AN INTERPRETIVE LANGUAGE FOR QUEUEING THEORY

A Thesis

Presented to the

Faculty of the Department of

Industrial and Systems Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

---

by

Loren Clark Chancellor

May 1973

ACKNOWLEDGEMENTS

I would like to thank the following persons for their aid in preparing this thesis:

Linda, my wife, for correcting and editing the rough drafts;

Charles E. Donaghey, Willard N. Ander, and R.L. Motard, the members of my thesis committee for their help;

And Marvin Smith for his advice and assistance throughout this project.

AN INTERPRETIVE LANGUAGE FOR QUEUEING THEORY

An Abstract of
a Thesis Presented to the
Faculty of the Department of
Industrial and Systems Engineering
University of Houston

In Partial Fulfillment
of the Requirements for the Degree
Master of Science

---

by
Loren Clark Chancellor
May 1973

ABSTRACT

This thesis presents a user oriented computer language for the solution of basic queueing theory models. The language provides the means via the use of digital computers for obtaining numerical results for certain queueing models. The interpreter for the language is coded in Fortran IV and can be used on most computer systems. The language is readily usable in a conversational mode as well as in a batch mode. Examples are presented to explain the utilization of the language.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 General Discussion

The solution of problems in waiting line or queueing theory has long been one of the major areas of interest in operations research and industrial engineering. Mathematical models are available to describe numerous queueing situations. Many of these models are complex to the extent that they require the use of digital simulation techniques to arrive at numerical approximations. "The theory has not progressed beyond the point where tractable solutions are available for more than a few idealized systems (4, p. 1)." Accurate mathematical formulas, tables, charts, and graphs are readily available but generally they apply only for the solution of a few idealized systems. "A study of these idealized congestion problems will develop an insight into the character of queueing action, and will provide upper bounds for design and reasonableness checks for more accurate simulation studies (4, p. 1)."

To analyze these situations a user-oriented, interpretive type language (called QUEUE) is presented. The language enables the user to input his data in a manner consistent with a description of a waiting line model.

The language is such that most spelling and punctuation mistakes will not cause an error in the problem. The language is coded in Fortran IV and hence may be implemented on almost any computer system. In addition, the program is written in such a way that it is quite easy for an experienced programmer to expand or incorporate any desired changes. All the test runs were made at the University of Houston on the Univac 1108 and the IBM 360/44 computer systems.

1.2 Chapter Summaries

The thesis consists of five chapters, of which Chapter 1 is the introduction. Chapter 2 describes the various principles and elements of queueing situations. Chapter 3 presents the user considerations. This chapter consists of a general description of the capabilities and limitations of QUEUE and of the various error messages. Also included in Chapter 3 is a discussion of the various input instructions and an explanation of the various options available. A description of the queueing models available plus an explanation of the output are presented. Chapter 4 explains the technical aspects of the interpreter. The logic and function of the main and subprograms are discussed. The notation used in the program is also discussed. Chapter 5 consists of a summary and conclusion.

It is hoped that QUEUE will be used as a teaching tool as well as for the solution of practical problems.

Chapter 2

PRINCIPLE ELEMENTS IN A QUEUEING SITUATION

2.0 Introduction

The most general and effective analytical approach to random flow systems is waiting line analysis, or queueing theory. A queueing situation--or, more simply, a "queue"--arises when a customer arrives at a service facility and, finding it busy, is forced to wait. The formation of waiting lines is, of course, a common phenomenon which occurs whenever the current demand for a service exceeds the current capacity to provide that service. Such a situation may occur with customers arriving at a supermarket checkout counter, aircraft arriving at an airport, messages arriving at a switching center, etc. (4, p. 2).

The ultimate goal of queueing theory is to achieve an economic balance between the cost of service and the cost associated with waiting for that service. Queueing theory itself does not directly solve this problem, but, it does provide vital information required for such a decision (3, p. 285).

Three elements must be characterized to describe a queueing system mathematically. These elements are the input traffic, the waiting line, and the service facility (see figure 1).

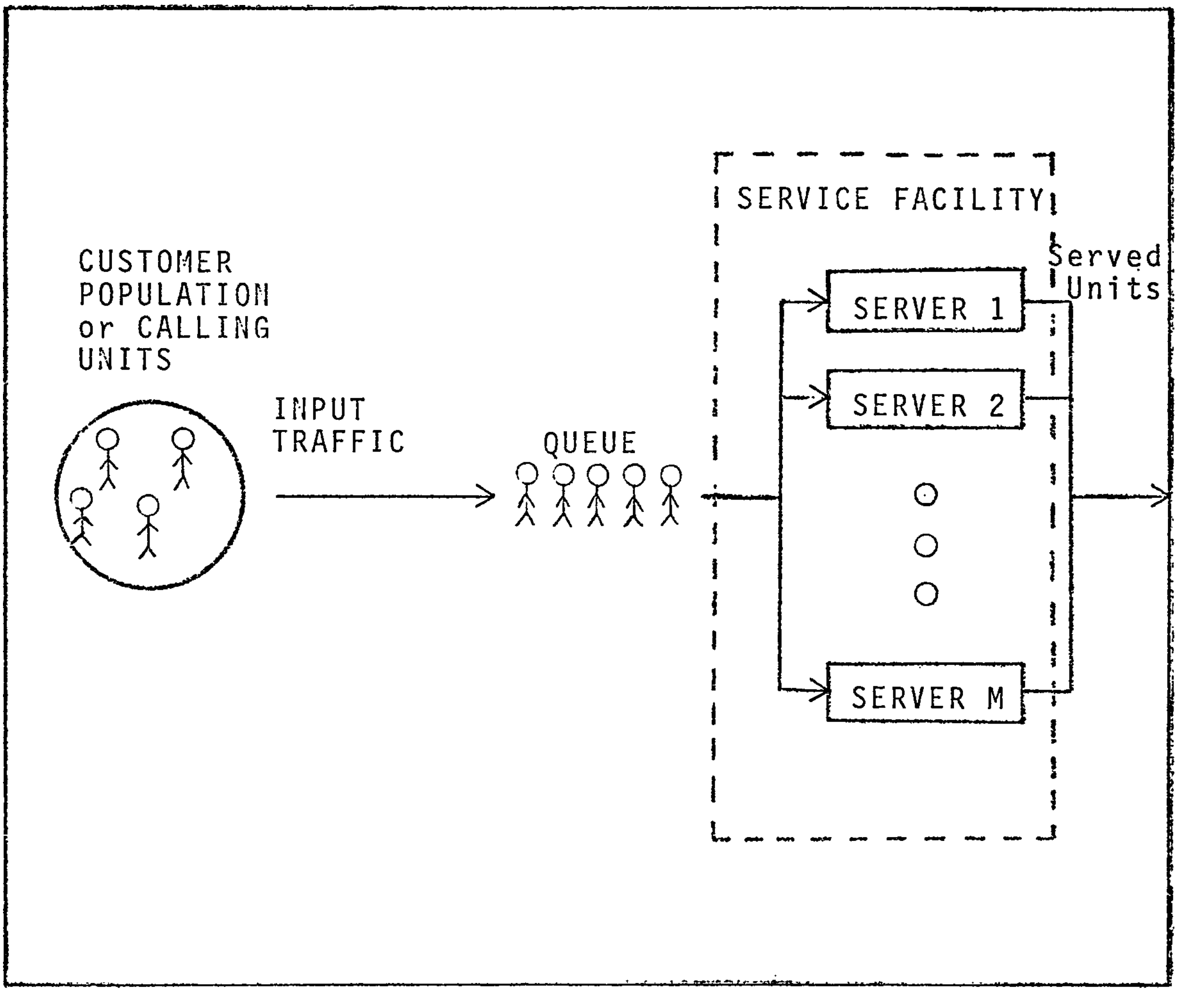

Figure 1

An Elementary Queueing System

These elements can then be placed in various configurations representing a particular model of a queueing system, and the effects of the elements on each other can be studied (4, p. 2).

2.1 Input Traffic or Source

The unit of traffic, or calling unit, that moves through a queueing system is called a "customer." The customer is drawn from the input source or "calling population." Two important characteristics of this population that determine the input traffic are the size of the population and the arrival rate of customers from the calling population to the queue (4, p. 2).

2.1.1 Calling Population

The size of the calling population is the total number of distinct potential customers. It may be assumed to be either infinite or finite (3, p. 287). In usual practice, if the population is sufficiently large such that the arrival rate is not affected by the depletion in the population caused by those units waiting for service and being served, then the population is assumed to be infinite. When the infinite population assumption is not valid, one must take into account the depletion of the population (11, p. 2). If the source population is small, there is a significant effect on the arrival rate to a queue.

With an infinite population, the arrivals are assumed to be independent, and the potential customers will never be depleted. An infinite population is closely exemplified by customer telephones in a local area, wherein calls arrive at a telephone exchange. An example of a finite population is the case of a single operator servicing a group of automatic production machines (3, p. 2; 11, p. 2).

2.1.2 Arrival Rate

There must be some statistical pattern by which calling units are generated over time. The simplest, but not very common, way for customers to arrive is at constant intervals, for instance, one customer every two minutes. More likely, the input to a queueing system is irregular, thus introducing an element of randomness. The arrival pattern must therefore be described by a probability distribution. There are several ways of doing this.

2.1.2.1 Arrival Distribution

One way is to sample the number of arrivals in a fixed-time time interval, say T seconds, and estimate, for all values of k from zero to infinity, the probability of exactly k arrivals in T seconds (4, p. 2). The most common assumption is that the number of calling units generated during any specific time has a Poisson distribution (11, p. 1; 3, p. 288). This is the case where arrivals occur at random, but at a certain average rate denoted by lambda (λ). Three basic assumptions must hold if the input traffic is to be considered Poissonian:

1. The traffic load must be independent of time; that is, peaking conditions must not be predictable.
2. Customer arrivals must be independent of past arrivals.
3. Simultaneous arrivals occur with a small chance of happening.

If we denote the probability of exactly k arrivals in T seconds as $P_k(T)$, the Poisson distribution is expressed by the formula (4, p. 3)

$$P_k(T) = \frac{(\lambda T)^k}{k!} * e^{-\lambda T} \quad ; \lambda > 0, \ k = 0, 1, 2, \ldots$$

2.1.2.2 Interarrival Distribution

An alternate way to describe the arrival pattern is to sample the time between arrivals and then estimate the probability F(t) that the interarrival time is less than a particular time t. When the input traffic is a Poisson process, the interarrival distribution is found to be the exponential distribution (see figure 2).

$$\text{Prob(Interarrival time} < t) = F(t) = 1 - e^{-\lambda t} = 1 - e^{-t/Ta}$$

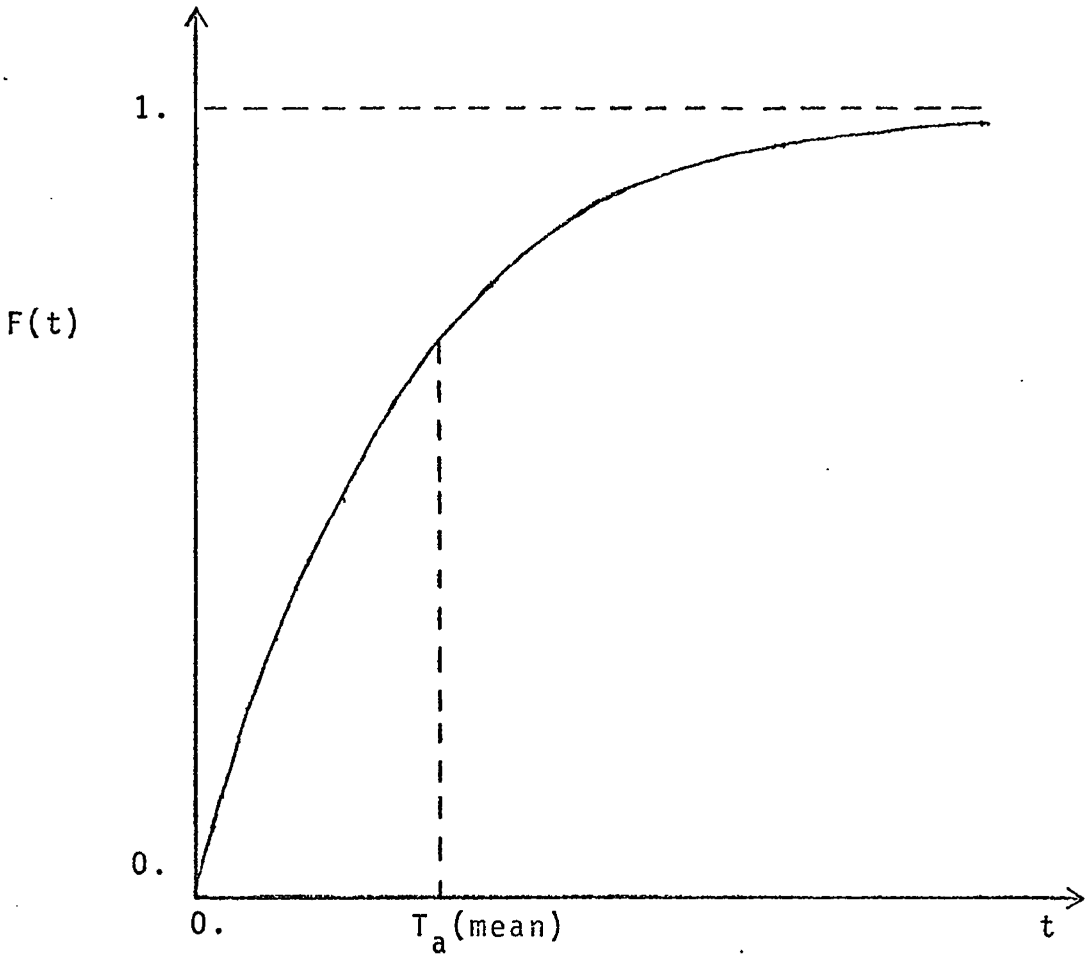

Figure 2

Exponential Interarrival Distribution

The mean interarrival time T_a, equals $1/\lambda$, that is, the reciprocal of the arrival rate, as one would expect. Note that F(t) is a continuous distribution in time while the $P_k(T)$ distribution is discrete.

Any unusual assumptions about the behavior of the calling units must also be specified. One example is balking, where a calling unit refuses to enter the system if the queue is too long, and is lost.

2.2 Service Facility

Customers require service of some kind. To satisfy this need, the service facility is made up of one or more "servers" in parallel. Each server can service only one customer at a time, although the service facility as a whole can service several customers simultaneously. In this report, a server is idle if and only if there are no customers waiting to be serviced. If the server is idle and a customer arrives, service begins immediately. Likewise it is assumed that upon completion of service, the next service begins immediately if there is a customer waiting. If no customer is waiting the server becomes idle.

2.2.1 Service Distribution (11, p. 2; 4, p. 3)

It will be assumed that individual customer service times are mutually independent, identically distributed random variables having a statistical distribution that can be approximated from actual observation. The mean service time will be denoted by T_s, or $1/\mu$, where mu (μ) represents the mean service rate.

The theoretical service distribution of great interest is the exponential distribution (see figure 3).

$$\text{Prob(Service time} < t) = H(t) = 1 - e^{-\mu t} = 1 - e^{-t/Ts}$$

In most cases of practical interest, service times are not exponentially distributed. However, the exponential form usually represents a worst-case assumption and often leads to great simplification in the form of the solution; it should be assumed when better information is lacking (4. p. 4). Distributions with variablility greater or less than that of the exponential distribution are called hyperexponential and hypoexponential (Erlang) distributions, respectively. For distributions whose special characteristics make analytic solutions too complicated, other techniques are used, the most generally applicable and useful of which is the Monte Carlo Simulation Method (11, p. 11).

2.3 Waiting Line or Queue (4, p. 4)

Whenever arrivals or service times of a queueing system fluctuate in a probabilistic manner, a waiting line may form, causing waiting time delays for the arriving customers. One of the purposes of queueing theory is to determine the expected size of these queues and the extent

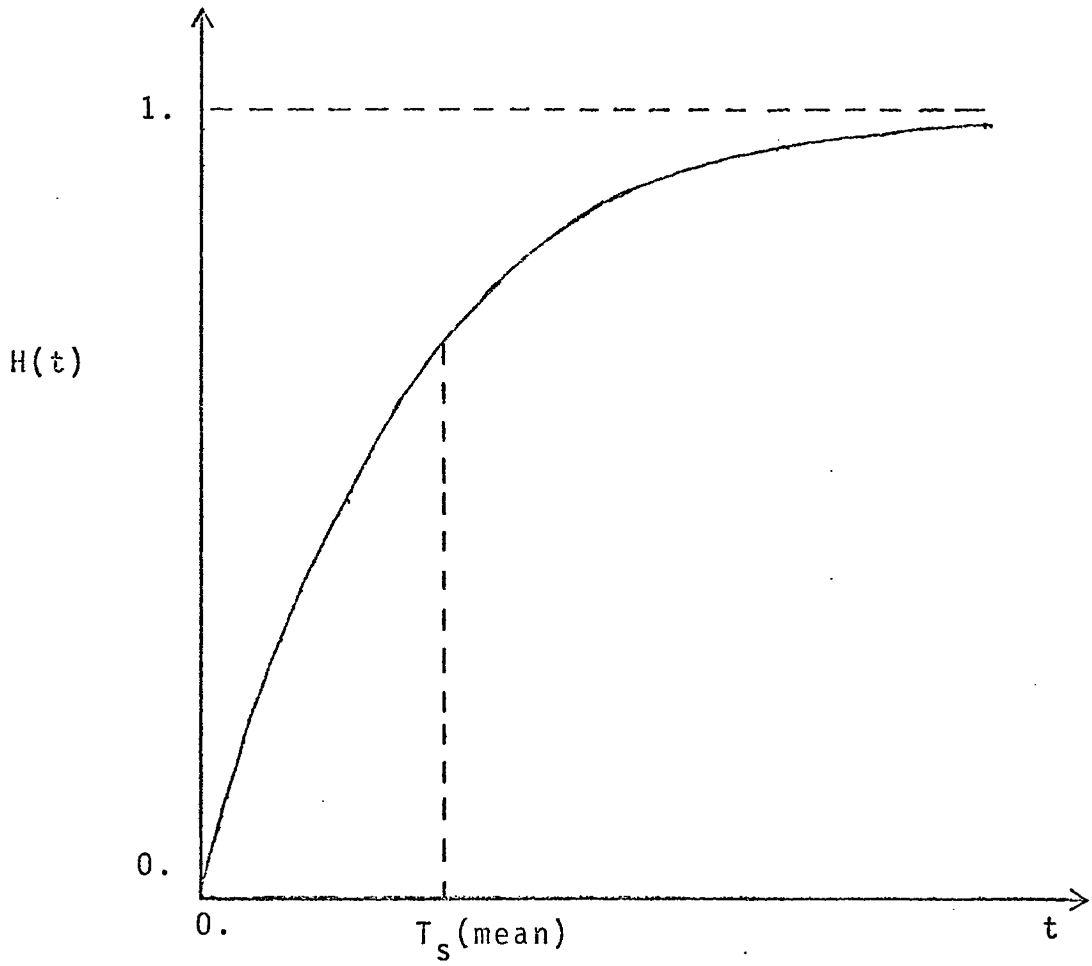

Figure 3

Exponential Service Time Distribution

of the waiting delays encountered. In this report a waiting line refers to a system of servers with a common queue. Thus a queue is only part of a waiting line in a system. A queue refers only to that part of a waiting line that is waiting for service to begin. Since this distinction is sometimes not made, or is even reversed, in the literature on queues, one should be careful to determine what a particular author means by these terms. There are two important characteristics to consider in the organization of a queue.

2.3.1 Waiting Line Limit or Capacity

A waiting line is characterized by the maximum permissible number of units that it can contain or hold. This is equal to the maximum size of the queue (number waiting in line for service to begin) plus the number of servers. For instance, if the capacity of a queueing system is 10 and there are 2 servers, then the maximum size of the queue is 8.

2.3.2 Queue Discipline (3, p. 288)

The queue discipline is the rule by which the service facility determines the next customer to be served. For example, it may be first-come-first-served, or random, or according to some priority procedure, et cetera. First-come-first-served, or FIFO, is usually implicitly assumed by queueing models unless stated otherwise.

2.4 Measures of Congestion (4, p. 5)

There are several measures of congestion drawn from statistics which describe the effects of a queueing system under a fluctuating traffic load.

2.4.1 Traffic Intensity

"Traffic intensity" is a dimensionless ratio, denoted by Psi (ψ), indicating the impact of a traffic stream upon a service system. It is defined as

$$\text{Psi} = \frac{\text{mean service time for customers}}{\text{mean time between customer arrivals}} .$$

$$\psi = T_s \;/\; T_a = \lambda T_s = \lambda \;/\; \mu$$

In general, the traffic intensity of any queueing system indicates the least number of servers required to handle a given traffic stream with no loss of customers.

2.4.2 Server Utilization

Closely related to the traffic intensity is the "server utilization." This quantity, denoted by Rho (ρ), measures the fraction of time that a single server is busy.

Server utilization, $\rho = \lambda T_s \;/\; m$, where m is the of servers.

At $\rho=1$, the server becomes satuated, working 100% of the time. However, queues become very large near system saturation, growing without bound when $\rho>1$. Practical considerations, such as response requirements or storage size, usually limit the input rate for a single server to 70-90% of the theoretical maximum.

2.4.3 Number of Customers

The two quantities of greatest practical interest in a queueing system that give a dynamic picture of the system performance are:

1. The mean number of customers in the system at any instant.
2. The mean of the length of time a customer spends in the system.

The variances of these parameters are also of great value.

The number of customers in the system may be expressed by a probability distribution P_k= Probability (k customers are in the system).

2.4.4 Queue and System Size

Usually one is interested in the number of customers in the system. In some cases, because of storage problems, one is more concerned about the mean number of customers in the queue alone. Both of these values, as well as their variances, can be derived from the distribution P_k, and are denoted by:

L = expected number of customers in system

σ_L^2 = variance of number in system

L_q = expected number of customers in the queue alone

$\sigma^2_{L_q}$ = variance of number in queue

2.4.5 Queue and System Time

One may be also interested in the time a customer waits in line and the total time spent in the system. These values are denoted by:

W = expected time in system (includes service)

σ^2_W = variance of time in system

W_q = expected waiting time in the queue

$\sigma^2_{W_q}$ = variance of waiting time in the queue

Chapter 3

USER CONSIDERATIONS

3.1 Introduction

There is a general lack of computer programming routines available for the solution of general queueing theory models. The purpose of this language is to provide numerical solutions to a wide variety of common queueing models. The purpose is not to instruct the user in the proper use of queueing theory, though examples will be given to illustrate the usefulness of the language.

Various mathematical formulas, tables, charts, and graphs are available in many excellent texts on queueing theory. These can be used to obtain solutions, but, considerable confusion can arise due to the difficulty of describing the model in the precise terminology necessary for a particular text. Many of the formulas are quite complex and require considerable calculations, some of which may require the use of tables or calculators. One then needs to check the notation, terminology, and assumptions used, since these are far from being standardized.

To overcome these difficulties an interpretive type of language is presented named QUEUE. The language enables the user to input his model in a form similar to the form that would normally be used in the description of a queueing model. The ease of describing and altering a particular model should provide the user with considerable insight into the various aspects of the queueing situation. The language is coded in Fortran IV and hence may be implemented on almost any computer system.

3.2 General Information

The input to the system is format free with one instruction permitted per card. Blank cards and comment cards may be inserted anywhere in the program. Input cards are printed as is, and imbedded blanks have no other meaning. When the first character punched in a card is a dollar sign "$", the card is considered to be a comment card and the contents of the card are ignored. The comma ",", the colon":", and the equal sign "=" are used as punctuation in the input instructions. In most cases, they may be interchanged in use.

3.3 Input Command Instructions

QUEUE input instructions can be started in any column of the punch card. The user does not need to be concerned with integer or real numbers. The maximum size of an input number is nine digits with no more than six digits to the right of the decimal point. Negative numbers will have no meaning and therefore will be considered positive. Instruction names and distributions may generally be abbreviated to one or two letters or they may be extended.

