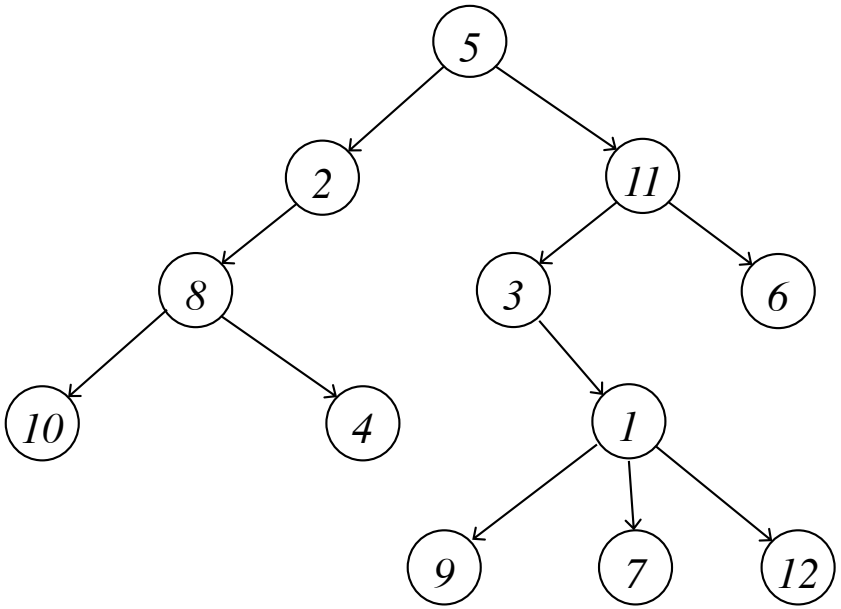
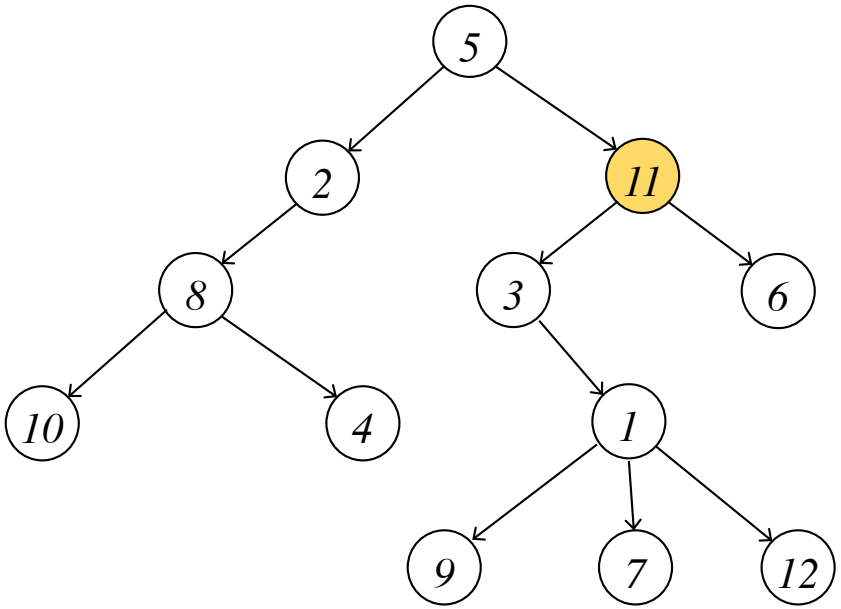
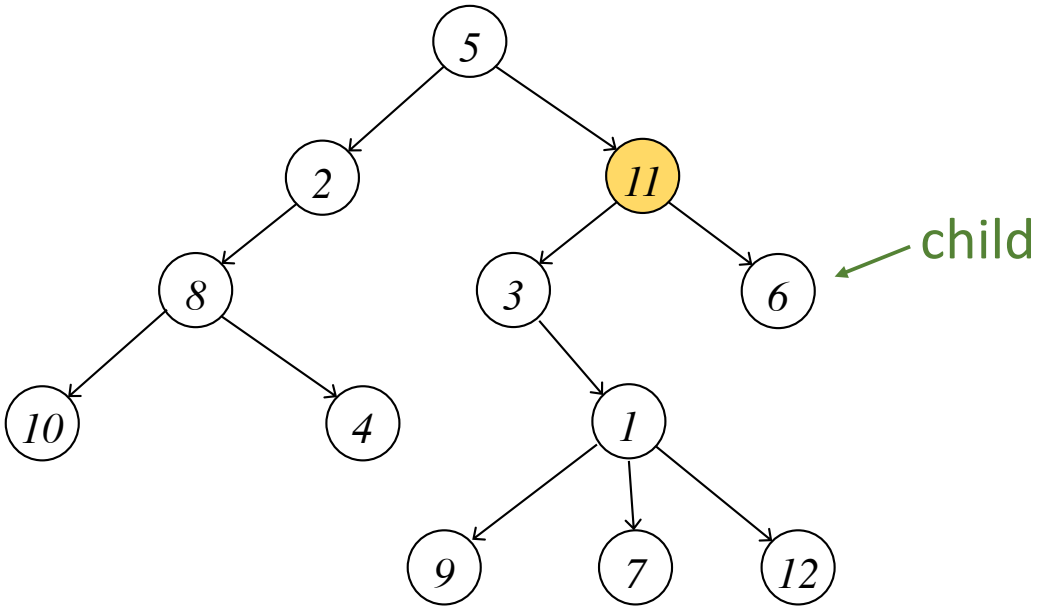
Rooted Trees


