

AD735144

Reproduced by
**NATIONAL TECHNICAL
INFORMATION SERVICE**
Springfield, Va. 22151

Unclassified

DOCUMENT CONTROL DATA - R & D

Security classification of title, text, abstract and indexes and diagrams is the same as the overall report. The study is sponsored by the U.S. Army (Corporate author)

University of Georgia

Unclassified

2b. GROUP

Unclassified

REPORT TITLE

PSEUDO RANDOM NUMBER GENERATORS FOR STATISTICAL APPLICATIONS

4. DESCRIPTIVE NOTES (Type of report and, inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

Lovick Edward Cannon, III

6. REPORT DATE

August, 1971

7a. TOTAL NO. OF PAGES

151

7b. NO. OF REFS

21

8a. CONTRACT OR GRANT NO.

9a. ORIGINATOR'S REPORT NUMBER(S)

b. PROJECT NO.

Technical Report No. 69

Department of Statistics & Computer Sci.

c.

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

d.

Themis No. 15

10. DISTRIBUTION STATEMENT

Reproduction in whole or part is permitted for any purpose of the
United States Government.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

Office of Naval Research
Washington, D. C.

13. ABSTRACT

Pseudo random number generating programs coded in machine language for the IBM 360/65 computer have been prepared for producing random values from the normal, exponential, and gamma distributions as well as from any discrete probability distribution. The schemes are based largely on combinations of sophisticated techniques first suggested by Marsaglia. The distributions simulated are exact within the word size of the computer and average production time per number is very short, e.g. 10 to 40 microseconds depending upon the particular distribution.

DD FORM 1473 (PAGE 1)

Unclassified

Security Classification

A-31408

Unclassified

KEY WORDS	GROUP A		GROUP B		GROUP C	
	ROLL	WT	ROLL	WT	ROLL	WT
PSEUDO RANDOM NUMBERS						
SIMULATION						
MONTE CARLO						
COMPUTER PROGRAMS						

DD FORM 1473 (11-63)
1 NOV 63

Unclassified
Security Classification

A-1100

THEMIS REPORT NUMBER 15

TECHNICAL REPORT NUMBER 69

PSEUDO RANDOM NUMBER GENERATORS FOR
STATISTICAL APPLICATIONS

LOVICK EDWARD CANNON, III

Reproduction in whole or in part is permitted for any
purpose of the United States Government. This Research
was supported, in part, by the Office of Naval Research,
Contract Number N00014-69-A-0423, NR 042-261.

ROLF BARGMANN
PRINCIPAL INVESTIGATOR

THE UNIVERSITY OF GEORGIA
DEPARTMENT OF STATISTICS AND COMPUTER SCIENCES
ATHENS, GEORGIA 30601

AUGUST 1971

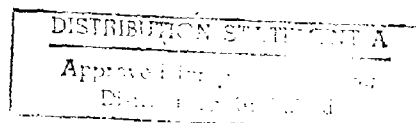
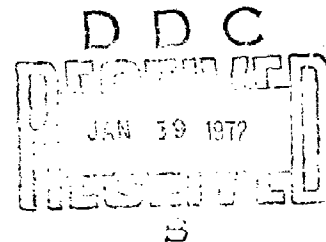


TABLE OF CONTENTS

Chapter		Page
1.	INTRODUCTION.....	1
2.	GENERATION OF DISCRETE RANDOM VARIABLES IN A COMPUTER.....	6
3.	GENERAL TECHNIQUES FOR THE GENERATION OF PSEUDO-RANDOM NUMBERS HAVING A CONTINU- OUS DISTRIBUTION FUNCTION.....	17
	3.1 The Composition Technique.....	17
	3.2 The Acceptance-Rejection Method.....	22
	3.3 Generation of Numbers with a Triangular Density.....	23
	3.4 The Distribution of the Minimum of a Random Number of Uniform (0,1) Random Variates: Some Special Results for the Exponential Distribution.....	25
	3.5 Generating Numbers from a 'Nearly Triangular' Density: Application to the Half Normal Density.....	28
	3.6 Generation of Random Variates from the Tail of the Half Normal Distribution.....	34

Chapter		Page
4.	GENERATION OF PSEUDO-RANDOM NUMBERS FROM THE EXPONENTIAL, GAMMA, AND NORMAL DISTRIBUTIONS.....	37
4.1	Generation of Exponentially Distri- buted Random Numbers.....	50
4.2	Generation of Gamma Distributed Random Variates.....	55
4.3	Generation of Random Variates having the Standard Normal Distribution.....	56
5.	RESULTS OF TESTS PERFORMED ON SEQUENCES OF PSEUDO-RANDOM NUMBERS.....	62
5.1	Random Numbers from the Uniform (0,1) Distribution.....	62
5.2	Random Numbers having a Discrete Distribution.....	69
5.3	Random Numbers from Continuous Distribution Functions.....	76
5.4	Discussion of Results.....	79
6.	DISCRIPTION AND USE OF GENERATOR PROGRAMS.....	80
6.1	GEN0.....	80
6.2	GEN1.....	81
6.3	GEN2.....	81
6.4	GEN3.....	82
6.5	GEN4.....	82
6.6	GEN5.....	84
6.7	GEN6.....	86
6.8	GEN7.....	87

	Page
REFERENCES.....	91
APPENDIX	
A Assembler Listing of GEN0.....	93
B Assembler Listing of GEN1.....	95
C Assembler Listing of GEN2.....	107
D Assembler Listing of GEN3.....	119
E Assembler Listing of GEN4.....	133
F Assembler Listing of GEN5.....	139
G Assembler Listing of GEN6.....	144
H Assembler Listing of GEN7.....	150

LIST OF TABLES

Table		Page
2.1	Compact Storage of Discrete Variates.....	11
4.0.1	Certain Values Involved in the Generation of Random Variates from $f(y)=(3/2)(1-y^2)$, $0<y<1$	40
4.0.2	Storage of Discrete Variates Required in the Generation of Random Variates from $f(y)=(3/2)(1-y^2)$, $0<y<1$	41
4.0.3	Values Required for the Generation of Random Variates from $f(y)=(3/2)(1-y^2)$, $0<y<1$	43
4.0.4	Values Required for the Generation of Random Variates from $f(y)=(3/2)(1-y^2)$, $0<y<1$	47
5.1.1	Results of Tests Performed on Random Numbers from the Uniform (0,1) Distribution.....	66
5.2.1	Results of Chi-Square Goodness of Fit Tests on Random Numbers from the Binomial Density.....	71
5.2.2	Results of Chi-Square Goodness of Fit Tests on Random Numbers from the Poisson Density.....	73
5.2.3	Results of Chi-Square Goodness of Fit Tests on Random Numbers from the Negative Binomial Density..	74
5.3.1	Results of Chi-Square Goodness of Fit Tests on Random Numbers from the Exponential Density.....	77
5.3.2	Results of Chi-Square Goodness of Fit Tests on Random Numbers from the Normal Density.....	78

Table		Page
5.3.3	Results of Chi-Square Goodness of Fit Tests on Transformed Gamma Random Numbers.....	79
6.1	Possible Instructional Alterations in GEN7.....	90

LIST OF ILLUSTRATIONS

Figure		Page
2.1	Flowchart for the Generation of Random Numbers from a Discrete Distribution.....	14
3.1	Decomposition of a Continuous Distribution.....	18
3.2	Illustration of Acceptance-Rejection Principle.....	22
3.3(a)	Triangular Distribution.....	24
3.3(b)	Transformed Triangular Distribution.....	24
3.4(a)	Decomposition of the Exponential Density into Rectangles and "Teeth".....	26
3.4(b)	Distribution of the "Tooth" from the Exponential Distribution between 0 and Δ	26
3.4(c)	Distribution of any Arbitrary "Tooth" from the Exponential Density Defined between $k \cdot \Delta$ and $(k + 1) \cdot \Delta$ for $k = 0, 1, \dots$	27
3.5(a)	Concave Triangular Densities from the Half Normal Distribution.....	29
3.5(b)	Convex Triangular Densities from the Half Normal Distribution.....	29
3.5(c)	Joint Density of the Minimum and the Maximum of Two Uniformly (0,1) Distributed Random Variates....	30
3.5(d)	Joint Distribution of the Minimum and the Maximum of Two Uniformly (0,1) Distributed Random Variates....	31

Figure		Page
3.6(a)	Joint Density of $x_1 = \sqrt{a^2 - 2\ln(u_1^2 + u_2^2)}$ $\cdot \sqrt{u_1^2/(u_1^2 + u_2^2)}$ and $x_2 = \sqrt{a^2 - 2\ln(u_1^2 + u_2^2)} \cdot \sqrt{u_2^2/(u_1^2 + u_2^2)}$	34
4.0(a)	Decomposition of $f(y) = (3/2)(1 - y^2)$, $0 < y < 1$	38
4.0(b)	Rectangular Portion of $f(y) = (3/2)(1 - y^2)$, $0 < y < 1$	38
4.0(c)	Triangular Portion of $f(y) = (3/2)(1 - y^2)$, $0 < y < 1$	38
4.0(d)	Application of the Acceptance-Rejection Principle to the Triangular Portion of $f(y) = (3/2)(1 - y^2)$, $0 < y < 1$	45
4.0(e)	Flowchart for the Generation of Random Variates from $f(y) = (3/2)(1 - y^2)$, $0 < y < 1$	48

CHAPTER 1

INTRODUCTION

With the increasing use of electronic computers, random sampling methods have become a very useful tool for providing solutions to problems involving probability as well as for problems of a deterministic nature. The availability of sequences of numbers which appear to be drawn at random from particular probability distributions is a vital ingredient in the random sampling process. Such numbers are referred to as pseudo-random numbers or, for convenience, simply as random numbers. This paper is concerned with the rapid and accurate generation of such numbers in a stored computer program.

The use of random sampling to estimate distribution functions originated with Student [17] in 1908. At that time and for some time to follow, necessary random numbers were obtained by drawing cards from a deck, counters from an urn, or by rolling dice. Such processes are very slow and make it quite difficult to insure randomness.

To facilitate the use of random numbers large tables of random digits were compiled. The first such table, published by Tippett [19] in 1927, consisted of 41,600 digits taken at random from census reports and combined into 10,400 four digit numbers. The requirement for a larger set of numbers led to the publication of 100,000 random digits in 1939 by Kendall and Babington-Smith [7]. These digits were generated by means of

a mechanical process and were the first to be so produced. Kendall and Babington-Smith are also responsible for developing many of the tests frequently applied to sequences of random numbers (5,6). Other such tables have since been constructed, primarily by the use of physical devices, culminating in the most extensive, "A Million Random Digits with 100,000 Normal Deviates" published by the RAND Corporation [15] in 1955.

The use of a physical device for the generation of random numbers on line with a computer is both expensive and difficult to maintain. The storing of a table of random numbers on magnetic tape or on cards is also an unsatisfactory method of generating numbers for use in a computer. This would necessitate the use of an input device; also the time required to read in the numbers would be excessive.

The development of arithmetic procedures for the generation of random numbers began in the 1940's with the introduction of computers. The first such procedure was suggested by von Neumann and Metropolis in 1946 and is described in [4]. It was also during this time that calculations involving random numbers received the picturesque name "Monte Carlo." The "middle-square" method proposed by von Neumann and Metropolis is simple, fast, and requires only an initial starting value. In this procedure each new number is produced by taking the middle n digits of the square of the previous n -digit number. As pointed out in [4] and in [16] the sequence of numbers produced using this method sooner or later degenerates to a cycle which often is very small, and at worst consists of only a single number. In addition some of the statistical tests performed on samples generated by the middle-square procedure have resulted in failures. Improvements on the von Neumann method are plagued by similar difficulties

[2] and [20].

Of the various arithmetic procedures sequences of random numbers with the best statistical properties and longest periods are generated by means of the congruence relation

$$x_{n+1} = a \cdot x_n + c \pmod{m}. \quad (1.1)$$

The representation (1.1) is termed the mixed congruential method. The multiplicative congruential method is defined by taking $c = 0$ in (1.1). Lehmer [8] is credited with the invention of the multiplicative method. In their very informative survey Hull and Dobell [3] prescribe conditions for a , m , c , and x_0 which insure maximum period. Other procedures for the generation of random numbers based upon reduced Fibonacci series [21] and upon transcendental numbers [3] are inferior to those based on the congruence relation (1.1) (See [4]). In [9] MacLaren and Marsaglia present a table look-up scheme which seems to offer some promise.

Random numbers generated by arithmetic procedures are not "truly random" in that the entire sequence is determined in advance and can be reproduced simply by using the same starting value, x_0 . The statistical behavior of sequences generated by both the mixed and multiplicative congruential methods is quite good, and, in fact, numbers generated in this fashion do appear to be drawn at random from the uniform (0,1) distribution

$$f(u) = 1, \quad 0 < u < 1. \quad (1.2)$$

The reproducibility of sequences is an advantage in debugging and in certain calculations it may be desirable to reproduce a given sequence.

Bargmann [1] has provided a procedure for the generation of independent uniform (0,1) random numbers on a binary computer. This procedure, which is a form of the multiplicative congruential method, is defined by

$$u_{n+1} = a \cdot u_n \pmod{2^{32}} \quad (1.3)$$

where a is chosen such that $a + 1$ and $a - 1$ end in as few zero bits as possible. This is insured by requiring that neither $a - 1$ nor $a + 1$ be divisible by 2^k for $k = 3, 4, \dots$. Choosing a as an odd power of 5 determines that the low order 3 digits will be 125, and neither 124 nor 126 is divisible by 2^k for $k = 3, 4, \dots$. Hence, $a = 5^{13}$ is a reasonable choice. This procedure requires that u_0 , the starting value, be an odd positive integer.

The use of this procedure results in a very fast computer program requiring only 3 operations: load, multiply, and store. A description of the usage of this program along with time and storage requirements is given in Chapter 6 of this paper. Results of statistical tests performed on this procedure are presented in Chapter 5, and a listing of the program is given in Appendix A.

The preceding discussion has dealt only with the generation of random numbers which appear to be uniformly distributed. In principle it should be very easy to obtain any other distribution from the uniform distribution, requiring only a solution to the equation

$$u = F(y) \quad (1.4)$$

for y , where u is uniformly distributed on the interval (0,1) and F is the

required cumulative distribution function. When the inverse of F is known, as for the exponential distribution, this is a simple matter:

$$y = -\ln(u). \quad (1.5)$$

However, the evaluation of the $\ln$ function may be somewhat time consuming. An alternative approach might be to store a value of y for each possible value of u based on the relation

$$y = F^{-1}(u). \quad (1.6)$$

To generate y , simply generate a uniform $(0,1)$ random number u which will determine the location of a stored value of y . Let u be given to 8 digits and let β be the base of the number system in which the digits are represented, then for this procedure a total of β^8 storage locations are required.

Various approximations have been proposed for the normal distribution. Most of these involve taking the sum of a fixed number of uniform $(0,1)$ deviates or the use of Chebyshev approximations [14] or a combination of the two [18]. Such approximations frequently lack accuracy and are either slow or space consuming.

The procedures suggested by Marsaglia et al. in [11],[12], and [13] for transforming uniform $(0,1)$ random numbers to random numbers having other distributions are superior to any encountered. They are fast, require minimal space, and are simple to program. In addition these procedures are completely accurate; the precision of the result is dependent only on the word size of the computer. These procedures along with programs and results of statistical tests are presented in detail in this paper.

CHAPTER 2

GENERATION OF DISCRETE RANDOM VARIABLES IN A COMPUTER

Let Y be a discrete random variable with point probabilities $p_i = \Pr(Y = y_i)$ for $i = 1, 2, \dots$. The direct way to generate Y in a computer is to generate a uniform $(0,1)$ random number u and put $Y = y_i$ if $p_1 + p_2 + \dots + p_{i-1} < u \leq p_1 + p_2 + \dots + p_i$. However, techniques based on this method lead to complicated programs that are excessively time consuming.

An alternative method proposed by Marsaglia [11] is simple to program and requires minimal time and storage. Let p_i for $i = 1, 2, \dots, n$ be expressed by k digits as $p_i = .\delta_{1i}\delta_{2i} \dots \delta_{ki}$. If the domain of the random variable is infinite, the probability distribution must be truncated at some p_n . The fastest method for generating Y is as follows.

Let β be the base of the number system in which the δ_{ji} 's are represented. In memory locations 0 to $(\beta^k - 1)$ store $\delta_{11}\delta_{21}\delta_{31} \dots \delta_{k1}$ y_1 's, $\delta_{12}\delta_{22} \dots \delta_{k2}$ y_2 's, $\dots$, $\delta_{1n}\delta_{2n} \dots \delta_{kn}$ y_n 's. If u is a uniform $(0,1)$ random number, $u = .d_1d_2 \dots d_k$, look up the number in location $d_1d_2 \dots d_k$ and let that be Y .

Though this may be the fastest method, it clearly requires an excessive amount of storage space. Even if the p_i 's are truncated to four digits, β^4 memory locations will be required.

Marsaglia [11] suggests a technique that offers a considerable reduction in storage space with very little sacrifice in execution time. For convenience assume that the p_i 's are truncated to four digits. We have the following situation:

Value of Y	Probability
y_1	$\cdot \delta_{11} \delta_{21} \delta_{31} \delta_{41}$
y_2	$\cdot \delta_{12} \delta_{22} \delta_{32} \delta_{42}$
y_n	$\cdot \delta_{1n} \delta_{2n} \delta_{3n} \delta_{4n}$

Define

$$P_0 = 0, \quad P_r = \beta^{-r} \sum_{i=1}^n \delta_{ri} \quad (2.0.1)$$

for $r = 1, 2, 3, 4$, and

$$\Pi_s = \sum_{j=1}^s \sum_{i=1}^n \delta_{ji} \quad (2.0.2)$$

for $s = 1, 2, 3, 4$ and $\Pi_0 = 0$. Segment memory locations 0 to $(\Pi_4 - 1)$ into four mutually exclusive sets such that set 1 consists of locations 0 to $(\Pi_1 - 1)$, set 2 comprises locations Π_1 to $(\Pi_2 - 1)$, and in general the s^{th} set occupies memory locations Π_{s-1} to $(\Pi_s - 1)$ for $s = 1, 2, 3, 4$. In each set s for $s = 1, 2, 3, 4$ each y_i is stored in δ_{si} locations for $i = 1, 2, \dots, n$, a total of $\Pi_s - \Pi_{s-1}$ locations. The total memory requirement is then $\Pi_4 = \sum_{j=1}^4 \sum_{i=1}^n \delta_{ji}$ locations. Now choose set 1 with probability P_1 , set 2 with probability P_2 , set 3 with probability P_3 , or

set 4 with probability P_4 . Having chosen a set, select at random a location within that set; the number occupying that location is the desired random variable.

This procedure gives the required discrete distribution, as can be seen by defining:

$$q_{ri} = \delta_{ri} / \sum_{i=1}^n \delta_{ri}$$

for $i = 1, 2, \dots, n$ and $r = 1, 2, 3, 4$. Thus q_{ri} is simply the probability of choosing y_i from set r . Then the probability of generating $Y = y_i$ is

$$\sum_{r=1}^4 P_r \cdot q_{ri} = \sum_{r=1}^4 \beta^{-r} \delta_{ri} = .\delta_{1i} \delta_{2i} \delta_{3i} \delta_{4i},$$

which is the probability $P(Y = y_i)$.

In order to select the proper set and location within that set, generate a uniform (0,1) random variable $u = .d_1 d_2 \dots$, and let $A\{j\}$ denote the contents of memory location j . Then if

$$\sum_{r=0}^{s-1} P_r \leq u \leq \sum_{r=0}^s P_r$$

put

$$Y = A \{d_1 d_2 \dots d_s + \pi_{s-1} - \beta^s \sum_{r=0}^{s-1} P_r \}.$$

The following example illustrates this procedure. The hexadecimal ($\beta=16$) number system is used as it is the representation of the IBM 360/65,

i	y_i	P_i
1	0	0.A000
2	1	0.3800
3	2	0.14A2
4	3	0.0F50
5	4	0.0202
6	5	0.020C

Defining P_r and Π_s as before gives

$$P_0 = 0$$

$$P_1 = 16^{-1} \sum_{i=1}^6 \delta_{1i} = 0.E$$

$$P_2 = 16^{-2} \sum_{i=1}^6 \delta_{2i} = 0.1F$$

$$P_3 = 16^{-3} \sum_{i=1}^6 \delta_{3i} = 0.00F$$

$$P_4 = 16^{-4} \sum_{i=1}^6 \delta_{4i} = 0.0010$$

and

$$\Pi_0 = 0$$

$$\Pi_1 = \sum_{j=1}^1 \sum_{i=1}^6 \delta_{ji} = E$$

$$\Pi_2 = \sum_{j=1}^2 \sum_{i=1}^6 \delta_{ji} = 2D$$

$$\Pi_3 = \sum_{j=1}^3 \sum_{i=1}^6 \delta_{ji} = 3C$$

$$\Pi_4 = \sum_{j=1}^4 \sum_{i=1}^6 \delta_{ji} = 4C.$$

The four sets are stored as follows:

A - TABLE

SET 1		SET 2				SET 3		SET 4	
Loc.	Con.	Loc.	Con.	Loc.	Con.	Loc.	Con.	Loc.	Con.
0	0	E	1	1E	3	2D	2	3C	2
1	0	F	1	1F	3	2E	2	3D	2
2	0	10	1	20	3	2F	2	3E	4
3	0	11	1	21	3	30	2	3F	4
4	0	12	1	22	3	31	2	40	5
5	0	13	1	23	3	32	2	41	5
6	0	14	1	24	3	33	2	42	5
7	0	15	1	25	3	34	2	43	5
8	0	16	2	26	3	35	2	44	5
9	0	17	2	27	3	36	2	45	5
A	1	18	2	28	3	37	3	46	5
B	1	19	2	29	4	38	3	47	5
C	1	1A	3	2A	4	39	3	48	5
D	2	1B	3	2B	5	3A	3	49	5
		1C	3	2C	5	3B	3	4A	5
		1D	3					4B	5

TABLE 2.1

To generate the desired random variable, let $u = .d_1d_2 \dots$ be a uniform (0,1) random number and let $A(j)$ denote the contents of memory location j and proceed as follows:

- 1) If $0 \leq u < 0.E$ put $Y = A\{d_1\}$
- 2) If $0.E0 \leq u < 0.FF$ put $Y = A\{d_1d_2 - D2\}$
- 3) If $0.FF0 \leq u < 0.FFF$ put $Y = A\{d_1d_2d_3 - FC3\}$
- 4) If $0.FFF0 \leq u$ put $Y = A\{d_1d_2d_3d_4 - FFB4\}$.

Examples:

$$u = 0.2170 \dots Y = A\{2\} = 0$$

$$u = 0.EF10 \dots Y = A\{EF-D2\} = A\{1D\} = 3$$

$$u = 0.FFFE \dots Y = A\{FFFE - FFB4\} = A\{4A\} = .5.$$

In this example $4C_{(16)}$ memory locations are required as compared to 16^4 memory locations for the fastest method. Suppose the times for certain operations in the computer are:

<u>Operation</u>	<u>Time</u>
Compare two integers	P
Subtract two integers	S
Look up an addressed location	L

Then the fastest method for generating Y requires a total time of $P + L$ and 16^4 memory locations. In the example presented only $4C_{(16)}$ memory locations are required and an average generation time of

$$0.E \cdot (P + L) + 0.F \cdot (2P + S + L) + 0.00F \cdot (3P + S + L) + 0.001$$

$$\cdot (3P + S + L) = 1.21_{(16)} \cdot P + 0.20_{(16)} \cdot S + L$$

A schematic for storing the four sets of discrete variables and the subsequent generation of the desired random variables is presented in figure 2.1.

FIGURE 2.1

Flowchart for the Generation of Random
Numbers from a Discrete Distribution

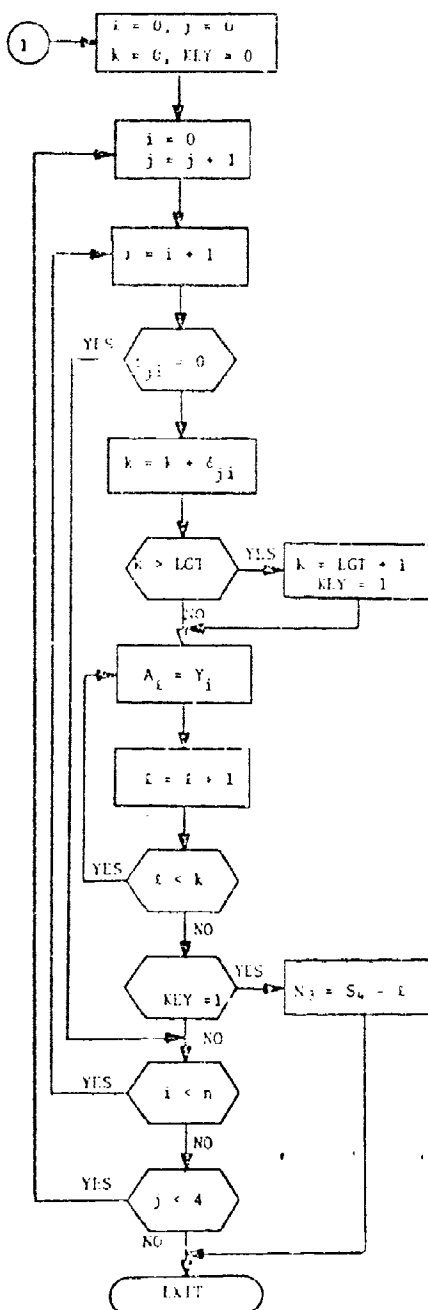
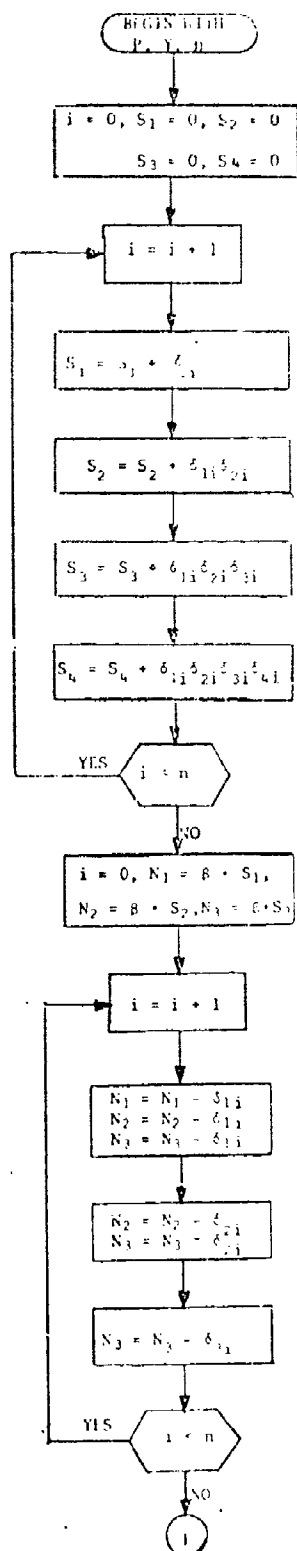
P is a vector of probabilities ($p_i = .\delta_{1i}\delta_{2i}\delta_{3i}\dots$)

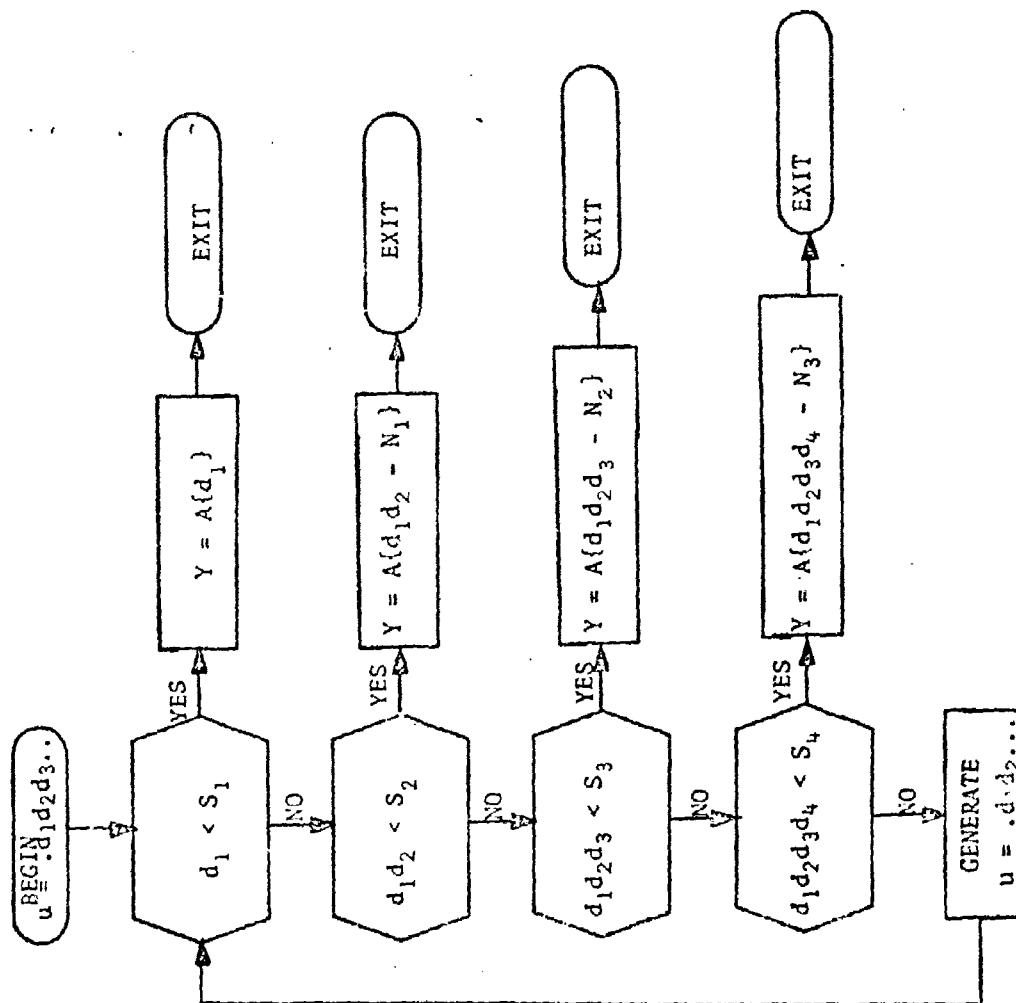
Y is a vector of the corresponding discrete variables (integer mode)

n is the number of elements in P and in Y.

β is the base of the number system in which the δ_{ij} 's are represented
(for the 360/65 $\beta = 16$, for a decimal machine $\beta = 10$).

LGT is the maximum length of the A-Table. If LGT is exceeded, the table
loading process ceases and N_3 is adjusted (since N_3 was calculated on the
basis that loading would be complete).





CHAPTER 3
GENERAL TECHNIQUES FOR THE GENERATION
OF PSEUDO-RANDOM NUMBERS HAVING A
CONTINUOUS DISTRIBUTION FUNCTION

Certain techniques of a rather general nature are used in some of the routines described in this paper. They are especially well suited for use in very fast computer programs. The principle of each technique is described below and reference will be made to these techniques as they are used in the specific generator programs.

3.1 The Composition Technique

The fundamental procedure used for generating pseudo-random numbers from a continuous probability distribution, as suggested by Marsaglia [10], is based upon a decomposition of the density function f into a mixture of 3 densities:

$$f(t) = p_1 g_1(t) + p_2 g_2(t) + p_3 g_3(t) \quad (3.1.1)$$

where $p_1 > p_2 > p_3$, $p_1 + p_2 + p_3 = 1$ and p_1 is very close to 1.

A random number from $f(t)$ is obtained as either a number from $g_1(t)$, from $g_2(t)$ or from $g_3(t)$ with respective probabilities p_1 , p_2 , p_3 . Consider a density $f(t)$ defined from all $t \geq a$ (figure 3.1).

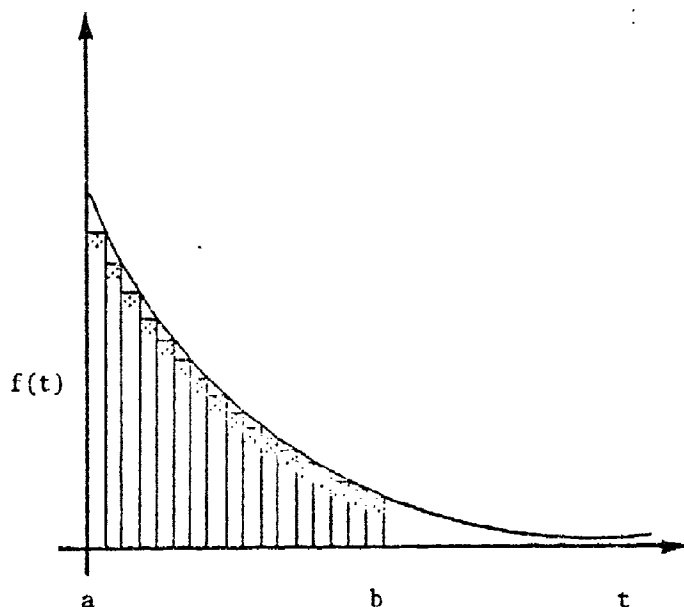


FIGURE 3.1

The density $g_1(t)$ is represented by the rectangles, appropriately standardized, including the shaded upper portions, and is defined for $a \leq t < b$. The width of the rectangles is Δ , a quantity whose value is dictated by the number system used within the computer. For a binary machine using hexadecimal arithmetic, $\Delta = 0.1_{(16)} = 0.0625_{(10)}$. The total area represented by the rectangles is p_1 which clearly is very close to 1. The density $g_2(t)$ is represented by the "triangular" regions, appropriately standardized, lying below $f(t)$ and above $g_1(t)$, and is also defined for $a \leq t < b$. The total area occupied by the "triangles" is p_2 . The density $g_3(t)$ is defined for $t \geq b$ and represents

the tail of $f(t)$. Its area is p_3 .

Random variables having density $g_1(t)$ may be rapidly generated as follows:

Define

$$N = (b-a)/\Delta \quad (3.1.2)$$

and

$$t_i = a + (i-1) \cdot \Delta \quad \text{for } i = 1, 2, 3, \dots, N. \quad (3.1.3)$$

Now assign each discrete t_i a probability

$$P_i = \Delta \cdot f(t_{i+1}) = .\delta_{1i}\delta_{2i}\dots \quad (3.1.4)$$

The P_i 's are simply the areas of the individual rectangles comprising $g_1(t)$. Based on (say) the high order 4 digits of the P_i 's store the discrete t_i 's in a table according to the technique described in Chapter 2 of this paper. Thus to generate a random variable Y from this portion of g_1 (unshaded in figure 3.1), generate a uniform (0,1) random number $u = .d_1d_2 \dots$. If

$$u < \sum_{i=1}^N .\delta_{1i}\delta_{2i}\delta_{3i}\delta_{4i}$$

then allow the 4 high order digits of u , $d_1d_2d_3d_4$, to locate a particular t_i from the discrete table, as described in Chapter 2, and set $Y = t_i + \Delta \cdot (.d_5d_6\dots)$.

Random variables from the residual of g_1 , corresponding to the remaining digits of the P_i 's ($.0000\delta_{5i}\delta_{6i}\dots$) and represented by the shaded region in figure 3.1, are generated according to the following scheme:

Let N , t_i , and P_i be defined as before (3.1.2), (3.1.3), and (3.1.4); and define:

$$H_i = .\delta_{1i}\delta_{2i}\delta_{3i}\delta_{4i} \quad (3.1.5)$$

(area of the unshaded portion of the i^{th} rectangle),

$$R_i = P_i - H_i \quad (3.1.6)$$

(area of the shaded portion of the i^{th} rectangle),

$$T_i = F(t_{i+1}) - F(t_i) - P_i \quad (3.1.7)$$

where

$$F(t_i) = \int_a^{t_i} f(t_i) dt.$$

(T_i is the area of the "triangular" region above the i^{th} rectangle), and

$$P'_i = \sum_{i=1}^N H_i \quad (3.1.8)$$

(total area occupied by the unshaded rectangles). It is obvious from the above definitions that

$$P_2 = \sum_{i=1}^N T_i, \quad (3.1.9)$$

$$P_1 = \sum_{i=1}^N P_i, \text{ and} \quad (3.1.10)$$

$$P_1 - P'_1 = \sum_{i=1}^N R_i \quad (3.1.11)$$

(See(3.1.1) for the significance of p_1 and p_2). Now store the t_i 's in N consecutive locations. The order in which the t_i 's are stored is not important*. However, for convenience let $D(1)$ denote the t -value occupying

* They may be overlapped with the previously stored t -values in order to save storage space.

the first location, $D(2)$ denote the t -value occupying the second location, etc. For each t_i define a pair of values $C(k)$ and $B(k)$ as follows:

$$C(1) = p_1' + R[D(1)], \quad (3.1.12)$$

$$B(k) = C(k) + T[D(k)], \text{ and} \quad (3.1.13)$$

$$C(k+1) = C(k) + R[D(k+1)] \quad (3.1.14)$$

for $k = 1, 2, \dots, N$. $R[D(k)]$ and $T[D(k)]$ denote respectively the area of the rectangle residue (shaded in figure 3.1) and the area of the "triangular" region corresponding to the t -value occupying location $D(k)$. It is apparent that

$$B(N) = p_1 + p_2. \quad (3.1.15)$$

If u is a uniform $(0,1)$ random number such that $p_1' \leq u < p_1 + p_2$, a random variable Y is generated from either the rectangular residues of g_1 or the "triangular" regions of g_2 as follows:

If $B(k-1) \leq u < C(k)$ then generate a new uniform $(0,1)$ random number v and set $Y = D(k) + \Delta \cdot v$. A random variable generated in this fashion will have the density represented by the shaded rectangles in figure 3.1.

If $C(k) \leq u < B(k)$, then the random variable Y must be generated from the "triangular" density of g_2 corresponding to the t -value occupying location $D(k)$. The method used depends upon the particular function $f(t)$. The same applies for random variables from g_3 for $t > b$.

The techniques used for the exponential and normal densities will be described subsequently.

Since p_1 is close to 1, most of the time a random variable from g_1 is generated. Though g_2 and g_3 may be complicated and require longer running time, the average time per generation is small since they must be handled so rarely.

