COMP1531

🍓 Other - Course Overview

Lecture 1.1

Author(s): Hayden Smith

(Download as PDF)

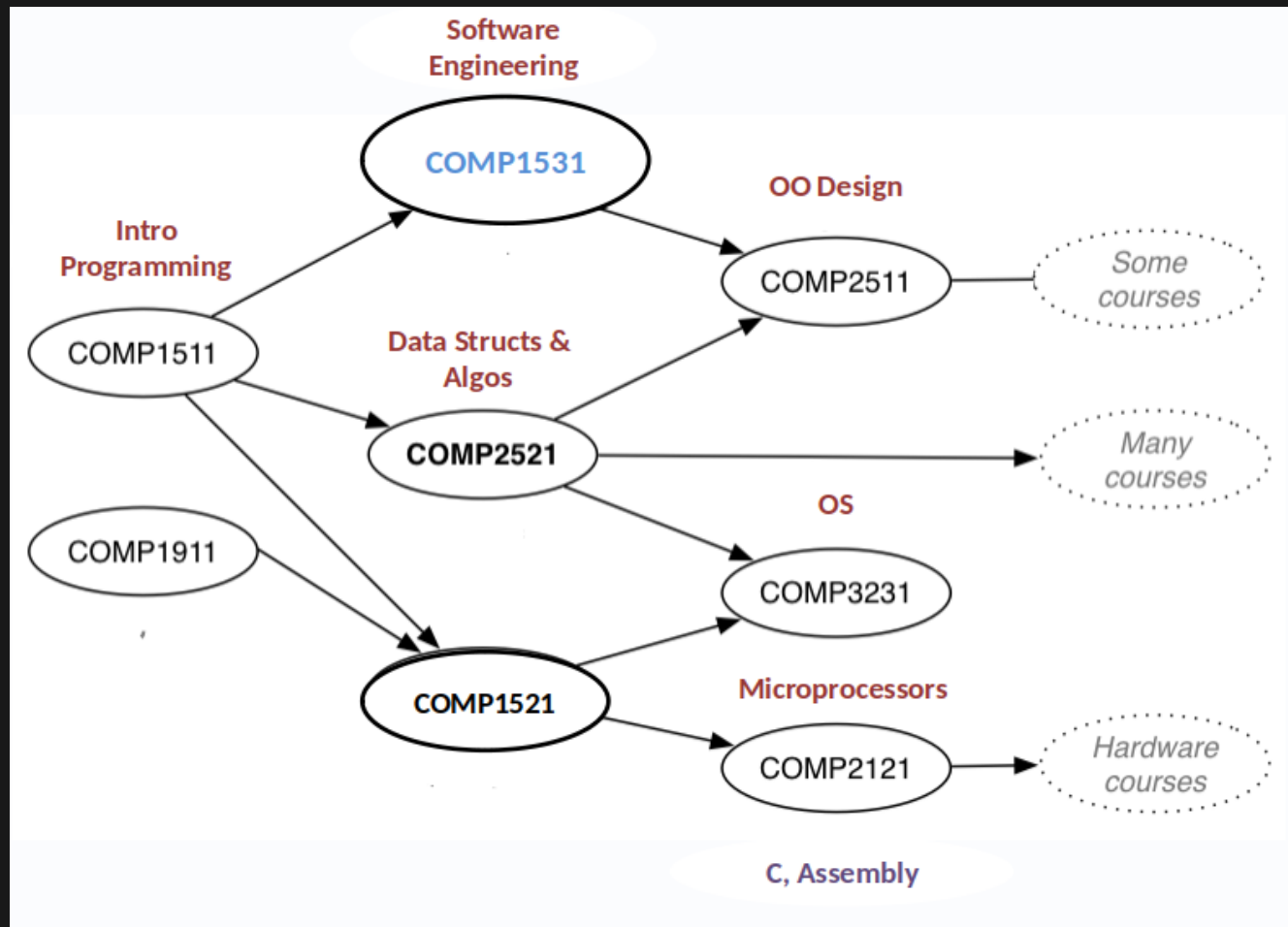# 🤔 Why Are We Here?

Fundamentals of software engineering.

🤔 Why Are We Here?

- **COMP1511**: Learning programming by writing code to solve problems

- **COMP1531**: Learning how to be a software engineer by architecting and building software in a team.

More on this soon...

🤔 Why Are We Here?

How this fits into your program

🤔 Why Are We Here?

"I took 1531 last year. I'm on an internship right now, and I have to say that it was the most useful course at UNSW I've taken so far. My day-to-day work is surprisingly similar to how the course worked, with sprints, testing, and using Git."

