UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 10

Intro to Graphs

Submitted by:
Balaoro, Judge Wayne B.

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.

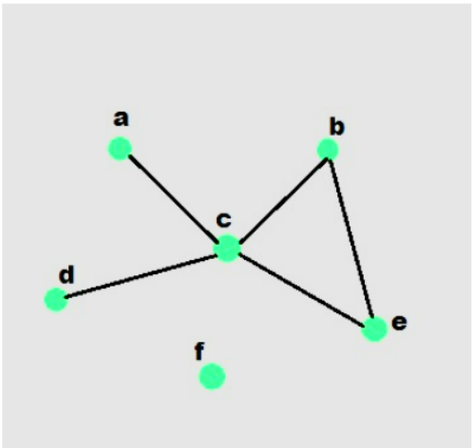

Figure 1. Sample graph with vertices and edges

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

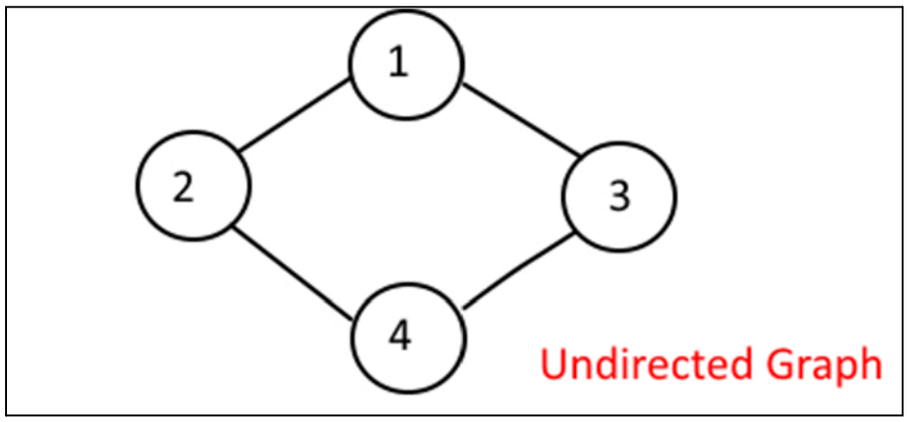

Figure 2. Undirected Graph

An undirected graph is a set of points that is also called as vertices that is connected by lines otherwise called as edges. The most feature is that all edges are two-way. For example, in figure 2 the edge between the vertex 1 and 2 can be traveled from 1 to 2 but also from 2 back to 1.

