Quiz 3

Course: cs241 - System Programming, CS Department
Date: February 17, 2006
Netid: Name: UIN:

Note: Completion of quiz is an individual effort. The quiz takes 10 minutes. The student gets additional 5 points for taking the quiz. *Each question has ONLY ONE ANSWER!!!*

---

1. **(1 Point) For a fixed number of processes, shortest-job-first scheduling algorithm**:
   - **I. Minimizes average waiting time**
   - **II. Minimizes average turn-around-time**
   - **III. Minimizes CPU throughput**
   - **IV. Maximizes average response-time**

Which answers are TRUE?

   a. only I & II
   b. only I & IV
   c. only I, II, & III
   d. only I, II, & IV
   e. only II & IV

2. **(1 Point) There are four queues in a multi-level queuing scheduling system: The first queue runs the First-Come-First-Serve scheduling policy, and the three other queues run Round-Robin Scheduling Policies with quantum = 8, 16, and 32 milliseconds respectively. What is the quantum of the First-Come-First-Served queue?**
   a. 8 milliseconds
   b. 64 milliseconds
   c. Infinity
   d. None of the above

3. **(1 Point) In a certain system, *N* processes arrive at the CPU ready queue at the same time. Which of the following scheduling algorithms will give the minimum average turnaround time of these *N* processes (assuming that the CPU ready queue is initially empty and that no new processes arrive at the CPU ready queue before the CPU finishes these *N* processes)?**
   a. First-Come, First Served
   b. Round-Robin
   c. Shortest-Job-First
   d. None of the above

4. **(1 Point) Application of Little's Law requires:**
    a. FIFO scheduling
    b. Shortest Job First scheduling
    c. Steady State in the system
    d. No restrictions

5. **(1 Point) In First-Come, First-Served scheduling policy, I/O bound processes may have to wait long in the ready queue waiting for a CPU bound job to finish. This is known as**
    a. Aging
    b. Priority inversion
    c. Priority inheritance
    d. Convoy effect
    e. None of the above

6. **(1 Point) In the five philosophers' problem, if each philosopher does**

    m[i].P(); m[(i+1)%5].P(); munch; m[(i+1)%5].V(); m[i].V();

    where *P()* represents the *Down()* operation, *V()* represents the *Up()* operation and *m[5]* represents semaphores for each philosopher *i*, there is a possibility of
    a. deadlock
    b. starvation
    c. both (a) and (b)
    d. none of the above

7. **(1 Point) If a process does *x.signal*, where *x* is a condition variable of a monitor and no process is awaiting condition x,**
    a. The signal operation has no effect
    b. The next process do *x.wait* does not get blocked
    c. None of the above

8. **(1 Point) The sequence of instructions (used in Processes) using *mutex* semaphore**

```
Up(mutex);
Critical section
Down(mutex);
```

satisfies mutual exclusion:
    a. True
    b. False

9. **(2 Points) Consider the reader-writer code below**:

```
semaphore mutex, wrt;
int readcount =0;
mutex =1; wrt = 1;
```

Writer:

```
Up(&wrt);

Write document

Down(&wrt);
```

Reader:

```
Down(&mutex);
readcount++;
if (readcount == 0) Down(&wrt);
Up(&mutex);

Read Document

Down(&mutex);

readcount--;
if (readcount == 1) Up(&wrt);
Up(&mutex);
```

Which statement is TRUE based on the above code?

(a) Multiple readers are satisfying mutual exclusion among each other when reading a shared document
(b) Writer and multiple readers are simultaneously reading and writing into the document
(c) The first reader blocks the entry of the writer to the critical region
(d) The last reader signals the writer to enter the critical region