GRAPH Series

DAY - 01

GoodLuck | Page No. 1

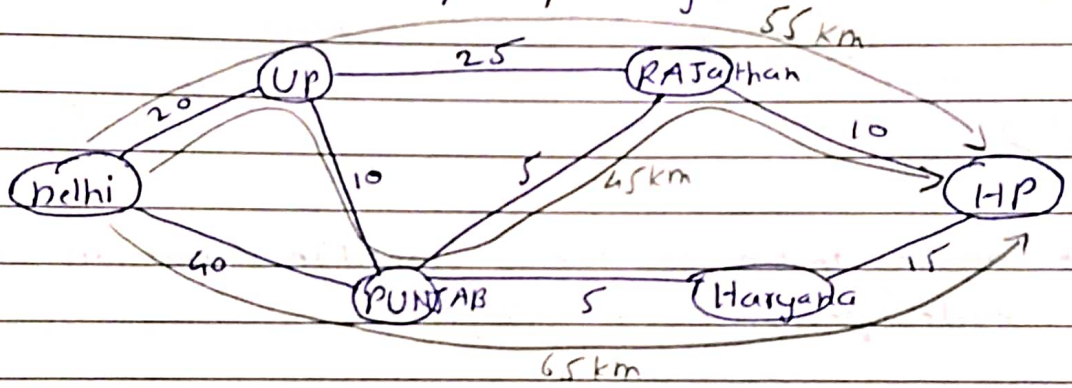
Date

* Graph Data Structure and real world examples on Graph:

I] INTRODUCTION TO GRAPH :-

- let's take a simple example:
- Priyanka lives in Delhi and she wants to meet her friend who lives in Himachal Pradesh. the shortest root to reach HP from Delhi will be ?

- We have the map represing roots as:



- We have so many paths but the shortest will be to follow:

(D) $\xrightarrow{20}$ UP $\xrightarrow{25}$ RAJasthan $\xrightarrow{10}$ (HP) $\Rightarrow$ 55 km

(D) $\xrightarrow{20}$ UP $\xrightarrow{10}$ PUNJab $\xrightarrow{5}$ RAJasthan $\xrightarrow{10}$ (HP) $\Rightarrow$ 45 km

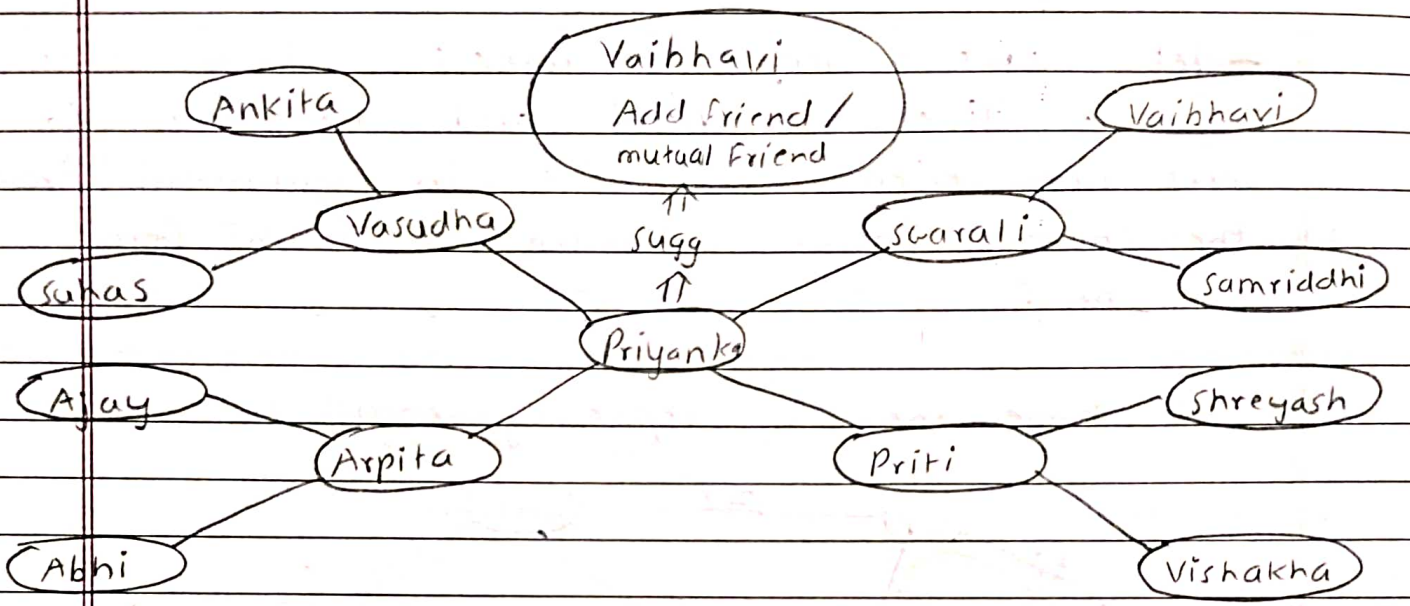
- Same way, we will find the shortest path roots afor each state shown in above map so that next time, to visit a friend from any of the states, we don't need to find the paths again
- Hence,

i] Find shortest path from (D) → (HP)

ii] Find shortest paths for all states from (D)

- Example - 2 :-

- Network in LinkedIn / Facebook / Social media apps:



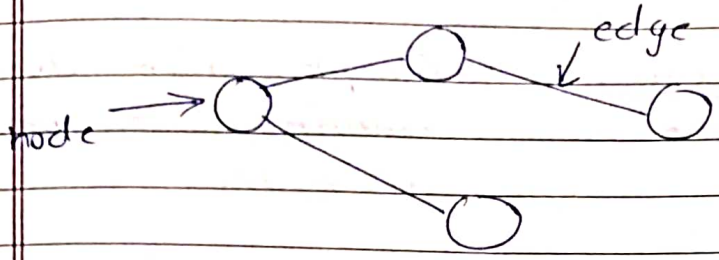
- Above shown is the LinkedIn network of node / person / user Priyanka

- LinkedIn gives us the suggestions as -
  "You may know Vaibhavi, add them to your n/w"
  Such suggestions are generated using the graph concept.

