DISCRETE PROBABILITY

GRAPH THEORY

ALGEBRAIC STRUCTURES

COMBINATORICS


TRIVIAL
NULL
COMPLETE
BIPARTITE
WHEEL
EMPTY
PLANAR
GRAPH THEORY
SPANNING
BINARY
TREES
PATH
CYCLE
K-REGULAR
ISOMORPHISM

trivial
GRAPH

one vertex,
no edges.

trivial GRAPH

null

GRAPH

n vertices,
no edges.

N_n

null

GRAPH

N_5

empty

GRAPH

no vertices,
no edges.

cycle GRAPH

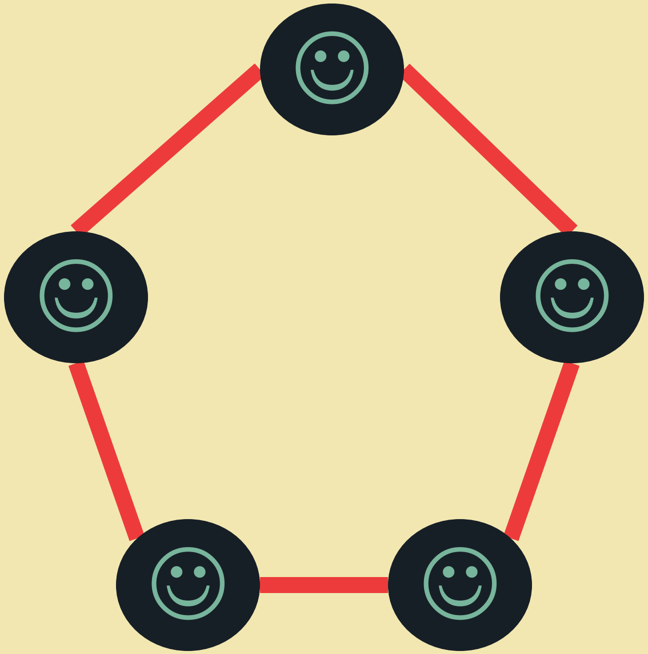
n vertices,

edges form a cycle of length n.

C_n

cycle

GRAPH



C_5

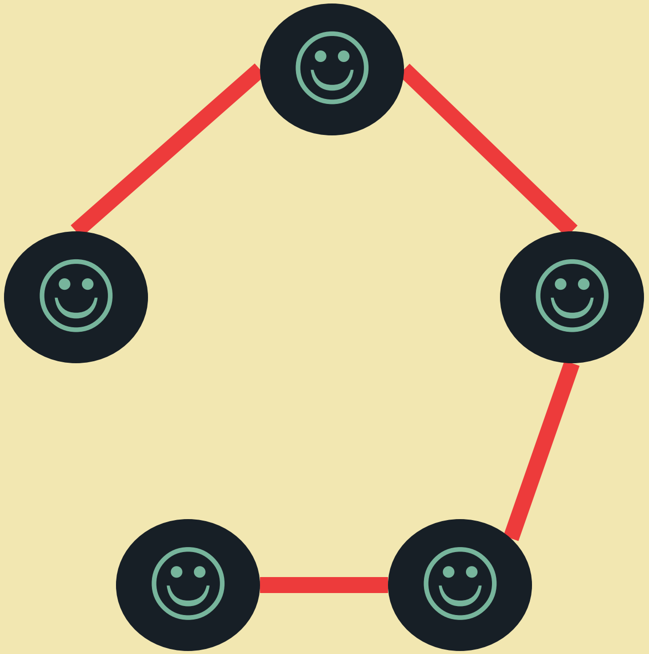
path GRAPH

n vertices,
remove one edge
from a cycle graph C_n.

P_n

path

GRAPH



P_5

wheel

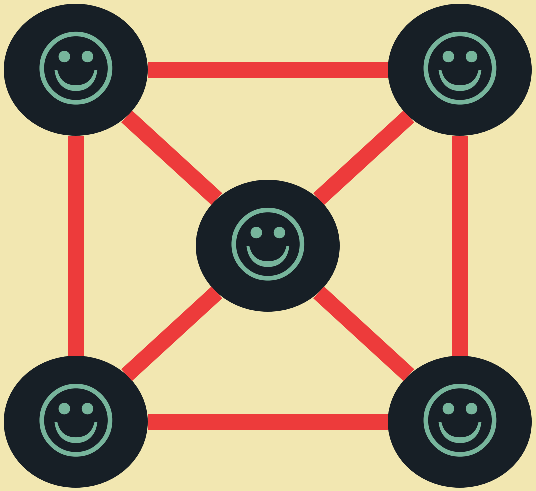
GRAPH

n vertices,

add a new vertex (hub) to a C_{n-1} and join this vertex to all n-1 vertices in C_{n-1}.

W_n

wheel
GRAPH



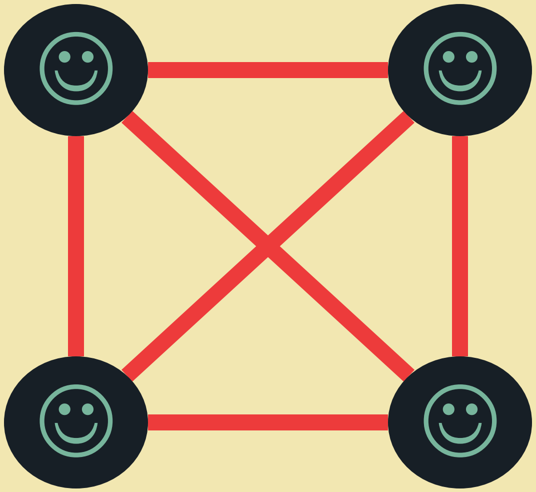
W_5

k-regular GRAPH

simple graph,
every vertex has a degree of k.

W_n

k-regular GRAPH



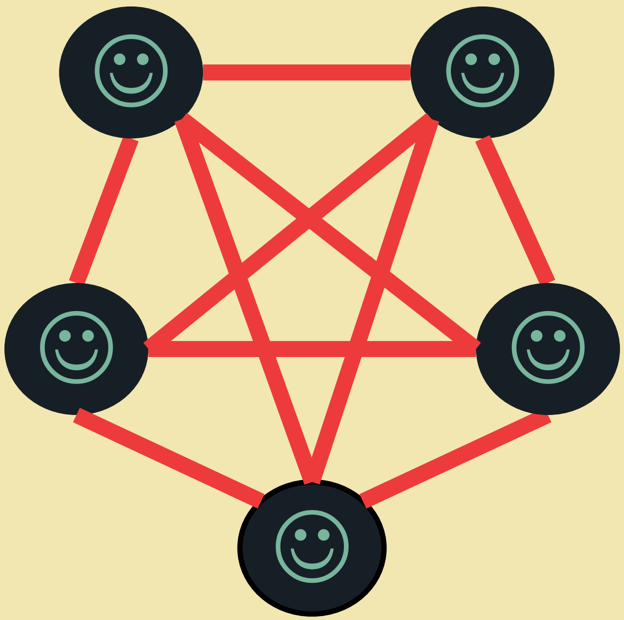
a 3-regular graph

complete
GRAPH

simple graph,
n vertices,
every vertex is adjacent to every other vertex in the graph.

K_n

complete GRAPH



K_5

hypercube

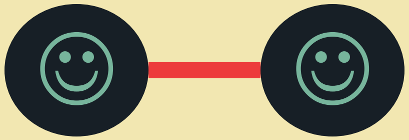
GRAPH

simple graph,

2^n vertices, $2^{n-1}n$ edges, n edges touching each vertex.

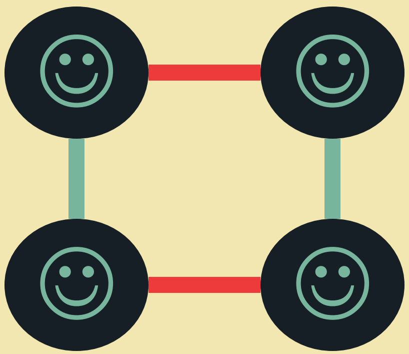
Q_n

hypercube
GRAPH



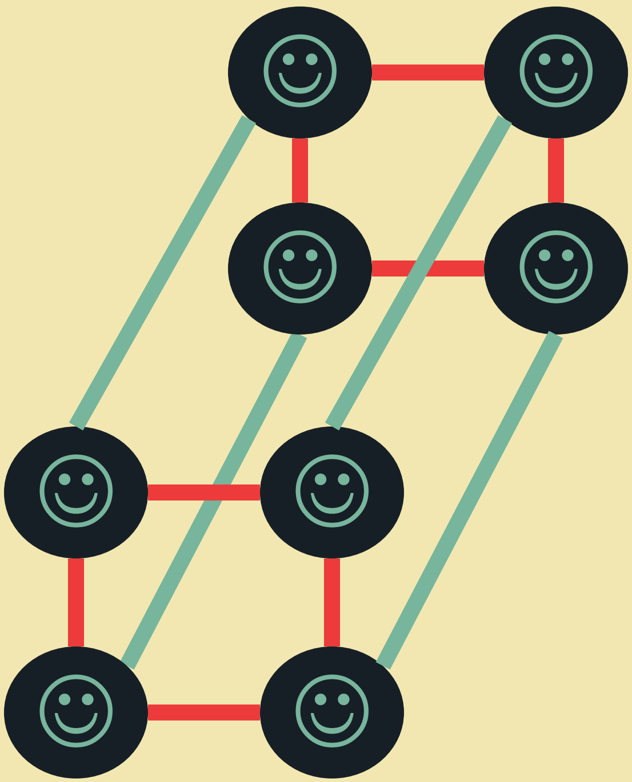
Q_1

hypercube GRAPH



Q_2

hypercube
GRAPH



Q_3

bipartite GRAPH

the vertices can be **partitioned into two sets** so that every edge in the graph only joins one vertex in one set to a vertex in the other set.

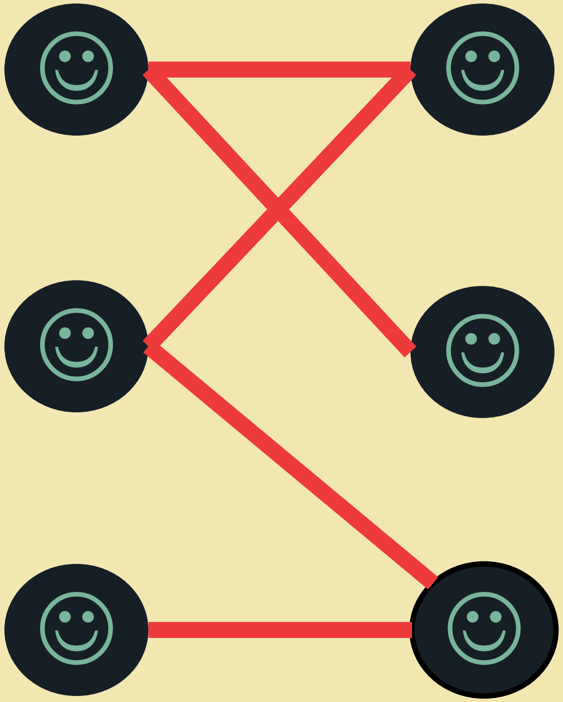
complete bipartite GRAPH

bipartite graph where every vertex in one set is adjacent to every vertex in the other set.

