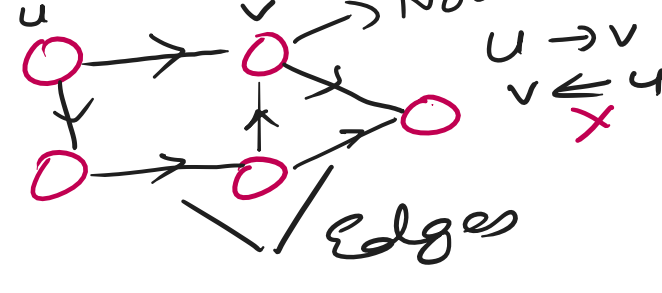


Introduction to Graphs:

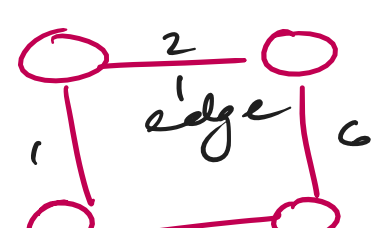
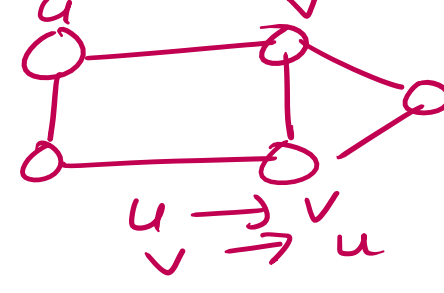
It is a non-linear data structure containing an entity called Node which are connected to each other by Edges. Google Maps

Depending on the connectivity of nodes, graphs are of two types: \rightarrow

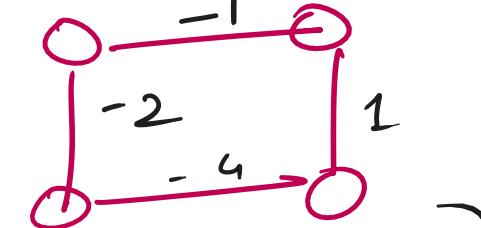
Directed



Undirected



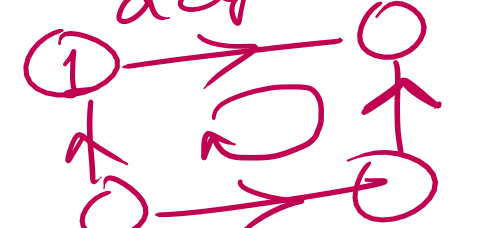
Weighted



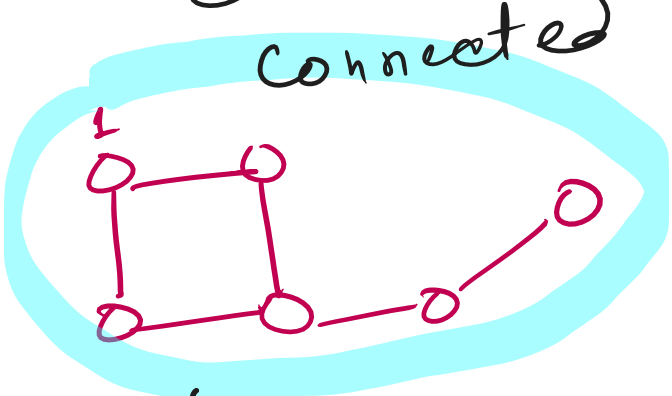
Unweighted



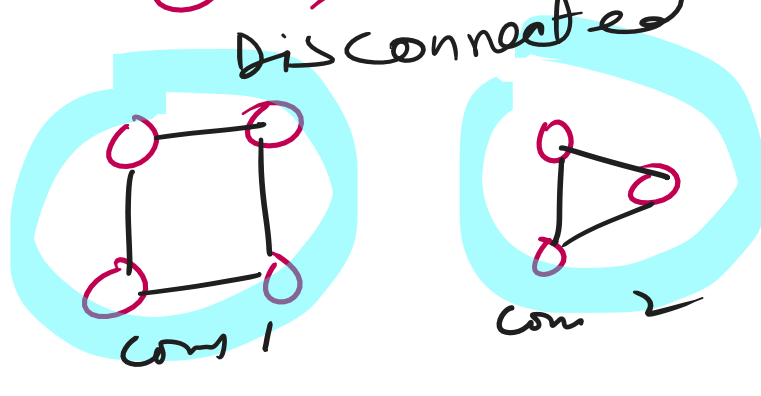
cyclic



acyclic

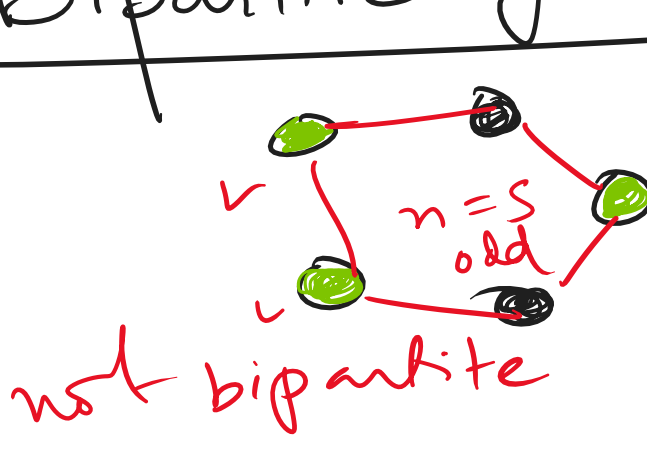


connected

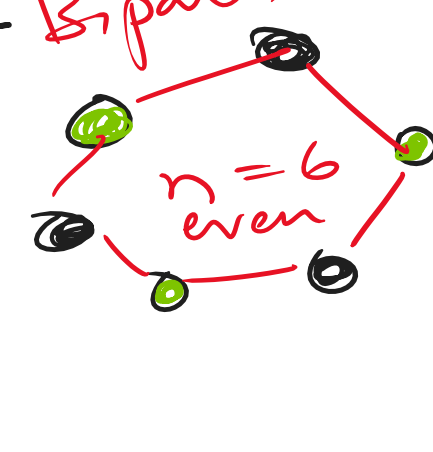


disconnected

Bipartite Graph:



