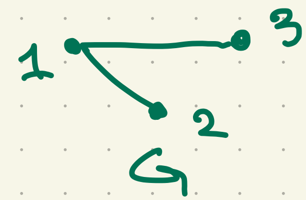
Rooted Trees

CMPT 225

Graphs

Graph: a pair $G = \langle V, E \rangle$, with
- V a set called "vertices" or "nodes"
- E a set of pairs from V, ie $E \subseteq V \times V$, called edges.

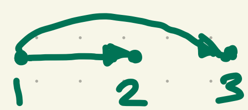
Eg: $G = \langle \{1,2,3\}, \{(1,2),(1,3)\} \rangle$

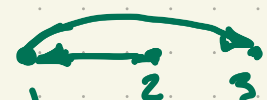


$(*)$

"G is directed": edges are ordered pairs (often called "arcs"):

$\langle \{1,2,3\}, \{(1,2),(1,3)\} \rangle \neq \langle \{1,2,3\}, \{(2,1),(1,3)\} \rangle$





"G is undirected": edges are sets

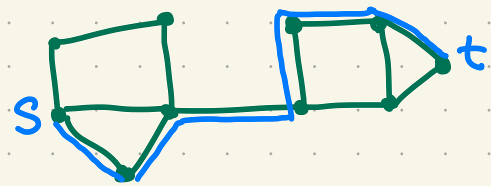
$\langle \{1,2,3\}, \{(2,1),(1,3)\} \rangle = \langle \{1,2,3\}, \{(1,2),(3,1)\} \rangle = (*)$

- By default, by "graph" we will mean "undirected graph"

<u>Path</u> in G of length n:

- sequence $\langle v_0, v_1, v_2, \dots v_n \rangle$ of vertices s.t. $(v_0,v_1),(v_1,v_2),(v_2,v_3)\dots$ are edges of G.

<u>s-t Path</u> in G: path $\langle s, \dots, t\rangle$ in G.

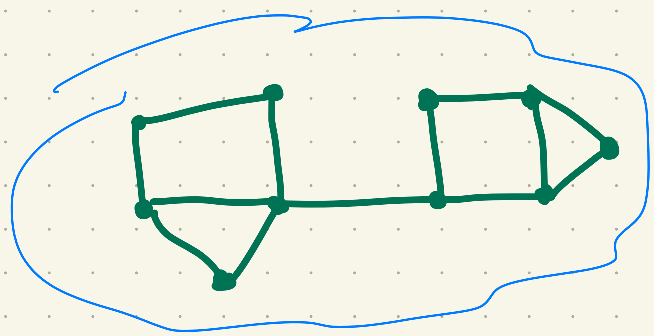


A s-t path of length 6 in G.

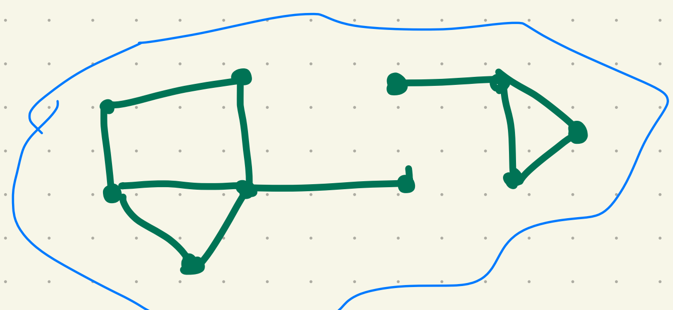
G.

Vertex t is <u>reachable</u> from s in G if there is an s-t path in G.

G is <u>connected</u> if for every pair $u,v \in V$, u is reachable from v.



Connected



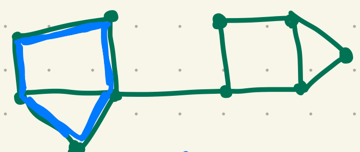
Not Connected.

Cycles & Trees

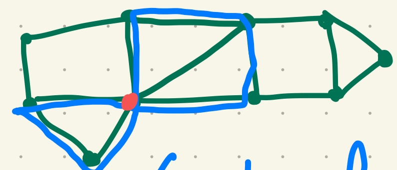
Cycle in G: path $\langle v_0, \dots v_{n-1}, v_n \rangle$ in G where $v_0 = v_n$.

