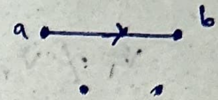
6. Go to step 2.

- **<u>Directed Graphs or Digraphs</u>** -

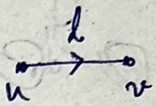


A digraph G is an ordered pair $G=(V, E)$ where V is a non-empty set & E is a multiset of ordered pairs of ~~vertices~~ elements in V (need not be distinct).

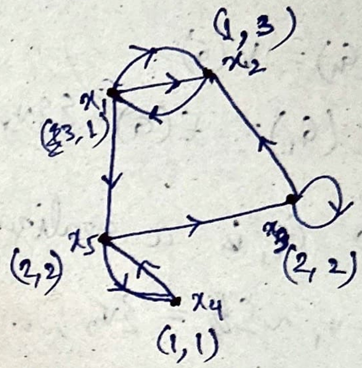
(a,b) is an edge but (b,a) is not. $(a,b) \neq (b,a)$

Elements of V → vertices

elements of E → edges or arcs



If $u, v \in E$ then u is called tail & v is called head of this edge l.



Outdegree of $x_1 = d^+(x_1)$
$= 3$

Indegree of $x_1 = d^-(x_1) = 1$

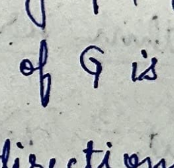
- <u>Theorem</u>: In every digraph D, we have

$$\sum_{x \in V(D)} d^+(x) = \sum_{x \in V(D)} d^-(x) = |E(D)|$$

- The graph obtained from a digraph D by removing all the directions is called the underlying graph of D.

Let G be a graph. An ~~orient~~ orientation of G is a digraph obtained from G by giving directions to all the edges of G.

Orientation of a complete graph is called a tournament.



Digraphs $D_1=(V_1, E_1)$ & $D_2=(V_2, E_2)$ are isomorphic iff $\exists$ a bijection

$f: V_1 \rightarrow V_2$ s.t $(u,v) \in E_1$ iff $(f(u), f(v)) \in E_2$

Q7 Find upto isomorphism of all tournaments on n 4 vertices.
(Round-Robin tournament)

2_6

Def^n: A vertex with maximum outdegree in a tournament is called a king.

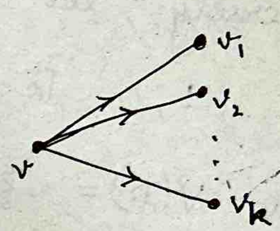
$D \to$ tournament, v is a king in D.

$u \in V(D)$, $u \neq v$. $\quad v \to u$ or $v \to w \to u$

• **Theorem:** Let D be a tournament and $v \in V(D)$ be a king. For any vertex $u \in V(D)$, there is a $v \to u$ directed path of length atmost 2.

Proof: If $(v, u) \in E(D)$, then we get a v-u directed path of length 1. Next, let $(v, u) \notin E(D)$. Since D is a tournament we get $(u, v) \in E(D)$.

Let $d^+(v) = k$, which is the maximum outdegree in D.



$|V(D)| = n$

If $(v_i, u) \in E(D)$ for some i, $1 \le i \le k$ then a directed path $v \to v_i \to u$ is of length 2.

Then it is done.

If $(u, v_i) \in E(D)$, $i = 1, 2, \dots, k$, then $d^+(u) \geq k+1$, which is a contradiction.

Hence $\exists$ i, $1 \le i \le k$ such that $(v_i, u) \in E(D)$.

$v \to v_i \to u$

Def^n: The outdegree sequence of a tournament is called its score sequence.

Def^n: A digraph D is said to be transitive if (u, v), $(v, w) \in E(D) \Rightarrow (u, w) \in E(D)$

• **Theorem:** A tournament D is transitive iff the score sequence of D is $n-1, n-2, \ldots, 1, 0$,

where $n = |V(D)|$.

Proof: Suppose D is transitive. Possible outdegrees are

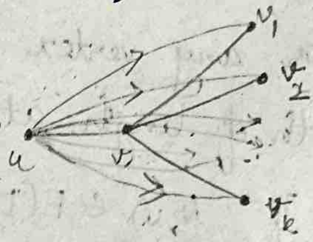
$0, 1, 2, \ldots, n-2, n-1$.

Claim - No two vertices in D have the same outdegree.

Let $u, v \in V(D)$, $\cancel{u \neq v}$ $u \neq v$.

We show that $d^+(u) \neq d^+(v)$.

W.L.O.G., let $(u, v) \in E(D)$. Let $d^+(v) = k$.



$\Rightarrow d^+(u) \geq k+1$.

$\Rightarrow d^+(u) \neq d^+(v)$

Hence the score seqn. of D is

$n-1, n-2, \ldots, 1, 0$.

Conversely, let the score sequence of D be $n-1, n-2, \ldots, 1, 0$. To show that D is transitive.

Let $V(D) = \{v_0, v_1, \ldots, v_{n-1}\}$ with $d^+(v_i) = i$, $i = 0, 1, \ldots, n-1$

$d^+(v_{n-1}) = n-1$

$d^+(v_{n-2}) = n-2$ ... and so on

$\Rightarrow (v_i, v_j) \in E(D)$ iff $i > j$, which is a transitive relation.

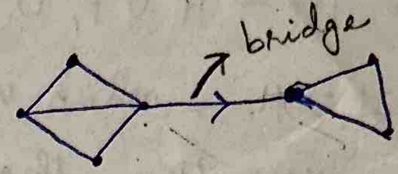
Hence, D is transitive.

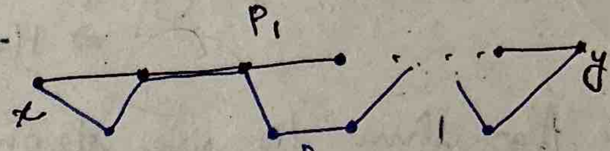
Defn: A digraph D is said to be weakly connected if the underlying graph of D is ~~conn~~ connected.

D is called strongly connected if for every pair $x, y \in V(D)$, $x \neq y$, $\exists$ a x-y directed path in D.

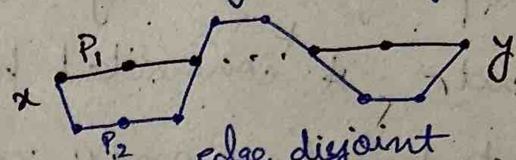
Defn: A graph G is said to be orientable if G has a strongly ~~con~~ connected orientation.

