Algorithm Design - Second Partial Exam
19-12-2023, *Time: 120 minutes*
Prof. Stefano Leonardi, Dr. Federico Fusco

Problem 1

Suppose you are acting as a consultant for the prestigious Caffarella Almone River Authority. They are currently doing a multi-hundred-euro business per year, and their revenue is constrained almost entirely by the rate at which they can unload ships that arrive in the port on the river. Here's a basic sort of problem they face. A ship arrives, with n containers of weight $w_1, w_2, \dots, w_n$. Standing on the dock is a set of trucks, each of which can hold K units of weight. (You can assume that K and each w_i is an integer.) You can stack multiple containers in each truck, subject to the weight restriction of K; the goal is to minimize the number of trucks needed to carry all the containers. This problem is NP-complete (you don't have to prove this). A greedy algorithm you might use for this is the following. Start with an empty truck, and begin piling containers $1, 2, 3, \dots$ into it until you reach a container overflowing the weight limit. Now declare this truck "loaded" and send it off; then continue the process with a fresh truck. By considering trucks one at a time, this algorithm may not achieve the most efficient way to pack the full set of containers into an available collection of trucks.

(a) Give an example of a set of weights, and a value of K, where this algorithm does not use the minimum possible number of trucks.

(b) Show, however, that the number of trucks used by this algorithm is within a factor of 2 of the minimum possible number, for any set of weights and any value of K.

Problem 2

Your friends at LignanoX have recently been doing some consulting work for companies that maintain large, publicly accessible Web sites, and they have come across the following Strategic Advertising Problem. A company comes to them with the map of a Web site, which we will model as a directed graph $G = (V, E)$. The company also provides a set of t trails typically followed by users of the site; we will model these trails as directed paths $P_1, P2, \dots, P_t$ in the graph G (i.e., each P_i is a path in G).

The company wants LignanoX to answer the following question for them: Given G, the paths $\{P_i\}$, and a number k, is it possible to place advertisements on at most k of the nodes in G, so that each path P_i includes at least one node containing an advertisement? We'll call this the Strategic Advertising Problem, with input G, $\{P_i : i = 1, ..., t\}$, and k. Prove that Strategic Advertising is NP-complete.

Problem 3

(a) Give the formal definition of polynomial-time reduction from a decision problem X to a decision problem Y and discuss its implications for the efficient solution of the two problems.

(b) Give the formal definition of efficient certifier for a decision problem X, and the definition of complexity class NP.

(c) Define formally the set cover and the vertex cover problems. Give an efficient certifier for the two problems and show a polynomial time reduction from one of the two problems to the other.

Problem 4 (Bonus)

We are given a set of m machines $M_1, \dots, M_m$ and a set of n jobs. Each job j has processing time t_j and is assigned to one of the machines. Denote by $A(i)$ the set of jobs assigned to machine i, and let $T_i = \sum_{j \in A(i)} t_j$ the load on machine M_i. The *load balancing* problem seeks to minimize the maximum load $T = \max_i T_i$ placed on a machine.

(a) Discuss whether the problem is NP-complete and define the concept of an approximation algorithm for the load balancing problem.

(b) Describe formally the greedy algorithm that first orders the jobs by decreasing processing time, and then assigns the jobs in order to the machine which currently has the minimum load.

(c) Prove that the greedy algorithm above achieves a $3/2$ approximation.