Stanford University
Computer Science Department
CS 240 Quiz 3
Spring 2003

June 3, 2003

This is an open-book exam. You have 50 minutes to answer all questions. Write all of your answers directly on the paper. Make your answers as concise as possible. Sentence fragments ok.

NOTE: We will take off points if a correct answer also includes incorrect or irrelevant information. (I.e., don't put in everything you know in hopes of saying the correct buzzword.)

Question	Score
1 - 2	
3 - 4	
5 - 6	
7	
total	

Stanford University Honor Code

In accordance with both the letter and the spirit of the Honor Code, I did not cheat on this exam nor will I assist someone else cheating.

Name and Stanford ID:

Signature:

Answer all of the following questions and, in a sentence or two, say *why* your answer holds. Your answer should be concrete enough that it is readily distinguishable from someone who has jotted down a string of high-level buzzwords.

1. The google search paper claims that a lot of care has been spent on data structures minimizing disk seeks. Give two examples supporting this claim (besides the obvious idea of caching).

2. The google file system paper claims that each chunk has an associated version number that is periodically incremented to detect stale replicas. Is it a problem if a replica goes down and comes back up between these increments? (Be very concrete as to why this does or does not cause problems.)

3. Write an MC extension that checks the rule "if a pointer is checked against null, do not dereference it." You may assume that the system will kill variables that are reassigned.

4. Pick some useful optimization from one of the papers that we discussed this quarter (besides the superpage paper) and describe how to implement it within a virtual machine monitor. Be concrete as to any difficulties that come from doing it in the the VMM rather than in the OS itself.

Question 5: Yes, we have no RAID (20 points)

1. (5 points) Assume you have a simple RAID system: four data disks (A,B,C,D) and one dedicated parity disk (P) that holds the bitwise xor of the data disks ($P = A\ xor\ B\ xor\ C\ xor\ D$).

   If the OS crashes while writing to the RAID system, what do you have to do after the crash to recover to a good state? What happens if a disk blows up during this recovery? (Assume that all redundant information lives on the parity disk and that you do not have a log.)

2. (10 points) Why might a log structured file system (LFS) run substantially faster than an update-in-place file system such as FFS on a RAID system? Give two weak points of LFS: do you expect these to be worse or better on a large RAID system?

3. (5 points) Jimbo likes your work and gives you an additional dedicated parity disk for the system above. Draw a picture with all six disks (two parity, four data) that shows how you'd organize your system so that it can survive *any* two disks blowing up.