

Solution set of Paper Operating System (2342572401)

Section A

Q1	a.	Mention two mass storage devices. ½ mark each for giving name of any mass storage device	1
	b.	Differentiate between mv and cp commands in Unix/Linux mv moves the file into the desired location and deletes it from original location cp copies the file into the desired location and retains the the copy in the original location	2
	c.	List two main functions of the Operating System. (Any two functions) 2 Marks	2
	d.	What is a thread? What are the benefits of using threads in a programming environment? 1 mark for defining thread ; 1 mark for listing any two benefits – responsiveness, resource sharing, economy, scalability	1+1
	e.	What is a two-level directory structure? At the top level there is master file directory indexed by user name or his/her id and each entry points to the next level in which each user has his/her own user file(UFD) directory. Thus at the second level each directory lists files of the same user (1 mark each for both levels)	2

f.	<p>What is the difference between logical and physical address?</p> <p>Logical address is the address generated by CPU</p> <p>Physical address is the actual address in the memory unit, which is loaded in Memory Address Register to fetch/write data in Main memory.</p>	2
g.	<p>What is dual mode of operation in the context of an Operating system?</p> <p>The OS operates in two modes – kernel/supervisor/system/privileged mode and user mode (2 mark)</p> <p>This is done in order to ensure the proper execution and protection of OS as well as different users programs running in a multiprogramming environment. (1 mark)</p>	3
h.	<p>Give an example for absolute path and relative path in a directory?</p> <p>Absolute and relative path - Absolute path gives the complete path to traverse to the desired location starting from the root directory, whereas relative path gives the path to traverse from current location to the desired location. (1 mark)</p> <p>Example for each (1 mark each)</p>	3
i.	<p>Briefly describe the working of Shortest Remaining Time First scheduling algorithm.</p> <p>Explanation should include :</p> <p>The preemptive variation of SJF. Whereas SJF schedules jobs based on the length of the next CPU burst required by the process, the SRTF assigns CPU to the process with the least time remaining to complete the execution. the SRTF preempts the currently executing process. (3 marks)</p>	3

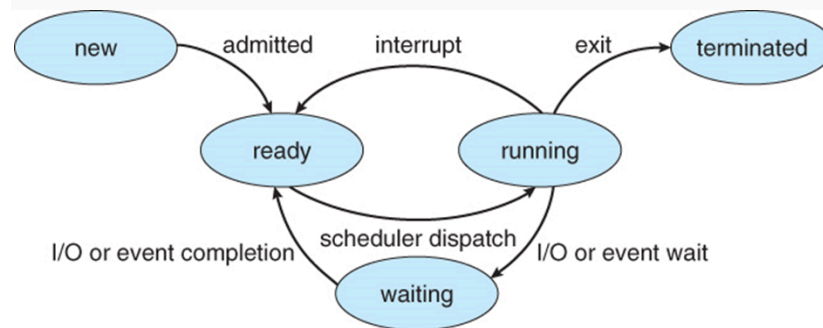
	j.	<p>What is meant by the term ‘virtual address space’ of a process.</p> <p>Virtual address space refers to the logical view of how a process is stored in the memory, This allows a large virtual view to the programmer whereas the physical address space available is much smaller. As a result the programmer does not need to worry about the amount of physical memory available. (2 mark)</p>	2
	k.	<p>List two system calls for each of the following:</p> <ul style="list-style-type: none"> i. Process control ii. Device management. <p>Process control - any two - end/abort, load, execute, create/terminate process. get/set process attributes, wait/signal event, allocate/free memory (2 mark)</p> <p>Device management - any two of - request/release device. read/write/reposition, get/set device attributes, logically attach/detach devices (2 mark)</p>	2+2
	l.	<p>What is a Process Control Block (PCB)? Elaborate the information stored in it.</p> <p>Each process is represented by a PCB. The purpose of PCB is to store all the information associated with a process, which is updated as when as state or any other attribute of the process changes (1 mark)</p> <p>Information stored - Process state, Program counter, CPU registers, CPU scheduling info, Memory management information, accounting info, I/O status information (any 3 with explanation)</p>	1+3

