Data Structure and Algorithm

Laboratory Activity No. 10

Intro to Graphs

|  |  |
| --- | --- |
| *Submitted by:* | *Instructor:* |
| LastName, FirstName MI. | Engr. Maria Rizette H. Sayo |

Month, DD, YYYY

# 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.

A diagram of a triangle with green dots

AI-generated content may be incorrect.

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

# Methods

* 1. 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

# Results

* + 1. *An* ***undirected graph*** *is a type of* [*graph*](https://www.geeksforgeeks.org/dsa/graph-data-structure-and-algorithms/) *where the edges have no specified direction assigned to the them.*
    2. *is defined as a type of* [*graph*](https://www.geeksforgeeks.org/dsa/introduction-to-graphs-data-structure-and-algorithm-tutorials/) *where the edges have a direction associated with them.*
    3. Nodes are a basic data structure which contain data and one or more links to other nodes.
    4. ertices are the fundamental units of the graph. Sometimes, vertices are also known as vertex or nodes. Every node/vertex can be labeled or unlabelled.
    5. A sequence of vertices where each adjacent pair is connected by an edge, representing a route through the graph.
    6. A path that starts and ends at the same vertex, with no repeated edges.
    7. A cycle that repeats only the first/last vertex, with all other vertices and edges being distinct.
    8. A sequence of vertices where each adjacent pair is connected by an edge, representing a route through the graph.
    9. The graph in which the graph is a cycle in itself, the minimum value of degree of each vertex is 2.
    10. A cycle that repeats only the first/last vertex, with all other vertices and edges being distinct.

# Conclusion

Graphs are fundamental data structures composed of nodes (vertices) and edges that model relationships and connections. Key distinctions include undirected graphs with bidirectional relationships versus directed graphs with one-way connections. Critical properties like degree,indegree, and outdegree quantify connectivity, while paths and cycles describe traversal patterns. Understanding these core concepts provides the foundation for analyzing network structures, solving routing problems, and modeling complex systems across computer science and real-world applications.

**References**

[1] GeeksforGeeks. (2025, July 23). *What is Undirected Graph? | Undirected Graph meaning*. GeeksforGeeks. <https://www.geeksforgeeks.org/dsa/what-is-unidrected-graph-undirected-graph-meaning/>

[2] GeeksforGeeks. (2025a, July 23). *What is Directed Graph? | Directed Graph meaning*. GeeksforGeeks. <https://www.geeksforgeeks.org/dsa/what-is-directed-graph-directed-graph-meaning/>

[3]*CS102: Data Structures and Algorithms: Nodes Cheatsheet | Codecademy*. (n.d.). Codecademy. <https://www.codecademy.com/learn/cspath-cs-102/modules/nodes/cheatsheet>

[4]GeeksforGeeks. (2025a, July 15). *Introduction to graph Data Structure*. GeeksforGeeks. https://www.geeksforgeeks.org/dsa/introduction-to-graphs-data-structure-and-algorithm-tutorials/