Graph Theory

Q. Why study Graph Theory?

For graph theory aspect:

1. In Data Structure, we are concerned about how to store a graph in an efficient manner
2. In Algorithms, we are concerned about the computational algorithms in graphs to find the shortest path, minimum spanning tree, etc
3. In Discrete Mathematics, we are concerned about
   (i) The terminologies
   (ii) Special types of graphs — from sets and relations perspective.
   (iii) Properties — we get them with the help of counting/combinatorics.

Some examples of graphs:-

1. Web graph — Web is itself a graph on which page rank algorithm works.
2. Networks (computers) — Computer network as a graph
3. Social graph — Facebook, twitter, etc.
4. Neural network — Graphs are used to mimic human brains.

Note: There are special databases called Graph databases whose primary job is to store large graphs like web graph, social graph, etc., and to operate on them.

Q. What is a graph?

A graph(G) is a 2 tuple, i.e., it consists of ~~two things~~:

(i) vertices – can be thought of as towns / cities in road networks

(ii) edges – can be thought of as the roads connecting two towns / cities. Roads can be two way or one way.

$$G = (V, E)$$

$V \downarrow$ set of vertices / nodes

$E \hookrightarrow$ set of edges

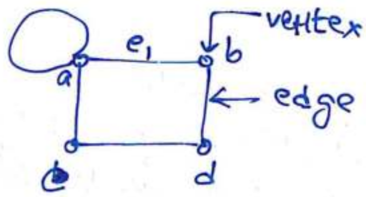


Fig: Undirected graph

$$e_1 = \underbrace{(a,b) \text{ or } (b,a)}_{\text{unordered pair}}$$

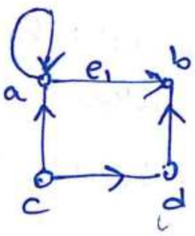


Fig: Directed graph

$$e_1 = \overbrace{(a,b)}^{\text{ordered pair (order is important)}}$$
$$e_1 \neq (b,a)$$

The edge e_1 is from a to b.

For directed graphs, $G_d = (V, E)$,

$$E \subseteq \underbrace{V \times V}_{\text{cross product}}$$

The set of edges (E) is a subset of the cross product of two sets.

From set theory perspective we know that a relation of two sets is a subset of the cross product of two sets.

Therefore we can think of directed graph edges forming a relation on the vertex set.

Incase of undirected graph, we can have (a,b) or (b,a). Therefore we cannot think of edges forming relations because in relations (a,b) is different from (b,a).

So undirected graphs can be thought of as set of vertices and a set of unordered pairs of vertices (edge set).

And similarly, directed graphs can be thought of as set of vertices and a set of ordered pairs of vertices (edge set).

Edge-labelled graph :

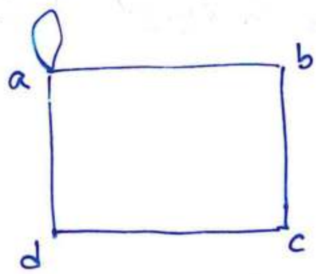


Fig: vertex-labelled graph

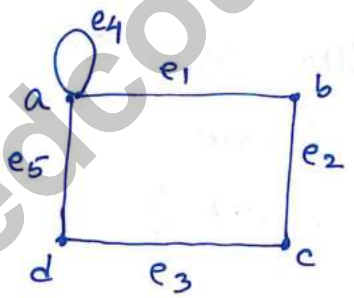
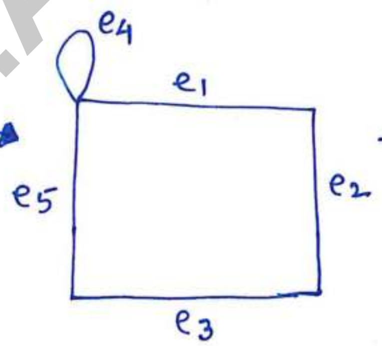


Fig: Edge-Labelled graph (Both vertex and edge are labelled)

Add edge labels

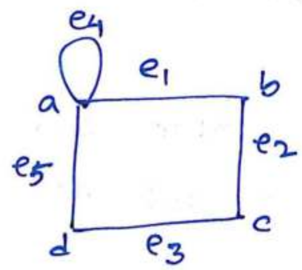


In a vertex labelled graph, if we add labels to the edges then the graph is called as edge-labelled graph.

Adjacent vertices :

Adjacent vertices implies there exists an edge between the vertices.

G :



The vertices a and b are adjacent because there exists an edge e_1 connecting both the vertices.

The vertices a and c are not adjacent because there exists no edge connecting both the vertices.

We can also say, the vertices a and b are adjacent because the edge e_1 is incident on vertices a and b.

Order of a graph :

$O(G) = \#$vertices

Means 'Number of'

$= 4$

Size of a graph :

size (G) = # edges

= Number of edges

= 5

Mixed graph :

The graph where some edges are undirectional. So, the directional and some are mixed graphs real world road networks are mixed graphs as some roads are one-way while others are bidirectional.

Simple Graph: It is a graph which has :-

(i) No self-loops

(ii) No multiple edges

G_0 :

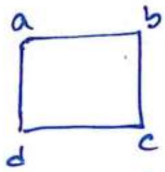
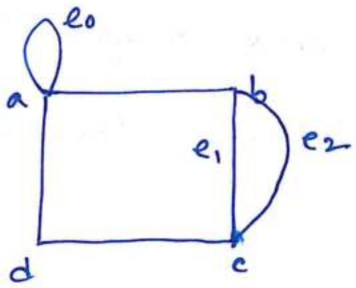


Fig: simple graph

G_1:



There exists an edge from vertex a to itself, this edge is called a <u>self loop</u>.

Between vertices b and c, there are two edges. These denote the <u>multiple edges</u>.

Multi-graph: It is a graph which

(i) Does allow self loops

(ii) Allows multiple edges.

In the graph G_1, we can see the multiple edges present between vertices b and c. But G_1 is not a multiple graph since it has a self loop.

~~while~~ G_2 is a multi-graph.

G_2 :

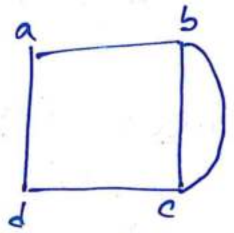


Fig: Multi-graph

The concept of multi-graph comes from multi-set in set theory, where a set containing same element repeating multiple times. Here $E = \{(b,c),(b,c), \dots\}$

The unordered pair (b, c) is repeating in the edge set E for graph $G_2(V, E)$.

Pseudo-graph :

A pseudograph is a non-simple graph in which both graph loops and multiple edges are permitted.

Empty graph : / Null graph :

Inconsistent terminology
Different authors use different definitions.

One possible definition :

1. Empty graph : An empty graph on n ~~nod~~ vertices consists of n isolated vertices with no edges. Such graphs are also called edgeless graphs
2. ~~An~~ Null graph : The empty graph with 0 ~~node~~ vertex.

Note : Null graph is used to refer to :-
(a) empty graph
(b) empty graph on 0 vertices.

Note : Therefore whenever empty graph /null graph is referred, their definitions should be clearly defined.

Interesting Example :

The graph G has:

(i) No self loops

(ii) No multiple edges

Therefore graph G is a simple graph.

G:

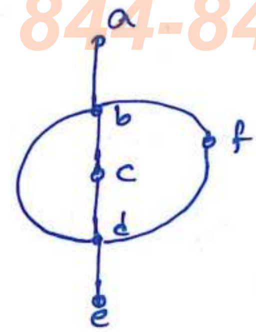


Fig: Simple Graph

Degree (vertex) :

Degree of a vertex, deg(v) = # edges incident on vertex v

Note: 1. Self loops count as 2.
E.g., vertex b has degree 4.

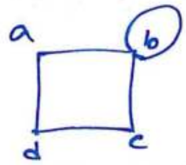


Fig: Undirected graph

2. If vertex has degree = 1, then it is called a pendant vertex.

3. A tree is special type of graph where the leaf nodes have degree = 1. A leaf node is also called as pendant vertex.

4. A lone vertex (isolated vertex) has degree = 0.

In case of directed graphs, vertex b has indegree = 2 and outdegree = 2.

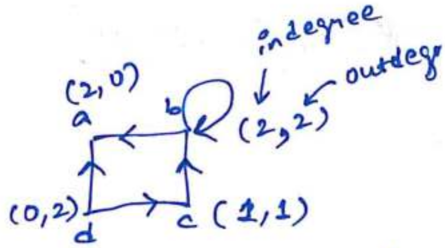


Fig: Directed graph

Note: 1. Sink : only incoming edges. Vertex a is a sink

2. Source : only outgoing edges. Vertex d is a source.

(*) Degree sequence :

Given an undirected graph, we write down the degrees of each vertices in increasing / decreasing order.

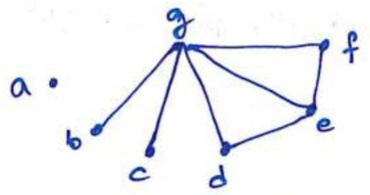


Fig : Undirected graph

deg (g) = 5
deg (e) = 3
deg (f) = deg (d) = 2
deg (c) = deg (b) = 1
deg (a) = 0

∴ The degree sequence : [0, 1, 1, 2, 2, 3, 5]
(or)
[5, 3, 2, 2, 1, 1, 0]

Note: For a given graph G, the degree sequence is unique.
But given a degree sequence, there can be multiple graphs with the same degree sequence.

(*) Min-degree : It is the minimum degree of any graph. It is represented by δ.

(*) Max-degree : It is the maximum degree of any graph. It is represented by Δ.

Given a simple undirected graph, the $\delta(G) = \delta$ = min. degree could be 0. And the max.degree $\Delta(G)$ or Δ could be (n-1).

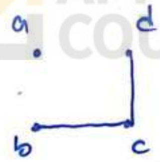


Fig: Graph with min-degree = 0 for vertex a.

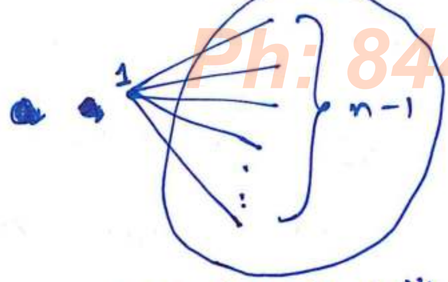
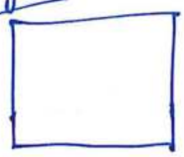


Fig: Graph with max-degree = (n−1) for vertex 1.

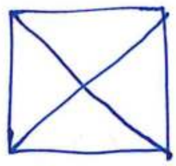
(*) **Regular Graph** : A regular graph is a graph where min. degree (δ) = max. degree (Δ)

2-regular



$\delta = 2 = \Delta$

3-regular



$\delta = 3 = \Delta$

(*) **Hand shaking Theorem (undirected graph)**:

The sum of degrees of vertices is equal to the twice the size of the graph. size of graph is the number of edges.

$$\therefore \sum_{v_i \in G} \deg(v_i) = 2 * \text{size}(G) = 2 * |E|$$

This theorem holds for any undirected graph (simple or non simple with self loops or non simple with multiple loops)

Note: Corollaries (results) that follow from the above theorem :-

(i) Sum of degrees is always even

(ii) No graph can have odd number of vertices of odd degree. (or) # odd degree vertices is even.

E.g., Given degree sequence: (1, 1, 2, 2, 3, 4, 5, 7)

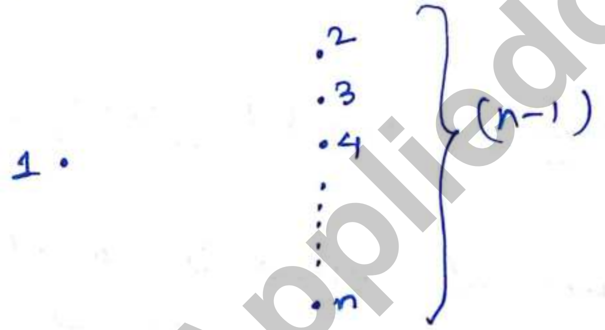
The number of odd degree vertices = 5

By add the 5 odd degree vertices, we don't get even number. Thus the 1st corollary is not satisfied.

(*) Maximum number of edges (simple undirected graph):

For combinatorial point of view, we have n vertices, and we can choose any 2 vertices and create an edge between them, ie., ${}^nC_2$.

Another way to find max. number of edges :-



$(n-1)+(n-2)+(n-3)+\dots+0$

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

$= {}^nC_2$

Note: If a simple undirected graph consists of ${}^nC_2$ edges, ie., for every pair of vertices there is an edge (or) all the possible edges are present, such a graph is called a complete graph.

$\boxed{(v_i, v_j)\ \forall\ v_i, v_j \text{ and } v_i \neq v_j}$

And a complete graph of n vertices is called as K_n

⊛ Relation between Regular Graphs and Complete Graphs:

For a k-regular graph, every vertex has k-edges.

Therefore, sum of all degrees $= n * k = 2 * |E|$

$\therefore \; |E| = \#$ of edges $= \frac{nk}{2}$

<u>Note:</u> An $(n-1)$-regular graph is a complete graph (K_n).

In this case, $K = (n-1)$

$$\therefore \; |E| = \frac{nK}{2} = \frac{n(n-1)}{2} = {}^nC_2$$

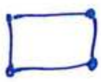
We know that a simple undirected graph with ${}^nC_2$ edges is a complete graph K_n.

$\therefore \; \boxed{(n-1)\text{-Regular graph} = K_n}$

<u>Note:</u> (1) If a graph is a complete graph $\Rightarrow$ regular graph

$K_n \Longrightarrow (n-1)$-regular graph

(2) A regular graph $\not\Rightarrow$ complete graph
(need not be)

E.g.,  $\left.\right\}$ 2-Regular graph
But it is not a complete graph.

⊛ Handshaking Theorem (Directed graph):

For a given directed graph, every edge will be counted once as indegree and once as outdegree.

That means the sum of indegrees and sum of outdegrees is same. $\boxed{\sum_{v_i \in G} \text{indeg}(v_i) = \sum_{v_i \in G} \text{outdeg}(v_i)}$

For any vertex, $\boxed{\deg(v) = \text{indegree}(v) + \text{outdegree}(v)}$

$\left\{ v_1 \longrightarrow v_2 \right\}$ outdegree $(v_1) = 1$
indegree $(v_2) = 1$

$$\therefore \boxed{\sum_{v_i \in G} indeg(v_i) = \sum_{v_i \in G} outdeg(v_i) = |E|}$$

(*) Degree sequence of directed graph :-

$$\underbrace{(1,2)}_{\substack{\text{(indegree, outdegree)} \\ \text{of} \\ \text{1st vertex}}}, (2,2), (2,y), \underbrace{(x,2)}_{\substack{\text{(indegree, outdegree)} \\ \text{of} \\ \text{4th vertex}}}$$

(indegree, outdegree) . . . . . . (indegree, outdegree)

To find the relation between x and y.

We know that the sum of indegrees = sum of outdegrees

$$\Rightarrow 1+2+2+x = 2+2+y+2$$

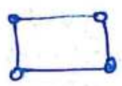
$$\Rightarrow \boxed{x = y+1}$$

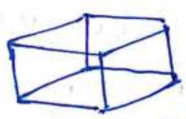
(*) Hypercube graph (Q_n) :-

Order $O(Q_n) = 2^n$ = # of vertices

A hypercube graph (Q_n) has 2^n vertices and it is n-regular.

Q_1 : 

Q_2 : 

Q_3 :  } It is 3-Regular and it has 2^3 vertices

Cube is a structure in 3D space

Q_4 : Hypercube is structure in 4D space which we can't visualize.

To get the number of edges, we use handshaking theorem :-

$$2 * \# \text{edges} = \overbrace{\underbrace{2^n}_{\#\text{ of vertices}} * \underbrace{n}_{\text{degree of each vertex}}}^{\text{sum of the degrees}}$$

$$\Rightarrow \# \text{edges} = \left(\frac{2^n \cdot n}{2}\right) = \left(2^{n-1} \cdot n\right)$$

⊛ Havell - Hakimi Theorem / Algorithm :

Purpose : It is used to check if a given degree sequence is valid degree sequence of a simple graph or not.

Graphical sequence : It is a term used for degree sequence of a simple graph.

Given a degree sequence in decreasing order, we want to find whether this sequence is a degree sequence of a simple graph.

Eg1. (5, 3, 3, 3, 2, 2, 1)

Odd degree vertices = 5

This is impossible as the number of odd degree vertices should be even, which means this can never be a graphical sequence.

E.g2. (5, 4, 3, 3, 3)

odd degree vertices = 4

Even though the number of odd degree is even, but this sequence is not a graphical sequence.

Because for any given graph, according to the Handshaking theorem, we will have even # of odd-degree vertices. But if we have even # of odd-degree vertices, ~~we~~ this does not imply that ~~we~~ ~~to~~ it is a valid graph.

Graph $\Rightarrow$ even # of odd degree vertices
$\not\Leftarrow$

E.g 3. S_1: ($\underline{5}$, $\underline{3}$, $\underline{3}$, $\underline{3}$, 2, 2, $\underline{1}$, $\underline{1}$)

The given degree sequence(S_1) doesnot violate the handshaking theorem as the number odd degree vertices is even.

Now ~~acco~~ ~~&~~ let us try to find if the given sequence is a valid degree sequence ~~accordo.~~ (graphical sequence) for a simple graph or not.

According to Havell-Hakimi theorem:

1. sort all the vertices in the decreasing order of degrees.

2. (a) Pick the largest element. Remove it from the list and construct the list again.

Let the largest element be d_1.

$\therefore d_1 = 5$

We remove d_1 from the list and reconstruct the list.

(3, 3, 3, 2, 2, 1, 1)

(b) Since $d_1 = 5$, take the first 5 numbers from the reconstructed list and subtract 1 from 5 numbers.

$(\overline{3,3,3,2,2},1,1)$

$\Downarrow$

$S_2 : (2,2,2,1,1,1,1)$

According to Havell-Hakimi theorem, if sequence S_1 is a valid graphical sequence IFF sequence S_2 is a valid graphical sequence

(3) Repeat steps (2a) and (2b) till the time we know whether the subsequent sequences are graphical sequences or not.

$\therefore \; S_2 : (2,2,2,1,1,1,1)$

$\therefore \; d_1 = 2$

$\therefore \; (2,2,1,1,1,1)$

Now, since $d_1 = 2$,

$(\overline{2,2},1,1,1,1)$

$\Downarrow$

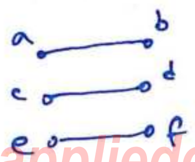
$S_3 : (1,1,1,1,1,1)$

Now, if S_3 is a valid graphical sequence, then S_2 is a valid graphical sequence.

And if S_2 is a valid graphical sequence, then S_1 is a valid graphical sequence.

We can draw the graph for the sequence S_3.

G:



$deg(a) = deg(b) = deg(c) = deg(d) = deg(e) = deg(f) = 1$

It is a simple graph with no self loops and no multiple paths

$\therefore S_3 : (1,1,1,1,1,1)$ is a valid graphical sequence.

$\therefore S_2$ and S_2 are also valid graphical sequence.

Eg. 4. $S_1 : (1, 0, 0)$

It is not a valid graphical sequence because

(i) It violates the handshaking theorem, as the number of odd degree vertices is not even.

(ii) Using Havell-Hakimi theorem:-

$d_1 = 1$

$S_2 : (-1, 0)$

Degree of a vertex can never be less than zero.

$\deg(v) \geq 0$

G:

The edge started from vertex 'a' has to end somewhere, but none of the other vertices can have an incident edge since their degree is zero.

$\therefore$ The degree sequence S_1 is invalid.

E.g. 5. S_1 : (5, 4, 3, 3, 3)

(i) S_1 doesnot violate Handshaking theorem.

(ii) ~~does~~ Using Havell - Hakimi theorem :

$d_1 = 5$

S_2 : (3, 2, 2, 2, ~~X~~)

No fifth element

∴ S_2 is not a valid graphical sequence.

∴ S_1 is not a valid graphical sequence

E.g. 6. S_1 : (5, 4, 3, 3, 3, 0, 0, 0, 0)

(i) It doesnot violate Handshaking theorem.

(ii) Using Havell - Hakimi theorem :

$d_1 = 5$

S_2 : (3, 2, 2, 2, -1, 0, 0, 0)

We cannot have -1 as a degree of a vertex. ∴ S_2 is not a valid graphical sequence.

∴ S_1 is ~~no~~ also not a graphical sequence.

E.g. 7 S_1 : (5, 5, 4, 4, 3, 2, 2, 1)

(i) It doesnot violate Handshaking theorem

(ii) Using Havell-Hakimi theorem :

∴ $d_1 = 5$

S_2 : (4, 3, 3, 2, 1, 2, 1)

We need to re-arrange S_2 in decreasing order

S_2' : (4, 3, 3, 2, 2, 1, 1)

∴ $d_1 = 4$

S_3 : (2, 2, 1, 1, 1, 1)

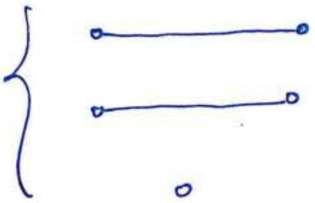
$\therefore d_1 = 2$

S_4 : (1, 0, 1, 1, 1)

We re-arrange S_4 in decreasing order :-

S_4' : (1, 1, 1, 1, 0)

We can continue ~~to~~ with Havell-Hakimi (or)
we can make an attempt at drawing the graph.

G :


The graph G is a valid simple graph.

$\therefore$ S_4' is a valid graphical sequence.

$\therefore$ S_3, S_2 and S_1 are also graphical sequences.

⊛ Havell-Hakimi Theorem (Definition) :

If $d_1, d_2, d_3, \ldots d_n$ is a decreasing sequence of non-negative integer, this is a graphical sequence IFF the resultant sequence

$$\underbrace{d_2-1,\ d_3-1,\ d_4-1,\ \ldots\ d_{d_1+1}-1}_{d_1 \text{ elements}},\ d_{d_1+2},\ d_{d_1+3}, \ldots d_n$$

is a graphical sequence.

Corollary: Any graphical sequence must have atleast one repetition.

E.g., For the given ~~graphical~~ sequence

$S_1 : (7, 6, 5, 4, 3, 2, 1, 0)$

It is not a graphical sequence because by Havell-Hakimi theorem,

$d_1 = 7$

$S_2 : (5, 4, 3, 2, 1, 0, \underline{\underline{-1}})$ ✗

Degree of a vertex cannot be less than zero.

In sequence S_1, there is no repetition.

Every graphical sequence must have ~~on~~ atleast one repetition.

