UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 10

Intro to Graphs

Submitted by:
AMPONG, J-KEVIN L.

Instructor:
Engr. Maria Rizette H. Sayo

10, 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.

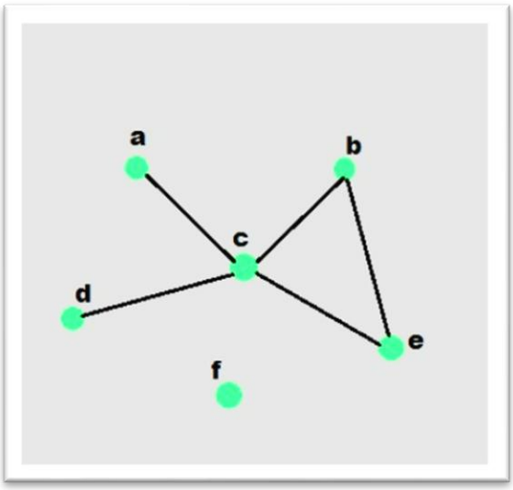

Figure 1. Sample graph with vertices and edges

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 set of vertices (or nodes) connected by edges that have no specific direction. An edge between vertex A and vertex B indicates a mutual, two-way relationship you can move from A to B and also from B to A.

Example: A social network where friendships are always reciprocal.

2. Directed Graph

A directed graph consists of vertices and edges, but each edge has a specific direction, often shown with an arrow. An edge from vertex A to vertex B represents a one-way connection movement is possible from A to B, but not necessarily from B to A.

Example: A flowchart showing task order, or a Twitter network where following is one-way.

3. Nodes

Nodes are the basic components of a graph. They represent individual entities or data points that can be connected by edges.

4. Vertex

A vertex (plural: vertices) refers to a single node in a graph. The collection of all vertices in a graph is usually represented by the symbol V.

5. Degree

The degree of a vertex, usually written as deg(v), is the number of edges connected to that vertex. In an undirected graph, it's the total count of edges incident to the vertex.

In a directed graph, the degree is the sum of its indegree and outdegree.

6. Indegree

The indegree of a vertex (denoted as $\deg^-(v)$) applies only to directed graphs. It represents the number of edges that enter or terminate at that vertex.

7. Outdegree

The outdegree of a vertex (denoted as $\deg^+(v)$) is also used only in directed graphs. It refers to the number of edges that originate or leave from that vertex.

8. Path

A path in a graph is an ordered sequence of vertices and edges where each edge connects the vertex before it to the vertex after it. A path begins and ends with a vertex.

9. Cycle

A cycle is a path that starts and ends at the same vertex and includes at least one edge. It forms a closed loop in which all edges are distinct.

10. Simple Cycle

A simple cycle (also called a simple circuit) is a cycle where the starting and ending vertex is the only one that repeats. No other vertex appears more than once in the path.

Example: In a graph with vertices A, B, and C — A → B → C → A is a simple cycle, while A → B → C → B → A is not, since B is revisited.

IV. Conclusion

The key principles of graph theory were clearly explained, presenting the graph as a data structure used to represent relationships between entities. Essential terms were defined, emphasizing the core components such as nodes (or vertices) and their connections through edges. The difference between undirected and directed graphs was highlighted as an important factor in modeling various real-world situations. Finally, the concepts of degree, indegree, and outdegree illustrate how connected each vertex is, while path, cycle, and simple cycle describe the possible routes or relationships that can occur within the graph.

References

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