**National University of Computer & Emerging Sciences (NUCES) Islamabad,** Department of Computer Science

DATA STRUCTURES – FALL 2023 LAB 13

Learning Outcomes

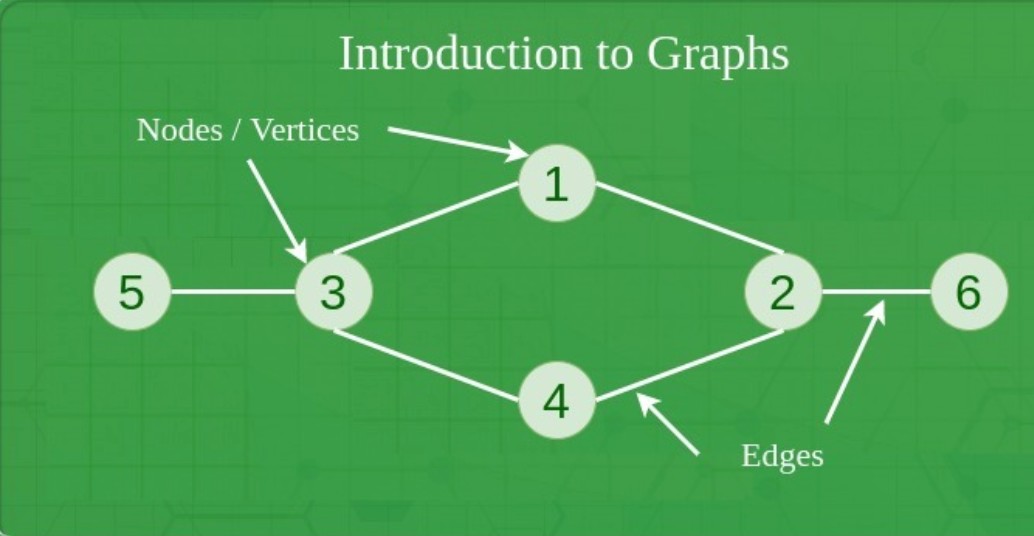
In this lab, you will implement the Graph data structure.

# Graph ADT

A Graph is a non-linear data structure consisting of vertices and edges. The vertices are sometimes also referred to as nodes and the edges are lines or arcs that connect any two nodes in the graph. More formally a Graph is composed of a set of vertices (V) and a set of edges (E). The graph is denoted by G (E, V).

# Structure of Graph

* **Vertices:** Vertices are the fundamental units of the graph. Sometimes, vertices are also known as vertex or nodes. Every node/vertex can be labeled or unlabelled.
* **Edges:** Edges are drawn or used to connect two nodes of the graph. It can be ordered pair of nodes in a directed graph. Edges can connect any two nodes in any possible way. There are no rules. Sometimes, edges are also known as arcs. Every edge can be labeled/unlabelled.



## Task 1 – Implementing the Graph ADT. (10 min)

To implement the graph ADT, create an adjacency list. To do this, create a structure having an integer (representing vertex of the graph) and a linked list object (representing all edges from that vertex to other vertices). Now, create an array of this structure. This array is an adjacency list.

## Task 2 – Adding vertices to a graph (20 min)

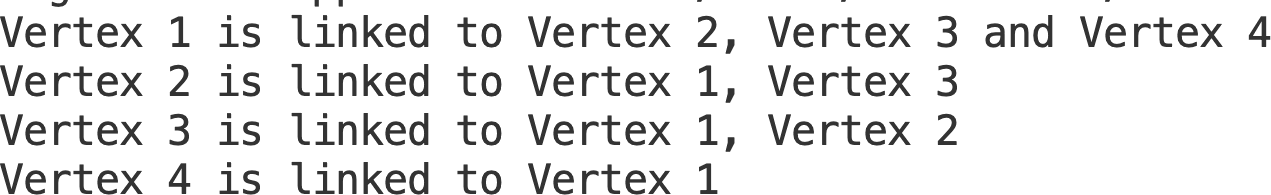
Write a code that adds vertices along with their edges to the adjacency list.

## Task 3 – Implementing an undirected graph (20 min)

Modify task 2 to implement an undirected graph such that if an edge is created between vertex a and vertex b, an edge between vertex b and vertex a is automatically created.

## Task 4 – Print the graph (30 min)

Write a code that prints the graph. The output should be a vertex and all the edges of that vertex. A sample output is shown below:



## Task 5 – Implement a Weighted graph (30 min)

Modify tasks 1 and tasks 2 to implement a weighted graph (A graph that in which each edge has a weight/cost in going from one vertex to another).

## Note: Your program should be menu-based program that allows the user to select and perform each task independently and provide the necessary input and output as specified in the tasks.