But having one repetition doesnot make the sequence a graphical sequence.

E.g., $S_1 : (5, 4, 3, 3, 3)$

S_1 has repetition but it is not a graphical sequence by Havell-Hakimi theorem.

As $d_1 = 5$ and $S_2 : (\underbrace{3, 2, 2, 2}, \underline{\times})$

only 4 numbers are present in the sequence

⊛ Average Degree Theorem (δ, Δ Theorem) :-

Given a graph G, let $|V| = n$ and $|E| = e$,

minimum degree $\le$ avg. degree $\le$ maximum degree

$$\Rightarrow \delta \le \frac{\text{sum of all the degree of the vertices}}{\text{number of vertices}} \le \Delta$$

$$\Rightarrow \boxed{\delta \le \frac{2e}{n} \le \Delta}$$

Ph: 844-844-0102

E.g Given $O(G) = 10 = n$

$\delta = 3$

what is the minimum number of edges in the graph?

$\delta \leq \frac{2e}{n}$

$\Rightarrow \frac{3 \times \cancel{10}^{5}}{\cancel{2}} \leq e$

$\Rightarrow e \geq 15$

$\therefore$ Minimum number of edges is greater than or equal to 15.

E.g. Given $n = 10$, $e = 16$

what is the minimum and maximum degree?

We know,

$\delta \leq \frac{2e}{n} \leq \Delta$

$\Rightarrow \delta \leq \frac{32}{10} \leq \Delta$

$\Rightarrow \delta \leq 3.2 \leq \Delta$

$\therefore$ Maximum possible value of min. degree is 3 and minimum possible value of max. degree is 4

$\boxed{\delta \leq 3}$ and $\boxed{\Delta \geq 4}$

$\therefore \delta \leq \left\lfloor \frac{2e}{n} \right\rfloor$ and $\Delta \geq \left\lceil \frac{2e}{n} \right\rceil$

Floor ceil

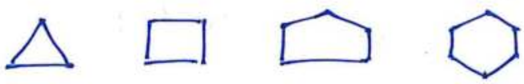
Mail: gatecse@appliedcourse.com

⊛ Cycle Graph (C_n) :

C_n is a cycle graph with n vertices and $n \geq 3$.

$n = |V|$ and $|E| = n$

Cycle graphs are 2-regular. Cycle graphs are polygons.

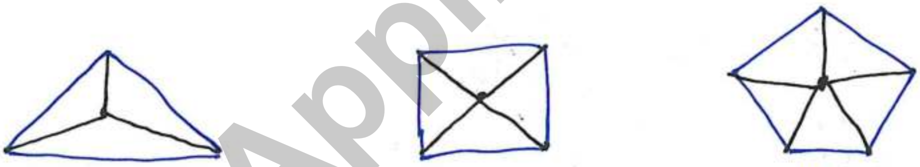
E.g.,  . . . .

⊛ Wheel Graph (W_n) :

W_n is a wheel graph with n vertices and $n \geq 4$ & $\boxed{e = 2(n-1)}$.

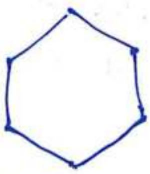
Construction of wheel graph :

cycle graph + one more vertex which is adjacent to all other vertices .



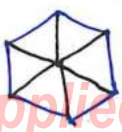
W_4 W_5 W_6

To construct W_n, we use C_{n-1}



G : C_{n-1}

C_{n-1} has $(n-1)$ edges. Now we add one more edge which is adjacent to all other vertices .

G : 

Due to the addition of the new vertex, more $(n-1)$ edges are added in the graph G.

Thus W_n has $2(n-1)$ edges.

Bipartite graph:

In a graph $G(V, E)$, if we can break the vertex set V in such a way that:

(i) $V = V_1 \cup V_2$, where V_1 & V_2 are sets

(ii) $V_1 \cap V_2 = \phi$, where V_1 & V_2 are disjoint sets

and if we have an edge (a, b) ~~then~~ such that $a \in V_1$, $b \in V_2$ or viceversa, i.e., vertex a and b cannot belong to the same set.

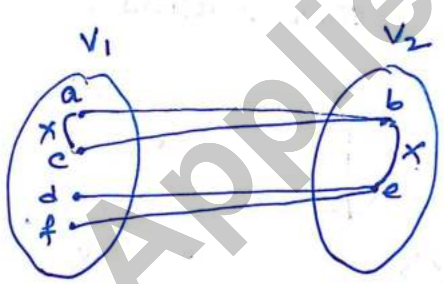


Fig: Bipartite graph G.

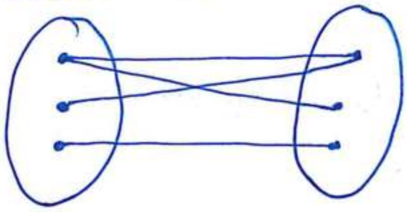
In this above figure, there cannot be an edge (a, c) and (b, e) since vertex a and c belong to set V_1, and vertex b and e belong to set V_2.

Note: Null Graph (ϕ_n) is a bipartite graph.

Because in Null graph we have n vertices and no edges. So, we can break/partition all the vertices into two sets and no edges violate the edge property of bipartite graph.

Application of Bipartite Graph:

(i) At e-commerce companies lot of interesting data analysis is done by considering all the customers as one set and all the products as the other set.



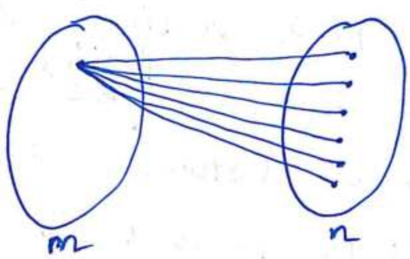
(ii) We can build a recommendation system using machine learning and graph theory.

Recommendation system: Given a customer what are the new products that we can suggest.

Recommendation systems are very popular and can be seen in Amazon, Facebook websites etc.

(*) Complete Bipartite Graph:

It is a bipartite graph which has total number of vertices of $(m+n)$. And total number of edges of $(m*n)$.

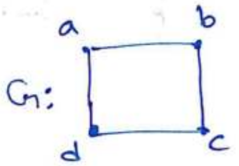


$\therefore |V| = m+n$

$|E| = m*n$

(*) Sub Graph:

Given $G = (V,E)$



Say we construct a new graph $G_1 = (V_1, E_1)$ such that $V_1 \subseteq V$ and $E_1 \subseteq E$

We can construct a subgraph G_1 from graph G in the following ways:-

(i) We pick edges $E_1 \subseteq E$ and if the vertex set $V_1 \subseteq V$ contains the corresponding incident vertices, then the subgraph G_1 is a valid subgraph.

E.g., From graph G, lets pick edges $\{e_1, e_4\}$

$\therefore E_1 = \{e_1, e_4\}$

Edges e_1 and e_4 are incident on vertices (a, b) and (a, d) respectively.

$\therefore$ If $V_1 = \{a, b, d\}$, then the subgraph G_1 is a valid subgraph.

Note: If $V_1 = \{a, b\}$, then the subgraph G_1 is not a valid subgraph.

(ii) We pick vertices $V_1 \subseteq V$.
In this case, for V_1, we need not have all the edges between the same vertices in graph G for subgraph G_1 to be present in subgraph G_1 to be a valid subgraph.

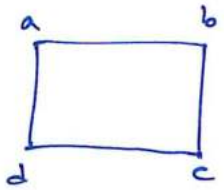
$\therefore$ If $V_1 = \{a, b, d\}$, then $E_1 = \{e_1\}$ or $\{e_1, e_4\}$ or ϕ are all valid subgraphs.

⊛ Induced Subgraph :

An induced subgraph is a subgraph where for a subset $S \subset V$, all the edges between the vertices of S should be the same edges between the same vertices of V.

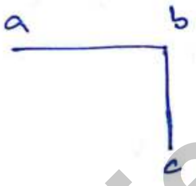
E.g.

G :

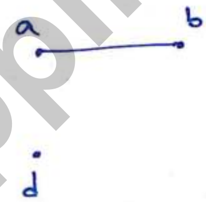


G(S) is an induced subgraph

G(S) :



But, G_1 :

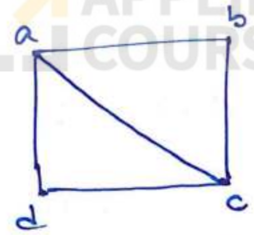


G_1 is a valid subgraph but not an induced subgraph as graph G has an edge (a,d) which is missing in the subgraph G_1.

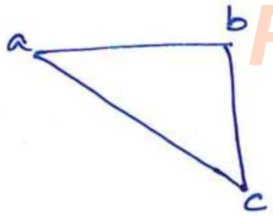
⊛ Clique :

A clique is an ~~induced subgraph~~ such that the induced sub-graph is complete.

Note: A complete graph is a graph where every pair of vertices has an edge.



Simple graph G

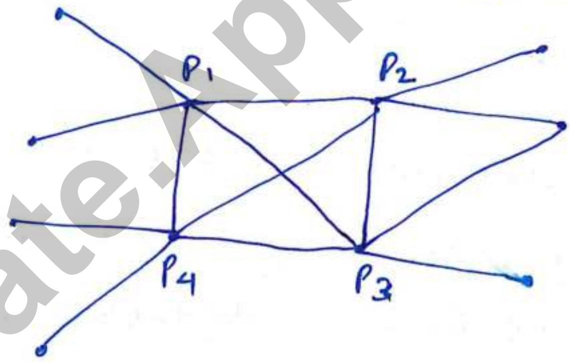


K_3

For the simple graph G, K_3 is an induced subgraph, and is complete. Therefore K_3 is a clique.

Application of clique:

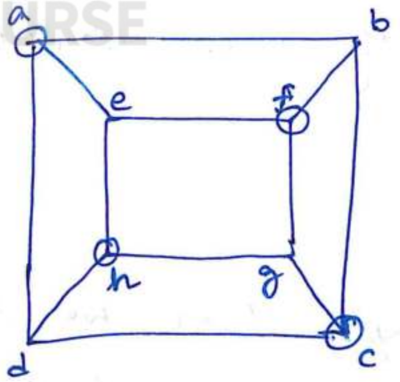
In social networks like Facebook, there are billions of people. If we represent each person as a vertex, for ~~four~~ classmates/close friends, we will see a clique being formed. So clique in social networks tell us about the strength of friendship (since every friend in the clique will be connected to others).



⊛ Independent Set:

Independent set can be thought of as anti-clique. It is a subset of vertices such that no two of them are adjacent.

G:

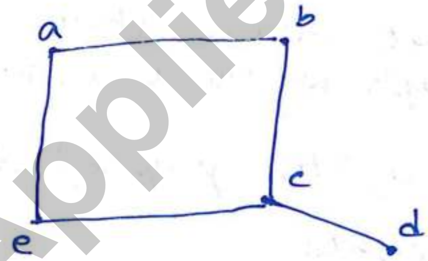


For the graph G, the subset of vertices ie., {a, f, h, c} form an independent set as no two vertices are adjacent to each other.

⊛ Walk and Paths (Undirected Graph) :

A walk is a sequence of vertices such that there is an edge between every pair of vertices. Let's consider the following undirected graph,

G:



edge

walk : a b c e a . . . . . . b c d

start   end

A path is a walk where vertices donot repeat except when start and end are the same vertex

path : a b c d

a b c e a

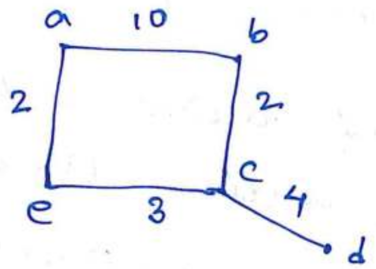
A path is also called as a simple path.

path length = # edges (number of)

Note: For directed graphs, we have directed paths which internally uses directed edges.

⊛ Weighted Graph:

A weighted graph is a graph where we have numeric values given to each edge of the graph. The weights can represent distance between two vertices.



The edge weight can be positive, negative values. If it is zero, that means there is no edge (in the context of a graph where edge weights means distance)

Path length (in weighted graphs) is the sum of the edge weights

Note: We encounter weighted graphs in Data Structures and Algorithms where we design shortest path Algorithms, minimum spanning tree algorithms etc.

In graph theory mostly we will study unweighted graphs (As far as GATE is concerned).

Path length for ~~a~~ the path a b c d = 16
$10 + 2 + 4$

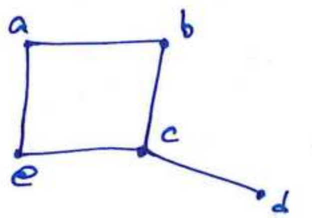
Note: In an unweighted graph, the path length is the number of edges itself.

⊛ Shortest Path:

A shortest path between vertices u and v is a simple path with minimum # edges.

The length of the shortest path u to v = $d(u,v)$

E.g.,



$d(a,d) = 3$
$d(a,b) = 1$
$d(a,c) = 2$

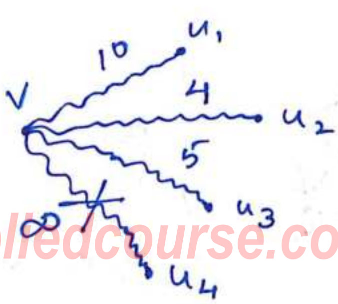
⊛ Eccentricity:

$$\epsilon(v) = \max_{u} d(v,u) \qquad \forall u \in G$$

Eccentricity of a vertex v is defined as the ~~the~~ maximum over for all the u, the shortest distance between v to u.

denotes an ~~path~~ edge between vertices a & b

denotes a path between vertices a & b



If there is no path between vertices v and u_4, the $\epsilon(v) = \infty$ as eccentricity takes the maximum value.

Note:

$a \rightsquigarrow b$ denotes a directed path between vertices a and b in a directed graph.

(*) Radius(G):

The radius of a graph is the minimum ~~of~~ eccentricity across all the vertices.

$$\text{Radius (Graph G)} = \min\left(\epsilon(v_1), \epsilon(v_2), \epsilon(v_3), \epsilon(v_4), \ldots \ldots \epsilon(v_n)\right)$$

$$\boxed{\text{Radius}(G) = \min_{v \in V} \epsilon(v)}$$ where V = set of vertices.

$$\text{Radius}(G) = \min_{v}\left(\max_{u} d(v,u)\right) \quad \forall\, u,v \in G$$

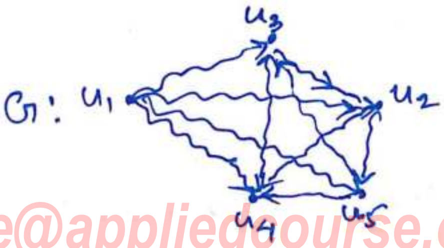
(*) Diameter (d(G)):

The diameter of a graph, $d(G)$, is the maximum eccentricity across all the vertices.

$$d(G) = \max_{v} \epsilon(v) = \max\left(\epsilon(v_1), \epsilon(v_2), \ldots \epsilon(v_n)\right)$$

= maximum shortest distance between a pair of vertices (u,v) = $\max_{u,v} d(u,v)$

where $u, v \in V$.



For all possible pair of vertices (u,v) find the shortest distances and take the maximum value.

$d(G)$ = largest of the shortest paths between any pair of vertices.

= shortest path length between two farthest vertices.

farthest: vertices which have the largest shortest path

For a null graph with 4 vertices (Φ_4),

diameter $(\Phi_4) = \infty$

Φ_4

Because, shortest path between all vertices to the other vertices is ∞.

$\therefore$ Diameter is ∞.

$\therefore$ Diameter $(\Phi_n) = \infty$

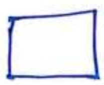
Likewise, for a complete graph (K_n),

diameter $(K_n) = 1$, because $d(v_i, v_j) = 1$ between any two vertices.

In case of cycle graph (C_n),

for C_3: 

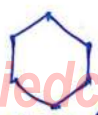
Diameter $(C_3) = 1$ $\quad \because C_3 = K_3$

for C_4: 

Diameter $(C_4) = 2$.

for C_5: 

Diameter $(C_5) = 2$

for C_6: 

Diameter $(C_6) = 3$

$\therefore$ $\boxed{\text{Diameter}(C_n) = \lfloor \frac{n}{2} \rfloor}$

floor

Solved Problems – 1

Q. The number of distinct simple graphs with upto three nodes is :

A. 15

B. 10

C. 7

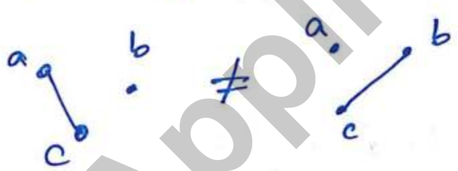
D. 9

Soln. For 3 vertices :–

0 edge 1 edge 2 edges 3 edges $\Big\} \rightarrow 4$

Note : The question doesnot mention about the vertices being labelled (or labelled graph).

In case of a Labelled graph,



we have to count the above cases as different.

For 2 vertices :–

0 edge 1 edge $\Big\} \rightarrow 2$

For 1 vertex :–

isolated/lone vertex $\Big\} \rightarrow 1$

For 0 vertex :-

$V = \phi$, $E = \phi$ $\Big\} \longrightarrow 1$

$\therefore$ Total number of distinct simple graphs with upto three nodes (vertices) is 8.

Note: Some authors don't consider 0 vertex to be a graph. Therefore out of the available options, option C, i.e, 7 is the best choice.

$\therefore$ Answer is C.

Q. How many undirected graphs (not necessarily connected) can be constructed out of a given set $V = \{v_1, v_2, v_3, \ldots \ldots v_n\}$ of n vertices?

A. $\frac{n(n-1)}{2}$

B. 2^n

C. $n!$

D. $2^{\frac{n(n-1)}{2}}$

Soln. Given number of edges = n (as the graph is undirected graph).

max # edges $= {}^nC_2 \longrightarrow K_n$

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

$\therefore |E| = \frac{n(n-1)}{2}$

Now when we are constructing a graph, we can select any subset of edges from $|E|$

From Counting, we know that for a set of size k, # subsets $= 2^k$.

$\therefore$ Number of undirected graphs can be constructed

$= 2^{\left(\frac{n(n-1)}{2}\right)}$

Answer is D.

4

Q. How many graphs on n labeled vertices exist which have atleast $\frac{(n^2-3n)}{2}$ edges?

A. $\left(\frac{n^2-n}{2}\right)C_{\left(\frac{n^2-3n}{2}\right)}$

B. $\sum_{K=0}^{\left(\frac{n^2-3n}{2}\right)} {}^{(n^2-n)}C_K$

C. $\left(\frac{n^2-n}{2}\right)C_n$

D. $\sum_{K=0}^{n} \left(\frac{n^2-n}{2}\right)C_K$

Soln Given n labeled vertices, i.e., every vertex has a label:

$v_1, v_2, v_3, \ldots\ldots v_n$

Since it's not mentioned, we are assuming the graph to be a simple and undirected.

For n vertices, the maximum number of edges $= \frac{n(n-1)}{2} = \frac{n^2-n}{2} = a$ (let)

Given that the graphs should have atleast $\left(\frac{n^2-3n}{2}\right)$ edges $= b$ (let)

$\#$edges $= e$ (let)

$$\therefore \frac{n^2-3n}{2} \le e \le \frac{n^2-n}{2}$$

$$\Rightarrow b \le e \le a$$

Note: If we calculate:

$$a - b$$

$$\Rightarrow \frac{n^2-n}{2} - \frac{n^2-3n}{2}$$

$$\Rightarrow \frac{n^2-n-n^2+3n}{2}$$

$$\Rightarrow \frac{2n}{2}$$

$$\Rightarrow n$$

$$\therefore \boxed{a-b=n}$$

Out of a edges we can select the following edges to construct a graph:

b edges $\rightarrow {}^aC_b = {}^aC_{a-b} = {}^aC_n = {}^{\frac{n(n-1)}{2}}C_n$

$(b+1)$ edges $\rightarrow {}^aC_{b+1} = {}^aC_{a-b-1} = {}^aC_{n-1} = {}^{\frac{n(n-1)}{2}}C_{n+1}$

$(b+2)$ edges $\rightarrow {}^aC_{b+2} = {}^aC_{n-2}$

$\vdots$

a edges $\rightarrow {}^aC_a = {}^aC_0$

Note: From combinatorics, we know that,

$$\boxed{{}^nC_k = {}^nC_{n-k}}$$

Now, when we add all the terms:

$${}^aC_n + {}^aC_{n-1} + {}^aC_{n-2} + \dots + {}^aC_0$$

we get, $\sum_{k=0}^{n} {}^{a}C_k = \sum_{k=0}^{n} {}^{\frac{n(n-1)}{2}}C_k$

$$= \boxed{\sum_{k=0}^{n} {}^{\frac{n^2-n}{2}}C_k} \leftarrow \text{Answer}$$

$\therefore$ Option D is the Answer.

Q. Let G be a complete undirected graph on 6 vertices. If vertices of G are labeled, then the number of distinct cycles of length 4 in G is equal to

A. 15

B. 30

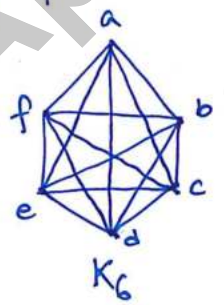
C. 90

D. 360

Soln. A cycle is a sequence of edges ~~vertices~~ that start and end at the same vertex.

The given graph is K_6

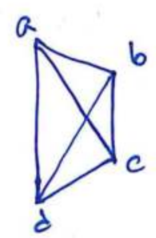
G:



K_6

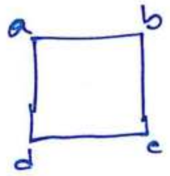
Let's consider that the cycle is formed using (a, b, c, d) vertices (4 vertices).

i.e.,

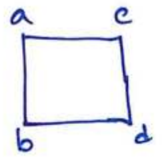


The possible distinct cycles of length 4 are :-

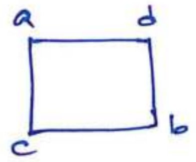
(i)



(ii)



(iii)



3 distinct cycles because of the labels.

Now we can choose 4 vertices out of 6 vertices in K_6 in the following ways :-

$^6C_4$

$\therefore$ Total number of distinct cycles of length 4 in a complete undirected graph of 6 vertices

$= {}^6C_4 * 3$

$= 15 * 3$

$= 45 \leftarrow$ Answer

Note! None of the options given are correct. And everyone got marks for attempting this question.

