Systems Comprehensive Exam, Spring 2006

January 10, 2005

1 Instructions

This is a closed-book, closed-notes exam with a total of 100 points. You may not use any external source for answering these questions, including but not limited to the Internet, books, notes, or othee people. Please direct any questions about this exam to Professor Bridges. Professor Bridges may be reached either in person in his office in 301B Farris, by phone at 277-3032 or 363-8798, or by email at `bridges@cs.unm.edu`. Turn your exam in to Professor Bridges or the front office by 5:00 PM MDT on Tuesday, January 10, 2006. Exams *will not* be accepted after this time except by prior arrangement with Professor Bridges.

Type or write your answers to the stated number of questions in each of the following three sections. Make any *reasonable* assumptions necessary to answer the question, but be sure to state any assumptions that you make.

2 Short Answer - Answer 3 of 4 (30 points)

Briefly answer 3 of the following 4 questions. Your answer should be no longer than *one* paragraph.

1. What is the *clock algorithm* used for in operating system implementation?
2. What are the advantages and disadvantages of CSMA/CD and token ring media access to high-bandwidth and low-bandwidth customers on a network?
3. Provide a simple example of how an interrupted (crashed) file system operating can leave the file system in an inconsistent state that must be fixed via `fsck` or journal replay in a UNIX-like file system.
4. What is cache affinity in the context of job scheduling?

3 Medium Answer - Answer 2 of 3 (40 points)

Provide detailed answers to two of the following three questions. Be sure to state any assumptions you make and to fully justify your answers. Limit your answers to approximately one to two pages in length.

1. **Computer Architecture and the Memory Wall.** A number of different architectural proposals have been made for dealing with the so-called *memory wall* problem. One of these, processors-in-memory (PIM) systems, suggests placing processing elements into each memory chip (e.g. DIMMS), so that they can get full access to memory bandwidths of the local memory instead of being limited by pin-out bandwidths.
   (a) Carefully define what is meant by the memory wall problem.
   (b) How have normal processor architectures (e.g. Intel x86/Motorola PowerPC) attempted to deal with this problem, and why is or isn't this approach sufficient?

(c) Does a PIM architecture like that described above address the memory wall problem, and what new difficulties does it present?

2. **Optimistic Synchronization and Concurrency Control.** Optimistic approaches to synchronization and concurrency control are frequently used in system design and implementation of asynchronous and parallel systems, from operating systems to databases to parallel simulations. Use a specific example of optimistic synchronization of your choosing to answer the following questions:

   (a) How does the optimistic synchronization and concurrecy control differ from the normal pessimistic approach both in general and in your example?

   (b) What are the general advantages and disadvantages of the optimistic and pessimistic approaches? For your example, under what conditions would an optimistic and pessimistic approaches be preferable?

4 Design - Answer 1 of 2 (30 points)

Provide a *full* and *detailed* answer to one of the following two questions. Be sure to state any assumptions you make and to fully justify your answer.

4.1 Flying Video on Demand

4.2 Disconnection in Sensor Systems

Networked sensor systems are crucial to emergency response in large scale disasters such as Hurricane Katrina. Such sensors may include water level/quality, weather, or photographic feeds on cell or telephone towers in cities. Unfortunately, such disasters tend to cause network disconnection, isolating each portion of the sensor network from central databases that normally collect and collate data from these systems. Without support for dealing with disconnection, such systems become useless in the situations in which they are potentially most helpful.

Consider a system where each sensor system has local power (e.g. solar/battery), limited storage, and connectivity to first responders in its immediate physical vicinity. The primary goal of this system is to make local sensor information available to local first responders while it logs updates to send to central facilities when long-range connectivity becomes available.

Such a system is similar in ways to a disconnected/weakly-connected file systems like Coda. Compare and contrast approaches for utilizing or dealing with the following issues with how they are handled in systems like Coda:

- Replication and consistency
- Caching
- Transactional semantics