Data Structure and Advanced Programming

Homework #13

Due: 2021/6/15 08:00am (CST)

NOTE: Please upload your C++ source codes (by copy-paste) to PDOGS before the due date and time.

The goal of this homework is to build a weighted directed graph and find the shortest path between two vertices. Please implement the graph in C++ to fulfill the required input and output. Please use an adjacency matrix to represent the graph.

The first line of input is the number of vertices and the vertex id start from zero. Each following line of input contains three space-separated integers and denoting the following edge information. For example, the following three space-separated integers represents a edge from vertex 0 to vertex 2 and the edge distance is 10. The value of distance is always positive.

source	*destination*	*distance*
0	2	10

The symbol "#" represents that the inputs for building the weighted directed graph is finished. The lines followed by "#" contain two space-separated integers and denoting the source and destination node that you need to find the distance of the shortest path between them. For example, the table below means that you need to output the shortest path distance from vertex 0 to vertex 3. There must be path between the two vertices in the given input data.

source	*destination*
0	3

Sample Input / Output

Sample input	**Sample output**
4 0 1 1 0 2 10 0 3 4 1 3 2 # 0 3 0 2	3 10

Here is the graph for the sample input: