CS 414	**Operating Systems**	Test 2
Name:	Spring 2003	Max Pts: 47

Important: This is an open book test. You can use any books, notes, or paper. *You are not allowed to use any communications device. Do not log into the computer during the test.* 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. [10 pt] In a certain architecture, we do not have a `test_and_set` instruction but do have an indivisible `swap` instruction that exchanges the contents of two memory variables in a single step. Show how can you use `swap` to implement binary semaphores.

2. [5 pt] What is the difference between a semaphore and a `condition` variable? Why can't we use semaphore in place of `condition` variable?

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

Process	Burst time	Arrival time
p_0	7	0
p_1	8	2
p_2	6	4
p_3	9	5
p_4	6	7

Give the average wait time and average turnaround time for each process using the following algorithms. Also compute the percentage of time when the system is busy with user processes.

(a) First in first out

(b) Shortest job next (no preemption)

(c) Shortest remaining time next

(d) Round robin, with a quantum of 3

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

4. [5 pt] Why do we need to implement the operations on processes and resources as kernel primitives?

5. [12 pt] Assume a system with five resource types, $C = \langle 7, 9, 9, 13, 11 \rangle$ (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	R_4
p_0	2	6	7	3	7
p_1	0	1	4	6	2
p_2	5	8	6	2	3
p_3	0	2	3	5	3
p_4	2	4	0	4	5

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

Process	R_0	R_1	R_2	R_3	R_4
p_0	2	1	4	2	3
p_1	0	1	4	6	2
p_2	1	3	1	1	3
p_3	0	2	0	4	3
p_4	2	2	0	0	0

Draw the process resource graph for this system. Are there any processes that are blocked/deadlocked? Which are they? Show this by reducing the graph as much as you can.