UBINET, Master 2 IFI Algorithms for telecommunications

First Exam, October 2021

1 hour 1/2

No documents are allowed. No computers, cellphones.

Instruction and comments: the points awarded for your answer will be based on the correctness of your answer as well as the clarity of the main steps in your reasoning. All proposed solutions must be proved. **All the exercises are independent.** *The points are indicated so you may adapt your effort.*

Exercise 1 (Dijkstra. 5 points, 20 minutes) Consider the graph H depicted in Figure 1.

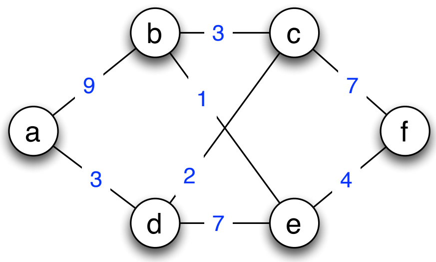

Figure 1: The weighted graph H. Integers on edges represent their length.

1. Give the definition of a shortest-path tree rooted in a.
2. Apply the Dijkstra algorithm on H to compute a shortest-path tree rooted in a and the distance between any vertex and the vertex a.

   You must explain the execution of the algorithm (you may write the table as seen during the lecture). In particular, indicate the order in which vertices are considered during the execution of the algorithm.
3. Give the obtained shortest-path tree rooted in a.

Exercise 2 (Flow. 7 points, 25 minutes) Consider the elementary network flow N depicted in Figure 2 (left) and the initial flow f from s to t in Figure 2 (right).

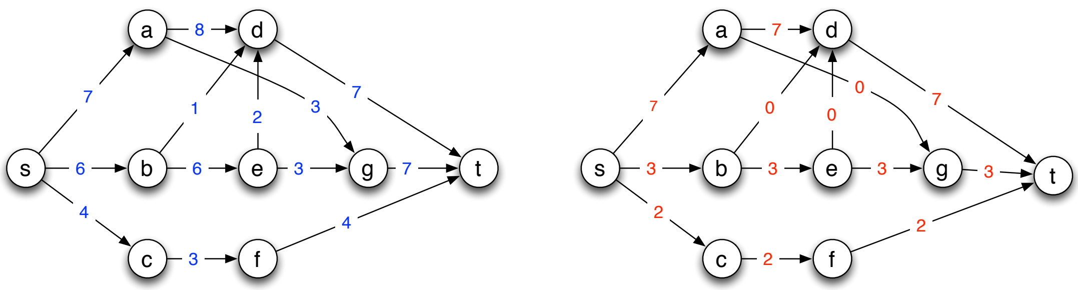

Figure 2: (left) Elementary network flow with arcs' capacity. (right) An initial flow f from s to t: a number close to an arc indicates the amount of flow along it.

1. What must be checked to show that f is a flow? What is the value of the initial flow f?
2. Apply the Ford-Fulkerson Algorithm to N starting from the flow f. All steps of the execution of the algorithm must be detailed.

   For each iteration, draw the auxiliary graph, give the chosen path and the amount of flow that you will push, and draw the network with the new flow.

3. What is the final value of the flow?

Exercise 3 (Graph. 8 points, 45 minutes)

1. What means that a graph is connected?
2. Give the definition of a tree.
3. Prove that any tree T has a vertex of degree 1.

   hint: consider a longest path in T.
4. Prove that any tree with n nodes has $n-1$ edges.

   hint: by induction on n.
5. Give the definition of a spanning tree of a connected graph.
6. Show that any connected graph with n vertices has at least $n-1$ edges.
7. Prove that any graph with n vertices that is connected and has exactly $n-1$ edges is a tree.
8. Prove that any graph with n vertices that is acyclic and has exactly $n-1$ edges is a tree.
9. Give the definitions of a matching M in a graph $G=(V,E)$ and of an M-augmenting path.
10. Give an algorithm that computes a maximum matching in any tree. Give its time-complexity.

    hint: give a polynomial time algorithm to compute an augmenting path in a tree.