Algorithms and Computability laboratory task

[Requirements 1](#_Toc550144979)

[Size of a graph 1](#_Toc855044899)

[Distance function 1](#_Toc1526801374)

[Maximal clique 1](#_Toc640929034)

[Maximal common subgraph 1](#_Toc1128125128)

[Maximum common induced subgraph 2](#_Toc458761033)

[A Maximum common edge subgraph 2](#_Toc113333470)

[Input 2](#_Toc752928578)

[Summary 2](#_Toc1750925865)

[Methods 2](#_Toc1941038810)

[Tests 2](#_Toc696350705)

[Technical Details 2](#_Toc949850464)

[Conclusions 3](#_Toc962481511)

# Requirements

## Size of a graph

In literature, one will most often find the definition of size of a graph as the number of graph’s edges. To be more specific, having a graph G = (V, E). Where V is the set of vertices, and E is the set of edges, size of a graph is defined by |E|.  
Since, one of the goals of the laboratory task is to define the computational complexity of algorithms used, throughout our work we will work with a different definition. We assume that size of a graph is the number of edges plus number of vertices (|V| + |E|). Solely to avoid a situation where a non-empty graph would be of size 0.

## Distance function

As a distance function we use the measure of graph edit distance. Graph edit distance measures the number of elementary operations needed to transform one graph isomorphic to another desired graph. Elementary operations include:

* Adding or removing isolated vertices
* Adding or removing edges

## Maximal clique

A *clique* of a graph *G* can be defined as a *complete subgraph* of *G.* A subgraph is a graph whose vertices and edges are a subset of another graph, in this case graph *G*. Complete graph is a graph, in which every pair of distinct vertices is connected by an edge. A clique is maximal if it is not possible to add another vertex to the clique, in other words a maximal clique is not a subset of any other clique of a given graph.

## Maximal common subgraph

We distinguish two categories of a maximal common subgraph: maximum common induced subgraph and maximum common edge subgraph.

### Maximum common induced subgraph

An *induced subgraph* is a graph constructed from a subset of vertices of a given graph G and all of the edges connecting the subset of vertices present in graph G. A maximum common induced subgraph of a given set of graphs Sis graph Hthat is an induced subgraph of each of the graphs in set S and graph H has as many vertices as possible.

### A Maximum common edge subgraph

A graph H is called a maximum common edge subgraph of given graph set S if it’s such a graph that is isomorphic to subgraphs of S and has got as many edges as possible.

#TODO: Add references to sections above

## Input

Input given to the program must follow certain rules that have been commonly agreed upon.

* Input should be given in a form of a .txt file containing descriptions of one or more graphs separated by a blank line.
* The first line of the file has the number of graphs present in the file.
* Next lines concern the description of graphs.
  + First line (second line in total in case of the first graph) contains the number of vertices of the following graph.
  + Next the rows of the adjacency matrix of a given graph are presented in a format of a number, followed by space or newline, in the case of end of a row.
  + After the adjacency matrix the file might contain some additional data about the graph.

## Summary

The purpose of this task is to create an implementation of two algorithms:

* One that finds a maximum common subgraph of two (or more) given graphs
* Another one that finds a maximal clique of a graph

The algorithms have been implemented in a form of single executable written in C language that accepts files in format presented in sections above. The implemented algorithms have been thoroughly tested. Results of these tests are presented in sections below.

# Methods

#TODO Find algorithms for finding maximal clique and maximal common subgraph

# Tests

#TODO run some tests, time them

# Technical Details

#TODO short with detailed instructions for compiling and running

# Conclusions

#TODO Conclusions