3.2 The Acceptance-Rejection Method

Consider the density $h(t)$ defined on the interval (ξ_0, ξ_1) . See figure 3.2. Now if u, v is a pair

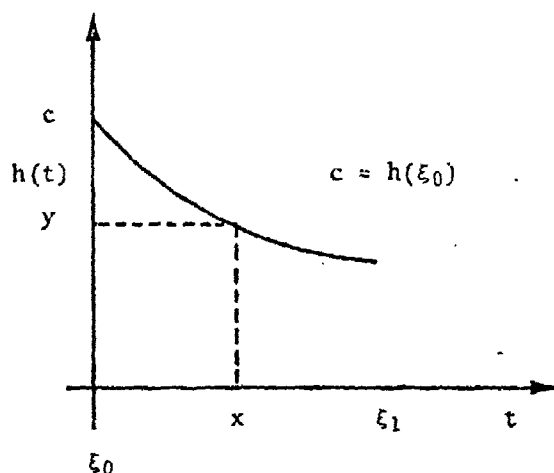


FIGURE 3.2

of independent uniform $(0,1)$ random numbers, $x = \xi_0 + u \cdot (\xi_1 - \xi_0)$ and $y = v \cdot c$ will have uniform densities on (ξ_0, ξ_1) and $(0, c)$ respectively. Suppose that $y \leq h(x)$. Then the conditional distribution function of x is given by

$$P[X \leq x \mid Y \leq h(x)] = \frac{\int_{\xi_0}^x h(t) dt}{\int_{\xi_0}^x h(t) dt} = \frac{\int_{\xi_0}^x h(t) dt}{\int_{\xi_0}^x h(t) dt}. \quad (3.2.1)$$

Consequently the conditional density of x given $y \leq h(x)$ is $h(x)$.

When a pair of independent uniform $(0,1)$ numbers, u and v , are generated and the resulting x and y satisfy the inequality $y \leq h(x)$, x is then a required random variate from $h(x)$. Otherwise the pair is rejected and a new pair is generated and checked. It is clear that the inequality will be satisfied with a high probability only if the value $c \cdot (\xi_1 - \xi_2)$ is close to one. If not, the procedure will produce a large proportion of inadmissible (u,v) pairs and reduce the efficiency of the scheme. Even when this proportion is small, the time required to check $y \leq h(x)$ is generally rather long compared with that of the table look-up procedures. Consequently, the acceptance-rejection method is best used for generating those infrequent values from the "triangular" and tail regions described in section 3.1.

3.3 Generation of Numbers with a Triangular Density

Let u and v be independent uniform $(0,1)$ random numbers. Then $T = \min(u,v)$ has the distribution function

$$G(t) = 1 - P[T > t] = 1 - P[u > t, v > t] = 1 - (1-t)^2, \quad 0 < t < 1, \quad (3.3.1)$$

and the density function

$$g(t) = 2(1-t), \quad 0 < t < 1. \quad (3.3.2)$$

See figure 3.3a. A linear transformation $T' = a + bT$ produces a random variable with density

$$g(t') = \frac{2}{b} - \frac{2}{b^2} (t' - a), \quad a < t' < a+b \quad (3.3.3)$$

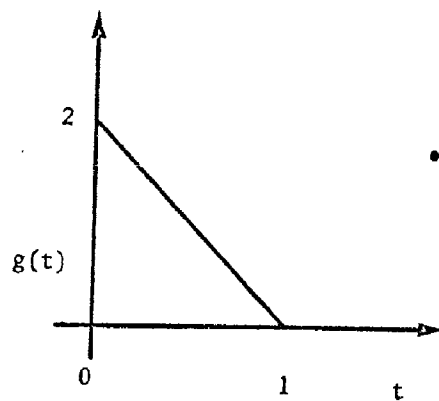


FIGURE 3.3(a)

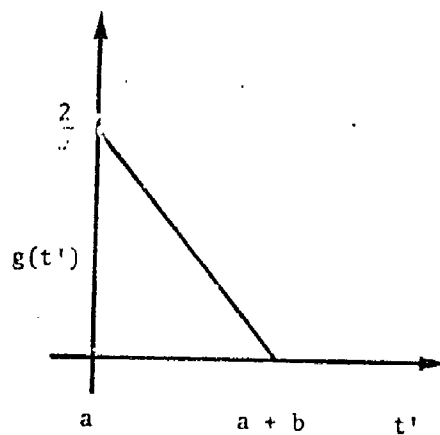


FIGURE 3.3(b)

3.4 The Distribution of the Minimum of a Random Number of Uniform (0,1) Random Variates: Some Special Results for the Exponential Distribution

Let $x = c \cdot \min(u_1, u_2, \dots, u_n)$ where the u_i are random independent uniform (0,1) variates, c is a positive constant, and N is a discrete valued random variable with probability function $P(N=n) = q(n)$ for $n = 1, 2, \dots$. Then the distribution function of x can be expressed as follows:

$$F(x) = \sum_{n=1}^{\infty} P[X \leq x \mid N = n] \cdot P[N = n]$$

$$= \sum_{n=1}^{\infty} [1 - (1 - x/c)^n] \cdot q(n)$$

$$F(x) = 1 - \sum_{n=1}^{\infty} (1 - x/c)^n \cdot q(n) \quad 0 < x < c \quad (3.4.1)$$

Now consider the special case

$$q(n) = c^n / [n! (e^c - 1)] \quad n = 1, 2, \dots$$

The distribution of x becomes

$$F(x) = 1 - \sum_{n=1}^{\infty} (c-x)^n / n! (e^c - 1) = 1 - (e^{c-x} - 1) / (e^c - 1)$$

$$F(x) = (1 - e^{-x}) / (1 - e^{-c}) \quad 0 < x < c \quad (3.4.3)$$

Thus x has the exponential distribution truncated on the right at the value c .

If, instead of the probability function given in (3.4.2), N has the distribution specified by

$$g(n) = \Delta^n / n! (e^\Delta - 1 - \Delta) \quad n = 2, 3, \dots, \quad (3.4.4)$$

where Δ is a constant as defined in section 3.1, then the distribution function of x becomes

$$F(x) = 1 - (e^{\Delta-x} - 1 - \Delta x) / (e^\Delta - 1 - \Delta), \quad 0 < x < \Delta, \quad (3.4.5)$$

with density

$$f(x) = (e^{-x} - e^{-\Delta}) / [1 - e^{-\Delta}(1 + \Delta)] \quad , \quad 0 < x < \Delta. \quad (3.4.6)$$

This is the distribution function of a random variable from the "tooth" of the exponential distribution between 0 and Δ . See figures 3.4(a) and 3.4(b).

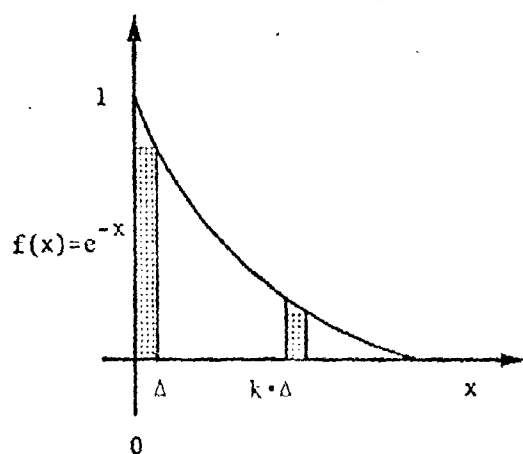


FIGURE 3.4(a)

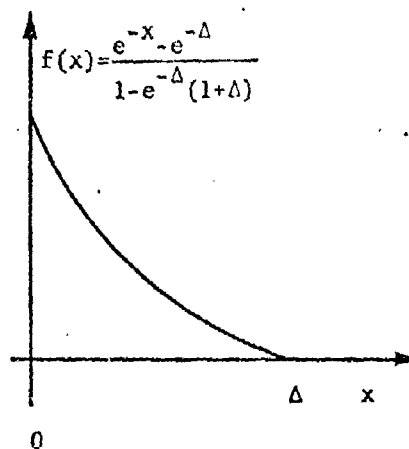


FIGURE 3.4(b)

Figure 3.4(c) shows the density of a random variable from any other such "tooth" defined by $k \cdot \Delta < z < (k+1) \cdot \Delta$.

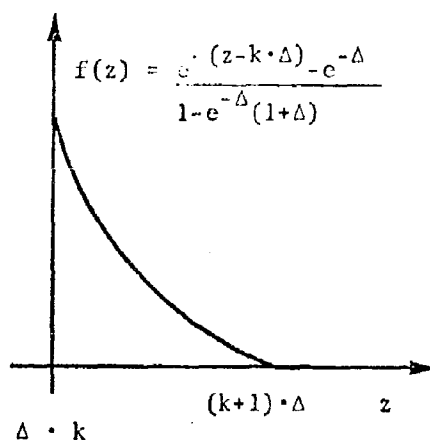


FIGURE 3.4(c)

It is obvious that all such densities are identical, a unique characteristic of the exponential distribution. Consequently, a random variable from the $(k+1)^{\text{st}}$ "tooth" is always produced by taking $z = k \cdot \Delta + x$ where the value $k \cdot \Delta$ is chosen by the technique described in section 3.1 and x is generated by taking Δ times the minimum of N uniform $(0,1)$ random numbers where N has the distribution (3.4.4). Its expected value when $\Delta = 0.1$ (16) is 2.02 or in general

$$E[N] = \Delta \cdot (e^{\Delta} - 1) / (e^{\Delta} - 1 - \Delta) \quad (3.4.7)$$

The same unique "no memory" characteristic of the exponential distribution used above is also used for obtaining random variates from g_3 , the tail of the exponential density where $t > c$.

Let $Y = W + X$ where W is discrete valued with probability function

$$P[W = w = c \cdot k] = Ae^{-w} = Ae^{-c \cdot k}, \quad k = 1, 2, 3, \dots, \quad (3.4.8)$$

$\Lambda = (1 - e^{-c})/e^{-c}$ and x is independent of w and has the distribution function (3.4.3). The distribution function of Y can be written as

$$G(y) = P[Y \leq y = w + x] = P[Y \leq w = c] + P[W = w, X \leq x] \quad (3.4.9)$$

Since the distribution function of W is

$$P[W \leq w = c \cdot k] = A \sum_{j=1}^k e^{-j \cdot c} = 1 - e^{-c \cdot k}$$

the expression for $G(y)$ becomes

$$G(y) = 1 - e^{-(w-c)} + Ae^{-w}(1 - e^{-x})/(1 - e^{-c}) = 1 - e^{c-y}, \quad (3.4.11)$$

$y > c.$

This is the required distribution function for variates from the density e^{-y} truncated on the left at $y = c$. Thus values of t from g_3 in the exponential case are generated as the sum of a discrete variate taking values $c, 2c, 3c, \dots$ and a continuous variate from the truncated exponential density on $(0, c)$.

3.5 Generating Numbers from a 'Nearly Triangular'

Density: Application to the Half Normal Density

The results of this section are applied to the half normal density

$$f(t) = \sqrt{2/\pi} e^{-1/2 t^2} \quad 0 < t < \infty \quad (3.5.1)$$

for the generation of random normal deviates to be described in chapter 4. Consider the concave and convex triangular densities (members of g_2) depicted in figures 3.5(a) and 3.5(b).

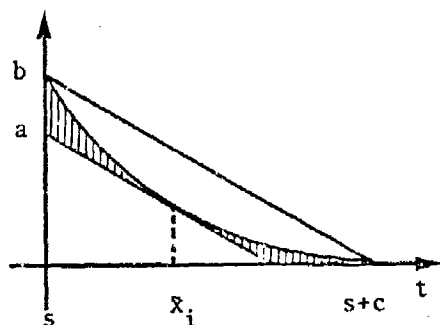


FIGURE 3.5(a)

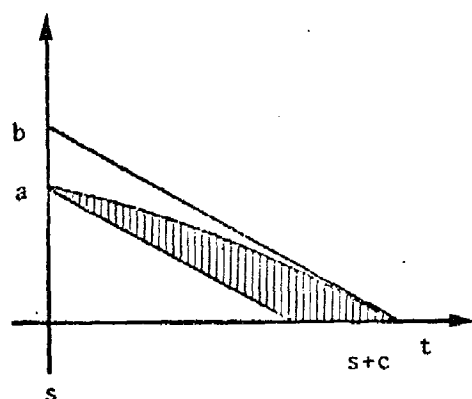


FIGURE 3.5(b)

The parallel chords and tangents enclosing $f(t)$ are determined by construction as indicated in the figures. In both cases, the inner right triangle represents most of the area under the density $F(t)$. This will be true in general as long as the ratio a/b is close to one. A random variate from the density represented by this triangle is generated as described in section 3.3. Infrequently, a random variate from the shaded area must be generated by the acceptance-rejection technique described in section 3.2. Marsaglia et al. [12] have combined these two procedures as follows.

- 1) Generate independent uniform (0,1) random variables u and v .
- 2) If $\max(u,v) \leq a/b$, put $t = s + c \cdot \min(u,v)$
- 3) If not, test $b|u-v| \leq f(s + c \cdot \min(u,v))$.

If yes, put $t = s + c \cdot \min(u,v)$.

If no, go to step 1 and try again.

In order to show that this procedure produces a variate t with the required density, we conveniently take $s = 0$ without loss of generality and proceed as follows.

Define

$$m = \min(u,v)$$

$$M = \max(u,v)$$

$$x = c \cdot m$$

$$y = b(M-m) = b(u-v) \quad (3.5.2)$$

Then the pair (x,y) is uniformly distributed over the triangle in figure 3.5(c)

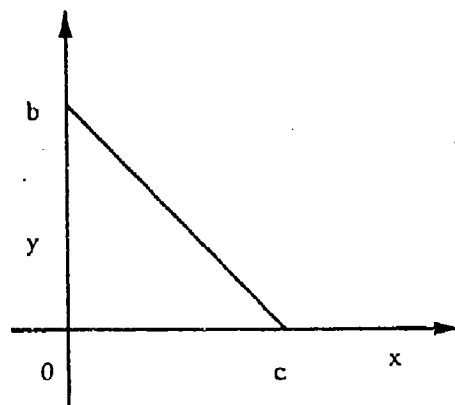


FIGURE 3.5(c)

To verify this, consider the joint distribution function of X and Y :

$$F(x,y) = P[X \leq x, Y \leq y] = P[m \leq x/c, M - m \leq y/b]$$

This is just the area of the cross hatched region in figure 3.5(d) which is seen to be

$$F(x,y) = 2 \cdot (y/b) \cdot (x/c) = 2xy/bc, \quad 0 < x < c, \quad (3.5.3) \\ 0 < y < b.$$

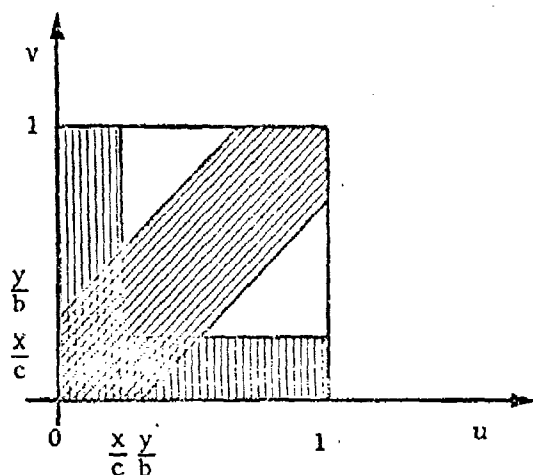


FIGURE 3.5(d)

Thus, (x,y) is a point randomly chosen from the larger triangle in figures 3.5(a) and 3.5(b). By the acceptance-rejection technique, if this point falls within the smaller triangle of figures 3.5(a) and 3.5(b), the value $x = t$ has the triangular density $-bx/c + a$. Now the condition $\max(u,v) \leq a/b$ is equivalent to:

$$bM \leq a$$

$$b(M-m) \leq a - bm = -bx/c + a$$

or

$$y \leq -bx/c + a \quad (3.5.4)$$

which means that (x,y) falls in the small triangle. If this condition is not satisfied, a second check is made to determine whether the point (x,y) falls within the shaded area between $f(t)$ and y_2 in figures 3.5(a) and 3.5(b). If so, then clearly

$$y = b(M - m) = b|u - v| \leq f(x) = f(c \cdot \min(u,v)) \quad (3.5.5)$$

as required by the second test. The acceptance-rejection principle insures that every x value passing at least one of these two tests will have the density $f(t)$. The first test is rapidly made; the second test involves evaluation of $f(x)$ and is time consuming. The first test will be passed with a probability

$$P[M \leq a/b] = (a/b)^2$$

as given by the distribution function of $M = \max(u,v)$. It is therefore desirable to have $(a/b)^2$ close to 1 to achieve a short average execution time. This will be the case if $f(t)$ is "nearly triangular."

For the half normal density the a_i/b_i ratios are obtained as follows.

Case (a) "Concave Triangles" (See figure 3.5(a)).

Let $t_i = i \cdot c = s$

$$t_{i+1} = (i+1)c = s + c, \quad i = 0, 1, 2, \dots \quad (3.5.6)$$

The equation of the chord and the tangent are respectively:

$$\begin{aligned} y_1 &= -b_i(x-s)/c + a_i \\ y_2 &= -b_i(x-s)/c + a_i \end{aligned} \quad (3.5.7)$$

The value of b_i is clearly

$$b_i = f(t_i) - f(t_{i+1}) = f(s) - f(s+c) \quad (3.5.8)$$

The slope of both lines, $-b_i/c$, must equal $f'(\bar{x}_i)$. Thus

$$-b_i/c = -\sqrt{2/\pi} \bar{x}_i e^{-\frac{1}{2}\bar{x}_i^2} = -\bar{x}_i f(\bar{x}_i) \quad (3.5.9)$$

This equation must be solved iteratively for $\bar{x}_i$, the abscissa of the point of tangency. Finally, this value is substituted into the tangent's equation to give

$$f(\bar{x}_i) = f(t_{i+1}) = (-b_i/c)(\bar{x}_i-s) + a_i. \quad (3.5.10)$$

Substituting the expression for b_i/c and solving for a_i gives

$$a_i = f(\bar{x}_i)[1 + \bar{x}_i(\bar{x}_i-s)] - f(s+c). \quad (3.5.11)$$

Consequently for "concave triangles"

$$a_i/b_i = [f(\bar{x}_i)\{1 + \bar{x}_i(\bar{x}_i-s)\} - f(s+c)]/[f(s) - f(s+c)] \quad (3.5.12)$$

Case (b) "Convex Triangles" (See figure 3.5(b))

In this case the point of tangency occurs at the end of the interval where $\bar{x}_i = t_{i+1} = s + c$. The slope of both lines is given by

$$-b_i/c = f'(s+c)$$

or
$$b = -cf'(s+c). \quad (3.5.13)$$

The value of a_i is simply

$$f(t_i) - f(t_{i+1}) = f(s) - f(s+c).$$

Consequently, for "convex triangles"

$$a_i/b_i = [f(s) - f(s+c)]/[-cf'(s+c)] \quad (3.5.14)$$

3.6 Generation of Random Variates from the Tail of the Half Normal Distribution

The procedure suggested by Marsaglia et al. [12] for the generation of random variates from the tail of the half normal distribution is based on the rejection principle described in section 3.2. The procedure is to generate a pair of independent half normal variates x_1 and x_2 such that the point (x_1, x_2) lies outside the quarter circle (see figure 3.6(a)).

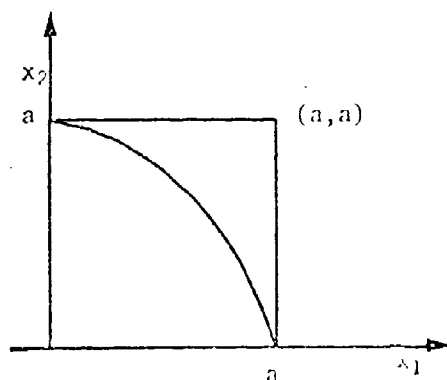


FIGURE 3.6(a)

This is done by generating pairs of uniform (0,1) random numbers u_1 and u_2 until $0 < u_1^2 + u_2^2 \leq 1$ and setting

$$x_1 = \sqrt{a^2 - 2\ln(u_1^2 + u_2^2)} \cdot \sqrt{u_1^2 / (u_1^2 + u_2^2)} \quad (3.6.1)$$

$$x_2 = \sqrt{a^2 - 2\ln(u_1^2 + u_2^2)} \cdot \sqrt{u_2^2 / (u_1^2 + u_2^2)} \quad (3.6.2)$$

To show that the point (x_1, x_2) lies outside the quarter-circle, let $w = -\ln(u_1^2 + u_2^2)$ where $0 < u_1^2 + u_2^2 \leq 1$. Then take the sum $x_1^2 + x_2^2 = a^2 + 2w$ where $0 \leq w < \infty$ so that $x_1^2 + x_2^2 \geq a^2$. If the point (x_1, x_2) , generated in this fashion, lies within the square (figure 3.6(a)), it is rejected and a new pair of uniform (0,1) random numbers is generated and tested. Otherwise the variable, x_1 or x_2 , whose value exceeds a is taken as the required random number.

In order to show that this procedure produces the desired result, let u_1 and u_2 be a pair of uniform (0,1) random numbers conditioned by $0 < u_1^2 + u_2^2 \leq 1$. Then the joint density of u_1 and u_2 is

$$f(u_1, u_2) = 4/\pi, \quad 0 < u_1 < 1, \quad 0 < u_2 < 1, \quad 0 < u_1^2 + u_2^2 \leq 1. \quad (3.6.3)$$

Now define:

$$X = u_1^2 + u_2^2, \quad 0 < X < 1, \text{ and} \quad (3.6.4)$$

$$Y = u_2/u_1, \quad 0 < Y < \infty. \quad (3.6.5)$$

Then the probability distribution of X is

$$F(x) = P(X \leq x) = (\pi \cdot x/4) / (\pi \cdot 1^2/4) = x \quad (3.6.6)$$

and the density of x is

$$f(x) = 1, \quad 0 < x < 1. \quad (3.6.7)$$

Consequently X is uniform on $(0,1)$. The joint density of X and Y is:

$$g(x,y) = f(u,v) \cdot |J| = (4/\pi) [1/2(1+y^2)] = 2/\pi(1+y^2) = g_1(x) \cdot g_2(y). \quad (3.6.8)$$

Hence X and Y are stochastically independent. Substitution of (3.6.4) and (3.6.5) into (3.6.1) and (3.6.2) yields:

$$x_1 = \sqrt{a^2 - 2\ln X} \cdot \sqrt{1/(1+Y^2)} \quad (3.6.9)$$

$$x_2 = \sqrt{a^2 - 2\ln X} \cdot \sqrt{Y^2/(1+Y^2)}. \quad (3.6.10)$$

The inverse transformation is:

$$X = e^{-\frac{1}{2}(X_1^2 + X_2^2 - a^2)} \quad (3.6.11)$$

$$Y = X_2/X_1 \quad (3.6.12)$$

The joint density of X_1 and X_2 is given by:

$$\begin{aligned} f(x_1, x_2) &= g(x, y) \cdot |J| \\ &= [2/\pi(1+y^2)](1+y^2) \cdot e^{-\frac{1}{2}(x_1^2 + x_2^2 - a^2)} \\ &= (\sqrt{2/\pi} e^{-\frac{1}{2}x_1^2}) (\sqrt{2/\pi} e^{-\frac{1}{2}x_2^2}) e^{a^2/2} \end{aligned} \quad (3.6.13)$$

$$a^2 \leq x_1^2 + x_2^2 < \infty.$$

Clearly x_1 and x_2 are independent random variables from the tail of the half normal distribution ($x_1, x_2 > a$).

CHAPTER 4
GENERATION OF PSEUDO-RANDOM NUMBERS FROM THE
EXPONENTIAL, GAMMA, AND NORMAL DISTRIBUTIONS

Before presenting the procedures for the generation of random numbers from the exponential, gamma, and normal distributions, a simple example is given demonstrating the use of the techniques discussed in Chapter 3 for obtaining random variates from any continuous distribution defined on a finite interval. In the interest of clarity all numbers are expressed to the base 10. Consider the distribution defined by

$$F(y) = (y/2)(3-y^2) \quad 0 < y < 1 \quad (4.0.1)$$

with density

$$f(y) = (3/2)(1-y^2) \quad 0 < y < 1. \quad (4.0.2)$$

Using the composition technique presented in 3.1, $f(y)$ is represented as a mixture of 2 densities (since there is no tail):

$$f(y) = p_1 g_1(y) + p_2 g_2(y). \quad (4.0.3)$$

As before, g_1 is a series of rectangles and g_2 represents the nearly triangular regions between g_1 and f . See figures 4.0(a), (b), and (c).

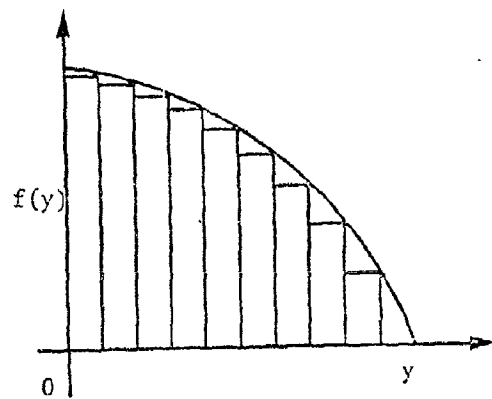


FIGURE 4.0(a)

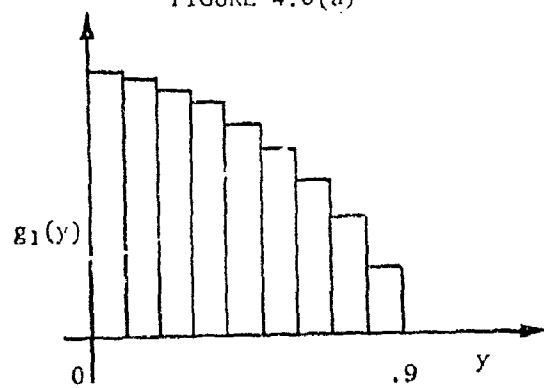


FIGURE 4.0(b)

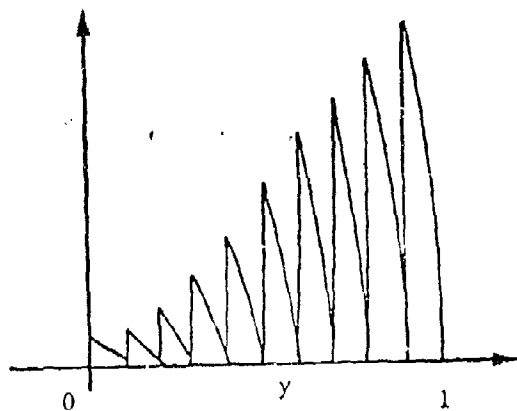


FIGURE 4.0(c)

As described in 3.1, define discrete variables

$$x_i = (i-1) \cdot \Delta \text{ for } i=1,2,\dots,10 \text{ and in this example } \Delta=0.1_{(10)}. \quad (4.0.4)$$

Now assign each x_i a probability P_i as follows:

$$P_i = \Delta \cdot f(x_{i+1}). \quad (4.0.5)$$

Now define a value T_i as given in (3.1.7), the area of the triangular region above the i^{th} rectangle:

$$T_i = F(x_{i+1}) - F(x_i) - P_i. \quad (4.0.6)$$

These values are recorded in table 4.0.1. At this point it is obvious that the values of p_1 and p_2 , defined in (4.0.3), are $p_1 = 0.9225$ and $p_2 = 0.0775$. Next, the four sets of the discrete values are stored, based on the high order of digits of the P_i 's.

x_i	$P_i = 0.1 \cdot f(x_{i+1})$	$F(x_{i+1}) - F(x_i)$	$T_i = F(x_{i+1}) - F(x_i) - P_i$
0.0	0.1485	0.1495	0.0010
0.1	0.1440	0.1465	0.0025
0.2	0.1365	0.1405	0.0040
0.3	0.1260	0.1315	0.0055
0.4	0.1125	0.1195	0.0070
0.5	0.0960	0.1045	0.0085
0.6	0.0765	0.0865	0.0100
0.7	0.0540	0.0655	0.0115
0.8	0.0285	0.0415	0.0130
0.9	0.0000	0.0145	0.0145
Total	0.9225	1.0	0.0775

TABLE 4.0.1

As presented in Chapter 2, define:

$$Q_0 = 0, \quad Q_r = 10^{-r} \cdot \sum_{i=1}^{10} \delta_{ri} \quad (4.0.7)$$

and

$$\Pi_0 = 0, \quad \Pi_r = \sum_{j=1}^r \sum_{i=1}^{10} \delta_{ji} \quad \text{for } r=1,2,3,4. \quad (4.0.8)$$

Using the above definitions define:

$$S_0 = 0, \quad S_k = 10^k \sum_{r=1}^k Q_r \quad (4.0.9)$$

and

$$N_k = \Pi_{k-1} - 10 \cdot S_{k-1} \quad \text{for } k=1,2,3,4. \quad (4.0.10)$$

The N_k 's are identical to the quantity

$$\Pi_{s-1} + \beta^s \cdot \sum_{r=0}^s P_r$$

in (2.0.3). Because of overlapping between the four sets, the N_k 's must be adjusted. Their values and the values of the S_k 's are recorded in table 4.0.3.

S-TABLE		N-TABLE	
Loc	Con	Loc	Con
0	5	0	2
1	87	1	-46
2	920	2	-852
3	9225	3	-9154

Table 4.0.3

The procedure for generating random numbers from g_1 is:

- 1) Generate a uniform (0,1) random number $u = 0.d_1d_2\dots$
- 2) If $d_1 < S(1)$, set $Y = A(d_1 + N(1)) + 0.1 \cdot (.d_5d_6\dots)$;
otherwise go to (3).
- 3) If $d_1d_2 < S(2)$, set $Y = A(d_1d_2 + N(2)) + 0.1 \cdot (.d_5d_6\dots)$;
otherwise go to (4).
- 4) If $d_1d_2d_3 < S(3)$, set $Y = A(d_1d_2d_3 + N(3)) + 0.1 \cdot (.d_5d_6\dots)$;
otherwise go to (5).
- 5) If $d_1d_2d_3d_4 < S(4)$, set $Y = A(d_1d_2d_3d_4 + N(4)) + 0.1 \cdot (.d_5d_6\dots)$;
otherwise generate a random variate from g_2 .

In order to generate random variates from g_2 , define:

$$C(1) = S_4 + T[A(1)] \text{ and} \quad (4.0.11)$$

$$C(k+1) = C(k) + T[A(k)] \text{ for } k=1,2,\dots,9. \quad (4.0.12)$$

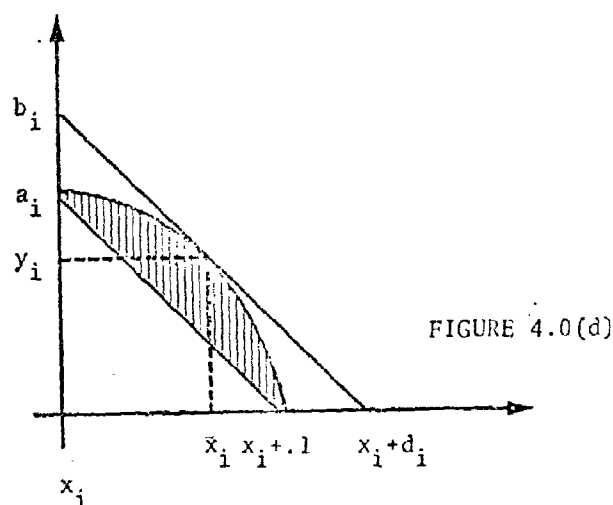
This is identical to the definition given in (3.1.13), except that in this example there is no rectangular residue. The notation $T[A(k)]$ denotes the area of the triangular region (the T_i 's in table 4.0.1) corresponding to the discrete value stored in the k^{th} location of the A-table (Table 4.0.2). The values of the $C(k)$'s are given in table 4.0.4. If u is a uniform (0,1) random number such that $u \geq S_4$, the proper triangle of g_2 may be chosen simply by testing:

$$u < C(k) \text{ until the condition is satisfied (note that } C_{(10)} = 1.0)$$

The discrete value occupying location k of the A-table denotes the correct triangle.

Having selected the proper triangle of g_2 , a random variate having the density of that triangle is to be generated. This is accomplished

through the use of the acceptance-rejection principle discussed in 3.2 and the technique given in 3.5. Consider a particular triangle or tooth from g_2 (figure 4.0(d)). The tangent and the chord are constructed as indicated in 3.5



The inner right triangle encloses most of the area under $f(x)$. A random variate from the density of this triangle is generated as described in 3.3. Occasionally, a random variate from the shaded region must be generated by the acceptance-rejection technique. The values of a_i , b_i , and d_i are obtained as follows:

$$-(a_i/0.1) = f'(x_i) = 3x_i \quad (4.0.13)$$

where x_i is the abscissa of the point of tangency and

$$a_i = f(x_i) - f(x_{i+1}). \quad (4.0.14)$$

Substituting (4.0.14) into (4.0.13) and solving for $\bar{x}_i$ yields:

$$\bar{x}_i = [f(x_i) - f(x_{i+1})]/0.3 \quad (4.0.15)$$

The ordinate of the point of tangency is:

$$y_i = f(\bar{x}_i) - f(x_{i+1}). \quad (4.0.16)$$

At $x = \bar{x}_i$, $y_1 = y_i$ and

$$b_i = y_i + (a_i/0.1)(\bar{x}_i - x_i) \quad (4.0.17)$$

The value of d_i may be obtained by noting that:

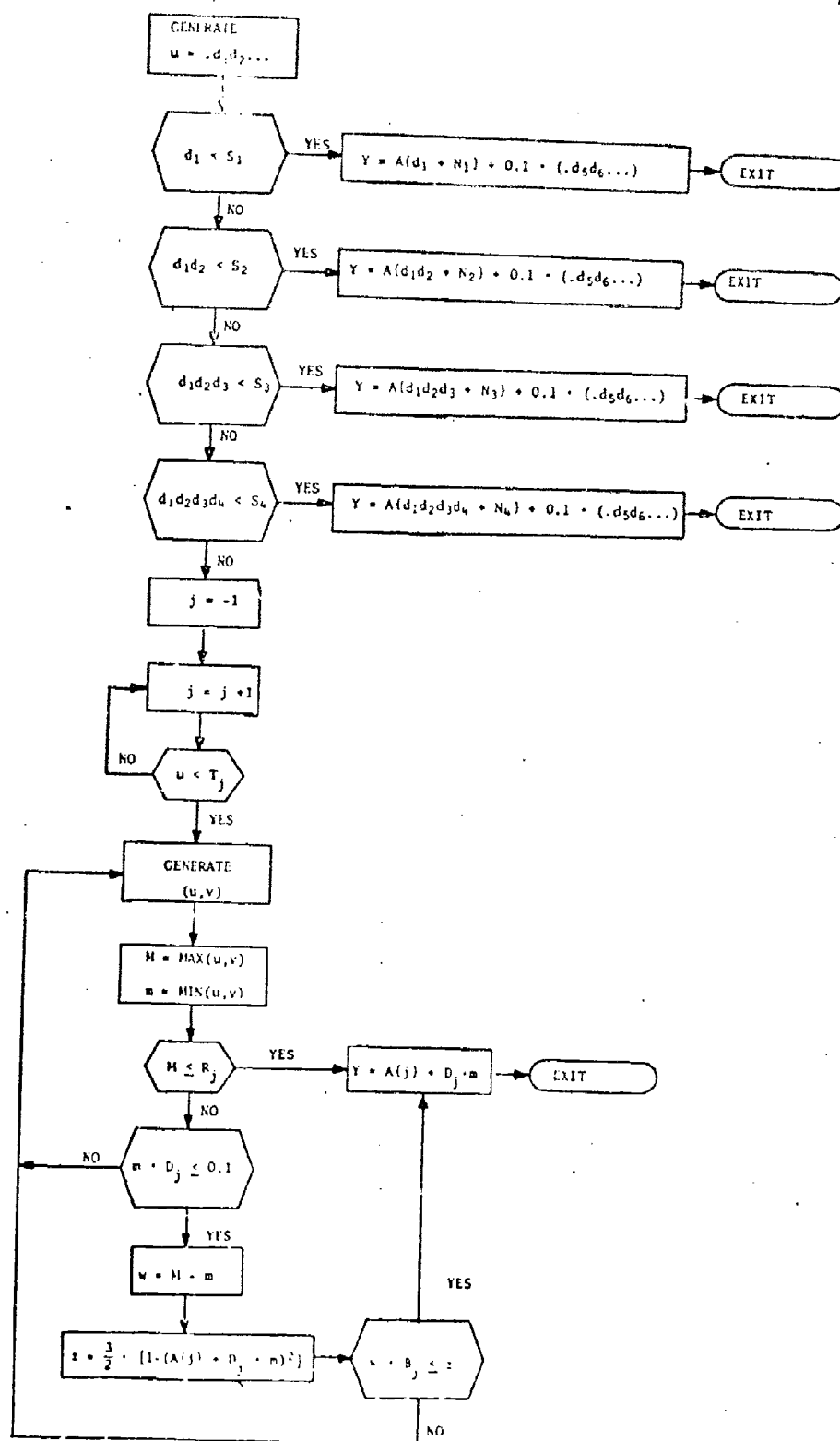
$$(a_i/0.1) = (b_i/d_i) \text{ and hence } d_i = (0.1 \cdot b_i/a_i). \quad (4.0.18)$$

The values of a_i/b_i , d_i , and b_i are recorded in table 4.0.4 in the R-table, D-table, and B-table respectively.

C-Table		R-Table		D-Table		R-Table	
Loc	Con	Loc	Con	Loc	Con	Loc	Con
0	.9370	0	.9870	0	.1013	0	.28875
1	.9500	1	.9855	1	.1015	1	.25875
2	.9555	2	.9655	2	.1036	2	.10875
3	.9580	3	.9231	3	.1083	3	.04875
4	.9650	4	.9730	4	.1028	4	.13875
5	.9690	5	.9524	5	.1050	5	.07875
6	.9700	6	.8000	6	.1250	6	.01875
7	.9815	7	.9836	7	.1017	7	.22875
8	.9915	8	.9811	8	.1019	8	.19875
9	1.0000	9	.9778	9	.1023	9	.16875

Table 4.0.4

FIGURE 4.0(e)
Flowchart for the Generation of Random
Variates from
 $f(y) = (3/2)(1 - y^2), 0 < y < 1$



The procedure for obtaining random variates from g_2 is:

- 1) Let u be a uniform (0,1) random number such that $u \geq S_4$.
- 2) Test $u < C(k)$ for $k = 0, 1, \dots, 9$ until the inequality is satisfied.
- 3) Generate independent uniform (0,1) random numbers u_1 and u_2 .
- 4) If $\max(u_1, u_2) \leq R(k)$, then set $Y = A(k) + D(k) \cdot \min(u_1, u_2)$; otherwise go to (5).
- 5) If $B(k) \cdot |u_1 - u_2| \leq f[A(k) + D(k) \cdot \min(u_1, u_2)]$, set $Y = A(k) + D(k) \cdot \min(u_1, u_2)$; otherwise go to step (3) and try again.

A schematic of the entire procedure for generating random variates from the density $f(y)$, given in (4.0.2), is presented in figure 4.0(e).

4.1 Generation of Exponentially Distributed Random Numbers

Applying the composition principle presented in 3.1, the exponential density function,

$$f(t) = e^{-t}, \quad 0 \leq t \leq \infty, \quad (4.1.1)$$

is represented as a mixture of three densities

$$f(t) = p_1 g_1(t) + p_2 g_2(t) + p_3 g_3(t). \quad (4.1.2)$$

As before, g_1 is a series of rectangles defined for $t \leq 4$, g_2 represents the toothlike region between g_1 and f defined for $t \leq 4$, and g_3 is the tail of f for $t \geq 4$. The probabilities p_1 , p_2 , and p_3 are respec-

tively the areas represented by g_1 , g_2 , and g_3 .

