COMP7000

Advanced Algorithms

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

School of Computing

Contents

General Information	2
Learning Outcomes	2
General Assessment Information	3
Assessment Tasks	6
Delivery and Resources	8
Policies and Procedures	8
Changes from Previous Offering	10

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, Lecturer (weeks 7-12)
Bernard Mans
bernard.mans@mq.edu.au

Lecturer (weeks 1-6)
Chris Chen
chris.chen1@mq.edu.au

Credit points
10

Prerequisites
Admission to MRes

Corequisites

Co-badged status

Unit description
Algorithms are the essence of computer science. In this unit we build on the undergraduate understanding of algorithms and look at interesting and useful algorithms, both fundamental and cutting edge. The particular material covered will depend on the cohort but may include topics such as approximation algorithms, exponential-time exact and parameterized algorithms, linear and constraint programming and fundamental graph algorithms such as max-flow algorithms, matching algorithms an so on. The unit will also employ appropriate tools from complexity theory to analyse the performance of the algorithms studied.

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: Explain key ideas in the field of algorithmics and the workings of key algorithms, and compare and evaluate algorithmic solutions for computational problems.

ULO2: Formally analyse algorithms.

ULO3: Implement key algorithms.

ULO4: Develop algorithmic solutions for computational problems by constructing new

algorithms and combining existing algorithms.

ULO5: Investigate topics in advanced algorithms and synthesise the output for presentation in oral and written form.

General Assessment Information

COMP7000 will be assessed and graded according to the University assessment and grading policies.

Submission Deadlines

Late Assessment Submission Penalty

Standard late penalties apply.

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 7thday (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.

Assessments where Late Submissions will be accepted: allSpecial 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.

Standards

The following general standards of achievement will be used to assess each of the assessment tasks with respect to the letter grades.

Pass: Has a basic understanding of the algorithms and concepts as discussed in class. Can describe and reproduce definitions and fundamental algorithms. Can perform a basic research investigation in the area and present the results of that research in rudimentary written and oral forms.

Credit: As for Pass plus: Is able to apply the algorithmic techniques we have discussed to derive solutions to computational problems. Can develop, generalise and apply the concepts discussed in class to address basic theoretical and practical questions, and can effectively communicate these insights. Shows more than basic insights into the results of a research investigation and is able to communicate those insights.

Distinction/High Distinction: As for Credit plus: Is able to generalise and synthesise knowledge to

address more complex topics beyond the material discussed in class. Can critically evaluate the limits of the techniques and algorithms discussed.

Assessment Process

These assessment standards will be used to give a numeric mark out of 100 to each assessment submission during marking. The mark will correspond to a letter grade for that task according to the University guidelines. The final raw mark for the unit will be calculated by combining the marks for all assessment tasks according to the percentage weightings shown in the assessment summary.

Requirements to Pass this Unit

To pass this unit you must:

- Achieve a total mark equal to or greater than 50%.

Projects

Assessment Type [1]: Project Indicative Time on Task [2]: 40 hours Due: **Weeks 1 -- 12** Weighting: **54%**

Students will be asked to complete 4 projects. These will consist of a combination of programming, program analysis and report writing.

On successful completion you will be able to:

- Explain key ideas in the field of algorithmics and the workings of key algorithms, and compare and evaluate algorithmic solutions for computational problems.
- Formally analyse algorithms.
- Implement key algorithms.
- Develop algorithmic solutions for computational problems by constructing new algorithms and combining existing algorithms.

Presentation

Assessment Type [1]: Presentation Indicative Time on Task [2]: 10 hours Due: **Week 12-13** Weighting: **10%**

An oral presentation supported by appropriate presentation materials.

On successful completion you will be able to:

- Explain key ideas in the field of algorithmics and the workings of key algorithms, and compare and evaluate algorithmic solutions for computational problems.
- Investigate topics in advanced algorithms and synthesise the output for presentation in oral and written form.

