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 №2

на тему:

**«Циклы, диапазоны и массивы»**

Выполнил:

ст. гр. 214302

Кранцевич В.С.

Проверил:

Усенко Ф. В.

Минск 2024

**Цель**: Закрепить понимание работы циклов, диапазонов и массивов в Kotlin. Научиться эффективно использовать эти конструкции для решения различных задач.

**Вариант 17**: Динамическое изменение размера массива: Напишите программу, которая создает массив с динамическим изменением размера. Пользователь должен иметь возможность добавлять и удалять элементы из массива, а программа — оптимизировать его память.

**КОД ПРОГРАММЫ**

import java.util.Scanner

fun main() {

try {

val scanner = Scanner(System.`in`)

var dynamicArray = IntArray(0)

while (true) {

println("\nМеню:")

println("1. Добавить элемент")

println("2. Удалить элемент")

println("3. Показать массив")

println("4. Выйти")

print("Выберите действие: ")

when (scanner.nextInt()) {

1 -> {

print("Введите элемент для добавления: ")

val element = scanner.nextInt()

val newArray = IntArray(dynamicArray.size + 1)

for (i in dynamicArray.indices) {

newArray[i] = dynamicArray[i]

}

newArray[dynamicArray.size] = element

dynamicArray = newArray

println("Элемент $element добавлен.")

}

2 -> {

if (dynamicArray.isEmpty()) {

println("Массив пуст, удалять нечего.")

} else {

print("Введите индекс элемента для удаления: ")

val index = scanner.nextInt()

if (index in 0 until dynamicArray.size) {

val newArray = IntArray(dynamicArray.size - 1)

for (i in 0 until index) {

newArray[i] = dynamicArray[i]

}

for (i in index until newArray.size) {

newArray[i] = dynamicArray[i + 1]

}

dynamicArray = newArray

println("Элемент с индексом $index удален.")

} else {

println("Некорректный индекс.")

}

}

}

3 -> {

if (dynamicArray.isEmpty()) {

println("Массив пуст.")

} else {

println("Текущий массив: ${dynamicArray.joinToString(", ")}")

}

}

4 -> {

println("Выход из программы.")

break

}

else -> {

println("Некорректный выбор. Попробуйте снова.")

}

}

}

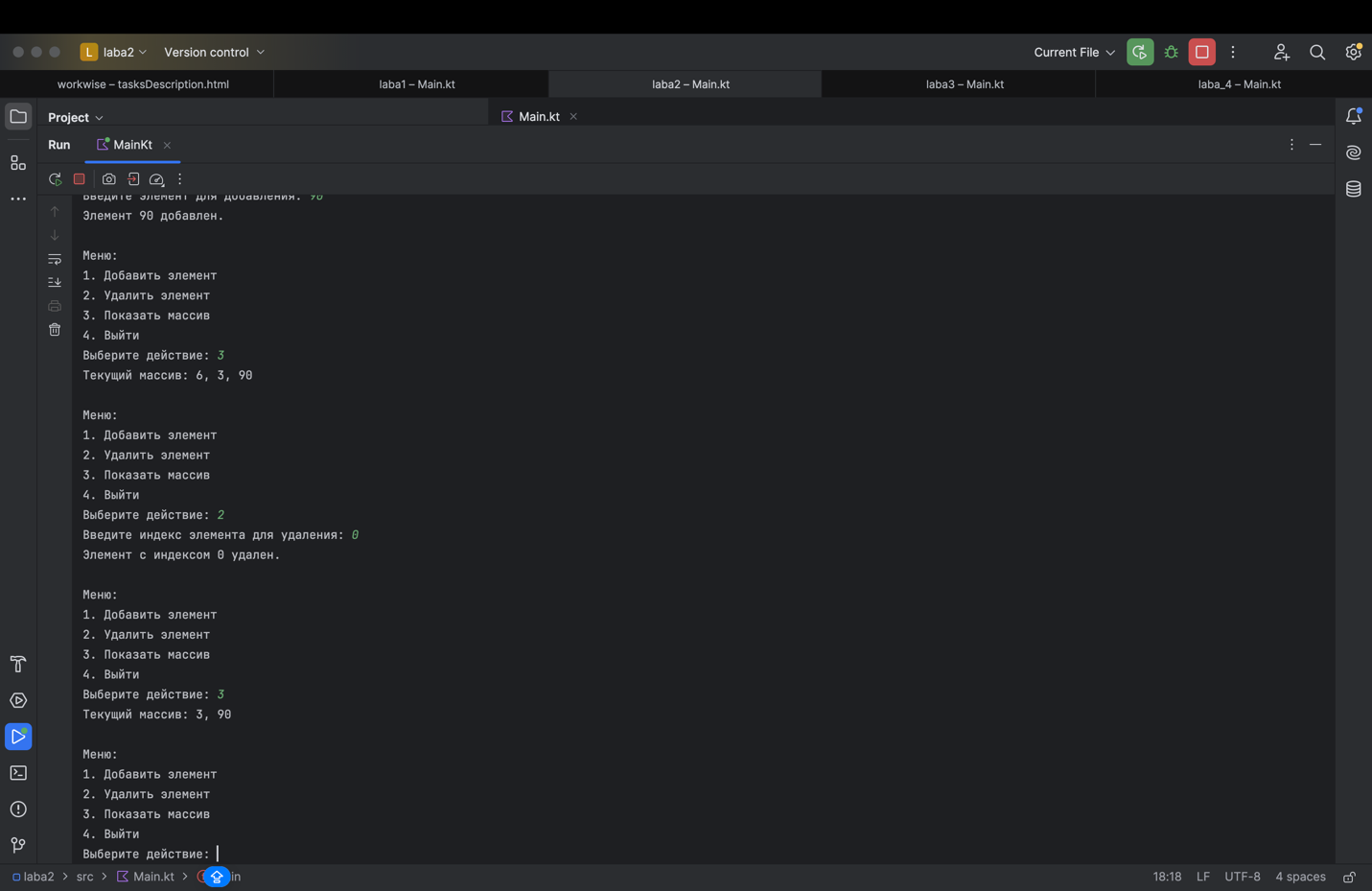
} catch (e: Exception) {

println("Ошибка ввода. Пожалуйста, введите корректные числовые значения.")

