The Hong Kong University of Science and Technology

UG Course Syllabus (Spring 2025-26)

[Course Title] Operating Systems

[Course Code] COMP3511

[No. of Credits] 3 credits

[Any pre-/co-requisites] COMP 2611 OR [(ELEC 2350) AND (COMP 2011 OR COMP 2012H)]

Name: Prof. Bo LI (L1), Prof. Kai CHEN (L2)

Email: bli@cse.ust.hk , kaichen@cse.ust.hk

Course Description

This is an introductory course on operating systems. The topics will include the basic concepts of operating systems, process and threads, inter-process communications, process synchronization, scheduling, memory allocation, page and segmentation, secondary storage, I/O systems, file systems, security and protection. It contains the key concepts as well as examples drawn from a variety of real systems such as Microsoft Windows and Linux.

List of Topics

Chapter 1: Introduction
Chapter 2: Operating System Structures
Chapter 3: Processes
Chapter 4: Threads and Concurrency
Chapter 5: CPU Scheduling
Chapter 6: Synchronization tools
Chapter 7: Synchronization examples
Chapter 8: Deadlocks
Chapter 9: Memory Management Strategies
Chapter 10: Virtual-Memory Management
Chapter 11: Mass Storage Systems
Chapter 12: I/O Systems
Chapter 13: File-System Interface
Chapter 14: File-System Implementation
Chapter 17: Protection

Intended Learning Outcomes (ILOs)

By the end of this course, students should be able to:

1. Describe what is an operating system and the role it plays. Recognize different types of operating systems and know the basic architecture of an operating system.

2. Understand the concepts related to processes and threads, including their creation, communication and scheduling. Identify and address deadlock, and can design algorithmic solutions to synchronization problems.

3. Explain how main memory and virtual memory are managed. Master algorithms involved in memory allocation and page replacement.

4. Know about the concepts and implementations of file systems. Comprehend the technologies like disk operations and RAID in mass-storage systems.

Assessment and Grading

This course will be assessed using criterion-referencing and grades will not be assigned using a curve. Detailed rubrics for each assignment are provided below, outlining the criteria used for evaluation.

Assessments:

Assessment Task	Contribution to Overall Course grade (%)	Due date
Mid-Term	20%	Around Week 8-9
Final examination	30%	During the University Exam Period
Homework1-4	20% (5% each)	HW1: Week 2-4 HW2: Week 5-7 HW3: Week 8-10 HW4: Week 11-13
Project 1-3	30% (10% each)	Project1: Week4-6 Project2: Week7-9 Project3: Week10-12

* Assessment marks for individual assessed tasks will be released within two weeks of the due date.

Mapping of Course ILOs to Assessment Tasks

Assessed Task	Mapped ILOs	Explanation
Midterm and Final Exam Homework1-4	ILO1, ILO2, ILO3. ILO4	The written assessments (i.e., midterm, final, homework) provides a holistic assessment to all the intended learning outcomes listed above.
Project1-3	ILO2. ILO3	The programming assignments provides hands-on practice focusing on operating system concepts, especially in the areas of ILO2 and ILO3

