Introduction

Table of Contents

[Applications 4](#_Toc141041180)

[Konigsburg Bridge 4](#_Toc141041181)

[Utility Problem 4](#_Toc141041182)

[Electrical Networks 5](#_Toc141041183)

[Seating Arrangement Problem 5](#_Toc141041184)

[Finite and Infinite Graphs 6](#_Toc141041185)

[Incidence 6](#_Toc141041186)

[Adjacency 6](#_Toc141041187)

[Degree 7](#_Toc141041188)

[Regular Graph 8](#_Toc141041189)

[Isolated Vertex 8](#_Toc141041190)

[Pendant Vertex 8](#_Toc141041191)

[Series Edges 8](#_Toc141041192)

[Null Graph 9](#_Toc141041193)

[Isomorphism 9](#_Toc141041194)

[Subgraphs 10](#_Toc141041195)

[Edge-Disjoint Subgraph 11](#_Toc141041196)

[Vertex-Disjoint Subgraphs 12](#_Toc141041197)

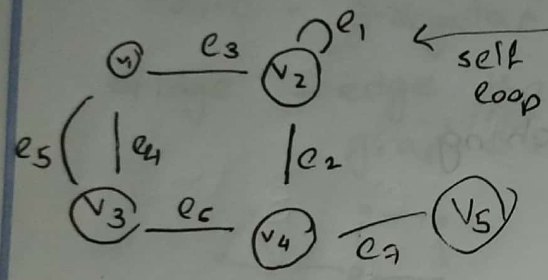
[Walks 12](#_Toc141041198)

[Paths 13](#_Toc141041199)

[Circuits 14](#_Toc141041200)

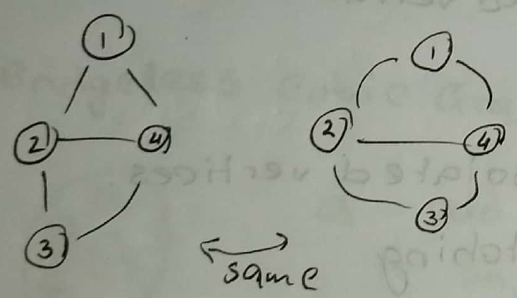
[Connected Graphs 15](#_Toc141041201)

A graph is described as , where is the set of **vertices** of the graph and is the set of **edges**. Typically, the vertices represent objects and the edges represent the relationships between the objects. If there is an edge, , between two vertices and , the vertices are said to be **related**. and are the two **endpoints** of .

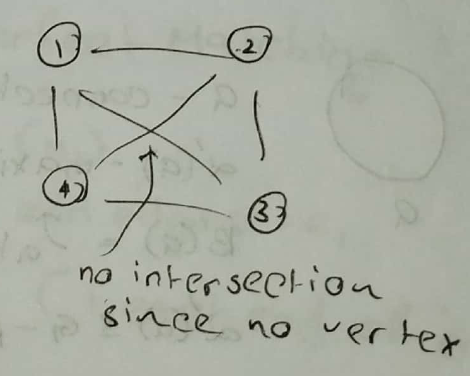


If the two end points of an edge are the same vertex, the edge is called a **self-loop**. If two edges, and , have the same end points, they are called **parallel edges**. Graphs that have no self-loops or parallel edges are called **simple graphs**. Unless otherwise state, it should always be assumed that a graph is a simple graph.

Two important characteristics of graphs are that the positions of the vertices and the shapes of the edges do not matter, and intersections only occur at vertices.



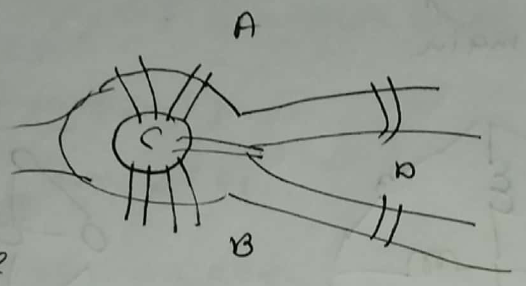
The graphs above are the same.



