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 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ П.М. Василевич

(подпись) гр. 214302

Минск, 2024

**Задание:** Создайте программу, которая динамически генерирует функции на основе пользовательского ввода (например, определение функции для нахождения максимума из трех чисел) и затем использует их для обработки данных.

**Решение**

fun main() {

val numbers = mutableListOf<Int>()

var length = (6..16).random()

for (x in 1..length) {

var i = (-20..20).random()

numbers += i;

}

println("\nНачальный массив:")

for (i in numbers) {

print("$i\t")

}

var number: Int? = null

do {

println(

"\n\tВыберите действие: \n1 - Найти максимальный элемент\n" +

"2 - Найти сумму элементов\n3 - Выбрать случайный элемент" +

"\n4 - Посчитать количество элементов\n5 - Найти среднее арифметическое элементов" +

"\n0 - ВЫЙТИ"

)

while (true) {

print("Введите число: ")

number = (readLine()?.trim(' '))?.toIntOrNull()

if (number != null) {

break

} else println("Введенное значение не является числом")

}

val operation: (List<Int>) -> Int

operation = when (number) {

1 ->

{ nums ->

nums.max()

}

2 ->

{ nums -> nums.sum() }

3 ->

{ nums -> nums.random() }

4 ->

{ nums -> nums.count() }

5 ->

{ nums -> nums.sum() / nums.size }

else -> return

}

val maxNumber = operation(numbers)

println("Результат операции: $maxNumber")

} while (number!=0)

}

**Контролные вопросы**

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

Функции в Kotlin объявляются с использованием ключевого слова fun, после которого следует имя функции, параметры и возвращаемый тип.

Пример:

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

return a \* b

}

Если функция состоит из одного выражения, её можно объявить как однострочную, без использования return.

Пример:

fun multiply(a: Int, b: Int) = a \* b

Отличия однострочной функции от обычной: отсутствие фигурных скобок и возвращаемого типа.

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

В Kotlin функции могут передаваться как параметры другим функциям. Это важная часть функционального программирования.

Пример:

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

return operation(a, b)

}

fun main() {

val sum = operate(5, 3) { x, y -> x - y }

println(sum) // Выводит 2

}

1. Что такое анонимная функция, и как она отличается от лямбда-выражения?

Анонимные функции похожи на лямбда-выражения, но могут иметь более сложное тело, включая несколько операторов. Анонимные функции определяются с использованием ключевого слова fun без имени функции.

Пример:

val multiply = fun(a: Int, b: Int): Int {

return a + b

}

println(multiply(3, 4)) // Выводит 7