Exam 1 Fall 2014

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. What is a context? Provide a detail description of all activities of a context switch.
4. 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.
5. Enumerate the major differences between kernel-supported threads and user-level threads.
6. Define the meaning of a race condition? Answer the question first and use an execution sequence with a clear and convincing argument to illustrate your answer.
7. Explain the progress and bounded waiting conditions and enumerate their differences.
8. Design a C program segment so that the main() creates two child processes with fork(), each of these two child processes creates two child processes, etc such that the parent child relationship is a perfectly balanced binary tree of depth n with main() at the root. The depth n have already stored a valid positive integer. The main() prints its PID, and each child process prints its PID and its parent’s PID.
9. Consider the following solution to the mutual exclusion problem for two processes *P*0 and *P*1. In the following, (*a, b*) \_ (*c, d*) holds if (*a > c*) or (*a* = *c* and *b* \_ *d*). Note that max(*a, b*) is the maximum function that returns the larger of *a* and *b*. For example, (3*,* 5) *>* (2*,* 7) and (4*,* 5) *>* (4*,* 2) both hold, because 3 *>* 2 for the former and 4 = 4 and 5 *>* 2 for the latter.

