Discrete Structures– Graphs: Trees

Dr. Ab Mosca (they/them)

Plan for Today

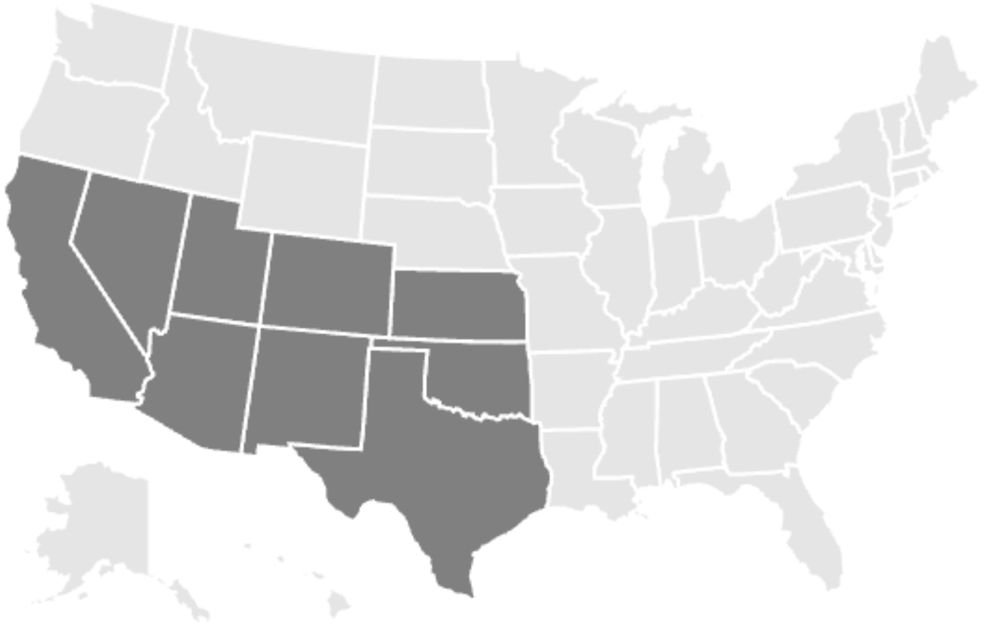
- Trees
  - Definition
  - Properties
  - Rooted
  - Spanning

Warm Up: Euler Paths Circuits

An ***Euler path***, in a graph or multigraph, is a path through the graph which uses every edge exactly once.

An ***Euler circuit*** is a Euler path which stops and starts at the same vertex.

You and your friends want to tour the southwest by car. You will visit the nine states highlighted, with the following rather odd rule: you must cross each border between neighboring states exactly once (so, for example, you must cross the Colorado-Utah border exactly once). Can you do it? If so, does it matter where you start your road trip?

