

# COMP2511

---

## WEEK 1

Object oriented programming!!!

UNSW

# A G E N D A

---

- Introduction
- Solving Design Problems
- Hello Java and Git Revision
- Abstraction

# INTRODUCTION

---

- Jeremy
- 3rd year Computer Science / Science (Mathematics)
- First time teaching COMP2511 🙏
- Did COMP2511 in 23T3!
- Taking COMP3141, COMP3900 and MATH2221
- Enjoys: Cooking and Competitive Programming



COMP2511!

# ADMIN STUFF

---



## Getting Help

1. Forum
2. Help Sessions
3. Email me  
(jeremy.le1@unsw.edu.au)

Email the course if you would like anything escalated to course admins (cs2511@cse.unsw.edu.au)



## Platforms

1. WebCMS3 (Course Info)
2. GitLab (Code)
3. EdStem Forum
4. Moodle / Echo360 (Lectures)



## Assessment Structure

Assignment I : 15%  
Assignment II: 20%  
Assignment III: 8% Bonus  
Coursework (Labs): 15%  
Final Exam: 50%



## Tutorials / Labs

- 1 hour tutorial, 2 hour lab
- Tutorial: Content + Demonstration
  - Labs: Exercises, General Help
  - At least ~50% of the exam

Slides & Code:  
<https://github.com/jeremyle56/tutoring/tree/24T2/24T2/cs2511>

## DESIGN PROBLEM

---

UNSW has decided that they want to create their own light rail, which takes students from upper campus to lower campus. Design a solution for this – how will it work? What will need to be changed about the campus layout for it to work?

# JAVA VS OTHER LANGUAGES

---

## Scopes and Code Blocks

- Similar to C and JavaScript/TypeScript, **Java** uses the curly brace syntax { and }.

## Classes and Object Oriented Program

- **Java**, Python, JavaScript/TypeScript support Object Oriented Programming (OOP)
  - This means they have classes and inheritance.
- All code in **Java** has to exist within a class.

## Types:

- C, TypeScript and Java are statically typed.
- Python, JavaScript are dynamically typed.

# JAVA VS OTHER LANGUAGES

---

## Memory:

- C allows for manual memory via the standard library.
- **Java**, Python, JavaScript, TypeScript have automatic memory management with a garbage collector.


## Compilation:

- C compiles into machine code and separate executable.
- **Java** and Python compile into byte code which is interpreted.
- TypeScript transpiles into JavaScript code which is interpreted.

# VSCODE EXTENSIONS


Ensure that you have the correct extensions installed

Extension: Extension Pack for Java



Extension Pack for Java


v0.26.0

Microsoft  microsoft.com | 27,360,183 | ★★★★★ (75)

Popular extensions for Java development that provides Java IntelliSense, debugg...

Disable

Uninstall




This extension is enabled globally.

DETAILS

FEATURES


CHANGELOG


Extension Pack (6)



Language Support for Java(TM)...


Java Linting, Intellisense, formatting, refac...

 Red Hat



Debugger for Java

A lightweight Java debugger for Visual Stu...

 Microsoft

Categories

Programming Languages

Linters

Debuggers

Formatters

Snippets

Extension Packs


Resources

Marketplace

Extension Pack for Java


EXTENSIONS: MARKETPLACE


java extension pack



Extension Pack for J...

Popular extensions for Java d...


 Microsoft



Java extension pack


The most common extension...


walkme



Spring Boot Ext...

A collection of extensions for ...


 VMware



Node.js-Extensio...

Popular VS Code extensions f...


Wade Anderson



Yet Another Jav...

This package contains my fav...


Guilherme Stella



Extension Pac...

JDK Auto-Configuration + Ext...

Pleiades



A Java Extension Pack

Java扩展包



# HELLO JAVA

---

Make a simple Java program that prints "Hello World" in the HelloWorld class.

```
public class HelloWorld {  
    Run | Debug  
    public static void main(String[] args) {  
        System.out.println("Hello World");  
    }  
}
```

# GIT REVISION

---

- Version Control software
- Allows you to save snapshots of code
- Combined with cloud providers / online storage, i.e. Gitlab / Github / Bitbucket, allows users to work on code together

**git add:** Stage files

**git commit:** Commit the staged files as a snapshot

**git push:** Push your new commits to an online origin

**git status:** State of current repository & branch

**git log:** History of current branch

# S U M

---

Inside a new file called Sum.java, write a program that uses the Scanner class which reads in a line of numbers separated by spaces, and sums them.

# SHOUTER

---

Inside a new file Shouter.java, Write a program that stores a message and has methods for getting the message, updating the message and printing it out in all caps. Write a main() method for testing this class.

# ABSTRACTION

---

## What is abstraction?

Abstraction is the process of generalising concrete details, such as attributes, away from the study of objects and systems to focus attention on details of greater importance.

## What examples of abstraction have we seen in previous courses?

Abstract Data Types (ADTs)

- Linked Lists
- Trees
- Graphs
- Etc...

In C:

- List.h file which provides an “interface” for List.c which is the implementation using the interface.

## How does abstraction allow us to write better software?

Reducing complexity of code, reducing the reliance of code and “connectedness” of the whole project. (Coupling)

This means changing one class (section of code) will only have little effect on other aspects of the codebase.

## Discuss how OOP takes abstraction to another level from what we have seen previously.

Abstraction is used to hide unnecessary information and display only necessary information to the users interacting.

Everything in Java is an object!

All objects provide an abstraction from each other.

# Lab Time!!!

---

YAYYYY

UNSW