3.3.1 Arrival Instruction

The arrival instruction has the general form:

ARRIVAL: distribution: parameter=values

1. Arrival Distributions
   a. POISSON or P
2. Arrival Parameters
   a. TA mean time between arrivals (time units)
   b. LAMBDA mean arrival rate (arrivals/unit time)
3. Value
   a. Number
   b. In the case of multi-stream models the numbers are separated by commas.

Examples:

```
ARRIVAL:POISSON:TA=3.5
ARRIVAL:POISSON:LAMBDA=.3,.4,.5,.6
A:P:TA=3.5
```

3.3.2 Service Instruction

The service instruction has the general form:

SERVICE: distribution: parameters=values

1. Service Distribution
   a. EXPONENTIAL or E
   b. GENERAL or G
   c. ERLANG-k or ER-k, where k is an integer
   d. CONSTANT or C

2. Service Parameters

   a. TS mean service time (time units)

   b. MU mean service rate (units served/ time unit)

   c. VAR variance for service time distribution, needed for the general distribution only.

Examples:

SERVICE:EXPONENTIAL:TS=4

SERVICE:GENERAL:TS:4.3, VAR=1.3

SERVICE,ERLANG-3,MU:6.22

S,C,TS,3

3.3.3 Channel Instruction

The channel instruction has the general form:

CHANNELS=k , where k is the number of service channels in parallel.

An optional form that can be used only when the arrival distribution is Poisson, the service distribution is exponential, capacity and population are unlimited, and the queue discipline is FIFO, is as follows:

CHANNELS:FIND:j%,T=k , where j and k are numbers.

Examples:

CHANNELS=4

CHANNELS:FIND:90%,T=40. (90% of arrivals to begin service by T time units)

The program assumes channels to be equal to 1 unless user specifies otherwise.

3.3.4 Population of Calling Units

The population instruction has the general form:

POPULATION=value

1. Value is a number or an alpha field, the alpha field implies that the population is infinite or unlimited.
2. Population is assumed to be infinite unless otherwise stated.

Examples:

POPULATION=10

POPULATION:INFINITE

P,UNLIMITED

3.3.5 Capacity of System

Maximum size for number of units in queue plus those being served is capacity of system. The general form of the capacity instruction is:

CAPACITY=value

1. Value has the same meaning as in Population Instruction.
2. Capacity is assumed to be unlimited unless otherwise stated.

Examples:

CAPACITY:10

CA:INF

3.3.6 Queue Discipline

Queue discipline is the rule by which the server picks the next customer to be served from the queue. The queue discipline instruction has the general form:

DISCIPLINE:discipline

1. Discipline
   a. FIFO
   b. NONPREEMPTIVE PRIORITY or NON
   c. PREEMPTIVE-RESUME PRIORITY or PRE
   d. NO PRIORITY MULTI-STREAM or NO P
2. FIFO is assumed unless otherwise stated.

Examples:

DISCIPLINE:NONPRE

D=FIFO

3.3.7 Solve Instruction

The solve instruction has the form:

SOL

Examples:

SOLUTION

SOLVE

The command SOLve will cause the calculations and printing of the results of the current model. After this instruction, the current model may be modified by changing a current instruction or assumption. This is done by simply adding the necessary instruction. An example of this will be illustrated later. To start a new problem, the NEXT instruction should be given. This will cause the program to re-initialize all variables and assumptions. If omitted the assumptions and variables remain the same as the preceding problem with the exception of those variables changed by a new instruction. To end the program the END instruction is used. This instruction is not necessary in batch mode.

3.3.8 Next Problem Instruction

The next problem instruction has the form:

N

Examples:

```
N
NEXT
NEXT PROBLEM PLEASE
```

3.3.9 End Instruction

The end of program instruction has the form:

END

3.4 Error Messages

Error messages have been incorporated into the QUEUE interpreter. These messages appear immediately after the command that caused the error. The message may be only a warning or a fatal error. If the error is fatal a message will state so and the interpreter will skip all instructions until a NEXT problem instruction is encountered. An error number is printed, but it is of no particular value to the user.

3.5 Queueing Models Available

In describing the following models the standard assumptions made in the input commands will remain. Description of model is of the form:

Arrival distribution/ Service distribution/ Number of servers/ Other restrictions

Model 1	Poisson/	Exponential /	1/	
Model 2	Poisson/	Exponential /	1/	Limited queue capacity
Model 3	Poisson/	Exponential /	1/	Limited source population
Model 4	Poisson/	Exponential /	>1/	
Model 5	Poisson/	Exponential /	>1/	Limited queue capacity
Model 6	Poisson/	Exponential /	>1/	Limited source population
Model 7	Poisson/	General /	1/	Variance is needed
	Poisson/	Constant /	1/	Variance is zero
Model 8	Poisson/	Erlang-k /	1/	
Model 11	Poisson/	Any available/	1/	Nonpreemptive priority
Model 12	Poisson/	Any available/	1/	Preemptive-resume priority
Model 13	Poisson/	Any available/	1/	No priority multi-stream

3.6 Output Statements

The output information is similar for each type of model available, but is not entirely the same. The user has no option on the content or format of the output. A description of the output provided follows.

Output common to all models:

Notation	Terminology
	Problem number
	Model type
	Type of arrival
	Size of source population
	Type of service distribution
	Number of service channels in parallel
	Queue Capacity of system
	Queue Discipline
TA	Mean Interarrival time
LAMBDA	Mean Arrival Rate
TS	Mean Service Time
MU	Mean Service Rate
RHO	Server Utilization
L	Mean number of customers in system
LQ	Mean number of customers in queue
W	Mean time in system
WQ	Mean time in queue

Description of additional output provided and applicable models

Notation	Terminology	Models
PO	Proportion of time system is idle	1-8
P(k)	Prob (no. of customers in system = k)	1-8
ENTRY	Mean entry rate	5
PSI	Traffic intensity	4-6,11-13
PNOWT	Prob. arrival will not have to wait	3,5,6
PD	Prob. arrival will have to wait	3,6
DELAY	Mean delay for those obliged to wait	3,6
TB	Interarrival time per customer	3,6
AVEC	Average number of empty channels	5,6
VARx	Variance of the variable x	1,4
x%tile	x percentile of confidence levels	1,4
DOWN	Proportion of time a customer is in system	3

```
$3.7      SAMPLE PROBLEMS AND RESULTS
$THIS IS A COMMENT
$EXAMPLE NUMBER ONE
$MODEL NUMBER ONE
$REFERENCE: PROBLEM 5-1, PAGE 87 WM. T. MORRIS
$         "MATERIALS HANDLING MANAGEMENT".
$THIS IS AN EXAMPLE OF A SINGLE CHANNEL MODEL WITH
$         POISSON ARRIVAL AND EXPONENTIAL SERVICE RATES.
$THE ASSUMPTIONS OF UNLIMITED POPULATION, INFINITE
$         QUEUE CAPACITY, AND FIFO SERVICE DISTRIBUTION
$         ARE MADE.
$THIS MODEL GIVES THE GREATEST AMOUNT OF OUTPUT OF ALL
$         THE SINGLE-STREAM MODELS.
$NOTE: IN ADDITION TO THE MEASURES OF CONGESTION
$         THE VARIANCES AND THE 90% AND 95% CONFIDENCE
$         LEVELS ARE GIVEN.
$THE USER MUST TAKE CARE TO KEEP ALL UNITS THE SAME.

$THE FOLLOWING ARE THE INPUT INSTRUCTIONS.
ARRIVAL:POISSON:LAMBDA=20
SERVICE:EXPONENTIAL:MU=25
SOLVE
```

PROBLEM NUMBER 1

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 1

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA= 0.0500

MEAN ARRIVAL RATE . . . . LAMBDA= 20.0000

MEAN SERVICE TIME . . . . TS= 0.0400

MEAN SERVICE RATE . . . . MU= 25.0000

SERVER UTILIZATION . . . . RHO= 0.8000

PROPORTION OF TIME SYSTEM IS IDLE . . . . P0= 0.2000
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . . P(1)= 0.1600
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . . P(2)= 0.1280
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . . P(3)= 0.1024
PROB(NO. OF CUSTOMERS IN SYSTEM = 4) . . P(4)= 0.0819
PROB(NO. OF CUSTOMERS IN SYSTEM = 5) . . P(5)= 0.0655
PROB(NO. OF CUSTOMERS IN SYSTEM = 6) . . P(6)= 0.0524
PROB(NO. OF CUSTOMERS IN SYSTEM = 7) . . P(7)= 0.0419
PROB(NO. OF CUSTOMERS IN SYSTEM = 8) . . P(8)= 0.0336
PROB(NO. OF CUSTOMERS IN SYSTEM = 9) . . P(9)= 0.0268
PROB(NO. OF CUSTOMERS IN SYSTEM =10) . . P(10)= 0.0215

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 4.0000

VARIANCE OF NUMBER IN SYSTEM . . . . VARL= 20.0000

90% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN 9.3189

95% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN 12.4251

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 3.2000

VARIANCE OF NUMBER IN QUEUE . . . . VARLQ= 18.5600

MEAN TIME IN SYSTEM . . . . W= 0.2000

VARIANCE OF TIME IN SYSTEM . . . . VARW= 0.0400

90% OF THE TIME, TIME IN SYSTEM IS LESS THAN 0.4600

95% OF THE TIME, TIME IN SYSTEM IS LESS THAN 0.6000

MEAN TIME IN QUEUE . . . . WQ= 0.1600

VARIANCE OF QUEUE TIME . . . . VARWQ= 0.0384

90% OF THE TIME, TIME IN QUEUE IS LESS THAN 0.4159

95% OF THE TIME, TIME IN QUEUE IS LESS THAN 0.5545

```
NEXT
$NEXT PROBLEM INSTRUCTION CAUSES ALL PREVIOUS INSTRUCTIONS
$        DESTROYED FROM MEMORY.
$THUS THE PROBLEM AFTER NEXT HAS TO BE DESCRIBED AS IF IT
$        WERE THE FIRST PROBLEM.
$EXAMPLE NUMBER TWO
$MODEL NUMBER TWO
$REFERENCE: PAGE 30 WM. W. HINES, "WAITING-LINE MODELS".
$THIS IS SIMILAR TO MODEL ONE WITH THE EXCEPTION OF THE
$        CAPACITY OF THE SYSTEM BEING LIMITED.
ARRIVAL:POISSON:LAMBDA=8
SERVICE: EXPONENTIAL:MU=8.222
CAPACITY OF SYSTEM=5
    SOLVE
```

PROBLEM NUMBER 2

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 2

ARRIVAL DISTRIBUTION IS PCISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS EQUAL TO 5

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA= 0.1250

MEAN ARRIVAL RATE . . . . LAMBDA= 8.0000

MEAN SERVICE TIME . . . . TS= 0.1216

MEAN SERVICE RATE . . . . MU= 8.2220

SERVER UTILIZATION . . . . RHO= 0.9730

PROPORTION OF TIME SYSTEM IS IDLE . . . . P0= 0.1783
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . . P(1)= 0.1735
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . . P(2)= 0.1688
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . . P(3)= 0.1642
PROB(NO. OF CUSTOMERS IN SYSTEM = 4) . . P(4)= 0.1598
PROPORTION OF TIME SYSTEM IS FULL . . . . P(5)= 0.1555

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 2.4202

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 1.5985

MEAN TIME IN SYSTEM . . . . W= 0.3025

MEAN TIME IN QUEUE . . . . WQ= 0.1809

```
NEXT PROBLEM PLEASE
$EXAMPLE NUMBER THREE
$MODEL NUMBER THREE
$REFERENCE: PAGE 41, I.B.M. MANUAL.
$SIX MACHINES / ONE CHANNEL.
$THIS MODEL IS SIMILAR TO MODEL ONE WITH THE EXCEPTION
$        THAT THE CALLING POPULATION IS LIMITED.
$THIS MODEL CONTAINS OUTPUT USING TERMINOLOGY ASSOCIATED
$        WITH THE CASE WHERE MACHINES ARE BEING SERVICED
$        BY ONE SERVER.  FOR EXAMPLE:
$        PROBABILITY OF DELAY
$        DELAY TIME
$        TIME BETWEEN BREAKDOWNS
$        DOWN TIME.
ARRIVAL,POISSON,TA=60
SERVICE=EXPON=TS=12
    POP=6
SOLVE
```

PROBLEM NUMBER 3

DESCRIPTICN AND ANALYSIS CF QUEUEING MODEL TYPE 3

ARRIVAL DISTRIBUTION IS PCISSON

SOURCE POPULATION IS OF SIZE 6

SERVICE TIME DISTRIBUTICN IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA= 60.0000

MEAN ARRIVAL RATE . . . . LAMBCA= 0.0167

MEAN SERVICE TIME . . . . TS= 12.0000

MEAN SERVICE RATE . . . . MU= 0.0833

SERVER UTILIZATION . . . . RHO= 0.2000

PROPORTION OF TIME SYSTEM IS IDLE . . . . PO= 0.1918
PROB(NO. CF CUSTOMERS IN SYSTEM = 1) . . P(1)= 0.2302
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . . P(2)= 0.2302
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . . P(3)= 0.1842
PROB(NO. CF CUSTOMERS IN SYSTEM = 4) . . P(4)= 0.1105
PROB(NO. CF CUSTOMERS IN SYSTEM = 5) . . P(5)= 0.0442
PROPORTION OF TIME SYSTEM IS FULL . . . . P(6)= 0.0088

PROBABILITY ARRIVAL WILL NOT HAVE TO WAIT PNOWT= 0.1918

PROBABILITY ARRIVAL WILL HAVE TO WAIT . . . . PD= 0.8082

MEAN DELAY FOR THOSE OBLIGED TO WAIT . . DELAY= 21.1496

INTERARRIVAL TIME PER CUSTOMER . . . . TB= 89.0921

PROPORTION OF TIME A CUSTCMER IS IN SYSTEM DCWN= 0.3265

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 1.9592

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 1.1511

MEAN TIME IN SYSTEM . . . . W= 29.0921

MEAN TIME IN QUEUE . . . . WQ= 17.0921

```
NEXT
$EXAMPLE NUMBER FOUR
$MODEL NUMBER FCUR
$REFERENCE: PAGE 37, I.B.M. MANUAL.
$THIS IS AN EXAMPLE OF A MULTI-CHANNEL MODEL WITH
$        POISSCN ARRIVAL AND EXPONENTIAL SERVICE RATES.
$THE REGULAR ASSUMPTIONS ARE MADE.
$NOTICE STURT FORM OF INSTRUCTIONS.
A,P,LAM=.5
S,E,TS=5
CHANNELS=3
SOLVE
```

PROBLEM NUMBER 4

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 4

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 3

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA= 2.0000

MEAN ARRIVAL RATE . . . . LAMBDA= 0.5000

MEAN SERVICE TIME . . . . TS= 5.0000

MEAN SERVICE RATE . . . . MU= 0.2000

TRAFFIC INTENSITY . . . . PSI= 2.5000

SERVER UTILIZATION . . . . RHO= 0.8333

PROPORTION OF TIME SYSTEM IS IDLE . . . . P0= 0.0449
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . . P(1)= 0.1124
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . . P(2)= 0.1404
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . . P(3)= 0.1170
PROB(NO. OF CUSTOMERS IN SYSTEM = 4) . . P(4)= 0.0975
PROB(NO. OF CUSTOMERS IN SYSTEM = 5) . . P(5)= 0.0813
PROB(NO. OF CUSTOMERS IN SYSTEM = 6) . . P(6)= 0.0677
PROB(NO. OF CUSTOMERS IN SYSTEM = 7) . . P(7)= 0.0564
PROB(NO. OF CUSTOMERS IN SYSTEM = 8) . . P(8)= 0.0470
PROB(NO. OF CUSTOMERS IN SYSTEM = 9) . . P(9)= 0.0392
PROB(NO. OF CUSTOMERS IN SYSTEM =10) . . P(10)= 0.0327
PROB(NO. OF CUSTOMERS IN SYSTEM =11) . . P(11)= 0.0272
PROB(NO. OF CUSTOMERS IN SYSTEM =12) . . P(12)= 0.0227
PROB(NO. OF CUSTOMERS IN SYSTEM =13) . . P(13)= 0.0189

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 6.0112

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 3.5112

VARIANCE OF NUMBER IN QUEUE . . . . VARLQ= 26.2948

MEAN TIME IN SYSTEM . . . . W= 12.0225

VARIANCE OF TIME IN SYSTEM . . . . VARW= 116.1343

MEAN TIME IN QUEUE . . . . WQ= 7.0225

VARIANCE OF QUEUE TIME . . . . VARWQ= 91.1343

```
NEXT
$EXAMPLE NUMBER FIVE
$MODEL NUMBER FCUR
$REFERENCE: PAGE 34, I.B.M. MANUAL.
$THIS EXAMPLE FINDS THE MINIMUM NUMBER OF CHANNELS
$        NECESSARY FCR 90% OF THE CUSTOMERS WILL WAIT
$        FOR SERVICE LESS THAN 4C TIME UNITS.
$THE OUTPUT LISTS THIS MINIMUM NUMBER OF SERVERS PLUS THE
$        CALCULATED %TILE VALUE FOR THIS NUMBER OF SERVERS.
ARRIVAL:P:TA=6
S:E,TS=30
CHANNELS:FIND:90%:T=40
SOLVE
```

PROBLEM NUMBER 5

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 4

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

THE NUMBER OF SERVICE CHANNELS IS TO BE FOUND
SUCH THAT 90.00% OF THE CUSTOMERS WAIT LESS THAN 40.00 TIME UNITS

NUMBER OF SERVICE CHANNELS NEEDED IS = 7

THIS GIVES AN ACTUAL PERCENTILE VALUE OF 97.75

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA= 6.0000

MEAN ARRIVAL RATE . . . . LAMBDA= 0.1667

MEAN SERVICE TIME . . . . TS= 30.0000

MEAN SERVICE RATE . . . . MU= 0.0333

TRAFFIC INTENSITY . . . . PSI= 5.0000

SERVER UTILIZATION . . . . RHO= 0.7143

PROPORTION OF TIME SYSTEM IS IDLE . . . . PO= 0.0060
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . . P(1)= 0.0299
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . . P(2)= 0.0747
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . . P(3)= 0.1245
PROB(NO. OF CUSTOMERS IN SYSTEM = 4) . . P(4)= 0.1556
PROB(NO. OF CUSTOMERS IN SYSTEM = 5) . . P(5)= 0.1556
PROB(NO. OF CUSTOMERS IN SYSTEM = 6) . . P(6)= 0.1297
PROB(NO. OF CUSTOMERS IN SYSTEM = 7) . . P(7)= 0.0926
PROB(NO. OF CUSTOMERS IN SYSTEM = 8) . . P(8)= 0.0662
PROB(NO. OF CUSTOMERS IN SYSTEM = 9) . . P(9)= 0.0473
PROB(NO. OF CUSTOMERS IN SYSTEM =10) . . P(10)= 0.0338
PROB(NO. OF CUSTOMERS IN SYSTEM =11) . . P(11)= 0.0241
PROB(NO. OF CUSTOMERS IN SYSTEM =12) . . P(12)= 0.0172

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 5.8104

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 0.8104

VARIANCE OF NUMBER IN QUEUE . . . . VARLQ= 4.2055

MEAN TIME IN SYSTEM . . . . W= 34.8622

VARIANCE OF TIME IN SYSTEM . . . . VARW=1022.2256

MEAN TIME IN QUEUE . . . . WQ= 4.8622

VARIANCE OF QUEUE TIME . . . . VARWQ= 122.2257

```
NEXT
$EXAMPLE NUMBER SIX
$MODEL NUMBER FIVE
$REFERENCE: PAGE 48, HINES.
$THIS MODEL IS SIMILAR TC MCDEL FOUR WITH THE EXCEPTION
$        THAT THE CAPACITY OF THE SYSTEM IS LIMITED.
$NOTE: THE ENTRY RATE IS GIVEN, THIS IS LESS THAN THE
$        ARRIVAL RATE DUE TO CUSTOMERS THAT BALK BECAUSE
$        THE SYSTEM IS FULL.
$THE MEAN NUMBER OF EMPTY CHANNELS IS ALSO GIVEN.
A,P,TA=3
SE,  EXP,TS=12
CHAN=5
CAPACITY OF SYSTEM IS=20
SOLVE PROBLEM
```

PROBLEM NUMBER 6

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 5

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 5

QUEUE CAPACITY OF SYSTEM IS EQUAL TO 20

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME	TA=	3.0000
MEAN ARRIVAL RATE	LAMBDA=	0.3333
MEAN ENTRY RATE	ENTRY=	0.3320
MEAN SERVICE TIME	TS=	12.0000
MEAN SERVICE RATE	MU=	0.0833
TRAFFIC INTENSITY	PSI=	4.0000
SERVER UTILIZATION	RHO=	0.8000
PROPORTION OF TIME SYSTEM IS IDLE	P0=	0.0132
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . .	P(1)=	0.0528
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . .	P(2)=	0.1055
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . .	P(3)=	0.1407
PROB(NO. OF CUSTOMERS IN SYSTEM = 4) . .	P(4)=	0.1407
PROB(NO. OF CUSTOMERS IN SYSTEM = 5) . .	P(5)=	0.1126
PROB(NO. OF CUSTOMERS IN SYSTEM = 6) . .	P(6)=	0.0901
PROB(NO. OF CUSTOMERS IN SYSTEM = 7) . .	P(7)=	0.0721
PROB(NO. OF CUSTOMERS IN SYSTEM = 8) . .	P(8)=	0.0576
PROB(NO. OF CUSTOMERS IN SYSTEM = 9) . .	P(9)=	0.0461
PROB(NO. OF CUSTOMERS IN SYSTEM =10) . .	P(10)=	0.0369
PROB(NO. OF CUSTOMERS IN SYSTEM =11) . .	P(11)=	0.0295
PROB(NO. OF CUSTOMERS IN SYSTEM =12) . .	P(12)=	0.0236
PROB(NO. OF CUSTOMERS IN SYSTEM =13) . .	P(13)=	0.0189
PROB(NO. OF CUSTOMERS IN SYSTEM =14) . .	P(14)=	0.0151
PROB(NO. OF CUSTOMERS IN SYSTEM =15) . .	P(15)=	0.0121
PROB(NO. OF CUSTOMERS IN SYSTEM =16) . .	P(16)=	0.0097
PROB(NO. OF CUSTOMERS IN SYSTEM =17) . .	P(17)=	0.0077
PROB(NO. OF CUSTOMERS IN SYSTEM =18) . .	P(18)=	0.0062
PROB(NO. OF CUSTOMERS IN SYSTEM =19) . .	P(19)=	0.0050
PROPORTION OF TIME SYSTEM IS FULL	P(20)=	0.0040
PROBABILITY ARRIVAL WILL NOT HAVE TO WAIT	PNOWT=	0.4530
MEAN NUMBER OF EMPTY CHANNELS	AVEC=	1.0158
MEAN NUMBER OF CUSTOMERS IN SYSTEM	L=	5.9188
MEAN NUMBER OF CUSTOMERS IN QUEUE	LQ=	1.9347
MEAN TIME IN SYSTEM	W=	17.8271

```
NEXT
$EXAMPLE NUMBER SEVEN
$MODEL NUMBER FIVE
$REFERENCE: PAGE 53, HINES.
$A SPECIAL CASE OF THE MULTI-CHANNEL MODEL WHERE THE
$        CAPACITY OF THE SYSTEM EQUALS THE NUMBER OF
$        SERVERS, THUS NC QUEUE IS ALLOWED.
A,P,TA=.5
SER,EXP,TS=10
CHAN=10
CAP=10
SOLVE
```

PROBLEM NUMBER 7

DESCRIPTION AND ANALYSIS CF QUEUEING MODEL TYPE 5

ARRIVAL DISTRIBUTION IS PCISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 10

