| **CS 4760** | **Operating Systems** | **Test 2** |
|---|---|---|
| Name: | Spring 2015 | Max Pts: 55 |

**Important**: This is an open book test. You can use any books, notes, or paper but no electronic device. *Do not log into the computer during the test, or use any electronic or communications device. Your cell phones must be on silent mode*. Any other device with an ON-OFF switch should have its switch in the OFF position. Any calculations and rough work can be done on the back side of the test pages. You will lose five points for not writing your name.

1. [6 pt] What is the advantage in having different time-quantum size at different levels of a multilevel feedback queue?

2. [8 pt] Suppose that a system is in an unsafe state in terms of resource management. Is it possible for the processes to complete their execution without entering a deadlocked state? If yes, show with an example. If no, explain the reason.

3. [10 pt] In the code below, three processes are competing for six resources labeled `A` through `F`.

```
void p0()
{
  while ( 1 )
  {
    get ( A );
    get ( B );
    get ( C );
    crit_sec ( A, B, C );
    release ( A );
    release ( B );
    release ( C );
  }
}
```

```
void p1()
{
  while ( 1 )
  {
    get ( D );
    get ( E );
    get ( B );
    crit_sec ( B, D, E );
    release ( D );
    release ( E );
    release ( B );
  }
}
```

```
void p2()
{
  while ( 1 )
  {
    get ( C );
    get ( F );
    get ( D );
    crit_sec ( C, D, F );
    release ( C );
    release ( F );
    release ( D );
  }
}
```

Use a process resource graph to show the possibility of a deadlock in this implementation.

4. [15 pt] Assume you have the following jobs to execute with one processor:

| Process | Burst time | Arrival time |
|---|---|---|
| $p_0$ | 6 | 0 |
| $p_1$ | 9 | 3 |
| $p_2$ | 7 | 5 |
| $p_3$ | 6 | 8 |
| $p_4$ | 2 | 11 |

Give the average wait time for this set of processes using the following algorithms. Specify the arbitration rule used for each algorithm, if needed.

(a) First in first out

(b) Shortest job next (non-preemptive)

(c) Shortest remaining time next (pre-emptive)

(d) Round robin, with a quantum of 7

(e) Round robin, with a quantum of 5 plus context switch time of 1

5. [6 pt] Why do we need to relocate processes at execution time?

6. [10pt] A machine has a memory of 64 frames, with each frame being 1K bytes. Current free-frame list is: 0x11, 0x18, 0x0D, 0x29, 0x01, 0x16, 0x3B, 0x12, 0x2A, 0x3C, 0x20, 0x1D, and 0x15. You just scheduled a process that requires 10 frames. Show the resulting page table. Show the translation of logical address 0x0D36 and 0x1222 into physical addresses using your page table. Express your result in hexadecimal. Indicate seg fault, if any.