Q. A graph $G = (V, E)$ satisfies $|E| \le 3|V| - 6$. The min-degree of G is defined as $\min_{v \in V} \{\text{degree}(v)\}$. Therefore min-degree of G cannot be :-

A. 3

B. 4

C. 5

D. 6

Ph: 844-844-0102

Soln. We know,

$$\delta \leq \text{avg-degree} \leq \Delta$$

$$\therefore \delta \leq \text{avg-degree}$$

$$\delta \leq \frac{2 \cdot e}{n}$$

It is given that $|E| \leq 3|V| - 6$,

$$\therefore \quad \delta \leq \frac{2 \cdot e}{n} \leq \frac{2}{n}(3n-6)$$

$$\Rightarrow \boxed{\frac{n\delta}{2} \leq 3n-6}$$

$\therefore$ The graph G has to satisfy this criteria.

~~The~~ Let's try with given options :-

$\delta = 3$

$$\frac{n\delta}{2} \leq 3n-6$$

$$\Rightarrow \frac{3n}{2} \leq 3n-6$$

$$\Rightarrow \frac{3n}{2} - 6 \geq 0$$

$$\Rightarrow \frac{3n}{2} \geq 6$$

$\Rightarrow \boxed{n \geq 4}$ ← It is possible

$\delta = 6$

$$\frac{6n}{2} \leq 3n-6$$

$\Rightarrow \boxed{3n \leq 3n-6}$ $\quad n = 0, 1, 2, 3, \ldots$

No non-negative integer can satisfy $\boxed{3n \leq 3n-6}$

$\therefore$ Option D is the answer.

Mail: gatecse@appliedcourse.com

Q. The 2^n vertices of a graph G corresponds to all subsets of a set of size n for $n \geq 6$. Two vertices of G are adjacent if and only if the corresponding sets intersect in exactly two elements. The number of vertices of degree zero in G is :-

A. 1

B. n

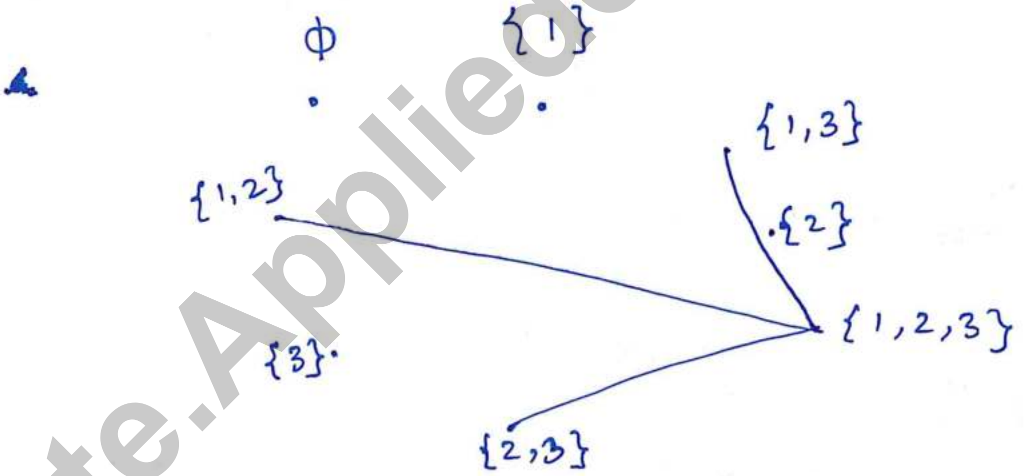
C. n+1

D. 2^n

Soln. Say $n = \{1, 2, 3\}$, i.e., size of $n = |n| = 3$

The subset of ~~a graph G~~ a set of size n is 2^n.

The 2^n subsets. And 2^n vertices corresponds to the



We get this from the criteria given that intersection of 2 sets is exactly 2 elements

$\therefore$ There is an edge between $\{1,2\}$ & $\{1,2,3\}$, $\{1,3\}$ & $\{1,2,3\}$ and $\{2,3\}$ & $\{1,2,3\}$

Note: Two vertices are said to be adjacent iff there exists an edge between them.

We see that vertices ϕ, $\{1\}$, $\{3\}$, $\{2\}$ have degree zero. Vertices $\{1,2\}$, $\{1,3\}$, $\{2,3\}$ have degree 1 and vertex $\{1,2,3\}$ have degree 3.

Given $n \geq 6$,

the vertices that will have degree zero are:-

ϕ — Null set

$\{1\}$, $\{2\}$, ⋮, $\{6\}$ — singleton set

degree = 0

vertices = 1 + 6 = (n+1)

$\{1,2\}$ ⋮ $\{1,2,3,4,5,6\}$

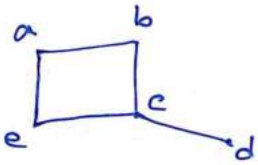
All these vertices will have non-zero degrees as they will be connected to ~~other~~ vertices with edges wherever the edge formation criteria satisfies.

ie., $\{1,2\}$ will be connected to $\{1,2,3\}$, $\{1,2,3,4\}$, $\{1,2,3,4,5\}$ and $\{1,2,3,4,5,6\}$ because the intersection of the sets result in exactly 2 elements ($\{1,2\}$).

∴ Option C is correct.

⊛ Bi-partite graph properties:

path: A path is a walk where vertices donot repeat ~~exp~~ except when start and end are the same vertex.



∴ path: a b c d

a b c e a ← cycle

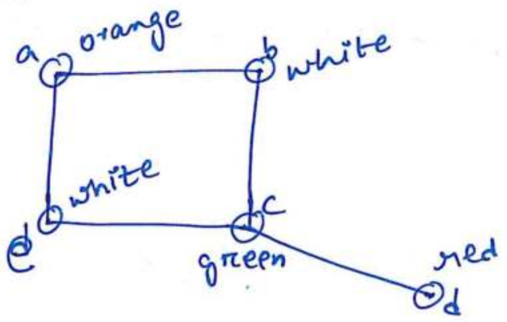
start (a), end (a)

cycle: A path where the start and end vertex is the same, such a path is called as cycle.

Note: cycle is different from a loop. A loop is from a vertex to itself.

Chromatic Number:

G:



Suppose we want to color the vertices such that no two adjacent vertices are of same color. This is basically called graph coloring.

The vertex 'a' if colored orange, vertex 'b' and vertex 'd' cannot take the same orange color.
∴ we can put white color to vertex b and vertex d.

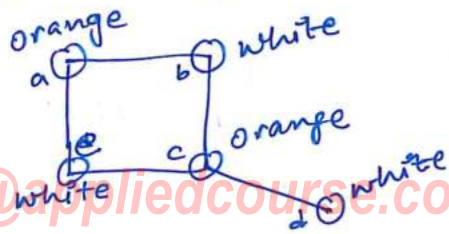
Vertex c can take orange color as vertex c is not adjacent to vertex a. Infact vertex c can take any color except for white.

Vertex d can take any color other than green as vertex c has been colored green.

Chromatic Number: what is the minimum # colors using which we can color the graph?

We have colored graph G with 4 colors. So can we color graph G in fewer than 4 colors?

G:



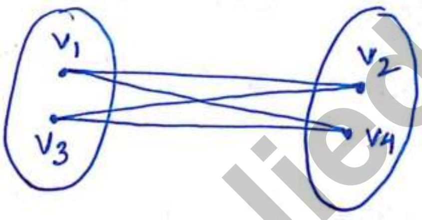
we try to minimize the number of colors required to color the graph by reusing the existing colors wherever possible.

Therefore, we are able to color the graph G using two colors. ∴ Chromatic number for the given graph is 2.

⊛ Important properties and theorems of bipartite graph.

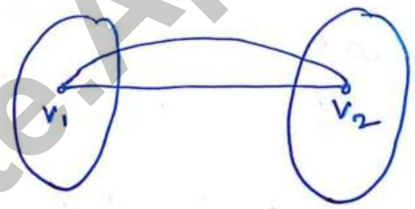
Theorem : A Graph G is bipartite iff G has no odd cycle.

Let's take a bipartite graph and prove G has no odd cycles.



There is an even cycle.

simple graph

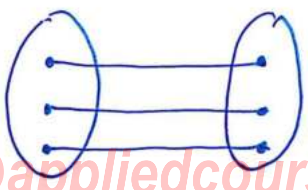


There is an even cycle.

multi-edge graph

Note :

Misconception : G is bipartite iff it has an even cycle.

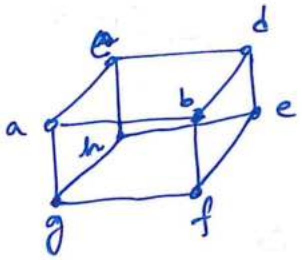


This is a bipartite graph that has no even cycle.

$\therefore$ No odd cycle present could also mean no cycles are present. No odd cycle present doesnot mean that even cycles should be present for a graph to be bipartite graph.

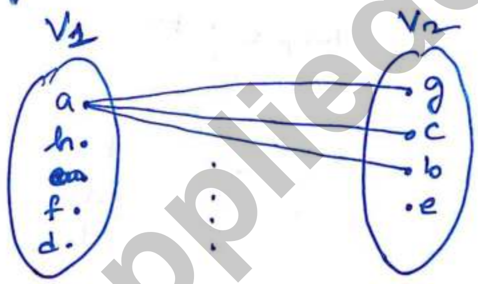
Let's consider the hyper-cuboid graph (Q_3)

Q_3 :



This graph has no odd cycles $\Rightarrow$ bipartite graph.

Let's try to draw Q_3 as a bipartite graph.

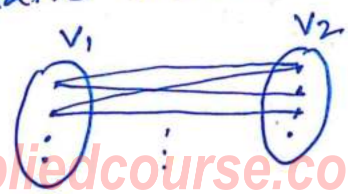


We are able to divide the vertex set V into two subsets in such a way that there are no edges between the vertices in the same vertex subset.

$\therefore$ Q_3 is a bipartite graph.

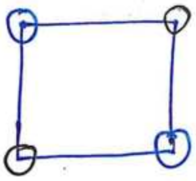
Theorem : A non-null graph is bipartite iff it is bi-chromatic.

Bichromatic means the chromatic # = 2.

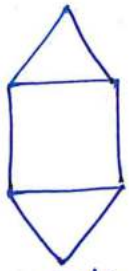


We can color subset V_1 and subset V_2 with different colors and we need only 2 colors.

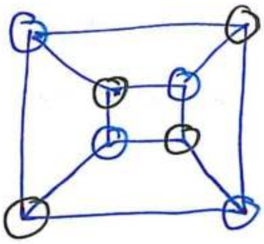
E.g., Let's find if the following graphs are bipartite or not.



Bipartite
2 coloring ✓
No odd cycles ✓

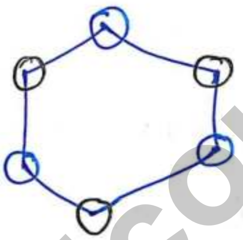


Not a bipartite graph
odd cycles present



Bipartite
2 coloring ✓
No odd cycles ✓

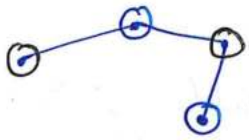
Not a bipartite graph.
odd cycle present



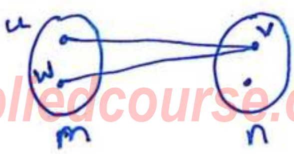
Bipartite graph
2 coloring ✓
No odd cycles ✓

Note: It is easier to check for 2-coloring in comparison to no odd cycles criteria.

⊛ Observation: An acyclic graph is always bipartite.



⊛ Observation: The diameter of a complete bipartite graph ($K_{m,n}$) ~~is~~ is 2.



diam $(K_{m,n})$

= A diameter is the shortest path between the farthest point

$= \max_{u,v} d(u,v)$

Since we are talking about a complete bipartite graph $(K_{m,n})$, between any two vertices in the same subset, the maximum distance is 2.

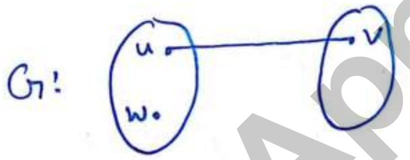
$\therefore$ diam$(K_{m,n}) = 2$

(*) **Property**: Ratio (diameter / chromatic number)

For complete bipartite graph,

Ratio (diam / chromatic number) $= \frac{2}{2} = 1$

For any bipartite graph (not complete bipartite graph),



the diameter can be between 2 to ∞. Since u to w, the distance is ∞ as w is not reachable from u.

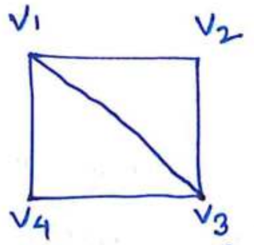
(*) **Representation of Graphs**:

In DS & Algo, we have seen two strategies of representing graphs:

(i) Adjacency Matrix
(ii) Adjacency List.
(iii) Incidence matrix.

1. Adjacency Matrix :

G:



Undirected simple graph

$$A: \begin{array}{c|cccc} & V_1 & V_2 & V_3 & V_4 \\ \hline V_1 & 0 & 1 & 1 & 1 \\ V_2 & 1 & 0 & 1 & 0 \\ V_3 & 1 & 1 & 0 & 1 \\ V_4 & 1 & 0 & 1 & 0 \end{array}_{n \times n}$$

for an edge, say (v_1, v_3), v_1 is the source vertex (row) and v_3 is the destination vertex (column).

The matrix size is $n \times n$ for n vertices in graph G. Therefore it is a square matrix.

Since it is an undirected simple graph, the diagonal elements are zero as there are no self loops.

This matrix is symmetric i.e., $A_{ij} = A_{ji}$.
since it is an undirected simple graph, if there is an edge from v_1 to v_2, then there is also an edge from v_2 to v_1.

This matrix is a 0-1 matrix, also called as binary matrix. It is because it is a simple graph with no multi-edges.

Adjacency matrix enable us to take a concept from Graph theory and translate to a concept in Linear Algebra.
And if we are given a square symmetric matrix, we can imagine it like a graph and map the problems in graph theory.

While studying Linear Algebra, we studied about
Eigen value and Eigen vector. Using the properties
of Eigen value and Eigen vector, we can also understand
graph theory. There is an area in graph Theory
called spectral graph theory which uses the
concepts of eigen value and eigen vector of adjacency
matrices to study properties of a graph.

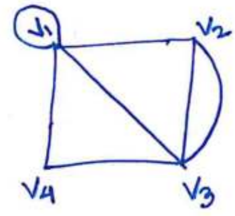
We can think of the whole internet as a graph.
One of the first algorithms that the co-founders
of google used to rank the search results is
called a page rank algorithm. The page rank
algorithm actually is based on the concepts of
eigen values and eigen vectors of the adjacency
matrix of the web graph. So whenever we are
using the search we are internally using the
inter-linking theory of graph theory and linear
algebra. This inter-linking theory is a very
powerful tool in mathematics.

Using adjacency matrix we represent :-

(i) loops :

(ii) multi-edges :

G :



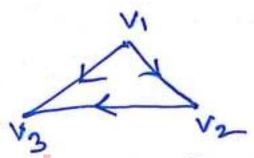
$$A : \begin{array}{c|cccc} & v_1 & v_2 & v_3 & v_4 \\ \hline v_1 & 2 & 1 & 1 & 1 \\ v_2 & 1 & 0 & 2 & 0 \\ v_3 & 1 & 2 & 0 & 1 \\ v_4 & 1 & 0 & 1 & 0 \end{array}$$

self loop

multi-edge

(iii) directed-edges :

G_1 :



$$A_1 : \begin{array}{c|ccc} & v_1 & v_2 & v_3 \\ \hline v_1 & 0 & 1 & 1 \\ v_2 & 0 & 0 & 1 \\ v_3 & 0 & 0 & 0 \end{array}$$

Directed edges

The adjacency matrix that we get need not be symmetric.

Because Incase of directed graph if we have an edge from v_1 to v_2, there may not be any edge from v_2 to v_1.

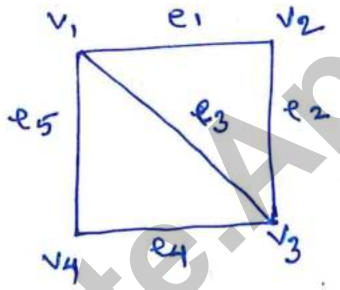
For undirected graph, the rowsum/column sum imply the number of edges that are incident on each vertex, ie., degree of a vertex.

For directed graph, row sum imply out-degree of a vertex and column sum imply in-degree of a vertex.

<u>Note</u>: row-sum and column sum are concepts in ~~Concepts~~ Linear Algebra and, out-degree and in-degree are concepts in Graph Theory.

(2) <u>Incidence</u> <u>Matrix</u>: (<u>Undirected graph</u>):

For incidence matrix, we need edge labelled graph.



$$\begin{array}{c|ccccc} & e_1 & e_2 & e_3 & e_4 & e_5 \\ \hline v_1 & 1 & 0 & 1 & 0 & 1 \\ v_2 & 1 & 1 & 0 & 0 & 0 \\ v_3 & 0 & 1 & 1 & 1 & 0 \\ v_4 & 0 & 0 & 0 & 1 & 1 \end{array}$$

We put 1's ~~for~~ and 0's in the incident matrix, depending on whether an edge is incident on a vertex or not.

Here, the row sum = degree of the vertex and the column sum = 2 (because an edge has to be incident on exactly 2 vertices)

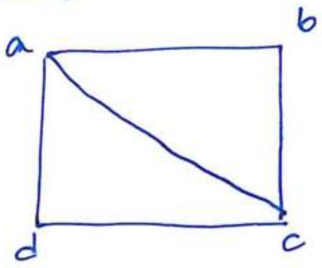
 Ph: 844-844-0102

Note: In case of self-loops, we will put value 2 in the incidence matrix.

In case of multi-edges, the incidence matrix will have multiple columns that are identical.

Note:
Incidence matrix for directed graph doesnot capture the direction information. Therefore for directed graphs we shouldnot use an incidence matrix.

③ Adjacency List:



For each vertex, we create a list of adjacent vertices

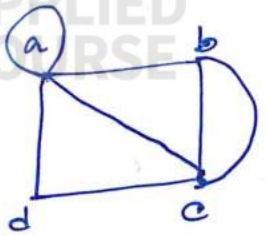
	list of adj. vertices
a	b, c, d
b	a, c
c	a, b, d
d	a, c

we can store this list using lists and sets (depending on the programming language)

we can represent self-loops using adjacency list.

For multi-graphs having multi-edges, we need to use multi-sets because, it allow duplicates while sets doesnot allow duplicate elements.

Mail: gatecse@appliedcourse.com

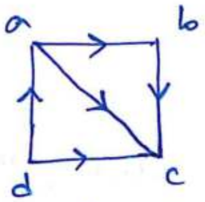


self loop

	list of adjacent vertices
a	a , b, c, d
b	a , c , c
c	a , b, d , b
d	a , c

multi-set (for storing multiple edges)

We can also represent directed graphs using adjacency lists.



	list of adjacent vertices
a	b , c
b	c
c	
d	a , c

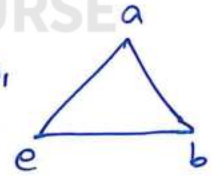
⊛ Set - Theoretic operations on Graphs:

In set theory, we have come across operations like Union, Intersection, complement, XOR and subtraction.

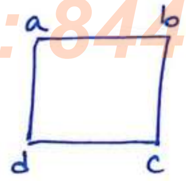
A graph is also two sets i.e, $G = (V, E)$ where V is the vertex set and E is the edge set.

Union : G_1

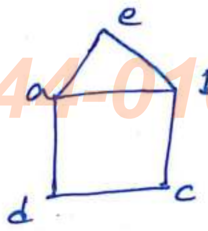
E.g. 1



$\cup$ G_2

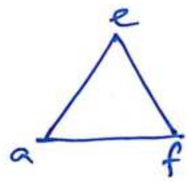


=

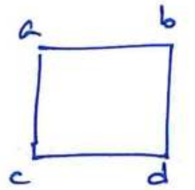


$$\boxed{\begin{aligned} V(G_1 \cup G_2) &= V(G_1) \cup V(G_2) \\ E(G_1 \cup G_2) &= E(G_1) \cup E(G_2) \end{aligned}}$$

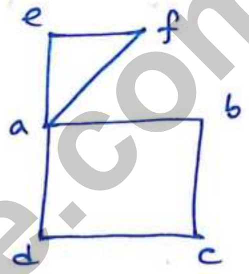
E.g. 2



$\cup$



=



Note: The graphs given have to be labelled so that union operation can be performed.

The union of two graphs is defined as:

(i) vertex set of the union is the union of the vertex sets ie, $V(G_1 \cup G_2) = V(G_1) \cup V(G_2)$.

(ii) Similarly, $E(G_1 \cup G_2) = E(G_1) \cup E(G_2)$

The principle of inclusion - exclusion (set theory) can be applied on graphs as graphs are formed using two sets, ie, vertex set and edge set.

$$|A \cup B| = |A| + |B| - |A \cap B|$$

Therefore, $n(G_1 \cup G_2) = n(G_1) + n(G_2) - n(G_1 \cap G_2)$

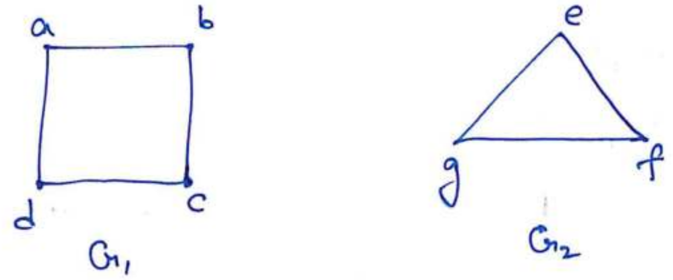
and, $e(G_1 \cup G_2) = e(G_1) + e(G_2) - e(G_1 \cap G_2)$

where $n(G) =$ #vertices in graph G

$e(G) =$ # edges in graph G.

Properties:

1. **Vertex disjoint**: The vertex sets of both the graphs are disjoint.



