Chapter 12: Trees

CS1231 Discrete Structures

Wong Tin Lok

National University of Singapore

2022/23 Semester 2

Trees sprout up just about everywhere in computer science [. . .]. Knuth (2006)

Why trees?

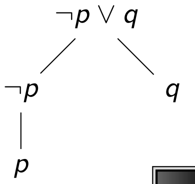
- They are useful in representing all kinds of situations where branching is involved.
- For this module, they provide a topic on which we showcase the proof techniques that we learnt.

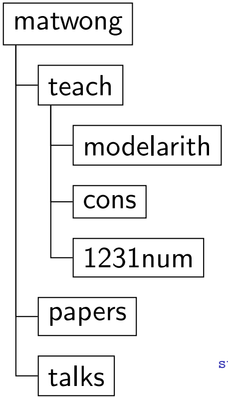
Question

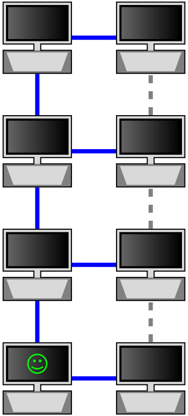
What exactly are trees?

My attempt at an answer

- Every node is grown out from the root.
- This growing may involve branching.
- Different branches stay separate.







https://tex.stackexchange.com/a/360109

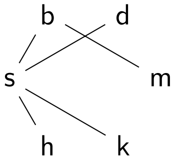
Trees as graphs

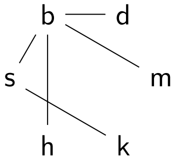
Definition 12.1.1

A *tree* is a connected acyclic undirected graph.

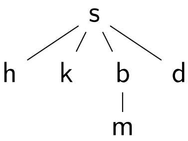
Example 12.1.2

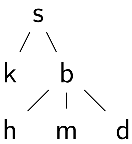
The following are drawings of trees.





One can redraw these respectively as follows.





A unique path between any two vertices

Reformulation of Theorem 11.2.5

An undirected graph with no loop is acyclic if and only if there is at most one path between any two vertices in the graph.

Proposition 12.1.3

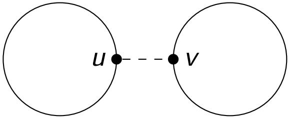
An undirected graph with no loop is a tree if and only if between any two vertices there is exactly one path in the graph. $\square$

Connected graphs with a minimal set of edges ($\Rightarrow$)

Theorem 12.1.4

A connected undirected graph G is a tree if and only if removing any edge disconnects G, i.e., for every $e \in \mathrm{E}(G)$, the graph $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{e\})$ is not connected.

Proof of $\Rightarrow$



Suppose G is a tree.

- Take any $uv \in \mathrm{E}(G)$.
- Being acyclic, we know G has no loop.
- So $u \neq v$, and thus uv is a path between u and v in G.
- In view of Proposition 12.1.3, this is the only path between u and v in G.
- Hence in $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{uv\})$ there is no path between u and v.
- This shows $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{uv\})$ is not connected. $\square$

Connected graphs with a minimal set of edges ($\Leftarrow$)

We prove this by contraposition. Suppose G is not a tree. As G is connected, we deduce that G must be cyclic. If G has a loop, say $uu \in \mathrm{E}(G)$, then $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{uu\})$ remains connected because by definition no path can have a loop. So suppose G has a cycle $x_1x_2 \ldots x_kx_1$. We show that $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{x_1x_k\})$ is connected; this will imply what we want.

Take any $a, b \in \mathrm{V}(G)$. We want a path between a and b in $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{x_1x_k\})$. Use the connectedness of G to find a path $P = y_0y_1 \ldots y_\ell$ in G where $y_0 = a$ and $y_\ell = b$. Either $x_1x_k \notin \mathrm{E}(P)$ or $x_1x_k \in \mathrm{E}(P)$.

Case 1: suppose $x_1x_k \notin \mathrm{E}(P)$. Then P is a path between a and b in $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{x_1x_k\})$.

Case 2: suppose $x_1x_k \in \mathrm{E}(P)$. Say $x_1 = y_r$ and $x_k = y_{r+1}$. In $(\mathrm{V}(G), \mathrm{E}(G) \setminus \{x_1x_k\})$, there is a path between a and b by Lemma 11.1.10 because

- between a and y_r there is a path $y_0y_1 \ldots y_r$;
- between y_r and y_{r+1} there is a path $x_1x_2 \ldots x_k$;
- between y_{r+1} and b there is a path $y_{r+1}y_{r+2} \ldots, y_\ell$.

