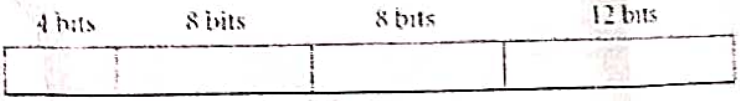
KONERU LAKSHMAIAH EDUCATION FOUNDATION
(Deemed to be University, Estd. u/s. 3 of UGC Act 1956)

B.Tech - Odd Sem End Semester Exam
Academic Year:2020-2021

19CS2106S - OPERATING SYSTEMS DESIGN - S

Set No: 3

Time: Max.Marks: 100

S NO	Answer All Questions	Choice	Options	Marks	CO
1	The contents of a directory is the mapping from names (strings of characters) to inode (e.g. integers interpreted as an index into a table). How would you implement this? Give an algorithm. Recall that most file names are short, but you need to handle arbitrarily long names efficiently. Also, you need to handle dynamic insertions and deletions from the directory. Assume a full path /a/b/c is given. To find this file, how many disk blocks you need to access. Illustrate with a diagram. How does the system find the root directory itself?	choice Q-2		10Marks	CO1
2	(i) Explain how the byte offset into a Unix file can be converted into the pair (disk block id, byte offset). (ii) Consider the organization of a UNIX file as represented by inode scheme. Assume that there are 12 direct block pointers, and a singly, doubly, and triply indirect pointer in each inode. Further, assume that the system block size and the disk sector size are both 8K. If the disk block pointer is 32 bits, with 8 bits to identify the physical disk and 24 bits to identify the physical block, then (a) What is the maximum file size supported by this system? (b) What is the maximum file system partition supported by this system? (c) Assuming no information other than that the file inode is already in main memory, how many disk accesses are required to access the byte in position 13,423,956?			10Marks	CO1
3	Answer the following	choice Q-4		15Marks	CO2
3.A.	Write a system program for implementation of File server, a client-server application in which the client sends the server a pathname and the server returns the contents of that file to the client using two pipes			8Marks	CO1
3.B	(i) How to allocate a block? List xv6 kernel code functions/algorithms and files directly or indirectly used for the execution of $ echo x > a (ii) When can inode and blocks be freed? List xv6 functions called for the execution of $ rm a			7Marks	CO1
4	Answer the following			15Marks	CO1
4.A	Explain how hard links and soft links differ with respective to i-node allocations. How do you unlink an opened file? The file system can be viewed as graph with i-nodes as nodes and directory entries as links. Using link can a program create cycles in the UNIX v6 i-node graph? If so, show a sequence of commands that creates a cycle. If not, how does v6 prevent cycles?			8Marks	CO1
4.B.	How read system calls work. Give algorithm. What are its input parameters and returns information? Describe xv6 functions. filealloc, filedup, and fileclose			7Marks	CO1
5	The traditional UNIX scheduler is a priority-based round robin scheduler (also called a multi-level round robin scheduler). How does the scheduler go about favoring I/O bound jobs over long-running CPU-bound jobs? For the given list of processes and service time: P1 120, P2 60, P3 180, p4 50, P5 300. Answer the following. (i) Draw a Gantt chart that shows the completion times for each process using first-come, first served CPU scheduling. (ii) Draw a Gantt chart that shows the completion times for each process using shortest-job-next CPU scheduling. (iii) Draw a Gantt chart that shows the completion times for each process using round-robin CPU scheduling with a time slice of 60. Calculate the average waiting time and turnaround time	choice Q-6		10Marks	CO2
6	In xv6, explain swtch function that does the job of switching between two contexts, and old one and a new one. The swtch function is called at two places what are they. Discuss proc context			10Marks	CO2
7	Answer the following	choice Q-8		15Marks	CO2
7.A	List the various sections of the disk image of an executable file in UNIX. In xv6, explain the algorithm for a system call that makes a process to overwrite itself with another executable image			8Marks	CO2
7.B	Write a system program that accepts two small numbers (< 50) as arguments and then sums the two in a child process. The sum should be returned by the child to the parent as its exit status, and the parent should print the sum. Illustrate how parent and child processes share files that are open before the child process is created			7Marks	CO2
8	Answer the following			15Marks	CO2
8.A.	In xv6, explain the purpose of init.c. How shell works give an algorithm? Write an algorithm for the Clock Handler			8Marks	CO2
8.B	Explain the design of the algorithms: intland, syscall. In Xv6, how system call works? List the data structures, functions, and files that are to be manipulated. How do system calls relate to the OS and to the concept of dual-mode (kernel mode and user mode) operation?			7Marks	CO2
9	(i) Consider a three-level page table organization as shown in the figure below. If a program is 4 Giga bytes, what is the total space needed for its page table (that is, the total space needed by directories and partial page tables)? (ii) Can you imagine the page size that is not of the power of 2? What are the disadvantages of such a page size? When a process exits, all its pages may not be placed immediately on the memory free list. Explain this behaviour 	choice Q-10		10Marks	CO3
10	(i) A process references five pages, A, B, C, D, and E, in the following order: A, B, C, D, A, B, E, A, B, C, D, E. Assume that the replacement algorithm is first-in-first-out and find the number of page transfers during this sequence of references starting with an empty main memory with three-page frames. Repeat for four-page frames. (ii) The main memory management policy is a page-based virtual memory and the replacement algorithm is the clock algorithm. If the R bit of a page that has just been moved to main memory is not set to one, provide an example in which this page is removed from main memory before being actually accessed			10Marks	CO3
11.	Answer the following	choice		15Marks	CO3

