```cpp
1  #include <array>
2  #include <random>
3  #include <fstream>
4
5  //The sample size for plotting final distribution - this many numbers will
     be drawn
6  constexpr size_t samplesize = 100000;
7
8  double f(double x){
9      if (x < 0){
10         return 0;
11     }else{
12         return exp( - x );
13     }
14 }
15
16 int main() {
17     std::random_device dev; //Responsible for getting a random seed from OS
18     std::mt19937_64 randomwalk(350);    //Mersenne Twister engine with the
         seed for generating pseudo-random numbers
19     std::mt19937_64 selector(350);     //Mersenne Twister engine with the
         seed for generating pseudo-random numbers
20     std::uniform_real_distribution<double> randomwalkdist(-1, 1); //
         distribution in range [-1, 1]
21     std::uniform_real_distribution<double> selectordist(0, 1); //
         distribution in range [0, 1]
22
23     std::array<double, samplesize> positions = {};
24     double position = 0;
25
26     std::ofstream outfile;  //file handle to save the results in a file
27     outfile.open("./output/metropolis exp.txt", std::ios::out |
         std::ios::trunc);
28
29     for (auto& x : positions) {   //loop over number of samples to be drawn
30         double proposed_position = position + randomwalkdist(randomwalk);
31
32         if(selectordist(selector) <= std::min(1.0, f(proposed_position) / f
           (position))){
33             position = proposed_position;
34         }
35
36         x = position;
37
38         outfile << x << std::endl; //write to the output file
39     }
40
41     outfile.close();    //when done, close the file.
42 }
```