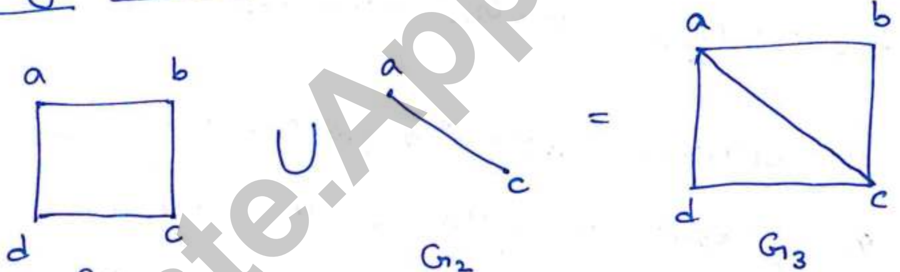
G_1 and G_2 are $\underbrace{\text{connected}}_{\exists\, v_i \rightsquigarrow v_j \ \forall i,j}$ but $G_1 \cup G_2$ is disconnected

path

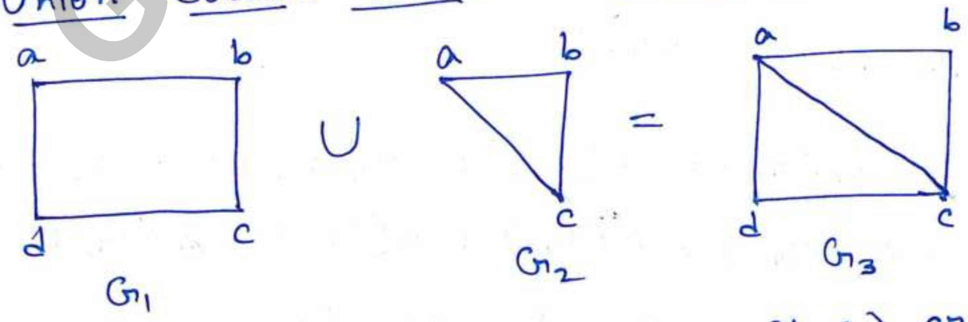
$\Longleftrightarrow$ $V(G_1)$ and $V(G_2)$ are disjoint sets.

(Bidirectional implication)

2. **Edge disjoint**: The edge sets of both graphs are disjoint.



3. **Union doesnot create multigraphs**:

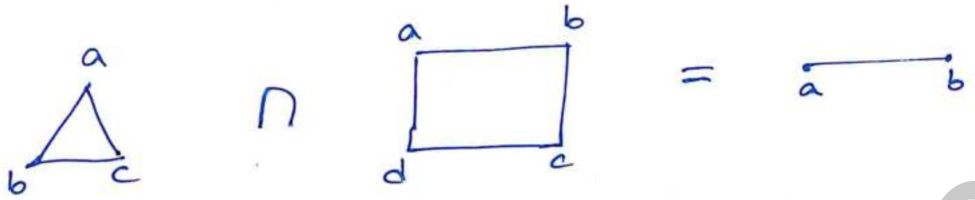


Even though edges (a, b) and (b, c) are present in graph G_1, the union of $(G_1 \cup G_2)$ doesnot result into multiple edges between vertices a and b, and vertices b and c.

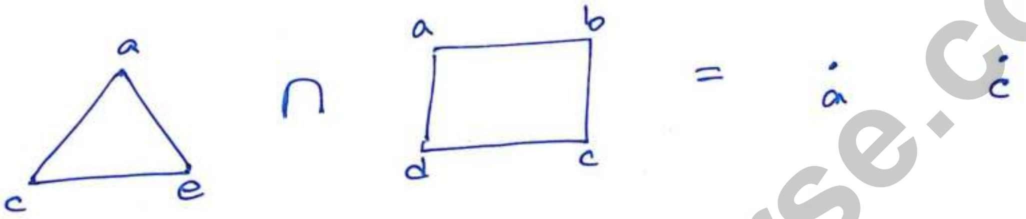
Intersection : $G_1 \cap G_2$

$$V(G_1 \cap G_2) = V(G_1) \cap V(G_2)$$
$$E(G_1 \cap G_2) = E(G_1) \cap E(G_2)$$

E.g ①



E.g ②



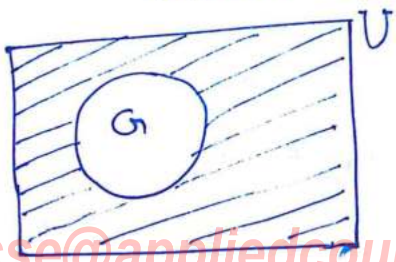
Properties :

1. $(V_1 \cap V_2) = \phi \iff (G_1 \cap G_2)$ has no vertices
   where $V(G_1) = V_1$ and $V(G_2) = V_2$

2. $(G_1 \cap G_2)$ is connected iff
   ⓐ $(V_1 \cap V_2)$ is a singleton set or is a single vertex.

   ⓑ If $|V_1 \cap V_2| > 1$ then
   $(E_1 \cap E_2)$ should lead to a connected graph on $(V_1 \cap V_2)$.

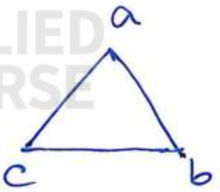
Complement : Here we look at the edge complements. Universe of vertices make no sense.



U = universal set
= all edges in a complete graph.

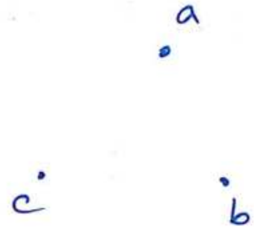
Eg. ①

Let G:



$= K_3$

$\overline{G} = G' =$

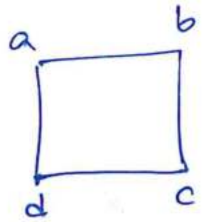


$= \phi_3$

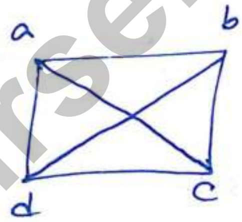
$$\boxed{\overline{G} = K_n - G}$$

E.g. ②

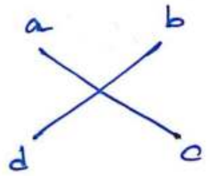
G:



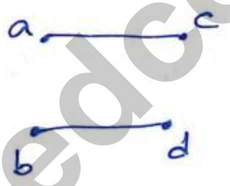
$K_4 =$



$\overline{G}$:



$=$



Properties:

① $e(\overline{G}) = \frac{n(n-1)}{2} - e(G)$

where,

edges in G-complement $= e(\overline{G})$

② $G, \overline{G}$: Both have same vertex set.

But, $E(G) \cap E(\overline{G}) = \phi$.

③ $\overline{\phi_n} = K_n$; $\overline{K_n} = \phi$

④ If $\left.\begin{array}{l} G_1 \cup G_2 = K_n \\ G_1 \cap G_2 = \phi \end{array}\right\} \overset{\text{IFF}}{\Longleftrightarrow} \begin{array}{l} G_1 = \overline{G_2} \\ G_2 = \overline{G_1} \end{array}$

(Q) $O(G) = 5 = n(G)$

$e(G) = 7$

$\therefore O(\bar{G}) = O(G) = 5$

And, $e(\bar{G}) = e(K_n) - e(G)$

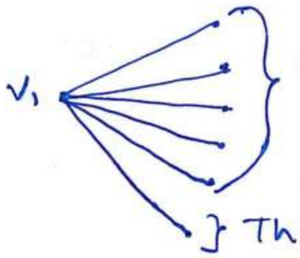
$= \frac{5 \times \cancel{4}^{2}}{\cancel{2}} - 7$

$= 3$

(Q) Degree sequence of $G = (\overset{v_1}{5}, \overset{v_2}{5}, \overset{v_3}{4}, \overset{v_4}{4}, \overset{v_5}{3}, \overset{v_6}{3}, \overset{v_7}{2})$

$\therefore n(G) = 7.$

Find the degree sequence of $\bar{G}$.



These edges are present in G.

This edge can be present in $\bar{G}$.

Every vertex $v \in V$ in $G(V, E)$ can connect to atmost $(n(G) - 1)$ vertices in order to form edges.

The $\bar{G}(V, \bar{E})$ will have a degree sequence of: $(x_1, x_2, x_3, x_4, x_5, x_6, x_7)$

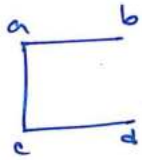
$\Rightarrow (v_1 + x_1 = n(G) - 1,\ v_2 + x_2 = n(G) - 1,\ v_3 + x_3 = n(G) - 1,\ v_4 + x_4 = n(G) - 1,\ v_5 + x_5 = n(G) - 1,\ v_6 + x_6 = n(G) - 1,\ v_7 + x_7 = n(G) - 1)$

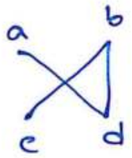
$\Rightarrow (x_1 = 1,\ x_2 = 1,\ x_3 = 2,\ x_4 = 2,\ x_5 = 3,\ x_6 = 3,\ x_7 = 4)$

$\Rightarrow (1, 1, 2, 2, 3, 3, 4)$

⑤ Complement of a connected graph may / may not connected.

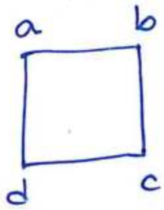
E.g. ①

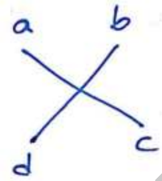
G : 

$\overline{G}$: 

Here G is connected and $\overline{G}$ is also connected.

E.g. ②

G : 

$\overline{G}$: 

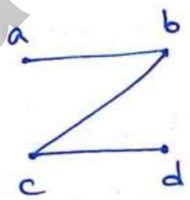
Here G is connected but $\overline{G}$ is not connected.

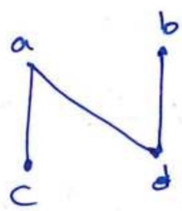
⑥ Complement of a disconnected graph is connected.

If G is disconnected, $\overline{G}$ is connected. This is because $\boxed{G \cup \overline{G} = K_n}$

Self-complementary Graphs :

E.g. ①

$G =$ 

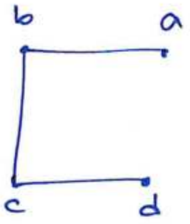
$\overline{G} =$ 

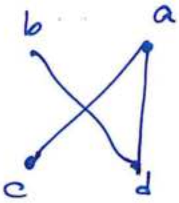
$$G \neq \overline{G}$$

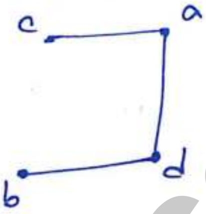
If we ignore vertex-labels, the graphs are same, such graphs are called isomorphic graphs, i.e., $G \cong \overline{G}$.

If a graph and its complementary graph are isomorphic to each other, then the graph is a self-complementary graph.

E.g ②

$G =$ 

$\bar{G} =$ 

$=$ 

If we ignore the vertex labels, then $G \equiv \bar{G}$.

$G \equiv \bar{G} \iff G$ is self-complementary.

⑦ If G is self-complementary, then

$n(G) = n(\bar{G})$

$e(G) = e(\bar{G})$

⑧ ϕ_n, K_n are not self complementary except when $n = 1$.

$G = \cdot$ $\qquad$ $\bar{G} = \cdot$

Null graph & complete graph

⑨ G is self complementary $\Rightarrow n(G) = 4x$ or $n(G) = 4x+1$ where $x \in$ Integers

$\Rightarrow e(G) = \frac{n(n-1)}{4} =$ Integers

$\Rightarrow$ diameter is 2 or 3.

E.g ③ Given $n(G) = 6$, can such graph be even self-complementary?

$n(G) = 6$

$\therefore e(G)$ should be $= \frac{n(n-1)}{4}$ edges

$$= \frac{6 \times 5}{4}$$

$$= \frac{30}{4}$$

$$= \frac{15}{2} \neq \text{Integer}.$$

$\therefore$ we cannot have a self-complementary graph with 6 vertices.

Proof:

We know,

$$\boxed{e(G) = e(\bar{G})}$$

There are no common edges between G and $\bar{G}$.

$G \cup \bar{G} = K_n$

$G \cap \bar{G} = \phi_n$

$\therefore \boxed{e(G) + e(\bar{G}) = K_n = \frac{n(n-1)}{2}}$

Thus, $e(G) = e(\bar{G}) = \left(\frac{n(n-1)}{2}\right)/2 = \frac{n(n-1)}{4} = \text{Integer}$

$\because \frac{n(n-1)}{4} = \text{Integer}$,

therefore, either n is divisible by 4 or, $(n-1)$ is divisible by 4.

Let, $\frac{n(n-1)}{4} = x$; where x is an integer.

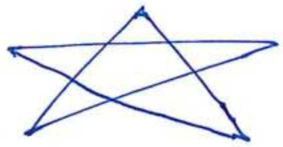
$\therefore \underbrace{n = 4x}_{}$ or $n - 1 = 4x$

$\Rightarrow \underbrace{n = 4x + 1}_{}$

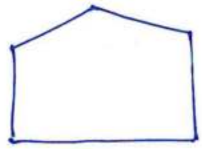
n is divisible by 4 with 0 remainder

n is divisible by 4 with 1 remainder

E.g. 4



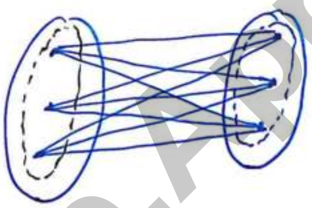
$\equiv$



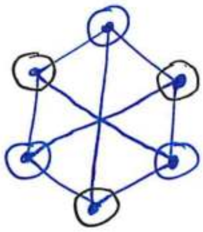
G $\qquad$ $\overline{G}$

The graph is self-complementary because the graph and its complement are isomorphic to each other.

E.g. 5



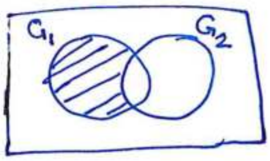
$\equiv$



$K_{3,3}$ $\qquad$ $K_{3,3}$.

Subtraction :

$G_1 - G_2 = G_1 \cap \overline{G_2}$



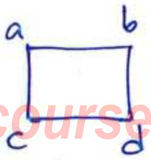
$G_1 - G_2$

$V(G_1 - G_2) = V(G_1)$

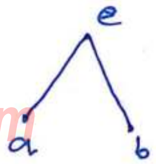
$E(G_1 - G_2) = E(G_1) - E(G_2)$

E.g. 1

$-$

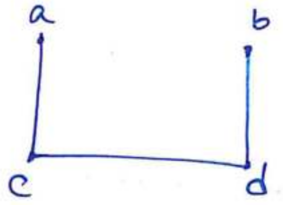


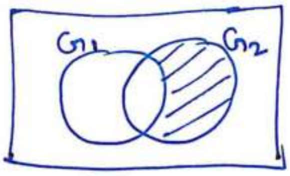
$=$



G_1 $\qquad$ G_2 $\qquad$ $G_1 - G_2$

$G_2 - G_1 =$

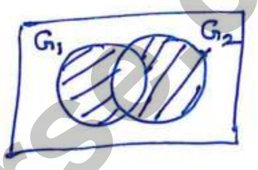




$G_2 - G_1$

XOR operation (Ring-sum) $\oplus$:

Present in one but not both.



$$G_1 \oplus G_2 = (G_1 - G_2) \cup (G_2 - G_1)$$

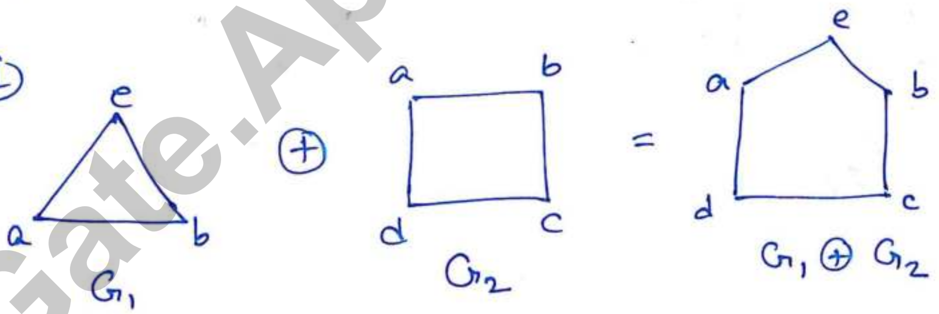
$$= (G_1 \cup G_2) - (G_1 \cap G_2)$$

In terms of vertex sets and edge sets :

$$V(G_1 \oplus G_2) = V(G_1) \cup V(G_2)$$

$$E(G_1 \oplus G_2) = E(G_1) \oplus E(G_2)$$

E.g. 1



The edge (a,b) is present in both G_1 and G_2, therefore edge (a,b) is not included in $(G_1 \oplus G_2)$. While other edges are present in only one of the graphs, therefore they are present in $(G_1 \oplus G_2)$.

Note: The vertex set of XOR will include all the vertices from both the graphs.

Q. $n(G_1) = 10$ $\quad e(G_1) = 8$

$n(G_2) = 5$ $\quad e(G_2) = 3$

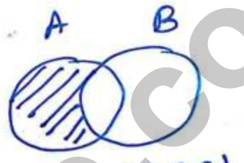
$e(G_1 \cap G_2) = 2$ $\quad n(G_1 \cap G_2) = 3$

① $V(G_1 - G_2) = V(G_1)$

$\therefore n(G_1 - G_2) = n(G) = 10$

② $e(G_1 - G_2) = ?$

$E(G_1 - G_2) = \overset{A}{E(G_1)} - \overset{B}{E(G_2)}$



$|A| - |A \cap B|$

$e(G_1 - G_2) = 8 - 2 = 6$

$= e(G_1) - e(G_1 \cap G_2)$

③ $n(G_1 \oplus G_2) = ?$

$V(G_1 \oplus G_2) = \underbrace{V(G_1)}_{A} \cup \underbrace{V(G_2)}_{B}$

$|A| = 10$ & $|B| = 5$ (given)

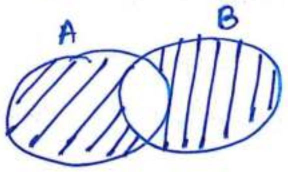
$|A \cup B| = |A| + |B| - |A \cap B|$

$= 10 + 5 - 3 = 12$

$\therefore n(G_1 \oplus G_2) = 12$

④ $e(G_1 \oplus G_2) = ?$

$\left| \underbrace{E(G_1)}_{A} \oplus \underbrace{E(G_2)}_{B} \right|$



$e(G_1 \oplus G_2) = e(G_1) + e(G_2) - 2(e(G_1 \cap G_2))$

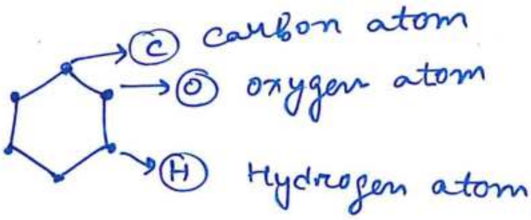
$= 8 + 3 - 2(2)$

$= 7.$

(*) Graph Isomorphisms :

Any chemical compound (organic or inorganic molecule can be represented as a graph.

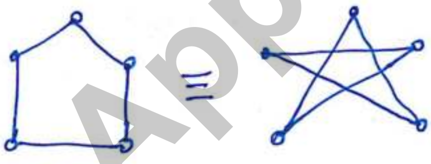
E.g.,



The connections between the atoms are the edges and the atoms are the vertices in the graph.

In organic chemistry, there are very lengthy and complex chemical compounds which can be represented by graph G_1. The same chemical compound maybe represented as graph G_2 by some one else.

E.g.,



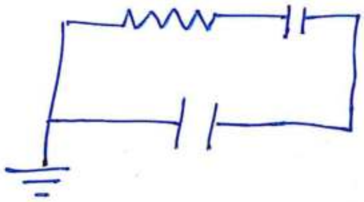
G_1 $\equiv$ G_2

Here G_1 and G_2 are representing the same chemical compound.

In computational chemistry, an area of chemistry, where chemists try to find new molecules or new chemical compounds with some interesting properties and they use computer science to determine the new compound easily. They check if at the graph theoretic level, if the two given compounds are one and the same.

Other major example is in VLSI (circuit design for electronics students).

Suppose there is a schematic diagram of an electronic circuit.

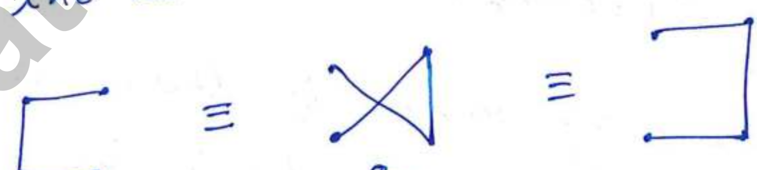


The above schematic diagram can be represented as graph where the resistors, capacitors can be represented as vertices and the connections can be represented as the edges.

In VLSI, circuits can have millions of resistors, capacitors and connections etc. Therefore, given two such circuits we want to determine whether they are one and the same. This is where graph isomorphism comes handy.

Isomorphism:

$G_1 \equiv G_2$ when the underlying unlabelled graphs are the same.

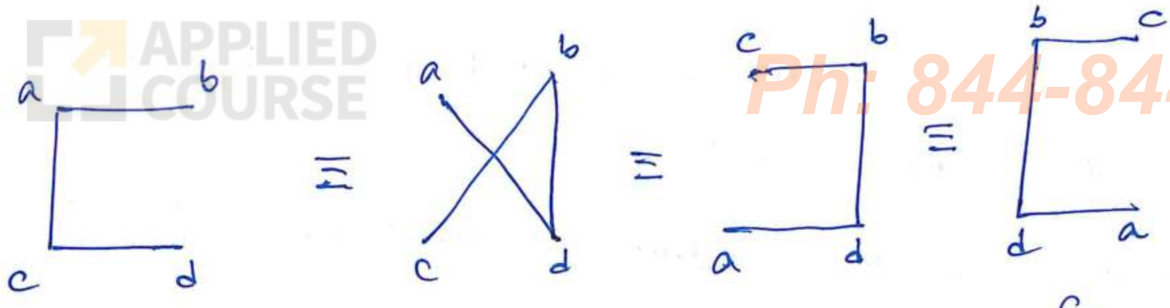


G_2

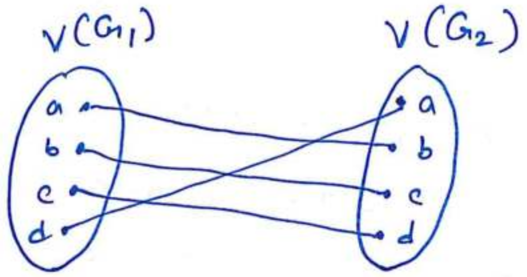
G_1

Another way to think is:

$G_1 \equiv G_2$ iff $\exists$ bijective function $f: V(G_1) \to V(G_2)$ which preserves adjacency.



G_1 G_2 G_2 G_2



$f(a) = b$
$f(b) = c$
$f(c) = d$
$f(d) = a$

For every element in $V(G_2)$, there is a pre-image in $V(G_1)$. Therefore, f is onto.

Given any element in $V(G_1)$, there should be a mapping to exactly one element of $V(G_2)$.

$\therefore f$ is one-one.

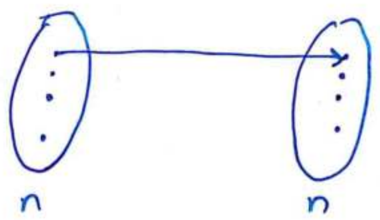
Hence f is bijective.