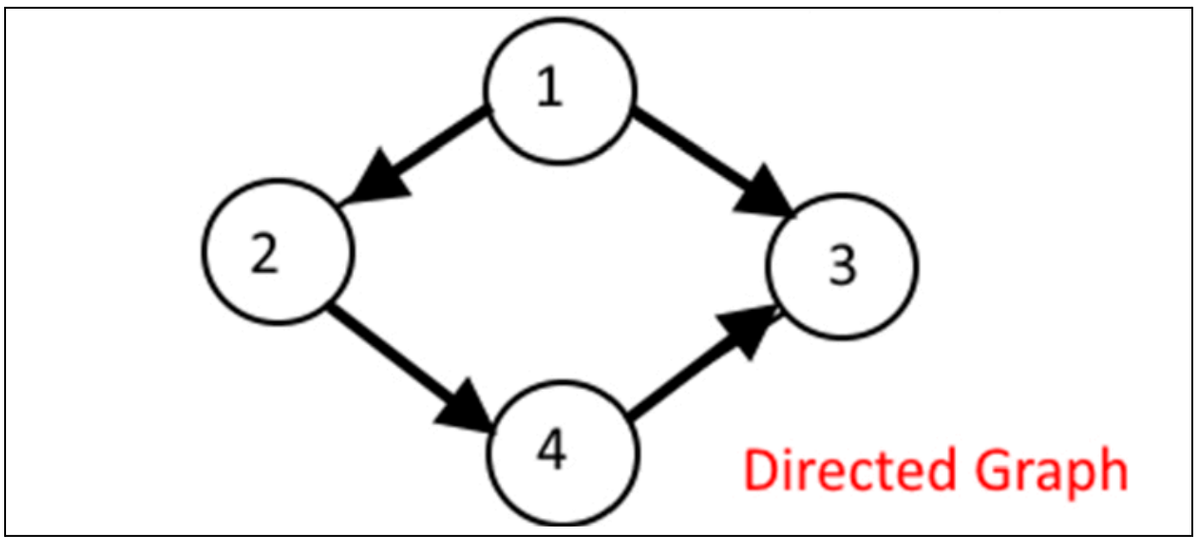

Figure 3. Directed Graph

A directed graph is also a collection of vertices connected by edges but this time it has a direction. The most feature is that all edges are one way. In Figure 3, the arrows on the edges only shows a direction of travel. For example, you can travel from vertex 1 to 2 but you can't go back to 1 from 2 since the arrow only points one way.

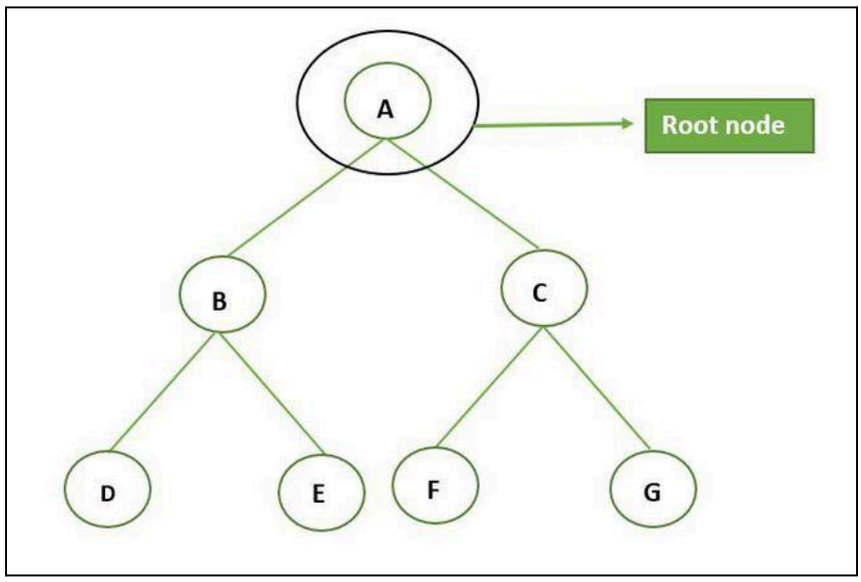

Figure 4. Nodes

Nodes are the foundation points that make up a graph. Figure 4. Shows a set of nodes from A through G. Each node represents individual items. In some graphs some have special roles like the Node A is specifically identified as the Root node

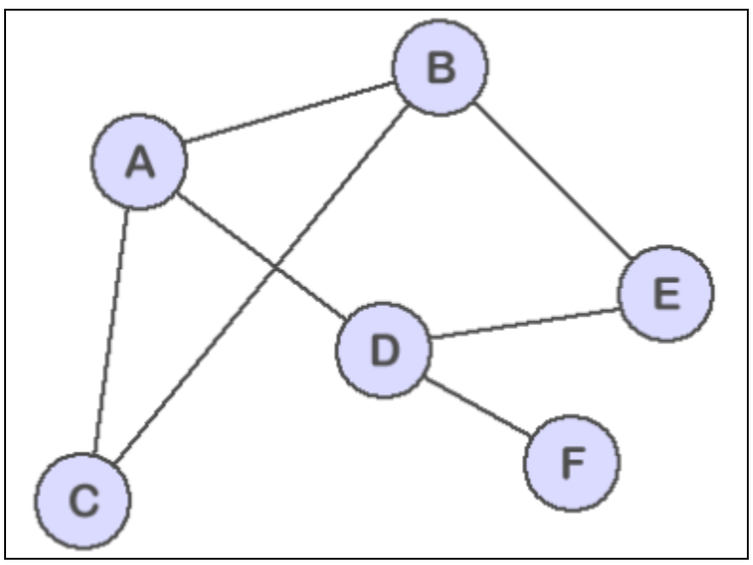

Figure 5. Vertex

The term Vertex is another term for a node. Its a point in the graph that represents an item. In Figure 5, the graph has six vertices which is point A, B, C, D, E, and F.

