

AD-A149 548

A COMPUTER SYSTEM FOR GENERATING SINGLE AND DOUBLE  
SAMPLING PLANS FOR ATT. (U) FLORIDA UNIV GAINESVILLE  
DEPT OF INDUSTRIAL AND SYSTEMS ENGIN.  
K W BEITLER ET AL. AUG 84 RR-84-35

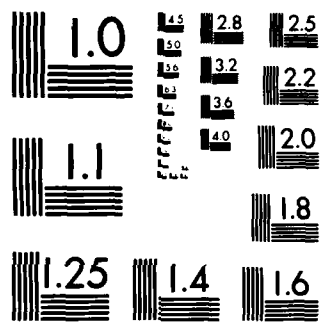
1/1

UNCLASSIFIED

F/G 9/2

NL


END



MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS 1963 A

12

A COMPUTER SYSTEM FOR GENERATING SINGLE AND DOUBLE  
SAMPLING PLANS FOR ATTRIBUTES DATA

Research Report No. 84-35

by

Kenneth W. Beitle

and

Richard S. Leavenworth

AD-A149 548

# RESEARCH REPORT

MIC FILE COPY

SDTIC  
ELECTE  
JAN 14 1985  
AT

Industrial & Systems  
Engineering Department  
University of Florida  
Gainesville, FL 32611

84 17 10 053

## TABLE OF CONTENTS

	PAGE
ABSTRACT.....	i
INTRODUCTION.....	1
DESCRIPTION OF PROGRAM SEGMENTS.....	1
List of Program Segments.....	1
Program Segment QMIL.FOR.....	2
Program Segment QASN.FOR.....	4
Program Segment QAFI.FOR.....	6
PROGRAM OPERATING INSTRUCTIONS AND OUTPUT.....	7
Initializing the Program.....	7
Running QASN.FOR.....	8
Running QAFI.FOR.....	10
Running QMIL.FOR.....	10
REFERENCES.....	13
APPENDIXES .....	14
Appendix A - Program Example Runs.....	15
Appendix B - Program Listing.....	32

ABSTRACT

*Average Sample Number (ASN),*

This study brings together the programs developed in three previous reports into a unified sampling system. One segment contains the single and double sampling plans of MIL-STD-105D along with subroutines that evaluate the normal, tightened and reduced plans including switching rules. Output includes system OC curves, and *Average Fraction Inspected (AFI)* ~~ASN~~, ~~AQO~~, and ~~AFI~~ curves in either graphical or tabular form. Another segment derives single and double sampling plans to satisfy two points designated on the OC curve. An algorithm is employed which seeks to minimize the ASN at a designated Acceptable Quality Level (AQL). The third segment is designed for use when rectifying inspection is employed. It derives single and double sampling plans based on one point on the OC curve, either the process average or a specified AQL value, intended to not exceed a specified *Average Outgoing Quality Limit* ~~AQL~~ and minimizing the AFI at the process average or specified AQL. Programming is written in FORTRAN IV and development was on a VAX 11-750 computer.

Accession for	
CRIP	
1	
<i>Attch on file</i>	
303	
or	
1	
A1	



## INTRODUCTION

This report brings together the results of three previous studies dealing with acceptance sampling using attributes data. The previous studies dealt with single and double acceptance sample plans. The first involved a computerized version of MIL-STD-105D. The second dealt with the design of double sampling plans in cases where rectifying inspection is not employed. An objective function is introduced which minimizes the average amount of inspection when the process is operating at what is considered an acceptable level. The third study assumes that rectifying inspection is planned based on some specified Average Outgoing Quality Limit and has as an objective function the minimization of inspection when the process is operating at an acceptable level. All computer programming is in ANSI FORTRAN IV and the programs were developed on a VAX 11-750 computer utilizing the VMS operating systems.

## DESCRIPTION OF THE PROGRAM SEGMENTS

### List of Program Segments

The programming system is composed of seven main segments. Each performs a specific activity with respect to the operation of the system or contains the code for executing the user's choice of sampling plans. These include:

1. Finding and evaluating MIL-STD-105D single and double sampling plans as a system of normal, tightened, and reduced inspection plans.
2. Deriving sampling plans to minimize the Average Sample Number (ASN), or average sample size;
3. Deriving sampling plans to minimize the Average Fraction Inspected (AFI) based on a specified Average Outgoing Quality Limit (AOQL).

The program segments are:

- QUALITY.COM A command file that compiles and links the program files into an executable file, QMAIN.EXE.
- QMAIN.FOR The main program control file from which are called the three sampling plan programs.
- QMIL.FOR The program containing MIL-STD-105D sampling plans and the analysis package.
- QASN.FOR The program for finding single and double sampling plans based on minimizing the ASN.
- QAFI.FOR The program for finding single and double rectifying inspection sampling plans based on minimizing the AFI.
- PROBS1.FOR A program for calculating probabilities, binomial or Poisson, for single sampling plans. It is used by QASN.FOR and QAFI.FOR.
- PROBD1.FOR A program for calculating probabilities, binomial or Poisson, for double sampling plans. It is used by QASN.FOR and QAFI.FOR.

Program Segment QMIL.FOR

The program QMIL.FOR contains all of the single and double sampling acceptance plans from MIL-STD-105D for normal, tightened, and reduced inspection. They are organized such that the user goes through identically the same procedure as if the Standard were being used. First, an Inspection Level is entered (Special Levels S-1 through S-4 or General Levels I, II, or III) followed by lot or batch size. This leads the program to the selection of the appropriate sample size code letter. The required AQL (Acceptable Quality Level) is entered next. This value must be one of the prescribed values in the Standard. Any other value will lead to extraneous and undesired results. Those values are, expressed in percent defective or defects per 100 units:

0.010, 0.015, 0.025, 0.040, 0.065, 0.10, 0.15, 0.25, 0.40, 0.65,  
1.0, 1.5, 2.5, 4.0, 6.5, 10, 25, 40, 65, 100, 150, 250, 400, 650, 1000.

Once a system of sampling plans has been obtained, the user is given the option of having the system analyzed. Output of the analysis may be either in tabular or graphical form. In tabular form, a range of values of  $p$  for which the calculations are to be made must be entered. If the user has no idea of the range of  $p$  required, it may be desirable to first obtain graphical output and then select the values of  $p$  for which accurate results are to be obtained. One caution with respect to graphical output is necessary at this point. In order to obtain clear, easily readable graphical output, the printer unit must be capable of and set on condensed printing. Graphical output uses more space than the standard 80 column pica type font allows. Hence, when restricted to 80 columns, two lines are used for each actual line of output.

Output of the analysis in tabular form appears as follows:

1. The value of  $p$  for which the calculation was made.
2. The probability of acceptance.
3. The average sample size (ASN).
4. The average outgoing quality (AOQ).
5. The average fraction inspected (AFI).

Item 3 applies only to those items in samples. Items 4 and 5 apply only if rectifying inspection is being employed; i.e., that all rejected lots are screened of defective units (defects) and are repaired/replaced in the lot in acceptable condition.

The graphical output provides the same information except that curves of the characteristics are plotted rather than numerical values printed. Because of limitations on the type of graphical output, the tabular output will be more accurate.

Analytical results of the program will not correspond exactly with those shown in MIL-STD-105D because the program treats the normal, tightened, and reduced inspection plans, along with the switching rules for changing from one



to the other, as a system of plans and analyzes the entire system rather than each plan individually. The system is analyzed as a Markov chain. Steady state probabilities of being in each of a series of states under normal, tightened and reduced inspection are calculated for each value of  $p$  in the effective range of the OC curve. These probabilities are then combined to give probabilities of being in normal, tightened and reduced inspection. From these values are calculated the probabilities of acceptance of lots or batches, the OC curve points, the ASN, AFI, and AOQ. As is the case in the Standard itself, each calculation assumes that lots are being formed from a process generating a constant value of  $p$ , i.e., a process in statistical control.

Details of the models and procedures are contained in reference (1).

#### Program Segment QASN.FOR

This program segment enables the user to design custom double sampling plans to minimize the average sample size based on two points on the operating characteristic curve. In the process, it also specifies the minimum single sampling plan that meets the OC curve requirements. The selection of plans based on ASN minimization are intended for use when rectifying inspection either is not or can not be used. Inputs to this program segment include a choice between the binomial and Poisson distributions for calculating probabilities of acceptance, the two specified points on the OC curve, a seed value for the sample size of the equivalent single sampling plan, and a value for the rejection number on the first sample of the double sampling plan.

The output consists of the minimum sampling plan, a series of double sampling plans, the double sampling plan found to have the minimum ASN evaluated at the input good quality level, and the maximum value for the ASN for the  $R1$  value originally input to the program.

The two points designated for the OC curve are related to the traditional Producer's and Consumer's risks. The program uses  $P_0$  to designate the producer's maximum acceptable quality level (an AQL as defined in MIL-STD-105D) or process average with the minimum probability of acceptance set at  $(1 - \alpha)$ . In this case,  $\alpha$  is the designated maximum producer's risk. The quality level  $P_1$  is the designated consumer's maximum acceptable quality level (RQL, or rejectable quality level) with a maximum probability of acceptance of  $\beta$ . In this case,  $\beta$  is the designed maximum Consumer's risk. The equations are of the form:

$$L(P_0) > 1 - \alpha$$

$$L(P_1) < \beta$$

where  $L(P)$  is the likelihood function (probability of acceptance formula) for the particular sampling plan being analyzed.

With these two equations alone, an infinite number of sampling plans, single and double, exist which satisfy the inequality constraints. A reasonable choice from among that group is that plan which provides for minimum inspection on the average. Thus an objective function is introduced that minimizes the ASN when the process is operating at or below the AQL. This value is designated ASN or  $ASN(P_0)$  on the computer output.

Users familiar with MIL-STD-105D will remember that double and multiple sampling plans in that standard are designed such that their respective ASN's never exceed the sample size of the single sampling plan with the same OC curve. The capability of establishing this additional constraint has been provided for in this program segment. As part of the output, the user receives the maximum value of the ASN reached for each plan analyzed. This value is designated  $ASN_{MAX}$ . (The program searches the ASN function as a function of  $p$  and outputs that value.) Thus the user may search through the

program output for those plans with a ASNMAX less than or equal to the minimum single sampling plan and select from them the plan with the minimum ASN(P0). It is worth noting at this point that, in most of the plans explored in the development of this segment, addition of this constraint did not result in changing the optimal choice.

Reference (2) presents detailed findings and a full description of the program's development.

#### Program Segment QAFI.FOR

This program segment develops single and double sampling plans which meet a specified Average Outgoing Quality Limit (AOQL). They are intended for use only when rectifying inspection (100% inspection of the balance of rejected lots) is employed. Calculations assumes that rectifying inspection is 100% effective and that the binomial distribution provides sufficient accuracy in calculating probabilities of acceptance.

Inputs to this segment include a Producer's Risk Point ( $P_0, 1-\alpha$ ), the desired AOQL (designated AOQL\*), and the lot or batch size. Outputs include the minimum single sampling plan satisfying the constraints, a series of double sampling plans all of which satisfy the AOQL constraint, and that plan which satisfies the constraints and minimizes the Average Fraction Inspected (AFI) when the process is operating at the quality level  $P_0$ . The quality level  $P_0$  may be interpreted either as an AQL value as defined in MIL-STD-105D or as a process average as defined and employed in the Dodge-Romig tables. As in the Dodge-Romig tables, the rejection numbers on the first and second samples,  $R_1$  and  $R_2$ , are set at the second acceptance number,  $C_2$ , plus one.

The objective function and constraint equations are of the form:

minimize  $AFI(P_0)$

subject to:

$$L(P_0) > 1 - \alpha$$

$$AOQL < AOQL^*$$

$$R_1 = R_2 = C_2 + 1$$

First, the program finds the minimum single sampling plan satisfying the likelihood function and AOQL constraints. This information is used internally to assist in setting bounds for the algorithm that solves for the double sampling plans. Then the algorithm shifts to seeking the double sampling plan satisfying the constraints. In its search, a number of plans are found and it is from this group that the plan that minimizes  $AFI(P_0)$  is selected. Output includes the sample sizes  $N_1$ , and  $N_2$ , acceptance numbers  $C_1$  and  $C_2$ , and the  $AFI(P_0)$ , designated simply as AFI.

Reference (3) provides a complete description and analysis of the development and operation of the algorithm and program.

## PROGRAM OPERATING INSTRUCTIONS AND OUTPUT

### Initializing the Program

In the following instructions, user inputs are in capital letters. A backward arrow (+) indicates a carriage return.

Program initialization begins by entering:

@QUALITY+

This instruction executes the command file (QUALITY.COM) which compiles and links the program files into a single executable file (QMAIN.EXE). If QMAIN.EXE has been created and exists in the file, this step may be eliminated.

Once QMAIN.EXE has been created, the user enters:

RUN QMAIN+

This command starts program execution. Very shortly the following message will appear on the terminal screen:

```
WHAT DO YOU WISH TO DO?
1-DERIVE SAMPLING PLANS TO MINIMIZE ASN
2-DERIVE SAMPLING PLANS TO MINIMIZE AFI
3-EVALUATE MIL-STD-105D SAMPLING SCHEME
4-EXIT THIS PROGRAM
```

Entering the number for your selection followed by a carriage return transfers program control to QASN.FOR, QAFI.FOR, QMIL.FOR, or the computer operating system. These instructions are repeated after each run of the program until the user exits with instruction (4).

#### Running QASN.FOR

Entering number 1 from the main transfers program control to QASN.FOR, the program for generating double sampling plans designed to minimize the ASN at quality level P0. What follows is the menu QASN.FOR.

The first question in this menu asks you to name an output file (8 alphanumeric characters or less).

```
WHAT DO YOU WISH TO CALL YOUR OUTPUT FILE?
```

User enters a name and +.

Subsequent questions are:

```
CODES FOR SELECTING APPR. PROB. DIST.
BINOMIAL          =1
POISSON           =2
```

User then enters the appropriate code. The Poisson uses the intensity parameter  $\lambda = np$  in terms of defects per 100 units.

```
SELECT
SAMPLE PLANS ONLY = 1
ASN VALUES ONLY = 2
OR BOTH = 3
```

The most useful of these choices is to enter 3. Entering 1 produces an output of a large number of sampling plans all of which meet the criteria for the two points on the OC curve. Output includes the values of  $C_1$ ,  $C_2$ , and  $N_1$ , the range over which  $N_2$  may operate, and the probabilities of acceptance at  $P_1$  and  $P_0$ , respectively. Entering 2 results in the production of a large number of plans satisfying the OC curve constraints and lists values of  $ASN_{MAX}$  and  $ASN(P_0)$ . When 3 is entered, the optimization algorithm is called in thus the output is limited to those plans which were considered candidates for the optimum. Output includes  $C_1$ ,  $C_2$ , and  $N_1$ , the effective range for  $N_2$ , and the  $ASN_{MAX}$  and  $ASN(P_0)$  when the smallest value of  $N_2$  is used. It then selects the global optimum plan for the input value of  $R_1$ .

INPUT ALPHA	User enters the Producer's Risk, $\alpha$
INPUT BETA	User enters the Consumer's Risk, $\beta$
INPUT $P_0$	User enters $P_0$
INPUT $P_1$	User enters $P_1$

INPUT A SEED FOR THE SINGLE SAMPLING NO.  
IF NO SEED AVAILABLE ENTER ZERO AS THE SEED VALUE

If the user has some idea what the sample size will be for the single sampling plan satisfying the OC curve requirements, a conservative value somewhat lower than that number may be entered. This option is useful only when the sample size is known to be large and user wishes to save some time. Its use may lead to nonoptimal results if the seed value is larger than the optimal value.

INPUT A VALUE FOR ( $R_1 - C_2$ )  
IF  $R_1 = C_2$  THEN THE VALUE WOULD BE 0  
IF  $R_1 > C_2$  THEN THE VALUE WOULD BE A POSITIVE NO.  
IF  $R_1 < C_2$  THEN THE VALUE WOULD BE A NEGATIVE NO.

This option allows the user to vary the value of  $R_1$  in the search for optimality. Each run of the program provides the optimal plan for a given selected value of  $R_1$ .

From this point on the program takes over generating the desired output.

### Running QAFI.FOR

Entering 2 from the main menu transfers control to QAFI.FOR, the program segment for finding rectifying inspection plans satisfying AOQL and AQL constraints while minimizing the AFI at the AQL level. The menu for QAFI.FOR leads the user as follows:

ENTER VALUE OF ALPHA	User enters Producer's Risk, $\alpha$
ENTER VALUE OF PO	User enters quality level PO
ENTER AOQL VALUE	User enters AOQL (in percent)
ENTER LOT SIZE	User enters lot size

The program then proceeds to generate a series of sampling plans for which  $R1 = R2 = C2 + 1$  and selects the optimal plan from among this group. Output includes the values of C1, C2, N1, N2, and AFI(PO), respectively, and the optimal plan.

### Running QMIL.FOR

Entering 3 from the main menu transfers control to QMIL.FOR, the program segment for finding single and double sampling plans from MIL-STD-105D and, at the user's discretion, evaluating the resulting normal, tightened and reduced inspection plans as a system. Output of the evaluation may be either graphical or tabular. The menu for this segment is as follows:

WHAT DO YOU WISH TO CALL YOUR OUTPUT FILE?

User enters a name for the output file (8 alphanumeric characters or less).

ENTER INSPECTION LEVEL IN QUOTATION MARKS  
E.G.,; SPECIAL: 'S1', 'S2', 'S3', 'S4'  
GENERAL: '1', '2', '3'

User enters the appropriate code followed by +. A note of caution here. Any error on entry will cause program failure and automatic exit. Depending on the terminal being used, it may be necessary to use apostrophe marks rather than quotation marks surrounding the entry.

ENTER LOT SIZE: User enters lot size+

ENTER AQL IN PERCENT. REMEMBER, ONLY A STANDARD  
AQL IS ALLOWABLE User enters AQL +

DO YOU WANT SINGLE ('S') OR DOUBLE ('D')  
SAMPLING PLANS; (NOTE: ENTER S OR D IN QUOTES).

The same caution for entry of the inspection level applies here as well.

The program then proceeds to select the desired sampling system from the Standard. Once the plans are displayed, user is asked:

DO YOU WANT SCHEME EVALUATION..?  
IF YES ENTER.....1  
IF NO ENTER.....2

If the answer to this question is no, program execution stops and the user is presented with the main menu. If the answer is yes, execution continues with:

DO YOU WANT A TABLE OR A GRAPH FORMAT?  
FOR GRAPH.....ENTER: 1  
FOR TABLE.....ENTER: 2

Entering 1+ for graphical format results in the output of a plot of the OC curve, the ASN curve, the AOQ curve, and the AFI curve. (Naturally, the AOO and AFI curves have no meaning unless rectifying inspection is intended.) In order for these curves to be readable, the printer must be capable of and set on compressed printing. The horizontal scale of each of the graphs (probability scale in the case of the OC curve) requires more than 80 columns. Thus, if an attempt is made to print graphs in standard 80 column format, the printing will occur on two consecutive lines making the whole thing look very weird.

When 2+ is entered, i.e., a tabular format is requested, the computer returns with the following questions:



SPECIFY THE NUMBER OF FRACTION DEFECTIVE VALUES..

The user must enter the number of values of  $p$  for which the equation is to be made.

ENTER THE FRACTION DEFECTIVE VALUE(S),  
(PUT A COMMA BETWEEN VALUES.).....

The user then enters the values as fractional quantities, not percents. In this case the output is a table containing  $p$ ,  $P(A)$ , ASN, AOQ and AFI, respectively. It may be useful to obtain a graphical output on a first run to obtain the effective range of  $p$  required for a good tabular output. Naturally, the table is more accurate than is the graph.