QUEUE CAPACITY OF SYSTEM IS EQUAL TO 10

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME	TA=	0.5000
MEAN ARRIVAL RATE	LAMBDA=	2.0000
MEAN ENTRY RATE	ENTRY=	0.9241
MEAN SERVICE TIME	TS=	10.0000
MEAN SERVICE RATE	MU=	0.1000
TRAFFIC INTENSITY	PSI=	20.0000
SERVER UTILIZATION	RHO=	2.0000
PROPORTION OF TIME SYSTEM IS IDLE	PO=	0.0000
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . .	P(1)=	0.0000
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . .	P(2)=	0.0000
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . .	P(3)=	0.0003
PROB(NO. OF CUSTOMERS IN SYSTEM = 4) . .	P(4)=	0.0013
PROB(NO. OF CUSTOMERS IN SYSTEM = 5) . .	P(5)=	0.0051
PROB(NO. OF CUSTOMERS IN SYSTEM = 6) . .	P(6)=	0.0169
PROB(NO. OF CUSTOMERS IN SYSTEM = 7) . .	P(7)=	0.0484
PROB(NO. OF CUSTOMERS IN SYSTEM = 8) . .	P(8)=	0.1210
PROB(NO. OF CUSTOMERS IN SYSTEM = 9) . .	P(9)=	0.2690
PROPORTION OF TIME SYSTEM IS FULL	P(10)=	0.5380
PROBABILITY ARRIVAL WILL NOT HAVE TO WAIT	PNOWT=	0.4620
MEAN NUMBER OF EMPTY CHANNELS	AVEC=	0.7593
MEAN NUMBER OF CUSTOMERS IN SYSTEM	L=	9.2407
MEAN NUMBER OF CUSTOMERS IN QUEUE	LQ=	0.0
MEAN TIME IN SYSTEM	W=	10.0000
MEAN TIME IN QUEUE	WQ=	0.0

```
NEXT
$EXAMPLE NUMBER EIGHT
$MODEL NUMBER SIX
$REFERENCE: PAGE 47, I.B.M. MANUAL.
$TWELVE MACHINES / TWO CHANNELS.
$COMPARE WITH EXAMPLE THREE WITH SIX MACHINES / ONE SERVER.
ARRIVAL:POISSON:TA=60
SERVICE:EXPONENTIAL:TS=12
POPULATION=12
CHANNELS=2
SOLUTION
```

PROBLEM NUMBER 8

DESCRIPTION ANC ANALYSIS CF QUEUEING MODEL TYPE 6

ARRIVAL DISTRIBUTION IS PCISSCN

SCURCE POPULATION IS OF SIZE 12

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 2

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA= 60.0000

MEAN ARRIVAL RATE . . . . LAMBDA= 0.0167

MEAN SERVICE TIME . . . . TS= 12.0000

MEAN SERVICE RATE . . . . MU= 0.0833

TRAFFIC INTENSITY . . . . PSI= 0.2000

SERVER UTILIZATION . . . . RHO= 0.1000

PROPORTION OF TIME SYSTEM IS IDLE . . . . P0= 0.0637
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . . P(1)= 0.1528
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . . P(2)= 0.1681
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . . P(3)= 0.1681
PROB(NO. OF CUSTOMERS IN SYSTEM = 4) . . P(4)= 0.1513
PROB(NO. OF CUSTOMERS IN SYSTEM = 5) . . P(5)= 0.1210
PROB(NO. OF CUSTOMERS IN SYSTEM = 6) . . P(6)= 0.0847
PROB(NO. OF CUSTOMERS IN SYSTEM = 7) . . P(7)= 0.0508
PROB(NO. OF CUSTOMERS IN SYSTEM = 8) . . P(8)= 0.0254
PROB(NO. OF CUSTOMERS IN SYSTEM = 9) . . P(9)= 0.0102
PROB(NO. OF CUSTOMERS IN SYSTEM =10) . . P(10)= 0.0031
PROB(NO. OF CUSTOMERS IN SYSTEM =11) . . P(11)= 0.0006
PROB(NO. OF CUSTOMERS IN SYSTEM =12) . . P(12)= 0.0001

PROBABILITY ARRIVAL WILL NOT HAVE TO WAIT PNOWT= 0.3846

PROBABILITY ARRIVAL WILL HAVE TO WAIT . . . . PD= 0.6154

MEAN DELAY FOR THOSE OBLIGED TO WAIT . . DELAY= 19.0632

INTERARRIVAL TIME PER CUSTOMER . . . . TB= 83.7307

MEAN NUMBER OF EMPTY CHANNELS . . . . AVEC= 0.2802

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 3.4010

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 1.6812

MEAN TIME IN SYSTEM . . . . W= 23.7307

MEAN TIME IN QUEUE . . . . WQ= 11.7307

NEXT PROBLEM


```
$EXAMPLE NUMBER NINE
$MODEL NUMBER SEVEN
$REFERENCE: PAGE 100, HINES.
$THIS MODEL IS A SINGLE CHANNEL MODEL WITH POISSON
$        ARRIVAL AND CONSTANT SERVICE RATES.
$THE REGULAR ASSUMPTIONS ARE MADE.
$THE FOLLOWING ARE EXAMPLES OF VARIOUS SERVICE DISTRIBUTIONS.
$NOTE: SINCE THE VARIANCE OF THE SERVICE TIME DECREASES AS
$        K(THE ERLANG CLASS) INCREASES, THERE IS A
$        CORRESPONDING DECREASE IN CONGESTION DUE TO THE
$        DECREASE IN RANDOMNESS.
$K=1 GIVES THE GREATEST VARIANCE, WHICH IS THE EXPONENTIAL
$        DISTRIBUTION.
$AS K=INFINITY THE VARIANCE EQUALS ZERO, THUS THE
$        SERVICE DISTRIBUTION IS CONSTANT.
A,P,LAM=1.2
SE,CONSTANT,MU=3
SOLUTION
```

PROBLEM NUMBER 9

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 7

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS CONSTANT

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA= 0.8333

MEAN ARRIVAL RATE . . . . LAMBDA= 1.2000

MEAN SERVICE TIME . . . . TS= 0.3333

MEAN SERVICE RATE . . . . MU= 3.0000

SERVER UTILIZATION . . . . RHO= 0.4000

PROPORTION OF TIME SYSTEM IS IDLE . . . . PO= 0.6000

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 0.5333

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 0.1333

MEAN TIME IN SYSTEM . . . . W= 0.4444

MEAN TIME IN QUEUE . . . . WQ= 0.1111

```
$EXAMPLE NUMBER TEN
$MODEL NUMBER EIGHT
$THIS MODEL IS A SINGLE CHANNEL MODEL WITH POISSON
$        ARRIVAL RATE AND ERLANG CLASS K SERVICE RATE.
$ERLANG CLASS 20
SE,ER-20,MU=3
$NOTE: SINCE ONLY THE SERVICE INSTRUCTICN IS CHANGED FRCM
$        THE PREVIOUS PRCBLEM, THAT THE NEXT PROBLEM
$        INSTRUCTICN WAS CMITTED.
SOLVE
```

PROBLEM NUMBER 10

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 8

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS ERLANG OF CLASS 20

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME	TA=	0.8333
MEAN ARRIVAL RATE	LAMBDA=	1.2000
MEAN SERVICE TIME	TS=	0.3333
MEAN SERVICE RATE	MU=	3.0000
SERVER UTILIZATION	RHO=	0.4000
PROPORTION OF TIME SYSTEM IS IDLE	PO=	0.6000
MEAN NUMBER OF CUSTOMERS IN SYSTEM	L=	0.5400
VARIANCE OF NUMBER IN SYSTEM	VARL=	0.6087
90% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN		1.5542
95% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN		2.1003
MEAN NUMBER OF CUSTOMERS IN QUEUE	LQ=	0.1400
MEAN TIME IN SYSTEM	W=	0.4500
VARIANCE OF TIME IN SYSTEM	VARW=	0.0477
90% OF THE TIME, TIME IN SYSTEM IS LESS THAN		0.7339
95% OF THE TIME, TIME IN SYSTEM IS LESS THAN		0.8867
MEAN TIME IN QUEUE	WQ=	0.1167

```
$EXAMPLE NUMBER ELEVEN
$MODEL NUMBER EIGHT
$ERLANG CLASS TEN
SE,ER-10,MU=3.
$NOTE: AGAIN CNLY THE SERVICE INSTRUCTION IS CHANGED.
SOLVE
```

PROBLEM NUMBER 11

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 8

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS ERLANG OF CLASS 10

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA=	0.8333
MEAN ARRIVAL RATE . . . . LAMBDA=	1.2000
MEAN SERVICE TIME . . . . TS=	0.3333
MEAN SERVICE RATE . . . . MU=	3.0000
SERVER UTILIZATION . . . . RHO=	0.4000
PROPORTION OF TIME SYSTEM IS IDLE . . . . PO=	0.6000
MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L=	0.5467
VARIANCE OF NUMBER IN SYSTEM . . . . VARL=	0.6311
90% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN	1.5794
95% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN	2.1355
MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ=	0.1467
MEAN TIME IN SYSTEM . . . . W=	0.4556
VARIANCE OF TIME IN SYSTEM . . . . VARW=	0.0586
90% OF THE TIME, TIME IN SYSTEM IS LESS THAN	0.7704
95% OF THE TIME, TIME IN SYSTEM IS LESS THAN	0.9399
MEAN TIME IN QUEUE . . . . WQ=	0.1222

```
$EXAMPLE NUMBER TWELVE

$MODEL NUMBER EIGHT

$ERLANG CLASS THREE

SERVICE:ERLANG-3:MU=3

SOLVE
```

PROBLEM NUMBER 12

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 8

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS ERLANG OF CLASS 3

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME . . . . TA=	0.8333
MEAN ARRIVAL RATE . . . . LAMBDA=	1.2000
MEAN SERVICE TIME . . . . TS=	0.3333
MEAN SERVICE RATE . . . . MU=	3.0000
SERVER UTILIZATION . . . . RHO=	0.4000
PROPORTION OF TIME SYSTEM IS IDLE . . . . PO=	0.6000
MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L=	0.5778
VARIANCE OF NUMBER IN SYSTEM . . . . VARL=	0.7417
90% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN	1.6974
95% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN	2.3002
MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ=	0.1778
MEAN TIME IN SYSTEM . . . . W=	0.4815
VARIANCE OF TIME IN SYSTEM . . . . VARW=	0.1139
90% OF THE TIME, TIME IN SYSTEM IS LESS THAN	0.9201
95% OF THE TIME, TIME IN SYSTEM IS LESS THAN	1.1563
MEAN TIME IN QUEUE . . . . WQ=	0.1481

```
$EXAMPLE NUMBER THIRTEEN
$MODEL NUMBER ONE
$ERLANG CLASS ONE IS THE SAME AS EXPONENTIAL DISTRIBUTION.
SE,ER-1,MU=3
SOLVE
```

PROBLEM NUMBER 13

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 1

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE FIFO

MEAN INTERARRIVAL TIME	TA=	0.8333
MEAN ARRIVAL RATE	LAMBDA=	1.2000
MEAN SERVICE TIME	TS=	0.3333
MEAN SERVICE RATE	MU=	3.0000
SERVER UTILIZATION	RHO=	0.4000
PROPORTION OF TIME SYSTEM IS IDLE	PO=	0.6000
PROB(NO. OF CUSTOMERS IN SYSTEM = 1) . .	P(1)=	0.2400
PROB(NO. OF CUSTOMERS IN SYSTEM = 2) . .	P(2)=	0.0960
PROB(NO. OF CUSTOMERS IN SYSTEM = 3) . .	P(3)=	0.0384
MEAN NUMBER OF CUSTOMERS IN SYSTEM	L=	0.6667
VARIANCE OF NUMBER IN SYSTEM	VARL=	1.1111
90% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN		1.5129
95% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN		2.2694
MEAN NUMBER OF CUSTOMERS IN QUEUE	LQ=	0.2667
VARIANCE OF NUMBER IN QUEUE	VARLQ=	0.5511
MEAN TIME IN SYSTEM	W=	0.5556
VARIANCE OF TIME IN SYSTEM	VARW=	0.3086
90% OF THE TIME, TIME IN SYSTEM IS LESS THAN		1.2778
95% OF THE TIME, TIME IN SYSTEM IS LESS THAN		1.6667
MEAN TIME IN QUEUE	WQ=	0.2222
VARIANCE OF QUEUE TIME	VARWQ=	0.1975
90% OF THE TIME, TIME IN QUEUE IS LESS THAN		0.7702
95% OF THE TIME, TIME IN QUEUE IS LESS THAN		1.1552

```
    NEXT
$EXAMPLES 14 - 16
$MODEL NUMBER 11 - 13
$REFERENCE: PAGE 25, I.B.M. MANUAL.
$THESE MODELS ARE SINGLE CHANNEL MODELS WITH POISSON
$        ARRIVAL RATES AND VARIOUS SERVICE DISTRIBUTIONS.
$THESE MODELS ARE MULTI-STREAM MODELS WHERE THE FIRST
$        STREAM IS CALLED PRIORITY CLASS ONE, ETC., CLASS
$        ONE HAS PRIORITY OVER CLASS TWO, ETC..
$THESE MODELS LIST RESULTS FOR EACH CLASS AS WELL AS THE
$        TOTAL RESULTS OF THE SYSTEM.
$SERVICE IS FIFO WITHIN A PRIORITY.
$THE HIGHER PRIORITY CLASS (LOWER NUMBER) IS SERVED FIRST.
$THE STREAMS ARE LISTED HIGHER PRIORITY FIRST.
ARRIVAL:POISSON:LAMBDA=.5,.1
SERVICE:CONSTANT:TS=.2
SERVICE:EXPONENTIAL:TS=5.
DISCIPLINE:NONPREEMPTIVE PRIORITY
          SOLVE
```

PRIORITY PROBLEM NUMBER P 1

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 11

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS 1 IS CONSTANT

SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS 2 IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE NONPREEMPTIVE PRIORITY

***** INFORMATION AND RESULTS FOR PRIORITY CLASS 1 *****

MEAN INTERARRIVAL TIME . . . . TA= 2.0000

MEAN ARRIVAL RATE . . . . LAMBDA= 0.5000

MEAN SERVICE TIME . . . . TS= 0.2000

MEAN SERVICE RATE . . . . MU= 5.0000

SERVER UTILIZATION . . . . RHO= 0.1000

CUMULATIVE UTILIZATION OF PRIORITY 1 UP TO 1 IS = 0.1000

TOTAL SERVER UTILIZATION FOR SYSTEM . . . . = 0.6000

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 1.4944

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 1.3944

MEAN TIME IN SYSTEM . . . . W= 2.9889

MEAN TIME IN QUEUE . . . . WQ= 2.7889

```
*****  INFORMATION AND RESULTS FOR PRIORITY CLASS  2  *****

MEAN INTERARRIVAL TIME . . . . . . . . . . . . . . TA=  10.0000

MEAN ARRIVAL RATE  . . . . . . . . . . . . . . LAMBDA=   0.1000

MEAN SERVICE TIME  . . . . . . . . . . . . . . . . TS=   5.0000

MEAN SERVICE RATE  . . . . . . . . . . . . . . . . MU=   0.2000

SERVER UTILIZATION . . . . . . . . . . . . . . .  RHO=   0.5000

CUMULATIVE UTILIZATION OF PRIORITY 1 UP TO  2 IS =   0.6000

TOTAL SERVER UTILIZATION FOR SYSTEM  . . . . . . . =   0.6000

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . . . .  L=   1.1972

MEAN NUMBER OF CUSTOMERS IN QUEUE  . . . . . . . LQ=   0.6972

MEAN TIME IN SYSTEM  . . . . . . . . . . . . . .  W=  11.9722

MEAN TIME IN QUEUE   . . . . . . . . . . . . . . WQ=   6.9722
```

```
*****  TOTAL RESULTS OF SYSTEM  *****

MEAN INTERARRIVAL TIME . . . . . . . . . . . . . . TA=     1.6667

MEAN ARRIVAL RATE  . . . . . . . . . . . . . . LAMBDA=     0.6000

MEAN SERVICE TIME  . . . . . . . . . . . . . . . . TS=     1.0000

MEAN SERVICE RATE  . . . . . . . . . . . . . . . . MU=     1.0000

TOTAL SERVER UTILIZATION FOR SYSTEM  . . . . . . . =       0.6000

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . . . .  L=       2.6917

MEAN NUMBER OF CUSTOMERS IN QUEUE  . . . . . . . LQ=       2.0917

MEAN TIME IN SYSTEM  . . . . . . . . . . . . . .  W=       4.4861

MEAN TIME IN QUEUE   . . . . . . . . . . . . . . WQ=       3.4861
```

```
$CHANGE PRIORITY TO PREEMPTIVE-RESUME.
$IN THIS MODEL, ARRIVALS WITH A HIGHER PRIORITY INTERRUPT
$        (PREEMPT) SERVICE OF A LOWER CLASS PRIORITY CUSTOMER.
$AN INTERRUPTED CUSTOMER WILL CONTINUE SERVICE WHERE
$        IT LEFT OFF WHEN IT BECOMES THE HIGHEST PRIORITY
$        CLASS CUSTOMER IN THE SYSTEM.
DISCIPLINE:PREEMPTIVE
SOLVE
```

PRIORITY PROBLEM NUMBER P 2

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 12

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS 1 IS CONSTANT

SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS 2 IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE PREEMPTIVE-RESUME PRIORITY

***** INFORMATION AND RESULTS FOR PRIORITY CLASS 1 *****

MEAN INTERARRIVAL TIME . . . . TA= 2.0000

MEAN ARRIVAL RATE . . . . LAMBDA= 0.5000

MEAN SERVICE TIME . . . . TS= 0.2000

MEAN SERVICE RATE . . . . MU= 5.0000

SERVER UTILIZATION . . . . RHO= 0.1000

CUMULATIVE UTILIZATION OF PRIORITY 1 UP TO 1 IS = 0.1000

TOTAL SERVER UTILIZATION FOR SYSTEM . . . . = 0.6000

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 0.1056

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 0.0056

MEAN TIME IN SYSTEM . . . . W= 0.2111

MEAN TIME IN QUEUE . . . . WQ= 0.0111

***** INFORMATION AND RESULTS FCR PRIORITY CLASS 2 *****

MEAN INTERARRIVAL TIME . . . . TA= 10.0000

MEAN ARRIVAL RATE . . . . LAMBDA= 0.1000

MEAN SERVICE TIME . . . . TS= 5.0000

MEAN SERVICE RATE . . . . MU= 0.2000

SERVER UTILIZATION . . . . RHO= 0.5000

CUMULATIVE UTILIZATION CF PRIORITY 1 UP TO 2 IS = 0.6000

TOTAL SERVER UTILIZATION FOR SYSTEM . . . . = 0.6000

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 1.2528

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 0.7528

MEAN TIME IN SYSTEM . . . . W= 12.5278

MEAN TIME IN QUEUE . . . . WQ= 7.5278

***** TOTAL RESULTS OF SYSTEM *****

MEAN INTERARRIVAL TIME	TA=	1.6667
MEAN ARRIVAL RATE	LAMBDA=	0.6000
MEAN SERVICE TIME	TS=	1.0000
MEAN SERVICE RATE	MU=	1.0000
TOTAL SERVER UTILIZATION FOR SYSTEM	=	0.6000
MEAN NUMBER OF CUSTOMERS IN SYSTEM	L=	1.2528
MEAN NUMBER OF CUSTOMERS IN QUEUE	LQ=	0.6528
MEAN TIME IN SYSTEM	W=	2.0880
MEAN TIME IN QUEUE	WQ=	1.0880

```
$CHANGE THE PRICRITY MODEL TO A NO PRIORITY MODEL STILL
$        CONTAINING MORE THAN ONE ARRIVAL STREAM.
$SERVICE DISCIPLINE IS FIFO.
CISCIPLINE:NC PRIORITY
SOLVE
```

PRIORITY PROBLEM NUMBER P 3

DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE 13

ARRIVAL DISTRIBUTION IS POISSON

SOURCE POPULATION IS CONSIDERED TO BE INFINITE

SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS 1 IS CONSTANT

SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS 2 IS EXPONENTIAL

NUMBER OF SERVICE CHANNELS IN PARALLEL = 1

QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED

QUEUE DISCIPLINE IS ASSUMED TO BE NONPRIORITY

***** INFORMATION AND RESULTS FOR PRIORITY CLASS 1 *****

MEAN INTERARRIVAL TIME . . . . TA= 2.0000

MEAN ARRIVAL RATE . . . . LAMBDA= 0.5000

MEAN SERVICE TIME . . . . TS= 0.2000

MEAN SERVICE RATE . . . . MU= 5.0000

SERVER UTILIZATION . . . . RHO= 0.1000

CUMULATIVE UTILIZATION OF PRIORITY 1 UP TO 1 IS = 0.1000

TOTAL SERVER UTILIZATION FOR SYSTEM . . . . = 0.6000

MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . L= 3.2375

MEAN NUMBER OF CUSTOMERS IN QUEUE . . . . LQ= 3.1375

MEAN TIME IN SYSTEM . . . . W= 6.4750

MEAN TIME IN QUEUE . . . . WQ= 6.2750

***** INFORMATION AND RESULTS FOR PRIORITY CLASS 2 *****

```
MEAN INTERARRIVAL TIME . . . . . . . . . . . . . . . TA=  10.0000
MEAN ARRIVAL RATE  . . . . . . . . . . . . . . . LAMBDA=   0.1000
MEAN SERVICE TIME  . . . . . . . . . . . . . . . . . TS=   5.0000
MEAN SERVICE RATE  . . . . . . . . . . . . . . . . . MU=   0.2000
SERVER UTILIZATION . . . . . . . . . . . . . . . .  RHO=   0.5000
CUMULATIVE UTILIZATION OF PRIORITY 1 UP TO  2 IS =   0.6000
TOTAL SERVER UTILIZATION FOR SYSTEM  . . . . . . . =   0.6000
MEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . . . .   L=   1.1275
MEAN NUMBER OF CUSTOMERS IN QUEUE  . . . . . . . LQ=   0.6275
MEAN TIME IN SYSTEM  . . . . . . . . . . . . . . .  W=  11.2750
MEAN TIME IN QUEUE   . . . . . . . . . . . . . . . WQ=   6.2750
```

***** TOTAL RESULTS OF SYSTEM *****

MEAN INTERARRIVAL TIME	TA=	1.6667
MEAN ARRIVAL RATE	LAMBDA=	0.6000
MEAN SERVICE TIME	TS=	1.0000
MEAN SERVICE RATE	MU=	1.0000
TOTAL SERVER UTILIZATION FOR SYSTEM	=	0.6000
MEAN NUMBER OF CUSTOMERS IN SYSTEM	L=	4.3650
MEAN NUMBER OF CUSTOMERS IN QUEUE	LQ=	3.7650
MEAN TIME IN SYSTEM	W=	7.2750
MEAN TIME IN QUEUE	WQ=	6.2750

```
$END OF DATA INSTRUCTION
$THIS IS LAST INSTRUCTION READ INCLUDING COMMENTS.
$AT THE END OF THE OUTPUT WILL BE A SUMMARY OF THE OUTPUT
$        FOR THE INDIVIDUAL MODELS DESCRIBED BY THE USER.
$A SUMMARY WILL ONLY BE PROVIDED IF MORE THAN ONE PROBLEM
$        OF EITHER THE SINGLE OR MULTI-STREAM MODELS
$        HAS BEEN SOLVED.
END
```

*** SUMMARY RESULTS OF SINGLE STREAM MODELS ***

PROBLEM NUMBER	PROB. SYSTEM IS EMPTY	MEAN NUMBER IN SYSTEM	MEAN NUMBER IN QUEUE	MEAN TIME IN SYSTEM	MEAN TIME IN QUEUE
1	0.200	4.000	3.200	0.200	0.160
2	0.178	2.420	1.598	0.303	0.181
3	0.192	1.959	1.151	29.092	17.092
4	0.045	6.011	3.511	12.022	7.022
5	0.006	5.810	0.810	34.862	4.862
6	0.013	5.919	1.935	17.827	5.827
7	0.000	9.241	0.0	10.000	0.0
8	0.064	3.401	1.681	23.731	11.731
9	0.600	0.533	0.133	0.444	0.111
10	0.600	0.540	0.140	0.450	0.117
11	0.600	0.547	0.147	0.456	0.122
12	0.600	0.578	0.178	0.481	0.148
13	0.600	0.667	0.267	0.556	0.222

*** SUMMARY RESULTS OF MULTI-STREAM MODELS ***

PROBLEM NUMBER	MEAN NUMBER IN SYSTEM	MEAN NUMBER IN QUEUE	MEAN TIME IN SYSTEM	MEAN TIME IN QUEUE
1	2.692	2.092	4.486	3.486
2	1.253	0.653	2.088	1.088
3	4.365	3.765	7.275	6.275

Chapter 4

THE QUEUE INTERPRETER

4.1 General Description and Main Program

The interpreter consists of a main program which reads the input from the user, interprets the instructions, and drives the other subprograms. The interpretation and execution proceed at the same time.