## Applications

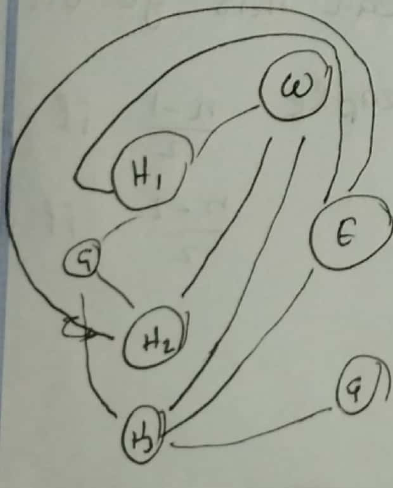
### Konigsburg Bridge

This is a famous problem that created the map shown below and asked whether it was possible to cross all of the edges exactly once. It was later shown to not be possible.



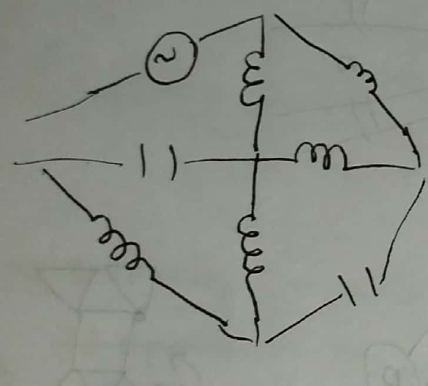
### Utility Problem

This problem requires us to supply three resources (water, gas and electricity), to three houses using pipes that do not cross or overlap). It was also shown to not be possible.



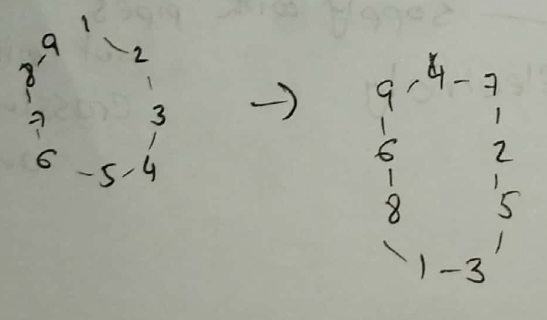
### Electrical Networks

Given an electrical network, we may need to find the total electric flow. This depends on various things like the nature and value of the components and how the components are connected.



### Seating Arrangement Problem

The problem is to seat a group of people in a round table in such a way that every day, the neighbours of each person is different. We need to calculate how many times this can be done with no repetitions. The diagrams below show two possible arrangements.



## Finite and Infinite Graphs

A **finite graph** is one where both the number of vertices and the number of edges is finite. In any other case, the graph is an **infinite graph**. It should be assumed that a graph is a finite graph unless otherwise assumed.

## Incidence

If a vertex is one endpoint of an edge , then and are said to be **incident** upon each other.

## Adjacency

If and are two endpoints of an edge, the two vertices are said to be **adjacent** to each other.

If two edges and share a common vertex, the edges are said to be **adjacent** to each other.

## Degree

The number of edges incident with a vertex is the **degree** of a vertex. For self-loops, a degree of 2 is counted. The degree of a vertex is denoted as .

For any graph, the total degree is the sum of the degrees of all its vertices. Since each edge contributes 2 degrees, there should be degrees in a graph.

Theorem: The number of vertices with odd degree in a graph is always even.

Proof:

We know that:

For the subset of vertices that have an even degree, the sum of the degrees must be even since the sum of a group of even numbers is always even. The total number of degrees is , which must also be an even number since any multiple of 2 is even. Since the only remaining term is the subset of vertices with odd degree, the sum of their degrees must also be even since, if they were odd, would have to be an odd number.

## Regular Graph

If every vertex of a graph has the same degree, it is called a **regular graph**.

## Isolated Vertex

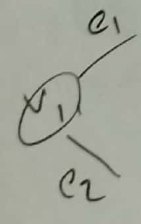
A vertex with no incident edges is called an **isolated vertex**.

## Pendant Vertex

A vertex with a **single incident edge** is called a **pendant vertex**.

