

### 9.3 SIMULATION

We next illustrate the idea of **simulation**. We consider a physical situation in which an element of chance is present and try to imitate the situation on the computer. Statistical conclusions can be drawn if the experiment is performed many times.

#### Loaded Die Problem

In simulation problems we must often produce random variables with a prescribed distribution. Suppose, for example, that we want to simulate the throw of a loaded die and that the probabilities of various outcomes have been determined as shown:

Outcome	1	2	3	4	5	6
Probability	.2	.14	.22	.16	.17	.11

If the random variable  $x$  is uniformly distributed in the interval  $(0,1)$ , then by breaking this interval into six subintervals of lengths given by the table, we can simulate the throw of this loaded die. For example, we agree that if  $x$  is in  $(0,0.2)$ , the die shows 1; if  $x$  is in  $[0.2,0.34)$ , the die shows 2, and so on. A program to count the outcome of 1000 throws of this die might be written as:

```
DIMENSION M(6),Y(6)
DATA L,N/256,1000/
DATA (Y(I),I=1,6)/0.2,0.34,0.56,0.72,0.89,1.0/
DO 2 I=1,6
  M(I)=0
2 CONTINUE
DO 5 I=1,N
  RANUM=RANDOM(L)
  DO 3 K=1,6
    IF(RANUM .LT. Y(K)) GO TO 4
3 CONTINUE
4 M(K)=M(K)+1
5 CONTINUE
PRINT 6,(M(I),I=1,6)
6 FORMAT(5X,6I5)
STOP
END
```

#### Computer Output

191 109 214 179 195 112

### Birthday Problem

An interesting problem that can be solved using simulation is the famous **birthday problem**. Suppose that in a room of  $N$  persons each of the 365 days of the year is equally likely to be someone's birthday. From probability theory, it can be shown that contrary to intuition, only 23 persons need be present for the chances to be better than 50-50 that at least 2 of them will have the same birthday! (It is always fun to try this experiment at a large party or in class to see it work in practice.)

Many people are curious about the theoretical reasoning behind this result, so we shall briefly discuss it before solving the simulation problem. After someone is asked his or her birthday, the chances that the next person asked will not have the same birthday are  $364/365$ . The chances that the third person's birthday will not match those of the first two people are  $363/365$ . The chances of two successive independent events occurring is the product of the probability of the separate events. (The sequential nature of the explanation does not imply that the events are dependent.) In general, the probability that the  $N$ th person asked will have a birthday different from that of anyone already asked is

$$\left(\frac{365}{365}\right)\left(\frac{364}{365}\right)\left(\frac{363}{365}\right)\cdots\left(\frac{365-(N-1)}{365}\right)$$

The probability that the  $N$ th person asked will provide a match is 1 minus this value. A table of the quantity  $1 - (365)(364)\cdots[365 - (N - 1)]/365^N$  shows that with 23 persons the chances are 50.7%; with 55 or more persons, the chances are 98.6% or almost theoretically certain that at least 2 out of 55 people will have the same birthday (see Table 9.2).

Without using probability theory, we can write a subroutine that uses the random-number generator to compute the approximate chances for groups of  $N$  persons. Clearly, all that is needed is to select  $N$  random integers from the set  $\{1, 2, 3, \dots, 365\}$  and to examine them in some way to determine if there is a match. By repeating this experiment a large number of times, we can compute the probability of at least one match in any gathering of  $N$  persons.

One way of writing a subroutine for simulating the birthday problem follows. In it we use the approach of checking off days in a calendar to find out if there is a match. Of course, there are numerous other ways of approaching this problem.

Function PROB calculates the probability of repeated birthdays for a seed  $L$ .

```
REAL FUNCTION PROB(N,L)
LOGICAL BRTHDY,COINCD
DATA M/1000/
SUM=0.0
DO 2 I=1,M
    COINCD=BRTHDY(N,L)
    IF(COINCD) SUM=SUM+1.0
2 CONTINUE
PROB=SUM/REAL(M)
RETURN
END
```

Logical function BRTHDY generates N random numbers and compares them. It returns a value of .TRUE. if these numbers contain at least one repetition and .FALSE. if all N numbers are different.

```

LOGICAL FUNCTION BRTHDY(N,L)
LOGICAL DAYS(365)
DO 2 I=1,365
  DAYS(I)=.FALSE.
2 CONTINUE
DO 3 I=1,N
  NUMBER=365.0*RANDOM(L)+1.0
  IF(DAYS(NUMBER)) THEN
    BRTHDY=.TRUE.
    RETURN
  END IF
  DAYS(NUMBER)=.TRUE.
3 CONTINUE
BRTHDY=.FALSE.
RETURN
END

```

The results of the theoretical calculations and the simulation are given in Table 9.2.

**TABLE 9.2**  
BIRTHDAY PROBLEM RESULTS

<i>N</i>	Theoretical	Simulation
5	.027	.028
10	.117	.110
15	.253	.255
20	.411	.412
22	.476	.462
23	.507	.520
25	.569	.553
30	.706	.692
35	.814	.819
40	.891	.885
45	.941	.936
50	.970	.977
55	.986	.987

### Two Dice Problem

Our next example again has an analytic solution. This is advantageous for us, since we wish to compare the results of Monte Carlo simulations with theoretical solutions. Consider the experiment of tossing two dice. For an (unloaded) die, the numbers 1, 2, 3, 4, 5, and 6 are equally likely to occur. We ask, "What is the probability of throwing a 12 (i.e., 6 appearing on each die) in 24 throws of the dice?"

There are six possible outcomes from each die for a total of 36 possible combinations. Only one of these combinations is a double 6, so 35 out of the 36 combinations are not correct. With 24 throws, we have  $(\frac{35}{36})^{24}$  as the probability of a wrong outcome. Hence,  $1 - (\frac{35}{36})^{24} = 0.49140$  is the answer. Not all problems of this type can be analyzed like this, so we model the situation using a random-number generator.

If we simulate this process, a single experiment consists of throwing the dice 24 times, and this experiment must be repeated a large number of times, say 1000. For the outcome of the throw of a single die, we need random integers uniformly distributed in the set {1,2,3,4,5,6}. If  $x$  is a random variable in (0,1), then  $6x + 1$  is a random variable in (1,7) and the integer part is a random integer in

{1,2,3,4,5,6}. Here is a program:

```
DATA L,M,N/256,0,1000/
DO 4 I=1,N
  DO 2 K=1,24
    I1=INT(6.0*RANDOM(L)+1.0)
    I2=INT(6.0*RANDOM(L)+1.0)
    IF(I1+I2 .LT. 12) GO TO 2
    M=M+1
  GO TO 3
2 CONTINUE
3 IF(MOD(I,100) .EQ. 0) THEN
  PROB=REAL(M)/REAL(I)
  PRINT 5,I,PROB
  END IF
4 CONTINUE
5 FORMAT(5X,I5,F10.5)
STOP
END
```

This program computes the probability of throwing a 12 in 24 throws of the dice at approximately "even money," that is, .505.