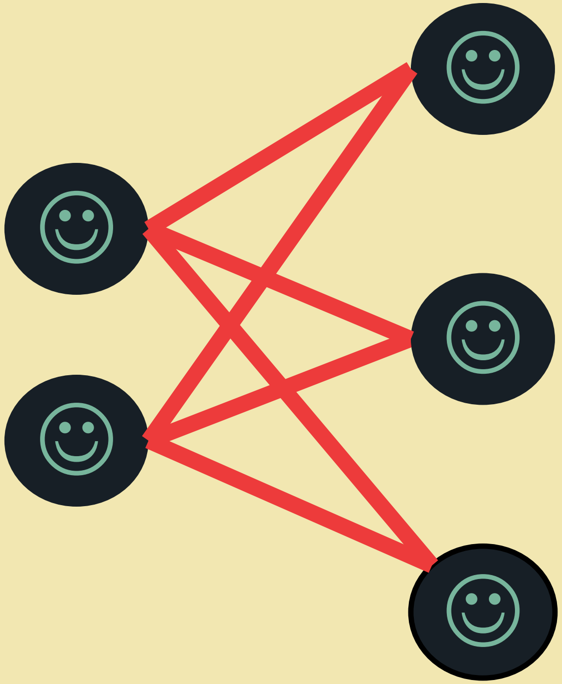
$K_{m,n}$

bipartite

GRAPH



complete
bipartite
GRAPH



$K_{2,3}$

complete bipartite
GRAPH

bipartite graph
where every vertex in one set is adjacent to every vertex in the other set.

$K_{m,n}$

planar GRAPH

graph which can be drawn on a plane such that its edges do not cross each other.

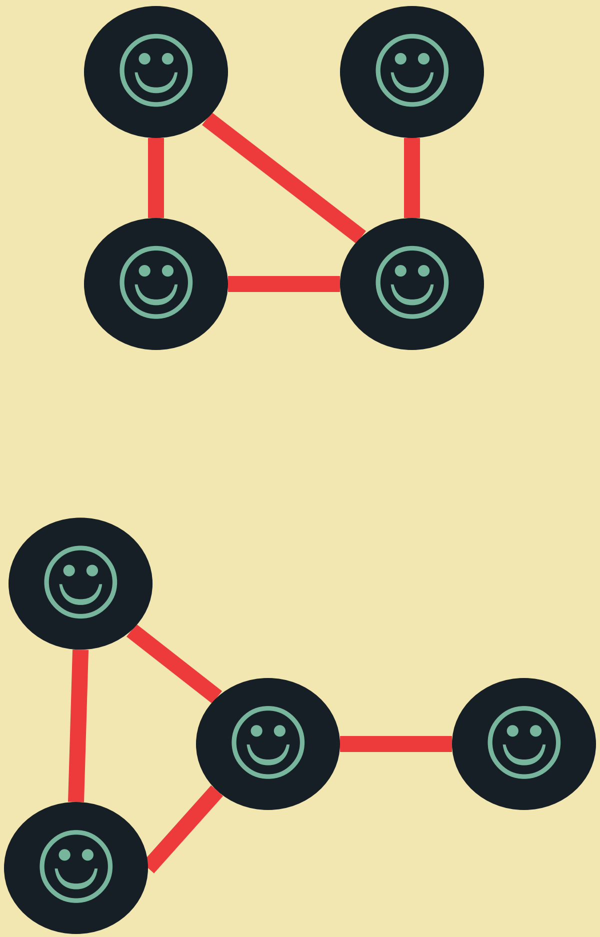
GRAPH G **ISOMORPHIC** to GRAPH H

if there exists a one-to-one correspondence ϕ from $V(G)$ to $V(H)$ such that ϕ preserves adjacency.

GRAPH G **ISOMORPHIC** to GRAPH H

$(u,v) \in E(G)$
if and only if
$(\phi(u), \phi(v)) \in E(H)$.

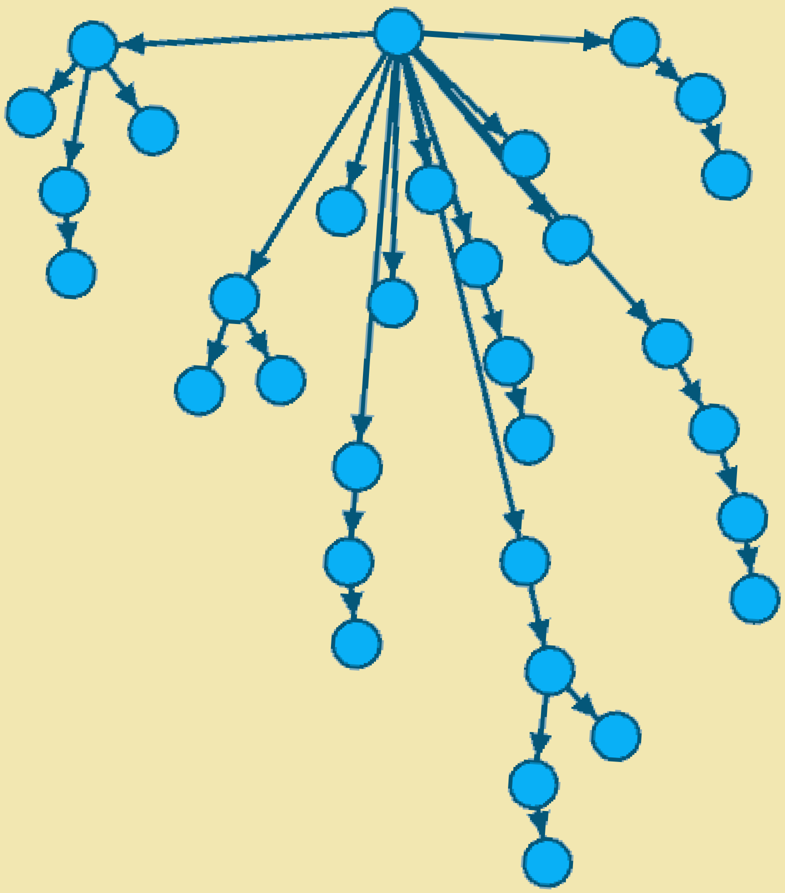
GRAPH G **ISOMORPHIC** to GRAPH H



TREE

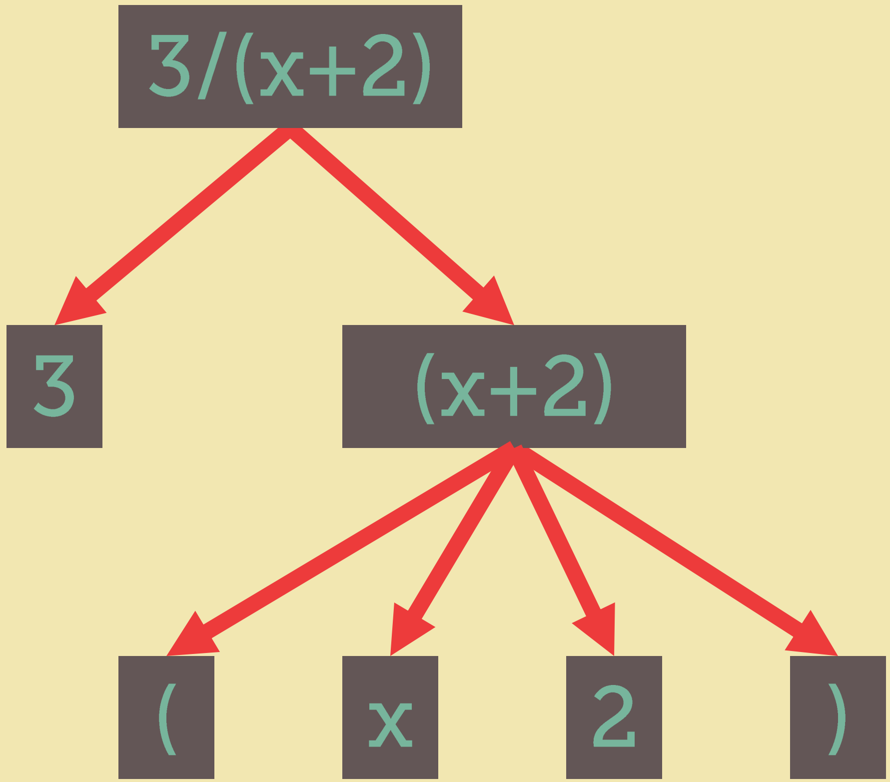
acyclic connected graph.

TREE

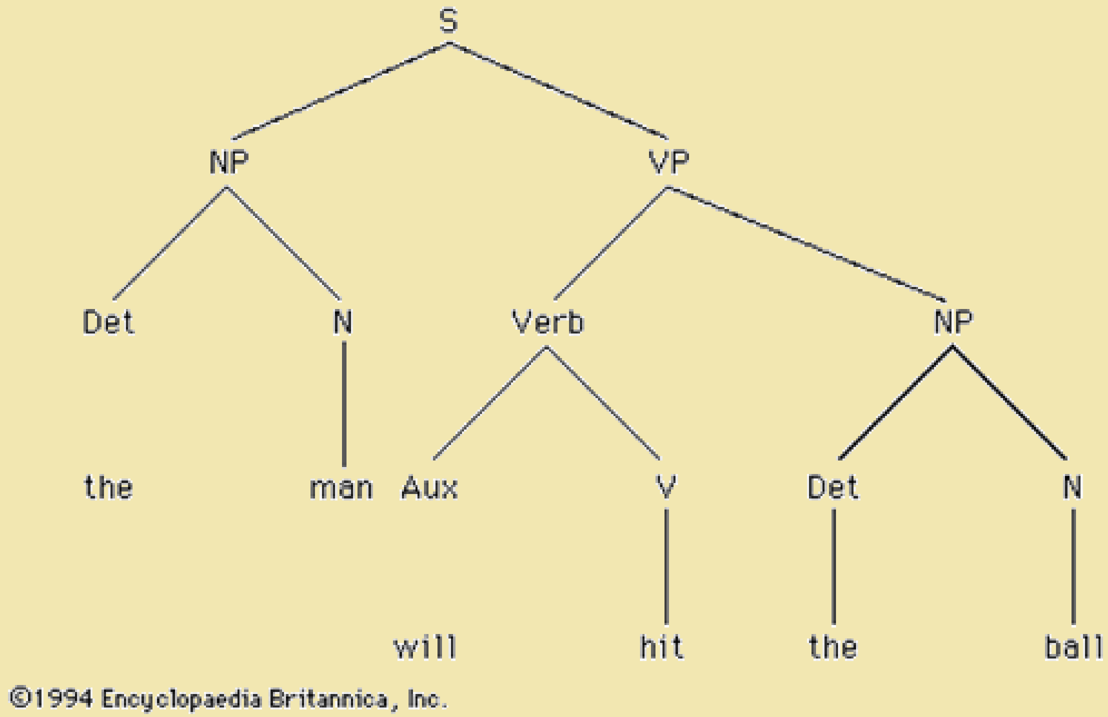


http://darcynorman.net/2012/04/03/discussion-visualization-with-gephi/

TREE

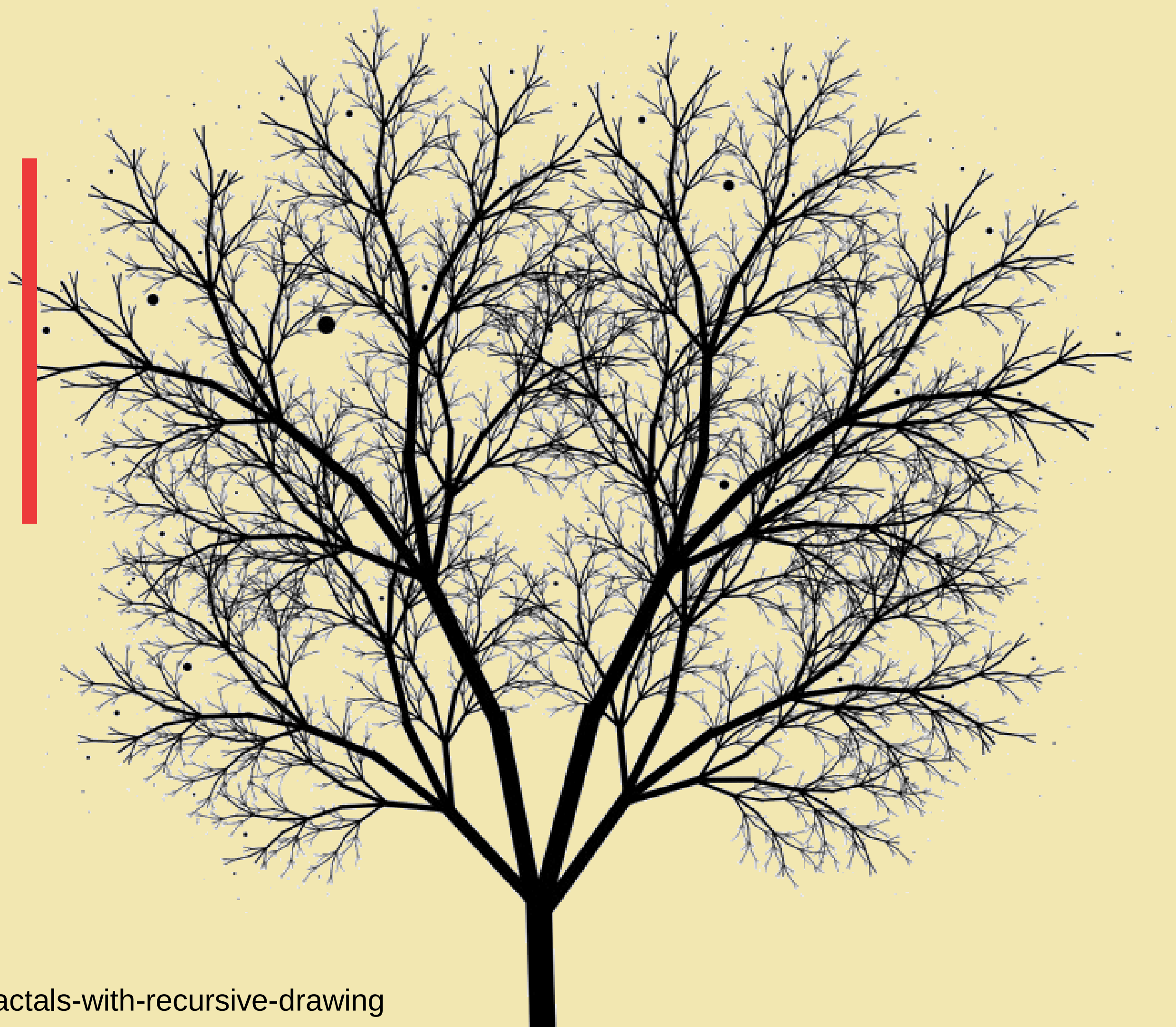


TREE



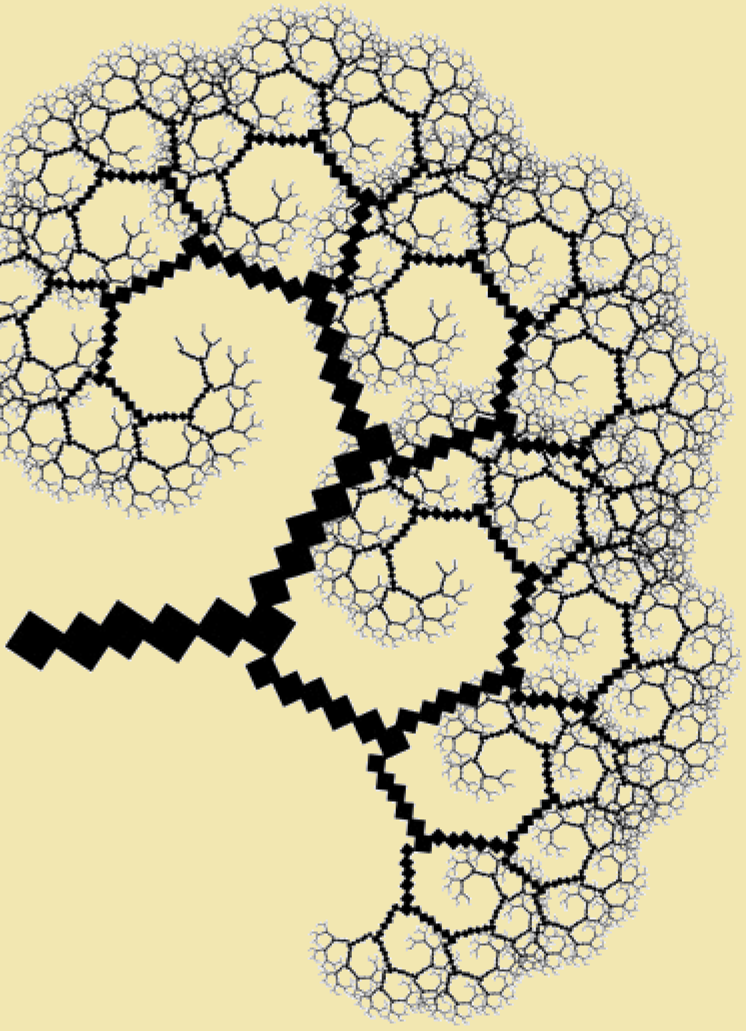
S
NP
VP
Det
N
Verb
NP
the
man
Aux
V
Det
N
will
hit
the
ball
©1994 Encyclopaedia Britannica, Inc.