## Series Edges

If two edges share a vertex and that vertex has a degree of two, meaning these two edges are the only incident edges for that vertex, the two edges are said to be in a **series**.

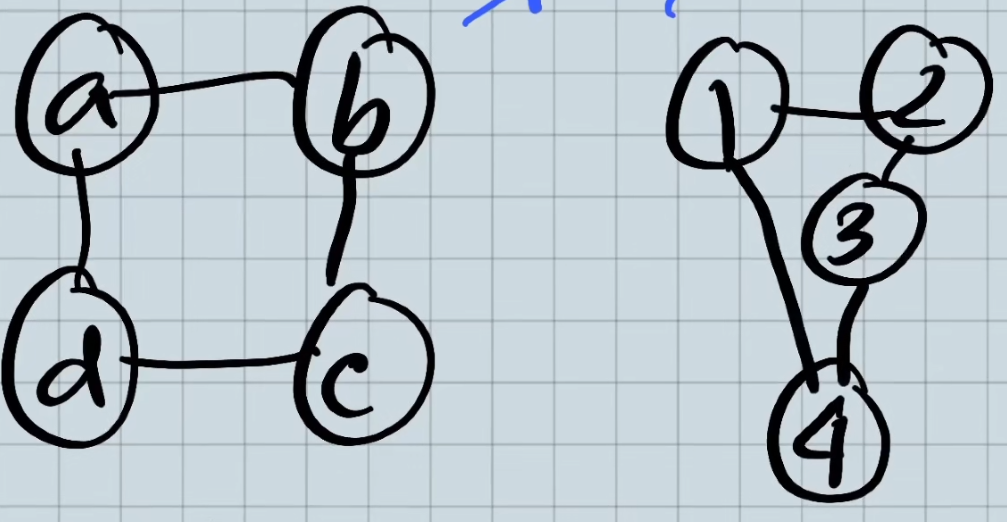


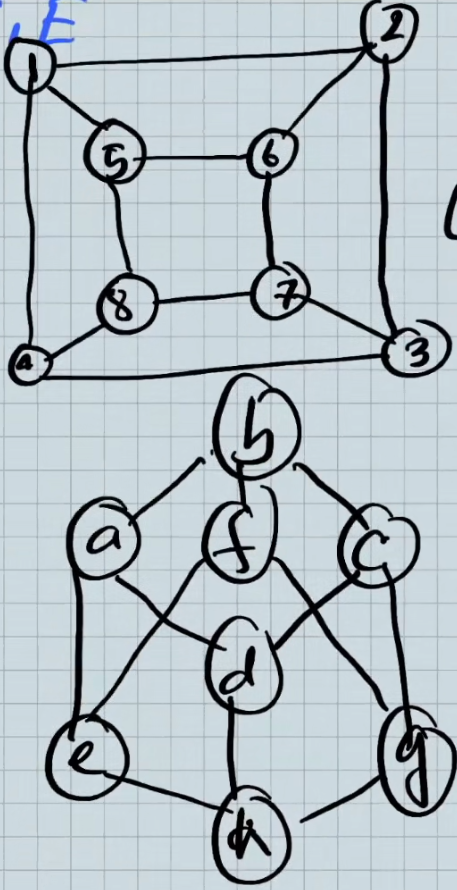
## Null Graph

A graph that has vertices but no edges, it is called a **null graph**.

## Isomorphism

In geometry, two figures are said to be **congruent** if they have identical geometric properties. In graph theory, a similar concept is used to define **isomorphism**, in that if two graph have the same graphical properties, they are said to be **isomorphic**. Formally, two graphs, and , are said to be isomorphic if they have a one-to-one correspondence between their vertices, and , and their edges, and , in such a way that their incidence relationships are preserved.





