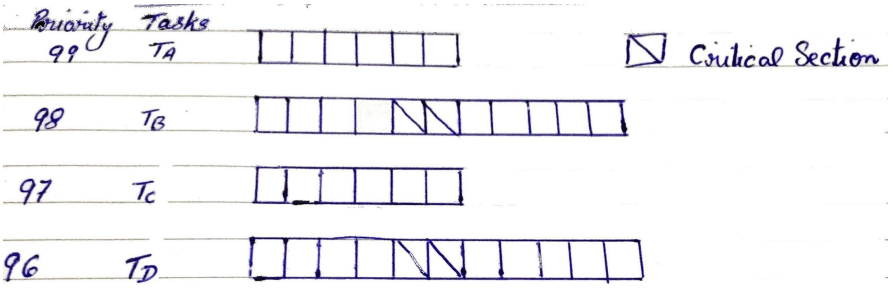
**Comprehensive Exam Operating Systems**
**Jun 2021**
Total marks: 45 marks, Duration: 2 hrs

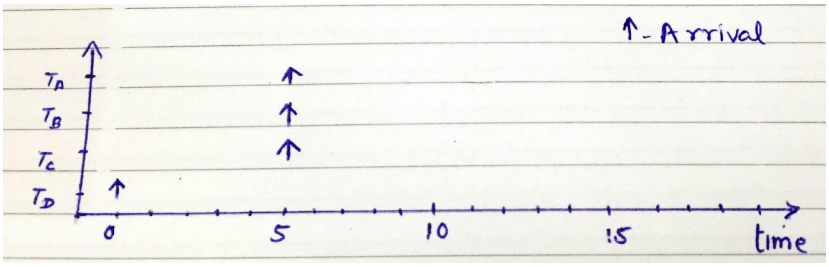
1) Real-time operating systems follow preemptive priority-based scheduling. Each arriving task has an associated deadline and priority. A task with a higher priority preempts a lower priority task. The execution of a task is considered successful only when it finishes within the deadline. Else it is considered a failure. Consider the following set of tasks.

| Task | Arrival Time | Deadline |
|---|---|---|
| TA | 5 | 7 |
| TB | 5 | 20 |
| TC | 5 | 70 |
| TD | 0 | 200 |




The diagram above shows the distribution of time before, during and after the critical section for each task. The number of boxes show the total duration of each task.
a) In the time-line shown below, depict the complete execution of all the processes and show which processes miss the deadline (extend the time axis as required). [Hint: priority inversion].

**[8m]**




b) Suppose that priority inheritance is implemented in the above system, wherein the *priority level of the lower priority process is raised to that of the highest process that is waiting for the lock. After the process releases the lock, its priority is lowered again.* Show the time-line for the execution of the processes under the priority inheritance scheme.

**[7m]**

2)

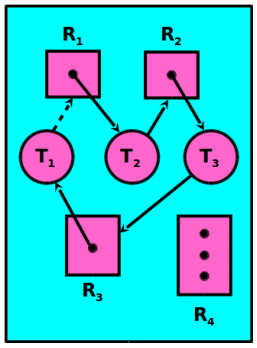
```
int func(int level){
   int i=0;

   for(i=0; i<level; i++) {
      if (fork() && fork())
         break;
   }
   return 0;
```

}
Write an expression of the total number of processes created (including the current process) in terms of level. **[4m]**

3) Threads share the heap but not the stack. What would happen if the threads used the same stack for execution? **[2m]**

4) Consider the resource allocation graph shown. Assume that the resources shown correspond to variables in a database. Assume that programs T1 and T2 are read-only and T3 is a write-only program. If T1 makes a request for Resource R1 (shown as dotted arrow) does there exist a deadlock? Analyse with respect to the 4 necessary and sufficient conditions for deadlock to occur. **[4m]**




**5)** A CPU generates 32-bit virtual addresses. The page size is 4 KB. The processor has a translation look-aside buffer (TLB) which can hold a total of 128 page table entries and is 4-way set associative. What will be the minimum size of the TLB tag? **[3m]**

6) A computer uses 46–bit virtual address, 32–bit physical address, and a three–level paged page table organization. The page table base register stores the base address of the first–level table (T1), which occupies exactly one page. Each entry of T1 stores the base address of a page of the second–level table (T2). Each entry of T2 stores the base address of a page of the third–level table (T3). Each entry of T3 stores a page table entry (PTE). The PTE is 32 bits in size. The processor used in the computer has a 1 MB 16 way set associative virtually indexed physically tagged cache. The cache block size is 64 bytes. What is the size of a page in KB in this computer? **[8m]**

7) A system uses 3 page frames for storing process pages in main memory. It uses the First in First out (FIFO) page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-

4 , 7, 6, 1, 7, 6, 1, 2, 7, 2

Also calculate the hit ratio and miss ratio. **[3m]**

8) Consider the following segment table-

| Segment No. | Base | Length |
|---|---|---|
| 0 | 1219 | 700 |
| 1 | 2300 | 14 |
| 2 | 90 | 100 |
| 3 | 1327 | 580 |
| 4 | 1952 | 96 |

Which of the following logical addresses will produce trap addressing error?

1. 0, 430
2. 1, 11
3. 2, 100
4. 3, 425
5. 4, 95

Calculate the physical address if no trap is produced. **[3m]**

**9)** Suppose a disk has 201 cylinders, numbered from 0 to 200. At some time the disk arm is at cylinder 100, and there is a queue of disk access requests for cylinders 30, 85, 90, 100, 105, 110, 135 and 145. If Shortest-Seek Time First (SSTF) is being used for scheduling the disk access, write down the sequence in which these requests are processed/serviced. **[3m]**