UNSW Course Outline

COMP3231 Operating Systems - 2024

Published on the 28 Jan 2024

General Course Information

Course Code : COMP3231
Year : 2024
Term : Term 1
Teaching Period : T1
Is a multi-term course? : No
Faculty : Faculty of Engineering
Academic Unit : School of Computer Science and Engineering
Delivery Mode : Multimodal
Delivery Format : Standard
Delivery Location : Kensington
Campus : Sydney
Study Level : Undergraduate
Units of Credit : 6

Useful Links

Handbook Class Timetable

Course Details & Outcomes

Course Description

Operating systems are an essential part of computer systems, so a course on operating systems is an essential part of any computer science or computer engineering program. This course provides an in-depth understanding of the underlying operating systems that students have implicitly relied upon when developing applications in the foundational courses within Computer Science and Engineering. The knowledge gained will continue to be relevant to your future career when developing systems and applications.

In general terms, the course aims to educate students in the basic concepts, components and behaviours of modern monolithic operating systems, including the relevant characteristics of hardware. Topics include processes, threads, concurrency, file systems, memory management and scheduling.

Students will apply some of the concepts learnt by implementing parts of a realistic teaching operating system.

Course Aims

The course aims to educate students on modern monolithic operating systems, i.e. the software that fills the gap between application software they will already be familiar with and computer hardware.

In particular, the course aims to provide practical experience within the operating system itself to give students a holistic view of the entire software stack running in a modern computer.

The development of a whole of system understanding of modern computer systems is a critical and valued skill enabling students to understand, analyse, and implement complex software systems at any level of the software stack, with a big picture perspective.

Course Learning Outcomes

Course Learning Outcomes
CLO1 : Understand the structure and key components within a modern monolithic operating system.
CLO2 : Apply common algorithms and implementation approaches that are used inside an operating system.
CLO3 : Analyse the behaviour of an operating system and its components.
CLO4 : Create new functionality within an existing operating system.

Course Learning Outcomes	Assessment Item
CLO1 : Understand the structure and key components within a modern monolithic operating system.	• ASST1 • ASST2 • ASST3 • Final Exam
CLO2 : Apply common algorithms and implementation approaches that are used inside an operating system.	• ASST1 • ASST2 • ASST3 • Final Exam
CLO3 : Analyse the behaviour of an operating system and its components.	• ASST1 • ASST2 • ASST3 • Final Exam
CLO4 : Create new functionality within an existing operating system.	• ASST1 • ASST2 • ASST3

Learning and Teaching Technologies

Moodle - Learning Management System | Blackboard Collaborate | EdStem | CSE Systems | Echo 360

Additional Course Information

Assumed Knowledge

It is assumed that an enrolled student is familiar with the organisation of a general-purpose computer (in particular, CPU, memory, bus, registers, machine instructions, interrupts/ exceptions).

Students are assumed to be competent programmers with the C programming language. More specifically, students should understand and easily apply pointers, function pointers, memory allocation (malloc()), and be comfortable navigating around a large existing code base. They should be confident implementing data structures and algorithms, and able to debug their implementation within an existing code base.

Students are assumed to be familiar with the git revision control system or capable of learning the basics quickly.

Assessments

Assessment Structure

Assessment Item	Weight	Relevant Dates
ASST1 Assessment Format: Individual	20%	Due Date: Week 4
ASST2 Assessment Format: Individual	20%	Due Date: Week 7
ASST3 Assessment Format: Individual	20%	Due Date: Week 10
Final Exam Assessment Format: Individual	40%	Due Date: To be scheduled during the normal exam period

Assessment Details

ASST1

Assessment Overview

This assignment has two aims: to familiarise students with the development tools and cycle of the teaching OS, and to familiarise students with the concurrent programming model within the OS.

Students will solve concurrent programming problems within the teaching OS itself. Upon completion, students will additionally provide a short recorded video overview of their work.

Students are marked using a combination of automated marking and manual marking via a rubric.

Students can expect to spend approximately 10 hours per week in the two weeks prior to submission.

Course Learning Outcomes

- CLO1 : Understand the structure and key components within a modern monolithic operating

system.
- CLO2 : Apply common algorithms and implementation approaches that are used inside an operating system.
- CLO3 : Analyse the behaviour of an operating system and its components.
- CLO4 : Create new functionality within an existing operating system.

Detailed Assessment Description

Details will be provided on the course website https://www.cse.unsw.edu.au/~cs3231/

Submission notes

Detailed submission notes will be provided on the course website.

ASST2

Assessment Overview

This assignment aims to improve students' understanding of the internal architecture of the teaching OS, the OS provision of system calls, and the system call's relationship to other components of the operating system.

Students will add new system calls to their operating system to extend its functionality. Upon completion, students will additionally provide a short recorded video overview of their work.

Students are marked using a combination of automated marking and manual marking via a rubric.

Students can expect to spend approximately 10 hours per week in the two weeks prior to submission.

