School of Computer Science and Engineering (CSE)

COMP9021 Principles of Programming

2024 Term 1

Week 1

Dr Rachid Hamadi (Lecturer in Charge)

Outline

- Course Introduction
- Moodle
- Ed Platform
- CSE Machines
- Academic Honesty and Plagiarism
- Week 1 - Python Programming Fundamentals
- Week 1 - From Problem Description to Python Program Example
- Week 1 - Notes 1 Turing Machines
- Q & A

Course Introduction

Course Introduction

- Moodle Course Website
  https://moodle.telt.unsw.edu.au/course/view.php?id=81675

- Lectures: **Monday/Wednesday 18:00 - 20:00**
  Location: **Keith Burrows Theatre (K-J14-G5)**

- Consultation/Tutorial: **1 hour per week**
  Location: **F2F** or **Online** using **Blackboard Collaborate**

Course Summary

- This course provides students with solid **conceptual knowledge** and **practical skills** of both **generic programming techniques** and **Python programming**

- The features of the language are covered to a significant depth, and there is a strong emphasis on problem solving

- There is a lot of contents to study in limited time and the learning curve is not gentle

Course Resources

- No textbook and the provided material is self-contained
- Ed resources
- Jupyter notebook sheets, together with static PDF files produced from those, will be provided as notes
- Some of the notes are complemented with automatically produced videos

Course Resources (cont'd)

- Jupyter notebook sheets offer many advantages over the more traditional lecture notes

- Cells that make up a Jupyter notebook sheet can be edited, cells can be added or deleted, cells that contain code can be executed, allowing students to guess what the output will be and check that the guess is correct

- This will let students play a more active role when they learn from existing code

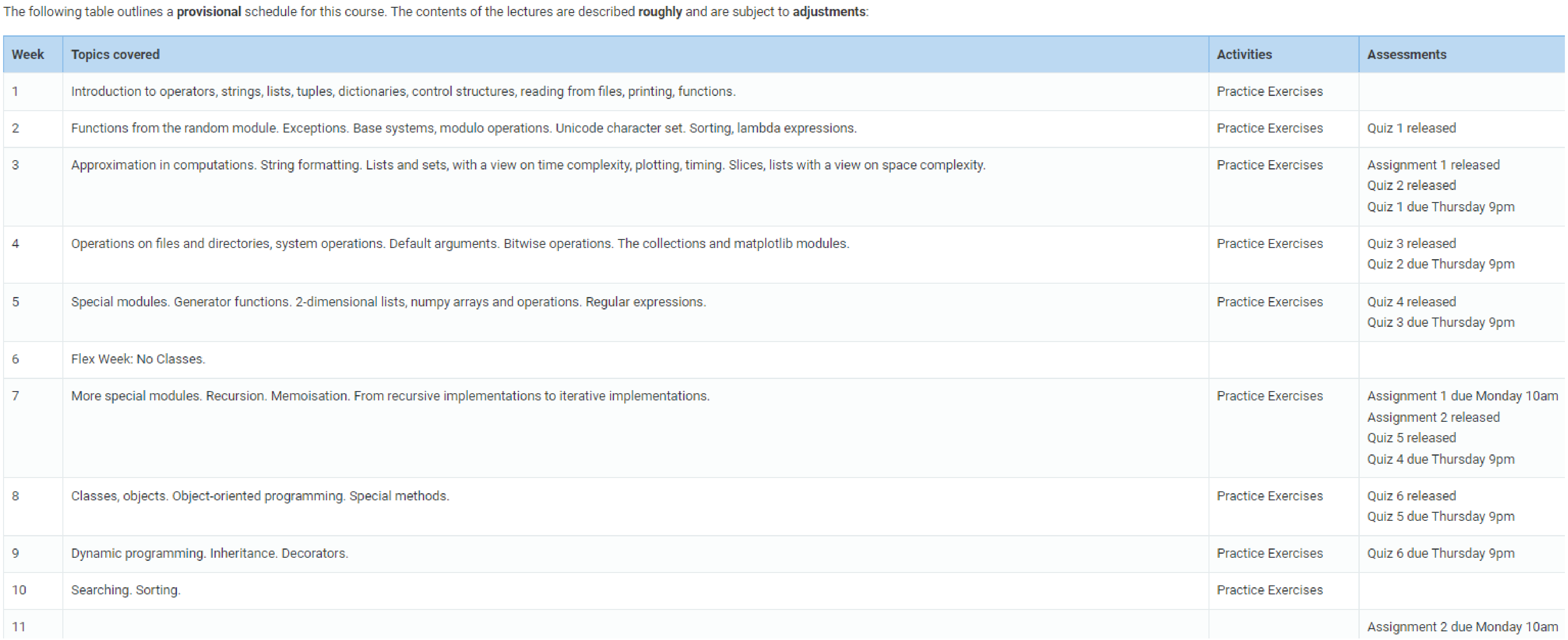
Assumed Knowledge

This course does not assume any prior knowledge of programming in general, or of Python programming in particular, as its content is self-contained for students with the expected mathematical background

