



COMP1010

Fundamentals of Computer Science

Session 1, In person-scheduled-weekday, North Ryde 2023

School of Computing

Contents

<u>General Information</u>	2
<u>Learning Outcomes</u>	2
<u>General Assessment Information</u>	3
<u>Assessment Tasks</u>	4
<u>Delivery and Resources</u>	7
<u>Unit Schedule</u>	8
<u>Policies and Procedures</u>	9
<u>Changes from Previous Offering</u>	11

Disclaimer

Macquarie University has taken all reasonable measures to ensure the information in this publication is accurate and up-to-date. However, the information may change or become out-dated as a result of change in University policies, procedures or rules. The University reserves the right to make changes to any information in this publication without notice. Users of this publication are advised to check the website version of this publication [or the relevant faculty or department] before acting on any information in this publication.

General Information

Unit convenor and teaching staff

Convenor, Lecturer

Michael Lay

michael.lay@mq.edu.au

Contact via Contact via Email

Please see iLearn

Convenor, Lecturer

Ansgar Fehnker

ansgar.fehnker@mq.edu.au

Contact via Contact via Email

4 Research Park Drive

Please see iLearn

Credit points

10

Prerequisites

(COMP1000 or COMP115) or admission to (BActStud or BActStudBSc or BAppFinBActStud or BActStudBProfPrac)

Corequisites

Co-badged status

Unit description

This unit studies programming as a systematic discipline and introduces more formal software design methods. Programming skills are extended to include elementary data structures and abstract data types. There is a strong emphasis on problem solving and algorithms, including aspects of correctness, complexity and computability.

Important Academic Dates

Information about important academic dates including deadlines for withdrawing from units are available at <https://www.mq.edu.au/study/calendar-of-dates>

Learning Outcomes

On successful completion of this unit, you will be able to:

ULO1: apply enhanced problem solving skills to develop algorithms

ULO2: implement programs from algorithms, showing an understanding of control flow

ULO3: adhere to standard software engineering practices, including documentation, unit testing and debugging

ULO4: compare different methods available for the same problem in terms of efficiency and other criteria

ULO5: demonstrate foundational learning skills including active engagement in their learning process

General Assessment Information

WEEKLY PROGRAMMING QUIZZES

Your **best 7 out of 8** weekly quizzes will be used for your final grade. Each weekly quiz is worth approximately 2.86% for a total of 20%. Thus, there is 1 grace attempt.

PRACTICAL EXAM 2 HURDLE

The practical exam will assess students on topics discussed from the entire semester. **This is a hurdle exam. You must get at least 40 out of 100 to clear this hurdle.** Two attempts are provided, with marks of the **second-attempt capped to the minimum passing mark.**

Attempt 1: Week 13 practical class

Attempt 2 : Final exam period

ASSIGNMENTS

The assignments will assess students on various topics discussed during the semester. Each assignment is composed of two parts, Part A and Part B.

- **Part A** will require you to submit your assignment files online using the relevant submission box on iLearn.

- **Part B** will be a 'live coding session', assessed during your registered workshop.

Part A will allow for late submission following the standard late submission policy as detailed below. Due to the assessment nature of Part B, **late submissions will not be accepted for Part B.**

Late Assessment Submission Penalty

Unless a Special Consideration request has been submitted and approved, a 5% penalty (of the total possible mark) will be applied each day a written assessment is not submitted, up until the 7th day (including weekends). After the 7th day, a grade of '0' will be awarded even if the assessment is submitted. Submission time for all written assessments is set at **11:55 pm**. A 1-hour grace period is provided to students who experience a technical concern.

For any late submission of time-sensitive tasks, such as scheduled tests/exams, performance assessments/presentations, and/or scheduled practical assessments/labs, students need to submit an application for [Special Consideration](#).

Assessments where Late Submissions will be accepted

In this unit, late submissions will be accepted as follows:

- Assignment 1 - Part A: YES, Standard Late Penalty applies
- Assignment 1 - Part B: NO
- Assignment 2 - Part A: YES, Standard Late Penalty applies
- Assignment 2 - Part B: NO
- Weekly Programming Quizzes: NO
- Practical Exam 1: NO
- Practical Exam 2: NO

Assessment Tasks

Name	Weighting	Hurdle	Due
Weekly Programming Quizzes	20%	No	Weeks 2-6, 9-11
Practical exam 1	20%	No	Week 7
Practical exam 2	40%	Yes	Week 13 and Final Exam Period
Assignment 1	10%	No	Part A: First week of Mid Session Break and Part B: Week 8
Assignment 2	10%	No	Part A: Week 10 and Part B: Week 12

Weekly Programming Quizzes

Assessment Type ¹: Practice-based task

Indicative Time on Task ²: 10 hours

Due: **Weeks 2-6, 9-11**

Weighting: **20%**

Weekly programming quizzes in the practical classes.