As described in 3.1 a random variable Y having density g_1 is generated as follows:

Define

$M = 4/\Delta$, where $\Delta = 10^{-1}$ for a decimal machine or $\Delta = 16^{-1}$ for a binary machine; and discrete variates

(4.1.3)

$t_i = (i-1) \cdot \Delta$ for $i = 1, 2, \dots, M$.

(4.1.4)

Each t_i is assigned a probability

$$P_i = \Delta \cdot f(t_{i+1}) = \Delta \cdot e^{-t_{i+1}} = \delta_{1i} \delta_{2i} \delta_{3i} \dots \quad (4.1.5)$$

Now, using the high order 4 octal digits (binary machine) or the high order 3 decimal digits (decimal machine), store the t_i 's according to the technique described in Chapter 2. As in the case of the example given in the beginning of this chapter, the sets of discrete variates are stored in the manner that permits maximum overlap. See the D-table in the program listing of GEN1 in the appendix; set 1 occupies locations 49A thru 4BE, set 2 occupies locations 40C through 4AC, and set 3 occupies locations 37D through 453. Locations 374 through 3B5 are used for the generation of variates from g_2 and the residual of g_1 as described in 3.1.

At this point define:

$$Q_0 = 0, \quad Q_r = e^{-r} \cdot \sum_{i=1}^M \delta_{ri} \quad (4.1.6)$$

$$H_0 = 0, \quad H = \sum_{j=1}^r \sum_{i=1}^M \delta_{ji} \quad \text{for } r = 1, 2, 3, 4 \quad (4.1.7)$$

for binary machines where the δ 's are octal digits and $\theta = 8$ or $r = 1, 2, 3$

for decimal machines where the δ 's are decimal digits and $\beta = 10$. Also define:

$$S_0 = 0, S_k = \beta^k \cdot \sum_{r=1}^k Q_r \quad (4.1.8)$$

and

$$N_k = \Pi_{k-1} - \beta \cdot S_{k-1} \text{ for } k = 1, 2, 3, 4 \quad (4.1.9)$$

in the binary case ($\beta = 2$) or $k = 1, 2, 3$ in the decimal case ($\beta = 10$).

The S-values and the N-values are recorded in the program listing of GEN1 in the appendix. Note that the N-values have been adjusted to allow for overlap between the sets of the D-table.

The procedure for the generation of random variates from this truncated portion of g_1 is: (binary case)

- 1) Generate a uniform (0,1) random number $u = .d_1 d_2 \dots$
- 2) If $d_1 d_2 < S_2$, set $Y = D\{d_1 d_2 + N_2\} + \Delta \cdot (.d_5 d_6 \dots)$;
otherwise go to (3).
- 3) If $d_1 d_2 d_3 < S_3$, set $Y = D\{d_1 d_2 d_3 + N_3\} + \Delta \cdot (.d_5 d_6 \dots)$;
otherwise go to (4).
- 4) If $d_1 d_2 d_3 d_4 < S_4$, set $Y = D\{d_1 d_2 d_3 d_4 + N_4\} + \Delta \cdot (.d_5 d_6 \dots)$;
otherwise generate a random variate from the residual of g_1 , g_2 ,
or g_3 .

Note that in the above procedure the testing begins with S_2 . This is because $\delta_{1i} = 0$ for all i since $0 < f(x) < 1$ for $0 < x < 4$.

Random variates from the residual of g_1 and from g_2 are generated by first selecting one of the rectangles of g_1 with appropriate probability and then testing to determine whether the required variate should

come from the tooth or from the residual corresponding to that rectangle.

This is accomplished as follows:

In M consecutive locations store each of the t_i 's once (locations 374 through 383 of the D-table in the listing of GEN1 in the appendix). For each t_i define a pair of values as given in (3.1.12), (3.1.13), and (3.1.14)

$$C(1) = S_0 + R[D(1)], \quad (4.1.10)$$

$$B(k) = C(k) + T[D(k)] \text{ and} \quad (4.1.11)$$

$$C(k+1) = C(k) + R[D(k+1)]. \quad (4.1.12)$$

In the exponential case

$$T[D(k)] = e^{-D(k)} - (1 + \Delta) e^{-(D(k)+\Delta)} \quad (4.1.13)$$

and

$$R[D(k)] = P[D(k)] - .\delta[1,D(k)]\delta[2,D(k)]\delta[3,D(k)]\delta[4,D(k)], \quad (4.1.14)$$

where $P[D(k)]$ is as defined in (4.1.5). The values $T[D(k)]$ and $R[D(k)]$ denote respectively the areas of the triangular region and rectangular residue lost upon truncation of $P[D(k)]$ corresponding to the t -value occupying location $D(k)$. The values of the $B(k)$'s and the $C(k)$'s are also recorded in the listing of GEN1 in the appendix. A random variate with a rectangular distribution is generated as described in 3.1, and in the exponential case a variate having the density of one of the teeth of g_2 is generated by the technique given in 3.4. The procedure for generating random variates from either the residual of g_1 or from g_2 is summarized as follows:

- 1) Let u be a uniform $(0,1)$ random number such that
 $S_4 \leq u < p_1 + p_2 = 1 - e^{-4}$.
- 2) Test $u < B(J)$ for $J = 1, 2, \dots, \overline{M}$, until the inequality is satisfied (note $B(\overline{M}) = 1 - e^{-4}$) for some $J = J'$.
- 3) If $u < C(J')$, generate a new uniform $(0,1)$ random number v and set $Y = D(J') + \Delta \cdot v$ (this produces a number from g_1); otherwise go to (4).
- 4) Generate a new uniform $(0,1)$ random number v and test
 $v < Q(n) = \sum_{j=2}^n \Delta^j / [j! (e^{\Delta} - 1 - \Delta)]$ for $n = 2, 3, \dots$
until the inequality is satisfied for $n = n'$.
- 5) Generate n' independent uniform $(0,1)$ random numbers and set
 $Y = D(J') + \Delta \cdot \min(u_1, u_2, \dots, u_{n'})$. This produces a number from g_2 .

A random variate Y having density g_3 is generated as a sum

$$Y = W + X \quad (4.1.15)$$

where X is exponential on $(0,4)$ and W has the distribution

$$P[W = w = 4 \cdot k] = e^{-4k} (e^4 - 1), \quad k = 1, 2, \dots \quad (4.1.16)$$

The procedure for generating a random variate from g_3 is summarized as follows:

- 1) Let u be a uniform $(0,1)$ random number such that $u \geq p_1 + p_2 = 1 - e^{-4}$.
- 2) Test $u < 1 - e^{-4(k+1)}$ for $k = 1, 2, \dots$ until the inequality is satisfied for some $k = k'$.

- 3) Set $W = 4 \cdot k'$ and generate a new uniform (0,1) random number v .
- 4) Set $u = (p_1 + p_2) \cdot v = (1 - e^{-4}) \cdot v$ and enter the procedure for generating exponentially distributed random variates on the interval (0,4) with $u = (1 - e^{-4}) \cdot v$ as the initial uniform number and generate X .
- 5) Set $Y = W + X$.

In [13] Marsaglia presents a schematic and the necessary constants for the generation of random variates from the exponential distribution. Application of this method results in a very fast generator program. As is evident from the discussion herein the method is also exact; the accuracy of the result is dependent only on the word size of the computer.

4.2 Generation of Gamma Distributed Random Variates

In order to produce random variates having the gamma distribution, the following result is used.

If x_i is a random variable with density

$$f(x_i) = e^{-x_i}, \quad 0 < x_i < \infty \quad \text{for } i = 1, 2, \dots, n \quad (4.2.1)$$

then the random variable

$$Y = \sum_{i=1}^n x_i \quad (4.2.2)$$

has the gamma density

$$g(y) = (1/\Gamma(n)) y^{n-1} e^{-y}, \quad 0 < y < \infty \quad (4.2.3)$$

when the x_i 's are stochastically independent. Consequently, the generation of random variates from the gamma distribution involves only a simple

modification of the exponential method.

- 1) Using the method given in 4.1, generate n exponentially distributed random numbers $x_1, x_2, \dots, x_n$.
- 2) Set $Y = \sum_{i=1}^n x_i$ (if a scale parameter $B \neq 1$ is desired, set $Y = B \cdot \sum_{i=1}^n x_i$).

4.3 Generation of Random Variates having the Standard Normal Distribution

Using the composition principle of 3.1, the absolute normal density

$$f(t) = \sqrt{2/\pi} e^{-1/2 t^2}, \quad t \geq 0 \quad (4.3.1)$$

is represented as a mixture of three densities

$$f(t) = p_1 g_1(t) + p_2 g_2(t) + p_3 g_3(t). \quad (4.3.2)$$

Again, g_1 is a series of rectangles defined for $t \leq 3$, g_2 represents the toothlike regions above g_1 and below f defined for $t \leq 3$, and g_3 is the tail of the density (4.3.1) defined for $t \geq 3$.

Random variates from g_1 are generated by means of the technique given in 3.1. Define

$$\begin{aligned} M &= 3/\Delta \quad \text{where } \Delta = 10^{-1} \text{ (decimal machine)} \\ \text{or } \Delta &= 16^{-1} \text{ (binary machine), and} \end{aligned} \quad (4.3.3)$$

discrete variables

$$t_i = (i - 1) \cdot \Delta \quad \text{for } i = 1, 2, \dots, M. \quad (4.3.4)$$

Assign each t_i a probability

$$P_i = \Delta \cdot f(t_{i+1}) = \Delta \cdot \sqrt{2/\pi} \cdot e^{-\frac{1}{2}t_{i+1}^2} = .t_{1i}t_{2i}\dots \quad (4.3.5)$$

Applying the storage technique described in Chapter 2, store the t_i 's based on the high order 4 octal digits (binary machine) or the high order 3 decimal digits. Note that as in the exponential case $t_{1i} = 0$ for all i , and that the sets of discrete values are overlapped (see the A-table in the listing of GEN3 in the appendix). Now define values

$$Q_0 = 0, \quad Q_r = \beta^{-r} \sum_{i=1}^M \delta_{ri} \quad (4.3.6)$$

and

$$\Pi_0 = 0, \quad \Pi_r = \sum_{j=1}^r \sum_{i=1}^M \delta_{ji}; \quad (4.3.7)$$

also define

$$S_0 = 0, \quad S_k = \beta^k \cdot \sum_{r=1}^k Q_r \quad (4.3.8)$$

and

$$N_i = \Pi_{k-1} - \beta \cdot S_{k-1}, \quad (4.3.9)$$

for a binary machine $r, k = 1, 2, 3, 4$ and $\beta = 8$ for a decimal machine, $r, k = 1, 2, 3$ and $\beta = 10$. The S-values and N-values are also recorded in the program listing of GEN3 (note that the N-values have been adjusted to allow for overlap between sets).

The procedure for the generation of random variates from this truncated portion of g_1 is: (binary case)

- 1) Generate a uniform (0,1) random number $u = .d_1d_2\dots$
- 2) If $d_1d_2 < S_2$, set $Y = A\{d_1d_2 + N_2\} + \Delta \cdot (.d_5d_6\dots)$;
otherwise go to (3).

- 3) If $d_1 d_2 d_3 < S_3$, set $Y = A\{d_1 d_2 d_3 + N_3\} + \Delta \cdot (.d_5 d_6 \dots)$;
otherwise go to (4).
- 4) If $d_1 d_2 d_3 d_4 < S_4$, set $Y = A\{d_1 d_2 d_3 d_4 + N_4\} + \Delta \cdot (.d_5 d_6 \dots)$;
otherwise generate a random variable from g_2 , g_3 , or the
residual of g_1 .

Random variates from the residual of g_1 and from g_2 are produced by first choosing one of the rectangles of g_1 with appropriate probability and then testing to determine whether the required variate should come from the rectangular residue of g_1 or from the toothlike region of g_2 . As presented in 3.1, this is accomplished as follows:

In M consecutive locations store each t_i once, and for each t_i define a pair of values as in (3.1.12), (3.1.13), and (3.1.14):

$$C(1) = S_4 + R[A(1)], \quad (4.3.10)$$

$$B(k) = C(k) + T[A(k)], \quad (4.3.11)$$

$$C(k+1) = C(k) + R[A(k+1)]. \quad (4.3.12)$$

Note that in the listing of GEN3 this portion of the A-table occupies locations 690 thru 6BF. However, for convenience it is assumed that location 690 correspond to $A(1)$. $R[A(k)]$ and $T[A(k)]$ denote respectively the area of the rectangular remnant and the area of the toothlike region corresponding to the t -value occupying location $A(k)$. For the normal case:

$$R[A(k)] = P[A(k)] - .\delta[1,A(k)]\delta[2,A(k)]\delta[3,A(k)]\delta[4,A(k)] \quad (4.3.13)$$

where $P[A(k)]$ is defined in (4.3.5) as

$$\begin{aligned} P[A(k)] &= \Delta \cdot f(A(k) + \Delta) = \Delta \cdot \sqrt{2/\pi} \cdot e^{-\frac{1}{2}(A(k)+\Delta)^2} \\ &= .\delta[1, A(k)]\delta[2, A(k)]\dots, \end{aligned} \quad (4.3.14)$$

and

$$T[A(k)] = F(A(k) + \Delta) - F(A(k)) - P[A(k)] \quad (4.3.15)$$

where

$$F(x) = \sqrt{2/\pi} \int_0^x e^{-\frac{1}{2}t^2} dt \quad (4.3.16)$$

Once having determined whether a variate with a rectangular density (g_1) or a variate with a toothlike density (g_2) is required, the schemes given in 3.1 and in 3.5 are used to produce the variate as required. This procedure is summarized as follows:

- 1) Let u be a uniform (0,1) random number such that $S_4 \leq u < p_1 + p_2 = F(3)$.
- 2) Test $u < B(k)$ for $k = 1, 2, \dots, M$ until the inequality is satisfied for some $k = k'$.
- 3) If $u < C(k')$, then generate a new uniform (0,1) random number v and put $Y = A(k') + \Delta \cdot v$ (this produces a variate from g_1); otherwise go to (4).
- 4) Generate independent uniform (0,1) random numbers u_1 and u_2 .
- 5) If $\max(u_1, u_2) \leq D(k')^*$, set $Y = A(k') + \Delta \cdot \min(u_1, u_2)$. This produces a variate from g_2 as described in 3.5; otherwise go to (6).
- 6) Set $W = -\frac{1}{2}(\Delta \cdot \min(u_1, u_2) - \Delta) \cdot [2 \cdot A(k') + \Delta \cdot \min(u_1, u_2) + \Delta]$

* The $D(k)$'s are tabled values of the a_i/b_i ratios given in (3.5.12) and in (3.5.14) corresponding to the t_i stored in $A(k')$.

and test $|u_1 - u_2| < E(k') \cdot (e^W - 1)^{**}$. If yes, set

$Y = A(k') + \Delta \cdot \min(u_1, u_2)$. If no go to step (4) and try again.

A random variate from the density g_3 is produced by the technique presented in 3.6. An absolute normal variable $|Y|$, conditioned by $|Y| > 3$, is required. The procedure is summarized as follows:

- 1) Generate pairs of uniform (0,1) random numbers u_1 and u_2 until the condition $u_1^2 + u_2^2 \leq 1$ is satisfied.
- 2) Form pairs x_1 and x_2 as described in (3.6.1) and (3.6.2)

$$x_1 = u_1 \cdot [(9 - 2\ln(u_1^2 + u_2^2))/(u_1^2 + u_2^2)]^{1/2}$$

$$x_2 = u_2 \cdot [(9 - 2\ln(u_1^2 + u_2^2))/(u_1^2 + u_2^2)]^{1/2}$$

- 3) Test $x_1 > 3$. If yes, set $Y = x_1$. If no, test $x_2 > 3$. If yes, put $Y = x_2$. If no, go to step (1) and repeat the procedure.

In [12] Marsaglia presents a schematic and the required constants for the generation of random variates from the absolute normal distribution. This method results in random variates having the density (4.3.1). In order to obtain random variates from the standard normal distribution.

$$f(t) = (1/\sqrt{2\pi}) e^{-1/2 t^2}, \quad -\infty < t < \infty, \quad (4.3.17)$$

a random + or - sign must be attached at some point in the procedure. In GEN3 this is done by generating a uniform (0,1) random number u and testing $u < 1/2$. If yes, a - sign is affixed to the variate; otherwise a positive variate is returned.

The procedures presented in this chapter derive their speed from the

** This expression is simply an alternative way of testing $b_1|u-v| \leq f(A(k') + \Delta \cdot \min(u_1, u_2))$.

fact that the vast majority of variates generated come from the truncated portion of g_1 . It requires only the generation of a uniform (0,1) random number, a series of compare instructions, and a table lookup. While the generation of variates from g_2 and g_3 may be somewhat time consuming, such variates are required only rarely. • Consequently, that average time per generation is quite fast. As is evident these methods are exact. Precision is limited only by the word size of the computer.

CHAPTER 5
RESULTS OF TESTS PERFORMED ON
SEQUENCES OF PSEUDO-RANDOM NUMBERS

5.1 Random Numbers From The
Uniform (0,1) Distribution

The tests performed on sequences of independent uniform (0,1) random numbers were suggested by MacLaren and Marsaglia in [9]. The stringency of these tests is justified in that the procedures for the generation of random variates from other distributions depend heavily upon the method used to obtain uniform (0,1) random numbers.

The tests made were chi-square (χ^2) goodness of fit tests on the distribution of the random numbers, pairs of random numbers, triples of random numbers, and the maximum and minimum of n random numbers. In general a sequence of ℓ variables $Y_1, Y_2, \dots, Y_\ell$ was calculated from the sequence of random uniform variates. The range of the Y_i was divided into m cells of equal probability, p , and the number of occurrences, O_i , in each cell counted. The χ^2 statistic

$$\chi^2 = \sum_{i=1}^m (O_i - \ell \cdot p)^2 / \ell \cdot p \quad (5.1.1)$$

with $m - 1$ degrees of freedom was calculated and transformed to a standard normal deviate

$$T = (2 \cdot \chi^2)^{1/2} - (2 \cdot (m - 1) - 1)^{1/2} \quad (5.1.2)$$

for each test. This form is valid for $m \geq 31$. The significance level of this normal deviate was then computed as

$$S = \int_0^T (1/\sqrt{2\pi}) e^{-\frac{1}{2}t^2} dt \quad (5.1.3)$$

In each test a total of 100,000 uniform numbers was generated. For convenience the numbers were generated in 10,000 sets of 10 uniform numbers each. Five separate runs were made and the tests performed as follows:

1) Uniformity. The unit interval was divided into 1000 segments. Each segment has an expected value of 100 occurrences. The χ^2 statistic was computed as

$$\chi^2 = \sum_{i=1}^{1000} (O_i - 100)^2/100 \quad (5.1.4)$$

where O_i is the number of observed occurrences in the i^{th} segment.

2) Pairs. Successive pairs of uniform numbers were taken as the coordinates of a point in the unit square. The unit square was partitioned into 100 cells, each with an expected value of 500 occurrences. The χ^2 statistic was computed as

$$\chi^2 = \sum_{i=1}^{10} \sum_{j=1}^{10} (O_{ij} - 500)^2/500 \quad (5.1.5)$$

where O_{ij} is the number of observed occurrences in the ij^{th} cell.

3) Triples. Successive triples of uniform numbers were taken as the coordinates of a point in the unit cube (every tenth number was skipped). The unit cube was partitioned into 1000 cells with an expected

value of 30 occurrences in each cell. The χ^2 statistic was computed as

$$\chi^2 = \sum_{i=1}^{10} \sum_{j=1}^{10} \sum_{k=1}^{10} (O_{ijk} - 30)^2 / 30 \quad (5.1.6)$$

where O_{ijk} is the number of observed occurrences in the ijk^{th} cell.

4) Maximum of n random values. If $u_1, u_2, \dots, u_n$ are independent uniform $(0,1)$ deviates, then

$$W = \max(u_1, u_2, \dots, u_n) \quad (5.1.7)$$

should have the distribution

$$P(W \leq a) = F(a) = a^n \text{ for } 0 < a < 1, \text{ and} \quad (5.1.8)$$

$$F(W) = [\max(u_1, u_2, \dots, u_n)]^n \quad (5.1.9)$$

should be uniformly distributed over the interval $(0,1)$. A total of 10,000 W 's were generated for each n , and $F(W)$ was tested for uniformity using the χ^2 goodness of fit test for 100 equal subintervals. The χ^2 statistic was computed as

$$\chi^2 = \sum_{i=1}^{100} (O_i - 100)^2 / 100 \quad (5.1.10)$$

where O_i is the number of observed occurrences in the i^{th} subinterval.

In this test n takes the values 2 and 5.

5) Minimum of n random values. This test is the same as that for the maximum of n except

$$W = \min(u_1, u_2, \dots, u_n) \text{ and} \quad (5.1.11)$$

and

$$F(W) = 1 - (1 - W)^n \quad (5.1.12)$$

which should be uniformly distributed over the unit interval. The 10,000 W 's were segmented into 100 equal subintervals and the χ^2 test computed as in (5.1.10). In this test n takes the values 3 and 10.

The results of these tests are reported in Table 5.1.1. The results compare favorably with those presented in [9] and justify the use of this particular uniform (0,1) random number generator.

TABLE 5.1.1

RUN #1

TEST	CHI-SQUARE	D.F.	SIGNIFICANCE
UNIFORMITY	975.400	999	0.30186688
PAIRS	89.6720	99	0.25987495
TRIPLES	1034.7333	999	0.78918171
MAX OF 2	97.1800	99	0.46241028
MAX OF 5	80.7800	99	0.09257838
MIN OF 3	109.5200	99	0.77766504
MIN OF 10	75.0400	99	0.03713434

RUN #2

TEST	CHI-SQUARE	D.F.	SIGNIFICANCE
UNIFORMITY	1047.1600	999	0.85902187
PAIRS	92.8440	99	0.34129536
TRIPLES	1022.4667	999	0.70302930
MAX OF 2	104.4800	99	0.66267689
MAX OF 5	96.9800	99	0.45671365
MIN OF 3	102.1000	99	0.60032472
MIN OF 10	97.7400	99	0.47836695

TABLE 5.1.1 (continued)

RUN #3

TEST	CHI-SQUARE	D.F.	SIGNIFICANCE
UNIFORMITY	1006.3400	999	0.56949803
PAIRS	93.8720	99	0.36930366
TRIPLES	1052.0000	999	0.88131258
MAX OF 2	93.5400	99	0.36019046
MAX OF 5	104.5000	99	0.66318213
MIN OF 3	102.7400	99	0.61749364
MIN OF 10	107.0600	99	0.72480337

RUN #4

TEST	CHI-SQUARE	D.F.	SIGNIFICANCE
UNIFORMITY	970.2600	999	0.26223433
PAIRS	103.9040	99	0.64798213
TRIPLES	1084.4667	999	0.97021768
MAX OF 2	83.2400	99	0.12861638
MAX OF 5	96.4200	99	0.44078290
MIN OF 3	76.7000	99	0.04945168
MIN OF 10	116.5000	99	0.89040197

TABLE 5.1.1 (continued)

RUN #5

TEST	CHI-SQUARE	D.F.	SIGNIFICANCE
UNIFORMITY	1052.4000	999	0.88303445
PAIRS	82.5080	99	0.11705985
TRIPLES	1013.5333	999	0.63124447
MAX OF 2	92.6200	99	0.33528034
MAX OF 5	98.8400	99	0.50965471
MIN OF 3	140.6600	99	0.99687357
MIN OF 10	136.3000	99	0.99333696

5.2 Random Numbers Having A Discrete Distribution

Sequences of random numbers from the binomial density

$$f(x) = \binom{N}{x} \cdot p^x \cdot (1-p)^{N-x}, \quad 0 < p < 1, \quad x=0,1,\dots,N \quad (5.2.1)$$

were generated for a variety of parameter combinations. The range of the variable was divided into k intervals of probability q_i , such that the expected value of each interval, $m \cdot q_i$, was at least 10. The χ^2 statistic was computed as

$$\chi^2 = \sum_{i=1}^k (O_i - m \cdot q_i)^2 / m \cdot q_i \quad (5.2.2)$$

where O_i is the number of observed occurrences in the i^{th} interval, and m is the size of the sample generated. The significance level was computed as

$$S = (1/\Gamma(v/2)) \cdot \int_0^{\chi^2/2} e^{-t} \cdot t^{(v/2 - 1)} dt \quad (5.2.3)$$

where v = degrees of freedom, in this case $v = k-1$. The results of these tests are recorded in table 5.2.1. A sample size of 1000 random numbers were generated for each test.

Random numbers from the Poisson density

$$f(x) = \lambda^x e^{-\lambda} / x!, \quad 0 < \lambda < \infty, \quad x = 0, 1, \dots, \infty \quad (5.2.4)$$

and from the negative binomial density

$$f(x) = \binom{r+x-1}{x} \cdot p^r \cdot (1-p)^x, \quad r \geq 0, \quad 0 < p < 1, \quad (5.2.5)$$

$$x = 0, 1, 2, \dots, \infty;$$

were tested in the same manner as random numbers from the binomial density (5.2.1). The results of these tests are recorded in table 5.2.2 and 5.2.3 respectively.

TABLE 5.2.1

RESULTS OF CHI-SQUARE GOODNESS OF FIT TESTS
ON RANDOM NUMBERS FROM THE BINOMIAL DENSITY

P	N	D.F.	CHI-SQUARE	SIGNIFICANCE
0.100	10	4	6.03494	0.8034458581
0.100	20	6	6.44102	0.6243600635
0.100	30	7	2.52631	0.0748969943
0.100	40	9	6.33123	0.2936355148
0.100	50	9	12.92191	0.8338284965
0.200	10	5	7.56431	0.8180614798
0.200	20	8	16.34847	0.9623445588
0.200	30	10	15.13733	0.8728654417
0.200	40	11	9.00770	0.3788180126
0.200	50	13	12.00444	0.4727203102
0.300	10	7	7.97911	0.6655606411
0.300	20	9	10.98540	0.7232888983
0.300	30	11	17.78713	0.9133467551
0.300	40	13	12.08372	0.4792085395
0.300	50	15	21.66071	0.8829858823
0.400	10	7	7.29406	0.6010818180
0.400	20	10	12.18194	0.7269356404
0.400	30	12	20.64248	0.9441325898
0.400	40	14	10.05124	0.2415723570
0.400	50	16	12.35961	0.2811236434
0.500	10	8	6.86156	0.4483592431
0.500	20	10	12.57912	0.7518355135
0.500	30	12	10.41700	0.4205727006
0.500	40	14	23.28293	0.9441810635
0.500	50	16	9.42171	0.1049637735
0.600	10	7	6.92477	0.5632454466
0.600	20	10	18.27415	0.9494885520
0.600	30	12	10.17474	0.3993657394
0.600	40	14	11.25580	0.3341658613
0.600	50	16	10.03645	0.1352818684

TABLE 5.2.1 (continued)

P	N	D.F.	CHI-SQUARE	SIGNIFICANCE
0.700	10	7	3.47126	0.1617396643
0.700	20	9	10.70221	0.7033259190
0.700	30	11	12.02199	0.6380075901
0.700	40	13	13.88283	0.6178529373
0.700	50	15	18.60988	0.7680404078
0.800	10	5	4.53985	0.5254310881
0.800	20	8	16.74590	0.9671335640
0.800	30	10	10.97150	0.6402573630
0.800	40	11	9.02074	0.3800224541
0.800	50	13	15.05620	0.6961423824
0.900	10	4	3.31137	0.4928666476
0.900	20	6	4.40204	0.3775593406
0.900	30	7	6.78871	0.5487935778
0.900	40	9	6.74965	0.3368342656
0.900	50	9	5.04256	0.1694175512

RESULTS OF CHI-SQUARE GOODNESS OF FIT TESTS ON
RANDOM NUMBERS FROM THE POISSON DENSITY.

LAM	D.F.	CHI-SQUARE	SIGNIFICANCE
5.00	10	13.57652	0.8068012578
10.00	15	8.51890	0.0986988385
15.00	17	16.33853	0.5000245564
20.00	20	18.35403	0.4359007928
25.00	21	28.70494	0.8787325544
30.00	23	24.94425	0.6468336304
35.00	25	26.91360	0.6397852204
40.00	27	43.19749	0.9750174127
45.00	28	22.55633	0.2450911949
50.00	30	23.26187	0.1956567510
55.00	30	42.01434	0.9286283430
60.00	31	34.81330	0.7087406320
65.00	33	29.49011	0.3573849816
70.00	34	29.73472	0.3231975827
75.00	35	26.18810	0.1410044103
80.00	35	26.80832	0.1619395204
85.00	36	21.61020	0.0278060381
90.00	37	28.29852	0.1527661009
95.00	38	30.51639	0.1992224404
100.00	39	37.10850	0.4435994584

TABLE 5.2.2

TABLE 5.2.3
RESULTS OF CHI-SQUARE GOODNESS OF FIT TESTS ON RANDOM
NUMBERS FROM THE NEGATIVE BINOMIAL DENSITY.

P	R	D.F.	CHI-SQUARE	SIGNIFICANCE
0.100	2	41	36.08627	0.3115372475
0.100	4	57	51.64328	0.3244292796
0.100	6	67	77.44254	0.8201664431
0.100	8	72	54.21390	0.0583581269
0.100	10	76	58.52535	0.0683503608
0.200	2	23	20.00081	0.3581367902
0.200	4	33	27.73007	0.2730820886
0.200	6	40	43.00986	0.6563763897
0.200	8	45	42.83166	0.4357673160
0.200	10	49	38.69528	0.1455417220
0.300	2	15	11.10495	0.2548804445
0.300	4	22	25.94653	0.7460151638
0.300	6	28	17.46899	0.0612580512
0.300	8	31	25.10221	0.2368731801
0.300	10	34	18.65952	0.0151991805
0.400	2	11	12.32864	0.6605459138
0.400	4	16	7.20666	0.0309309696
0.400	6	20	20.18198	0.5534007984
0.400	8	23	23.12594	0.5466209213
0.400	10	25	22.04679	0.3669595654
0.500	2	8	4.96920	0.2391366628
0.500	4	12	13.04976	0.6345655810
0.500	6	15	16.03409	0.6202105433
0.500	8	17	25.16863	0.9090016643
0.500	10	19	24.55579	0.8243107298
0.600	2	6	13.11971	0.9588261108
0.600	4	9	14.42509	0.8920130256
0.600	6	11	13.29194	0.7253265963
0.600	8	13	17.84469	0.8364937782

TABLE 5.2.3 (continued)

P	R	D.F.	CHI-SQUARE	SIGNIFICANCE
0.600	10	14	13.95647	0.5470410000
0.700	2	5	6.42472	0.7329438096
0.700	4	7	11.10162	0.8657526004
0.700	6	8	9.98931	0.7342226732
0.700	8	10	4.82568	0.0974859792
0.700	10	11	14.96953	0.8161040822
0.800	2	3	1.03482	0.2071720727
0.800	4	5	2.88184	0.2818034287
0.800	6	6	1.48994	0.0398398009
0.800	8	7	4.59512	0.2907654804
0.800	10	8	2.13612	0.0234479888
0.900	2	2	0.67942	0.2880246110
0.900	4	3	0.23420	0.0281120703
0.900	6	3	4.99992	0.8281970919
0.900	8	4	2.59324	0.3719796012
0.900	10	4	2.34658	0.3276973945

5.3 Random Numbers From Continuous Distribution Functions

Sequences of random numbers from the exponential

$$f(t) = e^{-t}, \quad 0 < t < \infty \quad (5.3.10)$$

and normal

$$f(x) = (1/\sqrt{2\pi}) e^{-\frac{1}{2}x^2}, \quad -\infty < x < \infty \quad (5.3.2)$$

densities were generated, and segmented into 20 intervals of approximately equal probability, $p_i = 0.05$. Samples of $m = 1000$ random numbers were generated and the χ^2 statistic computed as

$$\chi^2 = \sum_{i=1}^{20} (O_i - m \cdot p_i)^2 / m \cdot p_i \quad (5.3.3)$$

where O_i is the number of observed occurrences in the i^{th} interval. The significance level was calculated as given in (5.2.3) with $v = 19$. The results of these tests are given in tables 5.3.1 and 5.3.2.

Sequences of random numbers from the gamma density

$$f(t) = (1/\Gamma(n)) t^{n-1} e^{-t}, \quad 0 < t < \infty, \quad (5.3.4)$$

were tested in a somewhat different manner. If t is a variable with density (5.3.4) then

$$y = 2 \cdot (t)^{\frac{1}{2}} - (4 \cdot n - 1)^{\frac{1}{2}} \quad (5.3.5)$$

should have an approximate standard normal density when n is large.

Sequences of m gamma distributed random numbers were generated and using the transformation (5.3.5), transformed to m approximately standard normal variables. As before, the normal distribution was partitioned into 20

RESULTS OF CHI-SQUARE GOODNESS OF FIT TESTS ON RANDOM
NUMBERS FROM THE EXPONENTIAL DENSITY, $F(X)=\text{EXP}(-X)$.

D.F.	CHI-SQUARE	SIGNIFICANCE
19	20.76000	0.6497876
19	11.24000	0.0844989
19	17.64000	0.4534074
19	22.68000	0.7482694
19	13.68000	0.1979842
19	18.52000	0.5120074
19	11.48000	0.0933650
19	17.36000	0.4345072
19	24.24000	0.8128932
19	37.24000	0.9925960

TABLE 5.3.1

RESULTS OF CHI-SQUARE GOODNESS OF FIT TESTS ON RANDOM
NUMBERS FROM THE NORMAL DENSITY

D.F.	CHI-SQUARE	SIGNIFICANCE
19	19.59699	0.5808149783
19	25.42150	0.8528693863
19	18.29836	0.4974018037
19	6.71247	0.0044039802
19	14.75770	0.2621316817
19	19.91207	0.6001134403
19	20.76446	0.6500395713
19	23.50146	0.7840244000
19	11.96570	0.1129139927
19	18.02468	0.4792096579

TABLE 5.3.2

intervals of approximately equal probability $p_i = 0.05$, and the χ^2 statistic computed as before. The results of these tests are recorded in table 5.3.3.

RESULTS OF CHI-SQUARE GOODNESS OF FIT TESTS ON TRANSFORMED
GAMMA RANDOM NUMBERS

SAMPLE SIZE	N	D.F.	CHI-SQUARE	SIGNIFICANCE
100	50	19	12.30156	0.1276659399
100	100	19	14.27597	0.2326276646
100	200	19	13.79290	0.2043727108
200	50	19	13.99520	0.2160288470
200	100	19	20.25710	0.6207407158
200	200	19	23.61244	0.7885588803
500	50	19	23.13291	0.7684615439
500	100	19	21.91446	0.7114659110
500	200	19	32.02104	0.9689153752

TABLE 5.3.3

5.4 Discussion of Results.

It should be emphasized that no test or series of tests can assure the suitability of a given sequence of random numbers for a particular problem. When possible the sequence should be tested on a similar problem with known solution. However, the results presented here establish the fundamental reliability of the generation procedures presented in this paper. The significance levels obtained should follow the uniform distribution on (0,1) if the corresponding null hypothesis tested are valid. The above results are in agreement with this condition.

CHAPTER 6
DESCRIPTION AND USE
OF GENERATOR PROGRAMS

6.1 GENO

GENO supplies the user with a very fast procedure for the generation of independent uniform (0,1) random numbers with density

$$f(u) = 1, \quad 0 < u < 1. \quad (6.1.1)$$

The method used was suggested by Dr. Rolf Bargmann [1] and is a form of the multiplicative congruential procedure,

$$u_{n+1} = a \cdot u_n \bmod 2^{32} \quad (6.1.2)$$

where $a = 5^{13}$. This procedure requires that u_0 , the starting value, be an odd positive integer.

GENO is written as a subroutine with entry point URAN. It is called as follows:

$$U = \text{URAN}(\text{IOD}), \quad (6.1.3)$$

where "IOD" is the starting value and must be an odd positive integer.

The desired random number U in (6.1.3), is returned to the calling program in single precision real mode.

GENO requires 20 words (80 bytes) of storage space and an average time per generation of 25 usec.

6.2 GEN1

GEN1 provides the user with a fast procedure for the generation of exponentially distributed random numbers

$$f(x) = e^{-x}, \quad 0 < x < \infty. \quad (6.2.1)$$

The method used is that described in 4.1, and presented by Marsaglia in [13].

GEN1 is written as a subroutine with entry point RANEXP. It is called from a FORTRAN main program as follows:

$$X = \text{RANEXP}(\text{IOD}). \quad (6.2.2)$$

The parameter "IOD" primes the scheme for the generation of uniform (0,1) random numbers, "IOD" must be an odd positive integer to insure the proper generation of u. The required random number x in (6.2.2) is returned to the calling program in single precision real mode.

The average time per generation is approximately 70 usec. Memory requirement for GEN1 is 307 words (1228 bytes).

6.3 GEN2

GEN2 supplies the user with a fast procedure for the generation of gamma ($\alpha = n$, $\beta = 1$)

$$g(y) = 1/\Gamma(n) \cdot y^{n-1} \cdot e^{-y}, \quad 0 < y < \infty \quad (6.3.1)$$

distributed random numbers. The technique used is described in 4.2.

GEN2 is written as a subroutine with entry point RANGAM, and is called from a FORTRAN main program as follows:

$$Y = \text{RANGAM}(\text{IOD}, N). \quad (6.3.2)$$

As before, "IOD" must be an odd positive integer and $n = "N"$ in (6.3.1).

The required variate is returned in single precision real mode.

GEN2 requires 313 words of memory space and approximately $65 \cdot N$ usec per generation for $N > 1$.

6.4 GEN3

GEN3 supplies the user with a fast procedure for the generation of random numbers from the standard normal density

$$f(x) = (1/\sqrt{2\pi}) e^{-\frac{1}{2}x^2}, \quad -\infty < x < \infty. \quad (6.4.1)$$

The procedure is described in 4.3 and by Marsaglia in [12].

GEN3 is written as a subroutine and is called from a FORTRAN main program as follows:

$$X = \text{RANORM}(\text{IOD}). \quad (6.4.2)$$

The parameter "IOD" must be an odd positive integer, and the required variate X in (6.4.2) is returned to the calling program in single precision real mode.

Memory requirement for GEN3 is 442 words. The average time per generation is approximately 65 usec.

6.5 GEN4

GEN4 provides a fast procedure for the generation of random numbers from the binomial distribution

$$f(x) = \binom{n}{x} p^x (1-p)^{n-x}, \quad 0 < p < 1, \quad x=0,1,\dots,n. \quad (6.5.1)$$

GEN4 is written as a subroutine with two entry points, BSETUP and RANBI. The user calls BSETUP with the parameters "P" and "N" e.g.