Grading Rubrics

Rubrics

Course Learning Outcome	Exemplary (A- to A+)	Competent (C to B+)	Needs Work (D to C-)	Unsatisfied (F)
Describe what is an operating system and the role it plays. Recognize different types of operating systems and know the basic architecture of an operating system.	Demonstrates thorough understanding of operating system and its roles. Is able to accurately identify various operating systems. Can design a simple operating system independently.	Demonstrates sufficient understanding of operating system and its roles. Is able to identify multiple operating systems. Can design a simple operating system with instructions.	Demonstrates some preliminary understanding of operating system and its roles. Is able to identify only several operating systems. Only can design a simple operating system partially.	Demonstrates deficient understanding of operating system and its roles. Is not able to distinguish between different operating systems. Not able to design a simple operating system.
Understand the concepts related to processes and threads, including their creation, communication and scheduling. Identify and address deadlock, and can design algorithmic solutions to synchronization problems.	Demonstrates thorough understanding of processes and threads. Can accurately identify and address complex deadlock problems, and can design algorithmic solutions to sophisticated synchronization problems.	Demonstrates sufficient understanding of processes and threads. Can identify and address most deadlock problems, and can design algorithmic solutions to most synchronization problems.	Demonstrates basic understanding of processes and threads. Can identify and address some deadlock problems, and can design algorithmic solutions to some synchronization problems.	Demonstrates deficient knowledge of processes and threads. Can not identify and address deadlock problems, and can hardly design algorithmic solutions to any synchronization problems.
Explain how main memory and virtual memory are managed. Master algorithms involved in memory allocation and page replacement.	Demonstrates ability to fully explain how main memory and virtual memory are managed. Masters all the algorithms involved in memory allocation and page replacement.	Demonstrates sufficient ability to explain how main memory and virtual memory are managed. Masters most algorithms involved in memory allocation and page replacement.	Demonstrates preliminary understanding of how main memory and virtual memory are managed. Knows some algorithms involved in memory allocation and page replacement.	Not understand how main memory and virtual memory are managed. Knows few algorithms involved in memory allocation and page replacement.
Know about the concepts and implementations of file systems. Comprehend the technologies like disk operations and RAID in mass-storage systems.	Demonstrates strong understanding of the concepts and implementations of file systems. Deeply comprehends the technologies in mass-storage systems.	Demonstrates sufficient understanding of the concepts and implementations of file systems. Comprehends most of the technologies in mass-storage systems.	Demonstrates preliminary understanding of the concepts and implementations of file systems. Comprehends part of the technologies in mass-storage systems.	Has very limited understanding of the influence of the usage of the concepts and implementations of file systems. Knows little about the technologies in mass-storage systems.

Final Grade Descriptors:

Grades	Short Description	Elaboration on subject grading description
A	Excellent Performance	Demonstrates a comprehensive grasp of subject matter, expertise in problem-solving, and significant creativity in thinking. Exhibits a high capacity for scholarship and collaboration, going beyond core requirements to achieve learning goals.
B	Good Performance	Shows good knowledge and understanding of the main subject matter, competence in problem-solving, and the ability to analyze and evaluate issues. Displays high motivation to learn and the ability to work effectively with others.
C	Satisfactory Performance	Possesses adequate knowledge of core subject matter, competence in dealing with familiar problems, and some capacity for analysis and critical thinking. Shows persistence and effort to achieve broadly defined learning goals.
D	Marginal Pass	Has threshold knowledge of core subject matter, potential to achieve key professional skills, and the ability to make basic judgments. Benefits from the course and has the potential to develop in the discipline.
F	Fail	Demonstrates insufficient understanding of the subject matter and lacks the necessary problem-solving skills. Shows limited ability to think critically or analytically and exhibits minimal effort towards achieving learning goals. Does not meet the threshold requirements for professional practice or development in the discipline.

Course AI Policy

We do not explicit state that students cannot use GenAI for the homework. On the other hand, we clearly state that we will use cheating detection software for some programming assignments.

For written exams, no electronic devices except a simple calculation is allowed. It means that we implicitly prohibit the usage of GenAI on the written exams.

Communication and Feedback

Assessment marks for individual assessed tasks will be communicated via Canvas within two weeks of submission. Feedback on assignments will include specific details, e.g., strengths, areas for improvement. Students who have further questions about the feedback including marks should consult the instructor within five working days after the feedback is received.

Resubmission Policy

If a student misses a deadline, they have at most 1 day late to submit their work with a 10% penalty. It applies on all homework and project assignments. Other than that, we don't allow students to resubmit their work after the deadlines.

Required Texts and Materials

Textbooks

Operating System Concepts, 10th Edition
Abraham Silberschatz, Peter B. Galvin, Greg Gagne
John Wiley & Sons Ltd, April 2018
ISBN: 978-1-118-09375-7

Academic Integrity

Students are expected to adhere to the university's academic integrity policy. Students are expected to uphold HKUST's Academic Honor Code and to maintain the highest standards of academic integrity. The University has zero tolerance of academic misconduct. Please refer to Academic Integrity | HKUST – Academic Registry for the University's definition of plagiarism and ways to avoid cheating and plagiarism.

Additional Resources

Reference books

Operating Systems: Three Easy Pieces
Remzi Arpaci-Dusseau & Andrea Arpaci-Dusseau
ISBN: 978-1-985-08659-3