWS DO YOU WANT PRINTED ?')
600 FORMAT ('   THERE ARE ONLY', I4, ' COWS ELIGABLE TO CULL')
610 FORMAT (' TROUBLE WITH PRODUCTION CULLS')
    RETURN
    END


    SUBROUTINE INBLUP(KBCA, AVG)
C   *******************************************************************
C   *                                                                 *
C   *  This subroutine stores first lactation records for sires      *
C   *  and outputs them at the end of the run                        *
C   *                                                                 *
C   *******************************************************************
    DIMENSION IBLUP(8,20), REC(4), KBCA(4), AVG(4)
    NFL = 0
    NS = 0
    RETURN
    ENTRY BLUP(SIRE, IHRD, IYR, REC)
    NHYS = IHRD * 100 + IYR
    ISIRE = SIRE
    NFL = NFL + 1
    IF (NS .EQ. 0) GO TO 20
    DO 10 J = 1, NS
      IF (IBLUP(3,J) .EQ. ISIRE) GO TO 40
 10 CONTINUE
 20 CONTINUE
    NS = NS + 1
    IBLUP(1,NS) = IHRD
```

```
      IBLUP(2,NS) = NHYS
      IBLUP(3,NS) = SIRE
      IBLUP(4,NS) = 1
      DO 30 K = 1, 3
   30 IBLUP(K + 4,NS) = REC(K) / AVG(K) * 100. + KBCA(K)
      IBLUP(8,NS) = REC(4)
      GO TO 60
   40 DO 50 K = 1, 3
   50 IBLUP(K + 4,J) = REC(K) / AVG(K) * 100. + KBCA(K) + IBLUP(
     1K + 4,J)
      IBLUP(8,J) = REC(4) + IBLUP(8,J)
      IBLUP(4,J) = IBLUP(4,J) + 1
   60 RETURN
      ENTRY PBLUP
      DO 70 I = 1, NS
   70 CALL PRTUNF (IBLUP(1,I), 8, 13)
      RETURN
      END


      SUBROUTINE YOUNG(SINDEX, ISELYL, NSYL, IYR, IHRD, DAYR, DTYLG,
     1           DTCF, FDYLG, FDCF, AUTO)
C ********************************************************************
C *                                                                  *
C *  This subroutine moves the old calves to yearling array          *
C *                                                                  *
C ********************************************************************
      COMMON ISEED, EWK(4), GWK(4) /YNG/ YGCALF(8,200), NYCLF
      DIMENSION YRLG(18,100), YCALF(8,200), OUTYG(12,200), UCALF(8,150),
     1          DOCF(2,200), DYCF(2,200), CALF(18,150), YSP(18,10),
     2          RNE(4), ESIGMA(4), PCESD(4), SINDEX(4), ACBV(4),
     3          BVLYL(4), BVSYL(4), BVDYL(4), ISELYL(50), DUMY(4)


      REAL*4 SOLD /'SOLD'/, DEAD /'DEAD'/, BLANK /'    '/
C
      LOGICAL*1 AUTO, FALSE /F/, TRUE /T/, FINISH
C
      RETURN
C
C   UPDATE YEARLINGS AND CALCULATE FEED COSTS
C
      ENTRY YNGOUT(NYLG,NYLS,NYLD,BVLYL,BVSYL,BVDYL,TYFDC)
      FINISH = FALSE
      NYLG = 0
      NS = NSYL
      NYLS = 0
      NYLD = 0
      NTCF = 0
      TYFDC = 0.0
      DO 10 I = 1, 4
        BVLYL(I) = 0.0
        BVSYL(I) = 0.0
        BVDYL(I) = 0.0
   10 CONTINUE
      READ (2) NYYLG, NOCF
      IF (NYYLG .LE. 0) GO TO 140
      CALL REAMAT(YRLG, 18, NYYLG, 2)
      DO 120 I = 1, NYYLG
        BTH = YRLG(4,I)
```

```
20    IF (NSYL .LT. 1) GO TO 30
      IYL = YRLG(1,I)
      IF (IYL .GE. ISELYL(NYLS)) GO TO 60
30    DYRLG = DAYR - BTH
      DTPROB = DTCF * BTH / DAYR + DTYLG * DYRLG / DAYR
      DTRAN = FRAND(0.)
      IF (DTRAN .LT. DTPROB) GO TO 90
      NYLG = NYLG + 1
      DO 40 J = 1, 4
        YRLG(J,NYLG) = YRLG(J,I)
        YRLG(J + 6,NYLG) = YRLG(J + 6,I)
        YRLG(J + 10,NYLG) = YRLG(J + 10,I)
        YRLG(J + 14,NYLG) = YRLG(J + 14,I)
40    BVLYL(J) = BVLYL(J) + YRLG(J + 9,I)
      YRLG(5,NYLG) = YRLG(5,I)
      YRLG(6,NYLG) = YRLG(6,I)
      YYFDC = FDCF * BTH + FDYLG * DYRLG
      TYFDC = TYFDC + YYFDC
      OUTYG(10,I) = YYFDC
      CALL YNGETA(YRLG(2,NYLG), YRLG(3,NYLG), OUTYG(5,I),
     1     YRLG(15,NYLG), 4)
      IF (AUTO) GO TO 120
      OUTYG(9,I) = 0.0
      DO 50 J = 1, 4
        OUTYG(9,I) = OUTYG(9,I) + OUTYG(J + 4,I) * SINDEX(J)
50    OUTYG(J,I) = YRLG(J,I)
      OUTYG(11,I) = BLANK
      OUTYG(12,I) = BLANK
      GO TO 120
60    NS = NS - 1
      IF (NS .LE. 0) NSYL = 0
      IF (ISELYL(NSYL - NS) .LT. IYL) GO TO 60
      IF (ISELYL(NSYL - NS) .NE. IYL) GO TO 30
      NYLS = NYLS + 1
      DO 70 J = 1, 4
        BVSYL(J) = BVSYL(J) + YRLG(J + 9,I)
70    CONTINUE
      YYFDC = FDCF * BTH
      TYFDC = TYFDC + YYFDC
      IF (AUTO) GO TO 120
      DO 80 J = 1, 4
        OUTYG(J + 4,I) = 0.0
80    OUTYG(J,I) = YRLG(J,I)
      OUTYG(9,I) = 0.0
      OUTYG(10,I) = YYFDC
      OUTYG(11,I) = SOLD
      OUTYG(12,I) = BLANK
      GO TO 120
90    DEDAT = FRAND(0.) * DAYR
      IF (DEDAT .LT. BTH) BTH = DEDAT
      NYLD = NYLD + 1
      DO 100 J = 1, 4
        BVDYL(J) = BVDYL(J) + YRLG(J + 9,I)
100   CONTINUE
      YYFDC = BTH * FDCF + (DEDAT - BTH) * FDYLG
      TYFDC = TYFDC + YYFDC
      IF (AUTO) GO TO 120
      DO 110 J = 1, 4
        OUTYG(J + 4,I) = 0.0
110   OUTYG(J,I) = YRLG(J,I)
      OUTYG(9,I) = 0.0
      OUTYG(10,I) = YYFDC
```

```
        OUTYG(11,I) = DEAD
        OUTYG(12,I) = BLANK
  120 CONTINUE
      WRITE (12) NYLG
      IF (NYLG .LE. 0) GO TO 130
      CALL WRTMAT(YRLG, 18, NYLG, 12)
  130 IF (AUTO) GO TO 140
      CALL TITLYG
      CALL WRTYG(OUTYG, 12, NYYLG)
C
C   CALVES
C
  140 IF (NOCF .LE. 0) GO TO 150
      CALL REAMAT(YCALF, 8, NOCF, 2)
      CALL CALFUP(YCALF, OUTYG, CALF, UCALF, YSP, NOCF, NTCF, NLCF,
     1     NUCF, NYSP)
  150 IF (NYCLF .LE. 0) GO TO 160
      N = 1
      CALL CALFUP(YGCALF, OUTYG, CALF, UCALF, YSP, NYCLF, NTCF, NLCF,
     1     NUCF, NYSP)
  160 WRITE (12) NLCF, NUCF
      IF (NLCF .LE. 0) GO TO 170
      CALL WRTMAT(CALF, 18, NLCF, 12)
  170 IF (NUCF .LE. 0) GO TO 180
      CALL WRTMAT(UCALF, 8, NUCF, 12)
  180 IF (NYSP .LE. 0) GO TO 200
      DO 190 I = 1, NYSP
  190 WRITE (11) (YSP(J,I),J=1,18)
  200 IF (AUTO) RETURN
      IF (NTCF .LE. 0) RETURN
      CALL TITLCF
      CALL WRTYG(OUTYG, 12, NTCF)
      RETURN
      END
C
C
C
      SUBROUTINE CALVES(ESIGMA, PCESD, PINB, SINDEX, FDCF, FERSD, FERM,
     1           HDSD, HDM, DTCF, DTUB, IHRD, DAYR, IYR, AUTO)
C ******************************************************************
C *                                                                *
C *  This subroutine moves new-born calves to calves array         *
C *                                                                *
C ******************************************************************
     COMMON ISEED, EWK(4), GWK(4) /KIL/ DOCF(2,200), DYCF(2,200), NOD,
     1      NYD /CFBV/ BVLCF(4), BVYSP(4), BVSHCF(4), BVSBCF(4),
     2      BVDCF(4), NLCF, NYSP, NSHCF, NSBCF, NDCF, TCFDC
      DIMENSION YCALF(8,200), OUTYG(12,200), UCALF(8,150), CALF(18,150),
     1          YSP(18,10), SINDEX(4), PCESD(4), ESIGMA(4), RNE(4),
     2          PINB(4), ACBV(4), DUMY(4)
      REAL*4 HEIF /'HEIF'/, BULL /'BULL'/, SOLD /'SOLD'/, SAI /'S AI'/,
     1       DEAD /'DEAD'/, BLANK /'    '/
      LOGICAL*1 AUTO, FALSE /F/, TRUE /T/, FINISH, SECRD/T/
      RETURN
      ENTRY CALFUP(YCALF,OUTYG,CALF,UCALF,YSP,NCF,NTCF,NLC,NUCF,NYS)
      N = 1
      SECRD = .NOT. SECRD
      IF (SECRD) GO TO 60
      NTCF = 0
      NSBCF = 0
      NSHCF = 0
      NLCF = 0
```

```
      NYSP = 0
      NDCF = 0
      NUCF = 0
      TCFDC = 0.0
      NYR = (IYR + 10) * 100
      IF (NOD .LT. 1) GO TO 10
      CALL ISORT(DOCF, 2, NOD, 1, NOD, 1, 3, 0)
      GO TO 20
   10 DOCF(1,1) = 9000.
   20 IF (NYD .LT. 1) GO TO 30
      CALL ISORT(DYCF, 2, NYD, 1, NYD, 1, 3, 0)
      GO TO 40
   30 DYCF(1,1) = 9000.
   40 DO 50 I = 1, 4
        BVLCF(I) = 0.0
        ACBV(I) = 0.0
        BVYSP(I) = 0.0
        BVSHCF(I) = 0.0
        BVSBCF(I) = 0.0
        BVDCF(I) = 0.0
   50 CONTINUE
   60 DO 300 I = 1, NCF
        BTH = YCALF(4,I)
        IF (SECRD) GO TO 80
        BTH = BTH - DAYR
   70   IF (YCALF(3,I) .GE. DOCF(1,N)) GO TO 260
        GO TO 90
   80   IF (YCALF(3,I) .GE. DYCF(1,N)) GO TO 270
        IF (BTH .GT. DAYR) GO TO 280
   90   NTCF = NTCF + 1
        SIRE = YCALF(2,I)
        DAM = YCALF(3,I)
        OUTYG(2,NTCF) = SIRE
        OUTYG(3,NTCF) = DAM
        OUTYG(4,NTCF) = BTH
  100   DTRAN = FRAND(0.)
        IF (DTRAN .LT. DTUB) GO TO 220
        ISTAT = YCALF(1,I)
        DO 110 J = 1, 4
  110   ACBV(J) = ACBV(J) + YCALF(J + 4,I)
        DTPROB = DTCF * BTH / DAYR
        IF (DTRAN .LT. DTPROB) GO TO 210
        SEX = FRAND(0.)
        IF (SEX .LE. 0.5) GO TO 140
        IF (ISTAT .GT. 2) GO TO 160
  120   NLCF = NLCF + 1
        CALL MNDG1(ISEED, ESIGMA, RNE, EWK)
        DO 130 J = 1, 4
          BV = YCALF(J + 4,I)
          BVLCF(J) = BVLCF(J) + BV
          CALF(J + 6,NLCF) = RNE(J) * PCESD(J)
  130   CALF(J + 10,NLCF) = BV
        BUL = YCALF(2,I)
        INB = 10. * (BUL - AINT(BUL))
        IF (INB .NE. 1) GO TO 137
        DO 135 J = 1, 4
          CALF(J+6, NLCF) = CALF(J+6, NLCF) - PINB(J)
  135   CONTINUE
  137   CALL CLFETA(SIRE, DAM, OUTYG(5,NTCF), CALF(15,NLCF), 4)
        CALF(1,NLCF) = NYR + NLCF
        CALF(2,NLCF) = YCALF(2,I)
        CALF(3,NLCF) = YCALF(3,I)
```

```
      CALF(4,NLCF) = BTH
      CALF(5,NLCF) = HDM + 1 -(HDSD * FRANDN(0.0))
      CALF(6,NLCF) = FERM + 1 -(FERSD * FRANDN(0.0))
      YCFDC = (DAYR - BTH) * FDCF
      TCFDC = TCFDC + YCFDC
      IF (AUTO) GO TO 300
      OUTYG(1,NTCF) = NYR + NLCF
      OUTYG(9,NTCF) = OUTYG(5,NTCF) * SINDEX(1) + OUTYG(6,NTCF) *
     1 SINDEX(2) + OUTYG(7,NTCF) * SINDEX(3) + OUTYG(8,NTCF) * SINDEX(
     2 4)
      OUTYG(10,NTCF) = YCFDC
      OUTYG(11,NTCF) = HEIF
      OUTYG(12,NTCF) = BLANK
      GO TO 300
140   OUTYG(11,NTCF) = BULL
      IF (ISTAT .GT. 1) GO TO 180
      NYSP = NYSP + 1
      DO 150 J = 1, 4
        BV = YCALF(J + 4,I)
        BVYSP(J) = BVYSP(J) + BV
150   YSP(J + 6,NYSP) = BV
      YSP(1,NYSP) = IYR
      YSP(2,NYSP) = YCALF(2,I)
      YSP(3,NYSP) = YCALF(3,I)
      YSP(4,NYSP) = BTH
      YSP(5,NYSP) = IHRD
      YSP(6,NYSP) = (YCALF(1,I) - ISTAT) * 10000.
      CALL CLFETA(SIRE, DAM, YSP(11,NYSP), YSP(15,NYSP), 4)
      IF (AUTO) GO TO 300
      OUTYG(1,NTCF) = NYSP
      OUTYG(5,NTCF) = YSP(11,NYSP)
      OUTYG(6,NTCF) = YSP(12,NYSP)
      OUTYG(7,NTCF) = YSP(13,NYSP)
      OUTYG(8,NTCF) = YSP(14,NYSP)
      OUTYG(9,NTCF) = OUTYG(5,NTCF) * SINDEX(1) + OUTYG(6,NTCF) *
     1 SINDEX(2) + OUTYG(7,NTCF) * SINDEX(3) + OUTYG(8,NTCF) * SINDEX(
     2 4)
      OUTYG(10,NTCF) = 0.0
      OUTYG(12,NTCF) = SAI
      GO TO 300
160   OUTYG(11,NTCF) = HEIF
      NSHCF = NSHCF + 1
      DO 170 J = 1, 4
        BVSHCF(J) = BVSHCF(J) + YCALF(J + 4,I)
170   CONTINUE
      GO TO 200
180   NSBCF = NSBCF + 1
      DO 190 J = 1, 4
        BVSBCF(J) = BVSBCF(J) + YCALF(J + 4,I)
190   CONTINUE
200   IF (AUTO) GO TO 300
      CALL CLFETA(SIRE, DAM, OUTYG(5,NTCF), DUMY, 4)
      OUTYG(1,NTCF) = 0.0
      OUTYG(9,NTCF) = 0.0
      OUTYG(10,NTCF) = 0.0
      OUTYG(12,NTCF) = SOLD
      GO TO 300
210   DEDAT = FRAND(0.) * (DAYR - BTH) + BTH
      GO TO 230
220   DEDAT = 0.0
230   NDCF = NDCF + 1
      DO 240 J = 1, 4
```

```
        BVDCF(J) = BVDCF(J) + YCALF(J + 4,I)
  240   CONTINUE
        YCFDC = (DEDAT - BTH) * FDCF
        IF (YCFDC .LT. 0.0) YCFDC = 0.0
        TCFDC = TCFDC + YCFDC
        IF (AUTO) GO TO 300
        OUTYG(10,NTCF) = YCFDC
        OUTYG(1,NTCF) = 0.0
        OUTYG(4,NTCF) = BTH
        DO 250 J = 1, 4
  250   OUTYG(J + 4,NTCF) = 0.0
        OUTYG(9,NTCF) = 0.0
        OUTYG(11,NTCF) = BLANK
        OUTYG(12,NTCF) = DEAD
        GO TO 300
  260   N = N + 1
        IF (N .GT. NOD) DOCF(1,N) = 9000.
        IF (YCALF(3,I) .GT. DOCF(1,N - 1)) GO TO 70
        IF (DOCF(2,N - 1) .EQ. BTH) GO TO 300
        IF (DOCF(1,N) .NE. YCALF(3,I) .OR. DOCF(2,N) .NE. BTH)
     1     GO TO 70
        N = N - 1
        GO TO 300
  270   N = N + 1
        IF (N .GT. NYD) DYCF(1,N) = 9000.
        IF (YCALF(3,I) .GT. DYCF(1,N - 1)) GO TO 80
        IF (DYCF(2,N - 1) .EQ. BTH) GO TO 300
        IF (DYCF(1,N) .NE. YCALF(3,I) .OR. DYCF(2,N) .NE. BTH)
     1     GO TO 80
        N = N - 1
        GO TO 300
  280   NUCF = NUCF + 1
        DO 290 J = 1, 8
  290   UCALF(J,NUCF) = YCALF(J,I)
  300 CONTINUE
      NLC = NLCF
      NYS = NYSP
      RETURN
      END


