GRAPH DATA STRUCTURE

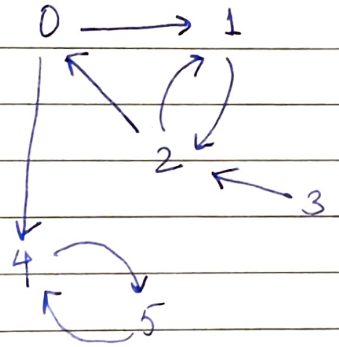
- vertices are linked by edges
- Nodes → vertices
- weights of edges

A graph G is defined by 2 sets :- V : set of vertices
E : set of edges
G(V, E)

No. of vertices $n = |V|$
No. of edges $m = |E|$

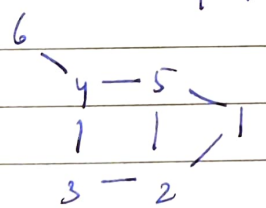
Vertices are always numbered $1, \ldots, n$ OR $0, \ldots, n-1$

Directed Graph → Each edge has got a direction



$V = \{0, 1, 2, 3, 4, 5\}$
$E = \{(0,1), (0,4), (1,2), (2,0), (2,1), (3,2), (4,5), (5,4)\}$

Undirected Graph



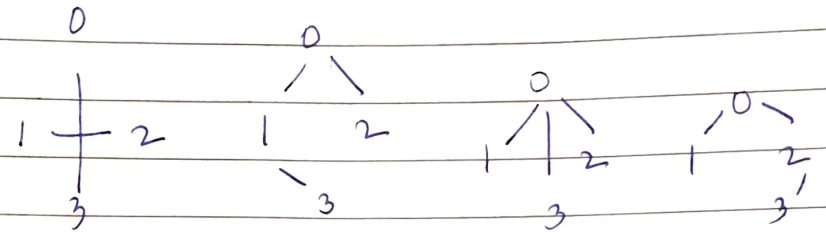
$V = \{1, 2, 3, 4, 5, 6\}$
$E = \{(1,2), (1,5), (2,5), (2,3), (3,4), (4,5), (4,6)\}$

Cycle:
A path whose start and end vertices are same, and no intermediate vertex gets repeated.

Tree: connected graph w/o cycles

Spanning Tree: Tree formed of graph edges which connect all the vertices of the graph.

Eg.

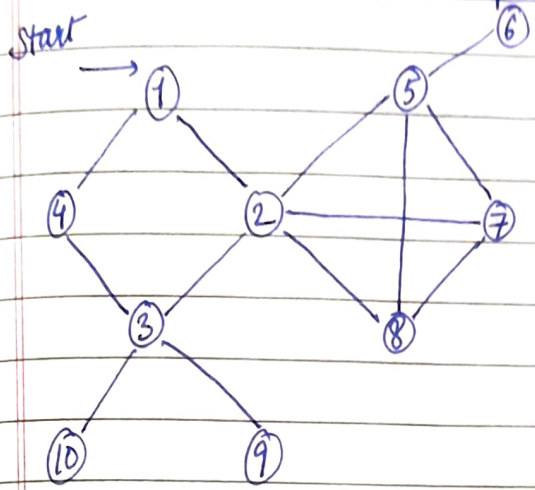


Complete Graph: Every vertex is having an edge to all other vertices.

2 most Common Representation of Graphs:- ① Adjacency Matrix
② Adjacency List

Refer lec 23 : slide 6-10

Breadth First Traversal for a Graph (queue is used)



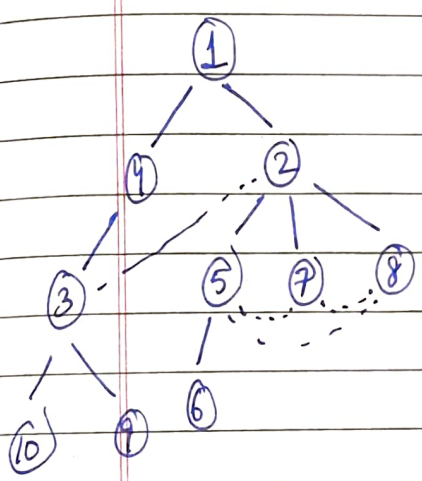
(eg.)

* We can start from anywhere
* We can visit adjacent vertices in any order when exploring a vertex

BFS: 1, 4, 2, 3, 5, 7, 8, 10, 9, 6

Q | 1 | 4 | 2 | 3 | 5 | 7 | 8 | 10 | 9 | 6 |

BFS Spanning Tree

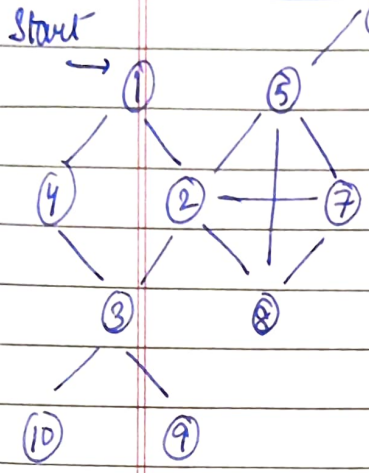


dotted lines → cross edges

* When at a vertex, explore it completely before exploring an adjacent vertex.
* Next vertex for exploration should be selected from queue only

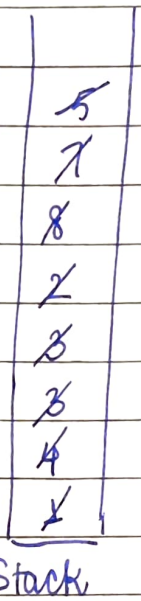
NOTE: There can be multiple correct answers

Depth First Traversal for a Graph (Stack is used)

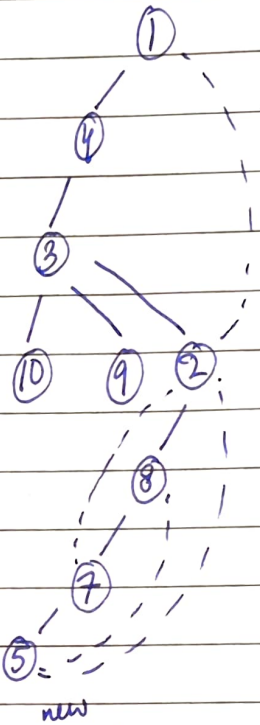


* We can start from any vertex

DFS: 1, 4, 3, 10, 9, 2, 8, 7, 5, 6



DFS Spanning Tree



← Dotted lines: back edges

* Once you have visited a new vertex, explore it first and suspend the old vertex in the stack.
* When There is no adjacent value of a vertex, go back by looking at the stack.