CSOR W4246 – Fall 2018

Homework 4—Theoretical part

Out: Wednesday, Nov 14, 2018
Due: 8pm, Thursday, Nov 29, 2018

Please keep your answers clear and concise. For all algorithms **you** *suggest, you must prove correctness and give the best upper bound that you can for the running time.*

You should not use any external resources *to solve this homework (not even your textbook) and* **you should not collaborate** *with your classmates. Failure to follow these instructions will have a negative impact on your performance in the second exam. I encourage you to work on all recommended exercises.*

You should write up solutions **entirely on your own**. *Similarity between your solutions and solutions of your classmates or solutions posted online will result in receiving a 0 in this assignment and possibly further disciplinary actions.*

You must submit your assignment as a pdf file. *Other file formats, such as jpg, doc, c, or py, will not be graded, and will automatically receive a score of 0. If you do not type your solutions, be sure that your hand-writing is legible, your scan is high-quality and your name is clearly written on your homework.*

1. (10 points) You should solve this problem **before** you answer the programming part of hw4.

   In formulating flow problems, we assume that there is at most one directed edge between any two nodes. Suppose we have a network with multi-edges: now there may be be multiple directed edges going from node i to node j, each possibly with different capacities and weights. Show how to reduce this problem to the standard one without multi-edges.

2. (20 points) Consider an instance of the Satisfiability problem where all literals in every clause are unnegated variables. Such an instance is called a *monotone* instance. For example,

   $$(x_1 \vee x_2) \wedge (x_1 \vee x_3) \wedge (x_2 \vee x_4)$$

   is a monotone instance. Monotone instances are trivially satisfiable: set *every* variable to 1 (true). However, the instance above would still be satisfied if, instead of setting $x_1 = x_2 = x_3 = x_4 = 1$, we only set $x_1 = x_2 = 1$. Our goal is to minimize the number of variables set to 1 so that the monotone instance evaluates to 1.

   State the decision version of this problem and prove it is NP-complete.

3. (20 points) A large store has m customers and n products and maintains an $m \times n$ matrix A such that $A_{ij} = 1$ if customer i has purchased product j; otherwise, $A_{ij} = 0$.

   Two customers are called *orthogonal* if they did not purchase any products in common. Your task is to help the store determine a maximum subset of orthogonal customers.

   Give an efficient algorithm for this problem or state the decision version of this problem and prove it is $\mathcal{NP}$-complete.

4. (20 points) You are asked to assist in the following crisis event.

   Due to large scale flooding, there is a set of n injured people distributed across a region that need to be rushed to hospitals. There are k hospitals in the region, and each of the n people needs to be brought to a hospital that is within a half-hour's driving time of their current location (so different people will have different options for hospitals, depending on where they are right now). However you do not want to overload any single hospital; instead, you want every hospital to receive at most $\lceil n/k \rceil$ people.

   Give an efficient algorithm for this problem.

5. (30 points) An advertising company underwent a thorough reorganization during which the m employees of the department of Creative Designers got laid off. However $n > m$ new positions have been created in the company. The human resources manager interviews all m employees regarding their interest in the positions and their qualifications. Then he assigns a score s_{ij} to each employee i for each position j (s)he is willing to accept reflecting how qualified employee i is for position j.

   The goal of the manager is to assign jobs to employees so that the sum of the scores of the employees who are assigned to jobs is maximized. A single job cannot be assigned to more than one employee and a single employee may not be assigned to more than one job.

   i. (4 points) Formulate an integer program (IP) for this problem.

   ii. (4 points) Formulate this problem in graph-theoretic terms and explain what the above IP finds.

   iii. (4 points) Now suppose that $s_{ij} = 1$ for every employee i that qualifies for a job j. Explain in graph-theoretic terms what the resulting IP finds.

   iv. Now consider the specific instance of the above problem where $m = 2$, $n = 3$, $s_{ij} = 1$ for all $1 \leq i \leq 2, 1 \leq j \leq 3$.

      i. (12 points) Write out the IP for this instance. Write the LP for its linear programming relaxation and take the dual of the relaxed LP (show all your work).

      ii. (6 points) Assume that the dual LP has an integral solution. Interpret the dual LP in graph-theoretic terms. That is, what do the dual variables encode? Which problem is the dual LP solving?

RECOMMENDED exercises: do NOT return, they will not be graded.)

1. *(Using reductions to prove $\mathcal{NP}$-completeness)*

   (a) A *clique* in an undirected graph $G = (V, E)$ is a subset S of vertices such that *all* possible edges between the vertices in S appear in E. Computing the maximum clique in a network (or the number of cliques of at least a certain size) is useful in analyzing social networks, where cliques corresponds to groups of people who all know each other.

   State the decision version of the above maximization problem and show that it is $\mathcal{NP}$-complete. *Hint: reduction from* `Independent Set`.

   (b) We say that G is a *subgraph* of H if, by deleting certain vertices and edges of H we obtain a graph that is, up to renaming of the vertices, identical to G.

   The following problem has applications, e.g., in pattern discovery in databases and in analyzing the structure of social networks.

   `Subgraph Isomorphism:` Given two undirected graphs G and H, determine whether G is a subgraph of H and if so, return the corresponding mapping of vertices in G to vertices in H.

   Show that `Subgraph Isomorphism` is $\mathcal{NP}$-complete.

   (c) Similarly, consider the following problem.

   `Dense Subgraph:` Given a graph G and two integers a and b, find a set of a vertices of G such that there are at least b edges between them.

   Show that `Dense Subgraph` is $\mathcal{NP}$-complete.

2. Suppose you are given n cities and a set of non-negative distances d_{ij} between pairs of cities.

   (a) Give an $O(n^2 2^n)$ dynamic programming algorithm to solve this instance of **TSP**; that is, compute the cost of the optimal tour and output the actual optimal tour.

   (b) What are the space requirements of your algorithm?

   Hint: Let $V = \{1, \ldots, n\}$ be the set of cities. Consider progressively larger subsets of cities; for every subset S of cities including city 1 and at least one other city, compute the shortest path that starts at city 1, visits all cities in S and ends up in city j, for every $j \in S$.