

```
1 #include <random>
2 #include <array>
3 #include <cmath>
4 #include <fstream>
5
6 double state = 5.0;
7
8 double random(){
9     state = fmod(state * 100019 + 10067, 101) * 1.0 / 101.0;
10    return state;
11 }
12
13 //The sample size for plotting final distribution - this many numbers will be drawn
14 constexpr size_t samplesize = 100;
15
16 typedef std::array<double, samplesize> Smaples;
17
18 double characteristic_function(Smaples samples, double lambda){
19     double sum = 0;
20     for(auto& sample : samples){ //Loop through the array
21         sum += exp(- sample * lambda);
22     }
23     sum /= samplesize;
24     return sum;
25 }
26
27 int main()
28 {
29     Smaples Z{}; //array to store the values, in case we need
30
31     std::ofstream outfile; //file handle to save the results in a file
32     outfile.open("./output/samples.txt", std::ios::out | std::ios::trunc);
33
34     for(auto& Zi : Z){ //Loop through the array to store the values
35         Zi = random(); // calculate Y1 and store in the array
36         outfile << Zi << std::endl; //write to the output file
37     }
38
39     outfile.close(); //when done, close the file.
40
41     outfile.open("./output/computed characteristic function.txt",
42                 std::ios::out | std::ios::trunc);
43
44     double a = 0, b = 5, N = 10000;
45
46     for (size_t i = 0; i < N; i++) {
47         outfile << a + (b - a) / N * i << "\t" << characteristic_function
48             (Z, a + (b - a) / N * i) << std::endl; //write to the output file
```

```
47     }  
48  
49     outfile.close();    //when done, close the file.  
50 }
```