Homework 1 — due in class on Thursday 7 September

Total number of points available on this homework is 180. Full credit is equivalent to 100 points, so you can skip some problems. Nevertheless, please try to do them all.

1. (20 pts) When is the PCB guaranteed to reflect the current state of a process? Is it possible that the PCB may not reflect the current state of the process? Why or why not?

2. (20 pts) During a context switch, the operating system executes instructions that choose the next process to execute. These instructions are typically at a fixed location in memory. Why?

3. (20 pts) Write a C program in which a parent forks a child, and the child forks a grandchild. The grandchild writes: “My pid is X, my parent’s pid is Y, and my grandparent’s pid is Z.” (where X, Y, and Z are the Unix process ids).

4. (15 pts) Compare and contrast Microsoft Windows and Apple Macintosh operating systems. What differences are perceptible to the user?

5. (20 pts) Should a graphical user interface be considered an integral part of the operating system? If we agree that MS-DOS is an operating system, then is MS-Windows also an operating system, or simply a colorful graphical user interface? Is X11 an integral part of Unix? Is Internet Explorer an integral part of MS-Windows?

6. (25 pts) Consider a directed graph $G = (V,E)$. (a) Describe an algorithm that will determine if in the graph there is a cycle (a sequence of edges that can be traversed to arrive at the original vertex). (b) Describe an algorithm that will determine, given graph G with no cycles and a pair v_i, v_j, if adding an edge $e = v_i \rightarrow v_j$, would create a cycle. (c) Discuss the complexity of your algorithms. How does the choice of data structure for graphs affect complexity? (d) Implement your algorithms, and run over a few example graphs.

7. (25pts) (problem 1.5 on p. 32 of textbook (Stallings)) Consider a computer system that contains an I/O module controlling a simple keyboard/printer teletype. The following registers are contained in the CPU and connected directly to the system bus:

   1. INPR: Input Register, 8 bits
   2. OUTR: Output Register, 8 bits
   3. FGI: Input Flag, 1 bit
   4. FGO: Output Flag, 1 bit
   5. IEN: Interrupt Enable, 1 bit

Keystroke input from the teletype and output to the printer are controlled by the I/O module. The teletype is able to encode an alphanumeric symbol to an 8-bit word and decode an 8-bit word into an alphanumeric symbol. The Input Flag is set when an 8-bit word enters the input register from the teletype. The Output Flag is set when a word is printed.

(a) Describe how the CPU, using the first four registers listed in this problem, can achieve I/O with the teletype.

(b) Describe how the function can be performed more efficiently by also employing IEN.

8. (35pts) Consider a memory manager within an operating system. User programs make requests to the memory manager to allocate a "page" of memory, and later they release the page back to the manager. We say that a page lives from the time of the allocation request to the time of the release request. More precisely, we define the lifetime of a page as the number of allocation requests for other pages that happen between the allocation request and the release request. Suppose that these lifetimes L are random, more precisely, that they are random variables, independent and identically distributed, and drawn from a distribution with a known probability density f_L.

(a) What is the expected value of the number of pages allocated, once the system is in a steady state?

(b) What can be said about the distribution of the number of pages allocated?