

COMP2511

The Art of Software Design (OO Design & Programming)

Course Introduction

Term 3, 2022

Our Team



Ashesh Mahidadia



Nick Patrikeos

Lecturer-in-charge:

Dr Ashesh Mahidadia <ashesh@cse.unsw.edu.au>

Course Admin:

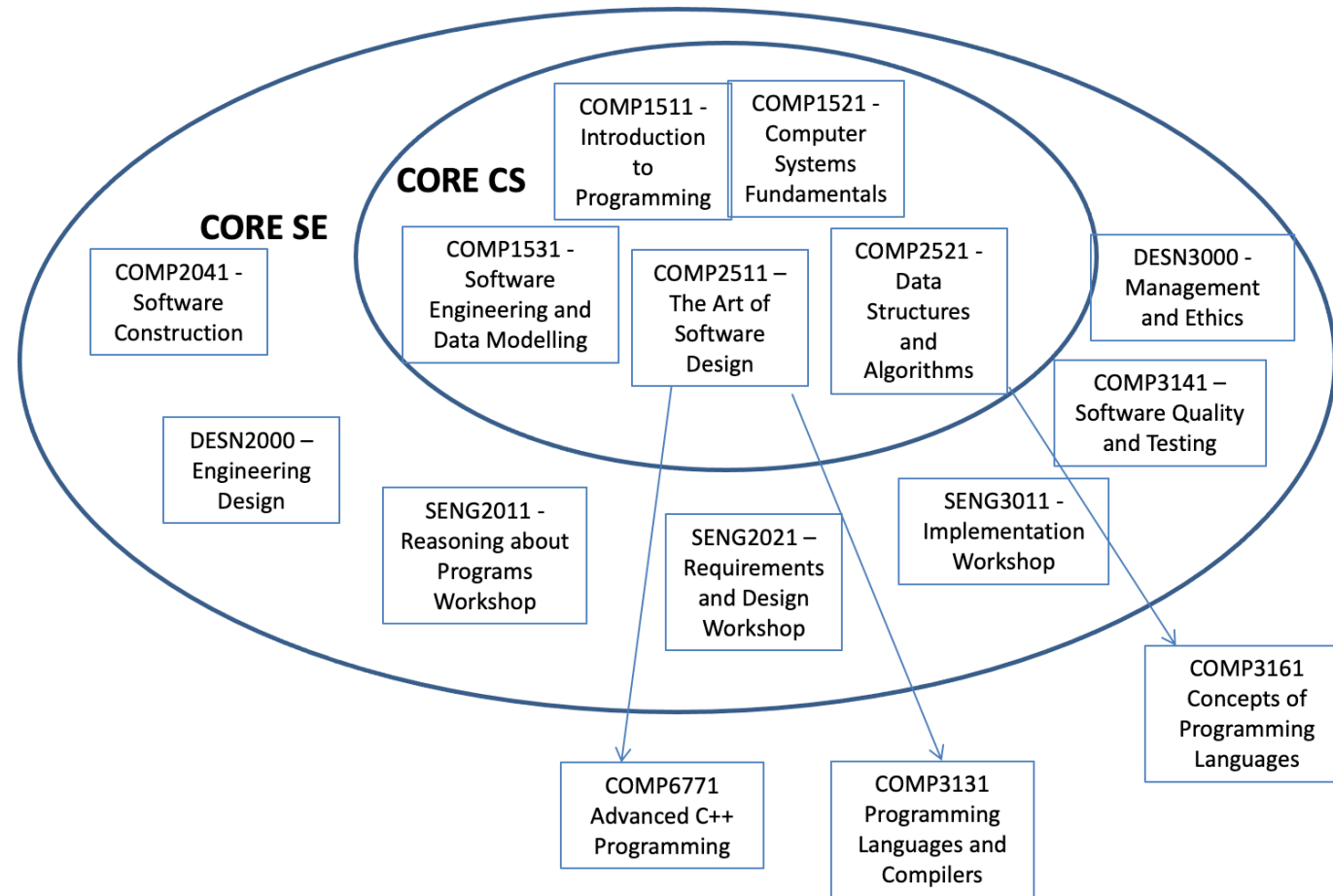
Nick Patrikeos <n.patrikeos@unsw.edu.au>,

Course Account Email: cs2511@cse.unsw.edu.au

*(Unless you specifically require to contact a member of the admin team, please use the **above email** for any queries related to the course.)*

Class Web: <http://webcms3.cse.unsw.edu.au/COMP2511/22T3/>

Course Context



The Story So Far: Course Context

- COMP1511: Solving problems with computers, the wonder and joy of programming
- COMP1521: Getting right down into the silicon
- COMP1531: Solving problems in a team; programming in the large
- COMP2521: Solving problems at scale using data structures and algorithms
- **COMP2511???**

COMP2511

- We can write code, but how do we write good code?
- **Designing elegant and beautiful software.**
- Shades of Grey - things aren't clear cut; writing good software is an art
- From programmers to designers.



