Unit 5: Trees.

Linear data structures are very useful when representing data in our programs, however, they are not useful to represent data that are related in a hierarchical way.

The lesson begins some of the main concepts of a tree: root, degree of a tree, internal and external nodes, depth, height, etc. We also study the main algorithms to traverse this structure.

Next, the binary search tree structure is presented. Binary search trees are more efficient structures than linear structures when storing information. Whereas in the lists the complexity when looking for, eliminating or deleting an element (in any position) and in the worst case, is linear, in a tree it is logarithmic. A binary search tree can degenerate into a list (for example, if you insert a sorted sequence of elements), and therefore, the complexity of its operations would become from being logarithmic to linear.

To avoid this problem, we also study how to balance a tree in size and in height.

At the end of this lesson, students should be able to:

- Understand the main properties of a tree.
- Define and implement the data structure for representing a binary tree.
- Implement the main traversal algorithms for a binary tree, such as preorder, postorder, inorder and by levels.
- Define and implement the data structure for representing a binary search tree.
- Know and apply the main algorithms to balance a binary search tree in size and in height.