SRM Institute of Science and Technology
College of Engineering and Technology
School of Computing
DEPARTMENT OF COMPUTING TECHNOLOGIES
SRM Nagar, Kattankulathur – 603203, Chengalpattu District, Tamilnadu
Academic Year: 2021-2022 (EVEN)

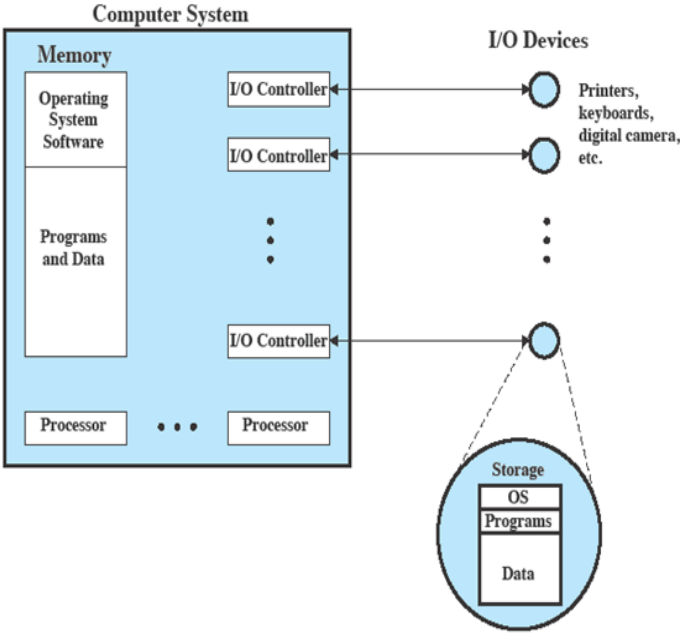
Test: CLAT-1 **Date: 4-4-2022**
Course Code & Title: 18CSC205J: Operating systems **Duration:** 1 Period
Year & Sem: II & IV **Max. Marks:** 25 Marks

Course Outcomes (CO):	*At the end of this course, learners will be able to:*
CO-1 :	*Express the fundamental concepts in operating system*

Program Outcomes (PO)												PSO		
1	2	3	4	5	6	7	8	9	10	11	12			
Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual & Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
3		*3*										*2*		

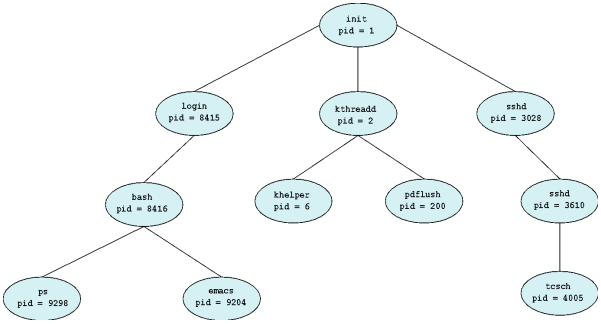
Part - A
(5 x 1 = 5 Marks) Instructions: Answer all

Q. No	Question	Marks	BL	CO	PO	PI Code
1	Multiprocessor system that computer systems have are also called a) Parallel Systems b) Tightly couple system c) loosely couple system **d) Both A and B**	**1**	**1**	**1**	**1**	**1.6.1**
2	Thread scheduling of operating system programs is done by a) Input b) Output **c) Operating system** d) Memory	**1**	**1**	**1**	**1**	**1.6.1**
3	Multiprogramming of computer system increases a) memory b) Storage **c) CPU utilization** d) Cost	**1**	**1**	**1**	**1**	**1.6.1**
4	Consider the following code fragment: if (fork() == 0) { a = a + 5; printf("%d,%dn", a, &a); } else { a = a –5; printf("%d, %dn", a, &a); } i) Let u, v be the values printed by the parent process, and x, y be the values printed by the child process. Which one of the following is TRUE? a) u = x + 10 and v = y b) u = x + 10 and v != y **c) u + 10 = x and v = y** d) u + 10 = x and v != y	**1**	**1**	**1**	**1**	**1.6.1**
5	Less-privileged mode in the mode of processor execution normally associated with the operating system is often referred to as the …… mode. **a) user mode** b) un-restricted mode c). Both A and B d). None of the above	**1**	**1**	**1**	**1**	**1.6.1**
	Part – B **(5 x 4 = 20 Marks) Instructions: Answer any 5**					
6	With a neat sketch illustrate Operating System as Resource Manager	**4**	**3**	**1**	**3**	**2.6.2**



7	Compare Effects on Resource Utilization with uniprogramming and multiprogrammimng		**4**	**4**	**1**	**2**	**2.6.4**

	Uniprogramming	Multiprogramming
Processor use	20%	40%
Memory use	33%	67%
Disk use	33%	67%
Printer use	33%	67%
Elapsed time	30 min	15 min
Throughput	6 jobs/hr	12 jobs/hr
Mean response time	18 min	10 min

8	Write short notes on Compatible Time-Sharing Systems • One of the first time-sharing operating systems ◦ Developed at MIT by a group known as Project MAC ◦ Ran on a computer with 32,000 *36*-bit words of main memory, with the resident monitor consuming 5000 of that ◦ To simplify both the monitor and memory management a program was always loaded to start at the location of the 5000th word **Time Slicing** ◦ System clock generates interrupts at a rate of approximately one every 0.2 seconds ◦ At each interrupt OS regained control and could assign processor to another user ◦ At regular time intervals the current user would be preempted and another user loaded in ◦ Old user programs and data were written out to disk ◦ Old user program code and data were restored in main memory when that program was next given a turn	**4**	**3**	**1**	**1**	**1.7.1**
9	Summarize the importance of threads • Takes less time to create a new thread than a process • Less time to terminate a thread than a process • Switching between two threads takes less time that switching processes • Threads can communicate with each other without invoking the kernel	**4**	**2**	**1**	**1**	**1.7.1**
10	With a neat sketch relate Tree of Processes in Linux  	**4**	**3**	**1**	**3**	**2.6.2**
11	Explain about process termination. Process executes last statement and then asks the operating system to delete it using the **exit()** system call. ◦ Returns status data from child to parent (via **wait()**) ◦ Process' resources are deallocated by operating system Parent may terminate the execution of children processes using the **abort()** system call. Some reasons for doing so: ◦ Child has exceeded allocated resources ◦ Task assigned to child is no longer required	**4**	**2**	**1**	**1**	**1.7.1**

	◦ The parent is exiting and the operating systems does not allow a child to continue if its parent terminates					