QF01/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Cyber Security Department

Study plan No.	**2024/2025**		**University Specialization**		**Cybersecurity**
Course No.	**0133314**		**Course name**		**Operating Systems**
Credit Hours	**3**		**Prerequisite Co-requisite**		**0133111**
Course type	☐ MANDATORY UNIVERSITY REQUIREMENT	☐ UNIVERSITY ELECTIVE REQUIREMENTS	☐ FACULTY MANDATORY REQUIREMENT	☐ Support course family requirements	☐ ✓ **Mandatory requirements** ☐ Elective requirements
Teaching style	☐ **Full online learning**		☐ ✓Blended learning		☐ Traditional learning
Teaching model	☐ ✓2Synchronous: 1asynchronous		☐ 2 face to face : 1synchronous		☐ **3 Traditional**

Faculty member and study divisions information (to be filled in each semester by the subject instructor)

Name	Academic rank	Office No.	Phone No.	E-mail	
Division number	**Time**	**Place**	**Number of students**	**Teaching style**	**Approved model**

Brief description

This course connect all computer architecture topics together, and help students to understand how properly the Oss are working. This course introduce the Operating System and Machine Architecture. Operating system and its instruction, the services provided by the OS, process management and its scheduling to the processor, type of scheduling and its algorithms, scheduling criteria's, the modern methods of design and implementation of OS, threads and its models and implementation, deadlock, type of algorithms for prevents the deadlock, manipulation with files, access to the files, the proper storage media for files, memory management, RAM, and VIRUAL memory, paging.

Learning resources

Course book information (Title, author, date of issue, publisher ... etc)	Silberschatz, galvin and Gagne, Operating System Concepts, Tenth Edtion, 2018			
Supportive learning resources (Books, databases, periodicals, software, applications, others)	1. Tomsho, Greg. Guide to operating systems. Cengage Learning, 2020. 2. Tanenbaum, Andrew S. Modern operating systems. China-Pub-Com, 2002. 3. Arpaci-Dusseau, Remzi H., and Andrea C. Arpaci-Dusseau. Operating systems: Three easy pieces. Arpaci-Dusseau Books, LLC, 2018. 4. Stallings, William. Operating systems: internals and design principles. Prentice Hall Press, 2011.			
Supporting websites				
The physical environment for teaching	☐ ✓**Class room**	☐ labs	☐ **Virtual educational platform**	☐ Others

QF01/0408-4.0E	Course Plan for Bachelor program - Study Plan Development and Updating Procedures/ Cyber Security Department

Necessary equipment and software	**Data show, Computer**
Supporting people with special needs	
For technical support	**E-learning and Open Educational Center. Computer Center**

Course learning outcomes (S= *Skills, C= Competences K= Knowledge,)*

No.	Course learning outcomes	The associated program learning output code
	Knowledge	
K1	Define OS, Computer-System organization and Architecture, including OS operation, Storage Structure and I/O structure. And Defining address binding, memory protection, Contiguous Allocation, Paging.	**PLO3,PLO5**
K2	Define and Suggest solutions for Deadlocks System Model and necessary conditions	**PLO5**
	Skills	
S1	Gain practical skills in system administration and configuration, including setting up user accounts, managing system resources, and configuring network services.	**PLO9**
S2	Develop the ability to diagnose and resolve common operating system issues, such as system crashes, performance bottlenecks, and software conflicts, including diagnostic tools and logs. And Acquire skills to interact with the operating system and develop applications using programming and scripting languages that are relevant to system-level tasks	**PLO8, PLO9**
	Competences	
C1	Ability to analyse and design the architecture of existing operating systems and implement basic operating system components, including creating processes, managing memory, handling I/O operations, and ensuring resource allocation and scheduling. And Diagnose and resolve complex system issues, and adapt to different operating system environments and handle unexpected challenges that may arise in real-world scenarios.	**PLO12**
C2	Evaluate the performance of operating systems and apply optimization techniques, including measuring and analysing system performance, identifying bottlenecks, and making appropriate improvement modifications.	**PLO12**

Mechanisms for direct evaluation of learning outcomes

Type of assessment / learning style	Fully electronic learning	Blended learning	Traditional Learning (Theory Learning)	Traditional Learning (Practical Learning)
First exam	0	**0**	0	0
Second / midterm exam	%30	**%30**	%30	%30
Participation / practical	0	**0**	0	0

applications				
Asynchronous interactive activities	%30	**%30**	%30	%30
final exam	%40	**%40**	%40	%40

Note: Asynchronous interactive activities are activities, tasks, projects, assignments, research, studies, projects, work within student groups ... etc, which the student carries out on his own, through the virtual platform without a direct encounter with the subject teacher.

