

I470 実践的アルゴリズム理論 (Theory of Advanced Algorithms)

Report (1)

2018 年度 2-1(10月-11月) 上原 隆平 (uehara@jaist.ac.jp)

出題 (Propose): 10月19日(金)

提出 (Deadline): 10月31日(水) 9:00

注意 (Note): レポートには氏名, 学生番号, 問題番号, 解答を, すべて書くこと. 紙は A4 で左上をホチキス止めすること. 片面使用でも両面使用でもよい. JAIST のアカウントから電子メールで PDF ファイルでレポートを出してもよい (Word は不可). その場合は件名を “I470 Report 1 (学生番号)” とし, メール本文にも学生番号と氏名を明記して, ファイル名は “学生番号.pdf” とすること. (Do not forget to write your name, student ID, problem ID, and answers on your report. The size of paper is A4, and staple them at the top left. You can use one side or both sides of paper. You can submit your report by email in PDF format from JAIST account (.doc or .docx is not allowed). In that case, the subject should be “I469 Report 1 (Student ID)”, you write your student ID and your name in the email, and the file name should be “StudentID.pdf”.)

以下の問題から 2 問以上選んで解け (各 10 点). (Choose and solve two or more problems from below (10 points each).)

Problem 1: アルゴリズム P3-A6 では最大区間和の値を求めた. このとき, 区間和の値だけでなく, 区間そのもの (始点と終点) を求めるにはどうすればよいか. (In algorithm P3-A6, it computes the maximum value of an interval. In this algorithm, how can you compute not only the maximum value of an interval, but also the start and end indices of the interval?)

Problem 2: コインが C_1 円, C_2 円, C_3 円と 3 種類あり, 貪欲法で支払うと, 支払うコインの枚数が最小になるとする. ここで $C_1 < C_2 < C_3$ とする. このとき, C_1, C_2, C_3 が満たすべき条件を示せ. (Assume that we have three different kinds of coins C_1 -yen, C_2 -yen, and C_3 -yen, and when we use greedy algorithm to pay, the number of coins is minimized. Let $C_1 < C_2 < C_3$. Then show the conditions that C_1, C_2, C_3 should satisfy.)

Problem 3: Kruskal のアルゴリズムで最小全域木を求めるときと, Prim のアルゴリズムで最小全域木を見つけるときとで, 解が異なることがあるか. ないならそれを証明し, あるならその具体例を示せ. ただしそれぞれのアルゴリズムの正当性はすでに証明済みとしてよい. (For finding the minimum spanning tree, is it possible that two solutions by Kruskal’s algorithm and Prim’s algorithm are different? If the answer is “no”, prove it. If the answer is “yes”, show an example. You can assume that the correctness of these algorithms has been already proved.)

Problem 4: 配列 $a[0], \dots, a[n-1]$ が与えられたとき, 次の条件を満たす添字の列 $(i_0, i_1, \dots, i_{k-1})$ を「長さ k の単調非減少列」と呼ぶ. (For a given array $a[0], \dots, a[n-1]$, the sequence $(i_0, i_1, \dots, i_{k-1})$ of indices is called a *monotone nondecreasing sequence*.)

- $i_0 < i_1 < \dots < i_{k-1}$
- $a[i_0] \leq a[i_1] \leq \dots \leq a[i_{k-1}]$

与えられた配列 $a[]$ の中で, 最長の単調非減少列を見つけるアルゴリズムを設計し, その実行時間を解析せよ. (For any given array $a[]$, design an algorithm that finds the maximum monotone nondecreasing sequence, and analyse its time complexity.)

Problem 5: i 番目のフィボナッチ数 F_i は、次のように定義される。(The i th Fibonacci number F_i is defined as follows:)

- $F_0 = F_1 = 1$
- $F_i = F_{i-1} + F_{i-2}$ for $i > 1$

一般の F_n を計算するアルゴリズムを設計し、その実行時間を示せ。設計と解析が重要なので、効率は良くても悪くてもよい。(Design an algorithm for computing F_n and show its running time. Since design and analysis are important, efficiency can be good or bad.)