CPSC 331 - Operating Systems

Instructor Marc Corliss

Homework 3

Due: 10/8/07 (beginning of class)

1. Define the difference between preemptive and nonpreemptive scheduling. State why strict nonpreemptive scheduling is unlikely to be used in a modern desktop machine.

2. Suppose that a scheduling algorithm (at the level of short-term CPU scheduling) 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?

3. Explain the differences in the degree to which the following scheduling algorithms discriminate in favor of short processes:

   (a) First-come, first-served

   (b) Round robin

4. Answer true or false below and justify your answer. With round robin scheduling:

   (a) A larger time slice will likely increase total system throughput.

   (b) A larger time slice will likely improve average job turnaround time.

   (c) A larger time slice will likely improve average job response time.

   (d) Jobs can be starved.

5. Consider the following set of processes, with the length of the CPU-burst time given in milliseconds:

Process	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

The processes are assumed to have arrived in the order P1, P2, P3, P4, P5, all at time 0.

(a) What is the turnaround time of each process using the following scheduling algorithms: first-come, first-served; shortest job first; nonpreemptive priority; and round robin?

(b) What is the waiting time of each process for each of the scheduling algorithms in (a)?

6. For each of the following scenarios, state whether or not deadlock is possible. You should assume no preemption unless stated otherwise, and that resources can be held by at most one process at a time. Briefly explain your answer.

   (a) Resources are numbered, and processes acquire resources in increasing order. A process may hold multiple resources at a time, but a resource can only be held by a single process at a time.

   (b) Processes are numbered (timestamped), and can acquire any number of resources. The manager for a particular resource will service pending requests in timestamp order.

   (c) A process can hold only one resource at a time.

   (d) There are 6 equivalent resources, meaning that a request can be serviced by allocating any available resource to the process. Each process requests 1 resource at a time but may need up to 2 resources simultaneously. There are 4 process in the system.

7. Take a careful look at Figure 3-11(b) in the textbook. If D asks for one more unit, does this lead to a safe or an unsafe state? What if the request came from C instead of D?

8. Consider the following snapshot of a system using Banker's algorithm:

	Allocation				Max			
	A	B	C	D	A	B	C	D
P0	0	0	1	2	0	0	1	2
P1	1	0	0	0	1	7	5	0
P2	1	3	5	4	2	3	5	6
P3	0	6	3	2	0	6	5	2
P4	0	0	1	4	0	6	5	6

Available			
A	B	C	D
1	5	2	0

(a) What is the content of the matrix **`Need`**?

(b) Is the system in a safe state?

(c) If a request from process P1 arrives for (0, 4, 2, 0) can the request be granted immediately?