CS140 Operating Systems and Systems Programming
Midterm Exam

February 8, 2008

(Total time = 50 minutes, Total Points = 50)

Name: (please print)_____

In recognition of and in the spirit of the Stanford University Honor Code, I certify that I will neither give nor receive unpermitted aid on this exam.

Signature:_____

This examination is close notes and close book. You may not collaborate in any manner on this exam. You have 50 minutes to complete the exam. Before starting, please check to make sure that you have all 11 pages.

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
Total	

1. (5 points) Your partner has already started looking at programming assignment 3 for the class where you add a virtual memory to your Pintos OS. He decides to implement a local replacement policy. He further claims that he can still use the clock LRU approximation algorithm except now rather than having a single clock there would be one clock per process in this system. Does this make sense? Explain your answer.

2. (5 points) In Pintos project #1 you had to implement the timer_sleep() alarm clock function that causes a thread to sleep the specified number of timer ticks. The timer was configured to go off 100 times per second so a timer tick is 10ms. Your partner is playing around with the implementation and discovered that the following two functions run for the same amount of time:

void Function1(void)
{
  int i;
  for (i = 0; i < 10; i++) {
  Compute(i);
  timer_sleep(1);
  }
}
	void Function2(void)
	{
	  timer_sleep(10);
	}

where Compute(int i) is a compute bound function that takes around 7ms each time it is called. Explain how these functions both these functions take the same amount of time even though Function1 appears to much more.

3. (5 points) Shared libraries have both benefits and overheads for a system.
   (a) Explain how shared libraries reduce the amount of disk space needed to storage the operating system environment.
   (b) Explain how shared libraries can result in an increased amount of memory used by page tables in the system.

4. (5 points) You are investigating a benchmark programs that runs slow because of high TLB miss overheads. For each of the suggested changes to the system, describe if you think it would help reduce the overheads, make the TLB overheads greater, or leave them unchanged. Be sure to justify your answer. Suggested changes:
   (a) Increasing the VM Page size.
   (b) Going from a hierarchical page table to a flat page table.
   (c) Set the page table entry write protect bits on all the program read-only pages.

5. (5 points) The original Unix CPU scheduler simply divided processes CPU usage by 2 every second. Describe the problem with this approach.

6. (5 points) Describe two reasons why modern operating systems have a ready queue per processor rather than a single queue of ready jobs.

7. (5 points) What would cause a job in a MLFQ CPU scheduler to get given a longer timeslice?

8. (5 points) (a) Describe the problem with using semaphores to synchronized access to data that is touched by both normal kernel code and by interrupt handlers.
   (b) Would monitors have the same problem?
   (c) Would using free wait synchronization have the same problem?
   Be sure to explain your answers.

9. (5 points) Explain why in places that the original Hoare semantic monitors has “if” statements, most current monitor implementations need to have a “while” statements.

10. (5 points) Assume that you were looking at the source code of an uniprocessor operating system and every critical section in the kernel was enclosed by special labels in the code that were exported to the scheduling module. Instead of having locks or disabling of interrupts, the address of the first and the instructions every critical section was available to the scheduler. Explain how the this system could be implementing critical sections.