Monday, 17 October, 2022 10:24 PM

Graphs

→ vertices : nodes, points, objects.
edges : arcs, lines, links, relations.

→ vertices set : V
edges set : E

$G = (V, E)$

$G = (\{a, b, c, d, e\}, \{ab, ac, cd, ce, de\})$

a & b are adjacent nodes (a, b)

$G = (\{v, w\}, \{vw\})$

$E = \{\}$
$C = \{\}$

BC: also allowed graph / multigraph

disjoint graph.

* Every graph can be considered disjoint graph.

Real life applications of graph:

1. Facebook

BC: also directed

2. Instagram

not directed graph

2. Weighted / un-weighted graphs

→ weighs : attributes to the edges of a graph

→ quantifiable

→ comparable

weighted graph

Weighted graph to directed graph	weighted to undirected graph	unweighted to directed graph	unweighted to undirected graph
Difficult	Difficult	Difficult	Easiest

Directed Acyclic graph (DAG)

DAG is a directed graph with no cycle or loop.

→ cyclic graph

→ DAG

Multi graph

A multi graph G is undirected graph in which multiple edges are allowed.

Multiple edges are both or more edges that connect the same two vertices.

Simple graph

Complete graph

→ every two vertices are adjacent.

K_n

Trees

is a special graph with no cycle.

Q. Let's say there is a tree with n nodes, how many edges are there in tree?? $(n - 1)$

Q. in relation of graph, max edges $\boxed{n_C_2}$

$\boxed{K_n}$

→ nodes
$E = n - 1$ nodes

Q.

$q = \{a, b, c, d, e, f, g\}$

$E = \{\}$

Q. connected graph

connected components

connected → graph
disconnected → multi-disconnected

→ H is a subgraph of a graph G which is a connected component of G.

H, H_2 → connected graph??

$a \to a$
$a \to b$
$c \to c$
$c \to d$

Graph representation

Q. Adjacency matrix

	a	b	c
a	0	1	1
b	1	0	1
c	1	1	0

symmetric matrix

n → no. of nodes
$n \times n$ square (3×3)

	a	b	c	d
a	0	1	1	1
b	0	0	1	1
c	0	0	0	1
d	0	0	0	0

q (vertices = 3)

Q. weighted graph

put weight value in vertices of a, b, c, d

Key:

any → Infinite

Imp → zero

→ nodes
→ edges

$E \to q(n)$

do → edges
→ edges

Q. adj. list

For every node, store in a list of its immediate adjacent nodes.

The same diagram with weights

→ $a \to [b, c]$
$b \to [c]$
$c \to [d]$
$d \to [\emptyset]$

→ $a \to [(b, 5), (c, 8)]$
$b \to [(c, 3)]$
$c \to [(d, 1)]$
$d \to [\emptyset]$

→ $a \to [b]$
$b \to [c]$
$c \to [d]$
$d \to [\emptyset]$

→ $a \to [(b, 5), (c, 8)]$
$b \to [(c, 3)]$
$c \to [(d, 1)]$
$d \to [\emptyset]$

→ $a \to [b]$
$b \to [c]$
$c \to [d]$
$d \to [\emptyset]$

Key: → use again.

Q. given a graph, list the name of its adjacents??

list

Q. given two nodes, check if they are directly connected??

matrix

Graph Traversals

Dont do not have cycles

→ $a \to b \to c \to d \to \ldots$
Infinite loop!

stuck

Two Traversal techniques

1. Breadth first search

(for a graph, start with a node and explore DFS through)

0	1	2	3	4	5	6

→ 0 1 2 3 4 5 6
→ 0 4 5 6 1 2 3
→ 0 4 5 1 6 3

```
bool S[V] = {0};
void DFS (int s)
{
    push s;
    for ele in q do ele = top();
        for (int i=0; i<n; i++)
            if ( (adj[ele][i]) && !S[i])
                S[i] = 1;
                enqueue(i);
                DFS(i);
        }
    }

main () {
    S[V] = {0};
    DFS(s);
    for i -> n:
        if ( (S[i]) == 0)
            DFS(i);
}
```

→ stackoverflow

Q. given an undirected graph, tell whether the graph is a connected graph or not??

0	1	2	3	4	5	6

1. Pick any random node.
2. Traverse DFS from there.
3. check if $S[V]$ has any node in false, then graph is disconnected.