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms

- implement programs from algorithms, showing an understanding of control flow
- demonstrate foundational learning skills including active engagement in their learning process

Practical exam 1

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 10 hours

Due: **Week 7**

Weighting: **20%**

Practical exam 1 covers all topics up to and including recursion

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms
- implement programs from algorithms, showing an understanding of control flow
- compare different methods available for the same problem in terms of efficiency and other criteria
- demonstrate foundational learning skills including active engagement in their learning process

Practical exam 2

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 25 hours

Due: **Week 13 and Final Exam Period**

Weighting: **40%**

This is a hurdle assessment task (see [assessment policy](#) for more information on hurdle assessment tasks)

Practical exam 2 covers topics from the entire semester.

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms
- implement programs from algorithms, showing an understanding of control flow
- compare different methods available for the same problem in terms of efficiency and other criteria

- demonstrate foundational learning skills including active engagement in their learning process

Assignment 1

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 20 hours

Due: **Part A: First week of Mid Session Break and Part B: Week 8**

Weighting: **10%**

Assignment 1 assesses students on the first 4 weeks of lecture content.

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms
- implement programs from algorithms, showing an understanding of control flow
- adhere to standard software engineering practices, including documentation, unit testing and debugging
- demonstrate foundational learning skills including active engagement in their learning process

Assignment 2

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 20 hours

Due: **Part A: Week 10 and Part B: Week 12**

Weighting: **10%**

Assignment 2 assesses students on contents from the entire semester.

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms
- implement programs from algorithms, showing an understanding of control flow
- compare different methods available for the same problem in terms of efficiency and other criteria
- demonstrate foundational learning skills including active engagement in their learning process

¹ If you need help with your assignment, please contact:

- the academic teaching staff in your unit for guidance in understanding or completing this type of assessment
- the [Writing Centre](#) for academic skills support.

² Indicative time-on-task is an estimate of the time required for completion of the assessment task and is subject to individual variation

Delivery and Resources

CLASSES

Each week you should attend

- two hours of lectures (details to be announced via iLearn),
- two hour practical class

For details of days, times and rooms, consult the [timetables webpage](#).

Note that Lectures and Practical classes commence in week 1.

You should have selected a practical class during enrolment. **You should attend the practical class in which you are enrolled.** You won't always get the class of your choice. Check availabilities via **eStudent** regularly. If ALL practical classes are full, only then, contact the convenor.

Please note that you are **required** to submit work regularly. You will get the help that you need by attending your practical class. Failure to submit work may result in you failing the unit (see the precise requirements in the "Grading Standards" section) or being excluded from the final examination.

TEXTS AND/OR MATERIALS

Lecture notes, Practical classes and Video tutorials: details to be announced via iLearn

Recommended Textbooks:

1. **T. Gaddis, Starting out with Java: From control structures through objects (Pearson), Global edition (6th). ISBN 9781292110653**
 - [Online edition](#) of this book is available through MQ Library. There can be up to 5 simultaneous accesses.
2. **Kathy Siera, Bert Bates, Head First Java, 2nd edition. ISBN 9780596009205**

TECHNOLOGY USED AND REQUIRED

Audio and Video Lecture

Digital recordings of lectures are available from within iLearn via **Active Learning Platform**.

Technology

- [Java SE](#) - download the latest Java SE to be compatible with the labs.
- [Eclipse](#) (preferred, troubleshooting provided) or [Visual Studio Code](#) (if you are proficient, independent) - the IDEs we shall be using during the session.
- Learning Management System [iLearn](#).
- <https://code2flow.com/> for better understanding of control flow.
- <http://codingbat.com/> for programming exercises.

Discussion Boards

The unit makes use of forums hosted within [iLearn](#). Please post questions there, they are monitored by the unit staff.

Unit Schedule

Note that three important themes will pervade the entire unit:

1. **Problem-solving.** A crucial skill for all of the weekly topics will be to write appropriate code to meet a given problem specification. This theme relates to the first two learning outcomes for this unit.
2. **Software development.** The use of JUnit testing framework is an important development practice that will be taught from the beginning and used throughout the unit. This theme relates to the third learning outcome of this unit.
3. **Comparing different solution methods.** Very often different algorithms are available for the same problem. Another important skill to develop throughout this unit is the ability to compare different algorithms in terms of efficiency and other criteria. This theme relates to the fourth learning outcome of this unit.