      SUBROUTINE INKIL(GEST)
C   *****************************************************************
C   *    This subroutine remove calves conceived by cows which      *
C   *    die or are sold before parturition                         *
C   *****************************************************************
      COMMON /KIL/ DOCF(2,200), DYCF(2,200), NOD, NYD
      NOD = 0
      NYD = 0
      RETURN
      ENTRY KILCF(COW,OLDAT,DUEDAT,FUTDAT,FIN)
      IF (FIN .GT. OLDAT) GO TO 10
      IF (OLDAT .LT. 0.0) GO TO 10
      NOD = NOD + 1
      DOCF(1,NOD) = COW
      DOCF(2,NOD) = OLDAT
   10 IF (FIN .GE. DUEDAT) GO TO 30
      IF (DUEDAT .GE. GEST) GO TO 20
      NOD = NOD + 1
      DOCF(1,NOD) = COW
      DOCF(2,NOD) = DUEDAT
```

```
      GO TO 30
   20 NYD = NYD + 1
      DYCF(1,NYD) = COW
      DYCF(2,NYD) = DUEDAT
   30 IF (FUTDAT .LT. 0.0) GO TO 40
      NYD = NYD + 1
      DYCF(1,NYD) = COW
      DYCF(2,NYD) = FUTDAT
   40 RETURN
      ENTRY REVCF(COW,OLDAT,DUEDAT,FUTDAT,FIN)
      IF (FIN .LT. OLDAT) GO TO 50
      IF (FIN .GT. DUEDAT) GO TO 90
      IF (DUEDAT .GT. GEST) GO TO 100
   50 N = NOD
   60 IF (COW .EQ. DOCF(1,N)) GO TO 70
      N = N - 1
      GO TO 60
   70 DOCF(1,N) = 0.0
      IF (N .EQ. 1) GO TO 80
      IF (COW .EQ. DOCF(1,N - 1)) DOCF(1,N - 1) = 0.0
   80 IF (DUEDAT .GT. GEST) GO TO 100
   90 IF (FUTDAT .LT. 0.0) RETURN
  100 N = NYD
  110 IF (COW .EQ. DYCF(1,N)) GO TO 120
      N = N - 1
      GO TO 110
  120 DYCF(1,N) = 0.0
      IF (N .EQ. 1) RETURN
      IF (COW .EQ. DYCF(1,N - 1)) DYCF(1,N - 1) = 0.0
      RETURN
      END



      SUBROUTINE SUMCOW(HRDCOW, OYRLAC, DYRLAC, OAPRD, DAPRD, OLDLAC,
     1           EPA, ETA, SCORE, AVG, IAJBCA, OSTART, DSTART, OSTOP,
     2           DSTOP, OFIN, DFIN, OLDAT, AGE, NCOW, IYR, DPROJ, DRYFD,
     3           DAYR, GEST, LINE, AUTO)
C  ******************************************************************
C  *   This subroutine accumulates years summaries for cows records  *
C  ******************************************************************
      COMMON /COWBV/ THBV(4), TCBV(4), FCBV(4), CLCBV(4), EPCBV(4),
     1       ETCBV(4), HCBV(4), DCBV(4), NBHF, NLCW, NFCUL, NLCUL,
     2       NPCUL, NTCUL, NHCUL, NDCW

      DIMENSION HRDCOW(40,250), OYRLAC(4,250), DYRLAC(4,250),
     1          OAPRD(3,250), DAPRD(3,250), EPA(4,250), ETA(4,250),
     2          OSTOP(250), DSTOP(250), AGE(250), SCORE(3,250), MBCA(3),
     3          IBCA(3), NBCA(3), AVG(4), IAJBCA(4), OSTART(250),
     4          DSTART(250), OFIN(250), DFIN(250), OLDAT(250),
     5          OLDLAC(4,250)

      REAL*4 POPEN /'OPEN'/, PBRED /'BRED'/, PFERT /
     1       'FERT'/, PLACT /'BCA '/, PEPA /'EPA '/, PETA /'ETA '/,
     2       PPROB /'HLTH'/, PDEAD /'DEAD'/

      LOGICAL*1 AUTO
      NC = 0
      NSH = 0
      NLCW = 0
```

```
      NFCUL = 0
      NLCUL = 0
      NPCUL = 0
      NTCUL = 0
      NHCUL = 0
      NDCW = 0
      NBHF = 0
      NOCW = NCOW
      NCOW = 0
      DO 10 J = 1, 4
        TCBV(J) = 0.
        THBV(J) = 0.
        FCBV(J) = 0.
        CLCBV(J) = 0.
        EPCBV(J) = 0.
        ETCBV(J) = 0.
        HCBV(J) = 0.
        DCBV(J) = 0.
   10 CONTINUE
      IF ( .NOT. AUTO) CALL FTITLE

      DO 360 I = 1, NOCW
        ISTAT = HRDCOW(7,I)
        GO TO (40, 20, 20, 80, 100, 120, 140, 160, 180), ISTAT
        WRITE (6,370) HRDCOW(1,I)
        GO TO 360
   20   NLCW = NLCW + 1
        DO 30 J = 1, 4
   30   TCBV(J) = TCBV(J) + HRDCOW(J + 14,I)
        GO TO 60
   40   NBHF = NBHF + 1
        DO 50 J = 1, 4
          THBV(4) = THBV(4) + HRDCOW(J + 14,I)
   50   CONTINUE
   60   DISPO = POPEN
        IF (HRDCOW(9,I) .GT. 0.0) GO TO 70
        LDATE = HRDCOW(10,I) - 0.5
        IF (LDATE .GE. 0) GO TO 200
        DISPO = PBRED
        LDATE = LDATE + DAYR + 1
        GO TO 200
   70   DISPO = PBRED
        LDATE = HRDCOW(9,I) - GEST + 0.5
        GO TO 200
   80   DO 90 J = 1, 4
   90   FCBV(J) = FCBV(J) + HRDCOW(J + 14,I)
        DISPO = PFERT
        NFCUL = NFCUL + 1
        LDATE = DFIN(I) + 0.5
        GO TO 220
  100   DO 110 J = 1, 4
  110   CLCBV(J) = CLCBV(J) + HRDCOW(J + 14,I)
        DISPO = PLACT
        NLCUL = NLCUL + 1
        LDATE = DFIN(I) + 0.5
        GO TO 220
  120   DO 130 J = 1, 4
  130   EPCBV(J) = EPCBV(J) + HRDCOW(J + 14,I)
        DISPO = PEPA
        NPCUL = NPCUL + 1
        LDATE = DFIN(I) + 0.5
        GO TO 220
```

```
140     DO 150 J = 1, 4
150     ETCBV(J) = ETCBV(J) + HRDCOW(J + 14,I)
        DISPO = PETA
        NTCUL = NTCUL + 1
        LDATE = DFIN(I) + 0.5
        GO TO 220
160     DO 170 J = 1, 4
170     HCBV(J) = HCBV(J) + HRDCOW(J + 14,I)
        DISPO = PPROB
        NHCUL = NHCUL + 1
        LDATE = DFIN(I) + 0.5
        GO TO 220
180     DO 190 J = 1, 4
190     DCBV(J) = DCBV(J) + HRDCOW(J + 14,I)
        DISPO = PDEAD
        NDCW = NDCW + 1
        LDATE = DFIN(I) + 0.5
        GO TO 220
200     NCOW = NCOW + 1
        DO 210 J = 1, 10
          HRDCOW(J,NCOW) = HRDCOW(J,I)
          HRDCOW(J + 10,NCOW) = HRDCOW(J + 10,I)
          HRDCOW(J + 20,NCOW) = HRDCOW(J + 20,I)
210     HRDCOW(J + 30,NCOW) = HRDCOW(J + 30,I)
220     DAYO = DAYR
        IF (ISTAT .GT. 3) DAYO = DFIN(I)
        FDCST = OYRLAC(4,I) + DYRLAC(4,I) + (DAYO - DSTOP(I) + DSTART(I)
   1    -OSTOP(I) + OSTART(I)) * DRYFD
        IF (AUTO) GO TO 360
        REC = HRDCOW(6,I)
        IF (DFIN(I) .EQ. DAYR .AND. DSTOP(I) .LT. 305.) GO TO 230
        IF (DSTOP(I) .LT. DPROJ) GO TO 230
        LACO = REC - 1
        LACD = REC
        GO TO 240
230     LACO = REC
        LACD = REC + 1
240     IF (REC .LT. 1.0) GO TO 320
        DO 250 J = 1, 3
          MBCA(J) = (HRDCOW(J + 22,I)/REC/AVG(J)) * 100. + IAJBCA(J) +
   1      100.5
          IBCA(J) = (OLDLAC(J,I)/AVG(J)) * 100. + IAJBCA(J) + 0.5
250     NBCA(J) = (HRDCOW(J + 18,I)/AVG(J)) * 100. + IAJBCA(J) + 0.5
260     IF (OFIN(I) .LE. 0.0) GO TO 290
        LSTR = OLDAT(I) + 0.5
        LYR = IYR
270     IF (LSTR .GE. 0) GO TO 280
        LSTR = LSTR + DAYR - 1.
        LYR = LYR - 1
        GO TO 270
280     LDAY = OSTOP(I) - OSTART(I) + 0.5
290     NSTR = HRDCOW(8,I) + 0.5
        NYR = IYR
300     IF (NSTR .GE. 0) GO TO 310
        NSTR = NSTR + DAYR - 1.
        NYR = NYR - 1
        GO TO 300
310     NDAY = DSTOP(I) - DSTART(I) + 0.5
        GO TO 350
320     DO 330 J = 1, 3
330     MBCA(J) = 0.0
        IF (DFIN(I) .LE. 0.0) GO TO 350
```

```
      DO 340 J = 1, 3
        NBCA(J) = (HRDCOW(J + 18,I)/AVG(J)) * 100. + IAJBCA(J) + 0.5
340   CONTINUE
      GO TO 290
350   CALL WRTLIN(HRDCOW(1,I), AGE(I), DISPO, LDATE, LSTR, LYR, LDAY,
     1     OYRLAC(1,I), FDCST, REC, MBCA, EPA(1,I), SCORE(2,I),
     2     ETA(1,I), SCORE(3,I), IBCA, OAPRD(1,I), NSTR, NYR, NDAY,
     3     DYRLAC(1,I), NBCA, DAPRD(1,I), OSTOP(I), DSTOP(I), OFIN(I),
     4     DFIN(I), OSTART(I), DSTART(I), LACO, LACD, HRDCOW(22,I),
     5     DAYR, DPROJ, LINE, 3, 4)
      IF (LINE .LT. 58) GO TO 360
      CALL TITLE
      LINE = 7
360 CONTINUE
370 FORMAT ('PROBLEMS WITH COW', F5.0, ' STATUS')
    RETURN
    END


    SUBROUTINE SUMOUT(QUOTA, PQMLK, EXCES, PEXMLK, PFAT, CRFAT, PPROT,
   1          CRPROT, PCOWP, PCOWF, PCOWH, PCOWD, PYLG, PCFO, PCFH,
   2          PCFB, PYSR, CFIX, CTRNP, NYSR, AVG, IAJBCA, AUTO,
   3          AUTFUL)
C  *****************************************************************
C  *   This subroutine accumulates years summaries for the herd    *
C  *****************************************************************
    COMMON /CFBV/ BVLCF(4), BVYSP(4), BVSHCF(4), BVSBCF(4), BVDCF(4),
   1       NLCF, NYSP, NSHCF, NSBCF, NDCF, TCFDC /COWBV/ THBV(4),
   2 TCBV(4), FCBV(4), CLCBV(4), EPCBV(4), ETCBV(4), HCBV(4), DCBV(4),
   3       NBHF, NLCW, NFCUL, NLCUL, NPCUL, NTCUL, NHCUL, NDCW
    DIMENSION TRPRD(3), RPRD(3), ARPRD(3), PBCA(4), BCA(4), ABCA(4),
   1         TACPRD(3), ACRPRD(3), ACBCA(4), IAJBCA(4), TAPRD(4),
   2         AVG(4), SINDEX(4), BVLYL(4), BVSYL(4), BVDYL(4)

    REAL *4  BVPS(4), PPS(4), BVUS(4), PUS(4), BVYS(4), PYS(4)

    LOGICAL*1 AUTO, AUTFUL
    RETURN
    ENTRY SUMHRD(TRPRD,PBCA,TAGE,TDP,NRLAC,TACPRD,ACBCA,ACAGE,ACDP,
   1      NACLAC,TAPRD,SINDEX,NYCUL,NCOW, IHRD, IYR)
    HLMLK = TAPRD(1) / 103.2
    IF (HLMLK .LE. 0.) RETURN
    FATT = TAPRD(2) / HLMLK
    PROTT = TAPRD(3) / HLMLK
    PEFAT = (FATT - CRFAT) * PFAT
    PEPROT = (PROTT - CRPROT) * PPROT
    PRICEQ = PQMLK + PEFAT + PEPROT
    PRICEX = PEXMLK + PEFAT + PEPROT
    TEXM = 0.0
    TSM = 0.0
    EXINC = 0.0
    TQM = HLMLK
    IF (TQM .LE. QUOTA) GO TO 20
    TQM = QUOTA
    TEXM = HLMLK - QUOTA
    IF (TEXM .LE. EXCES) GO TO 10
    TSM = TEXM - EXCES
    TEXM = EXCES
 10 EXINC = PRICEX * TEXM
 20 QINC = PRICEQ * TQM
    NPRCUL = NLCUL + NPCUL + NTCUL
```

```
      CSHIP = CTRNP * HLMLK
      SYCUL = NYCUL * PYLG
      SPCUL = NPRCUL * PCOWP
      SFCUL = NFCUL * PCOWF
      SHCUL = NHCUL * PCOWH
      SDCUL = NDCW * PCOWD
      CALL YNGOUT(NYLG, NOCFS, NYLD, BVLYL, BVSYL, BVDYL, TYFDC)
      SOCF = NOCFS * PCFO
      SBCF = (NSBCF + NYSP) * PCFB
      SHCF = NSHCF * PCFH
      SYSR = NYSR * PYSR
      CALL SUMATE(CONC, SVCF, CINT, SEMC, BVPS, PPS, BVUS, PUS, BVYS,
     1      PYS, IHRD, IYR)
      TLCF = NLCF + 1.E-6
      TYSP = NYSP + 1.E-6
      TSHCF = NSHCF + 1.E-6
      TSBCF = NSBCF + 1.E-6
      TDCF = NDCF + 1.E-6
      TLYL = NYLG + 1.E-6
      TSYL = NOCFS + 1.E-6
      TDYL = NYLD + 1.E-6
      TBHF = NBHF + 1.E-6
      TLCW = NLCW + 1.E-6
      TFCUL = NFCUL + 1.E-6
      TLCUL = NLCUL + 1.E-6
      TPCUL = NPCUL + 1.E-6
      TTCUL = NTCUL + 1.E-6
      THCUL = NHCUL + 1.E-6
      TDCW = NDCW + 1.E-6
      DO 23 I = 1, 4
        BVLCF(I) = BVLCF(I) / TLCF
        BVYSP(I) = BVYSP(I) / TYSP
        BVSHCF(I) = BVSHCF(I) / TSHCF
        BVSBCF(I) = BVSBCF(I) / TSBCF
        BVDCF(I) = BVDCF(I) / TDCF
        BVLYL(I) = BVLYL(I) / TLYL
        BVSYL(I) = BVSYL(I) / TSYL
        BVDYL(I) = BVDYL(I) / TDYL
        THBV(I) = THBV(I) / TBHF
        TCBV(I) = TCBV(I) / TLCW
        FCBV(I) = FCBV(I) / TFCUL
        CLCBV(I) = CLCBV(I) / TLCUL
        EPCBV(I) = EPCBV(I) / TPCUL
        ETCBV(I) = ETCBV(I) / TTCUL
        HCBV(I) = HCBV(I) / THCUL
        DCBV(I) = DCBV(I) / TDCW
   23 CONTINUE
      DO 26 I = 1, 3
        BVLCF(I) = BVLCF(I) / AVG(I) * 100. + IAJBCA(I)
        BVYSP(I) = BVYSP(I) / AVG(I) * 100. + IAJBCA(I)
        BVSHCF(I) = BVSHCF(I) / AVG(I) * 100. + IAJBCA(I)
        BVSBCF(I) = BVSBCF(I) / AVG(I) * 100. + IAJBCA(I)
        BVDCF(I) = BVDCF(I) / AVG(I) * 100. + IAJBCA(I)
        BVLYL(I) = BVLYL(I) / AVG(I) * 100. + IAJBCA(I)
        BVSYL(I) = BVSYL(I) / AVG(I) * 100. + IAJBCA(I)
        BVDYL(I) = BVDYL(I) / AVG(I) * 100. + IAJBCA(I)
        THBV(I) = THBV(I) / AVG(I) * 100. + IAJBCA(I)
        TCBV(I) = TCBV(I) / AVG(I) * 100. + IAJBCA(I)
        FCBV(I) = FCBV(I) / AVG(I) * 100. + IAJBCA(I)
        CLCBV(I) = CLCBV(I) / AVG(I) * 100. + IAJBCA(I)
        EPCBV(I) = EPCBV(I) / AVG(I) * 100. + IAJBCA(I)
        ETCBV(I) = ETCBV(I) / AVG(I) * 100. + IAJBCA(I)
```