5
2
11
8
3
6
10
4
1
9
7
12

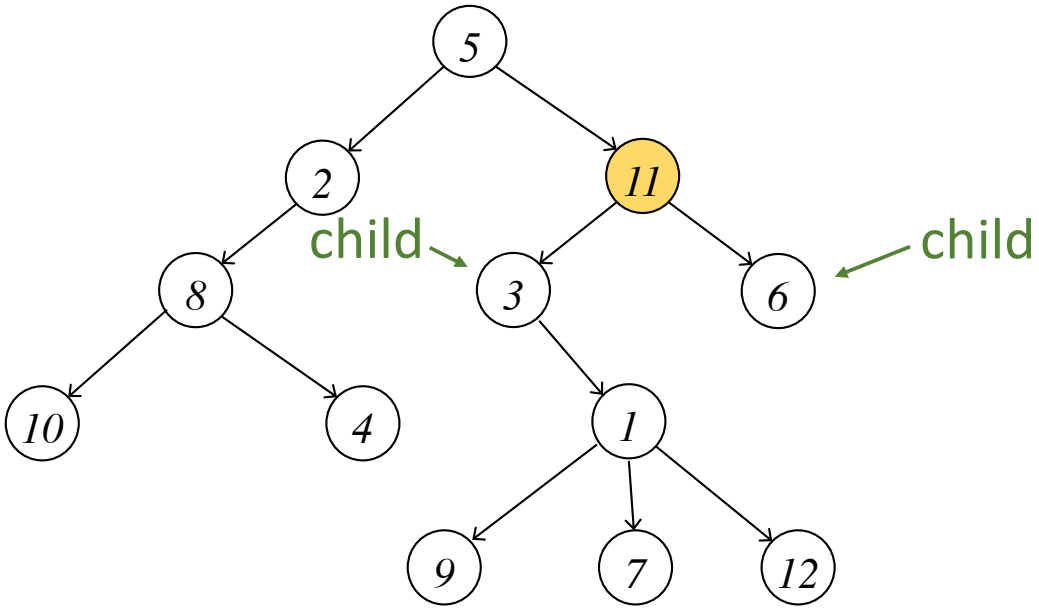
parent-child Relationship



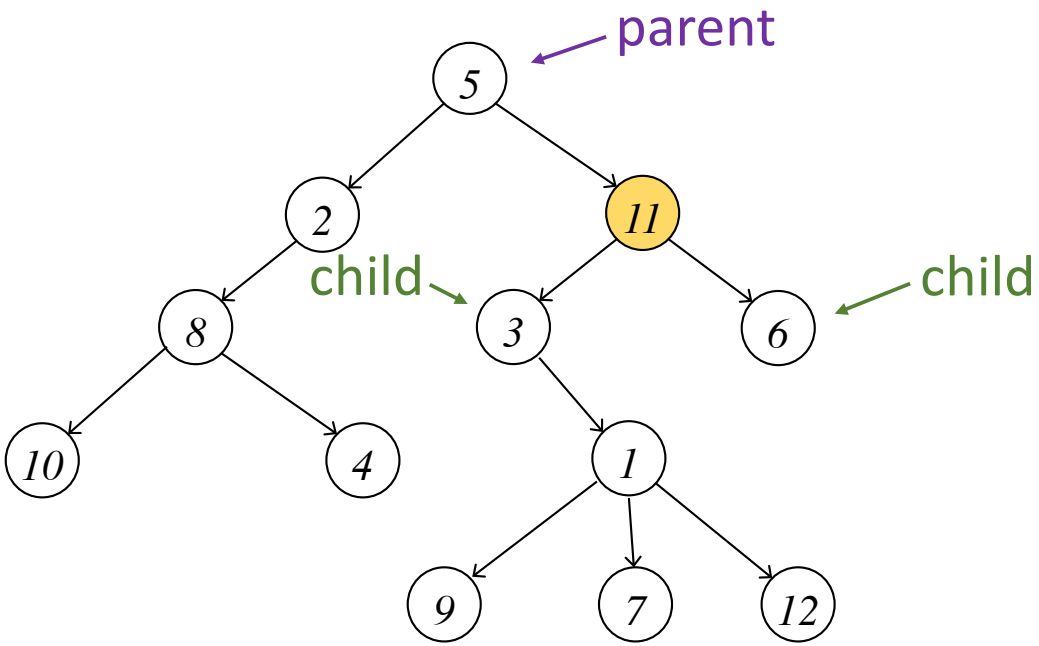
parent-child Relationship



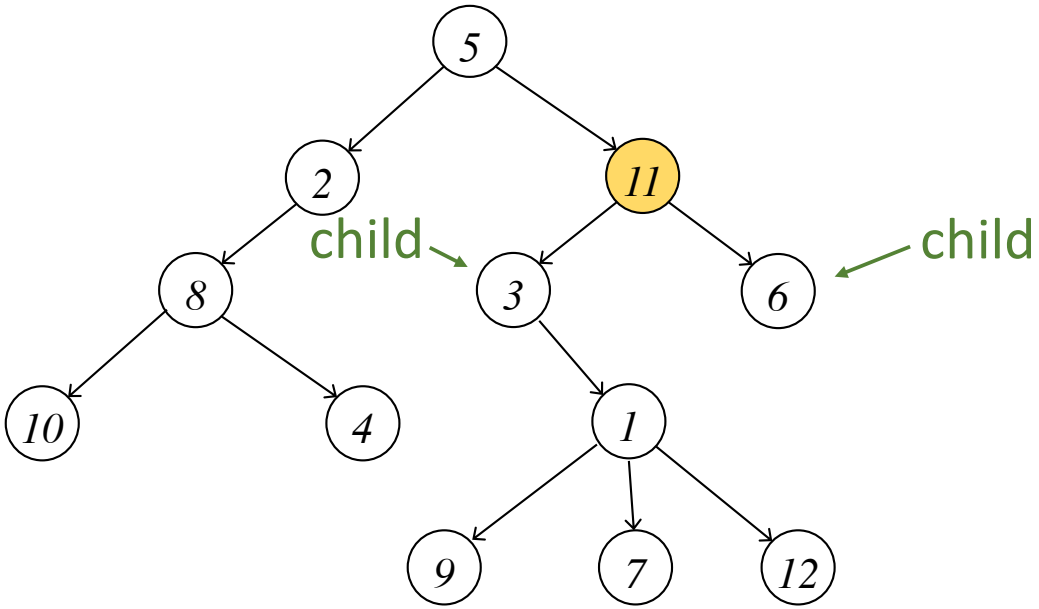
parent-child Relationship



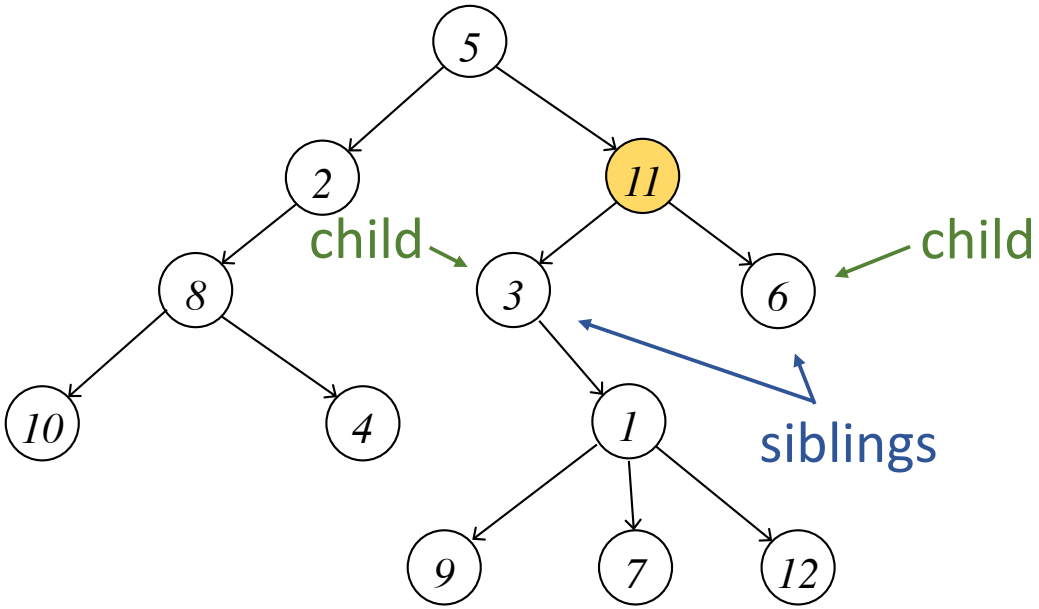
parent-child Relationship



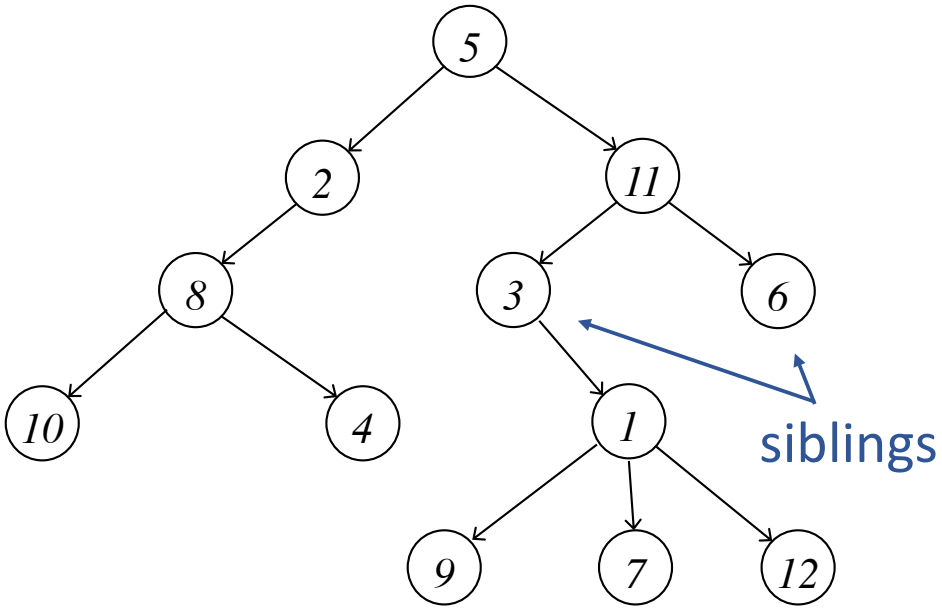
sibling Relationship



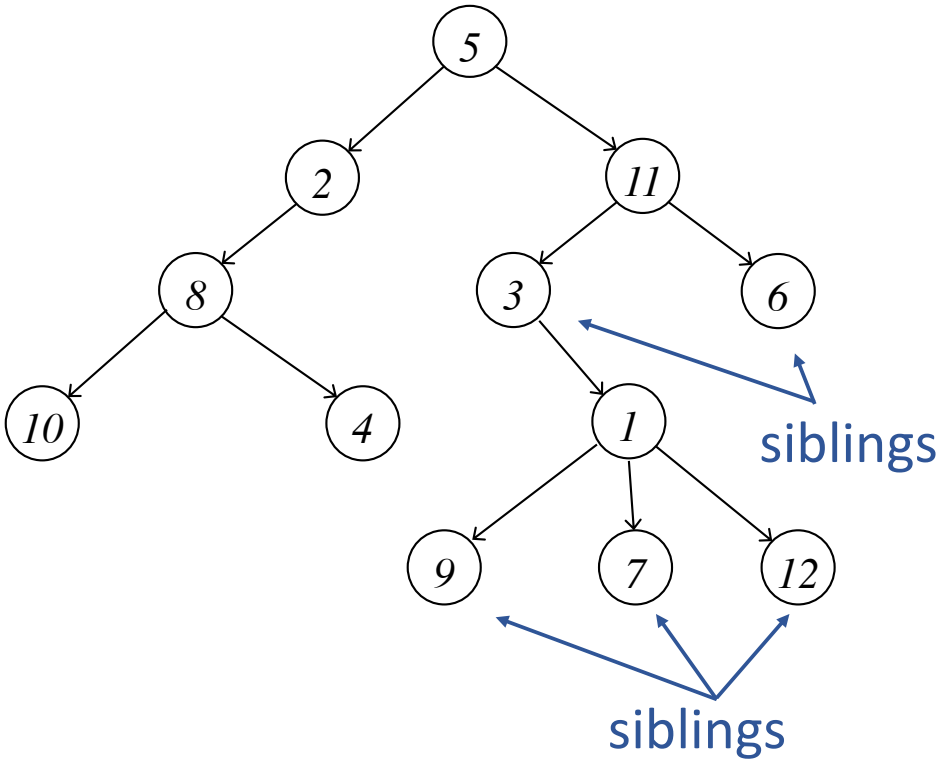
sibling Relationship



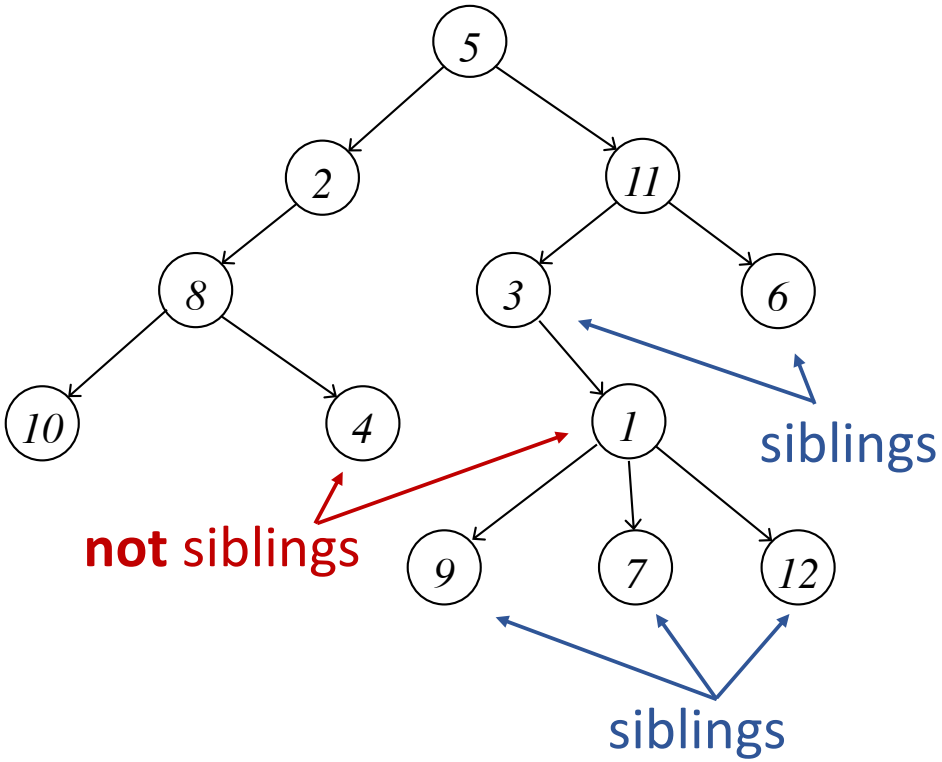
sibling Relationship



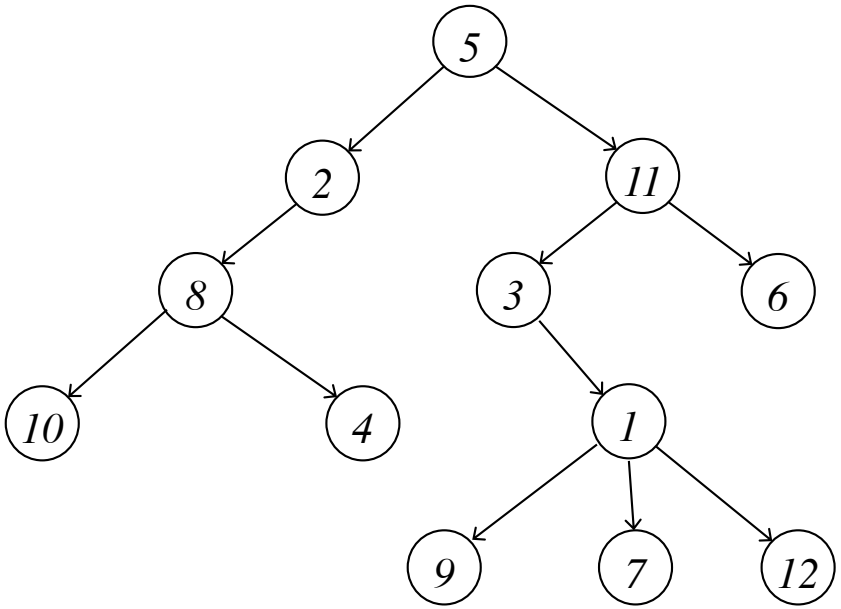
sibling Relationship



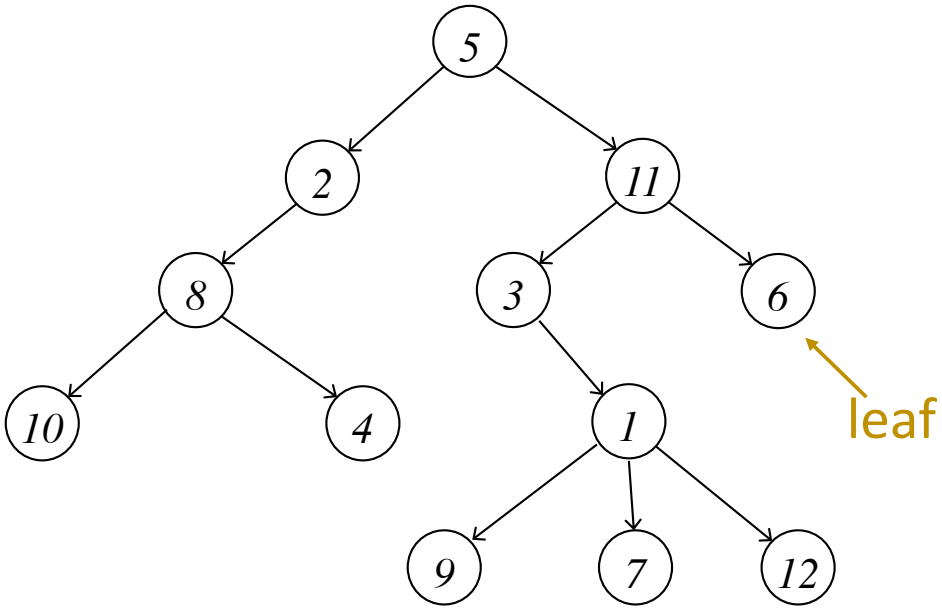
sibling Relationship



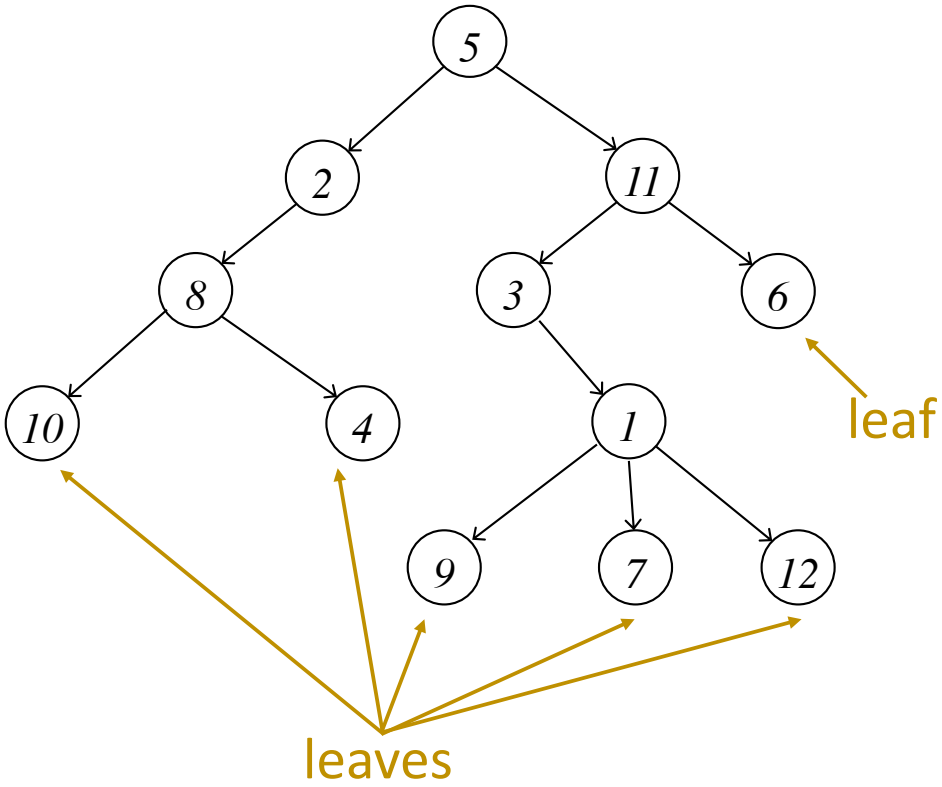
Kind of Nodes



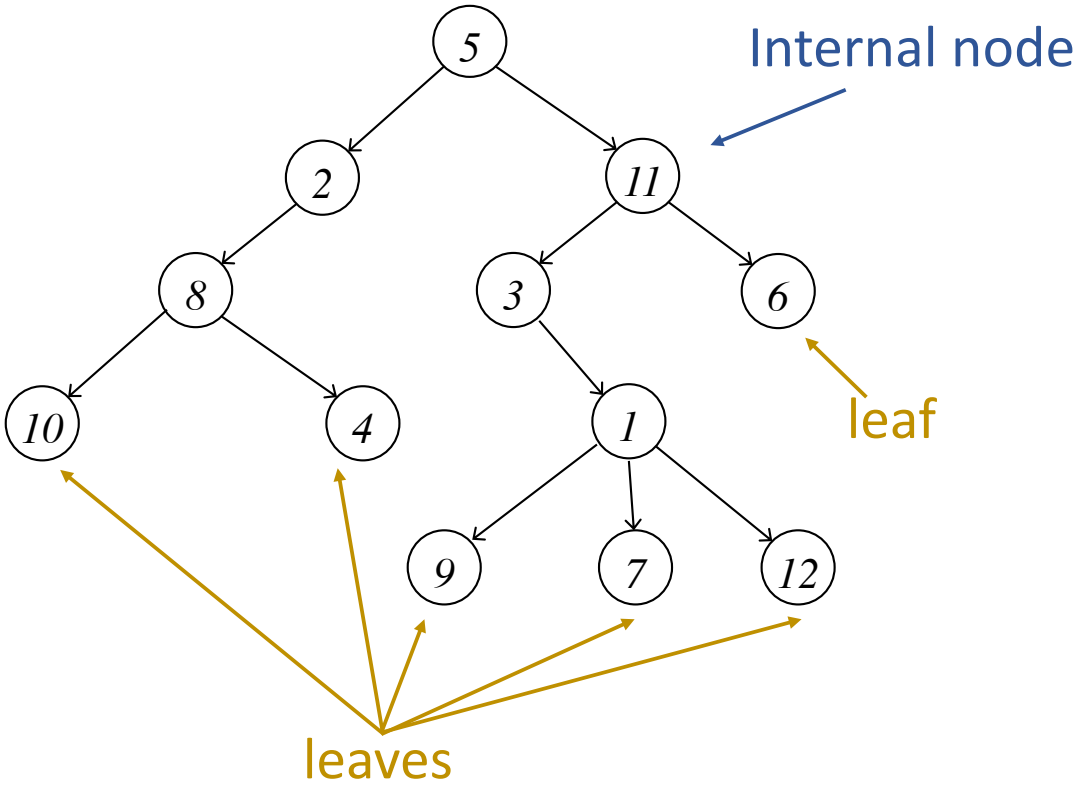
Kind of Nodes



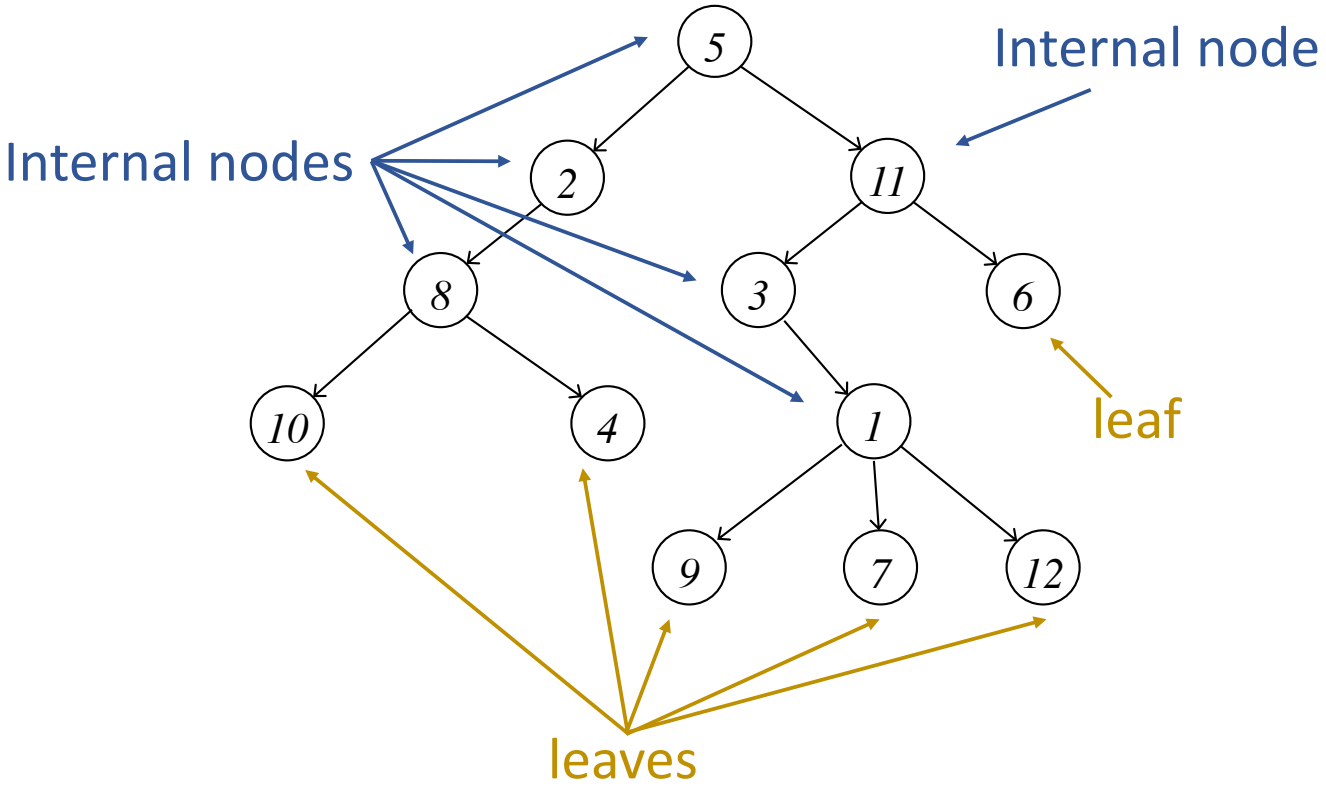
Kind of Nodes



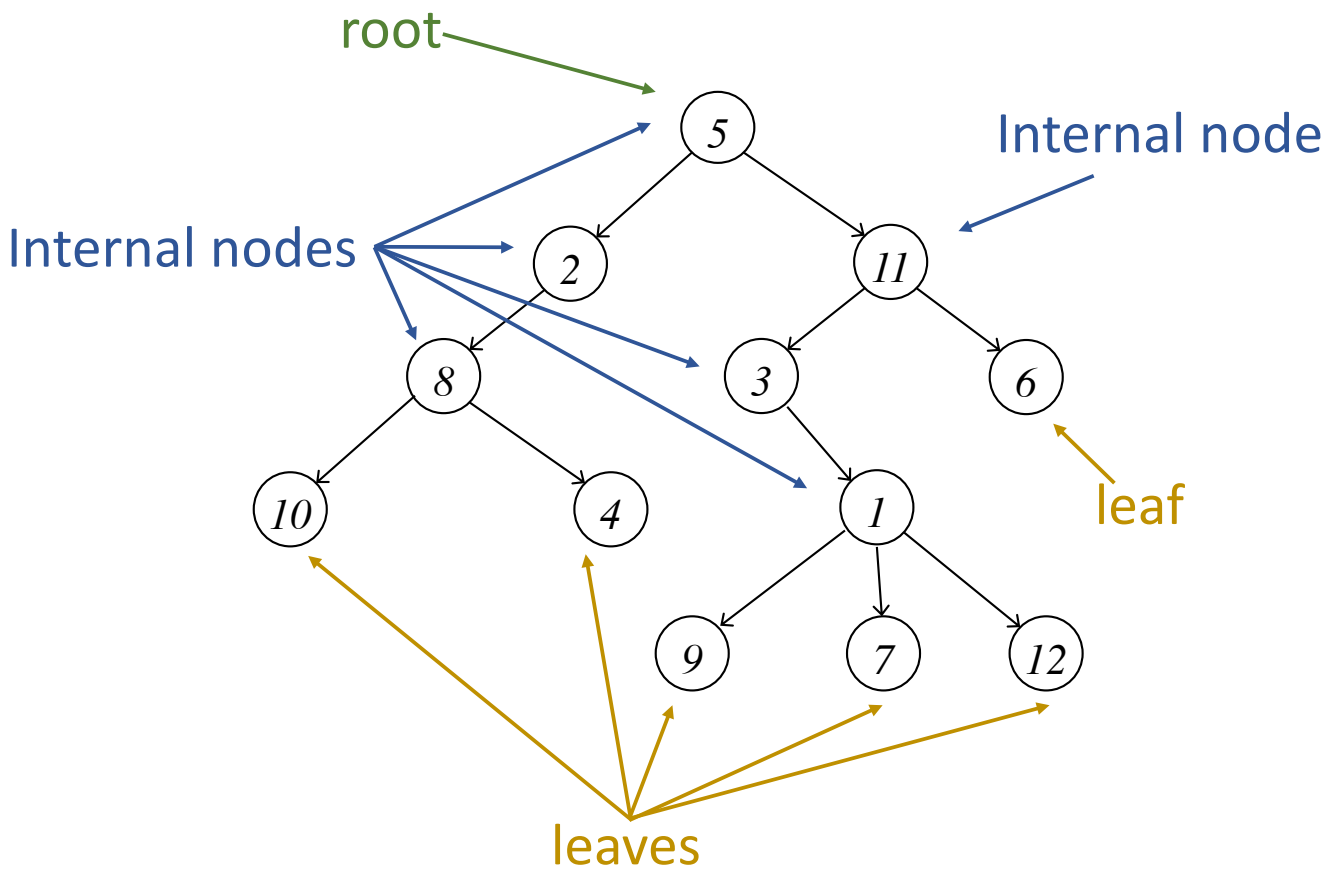
Kind of Nodes



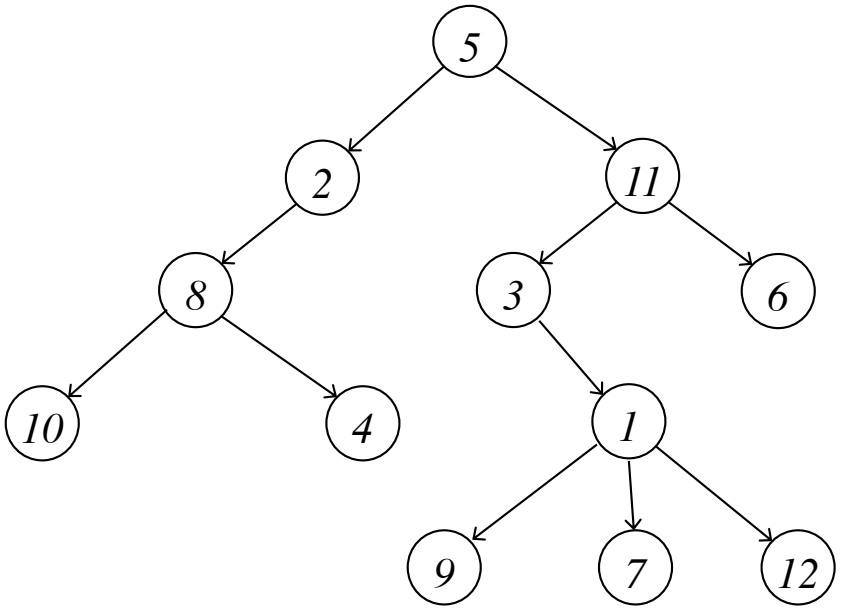
Kind of Nodes



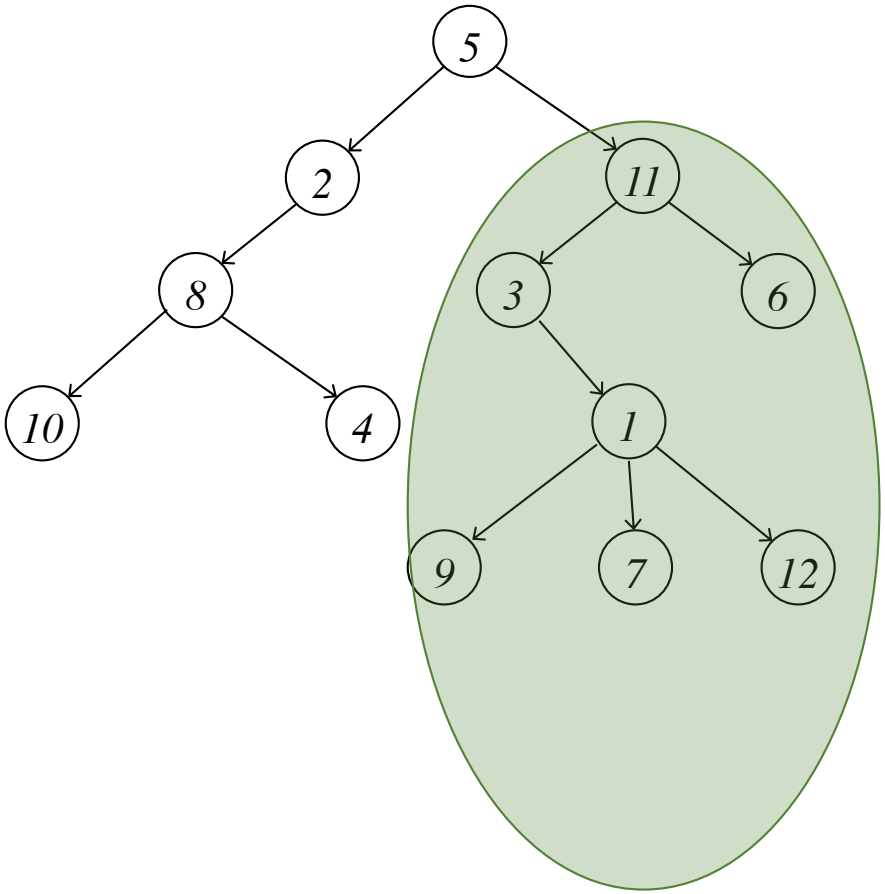
Kind of Nodes



