CSE 421/521 – Operating Systems
Fall 2013 - Homework Assignment #1

The homework assignments will **not** be graded. Please solve these problems this week as an exercise for self-assessment. You may expect similar questions in quizzes and in the exams. The solutions will be provided next week for your convenience.

Problem 1:

Direct memory access is used for high-speed I/O devices in order to avoid increasing the CPUś execution load.

a. How does the CPU interface with the device to coordinate the transfer?

b. How does the CPU know when the memory operations are complete?

c. The CPU is allowed to execute other programs while the DMA controller is transferring data. Does this process interfere with the execution of the user programs? If so, describe what forms of interference are caused.

Problem 2:

What is the main advantage of the microkernel approach to system design? How do user programs and system services interact in a microkernel architecture? What are the disadvantages of using the microkernel approach?

Problem 3:

Give detailed answer to each of the following questions:

(a) What is the difference between fork() and exec() on Unix?

(b) What resources are used when a thread is created? How do these differ from those used when a process is created?

(d) What are context switches used for and what does a typical context switch involve?

Problem 4:

Consider a multiprocessor system and a multithreaded program written using the many-to-many threading model. Let the number of user-level threads in the program be more than the number of processors in the system. Discuss the performance implications of the following scenarios.

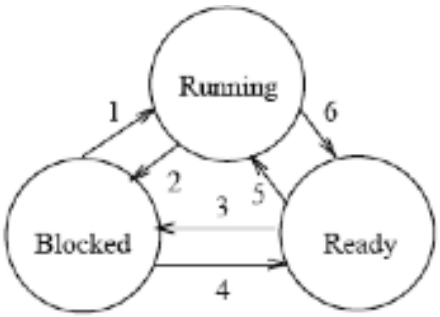
a. The number of kernel threads allocated to the program is less than the number of processors.

b. The number of kernel threads allocated to the program is equal to the number of processors.

c. The number of kernel threads allocated to the program is greater than the number of processors but less than the number of user-level threads.

Problem 5:

As shown below, processes can be in one of three states: running, ready and blocked. There are six possible state transitions (labeled 1-6). For each label, indicate whether the transition is valid or not valid. If valid, indicate when the transition is used for a process (i.e. give an example). If the transition is not valid then indicate why.



State transitions:

(a) 1: Blocked to Running
(b) 2: Running to Blocked
(c) 3: Ready to Blocked
(d) 4: Blocked to Ready
(e) 5: Ready to Running
(f) 6: Running to Ready