```
      HCBV(I) = HCBV(I) / AVG(I) * 100. + IAJBCA(I)
      DCBV(I) = DCBV(I) / AVG(I) * 100. + IAJBCA(I)
      BVPS(I) = BVPS(I) / AVG(I) * 100. + IAJBCA(I)
      BVUS(I) = BVUS(I) / AVG(I) * 100. + IAJBCA(I)
      BVYS(I) = BVYS(I) / AVG(I) * 100. + IAJBCA(I)
   26 CONTINUE
      TEXP = CFIX + TAPRD(4) + TYFDC + TCFDC + SEMC + CSHIP
      TINC = QINC + EXINC + SPCUL + SFCUL + SHCUL + SDCUL + SYCUL +
     1SOCF + SBCF + SHCF + SYSR
      PROFIT = TINC - TEXP
      IF (NRLAC .LT. 1 .OR. NACLAC .LT. 1) GO TO 40
      DO 30 J = 1, 3
        BCA(J) = PBCA(J) / NRLAC / AVG(J) * 100. + IAJBCA(J)
        ABCA(J) = ACBCA(J) / NACLAC / AVG(J) * 100. + IAJBCA(J)
        RPRD(J) = TRPRD(J) / NRLAC
   30 ACRPRD(J) = TACPRD(J) / NACLAC
      BCA(4) = PBCA(4) / NRLAC
      ABCA(4) = ACBCA(4) / NACLAC
      PAFAT = 100. * ACRPRD(2) / ACRPRD(1)
      PAPROT = 100. * ACRPRD(3) / ACRPRD(1)
      ACVAGE = ACAGE / NACLAC
      ACADP = ACDP / NACLAC
      PFAT = 100. * RPRD(2) / RPRD(1)
      PPROT = 100. * RPRD(3) / RPRD(1)
      AAGE = TAGE / NRLAC
      ADP = TDP / NRLAC
      GO TO 60
   40 NRLAC = 0
      NACLAC = 0
      AAGE = 0.
      ACVAGE = 0.
      PFAT = 0.
      PPROT = 0.
      PAFAT = 0.
      PAPROT = 0.
      ADP = 0.
      ACADP = 0.
      DO 50 J = 1, 3
        RPRD(J) = 0.
        ACRPRD(J) = 0.
        BCA(J) = 0.
        ABCA(J) = 0.
   50 CONTINUE
      BCA(4) = 0.
      ABCA(4) = 0.
   60 IF (AUTFUL) GO TO 80
      IF (AUTO) GO TO 80
      WRITE (10,170) NRLAC, RPRD, NACLAC, ACRPRD, AAGE, ADP, PFAT,
     1PPROT, ACVAGE, ACADP, PAFAT, PAPROT, BCA, ABCA
      WRITE (10,180) CINT, CONC, NLCW, NBHF, NYLG, NLCF
      WRITE (10,190) HLMLK, FATT, PROTT, CFIX, TQM, PRICEQ, QINC, TEXM,
     1PRICEX, EXINC, TAPRD(4), TSM
      WRITE (10,200) TYFDC, TCFDC, NPRCUL, SPCUL, SEMC, NFCUL, SFCUL,
     1CSHIP, NHCUL, SHCUL, NDCW, SDCUL, TEXP, NYCUL, SYCUL, NOCFS,
     2SOCF, NSBCF, SBCF, NSHCF, SHCF, NYSR, SYSR, TINC, PROFIT
   70 WRITE (6,100)
      WRITE (6,90) NRLAC, RPRD, AAGE, ADP, PFAT, PPROT, BCA
      WRITE (6,110)
      WRITE (6,90) NACLAC, ACRPRD, ACVAGE, ACADP, PAFAT, PAPROT, ABCA
      WRITE (6,120) CINT, CONC
      WRITE (6,130) NLCW, NBHF, NYLG, NLCF
      WRITE (6,140) CFIX, TAPRD(4), TYFDC, TCFDC, SEMC, CSHIP, TEXP
```

```
      WRITE (6,150) HLMLK, FATT, PROTT, TQM, PRICEQ, QINC, TEXM, PRICEX,
     1EXINC, TSM
      WRITE (6,160) NPRCUL, SPCUL, NFCUL, SFCUL, NHCUL, SHCUL, NDCW,
     1SDCUL, NYCUL, SYCUL, NOCFS, SOCF, NSBCF, SBCF, NSHCF, SHCF, NYSR,
     2SYSR, TINC, PROFIT

   80 NHYL = (IHRD - 1) * 100 + IYR
      WRITE (6,99) NACLAC, CINT, CONC, SVCF, NLCW, NBHF, HLMLK,
     1       NPRCUL, NFCUL
   99 FORMAT('  NO. LAC  CF INT   CONC   SV/CF    NLCW    NBHF   ',
     1  'HL MLK   NPRCUL  NFCUL', /, I8, F8.1, 2F8.3, 2I8, F8.0, 2I8)
      WRITE (15'NHYL)  IHRD, IYR, NRLAC, RPRD, BCA, AAGE, ADP, PFAT,
     1       PPROT, NACLAC, ACRPRD, ABCA, ACVAGE, ACADP, PAFAT, PAPROT,
     2       CINT, CONC, HLMLK, FATT, PROTT, TQM, PRICEQ, QINC,
     3       TEXM, PRICEX, EXINC, TSM, CFIX, TAPRD(4), TYFDC, TCFDC,
     4       SEMC, CSHIP, TEXP, SPCUL, SFCUL, SHCUL, SDCUL, SYCUL,
     5       SOCF, SHCF, SBCF, SYSR, TINC, PROFIT, NLCW, TCBV, NBHF,
     6       THBV, NFCUL, FCBV, NLCUL, CLCBV, NPCUL, EPCBV, NTCUL,
     7       ETCBV, NHCUL, HCBV, NDCW, DCBV, NYLG, BVLYL, NOCFS,
     8       BVSYL, NYLD, BVDYL, NLCF, BVLCF, NSHCF, BVSHCF, NSBCF,
     9       BVSBCF, NDCF, BVDCF, NYSP, BVYSP, BVPS, PPS, BVUS,
     *       PUS, BVYS, PYS, SINDEX

   90 FORMAT (16X, 'ROLLING HERD AVERAGES', //, 6X, I4, F10.0, 2F10.1, /
     1        /, 12X, 'AGE   DAYS MILK    % FAT  % PROTEIN', //, 6X,
     2        2F10.1, 2F10.2, //, 20X, 'BCA', //, 6X, 4F10.0)
  100 FORMAT (//, 20X, 'BEFORE CULLING')
  110 FORMAT (//, 20X, 'AFTER CULLING')
  120 FORMAT (//, 8X, 'CALVING INTERVAL', F6.1, /, 8X,
     1        'CONCEPTION RATE ', F6.3)
  130 FORMAT (//, 12X, 'LIVE ANIMALS AT YEAR END', //, 3X, I4, ' COWS',
     1        I6, ' BRED HEIFERS', I6, ' YEARLINGS', I6, ' CALVES')
  140 FORMAT (//, 22X, 'ECONOMIC SUMMARY', //, 25X, 'EXPENDITURES', /,
     1        8X, 'FIXED COSTS', F19.2, /, 8X, 'FEED COSTS', /, 16X,
     2        'COWS', F18.2, /, 11X, 'YEARLINGS', F18.2, /, 14X,
     3        'CALVES', F18.2, /, 8X, 'SEMEN COSTS', F19.2, /, 6X,
     4        'MILK SHIPPING CHARGES', F11.2, //, 5X,
     5        'TOTAL EXPENDITURES', F19.2)
  150 FORMAT (///, 26X, 'INCOME', //, 2X, F8.2, '  HECTOLITRES OF MILK',
     1        F8.2, ' FAT TEST', F8.2, '  PROTEIN TEST', //, 8X, F8.2,
     2        ' HL  QUOTA MILK AT', F6.2, '/ HL', F18.2, /, 8X, F8.2,
     3        ' HL EXCESS MILK AT', F6.2, '/ HL', F18.2, /, 8X, F8.2,
     4        ' HL SURPLUS MILK AT .000/ HL', 14X, '0.00')
  160 FORMAT (/, 6X, 'SOLD ANIMALS', /, 12X, I4, '  COWS (PRODUCTION)',
     1        F27.2, /, 12X, I4, '  COWS (FERTILITY)', F28.2, /, 12X, I4,
     2        '  COWS (HEALTH)', F31.2, /, 12X, I4, '    COWS DEAD',
     3        F33.2, /, 12X, I4, '         YEARLINGS', F30.2, /, 12X, I4,
     4        '      OLD CALVES', F31.2, /, 12X, I4, '    BULL CALVES',
     5        F31.2, /, 12X, I4, '   HEIFER CALVES', F30.2, /, 12X, I4,
     6        '  SELECTED YOUNG SIRES', F24.2, /, 10X, 'TOTAL INCOME',
     7        11X, F30.2, /, /, 10X, 'TOTAL PROFIT', 11X, F30.2)
  170 FORMAT (';', //, 50X, 'ROLLING HERD AVERAGES', /, '+', 49X,
     1        21('_'), //, 30X, 'BEFORE CULLING', 40X, 'AFTER CULLING', /
     2        /, 14X, 'NO. REC.', 3X, 'MILK', 6X, 'FAT', 5X, 'PROTEIN',
     3        24X, 'NO. REC.', 3X, 'MILK', 6X, 'FAT', 5X, 'PROTEIN', /,
     4        16X, I4, F10.0, 2F10.1, 26X, I4, F10.0, 2F10.1, //, 22X,
     5        'AGE   DAYS MILK    % FAT  % PROTEIN', 32X,
     6        'AGE   DAYS MILK    % FAT  % PROTEIN', //, 16X, 2F10.1,
     7        2F10.2, 26X, 2F10.1, 2F10.2, //, 35X, 'BCA', 11X, 'TYPE',
     8        35X, 'BCA', 16X, 'TYPE', //, 16X, 4F10.0, 16X, 4F10.0)
  180 FORMAT (//, 18X, 'CALVING INTERVAL', F7.1, 20X, 'CONCEPTION RATE',
     1        F7.3, //, 42X, 'LIVE ANIMALS AT ', 'YEAR END', //, 20X, I4,
```

```
    2          '  COWS', I10, ' BRED HEIFERS', I10, '  YEARLINGS', I10,
    3          ' CALVES')
  190 FORMAT (//, 50X, 'ECONOMIC SUMMARY', /, '+', 49X, 16('_'), //,
    1          30X, F8.2, '  HECTOLITRES OF MILK', F8.2, '  FAT TEST',
    2          F8.2, ' PROTEIN TEST', //, 15X, 'EXPENDITURES', 55X,
    3          'INCOME', //, 8X, 'FIXED COSTS', F19.2, 25X, F9.2,
    4          ' HL QUOTA MILK AT', F6.2, '/HL', F18.2, /, 8X,
    5          'FEED COSTS', 46X, F8.2, ' HL EXCESS MILK AT', F6.2, 'HL',
    6          F18.2, /, 10X, '-     COWS', F18.2, 25X, F9.2,
    7          'HL SURPLUS MILK AT .000/HL', 14X, '0.00')
  200 FORMAT (10X, '-YEARLINGS', F18.2, 28X, 'SOLD ANIMALS', /, 10X,
    1          '-   CALVES', F18.2, 28X, I4, '  COWS (PRODUCTION)', F27.2,
    2          /, 8X, 'SEMEN COSTS', F19.2, 28X, I4, '  COWS (FERTILITY)',
    3          F28.2, /, 6X, 'MILK SHIPPING COST', F14.2, 28X, I4,
    4          '  COWS (HEALTH)', F31.2, /, 66X, I4, '    COWS DEAD',
    5          F33.2, /, 5X, 'TOTAL EXPENDITURES', F15.2, 28X, I4, 7X,
    6          'YEARLINGS', F30.2, /, 66X, I4, '     OLD CALVES', F31.2, /
    7          , 66X, I4, '    BULL CALVES', F31.2, /, 66X, I4,
    8          '   HEIFER CALVES', F30.2, /, 66X, I4,
    9          '  SELECTED YOUNG SIRES', F24.2, /, 64X, 'TOTAL INCOME',
    *          10X, F30.2, //, 15X, 'TOTAL PROFIT', F28.2)
      RETURN
      END


      SUBROUTINE WRTLIN(PED, AGE, DISPO, LDAT, LSTR, LYR, LDAY, OLAC,
     1          FDCST, REC, MBCA, EPA, SEPA, ETA, SETA, IBCA, OAPRD,
     2          NSTR, NYR, NDAY, DLAC, NBCA, DAPRD, OSTOP, DSTOP, OFIN,
     3          DFIN, OSTART, DSTART, LACO, LACD, TYPE, DAYR, DPROJ,
     4          LINE, M, N)
C   *****************************************************************
C   *    This subroutine writes a cows record to the printer         *
C   *****************************************************************
      DIMENSION IPED(3), PED(M), STAT(2), OLAC(M), MBCA(3), EPA(N),
     1          ETA(N), IBCA(3), OAPRD(M), NBCA(3), DAPRD(M), DLAC(M),
     2          PRJ305(5)

      INTEGER*4 PRJ305 /'PROJ', 'ECTE', 'D TO', ' 305', ' DAY'/

      IPED(1) = PED(1)
      IPED(2) = PED(2)
      IPED(3) = PED(3)
      IAGE = AGE
      ITYPE = TYPE
      IREC = REC
      IOST = OSTOP + 0.5
      IDST = DSTOP + 0.5
      WRITE (10,50) IPED, IAGE, DISPO, LDAT, FDCST, ITYPE, IREC, MBCA,
     1EPA, SEPA, ETA, SETA
      LINE = LINE + 1
      IF (OFIN .LE. 0.0) GO TO 10
      WRITE (10,60) LACO, LSTR, LYR, LDAY, OLAC, IOST, OAPRD, IBCA
      LINE = LINE + 1
   10 IF (DSTOP .LT. DPROJ) GO TO 40
      IF (LACD .GT. IREC) GO TO 20
      IF (DFIN .EQ. DAYR) GO TO 30
      WRITE (10,60) LACD, NSTR, NYR, NDAY, DLAC, IDST, DAPRD, NBCA
      LINE = LINE + 1
      RETURN
   20 WRITE (10,70) LACD, NSTR, NYR, NDAY, DLAC, PRJ305, DAPRD, NBCA
```

```
      LINE = LINE + 1
      RETURN
   30 WRITE (10,80) LACD, NSTR, NYR, NDAY, DLAC, IDST, DAPRD, NBCA
      LINE = LINE + 1
      RETURN
   40 IF (DFIN .LE. 0.0 .OR. DSTOP .LE. 0.) RETURN
      WRITE (10,70) LACD, NSTR, NYR, NDAY, DLAC
      LINE = LINE + 1
      RETURN
   50 FORMAT (1X, I4, 2I5, I3, 1X, A4, I4, 29X, F6.0, 2I3, 1X, 3I4, 1X,
     1        5F5.1, 1X, 5F5.1)
   60 FORMAT (10X, 'LACTATION #', I3, I7, '/', I2, I4, F7.0, 2F6.1,
     1        '    COMPLETE RECORD', I4, ' DAYS MILKED ', F7.0, 2F6.1, 3X,
     2        3I4)
   70 FORMAT (10X, 'LACTATION #', I3, I7, '/', I2, I4, F7.0, 2F6.1, 6X,
     1        5A4, 9X, F7.0, 2F6.1, 3X, 3I4)
   80 FORMAT (10X, 'LACTATION #', I3, I7, '/', I2, I4, F7.0, 2F6.1, 9X,
     1        'RECORD TO', I4, ' DAYS', 8X, F7.0, 2F6.0, 3X, 3I4)
      END


      SUBROUTINE WRTYG(OUT, ICOL, IROW)
C   *******************************************************************
C   *  This subroutine writes a yearling or calf record to the printer *
C   *******************************************************************
      DIMENSION OUT(ICOL,IROW)
      WRITE (10,10) OUT
   10 FORMAT (2X, 4F7.0, 5F6.1, F8.2, 3X, A4, 3X, A4)
      RETURN
      ENTRY WRCUL(OUT,ICOL,IROW)
      WRITE (6,20) OUT
   20 FORMAT (1X, 8F7.0, 2X, A4)
      RETURN
      END


      SUBROUTINE PRINTL(INTARY, NO, IOTU, IFMT)
C   *******************************************************************
C   *  This subroutine writes bull use information to the printer     *
C   *******************************************************************
      DIMENSION INTARY(NO)
      IF (IFMT .GT. 10) GO TO 10
      WRITE (IOTU,20) INTARY
      RETURN
   10 WRITE (IOTU,30) INTARY
      RETURN
      ENTRY PRTUNF (INTARY, NO, IOP)
      WRITE (IOP) INTARY
   20 FORMAT (3X, 20I6)
   30 FORMAT (3X, 20(2X,A4))
      RETURN
      END
```

```
      SUBROUTINE WRTMAT(WRMATR, ICOL, IROW, INP)
C  *******************************************************************
C  *    This subroutine writes two dimensional matricies.            *
C  *******************************************************************
      DIMENSION WRMATR(ICOL,IROW)
      WRITE (INP) WRMATR
      RETURN
      END


      SUBROUTINE REAMAT(REMATR, ICOL, IROW, INP)
C  *******************************************************************
C  *    This subroutine reads two dimensional matricies.             *
C  *******************************************************************
      DIMENSION REMATR(ICOL,IROW)
      READ (INP) REMATR
   10 RETURN
      END


      SUBROUTINE REARRY(NUM, LIST, CONT)
C  *******************************************************************
C  *  This subroutine reads and sorts an array.                      *
C  *******************************************************************
      DIMENSION LIST(NUM)
      LOGICAL*1 CONT, ERCHK /T/, FL /F/
      CONT = FL
      READ (5,10,ERR=30) LIST
   10 FORMAT (I10)
      CALL ISORT(LIST, 1, NUM, 1, NUM, 1, 1, 0)
      WRITE (6,20) LIST
   20 FORMAT (I10)
      GO TO 40
   30 CONT = ERCHK
   40 RETURN
      END


      SUBROUTINE REARY2(NUM, LIST1, LIST2, LIST, CONT)
C  *******************************************************************
C  *  This subroutine reads two arrays, sorts the first and          *
C  *  puts the second in the same order.                             *
C  *******************************************************************
      DIMENSION LIST1(NUM), LIST2(NUM), LIST(2,NUM), LBUL(20)
      LOGICAL*1 CONT, ERCHK /T/, FL /F/
      CONT = FL
      READ (5,10,ERR=60) LIST
   10 FORMAT (2I10)
      CALL ISORT(LIST, 2, NUM, 1, NUM, 1, 1, 0)
      DO 20 I = 1, NUM
        LIST1(I) = LIST(1,I)
        LIST2(I) = LIST(2,I)
   20 CONTINUE
      GO TO 40
      ENTRY WRA(NUM,LIST1,LIST2,LIST)
      DO 30 I = 1, NUM
```

```
      LIST(1,I) = LIST1(I)
   30 LIST(2,I) = LIST2(I)
      GO TO 40
      ENTRY REANSR(NUM,LIST1,LIST2,LIST,CONT)
      CONT = FL
      READ (5,10,ERR=60) LIST
   40 WRITE (6,50) LIST
   50 FORMAT (2I10)
      GO TO 70
   60 CONT = ERCHK
   70 RETURN
      END


      FUNCTION RANDT(ISEED)
C     --------------
C         THIS FUNCTION GENERATES UNIFORM (0,1) RANDOM NUMBERS
C
      DOUBLE PRECISION Z, DN1MOD, DN1
      DATA DN1MOD /2147483647.D0/, DN1 /Z3920000000000000/
      Z = ISEED
      Z = DMOD(16807.D0*Z,DN1MOD)
      RANDT = Z * DN1
      ISEED = Z
      RETURN
      END


      FUNCTION RN(IX)
C
C         THIS FUNCTION GENERATES PAIRS OF NORMAL (0,1) RANDOM DEVIATES,
C         USING A MODIFICATION OF THE BOX-MUELLER METHOD.
C
      DATA I /1/
      IF (I .NE. 1) GO TO 30
      I = 2
   10 U = 2. * RANDT(IX) - 1.
      V = 2. * RANDT(IX) - 1.
      W = U * U + V * V
      IF (W - 1.) 20, 20, 10
   20 W = SQRT(-2.*ALOG(W)/W)
      RN = U * W
      RETURN
   30 I = 1
      RETURN
      END


