

Лабораторная работа 5

Задание 1

```
fun isPrime(number: Int): Boolean {  
    if (number < 2) return false  
    for (i in 2 until number) {  
        if (number % i == 0) {  
            return false  
        }  
    }  
    return true  
}
```

```
fun isPrime(number: Int): Boolean {  
    if (number < 2) return false  
    for (i in 2 until number) {  
        if (number % i == 0) {  
            return false  
        }  
    }  
    return true  
}
```

Задание 2

```
1 fun main() {  
2     for (i in 1 ≤ .. ≤ 20) {  
3         if (i % 2 == 0) {  
4             println(i)  
5         }  
6     }  
7 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.1\lib\idea_rt.jar=51717:C:\Program Files\JetBrains\IntelliJ IDEA Community Ed  
2  
4  
6  
8  
10  
12  
14  
16  
18  
20
```

Задание 3

```
fun main() {  
    print("Введите число N: ")  
    val n = readLine()!!.toInt()  
  
    var sum = 0  
    for (i in 1..n) {  
        sum += i  
    }  
  
    println("Сумма чисел от 1 до $n равна $sum")  
}
```

```
C:\Users\Student\Downloads\openjdk-22.0.2\bin\java.exe  
2  
4  
6  
8  
10  
12  
14  
16  
18  
20
```

Задание 4

```

1  ▶ fun main() {
2      print("Введите число для вычисления факториала: ")
3      val n = readLine()!!.toInt()
4
5      val factorial = calculateFactorial(n)
6
7      println("Факториал числа $n равен $factorial")
8  }
9
10  fun calculateFactorial(n: Int): Long {
11      if (n == 0 || n == 1) {
12          return 1
13      }
14      var result: Long = 1
15      for (i in 2 ≤ .. ≤ n) {
16          result *= i
17      }
18      return result
19  }
20

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.1\lib\idea_rt.jar-51747:C:\Program Files\JetBrains\Inte
Введите число для вычисления факториала: 5
Факториал числа 5 равен 120

Process finished with exit code 0

```

Задание 5

```

fun main() {
    print("Введите число для проверки на простоту: ")
    val number = readLine()!!.toInt()

    if (isPrime(number)) {
        println("$number является простым числом.")
    } else {
        println("$number не является простым числом.")
    }
}

fun isPrime(num: Int): Boolean {
    if (num <= 1) return false
    if (num == 2) return true

    for (i in 2 ≤ .. ≤ Math.sqrt(num.toDouble()).toInt()) {
        if (num % i == 0) return false
    }
    return true
}

```

```

C:\Users\Student\jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.1\lib\idea_rt.jar=51786:C:\Program
Введите число для проверки на простоту: 5
5 является простым числом.

Process finished with exit code 0

```

Задание 6

```

1  ▶ fun main() {
2      println("Таблица умножения от 1 до 10")
3      for (i in 1 ≤ .. ≤ 10) {
4          for (j in 1 ≤ .. ≤ 10) {
5              print("${i * j}\t")
6          }
7          println() // Переход на новую строку после каждой строки таблицы
8      }
9  }
10

```

Таблица умножения от 1 до 10									
1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

Задание 7

```

1  fun main() {
2      print("Введите количество чисел Фибоначчи (N): ")
3      val n = readLine()?.toIntOrNull() ?: return
4
5      if (n <= 0) {
6          println("Пожалуйста, введите положительное число.")
7          return
8      }
9
10     val fibonacci = mutableListOf<Int>()
11     var a = 0
12     var b = 1
13
14     for (i in 1..n) {
15         fibonacci.add(a)
16         val next = a + b
17         a = b
18         b = next
19     }
20
21     println("Первые $n чисел Фибоначчи: $fibonacci")
22 }

```

C:\Users\student\jdk\openjdk-22.0.2\bin\java.exe -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.1\lib\idea_rt.jar=92014:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.1
Введите количество чисел Фибоначчи (N): 8
Первые 8 чисел Фибоначчи: [0, 1, 1, 2, 3, 5, 8, 13]
Process finished with exit code 0