Adjacency preserving means if there is an edge $a \to b$ in G_1, there should be an edge $f(a) \to f(b)$ in G_2 and vice-versa.

(Q). To test if two graphs are isomorphic.

In algorithmic perspective, we might have to generate all possible bijections and then test if adjacencies are preserved.

From a set of size n to another set of size n, there are $n!$ possible bijections.



$n * (n-1) * (n-2) * \dots 1$
$= n!$

Therefore to test if two graphs are isomorphic, we have to generate all $n!$ bijections. And for each bijection we have to check if the adjacencies are preserved. It is a very time consuming method.

Mathematicians came up with some conditions that have to be met if G_1 and G_2 are isomorphic. If any one of those conditions are not met, the we can say that the two graphs are not isomorphic. These conditions are called invariants.

If $G_1 \equiv G_2$

then ① $n(G_1) = n(G_2)$

Note: This statement is similar to $A \Rightarrow B$, then $\neg B \Rightarrow \neg A$ (in propositional logic)

Using $(\neg B \Rightarrow \neg A)$, we can say that if $n(G_1) \neq n(G_2)$ then G_1 is not isomorphic to G_2.

4

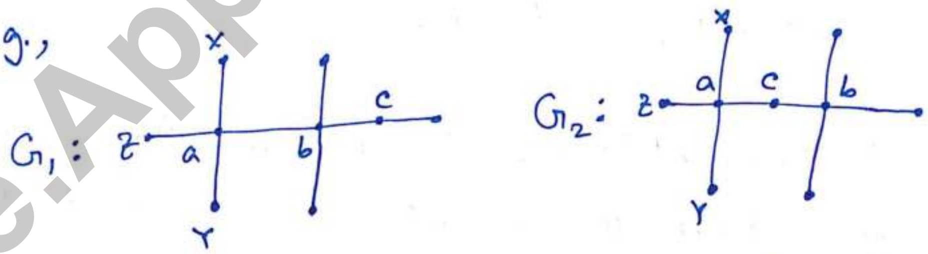
If $G_1 \equiv G_2$
then ② $e(G_1) = e(G_2)$

③ degree sequence G_1 = degree sequence G_2

④ #cycles of any length is same in G_1 and G_2.

⑤ Consider vertex $u \in V(G_1)$ and $v \in V(G_2)$, all the neighboring vertices for u and v should have same properties.

E.g.,



The vertex a in G_1 has a neighboring vertex b with degree 4.
But in G_2, none of neighboring vertices of a, ie, c, x, y, z have degree 4.

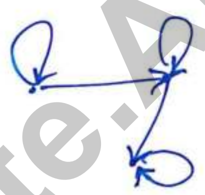
Thus neighboring vertices of a in G_1 and G_2 donot have the same properties. Hence $G_1 \not\equiv G_2$.

E.g.,
Given a directed graph represented as adjacency matrix. Symmetric matrix represents undirected graph.

$$\begin{array}{c|ccc} & v_1 & v_2 & v_3 \\ \hline v_1 & 1 & 1 & 0 \\ v_2 & 0 & 1 & 1 \\ v_3 & 0 & 0 & 1 \end{array} \overset{?}{\equiv} \begin{array}{c|ccc} & v_1 & v_2 & v_3 \\ \hline v_1 & 0 & 1 & 0 \\ v_2 & 0 & 1 & 0 \\ v_3 & 0 & 1 & 0 \end{array}$$

G_1 $\qquad$ G_2

The above graphs are not symmetric matrices, therefore these are directed graphs.
Vertex v_1 in G_1 has a degree 2 while there are no vertices in G_2 that has degree 2.
Therefore, these graphs are not isomorphic.
We can also draw the graphs and verify



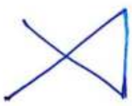
$\not\equiv$

G_1 $\qquad$ G_2

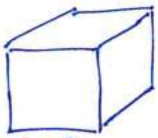
Common Isomorphisms :

Eg. ①

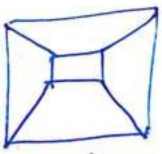
$\equiv$



Eg②

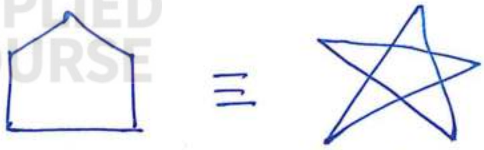


$\equiv$



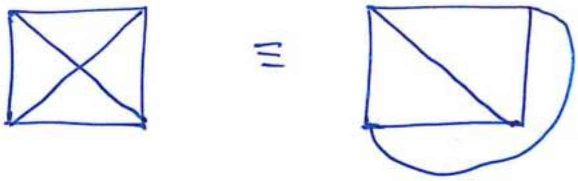
Q_3

Eg ③

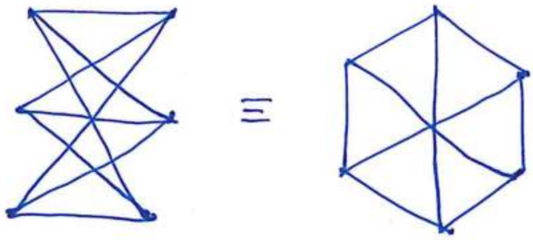


(pentagram)

Eg ④

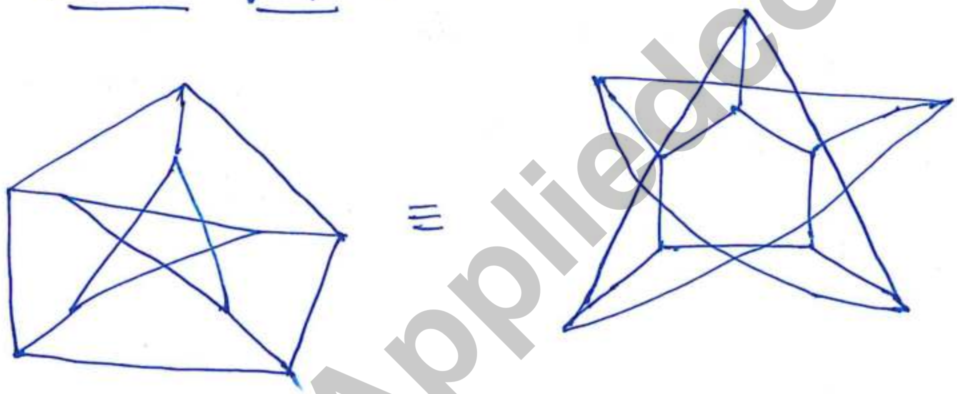


Eg ⑤



$K_{3,3}$

Eg ⑥ Peterson graph:



⊛ Connectivity of Graphs:

Connectivity is a very important topic in the real world. If we think of a road graph, we want connectivity from every town to all the other towns. In computer networks, there should be a path between any computer and a server, using which we can send our data to the server and get it back from the server.

computer

server

Connected graph: A graph G is connected iff

$$\forall\, v_i, v_j \in V(G)$$
$$\exists\, v_i \rightsquigarrow v_j$$

That is, a graph G is said to be connected if and only if for all vertices v_i, v_j that belongs to the vertex set $V(G)$, there exists a path from v_i to v_j.

If the graph is directed then the path will be a directed path. If the graph is undirected then the path will be undirected path.

In the algorithms textbook – 'Introduction to Algorithms by CLRS', the following notations have been used :

undirected path

directed path

Smallest connected graph:

If we take a null graph with one vertex (ϕ_1), it is a connected graph.

a

vertex a is connected to itself without any loop.
Therefore, it is the smallest connected graph.

Ph: 844-844-0102

Some interesting theorems :

1. In any graph G, which is connected / disconnected if there exists exactly 2 vertices of odd degree, x & y, then there exists a path from x to y.

Theorem: In any connected / disconnected G, if $\exists$ exactly 2 vertices x & y of odd degree then $\exists$ x ⟿ y.

Proof:

Case I: G is connected.

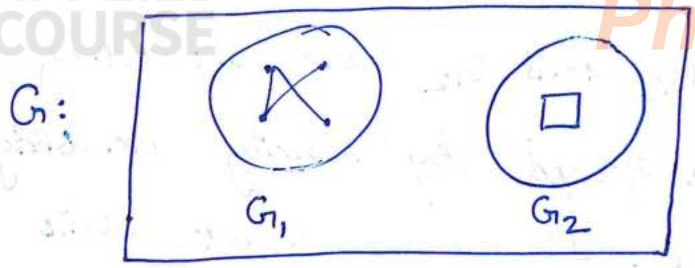
G has even number of odd vertices. Therefore it is a graph.

Given any two vertices x and y since G is connected, there is a path from x to y.

Case II: G is disconnected.

There is a related concept called connected components. Given any graph, if we can break it into subgraphs, such that, the subgraphs are connected but there exists no edge connecting vertices v_1 and v_2 where v_1 belongs to one subgraph and v_2 belongs to another subgraph. These subgraphs are called connected components.

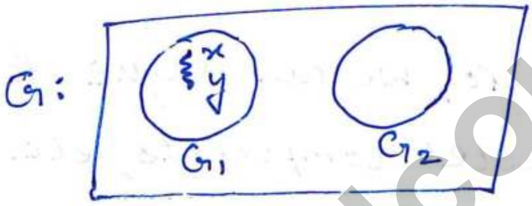
Mail: gatecse@appliedcourse.com



Here: G_1 and G_2 are connected components. But G is a disconnected graph.

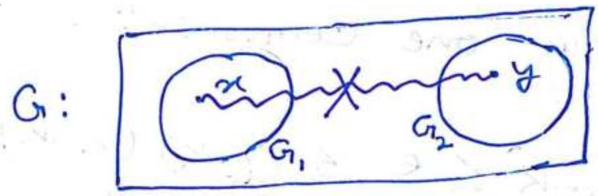
∴ If the graph G is disconnected, there are atleast two disconnected components in the graph.

<u>case 2a:</u> vertices x and y belong to the same connected component, i.e., $x \,\&\, y \in G_1$



If vertices x and y belong to the same connected component then there will be a path between x and y as G_1 is itself a connected subgraph.

<u>case 2b:</u>



Given that there are exactly two odd degree vertices in graph G. If we assume one such odd degree vertex is in G_1 and the other is in G_2. In that case, there cannot be a path from vertex x and vertex y since both G_1 and G_2 are connected components. Due to degree 1, these vertices will only be reachable within their components. Also, the connected components G_1 and G_2 are violating the property of a graph by having an odd number of vertex with odd degree. It means that this case is never possible.

So, if case 2b is possible, then G_1 and G_2 are not graphs. If G_1 and G_2 are not graphs, then they can never be connected components. It means, even if G is disconnected, we cannot construct connected components as G_1 and G_2.

Using the same logic, we can argue for 3 connected components, 4 connected components, etc.

2. Let us assume we have e edges and n vertices. Let us assume we have k components in a graph.

K can be 1. $\boxed{k=1 \iff G \text{ is connected}}$

If $k=1$ then G is connected. If G is connected, there is only one component.

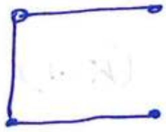
<u>Theorem:</u> $n-k \le e \le \frac{(n-k)(n-k+1)}{2}$

<u>Proof:</u> Let's consider the case when $k=1$, ie, it is a connected graph.

$\therefore \quad n-1 \le e \le \frac{(n-1)(n)}{2}$

∴ For a connected graph with n vertices, we need atleast $(n-1)$ edges if the graph has to be connected.

E.g., $n=4$

G: 

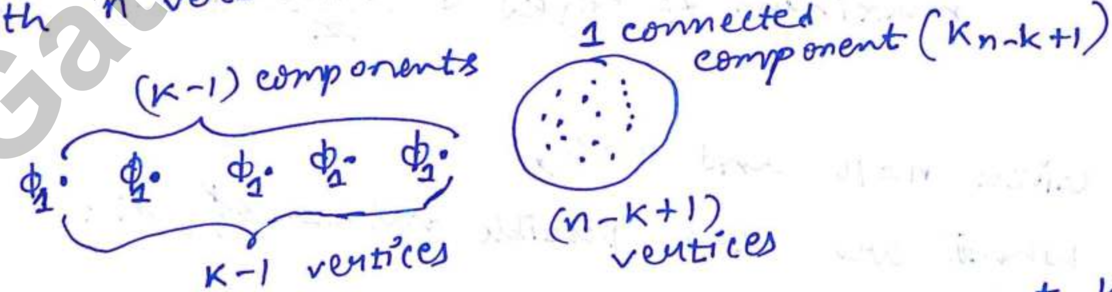
Graph G with $(n-1)$ edges $= (4-1) = 3$ edges is a connected graph.

Now, $\frac{(n-1)n}{2}$ is the number of edges in a complete graph.

Note: We are looking only at the simple graphs, no multi-graphs, no self loops.

∴ Number of edges in a simple graph has to be less than or equal to number of edges in the complete graph K_n.

Let's try to construct a K-component graph with n vertices



Let's imagine that the single component with $(n-k+1)$ vertices is completely connected, i.e., it is a complete graph.

∴ It is K_{n-k+1} graph.

If we have a component which is K_{n-k+1}, then the component has $\frac{(n-k)(n-k+1)}{2}$ edges as the other $(K-1)$ components are each fore with 1 vertex each and has no edges. Therefore we have $(K-1)$ null graphs (ϕ_1).

This is the optimal way to construct a k-component graph with n vertices to maximize the number of edges. If we break the graph in any other way, the number of edges will be fewer than the above graph construction.

(Q) Given $n=10$; $k=3$,

what is the minimum number of edges? And what is the maximum number of edges?

We know, $e \geq n-k$

$\therefore$ minimum # edges $= 10-3 = 7$.

maximum # edges $= \frac{7 \times \cancel{8}^{4}}{\cancel{2}} = 8$

(Q.) Given $n=10$ and $e=6$,

what are the possible values of k?

We know,

$n-k \leq e$

$\Rightarrow 10-k \leq 6$

$\Rightarrow \boxed{K \geq 4}$

Again we know,

$$6 \le \frac{(10-K)(10-K+1)}{2}$$

$$\Rightarrow 2 \times 6 \le (10-K)(10-K+1)$$

$$\Rightarrow 12 \le (10-K)(11-K)$$

$$\Rightarrow 12 \le K - 21K + 110$$

$$\Rightarrow \boxed{K^2 - 21K + 98 \geq 0}$$ Quadratic equation.

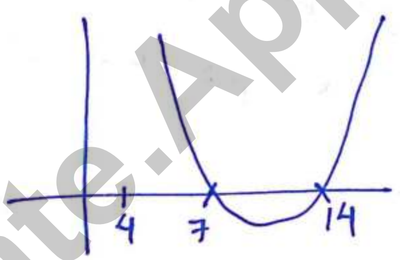
If we solve this for equality, i.e.,

$K^2 - 21K + 98 = 0$,

we get,

$K = 14$ or 7.

The above quadratic equation is actually an equation of parabola.



The equation is (≥ 0) when K is less than 7 and greater than 14.

The other equation, $\boxed{K \geq 4}$

∴ The possible values of $K \geq 4$, $K \le 7$ and $K \geq 14$.

Now since $n = 10$, the number of components can never exceed the number of vertices.

$\therefore$ The possible values of $k = 4, 5, 6, 7$.

(Q) What is the maximum number of edges in a disconnected graph G with n vertices?

Disconnected graph means $\boxed{k \geq 2}$.

We know,

$$\boxed{(n-k) \leq e \leq \frac{(n-k)(n-k+1)}{2}}$$

To maximize the number of edges, we need the smallest possible value of k, which is 2.

$$\therefore \quad e \leq \frac{(n-2)(n-2+1)}{2}$$

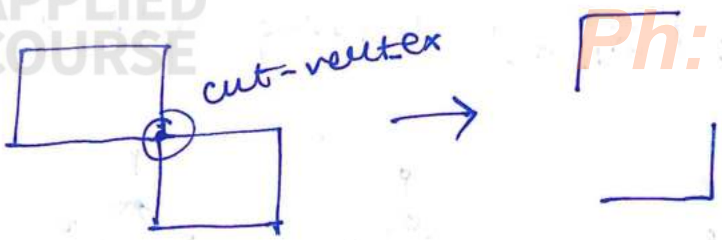
$$\Rightarrow e \leq \frac{(n-2)(n-1)}{2}$$

$\therefore$ For a disconnected graph with n vertices, the maximum possible edges we can have is $\frac{(n-2)(n-1)}{2}$.

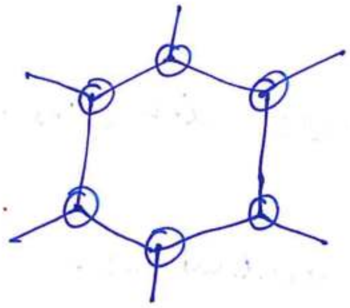
(*) Cut - Vertex :

For $v \in$ connected G such that removal of v disconnects G, such a vertex is called a cut vertex.

Eg:



Eg:

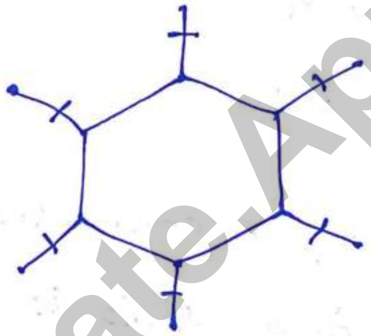


This graph has 6 cut-vertices

⊛ Cut-edge :

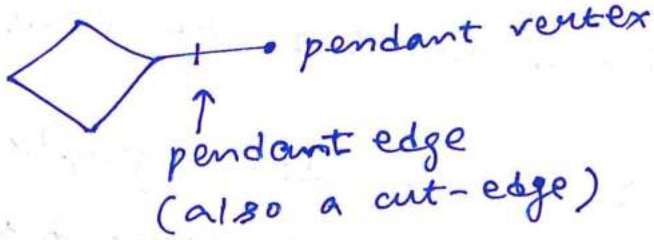
For edge e ∈ connected G s.t the removal of e disconnects G, such an edge is called a cut-edge.

E.g.

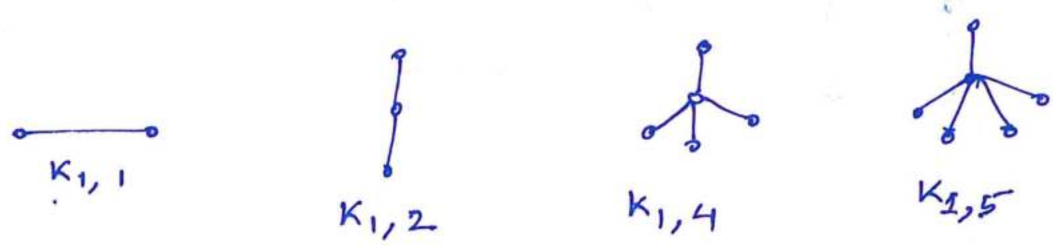


This graph has 6 cut-edges.

E.g.

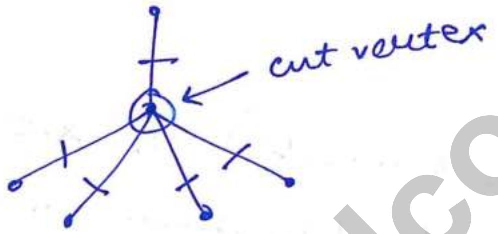


E.g. Star graph : $(K_{1,n})$ having $(n+1)$ vertices



How many cut-edges are there in $(K_{1,n})$?

For $K_{1,5}$, there are n cutedges.



$K_{1,5}$ has 1 cut-vertex.

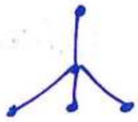
$\therefore$ # cut edges for $K_{1,n} = n$

cut vertices for $K_{1,n} = 1$

On removal of cut vertex, n connected components are formed. And this is the maximum number of components.

In computer science, star graphs are useful in distributed systems where every message comes to a central server and then it goes to the other places. But if the central server is itself down, then everything gets disconnected. The star network is based on the concept of star graph.

Also, diam $(K_{1,n}) = 2$ if $n+1 \geq 3$



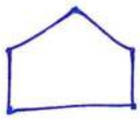
$K_{1,4}$

For diam $(K_{1,1}) = 1$

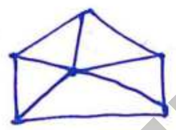
$K_{1,1}$

<u>Note:</u> For C_n, W_n and K_n there are no cut-vertices.

C_n :



W_n :



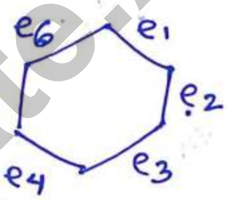
K_n :

Because if we remove any vertex, the remaining graph is still connected.

(*) <u>Cut-set of a connected G</u> :

It is a minimal set of edges whose removal disconnects G.

G :



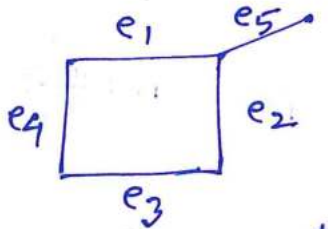
} Edge-labelled graph

$\{e_1, e_2\}$, $\{e_2, e_3\}$, $\{e_3, e_4\}$, $\{e_4, e_5\}$, $\{e_5, e_6\}$, $\{e_6, e_1\}$ are the minimal sets whose removal will disconnect the graph G.

Now, $\{e_1, e_2, e_3\}$ is not a cut set because it is not the minimal set as $\{e_1, e_2\}$ or $\{e_2, e_3\}$ exists which are strict or proper subsets of $\{e_1, e_2, e_3\}$ and $\{e_1, e_2\}$ & $\{e_2, e_3\}$

are themselves cutsets.

(Q) Given



which is not a cutset?

(a) $\{e_5\}$ $\longrightarrow$ 1 edge cutset

(b) $\{e_1, e_5\}$ $\longrightarrow$ Not a cutset

(c) $\{e_1, e_2\}$ $\longrightarrow$ 2 edge cutset

(d) $\{e_3, e_4\}$ $\longrightarrow$ 2 edge cutset.

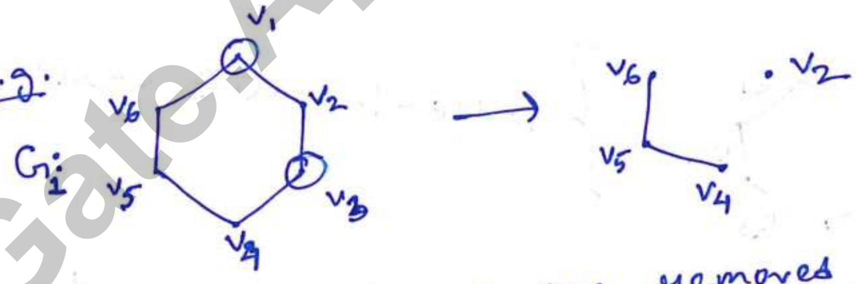
Connectivity-2

(*) vertex connectivity ($\alpha(G)$ or α):

It is the minimum number of vertices to be removed to disconnect a graph G.

