Resources / Course Outline

Course Outline

Contents

Course Details

Course Code	COMP6771
Course Title	Advanced C++ Programming
Convenor/Admin	Hayden Smith (/users/z3418003)
Lecturer(s)	Hayden Smith (/users/z3418003)
Contact for the course	cs6771@cse.unsw.edu.au
Classes	See here (https://webcms3.cse.unsw.edu.au/COMP6771/21T2/resources/59590)
Help Sessions	See here (https://webcms3.cse.unsw.edu.au/COMP6771/21T2/resources/59590)
Units of Credit	6
Course Website	http://cse.unsw.edu.au/~cs6771/21T2/ (http://cse.unsw.edu.au/~cs6771/21T2/)
Handbook Entry	http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP6771.html (http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP6771.html)

Course Summary

COMP6771 is an advanced programming course teaching practical aspects of intermediate/advanced C++ programming. The course focuses on teaching the fundamentals of C++, followed by exploring powerful abstractions that C++ enables. This course focuses on using abstractions as well as building abstractions.

COMP6771 is focused on modern, practical programming methods and tools. This course is designed for latter year CSE students with a reasonable degree of programming competencies.

The course is heavily supported by Christopher Di Bella (https://www.cjdb.com.au/) , a UNSW CSE graduate who is a well regarded expert on C++. His knowledge and expertise assists in forming and updating the course.

Our aim for students who complete this course satisfactorily is that they are highly competent in understanding C++ and it's core features, being able to build complex programs, data structures, and algorithms with C++, and being ready to immediately move into the workforce in areas that rely heavily on C++.

COMP6771 can be a challenging course for students due to the volume of work to complete in a 10 week period.

Assumed Knowledge

Before commencing this course, students should:

- Be competent in constructing and designing programs in the language C (from COMP1511/1917 or equivalent)
- Be competent in understanding object-oriented (OO) programming methods (from COMP2511)
- Be competent with the basics of git usage (pull, push, add, commit)

We will spend minimum time covering basics of C and OO such as pointers, pointer arithmetic, classes, objects, and memory.

Student Learning Outcomes

After completing this course, students will:

1. Design, build, and test C++ programs
2. Use abstractions (data structures, algorithms) to solve problems efficiently
3. Build abstractions to solve problems efficiently
4. Distinguish good, modern, widely-used practices from more outdated practices
5. Be confident setting up build and testing environments for C++ programs

This course contributes to the development of the following graduate capabilities:

Graduate Capability	Acquired in
Scholars capable of independent and collaborative enquiry, rigorous in their analysis, critique and reflection, and able to innovate by applying their knowledge and skills to the solution of novel as well as routine problems	Assignment completion and feedback
Entrepreneurial leaders capable of initiating and embracing innovation and change, as well as engaging and enabling others to contribute to change	Using tutorial and lecture knowledge to complete assignments
Professionals capable of ethical, self- directed practice and independent lifelong learning	Individual assignments completions
Global citizens who are culturally adept and capable of respecting diversity and acting in a socially just and responsible way	Participation in lectures and group work on assignment

Teaching Strategies

- **Lectures** : 4 hours of online lectures.
- **Tutorials** : 1 hour of tutorial per week to go through examples of work covered in lectures.
- **Assignments:** 3 major assignments that give you an opportunity to practice the lessons.
- **Exam** : Final exam to continue to test theoretical and practical knowledge.

Teaching Technologies

This course uses 4 key pieces of technology to engage you.

- **(Centre) Webcms3** : This is your spring-board, it's the easy place to go back to get to everything you need.
- **(Forum) EdStem** : *All* questions relating to the course should be posted in our forum, accessible via sidebar.
- **(Class) Zoom /YouTube** : Lectures are on YouTube, but tutorials and help sessions will be on Zoom.
- **(Code) Gitlab** : All tutorial and lecture code is hosted on Gitlab in your own personal repos, as well as assignments.

Teaching Rationale

Being a latter year CSE course, this course focuses on studying a topic (C++) in depth while assuming that students are mature and independent learners. Because of this a lot of time is focused on content delivery (4 hours of lectures + 1 hour of tutorial), with limited laboratory time. In this way, an expectation is placed on students that they are self-learners and can use the Piazza forum.

The course centres the teaching around 3 programming assignments, worth a total of 70% of the course assessment. We use these assignments to thoroughly put to practice what we teach you. This is a course where failure to make a satisfactory attempt at an assignment will make it difficult to complete further assignments (or the exam) to a high standard.

Student Conduct

The **Student Code of Conduct** (Information (https://student.unsw.edu.au/conduct) , Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)) sets out what the University expects from students as members of the UNSW community. As well as the learning, teaching and research environment, the University aims to provide an environment that enables students to achieve their full potential and to provide an experience consistent with the University's values and guiding principles. A condition of enrolment is that students *inform themselves* of the University's rules and policies affecting them, and conduct themselves accordingly.

In particular, students have the responsibility to observe standards of equity and respect in dealing with every member of the University community. This applies to all activities on UNSW premises and all external activities related to study and research. This includes behaviour in person as well as behaviour on social media, for example Facebook groups set up for the purpose of discussing UNSW courses or course work. Behaviour that is considered in breach of the Student Code Policy as discriminatory, sexually inappropriate, bullying, harassing, invading another's privacy or causing any person to fear for their personal safety is serious misconduct and can lead to severe penalties, including suspension or exclusion from UNSW.

If you have any concerns, you may raise them with your lecturer, or approach the School Ethics Officer (mailto:ethics-officer@cse.unsw.edu.au) , Grievance Officer (mailto:grievance-officer@cse.unsw.edu.au) , or one of the student representatives.

Plagiarism is defined as (https://student.unsw.edu.au/plagiarism) using the words or ideas of others and presenting them as your own. UNSW and CSE treat plagiarism as academic misconduct, which means that it carries penalties as severe as being excluded from further study at UNSW. There are several on-line sources to help you understand what plagiarism is and how it is dealt with at UNSW:

- Plagiarism and Academic Integrity (https://student.unsw.edu.au/plagiarism)
- UNSW Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

Make sure that you read and understand these. Ignorance is not accepted as an excuse for plagiarism. In particular, you are also responsible that your assignment files are not accessible by anyone but you by setting the correct permissions in your CSE directory and code repository, if using. Note also that plagiarism includes paying or asking another person to do a piece of work for you and then submitting it as your own work.

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW staff and students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.

If you haven't done so yet, please take the time to read the full text of

- UNSW's policy regarding academic honesty and plagiarism (https://student.unsw.edu.au/plagiarism)

The pages below describe the policies and procedures in more detail:

- Student Code Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)
- Student Misconduct Procedure (https://www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf)
- Plagiarism Policy Statement (https://www.gs.unsw.edu.au/policy/documents/plagiarismpolicy.pdf)
- Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

You should also read the following page which describes your rights and responsibilities in the CSE context:

- Essential Advice for CSE Students (https://www.engineering.unsw.edu.au/computer-science-engineering/about-us/organisational-structure/student-services/policies/essential-advice-for-cse-students)

Assessment

Item	Key Topics	Completion type	Due	Marks
Assignment 1	C++ Basics, STL	**Individual**	Week 3 Friday 8pm	15%
Assignment 2	Operator Overloading, OO, Resource Management	**Individual**	Week 7 Monday 8pm	25%
Assignment 3	Iterators, Exceptions, Templates	**Group (pairs)**	Week 10 Monday 8pm	30%
Final Exam	All topics	**Individual**	Exam Period	30%

Content Schedule

Week	Topics
1	Course intro, Getting started
2	C++ Basics, STL

3	Basic Object-Oriented Programming, Operator Overloading
4	Exceptions
5	Resource Management
6	Flexibility week
7	Templates
8	Iterators
9	Advanced Object-Oriented Programming
10	Extension Topics, Exams

Resources for Students

There is no definitive textbook list for this course. Students are able to complete this course without the use of an explicit resource, however, students are encouraged to seek out resources we recommend if they feel they need the help.

If we had to point you to a single resource, it would be:

- *Programming: Principles and Practice Using C++* (https://rads.stackoverflow.com/amzn/click/com/0321992784) (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.

For a more detailed list of resources, you can explore this Stack Overflow (https://stackoverflow.com/questions/388242/the-definitive-c-book-guide-and-list) article. Some of these resources may be better than others, and we do not endorse a particular one.

Special Consideration

If your work in this course is affected by unforeseen adverse circumstances, you should apply for Special Consideration. If your request is reasonable and your work has clearly been impacted, then

- for an assignment, you may be granted an extension
- for the Final Exam, you may be offered a Supplementary Exam

Note the use of the word "may". None of the above is guaranteed. It depends on you making a convincing case that the circumstances have clearly impacted your ability to work.

UNSW handles special consideration requests centrally (in the Student Lifecycle division), so all special consideration requests must be submitted via the UNSW Special Consideration (https://iaro.online.unsw.edu.au/special_consideration/home.login) website.

Special consideration requests must be accompanied by documentation, which will be verified by Student Lifecycle. Do not email the course convenor directly about special consideration.

Extensions on assignments will only be awarded if the majority of the team are affected.

If you cannot attend the Final Exam because of illness or misadventure, then you must submit a Special Consideration request, with documentation, through MyUNSW within 24 hours of the exam. If your request is reasonable, then you will be awarded a Supplementary Exam (aka "Supp").

Note that UNSW expects you to be available to sit Supplementary Exams if required. If you are awarded a Supp and do not attend, then your exam mark will be zero.

For further details on special consideration, see the UNSW Student website (https://student.unsw.edu.au/special-consideration) .

If you are registered with Disability Services, please forward your documentation to Hayden Smith (mailto:mailto:hsmith@cse.unsw.edu.au) within the first two weeks of semester.

Course Evaluation and Development

This course is evaluated each session using the myExperience system. Your feedback will be used to improve future offerings of the course.

Resource created 2 months ago (Sunday 11 April 2021, 08:15:58 PM), last modified a day ago (Thursday 27 May 2021, 03:34:25 PM).

Comments

🔖 🔍 (/COMP6771/21T2/forums/search?forum_choice=resource/59583) 💬 (/COMP6771/21T2/forums/resource/59583)

💬 Add a comment

There are no comments yet.