



COMP6291

Operating Systems

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

School of Computing

Contents

<u>General Information</u>	2
<u>Learning Outcomes</u>	2
<u>General Assessment Information</u>	3
<u>Assessment Tasks</u>	4
<u>Delivery and Resources</u>	6
<u>Unit Schedule</u>	8
<u>Policies and Procedures</u>	9
<u>Changes from Previous Offering</u>	11

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
Credit points 10
Prerequisites COMP6100
Corequisites
Co-badged status
Unit description This unit will introduce the concept of an operating system and describe its software architecture and interaction with modern computer hardware. The unit will cover topics on resource management of the central processing unit (CPU), memory, storage, network communication, and Input/Output (I/O) devices. Students will learn concepts such as multitasking, processes, address spaces, isolation, scheduling, concurrency, fairness, multithreading, synchronization, deadlock, virtual memory, interrupts, computer architecture, signals, kernels, user-space, file systems, the layered network stack, security, and virtualisation. The performance tradeoffs in the design of various components of the operating system will be discussed. Learning in this unit enhances student understanding of global challenges identified by the United Nations Sustainable Development Goals (UNSDGs) Industry, Innovation and Infrastructure

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:** Demonstrate an ability to apply resource management concepts and principles to application design and execution (CPU, memory, storage/files, network/I/O, processes).
- ULO2:** Develop software programs that support concurrency and multi-threading.
- ULO3:** Develop a simple device driver that demonstrates the connection between operating systems and hardware.

ULO4: Explain the structure/architecture and function of an Operating System.

ULO5: Demonstrate an understanding and ramifications of security and fault isolation.

ULO6: Demonstrate an understanding of virtualisation concepts.

General Assessment Information

Assessments

The assessment is divided into three major components, namely a final exam and two coding assignments. The weightings of each component are designated above. The Final exam will be held in rooms designated by the university during the exam period. Exams will be on iLearn. Questions will likely include short answer, fill in the blank, and multiple choice. More details will be posted on iLearn later in the term.

The coding assignment will be divided into two major subcomponents or labs, with lab 1 due mid semester (week 7) and lab 2 due at the end of the term (week 13). Each lab will have a code contribution component. Each lab will also have a viva component as part of the assessment where you will be asked to explain your code. See iLearn for more details. Lab assignments will primarily use the C programming language, but may include other components involving Makefiles, scripting, etc.

The use of AI in assessments is not allowed. Generative AI and LLMs such as ChatGPT cannot be used to produce code for assignments.

Late Assessment Submission

Late assessments are not accepted in this unit unless a [Special Consideration](#) for an extension has been submitted and approved. Short extensions are not supported.

Requirements to Pass this Unit

Whilst there are a number of learning activities and assessments that make up the unit, in order to pass the unit the only requirement is that you achieve a total mark equal to or greater than 50%.

There are no hurdle requirements for the unit.

Assessment Tasks

Name	Weighting		Due
Final exam	50%		Exam Period
Assignment 1	25%		Week 7
Assignment 2	25%		Week 13

Final exam

Assessment Type [1](#): Examination Indicative Time on Task [2](#): 40 hours. Due: Exam Period. Weighting: 50%.

An examination allows us to individually and securely assess student's mastery of the coursework material.

On successful completion you will be able to:

- Demonstrate an ability to apply resource management concepts and principles to application design and execution (CPU, memory, storage/files, network/I/O, processes).
- Explain the structure/architecture and function of an Operating System.
- Demonstrate an understanding and ramifications of security and fault isolation.
- Demonstrate an understanding of virtualisation concepts.

Assignments 1 and 2

Assessment Type ¹: Practice-based task Indicative Time on Task ²: 29 hours each. Due: Week 7 and 13. Weighting total: 50%. Release dates: Assignment 1 released by Week 2 Friday 11:55 pm, Assignment 2 released by week 7 Friday 11:55 pm

A semester-long programming task where students put all their skills to work creating operating system components or modules.

On successful completion you will be able to:

- Demonstrate an ability to apply resource management concepts and principles to application design and execution (CPU, memory, storage/files, network/I/O, processes).
- Develop software programs that support concurrency and multi-threading.
- Develop a simple device driver that demonstrates the connection between operating systems and hardware.
- Demonstrate an understanding and ramifications of security and fault isolation.
- Demonstrate an understanding of virtualisation concepts.

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	Groupwork/Individual	Short Extension	AI Approach
Foundations Lab Task	25%	No	Week 7	Individual	No	Observed

Name	Weighting	Hurdle	Due	Groupwork/Individual	Short Extension	AI Approach
Advanced Lab Task	25%	No	Week 13	Individual	No	Observed
Final exam	50%	No	Exam Period	Individual	No	Observed

Foundations Lab Task

Assessment Type [1](#): Practice-based task

Indicative Time on Task [2](#): 29 hours

Due: **Week 7**

Weighting: **25%**

Groupwork/Individual: Individual

Short extension [3](#): No

AI Approach: Observed

A programming task where students put all their skills to work creating operating system components or modules based on principles from the first half of the unit.

On successful completion you will be able to:

- Demonstrate an ability to apply resource management concepts and principles to application design and execution (CPU, memory, storage/files, network/I/O, processes).
- Develop software programs that support concurrency and multi-threading.
- Develop a simple device driver that demonstrates the connection between operating systems and hardware.
- Demonstrate an understanding and ramifications of security and fault isolation.
- Demonstrate an understanding of virtualisation concepts.