Lemma: A graph G has no bridges iff between every pair of vertices x & y there are ~~atl a~~ atleast two edge disjoint x-y paths.





edge not disjoint.



edge disjoint

• **Theorem:** A graph G is orientable iff G has no bridge.

Proof: If G is orientable, then G has no bridge.

Next let G is bridgeless.

$dg(x) \geq 2 \;\; \forall \; x \in V(G)$.

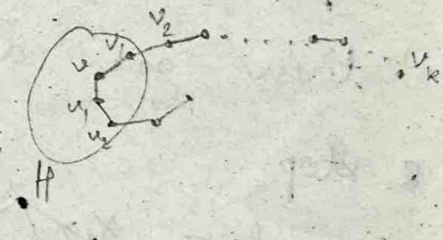
G contains a cycle C.

Let H be a maximal ~~indu~~ induced orientable subgraph of G. $H \neq \phi$. [$\because$ $C \subseteq H$]

Claim - $V(H) = V(G)$.

Suppose $V(H) \neq V(G)$. Then $\exists$ $u \in V(G)$ s.t $u \notin V(H)$.

Since G is connected & bridgeless there are atleast two edge disjoint v-u paths, P_1 & P_2, in G.



$$H_1 = H \cup P_1 \cup P_2$$

Let P_1: $v = v_0, v_1, \ldots, v_k = u$

P_2: $v = u_0, u_1, \ldots, u_r = u$.

Let v_i & u_j be last vertices of P_1 & P_2 respectively s.t $v_i, v_j \in V(H)$. (may be $v_i = v_j = v$)

Give orientation of $P_1 \cup P_2$ -

$(v_i, v_{i+1}), \ldots, (v_{k-1}, v_k = u), (v, u_{r-1}), (u_{r-1}, u_{r-2}), \ldots, (u_{j+1}, u_j)$.

We ~~have~~ ~~a~~ have a directed $u_j - v_j$ path in H.

$\Rightarrow H_1 = H \cup P_1 \cup P_2$ is orientable.

$H_1 \supset H$, which is a ~~contradiction~~ contradiction to the assumption that H is ~~maximal out~~ maximal orientable subgraph. Hence $V(H) = V(G)$

$\Rightarrow H = G$

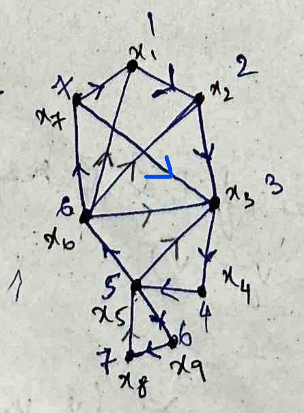
- Algorithm to give strongly connected orientation – (Hopcroft & Tarjan, 1973) –

Input – A connected bridgeless graph G.

Output – A strongly connected orientation of G.

1. Let $x \in V(G)$. Then $l(x) = 1$.
2. $L = \{x\}$, $U = V(G) - L$, $A = \phi$.
3. Let v be the vertex in L with (i) largest possible label & (ii) v be having a neighbour in U, say u.
4. $l(u) = l(v) + 1$ [$l(x)$ = label of x]
5. $L = L \cup \{u\}$, $U = U - \{u\}$, $A = A \cup \{(v, u)\}$
6. If $L \neq V(G)$ then go to step 3.
7. If ~~$\{(a,b)\}$~~ $\{a, b\}$ is an edge of G with no orientation then give a direction from a to b if $l(a) > l(b)$.
8. Stop

example –



$L = \{x_1, x_2\}$, $A = \{(1,2), (2,3)\}$ are set

$L = \{x_1, x_2, x_3\}$

- Trees – A tree is a connected acyclic ~~gra~~ graph.

~~Acyclic graphs graphs are called for forests.~~

A forest is a collection of trees. A degree one vertex in a tree (or ~~for~~ forest) is called a ~~leaf~~ leaf.

~~A deg degree~~

• **Theorem**: The following statements are equivalent for an n-vertex simple ~~gra~~ ~~graf~~ graph G,

(i) G is a tree (G is connected acyclic)

(ii) There is a unique path between every pair of vertices in G.

(iii) G is connected ~~with~~ ~~with~~ with $n-1$ edges.

(iv) G is acyclic with $n-1$ edges.

Proof: (i) $\Rightarrow$ (ii) and (ii) $\Rightarrow$ (i) (do yourself)

(i) $\Rightarrow$ (iii).

G is connected acyclic.

To prove that G has $n-1$ edges.

Induction on n.

$n=1$, . Result is true.

$n=2$, •—• ~~True~~

Assume result is true upto $n-1$ vertices.

~~Con~~ Consider the graph G with n vertices, $n \geq 2$.

Let l be an edge in G. $G-l$ is disconnected with components, say G_1 and G_2.

Let $|V(G_1)| = n_1$ & $|V(G_2)| = n_2$. From the assumption, $|E(G_1)| = n_1 - 1$. $|E(G_2)| = n_2 - 1$.

Now total no. of edges in $G = (n_1 - 1) + (n_2 - 1) + 1$.

$$= (n_1 + n_2) - 1$$

$$= n - 1 \quad [\because n_1 + n_2 = n]$$

Hence (i) $\Rightarrow$ (iii)

(iii) $\Rightarrow$ (i). To prove that G is acyclic.

Given $|V(G)| = n$ and $|E(G)| = n-1$. G is connected.

Induction ~~on~~ on n.

$n=1$, 

$n=2$,  Result is true.

Assume that result is true upto $n-1$ vertices.

Consider the graph G with n vertices.

Claim- G contains atleast one degree-one (pendant) vertex.

$$\sum_{x\in V(G)} deg(x) = 2e = 2(n-1) = 2n-2. \quad \text{—} \; \mathrm{I}$$

$deg(x) \geq 1$

If $deg(x) \geq 2$ $\forall x \in V(G)$, then $\sum_{x\in V(G)} deg(x) \geq 2n$

which ~~cont~~ contradicts to I.

Hence claim is ~~tru~~ true, i.e., G contains a pendant vertex.

<u>Remark</u>: Every tree has atleast two pendant vertices or leaves.

Let $v \in V(G)$ with $deg(v) = 1$. Then $G-v$ is connected with $n-1$ vertices. From the assumption, $G-v$ is acyclic.

