TREES

September 17 and 19, 2024

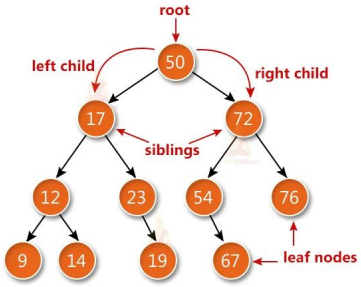
Aula 08 e 09

01. What is a Tree in programming?

Trees are well-known as a non-linear data structure. They don't store data in a linear way. They organize data hierarchically.

02. Terminology summary

- Root is the topmost node of the tree
- Edge is the link between two nodes
- Child is a node that has a parent node
- Parent is a node that has an edge to a child node
- Siblings have the same parent
- Leaf is a node that does not have a child node in the tree



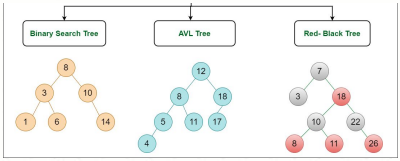
03. Properties of Tree Data Structure:

- Number of edges: An edge can be defined as the connection between two nodes. If a tree has N nodes then it will have (N-1) edges.
- Depth of a node: The depth of a node is defined as the length of the path from the root to that node.
- Height of a node: The height of a node can be defined as the length of the longest path from the node to a leaf node of the tree.
- Height of the Tree: The height of a tree is the length of the longest path from the root of the tree to a leaf node of the tree.
- Degree of a Node: The total count of subtrees attached to that node is called the degree of the node. The degree of a leaf node must be 0.
- The degree of a tree is the maximum degree of a node among all the nodes in the tree.

DEFINITION

COMMON TYPES

- Binary Search Tree (BST): A binary tree in which the left child contains values less than the parent, and the right child contains values greater.
  - Time complexity
    Average O(log n) Worst case O(n)
  - Space complexity
    Worst case O(n)
- AVL Tree: A self-balancing BST that maintains a height difference of no more than 1 between child nodes. This guarantees O(log n) time complexity for search, insertion, and deletion operations.
  - Time complexity
    Worst case O(log n)
  - Space complexity
    Worst case O(n)
- Red-Black Tree: Another self-balancing BST that uses color properties (red and black) to maintain balance. It also achieves O(log n) time complexity for search, insertion, and deletion operations.
  - Time complexity
    Worst case O(log n)
  - Space complexity
    Worst case O(n)
- Binary Tree: A tree in which each node can have at most two children.
  - Time complexity
    Worst case O(n)
  - Space complexity
    Worst case O(1)



TREE TRANSVERSAL

Tree traversal involves searching a tree data structure one node at a time, performing functions like checking the node for data or updating the node. There are two common classifications for tree traversal algorithms:

- **Depth-first search (DFS)** starts with the root node and first visits all nodes on one branch before backtracking.
- **Breadth-first search (BFS)** starts from the root node and visits all nodes at its current depth before moving to the next depth in the tree.