CSE325:OPERATING SYSTEMS LABORATORY

L:0 T:0 P:2 Credits:1

Course Outcomes: Through this course students should be able to

CO1 :: execute Linux commands and basic shell programming tasks for script development

CO2 :: identify and describe the role of environment variables, process image replacement in Linux systems

CO3 :: utilize file system-related system calls to manage files efficiently.

CO4 :: demonstrate the creation and management of processes and threads using Linux system calls

CO5 :: illustrate the use of synchronization mechanisms like mutexes and semaphores to prevent race conditions

CO6 :: explain and implement inter-process communication techniques such as pipes, shared memory, and message passing

List of Practicals / Experiments:

Process creation and threading

- Creating processes
- Creating Threads
- Process duplication using fork()
- Creating threads using pthread
- Environment variables
- Replacing process image using execlp

Inter-process communication

- Pipes, popen and pclose functions
- Stream pipes, passing file descriptors
- Shared memory
- Message passing
- Remote Procedure calls

Introduction to Linux

- Basic Linux Commands: ls, cat, man, cd, touch, cp, mv, rmdir, mkdir, rm, chmod, pwd
- System Calls: Read, Write, Open
- Lseek

Synchronization

- Synchronization with Mutexes
- Synchronization with semaphores
- Race Condition

shell programming

- variables
- standard input/output redirection
- shell arithmetic
- flow control and decision making

File and directory management using system calls

- File related system calls (open, read, write, lseek, close)
- Directory related system calls (opendir, readdir, closedir etc)

Text Books:

1. BEGINING LINUX PROGRAMMING by NEIL MATHEW & RICHARD STONES, WILEY

References:

1. OPERATING SYSTEM CONCEPTS by ABRAHAM SILBERSCHATZ, GALVIN, WILEY
2. UNIX CONCEPTS AND APPLICATIONS by SUMITABHA DAS, Tata McGraw Hill, India