Lecture 1

CS 105: Computer Programming C++
Instructor: Lee Parnell Thompson
Why Learn C++

- Still one of the top used languages in production.
- Can produce incredibly fast code
  - Closer to the machine hardware than many languages
- Many other languages are modeled off of C++
  - C#, Java, Javascript, etc…
- A LOT of legacy code is written in C++
- Still actively being developed.
  - c++11 (2011)
  - c++17 (2017)
Why Learn C++

From Redmonk.com. 2015
Course Overview

- Introduction to C++
- PreReqs: A grade of at least C- in CS 307, CS 313E, CS 314, CS 314H, EE 422C, or EE 322C
- Website: [https://www.cs.utexas.edu/~parnell/teaching/2015/fall/cs105/](https://www.cs.utexas.edu/~parnell/teaching/2015/fall/cs105/)
- Piazza Discussion: [https://piazza.com/utexas/fall2015/cs105](https://piazza.com/utexas/fall2015/cs105)
Programming Assignments

- 7 Programming Assignments (70% of grade)
  - No Pairs, Individual code only
  - Needs to compile on Lab Machines. (double check this!)
  - Needs to run with the example input (at the very least)
  - 2 Assignments can be up to 3 days late (10% off each day)
Intro to C++

- Created by Bjarne Stroustrup as an object oriented extension to C. (1983)
- Covers
  - Basic Syntax, Types, Control Flow
  - Arrays, References, pointers, classes
  - Friends, Operator Overloading, inheritance
  - Polymorphism, Templates
  - And more!
Today

- Review control structure
- Preprocessor directives
  - `#include <...>`
- Basic Functions
- Basic I/O
- How to Compile and Run
I/O Basics

- At the beginning need to include the necessary libraries
  - `#include <iostream> // cout, cin, endl`
  - If you want to avoid prefixing the namespace std
    - `using namespace std;`

- Input
  - `int bar; // declare variable`
  - `cin >> bar; // read in int from standard input`

- Output
  - `cout << bar; // output value of bar`
  - `cout << bar << endl; // output value of bar with end line`
Functions

- `<return value> <function name>( <parameters>)`  
  `{ <function body>}`

Example
- `int sum(int i, int j){ return i + j; }`

In C++ functions need to be declared before they are used:
- `int sum(int, int); /// declare that there is a function named sum`
- `int sum(int i, int j){return i + j;} /// provide implementation of sum`
C++ Compilation Basics

- 3 Basic steps
  - Preprocessor
    - expands #include and other things starting with #
  - Compiler
    - Creates object code (machine-language)
  - Linker
    - Links program with other libraries and creates executable
      - Program.o + otherLibrary.o -> executable.out
C++ Compilation Basics (cont)

- **Steps**
  - Create a c++ file with the suffix .cpp (example: myprog.cpp)
  - Compile the file
    - `g++ myprog.cpp` (will create an executable called a.out)
    - `g++ -o nice.exe myprog.cpp` (will create an executable called nice.exe)
  - Run the executable
    - `./cool.exe`

- **Some Flags**
  - `-o <executable name>` : choose exe name
  - `-Wall` : Warn all (show all warnings)