CS 574

Final Exam

December 10, 2007

The following exam is open book and open notes. You may feel free to use whatever additional reference material you wish, but **no electronic aids** are allowed. Please note the following instructions. There will be a ten point deduction for failure to comply with them:

- start each problem on a new sheet of paper. It's likely that the easiest way to turn in Question 7 is to fill it in on the exam paper and turn that in.
- write your Banner ID number, but not your name, on each sheet of paper you turn in.

Also, please note the following:

- show your work whenever appropriate. There can be no partial credit unless you show how you derived your answers.
- be succinct. You may lose points for facts that, while true, are not relevant to the question at hand.

You have until 12:30 to finish the exam. The questions are equally weighted.

1. Both Minix and Xen attempt to limit the damage that can be done by a device driver containing a bug.
   (a) What is the mechanism used by each to limit the damage?
   (b) Suppose the mouse driver has a bug. How much damage can it cause (*e.g.* is there a set of processes you can identify that would crash, could it bring down the whole system, etc), for each?
2. CPU schedulers for operating systems tend to use very complex algorithms, with lots of heuristics to try to enhance response to interactive inputs. CPU schedulers for virtual machines tend to be very simple, perhaps even to the point of simply alternating time slices between the virtual machines being hosted. Why are these very different choices both appropriate?
3. Let's consider a few of the differences between D-Bus and RPC.
   (a) How is D-Bus's execution model better suited to parallel execution than RPC's?
   (b) How is RPC better suited to a heterogeneous network environment thant D-Bus?
   (c) How does D-Bus differ from RPC in locating a service to communicate with?
4. One of the differences between Alewife and DASH is that when a cache line is dirty and a new processor reads it, the cache line ends up in different states in the two systems.
   (a) What states does it end up in, for each of them? And which processors have valid copies? Assume the processor that previously had it is named processor P_{old} and the new processor is named processor P_{new}.
   (b) What is the expected program behavior (*i.e.*, what do you think the programs running on the processors are going to do in the near future) that makes the DASH solution look like a good one? What is the expected behavior that makes the Alewife solution look like a good one?
5. Suppose you wanted to add process migration to a system that had distributed shared memory. What extra difficulties do you anticipate this might cause the memory consistency algorithm?
6. Compute the CRC of the message string `100111101101`
   using the polynomial $1+X+X^3$
   The input string is presented using the modern convention of most significant bit first, which is backwards from that used in the paper.

7. The following figure shows a portion of an OceanStore filesystem, using the same conventions as in Figure 2 of the OceanStore paper (the fact that two of the rounded boxes contain the same mask is deliberate).

   (a) Fill in the rectangular boxes.

   (b) Describe how a query for a file that hashes to `010101` is routed through the network.