Content:

- Trees
  - intro
- Binary tree
  - Properties

Lect. PhD. Lupsa Dana
Babes - Bolyai University
Computer Science and Mathematics Faculty
2022 - 2023

Trees

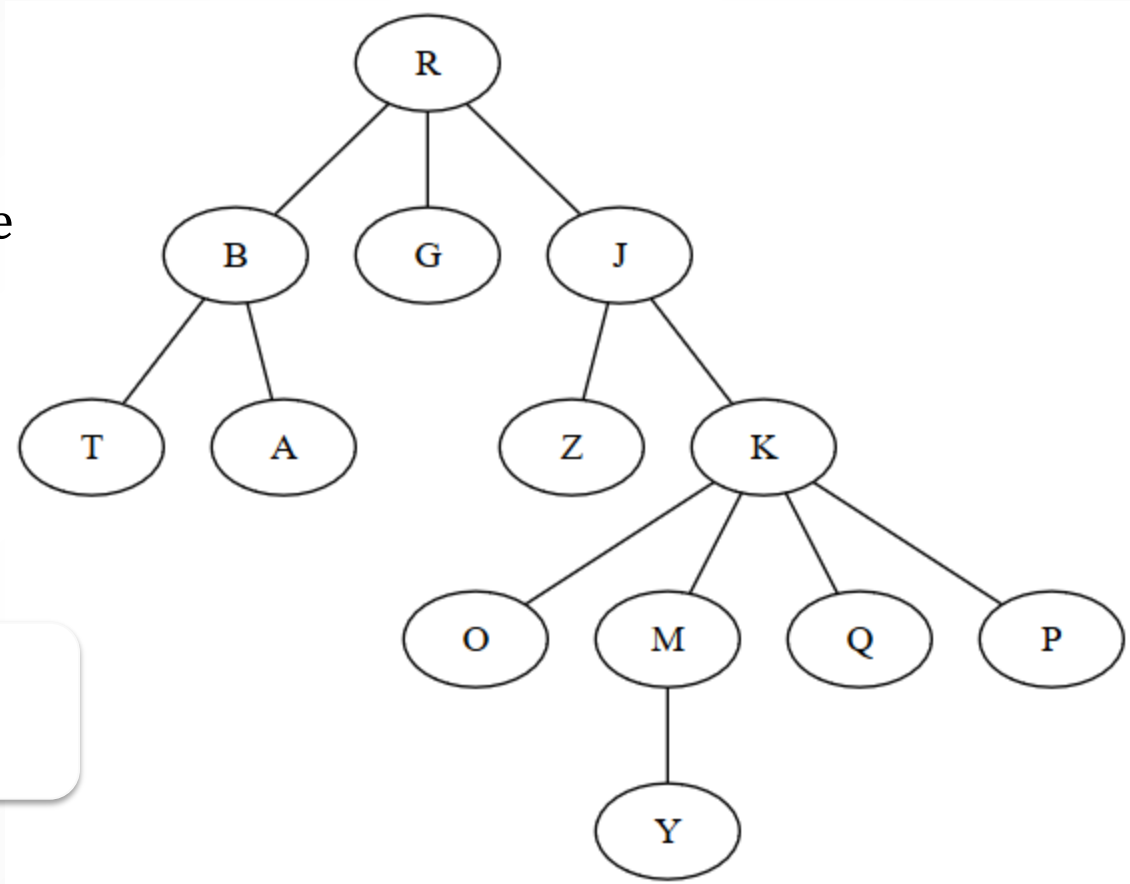
A tree is a finite set T of 0 or more nodes, with the following properties:

- If T is empty, then the tree is empty
- If T is not empty then:
  - There is a special node, R, called the root of the tree
  - The rest of the nodes are divided into k ($k \geq 0$) disjunct trees, T1, T2, ..., Tk The trees T1, T2, ..., Tk are called the subtrees (children) of R, and R is called the parent of the subtrees.

Trees - - terminology

- Rooted tree, ordered tree

- Root, child, parent;
- leaf nodes, internal node;
- The degree of a node
- The depth or level of a node
- The height of a node, the height of the tree



- Depth of the root of the tree ?
- Height of a leaf?

Trees - terminology

- Root, child, parent;

Ordered tree: a tree in which the order of the children is well defined and relevant (instead of having a set of children, each node has a list of children).

The degree of a node is defined as the number of children of the node.

- The nodes with the degree 0 (nodes without children) are called **leaf** nodes.

The depth or level of a node is the length of the path (measured as the number of edges traversed) from the root to the node.

- This path is unique.
- The root of the tree is at level 0 (and has depth 0).

The height of a node is the length of the longest path from the node to a leaf node.

- The height of the tree is defined as the height of the root node, i.e., the length of the longest path from the root to a leaf.

The nodes that are not leaf nodes are called **internal** nodes.

Binary trees

Binary tree: a tree in which each node has at most two children.

- In a binary tree we call the children of a node the left child and right child.
- Even if a node has only one child, we still have to know whether that is the left or the right one.

- A binary tree is called **full** if every internal node has exactly two children.
- A binary tree is called **complete** if all leaves are one the same level and all internal nodes have exactly 2 children.
- A binary tree is called **almost complete** if it is a complete binary tree except for the last level, where nodes are completed from left to right (binary heap - structure).
- A binary tree is called **balanced** if the difference between the height of the left and right subtrees of a node is at most 1.

Binary tree - properties

- A binary tree with N nodes has exactly N - 1 edges
  (this is true for every tree, not just binary trees)

- The number of nodes in a complete binary tree of height ***h*** is $2^{h+1} - 1$ (it is $1 + 2 + 4 + 8 + \ldots + 2^{h}$)
- The maximum number of nodes in a binary tree of height ***h*** is $2^{h+1} - 1$, if the tree is complete.
- The minimum number of nodes in a binary tree of height ***h*** is ***h*** + 1 , if the tree is degenerate.
- A binary tree with N nodes has a height
  between $\log_2 N$ and N - 1.

Binary tree - representation

- Representation using an array (see binary heap)

  Disadvantage:
  depending on the form of the tree, we might waste a lot of space.

- Linked representation
  - with dynamic allocation
  - on an array

BTNode:
info: TElem
left: ↑ BTNode
right: ↑ BTNode

BinaryTree:
root: ↑ BTNode