\- Previous COMP1531 student

🤔 Why Are We Here?

"I think the major project and the content of the course itself was incredibly insightful and applicable to real world team situations"

"I feel like I'm now able to work effectively in a software team."

"Helps and prepares for professional life in a way."

"Although content was quite heavy first few weeks, after that it slowed down allowing me to properly consolidate the content rather than rush through everything."

🧐 What Is Software Engineering?

What's the difference between **Computer Science** and **Software Engineering**?

🧐 What Is Software Engineering?

What's the difference between **Computer Science** and **Software Engineering**?

Computer Science is concerned with understanding how computers work and function - i.e. making a computer do things

🧐 What Is Software Engineering?

What's the difference between **Computer Science** and **Software Engineering**?

Computer Science is concerned with understanding how computers work and function - i.e. making a computer do things

Software Engineering is where we focus on building the right software for the right people, and making sure it's maintainable over time as it grows and people working on it change.

🧐 What Is Software Engineering?

What's the difference between **Computer Science** and **Software Engineering**?

A genius sitting in their garage alone writing code and Instagram's developers might both be engaging equally in areas of **computer science**, but vastly differ in areas of **software engineering**.

🧐 What Is Software Engineering?

What's the difference between **Computer Science** and **Software Engineering**?

- At UNSW, Software Engineering is an extension of Computer Science, where we give extra focus to how software systems are built, how to manage projects, and how to test software to provide quality assurance.
- Do you need to be a **Software Engineering student** to be employed as a **Software Engineer**?

🧐 What Is Software Engineering?

What's the difference between **Computer Science** and **Software Engineering**?

- At UNSW, Software Engineering is an extension of Computer Science, where we give extra focus to how software systems are built, how to manage projects, and how to test software to provide quality assurance.
- Do you need to be a **Software Engineering student** to be employed as a **Software Engineer**?

<p align="center">No</p>

🧐 What Is Software Engineering?

Software Engineering: *"The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches; that is, the application of engineering to software." (IEEE definition)*

🧐 What Is Software Engineering?

We want all of you to be good software engineers. That means we want you to be able to:

- 🔎 Extract and understand the problem you're trying to solve.
- 🛠️ Design and develop code that resists regressions when it changes over time (problem or people).
- ✅ Ensure that software is as safe as possible by verifying that it's correct.
- 🌏 Understand that the purpose of software is to make it available for use.
- 👬 Work together with a group of people.

🧐 What Is Software Engineering?

Being a software engineer is about realising that even though the code you write is intended for machines to use, it's even more important to consider other humans who read it.

🧐 What Is Software Engineering?

And the reason that we want you to be a good software engineer is simple.

Software engineers are employed by organisations. Organisations have an interest in de-risking themselves in the long term.

Good software engineers apply what they know to help make teams and software last for the long term. And it helps reduce big hiccups.

🐶 We're Going To Have Fun!

Software Engineering

- 🔍 Requirements
- 🎨 Design
- 🛠️ Development
- ✅ Correctness
- 🌍 Projects
- 👬 Coding together
- 📚 Full-stack

🐶 We're Going To Have Fun!

These topics are all about helping us iterate through the software development life cycle whilst building quality software and working well with others.

🔍 Requirements

Requirements focus on understanding a problem we're trying to solve. If we don't do this right, we might build something that works well for the wrong purpose. We learn about requirements engineering, use cases, and validation.

🎨 Design

Good design is about setting up good architecture for our system but also trying to write elegant code. We'll try and model systems and their complexity as well as write graceful and thoughtful code.

🛠️ Development

We need to actually code "the thing", and in this course we're using Javascript. We'll learn about Javascript's language features in multi-file projects.

✅ Correctness

Software needs to be verifiably correct as much as possible. We need confidence that it's doing "the thing" correctly. We'll learn about testing code statically and dynamically, exceptions, code coverage, and linting.

🌍 Projects

Code lives on more than just your computer. Code you use will be written by others, and code you write will need to be given to others. We learn about how to manage multi-package projects, and ensure that we have a continuous and automated way to integrate multiple people's code and deploy that to "users".

👬 Coding Together

Teamwork is both about people and about tools. We'll learn about how to work with version control software between people, as well as how to function as a group of programmers.

📚 Full-Stack

Many software projects are now web-based and have frontends and backends. We'll be working in similar systems using HTTP and data layers to write a functioning server.

🙏 Thank You

For this term every piece of COMP1531 content and code has been either written from scratch or modified from a previous term in a minor or major way.

We've done this to bring you new exciting content and also make the course smaller and easier.

We appreciate your patience throughout term as we make slight adjustments.

👂 Feedback

Or go to the form here.