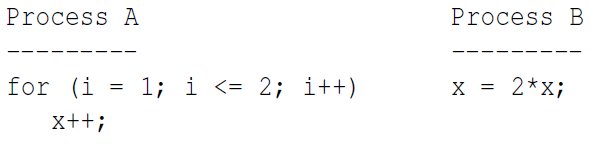
Exam 1 Spring 2017

1. Explain what are CPU modes. Explain their uses. How does the CPU know what mode it is in?
2. What is an atomic instruction? What would happen if multiple CPUs/cores execute their atomic instructions?
3. Draw the state diagram of a process from its creation to termination, including all transitions. Make sure you will elaborate every state and every transition in the diagram.
4. What is a context? Provide a detail description of all activities of a context switch.
5. Why is handling threads cheaper than handling processes?
6. Define the meaning of a race condition? Answer the question first and use execution sequences with a clear and convincing argument to illustrate your answer.
7. A good solution to the critical section problem must satisfy three conditions: mutual exclusion, progress and bounded waiting. Both progress and bounded waiting involve some form of waiting. Explain and differentiate the waiting in progress and bounded waiting.
8. Consider the following two processes, A and B, to be run concurrently using a shared memory for the int variable x. Assume that x is initialized to 0, and x must be loaded into a register before further computations can take place. What are all possible values of x after both processes have terminated?



1. Consider the following solution to the mutual exclusion problem for two processes P0 and P1. A process can be making a request REQUESTING, executing in the critical section IN CS, or having nothing to do with the critical section OUT CS. This status information, which is represented by an int, is saved in flag[i] of process Pi. Moreover, variable turn is initialized elsewhere to be 0 or 1. Note that flag[] and turn are global variables shared by both P0 and P1. Prove rigorously that this solution satisfies the mutual exclusion condition.

