CS & IT ENGINEERING

Data Structure

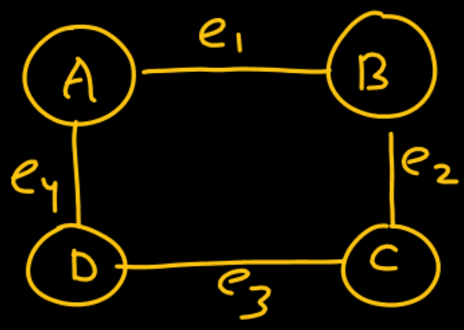
Graphs
In One Shot

By- Pankaj Sharma sir

Graphs

Non-linear data structure :

Graph



G(V, E)

set of vertices

set of Edges

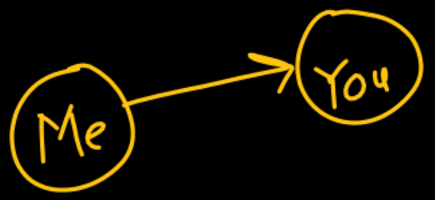
V = {A, B, C, D}

E = {e1, e2, e3, e4}

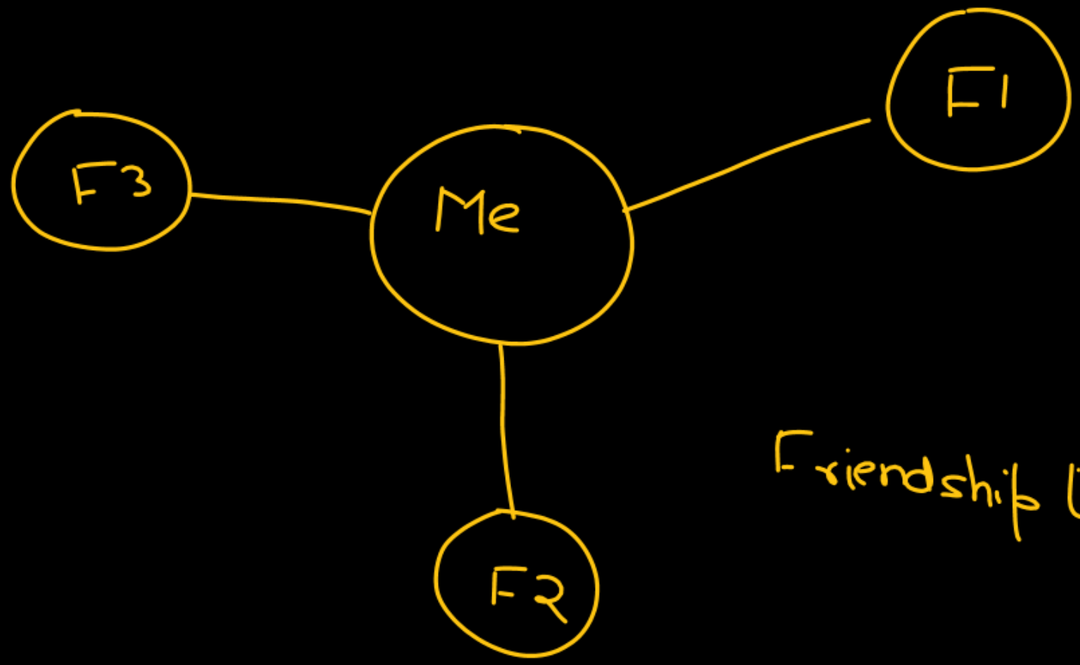
CS

WWW

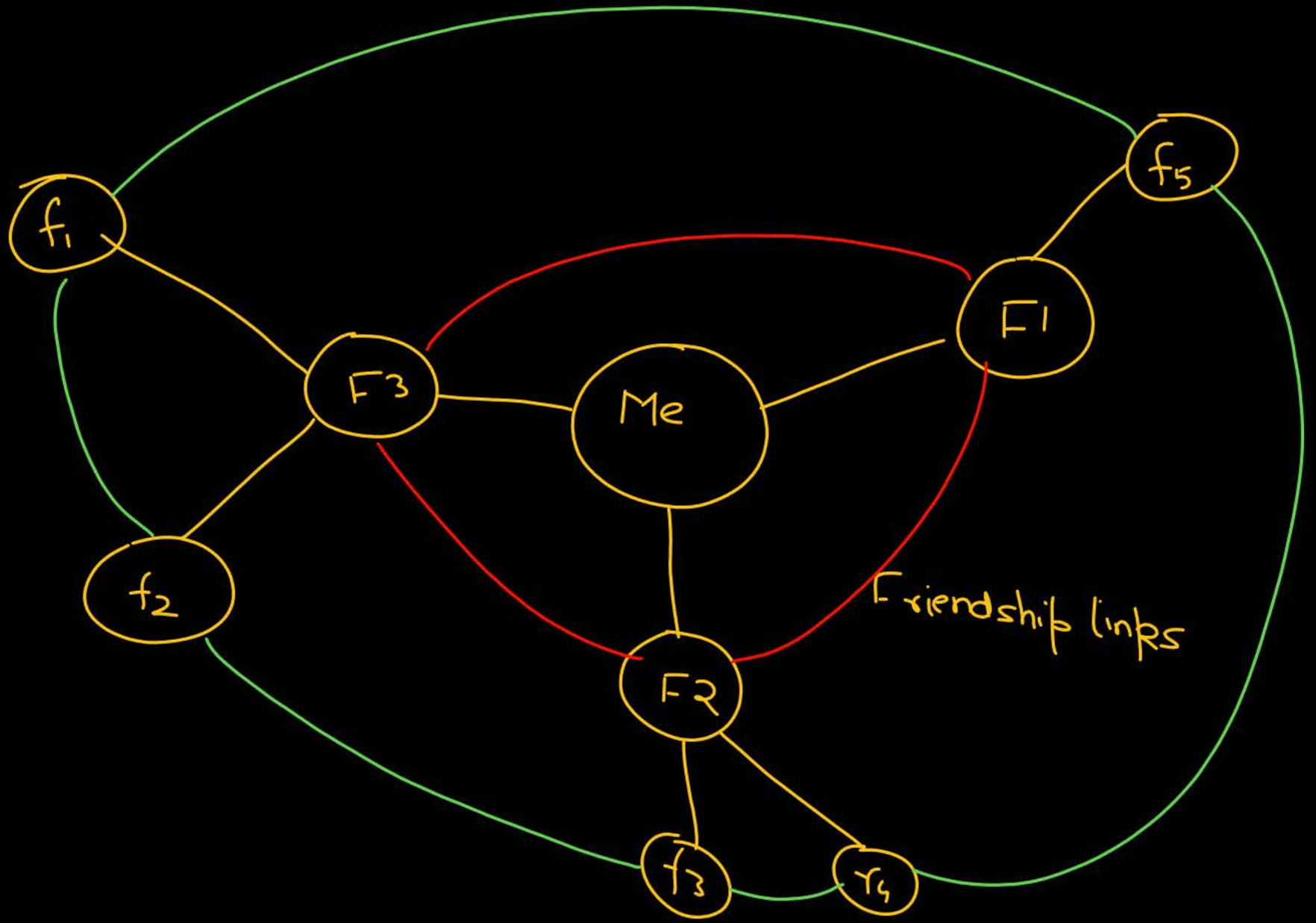
Social N/w

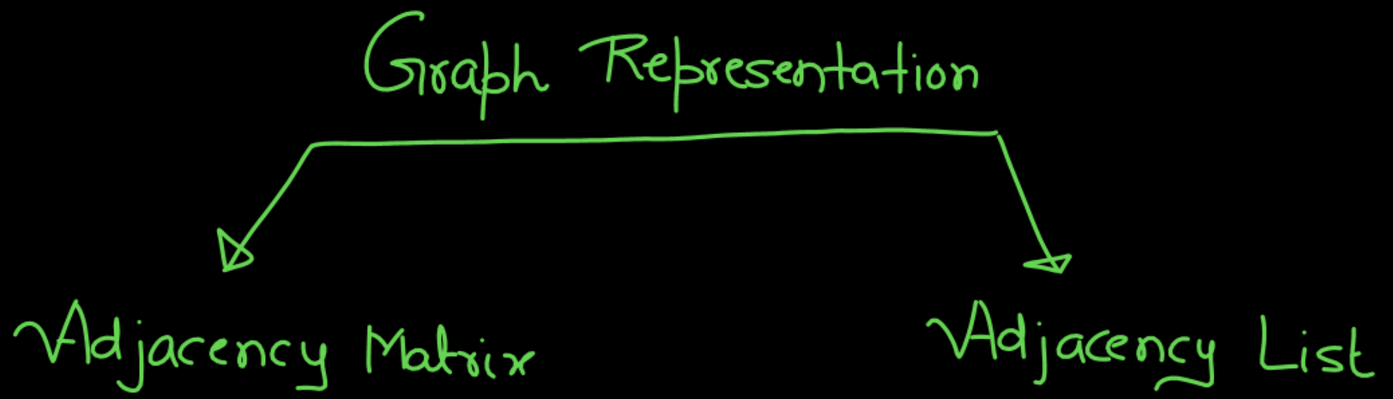


FB

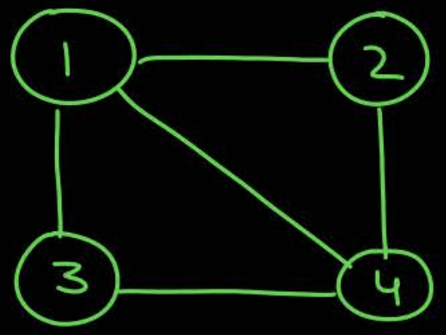


Friendship links


f1
f5
F1
F3
Me
f2
F2
f3
f4
Friendship links


Graph Representation
Adjacency Matrix
Adjacency List

$n = 4$

