UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 10

Intro to Graphs

Submitted by:
Catahan, Joshua A.

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.

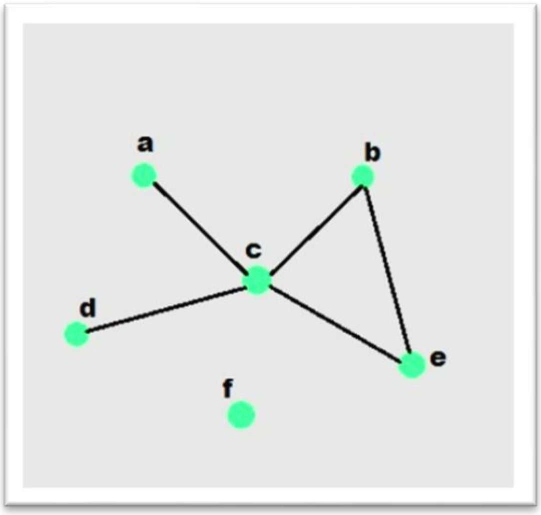

Figure 1. Sample graph with vertices and edges

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

Based on my research from geeksforgeeks.org:

1. Undirected Graph

An undirected graph is a type of graph where the edges have no direction. This means you can move between two connected nodes both ways. For example, if there's an edge between A and B, you can go from A to B or from B to A. It's commonly used to represent mutual relationships like friendships or road connections.

2. Directed Graph

A directed graph, also called a digraph, has edges that point in a specific direction. Each edge goes from one vertex to another, showing one-way movement. For instance, if there's an arrow from A to B, it doesn't mean you can go back from B to A unless another edge points that way. This kind of graph is often used in situations like following links between web pages or representing one-way roads.

3. Nodes

Nodes, also known as vertices, are the individual points in a graph that hold data or represent entities. They are basically the main elements that the edges connect. For example, in a social network, each person is a node connected to others through edges that represent friendships.

4. Vertex

A vertex is another term for a node. It represents one element or point in the graph. Each vertex can have several edges connecting it to other vertices, depending on how many relationships or links it has.

5. Degree

The degree of a vertex refers to the number of edges connected to it. It shows how many direct links that vertex has. In an undirected graph, it's simply the count of edges touching the vertex, while in a directed graph, it's divided into indegree and outdegree.

6. Indegree

Indegree means the number of edges that are coming into a vertex. It tells how many other vertices are pointing toward it. For example, in a graph of social media followers, the indegree would represent how many people are following a certain user.

7. Outdegree

Outdegree is the opposite of indegree. It's the number of edges that are going out from a vertex. Using the same social media example, the outdegree would show how many people a user is following.

8. Path

A path is a sequence of connected vertices where you can travel from one vertex to another by following the edges. For example, if you can go from A to B and then from B to C, the path is A–B–C. Paths are important in finding routes or connections in networks.

9. Cycle

A cycle happens when a path starts and ends at the same vertex without repeating any vertex in between. It’s like going around a loop and returning to where you started. Cycles often appear in graphs that have circular connections, such as feedback systems.

10. Simple Cycle

A simple cycle is a special type of cycle where no vertices or edges are repeated except for the starting and ending vertex. It’s the cleanest form of a cycle and helps identify loops in a graph without unnecessary repetitions.

IV. Conclusion

In this activity, I learned that graphs are important in showing relationships between data. I understood the difference between directed and undirected graphs and how terms like degree, path, and cycle describe their structure. Overall, graphs help solve real-life problems such as mapping routes or connecting networks, making them useful in programming and data analysis.

References

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

[2] GeeksforGeeks. (n.d.). *Graph Terminology in Data Structure.* Retrieved from https://www.geeksforgeeks.org/dsa/graph-terminology-in-data-structure/