- In n/w, each user is connecting to his/her first degree connections and gets the suggestions of the mutual friends connection

GoodLuck | Page No.
Date

* GRAPH :



- Graph is a non linear data structure consists of vertices and edges

linear DS
- Array
- Linked List
- Queue
- Stack

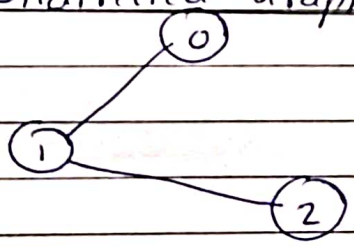
traversable in one line

Non-linear DS
- Tree
- Graph

non traversable in linear single time

* TYPES OF GRAPH :-

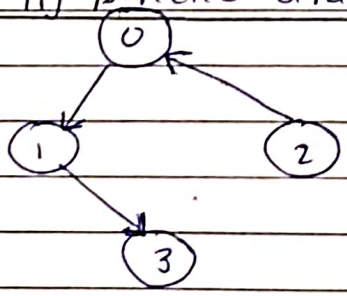
i] Undirected Graph:



- examples:

i] To represent Friends / connections on social media apps i.e. two friends are connected to each other

ii] Directed Graph:



- examples:

i] To represent Friends following each other in snapchat / Fb / IG

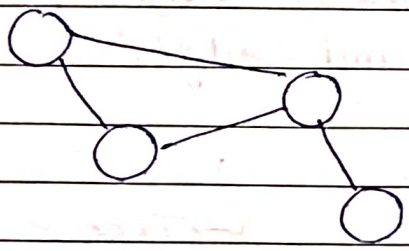
ii] To represent path between different cities

GoodLuck | Page No.
Date

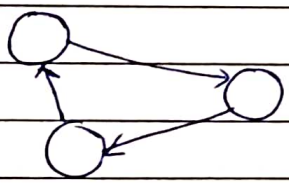
iii] Cyclic Graph

- Graph which contains atleast one cycle in it

- example:



(undirected cyclic)

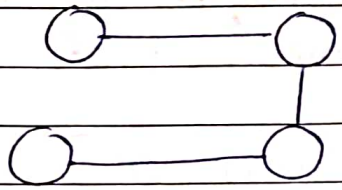


(directed cyclic)

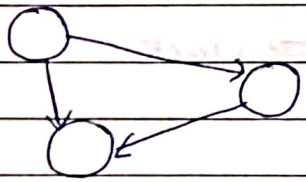
iv] Acyclic. Graph

- Graph which doesn't contain cycles between its nodes

- example:



(undirected acyclic)



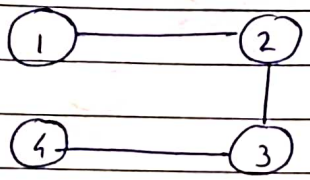
(directed acyclic)

GoodLuck | Page No.
Date

v] Connected Graph

- each node can be traversed from any of the node in graph.
- every node is reachable from every node.
- example:

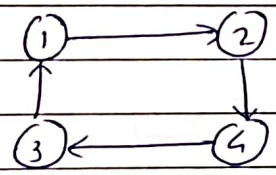
i)



can reach

i) 1 From 4 : 4-3-2-1
ii) 2 From 4 : 4-3-2
iii) 2 From 1 : 1-2

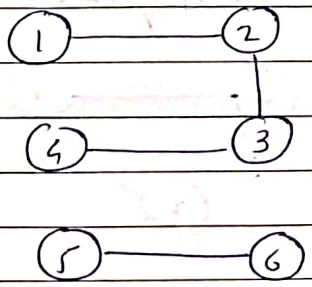
ii)



(directed)

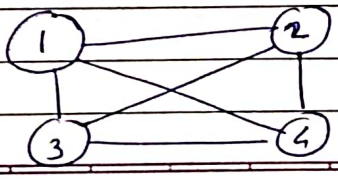
vi] Disconnected Graph

- not every node is reachable from each node of the graph
- example:



vii] Complete Graph:

- every node has an edge directly to every other node
- example:



→ no. of edges in a complete graph are always:

$$\frac{n * (n-1)}{2}$$

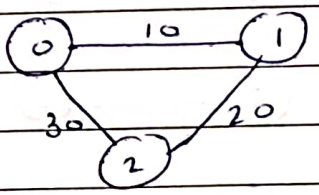
, n = no. of vertices

→ because every node is connected every other node i.e. rest (n-1) nodes

viii] Weighted Graph

- Edges have a weight or value or cost assigned to them between the nodes/ vertices.

- example:



ix] Unweighted Graph

- Edges don't have weights assigned to them

- example: