

AD-A096 370

NAVAL POSTGRADUATE SCHOOL MONTEREY CA

F/6 15/5

A STUDY OF MULTI-ECHOLON AND MULTI-LOCATION INVENTORY SYSTEM.(U)

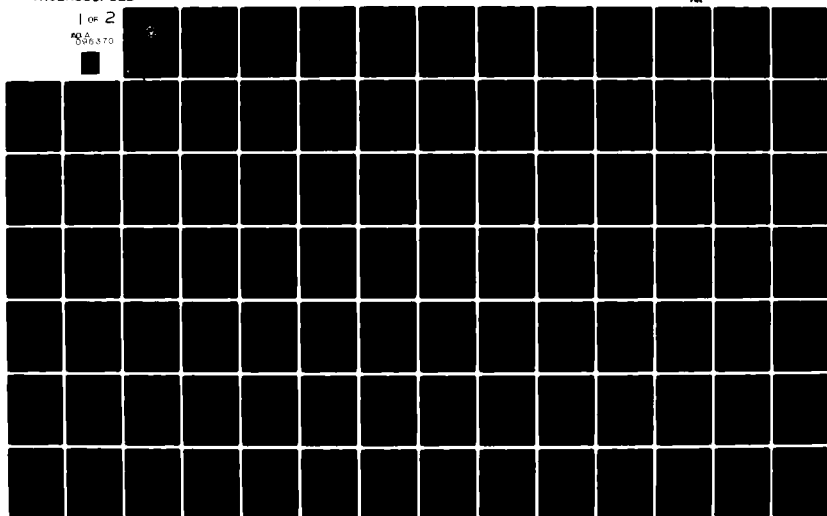
SEP 80 T BUEYUEKKARHAN

UNCLASSIFIED

NI

1 of 2

NO 6
000,370



AD A 096370

NAVAL POSTGRADUATE SCHOOL
Monterey, California



THESIS

DTIC
ELECTE
MAR 16 1981
S A D

A STUDY OF MULTI-ECHELON
AND MULTI-LOCATION INVENTORY SYSTEM

by

Turgut Büyükkarhan

September 1980

Thesis Advisor: F. Russell Richards

Approved for public release; distribution unlimited

DBG FILE COPY

81 3 16 081

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (6)	2. GOVT ACCESSION NO. AD-A096370	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Study of Multi-Echelon and Multi- Location Inventory System.		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1980
6. AUTHOR(s) (10) Turgut/Büyükkarhan		7. PERFORMING ORG. REPORT NUMBER
8. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		9. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		12. REPORT DATE September 1980
(12) 150		13. NUMBER OF PAGES 149 pages
		15. SECURITY CLASS. (of this report) Unclassified
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		18. DECLASSIFICATION/DOWNGRADING SCHEDULE
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Multi-echelon, multi-location inventory system, costs, back orders.		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The reorder point and reorder quantity for a multi-echelon inventory system consisting of two levels were determined through the use of a mathematical model and a computer simulation was used to verify the results. The main echelon supported two lower echelon stock points and reordered using a continuous review inventory policy. The two		

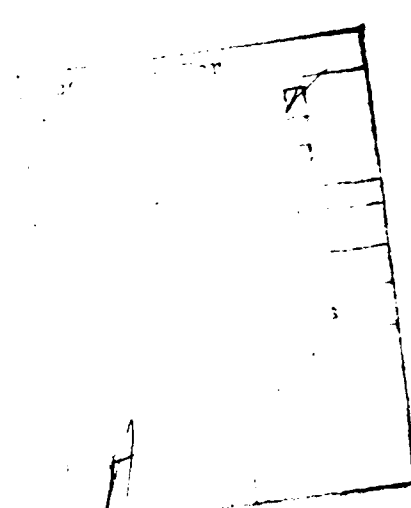
UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

lower echelon stock points operated under periodic review policies.

The measure of effectiveness used was to minimize total system costs subject to a constraint on the maximum number of back orders per year.

The results from the mathematical model were used as input values for the simulation and measures of effectiveness were compared. An alternative procedure was proposed and simulated; and the results of the three products were compared.



Approved for public release; distribution unlimited

A Study of Multi-Echelon
And Multi-Location Inventory System

by

Turgut Büyükkarhan
Lieutenant, Turkish Navy
B.S., Naval Postgraduate School, 1980

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the
NAVAL POSTGRADUATE SCHOOL
September 1980

Author

Büyükkarhan

Approved by:

J.R. Richards

Thesis Advisor

Allen F. Roland

Second Reader

Theodore T. Marshall

Chairman, Department of Operations Research

W. M. Woodin

Dean of Information and Policy Sciences

ABSTRACT

The reorder point and reorder quantity for a multi-echelon inventory system consisting of two levels were determined through the use of a mathematical model and a computer simulation was used to verify the results.

The main echelon supported two lower echelon stock points and reordered using a continuous review inventory policy. The two lower echelon stock points operated under periodic review policies.

The measure of effectiveness used was to minimize total system costs subject to a constraint on the maximum number of back orders per year.

The results from the mathematical model were used as input values for the simulation and measures of effectiveness were compared. An alternative procedure was proposed and simulated; and the results of the three products were compared.

TABLE OF CONTENTS

I.	INTRODUCTION-----	8
II.	A PROBABILISTIC MATHEMATICAL MODEL FOR MULTI- ECHELON INVENTORY SYSTEM-----	10
	A. DESCRIPTION OF MODEL-----	10
	B. OBJECTIVE OF MODEL-----	10
	C. ASSUMPTIONS OF PROBABILISTIC MATHEMATICAL MODEL-----	11
	1. Main System-----	11
	2. System One-----	13
	3. System Two-----	15
	D. ANALYTICAL SOLUTION TO THE MODEL-----	15
	1. Main System-----	16
	2. System One and System Two-----	19
	a. Ordering and Reviewing Costs-----	19
	b. Inventory Carrying Cost-----	19
	c. Stockout Costs-----	20
	3. Objective Function and Constraints-----	22
	4. Optimum Values of Operating Variables-----	23
	E. EXAMPLE PROBLEM-----	26
	1. Additional Assumptions-----	26
	2. Input Values-----	27
	a. Main System-----	28
	b. System One-----	28
	c. System Two-----	28
	3. Solution Procedure and Results-----	29
III.	COMPUTER SIMULATION FOR MULTI-ECHELON, MULTI- LOCATION SINGLE ITEM INVENTORY SYSTEMS-----	36
	A. DESCRIPTION OF THE SIMULATION MODEL-----	36
	1. Main System-----	36
	2. System One and System Two-----	38

B.	HOW TO USE THE PROGRAM-----	38
C.	RUNS-----	39
	1. Starting Conditions-----	39
	2. Results of Runs-----	40
	a. Main System-----	40
	b. System One-----	40
	c. System Two-----	41
	(1) Total Results-----	41
	(a) Main System-----	41
	(b) System One-----	42
	(c) System Two-----	42
	(2) Total Results-----	43
	3. Comparison of Both Starting Conditions-----	43
IV.	COMPARISON OF ANALYTICAL AND COMPUTER SIMULATION RESULTS-----	44
V.	AN ALTERNATIVE SOLUTION-----	49
	A. DESCRIPTION OF ALTERNATIVE PROCEDURE-----	49
	B. COMPUTER SIMULATION RESULTS FOR EARLY WARNING POLICY-----	50
	1. Main System Results-----	51
	2. System One Results-----	51
	3. System Two Results-----	51
	C. COMPARISON OF EARLY WARNING POLICY SIMULATION RESULTS WITH ANALYTICAL AND FIRST SIMULATION RESULTS-----	51
VI.	CONCLUSION-----	53
	APPENDIX A - Flowcharts of First Simulation Model-----	56
	APPENDIX B - Flowcharts of Early Warning Simulation Model-----	68
	APPENDIX C - Simulation Program; Versatec Plotter-----	80
	APPENDIX D - Simulation Program; No Versatec Plotter Output-----	101
	APPENDIX E - Early Warning Simulation Program-----	120

APPENDIX F - Versatec Output of the Program in Appendix C-	138
APPENDIX G - TI-59 Calculator Program for Analytical Solutions for Periodic Review Systems-----	141
APPENDIX H - Variable Definitions for Simulation Programs-	146
BIBLIOGRAPHY-----	148
INITIAL DISTRIBUTION LIST-----	149

I. INTRODUCTION

Despite the fact that all military supply systems and many large civilian corporations have multi-echelon and multi-location inventory systems, few multi-echelon inventory models are currently being used. The policies that have been used are the single echelon and single location inventory policies that attempt to minimize the total variable cost of a single location ignoring total system cost. The difficulties of optimizing a multi-echelon and multi-location inventory system are most likely due to the complexity of demand and the interdependency among the units in different echelons.

A mathematical model for a multi-echelon and multi-location inventory system is developed in Chapter II. The objective of this model is to minimize the total annual variable costs subject to a maximum allowable number of back orders per year for the entire system. This model consists of one first echelon location operating under a continuous review policy and two second echelon systems which operate under periodic review policies. An example is presented to obtain numerical results.

In Chapter III, a computer model is developed to simulate the multi-echelon, multi-location system. This simulation model was used to check the results obtained from

the mathematical model and to demonstrate the interaction between echelons. Graphical plots are generated which show the inventory position of each entity in the model.

In Chapter IV, comparisons are made of the simulation results and the results obtained from the mathematical model. The comparisons suggest some modifications to the operating policy.

The system was simulated again using the modified operating policy and substantial improvements in the effectiveness of the multi-echelon system were observed. These results are described in Chapter V.

Chapter VI summarizes the results of the research and concludes with some suggestions for additional research.

II. A PROBABILISTIC MATHEMATICAL MODEL FOR MULTI-ECHELON INVENTORY SYSTEM

A. DESCRIPTION OF MODEL

In this model there are three systems. One of these systems represents the highest echelon and is called the Main System. The other two, at separate locations, are the lower echelon and are called System One and System Two, respectively. Each system carries its own inventory, and receives random demands. System One and System Two are dependent upon the Main System but independent of each other. System One and System Two can only be resupplied by the Main System. In other words, they cannot order from any other suppliers. The Main System replenishes stocks by placing orders to external suppliers.

B. OBJECTIVE OF MODEL

The objective of this model is to minimize the total expected annual variable costs of three systems subject to a specified expected number of total back orders per year. In fact, this is the same as minimization of total yearly variable costs subject to a specified minimum level of customer satisfaction.

Based on these objectives, decision variables for each system will be calculated to achieve optimum levels for the entire multi-echelon supply system.

C. ASSUMPTIONS OF THE PROBABILISTIC MATHEMATICAL MODEL

1. Main System

It is assumed that the Main System under consideration consists of a single installation which utilizes transaction reporting. This system is governed by a (Q, r) type policy with back orders.

The assumptions in addition to the continuous review assumption are:

a. The cost of operating the information processing system is independent of Q (reorder quantity) and r (reorder point).

b. The unit cost C of the item is a constant independent of Q .

c. The back-order cost is constant (Π), per unit back ordered regardless of the length of time the back-orders exist.

d. There is never more than a single order outstanding. This assumption implies that when the reorder point is reached, there are no orders outstanding; therefore, the inventory position is equal to the net inventory. Thus, the reorder point will be the same regardless of whether it is based on the inventory position or net inventory.

e. Procurement lead times are independent and identically distributed (i.i.d) random variables with a gamma distribution.

f. All variables are treated as continuous.

g. The demands are Poisson distributed with the mean number of demands per year a constant λ_m .

h. The reorder point, r , which is based on the inventory position is positive.

With a back orders constraint, it is infeasible to wait until back orders exist before placing an order. Because of this and assumption (d) there will be no back orders outstanding at the reorder point. As was discussed in assumption (d), at the reorder point, the inventory position is equal to the on-hand inventory.

For this model, any one of the three inventory levels on hand, net, or inventory position can be used to define the reorder point; and the reorder point has the same value for any one of them. It should also be noted that to use the on-hand level it must be assumed that after an order arrives, it is sufficient to fill all back orders and raise the on-hand inventory level above the reorder point. If this ever failed to happen, the reorder point would never be reached again and the system would continue to accumulate back orders indefinitely.

When the reorder point is thought of in terms of the inventory position of the system, then assumption (d) guarantees that the on-hand inventory will always exceed the reorder point when an order arrives; otherwise, it would not be possible to have only a single order outstanding.

2. System One

This system, which is at the lower echelon, is governed by a periodic review policy. The operating doctrine is the most widely used type of periodic review which is the order up to R policy. All demands which occur when the system is out of stock are back ordered.

For this periodic review system the time between reviews will be denoted by T , and at each review time a sufficient quantity is ordered to bring the inventory position of the system up to a level, R , regardless of the amount of on-hand inventory. This policy dictates that at review times even if the inventory position is $R-1$, only one item must be ordered to bring the inventory position up to R , ignoring the high cost of placing the order. Despite its appearance of being illogical, this assumption simplifies the formulation of the system and is very unlikely to occur on high demand items. When an item experiences zero demand in a review period, there is no need for order because the inventory position is already at R .

The other assumptions are:

- a. The cost, J , of making a review is independent of the variables R and T .
- b. The unit cost, C , of the item is constant and independent of the quantity ordered.

c. Back orders are incurred only in very small quantities. This implies that when an order arrives, it is almost always sufficient to meet any outstanding back orders.

d. The backorder cost is constant, Π , per unit back ordered regardless of the length of time the back order exists.

e. Procurement lead times are i.i.d random variables with a gamma distribution.

f. Orders are received in the same sequence in which they were placed. It should be noted that for (Q, r) models, the two assumptions that orders were received in the sequence placed and that lead times are i.i.d random variables could not both hold rigorously, since there exists a positive probability that two successive orders could be separated by an arbitrarily short time interval. In this model, orders can never be more closely spaced than by an interval of length T . If T is large enough, it is possible, provided that there is a sufficiently small range of variation in the lead time, that both assumptions hold simultaneously.

g. The demands are Poisson distributed with mean, λ .

h. All variables are treated as continuous.

3. System Two

This system is identical to, but independent of, System One.

D. ANALYTICAL SOLUTION TO THE MODEL

As was mentioned previously, the objective of the model is to minimize total system costs subject to a constraint on the number of back orders per year.

The determination of various annual costs of each system may be made independently and placed into a common cost formula. Instead of doing this independently, it is better to find a main system cost expression, System One cost expression and System Two cost expression, and then to add them to each other to determine the total cost formulation and then use this formulation as the objective function of total system. The systems are actually tied together through the constraint on the number of back orders.

For the following notation, the subscript m will indicate that the variables being subscripted belong to the Main System where 1 and 2 indicate System One and System Two, respectively.

The costs that are of interest in this model for each system are the cost of placing an order, the cost of carrying inventory and back order costs.

1. Main System

As was discussed previously, the Main System has a transactions reporting policy or (Q, r) model.

In a continuous review system, a period is defined as the length of time between the receipt of two successive procurements. This time period is a random variable. Because the procurement lead times are random variables, the number of demands for a fixed time are random variables; and the number of items demanded per demand are also random variables. The review period is also a random variable. In the following material, we itemize the costs.

a. Procurement Cost

λ_m = Expected number of demands per year

Q_m = Order quantity

$\frac{\lambda_m}{Q_m}$ = Average number of procurements per year

A_m = Procurement cost per cycle

$\frac{\lambda_m}{Q_m} A_m$ = Procurement cost per year

b. Inventory Carrying Costs

Because of randomness, it is possible to accumulate a large number of back orders at the end of a cycle. To prevent this from occurring, it is advisable to provide a safety stock to buffer the system from excessive numbers

of back orders. The safety stock is the expected amount of stock on hand when an order arrives. The actual amount of stock on hand when a shipment arrives is clearly random.

S_m = Mean value of on-hand stock when an order arrives

After an order arrives, the expected on-hand inventory increases to $Q+S$ and is reduced to a value of S on the average just before the next order arrives. Therefore, the average on-hand inventory per cycle is:

$$\frac{(Q_m + S_m)}{2} + \frac{S_m}{2} = \frac{Q}{2} + S_m$$

To write S_m in terms of the reorder point, r_m , let us first assume that the lead time τ_m is fixed. Let

$\xi_{\tau}(x; r_m) = r_m - x$ be the net inventory at the time an order arrives,

where x

is the number of units demanded in lead time τ_m . Then,

$$S_m = E_{\tau_m} [\text{Net inventory}] = \int_0^{\infty} \xi_{\tau_m}(x; r_m) f(x; \tau_m) dx$$

where $f(x; \tau_m)$ = Density function of demand in time τ_m

$$S_m = \int_0^{\infty} (r_m - x) f(x; \tau_m) dx = r_m \int_0^{\infty} f(x; \tau_m) dx - \int_0^{\infty} x f(x; \tau_m) dx$$

$S_m = r_m - \mu_m$ where μ_m = Expected lead time demand.

$$\text{Then } \frac{Q_m}{2} + S_m = \frac{Q_m}{2} + r_m - \mu_m$$

I = Inventory carrying charge

C = Cost of an item

$$\text{Total holding cost/year} = IC (r_m - \mu_m + \frac{Q_m}{2})$$

c. Stockout Costs

$$\text{Let us define } \eta_{\tau_m}(x; r_m) = \begin{cases} 0 & \text{if } x - r_m < 0 \\ x - r_m & \text{if } x - r_m \geq 0 \end{cases}$$

where $\eta_{\tau_m}(x; r_m)$ is the number of back orders per cycle.

If $\bar{\eta}_m(r_m)$ = expected number of back orders per cycle, then

$$\bar{\eta}_m(r_m) = \int_0^{\infty} \eta_{\tau_m}(x; r_m) h(x) dx = \int_0^{\infty} (x - r_m) h(x) dx$$

$$= \int_{r_m}^{\infty} xh(x) dx - r_m H(r_m) \text{ where } H(r_m) = P[X > r_m]$$

and $h(x)$ = marginal distribution of leadtime demand.

Therefore, the expected number of back orders/year =

$$\frac{\lambda_m}{Q_m} [\int_{r_m}^{\infty} xh(x) dx - r_m H(r_m)]$$

and the expected cost of back orders/year =

$$\frac{\Pi_m \lambda_m}{Q_m} [\int_{r_m}^{\infty} xh(x) dx - r_m H(r_m)]$$

All the terms in the average annual variable cost K_m have now been found:

$$K_m = \frac{\lambda_m}{Q_m} A_m + IC \left[\frac{Q_m}{2} + r_m - \mu_m \right] + \frac{\pi \lambda_m}{Q_m} \left[r_m \int_0^\infty x h(x) dx - r_m H(r_m) \right]$$

2. System One and System Two

It was stated in the assumptions that System One and also System Two both follow a periodic review policy with an order up to R stockage policy. Since System One and System Two are identical to each other; the equations will be derived only for System One.

For convenience, a period is assumed to be the time between the receipt of two successive orders rather than between the placement of two successive orders. Costs are described as follows:

a. Ordering and Reviewing Costs

J = Reviewing cost/cycle

A = Ordering cost/cycle

$$\text{Ordering and reviewing cost/year} = \frac{J_1 + A_1}{T_1}$$

where T_1 is the period length defined in units of years.

b. Inventory Carrying Cost

The expected net inventory just prior to the arrival of an order is $R_1 - \mu_1 - \lambda_1 T_1$, where μ_1 = mean demand during lead time.

The mean rate of demand remains constant over time and the expected demand per period must be the expected

amount ordered, i.e., $\lambda_1 T_1$. If the expected net inventory immediately after the arrival of a procurement is $R_1 - \mu_1$, it is therefore $R_1 - \mu_1 - \lambda_1 T_1$ just prior to the arrival of a procurement.

The expected unit years of storage incurred per period is

$$T_1 \left[\frac{1}{2} (R_1 - \mu_1) + \frac{1}{2} (R_1 - \mu_1 - \lambda_1 T_1) \right] = T_1 \left[R_1 - \mu_1 - \frac{\lambda_1 T_1}{2} \right]$$

and average inventory carrying cost/year = IC $\left[R_1 - \mu_1 - \frac{\lambda_1 T_1}{2} \right]$.

c. Stockout Costs

First we assume the case where the procurement lead time is constant τ . An order placed at time t will arrive in the system at time $t + \tau$, and the next procurement will arrive in the system at time $t + \tau + T_1$. After the order is placed at time t , the inventory position of the system is R_1 . It is necessary to compute the expected number of back orders occurring between $t + \tau$ and $t + \tau + T_1$. A back order will occur in this period under assumption c if and only if the demand in the time period $\tau + T_1$ exceeds R_1 . Assumption c also assures that after the arrival of the order placed at time t , there will be no remaining back orders, and therefore they must all occur between times $t + \tau$ and $t + \tau + T_1$. Consequently the expected number of back orders incurred per period is

$$R_1 \int_{R_1}^{\infty} (x - R_1) f(x; \tau + T_1) dx \text{ where}$$

$f(x; \tau + T_1)$ = Demand distribution during time $\tau + T_1$.

When lead time is random with density $g(\tau_1)$ with $\tau_{\min}$ and $\tau_{\max}$ being lower and upper limits respectively and τ_1 and τ_2 , the lead times for the orders placed at times t and $t + T_1$, respectively, the expected number of back orders incurred per period is:

$$\begin{aligned} & \int_{\tau_{\min}}^{\tau_{\max}} \int_{\tau_{\min}}^{\tau_{\max}} R_1 \int_{R_1}^{\infty} (x - R_1) f(x; \tau_2 + T_1) g(\tau_2) g(\tau_1) dx d\tau_2 d\tau_1 \\ &= R_1 \int_{R_1}^{\infty} (x - R_1) \hat{h}(x; T_1) dx \end{aligned}$$

where $\hat{h}(x; T_1) = \int_{\tau_{\min}}^{\tau_{\max}} f(x; \tau_2 + T_1) g(\tau_2) d\tau_2$ which is the

demand distribution during time $\tau_2 + T_1$ when lead time is a random variable with density function $g(\tau_2)$. The average number of back orders incurred per year is;

$$E_1(R_1, T_1) = \frac{1}{T_1} R_1 \int_{R_1}^{\infty} (x - R_1) \hat{h}(x; T_1) dx \text{ and the average}$$

back order cost per year equals to $\Pi_1 E_1(R_1, T_1)$ and

$\bar{n}_1(r) = T_1 E_1(R_1, T_2)$ is the expected number of back orders per period.

Finally, the annual variable cost of System One is:

$$K_1 = \frac{L_1}{T_1} + IC \left[R_1 - \mu_1 - \frac{\lambda_1 T_1}{2} \right] + \pi_1 E_1 (R_1, T_1) \text{ and}$$

likewise the annual variable cost of System Two is:

$$K_2 = \frac{L_2}{T_2} + IC \left[R_2 - \mu_2 - \frac{\lambda_2 T_2}{2} \right] + \pi_2 E_2 (R_2, T_2)$$

where $L_i = J_i + A_i$.

3. Objective Function and Constraints

The objective function of the model consists of the total annual variable costs of each system. Therefore;

$K = K_m + K_1 + K_2$ which is equal to the minimization of;

$$\begin{aligned} K = & \frac{\lambda_m}{Q_m} A_m + IC \left[\frac{Q_m}{2} + r_m - \mu_m \right] + \frac{\pi_m \lambda_m}{Q_m} \bar{n}_m(r) + \frac{L_1}{T_1} \\ & + IC \left[R_1 - \mu_1 - \frac{\lambda_1 T_1}{2} \right] + \frac{\pi_1}{T_1} \bar{n}_1(r) + \frac{L_2}{T_2} + IC \left[R_2 - \mu_2 - \frac{\lambda_2 T_2}{2} \right] \\ & + \frac{\pi_2}{T_2} \bar{n}_2(r) \end{aligned}$$

subject to $\frac{\lambda_m}{Q_m} \bar{n}_m(r) + \frac{1}{T_1} \bar{n}_1(r) + \frac{1}{T_2} \bar{n}_2(r) \leq b$

where b is the specified total maximum number of back orders per year for the entire system.

The problem at hand is to calculate the optimum values of $Q_m, r_m, R_1, T_1, R_2, T_2$. Since the objective function and the constraint are non-linear functions of the

decision variables, we solve the problem using the Lagrange multiplier approach.

4. Optimum Values of Operating Variables

After including the Lagrange multiplier in the formulation, the new objective function becomes:

Minimize

$$\begin{aligned}
 L = & \frac{\lambda_m}{Q_m} A_m + IC\left[\frac{Q_m}{2} + r_m - \mu_m\right] + \frac{\Pi_m \lambda_m}{Q_m} \bar{n}_m(r) + \frac{L_1}{T_1} \\
 & + IC\left[R_1 - \mu_1 - \frac{\lambda_1 T_1}{2}\right] + \frac{\Pi_1}{T_1} \bar{n}_1(r) + \frac{L_2}{T_2} + IC\left[R_2 - \mu_2 - \frac{\lambda_2 T_2}{2}\right] \\
 & + \frac{\Pi_2}{T_2} \bar{n}_2(r) - \theta \left[\left(\frac{\lambda_m}{Q_m} \bar{n}_m(r) + \frac{1}{T_1} \bar{n}_1(r) + \frac{1}{T_2} \bar{n}_2(r) \right) - b \right]
 \end{aligned}$$

where θ is the Lagrange multiplier. It is through this Lagrange multiplier that the three systems are linked together mathematically.

The optimum values of the unknown variables can be found by taking the derivatives of the objective function with respect to Q_m , R_m , R_1 , T_1 , R_2 , T_2 , θ ; equating them to zero; solving the equations simultaneously; and ensuring the Kuhn-Tucker conditions are satisfied.

The derivatives were taken with respect to Q_m , r_m , R_1 , R_2 , θ . A different procedure was utilized to find T_1 and T_2 .

The derivatives:

$$L_{Q_m} = \frac{\partial L}{\partial Q_m} = - \frac{\lambda_m A_m}{Q_m^2} + \frac{IC}{2} - \frac{\lambda_m}{Q_m^2} \bar{n}_m(r) (\Pi_m - \theta) = 0$$

$$L_{r_m} = \frac{\partial L}{\partial r_m} = IC - \frac{\lambda_m}{Q_m} \hat{H}_m(r) (\Pi_m - \theta) = 0$$

$$L_{R_1} = \frac{L}{R_1} = IC - \frac{1}{T_1} H_1(R_1, T_1) (\Pi_1 - \theta) = 0$$

$$L_{R_2} = \frac{\partial L}{\partial R_2} = IC - \frac{1}{T_2} H_2(R_2, T_2) (\Pi_2 - \theta) = 0$$

$$L_\theta = \frac{\partial L}{\partial \theta} = \left[\left(\frac{\lambda_m}{Q_m} \bar{n}_m(r) + \frac{1}{T_1} \bar{n}_1(r) + \frac{1}{T_2} \bar{n}_2(r) \right) - b \right] = 0$$

The Kuhn-Tucker conditions:

$$L_{Q_m} \geq 0 \quad Q_m \cdot L_{Q_m} = 0 \quad Q_m \geq 0 \quad g \leq 0$$

$$L_{r_m} \geq 0 \quad r_m \cdot L_{r_m} = 0 \quad r_m \geq 0 \quad \theta \cdot g = 0$$

$$L_{R_1} \geq 0 \quad R_1 \cdot L_{R_1} = 0 \quad R_1 \geq 0 \quad \theta \leq 0$$

$$L_{R_2} \geq 0 \quad R_2 \cdot L_{R_2} = 0 \quad R_2 \geq 0$$

$$\text{where } g = \frac{\lambda_m}{Q_m} \bar{n}_m(r) + \frac{1}{T_1} \bar{n}_1(r) + \frac{1}{T_2} \bar{n}_2(r) - b$$

Therefore the following Kuhn-Tucker conditions must be satisfied:

$$-\frac{\lambda_m A_m}{Q_m^2} + \frac{IC}{2} - \frac{\lambda_m}{Q_m} \bar{\eta}_m(r) (\Pi_m - \theta) \geq 0$$

$$IC - \frac{\lambda_m}{Q_m} \hat{H}_m(r) (\Pi_m - \theta) \geq 0$$

$$IC - \frac{1}{T_1} \hat{H}_1(R_1, T_1) (\Pi_1 - \theta) \geq 0$$

$$IC - \frac{1}{T_2} \hat{H}_2(R_2, T_2) (\Pi_2 - \theta) \geq 0$$

$$-\frac{\lambda_m A_m}{Q_m} + \frac{IC Q_m}{2} - \frac{\lambda_m}{Q_m} \bar{\eta}_m(r) (\Pi_m - \theta) = 0$$

$$IC r_m - \frac{\lambda_m r_m}{Q_m} \hat{H}_m(r) (\Pi_m - \theta) = 0$$

$$IC R_1 - \frac{R_1}{T_1} \hat{H}_1(R_1, T_1) (\Pi_1 - \theta) = 0$$

$$IC R_2 - \frac{R_2}{T_2} \hat{H}_2(R_2, T_2) (\Pi_2 - \theta) = 0$$

$$Q_m \geq 0$$

$$r_m \geq 0$$

$$R_1 \geq 0$$

$$R_2 \geq 0$$

$$\frac{\lambda_m}{Q_m} \bar{\eta}_m(r) + \frac{1}{T_1} \bar{\eta}_1(r) + \frac{1}{T_2} \bar{\eta}_2(r) - b \leq 0$$

$$\theta \left[\frac{\lambda_m}{Q_m} \bar{\eta}_m(r) + \frac{1}{T_1} \bar{\eta}_1(r) + \frac{1}{T_2} \bar{\eta}_2(r) - b \right] = 0$$

$$\theta \leq 0$$

From these equations and conditions the optimum values of the unknown variables can be simplified to the following form and the exact values can be found by solving these equations simultaneously:

$$\hat{H}_m(r) = \frac{IC Q_m}{(\pi_m - \theta) \lambda_m}$$

$$Q_m = \frac{\sqrt{2\lambda_m [A_m + \pi_m (\pi_m - \theta)]}}{IC}$$

$$\hat{H}_1(R_1, T_1) = \frac{IC T_1}{(\pi_1 - \theta)}$$

$$\hat{H}_2(R_2, T_2) = \frac{IC T_2}{(\pi_2 - \theta)}$$

The procedure to find T_1 and T_2 and subsequently R_1 and R_2 is through iteration and trial and error. This is not unreasonable. In realistic cases other considerations not modelled here usually dictate the length of a period. In most cases the systems control a large number of different items and the same period length is used for each item. Thus, the period length is usually some convenient calendar or financial period, such as a month or a quarter. The procedure we use is shown in the following example.

E. EXAMPLE PROBLEM

1. Additional Assumptions

For this example it is assumed that each system has Poisson arrivals independent of each other. The customers

demand only one item at a time. System One and System Two are resupplied only from the main system and the main system is resupplied from outside suppliers. To be more realistic it is also assumed that the procurement lead times have a gamma distribution with different mean and variances.

The λ 's will be assumed daily demand or arrival rate per day rather than yearly values.

2. Input Values

Before providing the input values, some clarifications must be provided regarding the demand at the main system. When it is stated that $\lambda_m = 4/\text{day}$, $\lambda_1 = 3/\text{day}$ and $\lambda_2 = 1/\text{day}$. These represent the mean number of demands which arrive each day directly at the respective systems. However, since all demands at Systems One and Two eventually must filter up to the main system, the cumulative demand at the main system has an expected value of $\lambda_1 + \lambda_2 + \lambda_m = 8$. The main system supports not only Systems One and Two, but also has its own customer demands. The model assumes that the demand at the main system is the superposition of the direct demands at the main system and Systems One and Two. However, since the actual demands placed on the main system in the lower echelon are batched, the demand variability is much greater than would be expected by the superposition process. We will evaluate the seriousness of our assumption about the demand process at the main system with the simulation model described in the next chapter.

In this example the input values for each system are;

a. Main System

$\lambda_m = 8/\text{day}$ (considering the individual customer arrivals and the remainder being Systems One and Two average demands)

$C = \$50/\text{item}$

$I = 0.23$

$\Pi_m = \$5$

$A_m = \$500$

b. System One

$\lambda_1 = 3/\text{day}$

$C = \$50/\text{item}$

$I = 0.23$

$\Pi_1 = \$5$

$A_1 = \$100$

$J_1 = \$100$

c. System Two

$\lambda_2 = 1/\text{day}$

$C = \$50/\text{item}$

$I = 0.23$

$\Pi_2 = \$5$

$A_2 = \$100$

$J_2 = \$100$

The total system is expected not to exceed
 $b = 175$ total number of back orders at the end of a year.

3. Solution Procedure and Results

It was assumed that for each system, procurement lead times are gamma distributed random variables.

The gamma distribution is defined as:

$$g(t) = \frac{1}{\Gamma(\alpha) \beta^\alpha} t^{\alpha-1} e^{-\frac{t}{\beta}} \quad \text{with}$$

$$\mu = \alpha \cdot \beta \quad \text{and}$$

$$\text{Var} = \alpha \cdot \beta^2$$

We need the lead-time demand distribution for the case in which lead times are gamma distributed and the process generating demands is Poisson. This is derived below.

The Poisson distribution is

$$f(x) = \frac{(\lambda t)^x e^{-\lambda t}}{x!} \quad \text{and the demand distribution}$$

during lead time is:

$$\begin{aligned} f(x; \tau) &= \int_0^\infty f(x) g(t) dt \\ &= \int_0^\infty \frac{(\lambda t)^x e^{-\lambda t}}{x!} \frac{t^{\alpha-1} e^{-t/\beta}}{\Gamma(\alpha) \beta^\alpha} dt \\ &= \frac{\lambda^x}{\Gamma(\alpha) \beta^\alpha x!} \int_0^\infty t^{\alpha-1+x} e^{-(\frac{1}{\beta} + \lambda)t} dt \end{aligned}$$

$$= \frac{\lambda^x}{\Gamma(\alpha) \beta^\alpha x!} \frac{\Gamma(\alpha+x)}{(\frac{1}{\beta} + \lambda)^{\alpha+x}} \int_0^\infty \frac{(\frac{1}{\beta} + \lambda)^{\alpha+x} t^{\alpha-1+x} e^{-(\frac{1}{\beta} + \lambda)t}}{\Gamma(\alpha+x)} dt$$

$$\int_0^\infty \frac{(\frac{1}{\beta} + \lambda)^{\alpha+x} t^{\alpha-1+x} e^{-(\frac{1}{\beta} + \lambda)t}}{\Gamma(\alpha+x)} dt = 1 \quad \text{so}$$

$$f(x; \tau) = \frac{\lambda^x}{\Gamma(\alpha) \beta^\alpha x!} \frac{\Gamma(\alpha+x)}{(\frac{1}{\beta} + \lambda)^{\alpha+x}}$$

$$= \frac{\Gamma(\alpha+x)}{\Gamma(\alpha) x!} \left[\frac{\lambda}{(\frac{1}{\beta} + \lambda)} \right]^x \left[\frac{\frac{1}{\beta}}{\frac{1}{\beta} + \lambda} \right]^\alpha$$

$$= \binom{\alpha+x-1}{x} \left[\frac{\lambda}{\frac{1}{\beta} + \lambda} \right]^x \left[\frac{\frac{1}{\beta}}{\frac{1}{\beta} + \lambda} \right]^\alpha$$

$$= \binom{\alpha+x-1}{x} (1-\rho)^x \rho^\alpha$$

$$\mu = \frac{\alpha(1-\rho)}{\rho} = \frac{\alpha(1 - \frac{1/\beta}{1/\beta + \lambda})}{\frac{1/\beta}{1/\beta + \lambda}} = \lambda \alpha \beta$$

$$\text{Var} = \frac{\alpha(1-\rho)}{\rho^2} = \frac{\alpha(1 - \frac{1/\beta}{1/\beta + \lambda})}{\frac{1/\beta}{1/\beta + \lambda}^2} = \lambda \alpha \beta (1 + \lambda \beta)$$

In fact this is a Negative Binomial distribution

with

$$\rho = \frac{1/\beta}{1/\beta + \lambda}.$$

The parameters α and β for each system are:

$$\begin{array}{lll} \alpha_m = 12.8 & \alpha_1 = 5.43 & \alpha_2 = 2.5 \\ \beta_m = 3.125 & \beta_1 = 4.6 & \beta_2 = 6. \end{array}$$

The main system has a Negative Binomial distributed lead time demand with parameters:

$$\begin{aligned} \mu_m &= 8 \cdot (12.8) (3.125) = 320 \\ \text{var}_m &= 8 \cdot (12.8) (3.125) [1 + 8 \cdot (3.125)] = 8320 \end{aligned}$$

System One has the parameters:

$$\begin{aligned} \mu_1 &= 3 \cdot (5.43) \cdot (4.6) = 74.934 \\ \text{var}_1 &= 3 \cdot (5.43) (4.6) [1 + 3(4.6)] = 1109.0232 \end{aligned}$$

System Two has the parameters:

$$\begin{aligned} \mu_2 &= 1 \cdot (2.5) \cdot 6 = 15 \\ \text{var}_2 &= 1 \cdot (2.5) \cdot 6 [1 + 1(6)] = 105 \end{aligned}$$

Because the negative binomial is computationally intractable, we use the normal approximation for the calculation of lead time demand probabilities. The normal distributions are assumed to have the same mean and variance as the negative binomial distributions they replace.

