CME 305: Discrete Mathematics and Algorithms

Instructor: Professor Amin Saberi (saberi@stanford.edu)

Midterm – 02/17/11

This exam is closed notes/books/laptops. You will have until the end of class to ask any questions clarifying any of the problems. You may then set aside 3 more hours to work on the exam individually. The exam is due at 8pm today. Submit your work into the CME305 dropbox in the basement of Huang located across from room 041 (next to the HP Garage).

Problem 1. An *edge cover* C of a graph $G(V, E)$ is a subset of E such that for all $v \in V$ there exists $e \in C$ with $v \in e$ i.e. an edge set covering all vertices. Let C^* be a minimum edge cover, that is $|C^*| \leq |C|$ for all edge covers C of G. Prove that

$$|C^*| + |M^*| \leq |V|$$

where M^* is a maximal matching of G.

Problem 2: Given a graph $G(V, E)$ we want to partition the vertices of the graph into two (disjoint) sets A and B (i.e. $A \cup B = V$ and $A \cap B = \varnothing$) such that the number of edges with one endpoint in A and the other in B is maximized. In other words, we are looking for the maximum cut (A, B) of the graph.

Consider the following algorithm. We start with an arbitrary partition (A, B). If there exists a vertex $v \in V$ such that moving it to the other set in the partition increases the number of edges in the cut, do so. Proceed until there is no such vertex.

(a) Show that the algorithm stops.

(b) Show that when the algorithm stops, the size of the cut is at least half of the optimum.

Hint: Consider the neighbors of each vertex after the algorithm stops.

Problem 3: Consider the following problem: Given n items with sizes $a_1, a_2, \cdots a_n$ all in $(0, 1]$, find a packing in unit size bins that minimizes the number of bins used.

(a) Prove that the following algorithm is a factor 2 approximation: Consider the items in an arbitrary order. In the i^{th} step, suppose you have a list of partially packed bins, say $B_1, B_2, \ldots, B_k$. If possible, put a_i into any one of them. If a_i does not fit into any of these bins, open a new bin B_{k+1} and put a_i in it.

(b) Give an example on which the above algorithm does at least as bad as 5/3 of OPT, where OPT is the number of bins in the optimal packing.

(c) Consider a modification of the algorithm in part (a). At the i^{th} step, suppose you have a list of partially packed bins, say $B_1, B_2, \ldots, B_k$. You may only put a_i into bin B_k. If a_i does not fit into bin B_k, open a new bin B_{k+1} and put a_i in it. Prove that this modified algorithm also gives a factor 2 approximation.

Problem 4: A *vertex coloring* of a graph $G(V, E)$ is an assignment colors to each vertex of a graph such that no edge connects two identically colored vertices. A graph G is called k-vertex-colorable if and only if there exists a vertex coloring of G with k (or less) colors.

Give a polynomial time algorithm for coloring the vertices of a 3-vertex-colorable graph of size n with at most $O(\sqrt{n})$ colors.

Hint: Try coloring the neighborhood of some vertex.