

# I111 アルゴリズムとデータ構造 (Algorithms and Data Structures)

## レポート課題 2(Report 2)

2015, Term 1-1

上原 隆平 (Ryuhei Uehara)(Room I67b, uehara@jaist.ac.jp)

出題 (Propose): May 15 (Fri)

提出期限 (Deadline): May 27 (Wed), 10:50am.

注意 (Note): レポートには氏名, 学生番号, 問題, 解答を忘れずに書くこと. 電子メールで PDF ファイルを送って来ててもよい. Word ファイルは不可. (Do not forget to write your name, student ID, problems, and answers on your report. You can send your report by email in PDF file format. The report in Word file format is not accepted.)

以下の問題から適当に選んで日本語か英語で答えよ. 各 5 点か 10 点か 15 点で合計 15 点満点. (Answer some of the following problems in English or in Japanese. Each makes 5, 10, or 15 points, and 15 points in total.)

問題 1 (Problem 1): (5pts) 以下を証明せよ. (Prove the following)

$$\frac{n}{2} \log \frac{n}{2} \in \Theta(n \log n)$$

問題 2 (Problem 2): (5pts) クイックソートは, ピボットの選び方がまずいと遅くなる. 具体的に, どういう状況で遅くなるか説明せよ. (On the quick sort, if you do not choose pivot properly, it runs slower. Explain when it runs slower? Explain how to choose each pivot, and what data that makes it slower.)

問題 3 (Problem 3): (10pts) 値 1 を持つノードのみを持つ AVL 木に, 2,3,4,5,6,7,8,9,10 とデータを追加せよ. また, 普通の 2 分探索木にも同じことをして, できる木の深さを比較せよ. (For an AVL-tree with one node of data 1, add 2,3,4,5,6,7,8,9, and 10 as data. Repeat the same for the ordinary binary search tree, and compare their tree depths.)

問題 4 (Problem 4): (10pts) 講義で学んだソートアルゴリズムのうち, 安定なソートはどれか. 安定でないソートについて, どういう場合に安定でないか示せ. (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.)

問題 5 (Problem 5): (15pts) 計数ソートは線形時間で動作するが, それよりも遅いクイックソートの方がよく使われるのは何故か. また, 計数ソートはマシンモデルが RAM モデルであることに強く依存している. どこがどう依存しているのか, 議論せよ. (Counting sorting runs in linear time. However, quick sorting is more popular than counting sorting although it is slower. Explain why. The claim that “counting sorting runs in linear time” depends on the machine model which is RAM model. Make a discussion how it depends on.)