Задание 8

```

1 fun main() {
2     print("Введите первое число: ")
3     val a = readLine()?.toIntOrNull() ?: return
4
5     print("Введите второе число: ")
6     val b = readLine()?.toIntOrNull() ?: return
7
8     val gcd = euclideanGCD(a, b)
9     println("Наибольший общий делитель (НОД) чисел $a и $b: $gcd")
10 }
11
12 fun euclideanGCD(a: Int, b: Int): Int {
13     var x = a
14     var y = b
15
16     while (y != 0) {
17         val temp = y
18         y = x % y
19         x = temp
20     }
21     return x
22 }
23

```

```

C:\Users\Student\jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.1\lib
Введите первое число: 9
Введите второе число: 5
Наибольший общий делитель (НОД) чисел 9 и 5: 1

Process finished with exit code 0
|

```

Задание 9

```

1 fun main() {
2     print("Введите строку: ")
3     val input = readLine() ?: return
4
5     val reversed = input.reversed()
6     println("Обратный порядок: $reversed")
7 }
8

```

```

C:\Users\Student\jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.1\lib
Введите первое число: 9
Введите второе число: 5
Наибольший общий делитель (НОД) чисел 9 и 5: 1

Process finished with exit code 0

```

Задание 10

```
fun main() {  
    print("Введите число: ")  
    val input = readLine() ?: return  
  
    val sum = input.map { it.toString().toInt() }.sum()  
    println("Сумма цифр: $sum")  
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Fi  
Введите число: 6  
Сумма цифр: 6  
  
Process finished with exit code 0
```

Задание 11

```
fun areAnagrams(str1: String, str2: String): Boolean {  
    return str1.toLowercase().replace( oldValue: " ", newValue: "").toCharArray().sorted() ==  
           str2.toLowercase().replace( oldValue: " ", newValue: "").toCharArray().sorted()  
}  
  
fun main() {  
    println("Введите первую строку: ")  
    val string1 = readLine() ?: ""  
    println("Введите вторую строку: ")  
    val string2 = readLine() ?: ""  
  
    if (areAnagrams(string1, string2)) {  
        println("Строки являются анаграммами.")  
    } else {  
        println("Строки не являются анаграммами.")  
    }  
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Pro  
Введите первую строку:  
3  
Введите вторую строку:  
2  
Строки не являются анаграммами.  
  
Process finished with exit code 0  
|
```

Задание 12

```
fun generateSequence(start: Int, step: Int, count: Int): List<Int> {
    val sequence = mutableListOf<Int>()
    for (i in 0 until count) {
        sequence.add(start + i * step)
    }
    return sequence
}

fun main() {
    println("Введите начальное число: ")
    val start = readLine()?.toIntOrNull() ?: return
    println("Введите шаг: ")
    val step = readLine()?.toIntOrNull() ?: return
    println("Введите количество элементов: ")
    val count = readLine()?.toIntOrNull() ?: return

    val sequence = generateSequence(start, step, count)
    println("Сгенерированная последовательность: $sequence")
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\
Введите начальное число:
3
Введите шаг:
2
Введите количество элементов:
3
Сгенерированная последовательность: [3, 5, 7]

Process finished with exit code 0
```

Задание 13


```

fun main() {
    println("Таблица квадратов чисел от 1 до 20")
    println("-----")
    println("| Число | Квадрат          |")
    println("-----")

    for (i in 1..20) {
        val square = i * i
        println("| $i\t | $square\t |")
    }

    println("-----")
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java
Таблица квадратов чисел от 1 до 20
-----
| Число | Квадрат          |
-----
| 1 | 1 |
| 2 | 4 |
| 3 | 9 |
| 4 | 16 |
| 5 | 25 |
| 6 | 36 |
| 7 | 49 |

```

Задание 14

```

import kotlin.random.Random

fun main() {
    val randomNumbers = List(size = 10) { Random.nextInt(from = 1, until = 101) }
    println("Случайные числа от 1 до 100:")
    randomNumbers.forEach { println(it) }
}

```

```
C:\Users\Student\.jdk\openjdk-22.0
Случайные числа от 1 до 100:
35
56
28
33
2
45
37
100
64
38
```

