Random numbers

a quick overview of random numbers

What the heck does random mean?

From the Wik:

"Randomness means lack of pattern or predictability in events. Randomness suggests a non-order or non-coherence in a sequence of symbols or steps, such that there is no intelligible pattern or combination."

really random events

Physical systems:

- radioactive decay
- thermal noise (activity of electrons when no voltage applied)
- photoelectric effect (release of photons from a photoelectric source)

These are unpredictable

random algorithm and sequence

Random number generators generate random numbers using an ***algorithm***

By definition, an algorithm is predictable (that's what an algorithm is)

So how can a random number generator be really random?

it's the sequence

Remember:

"Randomness means lack of pattern…"

What is **random** in random number generators is the ***sequence***

Your cannot (through statistics or other means) predict the next number in a sequence based on the existing sequence.

10, 21, 78, 51, 58, 29, 14, 71, 71, 95

What comes next (range of 10-100)?

It's random. You cannot predict the next number (if the generator is any good).

the seed

Every random number generator starts with a ***seed***, a starting value

Weirdly, if you start the generator with the same seed you get the same sequence!

The algorithm generates the same sequence starting from the same seed.

Same sequence from same seed

You may say to yourself, "Same sequence from the same seed, how is that random?"

Remember, randomness in this sense is the predictability of the next number given only the sequence.

Also, same sequence from same seed is useful for testing!!!

only need to know 4

There are 16 random engines, 21 distributions. Remember 5 (2 engine, 3 distributions):

- `default_random_engine`
- `mt19937_64`
- `uniform_int_distribution<>`
- `uniform_real_distribution<>`
- `normal_distribution<>`

well, maybe one more

The better, more useful, engine is the Mersenne twister which has a horrible name: `mt19937_64`

The `default_random_engine` does exist but it is the default chosen by the implementors. You don't know which one it is (in fact, this engine had some problems in earlier g++ implementations).

The basics

dre is an instance of a random engine, a source of randomness

it takes a seed arg (not required)

distribution range inclusive

dist is a distribution (it is templated)

distribution always passed an engine as an argument!

```
#include<random>
std::mt19937_64 dre(seed)
std::uniform_int_distribution<long>
                          dist(10,100)
long my_long = dist(dre)
```


random_device

Entropy

Most modern computers (though perhaps not smaller devices such as cell phones) have various physical devices that generate "randomness":

- key stroke timings
- mouse movement
- video refresh
- all kinds of hardware stuff

Based on entropy in your device, if you have it, you can generate a random number:

- not recommended for a lot of random numbers, entropy might be limited
- good as a ***seed***
  - better than the current time, which was often used

random_device

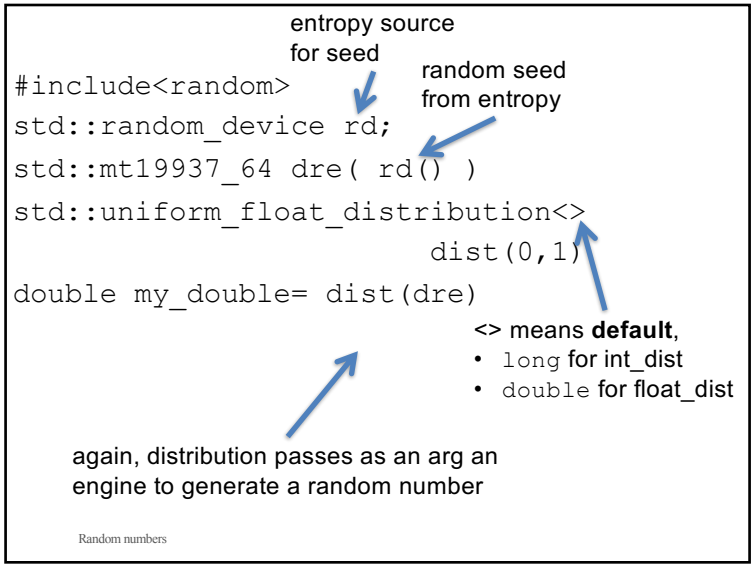
```
#include<random>
using std::random_device;

random_device rd;
cout << rd.entropy() << endl;
cout << rd.min() << endl;
cout << rd.max() << endl;
cout << rd() << endl;
```


rd.entropy()

returns 0 if in fact the random_device is not gathering info from hardware but some other random number generator.

At present, doesn't work on any implementation despite the fact that random_device does indeed work!



Pass engine by reference (never const)

You always pass a random engine by reference, never copy and never by const:

- Reference because if copied it will reset the sequence to the default seed (you get the same seq. every time)
- Never const because generating a number changes the engine each time.