SWEN20003
Object Oriented Software Development

Subject Introduction

Shanika Karunasekera
karus@unimelb.edu.au

University of Melbourne

Teaching Staff

Subject Coordinator and Lecturer:

- Shanika Karunasekera (karus@unimelb.edu.au)

Tutors:

- Eleanor McMurtry (Head Tutor)
- Ni Ding (Staff Tutor)
- Rohyl Joshi
- Jenny Yan
- Benedict Ong
- Asil Mian
- Nathan Gellie

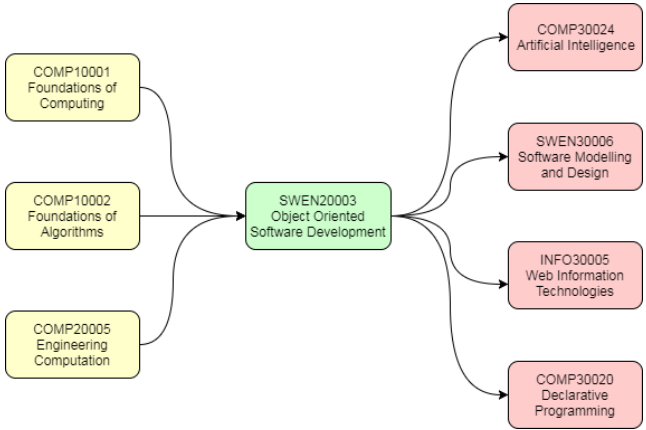
A bit about myself

- Shanika Karunasekera
  - Professor in the Department of Computing and Information Systems
  - Leader of the Software Engineering Discipline
- Education
  - PhD in Electrical Engineering - University of Cambridge, UK
  - B.Sc. (First Class Honours) in Electronic and Telecommunications Engineering - University of Moratuwa, Sri Lanka

A bit about myself

- Industry Experience
  - Distinguished Engineer, Software Architect (1995 - 2002)
    - Lucent Technologies, Bell Labs Innovation (AT&T Bell Labs), USA
- Academic Experience
  - Academic at the University of Melbourne since 2003
  - Teaching
    - Software Engineering and Distributed Computing
  - Research
    - Bigdata analytics
    - Distributed systems
    - Data steam mining

OOSD in Context



Learning Outcomes - Handbook

On completion of this subject you are expected to:

- Apply software design principles to object-oriented design
- Develop object-oriented models for a medium-sized software system
- Evaluate design trade-off of different designs
- Implement an object-oriented design in a suitable language
- Use commonly available object-oriented design frameworks for application development
- Apply knowledge of basic science and engineering fundamentals

Learning Outcomes - In Simple Language

On completion of this subject you will be able to:

- *Write* **Object Oriented** programs in Java
- *Use* an Integrated Development Environment (IDE) to develop software
- *Abstract* a problem specification the Object Oriented way
- *Design* a software solution, not just write it
- *Apply* programming techniques, frameworks, and conventions, to other Object Oriented languages

Lecture Schedule

SWEN20003 Object Oriented Software Development - Semester 2, 2020				
Block	**Week**	**Date**	**Lecture Topic**	**Assessment**
Java Foundations	1	3/08/2020	Subject Introduction	
		5/08/2020	Java - A Quick Tour	
OOP Fundamentals	2	10/08/2020	Classes and Objects 1	
		12/08/2020	Classes and Objects 2	
	3	17/08/2020	Classes and Objects 3	
		19/08/2020	Arrays and Strings	
	4	24/08/2020	Input and Output	
		26/08/2020	Software Tools/Bagel	
	5	31/08/2020	Inheritance I	
		2/09/2020	Inheritance 2	
		4/09/2020		Project 1 Released (20.59 pm)
	6	7/09/2020	Interfaces	
		9/09/2020	Revision	
Advanced OOP and Software Design	7	14/09/2020	**Mid Semester Test**	
		16/09/2020	Class Diagrams	
		18/09/2020		Project 1 Due (20:59 pm)
				Project 2 Released (20.59 pm)
	8	21/09/2020	Generics	
		23/09/2020	Collections and Maps	
	9	28/09/2020	Exceptions	
		30/09/2020	Design Patterns I	
		30/09/2020		Project 2A Due (20.59 pm)
	NT	5/10/2020		
		7/10/2020		
	10	12/10/2020	Design Patterns 2	
		14/10/2020	Sofware Testing and Design	
	11	19/10/2020	Asynchronous Programming	
		21/10/2020	Advanced Java Concepts	
		23/10/2020		Project 2B Due (20.59 pm)
Wrap Up	12	26/10/2020	Revision	
		28/10/2020	Wrapup - Exam	

Last Updated: 23-07-2020, 4:30 pm

Workshops

Each week there will be a 2 hour workshop:

- Workshops start in **week 2**
- Workshops contain practical exercises that give you hands-on experience on concepts covered in the lectures
- Doing these exercises will be important for you to understand the concepts, do your projects and final examination
- Workshops will be assessed: one mark will be assigned for each workshop, which will count 10% towards the final mark - more details in later slides

Assessment Breakdown

Project 1	8%
Project 2A	6%
Project 2B	16%
Mid-semester Test	10%
Workshops	10%
Final Exam	50%

Hurdle:
To pass the subject, students must obtain at least 50% overall, 15/30 in project work, and 30/60 in the mid-semester test and end-of-semester written examination combined.

Assessment Dates and Deadlines

Mid-semester Test	Monday 14th September (Week 7)
Project 1	Due 20:59pm, Friday 18th September (Week 7)
Project 2A	Due 20:59pm, Wednesday 30th September (Week 9)
Project 2B	Due 20:59pm, Friday 23rd October (Week 11)
Workshops	Assessed at the end of each workshop

Assessment - Workshops

- Workshops will be assessed:
  - students who complete workshop exercises and keep up with the subject topics generally perform well is the subject
  - last semester student feedback highlighted the lack of interactions with other students due to shifting to on line mode
- Each workshop is assigned 1 mark
- To obtain the 1 mark assigned for a workshop:
  - you must join the Zoom session for your assigned workshop;
  - join a small group your workshop demonstrator will assign you to during the workshop; and
  - in your group (or sometimes individually), work on a question selected by the workshop demonstrator and show a satisfactory solution (or a reasonable attempt)
- Final workshop mark will be the sum of workshop marks earned during the semester weighted to 10%

Textbook

- The subject content will be based off Absolute Java by Walter Savitch. You can buy the textbook (4th edition or higher) if you wish, but this is **not required**.

Editing Tools

- Any new language has a learning curve
- To reduce the impact, we'll be using Grok initially to give you programming practice
- Starting from week 2, we will introduce IntelliJ, an Integrated Development Environment (IDE)
- You will need to use an IDE for the Projects

Academic Misconduct

- Work with friends if you like on **workshop** questions
- All **assessed** work (projects) is to be done by you alone
- You can discuss overall approach to solving problems with peers or others
- **Do not** show your code to peers, in person or electronically, or ask peers for code
- When in doubt, ask lecturer or tutor
- Sophisticated software is used to identify cheating
- Cheating is grounds for disciplinary action; repeat offence is grounds for expulsion
- See `https://academichonesty.unimelb.edu.au/`

Student Representatives

The first two expressions of interest via email to me (karus@unimelb.edu.au) will be selected as student representatives.

Extra Resources

Practice (and exam-like) problems:

- Hackerrank
- Codecademy
- Codesignal

Changes this Semester

- We will be completely online - delivery and assessments
- Assessment is different from what is stated in the handbook
- Instead of tutorials we will have 2-hour workshops - labs that have hands-on practical questions
- Workshops assessed based on participation

How This Semester Works

- Lecture slides and a pre-recorded lecture for each lecture slot will be made available before the assigned lecture time
- During the assigned lecture times (Mondays 11am -12 pm and Wednesdays 2.15 pm - 3.15 pm) the lecturer will be on Zoom to answer questions related to the topic - for the remaining time after questions there will be discussions and quizzes
- Grok worksheets will reinforce your knowledge in the Java foundations, so please do them
  - **On Canvas go to Modules, scroll down to workshops, click on "Grok Start Here" to register with Grok - if you do not do this you will not be able to start using Grok.**
  - Most of the code should be intuitive, or at least make some sense
  - Some things won't "click" for a few weeks, when we add the finer details
- We will teach you Object Oriented Programming concepts throughout the semester, enabling you to write good **Object Oriented Programs in Java**

Grok

- Worksheets 1-7 now available
- Worksheets teach content to *complement* and *reinforce* the lectures
- Go at your own pace
- Not assessed, does not contribute to your marks
- Don't just answer the questions; it is assumed you will have **read the slides** as well

Let's take a quick look at the worksheets.

How to Better Engage in the Subject

- Attend lectures and participate in quizzes, activities and discussions
  - Anecdotal evidence shows that students who attend lectures do better in subjects
- Attend the workshop you are enrolled in and participate
  - Your demonstrators are the best resource for **specific** questions, particularly regarding the projects
- Be active on subject forums
- Practice programming -*practice makes you at least close to perfect!*
- Ask questions, **lots** of questions

Final Notes

- The slides will be available on Canvas, at least two days before the lecture
  - However, I reserve the right to do amendments to the slides all the way until the lecture (sometimes after the lecture in case I find errors, and I will let you know in such cases)
  - I will include the last updated date and time on Canvas, so please re-download the slides before the lecture in case there is an update after the time you downloaded
- The lecture recording will be available on Canvas, at least one day before the lecture time slot
- Please go through slides and the lecture recording and attend the Zoom lecture ready to ask questions and participate in quizzes