CALL BSETUP(P,N). (6.5.2)

BSETUP calculates point probabilities using the recursion relation

$$f(0) = (1 - P)^N \text{ and} \quad (6.4.3)$$

$$f(x+1) = f(x) \cdot P \cdot (N-x)/((x+1) \cdot (1-P)).$$

The four sets of discrete values are then stored using the procedure described in Chapter 2. BSETUP need be called only once for a given "P" and "N", but must be called at least once prior to calling RANBI.

RANBI addresses the very fast scheme that generates the required variates. It is called as follows:

X = RANBI(IOD) (6.4.4)

where "IOD" must be an odd positive integer, and X is returned in single precision real mode.

In its present form GEN4 requires the "P" be in the real mode, $0 < P < 1$, and that "N" be in the integer mode and not exceed 255. The program requires 948 words of memory and an average time per generation of approximately 35-40 usec. Figure 2.1 describes the flowchart of GEN4 except for the section that computes point probabilities.

If the user desires random numbers returned in the integer mode, he need only delete

STMF

199	ST	0, RES
200	MVI	RES, X'46'
201	AE	0, RES

and change the designation RANBI to IANBI in his calling statement and in statements

```

170          ENTRY RANBI
171          USING RANBI, 15
173          DC    CL7' ..RANBI'
174RANBI      STM   1,3, 24(13) .

```

Storage requirements may also be altered simply by changing

```

STMT
146          C      7, = F'2000'
222          DS     2000XL1.

```

6.5 GEN5

GEN5 supplies a fast routine for the generation of random numbers from the poisson density

$$f(x) = \lambda^x e^{-\lambda} / x! , \quad 0 < \lambda < \infty , \quad x = 0, 1, \dots \quad (6.5.1)$$

Since the domain of x is infinite, it is necessary to truncate the distribution. This is done at both low and high point values whenever a point probability is less than 16^{-4} .

GEN5 is written as a subroutine with two entry points, PSETUP and RANPOI. The user calls PSETUP with the parameter λ , e.g.

```
CALL PSETUP (ALAM). \quad (6.5.2)
```

PSETUP calculates point probabilities using the recursion relation

$$f(0) = e^{-\lambda}$$

$$f(x+1) = f(x) \cdot \lambda / (x+1), \quad (6.5.3)$$

and stores the four sets of discrete values by means of the technique described in Chapter 2. PSETUP need be called only once for a given λ , but must be called at least once before the initial call to RANPOI.

The entry point RANPOI addresses the very fast scheme that performs the actual generation of the required variate. It is called as follows:

$$X = \text{RANPOI}(\text{IOD}), \quad (6.5.4)$$

where "IOD" must be an odd positive integer and the desired random number X is returned in single precision real mode.

GEN5 requires 972 words of memory space and an average time per generation of approximately 35-40 usec. The flow of GEN5 follows that presented in figure 2.1 except for the section that computes the point probabilities.

If the user desires his result returned in the integer mode, he need only delete

STMT		
190	ST	0,RES
191	MVI	RES, X'46'
192	AE	0,RES

and change the designation RANPOI to IANPOI in his calling statement and in statements

STMT	
161	ENTRY RANPOI
162	USING RANPOI
164	DC CL7' RANPOI'
165 RANPOI	STM 1,3,24(13)

Storage requirement may also be altered by changing

STMT

```
137                      C      7, = F'2000'
211 A                      DS    2000XL1
```

6.6 GEN6

GEN6 provides a fast routine for generating random numbers from the negative binomial density

$$f(x) = \binom{x+r-1}{x} p^r (1-p)^x, \quad 0 < p < 1, x = 0, 1, \dots, \quad (6.6.1)$$

and $r \geq 0$.

The domain of x is infinite, consequently, the distribution is truncated at both low and high point values whenever a point probability is less than 16^{-4} .

GEN6 is written as a subroutine with two entry points, NBSETU and RANEBI. NBSETU receives the parameters, p and r , and calculates point probabilities using the recursion relation

$$f(0) = p^r \quad (6.6.2)$$

$$f(x+1) = f(x) \cdot (1-p) \cdot (x+r)/(x+1).$$

Having computed the point probabilities, NBSETU then stores the four sets of discrete values using the procedure described in Chapter 2. NBSETU is called as follows:

$$\text{CALL NBSETU(P,IR),} \quad (6.6.3)$$

where "IR" must be in the integer mode and $0 < "P" < 1$. NBSETU need be

called only once for a given "P" and "IR", but must be called at least once prior to the initial call to RANEBI.

RANEBI addresses the very fast scheme for the generation of the required variates, and is called as follows:

$$X = \text{RANEBI}(\text{IOD}), \quad (6.6.4)$$

where "IOD", the primer, must be an odd positive integer, and X is returned in single precision real mode.

GEN6 requires 944 words of storage space and an average time per generation in the 35-40 usec range. The flow-chart of GEN6 follows that of figure 2.1 except for the section of GEN6 that calculates the point probabilities.

6.7 GEN7

GEN7 provides a fast procedure for generating discrete random variables with any specified probability distribution denoted by a vector $P = \{p_1, p_2, \dots\}$. It is written as a subroutine with two entry points, DSETUP and RANDIS. DSETUP is called with 3 parameters, "P", "IX", "N"; where "P" is a vector of probabilities (real mode), "IX" is a vector of corresponding discrete values (integer mode), and "N" is the number of elements in vectors, e.g.

$$\text{CALL DSETUP}(P, IX, N). \quad (6.7.1)$$

The discrete values of the "IX" vector are stored in four sets based on the high order 4 digits of the elements of the "P" vector. This procedure is given in Chapter 2 and illustrated in figure 2.1. DSETUP need be called

only once for a particular set of parameters, however, it must be called at least once prior to the initial call to RANDIS.

RANDIS addresses the very fast scheme that performs the actual generation of the required variates. It is called as follows:

$$X = \text{RANDIS}(\text{IOD}), \quad (6.7.2)$$

where "IOD" must be an odd positive integer and "X", the generated variate is returned in single precision real mode.

In its present form GEN7 requires that the elements of the "IX" vector be in the integer mode and that no value exceed 255. Also, the parameter "N", the number of elements in the vectors, may not exceed 256. If the user wishes his discrete variables, members of the "IX" vector, to be in the real mode or to be in the integer mode but with values greater than 255, he need only make the changes indicated by table 6.1. If he desires to have his result returned in the integer mode he need only delete the following

STMT		
149	ST	0,RES
150	MVI	RES, X'46'
151	AE	0,RES,

and change the designation RANDIS to IANDIS in his calling statement and in the following

STMT		
120	ENTRY	RANDIS
121	USING	RANDIS, 15
123	DC	CL7' RANDIS'
124 RANDIS	STM	1,3,24(13).

Storage requirement may also be altered simply by changing

STMT

97

C

7, = F'2000'

168 A

DS

2000XL1

In its present form GEN7 requires 904 words of memory space and
35-40 usec per generation.

STMT	INTEGER > 255				REAL MODE			
	L	6, = F'1'	LA	6,4	LA	6,4		
87	C	7, = F'2000'	C	7, = F'SPAC'	C	7, = F'SPAC'		
97	L	2, -0(3,11)	L	2, 0(3,11)	LE	2, 0(3,11)		
99	STC	2, A(12)	ST	2, A(12)	STE	2, A(12)		
101	CR	0,4	C	0, = F'16'	C	0, = F'16'		
106	IC	0, A(2)	EX SLL	2, 2	EX SLL	2, 2		
148 EX	DS	2000 XL1	A DS	SPAC XL4	A DS	SPAC XL4		
168 A								
ALSO INSERT BETWEEN								
STMT 95 AND STMT 96								
SLL 8, 2								
AND DELETE STMT 149								
THROUGH STMT 151.								

TABLE 6.1

REFERENCES

- [1] Bargmann, R. E. private communication, Department of Statistics, University of Georgia, (November, 1969).
- [2] Forsythe, G. E. "Generation and Testing of Random Digits at the National Bureau of Standards, Los Angeles, Monte Carlo Method.", Nat. Bur. Stan., Appl. Math. Series 12, pp. 34-35, (1951).
- [3] Hull, T. E., and A. R. Dobell, "Random Number Generators." SIAM Rev., Vol. 4, pp. 230-254, (1962).
- [4] Jansson, B. Random Number Generators, Stockholm, Almqvist and Wiksell (distr.), (1966)
- [5] Kendall, M. G., and B. Babington-Smith, "Randomness and Random Sampling Numbers." J. Roy. Stat. Soc. 101, pp. 147-166, (1938)
- [6] Kendall, M. G., and B. Babington-Smith, "Second Paper on Random Sampling Numbers." J. Roy. Stat. Soc. Supplement 6, pp. 51-61, (1939).
- [7] Kendall, M. G., and B. Babington-Smith, "Tables of Random Sampling Numbers." Tracts for Computers, No. 24, Cambridge, (1939).
- [8] Lehmer, D. H. "Mathematical Methods in Large Scale Computing Units." Annals Comp. Laboratory Harvard Univ. 26, pp. 141-146, (1941).
- [9] MacLaren, M. D., and G. Marsaglia, "Uniform Random Number Generators." J. ACM 12, No. 1, pp. 85-89, (January, 1965).
- [10] Marsaglia, G. "Expressing a Random Variable in Terms of Uniform Random Variables." Ann. Math. Stat., Vol. 32, pp. 894-898, (1961)
- [11] Marsaglia, G. "Generating Discrete Random Variables in a Computer." Comm. ACM 6, No. 1, pp. 37-38, (January, 1963).
- [12] Marsaglia, G, and M. D. MacLaren, and T. A. Bray, "A Fast Procedure for Generating Normal Random Variables." Comm. ACM 7, No. 1, pp. 4-10, (January, 1964).

- [13] Marsaglia, G, and M. D. MacLaren, and T. A. Bray, "A Fast Procedure for Generating Exponential Random Variables." Comm. ACM 7, No. 5, pp. 298-300, (May, 1964).
- [14] Muller, M. E. "An Inverse Method for the Generation of Random Normal Deviates on Large Scale Computers." Math. Tables Other Aids 12, pp. 167-174, (1958).
- [15] Rand Corporation, A Million Random Digits with 100,000 Normal Deviates, Free Press, Glencoe, (1955).
- [16] Strawderman, W. E. "Generation and Testing of Pseudo-Random Numbers.", Technical Report No. 171, N00014-67-A-0112-0053 (NR-042-267), Stanford Univ., (February, 1971)
- [17] Student, "The Probable Error of a Mean." Biometrika, Vol. 6, pp. 1-25, (1908).
- [18] Teisgroew, D. "Distribution Sampling with High Speed Computers.", Diss., Univ. of North Carolina, (1953).
- [19] Tippett, L. H. C. "Random Sampling Numbers." Tracts for Computers, No. 15, Cambridge, (1927).
- [20] van Hoerner, S. "Herstellung von Zufallszahlen auf Rechenautomaten." Angew. Math. Physik, Vol. 8, pp. 26-52, (1957)
- [21] Wall, D. D. "Fibonacci Series Module n." The Am. Math. Monthly, Vol. 67, pp. 525-532, (1960).

APPENDIX A

Assembler Listing of GENO

NOT REPRODUCIBLE

LCC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

```

000000
1 GENO START 0
2 *****
3 * SOURCE---L.E.CANNON, UGA, DEPT. OF STAT., APR 1970
4 * PURPOSE---TO SUPPLY THE USER WITH A FAST PROCEDURE FOR GENERATING UN-
5 * IFORNIA, 11 RANDOM NUMBERS.
6 * USAGE---X=URAN(100). *IND* MUST BE AN ODD POSITIVE INTEGER. THE PE-
7 * QUIRED VARIATE IS RETURNED IN SINGLE PRECISION REAL MODE.
8 * EXAMPLE:X=URAN(56865) *X* WILL BE UNIFORM ON (0,1).
9 * METHOD---THE METHOD IS THE MULTIPLICATIVE CONGRUENTIAL PROCEDURE
10 * X(I+1)=X(I)*A
11 * WHERE A=5*13, AS SUGGESTED BY DR. R. F. BARNHANN.
12 *****
13 ENTRY URAN
14 USING URAN,15
15 DC XL1,07*
16 DC CL7* URAN*
17 URAN STM 1,3,24(13)
18 SEP 0,0
19 BRANCH R TEMP
20 M 2,RPAT
21 ST 3,LA
22 SRL 3,8
23 ST 3,RES
24 MVI RES,X'40*
25 AE 7,RES
26 LM 1,3,24(13)
27 RCR 15,14
28 L 3,LA
29 MVC BRANCH,TEMP
30 L 3,(1,1)
31 L 3,(1,3)
32 B BRANCH+4
33 RES DS 1F
34 RPAT DS 1F
35 LA XL4*48C27395*
36 END

```

CLEAR FROO FOR RESULT.
BRANCH TO TEMP.
GENERATE U IN R3.
STORE U FOR NEXT CALL.
CONVERT TO FLOATING POINT.

NORMALIZED RESULT IN FROO.
RETURN TO CALLING PROGRAM.

REPLACE INSTR. AT BRANCH BY INSTR. AT TEMP.
LOAD ADDRESS OF PRIMER.
LOAD PRIMER.
BRANCH TO INSTR. FOLLOWING BRANCH.

00000A 07
000001 404040E409C105
00000A 0013 0019
00000C 3A03
00000E 47E0 F028
000012 5020 F044
000016 5330 F048
00001A 8931 0005
00001E 5031 F040
000022 9240 F040
000026 7A01 F040
00002A 0813 0018
00002E 07FE
000030 5930 F048
000034 0201 F096 F028 0000E
00003A 5830 1000
00003E 5830 3000
000042 47F0 F00A
000048
00004C 48C27395
000050

APPENDIX B

Assembler Listing of GEN1

88-10314

SOURCE STATEMENT

SYM

ADDR1 ADDR2

2003 153480 237

၂၀၁၇

```

1 GEN1          START 0
2 .....
3 * SOURCE---L.E.CANNON, UGA, DEPT. OF STAT., APR 1970
4 * PURPOSE---TO SUPPLY THE USER WITH A FAST PROCEDURE FOR GENERATING EX-
5 * PANDY DISTRIBUTED RANDOM NUMBERS.
6 * USAGE---X=RAYEXP(II). II MUST BE AN ODD INTEGER. IT PRIMES THE GEN-
7 * ERATING SCHEME.
8 * EXAMPLE---X=RAYEXP(556951).
9 * METHOD---THE METHOD IS THAT OF MARSAGLIA AS DESCRIBED IN "COMJUNI-
10 * CATIONS OF ACM, VOL. 7, NO. 5, MAY 1964. THE PROGRAM DOCUMENTA-
11 * TION WILL FOLLOW THE NOTATION USED BY MARSAGLIA.
12 .....
13 ENTRY RANEXP
14 USING RANEXP,15
15 DC XCL1074
16 CC CL74 RANEXP,
17 RANEXP SYN 1,9,24(113)
18 BRANCH B YEMP
19
20 M 0,0
21 ST 5,LA
22 SR 6,6
23 NI
24
25 SLOL 4,4
26 C 4,4,370
27 RC 11,NA
28 IC 6,0,294(4)
29 SLL 5,6
30 BC 1,EX
31 SLOL 4,3
32 C 4,4,4561
33 RC 11,NA
34 IC 6,0,144(4)
35 SLL 5,3
36 SLOL 4,3
37 C 4,4,33630
38 RC 11,NC
39 S 4,4,33639
40 IC 6,0(4)
41 EX
42 FL
43 ST
44 MVI RESI,41
45 LE 0,RES
46 LM 1,9,24(113)
47 RCR 15,14
48 NC
49 LA 2,4
50 SR 3,3
51 CC 5,PI3)
52 DC 11,NO
53 SR 3,2
54 NE
55 CL 5,RI(3)

```

LOC	OBJECT CODE	ADDR1	ADDR2	SYMT	SOURCE STATEMENT
000094	47A0 F0A6	00008E		96	RC 11, NF
000098	5551 F034	0025C		97	CL 5, C(1)
00009C	47A0 F0C8	0000D0		98	BC 11, NF
0000A0	5C40 F130	000138		99	M 4, RPT
0000A4	5051 F124	000130		01	ST 5, LA
0000A8	1044			01	SR 4, 4
0000AA	0002			01	CO
0000AE	4343 F16C	00302		02	SPI 3, 2
0000B2	47F3 F05C	00374		03	IC 4, (1)
0000B6	5859 F124	00764		04	AC 15, FL
0000BA	0703 F0A4	00130		05	TEMP
0000BE	4352 1G04	00086		06	MVC
0000C0	5453 50D3	00300		07	L 5, J(1, 1)
0000C4	5C50 F124	0000C		08	L 5, C(1, 5)
0000C8	47E0 F0A4	00130		09	ST 5, LA
0000CC	5C40 F130	00110		10	M 4, RPT
0000D0	1482	00138		11	MF
0000D4	1482			12	LR 9, 2
0000D8	5554 F134	0013C		13	AR 9, 2
0000DC	47A0 F0CF	000D6		14	CL 5, (1, 1)
0000E0	5C43 F130	00138		15	RC 11, 0-6
0000E4	1482			16	M 4, RPT
0000E8	1485			17	SP 8, 2
0000EC	11C9			18	LR 5, 5
0000F0	5C40 F130	00138		19	LR 9, 2
0000F4	1482			20	SR 8, 2
0000F8	1482			21	NR 9, 2
0000FC	5C40 F130	00138		22	SR 8, 2
000100	1554			23	CL 9, 6
000104	47A0 F0E0	000E8		24	PC 11, 0-6
000108	1485			25	LR 6, 4
00010C	47A0 F0E4	000FC		26	BRM
000110	5053 F124	00130		27	ST 5, LA
000114	1485			28	LR 5, 4
000118	47F3 F0A0	000A8		29	AC 15, 00
00011C	5553 F134	0015C		30	AR 3, 2
000120	47A0 F0FE	00106		31	CL 5, (1, 1)
000124	5C43 F130	00134		32	BC 11, 00
000128	47F3 F0A0	00134		33	ST 3, J
000130	5553 F134	00134		34	J, X, 4, 4
000134	5C43 F130	00138		35	AF 0, J
000138	5053 F124	00130		36	M 4, RPT
000140	5C43 F134	00130		37	ST 5, LA
000144	47F3 F0A0	0015C		38	M 4, RPT
000148	5553 F134	0013C		39	AC 15, 00
000150	47F3 F0A0	0013C		40	DS 15, 00
000154	5553 F134	0013C		41	DS 15, 00
000158	47F3 F0A0	0013C		42	DS 15, 00
000160	5553 F134	0013C		43	DS 15, 00
000164	47F3 F0A0	0013C		44	DS 15, 00
000168	5553 F134	0013C		45	DS 15, 00
000170	47F3 F0A0	0013C		46	DS 15, 00
000174	5553 F134	0013C		47	DS 15, 00
000178	47F3 F0A0	0013C		48	DS 15, 00
000180	5553 F134	0013C		49	DS 15, 00
000184	47F3 F0A0	0013C		50	DS 15, 00
000188	5553 F134	0013C		51	DS 15, 00
000190	47F3 F0A0	0013C		52	DS 15, 00
000194	5553 F134	0013C		53	DS 15, 00
000198	47F3 F0A0	0013C		54	DS 15, 00
000200	5553 F134	0013C		55	DS 15, 00
000204	47F3 F0A0	0013C		56	DS 15, 00
000208	5553 F134	0013C		57	DS 15, 00
000210	47F3 F0A0	0013C		58	DS 15, 00
000214	5553 F134	0013C		59	DS 15, 00
000218	47F3 F0A0	0013C		60	DS 15, 00
000220	5553 F134	0013C		61	DS 15, 00
000224	47F3 F0A0	0013C		62	DS 15, 00
000228	5553 F134	0013C		63	DS 15, 00
000230	47F3 F0A0	0013C		64	DS 15, 00
000234	5553 F134	0013C		65	DS 15, 00
000238	47F3 F0A0	0013C		66	DS 15, 00
000240	5553 F134	0013C		67	DS 15, 00
000244	47F3 F0A0	0013C		68	DS 15, 00
000248	5553 F134	0013C		69	DS 15, 00
000250	47F3 F0A0	0013C		70	DS 15, 00
000254	5553 F134	0013C		71	DS 15, 00
000258	47F3 F0A0	0013C		72	DS 15, 00
000260	5553 F134	0013C		73	DS 15, 00
000264	47F3 F0A0	0013C		74	DS 15, 00
000268	5553 F134	0013C		75	DS 15, 00
000270	47F3 F0A0	0013C		76	DS 15, 00
000274	5553 F134	0013C		77	DS 15, 00
000278	47F3 F0A0	0013C		78	DS 15, 00
000280	5553 F134	0013C		79	DS 15, 00
000284	47F3 F0A0	0013C		80	DS 15, 00
000288	5553 F134	0013C		81	DS 15, 00
000290	47F3 F0A0	0013C		82	DS 15, 00
000294	5553 F134	0013C		83	DS 15, 00
000298	47F3 F0A0	0013C		84	DS 15, 00
000300	5553 F134	0013C		85	DS 15, 00
000304	47F3 F0A0	0013C		86	DS 15, 00
000308	5553 F134	0013C		87	DS 15, 00
000310	47F3 F0A0	0013C		88	DS 15, 00
000314	5553 F134	0013C		89	DS 15, 00
000318	47F3 F0A0	0013C		90	DS 15, 00
000320	5553 F134	0013C		91	DS 15, 00
000324	47F3 F0A0	0013C		92	DS 15, 00
000328	5553 F134	0013C		93	DS 15, 00
000330	47F3 F0A0	0013C		94	DS 15, 00
000334	5553 F134	0013C		95	DS 15, 00
000338	47F3 F0A0	0013C		96	DS 15, 00
000340	5553 F134	0013C		97	DS 15, 00
000344	47F3 F0A0	0013C		98	DS 15, 00
000348	5553 F134	0013C		99	DS 15, 00
000350	47F3 F0A0	0013C		100	DS 15, 00
000354	5553 F134	0013C		101	DS 15, 00
000358	47F3 F0A0	0013C		102	DS 15, 00
000360	5553 F134	0013C		103	DS 15, 00
000364	47F3 F0A0	0013C		104	DS 15, 00
000368	5553 F134	0013C		105	DS 15, 00
000370	47F3 F0A0	0013C		106	DS 15, 00
000374	5553 F134	0013C		107	DS 15, 00
000378	47F3 F0A0	0013C		108	DS 15, 00
000380	5553 F134	0013C		109	DS 15, 00
000384	47F3 F0A0	0013C		110	DS 15, 00
000388	5553 F134	0013C		111	DS 15, 00
000390	47F3 F0A0	0013C		112	DS 15, 00
000394	5553 F134	0013C		113	DS 15, 00
000398	47F3 F0A0	0013C		114	DS 15, 00
000400	5553 F134	0013C		115	DS 15, 00
000404	47F3 F0A0	0013C		116	DS 15, 00
000408	5553 F134	0013C		117	DS 15, 00
000410	47F3 F0A0	0013C		118	DS 15, 00
000414	5553 F134	0013C		119	DS 15, 00
000418	47F3 F0A0	0013C		120	DS 15, 00
000420	5553 F134	0013C		121	DS 15, 00
000424	47F3 F0A0	0013C		122	DS 15, 00
000428	5553 F134	0013C		123	DS 15, 00
000430	47F3 F0A0	0013C		124	DS 15, 00
000434	5553 F134	0013C		125	DS 15, 00
000438	47F3 F0A0	0013C		126	DS 15, 00
000440	5553 F134	0013C		127	DS 15, 00
000444	47F3 F0A0	0013C		128	DS 15, 00
000448	5553 F134	0013C		129	DS 15, 00
000450	47F3 F0A0	0013C		130	DS 15, 00
000454	5553 F134	0013C		131	DS 15, 00
000458	47F3 F0A0	0013C		132	DS 15, 00
000460	5553 F134	0013C		133	DS 15, 00
000464	47F3 F0A0	0013C		134	DS 15, 00
000468	5553 F134	0013C		135	DS 15, 00
000470	47F3 F0A0	0013C		136	DS 15, 00
000474	5553 F134	0013C		137	DS 15, 00
000478	47F3 F0A0	0013C		138	DS 15, 00
000480	5553 F134	0013C		139	DS 15, 00
000484	47F3 F0A0	0013C		140	DS 15, 00
000488	5553 F134	0013C		141	DS 15, 00
000490	47F3 F0A0	0013C		142	DS 15, 00
000494	5553 F134	0013C		143	DS 15, 00
000498	47F3 F0A0	0013C		144	DS 15, 00
000500	5553 F134	0013C		145	DS 15, 00
000504	47F3 F0A0	0013C		146	DS 15, 00
000508	5553 F134	0013C		147	DS 15, 00
000510	47F3 F0A0	0013C		148	DS 15, 00
000514	5553 F134	0013C		149	DS 15, 00
000518	47F3 F0A0	0013C		150	DS 15, 00
000520	5553 F134	0013C		151	DS 15, 00
000524	47F3 F0A0	0013C		152	DS 15, 00
000528	5553 F134	0013C		153	DS 15, 00
000530	47F3 F0A0	0013C		154	DS 15, 00
000534	5553 F134	0013C		155	DS 15, 00
000538	47F3 F0A0	0013C		156	DS 15, 00
000540	5553 F134	0013C		157	DS 15, 00
000544	47F3 F0A0	0013C		158	DS 15, 00
000548	5553 F134	0013C		159	DS 15, 00
000550	47F3 F0A0	0013C		160	DS 15, 00
000554	5553 F134	0013C		161	DS 15, 00
000558	47F3 F0A0	0013C		162	DS 15, 00
000560	5553 F134	0013C		163	DS 15, 00
000564	47F3 F0A0	0013C		164	DS 15, 00
000568	5553 F134	0013C		165	DS 15, 00
000570	47F3 F0A0	0013C		166	DS 15, 00
000574	5553 F134	0013C		167	DS 15, 00
000578	47F3 F0A0	0013C			

LCC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

00015C F1F2A171			111	B	DC	X*F1F2A021*
00016C F21F229D			112		DC	X*F21F229D*
00016A F2AC910C			113		DC	X*F2AC910C*
00016B F2C11730			114		DC	X*F2C11730*
00016D F2F3F518			115		DC	X*F2F3F518*
00017C F307A78R			116		DC	X*F307A78R*
00017A F317F405			117		DC	X*F317F405*
00017B F32A185C			118		DC	X*F32A185C*
00017C F327F6A5			119		DC	X*F327F6A5*
00018D F330751C			120		DC	X*F330751C*
00018A F427154E			121		DC	X*F427154E*
00018B F427154E			122		DC	X*F427154E*
00019C F4F4220F			123		DC	X*F4F4220F*
00019D F55C2594			124		DC	X*F55C2594*
00019A F60A7D7			125		DC	X*F60A7D7*
00019C F65C7C9F			126		DC	X*F65C7C9F*
00019D F6A471A9			127		DC	X*F6A471A9*
00019A F6A471A9			128		DC	X*F6A471A9*
00019B F729F734			129		DC	X*F729F734*
00019C F7951C8D			130		DC	X*F7951C8D*
00019D F7951C8D			131		DC	X*F7951C8D*
00019A F7F1E112			132		DC	X*F7F1E112*
00019B F7F1E112			133		DC	X*F7F1E112*
00019C F7F1E112			134		DC	X*F7F1E112*
00019D F7F1E112			135		DC	X*F7F1E112*
00019A F84976A8			136		DC	X*F84976A8*
00019B F84976A8			137		DC	X*F84976A8*
00019C F84976A8			138		DC	X*F84976A8*
00019D F84976A8			139		DC	X*F84976A8*
00019A F84976A8			140		DC	X*F84976A8*
00019B F84976A8			141		DC	X*F84976A8*
00019C F84976A8			142		DC	X*F84976A8*
00019D F84976A8			143		DC	X*F84976A8*
00019A F84976A8			144		DC	X*F84976A8*
00019B F84976A8			145		DC	X*F84976A8*
00019C F84976A8			146		DC	X*F84976A8*
00019D F84976A8			147		DC	X*F84976A8*
00019A F84976A8			148		DC	X*F84976A8*
00019B F84976A8			149		DC	X*F84976A8*
00019C F84976A8			150		DC	X*F84976A8*
00019D F84976A8			151		DC	X*F84976A8*
00019A F84976A8			152		DC	X*F84976A8*
00019B F84976A8			153		DC	X*F84976A8*
00019C F84976A8			154		DC	X*F84976A8*
00019D F84976A8			155		DC	X*F84976A8*
00019A F84976A8			156		DC	X*F84976A8*
00019B F84976A8			157		DC	X*F84976A8*
00019C F84976A8			158		DC	X*F84976A8*
00019D F84976A8			159		DC	X*F84976A8*
00019A F84976A8			160		DC	X*F84976A8*
00019B F84976A8			161		DC	X*F84976A8*
00019C F84976A8			162		DC	X*F84976A8*
00019D F84976A8			163		DC	X*F84976A8*
00019A F84976A8			164		DC	X*F84976A8*
00019B F84976A8			165		DC	X*F84976A8*

LDC	OBJECT CODE	ACDR1 ADDR2	SYMT	SOURCE	STATEMENT
000218	F080043A		166	DC	BASE 10 0.9857295160
000219	F010158A		167	DC	BASE 10 0.9807711667
000220	F010249A		168	DC	BASE 10 0.9809441911
000221	F02010CA		169	DC	BASE 10 0.9810791359
000222	F0302F11		170	DC	BASE 10 0.9812658742
000223	F0303A19		171	DC	BASE 10 0.9813940210
000224	F040484A		172	DC	BASE 10 0.9815146019
000225	F0404A7C		173	DC	BASE 10 0.9816124607
000226	F0404A91		174	DC	BASE 10 0.9816943811
000227	F1705FD		175	DC	BASE 10 0.9832340080
000228	F1705FD		176	DC	BASE 10 0.9832340080
000229	F1705FD		177	DC	BASE 10 0.9832340080
000230	F2402F8A		178	DC	BASE 10 0.9832340080
000231	F2402F8A		179	DC	BASE 10 0.9832340080
000232	F2402F8A		180	DC	BASE 10 0.9832340080
000233	F2402F8A		181	DC	BASE 10 0.9832340080
000234	F2402F8A		182	DC	BASE 10 0.9832340080
000235	F2402F8A		183	DC	BASE 10 0.9832340080
000236	F2402F8A		184	DC	BASE 10 0.9832340080
000237	F2402F8A		185	DC	BASE 10 0.9832340080
000238	F2402F8A		186	DC	BASE 10 0.9832340080
000239	F2402F8A		187	DC	BASE 10 0.9832340080
000240	F2402F8A		188	DC	BASE 10 0.9832340080
000241	F2402F8A		189	DC	BASE 10 0.9832340080
000242	F2402F8A		190	DC	BASE 10 0.9832340080
000243	F2402F8A		191	DC	BASE 10 0.9832340080
000244	F2402F8A		192	DC	BASE 10 0.9832340080
000245	F2402F8A		193	DC	BASE 10 0.9832340080
000246	F2402F8A		194	DC	BASE 10 0.9832340080
000247	F2402F8A		195	DC	BASE 10 0.9832340080
000248	F2402F8A		196	DC	BASE 10 0.9832340080
000249	F2402F8A		197	DC	BASE 10 0.9832340080
000250	F2402F8A		198	DC	BASE 10 0.9832340080
000251	F2402F8A		199	DC	BASE 10 0.9832340080
000252	F2402F8A		200	DC	BASE 10 0.9832340080
000253	F2402F8A		201	DC	BASE 10 0.9832340080
000254	F2402F8A		202	DC	BASE 10 0.9832340080
000255	F2402F8A		203	DC	BASE 10 0.9832340080
000256	F2402F8A		204	DC	BASE 10 0.9832340080
000257	F2402F8A		205	DC	BASE 10 0.9832340080
000258	F2402F8A		206	DC	BASE 10 0.9832340080
000259	F2402F8A		207	DC	BASE 10 0.9832340080
000260	F2402F8A		208	DC	BASE 10 0.9832340080
000261	F2402F8A		209	DC	BASE 10 0.9832340080
000262	F2402F8A		210	DC	BASE 10 0.9832340080
000263	F2402F8A		211	DC	BASE 10 0.9832340080
000264	F2402F8A		212	DC	BASE 10 0.9832340080
000265	F2402F8A		213	DC	BASE 10 0.9832340080
000266	F2402F8A		214	DC	BASE 10 0.9832340080
000267	F2402F8A		215	DC	BASE 10 0.9832340080
000268	F2402F8A		216	DC	BASE 10 0.9832340080
000269	F2402F8A		217	DC	BASE 10 0.9832340080
000270	F2402F8A		218	DC	BASE 10 0.9832340080
000271	F2402F8A		219	DC	BASE 10 0.9832340080
000272	F2402F8A		220	DC	BASE 10 0.9832340080

PAGE 5

FOIAPR70 6/10/71

LOC	OBJECT CODE	ACOR1 ADDR2	STMT	SOURCE STATEMENT	BASE 10
000314	FAT1F667		221	DC	X*FAT1F667
000315	FAT1F667		222	DC	X*FAT1F667
000316	FAT1F667		223	DC	X*FAT1F667
000317	FAT1F667		224	DC	X*FAT1F667
000318	FAT1F667		225	DC	X*FAT1F667
000319	FAT1F667		226	DC	X*FAT1F667
000320	FAT1F667		227	DC	X*FAT1F667
000321	FAT1F667		228	DC	X*FAT1F667
000322	FAT1F667		229	DC	X*FAT1F667
000323	FAT1F667		230	DC	X*FAT1F667
000324	FAT1F667		231	DC	X*FAT1F667
000325	FAT1F667		232	DC	X*FAT1F667
000326	FAT1F667		233	DC	X*FAT1F667
000327	FAT1F667		234	DC	X*FAT1F667
000328	FAT1F667		235	DC	X*FAT1F667
000329	FAT1F667		236	DC	X*FAT1F667
000330	FAT1F667		237	DC	X*FAT1F667
000331	FAT1F667		238	DC	X*FAT1F667
000332	FAT1F667		239	DC	X*FAT1F667
000333	FAT1F667		240	DC	X*FAT1F667
000334	FAT1F667		241	DC	X*FAT1F667
000335	FAT1F667		242	DC	X*FAT1F667
000336	FAT1F667		243	DC	X*FAT1F667
000337	FAT1F667		244	DC	X*FAT1F667
000338	FAT1F667		245	DC	X*FAT1F667
000339	FAT1F667		246	DC	X*FAT1F667
000340	FAT1F667		247	DC	X*FAT1F667
000341	FAT1F667		248	DC	X*FAT1F667
000342	FAT1F667		249	DC	X*FAT1F667
000343	FAT1F667		250	DC	X*FAT1F667
000344	FAT1F667		251	DC	X*FAT1F667
000345	FAT1F667		252	DC	X*FAT1F667
000346	FAT1F667		253	DC	X*FAT1F667
000347	FAT1F667		254	DC	X*FAT1F667
000348	FAT1F667		255	DC	X*FAT1F667
000349	FAT1F667		256	DC	X*FAT1F667
000350	FAT1F667		257	DC	X*FAT1F667
000351	FAT1F667		258	DC	X*FAT1F667
000352	FAT1F667		259	DC	X*FAT1F667
000353	FAT1F667		260	DC	X*FAT1F667
000354	FAT1F667		261	DC	X*FAT1F667
000355	FAT1F667		262	DC	X*FAT1F667
000356	FAT1F667		263	DC	X*FAT1F667
000357	FAT1F667		264	DC	X*FAT1F667
000358	FAT1F667		265	DC	X*FAT1F667
000359	FAT1F667		266	DC	X*FAT1F667
000360	FAT1F667		267	DC	X*FAT1F667
000361	FAT1F667		268	DC	X*FAT1F667
000362	FAT1F667		269	DC	X*FAT1F667
000363	FAT1F667		270	DC	X*FAT1F667
000364	FAT1F667		271	DC	X*FAT1F667
000365	FAT1F667		272	DC	X*FAT1F667
000366	FAT1F667		273	DC	X*FAT1F667
000367	FAT1F667		274	DC	X*FAT1F667
000368	FAT1F667		275	DC	X*FAT1F667

PAGE A

-FJAAPRTG 6/30/71

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	BASE	BASE
000393	17			276	DC	X*17*	BASE 10 1.4375
000394	1A			277	DC	X*18*	BASE 10 1.6875
000395	1A			278	DC	X*19*	BASE 10 1.6250
000396	23			279	DC	X*23*	BASE 10 2.1875
000397	1F			280	DC	X*1F*	BASE 10 1.9375
000398	22			281	DC	X*22*	BASE 10 2.1250
000399	25			282	DC	X*25*	BASE 10 2.3125
000400	21			283	DC	X*21*	BASE 10 2.0625
000401	1D			284	DC	X*1D*	BASE 10 1.8125
000402	28			285	DC	X*28*	BASE 10 2.5000
000403	10			286	DC	X*10*	BASE 10 3.0000
000404	33			287	DC	X*33*	BASE 10 3.1875
000405	2F			288	DC	X*2F*	BASE 10 2.2500
000406	3E			289	DC	X*3E*	BASE 10 2.9375
000407	29			290	DC	X*29*	BASE 10 3.8750
000408	26			291	DC	X*26*	BASE 10 2.5625
000409	19			292	DC	X*19*	BASE 10 2.3750
000410	2E			293	DC	X*2E*	BASE 10 3.5625
000411	37			294	DC	X*37*	BASE 10 2.8750
000412	2A			295	DC	X*2A*	BASE 10 3.4375
000413	26			296	DC	X*26*	BASE 10 2.6250
000414	32			297	DC	X*32*	BASE 10 3.1250
000415	2C			298	DC	X*2C*	BASE 10 2.8125
000416	2C			299	DC	X*2C*	BASE 10 2.7500
000417	2F			300	DC	X*2F*	BASE 10 3.9375
000418	3C			301	DC	X*3C*	BASE 10 3.7500
000419	27			302	DC	X*27*	BASE 10 2.4375
000420	3A			303	DC	X*3A*	BASE 10 3.2500
000421	3A			304	DC	X*3A*	BASE 10 3.6250
000422	18			305	DC	X*18*	BASE 10 3.0625
000423	10			306	DC	X*10*	BASE 10 3.5000
000424	35			307	DC	X*35*	BASE 10 3.8125
000425	02			308	DC	X*02*	BASE 10 3.6875
000426	03			309	DC	X*03*	BASE 10 0.1250
000427	03			310	DC	X*03*	BASE 10 0.1875
000428	03			311	DC	X*03*	BASE 10 0.1875
000429	03			312	DC	X*03*	BASE 10 0.1875
000430	03			313	DC	X*03*	BASE 10 0.1875
000431	03			314	DC	X*03*	BASE 10 0.1875
000432	03			315	DC	X*03*	BASE 10 0.1875
000433	03			316	DC	X*03*	BASE 10 0.1875
000434	03			317	DC	X*03*	BASE 10 0.1875
000435	03			318	DC	X*03*	BASE 10 0.1875
000436	03			319	DC	X*03*	BASE 10 0.1875
000437	03			320	DC	X*03*	BASE 10 0.1875
000438	03			321	DC	X*03*	BASE 10 0.1875
000439	03			322	DC	X*03*	BASE 10 0.1875
000440	03			323	DC	X*03*	BASE 10 0.1875
000441	03			324	DC	X*03*	BASE 10 0.1875
000442	03			325	DC	X*03*	BASE 10 0.1875
000443	03			326	DC	X*03*	BASE 10 0.1875
000444	03			327	DC	X*03*	BASE 10 0.1875
000445	03			328	DC	X*03*	BASE 10 0.1875
000446	03			329	DC	X*03*	BASE 10 0.1875
000447	03			330	DC	X*03*	BASE 10 0.1875