      SUBROUTINE DCSIG(SIGMA, UL, A, B)
C     ----------------
C   C
C         THIS SUBROUTINE DECOMPOSES THE SYMETRIC MATRIX OF VARIANCES-
C         COVARIANCES INTO ITS FACTOR (SQUARE ROOT).  DCSIG IS CALLED BY
C         MNDG FOR GENERATING MULTIVARIATE NORMAL DEVIATES.
C
```

```
      DIMENSION SIGMA(1), UL(1)
      DATA ZERO, ONE, FOUR, SIXTN, SIXTH /0.0, 1., 4., 16., .0625/
      A = ONE
      B = ZERO
      R1N = ONE / (4*SIXTN)
      IP = 1
      DO 90 I = 1, 4
        IQ = IP
        IR = 1
        DO 80 J = 1, I
          X = SIGMA(IP)
          IF (J .EQ. 1) GO TO 20
          DO 10 L = IQ, IP1
            X = X - UL(L) * UL(IR)
            IR = IR + 1
   10     CONTINUE
   20     IF (I .NE. J) GO TO 60
          A = A * X
          IF (SIGMA(IP) + X*R1N .LE. SIGMA(IP)) GO TO 100
   30     IF (ABS(A) .LE. ONE) GO TO 40
          A = A * SIXTH
          B = B + FOUR
          GO TO 30
   40     IF (ABS(A) .GE. SIXTH) GO TO 50
          A = A * SIXTN
          B = B - FOUR
          GO TO 40
   50     UL(IP) = ONE / SQRT(X)
          GO TO 70
   60     UL(IP) = X * UL(IR)
   70     IP1 = IP
          IP = IP + 1
          IR = IR + 1
   80   CONTINUE
   90 CONTINUE
      GO TO 120
  100 WRITE (6,110)
  110 FORMAT (' TROUBLE')
  120 RETURN
      END


      SUBROUTINE MNDG(ISEED, SIGMA, RVEC, WKVEC)
C     ---------------
C
C         THIS SUBROUTINE GENERATES SETS OF FOUR MULTIVARIATE NORMAL
C         DEVIATES, DISTRIBUTED WITH ZERO MEAN AND COVARIANCES MATRIX
C         SIGMA.  TO PRODUCE GENETICALLY CORRELATED NORMAL DEVIATES,
C         GSIGMA IS ENTERED FOR SIGMA, FOR ENVIRONMENTALLY CORRELATED
C         NORMAL DEVIATES, ESIGMA IS ENTERED.
C
      DIMENSION SIGMA(1), RVEC(4), WKVEC(4)
      CALL DCSIG(SIGMA, SIGMA, A, B)
      L = 0
      DO 10 I = 1, 4
        L = L + I
   10 SIGMA(L) = 1.0 / SIGMA(L)
      GO TO 20
      ENTRY MNDG1(ISEED,SIGMA,RVEC,WKVEC)
   20 DO 30 I = 1, 4
```

```
      DIMENSION SIGMA(1), UL(1)
      DATA ZERO, ONE, FOUR, SIXTN, SIXTH /0.0, 1., 4., 16., .0625/
      A = ONE
      B = ZERO
      R1N = ONE / (4*SIXTN)
      IP = 1
      DO 90 I = 1, 4
        IQ = IP
        IR = 1
        DO 80 J = 1, I
          X = SIGMA(IP)
          IF (J .EQ. 1) GO TO 20
          DO 10 L = IQ, IP1
            X = X - UL(L) * UL(IR)
            IR = IR + 1
   10     CONTINUE
   20     IF (I .NE. J) GO TO 60
          A = A * X
          IF (SIGMA(IP) + X*R1N .LE. SIGMA(IP)) GO TO 100
   30     IF (ABS(A) .LE. ONE) GO TO 40
          A = A * SIXTH
          B = B + FOUR
          GO TO 30
   40     IF (ABS(A) .GE. SIXTH) GO TO 50
          A = A * SIXTN
          B = B - FOUR
          GO TO 40
   50     UL(IP) = ONE / SQRT(X)
          GO TO 70
   60     UL(IP) = X * UL(IR)
   70     IP1 = IP
          IP = IP + 1
          IR = IR + 1
   80   CONTINUE
   90 CONTINUE
      GO TO 120
  100 WRITE (6,110)
  110 FORMAT (' TROUBLE')
  120 RETURN
      END

      SUBROUTINE MNDG(ISEED, SIGMA, RVEC, WKVEC)
C     ---------------
C
C       M.L. MCGILLIARD and D. EDLUND,  1979.     Mimiograph.
C
C         THIS SUBROUTINE GENERATES SETS OF FOUR MULTIVARIATE NORMAL
C         DEVIATES, DISTRIBUTED WITH ZERO MEAN AND COVARIANCES MATRIX
C         SIGMA.  TO PRODUCE GENETICALLY CORRELATED NORMAL DEVIATES,
C         GSIGMA IS ENTERED FOR SIGMA, FOR ENVIRONMENTALLY CORRELATED
C         NORMAL DEVIATES, ESIGMA IS ENTERED.
C
      DIMENSION SIGMA(1), RVEC(4), WKVEC(4)
      CALL DCSIG(SIGMA, SIGMA, A, B)
      L = 0
      DO 10 I = 1, 4
        L = L + I
   10 SIGMA(L) = 1.0 / SIGMA(L)
      GO TO 20
      ENTRY MNDG1(ISEED,SIGMA,RVEC,WKVEC)
   20 DO 30 I = 1, 4
```

```
   30 WKVEC(I) = RN(ISEED)
      L = 1
      DO 50 II = 1, 4
        RVEC(II) = 0.0
        DO 40 I = 1, II
          RVEC(II) = RVEC(II) + DBLE(WKVEC(I)) * DBLE(SIGMA(L))
   40   L = L + 1
   50 CONTINUE
      RETURN
      END


      SUBROUTINE TITLE
C   *****************************************************************
C   *  This subroutine titles the printer output                    *
C   *****************************************************************
     WRITE (10,10)
      ENTRY FTITLE
      WRITE (10,20)
   10 FORMAT (';')
   20 FORMAT (/, 32X, 'CURRENT PRODUCTION', 22X, 'MEAN BCA ', 10X,
     1        'EPA', 22X, 'ETA', /, 28X, 28('-'), ' FEED TYP NO   ', 10(
     2        '-'), 5X, 17('-'), 8X, 17('-'), /,
     3        '  COW SIRE DAM AGE STAT DAY FRESH DAYS   MILK  FAT ',
     4        ' PROT COST  SC RC  MLK FAT PRO  MILK FAT  PROT TYPE ',
     5        '$IND  MILK FAT  PROT TYPE $IND', /)
      RETURN
      ENTRY TITLYG
      WRITE (10,30)
   30 FORMAT (//, 35X, 'YEARLINGS', /, 45X, 'ETA', 14X, 'FEED', /, 3X,
     1        'HEIFER  SIRE    DAM   DATE    MILK  FAT',
     2        '  PROT  TYPE $INDEX  COSTS  STATUS')
      RETURN
      ENTRY TITLCF
      WRITE (10,40)
   40 FORMAT (//, 37X, 'CALVES', /, 45X, 'ETA', 14X, 'FEED', /, 5X,
     1        'CALF  SIRE    DAM   DATE    MILK  FAT',
     2        '  PROT  TYPE $INDEX  COSTS   SEX  STATUS')
      RETURN
      END


      SUBROUTINE FILES(IHRD)
C
C *****************************************************************
C   *                                                             *
C   *       This subroutine finds the herd files                  *
C   *                                                             *
C *****************************************************************
      CALL FTNCMD('ASSIGN 4=KINN:AI.U;')
      CALL FTNCMD ('ASSIGN 7=KINN:CHECK.RUN;')
      CALL FTNCMD('ASSIGN 13=KINN:DREC(LAST+1);')
      CALL FTNCMD('ASSIGN 11=KINN:YSIRE(LAST+1);')
      CALL FTNCMD('ASSIGN 14=KINN:BUSE(LAST+1);')
      CALL FTNCMD('ASSIGN 15=KINN:SUMS;')
      IF (IHRD .GT. 100) GO TO 1010
      GO TO (10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130,
     1140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260,
     2270, 280, 290, 300, 310, 320, 330, 340, 350, 360, 370, 380, 390,
```

```
     3400, 410, 420, 430, 440, 450, 460, 470, 480, 490, 500, 510, 520,
     4530, 540, 550, 560, 570, 580, 590, 600, 610, 620, 630, 640, 650,
     5660, 670, 680, 690, 700, 710, 720, 730, 740, 750, 760, 770, 780,
     6790, 800, 810, 820, 830, 840, 850, 860, 870, 880, 890, 900, 910,
     7920, 930, 940, 950, 960, 970, 980, 990, 1000), IHRD
      GO TO 1060
   10 CALL FTNCMD('ASSIGN 2=KINN:H1IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H1OUT;')
      CALL FTNCMD('ASSIGN 10=H1-2(LAST+1);')
      RETURN
   20 CALL FTNCMD('ASSIGN 2=KINN:H2IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H2OUT;')
      CALL FTNCMD('ASSIGN 10=H1-2(LAST+1);')
      RETURN
   30 CALL FTNCMD('ASSIGN 2=KINN:H3IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H3OUT;')
      CALL FTNCMD('ASSIGN 10=H3-4(LAST+1);')
      RETURN
   40 CALL FTNCMD('ASSIGN 2=KINN:H4IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H4OUT;')
      CALL FTNCMD('ASSIGN 10=H3-4(LAST+1);')
      RETURN
   50 CALL FTNCMD('ASSIGN 2=KINN:H5IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H5OUT;')
      CALL FTNCMD('ASSIGN 10=H5-6(LAST+1);')
      RETURN
   60 CALL FTNCMD('ASSIGN 2=KINN:H6IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H6OUT;')
      CALL FTNCMD('ASSIGN 10=H5-6(LAST+1);')
      RETURN
   70 CALL FTNCMD('ASSIGN 2=KINN:H7IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H7OUT;')
      CALL FTNCMD('ASSIGN 10=H7-8(LAST+1);')
      RETURN
   80 CALL FTNCMD('ASSIGN 2=KINN:H8IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H8OUT;')
      CALL FTNCMD('ASSIGN 10=H7-8(LAST+1);')
      RETURN
   90 CALL FTNCMD('ASSIGN 2=KINN:H9IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H9OUT;')
      CALL FTNCMD('ASSIGN 10=H9-10(LAST+1);')
      RETURN
  100 CALL FTNCMD('ASSIGN 2=KINN:H10IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H10OUT;')
      CALL FTNCMD('ASSIGN 10=H9-10(LAST+1);')
      RETURN
  110 CALL FTNCMD('ASSIGN 2=KINN:H11IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H11OUT;')
      CALL FTNCMD('ASSIGN 10=H11-12(LAST+1);')
      RETURN
  120 CALL FTNCMD('ASSIGN 2=KINN:H12IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H12OUT;')
      CALL FTNCMD('ASSIGN 10=H11-12(LAST+1);')
      RETURN
  130 CALL FTNCMD('ASSIGN 2=KINN:H13IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H13OUT;')
      CALL FTNCMD('ASSIGN 10=H13-14(LAST+1);')
      RETURN
  140 CALL FTNCMD('ASSIGN 2=KINN:H14IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H14OUT;')
      CALL FTNCMD('ASSIGN 10=H13-14(LAST+1);')
      RETURN
```

```
150 CALL FTNCMD('ASSIGN 2=KINN:H15IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H15OUT;')
    CALL FTNCMD('ASSIGN 10=H15-16(LAST+1);')
    RETURN
160 CALL FTNCMD('ASSIGN 2=KINN:H16IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H16OUT;')
    CALL FTNCMD('ASSIGN 10=H15-16(LAST+1);')
    RETURN
170 CALL FTNCMD('ASSIGN 2=KINN:H17IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H17OUT;')
    CALL FTNCMD('ASSIGN 10=H17-18(LAST+1);')
    RETURN
180 CALL FTNCMD('ASSIGN 2=KINN:H18IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H18OUT;')
    CALL FTNCMD('ASSIGN 10=H17-18(LAST+1);')
    RETURN
190 CALL FTNCMD('ASSIGN 2=KINN:H19IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H19OUT;')
    CALL FTNCMD('ASSIGN 10=H19-20(LAST+1);')
    RETURN
200 CALL FTNCMD('ASSIGN 2=KINN:H20IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H20OUT;')
    CALL FTNCMD('ASSIGN 10=H19-20(LAST+1);')
    RETURN
210 CALL FTNCMD('ASSIGN 2=KINN:H21IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H21OUT;')
    CALL FTNCMD('ASSIGN 10=H21-22(LAST+1);')
    RETURN
220 CALL FTNCMD('ASSIGN 2=KINN:H22IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H22OUT;')
    CALL FTNCMD('ASSIGN 10=H21-22(LAST+1);')
    RETURN
230 CALL FTNCMD('ASSIGN 2=KINN:H23IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H23OUT;')
    CALL FTNCMD('ASSIGN 10=H23-24(LAST+1);')
    RETURN
240 CALL FTNCMD('ASSIGN 2=KINN:H24IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H24OUT;')
    CALL FTNCMD('ASSIGN 10=H23-24(LAST+1);')
    RETURN
250 CALL FTNCMD('ASSIGN 2=KINN:H25IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H25OUT;')
    CALL FTNCMD('ASSIGN 10=H25-26(LAST+1);')
    RETURN
260 CALL FTNCMD('ASSIGN 2=KINN:H26IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H26OUT;')
    CALL FTNCMD('ASSIGN 10=H25-26(LAST+1);')
    RETURN
270 CALL FTNCMD('ASSIGN 2=KINN:H27IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H27OUT;')
    CALL FTNCMD('ASSIGN 10=H27-28(LAST+1);')
    RETURN
280 CALL FTNCMD('ASSIGN 2=KINN:H28IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H28OUT;')
    CALL FTNCMD('ASSIGN 10=H27-28(LAST+1);')
    RETURN
290 CALL FTNCMD('ASSIGN 2=KINN:H29IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H29OUT;')
    CALL FTNCMD('ASSIGN 10=H29-30(LAST+1);')
    RETURN
300 CALL FTNCMD('ASSIGN 2=KINN:H30IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H30OUT;')
```

```
    CALL FTNCMD('ASSIGN 10=H29-30(LAST+1);')
    RETURN
310 CALL FTNCMD('ASSIGN 2=KINN:H31IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H31OUT;')
    CALL FTNCMD('ASSIGN 10=H31-32(LAST+1);')
    RETURN
320 CALL FTNCMD('ASSIGN 2=KINN:H32IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H32OUT;')
    CALL FTNCMD('ASSIGN 10=H31-32(LAST+1);')
    RETURN
330 CALL FTNCMD('ASSIGN 2=KINN:H33IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H33OUT;')
    CALL FTNCMD('ASSIGN 10=H33-34(LAST+1);')
    RETURN
340 CALL FTNCMD('ASSIGN 2=KINN:H34IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H34OUT;')
    CALL FTNCMD('ASSIGN 10=H33-34(LAST+1);')
    RETURN
350 CALL FTNCMD('ASSIGN 2=KINN:H35IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H35OUT;')
    CALL FTNCMD('ASSIGN 10=H35-36(LAST+1);')
    RETURN
360 CALL FTNCMD('ASSIGN 2=KINN:H36IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H36OUT;')
    CALL FTNCMD('ASSIGN 10=H35-36(LAST+1);')
    RETURN
370 CALL FTNCMD('ASSIGN 2=KINN:H37IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H37OUT;')
    CALL FTNCMD('ASSIGN 10=H37-38(LAST+1);')
    RETURN
380 CALL FTNCMD('ASSIGN 2=KINN:H38IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H38OUT;')
    CALL FTNCMD('ASSIGN 10=H37-38(LAST+1);')
    RETURN
390 CALL FTNCMD('ASSIGN 2=KINN:H39IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H39OUT;')
    CALL FTNCMD('ASSIGN 10=H39-40(LAST+1);')
    RETURN
400 CALL FTNCMD('ASSIGN 2=KINN:H40IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H40OUT;')
    CALL FTNCMD('ASSIGN 10=H39-40(LAST+1);')
    RETURN
410 CALL FTNCMD('ASSIGN 2=KINN:H41IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H41OUT;')
    CALL FTNCMD('ASSIGN 10=H41-42(LAST+1);')
    RETURN
420 CALL FTNCMD('ASSIGN 2=KINN:H42IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H42OUT;')
    CALL FTNCMD('ASSIGN 10=H41-42(LAST+1);')
    RETURN
430 CALL FTNCMD('ASSIGN 2=KINN:H43IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H43OUT;')
    CALL FTNCMD('ASSIGN 10=H43-44(LAST+1);')
    RETURN
440 CALL FTNCMD('ASSIGN 2=KINN:H44IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H44OUT;')
    CALL FTNCMD('ASSIGN 10=H43-44(LAST+1);')
    RETURN
450 CALL FTNCMD('ASSIGN 2=KINN:H45IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H45OUT;')
    CALL FTNCMD('ASSIGN 10=H45-46(LAST+1);')
    RETURN
```

