ABET Computing Accreditation Commission

Faculty of Computing and Information Technology

Department of Information Systems

Spring 2018

E-COLLEGE AIMS

CPIS-222 Syllabus

Catalog Description

CPIS-222 Principles of Operating Systems
Credit: 3 (Theory: 3, Lab: 1, Practical: 1)
Prerequisite: CPCS-204 , CPIS-210
Classification: Department Required

The objective of this course is to present the basic concepts, modules and algorithms that work as intermediary programs between the user and the hardware, known as operating systems. It covers the basic concepts of recent operating systems, how they are designed and the way they work in terms of their efficiency and reliability. Also, it compares the techniques used inside the operating systems in terms of their speed and use of space. Topics include an overview of operating systems, operating system principles, CPU scheduling and dispatch, concurrency, memory management, and virtual memory.

Class Schedule

Meet 50 minutes 3 times/week or 80 minutes 2 times/week

Lab/Tutorial 90 minutes 1 times/week

Textbook

Abraham Silberschatz, Peter B. Galvin, Greg Gagne, , "Operating System Concepts", John Wiley & Sons Incorporated; 7 edition (2005-01)

ISBN-13 9780471694663 **ISBN-10** 0471694665

Grade Distribution

Week	Assessment	Grade %
3	Homework Assignments 1	3
5	Homework Assignments 2	3
6	Homework Assignments 3	3
8	Graded Lab Work 1	5
8	Exam 1	10
10	Graded Lab Work 2	5
10	Homework Assignments 4	3
12	Homework Assignments 5	3
13	Exam 2	10
14	Lab Exam	10
15	Project (Individual)	15
16	Exam	30

Last Articulated

December 18, 2017

Relationship to Student Outcomes

a	b	c	d	e	f	g	h	i	j
x	x							x	

Course Learning Outcomes (CLO)

By completion of the course the students should be able to

1. Explain the objectives and functions of a modern operating system (a)
2. Compare and contrast between operating system types (such as multiprogramming, timesharing and real time systems), and operating system structures (such as object-oriented, modular, micro-kernel, layered and virtual systems) (b)
3. Explain how a modern computer works: operating system control of the computer hardware and software such as interrupts, timer and dual mode, the different techniques used for doing input/output (a)
4. Compare the various architectures of multiple processor systems such as SMP, ASMP, MPP, Clustered and Grid computing. (i)
5. **Describe the various types of operating system services and how they are used either by parameterized commands through command line interpreters or by application software through system calls and APIs (i)**
6. Detect the different states that a task may pass through and the data structures and operations (such as context switching and dispatching) needed, given the source code of a simple task working in multiprogramming or timesharing environment. (b)
7. Discuss the types of processor scheduling such as short-term, medium-term, and long-term (a)
8. Compare between the different inter process communication models in terms of speed, convenience and implementation (i)
9. Describe the difference between processes and threads (i)
10. **Apply schedulling algorithms (FCFS, SJF, priority, Round Robin, Multilevel queue and multilevel feedback queue) and calculate their evaluation criteria (turnaround time,waiting time,...) for a given a list of processes and their arrival and burst time. (a)**
11. **Apply memory allocation algorithms (First fit, best fit, or Least recent used LRU,..) to a sequence of coming processes (or memory references), under different restrictions and memory arrangements (including virtual memory), to calculate evaluation criteria (percentage of memory usage, Number of page faults,..etc.). (a)**

CPIS-222 Syllabus

Topics Coverage Durations

Topics	Weeks
Operating System Overview	1
Computer System Organization Overview	1
Computer Architecture Overview	1
Operating-System Operation, Services and structures	1
Processes and Schedulers	1
InterProcess Communication and Threads	1
CPU Scheduling	2
Memory Management	1
Virtual Memory	1
Process Synchronization	2
Deadlock	1
Protection	1

Course Learning Outcomes (CLO)

By completion of the course the students should be able to

12. Summarize the potential run-time problems arising from concurrency (b)
13. **Analyze pseudocodes using synchronization techniques (spin locks, semaphores, monitors and/or condition variables) to support mutual exclusion in a critical section problem (such as bounded buffer, readers-writer, dining philosophers) or synchronization between two cooperating processes. (i)**
14. **Model the system state using the resource allocation graph (or current state table) to detect (or avoid) deadlock state and propose different techniques to prevent deadlock. (i)**
15. Construct the static/dynamic access matrix of a system to protect system resources. (i)

Coordinator(s)

Prof. Gibrael Elamin Abosamra, Professor