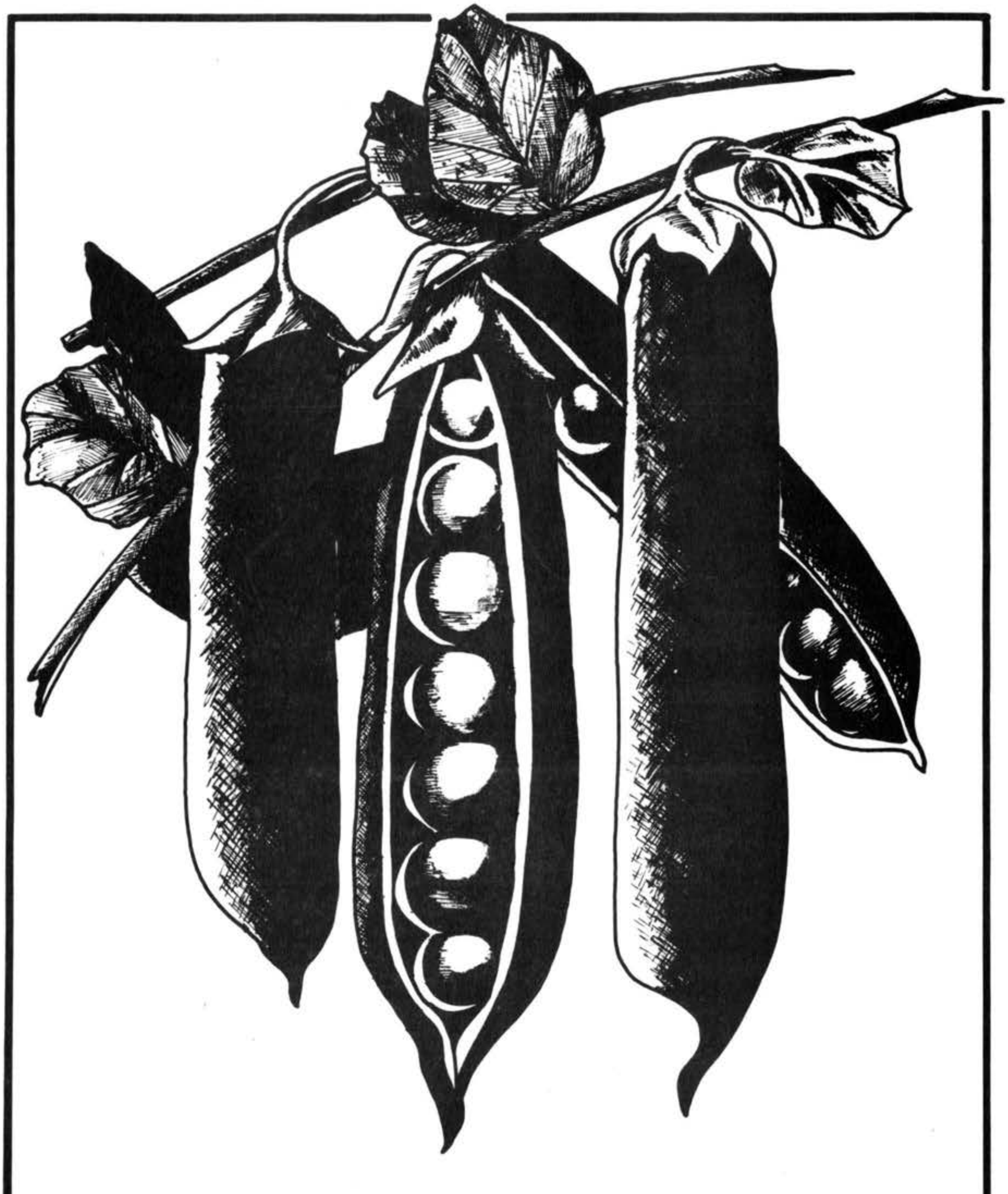


THESIS

PSIM: A CROP SCHEDULING SIMULATION

Submitted by

Sue Silver Mattingly

Horticulture Department

In partial fulfillment of the requirements

for the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Fall, 1979

COLORADO STATE UNIVERSITY

Fall, 1979

WE HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER OUR SUPERVISION BY SUE SILVER MATTINGLY ENTITLED PSIM: A CROP SCHEDULING SIMULATION BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE.

Committee on Graduate Work

James A. Smith

Elisabeth N. Derbyshire

Frank D. Moore III

Charles W. Bostrom

Adviser

ABSTRACT OF THESIS

PSIM: A CROP SCHEDULING SIMULATION

PSIM is a simulation model of raw product supply for a commercial pea canning operation. The primary purpose of the computer model is generation of planting and harvest schedules based on the heat unit system for predicting crop maturation.

The simulation will schedule up to 30 plantings, each consisting of 20 subplantings, for as many as ten cultivars. Output includes a planting schedule, a table of projected harvest dates based on historical temperature data, final harvest dates based on a current year's temperature data, and yield figures. Maturity and yield are treated as stochastic elements in the model and samples are taken from normal distributions for both characteristics.

Model rationale is presented along with a program listing, technical and user documentation. The heat unit system itself is examined in a review of the literature.

Sue Silver Mattingly
Horticulture Department
Colorado State University
Fort Collins, Colorado 80523
Fall, 1979

ACKNOWLEDGEMENTS

I would like to express sincere gratitude to my major adviser, Dr. C. W. Basham. His guidance and thoughtful contributions added immeasurably to the quality of the work.

I also want to extend my appreciation to Ms. E. N. Derbyshire for so generously sharing her programming expertise, and to Dr. F. D. Moore, III for his many constructive comments.

Special thanks to Ruth Elliott for the frontispiece illustration, and to M. Ali Harivandi for his moral support throughout.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

The development of agricultural systems models has increased greatly in recent years. Diversity in these models is also large due mainly to the interdisciplinary nature of agriculture. Although numerous analytic models exist, many agricultural modelers have turned to simulation to represent the dynamic and often stochastic systems of the real world. In situations where deterministic and optimum solutions do not exist, simulation makes it possible to experiment with alternative decision rules to determine which are most useful in studying a system's behavior. The core of this thesis, program PSIM, is a simulation model of raw product scheduling for a commercial pea canning operation. The model's primary purpose is generation of a planting schedule, yield, and harvest information.

Although most vegetable canners in the United States grow a small percentage of the crops they process (typically in order to cover the earliest and latest portions of the canning season when risk for the grower is highest), the majority of any canning crop is contracted to farmers near the cannery. In addition to selecting the sites where crops will be grown, the canner retains responsibility for cultivar selection, some cultural practices, and scheduling of planting dates. This involvement allows yield forecasts and attempts to schedule harvests to use cannery capacity as efficiently as possible.

A planting schedule is critical in any attempt to forecast raw product flow since plant development and maturation, within genetic limitations, are dynamic responses to environment. The primary scheduling tool used by the canning industry is the "degree day" or "heat unit" system. This system, which considers plant development only in relation to temperature, entails keeping various records for each season as well as compiling long term data. Maturity predictions must be made throughout a given growing season and assessments made of the probable yield outcome (both quantity and quality) for that season. Strategic managerial decisions are based on these assessments. Currently, such work is done via pencil and worksheet, and repeated reference to pertinent data is both a time-consuming and organizationally complex job. There is also little opportunity to experiment with decision rules since results are known only at season's end.

The simulation model developed here, by providing a vehicle to test decision rules, can aid in both development of a planting schedule and in testing assumptions made in the process of predicting crop maturity.

The model was developed for peas. However, other crops such as sweet corn and snap beans are also scheduled by the heat unit system. In its present form, PSIM is not developed to handle species other than peas, but with minor changes the program could be used in modelling other canning crops.

REVIEW OF LITERATURE

Introduction

A review of computer models clearly shows that the purpose for which it is designed determines a model's general form. Agricultural models may be theoretical, applied, or a combination of the two, may focus on such areas as economics, physiology or mechanization, and may vary widely in scale. (Purpose generally determines both focus and scale.) Once the purpose is clear, the modelling process itself becomes highly personalized: there are no books on how to model per se. For that reason, although a number of agricultural models were examined (1, 11, 21, 23, 39, 40, 41, 42, 48), the literature review for this model focuses on the basic concepts which determine the purpose and constitute working assumptions of the model itself. The primary purpose of the model is representation of that portion of the commercial canning industry which relies on a heat unit system for crop scheduling. Thus, the heat unit system is the key concept examined in the literature.

Heat Unit System: Definition and Use

The degree day (dday) or heat unit system (12) assumes that for each plant species there exists a base or threshold temperature below which growth does not occur. By subtracting a plant's base temperature from the daily mean temperature, a degree day accumulation is obtained. If the daily mean is less than or equal to the base temperature, the dday value is zero--i.e., no growth occurs. The specific dday or heat unit requirement[1] for a given cultivar to mature may be called the heat maturity constant, the summation constant, the varietal index, or the remainder index. This value is used as a predictor of crop maturation.

Selection of an appropriate base temperature is crucial to an accurate maturity forecast based on the heat unit system. In practice, the base temperature for a given location and cultivar is determined by a process of elimination--i.e., the heat unit summation from a series of plantings is calculated using several base temperatures and the one resulting in the least variability (as measured by minimum coefficient of variation) is selected.

Arnold (3) suggests that use of an incorrect base temperature is responsible for much of the variation in heat unit requirements reported in the literature. Findings of increased heat unit requirements in warm compared to cool parts of the season (16, 25, 38) and in warm compared to cool years (25) could, according to Arnold,

---

[1] Use of the term "heat" when referring to temperature is of course inaccurate, since calories and degrees are not equivalent. However, the terms "heat unit" and "heat requirement" are widely used both in the literature and by industry to describe plant/temperature interaction. Consequently, they are also used here in that context.

be the result of using a too high base temperature. Thus as the mean temperature during a particular developmental period increases, a high base temperature results in an increased heat unit requirement. (Likewise, use of a low base temperature would decrease the heat unit requirement under such conditions. Evidence from the literature shows a preponderance of increasing values, however, suggesting use of a too high base.) In any case, the greater the difference between the correct and the selected base temperature, the greater the variability in heat unit summations.

Arnold also states (3) that the data from which a base temperature is calculated should involve a temperature range "normally encountered" in the developmental phase for which it is to be used. Since, for example, the temperature range between blossoming and harvest of peas is likely to be higher than the range from planting to blossoming, he suggests that different base temperatures may be appropriate. Wang (44) states that not only is the base temperature not a constant throughout the life of a plant, but that no base temperature should be used for longer than a week. Determining base temperatures on a weekly basis, however, requires detailed knowledge of the relationship between development and temperature. Unfortunately, this relationship is not precisely known (4). Data available on optimum temperatures for germination and vegetative and reproductive growth could perhaps be used to calculate two or three base temperatures, each for a major developmental period. That this has not been done suggests that the error introduced by use of a single base temperature is not unacceptable. In addition, no work exists to indicate that use of several base temperatures increases accuracy enough to warrant the additional calculations.

The most widely used base temperature for peas is 40°F. Although Hope (22) varied base temperatures from 30° to 60° at two degree intervals and concluded that 36° was the best base temperature when heat summations are derived from daily means, Wang (44) cites evidence that no base temperature between 32° and 50° improves the accuracy of the heat unit system. It seems evident that, by employing a single base throughout the season, the heat unit system in effect relies on a mean base temperature.

Much criticism of the heat unit system stems from its assumption of a linear relationship between development and temperature. This criticism arises because it is well-known that growth rate is a curvilinear function of temperature, and most evidence also suggests curvilinearity in the developmental rate-temperature relationship (4).[1] One defense of the heat unit system in this regard is that the development-temperature relationship is probably near-linear over the range of temperatures normally encountered during a growing season. More to the point, however, is the fact that the heat unit system is not really concerned with the exact nature of either the temperature-growth rate or temperature-developmental rate relationship. It is essentially an empirical index of the maturation process based on a temperature parameter. As such, its value need only be assessed by the accuracy of its prediction of phenological events.

That the dday system ignores many factors which influence plant development is obvious. Humidity, soil fertility, solar radiation, duration of light, and soil moisture are the most conspicuously absent physical factors. Clearly, temperature does not influence development independent of these parameters. Difficulties arise,

---

[1] A distinction must be made between development and growth. Growth may be thought of as an increase in the weight or volume of either a single plant part or the plant as a whole. Development, on the other hand, generally refers to a sequential process which may begin with germination and proceed through the phases of juvenility, maturation (the shift from vegetative to reproductive growth) and senescence. Both growth and the ordered sequence of developmental events are the result of gene-environment interactions. However, the effect of a given environmental factor, e.g. temperature, on the rate of growth and its effect on the rate at which a plant progresses from one stage of development to another may not be exactly the same. And although the two effects may be related, the precise nature of such a relationship, if it exists, is unknown.

however, in assessing their impact either singly or in combination.

No work has been done, for example, on the effects of humidity on the development of peas. Nonnecke and coworkers (29), investigating the effects of both temperature and humidity on growth, conclude that there is no significant interaction between the two parameters in that regard.

Both soil fertility and solar radiation, on the other hand, are generally treated as constants in predicting plant response. In the case of fertility, which can be substantially controlled, this seems reasonable. The exclusion of solar radiation's effect on development is more difficult to justify since not all light effects on plants are reflected by temperature measurements. In areas and seasons of relatively low solar radiation more ddays are required at a given temperature to bring a plant to a given stage of development than in areas and seasons with high solar radiation (10). To date, however, no widely accepted method exists for incorporating insolation in maturation predictions.

Growth chamber work by Reath and Wittwer (34) on the photoperiod-temperature interaction indicates that longer daylight periods reduce the total heat requirement of peas. Other work (19, 31) corroborates these findings. Although the expression (a)ddays x (b)photoperiod, where (a) and (b) are constants, would therefore probably predict maturity more accurately than ddays alone, the magnitude of the photoperiod effect on developmental rate cannot be easily measured in the field (3). Until a system is developed for measuring this effect, and until a photoperiod x dday approach is validated under field conditions, the practical application of this information is limited.

The effect of moisture on pea development has been studied only indirectly--i.e., via its effect on yield, quality and growth. Smittle and Bradley (28) showed pea yields in Arkansas were increased by maintaining available soil moisture above 50%. A maintenance figure of 60% or above is suggested by other workers (27). In general, experimental evidence indicates that the effect of moisture on both yield and quality of peas is greatest during the period from flowering to maturity (17, 33). Maurer and coworkers (27), in comparing effects of varying soil water regimes, found a pre-blossom stress followed by adequate water to be optimum as far as both development and yield were concerned.

The ability to control many other variables--e.g. fertility, moisture, plant population and spacing--is one reason that temperature has attained its position as the parameter for prediction of pea development. Limited information on the relationship between development and environment is another. The question arises, then, as to the accuracy of maturity predictions based solely on temperature.

In field use of the heat unit system for peas, a range of + or - two days may occur in predicted maturity date as a result of non-temperature variables (37). This accuracy suggests that overall development of peas follows a close relationship to temperature.

A further advantage of this parameter is the availability of historical data. Temperature is almost the only climatic variable for which long-term records (i.e., minimum of ten years) exist, and these records are a primary tool of cannery operators.

A canner constructs a curve of normal monthly or seasonal temperatures for the area in which crops are grown. Planting operations are scheduled using these curves and current data. Temperatures used to calculate the normal curve are generally obtained from U.S. Weather Bureau statistics. A canner therefore has to choose data from the station nearest his fields. It is not uncommon to find temperature data in use which was recorded 15 to 25 miles from the field for which predictions are desired (16). Obviously, the variation in temperature over such a distance may be considerable.

In addition, the variation of temperature with height may exceed variation due to distance (25). Air temperatures at ground level and/or plant height often differ significantly from temperatures higher off the ground. (Temperatures used in calculations of heat units are usually based on measurements of air temperature in a

Stevenson screen supported 4 1/2 to 6 feet above the ground (20). Field topography, and objects such as trees, houses or hills surrounding the thermometer may contribute further variation to temperature measurements.

In other words, there is likely to be appreciable difference between temperature of the plant tissue in question or of the immediate plant environment and the temperature on which heat unit accumulations are based. However, with a number of years' experience in the same fields, using temperature data from one location to calculate summation constants, a consistent relationship can be established between temperature and plant response.

Some canners use soil as well as air temperatures for maturity prediction (37). Since both aeration and moisture modify plant growth, critical temperatures may differ according to soil type (46). Nevertheless, because soil temperature lags behind air temperature in the spring, ddays based on soil temperature may be more realistic, especially during planting season. In heat unit work with sweet corn, Arnold (5) found that soil temperature gave greater accuracy than air temperature when predicting plant development from planting to the fourth leaf stage. He suggests that air temperature becomes more important than soil temperature only when seed food reserves are depleted and the plant is completely dependent on its above-ground portion as a source of carbohydrates--i.e., at fourth leaf stage in the case of sweet corn. Haupt (19) reports that the development of peas up to flower initiation (and occasionally to anthesis) may take place at the expense of cotyledon reserves. This might therefore be a reasonable stage of development at which to shift from soil to air temperatures in calculating heat units for peas.

There may be some difficulty in obtaining representative long-term soil temperatures. Drainage, soil type, and depth of measurement will affect temperatures, as will ground cover. However, the literature provides no basis for adjustments relating field conditions to off-site data.

The heat unit system is used commercially to schedule plantings and harvests to prevent "bunching" and "breaks" in the flow of raw product to the cannery. Planting normally begins in spring as soon as the soil (or mean air) temperature reaches $40^{\circ}F$ and/or when the risk of loss to freeze declines to an acceptable level. Seeds sown earlier than this do not germinate immediately and such plantings usually result in bunching at harvest. Plantings are spaced at intervals equal to the normal heat unit accumulation per day at harvest (more rapid heat unit accumulation during harvest-time than during planting means the time interval between plantings must be greater than the interval between harvests). Daily cannery capacity determines the number of acres planted each day.

Since the heat unit accumulation intervals are derived from "normal" temperature curves in a given area, an even flow of peas into the cannery at harvest will occur only in the case of normal temperatures. Therefore, some variation in harvesting rate almost always occurs. The size of the first planting may be doubled or tripled to provide a backlog on which to work (12, 37). Once harvest is underway, however, a canner may or may not be able to avoid bunching and/or slack (i.e., by postponing harvest or harvesting at lower tenderometer readings).

Most commercial pea cultivars require approximately 420 ddays from the time they reach 80% bloom until they reach optimum maturity for harvest (37). Figures estimating the average number of ddays accumulated each day during the harvest period in a given locality are available--usually between 30 and 43 ddays/day (37). This information provides an opportunity to correct estimated harvest forecasts two to three weeks before actual harvest. Naturally, the accuracy of any yield forecast increases the later in the season it is made. Another common "check-point" occurs 350 heat units after 80% bloom (two to six days before harvest)--at this time fieldmen take samples for sieve size and tenderometer readings (37).

Attempts to improve the heat unit system are based primarily on variations in heat unit derivation. One approach establishes an upper limiting or "cut-off" temperature: all temperatures above a selected maximum are equated to the maximum. Arnold (5) points out that this approach is useful only if temperatures above the cut-off occur "frequently enough" to significantly affect the results--otherwise the cut-off is ineffectual. Hope's work (22) indicates that temperatures above 60°F affect the accuracy of the heat unit sum for peas and that 60° might therefore be a reasonable cut-off temperature. The same work, however, shows that modifying the maximum temperature is effective only when heat units are determined at the most satisfactory base temperature (36°F in Hope's study). Earlier work by Arnold (3) also emphasizes the importance of a correct base temperature in this regard; decreased variability in heat unit summations achieved by applying a cut-off temperature may simply counterbalance error introduced by a too high base temperature.

The heat unit modification employed by Gilmore and Rogers (15) for corn is a variation on the above approach. Before computing ddays, they subtract from the daily mean the number of degrees by which the daily maximum temperature exceeds an assumed optimum. Hope (22) found no increase in accuracy when applying this method to peas and Arnold (3) demonstrates graphically that the decrease in variability which Gilmore and Rogers obtained was due solely to the counterbalancing of a high base temperature.

Another variation of the heat unit system substitutes daily maximum temperatures for daily means. Hope (22), in a comparison with both imposition of temperature limits and the adjustment of Gilmore and Rogers, found the least variability in heat summations using this method. He suggests this as a short-cut which might be of value to the cannery operator.

Various workers have experimented with an "exponential index" which assumes that plant growth rates double with each increase of 18°F according to the rule of Arrhenius. This index favors various growing temperatures depending on their position relative to some threshold temperature below which growth does not occur. For example, if a plant grows at a rate of "one" at 40°F, it grows twice as fast at 58° and four times as fast at 76°. A maximum temperature, above which growth is assumed not to occur, may be used with this index but as Katz (25) suggests, errors in growth prediction still occur since temperatures above the optimal for good growth are ascribed unwarranted significance. The major problem with this approach is its failure to differentiate between growth and development.

The heat unit system employed by the canning industry is not theoretically faultless: the approach neglects factors other than temperature which influence plant development and difficulties arise in identifying threshold temperatures during different developmental stages. The system's popularity continues to be widespread, however, for the simple reason that no other system has been found which can adequately replace it. As used, the system contributes to a more orderly flow of raw materials of optimum maturity to the cannery. As Seaton (37) puts it, the system is "a yardstick--not a micrometer--to be used along with other tools such as the microscope and the tenderometer."

Temperature Effects on Pea Development and Yield

That plants respond differently to the same environmental factor during various stages of their life cycle is well-known. Accordingly, temperature optima and sensitivities of peas vary with developmental stage. Wang (46), on the basis of a compilation from various authors, lists 40° to $75^{\circ}F$ as an optimal range for seed germination. During vegetative and reproductive growth, temperatures in the range of 68° to $70^{\circ}F$ appear best (14,22) with injury occurring near $85^{\circ}F$ (14,16,22). Although the heat unit system assumes that day and night temperatures are of equal importance to plant growth, growth chamber studies by Karr, Linck and Swanson (24) show relatively well-defined "thermal sensitive periods": maximal sensitivity to high day temperatures occurred 9 to 11 days after full bloom, while maximal sensitivity to high night temperatures occurred 6 to 9 days after full bloom.

Field research, however, emphasizes more general trends in thermal response. It is known that the higher the mean temperature, the less time required for a pea plant to reach a given state of maturity (9,29). Thus, late plantings require less time from planting to emergence and from planting to bloom than do earlier plantings (13,16). Daylength may play a role in this phenomenon--i.e., long days reduce heat requirements. Early plantings, however, require fewer heat units to a given stage of maturity than do later plantings (13,16,25,38). It is possible that the higher daily mean temperatures which occur as a season progresses reflect a higher frequency of temperatures above the optimal for growth. This would also explain the finding that fewer heat units are required to mature a crop in cool than in warm seasons and/or years (25).

Apparently, then, heat units are not physiologically equivalent between years or seasons. As used, however, the dday system ascribes equal significance to heat units regardless of when during the season they accumulate. Thus it is impossible, using this system, to distinguish between a warm spring followed by a cool summer and a cool spring followed by a warm summer. As Wang (45) points out, the former conditions are associated with high yields, the latter with poor ones. The literature contains no comparison of development, distinct from yield, when a warm spring is followed by a cool summer or a cool spring is followed by a warm summer.

The yield-temperature relationship for peas has, in fact, been extensively studied. Boswell's observation (9) that the higher the mean temperature during growth, the lower plant weight, pod number and weight, and peas per plant has been confirmed by other workers (29, 38). Evidence indicates that high temperature decreases yields primarily by limiting plant size, thereby decreasing bearing area and number of blossoms (9, 14, 29, 38). According to Boswell, the deleterious effect of high temperature on yield may occur at any point during the life of the plant, although he found the closest inverse relationship between yield and temperature in the period between blossoming and harvest. Karr, Linck and Swanson (24) found high night temperatures more critical than high day temperatures in reducing pea yields; they observed maximum reductions of 25 versus 8% respectively. Their work indicates a roughly additive effect for the combination of high day and night temperatures.

Thus, late plantings of peas, which generally develop during periods of higher mean temperatures than earlier plantings, are associated with lower yields (8, 14, 28, 29, 38). Boswell (8) noted a yield reduction up to 50% when plantings were delayed by as little as a week or ten days; Smittle and Bradley (38) recorded a 67% reduction in plantings delayed four weeks.

Gritton and Ebert (16) suggest disease as an additional cause of lowered pea yields under high temperature conditions. They point out that until recently almost all of the pea cultivars grown for canning were susceptible to powdery mildew and that early plantings are less likely to be infected than later ones (ostensibly because the higher temperature of late season favors the causal organism). Spraying pea fields for mildew stabilized yields in late plantings of Alsweet peas even with "a number of days" in which the temperature reached a level "at which injury has been known to occur" in peas. That the same treatment failed to prevent yield reductions in the cultivar Perfected Freezer indicates that disease is not totally responsible for yield losses under high temperature conditions.

Yield/Quality Interaction in Peas

The study of yield in peas is complicated by two factors. In the first place, the value of growth analysis is limited: it is known that the total dry weight of peas increases rapidly after flowering, doubling in less than a week at its maximum rate, but the sources of materials for pod growth are unknown (18). In addition, pod growth rate is not correlated with leaf area index (18, 28). Essentially, the relationship between vegetative growth and horticultural yield has not been established, and potential yield can therefore not be predicted by standard growth analysis techniques.

The second complication is that pea maturity cannot reasonably be omitted from yield considerations since in this crop it is as important commercially and there is a large maturity x yield interaction. (In a typical contract, the price received by a grower may be adjusted by 20% according to maturity of the peas delivered to the cannery (18)). Because peas are harvested at an immature stage, a conflict arises between quality and yield requirements; a meaningful evaluation of yield therefore includes both weight/acre and the tenderometer (TR) grade at which the peas are harvested.

No universal curve relating yield and maturity exists. The relationship varies between fields, cultivars and seeding rates, and as a result of environmental factors such as wind, humidity and available soil moisture. It is therefore impossible to write a yield equation of the form "X pounds of peas per unit change in TR value." In general, however, within the limits of TR grades acceptable for processing, the yield of any cultivar increases as the peas mature (35).

This inverse relationship between yield and tenderness is apparently not linear (33). Hagedorn et al.'s work (17) to the contrary, the preponderance of evidence supports a curvilinear yield-tenderometer relationship in which the increase of yield per unit increase in TR declines with increasing TR value, particularly at high TR values. Pollard and coworkers (32), for example, found a greater increase in yield per unit TR below a TR of 102; Pumphrey, Ramig and Allmaras (33) cite diminishing increases above the range of 100 to 120 TR.

The suggestion is often made that a means of comparing pea yields at a common TR would be useful. Unfortunately, there is little published information in this regard. Norton, Bratz and Russell (30) compared yields of Dark-skinned Perfection peas over a 15 year period by converting yield to 100 TR--that is, at TR 100 yield was considered 100%. The authors assumed that other cultivars of the perfection type (i.e., large, wrinkle-seeded, "late" peas) would show a similar yield-tenderometer relationship. They discovered, however, that the conversion formula for many such cultivars differed significantly from the formula for Dark-skinned Perfection. Nevertheless, according to the authors, use of the yield-tenderometer chart developed for Dark-skinned Perfection is more reliable than a conversion formula based on relatively few yield-tenderometer combinations.

Pumphrey, Ramig and Allmaras (33) interpolate yields of Dark-skinned Perfection at 100 TR, then use the ratio of measured to interpolated yield to obtain "percent yield." From a scattergram of percent yield versus TR they calculate a least squares model: $Y = a + bX + cX^{\frac{1}{2}}$; where Y is percent yield, X is TR value, and a, b and c are statistically estimated parameters. The authors suggest that the scattergram alone provides more reliable data for prediction than do unadjusted yields. Use of some such technique seems justified since these data indicate percent yields change one to two units with each unit change in TR value. Hagedorn, Holm and Torrie (17) report "average increases in yield (pounds/acre) for each unit increase in tenderometer" for both Alaska and perfection types, but use of these figures is limited since the range for both types was approximately 30 pounds/acre.

Quality Measurement in Peas

The Food and Drug Administration of the United States Department of Agriculture establishes criteria for quality evaluation of canned peas. Quality of liquor, color, extent of defects and overall tenderness and maturity are used to differentiate between four grades: "Fancy" or grade A; "Extra-Standard" or grade B; "Standard" or grade C; and "Substandard." The grade of a given lot of canned product is determined by sampling; it cannot be precisely determined from raw product tests since a large number of variables are introduced during the canning process itself. However, because canners have different demands for each grade they do attempt approximate quality predictions for their fields.

Although quality predictions have in the past been based on such things as refractive index, freezing point and electrical conductivity of cell sap, and starch/sugar ratios, the industry currently relies on four other techniques. Three of the four are used for the raw as well as canned product, the remaining test is applied only to canned peas. When tests are run on field samples they are an important element in harvest scheduling: harvest dates are adjusted whenever possible to provide the needed quantity of a particular grade. Samples are taken at various "check-points" during the season, with test frequency increasing as harvest approaches (13,37).

The most important tool for measuring pea quality is the tenderometer, an instrument which measures the force necessary to shear peas between two grids. Resistance to shear, measured in pounds of pressure, is assumed to resemble the toughness experienced during chewing. As peas mature dry matter increases, the protein/non-protein nitrogen and starch/sugar ratios increase, and calcium migrates to the seed coat (skin) (6,36). These and other biochemical and morphological changes result in decreased tenderness and therefore higher TR readings (32,36).

Peas ranging from 80 to 160 TR may be canned with 100 representing an approximate optimum; close to 95% of peas harvested at 100 TR will pack "Fancy" and yield increases of 5 and 17% can be realized over peas harvested at TR 90 and 80 respectively (26). Both Ells (13) and Seaton (37) report that the increase in TR value per day is greater after a TR value of 100 is reached than before (12 versus 5 points per day in Ells' data). Reported daily increases in TR value at harvest stage vary from 5 to 30 points depending on cultivar and season (13,25,35). In general, an essentially linear relationship exists between heat unit accumulation and TR value (25,37), and the number of heat units accumulated during the harvest period increases with successive plantings. (In Colorado the difference between first and third planting was 13 heat units per day (13)).

Size, although not a reliable index of quality when used alone (36), is used in conjunction with TR reading for Alaska type cultivars. "Sieve size" is determined by the diameter of circular openings through which peas will pass without force or pressure. In general, the smaller the diameter of peas of a given cultivar, the lower the TR value and therefore the higher the quality. There are, however, considerable differences between cultivars in the size-tenderness relationship (9, 35), and this test can therefore not be used for cultivar comparisons.

Sieve size, like TR value, appears to increase more rapidly in later plantings than early ones (13). Although the majority of size increase takes place before TR 100 (13), increase in TR value during the harvest period is greater for larger sieve sizes than for smaller ones (17).

In brine flotation tests peas are placed in salt solutions ranging from 11 to 15%. To qualify as "Fancy," the number of peas which "sink" in an 11% solution must be lower than 12%. Each grade has its own maximum percentage of "sinkers" allowable for a given brine concentration. This test, based on specific gravity (with a low specific gravity signifying high quality), must be used in conjunction with other tests since error may be introduced either by air bubbles on the surface of peas or absorption of liquid by peas during measurement.

A low percentage of alcohol insoluble solids (AIS) is associated with more tender, less mature peas. Since water content of peas at harvest affects the AIS value, this test is not used on raw product. It is employed for the most mature canned peas--i.e., those which grade substandard--to determine whether they meet even the minimum Standards of Quality for Canned Peas as formulated by the FDA.

Lower quality has traditionally been associated with late plantings. This is not due to any inherent chemical difference between early and late sown peas — the lower quality product often secured from late plantings is primarily the result of such a rapid rate of maturing under high temperatures that it is difficult to harvest at the optimum stage (9,13).

The increased maturation rate of plantings which develop during periods of high temperature can be dramatic. Nonnecke, Adepipe and Ormrod (29) found a reduction of 23 days for late plantings of three different cultivars to reach maturity, and Ells (13) found daily increases in TR values greater by 10 to 20% for third versus first plantings. Katz' data (25) indicate the practical significance of timely harvests in this regard; the number of days required to raise the TR value from 100 to 120 for both Alaskas and sweets was only one half to one and one half days.

In general, then, late plantings (which usually develop under high temperatures) should be avoided both because of the harmful effects of high temperature on yield mentioned earlier and because of the increased difficulty of a timely harvest under high temperature conditions.

PROGRAM DESCRIPTION

The primary objective of program PSIM is creation of planting and harvesting schedules for canning peas. The simulation mimics standard industry practice in developing the schedule, then follows the crop through a growing season in order to predict harvest dates and yields. The basic structure of the simulation is summarized then followed by a detailed description.

Overview of Program PSIM

The program will perform the following operations:

Plant the total acres required for each cultivar:

---Plant the cultivars in order of ascending maturity constants.

---Begin planting on the earliest allowable date.

---Stop planting when the predicted (historical) harvest date for a given planting exceeds the harvest deadline.

Predict the harvest date for each planting based on maturity constants and historical temperature data.

Divide each planting into subplantings and determine for each subplanting:

---Ddays to 100 TR (based on a random sample of a normal distribution whose mean and standard deviation are user-supplied for each cultivar).

---Harvest date, assuming harvest at 100 TR, using ddays to maturity as calculated in the last step, and based on current year temperature data.

---The average number of ddays accumulated/day for five days preceding and five days following the date on which the subplanting reaches 100 TR.

---The date on which the subplanting reaches 80 TR.

---The date on which the subplanting reaches 140 TR.

---Yield at harvest, assuming harvest at 100 TR (based on a random sample of a normal distribution whose mean and standard deviation are user-supplied for each cultivar).

Sum yields for each day of harvest.

Starting with the first day of harvest, check all harvest days to see if total yield exceeds maximum cannery capacity. For each day on which maximum cannery capacity is exceeded, reschedule harvests where possible using the following logic:

---Reschedule the harvest of one or more subplantings to the date(s) on which the subplanting(s) reaches 80 TR. Do this only if the rescheduling does not cause maximum cannery capacity to be exceeded on another day (i.e., by the addition of this subplanting's yield at 80 TR). Reschedule as many subplantings as possible to reduce this day's total yield to no more than maximum cannery capacity. Reassign harvest dates and yields and record the lower TR reading for affected subplantings.

---If maximum cannery capacity is still exceeded after the foregoing procedure, check to see if any subplantings can be rescheduled for harvest at 140 TR within the same yield constraints. Reassign harvest dates and yields and record the higher TR readings where relevant.

Review yields on each harvest day. Record any dates on which yield falls below the minimum or exceeds the maximum cannery capacity.

Print simulation results and repeat user input.

****

PSIM is a FORTRAN simulation which uses temperature summations to schedule plantings and predict harvests for canning peas. The simulation will schedule a maximum of 30 plantings for up to ten cultivars. Each planting is divided into a maximum of 20 subplantings for which maturity date and yield are stochastically determined.

A temperature summation, in the form of the dday or heat unit system, is the parameter currently used by industry to schedule plantings. Predictions of harvest date are based initially on a "normal" year for the growing area and are then updated as a comparison is made between normal and current year temperatures. Program PSIM therefore requires two sets of temperature data:

an historical set and a current year set. In both cases, maximum and minimum daily temperatures are required for the period March 1 through September 30. (Mean daily temperature--i.e., maximum + minimum/2, ddays/day and accumulated ddays are calculated by the program for both sets of data. In the case of historical data spanning 25 years, the program calculates 25 mean temperatures for each day, sums them and divides by 25 to get the historical mean temperature for a given day). Users may supply air or soil temperatures or a combination of the two.

The simulation begins with a planting made either on a user-supplied date or on the first day of the current year when mean temperature is greater than or equal to the crop's base temperature. Cultivars are planted in order of ascending maturity constants. If two cultivars have identical maturity constants, the first cultivar entered by the user is the first planted.

Size of the planting depends on several things. The user may specify the total acre requirement and/or the total desired yield for each cultivar. If only desired yield is input, the required acreage is calculated:

$$\text{total acres required for cultivar X} = \frac{\text{total desired yield for cultivar X}}{\text{expected mean yield/A for cultivar X}}$$

The user must then specify the daily cannery capacity[1] and "acreage divisor" for each cultivar (default value for the latter is the cultivar's expected yield/A). With this information, a standard planting size (SPS) is calculated for each cultivar:

$$\text{SPS for cultivar X} = \frac{\text{daily cannery capacity}}{\text{acreage divisor for cultivar X}}$$

The objective here is to plant a number of acres on each planting day, the yield from which will fully use cannery capacity without exceeding it.

Although the size of the first planting is some multiple of SPS for the first cultivar (in accordance with standard industry practice), succeeding plantings of this and all other cultivars are SPS until the acre requirement of the cultivar being sown falls below SPS. At that point, two or more "partial plantings" are made. One partial planting completes the acre requirements of the cultivar being sown. Additional partial plantings (i.e., plantings of the next cultivar(s) to be scheduled smaller than the SPS of any cultivar in the simulation) are made in order to meet cannery capacity on the harvest date of

---

[1] The user is asked to supply three figures for cannery capacity. The figure supplied early in the simulation is used in the calculation described here. Later in the simulation, the user is asked to supply a maximum and minimum cannery capacity. The idea is to allow for a variation in number and length of working shifts at the cannery. If a canner prefers 12 hours of operation daily but can accommodate 8, 16 or 24 hours of operation, his minimum and maximum cannery capacity will vary accordingly.

the planting of the initial cultivar sown. In some instances, a small planting of only one additional cultivar is required, but plantings of more than one cultivar are made if the total acre requirement of a cultivar does not make up the difference between cannery capacity and expected yield. In all such cases, the size of partial plantings is determined by subtracting expected yield of the initial planting from daily cannery capacity and essentially re-calculating a SPS for the next cultivar or cultivars.

