



COMP1000

Introduction to Computer Programming

Session 2, In person-scheduled-weekday, North Ryde 2026

School of Computing

Contents

<u>General Information</u>	2
<u>Learning Outcomes</u>	3
<u>General Assessment Information</u>	3
<u>Assessment Tasks</u>	6
<u>Delivery and Resources</u>	8
<u>Unit Schedule</u>	9
<u>Policies and Procedures</u>	10
<u>Changes from Previous Offering</u>	13

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

Convener and lecturer

Ansgar Fehnker

ansgar.fehnker@mq.edu.au

Senior Teaching Assistant

Samantha Kuhn

samantha.kuhn@mq.edu.au

Senior Sessional Assistant

Aaron Chakerian

aaron.chakerian@mq.edu.au

Lecturer

Charanya Ramakrishnan

charanya.ramakrishnan@mq.edu.au

Credit points

10

Prerequisites

Corequisites

Co-badged status

Unit description

This unit is an introductory computer science unit, providing a practical introduction to basic computing and programming concepts. Students gain an understanding of, and practical experience in, computer programming; practical experience in implementing informal prose descriptions of problem solutions using an high-level language; an understanding of, and practical experience in, designing, coding, testing and debugging simple algorithms; and an understanding of the principle of incremental development. Other topics include the concept of program correctness; the differences between high-level languages, assembly languages and machine languages; the role played by compilers; and the execution of programs by computer hardware.

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 problem solving skills to develop algorithms that solve small to medium-sized computational problems

ULO2: design and write code to implement a program description in a high-level programming language

ULO3: use standard software engineering practices and tools to document, debug, test and manage their programs

ULO4: understand and apply appropriately the taught concepts such as variables, loops, functions, conditionals and compound data in the implementation of well-designed programs

ULO5: identify and describe ethical issues to academic and professional practice, and demonstrate active engagement in the learning process

General Assessment Information

Requirements to Pass

- Must obtain a mark of 50 (50% of 100) overall.

Programming Skills Demonstration

You will have to attend a live coding exercise demonstrating basic programming skills. It will carry 30 out of 100 marks. You will be asked to use the IDE to create, modify, change, extend, test, and debug a basic program. This is an in-person invigilated assessment, and the use of AI tools is not permitted. The assessment will take place during the workshops in week 6.

Please check iLearn for the exact dates, as dates may change if circumstances require it.

Before the *Programming Skills Demonstration*, there will be an online *Diagnostic Test*. It's highly recommended that you complete this formative assessment. If your *Diagnostic Test* mark is higher than your mark for the invigilated *Programming Skills Demonstration*, your final mark will be calculated using 10% of the *Diagnostic Test* mark and 90% of the mark for the *Programming Skills Demonstration*.

Programming Exam

There is one summative programming exam worth 30 out of 100 marks, assessing your understanding and application of the programming concepts covered in the unit. This is an in-person invigilated assessment, and the use of AI tools is not permitted. The assessment will take place during the workshop in week 11. This is an in-person invigilated assessment, and the use of AI tools is not permitted.

Before the *Programming Exam*, there will be an online *Practice Exam*. It's highly recommended

that you complete this formative assessment. If your *Practice Exam* mark is higher than your invigilated *Programming Exam*, your final mark will be calculated using 10% of the *Practice Exam* mark and 90% of the *Programming Exam* mark.

Project Handover and Review

Each week during your scheduled SGTA sessions, a teaching assistant will guide you through exercises related to the lecture topics. In addition to these exercises, your SGTA will also support you with your *Handover Project*. You are encouraged to submit your work-in-progress regularly to receive formative feedback. In **Week 7**, you are expected to submit a program for the formative *Mid-Session Checkpoint*. We will provide written formative feedback on your project's progress.

In **Week 12**, you will participate in an invigilated in-class *Handover Review* session. You will have to submit your *Handover Project* before this activity. The *Handover Review* is a closed-book, invigilated in-class activity. During this activity, you will complete a self-assessment of your submission against the minimum requirements. This self-assessment is intended to demonstrate your familiarity with the submitted *Handover Project*. It is an invigilated in-person assessment that will take place during your regular workshop hours.

Your grade for this assessment will be based on two components:

- The quality of your self-assessment during the handover
- The quality of the program you submit

Together, these components will contribute **40 marks out of 100** for the course.

If you fail to achieve a passing mark in the handover, you will have the opportunity to complete a **viva** (oral examination), covering the same topics. Unless a special consideration applies, a cap of 20 marks will be applied to the mark of the viva.

The use of AI during the *Handover Review* or second-chance vivas is not permitted. The use of AI tools to create the original submission is not explicitly prohibited. Students are, however, expected to be able to explain and answer questions on all aspects of their submission, including any parts that may have been generated with assistance.

The description of the project will be released before 20 September 2026.

It's highly recommended that you make a submission for the Week 7 formative *Mid-session Checkpoint*. If your mark for the *Mid-Session Checkpoint* is higher than the mark for the *Handover Project and Review*, your final mark will be calculated using 10% of the mark for the *Mid-Session Checkpoint* and 90% of the mark for the *Handover Project and Review*.

