Instructor Name: Siddhartha Nath
Email: s.nath@northeastern.edu

Northeastern University

Program Structures and Algorithms (INFO 6205)
Programming Assignment – 100 points

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. . .

Your program should be able to accept any input file using a command-line argument as follows.
your_program.py –input *input_graph_file*

Your program should output, **"The length of the shortest cycle is: $val"**, *where val is the length. It will be* 0 *when there are no cycles.*

For example, if my input file is as follows.
0: 1 1 2 4
1: 3 2
2: 3 1
3: 0 3

The output is, "The length of the shortest cycle is: 6", corresponding to the cycle $0 \xrightarrow{1} 1 \xrightarrow{2} 3 \xrightarrow{3} 0$.

For example, if my input file is as follows.
0: 1 1
1: 2 2
2: 3 3
3: 1 4 0 5

The output is, "The length of the shortest cycle is: 9", corresponding to the cycle $1 \xrightarrow{2} 2 \xrightarrow{3} 3 \xrightarrow{4} 1$.

For example, if my input file is as follows.
0: 1 1
1: 2 2

2: 3 3

The output is, "The length of the shortest cycle is: 0", since the graph is acyclic.

You must submit the following to receive full credit.

(1) *(20 points) Python or Java script or C++ file with comments and structure.*

(2) *(20 points) Describe your algorithm in English as comments. It* **MUST** *use Dijkstra's algorithm.*

(3) *(10 points) Testcase(s) on which you validated your program. Each testcase must be a testcase-i.txt file where $i = 1, 2, ..., n$ are indices to your testcase files. You MUST include at least one testcase file.*

(4) *(45 points) Valid output on a variety of inputs.*

(5) *(5 points) Please make sure you handle corner cases, and gracefully error out when you are provided with incorrect inputs, etc. Program crashes or errors or no outputs will be penalized.*