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
000100A 21				331	DC	X121*
000100A 21				332	DC	X121*
000100C 10				333	DC	X110*
000100C 10				334	DC	X110*
000100C 10				335	DC	X110*
000100C 2F				336	DC	X12F*
000100C 2F				337	DC	X12F*
000100C 3E				338	DC	X13F*
000100C 3E				339	DC	X13F*
000100C 3F				340	DC	X13F*
000100C 26				341	DC	X126*
000100C 26				342	DC	X126*
000100C 26				343	DC	X126*
000100C 39				344	DC	X139*
000100C 39				345	DC	X139*
000100C 39				346	DC	X139*
000100C 39				347	DC	X139*
000100C 39				348	DC	X139*
000100C 2E				349	DC	X12E*
000100C 2E				350	DC	X12E*
000100C 2E				351	DC	X12E*
000100C 37				352	DC	X137*
000100C 37				353	DC	X137*
000100C 37				354	DC	X137*
000100C 37				355	DC	X137*
000100C 37				356	DC	X137*
000100C 37				357	DC	X137*
000100C 20				358	DC	X120*
000100C 20				359	DC	X120*
000100C 20				360	DC	X120*
000100C 20				361	DC	X120*
000100C 20				362	DC	X120*
000100C 20				363	DC	X120*
000100C 20				364	DC	X120*
000100C 20				365	DC	X120*
000100C 20				366	DC	X120*
000100C 3F				367	DC	X13F*
000100C 3F				368	DC	X13F*
000100C 3F				369	DC	X13F*
000100C 3F				370	DC	X13F*
000100C 3F				371	DC	X13F*
000100C 3F				372	DC	X13F*
000100C 3F				373	DC	X13F*
000100C 3F				374	DC	X13F*
000100C 3F				375	DC	X13F*
000100C 3A				376	DC	X13A*
000100C 3A				377	DC	X13A*
000100C 3A				378	DC	X13A*
000100C 3A				379	DC	X13A*
000100C 3A				380	DC	X13A*
000100C 3A				381	DC	X13A*
000100C 3A				382	DC	X13A*
000100C 3A				383	DC	X13A*
000100C 3A				384	DC	X13A*
000100C 3A				385	DC	X13A*

PAGE 9

F08APR70 6/30/71

LDC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	BASE	IO
000431	39			386	DC X'18'	BASE	10 3.5000
000432	38			387	DC X'18'	BASE	10 3.5000
000433	30			388	DC X'3D'	BASE	10 3.8125
000434	10			389	DC X'1D'	BASE	10 3.8125
000435	30			390	DC X'1D'	BASE	10 3.8125
000436	10			391	DC X'3D'	BASE	10 3.8125
000437	38			392	DC X'18'	BASE	10 3.8125
000438	38			393	DC X'3B'	BASE	10 3.8125
000439	18			394	DC X'3B'	BASE	10 3.8125
000440	18			395	DC X'3B'	BASE	10 3.8125
000441	30			396	DC X'18'	BASE	10 3.8125
000442	02			397	DC X'02'	BASE	10 0.1250
000443	02			398	DC X'02'	BASE	10 0.1250
000444	C5			399	DC X'05'	BASE	10 0.1250
000445	C5			400	DC X'05'	BASE	10 0.1250
000446	C5			401	DC X'05'	BASE	10 0.1250
000447	C5			402	DC X'05'	BASE	10 0.1250
000448	C5			403	DC X'05'	BASE	10 0.1250
000449	C4			404	DC X'04'	BASE	10 0.2500
000450	C4			405	DC X'04'	BASE	10 0.2500
000451	C6			406	DC X'06'	BASE	10 0.3750
000452	C6			407	DC X'06'	BASE	10 0.3750
000453	0A			408	DC X'0A'	BASE	10 0.3750
000454	CE			409	DC X'0E'	BASE	10 0.3750
000455	07			410	DC X'07'	BASE	10 0.3750
000456	07			411	DC X'07'	BASE	10 0.3750
000457	CD			412	DC X'0D'	BASE	10 0.4375
000458	0F			413	DC X'0F'	BASE	10 0.4375
000459	CE			414	DC X'0E'	BASE	10 0.4375
000460	CF			415	DC X'0F'	BASE	10 0.4375
000461	0F			416	DC X'0F'	BASE	10 0.4375
000462	0F			417	DC X'0F'	BASE	10 0.4375
000463	CF			418	DC X'0F'	BASE	10 0.4375
000464	11			419	DC X'11'	BASE	10 1.0625
000465	11			420	DC X'11'	BASE	10 1.0625
000466	16			421	DC X'16'	BASE	10 1.0625
000467	16			422	DC X'16'	BASE	10 1.0625
000468	16			423	DC X'16'	BASE	10 1.0625
000469	12			424	DC X'12'	BASE	10 1.1250
000470	12			425	DC X'12'	BASE	10 1.1250
000471	12			426	DC X'12'	BASE	10 1.1250
000472	18			427	DC X'18'	BASE	10 1.5000
000473	18			428	DC X'18'	BASE	10 1.5000
000474	1F			429	DC X'1F'	BASE	10 1.5000
000475	1F			430	DC X'1F'	BASE	10 1.5000
000476	1E			431	DC X'1E'	BASE	10 1.5000
000477	19			432	DC X'19'	BASE	10 1.5625
000478	18			433	DC X'18'	BASE	10 1.5625
000479	1A			434	DC X'1A'	BASE	10 1.6875
000480	1A			435	DC X'1A'	BASE	10 1.6875
000481	1A			436	DC X'1A'	BASE	10 1.6875
000482	1A			437	DC X'1A'	BASE	10 1.6875
000483	1A			438	DC X'1A'	BASE	10 1.6875
000484	1A			439	DC X'1A'	BASE	10 1.6875
000485	1A			440	DC X'1A'	BASE	10 1.6875

PAGE 9

, FOBA PR70 6/30/71

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	BASE	LOC
000418 23				441	DC	BASE 10	2-1875
000419 1F				442	DC	BASE 10	1-9375
000420 22				443	DC	BASE 10	2-1250
000421 22				444	DC	BASE 10	2-1250
000422 22				445	DC	BASE 10	2-1250
000423 25				446	DC	BASE 10	2-3125
000424 25				447	DC	BASE 10	2-3125
000425 25				448	DC	BASE 10	2-3125
000426 21				449	DC	BASE 10	2-0625
000427 21				450	DC	BASE 10	2-0625
000428 21				451	DC	BASE 10	2-0625
000429 10				452	DC	BASE 10	1-8125
000430 10				453	DC	BASE 10	1-8125
000431 10				454	DC	BASE 10	1-8125
000432 10				455	DC	BASE 10	1-8125
000433 28				456	DC	BASE 10	2-5000
000434 28				457	DC	BASE 10	2-5000
000435 30				458	DC	BASE 10	3-0000
000436 2F				459	DC	BASE 10	2-9175
000437 29				460	DC	BASE 10	2-5625
000438 26				461	DC	BASE 10	2-3750
000439 2F				462	DC	BASE 10	2-3750
000440 2E				463	DC	BASE 10	2-8750
000441 32				464	DC	BASE 10	3-1250
000442 20				465	DC	BASE 10	2-8125
000443 2C				466	DC	BASE 10	2-7500
000444 27				467	DC	BASE 10	2-4375
000445 31				468	DC	BASE 10	2-625
000446 0C				469	DC	BASE 10	0-0000
000447 0C				470	DC	BASE 10	0-0000
000448 0C				471	DC	BASE 10	0-0000
000449 01				472	DC	BASE 10	0-0625
000450 04				473	DC	BASE 10	0-2500
000451 04				474	DC	BASE 10	0-2500
000452 04				475	DC	BASE 10	0-2500
000453 08				476	DC	BASE 10	0-6875
000454 08				477	DC	BASE 10	0-6875
000455 08				478	DC	BASE 10	0-6875
000456 08				479	DC	BASE 10	0-6875
000457 08				480	DC	BASE 10	0-6875
000458 08				481	DC	BASE 10	0-6875
000459 0C				482	DC	BASE 10	0-7500
000460 0C				483	DC	BASE 10	0-7500
000461 0C				484	DC	BASE 10	0-7500
000462 0C				485	DC	BASE 10	0-7500
000463 0C				486	DC	BASE 10	0-7500
000464 0C				487	DC	BASE 10	0-7500
000465 0C				488	DC	BASE 10	0-7500
000466 0D				489	DC	BASE 10	0-8125
000467 0D				490	DC	BASE 10	0-8125
000468 0D				491	DC	BASE 10	0-8125
000469 10				492	DC	BASE 10	1-0000
000470 10				493	DC	BASE 10	1-0000
000471 16				494	DC	BASE 10	1-3750
000472 16				495	DC	BASE 10	1-3750
000473 16				496	DC	BASE 10	1-3750

PAGE 10

FORMAPTC 6/30/73

LTC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	BASE IN	BASE IN
00046F 18	DC	X*19*	496	DC	X*19*	BASE IN	1-9000
000470 1A	DC	X*1A*	497	DC	X*1A*	BASE IN	1-5000
000471 1C	DC	X*1C*	498	DC	X*1C*	BASE IN	1-7500
000472 1C	DC	X*1C*	499	DC	X*1C*	BASE IN	1-7500
000473 1C	DC	X*1C*	500	DC	X*1C*	BASE IN	1-7500
000474 1C	DC	X*1C*	501	DC	X*1C*	BASE IN	1-7500
000475 1C	DC	X*1C*	502	DC	X*1C*	BASE IN	1-7500
000476 1E	DC	X*1E*	503	DC	X*1E*	BASE IN	1-7500
000477 1E	DC	X*1E*	504	DC	X*1E*	BASE IN	1-7500
000478 1E	DC	X*1E*	505	DC	X*1E*	BASE IN	1-7500
000479 1E	DC	X*1E*	506	DC	X*1E*	BASE IN	1-7500
00047A 19	DC	X*19*	507	DC	X*19*	BASE IN	1-7500
00047B 19	DC	X*19*	508	DC	X*19*	BASE IN	1-7500
00047C 17	DC	X*17*	509	DC	X*17*	BASE IN	1-7500
00047D 17	DC	X*17*	510	DC	X*17*	BASE IN	1-7500
00047E 17	DC	X*17*	511	DC	X*17*	BASE IN	1-7500
00047F 17	DC	X*17*	512	DC	X*17*	BASE IN	1-7500
000480 17	DC	X*17*	513	DC	X*17*	BASE IN	1-7500
000481 17	DC	X*17*	514	DC	X*17*	BASE IN	1-7500
000482 17	DC	X*17*	515	DC	X*17*	BASE IN	1-7500
000483 1A	DC	X*1A*	516	DC	X*1A*	BASE IN	1-7500
000484 1A	DC	X*1A*	517	DC	X*1A*	BASE IN	1-7500
000485 23	DC	X*23*	518	DC	X*23*	BASE IN	1-7500
000486 23	DC	X*23*	519	DC	X*23*	BASE IN	1-7500
000487 1F	DC	X*1F*	520	DC	X*1F*	BASE IN	1-7500
000488 1F	DC	X*1F*	521	DC	X*1F*	BASE IN	1-7500
000489 1F	DC	X*1F*	522	DC	X*1F*	BASE IN	1-7500
000490 20	DC	X*20*	523	DC	X*20*	BASE IN	1-7500
000491 20	DC	X*20*	524	DC	X*20*	BASE IN	1-7500
000492 20	DC	X*20*	525	DC	X*20*	BASE IN	1-7500
000493 33	DC	X*33*	526	DC	X*33*	BASE IN	1-7500
000494 24	DC	X*24*	527	DC	X*24*	BASE IN	1-7500
000495 24	DC	X*24*	528	DC	X*24*	BASE IN	1-7500
000496 24	DC	X*24*	529	DC	X*24*	BASE IN	1-7500
000497 29	DC	X*29*	530	DC	X*29*	BASE IN	1-7500
000498 35	DC	X*35*	531	DC	X*35*	BASE IN	1-7500
000499 2A	DC	X*2A*	532	DC	X*2A*	BASE IN	1-7500
000500 2A	DC	X*2A*	533	DC	X*2A*	BASE IN	1-7500
000501 2A	DC	X*2A*	534	DC	X*2A*	BASE IN	1-7500
000502 2A	DC	X*2A*	535	DC	X*2A*	BASE IN	1-7500
000503 2A	DC	X*2A*	536	DC	X*2A*	BASE IN	1-7500
000504 34	DC	X*34*	537	DC	X*34*	BASE IN	1-7500
000505 36	DC	X*36*	538	DC	X*36*	BASE IN	1-7500
000506 36	DC	X*36*	539	DC	X*36*	BASE IN	1-7500
000507 36	DC	X*36*	540	DC	X*36*	BASE IN	1-7500
000508 36	DC	X*36*	541	DC	X*36*	BASE IN	1-7500
000509 36	DC	X*36*	542	DC	X*36*	BASE IN	1-7500
000510 36	DC	X*36*	543	DC	X*36*	BASE IN	1-7500
000511 36	DC	X*36*	544	DC	X*36*	BASE IN	1-7500
000512 36	DC	X*36*	545	DC	X*36*	BASE IN	1-7500
000513 36	DC	X*36*	546	DC	X*36*	BASE IN	1-7500
000514 36	DC	X*36*	547	DC	X*36*	BASE IN	1-7500
000515 36	DC	X*36*	548	DC	X*36*	BASE IN	1-7500
000516 36	DC	X*36*	549	DC	X*36*	BASE IN	1-7500
000517 36	DC	X*36*	550	DC	X*36*	BASE IN	1-7500

LCC	OBJECT CODE	ADDR1	ADDR2	SYMT	SOURCE	STATEMENT
0004A6	09			551	DC	X'09'
0004A7	CC			552	DC	X'0C'
0004A8	0D			553	DC	X'0D'
0004A9	CE			554	DC	X'0E'
0004AA	10			555	DC	X'10'
0004AB	13			556	DC	X'13'
0004AC	02			557	DC	X'02'
0004AD	02			558	DC	X'02'
0004AE	02			559	DC	X'02'
0004AF	03			560	DC	X'03'
0004B0	03			561	DC	X'03'
0004B1	03			562	DC	X'03'
0004B2	05			563	DC	X'05'
0004B3	05			564	DC	X'05'
0004B4	C6			565	DC	X'06'
0004B5	0A			566	DC	X'06'
0004B6	C7			567	DC	X'07'
0004B7	0A			568	DC	X'0A'
0004B8	0A			569	DC	X'0A'
0004B9	C9			570	DC	X'09'
0004BA	CF			571	DC	X'0F'
0004BB	14			572	DC	X'14'
0004BC	15			573	DC	X'15'
0004BD	11			574	DC	X'11'
0004BE	12			575	DC	X'12'
				576	END	
0004C0	00000025-			577		=F'37'
0004C6	000001C8			578		=F'456'
0004CA	000003F17			579		=F'3863'
0004CC	00000537			580		=F'3639'

NOT REPRODUCIBLE

APPENDIX C

Assembler Listing of GEN2

LOC	ACT CODE	ADDR1 ADDR2	SYMT	SOURCE STATEMENT	FORAPR70
000000			1	GEN2 START 1	
000001	07	000000	2		
000002	07	000000	3	SOURCE---L.F. GAMMA, UKA, DEPT. OF STAT., APR 1970	
000003	07	000000	4	PURPOSE---TO SUPPLY THE USER WITH A FAST PROCEDURE FOR GENERATING P- MONETIAL AND GAMMA DISTRIBUTED RANDOM NUMBERS.	
000004	07	000000	5	USAGE-----RANGAM(1,N), N MUST BE AN INTEGER, IT PRINTS THE GENERATING SCHEME. P0 IS THE NUMBER OF EXPONENTIAL RANDOM NUMBERS SUMMED IN GENERATING A RANDOM NUMBER THAT IS GAMMA (1,N,0,0,1).	
000005	07	000000	6	EXAMPLE: RANGAM(550,50) P0 WILL BE GAMMA(550,N=1).	
000006	07	000000	7	NOTE-----IF N01 AN EXPONENTIALLY DISTRIBUTED RANDOM NUMBER WILL BE GENERATED.	
000007	07	000000	8	METHOD---THE METHOD IS THAT OF MARSAGLIA AS DESCRIBED IN "COMPUTER- CALCULUS OF ACM, VOL. 7, NO. 5, MAY 1964. THE PROGRAM OCCUR- RING WILL FOLLOW THE NOTATION USED BY MARSAGLIA.	
000008	07	000000	9	ENTRY RANGAM PROLOGUE, STANDARD FOR 360/AS.	
000009	07	000000	10	USING RANGAM, 15	
000010	07	000000	11	DC PL1(07)	
000011	07	000000	12	CL7: RANGAM	
000012	07	000000	13	14,12,12(13)	
000013	07	000000	14	11,4(11)	
000014	07	000000	15	11,0(11)	
000015	07	000000	16	12,1	
000016	07	000000	17	1A 10,1	
000017	07	000000	18	1A 10,1	
000018	07	000000	19	1A 10,1	
000019	07	000000	20	1A 10,1	
000020	07	000000	21	1A 10,1	
000021	07	000000	22	1A 10,1	
000022	07	000000	23	1A 10,1	
000023	07	000000	24	1A 10,1	
000024	07	000000	25	1A 10,1	
000025	07	000000	26	1A 10,1	
000026	07	000000	27	1A 10,1	
000027	07	000000	28	1A 10,1	
000028	07	000000	29	1A 10,1	
000029	07	000000	30	1A 10,1	
000030	07	000000	31	1A 10,1	
000031	07	000000	32	1A 10,1	
000032	07	000000	33	1A 10,1	
000033	07	000000	34	1A 10,1	
000034	07	000000	35	1A 10,1	
000035	07	000000	36	1A 10,1	
000036	07	000000	37	1A 10,1	
000037	07	000000	38	1A 10,1	
000038	07	000000	39	1A 10,1	
000039	07	000000	40	1A 10,1	
000040	07	000000	41	1A 10,1	
000041	07	000000	42	1A 10,1	
000042	07	000000	43	1A 10,1	
000043	07	000000	44	1A 10,1	
000044	07	000000	45	1A 10,1	
000045	07	000000	46	1A 10,1	
000046	07	000000	47	1A 10,1	
000047	07	000000	48	1A 10,1	
000048	07	000000	49	1A 10,1	
000049	07	000000	50	1A 10,1	
000050	07	000000	51	1A 10,1	
000051	07	000000	52	1A 10,1	
000052	07	000000	53	1A 10,1	
000053	07	000000	54	1A 10,1	
000054	07	000000	55	1A 10,1	

PAGE 3

F08APP70 6/30/71

LOC	INJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000148				111 LA	DS 1F
000149				112 J	DS 1F
000150	4PC27195			113 RPT	DC XL449C27195*
000151	00000000			114 Q	DC X00000000*
000152	00000000			115	DC X00000000*
000153	PARICFA7			116	DC X00000000*
000154	FEFA7A4A			117	DC X00000000*
000155	FEFA7A4A			118	DC X00000000*
000156	FEFA7A4A			119	DC X00000000*
000157	FEFA7A4A			120	DC X00000000*
000158	FEFA7A4A			121	DC X00000000*
000159	FEFA7A4A			122	DC X00000000*
000160	FEFA7A4A			123	DC X00000000*
000161	FEFA7A4A			124	DC X00000000*
000162	FEFA7A4A			125	DC X00000000*
000163	FEFA7A4A			126	DC X00000000*
000164	FEFA7A4A			127	DC X00000000*
000165	FEFA7A4A			128	DC X00000000*
000166	FEFA7A4A			129	DC X00000000*
000167	FEFA7A4A			130	DC X00000000*
000168	FEFA7A4A			131	DC X00000000*
000169	FEFA7A4A			132	DC X00000000*
000170	FEFA7A4A			133	DC X00000000*
000171	FEFA7A4A			134	DC X00000000*
000172	FEFA7A4A			135	DC X00000000*
000173	FEFA7A4A			136	DC X00000000*
000174	FEFA7A4A			137	DC X00000000*
000175	FEFA7A4A			138	DC X00000000*
000176	FEFA7A4A			139	DC X00000000*
000177	FEFA7A4A			140	DC X00000000*
000178	FEFA7A4A			141	DC X00000000*
000179	FEFA7A4A			142	DC X00000000*
000180	FEFA7A4A			143	DC X00000000*
000181	FEFA7A4A			144	DC X00000000*
000182	FEFA7A4A			145	DC X00000000*
000183	FEFA7A4A			146	DC X00000000*
000184	FEFA7A4A			147	DC X00000000*
000185	FEFA7A4A			148	DC X00000000*
000186	FEFA7A4A			149	DC X00000000*
000187	FEFA7A4A			150	DC X00000000*
000188	FEFA7A4A			151	DC X00000000*
000189	FEFA7A4A			152	DC X00000000*
000190	FEFA7A4A			153	DC X00000000*
000191	FEFA7A4A			154	DC X00000000*
000192	FEFA7A4A			155	DC X00000000*
000193	FEFA7A4A			156	DC X00000000*
000194	FEFA7A4A			157	DC X00000000*
000195	FEFA7A4A			158	DC X00000000*
000196	FEFA7A4A			159	DC X00000000*
000197	FEFA7A4A			160	DC X00000000*
000198	FEFA7A4A			161	DC X00000000*
000199	FEFA7A4A			162	DC X00000000*
000200	FEFA7A4A			163	DC X00000000*
000201	FEFA7A4A			164	DC X00000000*
000202	FEFA7A4A			165	DC X00000000*

PAGE 4

FOBAPR75 6/30/71

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
000224	F450771			166	DC	XIF450771
000225	F450771			167	DC	XIF450771
000226	F450771			168	DC	XIF450771
000227	F450771			169	DC	XIF450771
000228	F450771			170	DC	XIF450771
000229	F450771			171	DC	XIF450771
000230	F450771			172	DC	XIF450771
000231	F450771			173	DC	XIF450771
000232	F450771			174	DC	XIF450771
000233	F450771			175	DC	XIF450771
000234	F450771			176	DC	XIF450771
000235	F450771			177	DC	XIF450771
000236	F450771			178	DC	XIF450771
000237	F450771			179	DC	XIF450771
000238	F450771			180	DC	XIF450771
000239	F450771			181	DC	XIF450771
000240	F450771			182	DC	XIF450771
000241	F450771			183	DC	XIF450771
000242	F450771			184	DC	XIF450771
000243	F450771			185	DC	XIF450771
000244	F450771			186	DC	XIF450771
000245	F450771			187	DC	XIF450771
000246	F450771			188	DC	XIF450771
000247	F450771			189	DC	XIF450771
000248	F450771			190	DC	XIF450771
000249	F450771			191	DC	XIF450771
000250	F450771			192	DC	XIF450771
000251	F450771			193	DC	XIF450771
000252	F450771			194	DC	XIF450771
000253	F450771			195	DC	XIF450771
000254	F450771			196	DC	XIF450771
000255	F450771			197	DC	XIF450771
000256	F450771			198	DC	XIF450771
000257	F450771			199	DC	XIF450771
000258	F450771			200	DC	XIF450771
000259	F450771			201	DC	XIF450771
000260	F450771			202	DC	XIF450771
000261	F450771			203	DC	XIF450771
000262	F450771			204	DC	XIF450771
000263	F450771			205	DC	XIF450771
000264	F450771			206	DC	XIF450771
000265	F450771			207	DC	XIF450771
000266	F450771			208	DC	XIF450771
000267	F450771			209	DC	XIF450771
000268	F450771			210	DC	XIF450771
000269	F450771			211	DC	XIF450771
000270	F450771			212	DC	XIF450771
000271	F450771			213	DC	XIF450771
000272	F450771			214	DC	XIF450771
000273	F450771			215	DC	XIF450771
000274	F450771			216	DC	XIF450771
000275	F450771			217	DC	XIF450771
000276	F450771			218	DC	XIF450771
000277	F450771			219	DC	XIF450771
000278	F450771			220	DC	XIF450771

PAGE 5

FORADPTO 6/30/71

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

000100	F5755995			221	NC	X*F9755995	RASE 10 0.97458419708
000104	F5999911			222	NC	X*F9999911	RASE 10 0.97401048530
000109	F97401A			223	DC	X*F97401A	RASE 10 0.97545242719
00010C	F9C99AC			224	DC	X*F9C99AC	RASE 10 0.97577828654
000110	F5E05AC9			225	DC	X*F5E05AC9	RASE 10 0.97607948974
000114	F43340C4			226	DC	X*F43340C4	RASE 10 0.97656630754
000118	F413E707			227	DC	X*F413E707	RASE 10 0.97695777172
000120	F42E0A6F			228	DC	X*F42E0A6F	RASE 10 0.97727717817
000124	F4377605			229	DC	X*F4377605	RASE 10 0.97741931614
000128	F457547E			230	DC	X*F457547E	RASE 10 0.97761881458
000132	F46A9A9A			231	DC	X*F46A9A9A	RASE 10 0.977816711835
000136	F47A55E5			232	DC	X*F47A55E5	RASE 10 0.9781104768
000140	F48A73E7			233	DC	X*F48A73E7	RASE 10 0.9784552118
000144	F49A95E5			234	DC	X*F49A95E5	RASE 10 0.9787913322195
000148	F4A73E7E			235	DC	X*F4A73E7E	RASE 10 0.9791727549
000152	F4B73E7E			236	DC	X*F4B73E7E	RASE 10 0.9795429215
000156	F4C73E7E			237	DC	X*F4C73E7E	RASE 10 0.97994455327
000160	F4D73E7E			238	DC	X*F4D73E7E	RASE 10 0.98027484731
000164	F4E73E7E			239	DC	X*F4E73E7E	RASE 10 0.98061305333
000168	F4F73E7E			240	DC	X*F4F73E7E	RASE 10 0.980951267595
000172	F5073E7E			241	DC	X*F5073E7E	RASE 10 0.981293639290
000176	F5173E7E			242	DC	X*F5173E7E	RASE 10 0.98163745593
000180	F5273E7E			243	DC	X*F5273E7E	RASE 10 0.981981438111
000184	F5373E7E			244	DC	X*F5373E7E	RASE 10 0.982325453737
000188	F5473E7E			245	DC	X*F5473E7E	RASE 10 0.982669438579
000192	F5573E7E			246	DC	X*F5573E7E	RASE 10 0.9830138579
000196	F5673E7E			247	DC	X*F5673E7E	RASE 10 0.9833578745
000200	F5773E7E			248	DC	X*F5773E7E	RASE 10 0.98370189794
000204	F5873E7E			249	DC	X*F5873E7E	RASE 10 0.98404591996
000208	F5973E7E			250	DC	X*F5973E7E	RASE 10 0.9843900546
000212	F5A73E7E			251	DC	X*F5A73E7E	RASE 10 0.98473410449
000216	F5B73E7E			252	DC	X*F5B73E7E	RASE 10 0.98507815593
000220	F5C73E7E			253	DC	X*F5C73E7E	RASE 10 0.985422111
000224	F5D73E7E			254	DC	X*F5D73E7E	RASE 10 0.985766153737
000228	F5E73E7E			255	DC	X*F5E73E7E	RASE 10 0.98611018579
000232	F5F73E7E			256	DC	X*F5F73E7E	RASE 10 0.9864542111
000236	F6073E7E			257	DC	X*F6073E7E	RASE 10 0.9867982546
000240	F6173E7E			258	DC	X*F6173E7E	RASE 10 0.98714230593
000244	F6273E7E			259	DC	X*F6273E7E	RASE 10 0.987486348111
000248	F6373E7E			260	DC	X*F6373E7E	RASE 10 0.98783038579
000252	F6473E7E			261	DC	X*F6473E7E	RASE 10 0.988174428111
000256	F6573E7E			262	DC	X*F6573E7E	RASE 10 0.988518463737
000260	F6673E7E			263	DC	X*F6673E7E	RASE 10 0.98886250593
000264	F6773E7E			264	DC	X*F6773E7E	RASE 10 0.989206548111
000268	F6873E7E			265	DC	X*F6873E7E	RASE 10 0.98955058579
000272	F6973E7E			266	DC	X*F6973E7E	RASE 10 0.989894628111
000276	F6A73E7E			267	DC	X*F6A73E7E	RASE 10 0.990238663737
000280	F6B73E7E			268	DC	X*F6B73E7E	RASE 10 0.99058270593
000284	F6C73E7E			269	DC	X*F6C73E7E	RASE 10 0.990926748111
000288	F6D73E7E			270	DC	X*F6D73E7E	RASE 10 0.99127078579
000292	F6E73E7E			271	DC	X*F6E73E7E	RASE 10 0.991614828111
000296	F6F73E7E			272	DC	X*F6F73E7E	RASE 10 0.991958863737
000300	F7073E7E			273	DC	X*F7073E7E	RASE 10 0.99230290593
000304	F7173E7E			274	DC	X*F7173E7E	RASE 10 0.992646948111
000308	F7273E7E			275	DC	X*F7273E7E	RASE 10 0.99299098579
000312	F7373E7E				DC	X*F7373E7E	RASE 10 0.993335028111
000316	F7473E7E				DC	X*F7473E7E	RASE 10 0.993679063737
000320	F7573E7E				DC	X*F7573E7E	RASE 10 0.99402310593
000324	F7673E7E				DC	X*F7673E7E	RASE 10 0.994367148111
000328	F7773E7E				DC	X*F7773E7E	RASE 10 0.99471118579
000332	F7873E7E				DC	X*F7873E7E	RASE 10 0.995055228111
000336	F7973E7E				DC	X*F7973E7E	RASE 10 0.995399263737
000340	F7A73E7E				DC	X*F7A73E7E	RASE 10 0.99574330593
000344	F7B73E7E				DC	X*F7B73E7E	RASE 10 0.996087348111
000348	F7C73E7E				DC	X*F7C73E7E	RASE 10 0.99643138579
000352	F7D73E7E				DC	X*F7D73E7E	RASE 10 0.996775428111
000356	F7E73E7E				DC	X*F7E73E7E	RASE 10 0.997119463737
000360	F7F73E7E				DC	X*F7F73E7E	RASE 10 0.99746350593
000364	F8073E7E				DC	X*F8073E7E	RASE 10 0.997807548111
000368	F8173E7E				DC	X*F8173E7E	RASE 10 0.99815158579
000372	F8273E7E				DC	X*F8273E7E	RASE 10 0.998495628111
000376	F8373E7E				DC	X*F8373E7E	RASE 10 0.998839663737
000380	F8473E7E				DC	X*F8473E7E	RASE 10 0.99918370593
000384	F8573E7E				DC	X*F8573E7E	RASE 10 0.999527748111
000388	F8673E7E				DC	X*F8673E7E	RASE 10 0.99987178579
000392	F8773E7E				DC	X*F8773E7E	RASE 10 1.000215828111
000396	F8873E7E				DC	X*F8873E7E	RASE 10 1.000559863737
000400	F8973E7E				DC	X*F8973E7E	RASE 10 1.00090390593
000404	F8A73E7E				DC	X*F8A73E7E	RASE 10 1.001247948111
000408	F8B73E7E				DC	X*F8B73E7E	RASE 10 1.00159198579
000412	F8C73E7E				DC	X*F8C73E7E	RASE 10 1.001936028111
000416	F8D73E7E				DC	X*F8D73E7E	RASE 10 1.002280063737
000420	F8E73E7E				DC	X*F8E73E7E	RASE 10 1.00262410593
000424	F8F73E7E				DC	X*F8F73E7E	RASE 10 1.002968148111
000428	F9073E7E				DC	X*F9073E7E	RASE 10 1.00331218579
000432	F9173E7E				DC	X*F9173E7E	RASE 10 1.003656228111
000436	F9273E7E				DC	X*F9273E7E	RASE 10 1.004000263737
000440	F9373E7E				DC	X*F9373E7E	RASE 10 1.00434430593
000444	F9473E7E				DC	X*F9473E7E	RASE 10 1.004688348111
000448	F9573E7E				DC	X*F9573E7E	RASE 10 1.00503238579
000452	F9673E7E				DC	X*F9673E7E	RASE 10 1.005376428111
000456	F9773E7E				DC	X*F9773E7E	RASE 10 1.005720463737
000460	F9873E7E				DC	X*F9873E7E	RASE 10 1.00606450593
000464	F9973E7E				DC	X*F9973E7E	RASE 10 1.006408548111
000468	F9A73E7E				DC	X*F9A73E7E	RASE 10 1.00675258579
000472	F9B73E7E				DC	X*F9B73E7E	RASE 10 1.007096628111
000476	F9C73E7E				DC	X*F9C73E7E	RASE 10 1.007440663737
000480	F9D73E7E				DC	X*F9D73E7E	RASE 10 1.00778470593
000484	F9E73E7E				DC	X*F9E73E7E	RASE 10 1.008128748111
000488	F9F73E7E				DC	X*F9F73E7E	RASE 10 1.00847278579
000492	F9073E7E				DC	X*F9073E7E	RASE 10 1.008816828111
000496	F9173E7E				DC	X*F9173E7E	RASE 10 1.009160863737
000500	F9273E7E				DC	X*F9273E7E	RASE 10 1.00950490593
000504	F9373E7E				DC	X*F9373E7E	RASE 10 1.009848948111
000508	F9473E7E				DC	X*F9473E7E	RASE 10 1.01019298579
000512	F9573E7E				DC	X*F9573E7E	RASE 10 1.010537028111
000516	F9673E7E				DC	X*F9673E7E	RASE 10 1.010881063737
000520	F9773E7E				DC	X*F9773E7E	RASE 10 1.01122510593
000524	F9873E7E				DC	X*F9873E7E	RASE 10 1.011569148111
000528	F9973E7E				DC	X*F9973E7E	RASE 10 1.01191318579
000532	F9A73E7E				DC	X*F9A73E7E	RASE 10 1.012257228111
000536	F9B73E7E				DC	X*F9B73E7E	RASE 10 1.012601263737
000540	F9C73E7E				DC	X*F9C73E7E	RASE 10 1.01294530593
000544	F9D73E7E				DC	X*F9D73E7E	RASE 10 1.013289348111
000548	F9E73E7E				DC	X*F9E73E7E	RASE 10 1.01363338579
000552	F9F73E7E				DC	X*F9F73E7E	RASE 10 1.013977428111
000556	F9073E7E				DC	X*F9073E7E	RASE 10 1.014321463737
000560	F9173E7E				DC	X*F9173E7E	RASE 10 1.01466550593
000564	F9273E7E				DC	X*F9273E7E	RASE 10 1.015009548111
000568	F9373E7E				DC	X*F9373E7E	RASE 10 1.01535358579
000572	F9473E7E				DC	X*F9473E7E	RASE 10 1.015697628111
000576	F9573E7E				DC	X*F9573E7E	RASE 10 1.016041663737
000580	F9673E7E				DC	X*F9673E7E	RASE 10 1.01638570593
000584	F9773E7E				DC	X*F9773E7E	RASE 10 1.016729748111
000588	F9873E7E				DC	X*F9873E7E	RASE 10 1.01707378579
000592	F9973E7E				DC	X*F9973E7E	RASE 10 1.017417828111
000596	F9A73E7E				DC	X*F9A73E7E	RASE 10 1.017761863737
000600	F9B73E7E				DC	X*F9B73E7E	RASE 10 1.01810590593
000604	F9C73E7E				DC	X*F9C73E7E	RASE 10 1.018449948111
000608	F9D73E7E				DC	X*F9D73E7E	RASE 10 1.01879398579
000612	F9E73E7E				DC	X*F9E73E7E	RASE 10 1.019138028111
000616	F9F73E7E				DC	X*F9F73E7E	RASE 10 1.019482063737
000620	F9073E7E				DC	X*F9073E7E	RASE 10 1.01982610593
000624	F9173E7E				DC	X*F9173E7E	RASE 10 1.020170148111
000628	F9273E7E				DC	X*F9273E7E	RASE 10 1.02051418579
000632	F9373E7E				DC	X*F9373E7E	RASE 10 1.020858228111
000636	F9473E7E				DC	X*F9473E7E	RASE 10 1.021202263737
000640	F9573E7E				DC	X*F9573E7E	RASE 10 1.02154630593
000644	F9673E7E				DC	X*F9673E7E	RASE 10 1.021890348111
000648	F9773E7E				DC	X*F9773E7E	RASE 10 1.02223438579
000652	F9873E7E				DC	X*F9873E7E	RASE 10 1.022578428111
000656	F9973E7E				DC	X*F9973E7E	RASE 10 1.022922463737
000660	F9A73E7E				DC	X*F9A73E7E	RASE 10 1.02326650593
000664	F9B73E7E				DC	X*F9B73E	

