

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования
Кафедра проектирования информационно-компьютерных систем

Отчёт
по лабораторной работе №2
на тему:
ЦИКЛЫ, ДИАПАЗОНЫ И МАССИВЫ

Проверил

(подпись)

Усенко Ф.В.

Выполнил

(подпись)

Ахрамович С.П.
гр. 214301

Минск, 2024

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

ВАРИАНТ 2. Сложная сортировка пузырьком с оптимизацией.

КОД ПРОГРАММЫ

```
import kotlin.random.Random
import java.util.Scanner

fun bubbleSort(arr: Array<Int>) {
    val n = arr.size
    var swapped: Boolean

    for (i in 0 until n - 1) {
        swapped = false
        for (j in 0 until n - i - 1) {
            if (arr[j] > arr[j + 1]) {
                val temp = arr[j]
                arr[j] = arr[j + 1]
                arr[j + 1] = temp
                swapped = true
            }
        }
        if (!swapped) break
    }
}

fun randomArray(size: Int, range: Int): Array<Int> {
    val arr = Array(size) { 0 }
    for (i in 0 until size) {
        arr[i] = Random.nextInt(range)
    }
    return arr
}

fun inputArrayFromKeyboard(size: Int): Array<Int> {
    val arr = Array(size) { 0 }
    val scanner = Scanner(System.`in`)
    println("Введите $size чисел:")

    for (i in 0 until size) {
        var validInput = false
        while (!validInput) {
            try {
                print("Элемент $i: ")
                arr[i] = scanner.nextInt()
                validInput = true
            } catch (e: Exception) {
                println("Некорректный ввод. Введите целое число.")
                scanner.next()
            }
        }
    }
    return arr
}

fun main() {
    val scanner = Scanner(System.`in`)
```

```

var choice: Int? = null
var size: Int? = null

while (choice == null) {
    try {
        println("Выберите способ инициализации массива: ")
        println("1. Рандомные значения")
        println("2. Ввести значения с клавиатуры")
        print("Ваш выбор (1 или 2): ")
        choice = scanner.nextInt()
        if (choice !in 1..2) {
            println("Некорректный выбор. Введите 1 или 2.")
            choice = null
        }
    } catch (e: Exception) {
        println("Ошибка ввода! Введите корректное число (1 или 2).")
        scanner.next()
    }
}

while (size == null) {
    try {
        print("Введите размер массива: ")
        size = scanner.nextInt()
        if (size <= 0) {
            println("Размер массива должен быть больше 0.")
            size = null
        }
    } catch (e: Exception) {
        println("Ошибка ввода! Введите корректное целое число.")
        scanner.next()
    }
}

val arr: Array<Int> = when (choice) {
    1 -> {
        var range: Int? = null
        while (range == null) {
            try {
                print("Введите диапазон значений для рандома (0 до n): ")
                range = scanner.nextInt()
                if (range <= 0) {
                    println("Диапазон должен быть больше 0.")
                    range = null
                }
            } catch (e: Exception) {
                println("Ошибка ввода! Введите корректное целое число.")
                scanner.next()
            }
        }
        randomArray(size, range)
    }
    2 -> inputArrayFromKeyboard(size)
    else -> throw IllegalArgumentException("Неверный выбор!")
}

println("Исходный массив: ${arr.joinToString()}")
bubbleSort(arr)
println("Отсортированный массив: ${arr.joinToString()}")
}

```

Результат работы программы представлен на рисунке 1.

```
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program File
Выберите способ инициализации массива:
1. Рандомные значения
2. Ввести значения с клавиатуры
Ваш выбор (1 или 2): 1
Введите размер массива: 7
Введите диапазон значений для рандома (0 до n): 20
Исходный массив: 17, 9, 10, 3, 13, 14, 10
Отсортированный массив: 3, 9, 10, 10, 13, 14, 17

Process finished with exit code 0
```

Рисунок 1 – Результат работы программы

КОНТРОЛЬНЫЕ ВОПРОСЫ

1 Чем отличается цикл *while* от цикла *for*?

While выполняется пока истинно условие, а *for* заданное количество раз.

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

Цикл *do...while* выполняет тело цикла хотя бы один раз, даже если условие ложно, поскольку проверка условия происходит после выполнения кода. В *while* условие проверяется до выполнения кода.

3 Что такое диапазон в *Kotlin*, и как его создать?

Диапазон в *Kotlin* — это последовательность чисел, определяемая началом и концом с помощью оператора `...` Пример:

```
val range = 1..5
```

4 Как использовать оператор *in* для проверки принадлежности значения диапазону?

Оператор *in* проверяет, принадлежит ли значение диапазону. Пример:

```
val x = 3
if (x in 1..5) {
    println("$x is in the range")
}
```

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

Оператор *!in* проверяет, не принадлежит ли значение диапазону.
Пример:

```
val x = 6
if (x !in 1..5) {
    println("$x is not in the range")
}
```

6 Как работать с реверсированными диапазонами в *Kotlin*? Приведите пример.

Реверсирование диапазона происходит с помощью функции *downTo*.
Пример:

```
for (i in 5 downTo 1) {
    println(i)
}
```

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

Шаг диапазона задается с помощью функции *step*. Пример:

```
for (i in 1..10 step 2) {
    println(i)
}
```

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

Пусть *array* – массив.

array[i] – обращение по индексу

Если индекс выйдет за пределы массива, будет ошибка.

9 Как удалить элемент из массива, если известно его значение?

Для удаления элемента из массива можно использовать функцию *removeItem()*.

removeItem(array, value), где *array* – массив, а *value* – значение удаляемого элемента.

Вывод: Закреплено понимание работы циклов, диапазонов и массивов в *Kotlin*. Научились эффективно использовать эти конструкции для решения различных задач.