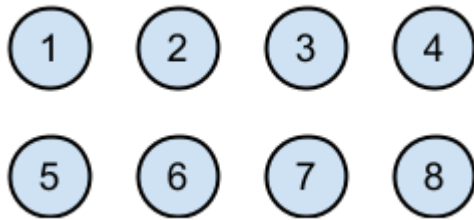
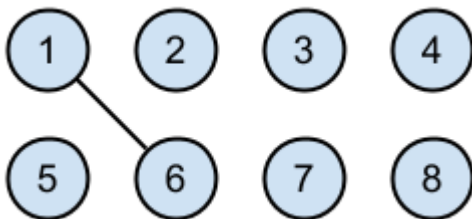


CONSIDER THE FOLLOWING PROBLEM

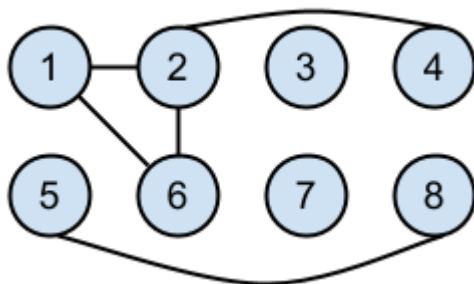
We have a set of elements. In this example I will use eight.



We can make a set of connections. For example we can connect 1 to 6.



We can make any number of connections and any two elements can be connected. Let's make the following connections: 1-2, 6-2, 2-4, 5-8



Now we need to be able to determine if two elements are connected, either directly or through a series of connections. 1 and 6 are connected, as are 6 and 4. But 7 and 4 are not connected, neither are 5 and 6.

TASK

Write a class `Network` in C#. It's not necessary to send us a program or console to run the test.

The constructor should take a positive integer value indicating the number of elements in the set. Passing an invalid value should throw an exception.

The class should also provide four public methods, `connect`, `disconnect`, `query` and `levelConnection`.

The `connect` method will take two integers indicating the elements to connect. This method should throw exceptions as appropriate.

The `disconnect` method will take two integers indicating the elements to disconnect. This method should throw exceptions as appropriate.

The query method will take two integers indicating the elements to disconnect. This method should throw exceptions as appropriate. It should return true if the elements are connected, directly or indirectly, and false if the elements are not connected.

The levelConnetion method will also take two integers and should also throw an exception as appropriate, returning an integer value. It should return 0 if the elements are not connected, 1 if the elements are directly connected and 2 or more when elements are indirectly connect, returning the number that represents how many connections there are between the searched elements.

The class can have as many private or protected members as needed for a good implementation

It's an opportunity to show us your skills. Don't use ChatGPT, we will review your code.