All SPS plantings are scheduled by the following method. Historical temperature data is searched for the historical harvest date of the most recently planted cultivar--i.e., the first date on which accumulated ddays are greater than or equal to that cultivar's maturity constant. The number of ddays accumulated on that harvest day are recorded and used to "space" the next planting. For example, the first planting of a cultivar whose maturity constant is 1350 ddays is made on March 1 and historical data show that 1352 ddays accumulate between March 1 and June 24. June 24 is therefore the historical harvest date for this planting. Historical data also show that 32 ddays accumulate on June 24. Program PSIM will schedule the next planting as soon after March 1 of the current year as 32 ddays accumulate. The scheduling of plantings is therefore directly connected to determination of historical harvest dates.

Scheduling of partial plantings differs from the above method since the desired harvest date is fixed--i.e., partial plantings made to fill cannery capacity must be harvested on the same day. Therefore, once historical harvest date is determined for the initial partial planting, plantings made to augment yield on that day are scheduled by subtracting the maturity constant(s) of a succeeding cultivar(s) from the accumulated ddays on harvest day of the initial planting. (See Results and Discussion for the problems inherent in this system of scheduling).

The planting schedule and prediction of historical harvest dates for each planting are accomplished in the first portion of the simulation. Each harvest date is compared to a user-supplied "harvest deadline"--i.e., the last day on which the user desires raw product. If the harvest deadline is exceeded, a message to that effect is printed along with the number of days by which the deadline is exceeded. Planting schedule and historical harvests are the first tabular output of the simulation.

The next portion of the simulation stochastically determines harvest date based on current year temperature data and yield. It is at this point in the simulation that use is made of a "field number" (user-supplied). The field number is used to divide each planting into subplantings, for each of which a current year harvest date and yield are determined. The creation of subplantings increases the number of random samples for each planting of both the maturity and yield distributions. Both distributions are assumed to be normal, with user-supplied means and standard deviations for each cultivar in the simulation.

The maturity constant for any cultivar is a mean with some variance in the ddays to 100 TR between fields planted on the same day. The same variance affects yield. By increasing the number of random samples of these two characteristics, it is assumed that final maturity dates and yields will more nearly approximate reality. Theoretically, the larger the field number, the greater the likelihood of realistic predictions. The maximum number of subplantings (i.e., the largest field number) allowed in program PSIM is 20. If five plantings are made of one cultivar and the field number is 20, the simulation outputs yield and maturity data for 100 samples of that cultivar.

The following information is calculated for each subplanting: subplant size, ddays to 100 TR, yield, the dates within the current year when the subplanting reaches 80 TR and 140 TR, the average number of ddays/day for five days preceding and five days following the date of 100 TR, and current year harvest date. Following is an explanation of the determination and use of each of the above characteristics.

Subplant size is simply the quotient of planting size divided by field number.

In this simulation 80, 100 and 140 TR are considered the critical maturity stages for peas, with 80 TR the earliest harvest stage, 100 TR an optimum (considering both quality and yield), and 140 TR the latest harvest stage. Since a maturity constant is the mean number of ddays to 100 TR for a given cultivar in a specific area, if a normal distribution of dday requirements is assumed and a standard deviation is known, then random samples of that distribution can provide a realistic picture of the actual variation in dday requirements. This is the approach used in program PSIM. The actual ddays to 100 TR for each subplant are the result of sampling such a distribution and applying the formula:

$$\text{ddays to 100 TR for subplant X} = \text{maturity constant for cultivar X} + \left(\text{random number} \times \text{standard deviation of maturity constant for cultivar X}\right)$$

(Program PSIM includes a pseudo-random number generator. The user is asked to supply two seeds for the generator for each run of the simulation).

An identical approach is used to determine the yield of each subplant. Taking a user-supplied expected yield/A (mean) and standard deviation for a cultivar:

$$\text{yield/A of subplant X} = \text{expected yield/A cultivar X} + \left(\text{random number} \times \text{standard deviation of expected yield/A cultivar X}\right)$$

The yield calculated in this step may or may not be the yield of this subplant in the final output of the simulation. Yield may, in fact, be re-calculated if the total yield on the date of 100 TR for this subplant exceeds maximum cannery capacity on that date. This is done as follows. After all subplantings have been assigned planting dates, and yields according to the above formula, yields are summed for all harvest dates. If cannery capacity is exceeded on a given harvest day, an attempt is made to reschedule harvests one of two ways.

First, if enough of the subplantings which reach 100 TR on this date can be harvested at 80 TR to bring yield on this date to no more than maximum cannery capacity (without exceeding maximum cannery capacity on another day), harvest for the required number of subplantings is rescheduled for 80 TR. If maximum cannery capacity is still exceeded after the foregoing procedure, a second rescheduling attempt is made. A check is made to see if any of the subplantings falling on this date can be rescheduled for harvest at 140 TR (again, without exceeding maximum cannery capacity on another day). If so, the required number of subplantings is rescheduled for harvest at 140 TR.

Such rescheduling, of course, affects yield. Since yield and TR reading are inversely related, harvesting at 80 TR will reduce yield, while harvest at 140 TR will increase it. The percentage by which yield of a subplant is reduced when harvest is at 80 TR, or increased at 140 TR, may be supplied by the user. Program PSIM assumes yield at 100 TR to be 100% and provides default values of 56% for yield at 80 TR, 136% at 140 TR (see Results and Discussion for rationale).

The dates on which a subplanting reaches 80 and 140 TR are also calculated by program PSIM. Users are asked to input one of two pieces of information to make this determination possible: the increase in TR points/day for each cultivar between the 80 and 100 TR stages and between the 100 and 140 TR stages, or, the number of ddays corresponding to an increase in one TR point for the same time periods (from 80 to 100 TR, from 100 to 140 TR) for each cultivar. If the increase in TR points/day is input, dates of 80 TR and 140 TR are calculated by the formulae:

$$\begin{array}{l}\text{date of 80 TR}\\\text{for subplant X}\end{array} = \begin{array}{l}\text{date of 100 TR}\\\text{for subplant X}\end{array} - \left(\frac{\text{20 TR points}}{\begin{array}{l}\text{TR points/day between}\\\text{80 and 100 TR}\end{array}}\right)$$

$$\begin{array}{l}\text{date of 140 TR}\\\text{for subplant X}\end{array} = \begin{array}{l}\text{date of 100 TR}\\\text{for subplant x}\end{array} + \left(\frac{\text{40 TR points}}{\begin{array}{l}\text{TR points/day between}\\\text{100 and 140 TR}\end{array}}\right)$$

If conversion factors for ddays/TR point are input, the program first calculates the increase in TR points/day:

$$\text{increase in TR points/day between 80 and 100 TR for subplant X} = \frac{\text{average number of ddays/day for five days preceding date of 100 TR of subplant X}}{\text{ddays/increase of one TR point between 80 and 100 TR for cultivar X}}$$

$$\text{increase in TR points/day between 100 and 140 TR for subplant X} = \frac{\text{average number of ddays/day for five days following date of 100 TR of subplant X}}{\text{ddays/increase of one TR point between 100 and 140 TR for cultivar X}}$$

The program then determines dates of 80 and 140 TR as shown above. In this case, the program uses the values it has calculated for the average number of ddays/day preceding or following the date of 100 TR.

The "average number of ddays/day for five days preceding and following the date of 100 TR" for each subplant are literal averages calculated from current year temperature data supplied by the user. There are two situations, however, in which the "averages" are for one day's data instead of for five day's. If 100 TR for a subplant falls on March 1 through March 5, the value stored as the average number of ddays/day preceding 100 TR is the number of ddays on the date of 100 TR. Likewise, if the date of 100 TR occurs after September 25, the value stored as the average ddays/day following 100 TR is the number of ddays on the date of 100 TR.

The harvest date of each subplanting is, barring any re-scheduling to accommodate cannery capacity, the date in the current year on which that subplant reaches 100 TR. As detailed in the explanation of yield values, actual harvest date may be moved either to the date of 80 TR or the date of 140 TR. Simulation output lists the TR reading at harvest along with the actual harvest date so a user knows when rescheduling has occurred.

Output from program PSIM includes, in addition to the above information for each subplant, the total number of pounds harvested for the season, with subtotals for yield at 80, 100 and 140 TR. It also lists any waste (i.e., the number of pounds by which, because harvests could not be rescheduled, maximum cannery capacity was exceeded) or slack (the number of pounds by which yield fell below minimum cannery capacity) both as seasonal totals and by date. Date of first and last harvest is also printed, and user input is repeated.

The foregoing pages are intended only as a description of the program. Assumptions made and their rationale, along with a discussion of model strengths and weaknesses, may be found under Results and Discussion. Sample program output, with legend, and a description of data used to validate the model are included in Appendix C. Figure 1 provides a very generalized model flowchart; Appendix B contains detailed flowcharts. Subroutine interrelationships and descriptions are in Appendix E. Appendix F contains a program listing and variable dictionary. Control cards for the simulation are in Appendix D, along with documentation.

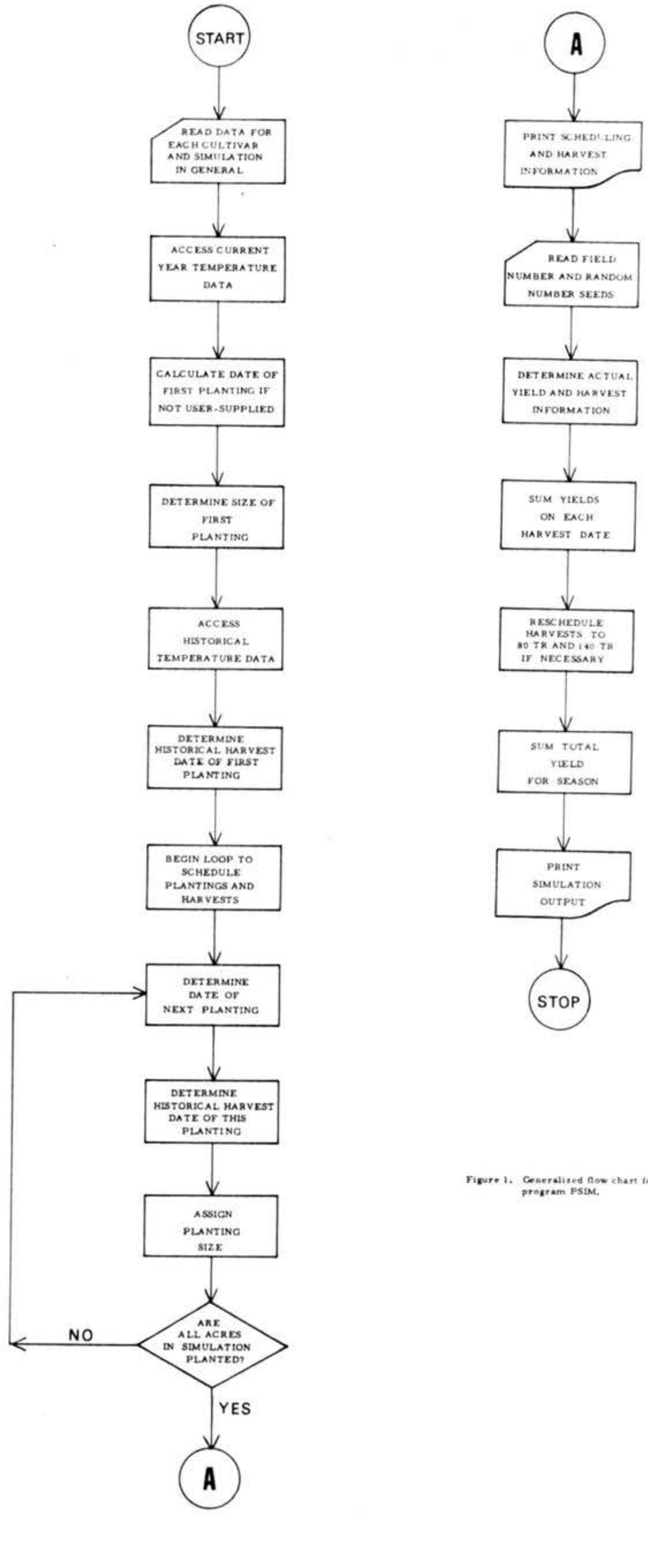


Figure 1. Generalized flow chart for program PSIM.

RESULTS AND DISCUSSION

Simulation is usually employed for non-deterministic systems in which the primary need is to test decision rules which govern system behavior. The scheduling of canning peas by the dday method fits this criterion well. Crop maturation is both dynamic and stochastic in nature. The inclusion of temperature and genetic variation makes it impossible to arrive at a single optimum combination of the inputs involved. Moreover, there is no way to formulate constraints or performance criteria as required by analytic techniques. It is profitable, however, to ask questions in the form, "What would happen if......?" --i.e., a change occurred in system input. In addition, because the heat unit system is already widely used by industry, a realistic context for the modelling effort exists.

The choice of FORTRAN over one of the simulation languages was based on three factors. The first is the universality of the language, which confers an immediate advantage in terms of program transportability. The second deciding factor was the extent to which choice of language affects model conception. The simulation languages investigated seemed to impose structures (e.g. compartmental flow) which would have necessitated an unnatural degree of contrivance in this particular case. And finally, more general familiarity with FORTRAN made it a reasonable choice.

In its present form, the model has limitations caused by assumptions made and, in some instances, scarcity of data. These are presented in detail in order that any user may be conscious of the model's limitations, intentions, and scope.

Program PSIM uses temperature as it is currently used by industry to schedule plantings and make maturity predictions. The method used to calculate mean temperature (i.e., using only maximum and minimum for each day) may be criticized for grossness: a more accurate method would undoubtedly incorporate some measure of temperature duration (4). The difficulty lies in the availability of data. Since many canners use United States Weather Bureau figures, use of more precise data, even if available to the modeller, would make the program difficult or impossible to use in many "real life" instances. Evidence that other modifications substantially increase the accuracy of the dday system is not convincing.

There is, on the other hand, nothing about the temperature calculations within the model which favors use of either soil or air temperatures. A combination of the two could easily be employed. Correspondence with several commercial canners[1] indicates a wide variation in the length of time for which soil temperatures are used.

---

[1] The following companies responded to the questionnaire in Appendix G: California Canners and Growers; Baker Canning Company; Fort Lupton Canning Company; Green Giant Company; Libby, McNeill and Libby; Stokely Van Camp, Inc.

One possible use of the simulation is a comparison of the accuracy of harvest predictions based on the two temperature sources.

The fact that the simulation requires a single year of current temperature data may seem to create an unrealistic situation, since a growing season must be complete before the simulation can be run. A user might well choose a year initially which seems to resemble the season upon which he is embarking. As the season progresses, temperatures from the actual year can be inserted into the data, so that predictions will become more realistic as time passes. This is, in fact, the best that can be done since temperature will never be absolutely predictable.

The model uses only seven months of temperature data, March through September. This period was chosen because the Almanac of the Canning, Freezing and Preserving Industries (2) showed no record of canning peas in the United States planted before March 1 or harvested after September 20. Although this decision seems practical in terms of reducing the amount of data required by the program, it created some cumbersome programming which, if foreseen, might have led to use of a full year's data. Specifically, a 7 by 31 array was used to store information by month and day. This means that array month one is calendar month three, and conversions between the two required care. In addition, months with 30 days had to be treated differently in many calculations than months with 31 days.

Generation of the planting schedule in PSIM follows the pattern used by industry. Several points should be mentioned, however. Cultivars are normally planted in order of ascending maturity constants. If two cultivars have identical constants (a fairly common situation), the cultivar which is entered into the computer first is planted first. This is probably unrealistic since a canner might plant two such cultivars concurrently. There are two ways a user can compensate for this characteristic. One is by dividing acre requirements for each cultivar, assigning each acreage a different name, and alternating their order of input. The other method, reasonable if the cultivars have significantly different price/cost ratios, is to input different acreage divisors, thereby reducing or increasing SPS for each cultivar.

Partial plantings may create an additional conflict in the simulation. This conflict is most likely to occur early in the season. As an example, imagine that the last planting of the first cultivar planted, with a maturity constant of 1300 ddays, is less than the SPS for that cultivar. This planting is assigned a historical harvest date of June 15, on which date a total of 1307 ddays has accumulated. A partial planting must now be made of the second cultivar, and this planting must be made at such a time that it will also have a historical harvest date of June 15. But if the maturity constant for the second cultivar is 1350 ddays (or any number greater than the number of ddays accumulated by June 15), a planting cannot be scheduled which meets the necessary harvest criterion. It seems unlikely that this situation will arise very often, if at all, in actual practice. If it occurs within the simulation, it can be corrected by increasing the acreage divisor of the first cultivar, thereby forcing a larger number of plantings which will occur over a longer period of time. Such a strategy may, of course, result in underuse of cannery capacity.

One of the decisions which the simulation allows a user to make is date of first planting. Although the program will provide the first date on which mean temperature is greater than or equal to the crop's base temperature, a user may specify a different date if he so chooses. This facility provides latitude which might be necessary for a number of reasons--e.g. a single warm day in the midst of a cold spell, an unusual amount of rain around planting time making fields too wet to enter, machinery breakdown, predicted overflow at the cannery from other crops.

The most fundamental of the simulation's limitations is the maximum it places on the number of cultivars, plantings and sub-plantings allowed in any one run: ten cultivars, 30 plantings of each cultivar, 20 subplantings of each planting. Simulations are notorious for their computer storage requirements, and PSIM is no exception. It was considered more important to allow for greater sampling of yield and maturity distributions, than to increase the number of different cultivars which could be planted at one time. If this is a serious limitation for a specific user, which is unlikely, he might partially compensate by running the simulation more than once using different cultivars each time while supplying a progressively later first planting date for successive runs. Historical harvest predictions, at least, would be the same as if all cultivars were run together.

The importance of the user-supplied field number must be stressed. The field number equals the number of subplantings for each planting, and thereby controls the number of samples taken for yield and maturity data. Particularly for cultivars with a large SPS, users are encouraged to input a large field number in order to make appropriate use of the program's stochastic facilities.

Yield and maturity, the two stochastic features of the simulation, are both assumed to follow normal distributions. This is a case of "best guess." The literature contains no specific data for peas; agricultural textbooks rarely make general statements about distributions of such complex variables as yield and maturity. It therefore seemed reasonable to rely on the ubiquitous normal distribution. The random number generator included in PSIM was available from the Colorado State University Statistical Laboratory. It provides random variates from a normal distribution whose mean is zero and whose standard deviation is one, with generated values ranging from minus infinity to infinity. When calculating yield and dday requirements, program PSIM rejects random numbers whose absolute value is greater than four. This is done to keep dday requirements from being either negative or unreasonably large; it also keeps yields from becoming "infinitely" large. Any negative yields are automatically set equal to zero. As with all pseudo-random number generators, the same series of numbers will always be generated for a given seed. This may prove useful in PSIM since it will allow a user to make sensitivity analyses of other system parameters.

Both yields and dday requirements generated in program PSIM are normally distributed. Figures 2 and 3 are based on 40 values generated in one run of the simulation. Chi-square values were not significant at the .01 level for either variable.

In this model, 80, 100 and 140 TR are used as the critical maturity stages for peas. There are few, if any, references in the literature to harvest below 80 TR. The general acceptance of 100 TR as optimum stems from the balance between quality and yield at that reading. Because canners will avoid packing substandard peas if possible, they try to harvest before a field reaches 150 TR (if a field does reach this reading and is harvested, nothing over four sieve is canned (13)). Since both the price/cost ratio and quality decrease with delay in harvest after maturity, 140 TR was chosen as the harvest cutoff.

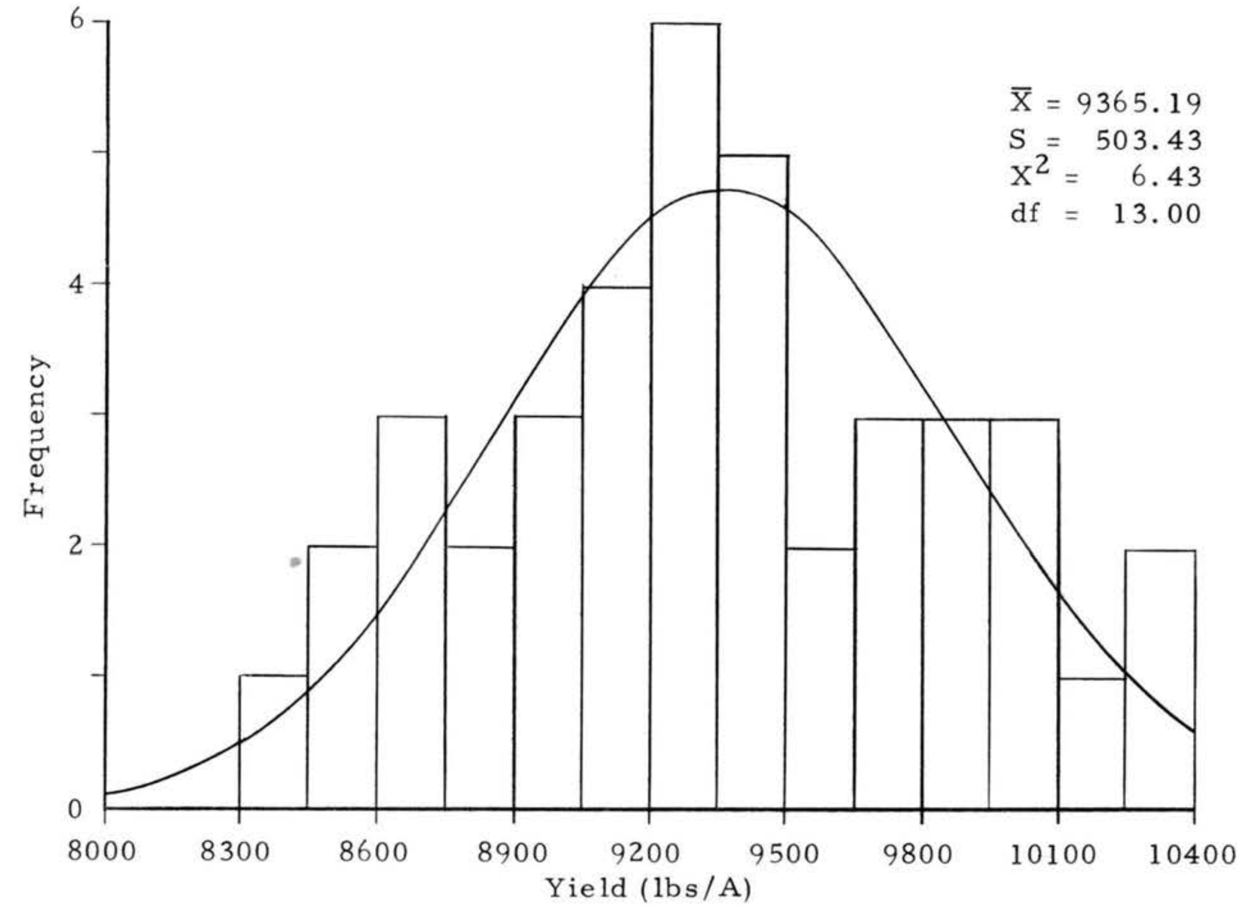


Figure 2. Histogram of yields generated by program PSIM with normal curve superimposed.

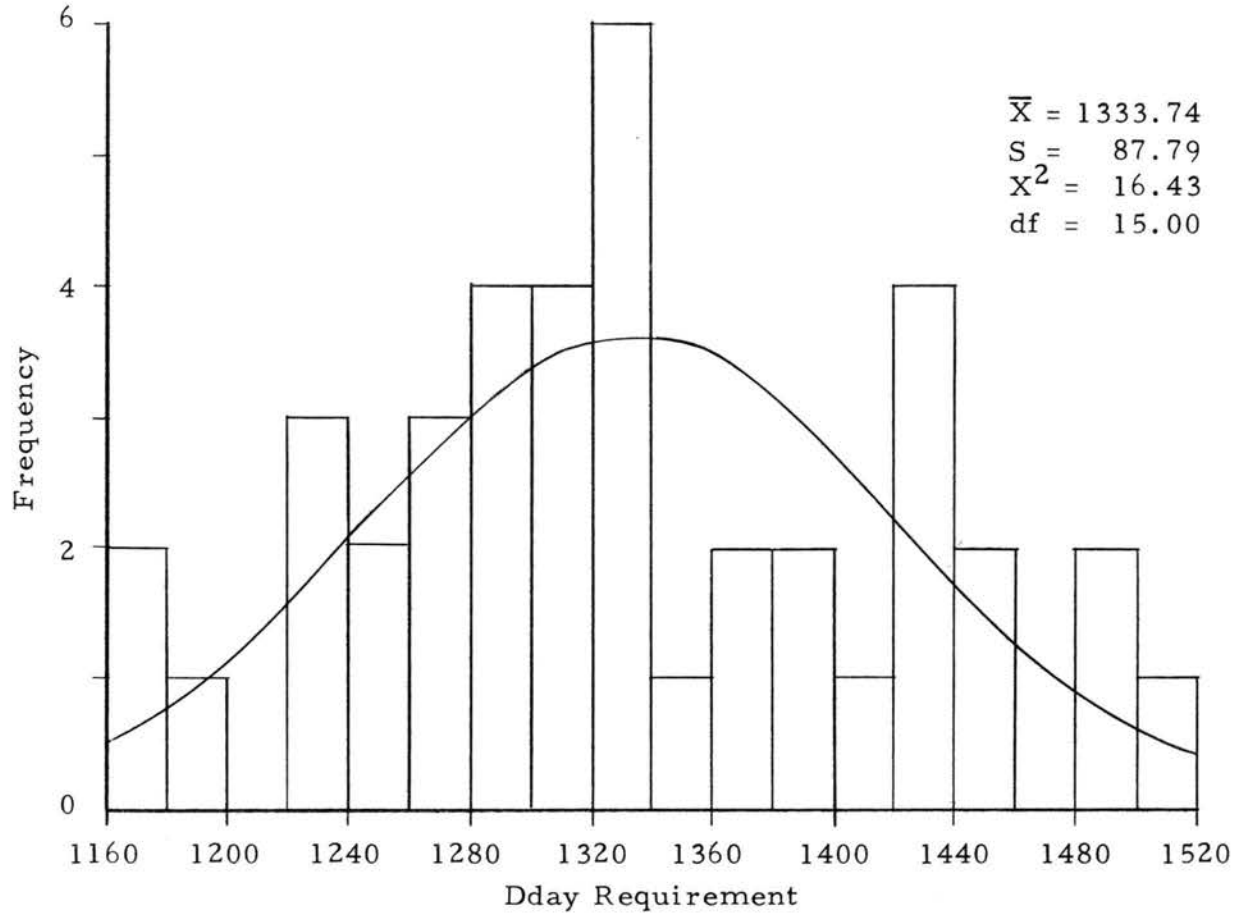


Figure 3. Histogram of dday requirements generated by program PSIM with normal curve superimposed.

The method of rescheduling yields to accommodate cannery capacity in PSIM assumes that it is more desirable to harvest at 80 than at 140 TR. In other words, a choice is made in favor of quality at the expense of quantity. One justification for this choice is that not only is product at 80 TR certain to be marketable, tenderer product brings a higher price. In addition, harvest at any TR includes a sieve size spread and while pounds of raw product harvested will be greater at 140 TR, some percentage of that product will be unprofitable to can. Another justification for this method of rescheduling harvest is bunching--i.e., yield will be reduced at 80 TR.

The relationship between yield and quality in peas is, as noted in the Literature Review, difficult to quantify. Of the few attempts made (17,30,32,33), the work of Norton and coworkers (30) was selected for use in this simulation. Although the authors' yield conversion formula is for 'Dark-Skinned Perfection' (DSP), their work indicates that use of the yield-tenderometer chart developed for DSP is more reliable for other cv's than a conversion formula based on relatively few yield-tenderometer combinations. Program PSIM therefore uses the figures of 136% yield at 140 TR and 56% yield at 80 TR, after Norton and coworkers (figures are rounded to the next highest percentage point). Because program users may have different figures for their cultivars, these values are provided as a user option. Default values were considered essential because of the extreme paucity of data in this area. The percent yield at 80 TR is an extrapolation of Norton's data, which begin at 81 TR. (The yield at 81 TR is 58.14%, yield at 82 TR is 60.64%--a difference of 2.5%. The value used in PSIM is 2.5% less than the value quoted for 81 TR. This adjustment of 2.5% for a change of 1 TR point seems justified by the few figures available in the literature (32, 33)).

There are a number of smaller assumptions made in the course of the simulation. In the subroutine which checks harvest dates against the user-supplied harvest deadline, it is inadvertently assumed that the deadline will not be exceeded by more than 61 days. More precisely, 61 is the largest number of days printed as the "number of days by which harvest deadline is exceeded." This limit is probably more than sufficient given the method by which plantings are scheduled. However, should this figure appear in output, a user must be aware that it is possibly low and should check output for the actual harvest date.

Another mechanical feature of the simulation of which users should be aware is the truncation which occurs in several computations. There are three steps where truncation occurs. The first is in calculating the planting size of partial plantings made to augment yield of an initial partial planting. Planting size is determined as follows:

$$\text{planting size cultivar X + 1} = \frac{\text{daily cannery capacity} - \left(\text{expected yield/A, partial planting for cultivar X} \times \text{planting size (A) of cultivar X}\right)}{\text{expected yield/A of cultivar X + 1}}$$

Because this division is done in integer arithmetic, truncation occurs, resulting in a total number of acres planted in this planting which is some portion of one acre less than strictly required to meet cannery capacity. This will almost always be an insignificant error since expected yields are means and actual yield is ultimately determined stochastically.

Truncation also occurs in determining the dates on which a subplant reaches 80 TR and 140 TR. Using the date of 80 TR as an example, and recalling the formula:

$$\text{date of 80 TR for subplant X} = \text{date of 100 TR for subplant X} - \frac{\text{20 TR points}}{\text{TR points/day between 80 TR and 100 TR}}$$

It is clear that integer arithmetic may result in truncation. The date arrived at may be some portion of a day early. This is trivial considering variance in maturity dates.

The third truncation occurs in yield calculations. Yield at 100 TR is initially stored as a real number, as are the values of per cent yield at the two extreme TR readings. Final yield is integer, however. Again, this is a small error relative to the numbers dealt with.

Standard industry practice is to double or triple the size of the first planting to provide a backlog on which to work. The planting multiple is user-supplied in PSIM. However, if the user-supplied multiple requires planting a number of acres exceeding the total acreage requirement of the first cultivar, the program automatically reduces the multiple stepwise by one until the required acreage is less than or equal to that cultivar's acreage requirement.

Users of program PSIM have a choice of providing the increase in TR points/day at harvest or providing the ddays/increase of one TR point at harvest. These values are then used to determine the date at which a subplant reaches 80 TR and 140 TR. The question arises as to the difference in dates arrived at using the two sets of figures. This depends largely on the values provided. If the values chosen for the two are based on general values available in the literature, dates calculated by the two methods are within one day of each other. (See Appendix C for sample output in which both methods are used). Providing the ddays/increase of one TR point does, however, require an extra step in the date calculation and will therefore involve an extra truncation.

A word about the role of the user-supplied acreage divisor: its significance must be fully recognized. The acreage divisor is used to determine SPS for each cultivar in the simulation. Mean yield is the default value since that value will result in a planting which theoretically uses daily cannery capacity to the full. A user may prefer a different divisor if his goal is to reflect an unusual value or cost for a given cultivar. A divisor smaller than mean yield might be used if cannery slack were more costly than bunching, whereas a divisor greater than mean yield might be used to minimize risk for a high value, high cost cultivar. It must be kept in mind, however, that any divisor other than expected yield will cause cannery capacity to be over or under-used in the scheduling stage of the simulation (whereas using expected yield results in a probability of .5 for both over and under-use). This discrepancy may or may not be smoothed out by the creation of subplantings and rescheduling of yields in the second stage of the simulation.

As with any model, program PSIM is less than a complete picture of the system it tries to simulate. Some of the model's weaknesses are fairly obvious. The number of cultivars and plantings it handles on one run is limited. At least some of the data required to run the simulation may not be known by, or easily available to, the user. The simulation does not take into account any vagaries of environment other than temperature. There are other weaknesses as well, which might have been minimized or done away with had a larger modelling effort been feasible.

One such weakness is the fact that yields can only be rescheduled to 80 TR or 140 TR, and then the smallest acreage which can be rescheduled is an entire subplanting. One defense of the first aspect is that dates of 80 TR and 140 TR are usually not spread over a period longer than a few days. The second aspect of the problem, not being able to reschedule a portion of a subplant, is simply unrealistic, but the smaller the subplantings, the less serious this will be.

Another weakness of the simulation is that it does not include much in the way of a quality breakdown. This is partly because no such breakdown was available in the work of Norton and his associates. So little data is available for such a breakdown it seemed best to avoid contrivance in an area where relationships are uncertain. In any event, a user would probably balk at accepting figures which were not well substantiated.

As discussed in the literature review, there is strong evidence that the maturity constant for a given cultivar is not the same throughout the growing season. There is no facility within PSIM to allow the user to vary maturity constant. There were no figures available on the relationship between maturity constant and phenological events, or an attempt might have been made to include this.

Neither does the simulation allow a user to change expected yield figures as the season progresses. Yet late plantings are almost always associated with lower yields than early plantings. This facility might reasonably be incorporated in the model in the future.

Actually, there are several directions in which future work might profitably extend. Further work on quality breakdown would be useful, although it would require considerable data collection and correlation. It would also be useful to incorporate a graphing subroutine to plot, after Seaton (37), accumulated ddays and mean temperature against time, both historically and for a current year. And eventually, a deterministic submodel could be developed to simulate the price/cost situation involved for canners.

It seems unlikely that program PSIM could be used in its present form for other species. This is primarily because of the pivotal role played in the simulation by TR readings in general and 80, 100 and 140 TR specifically. Different maturity tests are used for other canning crops. Enough of the program subroutines contain tests and calculations which would be relevant to other crops, however, that substantial portions of the program could be used in developing additional scheduling models.

The discussion so far has centered around limitations of program PSIM. Despite these, the simulation may prove useful to a canner in a number of ways. It must be stressed, from the beginning, that the simulation is intended as an aid to management, not as a substitute for it. In situations where the production process is subject to an uncertain environment, there are at least two possible management approaches. One approach would be an attempt to increase environmental control--this approach is clearly limited. Another approach is to attempt to create a hedge against the uncertainty involved. Once environmental instability is accepted, management can try to reduce the impact of this instability. This places the emphasis on formulation of "strategic," long-term policies rather than "tactical," short-term policies which relate to actions to be taken when confronted with a specific situation. Program PSIM, by allowing tests of decision rules, may be used with long-term policy formulation as a goal.

There are at least eight decision rules which program PSIM can be used to test. Both the base temperature and maturity constants can be easily tested. The size of individual plantings and the total acreage planted in a given season can also be varied. Temperature data may be varied to determine whether soil or air temperatures result in more accurate maturity predictions, and temperature data from sites at different distances from the fields can be compared for accuracy. Use of the heat unit system in general, both as a means of scheduling and as a guide to maturity prediction, can be assessed as the number of simulation runs increases. In addition, a canner may vary lengths of cannery shifts to determine plant efficiency.

However, perhaps as important as any decision rule tests, assembling data for the simulation will force a user to be aware of many of the assumptions and decisions he already makes in scheduling his crop. This, in turn, may spur him to keep better or different records or may raise questions about the data he has assembled. Correspondence with canners revealed considerable interest in a computer program along the lines of PSIM. This suggests that the data organization required to make use of the simulation may not be a large deterrent to its use.

Mention must be made at this point of model validation and verification. "A model is validated in relation to the purpose for which it is constructed, whereas a model is verified in relation to absolute truth" (47). Program PSIM is validated, in that it accurately produces the output required of it. Of course, the ultimate validation test is whether the model leads to better decisions than can be made using other techniques. Validation in this sense is yet to be done for program PSIM and is linked to sensitivity analyses of various decision parameters.

According to the above definition, PSIM has not been verified. Data used to validate the model came from the Fort Lupton Canning Company[1] in Fort Lupton, Colorado and from a review of the literature. Verification would require far more extensive data than is

---

[1] Many thanks to Mr. Ben Counter, President, Fort Lupton Canning Company, Fort Lupton, Colorado, for much of the data used to validate program PSIM.

available to anyone but a canner himself. In addition, climatic variability, the backbone of this simulation, necessitates replication of the model over time to accumulate sufficient results for in-depth analysis. The nature of the heat unit system for crop scheduling, however, is such that the greatest value of modelling lies in varying the levels of system inputs (i.e., in sensitivity analyses) in order to establish overall trends in system behavior. Elucidation of these trends, rather than arrival at any one optimum solution, is the ultimate goal of the simulation. It is hoped that the potential benefits of program PSIM in this regard provide sufficient motivation for its use by individual canners.

LITERATURE CITED

1. Anderson, R.L. and A. Maass. 1974. A simulation of irrigation systems. U.S.D.A. Tech. Bull. 1431. Washington, D.C.

2. Anonymous. 1977. The almanac of the canning, freezing and preserving industries. E.E. Judge and Son, Inc. Westminster, MD.

3. Arnold, C.Y. 1959. The determination and significance of the base temperature in a linear heat unit system. Proc. Amer. Soc. Hort. Sci. 74:430-445.

4. ________. 1960. Maximum-minimum temperatures as a basis for computing heat units. Proc. Amer. Soc. Hort. Sci. 76: 682-692.

5. ________. 1974 Predicting stages of sweet corn (*Zea mays* L.) development. J. Amer. Soc. Hort. Sci. 99(6):501-505.

6. Arthey, V.D. 1975. Quality of horticultural products. John Wiley and Sons. New York.

7. Baker, C.H. and R.D. Horrocks. 1973. A computer simulation of corn grain production. Trans. ASAE. 16:1027-1031.

8. Boswell, V.R. 1926. The influence of temperature on growth and yield of garden peas. Proc. Amer. Soc. Hort. Sci. 23: 162-168.

9. ________. 1929. Factors influencing yield and quality of peas--biophysical and biochemical studies. Maryland Agr. Expt. Sta. Bull. 306. pp. 341-382.

10. Capiro, J.M. 1971. The solar-thermal unit theory in relation to plant development and potential evapotranspiration. Montana Agr. Expt. Sta. Circ. 251. Mont. State Univ. Bozeman, MT.

