600.363/463 Algorithms
Assignment 3
Due Sept 30, 2013

I. Design a dynamic programming algorithm for the following problem:

Given a sequence $a_1 \square_1 a_2 \square_2 \cdots a_{n-1} \square_{n-1} a_n$, in which each a_i is a postive integer and each $\square_i$ is '+' or '-', compute a parenthesization of the expression such that the resulting value is the maximum possible. It suffices to compute the resulting value instead of the parenthesization. Estimate its speed.

For example, if the given sequence is 3 - 4 -5, ((3-4)-5) results in -6 while (3-(4-5)) results in 4. The second parenthesization results in the maximum possible value, and the output is 4.

II. Let A be an $m \times n$ array of numbers. In phase 1, we sort the rows and then in phase 2, we sort the columns. Prove that after both the phases are completed, the rows remain in sorted order.

III. Let $S_1, S_2, \cdots, S_m$ be nonempty subsets of $\{1, 2, \cdots, n\}$, and let the total number of elements in all the $S_i s$ be n. Design an $O(n)$ step algorithm for sorting the $S_i s$.

IV. In the standard Heap Sort, given any n numbers $a_1, a_2, \cdots, a_n$, we first arranged them into a complete binary tree and then satisfied the heap property in the order $a_n, a_{n-1}, \cdots, a_1$. If we want to heapify in the order $a_1, a_2, \cdots, a_n$, describe an appropriate algorithm and estimate its speed. (In this algorithm, for any $i \geq 1$, after a heap is constructed for $a_1, a_2, \cdots, a_i$, the next number a_{i+1} is brought into the heap in a bottom-up fashion.)