PAGE 4

F08BAP70 6/30/71

LPC	OBJECT CODE	ADDR1 ADDR2	STMT	SOURCE STATEMENT	BASE 10	BASE 10
J001A0 0E			276	DC X'1F0	BASE 10	0.8750
J001A1 0F			277	DC X'1F0	BASE 10	0.9375
J001A2 14			278	DC X'140	BASE 10	1.0625
J001A3 11			279	DC X'110	BASE 10	1.1250
J001A4 16			280	DC X'160	BASE 10	1.1875
J001A5 13			281	DC X'130	BASE 10	1.2500
J001A6 12			282	DC X'120	BASE 10	1.3125
J001A7 18			283	DC X'180	BASE 10	1.3750
J001A8 1F			284	DC X'1F0	BASE 10	1.4375
J001A9 1C			285	DC X'1C0	BASE 10	1.5000
J001A0 1E			286	DC X'1E0	BASE 10	1.5625
J001A1 15			287	DC X'150	BASE 10	1.6250
J001A2 17			288	DC X'170	BASE 10	1.6875
J001A3 1A			289	DC X'1A0	BASE 10	1.7500
J001A4 1A			290	DC X'1A0	BASE 10	1.8125
J001A5 1F			291	DC X'1F0	BASE 10	1.8750
J001A6 22			292	DC X'220	BASE 10	1.9375
J001A7 25			293	DC X'250	BASE 10	2.0000
J001A8 21			294	DC X'210	BASE 10	2.0625
J001A9 1D			295	DC X'1D0	BASE 10	2.1250
J001A0 29			296	DC X'290	BASE 10	2.1875
J001A1 30			297	DC X'300	BASE 10	2.2500
J001A2 33			298	DC X'330	BASE 10	2.3125
J001A3 2F			299	DC X'2F0	BASE 10	2.3750
J001A4 3F			300	DC X'3F0	BASE 10	2.4375
J001A5 29			301	DC X'290	BASE 10	2.5000
J001A6 36			302	DC X'360	BASE 10	2.5625
J001A7 3C			303	DC X'3C0	BASE 10	2.6250
J001A8 3F			304	DC X'3F0	BASE 10	2.6875
J001A9 3E			305	DC X'3E0	BASE 10	2.7500
J001A0 37			306	DC X'370	BASE 10	2.8125
J001A1 3A			307	DC X'3A0	BASE 10	2.8750
J001A2 3A			308	DC X'3A0	BASE 10	2.9375
J001A3 3D			309	DC X'3D0	BASE 10	3.0000
J001A4 2C			310	DC X'2C0	BASE 10	3.0625
J001A5 3F			311	DC X'3F0	BASE 10	3.1250
J001A6 3C			312	DC X'3C0	BASE 10	3.1875
J001A7 27			313	DC X'270	BASE 10	3.2500
J001A8 34			314	DC X'340	BASE 10	3.3125
J001A9 3A			315	DC X'3A0	BASE 10	3.3750
J001A0 3A			316	DC X'3A0	BASE 10	3.4375
J001A1 3A			317	DC X'3A0	BASE 10	3.5000
J001A2 3D			318	DC X'3D0	BASE 10	3.5625
J001A3 3A			319	DC X'3A0	BASE 10	3.6250
J001A4 3A			320	DC X'3A0	BASE 10	3.6875
J001A5 3A			321	DC X'3A0	BASE 10	3.7500
J001A6 3A			322	DC X'3A0	BASE 10	3.8125
J001A7 3A			323	DC X'3A0	BASE 10	3.8750
J001A8 3A			324	DC X'3A0	BASE 10	3.9375
J001A9 3A			325	DC X'3A0	BASE 10	4.0000
J001A0 3A			326	DC X'3A0	BASE 10	4.0625
J001A1 3A			327	DC X'3A0	BASE 10	4.1250
J001A2 3A			328	DC X'3A0	BASE 10	4.1875
J001A3 3A			329	DC X'3A0	BASE 10	4.2500
J001A4 3A			330	DC X'3A0	BASE 10	4.3125

FOBARATC 6/10/71

LFC	OBJECT CODE	ADDR1 ADDR2	STMT	SOURCE	STATEMENT	BASE 10	BASE 10
000107 14			331	DC	X1140	BASE 10	1.2500
000108 14			332	DC	X1140	BASE 10	1.2500
000109 12			333	DC	X1120	BASE 10	1.1250
000110 12			334	DC	X1120	BASE 10	1.1250
000111 12			335	DC	X1120	BASE 10	1.1250
000112 12			336	DC	X1120	BASE 10	1.1250
000113 12			337	DC	X1120	BASE 10	1.1250
000114 12			338	DC	X1120	BASE 10	1.1250
000115 25			339	DC	X1250	BASE 10	2.3125
000116 25			340	DC	X1250	BASE 10	2.3125
000117 25			341	DC	X1250	BASE 10	2.3125
000118 21			342	DC	X1210	BASE 10	2.0625
000119 21			343	DC	X1210	BASE 10	2.0625
000120 10			344	DC	X1100	BASE 10	1.0625
000121 10			345	DC	X1100	BASE 10	1.0625
000122 10			346	DC	X1100	BASE 10	1.0625
000123 10			347	DC	X1100	BASE 10	1.0625
000124 10			348	DC	X1100	BASE 10	1.0625
000125 10			349	DC	X1100	BASE 10	1.0625
000126 10			350	DC	X1100	BASE 10	1.0625
000127 10			351	DC	X1100	BASE 10	1.0625
000128 10			352	DC	X1100	BASE 10	1.0625
000129 10			353	DC	X1100	BASE 10	1.0625
000130 10			354	DC	X1100	BASE 10	1.0625
000131 10			355	DC	X1100	BASE 10	1.0625
000132 10			356	DC	X1100	BASE 10	1.0625
000133 10			357	DC	X1100	BASE 10	1.0625
000134 10			358	DC	X1100	BASE 10	1.0625
000135 10			359	DC	X1100	BASE 10	1.0625
000136 10			360	DC	X1100	BASE 10	1.0625
000137 10			361	DC	X1100	BASE 10	1.0625
000138 10			362	DC	X1100	BASE 10	1.0625
000139 10			363	DC	X1100	BASE 10	1.0625
000140 10			364	DC	X1100	BASE 10	1.0625
000141 10			365	DC	X1100	BASE 10	1.0625
000142 10			366	DC	X1100	BASE 10	1.0625
000143 10			367	DC	X1100	BASE 10	1.0625
000144 10			368	DC	X1100	BASE 10	1.0625
000145 10			369	DC	X1100	BASE 10	1.0625
000146 10			370	DC	X1100	BASE 10	1.0625
000147 10			371	DC	X1100	BASE 10	1.0625
000148 10			372	DC	X1100	BASE 10	1.0625
000149 10			373	DC	X1100	BASE 10	1.0625
000150 10			374	DC	X1100	BASE 10	1.0625
000151 10			375	DC	X1100	BASE 10	1.0625
000152 10			376	DC	X1100	BASE 10	1.0625
000153 10			377	DC	X1100	BASE 10	1.0625
000154 10			378	DC	X1100	BASE 10	1.0625
000155 10			379	DC	X1100	BASE 10	1.0625
000156 10			380	DC	X1100	BASE 10	1.0625
000157 10			381	DC	X1100	BASE 10	1.0625
000158 10			382	DC	X1100	BASE 10	1.0625
000159 10			383	DC	X1100	BASE 10	1.0625
000160 10			384	DC	X1100	BASE 10	1.0625
000161 10			385	DC	X1100	BASE 10	1.0625

FOIAPR70 6/10/71

115

LOC	OBJECT CODE	ANDRI ADOP?	STMT	SOURCE STATEMENT	BASE	LOC
00040E 27			384	DC X*77*	BASE 10 2.4375	
00040F 1A			387	DC X*7A*	BASE 10 3.6250	
000410 3A			388	DC X*7A*	BASE 10 3.6250	
000411 3A			389	DC X*7A*	BASE 10 3.6250	
000412 1A			390	DC X*7A*	BASE 10 3.6250	
000413 1A			391	DC X*7A*	BASE 10 3.6250	
000414 31			392	DC X*71*	BASE 10 3.6250	
000415 38			393	DC X*7A*	BASE 10 3.6250	
000416 1A			394	DC X*7A*	BASE 10 3.6250	
000417 3E			395	DC X*7A*	BASE 10 3.6250	
000418 3A			396	DC X*7A*	BASE 10 3.6250	
000419 1A			397	DC X*7A*	BASE 10 3.6250	
000420 1A			398	DC X*7A*	BASE 10 3.6250	
000421 1A			399	DC X*7A*	BASE 10 3.6250	
000422 1A			400	DC X*7A*	BASE 10 3.6250	
000423 1A			401	DC X*7A*	BASE 10 3.6250	
000424 1A			402	DC X*7A*	BASE 10 3.6250	
000425 1A			403	DC X*7A*	BASE 10 3.6250	
000426 1A			404	DC X*7A*	BASE 10 3.6250	
000427 1A			405	DC X*7A*	BASE 10 3.6250	
000428 1A			406	DC X*7A*	BASE 10 3.6250	
000429 1A			407	DC X*7A*	BASE 10 3.6250	
000430 1A			408	DC X*7A*	BASE 10 3.6250	
000431 1A			409	DC X*7A*	BASE 10 3.6250	
000432 1A			410	DC X*7A*	BASE 10 3.6250	
000433 1A			411	DC X*7A*	BASE 10 3.6250	
000434 1A			412	DC X*7A*	BASE 10 3.6250	
000435 1A			413	DC X*7A*	BASE 10 3.6250	
000436 1A			414	DC X*7A*	BASE 10 3.6250	
000437 1A			415	DC X*7A*	BASE 10 3.6250	
000438 1A			416	DC X*7A*	BASE 10 3.6250	
000439 1A			417	DC X*7A*	BASE 10 3.6250	
000440 1A			418	DC X*7A*	BASE 10 3.6250	
000441 1A			419	DC X*7A*	BASE 10 3.6250	
000442 1A			420	DC X*7A*	BASE 10 3.6250	
000443 1A			421	DC X*7A*	BASE 10 3.6250	
000444 1A			422	DC X*7A*	BASE 10 3.6250	
000445 1A			423	DC X*7A*	BASE 10 3.6250	
000446 1A			424	DC X*7A*	BASE 10 3.6250	
000447 1A			425	DC X*7A*	BASE 10 3.6250	
000448 1A			426	DC X*7A*	BASE 10 3.6250	
000449 1A			427	DC X*7A*	BASE 10 3.6250	
000450 1A			428	DC X*7A*	BASE 10 3.6250	
000451 1A			429	DC X*7A*	BASE 10 3.6250	
000452 1A			430	DC X*7A*	BASE 10 3.6250	
000453 1A			431	DC X*7A*	BASE 10 3.6250	
000454 1A			432	DC X*7A*	BASE 10 3.6250	
000455 1A			433	DC X*7A*	BASE 10 3.6250	
000456 1A			434	DC X*7A*	BASE 10 3.6250	
000457 1A			435	DC X*7A*	BASE 10 3.6250	
000458 1A			436	DC X*7A*	BASE 10 3.6250	
000459 1A			437	DC X*7A*	BASE 10 3.6250	
000460 1A			438	DC X*7A*	BASE 10 3.6250	
000461 1A			439	DC X*7A*	BASE 10 3.6250	
000462 1A			440	DC X*7A*	BASE 10 3.6250	

PAGE 9

F38APR70 6/30/71

LFC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	BASE 10	BASE 10
000445	1E			441	DC	X*1E*	1.0750	BASE 10 1.0750
000446	1E			442	DC	X*1E*	1.0750	BASE 10 1.0750
000447	1E			443	DC	X*1E*	1.0750	BASE 10 1.0750
000448	1A			444	DC	X*1A*	1.0750	BASE 10 1.0750
000449	1A			445	DC	X*1A*	1.0750	BASE 10 1.0750
000450	1A			446	DC	X*1A*	1.0750	BASE 10 1.0750
000451	1A			447	DC	X*1A*	1.0750	BASE 10 1.0750
000452	1A			448	DC	X*1A*	1.0750	BASE 10 1.0750
000453	1A			449	DC	X*1A*	1.0750	BASE 10 1.0750
000454	1A			450	DC	X*1A*	1.0750	BASE 10 1.0750
000455	1A			451	DC	X*1A*	1.0750	BASE 10 1.0750
000456	1A			452	DC	X*1A*	1.0750	BASE 10 1.0750
000457	1A			453	DC	X*1A*	1.0750	BASE 10 1.0750
000458	1A			454	DC	X*1A*	1.0750	BASE 10 1.0750
000459	1A			455	DC	X*1A*	1.0750	BASE 10 1.0750
000460	1A			456	DC	X*1A*	1.0750	BASE 10 1.0750
000461	1A			457	DC	X*1A*	1.0750	BASE 10 1.0750
000462	1A			458	DC	X*1A*	1.0750	BASE 10 1.0750
000463	1A			459	DC	X*1A*	1.0750	BASE 10 1.0750
000464	1A			460	DC	X*1A*	1.0750	BASE 10 1.0750
000465	1A			461	DC	X*1A*	1.0750	BASE 10 1.0750
000466	1A			462	DC	X*1A*	1.0750	BASE 10 1.0750
000467	1A			463	DC	X*1A*	1.0750	BASE 10 1.0750
000468	1A			464	DC	X*1A*	1.0750	BASE 10 1.0750
000469	1A			465	DC	X*1A*	1.0750	BASE 10 1.0750
000470	1A			466	DC	X*1A*	1.0750	BASE 10 1.0750
000471	1A			467	DC	X*1A*	1.0750	BASE 10 1.0750
000472	1A			468	DC	X*1A*	1.0750	BASE 10 1.0750
000473	1A			469	DC	X*1A*	1.0750	BASE 10 1.0750
000474	1A			470	DC	X*1A*	1.0750	BASE 10 1.0750
000475	1A			471	DC	X*1A*	1.0750	BASE 10 1.0750
000476	1A			472	DC	X*1A*	1.0750	BASE 10 1.0750
000477	1A			473	DC	X*1A*	1.0750	BASE 10 1.0750
000478	1A			474	DC	X*1A*	1.0750	BASE 10 1.0750
000479	1A			475	DC	X*1A*	1.0750	BASE 10 1.0750
000480	1A			476	DC	X*1A*	1.0750	BASE 10 1.0750
000481	1A			477	DC	X*1A*	1.0750	BASE 10 1.0750
000482	1A			478	DC	X*1A*	1.0750	BASE 10 1.0750
000483	1A			479	DC	X*1A*	1.0750	BASE 10 1.0750
000484	1A			480	DC	X*1A*	1.0750	BASE 10 1.0750
000485	1A			481	DC	X*1A*	1.0750	BASE 10 1.0750
000486	1A			482	DC	X*1A*	1.0750	BASE 10 1.0750
000487	1A			483	DC	X*1A*	1.0750	BASE 10 1.0750
000488	1A			484	DC	X*1A*	1.0750	BASE 10 1.0750
000489	1A			485	DC	X*1A*	1.0750	BASE 10 1.0750
000490	1A			486	DC	X*1A*	1.0750	BASE 10 1.0750
000491	1A			487	DC	X*1A*	1.0750	BASE 10 1.0750
000492	1A			488	DC	X*1A*	1.0750	BASE 10 1.0750
000493	1A			489	DC	X*1A*	1.0750	BASE 10 1.0750
000494	1A			490	DC	X*1A*	1.0750	BASE 10 1.0750
000495	1A			491	DC	X*1A*	1.0750	BASE 10 1.0750
000496	1A			492	DC	X*1A*	1.0750	BASE 10 1.0750
000497	1A			493	DC	X*1A*	1.0750	BASE 10 1.0750
000498	1A			494	DC	X*1A*	1.0750	BASE 10 1.0750
000499	1A			495	DC	X*1A*	1.0750	BASE 10 1.0750

PAGE 10

F08APR70 6/30/71

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	BASE 10	0.7500
00047C	CC			496	DC X'0C'	BASE 10	0.7500
00047D	CC			497	DC X'0C'	BASE 10	0.7500
00047E	CC			498	DC X'0C'	BASE 10	0.8125
00047F	ON			499	DC X'0D'	BASE 10	0.8125
000480	ON			500	DC X'0D'	BASE 10	0.8125
000481	10			501	DC X'10'	BASE 10	1.0000
000482	10			502	DC X'10'	BASE 10	1.0000
000483	16			503	DC X'10'	BASE 10	1.3750
000484	16			504	DC X'16'	BASE 10	1.3750
000485	16			505	DC X'16'	BASE 10	1.3750
000486	16			506	DC X'16'	BASE 10	1.3750
000487	18			507	DC X'18'	BASE 10	1.5000
000488	1A			508	DC X'1A'	BASE 10	1.5000
000489	1C			509	DC X'1C'	BASE 10	1.7500
00048A	1C			510	DC X'1C'	BASE 10	1.7500
00048B	1C			511	DC X'1C'	BASE 10	1.7500
00048C	1C			512	DC X'1C'	BASE 10	1.7500
00048D	1C			513	DC X'1C'	BASE 10	1.7500
00048E	1E			514	DC X'1E'	BASE 10	1.8750
00048F	1E			515	DC X'1E'	BASE 10	1.8750
000490	19			516	DC X'19'	BASE 10	1.5625
000491	19			517	DC X'19'	BASE 10	1.5625
000492	19			518	DC X'19'	BASE 10	1.5625
000493	19			519	DC X'19'	BASE 10	1.5625
000494	17			520	DC X'17'	BASE 10	1.4375
000495	17			521	DC X'17'	BASE 10	1.4375
000496	17			522	DC X'17'	BASE 10	1.4375
000497	17			523	DC X'17'	BASE 10	1.4375
000498	17			524	DC X'17'	BASE 10	1.4375
000499	17			525	DC X'17'	BASE 10	1.4375
00049A	17			526	DC X'17'	BASE 10	1.4375
00049B	1A			527	DC X'1A'	BASE 10	1.6875
00049C	1A			528	DC X'1A'	BASE 10	1.6875
00049D	23			529	DC X'23'	BASE 10	2.1875
00049E	23			530	DC X'23'	BASE 10	2.1875
00049F	1F			531	DC X'1F'	BASE 10	1.9375
0004A0	1F			532	DC X'1F'	BASE 10	1.9375
0004A1	1F			533	DC X'1F'	BASE 10	1.9375
0004A2	20			534	DC X'20'	BASE 10	2.0000
0004A3	20			535	DC X'20'	BASE 10	2.0000
0004A4	20			536	DC X'20'	BASE 10	2.0000
0004A5	20			537	DC X'20'	BASE 10	2.0000
0004A6	33			538	DC X'33'	BASE 10	3.1875
0004A7	24			539	DC X'24'	BASE 10	2.2500
0004A8	24			540	DC X'24'	BASE 10	2.2500
0004A9	24			541	DC X'24'	BASE 10	2.2500
0004AA	23			542	DC X'23'	BASE 10	2.5625
0004AB	35			543	DC X'35'	BASE 10	3.3125
0004AC	2A			544	DC X'2A'	BASE 10	2.6250
0004AD	2A			545	DC X'2A'	BASE 10	2.6250
0004AE	2H			546	DC X'2H'	BASE 10	2.6875
0004AF	2H			547	DC X'2H'	BASE 10	2.6875
0004B0	34			548	DC X'34'	BASE 10	3.2500
0004B1	34			549	DC X'34'	BASE 10	3.2500
0004B2	03			550	DC X'00'	BASE 10	0.0000

APPENDIX D
Assembler Listing of GEN3

NOT REPRODUCIBLE

PAGE 1

F73APR70 6/30/71

LOC OBJECT CODE ADDR1 ADDR2 STAT SOURCE STATEMENT

000000

```

1 GEN3 START 0
2 PRINT ON,NGEN,NOCATA
3
4 SOURCE--L.F.CANNON, UGA, DEPT. OF STAT., AUG 1970
5 PURPOSE--TO SUPPLY THE USER WITH A FAST PROCEDURE FOR GENERATING NORMALLY DISTRIBUTED RANDOM NUMBERS.
6 USAGE-----X-RANDNM(IND), IND MUST BE AN ODD POSITIVE INTEGER, IT PRIMES THE GENERATING SCHEME.
7
8 EXAMPLE--X-RANDNM(55865).
9 METHOD--THE METHOD IS THAT OF MARSAGLIA AS DESCRIBED IN "COMMUNICATIONS OF ACM", VOL. 7, NO. 1, JAN 1964. THE PROGRAM DOCUMENTATION WILL FOLLOW THE NOTATION USED BY MARSAGLIA.
10
11 ENTRY RANDOM
12
13
14 RANDM R 12(0,15)
15 DC XL1(07)
16 OC CL7(0) RANDM
17
18 STM 14,R,12(13)
19
20 9ALR 9,0
21 USING 9,0
22 LR 5,13
23 LA 13,ARCT
24 ST 13,ARCT,91
25 ST 5,4(0,13)
26
27 SP 6,6
28 M 4,9PAT
29 ST 5,LA
30 SP 0,0
31 SP 4,4
32
33 SLDL 4,6
34 CL 4,FF44
35 BC 11,11
36 SLL 5,6
37 IC 6,4(4)
38 R EX
39
40 SLDL 4,3
41 CL 4,FF475
42 RC 11,12
43 SLL 5,3
44 IC 6,4-326(4)
45 R EX
46
47 SLDL 4,3
48 CL 4,FF3561
49 RC 11,13
50 S 4,FF3703
51 IC 6,4(4)
52 LR 4,6
53 SPDL 4,16
54 ST 5,RES
55 L 5,LA
56 M 4,NPAT
57 ST 5,LA
58 LTR 5,5
59 BP 0+12

```

GENERATE UUNIFORM ON (0,1).
 RESULT TO BE PLACED IN FPRG.
 DID2 IN R4.
 44(RASF 107=56(RASE 9).
 IF DID2>444 BRANCH TO L1.
 RS=050607...
 OBTAIN TABLE VALUE.
 R4=DID203.
 475(RASE 101=731(RASE 9).
 IF DID203>475 BRANCH TO L2.
 RS=050607...
 OBTAIN TABLE VALUE.
 R4=DID20104.
 3961(RASE 101=7571(RASE 8).
 IF DID20104>3961 BRANCH TO L3.
 OBTAIN PROPER TABLE ADDRESS.
 OBTAIN TABLE VALUE.
 R4=DISCRETE TABLE VALUE.
 RS=TABLE R.V. WITH 0506... TACKED ON.
 CONVERT TO FLOATING POINT.
 ATTACH + OR --.

PAGE 2

FOBAPR70 6/30/71

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000092	92C1 8302	C0314		56	MVI RFS,X'C1*
000093	47F3 908C	0009F		57	RC 15,X*8
000094	9241 9302	00314		58	MVI RFS,X'41*
000095	7400 9702	00314		59	AF 0,RES
000096	5ED0 92A6	002C8		60	L 13,49FA44
000097	98F9 900C	000CC		61	LM 14,9,12(13)
000098	92FF 900C			62	MVI 12(13),X'FF*
000099	07FE			63	PCR 15,14
000100	5853 92FE	00310		64	L 5,LA
000101	0201 907E	00380		65	MVC BRANCH,TEMP
000102	5853 1000	00310		66	L 5,0(,1)
000103	5853 50C0	00000		67	L 5,0(,15)
000104	5050 92FE	00310		68	ST 5,LA
000105	47F3 9012	00024		69	B BRANCH+4
000106	5150 92FE	00310		70	L 5,LA
000107	4123 0094	00004		71	LA 2,4
000108	5553 909E	00000		72	CL 5,X'FF4F10F0*
000109	4780 918E	00100		73	BC 11,LA
000110	1167			74	LNR 6,2
000111	5556 83C6	00308		75	AR 6,2
000112	4780 90CA	0000C		76	CL 5,1(6)
000113	5555 848A	00498		77	RC 11,X*4
000114	4780 90FC	000FE		78	CL 5,C(6)
000115	5C63 92FA	003CC		79	RC 11,LA
000116	5053 92FE	00310		80	M 4,BPAT
000117	9850 0002	00002		81	ST 5,LA
000118	47F3 8184	001C6		82	SRL 6,2
000119	5C63 92FA	003CC		83	B L5
000120	1875			84	M 4,BPAT
000121	1875			85	LR 7,5
000122	5C63 92FA	003CC		86	M 4,BPAT
000123	5053 92FE	00310		87	ST 5,LA
000124	1875			88	CLR 7,5
000125	4780 9106	00118		89	RC 11,LA
000126	1857			90	LP 5,7
000127	1857			91	L 7,LA
000128	5973 92FE	00310		92	SRL 6,1
000129	9863 0001	000C1		93	SR 4,4
000130	1944			94	LM 4,0161
000131	4844 8546	00558		95	LPR 2,4
000132	1024			96	LQ 7,7
000133	8937 0010	00010		97	SRL 3,16
000134	1532			98	CLR 3,2
000135	4780 8176	00138		99	RC 11,LA
000136	9863 9001	00001		100	SRL 6,1
000137	47F0 9184	001C6		101	B L5
000138	8F40 0001	00001		102	SRL 6,1
000139	5053 929A	002AC		103	ST 5,MIN
000140	1875			104	SRL 7,5
000141	8970 0008	00008		105	SRL 7,5
000142	3070 929E	00280		106	ST 7,0IF
000143	9243 929E			107	MVI 107
000144	3070			108	SEP 2,2
000145	8450 000C	00280		109	AE 2,0IF
000146	8450 000C	0000C		110	SRL 5,12

NORMALIZED RESULT IN FPRO.

REPLACES INSTR. AT BRANCH.

R5=U.

IF U5=FF4F10F0(BASF 16) GO TO L4.

INCREMENT 1(P6).

COMPARE U TO R11.

IF U5=R(1) CONTINUE LOOP.

COMPARE U TO C(1).

IF C(1)=C(1) GO TO L4.

GENERATE A NEW U.

GENERATE U1.

SAVE U1.

GENERATE U2.

R5=MIN(U1,U2).

IF MAX(U1,U2)<C(1) GO TO L5.

R7=MAX-MIN.

CONVERT TO FLOATING POINT.

PAGE 3

FORAPR72 6/30/71

LFC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000158	4050 9242	00284		111	ST 5.0
000159	9240 9242	00284		112	MVI 4.0
000160	7640 9242	00604		113	LF 4.0
000161	7640 9242	00284		114	LER 6.4
000162	7640 9242	00284		115	SF 4.0
000163	7640 9242	00284		116	AE 4.0
000164	7640 9242	00284		117	SF 3.3
000165	7640 9242	00284		118	IC 3.0
000166	7640 9242	00284		119	SLL 3.1
000167	7640 9242	00284		120	ST 3.0
000168	7640 9242	00284		121	MVI 4.0
000169	7640 9242	00284		122	AE 6.0
000170	7640 9242	00284		123	MFR 4.0
000171	7640 9242	00284		124	ME 4.0
000172	7640 9242	00284		125	STE 4.0
000173	7640 9242	00284		126	CALL EXP(1.0)
000174	7640 9242	00284		127	SE 0.0
000175	7640 9242	00284		128	SLL 6.2
000176	7640 9242	00284		129	MF 0.0
000177	7640 9242	00284		130	L 5.0
000178	7640 9242	00284		131	CEP 0.0
000179	7640 9242	00284		132	AC 13.0
000180	7640 9242	00284		133	L 5.0
000181	7640 9242	00284		134	SEF 0.0
000182	7640 9242	00284		135	SRL 6.2
000183	7640 9242	00284		136	SR 4.0
000184	7640 9242	00284		137	IC 4.0
000185	7640 9242	00284		138	R FL
000186	7640 9242	00284		139	L 5.0
000187	7640 9242	00284		140	SR 7.0
000188	7640 9242	00284		141	M 4.0
000189	7640 9242	00284		142	SR 4.0
000190	7640 9242	00284		143	ST 5.0
000191	7640 9242	00284		144	SRL 5.0
000192	7640 9242	00284		145	ST 5.0
000193	7640 9242	00284		146	L 5.0
000194	7640 9242	00284		147	NR 4.0
000195	7640 9242	00284		148	ALR 7.0
000196	7640 9242	00284		149	L 5.0
000197	7640 9242	00284		150	M 4.0
000198	7640 9242	00284		151	ST 5.0
000199	7640 9242	00284		152	SR 4.0
000200	7640 9242	00284		153	ST 5.0
000201	7640 9242	00284		154	SRL 5.0
000202	7640 9242	00284		155	ST 5.0
000203	7640 9242	00284		156	L 5.0
000204	7640 9242	00284		157	NR 4.0
000205	7640 9242	00284		158	ALR 7.0
000206	7640 9242	00284		159	L 5.0
000207	7640 9242	00284		160	M 4.0
000208	7640 9242	00284		161	ST 5.0
000209	7640 9242	00284		162	SR 4.0
000210	7640 9242	00284		163	SRL 5.0
000211	7640 9242	00284		164	ST 5.0
000212	7640 9242	00284		165	L 5.0
000213	7640 9242	00284		166	NR 4.0
000214	7640 9242	00284		167	ALR 7.0
000215	7640 9242	00284		168	L 5.0
000216	7640 9242	00284		169	M 4.0
000217	7640 9242	00284		170	ST 5.0
000218	7640 9242	00284		171	SR 4.0
000219	7640 9242	00284		172	SRL 5.0
000220	7640 9242	00284		173	ST 5.0
000221	7640 9242	00284		174	L 5.0
000222	7640 9242	00284		175	NR 4.0
000223	7640 9242	00284		176	ALR 7.0
000224	7640 9242	00284		177	L 5.0
000225	7640 9242	00284		178	M 4.0
000226	7640 9242	00284		179	ST 5.0
000227	7640 9242	00284		180	SR 4.0
000228	7640 9242	00284		181	SRL 5.0
000229	7640 9242	00284		182	ST 5.0
000230	7640 9242	00284		183	L 5.0
000231	7640 9242	00284		184	NR 4.0
000232	7640 9242	00284		185	ALR 7.0
000233	7640 9242	00284		186	L 5.0
000234	7640 9242	00284		187	M 4.0
000235	7640 9242	00284		188	ST 5.0
000236	7640 9242	00284		189	SR 4.0
000237	7640 9242	00284		190	ST 5.0
000238	7640 9242	00284		191	SRL 5.0
000239	7640 9242	00284		192	ST 5.0
000240	7640 9242	00284		193	L 5.0
000241	7640 9242	00284		194	NR 4.0
000242	7640 9242	00284		195	ALR 7.0
000243	7640 9242	00284		196	L 5.0
000244	7640 9242	00284		197	M 4.0
000245	7640 9242	00284		198	ST 5.0
000246	7640 9242	00284		199	SR 4.0
000247	7640 9242	00284		200	ST 5.0
000248	7640 9242	00284		201	SRL 5.0
000249	7640 9242	00284		202	ST 5.0
000250	7640 9242	00284		203	L 5.0
000251	7640 9242	00284		204	NR 4.0
000252	7640 9242	00284		205	ALR 7.0
000253	7640 9242	00284		206	L 5.0
000254	7640 9242	00284		207	M 4.0
000255	7640 9242	00284		208	ST 5.0
000256	7640 9242	00284		209	SR 4.0
000257	7640 9242	00284		210	SRL 5.0
000258	7640 9242	00284		211	ST 5.0
000259	7640 9242	00284		212	L 5.0
000260	7640 9242	00284		213	NR 4.0
000261	7640 9242	00284		214	ALR 7.0
000262	7640 9242	00284		215	L 5.0
000263	7640 9242	00284		216	M 4.0
000264	7640 9242	00284		217	ST 5.0
000265	7640 9242	00284		218	SR 4.0
000266	7640 9242	00284		219	SRL 5.0
000267	7640 9242	00284		220	ST 5.0
000268	7640 9242	00284		221	L 5.0
000269	7640 9242	00284		222	NR 4.0
000270	7640 9242	00284		223	ALR 7.0
000271	7640 9242	00284		224	L 5.0
000272	7640 9242	00284		225	M 4.0
000273	7640 9242	00284		226	ST 5.0
000274	7640 9242	00284		227	SR 4.0
000275	7640 9242	00284		228	SRL 5.0
000276	7640 9242	00284		229	ST 5.0
000277	7640 9242	00284		230	L 5.0
000278	7640 9242	00284		231	NR 4.0
000279	7640 9242	00284		232	ALR 7.0
000280	7640 9242	00284		233	L 5.0
000281	7640 9242	00284		234	M 4.0
000282	7640 9242	00284		235	ST 5.0
000283	7640 9242	00284		236	SR 4.0
000284	7640 9242	00284		237	SRL 5.0
000285	7640 9242	00284		238	ST 5.0
000286	7640 9242	00284		239	L 5.0
000287	7640 9242	00284		240	NR 4.0
000288	7640 9242	00284		241	ALR 7.0
000289	7640 9242	00284		242	L 5.0
000290	7640 9242	00284		243	M 4.0
000291	7640 9242	00284		244	ST 5.0
000292	7640 9242	00284		245	SR 4.0
000293	7640 9242	00284		246	SRL 5.0
000294	7640 9242	00284		247	ST 5.0
000295	7640 9242	00284		248	L 5.0
000296	7640 9242	00284		249	NR 4.0
000297	7640 9242	00284		250	ALR 7.0
000298	7640 9242	00284		251	L 5.0
000299	7640 9242	00284		252	M 4.0
000300	7640 9242	00284		253	ST 5.0
000301	7640 9242	00284		254	SR 4.0
000302	7640 9242	00284		255	SRL 5.0
000303	7640 9242	00284		256	ST 5.0
000304	7640 9242	00284		257	L 5.0
000305	7640 9242	00284		258	NR 4.0
000306	7640 9242	00284		259	ALR 7.0
000307	7640 9242	00284		260	L 5.0
000308	7640 9242	00284		261	M 4.0
000309	7640 9242	00284		262	ST 5.0
000310	7640 9242	00284		263	SR 4.0
000311	7640 9242	00284		264	SRL 5.0
000312	7640 9242	00284		265	ST 5.0
000313	7640 9242	00284		266	L 5.0
000314	7640 9242	00284		267	NR 4.0
000315	7640 9242	00284		268	ALR 7.0
000316	7640 9242	00284		269	L 5.0
000317	7640 9242	00284		270	M 4.0
000318	7640 9242	00284		271	ST 5.0
000319	7640 9242	00284		272	SR 4.0
000320	7640 9242	00284		273	SRL 5.0
000321	7640 9242	00284		274	ST 5.0
000322	7640 9242	00284		275	L 5.0
000323	7640 9242	00284		276	NR 4.0
000324	7640 9242	00284		277	ALR 7.0
000325	7640 9242	00284		278	L 5.0
000326	7640 9242	00284		279	M 4.0
000327	7640 9242	00284		280	ST 5.0
000328	7640 9242	00284		281	SR 4.0
000329	7640 9242	00284		282	SRL 5.0
000330	7640 9242	00284		283	ST 5.0
000331	7640 9242	00284		284	L 5.0
000332	7640 9242				

PAGE 4

F304PR70 6/30/71

LTC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000742	7400 9402		006F4	188	AF 0-F*9.0°
000746	7000 924E		007C0	189	DF 0.5
000744	7000 924E		002C0	190	STE 0.5
				191	CALL SQ. (S)
000746	9240 9246	00288		203	MVT U1.X*40°
000746	1922			204	SEP 2.2
000746	7420 9246	00288		205	AE 2.01
000746	3020			206	NER 2.0
000746	7920 9406		006F8	207	CE 2-F*3.0°
000746	4720 927E		00290	208	RC 2.110
000746	9240 924A	0028C		209	PVI U2.X*40°
000746	1922			210	SER 2.2
000746	7420 924A			211	AF 2.02
000746	1020		0028C	212	NER 2.0
000746	1920			213	SFR 0.0
000746	7520 9406		006F4	214	CE 2-F*3.0°
000746	4720 918E		001C0	215	RC 1.114
000746	5450 927E		00110	216	L 5.14
000746	5450 927E		0010C	217	M 4.8PAT
000746	5050 927E		00310	218	ST 5.14
000746	1255			219	LTP 5.5
000746	4720 9292		00244	220	DP 5.5
000742	3125			221	LNFR 2.2
000746	4720 8090		00742	222	LER 0.2
000746				223	R OUT
000746				224	DS 1F
000746				225	OS 1F
000746				226	NS 1F
000746				227	DS 1F
000746				228	DS 1F
000746				229	DS 1F
000746				230	NS 1F
000746				231	NS 1F
000746				232	NS 1F
000746				233	NS 1F
000746				234	NS 1F
000746				235	NS 1F
000746				236	NS 1F
000746				237	NS 1F
000746				238	NS 1F
000746				239	NS 1F
000746				240	NS 1F
000746				241	NS 1F
000746				242	NS 1F
000746				243	NS 1F
000746				244	NS 1F
000746				245	NS 1F
000746				246	NS 1F
000746				247	NS 1F
000746				248	NS 1F
000746				249	NS 1F
000746				250	NS 1F
000746				251	NS 1F
000746				252	NS 1F
000746				253	NS 1F

CONVERT TO FLT. PT.

IF FPR23.0 BRANCH TO L10.
CONVERT TO FLT. PT.

IF FPR2=C3.0 BRANCH TO L4.

