UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT


Data Structure and Algorithm

Laboratory Activity No. 10

# Intro to Graphs

*Submitted by:*
Talagtag Mark Angel T.

*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.

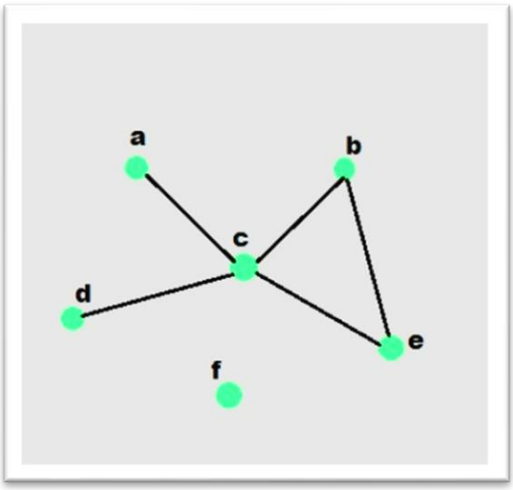


Figure 1. Sample graph with vertices and edges

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 in which edges have no direction. That means the connection between two nodes is bidirectional.
If there is an edge between node A and node B, you can travel from A to B and from B to A.
Example: Social networks like friendships where connections are mutual.

2. Directed Graph (Digraph)
A graph where edges have a direction, represented as arrows.
An edge from node A to node B means you can travel only from A to B, not necessarily from B to A.
Example: Twitter follows (A follows B doesn't mean B follows A).

3. Nodes
Also called vertices, nodes are the fundamental units or points in a graph where edges connect.
In a social network graph, nodes represent people.
The plural form is nodes.

4. Vertex
A single point or node in a graph.
The terms node and vertex are often used interchangeably.

5. Degree
The degree of a node in an undirected graph is the number of edges connected to it
It tells how many direct connections the node has.
For example, if a node has 3 edges connected, its degree is 3.

6. Indegree
In a directed graph, the indegree of a node is the number of edges coming into it.
It represents how many nodes are pointing to this node.

7. Outdegree
In a directed graph, the outdegree of a node is the number of edges going out from it.
It represents how many nodes this node is pointing to.

8. Path

A sequence of edges and nodes where each adjacent pair of nodes is connected by an edge.

Paths show a route from one node to another.

Example: A path from node A to node D could be A → B → C → D.

9. Cycle

A path that starts and ends at the same node, with all edges and intermediate nodes distinct (except the start/end node).

In other words, you can start at a node, follow edges, and come back to it without retracing any edge.

10. Simple Cycle

A cycle that does not repeat any nodes or edges, except for the start/end node.

No nodes (other than the first and last) appear more than once in the cycle.

It's the simplest form of a cycle without loops or repeated parts.

# IV. Conclusion

Graphs are fundamental structures in computer science and mathematics, used to model relationships between entities. Understanding key terms such as undirected and directed graphs, nodes (vertices), and concepts like degree, indegree, outdegree, paths, and cycles is essential for analyzing and designing efficient algorithms. These concepts help in representing and solving real-world problems ranging from social networks to transportation systems. Mastery of graph terminology provides a solid foundation for further study and practical applications in graph theory, data structures, and network analysis.

# References

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

West, D. B. (2001). *Introduction to Graph Theory* (2nd ed.). Prentice Hall.

Diestel, R. (2017). *Graph Theory* (5th ed.). Springer.

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

Graph Algorithms - GeeksforGeeks

Graph Data Structure