Graph basics concepts II

Description:

In this problem, your program will have to read a weighted graph (directed or non directed), and will perform one of three algorithms (topological sort, Kruskal, Dijkstra) on the graph (a letter will indicate the operation to be performed).

Input

The graph to be read comes in several lines: in the first line, you will have a value *d*: 0 (if the graph is a non directed graph), or 1 (if it is a directed one).

In the second line you will have a pair of numbers: *n* and *e*, the first is the quantity of nodes in the graph, and the second is the quantity of edges in the graph.

Then, you will have *e* lines, with triplets: *o*, *t* and *w*, where *o* is the source node of an edge, *t* is the target node, and *w* is the weight.

The triplets are ordered from the edges with origin at the root node (number 1), in an ascending way. Also, if there are more than one edge starting from one node, they are ordered by target node.

Finally, in the last line, comes a line with a unique letter: *g*, *h*, or *i*. Letter *g* means your program will have to perform a topological sort (the output will consist of the numbers of the nodes following a topological sorting (smallest numbered available vertex first). Letter *h* means your program will have to perform a Kruskal algorithm (the output will consist of two lines: the total cost of the minimun spanning tree, and the list of the edges for that tree -ordered from lower to higher cost-). Letter *i* means your program will have to perform a Dijkstra algorithm (the output will consist of the list of the paths from the node 1, to the rest of the nodes, and the cost of each path, ordered in an ascending way by cost). Take a look at the examples below.

The numbers of the nodes are consecutive: 1, 2, ...*n*

Output

You will have to show the right output for the graph and the letter read from the input. Take a look at the Examples below to see the format to follow.

Examples:

INPUT	OUTPUT
1 8 9 1 4 1 1 6 1 2 5 1 3 4 1 3 5 1 4 7 1 5 6 1 5 7 1 5 8 1 g	1 2 3 4 5 6 7 8

INPUT	OUTPUT

0 6 9 1 2 5 1 3 30 2 3 20 2 4 10 3 4 10 3 5 15 4 5 5 4 6 20 5 6 15 h	45 1-2 4-5 2-4 3-4 5-6

INPUT	OUTPUT
0 6 10 1 3 13 1 5 16 1 6 8 2 4 6 2 6 10 3 4 14 3 6 11 4 5 5 4 6 17 5 6 7 i	1-6 8 1-3 13 1-6-5 15 1-6-2 18 1-6-5-4 20