Recap :

connected graph : $\forall\, u, v .\ \exists$ path from u to v.

2 - vertex - connected : graph remains connected even after removal of any vertex

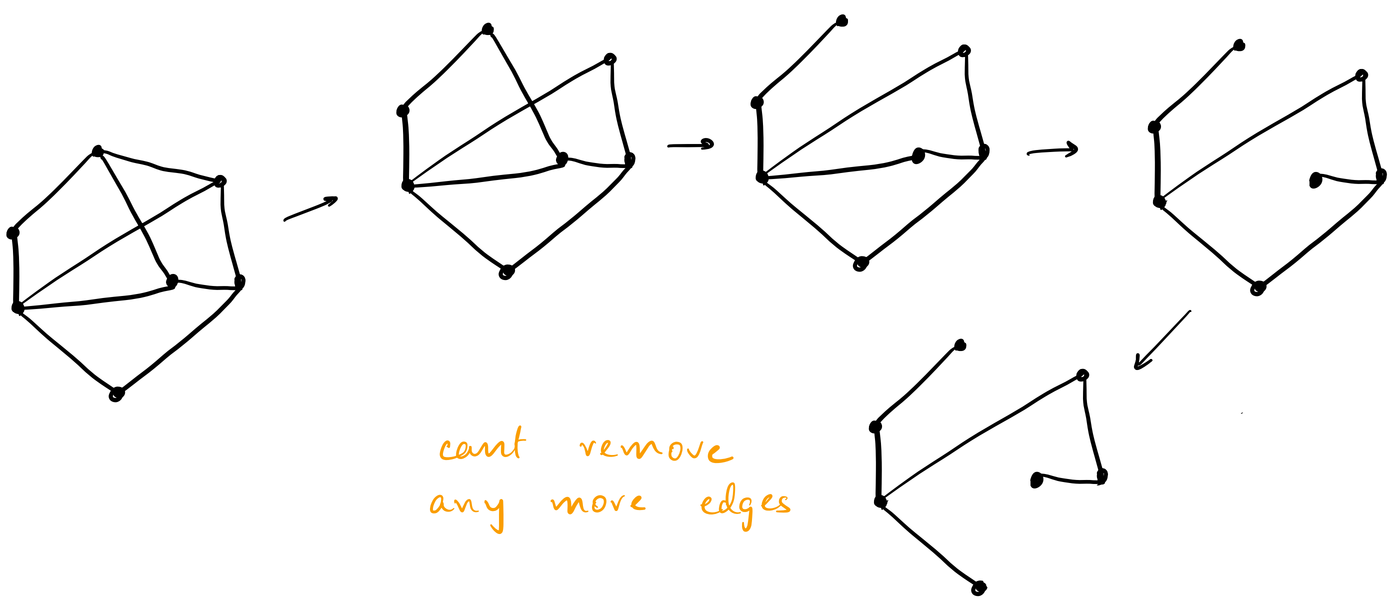
2 - edge - connected : graph remains connected even after removal of any edge.