## REFERENCES

1. Siddiqi, Azmat H. and R. S. Leavenworth, An Interactive Computerized Approach for Tabulating and Evaluating MIL-STD-105D. ISE Research Report No. 84-30, August, 1984.
2. Rangarajan, R. W., K.W. Beitler, and R. S. Leavenworth. Developing Double Sampling Plans for Attributes to Meet Sample Size Criteria. ISE Research No. Report 84-32, August, 1984.
3. Walker, Jo Ellen and R. S. Leavenworth. Designing Optimal A00L Sampling Plans - a Computerized Approach. ISE Research Report No. 84-1, May, 1984.

## APPENDIXES

Appendix A contains several example runs for each of the program segments.  
The entire program is listed in Appendix B.

APPENDIX A  
Program Example Runs

Example 1 QASN.FOR

WHAT DO YOU WISH TO DO?

- 1-DERIVE PLANS TO MINIMIZE ASN
- 2-DERIVE PLANS TO MINIMIZE AFI
- 3-EVALUATE MIL-STD-105D SAMPLING SCHEME
- 4-EXIT THIS PROGRAM

ENTER CHOICE

1  
WHAT DO YOU WISH TO CALL YOUR OUTPUT FILE?  
PR1

CODES FOR SELECTING APPR. PROB. DIST.

BINOMIAL               =1  
POISSON                =2

1  
SELECT  
SAMPLE PLANS ONLY =1  
ASN VALUES ONLY =2  
OR BOTH            =3

3  
INPUT ALPHA

.05  
INPUT BETA

.1  
INPUT P0

.015  
INPUT P1

.05  
INPUT A SEED FOR SINGLE SAMPLING NO.

IF NO SEED AVAILABLE ENTER ZERO AS THE SEED VALUE

0  
INPUT A VALUE FOR (R1-C2)

IF R1=C2 THEN THE VALUE WOULD BE 0

IF R1>C2 THEN THE VALUE WOULD BE A POSITIVE NO.

IF R1<C2 THEN THE VALUE WOULD BE A NEGATIVE NO.

1

ALPHA =0.0500      BETA =0.1000  
P0 =0.0150      P1 =0.0500

REJECTION NO. OF FIRST SAMPLE (R1) = C2+( 1)

\*SINGLE SAMPLING PLAN\*  
ACCEPTANCE NO. (C) = 6  
LOWER BOUND ON N (NS) = 209  
UPPER BOUND ON N (NL) = 220

DOUBLE SAMPLING PLANS

FOR C1= 0 C2= 6

(N1)	(N2S)	(N2L)	ASNMAX	ASN
59	166	166	212.4477	156.9425
60	164	165	211.5781	157.7704

DOUBLE SAMPLING PLANS

FOR C1= 1 C2= 6

(N1)	(N2S)	(N2L)	ASNMAX	ASN
89	139	139	201.8087	142.6555
90	136	137	200.3593	143.2139

DOUBLE SAMPLING PLANS

FOR C1= 2 C2= 6

(N1)	(N2S)	(N2L)	ASNMAX	ASN
115	115	116	191.8224	143.4053
116	112	114	190.8090	144.1031

DOUBLE SAMPLING PLANS

FOR C1= 3 C2= 6

(N1)	(N2S)	(N2L)	ASNMAX	ASN
139	94	96	186.4893	153.2870
140	91	94	185.9685	154.0707

DOUBLE SAMPLING PLANS

FOR C1= 4 C2= 6

(N1)	(N2S)	(N2L)	ASNMAX	ASN
161	82	82	188.3876	167.9648
162	76	80	187.3810	168.5728

DOUBLE SAMPLING PLANS

FOR C1= 5 C2= 6

(N1)	(N2S)	(N2L)	ASNMAX	ASN
184	75	87	196.2480	186.8779
185	57	80	194.3076	187.2266

\*SINGLE SAMPLING PLAN\*  
 ACCEPTANCE NO. (C) = 7  
 LOWER BOUND ON N (NS) = 234  
 UPPER BOUND ON N (NL) = 266

DOUBLE SAMPLING PLANS

FOR C1= 0 C2= 7

(N1)	(N2S)	(N2L)	ASNMAX	ASN
50	226	228	265.0669	169.8499
51	220	226	260.3219	169.2180
52	215	225	256.5318	169.0233
53	211	223	253.6964	169.2884

DOUBLE SAMPLING PLANS

FOR C1= 1 C2= 7

(N1)	(N2S)	(N2L)	ASNMAX	ASN
81	198	202	252.9615	148.9811
82	192	200	248.7259	148.9649
83	187	198	245.3612	149.2348

DOUBLE SAMPLING PLANS

FOR C1= 2 C2= 7

(N1)	(N2S)	(N2L)	ASNMAX	ASN
108	178	181	242.3765	147.3077
109	170	179	237.3196	147.2064
110	164	177	233.7746	147.5018

DOUBLE SAMPLING PLANS

FOR C1= 3 C2= 7

(N1)	(N2S)	(N2L)	ASNMAX	ASN
134	157	164	231.0574	156.3438
135	147	162	225.8645	156.3163
136	139	160	221.9088	156.5329

DOUBLE SAMPLING PLANS

FOR C1= 4 C2= 7

(N1)	(N2S)	(N2L)	ASNMAX	ASN
159	141	156	224.8282	171.6242
160	125	152	218.3522	171.4111
161	115	149	214.6785	171.7019



DOUBLE SAMPLING PLANS

FOR C1= 5 C2= 7

(N1)	(N2S)	(N2L)	ASNMAX	ASN
184	117	181	220.1528	190.2280
185	96	172	214.6608	190.2104
186	85	165	212.2600	190.7030

DOUBLE SAMPLING PLANS

FOR C1= 6 C2= 7

(N1)	(N2S)	(N2L)	ASNMAX	ASN
209	75	266	220.3671	210.8894
210	57	266	218.6382	211.4632

GLOBAL MINIMUM ASN(P0)= 142.66

CORRESPONDING N1 = 89

CORRESPONDING N2S = 139

CORRESPONDING C1 = 1

CORRESPONDING C2 = 6

Example 2 QAFI.FOR

WHAT DO YOU WISH TO DO?

- 1-DERIVE PLANS TO MINIMIZE ASN
- 2-DERIVE PLANS TO MINIMIZE AFI
- 3-EVALUATE MIL-STD-105D SAMPLING SCHEME
- 4-EXIT THIS PROGRAM

ENTER CHOICE

2

ENTER VALUE OF ALPHA

.05

ENTER VALUE OF P0

.015

ENTER AOQL VALUE

.02

ENTER LOT SIZE

2000

ALPHA= 0.050000  
P0= 0.015000  
AOQL= 0.020000  
N= 2000

\*\*\* SINGLE SAMPLING PLAN \*\*\*

NS=120  
C= 4  
AFI(P0)=0.093093

\*\* DOUBLE SAMPLING PLANS \*\*

C1	C2	N1	N2	AFI
0	4	29	91	0.056349
1	4	55	66	0.060894

C1	C2	N1	N2	AFI
0	5	28	120	0.051799
1	5	55	93	0.054159

C1	C2	N1	N2	AFI
0	6	28	147	0.050792
1	6	55	120	0.050957

C1	C2	N1	N2	AFI
0	7	28	174	0.051735
1	7	55	147	0.049855
2	7	82	121	0.056638

C1	C2	N1	N2	AFI
0	8	28	201	0.053922
1	8	55	174	0.050058
2	8	82	148	0.055690

SAMPLING PLAN MINIMUMS

C1= 1 C2= 7  
 N1= 55 N2=147  
 MINIMUM AFI=0.049855

Example 3 QMIL.FOR

WHAT DO YOU WISH TO DO?

- 1-DERIVE PLANS TO MINIMIZE ASN
- 2-DERIVE PLANS TO MINIMIZE AFI
- 3-EVALUATE MIL-STD-105D SAMPLING SCHEME
- 4-EXIT THIS PROGRAM

ENTER CHOICE

3

ENTER INSPECTION LEVEL IN QUOTATION MARKS

E.G. ; SPECIAL : 'S1', 'S2', 'S3', 'S4'

GENERAL : '1', '2', '3'

'2'

ENTER LOT SIZE:

2000

ENTER AQL IN PERCENT. REMEMBER, ONLY A  
STANDARD AQL IS ALLOWABLE

1.5

DO YOU WANT SINGLE ('S') OR DOUBLE ('D')

SAMPLING PLANS ; (NOTE: ENTER S OR D IN QUOTES).

'S'

THESE PLANS ARE:

\*:NORMAL:::TIGHTENED:::REDUCED::\*

AC 1= 5	AC 1= 3	AC 1= 2
RE 1= 6	RE 1= 4	RE 1= 5
N= 125	N= 125	N= 50

DO YOU WANT SCHEME EVALUATION..?

IF YES ENTER.....1

IF NO ENTER.....2

2

Example 4 QMIL.FOR

WHAT DO YOU WISH TO DO?

- 1-DERIVE PLANS TO MINIMIZE ASN
- 2-DERIVE PLANS TO MINIMIZE AFI
- 3-EVALUATE MIL-STD-105D SAMPLING SCHEME
- 4-EXIT THIS PROGRAM

ENTER CHOICE

3

ENTER INSPECTION LEVEL IN QUOTATION MARKS

E.G. ; SPECIAL : 'S1', 'S2', 'S3', 'S4'

GENERAL : '1', '2', '3'

'2'

ENTER LOT SIZE:

2000

ENTER AQL IN PERCENT. REMEMBER, ONLY A STANDARD AQL IS ALLOWABLE

1.5

DO YOU WANT SINGLE('S') OR DOUBLE('D')

SAMPLING PLANS ; (NOTE:ENTER S OR D IN QUOTES).

'D'

THESE PLANS ARE:

\*:NORMAL:::TIGHTENED:::REDUCED::\*

NORMAL			TIGHTENED			REDUCED		
AC 1=	2		AC 1=	1		AC 1=	0	
RE 1=	5		RE 1=	4		RE 1=	4	
N1=	80		N1=	80		N1=	32	
-----			-----			-----		
AC 2=	6		AC 2=	4		AC 2=	3	
RE 2=	7		RE 2=	5		RE 2=	6	
N2=	80		N2=	80		N2=	32	

DO YOU WANT SCHEME EVALUATION..?

IF YES ENTER.....1

IF NO ENTER.....2

1

DO YOU WANT A TABLE OR A GRAPH FORMAT ?

FOR GRAPH....ENTER: 1

FOR TABLE....ENTER: 2

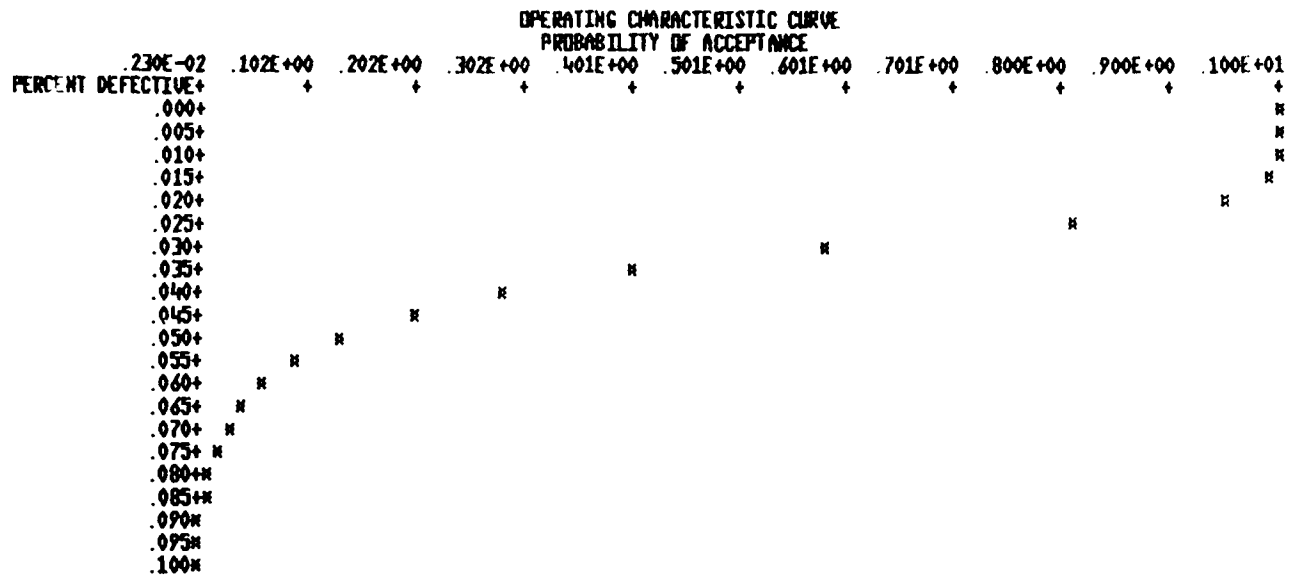
DEPT. OF ISE  
UNIVERSITY OF FLORIDA

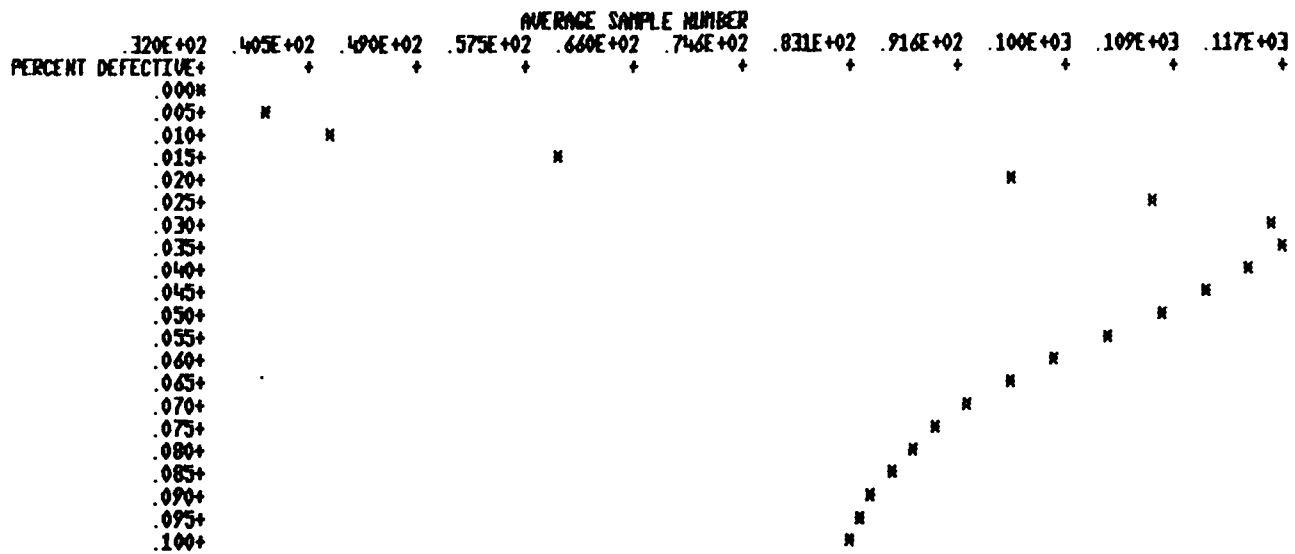
\*\*\*\*\*SAMPLING SYSTEM TO EVALUATE MIL-STD-105D\*\*\*\*\*

FOR: INSPECTION LEVEL 2  
LOT SIZE= 2000  
AQL= 1.50000  
SAMPLING PLAN D

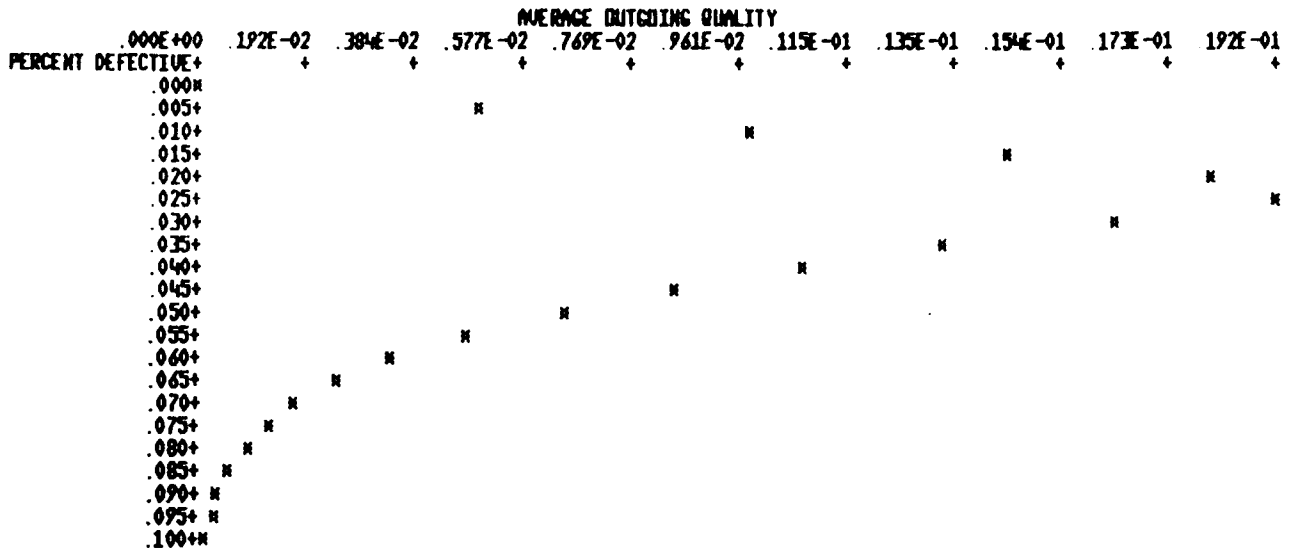
THESE PLANS ARE:

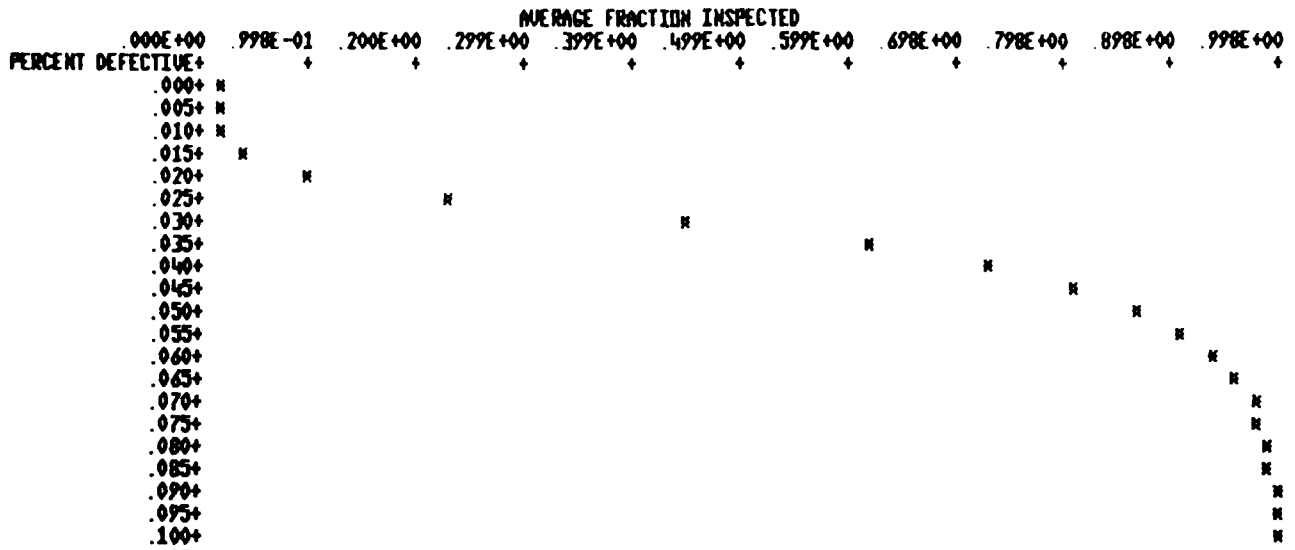
* NORMAL : : : : : TIGHTENED : : : : : REDUCED : *		
AC 1= 2	AC 1= 1	AC 1= 0
RE 1= 5	RE 1= 4	RE 1= 4
N1= 80	N1= 80	N1= 32
AC 2= 6	AC 2= 4	AC 2= 3
RE 2= 7	RE 2= 5	RE 2= 6
N2= 80	N2= 80	N2= 32











Example 5 QAFI.FOR

WHAT DO YOU WISH TO DO?