Weekly tasks

Assessment Type [1]: Problem set Indicative Time on Task [2]: 0 hours Due: **Weeks 1--12** Weighting: **36%**

Each week students will be asked to complete some exercises to test their understanding of the material.

On successful completion you will be able to:

- Explain key ideas in the field of algorithmics and the workings of key algorithms, and compare and evaluate algorithmic solutions for computational problems.
- Formally analyse algorithms.
- Implement key algorithms.
- Develop algorithmic solutions for computational problems by constructing new algorithms and combining existing algorithms.
- Investigate topics in advanced algorithms and synthesise the output for presentation in oral and written form.

[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

Assessment Tasks

Name	Weighting	Hurdle	Due
Weekly tasks	36%	No	Weeks 1 -- 12
Presentation	10%	No	Week 12-13
Projects	54%	No	Weeks 1--12

Assessment Tasks

Name	Weighting	Hurdle	Due	Groupwork/Individual	Short Extension	AI Approach
Projects	54%	No	Weeks 1--12		No	
Presentation	10%	No	Week 12-13		No	
Weekly tasks	36%	No	Weeks 1--12		No	

Projects

Assessment Type [1]: Project
Indicative Time on Task [2]: 40 hours
Due: **Weeks 1--12**
Weighting: **54%**
Groupwork/Individual:
Short extension [3]: No
AI Approach:

Students will be asked to complete 4 projects. These will consist of a combination of programming, program analysis and report writing.

On successful completion you will be able to:

- Explain key ideas in the field of algorithmics and the workings of key algorithms, and compare and evaluate algorithmic solutions for computational problems.
- Formally analyse algorithms.
- Implement key algorithms.
- Develop algorithmic solutions for computational problems by constructing new algorithms and combining existing algorithms.

Presentation

Assessment Type [1]: Presentation
Indicative Time on Task [2]: 10 hours
Due: **Week 12-13**
Weighting: **10%**
Groupwork/Individual:
Short extension [3]: No

AI Approach:

An oral presentation supported by appropriate presentation materials.

On successful completion you will be able to:

- Explain key ideas in the field of algorithmics and the workings of key algorithms, and compare and evaluate algorithmic solutions for computational problems.
- Investigate topics in advanced algorithms and synthesise the output for presentation in oral and written form.

Weekly tasks

Assessment Type [1]: Problem set
Indicative Time on Task [2]: 0 hours
Due: **Weeks 1--12**
Weighting: **36%**
Groupwork/Individual:
Short extension [3]: No
AI Approach:

Each week students will be asked to complete some exercises to test their understanding of the material.

On successful completion you will be able to:

- Explain key ideas in the field of algorithmics and the workings of key algorithms, and compare and evaluate algorithmic solutions for computational problems.
- Formally analyse algorithms.
- Implement key algorithms.
- Develop algorithmic solutions for computational problems by constructing new algorithms and combining existing algorithms.
- Investigate topics in advanced algorithms and synthesise the output for presentation in oral and written form.

---

[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

Classes

Each week has two hours of face-to-face class. These classes will be a mixture lecture material, discussion and in class tests.

Recommended Reading and References

There is no set text for the course, but the following *far from exhaustive* list of texts may be useful for reference, study and further reading:

- Skiena, Algorithm Design Manual, Spinger.
- Cormen, Leiserson, Rivest and Stein. Introductions to algorithms, Prentice Hall.
- Papadimitriou, Computational Complexity, Addison Wesley.
- Sipser, Introduction to the Theory of Computation, Thomson.

Unit Webpage, Materials and Technologies Used

The materials for the unit including notes, discussion fora, electronic submission links etc. will be through the iLearn system.

The programming projects can be done in any programming language subject to prior approval of the course convener. Languages can include Java and Python.

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 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.

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 the Service Connect Portal, 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

Student feedback from the previous offering of this unit was very positive overall, with students pleased with the clarity around assessment requirements and the level of support from the teaching staff. As such, no change to the delivery of the unit is planned, however, we will continue to strive to improve the level of support and the level of student engagement.

Unit information based on version 2024.01R of the Handbook