

SIXTH SEMESTER EXAMINATION-2006

OPERATING SYSTEM

Full Marks-70

Time : 3 Hours

Answer question No. 1 which is compulsory and any five questions from the remaining questions.

The figures in the right hand margin indicate full marks for the questions.

1. Answer the following questions : [2 X10]
- (a) What is an Operating System ? What is the advantage of Multiprogramming ?
 - (b) Define the essential properties of batch system and interactive system.
 - (c) What do you mean by Distributed System ? Why are distributed systems desirable ?
 - (d) Define and differentiate between Interactive computer system and Time-sharing systems.
 - (e) What is a process ? What is a PCB ?
 - (f) Defining and differentiate between swapping and spawning.
 - (g) Why should not user be allowed to set the interrupting clock ?
 - (h) Give several reasons why dead line scheduling is complex.

- (i) SSTF is some times referred as a cylinder oriented disk-scheduling scheme. Explain why.
 - (j) Distinguish between scheduling policy and scheduling mechanism.
2. (a) When do page faults occur ? Describe the actions taken by the operating system when a page fault occurs. [5]
- (b) Given memory partitions of 100K, 500K, 200K, 300K and 600K (in order), how would each of the First-fit, Best-fit and Worst-fit algorithms place processes of 212K, 417K, 112K and 426K (in order) ? Which algorithm makes the most efficient use of memory ? [5]
3. (a) Discuss the goal of FETCH STRATEGY and PLACEMENT STRATEGY in the context of virtual storage system with paging. [5]
- (b) State three advantages of placing functionality in a device controller, rather than in the kern. State three disadvantages. [5]
4. (a) Consider a demand-paging system with the following time-measured utilizations :
- CPU utilization 20%
- Paging disk 97.7%
- Other I/O devices 5%
- Which (if any) of the following will (probably) improve CPU utilization ? Explain your answer.
- (i) Install a faster CPU
 - (ii) Instal a bigger paging disk
 - (iii) Increase the degree of multiprogramming.

- (iv) Decrease the degree of multiprogramming. [5]
- (b) Suppose that a scheduling algorithm (at the level of short-term CPU scheduling) favours those processes that have used the least processor time in the recent past. Why will this algorithm favour I/O-bound programs and yet not permanently starve CPU-bound programs? [5]
5. (a) What resources are used when a thread is created? How do they differ from those used when a process is created? [5]
- (b) Consider a system consists of four resources of the same type that are shared by three processes, each of which needs at most two resources. Show that the system is deadlock-free. [5]
6. (a) Consider a system where the virtual memory page size is 2K (2048 bytes), and main memory consists of 4 page frames. Now consider a process which requires 8 pages of storage. At some point during its execution, the page table is as shown below: [5]

Virtual page	Valid	Physical page
0	No	..
1	No	
2	Yes	1
3	No	
4	Yes	3
5	No	
6	Yes	0
7	Yes	2

- (i) List the virtual address ranges for each virtual page.

- (ii) List the virtual address ranges that will result in a page fault.
- (b) What is context switching? Describe the action taken by a kernel to context switch (i) among threads and (ii) among processes. [5]
7. (a) An important class of Linux users would like to run large background jobs that use most of the memory of the machine but at a low priority so that they do not interfere with higher-priority processes that run the user interface code. In beta testing, this does not seem to work: no matter how low the priority of the background process, the user interface becomes unusable sluggish and remains so. How can this be? Explain in 3-4 sentences. Can you suggest a simple way to fix the problem? [5]
- (b) Give an example showing why FIFO is not an appropriate CPU scheduling scheme for interactive user. [5]
8. (a) Suppose three concurrent processes, P1, P2 and P3, executing in a machine with a single processor.
- P1 contains an instruction S1;
- P2 contains an instruction S2;
- P3 contains an instruction S3;

The three processes use a busy waiting semaphore `synch` and `wait () / signal ()` code to ensure that both S2 and S3 execute before S1. Then answer the following : [5

(i) Which process or processes require that `signal (synch)` be inserted appropriately ?

(ii) If `synch` can be initially set to any integer value, positive, negative or zero, what value must `synch` be initialized to ?

(iii) Write out the semaphore code that must be placed near S1 in P1 to ensure that both S2 and S3 execute before S1. Include "S1;" in your answer.

(b) Describe the Linux loadable module approach to the provision of operating system features. [5