```
460 CALL FTNCMD('ASSIGN 2=KINN:H46IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H46OUT;')
    CALL FTNCMD('ASSIGN 10=H45-46(LAST+1);')
    RETURN
470 CALL FTNCMD('ASSIGN 2=KINN:H47IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H47OUT;')
    CALL FTNCMD('ASSIGN 10=H47-48(LAST+1);')
    RETURN
480 CALL FTNCMD('ASSIGN 2=KINN:H48IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H48OUT;')
    CALL FTNCMD('ASSIGN 10=H47-48(LAST+1);')
    RETURN
490 CALL FTNCMD('ASSIGN 2=KINN:H49IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H49OUT;')
    CALL FTNCMD('ASSIGN 10=H49-50(LAST+1);')
    RETURN
500 CALL FTNCMD('ASSIGN 2=KINN:H50IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H50OUT;')
    CALL FTNCMD('ASSIGN 10=H49-50(LAST+1);')
    RETURN
510 CALL FTNCMD('ASSIGN 2=KINN:H51IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H51OUT;')
    CALL FTNCMD('ASSIGN 10=H51-52(LAST+1);')
    RETURN
520 CALL FTNCMD('ASSIGN 2=KINN:H52IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H52OUT;')
    CALL FTNCMD('ASSIGN 10=H51-52(LAST+1);')
    RETURN
530 CALL FTNCMD('ASSIGN 2=KINN:H53IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H53OUT;')
    CALL FTNCMD('ASSIGN 10=H53-54(LAST+1);')
    RETURN
540 CALL FTNCMD('ASSIGN 2=KINN:H54IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H54OUT;')
    CALL FTNCMD('ASSIGN 10=H53-54(LAST+1);')
    RETURN
550 CALL FTNCMD('ASSIGN 2=KINN:H55IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H55OUT;')
    CALL FTNCMD('ASSIGN 10=H55-56(LAST+1);')
    RETURN
560 CALL FTNCMD('ASSIGN 2=KINN:H56IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H56OUT;')
    CALL FTNCMD('ASSIGN 10=H55-56(LAST+1);')
    RETURN
570 CALL FTNCMD('ASSIGN 2=KINN:H57IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H57OUT;')
    CALL FTNCMD('ASSIGN 10=H57-58(LAST+1);')
    RETURN
580 CALL FTNCMD('ASSIGN 2=KINN:H58IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H58OUT;')
    CALL FTNCMD('ASSIGN 10=H57-58(LAST+1);')
    RETURN
590 CALL FTNCMD('ASSIGN 2=KINN:H59IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H59OUT;')
    CALL FTNCMD('ASSIGN 10=H59-60(LAST+1);')
    RETURN
600 CALL FTNCMD('ASSIGN 2=KINN:H60IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H60OUT;')
    CALL FTNCMD('ASSIGN 10=H59-60(LAST+1);')
    RETURN
610 CALL FTNCMD('ASSIGN 2=KINN:H61IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H61OUT;')
```

```
      CALL FTNCMD('ASSIGN 10=H61-62(LAST+1);')
      RETURN
  620 CALL FTNCMD('ASSIGN 2=KINN:H62IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H62OUT;')
      CALL FTNCMD('ASSIGN 10=H61-62(LAST+1);')
      RETURN
  630 CALL FTNCMD('ASSIGN 2=KINN:H63IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H63OUT;')
      CALL FTNCMD('ASSIGN 10=H63-64(LAST+1);')
      RETURN
  640 CALL FTNCMD('ASSIGN 2=KINN:H64IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H64OUT;')
      CALL FTNCMD('ASSIGN 10=H63-64(LAST+1);')
      RETURN
  650 CALL FTNCMD('ASSIGN 2=KINN:H65IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H65OUT;')
      CALL FTNCMD('ASSIGN 10=H65-66(LAST+1);')
      RETURN
  660 CALL FTNCMD('ASSIGN 2=KINN:H66IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H66OUT;')
      CALL FTNCMD('ASSIGN 10=H65-66(LAST+1);')
      RETURN
  670 CALL FTNCMD('ASSIGN 2=KINN:H67IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H67OUT;')
      CALL FTNCMD('ASSIGN 10=H67-68(LAST+1);')
      RETURN
  680 CALL FTNCMD('ASSIGN 2=KINN:H68IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H68OUT;')
      CALL FTNCMD('ASSIGN 10=H67-68(LAST+1);')
      RETURN
  690 CALL FTNCMD('ASSIGN 2=KINN:H69IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H69OUT;')
      CALL FTNCMD('ASSIGN 10=H69-70(LAST+1);')
      RETURN
  700 CALL FTNCMD('ASSIGN 2=KINN:H70IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H70OUT;')
      CALL FTNCMD('ASSIGN 10=H69-70(LAST+1);')
      RETURN
  710 CALL FTNCMD('ASSIGN 2=KINN:H71IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H71OUT;')
      CALL FTNCMD('ASSIGN 10=H71-72(LAST+1);')
      RETURN
  720 CALL FTNCMD('ASSIGN 2=KINN:H72IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H72OUT;')
      CALL FTNCMD('ASSIGN 10=H71-72(LAST+1);')
      RETURN
  730 CALL FTNCMD('ASSIGN 2=KINN:H73IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H73OUT;')
      CALL FTNCMD('ASSIGN 10=H73-74(LAST+1);')
      RETURN
  740 CALL FTNCMD('ASSIGN 2=KINN:H74IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H74OUT;')
      CALL FTNCMD('ASSIGN 10=H73-74(LAST+1);')
      RETURN
  750 CALL FTNCMD('ASSIGN 2=KINN:H75IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H75OUT;')
      CALL FTNCMD('ASSIGN 10=H75-76(LAST+1);')
      RETURN
  760 CALL FTNCMD('ASSIGN 2=KINN:H76IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H76OUT;')
      CALL FTNCMD('ASSIGN 10=H75-76(LAST+1);')
      RETURN
```

```
770 CALL FTNCMD('ASSIGN 2=KINN:H77IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H77OUT;')
    CALL FTNCMD('ASSIGN 10=H77-78(LAST+1);')
    RETURN
780 CALL FTNCMD('ASSIGN 2=KINN:H78IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H78OUT;')
    CALL FTNCMD('ASSIGN 10=H77-78(LAST+1);')
    RETURN
790 CALL FTNCMD('ASSIGN 2=KINN:H79IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H79OUT;')
    CALL FTNCMD('ASSIGN 10=H79-80(LAST+1);')
    RETURN
800 CALL FTNCMD('ASSIGN 2=KINN:H80IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H80OUT;')
    CALL FTNCMD('ASSIGN 10=H79-80(LAST+1);')
    RETURN
810 CALL FTNCMD('ASSIGN 2=KINN:H81IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H81OUT;')
    CALL FTNCMD('ASSIGN 10=H81-82(LAST+1);')
    RETURN
820 CALL FTNCMD('ASSIGN 2=KINN:H82IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H82OUT;')
    CALL FTNCMD('ASSIGN 10=H81-82(LAST+1);')
    RETURN
830 CALL FTNCMD('ASSIGN 2=KINN:H83IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H83OUT;')
    CALL FTNCMD('ASSIGN 10=H83-84(LAST+1);')
    RETURN
840 CALL FTNCMD('ASSIGN 2=KINN:H84IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H84OUT;')
    CALL FTNCMD('ASSIGN 10=H83-84(LAST+1);')
    RETURN
850 CALL FTNCMD('ASSIGN 2=KINN:H85IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H85OUT;')
    CALL FTNCMD('ASSIGN 10=H85-86(LAST+1);')
    RETURN
860 CALL FTNCMD('ASSIGN 2=KINN:H86IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H86OUT;')
    CALL FTNCMD('ASSIGN 10=H85-86(LAST+1);')
    RETURN
870 CALL FTNCMD('ASSIGN 2=KINN:H87IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H87OUT;')
    CALL FTNCMD('ASSIGN 10=H87-88(LAST+1);')
    RETURN
880 CALL FTNCMD('ASSIGN 2=KINN:H88IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H88OUT;')
    CALL FTNCMD('ASSIGN 10=H87-88(LAST+1);')
    RETURN
890 CALL FTNCMD('ASSIGN 2=KINN:H89IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H89OUT;')
    CALL FTNCMD('ASSIGN 10=H89-90(LAST+1);')
    RETURN
900 CALL FTNCMD('ASSIGN 2=KINN:H90IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H90OUT;')
    CALL FTNCMD('ASSIGN 10=H89-90(LAST+1);')
    RETURN
910 CALL FTNCMD('ASSIGN 2=KINN:H91IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H91OUT;')
    CALL FTNCMD('ASSIGN 10=H91-92(LAST+1);')
    RETURN
920 CALL FTNCMD('ASSIGN 2=KINN:H92IN;')
    CALL FTNCMD('ASSIGN 12=KINN:H92OUT;')
```

```
      CALL FTNCMD('ASSIGN 10=H91-92(LAST+1);')
      RETURN
  930 CALL FTNCMD('ASSIGN 2=KINN:H93IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H93OUT;')
      CALL FTNCMD('ASSIGN 10=H93-94(LAST+1);')
      RETURN
  940 CALL FTNCMD('ASSIGN 2=KINN:H94IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H94OUT;')
      CALL FTNCMD('ASSIGN 10=H93-94(LAST+1);')
      RETURN
  950 CALL FTNCMD('ASSIGN 2=KINN:H95IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H95OUT;')
      CALL FTNCMD('ASSIGN 10=H95-96(LAST+1);')
      RETURN
  960 CALL FTNCMD('ASSIGN 2=KINN:H96IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H96OUT;')
      CALL FTNCMD('ASSIGN 10=H95-96(LAST+1);')
      RETURN
  970 CALL FTNCMD('ASSIGN 2=KINN:H97IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H97OUT;')
      CALL FTNCMD('ASSIGN 10=H97-98(LAST+1);')
      RETURN
  980 CALL FTNCMD('ASSIGN 2=KINN:H98IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H98OUT;')
      CALL FTNCMD('ASSIGN 10=H97-98(LAST+1);')
      RETURN
  990 CALL FTNCMD('ASSIGN 2=KINN:H99IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H99OUT;')
      CALL FTNCMD('ASSIGN 10=H99-100(LAST+1);')
      RETURN
 1000 CALL FTNCMD('ASSIGN 2=KINN:H100IN;')
      CALL FTNCMD('ASSIGN 12=KINN:H100OUT;')
      CALL FTNCMD('ASSIGN 10=H99-100(LAST+1);')
      RETURN
 1010 I = IHRD / 100
      GO TO (1020, 1030, 1040, 1050), I
      GO TO 1060
 1020 CALL FTNCMD('ASSIGN 2=KINN:G100IN;')
      CALL FTNCMD('ASSIGN 12=KINN:G100OUT;')
      CALL FTNCMD('ASSIGN 10=-CHK(LAST+1);')
      RETURN
 1030 CALL FTNCMD('ASSIGN 2=KINN:G200IN;')
      CALL FTNCMD('ASSIGN 12=KINN:G200OUT;')
      CALL FTNCMD('ASSIGN 10=-CHK(LAST+1);')
      RETURN
 1040 CALL FTNCMD('ASSIGN 2=KINN:G300IN;')
      CALL FTNCMD('ASSIGN 12=KINN:G300OUT;')
      CALL FTNCMD('ASSIGN 10=-CHK(LAST+1);')
      RETURN
 1050 CALL FTNCMD('ASSIGN 2=KINN:G400IN;')
      CALL FTNCMD('ASSIGN 12=KINN:G400OUT;')
      CALL FTNCMD('ASSIGN 10=-CHK(LAST+1);')
      RETURN
 1060 WRITE (6,1070) IHRD
 1070 FORMAT ('   HERD', I4, '  NOT FOUND')
      STOP
      END
```

```
C*********************************************************************
C*                                                                  *
C*                    A. I. UPDATE PROGRAM                          *
C*                    --------------------                          *
C*                                                                  *
C*    Reads files -DREC and -BUSE (DREC and BUSE sorted by bull     *
C*    number) and updates ETA's, daughter averages, use, conceptions *
C*    and semen prices.                                             *
C*    Reads file YSIRE and adds the top young bulls to the A.I. file *
C*                                                                  *
C*      Data: H4H = (4-heritability)/heritability                   *
C*             G4 = 1/4 * genetic variance                          *
C*            VECG2 = environmental variance between cow + 1/2 genetic *
C*                                                       variance.  *
C*            VEH = environmental variance between herds.           *
C*            SPM & SPSD = mean and standard deviation of relative  *
C*                                        semen production.        *
C*            BFM & BFSD = mean and standard deviation of fertility. *
C*            TCOW = number of cows in the population.              *
C*            BASP = base semen price                               *
C*            DYR = Age at which a bull is removed from A.I.        *
C*            (Note: If more than 33 bulls are added each year      *
C*                   decrease DYR or increase the dimension of AI)  *
C*                                                                  *
C*********************************************************************

      REAL *4  AI(30,500), PETA(4), YBUL(18),
     *         PYS(15,500), SINDEX(4),
     1         H4H(4)/14.38, 10.76, 13.81, 12.33/, G4(4)/42642., 82.81,
     2         40.96, 0.6889/, VECG2(4)/499833., 670.34, 460.69, 6.5010/,
     3         VEH(4)/58081., 125.44, 59.29, 1.2769/, AVG(3)/7200., 280.,
     4         230./, BFM/.7/, BFSD/.02/, SPM/.5/, SPSD/.3/, TCOW/750./,
     5         BASP/15./, DYR/15./, UYR/1./
      INTEGER *4 IDREC(8), IUSE(5), ITB(4)
      LOGICAL *1 FALSE/F/, TRUE/T/, YYB/T/, BSWT/T/, USWT/T/, YNG/F/
C***                                                              ***
C***     Assign input-output units to files and initialize FRAND  ***
C***                                                              ***
      CALL FTNCMD('ASSIGN 1=DREC;')
      CALL FTNCMD('ASSIGN 2=BUSE;')
      CALL FTNCMD('ASSIGN 3=YSIRE;')
      CALL FTNCMD('ASSIGN 4=AI.T;')
      CALL FTNCMD('ASSIGN 13=AIOLD(LAST+1);')
      CALL FTNCMD('ASSIGN 14=AI.U;')
      IR = IRAND(0)
      F = IRAND(1000)
      F = RANDN(F)
C***                                                              ***
C***     Zero Totals                                              ***
C***                                                              ***
      TVIAL = 0.
      TCON = 0.
      ND = 0
      NHYS = 0
C***                                                              ***
C***     Read the current A. I. file and the first record from    ***
C***     files DREC , BUSE and YSIRE.                            ***
C***                                                              ***
      READ (4) NDAI, NAI, NUAI, NEAI, NPAI, NTAI
      CALL REAMAT(AI, 30, NTAI, 4)
      IF (NAI .LT. 20) TCOW = TCOW * 3.
      NND = 0
```

```
      NSR = 9000 + NDAI
C***    Check for bulls to be removed                                ***
      DO 95 I = 1, NAI
        IF (YR-AI(8,I) .LT. DYR) GO TO 77
        CALL WRTMAT (AI(1,I), 30, 1, 13)
        NND = NND + 1
        NSR = NSR + 1
   95 CONTINUE
      IF (NEAI .LE. NAI) GO TO 251
      I = NAI + 1
   77 IN = I
      JFAI = NSR + 1
  20  READ (1, END=30) IDREC
      IBSR = IDREC(3)
      IF (IBSR .LT. JFAI) GO TO 20
      GO TO 40
  30  IBSR = 0
      BSWT = FALSE
      NCDAI = 0
  40  READ (2, END=50) IUSE
      IUSR = IUSE(3)
      IF (IUSR .LT. JFAI) GO TO 40
      GO TO 60
  50  USWT = FALSE
  60  READ (3, END=70) YBUL
      GO TO 75
  70  YYB = FALSE
   75 WRITE (6, 450)
      READ (5,430,ERR=75) IYR
      YR = IYR
C***                                                                 ***
C***   Major Loop - Updates Sire Proofs                              ***
C***                                                                 ***
      NRAI = 0
      DO 250 I = IN, NEAI
        NSR = NSR + 1
C***                                                                 ***
C***        Add first lactation record from last herd and read another***
C***                                                                 ***
        IF (.NOT. BSWT) GO TO 160
        IBSR = IDREC(3)
        IF (NSR .NE. IBSR) GO TO 78
        GO TO 80
   78   IF (NSR .GT. IBSR) GO TO 397
        GO TO 140

  80    NHYS = NHYS + 1
        ND = ND + IDREC(4)
        DO 90 J = 1, 4
          ITB(J) = ITB(J) + IDREC(J+4)
  90    CONTINUE
        READ (1, END=100) IDREC
        IF (NSR .EQ. IDREC(3)) GO TO 80
        GO TO 110
 100    BSWT = FALSE
        NCDAI = NRAI + 1
C***                                                                 ***
C***        Adjust the bull's record for daughters, herds, ETA's     ***
C***        and daughter's average.                                  ***
C***                                                                 ***
 110    TND = ND
        TNHY = NHYS
```

```
      TD = TND + AI(13,I)
      DO 130 J = 1, 4
        BD = TND / (TND + H4H(J))
        SD = SQRT((1-BD) * G4(J) + VECG2(J) / TND + VEH(J) /TNHY)
        SETA = (AI(J+14,I) * .5 + FRANDN(0.) * SD) * BD
      IF (J .EQ. 4) GO TO 115
        SETA = SETA / AVG(J) * 100.
 115    AI(J+22,I) = (ITB(J) + AI(J+22,I) * AI(13,I)) / TD
        AI(J+26,I) = (SETA * TND + AI(J+26,I) * AI(13,I)) / TD
 130  CONTINUE
      AI(13,I) = TD
      AI(14,I) = AI(14,I) + TNHY
C***                                                                   ***
C***   Zero Totals                                                     ***
C***                                                                   ***
 140  DO 155 J = 1, 4
 155  ITB(J) = 0.
      ND = 0
      NHYS = 0
 160  IF (.NOT. USWT) GO TO 200
C***                                                                   ***
C***  Add use by last herd and read another                            ***
C***                                                                   ***
      IUSR = IUSE(3)
      IF (NSR .NE. IUSR) GO TO 165
      GO TO 170
 165  IF (NSR .GT. IUSR) GO TO 398
      GO TO 200

 170  TVIAL = TVIAL + IUSE(4)
      TCON = TCON + IUSE(5)
      READ (2, END=180) IUSE
      IF (NSR .EQ. IUSE(3)) GO TO 170
      GO TO 190
 180  USWT = FALSE
C***                                                                   ***
C***     Adjust bulls record for vials used and conceptions            ***
C***     and calculate a new semen price                               ***
C***                                                                   ***
C***                                                                   ***
 190  AI(11,I) = AI(11,I) + TVIAL
      AI(12,I) = AI(12,I) + TCON
      IF (AI(10,I) .LE. 0.) AI(10,I) = FRANDN(0.) * SPSD + SPM
      PRICE = BASP + TVIAL**2. / TCOW
      IF (AI(7,I) .GT. PRICE) PRICE = .5 * (PRICE + AI(7,I))
      AI(7,I) = PRICE
      TVIAL = 0.
      TCON = 0.
      GO TO 210
 200  AI(7,I) = .5 * (BASP + AI(7,I))
C***                                                                  ***
C***   move up location of eligible bulls to replace dead             ***
C***                                                                  ***
  210 NRAI = NRAI + 1
      DO 220 J = 1, 30
        AI(J,NRAI) = AI(J,I)
 220  CONTINUE
 250  CONTINUE
     IF (BSWT) NCDAI = NEAI
C***                                                                   ***
C***  Check if young bulls old enough to use and move up rest bulls***
C***                                                                   ***
```