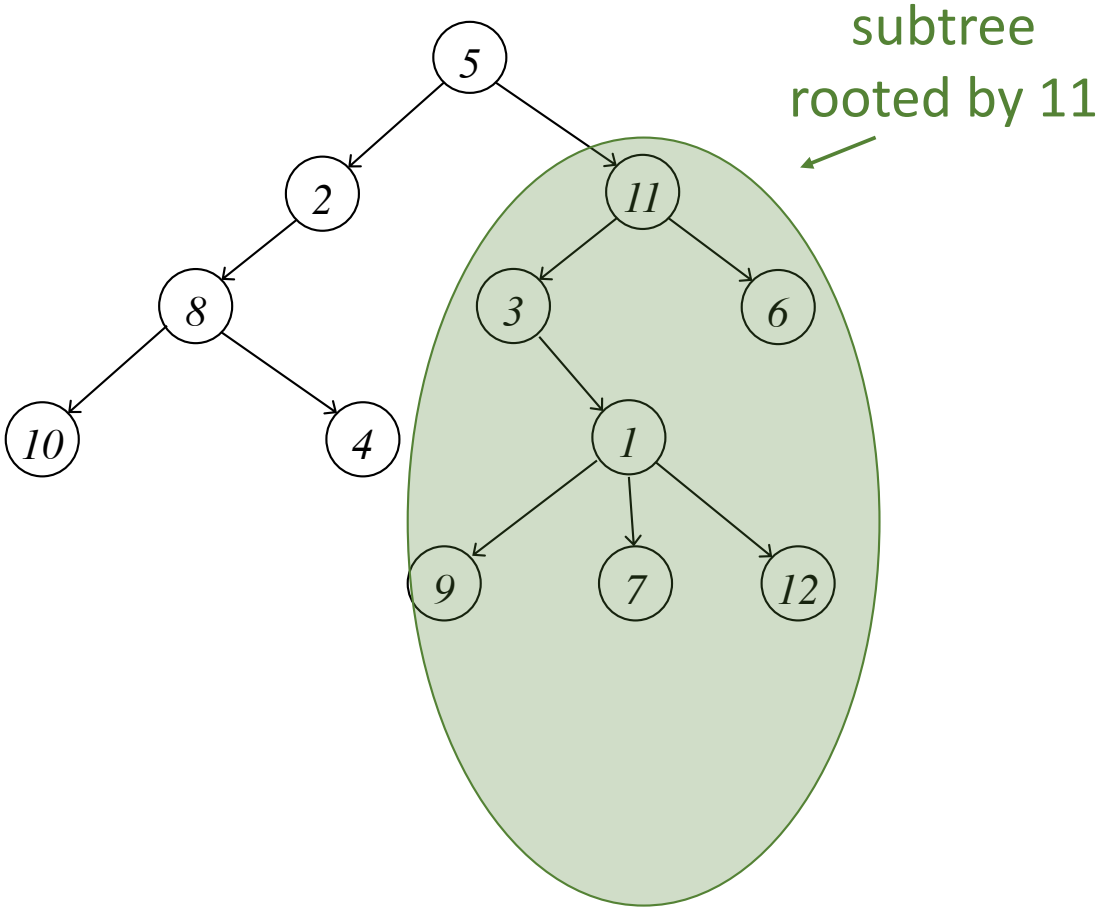
Subtrees



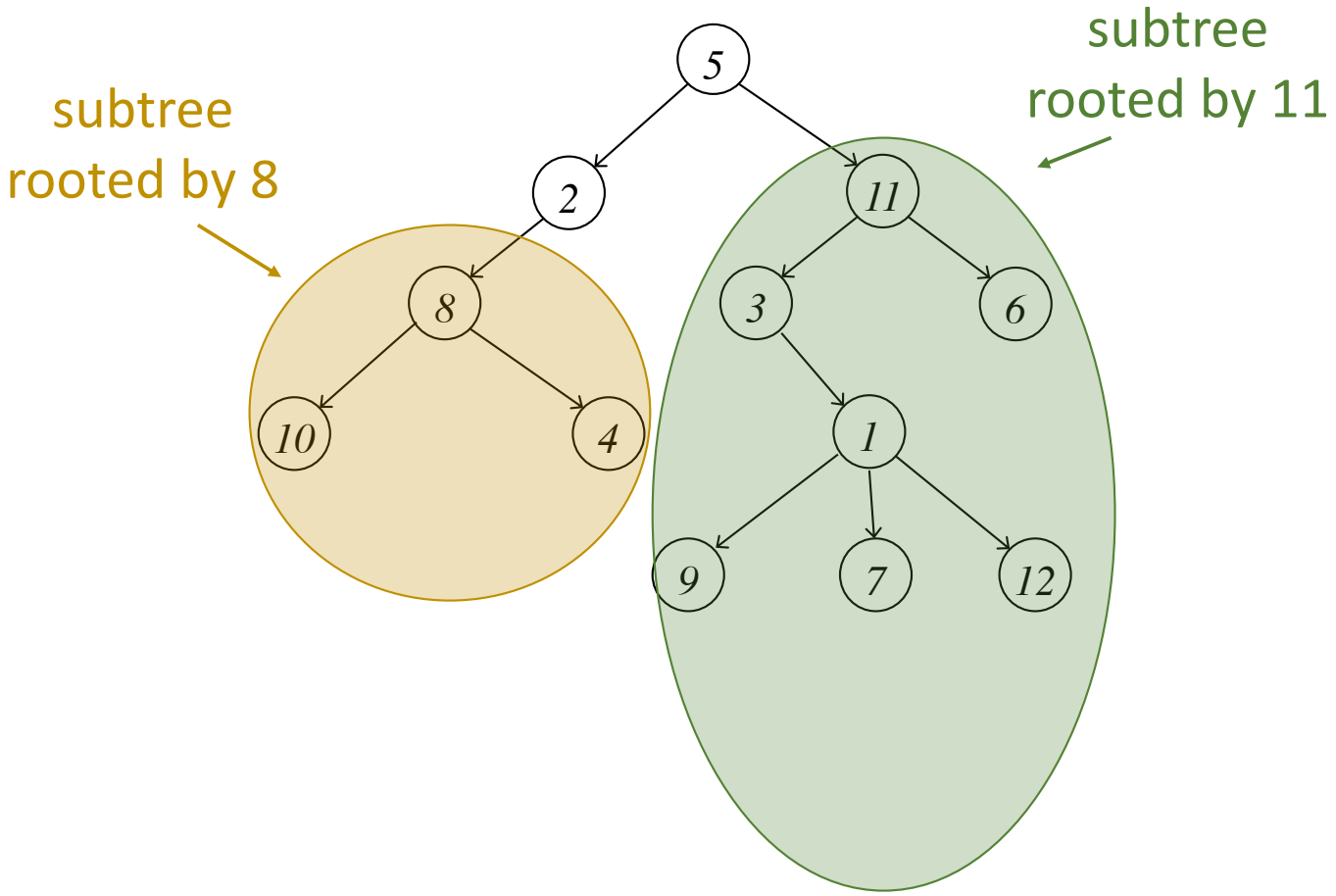
Subtrees



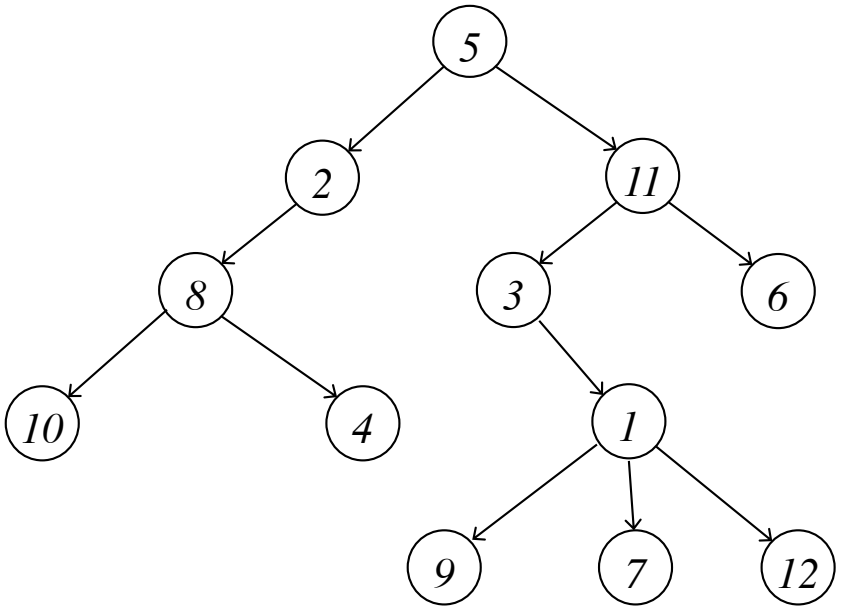
Subtrees



Subtrees

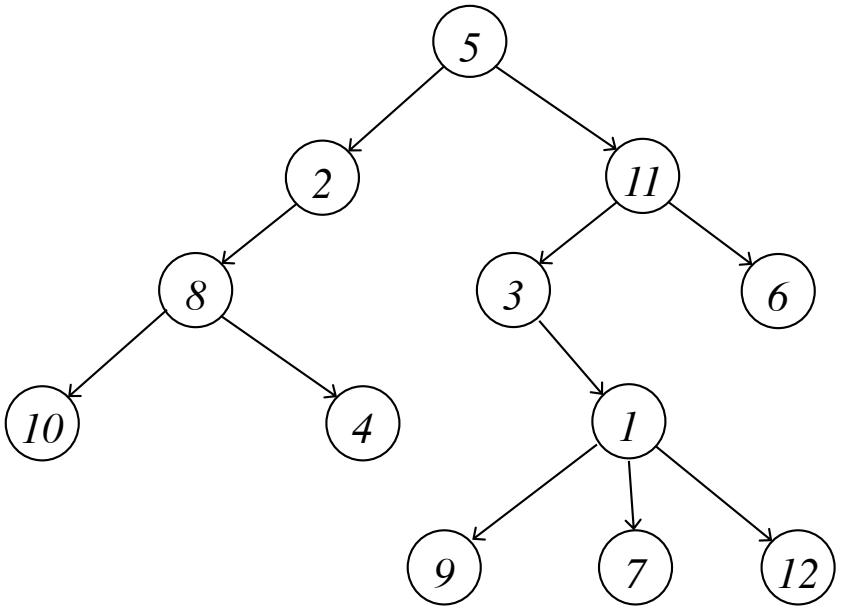


Definitions



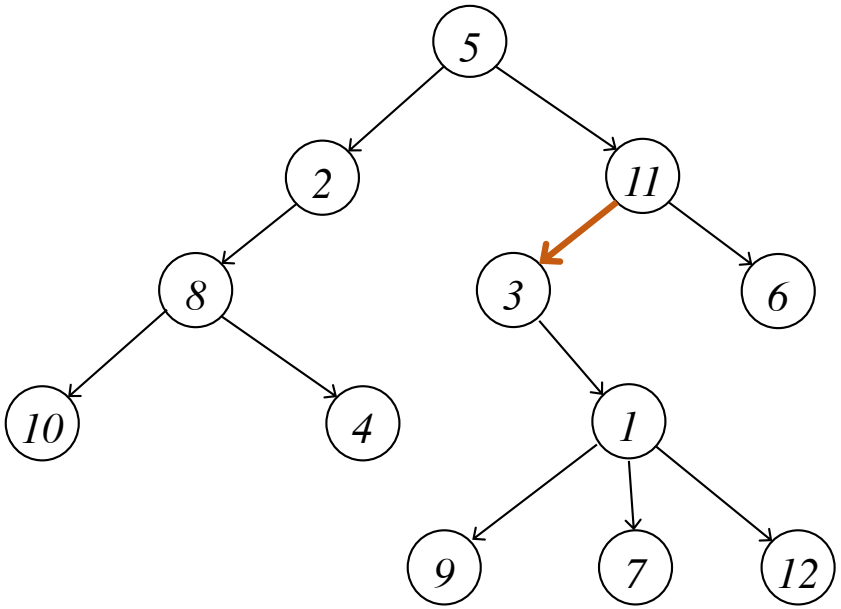
Definitions

Edge



Definitions

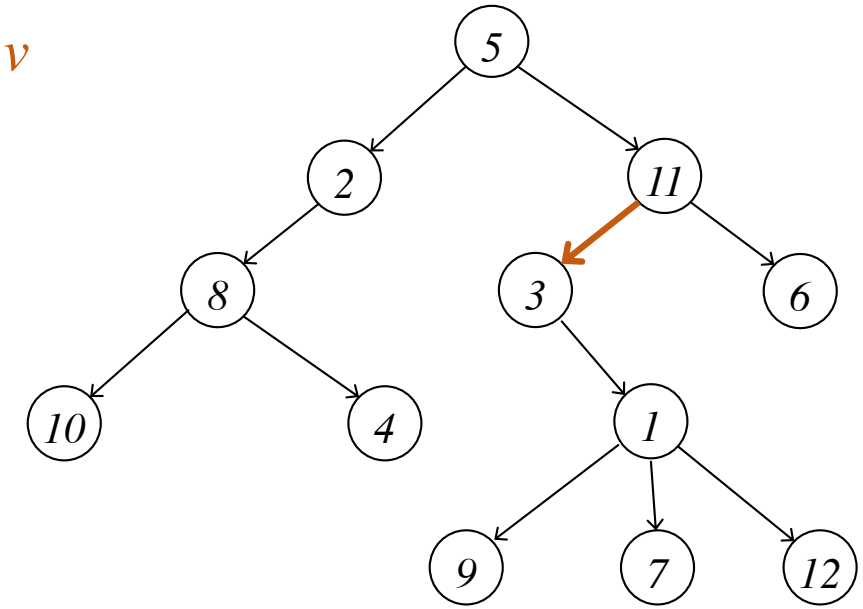
Edge



Definitions

Edge

(u, v) is an edge, if u is the parent of v

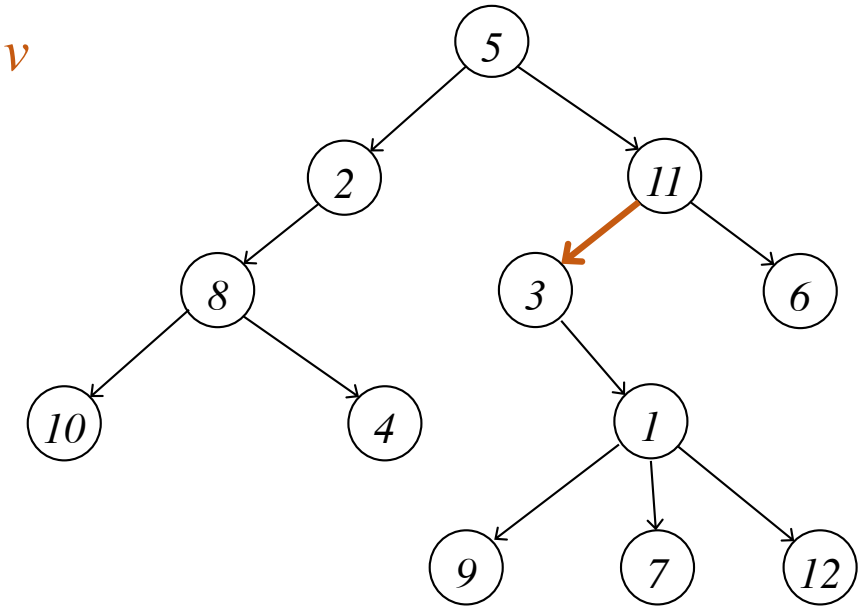


Definitions

Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

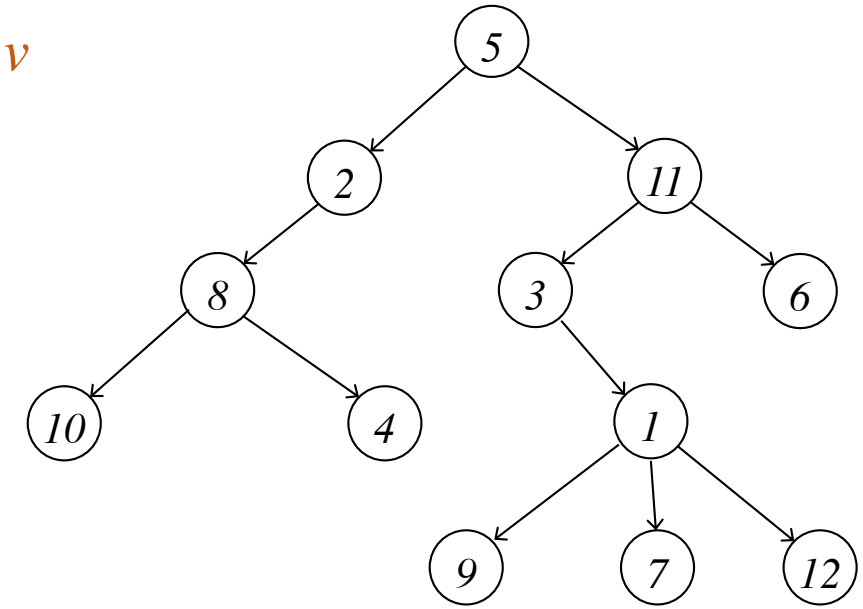


Definitions

Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge



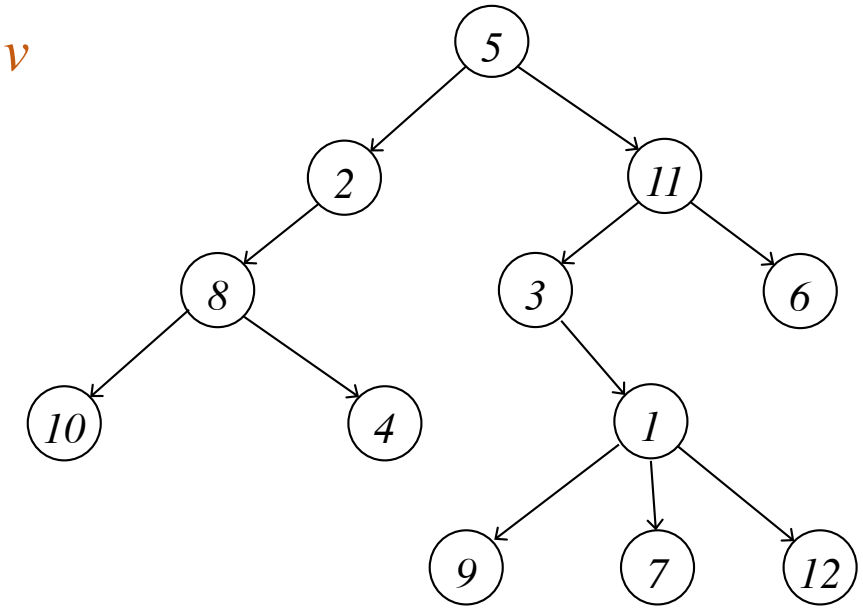
Definitions

Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

Path



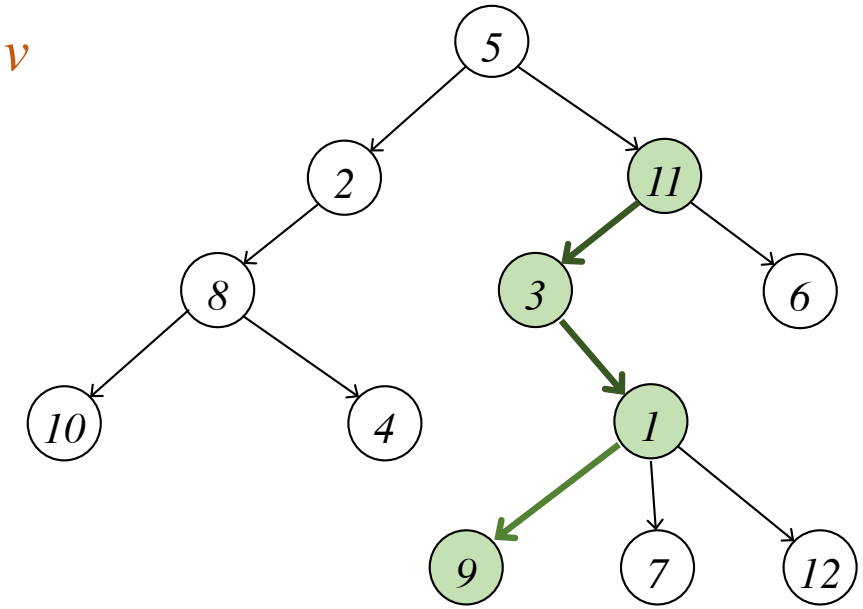
Definitions

Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

Path



Definitions

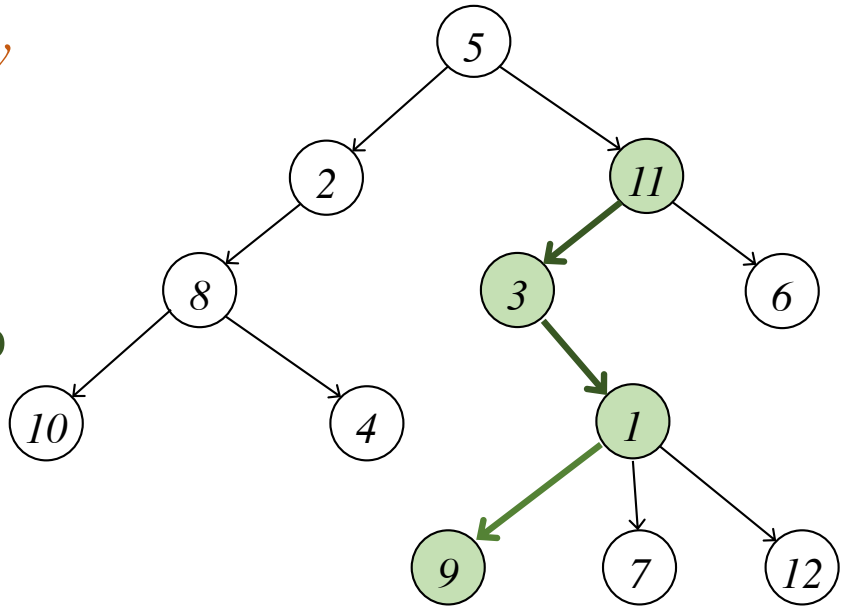
Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

