

CS140 Operating Systems and Systems Programming Midterm Exam

February 9th, 2004

(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 8 pages.

| | |
|--------------|--|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| Total | |

- 1) (9 points) Assume that you are adding a demand-paged virtual memory subsystem to an existing operating system that previously used base-and-bound protection and an Unix-like CPU scheduler.
 - (a) What extensions would you need to make to the CPU scheduler subsystem to avoid thrashing? Under what workloads would your extensions fail to stop thrashing?
 - (b) Would a compute-bound job that takes many page faults see its priority go up or down? Justify your answer.
 - (c) Could the introduction of a virtual memory system introduce priority inversion in a system that didn't have one before? Justify your answer.

- 2) (6 points) Assume that you have a virtual memory subsystem capable of supporting the addition of physical memory to a machine while it was running. Would adding more physical memory to a machine increase, decrease, or leave unchanged the working set of the processes running on the system? Explain your answer.

- 3) (7 points) Most available operating systems either implement a global LRU approximation page replacement algorithm using a clock algorithm or a local FIFO page replacement algorithm. Would it be possible to implement a local LRU approximation page replacement algorithm? If not, explain why. If so, describe how it would be implemented.

- 4) (9 points) For each of the parts of the address space listed below, describe what part of the system determines the location in memory (i.e. the virtual address) and the layout in memory (i.e. the address in relation to other objects in the same section). The choices of system parts are the compiler, linker, and OS/runtime library.
- (a) A statically allocated C data structure.
 - (b) A dynamically allocated C data structure (e.g. malloc).
 - (c) A local variable of a C subroutine.
- For each object describe the system part (compiler, linker, OS/runtime) that sets the layout and the system part that determines the location. They may be different.

- 5) (6 points) Describe the advantages and disadvantages of processor affinity CPU scheduling.

- 6) (7 points) What is the condition needed for deadlock that assigning rank ordering to locks is intended to avoid?

7) (6 points) Describe the problem with multiple threads executing the C statement:

```
{  
    static int I;  
    I++;  
}
```