CS 466/666 Spring 2014

Assignment 2

Due Noon, Monday May 26, 2014

1. [8 marks] Suppose you have the elements 1, …,10 stored in that order in a list.
   (a) How many elements inspections occur in performing the following sequence of searches: 3, 10, 3, 4, 5, 3, 5, 9, 8, 3.
   (b) Find the static ordering of the list that minimizes the number of comparisons to perform this sequence of requests for the above sequence. How many comparisons does it use?
   (c) Give a request sequence in which the move to front heuristic does better than the static optimal for the proposed sequence of requests. Justify this claim.

2. [8 marks] In class, we discussed a "doubling binary search" technique for finding the "min-max" approximation to the optimal binary search tree, and claimed that it ran in O(n) time even though it involved up to n "binary searches". The key point was that the search for the root of a given subrange took O(1 + lg(v)) time, where v is the position of the root relative to the subrange. (i.e. The root is the v^{th} node from the left or right, whichever is less). Prove the method takes O(n) time. To do so you may choose do it in the following two parts.
   a) To simplify things, first assume that an individual search takes Ceiling(lg v) steps, and show that the cost of determining the entire tree is O(n) (i.e. the amortized cost of a search is O(1)) [Hint: It is probably easiest to show an explicit constant, c, such that the runtime is less than cn]
   b) Complete the proof by using part a) to show that the entire process takes O(n) time even if the individual search takes $p + q$ Ceiling(lg v) time (for positive constants p and q.