

STL: random_shuffle

Tuesday, 14 November 2006

C++ STL (Standard Template Library) is an ANSI/ISO standard that has become an integral part of C++ distributions. STL can be grouped into 6 components: containers, generic algorithms, iterators, function objects, adaptors, and allocators.

Random_shuffle is one of the methods in generic algorithms component. To use this method, we must #include <algorithm> in the program. The method randomly put elements in a vector from [first, last). The method follows uniform distribution, i.e. the probability of each N element is 1/N [1]. The complexity of random_shuffle method is O(N) or linear.

An example below shows how random_shuffle is used to randomly generate and assign data in a vector of size 10.

```
#include <iostream.h>
#include <algorithm>
#include <vector>
#define TOTAL 10

using namespace std;

int main(char **argv, int argc){
    vector<int> number(TOTAL);
    for(int i=0; i<TOTAL; i++)
        number[i] = i;
    for(int i=0; i<TOTAL; i++){
        random_shuffle(number.begin(), number.end());
        copy(number.begin(), number.end(), ostream_iterator<int> (cout, " "));
        cout<<endl;
    }
    return EXIT_SUCCESS;
}
```

Output:

```
1 7 9 3 5 6 0 4 8 2
4 2 1 9 3 6 0 8 7 5
9 4 2 1 6 5 0 8 7 3
0 4 1 9 5 8 3 2 6 7
5 6 2 4 8 3 1 9 0 7
9 1 5 2 3 8 7 4 0 6
2 0 5 3 8 7 4 1 9 6
1 5 8 6 9 2 4 0 3 7
2 1 4 8 3 6 5 0 9 7
6 8 4 3 0 7 9 5 1 2
```

[1] Musser, D.R. Derge, G.J, and Saini, A., STL Tutorial and Reference Guide: C++ Programming with the Standard Template, 2nd edition, Addison Wesley, New Jersey, March 2001.