Path

$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge



Definitions

Edge

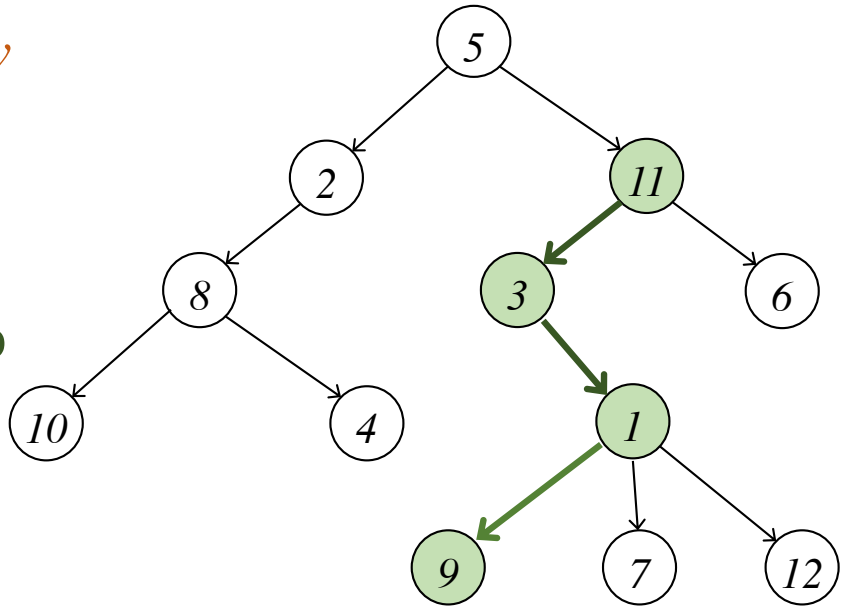
(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

Path

$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge

Example: $p_1=(11, 3, 1, 9)$ is a path



Definitions

Edge

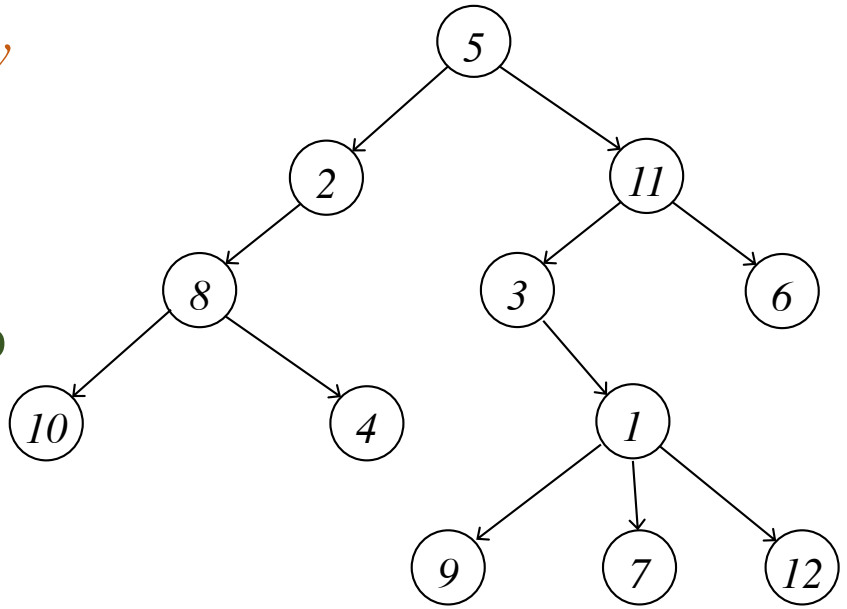
(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

Path

$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge

Example: $p_1=(11, 3, 1, 9)$ is a path



Definitions

Edge

(u, v) is an edge, if u is the parent of v

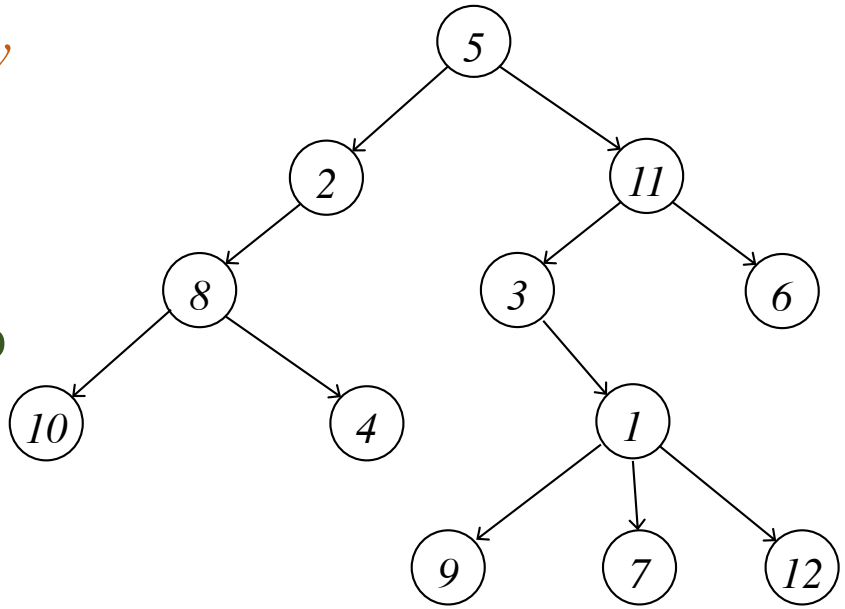
Example: $(11, 3)$ is an edge

Path

$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge

Example: $p_1=(11, 3, 1, 9)$ is a path

Length of a path



Definitions

Edge

(u, v) is an edge, if u is the parent of v

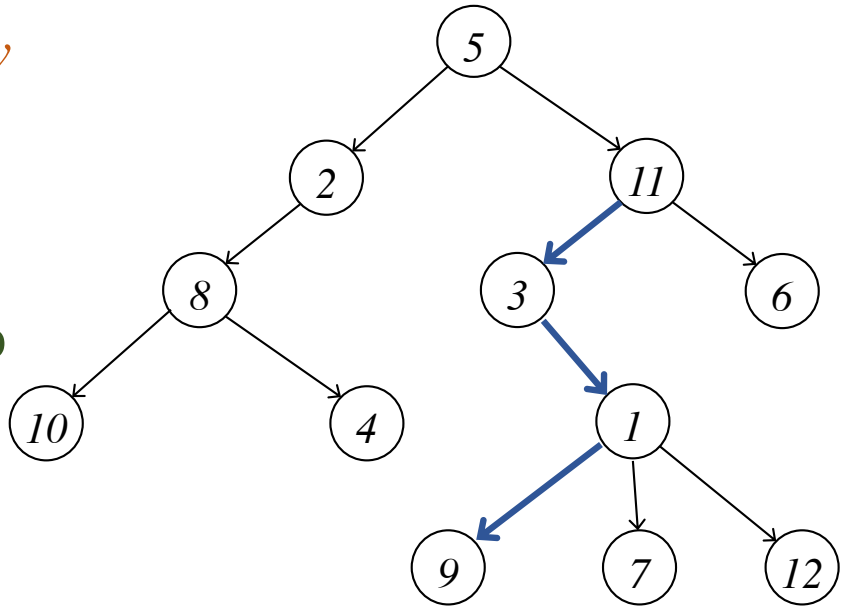
Example: $(11, 3)$ is an edge

Path

$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge

Example: $p_1=(11, 3, 1, 9)$ is a path

Length of a path



Definitions

Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

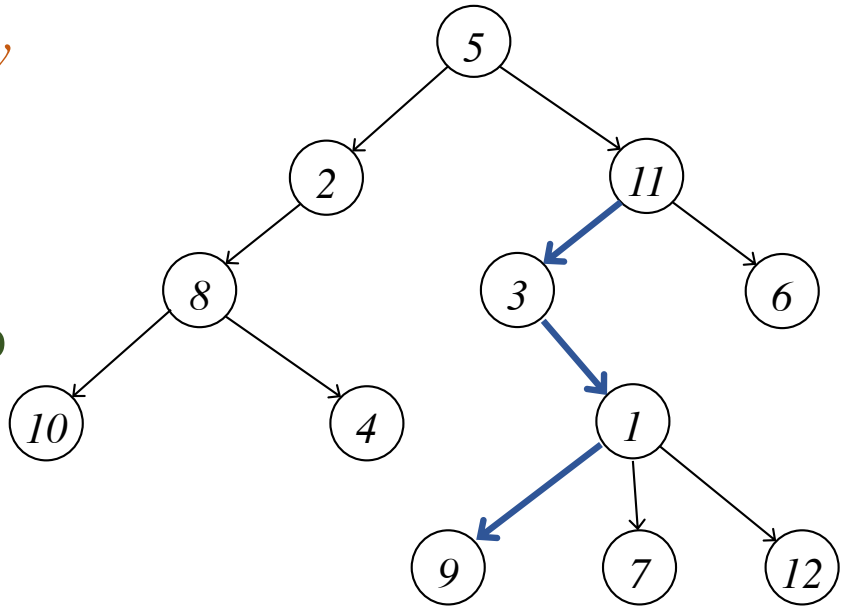
Path

$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge

Example: $p_1=(11, 3, 1, 9)$ is a path

Length of a path

$|p|$= # of edges in p



Definitions

Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

Path

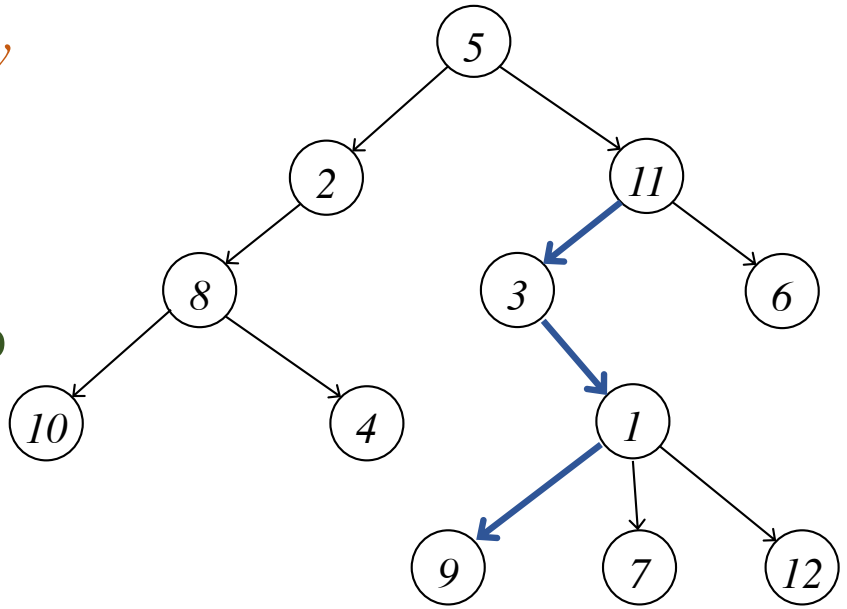
$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge

Example: $p_1=(11, 3, 1, 9)$ is a path

Length of a path

$|p|$= # of edges in p

Example: $|p_1|=3$



Definitions

Edge

(u, v) is an edge, if u is the parent of v

Example: $(11, 3)$ is an edge

Path

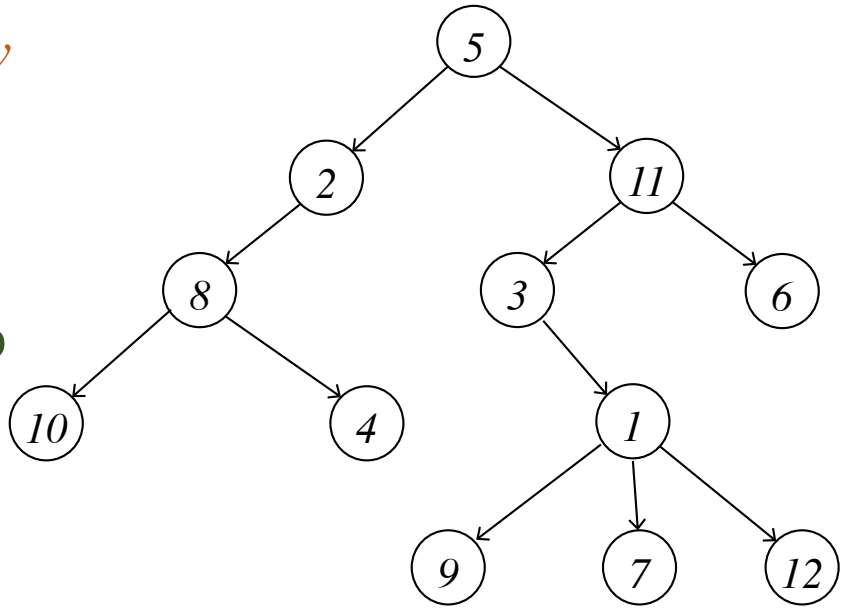
$p=(v_1, v_2, \ldots, v_k)$ is a path, if each two consecutive nodes, forms an edge

Example: $p_1=(11, 3, 1, 9)$ is a path

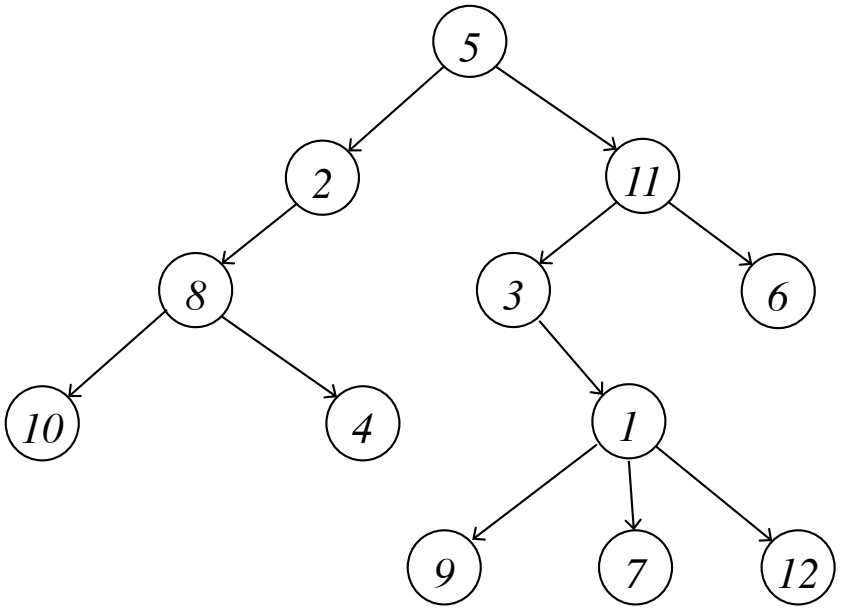
Length of a path

$|p|$= # of edges in p

Example: $|p_1|=3$

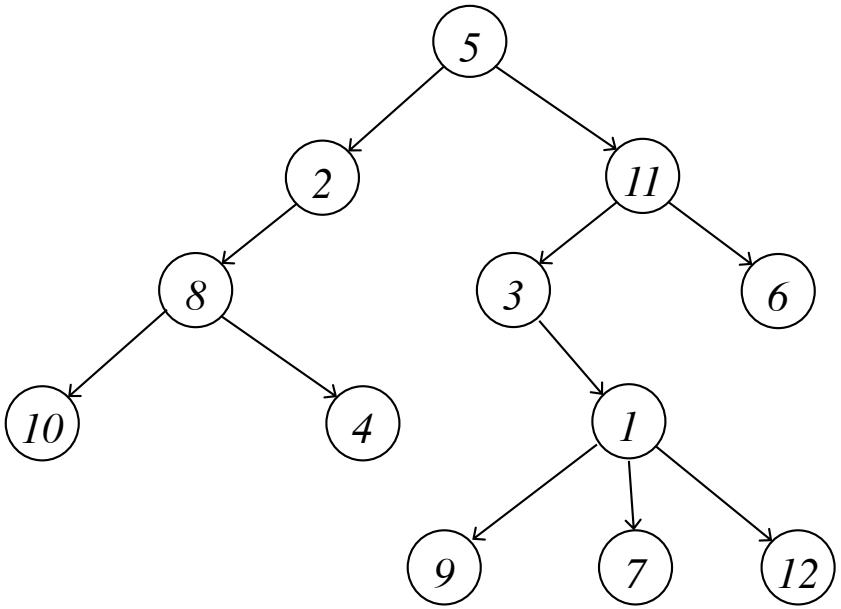


Definitions



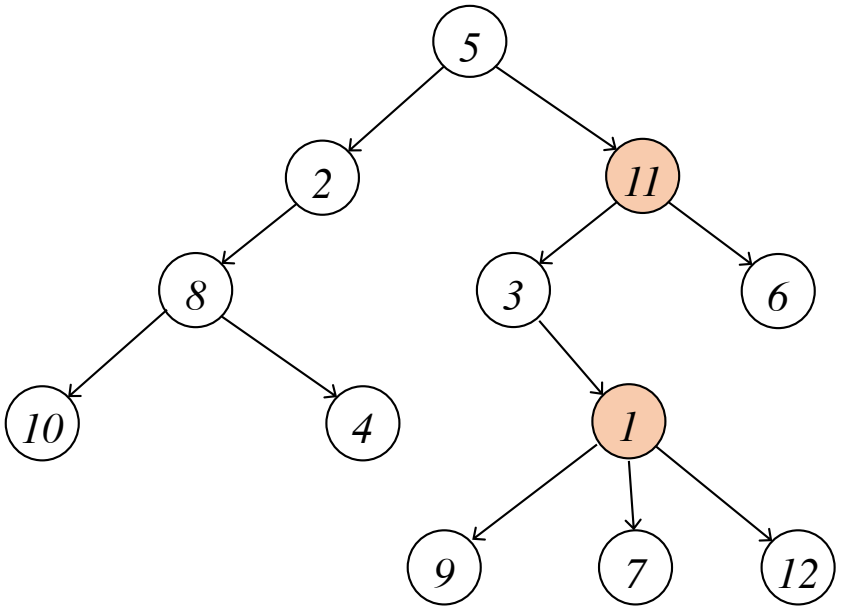
Definitions

Ancestor



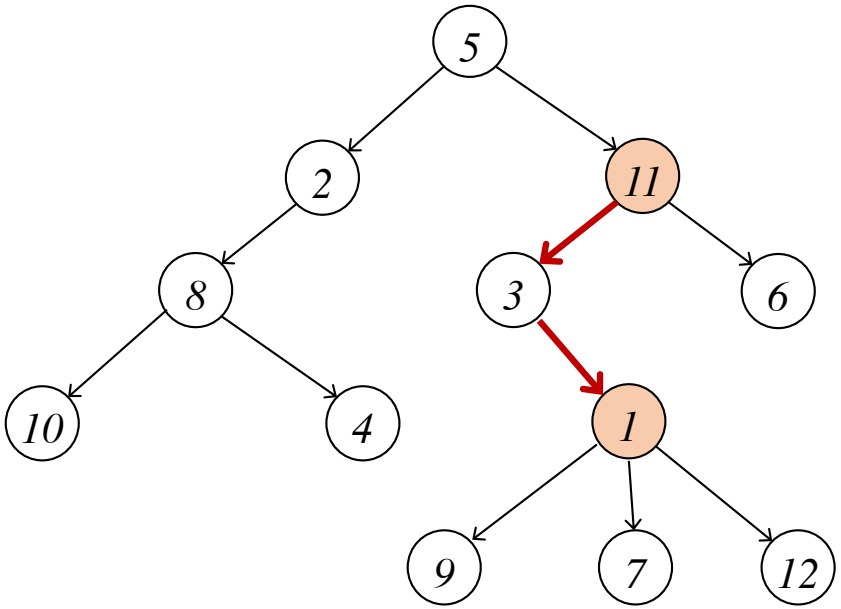
Definitions

Ancestor



Definitions

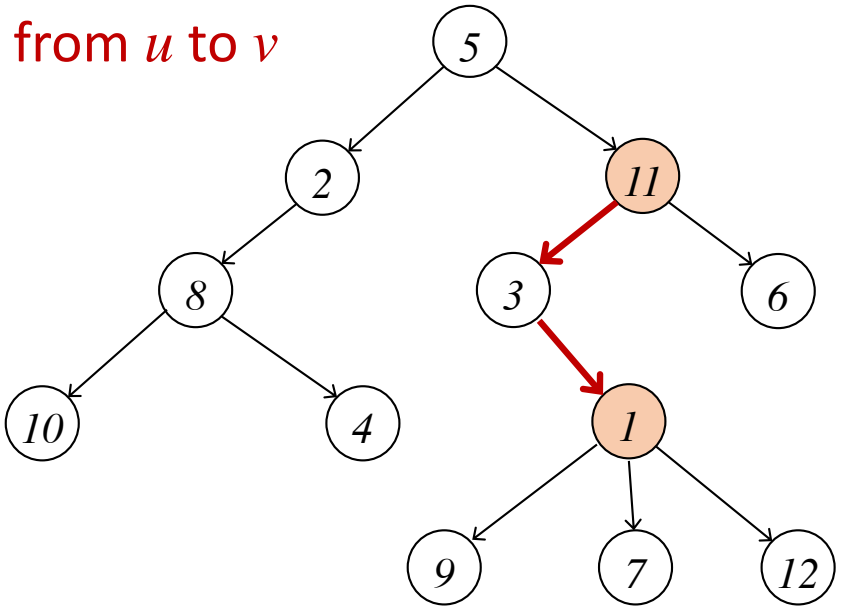
Ancestor



Definitions

Ancestor

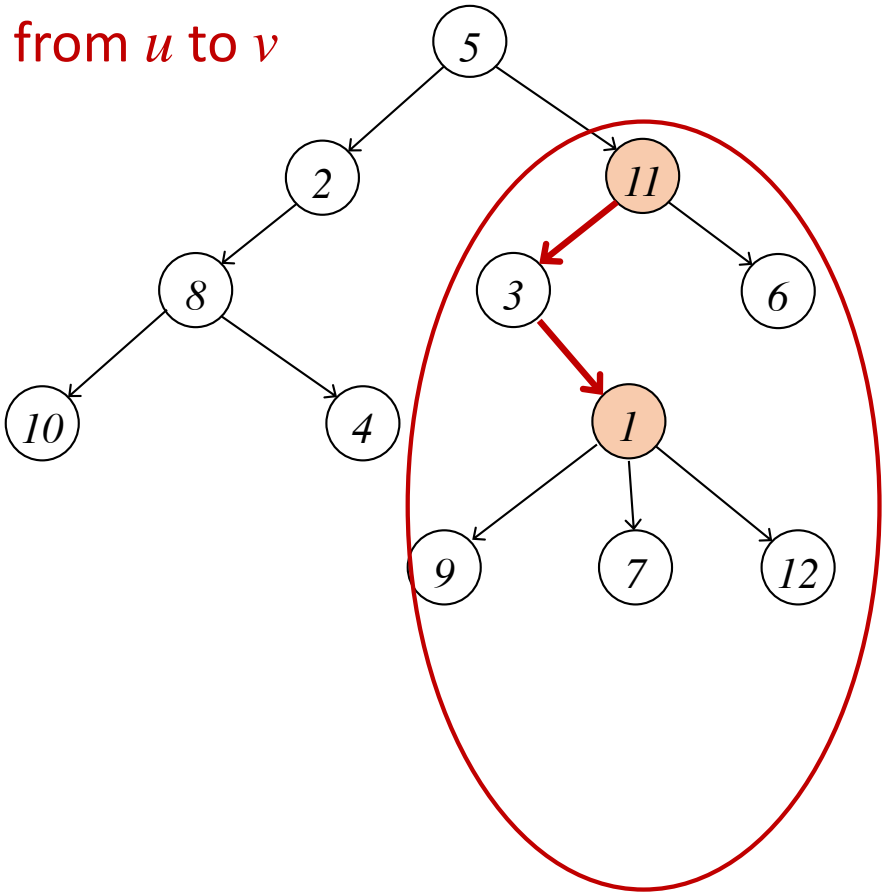
u is an ancestor of *v*, if there is a path from *u* to *v*