To solve this example problem, it is necessary to make an initial estimate for the Lagrange multiplier θ

which must be less than or equal to zero. After the initial estimate if the total number of back orders per year exceeds b , then the absolute value of θ should be increased gradually until the number of back orders converges to b .

For this problem $\theta = -1.5$ works very well. For the main system the formulas are:

$$Q_m = \sqrt{2\lambda_m [A_m + \bar{h}_m(r) (\Pi_m - \theta)] / IC}$$

$$H_m(r) = \frac{Q_m IC}{(\Pi_m - \theta) \lambda_m} = P [X \geq r_m]$$

Q_m is calculated to be 550 and

$$H_m(r) = P [X \geq r_m] = \frac{550 (0.23) \cdot 50}{(5+1.5) 2920} = 0.3332455216$$

From the inverse standard normal distribution TI-59 calculator program (Appendix G), the reorder level at the main system is found to be:

$$r_m = \sigma_m \cdot (0.4305333395) + \mu_m = 359.2706827$$

$$\bar{h}_m(r) = (\mu_m - r_m) H_m(r) + \sigma_m \phi\left(\frac{r_m - \mu_m}{\sigma_m}\right)$$

where ϕ is the functional value of standard normal distribution at 0.4305333395

$$\bar{h}_m(r) = 20.08139334. \text{ Using this value in the } Q_m$$

formula yields

$$Q_m = \frac{\sqrt{2.2920 [500 + 20.08139334 (5 + 1.5)]}}{(0.23) \cdot 50} = 568.84$$

which is not equal to the first estimated value. If Q_m is taken 569 then:

$$H_m(r) = \frac{569 (0.23) \cdot 50}{(5 + 1.5) 2920} = 0.3447576396$$

$$r_m = \sigma_m (0.399071201) + \mu_m = 356.4008941$$

$$\bar{n}_m(r) = 21.05458646$$

Using this value in the Q_m formula yields

$$Q_m = \frac{\sqrt{2.2920 [500 + 21.05438646 (5 + 1.5)]}}{(0.23) \cdot 50} = 568.69$$

which is very close to the initial Q_m value. A summary of the results for the main system follows:

$$Q_m = 569$$

$$S_m = 36.400891$$

$$\bar{n}_m(r) = 21.05438646/\text{period}$$

$$\bar{n}_m(r) = 108.047115/\text{year}$$

$$K_m = \$6796.50/\text{year}$$

To solve for the optimum values for Systems One and Two a different procedure is followed. The total annual cost of Systems One and Two is a convex function of the period length T . For different values of T there are different total annual

cost values. The minimum of these values is the optimum total annual variable cost and the corresponding T and R values are the optimum operating values. A TI-59 program was written to perform the line search for the best value of T. The program is found in Appendix G. User information and the features of the program are also in the same appendix. This program evaluates the R value, safety stock, back orders per period, back-orders per year, annual reviewing and ordering cost, annual holding cost, annual back order cost, and finally, the total annual cost.

Using this program the optimum values for Systems One and Two are:

$$R_1 = 306.809389$$

$$T_1 = 2.39 \text{ months (72.6958 days)}$$

$$S_1 = 13.78788901$$

$$\bar{n}_1(r) = 8.670289086/\text{period}$$

$$\bar{n}_1(r) = 43.53283223$$

$$K_1 = \$2634.41/\text{year}$$

$$R_2 = 134.9807003$$

$$T_2 = 4.07 \text{ months (123.7958333 days)}$$

$$S_2 = 0$$

$$\bar{n}_2(2) = 8.13480099/\text{period}$$

$$\bar{n}_2(r) = 23.98467122/\text{year}$$

$$K_2 = \$1377.56/\text{year}$$

The total back orders per year are:

$$\begin{aligned}\bar{n}_m(r) + \bar{n}_1(r) + n_2(r) &= 108.047 + 43.532 + 23.984 \\ &= 175.56\end{aligned}$$

which is the same as $b = 175.56$.

The total annual variable cost is:

$$K = K_m + K_1 + K_2 = 6796.50 + 2634.41 + 1377.56$$

$$K = \$10808.47.$$

III. COMPUTER SIMULATION FOR MULTI-ECHELON MULTI-LOCATION SINGLE ITEM INVENTORY SYSTEM

To check the analytical results, a computer simulation was written which uses the same operating assumptions and input parameters that were made for the analytical model. This model, however, simulates the real world more accurately since some of the simplifying assumptions required to obtain analytical results were not necessary in the simulation. Also, the demands placed at the main system from the lower echelon systems were batched as in the real world.

A. DESCRIPTION OF THE SIMULATION MODEL

As in the analytical case, there are three systems in the simulation model; the main system, System One and System Two. The flow charts of this program are in Appendix A.

1. Main System

This system uses a continuous review policy. When the stock level reaches the reorder point, it orders the amount Q . The decision variable for reordering is the inventory position. When an order is placed, the inventory position increases by an amount of Q . If the new inventory position is less than the reorder point, the system places an additional order for another amount of size Q . Then the

inventory position increases with one more Q . This continues until the inventory position exceeds r . The order policy is thus (nQ, r) . N is the smallest integer that will make the inventory position higher than the reorder point.

There are independent customer arrivals to the main system and also group demands that are placed by System One and System Two when those systems replenished their stocks at the end of their periods.

The number of units demanded per requisition is one. The number of units demanded for resupply to the lower echelon systems is random depending on the demands at the lower echelon systems and the parameters R_1, R_2 . The program is capable of allowing geometrically distributed quantities demanded per requisition. Back orders at the main system are satisfied by filling the back orders to individual customers first followed by filling any back orders due to System One and System Two on a first-come, first-served basis.

When a demand occurs from the lower echelon systems for which there is not sufficient on-hand inventory at the main system, the maximum amount is filled and the rest is put into the back order queue. As soon as a shipment arrives at the main system, the back orders are filled. The times between the customer arrivals are independent of each other and exponentially distributed. The lead times are also independent of each other and gamma distributed.

2. System One and System Two

Both systems operate identically. The only difference might be in the values of the system parameters. They have the same periodic review policy which is: order up to R at each review time. The program is also capable of making a decision at each review time regarding the placement of an order if the inventory position is less than or equal to a threshold value, is an (r, R) policy. If r is taken to be $R-1$, then the system orders up to R at each review time even if there is only one demand in a period.

The times between customer arrivals are independent of each other and exponentially distributed. The number of units demanded per requisition is one. The lead times are independent of each other and gamma distributed.

B. HOW TO USE THE PROGRAM

Two simulations are given in Appendix C and Appendix D. The programs differ primarily in the type of output that is generated. The program given in Appendix C produces a Versatec plot output showing the inventory position for the main system and the lower echelon systems. A simulated period of up to 4 or 5 years can be run with a Class K job.

On the Versatec output for each system, there are two plots: one plot has a small triangle at each point representing the inventory position. The second plot represents the net inventory. The first figure shows the main system

values, the second figure shows the System One values and the third figure shows System Two values all plotted versus time.

The second program can be used to simulate operations over arbitrarily long periods of time. There are no dimension restrictions. This program gives the net inventory at the end of each period for all the systems; total demand in the periods of both System One and System Two; demand during a lead time for the main system; average on-hand inventory; and average number of items short per day for each system.

The input required by the programs is described by the variable definition list given in Appendix H. The programs use the IMSL subroutines for random number generation.

C. RUNS

1. Starting Conditions

There were two different starting conditions entered for the main system. In the first case the inventory position and net inventory were set initially to equal the order quantity. In the second case, the inventory position and the net inventory were set equal to the reorder point plus the order quantity. For Systems One and Two the inventory positions and net inventories were set initially at R_i .

2. Results of Runs

In all runs the length of time was 10 years. The average yearly results were obtained by dividing the results for the 10 year period by 10.

Using the policy parameters determined by the mathematical model and the first set of starting conditions the following simulated results were obtained.

a. Main System

The number of orders in 10 years = 51.

The number of back orders in 10 years = 3026.

The average on-hand inventory over 10 years =
313.026.

The average safety stock when an order arrives =
56.54 units.

The average annual costs are:

Ordering costs:	\$2,550.00
Holding costs:	3,599.80
Back Order Costs:	<u>1,513.00</u>
Total Cost	\$7,662.80

b. System One

Number of back orders in 10 years = 587.

The average on-hand inventory over 10 years =
114.482 units.

The average annual costs are:

Ordering and Reviewing Costs:	\$1,004.18
Holding Costs:	1,316.54
Back Orders Costs:	<u>293.50</u>
Total Costs	\$2,614.22

c. System Two

Number of back orders in 10 years = 287.

The average on-hand inventory over 10 years =
57.618 units.

The average annual costs are:

Ordering and Reviewing Costs:	\$ 589.68
Holding Costs:	662.61
Back Orders Cost:	<u>143.50</u>
Total Costs	\$1,395.79

(1) Total results

The total number of back orders for the
whole system per year is $302.6 + 58.7 + 28.7 = 390$.

The total annual variable cost for the
whole system is $\$7,662.80 + \$2,614.22 + \$1,395.79 = \$11,672.81$.

The model was run again with the second set
of starting conditions and the same set of policy parameter
values. The simulation results are summarized below:

(a) Main System

The number of orders in 10 years = 50.

The number of back orders in 10 years =
1665.

The average on-hand inventory over
10 years = 346.937.

The average safety stock when an order
arrives = 77.46.

The average annual costs are;

Ordering Costs:	\$2,500.00
Holding Costs:	3,989.78
Back Order Costs:	<u>832.50</u>
Total Costs:	\$7,322.28

(b) System One

Number of back orders in 10 years = 582.

The average on-hand inventory over
10 years = 118.696 units.

The average annual costs are;

Ordering and Reviewing Costs:	\$1,004.18
Holding Costs:	1,365.00
Back Orders Costs:	<u>291.00</u>
Total Costs:	\$2,660.18

(c) System Two

Number of back orders in 10 years: 314.

The average on-hand inventory over 10 years =
57.401 units.

The average annual costs are;

Ordering and Reviewing Costs: \$ 589.68

Holding Costs: 660.11

Back Orders Costs: 157.00

Total Costs: \$1,406.79

(2) Total Results

The total number of back orders for whole system per year is $166.5 + 58.2 + 31.4 = 256$.

The total annual variable cost for whole system is $\$7,322.28 + \$2,660.18 + \$1,406.79 = \$11,389.25$.

3. Comparison of Both Starting Conditions

It is obvious that if both results are compared there is a considerable decrement on annual back orders but not a great difference in annual variable costs.

$$\frac{390-256}{390} = 0.3436$$

$$\frac{11672.81 - 11389.25}{11672.81} = 0.02429$$

IV. COMPARISON OF ANALYTICAL AND COMPUTER SIMULATION RESULTS

By comparing the analytical results with the simulation results, it is possible to evaluate the analytical model. If the measures of effectiveness predicted by the analytical model are reasonably close to those generated by the simulation, there is support for the analytical results. In the table below we compare the main measures of effectiveness: back orders and costs.

	<u>Total Back Orders Per Year</u>	<u>Total Variable Costs Per Year</u>
Analytical result	175.56	\$10,808.17
Simulation result	390.00	\$11,672.81 or
(For both starting conditions)	256.1	\$11,389.25

This table shows that the average yearly back orders are considerably higher than what is estimated by the mathematical solution. In fact, the total number of back orders generated in the simulation does not even meet the constraint. Since the solutions produced by the simulation model, especially with respect to the number of back orders do not agree with the mathematical model, this suggests that some of the assumptions made in deriving the mathematical results are not reasonable. The major differences between the simulation results and the analytical results are found in the measures

of effectiveness at the main system. The results for the lower echelon track reasonably closely.

The costs are also a little higher than what is expected but much closer percentage wise.

As an explanation for the large differences observed in the number of back orders at the main system with the simulation and analytical models, let us reexamine the assumptions we made that affect back orders. Back orders results from inadequate amounts of safety stock to protect the inventory system against excessively large numbers of demands in a lead time. The safety stock is manipulated by control of the reorder point r . After the reorder point is hit and an order placed, the system is totally at the mercy of the demands that occur during the lead time. If the variance of lead time demand is underestimated, then large stockouts will occur, even if the mean lead time demand is estimated accurately. The lead time demand is affected by two random quantities: (1) the demand distribution and (2) the lead time distribution.

For purposes of making the mathematics tractable, we assumed in our analytical model that demands at the main system flowed in at a smooth continuous rate λ which was taken to be the sum of the rates of demand incurred directly at the main system λ_m and those which occurred at the lower echelon systems λ_1 and λ_2 . The lead times were assumed to

be normally distributed with mean equal to $\lambda\tau$ and variance $= \lambda\tau(1-\lambda\beta)$ where τ is the mean value of the lead time.

Because of the relatively high demand rates used in the example runs, the normal assumption should be justified by the Central Limit Theorem. However, as is illustrated by the Versatec plot output (See Appendix F) the demands at the main system are far from smooth. What happens is the demands which arrive directly at the main system cause the inventory position to drop off smoothly. However, when the replenishment orders from the lower echelon are received, large drops occur in the inventory position of the main system. Recall that the lower echelon systems order in batches once each period from the main system. If the main system simply tries to average out demands (as assumed by the analytical model), it will sometimes have very large amounts of excess stocks when shipments arrive and sometimes very large numbers of back orders. The high variance in lead time demand caused by the irregular demand actually seen at the main system causes the problem. Since all demands eventually flow through the main system and this is assumed by our analytical model, the problem cannot lie in the value used for the mean lead time demand.

Let us explore further what happens when reorders are triggered at the main system. Demands directly at the main system eat away smoothly at the inventory position. Then a

very large quantity, say X , is demanded by one of the lower echelon systems. There is a very good chance that the large order placed by the lower echelon system will trigger a reorder by the main system. However, if the demand causes a large overshoot of the reorder point, the main system may have much less stock to live off of until the shipment arrives. For example, suppose the inventory position is $IP = 347$, the reorder point is $r = 300$, and System One places its resupply order for 200 units. An order will be placed by the main system but instead of having 300 units to keep it going until the order is received, it will have only 147 units. It is clear that if 300 is the amount of stock needed to provide reasonable protection against demands in a lead time, large numbers of back orders would be expected. Effectively, in the example above, the reorder level was not $r = 300$ but $r = 147$ and the safety stock negative. The impact of this surge in demand caused by the batching of demands received from the lower echelon systems is to reduce the "effective" reorder point from the value r to a value $r' < r$. Because the actual demand distribution witnessed by the main system is difficult to describe mathematically, the actual value of r' cannot be determined analytically. However, it is clearly less than r and may be much more so.

In the next chapter we describe a policy modification for operation of the main system that was suggested by the

observations above. The modification attempts to allow the main system to anticipate the surge of demands that will be received by the lower echelon systems.

V. AN ALTERNATIVE SOLUTION

A. DESCRIPTION OF ALTERNATIVE PROCEDURE

Due to the reasons mentioned in Chapter IV, the number of back orders found by simulation were much higher than the number of back orders predicted by the mathematical model. The question is how to run the multi-echelon system so that the large number of back orders seen earlier can be reduced.

Obviously, what needs to be done is to reduce the impact of the very large demands that occur when the lower echelon systems place their resupply orders. With modern day communication and data systems, it would be feasible to allow the main system to "see" every demand that occurs anywhere in the system. If the main system is given the visibility, it will be able to take action to get the stock on its shelves in anticipation of the large demands from the lower echelon systems. The main system can do this if it uses as its reorder point the pseudo inventory position which is like the inventory position except that it decreases only when direct customer demands are encountered at any of the three systems. The pseudo inventory position is unaffected by the batch replenishment demands placed by the lower echelon systems. For example, if a customer requests

a unit from System One, the pseudo inventory position at the main system decreases by one. However, if System One places an order for 200 units at the main system, the pseudo inventory position does not change.

This new policy is referred to in this thesis as the "early warning policy." Clearly, the pseudo inventory position will always reach the reorder point before the inventory position. Therefore, orders will always be placed earlier and consequently, the number of back orders should decrease. The price paid will be in terms of extra holding costs. The results of a simulation using the early warning policy should be more nearly like those of the mathematical model since, in effect, the mathematical model makes the assumption of early warning. Implicitly, assumption of demand which is the superposition of the direct demands occurring at the main system and Systems One and Two is equivalent to the "early warning assumptions."

In the next section, results are given of a simulation of the multi-echelon system with early warning. The flow-chart of the simulation model is given in Appendix B. The actual FORTRAN computer program is given in Appendix E.

B. COMPUTER SIMULATION RESULTS FOR EARLY WARNING POLICY

The same starting conditions and input parameters used in the previous simulation run were utilized here. The results are summarized as follows:

1. Main System Results

The number of orders in 10 years = 50.

Total number of back orders in 10 years = 126.

The average on-hand inventory = 488.093.

Average safety stock = 210.96.

The costs are:

Ordering costs	=	\$2,500.00
Holding costs	=	5,613.00
Back order costs	=	63.00
Total costs		<u>\$8,176.07</u>

2. System One Results

Total number of back orders in 10 years = 665.

The average on-hand inventory over 10 years = 117.846.

The costs are:

Ordering and reviewing costs	=	\$1,006.18
Holding Costs	=	1,355.23
Backorder costs	=	332.50
Total costs		<u>\$2,691.91</u>

3. System Two Results

Total number of back orders in 10 years = 312.

The average on-hand inventory over 10 years = 58.552.

The costs are:

Ordering and reviewing costs	=	\$ 589.68
Holding costs	=	673.35
Back orders costs	=	156.00
Total costs		<u>\$1,419.03</u>

C. COMPARISON OF EARLY WARNING POLICY SIMULATION RESULTS
WITH ANALYTICAL AND FIRST SIMULATION RESULTS

The comparison will be done with respect to total yearly back orders and total variable system costs (considering all three system entities).

	<u>Entire System</u>	
	<u>Number of Back Orders Per Year</u>	<u>Total Cost</u>
Analytical Result	175.56	\$10,808.47
First Simulation Results	256.1	11,389.25
Early Warning Simulation Results	110.3	12,287.01

Since the effects of implementing the early warning policy will be observed primarily at the main system, we also produce the results obtained for the main system individually.

	<u>Main System</u>	
	<u>Numbers of Back Orders Per Year</u>	<u>Total Cost</u>
Analytical Result	108.05	\$6,796.50
First Simulation Result	166.50	7,322.28
Early Warning Simulation Results	12.6	8,176.07

This table shows the differences better than the first one. The number of back orders decreases 88.3 percent, simultaneously as the costs increase about 10.44 percent. The reason for the higher cost is because the increment in safety stock increases the carrying costs.

VI. CONCLUSIONS

The differences between the results of the mathematical and the simulation model can be explained largely as a result of the assumptions made about the demand process. In the main system it was assumed that the demand was smooth (the superposition of three Poisson processes), but in the simulation model the actual demand at the main system was as it would be in actual practice. There were batches of demand placed by System One and System Two in addition to the individual customer demands directly at the main system. These demand batches in fact increased the variance in lead time demand beyond that modelled. This explains why many more back orders were generated in the simulation model than what was predicted by the mathematical model.

The simulation model developed in this thesis is useful for making comparisons and examining the effects of policy changes or parameter changes. Moreover, it is one of the best ways to check the reasonableness of all of the simplifying assumptions made in order to obtain analytical solutions.

The mathematical results described in this thesis do not adequately determine the reorder point. The predicted number of stockouts is much less than the simulated numbers. As explained earlier, this is probably due to the assumptions

made in the model about the variance of lead time demand. Further study is needed to determine what could be done in the analytical model to better approximate what happens in actual practice.

The early warning policy discussed in this thesis did provide for great reductions in the number of stockouts system wide. Since stockouts are probably the most important consideration in military supply systems, the early warning policy is recommended, even though the holding costs are larger. Additional study is required to see if the reorder level in the early warning policy can be reduced substantially from the value determined by the mathematical model. Preliminary evidence is that the reorder level can be reduced significantly (25 percent or so) without generating excessively many back orders if the early warning policy is used.

Our simulation models allow us to view the effect of changes but they cannot be used to optimize the values of the policy parameters. For that objective, additional work in the mathematical modelling area is required.

In this thesis we have tried to model a multi-echelon inventory system analytically, by linking together the individual echelons and locations through a single objective function and a constraint on backorders system wide. We, knowingly, were making various simplifying assumptions to facilitate the derivation of solutions. As reported above,

the resulting solution for the reorder level at the main system led to many more back orders than predicted. The other solutions; R_1 , R_2 , Q , T_1 , and T_2 appear to be satisfactory.

In order to model adequately the multi-echelon system, it will probably be necessary to build into the determination of the reorder lead at the main system the values of the parameters R_1 and R_2 at the lower echelon systems. This will be the only way to accurately describe the actual demand process that is observed at the main system. We recommend future work in this area.

APPENDIX A

FLOWCHARTS OF FIRST SIMULATION MODEL

Subroutine One = Ship arrivals to System One

Subroutine Two = Ship arrivals to System Two

Subroutine Three = Ship arrivals to Main System

Subroutine Four = Periodic review of System One

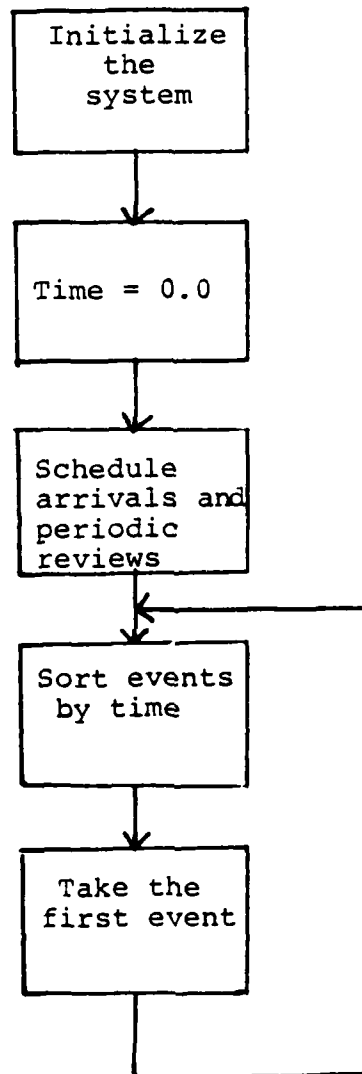
Subroutine Five = Periodic review of System Two

Subroutine Six = Shipment arrival to System One

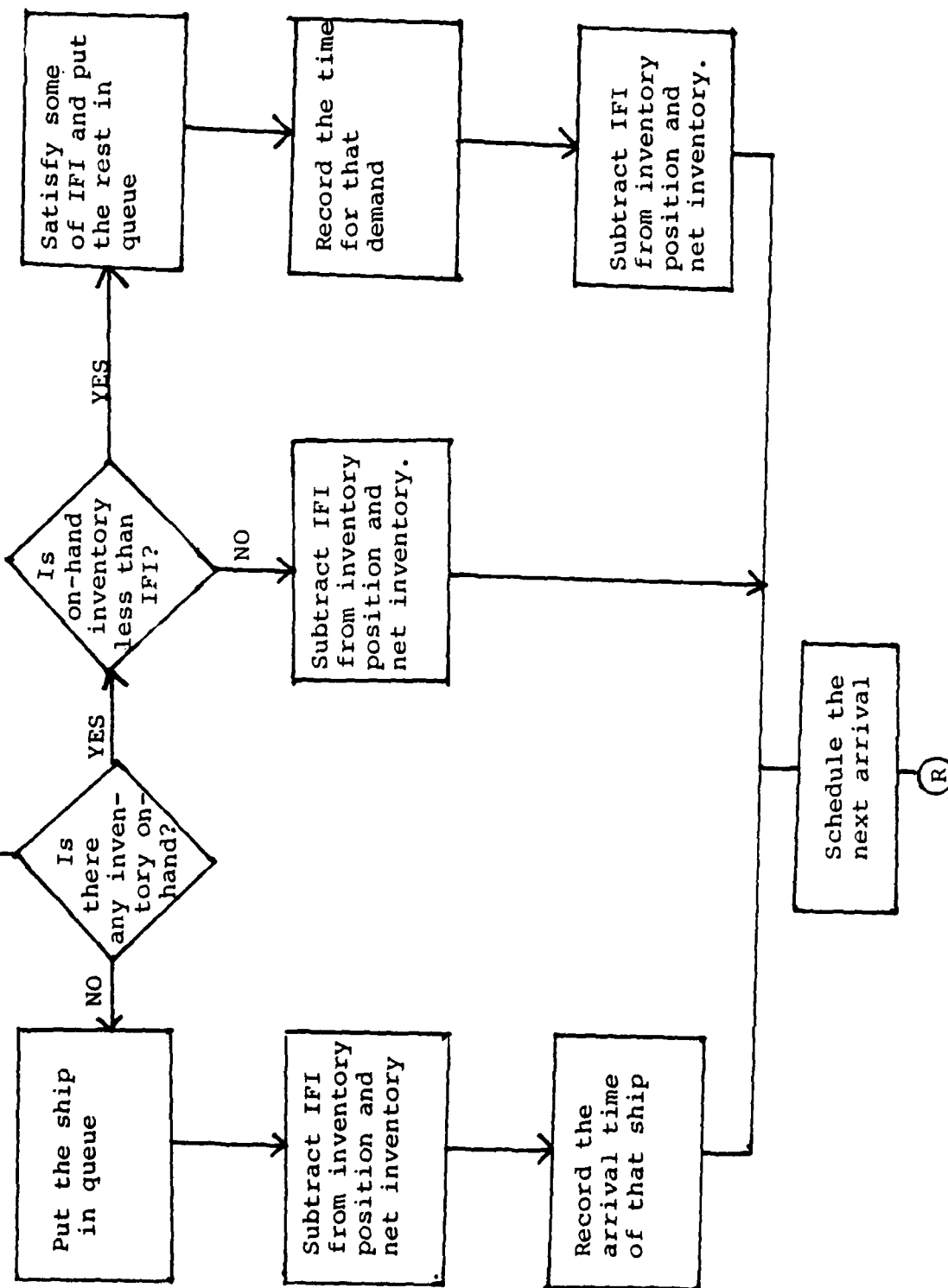
Subroutine Seven = Shipment arrival to System Two

