

**Title:**  Random Number Generation Using Linear Congruential Method

**Objective:**  Write a Program in C++/java for generating random numbers using Linear Congruential method such that

1. Period of the numbers generated is <=100
2. Density of the numbers generated is < 0.1

**Expected Outcome of Experiment:**

| **Index** | **Outcome** |
| --- | --- |
| CO3 | Generate pseudorandom numbers and perform statistical tests to measure the quality of a pseudorandom number generator. |

**Books/ Journals/ Websites referred:**

1. Jerry Banks, John Carson, Barry Nelson, and David M. Nichol, “Discrete Event System

Simulation”; Fifth Edition, Prentice-Hall.

2. Averill M Law, “System Modelling &amp; Analysis”; 4th Edition TMH.

3. Banks C M, Sokolowski J A, “Principles of Modelling and Simulation”, Wiley

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Pre Lab/ Prior Concepts:**

Random Number generation Techniques

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Theory:**

**Random Numbers:** Random numbers are necessary basic ingredient in simulation of almost all discrete systems. Most computer languages have a subroutine, object or function that will generate a random number. A simulation language generates random numbers that are used to generate event times and other random numbers.

**Properties:** A sequence of random number R1, R2 … must have two important statistical properties, Uniformity and Independence.

Some of the properties are

i) If the interval (0, 1) is divided into ‘n’ classes or subinterval of equal length, the expected number of observations in each interval is N/n where N is total number of observations.

ii) The probability of observing a value in a particular interval is independent of the previous drawn value.

**Problems faced in generating random numbers:**

1) The generated number may not be uniformly distributed.

2) The number may be discrete valued instead of continuous values.

3) The mean of the numbers may be too high or low

4) The variance of the number may be too high or low.

5) The numbers may not be independent e.g.

a. Autocorrelation between numbers

b. Numbers successively higher to lower than adjacent number.

**Criteria for random no. generator**:

1) The routine should be fast.

2) The routine should be portable.

3) The routine should have a sufficient long cycle. The cycle length or period represents the length of random number sequence before previous numbers begin to repeat themselves in an earlier order. A special case of cycling is degenerating. A routine degenerates when some number appear repeatedly which is unacceptable.

4) The random number should be replicable.

5) Most important, the generated random numbers should closely approximate to the ideal statistical properties of uniformity and independence.

**Linear Congruential Method:**

The Linear Congruential method produces a sequence of integers X1,X2,…

Between 0 and m-1 according to the following recursive relationship.

X i+1 (a x i + c) mod m I = 0, 1, 2…

The initial value X0 is called the seed, a is constant multiplier, c is the

Increment and m is the modulus. Maximal period can be achieved by a, c, m, X0 satisfying one of the following conditions

1) For m, a power of 2 (m = 2b) and c≠0 period p = 2b is achieved

provided c is relatively prime to m and a = 1+4k

2) For m = 2b and c = 0 , period p = 2b-2 is achieved provided X0 is odd and multiplier a = 3+8k or a = 5+8k , k = 0,1,2,…

3) For m a prime number and c = 0, period p = m-1 is achieved provided a has the property that the smallest integer is such that a k-1 is divisible by m is k = m-1.

**Algorithm:**

1. take input of a,c,m
2. take input of x0
3. apply the formula X (i+1)= (a\*(x(i)) + c) mod m
4. print value of new x
5. continue till repetition not found
6. if found break and print the index

**Code:**

a = int(input("Enter a = "));c = int(input("Enter c = "));m = int(input("Enter m = "));x = int(input("Enter the seed = "))

L = [x]

xN = (a\*x+c)%m

i = 0

while True:

xN = (a\*x+c)%m

if xN not in L:

L.append(xN)

x = xN

i+=1

else:

print("Random no repeated after",len(L)-1)

break

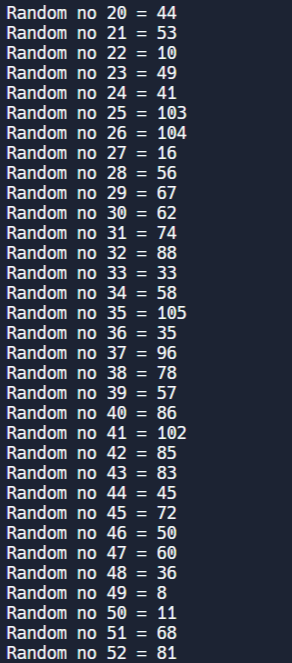
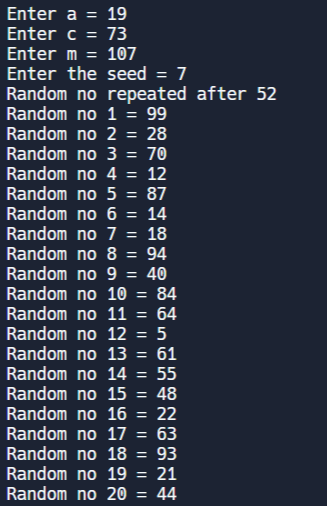
j = 0

for l in L[1:]:

j+=1

print("Random no",j,"=",l)

**Output:**

****

**Conclusion:**

We have understood and implemented the task of generating random numbers and understanding how they can be generated with the help of a formula.

**Post lab questions:**

Consider the multiplicative congruential generator under the following circumstances:

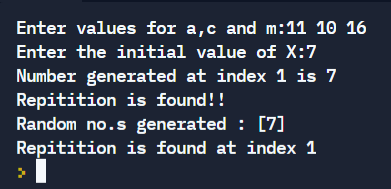
a) X0 = 7, a = 11, m = 16

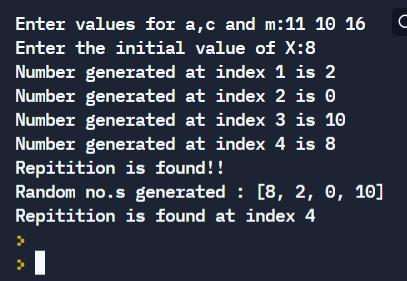
b) X0 = 8, a = 11, m = 16

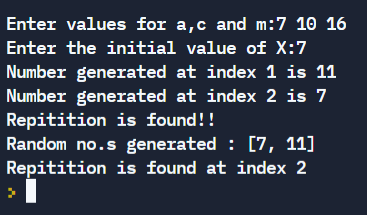
c) X0 = 7, a = 7, m = 16

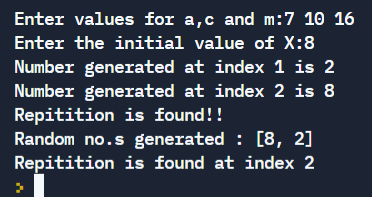
d) X0 = 8, a = 7, m = 16

Generate enough values in each case to complete a cycle. What inferences can be drawn? Is maximum period achieved?









The maximum period is 4