$\square$

Number of edges in a tree (1/2)

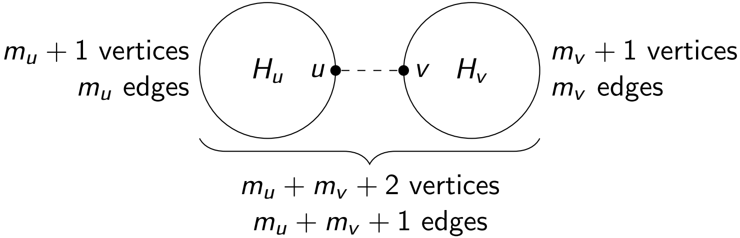
Theorem 12.1.6

Let G be a finite tree with at least one vertex. Then $|\mathsf{E}(G)| = |\mathsf{V}(G)| - 1$.

Sketch of a proof by strong induction on the number of vertices in G

(Base step) Let G be a tree with exactly one vertex. Note that G cannot have a loop because this would make G cyclic. So G cannot have any edge. Thus $|\mathsf{E}(G)| = 0 = 1 - 1 = |\mathsf{V}(G)| - 1$.

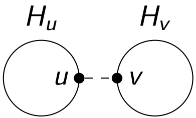
(Induction step) Let $k \in \mathbb{Z}^+$ such that the equation holds for all trees with at most k vertices. Consider a tree G with exactly $k+1$ vertices.



Number of edges in a tree (2/2)

Theorem 12.1.6

Let G be a finite tree with at least one vertex. Then $|\mathrm{E}(G)| = |\mathrm{V}(G)| - 1$.



Sketch of a proof by strong induction on the number of vertices in G

(Base step) ...

(Induction step) Let $k \in \mathbb{Z}^+$ such that the equation holds for all trees with at most k vertices. Consider a tree G with exactly $k+1$ vertices. Take any edge uv in G, where $u \neq v$. Define $G_{uv} = (\mathrm{V}(G), \mathrm{E}(G) \setminus \{uv\})$. Find connected components H_u, H_v of G_{uv} with $u \in \mathrm{V}(H_u)$ and $v \in \mathrm{V}(H_v)$. These connected components are distinct, and they are the only connected components of G_{uv}. By the definition of connected components, we know H_u, H_v are connected. As G is acyclic, both H_u and H_v as subgraphs of G must also be acyclic. Putting these together, we conclude that H_u and H_v are both trees. Hence the induction hypothesis tells us $|\mathrm{E}(H_u)| = |\mathrm{V}(H_u)| - 1$ and $|\mathrm{E}(H_v)| = |\mathrm{V}(H_v)| - 1$. Thus by the Addition Rule,

$$\begin{aligned} |\mathrm{E}(G)| &= |\mathrm{E}(H_u)| + |\{uv\}| + |\mathrm{E}(H_v)| \\ &= \big(|\mathrm{V}(H_u)| - 1\big) + 1 + \big(|\mathrm{V}(H_v)| - 1\big) = |\mathrm{V}(G)| - 1. \end{aligned}$$

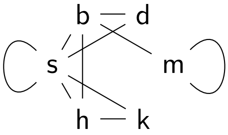
$\square$

Connected cyclic graphs (cycles)

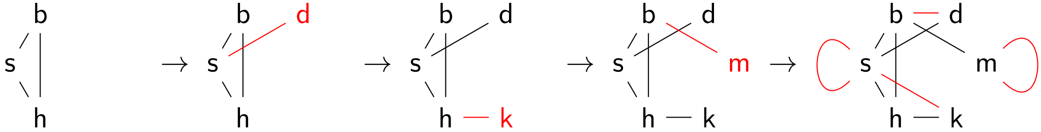
Theorem 12.1.8

Let G be a connected cyclic finite undirected graph. Then $|\mathrm{E}(G)| \geqslant |\mathrm{V}(G)|$.

Example



$\#\text{vertices} = 6$
$\#\text{edges} = 10$



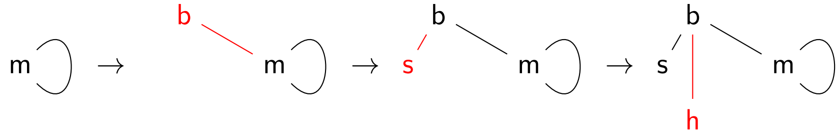
$\#\text{vertices} = 3$	$\#\text{vertices} = 4$	$\#\text{vertices} = 5$	$\#\text{vertices} = 6$	$\#\text{vertices} = 6$
$\#\text{edges} = 3$	$\#\text{edges} = 4$	$\#\text{edges} = 5$	$\#\text{edges} = 6$	$\#\text{edges} \geqslant 6$

Connected cyclic graphs (loops)

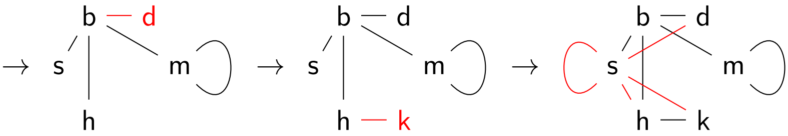
Theorem 12.1.8

Let G be a connected cyclic finite undirected graph. Then $|\mathrm{E}(G)| \geqslant |\mathrm{V}(G)|$.