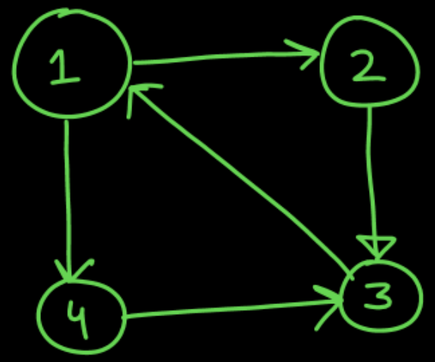


$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

$n \times n$ Matrix
in which Each entry
is either 0 or 1

Undirected
↳ $2|E|$

$A_{ij} = 1$ if node i is adjacent to node j
$= 0$, otherwise



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

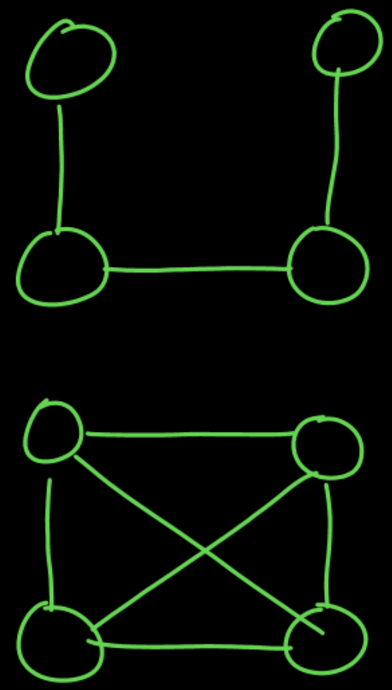
Adj. Matrix

10^{10} ⇒ users

$10^{10} \times 10^{10}$

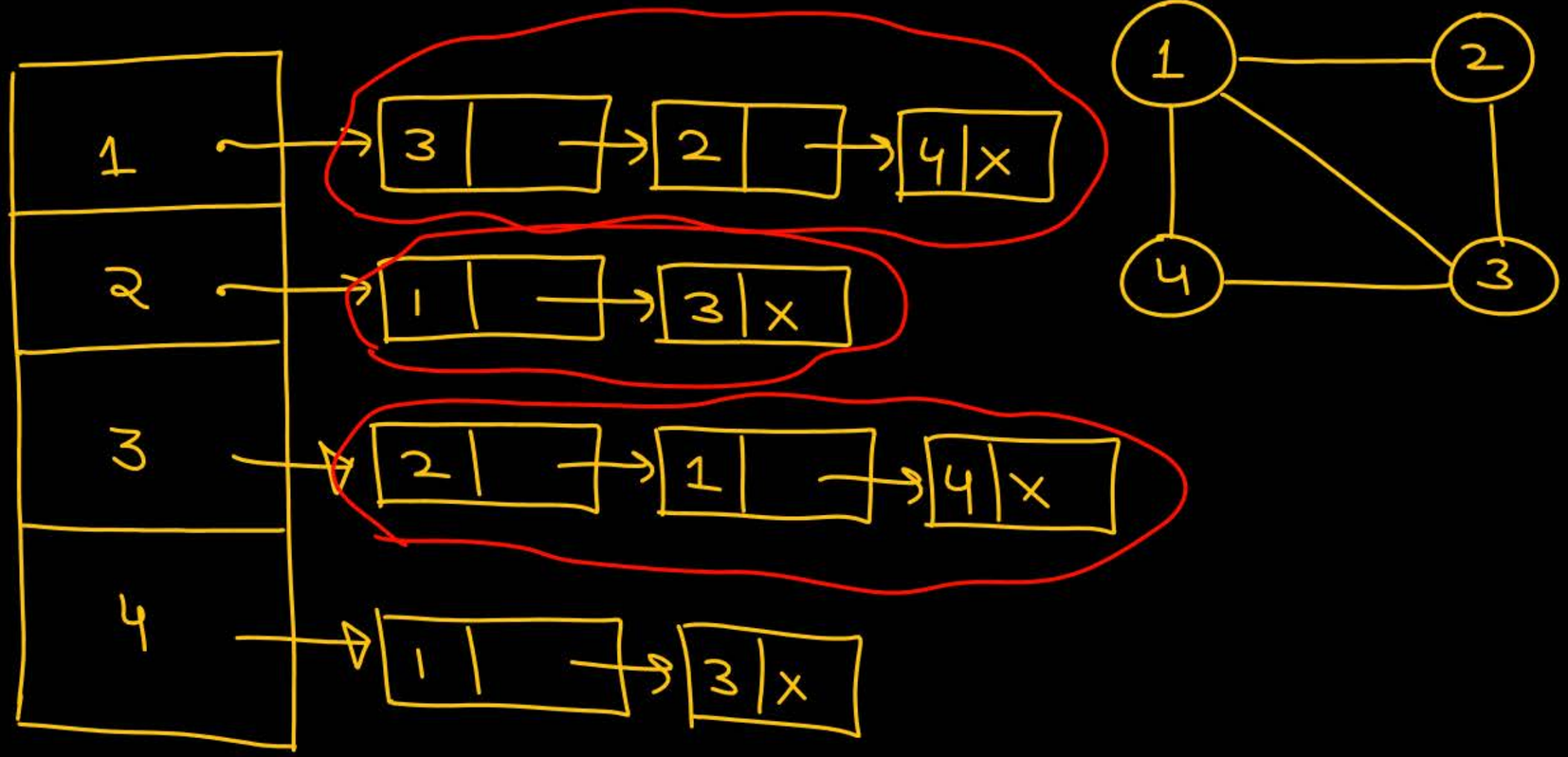
Sparse

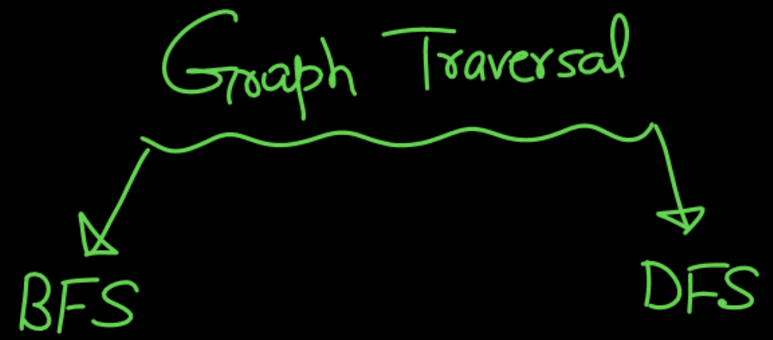
Entry 1



C++ ⇒ array of vector

Adjacency List



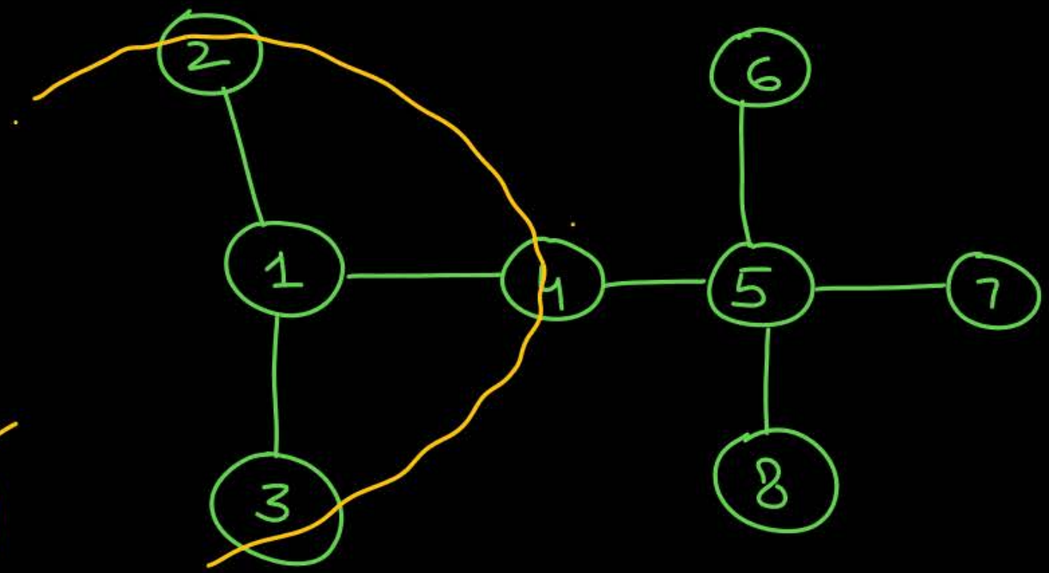


visiting a node :

Exploring a node :