Simple Path: all vertices are distinct

Simple Cycle: cycle $\langle v_0, \dots v_{n-1}, v_n \rangle$ where $\langle v_0, \dots, v_{n-1} \rangle$ is a simple path

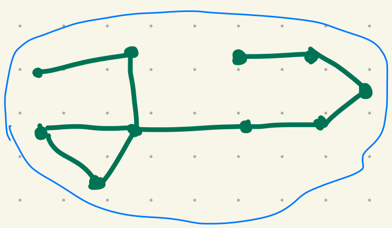


Simple cycle of length 5

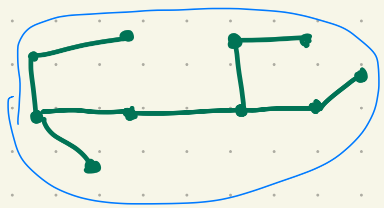


Cycle of length 7 (• repeats)

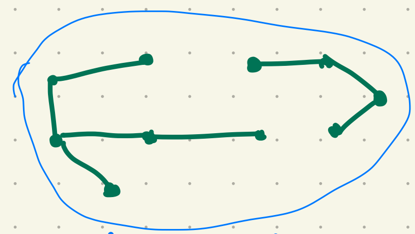
Tree: A connected, acyclic, graph.



Not a tree



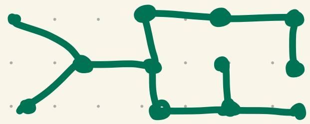
Tree



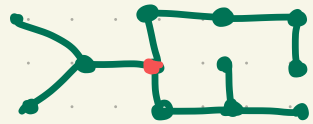
Not a tree

Fact: Every tree with n vertices has $n-1$ edges.

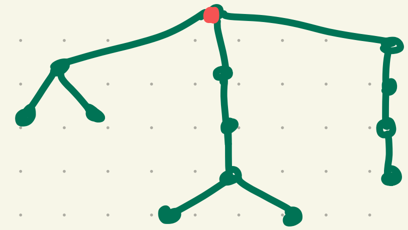
Rooted tree: tree with a distinguished vertex called the root.



Unrooted tree



Tree T with root •



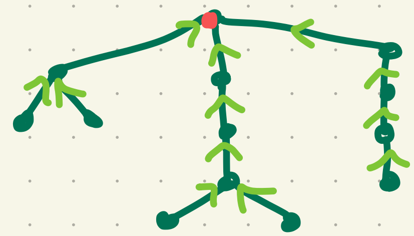
Alternate drawing of T.

Notice: in a tree there is a unique path between any two vertices.

So: a unique path from any vertex to the root.

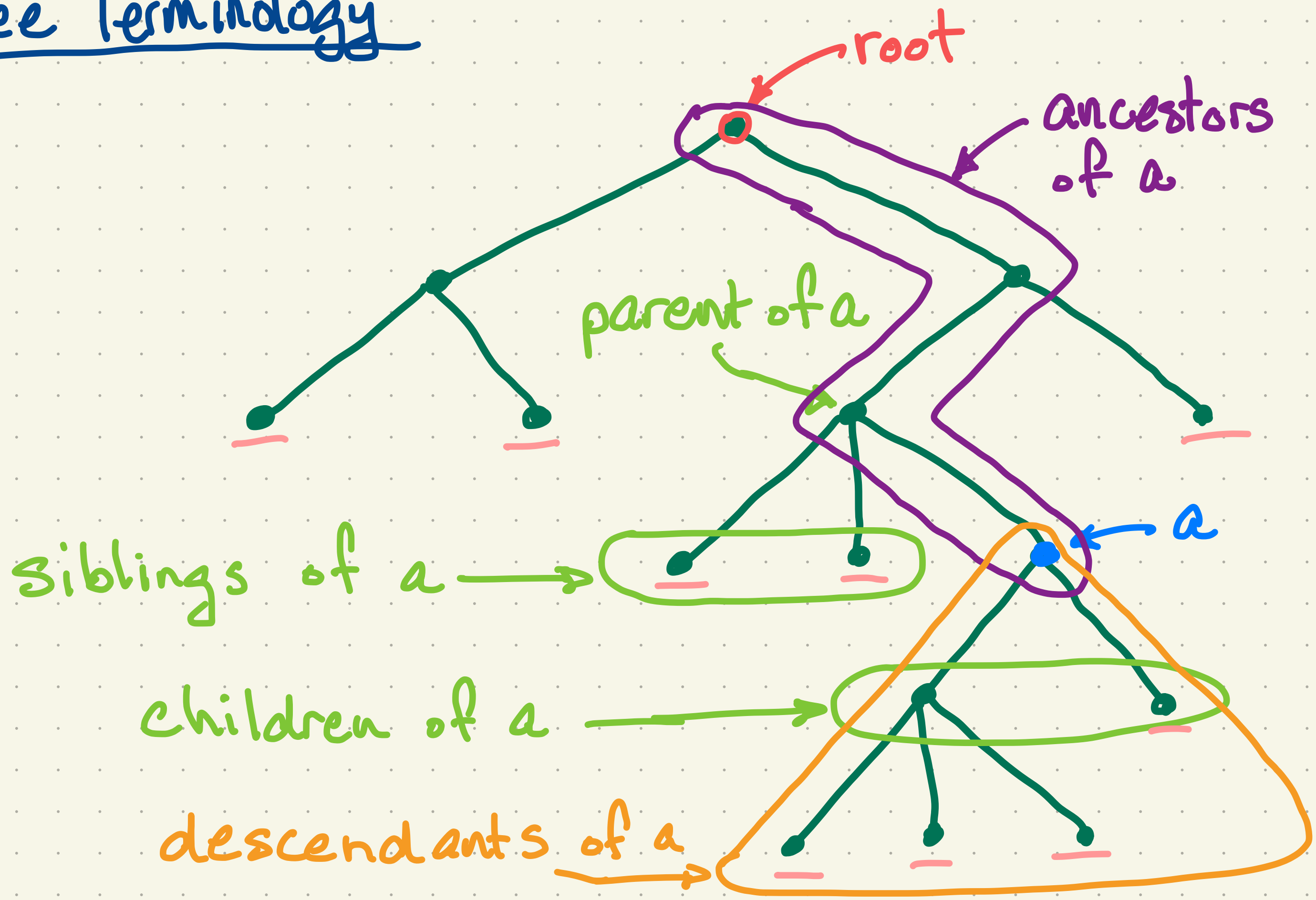
Thus: the root induces a direction on the edges

