



UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT



Data Structure and Algorithm

Laboratory Activity No. 10

Intro to Graphs

Submitted by:
Maringal, Czer Justine D.

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.

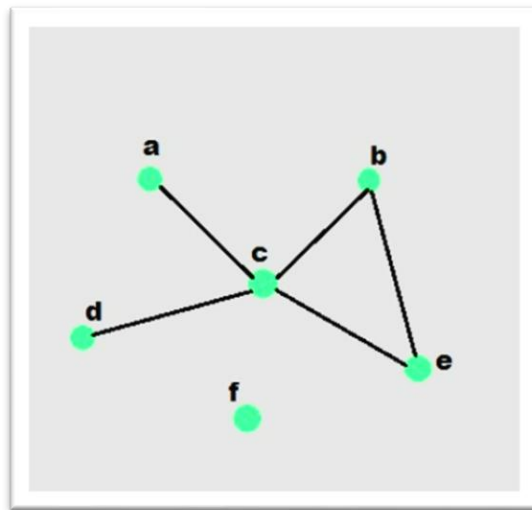


Figure 1. Sample graph with vertices and edges

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: A graph where edges have no direction; the connection between two vertices is bidirectional.
2. Directed Graph (Digraph): A graph where edges have a direction, pointing from one vertex to another.
3. Nodes / Vertices: The fundamental units of a graph that represent entities. The terms "node" and "vertex" are used interchangeably.
4. Edge: A connection between two vertices that represents a relationship.
5. Degree: The number of edges incident to a vertex. In an undirected graph, it's the count of connections.
6. Indegree: (For directed graphs) The number of edges coming into a vertex.
7. Outdegree: (For directed graphs) The number of edges going out from a vertex.
8. Path: A sequence of vertices where each adjacent pair is connected by an edge.
9. Cycle: A path that starts and ends at the same vertex, with no repeated edges.
10. Simple Cycle: A cycle that repeats only the first and last vertex, with no other repeated vertices.

IV. Conclusion

This laboratory activity provided a fundamental introduction to graphs as a non-linear data structure. By learning this terminology like vertices, edges, paths, and cycles, and the difference between directed and undirected graphs, I understand now the basic building blocks needed to work with graphs. Recognizing that graphs can be for understanding more complex algorithms and appreciating the importance of graphs in modeling real-world networks like social media, maps, and recommendation systems.

References

[1] Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2013). Data Structures and Algorithms in Python. Wiley.

[2] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press.

[3] "Graph Theory," GeeksforGeeks. [Online]. Available: <https://www.geeksforgeeks.org/graph-data-structure-and-algorithms/>