

CS140 Operating Systems and Systems Programming Midterm Exam

February 12th, 2003

(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 7 pages.

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
Total	

3) (4 points) Is it possible to have a machine that has a physical address space with more address bits than the total number of bits in the machine's page table entry? Explain your answer.

4) (4 points) Explain how reference bits can be emulated on an architecture that has a normal paging system but lacks hardware support for reference bits.

- 5) (6 points) Explain how the following algorithms decide how much memory to give a process:
- a) page fault frequency
 - b) working set algorithm

- 6) (4 points) Would it ever make sense to use a LRU page replacement algorithm with a per-process local replacement policy? Justify your answer.

- 7) (4 points) Is the following statement true or false? Segmentation has less internal fragmentation than standard paging schemes while paging schemes have less external fragmentation than segmentation schemes. Be sure to justify your answer.

- 8) (3 points) Does busy waiting ever make sense on a uniprocessor?

- 9) (4 points) A leading CPU manufacture has added a capability they call superthreading to their microprocessor. With superthreading, a single CPU can be made to look like a two-CPU multiprocessor. Since the two “logical” processors share the same physical processor, only one of the logical processors is running at any instant in time. The architecture supports extremely fast context switching where the CPU can switch instantly between running logical processors at any instruction boundary making it appear to the software that both processors are running at the same time even though they are really not. Given there is only one real CPU, would it ever make sense for an operating system running with superthreading enabled to use spin locks or should blocking locks always be used? Explain your answer.

- 10) (3 points) Explain why an operating system can save less state on a system call trap than on a page fault trap.

11) (4 points) Describe how wait-free or non-blocking synchronization allows processes to synchronize without locks.

12) (4 points) Describe how a multilevel feedback queue CPU scheduler segregates CPU-bound jobs from I/O-bound jobs.