Practical – 6

Aim: Implement a menu driven program to generate random numbers using: a) Triangular distribution b) Uniform distribution.

Program:

```
#include <random>
#include <iostream>
#include <iomanip>
#include <array>
#include <map>
using namespace std;

piecewise_linear_distribution<double> triangular_distribution(double min, double peak,
double max)
{
   array<double, 3> i{min, peak, max};
   array<double, 3> w{0, 1, 0};
   return piecewise_linear_distribution<double>{i.begin(), i.end(), w.begin()};
}

int main() {
   int choice;
   cout<<"Enter \n1. For Triangular distribution and \n2. For uniform distribution\n";
   cin>>choice;
   if(choice&1){
      random_device rd;
      mt19937 gen(rd());
      auto dist = triangular_distribution(0, 7, 10);

      map<int, int> hist;
      for (int i = 0; i < 4000; ++i) {
         double num = dist(gen);
         ++hist[num];
      }
      cout<<"Following are the random numbers generated : ";
      for(auto p : hist) {
         cout << p.second/10<< " ";
      }
      cout<<"\n enter 1 for the graph : \n";
      int x;
      cin>>x;
      if(x&1){
         for(auto p : hist) {
            cout << setw(2) << setfill('0') << p.first << ' '
               << string(p.second/10,'*') << '\n';
         }
      }
```

```
    }
    else{
        const int nrolls=500;
        const int nstars=95;
        const int nintervals=10;

        default_random_engine generator;
        uniform_real_distribution<double> distribution(0.0,1.0);

        int p[nintervals]={};
        cout<<"Random number generated : ";
        for (int i=0; i<nrolls; ++i) {
          double number = distribution(generator);
          ++p[int(nintervals*number)];
          cout<<number<<" ";
        }
        cout<<endl;
        cout << "uniform_real_distribution (0.0,1.0):" << endl;
        cout << fixed; cout.precision(1);

        for (int i=0; i<nintervals; ++i) {
          cout << float(i)/nintervals << "-" << float(i+1)/nintervals << ": ";
          cout << string(p[i]*nstars/nrolls,'*') << endl;
        }

    }
}
```

Output:

```
"C:\Users\Ankit Goyal\OneDrive\Documents\labs\8th Sem Lab\SSM\triangular and uniform distribution.exe"
Enter
1. For Triangular distribution and
2. For uniform distribution
1
Following are the random numbers generated : 6 17 27 40 51 61 72 65 44 12
 enter 1 for the graph :
1
00 ******
01 ****************
02 ***************************
03 ****************************************
04 ******************************************************
05 ************************************************************
06 ********************************************************************************
07 ********************************************************************
08 ****************************************
09 ************

Process returned 0 (0x0)   execution time : 2.928 s
Press any key to continue.
```