CSE421: Introduction to Algorithms March 8, 2022

Homework 8

Yin Tat Lee Due: Mar 9, 2022 (before class)

Unless otherwise mentioned, you always need to show your algorithm's runtime and prove that it outputs the correct answer. See Homework Guideline on Ed for more details.

1. (10 Marks) Given two decision problems X and Y, let $X \wedge Y$ denote the decision problem where the answer is yes if and only if the answer to both X and Y is yes. Similarly define $X \vee Y$. If X, Y are both in NP, does this imply that $X \wedge Y$ is in NP? Does it imply that $X \vee Y$ is in NP? If your answer is yes, give a short proof. If your answer is no, give a counterexample.

2. (10 Marks) Prove that the HW5 Problem 3 is NP-complete, i.e. the decision problem of determining whether there is a subset with total weight fulfilling the target values is NP-complete. (This justifies the dependency on α, β in the time complexity in the HW5 Problem 3.)

   Hint: You can use the fact that the Subset-Sum problem is NP-complete. In Subset-Sum, we are given positive numbers $w_1, ..., w_n$, and we want to know if there is a subset that adds up to exactly W.

3. (10 Marks) Consider the following decision problem Multiple-Shortest-Paths (MSP):

   Input: An unweighted, directed graph $G = (V, E)$, integers K and L, and K pairs of vertices $(s_1, t_1), \ldots, (s_K, t_K)$.
   Output: "Yes" if and only if there exists paths $P_1, \ldots, P_K$, where P_i goes from s_i to t_i, such that no two paths share a common vertex, and the total number of edges in these paths is at most L.

   Prove that MSP is NP-complete.

   Hint: Let the input to a 3-SAT instance be denoted $C_1 \wedge C_2 \cdots \wedge C_n$, where each clause is $C_i = a_{i1} \vee a_{i2} \vee a_{i3}$. Reduce this instance of 3-SAT to MSP.

   For each clause $C_i = a_{i1} \vee a_{i2} \vee a_{i3}$, create 5 vertices $s_i, v_{i1}, v_{i2}, v_{i3}, t_i$. Add edges appropriately. For each boolean variable x_k, create vertices s_k and t_k and add edges appropriately.

4. (**Extra Credit**) Read about the link-cut tree data structure. Explain what it does and why such data structure is possible in less than 6 sentence.