11. Chen, L.H., C.H. Miller and R.S. Sowell. 1975. Simulation models for harvesting of pickling cucumbers. J. Amer. Soc. Hort. Sci. 100(2):176-179.

12. Edey, S.N. 1977. Growing degree-days and crop production in Canada. Can. Dept. of Agr. Publ. 1635.

13. Ells, J.E. 1971. Canning crop notes. Coop. Ext. Ser. Colo. State Univ. Ft. Collins, CO. 10(4).

14. Fletcher, H.F., D.P. Ormrod, A.R. Maurer and B. Stanfield. 1966. Response of peas to environment I. Planting date and location. Can. J. Plnt. Sci. 46:77-85.

15. Gilmore, C.E., Jr. and J.S. Rogers. 1958. Heat units for measuring maturity in corn. Agron. J. 50:611-615.

16. Gritton, E.T. and R.D. Ebert. 1975. Interaction of planting date and powdery mildew on pea plant performance. J. Amer. Soc. Hort. Sci. 100(2):137-142.

17. Hagedorn, D.J., L.G. Holm and J.H. Torrie. 1955. Yield-quality relationships as influenced by maturity of canning peas. Wisc. Agr. Expt. Sta. Res. Bull. 187.

18. Hardwick, R.C. and G.M. Milbourn. 1967. Yield analysis in the vining pea. Agricul. Prog. 42:24-31.

19. Haupt, W. 1969. Pisum sativum L. in The induction of flowering. Cornell Univ. Press. Ithaca, N.Y. pp. 393-408.

20. Holmes, R.M. and G.W. Robertson. 1959. Heat units and crop growth. Can. Dept. of Agr. Publ. 1042.

21. Holt, D.A., R.J. Bula, G.E. Miles, M.M. Schreiber and R.M. Peart. 1975. Environmental physiology, modeling and simulation of alfalfa growth I. Conceptual development of SIMED. Agr. Expt. Sta. Res. Bull. 907. Purdue Univ. West Lafayette, IND.

22. Hope, G.W. 1962. Modification of the heat unit formula for peas. Can. J. Plnt. Sci. 42:15-21.

23. Johson, S.S. and M. Zahara. 1976. Prospective lettuce harvest mechanization: impact on labor. J. Amer. Soc. Hort. Sci. 101(4):378-381.

24. Karr, E.J., A.J. Linck and C.A. Swanson. 1959. The effect of high temperature during day and night periods on pea yields. Amer. J. Bot. 46:91-93.

25. Katz, Y.H. 1952. The relationship between heat unit accumulation and the planting and harvesting of canning peas. Agron. J. 44:74-78.

26. Kramer, A. and B.A. Twigg. 1970. Quality control for the food industry Vol. 1 Fundamentals. The AVI Publishing Company, Inc. Westport, CT.

27. Maurer, A.R., D.P. Ormrod and H.F. Fletcher. 1968. Response of peas to environment IV. Effect of five soil water regimes on growth and development of peas. Can. J. Plnt. Sci. 48:129-137.

28. Milbourn, G.M. and R.C. Hardwick. 1968. The growth of vining peas I. The effect of time of sowing. J. Agr. Sci. 70:393-402.

29. Nonnecke, I.L., N.O. Adepipe and D.P. Ormrod. 1971. Temperature and humidity effects on the growth and yield of pea cultivars. Can. J. Plnt. Sci. 51:479-484.

30. Norton, R.A., W.E. Bratz and T.S. Russell. 1968. An analysis of pea varieties and selections for freezing and canning in northwestern Washington--1967. Wash. Agr. Expt. Sta. Circ. 438.

31. Paton, D.M. 1968. Photoperiodic and temperature control of flower initiation in the late pea cultivar Greenfeast. Aust. J. Bio. Sci. 21:609-617.

32. Pollard, L.H., E.B. Wilcox and H.B. Peterson. 1947. Maturity studies with canning pea. Utah Agr. Expt. Sta. Bull. 328.

33. Pumphrey, F.V., R.E. Ramig and R.R. Allmaras. 1975. Yield-tenderness relationships in 'Dark Skinned Perfection' peas. J. Amer. Soc. Hort. Sci. 100(5):507-509.

34. Reath, A.N. and S.H. Wittwer. 1952. Effects of temperature and photoperiod on the development of pea varieties. Proc. Amer. Soc. Hort. Sci. 60:301-310.

35. Sayre, C.B., W.T. Tapley and D.W. Barton. 1953. Variety comparison of peas used for canning and freezing, 1952. N.Y. Agr. Expt. Sta. Bull. 758.

36. ________, J.J. Willaman and Z.I. Kertesz. 1931. Factors affecting the quality of commercial canning peas. N.Y. Agr. Expt. Sta. Tech. Bull. 176. Geneva, NY.

37. Seaton, H.L. 1955. Scheduling plantings and predicting harvest maturities for processing vegetables. Food Tech. 9:202-209.

38. Smittle, D. and G. Bradley. 1966. The effects of irrigation, planting and harvest dates on yield and quality of peas. Proc. Amer. Soc. Hort. Sci. 88:441-446.

39. Stapleton, H.N., D.R. Buxton, F.L. Watson, D.J. Nolting and D.N. Baker. 1973. COTTON: A computer simulation of cotton growth. Univ. of Ariz. Agr. Expt. Sta. Tech. Bull. 206. Tucson, AZ.

40. ________, M.D. Cannon and W.A. LePori. 1967. Cotton harvest-defoliation scheduling. ASAE. 10:226-232.

41. ________. 1970. Crop production system simulation. Trans. ASAE. 13(1):110-113.

42. Sullivan, G.H., G.E. Wilcox, R.Y. Uyeshiro and J.L. Robertson. 1974. Precipitation and the economic feasibility of harvest mechanization for processing tomatoes. HortSci. 9(4):389-391.

43. USDA. 1974. Agricultural statistics. Stat. Rep. Serv. U.S.D.A. Washington, D.C. 619 pp.

44. Wang, Jen-yu. 1960. A critique of the heat unit approach to plant response studies. Ecology. 41(4):785-790.

45. ________. 1962. The influence of seasonal temperature ranges on pea production. Proc. Amer. Soc. Hort. Sci. 80:436-438.

46. ________. 1972. Agricultural meterology. Milieu Information Service. San Jose, CA.

47. Wright, A. 1971. Farming systems, models and simulation. In Systems analysis in agricultural management. John Wiley and Sons. Australasia Pty. Ltd. Sydney.

48. Zusman, P. and A. Amiad. 1965. Simulation: a tool for farm planning under conditions of weather uncertainty. J. Farm Econ. 47:574-594.

APPENDICES

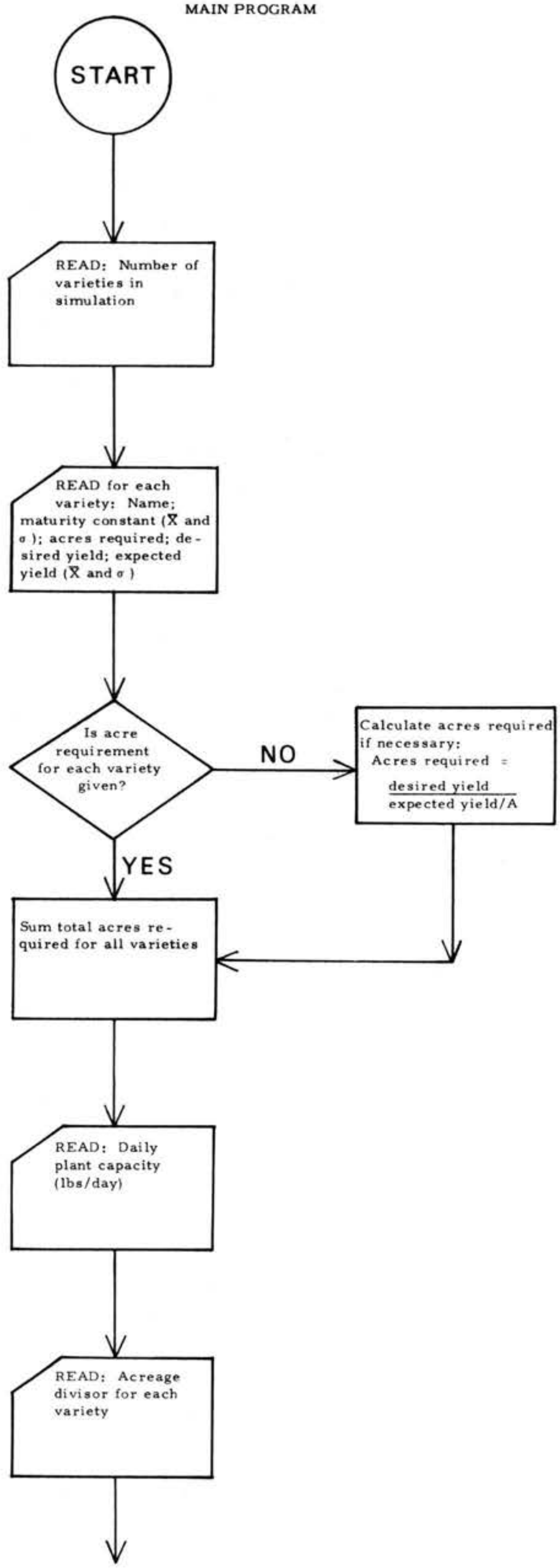
APPENDIX A

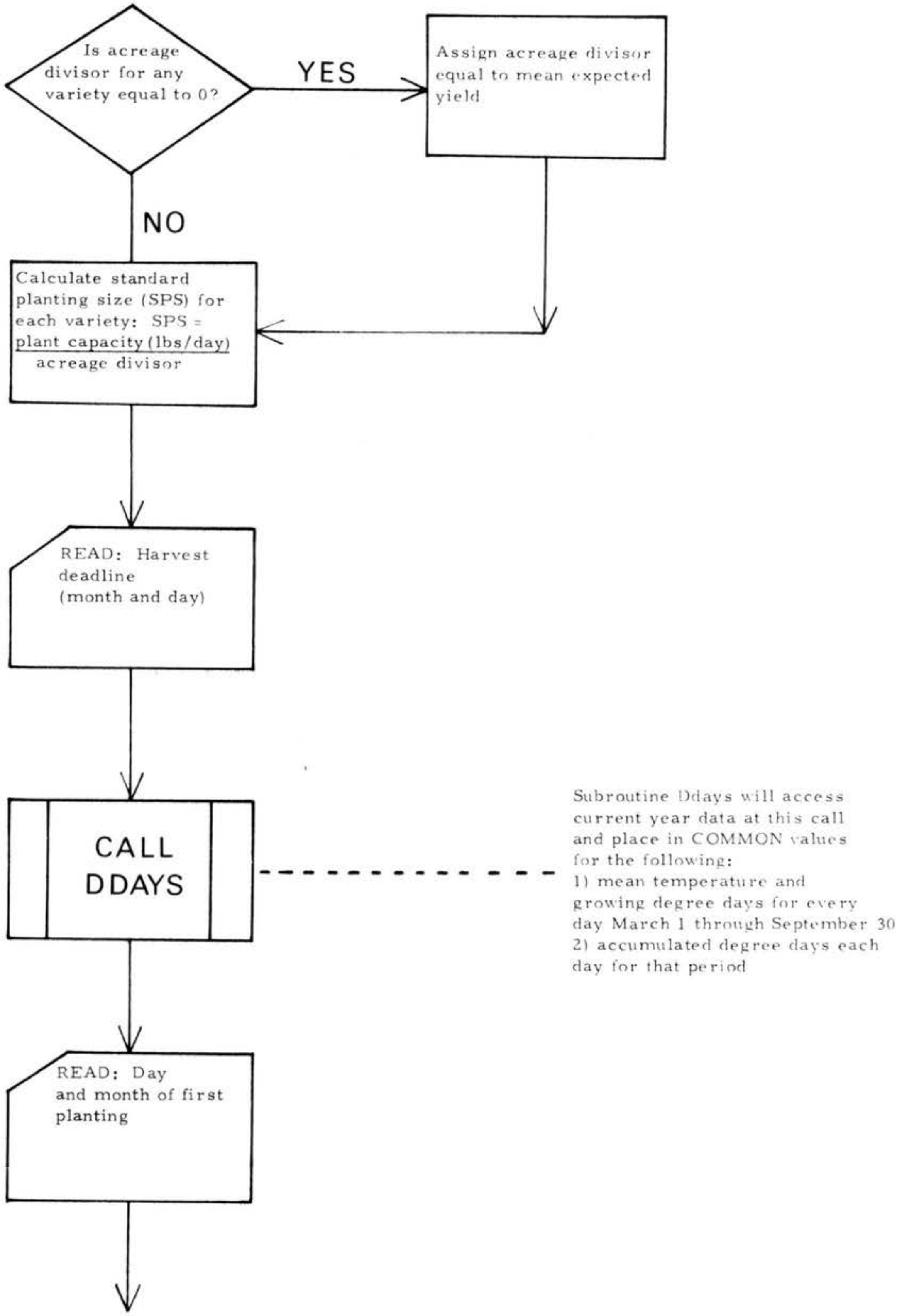
Abbreviations and Symbols Used

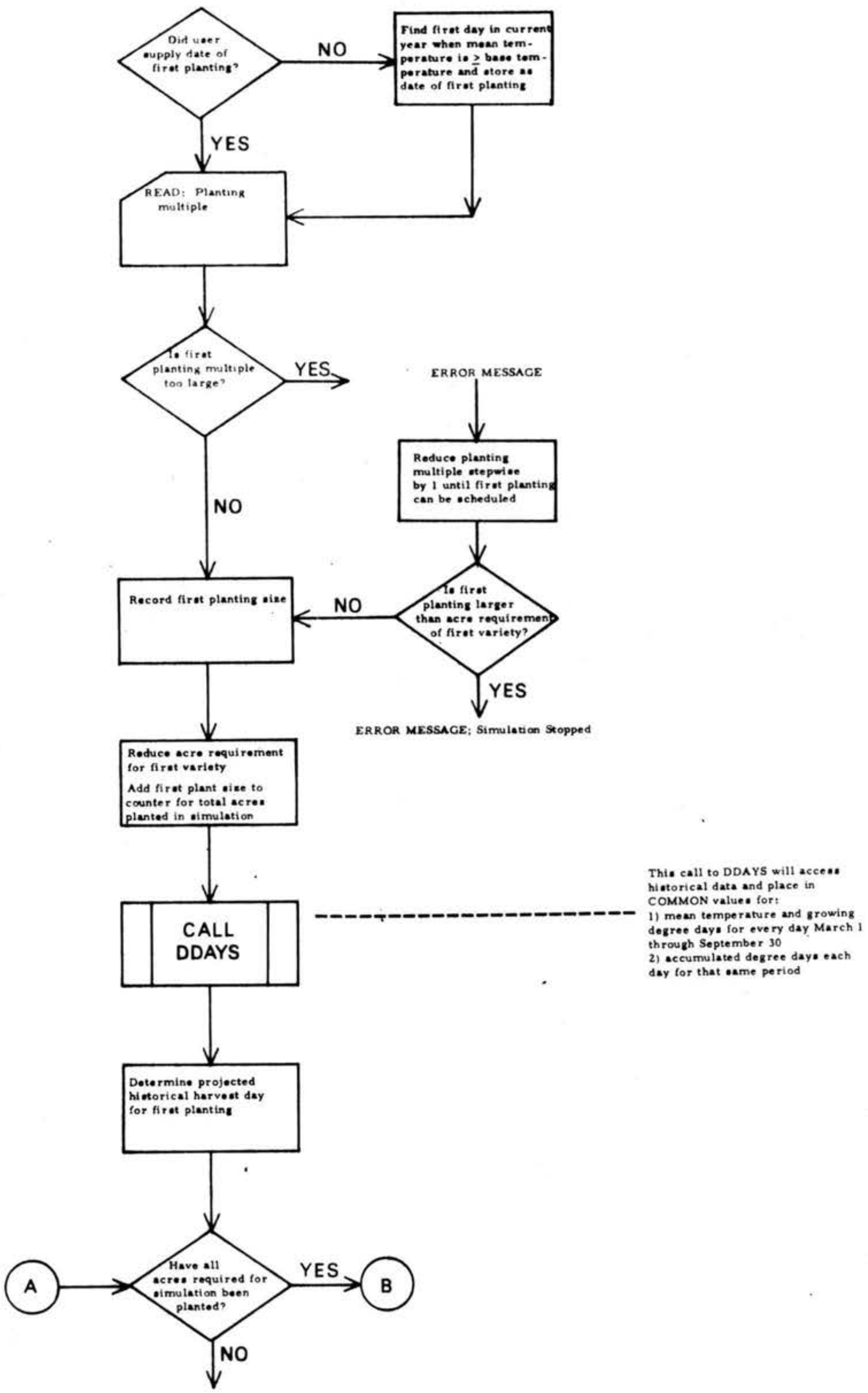
A	acre
dday	degree day
(CR)	carriage return
cv	cultivar
SPS	standard planting size
TR	tenderometer

APPENDIX B

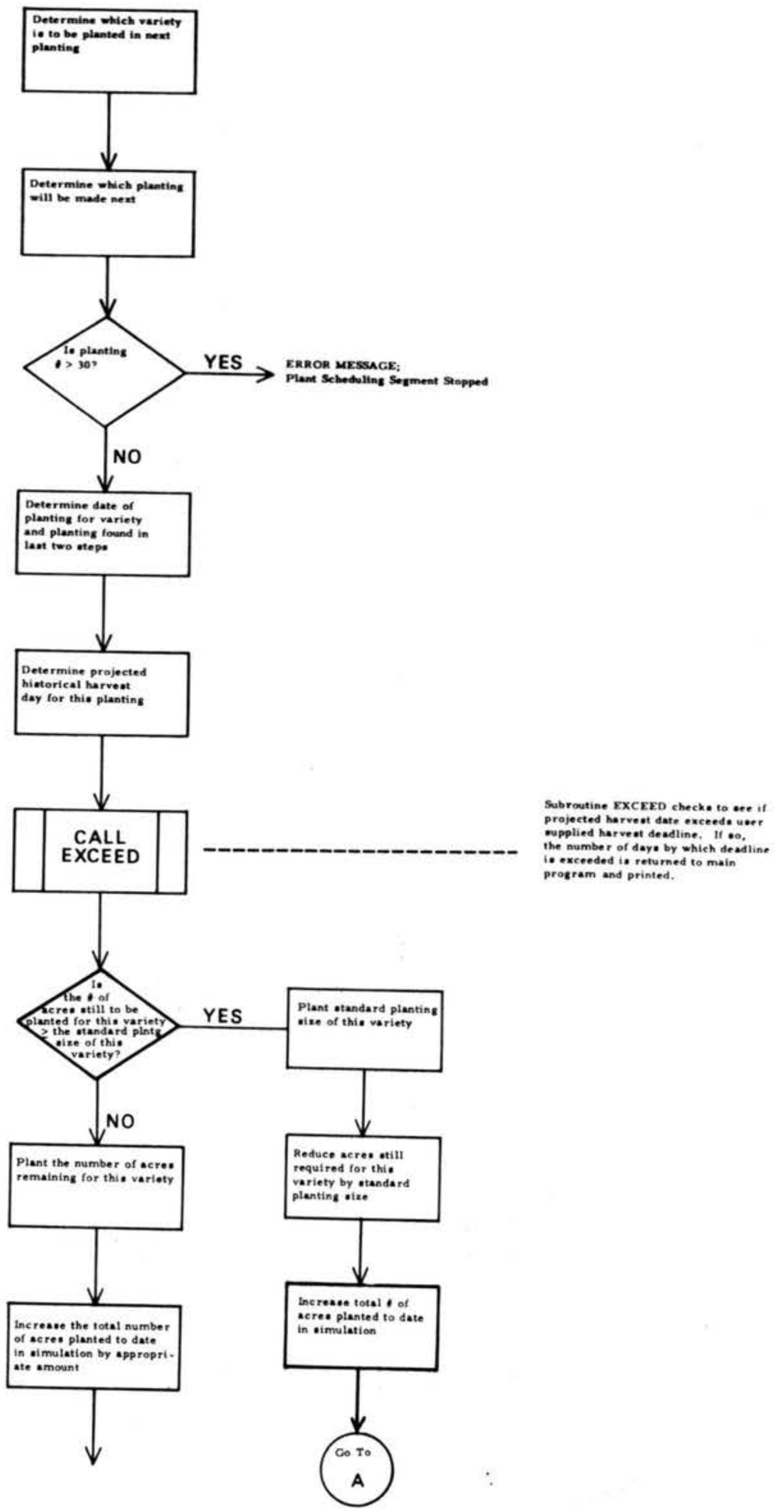
MAIN PROGRAM



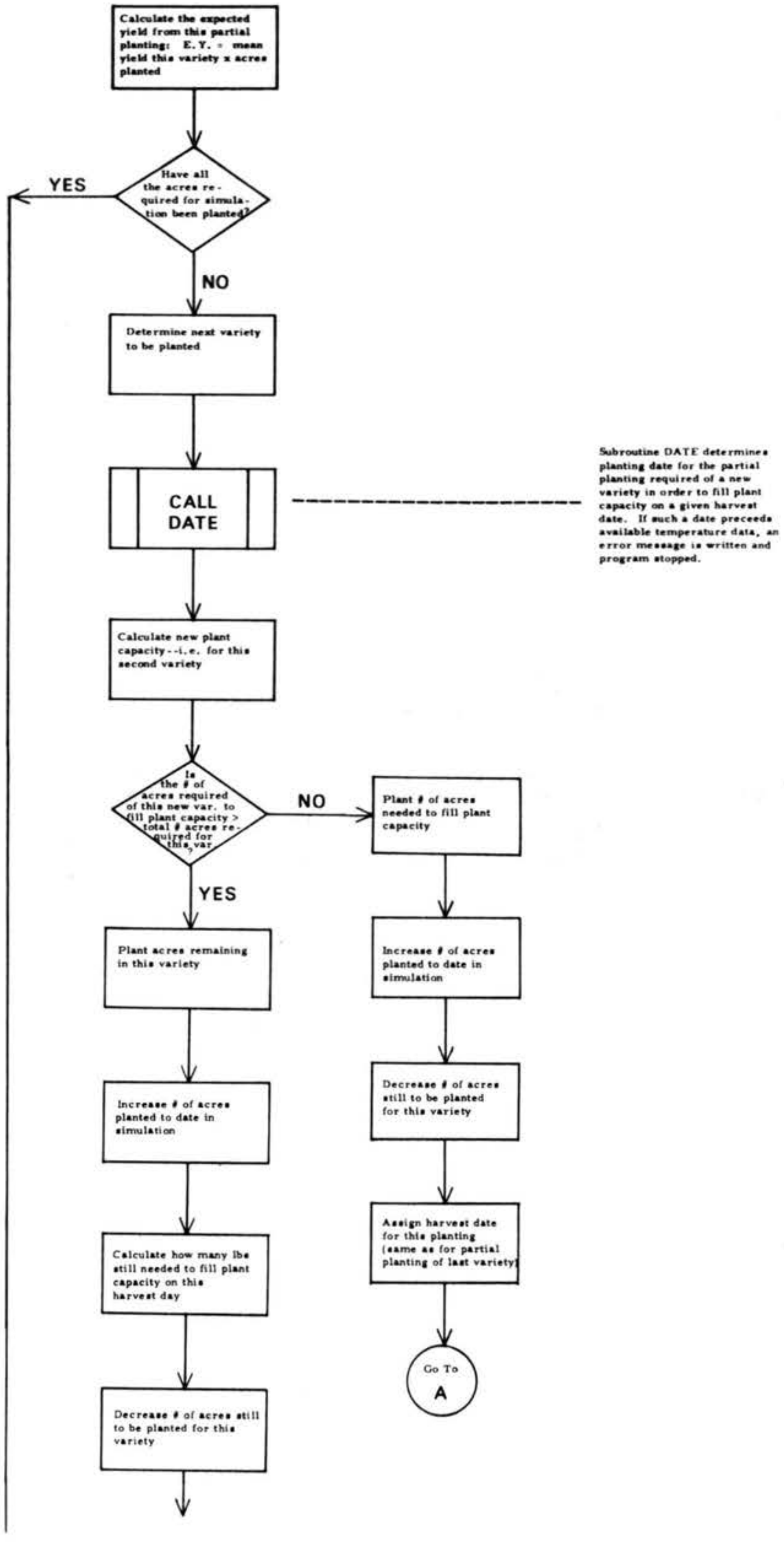

Is acreage divisor for any variety equal to 0?
YES
Assign acreage divisor equal to mean expected yield
NO
Calculate standard planting size (SPS) for each variety: SPS = plant capacity (lbs/day) / acreage divisor
READ: Harvest deadline (month and day)
CALL DDAYS
Subroutine Ddays will access current year data at this call and place in COMMON values for the following:
1) mean temperature and growing degree days for every day March 1 through September 30
2) accumulated degree days each day for that period
READ: Day and month of first planting



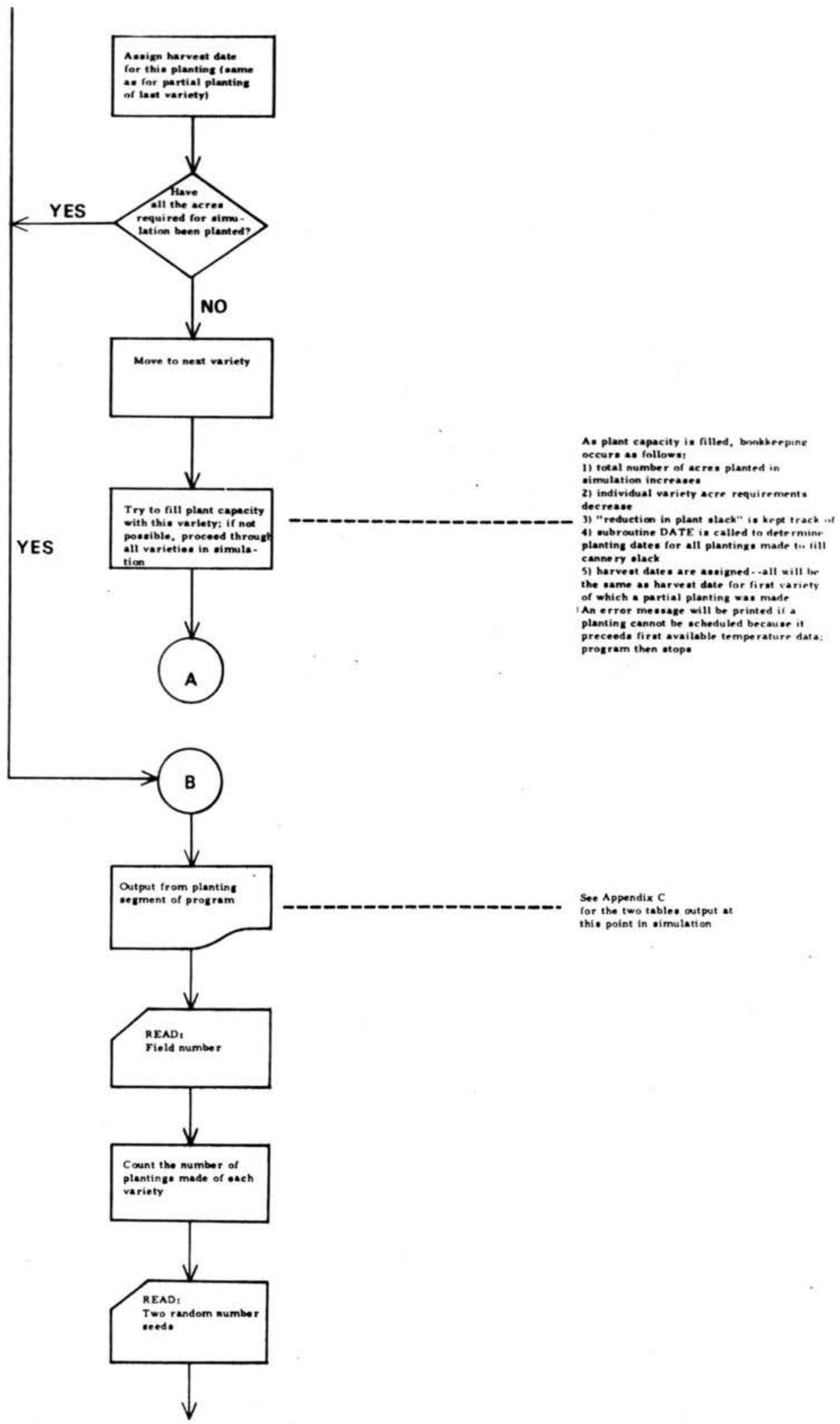
Did user supply date of first planting?
NO
Find first day in current year when mean temperature is ≥ base temperature and store as date of first planting
YES
READ: Planting multiple
Is first planting multiple too large?
YES
ERROR MESSAGE
Reduce planting multiple stepwise by 1 until first planting can be scheduled
NO
Record first planting size
NO
Is first planting larger than acre requirement of first variety?
YES
ERROR MESSAGE; Simulation Stopped
Reduce acre requirement for first variety
Add first plant size to counter for total acres planted in simulation
CALL DDAYS
This call to DDAYS will access historical data and place in COMMON values for:
1) mean temperature and growing degree days for every day March 1 through September 30
2) accumulated degree days each day for that same period
Determine projected historical harvest day for first planting
A
Have all acres required for simulation been planted?
YES
B
NO



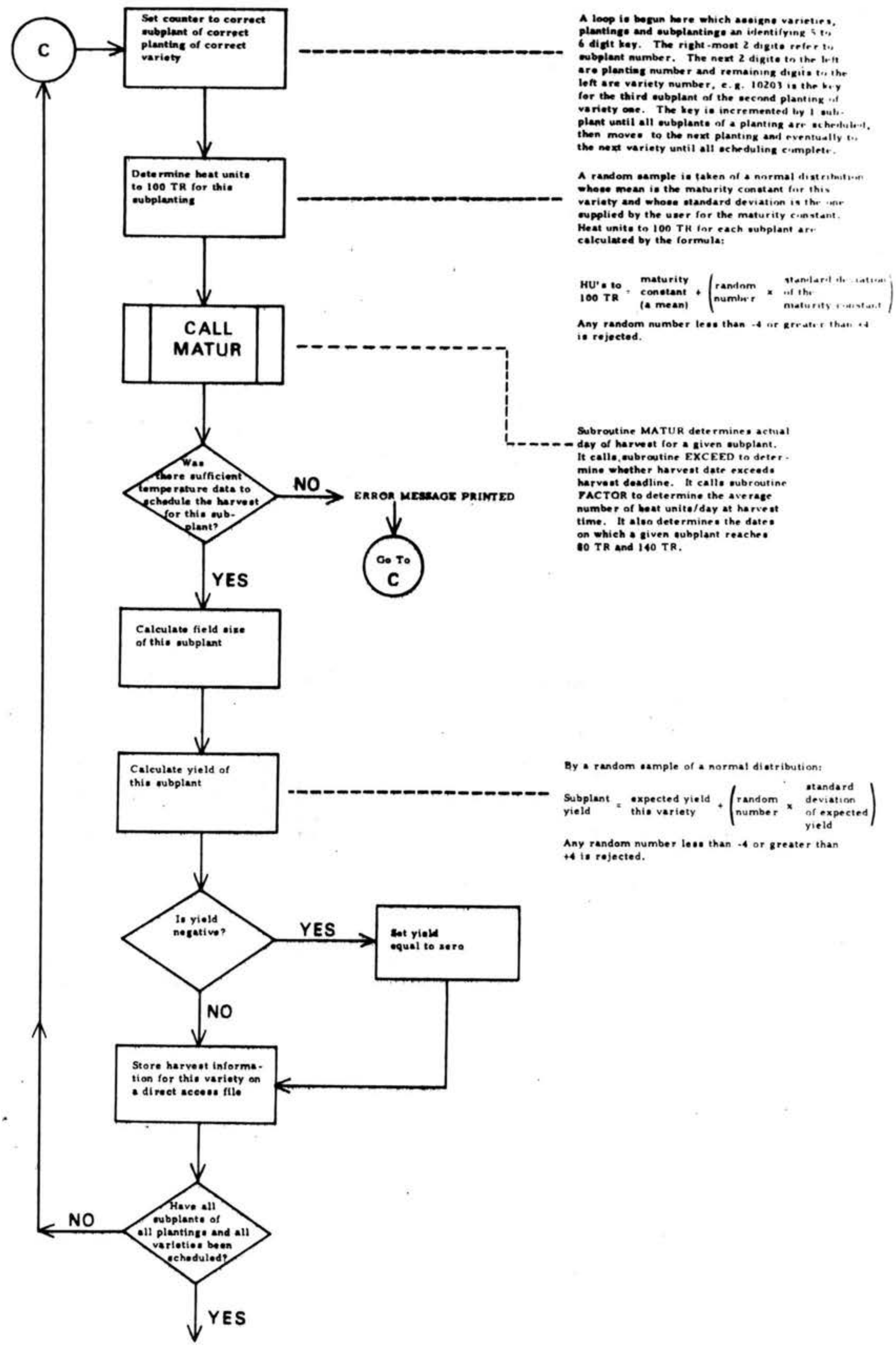
Determine which variety is to be planted in next planting
Determine which planting will be made next
Is planting # > 30?
YES
ERROR MESSAGE; Plant Scheduling Segment Stopped
NO
Determine date of planting for variety and planting found in last two steps
Determine projected historical harvest day for this planting
CALL EXCEED
Subroutine EXCEED checks to see if projected harvest date exceeds user supplied harvest deadline. If so, the number of days by which deadline is exceeded is returned to main program and printed.
Is the # of acres still to be planted for this variety > the standard plntg size of this variety?
YES
Plant standard planting size of this variety
NO
Plant the number of acres remaining for this variety
Reduce acres still required for this variety by standard planting size
Increase the total number of acres planted to date in simulation by appropri-ate amount
Increase total # of acres planted to date in simulation
Go To
A

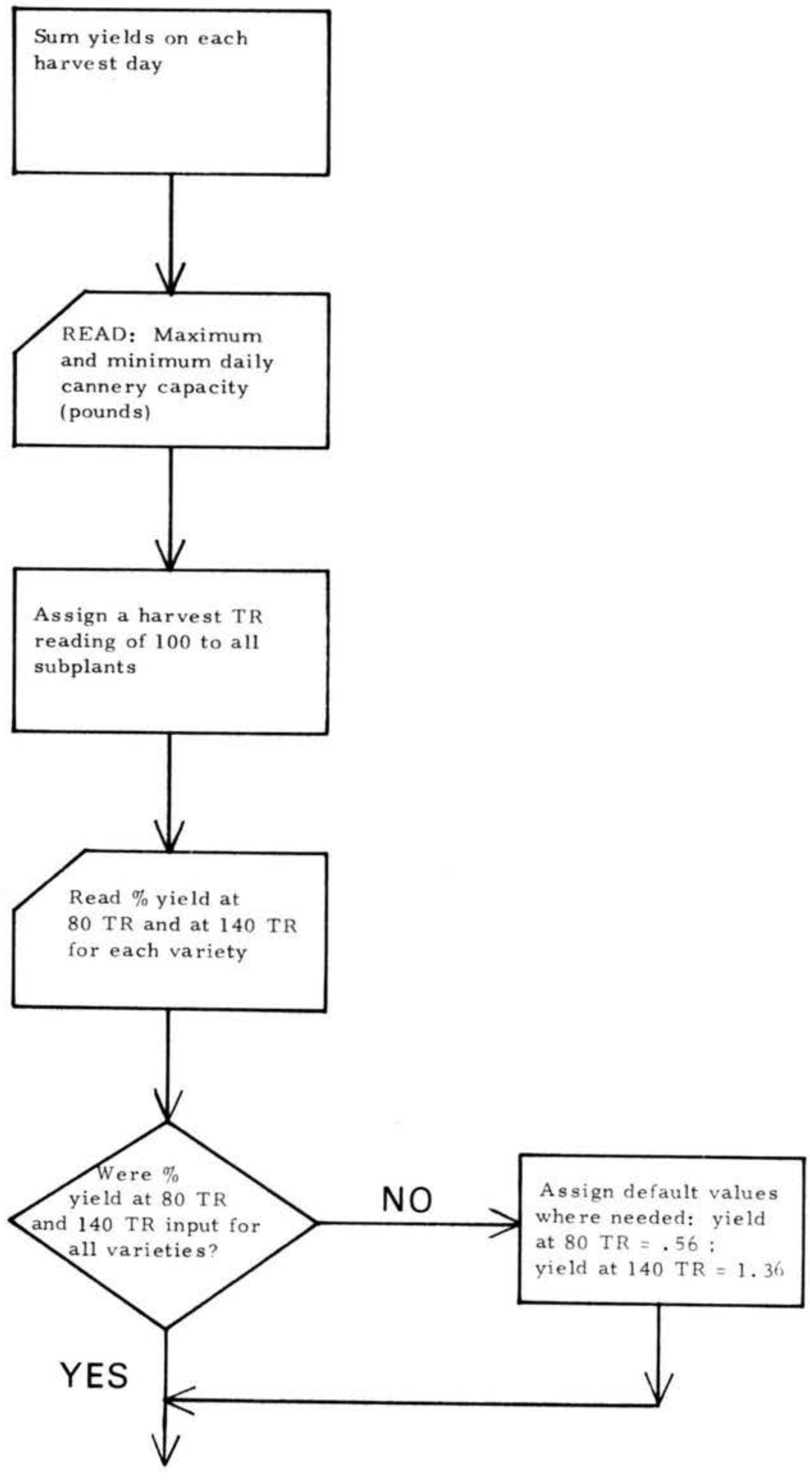


Calculate the expected yield from this partial planting: E.Y. = mean yield this variety x acres planted
Have all the acres required for simulation been planted?
YES
NO
Determine next variety to be planted
CALL DATE
Subroutine DATE determines planting date for the partial planting required of a new variety in order to fill plant capacity on a given harvest date. If such a date precedes available temperature data, an error message is written and program stopped.
Calculate new plant capacity--i.e. for this second variety
Is the # of acres required of this new var. to fill plant capacity > total # acres required for this var.?
NO
Plant # of acres needed to fill plant capacity
Increase # of acres planted to date in simulation
Decrease # of acres still to be planted for this variety
Assign harvest date for this planting (same as for partial planting of last variety)
Go To A
YES
Plant acres remaining in this variety
Increase # of acres planted to date in simulation
Calculate how many lbs still needed to fill plant capacity on this harvest day
Decrease # of acres still to be planted for this variety