The main program recognizes the commands, prints the input, sets various flags for the description of a queueing model, calls on subroutine EVALUE for the evaluation of numerical information, calls on subroutine SOLVE to determine the type of model being described by the user, and finally calls on subroutines TABOUT and POUT for a summary of the results of all models in the users program. The alphabetic characters needed for interpretation are first put in as BLOCK DATA, and all pertinent variables are initialized. The program reads only one card per command.

The characters of an input instruction are put in a vector L which is 80 elements in length. The main program then calls subroutine BLANK to remove the blanks and left justifies the vector L. The program will scan the L vector to determine the type of instruction found. If the first character is a dollar sign or a blank, the statement is a comment or a blank card and is printed and ignored. Otherwise the first few characters will determine the type of instruction given. At this point the main program is divided into sections according to the type of instruction. The program will then continue to scan the L vector until a delimiter character is found. This is generally any of the characters comma, colon, or equal sign. It is at this point that the various sections differ slightly in procedure.

The arrival instruction statement will be searched to determine the type of arrival, whether the interarrival times or the arrival rates are given, and the values of this parameter must then be evaluated and put into a vector TA or LAMBDA.

The service instruction statement is similar to the arrival instruction with the values being assigned to the variables TS or MU and to the vectors TTS and TMU. The variables TS and MU are used in the single-stream models in place of the corresponding vectors. There may be more than one service instruction in the multi-stream models if the service disciplines are different for some stream or priorty class. There can be only one arrival instruction since there is only one discipline allowed at this time, but there must be a value supplied for the parameter for each input stream.

The channel size instruction will be searched for either a number or the word FIND. In the first case the number is evaluated and the value stored in the variable ICHANS.

The next instruction is then read. The latter case will cause the program to scan L until another delimiter character is located. In this case, the next character after the delimiter must be the first character of a numerical field ending with a % sign. After the % sign, the next two characters in the L vector must be a delimiter followed by the letter T. This is followed by another delimiter character and a numerical field for the variable T. All numerical fields are evaluated by the subroutine EVALUE.

The queue capacity instruction statement is scanned until a delimiter character is located. The next character will either be part of an alphabetic field or a numerical field. If the field is alphabetic the capacity is assumed to be unlimited. Otherwise the field is evaluated with the value stored in the ICAP vector. ICAP=-1 means unlimited capacity.

The population instruction is similar to the queue capacity instruction with the vector IPOP used.

The queue discipline instruction is scanned until a delimiter character is found. The next field will be alphabetic and will be scanned to determine the type of discipline specified by the user. The flag IDISC is used to indicate this discipline.

The solve instruction statement causes the main program to call subroutine SOLVE. When control returns to the main program the proper problem counter IPROB or IPNUM is indexed by one and certain variables may be re-initialized. Then the next instruction is read.

The next problem instruction causes the main program to re-initialize all pertinent variables and to read the next instruction.

After reading a card the error flag N is checked to determine if there had been an error in the last set of instructions. If the flag is equal to -2, indicating there has been an error, then all instructions are ignored until the next problem instruction statement.

4.1.1 Subroutine EVALUE

The EVALUE subroutine interprets numbers in alpha format and converts same to decimal number. This decimal number is stored as a real number in the vector VALUE. The argument POINT is used as a pointer for the location of the number field within the vector L. When entering EVALUE, POINT is at one character before the first digit of the number field. When leaving EVALUE, POINT is one character past the last digit of the number field. The argument NUM is used to determine the position the number value is stored in the vector VALUE. This is always equal to one except when the multi-stream models are used. The argument KK is used to determine if more than one number field may be found, as in the multi-stream models.

The subroutine determines the width of the number field and the location of the decimal point if any. The field is divided into two parts, the whole number part or left field, and the fractional part or right field. Alpha format numerical characters are converted one digit at a time by calling subroutine DECODE. After conversion the value is constructed into a decimal number. When the construction is complete the number is stored in the vector VALUE and the next number field is converted if applicable. After all consecutive number fields are converted control is returned to the main program.

4.1.2 Subroutine DECODE(M,D)

Subroutine DECODE determines digit M in alpha format and returns a decimal value assigned to the argument D. If no digit is found, a value of 10.0, indicating an error, is returned to subroutine EVALUE.

4.1.3 Subroutine BLANK(IN,N)

Subroutine BLANK takes blanks out of the input string and left justifies the vector IN. The length of the vector IN is returned with the value N.

4.1.4 Subroutine BAD(N)

Subroutine BAD prints the error messages and returns a value of -2 or remains -1 if the error was that the model is not available. In this case there is no need to skip to the NEXT problem instruction since there has been no error in interpretation.

4.1.5 Subroutine SOLVE

Subroutine SOLVE checks various flags to determine type of model being described by the user. If the model is determined to be one of the single-stream models available then that model subroutine is called. If the model is one of the multi-stream models available then the subroutine MOMENT is called. If the model is not available then N is set equal to -1 and subroutine BAD is called, after this control is returned to the main program.

4.1.6 Function FACT(N)

The integer function FACT calculates the factorial of N.

4.1.7 Subroutine OUTPUT(K)

Subroutine OUTPUT prints a description of the problem and the results calculated from the various models. The argument K is used to indicate the model number.

4.1.8 Subroutine MOMENT

Subroutine MOMENT is called from subroutine SOLVE when multi-stream service distributions are specified. The subroutine calculates the first and second moments about the mean for each class of service distribution. The moments for the overall service time is also calculated. The total arrival rate, server utilization per class, and cumulative utilization to each class are also calculated in this subroutine. Subroutine MODL11, MODL12, or MODL13 is called depending on the type of discipline used, nonpreemptive, preemptive, preemptive-resume, or no-priority respectfully.

4.1.9 Subroutines MODL1-MODL8, and MODL11-MODL13

Subroutines MODL are a series of subroutines that calculate various measures of congestion and other useful information that will be printed by subroutine OUTPUT.

4.1.10 Subroutines PTAB and TABU

Subroutines PTAB and TABU tabulate common results of the multi-stream and single-stream models. The values of each problem is put in the appropiate table of arrays.

4.1.11 Subroutines POUT and OUTTAB

Subroutines POUT and OUTTAB are used to output the values tabulated from subroutines PTAB and TABU. This output will provide a convenient summary of results. This output is provided at the end of any program containing more than one problem of either the multi- or single-stream models.

4.1.12 BLOCK DATA

BLOCK DATA is used to initialize all pertinent variables. This is used mainly to give an integer value to alpha characters for use in numerical comparisons in the interpretation procedure.

Chapter 5

SUMMARY, CONCLUSION, AND RECOMMENDATIONS

5.1 Summary

A user oriented computer language was developed for the solution of basic queueing theory models. To avoid the burden of machine dependence, the QUEUE interpreter was coded in a language which is very much universal, and it is small enough to not be restricted to large computer systems. The language allows the user to solve certain queueing models with only a minimal amount of time required to learn the language. The language is readily usable in a conversational mode as well as in a batch mode. The language is currently on the University of Houston Computing Center's Univac 1108 System and the Engineering Systems Simulation Laboratory's I.B.M. 360/44 System. An example of the problems that can be solved using the language have also been presented.

5.2 Conclusions

The QUEUE language allows the user to easily describe and modify a queueing model. The output of the QUEUE language provides considerable information that can provide insight to even an inexperienced user. This information is quite useful in the design of a queueing system.

In order to facilitate the further expansion of QUEUE, the interpreter has been coded in such a way that any necessary changes, such as addition of new models, can be made without a significant number of alterations in the program.

5.3 Recommendations

It is hoped that QUEUE will be a useful tool in the hands of a non-computer oriented operation research analyst to utilize the computer for solving queueing theory problems. It is also hoped that QUEUE can be found as an effective tool for educational purposes.

BIBLIOGRAPHY

1. Donaghey, Charles E., and Osman S. Ozkul. STIL Systems Manual. Technical report number RE 12-69, Office of Naval Research Publication, September, 1969.

2. Hellerman, Herbert. Digital Computer System Principles. New York: McGraw Hill Book Company, 1967.

3. Hillier, Frederick S., and Gerald J. Lieberman. Introduction to Operations Research. San Francisco: Holden-Day, Inc., 1967.

4. I.B.M. Data Processing Techniques, Analysis of Some Queuing Models in Real-Time Systems, Form F20-0007-1. White Plains, New York: I.B.M.

5. Jaiswal, N. K. Priority Queues. New York: Academic Press, 1968.

6. McCracken, Danial D. A Guide to FORTRAN IV programming. New York: John Wiley and Sons, Inc., 1965.

7. Meyer, Paul L. Introductory Probability and Statistical Applications. Reading, Massachusetts: Addison-Wesley Publishing Company, 1970.

8. Morris, William T. Analysis of Materials Handling Management. Homewood, Illinois: Richard D. Irwin, Inc., 1962.

9. Morse, Philip M. Queues, Inventories and Maintenance. New York: John Wiley and Sons, Inc., 1958. Derives simple queueing problems in clear, simple terms.

10. Pollack, Bary W. (ed.). Compiler Techniques. Princeton: Auerbach Publishers Inc., 1972.

11. Ruiz-Palá, Ernesto, Carlos Àvila-Beloso, and William W. Hines. Waiting Line Models. New York: Reinhold Publishing Corporation, 1967. Contains many example problems and graphs illustrating results of various models.

12. Saaty, T. L. Elements of Queuing Theory. New York: McGraw-Hill, 1961. Collection of techniques and results for queueing problems, including many fairly recent developments; difficult to read in many places because of undigested survey of literature; contains excellent bibliography of papers.

