Dr. Christian Scheideler
Johns Hopkins University

Baltimore, Oct. 22 2003

Theory of Network Communication

Fall 2003

Assignment 6

Problem 13 (5 points)**:**
Prove that when using the cut-and-paste strategy with n units, the location of every object can be determined in at most $O(\log n)$ computations. (3 points)
(Hint: It suffices to show that every object is replaced at most $O(\log n)$ times when increasing the system one by one from 1 to n units; replaying these hops gives the time bound for the computations. In order to show that the distances bridged by replacements are essentially increasing exponentially, look at any two consecutive replacements of a data item.) (3 points)

In addition to this, show that for every perfectly faithful strategy it holds: If the number of units grows one by one from 1 to n, then on average an object has to be replaced $\Omega(\log n)$ times.
(2 points)
(Hint: compute the total number of object replacements, and divide it by the number of objects.)

Problem 14 (5 points)**:**
Consider the following simplification of the nearest neighbor strategy:

There is a hash function $h : U \to [0, 1)$ for the data elements and a hash function $g : \{1, \ldots, n\} \to [0, 1)$ for the storage units. Each data element d is placed at the storage unit u with minimum $g(u)$ so that $g(u) \geq h(d)$ (i.e. u is the closest successor of d in the hash space). If there is no such $g(u)$, take the unit u with smallest $g(u)$ among all the units.

Suppose now that we have 10 storage units with hash values 0.0, 0.1, 0.2, ..., 0.9. Write a C++ program in the Spheres environment that organizes these units in an undirected cycle so that the units are ordered in this cycle according to their hash values. Implement a Search and Reply method so that Search for value $x \in [0, 1)$ initiated at a unit u sends a search message (along edges of the cycle) to the unit v responsible for x (according to our strategy above). As soon as v has received the search message, it sends a reply message with its hash value back to u. As soon as u has received the reply message, it outputs the received hash value.

You can simply implement the methods without using the Sync mechanism. That should make your task easier.

Test the program by starting a single search request at any unit u for any value x. Print out the program and the results of 2 tests.