COMP 2511 Major Themes

- Develop an appreciation for elegantly written software, and how to create and maintain well-designed systems;
 - Apply principles and patterns to effectively design **flexible, maintainable** and **reusable systems**
- Understand different design paradigms and methodologies, their background and application;
 - Object-Oriented Paradigm
 - Functional Paradigm
 - Asynchronous Paradigm
 - Event-Driven Paradigm

COMP 2511 Major Themes

- Understand and apply the principles of Object-Oriented Design to solve problems;
 - Be able to follow a **systematic** OO Design process
 - Be able to interpret and use tools for OO Design
- Understand the role of and apply widely used Design Patterns to create extensible designs
 - Behavioural patterns
 - Structural patterns
 - Creational patterns
 - Programming patterns (exceptions, generic programming)
 - Testing patterns

COMP 2511 Major Themes

- Develop skills in both creating medium-scale systems from scratch, and working on existing systems as part of the Software Development Life Cycle;
 - Be able to analyse, refactor and work with code started by someone else
 - Create medium-scale systems using Java
- Work with an enterprise programming language and IDE
 - Java language
 - VSCode IDE
- Explore the human factors at play in Software Design and Engineering; including ambiguity, the unknown and trade-offs

Credit teaching material

- ❖ No text book, the lecture slides cover the required topics.
- ❖ However, you are strongly encouraged to read additional material and the reference books.
- ❖ In the lecture notes, some content and ideas are drawn from:
 - *Head First Design Patterns* , by Elisabeth Freeman and Kathy Sierra, The State University of New Jersey
 - *Refactoring: Improving the design of existing code* , by Martin Fowler
 - Material from many popular websites.

How do we obtain our educational objectives?

❖ **Lectures:** 4 hour lectures (9 weeks)

❖ **Tutorials:**

- ❖ A 1 hour tutorial session per week, which is scheduled before the lab.
- ❖ Online Tutorials/Labs will be run via **MS Teams** .
- ❖ Tutorials contribute to your Coursework Mark
- ❖ What if I can't make it?
- ❖ Tutorials are understanding-driven - interactive examples to illustrate concepts discussed in lectures
- ❖ Solutions and recording to tutorials posted at the end of each week

How do we obtain our educational objectives?

❖ Labs:

- ❖ 2 hours each week, straight after tutorial
- ❖ Similar to most CSE core courses
- ❖ Online Run via MS Teams
- ❖ Lab marking: Tutors will mark your completed labs
- ❖ Help/assistance with labs/assignments
- ❖ Oral assessments for Assignment I: Weeks 5, 7
- ❖ Oral assessments for Assignment II: Weeks 8, 9, 10

Assessments

Coursework (15%)

- ❖ Your coursework mark is made up of marks associated with tutorials and lab exercises.
- ❖ Your final coursework mark is out of 56 marks.
- ❖ Each week, you will be assessed on:
 - ❖ Breadth of Completion
 - ❖ Software Design: Practical Application
 - ❖ Software Design: Theoretical Understanding
 - ❖ Engagement (tutorials)
- ❖ Your coursework mark for each week will be out of 8. We will take your **best seven labs** towards your class mark, meaning that you are able to forfeit one week's labs without losing any marks towards the overall Class Mark.
- ❖ Bonus marks can also be achieved for forum participation that allow you to make up for areas lost in the class mark.
- ❖ There are additional choice activities which students may complete if they wish. They are non-assessable.
- ❖ For full information read the Coursework Assessment page on Confluence:
<https://unswcse.atlassian.net/wiki/spaces/cs2511/pages/27328513/Coursework>

Tutorials

To receive 2/2 for weekly tutorial participation, you will need to:

- ❖ Attend the full tutorial
- ❖ Ask good and logical questions throughout the tutorial
- ❖ Take initiative to answer questions and be engaged
- ❖ If a you cannot attend an in-person tutorial due to COVID-19 or other circumstances beyond your control, please email your tutor explaining your situation. Your engagement mark for that week will be the average of your other weeks' participation.
- ❖ In the event of widespread difficulties that prevent students from coming onto campus such as train strikes and floods, the course will provide alternative offerings for in-person tutorials as needed.