Today : given $G = (V, E)$, we want a MINIMAL CONNECTED SUBGRAPH $H = (V, F)$

as few edges as possible

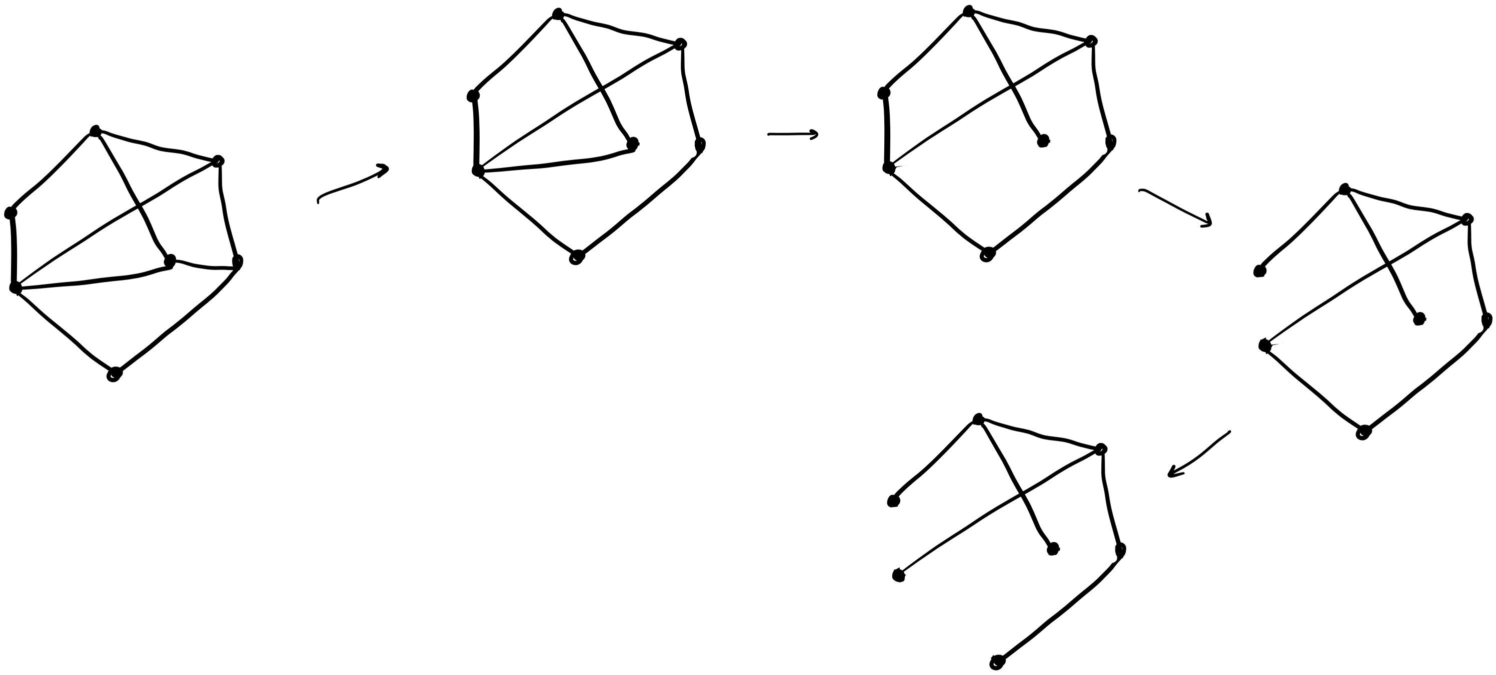
$F \subseteq E$

G $\xrightarrow{\text{remove edges}}$ H removing even one edge disconnects graph



cant remove any more edges

If we chose a different subset to remove, we'd get a different end graph.



We can end up with different subgraphs, but what do these have in common?

- if we remove any edge, graph gets disconnected
- have no cycles
- have same number $(n-1)$ edges
- between any two vertices, $\exists$ unique path
- if we add an edge to the graph, then we'll have a cycle in the graph.

All the above properties are equivalent ways of defining an important sub-class of graphs called TREES. We will prove that these properties are equivalent.

Thm 26.1 : Let $G = (V, E)$ be a connected graph. The following properties are equivalent.

(1) G has no cycles.

(2) Between any two vertices, there is a unique path.

(3) G is a maximal cycle-free graph : if we add any new edge, the graph will have a cycle

(4) G is a minimal connected graph : if we remove any edge, graph will get disconnected

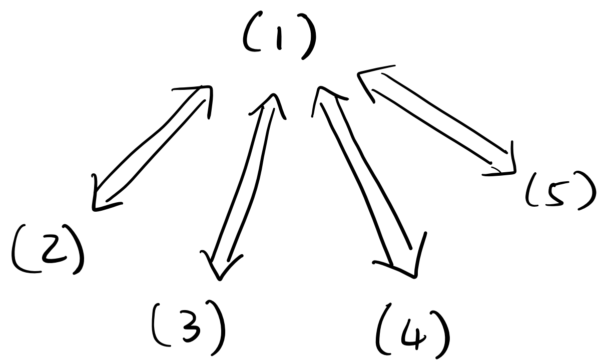
(5) $|E| = |V| - 1$.

How to prove that several (say t) statements are equivalent?

Naive : consider all ${}^tC_2$ pairs of statements, prove their equivalence.