```
 251 NFY = NEAI + 1
     NNAI = 0
       IF (NFY .GT. NTAI) GO TO 254
     DO 256 I = NFY, NTAI
       NRAI = NRAI + 1
       DO 253 J = 1, 30
         AI(J,NRAI) = AI(J,I)
253    CONTINUE
       IF (YNG) GO TO 256
       AGE = YR - AI(8,NRAI)
       IF (AGE .LT. UYR) GO TO 252
       NNAI = NNAI + 1
       GO TO 256
 252   YNG = TRUE
256  CONTINUE
C***                                                              ***
C***   Young bull selection                                       ***
C***                                                              ***
 254 NYSS = 0
     NFYAI = NRAI + 1
     IF (.NOT. YYB) GO TO 350
     WRITE (6,400)
     READ (5,430) NYSS, TIND, TTYP, TDTYP
     IF (NYSS .LE. 0) GO TO 350
 255 WRITE (6,410)
     READ(5,440) SINDEX
     ST = ABS(SINDEX(1) + SINDEX(2) + SINDEX(3) + SINDEX(4))
     IF (ST .LE. 0.) GO TO 255
     DO 258 J = 1, 4
       SINDEX(J) = SINDEX(J) / ST
 258 CONTINUE
     NPYS = 0
     GO TO 270
260  READ (3, END=290) YBUL
270  IF (YBUL(14) .LT. TTYP .OR. YBUL(18) .LT. TDTYP) GO TO 260
     BIND = YBUL(11) * SINDEX(1) + YBUL(12) * SINDEX(2) + YBUL(13)
    1        * SINDEX(3) + YBUL(14) * SINDEX(4)
     IF (BIND .LT. TIND) GO TO 260
     NPYS = NPYS + 1
     IF (NPYS .GT. 500) GO TO 390
     DO 280 J = 1, 14
       PYS(J,NPYS) = YBUL(J)
280  CONTINUE
     PYS(15,NPYS) = BIND
     GO TO 260
290  CALL ISORT (PYS,15,NPYS,1,NPYS,15,3,-1)
       N = 0
     ID = NTAI + NDAI + 9000
     IF (NPYS .LT. NYSS) GO TO 395
     DO 320 J = 1, NYSS
       N = N + 1
       NRAI = NRAI + 1
       AI(1,NRAI) = ID + N
       AI(2,NRAI) = PYS(2,J)
       AI(3,NRAI) = PYS(3,J)
       AI(4,NRAI) = PYS(6,J)
       AI(5,NRAI) = PYS(5,J)
       AI(6,NRAI) = BFM + 1 - EXP ( BFSD * FRANDN(0.))
       AI(7,NRAI) = BASP
       AI(8,NRAI) = PYS(1,J) + PYS(4,J) / 365.
       AI(9,NRAI) = 0.0
       AI(10,NRAI) = FRANDN(0.) * SPSD + SPM
```

```
      DO 300 K = 1, 4
        AI(K+10,NRAI) = 0.
        AI(K+14,NRAI) = PYS(K+6,J)
        AI(K+18,NRAI) = 0.
        AI(K+22,NRAI) = 0.
        AI(K+26,NRAI) = PYS(K+10,J)
300   CONTINUE
320 CONTINUE
    CALL ISORT(AI,30,500,NFYAI,NRAI,8,3,0)
    DO 325 J = NFYAI, NRAI
325 AI(1,J) = J + 9000
350 NPAI = NTAI - NND
    NTAI = NTAI + NYSS - NND
    NDAI = NDAI + NND
    NAI = MAX0 (NAI-NND, NCDAI)
    NUAI = NEAI - NND
    NEAI = NUAI + NNAI
380 WRITE(14) NDAI, NAI, NUAI, NEAI, NPAI, NTAI
    CALL WRTMAT(AI, 30, NTAI, 14)
    STOP
390 WRITE (6,420)
    STOP
395 WRITE(6,425) NPYS
    STOP
397 WRITE (6, 460)
    STOP
398 WRITE (6, 470)
400 FORMAT ('ENTER THE NUMBER OF BULLS TO ADD, THE MINIMUM INDEX,', /,
   1       ' THE MINIMUM ETA FOR TYPE AND THE DAMS MINIMUM ETA', /,
   2       ' FOR TYPE')
410 FORMAT ('ENTER SELECTION INDEX WEIGHTINGS FOR', /,
   1       'MILK, FAT, PROTEIN AND TYPE')
420 FORMAT ('TOO MANY YOUNG BULLS -', /,
   1       '    INCREASE THE MINIMUMS OR THE DIMENSIONS OF PYS')
425 FORMAT ('ONLY', I4, ' YOUNG BULLS  ELIGABLE -', /,
   1       'DECREASE MINIMUMS OR NUMBER OF BULLS SELECTED')
430 FORMAT (I10, 4F10.2)
440 FORMAT (5F10.3)
450 FORMAT ('ENTER THE CURRENT YEAR (NEXT)')
460 FORMAT ('ERROR :  DREC NOT SORTED PROPERLY')
470 FORMAT ('ERROR :  BUSE NOT SORTED PROPERLY')
    STOP
    END


      SUBROUTINE WRTMAT(WRMATR, ICOL, IROW, INP)
C  *****************************************************************
C  *   This subroutine writes two dimensional matrixes.            *
C  *****************************************************************
      DIMENSION WRMATR(ICOL,IROW)
      WRITE (INP) WRMATR
      RETURN
      END
```

```
      SUBROUTINE REAMAT(REMATR, ICOL, IROW, INP)
C   *******************************************************************
C   *    This subroutine reads two dimensional matrixes.              *
C   *******************************************************************
      DIMENSION REMATR(ICOL,IROW)
      READ (INP) REMATR
   10 RETURN
      END
```

```
C*********************************************************************
C*                    A. I. VIEW PROGRAM                             *
C*                    ------------------                             *
C*                                                                   *
C*      The user enters their index weightings and the number of     *
C*      top proven bulls they want to see.                           *
C*      The program ranks all A.I. bulls with daughter records by    *
C*      index score and lists the following information for the      *
C*      specified number of top bulls:                               *
C*          - SIRE, MATERNAL GRAND SIRE, ETA'S FOR MILK, FAT, PROTEIN, *
C*            TYPE, AND INDEX, CONCEPTION RATE, NUMBER DAUGHTER      *
C*            RECORDS, NUMBER OF HERDS AND SEMEN PRICE.              *
C*                                                                   *
C*      The program also can list all young sires eligible for use   *
C*                                                                   *
C*********************************************************************

      REAL *4  AI(30,500), OUT(12, 400), YUNG(8,100), SINDEX(4)
      LOGICAL *1  PVN/'P'/, YNG/'Y'/, CHR

      CALL FTNCMD('ASSIGN 4=KINN:AI.U;')
   10 WRITE (6,110)
      READ (5,150) SINDEX
      SDT = ABS(SINDEX(1) + SINDEX(2) + SINDEX(3) + SINDEX(4))
      IF (SDT .EQ. 0.) GO TO 10
      DO 20 I = 1, 4
        SINDEX(I) = SINDEX(I) / SDT
   20 CONTINUE
      READ (4) NDAI, NAI, NUAI, NEAI, NPAI, NTAI
      CALL REAMAT(AI, 30, NEAI, 4)
      NBP = 0
      NYB = 0
      DO 40 I = 1, NEAI
C*      Bulls with daughter records                                  *
        ND = AI(13,I)
        IF (ND .LE. 0) GO TO 30
        NBP = NBP + 1
        OUT(1,NBP) = AI(1,I)
        OUT(2,NBP) = AI(2,I)
        OUT(3,NBP) = AI(4,I)
        OUT(4,NBP) = AI(27,I)
        OUT(5,NBP) = AI(28,I)
        OUT(6,NBP) = AI(29,I)
        OUT(7,NBP) = AI(30,I)
        OUT(8,NBP) = OUT(4,NBP) * SINDEX(1) + OUT(5,NBP) * SINDEX(2) +
     1    OUT(6,NBP) * SINDEX(3) + OUT(7,NBP) * SINDEX(4)
        OUT(9,NBP) = AI(12,I) / (AI(11,I) + 1.E-6)
        OUT(10,NBP) = ND
        OUT(11,NBP) = AI(14,I)
        OUT(12,NBP) = AI(7,I)
        GO TO 40
C*      Young bulls                                                  *
   30   NYB = NYB + 1
        YUNG(1,NYB) = AI(1,I)
        YUNG(2,NYB) = AI(2,I)
        YUNG(3,NYB) = AI(4,I)
        YUNG(4,NYB) = AI(27,I)
        YUNG(5,NYB) = AI(28,I)
        YUNG(6,NYB) = AI(29,I)
        YUNG(7,NYB) = AI(30,I)
        YUNG(8,NYB) = YUNG(4,NYB) * SINDEX(1) + YUNG(5,NYB) * SINDEX(2)
     1    + YUNG(6,NYB) * SINDEX(3) + YUNG(7,NYB) * SINDEX(4)
```

```
   40  CONTINUE
       IF (NBP .LE. 0) GO TO 50
       CALL ISORT (OUT, 12, NBP, 1, NBP, 8, 3, -1)
C*       List bulls on the screen                                      *
   50  WRITE (6,120)
       READ (5,160) CHR
       CALL FINDST(PVN, 1, CHR, 1, 1, NX, 1, &60, &60)
       GO TO 70
   60  CALL FINDST(YNG, 1, CHR, 1, 1, NX, 1, &90, &90)
       GO TO 80
   70  WRITE (6,130) NBP
       IF (NBP .LE. 0) GO TO 50
       READ(5,170) NOPB
       IF (NOPB .GT. NBP) NOPB = NBP
       WRITE (6,140)
       CALL PNTMAT(OUT, 12, NOPB)
       GO TO 50
   80  WRITE (6,140)
       IF ( NYB .LE. 0) GO TO 50
       CALL PNTMAT(YUNG, 8, NYB)
       GO TO 50
   90  STOP
  110  FORMAT ('ENTER INDEX WEIGHTINGS FOR MILK, FAT, PROTEIN AND TYPE',
     1      /, '  THEY MUST NOT SUM TO "0"')
  120  FORMAT ('ENTER:  "Y" FOR YOUNG SIRES', /, '      OR "P" FOR',
     1       ' PROVEN SIRES', /, '      OR RETURN TO STOP')
  130  FORMAT ('HOW MANY SIRES DO YOU WANT LISTED ?', /,
     1       ' MAXIMUM', I4)
  140  FORMAT (33X, 'ETA', 15X, 'CONCEP-  NO.  NO.    SEMEN', /,
     1       'BULL   SIRE   MGS    MILK   FAT  PROT  TYPE  INDEX  TION',
     2       '  DAU. HERDS   PRICE')
  150  FORMAT (4F12.3)
  160  FORMAT (A1)
  170  FORMAT (I10)
       END


       SUBROUTINE PNTMAT(DMAT, NCOL, NROW)
       REAL DMAT(NCOL, NROW)
       IF (NCOL .LT. 12) GO TO 10
       WRITE (6,20) DMAT
       RETURN
   10  WRITE (6,30) DMAT
       RETURN
   20  FORMAT (F6.0, 1X, 2F6.0, 1X, 4F6.1, 1X, F6.1,
     1      F6.2, F7.0, F6.0, F8.2)
   30  FORMAT (F6.0, 1X, 2F6.0, 1X, 4F6.1, 1X, F6.1, 22X, '15.00')
       END


       SUBROUTINE REAMAT(REMATR, ICOL, IROW, INP)
C  *******************************************************************
C  *    This subroutine reads two dimensional matrixes.              *
C  *******************************************************************
       DIMENSION REMATR(ICOL,IROW)
       READ (INP) REMATR
   10  RETURN
       END
```

```
C   ********************************************************************
C   *                                                                  *
C   *                           CRDBASE                              *
C   *                                                                  *
C   *    This program reads in the herd summaries from the file        *
C   *    SUMS (after it has been sorted by herd and year) and          *
C   *    the variable names from VARNAMES formated 4X,3(7X,7A4).       *
C   *    It stores the summaries as a three dimensional matrix         *
C   *    in the file SUMMARIES and the names unformated in             *
C   *     VARIABLES ready for analysis.                                *
C   *                                                                  *
C   ********************************************************************
C
C
C
      INTEGER *4 NVAR/162/, VARN(7,162), HDNO(20)
      REAL *4  TSUM(7,24,162), SP(3), SB(4), SP2(3), SB2(4),
     1      SLCB(4), SBHB(4), SFCB(4), SBCB(4), SPCB(4), STCB(4),
     2      SHCB(4), SDCB(4), SLYB(4), SSYB(4), SDYB(4), SLFB(4),
     3      SHFB(4), SBFB(4), SDFB(4), SYSB(4), SPBB(4), SPBI(4),
     4      SUBB(4), SUBI(4), SYBB(4), SYBI(4), SIND(4)
C   *
      CALL FTNCMD ('ASSIGN 2=SUMS;')
      CALL FTNCMD ('ASSIGN 3=VARNAMES;')
      CALL FTNCMD ('ASSIGN 7=VARIABLES;')
      CALL FTNCMD ('ASSIGN 8=SUMMARY;')
      CALL FTNCMD ('SET UVCHECK=OFF;')
      WRITE (6, 100)
      READ (5, 110) NY, NH, NG, IFR
      ING = 0
      IG = 0
      NR = 0
      LH = 0
      IF (IFR .GT. 0) IG = 100
   10 DO 30 I = 1, NH
        LH = LH + 1
        IH = I + IG
        HDNO(LH) = IH
        IHL = (IH - 1) * 100
        DO 20 J = 1, NY
          IHY = IHL + J
          FIND(2'IHY)
          READ (2,END=60) LCH, LY, N1, SP, SB, R1, R2, R3, R4, N2, SP2,
     1        SB2, R5, R6, R7, R8, R9, R10, P1, P2, P3, P4, P5, P6,
     2        P7, P8, P9, P10, C1, C2, C3, C4, C5, C6, C7, T1, T2,
     3        T3, T4, T5, T6, T7, T8, T9, T10, T11, N3, SLCB, N4,
     4        SBHB, N5, SFCB, N6, SBCB, N7, SPCB, N8, STCB, N9, SHCB,
     5        N10, SDCB, N11, SLYB, N12, SSYB, N13, SDYB, N14, SLFB,
     6        N15, SHFB, N16, SBFB, N17, SDFB, N18, SYSB, SPBB, SPBI,
     7        SUBB, SUBI, SYBB, SYBI, SIND
          IF (LCH .NE. IH .OR. LY .NE. J) GO TO 85

          TSUM(J,LH,1) = N1
          TSUM(J,LH,2) = SP(1)
          TSUM(J,LH,3) = SP(2)
          TSUM(J,LH,4) = SP(3)
          TSUM(J,LH,5) = SB(1)
          TSUM(J,LH,6) = SB(2)
          TSUM(J,LH,7) = SB(3)
          TSUM(J,LH,8) = SB(4)
          TSUM(J,LH,9) = R1
          TSUM(J,LH,10) = R2
```

```
TSUM(J,LH,11) = R3
TSUM(J,LH,12) = R4
TSUM(J,LH,13) = N2
TSUM(J,LH,14) = SP2(1)
TSUM(J,LH,15) = SP2(2)
TSUM(J,LH,16) = SP2(3)
TSUM(J,LH,17) = SB2(1)
TSUM(J,LH,18) = SB2(2)
TSUM(J,LH,19) = SB2(3)
TSUM(J,LH,20) = SB2(4)
TSUM(J,LH,21) = R5
TSUM(J,LH,22) = R6
TSUM(J,LH,23) = R7
TSUM(J,LH,24) = R8
TSUM(J,LH,25) = R9
TSUM(J,LH,26) = R10
TSUM(J,LH,27) = P1
TSUM(J,LH,28) = P2
TSUM(J,LH,29) = P3
TSUM(J,LH,30) = P4
TSUM(J,LH,31) = P5
TSUM(J,LH,32) = P6
TSUM(J,LH,33) = P7
TSUM(J,LH,34) = P8
TSUM(J,LH,35) = P9
TSUM(J,LH,36) = P10
TSUM(J,LH,37) = C1
TSUM(J,LH,38) = C2
TSUM(J,LH,39) = C3
TSUM(J,LH,40) = C4
TSUM(J,LH,41) = C5
TSUM(J,LH,42) = C6
TSUM(J,LH,43) = C7
TSUM(J,LH,44) = T1
TSUM(J,LH,45) = T2
TSUM(J,LH,46) = T3
TSUM(J,LH,47) = T4
TSUM(J,LH,48) = T5
TSUM(J,LH,49) = T6
TSUM(J,LH,50) = T7
TSUM(J,LH,51) = T8
TSUM(J,LH,52) = T9
TSUM(J,LH,53) = T10
TSUM(J,LH,54) = T11
TSUM(J,LH,55) = N3
TSUM(J,LH,56) = SLCB(1)
TSUM(J,LH,57) = SLCB(2)
TSUM(J,LH,58) = SLCB(3)
TSUM(J,LH,59) = SLCB(4)
TSUM(J,LH,60) = N4
TSUM(J,LH,61) = SBHB(1)
TSUM(J,LH,62) = SBHB(2)
TSUM(J,LH,63) = SBHB(3)
TSUM(J,LH,64) = SBHB(4)
TSUM(J,LH,65) = N5
TSUM(J,LH,66) = SFCB(1)
TSUM(J,LH,67) = SFCB(2)
TSUM(J,LH,68) = SFCB(3)
TSUM(J,LH,69) = SFCB(4)
TSUM(J,LH,70) = N6
TSUM(J,LH,71) = SBCB(1)
TSUM(J,LH,72) = SBCB(2)
```