Since v is a pendant vertex, G is also acyclic.

(i) $\Leftrightarrow$ (iv)

(i) $\Rightarrow$ (iv) follows from (i) & (iii) because we have (i) $\Leftrightarrow$ (iii).

(iv) $\Rightarrow$ (i). To prove that G is connected.

Suppose ~~that~~ there are k components in G. We prove that $k=1$. Let the k components be

$G_1, G_2, \ldots, G_k$ with no. of vertices $n_1, n_2, \ldots, n_k$ respectively. $\sum_{i=1}^{k} n_i = n$

Every component is connected ~~with n-1 vertices~~ and acyclic (given). Hence each G_i is a tree.

Then G_i has $n_i - 1$ edges (from (iii) as (i)⇒(iii))

$$n-1 = |E(G)| = \sum_{i=1}^{k} |E(G_i)|$$

$$= \sum_{i=1}^{k} (n_i - 1)$$

$$\Rightarrow k = 1.$$

Hence graph G is connected.

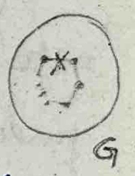
• **Spanning ~~tree~~ tree in a graph** – always connected

Let G be a graph. A subgraph H of G is called a spanning subgraph of G if $V(H) = V(G)$".

A subgraph H is called a spanning tree of G if

i) H is a tree.

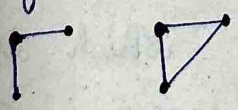
ii) $V(H) = V(G)$.



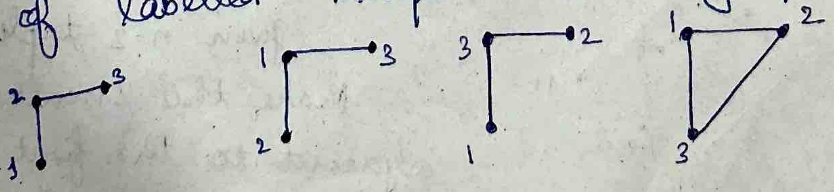
Every connected graph contains a spanning tree.

• **Labelled graph & unlabelled graph** –

No. of unlabelled simple connected graphs on 3 vertices (upto isomp)



No. of labelled simple connected graphs on 3 vertices.



(Cayley, 1889)

• **Theorem** : A labelled complete graph on n vertices has n^{n-2} spanning ~~subgraphs~~ trees.

OR The no. of labelled trees on n vertices is n^{n-2}.

Proof: (Prufer, 1918)

$X \to$ the collection of all labelled trees on n vertices.

To show - $|X| = n^{n-2}$

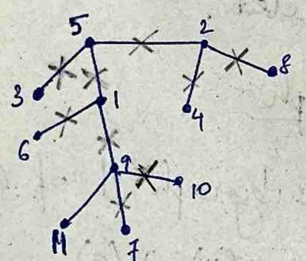
$Y = \{(a_1, a_2, \ldots, a_{n-2}) : a_i \in \{1, 2, \ldots, n\}\}$

$|Y| = n^{n-2}$

We get a 1-1 correspondence between X and Y below -

Consider an arbitrary element T in the set X, i.e., T is a labelled tree on n vertices.

(example) $n = 11$

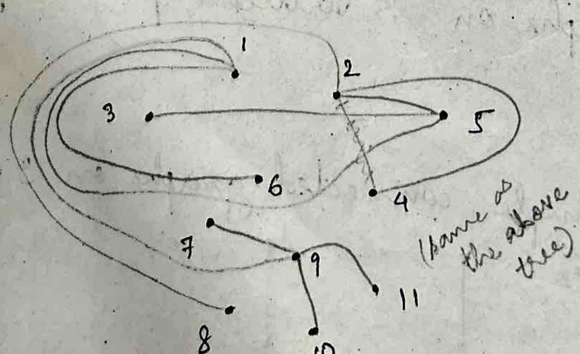


$\underbrace{(5, 2, 1, 9, 2, 5, 1, 9, 9)}_{\text{Prufer code of T}}$

Among all pendant vertices consider the smallest one & note down its neighbour. Then delete this smallest pendant vertex. Repeat this process till we left with a single edge.

Conversely consider the n-2 tuple $(5, 2, 1, 9, 2, 5, 1, 9, 9) = w$

Consider $S = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$



Find the smallest no. in S which does not appear in the given n-2 tuple w. Make this smallest no. adjacent to the first element in w. Then delete this

$w = (\cancel{5}, \cancel{2}, \cancel{1}, \cancel{9}, \cancel{2}, \cancel{5}, \cancel{1}, \cancel{9}, 9)$

$S = \{\cancel{1}, \cancel{2}, \cancel{3}, \cancel{4}, \cancel{5}, \cancel{6}, \cancel{7}, 8, 9, 10, 11\}$

smallest element in S & the 1st element in w. Repeat this process in the remaining of S and w. Finally make adjacent the remaining two symbols in S.

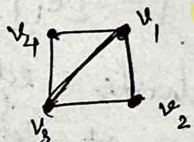
Hence $|X| = |Y| = n^{n-2}$.

Then we count the no. of spanning trees in an arbitrary labelled graph.

• **Theorem** (Matrix-tree theorem): Let G be a loop free connected labelled graph. Let A(G) be the adjacency matrix of G and let D(G) be the degree diagonal matrix of G whose i^{th} diagonal entry is the degree of the i^{th} vertex of G. Consider $L(G) = D(G) - A(G)$. Let L^* be the matrix ~~def~~ obtained from L(G) by deleting r^{th} row & s^{th} column. Then the total no. of spanning trees in G is equal to $(-1)^{r+s} \det(L^*)$.

(Note: L(G) is called the Laplacian matrix of G.)

Proof???

Q) Let G be the labelled graph 

. Find total no. of spanning trees (labelled) in G.

Solⁿ:

$$A(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

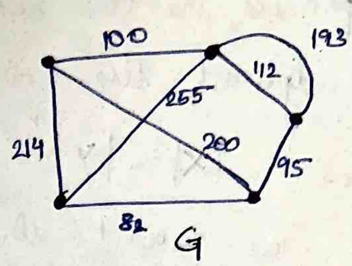
$$D(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \end{array}, \quad L(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \end{array}$$

Total no. of spanning trees in G is 8.

Spanning trees in a weighted graph -

$T \subseteq G$ is subgraph.

