3621511: Design and Analysis of Algorithms

Autumn 2019 course final exam

2019-11-01

This exam should have 4 questions printed on 2 pages.
If it does not, then contact the exam supervisor immediately!

Please write this on top of your every answer sheet:

- The name and date of this exam, as given on top of this page.
- Your own name.
- Either your UEF student ID number or your date of birth.
- A running count of your answer sheets in the form "sheet 1 of 2", "sheet 2 of 2".

The final lecture notes and the hints for exercise 7.5 proved this decision problem to be **NP**-complete:

Problem 1 (3-Dimensional Matching, 3DM).

Input: *Three finite sets X, Y and Z having the same size $|X| = |Y| = |Z| = q$. A set $M \subseteq X \times Y \times Z$ of triples.*

Output: *Does M contain a* full matching *$M' \subseteq M$ or not?*

Subset $M' \subseteq M$ is a matching *if all its two distict triples $\langle x_1, y_1, z_1\rangle \neq \langle x_2, y_2, z_2\rangle$ have $x_1 \neq x_2$, $y_1 \neq y_2$ and $z_1 \neq z_2$.*

Matching $M' \subseteq M'$ is full *if $|M'| = q$.*

In practice, we might not have entirely full matchings. Then it becomes interesting to see how close to full they can be. This word "close" can in turn be understood in (at least!) three ways:

(i) "A matching M'' is closer to full than matching M' is, if M'' contains all the same tuples of M as M' does, and more."

(ii) "A matching M'' is closer to full than matching M' is, if M'' contains at least as many elements of X, Y and Z as M' does, and more."

(iii) "A matching M'' is closer to full than matching M' is, if M'' contains at least all the same elements of X, Y and Z as M' does, and more."

The first way (i) raises this optimisation problem:

Problem 2 (Maximal 3DM on Set *inclusion*).

Input: *Three finite sets X, Y and Z. A set $M \subseteq X \times Y \times Z$ of triples.*

Output: *A matching $M' \subseteq M$ such that M does not have another matching $M'' \subseteq M$ such that $M' \subsetneq M''$.*

The second way (ii) raises this optimisation problem:

Problem 3 (Maximal 3DM on Set *size*).

Input: *Three finite sets X, Y and Z. A set $M \subseteq X \times Y \times Z$ of triples.*

Output: *A matching $M' \subseteq M$ such that M does not have another matching $M'' \subseteq M$ such that $|M'| < |M''|$.*

1. The first optimisation Problem 2 is in fact easy. To se this:
   (a) Give an algorithm for it. Justify briefly why your algorithm is correct using invariants. (5 pts.)
   (b) What kind of an algorithm is your algorithm (1a)? Is it Decrease-and-Conquer, Divide-and-Conquer, Dynamic Programming, or Greedy? Justify briefly your answer. (5 pts.)
   (c) Give a polynomial asymptotic time bound for your algorithm (1a). Explain briefly why this bound holds. (5 pts.)
2. The second optimisation Problem 3 is still hard. To see this:
   (a) Define the corresponding decision problem. (5 pts.)
   (b) Show that this decision problem (2a) is in class **NP**. (5 pts.)
   (c) Show that this decision problem (2a) is **NP**-hard. (5 pts.)
3. In your opinion, does way (iii) seem more like way (i) or way (ii)? That is, would you start looking for an efficient algorithm like in Problem 2 or an **NP**-hardness proof like in Problem 3?

   Since this is a question about your opinion, it will be graded according to how well you argue for it. But be brief. (5 pts.)
4. Explain briefly why each problem $A \in \mathbf{NP}$ has a deterministic algorithm $\mathcal{A}$ which runs in time $O(2^{p(n)})$ and space $O(p(n))$ where $p(n)$ is some polynomial of the input length n. (5 pts.)

THE END.
Maximum score: 40 pts.