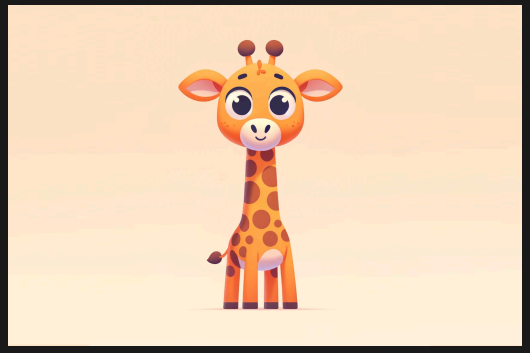
COMP6771

🥑 Welcome & Getting Started

Lecture 1.1

Author(s): Hayden Smith



(Download as PDF)

Why COMP6771?

Why COMP6771?

To learn C++ of course

Why COMP6771?

But why is C++ so interesting?

🤷‍♀️ Why COMP6771?

But why is C++ so interesting?

Some languages are "powerful" (e.g. Java, Python)

🤷‍♀️ Why COMP6771?

But why is C++ so interesting?

Some languages are "powerful" (e.g. Java, Python)

Some languages are "fast" i.e. high performance (e.g. C)

🤷‍♀️ Why COMP6771?

But why is C++ so interesting?

Some languages are "powerful" (e.g. Java, Python)

Some languages are "fast" i.e. high performance (e.g. C)

Some languages have wide adoption and understanding

🤷‍♀️ Why COMP6771?

But why is C++ so interesting?

Some languages are "powerful" (e.g. Java, Python)

Some languages are "fast" i.e. high performance (e.g. C)

Some languages have wide adoption and understanding

Few widely adopted languages are both powerful and fast

🤷‍♀️ Why COMP6771?

How would we describe C++?

Why COMP6771?

How would we describe C++?

By using the C++ design pillars: We want a lightweight-abstraction programming language We want to be more powerful than C, but still simpler than C We still write directly for hardware for performance reasons We want to provide object-oriented capabilities without it being a requirement.

Why COMP6771?

Where is C++ used?

🤷‍♀️ Why COMP6771?

Where is C++ used?

Operating systems

🤷‍♀️ Why COMP6771?

Where is C++ used?

Operating systems

Video Games

🤷‍♀️ Why COMP6771?

Where is C++ used?

Operating systems

Video Games

Fundamental computer applications (e.g. web browsers)

🤷‍♀️ Why COMP6771?

Where is C++ used?

Operating systems

Video Games

Fundamental computer applications (e.g. web browsers)

Robotics

🤷‍♀️ Why COMP6771?

Where is C++ used?

Operating systems

Video Games

Fundamental computer applications (e.g. web browsers)

Robotics

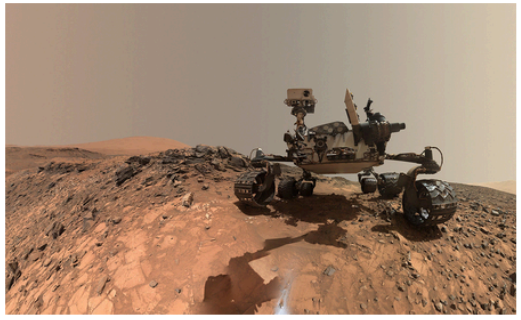
...pretty much anything that is complex and requires high performance

🤷‍♀️ Why COMP6771?

Where is C++ used?

WARGAMING.NET
LET'S BATTLE

Ps Ai



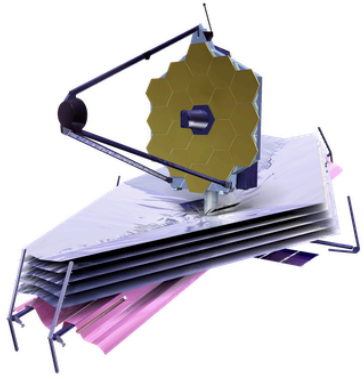
Mars Curiosity Rover, courtesy NASA/JPL-Caltech.

imc

Sandia National Laboratories

Windows 10

What's C++ good for?



James-Webb Telescope, courtesy NASA/JPL-Caltech.

Morgan Stanley

CSIRO

avast

CREATIVE ASSEMBLY™

NVIDIA

🤷‍♀️ Why COMP6771?

Relationship between C and C++

🤷‍♀️ Why COMP6771?

Relationship between C and 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

🤷‍♀️ Why COMP6771?

Relationship between C and 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

🤷‍♀️ Why COMP6771?

Relationship between C and 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

Well-written modern C++ code would be virtually unrecognisable to a C developer. For example, we don't use malloc, free, C-style arrays, C-style structs, C-style linked lists, and generally avoid explicit pointers.

🧠 Assumed Knowledge

- Version control (git)
- Procedural programming (C)
- Object-oriented programming (Java)

If you lack git assumed knowledge, we have lectures to provide you all the help.

📗 Summarised Learning Outcomes

1. skills in writing software using C++20
2. skills in using libraries to develop software
3. skills in using tools to build and test software
4. knowledge and understanding about unit testing
5. knowledge and understanding about reactive programming, object-oriented programming, and generic programming

Approach To Teaching

We are going to focus on the following:

- Learning most key parts of the C++ language
- Use a modern build system and use C++20
- Learn how to test our code
- Focus on good C++ code design

💻 Course Site

Our course site is a custom content management system called Ester. Ester was written by Hayden in April 2024.