~~subgraph~~ $w(T) = \sum_{e \in E(T)} w(e)$



Among all spanning trees in G, the ones having minimum weight are ~~call~~ called minimum spanning tree.

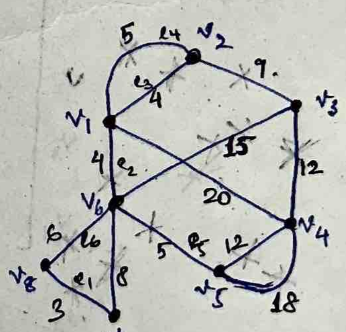
Algorithms to find minimum spanning trees in a weighted graph - &

Kruskal's Algorithm -

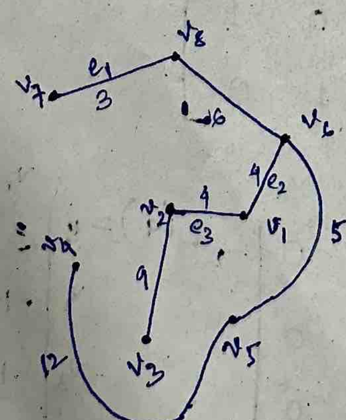
Input: A connected ~~wet~~ weighted graph G.

Output: A minimum spanning tree T of G.

1. List the edges of G in non-decreasing order of their weights.
2. $T = \{e_1\}$ = the 1st edge in the above list.
3. Then include edges in T from the list (given in 1) if I does not form a cycle with the edges already considered in T.
4. If $|E(T)| = n-1$, then output T otherwise go to step 3.



$T =$



$\{e_1, e_2, e_3\} \cup \{e_4\} \to$ forms a cycle.

$w(T) = 3+6+4+4+9+5+12 = 43$

• **Theorem**: The tree obtained in Kruskal's algorithm is a minimum ~~labelled tree~~ spanning tree in a weighted ~~labelled~~ connected graph G.

Proof: Let T be the Kruskal's tree & let T^* be a minimum spanning tree in G.

To show that $w(T) = w(T^*)$.

If $E(T) = E(T^*)$ then $T = T^*$ & we are done.

Then let $E(T) \neq E(T^*)$. So let e be the 1st edge of T which is not in T^*

$T^* \cup \{e\}$ contains a cycle, say C. We have $e \in C$. T does not contain C. $e \in E(T)$, $C \notin E(T)$

Then $\exists$ an edge e^* in C s.t $(e^* \neq e)$ s.t $e^* \notin E(T)$

Let $T_1 = T - e + e^*$. T_1 is also a spanning tree.

$w(e) \le w(e^*)$ [Kruskal's algorithm selects e then e^*]

$\Rightarrow w(T) \le w(T_1)$

If $E(T_1) = E(T^*)$, then $w(T_1) = w(T^*)$

$w(T) \le w(T_1) \le w(T^*)$

$\Rightarrow w(T) = w(T^*)$, because T^* is a min. spanning tree & $w(T) \le w(T^*)$.

?

If $E(T_1) \neq E(T^*)$ then repeat the above process for T_1 & T^*. After finite no. of steps we get a tree T_k s.t $E(T_k) = E(T^*)$ & hence we get the result.

• **Prim's Algorithm**–

Input: A connected weighted graph G.

Output: A minimum spanning tree of G.

1. Let $x \in V(G)$ & $T = \{x\}$
2. Select an edge e of minimum weight & that has one end vertex in T and another ~~end~~ end vertex in $V(G) - V(T)$.
3. $T = T \cup \{e\}$
4. If $|E(T)| = n-1$ then output T, otherwise go to step 2.

Note: T is a tree because it is acyclic & has n vertices & $n-1$ edges.

• **Theorem:** The tree obtained in Prim's algo. is a minimum spanning tree in a connected weighted graph G.

Proof: Let T be the Prim's tree & T^* be a minimum spanning tree in G. To show that $w(T) = w(T^*)$

If $E(T) = E(T^*)$, then we are done.

Otherwise, let e be the 1st edge (in order of inclusion of edges) in T ~~which~~ which is not in T^*.

Let $e = \{v, w\}$. Let T' be the subtree of T just before the ~~incl~~ inclusion of edge e. Let $v \in V(T')$.

$e \notin T^*$. Let e^* be an edge in T^* through vertex v.

$(T' \subseteq T^*)$. $w(e) \leq w(e^*)$

$T_1^* = T^* - e^* + e$. T_1^* is also a spanning tree.

$w(T^*) = w(T_1^*)$, since T^* is a minimum spanning tree.

? $\Rightarrow w(e^*) \leq w(e)$

Hence $w(e^*) = w(e)$

$\Rightarrow w(T_1^*) = w(T^*)$

Is $E(T_1^*) = E(T)$? ~~Yes~~ If yes, $w(T) = w(T_1^*) = w(T^*)$

If not repeat this process and after a finite no. of steps we get a ~~se~~ spanning tree T_k^* s.t

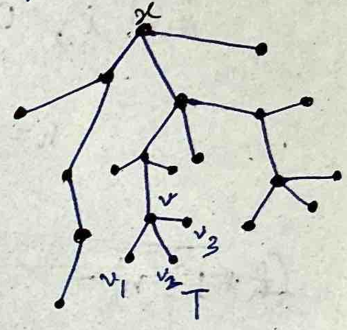
$$E(T) = E(T_k^*) \;\&\; w(T_k^*) = w(T^*)$$

and hence ~~we~~ we get the result.

• Rooted Tree :- This is a tree in which a vertex is fixed as the root vertex.

Let T be a rooted tree with x as the root vertex.

Let $v \in V(T) - x$. Let $P(v)$ be the unique path from ~~v to~~ v to x. Length of $P(v)$ is called the depth or level of v.



Height of T is the length of the longest path $P(v)$, $v \in V(T)$.

$h(T) = 5$.

• The neighbour of v in $P(v)$ is called the parent of v & the other ~~neighbour~~ neighbours of v in T are called ~~chi~~ ~~childness~~ children of V.

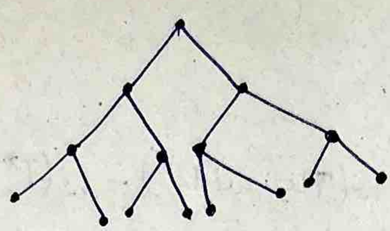
All the vertices in $P(v)$ are called ancestors of v.