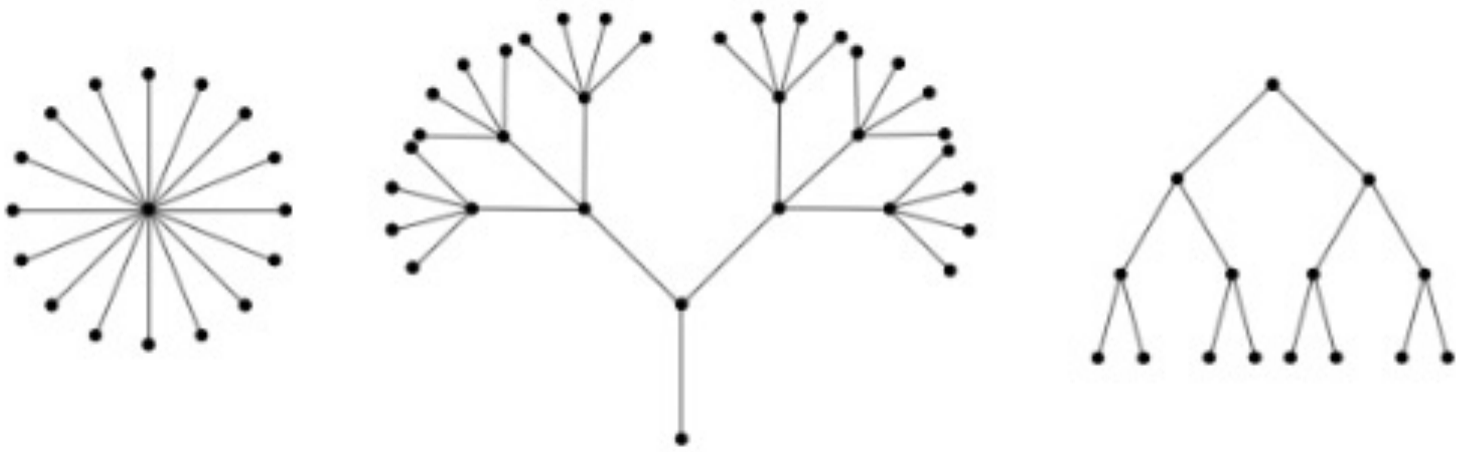


Definition: Tree

A ***tree*** is a connected graph containing no cycles.

A ***forest*** is a graph containing no cycles.

Note: this means a connected forest is a tree.



https://ptwiddle.github.io/Graph-Theory-Notes/s_intro_trees.html

Tree Properties

Proposition: A graph, T, is a tree if and only if between every pair of distinct vertices of T there is a unique path.

Corollary: A graph, F, is a forest if and only if between any pair of vertices in F there is as most one path.

Tree Properties

<u>Proposition:</u> A graph, T, is a tree if and only if between every pair of distinct vertices of T there is a unique path.

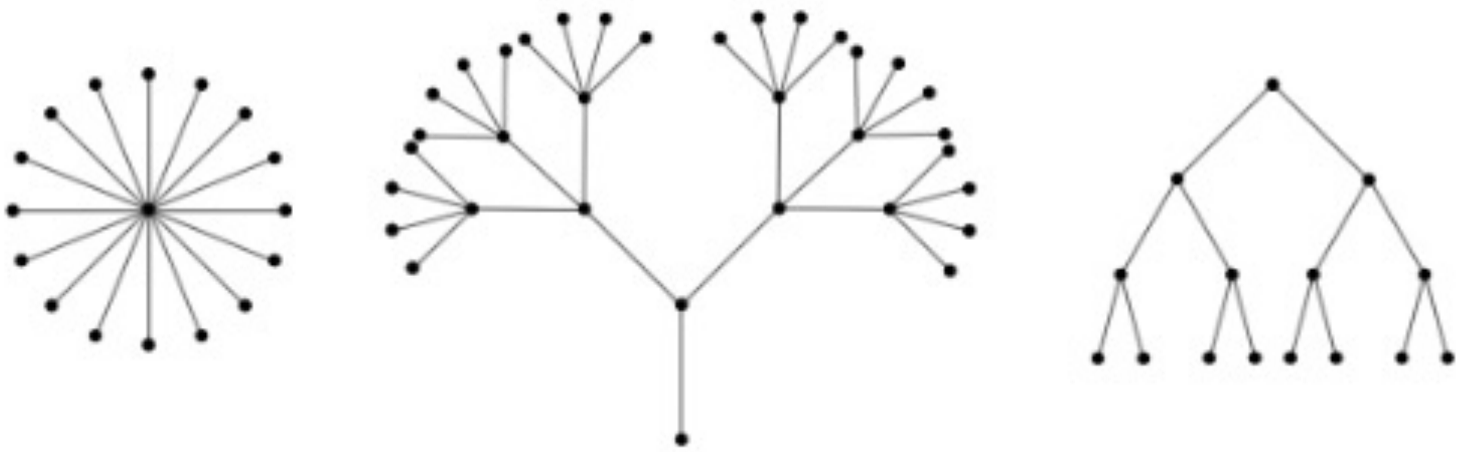
<u>Corollary:</u> A graph, F, is a forest if and only if between any pair of vertices in F there is as most one path.

Which of the following are trees?

- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{a, e\}, \{b, c\}, \{c, d\}, \{d, e\}\}$
- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{b, c\}, \{c, d\}, \{d, e\}\}$
- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{a, c\}, \{a, d\}, \{a, e\}\}$
- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{a, c\}, \{d, e\}\}$

Tree Properties

In a tree, vertices of degree one are called ***leaves***.



Which vertices in these trees are leaves?

Tree Properties

<u>Proposition:</u> Any tree with at least two vertices has at least two vertices of degree one.

<u>Proposition:</u> Let T be a tree with v vertices and e edges. Then $e = v - 1$.

Tree Properties

<u>Proposition:</u> Any tree with at least two vertices has at least two vertices of degree one.