13. Takacs, L. "Priority Queues," Operations Research, vol. 12, no. 1, 1964.

APPENDIX A

A Listing of QUEUE

```
C***  MAIN PROGRAM                                                      MAIN  10
      DATA ISIGN/'$'/                                                   MAIN  20
C***  COMMON BLOCKS 1 - 5 ARE LITERAL CONSTANTS                         MAIN  30
      COMMON/BLK1/ JAC,JBC,JCC,JDC,JEC,JFC,JGC,JHC,JIC,JJC,JKC,JLC,JMC  MAIN  40
      COMMON/BLK2/ JNC,JOC,JPC,JQC,JRC,JSC,JTC,JUC,JVC,JWC,JXC,JYC,JZC  MAIN  50
      COMMON/BLK3/ J0C,J1C,J2C,J3C,J4C,J5C,J6C,J7C,J8C,J9C              MAIN  60
      COMMON/BLK4/ JPLUS,JMINUS,JPER,JDIV,JMUL,JEQU,JBLANK,JCOMMA,JCOLONMAIN  70
      COMMON/BLK5/JPCENT                                                MAIN  80
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND                       MAIN  90
      COMMON/BLK7/ ARIVAL,KPRIOR,IDISC,FSERV,VAR,ERLANG                 MAIN 100
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU                 MAIN 110
      COMMON / BLK18 / ITYPE(10),ILANG(10),TTS(10),TMU(10),TVAR(10)     MAIN 120
      COMMON/BLK20/NUMS                                                 MAIN 130
      COMMON/BLK25/N                                                    MAIN 140
      COMMON / BLK26 /IPNUM                                             MAIN 150
C***  VARIABLE     TYPE     SIZE     COMMON BLOCK      COMMENT          MAIN 160
C***  PRCENT       R        1        6                 % VALUE          MAIN 170
C***  IPOP         I        1        6                 POPULATION SIZE  MAIN 180
C***  ICHANS       I        1        6                 NO. OF CHANNELS  MAIN 190
C***  T            R        1        6                 TIME             MAIN 200
C***  ICAP         I        1        6                 CAPACITY         MAIN 210
C***  IFIND        I        1        6                 FIND INSTRUCTION MAIN 220
C***  ARIVAL       I        1        7                 ARRIVAL FLAG     MAIN 230
C***  KPRIOR       I        1        7                 PRIORITY FLAG    MAIN 240
C***  IDISC        I        1        7                 DISCIPLINE FLAG  MAIN 250
C***  FSERV        I        1        7                 SERVICE FLAG     MAIN 260
C***  VAR          R        1        7                 VARIANCE OF TS   MAIN 270
C***  ERLANG       I        1        7                 CLASS OF ERLANG  MAIN 280
C***  TA           R        10       8                 INTERARRIVAL TIME MAIN 290
C***  TS           R        1        8                 SERVICE TIME     MAIN 300
C***  NUM          I        1        8                 NUMBER OF VALUES MAIN 310
C***                                                   RETURNED FROM EVALMAIN 320
C***  IPROB        I        1        8                 PROBLEM NUMBER   MAIN 330
C***  LAMBDA       R        10       8                 ARRIVAL RATE     MAIN 340
C***  MU           R        1        8                 SERVICE RATE     MAIN 350
C***  THE FOLLOWING ARE USED IN THE MULTI STREAM MODELS                 MAIN 360
```

```
C***  ITYPE          I          10        18                  SERVICE TYPE        MAIN 370
C***  ILANG          I          10        18                  CLASS OF ERLANG     MAIN 380
C***  TTS            R          10        18                  SERVICE TIME        MAIN 390
C***  TMU            R          10        18                  SERVICE RATE        MAIN 400
C***  TVAR           R          10        18                  VARIANCE OF TTS     MAIN 410
C***  NUMS           I          1         20                  SERVICE CLASS DISTMAIN 420
C***  NUMV           I          1                             VARIANCE INDEX      MAIN 430
C***  N              I          1         25                  ERROR NUMBER        MAIN 440
      INTEGER FSERV,SVAL,ERLANG,AVAL                                              MAIN 450
      INTEGER ARIVAL,POINT                                                        MAIN 460
      REAL LAMBDA,MU                                                              MAIN 470
      DIMENSION L(80),VALUE(10)                                                   MAIN 480
C***  KPRIOR INDICATES IF PRIORITY CLASSES ARE TO BE USED , 0=NO , 1=YESMAIN 490
C**** INITIAL PARAMETERS TO -1                                                    MAIN 500
      TS=-1.                                                                      MAIN 510
      MU=-1.                                                                      MAIN 520
      DO 11 I=1,10                                                                MAIN 530
      VALUE(I)=0.                                                                 MAIN 540
      TTS(I)=-1.                                                                  MAIN 550
      TMU(I)=-1.                                                                  MAIN 560
      TVAR(I)=-1.                                                                 MAIN 570
      ILANG(I)=0                                                                  MAIN 580
      TA(I)=-1.                                                                   MAIN 590
      LAMBDA(I)=-1.                                                               MAIN 600
   11 CONTINUE                                                                    MAIN 610
C***  IPROB IS THE PROBLEM NUMBER                                                 MAIN 620
      IPROB=1                                                                     MAIN 630
C***  IPNUM IS PRIORITY PROBLEM NUMBER                                            MAIN 640
      IPNUM=1                                                                     MAIN 650
C                                                                                 MAIN 660
C     ------------------------------------------------------------------------MAIN 670
C                                                                                 MAIN 680
    1 CONTINUE                                                                    MAIN 690
      VAR=0.0                                                                     MAIN 700
      KPRIOR=0                                                                    MAIN 710
C***  FSERV INDICATES TYPE OF SERVICE DISCIPLINE                                  MAIN 720
C***  FSERV=1 IS EXP, =2 IS GENERAL, =3 IS CONSTANT, =4 IS ERLANG                 MAIN 730
```

```
      FSERV=0                                                           MAIN 740
C***  ERLANG INDICATES CLASS OF ERLANG DISTRIBUTION                     MAIN 750
      ERLANG=0                                                          MAIN 760
C***  AVAL=1  MEANS INTERARRIVAL TIME IS BEING USED BY THE USER         MAIN 770
C***  AVAL=2  MEANS ARRIVAL RATE IS USED                                MAIN 780
C***  IF SERVICE TIME IS USED THEN SVAL=1                               MAIN 790
C***  IF SERVICE RATE IS USED THEN SVAL=2                               MAIN 800
      SVAL=0                                                            MAIN 810
C***  KK INDICATES IF MORE THAN ONE VALUE CAN BE FOUND IN SUB. EVALUE   MAIN 820
C***  KK = 1 MEANS ONLY ONE VALUE IS ALLOWED , KK=2 MEANS MORE THAN ONE MAIN 830
      KK=1                                                              MAIN 840
C***  N INDICATE ERROR NUMBER      N = 0  UNLESS OTHERWISE INDICATED    MAIN 850
      N=0                                                               MAIN 860
C***  PRCENT INDICATES PERCENTAGE VALUE IN CHANNELS INSTRUCTION         MAIN 870
      PRCENT=0.                                                         MAIN 880
C***  T INDICATES MAXIMUM WAITING TIME TO PROVIDE REQUIRED SERVICE      MAIN 890
      T=0.                                                              MAIN 900
C***  ICHANS INDICATES NUMBER OF PARALLEL CHANNELS                      MAIN 910
      ICHANS=1                                                          MAIN 920
C***  IFIND INDICATES IF NUMBER OF CHANNELS IS TO BE SOLVED FOR , 0=NO  MAIN 930
      IFIND=0                                                           MAIN 940
C***  IDISC INDICATES TYPE OF DISCIPLINE USED                           MAIN 950
C***  1=FIFO, 2=LIFO, 3=RANDOM, 4=NONPREEMPTIVE, 5=PRE, 6=NO PRIORITY   MAIN 960
      IDISC=1                                                           MAIN 970
C***  IPOP INDICATES SIZE OF SOURCE POPULATION, IPOP=-1 MEANS UNLIMITED MAIN 980
      IPOP=-1                                                           MAIN 990
C***  ICAP INDICATES SIZE OF QUEUE CAPACITY    ICAP=-1 MEANS INFINITE   MAIN1000
      ICAP=-1                                                           MAIN1010
    2 CONTINUE                                                          MAIN1020
      NUMS=0                                                            MAIN1030
      NUMV=0                                                            MAIN1040
    3 CONTINUE                                                          MAIN1050
C     ------------------------------------------------------------------MAIN1060
C                                                                       MAIN1070
C                                                                       MAIN1080
C***  READ NEXT INSTRUCTION CARD                                        MAIN1090
C                                                                       MAIN1100
```

```
      C                                                                 MAIN1110
   10 CONTINUE                                                          MAIN1120
      IF(N.GT.0) CALL BAD(N)                                            MAIN1130
      READ(5,12,END=99) (L(I),I=1,80)                                   MAIN1140
   12 FORMAT(80A1)                                                      MAIN1150
C***  L(80) CONTAINS THE INPUT INSTRUCTION IN ALPHA FORMAT               MAIN1160
      WRITE(6,13)(L(I),I=1,80)                                          MAIN1170
   13 FORMAT(1H0,10X,80A1)                                              MAIN1180
C***  REMOVE BLANKS AND LEFT JUSTIFY                                     MAIN1190
C***  J IS THE LENGTH OF INPUT STRING AFTER REMOVING BLANKS              MAIN1200
      CALL BLANK(L,J)                                                   MAIN1210
C***  IF BLANK OR COMMENT CARD, READ NEXT CARD                           MAIN1220
      IF(J.EQ.0) GO TO 10                                               MAIN1230
      IF(L(1).EQ.ISIGN) GO TO 10                                        MAIN1240
C***  CHECK INSTRUCTION TYPE                                             MAIN1250
C***  NEXT PROBLEM INSTRUCTION                                           MAIN1260
      IF(L(1).EQ.JNC.AND.L(2).EQ.JEC) GO TO 88                          MAIN1270
C***  IF LAST PROBLEM HAD AN ERROR THEN SKIP TO NEXT INSTRUCTION         MAIN1280
      IF( N .EQ. -2 ) GO TO 10                                          MAIN1290
C***  SOLVE INSTRUCTION                                                  MAIN1300
      IF(L(1).EQ.JSC.AND.L(2).EQ.JOC.AND.L(3).EQ.JLC) GO TO 87          MAIN1310
C***  ARRIVAL                                                            MAIN1320
      IF(L(1).EQ.JAC) GO TO 16                                          MAIN1330
C***  SERVICE                                                            MAIN1340
      IF(L(1).EQ.JSC) GO TO 32                                          MAIN1350
C***  CHANNEL SIZE                                                       MAIN1360
      IF(L(1).EQ.JCC.AND.L(2).EQ.JHC) GO TO 56                          MAIN1370
C***  CAPACITY OF QUEUE                                                  MAIN1380
      IF(L(1).EQ.JCC.AND.L(2).EQ.JAC) GO TO 65                          MAIN1390
C***  QUEUE DISCIPLINE TYPE                                              MAIN1400
      IF(L(1).EQ.JDC) GO TO 71                                          MAIN1410
C***  POPULATION SIZE                                                    MAIN1420
      IF(L(1).EQ.JPC) GO TO 82                                          MAIN1430
C***  END OF DATA INSTRUCTION                                             MAIN1440
      IF(L(1).EQ.JEC) GO TO 99                                          MAIN1450
      N=1                                                               MAIN1460
      GO TO 10                                                          MAIN1470
```

```
C     ------------------------------------------------------------------MAIN1480
C                                                                       MAIN1490
C                                                                       MAIN1500
C***  ARRIVAL INSTRUCTION  ***                                           MAIN1510
C                                                                       MAIN1520
C                                                                       MAIN1530
   16 CONTINUE                                                          MAIN1540
      AVAL=0                                                            MAIN1550
      DO 17 I=2,J                                                       MAIN1560
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JEQU) GO TO 18     MAIN1570
   17 CONTINUE                                                          MAIN1580
      N=2                                                               MAIN1590
      GO TO 10                                                          MAIN1600
C***  JJ IS USED AS A POINTER TO SCAN THE L STRING                       MAIN1610
   18 JJ=I+1                                                            MAIN1620
C***  CHECK TYPE OF ARRIVAL DISTRIBUTION                                 MAIN1630
      IF(L(JJ).EQ.JPO) GO TO 20                                         MAIN1640
      N=3                                                               MAIN1650
      GO TO 10                                                          MAIN1660
C***  ARRIVAL IS POISSON                                                 MAIN1670
C***  SET ARIVAL=1                                                       MAIN1680
   20 ARIVAL=1                                                          MAIN1690
      JJ=JJ+1                                                           MAIN1700
      DO 21 I=JJ,J                                                      MAIN1710
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JEQU) GO TO 22     MAIN1720
   21 CONTINUE                                                          MAIN1730
      N=2                                                               MAIN1740
      GO TO 10                                                          MAIN1750
   22 JJ=I+1                                                            MAIN1760
C***  CHECK IF ARRIVAL RATE OR INTERARRIVAL TIME IS GIVEN                 MAIN1770
      IF(L(JJ).EQ.JTC.AND.L(JJ+1).EQ.JAC) AVAL=1                        MAIN1780
      IF(L(JJ).EQ.JLC.AND.L(JJ+1).EQ.JAC.AND.L(JJ+2).EQ.JMC) AVAL=2     MAIN1790
      IF(AVAL.GT.0) GO TO 23                                            MAIN1800
      N=4                                                               MAIN1810
      GO TO 10                                                          MAIN1820
   23 CONTINUE                                                          MAIN1830
      JJ=JJ+2                                                           MAIN1840
```

```
C***  SEARCH FOR DELIMITER CHARACTER                                    MAIN1850
      DO 24 I=JJ,J                                                      MAIN1860
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JEQU) GO TO 25     MAIN1870
   24 CONTINUE                                                          MAIN1880
      N=2                                                               MAIN1890
      GO TO 10                                                          MAIN1900
C***  POINTS IS A POINTER AND IS NOW AT = SIGN                          MAIN1910
   25 POINT=I                                                           MAIN1920
      KK=2                                                              MAIN1930
C***  IF MORE THAN ONE NUMBER IS PERMITTED   KK=2                       MAIN1940
C***  CALL SUBROUTINE EVALUE  TO EVALUATE                               MAIN1950
      CALL EVALUE(POINT,VALUE,KK,NUM,L)                                 MAIN1960
      IF( N .EQ. -2 ) GO TO 10                                          MAIN1970
      IF(AVAL.EQ.1) GO TO 26                                            MAIN1980
      IF(AVAL.EQ.2) GO TO 28                                            MAIN1990
      N=4                                                               MAIN2000
      GO TO 10                                                          MAIN2010
   26 CONTINUE                                                          MAIN2020
C***  INTERARRIVAL TIMES USED                                           MAIN2030
      DO 27 II=1,NUM                                                    MAIN2040
      TA(II)=VALUE(II)                                                  MAIN2050
      LAMBDA(II)=1.0/VALUE(II)                                          MAIN2060
   27 CONTINUE                                                          MAIN2070
      GO TO 30                                                          MAIN2080
   28 CONTINUE                                                          MAIN2090
C***  ARRIVAL RATE IS USED                                              MAIN2100
      DO 29 II=1,NUM                                                    MAIN2110
      LAMBDA(II)=VALUE(II)                                              MAIN2120
      TA(II)=1.0/VALUE(II)                                              MAIN2130
   29 CONTINUE                                                          MAIN2140
   30 CONTINUE                                                          MAIN2150
C***  IF MORE THAN ONE VALUE IS GIVEN THIS INDICATES PRIORITIES ARE USEDMAIN2160
      IF(NUM.GT.1) KPRIOR=1                                             MAIN2170
      GO TO 10                                                          MAIN2180
C     ------------------------------------------------------------------MAIN2190
C                                                                       MAIN2200
C                                                                       MAIN2210
```

```
C***  SERVICE INSTRUCTION  ***                                          MAIN2220
C                                                                       MAIN2230
C                                                                       MAIN2240
   32 CONTINUE                                                          MAIN2250
      NUMS=NUMS+1                                                       MAIN2260
      NUMV=NUMS                                                         MAIN2270
      DO 33 I=2,J                                                       MAIN2280
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JEQU) GO TO 34     MAIN2290
   33 CONTINUE                                                          MAIN2300
      N=2                                                               MAIN2310
      GO TO 10                                                          MAIN2320
   34 JJ=I+1                                                            MAIN2330
C***  CHECK TYPE OF SERVICE                                              MAIN2340
      IF(L(JJ).EQ.JEC.AND.L(JJ+1).EQ.JRC) GO TO 38                      MAIN2350
      IF(L(JJ).EQ.JEC) GO TO 35                                         MAIN2360
      IF(L(JJ).EQ.JGO.OR.L(JJ+1).EQ.JAO) GO TO 36                       MAIN2370
      IF(L(JJ).EQ.JCC) GO TO 37                                         MAIN2380
      N=3                                                               MAIN2390
      GO TO 10                                                          MAIN2400
C***  EXPONENTIAL SERVICE                                                MAIN2410
   35 FSERV=1                                                           MAIN2420
      GO TO 42                                                          MAIN2430
C***  GENERAL OR ARBITRARY                                               MAIN2440
   36 FSERV=2                                                           MAIN2450
      GO TO 42                                                          MAIN2460
C***  CONSTANT                                                           MAIN2470
   37 FSERV=3                                                           MAIN2480
      GO TO 42                                                          MAIN2490
C***  ERLANG CLASS L                                                     MAIN2500
   38 FSERV=4                                                           MAIN2510
      JJ=JJ+2                                                           MAIN2520
      DO 39 I=JJ,J                                                      MAIN2530
      IF(L(I).EQ.JMINUS) GO TO 40                                       MAIN2540
   39 CONTINUE                                                          MAIN2550
      N=2                                                               MAIN2560
      GO TO 10                                                          MAIN2570
   40 CONTINUE                                                          MAIN2580
```

```
      KK=1
      JJ=I
      POINT=I
      CALL EVALUE(POINT,VALUE,KK,NUM,L)
      IF( N .EQ. -2 ) GO TO 10
      ERLANG=VALUE(NUM)
      IF(ERLANG.NE.1) GO TO 42
      FSERV=1
   42 JJ=JJ+1
      DO 43 I=JJ,J
C***  SEARCH FOR DELIMITER CHARACTER
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JEQU) GO TO 44
   43 CONTINUE
      N=2
      GO TO 10
   44 JJ=I+1
C***  CHECK IF SERVICE TIME OR RATE IS GIVEN
      IF(L(JJ).EQ.JTC.AND.L(JJ+1).EQ.JSC)  SVAL=1
      IF(L(JJ).EQ.JMC.AND.L(JJ+1).EQ.JUC)  SVAL=2
      IF(SVAL.GT.0) GO TO 45
      N=4
      GO TO 10
   45 CONTINUE
      JJ=JJ+2
      DO 46 I=JJ,J
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JEQU) GO TO 47
   46 CONTINUE
      N=2
      GO TO 10
C***  POINT IS AT A DELIMITER CHARACTER JUST PRIOR TO NUMBER FIELD
   47 POINT=I
      KK=2
      CALL EVALUE(POINT,VALUE,KK,NUM,L)
      IF( N .EQ. -2 ) GO TO 10
      IF(SVAL.EQ.1) GO TO 48
      IF(SVAL.EQ.2) GO TO 49
      N=4
```

```
      GO TO 10
C***  SERVICE TIME IS USED
   48 CONTINUE
      JKK=NUMS+NUM-1
      DO 4480 I=NUMS,JKK
      ITYPE(I)=FSERV
      JJF=I-NUMS+1
      TTS(I)=VALUE(JJF)
      TMU(I)=1.0/VALUE(JJF)
 4480 CONTINUE
      TS=VALUE(1)
      MU=1./VALUE(1)
      NUMS=JKK
      GO TO 50
C***  SERVICE RATE IS USED
   49 CONTINUE
      JKK=NUMS + NUM - 1
      DO 4490 I=NUMS,JKK
      ITYPE(I)=FSERV
      JJF=I-NUMS+1
      TMU(I)=VALUE(JJF)
      TTS(I)=1.0/VALUE(JJF)
 4490 CONTINUE
      TS=1./VALUE(1)
      MU=VALUE(1)
      NUMS=JKK
   50 CONTINUE
C***  IF SERVICE DISTRIBUTION IS GENERAL THEN VARIANCE IS NEEDED
      IF(FSERV.EQ.2) GO TO 51
      GO TO 55
   51 CONTINUE
      JJ=POINT+1
C***  POINT RETURNED FROM EVALUE AT DELIMITER CHARACTER
C***  NEXT WORD IS VAR
      IF(L(JJ).EQ.JVC.AND.L(JJ+1).EQ.JAC) GO TO 52
      N=5
      GO TO 10
```

```
   52 CONTINUE
      JJ=JJ+2
      DO 53 I=JJ,J
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JEQU) GO TO 54
   53 CONTINUE
      N=2
      GO TO 10
   54 POINT=I
      KK=2
      CALL EVALUE(POINT,VALUE,KK,NUM,L)
      IF( N .EQ. -2 ) GO TO 10
      JKK=NUMV+NUM
      DO 5550 I=NUMV,JKK
      JJF=I-NUMV+1
      TVAR(I)=VALUE(JJF)
 5550 CONTINUE
      VAR=VALUE(1)
      NUMV=JKK
   55 CONTINUE
      GO TO 10
C     ------------------------------------------------------------------
C
C
C***  CHANNEL SIZE INSTRUCTION  ***
C
C
   56 CONTINUE
C***  SEARCH FOR DELIMITER CHARACTER
      DO 57 I=3,J
      IF(L(I).EQ.JEQU.OR.L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA) GO TO 58
   57 CONTINUE
      N=2
      GO TO 10
C***  IS PROGRAM TO FIND NUMBER OF SERVERS
   58 IF(L(I+1).EQ.JFC) GO TO 59
      IFIND=0
      IF(L(I+1).GE.J0C.AND.L(I+1).LE.J9C) GO TO 64
```

```
      N=4                                                               MAIN3700
      GO TO 10                                                          MAIN3710
C***  TO FIND MINIMUM NUMBER OF SERVERS OR CHANNEL SIZE                 MAIN3720
   59 IFIND=1                                                           MAIN3730
      ICHANS=0                                                          MAIN3740
      JJ=I+5                                                            MAIN3750
C***  SEARCH FOR DELIMITER CHARACTER                                    MAIN3760
      IF(L(JJ).EQ.JCOMMA.OR.L(JJ).EQ.JCOLON.OR.L(JJ).EQ.JEQU) GO TO 61  MAIN3770
      WRITE(6,60)                                                       MAIN3780
   60 FORMAT(' *** MISSING PUNCTUATION ***')                            MAIN3790
   61 CONTINUE                                                          MAIN3800
      IF(L(JJ+1).EQ.JTO) N=4                                            MAIN3810
      IF(N.EQ.4) GO TO 10                                               MAIN3820
C***  EVALUATE PERCENT                                                  MAIN3830
      POINT=JJ                                                          MAIN3840
      CALL EVALUE(POINT,VALUE,KK,NUM,L)                                 MAIN3850
      IF( N .EQ. -2 ) GO TO 10                                          MAIN3860
      PRCENT=VALUE(NUM)                                                 MAIN3870
      IF(L(POINT).NE.JPCENT) WRITE(6,60)                                MAIN3880
      IF(L(POINT+1).NE.JCOMMA.AND.L(POINT+1).NE.JCOLON) WRITE(6,60)     MAIN3890
      JJ=POINT                                                          MAIN3900
      DO 62 I=JJ,J                                                      MAIN3910
      IF(L(I).EQ.JTO) GO TO 63                                          MAIN3920
   62 CONTINUE                                                          MAIN3930
      N=4                                                               MAIN3940
      GO TO 10                                                          MAIN3950
   63 CONTINUE                                                          MAIN3960
      POINT=I+1                                                         MAIN3970
C***  EVALUATE TIME                                                     MAIN3980
      CALL EVALUE(POINT,VALUE,KK,NUM,L)                                 MAIN3990
      IF( N .EQ. -2 ) GO TO 10                                          MAIN4000
      T=VALUE(NUM)                                                      MAIN4010
      GO TO 10                                                          MAIN4020
   64 CONTINUE                                                          MAIN4030
      POINT=I                                                           MAIN4040
C***  EVALUATE NUMBER OF CHANNELS                                       MAIN4050
      CALL EVALUE(POINT,VALUE,KK,NUM,L)                                 MAIN4060
```

```
         IF( N .EQ. -2 ) GO TO 10                                          MAIN4070
         ICHANS=VALUE(NUM)                                                 MAIN4080
         GO TO 10                                                          MAIN4090
C        ------------------------------------------------------------------MAIN4100
C                                                                          MAIN4110
C                                                                          MAIN4120
C***     QUEUE CAPACITY INSTRUCTION                                        MAIN4130
C                                                                          MAIN4140
C                                                                          MAIN4150
      65 CONTINUE                                                          MAIN4160
         DO 66 I=3,J                                                       MAIN4170
         IF(L(I).EQ.JEQU.OR.L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA) GO TO 67     MAIN4180
      66 CONTINUE                                                          MAIN4190
         N=2                                                               MAIN4200
