

LINUX PROGRAMMING (Common to CSE & IT)

III B. Tech. - I Semester
Course Code: A3CS19

L T P C
3 1 - 3

Course Overview:

This course explains the fundamental ideas behind the open source operating system approach to programming. Knowledge of Linux helps to understand OS level programming. Like the successful computer languages that came before, Linux is the blend of the best elements of its rich heritage combined with the innovative concepts required by its unique environment. This course involves kernel concepts, basics commands, shell scripting, file processing, Socket programming, Processes, Inter process communication. This course is presented to students by power point projections, course handouts, lecture notes, assignments, objective and subjective tests

Course Objectives:

1. To teach principles of operating system including File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking Commands, Basic Linux commands, Scripts and filters.
2. To familiarize fundamentals of the Bourne again shell (bash), shell programming, pipes, input and output redirection Control structures, arithmetic in shell interrupt processing, functions, debugging shell scripts.
3. To impart fundamentals of file concepts kernel support for file, File structure related system calls (file API's).
4. To facilitate students in understanding Inter process communication.
5. To facilitate students in understanding semaphore and shared memory.
6. To facilitate students in understanding process.

Course Outcomes:

1. Ability to use various Linux commands that are used to manipulate system operations at admin level and a prerequisite to pursue job as a Network administrator.
2. Ability to write Shell Programming using Linux commands.
3. Ability to design and write application to manipulate internal kernel level Linux File System.
4. Ability to develop IPC-API's that can be used to control various processes for synchronization.
5. Ability to develop Network Programming that allows applications to make efficient use of resources available on different machines in a network.

SYLLABUS

UNIT – I

INTRODUCTION TO LINUX AND LINUX UTILITIES: A brief history of LINUX, architecture of LINUX, features of LINUX, introduction to vi editor.

Linux commands- PATH, man, echo, printf, script, passwd, uname, who, date, stty, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, lp, od, tar, gzip, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, unlink, du, df, mount, umount, find, unmask, ulimit, ps, w, finger, arp, ftp, telnet, rlogin. Text Processing utilities and backup utilities, tail, head, sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, cpio

UNIT - II

Introduction to Shells: Linux Session, Standard Streams, Redirection, Pipes, Tee Command, Command Execution, Command-Line Editing, Quotes, Command Substitution, Job Control, Aliases, Variables, Predefined Variables, Options, Shell/Environment Customization.

Filters: Filters and Pipes, Concatenating files, Display Beginning and End of files, Cut and Paste, Sorting, Translating Characters, Files with Duplicate Lines, Count Characters, Words or Lines, Comparing Files.

UNIT - III

Grep: Operation, grep Family, Searching for File Content.
Sed :Scripts, Operation, Addresses, commands, Applications, grep and sed.

UNIX FILE STRUCTURE: Introduction to UNIX file system, inode (Index Node), file descriptors, system calls and device drivers.

File Management :File Structures, System Calls for File Management – create, open, close, read, write, lseek, link, symlink, unlink, stat, fstat, lstat, chmod, chown, Directory API – opendir, readdir, closedir, mkdir, rmdir, umask.

UNIT - IV

PROCESS AND SIGNALS: Process, process identifiers, process structure: process table, viewing processes, system processes, process scheduling, starting new processes: waiting for a process, zombie processes, orphan process, fork, vfork, exit, wait, waitpid, exec, signals functions, unreliable signals, interrupted system calls, kill, raise, alarm, pause, abort, system, sleep functions, signal sets. File locking: creating lock files, locking regions, use of read and write with locking, competing locks, other lock commands, deadlocks.

UNIT - V

INTER PROCESS COMMUNICATION: Pipe, process pipes, the pipe call, parent and child processes, and named pipes: fifos, semaphores: semget, semop, semctl, message queues: msgget, msgsnd, msgrcv, msgctl, shared memory: shmget, shmat, shmdt, shmctl, ipc status commands.

INTRODUCTION TO SOCKETS: Socket, socket connections - socket attributes, socket addresses, socket, connect, bind, listen, accept, socket communications.

TEXT BOOKS:

1. *W. Richard. Stevens (2005), Advanced Programming in the UNIX Environment, 3rd edition, Pearson Education, New Delhi, India.*
2. *Unix and shell Programming Behrouz A. Forouzan, Richard F. Gilberg. Thomson*

REFERENCES:

1. *Linux System Programming, Robert Love, O'Reilly, SPD.*
2. *Advanced Programming in the UNIX environment, 2nd Edition, W.R.Stevens, Pearson Education.*
3. *UNIX Network Programming, W.R. Stevens, PHI.*
UNIX for Programmers and Users, 3rd Edition, Graham Glass, King Ables, Pearson Education