Problem Set 9 (Optional)

This problem set is not due; it is optional.

Reading: Chapter 34

Problem 9-1. Prove that the following problems are in NP:

(a) $D_1 = \{\langle G, u, v, k\rangle |$ there exists a simple path in G between u and v, of length at least $k\}$

(b) $D_2 = \{G|$ it is possible to assign one of three "colors" to each vertex of G such that no two neighboring vertices are assigned the same color $\}$

Problem 9-2. A ***subgraph*** of a graph $G = (V, E)$ is a graph $G' = (V', E \cap (V' \times V'))$ where $V' \subseteq V$; i.e. it is a subset of the vertices together with all the edges of the original graph which are incident to these vertices.

Consider the problem LARGESTCOMMONSUBGRAPH: Given two graphs G_1 and G_2 and an integer k, determine whether there is a graph G with $\geq k$ edges which is a subgraph of both G_1 and G_2. (*Hint:* Reduce from CLIQUE.)

Problem 9-3. A ***perfect matching*** in an undirected graph $G = (V, E)$ is a collection of edges E' such that each node has exactly one edge of E' incident to it. (In other words, the degree of each node in $G' = (V, E')$ is exactly one.) Given a weight $w(i, j)$ on each edge (i, j), there is an $O(n^3)$ algorithm to find a perfect matching in G of minimum total weight; you may use such an algorithm as a subroutine in your solution to the following problem, without worrying about how it works.

Give an algorithm for the traveling salesman problem with triangle inequality that produces a solution within a factor of 3/2 of the optimal one. (*Hint:* The key to the first algorithm given in class is to convert a minimum spanning tree to an Eulerian graph, and then shortcut that to obtain a tour; find a way to do this in a less expensive way by using matching.)

Problem 9-4. Here we will see how a *decision* algorithm for an NP-complete problem can be used to efficiently find a *witness* for that problem (and others).

(a) Prove that if CLIQUE $\in P$, then there is a polynomial-time algorithm that takes a graph G and an integer k and finds a clique in G of size $\leq k$ if one exists, and outputs "NONE" otherwise.

(b) Under the same assumption (CLIQUE $\in P$), prove that there is a polynomial-time algorithm that takes a set of n cities and the distances between them and finds a minimum length traveling salesman tour through all the cities.

Problem 9-5. Consider the following definition of a ***randomized reduction***:

$A \leq_R B$ if there exists a polynomial time function f and a constant k such that

- $x \in A \Rightarrow \forall\, y \in \{0,1\}^{|x|^k},\ f(x,y) \in B$
- $x \notin A \Rightarrow Pr[f(x,y) \in B] \leq 1/2$ (probability taken over choice of $y \in \{0,1\}^{|x|^k}$)

(One way to interpret this definition is that y represents some random choices used in the reduction. That is, rather than letting f make random choices on its own, we give it a random string y to use for this purpose.)

Now, suppose $A \leq_R B$, where $B \in P$ (i.e., B has a poly-time algorithm). Describe a randomized (polynomial time) algorithm for A that is correct with probability at least $1 - 1/2^{100}$.