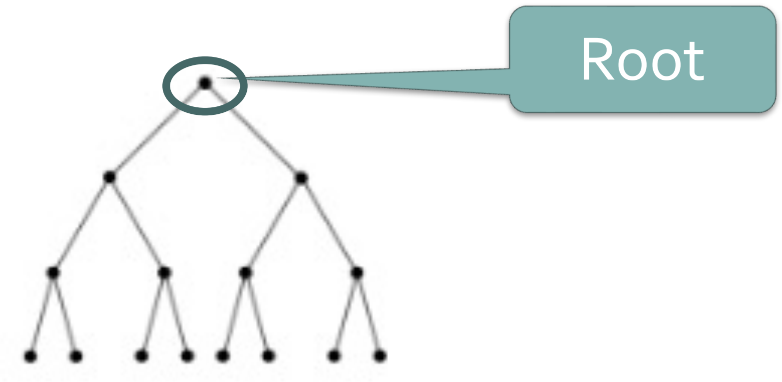
<u>Proposition:</u> Let T be a tree with v vertices and e edges. Then $e = v - 1$.

Which of the following violate the second proposition?

- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{a, e\}, \{b, c\}, \{c, d\}, \{d, e\}\}$
- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{b, c\}, \{c, d\}, \{d, e\}\}$
- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{a, c\}, \{a, d\}, \{a, e\}\}$
- $V = \{a, b, c, d, e\}, E = \{\{a, b\}, \{a, c\}, \{d, e\}\}$

Rooted Trees

We can identify one vertex in a tree as the ***root***. Then, every other vertex on the tree can be characterized by its position relative to the root.

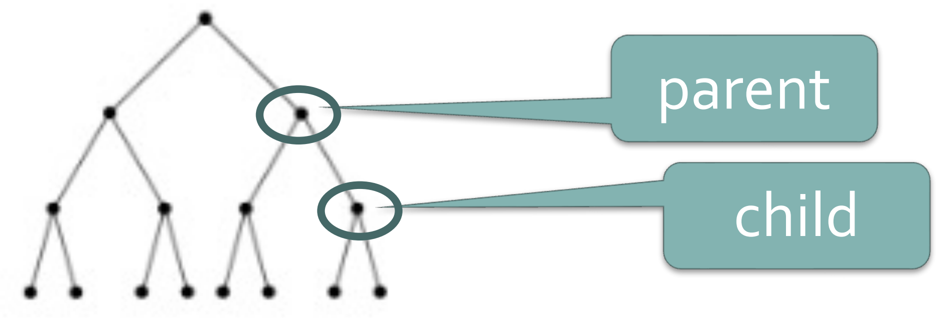


Rooted Trees

We can identify one vertex in a tree as the ***root***. Then, every other vertex on the tree can be characterized by its position relative to the root.

If two vertices are adjacent, we say the one closer to the root is the ***parent***, and the other is the ***child***.

The root of a tree is a parent, but not a child of any vertex. All non-root vertices have exactly one parent.

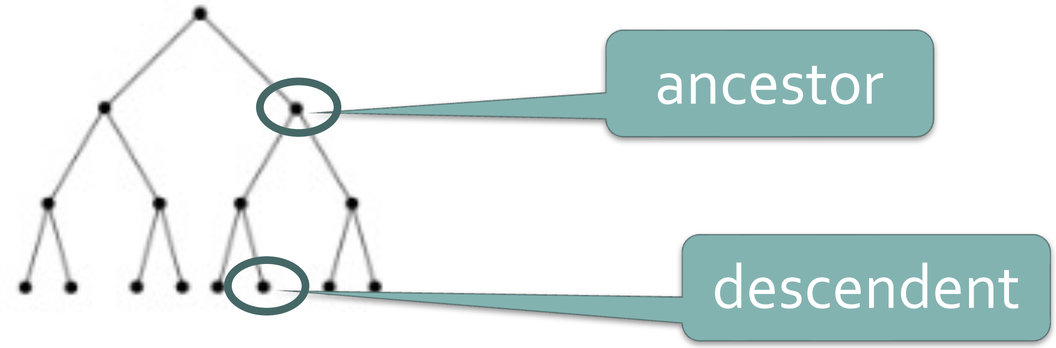


Rooted Trees

We can identify one vertex in a tree as the ***root***. Then, every other vertex on the tree can be characterized by its position relative to the root.

If two vertices are adjacent, we say the one closer to the root is the ***parent***, and the other is the ***child***.

In general, we say a vertex, v, is a ***descendent*** of a vertex, u, provided u is a vertex on the path from v to the root. Then, we would call u an ***ancestor*** of v.



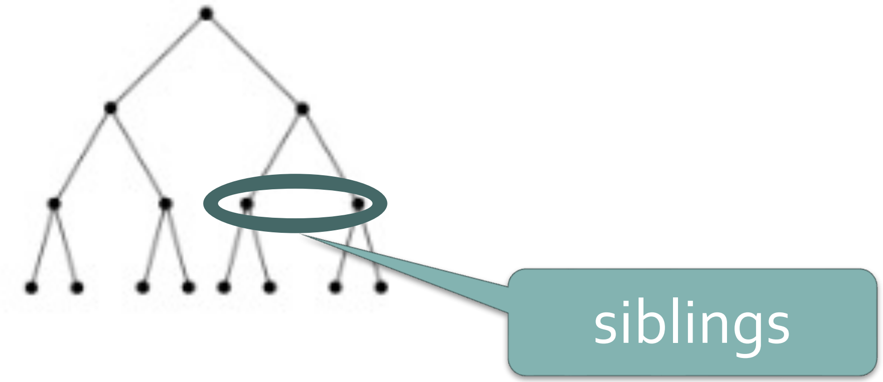
Rooted Trees

We can identify one vertex in a tree as the ***root***. Then, every other vertex on the tree can be characterized by its position relative to the root.

If two vertices are adjacent, we say the one closer to the root is the ***parent***, and the other is the ***child***.

In general, we say a vertex, v, is a ***descendent*** of a vertex, u, provided u is a vertex on the path from v to the root. Then, we would call u an ***ancestor*** of v.

Vertices with the same parent are called ***siblings***.



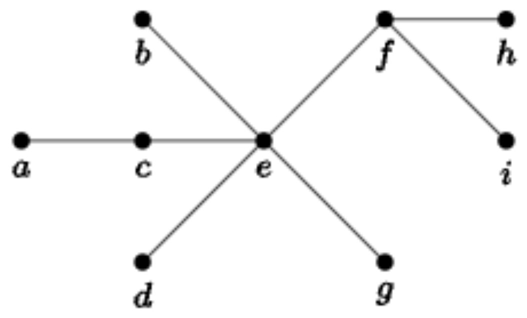
Rooted Trees

We can identify one vertex in a tree as the ***root***. Then, every other vertex on the tree can be characterized by its position relative to the root.

If two vertices are adjacent, we say the one closer to the root is the ***parent***, and the other is the ***child***.

In general, we say a vertex, v, is a ***descendent*** of a vertex, u, provided u is a vertex on the path from v to the root. Then, we would call u an ***ancestor*** of v.

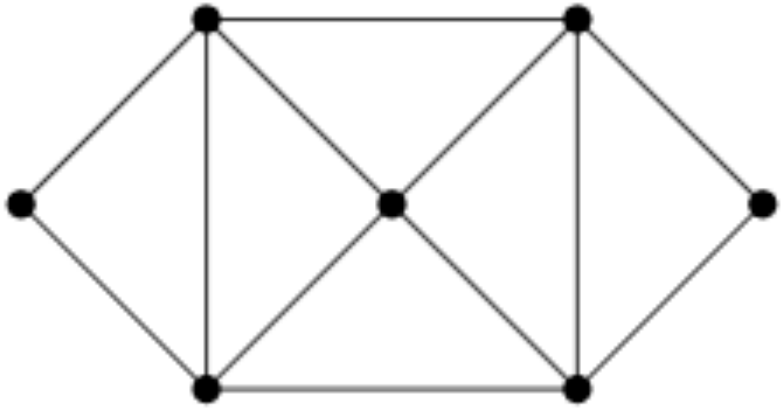
Vertices with the same parent are called ***siblings***.



Let f be the root. Label the other vertices.

Spanning Trees

Given a graph, G, a ***spanning tree*** of G is a subgraph of G which is a tree and includes all the vertices of G.



Find two different spanning trees of this graph.

Spanning Trees

Given a graph, G, a ***spanning tree*** of G is a subgraph of G which is a tree and includes all the vertices of G.

Every connected graph has a spanning tree.

Prove this claim.