If $w \in P(v) - v$. Then v is called a descendant of w.

v_1, v_2, v_3, are called siblings of each other.

A tree T is called a k-ary tree, $k \geq 2$, if all the non-pendant vertices of T have atmost k children.

If all the non-pendant vertices have exactly k ~~two~~ children is called strictly k-ary ~~tree~~ tree.



Complete binary or balanced binary tree

A strictly k-ary tree is called complete or ~~bal~~ balanced if all the leaves are at the same level.

• **Theorem:** Let T be a k-ary tree on n vertices. Let h be the height of T. Then

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

Proof: Let n_i be the no. of vertices in T at level i, $0 \le i \le h$.

$$n = \sum_{i=0}^{h} n_i \qquad 1 \le n_i \le k^i$$

$$\cancel{(h+1 \le n) \Rightarrow \; = \sum_{i=0}^{h} n_i \le \sum_{i=0}^{h} k^i}$$

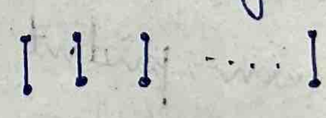
$$h+1 \le n = \sum_{i=0}^{h} n_i \le \sum_{i=0}^{h} k^i$$

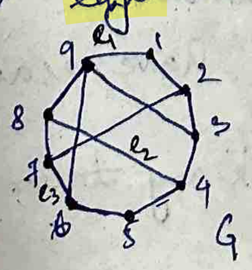
$$= \frac{k^{h+1}-1}{k-1}$$

• ~~Corol~~ Corollary: If T is a balanced k-ary tree then

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

• Matchings – A set M of independent or non-adjacent ~~ed~~ edges in a graph G is called a matching.



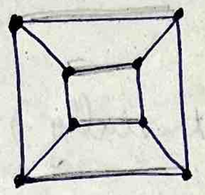


$M_1 = \{e_1, e_2, e_3\}$ is a matching in G.

M_1 saturates the vertices 1, 9, 4, 6, 7, 8.

A matching M of graph G is called a perfect matching if M saturates all the vertices of G.

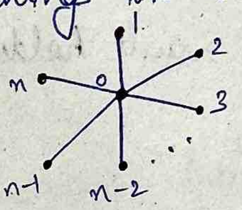
$K_{1,3}$ → has atmost 1 matching, so no perfect matching.



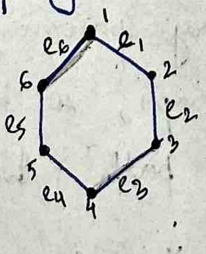
Q_3 → has perfect matching

A matching M in a graph G is called a maximum matching if M contains maximum possible no. of edges in G, i.e. M has maximum cardinality among all possible matchings in G.

$K_{n,1}$ –



M is called a maximal matching of G if M is not properly contained in any other matching of G.

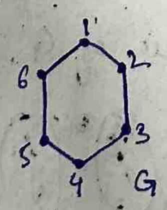


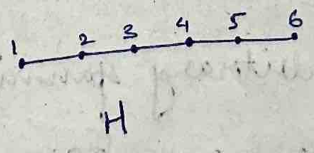
$M = \{e_3, e_6\}$. This is a maximal matching but not a maximum ~~matching~~ matching.

$M_1 = \{e_2, e_4, e_6\}$ is a maximum matching as well as perfect.

Tutorial 6 (12th Oct 2023):

1.





H is a subgraph of G but not an induced subgraph.

If graph contains a cycle, then it does not satisfy the property. In general, a forest or acyclic graph satisfies the given property.

2. T.P.T : $C(T) = K_1$ or K_2 , $C(T) \to$ center of tree

(vertex) (edge)

Pf: Obtain T_1 from T by deleting all the leaves.

Let $v \in V(T_1)$. Then $e_{T_1}(v) = e_T(v) - 1$, whenever $T_1 \neq \phi$.

Central vertices in T are non-pendant vertices provided that $T \neq K_2$ or

(If $T = K_1$ or K_2 , then result holds trivially.)

• K_1

$$\min_{\substack{v \in V(T) \\ v \text{ is non-pendant}}} (e_T(v) - 1) = \min_{u \in V(T_1)} (e_{T_1}(u))$$

—• K_2

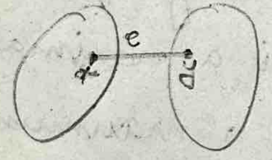
$$\Rightarrow \quad C(T) = C(T_1)$$

Similarly proceeding $T_1 \xrightarrow{p} T_2 \xrightarrow{p} \cdots \xrightarrow{p} T_k$. (p-operation-deletion of pendant vertices)

When $T_k = T_1$ or T_2 , result holds.

$C(T_k) = K_1$ or K_2

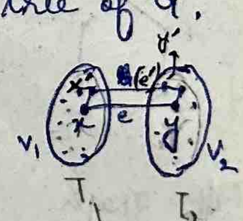
3. $G \to$ connected. $e \in E(G)$, $e \to$ bridge. Let $e = \{x, y\}$



The only path b/w x & y —e— in the graph G. Every spanning tree is a connected subgraph, so between every pair of vertices, there is a path. In the path, the only path between x & y is e. So e belongs to every spanning tree of G.

For converse part:-

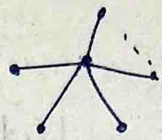
Let T be an arbitrary spanning tree of G. $e \notin E(T)$. Then T may have another path P between x & y.



$V(G) = V_1 \cup V_2$. Let $P: x = x_0, x_1, \ldots, x_i, x_{i+1}, \ldots, x_k = y$. be another x-y path in G.

(or T_1?) → $T_1' = T - e + e'$ is a spanning tree of G & $e \notin T_1'$.

4. a) Star –



b) Path –

5. No. of spanning trees in labelled $K_n - e = (n-2)n^{(n-3)}$

From Cayley's formula,

No. of spanning trees in $K_n = n^{(n-2)}$

Let C be the collection of all $n^{(n-2)}$ spanning trees in K_n. Then $|C| = n^{n-2}$.