Задание 15

```
fun main() {
    println("Введите строку:")
    val input = readLine()?.trim() ?: ""

    val isPalindrome = input.equals(input.reversed(), ignoreCase = true)

    if (isPalindrome) {
        println("Строка является палиндромом.")
    } else {
        println("Строка не является палиндромом.")
    }
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe
Введите строку:
3
Строка является палиндромом.

Process finished with exit code 0
```

Задание 16

```

fun main() {
    println("Введите число N:")
    val n = readLine()?.toIntOrNull()

    if (n != null && n > 0) {
        val sumOfSquares = (1 ≤ .. ≤ n).sumOf { it * it }
        println("Сумма квадратов чисел от 1 до $n равна $sumOfSquares")
    } else {
        println("Пожалуйста, введите положительное целое число.")
    }
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:
Введите число N:
6
Сумма квадратов чисел от 1 до 6 равна 91

Process finished with exit code 0

```

Задание 17

```

fun main() {
    println("Введите строку:")
    val input = readLine() ?: ""

    for (char in input) {
        println(char)
    }
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "
Введите строку:
4
4

Process finished with exit code 0
|

```

Задание 18

```
fun main() {
    println("Введите высоту лестницы:")
    val n = readLine()?.toIntOrNull() ?: return

    for (i in 1..n) {
        println("#".repeat(i))
    }
}
```

Введите высоту лестницы:

```
9
#
##
###
####
#####
#####
#####
#####
#####
```

Задание 19

```
fun main() {
    val array = intArrayOf(34, 12, 45, 23, 67, 89, 10, 56)
    println("Исходный массив: ${array.joinToString(separator: ", ")}")

    for (i in array.indices) {
        var minIndex = i
        for (j in (i + 1) until array.size) {
            if (array[j] < array[minIndex]) {
                minIndex = j
            }
        }
        // Меняем местами
        val temp = array[i]
        array[i] = array[minIndex]
        array[minIndex] = temp
    }

    println("Отсортированный массив: ${array.joinToString(separator: ", ")}")
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\JetB
Исходный массив: 34, 12, 45, 23, 67, 89, 10, 56
Отсортированный массив: 10, 12, 23, 34, 45, 56, 67, 89

Process finished with exit code 0
```

Задание20

```
fun main() {
    println("Введите нижнюю границу диапазона:")
    val lowerBound = readLine()?.toIntOrNull() ?: return
    println("Введите верхнюю границу диапазона:")
    val upperBound = readLine()?.toIntOrNull() ?: return

    fun isPrime(number: Int): Boolean {
        if (number < 2) return false
        for (i in 2 ≤ until < number) {
            if (number % i == 0) {
                return false
            }
        }
        return true
    }

    println("Простые числа в диапазоне от $lowerBound до $upperBound:")
    for (num in lowerBound ≤ .. ≤ upperBound) {
        if (isPrime(num)) {
            println(num)
        }
    }
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent
Введите нижнюю границу диапазона:
6
Введите верхнюю границу диапазона:
5
Простые числа в диапазоне от 6 до 5:

Process finished with exit code 0
```

Задание 21

```
import java.time.LocalDate
import java.time.YearMonth
import java.util.Scanner

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

    println("Введите год:")
    val year = scanner.nextInt()
    println("Введите месяц (от 1 до 12):")
    val month = scanner.nextInt()

    // Проверяем корректность введенного месяца
    if (month < 1 || month > 12) {
        println("Некорректный месяц. Пожалуйста, введите месяц от 1 до 12.")
        return
    }

    // Получаем количество дней в месяце
    val yearMonth = YearMonth.of(year, month)
    val daysInMonth = yearMonth.lengthOfMonth()

    println("Даты в месяце $month года $year:")
    for (day in 1..daysInMonth) {
        val date = LocalDate.of(year, month, day)
        println(date)
    }
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Progr
Введите год:
1904
Введите месяц (от 1 до 12):
4
Даты в месяце 4 года 1904:
1904-04-01
1904-04-02
1904-04-03
1904-04-04
1904-04-05
1904-04-06
1904-04-07
1904-04-08
1904-04-09
1904-04-10
1904-04-11
1904-04-12
1904-04-13
1904-04-14
1904-04-15
1904-04-16
1904-04-17
1904-04-18
1904-04-19
1904-04-20
1904-04-21
1904-04-22
1904-04-23
1904-04-24
1904-04-25
1904-04-26
1904-04-27
1904-04-28
1904-04-29
1904-04-30
```

