**CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY**

**FACULTY OF TECHNOLOGY & ENGINEERING**

**Devang Patel Institute of Advance Technology and Research**

**Department of Computer Engineering**

**Subject Name:** Operating System **Semester :** IV

**Subject Code:** CE248 **Academic year:** 2019-20

Practical List

**Instructions:**

ISO Practical Format: Aim, Software/Hardware Required, Knowledge Required, Theory, Algorithm/Flow chart, Program, Input and Output, Advantages and Disadvantages, Conclusion, Questions and Answers.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Aim Of the Practical** | **Hrs.** | **LO** | **PO** | **PEO** |
|  | **Working of Different Kernels:**  A. UNIX Architecture  B. Types of OS- Linux, Unix, MAC, Window etc.  C. Flavors of LINUX | 2 | 1 |  |  |
|  | Study of Unix Architecture and the following Unix commands with option:   |  |  | | --- | --- | | **User Access:** | login, logout, passwd, exit | | **Help:** | man, help | | **Directory:** | mkdir, rmdir, cd, pwd, ls, mv | | **Editor:** | vi, gedit, ed, sed | | **File Handling / Text Processing:** | cp, mv, rm, sort, cat, pg, lp, pr, file, find, more, cmp, diff, comm, head, tail, cut, grep, touch, tr, uniq | | **Security and Protection:** | chmod, chown, chgrp, newgrp | | **Information:** | learn, man, who, date, cal, tty, calendar, time, bc, whoami, which, hostname, history, wc | | **System Administrator:** | su or root, date, fsck, init 2, wall, shut down, mkfs, mount, unmount, dump, restor, tar, adduser, rmuser | | **Terminal:** | echo, printf, clear | | **Process:** | ps, kill, exec | | **I/O Redirection** (<, >, >>), **Pipe** ( | ), \*, gcc | | | 4 | 1 | 2 | 7 |
|  | 1. Write a shell script which calculatesnth Fibonacci number where n will be provided as input when prompted. 2. Write a shell script which takes one number from user and finds factorial of a given number. 3. Write a shell script to sort the number in ascending order. (Using array). | 2 | 3 | 5 | 3 |
|  | **Write programs using the following system calls of UNIX operating system: fork, exec, getpid, exit, wait, stat, readdir, opendir.**   1. Write a program to execute fork() and find out the process id by getpid() system call. 2. Write a program to execute following system call fork(), execl(), getpid(), exit(), wait() for a process. 3. Write a program to find out status of named file (program of working stat() system call). 4. Write a program for “ls” command implementation using opendir() & readdir() system call. | 2 | 1 | 4,3 | 8 |
|  | Process control system calls:  A. The demonstration of fork()  B. execve() and wait() system calls along with zombie and orphan states. | 2 | 2 | 4,3 | 3 |
|  | Write a C program in UNIX to implement Process scheduling algorithms and compare.  A. First Come First Serve (FCFS) Scheduling  B. Shortest-Job-First (SJF) Scheduling  C. Priority Scheduling (Non-preemption) after completion extend on Preemption.  D. Round Robin(RR) Scheduling | 4 | 1,3 | 3 | 1,6 |
|  | Thread management using pthread library. Write a simple program to understand it. | 3 | 1,2 | 2,8 | 1,7 |
|  | Write a C program in UNIXto implement Bankers algorithm for Deadlock Avoidance. | 2 | 3,2 | 6 | 2 |
|  | Write a C program in UNIXto perform Memory allocation algorithms and calculate Internal and External Fragmentation. (First Fit, Best Fit, Worst Fit). | 2 | 3,2 | 6 | 2 |
|  | Thread synchronization using counting semaphores and mutual exclusion using mutex. | 2 | 3 | 5,9 | 2,7 |
|  | Write a C program in UNIX to implement inter process communication (IPC) using Semaphore. | 2 | 1 | 4,3 | 8 |
|  | Kernel space programming: Implement and add a loadable kernel module to Linux kernel, demonstrate using insmod, lsmod and rmmod commands. A sample kernel space program should print the "Hello World" while loading the kernel module and "Goodbye World" while unloading the kernel module. | 3 | 2,3 | 5,7 | 3,8 |
|  | **Total Hours** | **30** |  |  |  |

**Additional Practical(s):**

1. To implement of Dinning Philoshopr problem

A. Dinning Phiolosphor

B. Reader-Writer

1. To implement Disk-Scheduling Algorithm(s).
2. H2O Building Problem
3. Dining Savages Problem
4. Sleeping Barber Problem