TREE

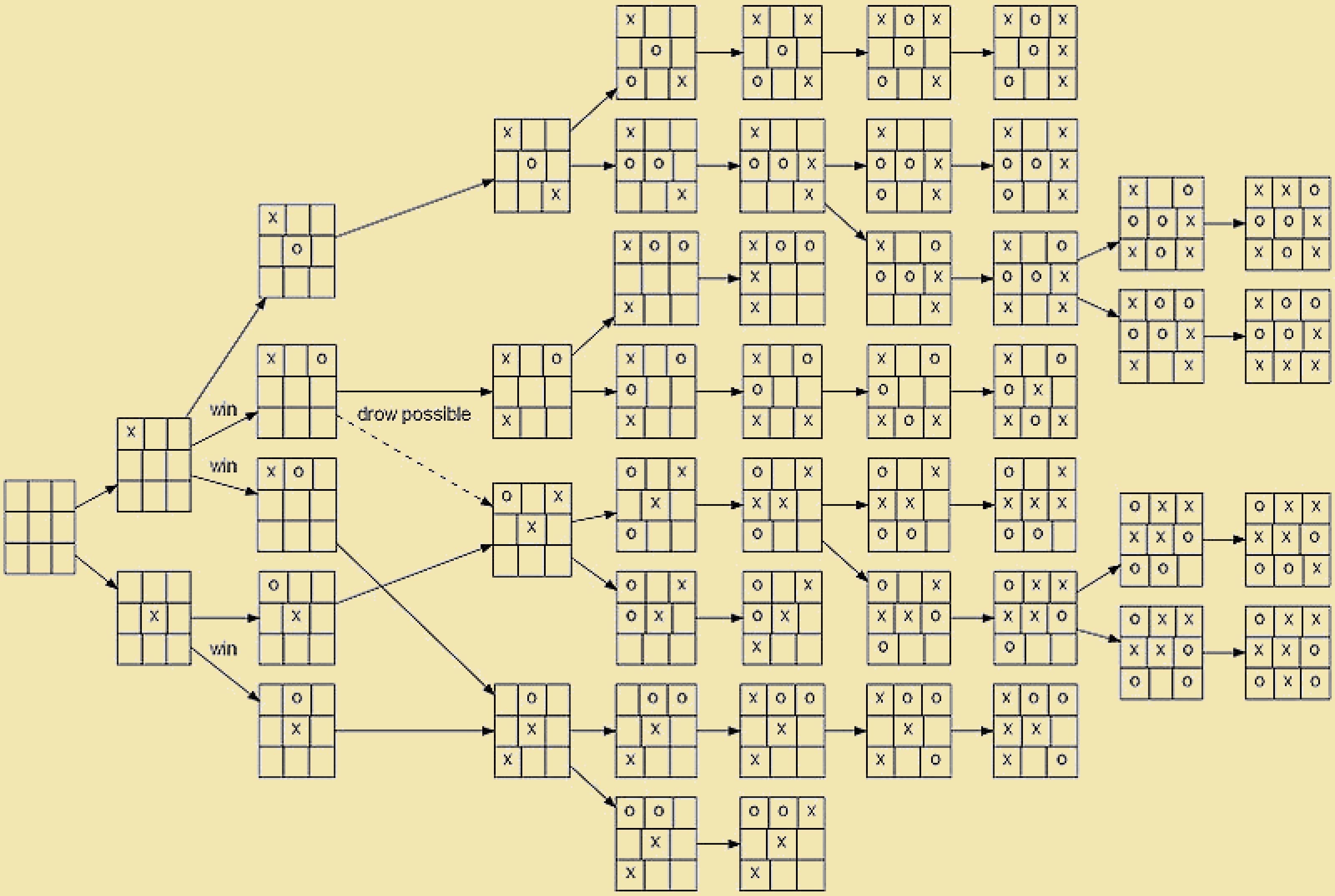


http://matthewjamestaylor.com/blog/create-fractals-with-recursive-drawing

TREE

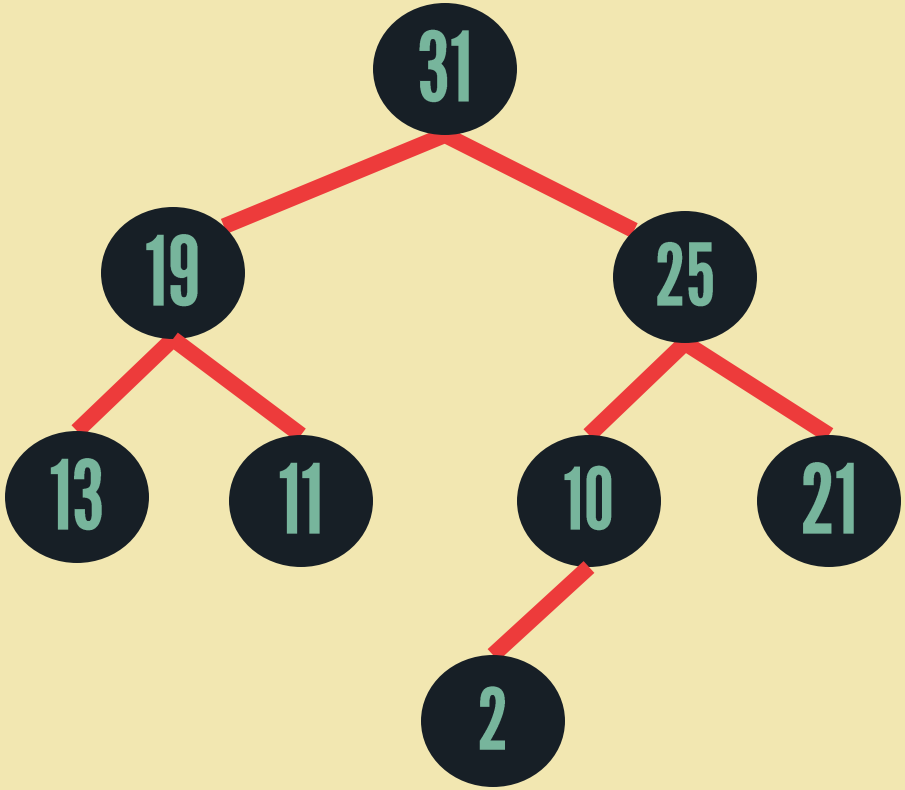


http://matthewjamestaylor.com/blog/create-fractals-with-recursive-drawing



by Agesscene

TREE

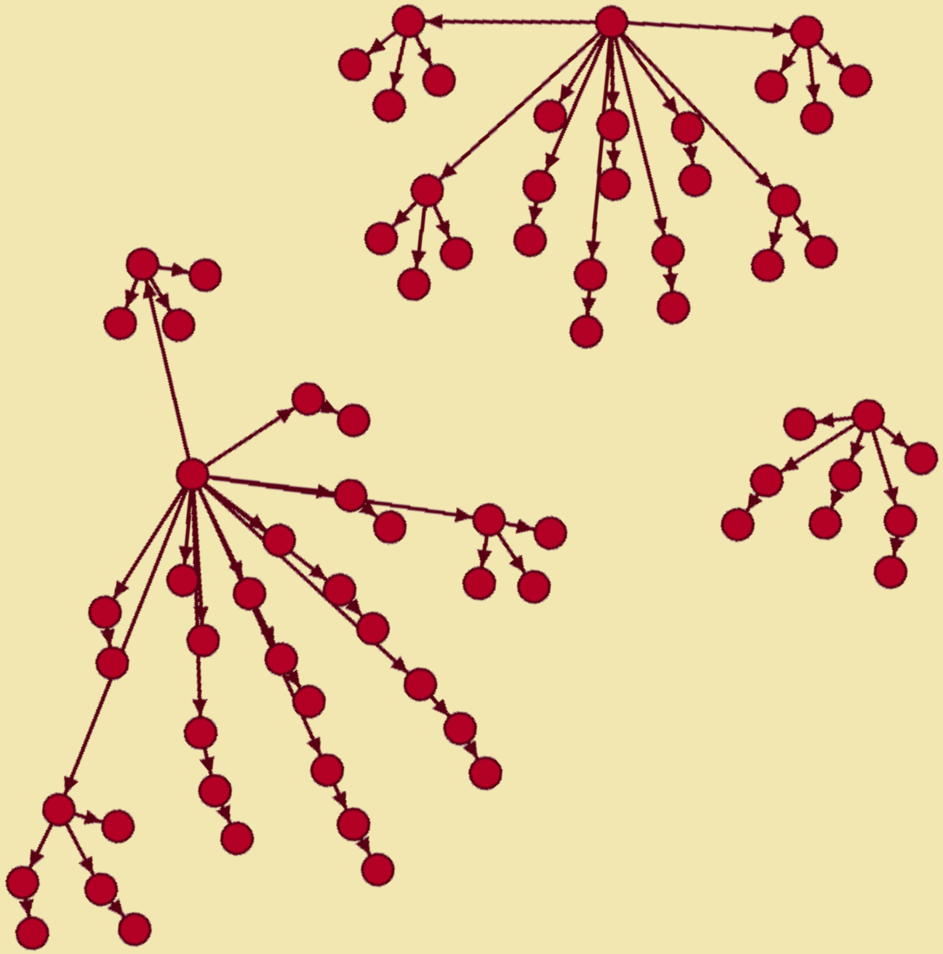


http://matthewjamestaylor.com/blog/create-fractals-with-recursive-drawing

FOREST

A graph whose components are trees.

FOREST

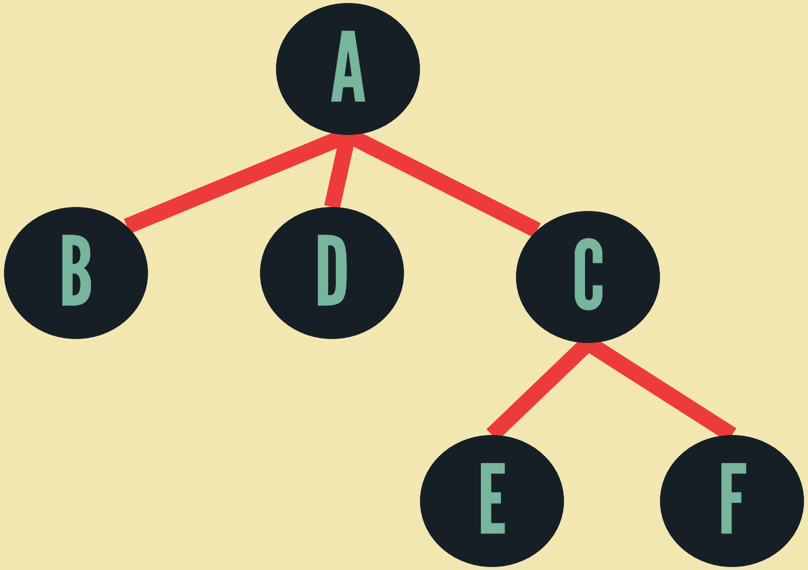


http://darcynorman.net/2012/04/03/discussion-visualization-with-gephi/

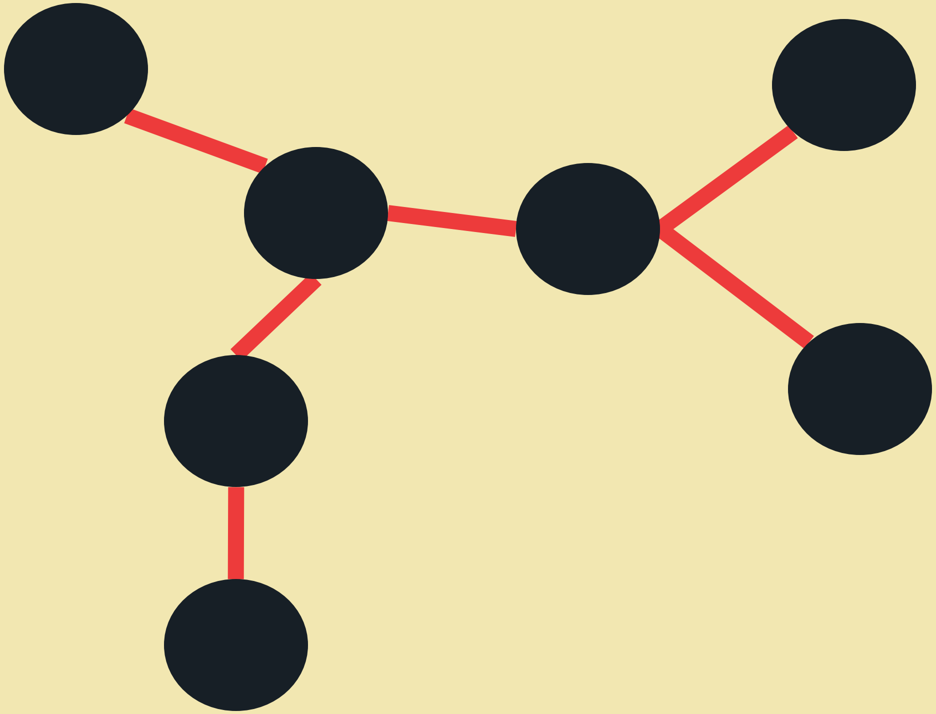
Theorems

GRAPH THEORY

In a tree, any two vertices are connected by a unique path.

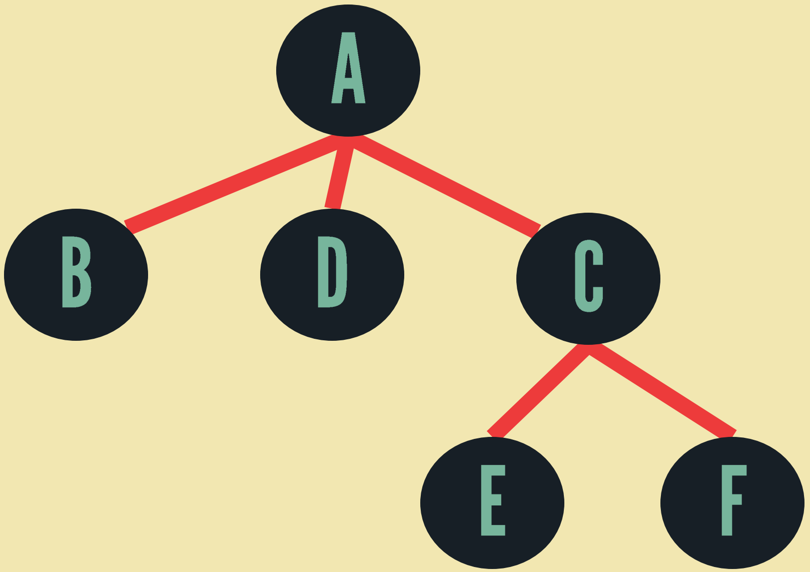


If graph G is a tree, then $|E(G)| = |V(G)| - 1$

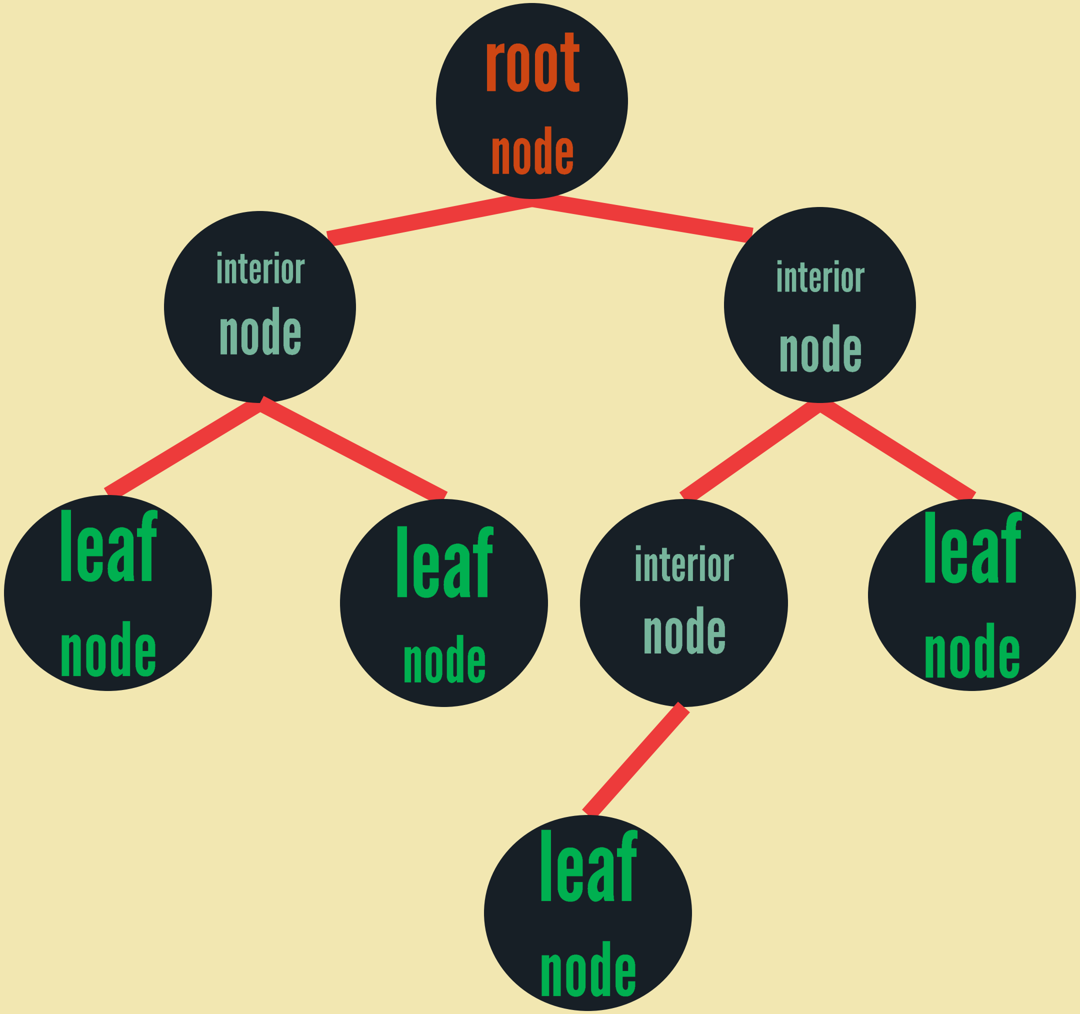


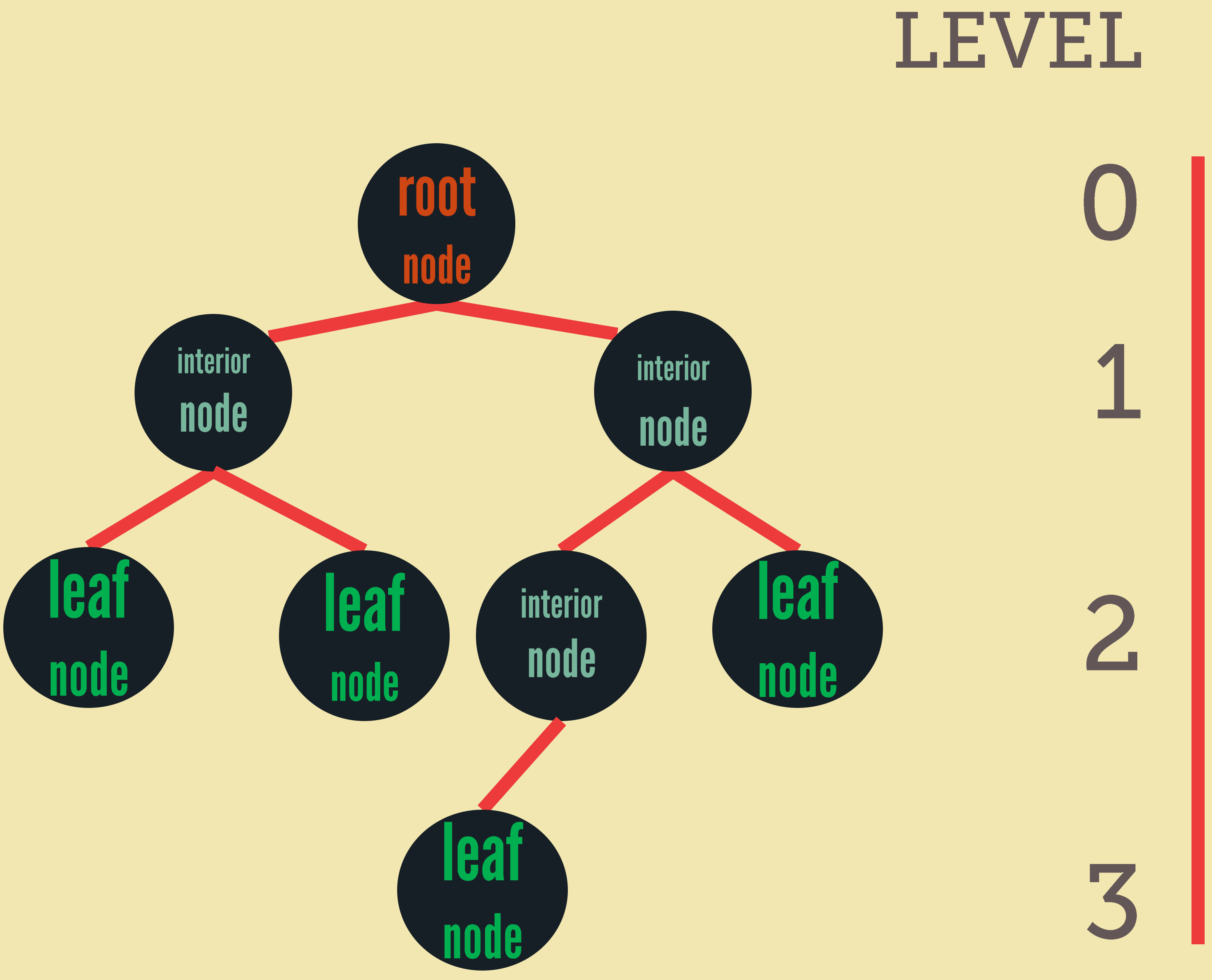
Every edge in a tree is a cut edge

(removing this edge will make the graph disconnected).