Late Submission for Assessments

In this unit, all assessments are in-person and in class. There are no separate submissions or submission deadlines.

As such, the following applies with respect to late submissions:

- Project Handover and Review: NO, unless Special Consideration is Granted

- Summative Exam: NO, unless Special Consideration is Granted
- Programming Skills Demonstration: NO, unless Special Consideration is Granted

Short extensions are **not offered** for any assessments in this unit.

The programming project must be submitted **before or at the start** of your in-class handover. **You will not be permitted to participate in the handover review unless your program has been submitted beforehand.**

Late Submission Penalties

All assessments in this unit are conducted **in person during scheduled class sessions**. There are **no standalone take-home submissions** and **no separate submission deadlines**.

Accordingly, late **submission penalties** do not apply.

Special Consideration

If you need extra time for an assessment due to serious or unexpected circumstances, you can apply for Special Consideration. Please keep the following in mind:

- For the **Programming Skills Demonstration** and the **Programming Exam**, students have the option to take part in a **second round** without needing to apply for Special Consideration.
- You should treat the **first round as your main opportunity**. The second round is there as a backup if you miss the first round due to unforeseen circumstances, or if you would like to try to improve your result.
- If you complete an attempt in any round, you will not be eligible for Special Consideration for that assessment.
- Before applying for Special Consideration for these two assessments, please check whether you are eligible for the second attempt. You should only submit a Special Consideration application if you were unable to complete **both** the first and second rounds. If this applies to you, make sure your application clearly explains why you could not take part in either opportunity.
- To apply for Special Consideration, visit the Special Consideration page [HERE](#)

AI Approach

- **Open** These assessments do not restrict the application of AI tools. However, it is expected that the use of AI is acknowledged in an appropriate manner, similar to the expectation to provide references for other resources or tools that are used in producing your work. Any use of AI should comply with the University's [Responsible and Ethical Use of Artificial Intelligence Policy](#).
- **Observed** These tasks are either fully or partly observed or invigilated. AI use will either

not be permitted or will be restricted during the assessment.

For more information, please see <https://students.mq.edu.au/study/assessment-exams/assessments/ai-and-assessment>

Please note that all assessments in this unit are **observed**. Following an invigilated assessment, we will review submitted work for indicators of AI use. Any cases where AI use is detected during an invigilated assessment will be reported to the Academic Integrity Committee.

Assessment Tasks

Name	Weighting	Hurdle	Due	Groupwork/Individual	Short Extension	AI Approach
Programming Skills Demonstration	30%	No	In week 6, during your registered SGTA.	Individual	No	Observed
Programming Exam	30%	No	Week 11, during your registered SGTA.	Individual	No	Observed
Project Handover and Review	40%	No	Week 12, during your registered SGTA.	Individual	No	Observed

Programming Skills Demonstration

Assessment Type ¹: Experiential task

Indicative Time on Task ²: 20 hours

Due: **In week 6, during your registered SGTA.**

Weighting: **30%**

Groupwork/Individual: Individual

Short extension ³: No

AI Approach: Observed

Weekly tasks during the SGTAs prepare students for an in-lab exercise where they must demonstrate programming skills, such as using standard software engineering tools to create, modify, debug, test, and manage programs. Students will have an opportunity to retake this task to improve their mark.

On successful completion you will be able to:

- apply problem solving skills to develop algorithms that solve small to medium-sized computational problems
- design and write code to implement a program description in a high-level programming language

- use standard software engineering practices and tools to document, debug, test and manage their programs
- understand and apply appropriately the taught concepts such as variables, loops, functions, conditionals and compound data in the implementation of well-designed programs
- identify and describe ethical issues to academic and professional practice, and demonstrate active engagement in the learning process

Programming Exam

Assessment Type ¹: Examination

Indicative Time on Task ²: 24 hours

Due: **Week 11, during your registered SGTA.**

Weighting: **30%**

Groupwork/Individual: Individual

Short extension ³: No

AI Approach: Observed

This is a summative assessment of the unit's content. Students will have an opportunity to retake the assessment to improve their marks.

On successful completion you will be able to:

- apply problem solving skills to develop algorithms that solve small to medium-sized computational problems
- design and write code to implement a program description in a high-level programming language
- use standard software engineering practices and tools to document, debug, test and manage their programs
- understand and apply appropriately the taught concepts such as variables, loops, functions, conditionals and compound data in the implementation of well-designed programs
- identify and describe ethical issues to academic and professional practice, and demonstrate active engagement in the learning process

Project Handover and Review

Assessment Type ¹: Reflection task

Indicative Time on Task ²: 28 hours

Due: **Week 12, during your registered SGTA.**

Weighting: **40%**

Groupwork/Individual: Individual

Short extension ³: No

AI Approach: Observed

Formative programming tasks during the session will contribute to a final programming project, submitted during an invigilated in-class handover. Students must demonstrate their understanding of and familiarity with the project. Both the quality of the submitted program and the student's performance during the handover will be assessed. Students who do not pass the handover will have the opportunity to complete a viva, covering the same topics as the handover, for a passing mark.

