



COMP1010

Fundamentals of Computer Science

Session 1, In person-scheduled-weekday, North Ryde 2024

School of Computing

Contents

<u>General Information</u>	2
<u>Learning Outcomes</u>	2
<u>General Assessment Information</u>	3
<u>Assessment Tasks</u>	5
<u>Delivery and Resources</u>	8
<u>Unit Schedule</u>	10
<u>Policies and Procedures</u>	12
<u>Changes from Previous Offering</u>	13
<u>Computing Drop-in Centre (CDC)</u>	14
<u>Changes since First Published</u>	14

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

Gaurav Gupta

gaurav.gupta@mq.edu.au

Contact via by email

Senior Teaching Assistant

Michael Lay

michael.lay@mq.edu.au

Contact via by email

Cameron Edmond

cameron.edmond@mq.edu.au

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

Module exams

There are three module exams, each worth 10% of the unit total.

Students have two attempts for each of the module exams, and the best attempt counts towards their final mark.

Exams are open in a certain window (typically 2 to 4 weeks) and students can self-enroll in sessions where they can sit these exams. Details (time, location) of the sessions will be announced via iLearn. Enrolment is on a first-come-first-serve basis. We will have sufficient sessions for all students. Enroll early to get a session that is convenient for you. Once a particular session is fully booked, we cannot manually add students to it. If you leave enrolling in these sessions until it's too late, you will not be granted a supplementary exam. Therefore, again, enroll as early as possible.

Students are required to select at least one slot that is not in the last week. For example, the Fundamentals exam is open from week 2 to week 4. Students have one attempt in the first two weeks (weeks 2, 3) and one more attempt in week 4.

Module	Worth	Exam time	Open	Close
Fundamentals	10%	40 minutes	Monday week 2	Saturday week 4
Classes and objects + ArrayList	10%	40 minutes	Monday week 7	Saturday first week of teaching recess
Recursion + Recursive Data Structures	10%	40 minutes	Monday week 10	Saturday week 12

Universal design for learning (UDL) - Module exams have been designed with universal design strategies in mind, which may remove the need for individual reasonable adjustments. All students will have up to 40 minutes to complete these exams which have been designed to be completed in 25 minutes. This extra time has been applied to all students. Students with IEAP arrangements do not need further overrides, and if they have more questions about UDL, they should contact the accessibility team at accessibility@mq.edu.au.

PRACTICAL EXAM HURDLE

The practical exam is designed as a 70-minute (all students get 105 minutes under UDL) closed-book iLearn quiz with CodeRunner questions that will assess students on topics discussed over the entire session. All students will get a chance to attempt the practical exam a second time, and the best of two attempts counts towards the final mark. Students with IEAP arrangements do not need further overrides, and if they have more questions about UDL, they should contact the accessibility team at accessibility@mq.edu.au.

The practical exam is open from **Monday week 13** to **Saturday week 16** and students can self-enroll in sessions where they can sit this exam. Details (time, location) of the sessions will be announced via iLearn. Enrolment is on a first-come-first-serve basis. We will have sufficient sessions for all students. Enroll early to get a session that is convenient for you. Once a particular session is fully booked, we cannot manually add students to it. If you leave enrolling in these sessions until it's too late, you will not be granted a supplementary exam. Therefore, again, enroll as early as possible.

IMPORTANT: This is a hurdle exam because it is the only assessment where we assess all topics taught during the session. You must get at least 40 out of 100 to clear this hurdle. If you fail this hurdle exam, you will not pass the unit (even if your raw mark in the unit is 50 or more).

ASSIGNMENTS

The assignments will assess students on various topics discussed during the session. Assignment 1 is an individual assessment worth 10%.

Assignment 2 is a group assignment composed of two parts,

- **Part A** (group component) will require you to submit your assignment files online using the relevant submission box on iLearn. **Only one group member should submit.**

- **Part B** (individual component) will be a live coding session designed as a 25-minute (all students get 40 minutes under UDL) closed-book iLearn quiz with CodeRunner questions, assessed during your registered workshop (or a time organized with the student wellbeing team, if applicable). Live coding for assignment 2 is open from **Monday week 13** to **Saturday week 16** and students can self-enroll in sessions where they can sit this exam. Details (time, location) of the sessions will be announced via iLearn. Enrolment is on a first-come-first-serve basis. We will have sufficient sessions for all students. Enroll early to get a session that is convenient for you. Once a particular session is fully booked, we cannot manually add students to it. If you leave enrolling in these sessions until it's too late, you will not be granted a supplementary exam. Therefore, again, enroll as early as possible.

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 of the task) will be applied for each day a written report or presentation 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. The submission time for all uploaded assessments is **11:55 pm**. A 1-hour grace period will be 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, please apply for Special Consideration.

For example, if an assessment is worth 100 marks, and you submit it 15 hours late, getting an original mark of 76, the mark after penalty will be $(76 - 5) = 71$. If you submit it 26 hours late, the mark would be 66, and so on.

Assessments where Late Submissions will be accepted

- Assignment 1: YES, Standard Late Penalty applies
- Assignment 2 - Part A: YES, Standard Late Penalty applies
- Assignment 2 - Part B: NO
- Module exams: NO
- Practical Exam: NO

Requirements to Pass this Unit

1. Achieve 50 or more marks overall.
2. Achieve 40 or more marks in the practical exam (you have 2 attempts).

Special Consideration

The Special Consideration Policy aims to support students who have been impacted by short-term circumstances or events that are serious, unavoidable, and significantly disruptive, and which may affect their performance in assessment. If you experience circumstances or events that affect your ability to complete the assessments in this unit on time, please inform the convenor and submit a Special Consideration request through ask.mq.edu.au.