Queue

BFS



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

1 2 3 4 5 6 7 8

1 distance

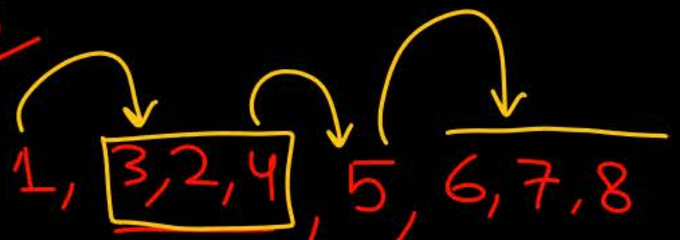
2 distance

3 distance

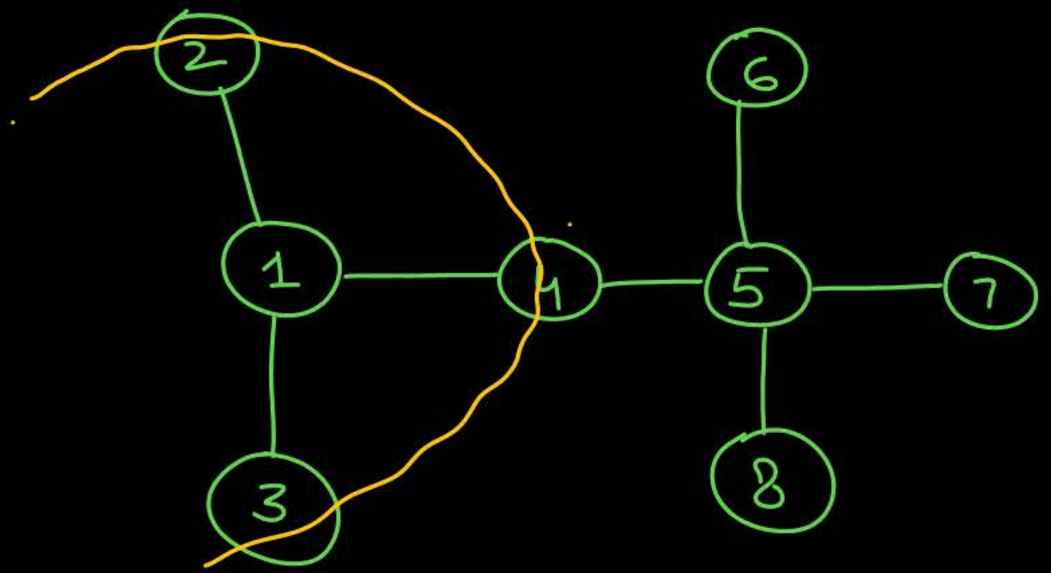
Queue

BFS

1, 4, 5



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



1 2 3 4 5 6 7 8

