Stanford University
Computer Science Department
CS 240 Quiz 2
Spring 2003

May 15, 2003

This is an open-book exam. You have 50 minutes to answer seven 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 the following seven questions and, in a sentence or two, say *why* your answer holds. The first six questions are worth 5 points each, the last 10 points.

1. Assume you can buy one of (1) infinitely fast disk, (2) infinitely fast network, or (3) infinitely fast CPU. Give an ordering of most to least important for the webservers and workloads described in the FLASH paper Figures 9 and 10.

2. Will a guest OS be more or less susceptible to livelock when running on VMware? Give two examples where the VMM can or cannot detect when a guest OS is livelocked or overloaded.

3. Suppose Jimbo wants to test an NFS server for livelock and runs a client with this loop:

```
while(1){
  send NFS READ RPC;
  wait for response;
}
```

Is the NFS server probably subject to livelock? What happens as Jimbo increases the number of client machines running this loop?

4. Does the end-to-end argument imply that lower levels can do a bad job?

5. You start sending NFS operations over a reliable, in-order, duplicate-suppressing network stream (e.g., TCP). Jimbo claims that this will fix all the non-idempotent problems of using NFS over UDP. Is this true? Give your intuition or a concrete counter example.

6. In LBFS, is the "count" returned by `GETHASH` redundant? What about the "count" taken by `CONDWRITE`? Give your reasoning why or why not.

7. (10 points) You notice that leases inherently involve request-response messages. You decide to try to save network traffic by building leases on top of UDP and doing your own retransmission by setting a timer and retransmitting if you didn't get the appropriate reply. When the server grants a read lease, what should it do on retranmission? If a client gets a write revocation for a lease it does not hold, what should it do? Are there other interesting cases?