Section B

Q2	a.	<p>In a multiprogramming uniprocessor system, how many processes can be in running state and how many processes can be in ready state at a particular time?</p> <p>In a multiprogramming uniprocessor system, only one process can be in the running state at any given time. This is because there is only a single CPU core available (1 mark)</p> <p>Whereas the number of processes that can be in the ready state is not fixed and can vary. It is determined by the degree of multiprogramming that the system is designed to handle. (1 mark)</p>	2
	b.	<p>What is the challenge faced by the designers of Operating System while implementing Shortest Job First scheduling algorithm.</p> <p>Main challenge for implementing the Shortest Job First (SJF) scheduling algorithm is</p> <p>Predicting Execution Time: One of the main challenges is the need for accurate knowledge of the burst time for each process.</p> <p>(2 marks)</p> <p>Or if any other challenges is written then mark accordingly</p>	2
	c.	<p>A process goes through various states from its creation to termination. Illustrate the states of a process using a diagram and explain briefly different states.</p> <p>A process in an operating system goes through various states from its creation to termination. Here's a simplified explanation along with a diagram:</p> <p>1. New: The process is being created.</p>	3+2

- | | | |
|--|---|--|
| | <p>2. Ready: The process is waiting to be assigned to a processor.</p> <p>3. Running: The process is executing instructions on the processor.</p> <p>4. Waiting: The process is not executing because it is waiting for some event to occur (like I/O completion).</p> <p>5. Terminated: The process has finished execution and is being removed from the system.</p> | |
|--|---|--|

(3 marks)



2 marks
for
(diagram)

	d.	<p>The following processes arrive in order P1, P2, P3, P4, P5 at time 0. Draw Gantt chart showing the execution of these processes using SJF(Shortest Job First). Calculate the average turnaround time and average wait time.</p> <table><tr><th>Process</th><th>Burst Time</th></tr><tr><td>P1</td><td>3</td></tr><tr><td>P2</td><td>2</td></tr><tr><td>P3</td><td>7</td></tr><tr><td>P4</td><td>4</td></tr><tr><td>P5</td><td>5</td></tr></table> <p>(3 marks Gantt chart)</p> <p>(1.5 marks for ATT)</p> <p>(1.5 marks for AWT)</p> <div><p>Q2d. SJF</p><table><tr><td>P1</td><td>P2</td><td>P3</td><td>P4</td><td>P5</td></tr><tr><td>3</td><td>2</td><td>7</td><td>4</td><td>5</td></tr></table><p>Using SJF</p><table><tr><td>P2</td><td>P1</td><td>P4</td><td>P5</td><td>P3</td></tr><tr><td>2</td><td>3</td><td>4</td><td>5</td><td>7</td></tr></table><p>Gantt chart</p><table><tr><td>P2</td><td>P1</td><td>P4</td><td>P5</td><td>21</td></tr><tr><td>0</td><td>2</td><td>5</td><td>9</td><td>14</td><td>21</td></tr></table><p>ATT = $\frac{2 + 5 + 9 + 14 + 21}{5} = \frac{51}{5} = 10.2$</p><p>AWT = $\frac{0 + 2 + 5 + 9 + 14}{5} = \frac{30}{5} = 6$</p></div>	Process	Burst Time	P1	3	P2	2	P3	7	P4	4	P5	5	P1	P2	P3	P4	P5	3	2	7	4	5	P2	P1	P4	P5	P3	2	3	4	5	7	P2	P1	P4	P5	21	0	2	5	9	14	21	3+3
Process	Burst Time																																													
P1	3																																													
P2	2																																													
P3	7																																													
P4	4																																													
P5	5																																													
P1	P2	P3	P4	P5																																										
3	2	7	4	5																																										
P2	P1	P4	P5	P3																																										
2	3	4	5	7																																										
P2	P1	P4	P5	21																																										
0	2	5	9	14	21																																									
Q3	a.	<p>What is the Unix Kernel?</p> <p>The Unix kernel is the core component of the Unix operating system (OS). It acts as the intermediary between</p>	2																																											

