HANDSONE1

Alessandra
De Lucrezia

Tree Traversals in Rust

This file will give the reasons for the implementation choices to solve the assignment.

First exercise
The goal of the *is_bst* function is to test whether a binary tree is represented by *self*. *nodes* is a binary search tree (BST). A binary search tree is defined by the following property: for each node in the tree, all nodes in its left subtree must have key values less than or equal to that of the node, while all nodes in its right subtree must have key values greater than or equal to that of the node.

The algorithm checks if the tree is empty and returns true, if it is not empty it continues recursively if one of the two conditions is verified "*node.key* < *min_key* || *node.key* > *max_key*" is not a BST return false, otherwise, it continues and at the end when all the nodes to be analyzed have finished it returns *true*.

Complexity O(n)

Second exercise
The *is_balanced* function checks whether a binary tree is balanced.
A binary tree is considered balanced if, for each of its nodes, the heights of its left and right subtrees differ by at most one.

The algorithm uses a recursive visit to calculate the height of the subtrees to verify this property. If an imbalance is detected at any point during the visit, the balanced variable is set to false. Ultimately, the function returns true if the tree is balanced and false otherwise.

Complexity O(n)

Third exercise
The *is_maxheap* function checks whether the provided binary tree is a max-heap.
A max-heap is a type of complete binary tree in which every node satisfies the max-heap property. The max-heap property states that the key value of each node is greater than or equal to the key values of its children.

In this algorithm, in addition to verifying the max heap property, it is also verified that the tree is complete using an auxiliary function that returns true if the tree is complete, and false otherwise.

Complexity O(n) + O(n) = O(2n) = O(n)

Others Considerations

The logic of the tests already provided was followed by looking at their implementations, tests were performed to verify the correctness of the implemented methods by testing them with various values.