CS 5633 Analysis of Algorithms – Spring 09

2/26/09

6. Homework

Due **3/5/09** before class

1. **B-trees (6 points)**
   (a) What is the maximum number of keys that can be stored in a B-tree with minimum degree k and height h? Your answer should depend on k and h.
   (b) The CPU time of B-Tree-Search is $O(k \log_k n)$. Show that if B-Tree-Search is changed to use **binary search** instead of linear search on the key then the CPU time is only $O(\log n)$, which is independent of k.

2. **B-trees (3 points)**
   Why does the minimum-degree parameter k of a B-tree have to be greater then 1? What would happen if we set $k = 1$?

3. **Constructing range trees (10 points)**
   (a) (5 points) Give pseudo code for a recursive procedure that constructs a 1D range tree for a **sorted** set of n numbers in $O(n)$ time. Justify your answer.
   (b) (1 point) Give pseudo code for a procedure that constructs a 1D range tree for a (possibly unsorted) set of n numbers in $O(n \log n)$ time. Justify your answer.
   (c) (5 points) Give pseudo code for a recursive procedure that constructs a 2D range tree for a set of n 2D points in $O(n \log n)$ time. Justify your answer. *(Hint: Use an approach similar to mergesort. Also use the ideas of problems (a) and (b).)*

4. **Range tree counting queries (5 points)**
   Show how to augment a d-dimensional range tree (without fractional cascading) of n elements such that range **counting** queries can be answered in $O(\log^d n)$ time. Argue that your augmentation does not change the asymptotic preprocessing time and the asymptotic space complexity. *Hint: Start with $d = 1$, and then generalize to higher dimensions.*