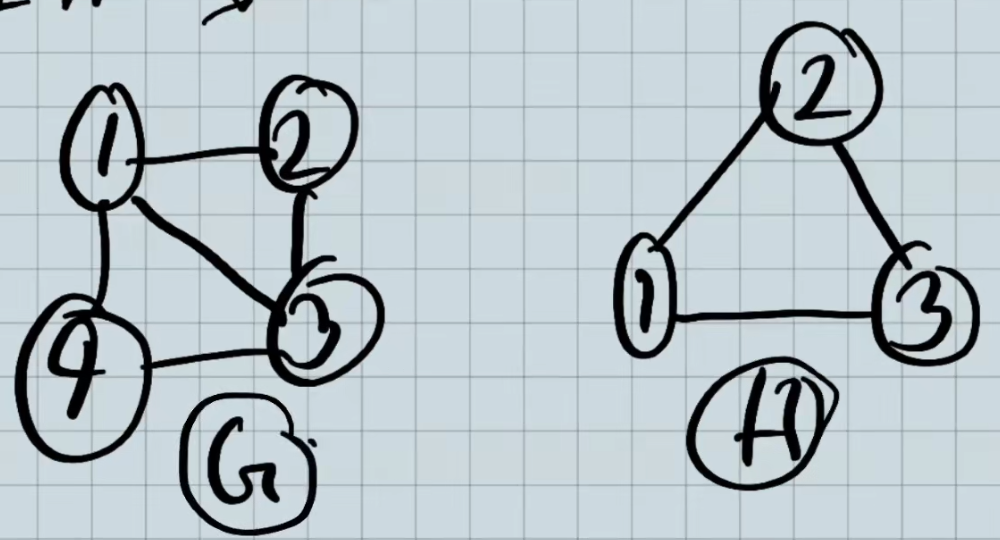
Thus, for isomorphic graphs:

* The number of vertices is the same.
* The number of vertices with a given degree is the same.
* The number of edges is the same.

These conditions are necessary but not sufficient.

## Subgraphs

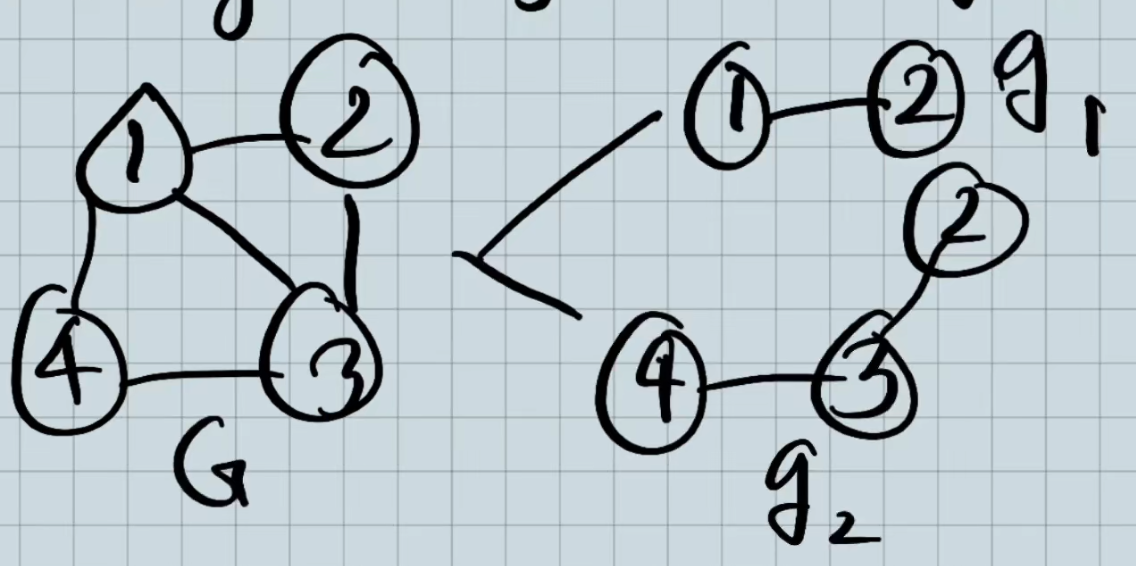
A graph is a **subgraph** of , , if and . In addition, the endpoints of each of the edges in must also be the endpoints of each of the edges of .



