Computer Systems Tutorial

November 17^{th} 2023

Exercise 1 A CPU scheduling algorithm determines an order for the execution of its scheduled processes. Given 5 processes to be scheduled on one processor, how many different schedules are possible? Assume that scheduling is nonpreemptive and no IO is made. How many different schedules are possible for n processes?

Exercise 2 Consider the following set of processes, with the length of the CPU burst time given in milliseconds:

Process	Burst time	Priority
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 arrive in the order listed all at time 0.

a Draw four Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF, nonpreemptive priority(a larger priority number implies a higher priority), and RR(quantum=2ms).

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)?

Exercise 3 The following processes are being scheduled using a preemptive, round-robin scheduling algorithm. Each process is assigned a numerical priority, with a higher number indicating a higher relative priority. In addition to the processes listed above, 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 (first executes the process with the highest priority and processes with the same priority are scheduled by RR).

Process	Priority	Burst	Arrival
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

a Show the scheduling order of the processes using a Gantt chart.

b What is the turnaround time for each process?

c What is the waiting time for each process?

d What is the CPU utilization rate (percentage of time the CPU is busy)?

Exercise 4 You and your friend John are working on a Parallel Programming project, which is an application that does not use IO functions. You have been told by your project manager that your software should have at least 6 times speedup as compared to its sequential implementation. You have also been told that your application will run on a system with 16 CPUs. Estimate the minimum percentage of the code that needs to be parallelized in your application, to meet the above performance requirement.

Exercise 5 Explain the difference between kernel threads and user threads.