Assessment Tasks

Name	Weighting	Hurdle	Due
Module exams	30%	No	Over the session (details below)
Assignment 1	10%	No	23:55 Sunday ending week 6
Assignment 2	20%	No	23:55 Sunday ending week 11

Name	Weighting	Hurdle	Due
<u>Practical exam</u>	40%	Yes	Weeks 13 to 16 (details below)

Module exams

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 25 hours

Due: **Over the session (details below)**

Weighting: **30%**

Multiple modules exams assessing individual topics over the session.

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

Assignment 1

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 15 hours

Due: **23:55 Sunday ending week 6**

Weighting: **10%**

Assignment 1 assesses students on problem-solving, data representation, and data analysis.

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
- 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 2

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 20 hours

Due: **23:55 Sunday ending week 11**

Weighting: **20%**

Assignment 2 assesses students on contents from the entire session, including recursive data structures.

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
- 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

Assessment Type ¹: Programming Task

Indicative Time on Task ²: 25 hours

Due: **Weeks 13 to 16 (details below)**

Weighting: **40%**

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

Practical exam covers topics from the entire session.

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

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

RESOURCES

We have developed several resources that can help you get a headstart as well as assist you over the session to stay on top of the game. These include,

- Practice questions
 - Codingbat: <https://codingbat.com/home/gaurav.gupta@mq.edu.au>
 - CodeRunner practice questions (on iLearn)
- YouTube videos
 - <https://www.youtube.com/playlist?list=PLA7fpUXGfHnEtL13WVPmz6eYO8QjgT-H1>
- [Computing Drop-in Centre](#) - The support centre is located in 4RPD G02 across Esc Cafe, and is open every weekday from 10:00-12:00, 13:00-15:00, 16:00-18:00. Although almost all support assistants are proficient in COMP1010, please use the live roster (to be announced on iLearn) to check which sessions have a support person skilled to help with COMP1010.

CLASSES

Each week you should attend

- a two-hour lecture, and,
- a two-hour practical class

For details of days, times, and rooms, consult your personalized timetables page.

WEEK 1 CLASSES

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.

TEXTS AND/OR MATERIALS

Lecture notes are available online at <https://softwaretechnologymq.github.io/>

Practical classes and Video teaching materials: 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.
- [Visual Studio Code](#) (preferred, troubleshooting provided) or [Eclipse](#) (if you are proficient, independent) - the IDEs we shall be using during the session.
- Learning Management System [iLearn](#).
- <https://code2flow.com/> for a 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.

Methods of Communication

We will communicate with you via your university email and through announcements on iLearn.

IMPORTANT -

1. Use discussion forums for general queries (that relate to a significant portion of the cohort)
2. Email the teaching team at comp1010@mq.edu.au for enquiries specific to you

COVID Information

For the latest information on the University's response to COVID-19, please refer to the Coronavirus infection page on the Macquarie website: <https://www.mq.edu.au/about/coronavirus-faqs>. Remember to check this page regularly in case the information and requirements change during session. If there are any changes to this unit in relation to COVID, these will be communicated via iLearn.

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

The schedule of activities is presented below.

Week	Topic	Pre-lecture readings (Section.COM P1010)	Assessment to be submitted on iLearn (23:55 on Sunday ending that week)	Open exams
1	Programming environment	1, 2		
2	Problem-solving, JUnit testing	3, 4		Fundamentals
3	Classes and Objects - 1	5, 6, 7		Fundamentals

4	Classes and Objects - 2	8, 9, 10		Fundamentals
5	List Interface, ArrayList class	16, 17, 18		
6	Recursion - 1	11, 12	Assignment 1	
7	Recursion - 2	13, 14, 15		Classes and objects + ArrayList
8	Recursive data structures - 1	22		Classes and objects + ArrayList
Teaching Recess Week 1				Classes and objects + ArrayList
Teaching Recess Week 2				
9	Recursive data structures - 2	23		
10	Sorting Algorithm(s)	19		Recursion + Recursive Data Structures
11	Stacks and Queues	20	Assignment 2	Recursion + Recursive Data Structures
12	Introduction to binary trees			Recursion + Recursive Data Structures
13	Transition to second-year programming units			1. Assignment 2 Live Coding 2. Practical exam
14 (Final exam period week 1)	-			1. Assignment 2 Live Coding 2. Practical exam
15 (Final exam period week 2)	-			1. Assignment 2 Live Coding 2. Practical exam
16 (Final exam period week 3)	-			1. Assignment 2 Live Coding 2. Practical exam

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

1. Number of assessments reduced
2. UDL applied to all in-class assessments

1. 50% extra time for all students
2. Exams not in Practical Class but in separate exam sessions
3. Added an (unassessed) introduction to binary trees

Computing Drop-in Centre (CDC)

COMP1010 is supported by the Computing Drop-in Centre (CDC) that operates daily (weekdays) from,

- 10:00 to 12:00
- 13:00 to 15:00
- 16:00 to 18:00

The web page at <https://students.mq.edu.au/study/faculties/science-and-engineering/drop-in-centre> contains further information including,

- location,
- the service agreement about what the centre can and cannot help you with,
- week in which the service begins,
- other units supported by the centre,
- roster (as not all time slots will have staff supporting every unit),

Changes since First Published

Date	Description
20/02/2024	preferred IDE updated
16/02/2024	Unit email address added. UDL query email added.
09/02/2024	Role updated to Senior Teaching Assistant
02/02/2024	Unit schedule typo fixed

Unit information based on version 2024.03 of the [Handbook](#)