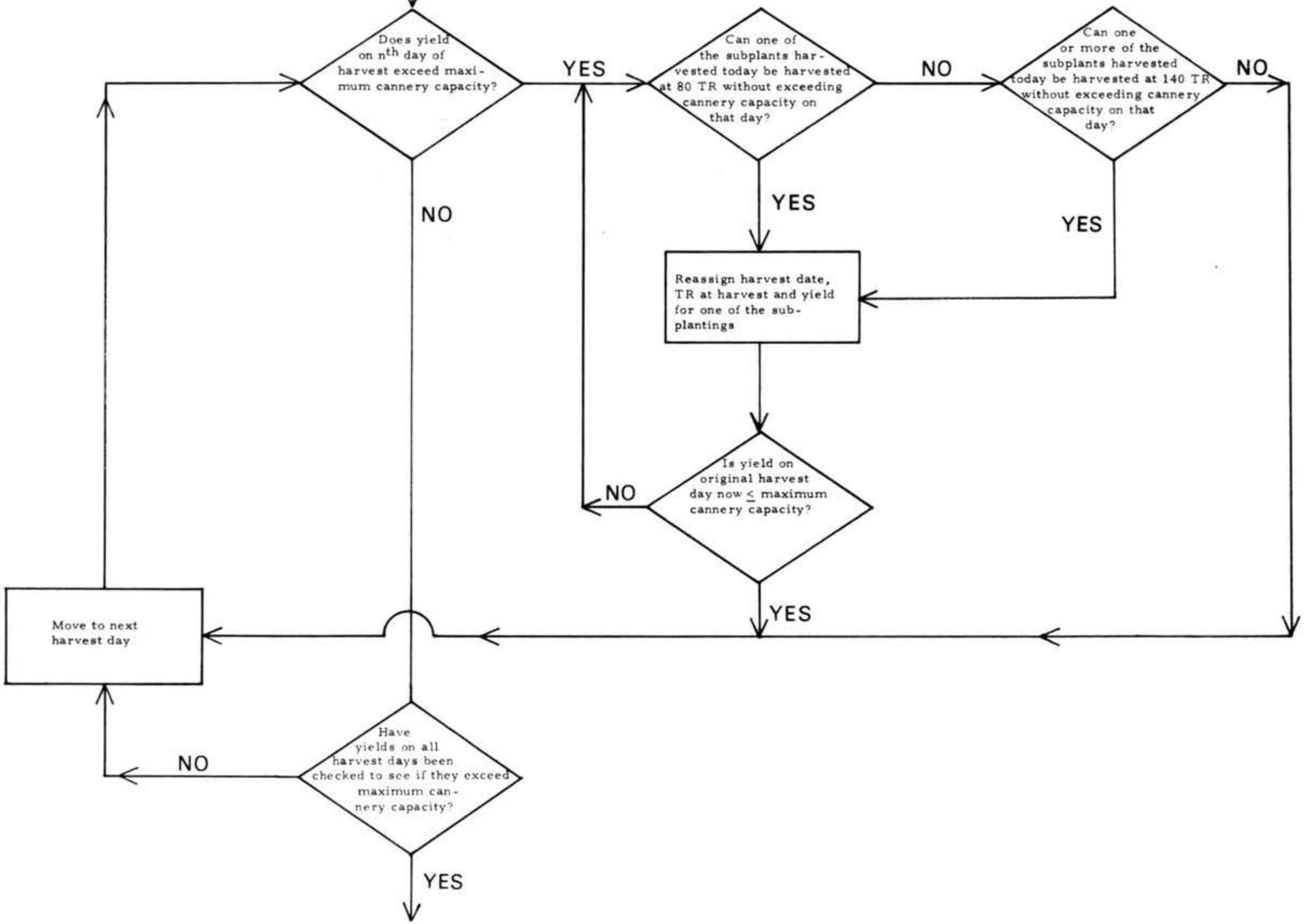


Assign harvest date for this planting (same as for partial planting of last variety)
Have all the acres required for simulation been planted?
YES
NO
Move to next variety
Try to fill plant capacity with this variety; if not possible, proceed through all varieties in simulation
YES
As plant capacity is filled, bookkeeping occurs as follows:
1) total number of acres planted in simulation increases
2) individual variety acre requirements decrease
3) "reduction in plant slack" is kept track of
4) subroutine DATE is called to determine planting dates for all plantings made to fill cannery slack
5) harvest dates are assigned--all will be the same as harvest date for first variety of which a partial planting was made
An error message will be printed if a planting cannot be scheduled because it preceeds first available temperature data; program then stops
A
B
Output from planting segment of program
See Appendix C for the two tables output at this point in simulation
READ: Field number
Count the number of plantings made of each variety
READ: Two random number seeds

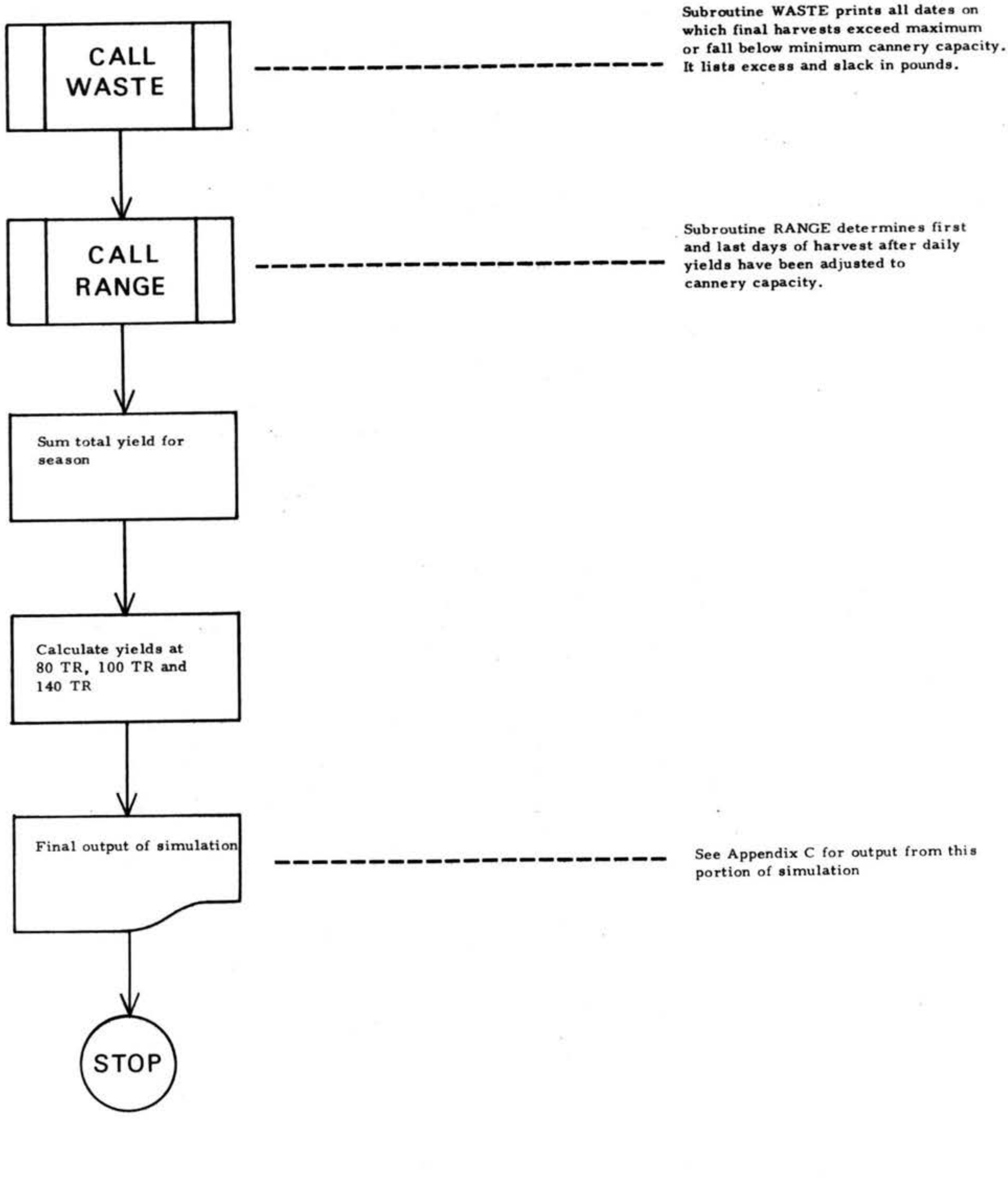

C
Set counter to correct subplant of correct planting of correct variety
A loop is begun here which assigns varieties, plantings and subplantings an identifying 5 to 6 digit key. The right-most 2 digits refer to subplant number. The next 2 digits to the left are planting number and remaining digits to the left are variety number, e.g. 10203 is the key for the third subplant of the second planting of variety one. The key is incremented by 1 subplant until all subplants of a planting are scheduled, then moves to the next planting and eventually to the next variety until all scheduling complete.
Determine heat units to 100 TR for this subplanting
A random sample is taken of a normal distribution whose mean is the maturity constant for this variety and whose standard deviation is the one supplied by the user for the maturity constant. Heat units to 100 TR for each subplant are calculated by the formula:
HU's to 100 TR = maturity constant (a mean) + (random number x standard deviation of the maturity constant)
Any random number less than -4 or greater than +4 is rejected.
CALL MATUR
Subroutine MATUR determines actual day of harvest for a given subplant. It calls subroutine EXCEED to determine whether harvest date exceeds harvest deadline. It calls subroutine FACTOR to determine the average number of heat units/day at harvest time. It also determines the dates on which a given subplant reaches 80 TR and 140 TR.
Was there sufficient temperature data to schedule the harvest for this subplant?
NO
ERROR MESSAGE PRINTED
Go To C
YES
Calculate field size of this subplant
Calculate yield of this subplant
By a random sample of a normal distribution:
Subplant yield = expected yield this variety + (random number x standard deviation of expected yield)
Any random number less than -4 or greater than +4 is rejected.
Is yield negative?
YES
Set yield equal to zero
NO
Store harvest information for this variety on a direct access file
Have all subplants of all plantings and all varieties been scheduled?
NO
YES



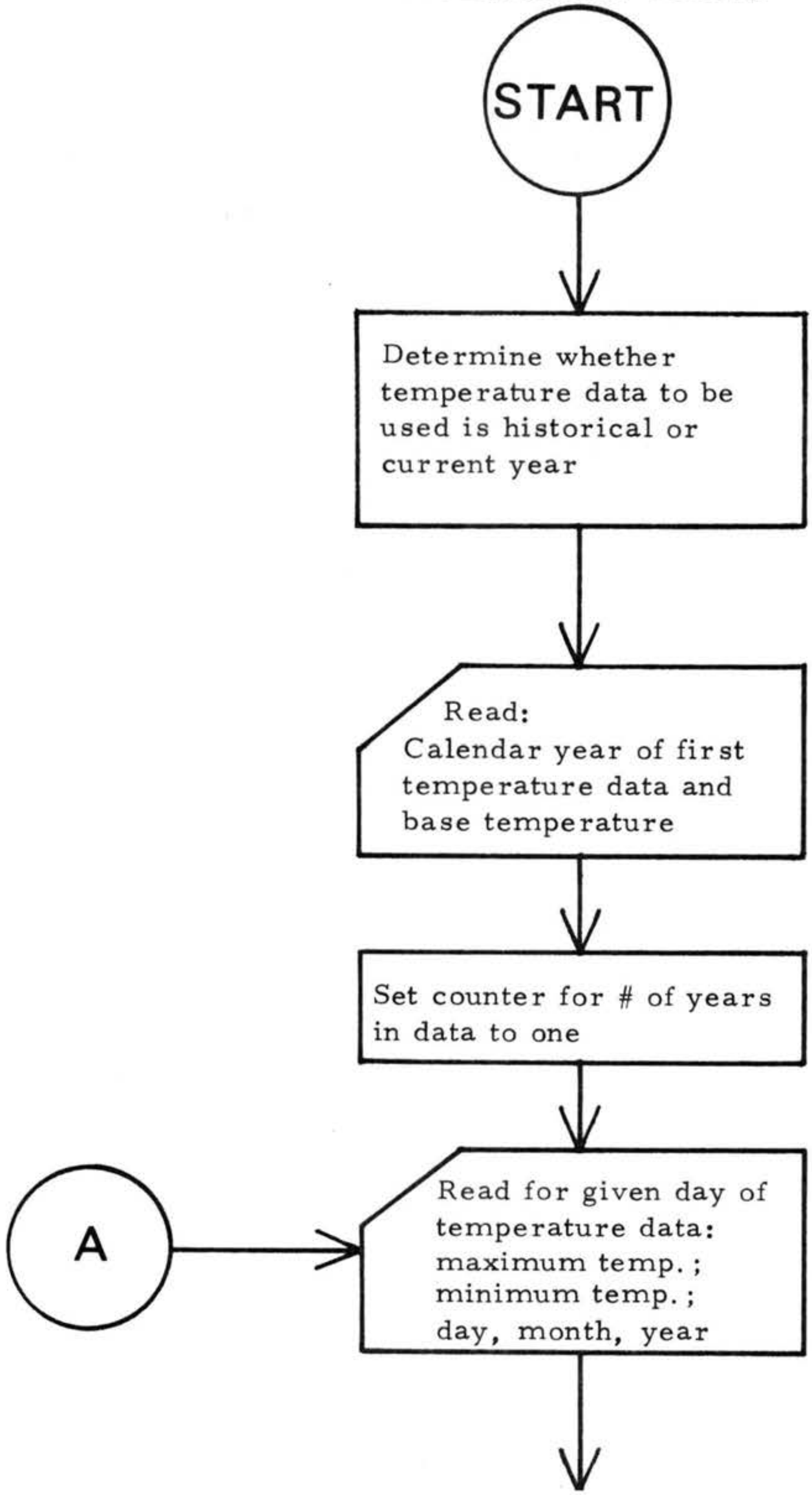
Sum yields on each harvest day
READ: Maximum and minimum daily cannery capacity (pounds)
Assign a harvest TR reading of 100 to all subplants
Read % yield at 80 TR and at 140 TR for each variety
Were % yield at 80 TR and 140 TR input for all varieties?
NO
Assign default values where needed: yield at 80 TR = .56 ; yield at 140 TR = 1.36
YES

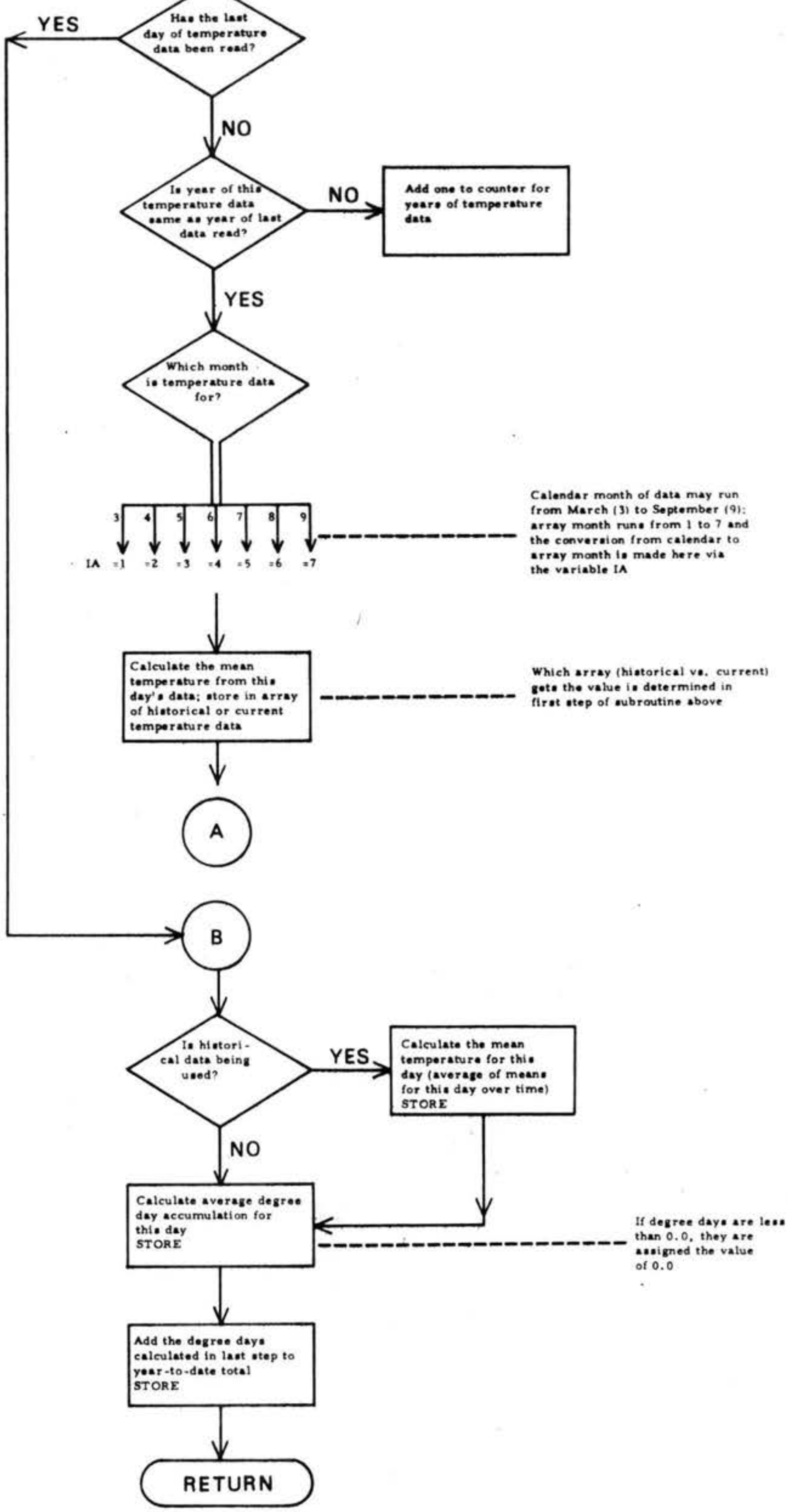


Does yield on nth day of harvest exceed maximum cannery capacity?
YES
Can one of the subplants harvested today be harvested at 80 TR without exceeding cannery capacity on that day?
NO
Can one or more of the subplants harvested today be harvested at 140 TR without exceeding cannery capacity on that day?
NO
YES
YES
Reassign harvest date, TR at harvest and yield for one of the subplantings
Is yield on original harvest day now ≤ maximum cannery capacity?
NO
YES
NO
Move to next harvest day
Have yields on all harvest days been checked to see if they exceed maximum cannery capacity?
NO
YES


CALL WASTE
Subroutine WASTE prints all dates on which final harvests exceed maximum or fall below minimum cannery capacity. It lists excess and slack in pounds.
CALL RANGE
Subroutine RANGE determines first and last days of harvest after daily yields have been adjusted to cannery capacity.
Sum total yield for season
Calculate yields at 80 TR, 100 TR and 140 TR
Final output of simulation
See Appendix C for output from this portion of simulation
STOP

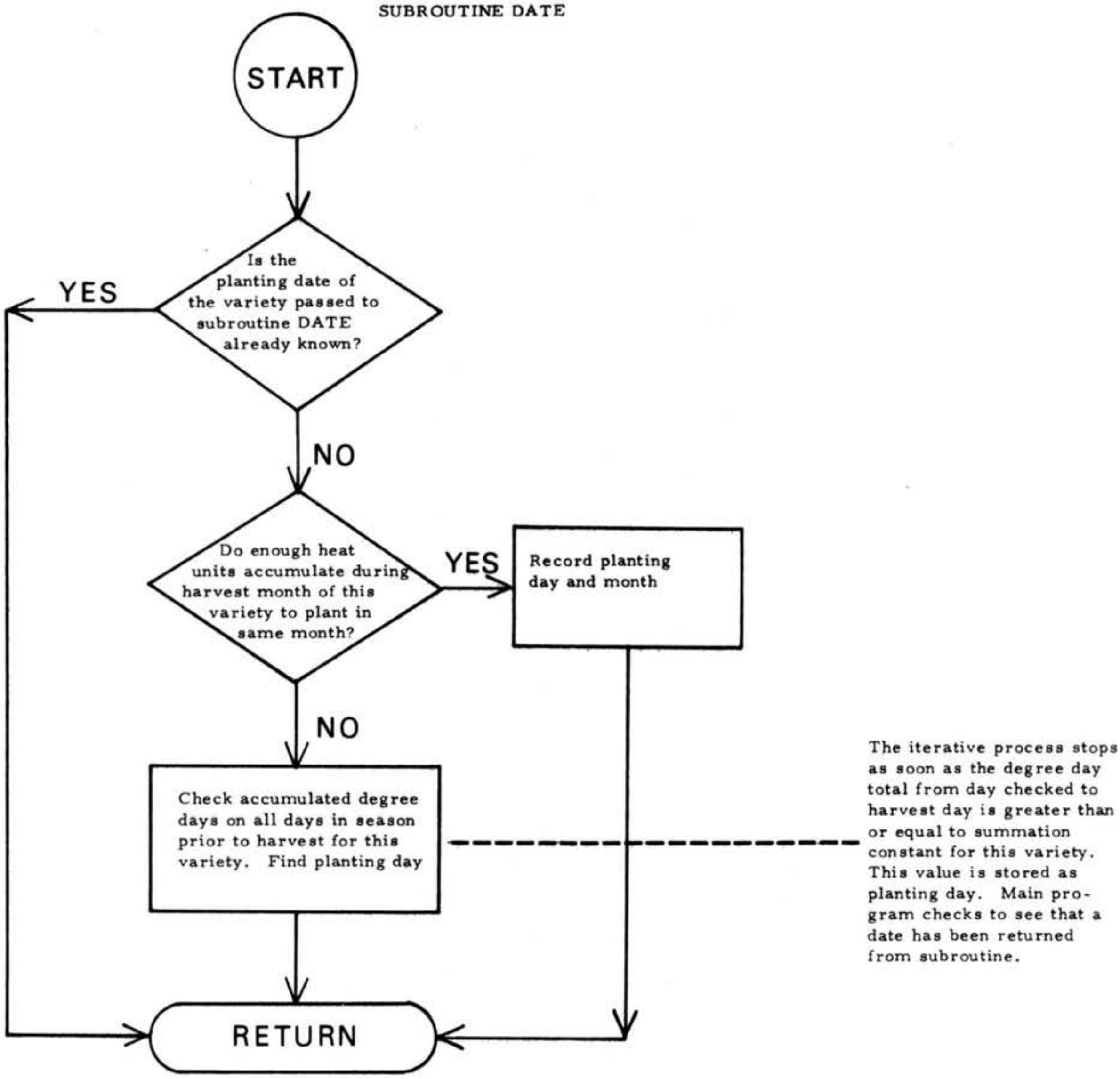
SUBROUTINE DDAYS



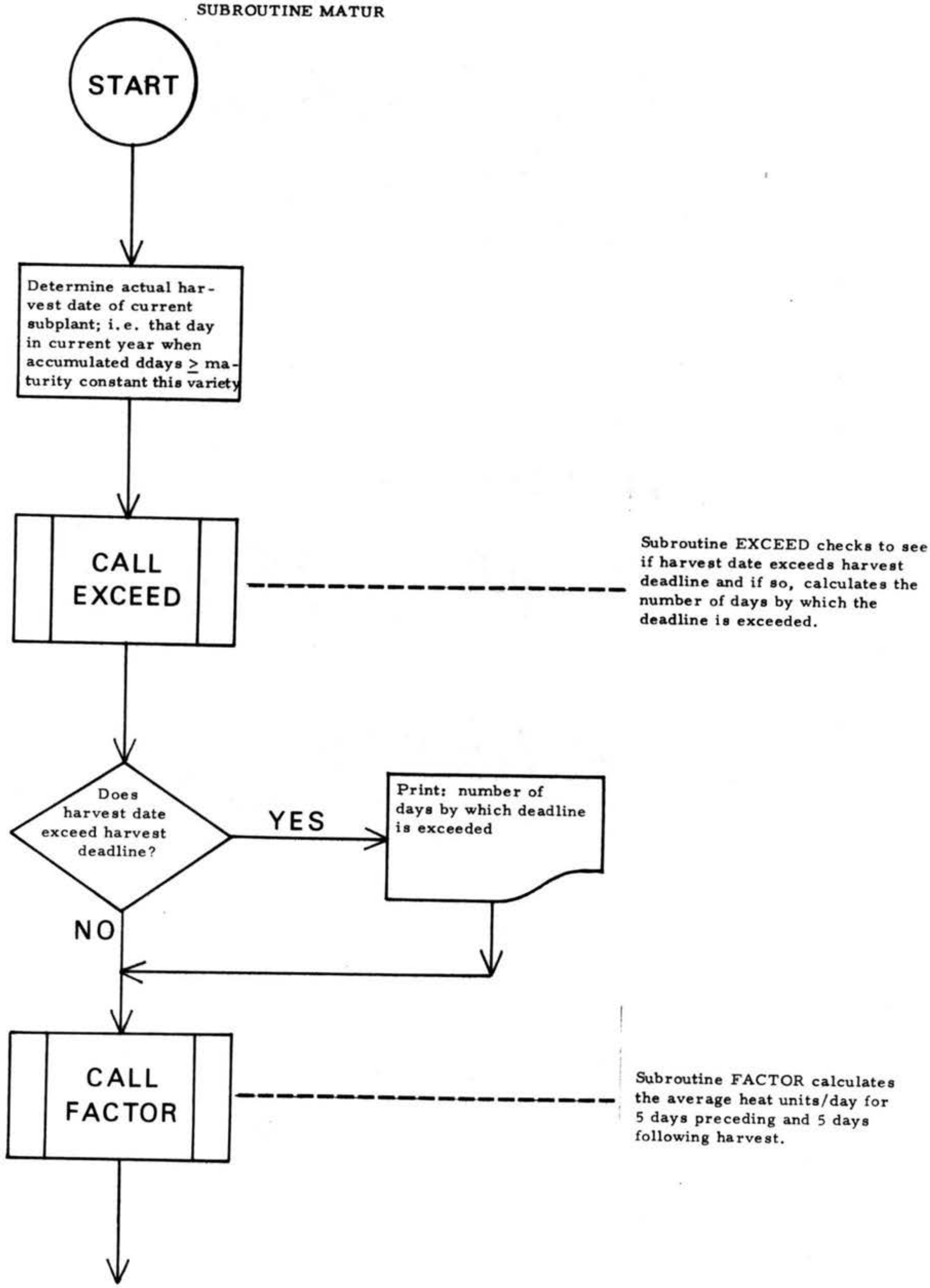


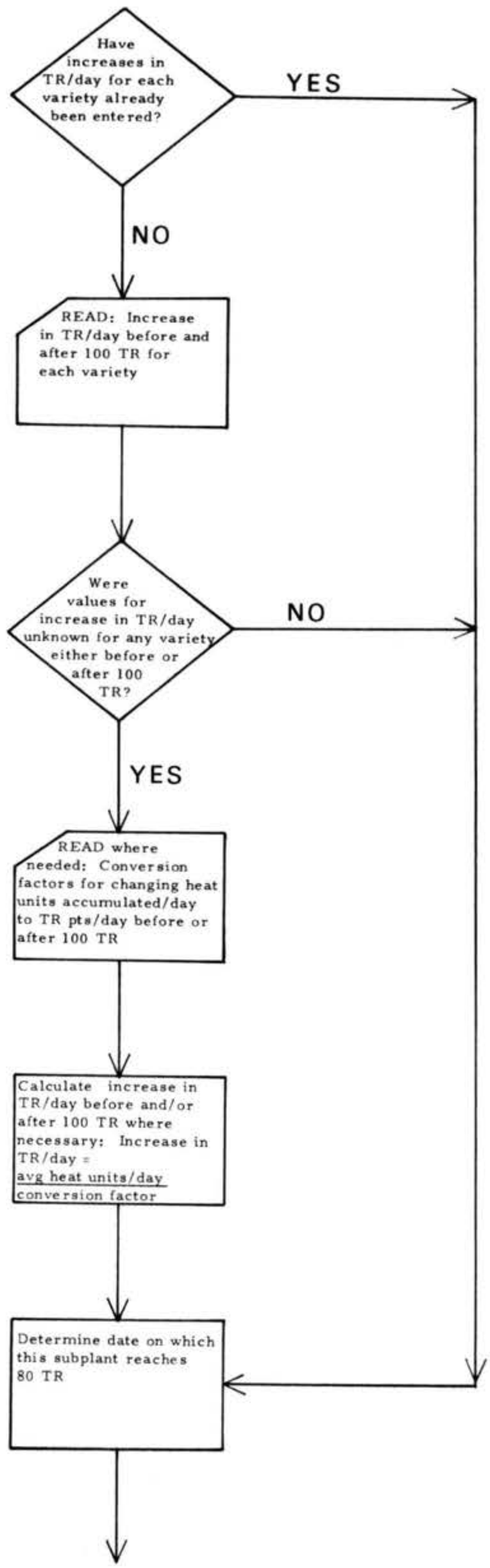
Has the last day of temperature data been read?
YES
NO
Is year of this temperature data same as year of last data read?
NO
Add one to counter for years of temperature data
YES
Which month is temperature data for?
3 4 5 6 7 8 9
IA =1 =2 =3 =4 =5 =6 =7
Calendar month of data may run from March (3) to September (9); array month runs from 1 to 7 and the conversion from calendar to array month is made here via the variable IA
Calculate the mean temperature from this day's data; store in array of historical or current temperature data
Which array (historical vs. current) gets the value is determined in first step of subroutine above
A
B
Is histori-cal data being used?
YES
Calculate the mean temperature for this day (average of means for this day over time) STORE
NO
Calculate average degree day accumulation for this day STORE
If degree days are less than 0.0, they are assigned the value of 0.0
Add the degree days calculated in last step to year-to-date total STORE
RETURN

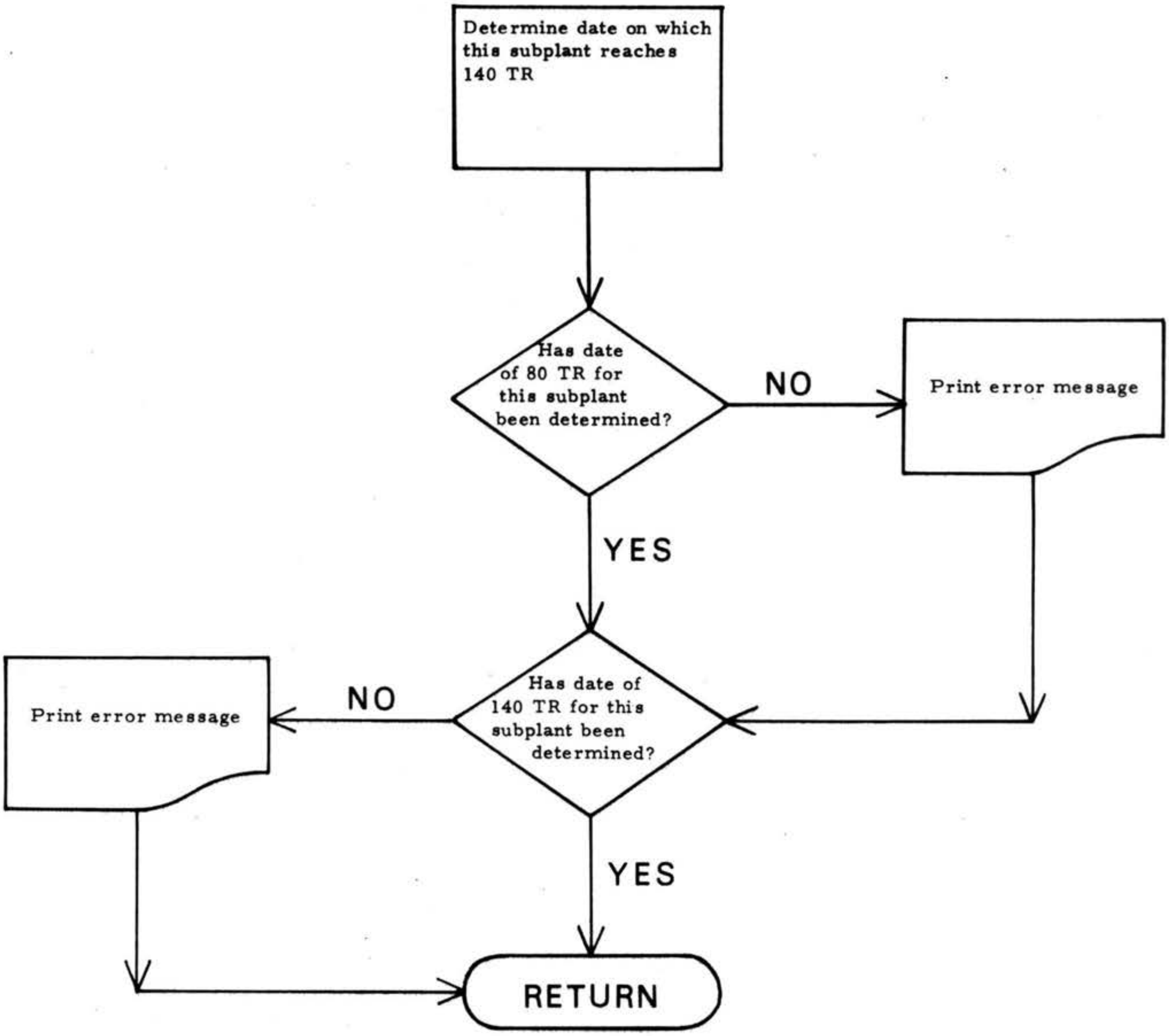
SUBROUTINE DATE



SUBROUTINE MATUR




Have increases in TR/day for each variety already been entered?
YES
NO
READ: Increase in TR/day before and after 100 TR for each variety
Were values for increase in TR/day unknown for any variety either before or after 100 TR?
NO
YES
READ where needed: Conversion factors for changing heat units accumulated/day to TR pts/day before or after 100 TR
Calculate increase in TR/day before and/or after 100 TR where necessary: Increase in TR/day = avg heat units/day / conversion factor
Determine date on which this subplant reaches 80 TR


Determine date on which this subplant reaches 140 TR
Has date of 80 TR for this subplant been determined?
NO
Print error message
YES
Has date of 140 TR for this subplant been determined?
NO
Print error message
YES
RETURN

Flowcharts for subroutines EXCEED, FACTOR, RANGE, WASTE, EROUT1 through EROUT10 and function RNDEV would be trivial or meaningless. See Appendix E for descriptions of each subroutine and a diagram of subroutine interrelationships.

APPENDIX C

Validation Data

Data used in the simulation run shown here came, in part, from the Fort Lupton Canning Company. All other values are from the literature or are estimates.

Standard deviations for maturity constants were not immediately available. Ten years of dday data on three standard freezing cultivars of peas from Asgrow Seed Company[1] provided a pooled variance of 7650.48 and from that came the value of 88 for a standard deviation.

Expected yield/A and standard deviation for yield were calculated from yield/A figures for processing peas 1959 through 1973 (2). Desired total yields were rounded figures based on expected yield x total acres. (Acreages for each cultivar were based on the approximate number of acres of each cultivar planted by the Fort Lupton Canning Company in 1977). Expected yield was used in all cases as the acreage divisor (i.e., by default).

Minimum cannery capacity was based on eight hours of processing 12,000 pounds/hour. Maximum cannery capacity assumed 16 hours at the same processing rate.

---

[1] Data provided by J. G. Lauderback, Associate Plant Breeder, Research Department, Asgrow Seed Company, Twin Falls, Idaho.

A harvest deadline of July 14 allowed for snap bean processing to begin.

Field number was kept low to minimize the amount of output in the sample.

Ell's (13) data for increase in TR points/day between 80 and 100 TR and between 100 and 140 TR was chosen because it was gathered in Colorado. This was used for three of the four cultivars in the run. Conversion factors of 6.00 and 3.00 were used for ddays/day to TR points/day between 80 and 100 TR and between 100 and 140 TR for the fourth cultivar. Since daily increases in TR readings increase as the season progresses, the first conversion factor needed to be greater than the second. In addition, values of 30 to 40 ddays/day at harvest (average) appear frequently in the literature. Combining these figures with Ell's data--i.e., 30/5 = 6 and 40/12 = 3.33--provided the conversion factors used here. The 30 to 40 ddays/day appear to be realistic on the basis of the simulation values for average ddays/day preceding and following 100 TR within the current year used.

Default values for yield at 80 and 140 TR were accepted for all four cultivars.

Key to Sample Output

(1) From left to right for each cultivar: name, maturity constant standard deviation of maturity constant, required acreage, desired total yield, mean yield/A, standard deviation of mean yield.

(2) Calendar year of current year temperature data; base temperature.

(3) Base temperature

(4) Day of first planting if not user-supplied

(5) Size (acres) of first planting

(6) First calendar year of historical temperature data; base temperature.

(7) Statements generated during the course of scheduling plantings; may or may not be of interest to user.

(8) Planting schedule

(9) Historical harvest dates

(10) Repetition of user's input. From left to right for each cultivar: name; maturity constant (heat units); standard deviation of maturity constant (heat units); total acres required for cultivar; desired yield (pounds); expected yield (pounds per acre); standard deviation of expected yield; SPS; fraction of yield at 80 TR; fraction of yield at 140 TR; TR points/day between 80 and 100 TR; TR points/day between 100 and 140 TR; ddays/increase of one TR point between 80 and 100 TR; ddays/increase of one TR point between 100 and 140 TR.