- 1-DERIVE PLANS TO MINIMIZE ASN
- 2-DERIVE PLANS TO MINIMIZE AFI
- 3-EVALUATE MIL-STD-105D SAMPLING SCHEME
- 4-EXIT THIS PROGRAM

ENTER CHOICE

3

ENTER INSPECTION LEVEL IN QUOTATION MARKS

E. G. ; SPECIAL : 'S1', 'S2', 'S3', 'S4'

GENERAL : '1', '2', '3'

'2'

ENTER LOT SIZE:

2000

ENTER AQL IN PERCENT. REMEMBER, ONLY A  
STANDARD AQL IS ALLOWABLE

1.5

DO YOU WANT SINGLE('S') OR DOUBLE('D')

SAMPLING PLANS ; (NOTE: ENTER S OR D IN QUOTES).

'D'

THESE PLANS ARE:

*:NORMAL:*****TIGHTENED:*****REDUCED:**		
AC 1= 2	AC 1= 1	AC 1= 0
RE 1= 5	RE 1= 4	RE 1= 4
N1= 80	N1= 80	N1= 32
AC 2= 6	AC 2= 4	AC 2= 3
RE 2= 7	RE 2= 5	RE 2= 6
N2= 80	N2= 80	N2= 32

DO YOU WANT SCHEME EVALUATION..?

IF YES ENTER.....1

IF NO ENTER.....2

1

DO YOU WANT A TABLE OR A GRAPH FORMAT ?

FOR GRAPH....ENTER: 1

FOR TABLE....ENTER: 2

2

SPECIFY THE NUMBER OF FRACTION DEFECTIVE VALUES..

12

ENTER THE FRACTION DEFECTIVE VALUE(S),

(PUT A COMMA BETWEEN VALUES.).....

.015, .02, .025, .03, .04, .05, .06, .065, .07, .08, .09, .1

SCHEME OPERATING CHARACTERISTICS

P	P(A)	ASN	AOQ	AFI
0.015	0.9943	59.80	0.0145	0.04
0.020	0.9467	95.47	0.0180	0.10
0.025	0.8106	107.15	0.0192	0.23
0.030	0.5785	115.85	0.0164	0.45
0.040	0.2833	114.88	0.0107	0.73
0.050	0.1364	107.39	0.0064	0.87
0.060	0.0621	99.40	0.0035	0.94
0.065	0.0413	95.86	0.0026	0.96
0.070	0.0274	92.75	0.0018	0.97
0.080	0.0120	87.90	0.0009	0.99
0.090	0.0053	84.67	0.0005	0.99
0.100	0.0023	82.65	0.0002	1.00

LIMIT NUMBER FOR REDUCED INSPECTION IS: 7

WHAT DO YOU WISH TO DO?

- 1-DERIVE PLANS TO MINIMIZE ASN
- 2-DERIVE PLANS TO MINIMIZE AFI
- 3-EVALUATE MIL-STD-105D SAMPLING SCHEME
- 4-EXIT THIS PROGRAM

ENTER CHOICE  
4  
FORTRAN STOP

APPENDIX B  
Program Listing

Program QMAIN.FOR

```

0001
0002 *****
0003 CONTROLING PROGRAM FOR THE QUALITY CONTROL
0004 INSPECTION SAMPLING SOFTWARE PACKAGE
0005
0006 DR. RICHARD S. LEAVENWORTH
0007 DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING
0008 UNIVERSITY OF FLORIDA
0009 GAINESVILLE, FLORIDA 32611
0010 C*****
0011 C
0012 BYTE OUTFIL(8)
0013 COMMON/BLK1/N25, N2I
0014 COMMON/BLK2/PS, PL
0015 COMMON/BLK3/N1
0016 COMMON/BLK4/ALPHA, BETA
0017 COMMON/BLK5/PO, P1
0018 COMMON/BLK6/C1, C2
0019 COMMON/BLK7/SUMLOG(4000)
0020 COMMON/BLK8/N
0021 COMMON/BLK9/C2MAX, C1MAX(15)
0022 COMMON/BLK10/NS, NL
0023 COMMON/BLK11/ASN, ASNMAX
0024 COMMON/BLK12/OUTFIL
0025 C
0026 NUM=0
0027 10 NUM=NUM+1
0028 WRITE(5, 15)
0029 15 FORMAT(////////, 23(' '), 'WHAT DO YOU WISH TO DO?')
0030 WRITE(5, 20)
0031 20 FORMAT(//, 15(' '), '1-DERIVE PLANS TO MINIMIZE ASN')
0032 WRITE(5, 25)
0033 25 FORMAT(/, 15(' '), '2-DERIVE PLANS TO MINIMIZE AFI')
0034 WRITE(5, 30)
0035 30 FORMAT(/, 15(' '), '3-EVALUATE MIL-STD-105D SAMPLING SCHEME')
0036 WRITE(5, 35)
0037 35 FORMAT(/, 15(' '), '4-EXIT THIS PROGRAM')
0038 WRITE(5, 40)
0039 40 FORMAT(////////, ' ENTER CHOICE')
0040 READ(5, 45) ICH
0041 45 FORMAT(I1)
0042 C
0043 IF ((ICH .NE. 1).AND.(ICH .NE. 2).AND.(ICH .NE. 3).AND.
0044 $(ICH .NE. 4)) THEN
0045 WRITE(5, 50)
0046 50 FORMAT(/, 15(/), ' YOU MUST ENTER 1, 2, 3, OR 4')
0047 GO TO 10
0048 ENDIF
0049 C
0050 IF((ICH .NE. 4) .AND. (NUM .EQ. 1)) THEN
0051 WRITE(5, *) ' WHAT DO YOU WISH TO CALL YOUR OUTPUT FILE?'
0052 READ(5, 55) OUTFIL
0053 55 FORMAT(10A1)
0054 CALL ASSIGN(1, OUTFIL)
0055 ENDIF
0056 C
0057 GO TO (60, 65, 70, 75), ICH

```

QMAIN\$MAIN

```

0058 C
0059 60 CALL QASN
0060 GO TO 10
0061 65 CALL QAFI
0062 GO TO 10
0063 70 CALL QMII
0064 GO TO 10
0065 75 STOP
0066 END

```

Program QASN.FOR

```

0001
0002
0003 *****
0004 QUALITY CONTROL DOUBLE SAMPLING PROGRAM TO ANALYSE
0005 DOUBLE SAMPLING PLANS, ASN(PO) AND ASNMAX.
0006 BINOMIAL AND POISSON PROBABILITY DISTRIBUTIONS USED.
0007
0008 PROGRAMED BY R. WARREN RANGARAJAN
0009 DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING
0010 UNIVERSITY OF FLORIDA
0011 GAINESVILLE, FLORIDA 32611
0012 *****
0013
0014 SUBROUTINE QASN
0015 DOUBLE PRECISION SUMLOG
0016 INTEGER C, C1, C2, C1MIN, C2MIN, R1, R11
0017 BYTE OUTFIL(8)
0018
0019 COMMON/BLK1/N2S, N2L
0020 COMMON/BLK2/PS, PL
0021 COMMON/BLK3/N1
0022 COMMON/BLK4/ALPHA, BETA
0023 COMMON/BLK5/PO, P1
0024 COMMON/BLK6/C1, C2
0025 COMMON/BLK7/SUMLOG(4000)
0026 COMMON/BLK8/N
0027 COMMON/BLK9/C2MAX, C1MAX(15)
0028 COMMON/BLK10/NS, NL
0029 COMMON/BLK11/ASN, ASNMAX
0030 COMMON/BLK12/OUTFIL
0031 *****
0032 BEGINNING INITIALIZATION
0033 *****
0034 N=0
0035 C2=1000
0036 ASNMIN=15000.
0037 C=-1
0038 *****
0039 INPUT FORMAT
0040 *****
0041 10 WRITE (5, 15)
0042 15 FORMAT (/// ' CODES FOR SELECTING APPR. PROB. DIST. '///
0043 115X, 'BINOMIAL ', 12X, '=1',
0044 2/15X, 'POISSON ', 13X, '=2')
0045 READ (5, *) K
0046 IF (K.GT.2.OR.K.LT.1) GOTO 10
0047 20 WRITE(5, 25)
0048 25 FORMAT(10X, 'SELECT'//16X, 'SAMPLE PLANS ONLY =1'
0049 1/16X, 'ASN VALUES ONLY =2'
0050 2/16X, 'OR BOTH =3')
0051 READ(5, *) KOPT
0052 IF (KOPT.GT.3.OR.KOPT.LT.1) GOTO 20
0053 WRITE (5, 30)
0054 30 FORMAT(10X, 'INPUT ALPHA ')
0055 READ (5, *) ALPHA
0056 WRITE (5, 35)
0057 35 FORMAT(10X, 'INPUT BETA ')
0058 READ (5, *) BETA

```

0058

```
0059 WRITE (5,40)
0060 40 FORMAT(10X, 'INPUT P0 ')
0061 READ (5,*) P0
0062 WRITE(5,45)
0063 45 FORMAT(10X, 'INPUT P1 ')
0064 READ (5,*) P1
0065 WRITE(5,50)
0066 50 FORMAT( 5X, 'INPUT A SEED FOR SINGLE SAMPLING NO. '//
0067 1' IF NO SEED AVAILABLE ENTER ZERO AS THE SEED VALUE')
0068 READ(5,*) NS
0069 WRITE(5,55)
0070 55 FORMAT( 5X, 'INPUT A VALUE FOR (R1-C2) '//
0071 1' IF R1=C2 THEN THE VALUE WOULD BE 0'//
0072 2' IF R1>C2 THEN THE VALUE WOULD BE A POSITIVE NO.'//
0073 3' IF R1<C2 THEN THE VALUE WOULD BE A NEGATIVE NO.' )
0074 READ(5,*) R11
0075
0076 WRITE (1,60)
0077 60 FORMAT('1', ///10X, 'DEPT. OF ISE '
0078 1/, 10X, 'UNIVERSITY OF FLORIDA ' /
0079 2/5X, 5(' '), 'DOUBLE SAMPLING SYSTEM TO MINIMIZE ASN',
0080 35(' '), 2X, /)
0081 WRITE (5,65) ALPHA, BETA, P0, P1
0082 WRITE (1,65) ALPHA, BETA, P0, P1
0083 65 FORMAT(//10X, 'ALPHA =', F6.4, 5X, 'BETA =', F6.4,
0084 1/10X, 'P0 =', F6.4, 8X, 'P1 =', F6.4)
0085 WRITE(5,70) R11
0086 WRITE(1,70) R11
0087 70 FORMAT(/5X, 'REJECTION NO. OF FIRST SAMPLE (R1) = C2+(', I3, ')')
0088 MC1=10.0/(P1/P0)
0089 75 C=C+1
0090 C*****
0091 C SINGLE SAMPLING PROCEDURE BEGINS
0092 C*****
0093 80 NS=NS+1
0094 C*****
0095 C COMPUTATION OF LOWER BOUND OF SINGLE SAMPLING PLAN
0096 C*****
0097 IF(K.EQ.1) CALL PROBS1(NS, P1, C, BXLEC, N)
0098 IF(K.EQ.2) CALL PROBS2(NS, P1, C, BXLEC, N)
0099 IF(BXLEC.GT.BETA) GOTO 80
0100 NLT=NS-1
0101 NL=MAX0(1, NLT)
0102 C*****
0103 C COMPUTATION OF UPPER BOUND OF SINGLE SAMPLING PLAN
0104 C*****
0105 85 NL=NL+1
0106 IF(K.EQ.1) CALL PROBS1(NL, P0, C, BXLEC)
0107 IF(K.EQ.2) CALL PROBS2(NL, P0, C, BXLEC)
0108 IF(BXLEC.GE.(1-ALPHA)) GOTO 85
0109 NL=NL-1
0110 C*****
0111 C TEST FOR FEASIBILITY
0112 C*****
0113 IF(NS.GT.NL) GOTO 75
0114 WRITE (5,90)
```



```

0115      WRITE(1, 70)
0116      90 FORMAT(10X, 'SINGLE SAMPLING PLAN')
0117      WRITE(5, 95) C, NS, NL
0118      WRITE(1, 95) C, NS, NL
0119      95 FORMAT(10X, 'ACCEPTANCE NO. (C) =', I2
0120      1, /10X, 'LOWER BOUND ON N (NS) =', I4
0121      2, /10X, 'UPPER BOUND ON N (NL) =', I4)
0122      C*****
0123      C      COMPUTATION OF DOUBLE SAMPLING PLAN BEGINS FOR EACH VALUE OF C2
0124      C*****
0125      IF(C.LT.C2) MC=C+MC1-1
0126      C2=C
0127      C
0128      R1 C2+R11
0129      C
0130      DO 140 K1 1, C2
0131      C1=K1-1
0132      C*****
0133      C      CALL SUBROUTINE TO COMPUTE THE FIRST SAMPLE NUMBER
0134      C*****
0135      CALL TRY1(NTRY, C1, P1, NS, BETA, K)
0136      N1=NTRY
0137      IF(NTRY.GT.NS) GOTO 145
0138      C
0139      C
0140      WRITE(5, 100)
0141      WRITE(1, 100)
0142      100  FORMAT(/10X, 'DOUBLE SAMPLING PLANS', /)
0143      WRITE(5, 105) C1, C2
0144      WRITE(1, 105) C1, C2
0145      105  FORMAT(/10X, 'FOR C1=', I2, 2X, 'C2=', I2, /)
0146      C
0147      NTEMP=N1
0148      IF(KOPT.EQ.1) WRITE(1, 110)
0149      IF(KOPT.EQ.1) WRITE(5, 110)
0150      IF(KOPT.EQ.2) WRITE(1, 112)
0151      IF(KOPT.EQ.2) WRITE(5, 112)
0152      IF(KOPT.EQ.3) WRITE(5, 115)
0153      IF(KOPT.EQ.3) WRITE(1, 115)
0154      110  FORMAT(11X, '(N1)', 3X, '(N2S) ( N2 ( (N2L)',
0155      1    6X, 'PS', 10X, 'PL'//)
0156      112  FORMAT(10X, 'N1', 3X, 'N2S', 4X, 'ASNMAX', 6X, 'ASN', //)
0157      115  FORMAT(10X, '(N1)', 3X, '(N2S)', 3X, '(N2L)', 4X,
0158      1    'ASNMAX', 6X, 'ASN'//)
0159      NTEMP1=NS
0160      C*****
0161      C      COMPUTATION OF SECOND SAMPLE FOR EACH VALUE OF FIRST SAMPLE
0162      C*****
0163      ASN=FLOAT(NS)*10
0164      DO 135 IZ=NTEMP, NTEMP1
0165      I=IZ
0166      IF ((NTEMP1-I) .LE. (C2*1.5)) GOTO 135
0167      C*****
0168      C      CALL SUBROUTINE TO COMPUTE SECOND SAMPLE
0169      C*****
0170      CALL TRY2(NS, NL, K, I, R1)
0171      IF(KOPT.NE.1) GOTO 125

```

GASN

```
0172 WRITE(1,120) I, N2S, N2L, PS, PL
0173 WRITE(5,120) I, N2S, N2L, PS, PL
0174 120 FORMAT(10X, I4, 5X, I4, 1X, N2, 1X, I4, 4X, F8.6, 4X, F8.6)
0175 GOTO 135
0176 C*****
0177 C TEST FOR FEASIBILITY
0178 C*****
0179 125 IF(N2S.GT.N2L) GOTO 135
0180 IF(N2S.LT.(C2-C1).OR.I.LE.C2) GOTO 135
0181 ASNTEM=ASN
0182 C*****
0183 C CALL SUBROUTINE TO COMPUTE ASN(PO) AND ASNMAX VALUES
0184 C*****
0185 CALL ASN(MC, NS, K, I, KOPT, C1MIN, C2MIN, N1MIN, N2MIN, ASNMIN)
0186 IF(KOPT.NE.3) GOTO 135
0187 WRITE(1,130) I, N2S, N2L, ASNMAX, ASN
0188 WRITE(5,130) I, N2S, N2L, ASNMAX, ASN
0189 130 FORMAT(10X, I3, 5X, I3, 5X, I3, 4X, 2(F8.4, 3X))
0190 IF(ASN.GT.ASNTEM) GOTO 140
0191 135 CONTINUE
0192 C
0193 140 CONTINUE
0194 C
0195 145 IF(C.LT.MC) GOTO 75
0196 IF(KOPT.EQ.1) GO TO 155
0197 WRITE(1,150) ASNMIN, N1MIN, N2MIN, C1MIN, C2MIN
0198 WRITE(5,150) ASNMIN, N1MIN, N2MIN, C1MIN, C2MIN
0199 150 FORMAT(///, 10X, 'GLOBAL MINIMUM ASN(PO)=', F8.2, //
0200 110X, 'CORRESPONDING N1 =', I5//
0201 210X, 'CORRESPONDING N2S =', I5//
0202 310X, 'CORRESPONDING C1 =', I2//
0203 410X, 'CORRESPONDING C2 =', I2)
0204 C
0205 155 RETURN
0206 END
```

```
0001 SUBROUTINE TRY1(NTRY, C1, P, NL, BETA, K)
0002 C*****
0003 C THIS SUBROUTINE COMPUTES FIRST SAMPLE NUMBER OF DOUBLE
0004 C SAMPLING PLAN BY AN INTEGER FORM OF BISECTION METHOD
0005 C*****
0006 INTEGER C1
0007 C
0008 NLARGE=NL
0009 NSMALL=0
0010 C
0011 10 NTRY=(NSMALL+NLARGE)/2.0
0012 C*****
0013 C CALL APPROPRIATE PROBABILITY SUBROUTINE FOR PROB. CALCULATIONS
0014 C*****
0015 15 IF(K.EQ.1) CALL PROBS1(NTRY, P, C1, BXLEC)
0016 IF(K.EQ.2) CALL PROBS2(NTRY, P, C1, BXLEC)
0017 IF(BXLEC.LE.BETA) GOTO 20
0018 NSMALL=NTRY
0019 GOTO 25
0020 20 NLARGE=NTRY
0021 25 IF(NSMALL.NE.(NLARGE-1)) GOTO 10
0022 NTRY=NLARGE
0023 RETURN
0024 END
```

```

0001      SUBROUTINE TRY2(NS, NL, K, J, R1)
0002      C *****
0003      C THIS SUBROUTINE COMPUTES THE SECOND SAMPLE NUMBER OF
0004      C THE DOUBLE SAMPLING NUMBER BY AN INTEGER BISECTION
0005      C METHOD. SEVERAL TESTS ARE DONE TO LOCATE THE PARAMETER
0006      C AT ITS TRUE POSITION.
0007      C *****
0008      C INTEGER C1, C2, R1
0009      C
0010      COMMON/BLK1/N2S, N2L
0011      COMMON/BLK2/PS, PL
0012      COMMON/BLK3/N1
0013      COMMON/BLK4/ALPHA, BETA
0014      COMMON/BLK5/PO, P1
0015      COMMON/BLK6/C1, C2
0016      C
0017      K1=C1+1
0018      C *****
0019      C SET LIMITS FOR COMPUTING N2S
0020      C *****
0021      NSMALL=NS-J
0022      NLARGE=NSMALL
0023      C *****
0024      C INDEXING TO SPECIFY WHAT BOUND (N2S OR N2L)
0025      C IS BEING COMPUTED
0026      C *****
0027      I=1
0028      C *****
0029      C INITIAL TEST AT EACH LIMIT
0030      C *****
0031      CALL PROBD1(J, NSMALL, P1, DPR0B, K, R1)
0032      IF(DPROB.LE.BETA) GOTO 40
0033      C *****
0034      C BISECTION METHOD
0035      C *****
0036      NLARGE=NL
0037      10 NTRY=(NSMALL+NLARGE)/2.0
0038      GOTO (15, 20), I
0039      C
0040      15 CALL PROBD1(J, NTRY, P1, DPR0B, K, R1)
0041      IF(DPROB.LE.BETA) GOTO 30
0042      GOTO 25
0043      20 CALL PROBD1(J, NTRY, PO, DPR0B, K, R1)
0044      IF(DPROB.LT.(1-ALPHA)) GOTO 30
0045      25 NSMALL=NTRY
0046      GOTO 35
0047      30 NLARGE=NTRY
0048      35 IF((NLARGE-NSMALL).GT.1) GOTO 10
0049      C *****
0050      C CHECK THE INDEX TO FIND WHERE THE PROCESS IS
0051      C *****
0052      GOTO (40, 45), I
0053      C *****
0054      C CHANGE THE INDEX AFTER N2S COMPUTATION
0055      C *****
0056      40 I=I+1
0057      C *****

