CS425/CS525 Final Exam

May 8th, 2008

Write your answers in the blue book(s). Justify your answers. Work alone. Do not use any notes or books.

There are four problems on this exam, each worth 20 points, for a total of 80 points. You have approximately three hours to complete this exam.

1 Message passing without failures (20 points)

Suppose you have an asynchronous message-passing system with a complete communication graph, unique node identities, and no failures. Show that any deterministic atomic shared-memory object can be simulated in this model, or give an example of a shared-memory object that can't be simulated.

2 A ring buffer (20 points)

Suppose you are given a *ring buffer object* that consists of $k \geq 1$ memory locations $a[0] \ldots a[k-1]$ with an atomic *shift-and-fetch* operation that takes an argument v and (a) shifts v into the buffer, so that $a[i] \leftarrow a[i+1]$ for each i less than $k-1$ and $a[k-1] \leftarrow v$; and (b) returns a snapshot of the new contents of the array (after the shift).

What is the consensus number of this object as a function of k?

3 Leader election on a torus (20 points)

An $n \times n$ torus is a graph consisting of n^2 nodes, where each node (i, j), $0 \leq i, j \leq n-1$, is connected to nodes $(i-1, j), (i+1, j), (i, j-1)$, and $(i, j+1)$, where all computation is done mod n.

Suppose you have an asynchronous message-passing system with a communication graph in the form of an $n \times n$ torus. Suppose further that each node has a unique identifier (some large natural number) but doesn't know the value of n. Give an algorithm for leader election in this model with the best message complexity you can come up with.

4 An overlay network (20 points)

A collection of n nodes—in an asynchronous message-passing system with a connected, bidirectional communications graph with $O(1)$ links per node—wish to engage in some strictly legitimate file-sharing. Each node starts with some input pair (k, v), where k is a key and v is a value, and the search problem is to find the value v corresponding to a particular key k.

1. Suppose that we can't do any preparation ahead of time. Give an algorithm for searching with the smallest asymptotic worst-case message complexity you can find as a function of n. You may assume that there are no limits on time complexity, message size, or storage space at each node.

2. Suppose now that some designated leader node can initiate a protocol ahead of time to pre-process the data in the nodes before any query is initiated. Give a pre-processing algorithm (that does not depend on which key is eventually searched for) and associated search algorithm such that the search algorithm minimizes the asymptotic worst-case message complexity. Here you may assume that there are no limits on time complexity, message size, or storage space for either algorithm, and that you don't care about the message complexity of the pre-processing algorithm.

3. Give the best lower bound you can on the total message complexity of the pre-processing and search algorithms in the case above.