ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

School of Computer and Communication Sciences

Homework 2 **Graph Theory Applications**

Date: May 14, 2013 Spring 2013

This is the second graded homework. The submission date is **24th May**. As always you can talk with any of your colleagues about the problems. But each of you has to write down the solution on her own. In addition, if you have discussed any problems with colleagues or if you have used any other resources (such as Google!), write this down on top of your answer sheet. This is standard scholarly practice. If we discover similarities of solutions beyond what was indicated, you will receive 0 points for this homework. We will not investigate who copied from whom. Good luck!

Problem 1. Consider a simple regular undirected graph with a special vertex v such that if it is deleted along with the edges incident on it, the graph becomes disconnected. Prove that for such a graph $\chi'(G) = \Delta + 1$.

Problem 2. Show how to use an algorithm for computing max-flow in a graph to compute the maximum matching in a bipartite graph. Further, using the max-flow min-cut theorem, deduce that the size of the maximum matching is equal to the size of the minimum vertex cover in a bipartite graph.

Problem 3. Imagine a group of families who all want to go to a picnic by car sharing. Unfortunately, the members of the families often quarrel among themselves! To ensure a good trip, they want to make sure that no two members of the same family travel in the same car. Given a set of n families with $F_1, F_2, \ldots, F_n$ members and a set of m cars with capacity $C_1, C_2, \ldots, C_m$, formulate a network flow problem to check whether it is feasible to arrange for the trip.

Problem 4. Assume there are n modules in a large program and there are two processors on which we want to run these modules. Let $a_1, \ldots, a_n$ and $b_1, \ldots, b_n$ be the cost incurred by each module to run in processor 1 and 2 respectively. Further, if two modules i and j are run in different processors, there is an associated communication cost c_{ij}. There is no communication cost between two modules if both of them run on the same processor. Can you show us a way to assign the modules to the two processors so that the total cost of running them is minimized ?