1 distance

2 distance

3 distance

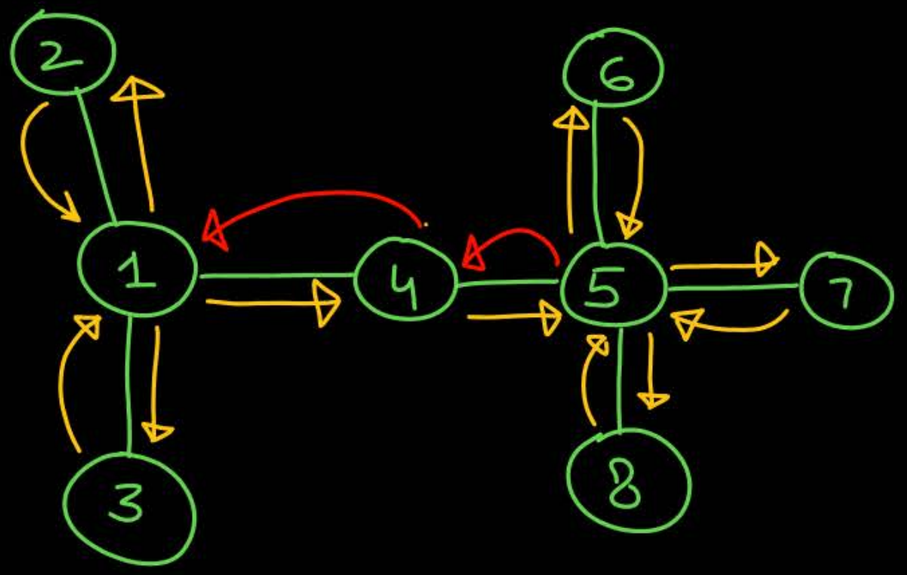
1, 2

Explore start | Explore start

Explore

Stop

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



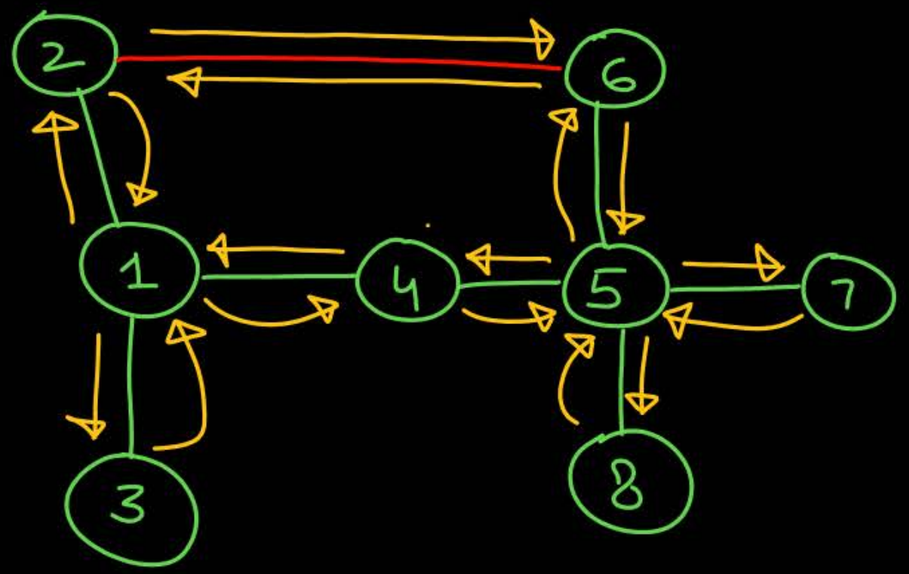
1,2,6,5,7,8,4,1,3

Source node

① Hashing

② Hashing

8:30




THANK YOU GW
SOLDIERS !