If there exists a cut vertex, then $\alpha(G) = 1$

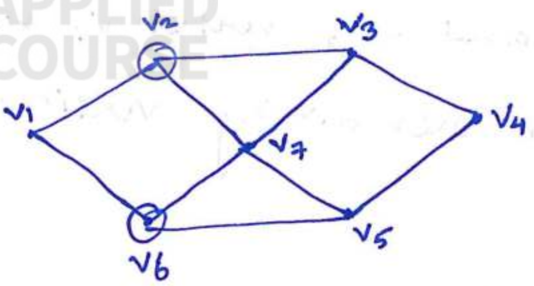
E.g:



If vertices v_1 and v_3 are removed, then there is no path from v_2 to the remaining vertices. ∴ For graph G_1, vertex connectivity or $\alpha(G) = 2$.

E.g.

G_2:



In the above graph G_2,

$$\kappa(G) \le \delta(G)$$

where $\delta(G)$ = minimum of the vertices degrees

If we notice, v_1 has degree 2. And $\kappa(G) = 2$ in this graph as removal of vertices v_2 and v_6 will disconnect v_1 from the remaining vertices.

From average degree theorem :-

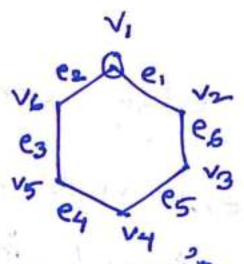
$$\kappa \le \overbrace{\underset{\substack{\uparrow \\ \text{min.} \\ \text{degree}}}{\delta} \le \frac{2e}{n} \le \underset{\substack{\uparrow \\ \text{max. degree}}}{\Delta}}$$

⊛ Edge-connectivity (λ) :-

It is the minimum number of edges to be removed to disconnect G.

It is also equal to the size of the smallest cut-set.

E.g. G_1:



In G_1, the min. degree (δ) and the max. degree (Δ) are the same

$\delta = \Delta = 2$.

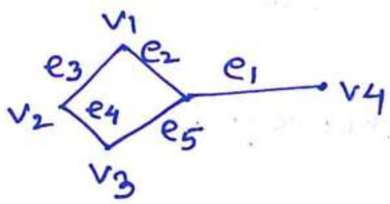
If we remove edges 1 and 2, vertex v_1 will get disconnected from the remaining vertices.

$\therefore \lambda(G) \leq \delta$

Since by removing all the edges connected to the vertex with minimum degree, we can disconnect the vertex with minimum degree.

Now, for every vertex that we remove, we also have to remove atleast one edge.

E.g.



If we remove vertex v_4, we have to remove edge e_1

If we remove vertex v_1, we have to remove edges e_2 and e_3.

$\therefore \alpha \leq \lambda$

Therefore,

$$\alpha \leq \lambda \leq \delta \leq \frac{2e}{n} \leq \Delta$$

Application of vertex and edge connectivity:

The telecommunications network or computer network are designed in such a way that the vertex connectivity is as high as possible since vertex connectivity tells us how many cell towers have to go down for the telecommunications network to become a disconnected network. This information is very useful in cases like tsunamis, earthquakes, etc.

And edge connectivity tells us how many of the connections (wired / wireless) can go down the network gets disconnected.

(Q) $n = 10$, $e = 16$

What are the possible values of α and λ?

$$\alpha \le \lambda \le \delta \le \frac{2 \cdot e}{n}$$

$$\frac{2 \cdot e}{n} = \frac{2 \times 16}{10} = 3.2$$

$\therefore \boxed{\alpha \le \lambda \le 3}$ [Since α and λ has to be integers]

Definition: A graph G is separable iff $\alpha = 1$

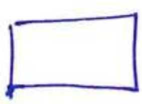
or

G is separable iff it is 1-connected

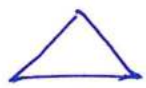
or

G is seperable iff it is a cut-vertex

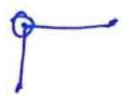
E.g.



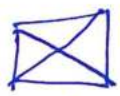
Not seperable



Not seperable



seperable



Not seperable

Definition :

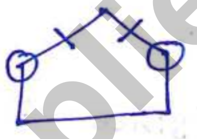
A graph G is called n-connected iff its vertex connectivity is equal to n.

n-connected G $\Leftrightarrow$ $\alpha = n$

Similarly, a graph G is called k-line connected graph iff its edge connectivity is equal to K.

k-line connected G $\Leftrightarrow$ $\lambda = K$

For cycle graph (C_n), $\alpha = 2$ and $\lambda = 2$.



For complete graph (K_n),

$\lambda = n-1$

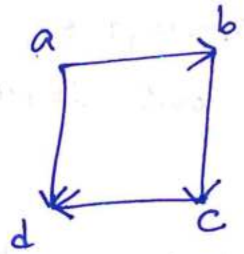
α = Not defined because whatever number of vertices we remove, it will still be connected.

(*) Directed graphs:

- disconnected
- connectivity
  - unilateral
  - strongly
  - weakly

Unilaterally connected graph: It is a graph where for all vertices x and y, there is directed path from $x \rightsquigarrow y$ or there is a directed path from $y \rightsquigarrow x$.

E.g.,



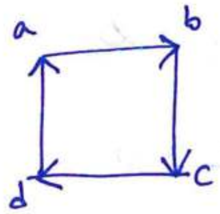
$c \not\rightsquigarrow a$
$a \rightsquigarrow c$

$\therefore \ \forall x, y, \ x \rightsquigarrow y$ or $y \rightsquigarrow x$

Strongly connected graph:

$\forall x, y, \ x \rightsquigarrow y$ AND $y \rightsquigarrow x$

E.g.



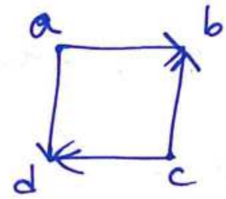
From every vertex to every other vertex, there is a path.

Weakly connected graph:

The underlying undirected graph is connected.

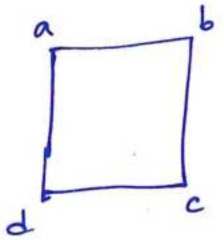
E.g.:

G_1:



Here vertices b and d are sink ie, it doesnot have any outdegree, it only has indegree.
And vertices a and c are source as they donot have any in-degree, they only have out-degree.

G_2:



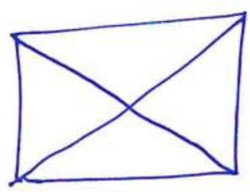
underlying undirected graph.
is not unilaterally connected graph as $a \not\rightsquigarrow c$ and $c \not\rightsquigarrow a$

Definition

girth (G) = length of the shortest cycle in G.

E.g.: acyclic G $\longrightarrow$ girth = ∞

$K_n \longrightarrow$ girth = 3



K_4

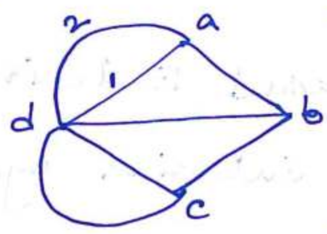
girth (K_4) = 3

Ph: 844-844-0102

⊛ Eulerian and Hamilton Graphs :

The seven bridges of Königsberg problem :

Can we devise a walk such that each bridge is crossed exactly once ?



The map is drawn as a graph where bridges are edges and land is vertex.

Walk : vertex can be repeated
edge can be repeated

a b d a b
1 (say using edge 1)

A walk can be open or closed

In closed walk, the starting and the ending vertex is the same. If we don't start and end at the same vertex, then it is called an open walk.

Trail : vertex can repeat
Edge cannot repeat
→ open
→ closed (circuit)

Path : vertex can't repeat
Edge can't repeat
→ open
→ closed (cycle)

Mail: gatecse@appliedcourse.com

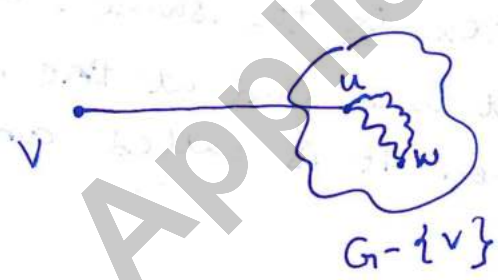
Eulerian Trail : Trail which visits each edge exactly once. It is also called as Eulerian path.

Eulerian circuit : Closed Eulerian trail.
It is also called as Eulerian cycle.

Eulerian Graph :
Definition:
G is Eulerian iff $\exists$ a Eulerian cycle.

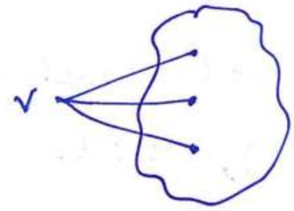
Another definition :

G is Eulerian $\Leftrightarrow$ connected and degree of each vertex is even.

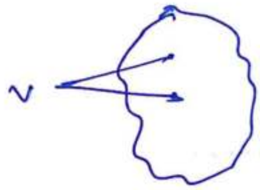


deg(v) = 1

Say we start from vertex v and reach vertex u. Suppose there is are two paths between vertex u and w such that each edge in the connected component in visited exactly using the two paths. On return to u we can't visit vertex v again since the only edge between v and u has already been visited. Therefore, if there exists an odd degree vertex, then the graph can't be Eulerian as Eulerian cycle doesnot exist.



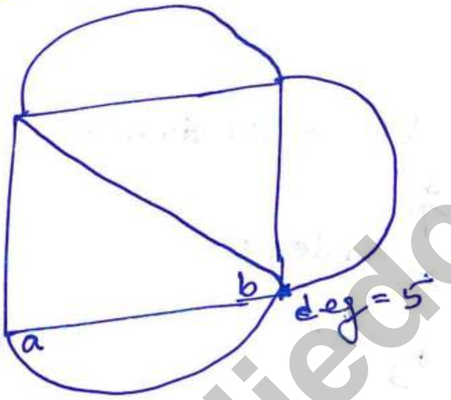
deg(v) = 3 → Not Eulerian graph



deg (v) = 2 → Eulerian graph.

E.g.:

G :



Graph G contains an Euler's path but not an Euler's cycle. Euler cycle is a closed Euler's path. We can start at vertex a and end at vertex b, or vice-versa as both vertices a and b have odd degrees.

Note: In a graph, there may be more than one unique Eulerian cycle.

Note: ϕ_n : not Eulerian because it is not connected to start with.

K-regular graph: If K = even, then the graph is eulerian.

K_n : $n \geq 3$ and n is odd $\longrightarrow$ Eulerian

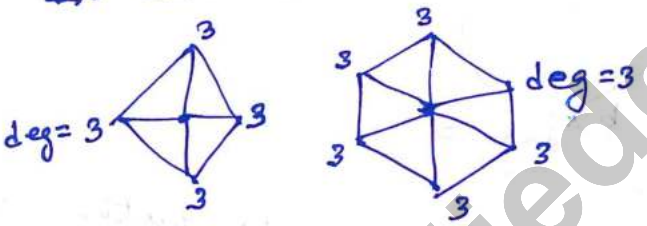
Explanation:

$n \geq 3$ means cycle is present because $n < 3$ means there can't be a cycle.

If n is odd, the degree of each vertex in a complete graph is $(n-1)$, which means $(n-1)$ is even.

C_n : $deg(v) = 2$ = even $\longrightarrow$ Eulerian

W_n : It can never be Eulerian.



As the degree of some vertices are odd, therefore, it is not Eulerian.

Unicursal Graph:

A graph is called an unicursal graph iff it contains Euler path.

OR,

A graph is called an unicursal graph iff there exists only 2 vertices of odd degree and graph has to be connected.

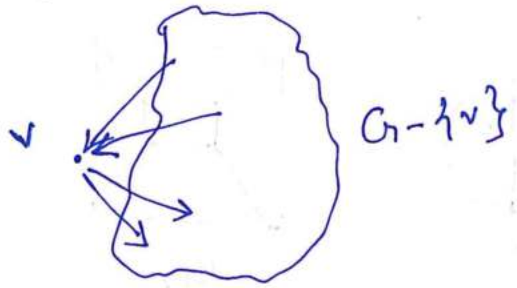
Traversable/Traceable graph:

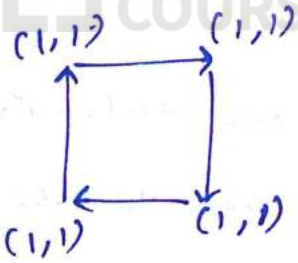
If a graph is either unicursal or eulerian, then is said to be traversable or traceable graph.

Given any graph, if the number of odd degree vertices = 0, ie., all vertices are even degree, then the graph is Eulerian.
If the number of odd degree vertices = 2, then the graph is Unicursal.
Eulerian basically means that a eulerian cycle exists. And unicursal means a eulerian path exists

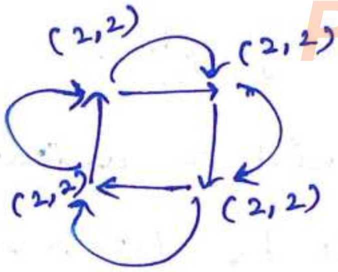
Directed Eulers Graph:

A graph is directed eulerian graph iff there exists a directed eulerian cycle.
Or, a graph is directed eulerian graph iff it is unilaterally connected and $\forall v$, indeg(v) = outdeg (v)





G_1:



G_2

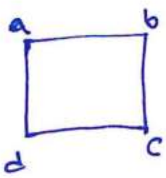
Directed euler graph G_1 and G_2 as they are unilaterally connected and for all vertices the indegree is same as the outdegree.

Note: Eulerian cycle : closed, Trail, each edge is used exactly once.

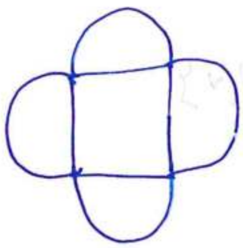
Hamiltonian Path : It is a path with each vertex being visited ~~exactly~~ once.

Hamiltonian cycle : It is a closed Hamiltonian path.

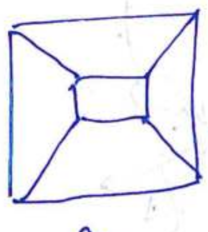
Note: In case of Hamiltonian path and Hamiltonian cycle, all edges may/maynot be traversed but all vertices must be visited.



G_1

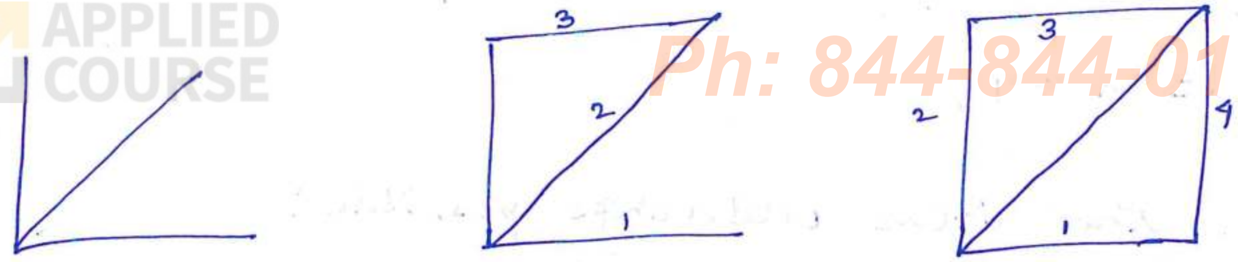


G_2



G_3

The graphs G_1, G_2 and G_3 have Hamiltonian path and Hamiltonian cycle.



No Hamiltonian path

Hamiltonian path exists

Hamiltonian cycle exists

ϕ_n : Not Hamiltonian as it is not connected

G

The Graph G has no edges to traverse between vertices.

K-regular : may or maynot be Hamiltonian.

K_n : $n \geq 3$ is always Hamiltonian

The # Hamiltonian cycles present in a complete graph?

A hamiltonian cycle is basically a circular permutation of the n vertices, as hamiltonian cycles are built using hamiltonian paths, ie, we can't repeat a vertex.

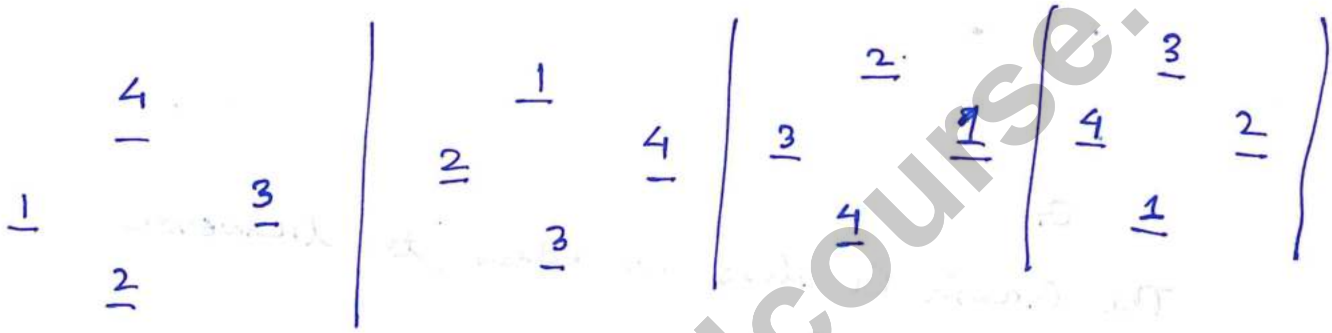
Let $n = 4$,

the linear orderings possible:

$$\underline{4}\ \underline{3}\ \underline{2}\ \underline{1} \longrightarrow 4! = 4 \times 3 \times 2 \times 1$$

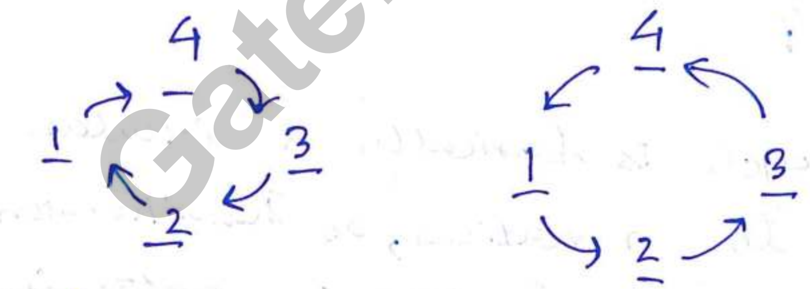
four places

the circular orderings possible:



The above orderings are one and the same.

Now each of these orderings can be written clock-wise and anticlockwise, which are also one and the same.



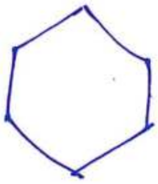
$\therefore$ Total number of Hamiltonian cycles

$$= \left(\frac{4!}{4}\right) / 2$$

$\therefore$ # Hamiltonian cycle = circular permutation of n numbers where clockwise and counter clockwise are the same.

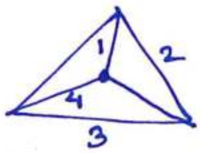
$$= \left(\frac{n!}{n}\right)/2 = \frac{(n-1)!}{2}.$$

C_n:

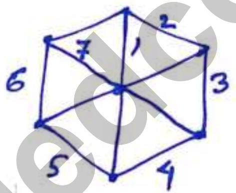


Circular graph is Hamiltonian.

W_n:



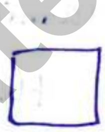
1-2-3-4



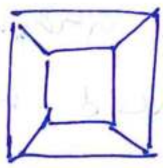
1-2-3-4-5-6-7

Wheel graphs are Hamiltonian.

Q_n:



Q_2



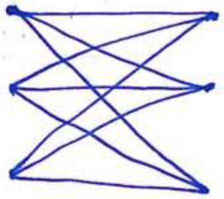
Q_3

Hypercube graphs Q_2 and Q_3 are Hamiltonian.

Note: Hypercube graphs may or maynot be Hamiltonian.

$K_{m,n}$: A complete bipartite graph is Hamiltonian if $m = n$ and $m, n \geq 2$

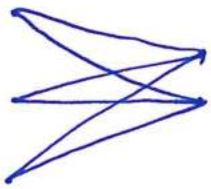
For $m=n$,



G_1

Hamiltonian cycle exists in G_1.

For $m \neq n$,



G_2

Hamiltonian cycle doesnot exist in G_2

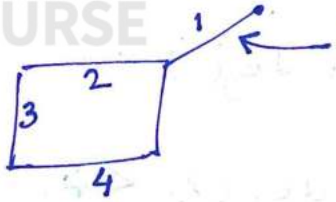
Theorem : If Graph G has Hamiltonian cycle, it implies G has Hamiltonian path.

But if a graph has Hamiltonian path, it need not have Hamiltonian cycle.

If G has Hamiltonian cycle $\Rightarrow$ G has Hamiltonian path

$\not\Leftarrow$

Eg.,



Hamiltonian path exists but Hamiltonian cycle not possible.

If graph is Hamiltonian $\Rightarrow$ No pendant edge
$\not\Leftarrow$

pendant edge $\Rightarrow$ Not a Hamiltonian graph.

$\left[\begin{array}{l} \because A \Rightarrow B \\ \therefore \neg B \Rightarrow \neg A \end{array} \right]$

Theorem:

Dirac Theorem:

If G is connected & $\forall v$, degree$(v) \geq n/2$ and $n \geq 3$ $\Rightarrow$ G is Hamiltonian graph.
$\not\Leftarrow$

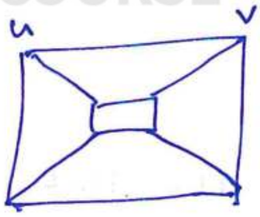
Theorem:

Ore's Theorem:

If G is connected & $\forall (u,v)$ $\deg(u) + \deg(v) \geq n$
$\Rightarrow$ G is Hamiltonian graph.
$\not\Leftarrow$

Note: Every graph that satisfies Dirac Theorem will also satisfy Ore's Theorem. Ore's Theorem is an extension to Dirac Theorem.

Example:



Q_3

$deg(u) = deg(v) = 3$

$deg(u) + deg(v) < n$

The Hypercube graph (Q_3) is Hamiltonian, has 8 vertices.

$\therefore deg(u) + deg(v) < 8$

This example shows all Hamiltonian graphs need not satisfy the ($\Leftarrow$) or converse of Dirac or Ore's theorem.