Better : Think of the t statements as t vertices. Find a tree T on these t vertices. If $\{i, j\}$ is an edge in the tree T, then prove equivalence of Stmt. i & Stmt. j. Need to prove only $(t-1)$ equivalences!

We will show



First, let us see what information can be derived from (1).

<u>Obs. 1</u>: If $G=(V,E)$ is connected and has no cycle, then there exists a vertex of deg exactly 1.

We already proved this when we showed that any connected graph where every vertex has even degree, has a cycle.

In fact, we can prove that if G is connected and has no cycle, then there are at least two vertices with deg. 1.

Obs 2: Let G be any graph, and let v be a vertex of deg. 1. Let G' be the graph obtained by deleting v.

G is connected and has no cycle

$\Updownarrow$

G' is connected and has no cycle.

Pf: If G is connected and has no cycle, then removing the vertex v will not create a cycle. Can it disconnect the graph? Take any $x, y \in V \setminus \{v\}$. Since v has deg 1, the path from x to y in G does not involve v. As a result, this path exists even in G'.

If G' is connected and has no cycle: then G remains connected after v is added to G'. Also note that since v has deg. 1, it cannot be part of any cycle. As a result, G does not contain a cycle.

∎

In order to prove Theorem 26.1, we need to prove (1) $\Rightarrow$ (2), (3), (4), (5), and (2) $\Rightarrow$ (1), (3) $\Rightarrow$ (1), (4) $\Rightarrow$ 1 and (5) $\Rightarrow$ 1.

<u>Part 1:</u> (1) $\Rightarrow$ (2), (3), (4), (5)

We will prove this using regular induction.

$P(n)$: for any cktd. graph $G = (V, E)$ on n vertices if G has no cycles, then $|E| = n-1$, removal of any edge disconnects the graph, addition of any edge results in a cycle, and $\forall$ u, v, there is a unique path from u to v.

Base case: $n = 1$ ✓

Suppose $P(n-1)$ holds.
Consider any graph $G = (V, E)$ on n vertices s.t. $G = (V, E)$ has no cycle. By Obs 1, G has a vertex of deg. 1, say v.

Let G' be the graph obtained by removing v.

Using Obs 2, we know G' is cktd. and has no cycles.

Hence, using $P(n-1)$, we get that

(a) G' has $n-2$ edges.

(b) G' gets disconnected if any edge is removed

(c) G' has a cycle if any edge is added.

(d) $\forall\, x, y$, there is a unique path from x to y in G'.

From (a), it follows that G has $n-1$ edges (recall, G' was obtained by deleting one vertex in G, which resulted in the deletion of one edge)

From (b), we get that G gets disconnected if some edge is removed. The edge being removed can either be the edge connecting v to G', or it can be one of the edges in E'. If the edge connecting v to G' is removed then clearly the graph gets disconnected. If any other edge is removed, then from (b) we get that G gets disconnected

From (d), we get that there is a unique path from x to y in G', for every $x, y \in V'$. This path exists in G too, and therefore, there exists at least one path from x to y. Since v has deg. 1, adding v to G' will not increase the number of paths from x to y. Finally, note that there's a unique path from v to every other vertex in V.

Adding an edge $\{x, y\}$ to the graph: we know that there exists a (unique) path from x to y in G. As a result, adding $\{x, y\}$ edge gives us a cycle.

Therefore, we have shown that (1) $\Rightarrow$ (2), (3), (4), (5).

$\blacksquare$

<u>Part 2</u>: (2) $\Rightarrow$ (1), (3) $\Rightarrow$ (1), (4) $\Rightarrow$ (1), (5) $\Rightarrow$ (1).

(2) $\Rightarrow$ (1): Follows from def.^{n} of path and cycle. If there is a cycle in G, then there exist two paths from x to y, for all distinct vertices x, y in the cycle.

(4) $\Rightarrow$ (1) : Suppose there is a cycle in G, then deletion of any edge in the cycle will not disconnect the graph.

(3) $\Rightarrow$ (1) : Since G is the maximal cycle free graph, G is clearly cycle-free. Moreover, it has to be connected. If it was not connected, then we can add an edge across components, without creating a cycle.

(5) $\Rightarrow$ (1) : This proof involves induction.

