TAD BinarySearchTree
BinarySearchTree = Node root, weight, height
{inv: left<raíz & right>raíz}
Operaciones principales: •   FindNode (Analyzer): Node x Value ---> Node   •   InsertNode (Modifier): Value ----> Tree   •   DeleteNode (Modifier): Value ----> Tree   •   CreateTree (Constructor):   ----> Tree   •   ShowTree (Analyzer): ----> Text
DeleteNode (Value)   "Delete a node from de binary search tree"   {pre: A value}   {post: Modified tree}
FindNode (node, value)   "Find a node in the tree that haves the value who we are looking for"   {pre: A value}   {post: Founded node}
CreateTree()   "Create a new empty tree"   {pre: TRUE}   {post: Empty tree}
ShowTree()   "Return all the nodes "   {pre: TRUE}   {post: árbol mostrado}

TAD AVL Tree
AVLTree = Node root, weight, height
{inv: left<raíz & right>raíz}
Operaciones principales: •   FindNode (Analyzer): Node x Value ---> Node •   InsertNode (Modifier): Value ----> Tree •   DeleteNode (Modifier): Value ----> Tree •   CreateTree (Constructor):   ----> Tree •   ShowTree (Analyzer): ----> Text
DeleteNode (Value) "Delete a node from de AVLTree " {pre: A value} {post: Modified tree}
FindNode (node, value) "Find a node in the tree that haves the value who we are looking for" {pre: A value} {post: Founded node}
CreateTree() "Create a new empty tree" {pre: TRUE} {post: Empty tree}
ShowTree() "Return all the nodes " {pre: TRUE} {post: árbol mostrado}
RightRotate() "Rotate the node to the right" {pre: The tree with the node that we want to rotate} {post: The tree with the node rotated}
LeftRotate() "Rotate the node to the left" {pre: The tree with the node that we want to rotate} {post: The tree with the node rotated}