- e.g. toward the root.



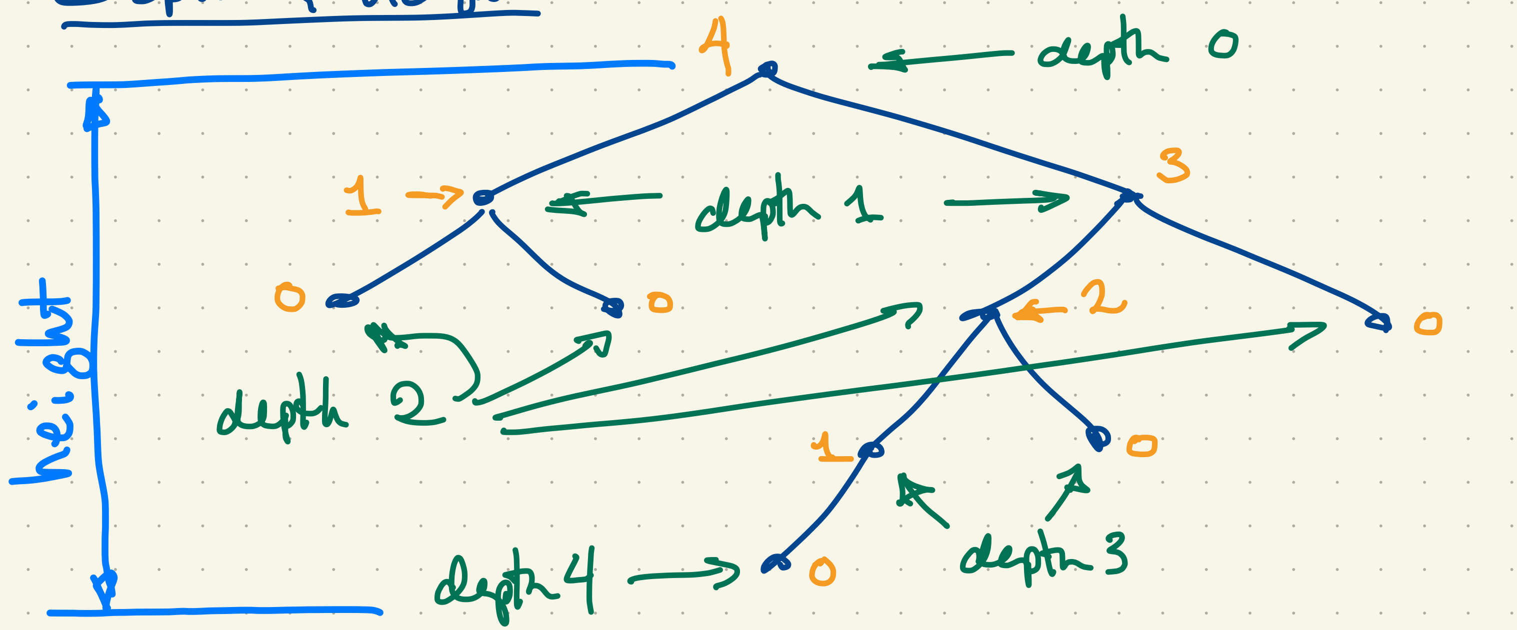
(sometimes "away from the root").

