The direct way to generate Y is to generate $\{U\}$ and put $Y=y_i$ if

$$p_1+p_2+\ldots+p_{i-1}<U<p_1+p_2+\ldots+p_i.$$

However, this method requires complicated machine programs that take too long.

An alternative way due to Marsaglia [26.53] is simple, fast, and seems to be well suited to high-speed computations. Let p_i for $i=1, 2, \ldots, n$ be expressed by k decimal digits as $p_i=.\delta_{1i}\delta_{2i}\ldots\delta_{ki}$ where the δ's are the decimal digits. (If the domain of the random variable is infinite, it is necessary to truncate the probability distribution at p_n.) Define

$$P_0=0,\ P_r=10^{-r}\sum_{i=1}^{n}\delta_{ri} \text{ for } r=1, 2, \ldots, k, \text{ and}$$

$$\Pi_s=\sum_{r=0}^{s}10^rP_r,\ s=1, 2, \ldots, k.$$

Number the computer memory locations by 0, 1, 2, . . . , Π_k-1. The memory locations are divided into k mutually exclusive sets such that the sth set consists of memory locations Π_{s-1}, $\Pi_{s-1}+1, \ldots, \Pi_s-1$. The information stored in the memory locations of the sth set consists of y_1 in δ_{s1} locations, y_2 in δ_{s2} locations, . . . , y_n in δ_{sn} locations.

Denote the decimal expansion of the uniform deviates generated by the computer by $u=\cdot d_1d_2d_3\ldots$ and finally let $a\{m\}$ be the contents of memory location m. Then if

$$\sum_{i=0}^{s-1}P_i\leq U<\sum_{i=0}^{s}P_i$$

put

$$y=a\left\{d_1d_2\ldots d_s+\Pi_{s-1}-10^s\sum_{i=1}^{s-1}P_i\right\}.$$

This method is perhaps the best all-around method for generating random deviates from a discrete distribution. In order to illustrate this method consider the problem of generating deviates from the binomial distribution with point probabilities

$$p_i=\binom{n}{i}p^i(1-p)^{n-i}$$

for $n=5$ and $p=.20$. The point probabilities to 4 D are

Value of Random Variable	*Point Probabilities*
0	$p_0=0.3277$
1	$p_1=.4096$
2	$p_2=.2048$
3	$p_3=.0512$
4	$p_4=.0064$
5	$p_5=.0003$

and thus $P_0=0$, $P_1=.9$, $P_2=.07$, $P_3=.027$, $P_4=.0030$ from which $\Pi_0=0$, $\Pi_1=9$, $\Pi_2=16$, $\Pi_3=43$, $\Pi_4=73$. The 73 memory locations are divided into 4 mutually exclusive sets such that

Set	*Memory Locations*
1	0, 1, . . . , 8
2	9, 10, . . . , 15
3	16, . . . , 42
4	43, . . . , 72

Among the nine memory locations of set 1, zero is stored $\delta_{10}=3$ times, 1 is stored $\delta_{11}=4$ times, 2 is stored $\delta_{12}=2$ times; the seven locations of set 2 store 0 $\delta_{20}=2$ times and 3 $\delta_{23}=5$ times; etc. A summary of the memory locations is set out below:

	Value of Random Variable					
	0	1	2	3	4	5
Frequency (set 1)	3	4	2	0	0	0
Frequency (set 2)	2	0	0	5	0	0
Frequency (set 3)	7	9	4	1	6	0
Frequency (set 4)	7	6	8	2	4	3

Then to generate the random variables if

$0\leq u<.9$	put	$y=a\{d_1\}$
$.9\leq u<.97$		$y=a\{d_1d_2-81\}$
$.97\leq u<.997$		$y=a\{d_1d_2d_3-954\}$
$.997\leq u<1.000$		$y=a\{d_1d_2d_3d_4-9927\}$

3. Generating a Continuous Random Variable

The method for generating deviates from a discrete distribution can be adapted to random variables having a continuous distribution. Let $F(y)$ be the cumulative distribution function and assume that the domain of the random variable is (a,b) where the interval is finite. (If the domain is infinite, it must be truncated at (say) the points a and b.) Divide the interval $(b-a)$ into n sub-intervals of length Δ $(n\Delta=b-a)$ such that the boundary of the ith interval is (y_{i-1}, y_i) where $y_i=a+i\Delta$ for $i=0, 1, \ldots, n$. Now define a discrete distribution having domain

$$\left\{z_i=\frac{y_i+y_{i-1}}{2}\right\}$$

with point probabilities $p_i=F(y_i)-F(y_{i-1})$. Finally, let W be a random variable having a uniform distribution on $\left(-\frac{\Delta}{2}, \frac{\Delta}{2}\right)$. This can be done by setting $W=\Delta\left(U-\frac{1}{2}\right)$. Then random