

- 1 A CPU-scheduling algorithm determines an order for the execution of its scheduled processes. Given  $n$  processes to be scheduled on one processor, how many different schedules are possible? Give a formula in terms of  $n$ .
- 2 Explain the difference between preemptive and nonpreemptive scheduling.
- 3 Suppose that the following processes arrive for execution at the times indicated. Each process will run for the amount of time listed. In answering the questions, use nonpreemptive scheduling, and base all decisions on the information you have at the time the decision must be made.

<u>Process</u>	<u>Arrival Time</u>	<u>Burst Time</u>
$P_1$	0.0	8
$P_2$	0.4	4
$P_3$	1.0	1

- a. What is the average turnaround time for these processes with the FCFS scheduling algorithm?
  - b. What is the average turnaround time for these processes with the SJF scheduling algorithm?
- 4 Consider the following set of processes, with the length of the CPU burst time given in milliseconds:

<u>Process</u>	<u>Burst Time</u>	<u>Priority</u>
$P_1$	2	2
$P_2$	1	1
$P_3$	8	4
$P_4$	4	2
$P_5$	5	3

The processes are assumed to have arrived in the order  $P_1, P_2, P_3, P_4, P_5$ , all at time 0.

- a. Draw four Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF, non-preemptive priority (a larger priority number implies a higher priority), and RR (quantum = 2).
- b. What is the turnaround time of each process for each of the scheduling algorithms in part a?
- c. What is the waiting time of each process for each of these scheduling algorithms?
- d. Which of the algorithms results in the minimum average waiting time (over all processes)?

- 5 The following processes are being scheduled using a preemptive, round-robin scheduling algorithm.

<u>Process</u>	<u>Priority</u>	<u>Burst</u>	<u>Arrival</u>
$P_1$	40	20	0
$P_2$	30	25	25
$P_3$	30	25	30
$P_4$	35	15	60
$P_5$	5	10	100
$P_6$	10	10	105

Each process is assigned a numerical priority, with a higher number indicating a higher relative priority. In addition to the processes listed below, the system also has an **idle task** (which consumes no CPU resources and is identified as  $P_{idle}$ ). This task has priority 0 and is scheduled whenever the system has no other available processes to run. The length of a time quantum is 10 units. If a process is preempted by a higher-priority process, the preempted process is placed at the end of the queue.

- Show the scheduling order of the processes using a Gantt chart.
  - What is the turnaround time for each process?
  - What is the waiting time for each process?
  - What is the CPU utilization rate?
- 6 What advantage is there in having different time-quantum sizes at different levels of a multilevel queueing system?
- 7 Many CPU-scheduling algorithms are parameterized. For example, the RR algorithm requires a parameter to indicate the time slice. Multilevel feedback queues require parameters to define the number of queues, the scheduling algorithms for each queue, the criteria used to move processes between queues, and so on.
- These algorithms are thus really sets of algorithms (for example, the set of RR algorithms for all time slices, and so on). One set of algorithms may include another (for example, the FCFS algorithm is the RR algorithm with an infinite time quantum). What (if any) relation holds between the following pairs of algorithm sets?
- Priority and SJF
  - Multilevel feedback queues and FCFS
  - Priority and FCFS
  - RR and SJF
- 8 Suppose that a CPU scheduling algorithm favors those processes that have used the least processor time in the recent past. Why will this algorithm favor I/O-bound programs and yet not permanently starve CPU-bound programs?