```

```

0058      C      TESTING EACH POSSIBLE CASES TO LOCATE
0059      C      THE LOWER BOUND AT ITS TRUE POSITION
0060      C      *****
0061      N2S=MAXO(O, NLARGE)
0062      CALL PRORD1(J, NLARGE, P1, DPROB, K, R1)
0063      PS=DPROB
0064      MTEMP=NLARGE-1
0065      NSMALL=MAXO(O, MTEMP)
0066      NLARGE=NI
0067      GOTO 10
0068      45 N2L=NSMALL
0069      CALL PRORD1(J, NSMALL, PO, DPROB, K, R1)
0070      PL=DPROB
0071      CALL PRORD1(J, NLARGE, PO, DPROB, K, R1)
0072      IF(DPROB.GE.(1-ALPHA)) N2L=NLARGE
0073      IF(DPROB.GE.(1-ALPHA)) PL=DPROB
0074      C
0075      50 RETURN
0076      END

```

```

0001      SUBROUTINE PROBS2(NN, P, C, BXLEC)
0002      C*****
0003      C      THIS SUBROUTINE COMPUTES CUMULATIVE POISON
0004      C      PROBABILITIES
0005      C*****
0006      INTEGER C
0007      C
0008      PP=P*NN
0009      TERM=1.0
0010      SUM=TERM
0011      C
0012      IF(C.EQ.0) GOTO 15
0013      DO 10 I=1,C
0014          TERM=TERM*PP/I
0015          SUM=SUM+TERM
0016      10 CONTINUE
0017      C
0018      15 BXLEC=SUM/EXP(PP)
0019      C
0020      RETURN
0021      END

```

```

0001      SUBROUTINE ASNN(MC, NS, K, N11, KOPT, C1MIN, C2MIN, N1MIN, N2MIN, ASNMIN)
0002      C*****
0003      C      THIS SUBROUTINE COMPUTES ASN(PO) VALUES AND
0004      C      ASNMAX VALUES.
0005      C*****
0006      DOUBLE PRECISION SUMLOG
0007      INTEGER C1MIN, C2MIN
0008      C
0009      COMMON/BLK1/N2S, N2L
0010      COMMON/BLK3/N1
0011      COMMON/BLK4/ALPHA, BETA
0012      COMMON/BLK5/PO, P1
0013      COMMON/BLK6/I1, I2
0014      COMMON/BLK7/SUMLOG(4000)
0015      COMMON/BLK8/N
0016      COMMON/BLK11/ASN, ASNMAX
0017      COMMON/BLK12/OUTFIL
0018      C*****
0019      C      INITIALIZATION
0020      C      COMPUTE P* (MAXIMUM PROB. FOR ASNMAX)
0021      C*****
0022      J=I1+1
0023      XXX=0.0
0024      IF(I1.GT.0) XXX=SUMLOG(I1)
0025      AKONST=10.**(SUMLOG(I2)+SUMLOG(N11-I2-1)-XXX-SUMLOG(N11-I1-1))
0026      TEMP=1.0/FLOAT(I2-I1)
0027      AKONST=AKONST**TEMP
0028      PSTAR=AKONST/(1.0+AKONST)
0029      IF(K.EQ.1) CALL PROBS1(N11, PSTAR, I2, BXLEC)
0030      IF(K.EQ.2) CALL PROBS2(N11, PSTAR, I2, BXLEC)
0031      TEMP=BXLEC
0032      IF(K.EQ.1) CALL PROBS1(N11, PSTAR, I1, BXLEC)
0033      IF(K.EQ.2) CALL PROBS2(N11, PSTAR, I1, BXLEC)
0034      TEMP1=TEMP-BXLEC
0035      ASNMAX=FLOAT(N11)+N2S*TEMP1
0036      C
0037      IF(K.EQ.1) CALL PROBS1(N11, PO, I2, BXLEC)
0038      IF(K.EQ.2) CALL PROBS2(N11, PO, I2, BXLEC)
0039      TEMP=BXLEC
0040      IF(K.EQ.1) CALL PROBS1(N11, PO, I1, BXLEC)
0041      IF(K.EQ.2) CALL PROBS2(N11, PO, I1, BXLEC)
0042      TEMP2=TEMP-BXLEC
0043      ASN=FLOAT(N11)+N2S*TEMP2
0044      IF(ASNMAX.GT.NS.OR.ASN.GT.ASNMIN) GOTO 10
0045      ASNMIN=ASN
0046      N1MIN=N11
0047      N2MIN=N2S
0048      C1MIN=I1
0049      C2MIN=I2
0050      C
0051      10 IF(KOPT.NE.2) GOTO 20
0052      WRITE(1,15) N11, N2S, ASNMAX, ASN
0053      WRITE(5,15) N11, N2S, ASNMAX, ASN
0054      15 FORMAT(10X, 2(I3, 3X), 2(F8.4, 3X))
0055      C
0056      20 RETURN
0057      END

```

Program QAFI.FOR

```

0001 C
0002 C*****
0003 C    QUALITY CONTROL PROGRAM TO DERIVE DOUBLE SAMPLING
0004 C    PLANS TO MINIMIZE AVERAGE FRACTION INSPECTED.
0005 C
0006 C    PROGRAMMED BY JO ELLEN WALKER
0007 C    DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING
0008 C    UNIVERSITY OF FLORIDA
0009 C    GAINESVILLE, FLORIDA 32611
0010 C*****
0011 C
0012 C    SUBROUTINE GAFI
0013 C    INTEGER C1,C2,C,CSTAR,C2M1
0014 C    INTEGER R1,R1M1
0015 C    DOUBLE PRECISION SUMLOG
0016 C    BYTE OUTFIL(8)
0017 C    COMMON/BLK4/ALPHA,BETA
0018 C    COMMON/BLK6/C1,C2
0019 C    COMMON/BLK7/SUMLOG(4000)
0020 C    COMMON/BLK8/N
0021 C    COMMON/BLK12/OUTFIL
0022 C*****
0023 C    INPUT PARAMETERS
0024 C*****
0025 C    KINDEX=1
0026 C    WRITE(5,10)
0027 C    10 FORMAT(10X,'ENTER VALUE OF ALPHA')
0028 C    READ(5,*)ALPHA
0029 C    WRITE(5,15)
0030 C    15 FORMAT(10X,'ENTER VALUE OF PO')
0031 C    READ(5,*)PO
0032 C    WRITE(5,20)
0033 C    20 FORMAT(10X,'ENTER AQGL VALUE')
0034 C    READ(5,*)AQGL
0035 C    WRITE(5,25)
0036 C    25 FORMAT(10X,'ENTER LOT SIZE')
0037 C    READ(5,*)NNN
0038 C
0039 C
0040 C
0041 C    WRITE(1,28)
0042 C    28 FORMAT('1',///10X,'DEPT. OF ISE '
0043 C    $/,10X,'UNIVERSITY OF FLORIDA '/
0044 C    $/5X,5('*'),'DOUBLE SAMPLING SYSTEM TO MINIMIZE AFI',
0045 C    $5('*'),2X,/)
0046 C    WRITE(5,30)ALPHA
0047 C    WRITE(1,30)ALPHA
0048 C    30 FORMAT(10X,'ALPHA=',2X,F8.6)
0049 C    WRITE(5,35)PO
0050 C    WRITE(1,35)PO
0051 C    35 FORMAT(10X,'PO=',2X,F8.6)
0052 C    WRITE(5,40)AQGL
0053 C    WRITE(1,40)AQGL
0054 C    40 FORMAT(10X,'AQGL=',2X,F8.6)
0055 C    WRITE(5,45)NNN
0056 C    WRITE(1,45)NNN
0057 C    45 FORMAT(10X,'N=',2X,I6)

```

GAF 1

```
0058 C*****
0059 C    COMPUTE SINGLE SAMPLING PLAN
0060 C*****
0061 C
0062 C    INITIALIZATION
0063 C
0064 C*****
0065     NS=1
0066     C=0
0067     N=0
0068     AFI=1. DO
0069 C
0070     WRITE(5, 50)
0071     WRITE(1, 50)
0072     50 FORMAT(///10X, '*** SINGLE SAMPLING PLAN ***')
0073 C*****
0074 C    FIND NS,C COMBO THAT SATISFIES L(PO) G.T. 1-ALPHA
0075 C*****
0076     55 CALL PROBS1(NS, PO, C, BXLEC)
0077     IF(BXLEC.LT. (1. DO-ALPHA))C=C+1
0078     IF(BXLEC.LT. (1. DO-ALPHA))NS=C+1
0079 C*****
0080 C    SEARCH TO FIND MIN NS VALUE SUCH THAT AOGL L.T. AOGL*
0081 C*****
0082     60 CALL SEARCH(NNN, C, NS, SAQG)
0083     IF(SAQG.LE. AOGL)GOTO 75
0084     NSTEMP=NNN
0085     CALL SEARCH(NNN, C, NSTEMP, SAQG)
0086     IF(SAQG.GT. AOGL)C=C+1
0087 C
0088 C
0089     IF(SAQG.GT. AOGL)NS=C+1
0090 C
0091 C
0092     BL=NS
0093     BH=NNN
0094     65 NSTEMP=IIDINT((BL+BH)/2. DO)
0095     CALL SEARCH(NNN, C, NSTEMP, SAQG)
0096     IF(SAQG.LE. AOGL)BH=NSTEMP
0097 C
0098     IF(SAQG.LE. AOGL)AOQ=SAQG
0099 C
0100     IF(SAQG.GT. AOGL)BL=NSTEMP
0101     IF((BH-BL).EQ. 1. DO)GOTO 70
0102     GOTO 65
0103     70 NS=NSTEMP
0104     IF(SAQG.GT. AOGL)NS=BH
0105 C*****
0106 C    CHECK THAT NS,C COMBO STILL SATISFIES L(PO) CONSTRAINT
0107 C*****
0108     75 CALL PROBS1(NS, PO, C, BXLEC)
0109     IF(BXLEC.GE. (1. DO-ALPHA))GOTO 80
0110     C=C+1
0111 C
0112     NS=C+1
0113 C
0114     GOTO 60
```

```

0115 C*****
0116 C   COMPUTE AFI
0117 C*****
0118   80 ATIPO=NS*BXL EC+NNN*(1.DO-BXL EC)
0119   AFIPO=ATIOPO/NNN
0120 C
0121   NSTAR=NS
0122 C
0123   WRITE(5,85)NS
0124   WRITE(1,85)NS
0125   85 FORMAT(//10X,'NS=',I3)
0126   WRITE(5,90)C
0127   WRITE(1,90)C
0128   90 FORMAT(10X,'C=',I2)
0129   WRITE(5,95)AFIPO
0130   WRITE(1,95)AFIPO
0131   95 FORMAT(10X,'AFI(PO)=',F8.6)
0132 C
0133   100 CSTAR=C
0134 C*****
0135 C   DOUBLE SAMPLING
0136 C*****
0137 C
0138   WRITE(5,105)
0139   WRITE(1,105)
0140   105 FORMAT(///10X,'** DOUBLE SAMPLING PLANS **\
0141   PLAN=1.DO
0142   C2=CSTAR
0143   110 R1=C2+1
0144   C2M1=C2-1
0145   C1PO=C1+1
0146   R1M1=R1-1
0147   DDATI=NNN
0148   TTCMIN=1.
0149 C
0150   C1=0
0151   JJ=0
0152 C
0153   WRITE(5,115)
0154   WRITE(1,115)
0155   115 FORMAT(///10X,'C1',6X,'C2',7X,'N1',8X,'N2',9X,'AFI',//)
0156 C*****
0157 C   CALCULATE FIRST SAMPLE NUMBER
0158 C
0159 C   FROM RESULTS OF PREVIOUS RUNS, IT WAS FOUND THAT N1 IS NOT LESS
0160 C   THAN NSTAR/8.  THUS, THE INITIAL VALUE OF N1 IS SET ACCORDINGLY.
0161 C*****
0162 C
0163   120 DO 165 LL=INT(NSTAR/8),NSTAR
0164   N1=LL
0165   IF(N1.LT.C2)N1=R1M1
0166 C*****
0167 C   CHECK B(N1,PO,C2) G.T. 1-ALPHA CONSTRAINT
0168 C*****
0169   125 CALL PROBS1(N1,PO,C2,BXL EC)
0170   IF(BXL EC.LT.(1.-ALPHA))GOTO 175
0171 C*****

```



```

0172 C CALCULATE SECOND SAMPLE
0173 C *****
0174 N2=NSTAR-N1
0175 C *****
0176 C CHECK THAT DOUBLE PROBABILITY G.T. 1-ALPHA
0177 C *****
0178 130 CALL PROBD1(N1,N2,PO,DPROB,KINDEX,R1)
0179 IF(DPROB.GE.(1.-ALPHA))GOTO 135
0180 IF(C1.EQ.C2M1)GOTO 175
0181 C1=C1+1
0182 GOTO 130
0183 C *****
0184 C FIND N2 VALUE SATISFYING AQGL L.T. AQGL*
0185 C *****
0186 135 CALL SRCH2(NNN,N1,N2,AQ1)
0187 IF(AQ1.LE.AQGL)GOTO 150
0188 C *****
0189 C N2 WILL NOT BE LESS THAN N1*9, THE INITIAL LOWER BOUND ON N2.
0190 C *****
0191 N2TEMP=N1*9
0192 C
0193 CALL SRCH2(NNN,N1,N2TEMP,AQ1)
0194 IF(AQ1.GT.AQGL)GOTO 165
0195 C
0196 BL=N2
0197 BH=N1*9
0198 140 N2TEMP=INT((BL+BH)/2)
0199 CALL SRCH2(NNN,N1,N2TEMP,AQ1)
0200 IF(AQ1.LE.AQGL)BH=N2TEMP
0201 IF(AQ1.LE.AQGL)FAQG=AQ1
0202 C
0203 IF(AQ1.GT.AQGL)BL=N2TEMP
0204 IF((BH-BL).EQ.1.)GOTO 145
0205 GOTO 140
0206 145 N2=N2TEMP
0207 IF(AQ1.GT.AQGL)N2=BH
0208 C *****
0209 C CHECK THAT BINOMIAL PROBABILITIES ARE G.T. 1-ALPHA
0210 C *****
0211 CALL PROBD1(N1,N2,PO,DPROB,KINDEX,R1)
0212 IF(DPROB.LT.(1.-ALPHA))GOTO 165
0213 150 CALL PROBS1(N1,PO,C1,BXLEC)
0214 PA2=DPROB-BXLEC
0215 C *****
0216 C COMPUTE ATI
0217 C *****
0218 DATI=N1*DPROB+N2*PA2+NNN*(1.-DPROB)
0219 C *****
0220 C IF THE ATI INCREASES, CONTINUE FOR 5 ADDITIONAL INCREASING
0221 C ITERATIONS. THEN INCREMENT C1 AND CONTINUE.
0222 C *****
0223 IF(JJ.EQ.4)GOTO 155
0224 IF(DATI.GE.DDATI)JJ=JJ+1
0225 IF(DATI.GE.DDATI)GOTO 165
0226 DDATI=DATI
0227 DDAQ=FAQG
0228 C

```

```

0229      K1=C1
0230      K2=C2
0231      K3=N1
0232      K4=N2
0233      C
0234      GOTO 165
0235      C*****
0236      C      MINIMUM OF CELL (TCMIN) FOUND
0237      C*****
0238      155 TCMIN=DDATI/NNN
0239      DDATI=NNN
0240      C
0241      WRITE(5,160)K1,K2,K3,K4,TCMIN
0242      WRITE(1,160)K1,K2,K3,K4,TCMIN
0243      160 FORMAT(10X,I2,6X,I2,5X,I4,6X,I4,7X,FB.6)
0244      C*****
0245      C      IF MINIMUM OF COLUMN IS FOUND, INCREASE C2
0246      C*****
0247      IF(TCMIN.GE.TTCMIN)GOTO 170
0248      KK1=K1
0249      KK2=K2
0250      KK3=K3
0251      KK4=K4
0252      TTCMIN=TCMIN
0253      TTAQG=DDAQQ
0254      C
0255      IF (C1.EQ.C2M1) GOTO 175
0256      C1=C1+1
0257      C
0258      JJ=0
0259      N1=KK3
0260      C
0261      GOTO 125
0262      C
0263      165 CONTINUE
0264      C*****
0265      C      MINIMUM OF COLUMN (TMIN) FOUND
0266      C*****
0267      170 TMIN=TTCMIN
0268      TADG=TTAQQ
0269      C*****
0270      C      IF MINIMUM SAMPLING PLAN FOUND, STOP
0271      C*****
0272      IF(TMIN.GE.PLAN)GOTO 190
0273      C
0274      PLAN=TMIN
0275      KKK1=KK1
0276      KKK2=KK2
0277      KKK3=KK3
0278      KKK4=KK4
0279      175 C2=C2+1
0280      C*****
0281      C      NEW BOUNDS ON SAMPLING PLAN CALCULATED FOR NEW VALUE OF C2
0282      C*****
0283      180 CALL SEARCH(NNN,C2,NSTAR,AQG)
0284      IF(AQG.LE.AOGL)GOTO 185
0285      IF(NSTAR.GT.NNN)GOTO 175
0286      NSTAR=NSTAR+1
0287      GOTO 180
0288      C
0289      C
0290      185 CALL PROBS1(NSTAR,PO,C2,BXLEC)
0291      GOTO 110
0292      C
0293      C
0294      190 WRITE(5,195)
0295      WRITE(1,195)
0296      195 FORMAT(/10X,'SAMPLING PLAN MINIMUMS')
0297      WRITE(5,200)KKK1,KKK2
0298      WRITE(1,200)KKK1,KKK2
0299      200 FORMAT(/10X,'C1=',I2,2X,'C2=',I2)
0300      WRITE(5,205)KKK3,KKK4
0301      WRITE(1,205)KKK3,KKK4
0302      205 FORMAT(10X,'N1=',I3,2X,'N2=',I3)
0303      WRITE(5,210)PLAN
0304      WRITE(1,210)PLAN
0305      210 FORMAT(10X,'MINIMUM AFI=',FB.6)
0306      215 RETURN
0307      END

