CSE 638 Analysis of Algorithms Spring 2007

Problem Set 1

Lecturer: Michael A. Bender *Due: Monday, 2/26/07*

1. Please review how 2-3 trees, 2-4 trees, and traditional B-trees work. To ensure that you understand, please answer the following question: What is the amortized modification cost (insertions plus deletes) for 2-3 trees and for 2-4 trees? (That is, what is the cost for modifying the structure after subtracting the searching cost?) What is the amortized number of memory transfers for updating a B-tree.

2. Here we revisit matrix multiplication where matrices are stored in row-major (or column-major) order. If $N^2 = O(M)$, then we know that the cost is just $O(\lceil N^2/B \rceil)$, the cost to scan the matrix. For very large N, the cost is $O(N^3/B\sqrt{M})$. What happens in between?

3. Please construct a PMA where the number of gaps is only a $\Theta(1/\log N)$ fraction of the array. What is the inserts and delete performance now, both in terms of element moves and memory transfers?

4. In my presentation of the PMA, the leaves of the tree in the PMA still had capacity $\Theta(\log n)$. Why did we make the leaves have at least this size?

5. Please give as efficient an algorithm for rebalancing as you can. (1) Please give an algorithm for rebalancing in place that moves each element at most twice. (2) Can you find an algorithm for rebalancing that moves each element only once?