Exam 1 Fall 2016

1. Explain interrupts and traps, and provide a detailed account of the procedure that an operating system handles an interrupt.
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. What is thread cancellation? How many commonly used thread cancellation types are there. Name the types and provide an explanation for each type.
6. Explain the meaning of thread safe?
7. A good solution to the critical section problem must satisfy three conditions: mutual exclusion, progress and bounded waiting. Explain the meaning of the progress condition. Does starvation violate the progress condition?
8. 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.
9. 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 has already been stored a valid positive integer. The main() prints its PID, and each child process prints its PID and its parent’s PID.
10. The following solution to the critical section problem has a global array flag[2] and an int variable turn. Prove rigorously that this solution satisfies the mutual exclusion condition.