```

```

0001          SUBROUTINE SEARCH(NNN, C, NS, AQQ)
0002 C*****
0003 C          SEARCH TO FIND VALUE OF PSTAR USING GOLDEN
0004 C          SECTION METHOD. INITIAL LIMITS OF 0 AND 1
0005 C*****
0006          INTEGER C
0007          DOUBLE PRECISION SUMLOG
0008          COMMON/BLK7/SUMLOG(4000)
0009          COMMON/BLK8/N
0010 C
0011          A1=0. DO
0012          A2=1. DO
0013          T=1. D-3
0014          R=5. D-1*(DSQRT(5. DO)-1. DO)
0015          H=A2-A1
0016          PLFT=A1+(R*R)*H
0017          PRT=A1+(R*H)
0018 C
0019 C
0020          CALL PROBS1(NS, PLFT, C, BXLEC)
0021          ATI=(NS*BXLEC)+NNN*(1. DO-BXLEC)
0022          AFI1=ATI/NNN
0023          AQQ1=PLFT*(1. DO-AFI1)
0024          CALL PROBS1(NS, PRT, C, BXLEC)
0025          ATI=(NS*BXLEC)+NNN*(1. DO-BXLEC)
0026          AFI2=ATI/NNN
0027          AQQ2=PRT*(1. DO-AFI2)
0028          GOTO 110
0029 C
0030 C
0031          100 CALL PROBS1(NS, PLFT, C, BXLEC)
0032          ATI=(NS*BXLEC)+NNN*(1. DO-BXLEC)
0033          AFI1=ATI/NNN
0034          AQQ1=PLFT*(1. DO-AFI1)
0035          GO TO 110
0036 C
0037          105 CALL PROBS1(NS, PRT, C, BXLEC)
0038          ATI=(NS*BXLEC)+NNN*(1. DO-BXLEC)
0039          AFI2=ATI/NNN
0040          AQQ2=PRT*(1. DO-AFI2)
0041 C
0042          110 IF(AQQ1.LT.AQQ2) GOTO 115
0043          A2=PRT
0044          H=PRT-A1
0045          IF(ABS(PRT-PLFT).LE.T)GOTO 120
0046 C
0047          PRT=PLFT
0048          PLFT=A1+(R*R)*H
0049          AQQ2=AQQ1
0050          GO TO 100
0051          115 A1=PLFT
0052          H=A2-PLFT
0053 C
0054          IF(ABS(PRT-PLFT).LE.T)GOTO 120
0055          PLFT=PRT
0056          PRT=A1+R*H
0057          AQQ1=AQQ2

```

```

0001 SUBROUTINE SRCH2(NNN, N1, N2, A0G)
0002 C
0003 INTEGER C1, C2, R1
0004 COMMON/BLK6/C1, C2
0005 C
0006 KINDEX=1
0007 A1=0.
0008 A2=1.
0009 T=.0001
0010 R=.5*(DSQRT(5.00)-1.)
0011 H=A2-A1
0012 PLFT=A1+(R*R)*H
0013 PRT=A1+(R*H)
0014 C
0015 C
0016 CALL PROBS1(N1, PLFT, C1, PA1)
0017 CALL PROBD1(N1, N2, PLFT, DPROB, KINDEX, R1)
0018 PA2=DPROB-PA1
0019 ATI=DPROB*N1+PA2*N2+NNN*(1.-DPROB)
0020 AFI1=ATI/NNN
0021 A0G1=PLFT*(1.-AFI1)
0022 CALL PROBS1(N1, PRT, C1, PA1)
0023 CALL PROBD1(N1, N2, PRT, DPROB, KINDEX, R1)
0024 PA2=DPROB-PA1
0025 ATI=DPROB*N1+PA2*N2+NNN*(1.-DPROB)
0026 AFI2=ATI/NNN
0027 A0G2=PRT*(1.-AFI2)
0028 GOTO 110
0029 C
0030 C
0031 100 CALL PROBS1(N1, PLFT, C1, PA1)
0032 CALL PROBD1(N1, N2, PLFT, DPROB, KINDEX, R1)
0033 PA2=DPROB-PA1
0034 ATI=DPROB*N1+PA2*N2+NNN*(1.-DPROB)
0035 AFI1=ATI/NNN
0036 A0G1=PLFT*(1.-AFI1)
0037 GOTO 110
0038 105 CALL PROBS1(N1, PRT, C1, PA1)
0039 CALL PROBD1(N1, N2, PRT, DPROB, KINDEX, R1)
0040 PA2=DPROB-PA1
0041 ATI=DPROB*N1+PA2*N2+NNN*(1.-DPROB)
0042 AFI2=ATI/NNN
0043 A0G2=PRT*(1.-AFI2)
0044 C
0045 C
0046 110 IF(A0G1.LT.A0G2)GOTO 115
0047 C
0048 C
0049 A2=PRT
0050 H=PRT-A1
0051 IF(ABS(PRT-PLFT).LE.T)GOTO 120
0052 PRT=PLFT
0053 PLFT=A1+(R*R)*H
0054 A0G2=A0G1
0055 GOTO 100
0056 115 A1=PLFT
0057 C
0058 C
0059 H=A2-PLFT
0060 IF(ABS(PRT-PLFT).LE.T)GOTO 120
0061 PLFT=PRT
0062 PRT=A1+R*H
0063 A0G1=A0G2
0064 GOTO 105
0065 C
0066 C
0067 120 PS=(PLFT+PRT)/2.
0068 CALL PROBS1(N1, PS, C1, PA1)
0069 CALL PROBD1(N1, N2, PS, DPROB, KINDEX, R1)
0070 PA2=DPROB-PA1
0071 AFI=(DPROB*N1+PA2*N2+NNN*(1.-DPROB))/NNN
0072 A0G=PS*(1.-AFI)
0073 RETURN
0074 END

```

Program QMIL.FOR

```

0001 *****
0002      MIL STD 105D
0003 *****
0004      THIS IS AN INTERACTIVE PROGRAM TO EVALUATE
0005      THE MIL STD 105D SAMPLING SCHEME.
0006      (**)
0007      PROGRAMMED BY: AZMAT H. SIDDIQI
0008      (***)
0009      INDUSTRIAL AND SYSTEMS ENGINEERING DEPT
0010      (***)
0011      UNIVERSITY OF FLORIDA, GAINESVILLE,
0012      (***)
0013      FLORIDA 32611
0014      (***)
0015      C*****
0016      C
0017      SUBROUTINE QMIL
0018      BYTE OUTFIL(8)
0019      CHARACTER I*2, CDL, SP
0020      INTEGER FS, FA, SA, AR, SR, AC, AT, AE, SC, ST, Z, A, B
0021      INTEGER NN, NFS, NFA, NAE, NSA, NAR, NAC, NAT, TN, TFS, TFA, TAE, TSA
0022      INTEGER TAR, TAC, TAT, RN, RFA, RAC, RAR, RAC, RAT, FRR, FTR, FNR, SNR
0023      $ , STR, SRR, RFS, S1, S2, S3, S4, S
0024      COMMON/SCL/L, SP, S, I, NQ, T, R, CDL, FS, D, J, K, AGL, M, FA, SA,
0025      $ AR, SR, AC, AT, AE, SC, ST, Z, NTN, NFS, NFA, NAE, NSA, NAR, NAC, NAT,
0026      $ TN, TFS, TFA, TAE, TSA, TAR, TAC, TAT, RN, RFS, RFA, RAE, RAR, RAC, RAT
0027      $ , FRR, FTR, FNR, SNR, STR, SRR, NNNN
0028      COMMON/BLK12/OUTFIL
0029
0030      C
0031      C
0032      C
0033      C
0034      C
0035      C
0036      C
0037      C
0038      C
0039      C
0040      C
0041      C
0042      C
0043      C
0044      C
0045      C
0046      C
0047      C
0048      C
0049      C
0050      C
0051      C
0052      C
0053      C
0054      C
0055      C
0056      C
0057      C

```

```

0058      WRITE(1,2)
0059      WRITE(5,2)
0060      FORMAT(27,16X)  THESE PLAINS ARE
0061      CALL CODE
0062      CALL SS
0063      CALL INDEX
0064      CALL VALUES
0065      S=0
0066      IF(RFA.LT.100)THEN
0067          FRR=RFA+1
0068      ENDIF
0069      IF(RAC.LT.100)THEN
0070          RFA=RAC
0071          FRR=RFA+3
0072      ENDIF
0073      IF(RAT.LT.100)THEN
0074          RFA=RAT
0075          FRR=RFA+2
0076      ENDIF
0077      IF(RAE.LT.100)THEN
0078          RFA=RAE
0079          FRR=RFA+5
0080      ENDIF
0081      IF(RAR.LT.100)THEN
0082          RFA=RAR
0083          FRR=RFA+4
0084      ENDIF
0085      IF(TFA.LT.100)THEN
0086          FTR=TFA+1
0087      ENDIF
0088      IF(TAT.LT.100)THEN
0089          TFA=TAT
0090          FTR=TFA+2
0091      ENDIF
0092      IF(TAC.LT.100)THEN
0093          TFA=TAC
0094          FTR=TFA+3
0095      ENDIF
0096      IF(TAR.LT.100)THEN
0097          TFA=TAR
0098          FTR=TFA+4
0099      ENDIF
0100      IF(TAE.LT.100)THEN
0101          TFA=TAE
0102          FTR=TFA+5
0103      ENDIF
0104      IF(NFA.LT.100)THEN
0105          FNR=NFA+1
0106      ENDIF
0107      IF(NAT.LT.100)THEN
0108          NFA=NAT
0109          FNR=NFA+2
0110      ENDIF
0111      IF(NAC.LT.100)THEN
0112          NFA=NAC
0113          FNR=NFA+3
0114      ENDIF

```

```

0115         IF (NAR.L.T. 100) THEN
0116             NIA=NAR
0117             FNR=NAR+1
0118         ENDIF
0119         IF (NAR.L.T. 100) THEN
0120             NFA=NAR
0121             FNR=NFA+5
0122         ENDIF
0123         IF (NSA.L.T. 100) THEN
0124             NTN=NSA
0125             SNR=NTN+1
0126         ENDIF
0127         IF (TSA.L.T. 100) THEN
0128             TN=TSA
0129             STR=TN+1
0130         ENDIF
0131         IF (SC.L.T. 100) THEN
0132             Z=SC
0133             SRR=Z+3
0134         ENDIF
0135         IF (ST.L.T. 100) THEN
0136             Z=ST
0137             SRR=Z+2
0138         ENDIF
0139         IF (SR.L.T. 100) THEN
0140             Z=SR
0141             SRR=Z+4
0142         ENDIF
0143         IF (RFS.GE.L.OR.FS.GE.L.OR.NQ.GE.L.OR.RN.GE.L) THEN
0144     C
0145     C:
0146     C: THIS SECTION CONTAINS THE FORMAT STATEMENTS
0147     C: NEEDED TO SET UP TABLES FOR THE PLAN STATISTICS
0148     C: AT THE DIFFERENT INSPECTION LEVELS.
0149     C: THE USER IS ASKED IF SCHEME EVALUATION IS
0150     C: DESIRED, IF SO THEN SUBROUTINE OC IS INVOKED.
0151     C:
0152     C
0153         WRITE(1, 12)
0154         WRITE(5, 12)
0155     12  FORMAT(10X, 'USE 100 PERCENT INSPECTION', /, 'AS SAMPLE',
0156     $    ' SIZE EQUALS OR EXCEEDS LOT SIZE')
0157         S=1
0158         ENDIF
0159         IF (S.EQ. 1) GO TO 60
0160         IF ((SP.EQ. 'S').OR. (SP.EQ. 'D').AND. CDL.NE. 'A')) THEN
0161             WRITE(1, 14)
0162             WRITE(5, 14)
0163     14  FORMAT(10X, '* NORMAL          TIGHTENED          ',
0164     $    ' REDUCED          *')
0165             WRITE(1, 16)
0166             WRITE(5, 16)
0167     16  FORMAT(10X, 54('-'))
0168             WRITE(1, 10) NFA, TFA, RFA
0169             WRITE(5, 10) NFA, TFA, RFA
0170     10  FORMAT(12X, 'AC 1=', I3, 10X, 'AC 1=', I7, 10X, 'AC 1=', I7)
0171             WRITE(1, 20) FNR, FTR, FRR

```

```

0172      WRITE (5, 20) INCL, FIC, ERR
0173 20  FORMAT (12X, 'RE 1=', I3, 10X, 'RE 1=', I7, 10X, 'RE 1=', I7)
0174      ENDIF
0175      IF (SP.EQ. 'D'.AND. CDL.NE. 'A') THEN
0176      WRITE (1, 25) FS, FS, RFS
0177      WRITE (5, 25) FS, FS, RFS
0178 25  FORMAT (12X, 'N1=', I5, 10X, 'N1=', I9, 10X, 'N1=', I9)
0179      WRITE (1, 28)
0180      WRITE (5, 28)
0181 28  FORMAT (10X, 54(' '))
0182      ENDIF
0183      IF (SP.EQ. 'D'.AND. CDL.NE. 'A') THEN
0184      WRITE (1, 30) NTN, TN, Z
0185      WRITE (5, 30) NTN, TN, Z
0186 30  FORMAT (12X, 'AC 2=', I3, 10X, 'AC 2=', I7, 10X, 'AC 2=', I7)
0187      WRITE (1, 40) SNR, STR, SRR
0188      WRITE (5, 40) SNR, STR, SRR
0189 40  FORMAT (12X, 'RE 2=', I3, 10X, 'RE 2=', I7, 10X, 'RE 2=', I7)
0190      WRITE (1, 45) FS, FS, RFS
0191      WRITE (5, 45) FS, FS, RFS
0192 45  FORMAT (12X, 'N2=', I5, 10X, 'N2=', I9, 10X, 'N2=', I9)
0193      ENDIF
0194      WRITE (1, 48)
0195      WRITE (5, 48)
0196 48  FORMAT (10X, 54(' '))
0197      IF (SP.EQ. 'S') THEN
0198      WRITE (1, 50) NQ, NQ, RN
0199      WRITE (5, 50) NQ, NQ, RN
0200 50  FORMAT (12X, 'N=', I6, 10X, 'N=', I10, 10X, 'N=', I10)
0201      WRITE (1, 51)
0202      WRITE (5, 51)
0203 51  FORMAT (10X, 54(' '))
0204      ENDIF
0205      IF ((SP.EQ. 'D'.AND. CDL.EQ. 'A'.AND. J.GE. 19).OR. (SP.EQ. 'D'.AND. CDL.
0206      $ EQ. 'A'.AND. M.LE. 16)) THEN
0207      WRITE (1, 54)
0208      WRITE (5, 54)
0209 54  FORMAT (10X, 'SAMPLE SIZE NOT AVAILABLE AT ANY INSPECTION',
0210      $      ' LEVEL', /, 'CORRESPONDING SINGLE SAMPLING PLAN WILL',
0211      $      ' BE USED: ')
0212      SP='S'
0213      GO TO 1
0214      ENDIF
0215      PRINT *, 'DO YOU WANT SCHEME EVALUATION..?'
0216      PRINT *, 'IF YES ENTER.....1'
0217      PRINT *, 'IF NO ENTER.....2'
0218      READ *, AZ
0219      IF (AZ.EQ. 2) GO TO 60
0220      IF (AZ.EQ. 1) GO TO 58
0221 58  NNNN=L
0222      IF (SP.EQ. 'S') JT=1
0223      IF (SP.EQ. 'D') JT=2
0224      CALL OC (NNNN, FS, RFS, RFA, SRR, FRR, NQ, RN, TFA, TN, FTR, STR, NFA, NTN
0225      $      , FNR, SNR, Z, JT, AGL)
0226 60  RETURN
0227      END

```



```

0001          SUBROUTINE GS
0002          C
0003          C*****
0004          C* THIS SUBROUTINE FINDS THE SAMPLE SIZE, NQ
0005          C* BOTH THE DOUBLE AND THE SINGLE SAMPLING PLANS
0006          C* AT NORMAL, TIGHTENED AND REDUCED INSPECTION
0007          C* LEVELS.
0008          C*****
0009          C
0010          CHARACTER I*2, CDL, SP
0011          INTEGER FS, FA, SA, AR, SR, AC, AT, AE, SC, ST, Z, NN, NFS, NFA, NAE, NSA
0012          INTEGER NAC, NAT, TN, TFS, TFA, TAE, TSA, TAR, TAC, TAT, RN, RFA, RAE
0013          INTEGER RAR, RAC, RAT, NAR, FRR, RFS, FTR, FNR, SNR, STR, SRR
0014          COMMON/SCL/L, SP, S, I, NQ, T, R, CDL, FS, D, J, K, AQL, M, FA, SA, AR
0015          $ , SR, AC, AT, AE, SC, ST, Z, NTN, NFS, NFA, NAE, NSA, NAR,
0016          $ NAC, NAT, TN, TFS, TFA, TAE, TSA, TAR, TAC, TAT, RN, RFS, RFA,
0017          $ , RAE, RAR, RAC, RAT, FRR, FTR, FNR, SNR, STR, SRR
0018          IF (SP.EQ.'S') THEN
0019             IF (CDL.EQ.'A') THEN
0020                NQ=2
0021             ENDIF
0022             IF (CDL.EQ.'B') THEN
0023                NQ=3
0024             ENDIF
0025             IF (CDL.EQ.'C') THEN
0026                NQ=5
0027             ENDIF
0028             IF (CDL.EQ.'D') THEN
0029                NQ=8
0030             ENDIF
0031             IF (CDL.EQ.'E') THEN
0032                NQ=13
0033             ENDIF
0034             IF (CDL.EQ.'F') THEN
0035                NQ=20
0036             ENDIF
0037             IF (CDL.EQ.'G') THEN
0038                NQ=32
0039             ENDIF
0040             IF (CDL.EQ.'H') THEN
0041                NQ=50
0042             ENDIF
0043             IF (CDL.EQ.'J') THEN
0044                NQ=80
0045             ENDIF
0046             IF (CDL.EQ.'K') THEN
0047                NQ=125
0048             ENDIF
0049             IF (CDL.EQ.'L') THEN
0050                NQ=200
0051             ENDIF
0052             IF (CDL.EQ.'M') THEN
0053                NQ=315
0054             ENDIF
0055             IF (CDL.EQ.'N') THEN
0056                NQ=500
0057             ENDIF