Задание 22

```

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

fun main() {
    val scanner = Scanner(System.`in`)
    val randomNumber = Random.nextInt(from: 1, until: 101) // Генерируем случайное число от 1 до 100
    var guess: Int
    var attempts = 0

    println("Угадай число от 1 до 100!")

    do {
        println("Введите ваше предположение:")
        guess = scanner.nextInt()
        attempts++

        when {
            guess < randomNumber -> println("Загаданное число больше.")
            guess > randomNumber -> println("Загаданное число меньше.")
            else -> println("Поздравляю! Вы угадали число $randomNumber за $attempts попыток.")
        }
    } while (guess != randomNumber)
}

```

```

C:\Users\Student\.jdk\openjdk-22.
Угадай число от 1 до 100!
Введите ваше предположение:
93
Загаданное число меньше.
Введите ваше предположение:
10
Загаданное число больше.
Введите ваше предположение:
56
Загаданное число меньше.
Введите ваше предположение:
45
Загаданное число меньше.
Введите ваше предположение:
|

```

Задание 23


```
import java.util.Scanner

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

    while (true) {
        println("Введите первую цифру или 'стоп' для выхода:")
        val input1 = scanner.nextLine()

        if (input1.lowercase() == "стоп") break

        println("Введите вторую цифру или 'стоп' для выхода:")
        val input2 = scanner.nextLine()

        if (input2.lowercase() == "стоп") break

        val num1 = input1.toDoubleOrNull()
        val num2 = input2.toDoubleOrNull()

        if (num1 != null && num2 != null) {
            println("Выберите операцию: 1 - Сложение, 2 - Умножение")
            val operation = scanner.nextLine()

            when (operation) {
                "1" -> println("Результат сложения: ${num1 + num2}")
                "2" -> println("Результат умножения: ${num1 * num2}")
                else -> println("Неверный выбор операции. Попробуйте снова.")
            }
        } else {
            println("Введите корректные цифры.")
        }
    }

    println("Вы вышли из программы.")
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files\
Введите первую цифру или 'стоп' для выхода:
4
Введите вторую цифру или 'стоп' для выхода:
3
Выберите операцию: 1 - Сложение, 2 - Умножение
1
Результат сложения: 7.0
Введите первую цифру или 'стоп' для выхода:
|
```

Задание 24

```

fun main() {
    // Исходная матрица
    val matrix = arrayOf(
        arrayOf(1, 2, 3),
        arrayOf(4, 5, 6),
        arrayOf(7, 8, 9)
    )

    // Вывод исходной матрицы
    println("Исходная матрица:")
    printMatrix(matrix)

    // Транспонирование матрицы
    val transposedMatrix = transposeMatrix(matrix)

    // Вывод транспонированной матрицы
    println("Транспонированная матрица:")
    printMatrix(transposedMatrix)
}

fun transposeMatrix(matrix: Array<Array<Int>>): Array<Array<Int>> {
    val rowCount = matrix.size
    val colCount = matrix[0].size
    val transposed = Array(colCount) { Array(rowCount) { 0 } }

    for (i in 0 until rowCount) {
        for (j in 0 until colCount) {
            transposed[j][i] = matrix[i][j]
        }
    }
    return transposed
}

fun printMatrix(matrix: Array<Array<Int>>) {
    for (row in matrix) {
        println(row.joinToString(separator: " "))
    }
}

```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin
Исходная матрица:
1 2 3
4 5 6
7 8 9
Транспонированная матрица:
1 4 7
2 5 8
3 6 9