Planar Graphs:

Say we are designing a complex electronic circuit. We have to place the electronic circuit on a board called Printed circuit Board (PCB). This board is basically a 2D plane. The open wiring between electronic components should not overlap inorder to avoid short circuit.

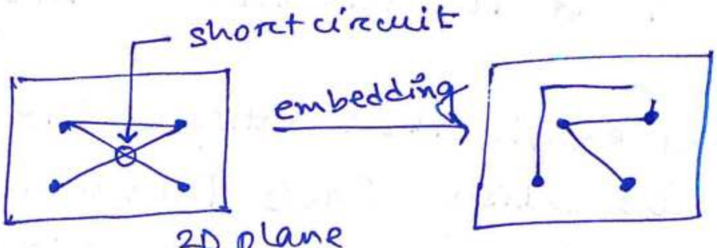


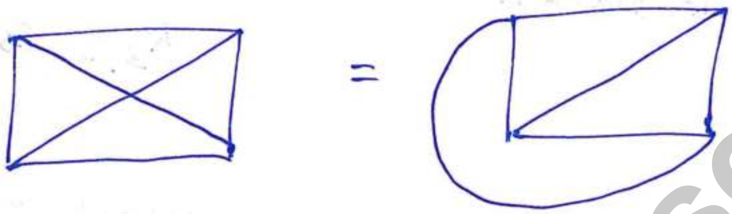
Fig: PCB

Fig: PCB

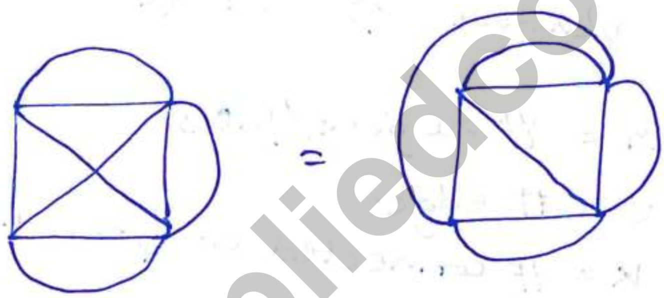
Usually complex electronic circuits have millions of electronic components.

A graph is a planar graph $\iff$ There exists an embedding in 2D plane such that no edges cross each other.

K_4 :



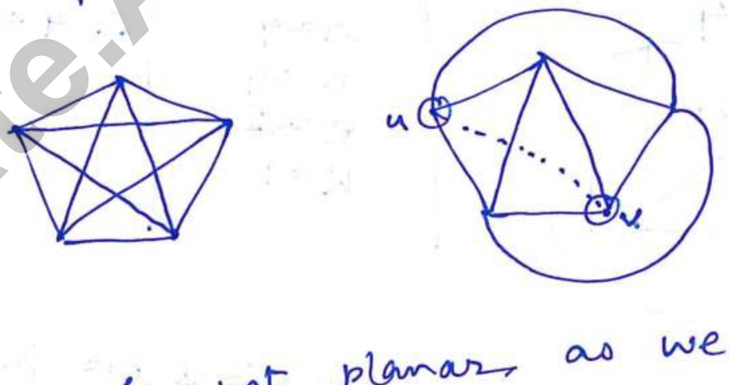
$\therefore$ K_4 is planar



G G

Graph G is planar

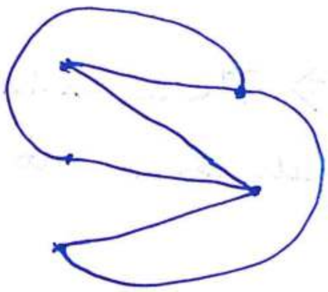
K_5 :



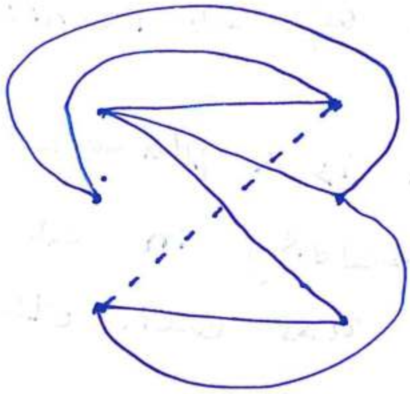
K_5 is not planar as we can't connect the edge (u,v) without overlapping other edges.

Note: K_n is planar iff $n \leq 4$.

$K_{m,n}$ is planar iff $m \le 2$ or $n \le 2$



$K_{3,2}$ is planar



$K_{3,3}$ is not planar

Euler's formula for planar graph :

If graph is planar $\Rightarrow$ $r = (e - n + k + 1)$

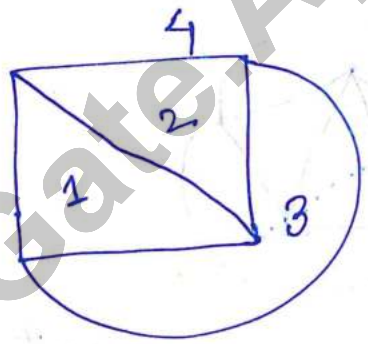
where r = # regions/faces

e = # edges

K = # connected component

n = # vertices

K_4 :



$r = 4$

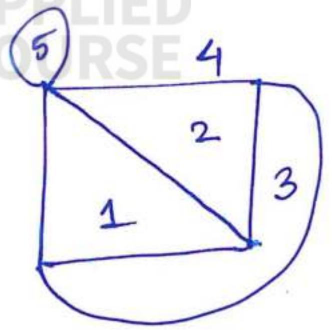
$n = 4$

$e = 6$

$K = 1$

$\therefore 4 = 6 - 4 + 1 + 1$ ✓

1, 2 and 3 are closed regions or faces while 4 is an open region/open face.



$\therefore 5 = 4 + 7 + 1 + 1$ ✓

Note:

If G is connected $\iff k = 1$

If G is connected and planar then,

$$\boxed{r = e - n + 2}$$

E.g. 3-regular graph, $O(G) = n = 10$, planar

faces = ?

$\therefore r = e - n + 1 + 1$

$\Rightarrow r = e - 8$

For 3-regular graph,

$\deg(v) = 3$ ✓

$e = \dfrac{3 \times 10}{2} = 15$

degree 3

n

because we have counted each edge 2 times.

$\therefore r = 15 - 8 = 7$

Eg. $n = 10$

size $= e = 15$

$K = 3$

$r = 15 - 10 + 3 + 1$

$r = 9$

closed faces = ?

For any graph,

closed faces = # faces − # open faces

$\because$ We know,

open faces $= 1$

$\therefore$ # closed faces $= 9 - 1 = 8$

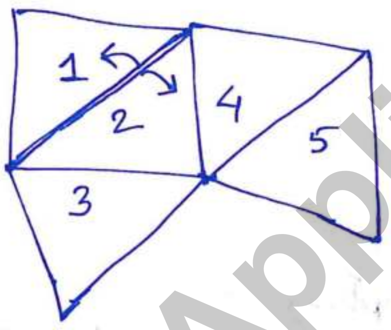
E.g.,

Given a planar graph, every region is bounded by 3 edges. $O(G) = 10$; connected.

regions = ?

$r = e - n + 2$

$\Rightarrow r = e - 10 + 2$



Each region is composed of 3 edges

$\therefore$ $\#\text{regions} = \frac{\#\text{edges}}{3}$

Now, every edge e is counted 2 times.

$\therefore \#\text{edges} = 2e$

$\therefore \boxed{r = \frac{2e}{3}}$

$\boxed{\frac{3r}{2} = e}$

$\Rightarrow \boxed{2e = 3r}$

$\therefore r = e - 10 + 2$

$\Rightarrow r = \frac{3r}{2} - 10 + 2$

$\Rightarrow r = 16$

$\therefore e = 24$

For any region, we need atleast 3 edges to create the region.

$\therefore$ $\boxed{2e \geq 3r}$ in a planar graph.

$K_{m,n}$: For planar, complete, bipartite graph.

$\therefore$ $m \leq 2$ or $n \leq 2$ to be planar.

$$r = \underbrace{mn}_{\#edge} - \underbrace{(m+n)}_{\#vertices} + 2$$

Kuratowski's Theorem : (Test for planarity)

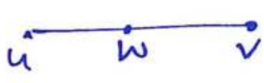
G is planar $\Longleftrightarrow$ G doesnot contain a subgraph that is a subdivision of K_5 or $K_{3,3}$

Note! We have seen that K_5 is the smallest complete graph which is non-planar. And $K_{3,3}$ is smallest complete bipartite graph which is non-planar.

Note! Subdivision / expansion :

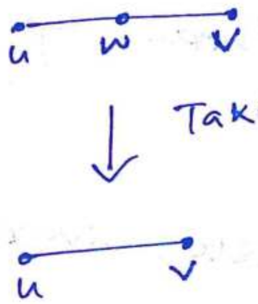
Given an edge (u,v) in following graph:

Subdivision of an edge is dividing an edge into 2 edges.



Note:

Smoothing:



Taking 2 edges and combining them into one edge.

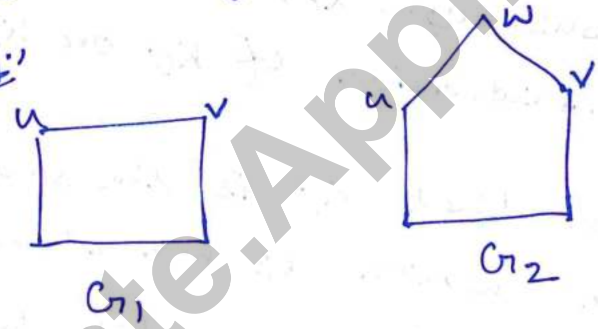
Note:

Homeomorphism:

$$\boxed{G_1 =_h G_2}$$

G_1 is homeomorphic to G_2 if we can arrive at G_2 by subdividing some edges in G_1, then $G_1 =_h G_2$

E.g.:



Here, $G_1 =_h G_2$

Theorem:
Given connected simple planar graph,

$\boxed{e \leq 3n-6}$.

Proof: $r = e - n + 2$

For all planar graphs, we know,

$2e \geq 3r$

$\therefore\ r \leq \frac{2e}{3}$

$\Rightarrow e - n + 2 \leq \frac{2}{3}e$

$\Rightarrow \frac{1}{3}e \leq n - 2$

$\Rightarrow \boxed{e \leq 3n-6}$

Theorem:
For connected simple planar graph with no triangles (no 3 edge based regions), then $e \leq 2n-4$.

Proof: Minimum number of edges required to form a region (with no 3 edge regions) is 4.

$\therefore$ # regions $= \frac{\text{\# edges}}{4}$

Every edge e is counted 2 times,

$\therefore$ # edges $= 2e$

$\therefore\ r = \frac{2e}{4} \Rightarrow \boxed{2e = 4r}$

For any region, we will need atleast 4 edges to create the region.

$\therefore \boxed{2e \geq 4r}$

$\therefore r = e - n + 2$

$\Rightarrow e - n + 2 \leq \frac{2e}{4}$

$\Rightarrow \frac{1}{2} e \leq n - 2$

$\Rightarrow \boxed{e \leq 2n - 4}$

<u>Property</u>: If we have a connected simple planar graph, then minimum degree (δ) is less than equal to 5,

i.e., $\delta \leq 5$

<u>Proof</u>: We know $\delta \leq \frac{2e}{n}$

$\delta \leq \frac{2(3n-6)}{n}$

$\boxed{\delta \leq 6 - \frac{6}{n}}$

If n is finite (considering only finite graphs here),

$\left(6 - \frac{6}{n}\right)$ will always be less than 6.

$\therefore \delta \leq 5$, as δ is an integer

Property: For complete simple planar graph with no triangles, $\delta \leq 3$

Proof:

$$\delta \leq \frac{2e}{n}$$

$$\delta \leq \frac{2(2n-4)}{n}$$

$$\delta \leq 4 - \frac{8}{n}$$

$\therefore \boxed{\delta \leq 3}$ for finite values of n.

(Q) Which cannot be # edges in a complete simple planar graph with $n = 10$.

a) 22
b) 23
c) 24
d) 26

Soln. Triangles can exist, therefore,

$$e \leq 3n - 6$$

$$e \leq (3 \times 10 - 6)$$

$$\boxed{e \leq 24}$$

$\therefore$ Option (d) is not possible.

⊛ Trees :

Tree ⟺ a connected acyclic graph

Tree is also called a minimally connected graph, because we can't remove even a single edge without making the graph disconnected.

Tree ⟺ connected graph with exactly $(n-1)$ edges.

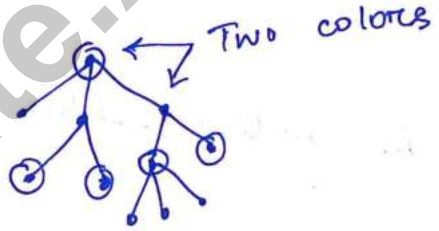
⟺ Acyclic graph with $(n-1)$ edges.

⟺ $\forall u,v\ \exists$ exactly one path $u \sim v$
(acyclic) (connected)

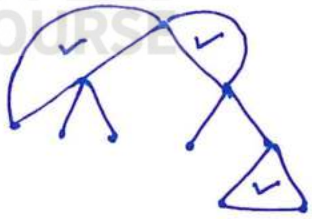
⟺ one, connected graph.

Properties

① Every tree is bi-chromatic and it is bipartite.



② Fundamental cycle :
Fundamental cycle is the cycle that we obtain by adding exactly one edge



Fundamental cycles in the tree.

fundamental cycles = $^nC_2$

Rooted Tree : Tree with a specific vertex choosen as a root. In a rooted tree, we have parent-child and siblings relationships. Leaf nodes and internal nodes are defined only for rooted trees. Leaf nodes are the nodes that has no child. Anything that is not leaf node is an internal node, including the root node itself.

depth of vertex u = depth (u) = distance(u, root).
Since there are no cycles, distance is the shortest distance between vertex u and root.

$$\text{height}(\text{Tree}) = \text{height}(T) = \max_u \text{depth}(u)$$

$$\text{level}(u) = 1 + \text{depth}(u)$$

$$\text{level}(\text{Tree}) = \text{level}(T) = 1 + \text{height}(T)$$

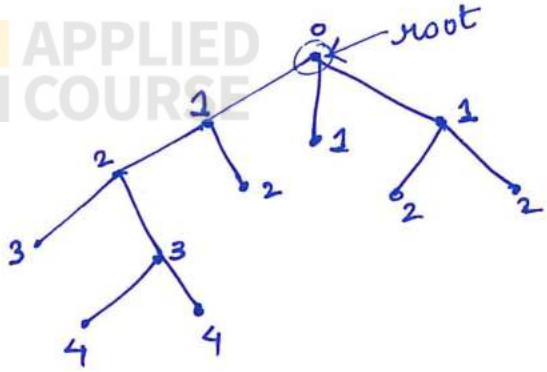


Fig: Tree T.

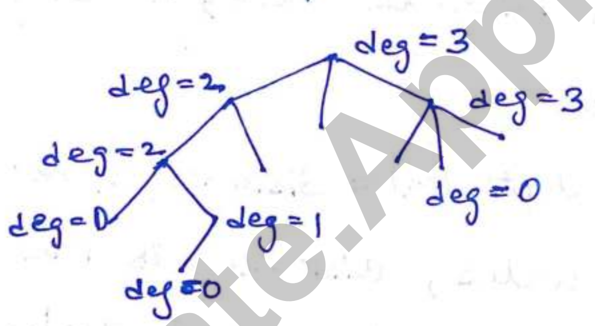
height(T) = 4

depth(root) = 0

K-ary Tree :

It is a rooted tree and the # children(u) is greater than equal to zero and less than equal to K.

$$0 \leq \# \text{children}(u) \leq K$$



Here deg(u) is the number of children.

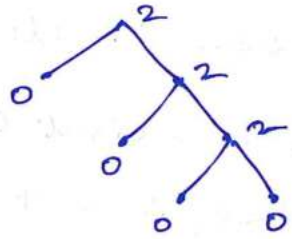
K = max degree = 3

∴ It is a 3-ary tree.

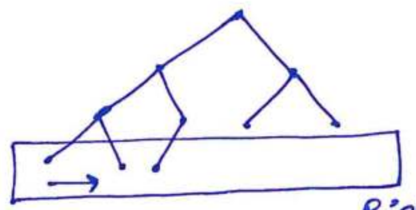
Binary tree = 2-ary tree.

<u>Full K-ary Tree</u>: It is a rooted tree and

#children (u) = 0 or k



<u>Complete k-ary Tree</u>: It is a rooted tree and all levels except the last are completely filled. And the last level is left adjusted.



Complete k-ary tree $\not\Leftrightarrow$ full k-ary tree

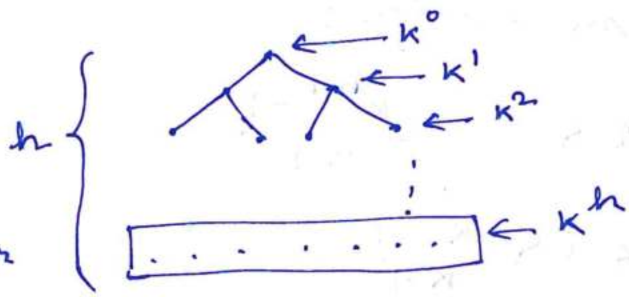
<u>Properties</u>:

<u>K-ary tree</u>

l = # leaves
n = # nodes
h = height
i = # internal nodes

<u>To get maximum</u>:-

① $l \leq k^h$



where,

k^h = maximum number of nodes at a height h.

② $n \leq \left(\frac{k^{h+1}-1}{k-1}\right)$

∴ Total number of nodes in the tree

= maximum # nodes in height 0 +

max # nodes in height 1 + ...

+ max # nodes in height h

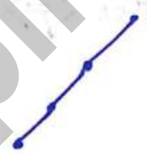
$= k^0 + k^1 + k^2 + \dots + k^h$

$= \frac{k^{h+1}-1}{k-1}$

③ $i \leq \frac{k^h-1}{k-1}$

To get minimum :

① $n \geq h+1$



$h=3$
$n=4$

Fig: chain/line
(k-ary tree)
(extremely skewed tree)

② $i \geq h$

To summarize :

$h+1 \leq n \leq \frac{k^{h+1}-1}{k-1}$

$h \leq i \leq \frac{k^h-1}{k-1}$

Ph: 844-844-0102

Note:
We know, $l \le k^h$

$$\boxed{\log_k l \le h}$$

$$h = \lceil \log_k \rceil$$

Spanning Tree of G :

Spanning Tree of a graph is a subgraph T(G) of G —

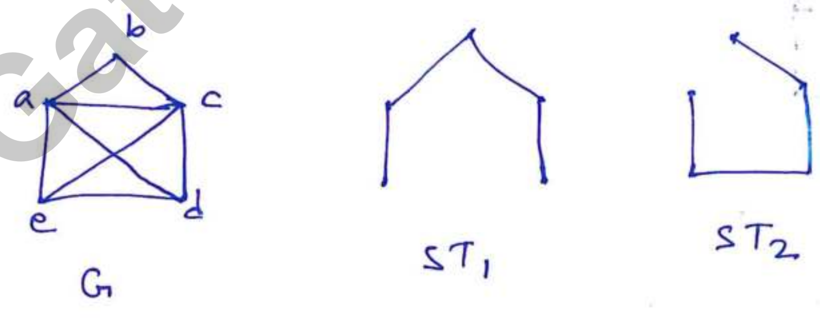
(i) Tree
(ii) includes / spans all vertices of G

Note: Minimum spanning Tree is covered in DS/Algo.

Property:

Every connected graph G has a Spanning Tree (ST)

Note: Spanning Tree need not be a rooted tree.



Mail: gatecse@appliedcourse.com

Strategies to count # Spanning Trees:

①

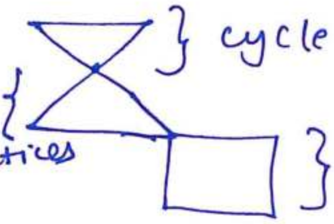
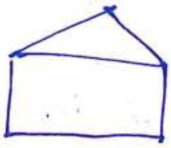


Fig: cycle disjoint graph

Cycle disjoint graphs have common vertices and no common edges.

$\therefore$ # spanning Tree $= {}^3C_2 \times {}^3C_2 \times {}^4C_3 = 36$

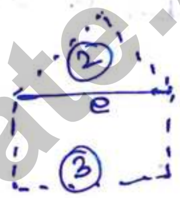
②



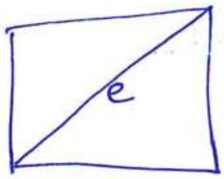
We count all ST without e $\longrightarrow {}^5C_1 = 5$

we count all ST with e $\longrightarrow {}^3C_1 \times {}^2C_1 = 6$

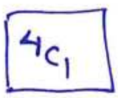
$= 11$

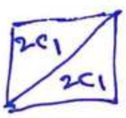


③



without e $—$ ${}^4C_1$

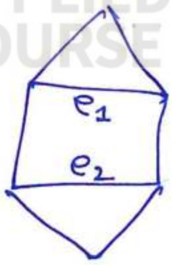




with e $—$ ${}^2C_1 \times {}^2C_1 = 4$

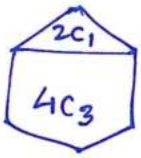
8

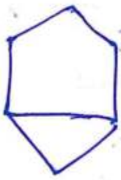
(4)



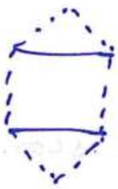
e_1 & e_2 both are not used $\} \rightarrow {}^6C_1 = 6$

e_1 is used but not e_2 $\} \rightarrow {}^2C_1 \times {}^4C_3 = 8$





e_2 is used but e_1 is not used $\} \rightarrow {}^2C_1 \times {}^4C_3 = 8$



e_1 and e_2 both are used $\} \rightarrow {}^2C_1 \times {}^2C_1 \times {}^2C_1 = 8$

30

Note:

$\text{Rank}(G) = n - k$

$\text{Nullity}(G) = e - \text{rank}(G) = e - n + k$

$\text{Rank} + \text{Nullity} = e$

$K = 1$
$n = 5$
$e = 6$

rank = 4
nullity = 2

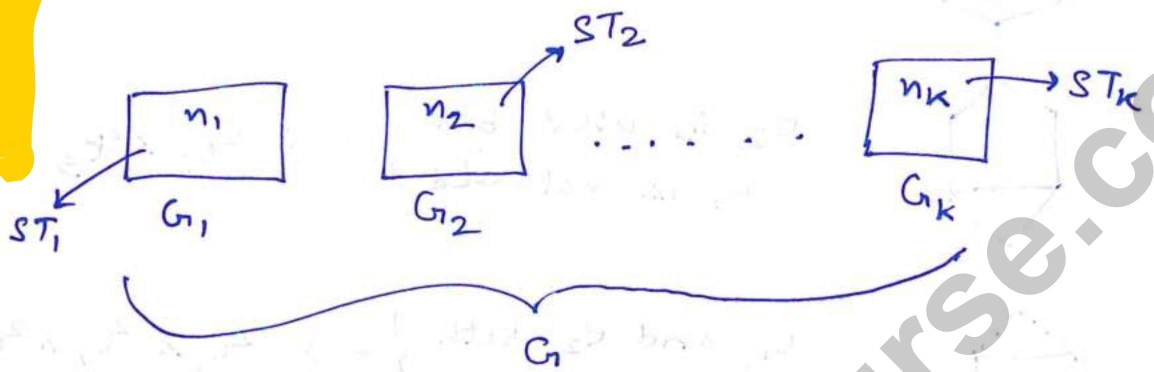
Case I: connected graph.

