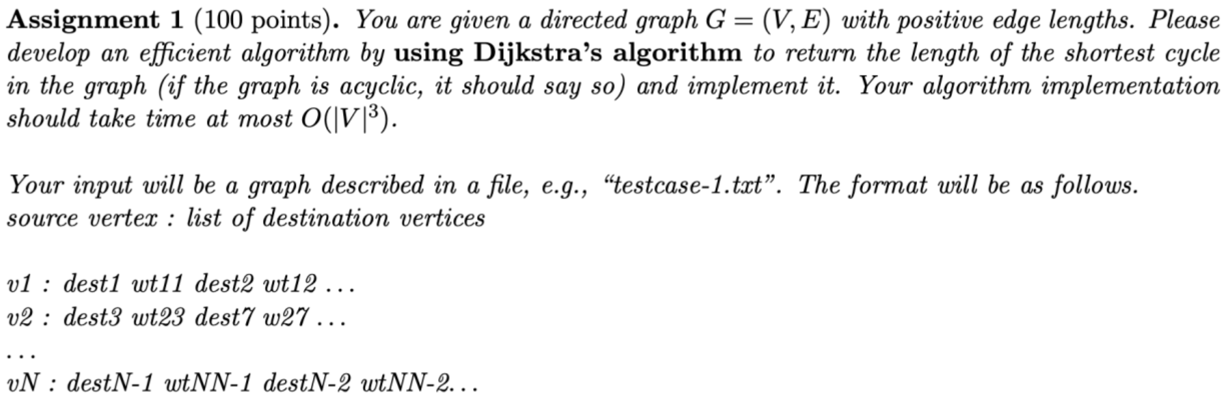

Assignment 1 (100 points). *You are given a directed graph $G = (V, E)$ with positive edge lengths. Please develop an efficient algorithm by* **using Dijkstra's algorithm** *to return the length of the shortest cycle in the graph (if the graph is acyclic, it should say so) and implement it. Your algorithm implementation should take time at most $O(|V|^3)$.*

Your input will be a graph described in a file, e.g., "testcase-1.txt". The format will be as follows.
source vertex : list of destination vertices

v1 : dest1 wt11 dest2 wt12 . . .
v2 : dest3 wt23 dest7 w27 . . .
. . .
vN : destN-1 wtNN-1 destN-2 wtNN-2. . .

1.https://github.com/OnetoTommy/Program-Structure-Algorithms-/tree/main/Course/Course_10/PA_1

2. Describe Algorithm (Dijkstra's Algorithm)

1. **Graph Representation:**
   The graph is stored as an adjacency list using a dictionary where keys are nodes, and values are lists of tuples representing neighbors and edge weights.
2. **Function dijkstra_shortest_cycle(start):**
   - This function uses **Dijkstra's Algorithm** to find the shortest cycle starting and ending at start.
   - It initializes a shortest_cycle dictionary where all nodes have infinite distance except the start node (set to 0).
   - A **priority queue (min-heap)** is used to always process the node with the smallest distance.
   - For each node, its neighbors are checked, and distances are updated if a shorter path is found.
   - **If a neighbor is the start node and it's not a direct back edge**, a cycle is detected, and its length is returned.
3. **Function finding_shortest_cycle():**
   - This function calls dijkstra_shortest_cycle(start) for every node in the graph.
   - It keeps track of the shortest cycle found.
   - If no cycle exists, it returns 0.
4. **Reading Input Graph from File:**
   - The input file contains graph edges and each row represents a source vertex followed by pairs of (neighbor, weight).
   - This is parsed into the adjacency list representation.
5. **Final Output:**
   The algorithm prints the length of the shortest cycle found like "**The length of the shortest cycle is: 8**"

3. Testcase(s) on which you validated your program.

Case-1: Acyclic

0 : 1 1

1 : 2 2

2 : 3 3

Case-2: Single Cycle

0 : 1 1 2 4

1 : 3 2

2 : 3 1

3 : 0 3

Case-3: Double Cycle

0 : 1 1

1 : 2 2

2 : 3 3

3 : 1 4 0 5

Case-4: Empty

Link: https://github.com/OnetoTommy/Program-Structure-Algorithms-/tree/main/Course/Course_10/PA_1