🥰 The Team

Lecturer in charge: Hayden Smith

Tutors: 20-30 undergraduate and postgraduate tutors

Guest lectures: two lined up so far!

👀 Assessment Structure

There is no direct assessment associated with Lectures or Tutorials

Item	Due	Weighting
Assignments	Due weeks 3, 4, 7, 10	70%
Exam	Exam Period	30%

 Teaching Methods

🟢 Lectures

- 2 x 2 hours per week

🟡 Tutorials

- You can attend none, any, or all tutorials
- Tutorials are for covering exercise questions
- Please respect that your tutor probably needs to leave at the end
- Tutorials are all in person

 Teaching Methods

🔵 Help Sessions

- Chance to talk to tutors and get help on matters to do with tutorial exercises and assignments
- Help sessions will contain 1-5 tutors who will be split between assisting students with questions, and marking labs off
- Please pay attention to how many tutors are in a given help session - we do our best to predict demand and adjust but please attend help sessions being prepared to wait

🟣 Assignments - In-Depth Skills

- Ass1: Basic algorithmic performance
- Ass2: Defining a simple data type
- Ass3: Defining a complex data type

🔥 Assignments are now due 4 days later this term 🔥

 Teaching Methods

🔴 Exam

- Final exam is:
  - A hurdle / double-pass
  - Closed book
  - In-person

🙏🏾 Getting Help

If you need help with something, go here:

1. EdStem (sidebar on Webcms3)
2. Help Sessions
3. cs6771@cse.unsw.edu.au

🙏🏾 Getting Help

Two primary resources:

- cppreference.com (do not use cplusplus.com)
- Bjarne Stroustrup
  - Programming Principles and Practice using C++
  - A Tour of C++

Getting Setup

If you need help getting "installed" for the course, you can follow our Getting Started Guide

🛠️ Main Tools

The main tools we use in this course are:

- Course Website
- Gitlab
- EdStem (forum)
  - Note Hayden does not directly monitor
- Your own computer! (or vlab)

🎭 Style Guide

In terms of nuanced code style, we provide methods throughout the course to automatically ensure your code complies to a standard style.

In terms of general best practices, you can find more information that in our best practices guide.

🤖 AI Usage Policy

The final exam does not allow for the usage of LLMs.

Besides that, you are welcome to use them as assistants (not code generation tools) throughout the course.

Please remember: AI is killing graduate roles, don't become collateral.

💝 Getting Along

- Understand the expectations around student conduct.
- Create an inclusive learning environment.
- Let's all treat each other with respect and understanding.

First Programs

```
1 #include <iostream>
2
3 int main()
4 {
5     // put "Hello world\n" to the character output
6     std::cout << "Hello, world!\n";
7 }
```

first.cpp

```
1 $ g++ -o hello hello.cpp
2 $ ./hello
```

First Programs

```cpp
#include <iostream>

#include "age.h"

int main()
{
    // put "Hello world\n" to the character output
    std::cout << getAge() << "\n";
}

int getAge()
{
    return 5;
}
```

age.cpp

```cpp
int getAge();
```

age.h

```
$ g++ -o age age.cpp
$ ./age
```

First Programs

```
1 #include <iostream>
2
3 #include "age.h"
4
5 int main()
6 {
7     std::cout << getAge() << "\n";
8 }
```

age_main.cpp

```
1 int getAge();
```

age.h

```
1 #include <iostream>
2
3 #include "age.h"
4
5 int getAge()
6 {
7     return 5;
8 }
```

age_lib.cpp

- We can compile and execute this too.
- Declarations in .h files, definitions in .c files

```
1 $ g++ -o age age_main.cpp age_lib.cpp
2 $ ./age
```

👴 The Problem With Classic Compiling

- Imagine having thousands of header and cpp files?
- You have a few options
  - Manually create each library and make sure you link all the dependencies
    - You would have to make sure you linked them all in the right order
  - Create one massive binary and give it all the headers and cpp files
    - Extremely slow
    - Hard to build just parts of the code (eg. To run tests on one file)
  - Makefiles
    - Unwieldy at large scale (hard to read and hard to write)
  - Any better options?

 Managing Larger Projects

- The solution to this chaos is to use build systems.
  - With these systems, you simply have to declare files and relationships between them, and the build system will figure out what to run for you.
  - In COMP6771 we will be using CMake for compilation in conjunction with VScode for editing.

Cpp-Intro

Let's follow the cpp intro setup activity together

📐 Principles Of Testing

- Test API, not implementation
  - Don't make tests brittle
    - If your code changes, your tests should change minimally
  - Make tests simple
    - It should be obvious what went wrong
    - Don't put if statements or loops in your tests
    - Any complex code should be put in a well-named function

🧪 Catch 2

Catch2 is just one particular framework you can use to test with C++. More information on it can be found here.

github.com/catchorg/Catch2/blob/devel/docs/Readme.md#top

🦻 Feedback

Or go to the form here.