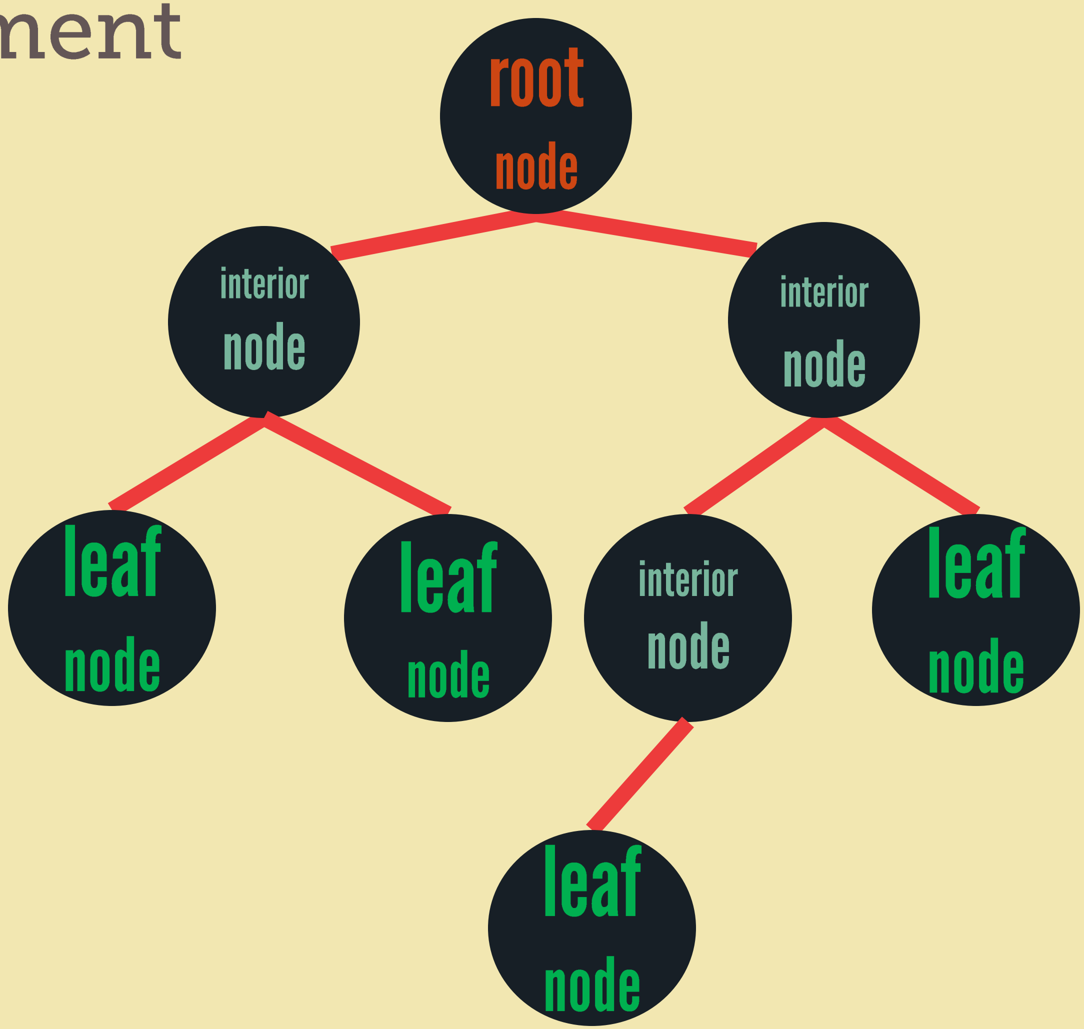
ROOT INTERIOR NODE LEAF NODE



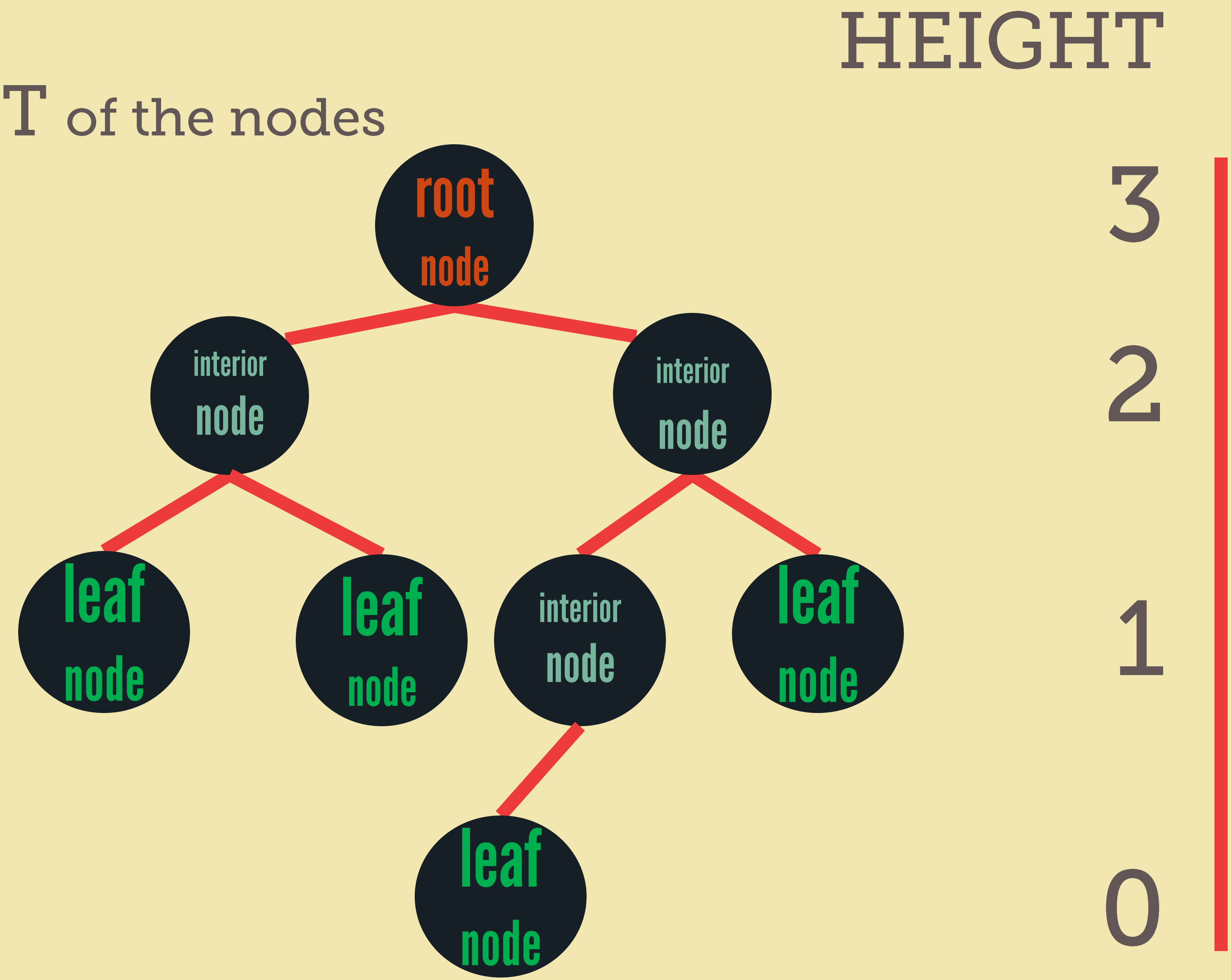


LEVEL
0
1
2
3
root node
interior node
interior node
leaf node
leaf node
interior node
leaf node
leaf node

