**Concepts of Operating Systems Duration: 26 hours (18 theory hours + 8 lab hours)**

Objective: To introduce Operating System concepts with Linux environment, and to learn Shell Programming.

Prerequisites: Knowledge of computer fundamentals

Evaluation: 35 marks (CCEE: 15 + Lab exam: 10 + Internals: 10)

Text Books: • Operating Systems Principles by Abraham Silberschatz, Peter Galvin& Greg Gagne / Wiley • Unix Concepts and Applications by Sumitabha Das / McGraw Hill References:

• Modern operating Systems by Andrew Tanenbaum & Herbert Bos/ Pearson

• Principles of Operating Systems by Naresh Chauhan / Oxford University Press

• Beginning Linux Programming by Neil Matthew & Richard Stones / Wrox

• Operating System : A Design-Oriented Approach by Charles Crowley / McGraw Hill

(Note: Each Session is of 2 hours)

**Session 1: Introduction to OS**

Lecture: • What is OS; How is it different from other application software; Why is it hardware dependent? • Different components of OS • Basic computer organization required for OS. • Examples of well-known OS including mobile OS, embedded system OS, Real Time OS, desktop OS server machine OS etc. ; How are these different from each other and why • Functions of OS • User and Kernel space and mode; Interrupts and system calls

No Lab

**Session 2: Introduction to Linux Lecture:**

• Working basics of file system • Commands associated with files/directories & other basic commands. Operators like redirection, pipe • What are file permissions and how to set them? • Permissions (chmod, chown, etc); access control list; network commands (telenet, ftp, ssh, sftp, finger) • System variables like – PS1, PS2 etc. How to set them

Shell Programming

• What is shell; What are different shells in Linux?

• Shell variables; Wildcard symbols

• Shell meta characters; Command line arguments; Read, Echo

Lab: (4 hours)

• Working with various OS commands

• Shell programs related to Session 2

**Session 3: Shell Programming**

Lecture: • Decision loops (if else, test, nested if else, case controls, while…until, for)

• Regular expressions; Arithmetic expressions

• More examples in Shell Programming

Lab: (4 hours) • Shell Programs related to Session 3

**Sessions 4 & 5: Processes Lecture:**

• What is process; preemptive and non-preemptive processes

• Process management; Process life cycle

• What are schedulers – Short term, Mediumterm and Long term.

• Process scheduling algorithms – FCFS, Shortest Job First, Priority, RR, Queue. Belady’s Anomaly

• Examples associated with scheduling algorithms to find turnaround time to find the better performing scheduler.

• Process creation using fork; waitpid and exec system calls; Examples on process creation; Parent and child processes

• Orphan and zombie processes

No Lab

**Sessions 6 & 7: Lecture: Memory Management**

• What are different types of memories; What is the need of Memory management

• Continuous and Dynamic allocation

• First Fit, Best Fit, worst Fit

• Compaction

• Internal and external fragmentation

• Segmentation – What is segmentation; Hardware requirement for segmentation; segmentation table and its interpretation

• Paging – What is paging; hardware required for paging; paging table; Translation look aside buffer

• Concept of dirty bit

• Shared pages and reentrant code

• Throttling

No Lab

**Session 8: Lecture: Virtual Memory**

• What is virtual memory

• Demand paging • Page faults

• Page replacement algorithms

No Lab

**Session 9: Lecture: Deadlock**

• Necessary conditions of deadlock

• Deadlock prevention and avoidance

• Semaphore

• Mutex

• Producer consumer problem

• Dead-lock vs Starvation

No Lab