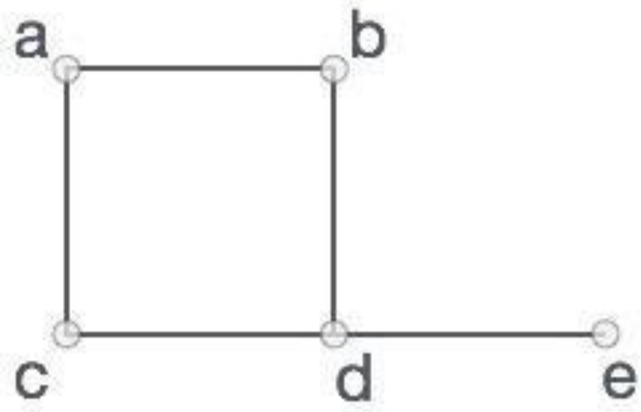
GRAPH
(CAT-201)

Design By:
Ms. Gurpreet kaur dhiman
Ms.Mandeep kaur
Chandigarh University-Gharuan

GRAPH

- A graph is a pictorial representation of a set of objects where some pairs of objects are connected by links. The interconnected objects are represented by points termed as **vertices**, and the links that connect the vertices are called **edges**.
- Formally, a graph is a pair of sets **(V, E)**, where **V** is the set of vertices and **E** is the set of edges, connecting the pairs of vertices. Take a look at the following graph



https://www.tutorialspoint.com/data_structures_algorithms/graph_data_structure.htm

GRAPH

- **Vertex** − Each node of the graph is represented as a vertex. In the following example, the labeled circle represents vertices. Thus, A to G are vertices. We can represent them using an array as shown in the following image. Here A can be identified by index 0. B can be identified using index 1 and so on.

- **Edge** − Edge represents a path between two vertices or a line between two vertices. In the following example, the lines from A to B, B to C, and so on represents edges. We can use a two-dimensional array to represent an array as shown in the following image. Here AB can be represented as 1 at row 0, column 1, BC as 1 at row 1, column 2 and so on, keeping other combinations as 0.

- **Adjacency** − Two node or vertices are adjacent if they are connected to each other through an edge. In the following example, B is adjacent to A, C is adjacent to B, and so on.

- − Path represents a sequence of edges between the two vertices. In the following example, ABCD represents a path from A to D.

GRAPH OPERATION

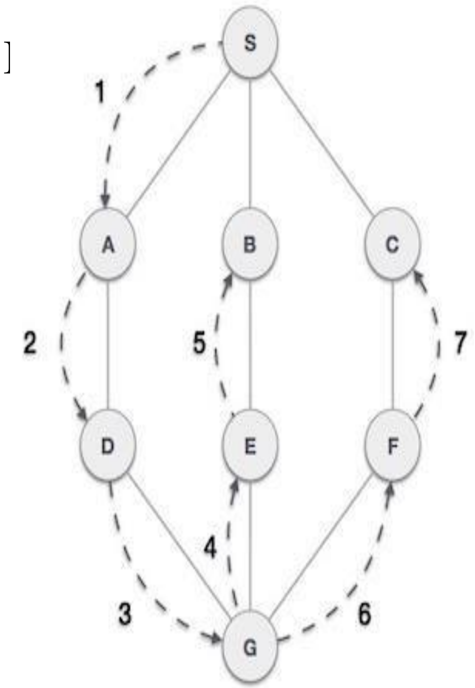
Following are basic primary operations of a Graph −

- **Add Vertex** − Adds a vertex to the graph.
- **Add Edge** − Adds an edge between the two vertices of the graph.
- **Display Vertex** − Displays a vertex of the graph.

GRAPH TRAVERSAL USING DFS

Depth First Search (DFS) algorithm traverses a graph in a depth ward motion and uses a stack to remember to get the next vertex to start a search, when a dead end occurs in any iteration.

- As in the example given above, DFS algorithm traverses from A to B to C to D first then to E, then to F and lastly to G. It employs the following rules.



- **Rule 1** − Visit the adjacent unvisited vertex. Mark it as visited. Display it. Push it in a stack.

- **Rule 2** − If no adjacent vertex is found, pop up a vertex from the stack. (It will pop up all the vertices from the stack, which do not have adjacent vertices.)

