

# Graph basics concepts

## Description:

In this problem, your program will have to read a weighted graph (directed or non directed), and perform some basic algorithms on the graph:

- Visualization of the graph (in the same format of the input)
- BFS traversal and show the paths from the node 1 (the root node), to the rest of the nodes.
- DFS traversal and show the paths from the node 1 (the root node), to the rest of the nodes.
- Visualization of connected components
- Visualization of bridges

## 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.

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

## Output

You will have to show all the information in the list showed above (in the **Description**) section. Take a look at the Example below to see the format to follow.

## Example

### Input:

```
0
7 8
1 2 1
1 3 1
2 4 1
3 4 1
4 5 1
4 6 1
5 7 1
6 7 1
```

### Output:

```
0
7 8
1 2 1
1 3 1
```

2 4 1  
3 4 1  
4 5 1  
4 6 1  
5 7 1  
6 7 1

BFS:  
1 2 3 4 5 6 7

BFS Paths:  
1  
1 2  
1 3  
1 2 4  
1 2 4 5  
1 2 4 6  
1 2 4 5 7

DFS:  
1 2 4 3 5 7 6

DFS Paths:  
1  
1 2  
1 2 4 3  
1 2 4  
1 2 4 5  
1 2 4 5 7 6  
1 2 4 5 7

Connected Components:  
C1: 1 2 3 4 5 6 7

Bridges:  
0