

SAMPLE PROBLEM LIST FOR CSE-336 LAB EXAM

CHAPTER : PROCESS SCHEDULING

Problem#A1 Consider a set of processes, with the length of the CPU burst given in milliseconds:

Process/Task/Job	Priority	Burst Time (ms)	Arrival Time (ms)
1	8	15	0
2	3	20	0
3	4	20	20
4	4	20	25
5	5	5	45
6	5	15	55

You may have to answer the following question(s) using the scheduling algorithm - FCFS / SJF / Priority / RR :

- Show the scheduling order of the processes using a Gantt chart.
- What is the response time for the process/task/job $P_i/T_i/J_i$?
- What is the turnaround time for the process/task/job $P_i/T_i/J_i$?
- What is the waiting time for the process/task/job $P_i/T_i/J_i$?

Problem#A2 You have to try for the following variants: SRF, PRR, EDF

- Shortest Remaining First (SRF) : The preemptive version of SJF where the selection of any process from the ready queue depends on the minimum (additional) CPU burst to complete the task.
- Progressive RR (PRR) : A variation of the RR scheduler where the time-quantum is increased by 10% if a process does not complete within its entire time-quantum. When the time time-quantum increases by 100% and any process blocks before using its entire time quantum, the time quantum will be reduced to its default value again.
- Earliest-deadline-first (EDF) : A variant of Priority scheduling where a deadline is given for each process and this deadline is treated as the priority of that process - the earlier the deadline, the higher the priority; the later the deadline, the lower the priority.

CHAPTER : DEADLOCKS

Problem#B1 Consider the following snapshot of a system having the resources (A, B, C, D) :

Process/Task/Job	Required Resources	Allocated Resources	Available Resources
1	6 4 7 3	3 1 4 1	2 2 2 4
2	4 2 3 2	2 1 0 2	
3	2 5 3 3	2 4 1 3	
4	6 3 3 2	4 1 1 0	
5	5 6 7 5	2 2 2 1	

You may have to answer the following question(s) using the banker's algorithm :

- Check whether the system is in a safe state or unsafe state?
- Demonstrate the order in which the threads may complete.
- If a request from thread T_i arrives for (2, 2, 2, 4), can the request be granted immediately?

Problem#B2 You have to try for the algorithm “Resource-Allocation-Graph” also.

In the LAB, the task list may be LIKE or PARTIAL of these problem(s)