Schedule of simultaneous / face-to-face encounters and their topics

Week	Subject	learning style*	Reference **
1	Introduction: What OS do (definition, user view)? History of OS System view of OS, Computer-System organization: OS operation, Storage Structure and I/O structure	Lectures	Lecture Notes
2	Operating System structure: Computer-system Architecture OS operation and Cashing, Operating System structure OS Services User OS Interface	Lectures	Lecture Notes
3	System calls system programs, Simple, layered, Modules and microkernel Structure	Lectures	Lecture Notes
4	Process: Virtual machines Processes: process Control block, states of process, process scheduling: queues, context switch, Process: schedulers Operations on processes process creation process termination	Lectures	Lecture Notes
5	Process: Inter-process communication: Shared-memory & Message-Passing, Threads: definition, Multithreading Models	Lectures	Lecture Notes
6	CPU Scheduling: CPU Scheduling: CPU and I/O bursts,	Lectures	Lecture Notes

	CPU Scheduling: SJF scheduling algorithms Scheduling Criteria and FCFS		
7	CPU Scheduling: SJF Non-Preemptive scheduling SJF Preemptive scheduling, CPU Scheduling: Exercises	Lectures	Lecture Notes
Midterm Exam (30%)			
9	CPU Scheduling: Priority scheduling algorithms Non-Preemptive Scheduling Preemptive Scheduling, CPU Scheduling: RR scheduling algorithm Multilevel and Multilevel-feedback queue scheduling algorithm	Lectures	Lecture Notes
10	Process Synchronization: Race Condition Critical Section & solution Hardware Instructions: TestAndSet() and Swap() Semaphores Spinlocks and problems with semaphores Bounded-buffer classical synchronization problem, Process Synchronization: Readers-Writers classical synchronization problem Dining-philosopher classical synchronization problem Process Synchronization: Monitors Transactions: definition, log-based recovery and checkpoints	Lectures	Lecture Notes
11	Deadlocks: System Model and necessary conditions, Deadlocks: Resource Allocation Graph	Lectures	Lecture Notes
12	Deadlocks: Deadlock prevention Deadlock Avoidance: Safe State and resource-allocation graph, Deadlocks: Deadlock Avoidance: Banker's Algorithm	Lectures	Lecture Notes

13	Deadlocks: Deadlock Avoidance: Banker's Algorithm Safety algorithm, Deadlocks: Deadlock Recovery: Process Termination and resource preemption	Lectures	Lecture Notes
14	Memory Management: definition, address binding, memory protection Contiguous Allocation, Paging, Memory Management: virtual memory Demand paging and Page Replacement	Lectures	Lecture Notes
15	Projects Discussion		
	Final Exam (40%)		

*** Learning styles: Lecture, flipped learning, learning through projects, learning through problem solving, participatory learning ... etc.**

**** Reference: Pages in a book, database, recorded lecture, content on the e-learning platform, video, website ... etc.**

This activities was designed using the **Project-Based Learning (PBL)**

Project : Process Synchronization (PBL)

Task / Activity	Reference	Expected Results
Explain race condition and its risks	Process Synchronization – Race Condition	Clear understanding of race problems
Identify and design critical section solutions	Critical Section & Solutions	Safe execution section design
Use hardware synchronization (TestAndSet / Swap)	Hardware Instructions: TestAndSet(), Swap()	Demonstrated mutual exclusion
Implement semaphores for process coordination	Semaphores	Controlled process synchronization
Analyze spinlocks and semaphore issues	Spinlocks & Problems	Identification of performance limitations
Solve bounded buffer problem	Bounded-Buffer Problem	Working producer–consumer solution
Solve readers–writers synchronization	Readers-Writers Problem	Fair and synchronized access
Solve dining philosophers problem	Dining Philosopher Problem	Deadlock-free coordinated processes
Implement monitors for synchronization	Monitors	Structured and safe synchronization
Prepare final documentation	PBL Documentation	Structured synchronization project report

Schedule of asynchronous interactive activities (in the case of e-learning and blended learning)

Week	Task / activity	Reference	Expected results
1			
2			
3			
4			
5			
6			
7			
8			