CSCI 4320 (Principles of Operating Systems), Fall 2002

Homework 3

Assigned: October 15, 2002.

Due: October 18, 2002, at noon. *Not accepted late.*

Credit: 20 points.

1 Reading

Be sure you have read chapter 2.

2 Problems

Answer the following questions. You may write out your answers by hand or using a word processor or other program, but please submit hard copy, either in class or in my mailbox in the department office.

1. (8 points) Five batch jobs (call them A through E) arrive at a computer center at almost the same time. Their estimated running times (in minutes) and priorities are as follows, with 5 indicating the highest priority:

job	*running time*	*priority*
A	10	3
B	6	5
C	2	2
D	4	1
E	8	4

For each of the following scheduling algorithms, determine the turnaround time for each job and the average turnaround time. Assume that all jobs are completely CPU-bound (i.e., they do not block).

- First-come, first-served (run them in alphabetic order by job name).
- Shortest job first.
- Round robin, using a time quantum of 1 minute.
- Round robin, using a time quantum of 2 minutes.
- Priority scheduling.

2. (4 points) In round-robin scheduling, normally the scheduler maintains a list of all runnable processes, with each process occurring exactly once in the list. What would happen if a process occurred more than once in the list? Can you think of a reason to allow this? If so, what?

3. (4 points) Recall that some proposed solutions to the mutual-exclusion problem (e.g., Peterson's algorithm) involve busy waiting. Do such solutions work if priority scheduling is being used and one of the processes involved has higher priority than the other(s)? Why or why not? How about if round-robin scheduling is being used? Why or why not?

4. (4 points) Suppose that a scheduling algorithm favors processes that have used the least amount of processor time in the recent past. Why will this algorithm favor I/O-bound processes yet not permanently starve CPU-bound processes?