Definitions

Ancestor

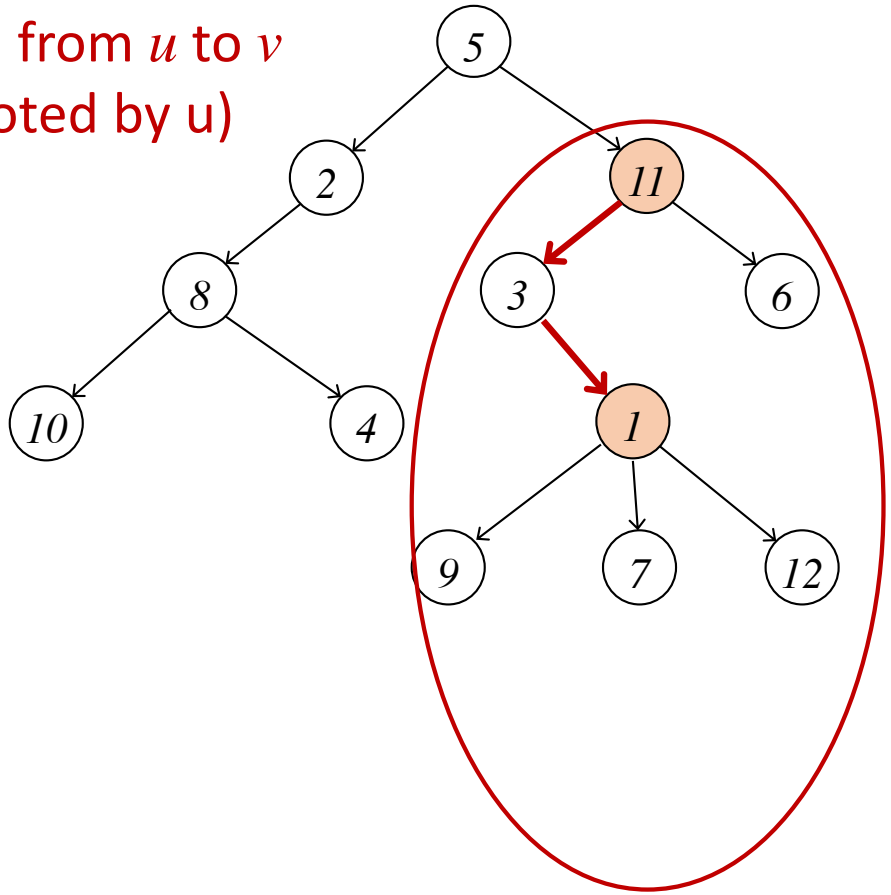
u is an ancestor of *v*, if there is a path from *u* to *v*



Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v*
(alternatively, if v is in the subtree rooted by u)

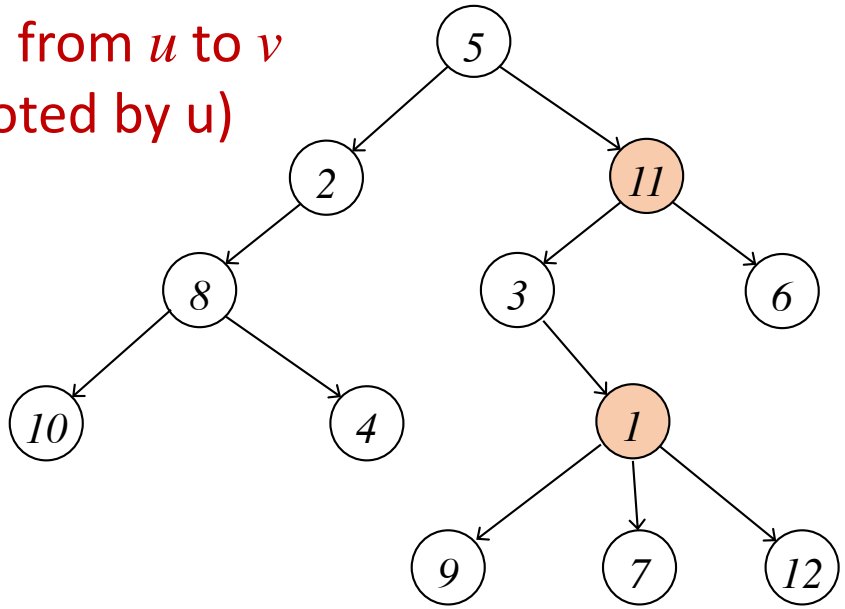


Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v*
(alternatively, if v is in the subtree rooted by u)
Examples:

•



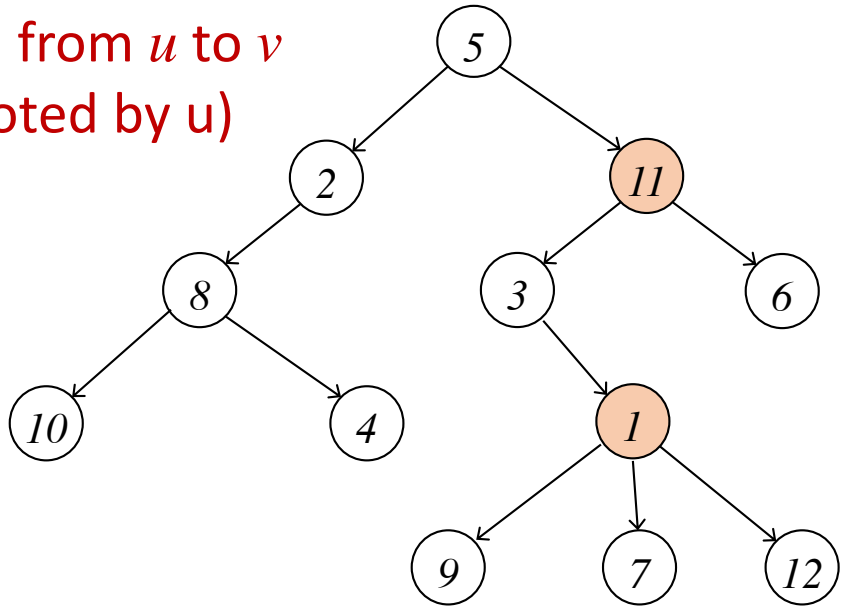
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*

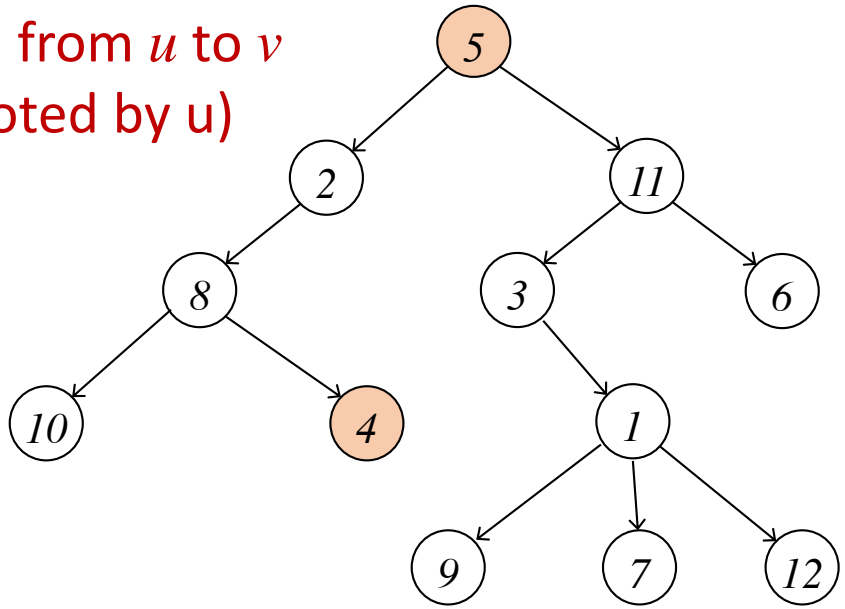


Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v*
(alternatively, if v is in the subtree rooted by u)
Examples:

- *11* is an ancestor of *1*
-



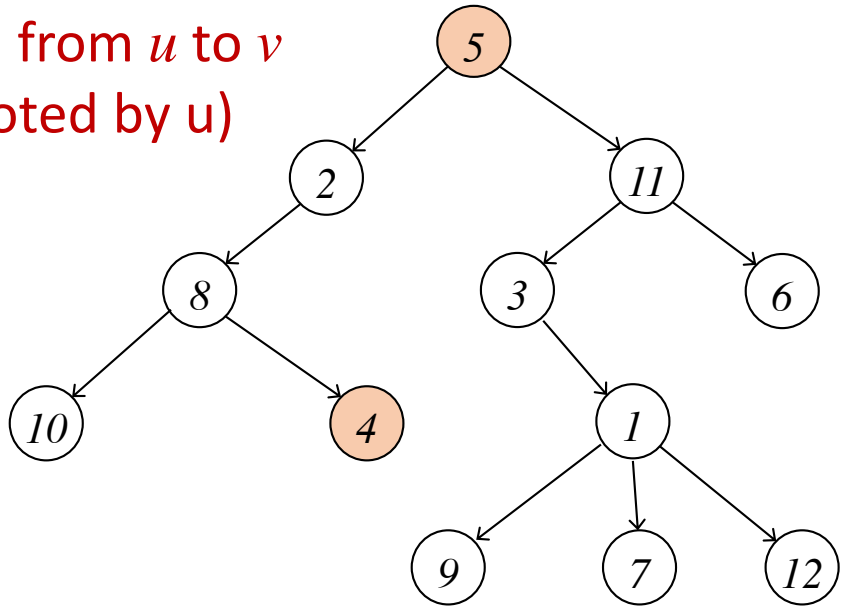
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*



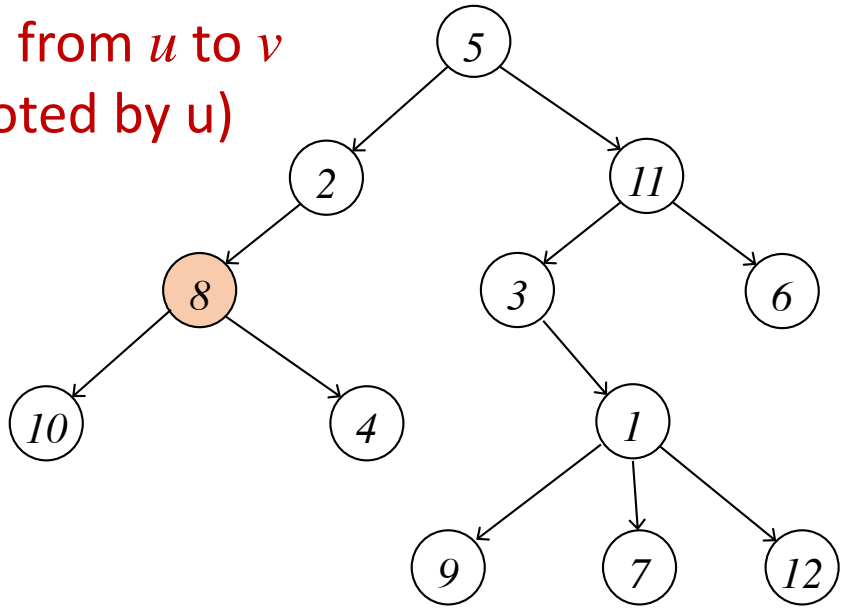
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
-



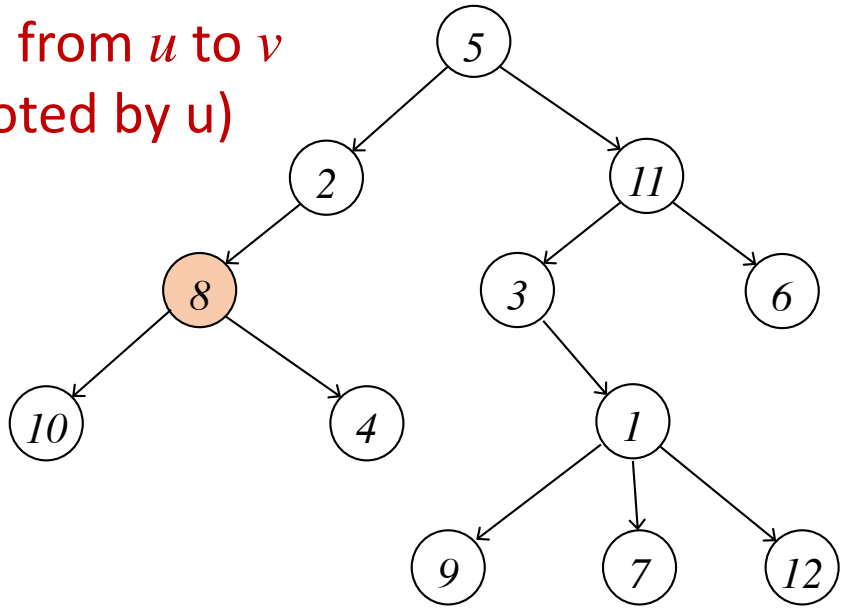
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*



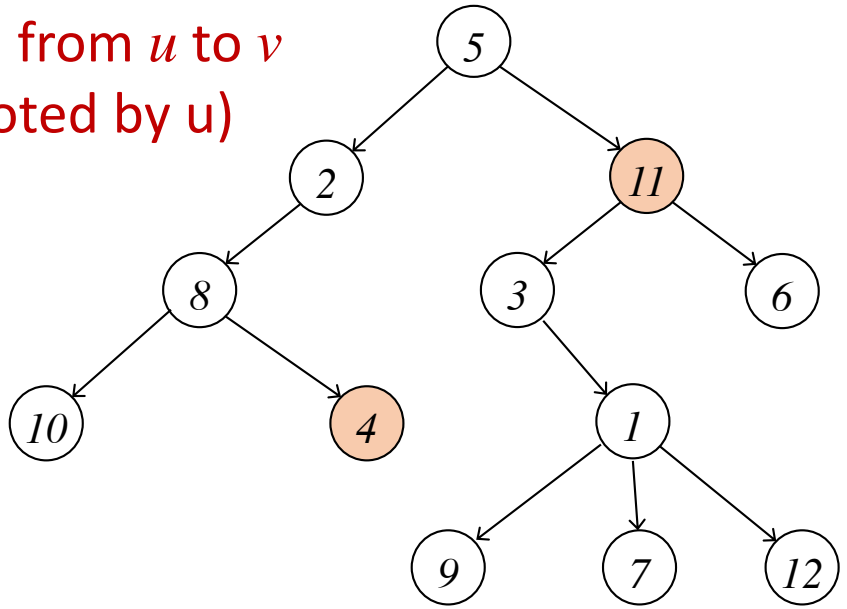
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
-



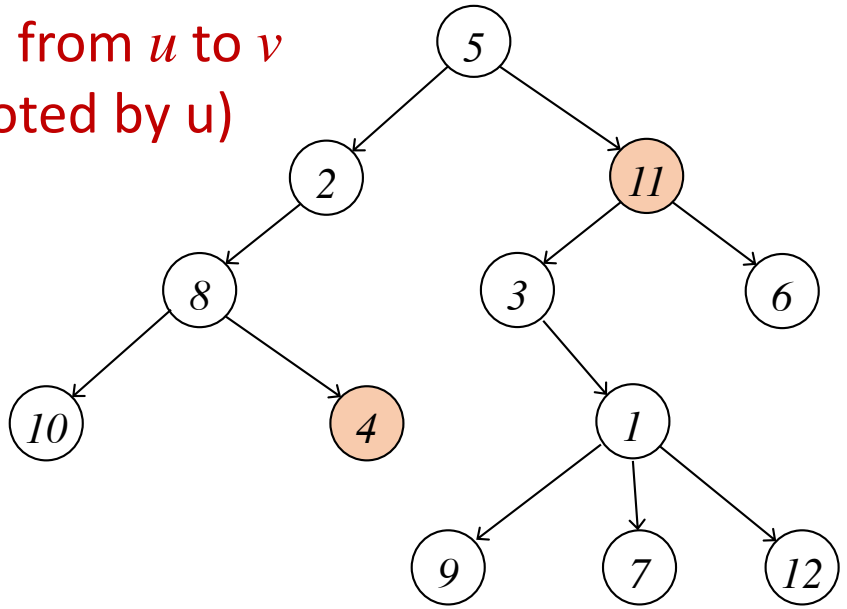
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is <u>not</u> an ancestor of *11*



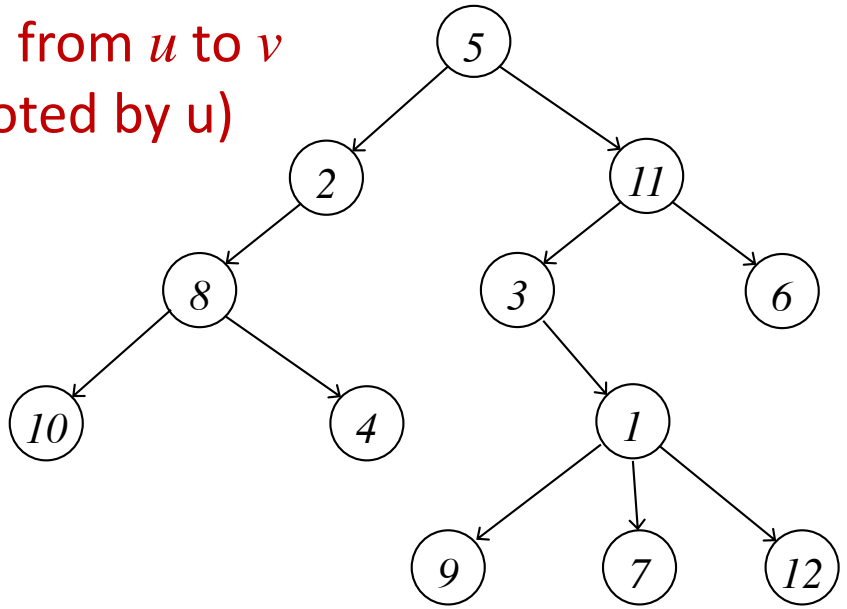
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