Rooted Tree Terminology



- The root has no parent;
- Leaves have no children
- Internal nodes are the non-leaves (sometimes root excluded too).

Depth & Height



Depth of node v = length of path from v to the root.

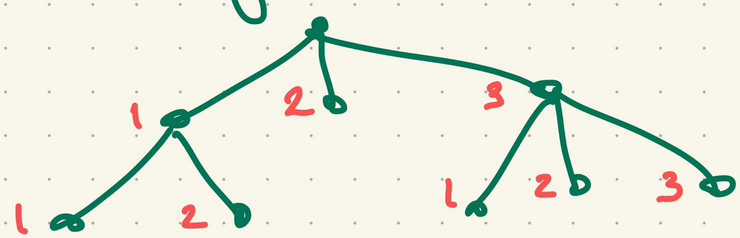
Height of node v = length of longest path from v to a descendent of v (eg. to a leaf)

Height of tree T = height of its root
= max height of any node in T
= max depth of any node in T.

A rooted tree is

- k-ary if no node has $> k$ children
- binary if no node has > 2 children
- ordered if the children of every node are ordered.

Eg: A ordered ternary tree:



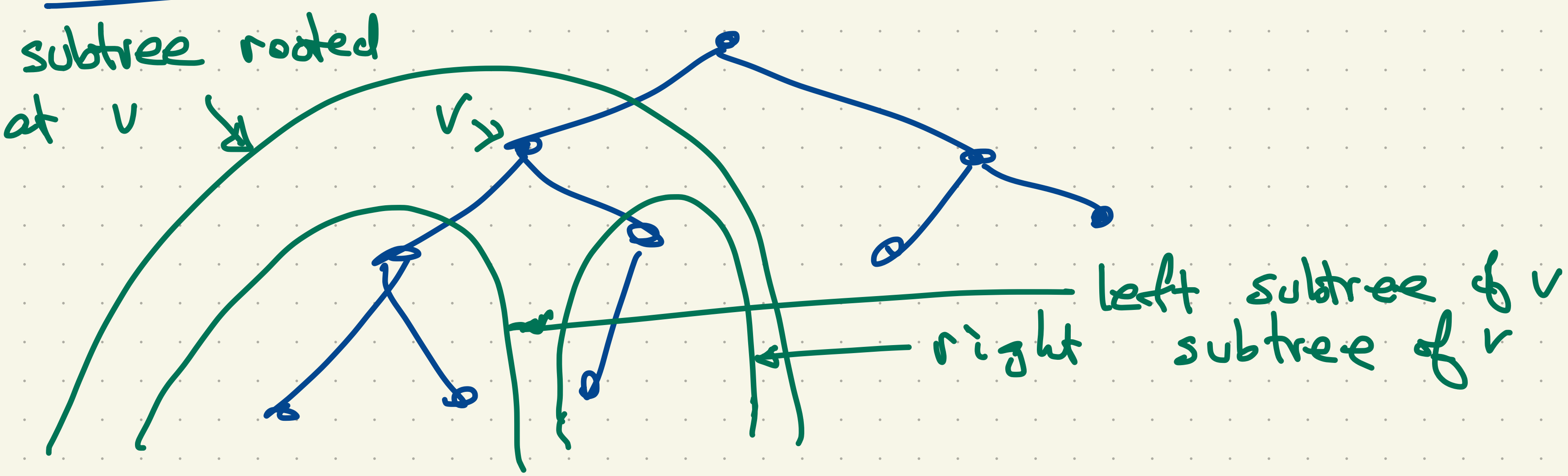
Notice: When we draw a tree, or represent it in a data structure, we order it.

In an ordered binary tree, every child of a node v is either the "left child of v" or the "right child of v".

Subtree rooted at v: tree with root v and containing all descendants of v.

In a binary tree;

- "left subtree of v" means the subtree rooted at the left child if v
- Sim. for "right subtree of v".



End