15-854 Approximation and Online Algorithms

Homework # 4 **due: March 15, 2000**

Groundrules: Same as before.

Exercises:

1. Consider the paging algorithm that, on a page fault, just throws out a random page from the cache. Prove that the competitive ratio of this algorithm is $\Omega(k)$.

2. (Exercise 4.6 in Borodin & El-Yaniv) We proved that the Marking algorithm is H_k-competitive for the paging problem when $N = k + 1$ (k is the size of the cache, N is the total number of pages in slow memory, and $H_k = 1 + 1/2 + \ldots + 1/k$). We also proved that the Marking algorithm is $2H_k$-competitive for general values of N. Give an example of a request sequence for $k = 2$ and $N = 4$ where the Marking algorithm is *not* H_k-competitive.

3. (Exercise 1.1 in Borodin & El-Yaniv) Recall that in the list accessing problem, we can swap pairs of neighbors for a cost of \$1 each (these are called "paid exchanges"), but when an item x is accessed, we can also move x up in the list for free. It is tempting to conjecture that the optimal offline algorithm never needs to use paid exchanges. Give an example of a request sequence showing that this is false (i.e., in this example you need to use paid exchanges to be optimal). Hint: you can do this with 4 requests to a list of 3 elements.