         GO TO 10                                                          MAIN4210
C***     CHECK FOR ALPHA CHARACTER                                         MAIN4220
      67 IF(L(I+1).LE.JZC.AND.L(I+1).GE.JAC) GO TO 69                      MAIN4230
         POINT=I                                                           MAIN4240
C***     EVALUATE QUEUE CAPACITY                                           MAIN4250
         CALL EVALUE(POINT,VALUE,KK,NUM,L)                                 MAIN4260
         IF( N .EQ. -2 ) GO TO 10                                          MAIN4270
         ICAP=VALUE(NUM)                                                   MAIN4280
         GO TO 10                                                          MAIN4290
      69 CONTINUE                                                          MAIN4300
C***     QUEUE CAPACITY IS UNLIMITED                                       MAIN4310
         ICAP=-1                                                           MAIN4320
         GO TO 10                                                          MAIN4330
C        ------------------------------------------------------------------MAIN4340
C                                                                          MAIN4350
C                                                                          MAIN4360
C***     QUEUE DISCIPLINE INSTRUCTION                                      MAIN4370
C                                                                          MAIN4380
C                                                                          MAIN4390
      71 CONTINUE                                                          MAIN4400
         DO 72 I=3,J                                                       MAIN4410
         IF(L(I).EQ.JEQU.OR.L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA) GO TO 73     MAIN4420
      72 CONTINUE                                                          MAIN4430
```

```
      N=2                                                               MAIN4440
      GO TO 10                                                          MAIN4450
   73 JJ=I+1                                                            MAIN4460
C***  DETERMINE TYPE OF DISCIPLINE AND SET FLAG                         MAIN4470
      IF(L(JJ).EQ.JFC) GO TO 75                                         MAIN4480
      IF(L(JJ).EQ.JLC) GO TO 76                                         MAIN4490
      IF(L(JJ).EQ.JRC) GO TO 77                                         MAIN4500
      IF(L(JJ).EQ.JNC.AND.L(JJ+2).EQ.JNC) GO TO 78                      MAIN4510
      IF(L(JJ).EQ.JPC) GO TO 79                                         MAIN4520
      IF(L(JJ).EQ.JNC.AND.L(JJ+2).EQ.JPC) GO TO 778                     MAIN4530
      N=6                                                               MAIN4540
      GO TO 10                                                          MAIN4550
   75 IDISC=1                                                           MAIN4560
      GO TO 80                                                          MAIN4570
   76 IDISC=2                                                           MAIN4580
      GO TO 80                                                          MAIN4590
   77 IDISC=3                                                           MAIN4600
      GO TO 80                                                          MAIN4610
   78 IDISC=4                                                           MAIN4620
      GO TO 80                                                          MAIN4630
   79 IDISC=5                                                           MAIN4640
      GO TO 80                                                          MAIN4650
  778 IDISC=6                                                           MAIN4660
   80 CONTINUE                                                          MAIN4670
      GO TO 10                                                          MAIN4680
C     ------------------------------------------------------------------MAIN4690
C                                                                       MAIN4700
C                                                                       MAIN4710
C***  POPULATION SOURCE INSTRUCTION                                     MAIN4720
C                                                                       MAIN4730
C                                                                       MAIN4740
   82 CONTINUE                                                          MAIN4750
      DO 83 I=3,J                                                       MAIN4760
      IF(L(I).EQ.JEQU.OR.L(I).EQ.JCOLON.OR.L(I).EQ.JCOMMA) GO TO 84     MAIN4770
   83 CONTINUE                                                          MAIN4780
      N=2                                                               MAIN4790
      GO TO 10                                                          MAIN4800
```

```
C***  CHECK FOR ALPHA CHARACTER                                          MAIN4810
   84 IF(L(I+1).LE.JZC.AND.L(I+1).GE.JAC) GO TO 86                       MAIN4820
      POINT=I                                                            MAIN4830
C***  EVALUATE SIZE OF SOURCE POPULATION                                 MAIN4840
      CALL EVALUE(POINT,VALUE,KK,NUM,L)                                  MAIN4850
      IF( N .EQ. -2) GO TO 10                                            MAIN4860
      IPOP=VALUE(NUM)                                                    MAIN4870
      GO TO 10                                                           MAIN4880
   86 CONTINUE                                                           MAIN4890
C***  SOURCE POPULATION IS UNLIMITED                                     MAIN4900
      IPOP=-1                                                            MAIN4910
      GO TO 10                                                           MAIN4920
C     ----------------------------------------------------------------MAIN4930
C                                                                        MAIN4940
C                                                                        MAIN4950
C***  SOLVE INSTRUCTION                                                  MAIN4960
C                                                                        MAIN4970
C                                                                        MAIN4980
   87 CONTINUE                                                           MAIN4990
      CALL SOLVE                                                         MAIN5000
      WRITE(6,101)                                                       MAIN5010
  101 FORMAT(1H1)                                                        MAIN5020
C***  TO SUPRESS TABULATING DATA FOR PRIORITY MODELS                     MAIN5030
      IF(IDISC.GE.4) GO TO 89                                            MAIN5040
      IPROB=IPROB+1                                                      MAIN5050
      GO TO 2                                                            MAIN5060
   89 IPNUM=IPNUM+1                                                      MAIN5070
      GO TO 3                                                            MAIN5080
C                                                                        MAIN5090
C                                                                        MAIN5100
C***  NEXT PROBLEM INSTRUCTION                                           MAIN5110
C                                                                        MAIN5120
C                                                                        MAIN5130
   88 CONTINUE                                                           MAIN5140
      GO TO 1                                                            MAIN5150
C     ----------------------------------------------------------------MAIN5160
C                                                                        MAIN5170
```

```
      C
      C***  END OF DATA
      C
      C
   99 CONTINUE
      IPROB=IPROB-1
      IPNUM=IPNUM-1
      IF(IPROB.LE.1) GO TO 999
      CALL OUTTAB(IPROB)
  999 IF(IPNUM.LE.1) STOP 98
      CALL POUT(IPNUM)
      STOP 99
      END
```

```
      SUBROUTINE EVALUE(POINT,VALUE,KK,NUM,L)                           EVAL  10
C***  SUBROUTINE EVALUATES NUMBER IN ALPHA FORMAT TO DECIMAL NUMBER      EVAL  20
C***     AND STORES IT IN VALUE ( NUM )                                 EVAL  30
C***  L IS THE INPUT INSTRUCTION STRING WITHOUT BLANKS                  EVAL  40
C***  POINT WHEN ENTERING IS AT ONE CHARACTER BEFORE NUMBER FIELD       EVAL  50
C***  POINT WHEN LEAVING IS AT ONE POINT AFTER END OF NUMBER FIELD      EVAL  60
C***  KK IS USED TO DETERMINE IF MORE THAN ONE NUMBER FIELD MAY BE FOUNDEVAL  70
C                                                                       EVAL  80
      COMMON/BLK4/ JPLUS,JMINUS,JPER,JDIV,JMUL,JEQU,JBLANK,JCOMMA,JCOLONEVAL  90
      COMMON/BLK5/JPCENT                                                EVAL 100
      COMMON/BLK25/N                                                    EVAL 110
      DIMENSION L(80),VALUE(10)                                         EVAL 120
C***  NUM IS THE NUMBER OF PRIORITY CLASSES                             EVAL 130
      INTEGER POINT                                                     EVAL 140
      NUM=0                                                             EVAL 150
C***  DETERMINED FIELD WIDTH                                            EVAL 160
C***  NUMBER FIELD CONTAINS DIGITS ONLY                                 EVAL 170
C***  A DECIMAL POINT SEPERATES LEFT FIELD FROM RIGHT FIELD             EVAL 180
   40 JK=1                                                              EVAL 190
      SUM=0.0                                                           EVAL 200
      IF(L(POINT+1) .EQ. JPLUS ) GO TO 45                               EVAL 210
      IF(L(POINT+1) .NE. JMINUS ) GO TO 47                              EVAL 220
      WRITE(6,46)                                                       EVAL 230
   46 FORMAT(' *** NEGATIVE NUMBER IS ASSUMED TO BE POSITIVE ***')      EVAL 240
   45 JK=2                                                              EVAL 250
C***  NUMST IS POINTER AT BEGINNING OF NUMBER FIELD                     EVAL 260
   47 NUMST=POINT+JK                                                    EVAL 270
C***  MAXIMUM NUMBER OF DECIMAL DIGITS IS 10                            EVAL 280
      JFIN=POINT+11                                                     EVAL 290
C***  SCAN FIELD UNTIL DELIMITER CHARACTER IS FOUND                     EVAL 300
      DO 50 I=NUMST,JFIN                                                EVAL 310
      IF(L(I).EQ.JPCENT) GO TO 55                                       EVAL 320
      IF(L(I).EQ.JPER.OR.L(I).EQ.JCOMMA.OR.L(I).EQ.JBLANK) GO TO 55     EVAL 330
      IF(L(I).EQ.JCOLON.OR.L(I).EQ.JEQU) GO TO 55                       EVAL 340
   50 CONTINUE                                                          EVAL 350
C***  IF THE FIELD IS GREATER THAN 10 DIGITS                            EVAL 360
```

```
          WRITE(6,52)(L(I),I=NUMST,JFIN)                                    EVAL 370
       52 FORMAT(' *** DATA FIELD IS TOO LONG ***',5X,10A1)                 EVAL 380
          N=8                                                               EVAL 390
          CALL BAD(N)                                                       EVAL 400
          RETURN                                                            EVAL 410
C***      NUMEN IS POINTER AT ONE SPACE PAST LAST CHARACTER OF FIELD        EVAL 420
       55 NUMEN=I                                                           EVAL 430
          POINT=NUMEN                                                       EVAL 440
C***      NPOS IS THE NUMBER OF DIGITS IN FIELD                             EVAL 450
          NPOS=NUMEN-NUMST                                                  EVAL 460
C***      IF NPOS IS ZERO THEN THAT SUBFIELD IS EMPTY                       EVAL 470
          IF(NPOS.EQ.0) GO TO 105                                           EVAL 480
          DO 100 I=1,NPOS                                                   EVAL 490
          J=NUMEN-I                                                         EVAL 500
          M=L(J)                                                            EVAL 510
C***      CALL DECODE TO CONVERT DIGIT FROM ALPHA (M) TO NUMERICAL (D)      EVAL 520
          CALL DECODE(M,D)                                                  EVAL 530
C***      IF D IS GREATER THAN 9 THEN M WAS NOT A DIGIT                     EVAL 540
          IF(D.LT.9.5) GO TO 70                                             EVAL 550
          WRITE(6,57) (L(J),J=NUMST,NUMEN)                                  EVAL 560
       57 FORMAT(' *** BAD DATA IN THE STRING ***',5X,20A1)                 EVAL 570
          N=8                                                               EVAL 580
          CALL BAD(N)                                                       EVAL 590
          RETURN                                                            EVAL 600
C***      CONSTRUCT LEFT SIDE OF FIELD                                      EVAL 610
       70 SUM=SUM+D*10.0**(I-1)                                             EVAL 620
      100 CONTINUE                                                          EVAL 630
C***      IF POINTER NUMEN IS AT A COMMA,BLANK,COLON,=,OR % THEN NO RIGHT   EVAL 640
C***         FIELD IS FOUND                                                 EVAL 650
      105 IF(L(NUMEN) .EQ. JCOMMA .OR. L(NUMEN) .EQ. JBLANK) GO TO 200      EVAL 660
          IF(L(NUMEN).EQ.JCOLON.OR.L(NUMEN).EQ.JEQU) GO TO 200              EVAL 670
          IF(L(NUMEN).EQ.JPCENT) GO TO 200                                  EVAL 680
C***      IF POINTER PLUS ONE IS AT DELIMITER CHARACTER THEN RIGHT FIELD    EVAL 690
C***         IS EMPTY.                                                      EVAL 700
          IF(L(NUMEN+1) .EQ. JCOMMA .OR. L(NUMEN+1) .EQ. JBLANK) GO TO 200  EVAL 710
          IF(L(NUMEN+1).EQ.JCOLON.OR.L(NUMEN+1).EQ.JEQU) GO TO 200          EVAL 720
          IF(L(NUMEN+1).EQ.JPCENT) GO TO 200                                EVAL 730
```

```
      DO 150 I=1,6                                                      EVAL 740
      J=NUMEN+I                                                         EVAL 750
      POINT=J                                                           EVAL 760
C***  A BLANK, COMMA, OR % SIGN INDICATES END OF FIELD.                 EVAL 770
      IF(L(J) .EQ. JBLANK ) GO TO 189                                   EVAL 780
      IF(L(J) .EQ. JCOMMA ) GO TO 190                                   EVAL 790
      IF(L(J).EQ.JPCENT) GO TO 190                                      EVAL 800
      M=L(J)                                                            EVAL 810
      CALL DECODE(M,D)                                                  EVAL 820
      IF(D.LT.9.8) GO TO 140                                            EVAL 830
      JK=NUMEN+6                                                        EVAL 840
      WRITE(6,57)(L(J),J=NUMST,JK)                                      EVAL 850
      N=3                                                               EVAL 860
      CALL PAD(N)                                                       EVAL 870
      RETURN                                                            EVAL 880
  140 CONTINUE                                                          EVAL 890
C***  CONSTRUCT RIGHT SIDE OF FIELD                                      EVAL 900
      SUM=SUM+D/(10.0**I)                                               EVAL 910
  150 CONTINUE                                                          EVAL 920
      POINT=NUMEN+7                                                     EVAL 930
      GO TO 200                                                         EVAL 940
C***  KK=3 INDICATES END OF DATA                                         EVAL 950
  189 KK=3                                                              EVAL 960
  200 CONTINUE                                                          EVAL 970
  190 CONTINUE                                                          EVAL 980
      IF(L(NUMEN+1).EQ.JBLANK) KK=3                                     EVAL 990
      NUM=NUM+1                                                         EVAL1000
      VALUE(NUM)=SUM                                                    EVAL1010
C***  KK=1  INDICATES ONLY ONE FIELD WAS PERMITTED                       EVAL1020
C***  IF KK=2 SEARCH FOR NEXT NUMBER FIELD                               EVAL1030
      IF(KK.NE.2) GO TO 201                                             EVAL1040
      GO TO 40                                                          EVAL1050
  201 CONTINUE                                                          EVAL1060
      RETURN                                                            EVAL1070
      END                                                               EVAL1080
```

```
      SUBROUTINE DECODE(M,D)                                            DECO  10
C***  SUBROUTINE DETERMINES DIGIT AND RETURNS A DECIMAL VALUE           DECO  20
C***     ASSIGNED TO D                                                  DECO  30
      COMMON/BLK3/ J0C,J1C,J2C,J3C,J4C,J5C,J6C,J7C,J8C,J9C              DECO  40
      IF(M .EQ. J0C ) GO TO 10                                          DECO  50
      IF(M .EQ. J1C ) GO TO 11                                          DECO  60
      IF(M .EQ. J2C ) GO TO 12                                          DECO  70
      IF(M .EQ. J3C ) GO TO 13                                          DECO  80
      IF(M .EQ. J4C ) GO TO 14                                          DECO  90
      IF(M .EQ. J5C ) GO TO 15                                          DECO 100
      IF(M .EQ. J6C ) GO TO 16                                          DECO 110
      IF(M .EQ. J7C ) GO TO 17                                          DECO 120
      IF(M .EQ. J8C ) GO TO 18                                          DECO 130
      IF(M .EQ. J9C ) GO TO 19                                          DECO 140
      D=10.0                                                            DECO 150
      GO TO 20                                                          DECO 160
   10 D=0.0                                                             DECO 170
      GO TO 20                                                          DECO 180
   11 D=1.0                                                             DECO 190
      GO TO 20                                                          DECO 200
   12 D=2.0                                                             DECO 210
      GO TO 20                                                          DECO 220
   13 D=3.0                                                             DECO 230
      GO TO 20                                                          DECO 240
   14 D=4.0                                                             DECO 250
      GO TO 20                                                          DECO 260
   15 D=5.0                                                             DECO 270
      GO TO 20                                                          DECO 280
   16 D=6.0                                                             DECO 290
      GO TO 20                                                          DECO 300
   17 D=7.0                                                             DECO 310
      GO TO 20                                                          DECO 320
   18 D=8.0                                                             DECO 330
      GO TO 20                                                          DECO 340
   19 D=9.0                                                             DECO 350
      GO TO 20                                                          DECO 360
```

```
      20 RETURN
         END
```

```
      SUBROUTINE BLANK(IN,N)                                            BLAN  10
C***  SUBROUTINE BLANK TAKES BLANKS OUT OF INPUT STRING                  BLAN  20
C***  IN  IS THE INPUT INSTRUCTION                                      BLAN  30
C***  N  IS THE RETURNED LENGTH OF STRING                               BLAN  40
      DIMENSION IN(80)                                                  BLAN  50
      DATA IREF/' '/                                                    BLAN  60
      J=1                                                               BLAN  70
      DO 10 I=1,80                                                      BLAN  80
      IF(IN(I).EQ.IREF)    GO TO 10                                     BLAN  90
      IN(J)=IN(I)                                                       BLAN 100
      J=J+1                                                             BLAN 110
   10 CONTINUE                                                          BLAN 120
      DO 20 I=J,80                                                      BLAN 130
      IN(I)=IREF                                                        BLAN 140
   20 CONTINUE                                                          BLAN 150
      N=J-1                                                             BLAN 160
      RETURN                                                            BLAN 170
      END                                                               BLAN 180
```

```
      SUBROUTINE BAD(N)
C***  SUBROUTINE BAD PRINTS ERROR MESSAGE AND SETS N= -2
      IF(N.EQ.-1) GO TO 8
      WRITE(6,11) N
      IF(N.LT.1.OR.N.GT.7) GO TO 9
      GO TO (1,2,3,4,5,6,7),N
    1 WRITE(6,12)
      GO TO 9
    2 WRITE(6,13)
      GO TO 9
    3 WRITE(6,14)
      GO TO 9
    4 WRITE(6,15)
      GO TO 9
    5 WRITE(6,16)
      GO TO 9
    6 WRITE(6,17)
      GO TO 9
    7 WRITE(6,20)
    9 CONTINUE
      N=-2
      WRITE(6,18)
      RETURN
    8 CONTINUE
      WRITE(6,19)
   11 FORMAT(' *** ERROR NUMBER ',I3)
   12 FORMAT(' *** UNRECOGNIZED INSTRUCTION STATEMENT ***')
   13 FORMAT(' *** MISSING OR INCORRECT PUNCTUATION ***')
   14 FORMAT(' *** UNAVAILABLE DISTRIBUTION TYPE ***')
   15 FORMAT(' *** INCORRECT STATEMENT FORMAT ***')
   16 FORMAT(' *** VARIANCE NOT GIVEN FOR GENERAL DISTRIBUTION ***')
   17 FORMAT(' *** UNAVAILABLE SERVICE DISCIPLINE ***')
   18 FORMAT(' *** PROBLEM TERMINATED EXECUTION CONTINUING WITH NEXT PRO
     1BLEM, IF ANY ***')
   19 FORMAT(' *** MODEL NOT AVAILABLE ***')
   20 FORMAT(' *** SYSTEM DOES NOT REACH STEADY-STATE'/'     DUE TO SERV
```

```
     15R LT9LIZATICN BEING LARGER THAN UNITY ***')                BAD  370
      RETURN                                                      BAD  380
      END                                                         BAD  390
```

```
      SUBROUTINE SOLVE                                                  SOLV  10
C***  SUBROUTINE SOLVE CHECKS VARIOUS FLAGS TO DETERMINE TYPE OF         SOLV  20
C***     MODEL BEING USED                                               SOLV  30
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND                       SOLV  40
      COMMON/BLK7/ ARIVAL,KPRIOR,IDISC,FSERV,VAR,ERLANG                 SOLV  50
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU                 SOLV  60
      COMMON/BLK25/N                                                    SOLV  70
      INTEGER FSERV,ARIVAL,PRCENT,ERLANG                                SOLV  80
      REAL LAMBDA,MU                                                    SOLV  90
C***  ERROR FLAG (N) IS SET TO -1 TO INDICATE MODEL IS NOT AVAILABLE IN SOLV 100
      N=-1                                                              SOLV 110
C***  ARRIVAL IS POISSON                                                SOLV 120
      IF(ARIVAL.EQ.1) GO TO 2                                           SOLV 130
    1 CALL BAD(N)                                                       SOLV 140
      RETURN                                                            SOLV 150
    2 CONTINUE                                                          SOLV 160
C***  CHECK SERVICE DISCIPLINE                                          SOLV 170
      IF(IDISC.LT.1.OR.IDISC.GT.6) GO TO 1                              SOLV 180
      GO TO (3,1,1,16,16,16),IDISC                                      SOLV 190
C***  DISCIPLINE IS FIFO                                                SOLV 200
    3 CONTINUE                                                          SOLV 210
C***  CHECK SERVICE DISTRIBUTION                                        SOLV 220
      IF(FSERV.LT.1.OR.FSERV.GT.4) GO TO 1                              SOLV 230
      GO TO (4,12,12,15),FSERV                                          SOLV 240
C***  EXPONENTIAL SERVICE                                               SOLV 250
    4 CONTINUE                                                          SOLV 260
      IF(ICHANS.NE.1) GO TO 8                                           SOLV 270
C***  ONE SERVER                                                        SOLV 280
      IF(IPOP.LT.0.AND.ICAP.LT.0) GO TO 5                               SOLV 290
      IF(IPOP.LT.0.AND.ICAP.GT.0) GO TO 6                               SOLV 300
      IF(IPOP.GT.0.AND.ICAP.LT.0) GO TO 7                               SOLV 310
      GO TO 1                                                           SOLV 320
    5 CALL MODL1                                                        SOLV 330
      RETURN                                                            SOLV 340
    6 CALL MODL2                                                        SOLV 350
      RETURN                                                            SOLV 360
```

```
      7 CALL MODL3
        RETURN
      8 CONTINUE
C***    MORE THAN ONE SERVER
        IF(IPOP.LT.0.AND.ICAP.LT.0) GO TO 9
        IF(IPOP.LT.0.AND.ICAP.GT.0) GO TO 10
        IF(IPOP.GT.0.AND.ICAP.LT.0) GO TO 11
        GO TO 1
      9 CALL MODL4
        RETURN
     10 CALL MODL5
        RETURN
     11 CALL MODL6
        RETURN
C***    GENERAL SERVICE DISTRIBUTION
     12 CONTINUE
        IF(ICHANS.NE.1) GO TO 1
        IF(IPOP.GT.0.OR.ICAP.GT.0) GO TO 1
        IF(FSERV.EQ.3) GO TO 14
        CALL MODL7
        RETURN
C***    CONSTANT SERVICE TIME
     14 CONTINUE
        VAR=0.0
        CALL MODL7
        RETURN
C***    ERLANG SERVICE    FIFO
     15 CONTINUE
        CALL MODL8
        RETURN
C***    PRIORITIES ARE USED
     16 CONTINUE
        IF(ICHANS.NE.1) GO TO 1
        IF(IDISC.EQ.4.OR.IDISC.EQ.5) GO TO 17
        IF(IDISC.EQ.6) GO TO 17
        GO TO 1
     17 CALL MOMENT
```

```
      RETURN                                        SCLV 740
      END                                           SOLV 750
```

```
      INTEGER FUNCTION FACT(N)
C***  FACT CALCULATES FACTORIAL OF N
      IF(N.EQ.0.OR.N.EQ.1) GO TO 2
      ISUM=1
      DO 1 I=1,N
      ISUM=ISUM*I
    1 CONTINUE
      FACT=ISUM
      RETURN
    2 CONTINUE
      FACT=1
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(K)                                              OUTP  10
C***  SUBROUTINE OUTPUT PRINTS A DESCRIPTION OF THE PROBLEM AND THE      OUTP  20
C***     RESULTS CALCULATED FROM THE VARIOUS MODELS.                    OUTP  30
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND                       OUTP  40
      COMMON/BLK7/ ARIVAL,KPRIOR,IDISC,FSERV,VAR,ERLANG                 OUTP  50
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU                 OUTP  60
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ          OUTP  70
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N          OUTP  80
      COMMON/BLK11/AVEC ,ENTRY,PSI                                      OUTP  90
      COMMON/BLK13/DOWN                                                 OUTP 100
      COMMON/BLK14/PNOWT                                                OUTP 110
      COMMON/BLK16/TERM                                                 OUTP 120
      COMMON / BLK18 / ITYPE(10),ILANG(10),TTS(10),TMU(10),TVAR(10)     OUTP 130
      COMMON / BLK19 / B1(10),B2(10),TB1,TB2,U(10),RO(10),TLAMB         OUTP 140
