Graph Data Structure

↳ Non-Linear Data Structure

1. Depth first Search
2. Breadth first Search

← Graph → Edges
→ Vertices / Nodes

↓ form a cycle undirected or directed

Tree → 1) Preorder 2) Postorder 3) Inorder

↳ Not form a cycle

Directed

Application

↳ Networking Edge (Social media platform)

Ram ← Priya

Two types → simple graph (no loops)

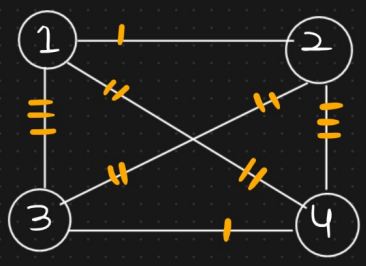
→ multigraph (cycle)

↓ loops are allowed

Degree

for each node

max degree ↕ complete graph



→ simple graph

→ undirected graph

→ Degree of all nodes = 3

↳ how many edges are connected to each node

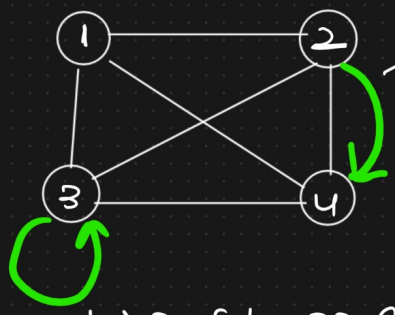
n = # nodes

(n-1)

Max Degree = n - 1

→ Min Degree = 0

(1)



→ Parallel edges are also allowed

→ Multigraph

↳ self loops are allowed

Degree(A) = 0

Null graph

A B Degree(B) = 0

C D Degree(D) = 0

Degree(C) = 0

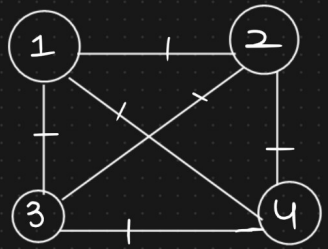
1) simple graph
   - null graph ↔ 0
   - complete graph ↔ n-1

Max Degree ↳ n-1

Min Degree → 0

2) multigraph

Properties of complete graph



n → # vertices or # nodes

Total number of edges = 6

sum of Degree

$\Rightarrow \frac{n * n-1}{2}$

$n = 4$

$\Rightarrow \frac{4 * 3}{2} = 6$

Sum of Degree = $4 * 3 = 12$

(4 ↓ # nodes, 3 ↳ Degree of each node)

$= n * (n-1)$

Total num of edges $= \frac{\text{sum of degree}}{2}$

sum of degree = Total num of edges $* 2$

$= 2 * E$

Relationship b/w # edges & # vertices

$E = \frac{V(V-1)}{2}$ — True

complete graph →

$2E = V^2 - V$

$E = O(V^2)$

By taking log on both sides;

$\log E = 2 \log V$

$\log E = O(\log V)$