In the context of connected graph:

$rank(G) = n-1$

$= \#$ edges in the S.T of G

Case II: Disconnected graph G:



The set of spanning tree in a disconnected graph is called a spanning forest (SF).

The size of $ST_1 = n_1 - 1$

The size of $ST_2 = n_2 - 1$

$\vdots$

The size of $ST_K = n_K - 1$

$(n_1 - 1) + (n_2 - 1) + \cdots + (n_K - 1) = \#$ edges in the Spanning Forest (SF)

$\Rightarrow \underbrace{(n_1 + n_2 + \cdots n_K)}_{} - K.1$

$rank(G) = (n - K) = \#$ edges in SF

Summary:

$rank(G) = \#$ edges in S.T or (S.F of G) based on whether the tree is connected or disconnected.

Nullity of graph G = Nullity(G) = Minimum # edges that need to be removed from the graph to make it a spanning tree or spanning forest.

Nullity(G) is also called cyclomatic complexity which is the number of edges to be removed to break all cycles.

Branch Set is the set of all edges in S.T/S.F

|Branch set| = Rank(G)

$$\boxed{|BS| = \text{Rank}(G)}$$

Chord set = set of edges that need to be removed from a graph to make it a spanning tree or spanning forest.

|CS| = Nullity(G)

(*) Graph Counting:

Enumeration of Graphs: Here we count the number of possible graphs (simple and undirected)

① # labelled graph with n vertices = $2^{{}^nC_2}$

② # simple labelled graphs given n, e = ${}^{\frac{n(n-1)}{2}}C_e$

③ # labelled Trees given $n = n^{(n-2)}$
$\uparrow$ Cayley's formula

ST in K_n $= n^{(n-2)}$

④ # rooted labelled trees given n vertices

$= \underbrace{n}_{\text{n ways to select root}} . n^{n-2} = n^{n-1}$

⑤ # labelled subgraphs of K_n (labelled)

$$= \underbrace{\sum_{i=1}^{n} \underbrace{{}^nC_i}_{i \text{ vertices}} \; \underbrace{2^{i(i-1)/2}}_{\#\text{graphs on } i \text{ vertices}}}_{\text{each of such graphs are subgraphs of } K_n.}$$

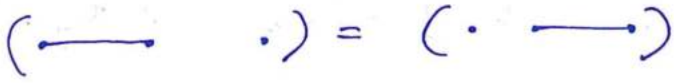
Eg. # graphs with n vertices and atleast $\frac{n(n-1)}{4}$ edges

$$= \underbrace{{}^{{}^nC_2}C_{\frac{n(n-1)}{4}}}_{\text{atleast}} + {}^{\frac{n(n-1)}{2}}C_{\left(\frac{n(n-1)}{4}+1\right)} + \dots + \underbrace{{}^{n(n-1)/2}C_{\frac{n(n-1)}{2}}}_{\text{atmost}}$$

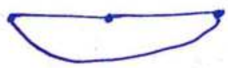
as number of edges with n vertices $= {}^nC_2 = \frac{n(n-1)}{2}$

unlabelled graphs with 3 vertices = 4

0 edges 

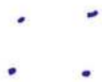
1 edges 

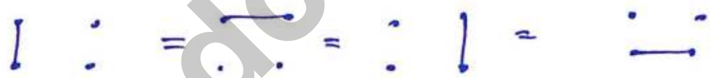
2 edges 

3 edges 

4

unlabelled graphs with n=4

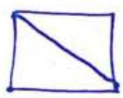
0 edges  —— 1

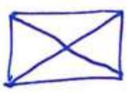
1 edges  —— 1

2 edges  —— 2

3 edges  —— 3

4 edges  —— 2

5 edges  —— 1

6 edges  —— 1

Total distinct possibilities = 11

(Q) # subgraphs with $e=3$, $n=5$ (labelled)

3 vertices & 3 edges $\longrightarrow$ ${}^5C_3 \cdot 1$

4 " & 3 edges $\longrightarrow$ ${}^5C_4 \cdot {}^{{}^4C_2}C_3$

5 " & 3 edges $\longrightarrow$ ${}^5C_5 \cdot {}^{{}^5C_1}C_3$

Answer : ${}^5C_3 + {}^5C_4 \cdot {}^{{}^4C_2}C_3 + {}^5C_5 \cdot {}^{{}^5C_1}C_3$.

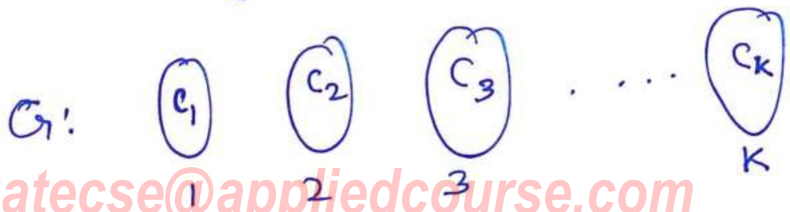
⊛ Graph coloring, Independence Sets & Dominating Sets.

Proper vertex coloring: Assigning a color to each vertex such that no two adjacent vertices have same color.

Min # colors required for vertex-coloring = chromatic number $\chi(G)$

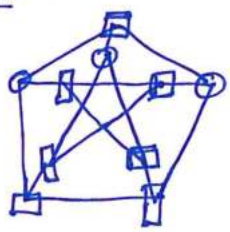
2-coloring $\Leftrightarrow$ bi-partite

K-coloring $\Leftrightarrow$ K-partite



Here, C_i is the i^{th} color.

Peterson Graph:



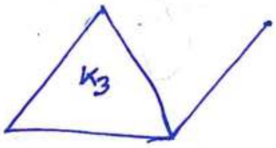
□, o, ▯ denote different colors.

$\chi(G) = 3$

① $1 \le \chi(G) \le n$

② Clique: complete subgraph of G

G:



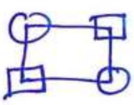
The clique number $\omega(G)$ is the largest size clique in a graph.

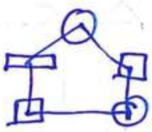
$\chi(K_n) = n$

G_1 is K_4

$\boxed{\chi \ge \omega}$ ie., we will need atleast ω colors to color the complete subgraph of G and remaining vertices may or maynot need new colors.

③ $\chi(C_n) = \begin{cases} 2 & \text{if even cycle} \\ 3 & \text{if odd cycle} \end{cases}$

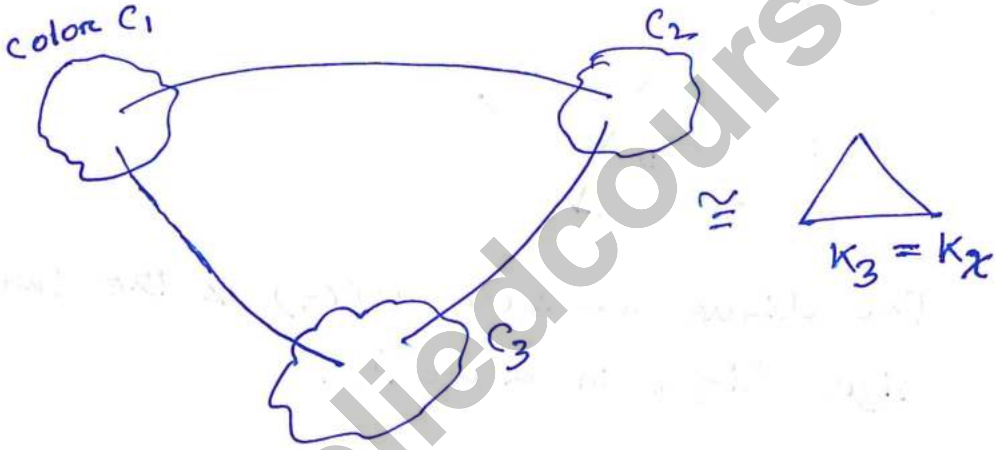




(4) $\dfrac{\chi(\chi-1)}{2} \le e$

$$\boxed{{}^{\chi}C_2 \le e}$$

Assume a graph is divided in such a way that the same color vertices are drawn together. Say, $\chi(G)=3$,



Now, given a complete graph with chromatic number(χ), # edges $\ge {}^{\chi}C_2$

(5) $\chi \le$ maximum degree or Δ, except K_n and odd C_n.

In case of K_n, $\chi(K_n) = n$
$\Delta(K_n) = (n-1)$

In case of odd C_n, $\chi(C_n) = 3$
$\Delta(C_n) = 2$

Therefore, $\boxed{\chi \le (\Delta+1) \le n}$

⑥ $\chi(\text{planar graph}) \le 4$ → 4 color Theorem

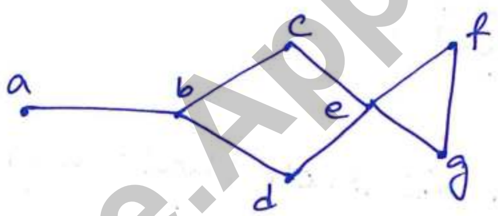
Four color theorem states that the vertices of a planar graph can be colored with atmost 4 colors so that no two vertices receive the same color.

⑦ $\chi(G) = 1 \iff \phi_n$

⑧ $\chi(\text{multigraph}) = \chi(\text{simple graph by dropping multiple edges})$

<u>Independence set</u> / <u>Anti-clique</u> <u>(IS)</u>:

set of vertices that are not adjacent to each other in a graph.



$\{a\}$ ✓
$\{b\}$ ✓
$\{a, c, d, f\}$ ✓

Independent sets

$\{a, c, d, e\}$ ✗

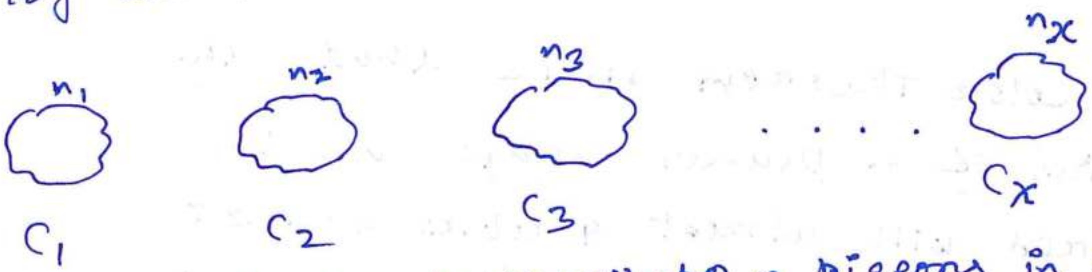
$\left.\begin{matrix} c - e \\ d - e \end{matrix}\right\}$ exists

Independence number (β_G) is the size of the largest sized IS.

$$\beta_G \geq \frac{n}{\chi(G)}$$

[can be proved using pigeon hole principle]

Let vertex represent pigeon and hole be represented by color.



We are trying to distribute n pigeons in χ holes.

$\exists$ one hole that has $\geq \frac{n}{\chi}$ vertices

Let the number of vertices $\geq \frac{n}{\chi}$

All these vertices are colored by same color. Therefore, they don't have edge between them.

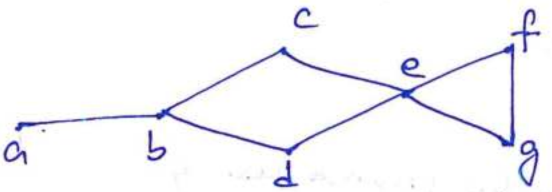
$\therefore$ They form independent set.

There exists an independent set which has $\geq \frac{n}{\chi}$ elements,

$$\therefore \boxed{\beta_G \geq \frac{n}{\chi(G)}}$$

<u>Maximal IS</u>: It is an IS such that any new vertex added will voilate the property of IS.

G:



$\{b, e\}$ → maximal IS

$\{a, c, d, f\}$ → maximal IS

$\{a, d, c, g\}$ → maximal IS

$$\boxed{\beta_G = \text{size of largest maximal IS}}$$

Q. How to obtain maximal IS?

① Start with min-degree vertex and ascending order of degree

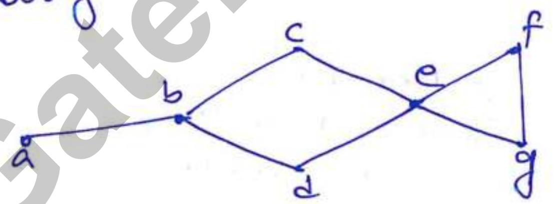
② Keep adding vertices by checking adjacency and ensuring that it doesnot violate the property of independent set.

In the graph G,

deg(a) = 1
deg(b) = 2
deg(c) = 2
deg(d) = 2
deg(f) = 2
deg(g) = 2
deg(e) = 4

∴ Maximal IS = {a, c, d, f}

Domination Set (DS): It is a set of vertices from which all vertices are just one step away.



{e, b} → minimal set

{a, c, d, f} → minimal set

Domination number (α):

= It is the size of the smallest DS.

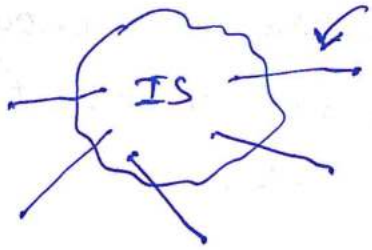
= It is the size of the smallest minimal DS.

= It is the size

If a set of vertices is Maximal IS

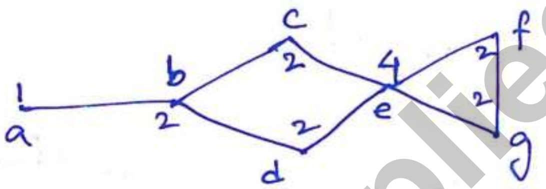
$\Rightarrow$ Dominating set

1 hop away



To determine the smallest DS ?

① start with maximal degree vertex
② Keep adding vertices in order as long as the set is a dominating set and you are reaching new vertices which are 1 hop away.



$\{e, b\}$

Theorem: $\alpha_G \leq \beta_G$

Domination number

Independence number

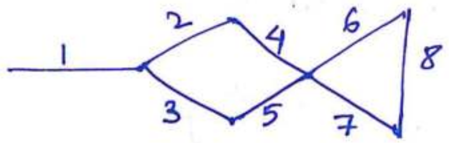
maximal IS $\Rightarrow$ D.S
$\not\Leftarrow$

$\therefore$ Independence # $\geq$ Domination number

(*) Matchings & Coverings:

Note: IS, DS and coloring are related to vertices.

Matchings: set of edges none of which are adjacent



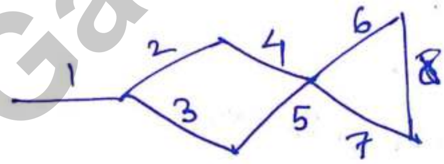
{1} ✓ — not maximal matching
{1,4} ✓ — " " "
{1,2} ✗ since they have common vertex.
{1,4,8} ✓ — maximal matching

Matching number (M_G) : size of largest maximal matching.

Q. To obtain maximal matching:

① start with an edge with least # of adjacent edges

② Keep adding edges without violating the matching property.

G:


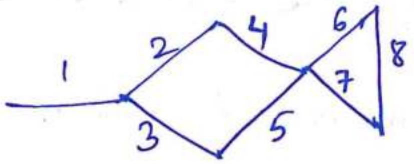
Edge 1 is adjacent to 2 vertices

Edge 2 ⟶ 3
Edge 3 ⟶ 3
Edge 4 ⟶ 4
Edge 5 ⟶ 4
Edge 6 ⟶ 4
Edge 7 ⟶ 4
Edge 8 ⟶ 2

{1,8,4}
{1,8,5} } maximal matching

$\therefore M_G = 3$

Covering: set of edges that covers all vertices



$\{1, 4, 6\}$ — Not a covering

$\{1, 2, 3, 4, 5, 6, 7, 8\}$ — covering

$\{1, 4, 8\}$ — Not a covering

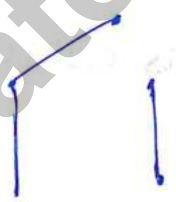
$\{1, 2, 3, 7, 6\}$ — covering (minimal covering)

Covering number (C_G): size of smallest minimal covering.

$\{1, 4, 5, 8\}$ — smallest minimal covering

$\therefore C_G = 4$

Theorem: $C_G \geq \lceil n/2 \rceil$



when n is odd

$\therefore \lceil n/2 \rceil$

when n is even

$\lceil n/2 \rceil$ holds

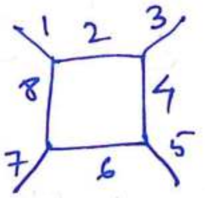
Theorem: Pendant edge is always part of the covering.

A Spanning Tree (ST) will always contain the pendant edge, otherwise spanning Tree (ST) is not spanning over all the vertices.

Build a covering set:

① Include all pendant edges.

② Include edges that cover the remaining vertices.



$\{1, 3, 5, 7\}$

$C_G = 4$.

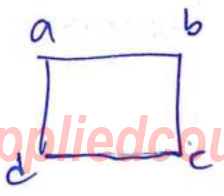
Theorem: (a) Covering need not be a matching

Covering → adjacent edges are allowed.

matching → adjacent edges are not allowed.

(b) Similarly, a matching need not be a covering. Because a matching need not cover all vertices.

Perfect matching: set of edges which are both matching and covering.



$\{ab, cd\}$ are not adjacent edges

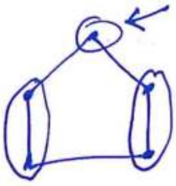
$\{ab, cd\}$ covers all vertices

$\therefore \{ab, cd\}$ is perfect matching.

Property: Perfect matching is possible when # vertices is even.

Note: Every graph with even # vertices need not be perfect matching.

Eg.

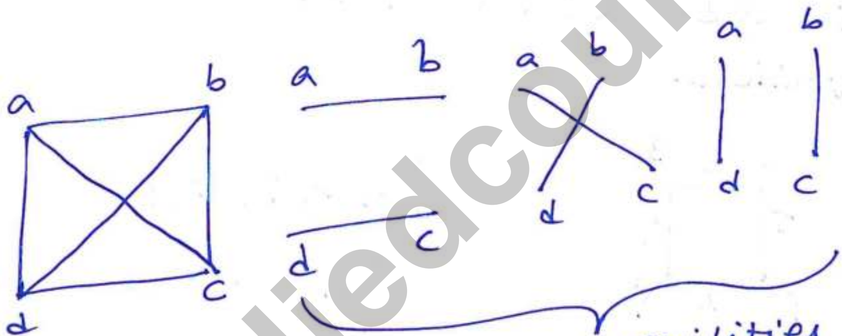


← not covered

when n = odd

Q. Given K_{2n}, how many perfect matchings possible?

K_4 :



3 possibilities

Given $2n$ vertices,

$\underbrace{1, 2}, \underbrace{3, 4}, \ldots \underbrace{2n}$

① order : $(2n)!$

② construct n pairs.

$n=2$,

$\underbrace{1, 2}, \underbrace{3, 4}$] — same
$\underbrace{3, 4}, \underbrace{1, 2}$

$n=4$,

$\underbrace{1, 2}, \underbrace{3, 4}, \underbrace{5, 6}, \underbrace{7, 8}$] — same
$\underbrace{5, 6}, \underbrace{3, 4}, \underbrace{1, 2}, \underbrace{7, 8}$

⋮

n pairs can be ordered in n! ways

$\therefore \frac{(2n)!}{n!}$

Now, every pair can be written in 2 ways.

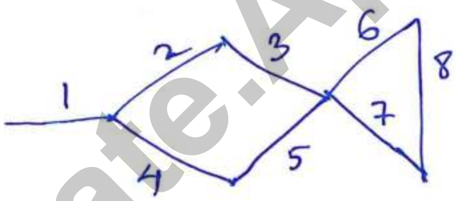
Eg: (1, 2) is same as (2, 1).

$\therefore \frac{(2n)!}{n!} \times \left(\frac{1}{2^n}\right)$

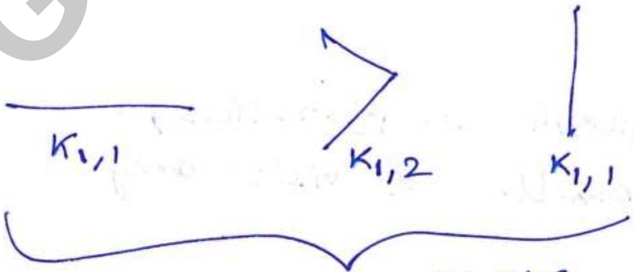
There are n pairs and each pair can be ordered in 2 ways.

$= \frac{(2n)!}{n!\, 2^n}$

Theorem: A covering is minimal iff every component is a star graph $(K_{1,n})$

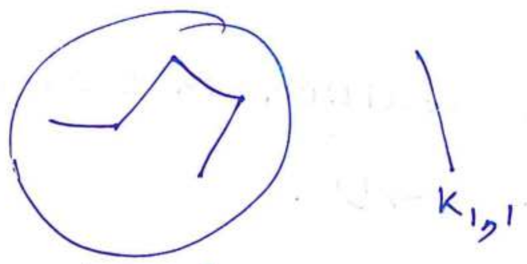


{1,3,5,8} minimal covering



Each components are star graphs

$\{1, 2, 3, 8, 5\} \rightarrow$ not minimal covering



This component is not a star graph

$\therefore \{1, 2, 3, 8, 5\}$ is not minimal.

Graph Theory Previous Year GATE Qn

Q. Let G be an undirected complete graph on n vertices, where $n > 2$. Then, the number of different Hamiltonian cycles in G is equal to: (GATE 2019)

(A) $n!$

(B) $n-1!$

(C) 1

(D) $(n-1)!/2$

In an undirected graph on n vertices, n permutation are possible to visit every node.

a) n different places (nodes) you can start

b) 2 (clockwise or anti-clockwise) different directions you travel.