I482F 実践的幾何アルゴリズム (Advanced Algorithms for Computational Geometry) Report (2)

2017 年度 III(11-12 月) 上原 隆平 (uehara@jaist.ac.jp)

出題 (Propose): 12 月 03 日 (日)

提出 (Deadline): 12 月 16 日 (土) 9:20

注意 (Note): レポートには氏名，学生番号，問題，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．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「I469 Report 2 (学生番号)」とし，添付ファイル名は「学生番号.pdf」とすること．(Do not forget to write your name, student ID, problems, 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. (.doc or .docx is not allowed). In that case, the subject should be "I469 Report 2 (Student ID)" and attached file name should be "student ID.pdf".)

以下の問題から 3 問以上選んで解け（各 10 点，25 点満点）．(Choose and solve three or more problems from below (10 points each, 25 points in total).)

問題 1: 凸多角形 P が頂点の列 $v_0, v_1, \ldots, v_{n-1}$ で与えられたとする．P の任意の 3 角形分割を考えると，頂点 v_0 に弦がつながっている場合と，つながっていない場合がある．頂点 v_0 に頂点がつながっていないとき，必ず v_1 と v_{n-1} をつなぐ弦が存在することを証明せよ．(Let P be a convex polygon desribed by a sequence $v_0, v_1, \ldots, v_{n-1}$ of vertices. For a triangulaton of P, the vertex v_0 is incident to a chord or not. Prove that there exists a chord joining v_1 and v_{n-1} when v_0 is not incident to a chord.)

問題 2: n 個の荷物 $\{o_1, o_2, \ldots, o_n\}$ がそれぞれ自然数の重さ $w(o_i)$ をもつとする．ここで $C = (\sum_{i=1}^{n} w(o_i))/2$ が自然数だったとする．n 個の荷物をちょうど半分，つまり重さ C ずつに分割する方法があるかどうかを判定する動的計画法に基づくアルゴリズムを示せ．計算時間も示すこと．(Let $\{o_1, o_2, \ldots, o_n\}$ be the set of n objects of natural number weights $w(o_i)$. Suppose that $C = (\sum_{i=1}^{n} w(o_i))/2$ is also a natural number. Then show an algorithm that decides if you can partition the set into half, that is, two subsets of weights C based on dynamic programming. Show the running time for the algorithm also.)

問題 3: 生起確率が $p(0 < p < 1)$ である事象を，成功するまで繰り返す．このときの試行の回数の期待値は $1/p$ であることを証明せよ．(We continue trying some event with probability $p(0 < p < 1)$ of occurrence until it succeeds. Then prove that the expected value of number of trials is $1/p$.)

問題 4: ある決定問題（つまり答は 0 か 1 としてよい）に対して，モンテカルロタイプの乱択アルゴリズム A が計算時間 $t(n)$ でいつでも停止して，確率 p で正解を返す（n は入力の長さ）とする．直感的には A を何度か実行して多数決を取れば，正解を返す確率を高めることができそうに思える．そこで以下のアルゴリズム A' を考えよう．

アルゴリズム A':

1. A を 2 回実行する．
2. 2 回の実行結果が一致すればそれを出力し，一致しなければステップ 1 に戻る

このアルゴリズム A' の計算時間の期待値 $t'(n)$ と正解を返す確率 p' を求めよ．$p' > p$ となるためには，確率 p に，どんな条件が必要か？(For a decision problem (i.e., the answer is “0” or “1”), we assume that a Monte Carlo type randomized algorithm A runs in $t(n)$ time and outputs a correct answer with probability p, where n is the length of the input. Intuitively, repeating the algorithm and taking majority, it seems that the correct answer will be obtained with higher probability. Therefore, we consider the following algorithm A':

Algorithm A':

1. Perform A twice.
2. If two outputs are the same, output it; otherwise, go to step 1.

Calculate the expected running time $t'(n)$ and the probability p' of correct answer of this algorithm A'. Show the condition of p that we will have $p' > p$.)

問題 5: 配列 $a[1], \ldots, a[n]$ と自然数 $k (1 \leq k \leq n)$ が与えられたとする．ここから k 個の要素をランダムに出力するアルゴリズムを示せ．アルゴリズムの計算時間も示すこと．ただし，同じ要素が 2 回以上出力されてはいけない．つまり k として n を与えると，配列 $a[]$ をシャッフルするアルゴリズムとなる．(Assume that an array $a[1], \ldots, a[n]$ is given. Show an algorithm that outputs k elements from $a[]$ randomly. Show the running time of the algorithm also. The algorithm should not output the same element twice or more. That is, when $k = n$, the algorithm shuffles the array $a[]$.)

問題 6: すべての都市を訪問して，元の都市に戻って来る最短経路を探す巡回セールスパーソン問題を考える．入力として，すべての辺に距離がついたグラフ $G = (V, E)$ が与えられたとする．それぞれの辺 $e \in E$ の距離は $w(e)$ と表記する．巡回セールスパーソン問題は NP 完全で非常に難しいが，G の最小全域木 T を求めるのは簡単である．そこで T の辺を (深さ優先探索などの方法で) それぞれ 2 回ずつ通って，すべての都市を訪れることにする．このときの経路長は，巡回セールスパーソン問題の最適解の 2 倍以下になることを示せ．(We consider the travelling salesperson problem for finding a shortest route for visiting all cities. For a given weighted graph $G = (V, E)$, $w(e)$ denotes the length of each $e \in E$. It is quite difficult to solve the travelling salesperson problem since it is NP-complete. However, it is quite easy to find the minimum spanning tree T of G. Therefore, we use each edge of T twice to visit all cities (e.g., in the way of depth first search manner). Then prove that the total length of the route is at most twice of the optimal solution of the salesperson problem.)