Subroutine Eight = Shipment arrival to Main System

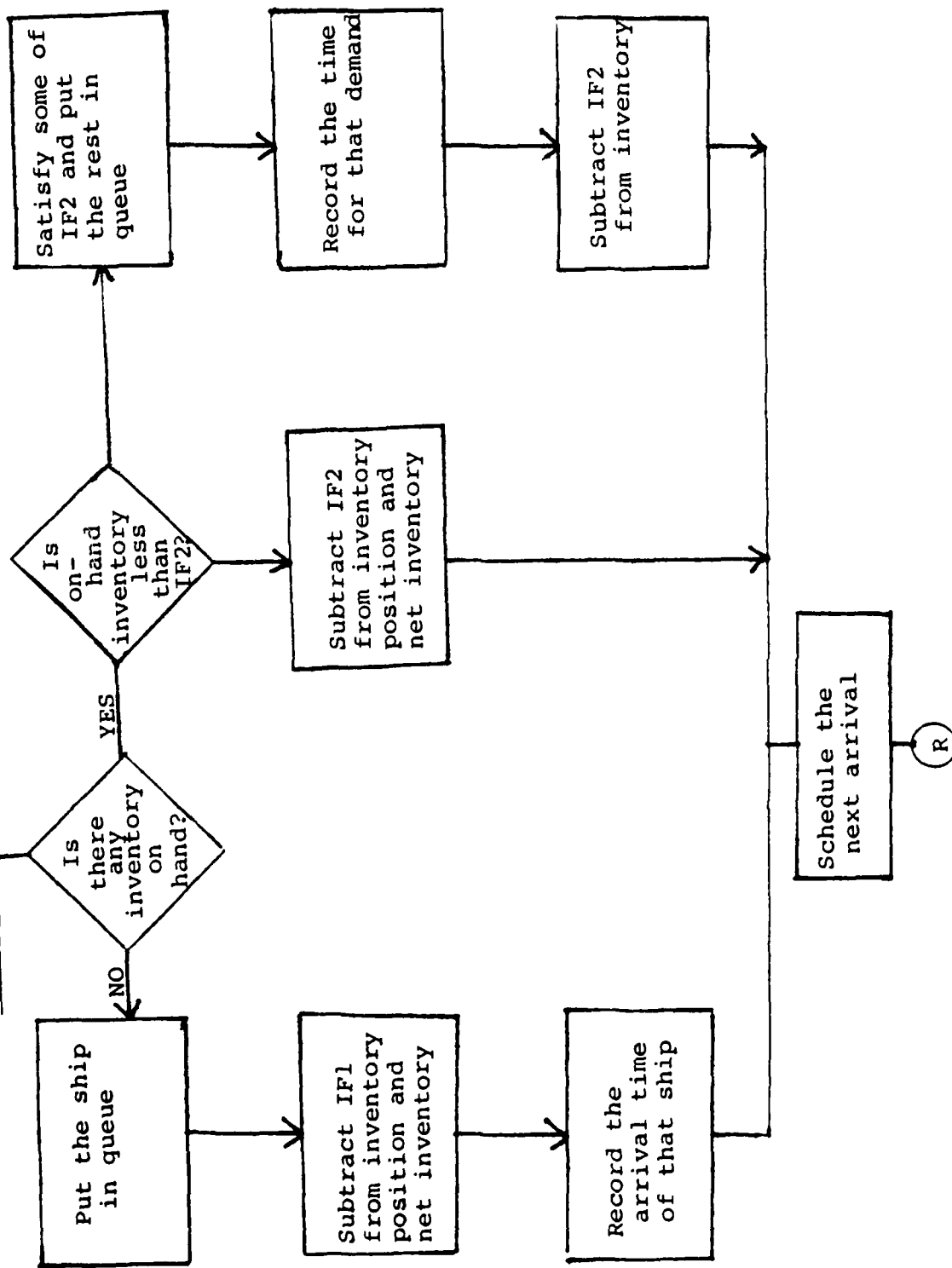
MAIN PROGRAM

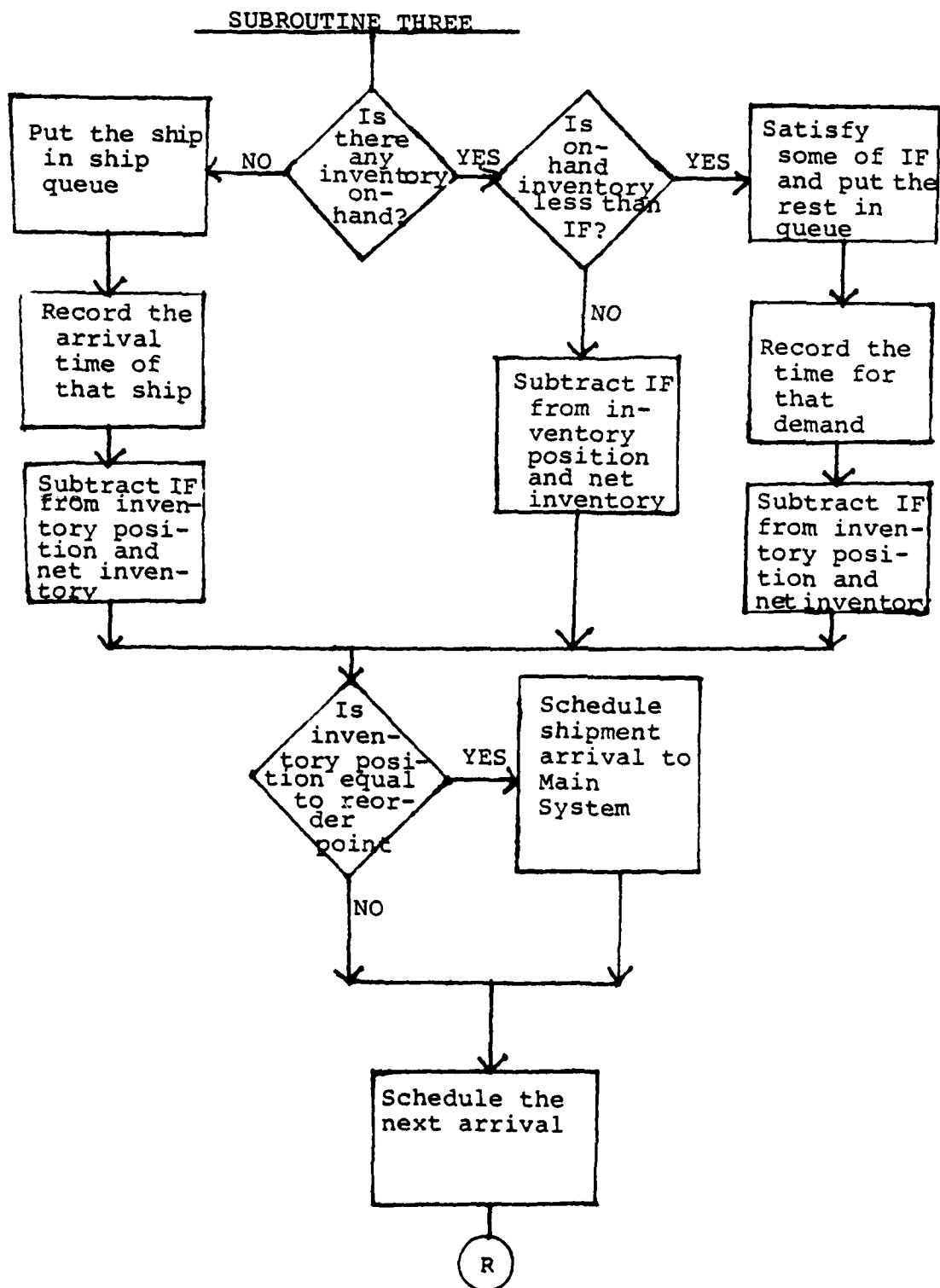


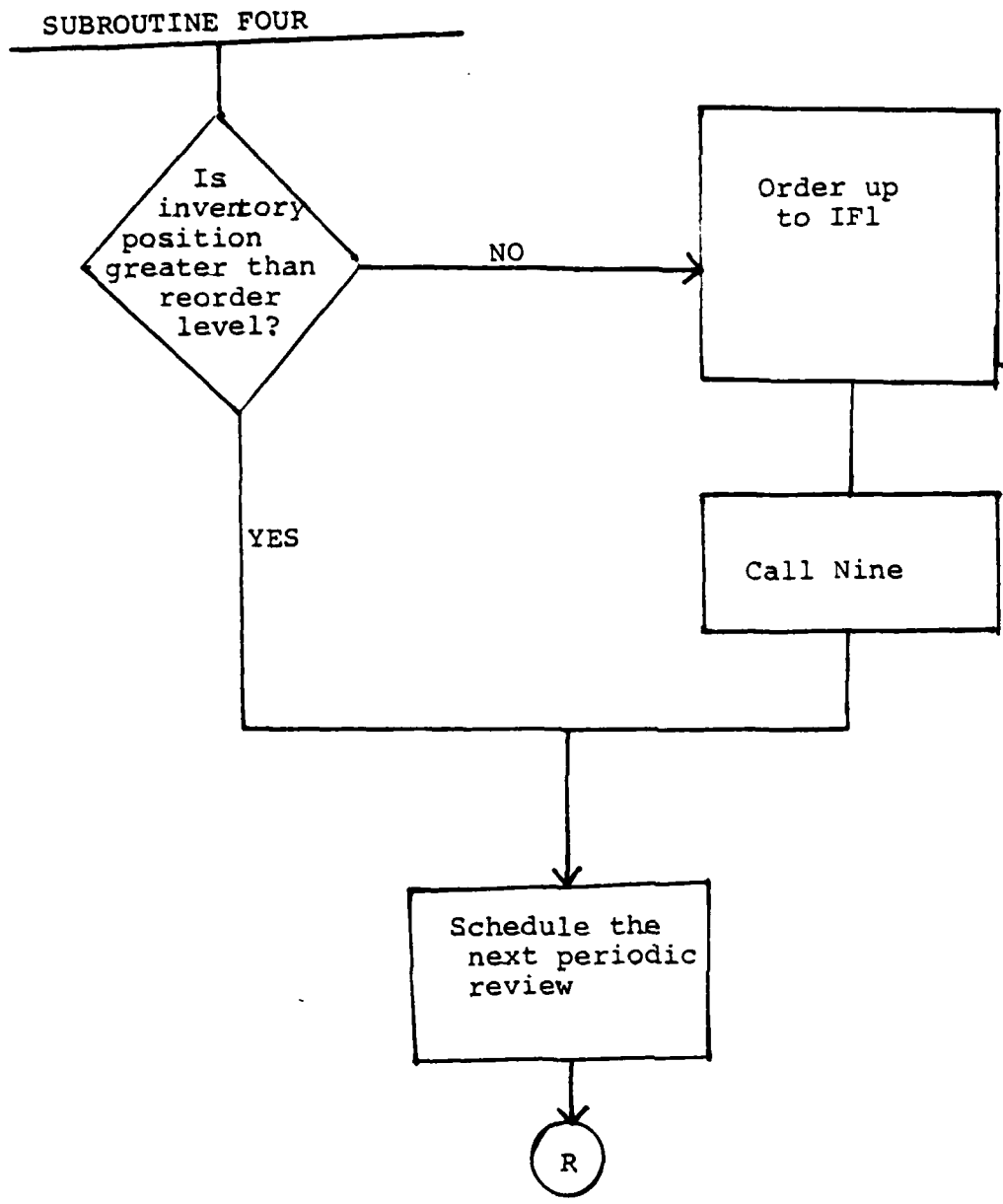
SUBROUTINE ONE

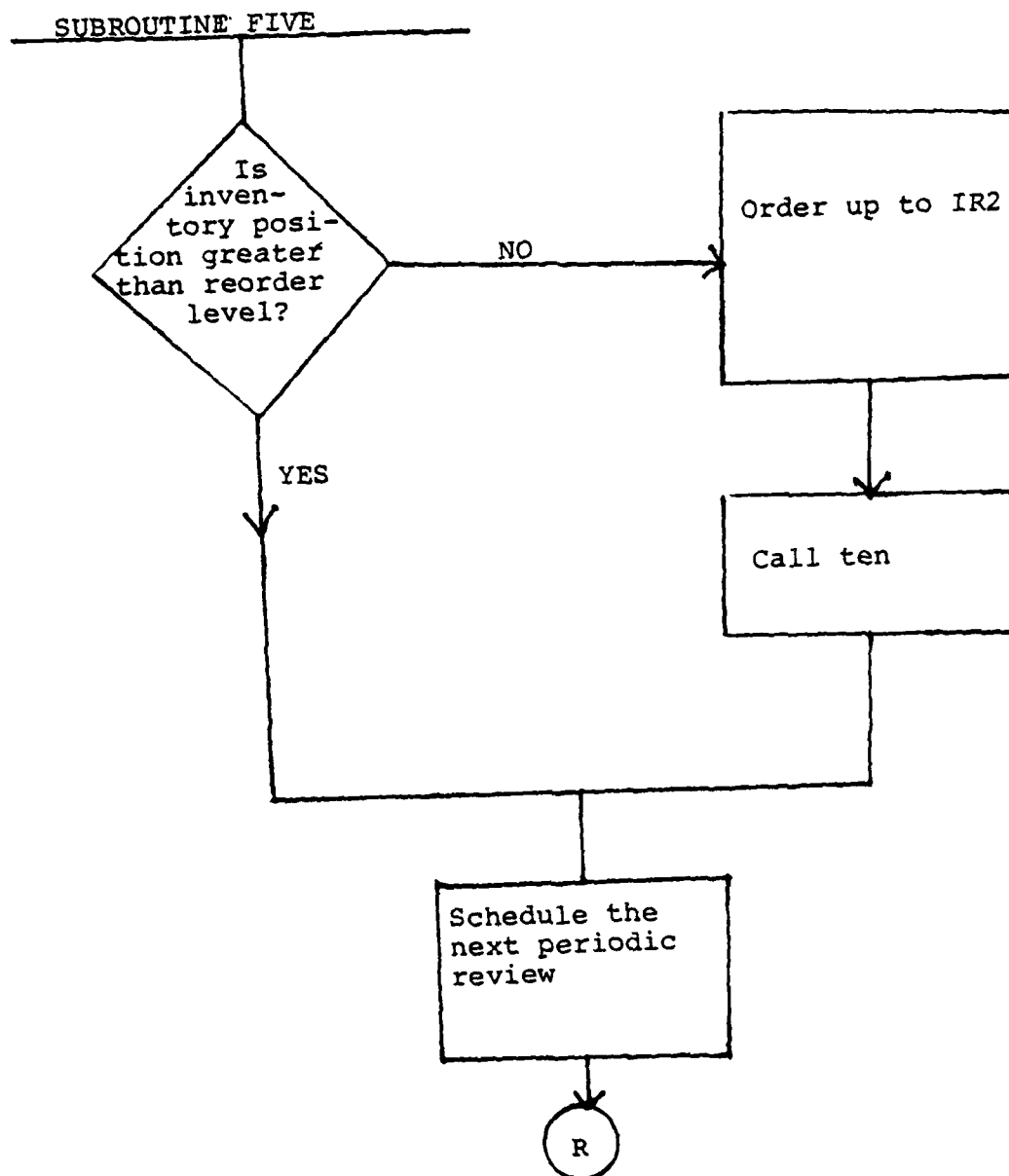


SUBROUTINE TWO

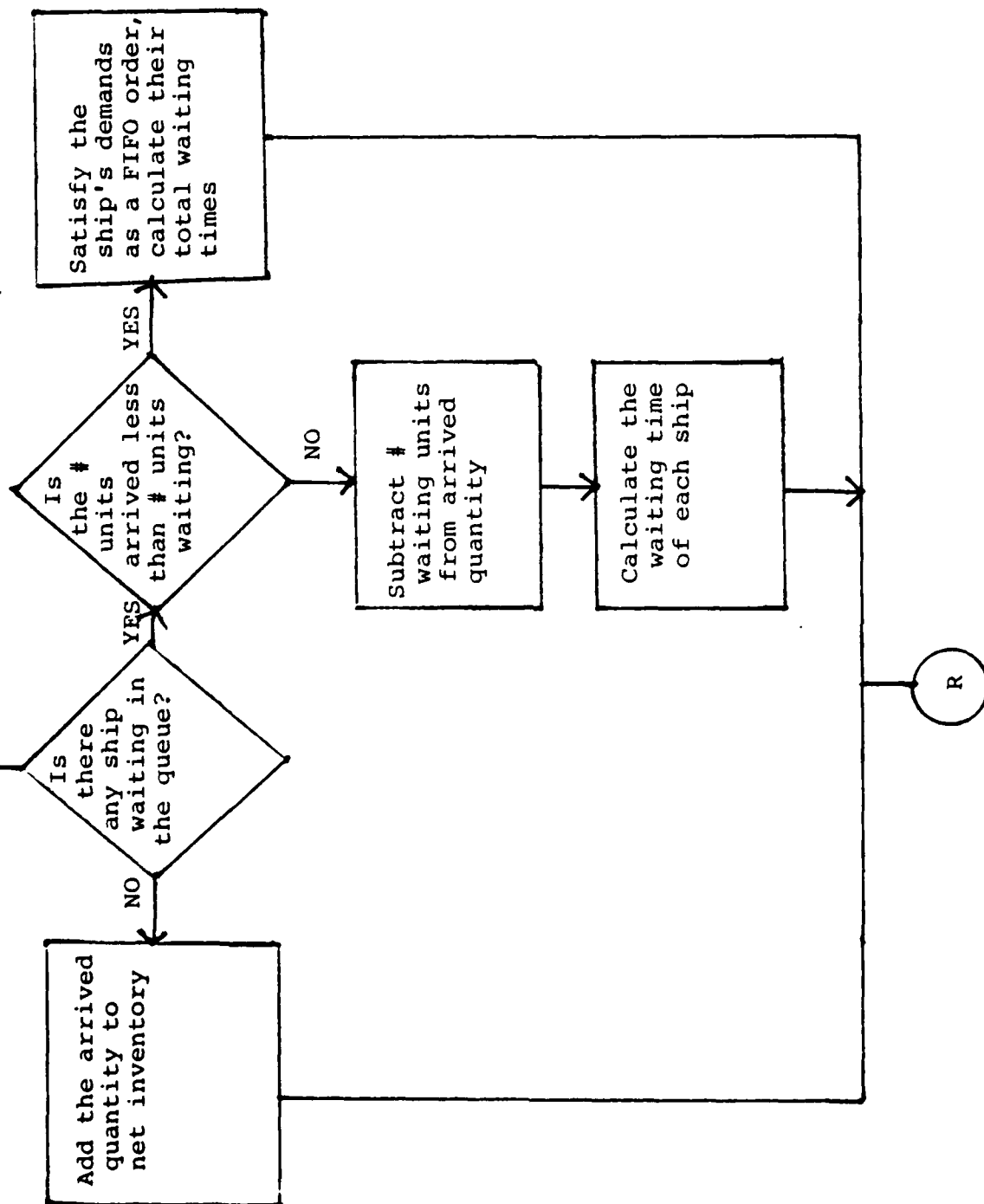




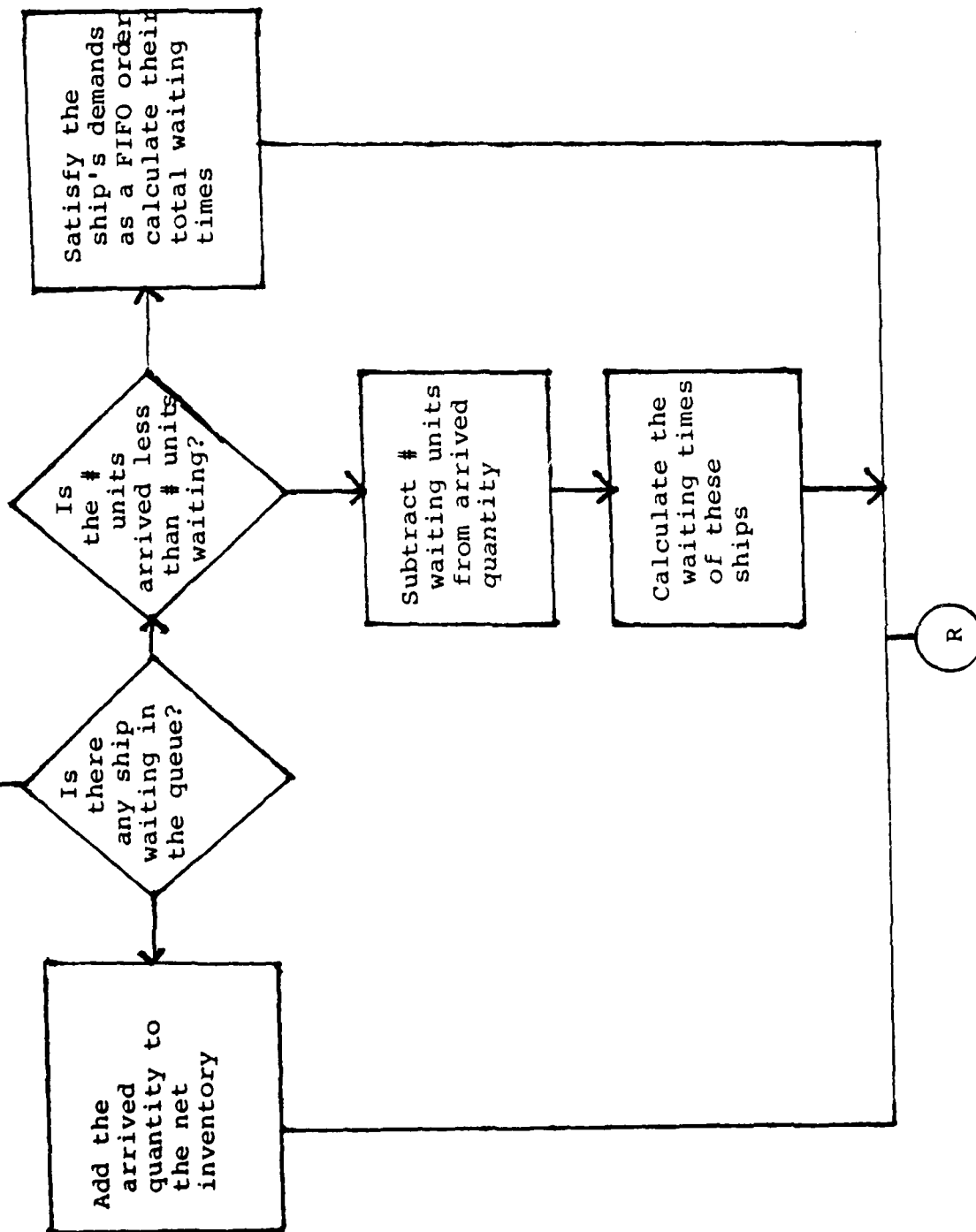




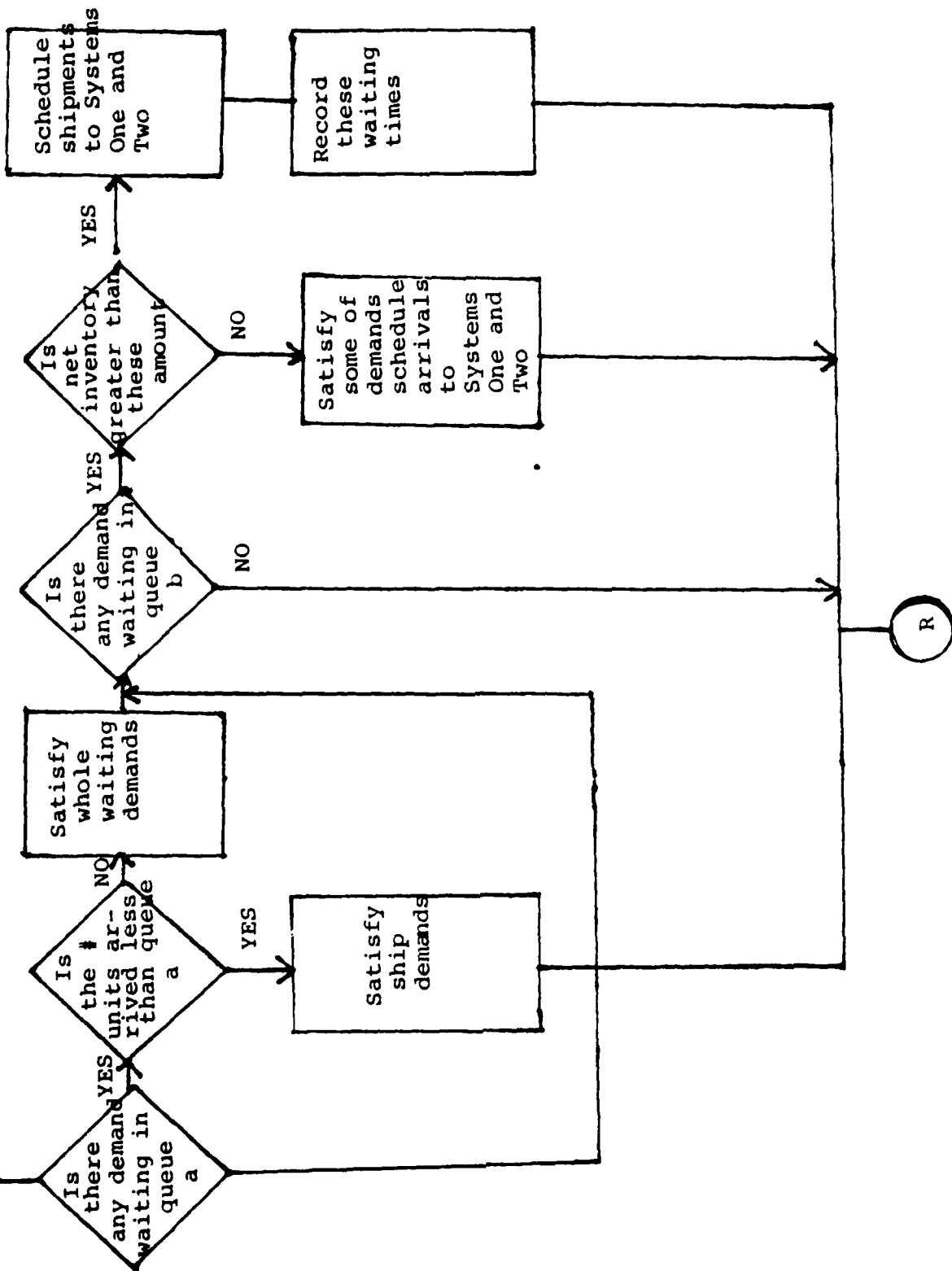
SUBROUTINE SIX



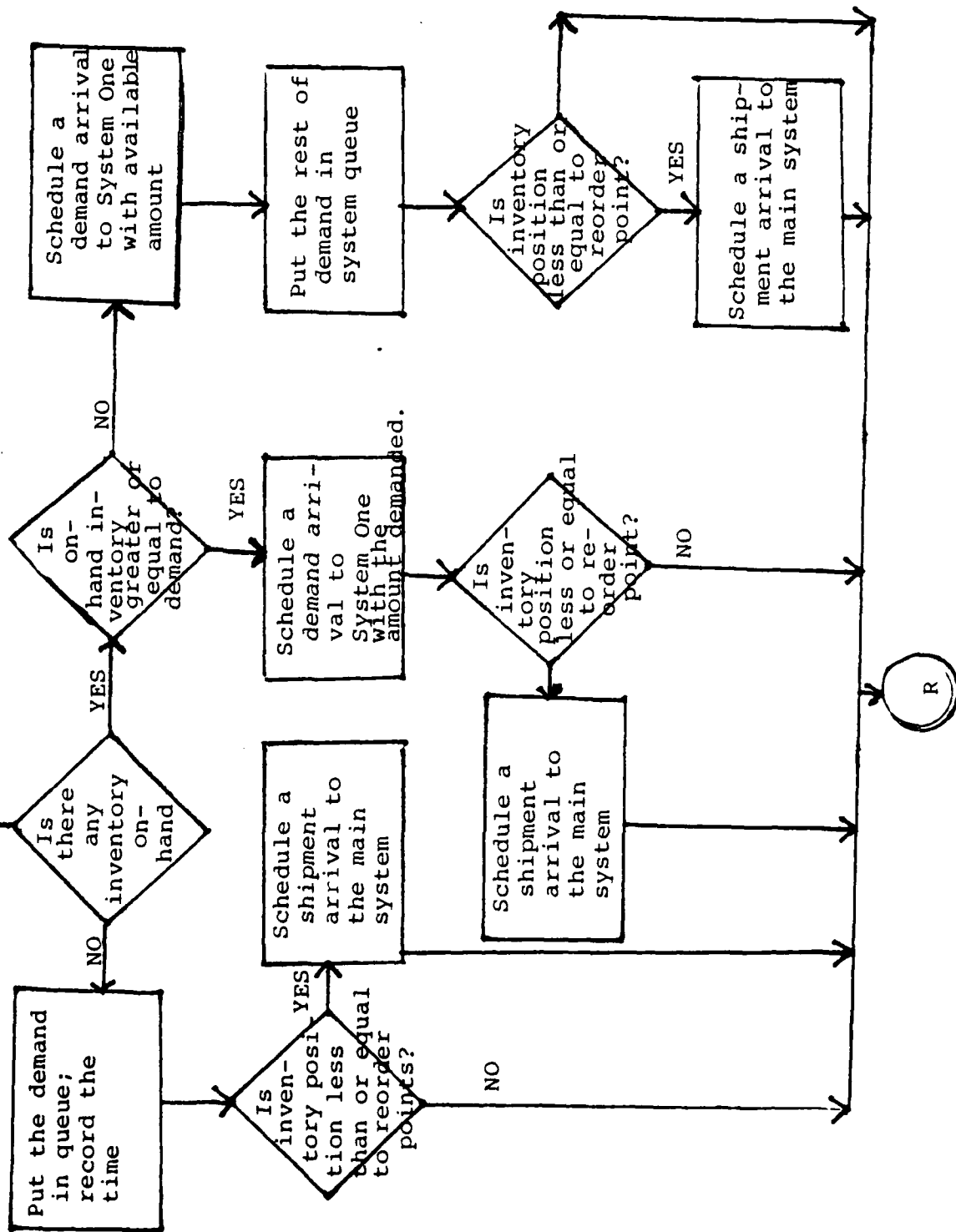
SUBROUTINE SEVEN



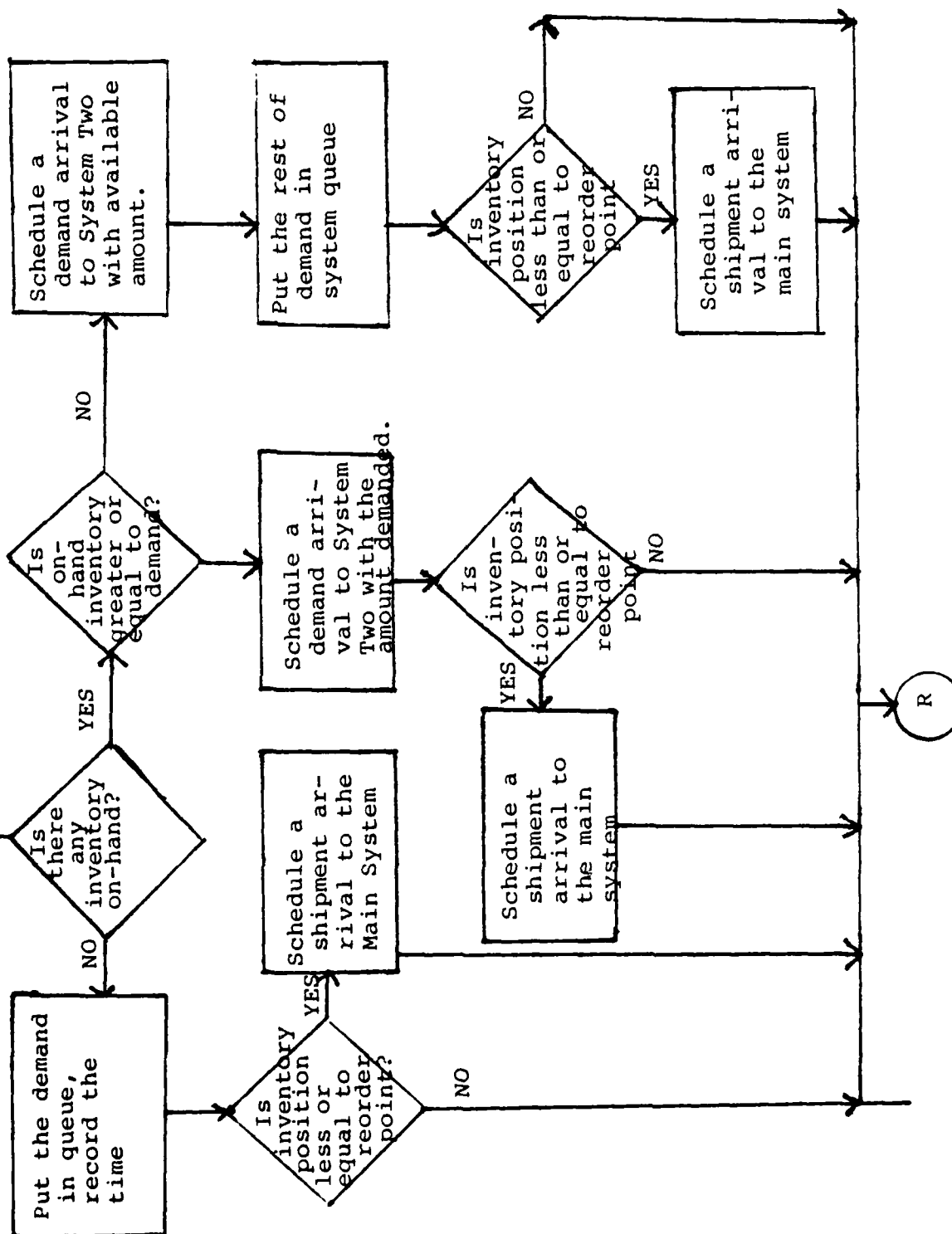
SUBROUTINE EIGHT



SUBROUTINE NINE



SUBROUTINE TEN

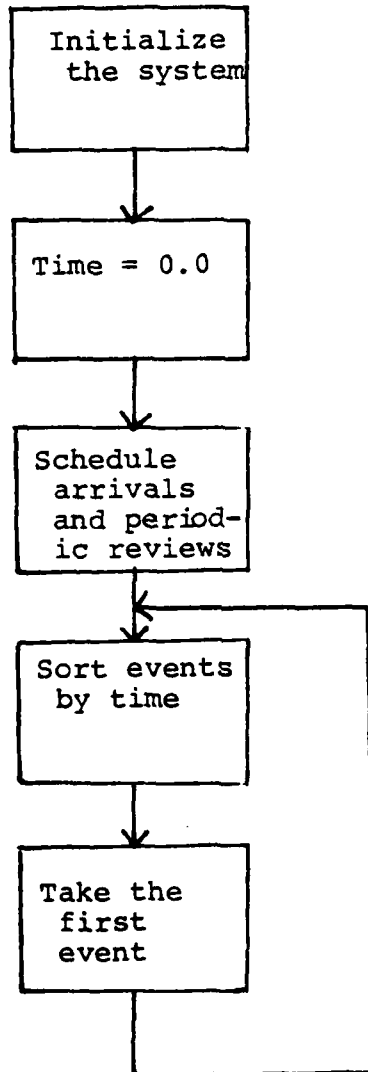


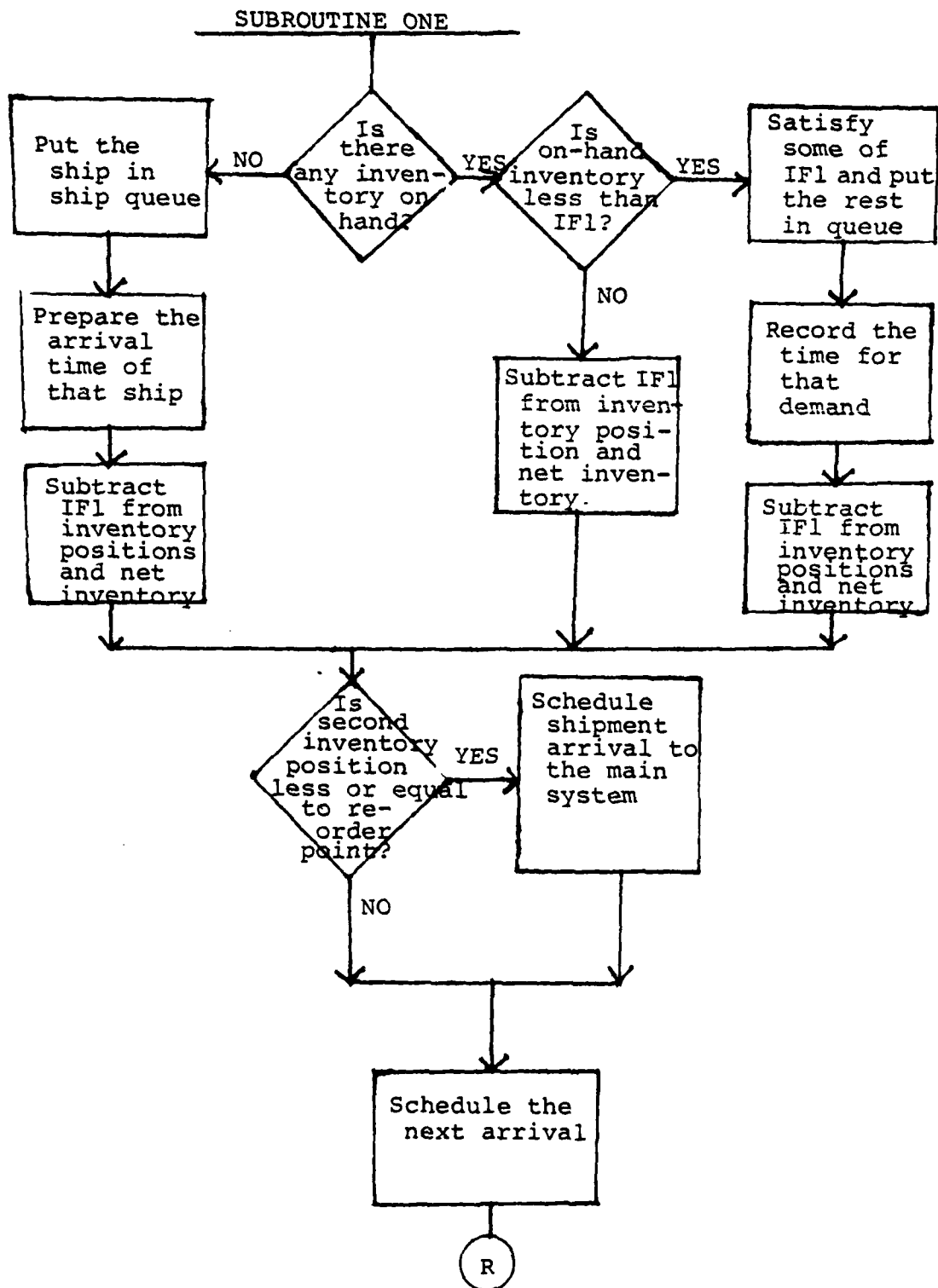
APPENDIX B

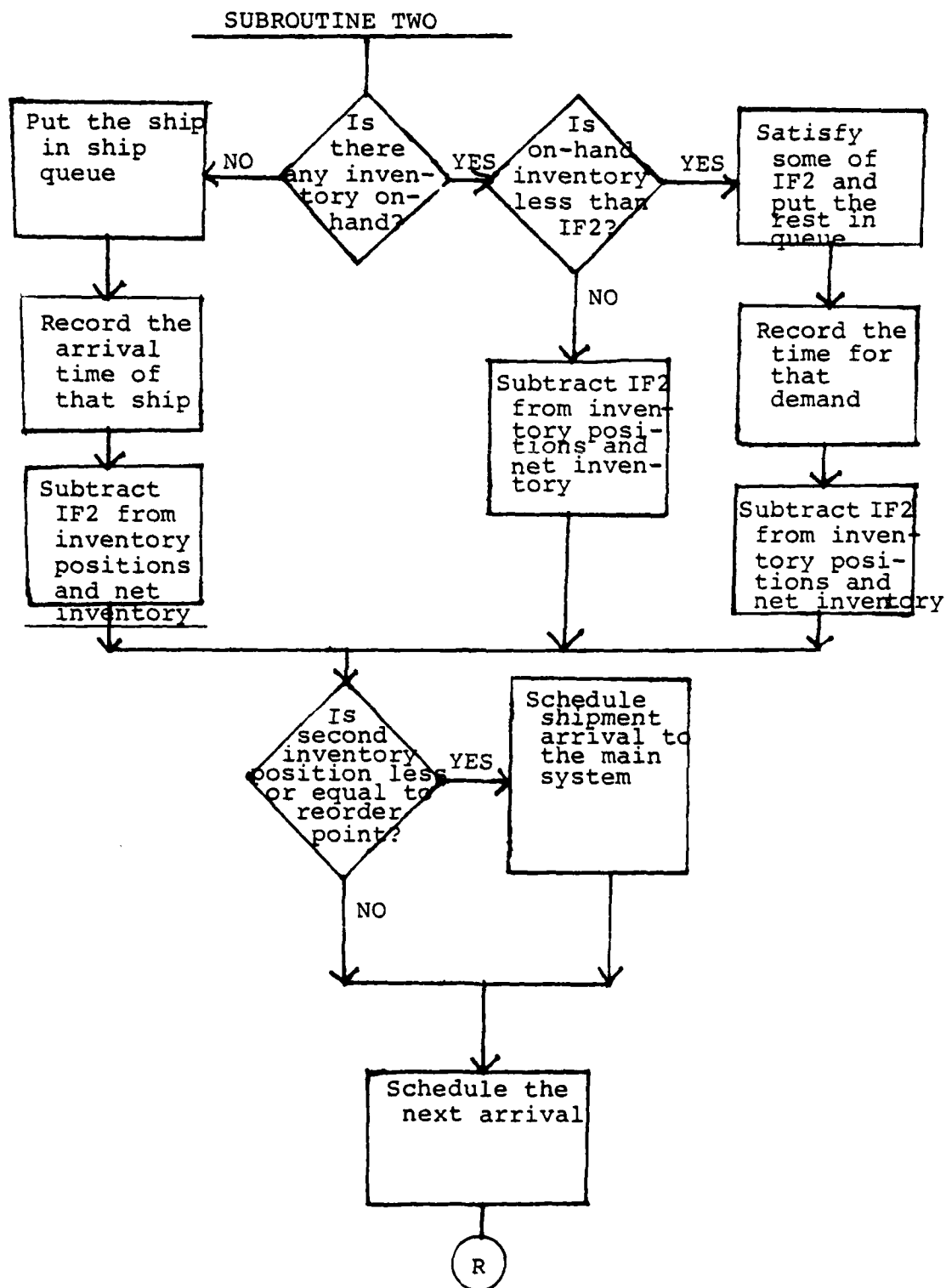
Flowcharts of Early Warning Simulation Model

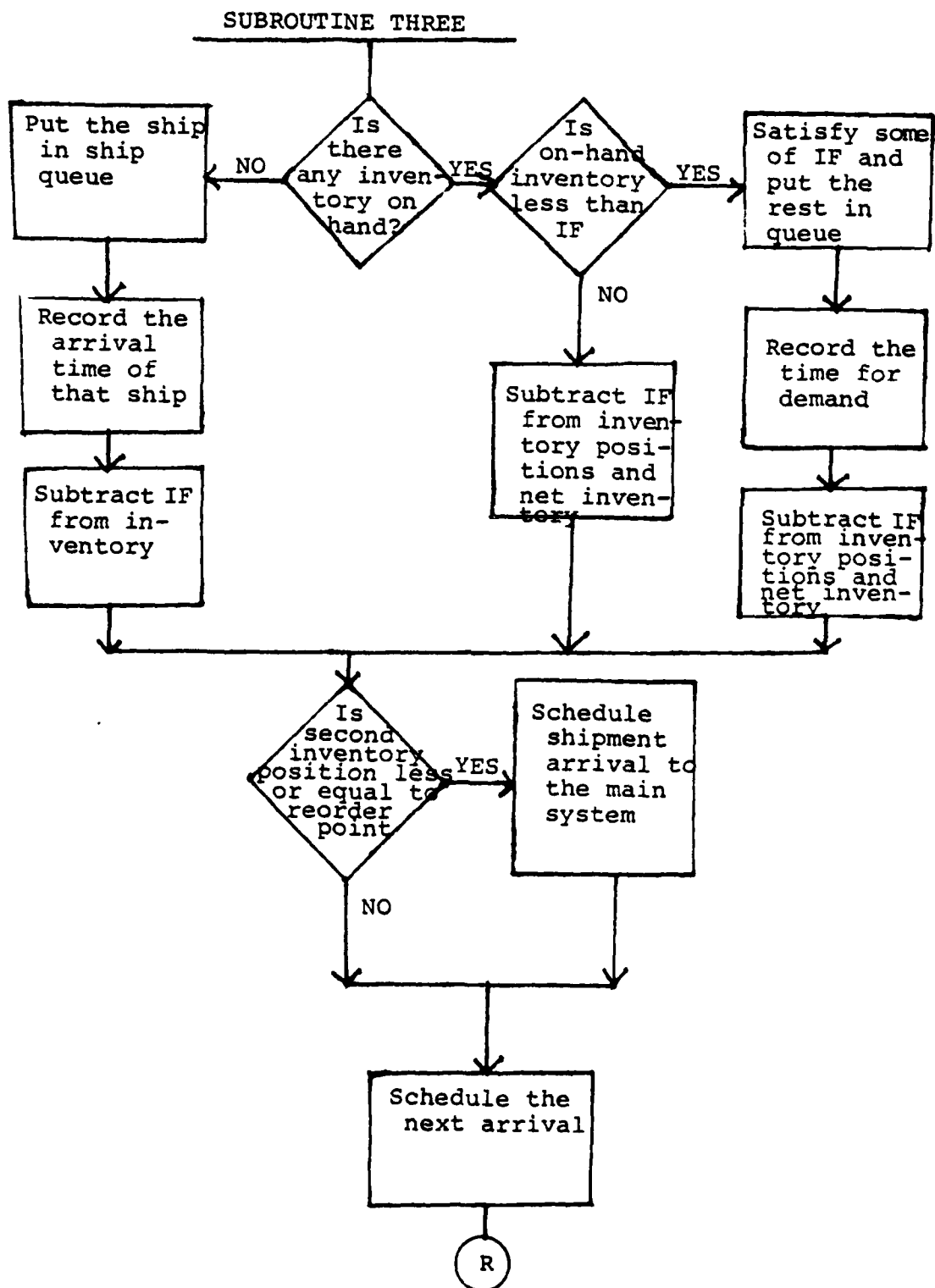
Subroutine One = Ship arrivals to System One
Subroutine Two = Ship arrivals to System Two
Subroutine Three = Ship arrivals to the Main System
Subroutine Four = Periodic review of System One
Subroutine Five = Periodic review of System Two
Subroutine Six = Shipment arrival to System One
Subroutine Seven = Shipment arrival to System Two
Subroutine Eight = Shipment arrival to the Main System

MAIN PROGRAM

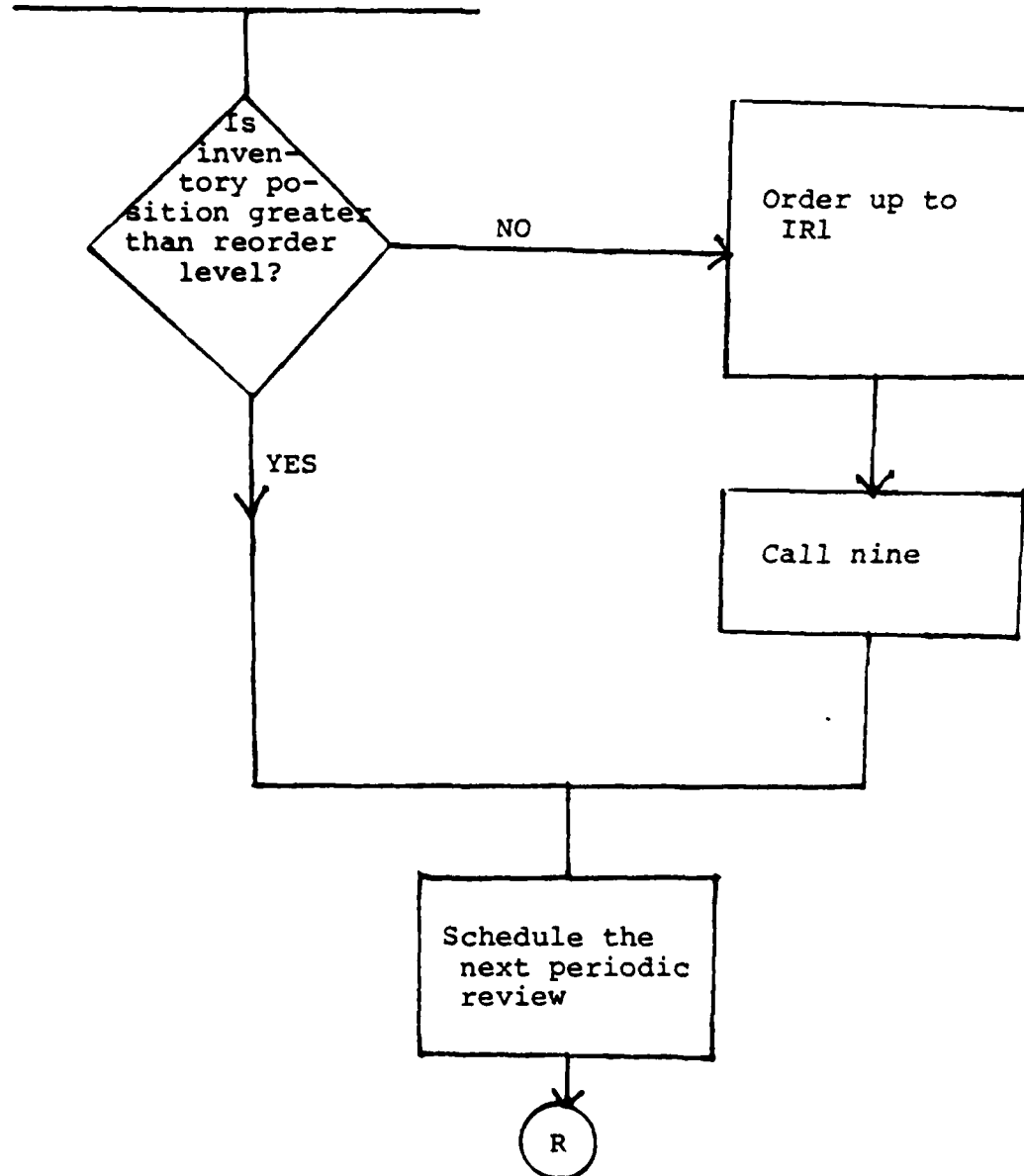


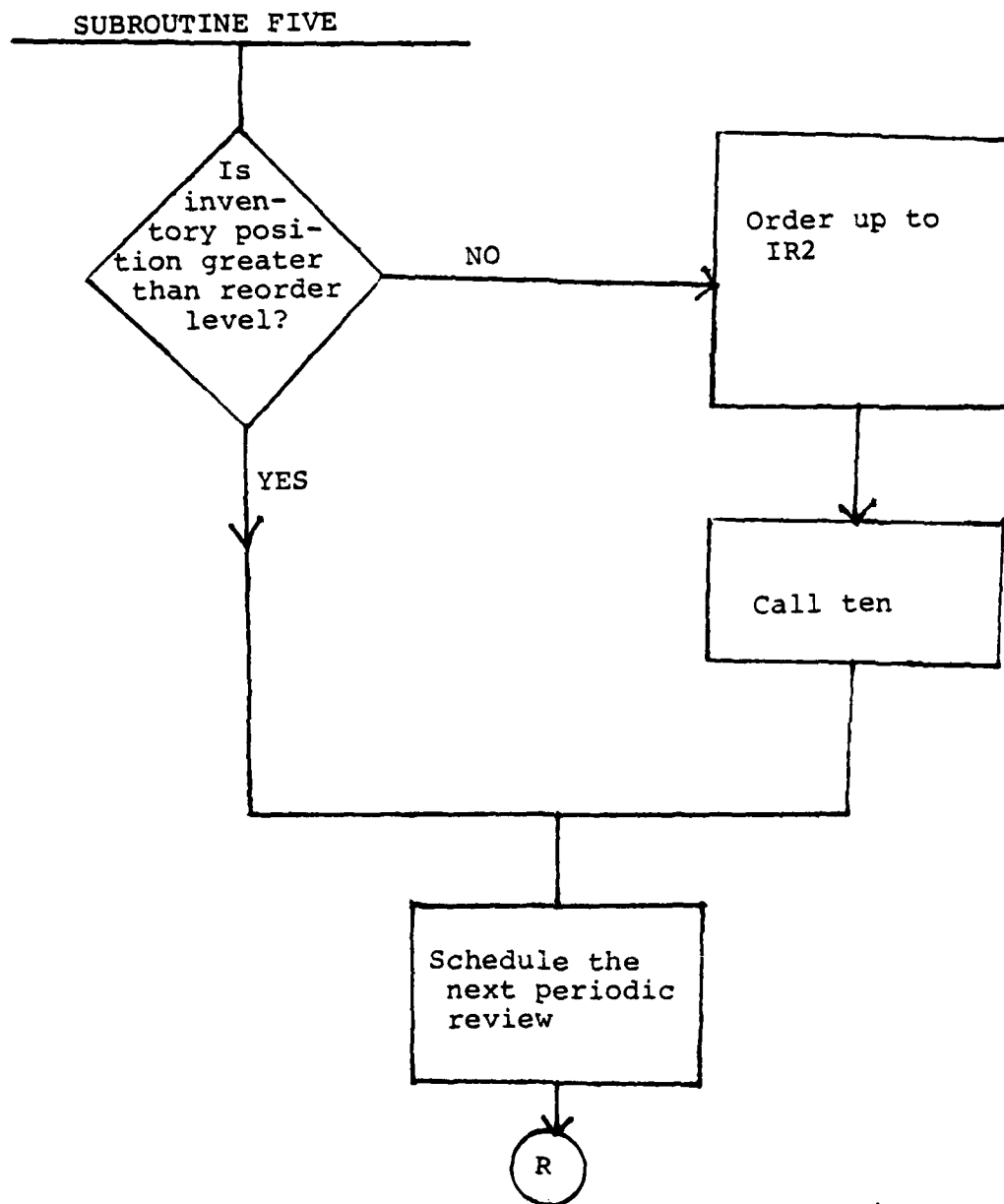




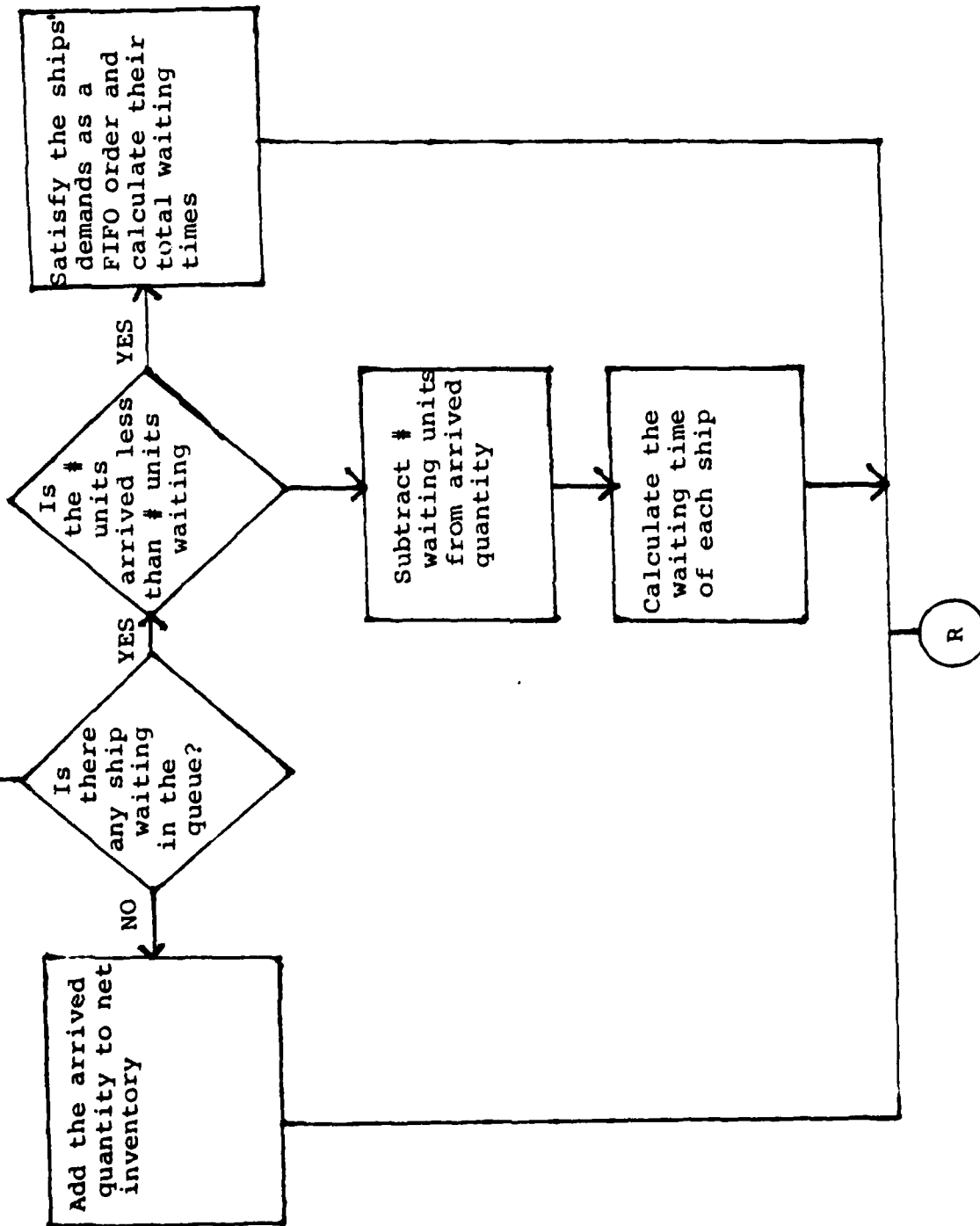


SUBROUTINE FOUR

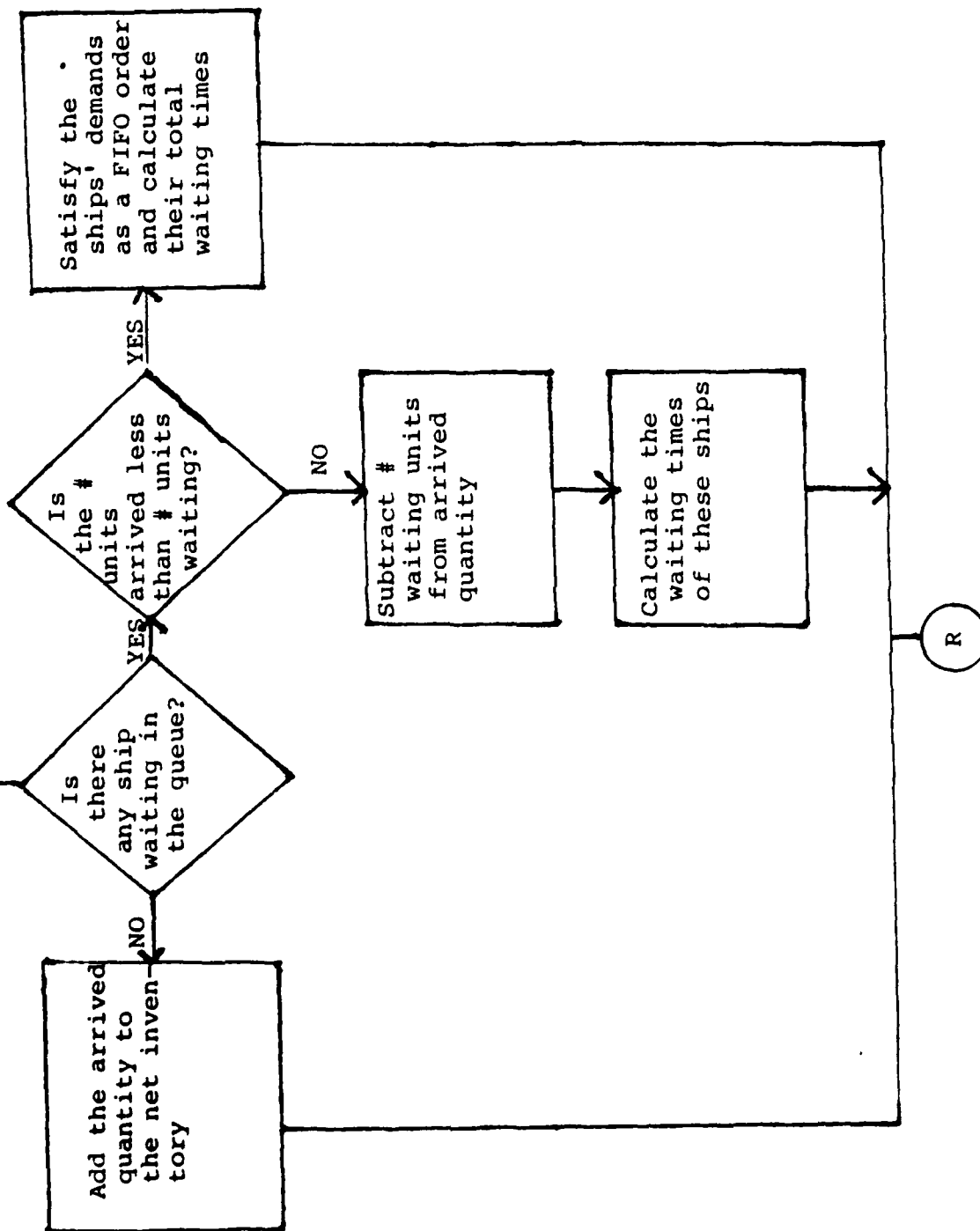




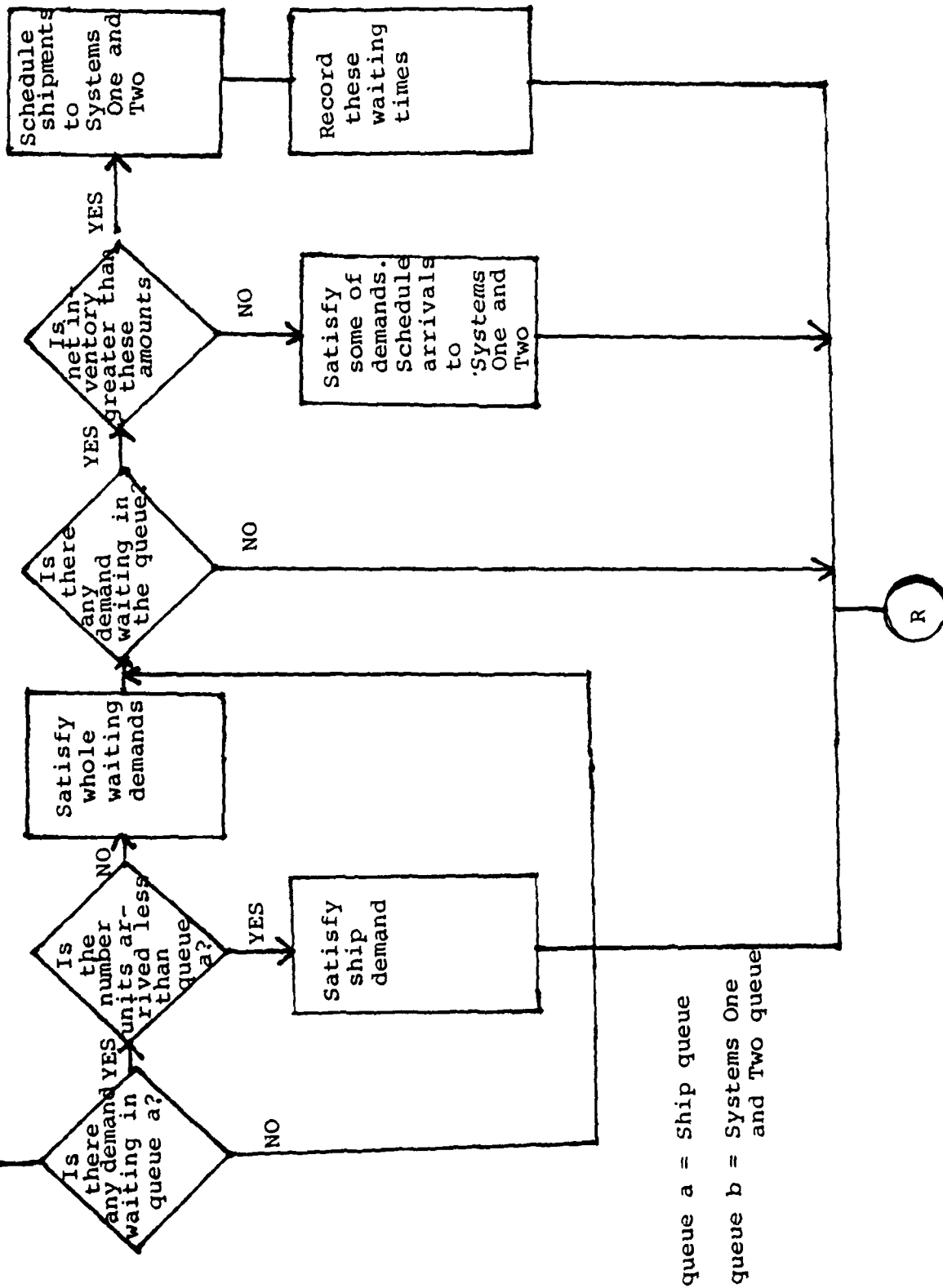
SUBROUTINE SIX



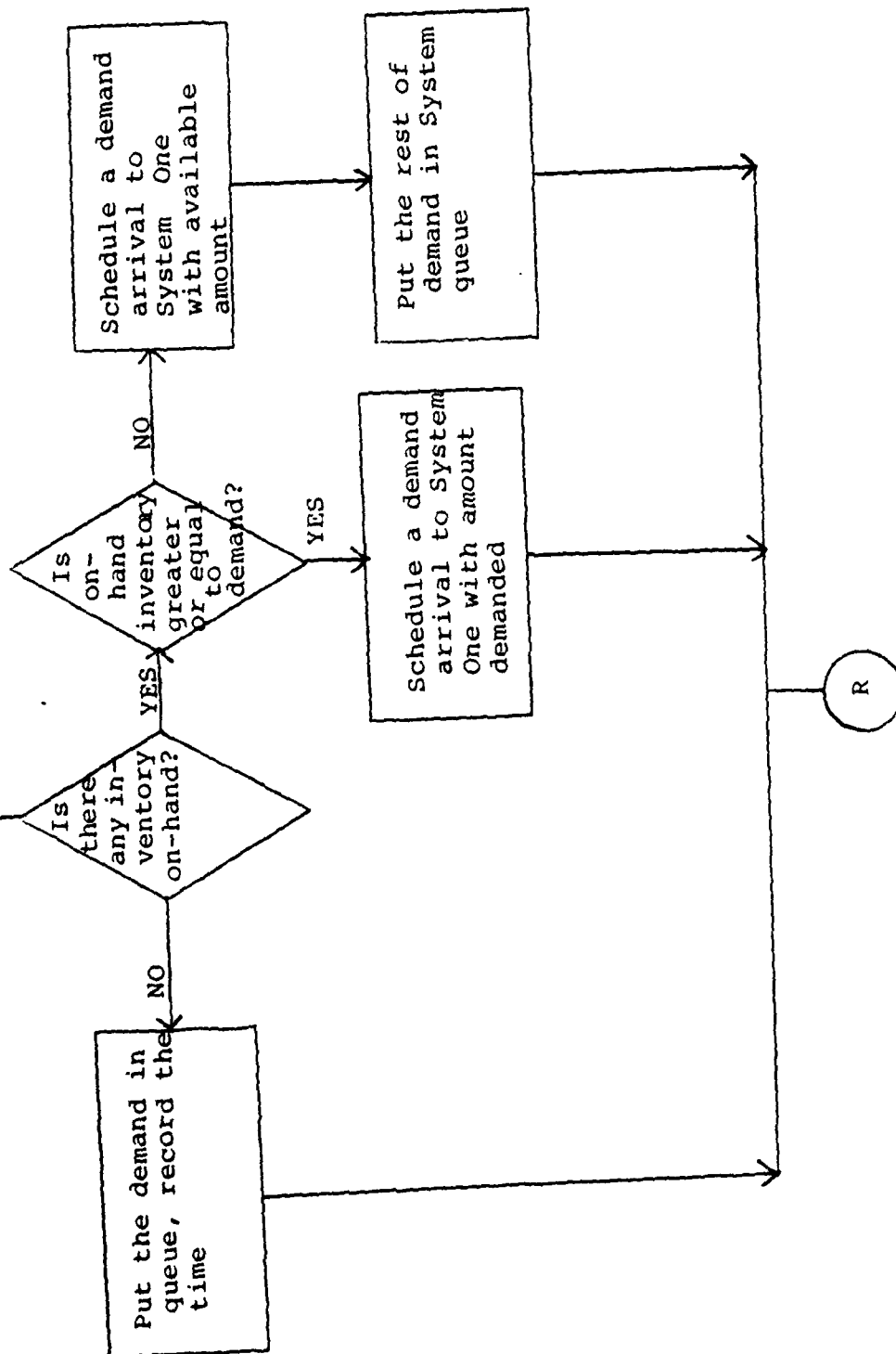
SUBROUTINE SEVEN



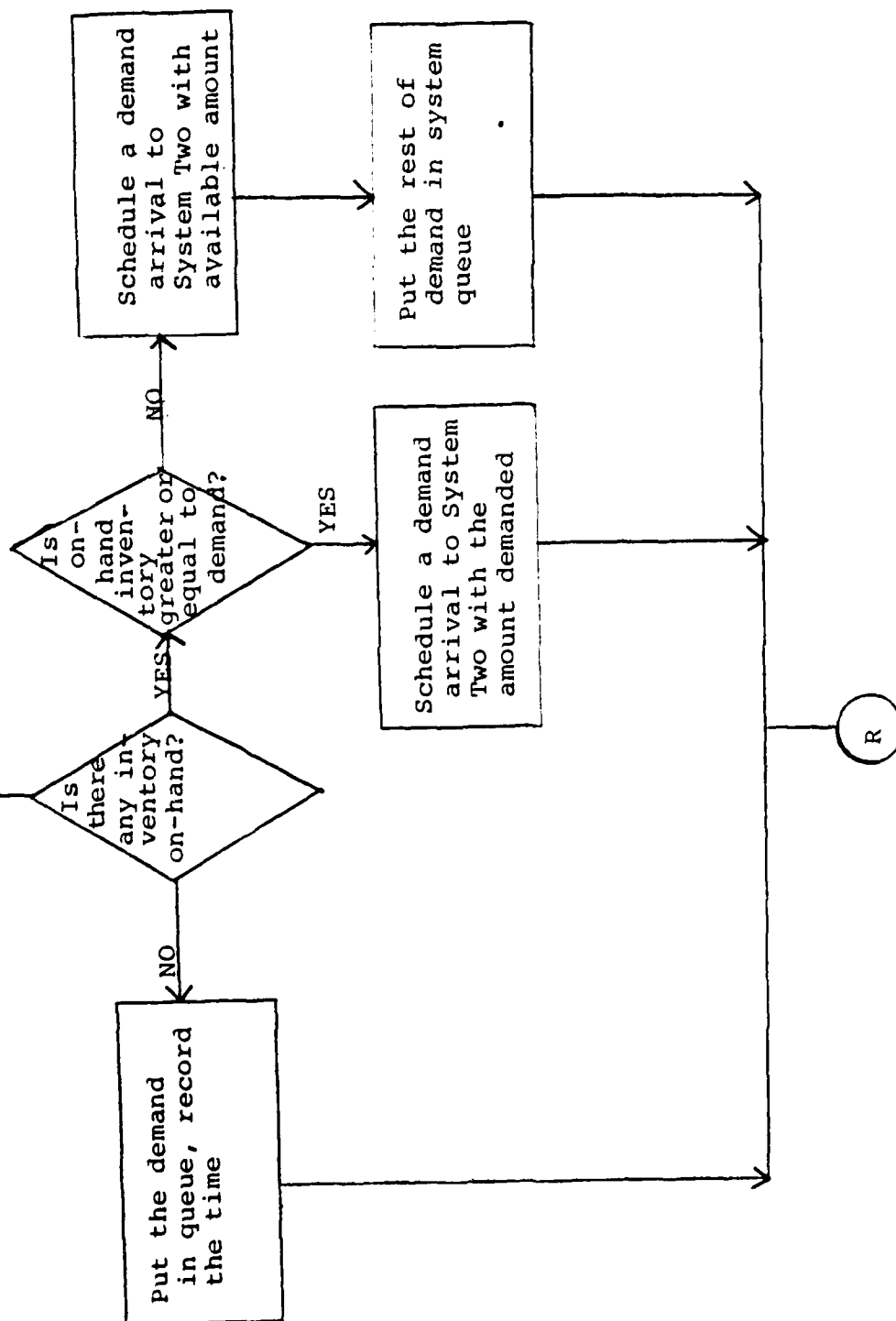
SUBROUTINE EIGHT



SUBROUTINE NINE



SUBROUTINE TEN



APPENDIX C

Simulation Program; Versatec Plotter

```

REAL*8 DSEED1,DSEED2,DSEED3,DSEED4,DSEED5,DSEED6,DSEED7,DSEED8,DSE
IED9
DIMENSION EVENT(900),IEVENT(900),IM(900),QU1(900),QU2(900),QU3(900
1),QS(900),IG1(900),IG2(900),IG3(900),IS(900),IA1(900),IA2(900),IA(
1),Q(900),Z(50),Z1(50),Z2(50),X(4000),X1(4000),X2(4000),V(4000),V1(4000
1),V2(4000),Y(4000),Y1(4000),Y2(4000),WK(50),WS(50),WZ(50),IK1(1),I
1,K2(1),IK3(1),S1(1),S2(1),S3(1)
NP=4000
NS=900
NM=30
DSEED1=123456.000
DSEED2=247658.000
DSEED3=365274.000
DSEED4=258732.000
DSEED5=541863.000
DSEED6=433215.000
DSEED7=651563.000
DSEED8=265418.000
DSEED9=167519.000
I=0
J=0
K=0
L=0
K1=1
K2=1
K3=1
TW1=0.0
TW2=0.0
TW3=0.0
TW4=0.0
TW5=0.0
TOH1=0.0
TOH2=0.0
DO 1 N=1,50
EVENT(N)=99999.
IEVENT(N)=0
QU1(N)=0.0
QU2(N)=0.0
QU3(N)=0.0
IG1(N)=0.0
IG2(N)=0.0
IG3(N)=0.0
IA1(N)=0.0
IA2(N)=0.0
IA(N)=0
1 CONTINUE
DO 2 N=1,10

```

```

IM(N)=0
QS(N)=0.0
IG(N)=0
2 CONTINUE
IF=0
IF1=0
IF2=0
ID=0
ID1=0
ID2=0
READ(5,100) IR,IS
100 FORMAT(2I10)
READ(5,110) IR1,IS1,T1
110 FORMAT(2I10,F10.0)
READ(5,110) IR2,IS2,T2
120 READ(5,120) YEAR
120 FORMAT(F10.0)
READ(5,130) A,B
130 FORMAT(2F4.0)
READ(5,130) A1,B1
READ(5,130) A2,B2
READ(5,140) XM1
READ(5,140) XM2
READ(5,140) XM3
140 FORMAT(F4.0)
READ(5,150) P1
READ(5,150) P2
READ(5,150) P3
150 FORMAT(F6.0)
IQ=IR+IS
IQ1=IR1
IQ2=IR2
XL=(YEAR/365.)*33.
TIME=0.0
IP=IR+IS
IP1=IR1
IP2=IR2
X(K1)=IP
V(K1)=IP
Y(K1)=TIME
X1(K2)=IQ1
V1(K2)=IP1
Y1(K2)=TIME
X2(K3)=IQ2
V2(K3)=IP2
Y2(K3)=TIME
CALL GGEXN (DSEED1,XM1,1,SI)
CALL GGEXT (DSEED7,1,PI,WZ,IK1)

```

```

IF1=1
DO 3 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 4
CONTINUE
3 4 EVENT(II)=TIME+S1(1)
    I EVENT(II)=1
    CALL GGEXN (DSEED2,XM2,1,S2)
    CALL GGEXT (DSEED8,1,P2,WS,IK2)
    IF2=1
DO 5 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 6
CONTINUE
5 6 EVENT(II)=TIME+S2(1)
    I EVENT(II)=2
    CALL GGEXN (DSEED3,XM3,1,S3)
    CALL GGEXT (DSEED9,1,P3,WK,IK3)
    IF=1
DO 7 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 8
CONTINUE
7 8 EVENT(II)=TIME+S3(1)
    I EVENT(II)=3
DO 9 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 10
CONTINUE
9 10 EVENT(II)=TIME+T1
    I EVENT(II)=4
DO 11 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 12
CONTINUE
11 12 EVENT(II)=TIME+T2
    I EVENT(II)=5
DO 13 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 14
CONTINUE
13 14 EVENT(II)=YEAR
    I EVENT(II)=9
    GO TO 26
15 EVENT(IN)=99999.
26 IN=1
DO 16 KI=2,NM

```

```

IF(EVENT(IN).GT.EVENT(KI)) IN=KI
16 CONTINUE
TIME=EVENT(IN)
IL=IEVENT(IN)
IJ=IN
GO TO (17,18,19,20,21,22,23,24,25),IL
17 CALL ONE (I,TIME,EVENT,IEVENT,QU1,IG1,IQ1,IP1,IF1,XM1,P1,DSEED1,DS
    LEED7,K2,NP,X1,Y1,V1,NM,NS,TOH1)
GO TO 15
18 CALL TWO (J,TIME,EVENT,IEVENT,QU2,IG2,IQ2,IP2,IF2,XM2,P2,DSEED2,DS
    LEED8,K3,NP,X2,Y2,V2,NM,NS,TOH2)
GO TO 15
19 CALL THREE (L,TIME,EVENT,IEVENT,QU3,IG3,IQ3,IP3,IF3,XM3,P3,DSEED3,DS
    LEED6,DSEED9,A,B,K1,NP,X,Y,V,NM,NS,IA,IR,IS,TOH)
GO TO 15
20 CALL FOUR (K,IQ,IQ1,IP1,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA1,IR,IR1,I
    S,IP,A,B,A1,B1,DSEED4,DSEED6,K1,K2,NP,X,Y,V,X1,Y1,V1,NM,T1,NS,ISI,
    TOH)
GO TO 15
21 CALL FIVE (K,IQ,IQ2,IP2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IR2,I
    S,IP,A,B,A2,B2,DSEED5,DSEED6,K1,K3,NP,X,Y,V,X2,Y2,V2,NM,T2,NS,IS2,
    TOH)
GO TO 15
22 CALL SIX (I,IJ,TIME,IA1,IQ1,QU1,IG1,TW1,IP1,K2,NP,X1,V1,Y1,NM,NS)
GO TO 15
23 CALL SEVEN (J,IJ,TIME,IA2,IQ2,QU2,IG2,TW2,IP2,K3,NP,X2,V2,Y2,NM,NS
    )
GO TO 15
24 CALL EIGHT (L,K,IJ,TIME,EVENT,IEVENT,IQ,IA,IA1,IA2,IG,QS,IM,QU3,IG
    13,A,B,A1,B1,A2,B2,TW3,TW4,TW5,DSEED4,DSEED5,K1,NP,X,Y,V,NM,NS,IP)
GO TO 15
25 TW1=TW1/YEAR
TW2=TW2/YEAR
TW3=TW3/YEAR
TW4=TW4/YEAR
TW5=TW5/YEAR
WRITE (6,200) TW1,TW2,TW3,TW4,TW5
200 FORMAT (:,TW1=,F10.4, TW2=,F10.4, TW3=,F10.4, TW4=,F10.4,
    ,TW5=,F10.4)
TOH=TOH/YEAR
TOH1=TOH1/YEAR
TOH2=TOH2/YEAR
WRITE (6,201) TOH,TOH1,TOH2
201 FORMAT (:,TOH=,F10.3, TOH1=,F10.3, TOH2=,F10.3)
CALL PLOTG (Y,X,K1,1,1,0,TIME,4,INVENTORY QUANTITY,18,0.0,0.0,
    10.0,0.0,XL,20.)
CALL PLOTG (Y,V,K1,2,1,52,TIME,4,INVENTORY QUANTITY,18,0.0,0.0,
    1,0.0,0.0,XL,20.)

