

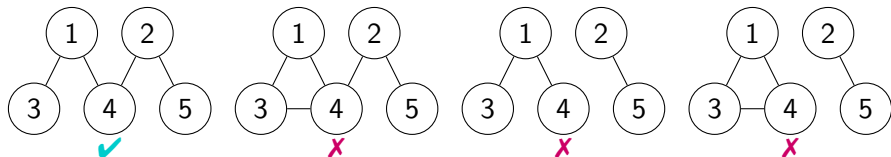
Tree

Arnaud Malapert, Gilles Menez, Marie Pelleau

Master Informatique, Université Côte d'Azur

Tree

In graph theory, a tree is an **undirected**, **acyclic**, **connected** graph



Tree $\implies n$ vertices, $n - 1$ edges

Search

- Breadth-first search
- Depth-first search
 - Pre-order
 - In-order
 - Post-order

Depth-first search

For binary trees

Three steps:

- (L) Visit the left sub-tree
- (R) Visit the right sub-tree
- (N) Visit the node

Pre-order(v) [NLR]

display v

Pre-order(left child of v)

Pre-order(right child of v)

Depth-first search

For binary trees

Three steps:

- (L) Visit the left sub-tree
- (R) Visit the right sub-tree
- (N) Visit the node

In-order(v) [LNR]

```
In-order(left child of  $v$ )  
display  $v$   
In-order(right child of  $v$ )
```

Depth-first search

For binary trees

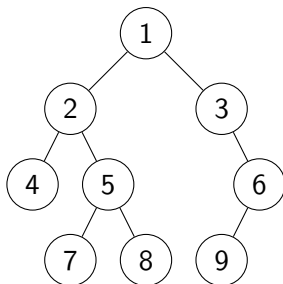
Three steps:

- (L) Visit the left sub-tree
- (R) Visit the right sub-tree
- (N) Visit the node

Post-order(v) [LRN]

```
Post-order(left child of  $v$ )
Post-order(right child of  $v$ )
display  $v$ 
```

Depth-first search



Pre-order 1 2 4 5 7 8 3 6 9

In-order 4 2 7 5 8 1 3 9 6

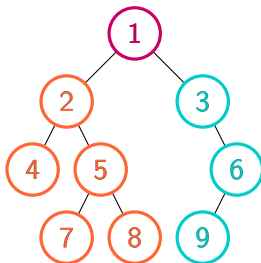
Post-order 4 7 8 5 2 9 6 3 1

Depth-first search

Given two traversals can a tree be retrieved?

- Pre-order and In-order ✓
- Post-order and In-order
- Pre-order and Post-order

Pre-order 1 2 4 5 7 8 3 6 9
In-order 4 2 7 5 8 1 3 9 6

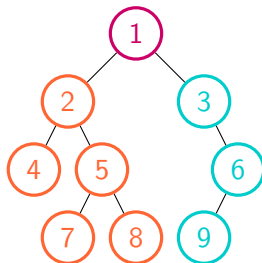


Depth-first search

Given two traversals can a tree be retrieved?

- Pre-order and In-order ✓
- Post-order and In-order ✓
- Pre-order and Post-order

Post-order 4 7 8 5 2 9 6 3 **1**
In-order 4 2 7 5 8 **1** 3 9 6

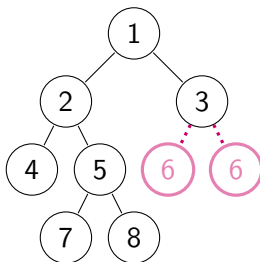


Depth-first search

Given two traversals can a tree be retrieved?

- Pre-order and In-order ✓
- Post-order and In-order ✓
- Pre-order and Post-order ✗

Pre-order 1 2 4 5 7 8 3 6 9
Post-order 4 7 8 5 2 9 6 3 1



Exercise 1: Is it a tree

Statement

Given N the number of nodes, M the number of edges and the list of edges, check if an unweighted, undirected graph is a tree

Example

Input:

3 2

1 2

2 3

Output:

YES

Exercise 1: Is it a tree

Statement

Given N the number of nodes, M the number of edges and the list of edges, check if an unweighted, undirected graph is a tree

What problems can arise?

- What do we know of N ?
- Of M ?

Exercise 1: Is it a tree

Solution 1: Build the graph

Build the graph with the list of edges

Check using BFS that all the nodes are visited once

Solution 2: Check on the list

if it can be a tree then

Maintain a visit array

Check that all the nodes are visited exactly once

Exercise 1: Is it a tree

More test cases

Input:

5 4

1 3

1 4

4 2

2 5

Output:

YES

Input:

5 5

1 3

1 4

4 2

2 5

3 4

Output:

NO

Input:

5 3

1 3

1 4

2 5

Output:

NO

Input:

5 4

1 3

1 4

2 5

3 4

Output:

NO

Exercise 2: Tree order

Statement

Given pre-order, post-order, and in-order traversals, determine if they can be of the same binary tree

Example

Input:

6

1 2 4 5 3 6

4 5 2 6 3 1

4 2 5 1 3 6

Output:

yes

Exercise 2: Tree order

Solution 1: Build the tree

Given two traversals build the tree

Generate the third traversal

Check that it matches the given one

Solution 2: Check the orders

Check the three traversals all at once

Exercise 2: Tree order

More test cases

Input:

9

1 2 4 5 7 8 3 6 9

4 7 8 5 2 9 6 3 1

4 2 7 5 8 1 3 9 6

Output:

yes

Input:

9

1 2 4 5 7 8 3 6 9

4 7 5 8 2 9 6 3 1

4 2 7 5 8 1 3 9 6

Output:

no