



National Computing Education Accreditation Council NCEAC



NCEAC.DOC.001-E

COURSE DESCRIPTION

NAME OF INSTITUTION	Lahore Garrison University
PROGRAM (S) TO BE EVALUATED	Computer Science , Fall 2022

Course Description : This course covers the introduction to operating systems, which serve as an interface between computer hardware and the user. The operating system is responsible for managing and coordinating processes and sharing limited resources of the computer. Students will be exposed to the techniques and algorithms for designing an operating system. Topics covered include process management, concurrency and synchronization, deadlock, memory management, file management, secondary storage management, and I/O management. At the end of the course, the student shall clearly understand the general concepts that underlie an operating system.

Course Code	CSC351																		
Course Title	Operating Systems																		
Credit Hours	3+1																		
Prerequisites by Course(s) and Topics	Data Structures and Algorithms																		
Assessment Instruments with Weights (homework, quizzes, midterms, final, programming assignments, lab work, etc.)	SESSIONAL (Quizzes, Assignments, Presentations) =25 % Midterm Exam =25 % Final Exam = 50%																		
Course Coordinator	Khola Farooq																		
URL (if any)																			
Current Catalog Description																			
Textbook (or Laboratory Manual for Laboratory Courses)	Operating Systems Concepts, 9th edition by Abraham Silberschatz																		
Reference Material	Modern Operating Systems, 4th edition by Andrew S. Tanenbaum o Operating Systems, Internals, and Design Principles, 9th edition by William Stallings o Davis, W.S. and Rajkumar, T.M. (2004) Operating Systems: A Systematic View. 3rd Ed.Harlow, Addison-Wesley.																		
Course Goals	At the end of the course, the students will be able to: Understand the characteristics of different Operating Systems structures and identify the Operating Systems' core functions. Analyse and evaluate the algorithms of the core functions of the Operating Systems and explain the major performance issues regarding the core functions. Demonstrate knowledge in applying system software and tools available in modern operating systems.																		
Course Learning Outcomes (CLOs):	<table><tr><td>At the end of the course the students will be able to:</td><td>Domain</td><td>BT Level*</td></tr><tr><td>Classify the characteristics of different structures of the Operating Systems and inspect the core functions and abstractions of the Operating Systems such as threads, semaphore etc.</td><td>C, P</td><td>C=2, P=1</td></tr><tr><td>Demonstrate the knowledge in applying system software and practise tools available in modern operating systems such as ubuntu, linux operating system.</td><td>C, P</td><td>C=3 P=2</td></tr><tr><td>Investigate and evaluate the algorithms of the core functions of the Operating Systems and simulate the performances regarding the core functions.</td><td>C, P</td><td>C=4 P=3</td></tr><tr><td>Evaluate and imitate the operating system's dead lock management techniques, memory management algorithms and resource management techniques.</td><td>C, P</td><td>C=5 P=3</td></tr><tr><td colspan="3">* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain</td></tr></table>	At the end of the course the students will be able to:	Domain	BT Level*	Classify the characteristics of different structures of the Operating Systems and inspect the core functions and abstractions of the Operating Systems such as threads, semaphore etc.	C, P	C=2, P=1	Demonstrate the knowledge in applying system software and practise tools available in modern operating systems such as ubuntu, linux operating system.	C, P	C=3 P=2	Investigate and evaluate the algorithms of the core functions of the Operating Systems and simulate the performances regarding the core functions.	C, P	C=4 P=3	Evaluate and imitate the operating system's dead lock management techniques, memory management algorithms and resource management techniques.	C, P	C=5 P=3	* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain		
At the end of the course the students will be able to:	Domain	BT Level*																	
Classify the characteristics of different structures of the Operating Systems and inspect the core functions and abstractions of the Operating Systems such as threads, semaphore etc.	C, P	C=2, P=1																	
Demonstrate the knowledge in applying system software and practise tools available in modern operating systems such as ubuntu, linux operating system.	C, P	C=3 P=2																	
Investigate and evaluate the algorithms of the core functions of the Operating Systems and simulate the performances regarding the core functions.	C, P	C=4 P=3																	
Evaluate and imitate the operating system's dead lock management techniques, memory management algorithms and resource management techniques.	C, P	C=5 P=3																	
* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain																			

Topics Covered in the Course, with Number of Lectures on Each Topic (assume 15-week instruction and one-hour lectures)	Week	Lecture	Topics Covered
	Week 1	1	Concepts Revision about Software, Hardware, Memory, User & Operating System Types, Features, Uses, Functions
		2	Installation of Virtual Machine
	Week 2	3	Installation of Ubuntu and error Solving
		4	Exploring Linux GUI Linux
	Week 3	5	Linux Basic Commands
		6	Linux Basic Commands
	Week 4	7	Linux Commands Short-cut keys in Linux
		8	Compiling and running C code in Linux
	Week 5	9	G++ Compiler Linux
		10	Process creation using system calls
	Week 6	11	Process Termination, wait () and Exist ()
		12	Implementing the Orphan process and Zombie process
	Week 7	13	Inter-process communication between processes using PIPES
		14	Implementing FCFS scheduling algorithm in C
	Week 8	1 hours	Mid Term
	Week 9	15	Implementing Round Robin scheduling algorithm in C
		16	Implement a Non-Preemptive priority scheduling algorithm
	Week 10	17	The preemptive priority scheduling algorithm
		18	Non-Preemptive Shortest Job First Scheduling algorithm
	Week 11	19	Mid Terms
		20	
	Week 12	21	Implementing Reader writer Problem in C
		22	Implementing Dining Philosopher's Problem in C
	Week 13	23	Implementing producer-consumer Problems in C
		24	Implementing Sleeping barber Problem in C
	Week 14	25	Implementing Banker's Algorithm for Deadlock avoidance in C
		26	Implementing Semaphore
	Week 15	27	Calculating Bit Map Overhead in Disk Free-Space Management in C
		28	Logical to Physical Address Translation in C
	Week 16	29	Calculating Paging Parameters and Paging Performance in C
		30	Simulate the Paging techniques of memory Management
	Week 17	2 hours	Final Term
Laboratory Projects/Experiments Done in the Course	Yes		
Programming Assignments Done in the Course	Yes		

Instructor Name Khola Farooq

Instructor Signature

Date