CS 4760	**Operating Systems**	**Test 2**
Name:	Spring 2016	Max Pts: 51

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. Change your cell phones to silent mode*. Any calculations and rough work can be done on the back side of the test pages. If there is a syntax error in any program segment, just write it down and you will get full credit for the problem. Please write legibly; if I cannot read what you wrote, I'll give you a zero. You will lose five points for not writing your name.

1. [6 pt] Explain the difference between response time and turnaround time. Give a clear example of the difference between these two terms.

2. [8 pt] One way to fix the deadlock problem is to deny circular wait. This can be achieved by creating a total order on all resources and allocating resources in order. Illustrate with an example why it cannot be achieved by creating a partial ordering on resources. Show the details of your example using a process resource graph.

3. [6 pt] Why do multilevel feedback queue schedulers vary the time quantum associated with different queues? What would be the disadvantage of having the same time quantum at all levels?

4. [6 pt] Once the system enters a deadlock state, someone has to pay a penalty to recover from deadlock. Explain clearly why there is no penalty-less way for deadlock recovery.

5. [15 pt] Assume you have the following jobs to execute with one processor:

Process	Burst time	Arrival time
p_0	3	0
p_1	5	1
p_2	9	3
p_3	5	6
p_4	8	7

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 6

(e) Round robin, with a quantum of 4 plus context switch time of 1

6. [10 pt] Assume a system with four resource types, $C = \langle 9, 7, 6, 6 \rangle$ (this is the total number of resources in the system, and not what is currently available), and the maximum claim table shown below.

Process	R_0	R_1	R_2	R_3
p_0	5	1	2	3
p_1	9	6	0	1
p_2	7	2	4	2
p_3	3	6	4	5
p_4	1	4	4	4

The resource allocator is considering allocating resources according to the following table:

Process	R_0	R_1	R_2	R_3
p_0	2	2	2	2
p_1	2	0	0	0
p_2	3	2	0	1
p_3	0	2	2	2
p_4	2	1	1	1

Run the safety algorithm on this system to determine if this state is safe. If it is safe, give the sequence in which processes can be run. If it is unsafe, enumerate the processes that may get involved in a deadlock. Show your steps from the algorithm.