		<p>the computer's hardware and the software applications that run on the OS. (2 marks)</p> <p style="text-align: center;">OR</p> <p>The kernel has several key responsibilities, including:</p> <ol style="list-style-type: none"> 1. Memory management: 2. Process management: 3. Device drivers: 4. System calls and security: etc. <p>(2 Marks)</p>	
	b.	<p>What is the purpose of pipes in shell script? Give an example</p> <p>Pipe is a command in Unix/Linux to connect command in sequence.</p> <p>Its symbol is . (1 Mark)</p> <p>Any example (1 mark)</p>	1+1
	c.	<p>Write a shell script to input a number N and print all the prime numbers between 1 and N. The program should be well documented.</p> <p>1 mark for input from the user</p> <p>3 mark for correct logic</p> <p>1 mark for documentation</p>	5

	d.	<p>A file named abc.txt contains following data</p> <p>Amit</p> <p>Bharat</p> <p>Sanjeev</p> <p>Sanjay</p> <p>Dhanraj</p> <p>Inderjit</p> <p>Sankalp</p> <p>What is the output of the following commands:</p> <ol style="list-style-type: none"> grep [s] abc.txt grep san [^j] cut -c 3,5 abc.txt <p>(i)</p> <p>No output (as 's' not 'S' is given) (3 marks)</p> <p>or</p> <p>Sanjeev</p> <p>Sanjay</p> <p>Sankalp</p> <p>(2 marks)</p> <p>(ii)</p> <p>Syntax error (2 marks) + reason for syntax error (1 mark)</p> <p>(as correct command is grep san[^j] abc.txt cut -c 3,5)</p>	3*2
Q4	a.	Give Unix/Linux commands to remove a file and remove a directory.	2

		<p>rm file name // for removing file (1 mark)</p> <p>rmdir directory name //for deleting directory (1mark)</p>	
	b.	<p>Give the advantage of layered design of the Operating System</p> <p>There are several advantages to layered design the of an operating system :</p> <ul style="list-style-type: none"> i. Modularity : This design promotes modularity as each layer performs only the tasks it is scheduled to perform. ii. Easy debugging : As the layers are discrete so it is very easy to debug. iii. Easy update : A modification made in a particular layer will not affect the other layers. iv. No direct access to hardware : v. Abstraction : <p>(any one of the advantage - 2 marks)</p>	2
	c.	<p>What is an API, what are the advantages of a programming environment which uses API.</p> <p>An API stands for Application Programming Interface, is a set of rules or protocols that enables software applications to communicate with each other. (1 mark)</p> <p>The advantages of using APIs in a programming environment are numerous:</p> <ol style="list-style-type: none"> 1. <u>Faster Development:</u> 2. <u>Cost Efficiency:</u> 3. <u>Extended Functionality:</u> 	1+1

		<p>4. <u>Microservices Architecture:</u></p> <p>5. <u>Security:</u></p> <p>6. <u>Revenue Generation:</u></p> <p>7. <u>User Satisfaction:</u></p> <p>(any two 0.5 each)</p>	
	d.	<p>Explain the difference between system call and system program.</p> <p>The system call creates an interface between the user program and the services of the operating system. (1mark)</p> <p>The system program provides convenient environment for program development and execution.</p> <p>(1 mark)</p>	2
	e.	<p>Each system call will require passing some parameter(s) to the Operating System. What are the three parameter passing techniques?</p> <p>1. Passing Parameters in Registers: This is the most common method where parameters are passed directly in the CPU registers.</p> <p>2. Block or Table in Memory: If there are more parameters than registers available, they can be stored in a block or table in memory.</p> <p>3. Stack Method: Parameters can also be pushed onto a stack in the user space, and then the operating system pops them off the stack in the kernel space.</p> <p>(3 marks)</p>	3

