ANURAG ENGINEERING COLLEGE

(An Autonomous Institution)

II Year B.Tech. CSE - II Sem

L	T	P	C
0	0	2	1

(CS406PC) OPERATING SYSTEMS LABORATORY

Prerequisites:

- A course on "Programming for Problem Solving", A course on "Computer Organization and Architecture".

Co-requisite:

- A course on "Operating Systems".

Course Objectives:

The objectives of this course are to provide:

- Analyze system calls that can offer operating system services
- Demonstrate various operating system concepts
- Understand and apply concepts towards new operating system design
- Understand the concept of Dead lock and its avoidance and page replacement algorithms
- Understand Unix commands, system calls

List of Experiments:

1. Write C programs to simulate the following CPU Scheduling algorithms
   a) FCFS b) SJF c) RoundRobin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system
   (open, read, write, close,fcntl, seek, stat, opendir, readdir)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms
   a) Pipes b) FIFOs c) Message Queues d) Shared Memory
6. Write C programs to simulate the following memory management techniques
   a) Paging b)Segmentation
7. Write C programs to simulate Page replacement policies
   a) FCFS b) LRU c) Optimal

Text Books:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition,John Wiley
2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

Reference Books:

1. Operating Systems – Internals and Design Principles, William Stallings, Fifth Edition–2005,Pearson Education/PHI
2. Operating System - A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI
4. UNIX Programming Environment, Kernighan and Pike, PHI/Pearson Education
5. UNIX Internals: The New Frontiers, U. Vahalia, Pearson Education

Course Outcomes:

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

1. Simulate and implement operating system concepts such as scheduling,
2. Able to implement C programs using Unix system calls
3. Implement the dead lock avoidance using banker's algorithm
4. Implement the producer and consumer problem and Page Replacement algorithms
5. Exercise interprocess communication.

CO-PO-PSO Mapping:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2
CO-1	H	M	M	L	L							L	M	L
CO-2	M	H	L	M	L							L	M	L
CO-3	M	L	H	L	L							M	M	H
CO-4	H	M	H										M	H
CO-5	H		M	L	L							L	M	L

H-HIGH **M**-MODERATE **L**-LOW