CS 5633 Analysis of Algorithms – Spring 08

2/26/08

6. Homework

Due **3/6/08** 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 $k = 1$?

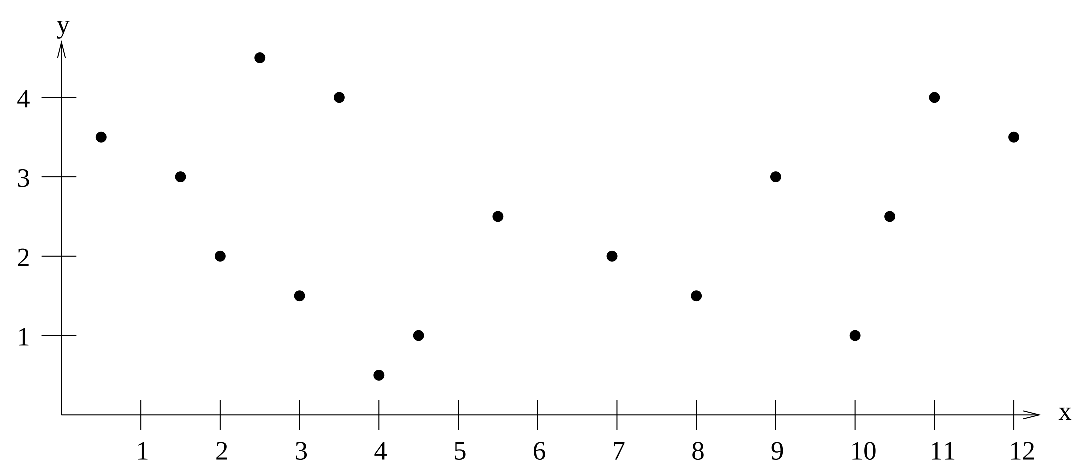
3. **Standard deviation (8 points)**
   The *standard deviation* of a set of n numbers $a_1, \ldots, a_n$ is defined as

$$\sqrt{\frac{\sum_{i=1}^{n} a_i^2 - (\sum_{i=1}^{n} a_i)^2/n}{n-1}}\,.$$

   (a) Show how to preprocess n numbers in linear time (and linear space) such that the standard deviation of any contiguous subarray can be computed in constant time. *(Hint: This is just a warmup exercise for b). It does not require any complicated data structures; only a bit of arithmetic.)*

   (b) Show how to preprocess n numbers in $O(n \log n)$ time such that the following queries can be performed in time $O(\log n)$: Insert a number, delete a number, return the standard deviation of a contiguous subarray. *(Hint: Augment a balanced search tree.)*

4. **Range trees (9 points)**
   Let $P = \{(0.5, 3.5), (1.5, 3), (2, 2), (2.5, 4.5), (3, 1.5), (3.5, 4), (4, 0.5), (4.5, 1), (5.5, 2.5), (7, 2), (8, 1.5), (9, 3), (10, 1), (10.5, 2.5), (11, 4), (12, 3.5)\}$ be a set of two-dimensional points.



FLIP OVER TO BACK PAGE ⟹

a) (3 points) Draw the primary range tree of P. (The keys are the x-coordinates; the leaves store the two-dimensional points, or pointers to them, but not just the x-coordinates).

b) (3 points) Draw all secondary range trees. (The keys are the y-coordinates; the leaves store the two-dimensional points, or pointers to them, but not just the x-coordinates). Notice that since there are duplicate y-coordinates the trees are not unique.

c) (3 points) Consider the query rectangle $[x_1 = 1, x_2 = 6] \times [y_1 = 1.5, y_2 = 4]$. Show how the range reporting query which prints out all points in the query rectangle proceeds in the range tree:

- Show the split nodes (in the primary tree, and in the secondary trees).
- Show the search paths (in the primary tree, and in the secondary trees).
- Show which secondary trees are queried.
- Show which points are output (mark the corresponding leaves in the secondary trees).

Below is another copy of the point set: