This Question Paper contains 3 page(s)

Unique Paper Code	: 32341302
Name of the Course	: B.Sc. (H) Computer Science CBCS
Name of the Paper	: Operating System
Semester	: III
Duration	: 3 hours
Maximum Marks	: 75
Year of Admission	: 2015-2018

Attempt any four questions. All Questions carry equal marks.

Q 1. Given memory partitions of 110 KB, 450 KB, 290 KB, 350 KB and 700 KB (in order), how would each of the first-fit, best-fit and worst-fit algorithms place processes of 200 KB, 410 KB, 110 KB and 420 KB (in that order) ? Which algorithm makes the most efficient use of memory?

Consider a paging system with the page table stored in memory. If a memory reference takes 100 nanoseconds how long does a paged memory reference take? If we add a TLB and 70 % of all page-table references are TLB hits, what is the effective memory reference time? (Assume that finding a page-table entry in the TLB takes 10 nanoseconds, if the entry is there.)

Assuming a 2-KB page size, what are the page numbers and offsets for the following address references (provided as decimal numbers): 5378, 15300, and 20500. Show calculation at each step.

Q 2. Consider a system consisting 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 using resource allocation graph. What is the effect of multithreading on deadlock?

At a particular time of computation the value of a counting semaphore is 10. Then 8 *wait ( )* operations and 5 *signal ( )* operations were completed on this semaphore. Determine the resulting value of the semaphore? Show the execution at each step.

A program has just read the fifth record. It next wants to read the eleventh record. How many records must the program read to input the eleventh record using Direct Access and Sequential Access methods? Explain your answer.

Q 3. Consider the following page-reference string: 3,1,2,4,2,3,1,5,6,2,1,2,3,7,6,3,2,1,2,3
How many page faults would occur for Optimal and Least recently used replacement algorithms, assuming three and four frames? Note that all frames are initially empty, and first unique pages will all cost one fault each.

Assume we have a demand-paged memory. The page table is held in registers. It takes 10 milliseconds to service a page fault if an empty page is available or the replaced page is not modified and 30 milliseconds if the replaced page is modified. Memory access time is 100 nanoseconds. Assume that the page to be replaced is modified 80 percent of the time. What is the maximum acceptable page-fault rate for an effective access time of no more than 100 nanoseconds?

If the total number of frames in main memory is 120 and there are 4 processes in the system with the demand as 40, 10, 90 and 60 frames, respectively. What will be the number of frames allocated using the equal and proportional allocation strategies?

Q 4. Consider the following set of processes, with the length of the CPU burst times given in milliseconds:

Processes	Burst Time	Priority	Arrival Time
P1	12	3	0
P2	10	4	1
P3	2	2	2
P4	3	1	3
P5	5	2	4

Draw four Gantt charts illustrating the execution of these processes using FCFS, pre-emptive SJF (equal burst length processes are scheduled in FCFS), a pre-emptive priority (small priority number means high priority, equal priority processes are scheduled in FCFS), and a RR (quantum=3) scheduling.
And calculate average waiting and turnaround time for all above mentioned scheduling algorithms.

Which of the following scheduling algorithms could result in starvation: Shortest-job first, Round robin and Priority. Justify your answer.

Q 5. Suppose that a disk drive has 3000 cylinders, numbered 0 to 2999. The drive is currently serving a request at cylinder 140, and the previous request was at cylinder 120. The queue of pending requests, in FIFO order, is

500, 1400, 910, 1700, 940, 1509, 1022, 170, 1300

Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests, using SSTF, C-LOOK and SCAN disk scheduling algorithms?

What will happen if a user program to get stuck in an infinite loop and never return control to the operating system. Explain how operating system deals with such problem?

Q 6. Consider the following code fragment executed by a process. What will be the output of this code? Justify your answer.

```
#include < unistd.h >
 int main ()
 {
     int i ;
     for (i=0; i<8; i++)
          if (i%2 == 0) fork ( ) ;
     return 0 ;
 }
```

List three major activities of an operating system with regard to *File Management* and *Memory Management*. Give justification for your answer.

Explain what may happen if setting the values of Base and Limit registers are not privileged instructions? Also why is it easy to add a new service in microkernel approach?