COSC 450 Operating System Mini-Test #2

1.

In=false at the beginning
There are two processes P_0 and P_1.

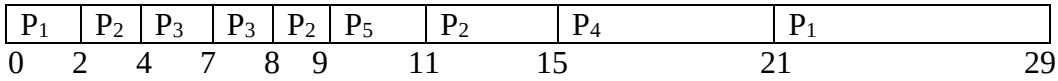
- P_0 is scheduled and P_0 try to go to critical section.
- P_0 read In = false, then time out. P_0 status change from running state to ready state.
- P_1 is scheduled. P_1 try to enter the critical section.
- P_1 read In =false and set In = ture then go to critical section.
- Sometimes later in the critical section, P_1 time out. P_1 status changes from running to ready state.
- P_0 rescheduled, and try to enter critical section. P_0 already read In =false before, P_0 set In = true again and enter the critical section.
- Now P_0 and P_1 are in critical section (violate mutual exclusion condition)

2.
Let's assume : empty = 0, full = N, mutex =1 at time T

- Producer is scheduled : produce item ,down mutex (now mutex =0), try to down empty. Since empty =0, producer cannot finish down operation and sleep on semaphore empty.
- Consumer is scheduled: down full (now full = N-1), then try to down mutex. Since mutex is already down by producer, consumer cannot finish down operation and sleep on semaphore mutex.
- Now producer and consumer sleep forever!

3. (2 pt.)

10 X X X X	8 7 X X X	8 5 4 X X	8 5 1 6 X	8 5 X 6 X	8 4 X 6 2	8 4 X 6 X	8 X X 6 X	8 X X X X



- Average waiting time = ((21 – 2) + ((8-4)+(11-9)) + 0 + (15 – 7) + 0)/5= 6.2

- Average turnaround time = (29 + (15 – 2) + (8 – 4) + (21 – 7) + (11 – 9))/5 = 12.4

4.

Sol) Lets assume a short-term scheduler use the priority to select a process from the ready queue. At time t_0 , there is only one process P_L with low priority in the ready queue. The short term scheduler select P_L and let it use CPU. Then P_L enter a critical region (section). At time t_1, a process P_H with higher priority becomes ready state. The short-term scheduler stop P_L to use CPU. Now P_H and P_L are in ready queue. The short-term scheduler select higher priority process P_H and let it use CPU. P_H try to get into the critical section. P_H must wait outside critical section since P_L is already in the critical section. Since P_L has lower priority, P_L never get change to use CPU. P_H never be able to enter critical session.