Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра проектирования информационно-компьютерных систем

Отчёт

по лабораторной работе №3

на тему:

**ФУНКЦИОНАЛЬНОЕ ПРОГРАММИРОВАНИЕ И ЛЯМБДА-ВЫРАЖЕНИЯ**

Проверил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ф.В. Усенко

(подпись)

Выполнил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Кушнер

(подпись) гр. 214301

Минск, 2024

**Цель:** Изучить функциональное программирование и лямбда-выражения.Написать программу, которая использует саморекурсивные лямбда-выражения для обхода и обработки бинарного дерева (например, подсчет листьев, поиск максимального значения).

**Код**

data class TreeNode<T>(  
 val value: T,  
 val left: TreeNode<T>? = null,  
 val right: TreeNode<T>? = null  
)  
  
fun main() {  
 val tree = TreeNode(  
 10,  
 left = TreeNode(  
 5,  
 left = TreeNode(3),  
 right = TreeNode(7)  
 ),  
 right = TreeNode(  
 15,  
 left = TreeNode(12),  
 right = TreeNode(18)  
 )  
 )  
  
 val countLeaves: (TreeNode<Int>?) -> Int = object : (TreeNode<Int>?) -> Int {  
 override fun invoke(node: TreeNode<Int>?): Int {  
 return if (node == null) {  
 0  
 } else if (node.left == null && node.right == null) {  
 1  
 } else {  
 invoke(node.left) + invoke(node.right)  
 }  
 }  
 }  
  
 val findMax: (TreeNode<Int>?) -> Int = object : (TreeNode<Int>?) -> Int {  
 override fun invoke(node: TreeNode<Int>?): Int {  
 return if (node == null) {  
 Int.MIN\_VALUE  
 } else {  
 val leftMax = invoke(node.left)  
 val rightMax = invoke(node.right)  
 maxOf(node.value, leftMax, rightMax)  
 }  
 }  
 }  
  
 val leavesCount = countLeaves(tree)  
 println("Количество листьев в дереве: $leavesCount")  
   
 val maxValue = findMax(tree)  
 println("Максимальное значение в дереве: $maxValue")  
}

**Вывод**: Изучил использование функционального программирования и лямбда-выражений на практике. Написал программу, которая использует саморекурсивные лямбда-выражения для обхода и обработки бинарного дерева (например, подсчет листьев, поиск максимального значения).