Example



$\#\text{vertices} = 1$, $\#\text{edges} = 1$ $\quad$ $\#\text{vertices} = 2$, $\#\text{edges} = 2$ $\quad$ $\#\text{vertices} = 3$, $\#\text{edges} = 3$ $\quad$ $\#\text{vertices} = 4$, $\#\text{edges} = 4$



$\#\text{vertices} = 5$, $\#\text{edges} = 5$ $\quad$ $\#\text{vertices} = 6$, $\#\text{edges} = 6$ $\quad$ $\#\text{vertices} = 6$, $\#\text{edges} \geqslant 6$

Connected cyclic graphs (procedure)

Theorem 12.1.8

Let G be a connected cyclic finite undirected graph. Then $|\mathsf{E}(G)| \geqslant |\mathsf{V}(G)|$.

Given a connected cyclic finite undirected graph G, run the following.

1. If G has a loop, then set $H_0 = (\{u\}, \{uu\})$ where $uu \in \mathsf{E}(G)$; else set H_0 to be any cycle in G.
2. Initialize $i = 0$.
3. While $\mathsf{V}(G) \setminus \mathsf{V}(H_i) \neq \varnothing$ do:
   3.1 Use the connectedness of G to find an edge $x_{i+1}y_{i+1}$ in G such that $x_{i+1} \in \mathsf{V}(H_i)$ and $y_{i+1} \notin \mathsf{V}(H_i)$.
   3.2 Set $H_{i+1} = \big(\mathsf{V}(H_i) \cup \{y_{i+1}\}, \mathsf{E}(H_i) \cup \{x_{i+1}y_{i+1}\}\big)$.
   3.3 Increment i to $i + 1$.

$|\mathsf{V}(H_0)| = |\mathsf{E}(H_0)| \geqslant 1.$

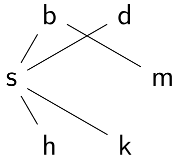
$|\mathsf{E}(H_{i+1})|$
$= |\mathsf{E}(H_i)| + 1$
$= |\mathsf{V}(H_i)| + 1$
$= |\mathsf{V}(H_{i+1})|.$

Analysis

As G is finite, this procedure stops and results in $H_0, H_1, \ldots, H_k$ where $k \in \mathbb{N}$. Notice $\mathsf{V}(H_k) = \mathsf{V}(G)$ as the stopping condition is reached. So, as each H_i is a subgraph of G,

$$|\mathsf{E}(G)| \geqslant |\mathsf{E}(H_k)| = |\mathsf{V}(H_k)| = |\mathsf{V}(G)|.$$

Characterizations of trees



Corollary 12.1.10

The following are equivalent for all finite undirected graphs G with at least one vertex.

(i) G is a tree, i.e., it is connected and acyclic.

(ii) G has no loop, and between any two vertices there is exactly one path in G.

(iii) G is connected, and removing any edge disconnects G.

(iv) G is connected and $|\mathsf{E}(G)| = |\mathsf{V}(G)| - 1$.

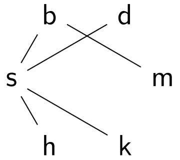
(v) G is acyclic, and adding any new edge makes G cyclic.

(vi) G is acyclic and $|\mathsf{E}(G)| = |\mathsf{V}(G)| - 1$.

Remark

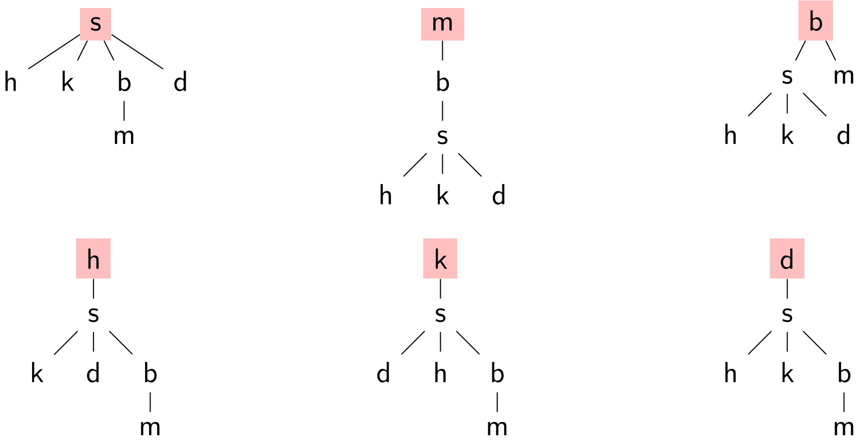
Clauses (v) and (vi) are handled in Tutorial Exercise 12.3 and Tutorial Exercise 12.4.

