UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 10

Intro to Graphs

Submitted by:
Directo, Hannah Thea B.

Instructor:
Engr. Maria Rizette H. Sayo

October 12, 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.

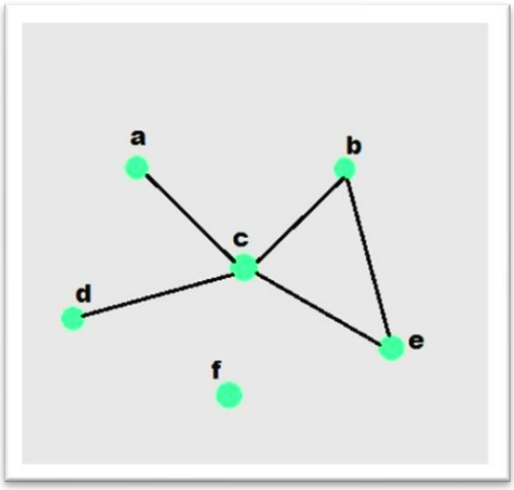

Figure 1. Sample graph with vertices and edges

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 in which edges have no direction. This means the connection between two vertices is bidirectional; if vertex A is connected to vertex B, then B is also connected to A. There is no concept of direction in the edge.

2. Directed Graph - A directed graph (or digraph) consists of vertices connected by edges that have a specific direction. Each edge is represented as an ordered pair of vertices (u, v), indicating movement from u to v. Directed graphs are used in situations where the relationship is one-way, such as following someone on social media or hyperlink connections between web pages.

3. Nodes - Nodes, also called vertices, are the fundamental units or points in a graph that represent entities such as people, cities, or web pages. Each node may be connected to one or more other nodes via edges. In both directed and undirected graphs, nodes serve as the basis for defining structure and relationships.

4. Vertex - A vertex is another term for a node in a graph. It serves as the endpoint of edges and represents an object or entity in the system being modeled. In graph theory literature, "vertex" is the standard term, while "node" is often used in applications like networking.

5. Degree - The degree of a vertex in an undirected graph is the number of edges connected to it. It represents how many direct connections the vertex has to other vertices. In a directed graph, the concept is further divided into indegree and outdegree.

6. Indegree - The indegree of a vertex in a directed graph is the number of edges coming into that vertex. It reflects how many other vertices point to this one. This concept is important in applications like analyzing citation networks or data flow.

7. Outdegree - is the number of edges directed *outward* from a vertex. It reflects how many vertices the current vertex is connected to or influences**.**

8. **Path** - A path in a graph is a sequence of vertices connected by edges. Each consecutive pair in the sequence must be directly connected by an edge, and in directed graphs, the direction of edges must be respected. Paths are essential for determining reachability and distances.

9. **Cycle -** A cycle is a closed path that begins and ends at the same vertex, with no other vertices repeated. Cycles are important in detecting loops or redundancies in networks, such as feedback systems or circular dependencies.

10. **Simple Cycle** - A simple cycle is a cycle where no vertices (except the starting/ending vertex) are repeated. This represents the minimal form of a loop and is used to detect foundational circular structures in graphs

IV. Conclusion

This laboratory activity introduced us to the basic concepts and structure of graphs. By learning important terms such as undirected and directed graphs, nodes, degrees, paths, and cycles, we gained a better understanding of how graphs are used to show connections and relationships between different elements. The activity helped us recognize the difference between one-way and two-way links, and how to measure the connections a point has in a system. Creating and representing graphs using Python dictionaries made the lesson more practical, showing how graphs can be used to solve real-life problems such as organizing data, analyzing networks, or mapping routes.

References.

[1] Bondy, J. A., & Murty, U. S. R. "*Graph theory"* (Vol. 244). Springer, 2008
https://doi.org/10.1007/978-1-84628-970-5

[2] GeeksforGeeks. "*What is Undirected Graph? | Undirected Graph meaning*." GeeksforGeeks, 2025, July 23
https://www.geeksforgeeks.org/dsa/what-is-unidirected-graph-undirected-graph-meaning/

[3] Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. "*Data structures and algorithms in Python*." Wiley, 2014
https://www.wiley.com/en-us/Data+Structures+and+Algorithms+in+Python-p-9781118290279