Semester: III						
OPERATING SYSTEMS **Category: PROFESSIONAL CORE COURSE** **(Theory and Practice)** **(Common to CS, IS, CD & CY)**						
Course Code	**:**	**CS235AI**		**CIE**	**:**	**100 + 50 Marks**
Credits: L:T:P	**:**	**3:0:1**		**SEE**	**:**	**100 + 50 Marks**
Total Hours	**:**	**45L + 30P**		**SEE Duration**	**:**	**3 + 3 Hours**

Unit-I	**10 Hrs.**
Introduction- Perspectives Business domain: Virtualization and Cloud Computing Application: Traditional computing, Mobile computing, Distributed systems **Introduction** Operating System introduction, Operating System structure, Operating system Operations. **System Structures** Operating system services, System Calls, Types of System calls **Process Management** Process concept, Process scheduling, Operations on processes	
Unit – II	**08 Hrs.**
Multithreaded programming Overview, Multicore programming, Multithreading models, Thread libraries - pthreads **CPU scheduling and Process Synchronization** Basic concepts, scheduling criteria, scheduling algorithms-FCFS, SJF, RR, priority, Real-time CPU scheduling	
Unit –III	**09 Hrs.**
Process Synchronization Background, The Critical section problem, Peterson's Solution **Process Synchronization** Synchronization hardware, Mutex locks, Semaphores, Classic problems of synchronization **Case study:** Implementation of classic synchronization problem using semaphores	
Unit –IV	**08 Hrs.**
Main Memory Management Background, Swapping, Contiguous memory allocation, Segmentation, Paging, Structure of page table. **Virtual memory** Background, Demand Paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing	
Unit –V	**10 Hrs.**
File Systems File Naming, File Structure, File Types, File Access, File Attributes, File Operations, An example program using File-System calls, File-System Layout, Implementing Files. **The Virtual File System:** The role of the Virtual File System (VFS), VFS data structure, Filesystem Types, Filesystem handling, Pathname lookup, Implementation of VFS System calls, File Locking.	

Course Outcomes: After completing the course, the students will be able to:-	
CO 1	Demonstrate the fundamental concepts of operating system like process management, file management, memory management and issues of synchronization.
CO 2	Analyze and interpret operating system concepts to acquire a detailed understanding of the course.
CO 3	Apply the operating systems concepts to address related new problems in computer science Domain.
CO 4	Design or develop solutions using modern tools to solve applicable problems in operating systems domain.
CO5	Extend the theoretical knowledge acquired through the course to demonstrate skills like investigation, effective communication, working in team/Individual, following ethical practices by implementing operating system concepts/applications and engage in lifelong learning.

Reference Books	
1.	Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin , Greg Gagne, 9th Edition, Incorporated, 2018, John Wiley & Sons, ISBN 978-1-265-5427-0
2.	Modern operating systems, Tanenbaum, Andrew, 4th Edition, Pearson Education, Inc 2009. ISBN 013359162X, 978-0133591620
3.	UNIX System Programming Using C++, Terrence Chan, 2011, Prentice Hall India, ISBN: 9788120314689 978-8120314689.
4.	Operating systems - A concept based Approach, D.M Dhamdhere, 3rd Edition, 2017, Tata McGraw-Hill, ISBN: 1259005585, 978-1259005589
5.	"xv6: a simple, Unix-like teaching operating system", https://pdos.csail.mit.edu/6.828/2014/xv6/book-rev8.pdf
6.	Understanding the LINUX Kernal, Daniel P Bovet and Marco Cesati, 3rd Edition, 17 November 2005, O'Reilly Publication, 9780596554910, 0596554915. (For Virtual File System of fifth unit)

Laboratory Component
PART A 1. Implementation of basic UNIX commands using file APIs- Write a program to implementcommandsls(-l option), cp, rm and mv using UNIX file APIs. 2. Apply the concepts of Process control system calls to build applications to demonstrate use of fork, execve, wait, getpid, exit system calls 3. Apply the pthread library to build Applications to demonstrate use of pthread library functions to create and manage threads. 4. Apply the concepts of Process/Thread synchronization to build Applications to demonstrate process/thread synchronization using semaphores and mutex. Implement Dining philosophers problem, reader-writer and producer-consumer. 5. Apply the concepts of Process/Thread synchronization for file access to build applications to demonstrate process/thread synchronization using file locks. 6. Apply the concepts of Static and Shared libraries to write a program to create and use static and shared libraries. Demonstrate the advantage of shared libraries over static libraries in terms of memory usage.

PART B
Open Ended Project

The students are expected to implement a mini project using operating system concepts andAPIs/system calls. They are required to form a team with constraint of maximum 3 persons in a team, select a problem/application of their choice to implement and to take confirmation from faculty incharge before starting the project. The objectives of project implementation are:

- Explore and understand underlying architecture, kernel structure and associated components for implementation of the project.
- Design and implement the solution using appropriate tools and platform.
- Documentation and submission of report

Sample projects can be kernel implementation from scratch, compiler implementation, assembler implementation, iOS system level programs, Android OS system level programs, Embedded OS system level programs, Raspberry Pi OS implementation, File System implementation and similar such projects.

RUBRIC FOR THE CONTINUOUS INTERNAL EVALUATION		
#	**COMPONENTS**	**MARKS**
1.	**QUIZZES:** Quizzes will be conducted in online/offline mode. **TWO QUIZZES** will be conducted & Each Quiz will be evaluated for 10 Marks. Each quiz is evaluated for 10 marks adding up to 20 MARKS	**20**
2.	**TESTS:** Students will be evaluated in test, descriptive questions with different complexity levels (Revised Bloom's Taxonomy Levels: Remembering, Understanding, Applying, Analyzing, Evaluating, and Creating). **TWO tests will be conducted**. Each test will be evaluated for **50Marks**, adding upto 100 Marks. **FINAL TEST MARKS WILL BE REDUCED TO 40 MARKS.**	**40**
3.	**EXPERIENTIAL LEARNING:** Students will be evaluated for their creativity and practical implementation of the problem. Case study based teaching learning (10), Program specific requirements (10), Video based seminar/presentation/demonstration (10) Designing & Modeling (10)**Phase 2 will be done in the exhibition mode (Demo/Prototype/any outcome). ADDING UPTO 40 MARKS**.	**40**
4.	**LAB:** Conduction of laboratory exercises, lab report, observation, and analysis (20 Marks), lab test (10 Marks) and Innovative Experiment/ Concept Design and Implementation (20 Marks) adding up to 50 Marks. THE FINAL MARKS WILL BE 50MARKS	**50**
	MAXIMUM MARKS FOR THE CIE (THEORY AND PRATICE)	**150**

RUBRIC FOR SEMESTER END EXAMINATION (THEORY)		
Q.NO.	**CONTENTS**	**MARKS**
	PART A	
1	Objective type of questions covering entire syllabus	20
	PART B (Maximum of THREESub-divisions only)	
2	Unit 1 : (Compulsory)	16
3 & 4	Unit 2 : Question 3 or 4	16
5 & 6	Unit 3 : Question 5 or 6	16
7 & 8	Unit 4 : Question 7 or 8	16
9 & 10	Unit 5: Question 9 or 10	16
	TOTAL	**100**

RUBRIC FOR SEMESTER END EXAMINATION (LAB)		
Q.NO.	**CONTENTS**	**MARKS**
1	Write Up	10
2	Conduction of the Experiments	20
3	Viva	20
	TOTAL	**50**