BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
First Semester 2003-2004

Course Title : OPERATING SYSTEMS Course No *CS C372*
Component : Test I (Regular) *Closed Book Component*
Weightage : 15% Max Marks: 15 Date : 15-09-2003

1. A paged virtual memory system uses a page size of 1024 bytes. Page table entries are 8 bytes long. The maximum virtual address space size of each process is 2 ^ 30 bytes. The size of the physical address space is also 2 ^ 30. What is the minimum number of levels of page tables needed in this system to ensure that each page table will fit entirely within a single frame? Draw a translation diagram that shows how virtual address translation can be performed in the system. Indicate how each address is broken down into components, and indicate the size of each component.
2. Consider a system with the following set of processes (P) , resources (R), and edges (E):

   P={P1,P2,P3,P4,P5,P6}
   R={R1,R2,R3,R4,R5};
   E= {P1→ R2,P2 → R3, P4 → R4, P5 → R5, R1 → P1, R2 → P2, R3 →P3, R3 → P4, R4 → P5, R4 → P6, R5 →P2 }
   Assume that all the instances of all resources have been allocated to processes

   (a) Draw a Resource Allocation Graph (RAG)
   (b) Draw a wait for graph (WFG) using the RAG
   (c) Use the RAG and WFG to find out if a deadlock exist or not
3. 4 batch jobs A through D arrive for execution at the times indicated

Process	Arrival Time	Total CPU Burst Time
A	0	8
B	1	4
C	4	6
D	5	3

The characteristics of each process is given below:
Process A: Totally CPU bound
Process B: Executes on CPU for 2 time units and follows it by I/O for another 2 time units. This cycle repeats
Process C: Totally CPU bound
Process D: Totally CPU bound
For each of the following scheduling algorithms, determine the average waiting and turn around times. Also draw the Gantt charts. Ignore context-switching overhead.

(a) FCFS
(b) RR (T.Quantum = 4)
(c) Preemptive SJF

4. Suppose that pages in a virtual address space are referenced in the following order:

   1 2 1 3 2 1 4 3 1 1 2 4 1 5 6 2 1

   3 empty frames are available. Show the contents of frames after every memory reference.

   (a) Find the lower bound of the number of page faults

   (b) Find the number of page faults for the following replacement algorithms

   i. LRU ii. Second Chance

5. Consider the following test program for an implementation of join. Here when a parent thread calls join on a child thread the parent does one of 2 things 1) if the child is still running the parent blocks until the child finishes. 2) if the child has finished the parent continues to execute without blocking.

```
Void ThreadTest()
{
        Thread *t;
        T=new Thread(“A”,1);
        t->setPriority(10);
        printf(“fee”);
        t->fork(A,0);
        printf(“foe”);
        t->join();
        printf(“fun”);
}
void A(int arg)
{
        Thread *t2;
        T2=new Thread(“B”,1);
        T2->setPriority(20);
        Printf(“foo”);
        T2->fork(B,0);
        Printf(“far”);
        T2->join();
        Printf(“fum”);
}
void B(int arg)
{
        printf(“fie”);
}
```

Assume that the scheduler run threads in round robin with no implicit time slicing (i.e non-preemptive scheduling), priorities are ignored, and threads are placed on queues in FIFO order. What will this program print out

Now assume that the scheduler runs threads according to priority (the high priority value thread on the ready queue will run first and when a thread is added to the ready queue it will preempt the current thread if the new thread has higher priority. If the priority of the main program is 0 then what will the test program print out?