UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 10

# Intro to Graphs

*Submitted by:*
Bron, Jhustine A.

*Instructor:*
Engr. Maria Rizette H. Sayo

October 11, 2025

# I. Objectives

Introduction

A graph is a visual representation of a collection of things where some object pairs are linked together. Vertices are the points used to depict the interconnected items, while edges are the connections between them. In this course, we go into great detail on the many words and functions related to graphs.

An undirected graph, or simply a graph, is a set of points with lines connecting some of the points. The points are called nodes or vertices, and the lines are called edges.

A graph can be easily presented using the python dictionary data types. We represent the vertices as the keys of the dictionary and the connection between the vertices also called edges as the values in the dictionary.

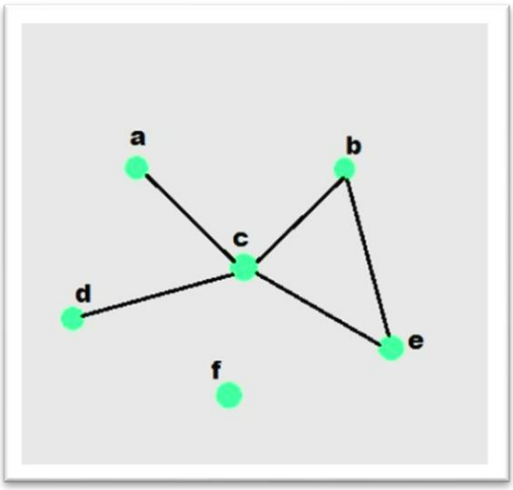



Figure 1. Sample graph with vertices and edges

This laboratory activity aims to implement the principles and techniques in:

- To introduce the Non-linear data structure – Graphs
- To discuss the importance of Graphs in programming

# II. Methods

A. Discuss the following terms related to graphs:

1. Undirected graph
2. Directed graph
3. Nodes
4. Vertex
5. Degree
6. Indegree
7. Outdegree

8. Path
9. Cycle
10. Simple Cycle

# III. Results

**Undirected Graph** – An undirected graph is a type of graph where the edges have no direction. This means that if two vertices are connected, you can move between them both ways.

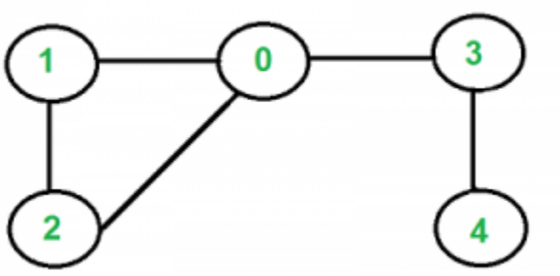



*Figure 1. Undirected Graph*

**Directed Graph** – A directed graph, or digraph, has edges that point in a specific direction. It shows a one-way relationship between two vertices, usually represented with arrows.

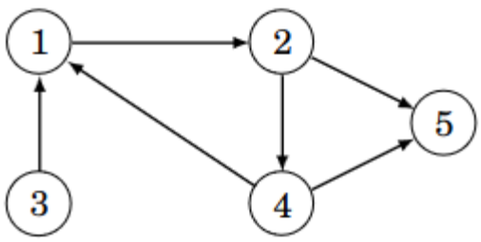



*Figure 2 Directed Graph*

**Nodes** – Nodes are the basic elements of a graph that represent objects, data points, or entities. In real-world examples, nodes can represent things like computers in a network or people in a social group.

**Vertex** – A vertex is another term for a node. It's a point in the graph where edges meet or connect to other vertices.

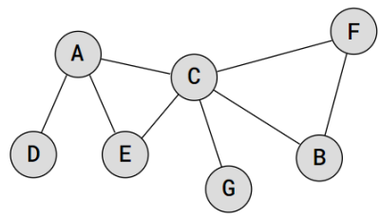
**Degree** – The degree of a vertex is the number of edges connected to it. In an undirected graph, this counts all the connections a vertex has.

**Indegree** – In a directed graph, the indegree refers to how many edges are coming *into* a vertex from other vertices.

**Outdegree** – The outdegree is the number of edges that go *out* from a vertex to other vertices in a directed graph.

**Path** – A path is a sequence of connected edges that lets you travel from one vertex to another. It shows how one point can reach another within the graph.

**Cycle** – A cycle is a path that starts and ends at the same vertex without reusing any edges. It




```
Is cyclic:
```

*Figure 3 Cyclic graph*

forms a closed loop within the graph.

**Simple Cycle** – A simple cycle is a cycle that doesn't visit any vertex more than once, except for the starting and ending vertex. It helps identify clean, non-repetitive loops in a graph.

# IV. Conclusion

In conclusion, these terms help us understand the structure and behavior of graphs, which are used to represent relationships and connections between objects. Knowing the difference between directed and undirected graphs, and concepts like degree, path, and cycle, makes it easier to analyze how elements interact or connect in a system. Overall, understanding these basics is important since graphs are used in many real-world applications like computer networks, social media, and data organization.

# References


GeeksforGeeks, “What is Undirected Graph? | Undirected Graph meaning,” *GeeksforGeeks*, Jul. 23, 2025. https://www.geeksforgeeks.org/dsa/what-is-unidirected-graph-undirected-graph-meaning/

“13. Directed graphs – Data Structures and Algorithms.” https://tira.mooc.fi/spring-2025/chap13/

GeeksforGeeks, “Graph terminology in data structure,” *GeeksforGeeks*, Jul. 23, 2025. https://www.geeksforgeeks.org/dsa/graph-terminology-in-data-structure/

Krivalar, “Graph - degree,” *Krivalar.com*. https://www.krivalar.com/graph-degree-indegree-outdegree

“W3Schools.com.” https://www.w3schools.com/dsa/dsa_algo_graphs_cycledetection.php

“Simple path and simple cycle in Graph DataStructure,” *Stack Overflow*. https://stackoverflow.com/questions/62755631/simple-path-and-simple-cycle-in-graph-datastructure