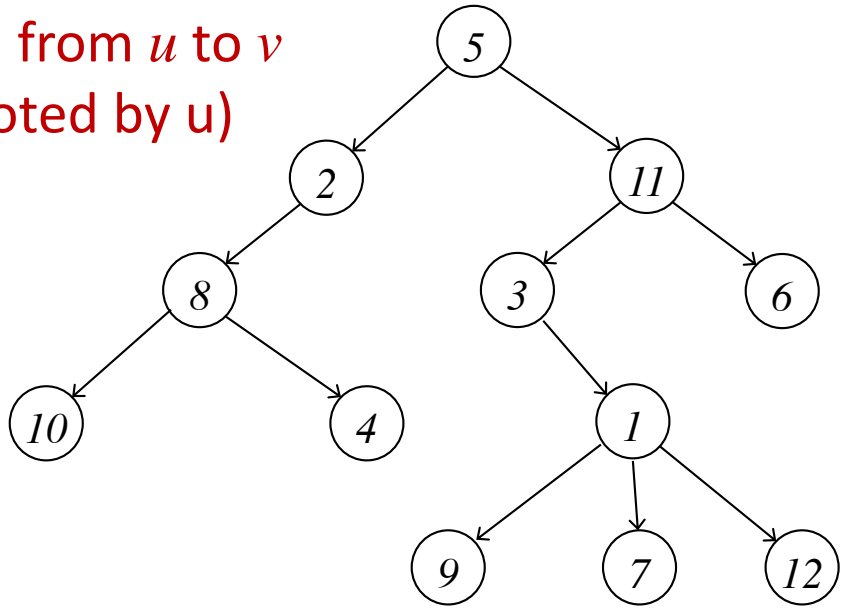
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



Descendant

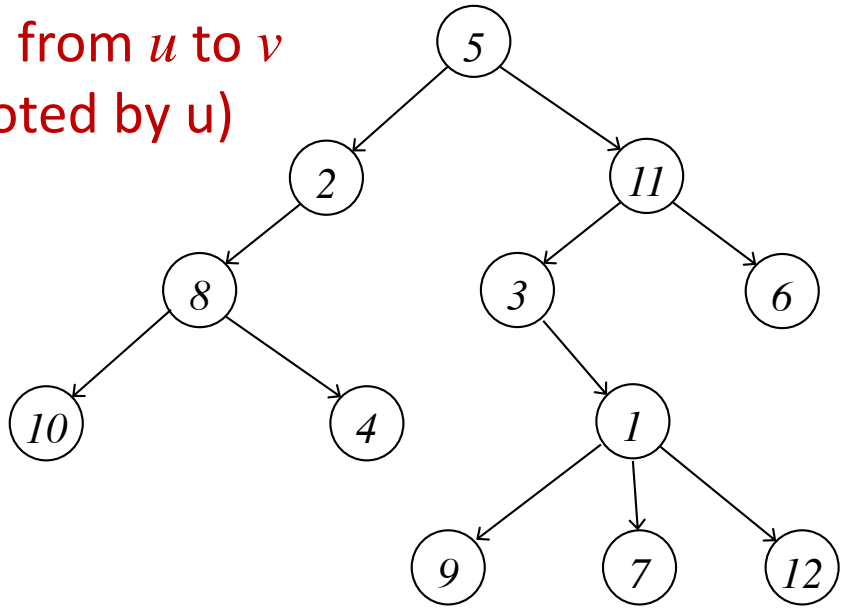
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



Descendant

u is a descendant of *v*, if *v* is an ancestor of *u*

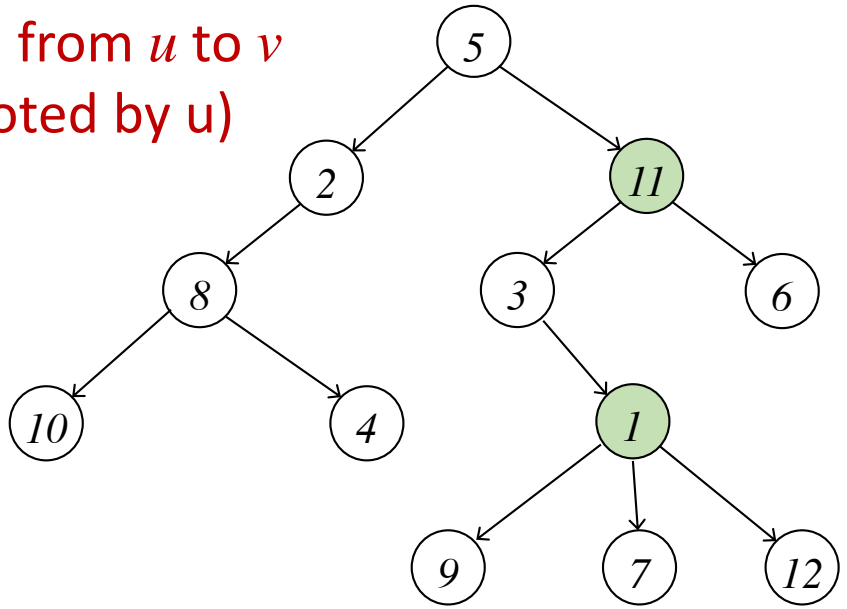
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



Descendant

u is a descendant of *v*, if *v* is an ancestor of *u*

Examples:

-

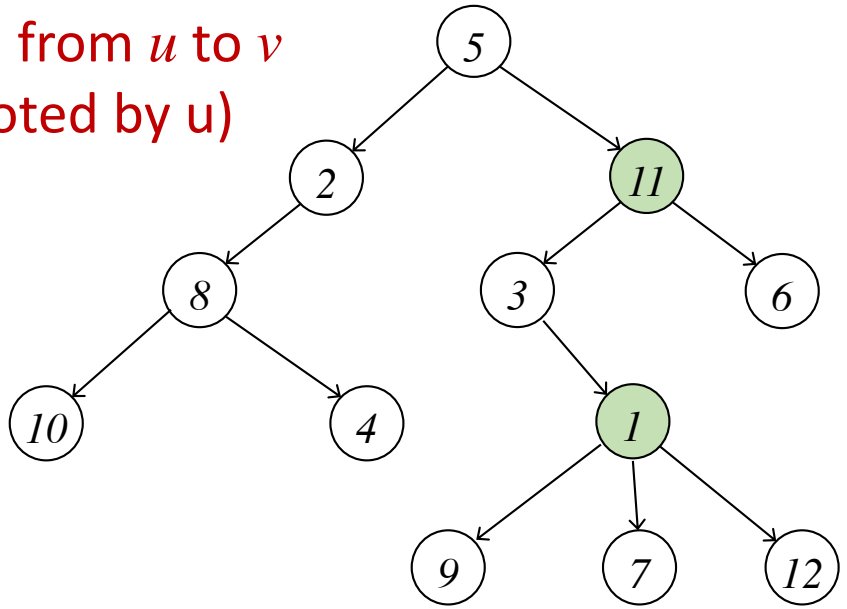
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



Descendant

u is a descendant of *v*, if *v* is an ancestor of *u*

Examples:

- *1* is a descendant of *11*

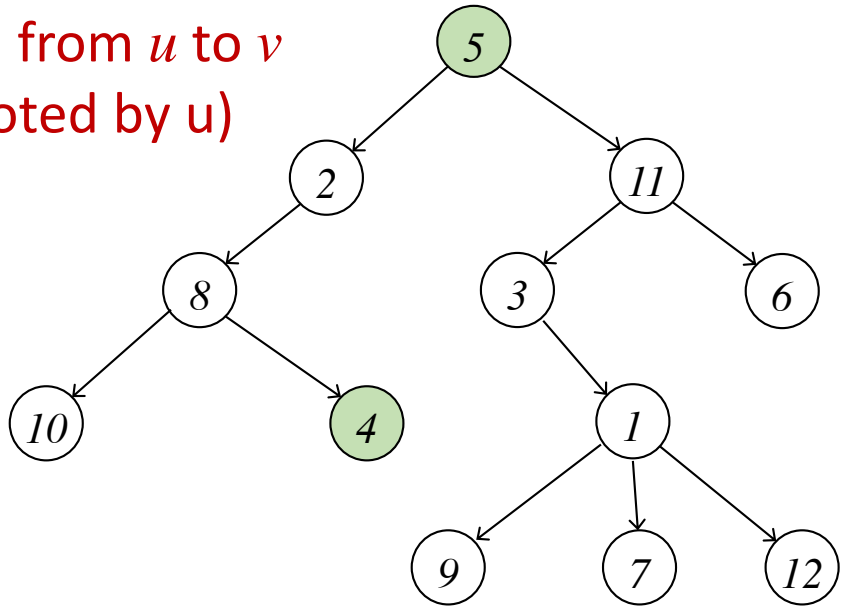
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



Descendant

u is a descendant of *v*, if *v* is an ancestor of *u*

Examples:

- *1* is a descendant of *11*
-

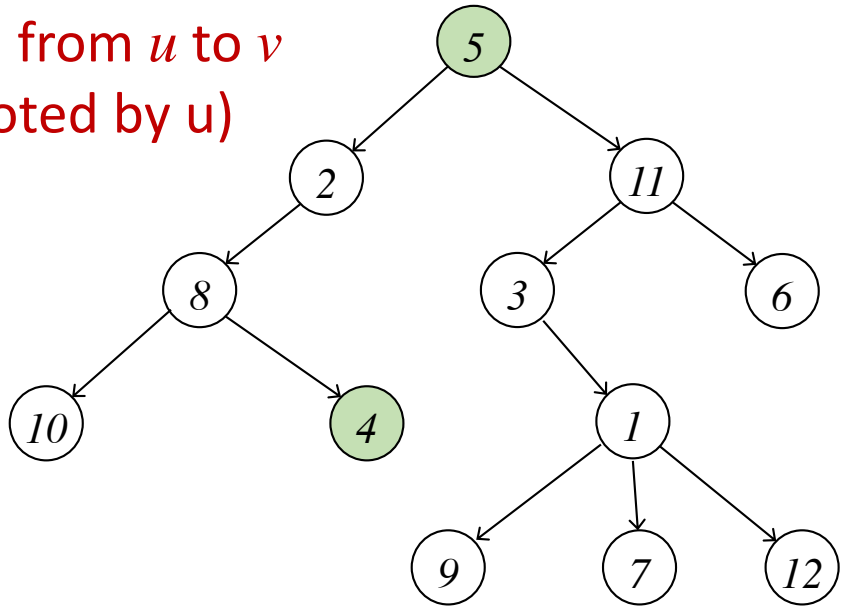
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



Descendant

u is a descendant of *v*, if *v* is an ancestor of *u*

Examples:

- *1* is a descendant of *11*
- *4* is a descendant of *5*

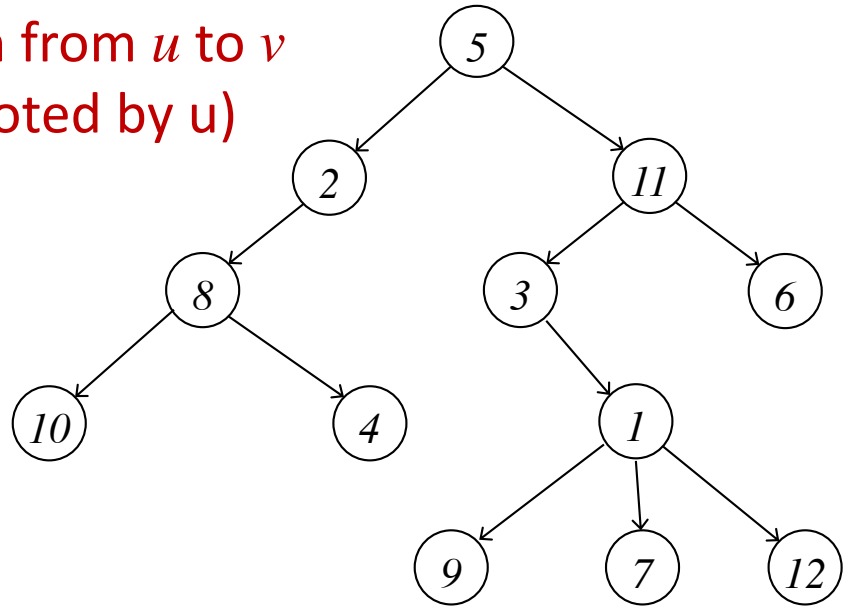
Definitions

Ancestor

u is an ancestor of *v*, if there is a path from *u* to *v* (alternatively, if v is in the subtree rooted by u)

Examples:

- *11* is an ancestor of *1*
- *5* is an ancestor of *4*
- *8* is an ancestor of *8*
- *4* is not an ancestor of *11*



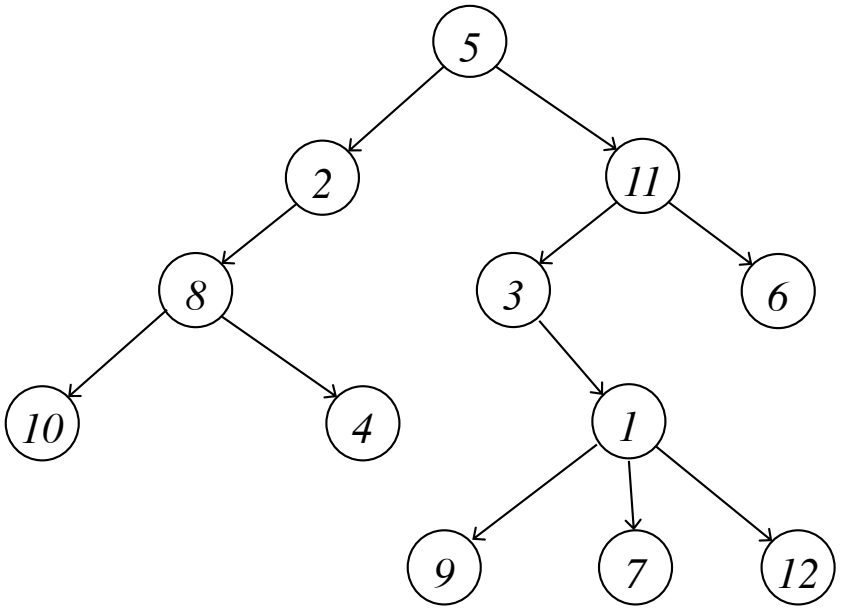
Descendant

u is a descendant of *v*, if *v* is an ancestor of *u*

Examples:

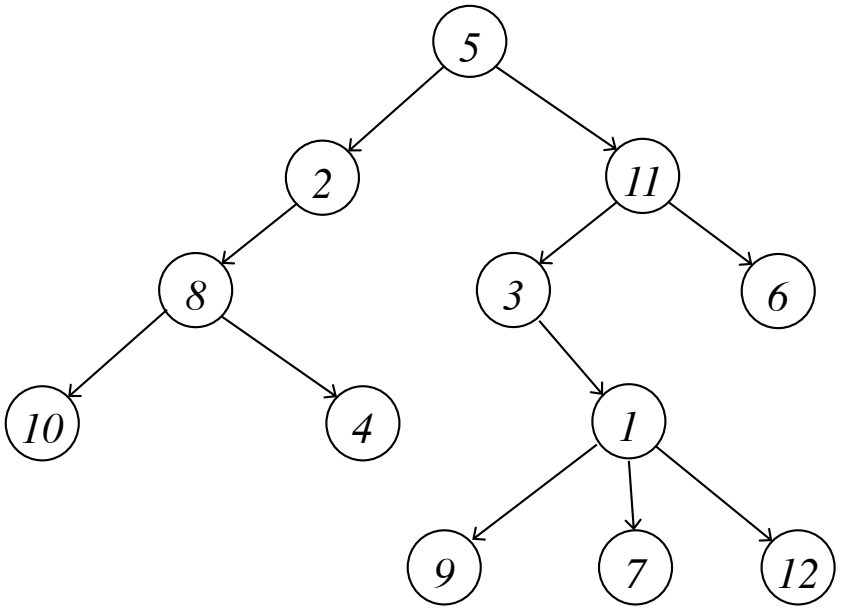
- *1* is a descendant of *11*
- *4* is a descendant of *5*