not bipartite



bipartite

Graph Representation Interview Question

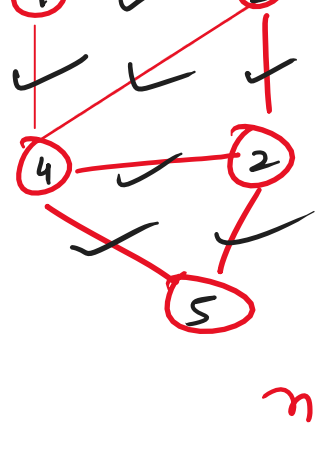
* Given a graph (undirected), represent the given graph using

(i) Adjacency matrix

(ii) Adjacency list

Matrix

	1	2	3	4	5
1	0	0	1	1	0
2	0	0	1	1	1
3	1	1	0	1	0
4	1	1	1	0	1
5	0	1	0	1	0



List

Node	Neighbours
1	3, 4
2	3, 4, 5
3	1, 2, 4
4	1, 2, 3, 5
5	2, 4

$n = 5$ $e = 7$



Adj list

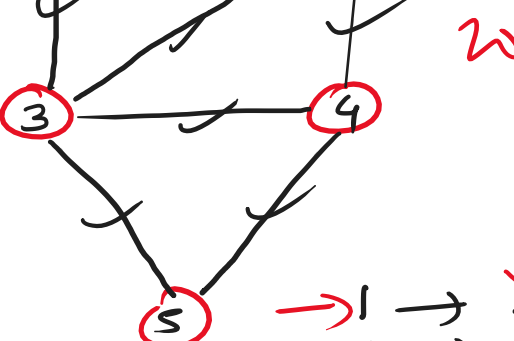
1	2
2	1, 3
3	1, 2

Graph Traversal Questions:

BFS Traversal:

Level Order Traversal

start node: 1



O/p: 1, 2, 3, 4, 5

who are your neighbours?

$E \log V$

adj list:

1	2, 3
2	1, 3, 4
3	1, 2, 4, 5
4	2, 3, 5
5	3, 4

visited

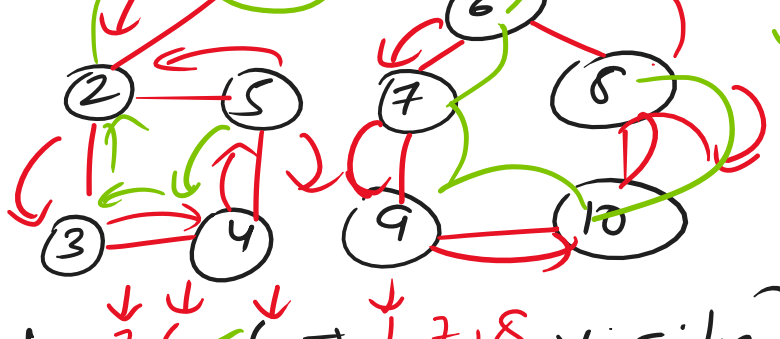
1	T
2	T
3	T
4	T
5	T

queue

1
2
3
4
5

Depth First Traversal: DFS

starting node = 1



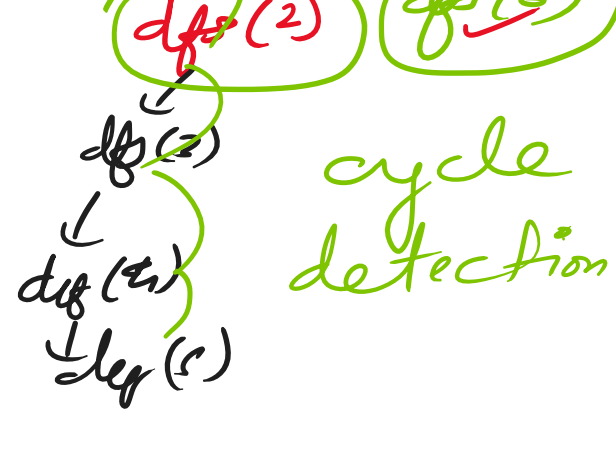
Recursion

Intuition?

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

visited

1	T
2	T
3	T
4	T
5	T
6	T
7	T
8	T
9	T
10	T



cycle detection

* Graph Algos:

positive weights

Dijkstra's Algorithm (shortest distance)

Prim's Algorithm (MST)

Minimum Spanning Tree

Kruskal's Algorithm

Bellman Ford

Floyd Warshall

Topological Sort \rightarrow Disjoint Set

DP

Recursion Tree \rightarrow DP

0, 1, 1, 2, 3, 5, 8, 13, 21, ...

0 1 2 3 4 5 6 7 8

$f(0) = 0$ $f(n) = f(n-1) + f(n-2)$

$f(1) = 1$

dp =

overlapping

subproblems

store \rightarrow Tabulation (array)

Rec + Mem + Tab

Space Optim

$S \subset O(1)$