```

```

CALL PLOTG (Y1,X1,K2,1,1,0,'TIME',4,'INVENTORY QUANTITY',18,0.0,0.0,
10.0,0.0,0.0,XL,20.0)
CALL PLOTG (Y1,V1,K2,2,1,52,'TIME',4,'INVENTORY QUANTITY',18,0.0,0.0,
10.0,0.0,0.0,XL,20.0)
CALL PLOTG (Y2,X2,K3,1,1,0,'TIME',4,'INVENTORY QUANTITY',18,0.0,0.0,
10.0,0.0,0.0,XL,20.0)
CALL PLOTG (Y2,V2,K3,2,1,52,'TIME',4,'INVENTORY QUANTITY',18,0.0,0.0,
10.0,0.0,0.0,XL,20.0)
CALL PLOT (0.0,0.0,0.0,999)
STOP
DEBUG SUBCHK
END
SUBROUTINE ONE (I,TIME,EVENT,IEVENT,QU1,IG1,IQ1,IPI,IF1,XM1,PI,DSE
IED1,DSEED7,K2,NP,X1,Y1,V1,NM,NS,TOH1)
REAL*8 DSEED1,DSEED7
DIMENSION EVENT(NS), IEVENT(NS),QU1(NS),IG1(NS),X1(NP),Y1(NP),V1(NP
1),WZ(50),IK1(1),SI(1)
SS=Y1(K2)
K2=K2+1
X1(K2)=IQ1
V1(K2)=IPI
Y1(K2)=TIME
IF(IQ1.GT.0) GO TO 1
I=I+1
IG1=IQ1-IF1
IPI=IPI-IF1
QU1(I)=TIME
IG1(I)=IF1
K2=K2+1
X1(K2)=IQ1
V1(K2)=IPI
Y1(K2)=TIME
GO TO 3
1 WM=0.0
WM=(TIME-SS)*X1(K2)
TOH1=TOH1+WM
IF(IQ1.LT.IF1) GO TO 2
IG1=IQ1-IF1
IPI=IPI-IF1
K2=K2+1
X1(K2)=IQ1
V1(K2)=IPI
Y1(K2)=TIME
GO TO 3
2 I=I+1
IG1=IQ1-IF1
IPI=IPI-IF1
QU1(I)=TIME

```

```

IG1(I)=IF1-IQ1
K2=K2+1
X1(K2)=IQ1
V1(K2)=IP1
Y1(K2)=TIME
3 CALL GGEXN (DSEED1,XM1,I,S1)
IF1=1
DO 4 KI=1,NM
  II=KI
  IF(EVENT(II).EQ.99999.) GO TO 5
4 CONTINUE
  WRITE(6,100)
100 FORMAT(' EVENT LIST IS FULL')
5 EVENT(II)=TIME+S1(I)
  IEVENT(II)=1
  RETURN
  DEBUG SUBCHK
  END
SUBROUTINE TWO (J,TIME,EVENT,IEVENT,QU2,IG2,IQ2,IP2,IF2,XM2,P2,DSE
IED2,DSEED8,K3,NP,X2,Y2,V2,NM,NS,TOH2)
  REAL*8 DSEED2,DSEED8
  DIMENSION EVENT(NS),IEVENT(NS),QU2(NS),IG2(NS),X2(NP),Y2(NP),V2(NP
  SS=Y2(K3)
  K3=K3+1
  X2(K3)=IQ2
  V2(K3)=IP2
  Y2(K3)=TIME
  IF(IQ2.GT.0) GO TO 1
  J=J+1
  IG2=IQ2-IF2
  IP2=IP2-IF2
  QU2(J)=TIME
  IG2(J)=IF2
  K3=K3+1
  X2(K3)=IQ2
  V2(K3)=IP2
  Y2(K3)=TIME
  GO TO 3
1 MW=0.0
  MW=(TIME-SS)*X2(K3)
  TOH2=TOH2+MW
  IF(IQ2.LT.IF2) GO TO 2
  IP2=IP2-IF2
  K3=K3+1
  X2(K3)=IQ2

```

```

V2(K3)=IP2
Y2(K3)=TIME
GO TO 3
2 J=J+1
IQ2=IQ2-IF2
IP2=IP2-IF2
QU2(J)=TIME
IG2(J)=IF2-IQ2
K3=K3+1
X2(K3)=IQ2
V2(K3)=IP2
Y2(K3)=TIME
3 CALL GGEXN (DSEED2,XM2,1,S2)
CALL GGEXT (DSEED8,1,P2,WS,IK2)
IF2=1
DO 4 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.1 GO TO 5
4 CONTINUE
WRITE (6,100)
100 FORMAT (1, EVENT LIST IS FULL')
5 EVENT(II)=TIME+S2(1)
IEVENT(II)=2
RETURN
DEBUG SUBCHK
END
SUBROUTINE THREE (L,TIME,EVENT,IEVENT,QU3,IG3,IQ,IP,IF,XM3,P3,DSEE
1D3,DSEED6,DSEED9,A,B,K1,NP,X,Y,V,NM,NS,IA,IR,IS,IOH}
REAL*8 DSEED3,DSEED6,DSEED9
DIMENSION EVENT(NS),IEVENT(NS),QU3(NS),IG3(NS),Z(50),X(NP),Y(NP),V
1(NP),IA(NS),WK(50),IK3(1),S3(1),R(1)
SS=Y(K1)
KI=KI+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
IF(IQ.GT.0) GO TO 1
L=L+1
IP=IP-IF
IQ=IQ-IF
QU3(L)=TIME
IG3(L)=IF
KI=KI+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
GO TO 3
1 WM=0.0

```

```

NW=(TIME-SS)*X(K1)
TOH=TOH+NW
IF(IQ.LT.IF) GO TO 2
IQ=IQ-IF
IP=IP-IF
KI=KI+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
GO TO 3
2 L=L+1
  QU3(L)=TIME
  IG3(L)=IF-IQ
  IQ=IQ-IF
  IP=IP-IF
  KI=KI+1
  X(K1)=IQ
  V(K1)=IP
  Y(K1)=TIME
3 IF(IP.GT.IS) GO TO 6
  CALL GGAMS (DSEED6,A,B,1,Z,R)
  DO 4 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 5
    4 CONTINUE
    WRITE (6,100)
    100 FORMAT (1 EVENT LIST IS FULL)
    5 EVENT(II)=TIME+R(1)
    IEVENT(II)=8
    ML=0
    70 ML=ML+1
    IP=IP+IR
    IF(IP.LT.IS) GO TO 70
    IA(II)=ML*IR
    KI=KI+1
    X(K1)=IQ
    V(K1)=IP
    Y(K1)=TIME
    6 CALL GGEXN (DSEED3,XM3,1,S3)
    CALL GGEO (DSEED9,1,P3,WK,K3)
    IF=1
    DO 7 KI=1,NM
      II=KI
      IF(EVENT(II).EQ.99999.) GO TO 8
      7 CONTINUE
      8 WRITE (6,100)
      IEVENT(II)=TIME+S3(1)
      IEVENT(II)=3

```

```

RETURN SUBCHK
DEBUG SUBCHK
END
SUBROUTINE FOUR (K,IQ,IQ1,IPI,TIME,EVENT,IM,IG,QS,IA,IA1,IR
1,IR1,IS,IP,A,B,A1,B1,DSEED4,DSEED6,K1,K2,NP,X,Y,V,X1,Y1,V1,NM,T1,N
1S,IS1,TOH)
REAL*8 DSEED4,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IA1(NS),QS(NS),IG(NS),IA(NS)
1,X(NP),Y(NP),V(NP),X1(NP),Y1(NP),V1(NP)
1,DI=0
IF(IPI.GT.IS1) GO TO 1
K2=K2+1
X1(K2)=IQ1
V1(K2)=IPI
V1(K2)=TIME
ID1=IR1-IPI
IPI=IR1
K2=K2+1
X1(K2)=IQ1
V1(K2)=IPI
V1(K2)=TIME
CALL NINE (K,IQ,IDI,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA1,IR,IS,IP,A,B
1,A1,B1,DSEED4,DSEED6,K1,NP,X,Y,V,NM,NS,TOH)
1 DO 2 KI=1,NM
1 KI=KI
IF(EVENT(KI).EQ.999999.) GO TO 3
2 CONTINUE
WRITE (6,100)
100 FORMAT (' EVENT LIST IS FULL')
3 EVENT(KI)=TIME+T1
EVENT(KI)=4
RETURN
DEBUG SUBCHK
END
SUBROUTINE FIVE (K,IQ,IQ2,IP2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR
1,IR2,IS,IP,A,B,A2,B2,DSEED5,DSEED6,K1,K3,NP,X,Y,V,X2,Y2,V2,NM,T2,N
1S,IS2,TOH)
REAL*8 DSEED5,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IA2(NS),QS(NS),IG(NS),IA(NS)
1,X(NP),Y(NP),V(NP),X2(NP),Y2(NP),V2(NP)
1,DI2=0
IF(IP2.GT.IS2) GO TO 1
K3=K3+1
X2(K3)=IQ2
V2(K3)=IP2
V2(K3)=TIME
ID2=IR2-IP2
IP2=IR2

```

```

K3=K3+1
X2(K3)=IQ2
V2(K3)=IP2
Y2(K3)=TIME
CALL TEN(K,IQ,ID2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IS,IP,A,B,
IA2,B2,DSEED5,DSEED6,K1,NP,X,Y,V,NH,NS,TOH)
1 DO 2 KI=1,NM
  II=KI
  IF(EVENT(II).EQ.999999.) GO TO 3
2 CONTINUE
3 WRITE(6,100)
100 FORMAT(11 EVENT LIST IS FULL)
3 EVENT(II)=TIME+T2
  IEVENT(II)=5
  RETURN
  DEBUG SUBCHK
  END
SUBROUTINE SIX (I,IJ,TIME,IAL,IQ1,QU1,IG1,TW1,IP1,K2,NP,X1,V1,Y1,N
IM,NS)
DIMENSION IAL(NS),IG1(NS),QU1(NS),X1(NP),Y1(NP),Y1(NP)
K2=K2+1
X1(K2)=IQ1
V1(K2)=IP1
Y1(K2)=TIME
IQ1=IQ1+IAL(IJ)
K2=K2+1
X1(K2)=IQ1
V1(K2)=IP1
Y1(K2)=TIME
ITD=0
IF(1.EQ.0) GO TO 8
DO 1 N=1,I
  ITD=ITD+IG1(N)
1 CONTINUE
IF(IAL(IJ).LT.ITD) GO TO 3
DO 2 N=1,I
  W=0.
  W=(TIME-QU1(N))*IG1(N)
  QU1(N)=0.
  TW1=TW1+W
  IG1(N)=0
2 CONTINUE
  IAL(IJ)=0
  I=0
  GO TO 8
3 DO 5 N=1,I
  W=0.
  IF(IAL(IJ).GE.IG1(N)) GO TO 4

```

```

IF(IAL(IJ).EQ.0) GO TO 6
IG1(N)=IG1(N)-IA1(IJ)
W={TIME-QU1(N)}*IA1(IJ)
TW1=TW1+W
GO TO 6
4 IAL(IJ)=IAL(IJ)-IG1(N)
W={TIME-QU1(N)}*IG1(N)
TW1=TW1+W
QU1(N)=0.
IG1(N)=0
5 CONTINUE
6 MM=0
DO 7 N=1,I
IF(QU1(N).EQ.0.) GO TO 7
MM=MM+1
QU1(MM)=QU1(N)
IG1(MM)=IG1(N)
7 CONTINUE
I=MM
8 RETURN
DEBUG SUBCHK
END
SUBROUTINE SEVEN (J,IJ,TIME,IA2,IQ2,QU2,IG2,TW2,IP2,K3,NP,X2,V2,Y2
1 ,NM,NS)
DIMENSION IA2(NS),IG2(NS),QU2(NS),X2(NP),V2(NP),Y2(NP)
K3=K3+1
X2(K3)=IQ2
V2(K3)=IP2
Y2(K3)=TIME
IQ2=IQ2+IA2(IJ)
K3=K3+1
X2(K3)=IQ2
V2(K3)=IP2
Y2(K3)=TIME
ITD=0
IF(J.EQ.0) GO TO 8
DO 1 N=1,IJ
ITD=ITD+IG2(N)
1 CONTINUE
IF(IA2(IJ).LT.ITD) GO TO 3
DO 2 N=1,J
W=0.
W={TIME-QU2(N)}*IG2(N)
QU2(N)=0.
TW2=TW2+W
IG2(N)=0
2 CONTINUE
IA2(IJ)=0

```

```

J=0
GO TO 8
3 DO 5 N=1,J
  W=0.
  IF(IA2(IJ).GE.IG2(N)) GO TO 4
  IF(IA2(IJ).EQ.0) GO TO 6
  IG2(N)=IG2(N)-IA2(IJ)
  W=(TIME-QU2(N))*IA2(IJ)
  TW2=TW2+W
  GO TO 6
4 IA2(IJ)=IA2(IJ)-IG2(N)
  W=(TIME-QU2(N))*IG2(N)
  TW2=TW2+W
  QU2(N)=0.
  IG2(N)=0
5 CONTINUE
6 JJ=0
DO 7 N=1,J
  IF(QU2(N).EQ.0.) GO TO 7
  JJ=JJ+1
  QU2(JJ)=QU2(N)
  IG2(JJ)=IG2(N)
7 CONTINUE
  J=JJ
8 RETURN
  DEBUG SUBCHK
END
SUBROUTINE EIGHT (L,K,IJ,TIME,EVENT,IEVENT,IQ,IA,IA1,IA2,IG,QS,IM,
1QU3,IG3,A,B,A1,B1,A2,B2,TW3,TW4,TW5,DSEED4,DSEED5,K1,NP,X,Y,V,NM,N
1S,IP)
  REAL*8 DSEED4,DSEED5
  DIMENSION EVENT(NS), IEVENT(NS), IA(NS), IA1(NS), IA2(NS), IG(NS), QS(NS
1), IM(NS), QU3(NS), IG3(NS), Z1(50), Z2(50), X(NP), Y(NP), R1(1), R2(
1)
  ITD=0
  JZ=0
  K1=K1+1
  X(K1)=IQ
  V(K1)=IP
  Y(K1)=TIME
  IQ=IQ+IA(IJ)
  K1=K1+1
  X(K1)=IQ
  V(K1)=IP
  Y(K1)=TIME
  IF(L.EQ.0) GO TO 8
  DO 1 N=1,L
    ITD=ITD+IG3(N)

```

```

1  CONTINUE
  IF(IA(IJ).LT.ITD) GO TO 3
  DO 2 N=1,L
    W=0.
    QU3(N)=0.
    TW3=TW3+W
    IG3(N)=0
  2  CONTINUE
    IA(IJ)=IA(IJ)-ITD
    L=0
    GO TO 8
  3  DO 5 N=1,L
    W=0.
    IF(IA(IJ).GE.IG3(N)) GO TO 4
    IF(IA(IJ).EQ.0) GO TO 6
    IG3(N)=IG3(N)-IA(IJ)
    W=(TIME-QU3(N))*IA(IJ)
    TW3=TW3+W
    GO TO 6
  4  IA(IJ)=IA(IJ)-IG3(N)
    W=(TIME-QU3(N))*IG3(N)
    TW3=TW3+W
    QU3(N)=0.
    IG3(N)=0
  5  CONTINUE
  6  LL=0
    DO 7 N=1,L
    IF(QU3(N).EQ.0.) GO TO 7
    LL=LL+1
    QU3(LL)=QU3(N)
    IG3(LL)=IG3(N)
  7  CONTINUE
    L=LL
    GO TO 31
  8  IF(K.EQ.0) GO TO 31
    ITD=0
    DO 9 N=1,K
    ITD=ITD+IG(N)
  9  CONTINUE
    IF(IA(IJ).LT.ITD) GO TO 16
    IAA=0
    DO 10 N=1,K
    IF(IM(N).NE.1) GO TO 10
    W=0.
    W=(TIME-QS(N))*IG(N)
    TW4=TW4+W
    IAA=IAA+IG(N)

```

```

QS(N)=0.
IG(N)=0
IM(N)=0
10 CONTINUE
IF(IAA.EQ.0) GO TO 32
DO 11 KI=1,NM
  I=KI
  IF(EVENT(II).EQ.99999.) GO TO 12
11 CONTINUE
  WRITE (6,100)
100 FORMAT (11 EVENT LIST IS FULL)
12 CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
  EVENT(II)=TIME+R1(I)
  IEVENT(II)=6
  IAL(II)=IAA
32 IAB=0
DO 13 N=1,K
  IF(IM(N).NE.2) GO TO 13
  W=0.
  W=(TIME-QS(N))*IG(N)
  TW5=TW5+W
  IAB=IAB+IG(N)
  QS(N)=0.
  IG(N)=0
  IM(N)=0
13 CONTINUE
  IF(IAB.EQ.0) GO TO 31
DO 14 KI=1,NM
  I=KI
  IF(EVENT(II).EQ.99999.) GO TO 15
14 CONTINUE
  WRITE (6,100)
15 CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
  EVENT(II)=TIME+R2(1)
  IEVENT(II)=7
  IAZ(II)=IAB
GO TO 31
16 IAA=0
  IAB=0
DO 20 N=1,K
  IF(IM(N).NE.1) GO TO 17
  IF(IA(IJ).GE.IG(N)) GO TO 18
  IG(N)=IG(N)-IA(IJ)
  W=(TIME-QS(N))*IA(IJ)
  TW4=TW4+W
  IAA=IAA+IA(IJ)
GO TO 21
17 IF(IM(N).NE.2) GO TO 20

```

```

IF(IA(IJ).GE.IG(N)) GO TO 19
IG(N)=IG(N)-IA(IJ)
W=(TIME-QS(N))*IA(IJ)
TW5=TW5+W
IAB=IAB+IA(IJ)
GO TO 21
18 W=0.
IAA=IAA+IG(N)
IA(IJ)=IA(IJ)-IG(N)
W=(TIME-QS(N))*IG(N)
TW4=TW4+W
QS(N)=0.
IG(N)=0.
IM(N)=0.
GO TO 20
19 W=0.
IAB=IAB+IG(N)
IA(IJ)=IA(IJ)-IG(N)
W=(TIME-QS(N))*IG(N)
TW5=TW5+W
QS(N)=0.
IG(N)=0.
IM(N)=0.
CONTINUE
20 IF(IAA.EQ.0) GO TO 24
21 DO 22 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 23
CONTINUE
22 WRITE (6,100)
23 CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
EVENT(II)=TIME+R1(I)
IEVENT(II)=6
IA(II)=IAA
24 IF(IAB.EQ.0) GO TO 27
DO 25 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 26
CONTINUE
25 WRITE (6,100)
26 CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
EVENT(II)=TIME+R2(I)
IEVENT(II)=7
IA2(II)=IAB
27 NN=0
DO 30 N=1,K
IF(QS(N).EQ.0.) GO TO 30
NN=NN+1

```

```

IF(IM(N).NE.1) GO TO 28
IM(NN)=1
GO TO 29
28 IM(NN)=2
29 QS(NN)=QS(N)
IG(NN)=IG(N)
30 CONTINUE
K=NN
31 RETURN
DEBUG SUBCHK
END
SUBROUTINE NINE (K,IQ,ID1,TIME,EVENT,IEVENT,IM,IG,QS,IA,IAL,IR,IS,
1IP,A,B,AL,B1,DSEED4,DSEED6,K1,NP,X,Y,V,NM,NS,TOH)
REAL*8 DSEED4,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IAL(NS),QS(NS),IG(NS),IA(NS)
1,Z(50),Z1(50),X(NP),Y(NP),V(NP),R(1),R1(1)
SS=Y(K1)
K1=K1+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
IF(IQ.LE.0) GO TO 11
NW=0.0
NW=(TIME-SS)*X(K1)
TOH=TOH+NW
IF(IQ.GE.ID1) GO TO 6
CALL GGAMS (DSEED4,AL,B1,1,Z1,R1)
DO 1 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 2
1 CONTINUE
WRITE (6,100)
100 FORMAT (1 EVENT LIST IS FULL)
2 IEVENT(II)=6
IAL(II)=IQ
K=K+1
QS(K)=TIME
IG(K)=ID1-IQ
IM(K)=1
IQ=IQ-ID1
IP=IP-ID1
K1=K1+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
IF(IP.GT.IS) GO TO 14
DO 3 KI=1,NM

```

AD-A096 370

NAVAL POSTGRADUATE SCHOOL MONTEREY CA

F/6 15/5

A STUDY OF MULTI-ECHOLON AND MULTI-LOCATION INVENTORY SYSTEM.(U)

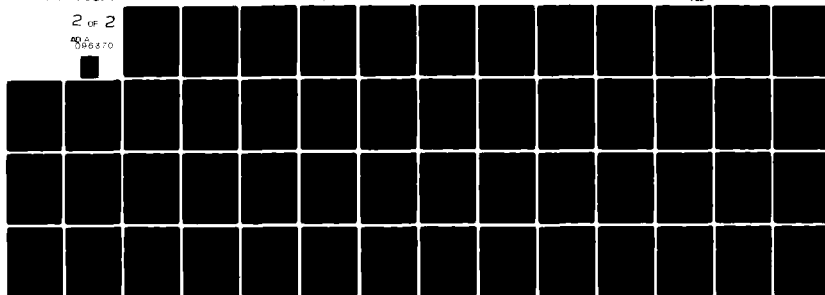
SEP 80 T BUEYUEKKARHAN

UNCLASSIFIED

NI

2 OF 2

AD-A096370



END

DATE

FILED

4-81

DTIC

```

11=KI
IF(EVENT(11).EQ.99999.) GO TO 4
3 CONTINUE
WRITE (6,100)
4 CALL GGAMS (DSEED6,A,B,1,Z,R)
EVENT(11)=TIME+R(1)
IEVENT(11)=8
ML=0
70 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 70
IA(11)=ML*IR
KI=KI+1
X(KI)=IQ
V(KI)=IP
Y(KI)=TIME
GO TO 14
6 CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
DO 7 KI=1,NM
11=KI
IF(EVENT(11).EQ.99999.) GO TO 8
7 CONTINUE
WRITE (6,100)
8 EVENT(11)=TIME+R1(1)
IA1(11)=ID1
IQ=IQ-ID1
IP=IP-ID1
KI=KI+1
X(KI)=IQ
V(KI)=IP
Y(KI)=TIME
IF(IP.GT.IS) GO TO 14
DO 9 KI=1,NM
11=KI
IF(EVENT(11).EQ.99999.) GO TO 10
9 CONTINUE
WRITE (6,100)
10 CALL GGAMS (DSEED6,A,B,1,Z,R)
EVENT(11)=TIME+R(1)
IEVENT(11)=8
ML=0
71 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 71
IA(11)=ML*IR
KI=KI+1
X(KI)=IQ

```

```

V(K1)=IP
Y(K1)=TIME
GO TO 14
11 K=K+1
QS(K)=TIME
IG(K)=ID1
IM(K)=1
IQ=IQ-ID1
IP=IP-ID1
K1=K1+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
IF(IP.GT.IS) GO TO 14
DO 12 K1=1,NM
I1=K1
IF(EVENT(I1).EQ.99999.) GO TO 13
12 CONTINUE
WRITE (6,100)
13 CALL GSAMS (DSEED6,A,8,1,Z,R)
EVENT(I1)=TIME+R(1)
IEVENT(I1)=8
ML=0
72 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 72
IA(I1)=ML*IR
K1=K1+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
14 RETURN
DEBUG SUBCHK
END
SUBROUTINE TEN (K,IQ,ID2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IS,I
IP,A,8,A2,B2,DSEED5,DSEED6,K1,NP,X,Y,V,NM,NS,TOH)
REAL*8 DSEED5,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IA2(NS),QS(NS),IG(NS),IA(NS)
1,Z(50),Z2(50),X(NP),Y(NP),V(NP),R(1),R2(1)
SS=Y(K1)
K1=K1+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
IF(IQ.LE.0) GO TO 11
WW=0.0
WW=(TIME-SS)*X(K1)
TOH=TOH+WW

```

```

IF(IQ,SE,1D2) GO TO 6
CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
DO 1 KI=1,NM
  I1=KI
  IF(EVENT(I1).EQ.99999.) GO TO 2
  1 CONTINUE
  WRITE (6,100)
  100 FORMAT (1, EVENT LIST IS FULL)
  2 EVENT(I1)=TIME+R2(1)
  IEVENT(I1)=7
  IA2(I1)=IQ
  K=K+1
  QS(K)=TIME
  IG(K)=1D2-IQ
  IM(K)=2
  IQ=IQ-1D2
  IP=IP-1D2
  KI=KI+1
  X(KI)=IQ
  V(KI)=IP
  Y(KI)=TIME
  IF(IP.GT.IS) GO TO 14
  DO 3 KI=1,NM
    I1=KI
    IF(EVENT(I1).EQ.99999.) GO TO 4
    3 CONTINUE
    WRITE (6,100)
    4 CALL GGAMS (DSEED6,A,8,1,Z,R)
    EVENT(I1)=TIME+R(1)
    IEVENT(I1)=8
    ML=0
  70 ML=ML+1
  IP=IP+IR
  IF(IP.LT.IS) GO TO 70
  IA(I1)=ML+IR
  KI=KI+1
  X(KI)=IQ
  V(KI)=IP
  Y(KI)=TIME
  GO TO 14
  6 CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
  DO 7 KI=1,NM
    I1=KI
    IF(EVENT(I1).EQ.99999.) GO TO 8
    7 CONTINUE
    WRITE (6,100)
    8 EVENT(I1)=TIME+R2(1)
    IEVENT(I1)=7

```

```

IA2(I1)=ID2
IQ=IQ-ID2
IP=IP-ID2
KI=KI+1
X(KI)=IQ
V(KI)=IP
Y(KI)=TIME
IF(IP.GT.IS) GO TO 14
DO 9 KI=1,NM
I1=KI
IF(EVENT(I1).EQ.99999.) GO TO 10
9 CONTINUE
WRITE (6,100)
10 CALL GGAMS (DSEED6,A,B,1,Z,R)
EVENT(I1)=TIME+R(1)
IEVENT(I1)=8
ML=0
71 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 71
IA(I1)=ML*IR
KI=KI+1
X(KI)=IQ
V(KI)=IP
Y(KI)=TIME
GO TO 14
11 K=K+1
QS(K)=TIME
IG(K)=ID2
IM(K)=2
IQ=IQ-ID2
IP=IP-ID2
KI=KI+1
X(KI)=IQ
V(KI)=IP
Y(KI)=TIME
IF(IP.GT.IS) GO TO 14
DO 12 KI=1,NM
I1=KI
IF(EVENT(I1).EQ.99999.) GO TO 13
12 CONTINUE
WRITE (6,100)
13 CALL GGAMS (DSEED6,A,B,1,Z,R)
EVENT(I1)=TIME+R(1)
IEVENT(I1)=8
ML=0
72 ML=ML+1
IP=IP+IR

```

```
IF(IP.LT.IS) GO TO 72
IA(I1)=ML*IR
K1=K1+1
X(K1)=IQ
V(K1)=IP
Y(K1)=TIME
14 RETURN SUBCHK
END
```

APPENDIX D

Simulation Program; No Versatec Plotter Output

```

REAL*8 DSEED1,DSEED2,DSEED3,DSEED4,DSEED5,DSEED6,DSEED7,DSEED8,DSEED9
1ED9
DIMENSION EVENT(900), IEVENT(900), IM(900), QU1(900), QU2(900), QU3(900), QU4(900), QU5(900),
1), QS(900), IG1(900), IG2(900), IG3(900), IG4(900), IG5(900), IA1(900), IA2(900), IA3(900), IA4(900), IA5(900),
1900), Z1(50), Z2(50), Z3(50), WK(50), WS(50), WZ(50), IK1(1), IK2(1), IK3(1), IK4(1), IK5(1),
1, SI(1), S2(1), S3(1), KA(10), AK(10),
NS=900
NM=30
DSEED1=123456.000
DSEED2=247658.000
DSEED3=365274.000
DSEED4=258732.000
DSEED5=541863.000
DSEED6=433215.000
DSEED7=651563.000
DSEED8=265418.000
DSEED9=167519.000
I=0
J=0
K=0
L=0
SS=0.0
SS1=0.0
SS2=0.0
TW1=0.0
TW2=0.0
TW3=0.0
TW4=0.0
TW5=0.0
TOH=0.0
TOH1=0.0
TOH2=0.0
DO 1 N=1,50
EVENT(N)=99999.
IEVENT(N)=0
QU1(N)=0.0
QU2(N)=0.0
QU3(N)=0.0
IG1(N)=0
IG2(N)=0
IG3(N)=0
IA1(N)=0
IA2(N)=0
IA(N)=0
IA(N)=0
CONTINUE
DO 2 N=1,10
IM(N)=0
QS(N)=0.0

```

```

TUR000010
TUR000020
TUR000030
TUR000040
TUR000050
TUR000060
TUR000070
TUR000080
TUR000090
TUR000100
TUR000110
TUR000120
TUR000130
TUR000140
TUR000150
TUR000160
TUR000170
TUR000180
TUR000190
TUR000200
TUR000210
TUR000220
TUR000230
TUR000240
TUR000250
TUR000260
TUR000270
TUR000280
TUR000290
TUR000300
TUR000310
TUR000320
TUR000330
TUR000340
TUR000350
TUR000360
TUR000370
TUR000380
TUR000390
TUR000400
TUR000410
TUR000420
TUR000430
TUR000440
TUR000450
TUR000460
TUR000470
TUR000480

```

TUR00490
TUR00500
TUR00510
TUR00520
TUR00530
TUR00540
TUR00550
TUR00560
TUR00570
TUR00580
TUR00590
TUR00600
TUR00610
TUR00620
TUR00630
TUR00640
TUR00650

TUR00790
TUR00800
TUR00810
TUR00820
TUR00830
TUR00840
TUR00850
TUR00860
TUR00870

TUR00890
TUR00900
TUR00910
TUR00920
TUR00930
TUR00940
TUR00950
TUR00960

```

IG(N)=0
KA(N)=0
AK(N)=0.0
2 CONTINUE
IF=0
IF1=0
IF2=0
ID=0
ID1=0
ID2=0
READ(5,100) IR,IS
100 FORMAT(2I10)
READ(5,110) IR1,IS1,T1
110 FORMAT(2I10,F10.0)
READ(5,120) IR2,IS2,T2
120 FORMAT(F10.0)
READ(5,130) A,B
130 FORMAT(2F4.0)
READ(5,140) A1,B1
READ(5,150) A2,B2
READ(5,160) XM1
READ(5,170) XM2
READ(5,180) XM3
140 FORMAT(F4.0)
READ(5,190) P1
READ(5,200) P2
READ(5,210) P3
150 FORMAT(F6.0)
IQ=IR+IS
IQ1=IR1
IQ2=IR2
IQQ=100000
IQQ1=100000
IQQ2=100000
ND11=100000
ND22=100000
XL=(YEAR/365.)*48.
TIME=0.0
IP=IR+IS
IP1=IR1
IP2=IR2
CALL GGEXN (DSEED1,XM1,1,SI)
CALL GGEXT (DSEED7,1,P1,WZ,IK1)
IF1=1
DO 3 KI=1,NM
11=KI
IF(EVENT(11).EQ.99999.) GO TO 4

```

```

3 CONTINUE
4 EVENT(I)=TIME+S1(I)
  EVENT(I)=1
  CALL GGEXN (DSEED2,XM2,1,S2)
  CALL GGEXT (DSEED8,1,P2,WS,IK2)
  IF2=1
  DO 5 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 6
  CONTINUE
5 EVENT(I)=TIME+S2(I)
  EVENT(I)=2
  CALL GGEXN (DSEED3,XM3,1,S3)
  CALL GGEXT (DSEED9,1,P3,WK,IK3)
  IF=1
  DO 7 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 8
  CONTINUE
7 EVENT(I)=TIME+S3(I)
  EVENT(I)=3
  DO 9 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 10
  CONTINUE
9 EVENT(I)=TIME+T1
  EVENT(I)=4
  DO 11 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 12
  CONTINUE
11 CONTINUE
12 EVENT(I)=TIME+T2
  EVENT(I)=5
  DO 13 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 14
  CONTINUE
13 CONTINUE
14 EVENT(I)=YEAR
  EVENT(I)=9
  GO TO 26
15 EVENT(IN)=99999.
  EVENT(IN)=0
26 IN=1
  DO 16 KI=2,NM
    IF(EVENT(IN).GT.EVENT(KI)) IN=KI
16 CONTINUE
  TIME=EVENT(IN)
  IL=IEVENT(IN)

```

```

TUR00970
TUR00980
TUR00990
TUR01000
TUR01010
TUR01020
TUR01030
TUR01040
TUR01050
TUR01060
TUR01070
TUR01080
TUR01090
TUR01100
TUR01110
TUR01120
TUR01130
TUR01140
TUR01150
TUR01160
TUR01170
TUR01180
TUR01190
TUR01200
TUR01210
TUR01220
TUR01230
TUR01240
TUR01250
TUR01260
TUR01270
TUR01280
TUR01290
TUR01300
TUR01310
TUR01320
TUR01330
TUR01340
TUR01350
TUR01360
TUR01370
TUR01380
TUR01390
TUR01400
TUR01410
TUR01420
TUR01430
TUR01440

```

```

IJ=IN
GO TO (17,18,19,20,21,22,23,24,25),IL
17 CALL ONE (I,TIME,EVENT,IEVENT,QU1,IG1,IQ1,IP1,IF1,XM1,P1,DSEED1,DS
   LEED7,NM,NS,TOH1,SS1,IQQ1)
GO TO 15
18 CALL TWO (J,TIME,EVENT,IEVENT,QU2,IG2,IQ2,IP2,IF2,XM2,P2,DSEED2,DS
   LEED8,NM,NS,TOH2,SS2,IQQ2)
GO TO 13
19 CALL THREE (L,TIME,EVENT,IEVENT,QU3,IG3,IQ3,IP3,IF3,XM3,P3,DSEED3,DS
   LEED6,DSEED9,A,B,NM,NS,IA,IR,IS,TOH,SS,KA,AK,IQQ)
GO TO 15
20 CALL FOUR (K,IQ,IQ1,IP1,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA1,IR,IR1,ITUR01560
   IS,IP,A,B,A1,B1,DSEED4,DSEED6,NM,II,NS,ISI,TOH,SS,KA,AK,IQQ,IQQ1,IQQ1,NDITUR01570
   111)
GO TO 15
21 CALL FIVE (K,IQ,IQ2,IP2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IR2,ITUR01590
   IS,IP,A,B,A2,B2,DSEED5,DSEED6,NM,I2,NS,IS2,TOH,SS,KA,AK,IQQ,IQQ2,NDITUR01600
   122)
GO TO 15
22 CALL SIX (I,IJ,TIME,IA1,IQ1,QU1,IG1,TW1,IP1,NM,NS)
GO TO 15
23 CALL SEVEN (J,IJ,TIME,IA2,IQ2,QU2,IG2,TW2,IP2,NM,NS)
GO TO 15
24 CALL EIGHT (L,K,IJ,TIME,EVENT,IEVENT,IQ,IA,IA1,IA2,IG,IS,IM,QU3,IG
   13,A,B,A1,B1,A2,B2,TW3,TW4,TW5,DSEED4,DSEED5,NM,NS,IP,KA,AK,IQQ)
GO TO 15
25 TW1=TW1/YEAR
   TW2=TW2/YEAR
   TW3=TW3/YEAR
   TW4=TW4/YEAR
   TW5=TW5/YEAR
WRITE (6,200) TW1,TW2,TW3,TW4,TW5
200 FORMAT ('TW1=',F10.4,'TW2=',F10.4,'TW3=',F10.4,'TW4=',F10.4,'
   1TW5=',F10.4)
TOH=TOH/YEAR
TOH1=TOH1/YEAR
TOH2=TOH2/YEAR
WRITE (6,201) TOH,TOH1,TOH2
201 FORMAT ('TOH=',F10.3,'TOH1=',F10.3,'TOH2=',F10.3)
STOP
DEBUG SUBCHK
END
SUBROUTINE ONE (I,TIME,EVENT,IEVENT,QU1,IG1,IQ1,IP1,IF1,XM1,P1,DSE
   LEED1,DSEED7,NM,NS,TOH1,SS1,IQQ1)
REAL*8 DSEED1,DSEED7
DIMENSION EVENT(NS),IEVENT(NS),QU1(NS),IG1(NS),WZ(50),IK1(1),SI(1)
IF(IQ1.GT.0) GO TO 1
I=I+1

