UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 10

# Intro to Graphs

*Submitted by:*
Acebedo, Sebastian C.

*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.

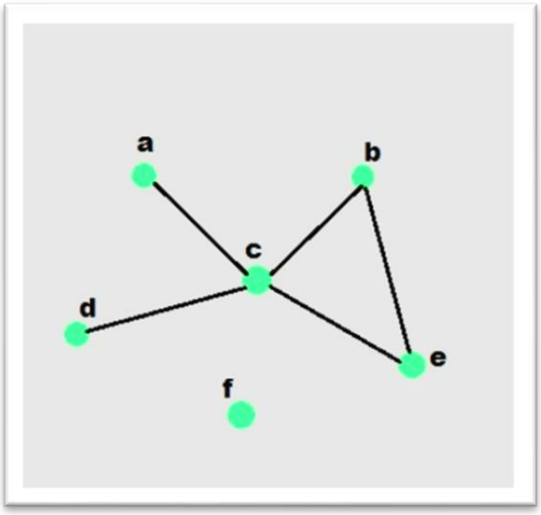


Figure 1. Sample graph with vertices and edges

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

**1. Undirected Graph -** An undirected graph is a graph where the edges don't have a direction. This means the connection between two nodes goes both ways.

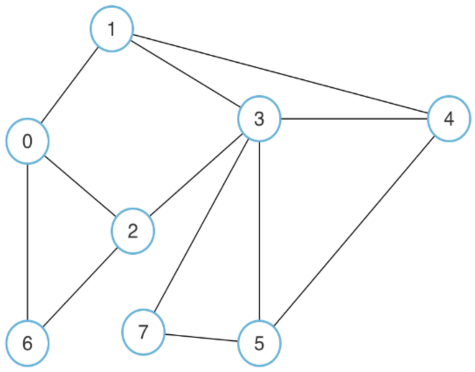


Figure 1

Please refer to this link: https://ds1-iiith.vlabs.ac.in/exp/depth-first-search/basics-of-graphs.html

2. **Directed Graph** - A directed graph has edges that go in a specific direction. Each edge points from one node to another.

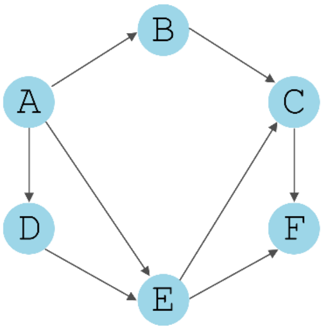


Figure 2

Please refer to this link: https://fity.club/lists/suggestions/directed-graph/

**3. Nodes -** Nodes are the points in a graph. They represent elements like objects or positions in the graph.

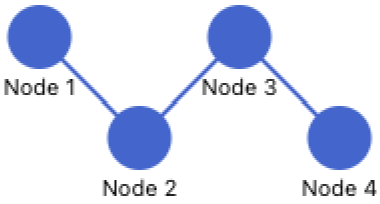


Figure 3

Please refer to this link: https://dash14.github.io/v-network-graph/examples/

4. **Vertex** - A vertex is another word for a node. They mean the same thing.

5. **Degree** - The degree of a node is the number of edges connected to it. In undirected graphs, it counts all edges touching that node.

6. **Indegree** - Indegree is the number of edges coming **into** a node in a directed graph.

7. **Outdegree** - Outdegree is the number of edges going **out from** a node in a directed graph.

8. **Path -** A path is a sequence of nodes connected by edges. It shows a way to move from one node to another.

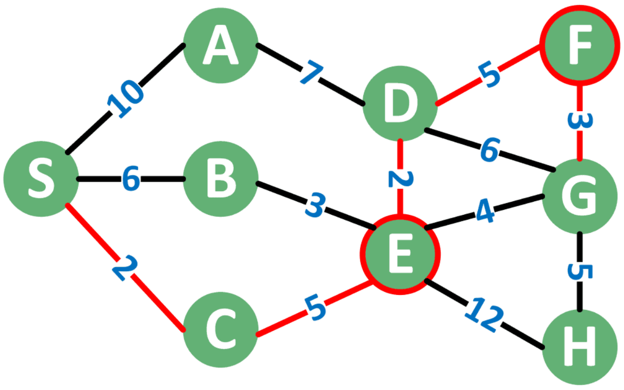
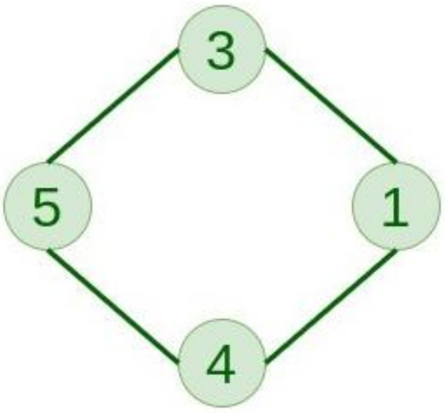


Figure 4

Please refer to this link: https://www.baeldung.com/cs/shortest-path-to-nodes-graph

**9.** **Cycle -** A cycle is a path that starts and ends at the same node, forming a loop.



Cycle Graph

Figure 5

Please refer to this link: https://www.geeksforgeeks.org/introduction-to-graphs/

10. **Simple Cycle** - A simple cycle is a cycle where no node is repeated except the starting and ending node.

# IV. Conclusion

In conclusion, graphs are a way to represent items and their connections using nodes (or vertices) and edges. They can be undirected, where edges have no direction, or directed, where edges point from one node to another. Each node has a degree, and in directed graphs, we also have indegree and outdegree. Paths show how nodes are connected, and cycles are paths that start and end at the same node, with simple cycles being those where no node is repeated except the start and end. Graphs can be easily represented in Python using dictionaries, with vertices as keys and edges as values. Understanding these concepts is important for analyzing relationships and connections in different kinds of data.

# References

[1] Co Arthur O.. “University of Caloocan City Computer Engineering Department Honor Code,” UCC-CpE Departmental Policies, 2020.