}

}

**СКРИНШОТЫ РАБОЧИХ ОКОН**



**ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Какие типы циклов существуют в Kotlin, и как они используются?

В Kotlin доступны три типа циклов:

1. for: Для итерации по диапазонам, коллекциям или массивам.
2. while: Выполняет блок кода, пока условие истинно.
3. do...while: Выполняет блок кода как минимум один раз, а затем проверяет условие.

2. Как работает цикл for для итерации по элементам массива? Приведите пример.

Цикл for перебирает элементы массива:

val numbers = arrayOf(1, 2, 3, 4, 5)

for (num in numbers) {

println(num)

}

3. Чем отличается цикл while от цикла for?

* while: Используется, когда количество итераций заранее неизвестно.
* for: Используется, когда нужно перебрать фиксированное количество элементов или диапазон.

Пример while:

var i = 0

while (i < 5) {

println(i)

i++

}

4. Какую роль выполняет цикл do...while, и в чем его отличие от while?

Цикл do...while выполняется как минимум один раз, даже если условие ложно:

var i = 5

do {

println(i)

i--

} while (i > 0)

5. Что такое диапазон в Kotlin, и как его создать?

Диапазон (range) — это последовательность значений, создаваемая с помощью .. или rangeTo().

Пример:

val range = 1..10 // Диапазон от 1 до 10 включительно

6. Как использовать оператор in для проверки принадлежности значения диапазону?

Оператор in проверяет, входит ли значение в диапазон:

val number = 5

if (number in 1..10) {

println("Число в диапазоне")

}

7. Что произойдет, если использовать оператор !in с диапазоном? Приведите пример.

Оператор !in проверяет, не принадлежит ли значение диапазону:

val number = 15

if (number !in 1..10) {

println("Число вне диапазона")

}

8. Как работать с реверсированными диапазонами в Kotlin? Приведите пример.

Для реверса диапазона используется функция reversed():

for (i in (1..5).reversed()) {

println(i) // 5, 4, 3, 2, 1

}

9. Что такое шаг диапазона, и как его задать? Приведите пример.

Шаг задаётся с помощью step:

for (i in 1..10 step 2) {

println(i) // 1, 3, 5, 7, 9

}

10. Как создать массив чисел в Kotlin? Приведите пример.

Массив создаётся с помощью функции arrayOf или IntArray:

val numbers = arrayOf(1, 2, 3, 4, 5)

val intArray = IntArray(5) { it \* 2 } // [0, 2, 4, 6, 8]

11. Как обратиться к элементу массива по индексу? Что будет, если индекс выйдет за пределы массива?

Элементы массива доступны через оператор []:

val numbers = arrayOf(10, 20, 30)

println(numbers[1]) // 20

Если индекс выходит за пределы, будет выброшено исключение ArrayIndexOutOfBoundsException.

12. Как изменить значение элемента массива в Kotlin? Приведите пример.

val numbers = arrayOf(1, 2, 3)

numbers[1] = 10

println(numbers[1]) // 10

13. Как перебрать элементы массива с использованием цикла for?

val numbers = arrayOf(1, 2, 3, 4)

for (num in numbers) {

println(num)

}

14. Что такое массив фиксированного размера, и как его создать в Kotlin?

Массив фиксированного размера — это массив, размер которого нельзя изменить после создания:

val fixedArray = Array(5) { it \* 2 } // [0, 2, 4, 6, 8]

15. Как отсортировать массив чисел в порядке возрастания? Приведите пример.

val numbers = arrayOf(5, 2, 8, 1)

numbers.sort()

println(numbers.joinToString()) // 1, 2, 5, 8

16. Как удалить элемент из массива, если известно его значение?

Для этого можно использовать функции filter или преобразование массива в список:

val numbers = arrayOf(1, 2, 3, 4)

val filtered = numbers.filter { it != 3 }.toTypedArray()

println(filtered.joinToString()) // 1, 2, 4

17. Чем полезны методы joinToString и sort для работы с массивами?

* joinToString: Преобразует массив в строку с разделителем.
* sort: Сортирует массив.

Пример:

val numbers = arrayOf(3, 1, 2)

numbers.sort()

println(numbers.joinToString(", ")) // 1, 2, 3

18. Как объединить два массива в один? Приведите пример кода.

val array1 = arrayOf(1, 2, 3)

val array2 = arrayOf(4, 5, 6)

val combined = array1 + array2

println(combined.joinToString()) // 1, 2, 3, 4, 5, 6

19. Как проверить, упорядочен ли массив по возрастанию?

val numbers = arrayOf(1, 2, 3, 4)

val isSorted = numbers.sortedArray().contentEquals(numbers)

println(isSorted) // true

20. Как подсчитать количество элементов в массиве, соответствующих заданному условию?

Используйте функцию count:

val numbers = arrayOf(1, 2, 3, 4, 5)

val count = numbers.count { it % 2 == 0 }

println(count) // 2 (четные числа: 2, 4)

**ВЫВОД**

В ходе лабораторной работы изучены работы циклов, диапазонов и массивов в *Kotlin*. А также эффективно использованы эти конструкции для решения различных задач.