CS 181b

Advanced Topics in Algorithms

Spring 2002

Problem Set 3a

Due Thursday, February 7 in class

These problems are not particularly tricky but they do require that you understand the analysis we did in class and modify it appropriately. Please explain your analysis carefully. You may refer to the proof we did in class without redoing it, but make sure that you are very explicit and clear about what you are referring to and what you are modifying.

1. **[15 Points] Fractional MTF for the List Access Problem.** Consider a variant of the MTF online algorithm called MTF_d where d is a positive integer. MTF_d is like MTF except that when a FIND operation finds the element at location i, the algorithm moves it $i/d - 1$ positions closer to the front of the list. Notice that MTF_1 is regular old MTF. Show that MTF_d is $2d$-competitive.

2. **[15 Points] The Dynamic List Access Problem.** In class we looked at the MTF online algorithm for the List Access Problem. Specifically, we looked at the **Static** List Access Problem where only FINDs were allowed - there were no INSERT and DELETE operations permitted in the request sequence.

   Now, consider the more general **Dynamic** List Access Problem which differs from the static version in three ways:

   (a) Any FIND operation may fail. That is, the item might not be found. In this case, the actual cost is $\ell + 1$ where ℓ was the length of the list at the time that the FIND was performed.

   (b) INSERT operations are permitted in the request sequence. An inserted object is placed at the end of the list. If the length of the list prior to the INSERT was ℓ, then the actual cost of this operation is $\ell + 1$. After inserting the element at cost $\ell + 1$, you may move it to any position earlier in the list at no cost (just like in a successful FIND operation).

   (c) DELETE operations are permitted in the request sequence. The cost of the DELETE is simply the cost of finding the object. (However, the potential function changes as a consequence of the DELETE, so a DELETE differs from a FIND in this way!)

   Show that when MTF is applied to the Dynamic List Access Problem, it still maintains a competitive ratio of 2. (That is, it is 2-competitive with respect to an optimal offline algorithm.)