(11) Final harvest results. From left to right for each subplanting: identification number (cultivar, planting, subplanting); actual harvest date; subplanting size (acres); TR reading at harvest; subplant yield in pounds; actual ddays to 100 TR for subplant; average ddays/day for five days preceding 100 TR; average ddays/day for five days following 100 TR; date on which subplant reaches 80 TR; date on which subplant reaches 140 TR.

```
ENTER NO. OF VARS. IN SIMULATION
? 4

ENTER FOR EACH VARIETY
? "TARGET",1350,88,200,500000,2558,138
? "H482",1430,88,6,15000,2558,138                        (1)
? "CHARGER",1430,88,49,122500,2558,138
? "H432",1545,88,125,312500,2558,138

ENTER DAILY PLANT CAP IN LBS
? 96000

ENTER ACRAGE DIVISOR FOR EACH VARIETY
? 00000
? 00000
? 00000
? 00000

ENTER HARVEST DEADLINE, MONTH AND DAY
? 7,14
 75    40.                                               (2)
  40.                                                    (3)

ENTER FIRST PLANT MONTH AND DAY
? 00,00
 3    1                                                  (4)

ENTER PLANTING MULTIPLE
? 2
  74                                                     (5)
 49    40.                                               (6)
 HARVEST DATE 1ST PLANT VAR 1 IS  6   24  HUS ON THIS DATE ARE     30.25
 VAR BEING PLNTD IS  TARGET      PLNTNG NO.   2  WILL BE SCHEDULED NEXT
 PLANTING NO.   2  OF VAR  TARGET       OCCURS  3  19
 HIST HU'S ON HARVEST OF PLNTNG   2  OF VAR   1  ARE     30.25
 HARVEST DATE FOR PLNTNG   2  OF  VAR  TARGET      IS  6   24
 VAR BEING PLNTD IS  TARGET      PLNTNG NO.   3  WILL BE SCHEDULED NEXT
 PLANTING NO.   3  OF VAR  TARGET       OCCURS  4   4
 HIST HU'S ON HARVEST OF PLNTNG   3  OF VAR   1  ARE     30.25
 HARVEST DATE FOR PLNTNG   3  OF  VAR  TARGET      IS  6   24
 VAR BEING PLNTD IS  TARGET      PLNTNG NO.   4  WILL BE SCHEDULED NEXT
 PLANTING NO.   4  OF VAR  TARGET       OCCURS  4   7
 HIST HU'S ON HARVEST OF PLNTNG   4  OF VAR   1  ARE     29.21
 HARVEST DATE FOR PLNTNG   4  OF  VAR  TARGET      IS  6   25
 VAR BEING PLNTD IS  TARGET      PLNTNG NO.   5  WILL BE SCHEDULED NEXT
 PLANTING NO.   5  OF VAR  TARGET       OCCURS  4  16
 HIST HU'S ON HARVEST OF PLNTNG   5  OF VAR   1  ARE     32.27
 HARVEST DATE FOR PLNTNG   5  OF  VAR  TARGET      IS  6   27
 PLNTNG   5  OF VAR  TARGET       IS    15  ACRES     0  ACRES OF THIS VAR REMAIN TO BE PLANTED
 MATURITY CONST FOR   H482         IS  1430
 1ST PLANT VAR   2  SCHEDULED   4   5
 DATE OF 1ST PLNT VAR  H482          IS  4   5
 NEW PLANT CAP IS     57630  ACRE REQMT FOR VAR  H482          IS      6                      (7)
 1ST PLANT VAR   3  SCHEDULED   4   5
 VAR BEING PLNTD IS  CHARGER     PLNTNG NO.   2  WILL BE SCHEDULED NEXT
 PLANTING NO.   2  OF VAR  CHARGER      OCCURS  4  21
 HIST HU'S ON HARVEST OF PLNTNG   2  OF VAR   3  ARE     32.40
 HARVEST DATE FOR PLNTNG   2  OF  VAR  CHARGER     IS  7    1
 PLNTNG   2  OF VAR  CHARGER      IS    33  ACRES     0  ACRES OF THIS VAR REMAIN TO BE PLANTED
 MATURITY CONST FOR   H432         IS  1545
 1ST PLANT VAR   4  SCHEDULED   4   9
 DATE OF 1ST PLNT VAR  H432          IS  4   9
 NEW PLANT CAP IS     11586  ACRE REQMT FOR VAR  H432          IS    125
 VAR BEING PLNTD IS  H432        PLNTNG NO.   2  WILL BE SCHEDULED NEXT
 PLANTING NO.   2  OF VAR  H432         OCCURS  4  24
 HIST HU'S ON HARVEST OF PLNTNG   2  OF VAR   4  ARE     31.50
 HARVEST DATE FOR PLNTNG   2  OF  VAR  H432        IS  7    5
 VAR BEING PLNTD IS  H432        PLNTNG NO.   3  WILL BE SCHEDULED NEXT
 PLANTING NO.   3  OF VAR  H432         OCCURS  4  26
 HIST HU'S ON HARVEST OF PLNTNG   3  OF VAR   4  ARE     33.17
 HARVEST DATE FOR PLNTNG   3  OF  VAR  H432        IS  7    6
 VAR BEING PLNTD IS  H432        PLNTNG NO.   4  WILL BE SCHEDULED NEXT
 PLANTING NO.   4  OF VAR  H432         OCCURS  5   2
 HIST HU'S ON HARVEST OF PLNTNG   4  OF VAR   4  ARE     31.38
 HARVEST DATE FOR PLNTNG   4  OF  VAR  H432        IS  7    8
 VAR BEING PLNTD IS  H432        PLNTNG NO.   5  WILL BE SCHEDULED NEXT
 PLANTING NO.   5  OF VAR  H432         OCCURS  5   4
 HIST HU'S ON HARVEST OF PLNTNG   5  OF VAR   4  ARE     32.79
 HARVEST DATE FOR PLNTNG   5  OF  VAR  H432        IS  7    9
 PLNTNG   5  OF VAR  H432         IS    10  ACRES     0  ACRES OF THIS VAR REMAIN TO BE PLANTED
```

VARIETY	PLANTING NUMBER	PLANTING DATE		PLANTING SIZE
TARGET	1	3	1	74
TARGET	2	3	19	37
TARGET	3	4	4	37
TARGET	4	4	7	37
TARGET	5	4	16	15
H482	1	4	5	6
CHARGER	1	4	5	16
CHARGER	2	4	21	33
H432	1	4	9	4
H432	2	4	24	37
H432	3	4	26	37
H432	4	5	2	37
H432	5	5	4	10

⑧

VARIETY	PLANTING NUMBER	HARVEST DATE	
TARGET	1	6	24
TARGET	2	6	24
TARGET	3	6	24
TARGET	4	6	25
TARGET	5	6	27
H482	1	6	27
CHARGER	1	6	27
CHARGER	2	7	1
H432	1	7	1
H432	2	7	5
H432	3	7	6
H432	4	7	8
H432	5	7	9

⑨

```
ENTER FIELD NUMBER
? 2


ENTER 2 RANDM NO. SEEDS:ODD INTGRS FRM 1 TO 1048576
? 3,13


ENTER INCREASE IN TR/DAY BEFORE 100 TR AND AFTER FOR VAR    1
? 5,12


ENTER INCREASE IN TR/DAY BEFORE 100 TR AND AFTER FOR VAR    2
? 5,12


ENTER INCREASE IN TR/DAY BEFORE 100 TR AND AFTER FOR VAR    3
? 5,12


ENTER INCREASE IN TR/DAY BEFORE 100 TR AND AFTER FOR VAR    4
? 0,0
 ENTER CONVERSION FACTOR FOR HU'S/DAY TO TR POINTS/DAY BEFORE 100 TR FOR VAR    4
? 6.00
 ENTER CONVERSION FACTOR FOR HU'S/DAY TO TR POINTS/DAY AFTER 100 TR FOR VAR    4
? 3.00
OCURRENT HARVEST DATE FOR SUBPLNT   1  OF PLNTNG   5  OF VAR   4  EXCEEDS HARVEST DEADLINE BY     1  DAYS


ENTER MAX AND MIN DAILY CANNERY CAPACITY (LBS)
? 192000,96000


ENTER PERCENT YIELD (DECIMAL) AT 80 AND 140 TR FOR EACH VAR
? 0.00,0.00
? 0.00,0.00
? 0.00,0.00
? 0.00,0.00
                HARVEST DATE          LBS BELOW MIN CANNERY CAPACTY
                     6  28                       50309
                     6  29                       69920
                     7   1                       50429
                     7   3                       46518
                     7   4                       66826
                     7   6                       53015
                     7   7                       19083
                     7   8                       49890
                     7   9                        6183
                     7  12                       48375
                     7  13                       35807
                     7  15                       82970
```

NAME	MATUR CONST HUS	SD MAT CONST HUS	ACRES REQRD	DISIRED YIELD LBS	EXPECTD YIELD PPA	SD EXPECTD YIELD	STRD PLNT SIZE	YIELD AT 80TR	YIELD AT 140TR	TR DAY BEFORE 100TR	TR DAY AFTER 100TR	EARLY HU TR CONVER	LATE HU TR CONVER
TARGET	1350	88	200	500000	2558	138	37	.5600	1.3600	5	12	0.00	0.00
H482	1430	88	6	15000	2558	138	37	.5600	1.3600	5	12	0.00	0.00
CHARGER	1430	88	49	122500	2558	138	37	.5600	1.3600	5	12	0.00	0.00
H432	1545	88	125	312500	2558	138	37	.5600	1.3600	5	11	6.00	3.00

(10)

NO. OF CVS. IN SIMULATION: 4
DAILY CANNERY CAPACITY: 96000
MAX CANNERY CAPACITY: 192000
MIN CANNERY CAPACITY: 96000
HARVEST DEADLINE: 7 14
PLANTING MULTIPLE FOR 1ST PLANT: 2
FIELD NUMBER: 2

IDENT CV,P,SP	HARVEST DATE	FIELD SIZE(A)	TR AT HARVEST	YIELD LBS	HUS TO 100TR	HUS DAY BEFORE 100TR	HUS DAY AFTER 100TR	DATE OF 80TR	DATE OF 140TR
10101	6 30	37.00	100	99719.01	1437.43	31.10	35.30	6 26	7 3
10102	6 27	37.00	100	93602.56	1332.02	26.80	34.60	6 23	6 30
10201	7 2	18.50	100	50530.24	1460.55	34.40	33.90	6 28	7 5
10202	6 27	18.50	100	45919.08	1301.61	26.80	34.60	6 23	6 30
10301	6 28	18.50	100	45691.94	1293.78	29.50	35.30	6 24	7 1
10302	7 2	18.50	100	49846.26	1436.97	34.40	33.90	6 28	7 5
10401	7 3	18.50	100	49482.04	1424.42	34.60	32.60	6 29	7 6
10402	6 27	18.50	100	44346.93	1247.42	26.80	34.60	6 23	6 30
10501	7 7	7.50	100	21487.17	1545.74	34.30	29.50	7 3	7 10
10502	7 1	7.50	100	19111.98	1343.79	31.50	34.30	6 27	7 4
20101	7 2	11.00	100	30128.42	1437.64	34.40	33.90	6 28	7 5
20102	7 1	11.00	100	26460.23	1376.18	31.50	34.30	6 27	7 4
30101	7 4	11.00	100	29174.50	1490.09	35.30	32.30	6 30	7 7
30102	6 29	11.00	100	26080.11	1310.70	30.50	35.70	6 25	7 2
30201	7 6	16.50	100	42985.59	1460.09	35.30	30.60	7 2	7 9
30202	7 2	16.50	100	40063.13	1347.14	34.40	33.90	6 28	7 5
40101	7 7	2.00	100	5247.23	1586.84	34.30	29.50	7 3	7 10
40102	7 7	2.00	100	5186.55	1567.49	34.30	29.50	7 3	7 10
40201	7 8	18.50	100	46110.84	1503.22	33.90	29.60	7 4	7 11
40202	7 7	18.50	100	44997.21	1464.83	34.30	29.50	7 3	7 10
40301	7 12	18.50	100	47625.29	1555.42	30.60	33.40	7 8	7 15
40302	7 9	18.50	100	45747.39	1490.69	32.60	30.10	7 5	7 12
40401	7 9	18.50	100	44070.52	1432.89	32.60	30.10	7 5	7 12
40402	7 13	18.50	100	47578.49	1553.81	29.50	34.30	7 9	7 16
40501	7 15	5.00	100	13030.25	1575.64	30.10	34.10	7 11	7 18
40502	7 13	5.00	100	12615.46	1522.74	29.50	34.30	7 9	7 16

(11)

DATE OF FIRST HARVEST: 6 27
DATE OF LAST HARVEST: 7 15
TOTAL LBS HARVESTED AT 80 TR: 0
TOTAL LBS HARVESTED AT 100 TR: 1026828
TOTAL LBS HARVESTED AT 140 TR: 0
TOTAL LBS WASTE DURING SEASON: 0
TOTAL LBS SLACK DURING SEASON: 579325
TOTAL LBS HARVESTED FOR SEASON: 1026828

APPENDIX D

Guide to Technical Program Structure

Execution of program PSIM with five cultivars and a total of 28 subplantings requires approximately 80 K storage and 10 seconds.

The time-sharing version of program PSIM references six files: TAPE1, TAPE2, TAPE4, TAPE6, TAPE7 and BOSS. The batch version differs in three ways: 1) commands contained in BOSS constitute the first logical record in the job deck; 2) the FORTRAN program may be included as the second logical record of the deck instead of being stored as TAPE7; 3) non-temperature data required by the simulation is punched on cards and accessed as TAPE5.

TAPE1 and TAPE2 contain historical and current year temperature data for both modes. Likewise, TAPE6 is the line printer for both modes. In time-sharing mode, TAPE7 is the executable FORTRAN program and BOSS is the procedure file containing control statements necessary to access all other files, compile, load and execute the simulation. TAPE4 is a direct access scratch file used to store elements of array AWS for all subplants.

Control statements required to run PSIM will vary from installation to installation; control statements required at Colorado State University on the CDC 171 with the NOS version 1 operating system are included below to give programmers at other installations an idea of the analogous commands for other machines.

Program card. The program card is unique to CDC systems and not ANSI FORTRAN. It is the first card of any FORTRAN program. Its function is to specify system input, output and tape or disk files. Program cards for the time-sharing and batch modes of PSIM differ slightly: in batch mode the CDC system input file is the card reader and is specified as such to the program by use of the parameter INPUT. (The expression TAPE5 = INPUT equivalences the card reader with TAPE5). In time-sharing mode, a character list input format, specified by the expression INPUT = /72, allows a user to input data with commas for delimiters. This is not ANSI FORTRAN but was included to obviate the need for input formatting.

The CDC output file is the line printer and is specified by the parameter OUTPUT. The expression TAPE6 = OUTPUT equivalences the line printer with TAPE6. The program card for PSIM, batch mode, is:

```
PROGRAM PSIM(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=
    OUTPUT,TAPE1,TAPE2)
```

For time-sharing mode the program card is:

```
PROGRAM PSIM(INPUT=/72,OUTPUT,TAPE6=OUTPUT,
    TAPE1,TAPE2)
```

TAPE1 and TAPE2. Temperature data may be put on files DATAH and DATAC (which are then equivalenced to TAPE1 and

TAPE2 by procedure file BOSS or by control statements within a deck) using the following deck structure:

```
JOB.
USER,_____.
DEFINE,DATAH.
COPYCR,INPUT,DATAH.
DEFINE,DATAC.
COPYCR,INPUT,DATAC.
7/8/9
historical temperature data
7/8/9
current year temperature data
6/7/8/9
```

The first two cards listed above are job and user cards required by the CDC system. The 7/8/9 cards (multipunched in column one) are logical record separators, while the 6/7/8/9 card (also multipunched in column one) indicates end of file. See user documentation for temperature data formats.

Procedure file listing. Job and user cards are not required in the time-sharing procedure file. In time-sharing mode the simulation is run by entering the command, CALL, BOSS. The FILE statement is required to create TAPE4, the direct access file, and contains a series of parameters which determine file organization. Deck structure for the batch run assumes that temperature data has already been copied to DATAH and DATAC.

Procedure file, time-sharing mode, with PSIM stored as TAPE7:

```
ATTACH, TAPE7.
ATTACH, TAPE1=DATAH.
ATTACH, TAPE2=DATAC.
$FTN, A, TS, T, X, L=0, SEQ, B=LGO, I=TAPE7, REW.
FILE, TAPE4, FO=DA, RT=F, FL=120, MRL=120, MNR=120, KL=
    10, HMB=5.
LDSET, FILES=TAPE4.
LGO.
FORCE.
EXIT.
DAYFILE, L.
REPLACE, L.
REWIND, ZZZZZEF.
COPYSBF, ZZZZZEF, L.
REPLACE, L.
```

Control cards and deck structure for batch mode:

```
JOB.
USER, _____.
ATTACH, TAPE1=DATAH.
ATTACH, TAPE2=DATAC.
$FTN.
FILE, TAPE4, FO=DA, RT=F, FL=120, MRL=120, MNR=120, KL=
    10, HMB=5.
LDSET, FILES=TAPE4.
LGO.
FORCE.
EXIT.
REWIND, ZZZZZEF.
COPYSBF, ZZZZZEF, OUTPUT.
7/8/9
FORTRAN program
7/8/9
non-temperature data
6/7/8/9
```

<u>TAPE4.</u> This is a direct access (DA) scratch file. Although similar files exist for most systems, packages differ between installations. The FORTRAN commands used in program PSIM to access TAPE4 are listed with brief explanations to give programmers an

idea of the analogous commands which may be needed at other installations.

CALL FILEDA(TAPE4,3LLFN,5LTAPE4). This call establishes the file information table (FIT), a user-declared, 35 word array used for communication between DA routines. TAPE4 is delcared as the name of the FIT.

CALL OPENM(TAPE4,3LNEW) and CALL OPENM(TAPE4, 3LI-O). "OPENM" is the routine which opens a file, making it accessible to the program. "3LNEW" signifies a creation run; when replaced by "3LI-O," it means that the file will be used for both input and output.

CALL PUT(TAPE4,AWS,0,0,0,0,EROUT__). This call writes the data record, stored in array AWS, to the DA file whose FIT is named TAPE4. (The first element of AWS is an identifying number, or key). The four zeroes indicate inapplicable parameters. EROUT__ is the routine called if an error occurs in writing the record to the DA file.

CALL REPLC(TAPE4,TWS,120,0,0,0,EROUT__). This call replaces one 120 character record with another. Array TWS is the same size as AWS and its first element contains the key for the record to be replaced. Zeroes indicate inapplicable parameters. EROUT__ is the routine called if an error occurs during record replacement.

CALL GET(TAPE4,AWS,AKEY,0,0,0,EROUT__). This call will read one record's worth of data from the DA file TAPE4. "AKEY" provides the identifying number which determines which record will be read. Subroutine EROUT__ will be called if an error occurs in data retrieval.

CALL CLOSEM(TAPE4). This call deactivates the file associated with the FIT named TAPE4.

User Documentation

Program PSIM may be operated in either batch or time-sharing mode. Two basic types of data are required by the simulation. One is temperature data for the area simulated. Historical or long term temperature data is required in order to schedule plantings; temperature for the current year is required to determine harvest dates. Both sets of data are input in the same format and accessed as needed by program PSIM. A user may input either soil or air temperatures or a combination of the two: data is read from cards into storage and is then accessed during program execution. (See Guide to Technical Program Structure).

Formatting for both sets of temperature data appears below, followed by instructions for remaining data. All data required by the simulation is listed in Table 1. Documentation for non-temperature data includes formatting instructions for a punched deck as well as

Table 1. User-supplied information for program PSIM. (a) Optional information; default values provided. (b) See user documentation for explanatory notes. (c) Required only if increase in tenderometer/day before and/or after 100 tenderometer is not provided.

Temperature Data:

Base temperature of crop
For each day of current year from March 1 to September 30: maximum and minimum temperature
For each day of X years (historical data) from March 1 to September 30: maximum and minimum temperature

Data for Each Cultivar in Simulation:

Name
Maturity constant mean
Maturity constant standard deviation
Total acres required (a)
Total desired yield in pounds
Expected mean yield in pounds/acre
Standard deviation for expected yield
Acreage divisor (b)
Increase in tenderometer/day before 100 tenderometer and after (a)
Conversion factor for heat units/day to tenderometer points/day before 100 tenderometer and after (c)
Per cent yield at 80 and 140 tenderometer (a)

General Information:

Number of cultivars in simulation
Daily cannery capacity in pounds
Maximum and minimum daily cannery capacity
Harvest deadline
Date of first plant (a)
Planting multiple for first planting
Field number (b)
Two random number seeds

user response samples for a time-sharing job. The time-sharing version of PSIM allows a user to input data, in response to computer prompters, with no formatting directives--i.e., when more than one piece of information is required in response to a prompter commas are the only necessary delimiters. System control commands needed to make the program available for batch and interactive modes are given in the Guide to Technical Program Structure.

In the documentation presented here, computer prompters follow the symbol "C" in upper case letters. Explanatory notes with sample user responses follow the symbol "U". The letters "CR" which follow interactive user responses indicate that the carriage return key must be pressed after data entry. To clarify column conventions and field lengths for a batch job, column numbers head the sample responses. FORTRAN format statements are included for users acquainted with that language.

Interactive users should examine each response before pressing carriage return, since corrections can easily be made at this time: backspacing "erases" all characters passed over. (The user must therefore re-enter all input over which he has backspaced, including characters correctly entered on a previous attempt). Once the carriage return has been pressed, a new run must be initiated to correct input.

The computer responds in various ways to input different from what it expects. If character data is entered when numbers are expected, the computer responds by printing the field in which the incorrect data was detected along with the message: ERROR, RETYPE RECORD AT THIS FIELD. If zeroes are valid user responses (i.e., to trigger default values), the zeroes must be entered in the correct field before pressing carriage return (failure to enter zeroes results in the message, END OF INFORMATION ENCOUNTERED and the run is automatically terminated).

Although what follows is designed to be sufficient documentation for simulation use, explanatory notes are kept to a minimum for the sake of clarity. Users are referred to both the program description and the discussion of PSIM for a thorough explanation of the logic used and assumptions made by the model.

Temperature Data Documentation

User supplied temperature data is assumed by program PSIM to be complete: missing values are not discerned by the program and will cause inaccurate calculations of daily means and accumulated ddays. Users may therefore wish to verify their data before program execution.

Long term and current year data are identical in format. The following descriptions can therefore be used for both sets of data.

CARD 1

Data required:

LAST two digits of the year in which temperature data begins (integer, minimum value 0, maximum value 99).

BASE temperature, ^{o}F, of crop being scheduled (real number, minimum value 1.0, maximum value 99.0).

Sample input: for data beginning in 1950, for a crop whose base temperature is $40^{o}F$.

Cols.	3-4	7-10
	50	40.0

Begin in column: 3

Field lengths: 2, skip 2, 4

FORTRAN format: 1X,I2,2X,F4.0

CARDS 2 through n

Where n is the last data card. Cards must have all fields filled. Data for two different months, but not different years, may appear on one card. Data must run from March 1 through September 30. Five day's data appears on each card and the final card for each year will have zeroes in the last five fields (i.e., what would be September 31).

Data required:

MAXIMUM temperature in degrees F (integer, minimum value -999, maximum value 9999).

MINIMUM temperature in degrees F (integer, minimum value -999, maximum value 9999).

FINAL two digits of the year (integer, minimum value 0, maximum value 99).

MONTH in numerical form (integer, minimum value 3, maximum value 9).

DAY (integer, minimum value 1, maximum value 31).

Sample input: for the first two and the last repetition of daily format on two cards (the first and last cards for the year 1950). Raw data is given in Table 2; data placed in the proper fields appear in Table 3. Note the following conventions:

Begin in column: 1

Field lengths: 4, 4, skip 1, 2, 2, 2, skip 1 (repeat five times/card)

FORTRAN format: 5(2I4,1X,3I2,1X)

All fields are right justified.

Documentation for Non-temperature Data

C: ENTER NO. OF VARS. IN SIMULATION

U: Program PSIM will handle a maximum of ten cultivars on any given run.

Sample input: for two cultivars.

Table 2. Temperature data for program PSIM. Three days' data from the first data card for 1950, two days' from the last card for 1950.

Month	Day	Year	Max. Temp. °F	Min. Temp. °F
March	1	1950	49	28
	2	1950	60	27
	5	1950	63	34
September	27	1950	89	65
	28	1950	92	68

Table 3. Temperature data formats for data cards. The last five fields of the last card for each year will be zero filled.

Cols.	1-4	5-8	10-11	12-13	14-15	17-20	21-24	26-27
	max	min	yr	mo	day	max	min	yr
Card 1 (March)	49	28	50	3	1	60	27	50
Card n (Sept)	89	65	50	9	27	92	68	50

Cols.	28-29	30-31	...	65-68	69-72	74-75	76-77	78-79
	mo	day		max	min	yr	mo	day
Card 1 (March)	3	2	...	63	34	50	3	5
Card n (Sept)	9	28	...	0	0	0	0	0

Batch mode	Time-sharing mode
Cols. 1-2	User response: 2 (CR)
2	
Default value: none Begin in column: 1 Field length: 2 FORTRAN format: I2 The number is right justified.	Default value: none Field length: 2

C: ENTER FOR EACH VARIETY

U: Seven initial items are required for each cultivar in the simulation. Numbers are all integer, no decimals allowed. Data must be entered in the order shown below, all items for one cultivar on a single line. Cultivars will be planted in the order in which they are input; the user will therefore generally input cultivars in order of ascending maturity constants.

Sample input: for two cultivars

Batch mode

Cols. 1-10	12-18	20-26	28-34	36-42	44-50	52-58
TARGET	1350	88	200	500000	2558	138
H482	1430	88	6	15000	2558	138

Time-sharing mode

"TARGET",1350,88,200,500000,2558,138 (CR)

"H482",1430,88,6,15000,2558,138 (CR)

Note: The first item of the input list, cultivar name, must be enclosed in quotation marks. Items from the above sample input are listed in Table 4. They appear in order from left to right, with relevant parameters and comments. The FORTRAN format for each line is: A10, 1X, 5(I7, 1X), I7.

C: ENTER DAILY PLANT CAP IN LBS

U: This figure is intended to reflect an average daily processing capacity at the cannery. Maximum capacity (e.g. capacity with an additional work shift) and minimum capacity (i.e., a volume below which slack exists in the cannery operation) are input later in the simulation. The maximum pounds/day allowed here is 9999999 pounds.

Sample input: for a daily capacity of 96,000 pounds.

Batch mode	Time-sharing mode
Cols. 1-7	User response: 96000 (CR)
96000	
Default value: none Begin in column: 1 Field length: 7 FORTRAN format: I7 The number is right justified.	Default value: none Field length: 7

C: ENTER ACREAGE DIVISOR FOR EACH VARIETY

U: This response allows the user to determine how many acres of a given cultivar will be planted on any one planting day (i.e., a

Table 4. User-supplied information for each cultivar in the simulation, with input format and descriptive comments.

Item	Cols.	Comments
Variety Name	1-10	Default value: none. The name may consist of letters, numbers or a combination of the two. It need not use, but may not exceed, 10 columns in length. The name is left justified.
Maturity Constant	12-18	Default value: none. This is the mean number of degree days to maturity (100 tenderometer). Units are degree days. The number is right justified.
Standard Deviation for Maturity Constant	20-26	Default value: none. Units are degree days. The number is right justified.
Acres Required	28-34	Default value: if user inputs zero, an acre requirement will be calculated using the formula: $\text{Acres Required} = \frac{\text{Desired yield (lbs.)}}{\text{Expected mean yield (lbs./acre)}}$ Units for input are acres. The number is right justified.
Total Desired Yield	36-42	Default value: none. Units are pounds. The number is right justified.
Expected Yield/Acre	44-50	Default value: none. This is mean yield/acre. Units are pounds. The number is right justified.
Standard Deviation for Expected Yield	52-58	Default value: none. This deviation is for yield/acre. Units are pounds. The number is right justified.

"standard planting size" (SPS) for each cultivar). The largest admissable divisor is 99999. A reasonable user response might be the expected yield for a cultivar, since SPS is computed in the following manner:

$$\text{SPS for cv X} = \frac{\text{daily plant capacity (lbs.)}}{\text{acreage divisor for cv X}}$$

Expected yield is, in fact, the default value assigned by the program if user input for a given acreage divisor is zero. There may be, however, situations in which the user prefers a different divisor. One such situation would occur if the user wished planting size to reflect an unusual value or cost for a given cultivar. For example, if the price of seed for one cultivar was much higher than for other cultivars, using a larger acreage divisor would minimize the chances of losing raw product at harvest time due to bunching. Similarly, a lower fixed cost for a given cultivar might suggest use of a high acreage divisor to minimize the chances of cannery slack at harvest. Also, if the user knows the cost of bunching and slack at the cannery he might vary acreage divisors to reflect a difference in those costs.

<u>Sample input</u>: for two cultivars, one of which will take the default value of its mean yield.

Batch mode	Time-sharing mode
Cols. 1-5	User response: 00000 (CR)
00000	1500 (CR)
1500	
Default value: if user inputs zeroes, expected yield/A is used to compute SPS Begin in column: 1 Field length: 5 FORTRAN format: I5 The number is right justified.	Default value: same as in batch mode Field length: 5

C: ENTER HARVEST DEADLINE, MONTH AND DAY

U: Input the last day on which harvest is desired, month first. Since temperature data handled by PSIM runs from March 1 through September 30, values accepted by the computer for months are 3 through 9, for days, 1 through 31.

Sample input: for harvest deadline of September 14.

Batch mode	Time-sharing mode
Cols. 1-2 6-7	User response: 9,14 (CR)
9 14	
Default value: none Begin in column: 1 Field length: 2, skip 3, 2 FORTRAN format: I2, 3X, I2 The numbers are right justified.	Default value: none Field length: 2,2

C: ENTER FIRST PLANT MONTH AND DAY

U: A canner may choose as the first day of planting that day when the mean soil or air temperature equals or exceeds the crop's base temperature. The default date provided by the program is determined on that basis, using temperature data provided by the user. If the user wishes a different date, he may input that date here. Values accepted by the computer for months are 3 through 9, for days, 1 through 31.

Sample input: for user-supplied first planting date of March 1.

Batch mode	Time-sharing mode
Cols. 1-2 6-7	User response: 3,1 (CR)
3 1	
Default value: if user inputs zeroes for both day and month, the first planting will be made on the first day of the current year when mean temperature equals or exceeds the crop's base temperature. Begin in column: 1 Field length: 2, skip 3, 2 FORTRAN format: I2, 3X,I2 The numbers are right justified.	Default value: same as in batch mode Field length: 2,2

C: ENTER PLANTING MULTIPLE

U: The first planting made will be the SPS of the first cultivar multiplied by this number. Maximum admissable multiple is 9, minimum is 1.

Sample input: for a first planting which is double the SPS.

Batch mode	Time-sharing mode
Col. 1	User response: 2 (CR)
2	
Default value: none Begin in column: 1 Field length: 1 FORTRAN format: I1	Default value: none Field length: 1

Note: If size of the first planting (i.e., planting multiple x SPS for cultivar one) exceeds the acre requirement for cultivar one, program PSIM proceeds to reduce the planting multiple stepwise by one until first planting size is less than or equal to the first cultivar's acre requirement. A message will inform the user of the error (see error diagnostics in this appendix) and final output will print the value used as a planting multiple. If the smallest multiple of SPS--i.e., SPS x 2--also exceeds acre requirements for cultivar one, the simulation terminates and a "fatal error" message is printed. (Program PSIM will not use a multiple of one unless the user inputs it).

C: ENTER FIELD NUMBER

U: The purpose of the field number in PSIM is to make the simulation as realistic as possible in terms of its probablistic calculations. It is unlikely that all acres planted on a given day will reach maturity at the same time. It is also unlikely that yields will be the same for all acres planted on one day. The error caused by assuming uniformity in these two characteristics will increase as the size of daily plantings increases. The field number allows a user to reduce this error in the following manner. Degree days to maturity (100 TR) and yield are treated as random variables. Both values are generated by sampling a normal probability distribution whose mean and standard deviation have been supplied by the user at the beginning of the simulation. The field number actually represents the number of times the distributions for these two characteristics are sampled. If, therefore, 25 acres are planted on one day and the field number is five, five values for degree days to 100 TR and five values for yield will be generated for that planting. In this case it may be visualized as five different fields or subplantings (five acres/subplanting), each with its own "dday to 100 TR" requirement, and its own yield. Thus, the larger the field number, the more variation accounted for in terms of yield and maturity date. Users with large plantings (due either to a large daily cannery capacity or use of a small acreage divisor) may wish to increase the field number in order to get more realistic yield statistics and harvest dates. The maximum admissable field number in PSIM is 20.

Sample input: for a field number of five.

Batch mode	Time-sharing mode
Cols. 1-2	User response: 5 (CR)
5	
Default value: none Begin in column: 1 Field length: 2 FORTRAN format: I2 The number is right justified.	Default value: none Field length: 2

C: ENTER 2 RANDOM NO. SEEDS; ODD INTEGERS FRM 1 TO 1048576

U: Program PSIM uses a random number generator in sampling probability distributions for ddays to maturity and yields. Two odd integers ("seeds") are required to trigger this generator for a given simulation run. The same series of random numbers will be generated each time the same two seeds are provided. New seeds should therefore be supplied if other parameters remain unchanged and more than one run is desired.

Sample input: for the random number seeds of 3 and 32401.

Batch mode	Time-sharing mode
Cols. 1-7 11-17	User response: 3, 32401 (CR)
3 32401	
Default value: none Begin in column: 1 Field length: 7, skip 3, 7 FORTRAN format: I7, 3X, I7 The numbers are right justified.	Default value: none Field length: 7, 7

C: ENTER INCREASE IN TR/DAY BEFORE 100TR AND AFTER FOR VAR____

U: The user may supply two values for each cultivar: the first value is the number of TR points the cultivar accumulates/day before reaching 100 TR; the second value is the number of TR points accumulated/day after reaching 100 TR. If such figures are not available, in one or both cases, the user may input zeroes for both values and will then be asked to provide the number of heat units that that cultivar accumulates before increasing one TR point. (See next computer response). Program PSIM will then compute values for increase in TR/day based on average heat unit accumulation/day before and after harvest date in the current year (see program description for further explanation). Maximum values for increase in TR/day before and after 100 TR are 99.

Sample input: for a cultivar in which increase in TR/day before 100 TR is five, and where increase after 100 TR is unknown.

Batch mode	Time-sharing mode
Cols. 1-2 6-7 5 00	User response: 5,00 (CR)
Default value: if user inputs zeroes for any case, he must supply data for the next computer response. Begin in column: 1 Field length: 2, skip 3, 2 FORTRAN format: I2, 3X, I2 The numbers are right justified.	Default value: same as in batch mode Field length: 2,2

C: ENTER CONVERSION FACTOR FOR HU'S/DAY TO TR PTS/DAY BEFORE 100 TR FOR VAR____

U: This prompter will appear only if the user has input zeroes in response to the first half of the previous input request. (If using a punched deck, the user will therefore only need data for this information if he did not provide all information in the last request. If zeroes were input for more than one cultivar, cards with conversion factors should appear in the same order the cultivars are planted, one conversion factor/card). This prompter requests the number of heat units a cultivar accumulates before 100 TR before an increase of one TR point. Given this information, the program computes increase in TR/day before 100 TR in the following manner:

$$\text{increase in TR/day before 100 TR} = \frac{\text{avg. heat units/day before 100 TR}}{\text{heat units accumulated/unit increase in TR}}$$

Input here must be a real number with two places to the right of the decimal. Maximum value is 9.99.

Sample input: for a cultivar whose increase in TR/day before 100 TR was unknown.

Batch mode	Time-sharing mode
Cols. 1-4 6.00	User response: 6.00 (CR)
Default value: none Begin in column: 1 Field length: 4 FORTRAN format: F4.2 The number is right justified.	Default value: none Field length: 4

C: ENTER CONVERSION FACTOR FOR HU'S/DAY TO TR PTS/DAY AFTER 100 TR FOR VAR____

U: This prompter occurs only if the increase in TR/day after 100 TR for a given cultivar was unknown (input as zeroes). Batch users will need data here only if that is the case. (If zeroes were input for more than one cultivar, cards with conversion factors should appear in the same order the cultivars are planted, one conversion factor/ card). This prompter requests the number of heat units a cultivar accumulates after 100 TR before an increase of one TR point. Given this information, the program computes increase in TR/day after 100 TR by the formula:

$$\text{increase in TR/day after 100 TR} = \frac{\text{avg. heat units/day after 100 TR}}{\text{heat units accumulated/unit increase in TR}}$$

Input here must be a real number with two places to the right of the decimal. Maximum value is 9.99.

Sample input: for a cultivar whose increase in TR/day after 100 TR was unknown.

Batch mode	Time-sharing mode
Cols. 1-4	User response: 3.00 (CR)
3.00	
Default value: none Begin in column: 1 Field length: 4 FORTRAN format: F4.2 The number is right justified.	Default value: none Field length: 4

C: ENTER MAX AND MIN DAILY CANNERY CAPACITY (LBS.)

U: This input allows for the possibility of either an extra working shift or a shorter shift at the cannery. At the beginning of the simulation, the user inputs daily cannery capacity. If the input given is inflexible, the user need only supply the same figures twice more at this point. In the more usual case, where there is flexibility, new capacities can be entered here. Values are integer and the maximum in both cases is 999999 (pounds).

Sample input: for a maximum daily cannery capacity of 192000 pounds and a minimum capacity of 96000 pounds.