	f.	<p>List any 4 categories of system programs.</p> <ol style="list-style-type: none"> 1. File Management - Tools that handle the creation, deletion, copying, renaming, and management of files and directories. 2. Command Line Interfaces (CLI) - Text-based interfaces that allow users to directly issue commands to the operating system. 3. Device Drivers - Software that enables communication between the operating system and hardware devices. 4. Program Loading and Execution - Loaders that bring programs into memory and prepare them for execution. 5. Data Management - Systems that manage databases and other structured data formats. 6. Communication Software - Programs that manage data transmission between computers, such as network and internet protocols. 7. <u>Utility Programs</u> <p>Any 4 (4 marks)</p>	4
Q5	a.	<p>What is the purpose of a page table in memory management?</p> <p>To store mappings of physical addresses and virtual addresses</p> <p>Or</p> <p>Helps in address translation between physical addresses and virtual addresses</p> <p>(1 mark)</p>	1

	b. .	<p>A memory has a page size of 1 KB. determine the page number and offset for the following addresses</p> <p>i. 3085</p> <p>ii. 4205</p> <p>iii. 65000</p>	3*2
		<p>(2 Marks each part)</p> <p>i. Page number: $3085/1024 = 3$</p> <p>Page offset : $3085\%1024 = 13$</p> <p>ii. Page number: $4205/1024 = 4$</p> <p>Page offset : $4205\%1024 = 109$</p> <p>iii Page number: $65000/1024 = 63$</p> <p>Page offset : $65000\%1024 = 488$</p>	
	c. .	<p>Given memory partition sizes of 200KB, 600KB, 100KB, 300KB and 450 KB. How would the processes of sizes 330KB, 250KB, 500KB and 350 KB be placed in the memory for first fit, best fit and worst fit algorithms. Calculate internal and external fragmentation in all three algorithms.</p>	3*2+2
		<p>Taking Fixed Partition</p> <p>First fit (1.5 marks)</p> <p>600 kB -> 330KB</p>	

		<p>300 KB -> 250 KB</p> <p>450 KB-> 350 KB</p> <p>Left out processes (500 KB)</p> <p>Best Fit (2 marks)</p> <p>450 KB-> 330 KB</p> <p>300 KB->250KB</p> <p>600 KB->500KB</p> <p>Left out processes (350 KB)</p> <p>Worst Fit (1.5 marks)</p> <p>600 KB-> 330 KB</p> <p>450 KB->250KB</p> <p>Left out processes (500 KB, 350 KB)</p> <p>(3 mark : 0.5 marks for each correct))</p> <p>First fit : Internal Fragmentation = 420 KB</p> <p>External Fragmentation = 300KB</p> <p>Best fit : Internal Fragmentation = 270 KB</p> <p>External Fragmentation = 300 KB</p> <p>Worst fit:</p> <p>Internal Fragmentation = 470 KB</p> <p>External Fragmentation = 600 KB</p> <p>OR</p> <p>Taking Variable partition</p> <p>First fit (1.5 marks)</p>	
--	--	--	--

		<p>600 kB -> 330KB (270 KB partition)</p> <p>270 KB -> 250 KB (20 KB partition)</p> <p>450 KB-> 350 KB (100 KB partition)</p> <p>Left out processes (500 KB)</p> <p>Best Fit (2 marks)</p> <p>450 KB-> 330 KB (120 KB partition)</p> <p>300 KB->250KB (50 KB partition)</p> <p>600 KB->500KB (100 KB partition)</p> <p>Left out processes (350 KB)</p> <p>Worst Fit (1.5 marks)</p> <p>600 KB-> 330 KB (270 KB partition)</p> <p>450 KB->250KB (200 KB partition)</p> <p>Left out processes (500 KB, 350 KB)</p> <p>(3 mark : 0.5 marks for each correct answer)</p> <p>First fit : Internal Fragmentation = 0 KB</p> <p>External Fragmentation = 720KB</p> <p>Best fit : Internal Fragmentation = 0 KB</p> <p>External Fragmentation = 570 KB</p> <p>Worst fit:</p> <p>Internal Fragmentation = 0 KB</p> <p>External Fragmentation = 1070 KB</p>	
--	--	--	--

