COL 351

TUTORIAL SHEET 5

1. You are given a positive integer N. You want to reach N by starting from 1, and performing one of the following two operations at each step: (i) increment the current number by 1, or (ii) double the current number. For example, if $N = 10$, you can start with 1, and then go in the sequence $1, 2, 4, 8, 9, 10$, or $1, 2, 4, 5, 10$. Give an efficient algorithm, which given the number N finds the minimum number of such operations to go from 1 to N.

2. Suppose you are given two sets of n points, one set $\{p_1, \ldots, p_n\}$ on the $y = 0$ line and the other set $\{q_1, \ldots, q_n\}$ on the $y = 1$ line. Now we draw n line segments given by $[p_i, q_i], i = 1, \ldots, n$. Describe and analyze a divide-and-conquer algorithm to determine how many pairs of these line segments intersect, in $O(n \log n)$ time.

3. You are given a rooted binary tree T where each vertex v has an integer $val(v)$ stored in it (you can assume that all the integers involved are distinct). A vertex v is said to be a *local minimum* if $val(v) \leq val(w)$ for all the neighbours w of v (i.e., its value is at most the value at its children and at its parent). Show how to find a local minimum in $O(H)$ time, where H is the height of T.

   Solve the same problem when the graph is an $n \times n$ grid graph. An $n \times n$ grid graph has vertices labelled (i, j), where $1 \leq i, j \leq n$ and (i, j) is adjacent to $(i-1, j), (i+1, j), (i, j-1), (i, j+1)$ (with appropriate restrictions at the boundary). The time taken by the algorithm should be $O(n)$.

4. (a) Let $n = 2^l - 1$ for some positive integer l. Suppose someone claims to hold an unsorted array $A[1 \ldots n]$ of distinct l-bit strings; thus, exactly one l-bit string does not appear in A. Suppose further that the only way we can access A is by calling the function $FB(i, j)$, which returns the j^{th} bit of the string $A[i]$ in $O(1)$ time. Describe an algorithm to find the missing string in A using only $O(n)$ calls to FB.

   (b) (**Hard**) Now suppose $n = 2^l - k$ for some positive integers k and l, and again we are given an array $A[1 \ldots n]$ of distinct l-bit strings. Describe an algorithm to find the k strings that are missing from A using only $O(n \log k)$ calls to FB.

5. You are given n charged particles located at points with coordinates $\{1, 2, \ldots, n\}$ on the real line. The point at coordinate i has charge q_i. Note that the total Coulomb force on a particle at location i is given by

$$-\sum_{j<i} \frac{q_i q_j}{(i-j)^2} + \sum_{j>i} \frac{q_i q_j}{(i-j)^2}.$$

   Give an $O(n \log n)$ time algorithm to compute the total force on each of the particles.

6. We are given a sequence of n distinct integers $a_1, \ldots, a_n$ in an array. An inversion is defined as a pair (i, j) such that $i < j$ but $a_i > a_j$. Give an $O(n \log n)$ time to count the number of inversions in the array.

   A strong inversion is defined as a pair (i, j) such that $i < j$ but $a_i > 2a_j$. Give an $O(n \log n)$ time to count the number of strong inversions in the array.