Noella James
Collaborators: none

1 Problem 2-1: Pedantic Sandy

1.1 (Problem Description)

The problem that we are trying to solve is a variation of Dijkstra's algorithm. We need to write an algorithm that finds the shortest path to all the vertices such that each of the paths will end on Sally's left foot.

1.2 (Algorithm to Construct G')

Input : Graph $G = (V, E, w : E \to \mathbb{R}^{\geq 0})$ weighted w/non-negative weights.
Output : Graph G' such that the vertices in the graph G' contains each vertex from G and the foot used to step on the vertex.
Graph $G' = (V', E', w' : E' \to \mathbb{R}^{\geq 0})$ where $V' = V \times \{left\} \bigcup V \times \{right\}$, edge set $E' = \{((u, left), (v, right)) | (u, v) \in E\} \bigcup \{((u, right), (v, left)) | (u, v) \in E\}$ and weight function $w'((u, \cdot), (v, \cdot)) = w((u, v))$.

Algorithm 1 Construct G'

procedure CONSTRUCT $G'(G)$ ▷ $G(V, E)$ weighted graph w/non-negative weights
  for all vertices $v \in V$ **do**
    $V' \leftarrow (v, left)$
    $V' \leftarrow (v, right)$
  end for ▷ $n = |V|$ Complexity $= O(n)$
  for all edges $e = (u, v) \in E$ **do**
    $E' \leftarrow ((u, left), (v, right))$
    $E' \leftarrow ((u, right), (v, left))$
  end for ▷ $m = |E|$ Complexity $= O(m)$
  for all weights $p = (u, v) \in w$ **do**
    $w' \leftarrow ((u, left), (v, right))$
    $w' \leftarrow ((u, right), (v, left))$
  end for ▷ $m = |E|$ Complexity $= O(m)$
  return $G' = (V', E', w' : E' \to \mathbb{R}^{\geq 0})$.
end procedure ▷ Total Complexity $= O(n + 2m)$

1.3 Find shortest paths to left vertices

Input : Graph $G = (V, E, w : E \to \mathbb{R}^{\geq 0})$ weighted w/non-negative weights.

Output : A set S that contains all the shortest paths to each vertex where the foot used to step on the last vertex is the left foot.

Algorithm 2 Find shortest left foot path

procedure FIND SHORTEST LEFT FOOT PATH(G, s) $\triangleright$ $G(V, E)$ weighted graph w/non-negative weights, and s is a vertex in G
  $G' \leftarrow$ Construct $G'(G)$ $\triangleright$ $n = |V|, m = |E|$ Complexity $= O(n + 2m)$
  $S' \leftarrow$ Dijkstra's $(G', (s, left))$ $\triangleright$ $n = |V|, m = |E|$ Complexity $= O((n + 2m) \log n)$
  $S \leftarrow \{\}$
  for all $(v, left) \in S'$ **do**
    $S \leftarrow S \bigcup \{v\}$
  end for $\triangleright$ $n = |V|$ Complexity $= O(n)$
  return S. $\triangleright$ If a given vertex u is not in S, then no path exists
end procedure $\triangleright$ $n = |V|$ Complexity $= O((n + 2m)(\log n + 1))$

1.4 Complexity

The construction of G' has a complexity of $O(n + 2m)$. Dijkstra's Algorithm has a complexity of $O((n + 2m) \log n)$. The complexity of going through all the vertices in S' that use the left foot and end with the left foot is $O(n)$. Therefore, the total complexity is $O((n + 2m)(\log n + 1))$.

1.5 Correctness Proof

Note 1.1. The correctness proof depends on the correctness of Dijkstra's Algorithm.

Lemma 1.1. *The algorithm above always provides the shortest path, if one exists, such that the last vertex will be the left foot.*

Proof. By Induction
Base Case: $G = \{s\}$, no additional path to be written since Sally can't go anywhere.
Induction Hypotheses: Lemma is *true* for certain s.
Induction Step: We'll prove that the lemma is still *true* after we add a new vertex v to G.
For this vertex v, we will be adding $(v, left)$ and $v, right)$ to V'. We also will be adding every edge e from and to v to E'. Similarly, we will be adding every weight p to w'. Thus, it should be possible to run Dijkstra's Algorithm on Graph G' and compute the shortest paths to $(v, left)$ and thus to v. $\square$