Course Learning Outcomes

- CLO1 : Understand the structure and key components within a modern monolithic operating system.
- CLO2 : Apply common algorithms and implementation approaches that are used inside an operating system.
- CLO3 : Analyse the behaviour of an operating system and its components.
- CLO4 : Create new functionality within an existing operating system.

Detailed Assessment Description

Details will be provided on the course website https://www.cse.unsw.edu.au/~cs3231/

Submission notes

Details submission instructions will be provided on the course website.

ASST3

Assessment Overview

This assessment aims to impart a detailed understanding of virtual memory.

Students will implement parts of the virtual memory subsystem of the teaching OS to provide virtual memory management to applications running on their OS. Upon completion, students will additionally provide a short recorded video overview of their work.

Students are marked using a combination of automated marking and manual marking via a rubric.

Students can expect to spend approximately 10 hours per week in the two weeks prior to submission.

Course Learning Outcomes

- CLO1 : Understand the structure and key components within a modern monolithic operating system.
- CLO2 : Apply common algorithms and implementation approaches that are used inside an operating system.
- CLO3 : Analyse the behaviour of an operating system and its components.
- CLO4 : Create new functionality within an existing operating system.

Detailed Assessment Description

Details will be provided on the course website https://www.cse.unsw.edu.au/~cs3231/

Submission notes

Details submission instructions will be provided on the course website.

Final Exam

Assessment Overview

The final exam is in the form of a 2 hour, open book, online exam.

Course Learning Outcomes

- CLO1 : Understand the structure and key components within a modern monolithic operating system.
- CLO2 : Apply common algorithms and implementation approaches that are used inside an operating system.
- CLO3 : Analyse the behaviour of an operating system and its components.

Detailed Assessment Description

Precise details of how to undertake the final exam will be provided in week 10 of term.

Hurdle rules

A final exam mark of 40% is required to pass the course.

General Assessment Information

Grading Basis

Standard

Requirements to pass course

In addition to the Final Exam hurdle, students require a composite mark of 50 out of 100 to pass the course.

Course Schedule

Teaching Week/Module	Activity Type	Content
Week 1 : 12 February - 18 February	Lecture	• OS Overview • Processes and Threads • Concurrency and Synchronisation
Week 2 : 19 February - 25 February	Lecture	• Concurrency and Synchronisation • Deadlock
Week 3 : 26 February - 3 March	Lecture	• Processes and Thread Implementation • System Calls • R3000 Overview
Week 4 : 4 March - 10 March	Lecture	• Computer Hardware, Memory Hierarchy and Caching • File Management
Week 5 : 11 March - 17 March	Lecture	• File Management • The ext2 file system
Week 7 : 25 March - 31 March	Lecture	• The ext3 file system • Memory Management • Virtual Memory
Week 8 : 1 April - 7 April	Lecture	• Virtual Memory
Week 9 : 8 April - 14 April	Lecture	• Virtual Memory • Multiprocessors • Scheduling
Week 10 : 15 April - 21 April	Lecture	• Last Lecture and Course Wrap-up

Attendance Requirements

Students are strongly encouraged to attend all classes and review lecture recordings.

General Schedule Information

The provided schedule is indicative and subject ot change.

Course Resources

Recommended Resources

Textbook

- A. Tannenbaum and H. Bos, *Modern Operating Systems* , 4th ed.
  - Print: https://www.bookshop.unsw.edu.au/details.cgi?ITEMNO=9781292061429
  - Digital: https://unswbookshop.vitalsource.com/products/-v9781292061955

Reference Books

For Operating Systems:

- A. Silberschatz, P.B. Galvin and Greg Gagne, *Operating System Concepts*
- William Stallings, *Operating Systems: Internals and Design Principles*
- A. Tannenbaum, A. Woodhull, *Operating Systems--Design and Implementation*
- John O'Gorman, *Operating Systems* , MacMillan, 2000
- Uresh Vahalla, *UNIX Internals: The New Frontiers* , Prentice Hall, 1996

For the C language:

- B Kernighan and D. Ritchie, *The C Programming Language* , 2nd ed, Prentice Hall.
- S. Harbison and G. Steele, *C: A Reference Manual* , Prentice Hall.

Copies of lecture slides, manuals, and other information can be found under the course's WWW home page at URL https://www.cse.unsw.edu.au/~cs3231/

Course Evaluation and Development

This course is evaluated each session using the myExperience system **and occassionally my own surveys.**

The surveys, a detailed analysis, and developments as a result are documented on the course website https://www.cse.unsw.edu.au/~cs3231/surveys.php

Staff Details

Position	Name	Email	Location	Phone	Availability	Equitable Learning Services Contact	Primary Contact
Convenor	Kevin Elphinstone					Yes	No
	OS Admin Team					No	Yes

Other Useful Information

Academic Information

I. Special consideration and supplementary assessment

If you have experienced an illness or misadventure beyond your control that will interfere with your assessment performance, you are eligible to apply for Special Consideration prior to, or within 3 working days of, submitting an assessment or sitting an exam.

Please note that UNSW has a Fit to Sit rule, which means that if you sit an exam, you are declaring yourself fit enough to do so and cannot later apply for Special Consideration.

For details of applying for Special Consideration and conditions for the award of supplementary assessment, please see the information on UNSW's Special Consideration page.

II. Administrative matters and links

All students are expected to read and be familiar with UNSW guidelines and polices. In particular, students should be familiar with the following:

- Attendance
- UNSW Email Address
- Special Consideration
- Exams
- Approved Calculators
- Academic Honesty and Plagiarism
- Equitable Learning Services

III. Equity and diversity

Those students who have a disability that requires some adjustment in their teaching or learning environment are encouraged to discuss their study needs with the course convener prior to, or at the commencement of, their course, or with the Equity Officer (Disability) in the Equitable Learning Services. Issues to be discussed may include access to materials, signers or note-takers, the provision of services and additional exam and assessment arrangements. Early notification is essential to enable any necessary adjustments to be made.

IV. Professional Outcomes and Program Design

Students are able to review the relevant professional outcomes and program designs for their streams by going to the following link: https://www.unsw.edu.au/engineering/student-life/student-resources/program-design.

Note: This course outline sets out the description of classes at the date the Course Outline is published. The nature of classes may change during the Term after the Course Outline is published. Moodle or your primary learning management system (LMS) should be consulted for the up-to-date class descriptions. If there is any inconsistency in the description of activities between the University timetable and the Course Outline/Moodle/LMS, the description in the Course Outline/Moodle/LMS applies.

Academic Honesty and Plagarism

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. *Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.*

Plagiarism is a type of intellectual theft. It can take many forms, from deliberate cheating to accidentally copying from a source without acknowledgement. UNSW has produced a website with a wealth of resources to support students to understand and avoid plagiarism, visit: student.unsw.edu.au/plagiarism. The Learning Centre assists students with understanding academic integrity and how not to plagiarise. They also hold workshops and can help students one-on-one.

You are also reminded that careful time management is an important part of study and one of the identified causes of plagiarism is poor time management. Students should allow sufficient time for research, drafting and the proper referencing of sources in preparing all assessment tasks.

Repeated plagiarism (even in first year), plagiarism after first year, or serious instances, may also be investigated under the Student Misconduct Procedures. The penalties under the procedures can include a reduction in marks, failing a course or for the most serious matters (like plagiarism in an honours thesis or contract cheating) even suspension from the university. The Student Misconduct Procedures are available here:

www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf

Submission of Assessment Tasks

Work submitted late without an approved extension by the course coordinator or delegated authority is subject to a late penalty of five percent (5%) of the maximum mark possible for that assessment item, per calendar day.

The late penalty is applied per calendar day (including weekends and public holidays) that the assessment is overdue. There is no pro-rata of the late penalty for submissions made part way through a day. This is for all assessments where a penalty applies.

Work submitted after five days (120 hours) will not be accepted and a mark of zero will be awarded for that assessment item.

For some assessment items, a late penalty may not be appropriate. These will be clearly indicated in the course outline, and such assessments will receive a mark of zero if not completed by the specified date. Examples include:

- Weekly online tests or laboratory work worth a small proportion of the subject mark;
- Exams, peer feedback and team evaluation surveys;
- Online quizzes where answers are released to students on completion;
- Professional assessment tasks, where the intention is to create an authentic assessment that has an absolute submission date; and,
- Pass/Fail assessment tasks.

Faculty-specific Information

Engineering Student Support Services – The Nucleus - enrolment, progression checks, clash requests, course issues or program-related queries

Engineering Industrial Training – Industrial training questions

UNSW Study Abroad – study abroad student enquiries (for inbound students)

UNSW Exchange – student exchange enquiries (for inbound students)

UNSW Future Students – potential student enquiries e.g. admissions, fees, programs, credit transfer

Phone

(+61 2) 9385 8500 – Nucleus Student Hub

(+61 2) 9385 7661 – Engineering Industrial Training

(+61 2) 9385 3179 – UNSW Study Abroad and UNSW Exchange (for inbound students)

School Contact Information

CSE Help! - on the Ground Floor of K17

- For assistance with coursework assessments.

The Nucleus Student Hub - https://nucleus.unsw.edu.au/en/contact-us

- Course enrolment queries.

Grievance Officer - grievance-officer@cse.unsw.edu.au

- If the course convenor gives an inadequate response to a query or when the courses convenor does not respond to a query about assessment.

Student Reps - stureps@cse.unsw.edu.au

- If some aspect of a course needs urgent improvement. (e.g. Nobody responding to forum queries, cannot understand the lecturer)

You should **never** contact any of the following people directly:

- Vice Chancellor

- Pro-vice Chancellor Education (PVCE)

- Head of School

- CSE administrative staff

- CSE teaching support staff

They will simply bounce the email to one of the above, thereby creating an unnecessary level of indirection and a delay in the response.