Q6	a.	<p>For a paging environment in the main memory, the logical address has two parts - page number and offset. The size of the page frame is 2K. What should be the size of the offset (in bits)</p> <p>Page size = 2K</p> <p>= $2 * 2^{10} = 2^{11}$</p> <p>11 bits will be the offset</p> <p>1 Mark (concept) + 2 Marks(for correct answer)</p>	3
	b.	<p>What is preemptive and non-preemptive process scheduling? Name two algorithms for each and justify your categorization into preemptive and non-preemptive.</p> <p>Definition Preemptive and non-preemptive process scheduling</p> <p>2 marks</p> <p>Example of Preemptive process scheduling: RR, SRTF</p> <p>1Marks + 1 marks(Explanation)</p> <p>Example of non-Preemptive process scheduling: FCFS, SJF</p> <p>1 Marks + 1 Marks(Explanation)</p>	2+4
	c.	<p>What are the three methods to access data in a file? Give comparison in terms of efficiency and ease of implementation.</p> <p>Data in files can be accessed using different access methods - sequential access, direct/relative access, indexed/ indexed sequential access (1.5 mark)</p>	6

		<p>Sequential access is the easiest to implement. The number of access required will depend on where the data is stored. To reach records towards the end, a lot many accesses are required. It is slower than other methods on an average (1.5 marks)</p> <p>Implementation of Direct access method is more complicated than a sequential method. The success depends on the algorithm/hashing technique used to hit the desired location. Only one access is required to get data. Direct access is faster on an average. (1.5 mark)</p> <p>Implementation of indexed method is more complicated than sequential access. Indexed access is faster than sequential access and comparable to Direct access. Access is made to the block that contains the data and the next access(s) to hit at the exact target. Requires two accesses. (1.5 marks)</p>	
Q7	a.	<p>Draw a labelled diagram showing structure of a magnetic disk.</p> <p>Diagram (2) + labelling (2)</p>	4
	b.	<p>What is demand paging? What is Page fault? What are the steps to handle page fault?</p> <p>Definition demand paging (1 Mark)</p> <p>Definition Page fault (1 Mark)</p> <p>Steps to handle page faults (3 Marks)</p> <ul style="list-style-type: none"> a. Identification of page fault b. Virtual page determination c. Handling the fault <p>(brief explanation should be there)</p>	1+1+3

	<p>c.</p> <p>What is transfer rate, seek time and rotational latency in a disk? A disk has average seek time of 12 ms and spins at 7200 RPM (revolutions per minute). Data transfer rate for the disk is 4MB/sec. Calculate the time required to transfer 8KB data to the disk. Assume there are no other time overheads/delays.</p> <p>Definitions of transfer rate, seek time and rotational latency in a disk (3 mark)</p> <p>Rotational delay = 4.166 ms</p> <p>Calculation of rotational delay</p> <p>7200 RPM implies 120 Rotations per second</p> <p>1 rotation takes 1/120 sec, time for half rotation = 1/240s=4.166 ms (1 Mark)</p> <p>Transfer time = 8KB/4MB/s =2ms (approx) or 1.95 (exact)</p> <p>(1 mark)</p> <p>Total Transfer time = seek time + avg rotational delay + data transfer time = 12 + 4.166 + 2 = 18.166 ms (1 Mark)</p>	3+3
--	--	-----