Assignment I (20%)

- ❖ Specification released beginning of Week 2.
- ❖ Due Friday Week 4.
- ❖ Completed **individually**
- ❖ Oral assessment (interview) on your work in your Week 5 or Week 7 lab

Assignment II (25 %)

- ❖ The marking criteria for the project will be outlined in the specification which will be released at the beginning of Week 5.
- ❖ **Pairs** formed within your tutorial
- ❖ Pairs formed by end of Week 3.
- ❖ Due **end of Week 8**.
- ❖ Bonus marks for pairs who submit early (end of Week 7), allowing them to continue on to Assignment III
- ❖ Measures in place to ensure a difficult partner doesn't take a toll on marks
- ❖ Oral assessment on your work in your Week 8, 9 or 10 lab

Assignment III (5% Core, 5% Bonus)

- ❖ A more challenging real-world problem that incorporates Design Principles and Pattern discussed in the course
- ❖ For students who wish to extend themselves and score highly in the course
- ❖ Can be completed in a pair or individually
- ❖ Assignment spec released beginning of Week 8
- ❖ Assignment due end of Week 10

Why are we changing the structure this term?

- ❖ Better align the course assessment with the course learning outcomes;
- ❖ Enable a more accurate assessment of students' performance;
- ❖ Decrease the burden on students while maintaining quality education;
- ❖ Enable students to focus on the design of their software in the major assessments;
- ❖ Enable tutors and course staff more opportunities to provide feedback and guidance on design;

Exam (35%)

- ❖ In 22T3 the COMP2511 exam will be held in person in the CSE Labs, and invigilated .
- ❖ International students who are not based in Sydney will be able to complete the exam online. Students who are in this position will need to apply for Special Consideration and show proof of residency at a location outside of Sydney. Students who are based in Sydney are required to take the exam in person, even if they have enrolled in online classes.
- ❖ Students are eligible for a Supplementary Exam if and only if:
 - Students cannot attend the final exam due to illness or misadventure. Students must formally apply for a special consideration, and it must be approved by the respective authority.

Assumed Knowledge

- ❖ Confident programmers
 - Familiar with C and Python programming concepts
- ❖ Able to work in a team
 - Git
 - Working with others
- ❖ Understand basic testing principles
- ❖ Understand basic software engineering design principles (DRY, KISS)

Assumed Knowledge

- ❖ What we don't assume:
 - Knowledge of Java
 - Understanding of Object-Oriented Programming
- ❖ **This is not a Java course**

Course philosophy

- ❖ A step up from first year courses
- ❖ Challenging but achievable
- ❖ Develop skills in time management, teamwork as well as critical thinking
- ❖ Highly rewarding

Support

- ❖ Supporting you is our job :)
- ❖ Help Sessions
 - Lots of them with fantastic tutors
 - Feedback on work, help with problems, clarifying ideas
 - You are expected to have done your own research and debugging before arriving

Support

- ❖ Course Forum (Ed)
 - Ask questions and everyone can see the answers!
 - Make private posts for sharing code
 - Response time
- ❖ Course Account - cs2511@cse.unsw.edu.au
 - Sensitive/personal information
- ❖ During the project - your tutor

Support

- ❖ Go to help sessions for help on concepts
- ❖ Post on the forum if you need more immediate lab feedback
- ❖ There are no late extensions on labs unless in extenuating circumstances - email cs2511@cse.unsw.edu.au

Support - UNSW

- ❖ **Special Consideration** -
<https://student.unsw.edu.au/special-consideration>
- ❖ **Equitable Learning Services** -
<https://student.unsw.edu.au/els>

Mental Health & Wellbeing

- ❖ UNSW Psychology & Wellness - <https://student.unsw.edu.au/mhc>
- ❖ UNSW Student Advisors - <https://student.unsw.edu.au/advisors>
- ❖ Reach out to us at cs2511@cse.unsw.edu.au
- ❖ Check in with each other
- ❖ Talk to someone

Technology Stack

- ❖ Java Version – JDK 11
- ❖ VSCode
- ❖ Gradle 7.2 (Week 3 onwards)
- ❖ Gitlab (+ CI pipelines)

Feedback

- ❖ We love feedback :)
- ❖ Changes made to the course this term based on constructive student feedback
- ❖ We always want to continuously improve
- ❖ Feedback form
- ❖ Course account
- ❖ Student representatives

Respect

❖ Yourself, each other, course staff

Let's have a fantastic T3!!!