```

```

100 IQ1=IQ1-IF1
101 IP1=IP1-IF1
102 IQQ1=IQQ1-IF1
103 QU1(1)=TIME
104 IG1(1)=IF1
105 GO TO 3
106 1 WW=0.0
107 WW=(TIME-SS1)*IQ1
108 TOH1=TOH1+WW
109 IF(IQ1-LI-IF1) GO TO 2
110 IQ1=IQ1-IF1
111 IP1=IP1-IF1
112 IQQ1=IQQ1-IF1
113 GO TO 3
114 2 I=I+1
115 IQ1=IQ1-IF1
116 IP1=IP1-IF1
117 IQQ1=IQQ1-IF1
118 QU1(1)=TIME
119 IG1(1)=IF1-IQ1
120 CALL GGEON (DSEED1,XM1,I,S1)
121 CALL GGEOT (DSEED7,I,P1,WZ,IK1)
122 IF1=1
123 DO 4 KI=1,NM
124 IF(KI-1) GO TO 5
125 IF(EVENT(11).EQ.99999.) GO TO 5
126 CONTINUE
127 WRITE (6,100)
128 FORMAT (1,EVENT LIST IS FULL)
129 EVENT(11)=TIME+S1(1)
130 IEVENT(11)=1
131 SS1=TIME
132 RETURN
133 DEBUG SUBCHK
134 END
135 SUBROUTINE TWJ (J,TIME,EVENT,IEVENT,QU2,IG2,IQ2,IP2,IF2,XM2,P2,DSE
136 ED2,DSEED8,NM,NS,TOH2,SS2,IQQ2)
137 REAL*8 DSEED2,DSEED8
138 DIMENSION EVENT(NS),IEVENT(NS),QU2(NS),IG2(NS),WS(50),IK2(1),S2(1)
139 IF(IQ2.GT.0) GO TO 1
140 J=J+1
141 IQ2=IQ2-IF2
142 IP2=IP2-IF2
143 IQQ2=IQQ2-IF2
144 QU2(J)=TIME
145 IG2(J)=IF2
146 GO TO 3
147 1 WW=0.0

```

```

TUR01930
TUR01940
TUR01950
TUR01960
TUR01970
TUR01980
TUR01990
TUR02000
TUR02010
TUR02020
TUR02030
TUR02040
TUR02050
TUR02060
TUR02070
TUR02080
TUR02090
TUR02100
TUR02110
TUR02120
TUR02130
TUR02140
TUR02150
TUR02160
TUR02170
TUR02180
TUR02190
TUR02200
TUR02210
TUR02220
TUR02230
TUR02240
TUR02250
TUR02260
TUR02270
TUR02280
TUR02290
TUR02300
TUR02310
TUR02320
TUR02330
TUR02340
TUR02350
TUR02360
TUR02370
TUR02380
TUR02390
TUR02400

```

```

      WW=(TIME-SS2)*IQ2
      TOH2=TOH2+WW
      IF(IQ2-LT-IF2) GO TO 2
      IQ2=IQ2-IF2
      IP2=IP2-IF2
      IQQ2=IQQ2-IF2
      GO TO 3
2     J=J+1
      IQ2=IQ2-IF2
      IP2=IP2-IF2
      IQQ2=IQQ2-IF2
      IQ2(J)=TIME
      IQ2(J)=IF2-IQ2
3     CALL GGEXN (DSEED2,XM2,1,S2)
      CALL GGEXT (DSEED8,1,P2,WS,IK2)
      IF2=1
      DO 4 KI=1,NM
      I1=KI
      IF(EVENT(I1).EQ.99999.) GO TO 5
      4 CONTINUE
      WRITE (6,100)
100  FORMAT (' EVENT LIST IS FULL.')
      EVENT(I1)=TIME+S2(1)
      I1=I1+1
      IF(I1)=2
      SS2=TIME
      RETURN
      DEBUG SUBCHK
      END
      SUBROUTINE THREE (L, TIME, EVENT, IEVENT, QU3, IG3, IQ, IP, IF, XM3, P3, DSEED9, A, B, NM, NS, IA, IR, IS, TO, SS, KA, AK, IQQ)
      REAL*8 DSEED3, DSEED6, DSEED9
      DIMENSION EVENT(NS), IEVENT(NS), QU3(NS), IG3(NS), Z(50), IA(NS), WK(50)
      1 IK3(1), S3(1), R(1), KA(10), AK(10)
      IF(IQ>1.0) GO TO 1
      L=L+1
      IP=IP-IF
      IQ=IQ-IF
      IQQ=IQQ-IF
      QU3(L)=TIME
      IG3(L)=IF
      GO TO 3
1     WW=0.0
      WW=(TIME-SS)*IQ
      TOH=TOH+WW
      IF(IQ-LT-IF) GO TO 2
      IQ=IQ-IF
      IP=IP-IF

```

```

TUR02410
TUR02420
TUR02430
TUR02440
TUR02450
TUR02460
TUR02470
TUR02480
TUR02490
TUR02500
TUR02510
TUR02520
TUR02530
TUR02540
TUR02550
TUR02560
TUR02570
TUR02580
TUR02590
TUR02600
TUR02610
TUR02620
TUR02630
TUR02640
TUR02650
TUR02660
TUR02670
TUR02680
TUR02690
TUR02700
TUR02710
TUR02720
TUR02730
TUR02740
TUR02750
TUR02760
TUR02770
TUR02780
TUR02790
TUR02800
TUR02810
TUR02820
TUR02830
TUR02840
TUR02850
TUR02860
TUR02870
TUR02880

```

```

2 GO TO 3
  L=L+1
  QU3(L)=TIME
  I3(L)=IF-IQ
  IQ=IQ-IF
  IP=IP-IF
  IF(IP.GT.IS) GO TO 6
  3 CALL GGAMS (DSEED6,A,B,1,Z,R)
    DO 4 KI=1,NM
      I=KI
      IF(EVENT(II).EQ.99999.) GO TO 5
      4 CONTINUE
        WRITE (6,100)
        100 FORMAT (11 EVENT LIST IS FULL.)
        5 EVENT(II)=TIME+R(1)
        IEVENT(II)=8
        ML=0
        70 ML=ML+1
        IP=IP+IR
        IF(IP.LT.IS) GO TO 70
        IA(II)=ML*IR
        DO 60 KT=1,10
          L5=KT
          IF(AK(L5).EQ.0.0) GO TO 61
          60 CONTINUE
            61 KA(L5)=IQ
            AK(L5)=EVENT(II)
            6 CALL GGEXN (DSEED3,XM3,1,S3)
            CALL GGEXT (DSEED9,1,P3,WK,IK3)
            IF=1
            DO 7 KI=1,NM
              II=KI
              IF(EVENT(II).EQ.99999.) GO TO 8
              7 CONTINUE
                WRITE (6,100)
                8 EVENT(II)=TIME+S3(1)
                IEVENT(II)=3
                SS=TIME
                RETURN
                DEBUG SUBCHK
                END
                SUBROUTINE FOUR (K,IQ,IQ1,IPL,TIME,EVENT,IEVENT,IM,IG,QS,IA,IAL,IR)
                  1,IR1,IS,IP,A,B,A1,B1,DSEED4,DSEED6,NM,T1,NS,ISI,TCH,SS,KA,AK,IQQ,IITUR03320
                  1QQ1,ND11)
                  REAL*8 DSEED4,DSEED6
                  DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IAL(NS),QS(NS),IG(NS),IA(NS),
                  1,KA(10),AK(10)
TUR02890
TUR02900
TUR02910
TUR02920
TUR02930
TUR02940
TUR02950
TUR02960
TUR02970
TUR02980
TUR02990
TUR03000
TUR03010
TUR03020
TUR03030
TUR03040
TUR03050
TUR03060
TUR03070
TUR03080
TUR03090
TUR03100
TUR03110
TUR03120
TUR03130
TUR03140
TUR03150
TUR03160
TUR03170
TUR03180
TUR03190
TUR03200
TUR03210
TUR03220
TUR03230
TUR03240
TUR03250
TUR03260
TUR03270
TUR03280
TUR03290
TUR03300
TUR03310
TUR03320
TUR03330
TUR03340
TUR03350
TUR03360

```

```

ND1=ND11-IQQ1
ND11=0
WRITE (6,250) ND1
250 FORMAT (/, ' AMOUNT OF DEMAND IN A PERIOD OF 1=', I10)
ID1=0
IF (IP1.GT.IS1) GO TO 1
ID1=IR1-IP1
IP1=IR1
CALL NINE (K, IQ, ID1, TIME, EVENT, IEVENT, IM, IG, QS, IA, IAL, IR, IS, IP, A, B,
1, AI, BI, DSEED4, DSEED6, NM, NS, TOH, SS, KA, AK, IQQ)
1 DO 2 KI=1, NM
I=KI
IF (EVENT(II).EQ.99999.) GO TO 3
2 CONTINUE
WRITE (6, 100)
100 FORMAT (/, ' EVENT LIST IS FULL.')
3 EVENT(II)=TIME+T1
IEVENT(II)=4
ND11=IQQ1
RETURN
DEBUG SUBCHK
END
SUBROUTINE FIVE (K, IQ, IQ2, IP2, TIME, EVENT, IEVENT, IM, IG, QS, IA, IA2, IR, IS, IP, A, B,
1, IR2, IS, IP, A, B, A2, B2, DSEED5, DSEED6, NM, T2, NS, IS2, TOH, SS, KA, AK, IQQ, I)
1, IQ2, ND22)
REAL*8 DSEED5, DSEED6
DIMENSION EVENT(NS), IEVENT(NS), IA2(NS), QS(NS), IG(NS), IA(NS)
1, KA(10), AK(10)
ND2=ND22-IQQ2
ND22=0
WRITE (6,250) ND2
250 FORMAT (/, ' AMOUNT OF DEMAND IN A PERIOD OF SYSTEM 2=', I10)
ID2=0
IF (IP2.GT.IS2) GO TO 1
ID2=IR2-IP2
IP2=IR2
CALL TEN (K, IQ, ID2, TIME, EVENT, IEVENT, IM, IG, QS, IA, IA2, IR, IS, IP, A, B,
1, A2, B2, DSEED5, DSEED6, NM, NS, TOH, SS, KA, AK, IQQ)
1 DO 2 KI=1, NM
I=KI
IF (EVENT(II).EQ.99999.) GO TO 3
2 CONTINUE
WRITE (6, 100)
100 FORMAT (/, ' EVENT LIST IS FULL.')
3 EVENT(II)=TIME+T2
IEVENT(II)=5
ND22=IQQ2
RETURN

```

TURO3370
TURO3380
TURO3390
TURO3400
TURO3410
TURO3420
TURO3430
TURO3440
TURO3450
TURO3460
TURO3470
TURO3480
TURO3490
TURO3500
TURO3510
TURO3520
TURO3530
TURO3540
TURO3550
TURO3560
TURO3570
TURO3580
TURO3590
TURO3600
TURO3610
TURO3620
TURO3630
TURO3640
TURO3650
TURO3660
TURO3670
TURO3680
TURO3690
TURO3700
TURO3710
TURO3720
TURO3730
TURO3740
TURO3750
TURO3760
TURO3770
TURO3780
TURO3790
TURO3800
TURO3810
TURO3820
TURO3830
TURO3840

```

DEBUG SUBCHK
END
SUBROUTINE SIX (I,IJ,TIME,IAL,IQI,QUL,IGL,TWL,IPL,NM,NS)
DIMENSION IAL(NS),IGI(NS),QUL(NS)
WRITE (6,50) IQI
50 FORMAT (/, ' EXPECTED NET INVENTORY OF SYSTEM 1 AT THE END OF CYCLE
1= ',I6)
IAL=IAL+IAL(IJ)
ITD=0
IF(I.EQ.0) GO TO 8
DO 1 N=1,I
ITD=ITD+IGI(N)
1 CONTINUE
IF(IAL(IJ).LT.ITD) GO TO 3
DO 2 N=1,I
W=0.
W=(TIME-QUL(N))*IGI(N)
QUL(N)=0.
TWL=TWL+W
IGI(N)=0
2 CONTINUE
IAL(IJ)=0
I=0
GO TO 8
3 DO 5 N=1,I
W=0.
IF(IAL(IJ).GE.IGI(N)) GO TO 4
IF(IAL(IJ).EQ.0) GO TO 6
IGI(N)=IGI(N)-IAL(IJ)
W=(TIME-QUL(N))*IAL(IJ)
TWL=TWL+W
GO TO 6
4 IAL(IJ)=IAL(IJ)-IGI(N)
W=(TIME-QUL(N))*IGI(N)
TWL=TWL+W
QUL(N)=0.
IGI(N)=0
5 CONTINUE
MM=0
DO 7 N=1,I
IF(QUL(N).EQ.0.) GO TO 7
MM=MM+1
QUL(MM)=QUL(N)
IGI(MM)=IGI(N)
7 CONTINUE
I=MM
8 RETURN
DEBUG SUBCHK

```

```

TUR03850
TUR03860
TUR03870
TUR03880
TUR03890
TUR03900
TUR03910
TUR03920
TUR03930
TUR03940
TUR03950
TUR03960
TUR03970
TUR03980
TUR03990
TUR04000
TUR04010
TUR04020
TUR04030
TUR04040
TUR04050
TUR04060
TUR04070
TUR04080
TUR04090
TUR04100
TUR04110
TUR04120
TUR04130
TUR04140
TUR04150
TUR04160
TUR04170
TUR04180
TUR04190
TUR04200
TUR04210
TUR04220
TUR04230
TUR04240
TUR04250
TUR04260
TUR04270
TUR04280
TUR04290
TUR04300
TUR04310
TUR04320

```

```

END
SUBROUTINE SEVEN (J,I,J,TIME,IA2,IQ2,QU2,IG2,TW2,IP2,NM,NS)
DIMENSION IA2(NS),IG2(NS),QU2(NS)
WRITE (6,50) IQ2
50 FORMAT (/, ' EXPECTED NET INVENTORY OF SYSTEM 2 AT THE END OF CYCLE',I6)
1 IQ2=IQ2+IA2(IJ)
ITD=0
IF (J.EQ.0) GO TO 8
DO 1 N=1,J
ITD=ITD+IG2(N)
1 CONTINUE
IF (IA2(IJ).LT.ITD) GO TO 3
DO 2 N=1,J
W=0.
W=(TIME-QU2(N))*IG2(N)
QU2(N)=0.
TW2=TW2+W
IG2(N)=0
2 CONTINUE
IA2(IJ)=0
J=0
GO TO 8
3 DO 5 N=1,J
W=0.
IF (IA2(IJ).GE.IG2(N)) GO TO 4
IF (IA2(IJ).EQ.0) GO TO 6
IG2(N)=IG2(N)-IA2(IJ)
W=(TIME-QU2(N))*IA2(IJ)
TW2=TW2+W
GO TO 6
4 IA2(IJ)=IA2(IJ)-IG2(N)
W=(TIME-QU2(N))*IG2(N)
TW2=TW2+W
QU2(N)=0.
IG2(N)=0
5 CONTINUE
JJ=0
DO 7 N=1,J
IF (QU2(N).EQ.0.) GO TO 7
JJ=JJ+1
QU2(JJ)=QU2(N)
IG2(JJ)=IG2(N)
7 CONTINUE
J=JJ
8 RETURN
DEBUG SUBCHK
END

```

TUR04330
 TUR04340
 TUR04350
 TUR04360
 TUR04370
 TUR04380
 TUR04390
 TUR04400
 TUR04410
 TUR04420
 TUR04430
 TUR04440
 TUR04450
 TUR04460
 TUR04470
 TUR04480
 TUR04490
 TUR04500
 TUR04510
 TUR04520
 TUR04530
 TUR04540
 TUR04550
 TUR04560
 TUR04570
 TUR04580
 TUR04590
 TUR04600
 TUR04610
 TUR04620
 TUR04630
 TUR04640
 TUR04650
 TUR04660
 TUR04670
 TUR04680
 TUR04690
 TUR04700
 TUR04710
 TUR04720
 TUR04730
 TUR04740
 TUR04750
 TUR04760
 TUR04770
 TUR04780
 TUR04790
 TUR04800

```

SUBROUTINE EIGHT (L,K,IJ,TIME,EVENT,IEVENT,IQ,IA,IA1,IA2,IG,QS,IM,
1 QU3,IG3,A,B,A1,B1,A2,B2,TW3,TW4,TW5,DSEED3,NM,NS,IP,KA,AK,
1 QQ)
REAL*8 DSEED4,DSEED5
DIMENSION EVENT(NS),IEVENT(NS),IA(NS),IA1(NS),IA2(NS),IG(NS),QS(NS),
1 IM(NS),QU3(NS),IG3(NS),Z1(50),Z2(50),R1(1),R2(1),KA(10),AK(10),
1 ITD=0
JZ=0
JZ=IQ
DO 60 KT=1,10
L5=KT
IF(AK(L5).EQ.TIME) GO TO 61
60 CONTINUE
61 ND=KA(L5)-IQ
KA(L5)=0
AK(L5)=0.
WRITE (6,250) ND,TIME
250 FORMAT (/, ' DEMAND DURING LEAD TIME=',I10,' TIME=',F15.5)
WRITE (6,50) JZ,IQ,TIME
50 FORMAT (/, ' EXPECTED NET INVENTORY OF MAIN SYSTEM AT THE END OF
1 CYCLE=',I6,2X,' IQ=',I6,2X,' TIME=',F10.4)
IF(L.EQ.0) GO TO 8
DO 1 N=1,L
1 ITD=ITD+IG3(N)
1 CONTINUE
IF(IA(IJ).LT.ITD) GO TO 3
DO 2 N=1,L
W=0.
W=(TIME-QU3(N))*IG3(N)
QU3(N)=0.
TW3=TW3+W
IG3(N)=0
2 CONTINUE
IA(IJ)=IA(IJ)-ITD
L=0
GO TO 8
3 DO 5 N=1,L
W=0.
IF(IA(IJ).GE.IG3(N)) GO TO 4
IF(IA(IJ).EQ.0) GO TO 6
IG3(N)=IG3(N)-IA(IJ)
W=(TIME-QU3(N))*IA(IJ)
TW3=TW3+W
GO TO 6
4 IA(IJ)=IA(IJ)-IG3(N)
W=(TIME-QU3(N))*IG3(N)
TW3=TW3+W

```

TURA04810
TURA04820
TURA04830
TURA04840
TURA04850
TURA04860
TURA04870
TURA04880
TURA04890
TURA04900
TURA04910
TURA04920
TURA04930
TURA04940
TURA04950
TURA04960
TURA04970
TURA04980
TURA04990
TURA05000
TURA05010
TURA05020
TURA05030
TURA05040
TURA05050
TURA05060
TURA05070
TURA05080
TURA05090
TURA05100
TURA05110
TURA05120
TURA05130
TURA05140
TURA05150
TURA05160
TURA05170
TURA05180
TURA05190
TURA05200
TURA05210
TURA05220
TURA05230
TURA05240
TURA05250
TURA05260
TURA05270
TURA05280

TUR05290
TUR05300
TUR05310
TUR05320
TUR05330
TUR05340
TUR05350
TUR05360
TUR05370
TUR05380
TUR05390
TUR05400
TUR05410
TUR05420
TUR05430
TUR05440
TUR05450
TUR05460
TUR05470
TUR05480
TUR05490
TUR05500
TUR05510
TUR05520
TUR05530
TUR05540
TUR05550
TUR05560
TUR05570
TUR05580
TUR05590
TUR05600
TUR05610
TUR05620
TUR05630
TUR05640
TUR05650
TUR05660
TUR05670
TUR05680
TUR05690
TUR05700
TUR05710
TUR05720
TUR05730
TUR05740
TUR05750
TUR05760

```

QU3(N)=0.
IG3(N)=0
5 CONTINUE
6 LL=0
DO 7 N=1,L
  IF(QU3(N).EQ.0.) GO TO 7
  LL=LL+1
  QU3(LL)=QU3(N)
  IG3(LL)=IG3(N)
7 CONTINUE
  L=LL
  GO TO 31
8 IF(K.EQ.0) GO TO 31
  ITD=0
DO 9 N=1,K
  ITD=ITD+IG(N)
9 CONTINUE
  IF(IA(IJ).LT.ITD) GO TO 16
  IAA=0
DO 10 N=1,K
  IF(IM(N).NE.1) GO TO 10
  W=0
  W=(TIME-QS(N))*IG(N)
  TW4=TW4+W
  IAA=IAA+IG(N)
  QS(N)=0.
  IG(N)=0
  IM(N)=0
10 CONTINUE
  I=(IAA.EQ.0) GO TO 32
  DO 11 KI=1,NM
  I1=KI
  IF(EVENT(I1).EQ.99999.) GO TO 12
  11 CONTINUE
  WRITE (6,100)
100 FORMAT (10 EVENT LIST IS FULL)
12 CALL GSAMS (DSEED4,A1,B1,I,Z1,R1)
  EVENT(I1)=TIME+R1(1)
  I1=KI
  IAL(I1)=IAA
32 IAB=0
DO 13 N=1,K
  IF(IM(N).NE.2) GO TO 13
  W=0.
  W=(TIME-QS(N))*IG(N)
  TW5=TW5+W
  IAB=IAB+IG(N)
  QS(N)=0.

```

TUR05770
 TUR05780
 TUR05790
 TUR05800
 TUR05810
 TUR05820
 TUR05830
 TUR05840
 TUR05850
 TUR05860
 TUR05870
 TUR05880
 TUR05890
 TUR05900
 TUR05910
 TUR05920
 TUR05930
 TUR05940
 TUR05950
 TUR05960
 TUR05970
 TUR05980
 TUR05990
 TUR06000
 TUR06010
 TUR06020
 TUR06030
 TUR06040
 TUR06050
 TUR06060
 TUR06070
 TUR06080
 TUR06090
 TUR06100
 TUR06110
 TUR06120
 TUR06130
 TUR06140
 TUR06150
 TUR06160
 TUR06170
 TUR06180
 TUR06190
 TUR06200
 TUR06210
 TUR06220
 TUR06230
 TUR06240

```

IG(N)=0
IM(N)=0
13 CONTINUE
  IF(IAB.EQ.0) GO TO 31
  DO 14 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 15
  14 CONTINUE
  WRITE (6,100)
  15 CALL GGAMS (DSEED5,A2,82,1,Z2,R2)
  EVENT(II)=TIME+R2(1)
  IEVENT(II)=7
  IA2(II)=IAB
  GO TO 31
  16 IAA=0
  IAB=0
  DO 20 N=1,K
    IF(IM(N).NE.1) GO TO 17
    IF(IA(IJ).GE.IG(N)) GO TO 18
    IG(N)=IG(N)-IA(IJ)
    W=(TIME-QS(N))*IA(IJ)
    TW4=TW4+W
    IAA=IAA+IA(IJ)
    GO TO 21
  17 IF(IM(N).NE.2) GO TO 20
    IF(IA(IJ).GE.IG(N)) GO TO 19
    IG(N)=IG(N)-IA(IJ)
    W=(TIME-QS(N))*IA(IJ)
    TW5=TW5+W
    IAB=IAB+IA(IJ)
    GO TO 21
  18 W=0.
    IAA=IAA+IG(N)
    IA(IJ)=IA(IJ)-IG(N)
    W=(TIME-QS(N))*IG(N)
    TW4=TW4+W
    QS(N)=0.
    IG(N)=0
    IM(N)=0
    GO TO 20
  19 W=0.
    IAB=IAB+IG(N)
    IA(IJ)=IA(IJ)-IG(N)
    W=(TIME-QS(N))*IG(N)
    TW5=TW5+W
    QS(N)=0.
    IG(N)=0
    IM(N)=0

```

```

20 CONTINUE
21 IF(IAA.EQ.0) GO TO 24
   DO 22 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 23
22 CONTINUE
   WRITE (6,100)
23 CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
   EVENT(II)=TIME+R1(1)
   IEVENT(II)=6
   IAI(II)=IAA
24 IF(IAB.EQ.0) GO TO 27
   DO 25 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 26
25 CONTINUE
   WRITE (6,100)
26 CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
   EVENT(II)=TIME+R2(1)
   IEVENT(II)=7
   IA2(II)=IAB
27 NN=0
   DO 30 N=1,K
   IF(QS(N).EQ.0.) GO TO 30
   NN=NN+1
   IF(IM(N).NE.1) GO TO 28
   IM(NN)=1
   GO TO 29
28 IM(NN)=2
29 QS(NN)=QS(N)
   IG(NN)=IG(N)
30 CONTINUE
   K=NN
31 RETURN
   END
   SUBROUTINE MIVE (K,IQ,IO1,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA1,IR,IS,
   IIP,A,B,A1,B1,DSEED4,DSEED6,NM,NS,TOH,SS,KA,AK,IQQ)
   REAL*8 DSEED4,DSEED6
   DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IA1(NS),QS(NS),IG(NS),IA(NS),
   IZ(50),ZI(50),R(1),RI(1),KA(10),AK(10)
   IO=0
   IF(IQ.LE.0) GO TO 11
   NM=0
   WW=(TIME-SS)*IQ
   TOH=TOH+WW
   IF(IQ.GE.IO1) GO TO 6
   CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)

```

```

TUR06250
TUR06260
TUR06270
TUR06280
TUR06290
TUR06300
TUR06310
TUR06320
TUR06330
TUR06340
TUR06350
TUR06360
TUR06370
TUR06380
TUR06390
TUR06400
TUR06410
TUR06420
TUR06430
TUR06440
TUR06450
TUR06460
TUR06470
TUR06480
TUR06490
TUR06500
TUR06510
TUR06520
TUR06530
TUR06540
TUR06550
TUR06560
TUR06570
TUR06580
TUR06590
TUR06600
TUR06610
TUR06620
TUR06630
TUR06640
TUR06650
TUR06660
TUR06670
TUR06680
TUR06690
TUR06700
TUR06710
TUR06720

```

```

DO 1 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 2
1 CONTINUE
WRITE (6,100)
100 FORMAT (1,EVENT LIST IS FULL*)
2 EVENT(II)=TIME+R1(1)
IAI(II)=6
IAI(II)=IQ
K=K+1
QS(K)=TIME
IG(K)=ID1-IQ
IM(K)=1
IQ=IQ-ID1
IP=IP-ID1
IQ=IQ-ID1
IF(IP.GT.IS) GO TO 14
DO 3 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 4
3 CONTINUE
WRITE (6,100)
4 CALL GGAMS (DSEED6,A,B,1,Z,R)
EVENT(II)=TIME+R(1)
EVENT(II)=8
ML=0
70 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 70
IAI(II)=ML+IR
DO 60 KT=1,10
15=KT
IF(AK(L5).EQ.0.) GO TO 61
60 CONTINUE
61 KAI(L5)=IQ
AK(L5)=EVENT(II)
GO TO 14
6 CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
DO 7 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 8
7 CONTINUE
WRITE (6,100)
8 EVENT(II)=TIME+R1(1)
EVENT(II)=6
IAI(II)=ID1
IQ=IQ-ID1
IP=IP-ID1

```

```

TUR06730
TUR06740
TUR06750
TUR06760
TUR06770
TUR06780
TUR06790
TUR06800
TUR06810
TUR06820
TUR06830
TUR06840
TUR06850
TUR06860
TUR06870
TUR06880
TUR06890
TUR06900
TUR06910
TUR06920
TUR06930
TUR06940
TUR06950
TUR06960
TUR06970
TUR06980
TUR06990
TUR07000
TUR07010
TUR07020
TUR07030
TUR07040
TUR07050
TUR07060
TUR07070
TUR07080
TUR07090
TUR07100
TUR07110
TUR07120
TUR07130
TUR07140
TUR07150
TUR07160
TUR07170
TUR07180
TUR07190
TUR07200

```

```

100=IQQ-ID1
IF(IP.GT.IS) GO TO 14
DO 9 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 10
9 CONTINUE
WRITE (6,100)
10 CALL GGAMS (DSEED6,A,8,1,Z,R)
EVENT(II)=TIME+R(1)
IEVENT(II)=8
ML=0
71 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 71
IA(II)=ML*IR
DO 63 KT=1,10
L5=KT
IF(AK(L5).EQ.0.0) GO TO 64
63 CONTINUE
64 KA(L5)=IQQ
AK(L5)=EVENT(II)
GO TO 14
11 K=K+1
QS(K)=TIME
IG(K)=ID1
IM(K)=1
IQ=IQ-ID1
IP=IP-ID1
100=IQQ-ID1
IF(IP.GT.IS) GO TO 14
DO 12 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 13
12 CONTINUE
WRITE (6,100)
13 CALL GGAMS (DSEED6,A,8,1,Z,R)
EVENT(II)=TIME+R(1)
IEVENT(II)=8
ML=0
72 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 72
IA(II)=ML*IR
SS=TIME
14 RETURN
DEBUG SUBCHK
END
SUBROUTINE TEN (K,IQ,ID2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IS,ITUR07210
TUR07220
TUR07230
TUR07240
TUR07250
TUR07260
TUR07270
TUR07280
TUR07290
TUR07300
TUR07310
TUR07320
TUR07330
TUR07340
TUR07350
TUR07360
TUR07370
TUR07380
TUR07390
TUR07400
TUR07410
TUR07420
TUR07430
TUR07440
TUR07450
TUR07460
TUR07470
TUR07480
TUR07490
TUR07500
TUR07510
TUR07520
TUR07530
TUR07540
TUR07550
TUR07560
TUR07570
TUR07580
TUR07590
TUR07600
TUR07610
TUR07620
TUR07630
TUR07640
TUR07650
TUR07660
TUR07670
ITUR07680

```

```

1P,A,B,A2,B2,DSEED5,DSEED6,NM,NS,TOH,SS,KA,AK,IQQ)
REAL*8 DSEED5,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IA2(NS),QS(NS),IG(NS),IA(NS)
1,Z(50),Z2(50),R(1),R2(1),KA(10),AK(10)
ID=0
IF(IQ.LE.0) G3 TO 11
WM=0
WM=(TIME-SS)*IQ
TOH=TOH+WM
IF(IQ.GE.ID2) GO TO 6
CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
DO 1 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 2
1 CONTINUE
WRITE (6,100)
100 FORMAT (1,EVENT LIST IS FULL,')
2 EVENT(II)=TIME+R2(1)
IEVENT(II)=7
IA2(II)=IQ
K=K+1
QS(K)=TIME
IG(K)=ID2-IQ
IM(K)=2
IQ=IQ-ID2
IP=IP-ID2
IQQ=IQQ-ID2
IF(IP.GT.IS) GO TO 14
DO 3 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 4
3 CONTINUE
WRITE (6,100)
4 CALL GGAMS (DSEED6,A,B,1,Z,R)
IEVENT(II)=8
ML=0
70 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 70
IA(II)=ML+IR
DO 60 KT=1,10
L5=KT
IF (AK(L5).EQ.0.0) GO TO 61
60 CONTINUE
61 KA(L5)=IQQ
AK(L5)=EVENT(II)
GO TO 14

```

```

TUR07690
TUR07700
TUR07710
TUR07720
TUR07730
TUR07740
TUR07750
TUR07760
TUR07770
TUR07780
TUR07790
TUR07800
TUR07810
TUR07820
TUR07830
TUR07840
TUR07850
TUR07860
TUR07870
TUR07880
TUR07890
TUR07900
TUR07910
TUR07920
TUR07930
TUR07940
TUR07950
TUR07960
TUR07970
TUR07980
TUR07990
TUR08000
TUR08010
TUR08020
TUR08030
TUR08040
TUR08050
TUR08060
TUR08070
TUR08080
TUR08090
TUR08100
TUR08110
TUR08120
TUR08130
TUR08140
TUR08150
TUR08160

```

```

6 CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
DO 7 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 8
7 CONTINUE
WRITE (6,100)
8 EVENT(II)=TIME+R2(1)
IA2(II)=ID2
IQ=IQ-ID2
IP=IP-ID2
IQ=IQ-ID2
IP=IP-ID2
IF(IP.GT.IS) GO TO 14
DO 9 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 10
9 CONTINUE
WRITE (6,100)
10 CALL GGAMS (DSEED6,A,B,1,Z,R)
EVENT(II)=TIME+R(1)
IEVENT(II)=8
ML=0
71 ML=ML+1
IP=IP+IR
IF(IP.LT.IS) GO TO 71
IA(II)=ML*IR
DO 63 KT=1,10
L5=KT
IF(AK(L5).EQ.0.0) GO TO 64
63 CONTINUE
64 KA(L5)=IQ
AK(L5)=EVENT(II)
GO TO 14
11 K=K+1
QS(K)=TIME
IG(K)=ID2
IM(K)=2
IQ=IQ-ID2
IQ=IQ-ID2
IP=IP-ID2
IF(IP.GT.IS) GO TO 14
DO 12 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 13
12 CONTINUE
13 CALL GGAMS (DSEED6,A,B,1,Z,R)
EVENT(II)=TIME+R(1)

```

```

TUR08170
TUR08180
TUR08190
TUR08200
TUR08210
TUR08220
TUR08230
TUR08240
TUR08250
TUR08260
TUR08270
TUR08280
TUR08290
TUR08300
TUR08310
TUR08320
TUR08330
TUR08340
TUR08350
TUR08360
TUR08370
TUR08380
TUR08390
TUR08400
TUR08410
TUR08420
TUR08430
TUR08440
TUR08450
TUR08460
TUR08470
TUR08480
TUR08490
TUR08500
TUR08510
TUR08520
TUR08530
TUR08540
TUR08550
TUR08560
TUR08570
TUR08580
TUR08590
TUR08600
TUR08610
TUR08620
TUR08630
TUR08640