First, note that (5) implies that G has a deg. 1 vertex. Since G is connected, every vertex has deg. ≥ 1. We also know that

$$\sum_{v \in V} \deg(v) = 2|E| = 2n-2.$$

If every vertex has deg. ≥ 2, then

$$\sum_{v \in V} \deg(v) \geq 2n.$$

$P(n)$: For any cktd. graph on n vertices, if $|E| = n-1$, then G has no cycle.

Base case : $n = 1$ ✓

Induction step : $P(n-1) \Rightarrow P(n)$.

Take any cktd graph on n vertices s.t. $|E| = n-1$. This graph has a deg. 1 vertex, say v.

Let G' be the graph obtained by removing v. G' has $n-2$ edges, and is connected. Hence, using $P(n-1)$, G' has no cycle.

Using Obs. 2, if G' has no cycle and is connected, then G also has no cycle.

$\blacksquare$

Several computational problems, which are hard for general graphs, have efficient algorithms for trees. Consider the problem of "Tree Isomorphism". Given two trees $T_1 = (V_1, E_1)$, $T_2 = (V_2, E_2)$, output 1 iff $\exists$ bijection $f: V_1 \to V_2$ s.t. $\forall\, u, v \in V_1$, $(u, v) \in E_1 \iff (f(u), f(v)) \in E_2$.

For general graphs, we don't have efficient algorithms for graph isomorphism. However, for trees, we have efficient algorithms to check if two trees are isomorphic.

Here, we will consider "rooted trees".

$(T_1 = (V_1, E_1), r_1) \cong (T_2 = (V_2, E_2), r_2)$ the two rooted trees are isomorphic

iff $\exists\ f : V_1 \to V_2$ s.t.

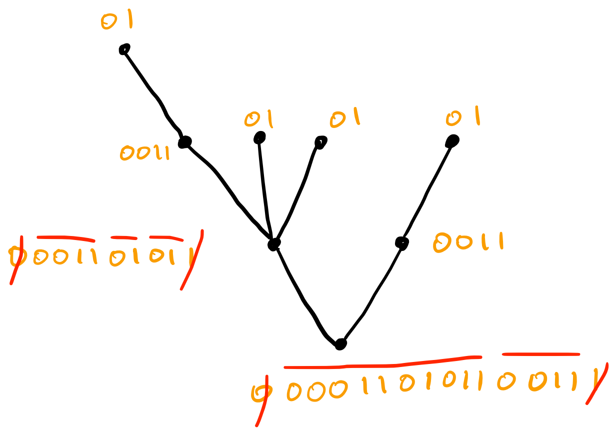
$f(r_1) = r_2$ and for all $u, v \in V_1$,

$(u, v) \in E_1 \iff (f(u), f(v)) \in E_2$.

We will assign a binary codeword c for every rooted tree. This codeword has length $2n$, and if $(T_1, r_1) \cong (T_2, r_2)$, then they will have the same codeword.

We will start with the leaves, and assign them "01". Suppose a node w has t children $T_1, T_2, \dots, T_t$, and let c_i be the code for T_i. The code for the tree rooted at w will be $0\ \mathrm{sort}(c_1 \cdots c_t)\ 1$.

Example :



Prove that this gives a unique codeword to every family of isomorphic rooted trees.

— End of Trees —

BIPARTITE GRAPHS :

Next, we consider another well-studied class of graphs – bipartite graphs.

<u>Def</u> : $G = (X \cup Y, E)$ is a bipartite graph if for all $x_1, x_2 \in X$, $\{x_1, x_2\} \notin E$, for all $y_1, y_2 \in Y$, $\{y_1, y_2\} \notin E$.

<u>Theorem 26.2</u> : A graph $G = (V, E)$ is bipartite iff it has no odd length cycle.

Proof : If G is bipartite, then clearly, there cant be an odd length cycle.

If G has no odd length cycle, and suppose G has k connected components. Let v_i be a vertex in i^{th} connected component. Consider the BFS tree rooted at v_i. Note : in a BFS tree, all edges are either at same level, or from level j to level $j+1$ / $j-1$. Since there are no odd length cycles, there are no edges at same level.

Hence we can partition the i^{th} component into 2 partitions : vertices at odd levels of the BFS tree, and vertices at even levels.

$\blacksquare$

Next lecture : matchings in bipartite graphs.