Introduction to Algorithms Programming Assignment 2

This programming assignment is due by midnight of 6/7 (Friday). Please submit the Python code with proper comments via eeclass. 20% penalty will be applied if it is submitted on 6/8. No submission will be accepted after 6/8.

This programming assignment asks you to run the appropriate single-source shortest paths algorithm for a given graph.

We have covered Dijkstra's algorithm and the Bellman-Ford algorithm, each of them suitable for a different type of graphs. In the textbook, you also learn the algorithm for directed acyclic graphs. Now do the following for this programming assignment:

1. Read from a file called input.txt, which will specify the information of a graph. (The format of the input is described below.)
2. Determine what type of graph is specified in the input file. It could be:
   A. A graph with negative weight cycles
   B. A graph with negative weight edges but no negative weight cycles
   C. A graph with no negative weight edges
   D. A directed acyclic graph
3. Afterward, apply the best (i.e., the one with the lowest time complexity) single-source shortest paths algorithm to identify all shortest paths from source vertex 0. Note that if a negative weight cycle is reachable from vertex 0, the program should stop with a warning message such as "No shortest paths can be found."

Given a sample graph in Figure 1, its input file is given below. The first line indicates the number of vertices and the number of edges. Each of the remaining lines represents an edge, where the first two numbers indicate the two ends of the edge (and the order indicates the direction) and the last number is the weight of the edge.

```
===
4	5
0	1	10
0	2	3
0	3	2
1	3	7
2	3	6
```

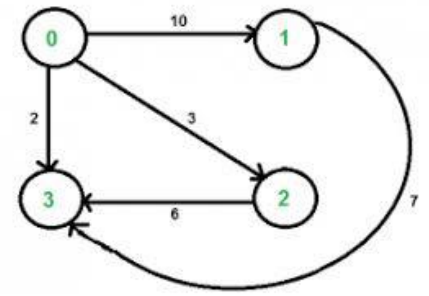

Figure 1. A sample graph