Any vertex in a tree can be a root



Example 12.2.2

Here are drawings of the tree on the right
with different choices of roots drawn at the top.



Rooted trees

Definition 12.2.1

A *rooted tree* is a tree with a distinguished vertex called the *root*. In a finite rooted tree,

(1) the *height* is the length of a longest path between the root and some vertex;

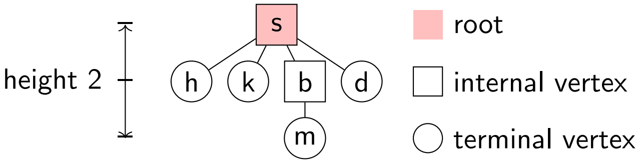
(2) every vertex y except the root has a *parent*, which is defined to be the vertex x such that the edge xy is in the path (unique by Proposition 12.1.3) between y and the root;

(3) a vertex y is said to be a *child* of a vertex x if x is the parent of y;

(4) a *terminal vertex* or a *leaf* is a vertex with no child;

(5) an *internal vertex* or a *parent* is a vertex that is not terminal.

Illustration



(1) s is a parent of h, k, b, d but not of m;

(2) h, k, b, d are children of s, but m is not.

The number of terminal vertices versus the height

Proposition 12.2.3

Let T be a finite rooted tree of height h in which every vertex has at most two children. Then T has at most 2^h terminal vertices.

Proof

As there is a unique path between the root and any terminal vertex in view of Proposition 12.1.3, we may equivalently count the paths

$$x_0 x_1 \dots x_\ell$$

where x_0 is the root and x_ℓ is a terminal vertex. By the definition of heights, such $\ell \leqslant h$. For each $i \in \{1, 2, \dots, h\}$ such that $x_1, x_2, \dots, x_{i-1}$ are chosen,

- if x_{i-1} has no child, then x_{i-1} is already a terminal vertex, and thus there is no x_i;
- if x_{i-1} has a child, then x_{i-1} has one or two children by assumption, and x_i must be one of these children.

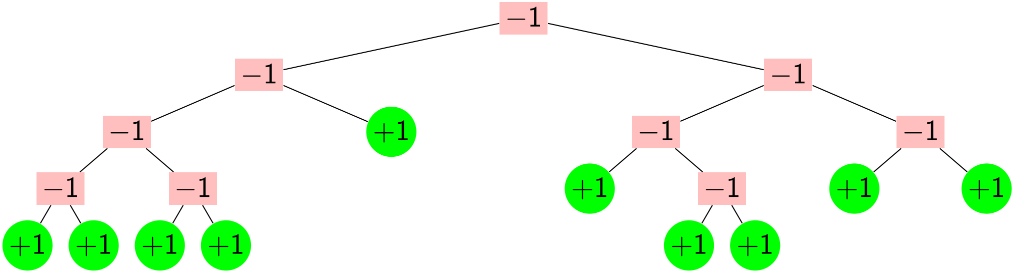
Here, if x_{i-1} has a child, then there must be an x_i, because otherwise $x_\ell = x_{i-1}$, which is not a terminal vertex. Thus, there are at most $\underbrace{2 \times 2 \times \cdots \times 2}_{h\text{-many } 2\text{'s}} = 2^h$ such paths. $\square$

Terminal versus internal vertices (knockout tournament)

Theorem 12.2.5

Let T be a finite rooted tree in which every internal vertex has exactly two children. If T has exactly t terminal vertices, then T has exactly $t-1$ internal vertices.

Proof idea



Terminal versus internal vertices (counting)

Theorem 12.2.5

Let T be a finite rooted tree in which every internal vertex has exactly two children. If T has exactly t terminal vertices, then T has exactly $t-1$ internal vertices.

Proof

Denote by i the number of internal vertices in T. Since every vertex is either internal or terminal but not both, the tree T has exactly $i+t$ vertices. We count the number of (x, y)'s where y is a child of x in T in two different ways.

- The variable x can only take one of i internal vertices, and then the variable y can only take one of the two children the internal vertex chosen. So there are exactly $2i$ ways to substitute objects into (x, y) by the General Multiplication Rule.
- The variable y can only take one of $i+t$ vertices in T, except the root, and then the variable x can only take the parent of this vertex. So there are exactly $i+t-1$ ways to substitute objects into (x, y) by the General Multiplication Rule.

It follows that $2i = i+t-1$. Hence $i = t-1$. $\square$