Module 07: Mathematical Functions

Intro to Computer Science 1 - C++
Professor Scott Frees

Textbook

The following is (loosely) covered in section 4.2 of the text.

Functions

We've seen two examples of calling functions

- `sizeof(datatype)` - returns number of bytes used to store the given data type
- `time(0)` - returns the number of seconds since January 1, 1970

Calling Functions

Functions: `Y = F(x)`

- Y: the value "returned by the function"
- F: the name of the function
- x: argument or "parameter"

We can *pass* multiple parameters to a function - separated by commas

```
Y = F(a, b, c);
```

Math Library

`cmath` library defines many commonly used mathematical functions

- `pow(double, double)`
- `sqrt(double)`
- `sin(double), cos(double), tan(double)`
- `abs(any number)`
- many others

Each of these functions return doubles as well

Include statements

To use cout and cin, we must include iostream

```
#include <iostream>
```

To use time function, we must include ctime

```
#include <ctime>
```

Likewise, we must now include cmath

```
#include <cmath>
```

Note - on Visual Studio you can get away with *not* including cmath - but this is very bad practice!

Programming Example 06

The time it takes an object to fall depends on its height (lets call this distance "X") and the acceleration of objects (9.8m/sec^2 on earth)

$$T = \sqrt{\frac{2X}{A}}$$

Let a user enter a distance, print out time (in seconds) it will take to hit the ground

Units

We always want to think about how a user works with our program:

Bad example:

“Enter height: “

Better example:

“Please enter height (in meters): “

You should always include units in your prompt, and in your output. Things need to be clear!

Summary

Once you understand how to call functions, you can use *any function!*

There is nothing new about using `cos`, `sin`, `sqrt` that you don't already know from learning to use `pow`!