COMP1010
Fundamentals of Computer Science
Session 2, Special circumstances 2021

Department of Computing

Contents

General Information 2
Learning Outcomes 4
General Assessment Information 5
Assessment Tasks 5
Delivery and Resources 8
Unit Schedule 9
Policies and Procedures 10
Changes from Previous Offering 12
Grading Standards 12

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.

Notice
Some on-campus classes have moved online for the first two weeks of Session, before returning to campus in Week 3. If you are studying a unit outside of the primary Session 2 timetable, please contact your teaching staff team for further details.

Some classes/teaching activities cannot be moved online and must be taught on campus. To find out if you are enrolled in one of these classes/teaching activities, you can check to see if your unit is on the list of units with mandatory on-campus classes/teaching activities.

Your Unit Convenor will provide more information via an iLearn announcement when your iLearn unit becomes available.
General Information
Unit convenor and teaching staff

Convenor, Lecturer
Gaurav Gupta
gaurav.gupta@mq.edu.au
To be advised via iLearn

Lecturer
Bernad Mans
bernard.mans@mq.edu.au
To be advised via iLearn

Content Developer
Emilie Jemima Hyslop
emiliejemima.hyslop@mq.edu.au

Tutor
Naila Mukhtar
naila.mukhtar@mq.edu.au

Tutor
Nataly Falero
nataly.falero@mq.edu.au

Tutor
David Sanders
david.sanders@mq.edu.au

Tutor
Afroja Akter
afroja.akter@mq.edu.au

Tutor
Hemant Kumar
hemant.kumar@mq.edu.au

Tutor
Kshitij Bhargava
kshitij.bhargava@mq.edu.au

Tutor
Liam Ralston
liam.ralston@mq.edu.au

Tutor
Zawar Hussain
Image content: This page contains information about the unit COMP1010 Fundamentals of Computer Science. It includes details about tutor contacts, credit points, prerequisites, corequisites, co-badged status, unit description, important academic dates, and learning outcomes.

**Tutor**
- Alex Taylor
  - alexander.taylor@mq.edu.au
- Evan Gillard
  - evan.gillard@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://students.mq.edu.au/important-dates](https://students.mq.edu.au/important-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 the underlying architecture of the computer
- **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

Late Submission
For any assessment, extensions will only be granted if your askMQ application for Special Consideration is approved.

Weekly submissions
Late submissions are NOT accepted for weekly submissions unless you have special considerations approval.

Assignments
There will be a deduction of 20% (of the total marks) made from the total awarded mark for each 24 hour period or part thereof that the submission is late. For example,

- an assignment that is late by anywhere between one second and 24 hours will get a 20% penalty. Thus, if your raw mark is 68, it will become 48.
- an assignment that is late by anywhere between 24 hours 1 second and 48 hours will get a 40% penalty. Thus, if your raw mark is 68, it will become 28.
- and so on...

Submissions for assignments will NOT be accepted after the 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>Quizzes</td>
<td>20%</td>
<td>No</td>
<td>Week 6, 10, 12</td>
</tr>
<tr>
<td>Assignments</td>
<td>30%</td>
<td>No</td>
<td>Week 3, 8, 13</td>
</tr>
<tr>
<td>Weekly submissions</td>
<td>20%</td>
<td>Yes</td>
<td>Weeks 2 to 11</td>
</tr>
<tr>
<td>End-of-semester exam</td>
<td>30%</td>
<td>No</td>
<td>Exam period (weeks 14, 15)</td>
</tr>
</tbody>
</table>

Quizzes
Assessment Type ¹: Quiz/Test
Indicative Time on Task ²: 16 hours
Due: Week 6, 10, 12
Weighting: 20%

Online quizzes to assess the understanding of the several topics taught throughout the unit.
Students get only one attempt for each quiz.

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
- 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

Assignments
Assessment Type 1: Programming Task
Indicative Time on Task 2: 39 hours
Due: Week 3, 8, 13
Weighting: 30%

Take-home assignments during the semester to assess problem-solving skills in the domain of the unit topics.

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

Weekly submissions
Assessment Type 1: Practice-based task
Indicative Time on Task 2: 20 hours
Due: Weeks 2 to 11
Weighting: **20%**
This is a hurdle assessment task (see assessment policy for more information on hurdle assessment tasks)

Weekly submissions from the practice package provided. To clear the hurdle, students must submit **at least 8 out of 10** submissions that satisfy the minimum requirements, which may be different for each week.

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
- 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

**End-of-semester exam**
Assessment Type: Examination
Indicative Time on Task: 10 hours
Due: **Exam period (weeks 14, 15)**
Weighting: **30%**

End of semester exam to assess achievement of learning outcomes 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 the underlying architecture of the computer
- 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
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 Learning Skills Unit for academic skills support.

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 (delivered online, details to be announced via iLearn),
- two hour practical class

For details of days, times and rooms, consult the [timetables webpage](https://unitguides.mq.edu.au/unit_offerings/139998/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](https://rebrand.ly/COMP1010LectureNotes)

Practical classes: [https://rebrand.ly/COMP1010practicals](https://rebrand.ly/COMP1010practicals)


**Recommended Textbooks:**

   - 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** (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.

