CS 5633 Analysis of Algorithms – Spring 05

2/22/05

3. Homework

Due **3/1/05** before class

1. **Quicksort decision tree (5 points)**
   The decision tree on the slides in class was for insertion sort of three elements. Draw the decision tree for deterministic quicksort, with the 1st array element as the pivot, for three elements.

2. **Leafs in decision trees (6 points)**
   Consider a decision tree for a comparison sort of n elements, and pick an arbitrary leaf l.

   a) What does l represent?

   b) What does the path from l to the root represent?

   c) What does **the length** of the path from l to the root represent?

3. **Radix sort and counting sort (6 points)**

   a) Given n numbers between 0 and $n^2 - 1$. How fast does counting sort sort them? How fast does radix sort sort them? Which one is the fastest?

   b) Given n numbers between 0 and $2^n - 1$. How fast does counting sort sort them? How fast does radix sort sort them? Which one is the fastest?

   c) Given n numbers between 0 and $\log n - 1$. How fast does counting sort sort them? How fast does radix sort sort them? Which one is the fastest?

4. **Radix sort with most significant digit first (5 points)**
   Try to sort the numbers

   $$436, 329, 318, 457, 355, 435, 316, 555, 327, 456, 444$$

   using radix sort but starting with the **most** significant digit (so, from left to right, not from right to left).

   Why would a program that implements this strategy be much more complicated than the radix sort that starts with the least significant digit? (Hint: What kind of variables or data structures would you have to maintain?)

5. **Randomized select (5 points)**
   Consider a version of deterministic select where you divide the elements into groups of 3. Show where you run into problems when trying to prove a runtime of $O(n)$.