Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur
Mid-semester Examination, Spring 2014
CS30002: Operating Systems

Full Marks: 40 **Time: 2 Hours**

Instructions: Answer ANY Four (4) questions

1. (a) Consider a processor does not have a TSL (Test and Set lock) instruction, but does have an instruction "SWAP REG MEM" to swap the content of a register (REG) and a memory word (MEM) in a single indivisible (atomic) action. Show how you can use SWAP to implement mutual exclusion. Propose the ***entry_section*** and ***exit_section*** code modules.

   (b) Explain how kernel data structures are protected from being modified by the user code.

   (c) Draw a unified process state diagram showing the functionality of Long term, short term, and medium term scheduler (swapper). Very brief explanation is required.

   (d) List up all the steps that occur when your program divides some number by 0.

   [4+2+2+2]

2. (a) Consider the following solution for the critical section problem.
   The two processes, P_0 and P_1, share the following variables

```
boolean flag[2]
int turn;     /* initialized to 0 or 1*/
```

   The structure of the Process i (i=0 or 1) is as follows

```
do{
       flag[i]=true;
       while(flag[j])
       {
              If(turn==j)
              {
                     flag[i]=false;
                     while(turn==j);
                     flag[i]=true;
              }
       }
Critical section();
       turn=j;
       flag[i]=false;
Non-Critical section();
} while(1);
```

"The above solution satisfies all the necessary requirements of the critical section problem". Prove or disprove with proper justification.

(b) Consider a system which implements (i) preemptive priority-based CPU scheduler and (ii) Peterson's solution for mutual exclusion. In such a system, is it possible that a high priority process gets delayed indefinitely because of the presence of lower priority processes? Assume that no process does any I/O and there is no deadlock in the system.

(c) Consider three CPU-intensive processes, which require 30, 20 and 10 time units and arrive at times 0, 2 and 6, respectively. How many context switches are needed if the operating system implements a preemptive shortest remaining time first scheduling algorithm? Assume that time units are integral and do not count the context switches at time zero and at the end. [5+2.5+2.5]

3. (a) Explain the differences between (i) Busy waiting and blocking (ii) System call and library function.

   (b) Explain the utility of a device driver.

   (c) Consider the following set of processes, with length of the CPU bursts given in milliseconds.

Process	Arrival time	Burst Time	Priority number
0	0	5	4
1	2	6	2
2	4	3	1
3	7	5	3

   Assume the following two points
   (1) Lower priority number indicates higher priority
   (2) The first response (say a **printf**) for the process P_0, P_1, P_2 and P_3 comes after 3 ms, 1 ms, 1 ms and 4 ms of their respective execution.

   Draw the Gnatt chart and illustrate the execution of these processes under the following scheduling algorithms
   i. Non-preemptive priority scheduling
   ii. Preemptive priority scheduling
   iii. Round Robin (quantum=2) where the next process is selected based on priority. Suspended process does not take part in the competition. High priority process does not preempt a running process.

   Compute the (a) average waiting time (b) average response time for each case. [2.5+1.5+6]

4. (a) State the bounded buffer producer-consumer problem. Show an application of the problem in the real operating system context.