CS 4760	**Operating Systems**	**Test 2**
Name:	Spring 2010	Max Pts: 53

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. Switch off your cell phones*. 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. You will lose five points for not writing your name.

1. [6 pt] Differentiate between response time and turnaround time. Give some example situation to illustrate those.

2. [6 pt] How can you prevent the circular wait condition to prevent deadlocks. Is there a drawback to your solution? Describe it.

3. [15 pt] Assume you have the following jobs to execute with one processor:

Process	Burst time	Arrival time
p_0	9	0
p_1	7	1
p_2	7	4
p_3	8	7
p_4	2	9

Give the average wait time for this set of processes using the following algorithms.

(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 3

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

4. [6 pt] Consider a fixed partitioning scheme with equal-size partitions of 2^{16} bytes and a total main memory size of 2^{24} bytes. A process table is maintained that includes a partition for each resident process. How many bits are required for the pointer?

5. [10 pt] Consider a system with the following set of processes and states:

$$P = \{p_0, p_1\},\ S = \{s_0, s_1, s_2, s_3\}$$

State changes due to processes are:

$$\begin{array}{ll} p_0(s_0) = \{s_1, s_3\} & p_1(s_0) = \Omega \\ p_0(s_1) = \Omega & p_1(s_1) = \Omega \\ p_0(s_2) = \{s_0\} & p_1(s_2) = \{s_2, s_3\} \\ p_0(s_3) = \Omega & p_1(s_3) = \{s_2\} \end{array}$$

Draw the corresponding state change diagram. Is the system safe? Is it deadlocked? Is there a knot in the system?

6. [10 pt] You have a physical memory of 64MB, starting at address 0. Your operating system requires at least 10MB all the time. Consider the arrival of processes as follows:

Process	Burst time	Arrival time	Memory needed
p_0	37	0	32MB
p_1	49	5	2MB
p_2	8	13	16MB
p_3	4	22	8MB
p_4	8	26	8MB
p_5	6	32	2MB
p_6	2	41	2MB
p_7	5	50	8MB

Show the layout of memory, using first-fit algorithm, at times 20, 40, 60, 80, and 100.