Batch mode	Time-sharing mode
Cols. 1-6 12-17	User response: 192000,96000 (CR)
192000 96000	
Default value: none Begin in column: 1 Field length: 6, skip 5, 6 FORTRAN format: I6, 5X,I6 The numbers are right justified.	Default value: none Field length: 6,6

C: ENTER PER CENT YIELD (DECIMAL) AT 80 AND 140 TR FOR EACH VAR

U: If yield on a given harvest day exceeds maximum cannery capacity, program PSIM tries to reschedule part of that day's yield. The first attempt is to reschedule a portion of harvest at 80 TR. If this is impossible because doing so exceeds maximum cannery capacity on some other day, an attempt is made to reschedule a portion of the harvest at 140 TR. In both types of rescheduling, yield for the rescheduled acreage will clearly change. This program assumes that yield at 100 TR is 100% and that yield at a different TR will be some per cent of yield at 100 TR--yield at 80 TR will be less, yield at 140 TR will be greater. The user may supply his own figures here or accept the default values of 56% at 80 TR and 136% at 140 TR. Values are input as decimal fractions, two places to the right of the decimal point. Minimum admissable value at 80 TR is therefore 0.01, maximum at 140 TR is 9.99.

Sample input: for 60% yield at 80 TR and 120% yield at 140 TR.

Batch mode	Time-sharing mode
Cols. 1-6 10-15 .60 1.20	User response: .60,1.20 (CR)
Default values: .56 and 1.36 Begin in column: 1 Field length: 4, skip 3, 4 FORTRAN format: F4.2, 3X, F4.2 The numbers are left justified with two places to the right of the decimal.	Default values: same as in batch mode Field length: 4,4

Error Diagnostics

Two types of error messages are included in program PSIM. The first type, errors one through eight, are followed by their cause. Not all of these errors result in termination of the run (those which do are followed here by (F), i.e., "fatal"), but non-fatal errors do affect program output. The second set of error messages given here pertains to the programming structure of PSIM. Should any of these messages appear, the program listing should be examined by a competent FORTRAN programmer.

1. ACRES REQUIRED FOR VAR____EXCEED SIMULATION LIMIT (F)

Program PSIM will schedule a maximum of 30 plantings of each cultivar in the simulation. This message is generated if the total acres required of a given cultivar cannot be scheduled in 30 plantings. An increase in SPS (by decreasing the acreage divisor for this cultivar) may correct this error.

2. CURRENT HARVEST DATE FOR SUBPLNT____OF PLNTNG____ OF VAR____EXCEEDS HARVEST DEADLINE BY____DAYS

This message is generated whenever the date on which a subplant reaches 100 TR exceeds the harvest deadline.

3. FIRST PLANT CANNOT BE SCHEDULED (F)

If the smallest multiple of the first cultivar's SPS (i.e., twice the SPS) exceeds the total number of acres required for that cultivar, the first planting is not scheduled and the simulation terminates. This program exit can be by-passed if the user supplies a planting multiple of one.

4. FIRST PLNT OF____NOT SCHEDULED (F)

This message is generated for the same reason as message 6, with one difference: the cultivar which cannot be scheduled is the second cultivar of which a partial planting must be made in order to fill cannery capacity on a given day. This situation indicates small acreage requirements for two or more cultivars and a wide spread in maturity constants for those cultivars.

5. FIRST PLNT OF____NOT SCHEDULED YET (F)

The size of scheduled plantings is determined by daily cannery capacity and expected yield of the cultivar being planted. If cannery capacity cannot be filled by the number of acres left of a given cultivar, a "partial" planting is made of the next cultivar. In this case, planting date for the new cultivar is scheduled in such a way that its historical harvest date will be the same as that of the last planting of the previous cultivar. If total ddays accumulated at harvest of the last planting of the previous cultivar are less than the maturity constant of the new cultivar, a planting cannot be scheduled and this message is generated. (See Results and Discussion for description of how this problem may occur).

6. INSUFFICIENT TEMP DATA AVAILABLE; FIRST PLNT OF VAR____CANNOT BE SCHEDULED

This message occurs if a partial planting would have to be made prior to March 1 for harvest on the same day as a previous planting of another cultivar. Error message 4 or 5 will also be generated in this case.

7. SUBPLNT____OF PLNTNG____FOR VAR____NOT SCHEDULED; INSUFFICIENT TEMP DATA

This message is generated if the number of accumulated ddays required to bring a subplant to 100 TR (determined stochastically) is greater than the total accumulated ddays on September 30 of the current year. Such a subplanting would reach 100 TR sometime after that date.

8. 1ST PLANT MULTIPLE TOO LARGE; FIRST PLANT SIZE EXCEEDS TOTAL ACRE REQMNT FOR VAR____

This message appears if the user supplied planting multiple requires a larger acreage than the total number of acres required for the first cultivar planted. Program PSIM will print this message before attempting to reduce the planting multiple stepwise in decrements of one until the first planting can be scheduled. If the simulation runs normally after this message, the user should check output to determine actual size of first planting.

Program Structure Errors

DATE OF 80 TR FOR SUBPLNT____OF PLNTNG____OF VAR____ NOT DETERMINED

DATE OF 140 TR FOR SUBPLNT____OF PLNTNG____OF VAR____ NOT DETERMINED

HU'S NOT AVG'D FOR 5 DAYS FOLLOWING HARVEST OF VAR____ PLNTNG____SUBPLNT____

HU'S NOT AVG'D FOR 5 DAYS PRECEDING HARVEST OF VAR____
PLNTNG____SUBPLNT____

ERROR IN DA FILE INPUT BETWEEN STMTS____AND____; KEY IS
____

ERROR IN RETRIEVAL FROM DA FILE BETWEEN STMTS____AND
____; KEY IS____

ERROR IN RETRIEVAL FROM DA FILE IN SUBROUTINE RANGE;
KEY IS____

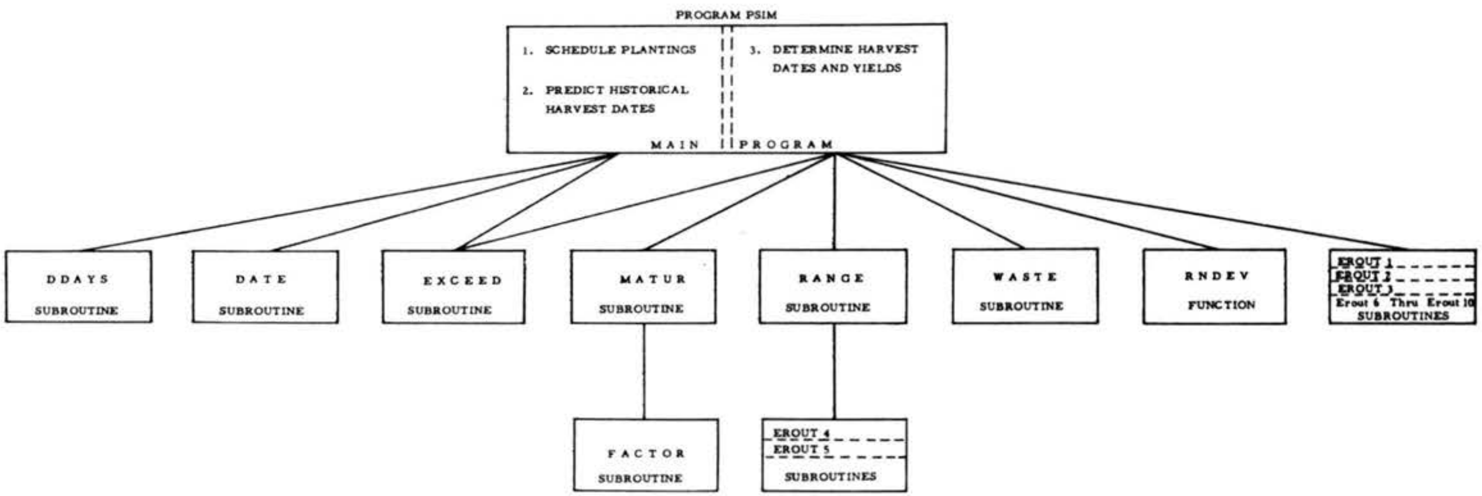
ERROR IN REPLACEMENT OF DA RECORD BETWEEN STMTS____
AND____; KEY IS____

The first four errors in program structure are not fatal but indicate a programming limitation. For the last four errors listed, all of which pertain to the direct access file system used to store contents of the working storage array AWS, the additional information below will also appear:

ERROR(EROUT__) (the error type in octal representation)

PROGRAM TERMINATES. FIT FROM EROUT__ (the file information table from the direct access file in octal).

APPENDIX E



PROGRAM ORGANIZATION SHOWING RELATIONSHIP BETWEEN MAIN PROGRAM AND SUBROUTINES SEE APPENDIX B FOR INDIVIDUAL FLOW CHARTS WITH DETAILS

Program Subroutines

The following subroutines are included in program PSIM.

Name: DDAYS Arguments: none

DDAYS is a temperature routine which uses both historical and current year data to calculate mean temperatures and growing ddays for every day from March 1 through September 30. Ddays are calculated from a user-supplied base temperature. The subroutine stores accumulated ddays every day for the same period both historically and in the current year.

Name: DATE Arguments: none

DATE determines planting date of a new cultivar when a planting is necessitated by partial planting of a preceding cultivar.

Name: EXCEED Arguments: (NARKA, NARKB)

EXCEED checks to see if the harvest date for a given planting or subplanting exceeds the user-supplied harvest deadline. If so, the number of days by which the deadline is exceeded is calculated.

Name: MATUR Arguments: (LL, MM, NN)

MATUR determines the day of harvest for a given subplanting based on the number of ddays to 100 TR. It calls subroutine EXCEED to determine whether the harvest date exceeds the harvest deadline. It calls subroutine FACTOR to determine the average number of heat units/day for five days preceding and five days following harvest. It also determines the dates on which each subplant reaches 80 TR and 140 TR.

Name: FACTOR Arguments: (KHAD, KHAM, L1, M1, N1)

FACTOR calculates the average heat units/day for the five days preceding and the five days following harvest for a given subplant.

Name: RNDEV Arguments: (X2)

FUNCTION RNDEV is a psuedo-random number generator. The random numbers are returned to the main program for use in program calculations.

Name: RANGE Arguments: (IFNUM)

RANGE determines the first and last days of harvest after all harvests have been rescheduled to accommodate cannery capacity.

Name: WASTE Arguments: (MAX, MIN)

WASTE prints all dates on which final harvests either exceed maximum or fall below minimum cannery capacity. It lists excess and slack in pounds.

Name: EROUT1 - EROUT10 Arguments: none

Ten error routines are included in program PSIM. Each of these routines corresponds to either an input or access attempt on a direct access file. The routines are included as diagnostic aids should any problems arise in connection with this particular aspect of program structure. Diagnostic information is printed in octal and will require programming expertise to interpret. See Appendix D for a description of the calls to the direct access file.

APPENDIX F

Dictionary of Program Variables

Variable Name	Dimensions of Array	Where used: (main program and/or SUBROUTINE)[1]	Description
A		RNDEV	used to generate a random number
AKEY		Main, RANGE	a five to six digit ID number for each subplanting; the right-most two digits denote subplant number, the middle two digits denote planting number and the left-most digit or digits is the variety number; same as the first element of array AWS
AVGHI	10,30,20	Main, MATUR, FACTOR	average of the ddays/day for five days following 100 TR for each subplant
AVGLO	10,30,20	Main, MATUR, FACTOR	average of the ddays/day for five days preceding 100 TR for each subplant
AWS	12	Main, RANGE, MATUR, FACTOR, EROUT1-EROUT10	working storage array used to transmit descriptive information one record at a time to direct access file[2]
B		Main, EXCEED, MATUR	logical variable used to determine if date being checked in subroutine EXCEED is historical or current year harvest date
BASE		Main	base temperature used in simulation for determining planting and harvest dates
CONV1 CONV2	10 10	Main, MATUR	conversion factors supplied by user to convert heat units/day at harvest to TR points/day; CONV1 is used prior to 100 TR, CONV2 after 100 TR
COUNT		Main	counter to sum acres in subplants of one planting
DD		DDAYS	degree days
DDREF		DDAYS	same as BASE
DDTOT		DDAYS	counter used to sum ddays through season

[1] Variables may be included in unnamed COMMON in places other than those listed here.

[2] The 12 elements of AWS are as follows: 1) same as AKEY; 2) ddays to 100 TR for this subplant; 3) harvest day this subplant; 4) harvest month this subplant; 5) average ddays/day five days preceding 100 TR this subplant; 6) average ddays/day five days following 100 TR this subplant; 7) day of 80 TR this subplant; 8) month of 80 TR this subplant; 9) day of 140 TR this subplant; 10) month of 140 TR this subplant; 11) subplant size (acres); 12) subplant yield (lbs).

Variable Name	Dimensions of Array	Where used: (main program and/or SUBROUTINE)	Description
ENDPL		Main	defined as 999.99 if harvest deadline exceeded
H		Main	random number seed
HHUS		Main	ddays accumulated on historical harvest date
IA		DDAYS	converts array month to calendar month
IACPL		Main	counter for total acres planted in simulation
IACRQ	10	Main	acre requirements for each cultivar in simulation
IACTOT		Main	counter for total number of acres planted on a given day
IARQS		Main	sum of all elements in IACRQ
IBNCH		Main, WASTE	pounds of bunching on given harvest day
ICAP		Main	daily cannery capacity (lbs/day)
IDAY	5	DDAYS	reads day from temperature data
IDDUM		Main	first day in current year when mean temperature is greater than or equal to base temperature; day of first planting
IDESY	10	Main	desired total yield for each variety
IDIFF		MATUR	defined only if a subplant reaches 80 TR in a different month that that in which it reaches 100 TR; the number of days from 80 TR to the end of the month in which 80 TR occurs
IFAC1 IFAC2	10 10	Main, MATUR	the number of TR points added/day between 80 TR and 100 TR and between 100 TR and 140 TR
IFHD IFHM		Main, RANGE	day of first harvest; month of first harvest
IFNUM		Main, RANGE	field number
IHAD IHAM		Main	historical harvest day; historical harvest month
IHARD IHARM	10, 30 10, 30	Main, EXCEED	historical harvest day and historical harvest month in an array whose subscripts refer to cultivar number and planting number

Variable Name	Dimensions of Array	Where used: (main program and/or SUBROUTINE)	Description
IHDD IHDM		Main	day of harvest deadline; month of harvest deadline
IHUMN	10	Main	maturity constants for each cultivar
IHUSD	10	Main	standard deviations of maturity constants for each cultivar
IMDUM		Main	first month in current year when mean temperature is greater than or equal to the base temperature; month of first planting; paired with IDDUM
IPDD IPMD	10,30 10,30	Main, MATUR	planting day and planting month for all plantings of all cultivars
IPSZ	10,30	Main	size of plantings (acres) for all plantings of all cultivars
IRECAP		Main	revised daily cannery capacity; used when a partial planting of one cultivar requires a partial planting of one or more additional cultivars
IREXY		Main	revised expected yield; used for partial plantings
ISIM		Main, RANGE, MATUR	number of cultivars in simulation
ISLAK		Main, WASTE	slack (lbs) at cannery on given harvest day
ISLAKT		Main, WASTE	total lbs of slack at cannery during season
ISSIZE	10	Main	SPS for each cultivar
ITMN ITMX	5 5	DDAYS	read daily minimum and maximum temperatures from temperature data
ITOTAL		Main	total number of pounds harvested during season
ITRHAR	10,30,20	Main	TR readings of each subplant at harvest
IVAR	10	Main	names of all cultivars in simulation
IWASTE		Main, WASTE	total pounds wasted during season; cannery excess
IWS	12	Main	same as AWS but integer values instead of real

Variable Name	Dimensions of Array	Where used: (main program and/or SUBROUTINE)	Description
IX		Main, MATUR	number of days by which the harvest deadline is exceeded in a given case
IXYMN	10	Main	expected yield (mean lbs/A) for each cultivar
IXYSD	10	Main	standard deviation of expected yield for each cultivar
IYEAR		DDAYS	non-array representation of year of temperature data
IYR	5	DDAYS	reads in year of temperature data
I100TR I140TR I80TR		Main	total number of pounds harvested at 100 TR, 140 TR and 80 TR during season
JACRQ	10	Main	same as IACRQ except value changes during simulation as planting progresses
JDAY		Main	current year harvest day of one subplant at a time; used to sum daily yields
JHAD JHAM		MATUR	marker variables representing the actual harvest day and month (respectively) for a given subplant
JHDD JHDM		EXCEED	same as IHDD and IHDM; values passed via COMMON
JMO		Main	harvest month of one subplant at a time, current year; paired with JDAY
JSSIZE	10	Main	same as ISSIZE except value changes during simulation if partial plantings are made
JX		EXCEED	same as IX; value passed via COMMON
JYSUM	7, 31	Main, WASTE	sum of actual yields on each harvest day
LACK		Main	yield (lbs) required of partial planting to fill cannery capacity
LAST		Main	takes value of elements in LIMIT when they are needed as array elements
LHD LHM		Main, RANGE	last harvest day and last harvest month
LIM		Main, RANGE	same use as LAST

Variable Name	Dimensions of Array	Where used: (main program and/or SUBROUTINE)	Description
LIMIT	10	Main, RANGE	number of plantings made for each cultivar
LL		Main, MATUR	argument passed to MATUR; denotes cultivar being planted
LOOPA LOOPB		WASTE	counters used to avoid repeat of headlines
LPD LPM		Main	last planting scheduled, day and month
L1		FACTOR	argument passed to FACTOR; denotes cultivar being planted
MARKA MARKB		Main	identify cultivar and planting being scheduled
MARKC MARKD		MATUR	dummy variables used to represent planting day and month (respectively) for planting being processed by MATUR
MARKE MARKF		Main	non-array representation for day and month (respectively) on which a given subplant reaches 80 TR
MAX		Main, WASTE	upper limit of cannery capacity (lbs/day)
MIN		Main, WASTE	lower limit of cannery capacity (lbs/day)
MO		Main	passed to subroutine DATE as cultivar for which a partial planting is to be made
MON		Main	same use as MO1
MONTH		DDAYS	non-array representation of month of temperature data
MO1		Main, WASTE	actual harvest month; calendar date instead of array date
MM		MATUR	argument passed to MATUR; denotes the planting being processed
MULT		Main	planting multiple for first planting
M1		FACTOR	argument passed to FACTOR; denotes planting being processed
N		DDAYS, DATE, EROUT1-EROUT10	in DDAYS: the tape number temperature data is read from; in DATE: the cultivar being planted; in EROUT1 through EROUT10: used to determine error number in direct access file (octal)

Variable Name	Dimensions of Array	Where used: (main program and/or SUBROUTINE)	Description
NARKA NARKB		EXCEED	denote cultivar and planting being processed
NDAY		DDAYS	non-array representation of day of temperature data
NHAD NHAM		DATE	historical harvest day and month of cultivar being processed
NHUMN	10	DATE	same as IHUMN
NN		MATUR	argument passed to MATUR; denotes the sub-planting being processed
NPD		Main	next planting day
NPDD	10,30	DATE	same as IPDD
NPM		Main	next planting month
NPMD	10,30	DATE	same as IPMD
NYR		DDAYS	calendar year of temperature data
N1		FACTOR	argument passed to FACTOR; denotes sub-planting being processed
RAND		Main	assigned the value of random numbers generated in RNDEV
SDDC SDDH	7,31 7,31	DATE	average ddays/day March 1 through September 30, current year and historical data respectively
SDDTOC SDDTOH	7,31 7,31	DATE	total number of accumulated ddays on each day, March 1 through September 30, current year and historical data respectively
SMC SMH	7,31 7,31	DATE	mean daily temperature March 1 through September 30, current year and historical data respectively
SREF		DATE	same as BASE
TDD	7,31	DDAYS	ddays/day March 1 through September 30, historical data
TDDC TDDH	7,31 7,31	Main, FACTOR	same as SDDC and SDDH

Variable Name	Dimensions of Array	Where used: (main program and/or SUBROUTINE	Description
TDDTOC TDDTOH	7,31 7,31	Main	same as SDDTOC and SDDTOH
TDDTOT TDDTO1	7,31 7,31	DDAYS	same as SDDTOC and SDDTOH
TDD1	7,31	DDAYS	ddays/day March 1 through September 30, current year
TLHM		RANGE	marker for last harvest month; "temporary" last harvest month
TM	7,31	DDAYS	same as SMH
TMC TMH	7,31 7,31	Main	same as SMC and SMH
TMEAN		DDAYS	mean temperature
TM1	7,31	DDAYS	same as SMC
TWS	12	Main	same as AWS except that this working storage area is temporary
U1 U2		RNDEV	used in generation of random numbers
X2		RNDEV	used in generation of random numbers
Y		Main	random number seed
YFACT1 YFACT2	10 10	Main	per cent yield of each cultivar at 80 TR and 140 TR respectively
YRSUM		DDAYS	counter for number of years in temperature data

Notes on the Program Listing

The program listing included here is the time-sharing version of PSIM. The batch version differs in two ways. First, the program card is different (see Guide to Technical Program Structure in Appendix D for a comparison of the two cards). Second, READ statements in the batch version are formatted. Formats which appear along with READ statements in this listing, though not used, were left to simplify conversion to batch mode. Thus, the first such example in this listing is:

```
      READ*,ISIM
   20 FORMAT (I2)
```

To convert the program to batch mode, a user must alter the READ statement as follows:

```
      READ (5,20)
```

where 5 refers to TAPE5 which is equivalenced to system INPUT in the program card, and where 20 is the number of the format statement used.

The next instance is:

```
      READ*,IVAR(N),IHUMN(N),IHUSD(N),IACRQ(N),IDESY(N),
      IXYMN(N),IXYSD(N)
   40 FORMAT(A10,1X,5(I7,1X),I7)
```

The necessary alteration would be:

```
      READ (5,40) IVAR(N),IHUMN(N),IHUSD(N),IACRQ(N),
      IDESY(N),IXYMN(N),IXYSD(N)
```