Advanced Lab Task

Assessment Type [1](#): Practice-based task

Indicative Time on Task [2](#): 29 hours

Due: **Week 13**

Weighting: **25%**

Groupwork/Individual: Individual

Short extension [3](#): No

AI Approach: Observed

A programming task where students put all their skills to work creating operating system components or modules based on principles from the second half of the unit.

On successful completion you will be able to:

- Demonstrate an ability to apply resource management concepts and principles to application design and execution (CPU, memory, storage/files, network/I/O, processes).
- Develop software programs that support concurrency and multi-threading.
- Develop a simple device driver that demonstrates the connection between operating systems and hardware.
- Demonstrate an understanding and ramifications of security and fault isolation.
- Demonstrate an understanding of virtualisation concepts.

Final exam

Assessment Type ¹: Examination

Indicative Time on Task ²: 40 hours

Due: **Exam Period**

Weighting: **50%**

Groupwork/Individual: Individual

Short extension ³: No

AI Approach: Observed

You will undertake a final examination during the formal examination period.

On successful completion you will be able to:

- Demonstrate an ability to apply resource management concepts and principles to application design and execution (CPU, memory, storage/files, network/I/O, processes).
- Explain the structure/architecture and function of an Operating System.
- Demonstrate an understanding and ramifications of security and fault isolation.
- Demonstrate an understanding of virtualisation concepts.

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 you should attend two hours of lectures, and a two hour practical workshop. For

details of scheduled classes consult publish.mq.edu.au.

Note that practicals workshops (lab sessions) commence in week 1. The week-by-week details of the practical (lab) classes will be available from iLearn.

You must attend the practical that you are enrolled in.

Textbook and Reading Materials

Required Text Book

"Operating Systems Concepts" 10th edition, by Silberschatz, Gagne and Galvin. 2021. We will assign weekly readings from the textbook during lecture and/or on iLearn.

The text book is available in electronic form online, or you can purchase a printed copy from a book seller of your choice. It may be available in the Library.

Recommended Text

"The C Programming Language" 2nd edition, Brian W Kernighan and Dennis M Ritchie, Prentice-Hall 1988.

This small book is the classic reference on C programming.

Unit Websites

COMP2291 / COMP6291 is administered via [iLearn \(http://ilearn.mq.edu.au/\)](http://ilearn.mq.edu.au/).

This unit outline can be found in the university's [unit guides](#)

We will use iLearn as our main platform for making announcements, posting assignments, releasing lecture slides and other content, submitting assignments, conducting exams, and posting results. We will use the announcements forums on iLearn to post important messages to the class. Students may ask questions in a separate general iLearn forum, where we will answer them.

Lecture Recordings

Digital recordings of lectures may be available. They will be linked from iLearn.

Technologies Used and Required

COMP2291 / COMP6291 is a BYOD (Bring Your Own Device) unit. You will be expected to bring your own laptop computer (Windows, Mac, or Linux) to the workshop, install and configure the required software, and incorporate secure practices into your daily work (and play!) routines.

General Notes

In this unit, you should do the following:

- Review recorded lecture materials.
- Attend your weekly Practical session.
- Attend lectures, take notes, ask questions.

- Work on any assignments that have been released.

Note that Workshops commence in week 1.

Students will attend weekly workshops in person, where they will receive additional explanations on class assignments and topics, and may be asked to explain their code in vivas, etc. Students should use these workshops to ask questions of their tutor.

Communication Methods in COMP2291 / COMP6291

All announcements about unit-related matters will be communicated through iLearn. It is the student's responsibility to ensure they check iLearn announcements, forums, and other sections regularly.

Students are encouraged to use the iLearn forums for asking questions about unit content and concepts. Where questions are about specific details in an assessment submission.

Emails to the convenor, lecturer and/or tutors need to be preceded with a title identifying the class: COMP 2291 or 6291. Any one-on-one communication with unit staff that is via email must be done through the student's official university email account (the one ending with '[@student.s.mq.edu.au](mailto:@students.mq.edu.au)'). There may be occasions where unit staff will email a student directly to their @students.mq.edu.au email address. It is the student's responsibility to ensure they check their official university email regularly for communications from the university staff.

Students may contact the convenor at the posted email above to have questions that cannot be otherwise answered by tutors or in the general forum, such as questions about policy in the class, but should avail themselves first of the general forum and tutors at their workshops to have technical questions about their assignments answered.

Results will be released on iLearn in a timely manner, subject to certain acceptable delays such as accommodating special consideration late assessments.

Drop-In Centre

There may be support at the Drop-In Centre in 4RPD. Please contact them for more details.

Unit Schedule

The tentative weekly schedule is as follows, and may be subject to change. See iLearn for more details.

Week 1: Introduction to C and OS

Weeks 2-4: Processes, threads, synchronization and deadlock

Weeks 5-6: Virtual memory

Week 7: Input/Output

Weeks 8-9: Storage and File Systems

Week 10: CPU Scheduling

Week 11: Networking

Week 12: Security

Week 13: Virtual Machines

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](#).

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

There were no substantial changes from the previous offering.

Unit information based on version 2026.04 of the [Handbook](#)