CSC364 Summer 2002 – Homework 3

The following questions are assigned each week. You need only hand in those questions marked with a star * as part of your assignment, and only these questions will be marked. Make sure you include a completed and signed cover page as your first page.

Week 9

*1. (hand in) Define the ALL-PAIRS-SHORTEST-PATHS-LENGTH problem (APSP):
Given: a directed graph G and edge weight function $w : E \to \mathbb{R}^{\geq 0}$,
Find: the minimum weight of a path between u and v, for every pair of vertices u and v.

Define the ALL-PAIRS-LONGEST-PATHS-LENGTH problem (APLP):
Given: a directed graph G and edge weight function $w : E \to \mathbb{R}^{\geq 0}$,
Find: the maximum weight of a *simple* path between u and v, for every pair of vertices u and v.

Consider Floyd's algorithm for solving APSP. Can you slightly modify Floyd's algorithm to solve APLP? How can you do this modification or why is it not possible?

2. Define the optimization problem LONGEST-PATH-LENGTH (LPL):
Given: a directed graph G and two vertices u, v
Find: the maximum length of a simple path between u and v in G.

Define the decision problem LONGEST-PATH (LP):
Given: a directed graph G, two vertices u, v, and an integer k,
Find: does there exist a simple path from u to v in G of at least k edges?

Show that the optimization problem LPL can be solved in polynomial time if and only if the decision problem LP can be solved in polynomial time (i.e., iff LP $\in \mathbf{P}$).

Week 10

3. Let 2-CNF-SAT be the set of satisfiable boolean formulas in CNF with exactly 2 literals per clause. Show that 2-CNF-SAT $\in$ **P**. Make your algorithm as efficient as possible.
   Hint: Observe that $x \vee y$ is equivalent to $\neg x \rightarrow y$. Reduce 2-CNF-SAT to a problem on a directed graph that is efficiently solvable.

*4. (hand in) Linear programming when we restrict the variables to be zero or one is called **0-1 integer programming**. Specifically, the 0-1 integer programming problem asks, when given an integer $m \times n$ matrix A and an integer m-vector b, whether there exists an n-vector whose elements are selected from $\{0, 1\}$ such that $Ax \leq b$.
   Prove that 0-1 integer programming is **NP**-complete.
   Hint: Reduce from 3-CNF-SAT.

5. Complete the proof for Theorem 7 of the online notes (page 11-12), that PARTITION is **NP**-complete.

Week 11

(note that there are two hand in problems this week!)

*6. (hand in) Define the SUBGRAPH-ISOMORPHISM decision problem as follows:
   Given: graphs $G = (V_1, E_1)$ and $H = (V_2, E_2)$,
   Decide: does G contain a subgraph isomorphic to H?
   (i.e., are there subsets $V \subseteq V_1$ and $E \subseteq E_1$ such that $|V| = |V_2|$, $|E| = |E_2|$ and there exists a one-to-one mapping $f : V_2 \rightarrow V$ satisfying $\{u, v\} \in E_2$ iff $\{f(u), f(v)\} \in E$?)

   Prove that SUBGRAPH-ISOMORPHISM is **NP**-complete.
   Hint: Consider one of the problems CLIQUE, HC or HP.

7. In a graph G, a Hamiltonian path is a simple path that includes every vertex of G. A Hamiltonian cycle is a simple cycle that includes every vertex of G.

   Define the decision problems as:
   HP: given a graph G, does G contain a Hamiltonian path?
   HC: given a graph G, does G contain a Hamiltonian cycle?

   Prove both HC $\leq_p$ HP and HP $\leq_p$ HC, showing that HC $\equiv_p$ HP.
   Hint: It is not as easy as it looks!

*8. (hand in) Ontario has recently deregulated its energy market. This new "free" market works by matching producers of energy (power plants) with consumers of energy (a public utility, a factory). Each producer makes a "sell" order indicating price the producer will sell energy for (in cents per kilowatt) and the number of kilowatts the producer will sell. Each consumer makes a "buy" order indicating the price the consumer is willing to purchase energy for and the amount the consumer wishes to purchase. If the price of a buy order is higher than or equal to the price on a sell order, a sale takes place and the consumer then sells x kilowatts to the consumer where x is the minimum of the amount the consumer will buy or the producer will sell.

The Ontario Energy Board approaches you with the following problem. Consumers and producers want the ability to create a "secret" order. A secret order indicates that the producer (or consumer) is willing to sell (buy) all of its energy at a lower (higher) price, but only if they are guaranteed to sell (buy) a minimum amount. Each secret order has a price and a minimum amount. For example, suppose Ontario Hydro currently has a sell order on the market stating that it is willing to sell up to 2,000,000 kilowatts at 4 cents per kilowatt. However, Ontario Hydro is also willing to sell at 3 cents a kilowatt if it can sell at least 1,000,000 kilowatts.

The problem Ontario Energy Board has is that it wants the program to automatically match secret orders. That is, if there are enough sellers of energy who are selling at or below the secret buy price of an order (including any secret sell prices) and the total amount is at least the minimum set by the secret buy, then we have a match. However, if the match includes some secret sell orders, then we may also need to include some other orders on the buy side so that the secret sell gets its minimum.

Consider the following example.

Seller	Price	Amount	Secret Price	Secret Min
Bill's Wind Farm	3¢	250,000	—	—
Ontario Hydro	4¢	2,000,000	3¢	1,750,000

Buyer	Price	Amount	Secret Price	Secret Min
Ford Motors	2.5¢	1,500,000	3¢	1,000,000
Toronto Hydro	2¢	1,500,000	3¢	1,200,000

If we did not have secret orders, no sales would occur in this example because the producers are asking for a higher price than the consumers want to spend. However, if we look at the secret price, then we have a sale.

Ontario Hydro will sell at 3 cents if it can sell at least 1,750,000 kW. This minimum can be met if we sell to both Ford Motors and Toronto Hydro. However, Ford Motors and Toronto Hydro will only buy at 3 cents if they can get, combined, 2,200,000. Ontario Hydro is only selling 2,000,000, but if we include Bill's Wind Farm, then we have a total amount of 2,250,000 kW which satisfies all the minimum quantities. Thus, Bill's Wind Farm and Ontario Hydro will sell a total of 2,250,000 at 3 cents to Ford Motors and Toronto Hydro.

Here are the specifications of the algorithm that Ontario wants you to design. The input to the algorithm will be a list of buy and sell orders, each with at most one secret price. The output is a series of sales.

You are to either design an algorithm or prove that the decision version of the problem is **NP**-complete. If an algorithm is possible, will it be feasible for the Ontario market? The market must be real-time so any algorithm which is worse than $O(n^2)$ will not be feasible.