There are 11 READ statements in the main program and three in subroutine MATUR. Each of these must be altered as in these examples before the program is executed in batch mode.

```
      PROGRAM PSIM(INPUT=/72,OUTPUT,TAPE6=OUTPUT,TAPE1,TAPE2)
C
C     PROGRAM PSIM IS A CROP SCHEDULING SIMULATION BASED ON THE HEAT
C     UNIT SYSTEM FOR PREDICTION OF CROP MATURATION. SEE PROGRAM DOCU-
C     MENTATION FOR DETAILS.
C
C
C     ALPHABETICAL DICTIONARY FOR PROGRAM PSIM
C     AKEY  EQUAL TO 1ST ELEMENT OF ARRAY AWS;ID OF VAR,PLNTNG,SUBPLNTNG
C     AVGHI  3 DIMEN ARRAY STORING AVG NO. HUS/DAY FOR 5 DAYS FOLLOWNG
C       100 TR FOR EACH SUBPLNT
C     AVGLO  3 DIMEN ARRAY STORING AVG NO. HUS/DAY FOR 5 DAYS PRECEDNG
C       100 TR FOR EACH SUBPLNT
C     AWS  1 DIMENSIONAL ARRAY OF WORKING STORAGE FOR RECORDS TRANS-
C       MITTED TO DIRECT ACCESS FILE TAPE4; 12 ELEMENTS EACH DESCRIBING
C       1 SUBPLANT. 1)ID NO.;2)HU'S TO 100TR;3)HARVEST DAY;4)HARVEST MO
C       5)AVG HUS 5 DAYS PRECEDING 100TR;6)AVG HUS 5 DAYS FOLLOWING 100
C       TR;7)DAY OF 80TR;8)MO.OF 80TR;9)DAY OF 140TR;10)MO. OF 140TR;
C       11)SBPLNT SIZE(ACRES); 12)SUBPLNT YIELD(LBS)
C     B  LOGICAL VARIABLE USED IN SUBROUTINE EXCEED TO DETERMINE IF DATE
C        BEING CHECKED IS HISTORICAL OR ACTUAL HARVEST DATE
C     BASE  BASE TEMP USED IN SIM TO CALC HU'S
C     COUNT  COUNTER TO SUM ACRES IN SUBPLNTS OF 1 PLNTNG
C     ENDPL  DEFINED AS 999.99 IF HARVEST DEADLINE EXCEEDED
C     H  RANDOM NUMBER SEED
C     HHUS  HISTOR HU'S ACCUM'D ON PROJECTD HARVEST DAY
C     IACPL  COUNTER FOR TOTAL ACRES PLNTD IN SIMULATN
C     IACRQ  1 DIMENS ARRAY OF ACRE REQRMNTS FOR EACH VAR IN SIMULATN
C     IACTOT  COUNTER FOR TOTAL NO. OF ACRES PLNTD ON GIVEN DAY
C     IARQS  SUM OF ALL ELEMENTS IN ARRAY IACRQ
C     IBNCH  LBS OF BUNCHING ON GIVEN HARVEST DAY
C     ICAP  DAILY PLNT CAPACITY (LBS/DAY)
C     ICHEK  SUMS ACRES PLNTD IN SIMULATN
C     IDDUM  FIRST DAY IN CURRENT YR WHEN MEAN TEMP .GE. BASE TEMP OR
C       DAY OF FIRST PLANT
C     IDESY  1 DIMENS ARRAY OF DESIRED TOTAL YIELD FOR EACH VAR
C     IDIV  1 DIMENS ARRAY OF DIVISORS USED TO CALCULATE STDRD PLANTING
C       SIZE; NUMERATOR IS ICAP
C     IFHD  DAY OF FIRST HARVEST
C     IFHM  MONTH OF FIRST HARVEST
C     IFNUM  FIELD NUMBER
C     IHAD  PROJECTED HARVEST DAY (HISTOR DATA)
C     IHAM  PROJECTED HARVEST MONTH (HISTOR DATA)
C     IHARD  2 DIMENS ARRAY OF PROJECTED HARVEST DAYS OF EACH PLNTNG
C     IHARM  2 DIMENS ARRAY OF PROJECTED HARVEST MONTHS OF EACH PLNTNG
C     IHDD   DAY OF HARVEST DEADLINE
C     IHDM   MONTH OF HARVEST DEADLINE
C     IHUMN  1 DIMENS ARRAY OF MATURITY CONSTANTS FOR EACH VAR
C     IHUSD  1 DIMENS ARRAY OF MATURITY CONSTANT STNDRD DEVIATNS FOR
C       EACH VARIETY
C     IMDUM  MONTH IN CURRENT YR WHEN MEAN TEMP .GE. BASE TEMP,OR
C     MONTH OF FIRST PLANT; PAIRED WITH IDDUM
C     IPDD  2 DIMMENS ARRAY OF PLNTNG DAYS FOR ALL PLNTNGS
C     IPMD  2 DIMENS ARRAY OF PLNTNG MONTHS FOR ALL PLNTNGS
C     IPSZ  2 DIMENS ARRAY OF PLNTNG SIZES
C     IRECAP  REVISED DAILY CANNERY CAPACITY
C     IREXY    REVISED EXPECTED YIELD
C     ISIM  NO. OF VARS. IN SIMULATN
C     ISLAK  SLACK (LBS) AT CANNERY ON GIVEN HARVEST DAY
C     ISLAKT  TOTAL LBS SLACK FOR SEASON
C     ISSIZE  1 DIMENS ARRAY OF STANDARD PLNTNG SIZE FOR EACH VAR
```

```
C       ITOTAL  TOTAL NO. OF LBS. HARVESTED DURING SEASON
C       ITRHAR  3 DIMENS ARRAY OF TR TEADINGS AT HARVEST OF EACH SUBPLNT
C       IWASTE  TOTAL LBS WASTED DURING SEASON; CANNERY EXCESS
C       IWS  1 DIMENS ARRAY, 7 ELEMENTS; USED TO CONVERT REAL VALUES FROM
C         AWS TO INTEGERS
C       IVAR  1 DIMENS ARRAY OF NAMES OF ALL VARS
C       IX  NO. OF DAYS BY WHICH HARVEST DEADLINE EXCEEDED
C       IXYMN  1 DIMENS ARRAY OF MEAN EXPECTED YIELDS (LBS/ACRE) FOR EACH
C         VARIETY
C       IXYSD  1 DIMENS ARRAY OF STANDRD DEVIATNS FOR EXPECTED YIELD OF
C         EACH VARIETY
C       I80TR  TOTAL NO. LBS. HARVESTED AT 80 TR
C       I100TR  TOTAL NO. LBS. HARVESTED AT 100 TR
C       I140TR  TOTAL NO. LBS. HARVESTED AT 140 TR
C       JACRQ  SAME AS IACRQ EXCEPT VALUE CHANGES DURING SIMULATN AS
C         PLANTING PROGRESSES
C       JDAY  HARVEST DAY OF GIVEN SUBPLNT; USED TO SUM DAILY YIELDS
C       JMO  HARVEST MONTH OF GIVEN SUBPLANT; PAIRED WITH JDAY
C       JSSIZE  SAME AS ISSIZE EXCEPT VALUE CHANGES IF PARTIAL PLNTNG MADE
C       JYSUM  2 DIMENS ARRAY OF ACTUAL YIELDS ON EACH HARVEST DAY
C       LACK  YIELD (LBS) REQUIRD OF PARTIAL PLNTNG TO FILL CANNERY CAPAC
C       LAST  TAKES VALUE OF ELEMENTS IN ARRAY LIMIT WHEN NEEDED AS ARRAY
C         ELEMENT
C       LHD  LAST HARVEST DAY
C       LHM  LAST HARVEST MONTH
C       LIM  SAME USE AS VARIABLE LAST
C       LIMIT  1 DIMENS ARRAY OF NO. OF PLNTNGS MADE FOR EACH VARIETY
C       LPD  LAST PLANTNG DAY
C       LPM  LAST PLANTING MONTH
C       MARKA  IDENTIFIES VARIETY BEING PLNTED
C       MARKB  IDENTIFIES PLNTNG NUMBER OF VAR BEING PLNTD
C       MARKE  NON-ARRAY REPRESENTATN OF DAY ON WHICH 1 SUBPLNT REACHES
C         80 TR
C       MARKF  NON-ARRAY REPRESENTATN OF MONTH ON WHICH 1 SUBPLNT REACHES
C         80TR
C       MAX  UPPER LIMIT OF CANNERY CAPACITY (LBS/DAY)
C       MIN  LOWER LIMIT OF CANNERY CAPACITY (LBS/DAY)
C       MO  PASSED TO SUBROUTINE DATE AS VAR FOR WHICH PARTIAL PLNTNG TO
C         BE MADE
C       MO1  ACTUAL HARVEST MONTH; CALENDAR DATE INSTEAD OF ARRAY DATE
C       MON  SAME USE AS MO1
C       MULT  FIRST PLANTING MULTIPLE
C       MULT2  ASSIGNED A VALUE OF MULT-1; USED AS LIMIT IN LOOP WHICH RESETS
C         PLANTING MULTIPLE
C       NPD  NEXT PLNTNG DAY
C       NPM  NEXT PLNTNG MONTH
C       RAND  RANDOM NUMBER; TAKES ON VALUE RETURNED FROM FUNCTION
C         RNDEV
C       TDDC  2 DIMENS ARRAY OF AVG HU'S/DAY MAR THRU SEPT CURRENT YR
C       TDDH  2 DIMENS ARRAY OF AVG HU'S/DAY MAR THRU SEPT HISTOR DATA
C       TDDTOC  2 DIMENS ARRAY OF ACCUM'D HU'S MAR THRU SEPT CURRENT YR
C       TDDTOH  2 DIMENS ARRAY OF ACCUM'D HU'S MAR THRU SEPT HISTOR DATA
C       TMC  2 DIMENS ARRAY OF MEAN DAILY TEMPS MAR THRU SEPT CURRENT YR
C       TMH  2 DIMENS ARRY OF MEAN DAILY TEMPS MAR THRU SEPT HISTOR DATA
C       TWS  SAME AS AWS BUT TEMPORARY STORAGE USED IF HARVESTS RESCHEDLD
C         TO MEET FACTORY CAPACITY
C       Y  RANDOM NUMBER SEED
C       YFACT1  1 DIMENS ARRAY OF PERCENT YIELD AT 80 TR FOR EACH VAR
C       YFACT2  1 DIMENS ARRAY OF PERCENT YIELD AT 140 TR FOR EACH VAR
C
```

```
C
      LOGICAL B
      DIMENSION IVAR(10), IHUSD(10), IACRQ(10), IDESY(10), IXYMN(10)
      DIMENSION IXYSD(10),ISSIZE(10),JACRQ(10),JSSIZE(10),IPSZ(10,30
+     ),ITRHAR(10,30,21),JYSUM(7,31),LIMIT(10),YFACT1(10).YFACT2(10),
+     IDIV(10)
C
      COMMON BASE, TMH(7,31), TDDH(7,31), TDDTOH(7,31), TMC(7,31), TDDC(
+     7,31),TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,
+     30),IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7),ISIM,
+     AVGLO(10,30,21),AVGHI(10,30,21),CONV1(10),CONV2(10),IFAC1(10),
+     IFAC2(10),JYSUM,LIMIT,IWASTE,IFHM,IFHD,LHD,LHM,B,ISLAKT
C
      EXTERNAL EROUT1, EROUT2, EROUT3, EROUT4, EROUT5, EROUT6, EROUT7,
+     EROUT8, EROUT9, EROUT10
C                                     READ DATA FOR EACH VAR + SIMULTN
      CALL FILEDA(TAPE4,3LLFN,5LTAPE4)
      CALL OPENM(TAPE4,3LNEW)
C
      DO2N=1,7
      DO2K=1,31
        TMH(N,K)=0.0
        TDDH(N,K)=0.0
        TDDTOH(N,K)=0.0
        TMC(N,K)=0.0
        TDDC(N,K)=0.0
        TDDTOC(N,K)=0.0
    2   CONTINUE
      BASE=0.0
C
      WRITE(6,10)
   10 FORMAT(//,*ENTER NO. OF VARS. IN SIMULATION*)
C
      READ*,ISIM
   20 FORMAT(I2)
C
      WRITE(6,30)
   30 FORMAT(//,*ENTER FOR EACH VARIETY*)
C
      DO60N=1,ISIM
         READ*,IVAR(N),IHUMN(N),IHUSD(N),IACRQ(N),IDESY(N),IXYMN(N),
+        IXYSD(N)
   40 FORMAT(A10,1X,5(I7,1X),I7)
   60   CONTINUE
C
      DO130J=1,ISIM
        IF(IACRQ(J).EQ.0) IACRQ(J)=IDESY(J)/IXYMN(J)
        JACRQ(J)=IACRQ(J)
  130   CONTINUE
C
      IARQS=0
      DO150K=1,ISIM
        IARQS=IARQS + IACRQ(K)
  150   CONTINUE
C
      WRITE(6,160)
  160 FORMAT(//,*ENTER DAILY PLANT CAP IN LBS*)
C
      READ*,ICAP
  180 FORMAT(I7)
```

```
C
      DO 190 J=1,ISIM
        ISSIZE(J)=0
  190   CONTINUE
      WRITE(6,195)
  195 FORMAT(//,*ENTER ACRAGE DIVISOR FOR EACH VARIETY*)
      DO 205 K=1,ISIM
      READ*,IDIV(K)
  200   FORMAT(I5)
  205   CONTINUE
C
      DO 210 L=1,ISIM
        IF(IDIV(L).EQ.0) IDIV(L)=IXYMN(L)
        ISSIZE(L)=ICAP/IDIV(L)
        JSSIZE(L)=ISSIZE(L)
  210   CONTINUE
C
      WRITE(6,220)
  220 FORMAT(//,*ENTER HARVEST DEADLINE, MONTH AND DAY*)
C
      READ*,IHDM,IHDD
  240 FORMAT(I2,3X,I2)
C
      IACPL=0
C                                   ACCESS CURRENT YR DATA IN DDAYS
      CALL DDAYS
      WRITE(6,250) BASE
  250 FORMAT(1X,F4.0)
C                                   CALC 1ST PLNT DATE IF NOT USER SUPPLD
      WRITE(6,255)
  255 FORMAT(//,*ENTER FIRST PLANT MONTH AND DAY*)
C
      READ*,IMDUM,IDDUM
  256 FORMAT(I2,3X,I2)
C
      IF(IMDUM.NE.0) GOTO 275
C
      DO270J=1,7
      DO270K=1,31
       IF(IDDUM.NE.0) GOTO 275
        IF(TMC(J,K).GE.BASE) IDDUM=K
        IF(IDDUM.EQ.K) IMDUM=J+2
  270   CONTINUE
C
  275 WRITE(6,280) IMDUM, IDDUM
  280 FORMAT(1X,I1,3X,I2)
C
  300 IPDD(1,1)=IDDUM
      IPMD(1,1)=IMDUM
      LPD=IDDUM
      LPM=IMDUM
C                                   DETRMNE 1ST PLNT SIZE
      WRITE(6,310)
  310 FORMAT(//,*ENTER PLANTING MULTIPLE*)
      READ*,MULT
  330 FORMAT(I1)
C
      IF((MULT*ISSIZE(1)).GT.IACRQ(1)) GOTO 1330
      IF((MULT*ISSIZE(1)).LE.IACRQ(1)) GOTO410
```

```
C
  360 MULT2 = MULT - 1
      DO 390 N=1,MULT2
        IF(IPSZ(1,1).NE.0) GOTO 440
        IF(((MULT-N)*ISSIZE(1)).LE.IACRQ(1))
+       IPSZ(1,1)=(MULT-N)*ISSIZE(1)
        IF(IPSZ(1,1).EQ.((MULT-N)*ISSIZE(1))) MULT = MULT-N
        IF((N.EQ.MULT).AND.(IPSZ(1,1).EQ.0)) GOTO 1390
  390   CONTINUE
C
      IF(IPSZ(1,1).NE.0) GOTO 415
  410 IPSZ(1,1)=MULT*ISSIZE(1)
C
  415 WRITE(6,420) IPSZ(1,1)
  420 FORMAT(1X,I4)
C
  440 JACRQ(1)=JACRQ(1)-IPSZ(1,1)
      IACPL=IACPL + IPSZ(1,1)
C                                 ACCESS HIST DATA IN DDAYS
      CALL DDAYS
C                                 DETRMNE HIS HRVST DATE 1ST PLNTNG
      DO470J=1,7
      DO470K=1,31
        IF(IHARD(1,1).NE.0) GOTO 475
        IF((J.LT.IMDUM).OR.((J.EQ.IMDUM).AND.(K.LE.IDDUM))) GOTO 470
        IF(((J.EQ.2).OR.(J.EQ.4).OR.(J.EQ.7)).AND.(K.EQ.31)) GOTO 470
        IF((TDDTOH(J,K)-TDDTOH(IMDUM-2,IDDUM)).GE.FLOAT(IHUMN(1)))
+       IHARD(1,1) =K
        IF(IHARD(1,1).EQ.K) IHARM(1,1)=J+2
        IF(IHARD(1,1).EQ.K) HHUS=TDDH(J,K)
C
        IF(IHARD(1,1).EQ.K) LHD=K
        IF(IHARD(1,1).EQ.K) LHM=J+2
  470   CONTINUE
C
  475 WRITE(6,480) IHARM(1,1), IHARD(1,1), HHUS
  480 FORMAT(1H ,*HARVEST DATE 1ST PLANT VAR 1 IS*,2X,I1,3X,I2,2X,*HUS*,
+     1X,*ON THIS DATE ARE*,F10.2)
C
  500 ENDPL=0.0
C                                 BEGIN LOOP TO SCHEDLE PLNTNGS,HARVSTS
  530 DO1450L=1,ISIM
  540   IF((ENDPL.EQ.999.99).OR.(IACPL.EQ.IARQS)) GOTO1300
C
        MARKA=0
        MARKB=1
C                                 DETRMNE VAR,PLNTNG TO SCHEDLE NEXT
  590   IF((JACRQ(L).GT.0).AND.(JSSIZE(L).GT.0)) MARKA=L
        IF(MARKA.NE.L) GOTO 1450
        IF((MARKA.EQ.1).AND.((IACRQ(1)-JACRQ(1)).NE.0)) MARKB =(((IACRQ
+       (1)-JACRQ(1))-IPSZ(1,1))/ISSIZE(1)) + 2
C
        IF((MARKA.EQ.L).AND.(L.NE.1).AND.((IACRQ(L)-JACRQ(L)).NE.0))
+       MARKB= ((IACRQ(L)-JACRQ(L))/ISSIZE(L))+1
C
        IF(IPSZ(MARKA,MARKB).NE.0) MARKB=MARKB+1
C
        IF(MARKB.GT.30) GOTO 1440
```

```
C
      WRITE(6,600) IVAR(MARKA), MARKB
  600 FORMAT(1H ,*VAR BEING PLNTD IS*,2X,A10,2X,*PLNTNG NO.*,2X,I2,2X,
     +*WILL BE SCHEDULED NEXT*)
C
  620 NPD =0
C                                DETRMNE DATE OF NEXT PLNTNG
      DO650LL=1,7
      DO650M=1,31
        IF(NPD.NE.0) GOTO 680
        IF((((M.GT.LPD).AND.(LL.EQ.(LPM-2))).OR.(LL.GT.(LPM-2))).AND.
     +  ((TDDTOC(LL,M)-TDDTOC(LPM-2,LPD)).GE.HHUS)) NPD=M
        IF(NPD.EQ.M) NPM = LL
  650   CONTINUE
C
  680 IPDD(MARKA,MARKB)=NPD
      IPMD(MARKA,MARKB) = NPM + 2
C
      LPD=NPD
      LPM=NPM+2
C
      WRITE(6,690) MARKB, IVAR(MARKA), LPM, LPD
  690 FORMAT(1H ,*PLANTING NO.*,2X,I2,2X,*OF VAR*,2X,A10,2X,*OCCURS*,2X
     +,I1,2X,I2)
C
  710 IHAD=0
C                                DETRMNE HIST HARVST DATE THIS PLNTNG
  740 DO770J=1,7
      DO770K=1,31
        IF(IHAD.NE.0) GOTO 800
        IF(((J.EQ.2).OR.(J.EQ.4).OR.(J.EQ.7)).AND.(K.EQ.31))GOTO 770
C
      IF((((K.GT.NPD).AND.(J.EQ.NPM)).OR.(J.GT.NPM)).AND.((TDDTOH(J,
     +  K)-TDDTOH(NPM,NPD)).GE.FLOAT(IHUMN(MARKA)))) IHAD = K
        IF(IHAD.EQ.K) IHAM=J
        IF(IHAD.EQ.K) HHUS = TDDH(J,K)
  770   CONTINUE
C
  800 IHARD(MARKA,MARKB) = IHAD
      IHARM(MARKA,MARKB) = IHAM+2
C
      WRITE(6,810) MARKB, MARKA, HHUS
  810 FORMAT(1X,*HIST HU'S ON HARVEST OF PLNTNG*,2X,I2,2X,*OF VAR*,2X,
     +I2,2X,*ARE*,F10.2)
C
      IX = 0
      B = .FALSE.
C                                DOES HARVEST DATE EXCEED DEADLINE?
      CALL EXCEED(MARKA,MARKB)
C
      IF(IX.NE.0) ENDPL = 999.99
      IF(IX.EQ.0) GOTO 910
C
  850   WRITE(6,880)MARKB, IVAR(MARKA), IX
  880 FORMAT(1H ,*HIST HARVEST DATE OF PLNTNG*,2X,I2,2X,*OF VAR*,3X,
     +  A10,3X,*EXCEEDS HARVEST DEADLINE BY*,2X,I4,2X,*DAYS*)
C
      IF(ENDPL.EQ.999.99) GOTO 1460
C
  910 LHD=IHARD(MARKA,MARKB)
      LHM=IHARM(MARKA,MARKB)
```

```
C
      WRITE(6,912) MARKB, IVAR(MARKA), LHM, LHD
  912 FORMAT(1H ,*HARVEST DATE FOR PLNTNG*,2X,I2,2X,*OF VAR*,2X,A10,2X,
+      *IS*,2X,I1,3X,I2)
      IF(IACPL.EQ.IARQS) GOTO 1450
C                                  CANNERY CAPACITY FILLED BY THIS VAR?
      IF(JACRQ(MARKA).GE.JSSIZE(MARKA)) GOTO 1260
C                                  PARTIAL PLNTNGS SCHEDLD THRU STMT 1260
      IREXY = 0
C
      IPSZ(MARKA,MARKB) = JACRQ(MARKA)
      JACRQ(MARKA) = JACRQ(MARKA)-IPSZ(MARKA,MARKB)
      IACPL = IACPL + IPSZ(MARKA,MARKB)
      IREXY = IXYMN(MARKA)*IPSZ(MARKA,MARKB)
      JSSIZE(MARKA) = 0
C
       WRITE(6,915)MARKB,IVAR(MARKA),IPSZ(MARKA,MARKB),JACRQ(MARKA)
  915  FORMAT(1X,*PLNTNG*,2X,I2,2X,*OF VAR*,2X,A10,2X,*IS*,2X,I4,2X,
+      *ACRES*,2X,I4,2X,*ACRES OF THIS VAR REMAIN TO BE PLANTED*)
C
       IF(IACRQ(MARKA + 1).EQ.0) GOTO 1460
      IF(IACPL.EQ.IARQS) GOTO 1300
C
      WRITE(6,920) IVAR(MARKA+1), IHUMN(MARKA+1)
  920 FORMAT(1X,*MATURITY CONST FOR*,3X,A10,2X,*IS*,2X,I4)
      MO=MARKA + 1
C
      CALL DATE(MO,IHAM,IHAD)
C
      IF(IPDD(MARKA+1,1).EQ.0) GOTO 1420
C
C
      WRITE(6,1210) IVAR(MARKA+1), IPMD(MARKA+1,1), IPDD(MARKA+1,1)
 1210 FORMAT(1H ,*DATE OF 1ST PLNT VAR*,2X,A10,2X,*IS*,2X,I1,2X,I2)
C
 1230 IF((IPMD(MARKA+1,1).EQ.LPM).AND.(IPDD(MARKA+1,1).GT.LPD)) LPD =
+     IPDD(MARKA+1,1)
C
      IF(IPMD(MARKA+1,1).GT.LPM) LPM = IPMD(MARKA+1,1)
      IF(IPMD(MARKA+1,1).GT.LPM) LPD=IPDD(MARKA+1,1)
C
      IRECAP = ICAP-IREXY
C
      WRITE(6,1235) IRECAP, IVAR(MARKA+1), JACRQ(MARKA+1)
 1235 FORMAT(1H ,*NEW PLANT CAP IS*,2X,I7,2X,*ACRE REQMT FOR VAR*,2X,A10
+     ,2X,*IS*,2X,I4)
C
      IF(IRECAP/IXYMN(MARKA+1).GE.JACRQ(MARKA+1)) GOTO 1240
C
      IF(IRECAP/IXYMN(MARKA+1).LT.JACRQ(MARKA+1)) IPSZ(MARKA+1,1) =
+     IRECAP/IXYMN(MARKA+1)
      JACRQ(MARKA+1) = JACRQ(MARKA+1)-IPSZ(MARKA+1,1)
C
      GOTO 1250
C
 1240 IPSZ(MARKA+1,1) = JACRQ(MARKA+1)
      IACPL = IACPL + IPSZ(MARKA+1,1)
      LACK=IRECAP-(IPSZ(MARKA+1,1)*IXYMN(MARKA+1))
      JACRQ(MARKA+1) = 0
```

```
C
        IHARD(MARKA+1,1) = IHARD(MARKA,MARKB)
        IHARM(MARKA+1,1) = IHARM(MARKA,MARKB)
C
        IF(IACPL.EQ.IARQS) GOTO 1300
C
        MO = MARKA+2
C
        DO1245IM=MO,ISIM
          IF(LACK.EQ.0) GOTO 1246
          IF((LACK/IXYMN(IM)).LE.JACRQ(IM)) IPSZ(IM,1)=LACK/IXYMN(IM)
          IF(IPSZ(IM,1).EQ.LACK/IXYMN(IM)) IACPL=IACPL+IPSZ(IM,1)
          IF(IPSZ(IM,1).EQ.LACK/IXYMN(IM)) JACRQ(IM)=JACRQ(IM)-IPSZ(IM,1)
          IF(IPSZ(IM,1).EQ.LACK/IXYMN(IM)) LACK=0
          IF(LACK.EQ.0) GOTO 1241
C
          IPSZ(IM,1) = JACRQ(IM)
          LACK=LACK-(IPSZ(IM,1)*IXYMN(IM))
          IACPL = IACPL + IPSZ(IM,1)
          JACRQ(IM) = 0
C
  1241    CALL DATE(IM,IHAM,IHAD)
          IF(IPDD(IM,1).EQ.0) GOTO 1412
C
          IF((IPMD(IM,1).EQ.LPM).AND.(IPDD(IM,1).GT.LPD)) LPD=IPDD(IM,1)
C
          IF(IPMD(IM,1).GT.LPM) LPM=IPMD(IM,1)
          IF(IPMD(IM,1).GT.LPM) LPD = IPDD(IM,1)
C
          IHARD(IM,1) = IHARD(MARKA,MARKB)
          IHARM(IM,1) = IHARM(MARKA,MARKB)
  1245    CONTINUE
C
  1246  GOTO 1450
C
  1250  IHARD(MARKA+1,1) = IHARD(MARKA,MARKB)
        IHARM(MARKA+1,1) = IHARM(MARKA,MARKB)
C                                  RETURN,VIA CONTINUE,TO STMT 530
        GOTO1450
C                                  ASSIGN PLNTNG SIZE; RETURN TO STMT540
  1260  IPSZ(MARKA,MARKB) = JSSIZE(MARKA)
        JACRQ(MARKA) = JACRQ(MARKA) - JSSIZE(MARKA)
        IACPL = IACPL + IPSZ(MARKA,MARKB)
C
        GOTO  540
C
  1300  CONTINUE
C
        GOTO 1460
C                                  ERROR MESSGES HERE TO STMT 1460
  1330 WRITE(6,1360) IVAR(1)
  1360 FORMAT(1H ,*ERROR*,3X,*1ST PLANT MULTIPLE TOO LARGE*,3X,
     +      *FIRST PLANT SIZE EXCEEDS TOTAL ACRE REQMNT FOR VAR*,3X,A10)
C
       GOTO 360
C
  1390 WRITE(6,1410)
  1410 FORMAT(1H ,*FATAL ERROR*,3X,*FIRST PLANT CANNOT BE SCHEDULED*)
       GOTO 2720
```

```
C
  1412 WRITE(6,1415) IVAR(MO)
  1415 FORMAT(1H ,*FATAL ERROR*,3X,*FIRST PLNT OF*,2X,A10,2X,*NOT SCHEDULED
+      YET*)
       GOTO 2720
C
C
  1420 WRITE (6,1430) IVAR(MARKA+1)
  1430 FORMAT(1H ,*FATAL ERROR*,3X,*FIRST PLNT OF*,2X,A10,2X,*NOT SCHEDLD*)
       GOTO 2720
C
  1440 WRITE(6,1442) MARKA
  1442 FORMAT(1H ,*FATAL ERROR  ACRES REQUIRD FOR VAR*,2X,I2.2X,*EXCEEDS
+      SIMULATION LIMIT*)
       GOTO 2720
C
  1450 CONTINUE
  1460 CONTINUE
C                                  PRINT SCHEDLNG AND HARVEST INFO
       WRITE(6,1465)
  1465 FORMAT(1H ,7X,*VARIETY*,13X,*PLANTING NUMBER*,13X,*PLANTING DATE*,
+      13X,*PLANTING SIZE*)
C
       DO1475I=1,ISIM
       DO1475J=1,30
         IF(IPMD(I,J).EQ.0)GOTO 1475
         WRITE(6,1470) IVAR(I),J,IPMD(I,J),IPDD(I,J),IPSZ(I,J)
  1470   FORMAT(1X//6X,A10,17X,I2,25X,I1,3X,I2,18X,I4)
  1475   CONTINUE
C
       WRITE(6,1480)
  1480 FORMAT(1H ,7X,*VARIETY*,13X,*PLANTING NUMBER*,13X,*HARVEST DATE*)
C
       DO1495M=1,ISIM
       DO1495N=1,30
         IF(IHARM(M,N).EQ.0) GOTO 1495
         WRITE(6,1490) IVAR(M), N, IHARM(M,N), IHARD(M,N)
  1490   FORMAT(1X//6X,A10,17X,I2,25X,I1,3X,I2)
  1495   CONTINUE
C
  1520 CONTINUE
C                                  ZERO IN ARRAY "LIMIT"
       DO1530I=1,10
         LIMIT(I) = 0
  1530   CONTINUE
C
       WRITE(6,1570)
  1570 FORMAT(//,*ENTER FIELD NUMBER*)
C
       READ*,IFNUM
  1590 FORMAT(I2)
C                                  RECORD # PLNTNGS OF EACH VARIETY
       DO1610J=1,ISIM
       DO1610K=1,30
         IF(LIMIT(J).NE.0) GOTO 1610
         IF(IPSZ(J,K).EQ.0) LIMIT(J) = K-1
  1610   CONTINUE
C                                  READ RANDOM NO. SEEDS
```

```
      WRITE(6,1620)
 1620 FORMAT(//,*ENTER 2 RANDM NO. SEEDS:ODD INTGRS FRM 1 TO 1048576*)
      READ*,H,Y
 1625 FORMAT(I7,3X,I7)
C                                LOOP CALCS VALUES IN AWS FOR EACH VAR
      DO1840L=1,ISIM
        LIM=LIMIT(L)
C
        DO1840M=1,LIM
          COUNT = 0.0
C
          DO1840N=1,IFNUM
            DO1850IN=1,12
              AWS(IN)=0.0
 1850         CONTINUE
C
            IACTOT=0
C                                GENERATE KEY FOR DIRECT ACCESS FILE
            AWS(1)=FLOAT((L*10000)+(M*100)+N)
C                                DETRMNE HUS TO 100TR THIS SUBPLNT
 1628       RAND = RNDEV(H)
            IF(ABS(RAND).GT.FLOAT(4)) GOTO 1628
            AWS(2) = FLOAT(IHUMN(L))+(RAND*FLOAT(IHUSD(L)))
C                                CALC AWS(3)-AWS(10) VIA MATUR, FACTOR
            CALL MATUR(L,M,N)
C
            IF(AWS(3).EQ.0.0) GOTO 1800
C                                CALC FIELD SIZE THIS SUBPLNT
            IF((M.NE.1).OR.((M.EQ.1).AND.(L.EQ.1))) GOTO 1630
            LAST=LIMIT(L-1)
            IF((M.EQ.1).AND.((IPMD(L,M).EQ.IPMD(L-1,LAST)).AND.(IPDD(L,M
+           ).EQ.IPDD(L-1,LAST)))) GOTO 1650
C
 1630       IF((M.EQ.LIM).AND.((IPMD(L,M).EQ.IPMD(L+1,1)).AND.(IPDD(L,M)
+           .EQ.IPDD(L+1,1)))) GOTO 1710
C
            IACTOT = IACTOT + IPSZ(L,M)
            AWS(11)=FLOAT(IACTOT)/FLOAT(IFNUM)
            COUNT=COUNT+AWS(11)
C
C
            GOTO 1760
C
 1650       IACTOT=IACTOT + IPSZ(L,M)+IPSZ(L-1,LAST)
              IF(.NOT.((IPMD(L,M).EQ.IPMD(L-NM,LAST)).AND.(IPDD(L,M).EQ.
+             IPDD(L-NM,LAST)))) GOTO 1690
C
              IACTOT = IACTOT+IPSZ(L-NM,LAST)
 1670         CONTINUE
C
 1690       AWS(11)=FLOAT(IACTOT)/FLOAT(IFNUM)
            COUNT=COUNT+AWS(11)
            IF((COUNT.LT.FLOAT(IPSZ(L,M))).AND.(N.EQ.IFNUM)) TWS(11)=
+           FLOAT(IPSZ(L,M))-COUNT
            GOTO 1760
C
 1710       IACTOT=IACTOT+IPSZ(L,M)+IPSZ(L+1,1)
            DO1730NM=2,ISIM
              IF(.NOT.((IPMD(L,M).EQ.IPMD(L+NM,1)).AND.(IPDD(L,M).EQ.
+             IPDD(L+NM,1)))) GOTO 1750
              IACTOT=IACTOT+IPSZ(L+NM,1)
 1730         CONTINUE
```

```
C
  1750        AWS(11)=FLOAT(IACTOT)/FLOAT(IFNUM)
              COUNT=COUNT+AWS(11)
C                                     CALC YIELD OF THIS SUBPLNT
  1755        RAND = RNDEV(Y)
              IF(ABS(RAND).GT.FLOAT(4)) GOTO 1755
  1760        AWS(12) = (FLOAT(IXYMN(L))+(RAND*FLOAT(IXYSD(L))))*AWS(11)
              IF(AWS(12).LT.0.0) AWS(12) = 0.0
              AKEY=AWS(1)
C                                     STORE AWS ON DIRECT ACCESS FILE
              CALL PUT(TAPE4,AWS,0,0,0,0,EROUT1)
C
  1770        GOTO 1840
C
  1800        WRITE(6,1810) N, M, L
  1810        FORMAT(1H0,*ERROR*,2X,*SUBPLNT*,2X,I2,2X,*OF PLNTNG*,2X,I2,
+             2X,*FOR VAR*,2X,I2,2X,*NOT SCHEDULED; INSUFFICIENT TEMP DATA
+             *)
C
  1840    CONTINUE
C                                     REOPEN DIRECT ACCESS FILE FOR I/O
       CALL CLOSEM(TAPE4)
       CALL OPENM(TAPE4,3LI-O)
C                                     ZERO IN ARRAY "JYSUM"
       DO1870J=1,7
       DO1870K=1,31
         JYSUM(J,K) = 0
  1870    CONTINUE
C                                     SUM YIELDS ON EACH HARVEST DAY
       DO1890I=1,ISIM
         LIM=LIMIT(I)
C
         DO1890J=1,LIM
         DO1890K=1,IFNUM
         AKEY=FLOAT((I*10000)+(J*100)+K)
         CALL GET(TAPE4,AWS,AKEY,0,0,0,EROUT3)
         IF(AWS(3).EQ.0.0) GOTO 1890
         JMO=IFIX(AWS(4))-2
         JDAY=IFIX(AWS(3))
         JYSUM(JMO,JDAY)=JYSUM(JMO,JDAY)+IFIX(AWS(12))
  1890    CONTINUE
C
       WRITE(6,1910)
  1910 FORMAT(//,*ENTER MAX AND MIN DAILY CANNERY CAPACITY (LBS)*)
C
       READ*,MAX,MIN
  1930 FORMAT(I6,5X,I6)
C                                     ASSIGN TR 100 TO ALL SBPLNTS
       DO2070I=1,10
       DO2070J=1,30
       DO2070K=1,21
         ITRHAR(I,J,K)= 100
  2070    CONTINUE
C                                     READ OR ASSIGN % YIELD AT 80,140 TR
       WRITE(6,2080)
  2080 FORMAT(1H ,*ENTER PER CENT YIELD (DECIMAL) AT 80 AND 140 TR FOR*,
+      1X,*EACH VAR*)
```

```
C
      DO2105I=1,ISIM
        READ*,YFACT1(I),YFACT2(I)
 2090   FORMAT(F6.4,3X,F6.4)
 2105   CONTINUE
      DO 2115 J=1,ISIM
        IF(YFACT1(J).EQ.0.00) YFACT1(J) = 0.56
        IF(YFACT2(J).EQ.0.00) YFACT2(J) = 1.36
 2115   CONTINUE
C                                   LOOP RESCHEDLS HARVESTS TO 80,140TR
      I80TR=0
      I140TR=0
 2120 DO2290L=1,7
      DO2290M=1,31
        IF(JYSUM(L,M).LE.MAX) GOTO 2290
C
        DO2150IV=1,ISIM
          LIM=LIMIT(IV)
C
        DO2150IP=1,LIM
        DO2150IS=1,IFNUM
        AKEY=FLOAT((IV*10000)+(IP*100)+IS)
        CALL GET(TAPE4,AWS,AKEY,0.0,0,EROUT6)
        IWS(4) = IFIX(AWS(4))
        IWS(3) = IFIX(AWS(3))
        IF(.NOT.((IWS(4).EQ.(L+2)).AND.(IWS(3).EQ.M))) GOTO 2150
          IF((ITRHAR(IV,IP,IS).NE.100).AND.(IV.EQ.10).AND.(IP.EQ.30).AND
+         .(IS.EQ.21)) GOTO 2170
          IF(ITRHAR(IV,IP,IS).NE.100) GOTO 2150
C
        MARKE = IFIX(AWS(8))-2
        MARKF = IFIX(AWS(7))
        IF((JYSUM(MARKE,MARKF).GT.MAX).OR.((JYSUM(MARKE,MARKF)+IFIX(AWS(
+       12)*YFACT1(IV))).GT.MAX)) GOTO 2150
C
        DO2155LO=1,12
          TWS(LO)=AWS(LO)
 2155     CONTINUE
C
        JYSUM(L,M)=JYSUM(L,M)-IFIX(AWS(12))
        TWS(12)=TWS(12)*YFACT1(IV)
        I80TR=I80TR + IFIX(TWS(12))
          ITRHAR(IV,IP,IS)=80
        TWS(4)=AWS(8)
        TWS(3)=AWS(7)
        JYSUM(MARKE,MARKF)=JYSUM(MARKE,MARKF)+IFIX(TWS(12))
C                                   REPLC HARVST INFO FOR RESCHDLD SBPLNT
        CALL REPLC(TAPE4,TWS,120,0,0,0,EROUT7)
C
          IF(JYSUM(L,M).LE.MAX) GOTO 2290
 2150     CONTINUE
C
 2170   IF(JYSUM(L,M).LE.MAX) GOTO 2290
C
        DO2220IV=1,ISIM
          LIM=LIMIT(IV)
C
```

```
      DO2220IP=1,LIM
      DO2220IS=1,IFNUM
      AKEY=FLOAT((IV*10000)+(IP*100)+IS)
      CALL GET(TAPE4,AWS,AKEY,0,0,0,EROUT8)
      IWS(4) = IFIX(AWS(4))
      IWS(3) = IFIX(AWS(3))
      IF(.NOT.((IWS(4).EQ.(L+2)).AND.(IWS(3).EQ.M))) GOTO 2220
      IF((ITRHAR(IV,IP,IS).NE.100).AND.(IV.EQ.ISIM).AND.(IP.EQ.LIM)
+     .AND.(IS.EQ.IFNUM)) GOTO 2290
        IF(ITRHAR(IV,IP,IS).NE.100) GOTO 2220
C
      MARKE = IFIX(AWS(10))-2
      MARKF = IFIX(AWS(9))
      IF((JYSUM(MARKE,MARKF).GT.MAX).OR.((JYSUM(MARKE,MARKF)+IFIX(AWS(
+     12)*YFACT2(IV))).GT.MAX)) GOTO 2220
C
      DO2215LP=1,12
        TWS(LP)=AWS(LP)
 2215   CONTINUE
C
      JYSUM(L,M)=JYSUM(L,M)-IFIX(AWS(12))
      TWS(12)=TWS(12)*YFACT2(IV)
      I140TR=I140TR + IFIX(TWS(12))
        ITRHAR(IV,IP,IS)= 140
      TWS(4)=AWS(10)
      TWS(3)=AWS(9)
      JYSUM(MARKE,MARKF)=JYSUM(MARKE,MARKF)+IFIX(TWS(12))
C                             REPLC HARVST INFO FOR RESCHDLD SBPLNT
      CALLREPLC(TAPE4,TWS,120,0,0,0,EROUT9)
C
 2220   CONTINUE
C
 2290 CONTINUE
C
    CALL CLOSEM(TAPE4)
C
    CALL WASTE(MAX,MIN)
    CALL RANGE(IFNUM)
C                             SUM TOTAL YIELD FOR SEASON
    ITOTAL=0
    DO2320J=1,7
    DO2320K=1,31
      IF((J.LT.(IFHM-2)).OR.((J.EQ.(IFHM-2)).AND.(K.LT.IFHD)))GOTO2320
      IF((J.GT.(LHM-2)).OR.((J.EQ.(LHM-2)).AND.(K.GT.LHD))) GOTO 2330
      ITOTAL=ITOTAL+JYSUM(J,K)
 2320 CONTINUE
C
 2330 I100TR=ITOTAL-(I80TR+I140TR)
C                             OUTPUT FORMATS HERE TO PROGRAM'S END
    WRITE(6,2350)
 2350 FORMAT(1H ,13X,*MATUR*,3X,*SD MAT*,13X,*DISIRED*,3X,*EXPECTD*,4X,
+    *SD*,7X,*STRD*,4X,*YIELD*,5X,*YIELD*,4X,*TR DAY*,3X,*TR DAY*,5X,
+    *EARLY*,4X,*LATE*/* *,13X,*CONST*,4X,*CONST*,4X,*ACRES*,5X,*YIELD*,
+    5X,*YIELD*,3X,*EXPECTD*,4X,*PLNT*,6X,*AT*,8X,*AT*,5X,*BEFORE*,4X,
+    *AFTER*,5X,*HU TR*,3X,*HU TR*/* *,4X,*NAME*,6X,*HUS*,6X,*HUS*,5X,
+    *REQRD*,6X,*LBS*,7X,*PPA*,5X,*YIELD*,5X,*SIZE*,5X,*80TR*,5X,*140TR*,
+    5X,*100TR*,4X,*100TR*,4X,*CONVER*,2X,*CONVER*)
C
```

```
      DO2380L=1,ISIM
        WRITE(6,2370) IVAR(L),IHUMN(L),IHUSD(L),IACRQ(L),IDESY(L),IXYMN(
     +  L),IXYSD(L),ISSIZE(L),YFACT1(L),YFACT2(L),IFAC1(L),IFAC2(L),
     +  CONV1(L),CONV2(L)
 2370   FORMAT(1H ,A10,4X,I4,4X,I4,3X,I7,4X,I7,6X,I4,4X,I4,5X,I4,3X,F6.4
     +  ,4X,F6.4,6X,I2,7X,I2,8X,F4.2,4X,F4.2)
 2380   CONTINUE
C
      WRITE(6,2400) ISIM
 2400 FORMAT(1H ,*NO. OF CVS. IN SIMULATION:*,3X,I2)
C
      WRITE(6,2410) ICAP
 2410 FORMAT(1H ,*DAILY CANNERY CAPACITY:*,3X,I7)
C
      WRITE(6,2420) MAX
 2420 FORMAT(1H ,*MAX CANNERY CAPACITY:*,3X,I6)
C
      WRITE(6,2440) MIN
 2440 FORMAT(1H ,*MIN CANNERY CAPACITY:*,3X,I6)
C
      WRITE(6,2460) IHDM,IHDD
 2460 FORMAT(1H ,*HARVEST DEADLINE:*,3X,I2,3X,I2)
C
      WRITE(6,2480) MULT
 2480 FORMAT(1H ,*PLANTING MULTIPLE FOR 1ST PLANT:*,3X,I1)
C
      WRITE(6,2500) IFNUM
 2500 FORMAT(1H ,*FIELD NUMBER:*,3X,I2)
C
      WRITE(6,2520)
 2520 FORMAT(1H ,81X,*HUS DAY*,6X,*HUS DAY*,7X,*DATE*,9X,*DATE*/* *,4X
     +      *IDENT*,7X,*HARVEST*,7X,*FIELD*,8X,*TR AT*,8X,*YIELD*,8X,*HUS TO*,
     +      7X,*BEFORE*,7X,*AFTER*,9X,*OF*,11X,*OF*/* *,3X,*CV,P,SP*,7X,*DATE*,
     +      8X,*SIZE(A)*,7X,*HARVEST*,7X,*LBS*,10X,*100TR*,8X,*100TR*,7X,*100TR*,
     +      8X,*80TR*,9X,*140TR*)
C
      CALL OPENM(TAPE4,3LI-O)
C
      DO2560L=1,ISIM
        LIM=LIMIT(L)
      DO2560M=1,LIM
      DO2560N=1,IFNUM
        AKEY=FLOAT((L*10000)+(M*100)+N)
        CALL GET(TAPE4,AWS,AKEY,0,0,0,EROUT10)
        IWS(1)=IFIX(AWS(1))
        IWS(2)=IFIX(AWS(3))
        IWS(3)=IFIX(AWS(4))
        IWS(4)=IFIX(AWS(7))
        IWS(5)=IFIX(AWS(8))
        IWS(6)=IFIX(AWS(9))
        IWS(7)=IFIX(AWS(10))
C
      WRITE(6,2540) IWS(1),IWS(3),IWS(2),AWS(11),ITRHAR(L,M,N),AWS(12),
     +      AWS(2),AWS(5),AWS(6),IWS(5),IWS(4),IWS(7),IWS(6)
 2540 FORMAT(1H ,3X,I6,7X,I2,2X,I2,7X,F7.2,8X,I3,7X,F10.2,4X,F7.2,8X,F6.2,
     +      6X,F6.2,7X,I2,2X,I2,7X,I2,2X,I2)
C
 2560 CONTINUE
C
      CALL CLOSEM(TAPE4)
```

```
C
      WRITE(6,2590) IFHM, IFHD
 2590 FORMAT(1H ,*DATE OF FIRST HARVEST:*,3X,I2,2X,I2)
C
      WRITE(6,2610) LHM, LHD
 2610 FORMAT(1H ,*DATE OF LAST HARVEST:*,3X,I2,3X,I2)
C
      WRITE(6,2630) I80TR
 2630 FORMAT(1H ,*TOTAL LBS HARVESTED AT 80 TR:*,3X,I6)
C
      WRITE(6,2650) I100TR
 2650 FORMAT(1H ,*TOTAL LBS HARVESTED AT 100 TR:*,3X,I10)
C
      WRITE(6,2670) I140TR
 2670 FORMAT(1H ,*TOTAL LBS HARVESTED AT 140 TR:*,3X,I6)
C
      WRITE(6,2680) IWASTE
 2680 FORMAT(1H ,*TOTAL LBS WASTE DURING SEASON:*,3X,I10)
C
      WRITE(6,2685) ISLAKT
 2685 FORMAT(1H ,*TOTAL LBS SLACK DURING SEASON:*,3X,I10)
      WRITE(6,2690) ITOTAL
 2690 FORMAT(1H ,*TOTAL LBS HARVESTED FOR SEASON:*,3X,I10)
C
C
 2720 CONTINUE
      STOP
      END
C
C
C
C
      SUBROUTINE DDAYS
C
C     SUBROUTINE DDAYS IS A TEMPERATURE PROGRAM WHICH USES BOTH HISTOR-
C     ICAL DATA AND DATA FROM A CURRENT YEAR.  IT CALCULATES MEAN TEMPS
C     AND GROWING DEGREE DAYS FOR EVERY DAY FROM MAR 1 THRU SEPT 30 BOTH
C     HISTORICALLY AND FOR CURRENT YEAR. DEGREE DAYS ARE DETERMINED FROM
C     A USER SUPPLIED BASE TEMP. IN ADDITION, THE SUBROUTINE STORES
C     ACCUMULATED DEGREE DAYS EVERY DAY FOR THE SAME PERIOD BOTH HISTOR-
C     ICALLY AND IN CURRENT YEAR.
C
C     ALPHABETICAL DICTIONARY, SUBROUTINE DDAYS:
C     DD  DEGREE DAYS
C     DDREF  BASE TEMPERATURE
C     DDTOT  COUNTER USED TO SUM DEGREE DAYS THRU SEASON
C     IA  CONVERTS ARRAY MONTH TO CALENDAR MONTH
C     IDAY  READS IN DAY FROM TEMP DATA; 1 DIMENS ARRAY
C     IMO  1 DIMENS ARRAY TO READ IN MONTH FROM TEMP DATA
C     ITMN  1 DIMENS ARRAY TO READ IN MIN TEMP FOR DAY
C     ITMX  1 DIMENS ARRAY TO READ IN MAX TEMP FOR DAY
C     IYEAR  NON-ARRAY REPRESENTATION OF YR OF TEMP DATA
C     IYR  1 DIMENS ARRAY TO READ IN YEAR OF TEMP DATA
C     L  WHEN VALUE IS ONE, DATA PROCESSED IS HISTORICAL, WHEN L IS 2
C        CURRENT YEAR DATA IS USED
C     MONTH  NON-ARRAY REPRESENTATN OF MONTH OF TEMP DATA
C     N  IS TAPE TEMP DATA IS READ FROM--EITHER CURRENT OR HISTORICAL
C     NDAY  NON-ARRAY REPRESENTATN OF DAY OF TEMP DATA
C     NYR  CALENDAR YR OF TEMP DATA
C     TDD  AVG DEGREE DAYS MAR 1 TO SEPT 30 HISTOR DATA
```

```
C       TDD1  AVG DEGREE DAYS MAR 1 TO SEPT 30 CURRENT DATA
C       TDDTOT  2 DIMENS ARRAY OF ACCUM'D HU'S MAR THRU SEPT HISTOR DATA
C       TDDTO1  2 DIMENS ARRAY OF ACCUM'D HU'S MAR THRU SEPT CURRENT DATA
C       TM  2 DIMENS ARRAY OF MEAN TEMPS MAR THRU SEPT HISTOR DATA
C       TM1  2 DIMENS ARRAY OF MEAN TEMPS MAR THRU SEPT CURRENT DATA
C       TMEAN  MEAN TEMPERATURE
C       YRSUM  COUNTS TOTAL NO. OF YRS IN TEMP DATA
C
C       OTHER VARIABLES WHICH APPEAR IN COMMON BUT AREN'T DEFINED HERE ARE
C       NOT USED IN SUBROUTINE DDAYS
C
        DIMENSION ITMX(5), ITMN(5), IYR(5), IMO(5), IDAY(5)
C
        COMMON DDREF, TM(7,31), TDD(7,31), TDDTOT(7,31), TM1(7,31)
        COMMON TDD1(7,31),TDDTO1(7,31),IPDD(10,30),IPMD(10,30), IHUMN(10)
C                                      HIST OR CURRNT DATA USE SET BY N, L
        IF(DDREF.EQ.0.0)  N=2
        IF(DDREF.EQ.0.0)  L=2
        IF(DDREF.NE.0.0)  N=1
        IF(DDREF.NE.0.0)  L=1
C                                      READ TEMP INFO
        READ(N,1530) NYR, DDREF
   1530 FORMAT(2X,I2,2X,F4.0)
C
        WRITE(6,1532)NYR, DDREF
   1532 FORMAT(1X,I2,3X,F4.0)
        YRSUM = 1.0
        DDTOT = 0.0
C
   1540 CONTINUE
        READ(N,1560) ((ITMX(I),ITMN(I),IYR(I),IMO(I),IDAY(I)),I=1,5)
   1560 FORMAT(5(2I4,1X,3I2,1X))
        IF(EOF(N)) 1630,1580
   1580 CONTINUE
C
        DO1600I=1,5
          IYEAR=IYR(I)
          IF(IYEAR.EQ.0) GOTO 1600
          IF(IYEAR.NE.NYR) YRSUM=YRSUM+1.0
          IF(IYEAR.NE.NYR) NYR=IYEAR
C                                      CONVERT CALENDAR MO TO ARRAY MONTH
          MONTH=IMO(I)
          IF(MONTH.EQ.3) IA=1
          IF(MONTH.EQ.4) IA=2
          IF(MONTH.EQ.5) IA=3
          IF(MONTH.EQ.6) IA=4
          IF(MONTH.EQ.7) IA=5
          IF(MONTH.EQ.8) IA=6
          IF(MONTH.EQ.9) IA=7
C                                      CALC MEAN TEMPS AND SUM FOR EACH DATE
          NDAY=IDAY(I)
          TMEAN=(FLOAT(ITMX(I)) + FLOAT(ITMN(I)))/2.0
          IF(L.EQ.2) TM1(IA,NDAY)=TM1(IA,NDAY) + TMEAN
          IF(L.EQ.1) TM(IA,NDAY)=TM(IA,NDAY) + TMEAN
   1600 CONTINUE
C
        GOTO 1540
C                                      FOR HIST DATA: CALC DDAYS, ACCUMD HUS
```

```
  1630 CONTINUE
       IF(YRSUM.EQ.1.0) GOTO 1700
       DO1650J=1,7
       DO1650K=1,31
         IF(((J.EQ.2).OR.(J.EQ.4).OR.(J.EQ.7)).AND.(K.EQ.31)) GOTO 1650
         TM(J,K) = TM(J,K)/YRSUM
C
         DD=TM(J,K) - DDREF
         IF(DD.LT.0.0) DD=0.0
         TDD(J,K) =DD
         DDTOT=DDTOT + DD
         TDDTOT(J,K) = DDTOT
  1650   CONTINUE
C
       GOTO 1900
C                                      FOR CURR DATA: CALC DDAYS, ACCUMD HUS
  1700 DO1800J=1,7
       DO1800K=1,31
         IF(((J.EQ.2).OR.(J.EQ.4).OR.(J.EQ.7)).AND.(K.EQ.31)) GOTO 1800
         TM1(J,K) = TM1(J,K)/YRSUM
C
         DD= TM1(J,K)-DDREF
         IF(DD.LT.0.0) DD=0.0
         TDD1(J,K) = DD
         DDTOT = DDTOT + DD
         TDDTO1(J,K) = DDTOT
  1800   CONTINUE
C
  1900 CONTINUE
C
       RETURN
       END
C
C
       SUBROUTINE DATE (N,NHAM,NHAD)
C
C      SUBROUTINE DATE DETERMINES PLANTING DATE FOR PLANTINGS OF A NEW
C      VAR NECESSITATED WHEN PARTIAL PLANTING OF PRECEEDING VAR IS MADE
C
C      N  VARIETY NUMBER
C      NHAD  PROJECTED HARVEST DAY OF VARIETY N
C      NHAM  PROJECTED HARVEST MONTH OF VARIETY N
C      ALL OTHER VARIABLES USED ARE IN COMMON AND THEIR DEFINITIONS MAY
C      BE FOUND IN DICTIONARY FOR MAIN PROGRAM BY DETERMINING THE ANALA-
C      GOUS COMMON LOCATION
C
       COMMON SREF, SMH(7,31), SDDH(7,31), SDDTOH(7,31), SMC(7,31), SDDC(
+      7,31), SDDTOC(7,31),NPDD(10,30),NPMD(10,30), NHUMN(10)
C                                      DOES PLNT DAY FALL WITHIN HARVST MO?
       DO1920I=1,30
         IF(NPDD(N,1).NE.0) GOTO 1960
         IF(I.GE.NHAD) GOTO 1940
C
         IF((SDDTOH(NHAM,NHAD)-SDDTOH(NHAM,NHAD-I)).GE.FLOAT(NHUMN(N)))
+        NPDD(N,1) = NHAD-I
  1920   CONTINUE
C
  1940 IF(NPDD(N,1).EQ.0) GOTO 1980
```

```
C
 1960 NPMD(N,1) = NHAM + 2
      WRITE(6,1961) N, NPMD(N,1), NPDD(N,1)
 1961 FORMAT(1H ,*1ST PLNT VAR*,2X,I2,2X,*SCHEDULED*,2X,I1,2X,I2)
      GOTO3100
C
 1980 IF(NHAM.EQ.1) GOTO 2080
C                                 DETRMNE PLNTNG DAY AND MONTH
      DO2020J=1,6
      DO2020K=1,31
        IF(NPDD(N,1).NE.0) GOTO 3000
        IF(((NHAM-J.EQ.2).OR.(NHAM-J.EQ.4)).AND.(K.EQ.31)) GOTO 2020
        IF(((NHAM-J.EQ.2).OR.(NHAM-J.EQ.4)).AND.(K.NE.31)) GOTO 2000
C
        IF((SDDTOH(NHAM,NHAD)-SDDTOH(NHAM-J,32-K)).GE.FLOAT(NHUMN(N)))
+       NPDD(N,1) = 32-K
        IF(NPDD(N,1).EQ.(32-K)) NPMD(N,1) = NHAM-J+2
        GOTO 2020
C
 2000   IF((SDDTOH(NHAM,NHAD)-SDDTOH(NHAM-J,31-K)).GE.FLOAT(NHUMN(N)))
+       NPDD(N,1) = 31-K
        IF(NPDD(N,1).EQ.(31-K)) NPMD(N,1) = NHAM-J+2
 2020   CONTINUE
C
      GOTO 3000
C
 2080 WRITE(6,2085) N
 2085 FORMAT(1H0,*INSUFFICIENT TEMP DATA AVAILABLE; FIRST PLNT OF
+     VAR*,2X,I2,2X,*CANNOT BE SCHEDULED*)
      GOTO 3100
C
 3000 WRITE(6,3010) N, NPMD(N,1), NPDD(N,1)
 3010 FORMAT(1H ,*1ST PLANT VAR*,2X,I2,2X,*SCHEDULED*,2X,I2,2X,I2)
C
 3100 RETURN
      END
C
C
C
      SUBROUTINE EXCEED(NARKA,NARKB)
C
C     SUBROUTINE EXCEED CHECKS TO SEE IF THE PREDICTED HARVEST DATE FOR
C     A GIVEN PLNTNG OR SUBPLANTING EXCEEDS HARVEST DEADLINE.  IF SO, THE
C     NO., OF DAYS BY WHICH DEADLINEIS EXCEEDED IS CALCULATED.
C
C     NARKA, NARKB ARE VAR AND PLANTING NOS. ALL OTHER VARIABLES FROM
C     COMMON AND DEFINED IN MAIN PROGRAM DICTIONARY.
C
      LOGICAL B
      COMMON BASE, TMH(7,31), TDDH(7,31), TDDTOH(7,31), TMC(7,31), TDDC(
+     7,31),TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),JX,IHARM(10,
+     30),IHARD(10,30),JHDM,JHDD,TAPE4(35),AWS(12),TWS(12),IWS(7),ISIM,
+     AVGLO(10,30,21),AVGHI(10,30,21),CONV1(10),CONV2(10),IFAC1(10),
+     IFAC2(10),JYSUM(7,31),LIMIT(10),IWASTE,IFHM,IFHD,LHD,LHM,B
C
      IF(B) GOTO 100
C                                 FOR HIST HARVESTS ONLY
      IF((IHARM(NARKA,NARKB).EQ.JHDM).AND.(IHARD(NARKA,NARKB).GT.JHDD)
+     ) JX = IHARD(NARKA,NARKB)-JHDD
C
```

```
      IF((IHARM(NARKA,NARKB).GT.JHDM).AND.((IHARM(NARKA,NARKB).EQ.4)
     +   .OR.(IHARM(NARKA,NARKB).EQ.6).OR.(IHARM(NARKA,NARKB).EQ.8).OR.
     +   (IHARM(NARKA,NARKB).EQ.9))) JX=(31-JHDD)+IHARD(NARKA,NARKB)
C
      IF((IHARM(NARKA,NARKB).GT.JHDM).AND.((IHARM(NARKA,NARKB).EQ.3).OR.
     +   (IHARM(NARKA,NARKB).EQ.5).OR.(IHARM(NARKA,NARKB).EQ.7))) JX = (30-
     +   JHDD) + IHARD(NARKA,NARKB)
      GOTO 200
C                                 FOR ACTUAL HARVESTS ONLY
  100 IWS(4)=IFIX(AWS(4))
      IWS(3)=IFIX(AWS(3))
      IF((IWS(4).EQ.JHDM).AND.(IWS(3).GT.JHDD)) JX=IWS(3)-JHDD
C
      IF((IWS(4).GT.JHDM).AND.((IWS(4).EQ.4).OR.(IWS(4).EQ.6).OR.(IWS(4)
     +   .EQ.8).OR.(IWS(4).EQ.9))) JX=(31-JHDD)+IWS(3)
C
      IF((IWS(4).GT.JHDM).AND.((IWS(4).EQ.3).OR.(IWS(4).EQ.5).OR.(IWS(4)
     +   .EQ.7))) JX=(30-JHDD)+IWS(3)
  200 RETURN
      END
C
C
C
      SUBROUTINE MATUR(LL,MM,NN)
C
C     SUBROUTINE MATUR DETERMINES ACTUAL DAY OF HARVEST FOR A GIVEN
C       SUBPLANT BASED ON ACTUAL HEAT UNITS TO MATURITY (100 TR). IT
C       CALLS SUBROUTINE EXCEED TO DETERMINE WHETHER HARVEST DAY EXCEEDS
C       HARVEST DEADLINE. IT CALLS SUBROUTINE FACTOR TO DETERMINE
C       AVG NO. OF HEAT UNITS AT TIME OF HARVEST. IT ALSO DETERMINES
C       DATES ON WHICH THE GIVEN SUBPLANT REACHES 80 TR AND 140 TR.
C
C     CONV1, CONV2  ARE OPTIONAL CONVERSION FACTORS SUPPLIED BY USER TO
C       CONVERT HEAT UNITS/DAY AT HARVEST TO TR POINTS/DAY. CONV1 IS
C       USED PRIOR TO 100 TR, CONV2 AFTER 100 TR.
C     IDIFF  IS DEFINED ONLY IF A SUBPLNT REACHES 80TR IN A DIFFERENT
C       MONTH THAN THAT IN WHICH IT REACHES 100TR. IDIFF IS THEN THENO.
C       OF DAYS FROM 80TR TO THE END OF THE MONTH IN WHICH 80TR OCCURS.
C     IFAC1, IFAC2  ARE THE NO. OF TR POINTS ADDED/DAY DURING HARVEST
C       PRIOR TO 100 TR AND AFTER, RESPECTIVELY.
C     JHAD, JHAM  ARE MARKER VARIABLES REPRESENTINGTHE ACTUAL HARVEST
C       DAY AND MONTH (RESPECTIVELY) FOR A GIVEN SUBPLANT
C     LL  IS THE VARIABLE DENOTING VARIETY BEING HARVESTED
C     MARKC, MARKD  ARE"DUMMY"VARIABLES USED TO REPRESENT PLNTING DAY
C       AND MONTH (RESPECTIVELY) FOR PLNTING BEING PROCESSED
C     MM  IS THE VARIABLE DENOTING PLANTING BEING HARVESTED
C     NN  IS THE VARIABLE DENOTING SUBPLNTING BEING HARVESTED
C
C     ALL OTHER VARIABLES USED IN SUBROUTINE MATUR ARE FROM COMMON BLOCK
C       AND THEIR DEFINITIONS MAY BE FOUND ELSEWHERE.
C
      LOGICAL B
      DIMENSION CONV1(10), CONV2(10), IFAC1(10), IFAC2(10)
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
     +   ,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
     +   IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7),ISIM,
     +   AVGLO(10,30,21),AVGHI(10,30,21),CONV1,CONV2,IFAC1,IFAC2,JYSUM(7,31
     +   ),LIMIT(10),IWASTE,IFHM,IFHD,LHD,LHM,B
C
```

```
      JHAD = 0
      JHAM = 0
      MARKC=IPDD(LL,MM)
      MARKD = (IPMD(LL,MM)-2)
C                                   DETRMNE HARVEST DAY, MONTH
      DO50J=1,7
      DO50K=1,31
        IF(JHAD.NE.0) GOTO 70
        IF(((J.EQ.2).OR.(J.EQ.4).OR.(J.EQ.7)).AND.(K.EQ.31)) GOTO 50
C
        IF((((K.GT.MARKC).AND.(J.EQ.MARKD)).OR.(J.GT.MARKD)).AND.
+       ((TDDTOC(J,K)-TDDTOC(MARKD,MARKC)).GE.AWS(2))) JHAD=K
C
        IF(JHAD.EQ.K) JHAM = J+2
  50    CONTINUE
C
      IF(JHAD.EQ.0) GOTO 550
C
  70  AWS(3)=FLOAT(JHAD)
      AWS(4)=FLOAT(JHAM)
C
      IX = 0
      B = .TRUE.
C                                   DOES HARVST DATE EXCEED DEADLINE?
      CALL EXCEED(LL,MM)
      IF(IX.NE.0) GOTO 380
C                                   CALC AVG HU'S AT HARVEST
  90  CALL FACTOR(JHAD,JHAM,LL,MM,NN)
C
      IF(IFAC1(LL).NE.0) GOTO 190
C                                   1 TIME/VAR: READ TR/DAY NEAR 100 TR
      DO130J=1,ISIM
        WRITE(6,110) J
 110    FORMAT(//,*ENTER INCREASE IN TR/DAY BEFORE 100 TR AND AFTER*,1X,
+       *FOR VAR*,3X,I2)
C
      READ*,IFAC1(J),IFAC2(J)
 120    FORMAT(I2,3X,I2)
 130    CONTINUE
C                                   READ HU'S/DAY TO TR PTS/DAY IF NEEDED
      DO140I=1,ISIM
        IF(IFAC1(I).NE.0) GOTO 137
        WRITE(6,135) I
 135    FORMAT(1H ,*ENTER CONVERSION FACTOR FOR HU'S/DAY TO TR*,1X,
+       *POINTS/DAY BEFORE 100 TR FOR VAR*,3X,I2)
C                                   CALC TR/DAY BEFORE 100 TR
      READ*,CONV1(I)
 136    FORMAT(F4.2)
        IFAC1(I)=IFIX(AVGLO(LL,MM,NN)/CONV1(I))
C
 137    IF(IFAC2(I).NE.0)  GOTO 140
        WRITE(6,138) I
 138    FORMAT(1H ,*ENTER CONVERSION FACTOR FOR HU'S/DAY TO TR*,1X,
+       *POINTS/DAY AFTER 100 TR FOR VAR*,3X,I2)
C                                   CALC TR/DAY AFTER 100 TR
      READ*,CONV2(I)
 139    FORMAT(F4.2)
        IFAC2(I)=IFIX(AVGHI(LL,MM,NN)/CONV2(I))
 140    CONTINUE
C                                   CALC DATE SBPLNT REACHES 80 TR
```

```
  190 IF((JHAD-(20/IFAC1(LL))).GE.1)AWS(7)=FLOAT(JHAD-(20/IFAC1(LL)))
      IF((JHAD-(20/IFAC1(LL))).GE.1)AWS(8)=FLOAT(JHAM)
C
      IF(AWS(7).NE.0.0) GOTO 220
C
      IF((JHAD-(20/IFAC1(LL))).LT.1) IDIFF=IABS(JHAD-(20/IFAC1(LL)))
      IF(((JHAD-(20/IFAC1(LL))).LT.1).AND.((JHAM-2.EQ.2).OR.(JHAM-2.EQ.4
+     ).OR.(JHAM-2.EQ.6).OR.(JHAM-2.EQ.7)))AWS(7)=FLOAT(31-IDIFF)
C
      IF(AWS(7).EQ.(FLOAT(31-IDIFF))) AWS(8)=FLOAT(JHAM-1)
      IF(AWS(7).NE.0.0) GOTO 220
C
      AWS(7)=FLOAT(30-IDIFF)
      AWS(8)=FLOAT(JHAM-1)
C                                  CALC DATE SBPLNT REACHES 140 TR
 220 IF((JHAD+(40/IFAC2(LL))).LE.30)AWS(9)=FLOAT(JHAD+(40/IFAC2(LL)))
      IF((JHAD+(40/IFAC2(LL))).LE.30)AWS(10)=FLOAT(JHAM)
      IF(AWS(9).NE.0.0) GOTO 300
C
      IF(((JHAD+(40/IFAC2(LL))).EQ.31).AND.((JHAM-2.EQ.1).OR.(JHAM-2.EQ.
+     3).OR.(JHAM-2.EQ.5).OR.(JHAM-2.EQ.6))) AWS(9)=31.0
      IF(AWS(9).EQ.31.0) AWS(10)=FLOAT(JHAM)
C
      IF(((JHAD+(40/IFAC2(LL))).EQ.31).AND.((JHAM-2.EQ.2).OR.(JHAM-2.EQ.
+     4).OR.(JHAM-2.EQ.7))) AWS(9)=1.0
      IF(AWS(9).EQ.1.0) AWS(10)=FLOAT(JHAM+1)
C
      IF(AWS(9).NE.0.0) GOTO 300
C
      IF(((JHAD+(40/IFAC2(LL))).GT.31).AND.((JHAM-2.EQ.2).OR.(JHAM-2.EQ.
+     4).OR.(JHAM-2.EQ.7))) AWS(9)=FLOAT((JHAD+(40/IFAC2(LL)))-30)
C
      IF(((JHAD+(40/IFAC2(LL))).GT.31).AND.((JHAM-2.EQ.1).OR.(JHAM-2.EQ.
+     3).OR.(JHAM-2.EQ.5).OR.(JHAM-2.EQ.6)))AWS(9)=FLOAT((JHAD+(40/IFAC2
+     (LL)))-31)
C
      AWS(10)=FLOAT(JHAM+1)
C
  300 IF((AWS(7).EQ.0.0).OR.(AWS(8).EQ.0.0)) GOTO 440
C
  330 IF((AWS(9).EQ.0.0).OR.(AWS(10).EQ.0.0)) GOTO 470
C
      GOTO 550
C
  380 WRITE(6,390) NN,MM,LL,IX
  390 FORMAT(1H0,*CURRENT HARVEST DATE FOR SUBPLNT*,2X,I2,2X,*OF PLNTNG*
+     ,2X,I2,2X,*OF VAR*,2X,I2,2X,*EXCEEDS HARVEST DEADLINE BY*,2X,I4,2X
+     ,*DAYS*)
      GOTO 90
C
  440 WRITE(6,450) NN,MM,LL
  450 FORMAT(1H0,*ERROR: DATE OF 80 TR FOR SUBPLNT*,2X,I2,2X,*OF PLNTNG*
+     ,2X,I2,2X,*OF VAR*,2X,I2,2X,*NOT DETERMINED*)
      GOTO 330
C
  470 WRITE(6,480) NN,MM,LL
  480 FORMAT(1H0,*ERROR: DATE OF 140 TR FOR SUBPLNT*,2X,I2,2X,*OF PLNTNG
+     *,2X,I2,2X,*OF VAR*,2X,I2,2X,*NOT DETERMINED*)
C
  550 RETURN
      END
```

```
C
C
      SUBROUTINE FACTOR(KHAD,KHAM,L1,M1,N1)
C
C     SUBROUTINE FACTOR CALCULATES AVG HU'S/DAY AT HARVEST. IT RETURNS 2
C     VALUES. AVGLO GIVES THE AVG NO. OF HU'S FOR 5 DAYS PRECEEDNG 100TR
C     FOR A GIVEN SUBPLNT; AVGHI GIVES THE AVG NO. OF HU'S FOR 5 DAYS
C     FOLLOWING 100TR FOR A GIVEN SUBPLNT.
C
C
C     KHAD,KHAM ARE ACTUAL HARVEST DAY AND MONTH AND ARE PASSED FROM SUB
C     ROUTINE MATUR. ALL OTHER VARIABLES FROM COMMON AND DEFINED IN MAIN
C     PROGRAM DICTIONARY.
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31
     +),TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
     +IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7),ISIM,
     +AVGLO(10,30,21),AVGHI(10,30,21)
C                                     AWS(5)=AVG HUS 5 DAYS BEFORE 100TR
      IF(KHAD.GE.6) AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-2,
     +KHAD-2)+TDDC(KHAM-2,KHAD-3)+TDDC(KHAM-2,KHAD-4)+TDDC(KHAM-2,KHAD-5
     +))/5.0
C
      IF(AWS(5).NE.0.0) GOTO 80
C
      IF((KHAM-2.EQ.1).AND.(KHAD.LE.5)) AWS(5)=TDDC(KHAM-2,KHAD)
C
      IF(AWS(5).NE.0.0) GOTO 80
C
      IF(((KHAM-2).EQ.2).OR.((KHAM-2).EQ.4).OR.((KHAM-2).EQ.6).OR.((KHAM
     +-2).EQ.7)) GOTO 50
C                                     5 STMTS FOR MO.S WITH 30 DAYS
      IF(KHAD.EQ.5) AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-2,
     +KHAD-2)+TDDC(KHAM-2,KHAD-3)+TDDC(KHAM-2,KHAD-4)+TDDC(KHAM-3,30))/
     +5.0
C
      IF(KHAD.EQ.4) AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-2,
     +KHAD-2)+TDDC(KHAM-2,KHAD-3)+TDDC(KHAM-3,30)+TDDC(KHAM-3,29))/5.0
C
      IF(KHAD.EQ.3) AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-2,
     +KHAD-2)+TDDC(KHAM-3,30)+TDDC(KHAM-3,29)+TDDC(KHAM-3,28))/5.0
C
      IF(KHAD.EQ.2) AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-3,30)
     ++TDDC(KHAM-3,29)+TDDC(KHAM-3,28)+TDDC(KHAM-3,27))/5.0
C
      IF(KHAD.EQ.1) AWS(5)=(TDDC(KHAM-3,30)+TDDC(KHAM-3,29)+
     +TDDC(KHAM-3,28)+TDDC(KHAM-3,27)+TDDC(KHAM-3,26))/5.0
C
      GOTO 80
C                                     5 STMTS FOR MO.S WITH 31 DAYS
   50 IF(KHAD.EQ.5) AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-2,
     +KHAD-2)+TDDC(KHAM-2,KHAD-3)+TDDC(KHAM-2,KHAD-4)+TDDC(KHAM-3,31))/
     +5.0
C
      IF(KHAD.EQ.4)AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-2,
     +KHAD-2)+TDDC(KHAM-2,KHAD-3)+TDDC(KHAM-3,31)+TDDC(KHAM-3,30))/5.0
C
      IF(KHAD.EQ.3)AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-2,
     +KHAD-2)+TDDC(KHAM-3,31)+TDDC(KHAM-3,30)+TDDC(KHAM-3,29))/5.0
C
      IF(KHAD.EQ.2)AWS(5)=(TDDC(KHAM-2,KHAD-1)+TDDC(KHAM-3,31)
     ++TDDC(KHAM-3,30)+TDDC(KHAM-3,29)+TDDC(KHAM-3,28))/5.0
```

```
C
      IF(KHAD.EQ.1)AWS(5)=(TDDC(KHAM-3,31)+TDDC(KHAM-3,30)+
     +     TDDC(KHAM-3,29)+TDDC(KHAM-3,28)+TDDC(KHAM-3,27))/5.0
C
   80 IF(AWS(5).EQ.0.0) GOTO 250
C
  100 IF(((((KHAM-2).EQ.1).OR.((KHAM-2).EQ.3).OR.((KHAM-2).EQ.5).OR.((
     +     KHAM-2).EQ.6)).AND.(KHAD.EQ.26)).OR.(KHAD.LE.25))AWS(6)=
     +     (TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-2,KHAD+2)+TDDC(KHAM-2,KHAD+3)+TDDC(
     +     KHAM-2,KHAD+4)+TDDC(KHAM-2,KHAD+5))/5.0
C
      IF(AWS(6).NE.0.0) GOTO 400
C
      IF(KHAM-2.EQ.7) AWS(6)=TDDC(KHAM-2,KHAD)
C
      IF(((KHAM-2).EQ.2).OR.((KHAM-2).EQ.4)) GOTO 200
C                                     5 STMTS FOR MO.S WITH 31 DAYS
      IF(KHAD.EQ.27) AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-2,
     +     KHAD+2)+TDDC(KHAM-2,KHAD+3)+TDDC(KHAM-2,KHAD+4)+TDDC(KHAM-1,1))/
     +     5.0
C
      IF(KHAD.EQ.28) AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-2,
     +     KHAD+2)+TDDC(KHAM-2,KHAD+3)+TDDC(KHAM-1,1)+TDDC(KHAM-1,2))/5.0
C
      IF(KHAD.EQ.29) AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-2,
     +     KHAD+2)+TDDC(KHAM-1,1)+TDDC(KHAM-1,2)+TDDC(KHAM-1,3))/5.0
C
      IF(KHAD.EQ.30) AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-1,1)+
     +     TDDC(KHAM-1,2)+TDDC(KHAM-1,3)+TDDC(KHAM-1,4))/5.0
C
      IF(KHAD.EQ.31)AWS(6)=(TDDC(KHAM-1,1)+TDDC(KHAM-1,2)+TDDC
     +     (KHAM-1,3)+TDDC(KHAM-1,4)+TDDC(KHAM-1,5))/5.0
C
      GOTO 290
C                                     5 STMTS FOR MO.S WITH 30 DAYS
  200 IF(KHAD.EQ.26) AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-2,
     +     KHAD+2)+TDDC(KHAM-2,KHAD+3)+TDDC(KHAM-2,KHAD+4)+TDDC(KHAM-1,1))/
     +     5.0
C
      IF(KHAD.EQ.27) AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-2,
     +     KHAD+2)+TDDC(KHAM-2,KHAD+3)+TDDC(KHAM-1,1)+TDDC(KHAM-1,2))/5.0
C
      IF(KHAD.EQ.28) AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-2,
     +     KHAD+2)+TDDC(KHAM-1,1)+TDDC(KHAM-1,2)+TDDC(KHAM-1,3))/5.0
C
      IF(KHAD.EQ.29)AWS(6)=(TDDC(KHAM-2,KHAD+1)+TDDC(KHAM-1,1)+
     +     TDDC(KHAM-1,2)+TDDC(KHAM-1,3)+TDDC(KHAM-1,4))/5.0
C
      IF(KHAD.EQ.30) AWS(6)=(TDDC(KHAM-1,1)+TDDC(KHAM-1,2)+TDDC
     +     (KHAM-1,3)+TDDC(KHAM-1,4)+TDDC(KHAM-1,5))/5.0
C
      GOTO 290
C                                     ERROR MESSAGES FOLLOW
  250 WRITE(6,260) L1,M1,N1
  260 FORMAT(1H ,*ERROR  HU'S NOT AVGD FOR 5 DAYS PRECEDNG HARVEST OF
     +     VAR*,2X,I2,2X,*PLNTNG*,2X,I2,2X,*SUBPLNT*,2X,I2)
C
      GOTO 100
C
  290 IF(AWS(6).NE.0.0) GOTO 400
```

```
C
      WRITE(6,320) L1,M1,N1
  320 FORMAT(1H0,*ERROR: HU'S NOT AVGD FOR 5 DAYS FOLLOWING HARVEST OF
     +VAR*,2X,I2,2X,*PLNTNG*,2X,I2,2X,*SUBPLNT*,2X,I2)
C                                     AWS(5) AND AWS(6) STORED BY SUBPLNT
  400 AVGLO(L1,M1,N1)=AWS(5)
      AVGHI(L1,M1,N1)=AWS(6)
C
      RETURN
      END
C
C
C
      FUNCTION RNDEV(X2)
C     FUNCTION TO DEVELOP RANDOM NORMAL VARIATES FROM RECTANGULAR VAR
C
C     U1 AND U2 ARE RANDOM VARIABLES FROM THE SAME RECTANGULAR DENSITY F
C     RNDEV AND X2 ARE PAIR OF RANDOM DEVIATES FROM THE SAME NORMAL DIST
C     RNDEV AND X2 ARE INDEPENDENT
C
      U1 = RANF(1)
      U2 = RANF(1)*6.2831853071
      A = (-2.*ALOG(U1))**.5
      X2=A*SIN(U2)
      RNDEV = A*COS(U2)
C
      RETURN
      END
C
C
C
      SUBROUTINE EROUT1
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
     +,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
     +IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
      WRITE(6,10) AWS(1)
   10 FORMAT(1H0,*ERROR IN DA FILE INPUT BETWEEN STMTS 1760 AND 1770;
     +KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT1)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT1*,3(O20,4X))
C
      RETURN
      END
C
C
C
      SUBROUTINE EROUT2
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
     +,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
     +IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
      WRITE(6,10) TWS(1)
C
   10 FORMAT(1H0,*ERROR IN DA FILE INPUT BETWEEN STMTS 1770 AND 1800;
     +KEY IS*,2X,F7.1)
C
```

```
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT2)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT2*,3(O20,4X))
C
      RETURN
      END
C
C
C
      SUBROUTINE EROUT3
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
+     ,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
+     IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
C
      WRITE(6,10) AWS(1)
   10 FORMAT(1HO,*ERROR IN RETRIEVAL FROM DA FILE BETWEEN STMTS 1870
+     AND 1890; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT. MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1HO,*ERROR(EROUT3)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1HO,*FIT FROM EROUT3*,3(O20,4X))
C
      RETURN
      END
C
C
C
      SUBROUTINE EROUT4
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
+     ,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
+     IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
C
      WRITE(6,10) AWS(1)
   10 FORMAT(1HO,*ERROR IN RETRIEVAL FROM DA FILE BETWEEN STMTS 2010 AND
+     2030; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT4)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT4*,3(O20,4X))
C
      RETURN
      END
C
C
C
      SUBROUTINE EROUT5
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
+     ,TDDTOC(7,31),IPDD(10,30),IPMN(10,30),IHUMN(10),IX,IHARM(10,30),
+     IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
```

```
C
      WRITE(6,10) AWS(1)
   10 FORMAT(1H0,*ERROR IN RETRIEVAL FROM DA FILE BETWEEN STMTS 2030 AND
     +   2050; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT5)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT5*,3(O20,4X))
C
      RETURN
      END
C
C
C
      SUBROUTINE EROUT6
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
     +,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
     +IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
C
      WRITE(6,10) AWS(1)
   10 FORMAT(1H0,*ERROR IN RETRIEVAL FROM DA FILE BETWEEN STMTS 2120 AND
     +   2150; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT6)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT6*,3(O20,4X))
C
      RETURN
      END
C
C
C
      SUBROUTINE EROUT7
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
     +,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
     +IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
C
      WRITE(6,10) TWS(1)
   10 FORMAT(1H0,*ERROR IN REPLACEMENT OF DA RECORD BETWEEN STMTS 2120
     +   AND 2150; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT7)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT7*,3(O20,4X))
C
      RETURN
      END
C
```

```
C
      SUBROUTINE EROUT8
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
+     ,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
+     IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
C
      WRITE(6,10)  AWS(1)
   10 FORMAT(1H0,*ERROR IN RETRIEVAL FROM DA FILE BETWEEN STMTS 2170 AND
+     2220; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT8)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT8*,3(O20,4X))
C
      RETURN
      END
C
C
      SUBROUTINE EROUT9
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
+     ,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
+     IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
C
      WRITE(6,10) TWS(1)
   10 FORMAT(1H0,*ERROR IN REPLACEMENT OF DA RECORD BETWEEN STMTS 2170
+     AND 2220; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT9)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT9*,3(O20,4X))
C
      RETURN
      END
C
C
      SUBROUTINE EROUT10
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
+     ,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
+     IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7)
C
      WRITE(6,10) AWS(1)
   10 FORMAT(1H0,*ERROR IN RETRIEVAL FROM DA FILE BETWEEN STMTS 2520 AND
+     2540; KEY IS*,2X,F7.1)
C
      N=SHIFT(TAPE4(5).AND.(.NOT.MASK(26)),-18)
      PRINT 30, N
   30 FORMAT(1H ,*ERROR(EROUT10)*,O3,*PROGRAM TERMINATES*)
C
      WRITE(6,40) TAPE4
   40 FORMAT(1H ,*FIT FROM EROUT10*,3(O20,4X))
C
      RETURN
      END
```

```
C
      SUBROUTINE WASTE(MAX,MIN)
C
C     SUBROUTINE WASTE PRINTS ALL DATES ON WHICH FINAL HARVESTS EITHER
C     EXCEED MAX OR FALL BELOW MIN CANNERY CAPACITY.  IT LISTS EXCESS
C     AND SLACK IN POUNDS
C
C     ALPHABETICAL DICTIONARY
C     IBNCH  LBS OF BUNCHING ON GIVEN HARVEST DAY
C     ISLAK  SLACK (LBS) AT CANNERY ON GIVEN HARVEST DAY
C     IWASTE  SUM OF ALL LBS BUNCHED;  CANNERY EXCESS
C     LOOPA, LOOPB   COUNTERS TO AVOID HEADLINE REPEATS
C     MO1  ACTUAL HARVEST MONTH; CALENDAR DATE INSTEAD OF ARRAY DATE
C
       COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
+     ,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
+     IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7),ISIM,
+     AVGLO(10,30,21),AVGHI(10,30,21),CONV1(10),CONV2(10),IFAC1(10),
+     IFAC2(10),JYSUM(7,31),LIMIT(10),IWASTE,IFHM,IFHD,LHD,LHM,B,ISLAKT
C
        ISLAKT = 0
      IWASTE=0
      LOOPA=0
      LOOPB=0
      DO 200 J=1,7
      DO 200 K=1,31
        IF(JYSUM(J,K).EQ.0) GOTO 200
        IF((JYSUM(J,K).LE.MAX).AND.(JYSUM(J,K).GE.MIN)) GOTO 200
        IF(JYSUM(J,K).GT.MAX)  GOTO 80
C                                   CALC AMT OF SLACK AT CANNERY
        ISLAK = MIN-JYSUM(J,K)
        MO1 = J+2
        IF(LOOPA.GT.0) GOTO 40
        WRITE(6,20)
   20 FORMAT(1H ,20X,*HARVEST DATE*,10X,*LBS BELOW MIN CANNERY CAPACTY*)
C
   40   WRITE(6,60) MO1, K, ISLAK
   60   FORMAT(* *,23X,I2,2X,I2,26X,I6)
        LOOPA = LOOPA + 1
        ISLAKT = ISLAKT + ISLAK
        GOTO 200
C                                   CALC AMT OF BUNCHNG AT CANNERY
   80   IBNCH= JYSUM(J,K)-MAX
        IWASTE = IWASTE + IBNCH
        MO1 = J+2
        IF(LOOPB.GT.0) GOTO 100
        WRITE(6,90)
   90 FORMAT(1H ,20X,*HARVEST DATE*,10X,*LBS ABOVE MAX CANNERY CAPACTY*)
C
  100   WRITE(6,120) MO1,K,IBNCH
  120   FORMAT(* *,23X,I2,2X,I2,26X,I10)
        LOOPB = LOOPB + 1
C
  200   CONTINUE
      RETURN
      END
C
```

```
C
      SUBROUTINE RANGE(IFNUM)
C
C     SUBROUTINE RANGE DETERMINES FIRST AND LAST DAYS OF HARVEST
C     AFTER YIELDS ADJUSTED TO CANNERY CAPACITY
C
C     ALL VARIABLES DEFINED IN MAIN PROGRAM DICTIONARY
C
      COMMON BASE,TMH(7,31),TDDH(7,31),TDDTOH(7,31),TMC(7,31),TDDC(7,31)
     +,TDDTOC(7,31),IPDD(10,30),IPMD(10,30),IHUMN(10),IX,IHARM(10,30),
     +IHARD(10,30),IHDM,IHDD,TAPE4(35),AWS(12),TWS(12),IWS(7),ISIM,
     +AVGLO(10,30,21),AVGHI(10,30,21),CONV1(10),CONV2(10),IFAC1(10),
     +IFAC2(10),JYSUM(7,31),LIMIT(10),IWASTE,IFHM,IFHD,LHD,LHM
C
      CALL OPENM(TAPE4,3LI-O)
      IFHD = 31
      IFHM = 9
      DO 100 I=1,ISIM
        LIM = LIMIT(I)
C                                 DETERMINE 1ST DAY OF HARVEST
        DO 100 J=1,LIM
        DO 100 K=1,IFNUM
        AKEY = FLOAT((I*10000)+(J*100)+K)
        CALL GET(TAPE4,AWS,AKEY,0,0,0,EROUT4)
        IF(((AWS(4).EQ.IFHM).AND.(AWS(3).LT.IFHD)).OR.(AWS(4).LT.IFHM))
     +  IFHM = AWS(4)
        IF(((AWS(4).EQ.IFHM).AND.(AWS(3).LT.IFHD)).OR.(AWS(4).LT.IFHM))
     +  IFHD = AWS(3)
  100   CONTINUE
C
      LHD = 1
      LHM = 3
      DO 200 I=1,ISIM
        LIM = LIMIT(I)
C                                 DETERMINE LAST DAY OF HARVEST
        DO 200 J=1,LIM
        DO 200 K=1,IFNUM
        AKEY = FLOAT((I*10000)+(J*100)+K)
        CALL GET(TAPE4,AWS,AKEY,0,0,0,EROUT5)
        IF(AWS(4).GT.LHM) LHM=AWS(4)
        IF(((AWS(4).EQ.LHM).AND.(AWS(3).GT.LHD)).OR.(LHM.GT.TLHM))LHD=
     +  AWS(3)
        TLHM=LHM
  200   CONTINUE
C
      CALL CLOSEM(TAPE4)
      RETURN
      END
C
```