```
TSUM(J,LH,73) = SBCB(3)
TSUM(J,LH,74) = SBCB(4)
TSUM(J,LH,75) = N7
TSUM(J,LH,76) = SPCB(1)
TSUM(J,LH,77) = SPCB(2)
TSUM(J,LH,78) = SPCB(3)
TSUM(J,LH,79) = SPCB(4)
TSUM(J,LH,80) = N8
TSUM(J,LH,81) = STCB(1)
TSUM(J,LH,82) = STCB(2)
TSUM(J,LH,83) = STCB(3)
TSUM(J,LH,84) = STCB(4)
TSUM(J,LH,85) = N9
TSUM(J,LH,86) = SHCB(1)
TSUM(J,LH,87) = SHCB(2)
TSUM(J,LH,88) = SHCB(3)
TSUM(J,LH,89) = SHCB(4)
TSUM(J,LH,90) = N10
TSUM(J,LH,91) = SDCB(1)
TSUM(J,LH,92) = SDCB(2)
TSUM(J,LH,93) = SDCB(3)
TSUM(J,LH,94) = SDCB(4)
TSUM(J,LH,95) = N11
TSUM(J,LH,96) = SLYB(1)
TSUM(J,LH,97) = SLYB(2)
TSUM(J,LH,98) = SLYB(3)
TSUM(J,LH,99) = SLYB(4)
TSUM(J,LH,100) = N12
TSUM(J,LH,101) = SSYB(1)
TSUM(J,LH,102) = SSYB(2)
TSUM(J,LH,103) = SSYB(3)
TSUM(J,LH,104) = SSYB(4)
TSUM(J,LH,105) = N13
TSUM(J,LH,106) = SDYB(1)
TSUM(J,LH,107) = SDYB(2)
TSUM(J,LH,108) = SDYB(3)
TSUM(J,LH,109) = SDYB(4)
TSUM(J,LH,110) = N14
TSUM(J,LH,111) = SLFB(1)
TSUM(J,LH,112) = SLFB(2)
TSUM(J,LH,113) = SLFB(3)
TSUM(J,LH,114) = SLFB(4)
TSUM(J,LH,115) = N15
TSUM(J,LH,116) = SHFB(1)
TSUM(J,LH,117) = SHFB(2)
TSUM(J,LH,118) = SHFB(3)
TSUM(J,LH,119) = SHFB(4)
TSUM(J,LH,120) = N16
TSUM(J,LH,121) = SBFB(1)
TSUM(J,LH,122) = SBFB(2)
TSUM(J,LH,123) = SBFB(3)
TSUM(J,LH,124) = SBFB(4)
TSUM(J,LH,125) = N17
TSUM(J,LH,126) = SDFB(1)
TSUM(J,LH,127) = SDFB(2)
TSUM(J,LH,128) = SDFB(3)
TSUM(J,LH,129) = SDFB(4)
TSUM(J,LH,130) = N18
TSUM(J,LH,131) = SYSB(1)
TSUM(J,LH,132) = SYSB(2)
TSUM(J,LH,133) = SYSB(3)
TSUM(J,LH,134) = SYSB(4)
```

```
      TSUM(J,LH,135) = SPBB(1)
      TSUM(J,LH,136) = SPBB(2)
      TSUM(J,LH,137) = SPBB(3)
      TSUM(J,LH,138) = SPBB(4)
      TSUM(J,LH,139) = SPBI(1)
      TSUM(J,LH,140) = SPBI(2)
      TSUM(J,LH,141) = SPBI(3)
      TSUM(J,LH,142) = SPBI(4)
      TSUM(J,LH,143) = SUBB(1)
      TSUM(J,LH,144) = SUBB(2)
      TSUM(J,LH,145) = SUBB(3)
      TSUM(J,LH,146) = SUBB(4)
      TSUM(J,LH,147) = SUBI(1)
      TSUM(J,LH,148) = SUBI(2)
      TSUM(J,LH,149) = SUBI(3)
      TSUM(J,LH,150) = SUBI(4)
      TSUM(J,LH,151) = SYBB(1)
      TSUM(J,LH,152) = SYBB(2)
      TSUM(J,LH,153) = SYBB(3)
      TSUM(J,LH,154) = SYBB(4)
      TSUM(J,LH,155) = SYBI(1)
      TSUM(J,LH,156) = SYBI(2)
      TSUM(J,LH,157) = SYBI(3)
      TSUM(J,LH,158) = SYBI(4)
      TSUM(J,LH,159) = SIND(1)
      TSUM(J,LH,160) = SIND(2)
      TSUM(J,LH,161) = SIND(3)
      TSUM(J,LH,162) = SIND(4)
 20   CONTINUE
 30 CONTINUE
    IF (NR .GE. NG) GO TO 35
    NR = NR + 1
    IG = IG + 100
    GO TO 10
 35 IMS = LH * NY
    DO 40 I = 1, NVAR
      CALL WRTMAT (TSUM(1,1,I), IMS, I, 8)
 40 CONTINUE
 45 DO 50 I = 1, 54
      READ (3, 120) (VARN(J,I), J=1,7), (VARN(J,I+54), J=1,7) ,
   1        (VARN(J,I+108), J=1,7)
 50 CONTINUE
    CALL INMAT (VARN, 7, 162, 7)
    CALL INMAT (HDNO, LH, 1, 7)
    STOP
 60 WRITE(6,130) IH, J
    STOP
 85 WRITE (6, 90) IH, J, LH, LY
 90 FORMAT (/, ' ERROR  NOT PROPERLY SORTED OR MISSING HERD YEAR',
   1       /, '    LOOKING FOR HERD', I4, '  YEAR', I4,
   2       '    FOUND HERD', I4, '  YEAR', I4)
100 FORMAT ('ENTER  NUMBER OF: YEARS; HERDS; GROUPS; AND 1 IF NO ',
   1       'STUDENT HERDS')
110 FORMAT (6I10)
120 FORMAT (4X, 3(7X,7A4))
130 FORMAT ('CAN NOT FIND HERD', I4, '  YEAR', I3)
    STOP
    END
```

```
      SUBROUTINE WRTMAT(WRMATR, ISIZ, LBL, INP)
C *******************************************************************
C *    This subroutine writes two dimensional matrixes.             *
C *******************************************************************
      DIMENSION WRMATR(ISIZ)
      WRITE (INP'LBL) WRMATR
      RETURN
      END
      SUBROUTINE INMAT (IMAT,ICOL,IROW,INP)
      INTEGER *4  IMAT(ICOL,IROW)
      WRITE(INP) IMAT
      RETURN
      END
```

```
C   **********************************************************************
C   *                                                                    *
C   *                              STANL                              *
C   *                                                                    *
C   *    This interactive program does the final statistical analysis    *
C   *    and graphing.  It reads variable names and numbers from         *
C   *    VARIABLES and data from SUMMARIES. Titles and specifications    *
C   *    are entered interactively.  It does an analysis of covariance   *
C   *    with years as the covariate, tests for common slopes within     *
C   *    groups and between groups and does a Student Knewman Keuls      *
C   *    test (SNK) where significant differences in slopes or means     *
C   *    are found.  A graph for each group plots all points and the     *
C   *    common regression and and a final graph has all group           *
C   *    regression lines and the overall common regression.             *
C   *    The statistical analysis is written to the file -STAT and the   *
C   *    the data to -DAT and the graphs to -PLOT.  -PLOT should be      *
C   *    copied to the priter if the program was run with the ap-        *
C   *    pendix "*PRPLOT" , otherwise it should be copied to the         *
C   *    plotter.                                                        *
C   *                                                                    *
C   *    CURRENT MAXIMUMS:    50 years;    100 herds/group;   10 groups; *
C   *                         SNK's test is not done if more than 20     *
C   *                         herds/group  (to modify look at FINDRG)    *
C   **********************************************************************
C   *
C   *
C   *
      INTEGER*4 TNO, HDF, BGDF, TDF, TRDF, GRDF, BDF, MDF, CRDF, GDF,
     1          GMDF, IHGN(100,10), IHHN(100), IGN(10), YLABEL(20),
     2          FSTYR/1/, LSTYR /14/, GLABEL(13,10), NH(100), VARB(7),
     3          VARNAM(7,162), HNO(20)

      EQUIVALENCE (YLABEL(14),VARB(1))

      REAL*4 COV(50), YIELD(50), HMN(100), HB(100), HA(100), GB(10),
     1       GBSE(10), GMN(10), GMNSE(10), GA(10), MSSQ, MMS, MFT, MP,
     2       GBD(10), GMND(10), HBD(100), HBSE(100), EPX(2), TEPY(2),
     3       TSUM(20,40), TEX(20,40), HMND(100)

      LOGICAL*1 FALSE/F/, TRUE/T/, FTGP/T/, CONT/F/ , SNG, FTVB/T/,
     1       PLTP(10)

      DATA MAXGP /10/, MAXYR /7/, MIN /1/, MAXSNK /20/, NTH /9/,
     1    NTY /7/, NTV/162/, DSX/10./, DSY/8./
C   *
      CALL FTNCMD('ASSIGN 1=STUD.SIM;')
      CALL FTNCMD('ASSIGN 2=VARIABLES;')
      CALL FTNCMD('ASSIGN 3=SUMMARY;')
      CALL FTNCMD('ASSIGN 7=-STAT;')
      CALL FTNCMD('ASSIGN 8=-DAT;')
      CALL FTNCMD('ASSIGN 9=-PLOT;')
      CALL DIMVAR (TSUM, TEX, VARNAM, HNO, NTV, NTY, NTH)
      GO TO 20
C   *                                                                    *
C   *     Input choices and titles                                       *
C   *                                                                    *
   10 WRITE (6,320) MIN, MAXYR
   20 WRITE (6,220)
      READ (5,230) NG
      IF (NG .LT. 2) GO TO 160
      WRITE (6,240)
      READ (5,230) IFY, LY
```

```
      IF (IFY .LT. MIN .OR. LY .GT. MAXYR) GO TO 10
      IAY = IFY - 1
      NY = LY - IAY
      XMIN = IAY
      IDF = NY
      IPR = DSX / 2.
      ICT = 0
   22 XMAX = IPR
      IF (IDF .LE. IPR) GO TO 25
      ICT = ICT + 1
      IPR = DSX * ICT
      GO TO 22
   25 WRITE (7,170) IFY, LY
      WRITE (6,180)
      READ (5,190) (YLABEL(L),L=1,20)
      WRITE (7,200) (YLABEL(L),L=1,19)
      CALL ALSIZE(DSX, DSY)
C  *                                                                 *
C  *    For years find minimum and maximum and calculate mean and    *
C  *    variance                                                     *
C  *                                                                 *
      X = 0
      SX = 0
      DO 30 I = IFY, LY
        X = X + I
        SX = SX + I ** 2
        COV(I) = I
   30 CONTINUE
      CALL SMAT(COV, NY)
      XD = SX - X ** 2 / NY
      TMNX = X / NY
      EPX(1) = COV(1)
      EPX(2) = COV(NY)
C  *                                                                 *
C  *    BEGIN LOOP FOR EACH VARIABLE                                 *
C  *                                                                 *
      NVAR = 0
      GO TO 40
   35 WRITE (6,202) IHHN(IH)
   40 NVAR = NVAR + 1
      GXD = 0.
      GXYD = 0.
      GYD = 0.
      GCMN = 0.
      TY = 0.
      TSY = 0.
      TXY = 0.
      TSX = 0.
      TX = 0.
      TNO = 0
      TEPY(1) = 1.E10
      TEPY(2) = -1.E10
      GRSSQ = 0.
      WRITE (6, 205) NVAR
      CALL INIVAR (IAY, NY, VARB, SNG)
      IF (SNG) GO TO 45
      WRITE (6,210)
      READ (5,190) VARB
   45 WRITE (7, 215) NVAR, VARB
      DO 100 I = 1, NG
        HY = 0.
        HSY = 0.
```

```
      HXY = 0.
      HSX = 0.
      HX = 0.
      HYD = 0.
      HXYD = 0.
      HXD = 0.
      HRSSQ = 0.
      IF ( .NOT. FTVB) GO TO 50
      WRITE (6,250) I
      READ (5,190) (GLABEL(L,I),L=1,13)
      WRITE (6,260)
      PLTP(I) = FALSE
      READ (5,280) NH(I), PLTP(I)
      WRITE (6,265) NH(I)
      CALL IARAY(IHGN(1,I), HNO, NH(I), NTH)
      FTGP = FALSE
   50 IF (.NOT. PLTP(I)) GO TO 65
      DO 60 J = 1, 13
        YLABEL(J) = GLABEL(J,I)
   60 CONTINUE
      CALL ALSCAL (XMIN, XMAX, 0.0, 0.0)
      CALL ALAXIS('YEAR', 4, YLABEL, 80)
C  *                                                                  *
C  *  calculate herd sum sq., mean, slope and intercept and add      *
C  *  to herd totals                                                  *
C  *                                                                  *
   65 JH = NH(I)
      DO 90 IH = 1, JH
        CALL HRDVAR(YIELD, IHGN(IH,I))
        CALL DMAT (YIELD, NVAR)
        IHHN(IH) = HNO(IHGN(IH,I))
        JHSY = IH + 240
        Y = 0.
        SY = 0.
        XY = 0.
C  *  Plot points                                                     *
        IF (.NOT. PLTP(I)) GO TO 70
        CALL ALGRAF(COV, YIELD, -NY, -JHSY)
   70   DO 80 K = 1, NY
          Y = Y + YIELD(K)
          SY = SY + YIELD(K) ** 2
          XY = XY + YIELD(K) * COV(K)
   80   CONTINUE
        TNO = TNO + NY
        HY = HY + Y
        HSY = HSY + SY
        HXY = HXY + XY
        HSX = HSX + SX
        HX = HX + X
        YD = SY - Y ** 2 / NY
      IF (YD .EQ. 0.) GO TO 35
        XYD = XY - X * Y / NY
        HBD(IH) = XD
        HMND(IH) = NY
        HMN(IH) = Y / NY
        HB(IH) = XYD / XD
        HA(IH) = HMN(IH) - HB(IH) * TMNX
        HYD = HYD + YD
        HXYD = HXYD + XYD
        HXD = HXD + XD
        HRSSQ = HRSSQ + YD - HB(IH) * XYD
   90 CONTINUE
```

```
C  *                                                                    *
C  *    Calculate group sums squares, mean, slope and intercept and     *
C  *    add to group and overall totals.                                *
C  *                                                                    *
        NTG = NY * JH
        QXD = HSX - HX ** 2 / NTG
        QXYD = HXY - HX * HY / NTG
        QYD = HSY - HY ** 2 / NTG
        GXD = GXD + QXD
        GXYD = GXYD + QXYD
        GYD = GYD + QYD
        GB(I) = QXYD / QXD
        GBD(I) = QXD
        GMN(I) = HY / NTG
        GCMN = GCMN + GMN(I)
        GMND(I) = JH * NY
        GA(I) = GMN(I) - GB(I) * TMNX
        TY = TY + HY
        TSY = TSY + HSY
        TXY = TXY + HXY
        TSX = TSX + HSX
        TX = TX + HX
        CRSSQ = QYD - GB(I) * QXYD
        GRSSQ = GRSSQ + CRSSQ
        CALL SLINE(EPX, TEPY, GB(I), GA(I), -45, PLTP(I))
        CALL ALDONE
        CRDF = NTG - 2
        GMSSQ = HYD - GB(I) * HXYD
        GMDF = NTG - JH - 1
        CRMS = CRSSQ / CRDF
        GMED = SQRT (CRMS / CRDF)
        GMMS = GMSSQ / GMDF
        GBED = SQRT (GMMS / HXD)
        HDF = NY * JH - 2 * JH
        GDF = JH - 1
        IGN(I) = I
        IF (JH .LT. 2) GO TO 100
        WRITE (7, 217) I
        CALL ANCOUT (NTG, QYD, GDF, HDF, HRSSQ, GMDF, GMSSQ, CRSSQ,
     1       GMN(I), GMED, GB(I), GBED, GA(I), HB, HBD, HMN, HMND,
     2       HA, IHHN, JH, 1)
  100 CONTINUE
C  *                                                                    *
C  *    Calculate total sums of squares, mean, slope, intercept and     *
C  *    all mean squares, F-values and probabilities                    *
C  *                                                                    *
      TXD = TSX - TX ** 2 / TNO
      TXYD = TXY - TX * TY / TNO
      TYD = TSY - TY ** 2 / TNO
      TMN = TY / TNO
      TB = TXYD / TXD
      TSSQ = TYD - TB * TXYD
      TDF = TNO - 2
      TMS = TSSQ / TDF
      GCB = GXYD / GXD
      GCMN = GCMN / NG
      GCA = GCMN - GCB * TMNX
      CMNSE = SQRT(TMS / TDF)
      TRSSQ = GYD - GCB * GXYD
      TRDF = TNO - NG - 1
      TRMS = TRSSQ / TRDF
      CBSE = SQRT(TRMS / GXD)
```

```
      GRDF = TNO - 2 * NG
      BDF = NG - 1
      MDF = NG - 1
      WRITE (7, 219)
      CALL ANCOUT (TNO, TYD, MDF, GRDF, GRSSQ, TRDF, TRSSQ, TSSQ,
     1     GCMN, CMNSE, GCB, CBSE, GCA, GB, GBD, GMN, GMND, GA, IGN,
     2     NG, 2)
C  *                                                                  *
C  *    Set axis scale for well placed graph                          *
C  *                                                                  *
      PWY = 10.
  112 PWY = PWY / 10.
      PWR = (TEPY(2) - TEPY(1)) / PWY
      IF (PWR .LE. 0.) GO TO 112
  114 IF (PWR .LT. DSY) GO TO 116
      PWY = PWY * 10.
      PWR = (TEPY(2) - TEPY(1)) / PWY
      GO TO 114
  116 SPW = PWY / 10.
      IMIN = TEPY(1) / SPW - 1.
      IMAX = TEPY(2) / SPW + 1.
      IND = DSY
  110 IREM = MOD (IMAX-IMIN, IND)
      IF (IREM .LE. 1) GO TO 120
      ITR = (IND -  IREM) / 2
      IF (ITR .LT. 1) ITR = 1
      IMAX = IMAX + ITR
      IMIN = IMIN - ITR
      GO TO 110
  120 IF (IREM .EQ. 1) IMIN = IMIN + 1
      YMIN = IMIN * SPW
      YMAX = IMAX * SPW
      CALL ALSCAL(XMIN, XMAX, YMIN, YMAX)
      CALL ALAXIS('YEAR', 4, VARB, 28)
C  *                                                                  *
C  *    Plot common and group regressions                             *
C  *                                                                  *
      CALL SLINE(EPX, TEPY, GCB, GCA, -45, TRUE)
      DO 130 I = 1, NG
        CALL SLINE(EPX, TEPY, GB(I), GA(I), I, TRUE)
  130 CONTINUE
  140 CALL ALDONE
      IF (NVAR .GE. 10) GO TO 150
      FTVB = FALSE
      WRITE (6,350)
      READ (5,360) CONT
      IF (.NOT. CONT) GO TO 40
  150 CALL OMAT(NH, NG)
      WRITE (7,370)
      STOP
  160 WRITE (6,330)
      STOP
  170 FORMAT (';', ///, 2X, 'ANALYSIS OF COVARIANCE AND SLOPE TEST ',
     1     'WITH YEARS (', I3, '-', I3, ') AS THE COVARIATE', /, '+',
     2       1X, 75('_'))
  180 FORMAT (/, 'ENTER THE TITLE FOR THIS RUN', /, '( LABEL WILL BE ',
     1  'TRUNCATED TO 75 CHARACTERS )')
  190 FORMAT (20A4)
  200 FORMAT (//, 2X, 19A4, /, '+', 1X, 76('_'))
  202 FORMAT (' ERROR:  ALL VALUES FOR HERD', I4, ' WERE EQUIVALENT')
  205 FORMAT (/, 'VARIABLE  #', I3)
  210 FORMAT (/, 'ENTER THE VARIABLE NAME', /, '( LABEL WILL BE ',
```