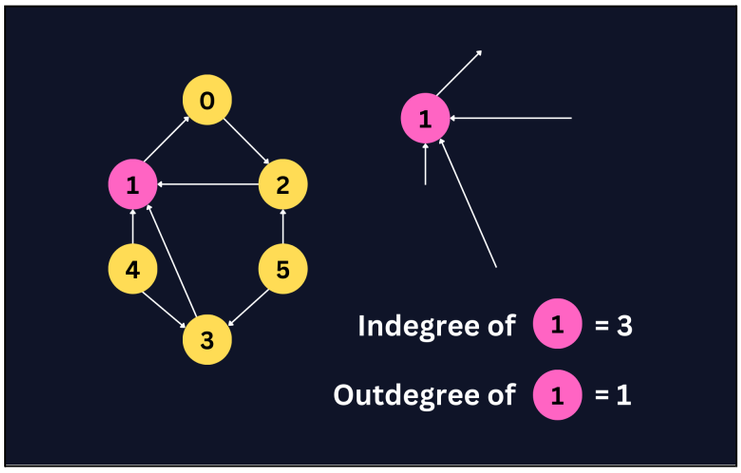

Figure 6. Degree, Indegree and Outdegree

The **degree** of a vertex is the total number of the edges that are connected to it. It is a count of all the connected, like both incoming and outgoing. In the example shown on Figure 6, the vertex 1 has 3 incoming edges and 1 outgoing edge, therefore the total degree is 4.

In a directed graph, the **indegree** in the vortex is the number of edges that points to it. On the Figure 6, it shows there are 3 edges pointing to one which is the item 0, 2, and 4 so the indegree of vertex 1 is 3.

Lastly, the outdegree of a vertex is the opposite of indegree, which in this time the we count it as the number of edges that are pointing away from it. As shown in Figure 6, there is only one edge pointing away from vertex 1. This means that the outdegree of vertex 1 is 1.

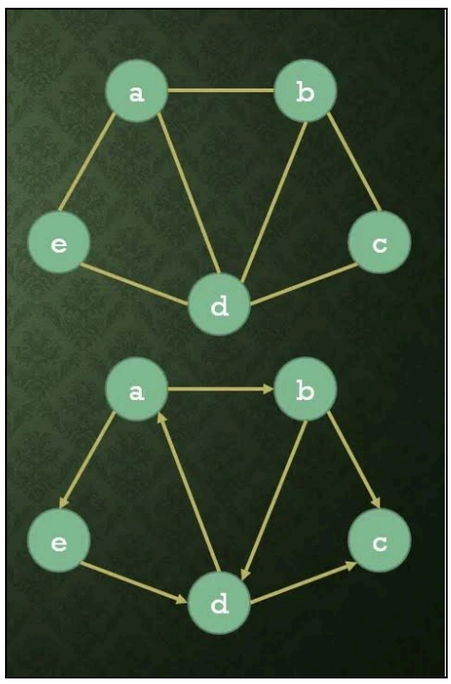

Figure 7. Path

A **Path** in a graph is a sequence of vertices connected by edges, like a road from start to an end point. Figure 7, shows a graph where it has many paths. For example, a path from vertex e to c can be traced as e - a - b - c. The length of this path is 3 because it passes three edges.

