Course Title	**Operating Systems Lab**		
Course Code	**CC-311-L**		
Credit Hours	3 (2,1)		
Category	Computing Core		
Prerequisite	None		
Co-Requisite	None		
Follow-up	None		
Course Introduction	To help students gain a general understanding of the principles and concepts governing the functions of operating systems and acquaint students with the layered approach that makes design, implementation and operation of the complex OS possible.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand the characteristics of different structures of the Operating Systems and the core functions of the Operating Systems	C2 (Understand)	1,2
	CLO2: Analyze and evaluate the algorithms of the core functions of Operating Systems and explain major performance issues with regard to the core functions	C6 (Evaluate)	3
	CLO3: Demonstrate knowledge in applying system software and tools available in modern operating systems	C3 (Demonstrate)	3,4
Course Description	**Introduction:** Operating systems basics, system calls, process concept and scheduling, inter-process communication, multithreaded programming, multithreading models, threading issues. **Process Scheduling:** Algorithms, thread scheduling, multiple-processor scheduling, synchronization, critical section, synchronization hardware, synchronization problems, deadlocks, detecting and recovering from deadlocks. **Memory Management:** swapping, contiguous memory allocation, segmentation & paging, virtual memory management, demand paging, thrashing, memory-mapped files. **File Systems**: file concept, directory and disk structure, directory implementation, free space management, disk structure and scheduling, swap space management. **System Protection**: Virtual machines, operating system security. **Interacting with Linux Operating System:** Virtualization and Hypervisors. **File System Architecture:** Schematic view of a standard UNIX file system. **File System Mounting:** Introduction to the concept of file system mounting. Linux configuration files related to file system mounting. **File Permissions:** standard file permissions. Use of chmod and chown commands. **Device files:** Seven File Types in Linux and the concept of device files. **Terminal Attributes:** Overview of Terminal Devices and current attributes of the terminal driver. **Hard and Soft Links. Managing services using systemd:** Introduction to Linux system daemon. Shell commands to manage services using systemctl. **Booting process of a Linux system.** **Linux System Programming:** Linux System Call Interface, Use of GNU gcc compiler **Process Creation and Termination:** getpid(), getppid(), fork(), exit(), wait() and execl() system calls. **File management in Linux.** Concept of PPFDT. Concept of input, output and error redirection. **Inter Process Communication:** Linux IPC tools, Pipes, FIFOS and Sockets. Use of pipes and fifos on a Linux terminal. **Signals:** Signal delivery and execution of a signal handler. Synchronous and Asynchronous signals. **Threads and Scheduling:** Writing multi-threaded C programs using library calls from the POSIX pthread library like pthread_create(), pthread_join(), and pthread_exit().		
Text Book(s)	1. A. Silberschatz, P. B. Galvin, G. Gagne, Operating Systems Concepts, 9th Edition, Wiley, 2012, ISBN: 1118063333.		
Reference Material	1. Andrew S. Tanenbaum, Herbert Bos, Modern Operating Systems, 4th Edition, Pearson, 2014, ISBN: 013359162X. 2. William Stallings, Operating Systems: Internals and Design Principles, 9th Edition, Pearson, 2017, ISBN: 0134670957.		