Assignment 3

This assignment is due on November 12.

1. (10%) Suppose that three different processes, P_1, P_2, and P_3, share the variables `x` and `y` and execute the following code fragments concurrently, and that the initial value of `y` is 5. The instructions to add and subtract in memory are not atomic.

P_1	P_2	P_3
`x = y;`	`x = y;`	`x = y;`
`x = x + 1;`	`x = x + 1;`	`x = x - 1;`
`y = x;`	`y = x;`	`y = x;`

   (a) What are all of the possible final values of y?

   (b) If all of the above instructions were atomic, how many different interleaved sequences could there be?

2. (15%) Explain why disabling interrupts does not provide a solution for the critical section problem in multiprocessor systems. Give an example to support your argument.

3. (25%) Modify the solution to the Weak Reader Priority Readers-Writers Problem (in Chapter 6 slides) to create a solution to the Strong Reader Priority Readers-Writers Problem. Hint – it takes only a very little change to accomplish this.

4. (30%) Consider the following snapshot of a resource allocation system. There are no current outstanding, unsatisfied requests.

	Maximum Claim				**Current Allocation**				**Current Need**			
process	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4
P_1	0	0	1	2	0	0	1	2				
P_2	2	7	5	0	2	0	0	0				
P_3	6	6	5	6	0	0	3	4				
P_4	4	3	5	6	2	3	5	4				
P_5	0	6	5	2	0	3	3	2				

Available			
R_1	R_2	R_3	R_4
2	1	0	0

   (a) Compute what each process might still request and display it in the columns labeled "Current Need."

   (b) Assuming that each process makes the requests you computed in the first question, is the resulting system in a safe or an unsafe state? Explain why or why not.

   (c) Is the resulting state a deadlock state? Why or why not?

   (d) Which processes, if any, are or may become deadlocked?

   (e) Draw a resource allocation graph that represents the system.

5. (20%) A computer system has a total of 150 units of memory, currently allocated to three processes as shown below:

Process ID	Maximum Claim	Allocation
1	70	45
2	60	40
3	60	15

Apply the Banker's Algorithm to determine whether it would be safe to grant each of the following requests. If your answer is yes, provide a safe sequence that justifies the answer. If your answer is no, show how the safety algorithm fails in this case.

(a) A fourth process arrives with a maximum claim of 60 memory units and an initial request for 25 units.

(b) A fourth process arrives with a maximum claim of 60 memory units and an initial request for 35 units.

Submitting the Solution

You must type your assignment. Handwritten assignments will not be accepted. You may, if you wish, submit it electronically instead of handing it in. This will save paper. If you choose to do this, then login to eniac remotely or go to Lab 1000G and login there and follow these instructions:

- The file must be either a plain text file or a PDF document;
- It must be named *hwk3_username* (with a .pdf extension if it is a PDF file);
- It must be placed in the directory `/data/yoda/b/student.accounts/cs340/projects/hwk3`;
- It must have permission 0600. If you (still) do not know how to do this, use the command `chmod 0600 filename`.

Do not submit it by hand if you also submit electronically. That is a waste of paper. Fear not; I will be able to read your assignment if you put it on-line.