CMPE 242 Operating Systems

Department: Computer Engineering

Program Name: Computer Engineering | **Program Code:** 25

Course Number: CMPE242 | **Credits:** 4 Cr | **Year/Semester:**2019-2020 SPRING

☒ Required Course □ Elective Course (click on and check the appropriate box)

Prerequisite(s): CMPE112

Catalog Description:
Operating system definition, simple batch systems, multiprogramming, time-sharing, personal computer systems, parallel systems. introduction to process, process scheduling, operations on processes, cooperating processes, interprocess communications, interrupts, threads, process synchronization, critical-section problem, atomic instructions, semaphores, synchronization problems, CPU scheduling, scheduling criteria and algorithms, multiple processes and real-time scheduling, algorithm evaluation, deadlocks, characterization and handling of deadlocks, deadlock prevention avoidance and detection, deadlock recovery, memory management and virtual memory, address spaces, swapping, memory allocation, paging, segmentation.

Course Web Page: https://staff.emu.edu.tr/gurcuoz/en/teaching/cmpe242

Textbook(s):
Abraham Silberschatz, Peter Baer Galvin and Greg Gagne: Operating System Concepts, 9th edition, John Wiley & Sons, Inc., 2014. ISBN: 978-1-118-09375-7.

Indicative Basic Reading List :

1. Andrew S. Tanenbaum, Modern Operating Systems, Second Edition, Prentice Hall, 2001.
2. H. M. Deitel, P. J. Deitel, and D. R. Choffnes: Operating Systems, 3rd Edition, Pearson Education, 2004. ISBN: 0-13-124696-8.

Topics Covered and Class Schedule:
(4 hours of lectures per week)

Week 1	Operating system definition, simple batch systems, multiprogramming, time-sharing, personal computer systems, parallel systems. (SIL: ch1)
Week 2	Computer system structures, interrupts. Operating system structures, system calls. (SIL:ch1, ch2)
Week 3	Introduction to process, process scheduling, operations on processes, cooperating processes, interprocess communication (SIL: ch3), threads (SIL: ch4)
Weeks 4-6	Process synchronization; Critical-section problem, synchronizing hardware, semaphores, synchronization problems, critical regions, process monitors, conditional variables. (SIL:ch5)
Week 7	CPU scheduling; Basic concepts. Criteria and algorithms, multiple process and real-time scheduling, algorithm evaluation. (SIL:ch6)
Weeks 8-9	**Midterm Examination**
Week 10	CPU scheduling (Continue)
Weeks 11-12	Deadlocks; Characterization and handling of deadlocks, deadlock prevention avoidance and detection, deadlock recovery. (SIL:ch7)
Weeks 13-14	Memory management and virtual memory; Address spaces, swapping, contiguous allocation, paging, segmentation. (SIL:ch8, ch9)
Weeks 15-17	**Final Examination**

Laboratory Schedule:
(2 hours of laboratory per week)

Week 4	Lab 1: Interfacing with UNIX and understanding the important Shell commands
Week 5	Lab 2: Process Synchronization
Week 6	Lab 3: Process Synchronization (Cont.)
Week 7	Lab 4: CPU Scheduling Algorithms
Week 11	Lab 5: CPU Scheduling Algorithms (Cont.)
Week 13	Lab 6: Deadlocks
Week 14	Lab 7: Memory Management

Course Learning Outcomes:

At the end of the course, student must be able to

1. Understand the role and purpose of operating systems.
2. Understand the concept of how programming languages, operating systems, and hardware architectures interact.
3. Understand the concept of a process and concurrency problems: synchronization, mutual exclusion, deadlocks.
4. Describe concurrent execution using states and state diagrams, ready lists, process control blocks, dispatching and context switching, interrupt handling in a concurrent environment.
5. Identify scheduling policies (e.g. the issues involved with preemptive vs. nonpreemptive scheduling).
6. Know methods of deadlock avoidance, detection, prevention and recovery. Identify solution strategies, including semaphores, monitors and condition variables.
7. Describe physical memory and memory management, including overlays, swapping, partitions, paging and segmentation, page placement and replacement policies, working sets and caching.

Assessment	**Method**	**No**	**Percentage**
	Midterm Exam(s)	1	43%
	Lab Work(s)	5	7 %
	Attendance(s)	1	-
	Final Examination	1	50%

Contribution of Course to Criterion 5

Credit Hours for:
Mathematics & Basic Science : 0
Engineering Sciences and Design : 4
General Education : 0

Relationship of Course to Program Outcomes

The course has been designed to contribute to the following program outcomes:

1) an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics

2) an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors

Lab Policy: There are no exemptions from labs.

Attendance and Participation: Attendance to every lecture is mandatory.

Make-Up Policy: Only one comprehensive make-up examination will be given to those who miss any of the exams. The make-up exam will be given to only those who provide a **medical report** (doctor's report approved by the Student Health Center) within 3 days after the examination. Students, who miss both midterm and final exams, will get **NG** grade.

Prepared by: Gürcü Öz, Hakan Altınçay	**Date Prepared:** 16 February 2020