CS261 Problem Set One

Due Date: Tuesday, February 5, 2008

1. **(15pt)** Design an algorithm that constructs an Eulerian walk on a connected even-degree graph in $O(m)$ time, where m is the number of edges.

2. **(10pt)** Given an algorithm (black-box) that computes a maximum cardinality matching, explain how to obtain a minimum edge cover.

3. **(10pt)** Prove that the size of a *maximal* matching is within a factor of 2 of the size of a *maximum* matching in a graph.

4. **(15pt)** Consider the following variant of TSP: Given an undirected weighted graph $G = (V, E)$, find a cycle of minimum total cost that visits all the nodes. You are allowed to re-visit nodes. Graph G is not necessarily complete. Can you modify Christophedes algorithm to apply to this variant ? If yes, describe how and prove correctness and approximation factor.

5. **(10pt)** Expressing problems as Integer Programs.

   (a) **(5pt)** Express the Maximum Cardinality Matching problem as an IP.

   (b) **(5pt)** Express the Maximum Independent Set problem as an IP.

6. **(10pt)** The general Maximum Clique (MC) problem is to find a maximum set of nodes in an undirected graph that form a clique, i.e. there is an edge connecting each two nodes in the set. The decision version of this problem is "given k, decide whether there is a clique of size k in the input graph". Consider the Maximum Clique problem restricted to graphs where every vertex has degree at most 3, call it MC3.

   (a) **(2pt)** Is MC3 in NP ? (Explain)

   (b) **(3pt)** What is wrong with the following proof of NP-completeness for MC3? We know that the MC problem in general graphs is NP-complete, so it is enough to present a reduction from MC3 to MC. Given a graph G of degree 3 and a parameter k, the reduction leaves the graph and the parameter unchanged: clearly the output of the reduction is a possible input for the MC problem. Furthermore, the answer is YES for the MC problem if and and only if it was YES for the MC3 problem. This proves the correctness of the reduction and the NP-completeness.

   (c) **(5pt)** Give a polynomial running time algorithm for MC3.