```

```

0058      IF (CDL.EQ. 'P') THEN
0059          NG=800
0060      ENDIF
0061      IF (CDL.EQ. 'Q') THEN
0062          NG=1250
0063      ENDIF
0064      IF (CDL.EQ. 'R') THEN
0065          NG=2000
0066      ENDIF
0067      IF (CDL.EQ. 'S') THEN
0068          NG=3150
0069      ENDIF
0070      IF (CDL.EQ. 'A'.OR.CDL.EQ. 'B'.OR.CDL.EQ. 'C') THEN
0071          RN=2
0072      ENDIF
0073      IF (CDL.EQ. 'D') THEN
0074          RN=3
0075      ENDIF
0076      IF (CDL.EQ. 'E') THEN
0077          RN=5
0078      ENDIF
0079      IF (CDL.EQ. 'F') THEN
0080          RN=8
0081      ENDIF
0082      IF (CDL.EQ. 'G') THEN
0083          RN=13
0084      ENDIF
0085      IF (CDL.EQ. 'H') THEN
0086          RN=20
0087      ENDIF
0088      IF (CDL.EQ. 'J') THEN
0089          RN=32
0090      ENDIF
0091      IF (CDL.EQ. 'K') THEN
0092          RN=50
0093      ENDIF
0094      IF (CDL.EQ. 'L') THEN
0095          RN=80
0096      ENDIF
0097      IF (CDL.EQ. 'M') THEN
0098          RN=125
0099      ENDIF
0100      IF (CDL.EQ. 'N') THEN
0101          RN=200
0102      ENDIF
0103      IF (CDL.EQ. 'P') THEN
0104          RN=315
0105      ENDIF
0106      IF (CDL.EQ. 'Q') THEN
0107          RN=500
0108      ENDIF
0109      IF (CDL.EQ. 'R') THEN
0110          RN=800
0111      ENDIF
0112  ENDIF
0113      IF (SP.EQ. 'D'.AND.CDL.EQ. 'A') THEN
0114          FS=0

```

```

0115      ENDIF
0116      IF (SP.EQ. 'D').AND. CDL.NE. 'A') THEN
0117      IF (CDL.EQ. 'B') THEN
0118      FS=2
0119      ENDIF
0120      IF (CDL.EQ. 'C') THEN
0121      FS=3
0122      ENDIF
0123      IF (CDL.EQ. 'D') THEN
0124      FS=5
0125      ENDIF
0126      IF (CDL.EQ. 'E') THEN
0127      FS=8
0128      ENDIF
0129      IF (CDL.EQ. 'F') THEN
0130      FS=13
0131      ENDIF
0132      IF (CDL.EQ. 'G') THEN
0133      FS=20
0134      ENDIF
0135      IF (CDL.EQ. 'H') THEN
0136      FS=32
0137      ENDIF
0138      IF (CDL.EQ. 'J') THEN
0139      FS=50
0140      ENDIF
0141      IF (CDL.EQ. 'K') THEN
0142      FS=80
0143      ENDIF
0144      IF (CDL.EQ. 'L') THEN
0145      FS=125
0146      ENDIF
0147      IF (CDL.EQ. 'M') THEN
0148      FS=200
0149      ENDIF
0150      IF (CDL.EQ. 'N') THEN
0151      FS=315
0152      ENDIF
0153      IF (CDL.EQ. 'P') THEN
0154      FS=500
0155      ENDIF
0156      IF (CDL.EQ. 'Q') THEN
0157      FS=800
0158      ENDIF
0159      IF (CDL.EQ. 'R') THEN
0160      FS=1250
0161      ENDIF
0162      IF (CDL.EQ. 'S') THEN
0163      FS=2000
0164      ENDIF
0165      ENDIF
0166      IF ((SP.EQ. 'D').AND. (CDL.EQ. 'B'.OR. CDL.EQ. 'C')) THEN
0167      RFS=0
0168      ENDIF
0169      IF (SP.EQ. 'D').AND. CDL.NE. 'B'.AND. CDL.NE. 'C') THEN
0170      IF (CDL.EQ. 'D') THEN
0171      RFS=2

```

```

0172
0173
0174
0175
0176
0177
0178
0179
0180
0181
0182
0183
0184
0185
0186
0187
0188
0189
0190
0191
0192
0193
0194
0195
0196
0197
0198
0199
0200
0201
0202
0203
0204
0205
0206
0207
0208
0209
0210
0211

ENDIF
RETURN
END

ENDIF
IF (CDL.EQ. 'E') THEN
RFS=3
ENDIF
IF (CDL.EQ. 'F') THEN
RFS=5
ENDIF
IF (CDL.EQ. 'G') THEN
RFS=8
ENDIF
IF (CDL.EQ. 'H') THEN
RFS=13
ENDIF
IF (CDL.EQ. 'J') THEN
RFS=20
ENDIF
IF (CDL.EQ. 'K') THEN
RFS=32
ENDIF
IF (CDL.EQ. 'L') THEN
RFS=50
ENDIF
IF (CDL.EQ. 'M') THEN
RFS=80
ENDIF
IF (CDL.EQ. 'N') THEN
RFS=125
ENDIF
IF (CDL.EQ. 'P') THEN
RFS=200
ENDIF
IF (CDL.EQ. 'Q') THEN
RFS=315
ENDIF
IF (CDL.EQ. 'R') THEN
RFS=500
ENDIF

```

```

0001          SUBROUTINE INDEX
0002          C
0003          C*****
0004          C** THIS SUBROUTINE LOCATES THE CELL IN THE **
0005          C** TABLES PRESENTED IN MIL-STD-105D CONTAINING **
0006          C** ACCEPTANCE AND REJECTION NUMBERS FOR ANY **
0007          C** PARTICULAR COMBINATION OF AQL AND CODE LETTER. **
0008          C*****
0009          C
0010          CHARACTER I*2, CDL, SP
0011          INTEGER FS, FA, SA, AR, SR, AC, AT, AE, SC, ST, Z
0012          COMMON/SCL/L, SP, S, I, NG, T, R, CDL, FS, D, J, K, AQL, M, FA, SA, AR,
0013          * SR, AC, AT, AE, SC, ST, Z
0014          IF (CDL.EQ. 'A') THEN
0015             K=1
0016             ELSE IF (CDL.EQ. 'B') THEN
0017                K=2
0018             ELSE IF (CDL.EQ. 'C') THEN
0019                K=3
0020             ELSE IF (CDL.EQ. 'D') THEN
0021                K=4
0022             ELSE IF (CDL.EQ. 'E') THEN
0023                K=5
0024             ELSE IF (CDL.EQ. 'F') THEN
0025                K=6
0026             ELSE IF (CDL.EQ. 'G') THEN
0027                K=7
0028             ELSE IF (CDL.EQ. 'H') THEN
0029                K=8
0030             ELSE IF (CDL.EQ. 'J') THEN
0031                K=9
0032             ELSE IF (CDL.EQ. 'K') THEN
0033                K=10
0034             ELSE IF (CDL.EQ. 'L') THEN
0035                K=11
0036             ELSE IF (CDL.EQ. 'M') THEN
0037                K=12
0038             ELSE IF (CDL.EQ. 'N') THEN
0039                K=13
0040             ELSE IF (CDL.EQ. 'P') THEN
0041                K=14
0042             ELSE IF (CDL.EQ. 'Q') THEN
0043                K=15
0044             ELSE IF (CDL.EQ. 'R') THEN
0045                K=16
0046             ELSE IF (CDL.EQ. 'S') THEN
0047                K=17
0048             ENDIF
0049          IF (AQL.EQ. 0.010) THEN
0050             J=1
0051             ELSE IF (AQL.EQ. 0.015) THEN
0052                J=2
0053             ELSE IF (AQL.EQ. .025) THEN
0054                J=3
0055             ELSE IF (AQL.EQ. 0.040) THEN
0056                J=4
0057             ELSE IF (AQL.EQ. 0.065) THEN

```

```

0058      J=5
0059      ELSE IF (AQL.EQ.0.10) THEN
0060      J=6
0061      ELSE IF (AQL.EQ.0.15) THEN
0062      J=7
0063      ELSE IF (AQL.EQ.0.25) THEN
0064      J=8
0065      ELSE IF (AQL.EQ.0.4) THEN
0066      J=9
0067      ELSE IF (AQL.EQ.0.65) THEN
0068      J=10
0069      ELSE IF (AQL.EQ.1) THEN
0070      J=11
0071      ELSE IF (AQL.EQ.1.5) THEN
0072      J=12
0073      ELSE IF (AQL.EQ.2.5) THEN
0074      J=13
0075      ELSE IF (AQL.EQ.4) THEN
0076      J=14
0077      ELSE IF (AQL.EQ.6.5) THEN
0078      J=15
0079      ELSE IF (AQL.EQ.10) THEN
0080      J=16
0081      ELSE IF (AQL.EQ.15) THEN
0082      J=17
0083      ELSE IF (AQL.EQ.25) THEN
0084      J=18
0085      ELSE IF (AQL.EQ.40) THEN
0086      J=19
0087      ELSE IF (AQL.EQ.65) THEN
0088      J=20
0089      ELSE IF (AQL.EQ.100) THEN
0090      J=21
0091      ELSE
0092      J=22
0093      ENDIF
0094      M=J+K
0095      RETURN
0096      END

```

0001  
0002  
0003  
0004  
0005  
0006  
0007  
0008  
0009  
0010  
0011  
0012  
0013  
0014  
0015  
0016  
0017  
0018  
0019  
0020  
0021  
0022  
0023  
0024  
0025  
0026  
0027  
0028  
0029  
0030  
0031  
0032  
0033  
0034  
0035  
0036  
0037  
0038  
0039  
0040  
0041  
0042  
0043  
0044  
0045  
0046  
0047  
0048  
0049  
0050  
0051  
0052  
0053  
0054  
0055  
0056  
0057

SUBROUTINE VALUES

```
C
C*****
C** THIS SUBROUTINE FINDS ACCEPTANCE AND REJECTION **
C** NUMBERS FOR THE CELL IDENTIFIED BY SUBROUTINE INDEX. **
C*****
C
CHARACTER I*2, CDL, SP
INTEGER FS, FA, SA, AR, SR, AC, AT, AE, SC, ST, Z, NN, NFS, NFA, NAE, NSA
INTEGER NAR, NAC, NAT, TN, TFS, TFA, TAE, TAR, TAC, TAT, RN, RFA, RAE
INTEGER FRR, FTR, FNR, SNR, STR, SRR, RAR, RAC, RAT, RFS, TSA
COMMON/SCL/L, SP, S, I, NQ, T, R, CDL, FS, D, J, K, AGL, M, FA, SA,
$ AR, SR, AC, AT, AE, SC, ST, Z, NTN, NFS, NFA, NAE, NSA, NAR, NAC, NAT, TN
$ , TFS, TFA, TAE, TSA, TAR, TAC, TAT, RN, RFS, RFA, RAE, RAR, RAC, RAT
$ , FRR, FTR, FNR, SNR, STR, SRR
NFA=100
TFA=100
RFA=100
NAT=100
TAT=100
RAT=100
TAC=100
RAC=100
TAR=100
RAR=100
TAE=100
RAE=100
TSA=100
NAC=100
NAR=100
NAE=100
NSA=100
ST=100
SC=100
SR=100
IF(SP.EQ. 'S') THEN
IF(M. GE. 2. AND. M. LE. 16) NFA=0
IF(M. GE. 26) NFA=21
IF(M. EQ. 25) NFA=14
IF(M. EQ. 24) NFA=10
IF(M. EQ. 23) NFA=7
IF(M. EQ. 22) NFA=5
IF(M. EQ. 21) NFA=3
IF(M. EQ. 20) NFA=2
IF(M. EQ. 19) NFA=1
IF(M. EQ. 17. AND. J. NE. 16) NFA=0
IF(M. EQ. 18. AND. J. NE. 18) NFA=1
IF(K. EQ. 16. AND. J. LE. 2) NFA=0
IF(J. EQ. 16. AND. K. LE. 2) NFA=1
IF(M. GE. 26) TFA=18
IF(M. EQ. 25) TFA=12
IF(M. EQ. 24) TFA=8
IF(M. EQ. 23) TFA=5
IF(M. EQ. 22) TFA=3
IF(M. EQ. 21) TFA=2
IF(M. GE. 18. AND. M. LE. 20. AND. J. NE. 16) TFA=1
IF(K. EQ. 16. AND. J. EQ. 2) TFA=2
```

```

0058      IF (M. LE. 17. AND. J. LE. 15) RFA=0
0059      IF (J. EQ. 16. AND. K. LE. 2) RFA=1
0060      IF (M. GE. 26) RAC=10
0061      IF (M. EQ. 25) RAC=7
0062      IF (M. EQ. 24) RAC=5
0063      IF (M. EQ. 23) RAC=3
0064      IF (M. EQ. 22. AND. J. LE. 19) RAC=2
0065      IF (M. EQ. 21) RAC=1
0066      IF (M. EQ. 20. AND. J. LE. 18) RAT=1
0067      IF (M. EQ. 19. AND. J. LE. 17) RAT=0
0068      IF (M. GE. 2. AND. M. LE. 16) RFA=0
0069      IF (M. EQ. 17. AND. J. LE. 15) RFA=0
0070      IF (J. LE. 2) RFA=0
0071      IF (J. EQ. 16. AND. K. LE. 3) RAT=0
0072      IF (J. EQ. 17. AND. K. EQ. 1) RAT=0
0073      IF (J. EQ. 21. AND. K. LE. 2) RFA=5
0074      IF (J. EQ. 20. AND. K. EQ. 1) RFA=3
0075      IF (J. EQ. 20. AND. K. EQ. 2) RAT=3
0076      IF (J. EQ. 19. AND. K. EQ. 1) RFA=2
0077      IF (J. EQ. 19. AND. K. EQ. 2) RAT=2
0078      IF (J. EQ. 18. AND. K. EQ. 1) RFA=1
0079      IF (J. EQ. 18. AND. K. EQ. 2) RAT=1
0080
0081      ENDIF
0082      IF (SP. EQ. 'D') THEN
0083          IF (M. LE. 16. AND. J. LE. 15) THEN
0084              NAT=0
0085              NSA=1
0086              TAT=0
0087              TSA=1
0088              RAT=0
0089              ST=0
0090              END IF
0091          IF (K. EQ. 16. AND. J. LE. 2) THEN
0092              NAT=0
0093              NSA=1
0094              TAT=0
0095              TSA=1
0096              RAT=0
0097              ST=0
0098              ENDIF
0099          IF (K. EQ. 1. AND. J. EQ. 18) THEN
0100              NSA=3
0101              NAC=0
0102              TAT=0
0103              TSA=1
0104              ENDIF
0105          IF (M. GE. 26) THEN
0106              NAE=11
0107              NSA=26
0108              ENDIF
0109          IF (M. EQ. 25) THEN
0110              NAR=7
0111              NSA=18
0112              ENDIF
0113          IF (M. EQ. 24) THEN
0114              NAR=5
0115              NSA=12

```



## VALUES

```
0115      ENDIF
0116      IF (M. EQ. 23) THEN
0117      NAR=3
0118      NSA=8
0119      ENDIF
0120      IF (M. EQ. 22) THEN
0121      NAC=2
0122      NSA=6
0123      ENDIF
0124      IF (M. EQ. 21) THEN
0125      NAC=1
0126      NSA=4
0127      ENDIF
0128      IF (M. EQ. 20) THEN
0129      NAC=0
0130      NSA=3
0131      ENDIF
0132      IF ((M. EQ. 19. AND. J. GT. 2). OR. (M. EQ. 18. AND. J. GT. 2)) THEN
0133      NAT=0
0134      NSA=1
0135      ENDIF
0136      IF (J. EQ. 16. AND. K. LE. 3) THEN
0137      NAT=0
0138      NSA=1
0139      ENDIF
0140      IF (M. GE. 26) THEN
0141      TAE=9
0142      TSA=23
0143      ENDIF
0144      IF (M. EQ. 25) THEN
0145      TAR=6
0146      TSA=15
0147      ENDIF
0148      IF (M. EQ. 24) THEN
0149      TAR=3
0150      TSA=11
0151      ENDIF
0152      IF (M. EQ. 23) THEN
0153      TAC=2
0154      TSA=6
0155      ENDIF
0156      IF (M. EQ. 22) THEN
0157      TAC=1
0158      TSA=4
0159      ENDIF
0160      IF (M. EQ. 21) THEN
0161      TAC=0
0162      TSA=3
0163      ENDIF
0164      IF (M. GE. 18. AND. M. LE. 20. AND. J. GE. 3) THEN
0165      TAT=0
0166      TSA=1
0167      ENDIF
0168      IF (M. EQ. 26) THEN
0169      RAE=5
0170      SR=12
0171      ENDIF
```

```

0172          IF (M. EQ. 25) THEN
0173          RAR=3
0174          SR=8
0175          ENDIF
0176          IF (M. EQ. 24) THEN
0177          RAR=2
0178          SC=6
0179          ENDIF
0180          IF (M. EQ. 23) THEN
0181          RAR=1
0182          SC=4
0183          ENDIF
0184          IF (M. EQ. 22) THEN
0185          RAR=0
0186          SC=3
0187          ENDIF
0188          IF (M. EQ. 21) THEN
0189          RAR=0
0190          SR=1
0191          ENDIF
0192          IF (M. EQ. 20) THEN
0193          RAC=0
0194          SR=0
0195          ENDIF
0196          IF (M. EQ. 19) THEN
0197          RAT=0
0198          ST=0
0199          ENDIF
0200          IF (M. EQ. 18. AND. J. GE. 3. AND. J. LE. 15) THEN
0201          RAT=0
0202          ST=0
0203          ENDIF
0204          ENDIF
0205          RETURN
0206          END

```

```

0001          SUBROUTINE CODE
0002          C
0003          C*****
0004          C** THIS SUBROUTINE IDENTIFIES THE CODE LETTER ASSOCIATED **
0005          C** WITH EITHER THE SPECIAL OR GENERAL INSPECTION LEVELS **
0006          C** FOR ANY GIVEN LOT OR BATCH SIZE. **
0007          C*****
0008          C
0009          CHARACTER I*2,CDL,SP
0010          INTEGER S1,S2,S3,S4
0011          COMMON/SCL/L,SP,S,I,NG,T,R,CDL
0012          IF(I.EQ.'S1')THEN
0013              IF(L.GE.2.AND.L.LE.50)CDL='A'
0014              IF(L.GE.51.AND.L.LE.500)CDL='B'
0015              IF(L.GE.501.AND.L.LE.35000)CDL='C'
0016              IF(L.GE.35001)CDL='D'
0017          ENDIF
0018          IF(I.EQ.'S2')THEN
0019              IF(L.GE.2.AND.L.LE.25)CDL='A'
0020              IF(L.GE.26.AND.L.LE.150)CDL='B'
0021              IF(L.GE.151.AND.L.LE.1200)CDL='C'
0022              IF(L.GE.1201.AND.L.LE.35000)CDL='D'
0023              IF(L.GE.35001)CDL='E'
0024          ENDIF
0025          IF(I.EQ.'S3')THEN
0026              IF(L.GE.2.AND.L.LE.15)CDL='A'
0027              IF(L.GE.16.AND.L.LE.25)CDL='B'
0028              IF(L.GE.26.AND.L.LE.90)CDL='C'
0029              IF(L.GE.91.AND.L.LE.150)CDL='D'
0030              IF(L.GE.151.AND.L.LE.500)CDL='E'
0031              IF(L.GE.501.AND.L.LE.1200)CDL='F'
0032              IF(L.GE.1201.AND.L.LE.10000)CDL='G'
0033              IF(L.GE.10001.AND.L.LE.35000)CDL='H'
0034              IF(L.GE.35001.AND.L.LE.500000)CDL='J'
0035              IF(L.GT.500000)CDL='K'
0036          ENDIF
0037          IF(I.EQ.'S4')THEN
0038              IF(L.GE.2.AND.L.LE.15)CDL='A'
0039              IF(L.GE.16.AND.L.LE.25)CDL='B'
0040              IF(L.GE.26.AND.L.LE.90)CDL='C'
0041              IF(L.GE.91.AND.L.LE.150)CDL='D'
0042              IF(L.GE.151.AND.L.LE.500)CDL='E'
0043              IF(L.GE.501.AND.L.LE.1200)CDL='F'
0044              IF(L.GE.1201.AND.L.LE.10000)CDL='G'
0045              IF(L.GE.10001.AND.L.LE.35000)CDL='H'
0046              IF(L.GE.35001.AND.L.LE.500000)CDL='J'
0047              IF(L.GT.500000)CDL='K'
0048          ENDIF
0049          IF(I.EQ.'1')THEN
0050              IF(L.GE.2.AND.L.LE.15)CDL='A'
0051              IF(L.GE.16.AND.L.LE.25)CDL='B'
0052              IF(L.GE.26.AND.L.LE.90)CDL='C'
0053              IF(L.GE.91.AND.L.LE.150)CDL='D'
0054              IF(L.GE.151.AND.L.LE.280)CDL='E'
0055              IF(L.GE.281.AND.L.LE.500)CDL='F'
0056              IF(L.GE.501.AND.L.LE.1200)CDL='G'
0057              IF(L.GE.1201.AND.L.LE.3200)CDL='H'
0058              IF(L.GE.3201.AND.L.LE.10000)CDL='J'
0059              IF(L.GE.10001.AND.L.LE.35000)CDL='K'