HEIGHT of the tree = greatest level assignment

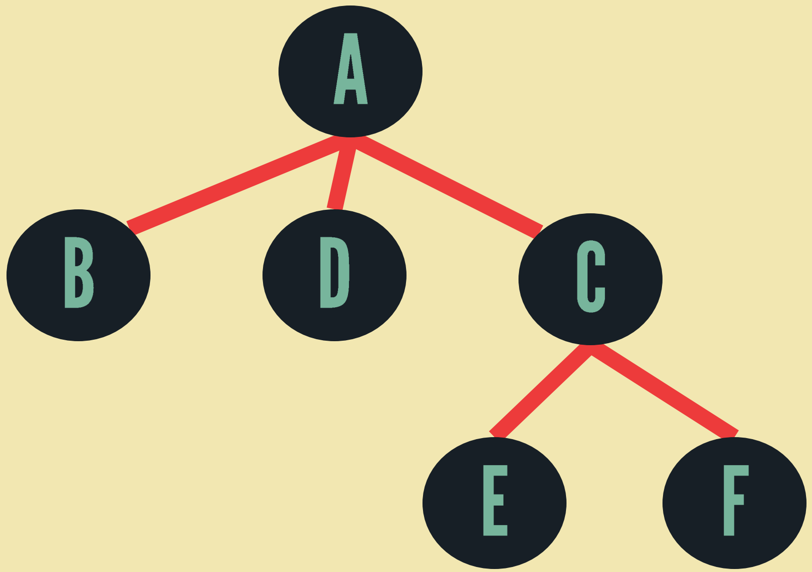


HEIGHT of the nodes

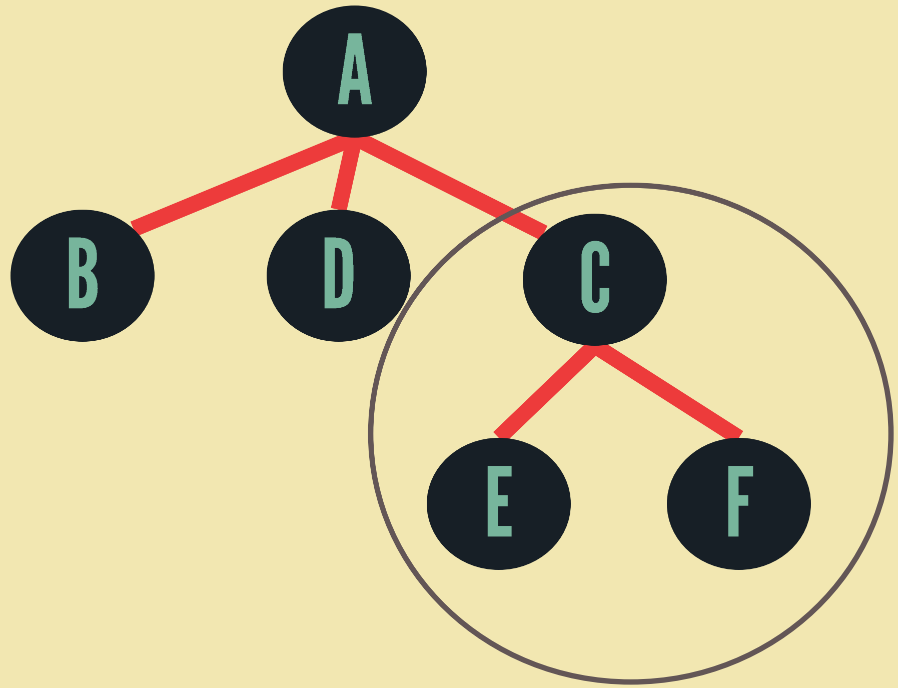


SIBLINGS PARENT CHILD

ANCESTORS DESCENDANTS

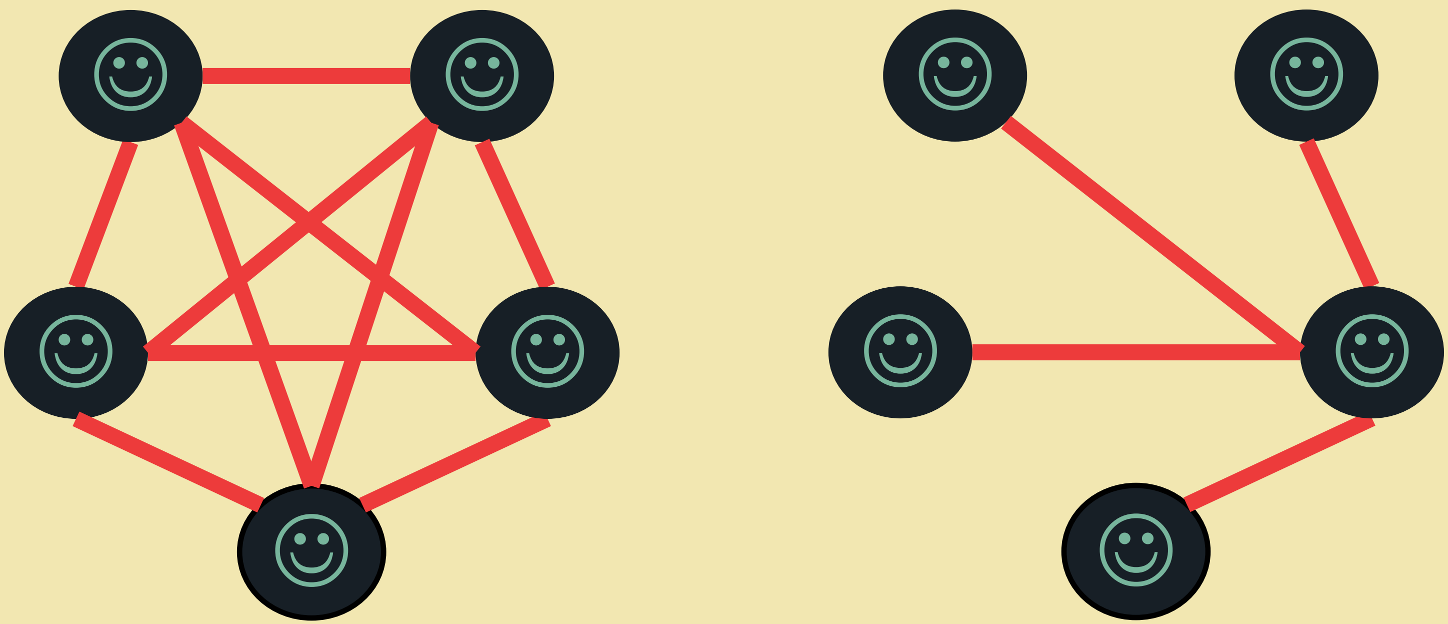


SUBTREE rooted at C



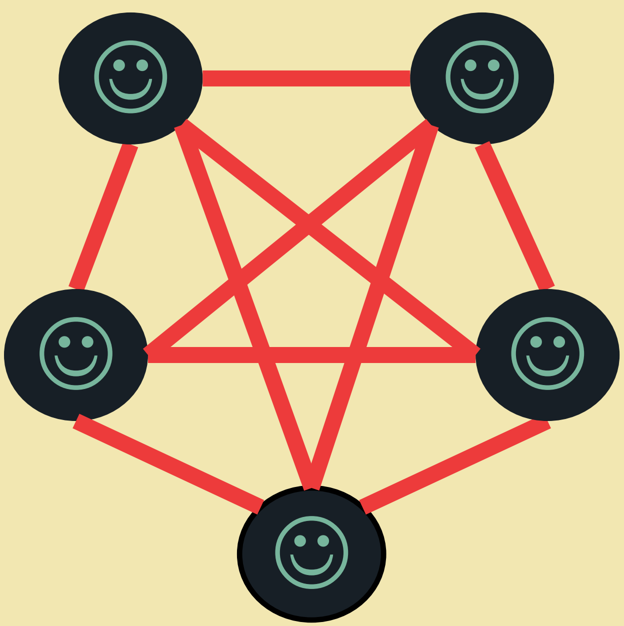
Spanning Tree of a graph G

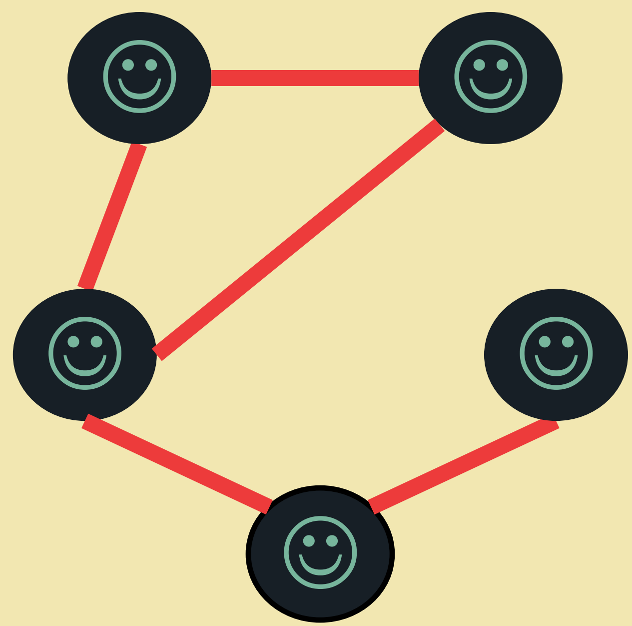
A spanning subgraph T of G such that T is a tree.



Spanning Tree of a graph G

A spanning subgraph T of G such that T is a tree.

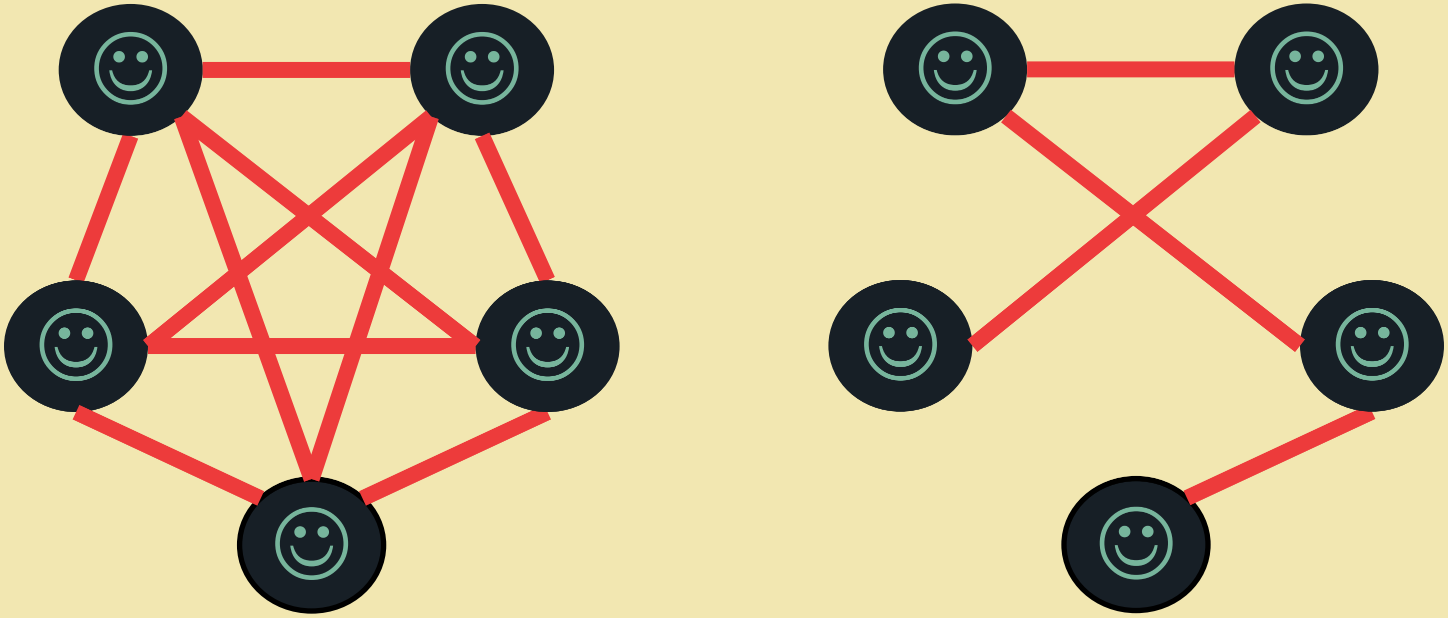




(not a tree)

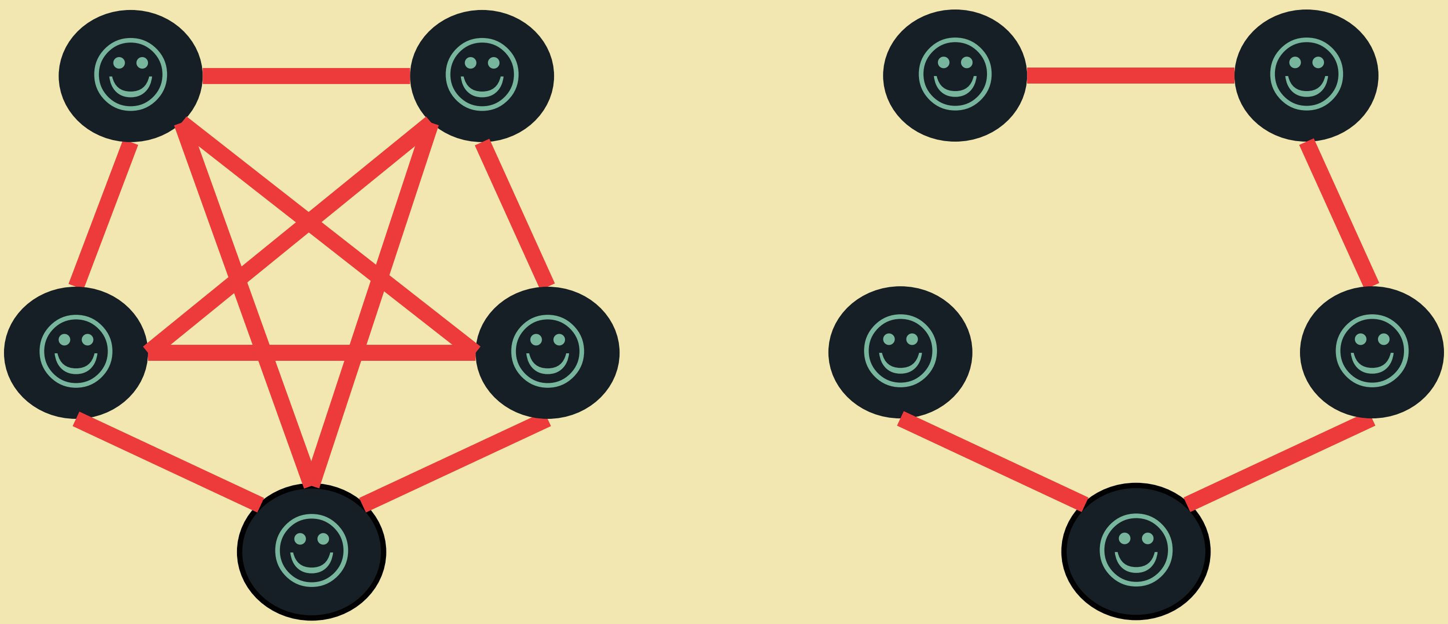
Spanning Tree of a graph G

A spanning subgraph T of G such that T is a tree.



Spanning Tree of a graph G

A spanning subgraph T of G such that T is a tree.