No. of spanning trees in C which ~~cont~~ contains e

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

$$= 2n^{(n-3)}$$

Total no. of spanning trees in $K_n - e = n^{n-2} - 2n^{n-3}$

7. In strictly binary tree –

$\deg(\text{root}) = 1$

$\deg(\text{pendant vertices}) = 1$

$\deg(\text{non-pendant vertices}) = 3$

$k \to$ no. of leaves

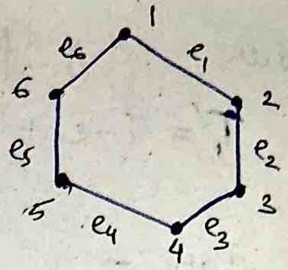
$\sum_i \deg(x) = 1 + k + 3(n-k-1) = 2(n-1)$

$\Rightarrow n = 2k - 1 \to$ ~~odd~~ odd

$\Rightarrow k = \frac{n+1}{2}$ → sol^n for Q8.

9. $n = 12$, $h = 3$

• Matchings (contd...) –



$M_1 = \{e_3, e_6\}$,

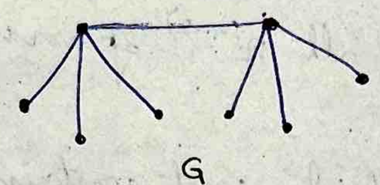
$M_2 = \{e_2, e_4, e_6\}$

$M_1 \rightarrow$ a maximal matching but not a maximum matching.

$M_2 \rightarrow$ a maximum matching.

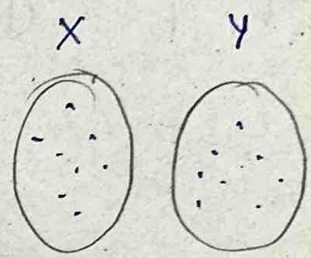
Every maximum matching is a maximal matching.

~~Perfect~~



The size of a maximum matching in G is 2.

• Job Assignment Problem –



m jobs, n applicants, $n \geq m$

• **Theorem**: (Philip Hall, 1935)

(also called Hall's marriage theorem)

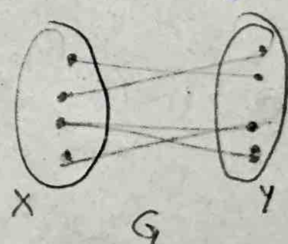
Let $G = X \cup Y$ be a bipartite graph. Graph G has a matching that saturates X iff for every subset S of X, we have $|N(S)| \geq |S|$, where N(S) is the set of all neighbours of vertices in S.

[$|N(S)| \geq |S| \rightarrow$ is known as Hall's condition.]

proof????

Corollary: Every ~~k-subgraph~~ k-regular, $k \geq 1$, bipartite graph has a perfect matching.

Proof: Let $G = X \cup Y$ be a k-regular bipartite graph.



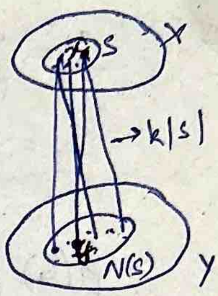
No. of edges in $G = k \cdot |X|$

No. of edges in $G = k \cdot |Y|$

$\Rightarrow k|X| = k|Y| \Rightarrow |X| = |Y|$

Let S be a subset of X.

No. of edges from S to $N(S)$ is $= k|S|$
$= m$ ~~[illegible]~~



No. of edges from $N(S)$ to $X = k \cdot |N(S)|$

For any $y \in N(S)$ and any edge $\{y, z\}$ through y, z may be in S or may not ~~be~~ be in S.

$$k \cdot |N(S)| \geq k \cdot |S|$$

$E_1 \to$ set of edges from S to $N(S)$ ~~to X~~.

$E_2 \to$ set of edges from $N(S)$ to X.

$E_1 \subseteq E_2 \Rightarrow |E_1| \leq |E_2|$

$\Rightarrow k \cdot |S| \leq k \cdot |N(S)|$

$\Rightarrow |S| \leq |N(S)| \quad \forall S \subseteq X$.

Then from Hall's Theorem $\exists$ a matching M of G s.t M saturates X. Since $|X| = |Y|$, it is a perfect matching of G.

- Some notations - For any loop-free graph G,

$\alpha(G) =$ independence no., i.e, the cardinality of a maximum independent set

$\alpha'(G) =$ edge independence number, i.e, cardinality of a maximum matching.

Defn: A subset $S \subseteq V(G)$ is called a (vertex) covering of G if S contains atleast one end vertex of every edge in G.

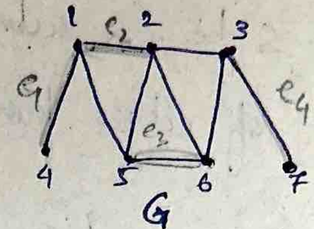
$\beta(G) = \min\{|S| \mid S$ is a vertex covering of $G\}$
↳ vertex covering no. of G

Defn: A subset $W \subseteq E(G)$ ~~[illegible]~~ $\delta(G) \geq 1$) is called an edge covering of G if W contains an edge through every vertex in G.

edge covering no. of G ← $\beta'(G) = \min\{|W| : W$ is an edge covering of $G\}$

$S_1 = \{1, 2, 3, 5\} \rightarrow$ vertex cover

$S_2 = \{1, 3, 5\} \rightarrow$ not a vertex cover



$\beta(G) = 4$, $\beta'(G) = 4$

$\alpha(G) = 3$, $\alpha'(G) = 3$ $x = \{7, 2, 4\} \rightarrow$ independent set

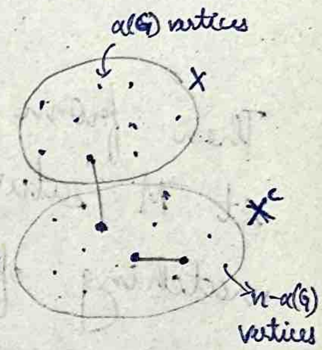
$\alpha(G) + \beta(G) = 7 = n$

$\alpha'(G) + \beta'(G) = 7 = n$

• **Theorem:** For any connected (not required) graph G, $\alpha(G) + \beta(G) = n$, where n is the number of vertices.

Proof: Let X be an independent set with $|X| = \alpha(G)$.

Since X is an independent set, for every edge l in G, either l is contained in X^c or one end vertex of l is in X, and other is in X^c.