APPENDIX G

Industry Questionnaire

The following questionnaire was sent to a dozen commercial canners, not all of whom responded. Canners were told that their responses would be kept confidential, with the exception of the Fort Lupton Canning Company which agreed to provide data for simulation development. Respondents were: California Canners and Growers; Baker Canning Company; Fort Lupton Canning Company; Green Giant Company; Libby, McNeill and Libby; and Stokely Van Camp, Inc.

Please Answer All Applicable Questions

What variety (-ies) of peas do you can?

How many weeks per year do you can peas?

What is your daily production capacity at one cannery?

_____pounds raw product   _____cans finished product

How do you determine the number of acres of peas to plant given a fixed cannery capacity?

What is the maximum length of time the raw product can be held after harvest before processing?

What method of maturity testing do you use?

_____tenderometer _____soluble solids

_____A.I.S. _____other (please specify)

____

What is the schedule of maturity testing?

____80% bloom ____other (please specify)

____

Do you use heat unit accumulation to schedule plantings?

Do you use daylength, light intensity and/or soil moisture levels in scheduling plantings?

What type of temperature measurements do you take?

_____air _____soil _____combination

If you use soil temperature, is this strictly pre-emergence?

What method do you use for predicting what percentage of a field will grade "fancy," "standard" etc.?

Do you currently employ a computer program for scheduling planting and harvesting operations?

Please use the rest of this page for any additional comments you may have.