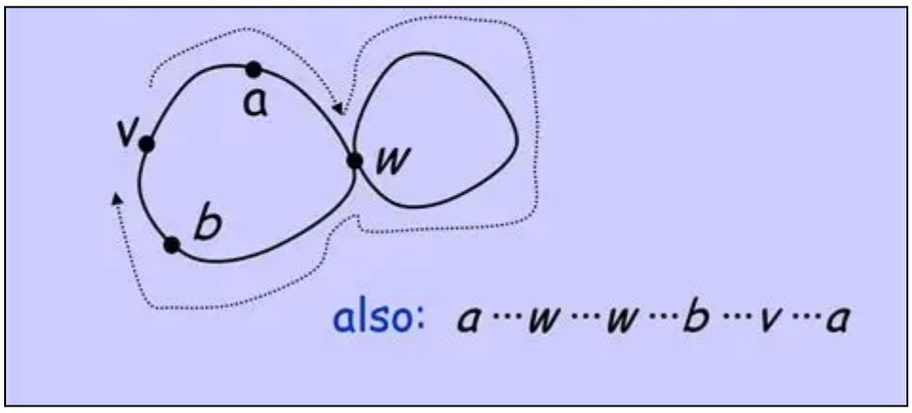

Figure 8. Cycle

A Cycle is a path in graph that starts on the point and ends with the same point, forming like a closed loop. The graph in Figure 8, shows an example of a cycle. A path starting at vertex a can travel through w, b, and v before returning to the starting vertex a which completes a cycle.

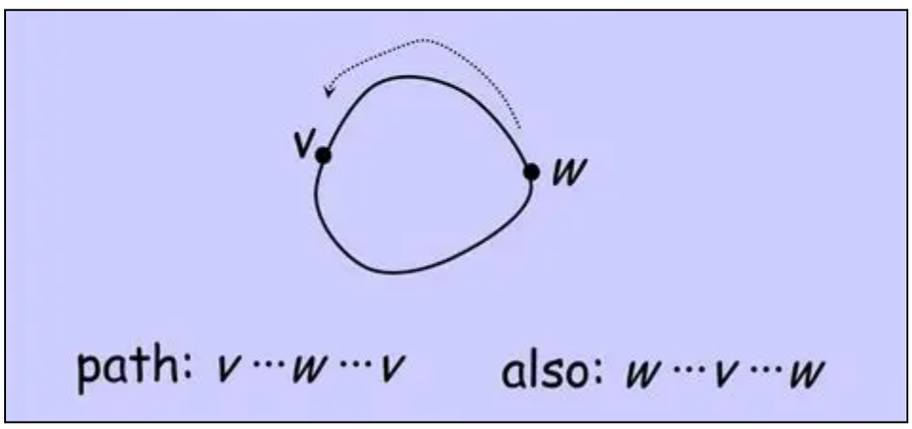

Figure 9. Simple Cycle

A Simple Cycle is a cycle where no vertices are repeated, with the exception for the starting and ending vertex. In Figure 9, it shows a basic example of the path that starts at vertex v and goes to w and returns directly to v. Because no other vertices are gone through, no vertices are repeated in the middle of the path so it is considered a simple cycle.

IV. Conclusion

In this laboratory report 10, I learned the basics of graphs by defining and looking at the figures and learning how it works for key terms like nodes, paths, cycles, etc. This helped me to understand the simple parts of graphs and how they connect. By finishing this laboratory report I now have a good foundation for understanding why this is essential and how it can be used in programming.

References

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

[2] Your-Name. (n.d.). *Graph in DSA*. Bagni Blog. https://bagniblog.blogspot.com/2018/01/graph-in-dsa.html

[3] Verma, A. (2024, January 6). *Introduction to graph Data Structure*. Innoskrit: Coder’s Obsession. https://learn.innoskrit.in/graph/introduction-to-graph-data-structure/

[4] Urooj Ainuddin. (2022, November 15). *DSA lecture 24* [Video]. YouTube. https://www.youtube.com/watch?v=sVPPGfTLk-8

[5] GeeksforGeeks. (2022, November 26). *What are the different types of Nodes in a Tree*. GeeksforGeeks. https://www.geeksforgeeks.org/dsa/what-are-the-different-types-of-nodes-in-a-tree/

[6] Cabello, J. G. (2018). A model towards global demographics: an application—a universal bank branch geolocator based on branch size. *Soft Computing*, *23*(16), 7193–7205. https://doi.org/10.1007/s00500-018-3362-z