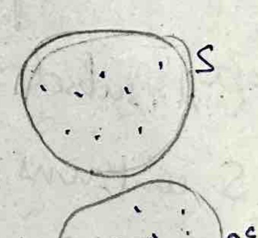
$\Rightarrow X^c$ is a vertex covering of G.

$$n - \alpha(G) = |X^c| \geq \beta(G) \quad [\because \beta(G) \text{ is minimum}]$$

$$\Rightarrow n \geq \alpha(G) + \beta(G) \quad \text{— (I)}$$

Let S be a vertex covering of G with $|S| = \beta(G)$.

S contains atleast one end vertex of every edge of G. So no two vertices in S^c are adjacent. Hence S^c is an independent set.



$$n - \beta(G) = |S^c| \leq \alpha(G)$$

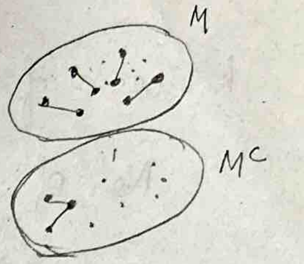
$$\Rightarrow n \leq \alpha(G) + \beta(G) \quad \text{— (II)}$$

From (I) & (II), $\alpha(G) + \beta(G) = n$

• **Theorem**: For every graph G with $\delta(G) > 0$, $\alpha'(G) + \beta'(G) = n$, where $n = |V(G)|$.

delta g is the minimum of all degrees of vertices

Proof: Let M be a matching with $|M| = \alpha'(G)$, i.e., M is a maximum matching.



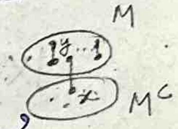
If M saturates all the vertices of G then M is an edge covering of G.

$$\alpha'(G) = |M| = \frac{n}{2} = \beta'(G)$$

So let M does not saturate all the vertices of G, check

Then $M^c \neq \phi$. If M^c contains an edge l of G then $\underline{M \cup \{l\}}$ is a matching with more edges in M, which is a contrad^n. So no two vertices in M^c are adjacent.

Let $x \in M^c$. Since x is not an isolated vertex, $\exists$ an edge e_x in G s.t $e_x = \{x, y\}$, $y \in V(M_y)$.



Let $W = \{e_x : x \in M^c,\ e_x$ is an edge through $x\}$, ~~$Y = M \cup W$~~

$$|W| = |M^c| = n - 2\alpha'(G).$$

(No. of vertices covered by $M = 2\alpha'(G)$.
No. of vertices in $M^c = n - 2\alpha'(G)$)

Then $Y = M \cup W$ is an edge covering of G.

$$\Rightarrow |Y| \geq \beta'(G)$$

$$|Y| = |M| + |W| = \alpha'(G) + n - 2\alpha'(G)$$
$$= n - \alpha'(G)$$

So $n - \alpha'(G) \geq \beta'(G)$ —— (I)

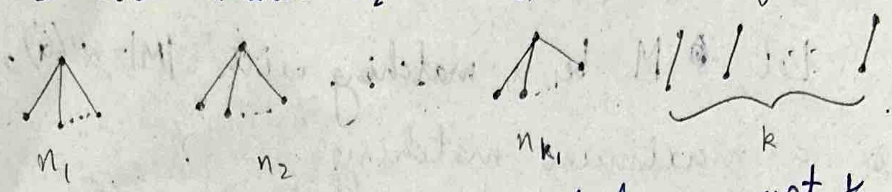
$$\Rightarrow n \geq \alpha'(G) + \beta'(G)$$

Let Q be an edge covering of G with $|Q| = \beta'(G)$.

Since Q is a minimum edge covering,

Edges in Q do not form cycles or paths of length greater than equal to 3.

So the connected components formed by the edges in $\mathcal{G}$ are either K_2 or $K_{n_i,1}$, $n_i \geq 2$



k_1 no. of star components which are not K_2.

No. of K_2 is $= k$

$M = \{l_1, l_2, \ldots, l_k; e_1, e_2, \ldots, e_{k_1}\}$

M is a ~~matching~~ matching of G.

$|M| \leq \alpha'(G)$

$\boxed{|M| = k + k_1}$, $n = 2k = \boxed{\sum n_i + k_1}$ check

$|X| = n - 2|M| = |\mathcal{Q}| - |M|$

X → set of vertices in G unsaturated by M.

$\Rightarrow |M| = n - |\mathcal{Q}| = n - \beta'(G)$

$n - \beta'(G) = |M| \leq \alpha'(G)$?

$\Rightarrow n \leq \alpha'(G) + \beta'(G)$ — (II)

From (I) and (II),

$n = \alpha'(G) + \beta'(G)$

- Connectivity in Graphs -

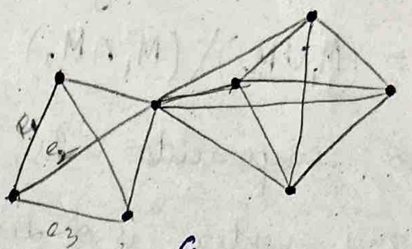
Defn: Let G be a connected and non-complete graph.
The vertex connectivity of G, denoted by $\kappa(G)$, is the minimum number of vertices (in G); removal of which makes the graph disconnected.

Similarly, we define edge-connectivity, denoted by $\kappa'(G)$, ~~is~~ the min. no. of edges, the removal of which makes G disconnected.

We take $\kappa(K_n) = n-1 = \kappa'(K_n)$

Construct a network (graph G) on 8 vertices & 16 edges with maximum possible vertex connectivity & edge connectivity.

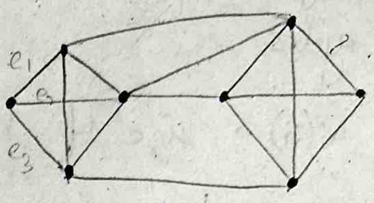
$n=8$, $e=16$.



