CONCORDIA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

COMP 6651: Algorithm Design Techniques

Fall 2019

Quiz # 4

First Name	Last Name	ID#

Question 1

Recall what is the problem solved by the algorithm of Dijkstra: input (including assumptions) and output

Recall the algorithm of Dijkstra

Problem solved by the algorithm of Dijkstra **3 points**

Algorithm of Dijkstra solves the single-source shortest paths problem, i.e.,
Graph $G = (V, L)$ with non-negative weights.
From a given source vertex $v_s \in V$, find the shortest-path weights $\textsc{weight}(v_s \rightsquigarrow v)$ for all $v \in V$. If all edge weights $w(u, v)$ are nonnegative, all shortest-path weights must exist.

Algorithm of Dijkstra **7 points**

Dijkstra's algorithm

$d[s] \leftarrow 0$
for each $v \in V - \{s\}$
  do $d[v] \leftarrow \infty$
$S \leftarrow \varnothing$
$Q \leftarrow V$ $\triangleright$ Q is a priority queue maintaining $V - S$
while $Q \neq \varnothing$
  do $u \leftarrow$ EXTRACT-MIN(Q)
    $S \leftarrow S \cup \{u\}$
    for each $v \in Adj[u]$
      do if $d[v] > d[u] + w(u, v)$
        then $d[v] \leftarrow d[u] + w(u, v)$

relaxation step

Implicit DECREASE-KEY

 Introduction to Algorithms Day 29 L17.8

Let $G = (V, E)$ a directed graph. It is said that a flow is maximum if there is no augmenting path in the residual graph of G. However, you discover the following algorithm in a book on Operations Research where they propose the following algorithm:

1. Look for augmenting paths in the original graph where an augmenting path is defined as a path made of links for which the flow value is smaller than the capacity value. As long as you can find an augmenting path in the original graph, iterate on the original graph, searching, at each iteration, for the augmenting path that can carry the largest possible amount of flow.
2. When there exists no more augmenting path on the original graph, built the residual graph, look for an augmenting path on the residual graph. If there exists none, EXIT, otherwise update the flows on the original graph and go back step 1.

a. What is the complexity of step 1. Provide a detailed complexity analysis.

b. Is the algorithm exact? Justify your answer.

a. The complexity of step 1 is the same as the complexity of the Ford Fulkerson algorithm, as in the worst case, one can reach the optimal solution using only step 1: $O(|E|f^\star)$, where $f^\star$ is the value of the maximum flow.

5 points

b. The algorithm is exact. Indeed, the second step corresponds to the Ford Fulkerson algorithm, while the first step is a heuristic.

5 points