NOTE: This is a tentative schedule and subject to minor changes.
<table>
<thead>
<tr>
<th>No.</th>
<th>Topic</th>
<th>Weeks</th>
<th>Assessment</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Problem-solving, JUnit testing</td>
<td>15, 16</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>Classes and Objects - 1</td>
<td>17, 18, 19</td>
<td>Assignment 1 (Arrays) - 5%</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Classes and Objects - 2</td>
<td>20, 21, 22</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>Recursion - 1</td>
<td>23, 24, 25</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>Recursion - 2</td>
<td>26, 27</td>
<td>Quiz 1 (Classes) - 5%</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>List Interface, ArrayList class</td>
<td>28</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td><strong>2 weeks' recess</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Iterators</td>
<td>29, 30</td>
<td>Assignment 2 (Classes) - 10%</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>Custom-built ArrayList</td>
<td>31</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>10</td>
<td>Recursive data structures - 1</td>
<td>32</td>
<td>Quiz 2 (Recursion) – 7%</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>Recursive data structures - 2</td>
<td>33</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>12</td>
<td>Advanced topics</td>
<td>34</td>
<td>Quiz 3 (ArrayList class) - 8%</td>
<td>No</td>
</tr>
<tr>
<td>13</td>
<td>Revision</td>
<td></td>
<td>Assignment 3 (Recursive data structures) - 15%</td>
<td>No</td>
</tr>
<tr>
<td>14, 15</td>
<td>Final Exam – 30%</td>
<td></td>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>

**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). 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). 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).

**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

**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 help you improve your marks and take control of your study.

- **Getting help with your assignment**
- **Workshops**
- **StudyWise**
- **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 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.

Changes from Previous Offering
The number of assignments has been reduced from 4 to 3, weights have been adjusted.

The number of quizzes has been reduced from 4 to 3, weights have been adjusted.

Grading Standards
The unit will be graded according to the following general descriptions of the letter grades as specified by Macquarie University. In the course of the unit, these grade descriptions will be discussed with respect to example projects.

**High Distinction (HD, 85-100):** provides consistent evidence of deep and critical understanding in relation to the learning outcomes. There is substantial originality and insight in identifying, generating and communicating competing arguments, perspectives or problem solving approaches; critical evaluation of problems, their solutions and their implications; creativity in application as appropriate to the discipline.

In the context of this unit, the project has a good design, and has used some data that is interesting or non-obvious, or has required some effort to obtain or use. It involves a good analysis of the data, and fairly extensively draws on the techniques and tools presented in the unit and possibly on others discovered independently by the student. The project is described in a report and a presentation that are well-structured and essentially free from errors; these would be of a standard that could be presented at a conference with little or no polishing.

**Distinction (D, 75-84):** provides evidence of integration and evaluation of critical ideas, principles and theories, distinctive insight and ability in applying relevant skills and concepts in relation to learning outcomes. There is demonstration of frequent originality in defining and analysing issues or problems and providing solutions; and the use of means of communication appropriate to the discipline and the audience.

In the context of this unit, the project has a good design, and has used some data that is interesting or non-obvious, or has required some effort to obtain or use. It involves a good analysis of the data, and fairly extensively draws on the techniques and tools presented in the unit. The project is described in a report and a presentation that are well-structured and mostly free from errors; these would be of a standard that could be presented at a conference with some polishing.
Credit (Cr, 65-74): provides evidence of learning that goes beyond replication of content knowledge or skills relevant to the learning outcomes. There is demonstration of substantial understanding of fundamental concepts in the field of study and the ability to apply these concepts in a variety of contexts; convincing argumentation with appropriate coherent justification; communication of ideas fluently and clearly in terms of the conventions of the discipline.

In the context of this unit, the project has a sound design, and demonstrates some thought in the choice of data. It involves a good analysis of the data, and uses a reasonable number of the techniques and tools presented in the unit. The project is described in a report and a presentation that are well-structured and mostly free from errors.

Pass (P, 50-64): provides sufficient evidence of the achievement of learning outcomes. There is demonstration of understanding and application of fundamental concepts of the field of study; routine argumentation with acceptable justification; communication of information and ideas adequately in terms of the conventions of the discipline. The learning attainment is considered satisfactory or adequate or competent or capable in relation to the specified outcomes.

In the context of this unit, the project has a satisfactory design and uses some easily accessible data. It involves a successful, or nearly successful, analysis of data, and shows some familiarity with tools or techniques presented in the unit. The project is described in a satisfactory report and presentation.

Fail (F, 0-49): does not provide evidence of attainment of learning outcomes. There is missing or partial or superficial or faulty understanding and application of the fundamental concepts in the field of study; missing, undeveloped, inappropriate or confusing argumentation; incomplete, confusing or lacking communication of ideas in ways that give little attention to the conventions of the discipline.