PES INSTITUTE OF TECHNOLOGY, BANGALORE  
(Autonomous Institute under VTU, Belgaum)  
100 Ft. Ring road, B.S.K. III Stage, Bangalore-560 085

SEMESTER END EXAMINATION (SEE) B. E. 5th SEMESTER – December, 2010  
CS 302 – OPERATING SYSTEMS

DURATION : 3 HOURS  
Note: ALL QUESTIONS ARE COMPULSORY  
MAX. MARKS : 100

1. a. It is always generally desired to have data and programs to reside in main memory permanently, but rather this arrangement is usually not possible. Give reasons.  
5

1 b. Compare the throughput of C-SCAN with that of SCAN with an example.  
5

1 c. Explain each of the following terms: online, real time, Bit Vector, Access Matrix  
5

1 d. Relate the principle of least privilege to the concepts of user mode, kernel mode and privileged instructions.  
5

2. a. The ability of one process to spawn(produce) a new process is an important capability, but it is not without its dangers. Consider the consequences of allowing a user to run the process given below. Note: fork() is a system call that spawns a child process.

```c
int main() { while(true) { fork() } }
```

i. Assuming that a system allowed such a process to run, what would the consequences be?

ii. Suppose that you as an operating systems designer have been asked to build in safeguards against such processes. We know (from the “Halting Problem” of computability theory) that it is impossible, in the general case, to predict the path of execution a program will take. What are the consequences of this fundamental result from computer science on your ability to prevent processes like the above from running?

iii. Suppose you decide that it is inappropriate to reject certain processes, and that the best approach is to place certain runtime controls on them. What controls might the operating system use to detect processes like the above at runtime?

iv. Would the controls you propose hinder a process’s ability to spawn new processes?

v. How would the implementation of the controls you propose affect the design of the system’s process handling mechanisms?

2 b. Consider a single-processor timesharing system that supports a large number of interactive users. Each time a process gets the processor, the interrupting clock is set to interrupt after the quantum expires. This allows the operating system to prevent any single process from monopolizing the processor and to provide rapid responses to interactive processes. Assume a single quantum for all processes on the system.

i. What would be the effect of setting the quantum to an extremely large value, say 5 minutes?

ii. What if the quantum were set to an extremely small value, say a few processor cycles?

iii. Obviously, an appropriate quantum must be between the values in (i) and (ii). Suppose you could turn a dial and vary the quantum, starting with a small value and gradually increasing. How would you know when you had chosen the ‘right’ value?

iv. What factors make this value right from the user’s standpoint?

v. What factors make it right from the system’s standpoint?

2 c. What features should operating systems incorporate to deal with the possibility that processes could start to wait for an event that might never happen?

2 d. Would an algorithm that performs several independent calculations concurrently (e.g., matrix multiplication) be more efficient if it used threads, or if it did not use threads?
3 a. Give an example of a simple resource deadlock involving three processes and three resources. Draw the appropriate resource-allocation graph.

3 b. Define and discuss
   i. Preemptible resource
   ii. non-preemptible resource
   iii. shared resource
   iv. reentrant code

3 c. Consider the following snapshot of a system:

   | Allocation | Max | Available |
   | A B C D    | A B C D | A B C D |
   | P0 1 1 1 9 | 3 3 3 15 | 3 4 3 3 |
   | P1 11 12 11 11 | 21 22 11 11 |
   | P2 6 7 1 1 | 12 10 3 3 |
   | P3 2 1 2 9 | 4 3 3 10 |
   | P4 4 4 1 1 | 5 4 1 1 |

   Find Rmax

   Applying Bankers algorithm is the system safe?
   Can a request by P2(5,2,0,1) then by p3 (1,2,1,1) be granted immediately?

3 d. State the four necessary conditions for a deadlock to exist. Give a brief intuitive argument for the necessity of each individual condition.

4 a. Show how multilevel feedback queues accomplish each of the following scheduling goals.
   i. favor short processes
   ii. favor I/O-bound processes to improve I/O device utilization.
   iii. determine the nature of a process as quickly as possible and schedule the process

4 b. In hierarchical memory systems, a certain amount of overhead is involved in moving programs and data between the various levels of the hierarchy. Discuss why the benefits derived from such systems justify the overhead involved.

4 c. As a systems programmer in a large computer installation using a fixed-partition multiprogramming system, you have the task of determining if the current partitioning of the system should be altered.
   i. What information would you need to help you make your decision?
   ii. If you had this information readily available, how would you determine the ideal partitioning?
   iii. What are the consequences of repartitioning such a system?

4 d. A variable-partition multiprogramming system uses a free memory list to track available memory. The current list contains entries of 150KB, 360KB, 400KB, 625KB, and 200KB. The system receives requests for 215KB, 171KB, 86KB, and 481KB, in that order. Describe the final contents of the free memory list if the system used each of the following memory placement strategies First Fit, best fit, worst fit.

5 a. Consider a pure paging system that uses 32-bit addresses (each of which specifies one byte of memory), contains 128MB of main memory and has a page size of 8KB.
   i. How many page frames does the system contain?
   ii. How many bits does the system use to maintain the displacement, d?
   iii. How many bits does the system use to maintain the page number, p?

5 b. A system receives a series of page references in the following order: 1, 1, 3, 5, 2, 2, 6, 8, 7, 6, 2, 1, 5, 5, 5, 1, 4, 9, 7, 7. The system has five page frames. If all of the frames are initially empty, calculate the number of page faults using FIFO & Optimal algorithms.

5 c. Why does Linux prevent users without root privileges from creating real-time processes?

5 d. Some systems implement file sharing by allowing several users to read a single copy of a file simultaneously. Others provide a copy of the shared file to each user. Discuss the relative merits of each approach.