```

TUR08650
TUR08660
TUR08670
TUR08680
TUR08690
TUR08700
TUR08710
TUR08720
TUR08730
TUR08740

IEVENT(11)=8
ML=0
72 ML=ML+1
IF(IP+IR
IF(IP.LT.15) GO TO 72
14 IA(11)=ML*IR
SS=TIME
RETURN
DEBUG SUBCHK
END

Early Warning Simulation Program

```

REAL*8  DSEED1,DSEED2,DSEED3,DSEED4,DSEED5,DSEED6,DSEED7,DSEED8,DSEED9,DSEED10
1ED9
DIMENSION EVENT(900), IEVENT(900), IM(900), QU1(900), QU2(900), QU3(900), QU4(900), QU5(900),
1), QS(900), IG1(900), IG2(900), IG3(900), IG4(900), IG5(900), IA1(900), IA2(900), IA3(900), IA4(900),
1900), Z1(50), Z2(50), Z3(50), WK(50), WS(50), IK1(1), IK2(1), IK3(1), IK4(1), IK5(1), IK6(1), IK7(1),
1, SI(1), S2(1), S3(1), KA(10), AK(10)
NS=900
NM=30
DSEED1=123456.0D0
DSEED2=247658.0D0
DSEED3=365274.0D0
DSEED4=258732.0D0
DSEED5=541863.0D0
DSEED6=433215.0D0
DSEED7=651563.0D0
DSEED8=265418.0D0
DSEED9=167519.0D0
I=0
J=0
K=0
L=0
SS=0.0
SS1=0.0
SS2=0.0
TW1=0.0
TW2=0.0
TW3=0.0
TW4=0.0
TW5=0.0
TOH=0.0
TOH1=0.0
TOH2=0.0
DO 1 N=1,50
EVENT(N)=99999.
IEVENT(N)=0
QU1(N)=0.0
QU2(N)=0.0
QU3(N)=0.0
IG1(N)=0
IG2(N)=0
IG3(N)=0
IA1(N)=0
IA2(N)=0
IA3(N)=0
IA4(N)=0
IA5(N)=0
CONTINUE
DO 2 N=1,10
IM(N)=0
Q1(N)=0.0
Q2(N)=0.0
Q3(N)=0.0
Q4(N)=0.0
Q5(N)=0.0
Q6(N)=0.0
Q7(N)=0.0
Q8(N)=0.0
Q9(N)=0.0
Q10(N)=0.0
Q11(N)=0.0
Q12(N)=0.0
Q13(N)=0.0
Q14(N)=0.0
Q15(N)=0.0
Q16(N)=0.0
Q17(N)=0.0
Q18(N)=0.0
Q19(N)=0.0
Q20(N)=0.0
Q21(N)=0.0
Q22(N)=0.0
Q23(N)=0.0
Q24(N)=0.0
Q25(N)=0.0
Q26(N)=0.0
Q27(N)=0.0
Q28(N)=0.0
Q29(N)=0.0
Q30(N)=0.0
Q31(N)=0.0
Q32(N)=0.0
Q33(N)=0.0
Q34(N)=0.0
Q35(N)=0.0
Q36(N)=0.0
Q37(N)=0.0
Q38(N)=0.0
Q39(N)=0.0
Q40(N)=0.0
Q41(N)=0.0
Q42(N)=0.0
Q43(N)=0.0
Q44(N)=0.0
Q45(N)=0.0
Q46(N)=0.0
Q47(N)=0.0
Q48(N)=0.0
Q49(N)=0.0
Q50(N)=0.0
Q51(N)=0.0
Q52(N)=0.0
Q53(N)=0.0
Q54(N)=0.0
Q55(N)=0.0
Q56(N)=0.0
Q57(N)=0.0
Q58(N)=0.0
Q59(N)=0.0
Q60(N)=0.0
Q61(N)=0.0
Q62(N)=0.0
Q63(N)=0.0
Q64(N)=0.0
Q65(N)=0.0
Q66(N)=0.0
Q67(N)=0.0
Q68(N)=0.0
Q69(N)=0.0
Q70(N)=0.0
Q71(N)=0.0
Q72(N)=0.0
Q73(N)=0.0
Q74(N)=0.0
Q75(N)=0.0
Q76(N)=0.0
Q77(N)=0.0
Q78(N)=0.0
Q79(N)=0.0
Q80(N)=0.0
Q81(N)=0.0
Q82(N)=0.0
Q83(N)=0.0
Q84(N)=0.0
Q85(N)=0.0
Q86(N)=0.0
Q87(N)=0.0
Q88(N)=0.0
Q89(N)=0.0
Q90(N)=0.0
Q91(N)=0.0
Q92(N)=0.0
Q93(N)=0.0
Q94(N)=0.0
Q95(N)=0.0
Q96(N)=0.0
Q97(N)=0.0
Q98(N)=0.0
Q99(N)=0.0
Q100(N)=0.0
Q101(N)=0.0
Q102(N)=0.0
Q103(N)=0.0
Q104(N)=0.0
Q105(N)=0.0
Q106(N)=0.0
Q107(N)=0.0
Q108(N)=0.0
Q109(N)=0.0
Q110(N)=0.0
Q111(N)=0.0
Q112(N)=0.0
Q113(N)=0.0
Q114(N)=0.0
Q115(N)=0.0
Q116(N)=0.0
Q117(N)=0.0
Q118(N)=0.0
Q119(N)=0.0
Q120(N)=0.0
Q121(N)=0.0
Q122(N)=0.0
Q123(N)=0.0
Q124(N)=0.0
Q125(N)=0.0
Q126(N)=0.0
Q127(N)=0.0
Q128(N)=0.0
Q129(N)=0.0
Q130(N)=0.0
Q131(N)=0.0
Q132(N)=0.0
Q133(N)=0.0
Q134(N)=0.0
Q135(N)=0.0
Q136(N)=0.0
Q137(N)=0.0
Q138(N)=0.0
Q139(N)=0.0
Q140(N)=0.0
Q141(N)=0.0
Q142(N)=0.0
Q143(N)=0.0
Q144(N)=0.0
Q145(N)=0.0
Q146(N)=0.0
Q147(N)=0.0
Q148(N)=0.0
Q149(N)=0.0
Q150(N)=0.0
Q151(N)=0.0
Q152(N)=0.0
Q153(N)=0.0
Q154(N)=0.0
Q155(N)=0.0
Q156(N)=0.0
Q157(N)=0.0
Q158(N)=0.0
Q159(N)=0.0
Q160(N)=0.0
Q161(N)=0.0
Q162(N)=0.0
Q163(N)=0.0
Q164(N)=0.0
Q165(N)=0.0
Q166(N)=0.0
Q167(N)=0.0
Q168(N)=0.0
Q169(N)=0.0
Q170(N)=0.0
Q171(N)=0.0
Q172(N)=0.0
Q173(N)=0.0
Q174(N)=0.0
Q175(N)=0.0
Q176(N)=0.0
Q177(N)=0.0
Q178(N)=0.0
Q179(N)=0.0
Q180(N)=0.0
Q181(N)=0.0
Q182(N)=0.0
Q183(N)=0.0
Q184(N)=0.0
Q185(N)=0.0
Q186(N)=0.0
Q187(N)=0.0
Q188(N)=0.0
Q189(N)=0.0
Q190(N)=0.0
Q191(N)=0.0
Q192(N)=0.0
Q193(N)=0.0
Q194(N)=0.0
Q195(N)=0.0
Q196(N)=0.0
Q197(N)=0.0
Q198(N)=0.0
Q199(N)=0.0
Q200(N)=0.0
Q201(N)=0.0
Q202(N)=0.0
Q203(N)=0.0
Q204(N)=0.0
Q205(N)=0.0
Q206(N)=0.0
Q207(N)=0.0
Q208(N)=0.0
Q209(N)=0.0
Q210(N)=0.0
Q211(N)=0.0
Q212(N)=0.0
Q213(N)=0.0
Q214(N)=0.0
Q215(N)=0.0
Q216(N)=0.0
Q217(N)=0.0
Q218(N)=0.0
Q219(N)=0.0
Q220(N)=0.0
Q221(N)=0.0
Q222(N)=0.0
Q223(N)=0.0
Q224(N)=0.0
Q225(N)=0.0
Q226(N)=0.0
Q227(N)=0.0
Q228(N)=0.0
Q229(N)=0.0
Q230(N)=0.0
Q231(N)=0.0
Q232(N)=0.0
Q233(N)=0.0
Q234(N)=0.0
Q235(N)=0.0
Q236(N)=0.0
Q237(N)=0.0
Q238(N)=0.0
Q239(N)=0.0
Q240(N)=0.0
Q241(N)=0.0
Q242(N)=0.0
Q243(N)=0.0
Q244(N)=0.0
Q245(N)=0.0
Q246(N)=0.0
Q247(N)=0.0
Q248(N)=0.0
Q249(N)=0.0
Q250(N)=0.0
Q251(N)=0.0
Q252(N)=0.0
Q253(N)=0.0
Q254(N)=0.0
Q255(N)=0.0
Q256(N)=0.0
Q257(N)=0.0
Q258(N)=0.0
Q259(N)=0.0
Q260(N)=0.0
Q261(N)=0.0
Q262(N)=0.0
Q263(N)=0.0
Q264(N)=0.0
Q265(N)=0.0
Q266(N)=0.0
Q267(N)=0.0
Q268(N)=0.0
Q269(N)=0.0
Q270(N)=0.0
Q271(N)=0.0
Q272(N)=0.0
Q273(N)=0.0
Q274(N)=0.0
Q275(N)=0.0
Q276(N)=0.0
Q277(N)=0.0
Q278(N)=0.0
Q279(N)=0.0
Q280(N)=0.0
Q281(N)=0.0
Q282(N)=0.0
Q283(N)=0.0
Q284(N)=0.0
Q285(N)=0.0
Q286(N)=0.0
Q287(N)=0.0
Q288(N)=0.0
Q289(N)=0.0
Q290(N)=0.0
Q291(N)=0.0
Q292(N)=0.0
Q293(N)=0.0
Q294(N)=0.0
Q295(N)=0.0
Q296(N)=0.0
Q297(N)=0.0
Q298(N)=0.0
Q299(N)=0.0
Q300(N)=0.0
Q301(N)=0.0
Q302(N)=0.0
Q303(N)=0.0
Q304(N)=0.0
Q305(N)=0.0
Q306(N)=0.0
Q307(N)=0.0
Q308(N)=0.0
Q309(N)=0.0
Q310(N)=0.0
Q311(N)=0.0
Q312(N)=0.0
Q313(N)=0.0
Q314(N)=0.0
Q315(N)=0.0
Q316(N)=0.0
Q317(N)=0.0
Q318(N)=0.0
Q319(N)=0.0
Q320(N)=0.0
Q321(N)=0.0
Q322(N)=0.0
Q323(N)=0.0
Q324(N)=0.0
Q325(N)=0.0
Q326(N)=0.0
Q327(N)=0.0
Q328(N)=0.0
Q329(N)=0.0
Q330(N)=0.0
Q331(N)=0.0
Q332(N)=0.0
Q333(N)=0.0
Q334(N)=0.0
Q335(N)=0.0
Q336(N)=0.0
Q337(N)=0.0
Q338(N)=0.0
Q339(N)=0.0
Q340(N)=0.0
Q341(N)=0.0
Q342(N)=0.0
Q343(N)=0.0
Q344(N)=0.0
Q345(N)=0.0
Q346(N)=0.0
Q347(N)=0.0
Q348(N)=0.0
Q349(N)=0.0
Q350(N)=0.0
Q351(N)=0.0
Q352(N)=0.0
Q353(N)=0.0
Q354(N)=0.0
Q355(N)=0.0
Q356(N)=0.0
Q
```

TUR00490
TUR00500
TUR00510
TUR00520
TUR00530
TUR00540
TUR00550
TUR00560
TUR00570
TUR00580
TUR00590
TUR00600
TUR00610
TUR00620
TUR00630
TUR00640
TUR00650

TUR00790
TUR00800
TUR00810
TUR00820
TUR00830
TUR00840
TUR00850
TUR00860
TUR00870

TUR00900
TUR00910
TUR00920
TUR00930
TUR00940
TUR00950
TUR00960

```

IG(N)=0
KA(N)=0
AK(N)=0.0
2 CONTINUE
IF=0
IF1=0
IF2=0
ID=0
ID1=0
ID2=0
READ(5,100) IR,IS
100 FORMAT(2I10) IRL,IS1,T1
110 READ(5,110) F10.0)
110 FORMAT(2I10,F10.0)
120 READ(5,120) YEAR
120 FORMAT(F10.0)
130 READ(5,130) A,B
130 FORMAT(2F4.0)
140 READ(5,140) A1,B1
140 READ(5,140) A2,B2
140 READ(5,140) XM1
140 READ(5,140) XM2
140 READ(5,140) XM3
140 FORMAT(F4.0) P1
140 READ(5,150) P2
140 READ(5,150) P3
150 READ(5,150)
150 FORMAT(F6.0)
IQ=IR+IS
IQ1=IR1
IQ2=IR2
IQQ=100000
IQQ1=100000
IQQ2=100000
ND11=100000
ND22=100000
XL=(YEAR/365.)*48.
TIME=0.0
IP=IR+IS
IPP=IR+IS
IP1=IR1
IP2=IR2
CALL GGEXN (DSEED1,XM1,1,S1)
CALL GGEXT (DSEED7,1,PI,W2,IK1)
IF1=1
DO 3 KI=1,NM
II=KI

```

```

3  IF(EVENT(II).EQ.99999.) GO TO 4
4  CONTINUE
   EVENT(II)=TIME+S1(II)
   IEVENT(II)=1
   CALL GGEXN (DSEED2,XM2,1,S2)
   CALL GGEXT (DSEED8,1,P2,WS,IK2)
   IF2=1
   DO 5 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 6
5  CONTINUE
6  EVENT(II)=TIME+S2(II)
   IEVENT(II)=2
   CALL GGEXN (DSEED3,XM3,1,S3)
   CALL GGEXT (DSEED9,1,P3,WK,IK3)
   IF=1
   DO 7 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 8
7  CONTINUE
8  EVENT(II)=TIME+S3(II)
   IEVENT(II)=3
   DO 9 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 10
9  CONTINUE
10 EVENT(II)=TIME+T1
   IEVENT(II)=4
   DO 11 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 12
11 CONTINUE
12 EVENT(II)=TIME+T2
   IEVENT(II)=5
   DO 13 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 14
13 CONTINUE
14 EVENT(II)=YEAR
   IEVENT(II)=9
   GO TO 26
15 EVENT(IN)=99999.
   IEVENT(IN)=0
26 IN=I
   DO 16 KI=2,NM
   IF(EVENT(IN).GT.EVENT(KI)) IN=KI
16 CONTINUE
   TIME=EVENT(IN)

```

```

TUR00970
TUR00980
TUR00990
TUR01000
TUR01010
TUR01020
TUR01030
TUR01040
TUR01050
TUR01060
TUR01070
TUR01080
TUR01090
TUR01100
TUR01110
TUR01120
TUR01130
TUR01140
TUR01150
TUR01160
TUR01170
TUR01180
TUR01190
TUR01200
TUR01210
TUR01220
TUR01230
TUR01240
TUR01250
TUR01260
TUR01270
TUR01280
TUR01290
TUR01300
TUR01310
TUR01320
TUR01330
TUR01340
TUR01350
TUR01360
TUR01370
TUR01380
TUR01390
TUR01400
TUR01410
TUR01420
TUR01430
TUR01440

```

```

IL=IEVENT(IN)
IJ=IN
GO TO (17,18,19,20,21,22,23,24,25),IJ
17 CALL ONE (I,TIME,EVENT,IEVENT,QU1,IG1,IQ1,IP1,IF1,XM1,P1,DSEED1,DS
1EED7,NM,NS,TOH1,SS1,IQQ1,IPP,IS,A,B,IR,DSEED6,IA,KA,AK,IQQ)
GO TO 15
18 CALL TWO (J,TIME,EVENT,IEVENT,QU2,IG2,IQ2,IP2,IF2,XM2,P2,DSEED2,DS
1EED8,NM,NS,TOH2,SS2,IQQ2,IPP,IS,A,B,IR,DSEED6,IA,KA,AK,IQQ)
GO TO 15
19 CALL THREE (L,TIME,EVENT,IEVENT,QU3,IG3,IQ3,IP3,IF3,XM3,P3,DSEED3,DS
1EED9,A,B,NM,NS,IA,IR,IS,TOH,SS,KA,AK,IQQ,IPP)
GO TO 15
20 CALL FOUR (K,IQ,IQ1,IPI,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA1,IR,IR1,
IS,IP,A,B,A1,B1,DSEED4,DSEED6,NM,I1,NS,IS1,TOH,SS,KA,AK,IQQ,IQQ1,NDI)
GO TO 15
21 CALL FIVE (K,IQ,IQ2,IP2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IR2,
IS,IP,A,B,A2,B2,DSEED5,DSEED6,NM,I2,NS,IS2,TOH,SS,KA,AK,IQQ,IQQ2,NDI)
GO TO 15
22 CALL SIX (I,IJ,TIME,IAL,IQ1,QU1,IG1,TW1,IPI,NM,NS)
GO TO 15
23 CALL SEVEN (J,IJ,TIME,IA2,IQ2,QU2,IG2,TW2,IP2,NM,NS)
GO TO 15
24 CALL EIGHT (L,K,IJ,TIME,EVENT,IEVENT,IQ,IA,IA1,IA2,IG,QS,IM,QU3,IG
13,A,B,A1,B1,A2,B2,TW3,TW4,TW5,DSEED4,DSEED5,NM,NS,IP,KA,AK,IQQ)
GO TO 15
25 TW1=TW1/YEAR
TW2=TW2/YEAR
TW3=TW3/YEAR
TW4=TW4/YEAR
TW5=TW5/YEAR
WRITE (6,200) TW1,TW2,TW3,TW4,TW5
200 FORMAT (1,TW1=,F10.4, TW2=,F10.4, TW3=,F10.4, TW4=,F10.4,
1TW5=,F10.4)
TOH=TOH/YEAR
TOH1=TOH1/YEAR
TOH2=TOH2/YEAR
WRITE (6,201) TOH,TOH1,TOH2
201 FORMAT (1,TOH=,F10.3, TOH1=,F10.3, TOH2=,F10.3)
STOP
DEBUG SUBCHK
END
SUBROUTINE ONE (I,TIME,EVENT,IEVENT,QU1,IG1,IQ1,IP1,IF1,XM1,P1,DSE
1EDI,DSEED7,NM,NS,TOH1,SS1,IQQ1,IPP,IS,A,B,IR,DSEED6,IA,KA,AK,IQQ)
REAL*8 DSEED1,DSEED7,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),QU1(NS),IG1(NS),WZ(50),IK1(1),SI(1)
1,Z(50),R(1),IA(NS),KA(10),AK(10)
TUR01450
TUR01460
TUR01470
TUR01480
TUR01490
TUR01500
TUR01510
TUR01520
TUR01530
TUR01540
TUR01550
TUR01560
TUR01570
TUR01580
TUR01590
TUR01600
TUR01610
TUR01620
TUR01630
TUR01640
TUR01650
TUR01660
TUR01670
TUR01680
TUR01690
TUR01700
TUR01710
TUR01720
TUR01730
TUR01740
TUR01750
TUR01760
TUR01770
TUR01780
TUR01790
TUR01800
TUR01810
TUR01820
TUR01830
TUR01840
TUR01850
TUR01860
TUR01870
TUR01880
TUR01890
TUR01900
TUR01910
TUR01920

```

```

IPP=IPP-IF1
IQQ=IQQ-IF1
IF(IPP.GT.IS) GO TO 6
CALL GGAMS (DSEED6,A,B,1,Z,R)
DO 7 KI=1,NM
  IF=KI
  IF(EVENT(II).EQ.99999.) GO TO 8
  7 CONTINUE
  WRITE (6,100)
  8 EVENT(II)=TIME+R(1)
  IEVENT(II)=8
  ML=0
  70 ML=ML+1
  IPP=IPP+IR
  IF(IPP.LT.IS) GO TO 70
  IA(II)=ML*IR
  DO 60 KT=1,10
    LS=KT
    IF(AK(LS).EQ.3.0) GO TO 61
    60 CONTINUE
    KALLS)=IQQ
    61 AK(LS)=EVENT(II)
    6 IF(IQL.GT.0) GO TO 1
    I=I+1
    IQ1=IQL-IF1
    IPI=IPI-IF1
    IQQ1=IQQ1-IF1
    QUI(1)=TIME
    IGL(1)=IF1
    GO TO 3
    1 MW=0.0
    MW=(TIME-SS1)*IQL
    TOML=TOHL+MW
    IF(IQL.LT.IF1) GO TO 2
    IQ1=IQL-IF1
    IPI=IPI-IF1
    IQQ1=IQQ1-IF1
    GU TO 3
    2 I=I+1
    IQ1=IQL-IF1
    IPI=IPI-IF1
    IQQ1=IQQ1-IF1
    QUI(1)=TIME
    IGL(1)=IF1-IF1
    3 CALL GGEN (DSEED7,M1,1,S1)
    IF1=1
    DO 4 KI=1,NM

```

```

TUR01930
TUR01940
TUR01950
TUR01960
TUR01970
TUR01980
TUR01990
TUR02000
TUR02010
TUR02020
TUR02030
TUR02040
TUR02050
TUR02060
TUR02070
TUR02080
TUR02090
TUR02100
TUR02110
TUR02120
TUR02130
TUR02140
TUR02150
TUR02160
TUR02170
TUR02180
TUR02190
TUR02200
TUR02210
TUR02220
TUR02230
TUR02240
TUR02250
TUR02260
TUR02270
TUR02280
TUR02290
TUR02300
TUR02310
TUR02320
TUR02330
TUR02340
TUR02350
TUR02360
TUR02370
TUR02380
TUR02390
TUR02400

```

```

11=KI
IF(EVENT(11).EQ.99999.) GO TO 5
4 CONTINUE
WRITE (6,100)
100 FORMAT (' EVENT LIST IS FULL')
5 EVENT(11)=TIME+SL(1)
EVENT(11)=1
SSL=TIME
RETURN
DEBUG SUBCHK
END
SUBROUTINE TWO (J, TIME, EVENT, IEVENT, QU2, IG2, IQ2, IP2, IF2, XM2, P2, DSEED6, IA, KA, AK, IQQ)
REAL*8 DSEED2, DSEED8, DSEED6
DIMENSION EVENT(NS), IEVENT(NS), QU2(NS), IG2(NS), WS(50), IK2(1), S2(1)
1, 2(50), R(1), IA(NS), KA(10), AK(19)
1 IPP=IPP-IF2
IQQ=IQQ-IF2
IF(IPP.GT.15) GO TO 6
CALL GGAMS (DSEED6,A,B,1,Z,R)
DO 7 KI=1,NM
11=KI
IF(EVENT(11).EQ.99999.) GO TO 8
7 CONTINUE
8 WRITE (6,100)
EVENT(11)=TIME+R(1)
IEVENT(11)=8
ML=0
ML=ML+1
IPP=IPP+IR
IF(IPP.LT.15) GO TO 70
IA(11)=ML*IR
DO 60 KT=1,10
L5=KT
IF(AK(L5).EQ.0.0) GO TO 61
60 CONTINUE
61 KA(L5)=IQQ
AK(L5)=EVENT(11)
6 IF(IQ2.GT.0) GO TO 1
J=J+1
IQ2=IQ2-IF2
IP2=IP2-IF2
IQQ=IQQ-IF2
QU2(J)=TIME
IG2(J)=IF2
GO TO 3
1 WM=0.0
WM=(TIME-SS2)*IQ2

```

```

TUR02410
TUR02420
TUR02430
TUR02440
TUR02450
TUR02460
TUR02470
TUR02480
TUR02490
TUR02500
TUR02510
TUR02520
TUR02530
TUR02540
TUR02550
TUR02560
TUR02570
TUR02580
TUR02590
TUR02600
TUR02610
TUR02620
TUR02630
TUR02640
TUR02650
TUR02660
TUR02670
TUR02680
TUR02690
TUR02700
TUR02710
TUR02720
TUR02730
TUR02740
TUR02750
TUR02760
TUR02770
TUR02780
TUR02790
TUR02800
TUR02810
TUR02820
TUR02830
TUR02840
TUR02850
TUR02860
TUR02870
TUR02880

```

```

TOH2=TOH2+WW
IF(IQ2.LT.IF2) GO TO 2
IC2=IQ2-IF2
IP2=IP2-IF2
IQ2=IQ2-IF2
GO TO 3
2 J=J+1
IP2=IP2-IF2
IQ2=IQ2-IF2
QU2(J)=TIME
IQ2(J)=IF2-IQ2
3 CALL GGEXN (DSEED2,XM2,1,S2)
CALL GGEXT (DSEED8,1,P2,WS,IK2)
IF2=1
DO 4 KI=1,NM
II=KI
IF(EVENT(II).EQ.999999.) GO TO 5
CONTINUE
4 WRITE (6,100)
100 FORMAT (10 EVENT LIST IS FULL')
5 EVENT(II)=TIME+S2(1)
EVENT(II)=2
SS2=TIME
RETURN
DEBUG SUBCHK
END
SUBROUTINE THREE (L,TIME,EVENT,IEVENT,QU3,IG3,IQ,IP,IF,XM3,P3,DSEED9,
103,DSEED6,DSEED9,A,B,NM,NS,IA,IR,IS,TOH,SS,KA,AK,IQQ,IPP)
REAL*8 DSEED3,DSEED6,DSEED9
DIMENSION EVENT(NS),IEVENT(NS),QU3(NS),IG3(NS),Z(50),IA(NS),WK(50)
1 IK3(1),S3(1),R(1),KA(10),AK(10)
IPP=IPP-IF
IF(IQ.GT.0) GO TO 1
L=L+1
IP=IP-IF
IQ=IQ-IF
IQQ=IQ2-IF
QU3(L)=TIME
IG3(L)=IF
GO TO 3
1 WW=0.0
WH=(TIME-SS)*IQ
TOH=TOH+WW
IF(IQ.LT.IF) GO TO 2
IQ=IQ-IF
IQQ=IQQ-IF
IP=IP-IF

```

```

TUR02890
TUR02900
TUR02910
TUR02920
TUR02930
TUR02940
TUR02950
TUR02960
TUR02970
TUR02980
TUR02990
TUR03000
TUR03010
TUR03020
TUR03030
TUR03040
TUR03050
TUR03060
TUR03070
TUR03080
TUR03090
TUR03100
TUR03110
TUR03120
TUR03130
TUR03140
TUR03150
TUR03160
TUR03170
TUR03180
TUR03190
TUR03200
TUR03210
TUR03220
TUR03230
TUR03240
TUR03250
TUR03260
TUR03270
TUR03280
TUR03290
TUR03300
TUR03310
TUR03320
TUR03330
TUR03340
TUR03350
TUR03360

```

```

GO TO 3
2 L=L+1
  QU3(L)=TIME
  IG3(L)=IF-IQ
  IQ=IQ-IF
  IP=IP-IF
  IQ=IQ-IF
  IF(IPP.GT.IS) GO TO 6
3 CALL GGAMS (DSEED6,A,B,1,Z,R)
  DO 4 KI=1,NM
  II=KI
  IF(EVENT(II).EQ.99999.) GO TO 5
4 CONTINUE
  WRITE (6,100)
100 FORMAT (' EVENT LIST IS FULL')
5 EVENT(II)=TIME+R(1)
  IEVENT(II)=8
  ML=0
70 ML=ML+1
  IPP=IPP+IR
  IF(IPP.LT.IS) GO TO 70
  IA(II)=ML*IR
  DO 60 KT=1,10
  L5=KT
  IF(AK(L5).EQ.0.0) GO TO 61
60 CONTINUE
61 KA(L5)=IQ
  AK(L5)=EVENT(II)
  IPP=IR+IS
6 CALL GGEXN (DSEED3,XM3,1,S3)
  IF=1
  DO 7 KI=1,NM
  II=KI
  IF(EVENT(II).EQ.99999.) GO TO 8
7 CONTINUE
  WRITE (6,100)
8 EVENT(II)=TIME+S3(1)
  SS=TIME
  RETURN
  DEBUG SUBCHK
  END
  SUBROUTINE FOUR (K,IQ,IQL,IPL,TIME,EVENT,IEVENT,IM,IG,QS,IA,IAL,IR,TUR03370
1,IR1,IS,IP,A,B,AL,B1,DSEED4,DSEED6,NM,I1,NS,ISI,TOH,SS,KA,AK,IQQ,I
1QQ1,ND1)
  REAL*8 DSEED4,DSEED6
  DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IAL(NS),QS(NS),IG(NS),IA(NS)TUR03380
TUR03380
TUR03380
TUR03390
TUR03390
TUR03400
TUR03410
TUR03420
TUR03430
TUR03440
TUR03450
TUR03460
TUR03470
TUR03480
TUR03490
TUR03500
TUR03510
TUR03520
TUR03530
TUR03540
TUR03550
TUR03560
TUR03570
TUR03580
TUR03590
TUR03600
TUR03610
TUR03620
TUR03630
TUR03640
TUR03650
TUR03660
TUR03670
TUR03680
TUR03690
TUR03700
TUR03710
TUR03720
TUR03730
TUR03740
TUR03750
TUR03760
TUR03770
TUR03780
TUR03790
TUR03800
TUR03810
TUR03820
TUR03830
TUR03840

```

```

1 KA(10),AK(10)
ND1=ND1-IQQ1
ND11=0
WRITE (6,250) ND1
250 FORMAT (/, ' AMOUNT OF DEMAND IN A PERIOD OF 1=',I10)
ID1=0
IF (IP1.GT.IS1) GO TO 1
ID1=IR1-IP1
IP1=IR1
CALL NINE (K,IQ,ID1,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA1,IR,IS,IP,A,B,
1,AL,B1,DSEED4,DSEED6,NM,NS,TOH,SS,KA,AK,IQQ}
1 DO 2 KI=1,NM
I1=KI
IF (EVENT(I1).EQ.99999.) GO TO 3
2 CONTINUE
WRITE (6,100)
100 FORMAT (/, ' EVENT LIST IS FULL')
3 EVENT(I1)=TIME+T1
IEVENT(I1)=4
ND11=IQQ1
RETURN
DEBUG SUBCHK
END
SUBROUTINE FIVE (K,IQ,IQ2,IP2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,
1,IR2,IS,IP,A,B,A2,B2,DSEED5,DSEED6,NM,T2,NS,IS2,TOH,SS,KA,AK,IQQ,
1,QQ2,ND22)
REAL*8 DSEED5,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),IA2(NS),QS(NS),IG(NS),IA(NS)
1 KA(10),AK(10)
ND2=ND22-IQQ2
ND22=0
WRITE (6,250) ND2
250 FORMAT (/, ' AMOUNT OF DEMAND IN A PERIOD OF SYSTEM 2=',I10)
ID2=0
IF (IP2.GT.IS2) GO TO 1
ID2=IR2-IP2
IP2=IR2
CALL TEN (K,IQ,ID2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IS,IP,A,B,
1A2,B2,DSEED5,DSEED6,NM,NS,TOH,SS,KA,AK,IQQ}
1 DO 2 KI=1,NM
I1=KI
IF (EVENT(I1).EQ.99999.) GO TO 3
2 CONTINUE
WRITE (6,100)
100 FORMAT (/, ' EVENT LIST IS FULL')
3 EVENT(I1)=TIME+T2
IEVENT(I1)=5
ND22=IQQ2

```

TUR03850
 TUR03860
 TUR03870
 TUR03880
 TUR03890
 TUR03900
 TUR03910
 TUR03920
 TUR03930
 TUR03940
 TUR03950
 TUR03960
 TUR03970
 TUR03980
 TUR03990
 TUR04000
 TUR04010
 TUR04020
 TUR04030
 TUR04040
 TUR04050
 TUR04060
 TUR04070
 TUR04080
 TUR04090
 TUR04100
 TUR04110
 TUR04120
 TUR04130
 TUR04140
 TUR04150
 TUR04160
 TUR04170
 TUR04180
 TUR04190
 TUR04200
 TUR04210
 TUR04220
 TUR04230
 TUR04240
 TUR04250
 TUR04260
 TUR04270
 TUR04280
 TUR04290
 TUR04300
 TUR04310
 TUR04320

```

RETURN
DEBUG SUBCHK
END
SUBROUTINE SIX (I,IJ,TIME,IAl,IQI,QUl,IPl,NM,NS)
DIMENSION IAl(NS),IGl(NS),QUl(NS)
WRITE (6,50) IQI
50 FORMAT (/, ' EXPECTED NET INVENTORY OF SYSTEM 1 AT THE END OF CYCLE
1= ',I6)
IQI=IQI+IAl(IJ)
ITD=0
IF (I.EQ.0) GO TO 8
DO 1 N=1,I
ITD=ITD+IGl(N)
1 CONTINUE
IF (IAl(IJ).LT.ITD) GO TO 3
DO 2 N=1,I
W=0.
W=(TIME-QUl(N))*IGl(N)
QUl(N)=0.
TWI=TWI+W
IGl(N)=0
2 CONTINUE
IAl(IJ)=0
I=0
GO TO 8
3 DO 5 N=1,I
W=0.
IF (IAl(IJ).GE.IGl(N)) GO TO 4
IF (IAl(IJ).EQ.0) GO TO 6
IGl(N)=IGl(N)-IAl(IJ)
W=(TIME-QUl(N))*IAl(IJ)
TWI=TWI+W
GO TO 6
4 IAl(IJ)=IAl(IJ)-IGl(N)
W=(TIME-QUl(N))*IGl(N)
TWI=TWI+W
QUl(N)=0.
IGl(N)=0
5 CONTINUE
6 MM=0
DO 7 N=1,I
IF (QUl(N).EQ.0.) GO TO 7
MM=MM+1
QUl(MM)=QUl(N)
IGl(MM)=IGl(N)
7 CONTINUE
I=MM
8 RETURN

```

TUR04330
TUR04340
TUR04350
TUR04360
TUR04370
TUR04380
TUR04390
TUR04400
TUR04410
TUR04420
TUR04430
TUR04440
TUR04450
TUR04460
TUR04470
TUR04480
TUR04490
TUR04500
TUR04510
TUR04520
TUR04530
TUR04540
TUR04550
TUR04560
TUR04570
TUR04580
TUR04590
TUR04600
TUR04610
TUR04620
TUR04630
TUR04640
TUR04650
TUR04660
TUR04670
TUR04680
TUR04690
TUR04700
TUR04710
TUR04720
TUR04730
TUR04740
TUR04750
TUR04760
TUR04770
TUR04780
TUR04790
TUR04800

```

DEBUG SUBCHK
END
SUBROUTINE SEVEN (J,I,J,TIME,IA2,IQ2,QU2,IG2,TW2,IP2,NM,NS)
DIMENSION IA2(NS),IG2(NS),QU2(NS)
WRITE (6,50) IQ2
50 FORMAT (/, ' EXPECTED NET INVENTORY OF SYSTEM 2 AT THE END OF CYCLE',
1, ' ', I6)
IQ2=IQ2+IA2(IJ)
ITD=0
IF (J.EQ.0) GO TO 8
DO 1 N=1,J
ITD=ITD+IG2(N)
1 CONTINUE
IF (IA2(IJ).LT.ITD) GO TO 3
DU 2 N=1,J
W=0
W=(TIME-QU2(N))*IG2(N)
QU2(N)=0
TW2=TW2+W
IG2(N)=0
2 CONTINUE
IA2(IJ)=0
J=0
GO TO 8
3 DO 5 N=1,J
W=0
IF (IA2(IJ).GE.IG2(N)) GO TO 4
IF (IA2(IJ).EQ.0) GO TO 6
IG2(N)=IG2(N)-IA2(IJ)
W=(TIME-QU2(N))*IA2(IJ)
TW2=TW2+W
GO TO 6
4 IA2(IJ)=IA2(IJ)-IG2(N)
W=(TIME-QU2(N))*IG2(N)
TW2=TW2+W
QU2(N)=0
IG2(N)=0
5 CONTINUE
JJ=0
DO 7 N=1,J
IF (QU2(N).EQ.0.) GO TO 7
JJ=JJ+1
QU2(JJ)=QU2(N)
IG2(JJ)=IG2(N)
7 CONTINUE
J=JJ
8 RETURN
DEBUG SUBCHK

```

```

TUR04810
TUR04820
TUR04830
TUR04840
TUR04850
TUR04860
TUR04870
TUR04880
TUR04890
TUR04900
TUR04910
TUR04920
TUR04930
TUR04940
TUR04950
TUR04960
TUR04970
TUR04980
TUR04990
TUR05000
TUR05010
TUR05020
TUR05030
TUR05040
TUR05050
TUR05060
TUR05070
TUR05080
TUR05090
TUR05100
TUR05110
TUR05120
TUR05130
TUR05140
TUR05150
TUR05160
TUR05170
TUR05180
TUR05190
TUR05200
TUR05210
TUR05220
TUR05230
TUR05240
TUR05250
TUR05260
TUR05270
TUR05280