Definitions



Definitions

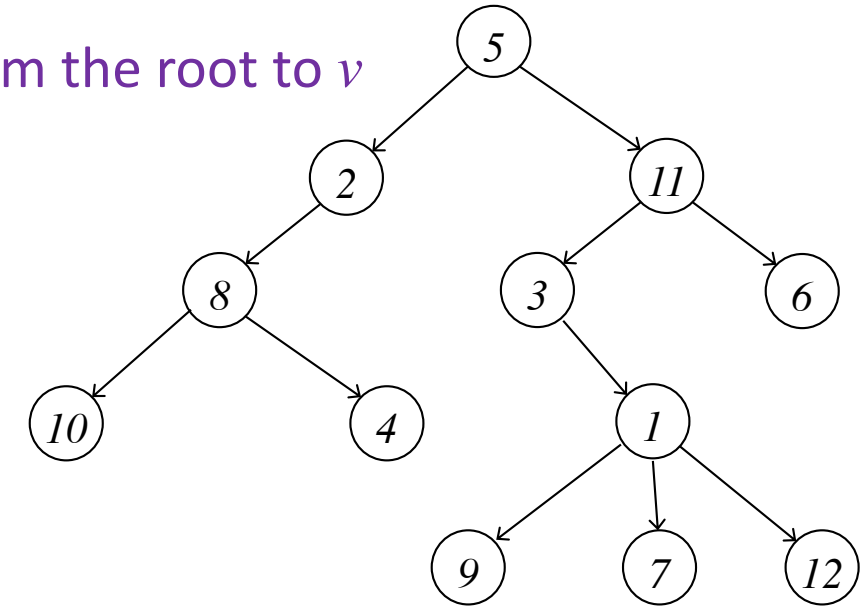
Depth of a node



Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

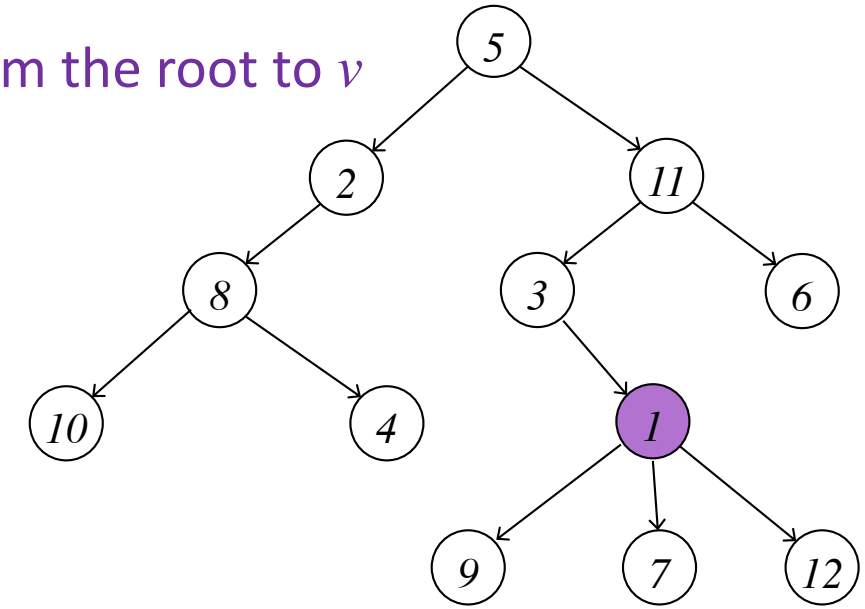


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = ?*

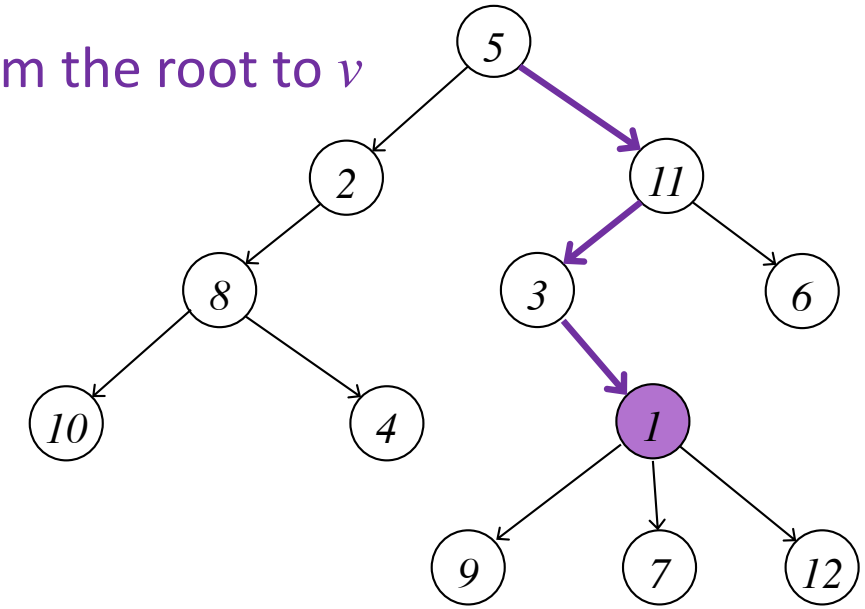


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = ?*

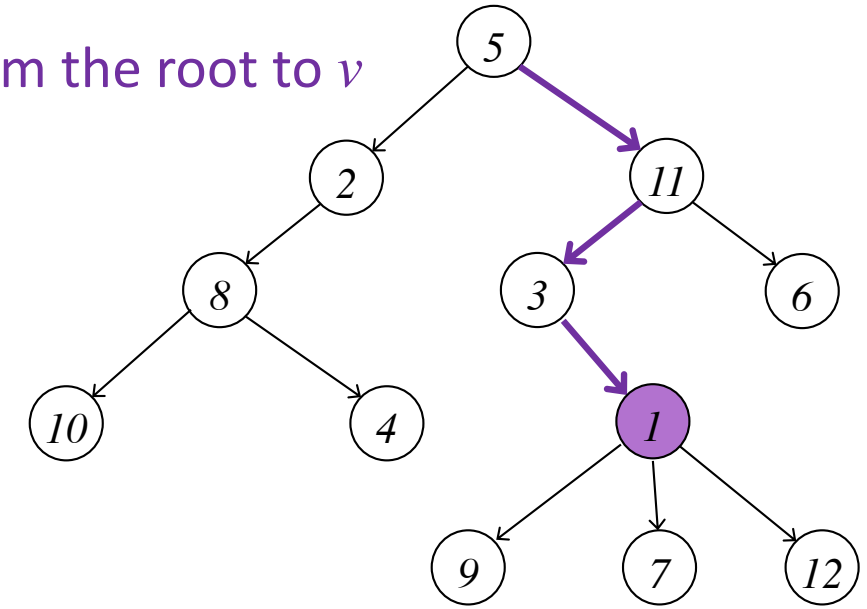


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*

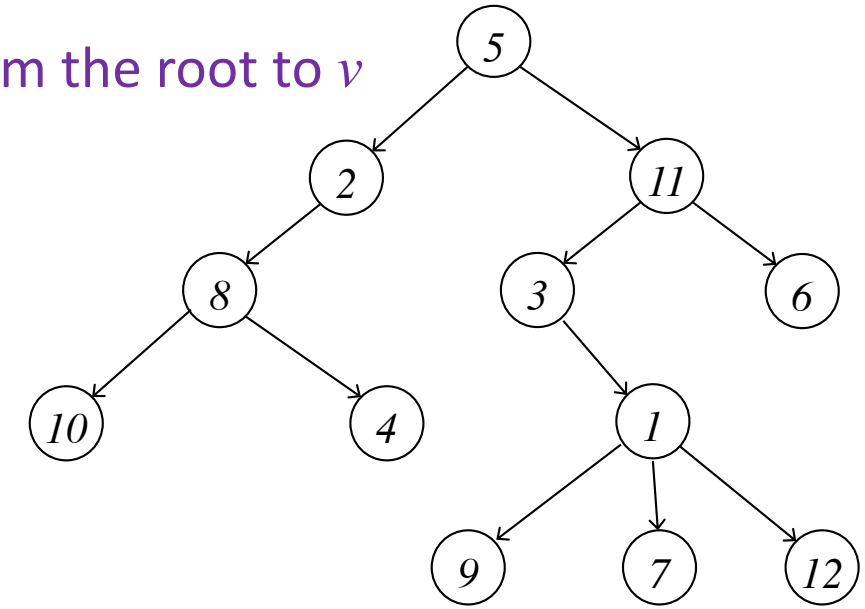


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*

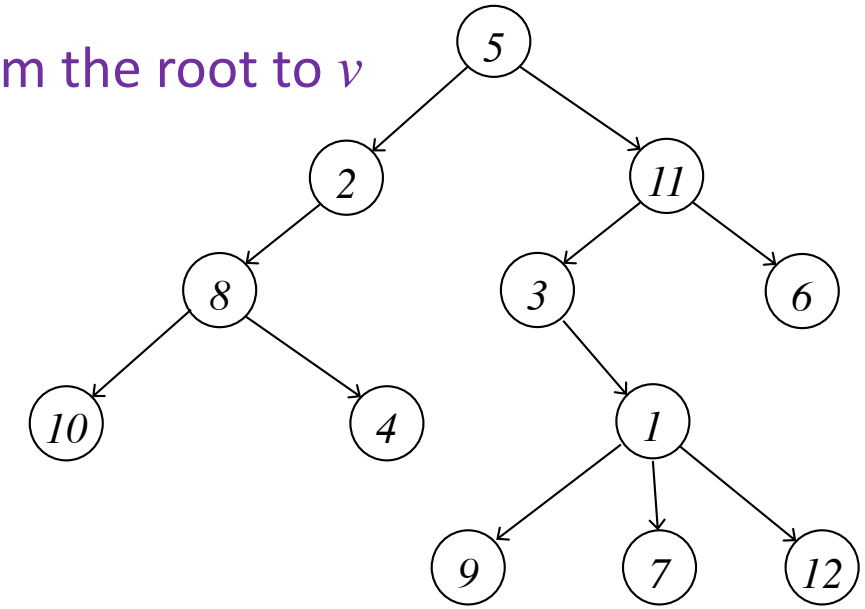


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*



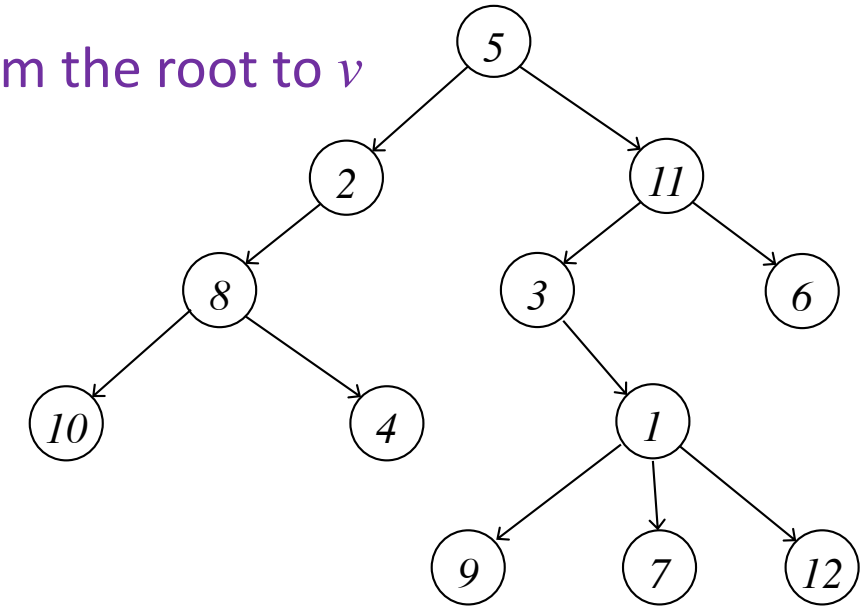
Level 0

Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*

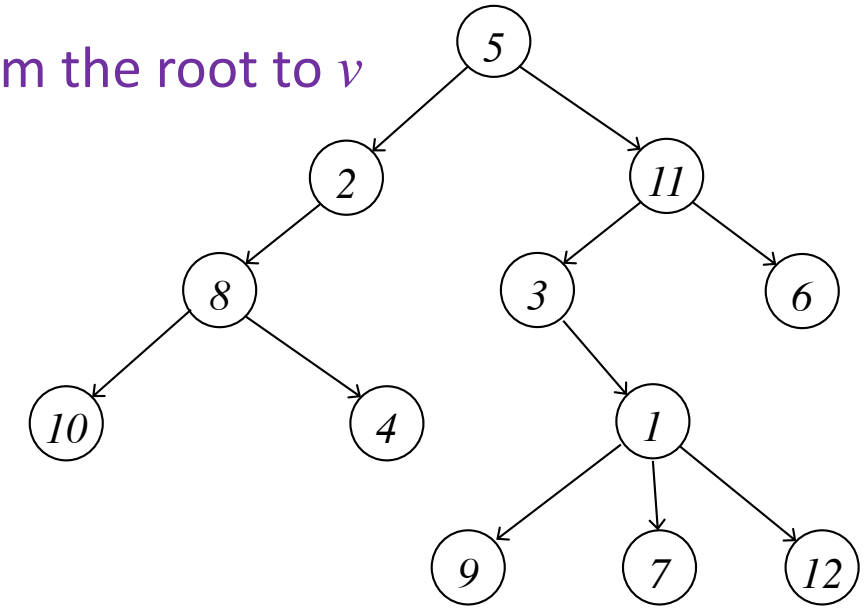


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*

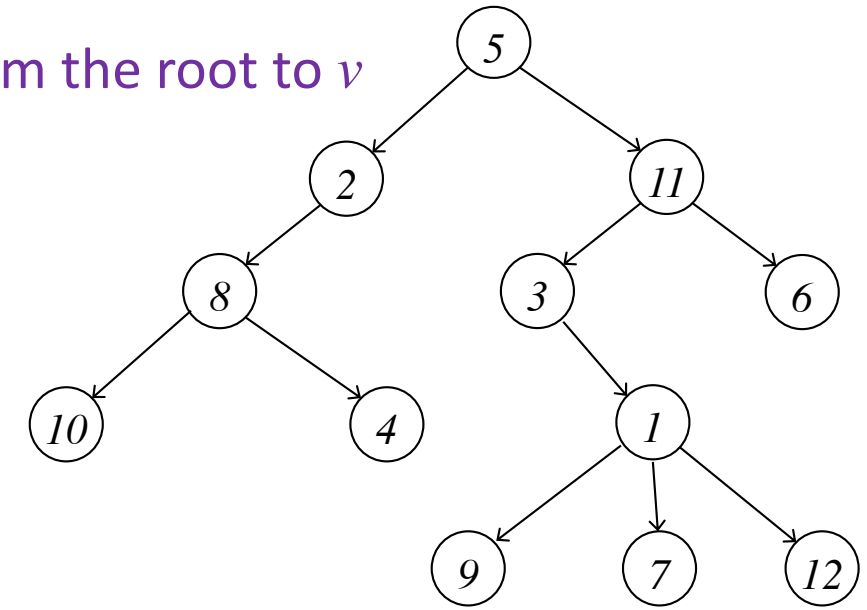


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*

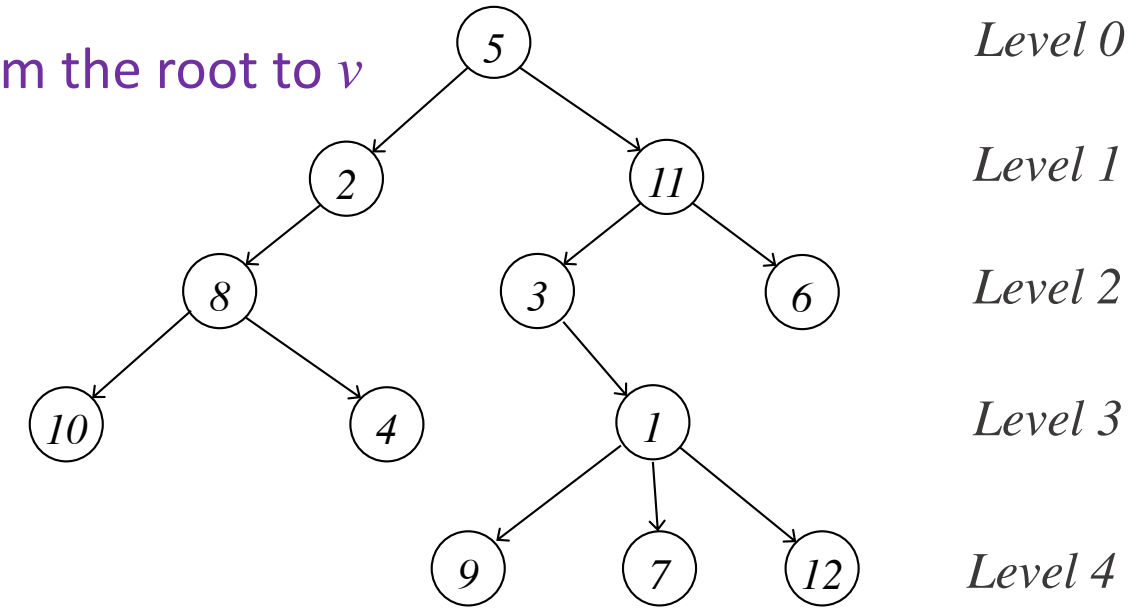


Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*



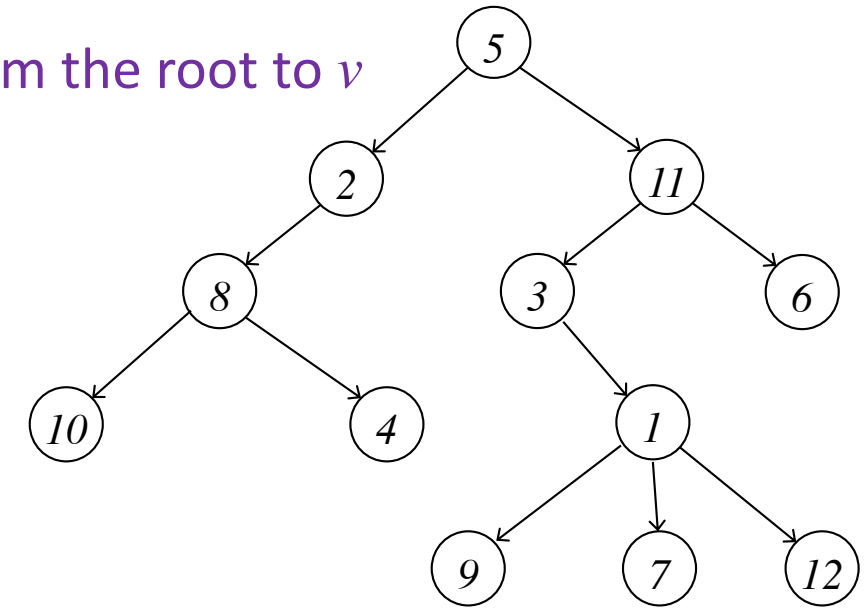
Height of a tree:

Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*



Height of a tree:

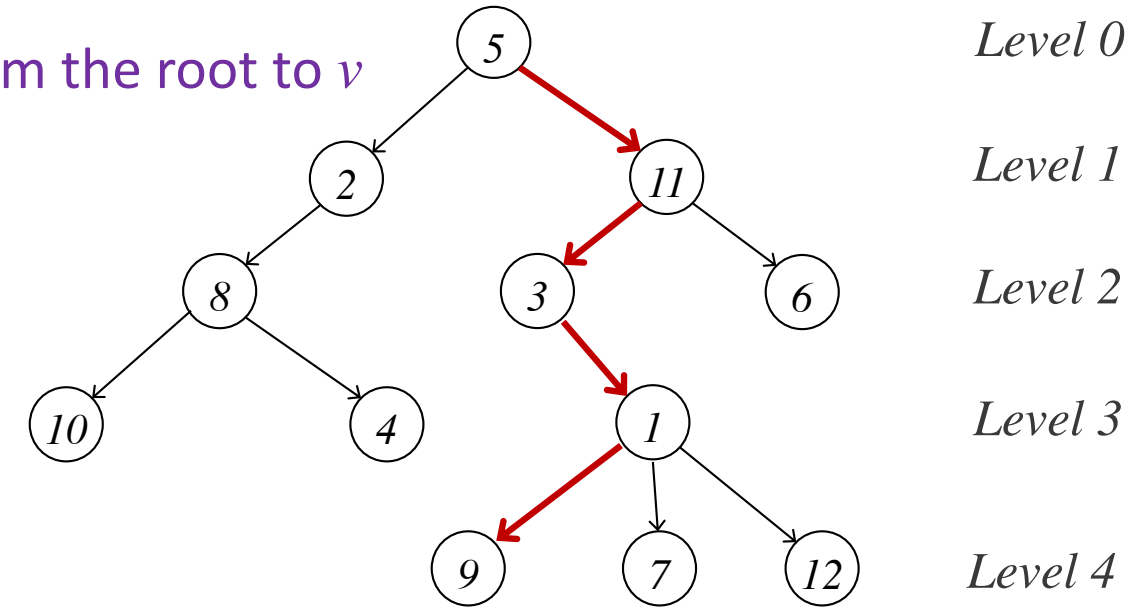
height(T) is the length of a longest path in *T*.

Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*



Height of a tree:

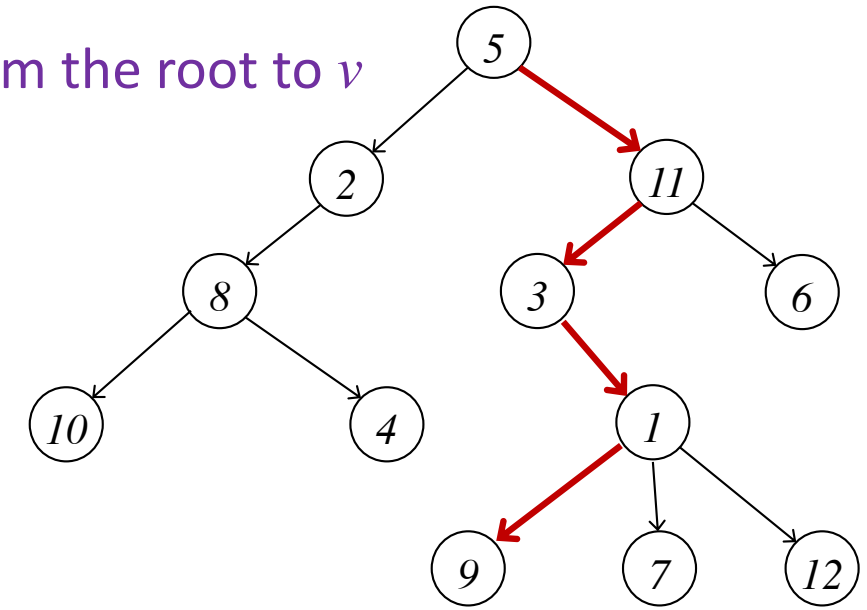
height(T) is the length of a longest path in *T*.

Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*



Height of a tree:

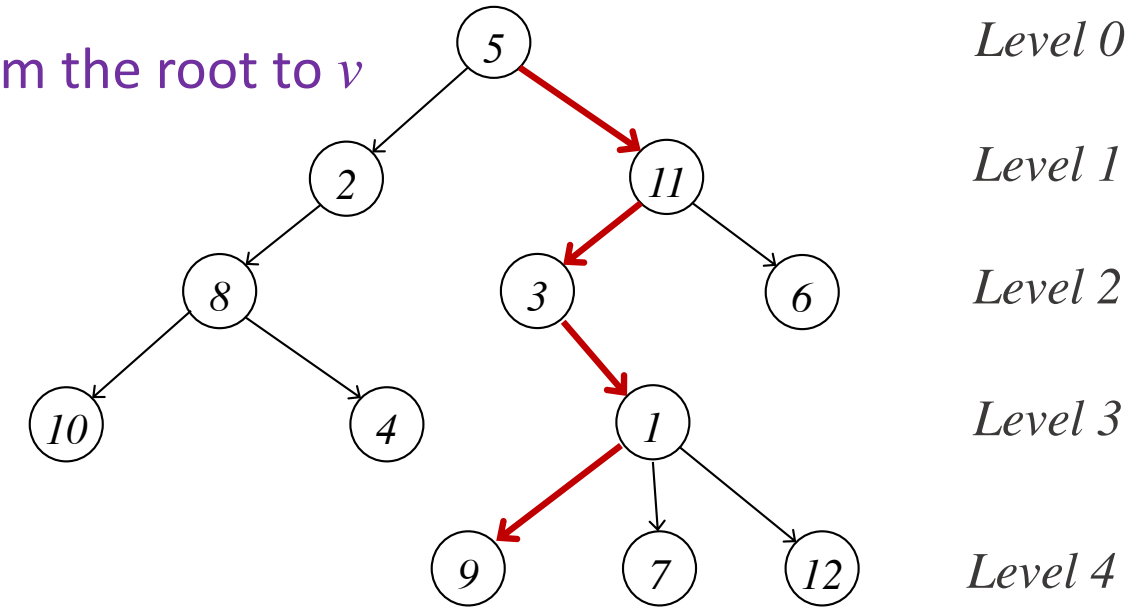
height(T) is the length of a longest path in *T*.

Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*



Height of a tree:

height(T) is the length of a longest path in *T*.