G₁

$\kappa(G_1) = 1$, $\kappa'(G_1) = 3$

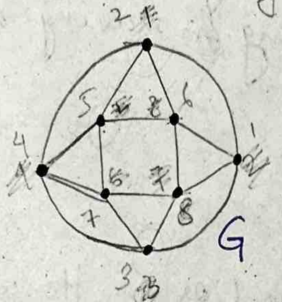
edge cut $= \{e_1, e_2, e_3\}$



G₂

$\kappa'(G_2) = 3 = \kappa(G_2)$

edge cut $= \{e_1, e_2, e_3\}$

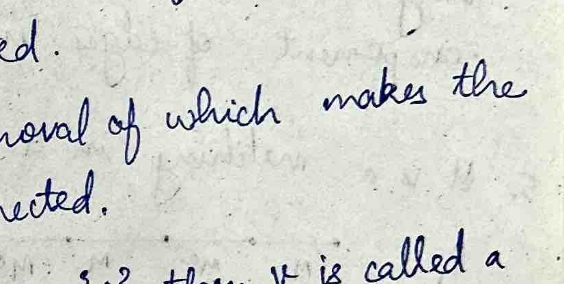


G

$\kappa(G) = 4$, $\{2,3,5,7\} \to$ vertex cut

- vertex cut – set of vertices removal of which makes the graph disconnected.
- edge cut – set of edges removal of which makes the graph disconnected.
- vertex cut is a single vertex $\{v\}$, then v is called a cut vertex.
- Edge-cut is a single edge $\{l\}$, then l is called a cut-edge or bridge.

Tutorial Sheet-7 (19th Oct 2023):



1. C_n –

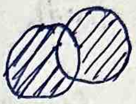
Minimum size of ~~minimal~~ maximal matching $= \left\lceil \frac{n}{3} \right\rceil$

$V(C_n) = \{1, 2, \ldots, n\}$, ~~We can take edges {1,2}, {4,5}, {7,8} and so on.~~

2. T.S.T: Every tree has atmost one perfect matching.

Let T be having two perfect matchings M_1 & M_2.

$M_1 \Delta M_2 \rightarrow$ symmetric difference



$= (M_1 \cup M_2) \setminus (M_1 \cap M_2)$

~~The components~~ In the connected components of $M_1 \Delta M_2$ every vertex is of deg 2. Hence the components induce cycles

3. $\alpha'(G) \rightarrow$ size of maximal matching.

Let M_1 be a maximal matching having $\alpha'(G)$ no. of edges. $|M_1| \geq \frac{\alpha'(G)}{2}$

4. M_1, M_2 are matchings.

Every vertex in $M_1 \Delta M_2$ has degree 1 or 2. Hence a component of edges in $M_1 \Delta M_2$ is a path or cycle.

5. M is a matching in G.

DO yourself

P_1: $M \quad M^c \quad M \quad M^c \quad M \quad M^c$ $\rightarrow$ M-alternating path

P_2: x $M^c \quad M \quad M^c \quad M \quad M^c$ y $\rightarrow$ M-augmenting path

x, y are unsaturated by edges in M.

M is a maximum matching $\Leftrightarrow$ G contains no M-augmenting path.

Let G be not having any M-augmenting path.

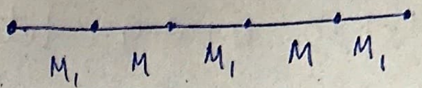
To prove M is a maximum matching.

Suppose M is not a maximum matching. Then let M_1 be a maximum matching.

$|M_1| > |M|$. Look at the components of ~~$M_1 \Delta M_2$~~ $M_1 \Delta M$. ~~Conversely~~ Conversely let M be not a ~~maximum~~ matching ~~We show~~ that G contains an

M-augmented path. Then $\exists$ a matching M_1 s.t $|M_1| > |M|$

Components of $M_1 \Delta M$ -

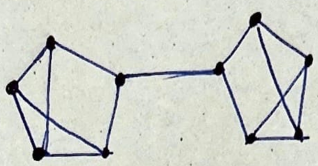


A component of $M_1 \Delta M$ contains an M-augmented path.

6. G is 3-regular, $\kappa(G) = 1$.

T.P.T: $|V(G)| \geq 10$

Let x be a cut vertex. $G - x$ is disconnected.

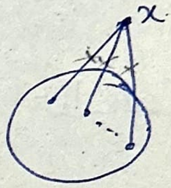


→ example

• Theorem (Whitney): For every simple connected graph G,

$$\kappa(G) \leq \kappa'(G) \leq \delta(G).$$

Proof: $\kappa'(G) \leq \delta(G)$ is obvious. $x \in V(G)$, $deg(x) = \delta(G)$. Then



S → set of all edges incident on x. Then

S is ~~an~~ an edge-cut. So $\kappa'(G) \leq |S| = \delta(G)$

$\Rightarrow \kappa'(G) \leq \delta(G)$

$\kappa(G) \leq \kappa'(G)$

To show - $\kappa(G) \leq \kappa'(G)$

Let S be an edge-cut with $|S| = \cancel{q}$ $\kappa'(G) = m$, Then

$G - S$ is disconnected.

We show that $\kappa(G) \leq m$. $G - S$ has two components, say G_1 & G_2.

Let S_i, $i = 1, 2$ be the set of end vertices of edges in S which belong ~~to~~ to G_i.

$|S_1|, |S_2| \leq m = |S|$

$\boxed{S_1 = V(G_1) \quad S_2 = V(G_2)}$

If $S_1 \neq V(G_1)$ or $S_2 \neq V(G_2)$ or both, then S_1 or S_2 is a vertex-cut of G. So we get $\kappa(G) \leq |S_1|$ or $|S_2|$ $= m$

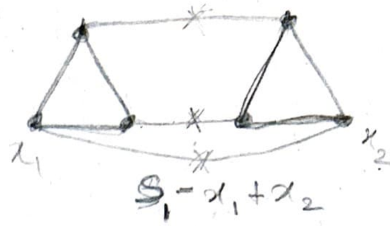
Next we have $S_1 = V(G_1)$ & $S_2 = V(G_2)$. $l = \{x_1, x_2\}$

$|S| \geq 2$ W.L.O.G. Let $l \in S$ with

$l = \{x_1, x_2\}$. $S_1' = S_1 - x_1 + x_2$

is a ~~...~~ vertex-cut.

and $|S_1'| = |S_1| \leq m$

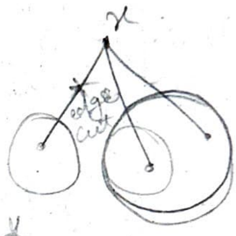


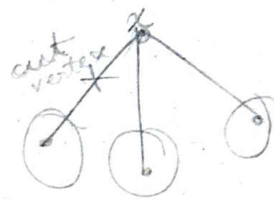
Tutorial 7 (30th Oct 2023):

7. $\kappa(G) \leq \kappa'(G) \leq 3$

If $\kappa(G) = 3 \Rightarrow \kappa'(G) = 3$

Case I - If $\kappa(G) = 1$,





Case II - $\kappa(G) = 2$