On successful completion you will be able to:

- apply problem solving skills to develop algorithms that solve small to medium-sized computational problems
- design and write code to implement a program description in a high-level programming language
- use standard software engineering practices and tools to document, debug, test and manage their programs
- understand and apply appropriately the taught concepts such as variables, loops, functions, conditionals and compound data in the implementation of well-designed programs
- identify and describe ethical issues to academic and professional practice, and demonstrate active engagement in the 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
- [Academic Success](#) 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.

³ An automatic short extension is available for some assessments. Apply through the [Service Connect Portal](#).

Delivery and Resources

WEEK 1

Note that lectures commence in week 1. Workshops scheduled on Monday after the lecture and from Tuesday to Friday will also commence in week 1. Workshops scheduled before the lecture on Monday will commence in week 2.

CLASSES

Each week you should attend

- two-hour lecture
- two-hour workshop

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

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

TEXTS AND/OR MATERIALS

- Lecture notes, The Big Book of Assessments, and SGTA exercises are available on iLearn.
- Online tutorials on the Processing website: <https://processing.org/tutorials/>
- Learning Processing site (examples, exercises, videos): <http://learningprocessing.com/>

Textbook:

- **Learning Processing: A Beginner's Guide to Programming Images, Animation, and Interaction**, by Daniel Shiffman. Second edition, 2015. ISBN: 978-0123944436.
IMPORTANT: Online/Digital version is available [here on MQ Library](#). Click on "Elsevier ScienceDirect Books - Single Purchased Titles", log in using OneID credentials, and "Download all chapters".

Technology

- Programming environment: [Processing IDE](#)

Methods of Communication

The unit makes use of forums hosted within [iLearn](#). Please post questions there, they are monitored by the unit staff. For personal questions regarding the unit, please contact the super tutors or unit conveners.

Unit Schedule

Note, that dates and deadlines may change if circumstance requires. Please check iLearn for the latest updates.

Week	Topic	Assessment Activity
1	Getting Started in Processing	

2	Variables and arithmetic	
3	Decisions: Ifs, booleans, and boolean algebra	
4	Iteration: While and for-loops	Formative Feedback: Diagnostic Test
5	Loops and Ifs	
6	Functions	Summative Assessment: Programming Skills Demonstration
7	Objects and Classes	Formative Feedback: Mid-project Checkpoint
8	Arrays and array algorithms	
9	Understanding Data Passing: Value vs. Reference	
10	Program Design and Problem Solving	Formative Assessment: Practice Exam
11	Theory, Physics, and Processing Goodies.	Summative Programming Exam
12	Beyond Processing	Summative Assessment: Project Handover and Review
13	Revision	Summative Assessment: Project Handover and Review - Vivas

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 connect.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](#).

Academic Integrity

Using the work or ideas of another person, whether intentionally or not, and presenting them as your own without clear acknowledgement of the source is called [Plagiarism](#).

Macquarie University promotes awareness of information ethics through its [Academic Integrity Policy](#). This means that:

- all academic work claimed as original must be the work of the person making the claim;
- all academic collaborations of any kind must be acknowledged;
- academic work must not be falsified in any way; and
- when the ideas of others are used, these ideas must be acknowledged appropriately.

All breaches of the [Academic Integrity Policy](#) are serious and penalties apply. Students should be aware that they may fail an assessment task, a unit or even be excluded from the University for breaching the Academic Integrity Policy.

Assessment Policy

Students should familiarise themselves with their responsibilities under the [Assessment Policy](#), and notably the [Final Examination Procedure](#).

Grade Appeals

A student who has been awarded a final grade for a unit has the right to appeal that grade as outlined in the [Assessment Policy](#). Grade appeals apply to the final mark and the grade a student receives for a unit of study. They do not apply to results received for individual

assessment tasks.

Grade appeals must be submitted via ask.mq.edu.au within 15 working days from the published result date for the relevant unit. Before submitting a Grade Appeal, please ensure that you read the [Assessment Policy](#) and note valid grounds for appeals.

Students are expected to seek feedback on individual assessment tasks prior to the award of a final grade. Students also have the right to request generic feedback from the teaching staff on their overall performance in the unit, including in a final examination. This can be done at any time in the six-month period starting from the day on which the final grade of the relevant unit is published.

Student Support

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

Academic Success

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

- [Ask a Study Coach](#)
- [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 the [Service Connect Portal](#), or contact [Service Connect](#).

IT Help

For help with University computer systems and technology, visit <https://students.mq.edu.au/support/technology/service-desk>.

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

We value student feedback to be able to improve the way we offer our units continually. As such, we encourage students to provide constructive feedback via student surveys, to the teaching staff directly, or via the FSE Student Experience & Feedback link on the iLearn page.

To reduce the clustering of exams and assessment, we will spread second-round attempts over a longer period. However, because of Faculty deadlines and administrative requirements, there will be a final date after which we cannot offer any more second-chance attempts. Students should not expect unlimited opportunities for a second attempt.

Unit information based on version 2026.02 of the [Handbook](#)