COMP6010
Foundations of Computer Programming
Session 1, In person-scheduled-weekday, North Ryde 2024

School of Computing

Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Information</td>
<td>2</td>
</tr>
<tr>
<td>Learning Outcomes</td>
<td>2</td>
</tr>
<tr>
<td>General Assessment Information</td>
<td>3</td>
</tr>
<tr>
<td>Assessment Tasks</td>
<td>4</td>
</tr>
<tr>
<td>Delivery and Resources</td>
<td>7</td>
</tr>
<tr>
<td>Policies and Procedures</td>
<td>8</td>
</tr>
<tr>
<td>Changes from Previous Offering</td>
<td>10</td>
</tr>
<tr>
<td>Computing Drop-in Centre (CDC)</td>
<td>10</td>
</tr>
</tbody>
</table>

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 Email
By appointment

Lecturer
Mark Dras
mark.dras@mq.edu.au

Credit points
10

Prerequisites
Corequisites
Co-badged status

Unit description
This unit provides a foundation-level study of programming. The topics covered include programming environment and the process of program execution, variables, operators, boolean logic, control structures including conditions and loops, functions, storage of collections of items and performing operations on the same, and file management.

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 development skills such as test-driven development and debugging
ULO4: Understand and apply important foundation-level programming concepts of variables, operators, boolean logic, and control structures.
ULO5: Understand and apply important foundation-level programming concepts of functions and collections.

General Assessment Information

Requirements to Pass the Unit

To pass this unit, you must have:

- An overall mark of 50% or higher in the unit, AND,
- A mark of 50% or higher in practical exam 2.

Assignments (10% + 10%)

The programming assignments requires students to solve a real-life problem based on the contents covered during the semester. Submission of assignments will be done using the iLearn submission box.

Quizzes (20%)

4 quizzes over the course of 13 weeks. These quizzes will be taken during your registered practical class. Exact weeks can be found in the assessment tasks table.

Practical Exam 1 (20%)

In-class practical exam assessing contents covered during the first half of semester. This will be taken during your registered practical class.

Practical Exam 2 (40%)

In-class practical exam assessing contents during the entire semester. This will be taken during your registered practical class. Practical Exam 2 is a hurdle assessment, more information about the hurdle requirement for Practical Exam 2 is provided in the next section.

HURDLE

Practical Exam 2 is a hurdle task. You must make a serious attempt at this assessment. A serious attempt is considered an attempt where you have either completed all questions, or tried to complete as much of the exam as possible. This assessment is a hurdle assessment as it is the only assessment where we assess all topics during the session. There are two attempts at the hurdle assessment and the best mark of two attempts will be used. Both attempts are uncapped (eligible for full marks). The first attempt will be in the week 12 practical class and second in the week 13 practical class. You must get at least 50% in practical exam 2 to clear
**Assessment Tasks**

<table>
<thead>
<tr>
<th>Name</th>
<th>Weighting</th>
<th>Hurdle</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quizzes</td>
<td>20%</td>
<td>No</td>
<td>Weeks 4, 7, 9, 11</td>
</tr>
<tr>
<td>Practical Exam 1</td>
<td>20%</td>
<td>No</td>
<td>Week 8</td>
</tr>
<tr>
<td>Practical Exam 2</td>
<td>40%</td>
<td>Yes</td>
<td>Week 12 and 13</td>
</tr>
<tr>
<td>Assignment 1</td>
<td>10%</td>
<td>No</td>
<td>First week of mid semester break</td>
</tr>
<tr>
<td>Assignment 2</td>
<td>10%</td>
<td>No</td>
<td>Week 13</td>
</tr>
</tbody>
</table>

**Quizzes**

Assessment Type 1: Quiz/Test
Indicative Time on Task: 20 hours
Due: Weeks 4, 7, 9, 11
Weighting: 20%

Quizzes over the course of 13 weeks in practical classes (exact weeks to be determined based on timetabling and published in the unit guide).

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 development skills such as test-driven development and debugging
- Understand and apply important foundation-level programming concepts of variables, operators, boolean logic, and control structures.

Practical Exam 1
Assessment Type: Programming Task
Indicative Time on Task: 15 hours
Due: Week 8
Weighting: 20%

In-class practical exam as the invigilated assessment assessing contents covered during the first half of 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.
- Adhere to standard software development skills such as test-driven development and debugging
- Understand and apply important foundation-level programming concepts of variables, operators, boolean logic, and control structures.

Practical Exam 2
Assessment Type: Programming Task
Indicative Time on Task: 30 hours
Due: **Week 12 and 13**
Weighting: **40%**

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

In-class practical exam as the summative invigilated assessment assessing contents covered during 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.
- Adhere to standard software development skills such as test-driven development and debugging
- Understand and apply important foundation-level programming concepts of variables, operators, boolean logic, and control structures.
- Understand and apply important foundation-level programming concepts of functions and collections.

**Assignment 1**

Assessment Type 1: Programming Task
Indicative Time on Task 2: 10 hours
Due: **First week of mid semester break**
Weighting: **10%**

Programming assignment that requires students to solve a real-life problem based on the contents covered during the 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.
- Adhere to standard software development skills such as test-driven development and debugging
- Understand and apply important foundation-level programming concepts of variables, operators, boolean logic, and control structures.
Assignment 2

Assessment Type 1: Programming Task
Indicative Time on Task 2: 10 hours
Due: Week 13
Weighting: 10%

Programming assignment that requires students to solve a real-life problem based on the contents covered during the 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.
- Adhere to standard software development skills such as test-driven development and debugging
- Understand and apply important foundation-level programming concepts of variables, operators, boolean logic, and control structures.
- Understand and apply important foundation-level programming concepts of functions and collections.

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

Week 1 Classes
Lectures start in week 1.
Practical classes start in week 1.

Methods of Communication
We will communicate with you via your university email and through announcements on iLearn.
Queries to convenors can either be placed on the iLearn discussion board or sent to the unit convenor via the contact email on iLearn.

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 semester. If there are any changes to this unit in relation to COVID, these will be communicated via iLearn.

Resources

Lecture notes: Check iLearn and also refer to https://softwaretechnologymq.github.io/

Python (3.11.2 at the time of writing this guide): https://python.org/

Visual Studio Code: https://code.visualstudio.com/

Policies and Procedures

Macquarie University policies and procedures are accessible from Policy Central (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). 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) 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
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
We value student feedback to be able to continually improve the way we offer our units. 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 in the iLearn page.

Practical Exam 2 second attempt has been uncapped.

Computing Drop-in Centre (CDC)
COMP6010 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),
- zoom links for the evening sessions.