Some properties of subgraphs are:

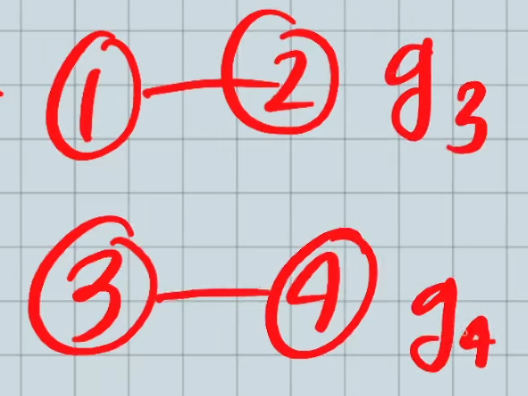
* If , and , then .
* A subgraph can contain a single vertex (with no edges).

## Edge-Disjoint Subgraph



Both and are subgraphs of but these two subgraphs have no edges in common. Thus, they are called **edge-disjoint subgraphs**. .

## Vertex-Disjoint Subgraphs

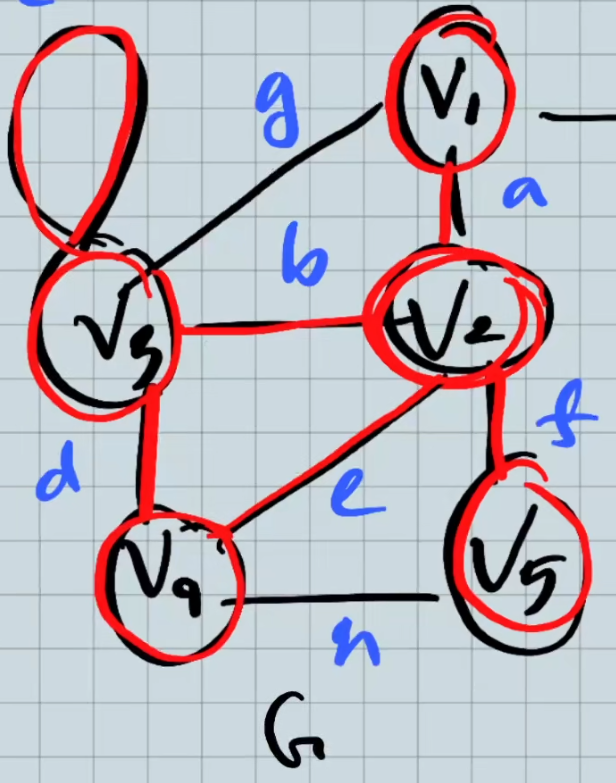


and are subgraphs of that do not even have vertices in common. Thus, they are called **Vertex-Disjoint Subgraphs**. Since there are no vertices in common, by default, there won’t be any edges in common.

## Walks

A **walk** is a finite sequence that alternates between vertices and edges. It will begin with a vertex and end with a vertex in such a way that it is incident with the edges that are preceding and following it. We must ensure that no edge appears more than once.

The idea behind that complicated definition is simple. In the graph below, one possible walk is . Thus, we went from one vertex to another using different edges, without repeating any of the edges. We do not have to cover all of the edges or all of the vertices.



Notice that the walk also creates a **subgraph**.

The two vertices at the start and end of the walk are called the **terminal vertices**. If the terminal vertices are the same vertex, the walk is called a **closed walk**. If they are different vertices, the walk is called an **open walk**.

## Paths

An **open walk** that has **no repeated vertices** is called a **path**. A **closed walk** can never be a path because at least the terminal vertices will repeat.

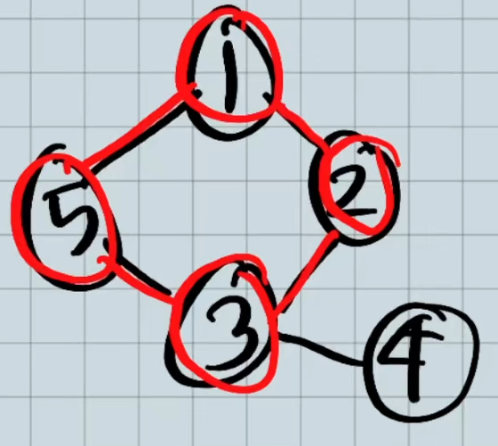


The number of edges in a path is called the **length** of a path. For the above path, the length is 4. A path can even be of length 1, as long as the edge being considered is **not a self-loop**. Self-loops can never be a part of a path since it would cause a repeated vertex.

