

CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY
FACULTY OF TECHNOLOGY & ENGINEERING
 U & P U Patel Department of Computer Engineering

Subject Name: Internals of Operating System
Subject Code: CE347

Semester : 6th
Academic Year: Dec-Apr 2020

Practical List

Sr.No.		Aim of Practical
1.		Collect the following basic information about your machine using proc. a. How many CPU cores does the machine have? b. How much memory, and what fraction of it is free? c. How many context switches has the system performed since bootup? d. How many processes has it forked since bootup? e. How many processors does your machine have? f. What is the frequency of each processor? g. Find out various states of process at time of observation.
2.	[A]	Implement copy command using open, create, read, write, access and close system call. Be sure to include all necessary error checking including, ensuring the source file exists. Test your program with following specifications. a. File extension with .txt , .c , .zip , .exe , .tar b. Copy whole directory. Assignment: a. Find the real time, processor time, user space time and kernel space time for copy command implementation. b. Copy the source file in two different files parallelly. Find out does it take same time to finish. (Create two threads) (Files has to be atleast 100 MB) c. Once you have correctly designed and tested the program, run the program using a utility (<i>ptrace</i>) that traces system calls. (Find out which system call/calls program has made internally)
	[B]	Write a program for 'ls' command using 'opendir()' and 'readdir()' system call.
3.	[A]	Write a C program that will print parent process id and child process id. Mention error checking if child process is not created.
	[B]	In continuation of part (a), write a C program where parent process wait for child process to terminate.
	[C]	Write a C program using <code>execvp()</code> system call which will count the characters from file 'wc', using program 'p.c'.
		Assignment: Write a C program that will sleep the process for 5000 seconds. Run the same program for three times sequentially. Note down your observations for following questions. a. Every time you execute the program, does that process have the same process id?

		<p>b. Note down following characteristics of process control block using <i>proc</i> and <i>top</i> command.</p> <ul style="list-style-type: none"> • Process state • Process number • Program counter • Memory limits • Open files list • Number of voluntary context
4.		<p>Write a program uses dup() and dup2() system call that will prove the following sentence: <i>“dup() always uses the smallest available (unused) file descriptor whereas dup2() uses new file descriptor.”</i></p>
5.		<p>Write the following programs using inter process communication – shared memory. The program ‘writer.c’ will print 1 to 100 in shared memory region. Another program ‘reader.c’ that will read all the numbers from shared memory to make addition of it and display it. Assignment: Solve above issue using pipe().</p>
6.		<p>Consider a process executing on a CPU. Give an example scenario that can cause the process to undergo: (a) A voluntary context switch. (b) An involuntary context switch Write the program for both the cases.</p>
7.		<p>Write a C program to implement below system call. iget() , iput() Assignment: Implement namei() system call.</p>
8.	[A]	<p>Create 1GB swap area in your linux partition and free it. Check the allocation of swap space. Execute following commands to monitor swap space in linux.</p> <ol style="list-style-type: none"> Swapon use of /proc/swaps free top atop htop glances vmstat
	[B]	<p>Write the simulation Paging Algorithms program for demand paging and show the page scheduling and total number of page faults according to FIFO, LRU, and optimal page replacement algorithm. Assume the memory of ‘n’ frames.</p>
9.		<p>Assume that processes communicate by send/receive messages, which are unreliable, e.g. messages may be lost during send/recv. After sending a message, a process expects a reply. If it does not receive a reply within ‘t’ seconds, it re-sends the same message again. If it receives a reply within ‘t’ seconds, it must not send the same message again. Design an algorithm for the sending process and implement it.</p>
10.		<p>Implementation of a device driver to find reverse string in kernel mode.</p>