F08APR70 6/30/71

124

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
000348	422A/448			234	DC	E442-444A6660
000349	417A31F			255	DC	E47-637254400
000350	418A0A6			256	DC	E48-1A76A4300
000351	421A3333			257	DC	E49-199999990
000352	418A31F			258	DC	E47-90336A650
000353	417A21AE			259	DC	E47-152A62R770
000354	42AC000			260	DC	E46-00
000355	418A31F			261	DC	E45-8341557520
000356	416A6A6A			262	DC	E46-5255761150
000357	418A31F			263	DC	E46-9315169230
000358	42800100			264	DC	E42R-00
000359	418A31F			265	DC	E46-1A1A7A4740
000360	418A31F			266	DC	E45-2672873740
000361	418A31F			267	DC	E425A-00
000362	418A31F			268	DC	E44-330A93A330
000363	418A31F			269	DC	E45-362210A550
000364	418A31F			270	DC	E45-335302A940
000365	418A31F			271	DC	E45-5173576950
000366	418A31F			272	DC	E45-6285067430
000367	418A31F			273	DC	E45-121170280
000368	418A31F			274	DC	E45-9249270530
000369	418A31F			275	DC	E41-194677340
000370	418A31F			276	DC	E45-333333330
000371	418A31F			277	DC	E45-00
000372	418A31F			278	DC	E410-5551C1650
000373	418A31F			279	DC	E47-347499A600
000374	418A31F			280	DC	E46-722A165600
000375	418A31F			281	DC	E45-9938656320
000376	418A31F			282	DC	E47D222830
000377	418A31F			283	DC	E46A403030
000378	418A31F			284	DC	E46A403030
000379	418A31F			285	DC	E46A403030
000380	418A31F			286	DC	E46A403030
000381	418A31F			287	DC	E46A403030
000382	418A31F			288	DC	E46A403030
000383	418A31F			289	DC	E46A403030
000384	418A31F			290	DC	E46A403030
000385	418A31F			291	DC	E46A403030
000386	418A31F			292	DC	E46A403030
000387	418A31F			293	DC	E46A403030
000388	418A31F			294	DC	E46A403030
000389	418A31F			295	DC	E46A403030
000390	418A31F			296	DC	E46A403030
000391	418A31F			297	DC	E46A403030
000392	418A31F			298	DC	E46A403030
000393	418A31F			299	DC	E46A403030
000394	418A31F			300	DC	E46A403030
000395	418A31F			301	DC	E46A403030
000396	418A31F			302	DC	E46A403030
000397	418A31F			303	DC	E46A403030
000398	418A31F			304	DC	E46A403030
000399	418A31F			305	DC	E46A403030
000400	418A31F			306	DC	E46A403030
000401	418A31F			307	DC	E46A403030
000402	418A31F			308	DC	E46A403030

PAGE 6

FORADPTO 6/30/71

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000444	FD4A019F			309	DC X*FD4A019F
000444	FD391070			310	DC X*FD391070
00044C	FD79158F			311	DC X*FD79158F
000450	FD4F5076			312	DC X*FD4F5076
000454	FD4311EA			313	DC X*FD4311EA
000458	FD7A7AF			314	DC X*FD7A7AF
00045C	FD49A2EA			315	DC X*FD49A2EA
000460	FD49A0E			316	DC X*FD49A0E
000464	FD4CA777			317	DC X*FD4CA777
000468	FD4DA1D			318	DC X*FD4DA1D
00046C	FD4CE049			319	DC X*FD4CE049
000470	FD4F0F0C			320	DC X*FD4F0F0C
000474	FD4735F6			321	DC X*FD4735F6
000478	FD4F2273			322	DC X*FD4F2273
00047C	FD4F5511			323	DC X*FD4F5511
000480	FD4F3425			324	DC X*FD4F3425
000484	FD4F4149			325	DC X*FD4F4149
000488	FD4F0C23			326	DC X*FD4F0C23
000490	FD4F30E3			327	DC X*FD4F30E3
000494	FD4F17F0			328	DC X*FD4F17F0
000498	FD4F0C7A			329	DC X*FD4F0C7A
000500	FD4F17204			330	DC X*FD4F17204
000504	FD4F3497			331	DC X*FD4F3497
000508	FD4F0F0F			332	DC X*FD4F0F0F
000512	FD4F4018			333	DC X*FD4F4018
000516	FD4F4F52			334	DC X*FD4F4F52
000520	FD4F4C77			335	DC X*FD4F4C77
000524	FD4F4787			336	DC X*FD4F4787
000528	FD4F454C8			337	DC X*FD4F454C8
000532	FD4F4356			338	DC X*FD4F4356
000536	FD4F43C1			339	DC X*FD4F43C1
000540	FD4F43C6			340	DC X*FD4F43C6
000544	FD4F43C6			341	DC X*FD4F43C6
000548	FD4F43C6			342	DC X*FD4F43C6
000552	FD4F43C6			343	DC X*FD4F43C6
000556	FD4F43C6			344	DC X*FD4F43C6
000560	FD4F43C6			345	DC X*FD4F43C6
000564	FD4F43C6			346	DC X*FD4F43C6
000568	FD4F43C6			347	DC X*FD4F43C6
000572	FD4F43C6			348	DC X*FD4F43C6
000576	FD4F43C6			349	DC X*FD4F43C6
000580	FD4F43C6			350	DC X*FD4F43C6
000584	FD4F43C6			351	DC X*FD4F43C6
000588	FD4F43C6			352	DC X*FD4F43C6
000592	FD4F43C6			353	DC X*FD4F43C6
000596	FD4F43C6			354	DC X*FD4F43C6
000600	FD4F43C6			355	DC X*FD4F43C6
000604	FD4F43C6			356	DC X*FD4F43C6
000608	FD4F43C6			357	DC X*FD4F43C6
000612	FD4F43C6			358	DC X*FD4F43C6
000616	FD4F43C6			359	DC X*FD4F43C6
000620	FD4F43C6			360	DC X*FD4F43C6
000624	FD4F43C6			361	DC X*FD4F43C6
000628	FD4F43C6			362	DC X*FD4F43C6
000632	FD4F43C6			363	DC X*FD4F43C6

PAGE 7

FOBARP70 4/30/71

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
700530	FOBARP70			364	DC	X*FOBARP70
700531	FOBARP70			365	DC	X*FOBARP70
700532	FOBARP70			366	DC	X*FOBARP70
700533	FOBARP70			367	DC	X*FOBARP70
700534	FOBARP70			368	DC	X*FOBARP70
700535	FOBARP70			369	DC	X*FOBARP70
700536	FOBARP70			370	DC	X*FOBARP70
700537	FOBARP70			371	DC	X*FOBARP70
700538	FOBARP70			372	DC	X*FOBARP70
700539	FOBARP70			373	DC	X*FOBARP70
700540	FOBARP70			374	DC	X*FOBARP70
700541	FOBARP70			375	DC	X*FOBARP70
700542	FOBARP70			376	DC	X*FOBARP70
700543	FOBARP70			377	DC	X*FOBARP70
700544	FOBARP70			378	DC	X*FOBARP70
700545	FOBARP70			379	DC	X*FOBARP70
700546	FOBARP70			380	DC	X*FOBARP70
700547	FOBARP70			381	DC	X*FOBARP70
700548	FOBARP70			382	DC	X*FOBARP70
700549	FOBARP70			383	DC	X*FOBARP70
700550	FOBARP70			384	DC	X*FOBARP70
700551	FOBARP70			385	DC	X*FOBARP70
700552	FOBARP70			386	DC	X*FOBARP70
700553	FOBARP70			387	DC	X*FOBARP70
700554	FOBARP70			388	DC	X*FOBARP70
700555	FOBARP70			389	DC	X*FOBARP70
700556	FOBARP70			390	DC	X*FOBARP70
700557	FOBARP70			391	DC	X*FOBARP70
700558	FOBARP70			392	DC	X*FOBARP70
700559	FOBARP70			393	DC	X*FOBARP70
700560	FOBARP70			394	DC	X*FOBARP70
700561	FOBARP70			395	DC	X*FOBARP70
700562	FOBARP70			396	DC	X*FOBARP70
700563	FOBARP70			397	DC	X*FOBARP70
700564	FOBARP70			398	DC	X*FOBARP70
700565	FOBARP70			399	DC	X*FOBARP70
700566	FOBARP70			400	DC	X*FOBARP70
700567	FOBARP70			401	DC	X*FOBARP70
700568	FOBARP70			402	DC	X*FOBARP70
700569	FOBARP70			403	DC	X*FOBARP70
700570	FOBARP70			404	DC	X*FOBARP70
700571	FOBARP70			405	DC	X*FOBARP70
700572	FOBARP70			406	DC	X*FOBARP70
700573	FOBARP70			407	DC	X*FOBARP70
700574	FOBARP70			408	DC	X*FOBARP70
700575	FOBARP70			409	DC	X*FOBARP70
700576	FOBARP70			410	DC	X*FOBARP70
700577	FOBARP70			411	DC	X*FOBARP70
700578	FOBARP70			412	DC	X*FOBARP70
700579	FOBARP70			413	DC	X*FOBARP70
700580	FOBARP70			414	DC	X*FOBARP70
700581	FOBARP70			415	DC	X*FOBARP70
700582	FOBARP70			416	DC	X*FOBARP70
700583	FOBARP70			417	DC	X*FOBARP70
700584	FOBARP70			418	DC	X*FOBARP70

PAGE 8

FORNAPR70 6/30/71

LTC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
0003AA	FEF9			419	DC	X*FEF9
0003AC	D645			420	DC	X*D645
0003AE	FSN7			421	DC	X*FEAD
0003AO	FETS			422	DC	X*FE7A
0003H7	FCAC			423	DC	X*FCAC
0003P4	FCAA			424	DC	X*FCAA
0003R5	FRAS			425	DC	X*FB25
0003V9	00			426	DC	X*00
0003V9	00			427	DC	X*00
0003V9	00			428	DC	X*00
0003V9	00			429	DC	X*00
0003V9	00			430	DC	X*00
0003V9	00			431	DC	X*00
0003V9	00			432	DC	X*00
0003V9	00			433	DC	X*00
0003V9	00			434	DC	X*00
0003V9	00			435	DC	X*00
0003V9	00			436	DC	X*00
0003V9	00			437	DC	X*00
0003V9	00			438	DC	X*00
0003V9	00			439	DC	X*00
0003V9	00			440	DC	X*00
0003V9	00			441	DC	X*00
0003V9	00			442	DC	X*00
0003V9	00			443	DC	X*00
0003V9	00			444	DC	X*00
0003V9	00			445	DC	X*00
0003V9	00			446	DC	X*00
0003V9	00			447	DC	X*00
0003V9	00			448	DC	X*00
0003V9	00			449	DC	X*00
0003V9	00			450	DC	X*00
0003V9	00			451	DC	X*00
0003V9	00			452	DC	X*00
0003V9	00			453	DC	X*00
0003V9	00			454	DC	X*00
0003V9	00			455	DC	X*00
0003V9	00			456	DC	X*00
0003V9	00			457	DC	X*00
0003V9	00			458	DC	X*00
0003V9	00			459	DC	X*00
0003V9	00			460	DC	X*00
0003V9	00			461	DC	X*00
0003V9	00			462	DC	X*00
0003V9	00			463	DC	X*00
0003V9	00			464	DC	X*00
0003V9	00			465	DC	X*00
0003V9	00			466	DC	X*00
0003V9	00			467	DC	X*00
0003V9	00			468	DC	X*00
0003V9	00			469	DC	X*00
0003V9	00			470	DC	X*00
0003V9	00			471	DC	X*00
0003V9	00			472	DC	X*00
0003V9	00			473	DC	X*00

PAGE 9

F78APR70 6/30/71

LDC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0005E8 C6				474	DC X'0A'
0005E9 C7				475	DC X'07'
0005FA C7				476	DC X'07'
0005FA C7				477	DC X'07'
0005FC C5				478	DC X'09'
0005FD C9				479	DC X'09'
0005FE C9				480	DC X'09'
0005FF CA				481	DC X'0A'
0005F0 CA				482	DC X'0A'
0005F1 CF				483	DC X'0F'
0005F2 CF				484	DC X'0F'
0005F3 CF				485	DC X'0F'
0005F4 CF				486	DC X'0F'
0005F5 CF				487	DC X'0F'
0005FA 10				488	DC X'10'
0005F7 10				489	DC X'10'
0005F8 10				490	DC X'10'
0005F9 10				491	DC X'10'
0005FA 11				492	DC X'11'
0005FA 11				493	DC X'11'
0005FC 11				494	DC X'11'
0005FD 14				495	DC X'16'
0005FE 18				496	DC X'1A'
0005FF 14				497	DC X'1A'
000600 14				498	DC X'1A'
000601 14				499	DC X'18'
000602 14				500	DC X'14'
000603 14				501	DC X'14'
000604 15				502	DC X'19'
000605 14				503	DC X'1A'
000606 1A				504	DC X'1A'
000607 14				505	DC X'1A'
000608 1A				506	DC X'1A'
000609 1A				507	DC X'1A'
00060A 1A				508	DC X'1A'
00060B 1B				509	DC X'1A'
00060C 1A				510	DC X'1A'
00060D 1B				511	DC X'1B'
00060F 1B				512	DC X'1A'
00060F 1D				513	DC X'1D'
000610 1D				514	DC X'1D'
000611 1F				515	DC X'1F'
000612 27				516	DC X'20'
000613 2C				517	DC X'20'
000614 20				518	DC X'20'
000615 22				519	DC X'22'
000616 23				520	DC X'23'
000617 23				521	DC X'23'
000618 27				522	DC X'27'
000619 00				523	DC X'00'
00061A-01				524	DC X'01'
00061A 05				525	DC X'05'
00061C 05				526	DC X'05'
00061E 05				527	DC X'05'
00061F 05				528	DC X'05'

PAGE 10

F08APR70 6/30/71

LOC	OBJECT CODE	ADDR1	ADDR2	SYMT	SOURCE STATEMENT
000A1F 05				529	DC X'05'
000A20 07				530	DC X'07'
000A21 0F				531	DC X'08'
000A22 0A				532	DC X'0A'
000A23 0E				533	DC X'0E'
000A24 08				534	DC X'08'
000A25 0A				535	DC X'0A'
000A26 0A				536	DC X'0A'
000A27 0C				537	DC X'0C'
000A28 0D				538	DC X'0D'
000A29 0F				539	DC X'0F'
000A2A 10				540	DC X'10'
000A2B 11				541	DC X'11'
000A2C 12				542	DC X'12'
000A2D 12				543	DC X'12'
000A2E 12				544	DC X'12'
000A2F 13				545	DC X'13'
000A30 13				546	DC X'13'
000A31 13				547	DC X'13'
000A32 14				548	DC X'14'
000A33 14				549	DC X'14'
000A34 15				550	DC X'15'
000A35 1A				551	DC X'1A'
000A36 1A				552	DC X'1A'
000A37 19				553	DC X'19'
000A38 19				554	DC X'19'
000A39 19				555	DC X'19'
000A3A 19				556	DC X'19'
000A3B 1A				557	DC X'1A'
000A3C 1C				558	DC X'1C'
000A3D 1C				559	DC X'1C'
000A3E 1C				560	DC X'1C'
000A3F 1C				561	DC X'1C'
000A40 1D				562	DC X'1D'
000A41 1D				563	DC X'1D'
000A42 1E				564	DC X'1E'
000A43 1F				565	DC X'1F'
000A44 1F				566	DC X'1F'
000A45 1F				567	DC X'1F'
000A46 1F				568	DC X'1F'
000A47 21				569	DC X'21'
000A48 21				570	DC X'21'
000A49 22				571	DC X'22'
000A4A 24				572	DC X'24'
000A4B 25				573	DC X'25'
000A4C 2A				574	DC X'2A'
000A4D 00				575	DC X'00'
000A4E 01				576	DC X'01'
000A4F 03				577	DC X'03'
000A50 03				578	DC X'03'
000A51 03				579	DC X'03'
000A52 04				580	DC X'04'
000A53 07				581	DC X'07'
000A54 07				582	DC X'07'
000A55 0D				583	DC X'0D'

PAGE 11

FOBAPR70 6/30/71

LFC	OBJECT CODE	ACDR1	ADDR2	STMT	SOURCE	STATEMENT
000456	02			584	DC	X*0E*
000457	0E			585	DC	X*0E*
000458	0F			586	DC	X*0F*
000459	10			587	DC	X*10*
00045A	1C			588	DC	X*10*
00045B	11			589	DC	X*11*
00045C	12			590	DC	X*11*
00045D	13			591	DC	X*11*
00045E	14			592	DC	X*13*
00045F	14			593	DC	X*14*
000460	14			594	DC	X*14*
000461	15			595	DC	X*14*
000462	15			596	DC	X*15*
000463	15			597	DC	X*15*
000464	15			598	DC	X*15*
000465	15			599	DC	X*15*
000466	17			600	DC	X*17*
000467	18			601	DC	X*18*
000468	18			602	DC	X*18*
000469	18			603	DC	X*18*
000470	18			604	DC	X*18*
000471	1C			605	DC	X*1C*
000472	1C			606	DC	X*1C*
000473	1F			607	DC	X*1E*
000474	1E			608	DC	X*1E*
000475	1F			609	DC	X*1E*
000476	21			610	DC	X*21*
000477	21			611	DC	X*21*
000478	24			612	DC	X*21*
000479	24			613	DC	X*24*
000480	24			614	DC	X*24*
000481	24			615	DC	X*24*
000482	25			616	DC	X*25*
000483	25			617	DC	X*25*
000484	25			618	DC	X*25*
000485	2A			619	DC	X*2A*
000486	2A			620	DC	X*2A*
000487	2A			621	DC	X*2A*
000488	28			622	DC	X*28*
000489	29			623	DC	X*28*
000490	29			624	DC	X*29*
000491	29			625	DC	X*29*
000492	2C			626	DC	X*29*
000493	2A			627	DC	X*29*
000494	25			628	DC	X*29*
000495	2A			629	DC	X*2A*
000496	2A			630	DC	X*2A*
000497	2A			631	DC	X*2A*
000498	2A			632	DC	X*2A*
000499	2A			633	DC	X*20*
000500	2A			634	DC	X*20*
000501	2A			635	DC	X*20*
000502	2C			636	DC	X*2C*
000503	2C			637	DC	X*2C*
000504	2D			638	DC	X*2D*

FOBAPR70 6/30/71

LCC	OBJECT CODE	AMRI ADDR2	STMT	SOURCE STATEMENT
000000	00		639	DC X*20*
000000	2E		640	DC X*2E*
000000	2F		641	DC X*2F*
000000	0F		642	DC X*0F*
000000	12		643	DC X*12*
000000	0C		644	DC X*0C*
000000	0F		645	DC X*0E*
000000	11		646	DC X*11*
000000	13		647	DC X*13*
000000	0D		648	DC X*0D*
000000	10		649	DC X*10*
000000	17		650	DC X*17*
000000	15		651	DC X*15*
000000	CA		652	DC X*CA*
000000	19		653	DC X*19*
000000	09		654	DC X*09*
000000	19		655	DC X*19*
000000	0A		656	DC X*0A*
000000	0A		657	DC X*06*
000000	07		658	DC X*07*
000000	1A		659	DC X*1A*
000000	1E		660	DC X*1E*
000000	19		661	DC X*19*
000000	05		662	DC X*05*
000000	1F		663	DC X*1F*
000000	10		664	DC X*10*
000000	1F		665	DC X*04*
000000	1F		666	DC X*1E*
000000	21		667	DC X*21*
000000	03		668	DC X*03*
000000	28		669	DC X*24*
000000	24		670	DC X*24*
000000	22		671	DC X*22*
000000	01		672	DC X*01*
000000	26		673	DC X*26*
000000	2C		674	DC X*2C*
000000	00		675	DC X*00*
000000	25		676	DC X*25*
000000	29		677	DC X*29*
000000	2A		678	DC X*2A*
000000	2E		679	DC X*2E*
000000	20		680	DC X*20*
000000	2F		681	DC X*2F*
000000	14		682	DC X*14*
000000	02		683	DC X*02*
000000	09		684	DC X*09*
000000	16		685	DC X*16*
000000	20		686	DC X*20*
000000	23		687	DC X*23*
000000	27		688	DC X*27*
000000	27		689	DC X*27*
000000	0000002C		690	END
000000	00000010A		691	-F*44*
000000	000000F79		692	-F*475*
000000			693	-F*3961*

PAGE 13

F084PR70 6/30/71

LDC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0006CC	00000E77			694	=F'3703'
000600	FF4F13F0			655	=X'FF4F10F0'
000A04	40130000			696	=E'0.0625'
0006DF	40200000			697	=E'0.5'
000ADC	41100000			698	=E'1.0'
000AF0	C1200000			699	=E'-2.0'
000AE4	41900000			700	=E'9.0'
000AE8	41300000			701	=E'3.0'

NOT REPRODUCIBLE

APPENDIX E

Assembler Listing of GEN4

PAGE 1

F02APR70 6/30/71

LDC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

000000

```

1 GEN4 START 0
2 SOURCE---L.E. CANNON, UGA, DEPT. OF STAT., JUN 1970
3 PURPOSE---TO SUPPLY THE USER WITH A FAST PRCGEMIR FOR GENERATING
4 RANDOMLY DISTRIBUTED RANDOM NUMBERS.
5 USAGE-----1. CALL RSETUP(P,IR). P IS THE PROBABILITY OF A SUCCESS.
6 IR IS THE RANGE OF THE VARIABLE. RSETUP SETS UP THE TABLE
7 FROM WHICH RANDI GENERATES A RANDOM NUMBER. RSETUP NEED NOT
8 BE CALLED ONLY ONCE FOR A GIVEN SET OF PARAMETERS.
9 P. RPARAM(1) TO 1000 MUST BE AN ODD POSITIVE INTEGER. IT
10 PRIMES THE GENERATING SCHEME. X IS THE RANDOM NUMBER GENERATED.
11 IF X IS RETURNED IN THE REAL MODE.
12 EXAMPLE---CALL RSETUP(10,5,2)
13          DO 1 I=1,100
14          X=RANDI(55,55)
15          ***
16          ***
17          ***
18          ***
19          ***
20          ***
21          ***
22          ***
23          ***
24          ***
25          ***
26          ***
27          ***
28          ***
29          ***
30          ***
31          ***
32          ***
33          ***
34          ***
35          ***
36          ***
37          ***
38          ***
39          ***
40          ***
41          ***
42          ***
43          ***
44          ***
45          ***
46          ***
47          ***
48          ***
49          ***
50          ***
51          ***
52          ***
53          ***
54          ***
55          ***

```

ENTRY RSETUP
R 12(1,15)
DC X(1,15)
DC CL71 RSETUP
STM 14,17,12(13)
RALR 10,0
USING 7,10
LR 14,13
LA 13,ARF2
ST 13,R(0,14)
ST 14,4(C,13)
L 2,7(1,1)
LF 2,0(1,2)
LF 4,7(1,0)
LER 0,4
SFR 4,2
STE 4,0
L 6,4(1,1)
L 6,0(1,6)
LR 5,4
SEL 5,2
SER 6,4
ST 6,NUM
MVI NUM,X(4,4)
AE A,NUM
L 7,7(1,1)
AR 6,7
MER 0,4
RNM 6,7,0-2
LA 4,4
SP 3,3
SER 4,4
LA 12,1
CE 0,LLIM
RC 4,10
ST 3,XST

IF 0=0,LLIM
BRANCH TO LN
OTHERWISE STORE R3=0 INTO XST.

NOT REPRODUCIBLE

CONVERT N TO WORDS.
CONVERTING TO FLOATING POINT.
FPR0=Q=0.
FPR4=1-P=Q.
R6=N.
FPR0=Q=0.
FPR4=1-P=Q.
R6=N.
CONVERT N TO WORDS.
CONVERTING TO FLOATING POINT.
FPR0=Q=0.
FPR4=1-P=Q.
R6=N.

PAGE 7

F084PR70 6/10/71

NOT REPRODUCIBLE

LPC	OBJECT CODE	ADDR1	ADDR2	SYMT	SOURCE STATEMENT
000074	7C03 A2R6	002C8		56	STF 0.P(1)
000075	1A34			57	AR 3.4
000076	7A40 AF4E	002C0		58 L1	AF 4.E-1.0*
000077	10C4			59	DER 0.4
000078	3C06			60	MER 0.6
000079	7C00 A2A2	002R4		61	DE 0.0
000080	3C02			62	MER 0.2
000081	7C03 A2B2	002C4		63	CE 0.0LLIM
000082	4240 A0R8	000CA		64	BC 4.CONVER
000083	7C03 A7R4	002C8		65	STF 0.P(1)
000084	7R40 AE4E	002C0		66	SE 6.E-1.0*
000085	A734 A76B	0007A		67	BYLE 3.4LL
000086	47F3 A0D8	000CA		68	R CONVER
000087	7A40 AF4E	002C0		69 L0	AE 4.E-1.0*
000088	3C04			70	DER 0.4
000089	3C04			71	MER 0.4
000090	7C00 A2A2	002R4		72	DF 0.0
000091	3C02			73	MER 0.2
000092	1A54			74	SR 5.4
000093	7A40 AE4E	002C0		75	SE 6.E-1.0*
000094	5A00 AE4E	002C0		76	A 12.E-1.0*
000095	7A40 A2R2	002C4		77	CE 0.0LLIM
000096	4240 A0R8	000CA		78	BC 4.L0
000097	50C0 A7AE	002C0		79	ST 12.X57
000098	1A54			80	SR 5.4
000099	47F3 A0B8	0007A		81	RC 15.L1
000100	1A54			82	CONVER
000101	1A54			83	SR 5.4
000102	1A54			84	SR 3.3
000103	7A40 A7R4	002C0		85 L3	LE 2.P(3)
000104	7E20 A0R4	002EC		86	AU 2.7FRQ
000105	7C03 A2R6	002C8		87	STE 2.P(3)
000106	5A03 A7R4	002C0		88	L 9.P(1)
000107	47F3 A0C8	0000B		89	SLL 9.8
000108	52C3 A2R6	002C0		90	ST 9.P(3)
000109	47F3 A0RE	0000D		91	BYLE 3.4.L3
000110	1A54			92	SP 12.12
000111	1A54			93	SR 11.11
000112	1A54			94	SR 7.7
000113	1A54			95	SR 6.4
000114	1A54			96	SR 3.4
000115	1A54			97	SR 8.8
000116	5A03 A2R6	002C0		98 L4	L 9.P(3)
000117	47F3 A0C8	002C0		99	SLOL 8.4
000118	5A03 A0R4	00004		100	AR 6.8
000119	1A54	00004		101	SLOL 8.4
000120	1A54	00004		102	AR 7.8
000121	1A54	00004		103	SLOL 8.4
000122	1A54	00004		104	AR 11.8
000123	1A54	00004		105	SLOL 8.4
000124	1A54	00004		106	AR 12.8
000125	1A54	00004		107	BYLE 3.4.L4
000126	1A54	00004		108	ST 6.5
000127	1A54	00004		109	ST 7.5+4
000128	1A54	00004		110	ST 11.5+8

AND STORE Q00M INTO P(1).

FPR4=X+1.

FPR0=P(X)/(X+1).

FPR0=P(X)/(X+1)/(X+1).

FPR0=P(X)/(X+1)/(X+1)/(X+1).

IF FPR0=P(X)/(X+1)/(X+1)/(X+1) Q00M INTO P(1).

BRANCH TO CONVER

OTHERWISE STORE FPR0 INTO P(1).

FPR4=X+1.

CONTINUE

THIS SECTION OF CODE COMPUTES SUCCESSIVE PROBABILITIES UNTIL A PROBABILITY>=LLIM IS ATTAINED, THE PROPER

STARTING POINT IS THEN STORED AND A BRANCH IS

MADE TO L1 FOR FURTHER CALCULATIONS.

R3=1.

CONVERTING TO FIXED POINT.

R9=P(1).

STORE CONVERTED P(1).

CONTINUE

R3=1.

CLEAR R8.

R9=P(1).

R8=0.

R6=SUN(01).

R8=0102.

R7=SUN(0102).

R8=010203.

R11=SUN(010203).

R8=01020304.

R12=SUN(01020304).

CONTINUE.

STORE SUM(01) INTO S(1).

STORE SUM(0102) INTO S(1).

STORE SUM(010203) INTO S(2).

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000124	5000	A6C2		111	ST 12-S+12
000128	A6B2			112	SLL 11,6
00012C	A6B0			113	SLL 7,6
000130	8660	A004		114	SLL 6,6
000134	1A33			115	SR 3,3
000136	1A33			116	SP 8,8
000138	5853	A286		117	L 9,P(13)
00013C	8020	C004		118	SLOL 8,6
000143	1A69			119	SR 6,6
000147	1A78			120	SR 7,6
00014A	1593			121	SR 11,8
000146	1A73			122	SR 8,8
000149	AC03	0004		123	SLOL 9,4
00014C	1B78			124	SR 7,6
00014E	1B84			125	SR 11,8
000150	1B84			126	SR 8,8
000152	8781	0004		127	SLOL 9,4
000154	1B98			128	SP 11,8
000159	A734	A124		129	BR LF 3,6,L5
00015C	1BCC			130	ST 12,12
00015E	AC63	A6CA		131	ST 6,6
000162	5C79	A6CA		132	ST 7,6,6
000164	5081	A6CE		133	ST 11,6+8
00016A	1A77			134	SR 7,7
00016C	1A0C			135	SP 0,0
00016E	5167	3001		136	LA 6,1
000172	1A31			137	SR 3,3
000174	5A23	A286		138	L 7,7,ST
000175	1A84			139	SR 8,8
00017A	5853	A286		140	L 9,P(13)
00017E	6C83	0004		141	SLOL 8,6
000182	5C93	A286		142	ST 9,P(13)
000184	5980	A43A		143	C 8,6,ST
00018A	4780	A192		144	AC 8,8
00018C	1A78			145	AR 7,5
00018F	5970	AFAE		146	C 7,6,7+C000
000194	4720	A1A0		147	RC 2,0,9
000198	1A74			148	SR 7,6
00019A	422C	A6DF		149	ST 2,6,1(12)
00019E	87C4	A192		150	DALE 12,6,6-6
0001A2	1A75			151	AR 7,6
0001A4	1A75			152	AP 2,6
0001A6	8734	A166		153	RYLE 3,6,L7
0001A8	1A04			154	AR 0,6
0001C 1	1506			155	CR 0,6
0001C4	4740	A160		156	RC 4,6
0001C7	5A33	A6C2		157	L 3,5+12
000182	5A33	A6C2		158	S 3,6+8
000184	5A33	A6CE		159	CR 3,12
00018A	153C			160	BC 8,OUT
00018C	4730	A188		161	SR 3,12
00018E	1B3C			162	A 3,6+8
0001C2	5A33	A6CE		163	ST 3,6+8
0001C4	5030	A6CE		164	DS 0H
0001CA	5A09	A75E		165	L 13-AREA+4

PAGE 4

F38APR73 6/30/73

NOT REPRODUCIBLE

LDC	OBJECT CODE	ACDR1 ADDR2	STMT	SOURCE STATEMENT
000101	98FC 000C	0000C	166	LM 14,12,12(13)
000102	92FF 000C	0000C	167	MVI 12(13),X'FF'
000106	07FE		168	BCR 15,14
			169	OROP 10
			170	ENTRY RANR1
			171	USING RANR1,15
			172	DC XLI'D7'
000109	07		173	DC CL7' RANR1'
000110	4C4009C105C2C9	0001B	174	STM 1,3,24(13)
000111	9013 001B		175	SR 0,0
000112	3E09		176	TEMP
000113	47F0 F073		177	2,RPAT
000114	5C20 F504		178	3,LA
000115	5030 F5C8		179	SR 2,2
000116	1321		180	SR 0,0
000117	1850		181	SLDL 2,4
000118	0004	00004	182	C 2,4
000119	5920 F4E4	00008	183	RC 7,5
000120	4740 F05A	0023A	184	RC 4,EX
000121	0020 0004	00004	185	SLDL 2,4
000122	5920 F4FC	0000C	186	C 2,5,4
000123	47F0 F073	0021A	187	RC 11,N2
000124	5320 F4F4	00A0B	188	S 2,4
000125	47F0 F05A	0023A	189	A 4,EX
000126	0004	00004	190	SLDL 2,4
000127	5920 F4F3	00A0B	191	C 2,5,8
000128	4790 F4C4	0022A	192	RC 11,N2
000129	5920 F4FC	00A0C	193	S 2,N,4
000130	47F0 F05A	0023A	194	B 4,EX
000131	0004	00004	195	SLDL 2,4
000132	5920 F4F4	00A04	196	C 2,5,12
000133	47F0 F0C6	001FA	197	RC 11,BRANCH
000134	5320 F5C0	006FC	198	S 7,N,8
000135	4302 F510	006FC	199	IC 0,A12
000136	4001 F5DC	002BC	200	ST 0,RES
000137	9246 F5CC		201	MVI PES,144'
000138	7400 F7DC		202	AE 0,RES
000139	9813 001B	0001B	203	LM 1,3,24(13)
000140	07FE		204	BCR 15,14
000141	5830 F5C8	006FC	205	L 3,LA
000142	0203 F006	00250	206	BRANCH,TEMP
000143	5830 1000	00000	207	L 3,0(1)
000144	5830 1000	00000	208	L 3,0(1)
000145	5C30 F508	006FC	209	ST 3,LA
000146	47F0 F0C4	001EA	210	R BRANCH,4
000147			211	DS 1,RF
000148			212	DS 1,RF
000149			213	DS 1,RF
000150			214	DS 1,RF
000151	3CFEAB92		215	DC E'0,0(0015)
000152			216	DS 256F
000153			217	DS 4F
000154			218	DS 3F
000155	40C27395		219	DC XL4'48C27395'
000156			220	DS 1F

BRANCH TO CALLING ROUTINE.

BRANCH MADE ON FIRST CALL ONLY.
GENERATE NEW U.
STORE FOR NEXT CALL.R2=D1.
IF D1CS THEN
BRANCH TO FX
R2=D1D7.
IF D1D2>=S(11) THEN
BRANCH TO N2
OTHERWISE OBTAIN PROPER TABLE ADDRESS
BRANCH TO EX.
R2=D1D2D7.
IF D1D2D7>=S(12) THEN
BRANCH TO N3
OBTAIN PROPER TABLE ADDRESS
BRANCH TO EX.
R2=D1D2D7D4.
IF D1D2D7D4>=S(13) THEN
GO TO BRANCH(OBTAIN NEW U).
OBTAIN PROPER TABLE ADDRESS.
OBTAIN PROPER DISCRETE VALUE.
CONVERT TO FLOATING POINT.

NORMALIZED FLT. PT. RESULT IN FPRQ.

RETURN.
REPLACES THE INSTRUCTION AT BRANCH.
REPLACE INSTRUCTION AT BRANCH.
LOAD PRIMARILY DONE ON FIRST CALL.

GO TO INSTRUCTION FOLLOWING BRANCH.

PAGE 5

FO8APR70 6/30/71

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0004EC	40070700			221 ZERO	DC X'400C0000'
0006F0				222 A	DS 2000XL1
				223	END
003FCC	411C0000			224	=E'1.0'
000EC4	FFFFFFFF			225	=F'-1'
000FCA	00070701			226	=F'1'
000FCC	00030700			227	=F'0'
000FDD	00000700			228	=F'2000'

NOT REPRODUCIBLE

APPENDIX F

Assembler Listing of GEN5

PAGE 1

FOBAPR70 6/30/71

LCC OBJECT CODE ACORL ADDR2 STMT SOURCE STATEMENT

000000

```

1 GEN5 START 0
2 PRINT ON MORE MODATA
3 *****
4 SOURCE---L.F. CANNON, UGA, DEPT. OF STAT., JUN 1970
5 PURPOSE---TO SUPPLY THE USER WITH A FAST PROCEDURE FOR GENERATING
6 POISSON DISTRIBUTED RANDOM NUMBERS.
7 USAGE-----1. CALL PSETUP(LAM). LAM IS THE MEAN OF THE POISSON DIS-
8 TRIBUTION TO BE GENERATED. PSETUP SETS UP THE TABLE FROM
9 RANDOM GENERATES A RANDOM NUMBER. PSETUP NEED BE CALLED
10 ONLY ONCE FOR A GIVEN LAM.
11 2. X=RVNPOI(IID). IID MUST BE AN ODD INTEGER. IT PRIMES
12 THE GENERATING SCHEME. X REPRESENTS THE POISSON VARIABLE.
13 DO 1 1-100
14 X=RVNPOI(56951)
15 *****
16 METHOD---THE METHOD IS THAT OF MARSAGLIA AS DESCRIBED IN "CONN. ACM
17 4.11(JAN. 1963), 37.
18 *****
19 ENTRY PSETUP
20 PSETUP B 12(0,15)
21 DC X11070
22 DC CL70 PSETUP*
23 STM 14,12,12(13)
24 BALR 10,0
25 USING 4,10
26 LR 14,13
27 LA 13,AKFA
28 ST 13,0(0,14)
29 ST 14,4(0,13)
30 L 2,0(1,1)
31 LE 2,0(1,2)
32 LNER 2,2
33 STE 2,LAM
34 CALL EXP(LAM)
35 LPER 2,2
36 SP 5,5
37 LA 6,4
38 LA 7,10(2)
39 L 12,0(1)
40 SER 4,4
41 CF 0,LLIM
42 BC 4,0
43 STE 0,PIS)
44 ST 5,XST
45 AP 5,6
46 AE 4,0(1)
47 MER 0,2
48 DER 0,6
49 CE 4,CONVER
50 BC 0,LLIM
51 STF 0,PIS)
52 RILE 5,6,12
53 B CONVEP
54 MER 0,2
55 *****
56 FPR2=X.
57 FPR0=EXP(-LAM)*LAM.
58 FPR0=EXP(-LAM)*LAM/X.
59 IF FPR0<LLIM.
60 BRANCH TO LO.
61 OTHERWISE STORE INTO P(0).
62 STORE STARTING POINT.
63 FPR4=X.
64 FPR0=EXP(-LAM)*LAM.
65 FPR0=EXP(-LAM)*LAM/X.
66 IF FPR0<LLIM.
67 BRANCH TO CONVEP.
68 OTHERWISE STORE FPR0 INTO P(1).
69 CONTINUE
70 *****
71 THIS SECTION OF CODE
72 COMPUTES SUCCESSIVE PROBABILITIES

```

PAGE 2

FOBAPR70 6/30/71

USING THE RECURSIVE RELATION UNTIL
A PROABILITY>LLIM IS ATTAINED.
THE PROPER STARTING POINT IS STORED.
AND A BRANCH IS
MADE TO L2 FOR SUB-
SEQUENT CALCULATIONS.

R3=L.
CONVERTING TO FIXED POINT.

R9=PI1).

STORE CONVERTED PI1).
CONTINUE

RR=L.
CLEAR RR.

R9=PI1).

RR=01.

R6=SUM(01).

RR=0102.

R7=SUM(010201).

RR=010201.

R11=SUM(010201).

RR=01020104.

R12=SUM(01020104).

CONTINUE.

STORE SUM(01) INTO S(1).

STORE SUM(0102) INTO S(1).

STORE SUM(010201) INTO S(1).

STORE SUM(01020104) INTO S(1).

R11=SUM(010201)16.

R7=SUM(0102)16.

R4=SUM(01)16.

R3=L.

CLEAR RR.

R9=PI1).

RR=01.

R6=SUM(01)16-SUM(01).

R7=SUM(0102)16-SUM(01).

R11=SUM(010201)16-SUM(01).

CLEAR RR.

RR=02.

R7=SUM(0102).

R11=SUM(0102).

CLEAR RR.

RR=03.

R11=SUM(0102).

CONTINUE.

