L	T	P	C
0	0	2	1

Course Code: CSE309
Semester: V

OPERATING SYSTEMS LABORATORY

Course Objectives

This course will help the learner to gain pragmatic knowledge on the different modules of operating system by simulating them and analysing their performance differences.

1. Program to create of a child process from a parent process using fork system call and make them communicate between them using pipe.
2. a. Program to implement IPC using Shared Memory System Call.
   b. Program to implement IPC with the help of Message Queues.
3. Programs for the simulation of uni-processor scheduling algorithms and analyze their performances.
4. Program to experiment multi-processor scheduling.
5. Program for the simulation of thread scheduling approaches.
6. Program to implement peterson's algorithm for enforcing mutual exclusion.
7. a. Program to demonstrate for producer-consumer problem using semaphore.
   b. Program to apply semaphore for tackling reader-writer problem.
8. Program to apply banker's algorithm for deadlock avoidance.
9. Program to implement deadlock detection algorithm.
10. Program to implement dining philosopher's problem without causing deadlocks.
11. a. Program to simulate page replacement algorithms and to compute number of page faults
    b. Program to simulate address translation under paging.
12. Program to implement disk scheduling algorithms.

Additional Experiments:

13. Program to simulate dynamic partitioning and buddy system.
14. Program to demonstrate file allocation techniques.

COURSE LEARNING OUTCOMES

Upon successful completion of this course, the learner will be able to

- Develop program that create parent/child or concurrent process and carry out communication between them using pipe or IPC methods
- Analyze uni-processor, multi-processor and thread scheduling algorithms by simulating them and providing them with sample data
- Implement mutual exclusion based on semaphores to prevent concurrent issues under classic problems of concurrency
- Demonstrate deadlock avoidance and detection strategies
- Simulate memory management schemes and disk scheduling schemes
- Demonstrate file allocation techniques