

Homework 1 - Algorithm Design

Sapienza University of Rome

Marco Costa, 1691388

November 20, 2019

Exercise 1

Exercise 2

Problem: Given a weighted tree T with n nodes, find the complete graph G of minimum weight such that $T \subseteq G$ and T is the unique minimum spanning tree of G .

Idea: Insert edges that are not in the tree so as to obtain the complete graph. These edges must have a greater weight than those of the tree, so that T is the only MST of G .

Hint: Since *Kruskal's* algorithm uses the *Union-Find* structures for representing the cost, we use this structures to solve the problem.

First solution:

Algorithm 1 Find complete graph

```

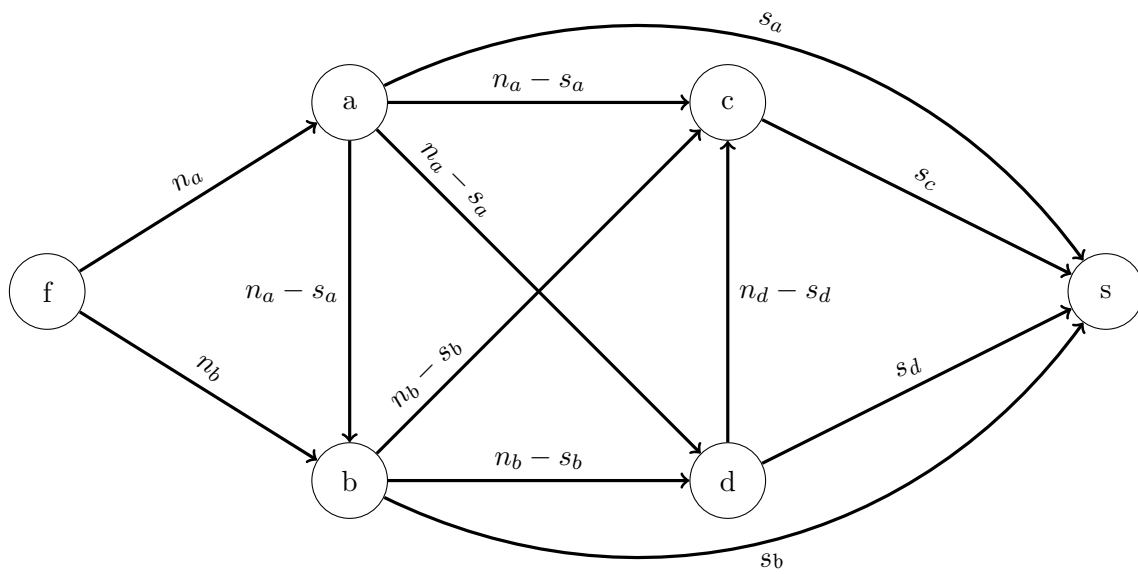
1: for  $u$  in  $set_1$  do
2:   for  $v$  in  $set_2$  do
3:     if  $e_{new} = (u, v)$  not in  $E$  then
4:        $e_{new}.setWeight(e.getWeight() + 1)$ 
5:        $G.addEdge(e_{new})$ 

```

Cost: $O(|E|)$, but $|E| = \frac{|V|(|V|-1)}{2}$ and $|V| = n$, so $|E| = \frac{n^2-n}{2}$ and the cost is $O(n^2)$

Second solution:

Exercise 3



Exercise 4