LCC	OBJECT CODE	ACORI ADDR2	STMT	SOURCE STATEMENT	DER	0,4
000000	1004		67	DER	0,4	12,0,0,1
000001	5100	00E9A	68	A	0,4	0,LLIM
000002	7000	0029A	69	CE	0,4	0,LLIM
000003	4700	0009A	70	BC	0,4	0,LLIM
000004	5200	0029A	71	ST	12,0,0,1	12,0,0,1
000005	4700	0009A	72	RC	15,0,1,2	15,0,1,2
000006	1400	0000A	73	CONVER	SR	5,6
000007	4100	0000A	74	LA	4,4	4,4
000008	1400	0029C	75	SR	3,3	3,3
000009	7000	0029C	76	LE	2,0,0,1	2,0,0,1
000010	7000	0029C	77	AU	2,0,0,1	2,0,0,1
000011	7000	0029C	78	STE	2,0,0,1	2,0,0,1
000012	7000	0029C	79	L	9,0,0,1	9,0,0,1
000013	7000	0029C	80	SLL	9,0,0,1	9,0,0,1
000014	7000	0029C	81	ST	9,0,0,1	9,0,0,1
000015	7000	0029C	82	SR	17,0,1,2	17,0,1,2
000016	7000	0029C	83	SR	11,0,1,1	11,0,1,1
000017	7000	0029C	84	SR	7,7	7,7
000018	7000	0029C	85	SR	6,6	6,6
000019	7000	0029C	86	SR	3,3	3,3
000020	7000	0029C	87	SR	8,4	8,4
000021	7000	0029C	88	SLDL	8,4	8,4
000022	7000	0029C	89	AR	6,6	6,6
000023	7000	0029C	90	SLDL	8,4	8,4
000024	7000	0029C	91	AR	6,6	6,6
000025	7000	0029C	92	SLDL	8,4	8,4
000026	7000	0029C	93	AR	6,6	6,6
000027	7000	0029C	94	SLDL	8,4	8,4
000028	7000	0029C	95	AR	11,0,1,1	11,0,1,1
000029	7000	0029C	96	SLDL	8,4	8,4
000030	7000	0029C	97	AR	12,0,0,1	12,0,0,1
000031	7000	0029C	98	AR	12,0,0,1	12,0,0,1
000032	7000	0029C	99	ST	6,6	6,6
000033	7000	0029C	100	ST	7,7	7,7
000034	7000	0029C	101	ST	11,0,1,1	11,0,1,1
000035	7000	0029C	102	ST	12,0,0,1	12,0,0,1
000036	7000	0029C	103	SL	11,0,1,1	11,0,1,1
000037	7000	0029C	104	SLL	7,7	7,7
000038	7000	0029C	105	SLL	6,6	6,6
000039	7000	0029C	106	SR	3,3	3,3
000040	7000	0029C	107	SR	8,4	8,4
000041	7000	0029C	108	L	9,0,0,1	9,0,0,1
000042	7000	0029C	109	SR	6,6	6,6
000043	7000	0029C	110	SR	7,7	7,7
000044	7000	0029C	111	SR	11,0,1,1	11,0,1,1
000045	7000	0029C	112	SR	12,0,0,1	12,0,0,1
000046	7000	0029C	113	SR	8,4	8,4
000047	7000	0029C	114	SR	7,7	7,7
000048	7000	0029C	115	SR	11,0,1,1	11,0,1,1
000049	7000	0029C	116	SR	12,0,0,1	12,0,0,1
000050	7000	0029C	117	SR	8,4	8,4
000051	7000	0029C	118	SR	7,7	7,7
000052	7000	0029C	119	SR	11,0,1,1	11,0,1,1
000053	7000	0029C	120	SR	12,0,0,1	12,0,0,1
000054	7000	0029C	121	SR	8,4	8,4
000055	7000	0029C	122	SR	7,7	7,7
000056	7000	0029C	123	SR	11,0,1,1	11,0,1,1
000057	7000	0029C	124	SR	12,0,0,1	12,0,0,1
000058	7000	0029C	125	SR	8,4	8,4
000059	7000	0029C	126	SR	7,7	7,7
000060	7000	0029C	127	SR	11,0,1,1	11,0,1,1
000061	7000	0029C	128	SR	12,0,0,1	12,0,0,1
000062	7000	0029C	129	SR	8,4	8,4
000063	7000	0029C	130	SR	7,7	7,7
000064	7000	0029C	131	SR	11,0,1,1	11,0,1,1
000065	7000	0029C	132	SR	12,0,0,1	12,0,0,1
000066	7000	0029C	133	SR	8,4	8,4
000067	7000	0029C	134	SR	7,7	7,7
000068	7000	0029C	135	SR	11,0,1,1	11,0,1,1
000069	7000	0029C	136	SR	12,0,0,1	12,0,0,1
000070	7000	0029C	137	SR	8,4	8,4
000071	7000	0029C	138	SR	7,7	7,7
000072	7000	0029C	139	SR	11,0,1,1	11,0,1,1
000073	7000	0029C	140	SR	12,0,0,1	12,0,0,1
000074	7000	0029C	141	SR	8,4	8,4
000075	7000	0029C	142	SR	7,7	7,7
000076	7000	0029C	143	SR	11,0,1,1	11,0,1,1
000077	7000	0029C	144	SR	12,0,0,1	12,0,0,1
000078	7000	0029C	145	SR	8,4	8,4
000079	7000	0029C	146	SR	7,7	7,7
000080	7000	0029C	147	SR	11,0,1,1	11,0,1,1
000081	7000	0029C	148	SR	12,0,0,1	12,0,0,1
000082	7000	0029C	149	SR	8,4	8,4
000083	7000	0029C	150	SR	7,7	7,7
000084	7000	0029C	151	SR	11,0,1,1	11,0,1,1
000085	7000	0029C	152	SR	12,0,0,1	12,0,0,1
000086	7000	0029C	153	SR	8,4	8,4
000087	7000	0029C	154	SR	7,7	7,7
000088	7000	0029C	155	SR	11,0,1,1	11,0,1,1
000089	7000	0029C	156	SR	12,0,0,1	12,0,0,1
000090	7000	0029C	157	SR	8,4	8,4
000091	7000	0029C	158	SR	7,7	7,7
000092	7000	0029C	159	SR	11,0,1,1	11,0,1,1
000093	7000	0029C	160	SR	12,0,0,1	12,0,0,1
000094	7000	0029C	161	SR	8,4	8,4
000095	7000	0029C	162	SR	7,7	7,7
000096	7000	0029C	163	SR	11,0,1,1	11,0,1,1
000097	7000	0029C	164	SR	12,0,0,1	12,0,0,1
000098	7000	0029C	165	SR	8,4	8,4
000099	7000	0029C	166	SR	7,7	7,7
000100	7000	0029C	167	SR	11,0,1,1	11,0,1,1
000101	7000	0029C	168	SR	12,0,0,1	12,0,0,1
000102	7000	0029C	169	SR	8,4	8,4
000103	7000	0029C	170	SR	7,7	7,7
000104	7000	0029C	171	SR	11,0,1,1	11,0,1,1
000105	7000	0029C	172	SR	12,0,0,1	12,0,0,1
000106	7000	0029C	173	SR	8,4	8,4
000107	7000	0029C	174	SR	7,7	7,7
000108	7000	0029C	175	SR	11,0,1,1	11,0,1,1
000109	7000	0029C	176	SR	12,0,0,1	12,0,0,1
000110	7000	0029C	177	SR	8,4	8,4
000111	7000	0029C	178	SR	7,7	7,7
000112	7000	0029C	179	SR	11,0,1,1	11,0,1,1
000113	7000	0029C	180	SR	12,0,0,1	12,0,0,1
000114	7000	0029C	181	SR	8,4	8,4
000115	7000	0029C	182	SR	7,7	7,7
000116	7000	0029C	183	SR	11,0,1,1	11,0,1,1
000117	7000	0029C	184	SR	12,0,0,1	12,0,0,1
000118	7000	0029C	185	SR	8,4	8,4
000119	7000	0029C	186	SR	7,7	7,7
000120	7000	0029C	187	SR	11,0,1,1	11,0,1,1
000121	7000	0029C	188	SR	12,0,0,1	12,0,0,1
000122	7000	0029C	189	SR	8,4	8,4
000123	7000	0029C	190	SR	7,7	7,7
000124	7000	0029C	191	SR	11,0,1,1	11,0,1,1
000125	7000	0029C	192	SR	12,0,0,1	12,0,0,1
000126	7000	0029C	193	SR	8,4	8,4
000127	7000	0029C	194	SR	7,7	7,7
000128	7000	0029C	195	SR	11,0,1,1	11,0,1,1
000129	7000	0029C	196	SR	12,0,0,1	12,0,0,1
000130	7000	0029C	197	SR	8,4	8,4
000131	7000	0029C	198	SR	7,7	7,7
000132	7000	0029C	199	SR	11,0,1,1	11,0,1,1
000133	7000</					

PAGE 3

FORAPR70 6/30/71

LNC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

```

00138 5060 A694      006AC 122 ST 6,N
00139 5070 A69F      006BD 123 ST 7,N+4
00140 5080 A6A2      006BE 124 ST 11,N+R
00141 1877           006BF 125 SR 7,7
00142 1800           00001 126 SR 0,0
00143 4160 0001      00294 127 LA 6,1
00144 5820 A282      00294 128 L6
00145 1898           0029C 129 L7
00146 5953 A284      0029C 130 L7
00147 5953 A284      0029C 131 L 9,9(13)
00148 5953 A284      0029C 132 SLDL 8,4
00149 5953 A284      0029C 133 ST 9,P(13)
00150 5953 A284      0029C 134 C 8,8+0
00151 5953 A284      0029C 135 BC 8,8
00152 5953 A284      0029C 136 AR 7,7
00153 5953 A284      0029C 137 C 7,7+2000
00154 5953 A284      0029C 138 BC 2,1,9
00155 5953 A284      0029C 139 SR 7,6
00156 5953 A284      0029C 140 STC 7,4(12)
00157 5953 A284      0029C 141 MYLF 12,6,8-4
00158 5953 A284      0029C 142 AR 7,6
00159 5953 A284      0029C 143 AR 2,6
00160 5953 A284      0029C 144 MYLE 3,5,1,7
00161 5953 A284      0029C 145 AR 0,5
00162 5953 A284      0029C 146 CR 0,5
00163 5953 A284      0029C 147 BC 4,1,5
00164 5953 A284      0029C 148 L 3,5,1,2
00165 5953 A284      0029C 149 S 3,N+R
00166 5953 A284      0029C 150 CR 3,1,2
00167 5953 A284      0029C 151 AC 3,1,2
00168 5953 A284      0029C 152 SR 3,1,2
00169 5953 A284      0029C 153 A 3,N+R
00170 5953 A284      0029C 154 ST 3,N+R
00171 5953 A284      0029C 155 DS 0,N
00172 5953 A284      0029C 156 L 13,AREA+4
00173 5953 A284      0029C 157 LM 14,12,12(13)
00174 5953 A284      0029C 158 MVI 12(13),X+FF
00175 5953 A284      0029C 159 BCR 15,14
00176 5953 A284      0029C 160 DROP 10
00177 5953 A284      0029C 161 ENTRY RANP01
00178 5953 A284      0029C 162 USING RANP01,15
00179 5953 A284      0029C 163 DC XLI,07
00180 5953 A284      0029C 164 DC CL7, RANP01
00181 5953 A284      0029C 165 STM 1,3,24(13)
00182 5953 A284      0029C 166 SER 0,0
00183 5953 A284      0029C 167 IFMP
00184 5953 A284      0029C 168 M 2,SPAT
00185 5953 A284      0029C 169 ST 3,LA
00186 5953 A284      0029C 170 SR 2,2
00187 5953 A284      0029C 171 SR 0,0
00188 5953 A284      0029C 172 SLDL 2,4
00189 5953 A284      0029C 173 C 2,5
00190 5953 A284      0029C 174 BC 4,EX
00191 5953 A284      0029C 175 SLDL 2,4
00192 5953 A284      0029C 176 C 2,5+4

```

STORE R4 INTO N(01).
STORE R7 INTO N(11).
STORE R11 INTO N(12).

R3=L.
LOAD R2 WITH STARTING VALUE.
CLEAR RA.
R9=P(11).
R4=01,RR=02,RR=03,RR=04.
RESTORE SHIFTED P(11).
IF RR=0.
BRANCH TO L8.
OTHERWISE STORE R2(DISCRETE R.V.)
CHECK THAT TABLE LENGTH IS NOT EXCEEDED.
THE NUMBER OF TIMES CORRESPONDING
TO THE DIGIT IN RA.
INCREMENT DISCRETE R.V.
CONTINUE.
CONTINUE L6 LOOP
UNTIL FIRST 4 DIGITS OF
EACH P(11) HAVE BEEN SHIFTED.
THIS SECTION OF CODE
RESPTS N(3)
IF TABLE LENGTH IS EXCEEDED.
BRANCH TO CALLING ROUTINE.

BRANCH MADE ON FIRST CALL ONLY.
GENERATE NEW U.
STORE FOR NEXT CALL.

P2=01.
IF N(1) THEN
BRANCH TO EX
P2=01D2.
IF 01D2=S(1) THEN

PAGE 4
FO8APR70 6/30/71

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0001E4	4780 F036		001FC	177	PC 11,N2
0001F8	5820 F4F2		006AC	178	S 2,N
0001FC	47F0 F05A		00214	179	B EX
0001F0	8020 F004		00004	180	N2
0001F4	5920 F4EA		006A4	181	SLOL 2,4
0001F8	4780 F04A		00204	182	C 2,5+8
0001FC	5820 F4F6		00680	183	BC 11,N3
000200	47F0 F05A		00214	184	S 2,N+4
000204	8020 F004		00604	185	B EX
000208	5820 F4FE		006A8	186	SLOL 2,4
00020C	4780 F004		001C0	187	C 2,5+12
000210	5820 F4FA		00684	188	BC 11,BRANCH
000214	4320 F50A		006C4	189	S 2,N+8
000218	5000 F00A		00290	190	IC 0,A(2)
00021C	9240 F006		00290	191	ST 0,RES
000220	7A00 F006		00290	192	MVI RES,X+46
000224	9810 F01A		00018	193	AE 0,RES
000228	C7FE			194	L4 1,3,24(113)
00022A	5830 F502			195	BCR 15,14
00022E	D030 F006			196	L 3,LA
000234	5E30 1000			197	MVC BRANCH,TEMP
000238	5830 3G00			198	L 3,0(1,1)
00023C	5030 F502			199	L 3,0(1,3)
000240	47F0 F00A		001C4	200	ST 3,LA
000244				201	B BRANCH+4
00028C				202	DS 18F
000290				203	DS 1F
000294				204	DS 1F
000298	3CF8A8B2			205	DS 1F
00029C				206	DC F,0,0,0,0,15
0002A0				207	DS 256F
0002AC				208	DS 4F
0006A8	48C27395			209	DS 3F
0006AC				210	DC XL4,48C27395
0006C0	4000C000			211	DS 1F
0006C4				212	DC X,4000C000
000F98	00003001			213	DS 2000XL1
000F9C	41103000			214	END
000FA0	00010000			215	=F,1
000FA4	00000700			216	=E,1
				217	=F,0
					=F,20C0

BRANCH TO N2
OTHERWISE OBTAIN PROPER TABLE ADDRESS
BRANCH TO EX.
R2=D10203.
IF D10203>S(2) THEN
BRANCH TO N3
OBTAIN PROPER TABLE ADDRESS
BRANCH TO EX.
R2=D10203D4.
IF D10203D4>S(3) THEN
GO TO BRANCH(OBTAIN NEW U).
OBTAIN PROPER TABLE ADDRESS.
OBTAIN PROPER DISCRETE VALUE.
CONVERT TO FLOATING POINT.
NORMALIZED FLT. PT. RESULT IN FPR0.
RETURN.
REPLACES THE INSTRUCTION AT BRANCH.
REPLACE INSTRUCTION AT BRANCH.
LOAD PRIMER(ONLY DONE ON FIRST CALL).
GO TO INSTRUCTION FOLLOWING BRANCH.

APPENDIX G

Assembler Listing of GEN6

FOBAPR70 6/30/71

LOC OBJECT CODE ANORI ADDR2 STPT SOURCE STATEMENT

000000
1 GEN6 START 0
2
3
4 SOURCE---L.F. CANNON-UGA,DEPT. OF STAT., JUN 1970
5 PURPOSE---TO SUPPLY THE USER WITH A FAST PROCEDURE FOR GENERATING NE-
6 GATIVE BINOMIAL DISTRIBUTED RANDOM NUMBERS.
7 USAGE-----1. CALL MRSETUP,IRI. PROBABILITY OF A SUCCESS, IR-NUM-
8 RI GENERATES A RANDOM NUMBER. MRSETUP SETS UP THE TABLE FROM WHICH RAN-
9 ONCE FOR A GIVEN P AND IR.
10 2. X-RANGE(100), 'IDD' MUST BE AN ODD INTEGER, IT PRIMES
11 THE GENERATING SCHEME. X REPRESENTS THE NUMBER OF FAILURES
12 BEFORE THE IRI TH SUCCESS.
13 EXAMPLE---CALL MRSETUP(0.5,10)
14 DO 1 1-1,100
15 X-RANGE(1-095)
16
17 METHOD---THE METHOD IS THAT OF MARSAGLIA AS DESCRIBED IN COMM. ACM
18 6,1(JAN. 1963),37.
19
20 ENTRY MRSETUP
21 8 1210,151
22 DC XL1,07.
23 DC CL7, MRSETUP
24 SIM 14,17,12(13)
25 HALR 10,0
26 USING 0,10
27 LR 14,13
28 LA 17,AREA
29 ST 13,410,14
30 ST 14,410,13
31 L 4,01,11
32 LE 4,01,41
33 LE 2,0E,1
34 SER 2,4
35 L 6,41,11
36 L 6,01,41
37 ST 6,SUC
38 MVI SUC,X*46
39 LE 0,0E,1
40 SER 6,6
41 AE 6,SUC
42 BC R,SKIP
43 L 7,0E,1
44 AR 6,7
45 RFR 0,4
46 RHM 6,7,0-2
47 SR 5,5
48 LA 6,4
49 LA 7,1020
50 L 12,0F,1
51 SER 4,4
52 CE 0,LLIM
53 BC 4,LO
54 STE 0,PI51
55 ST 5,XST

LOAD ADDRESS OF P.
LOAD P INTO FPR4.
CALCULATE Q01-P.
LOAD ADDRESS OF R.
CONVERT R TO FLOATING POINT.

FPR4-R INORMALIZED FLT. PT.1.
IF R-0 BRANCH TO SKIP

CALCULATE P00R.

IF P00RCLLIM.
BRANCH TO L0.
OTHERWISE STORE P00R INTO P(0).
STORE R5-0 INTO XST.

PAGE 2

FOGAPR70 6/30/71

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000079	1A96			56	AR 5.4
00007A	7A43	000080		57	AE 4.5E+10
00007B	1C06			58	MFR 0.6
000080	1D04			59	DER 0.4
000082	1C02			60	MFR 0.2
000084	7900	00284		61	CE 0.2
000085	4740	0008F		62	BC 4.0LLIM
000086	7035	00288		63	STE 0.0P151
000087	7A63	0008F		64	AE 4.5E+10
000088	4756	0007A		65	AXLE 5.4E+12
000089	47F3	0008F		66	B CONVER
000090	7A43	0008F		67	AE 4.5E+10
000091	1C04			68	MFR 0.6
000092	70C4			69	DER 0.4
000093	1C07			70	MFR 0.2
000094	5ACD	0008B		71	A 12.5E+10
000095	7A43	0008F		72	AE 4.5E+10
000096	7903	00284		73	CE 0.2
000097	4743	0008F		74	BC 4.0LLIM
000098	50C0	00288		75	ST 12.5E+12
000099	47F0	0007A		76	PC 15.4E+12
000100	1A96			77	SW 5.6
000101	1A96	0008F		78	LA 4.4
000102	1A96			79	SR 3.3
000103	7A43	00284		80	LE 2.0P131
000104	7A43	0008F		81	AU 2.0P131
000105	7A43	00288		82	STE 2.0P131
000106	5493	00288		83	L 9.4
000107	5500	0008A		84	SLL 9.4
000108	5500	00288		85	ST 9.4P131
000109	7A43	0008F		86	DALE 3.4E+13
000110	1A96			87	SR 12.12
000111	1A96			88	SR 11.11
000112	1A96			89	SR 7.7
000113	1A96			90	SR 6.4
000114	1A96			91	SR 3.3
000115	1A96			92	SR 0.2
000116	1A96			93	L 9.4P131
000117	1A96	00288		94	AR 6.4
000118	1A96	0008F		95	SLL 8.4
000119	1A96	0008F		96	AR 7.8
000120	1A96	0008F		97	AR 11.8
000121	1A96	0008F		98	AR 12.8
000122	1A96	0008F		99	AR 13.4
000123	1A96	0008F		100	AR 14.4
000124	1A96	0008F		101	AR 15.4
000125	1A96	0008F		102	AR 16.4
000126	1A96	0008F		103	AR 17.4
000127	1A96	0008F		104	AR 18.4
000128	1A96	0008F		105	AR 19.4
000129	1A96	0008F		106	AR 20.4
000130	1A96	0008F		107	AR 21.4
000131	1A96	0008F		108	AR 22.4
000132	1A96	0008F		109	AR 23.4
000133	1A96	0008F		110	AR 24.4

FPR0=X+1.
FPR0=PI(X)*(X+R).
FPR0=PI(X)*(X+R)/(X+1).
FPR0=PI(X)*(X+R)/(X+1).
IF PI(X) < 0.1LLIM.
BRANCH TO CONVER.
OTHERWISE STORE FPR0=PI(X) INTO P151.
INCREMENT X+R.
CONTINUE.

THIS SECTION OF CODE
CALCULATES SUCCESSIVE PROBABILITIES, USING THE RECURSIVE RELATION, UNTIL A PROBABILITY > 0.1LLIM IS OBTAINED.
THE CORRESPONDING X-VALUE IS STORED INTO XST AND A BRANCH IS MADE TO L2 FOR FURTHER CALCULATIONS.

A3=1.
CONVERTING TO FIXED POINT.

CONTINUE

A3=1.
CLEAR AB.
PQ=PI1.
R0=0.
R6=SUM(D1).
R8=0.
R7=SUM(D1D2).
R9=0.
R11=SUM(D1D2D3).
R12=SUM(D1D2D3D4).
CONTINUE.
STORE SUM(D1) INTO S1(1).
STORE SUM(D1D2) INTO S1(2).
STORE SUM(D1D2D3) INTO S1(3).
STORE SUM(D1D2D3D4) INTO S1(4).
R11=SUM(D1D2D3D4).
R7=SUM(D1D2D3D4).
R6=SUM(D1D2D3D4).
R3=1.

FOR APR 70 6/30/71

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
00012C	1A09			111 LS	SR
00012E	5E03 42A6			112	L
000132	8E02 0004			113	SLNL
000136	1A05			114	SR
00013A	1A78			115	SR
00013A	1A09			116	SR
00013C	1A09			117	SR
00013C	0E03 0004			118	SLNL
000142	1A78			119	SR
000144	1A04			120	SR
000146	1A04			121	SR
00014C	9A73 0004			122	SLNL
00014E	1A78			123	SR
00014E	8736 A11A			124	SR
000152	1ECC			125	SR
000154	5063 46A6			126	ST
000159	5C79 46BA			127	ST
00015C	5C03 4A8E			128	ST
00015A	1A77			129	SR
000157	1A03			130	SR
00015A	416A 0001			131	LA
000159	1A33			132	SR
00015A	9229 129F			133	L
00015E	1A04			134	SR
000170	5A53 42A6			135	L
000174	6200 0004			136	SLNL
000178	5C03 12A6			137	ST
00017C	5C03 4E8A			138	BC
00017A	4783 418A			139	BC
000184	1A78			140	AR
000186	5973 4FAE			141	AR
00018A	4723 4196			142	AC
00018F	1A7A			143	SR
000190	422C 46CE			144	STC
000194	87C6 417E			145	SR
000198	1A76			146	AR
00019A	1A7A			147	AR
00019C	8736 415C			148	AR
0001A7	1A09			149	AR
0001A2	1904			150	CR
0001A6	4743 415A			151	AC
0001A9	5A70 4A82			152	L
0001AC	5A70 4A9E			153	S
0001B7	193C			154	CR
0001B2	4783 41AF			155	BC
0001B6	1F3C			156	SR
0001B8	5A30 4A8E			157	A
0001BC	5073 46BE			158	ST
0001C0	8E1C			159	DS
0001CA	5E09 4252			160	L
0001CA	985C 000C			161	LM
0001CA	52FF 000C			162	MVI
0001CC	07FE			163	BCR
				164	DROP
				165	ENTRY

CLEAR R4.
 R9=PI11.
 R4=01.
 R4=SUM(D11)+16-SUM(D11).
 R7=SUM(D12)+16-SUM(D11).
 R11=SUM(D12D3)+16-SUM(D11).
 CLEAR R8.
 R8=02.
 R7=R7-SU*(D21).
 P11=R11-SUM(D21).
 CLEAR R8.
 PA=03
 R11=R11-SUM(D33).
 CONTINUE.
 STORE R6 INTO M(3).
 STORE R7 INTO N(1).
 STORE R11 INTO N(2).
 R2=1.
 LOAD R2 WITH STARTING VALUE.
 CLEAR R4.
 R9=PI11.
 R2=01, R4=02, R9=03, R8=04.
 R2=01, R4=02, R9=03, R8=04.
 IF R4=0.
 R4=0.
 OTHERWISE STORE R2 INTO DISCRETE R.V.1
 CHECK THAT TABLE LENGTH IS NOT EXCEEDED.
 THE NUMBER OF TIMES CORRESPONDING
 TO THE DIGIT IN R8.
 INCREMENT DISCRETE R.V.
 CONTINUE.
 CONTINUE L6 LOOP
 UNTIL FIRST 4 DIGITS OF
 EACH P(1) HAVE BEEN SHIFTED.
 THIS SECTION OF CODE
 RESETS N(3)
 IF TABLE LENGTH IS EXCEEDED.

BRANCH TO CALLING ROUTINE.

FO8APR70 6/30/71

LDC	OBJECT CODE	ADDR1	ADDR2	VTMT	SOURCE STATEMENT
000106	000106			166	USING RANER1,15
000107	000107			167	OC REL'07.
000108	000108			168	OC CLT' RANER1'
000109	000109			169	STM 1,3,25(13)
000110	000110			170	OC RANER1
000111	000111			171	BRANCH
000112	000112			172	TEMP
000113	000113			173	ST 3,1A
000114	000114			174	SR 2,2
000115	000115			175	SR 2,2
000116	000116			176	SLDL 2,4
000117	000117			177	C 2,5
000118	000118			178	OC 4,EX
000119	000119			179	SLDL 2,4
000120	000120			180	C 2,5,4
000121	000121			181	AC 11,N2
000122	000122			182	S 2,N
000123	000123			183	B 6X
000124	000124			184	SLDL 2,4
000125	000125			185	C 2,5,4
000126	000126			186	AC 11,N3
000127	000127			187	S 2,N,4
000128	000128			188	R 6X
000129	000129			189	SLDL 2,4
000130	000130			190	C 2,5,4
000131	000131			191	AC 11,N3
000132	000132			192	S 2,N,4
000133	000133			193	IC 0,4(2)
000134	000134			194	ST 2,4,5
000135	000135			195	MVI RES,X'46'
000136	000136			196	AF 0,RES
000137	000137			197	LM 1,3,2,1(13)
000138	000138			198	RGR 15,14
000139	000139			199	TEMP
000140	000140			200	MVC RANER1,TEMP
000141	000141			201	L 3,0(1,1)
000142	000142			202	L 3,0(1,3)
000143	000143			203	ST 3,1A
000144	000144			204	R BRANCH,4
000145	000145			205	DS 1,AF
000146	000146			206	DS 1F
000147	000147			207	DS 1F
000148	000148			208	DS 1F
000149	000149			209	DS 2,56F
000150	000150			210	DS 2,56F
000151	000151			211	DS 4F
000152	000152			212	DS 3F
000153	000153			213	DS 3F
000154	000154			214	DS 1F
000155	000155			215	DS 1F
000156	000156			216	DS 200,11
000157	000157			217	END
000158	000158			218	-E'1'
000159	000159			219	-E'1'
000160	000160			220	-E'1'

BRANCH MADE ON FIRST CALL ONLY.
GENERATE NEW U.
STORE FOR NEXT CALL.

R2-D1.
IF 01<5 THEN
BRANCH TO EX
P2-D107.
IF 01022=5(11) THEN
BRANCH TO N2
OTHERWISE OBTAIN PROPER TABLE ADDRESS
BRANCH TO EX.
P2-D102D3.
IF 01022=5(2) THEN
BRANCH TO N3
OBTAIN PROPER TABLE ADDRESS
BRANCH TO EX.
P2-D102D4.
IF 01023=5(13) THEN
GO TO BRANCHIDRAIN NEW U.
OBTAIN PROPER TABLE ADDRESS.
OBTAIN PROPER DISCRETE VALUE.
CONVERT TO FLOATING POINT.
NORMALIZED FLT. PT. RESULT IN FPRC.

RETURN.
REPLACES THE INSTRUCTION AT BRANCH.
REPLACE INSTRUCTION AT BRANCH.
LOAD PRIMARILY DONE ON FIRST CALL.

GO TO INSTRUCTION FOLLOWING BRANCH.

PAGE 5

F08APR70 6/30/71

LOC	OBJECT CODE	ADDR1 ADDR2	STMT	SOURCE STATEMENT
CONFAC	0C070000		221	-F000
000ECO	00000700		222	-F02000

APPENDIX H
Assembler Listing of GEN7

NOT REPRODUCIBLE

PAGE 1

F04APR70 6/30/71

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

000000

```

1 GEN7 START 0
2 .....
3 SOURCE---L.E. CANNON, UCA, DEPT. OF STAT., JUN 1970.
4 PURPOSE---TO SUPPLY THE USER WITH A FAST PROCEDURE FOR GENERATING DIS-
5 CRET RANDOM VARIABLES WITH A SPECIFIED PROBABILITY.
6 USAGE---1. CALL DSETUP(P,X,N), 'P' AND 'X' ARE VECTORS OF PROBA-
7 BILITIES AND THE CORRESPONDING DISCRETE VARIABLES ( MUST
8 BE INTEGERS). 'N' IS THE NUMBER OF ELEMENTS IN THE
9 VECTORS. DSETUP SETS UP THE TABLE FROM WHICH RANDIS GENER-
10 ATES A RANDOM NUMBER. DSETUP NEED BE CALLED ONLY ONCE FOR
11 A GIVEN 'P', 'X', AND 'N'.
12 2. RANDIS(TOD), 'TOD' MUST BE AN ODD INTEGER, IT PRIMES
13 THE GENERATING SCHEME. X REPRESENTS THE DESIRED RANDOM
14 VARIABLE. RETURNED IN FLOATING POINT.
15 METHOD---THE METHOD IS THAT OF MARSAGLIA AS DESCRIBED IN 'COMM. ACM'
16 8, 11, JAN 1963, 37.
17 .....
18 ENTRY TSETUP
19 0 1210,15)
20 DC XL1'07.
21 NC CL7' TSETUP
22 STN 14,12,12(13)
23 BALR 17,0
24 USING 0,10
25 LP 14,13
26 LA 13,APFA
27 ST 13,8(0,14)
28 ST 14,4(0,13)
29 L 5,8(1,1)
30 L 5,0(1,5)
31 SLL 5,2
32 L 11,0(1,1)
33 LA 4,4
34 SR 4,4
35 SP 3,3
36 LE 2,0(1,11)
37 AU 2,2EFD
38 STF 2,P(1)
39 L 9,P(1)
40 SLL 9,8
41 ST 9,P(1)
42 DMLE 3,4,13
43 SR 12,12
44 SR 11,11
45 SR 7,7
46 SR 6,6
47 SP 3,3
48 L4
49 L 9,P(1)
50 SLDL 9,4
51 AR 6,8
52 SLDL 8,4
53 AR 7,8
54 SLDL 8,4
55 AR 11,8

```

LOAD N INTO R5
CONVERT N TO WORDS (1 WORD= 4 BYTES).
LOAD ADDRESS OF P VECTOR INTO R11.

R3=1.
LOAD LOAD PPR2 WITH P(1) FROM PARM. LIST.
CONVERTING TO FIX POINT.

CONTINUE

R3=1.
CLEAR RA.
R9=P(1).
R4=01.
R6=SUM(01).
R7=SUM(01D2).
R8=01D203.
R11=SUM(01D203).

NOT REPRODUCIBLE

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000076	8180 0004	00004	00004	56	SLML 9.4
000077	1AC9	0005E	0005E	57	AP 12.8
000078	B734 A04C	0005E	0005E	58	DXLE 3.4.14
000079	5CA3 A00A	0005E	0005E	59	ST 6.5
000080	5070 A00E	0005E	0005E	60	ST 7.5.4
000081	5070 A012	0005E	0005E	61	ST 11.5.8
000082	5070 A016	0005E	0005E	62	ST 12.5.12
000083	5070 A004	0005E	0005E	63	SL 11.4
000084	8370 0004	00004	00004	64	SL 7.4
000085	8370 0004	00004	00004	65	SL 6.4
000086	8370 0004	00004	00004	66	SR 3.4
000087	8370 0004	00004	00004	67	SR 8.4
000088	8370 0004	00004	00004	68	L 9.4(3)
000089	8370 0004	00004	00004	69	SL 8.4
000090	8370 0004	00004	00004	70	SR 4.4
000091	8370 0004	00004	00004	71	SR 7.8
000092	8370 0004	00004	00004	72	SR 11.8
000093	8370 0004	00004	00004	73	SR 8.4
000094	8370 0004	00004	00004	74	SL 8.4
000095	8370 0004	00004	00004	75	SR 7.8
000096	8370 0004	00004	00004	76	SR 11.8
000097	8370 0004	00004	00004	77	SR 8.8
000098	8370 0004	00004	00004	78	SL 8.4
000099	8370 0004	00004	00004	79	SL 11.8
000100	8370 0004	00004	00004	80	DXLE 3.4.15
000101	8370 0004	00004	00004	81	SR 12.12
000102	8370 0004	00004	00004	82	ST 6.4
000103	8370 0004	00004	00004	83	ST 7.4.4
000104	8370 0004	00004	00004	84	ST 11.4.8
000105	8370 0004	00004	00004	85	SR 7.7
000106	8370 0004	00004	00004	86	SR 0.0
000107	8370 0004	00004	00004	87	L 6.4.11
000108	8370 0004	00004	00004	88	L 11.4.11
000109	8370 0004	00004	00004	89	SR 3.4
000110	8370 0004	00004	00004	90	SR 8.4
000111	8370 0004	00004	00004	91	L 9.4(3)
000112	8370 0004	00004	00004	92	SL 8.4
000113	8370 0004	00004	00004	93	ST 9.4(3)
000114	8370 0004	00004	00004	94	C 8.4.11
000115	8370 0004	00004	00004	95	DC 8.4.11
000116	8370 0004	00004	00004	96	AR 7.4
000117	8370 0004	00004	00004	97	C 7.4.2000
000118	8370 0004	00004	00004	98	RC 2.4.9
000119	8370 0004	00004	00004	99	L 2.0(3.11)
000120	8370 0004	00004	00004	100	SR 7.6
000121	8370 0004	00004	00004	101	STC 2.4.112
000122	8370 0004	00004	00004	102	DXLE 12.6.4-4
000123	8370 0004	00004	00004	103	AR 7.6
000124	8370 0004	00004	00004	104	DXLE 3.4.17
000125	8370 0004	00004	00004	105	AR 0.4
000126	8370 0004	00004	00004	106	CR 0.4
000127	8370 0004	00004	00004	107	RC 4.4.6
000128	8370 0004	00004	00004	108	L 3.5.12
000129	8370 0004	00004	00004	109	S 3.4.4
000130	8370 0004	00004	00004	110	CH 3.12

R8-D1020304.
 R12-SUM(01020304).
 CONTINUE.
 STORE SUM(01) INTO S(01).
 STORE SUM(02) INTO S(02).
 STORE SUM(03) INTO S(03).
 STORE SUM(04) INTO S(04).
 R11-SUM(01020304)=16.
 R7-SUM(01020304)=16.
 R4-SUM(01020304)=16.
 R1=1.
 CLEAR R4.
 R9=0.
 R8-D1.
 R4-SUM(01020304)=16-SUM(01).
 R7-SUM(01020304)=16-SUM(02).
 R11-SUM(01020304)=16-204(01).
 CLEAR R4.
 R4=0.
 R7-R7-SUM(02).
 R11-R11-SUM(02).
 CLEAR R4.
 R8-D3.
 R11-R11-SUM(03).
 CONTINUE.
 STORE R4 INTO M(01).
 STORE R7 INTO M(02).
 STORE R11 INTO M(03).
 LOAD R11 WITH ADDRESS OF X VECTOR.
 R3=1.
 CLEAR R4.
 R9=0.
 R8-D1, R8-D2, R8-D3, R8-D4.
 RESTORE SHIFTED P(1).
 IF R8=0.
 BRANCH TO L8.
 OTHERWISE STORE R2IDISCRETE R.V.
 CHECK THAT TABLE LENGTH IS NOT EXCEEDED.
 LOAD R2 WITH DISCRETE VALUE.
 THE NUMBER OF TIMES CORRESPONDING
 TO THE DIGIT IN R4.
 CONTINUE.
 CONTINUE L8 LOOP
 UNTIL FIRST 4 DIGITS OF
 EACH P(1) HAVE BEEN SHIFTED.
 THIS SECTION OF CODE
 RESETS M(1)

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000126	4780 A122		00134	111	AC 9.0UT
000127	181C			112	SR 3.12
000128	5830 A622		00634	113	A 3.0M+8
000129	5830 A622			114	ST 3.0M+8
000130	5830 A622			115	DS 0M
000131	5830 A622			116	L 13.0M+8
000132	5830 A622			117	L 14.12.12(113)
000133	98EC 000C			118	BCR 15.14
000134	98EC 000C			119	DRCP 10
000135	98EC 000C			120	ENTRY RANDIS
000136	98EC 000C			121	USING RANDIS.15
000137	98EC 000C			122	DC 41.07
000138	98EC 000C			123	DC 41.07
000139	98EC 000C			124	ST 0.3.20(13)
000140	98EC 000C			125	SFR 0.0
000141	98EC 000C			126	R 2.0M
000142	98EC 000C			127	M 2.0M
000143	98EC 000C			128	ST 3.1A
000144	98EC 000C			129	SR 2.7
000145	98EC 000C			130	SR 0.0
000146	98EC 000C			131	SLOL 2.4
000147	98EC 000C			132	C 2.5
000148	98EC 000C			133	AC 4.0M
000149	98EC 000C			134	SLOL 2.4
000150	98EC 000C			135	C 2.5+4
000151	98EC 000C			136	PC 11.02
000152	98EC 000C			137	S 2.0M
000153	98EC 000C			138	B EX
000154	98EC 000C			139	SLOL 2.4
000155	98EC 000C			140	C 2.5+8
000156	98EC 000C			141	RC 11.0M
000157	98EC 000C			142	S 2.0M+4
000158	98EC 000C			143	R EX
000159	98EC 000C			144	SLOL 2.4
000160	98EC 000C			145	C 2.5+12
000161	98EC 000C			146	BC 11.0M
000162	98EC 000C			147	S 2.0M+4
000163	98EC 000C			148	IC 0.0(12)
000164	98EC 000C			149	ST 0.0(12)
000165	98EC 000C			150	RES.0.0
000166	98EC 000C			151	AE 0.0(12)
000167	98EC 000C			152	LM 0.0.20(13)
000168	98EC 000C			153	ACR 15.14
000169	98EC 000C			154	L 3.1A
000170	98EC 000C			155	BRANCH.15
000171	98EC 000C			156	L 3.0(12)
000172	98EC 000C			157	L 3.0(12)
000173	98EC 000C			158	ST 3.1A
000174	98EC 000C			159	B BRANCH+4
000175	98EC 000C			160	DS 15F
000176	98EC 000C			161	DS 15F
000177	98EC 000C			162	DS 256F
000178	98EC 000C			163	DS 4F
000179	98EC 000C			164	DS 3F
000180	98EC 000C			165	DC XL4.48C27395

IF TABLE LENGTH IS EXCEEDED.

BRANCH TO CALLING ROUTINE.

BRANCH MADE ON FIRST CALL ONLY.
GENERATE NEW U.
STORE FOR NEXT CALL.

R2=01.
IF 01CS THEN
BRANCH TO EX
R2=0107.
IF 0102>S(1) THEN
BRANCH TO N2
OTHERWISE OBTAIN PROPER TABLE ADDRESS
SPANW TO EX.
R2=010203.
IF 010203>S(12) THEN
BRANCH TO N3
OBTAIN PROPER TABLE ADDRESS
R2=01020304.
IF 01020304>S(13) THEN
GO TO BRANCH(15)
OBTAIN PROPER TABLE ADDRESS.
OBTAIN PROPER DISCRETE VALUE.
CONVERT TO FLOATING POINT.
NORMALIZED FLT. PT. RESULT IN FPRO.
RETURN.
REPLACES THE INSTRUCTION AT BRANCH.
REPLACE INSTRUCTION AT BRANCH.
LOAD PRIMER(ONLY DONE ON FIRST CALL).
GO TO INSTRUCTION FOLLOWING BRANCH.

PAGE 4

F08APR70 6/30/71

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT
00063C				166	LA	DS 1F
000640	40000000			167	ZERO	DC X'40000000'
000644				168	A	DS 2000XL1
000618	00000001			169		END
00061C	00000000			170		=F'1'
000620	00000700			171		=F'0'
				172		=F'2000'

Best Available Copy