COMP6771 Week 1
Course Outline

What is C++

C++ is a more complex language than C

C++ code is much simpler than C code

C++ is a powerful, low level language, that allows you to write code that is fast, concise, and easy to understand

There is nothing between C++ and the hardware

    Unlike Java, Python, Javascript - which contain an abstraction layer

C++ has zero over-head abstractions - abstractions are a big part

    STL library has abstract containers, algorithms, and iterators

C++ has a principle of don't pay for what you don't use

    You can use OOP, but you're not obliged to pay the extra runtime cost for polymorphism

    Things automatically get freed up, but you don't pay for garbage collection

In this course we will focus on C++17

What is C++ used for

Operating systems

    Despite myths, C++ is as fast as C, but far more concise and easier to write correct code

- Low latency software (eg. high frequency trading)
  - C++ can have direct control over the hardware
- Games
- Just about anything you can think of

C++ **IS NOT** C

- C++ is backwards compatible with C, so it's easy to think that you can build your C++ understanding directly on top of your C understanding
- However, while valid C code is often valid C++, good C is is almost never good C++ code
  - Over the years C++ continues to diverge from C
- For example, when we teach you best practice, we will not be using:
  - malloc
  - free
  - C-style arrays
  - C-style strings
- And will be sometimes discouraging use of:
  - raw pointers (char *, int *)

What will be taught in this course

- You will be taught how to understand what C++ can do, how C++ works, and modern best practices around writing code. We will cover:
    - STL containers, iterators, algorithms
    - Object Oriented Programming
    - Operator Overloading
    - Iterators
    - Templates

What will be taught in this course

- In 2019 we will also emphasise preparing students for writing C++ in industry.
- We will be teaching:
  - A test framework this year (Catch2)
  - A build system (bazel)
  - Developer tooling (how to use a debugger)

Staff

Lecturer in charge:

Hayden Smith

Lecturers:

Hayden Smith
Matthew Stark
(Guest) Christopher Di Bella

Tutors:

Adam Tizzone, Gaganjot Singh, Gary Bai, Ivor Metcalf, Hayden Smith, Matthew Phillips, Nathaniel Shead, Nicholas Malecki, Nicholas Mulianto, Simon Haddad

Where to get help

Your question-answer hierarchy

1. Webcms3 forum (on every page)
2. Your tutor (Emails on your tutorial page)
3. Lecturer(s) cs6771@cse.unsw.edu.au
4. Hayden hsmith@cse.unsw.edu.au (for personal)

Questions that are non-personal will only be answered on the forum.

Where to get resources

- Books
  - *Programming: Principles and Practice Using C++* (Bjarne Stroustrup, 2nd Edition - May 25, 2014) (**updated for C++11/C++14**) An introduction to programming using C++ by the creator of the language. A good read, that assumes no previous programming experience, but is not only for beginners.
  - Here is a list, but use with caution: https://stackoverflow.com/questions/388242/the-definitive-c-book-guide-and-list
- Websites:
  - We recommend: cppreference.com
  - Do not use: cplusplus.com
    - It frequently is first in search results.
    - Frequently inaccurate and outdated

Course Repository

- The course repository is at https://github.com/cs6771/comp6771
    - Contains the lecture and tutorial code
    - Will get updated as the course progresses

Schedule & Structure

- Please see course outline for full course schedule
- Teaching provided includes:
  - 3 hours of physically delivered lectures (recorded)
  - 1 optional hour of online delivered lectures (catch up, guest lecturing)
  - 1 hour of tutorial with your tutor

Assessment

Assessment	Weighting	Date of completion
Assignment 1	15%	Week 3
Assignment 2	15%	Week 6
Assignment 3	20%	Week 9
Exam	50%	Exam Period

Assessment

- Final marks *may* be scaled
- Final exam:
  - Hurdle exam
  - Done on CSE lab machines - prac and theory components
- Assignments:
  - Will have an emphasis on testing
  - Plagiarism will not be tolerated. Immediate 0 for assignment.
  - Version control knowledge is assumed and recommended
  - Late penalties are 2% every hour off the maximum mark after due date

Feedback

In your first tutorial your class will choose
a class rep to represent your class to
simplify feedback processes.

Zoox

Top 10 performing students, pending an interview with Hayden, will be referred to a personal contact at Zoox for employment opportunities.

https://zoox.com/