Introduction to Operating System **Exam Questions from Previous Years**

1. **Mutual Exclusion and Synchronization: Semaphores**
   Use semaphores to coordinates three types of processes in the problem described here. Santa Claus sleeps in his shop at the North pole and can only be awakened by either (1) all nine reindeer being back from their vacation, or (2) by some of the elves having difficulty in making toys. Elves can wake Santa up when three of them have problems. When three elves are having their problems solved, any other elves wishing to visit Santa must wait for the three elves to return. "Three elves waiting" has higher priority than all reindeer getting back. That is because without any toys Santa cannot go on h is gift_giving spree. All the elves' difficulty should be solved before Santa leav es. Solve this problem using semaphores.

2. Consider a system with 5 processes P1, P2, P3, P4 and P5 which arrive in that order at time t=0. The details about the processes are given below.

PROCESS	BURST TIME
P1	8
P2	2
P3	1
P4	2
P5	5

   a) (10 points) Give the Gantt chart illustrating the execution of these processes using (i) FCFS (ii) RR with quantum 2.
   b) (4 points) What is the turn-around time for P3 using each of the schedules in (a).
   c) (4 points) What is the average waiting time for the jobs using FCFS?

3. (3 X 5 = 15 points) Five jobs A through E arrive at a almost the same time. The estimated processing times are: 10,6,2, 4 and 8 units. For each of the scheduling algorithm determine the mean process completion time. Show the time chart for ea ch.

   (a) Round robin with quantum size 2 units.
   (b) Priority scheduling (Assume 3,2,2,1 and 4 for A to E respectively)
   (c) Shortest first.

4. (14 points) Consider the following set of processes:

Process Name	Arrival Time	Processing Time
P1	0	5
P2	1	2
P3	1	3
P4	7	4
P5	10	1

   (a) (5 points) Assume that shortest process first but with no preemption policy is used for scheduling the processes. Draw the timing diagram that clearly shows the starting and ending (completion) time of each process.

(b) (5 points) Repeat the same but assume that the scheduling policy now is shortest process first with preemption allowed.

(c) (4 points) Determine the mean turnaround time for these two policies.

5. (8 points) Assume the processing time (CPU-time requests) bursts for a process: 8, 9, 10, 10, 11, 12, 14. Predict the time duration of the next request for processing using (i) simple averaging and (ii) exponential averaging with alpha = 0.8.

6. (5 points) What is Rate Monotonic Scheduling (RMS)?

7. (2 points) What is the condition for determining the schedulability using RMS?

8. (4 points) Determine if the following tasks can be successfully scheduled using RMS? Show all the steps.

```
Task P1: C1 = 30 ; T1 = 100;
Task P2: C2 = 50 ; T2 = 200;
Task P3: C3 = 10 ; T3 = 100;
```

9. Consider the dining philosopher problem discussed in Fig.2.20, a copy of which is provided.

   (a) (5 points) Why is the state variable set to HUNGRY in the procedure *take forks*?

   (b) (4 points) Explain which operation wakes up the ith philosopher waiting on s[i] in procedure *take forks*.

   (c) (6 points) Consider the procedure *put forks*. Suppose the variable em state[i] was set to THINKING after the two calls to *test* rather than before. How would this change affect the solution for the case of 3 philosophers? For 100 philosophers?

10. (16 points) Consider the solution to the bounded-buffer producer/consumer problem given below:

```
Assume BUFSIZE, AVAIL and MUTEX are semaphores are initailized to 125, 0
and
1 respectively.
Producer: repeats
        { produce();
          wait(BUFSIZE);
          wait(MUTEX);
          append();
          signal(MUTEX);
          signal(AVAIL);}
Consumer: repeats
        { wait(AVAIL);
          wait(MUTEX);
          take();
          signal(MUTEX);
          signal(BUFSIZE);
          consume(); }
```

How would the meaning of the program change if the following were interchanged?
(a) wait(BUFSIZE); wait(MUTEX) in the producer;
(b) signal(MUTEX); signal(AVAIL) in the producer;

(c) wait(AVAIL); wait(MUTEX) in the consumer;
(d) signal(MUTEX); signal(BUFSIZE) in the consumer;

11. Process Description and Control: Process States Draw the state diagram that explains the life cycle of a process. Give an example for an event/condition that will bring about each of the transitions in the diagram.

12. Thread description and control a. What is the difference between a kernel level thread and a user level thread? List two distiguishing characteristics. b. List three resources local to a thread. c. Barrier like synchronization primitives: design details.