Université de Nice-Sophia Antipolis *UFR Sciences*
1st Year Int. Master in Computer Science 2017–2018

Module: Resolution Methods
Submodule: Combinatorial Optimization

Date: Jan 11th, 2018
Duration: 45min
2 pages manuscript Documents allowed

Grade	**The utilisation of any electronic device is forbidden: computers, cellphones, calculators, books. Only two pages of manuscript notes are allowed.**

Part 2: Graph Algorithm

1. (1 point) Let Π be a maximization problem and let $c > 1$. What is a $\frac{1}{c}$-approximation algorithm for Π?

2. (1 point) Recall the definition of the Load-Balancing problem and explain why it may be interesting to have an approximation algorithm to solve it rather than an exact algorithm.

3. (1 point) Give a 3/2-approximation algorithm to solve the Load-Balancing problem. (only explain it in few sentences, no proofs required.)

4. (2 points) Recall the definition of the Traveling Salesman Problem and give a 2-approximation algorithm to solve it. (only explain it in few sentences, no proofs required.)

5. (5 points) Let $G = (V, E)$ be a connected graph. Recall that the *distance* $dist_G(u, v)$ between two vertices u and v in G is the length (number of edges) of a shortest path between u and v in G. The diameter $Diam(G)$ of G is the maximum distance between two vertices, i.e.,

$$Diam(G) = \max_{u,v \in V} dist_G(u, v).$$

(a) (1 point) What are the diameters of each of the 3 graphs depicted in Figure ??.

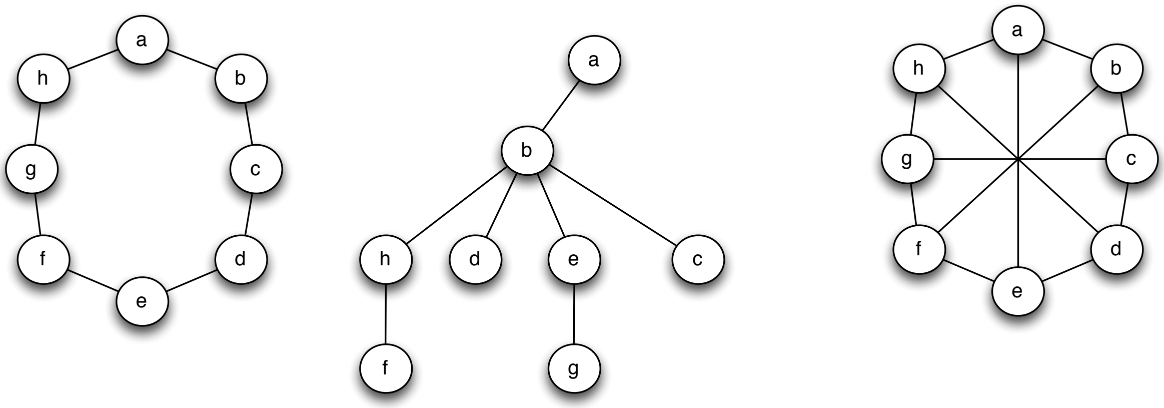

Figure 1: Three connected graphs.

(b) (1 point) Give an algorithm that computes the diameter of any graph. What is its time-complexity?

(You may assume that you are given an algorithm $\mathcal{A}$ that computes the distance between one vertex and all other vertices in time $O(|E|)$.)

(c) (3 points) Consider the following algorithm.

Algorithm: Let v_0 be any vertex of G. Execute one Breadth First Search (BFS) from v_0 and let δ be the maximum obtained label. Return δ.

Recall that the BFS algorithm consists as follows: It starts from v_0 which is labelled with 0 and all other vertices are unlabelled. Then, while there are some unlabelled vertices, every unlabelled vertex adjacent to a vertex labelled i becomes labelled $i + 1$.

(a) (1 point) Apply the algorithm on the tree in the center of Figure **??** with $v_0 = e$.

(b) (1 point) Prove that this is a $\frac{1}{2}$-approximation algorithm for computing the diameter of any graph.

(c) (1 point) Give an example of a graph G where the value returned by the algorithm is half the diameter of G.