

Random Numbers in Traditional C++

rand()

- rand() is inherited from C and is declared in <cstdlib>
- Calling rand() returns the next number in a sequence of pseudo-random numbers
- It returns an integer between 0 and RAND_MAX (which is usually 32767) inclusive

```
// Print out a pseudo-random integer
```

```
cout << rand() << endl;
```

rand() Examples

- With a little arithmetic, we can get numbers within a different range of values

```
// Print out a pseudo-random floating-point number with value between 0 and 1
cout << 1.0*rand()/RAND_MAX << endl;           // Convert the result to double!
```

```
// Print out 10 pseudo-random integers with values between 1 and 100
for (int i = 0; i < 10; ++i)
    cout << (99*rand()/RAND_MAX + 1) << endl;
```

Seeding a PRNG

- To seed the generator used by `rand()`, we call `srand()`
- Traditionally, programmers use the current time to provide a "random" seed
 - `time()` is a C function, defined in `<ctime>`
`srand(time(0));` *// time(0) returns the current time*
- The value returned by `time()` only changes once per second
- This makes it easier to guess the seed, which makes it easier to guess the random sequence