Fall 2007 **CMSC 858K: Homework 1** Samir Khuller

Due in class: September 6.

Warning: some of the problems require thought – do not wait until the last day to start working on them!

If you cannot come up with algorithms that run in the required time, then provide (correct) slower algorithms for partial credit. Write your answers using *pseudo-code* in the same style as the textbook. These make the algorithm description precise, and easy to read (as opposed to code in C or some other language).

Please also provide a proof of correctness.

(1) Given n unit length jobs, with release times and deadlines (all data integral), how can we maximize the number of jobs that are scheduled after they are released, but before their deadline?

(2) Given n *unit* length jobs each with an associated profit p_i, and with deadline d_i (deadlines are integral), how would we maximize the total profit? We only get the profit for a job if we schedule a job by its deadline. Design a simple greedy algorithm.

(3) A bipartite graph is one whose vertex set can be partitioned into two sets A and B, such that each edge in the graph goes between a vertex in A and a vertex in B. (No edges between nodes in the same set are allowed.)

Give an $O(|E| + |V|)$ algorithm that takes an input graph (represented in adjacency list form) and decides if the graph is bipartite. If the graph is bipartite, the algorithm should also produce the bipartition.

(4) At a wine tasting there are two kinds of wines, *merlot* and *pinot noir*. The glasses are not labeled. Some tasters are present, and each taster sips two glasses of wine and says "these two are the same" or says "these two are different". Given this information, how can we tell if this information is consistent?

(5) (This problem may be discussed with others.) An *Euler tour* of a connected graph is a cycle that traverses each edge of G exactly once, although it may visit a vertex more than once. Show that G has an Euler tour if and only if each vertex has even degree. Design a fast algorithm to output the Euler tour. (For full credit, the algorithm should run in linear time.)