Process finished with exit code 0
```

Задание 25

```
fun main() {
    println("Кубы чисел от 1 до 10:")

    for (i in 1..10) {
        val cube = i * i * i
        println("Куб числа $i: $cube")
    }
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin
Кубы чисел от 1 до 10:
Куб числа 1: 1
Куб числа 2: 8
Куб числа 3: 27
Куб числа 4: 64
Куб числа 5: 125
Куб числа 6: 216
Куб числа 7: 343
Куб числа 8: 512
Куб числа 9: 729
Куб числа 10: 1000

Process finished with exit code 0
|
```

Задание 26

```

fun main() {
    println("Введите число N:")
    val n = readLine()?.toIntOrNull()

    if (n != null && n > 0) {
        var evenSum = 0
        var oddSum = 0

        for (i in 1 ≤ .. ≤ n) {
            if (i % 2 == 0) {
                evenSum += i
            } else {
                oddSum += i
            }
        }

        println("Сумма четных чисел от 1 до $n: $evenSum")
        println("Сумма нечетных чисел от 1 до $n: $oddSum")
    } else {
        println("Пожалуйста, введите положительное целое число.")
    }
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\
Введите число N:
4
Сумма четных чисел от 1 до 4: 6
Сумма нечетных чисел от 1 до 4: 4

Process finished with exit code 0
|

```

Задание 27

```

fun main() {
    println("Введите число N:")
    val n = readLine()?.toIntOrNull()
    if (n != null && n > 0) {
        printPyramid(n)
    } else {
        println("Некорректный ввод. Пожалуйста, введите положительное целое число.")
    }
}

fun printPyramid(n: Int) {
    for (i in 1 ≤ .. ≤ n) {
        // Печать пробелов для выравнивания
        repeat( times: n - i) {
            print(" ")
        }
        // Печать чисел
        for (j in 1 ≤ .. ≤ i) {
            print("$j ")
        }
        // Переход на новую строку
        println()
    }
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin
Введите число N:
9

    1
  1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5 6 7
1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8 9

Process finished with exit code 0
|

```

Задание 28

```

fun main() {
    println("Введите количество чисел (N):")
    val n = readLine()?.toIntOrNull()
    if (n != null && n > 0) {
        val numbers = mutableListOf<Int>()
        println("Введите $n чисел:")
        for (i in 1..n) {
            val number = readLine()?.toIntOrNull()
            if (number != null) {
                numbers.add(number)
            } else {
                println("Некорректный ввод. Пожалуйста, введите целое число.")
                i-1 // Уменьшаем счетчик, чтобы повторить ввод
            }
        }
        // Сортировка чисел
        numbers.sort()
        // Вывод отсортированных чисел
        println("Числа в порядке возрастания:")
        for (number in numbers) {
            print("$number ")
        }
    } else {
        println("Некорректный ввод. Пожалуйста, введите положительное целое число.")
    }
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe
Введите количество чисел (N):
3
Введите 3 чисел:
3
3
3
Числа в порядке возрастания:
3 3 3
Process finished with exit code 0
|

```

Задание 29

```

fun main() {
    println("Введите число N:")
    val n = readLine()?.toIntOrNull()

    if (n != null && n > 0) {
        val sum = calculateSeriesSum(n)
        println("Сумма ряда 1 + 1/2 + 1/3 + ... + 1/$n = $sum")
    } else {
        println("Некорректный ввод. Пожалуйста, введите положительное целое число.")
    }
}

fun calculateSeriesSum(n: Int): Double {
    var sum = 0.0
    for (i in 1..n) {
        sum += 1.0 / i
    }
    return sum
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C:\Program Files
Введите число N:
4
Сумма ряда 1 + 1/2 + 1/3 + ... + 1/4 = 2.0833333333333333

Process finished with exit code 0
|

```

Задание 30

```

fun main() {
    println("Введите целое число:")
    val number = readLine()?.toIntOrNull()
    if (number != null) {
        val binaryString = convertToBinary(number)
        println("Число $number в двоичной системе: $binaryString")
    } else {
        println("Некорректный ввод. Пожалуйста, введите целое число.")
    }
}

fun convertToBinary(number: Int): String {
    return Integer.toBinaryString(number)
}

```



```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe
```

```
Введите целое число:
```

```
4
```

```
Число 4 в двоичной системе: 100
```

```
Process finished with exit code 0
```