UML-диаграмма Binary_Tree

BinaryTreeNode

- + key: int + data: int*
- + left_child: BinaryTreeNode*
- + right_child: BinaryTreeNode*
- + BinaryTreeNode(key: int, data: int*):
 - «constructor»

BinaryTree

- + root: BinaryTreeNode*
- + BinaryTree(): «constructor»
- + PRINT(root: BinaryTreeNode*): void
- + ADD(key: int, data: int*): int
- + FIND(root: BinaryTreeNode*, key: int): BinaryTreeNode*
- + REMOVE(root: BinaryTreeNode*, key: int): BinaryTreeNode*
- parent_node(root: BinaryTreeNode*, key: int): BinaryTreeNode*
- min_node(root: BinaryTreeNode*): BinaryTreeNode*
- max_node(root: BinaryTreeNode*): BinaryTreeNode*
- find_parent_for_new_node(root: BinaryTreeNode*, key: int):
 BinaryTreeNode*

UML-диаграмма B_Tree

B_Tree

- + root: B_Node*
- + B_Tree(): «constructor»
- + PRINT(root: B_Node*): void
- + ADD(key: int): void
- + FIND(key: int, root: B_Node*): B_Node*
- + REMOVE(key: int): B_Node*
- simple_insert(key: int, node: B_Node*): void
- sort(node: B_Node*): void*
- restruct(node: B_Node*): B_Node*
- find_parent (node: B_Node*, root: B_Node*): B_Node*