11 A	Give examples of why processes might wish to share parts of their address spaces. What kind of memory management hardware would support this sharing? How can memory protection be enforced when sharing is allowed?	Q-12			
11 B	Please give your version of the realloc() call using system calls to allocate the swap space for the process to be swapped out. You can descript it with a C-like algorithm			8Marks	CO3
12	Answer the following			7Marks	CO3
12 A	List Fields of region table. When does kernel pushes a context layer? With an algorithm to allocate and initialize text region. Illustrate relationship of inode table and region table for shared text			15Marks	CO3
12 B	For a given virtual address of a segmented system in binary number, write and explain segmentation control flow algorithm using segment registers x bits y bits Segment no. | Byte offset in Segment — Virtual address: address field of an instruction Maximum no. of Segments is 2^x Maximum Segment size is 2^y			8Marks 7Marks	CO3 CO3
13	A 'big reader' lock provides multiple reader, single writer semantics optimized for workloads in which updates are rare. Implement 'big reader' software locks and solve readers and writer's problem	choice Q-14		10Marks	CO4
14	What three segments are usually found in the memory allocated to a process? Distinguish between an activation record and a stack frame. What is contained in a stack frame? How long does a stack frame last? Illustrate backtrace and Depict stack frames for a sample C program which call 3 user defined functions and malloc function			10Marks	CO4
15	Answer the following	choice Q-16		15Marks	CO4
15 A	Many systems classify library functions as thread-safe or thread-unsafe. What causes a function to be unsafe for use by a multithreaded application? Illustrate the implementation of concurrent linked list that only allows one thread to access the entire list at any instant			8Marks	CO4
15 B	Write a program to demonstrate deadlock using semaphores. Show how Dijkstra's P and V semaphore operations could be implemented using Lock with Queues, Test-and-set, Yield, and Wakeup			7Marks	CO4
16	Answer the following			15Marks	CO4

16 A	Illustrate system V semaphores data structures by giving algorithm for semop(). Initial values of semaphores x and y are 0 and 1, respectively. What is the typical output that is generated by processes P1 and P2 when executing concurrently?
	P1
	...
	while(1){
	P(x);
	print ("A");
	V(y);
	}
	P2
	...
	while(1){
	P(y);
	print("B");
	V(x);
	}
				8Marks	CO4

16 B	Considering a system with five processes P1 through P5 and three resources of type A, B, C. Resource type A has 10 instances, B has 5 instances and type C has 7 instances. Suppose at time t0 following snapshot of the system has been taken: 1.what will be the content of the need matrix? 2. is the system in a safe state? if yes then what is the safe sequence? 3. what will happen if process p3 requests one additional instance of resource type C and two instances of resource type A ?			7Marks	CO4

Process	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
P1	2	0	2	6	5	3	1	0	1
P2	0	1	1	2	2	1			
P3	2	0	0	2	0	1			
P4	3	3	1	3	0	0			
P5	2	1	2	4	2	2			

[object HTMLDivElement]