```
   1'TRUNCATED TO 28 CHARACTERS )')
215 FORMAT (///, 10X, 'VARIABLE #', I4, 6X, 7A4, /, '+', 28X,
   1        28('_'), //)
217 FORMAT (//, 20X, 'GROUP', I4)
219 FORMAT (//, 21X, 'FINAL - OVERALL', /)
220 FORMAT ('ENTER THE NUMBER OF GROUPS')
230 FORMAT (3I10)
240 FORMAT (/, 'ENTER THE FIRST YEAR AND LAST YEAR')
250 FORMAT (/, 'ENTER GROUP', I4, '  LABEL ', /, 'GROUP NO., HERDS ',
   1'AND TREATMENT', /, '( LABEL WILL BE TRUNCATED TO 52 CHARACTERS',
   1 ' )')
260 FORMAT (/, 'ENTER THE NUMBER OF HERDS IN THIS GROUP', /,
   1        ' (PUT A "T" AT THE END TO PLOT HERDS)')
265 FORMAT (/, 'ENTER THE', I3, ' HERDS ')
280 FORMAT (I10, L1)
290 FORMAT (/, 10X, 'GROUP', I4, '  TEST OF COMMON SLOPES', /, 12X,
   1        'F-VALUE', F12.4, 7X, 'DF', I4, ' ,', I4, 7X, 'PROB.',
   2        F12.5)
320 FORMAT (/, 'ERROR  ONLY YEARS', I4, '  TO', I4, ' AVAILABLE')
330 FORMAT (/, 'ERROR  THERE MUST BE AT LEAST 2 GROUPS')
340 FORMAT (/, 'ERROR   MAXIMUM IS', I4, ' MINIMUM IS', I4)
350 FORMAT (/, 'PRESS RETURN FOR ANOTHER VARIABLE OR ENTER "T"',
   1        ' TO STOP')
360 FORMAT (L1)
370 FORMAT (/, 8X, 'ANALYSIS COMPLETE')
    END


      SUBROUTINE DIMVAR (VAR,TWT,VARNAM,HNO,NVX,NTY,NTH)
C   ******************************************************************
C   *                                                                *
C   *    This subroutine returns an array of the desired yield       *
C   *    variable for the given herd over the specified years.       *
C   *    It can be simple variables or the weigted average of a number *
C   *    of variables.                                               *
C   *                                                                *
C   ******************************************************************
C   *
C   *
      REAL*4 HERD(50), SWT(50), SN(100), VAR(NTY,NTH), TWT(NTY,NTH)
      INTEGER*4 LV(100), LW(100), VARNAM(7,NVX), CVAR(7),
     1        HNO(NTH)
      LOGICAL *1  TRUE/T/, FALSE/F/, SMP, CHNG
      READ (2) VARNAM
      READ (2) HNO
      KTV = 1
      RETURN

C   *                                                                *
C   *    Define calculations                                         *
C   *                                                                *
      ENTRY INIVAR (IAY, NY, CVAR, SMP)
      IF (TRUE) GO TO 3
   10 WRITE (6,130)
      READ (5,140) KTV
      GO TO (3, 11, 20), KTV
    2 WRITE (6, 155)
   3  WRITE (6,150)
      READ (5,140,ERR=2) LVS
      WRITE (6,160) (VARNAM(J,LVS), J=1,7)
      READ (5,145) CHNG
```

```
      IF (CHNG) GO TO 3
      DO 5 I = 1, 7
        CVAR(I) = VARNAM(I,LVS)
    5 CONTINUE
      SMP = TRUE
      READ (3'LVS) VAR
      RETURN
   11 READ (5, 140) KV
      IF (KV .GT. 20) GO TO 50
      DO 14 I = 1, KV
   12   WRITE (6,170) I
        READ (5,140) LV(I), SN(I)
        WRITE (6,180) (VARNAM(J,LV(I)), J=1,7)
        READ (5,140) ICK
        IF (ICK .NE. 0) GO TO 12
   14 CONTINUE
      SMP = FALSE
      RETURN
   20 READ (5, 140) KV
      IF (KV .GT. 20) GO TO 50
      DO 40 I = 1, KV
   30   WRITE (6,170) I
        READ (5,140) LV(I), LW(I), SN(I)
        WRITE (6,180) (VARNAM(J,LV(I)), J=1,7), (VARNAM(J,LW(I)),J=1,7)
        READ (5,140) ICK
        IF (ICK .NE. 0) GO TO 30
   40 CONTINUE
      WRITE (6,180)
      RETURN
   50 WRITE (6,190)
      GO TO 10
C  *
C  *
C  *
      ENTRY HRDVAR(HERD,NH)
      IF (KTV .GT. 1) GO TO 80
C  *    Simple variable                                                *
   60 DO 70 I = 1, NY
        HERD(I) = VAR(I+IAY,NH)
   70 CONTINUE
      RETURN
C  *    Calculate weighted average                                     *
   80 DO 90 I = 1, NY
        HERD(I) = 0.0
        SWT(I) = 0.0
   90 CONTINUE
      DO 120 K = 1, NV
        READ (3'LV(K)) VAR
        READ (3'LW(K)) TWT
        DO 100 I = 1, NY
          WT = TWT(I+IAY,NH) * SN(K)
          HERD(I) = HERD(I) + VAR(I+IAY,NH) * WT
          SWT(I) = SWT(I) + WT
  100   CONTINUE
        DO 110 I = 1, NY
          HERD(I) = HERD(I) / SWT(I)
  110   CONTINUE
  120 CONTINUE
  130 FORMAT ('HOW MANY SUMMARY VARIABLES ARE TO BE AVERAGED ?', /,
     1        'PRESS RETURN FOR ONLY ONE.')
  140 FORMAT (2I10, F10.2)
  145 FORMAT (L1)
```

```
 150 FORMAT ('ENTER THE VARIABLE LOCATION NUMBER.')
 155 FORMAT ('ERROR:   DID YOU FORGET A COMMA ?')
 160 FORMAT (/, 'THE VARIABLE IS:', 4X, 7A4, /, 'PRESS RETURN IF OK',
    1'... OR ENTER "T" TO RETRY.')
 170 FORMAT ('ENTER THE VARIABLE', I4, ' LOCATION NUMBER,', /,
    1        '       THE WEIGHT VARIABLE LOCATION NUMBER,', /,
    2        '   AND A REAL NUMBER WEIGHTING FACTOR (OPTIONAL).')
 180 FORMAT ('THE VARIABLE IS ', 7A4, /, 'THE WEIGHT VARIABLE IS ',
    1        7A4, /, 'PRESS RETURN IF OK...OR ENTER 1 TO RETRY.')
 190 FORMAT ('ERROR THE MAXIMUM IS 20', //)
     RETURN
     END


     SUBROUTINE ANCOUT (TOB, TSSQ, MDF, E1DF, E1SSQ, E2DF,
    1           E2SSQ, E3SSQ, CMN, CMNSE, CB, CBSE, CA, B, BED,
    2           AM, AMED, A, ILBL, NO, IT)

     REAL *4  B(100), BED(100), AM(100), AMED(100), A(100),
    1      AMSE(100), BSE(100)
     REAL *8  TAN(2)/' HERD    ', 'GROUP    '/

     INTEGER *4  TOB, E1DF, MDF, E2DF, E3DF, TDF, CRDF/1/,
    1         ILBL(100)
C  *
     TDF = TOB - 1
     TMS = TSSQ / TDF
     BSSQ = E2SSQ - E1SSQ
     BMS = BSSQ / MDF
     E1MS = E1SSQ / E1DF
     BFV = BMS / E1MS
     BPB = FPROB (BFV, MDF, E1DF)
     AMSSQ = E3SSQ - E2SSQ
     AMMS = AMSSQ / MDF
     E2MS = E2SSQ / E2DF
     AMFV = AMMS / E2MS
     AMPB = FPROB (AMFV, MDF, E2DF)
     CRSSQ = TSSQ - E3SSQ
     CRFV = CRSSQ / E2MS
     CRPB = FPROB ( CRFV, CRDF, E2DF)


C  *                                                                 *
C  *   Output ANCOVA table and table of means, slopes and intercepts *
C  *                                                                 *
     WRITE (7,300) TOB, TDF, TSSQ, MDF, BSSQ, BMS, BFV, BPB, E1DF,
    1      E1SSQ, E1MS, MDF, AMSSQ, AMMS, AMFV, AMPB, CRDF, CRSSQ,
    2      CRSSQ, CRFV, CRPB, E2DF, E2SSQ, E2MS
     CALL SNK(B, BSE, BED, E1MS, E1DF, ILBL, NO, IT, 2, BPB)
     CALL SNK(AM, AMSE, AMED, E2MS, E2DF, ILBL, NO, IT, 1, AMPB)
     WRITE (7,310) TAN(IT), CMN, CMNSE, CB, CBSE, CA, (ILBL(I), AM(I),
    1      AMSE(I), B(I), BSE(I), A(I), I=1,NO)
     RETURN
 300 FORMAT (15X, 'ANALYSIS OF COVARIANCE  (', I4, ' OBSERVATIONS)'
    1      , //, 7X, 'SOURCE', 5X, 'DF', 6X, 'SUM SQ', 5X, 'MEAN SQ',
    2        5X, 'F-VALUE', 7X, 'PROB', //, 9X, 'TOTAL', I6,
    *        2X, G13.5, /, 8X, 'SLOPES', I6, 2X, G13.5,
    3        G12.4, G12.4, G11.3, /, 7X, 'ERROR 1', I6, 2X, G13.5,
    4        G12.4, /, 9X, 'MEANS', I6, 2X, G13.5, G12.4, G12.4, G11.3,
    5        /, 2X, 'COMMON SLOPE', I6, 2X, G13.5, G12.4, G12.4, G11.3,
```

```
    6        /, 7X, 'ERROR 2', I6, 2X, G13.5, G12.4)
  310 FORMAT (//, 2X, A8, 5X, 'MEAN', 6X, 'S.E.', 8X, 'SLOPE',
    1        6X, 'S.E.', 5X, 'INTERCEPT', /, 1X, 'COMMON', F12.3,
    2        F10.3, F13.3, F10.3, F14.3, 10(/,1X,I5,F13.3,F10.3,F13.3,
    3          F10.3, F14.3))
      END


      SUBROUTINE SNK(AMN, SE, SED, SEMS, NDF, LBL, NS, JS, MS, PR)
C   ******************************************************************
C   *                                                                *
C   *    This subroutine completes the calculation of the standard   *
C   *    errors, can output the values and standard errors and does  *
C   *    a students kneuman kuels test if significant differences and *
C   *    more than two values.                                       *
C   *                                                                *
C   ******************************************************************
C   *
C   *
      REAL*4 SE(NS), SED(NS), AMN(NS), AMNLB(2,100), RNG(100)
      INTEGER*4 LBL(NS), LSET(100), NLBL(100)

      REAL*8 QUAT(3) /'   HERD ', '  GROUP ', ' COMMON '/, PARM(2) /
     1       '   MEAN ', '  SLOPE '/
      LOGICAL*1 SIG

      DO 10 I = 1, NS
        SE(I) = SQRT(SEMS/SED(I))
   10 CONTINUE
   20 IF (PR .GT. .05 .OR. NS .LT. 3) RETURN
      IF (NS .GT. 20) GO TO 130
      WRITE (7,170) QUAT(JS), PARM(MS)
      CALL FINDRG(RNG, NS - 1, NDF)
      DO 30 I = 1, NS
        AMNLB(1,I) = AMN(I)
        AMNLB(2,I) = I
   30 CONTINUE
      CALL ISORT(AMNLB, 2, 100, 1, NS, 1, 3, 0)
      LSET(NS) = NS
      NI = NS - 1
      NSET = 0
      LLS = 0
      DO 90 I = 1, NI
        IC = MAX0 (I+1, LLS)
        IRV = IC + NS
        DO 50 J = IC, NS
          JC = IRV - J
          CALL SIGCHK(RNG(JC - I), AMNLB(1,I), AMNLB(1,JC), SEMS,
     1          SED(INT(AMNLB(2,I))), SED(INT(AMNLB(2,JC))), SIG)
          IF ( .NOT. SIG) GO TO 60
   50   CONTINUE
        LSET(I) = I
        LIS = I
        GO TO 70
   60   LSET(I) = JC
        LIS = JC
   70   IF (LIS .GT. LLS) NSET = NSET + 1
        LLS = LIS
   90 CONTINUE
      IF (LSET(NI) .LT. NS) NSET = NSET + 1
  100 WRITE (7,180) NSET
```

```
      LID = 1
      DO 120 I = 1, NS
        IF (LSET(I) .LT. LID) GO TO 120
        IL = LSET(I)
        N = 0
        DO 115 J = LID, IL
          N = N + 1
          NLBL(N) = LBL(INT(AMNLB(2,J)))
  115   CONTINUE
        LID = IL + 1
        CALL PRTLN (NLBL, N)
  120 CONTINUE
      RETURN
  130 WRITE (6,200)
  140 FORMAT (/, 6X, A8, 20I12)
  150 FORMAT (5X, A8, 1X, 20F12.3)
  160 FORMAT (6X, ' S.E.     ', 20F12.3)
  170 FORMAT (/, '    STUDENT NEWMAN KUELS TEST  -', 2A8, '''S', /)
  180 FORMAT (/, 3X, 'THERE ARE', I5, '  HOMOGENOUS SUBSETS', /)
  200 FORMAT ('MORE THAN 20 SO NO MULTIPLE RANGE TEST')
      RETURN
      END


      SUBROUTINE PRTLN (LARR, ND)
      DIMENSION LARR(ND)
      WRITE (7, 10) LARR
      RETURN
   10 FORMAT (3X, '(   ', 16(I4,','))
      END


      SUBROUTINE FINDRG(RNG, N, NDF)
C  *
C  *for N<20
C  *
      REAL*4 RNG(N), STUD(35)
      INTEGER *4  MDF(6)/20,24,30,40,60,120/
      WRITE (7,40) NDF
      IF (NDF .GT. 20 .AND. NDF .LE. 120) GO TO 10
      READ (1'NDF) RNG
      WRITE (7,50) RNG
      RETURN
   10 LN = 1
   20 LN = LN + 1
      IF (MDF(LN) .LT. NDF) GO TO 20
      IHDF = MDF(LN)
      ILDF = MDF(LN - 1)
      DIF = IHDF - ILDF
      DIL = NDF - ILDF
      DFR = DIL / DIF
      READ (1'ILDF) RNG
      READ (1'IHDF) STUD
      DO 30 I = 1, N
        RNG(I) = RNG(I) - DFR * (RNG(I) - STUD(I))
   30 CONTINUE
      WRITE (7,50) RNG
   40 FORMAT (3X, 'RANGES FOR ALPHA=0.05 AND', I4, ' DF')
   50 FORMAT (1X, 8F10.4)
      RETURN
```

```
      END


      SUBROUTINE SIGCHK(Q, VL1, VL2, SMS, D1, D2, SIG)
      LOGICAL*1 TRUE /T/, FALSE /F/, SIG
      SIG = FALSE
      TS = SQRT(SMS)
      DS = SQRT((D1 + D2)/(2*D1*D2))
      CRIT = Q * TS * DS
      DIF = VL2 - VL1
      IF (DIF .GT. CRIT) SIG = TRUE
      IF (DIF .LT. 0.) WRITE (6, 10)
      RETURN
   10 FORMAT ('PROBLEMS WITH STUDENT KNEWMAN KEULS TEST')
      END
C
C
C
      SUBROUTINE IARAY (IAR, HNO, IS, MAX)
      INTEGER *4  IAR(IS), HNO(MAX)
   10 READ (5, 50) IAR
      DO 20 I = 1, IS
        DO 15 J = 1, MAX
          IF (IAR(I) .EQ. HNO(J)) GO TO 17
   15   CONTINUE
        GO TO 30
   17   IAR(I) = J
   20 CONTINUE
      RETURN
   30 WRITE (6, 60) I, IAR(I)
      GO TO 10
   50 FORMAT (10I8)
   60 FORMAT (/, 'THE', I4, 'TH HERD', I4, ' DOES NOT EXIST', /,
     1     'RE-ENTER THE LINE')
      END



      SUBROUTINE SLINE(X, TY, B, A, N, LIN)
C   ********************************************************************
C   *    This subroutine plots and labels a regression line and returns  *
C   *    extreme Y's for final graph scale.                              *
C   ********************************************************************
      DIMENSION X(2), Y(2), TY(2)

      LOGICAL *1 LIN
C
C**      STATEMENT FUNCTION                                          ***
C
      YV(XV) = A + XV * B
C   *
      Y(1) = YV(X(1))
      Y(2) = YV(X(2))
      TY(1) = AMIN1(TY(1),Y(1), Y(2))
      TY(2) = AMAX1(TY(2),Y(2), Y(1))
      IF (.NOT. LIN) RETURN
      NSG = N + 240
      CALL ALGRAF(X(1), Y(1), -1, -NSG)
      CALL ALGRAF(X(2), Y(2), -1, -NSG)
      RETURN
```

```
C     CX = (X(2) - X(1)) / 50.
C     GX = X(2)
C     GY = YV(GX)
C     CALL ALGRAF(GX, GY, -1, -NSG)
C     IF (NSG .EQ. 195) RETURN
C     GX = GX - 2. * CX
C     GY = YV(GX)
C     CALL ALGRAF(GX, GY, -1, -215)
C     GX = GX - CX
C     GY = YV(GX)
C     CALL ALGRAF(GX, GY, -1, -199)
C     RETURN
      END


      SUBROUTINE SMAT(COV, NY)
      REAL*4 OUT(1000,10), COV(NY), YLD(50)
      INTEGER*4 NVC /0/, IC /1/, NHG(10)
      NAY = NY - 1
      RETURN
C  *
      ENTRY DMAT (YLD, NVR)
      IF (NVC .NE. NVR) IC = 1
      NVC = NVR
      LC = IC + NAY
      N = 0
      DO 10 I = IC, LC
        N = N + 1
        OUT(I,NVC) = YLD(N)
   10 CONTINUE
      IC = LC + 1
      RETURN
C  *
      ENTRY OMAT(NHG,NG)
      IY = 0
      DO 40 I = 1, NG
        NH = NHG(I)
        DO 30 J = 1, NH
          DO 20 K = 1, NY
            IY = IY + 1
            WRITE (8,50) I, COV(K), (OUT(IY,IV),IV=1,NVC)
   20     CONTINUE
   30   CONTINUE
   40 CONTINUE
   50 FORMAT (I4, 20F12.4)
      RETURN
      END
```