      COMMON/BLK20/NUMS                                                 OUTP 150
      COMMON/BLK21/TPSI                                                 OUTP 160
      COMMON / BLK22 / TQ,TTQ(10),TQAVG,TSYS(10),TLQ(10),TAVG,TL(10)    OUTP 170
      COMMON / BLK24 / TB,PD,DELAY                                      OUTP 180
      COMMON / BLK26 /IPNUM                                             OUTP 190
C***  SEE MODL1 FOR DESCRIPTION OF BLK9 AND BLK10                       OUTP 200
C***  SEE MOMENT FOR DESCRIPTION OF BLK19                               OUTP 210
C***  SEE MODL11 FOR DESCRIPTION OF BLK22                               OUTP 220
C***  VARIABLE     TYPE     SIZE     COMMON BLOCK      COMMENT          OUTP 230
C***  AVEC         R        1        11                MEAN EMPTY CHANNELOUTP 240
C***  ENTRY        R        1        11                ENTRY RATE       OUTP 250
C***  PSI          R        1        11                TRAFFIC INTENSITY OUTP 260
C***  DOWN         R        1        13                PROB. OF BEING SYSOUTP 270
C***  PNOWT        R        1        14                PROB. NO WAIT    OUTP 280
C***  TERM         R        1        16                CALCULATED %TILE OUTP 290
C***                                                   VALUE OF FIND INSTOUTP 300
C***  TB           R        1        24                TIME BETWEEN     OUTP 310
C***                                                   BREAKDOWNS OR    OUTP 320
C***                                                   INTERARRIVAL TIME OUTP 330
C***                                                   PER CUSTOMER     OUTP 340
C***  PD           R        1        24                PROB. OF DELAY   OUTP 350
C***  DELAY        R        1        24                MEAN DELAY FOR   OUTP 360
```

```
C***                                                  THOSE MACHINES     OUTP 370
C***                                                  OBLIGED TO WAIT    OUTP 380
      INTEGER ARIVAL,FSERV,ERLANG                                        OUTP 390
      REAL MU,LAMBDA,L,LQ                                                OUTP 400
   88 FORMAT('1PRIORITY PROBLEM NUMBER   P',I2)                          OUTP 410
    9 FORMAT('1PROBLEM NUMBER',I5)                                       OUTP 420
   10 FORMAT('0DESCRIPTION AND ANALYSIS OF QUEUEING MODEL TYPE',I3)      OUTP 430
   11 FORMAT('0ARRIVAL DISTRIBUTION IS POISSON')                         OUTP 440
   12 FORMAT('0SOURCE POPULATION IS CONSIDERED TO BE INFINITE')          OUTP 450
   13 FORMAT('0SOURCE POPULATION IS OF SIZE',I3)                         OUTP 460
   14 FORMAT('0SERVICE TIME DISTRIBUTION IS EXPONENTIAL')                OUTP 470
  614 FORMAT('0SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS',I3,'  IS EXOUTP 480
     1PONENTIAL')                                                        OUTP 490
   15 FORMAT('0SERVICE TIME DISTRIBUTION IS GENERAL')                    OUTP 500
  615 FORMAT('0SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS',I3,'  IS GEOUTP 510
     1NERAL')                                                            OUTP 520
   16 FORMAT('0SERVICE TIME DISTRIBUTION IS CONSTANT')                   OUTP 530
  616 FORMAT('0SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS',I3,'  IS COOUTP 540
     1NSTANT')                                                           OUTP 550
   17 FORMAT('0SERVICE TIME DISTRIBUTION IS ERLANG OF CLASS ',I2)        OUTP 560
  617 FORMAT('0SERVICE TIME DISTRIBUTION FOR PRIORITY CLASS',I3,'  IS EROUTP 570
     1LANG OF CLASS',I4)                                                 OUTP 580
   18 FORMAT('0NUMBER OF SERVICE CHANNELS IN PARALLEL = ',I2)            OUTP 590
11118 FORMAT('0NUMBER OF SERVICE CHANNELS NEEDED IS   = ',I2)            OUTP 600
11018 FORMAT('0THIS GIVES AN ACTUAL PERCENTILE VALUE OF',F6.2)           OUTP 610
   19 FORMAT('0THE NUMBER OF SERVICE CHANNELS IS TO BE FOUND'/' SUCH THAOUTP 620
     1T ',F5.2,'% OF THE CUSTOMERS WAIT LESS THAN',F6.2,' TIME UNITS')  OUTP 630
   20 FORMAT('0QUEUE CAPACITY OF SYSTEM IS CONSIDERED TO BE UNLIMITED') OUTP 640
   21 FORMAT('0QUEUE CAPACITY OF SYSTEM IS EQUAL TO ',I2)                OUTP 650
   22 FORMAT('0QUEUE DISCIPLINE IS ASSUMED TO BE FIFO')                  OUTP 660
   25 FORMAT('0QUEUE DISCIPLINE IS ASSUMED TO BE NONPREEMPTIVE PRIORITY'OUTP 670
     1)                                                                  OUTP 680
   27 FORMAT('0QUEUE DISCIPLINE IS ASSUMED TO BE NONPRIORITY')           OUTP 690
  625 FORMAT(///'0*****  INFORMATION AND RESULTS FOR PRIORITY CLASS',I3,OUTP 700
     1'  *****'/)                                                        OUTP 710
 6655 FORMAT('0CUMULATIVE UTILIZATION OF PRIORITY 1 UP TO',I3,' IS =',F9OUTP 720
     1.4)                                                                OUTP 730
```

```
      6656 FORMAT('OTOTAL SERVER UTILIZATION FOR SYSTEM  . . . . . . . =',F9.4)OUTP 740
      6657 FORMAT(1H1)                                                          OUTP 750
      6658 FORMAT(///'0*****  TOTAL RESULTS OF SYSTEM  *****')                  OUTP 760
        26 FORMAT('OQUEUE DISCIPLINE IS ASSUMED TO BE PREEMPTIVE-RESUME PRIOROUTP 770
          1ITY')                                                                OUTP 780
        61 FORMAT('OMEAN INTERARRIVAL TIME . . . . . . . . . . . . . . TA=',F9.4)OUTP 790
        62 FORMAT('OMEAN ARRIVAL RATE  . . . . . . . . . . . . . LAMBDA=',F9.4)OUTP 800
       662 FORMAT('OMEAN ENTRY RATE  . . . . . . . . . . . . .  ENTRY=',F9.4)OUTP 810
        63 FORMAT('OMEAN SERVICE TIME  . . . . . . . . . . . . . . . TS=',F9.4)OUTP 820
       663 FORMAT('OVARIANCE OF SERVICE TIME . . . . . . . . . . .  VAR=',F9.4)OUTP 830
        64 FORMAT('OMEAN SERVICE RATE  . . . . . . . . . . . . . . . MU=',F9.4)OUTP 840
        65 FORMAT('OSERVER UTILIZATION . . . . . . . . . . . . . .  RHO=',F9.4)OUTP 850
       665 FORMAT('OTRAFFIC INTENSITY  . . . . . . . . . . . . . .  PSI=',F9.4)OUTP 860
        66 FORMAT('OPROPORTION OF TIME SYSTEM IS IDLE  . . . . . . . PO=',F9.4)OUTP 870
       666 FORMAT('OPROBABILITY ARRIVAL WILL NOT HAVE TO WAIT   PNOWT=',F9.4)OUTP 880
       668 FORMAT('OPROBABILITY ARRIVAL WILL HAVE TO WAIT  . . . . . PC=',F9.4)OUTP 890
       669 FORMAT('OMEAN DELAY FOR THOSE OBLIGED TO WAIT   . .  DELAY=',F9.4)OUTP 900
       670 FORMAT('OINTERARRIVAL TIME PER CUSTOMER . . . . . . . . . TB=',F9.4)OUTP 910
        67 FORMAT(' PROB( NO. OF CUSTOMERS IN SYSTEM =',I2,')  . .  P(',I2,')OUTP 920
          1=',F9.4)                                                             OUTP 930
       667 FORMAT('OMEAN NUMBER OF EMPTY CHANNELS  . . . . . . . . AVEC=',F9.4)OUTP 940
        68 FORMAT('OMEAN NUMBER OF CUSTOMERS IN SYSTEM . . . . . .  L=',F9.4)OUTP 950
        69 FORMAT('OVARIANCE OF NUMBER IN SYSTEM . . . . . . . . . VARL=',F9.4)OUTP 960
        70 FORMAT('090% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN    ',F9.4)OUTP 970
        71 FORMAT('095% OF THE TIME, NUMBER IN SYSTEM IS LESS THAN    ',F9.4)OUTP 980
        72 FORMAT('OMEAN NUMBER OF CUSTOMERS IN QUEUE  . . . . . . . LQ=',F9.4)OUTP 990
        73 FORMAT('OVARIANCE OF NUMBER IN QUEUE  . . . . . . .  VARLQ=',F9.4)OUTP1000
        74 FORMAT('OMEAN TIME IN SYSTEM  . . . . . . . . . . . . .  W=',F9.4)OUTP1010
        75 FORMAT('OVARIANCE OF TIME IN SYSTEM . . . . . . . . . . VARW=',F9.4)OUTP1020
        76 FORMAT('090% OF THE TIME, TIME IN SYSTEM IS LESS THAN      ',F9.4)OUTP1030
        77 FORMAT('095% OF THE TIME, TIME IN SYSTEM IS LESS THAN      ',F9.4)OUTP1040
        78 FORMAT('OMEAN TIME IN QUEUE  . . . . . . . . . . . . . . WQ=',F9.4)OUTP1050
        79 FORMAT('OVARIANCE OF QUEUE TIME . . . . . . . . . . .  VARWQ=',F9.4)OUTP1060
        80 FORMAT('090% OF THE TIME, TIME IN QUEUE IS LESS THAN       ',F9.4)OUTP1070
        81 FORMAT('095% OF THE TIME, TIME IN QUEUE IS LESS THAN       ',F9.4)OUTP1080
        82 FORMAT(' PROPORTION OF TIME SYSTEM IS FULL  . . . .  P(',I2,')=',FOUTP1090
          19.4)                                                                 OUTP1100
```

```
  882 FORMAT('0PROPORTION OF TIME A CUSTOMER IS IN SYSTEM   DOWN=',F9.4)OUTP1110
      NN=50                                                             OUTP1120
      IF(K.GE.11) WRITE(6,88) IPNUM                                     OUTP1130
      IF(K.LT.11) WRITE(6,9) IPROB                                      OUTP1140
      WRITE(6,10) K                                                     OUTP1150
      WRITE(6,11)                                                       OUTP1160
      GO TO (1,2,3,4,5,6,7,8,101,101,111,111,111),K                     OUTP1170
  101 CALL BAD(N)                                                       OUTP1180
C***  MODEL 1                                                           OUTP1190
    1 WRITE(6,12)                                                       OUTP1200
      WRITE(6,14)                                                       OUTP1210
      WRITE(6,18) ICHANS                                                OUTP1220
      WRITE(6,20)                                                       OUTP1230
      WRITE(6,22)                                                       OUTP1240
 1110 CONTINUE                                                          OUTP1250
      WRITE(6,61) TA(NUM)                                               OUTP1260
      WRITE(6,62) LAMBDA(NUM)                                           OUTP1270
      WRITE(6,63) TS                                                    OUTP1280
      WRITE(6,64) MU                                                    OUTP1290
      WRITE(6,65) RHO                                                   OUTP1300
      WRITE(6,66) PO                                                    OUTP1310
      IF(K.EQ.8) GO TO 11111                                            OUTP1320
      DO 1111 I=1,N                                                     OUTP1330
      WRITE(6,67) I,I,P(I)                                              OUTP1340
 1111 CONTINUE                                                          OUTP1350
11111 CONTINUE                                                          OUTP1360
      WRITE(6,68) L                                                     OUTP1370
      WRITE(6,69) VARL                                                  OUTP1380
      WRITE(6,70) PIN90                                                 OUTP1390
      WRITE(6,71) PIN95                                                 OUTP1400
      WRITE(6,72) LQ                                                    OUTP1410
      IF(K.EQ.8) GO TO 1109                                             OUTP1420
      WRITE(6,73) VARLQ                                                 OUTP1430
 1109 CONTINUE                                                          OUTP1440
      WRITE(6,74) W                                                     OUTP1450
      WRITE(6,75) VARW                                                  OUTP1460
      WRITE(6,76) PIW90                                                 OUTP1470
```

```
      WRITE(6,77) PIW95                                                 OUTP1480
      WRITE(6,78) WQ                                                    OUTP1490
      IF(K.EQ.8) RETURN                                                 OUTP1500
      WRITE(6,79) VARWQ                                                 OUTP1510
      WRITE(6,80) PIWQ90                                                OUTP1520
      WRITE(6,81) PIWQ95                                                OUTP1530
      RETURN                                                            OUTP1540
C***  MODEL 2                                                           OUTP1550
    2 CONTINUE                                                          OUTP1560
      WRITE(6,12)                                                       OUTP1570
      WRITE(6,14)                                                       OUTP1580
      WRITE(6,18) ICHANS                                                OUTP1590
      WRITE(6,21) ICAP                                                  OUTP1600
      WRITE(6,22)                                                       OUTP1610
 1113 CONTINUE                                                          OUTP1620
      WRITE(6,61) TA(NUM)                                               OUTP1630
      WRITE(6,62) LAMBDA(NUM)                                           OUTP1640
      IF(K.EQ.5) WRITE(6,662) ENTRY                                     OUTP1650
      WRITE(6,63) TS                                                    OUTP1660
      WRITE(6,64) MU                                                    OUTP1670
      IF(K.EQ.5.OR.K.EQ.6) WRITE(6,665) PSI                             OUTP1680
      WRITE(6,65) RHO                                                   OUTP1690
      WRITE(6,66) PO                                                    OUTP1700
      IF(K.EQ.7) GO TO 1116                                             OUTP1710
      IF(K.EQ.8) GO TO 1116                                             OUTP1720
      NA=N-1                                                            OUTP1730
      IF(K.EQ.6) NA=N                                                   OUTP1740
      DO1112 I=1,NA                                                     OUTP1750
      WRITE(6,67) I,I,P(I)                                              OUTP1760
 1112 CONTINUE                                                          OUTP1770
      IF(K.EQ.6) GO TO 11016                                            OUTP1780
      WRITE(6,82) N,P(N)                                                OUTP1790
11016 CONTINUE                                                          OUTP1800
      IF(K.EQ.3.OR.K.EQ.5.OR.K.EQ.6) WRITE(6,666) PNCWT                 OUTP1810
      IF(K.EQ.3.OR.K.EQ.6) WRITE(6,668) PD                              OUTP1820
      IF(K.EQ.3.OR.K.EQ.6) WRITE(6,669) DELAY                           OUTP1830
      IF(K.EQ.3.OR.K.EQ.6) WRITE(6,670) TB                              OUTP1840
```

```
      IF(K.EQ.5.OR.K.EQ.6) WRITE(6,667) AVEC                          OUTP1850
      IF(K.EQ.3) WRITE(6,882) DOWN                                    OUTP1860
 1116 CONTINUE                                                        OUTP1870
      WRITE(6,68) L                                                   OUTP1880
      WRITE(6,72) LQ                                                  OUTP1890
      WRITE(6,74) W                                                   OUTP1900
      WRITE(6,78) WQ                                                  OUTP1910
      RETURN                                                          OUTP1920
C***  MODEL 3                                                         OUTP1930
    3 CONTINUE                                                        OUTP1940
      WRITE(6,13) IPOP                                                OUTP1950
      WRITE(6,14)                                                     OUTP1960
      WRITE(6,18) ICHANS                                              OUTP1970
      WRITE(6,20)                                                     OUTP1980
      WRITE(6,22)                                                     OUTP1990
      GO TO 1113                                                      OUTP2000
C***  MODEL 4                                                         OUTP2010
    4 CONTINUE                                                        OUTP2020
      WRITE(6,12)                                                     OUTP2030
      WRITE(6,14)                                                     OUTP2040
      IF(IFIND.EQ.1) GO TO 45                                         OUTP2050
      WRITE(6,18) ICHANS                                              OUTP2060
      GO TO 46                                                        OUTP2070
   45 WRITE(6,19) PRCENT,T                                            OUTP2080
      WRITE(6,11118) ICHANS                                           OUTP2090
      WRITE(6,11018) TERM                                             OUTP2100
   46 CONTINUE                                                        OUTP2110
      WRITE(6,20)                                                     OUTP2120
      WRITE(6,22)                                                     OUTP2130
      WRITE(6,61) TA(NUM)                                             OUTP2140
      WRITE(6,62) LAMBDA(NUM)                                         OUTP2150
      WRITE(6,63) TS                                                  OUTP2160
      WRITE(6,64) MU                                                  OUTP2170
      WRITE(6,665) PSI                                                OUTP2180
      WRITE(6,65) RHO                                                 OUTP2190
      WRITE(6,66) PO                                                  OUTP2200
      DO 1114 I=1,N                                                   OUTP2210
```

```
            WRITE(6,67) I,I,P(I)                                          OUTP2220
       1114 CONTINUE                                                      OUTP2230
            WRITE(6,68) L                                                 OUTP2240
            WRITE(6,72) LQ                                                OUTP2250
            WRITE(6,73) VARLQ                                             OUTP2260
            WRITE(6,74) W                                                 OUTP2270
            WRITE(6,75) VARW                                              OUTP2280
            WRITE(6,78) WQ                                                OUTP2290
            WRITE(6,79) VARWQ                                             OUTP2300
            RETURN                                                        OUTP2310
      C***  MODEL 5                                                       OUTP2320
          5 CONTINUE                                                      OUTP2330
            WRITE(6,12)                                                   OUTP2340
            WRITE(6,14)                                                   OUTP2350
            WRITE(6,18) ICHANS                                            OUTP2360
            WRITE(6,21) ICAP                                              OUTP2370
            WRITE(6,22)                                                   OUTP2380
            GO TO 1113                                                    OUTP2390
      C***  MODEL 6                                                       OUTP2400
          6 CONTINUE                                                      OUTP2410
            WRITE(6,13) IPOP                                              OUTP2420
            WRITE(6,14)                                                   OUTP2430
            WRITE(6,18) ICHANS                                            OUTP2440
            WRITE(6,20)                                                   OUTP2450
            WRITE(6,22)                                                   OUTP2460
            GO TO 1113                                                    OUTP2470
      C***  MODEL 7                                                       OUTP2480
          7 CONTINUE                                                      OUTP2490
            WRITE(6,12)                                                   OUTP2500
            IF(VAR.LT..0001) GO TO 1107                                   OUTP2510
            WRITE(6,15)                                                   OUTP2520
            WRITE(6,663) VAR                                              OUTP2530
            GO TO 1108                                                    OUTP2540
       1107 WRITE(6,16)                                                   OUTP2550
       1108 CONTINUE                                                      OUTP2560
            WRITE(6,18) ICHANS                                            OUTP2570
            WRITE(6,20)                                                   OUTP2580
```

```
      WRITE(6,22)                                                       OUTP2590
      GO TO 1113                                                        OUTP2600
C***  MODEL 8                                                           OUTP2610
    8 CONTINUE                                                          OUTP2620
      WRITE(6,12)                                                       OUTP2630
      WRITE(6,17) ERLANG                                                OUTP2640
      WRITE(6,18) ICHANS                                                OUTP2650
      WRITE(6,20)                                                       OUTP2660
      WRITE(6,22)                                                       OUTP2670
      GO TO 1110                                                        OUTP2680
C***  MODL11, MODL12, AND MODL13                                        OUTP2690
  111 CONTINUE                                                          OUTP2700
      WRITE(6,12)                                                       OUTP2710
      DO 114 I=1,NUMS                                                   OUTP2720
      IF(ITYPE(I).EQ.1) WRITE(6,614) I                                  OUTP2730
      IF(ITYPE(I).EQ.2) WRITE(6,615) I                                  OUTP2740
      IF(ITYPE(I).EQ.3) WRITE(6,616) I                                  OUTP2750
      IF(ITYPE(I).EQ.4) WRITE(6,617) I                                  OUTP2760
  114 CONTINUE                                                          OUTP2770
      WRITE(6,18) ICHANS                                                OUTP2780
      WRITE(6,20)                                                       OUTP2790
      IF(K.EQ.11) WRITE(6,25)                                           OUTP2800
      IF(K.EQ.12) WRITE(6,26)                                           OUTP2810
      IF(K.EQ.13) WRITE(6,27)                                           OUTP2820
      DO 117 I=1,NUMS                                                   OUTP2830
      WRITE(6,625) I                                                    OUTP2840
      WRITE(6,61) TA(I)                                                 OUTP2850
      WRITE(6,62) LAMBDA(I)                                             OUTP2860
      WRITE(6,63) TTS(I)                                                OUTP2870
      WRITE(6,64) TMU(I)                                                OUTP2880
      WRITE(6,65) RO(I)                                                 OUTP2890
      WRITE(6,6655) I,U(I)                                              OUTP2900
      WRITE(6,6656) TPSI                                                OUTP2910
      WRITE(6,68) TL(I)                                                 OUTP2920
      WRITE(6,72) TLQ(I)                                                OUTP2930
      WRITE(6,74) TSYS(I)                                               OUTP2940
      IF(K.EQ.13) WRITE(6,78) TQ                                        OUTP2950
```

```
          IF(K.LT.13) WRITE(6,78) TTQ(I)                                   OUTP2960
          WRITE(6,6657)                                                    OUTP2970
      117 CONTINUE                                                         OUTP2980
          WRITE(6,6658)                                                    OUTP2990
          TAA=1./TLAMB                                                     OUTP3000
          WRITE(6,61) TAA                                                  OUTP3010
          WRITE(6,62) TLAMB                                                OUTP3020
          WRITE(6,63) TB1                                                  OUTP3030
          TAB=1./TB1                                                       OUTP3040
          WRITE(6,64) TAB                                                  OUTP3050
          WRITE(6,6656) TPSI                                               OUTP3060
          WRITE(6,68) L                                                    OUTP3070
          WRITE(6,72) LQ                                                   OUTP3080
          WRITE(6,74) TAVG                                                 OUTP3090
          WRITE(6,78) TQAVG                                                OUTP3100
          RETURN                                                           OUTP3110
          END                                                              OUTP3120
```

```
      SUBROUTINE MOMENT                                                 MOME  10
C***  SUBROUTINE MOMENT IS CALLED FROM SUBROUTINE SOLVE WHEN PRIORITY   MOME  20
C***     MODEL OR MULTI-CLASS SERVICE DISTRIBUTIONS ARE SPECIFIED       MOME  30
C***  REFERENCES        MEYER  PAGES 159,160,190                        MOME  40
C***  REFERENCES        IBM MANUAL  PAGES 12-14                         MOME  50
C***  RO(I) IS FRACTION OF SERVER UTILIZATION DUE TO TRAFFIC IN I-TH CLAMOME  60
C***  TLAMB IS TOTAL MEAN ARRIVAL RATE                                  MOME  70
C***  B1 AND B2 ARE FIRST TWO MOMENTS ABOUT MEAN                        MOME  80
C***  VAR = B2 - B1**2                                                  MOME  90
C***  TB1 AND TB2 ARE MOMENTS FOR OVERALL SERVICE TIME                  MOME 100
C***  U(I) IS CUMULATIVE UTILIZATION DUE TO TRAFFIC OF PRIORITY 1 UP TO MOME 110
C***  M IS CLASS OF ERLANG DISTRIBUTION WHICH CORRESPONDS TO SERVICE    MOME 120
C***     DISTRIBUTION OF CLASS I.                                       MOME 130
C***  J=ITYPE(I)  IS TYPE OF SERVICE DISTRIBUTION OF CLASS I.           MOME 140
C***  1= EXPONENTIAL, 2= GENERAL, 3= CONSTANT, 4= ERLANG                MOME 150
C***  TPSI  IS TOTAL UTILIZATION OF THE SERVER                          MOME 160
      COMMON/BLK7/ ARIVAL,KPRIOR,IDISC,FSERV,VAR,ERLANG                 MOME 170
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU                 MOME 180
      COMMON / BLK18 / ITYPE(10),ILANG(10),TTS(10),TMU(10),TVAR(10)     MOME 190
      COMMON / BLK19 / B1(10),B2(10),TB1,TB2,U(10),RO(10),TLAMB         MOME 200
      COMMON/BLK20/NUMS                                                 MOME 210
      COMMON/BLK21/TPSI                                                 MOME 220
C***  VARIABLE      TYPE      SIZE      COMMON BLOCK      COMMENT        MOME 230
C***  B1            R         10        19                FIRST MOMENT   MOME 240
C***  B2            R         10        19                SECOND MOMENT  MOME 250
C***  TB1           R         1         19                TOTAL FIRST MOMENTMOME 260
C***  TB2           R         1         19                TOTAL SECOND MO.  MOME 270
C***  U             R         10        19                CUMULATIVE RO  MOME 280
C***  RO            R         10        19                SERVER UTILIZATIONMOME 290
C***  TLAMB         R         1         19                TOTAL ARRIVAL RATEMOME 300
      REAL LAMBDA                                                       MOME 310
      N=111                                                             MOME 320
      SUML=0.                                                           MOME 330
C***  LOOP TO DETERMINE B1 AND B2, TLAMB, RO(I)                         MOME 340
      DO 12 I=1,NUMS                                                    MOME 350
      TSQ=TTS(I)*TTS(I)                                                 MOME 360
```

```
      M=ILANG(I)
      J=ITYPE(I)
      IF(J.LT.1.OR.J.GT.4) CALL BAD(N)
      GO TO (1,2,3,4),J
    1 B1(I)=TTS(I)
      B2(I)=2.*TSQ
      GO TO 11
    2 B1(I)=TTS(I)
      B2(I)=TVAR(I)+TSQ
      GO TO 11
    3 B1(I)=TTS(I)
      B2(I)=TSQ
      GO TO 11
    4 B1(I)=(FACT(M)/FACT(M-1))*(TTS(I)/M)
      B2(I)=(M+1)/M*(TSQ)
   11 TLAMB=SUML+LAMBDA(I)
      RO(I)=LAMBDA(I)*B1(I)
      SUML=TLAMB
   12 CONTINUE
      TB1=0.
      TB2=0.
      SUMU=0.
C***  LOOP TO DETERMINE TB1, TB2, U(I)
      DO 13 I=1,NUMS
      TB1=LAMBDA(I)/TLAMB*B1(I)+TB1
      TB2=LAMBDA(I)/TLAMB*B2(I)+TB2
      U(I)=SUMU+RO(I)
      SUMU=U(I)
   13 CONTINUE
      IF(SUMU.LT.1.) GO TO 5
      N=7
      CALL BAD(N)
      RETURN
    5 CONTINUE
      TPSI=TLAMB*TB1
C***  NONPREEMPTIVE
      IF(IDISC.EQ.4) GO TO 14
```

```