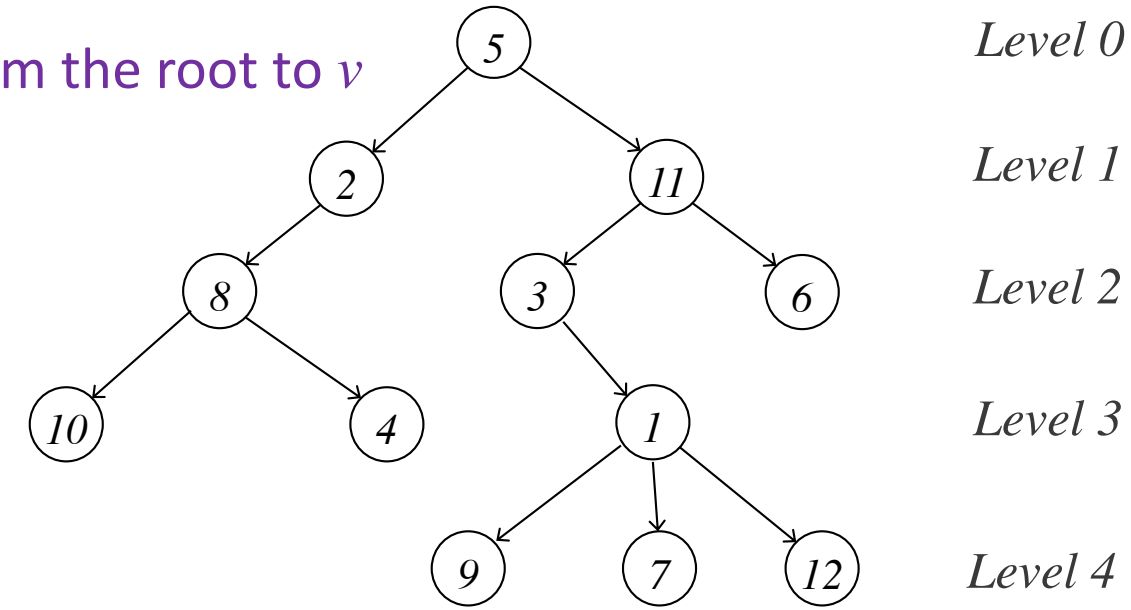
Example: *height(T) = 4*

Definitions

Depth of a node

depth(v) is the length of the path from the root to *v*

Example: *depth(1) = 3*



Height of a tree:

height(T) is the length of a longest path in *T*.

Example: *height(T) = 4*

Definitions

Definitions

Binary Tree

Definitions

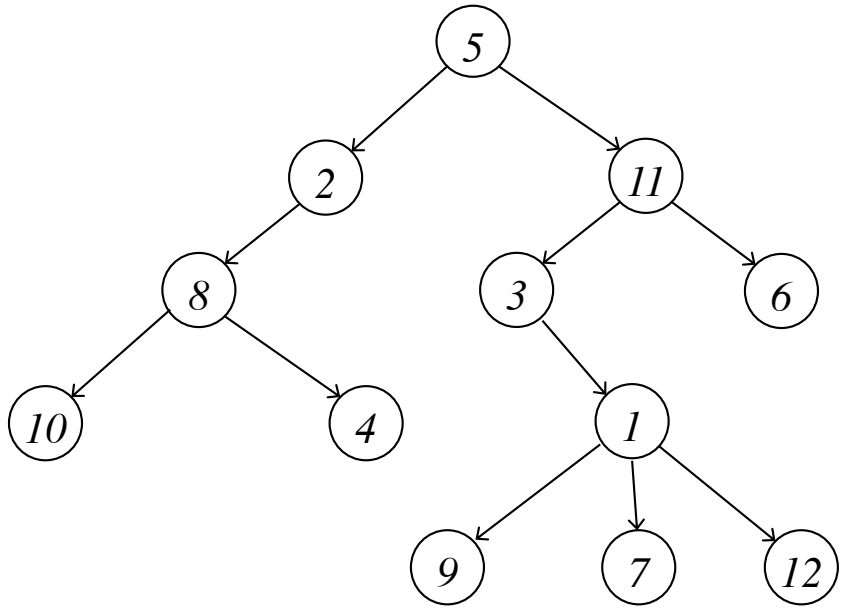
Binary Tree

A tree T is a *binary tree*, if the number of children of each node in T is ≤ 2

Definitions

Binary Tree

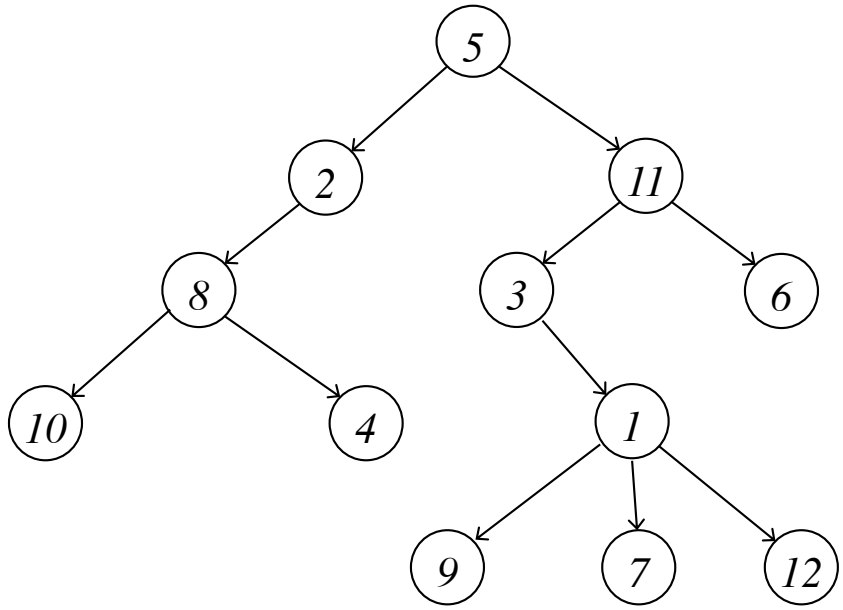
A tree T is a *binary tree*, if the number of children of each node in T is ≤ 2



Definitions

Binary Tree

A tree T is a *binary tree*, if the number of children of each node in T is ≤ 2

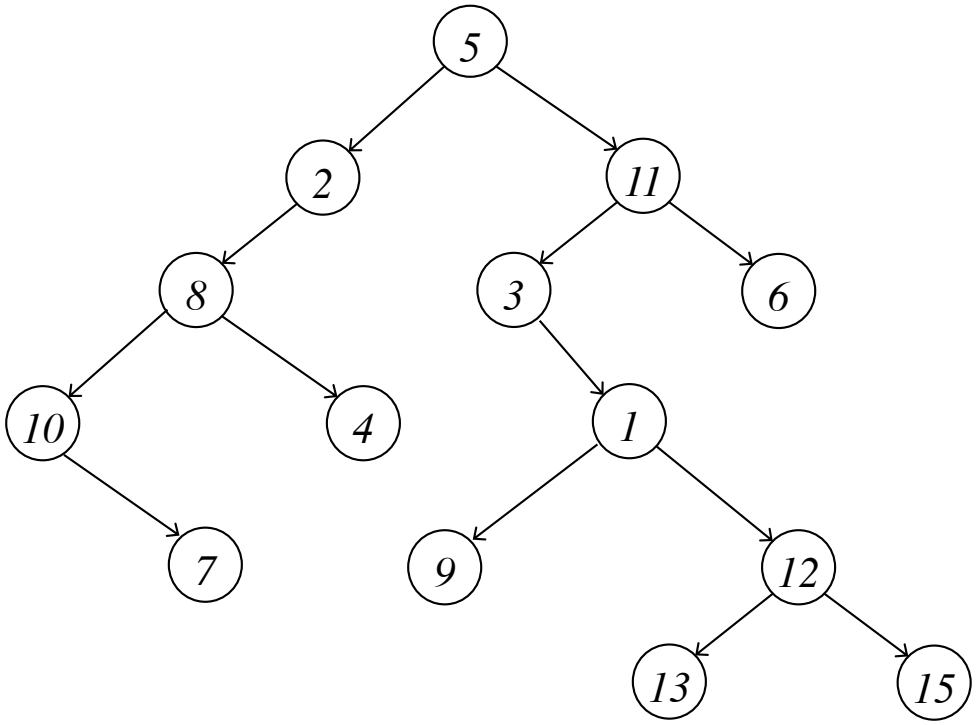


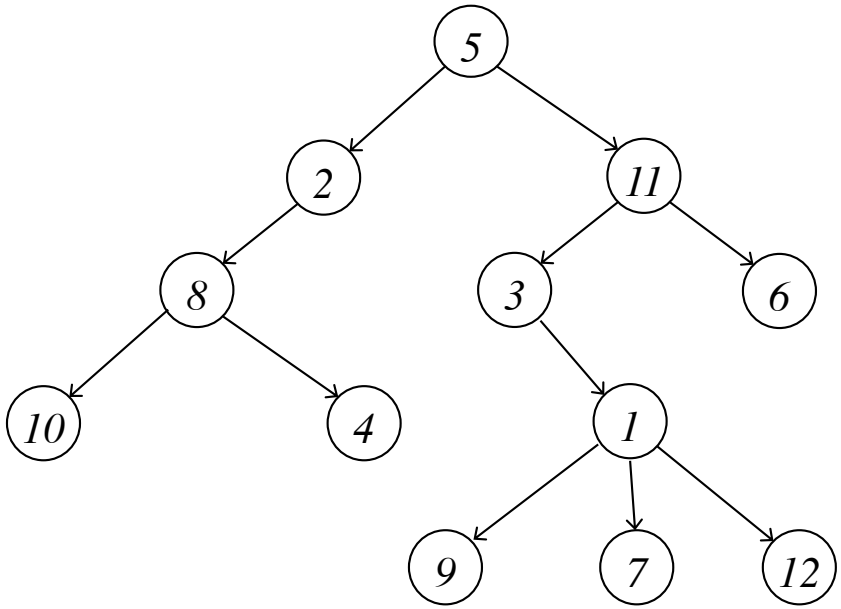
Not a Binary Tree

Definitions

Binary Tree

A tree T is a *binary tree*, if the number of children of each node in T is ≤ 2



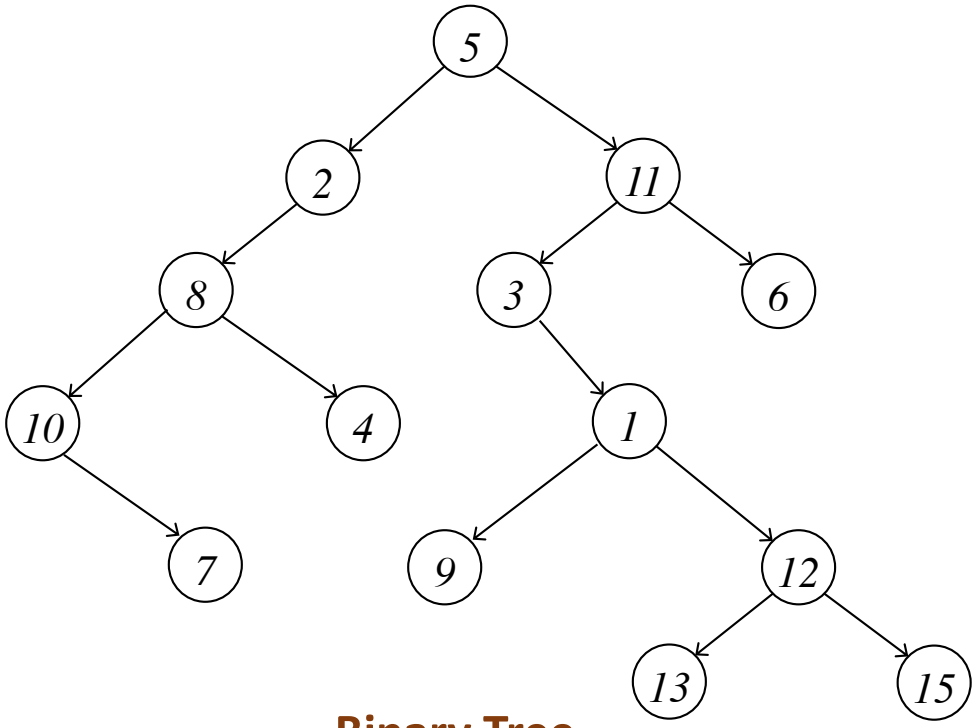


Not a Binary Tree

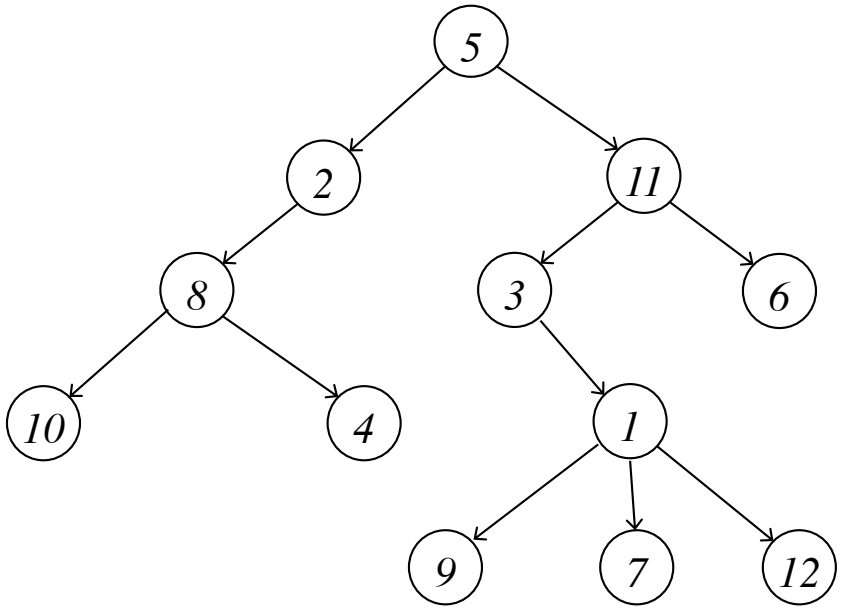
Definitions

Binary Tree

A tree T is a *binary tree*, if the number of children of each node in T is ≤ 2



Binary Tree



Not a Binary Tree

Definitions

Definitions

Definitions

Full (Proper) Binary Tree

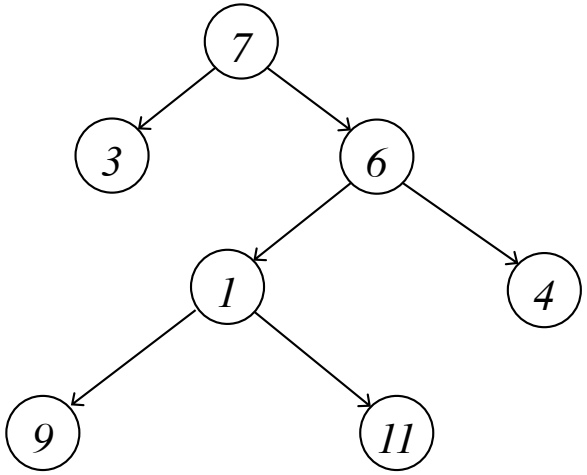
Definitions

Full (Proper) Binary Tree

Complete Binary Tree

Definitions

Full (Proper) Binary Tree

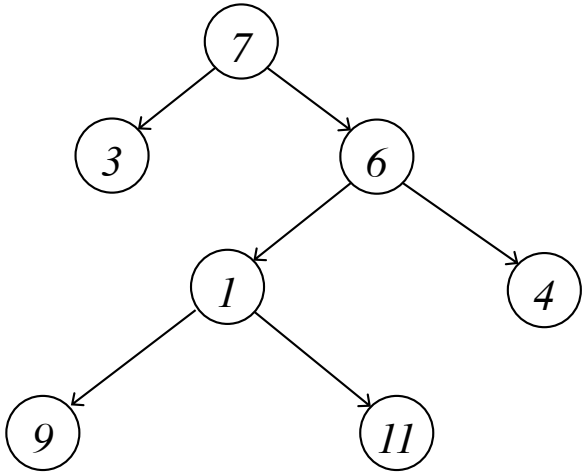


Complete Binary Tree

Definitions

Full (Proper) Binary Tree

A binary tree T is a *full binary tree*, if the number of children of each node in T is either 2 or 0

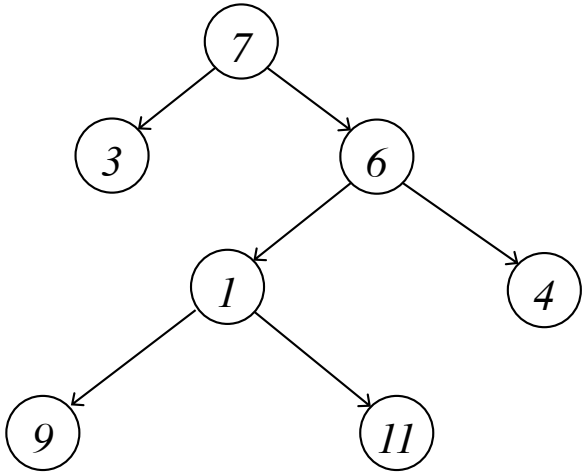


Complete Binary Tree

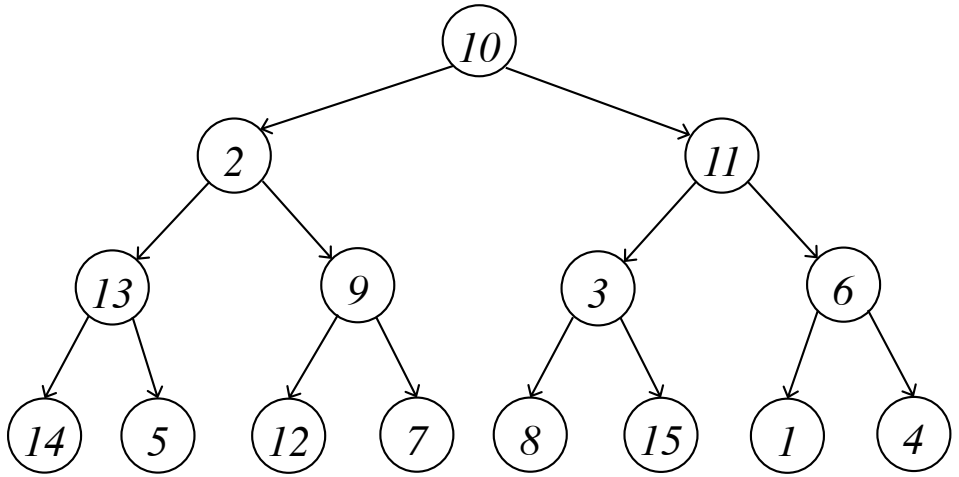
Definitions

Full (Proper) Binary Tree

A binary tree T is a *full binary tree*, if the number of children of each node in T is either 2 or 0



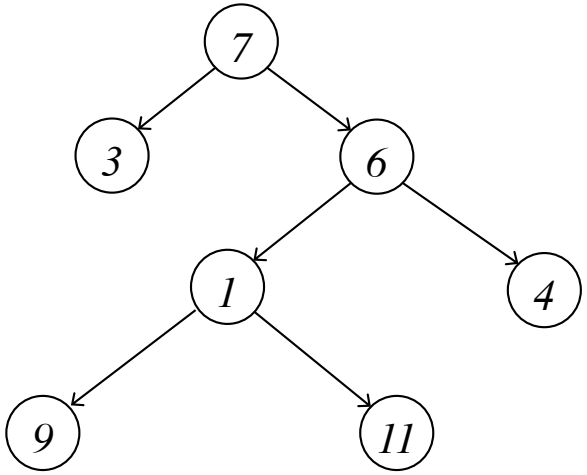
Complete Binary Tree

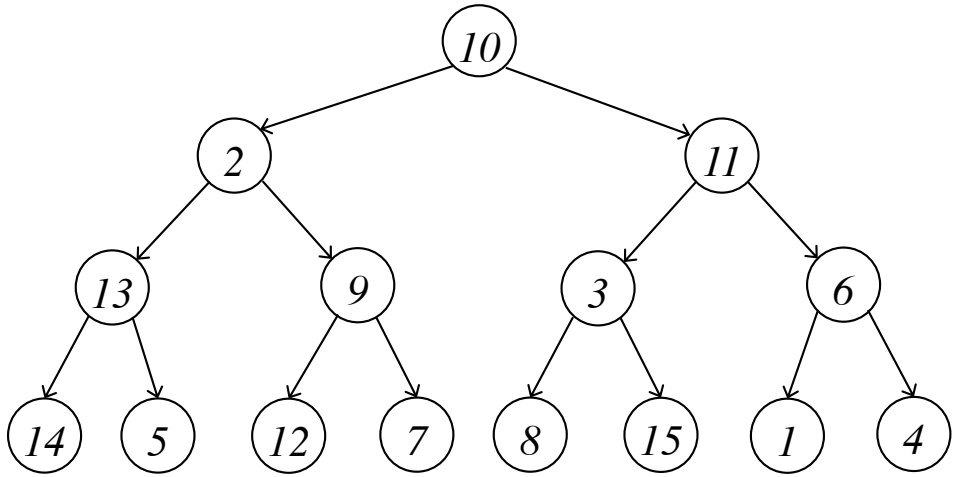


Definitions

Full (Proper) Binary Tree

A binary tree T is a *full binary tree*, if the number of children of each node in T is either 2 or 0



Complete Binary Tree

A binary tree T is a *complete binary tree*, if all the levels of T contain all possible nodes