Report on "Introduction to Algorithms and Data Structures"

January 22–26, 2018

Ryuhei Uehara (uehara@jaist.ac.jp)

Do not forget to write your name, student ID, problems, and answers on your report. In PDF format, any style (scanned hand-written, Word, etc) is acceptable. Choose any problems that make 100 in total, then answer them in English. (If you choose more, I'll take from better scores.)

Deadline: February 11 (Sun).

Problem 1: (10 pts) Let x, y be two integer variables whose values are already set. We perform the following substitutions. Give some concrete values and check the sequence of substitutions. What does this sequence aim at?

$x = x + y;$

$y = x - y;$

$x = x - y;$

Problem 2: (10 pts) Prove the following

$$26n^2 + n + 2018 \in O(n^3)$$
$$26n^2 + n + 2018 \notin O(n)$$

Problem 3: (10 pts) Prove the following two sets are the same sets.

$$O(\log_3 n) = O(\log_{100} n)$$

Problem 4: (10 pts) Assume that you design a quick sort so that the pivot is the first element in each step. Then explain when this quick sort runs slower than expected.

Problem 5: (10pts) Counting sort runs in $O(n)$ time. However, quick sort is more popular than counting sort although it runs $O(n \log n)$ time, which is slower. Explain why.

Problem 6: (20 pts) For any given string s of "(" and ")" like $s =$()(()))(), we want to check if this string is legal or not in a natural manner. Describe and show the sketch of an algorithm and data structure to solve this problem. Estimate its running time also.

Problem 7: (20 pts) Let data 1,2,3,4,5,6,7,8,9,10 are given in this order. Then construct the AVL-tree and the binary search tree for this set of data, and compair their levels.

Problem 8: (20 pts) In the course, you learnt several sorting algorithms. There are two groups; stable sorting and non-stable sorting. Indicate stable or non-stable for each sorting. For non-stable sorting, explain when it is not stable.

Problem 9: (20 pts) Suppose that the array $a[0], a[1], \ldots, a[n-1]$ consists of n real numbers. We like to compute the function $f(x) = a[0] + a[1]x + a[2]x^2 + \cdots + a[n-1]x^{n-1}$. Consider the following two algorithms:

1. Following the definition, compute $a[0] + a[1] \times x + a[2] \times x \times x + a[3] \times x \times x \times x + \cdots + a[n-1] \times x \times \cdots \times x$ step by step.

2. Compute after the folllwing modification: $a[0] + x \times (a[1] + x \times (a[2] + x \times (a[3] + x \times (a[4] + \cdots + x \times (a[n-2] + x \times a[n-1])))))$

Evaluate the number of summation and multiplication operations respectively as functions of n, and discuss which is a better way.

Problem 10: (30 pts) You want to shuffle the data which is in $a[0], a[1], \ldots, a[n-1]$ by randomization. You can use a random generator function $rand(k)$ that returns an integer i with $0 \leq i < k$ uniformly at random. Then, show a shuffle algorithm for $a[]$. That is, the algorithm outputs each possible permutations of $a[]$ with the same probability. Analyze time complexity of the algorithm. (Hint: there are several ways, but there exists a simple $O(n)$ time algorithm.)