Learning outcomes

1. Design, implement and test programs written in a language with procedural, object-oriented, and functional constructs
2. Apply Python language, including advanced syntax and programming techniques
3. Analyse what happens behind the scene when operating on Python data types, with an understanding of efficiency and memory use
4. Understand fundamental data structures and algorithms
5. Design programs to solve small to medium scale problems
6. Create clear, reliable, well-structured, well-tested, well-documented programs
7. Apply appropriate tools, in particular, for editing, testing and debugging

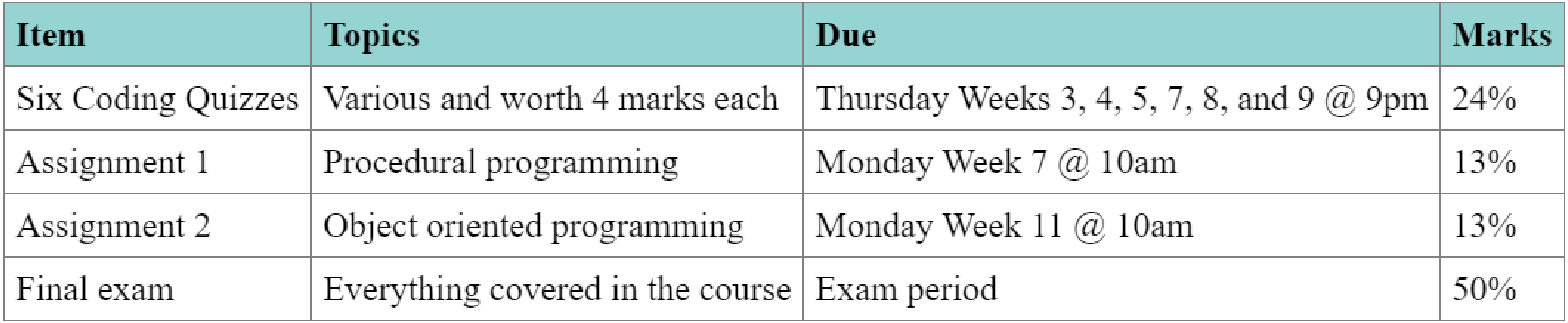
Course Schedule



The following table outlines a **provisional** schedule for this course. The contents of the lectures are described **roughly** and are subject to **adjustments**:

Week	Topics covered	Activities	Assessments
1	Introduction to operators, strings, lists, tuples, dictionaries, control structures, reading from files, printing, functions.	Practice Exercises	
2	Functions from the random module. Exceptions. Base systems, modulo operations. Unicode character set. Sorting, lambda expressions.	Practice Exercises	Quiz 1 released
3	Approximation in computations. String formatting. Lists and sets, with a view on time complexity, plotting, timing. Slices, lists with a view on space complexity.	Practice Exercises	Assignment 1 released Quiz 2 released Quiz 1 due Thursday 9pm
4	Operations on files and directories, system operations. Default arguments. Bitwise operations. The collections and matplotlib modules.	Practice Exercises	Quiz 3 released Quiz 2 due Thursday 9pm
5	Special modules. Generator functions. 2-dimensional lists, numpy arrays and operations. Regular expressions.	Practice Exercises	Quiz 4 released Quiz 3 due Thursday 9pm
6	Flex Week: No Classes.		
7	More special modules. Recursion. Memoisation. From recursive implementations to iterative implementations.	Practice Exercises	Assignment 1 due Monday 10am Assignment 2 released Quiz 5 released Quiz 4 due Thursday 9pm
8	Classes, objects. Object-oriented programming. Special methods.	Practice Exercises	Quiz 6 released Quiz 5 due Thursday 9pm
9	Dynamic programming. Inheritance. Decorators.	Practice Exercises	Quiz 6 due Thursday 9pm
10	Searching. Sorting.	Practice Exercises	
11			Assignment 2 due Monday 10am

Assessment Tasks Overview



Item	Topics	Due	Marks
Six Coding Quizzes	Various and worth 4 marks each	Thursday Weeks 3, 4, 5, 7, 8, and 9 @ 9pm	24%
Assignment 1	Procedural programming	Monday Week 7 @ 10am	13%
Assignment 2	Object oriented programming	Monday Week 11 @ 10am	13%
Final exam	Everything covered in the course	Exam period	50%

Moodle

Moodle

Used for

- Lecture Recordings
- Online Consultations/Tutorials

Ed Platform

Ed Platform

Used for

- Everything else
- Quizzes submission
- Assignments submission

CSE Machines

CSE Machines

Used for

- Practicing (IDLE, Spyder, Jupyter Notebook)
- Final Exam
- https://vlabgateway.cse.unsw.edu.au/

Academic Honesty and Plagiarism

Week 1 - Python Programming Fundamentals

Week 1 - From Problem Description to Python Program Example

Week 1 - Notes 1 Turing Machines

Q & A