Unit Schedule

NOTE: This is a tentative schedule and subject to minor changes.

Week	Topic	Pre-lecture readings from Lecture Notes (COMP1010)	Assessments Due	Weekly Quiz
1	Programming environment	1, 2		
2	Problem-solving, JUnit testing	3, 4		Yes
3	Classes and Objects - 1	5, 6, 7		Yes

4	Classes and Objects - 2	8, 9, 10		Yes
5	Recursion - 1	11, 12		Yes
6	Recursion - 2	13, 14, 15		Yes
7	List Interface, ArrayList class	16, 17, 18	Practical Exam 1 - 20%	
	Break		Assignment 1 - Part A (Sunday 11:55pm)	
	Break			
8	Recursive data structures - 1	22	Assignment 1 - Part B	
9	Recursive data structures - 2	23		Yes
10	Sorting Algorithm(s)	19	Assignment 2 - Part A (Sunday 11:55pm)	Yes
11	Stacks and Queues	20		Yes
12	Special Topic		Assignment 2 - Part B	
13	TBA		Practical Exam 2 - 40%	
	Final Exam Period (Date TBA)		Practical Exam 2 - 40% (Attempt 2, capped at 40 marks)	

Policies and Procedures

Macquarie University policies and procedures are accessible from [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au). Students should be aware of the following policies in particular with regard to Learning and Teaching:

- [Academic Appeals Policy](#)
- [Academic Integrity Policy](#)
- [Academic Progression Policy](#)
- [Assessment Policy](#)
- [Fitness to Practice Procedure](#)
- [Assessment Procedure](#)
- [Complaints Resolution Procedure for Students and Members of the Public](#)

- [Special Consideration Policy](#)

Students seeking more policy resources can visit [Student Policies \(https://students.mq.edu.au/support/study/policies\)](https://students.mq.edu.au/support/study/policies). It is your one-stop-shop for the key policies you need to know about throughout your undergraduate student journey.

To find other policies relating to Teaching and Learning, visit [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au) and use the [search tool](#).

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: <https://students.mq.edu.au/admin/other-resources/student-conduct>

Results

Results published on platform other than [eStudent](#), (eg. iLearn, Coursera etc.) or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in [eStudent](#). For more information visit ask.mq.edu.au or if you are a Global MBA student contact globalmba.support@mq.edu.au

Academic Integrity

At Macquarie, we believe [academic integrity](#) – honesty, respect, trust, responsibility, fairness and courage – is at the core of learning, teaching and research. We recognise that meeting the expectations required to complete your assessments can be challenging. So, we offer you a range of resources and services to help you reach your potential, including free [online writing and maths support](#), [academic skills development](#) and [wellbeing consultations](#).

Student Support

Macquarie University provides a range of support services for students. For details, visit <http://students.mq.edu.au/support/>

The Writing Centre

[The Writing Centre](#) provides resources to develop your English language proficiency, academic writing, and communication skills.

- [Workshops](#)
- [Chat with a WriteWISE peer writing leader](#)
- [Access StudyWISE](#)
- [Upload an assignment to Studiosity](#)
- [Complete the Academic Integrity Module](#)

The Library provides online and face to face support to help you find and use relevant information resources.

- [Subject and Research Guides](#)

- [Ask a Librarian](#)

Student Services and Support

Macquarie University offers a range of [Student Support Services](#) including:

- [IT Support](#)
- [Accessibility and disability support](#) with study
- Mental health [support](#)
- [Safety support](#) to respond to bullying, harassment, sexual harassment and sexual assault
- [Social support including information about finances, tenancy and legal issues](#)
- [Student Advocacy](#) provides independent advice on MQ policies, procedures, and processes

Student Enquiries

Got a question? Ask us via [AskMQ](#), or contact [Service Connect](#).

IT Help

For help with University computer systems and technology, visit http://www.mq.edu.au/about_us/offices_and_units/information_technology/help/.

When using the University's IT, you must adhere to the [Acceptable Use of IT Resources Policy](#). The policy applies to all who connect to the MQ network including students.

Changes from Previous Offering

Quizzes have been replaced with two in class Practical Exams.

Weekly submissions have been replaced with weekly in class CodeRunner quizzes.

There are now two assignments instead of three. Each assignment is now comprised of two parts, an online submission and a live coding session.