public member function

std::**uniform_int_distribution::operator()**

<random>

```
(1) template<class URNG>
    result_type operator()(URNG& g);
(2) template<class URNG>
    result_type operator()(URNG& g, const param_type& parm);
```

Generate random number

Returns a new random number that follows the distribution's parameters associated to the object (version *1*) or those specified by *parm* (version *2*).

The generator object (*g*) supplies uniformly-distributed random integers through its `operator()` member function. The uniform_int_distribution object transforms the values obtained this way so that successive calls to this member function with the same arguments produce values that follow a *uniform distribution* within the appropriate range.

Parameters

`g`
A uniform random number generator object, used as the source of randomness.
`URNG` shall be a *uniform random number generator* type, such as one of the standard generator classes.

`parm`
An object representing the distribution's parameters, obtained by a call to member function param.
`param_type` is a member type.

Return value

A new random number.
`result_type` is a member type, defined as an alias of the first class template parameter (`IntType`).

Example

```
// uniform_int_distribution::operator()
#include <iostream>
#include <chrono>
#include <random>

int main()
{
  // construct a trivial random generator engine from a time-based seed:
  unsigned seed = std::chrono::system_clock::now().time_since_epoch().count();
  std::default_random_engine generator (seed);

  std::uniform_int_distribution<int> distribution(1,10);

  std::cout << "some random numbers between 1 and 10: ";
  for (int i=0; i<10; ++i)
    std::cout << distribution(generator) << " ";

  std::cout << std::endl;

  return 0;
}
```

Possible output:

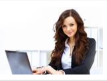
```
some random numbers between 1 and 10: 3 2 1 2 7 10 6 2 4 8
```

Complexity

Amortized constant (a constant number of invocations of `g.operator()`).

See also

uniform_int_distribution::param Distribution parameters (public member function)



Try M-
Files for
Free