C***   PREEMPTIVE-RESUME                                                  MOME 740
       IF(IDISC.EQ.5) GO TO 15                                            MOME 750
C***   NO PRIORITY                                                        MOME 760
       IF(IDISC.EQ.6) GO TO 16                                            MOME 770
       CALL BAD(N)                                                        MOME 780
    14 CALL MODL11                                                        MOME 790
       RETURN                                                             MOME 800
    15 CALL MODL12                                                        MOME 810
       RETURN                                                             MOME 820
    16 CONTINUE                                                           MOME 830
       CALL MODL13                                                        MOME 840
       RETURN                                                             MOME 850
       END                                                                MOME 860
```

```
      SUBROUTINE PTAB(A,B,C,D)                                          PTAB  10
C***  SUBROUTINE TABULATES PRIORITY RESULTS INTO A TABLE                PTAB  20
C***  P IS INSERTED IN FRONT OF NORMAL SYMBOL NAME                      PTAB  30
C***  BLK27 IS IN COMMON WITH POUT                                      PTAB  40
      COMMON / BLK26 /IPNUM                                             PTAB  50
      COMMON / BLK27 / PL(20),PLQ(20),PW(20),PWQ(20)                    PTAB  60
      PL(IPNUM)=A                                                       PTAB  70
      PLQ(IPNUM)=B                                                      PTAB  80
      PW(IPNUM)=C                                                       PTAB  90
      PWQ(IPNUM)=D                                                      PTAB 100
      RETURN                                                            PTAB 110
      END                                                               PTAB 120
```

```
      SUBROUTINE POUT(IPNUM)
      COMMON / BLK27 / PL(20),PLQ(20),PW(20),PWQ(20)
      WRITE(6,8)
    8 FORMAT('1*** SUMMARY RESULTS OF MULTI-STREAM MODELS ***')
      WRITE(6,9)
    9 FORMAT('0PROBLEM',7X,'MEAN',9X,'MEAN',9X,'MEAN',9X,'MEAN'/' NUMBER
     1',8X,'NUMBER',7X,'NUMBER',7X,'TIME',9X,'TIME'/15X,'IN',11X,'IN',11
     2X,'IN',11X,'IN'/15X,'SYSTEM',7X,'QUEUE',8X,'SYSTEM',7X,'QUEUE'///)
   10 FORMAT(6X,I2,3(7X,F6.3),6X,F6.3//)
      DO 1 I=1,IPNUM
      WRITE(6,10) I,PL(I),PLQ(I),PW(I),PWQ(I)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TABU(A,B,C,D,E,IPROB)                                   TABU  10
C***  SUBROUTINE TABULATES COMMON RESULTS INTO A TABLE OF ARRAYS          TABU  20
C***  BLK12 IS IN COMMON WITH SUBROUTINE OUTTAB                           TABU  30
      COMMON/BLK12/TPO(20),TL(20),TLQ(20),TW(20),TWQ(20)                  TABU  40
C***  T IS INSERTED IN FRONT OF NORMAL SYMBOL NAME                        TABU  50
      TPO(IPROB)=A                                                        TABU  60
      TL(IPROB)=B                                                         TABU  70
      TLQ(IPROB)=C                                                        TABU  80
      TW(IPROB)=D                                                         TABU  90
      TWQ(IPROB)=E                                                        TABU 100
      RETURN                                                              TABU 110
      END                                                                 TABU 120
```

```
      SUBROUTINE CUTTAB(IPROB)
C***  SUBROUTINE PRINTS RESULTS OF SUBROUTINE TABU
      COMMON/BLK12/TPO(20),TL(20),TLQ(20),TW(20),TWQ(20)
      WRITE(6,8)
    8 FORMAT('1*** SUMMARY RESULTS OF SINGLE STREAM MODELS ***')
    9 FORMAT('OPROBLEM      PROB.        MEAN          MEAN          MEAN
     1    MEAN'/' NUMBER        SYSTEM        NUMBER        NUMBER        TIME
     2        TIME'/14X,'IS',10X,'IN',10X,'IN',10X,'IN',10X,'IN'/14X,'EMP
     3TY',7X,'SYSTEM',6X,'QUEUE        SYSTEM        QUEUE'///)
   10 FORMAT(6X,I2,4(6X,F6.3),6X,F6.3//)
      WRITE(6,9)
      DO 1 I=1,IPROB
      WRITE(6,10) I,TPO(I),TL(I),TLQ(I),TW(I),TWQ(I)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MODL1                                                  MODL  10
C***  POISSON / EXPONENTIAL SERVICE / ONE SERVER                        MODL  20
C***  FIFO ASSUMED                                                      MODL  30
C***  UNLIMITED QUEUE                                                   MODL  40
C***  UNLIMITED POPULATION                                              MODL  50
C***  MODL SUBROUTINES ARE A SERIES OF SUBROUTINES TO CALCULATE RESULTS MODL  60
C***  OF VARIOUS MODELS                                                 MODL  70
C***  ROUTINES ARE CALLED FROM SUBROUTINE SOLVE                         MODL  80
C***  BLK6 AND BLK8 ARE IN COMMON WITH MAIN, SOLVE, AND OUTPUT          MODL  90
C***  BLK9 AND BLK10 ARE IN COMMON WITH SUBROUTINE OUTPUT               MODL 100
C***  VAR BEFORE SYMBOL REFERS TO VARIANCE OF VARIABLE                  MODL 110
C***  THE FOLLOWING SYMBOLS ARE SIMILAR FOR ALL MODL SUBROUTINES        MODL 120
C***  PI REFERS TO 90 AND 95 PERCENTILE OF VARIABLE                     MODL 130
C***  Q AFTER SYMBOL REFERS TO QUEUE OR WAITING LINE ONLY NOT SYSTEM    MODL 140
C***  PO IS PROPORTION OF TIME SYSTEM IS IDLE                           MODL 150
C***  P(N) IS PROBABILITY N NUMBER OF CUSTOMERS ARE IN SYSTEM           MODL 160
C***  N IS ALSO USED AS ERROR NUMBER                                    MODL 170
C***  L IS MEAN NUMBER OF CUSTOMERS IN SYSTEM                           MODL 180
C***  W IS MEAN TIME IN SYSTEM                                          MODL 190
C***  K IS MODEL NUMBER                                                 MODL 200
C***  RHO IS SERVER UTILIZATION                                         MODL 210
C***  RHO SHOULD BE LESS THAN ONE                                       MODL 220
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND                       MODL 230
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU                 MODL 240
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ          MODL 250
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N          MODL 260
      REAL LAMBDA,L,LQ,MU                                               MODL 270
      K=1                                                               MODL 280
      RHO=TS/TA(NUM)                                                    MODL 290
C***  IF RHO GE 1 THEN SYSTEM DOES NOT REACH STEADY-STATE               MODL 300
      IF(RHO.LT.1.) GO TO 1                                             MODL 310
C***  ERROR NUMBER                                                      MODL 320
      N=7                                                               MODL 330
      CALL BAD(N)                                                       MODL 340
      RETURN                                                            MODL 350
    1 PO=1.-RHO                                                         MODL 360
```

```
      PO2=PO*PO
      RHO2=RHO*RHO
      SUM=PO
C***  LOOP TO CALCULATE P(N)
      DO 3 N=1,10
      P(N)=PO*RHO**N
      SUM=SUM+P(N)
      IF(SUM.GT..95) GO TO 4
    3 CONTINUE
C***  SIZE OF P VECTOR
      N=10
    4 L=RHO/PO
      VARL=L+ (L*L)
      LQ=L-RHO
      VARLQ=(RHO2*(1.+RHO-RHO2)) / PO2
      W= TS / PO
      VARW = (TS*TS) / PO2
      PIW90 = (2.3*TS) / PO
      PIW95 = (3.0*TS) / PO
      PIN90 =(ALOG10(.10) / ALOG10(RHO))-1.0
      PIN95 =(ALOG10(.05) / ALOG10(RHO))-1.0
      WQ= (RHO*TS) / PO
      VARWQ = ( (2.0 - RHO) * RHO * TS**2) / PO2
      PIWQ90 = WQ * (1.0/RHO) * ALOG((100.*RHO)/10.)
      PIWQ95 = WQ * (1.0/RHO) * ALOG((100.*RHO)/5.0)
      IF(PIN90.LE.0.) PIN90=0.0
      IF(PIN95.LE.0.) PIN95=0.0
      IF(PIWQ90.LE.0.) PIWQ90=0.0
      IF(PIWQ95.LE.0.) PIWQ95=0.0
      CALL TABU(PO,L,LQ,W,WQ,IPROB)
      CALL OUTPLT(K)
      RETURN
      END
```

```
      SUBROUTINE MODL2
C***  POISSON / EXPONENTIAL / 1 / LIMITED QUEUE = ICAP
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N
      REAL L,LQ,MU
      REAL LAMBDA
      K=2
      J=ICAP + 1
      XJ = FLOAT(J)
      RHO = TS / TA(NUM)
C***  A AND B ARE TEMPORARY VARIABLES
      A=1.0 - RHO**J
      B= 1.0 - RHO
      PO=B/A
      DO 1 N=1,ICAP
      P(N)= (B/A) * RHO**N
    1 CONTINUE
      N=ICAP
      L=RHO*((1.0-XJ*(RHO**ICAP) + ICAP * RHO**J) /(B*A))
      LQ=L-1.+PO
      W=TA(NUM)*L
      WQ=W-TS
      CALL TABU(PO,L,LQ,W,WQ,IPROB)
      CALL OUTPUT(K)
      RETURN
      END
```

```
      SUBROUTINE MODL3
C***  POISSON / EXPONENTIAL / 1 / LIMITED POPULATION = IPOP
C***  REFERENCES         PAGE 39 IBM JOURNAL, P 304 HILLIER, & P 56 HINES
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N
      COMMON/BLK13/DOWN
      COMMON / BLK14 / PNOWT
      COMMON / BLK24 / TB,PD,DELAY
      REAL L,LQ,MU
      REAL LAMBDA
      INTEGER FACT,PRCENT
      K=3
      RHO=TS/TA(NUM)
      IF(RHO.LT.1.) GO TO 13
      N=7
      CALL BAD(N)
      RETURN
   13 CONTINUE
      IFACT=FACT(IPOP)
      SUMPO=1.0
      A=0.0
      DO 1 I=1,IPOP
      JJ=IPOP-I
      A=(IFACT/FACT(JJ))*RHO**I
      P(I)=A
      SUMPO=SUMPO+A
    1 CONTINUE
      PO=1./SUMPO
      DO 2 N=1,IPOP
      P(N)=P(N)*PO
    2 CONTINUE
      N=IPOP
      L=IPOP-(1.0/RHO)*(1.0-PO)
      LQ=IPOP-((LAMBDA(NUM)+MU)/LAMBDA(NUM))*(1.-PO)
```

```
C***  TB IS MEAN TIME BETWEEN BREAKDOWNS              MODL 370
      TB=IPOP*TS/(1.-PO)                              MODL 380
C***  RM  IS SERVER UTILIZATION                       MODL 390
C***  DOWN  IS THE PROBABILITY MACHINE K IS DOWN      MODL 400
      RM=1.-PO                                        MODL 410
      DOWN=1.-((TA(NUM)/TS)/IPOP)*RM                  MODL 420
      W=TB-TA(NUM)                                    MODL 430
      WQ=W-TS                                         MODL 440
      PNOWT=PO                                        MODL 450
      PD= 1. - PNOWT                                  MODL 460
      DELAY= WQ / PD                                  MODL 470
      CALL TABU(PO,L,LQ,W,WQ,IPROB)                   MODL 480
      CALL OUTPUT(K)                                  MODL 490
      RETURN                                          MODL 500
      END                                             MODL 510
```

```
      SUBROUTINE MODL4
C***  POISSON / EXPONENTIAL / >1  = ICHANS
      COMMON/BLK6/PRCENT,IPCP,ICHANS,T,ICAP,IFIND
      COMMON / BLK8 / TA(10),TS,NUM,IPRCB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N
      COMMON/BLK11/AVEC ,ENTRY,PSI
      COMMON/BLK16/TERM
      REAL L,LQ,MU
      REAL LAMBDA
      INTEGER FACT
C***  PSI IS TRAFFIC INTENSITY,  THIS IS A DIMENSIONLESS RATIO
C***     INDICATING THE IMPACT OF A TRAFFIC STREAM UPON A
C***     SERVICE SYSTEM.
C***  RATIO OF MEAN SERVICE TIME FOR CUSTOMERS     TO
C***  MEAN TIME BETWEEN CUSTOMERS ARRIVALS
      PSI=LAMBDA(NUM)/MU
C***  IF IFIND IS 1 THEN NUMBER OF CHANNELS IS TO BE FOUND
C***  START WITH CHANNELS EQUAL TO FIRST INTEGER GREATER THAN PSI
C***  THIS IS NECESSARY FOR RHO TO BE LESS THAN 1
      IF(IFIND.EQ.1) ICHANS=PSI+1.01
   10 CONTINUE
      RHO=PSI/FLOAT(ICHANS)
      IF(RHO.LT.1.) GO TO 13
      N=7
      CALL BAD(N)
      RETURN
   13 CONTINUE
      MM=ICHANS-1
      KK=0
      SUM=0.
      DO 1 K=KK,MM
      D=FACT(K)
      SUM=SUM+PSI**K/D
    1 CONTINUE
      K=MM
```

```
      PO=1./(SUM+(PSI**ICHANS)/( FLOAT(FACT(ICHANS))*(1.-RHO)))
      SUM=PO
      IF(MM.EQ.0) GO TO 22
      DO 2 I=1,MM
      P(I)=((PSI**I)/FACT(I))*PO
      SUM=SUM+P(I)
    2 CONTINUE
   22 CONTINUE
      Q=1.-SUM
      IF(IFIND.EQ.0) GO TO 23
      PM=Q
      POWER=-1.*(FLOAT(ICHANS)-PSI)*T/TS
C***  TERM IS %TILE VALUE CALCULATED FOR THIS NUMBER OF SERVERS
      TERM=1.-PM*EXP(POWER)
      TERM=100.*TERM
C***  WHEN TERM IS .GE. TO PRCENT THEN MINIMUM NUMBER OF CHANNELS
C***     IS FOUND
      IF(TERM.GE.PRCENT) GO TO 23
      ICHANS=ICHANS+1
      GO TO 10
   23 CONTINUE
      JJ=ICHANS+10
      A=FACT(ICHANS)
      DO 3 N=ICHANS,JJ
      KK=N-ICHANS
      P(N)=(PSI**N*PO)/(A*ICHANS**KK)
      SUM=SUM+P(N)
      IF(SUM.GT..95) GO TO 4
    3 CONTINUE
      N=JJ
    4 CONTINUE
      LQ=(PO*PSI**ICHANS *RHO)/(FACT(ICHANS)*(1.0-RHO)**2)
      L=LQ+PSI
      WQ=LQ*TA(NUM)
      W=WQ+TS
      A=(1.-RHO)*(1.-RHO)
      VARWQ=((2.-Q)*Q*TS*TS)/(ICHANS*ICHANS*A)
```

```
      VARW=VARWQ+TS*TS                                                  MODL 740
      VARLQ=(RHO*Q*(1.+RHO-RHO*Q))/A                                    MODL 750
      CALL TABU(PO,L,LQ,W,WQ,IPROB)                                     MODL 760
      K=4                                                               MODL 770
      CALL OUTPUT(K)                                                    MODL 780
      RETURN                                                            MODL 790
      END                                                               MODL 800
```

```
      SUBROUTINE MODL5
C***  POISSON / EXPONENTIAL / >1 / LIMITED QUEUE = ICAP
C***  REFERENCES           PAGES 310-311 HILLIER AND LIEBERMAN, AND PAGES 4
C***                       HINES
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N
      COMMON/BLK11/AVEC ,ENTRY,PSI
      COMMON/BLK14/PNOWT
      REAL L,LQ,MU
      REAL LAMBDA
      INTEGER FACT
      PSI=LAMBDA(NUM)/MU
      RHO=PSI/ICHANS
   10 FORMAT('0***** WARNING CAPACITY IS LESS THAN NUMBER OF CHANNELS')
      IF(ICAP.LT.ICHANS) WRITE(6,10)
C***  JK IS MINIMUM OF CAPACITY OR CHANNELS
C***  CAPACITY IS GENERALLY AT LEAST AS LARGE AS THE NUMBER OF CHANNELS
      JK=MIN0(ICAP,ICHANS)
      JK=JK-1
      SUM1=1.0
      DO 1 I=1,JK
      SUM1=SUM1+(PSI**I)/FACT(I)
    1 CONTINUE
      IE=ICAP-ICHANS+1
      SUM2=(1.0-RHO**IE)/(1.0-RHO)
      PO=1.0/(SUM1+((PSI**ICHANS)/FACT(ICHANS))*SUM2)
      DO 3 N=1,JK
      P(N)=(PSI**N/FACT(N))*PO
    3 CONTINUE
      S=FACT(ICHANS)
      N=ICHANS
      P(ICHANS)=((PSI**N)/S)*PO
      J=ICHANS+1
      DO 4 N=J,ICAP
```

```
      NM1=N-1                                                           MODL 370
      P(N)=P(NM1)*PSI/ICHANS                                            MODL 380
    4 CONTINUE                                                          MODL 390
      N=MAX0(ICAP,ICHANS)                                               MODL 400
      IF(ICAP.GT.ICHANS) GO TO 6                                        MODL 410
      LQ=0.                                                             MODL 420
      SUM=0.                                                            MODL 430
      DO 5 I=1,ICAP                                                     MODL 440
      SUM=SUM+I*P(I)                                                    MODL 450
    5 CONTINUE                                                          MODL 460
      L=SUM                                                             MODL 470
    6 CONTINUE                                                          MODL 480
      I=ICAP-ICHANS                                                     MODL 490
      LQ=(P0*PSI**ICHANS*RHO)/(S*(1.-RHO)*(1.-RHO))*(1.-RHO**I-I*RHO**I*MODL 500
     1(1.-RHO))                                                         MODL 510
      SUM1=0.                                                           MODL 520
      SUM2=P0                                                           MODL 530
      J=ICHANS-1                                                        MODL 540
      DO 7 I=1,J                                                        MODL 550
      SUM1=SUM1+I*P(I)                                                  MODL 560
      SUM2=SUM2+P(I)                                                    MODL 570
    7 CONTINUE                                                          MODL 580
C***  PROB OF NO WAITING TIME                                           MODL 590
      PNOWT=SUM2                                                        MODL 600
      L=SUM1+LQ+ICHANS*(1.-SUM2)                                        MODL 610
      SUM=ICHANS*P0                                                     MODL 620
      DO 8 I=1,ICHANS                                                   MODL 630
      SUM=SUM+(ICHANS-I)*P(I)                                           MODL 640
    8 CONTINUE                                                          MODL 650
      IL=ICAP-ICHANS                                                    MODL 660
C***  ENTRY RATE TO SYSTEM, THIS IS LESS THAN ARRIVAL RATE DUE TO LOSSESMODL 670
      ENTRY=LAMBDA(NUM)*(1.-P(ICAP))                                    MODL 680
      W=L/ENTRY                                                         MODL 690
      WQ=LQ/ENTRY                                                       MODL 700
      SUM=ICHANS*P0                                                     MODL 710
      DO 9 I=1,ICHANS                                                   MODL 720
      SUM=SUM+(ICHANS-I)*P(I)                                           MODL 730
```

```
    9 CONTINUE                                              MODL 740
C***  AVERAGE NUMBER OF EMPTY CHANNELS                      MODL 750
      AVEC=SUM                                              MODL 760
      CALL TABU(PO,L,LQ,W,WQ,IPROB)                         MODL 770
      K=5                                                   MODL 780
      CALL OUTPUT(K)                                        MODL 790
      RETURN                                                MODL 800
      END                                                   MODL 810
```

```
      SUBROUTINE MODL6
C***  POISSON / EXPONENTIAL / >1 / LIMITED POPULATION
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N
      COMMON/BLK11/AVEC ,ENTRY,PSI
      COMMON / BLK14 / PNOWT
      COMMON / BLK24 / TB,PD,DELAY
      REAL L,LQ,MU
      REAL LAMBDA
      INTEGER FACT,PRCENT
      PSI=LAMBDA(NUM)/MU
      RHO=PSI/ICHANS
      IF(RHO.LT.1.) GO TO 13
      N=7
      CALL BAD(N)
      RETURN
   13 CONTINUE
      N=IPOP
      M=ICHANS
      NF=FACT(N)
      MF=FACT(M)
      SUM1=1.
      SUM2=0.
      DO 1 I=1,M
      A=(NF*PSI**I)/(FACT(N-I)*FACT(I))
      SUM1=SUM1+A
      P(I)=A
    1 CONTINUE
      MM=M+1
      DO 2 I=MM,N
      NE=I-M
      B=(NF*PSI**I)/(MF*M**NE*FACT(N-I))
      P(I)=B
      SUM2=SUM2+B
```

```
    2 CONTINUE                                                          MODL 370
      PO=1./(SUM1+SUM2)                                                 MODL 380
      DO 3 I=1,N                                                        MODL 390
      P(I)=P(I)*PO                                                      MODL 400
    3 CONTINUE                                                          MODL 410
      LQ=0.                                                             MODL 420
      DO 4 I=MM,N                                                       MODL 430
      LQ=LQ+(I-M)*P(I)                                                  MODL 440
    4 CONTINUE                                                          MODL 450
C***  AVERAGE NUMBER OF EMPTY CHANNELS                                  MODL 460
      AVEC=M*PO                                                         MODL 470
      PNOWT=PO                                                          MODL 480
      DO 5 I=1,M                                                        MODL 490
      AVEC=AVEC+(M-I)*P(I)                                              MODL 500
      PNOWT=PNOWT+P(I)                                                  MODL 510
    5 CONTINUE                                                          MODL 520
      L=M+LQ-AVEC                                                       MODL 530
      WQ=LQ/(LAMBDA(NUM)*(N-L))                                         MODL 540
      W=L/(LAMBDA(NUM)*(N-L))                                           MODL 550
      TB=TS+TA(NUM)+WQ                                                  MODL 560
      PD= 1. - PNOWT                                                    MODL 570
      DELAY= WQ / PD                                                    MODL 580
      CALL TABU(PO,L,LQ,W,WQ,IPROB)                                     MODL 590
      K=6                                                               MODL 600
      CALL OUTPUT(K)                                                    MODL 610
      RETURN                                                            MODL 620
      END                                                               MODL 630
```

```
      SUBROUTINE MODL7
C***  POISSON / GENERAL / 1 / VARIANCE = VAR
C***  IF VAR = 0 THEN SERVICE DISTRIBUTION IS CONSTANT
C***  REFERENCES       PAGES 11-13 IBM JOURNAL AND PAGE 301 HILLIER
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND
      COMMON/BLK7/ ARIVAL,KPRIOR,IDISC,FSERV,VAR,ERLANG
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N
      REAL L,LQ,MU
      REAL LAMBDA
      A=LAMBDA(NUM)
      RHO=A/MU
      IF(RHO.LT.1.) GO TO 1
      N=7
      CALL BAD(N)
      RETURN
    1 PO=1.-RHO
      LQ=A*A*VAR+RHO*RHO/(2.*(1.-RHO))
      L=RHO+LQ
      WQ=LQ/A
      W=WQ+TS
      CALL TABU(PO,L,LQ,W,WQ,IPROB)
      K=7
      CALL OUTPUT(K)
      RETURN
      END
```

```
      SUBROUTINE MODL8
C***  POISSON / ERLANG - K / 1       CLASS FLAG IS ERLANG
C***  ERLANG CLASS IS EQUAL TO MEAN**2/VARIANCE
C***  REFFRENCES        PAGES 13-20 IBM JOURNAL, AND PAGES 302-303 HILLI
C***  SPECIAL CASE OF GENERAL WITH VARIANCE EQUAL TO 1./ERLANG*MU**2
      COMMON/BLK6/PRCENT,IPOP,ICHANS,T,ICAP,IFIND
      COMMON/BLK7/ ARIVAL,KPRICR,IDISC,FSERV,VAR,ERLANG
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,P0,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON/BLK10/W,VARW,PIW90,PIW95,WQ,VARWQ,PIWQ90,PIWQ95,N
      COMMON/BLK11/AVEC ,ENTRY,PSI
      REAL L,LQ,MU
      INTEGER ERLANG
      REAL LAMBDA
      A=LAMBDA(NUM)
      VAR=TS*TS/ERLANG
      RHO=A/MU
      IF(RHO.LT.1.) GO TO 13
      N=7
      CALL BAD(N)
      RETURN
   13 CONTINUE
      LQ=(A*A*VAR+RHO*RHO)/(2.*(1.-RHO))
      WQ=LQ/A
      W=WQ+TS
      L=A*W
      P0=1.-RHO
      B=(RHO/(1.-RHO)**2)
      C=RHO*(10.-RHO)/6.
      D=(3.-3.*RHO+RHO*RHO)/ERLANG
      E=(RHO*(8.-5.*RHO))/(6.*ERLANG*ERLANG)
      VARL=B*(1.-RHO/2.*(3.-C-D-E))
      F=(TS/(1.-RHO))**2
      G=1.-RHO*(4.-RHO)*(1.-1./ERLANG)/6.
      H=1.+1./ERLANG
      C=1.-(RHO/2.)*(1.-1./ERLANG)
```

```
      CO=C*O                                                            MODL 370
      VARW=F*(G*H-CO)                                                   MODL 380
      SIGMAW=SQRT(VARW)                                                 MODL 390
      SIGMAL=SQRT(VARL)                                                 MODL 400
      PIN90=L+1.3*SIGMAL                                                MODL 410
      PIN95=L+2.0*SIGMAL                                                MODL 420
      PIW90=W+1.3*SIGMAW                                                MODL 430
      PIW95=W+2.0*SIGMAW                                                MODL 440
      CALL TABU(PO,L,LQ,W,WQ,IPROB)                                     MODL 450
      K=8                                                               MODL 460
      CALL OUTPUT(K)                                                    MODL 470
      RETURN                                                            MODL 480
      END                                                               MODL 490
```

```
      SUBROUTINE MODL11
C***  POISSON/ MANY/ 1/ NONPREEMPTIVE PRIORITY
C***  REFERENCE         PAGES 24-26 IBM JOURNAL
C***  SUBROUTINE IS CALLED FROM SUB MOMENT, USED TO CALCULATE OUTPUT.
C***  THE FOLLOWING SYMBOLS ARE THE SAME FOR MODL11, MODL12, AND MODL13
C***  TO  MEAN WAIT TIME
C***  TTQ(I)  MEAN WAITING TIME FOR SERVICE OF AN ARRIVING I-CUSTOMER
C***  TSYS(I)  MEAN TIME FOR A I-CUSTOMER TO GET THROUGH THE SYSTEM
C***  TQAVG  AVERAGE WAITING TIME FOR ALL CUSTOMERS
C***  TAVG  AVERAGE TIME IN SYSTEM FOR ALL CUSTOMERS
C***  TLQ(I)  MEAN NUMBER OF I-CUSTOMERS WAITING IN LINE
C***  LQ  MEAN NUMBER OF TOTAL CUSTOMERS WAITING IN LINE
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,P0,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON / BLK19 / B1(10),B2(10),TB1,TB2,U(10),RO(10),TLAMB
      COMMON/BLK20/NUMS
      COMMON/BLK21/TPSI
      COMMON / BLK22 / TQ,TTQ(10),TQAVG,TSYS(10),TLQ(10),TAVG,TL(10)
      REAL LAMBDA,LQ,L,MU
      TQAVG=0.
      DO 2 I=1,NUMS
      J=I-1
      IF (J.EQ.0) GOTO 1
      TTQ(I)=(TLAMB*TB2)/(2.*(1.-U(J))*(1.-U(I)))
      GO TO 2
    1 CONTINUE
      TTQ(I)=(TLAMB*TB2)/(2.*(1.-U(I)))
    2 CONTINUE
      DO 3 I=1,NUMS
      TQAVG=LAMBDA(I)/TLAMB*TTQ(I)+TQAVG
      TSYS(I)=TTQ(I)+B1(I)
      TLQ(I)=LAMBDA(I)*TTQ(I)
      TL(I)=TLQ(I)+RO(I)
    3 CONTINUE
      TAVG=TQAVG+TB1
      LQ=TLAMB*TQAVG
```

```
      L=LQ+TPSI                                                    MODL 370
      CALL PTAB(L,LQ,TAVG,TQAVG)                                   MODL 380
      K=11                                                         MODL 390
      CALL OUTPLT(K)                                               MODL 400
      RETURN                                                       MODL 410
      END                                                          MODL 420
```

```
      SUBROUTINE MODL12
C***  POISSON/ MANY/ 1/ PREEMPTIVE-RESUME PRIORITY
C***  REFERENCE          PAGES 24-26 IBM JOURNAL
C***  SIMILAR TO MODL11
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON / BLK19 / B1(10),B2(10),TB1,TB2,U(10),RO(10),TLAMB
      COMMON/BLK20/NUMS
      COMMON/BLK21/TPSI
      COMMON / BLK22 / TQ,TTQ(10),TQAVG,TSYS(10),TLQ(10),TAVG,TL(10)
      REAL LAMBDA,L,LQ,MU
      DO 3 I=1,NUMS
      JJ=I-1
      A=0.
         DO 1 J=1,I
         A=A+LAMBDA(J)*B2(J)
    1    CONTINUE
      C=B1(I)+A/(2.*(1.-U(I)))
      IF(JJ.EQ.0) GO TO 2
      TSYS(I)=1./(1.-U(JJ))*C
      GO TO 3
    2 TSYS(I)=C
    3 CONTINUE
      TAVG=0.
      DO 4 I=1,NUMS
      TTQ(I)=TSYS(I)-B1(I)
      TAVG=TSYS(I)*LAMBDA(I)/TLAMB
      TL(I)=LAMBDA(I)*TSYS(I)
      TLQ(I)=TL(I)-RO(I)
    4 CONTINUE
      L=TAVG*TLAMB
      TQAVG=TAVG-TB1
      LQ=L-TPSI
      CALL PTAB(L,LQ,TAVG,TQAVG)
      K=12
      CALL OUTPUT(K)
```

```
      RETURN
      END
```

```
      SUBROUTINE MODL13
C***  POISSON/ MANY/ 1/ NO PRIORITY
C***  REFERENCE          IBM PAGE 25
      COMMON / BLK8 / TA(10),TS,NUM,IPROB,LAMBDA(10),MU
      COMMON / BLK9 / RHO,PO,P(50),L,VARL,PIN90,PIN95,LQ,VARLQ
      COMMON / BLK19 / B1(10),B2(10),TB1,TB2,U(10),RO(10),TLAMB
      COMMON/BLK20/NUMS
      COMMON/BLK21/TPSI
      COMMON / BLK22 / TQ,TTQ(10),TQAVG,TSYS(10),TLQ(10),TAVG,TL(10)
      REAL LAMBDA,L,LQ,MU
      RHO=TB1*TLAMB
      TQ=TLAMB*TB2/(2.*(1.-RHO))
      TAVG=TQ+TB1
      DO 1 I=1,NUMS
      TSYS(I)=TQ+B1(I)
      TL(I)=LAMBDA(I) * TSYS(I)
      TLQ(I)=LAMBDA(I) * TQ
    1 CONTINUE
      TQAVG=TQ
      TPSI=RHO
      L=TLAMB * TAVG
      LQ=TLAMB * TQ
      CALL PTAB(L,LQ,TAVG,TQAVG)
      K=13
      CALL OUTPUT(K)
      RETURN
      END
```

```
      BLOCK DATA                                                        BLOC  10
      COMMON/BLK1/ JAC,JBC,JCC,JDC,JEC,JFC,JGC,JHC,JIC,JJC,JKC,JLC,JMC  BLOC  20
      COMMON/BLK2/ JNC,JOC,JPC,JQC,JRC,JSC,JTC,JUC,JVC,JWC,JXC,JYC,JZC  BLOC  30
      COMMON/BLK3/ J0C,J1C,J2C,J3C,J4C,J5C,J6C,J7C,J8C,J9C              BLOC  40
      COMMON/BLK4/ JPLUS,JMINUS,JPER,JDIV,JMUL,JEQU,JBLANK,JCOMMA,JCOLONBLOC  50
      COMMON/BLK5/JPCENT                                                BLOC  60
      DATA JAC,JBC,JCC,JDC,JEC,JFC,JGC/ 'A','B','C','D','E','F','G' /   BLOC  70
      DATA JHC,JIC,JJC,JKC,JLC,JMC,JNC/ 'H','I','J','K','L','M','N' /   BLOC  80
      DATA JOC,JPC,JQC,JRC,JSC,JTC,JUC/ 'O','P','Q','R','S','T','U' /   BLOC  90
      DATA JVC,JWC,JXC,JYC,JZC,J0C,J1C/ 'V','W','X','Y','Z','0','1' /   BLOC 100
      DATA J2C,J3C,J4C,J5C,J6C,J7C,J8C/ '2','3','4','5','6','7','8' /   BLOC 110
      DATA J9C,JPLUS,JMINUS,JPER,JDIV /  '9','+','-','.','/' /          BLOC 120
      DATA JMUL,JEQU,JBLANK,JCOMMA,JCOLON / '*','=',' ',',',':' /       BLOC 130
      DATA JPCENT/'%'/                                                  BLOC 140
      END                                                               BLOC 150
```