COMP1010
Fundamentals of Computer Science
Session 1, Weekday attendance, North Ryde 2020
Dept of Computing

Contents

<table>
<thead>
<tr>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Information</td>
<td>2</td>
</tr>
<tr>
<td>Learning Outcomes</td>
<td>3</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>Unit Schedule</td>
<td>8</td>
</tr>
<tr>
<td>Policies and Procedures</td>
<td>9</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 (weeks 1 to 8)
Gaurav Gupta
gaurav.gupta@mq.edu.au

Lecturer (weeks 9 to 12)
Matthew Roberts
matthew.roberts@mq.edu.au

Tutor
Daniel Sutantyo
daniel.sutantyo@mq.edu.au

Tutor
Sophie Kaelin
sophie.kaelin@mq.edu.au

Tutor
John Kim
j.kim@mq.edu.au

Tutor
Cooper Timewell
cooper.timewell@mq.edu.au

Tutor
Michael Lay
michael.lay@mq.edu.au

Tutor
Sarah Stanton
sarah.stanton@mq.edu.au

Tutor
Thomas Yap
thomas.yap@mq.edu.au

Credit points
10
<table>
<thead>
<tr>
<th>Important Academic Dates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information about important academic dates including deadlines for withdrawing from units are available at <a href="https://students.mq.edu.au/important-dates">https://students.mq.edu.au/important-dates</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Learning Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ULO1</strong>: apply enhanced problem solving skills to develop algorithms</td>
</tr>
<tr>
<td><strong>ULO2</strong>: implement programs from algorithms, showing an understanding of the underlying architecture of the computer</td>
</tr>
<tr>
<td><strong>ULO3</strong>: adhere to standard software engineering practices, including documentation, unit testing and debugging</td>
</tr>
<tr>
<td><strong>ULO4</strong>: compare different methods available for the same problem in terms of efficiency and other criteria</td>
</tr>
<tr>
<td><strong>ULO5</strong>: demonstrate foundational learning skills including active engagement in their learning process</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>General Assessment Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Students with reasonable adjustments</strong></td>
</tr>
<tr>
<td>Students with reasonable adjustment approvals will sit practical exams in relevant weeks on Friday 18:00 (in room 09WW-121)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Practical Exam 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>For practical exam 3, the <strong>best (out of 2 attempts) mark</strong> counts towards the final grade.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Late Submission</th>
</tr>
</thead>
<tbody>
<tr>
<td>No extensions will be granted without an approved application for Special Consideration. There will be a deduction of 20% of the total available marks made from the total awarded mark for</td>
</tr>
</tbody>
</table>
each 24 hour period or part thereof that the submission is late. For example, 25 hours late in submission for an assignment worth 10 marks – 40% penalty or 4 marks deducted from the total. No submission will be accepted after solutions have been posted.

**Assessment Tasks**

<table>
<thead>
<tr>
<th>Name</th>
<th>Weighting</th>
<th>Hurdle</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignment 1</td>
<td>5%</td>
<td>No</td>
<td>Week 6</td>
</tr>
<tr>
<td>Assignment 2</td>
<td>10%</td>
<td>No</td>
<td>Week 12</td>
</tr>
<tr>
<td>Practical Exam 1</td>
<td>5%</td>
<td>No</td>
<td>Week 4</td>
</tr>
<tr>
<td>Practical Exam 2</td>
<td>15%</td>
<td>No</td>
<td>Week 7</td>
</tr>
<tr>
<td>Practical Exam 3</td>
<td>30%</td>
<td>Yes</td>
<td>Week 13, 14</td>
</tr>
<tr>
<td>Final exam</td>
<td>35%</td>
<td>No</td>
<td>Exam Period</td>
</tr>
</tbody>
</table>

**Assignment 1**

Assessment Type 1: Programming Task  
Indicative Time on Task 2: 6 hours  
Due: **Week 6**  
Weighting: 5%

Assignment to assess problem solving skills, understanding of primitive data types and operations, control structures, arrays and methods.

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms
- implement programs from algorithms, showing an understanding of the underlying architecture of the computer
- 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: 15 hours
Assignment to assess problem solving skills and understanding of data structures (lists).

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms
- implement programs from algorithms, showing an understanding of the underlying architecture of the computer
- 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 1
Assessment Type 1: Programming Task
Indicative Time on Task 2: 5 hours
Due: Week 4
Weighting: 5%

Practical exam to assess problem solving skills with primitive data types and arrays

On successful completion you will be able to:

- apply enhanced problem solving skills to develop algorithms
- implement programs from algorithms, showing an understanding of the underlying architecture of the computer
- 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 2
Assessment Type 1: Programming Task
Indicative Time on Task 2: 10 hours
Due: Week 7
Weighting: 15%
Practical exam to assess problem solving, classes and objects, 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 the underlying architecture of the computer
• 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 3
Assessment Type 1: Programming Task
Indicative Time on Task 2: 15 hours
Due: Week 13, 14
Weighting: 30%
This is a hurdle assessment task (see assessment policy for more information on hurdle assessment tasks)

Practical exam to assess problem solving skills, recursion, classes and objects, and 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 the underlying architecture of the computer
• 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

Final exam
Assessment Type 1: Examination
Indicative Time on Task 2: 15 hours
Due: Exam Period
Weighting: 35%

Final exam to assess overall understanding of concepts covered in the unit.

On successful completion you will be able to:
• apply enhanced problem solving skills to develop algorithms
• implement programs from algorithms, showing an understanding of the underlying architecture of the computer
• 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

1 If you need guidance or support to understand or complete this type of assessment, please contact the Learning Skills Team

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

• three hours of lectures,
• two hour practical class

For details of days, times and rooms consult the [timetables webpage](https://unitguides.mq.edu.au/unit_offerings/123022/unit_guide/print).

**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: https://rebrand.ly/COMP1010LectureNotes

Video tutorials: https://rebrand.ly/COMP1010VideoTutorials


• Online edition of this book is available through MQ Library. There can be up to 5 simultaneous accesses. Click on "Full text available at: 2018 eTextbooks" and login with OneID and password.

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 and Visual Studio Code - the IDEs we shall be using during the session.
• Learning Management System iLearn
• 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.** Use of the JUnit testing framework is an important development practice which 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.
### Policies and Procedures

Macquarie University policies and procedures are accessible from Policy Central ([https://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central](https://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central)). 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
- Grade Appeal Policy
- Complaint Management Procedure for Students and Members of the Public
- Special Consideration Policy *(Note: The Special Consideration Policy is effective from 4 December 2017 and replaces the Disruption to Studies Policy.)*

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

If you would like to see all the policies relevant to Learning and Teaching visit Policy Central ([https://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central](https://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central)).
Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: https://students.mq.edu.au/study/getting-started/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

Student Support

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

Learning Skills

Learning Skills (mq.edu.au/learningskills) provides academic writing resources and study strategies to improve your marks and take control of your study.

• Workshops
• StudyWise
• Academic Integrity Module for Students
• Ask a Learning Adviser

Student Enquiry Service

For all student enquiries, visit Student Connect at ask.mq.edu.au

If you are a Global MBA student contact globalmba.support@mq.edu.au

Equity Support

Students with a disability are encouraged to contact the Disability Service who can provide appropriate help with any issues that arise during their studies.

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.