- **Rule 3** − Repeat Rule 1 and Rule 2 until the stack is empty.

GRAPH TRAVERSAL USING BFS

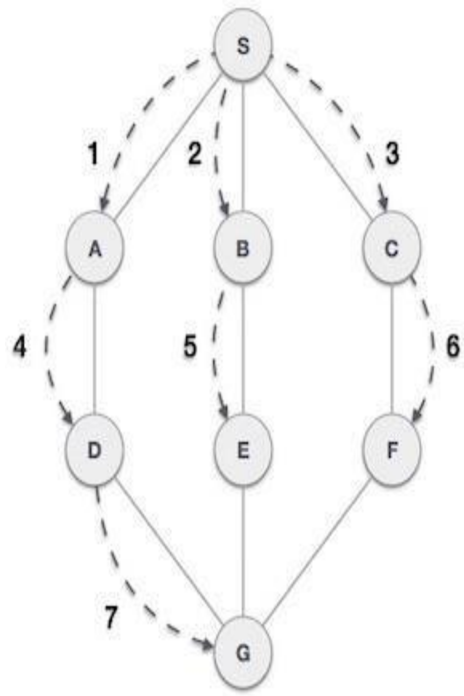
Breadth First Search (BFS) algorithm traverses a graph in a breadth ward motion and uses a queue to remember to get the next vertex to start a search, when a dead end occurs in any iteration.

- As in the example given above, BFS algorithm traverses from A to B to E to F first then to C and G lastly to D. It employs the following rules.

- **Rule 1 −** Visit the adjacent unvisited vertex. Mark it as visited. Display it. Insert it in a queue.

- **Rule 2 −** If no adjacent vertex is found, remove the first vertex from the queue.

- **Rule 3 −** Repeat Rule 1 and Rule 2 until the queue is empty.



Adjacency Matrix

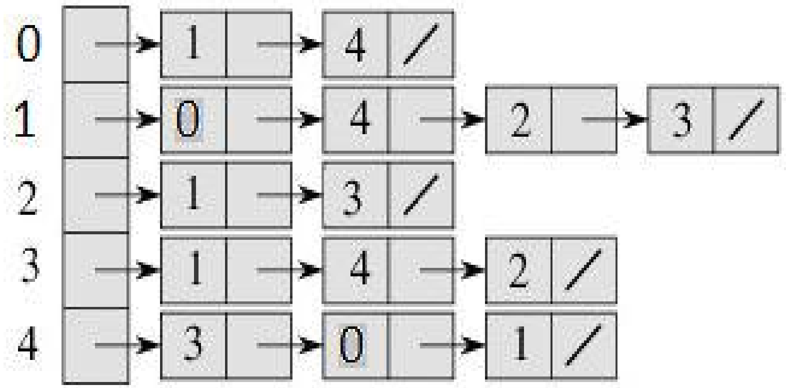
Let G=(V,E) be a graph with n vertices.

- The adjacency matrix of G is a two-dimensional n by n array, say adj_mat
- If the edge (vi, vj) is in E(G), adj_mat[i][j]=1
- If there is no such edge in E(G), adj_mat[i][j]=0
- The adjacency matrix for an undirected graph is symmetric; the adjacency matrix for a digraph need not be symmetric

	0	1	2	3	4
0	0	1	0	0	1
1	1	0	1	1	1
2	0	1	0	1	0
3	0	1	1	0	1
4	1	1	0	1	0

Adjacency List

An array of linked lists is used. Size of the array is equal to number of vertices. Let the array be array[i]. An entry array[i] represents the linked list of vertices adjacent to the ***i***th vertex. This representation can also be used to represent a weighted graph. The weights of edges can be stored in nodes of linked lists. Following is adjacency list representation of the above graph.



https://www.tutorialspoint.com/data_structures_algorithms/graph_data_structure.htm

FAQ

- How BFS and DFS are different?
- Define adjacency matrix?
- How graph is represented in the form list?

Bibliography

- Seymour Lipschutz, Schaum's Outlines Series Data structures TMH
- Introduction to Data Structures Applications, Trembley&Soreson, Second Edition, Pearson Education
- www.geeksforgeeks.org/analysis-of-algorithms-set-3asymptotic-notations/
- www.tutorialspoint.com/data_structures_algorithms/asymptotic_analysis.htm

Thank You