```

```

END
SUBROUTINE EIGHT (L,K,IJ,TIME,EVENT,IEVENT,IQ,IA,IA1,IA2,IG,QS,IM,
1 IQ3,IG3,A,B,A1,B1,A2,B2,TW3,TW4,TW5,DSEED4,DSEED5,NM,NS,IP,KA,AK,
1 IQQ)
REAL *8 DSEED4,DSEED5
DIMENSION EVENT(NS),IEVENT(NS),IA(NS),IA1(NS),IA2(NS),IG(NS),QS(NS),
1 IM(NS),QU3(NS),IG3(NS),Z1(50),Z2(50),R1(1),R2(1),KA(10),AK(10)
ITD=0
JZ=0
JZ=IQ
DO 60 KT=1,10
L5=KT
IF(AK(L5).EQ.TIME) GO TO 61
60 CONTINUE
61 ND=KA(L5)-IQQ
KA(L5)=0
AK(L5)=0
WRITE (6,250) ND,TIME
FORMAT (/, ' DEMAND DURING LEAD TIME=',I10, ' TIME=',F15.5)
250 IQ=IQ+IA(IJ)
JZ=IQ,TIME
WRITE (6,50)
50 FORMAT (/, ' EXPECTED NET INVENTORY OF MAIN SYSTEM AT THE END OF CYCLE=',I6,2X, ' IQ=',I6,2X, ' TIME=',F10.4)
1 CLE=,I6,2X, ' GO TO 8
IF(L.EQ.0) GO TO 8
DO 1 N=1,L
ITD=ITD+IG3(N)
1 CONTINUE
IF(IA(IJ).LT.ITD) GO TO 3
DO 2 N=1,L
W=0
W=(TIME-QU3(N))*IG3(N)
QU3(N)=0
TW3=TW3+W
IG3(N)=0
2 CONTINUE
IA(IJ)=IA(IJ)-ITD
L=0
GO TO 8
DO 5 N=1,L
W=0
IF(IA(IJ)-GE-IG3(N)) GO TO 4
IF(IA(IJ).EQ.0) GO TO 6
IG3(N)=IG3(N)-IA(IJ)
W=(TIME-QU3(N))*IA(IJ)
TW3=TW3+W
GO TO 5
4 IA(IJ)=IA(IJ)-IG3(N)
W=(TIME-QU3(N))*IG3(N)

```

TURO5290
TURO5300
TURO5310
TURO5320
TURO5330
TURO5340
TURO5350
TURO5360
TURO5370
TURO5380
TURO5390
TURO5400
TURO5410
TURO5420
TURO5430
TURO5440
TURO5450
TURO5460
TURO5470
TURO5480
TURO5490
TURO5500
TURO5510
TURO5520
TURO5530
TURO5540
TURO5550
TURO5560
TURO5570
TURO5580
TURO5590
TURO5600
TURO5610
TURO5620
TURO5630
TURO5640
TURO5650
TURO5660
TURO5670
TURO5680
TURO5690
TURO5700
TURO5710
TURO5720
TURO5730
TURO5740
TURO5750
TURO5760

TUR05770
 TUR05780
 TUR05790
 TUR05800
 TUR05810
 TUR05820
 TUR05830
 TUR05840
 TUR05850
 TUR05860
 TUR05870
 TUR05880
 TUR05890
 TUR05900
 TUR05910
 TUR05920
 TUR05930
 TUR05940
 TUR05950
 TUR05960
 TUR05970
 TUR05980
 TUR05990
 TUR06000
 TUR06010
 TUR06020
 TUR06030
 TUR06040
 TUR06050
 TUR06060
 TUR06070
 TUR06080
 TUR06090
 TUR06100
 TUR06110
 TUR06120
 TUR06130
 TUR06140
 TUR06150
 TUR06160
 TUR06170
 TUR06180
 TUR06190
 TUR06200
 TUR06210
 TUR06220
 TUR06230
 TUR06240

```

TW3=TW3+W
QU3(N)=0.
IG3(N)=0
5 CONTINUE
6 LL=0
DO 7 N=1,L
  IF(QU3(N).EQ.0.) GO TO 7
  LL=LL+1
  QU3(LL)=QU3(N)
  IG3(LL)=IG3(N)
7 CONTINUE
  L=LL
  GO TO 31
8 IF(K.EQ.0) GO TO 31
  ITD=0
DO 9 N=1,K
  ITD=ITD+IG(N)
9 CONTINUE
  IF(IA(IJ).LT.ITD) GO TO 16
  IAA=0
DO 10 N=1,K
  IF(IM(N).NE.1) GO TO 10
  W=0
  W=(TIME-QS(N))*IG(N)
  TW4=TW4+W
  IAA=IAA+IG(N)
  QS(N)=0.
  IG(N)=0
  IM(N)=0
10 CONTINUE
  IF(IAA.EQ.0) GO TO 32
  DO 11 XI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 12
  11 CONTINUE
    WRITE (6,100)
  100 FORMAT (' EVENT LIST IS FULL')
  12 CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
    EVENT(II)=TIME+R1(1)
    IEVENT(II)=6
    IAI(II)=IAA
  32 IAB=0
DO 13 N=1,K
  IF(IM(N).NE.2) GO TO 13
  W=0
  W=(TIME-QS(N))*IG(N)
  TW5=TW5+W
  IAB=IAB+IG(N)

```

TUR06250
TUR06260
TUR06270
TUR06280
TUR06290
TUR06300
TUR06310
TUR06320
TUR06330
TUR06340
TUR06350
TUR06360
TUR06370
TUR06380
TUR06390
TUR06400
TUR06410
TUR06420
TUR06430
TUR06440
TUR06450
TUR06460
TUR06470
TUR06480
TUR06490
TUR06500
TUR06510
TUR06520
TUR06530
TUR06540
TUR06550
TUR06560
TUR06570
TUR06580
TUR06590
TUR06600
TUR06610
TUR06620
TUR06630
TUR06640
TUR06650
TUR06660
TUR06670
TUR06680
TUR06690
TUR06700
TUR06710
TUR06720

```

13  QS(N)=0.
    IG(N)=0.
    IM(N)=0.
    IF(IAB.EQ.0) GO TO 31
    DO 14 KI=1,NM
    I=KI
    IF(EVENT(II).EQ.99999.) GO TO 15
14  CONTINUE
    WRITE (6,100)
15  CALL GGAMS (DSEED5,A2,B2,1,22,R2)
    EVENT(II)=TIME+R2(1)
    IF(EVENT(II))=7
    I2(II)=IAB
    GO TO 31
16  IAA=0
    IAB=0
    DO 20 N=1,K
    IF(IM(N).NE.1) GO TO 17
    IF(IA(IJ).GE.IG(N)) GO TO 18
    IG(N)=IG(N)-IA(IJ)
    W=(TIME-QS(N))*IA(IJ)
    TW4=TW4+W
    IAA=IAA+IA(IJ)
    GO TO 21
17  IF(IM(N).NE.2) GO TO 20
    IF(IA(IJ).GE.IG(N)) GO TO 19
    IG(N)=IG(N)-IA(IJ)
    W=(TIME-QS(N))*IA(IJ)
    TW5=TW5+W
    IAB=IAB+IA(IJ)
    GO TO 21
18  W=0.
    IAA=IAA+IG(N)
    IA(IJ)=IA(IJ)-IG(N)
    W=(TIME-QS(N))*IG(N)
    TW4=TW4+W
    QS(N)=0.
    IG(N)=0.
    IM(N)=0.
    GO TO 20
19  W=0.
    IAB=IAB+IG(N)
    IA(IJ)=IA(IJ)-IG(N)
    W=(TIME-QS(N))*IG(N)
    TW5=TW5+W
    QS(N)=0.
    IG(N)=0.

```

```

IM(N)=0
20 CONTINUE
21 IF(IAA.EQ.0) GO TO 24
   DO 22 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 23
22 CONTINUE
   WRITE(6,100)
23 CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
   EVENT(II)=TIME+R1(1)
   EVENT(II)=6
   IAL(II)=IAA
24 IF(IAB.EQ.0) GO TO 27
   DO 25 KI=1,NM
   II=KI
   IF(EVENT(II).EQ.99999.) GO TO 26
25 CONTINUE
   WRITE(6,100)
26 CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
   EVENT(II)=TIME+R2(1)
   IAZ(II)=IAB
27 NN=0
   DO 30 N=1,K
   IF(QS(N).EQ.0.) GO TO 30
   NN=NN+1
   IF(IM(N).NE.1) GO TO 28
   IM(NN)=1
   GO TO 29
28 IM(NN)=2
29 QS(NN)=QS(N)
   IG(NN)=IG(N)
30 CONTINUE
   K=NN
31 RETURN
   END
   SUBROUTINE NIVE (K,IQ,IDL,TIME,EVENT,IEVENT,IM,IG,QS,IA,IAL,IR,IS,
   IIP,A,B,A1,B1,DSEED4,DSEED6,NM,NS,TOH,SS,KA,AK,IQQ)
   REAL*8 DSEED4,DSEED6
   DIMENSION EVENT(NS),IEVENT(NS),IA(NS),IG(NS),IA(NS),
   IZ(50),ZI(50),R(1),RI(1),KA(10),AK(10)
   ID=0
   IF(IQ.LE.0) GO TO 11
   MW=0.0
   MW=(TIME-SS)*IQ
   TOH=TOH+MW
   IF(IQ.GE.ID1) GO TO 6

```

```

TUR06730
TUR06740
TUR06750
TUR06760
TUR06770
TUR06780
TUR06790
TUR06800
TUR06810
TUR06820
TUR06830
TUR06840
TUR06850
TUR06860
TUR06870
TUR06880
TUR06890
TUR06900
TUR06910
TUR06920
TUR06930
TUR06940
TUR06950
TUR06960
TUR06970
TUR06980
TUR06990
TUR07000
TUR07010
TUR07020
TUR07030
TUR07040
TUR07050
TUR07060
TUR07070
TUR07080
TUR07090
TUR07100
TUR07110
TUR07120
TUR07130
TUR07140
TUR07150
TUR07160
TUR07170
TUR07180
TUR07190
TUR07200

```

```

CALL GGAMS (DSEED4,A1,B1,1,Z1,R1)
DO 1 KI=1,NM
  II=KI
  IF(EVENT(II).EQ.99999.) GO TO 2
  1 CONTINUE
  WRITE (6,100)
  100 FORMAT (1,EVENT LIST IS FULL)
  2 EVENT(II)=TIME+R1(1)
  IF(EVENT(II)=6
    IAL(II)=IQ
    K=K+1
    QS(K)=TIME
    IG(K)=ID1-IQ
    IM(K)=1
    IQ=IQ-ID1
    IP=IP-ID1
    IF(IP.GT.IS) GO TO 14
    DO 60 KT=1,10
      L5=KT
      IF(AK(L5).EQ.0.) GO TO 61
      60 CONTINUE
      61 KA(L5)=IQ
      AK(L5)=EVENT(II)
      IP=IR+IS
      GO TO 14
  6 DO 7 KI=1,NM
    II=KI
    IF(EVENT(II).EQ.99999.) GO TO 8
    7 CONTINUE
    WRITE (6,100)
    8 EVENT(II)=TIME+R1(1)
    IF(EVENT(II)=6
      IAL(II)=ID1
      IQ=IQ-ID1
      IP=IP-ID1
      IF(IP.GT.IS) GO TO 14
      DO 63 KT=1,10
        L5=KT
        IF(AK(L5).EQ.0.) GO TO 64
        63 CONTINUE
        64 KA(L5)=IQ
        AK(L5)=EVENT(II)
        IP=IR+IS
        GO TO 14
  11 K=K+1
    QS(K)=TIME
    IG(K)=ID1

```

```

TUR07210
TUR07220
TUR07230
TUR07240
TUR07250
TUR07260
TUR07270
TUR07280
TUR07290
TUR07300
TUR07310
TUR07320
TUR07330
TUR07340
TUR07350
TUR07360
TUR07370
TUR07380
TUR07390
TUR07400
TUR07410
TUR07420
TUR07430
TUR07440
TUR07450
TUR07460
TUR07470
TUR07480
TUR07490
TUR07500
TUR07510
TUR07520
TUR07530
TUR07540
TUR07550
TUR07560
TUR07570
TUR07580
TUR07590
TUR07600
TUR07610
TUR07620
TUR07630
TUR07640
TUR07650
TUR07660
TUR07670
TUR07680

```

```

IM(K)=1
IQ=IQ-ID1
IP=IP-ID1
14 SS=TIME
RETURN
DEBUG SUBCHK
END
SUBROUTINE TEN (K,IQ,ID2,TIME,EVENT,IEVENT,IM,IG,QS,IA,IA2,IR,IS,
IP,A,B,A2,B2,DSEED5,DSEED6,NM,NS,TOH,SS,KA,AK,IQQ)
REAL*8 DSEED5,DSEED6
DIMENSION EVENT(NS),IEVENT(NS),IM(NS),IA2(NS),QS(NS),IG(NS),IA(NS)
1,Z(50),Z2(50),R(1),R2(1),KA(10),AK(10)
ID=0
IF(IQ.LE.0) GO TO 11
IF(IQ.LE.0) GO TO 11
HW=0.0
WH=(TIME-SS)*IQ
TOH=TOH+HW
IF(IQ.GE.ID2) GO TO 6
CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
DO 1 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 2
1 CONTINUE
100 WRITE (6,100)
2 EVENT(II)=TIME+R2(1)
IEVENT(II)=7
IA2(II)=IQ
K=K+1
QS(K)=TIME
IG(K)=ID2-IQ
IM(K)=2
IQ=IQ-ID2
IP=IP-ID2
IF(IP.GT.IS) GO TO 14
DO 60 KT=1,10
L5=KT
IF (AK(L5).EQ.0.0) GO TO 61
60 CONTINUE
61 KA(L5)=IQ
AK(L5)=EVENT(II)
IP=IR+IS
GO TO 14
6 CALL GGAMS (DSEED5,A2,B2,1,Z2,R2)
DO 7 KI=1,NM
II=KI
IF(EVENT(II).EQ.99999.) GO TO 8
7 CONTINUE

```

```

TUR07690
TUR07700
TUR07710
TUR07720
TUR07730
TUR07740
TUR07750
TUR07760
TUR07770
TUR07780
TUR07790
TUR07800
TUR07810
TUR07820
TUR07830
TUR07840
TUR07850
TUR07860
TUR07870
TUR07880
TUR07890
TUR07900
TUR07910
TUR07920
TUR07930
TUR07940
TUR07950
TUR07960
TUR07970
TUR07980
TUR07990
TUR08000
TUR08010
TUR08020
TUR08030
TUR08040
TUR08050
TUR08060
TUR08070
TUR08080
TUR08090
TUR08100
TUR08110
TUR08120
TUR08130
TUR08140
TUR08150
TUR08160

```

```

      8  WRITE (6,100)
         1 EVENT(11)=TIME+R2(1)
         2 IEVENT(11)=7
         3 I A2(11)=ID2
         4 IQ=IQ-ID2
         5 IP=IP-ID2
         6 IF(IP.GT.IS) GO TO 14
         7 DO 63 KT=1,10
         8 L5=KT
         9 IF(AK(L5).EQ.0.0) GO TO 64
        10 63 CONTINUE
        11 KA(L5)=IQ
        12 AK(L5)=EVENT(11)
        13 IP=IR+IS
        14 GO TO 14
        15 K=K+1
        16 QS(K)=TIME
        17 IG(K)=ID2
        18 IM(K)=2
        19 IQ=IQ-ID2
        20 IP=IP-ID2
        21 SS=TIME
        22 RETURN
        23 DEBUG SUBCHK
        24 END

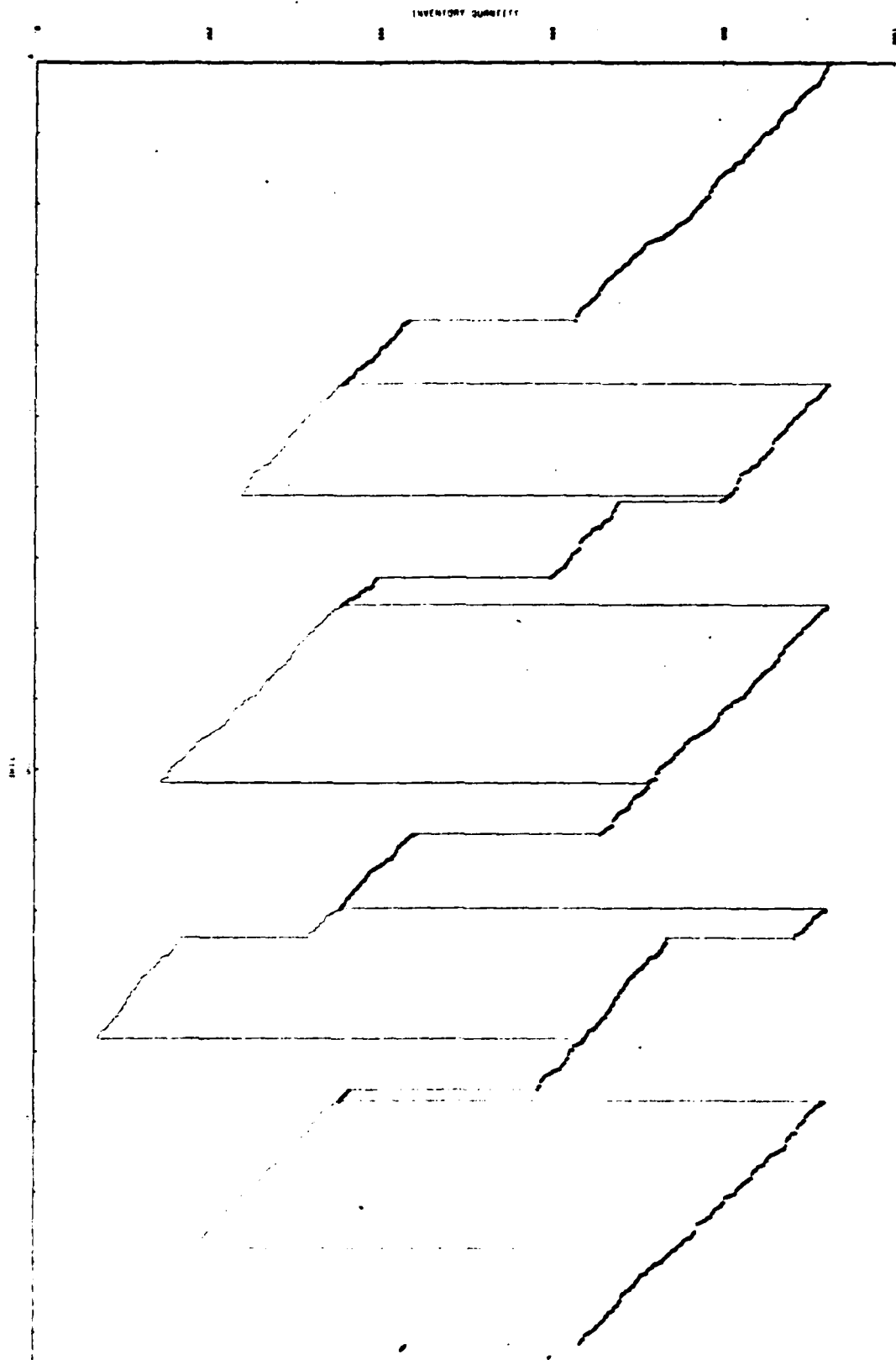
```

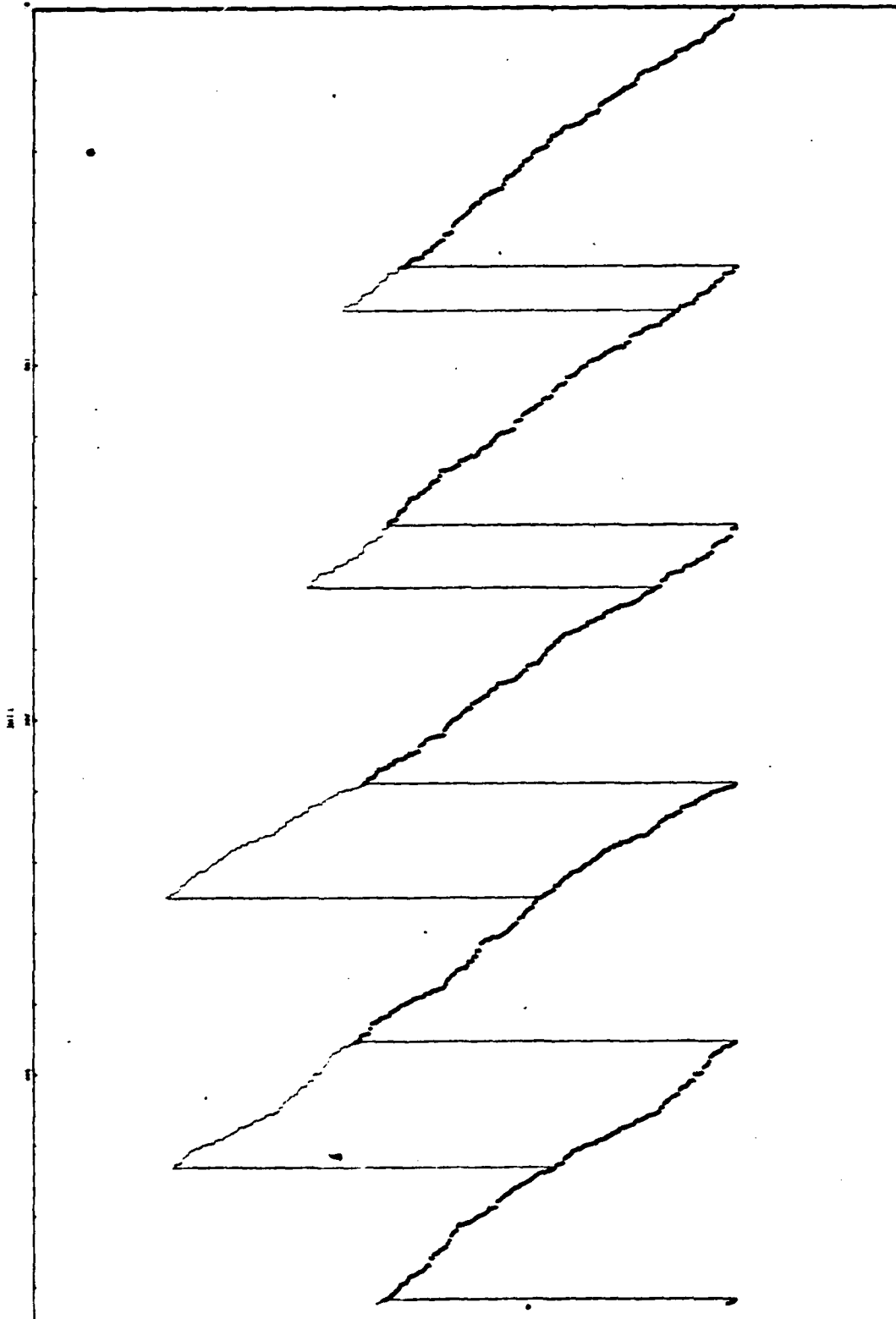
```

TUR08170
TUR08180
TUR08190
TUR08200
TUR08210
TUR08220
TUR08230
TUR08240
TUR08250
TUR08260
TUR08270
TUR08280
TUR08290
TUR08300
TUR08310
TUR08320
TUR08330
TUR08340
TUR08350
TUR08360
TUR08370
TUR08380
TUR08390
TUR08400
TUR08410

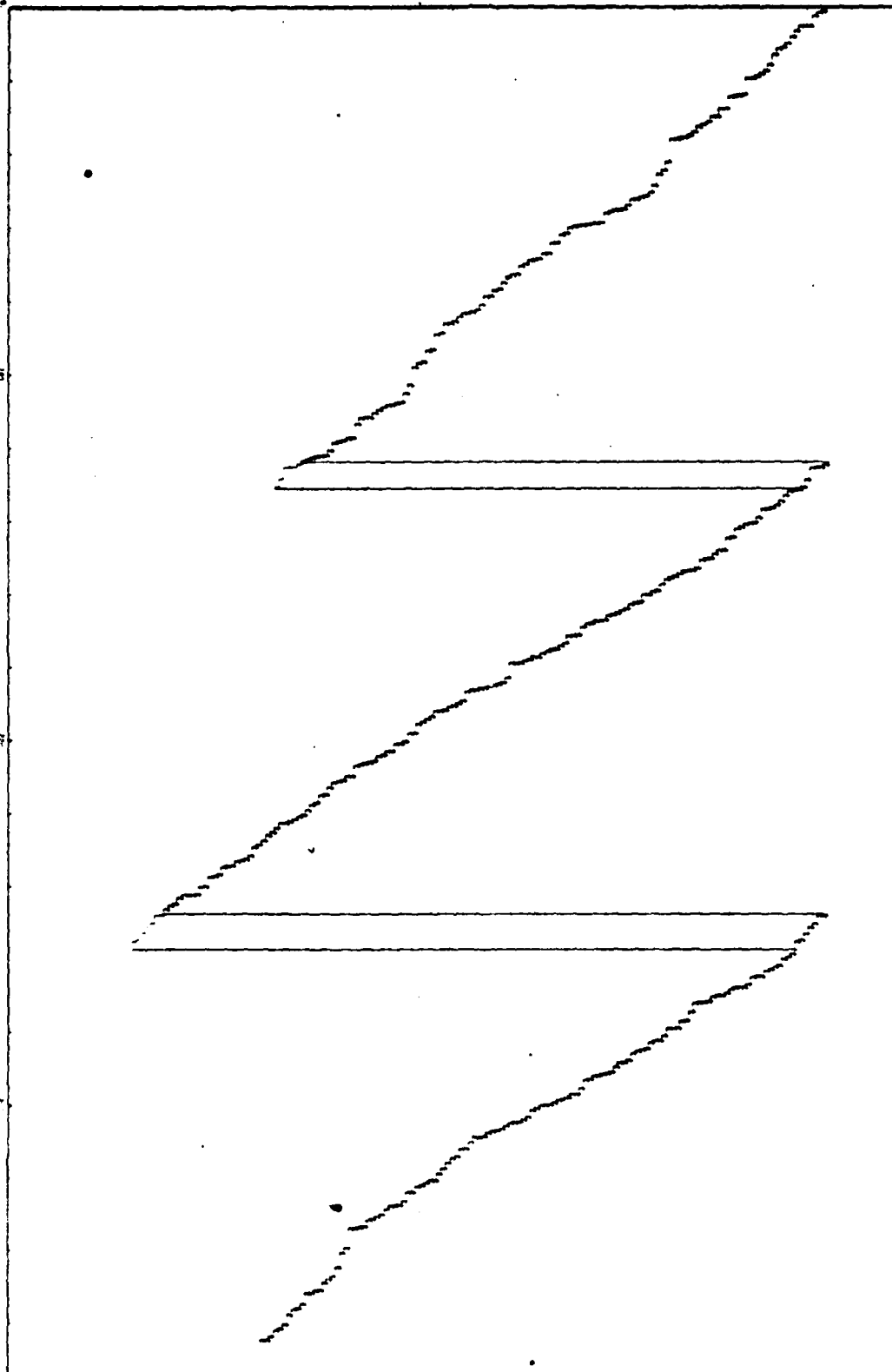
```

APPENDIX F
Versatec Output of the Program in Appendix C





INVENTORY QUANTITY



APPENDIX G

TI-59 Calculator Program for Analytical Solutions for Periodic Review Systems

This program was written for probabilistic periodic review inventory models having gamma distributed lead times and Poisson arrivals and uses. Normal distribution for demand during lead time instead of Negative Binomial.

Input Requirements

The following variables should be stored in the registers shown before the variables.

STO 01 = λ (arrival rate per day)

STO 02 = C

STO 03 = I

STO 04 = Π

STO 05 = J

STO 06 = A

STO 07 = $\alpha-1$

STO 08 = β

This program also requires the TI-59 applied statistics module.

Enter	Press	Display
T	A	R
	B	S
	C	$\eta(r)/\text{period}$
	D	$\eta(r)/\text{year}$
	E	Annual review and order cost
	A'	Annual inventory carrying cost
	B'	Annual shortage cost
	C'	Total annual variable cost

000	75	LEL
001	11	R
002	55	-
003	01	1
004	02	2
005	95	=
006	42	STD
007	20	20
008	53	.
009	53	.
010	43	FCL
011	02	02
012	65	.
013	43	FCL
014	03	03
015	65	.
016	43	FCL
017	20	20
018	34	.
019	35	-
020	43	FCL
021	04	04
022	54	.
023	42	STD
024	21	21
025	53	.
026	43	FCL
027	01	01
028	65	.
029	43	FCL
030	08	08
031	65	.
032	53	.
033	43	FCL
034	07	07
035	85	+
036	01	1
037	54	.
038	54	.
039	42	STD
040	22	22
041	53	.
042	53	.
043	43	FCL
044	01	01
045	65	.
046	43	FCL
047	08	08
048	54	.
049	85	+
050	01	1
051	54	.

052	42	STD
053	23	23
054	50	.
055	03	3
056	06	6
057	05	5
058	55	.
059	43	FCL
060	01	01
061	55	.
062	43	FCL
063	20	20
064	54	.
065	42	STD
066	24	24
067	53	.
068	43	FCL
069	12	12
070	85	+
071	43	FCL
072	24	24
073	54	.
074	42	STD
075	15	15
076	53	.
077	53	.
078	53	.
079	43	FCL
080	23	23
081	65	.
082	43	FCL
083	23	23
084	54	.
085	85	+
086	43	FCL
087	24	24
088	54	.
089	34	FCL
090	54	.
091	42	STD
092	26	26
093	53	.
094	01	1
095	55	.
096	53	.
097	13	FCL
098	21	21
099	33	.
100	54	.
101	54	.
102	23	LEL
103	21	.

104	42	STD
105	27	27
106	53	.
107	43	FCL
108	27	27
109	75	.
110	53	.
111	53	.
112	02	2
113	23	.
114	05	5
115	01	1
116	05	5
117	05	5
118	01	1
119	07	7
120	85	+
121	53	.
122	00	0
123	93	.
124	08	8
125	00	0
126	02	2
127	08	8
128	05	5
129	03	3
130	65	.
131	43	FCL
132	27	27
133	54	.
134	85	+
135	53	.
136	43	FCL
137	27	27
138	33	33
139	65	.
140	93	.
141	00	0
142	01	1
143	00	0
144	03	3
145	02	2
146	08	8
147	54	.
148	54	.
149	55	.
150	93	.
151	01	1
152	95	+
153	53	.
154	01	1
155	53	.

156	04	4
157	03	3
158	02	2
159	07	7
160	08	8
161	08	8
162	65	6
163	43	RCL
164	27	27
165	54	5
166	85	+
167	53	5
168	43	RCL
169	27	27
170	33	33
171	65	6
172	00	0
173	93	.
174	01	1
175	08	8
176	09	9
177	02	2
178	06	6
179	09	9
180	54	5
181	85	+
182	53	5
183	43	RCL
184	27	27
185	33	33
186	65	6
187	43	RCL
188	27	27
189	65	6
190	00	0
191	93	.
192	00	0
193	00	0
194	01	1
195	03	3
196	00	0
197	08	8
198	54	5
199	54	5
200	54	5
201	54	5
202	42	STO
203	28	28
204	53	5
205	53	5
206	43	RCL
207	26	26

208	45	4
209	43	RCL
210	28	28
211	54	5
212	65	6
213	43	RCL
214	25	25
215	54	5
216	42	STO
217	50	50
218	91	R/S
219	76	LBL
220	12	E
221	53	5
222	43	RCL
223	50	50
224	75	-
225	43	RCL
226	25	25
227	54	5
228	42	STO
229	51	51
230	91	R/S
231	76	LBL
232	13	0
233	52	5
234	53	5
235	40	RCL
236	28	28
237	36	PGM
238	19	19
239	11	A
240	65	6
241	43	RCL
242	26	26
243	54	5
244	65	6
245	53	5
246	43	RCL
247	21	21
248	65	6
249	53	5
250	43	RCL
251	25	25
252	75	-
253	43	RCL
254	50	50
255	54	5
256	54	5
257	54	5
258	42	STO
259	50	50

260	91	R/S
261	76	LBL
262	14	D
263	53	5
264	43	RCL
265	52	52
266	55	-
267	43	RCL
268	20	20
269	54	5
270	42	STO
271	53	53
272	91	R/S
273	76	LBL
274	15	E
275	53	5
276	53	5
277	43	RCL
278	05	05
279	85	+
280	43	RCL
281	06	06
282	54	5
283	55	+
284	43	RCL
285	20	20
286	54	5
287	42	STO
288	54	54
289	91	R/S
290	76	LBL
291	16	A
292	53	5
293	53	5
294	53	5
295	43	RCL
296	50	50
297	75	-
298	43	RCL
299	22	22
300	54	5
301	75	-
302	53	5
303	00	0
304	93	.
305	05	5
306	65	6
307	43	RCL
308	24	24
309	54	5
310	54	5
311	65	6

312	43	RCL
313	02	02
314	65	X
315	43	RCL
316	03	03
317	54	)
318	42	STD
319	55	55
320	91	R/S
321	76	LBL
322	17	B'
323	53	(
324	43	RCL
325	53	53
326	65	X
327	43	RCL
328	04	04
329	54	)
330	42	STD
331	56	56
332	91	R/S
333	76	LBL
334	18	C'
335	53	(
336	43	RCL
337	54	54
338	85	+
339	43	RCL
340	55	55
341	85	+
342	43	RCL
343	56	56
344	54	)
345	42	STD
346	57	57
347	91	R/S

APPENDIX H

Variable Definitions for Simulation Programs

IQ = Net inventory

IP = Inventory position

IPP = Second inventory position for early warning system

IR = Order quantity

IS = Reorder level

ID = Number of items demanded by systems

IF = Number of items demanded per demand by ships

QU = Ship queue

QS = Lower echelon's group demand queue in the main system

IG = Amount of demand for each demand waiting in the queue

IA = Amount of shipment arrived

IM = Index. If it is equal to 1, that means that the demand waiting in the main system queue to be filled belongs to System One.

TW = Total waiting time

TOH = Total average on-hand inventory

T = Length of a period

EVENT = This indicates the subroutines

IEVENT = This indicates time of subroutines scheduled

X = Net inventory variable for Versatec plotter

V = Inventory position variable for Versatec plotter

Y = Time for Versatec plotter

WK = Work space for geometric random variable
 WS = Work space for geometric random variable
 WZ = Work space for geometric random variable
 S = Exponential random number
 IK = Geometric random number
 WW = Increment
 I = Indicates the number of ships waiting in the ship queue at System One
 J = Indicates the number of ships waiting in the ship queue at System Two
 L = Indicates the number of ships waiting in the ship queue at the main system
 K = Indicates the number of demand batches waiting in the group demand queue at the main system
 SS = Time indicator
 IQQ = Counter for ship arrivals to indicate the number of items demanded per period
 ML = Multiplier for the number of batches of demand to be ordered from outside supplier
 KA = Indicates the last change on net inventory of the main system in order to get the number of items demanded in a lead time
 AK = Time of last change
 A = Scale parameter for a gamma distribution
 B = Shape parameter for a gamma distribution
 XM = Ship arrival rate per day
 P = Probability of success for geometric distribution

BIBLIOGRAPHY

Hadley, G., and Whitin, T.M., Analysis of Inventory Systems,
Prentice-Hall, 1963.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
4. Asso. Prof. F. Russell Richards, Code 55Rh Department of Operations Research Naval Postgraduate School Monterey, California 93940	4
5. LT Turgut Büyükkarhan Cennet Mah. Hurriyet Cadde Yildiz Apt. No: 64 A Blok Daire 12 Küçükçekmece/Istanbul TURKEY	1

ATE
LMED
-8