```

```

0058      IF (L. GE. 35001. AND. L. LE. 150000) CDL= 'L'
0059      IF (L. GE. 150001. AND. L. LE. 500000) CDL= 'M'
0060      IF (L. GE. 500001) CDL= 'N'
0061      ENDIF
0062      IF (I. EQ. '2') THEN
0063          IF (L. GE. 2. AND. L. LE. 8) CDL= 'A'
0064          IF (L. GE. 9. AND. L. LE. 15) CDL= 'B'
0065          IF (L. GE. 16. AND. L. LE. 25) CDL= 'C'
0066          IF (L. GE. 26. AND. L. LE. 50) CDL= 'D'
0067          IF (L. GE. 51. AND. L. LE. 90) CDL= 'E'
0068          IF (L. GE. 91. AND. L. LE. 150) CDL= 'F'
0069          IF (L. GE. 151. AND. L. LE. 280) CDL= 'G'
0070          IF (L. GE. 281. AND. L. LE. 500) CDL= 'H'
0071          IF (L. GE. 501. AND. L. LE. 1200) CDL= 'J'
0072          IF (L. GE. 1201. AND. L. LE. 3200) CDL= 'K'
0073          IF (L. GE. 3201. AND. L. LE. 10000) CDL= 'L'
0074          IF (L. GE. 10001. AND. L. LE. 35000) CDL= 'M'
0075          IF (L. GE. 35001. AND. L. LE. 150000) CDL= 'N'
0076          IF (L. GE. 150001. AND. L. LE. 500000) CDL= 'P'
0077          IF (L. GE. 500001) CDL= 'Q'
0078      ENDIF
0079      IF (I. EQ. '3') THEN
0080          IF (L. GE. 2. AND. L. LE. 8) CDL= 'B'
0081          IF (L. GE. 9. AND. L. LE. 15) CDL= 'C'
0082          IF (L. GE. 16. AND. L. LE. 25) CDL= 'D'
0083          IF (L. GE. 26. AND. L. LE. 50) CDL= 'E'
0084          IF (L. GE. 51. AND. L. LE. 90) CDL= 'F'
0085          IF (L. GE. 91. AND. L. LE. 150) CDL= 'G'
0086          IF (L. GE. 151. AND. L. LE. 280) CDL= 'H'
0087          IF (L. GE. 281. AND. L. LE. 500) CDL= 'J'
0088          IF (L. GE. 501. AND. L. LE. 1200) CDL= 'K'
0089          IF (L. GE. 1201. AND. L. LE. 3200) CDL= 'L'
0090          IF (L. GE. 3201. AND. L. LE. 10000) CDL= 'M'
0091          IF (L. GE. 10001. AND. L. LE. 35000) CDL= 'N'
0092          IF (L. GE. 35001. AND. L. LE. 150000) CDL= 'P'
0093          IF (L. GE. 150001. AND. L. LE. 500000) CDL= 'Q'
0094          IF (L. GE. 500001) CDL= 'R'
0095      ENDIF
0096      RETURN
0097      END

```

```

0001          SUBROUTINE OC(NNNN, IFS, IRFS, IRFA, ISRR, IFRR, NG, IRN, ITFA
0002          + ITN, IFTR, ISTR, NFA, NTN, IFNR, ISNR, IZ, J, AQL)
0003      C
0004      C*****
0005      C** THIS SUBROUTINE EVALUATES THE SCHEME OPERATING **
0006      C** CHARACTERISTICS, FOR THE REQUIRED SAMPLING PLANS. **
0007      C*****
0008      C
0009          REAL*8 AQ(20, 20), BQ(20, 20)
0010          DIMENSION N(30), PR(30), V(100, 50), P(30), TPA(30, 4), ASN(30, 4)
0011          DIMENSION AQC(30, 4), ATI(30, 4), NN(4), TM(20, 20), ZP(20, 20)
0012          DIMENSION ASNC(30), AQC(30), ATIC(30), AFI(30), PA(30)
0013          INTEGER A(50), R(50), HI, AA(2, 4), RR(2, 4)
0014          IF(J.EQ.2) GO TO 197
0015          IF(J.EQ.1) GO TO 172
0016      172      NN(1)=NG
0017              NN(2)=NG
0018              NN(3)=IRN
0019              NN(4)=IRN
0020              AA(1, 4)=IFRR-1
0021              RR(1, 3)=IRFA+1
0022              GO TO 272
0023      197      NN(1)=IFS
0024              NN(2)=IFS
0025              NN(3)=IRFS
0026              NN(4)=IRFS
0027              AA(1, 4)=IRFA
0028              AA(2, 4)=ISRR-1
0029              RR(1, 3)=IFRR
0030      272      AA(1, 1)=ITFA
0031              AA(2, 1)=ITN
0032              RR(1, 1)=IFTR
0033              RR(2, 1)=ISTR
0034              AA(1, 2)=NFA
0035              AA(2, 2)=NTN
0036              RR(1, 2)=IFNR
0037              RR(2, 2)=ISNR
0038              AA(1, 3)=IRFA
0039              AA(2, 3)=IZ
0040              RR(2, 3)=IZ+1
0041              RR(1, 4)=IFRR
0042              RR(2, 4)=ISRR
0043              PRINT *, 'DO YOU WANT A TABLE OR A GRAPH FORMAT ?'
0044              PRINT *, 'FOR GRAPH....ENTER: 1'
0045              PRINT *, 'FOR TABLE....ENTER: 2'
0046              READ *, XTC
0047              IF(XTC.EQ.1) GO TO 733
0048              PRINT *, 'SPECIFY THE NUMBER OF FRACTION DEFECTIVE VALUES..'
0049              READ *, J1
0050              PRINT *, 'ENTER THE FRACTION DEFECTIVE VALUE(S), '
0051              PRINT *, '(PUT A COMMA BETWEEN VALUES.).....'
0052              READ *, (P(I), I=1, J1)
0053              WRITE(1, 27)
0054              WRITE(5, 27)
0055              GO TO 447
0056      733      J1=21
0057              DO 55 I=1, J1

```

00

```

0058          P(I) = (I-1)/20.0
0059          01  P(I) = P(I)/10.0
0060          44  DO 105 I = 1, J1
0061          DO 10 I = 1, 4
0062          DO 17 J = 1, J
0063          A(I) = AA(I, I, M)
0064          R(I) = RR(I, I, M)
0065          N(I) = NN(I, M)
0066          17  CONTINUE
0067          NNN = NNNN
0068          M = INT(NNN * P(I))
0069          ZNK = NNN * P(L)
0070          ZN = M
0071          IF (ZNK - ZN .LE. 0.5) GO TO 148
0072          M = M + 1
0073          148  CONTINUE
0074          SUM = 0
0075          I1 = R(I)
0076          C
0077          C
0078          C THE NEXT STATEMENTS CALCULATE THE PROBABILITY THAT
0079          C THERE ARE (I-1) DEFECTIVES IN THE FIRST SAMPLE.
0080          C
0081          C
0082          DO 20 J = 1, I1
0083          V(I, 1) = PP(N(I), P(L), I-1)
0084          20  SUM = SUM + V(I, 1)
0085          V(R(I)+1, 1) = 1. - SUM
0086          IF (A(I) .LE. 0) GO TO 21
0087          I1 = A(I)
0088          C
0089          C
0090          C THE FOLLOWING STATEMENTS CALCULATE THE PROB. THAT THERE
0091          C ARE LESS THAN OR EQUAL TO A(I) DEFECTIVE ITEMS.
0092          C
0093          C
0094          DO 30 I = 1, I1
0095          C
0096          C
0097          C THE NEXT SECTION CALCULATES THE INTERMEDIATE PROBABILITIES
0098          C OF CONTINUED SAMPLING FOR THE DOUBLE SAMPLING PLANS.
0099          C
0100          C
0101          C
0102          30  V(A(I)+1, 1) = V(A(I)+1, 1) + V(I, 1)
0103          21  IF (J .EQ. 1) GO TO 41
0104          LOW = 1
0105          I = 2
0106          NNN = NNN - N(I-1)
0107          NR = R(I)
0108          LOW = MAX0(LOW, A(I-1)+2)
0109          HI = R(I-1)
0110          DO 60 I1 = 1, NR
0111          V(I1, I) = 0
0112          IF (I1 .LT. LOW) GO TO 60
0113          IF (A(I) .EQ. (-1)) GO TO 61
0114          IF (A(I)+1 .NE. I1) GO TO 61

```

```

0115          NMM=MINO(I1, I1)
0116          DO 70 I2=LOW , MMM
0117             I4=I1-I2+1
0118             DO 50 I1=1, NR
0119                PR(I1)=PP(N(I), P(L), I1, NNN, M, I2+1)
0120          50    CONTINUE
0121             DO 80 I3=1, I4
0122             80  V(I1, I)=V(I1, I)+V(I2, I-1)*PR(I3)
0123             70  CONTINUE
0124             GO TO 60
0125          61    MMM=MINO(HI, I1)
0126             DO 90 I2=LOW , MMM
0127                I3=I1-I2+1
0128             DO 743 IJ=1, NR
0129          743    PR(IJ)=PP(N(I), P(L), IJ-1)
0130             90  V(I1, I)=V(I1, I)+V(I2, I-1)*PR(I3)
0131             60  CONTINUE
0132             V(R(I)+1, I)=0
0133             DO 110 I1=LOW , HI
0134             SUM=0
0135             I3=R(I)-I1+1
0136             DO 120 I2=1, I3
0137             SUM=SUM+PR(I2)
0138          110    V(R(I)+1, I)=V(R(I)+1, I)+(1.-SUM)*V(I1, I-1)
0139             41  CONTINUE
0140          C
0141          C
0142          C THE NEXT SECTION EVALUATES THE SCHEME FOR THE SINGLE
0143          C SAMPLING PLAN (OR THE FIRST STAGE IN DOUBLE SAMPLING).
0144          C
0145          C
0146          121    ASN(L, LM)=0
0147                TPA(L, LM)=0
0148                SS=0
0149                DO 130 I=1, J
0150                SS=SS+N(I)
0151                IF(A(I).EQ.(-1)) GO TO 131
0152                ASN(L, LM)=ASN(L, LM)+(V(A(I)+1, I)+V(R(I)+1, I))*SS
0153                TPA(L, LM)=TPA(L, LM)+V(A(I)+1, I)
0154                GO TO 130
0155          131    ASN(L, LM)=ASN(L, LM)+V(R(I)+1, I)*SS
0156          130    CONTINUE
0157                XXX=0
0158                SSS=0
0159                VVV=0
0160                DO 133 I=1, J
0161                SSS=SSS+N(I)
0162                XXX=XXX+SSS*V(A(I)+1, I)
0163                VVV=VVV+(NNNN-SSS)*V(A(I)+1, I)
0164          133    CONTINUE
0165                ATI(L, LM)=XXX+(NNNN*(1.0-TPA(L, LM)))
0166                AOG(L, LM)=(VVV*P(L))/NNNN
0167          140    CONTINUE
0168          10    CONTINUE
0169          19    FORMAT(10X, F5.3, 2X, F9.4, F10.2, 4X, F6.4, 1X, F10.2)
0170          27    FORMAT(///, 16X, 'SCHEME OPERATING CHARACTERISTICS ', /, 10X
0171          $      , 'P: ', P(A), ' ASN: ', ASN, ' AOG: ', AFI')

```

```

0172          DO 1 I=1,20
0173          DC 1 J=1,70
0174          TM(I,J)=0
0175          1
0176          C
0177          C THIS SECTION FIRST CALCULATES THE PROB. OF BEING
0178          C IN NORMAL, TIGHTENED OR REDUCED INSPECTION LEVELS AND
0179          C THEN COMPUTES THE PROB. OF PASSING THE LIMIT NUMBER
0180          C CRITERIA.
0181          C
0182          C
0183          T=TPA(L, 1)
0184          S=TPA(L, 2)
0185          RR1=TPA(L, 3)
0186          R1=TPA(L, 4)
0187          NZ=10*ASN(L, 2)
0188          CALL LNM(AQL, NZ, LNC)
0189          Q=B(NZ, P(L), LNC)
0190          C
0191          C
0192          C THIS SECTION READS IN THE PROBABILITY TRANSITION MATRIX.
0193          C
0194          C
0195          TM(1, 1)=1.-T
0196          TM(1, 2)=T
0197          TM(2, 1)=1.-T
0198          TM(2, 3)=T
0199          TM(3, 1)=1.-T
0200          TM(3, 4)=T
0201          TM(4, 1)=1.-T
0202          TM(4, 5)=T
0203          TM(5, 1)=1.-T
0204          TM(5, 6)=T
0205          TM(6, 7)=1.-S
0206          TM(6, 11)=S
0207          TM(7, 1)=1.-S
0208          TM(7, 8)=S
0209          TM(8, 1)=1.-S
0210          TM(8, 9)=S
0211          TM(9, 1)=1.-S
0212          TM(9, 10)=S
0213          TM(10, 1)=1.-S
0214          TM(10, 14)=S
0215          TM(11, 7)=1.-S
0216          TM(11, 12)=S
0217          TM(12, 7)=1.-S
0218          TM(12, 13)=S
0219          TM(13, 7)=1.-S
0220          TM(13, 14)=S
0221          TM(14, 7)=1.-S
0222          TM(14, 15)=S
0223          TM(15, 7)=1.-S
0224          TM(15, 16)=S
0225          TM(16, 7)=1.-S
0226          TM(16, 17)=S
0227          TM(17, 7)=1.-S
0228          TM(17, 18)=S

```



```

0230      TM(18, 7)=1.5
0231      TM(18, 19)=S
0232      TM(19, 7)=1.5
0233      TM(19, 19)=(1.-Q)*S
0234      TM(20, 6)=1.-RR1
0235      TM(20, 20)=RR1
0236      NR=20
0237      NC=20
0238      DO 100 I=1, NR
0239      DO 100 JJ=1, NC
0240      IF (JJ-1) 111, 15, 111
0241      15  ZP(JJ, I)=TM(I, JJ)-1.0
0242      GO TO 100
0243      111  ZP(JJ, I)=TM(I, JJ)
0244      100  CONTINUE
0245      DO 115 I=2, NR
0246      DO 115 JJ=1, NC
0247      KL=I-1
0248      AQ(I, JJ)=ZP(KL, JJ)
0249      115  CONTINUE
0250      DO 18 JJ=1, NC
0251      AQ(1, JJ)=1.0
0252      18  CONTINUE
0253      CALL INVERT(NR, AQ, BQ)
0254      PA(L)=0
0255      ASNC(L)=0
0256      ADQC(L)=0
0257      ATIC(L)=0
0258      DO 161 I=1, NR
0259      IF(I.GT.5)GO TO 200
0260      ZZ=T
0261      Z1=ASN(L, 1)
0262      Z2=AQG(L, 1)
0263      Z3=ATI(L, 1)
0264      GO TO 201
0265      200  IF(I.GT.19) GO TO 210
0266      ZZ=S
0267      Z1=ASN(L, 2)
0268      Z2=AQG(L, 2)
0269      Z3=ATI(L, 2)
0270      GO TO 201
0271      210  ZZ=R1
0272      Z1=ASN(L, 4)
0273      Z2=AQG(L, 4)
0274      Z3=ATI(L, 4)
0275      C
0276      C .....
0277      C THE FOLLOWING STATEMENTS MULTIPLY THE PROBABILITIES
0278      C OF BEING IN THE VARIOUS STATES BY THE CHARACTERISTICS
0279      C OF EACH STATE.
0280      C .....
0281      C
0282      201  PA(L)=PA(L)+BQ(I, 1)*ZZ
0283      ASNC(L)=ASNC(L)+BQ(I, 1)*Z1
0284      ADQC(L)=ADQC(L)+BQ(I, 1)*Z2
0285      ATIC(L)=ATIC(L)+BQ(I, 1)*Z3

```

```

0286      AFI(L)=ATJC(L)/NNNN
0287      151  CONTINUE
0288      IF(XTC.NE.2) GO TO 101
0289      WRITE(1,19) P(L),PA(L),ASNC(L),AQQC(L),AFI(L)
0290      WRITE(5,19) P(L),PA(L),ASNC(L),AQQC(L),AFI(L)
0291      101  CONTINUE
0292      IF(XTC.EQ.2) GO TO 556
0293      WRITE(1,45)
0294      WRITE(5,45)
0295      WRITE(1,46)
0296      WRITE(5,46)
0297      CALL PLOTQC(P,PA)
0298      WRITE(1,47)
0299      WRITE(5,47)
0300      CALL PLOTQC(P,ASNC)
0301      WRITE(1,48)
0302      WRITE(5,48)
0303      CALL PLOTQC(P,AQQC)
0304      WRITE(1,49)
0305      WRITE(5,49)
0306      CALL PLOTQC(P,AFI)
0307      45  FORMAT(1H1,50X,'OPERATING CHARACTERISTIC CURVE')
0308      46  FORMAT(53X,'PROBABILITY OF ACCEPTANCE')
0309      47  FORMAT(1H1,50X,'AVERAGE SAMPLE NUMBER')
0310      48  FORMAT(1H1,50X,'AVERAGE OUTGOING QUALITY')
0311      49  FORMAT(1H1,50X,'AVERAGE FRACTION INSPECTED')
0312      GO TO 898
0313      556 CONTINUE
0314      WRITE(1,555),LNC
0315      WRITE(5,555),LNC
0316      555 FORMAT(10X,'LIMIT NUMBER FOR REDUCED INSPECTION IS:',I9)
0317      898 RETURN
0318      END

```

```
0001          FUNCTION PP(N,P,K)
0002             Q=1. P
0003             PP=Q**N
0004             IF (K.EQ.0)RETURN
0005             DO 20 I=1,K
0006 70          PP=PP*P*(N-I+1)/(Q*I)
0007             RETURN
0008             END
```

```

0001      FUNCTION B(N,P,K)
0002      PN=P*N
0003      Q=1.-P
0004      IF (PN.GT.0.) GO TO 1
0005      B=EXP(-PN)
0006      Z=EXP(-PN)
0007      IF (K.EQ.0) RETURN
0008      DO 10 I=1,K
0009          Z=Z*PN/I
0010      10  B=B+Z
0011      RETURN
0012      1  B=Q**N
0013          Z=Q**N
0014      IF (K.EQ.0) RETURN
0015      DO 20 I=1,K
0016          Z=Z*(N-I+1)*P/(Q*I)
0017      20  B=B+Z
0018      RETURN
0019      END

