Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Дисциплина: Программирование мобильных информационных систем

Отчёт

по лабораторной работе №3

на тему

**Функциональное программирование и лямбда-выражения**

Выполнил: Проверил:

ст. гр. 214302 Усенко Ф.В.

Коршов М.И.

Минск 2024

Вариант 15. Функция для анализа потоков данных: Напишите программу, которая принимает поток данных (например, из файла или сети), применяет к ним несколько лямбда-выражений для фильтрации, преобразования и агрегации, и выводит результат в реальном времени.

Листинг кода:

import java.io.File

fun main() {

val inputFileReader = File("file.txt").bufferedReader()

var lineCount = 0

var totalSum = 0

val transformSpacesToOnes = fun(line: String): String {

return line.replace(" ", "1")

}

inputFileReader.lineSequence().forEach { line ->

if (line.isNotBlank()) {

val newLine = transformSpacesToOnes(line)

var currentSum = 0

newLine.filter { char -> char.isDigit() }.forEach { digitChar -> currentSum += digitChar.digitToInt() }

totalSum += currentSum

lineCount++

println("Processed line: $newLine, Current sum: $currentSum, Line count: $lineCount")

}

}

println("Final sum of values: $totalSum")

println("Total lines processed: $lineCount")

}

Контрольные вопросы:

1. Как объявить функцию в *Kotlin*? В чем разница между обычной функцией и однострочной функцией?

В *Kotlin* функции можно объявлять с помощью ключевого слова *fun*. Есть два основных типа функций: обычные и однострочные.

Обычная функция:

fun add(a: Int, b: Int): Int {

return a + b

}

Однострочная функция:

fun add(a: Int, b: Int): Int = a + b

Основное отличие однострочной функции от обычной заключается в отсутствии *return*.

1. Что такое функция высшего порядка, и как её использовать? Приведите пример.

Функция высшего порядка – это функция, которая принимает другую функцию в качестве параметра или возвращает функцию. Это позволяет создавать гибкие и переиспользуемые компоненты.

Пример:

fun calculate(operation: (Int, Int) -> Int, a: Int, b: Int): Int {

return operation(a, b)

}

fun main() {

val sum = calculate({ x, y -> x + y }, 5, 3) // 8

val product = calculate({ x, y -> x \* y }, 5, 3) // 15

println("Sum: $sum, Product: $product")

}

1. Как передать функцию в качестве параметра другой функции? Приведите пример.

Ниже представлен пример, где функция *calculate* принимает другую функцию в качестве параметра.

Пример:

fun operateOnNumbers(a: Int, b: Int, operation: (Int, Int) -> Int): Int {

return operation(a, b)

}

fun main() {

val result = operateOnNumbers(10, 5, { x, y -> x - y }) // 5

println("Result: $result")

}