College of Engineering, Pune
End Semester Exam

T.Y. B. Tech. (Computer Engineering) (CT-315)- (Operating Systems)

Day & Date- Friday, 26th April 2013 **Time: -** 2 pm to 5 pm

Maximum Marks: 50 **Duration** – 3 hrs

Instructions:

1. All questions are compulsory. Numbers in the right-most column indicate marks.
2. Write indented and commented code.
3. This is an open book exam. Any amount of printed matter is permitted but exchange is not permitted.
4. Use of electronic gadgets (including calculators) of any kind is *not* permitted.
5. Make suitable assumptions if required, and state them in your answer.
6. Write precise answers. *Irrelevant parts of answers will invite penalty.*
7. All diagrams must be neat and clean. You may use pencil for drawing diagrams.

Q 1		**Write answers in few words or one line as asked. Draw diagrams whenever possible.**	1 * 10
	A	What is the maximum possible size of a file on ext2 layout? Why?	
	B	For a 2 level hierarchical paging scheme on a 64 bit processor, where the 64 bits are split as 16-16-32 with 32 bits for offset and 16 bits each for the page directories, what is the size of the second level page directory with 8 byte page entries?	
	C	Write names of any 4 types of I/O buses.	
	D	Write implementation of a spin lock using compare and swap instruction. (Assume that the compare and swap instruction is given to you as a function).	
	E	Draw diagram of memory allocations done by a kernel buddy allocator, with 128 bytes of memory, after following requests: 3 bytes, 13 bytes, 51 bytes	
	F	`add A, B, C, D, E` Given the above instruction which adds memory addresses A, B, C, D and stores the result in memory location E; the size of the instruction is 6 bytes on a 32 bit machine; What is the maximum number of page faults that can occur while executing the instruction?	

	G	What is the output of this program?
		`#include <stdio.h>`
		`int main() {`
		`  static int i = 3;`
		`  if(i > 0) {`
		`  i--;`
		`  main();`
		`  execl("/bin/echo", "/bin/echo", "yes", NULL);`
		`  } else {`
		`  execl("/bin/echo", "/bin/echo", "no", NULL);`
		`  }`
		`}`	

	H	What is the benefit of spinlocks on multiprocessor systems against uniprocessor systems?	
	I	Write names of any 3 virtualisation software.	
	J	Hard links share the inode number, while soft links do not. Why?.	
Q2		**Write all answers in maximum 2-3 sentences each.**	2 * 10
	A	Which of the following disk scheduling algorithms is truly fair. Why? FCFS, SCAN, C-SCAN.	
	B	Assume that you are monitoring the rate at which the pointer in the clock algorithm (which indicates the candidate page for replacement) is moving. What can you say about the following behaviour: a) Pointer is moving fast b) Pointer is moving slow	
	C	Find total number of page faults (Including initial page faults) for the reference string	

		2 1 0 1 3 4 2, with 3 page frames using LRU replacement.	
	D	Write a program using pipe and shared memory such that process A reads x, y from user and sends x, y to process B using pipe and processs B sends back x^y (x power y) to process A using shared memory and then process A prints it.	
	E	State True/False with reason: The file /proc/filesystem can not be deleted.	
	F	State True/False with reason: Unlike dynamic loading, dynamic linking requires some support from operating system.	
	G	Which of the given sequences is not possible and why (do not assume any priority for interrupts) 1) Process A is running, I/O complete interrupt arrives, Timer interrupt arrives, scheduler runs, I/O interrupt handler runs, Proces B is running. 2) Process A is running, Timer interrupt arrives, scheduler runs, Proces B is running, I/O complete interrupt arrives, I/O interrupt handler runs, 3) Process B is running, Timer interrupt arrives, I/O complete interrupt arrives, I/O interrupt handler runs, scheduler runs, Proces A is running, 4) Process A is running, Proces B is running, I/O complete interrupt arrives, I/O interrupt handler runs, Timer interrupt arrives, scheduler runs,	
	H	Suppose a user renames a file. Assuming ext2 layout, *only* list all the data structure on disk which can change as a result of this (no need to explain, just listing is sufficient).	
	I	Write one difficult question on operating systems.	
	J	Write a kernel module, with an integer module parameter available with read-write permissions. The module prints square of the number on module-exit.	
Q3	**A**	Assuming that you have been given the `ext2_super_block *sb` pointer and `ext2_group_desc gd[] array` structures in global variables, and a file descriptor `fd` to read the device file, write two functions which do the following a) verify that total number of free inodes in superblock equals the sum of free inodes in all group descriptors. b) Given an inode number of a soft link file, prints the name of the file which it points to.	**5**
	B	Write complete implementation of a read-write lock with readers always getting preference over writers. Assume that you have been given the Condition data type.	**5**
	C	Consider a system running 10 I/O bound and 1 CPU bound task. Assume that the I/O bound tasks issue an I/O every 1 ms and each I/O completes in 10 ms. Contxt switch overhead is 0.1ms. All processes are long running tasks. What is the CPU utilisation in round robin scheduling (a) with time quntum of 1 ms (b) with time quantum of 10 ms?	**5**
	D	Consider a modified dinining philosophers problem, where all the chopsticks are kept in the center of the table. Each philospher still has to pick up *any* two sticks and eat. However request can be made only for one chopstick at a time. Assume that there are n philosophers and each chopstick is protected by a semaphore. Let's say chopsticks are named C_1 .... C_n and spinlocks are named S_1 .... S_n. Assume existence of a function *Available(S)* which tells whether semaphore S is available or not. Also use the function *Try_Down(S)* Which tries to do a *Down()* on a semaphore and returns with failure if that is not possible. Write code showing how each philospher will use semaphores and can eat without causing a deadlock.	**5**