```

```

0001          SUBROUTINE INVERT(NR, AQ, BQ)
0002      (
0003      (*****
0004      (** THIS SUBROUTINE INVERTS THE MATRIX BY GAUSSIAN **
0005      (** ELIMINATION. **
0006      (*****
0007      (
0008          REAL*8 AQ(20, 20), BQ(20, 20), ZTAMP, ATEMP
0009          DO 15 I=1, NR
0010              DO 10 J=1, NR
0011                  BQ(I, J)=0.0
0012              15 BQ(I, I)=1.0
0013              DO 35 I=1, NR
0014                  ZTAMP=AQ(I, I)
0015                  DO 20 J=1, NR
0016                      AQ(I, J)=AQ(I, J)/ZTAMP
0017                      BQ(I, J)=BQ(I, J)/ZTAMP
0018                  DO 30 II=1, NR
0019                      IF(I.EQ.II) GO TO 30
0020                      ATEMP=AQ(II, I)
0021                      DO 25 J=1, NR
0022                          AQ(II, J)=AQ(II, J)-AQ(I, J)*ATEMP
0023                          BQ(II, J)=BQ(II, J)-BQ(I, J)*ATEMP
0024                  30 CONTINUE
0025                  35 CONTINUE
0026          RETURN
0027      END

```

```

0001          SUBROUTINE PLOTDC(P,T)
0002          C
0003          C*****
0004          C* THIS SUBROUTINE PLOTS DC, ASN, AOP AND AFI CURVES FOR **
0005          C* THE FULL RANGE OF INCOMING FRACTION DEFECTIVE. **
0006          C*****
0007          C
0008          DIMENSION U(11),T(30),P(30),LINE(101)
0009          INTEGER ASTERK,BLANK,PLUS
0010          DATA ASTERK,BLANK,PLUS/'*',' ','+'
0011          DO 1 K=1,101
0012          1 LINE(K)=BLANK
0013             XMAX=T(1)
0014             XMIN=T(1)
0015             DO 2 I=1,21
0016                IF(T(I).LT.XMIN) XMIN=T(I)
0017                IF(T(I).GT.XMAX) XMAX=T(I)
0018          2 CONTINUE
0019             IF(XMAX.LT.1) XMIN=0
0020             RANGE=XMAX-XMIN
0021             RG=RANGE/10
0022             U(1)=XMIN
0023             DO 3 I=2,11
0024          3 U(I)=U(I-1)+RG
0025                WRITE(1,130)(U(I),I=1,11)
0026                WRITE(5,130)(U(I),I=1,11)
0027             DO 4 K=1,101,10
0028          4 LINE(K)=PLUS
0029                WRITE(1,100)(LINE(K),K=1,101)
0030                WRITE(5,100)(LINE(K),K=1,101)
0031             DO 6 K=1,21
0032             DO 5 I=1,101
0033          5 LINE(I)=BLANK
0034                KPRINT=100*(T(K)-XMIN)/RANGE+1.5
0035                LINE(1)=PLUS
0036                LINE(KPRINT)=ASTERK
0037                WRITE(1,140)P(K), (LINE(I), I=1,101)
0038                WRITE(5,140)P(K), (LINE(I), I=1,101)
0039          6 CONTINUE
0040          100 FORMAT(4X, 'PERCENT DEFECTIVE',101A1)
0041          130 FORMAT(12X,11(2X,E8.3))
0042          140 FORMAT(17X,F4.3,101A1)
0043          RETURN
0044          END

```

```

0001          SUBROUTINE LNC(AQL,NZ,LNC)
0002          C
0003          C*****
0004          C**      THIS SUBROUTINE FINDS THE LIMIT NUMBER **
0005          C**      FOR REDUCED INSPECTION. **
0006          C*****
0007          C
0008          LNC=222
0009          IF(NZ.LE.29.AND.AQL.LE.15) LNC=0
0010          IF(NZ.GE.30.AND.NZ.LE.49.AND.AQL.LE.10) LNC=0
0011          IF(NZ.GE.50.AND.NZ.LE.79.AND.AQL.LE.6.5) LNC=0
0012          IF(NZ.GE.80.AND.NZ.LE.129.AND.AQL.LE.4.0) LNC=0
0013          IF(NZ.GE.130.AND.NZ.LE.199.AND.AQL.LE.2.5) LNC=0
0014          IF(NZ.GE.200.AND.NZ.LE.319.AND.AQL.LE.1.5) LNC=0
0015          IF(NZ.GE.320.AND.NZ.LE.499.AND.AQL.LE.1.0) LNC=0
0016          IF(NZ.GE.500.AND.NZ.LE.799.AND.AQL.LE.0.65) LNC=0
0017          IF(NZ.GE.800.AND.NZ.LE.1249.AND.AQL.LE.0.40) LNC=0
0018          IF(NZ.GE.1250.AND.NZ.LE.1999.AND.AQL.LE.0.25) LNC=0
0019          IF(NZ.GE.2000.AND.NZ.LE.3149.AND.AQL.LE.0.15) LNC=0
0020          IF(NZ.GE.3150.AND.NZ.LE.4999.AND.AQL.LE.0.10) LNC=0
0021          IF(NZ.GE.5000.AND.NZ.LE.7999.AND.AQL.LE.0.065) LNC=0
0022          IF(NZ.GE.8000.AND.NZ.LE.12499.AND.AQL.LE.0.040) LNC=0
0023          IF(NZ.GE.12500.AND.NZ.LE.19799.AND.AQL.LE.0.025) LNC=0
0024          IF(NZ.GE.20000.AND.NZ.LE.31499.AND.AQL.LE.0.015) LNC=0
0025          IF(NZ.GE.31500.AND.NZ.LE.49779.AND.AQL.LE.0.010) LNC=0
0026          IF(LNC.EQ.0) GO TO 10
0027          IF(NZ.LE.29) THEN
0028              IF(AQL.EQ.25) LNC=2
0029              IF(AQL.EQ.40) LNC=4
0030              IF(AQL.EQ.65) LNC=8
0031              IF(AQL.EQ.100) LNC=14
0032          ENDIF
0033          IF(NZ.GE.30.AND.NZ.LE.49) THEN
0034              IF(AQL.EQ.15) LNC=1
0035              IF(AQL.EQ.25) LNC=3
0036              IF(AQL.EQ.40) LNC=7
0037              IF(AQL.EQ.65) LNC=13
0038              IF(AQL.EQ.100) LNC=22
0039          ENDIF
0040          IF(NZ.GE.50.AND.NZ.LE.79) THEN
0041              IF(AQL.EQ.10) LNC=2
0042              IF(AQL.EQ.15) LNC=3
0043              IF(AQL.EQ.25) LNC=7
0044              IF(AQL.EQ.40) LNC=14
0045              IF(AQL.EQ.65) LNC=25
0046              IF(AQL.EQ.100) LNC=40
0047          ENDIF
0048          IF(NZ.GE.80.AND.NZ.LE.129) THEN
0049              IF(AQL.EQ.6.5) LNC=2
0050              IF(AQL.EQ.10) LNC=4
0051              IF(AQL.EQ.15) LNC=7
0052              IF(AQL.EQ.25) LNC=14
0053              IF(AQL.EQ.40) LNC=24
0054              IF(AQL.EQ.65) LNC=42
0055              IF(AQL.EQ.100) LNC=68
0056          ENDIF
0057          IF(NZ.GE.130.AND.NZ.LE.199) THEN

```

```

0059 IF (AQL.EQ.4.0)LNC=2
0060 IF (AQL.EQ.6.5)LNC=4
0061 IF (AQL.EQ.10)LNC=7
0062 IF (AQL.EQ.15)LNC=13
0063 IF (AQL.EQ.25)LNC=25
0064 IF (AQL.EQ.40)LNC=42
0065 IF (AQL.EQ.65)LNC=72
0066 IF (AQL.EQ.100)LNC=115
0067 ENDIF
0068 IF (NZ.GE.200.AND.NZ.LE.319) THEN
0069 IF (AQL.EQ.2.5)LNC=2
0070 IF (AQL.EQ.4.0)LNC=4
0071 IF (AQL.EQ.6.5)LNC=8
0072 IF (AQL.EQ.10)LNC=14
0073 IF (AQL.EQ.15)LNC=22
0074 IF (AQL.EQ.25)LNC=40
0075 IF (AQL.EQ.40)LNC=68
0076 IF (AQL.EQ.65)LNC=115
0077 IF (AQL.EQ.100)LNC=181
0078 ENDIF
0079 IF (NZ.GE.320.AND.NZ.LE.499) THEN
0080 IF (AQL.EQ.1.5)LNC=1
0081 IF (AQL.EQ.2.5)LNC=4
0082 IF (AQL.EQ.4)LNC=8
0083 IF (AQL.EQ.6.5)LNC=14
0084 IF (AQL.EQ.10)LNC=24
0085 IF (AQL.EQ.15)LNC=39
0086 IF (AQL.EQ.25)LNC=68
0087 IF (AQL.EQ.65)LNC=113
0088 IF (AQL.EQ.100)LNC=189
0089 ENDIF
0090 IF (NZ.GE.500.AND.NZ.LE.799) THEN
0091 IF (AQL.EQ.1.0)LNC=2
0092 IF (AQL.EQ.1.5)LNC=3
0093 IF (AQL.EQ.2.5)LNC=7
0094 IF (AQL.EQ.4.0)LNC=14
0095 IF (AQL.EQ.6.5)LNC=25
0096 IF (AQL.EQ.10)LNC=40
0097 IF (AQL.EQ.15)LNC=63
0098 IF (AQL.EQ.25)LNC=110
0099 IF (AQL.EQ.40)LNC=181
0100 ENDIF
0101 IF (NZ.GE.800.AND.NZ.LE.1249) THEN
0102 IF (AQL.EQ.0.65)LNC=2
0103 IF (AQL.EQ.1.0)LNC=4
0104 IF (AQL.EQ.1.5)LNC=7
0105 IF (AQL.EQ.2.5)LNC=14
0106 IF (AQL.EQ.4.0)LNC=24
0107 IF (AQL.EQ.6.5)LNC=42
0108 IF (AQL.EQ.10)LNC=68
0109 IF (AQL.EQ.15)LNC=105
0110 IF (AQL.EQ.25)LNC=181
0111 ENDIF
0112 IF (NZ.GE.1250.AND.NZ.LE.1999) THEN
0113 IF (AQL.EQ.0.40)LNC=2
0114 IF (AQL.EQ.0.65)LNC=4
IF (AQL.EQ.1.0)LNC=7

```



```

0115          IF (AQL.EQ.1.5) LNC=10
0116          IF (AQL.EQ.2.5) LNC=24
0117          IF (AQL.EQ.4.0) LNC=40
0118          IF (AQL.EQ.6.5) LNC=69
0119          IF (AQL.EQ.10) LNC=110
0120          IF (AQL.EQ.15) LNC=169
0121          ENDIF
0122          IF (NZ.GE.2000.AND.NZ.LE.3149) THEN
0123              IF (AQL.EQ.0.25) LNC=2
0124              IF (AQL.EQ.0.40) LNC=4
0125              IF (AQL.EQ.0.65) LNC=8
0126              IF (AQL.EQ.1.0) LNC=14
0127              IF (AQL.EQ.1.5) LNC=22
0128              IF (AQL.EQ.2.5) LNC=40
0129              IF (AQL.EQ.4.0) LNC=68
0130              IF (AQL.EQ.6.5) LNC=105
0131              IF (AQL.EQ.10) LNC=181
0132          ENDIF
0133          IF (NZ.GE.3150.AND.NZ.LE.4999) THEN
0134              IF (AQL.EQ.0.15) LNC=1
0135              IF (AQL.EQ.0.25) LNC=4
0136              IF (AQL.EQ.0.40) LNC=8
0137              IF (AQL.EQ.0.65) LNC=14
0138              IF (AQL.EQ.1.0) LNC=24
0139              IF (AQL.EQ.1.5) LNC=38
0140              IF (AQL.EQ.2.5) LNC=67
0141              IF (AQL.EQ.4.0) LNC=111
0142              IF (AQL.EQ.6.5) LNC=186
0143          ENDIF
0144          IF (NZ.GE.5000.AND.NZ.LE.7999) THEN
0145              IF (AQL.EQ.0.10) LNC=2
0146              IF (AQL.EQ.0.15) LNC=3
0147              IF (AQL.EQ.0.25) LNC=7
0148              IF (AQL.EQ.0.40) LNC=14
0149              IF (AQL.EQ.0.65) LNC=25
0150              IF (AQL.EQ.1.0) LNC=40
0151              IF (AQL.EQ.1.5) LNC=63
0152              IF (AQL.EQ.2.5) LNC=110
0153              IF (AQL.EQ.4.0) LNC=181
0154          ENDIF
0155          IF (NZ.GE.8000.AND.NZ.LE.12499) THEN
0156              IF (AQL.EQ.0.065) LNC=2
0157              IF (AQL.EQ.0.10) LNC=4
0158              IF (AQL.EQ.0.15) LNC=7
0159              IF (AQL.EQ.0.25) LNC=14
0160              IF (AQL.EQ.0.40) LNC=24
0161              IF (AQL.EQ.0.65) LNC=42
0162              IF (AQL.EQ.1.0) LNC=68
0163              IF (AQL.EQ.1.5) LNC=105
0164              IF (AQL.EQ.2.5) LNC=181
0165          ENDIF
0166          IF (NZ.GE.12500.AND.NZ.LE.19999) THEN
0167              IF (AQL.EQ.0.040) LNC=2
0168              IF (AQL.EQ.0.065) LNC=4
0169              IF (AQL.EQ.0.10) LNC=7
0170              IF (AQL.EQ.0.15) LNC=13
0171              IF (AQL.EQ.0.25) LNC=24

```

```

0172          IF (AQL.EG.0.40)LNC=40
0173          IF (AQL.EG.0.65)LNC=69
0174          IF (AQL.EG.1.0)LNC=110
0175          IF (AQL.EG.1.5)LNC=169
0176      ENDIF
0177      IF (NZ.GE.20000.AND.NZ.LE.31499) THEN
0178          IF (AQL.EG.0.025)LNC=2
0179          IF (AQL.EG.0.040)LNC=4
0180          IF (AQL.EG.0.065)LNC=8
0181          IF (AQL.EG.0.10)LNC=14
0182          IF (AQL.EG.0.15)LNC=22
0183          IF (AQL.EG.0.25)LNC=40
0184          IF (AQL.EG.0.40)LNC=68
0185          IF (AQL.EG.0.65)LNC=115
0186          IF (AQL.EG.1.0)LNC=181
0187      ENDIF
0188      IF (NZ.GE.31500.AND.NZ.LE.49999) THEN
0189          IF (AQL.EG.0.015)LNC=1
0190          IF (AQL.EG.0.025)LNC=4
0191          IF (AQL.EG.0.040)LNC=8
0192          IF (AQL.EG.0.065)LNC=14
0193          IF (AQL.EG.0.10)LNC=24
0194          IF (AQL.EG.0.15)LNC=38
0195          IF (AQL.EG.0.25)LNC=67
0196          IF (AQL.EG.0.40)LNC=111
0197          IF (AQL.EG.0.65)LNC=186
0198      ENDIF
0199      IF (NZ.GE.50000) THEN
0200          IF (AQL.EG.0.010)LNC=2
0201          IF (AQL.EG.0.015)LNC=3
0202          IF (AQL.EG.0.025)LNC=7
0203          IF (AQL.EG.0.040)LNC=14
0204          IF (AQL.EG.0.065)LNC=25
0205          IF (AQL.EG.0.10)LNC=40
0206          IF (AQL.EG.0.15)LNC=63
0207          IF (AQL.EG.0.25)LNC=110
0208          IF (AQL.EG.0.40)LNC=181
0209          IF (AQL.EG.0.65)LNC=301
0210      ENDIF
0211      CONTINUE
0212      RETURN
0213      END

```

10

Program PROBS1.FOR

```

0001 SUBROUTINE PROBS1(NN,P,Q,BXLEC)
0002 C *****
0003 C THIS SUBROUTINE COMPUTES CUMULATIVE BINOMIAL
0004 C PROBABILITIES
0005 C *****
0006 C INTEGER C
0007 C DOUBLE PRECISION SUMLOG
0008 C
0009 C COMMON/BLK7/SUMLOG(4000)
0010 C COMMON/BLK8/N
0011 C
0012 C Q=1.-P
0013 C *****
0014 C BINOMIAL PROB. WHEN C=0
0015 C *****
0016 C CSUMS=Q**NN
0017 C IF (C.EQ.0) GOTO 45
0018 C *****
0019 C AVOID RECOMPUTING SUMLOG(I)'S ALREADY IN MEMORY
0020 C *****
0021 C IF (N-NN) 10,25,25
0022 C 10 M=N+1
0023 C *****
0024 C COMPUTE N SUMLOGS-EQUIVALENT TO N-FACTORIAL
0025 C *****
0026 C IF (M.GT.1) GOTO 15
0027 C SUMLOG(1)=0.
0028 C IF (NN.LE.1) GOTO 25
0029 C M=2
0030 C 15 DO 20 I=M, NN
0031 C SUMLOG(I)=DLOG10(DFLOAT(I))+SUMLOG(I-1)
0032 C 20 CONTINUE
0033 C *****
0034 C COMPUTE C SUMS-EQUIVALENT TO SSUM OF PROB.COMPIN.
0035 C I.E. CUMULATIVE BINOMIAL DISTRIBUTION COMPUTATION
0036 C *****
0037 C 25 IF (NN.GT.N) N=NN
0038 C *****
0039 C DETERMINE BEST NUMBER HANDLING LOOP
0040 C *****
0041 C IF (NN.GT.300) GOTO 35
0042 C DO 30 K=1,C
0043 C CSUMS=10.**(SUMLOG(NN)-SUMLOG(NN-K)-SUMLOG(K))
0044 C *P**K*Q**(NN-K)+CSUMS
0045 C 30 CONTINUE
0046 C GOTO 45
0047 C *****
0048 C LOOP FOR LARGE EXPONENTS
0049 C *****
0050 C 35 DO 40 K=1,C
0051 C CSUMS=10.**(SUMLOG(NN)-SUMLOG(NN-K)-SUMLOG(K)
0052 C +K*DLOG10(DBLE(P))+(NN-K)*DLOG10(DBLE(Q)))+CSUMS
0053 C 40 CONTINUE
0054 C
0055 C 45 BXLEC = CSUMS
0056 C RETURN
0057 C END

```

Program PROBD1.FOR

```
0001      SUBROUTINE PROBD1(N1,N2,P,DPROB,C1,C2)
0002      C*****
0003      C      THIS SUBROUTINE COMPUTES DOUBLY PROBABILITIES FOR
0004      C      COMPUTING SECOND SAMPLE NUMBER OF DOUBLE SAMPLING NUMBER
0005      C*****
0006      COMMON/BLK6/C1,C2
0007      INTEGER C1,C2,R1
0008      C
0009      IF(K.EQ.1) CALL PROBS1(N1,P,C1,BXLEC)
0010      IF(K.EQ.2) CALL PROBS2(N1,P,C1,BXLEC)
0011      DPROB=BXLEC
0012      TEMP=BXLEC
0013      NTEMP=C1+1
0014      KTEMP=R1-1
0015      DO 10 IX=NTEMP,KTEMP
0016         I=IX
0017         J=C2-I
0018         IF(K.EQ.1) CALL PROBS1(N1,P,I,BXLEC)
0019         IF(K.EQ.2) CALL PROBS2(N1,P,I,BXLEC)
0020         PROB1=BXLEC-TEMP
0021         TEMP=BXLEC
0022         IF(K.EQ.1) CALL PROBS1(N2,P,J,BXLEC)
0023         IF(K.EQ.2) CALL PROBS2(N2,P,J,BXLEC)
0024         DPROB=DPROB+(PROB1*BXLEC)
0025      10 CONTINUE
0026      C
0027      RETURN
0028      END
```

**END**

**FILMED**

2-85

**DTIC**