The terminal vertices of a path have a degree of 1, but the non-terminal vertices have a degree of 2.

## Circuits

For a **closed walk**, other than the terminal vertices, if none of the vertices repeat, it is called a **circuit**. A self-loop is also a circuit.

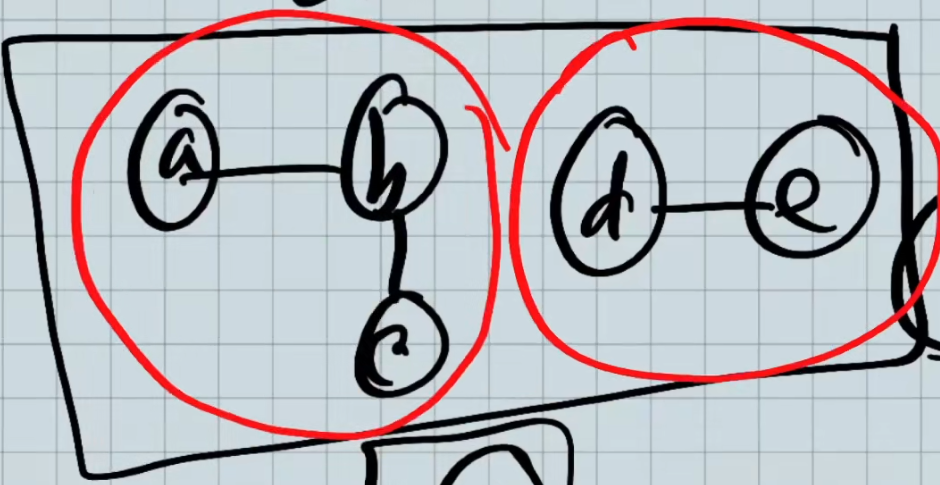


## Connected Graphs

A graph is **connected** if and only if there is at least one path between any pair of vertices. If this is not true, the graph is called a **disconnected graph**.

If a graph has a **single vertex**, it is also considered to be connected.

Each of the **connected subgraphs** inside a disconnected graph is called a **component**. In the graph below, there are two components.



**Theorem 2.1**: A graph is disconnected if and only if the vertex set of , , can be partitioned into two non-empty and disjoint sets, and , such that there is no edge in with one end vertex in and the other end vertex in .

Proof:

Suppose we create a graph that follows the setup as defined in the theorem. If this is the case, then for any vertex in , there isn’t any edge between that vertex and any of the vertices in . Thus, we cannot form a path from any vertex in to any vertex in . This violates the definition of a connected graph, which requires that ever vertex have a path to every other vertex in the graph. As such, the graph must be disconnected.

On the flip side, if there is a graph that is disconnected, we can select any vertex and place along with any vertices to which there is a path from in . For the remaining vertices, we can place them in . Since the graph is disconnected, we know for sure that will not be empty. We also know that these two sets do not share any edges because if any vertex in shared an edge with any vertex in , it would have been placed in .

**Theorem 2.2**: If a graph has exactly two vertices of odd degree, there must be a path between them.

Proof:

If is connected, then the relationship definitely holds because there is always a path from any vertex to any other vertex.

If is disconnected, then suppose there are two components, and . The only way for the two vertices with odd degrees, and , to not have a path between them is if one vertex is in and the other is in . However, if this is the case, then the number of vertices with odd degree in and the number of vertices with odd degree in would be odd. A subgraph is also a graph, and a graph cannot have an odd number of vertices with odd degree (as proven in a previous theorem). Thus, this case cannot exist and and must both be in the same component and thus have a path between them.