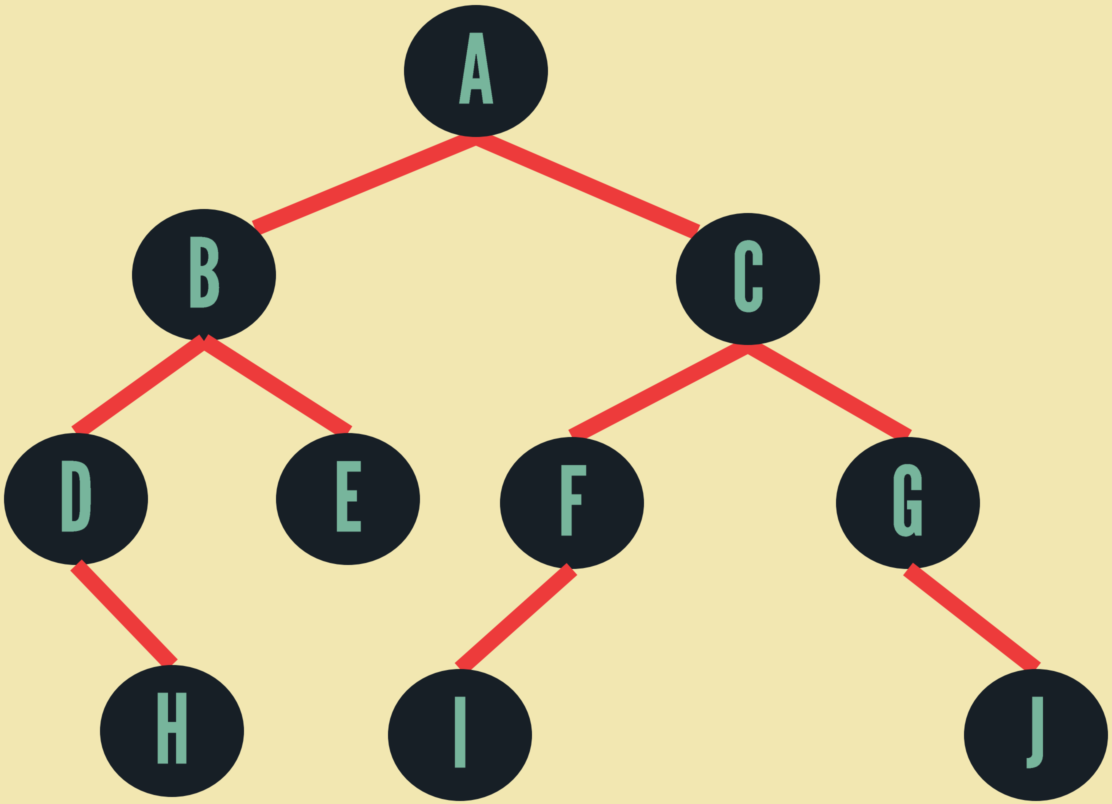


Binary Trees

A tree where each node has either

- no children
- a left child
- a right child
- both left and right child

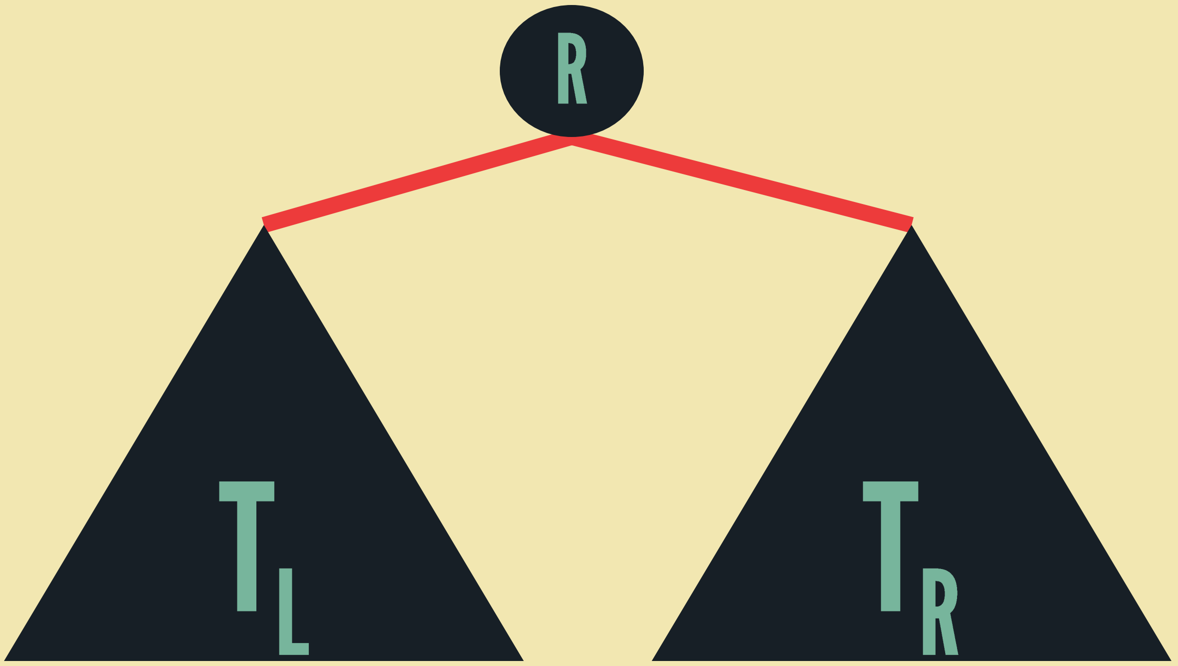
Binary Trees



Binary Tree Traversal

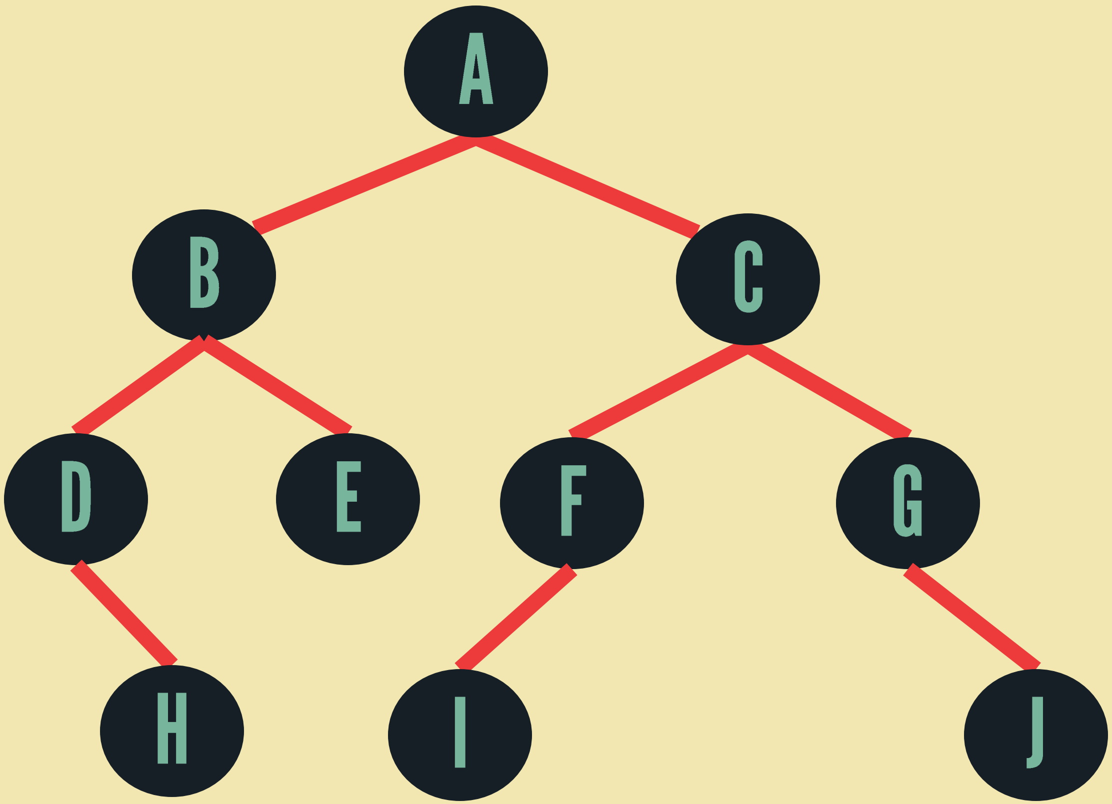
Systematic way of listing down the nodes of a binary tree.

Binary Tree Traversal



Preorder Traversal

root + preorder of T_L + preorder of T_R


A
B
C
D
E
F
G
H
I
J

Postorder Traversal

postorder of T_L + postorder of T_R + root

Inorder Traversal

preorder of T_L + root + preorder of T_R

Expression Trees

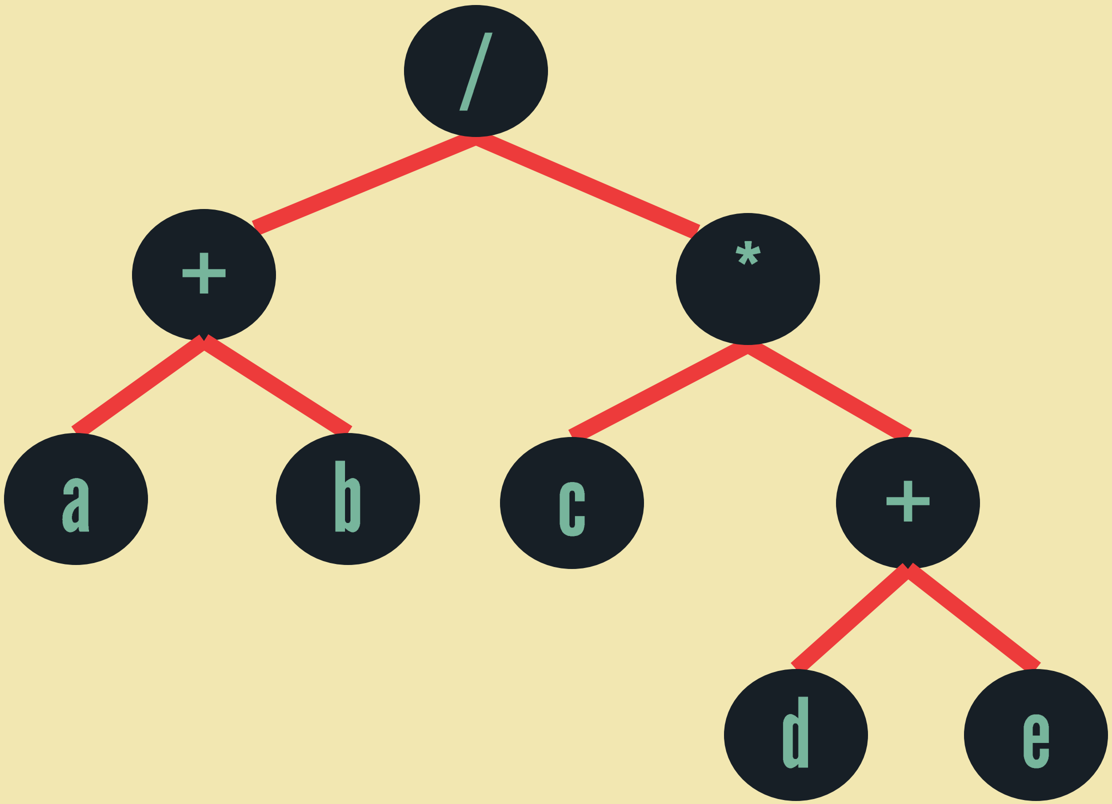
Binary trees where

- each interior node contains an operator
- each leaf contains an operand

Expression Trees

a + b
a*(b - c)

Expression Trees



Expression Trees

Different forms of the expressions

Infix form	inorder traversal
Prefix form	preorder traversal
Postfix form	postorder traversal