

Гончарова Е.Р

1. Генератор случайных чисел

```
import kotlin.random.Random

fun main() {
    val randomNumbers = List(10) { Random.nextInt(1, 100) }
    println("Случайные числа: $randomNumbers")
}

C:\Users\USER\.jdk\openjdk-23.0.2\bin\java.exe "-javaagent
Случайные числа: [80, 20, 31, 2, 86, 69, 27, 85, 45, 69]

Process finished with exit code 0
```

2. Строковый анализатор

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

    val vowels = "аеёиоуыэюяАЕЁИОУЫЭЮЯ"
    val consonants = "бвгджзйклмнпрстфхцчшщБВГДЖЗЙКЛМНПРСТФХЦЧШЩ"

    var vowelCount = 0
    var consonantCount = 0

    for (char in input) {
        when {
            char in vowels -> vowelCount++
            char in consonants -> consonantCount++
            char.isLetter() -> {}
        }
    }

    println("Количество гласных букв: $vowelCount")
    println("Количество согласных букв: $consonantCount")
}

C:\Users\USER\.jdk\openjdk-23
Введите строку:
Сибун Лазурная башня
Количество гласных букв: 9
Количество согласных букв: 9
```

3. Конвертер валют

```
fun main() {
    val usdToEurRate = 0.85
    val eurToUsdRate = 1 / usdToEurRate

    println("Выберите валюту для конвертации:")
    println("1: Доллары (USD)")
    println("2: Евро (EUR)")

    val choice = readLine()?.toIntOrNull()

    if (choice == null || (choice != 1 && choice != 2)) {
        println("Неверный выбор. Выберите 1 или 2.")
        return
    }
}
```

```
println("Введите сумму для конвертации:")
val amount = readLine()?.toDoubleOrNull()

if (amount == null || amount < 0) {
    println("Неверная сумма. Введите положительное число.")
    return
}

when (choice) {
    1 -> {
        val convertedAmount = amount * usdToEurRate
        println("$amount USD = $convertedAmount EUR")
    }
    2 -> {
        val convertedAmount = amount * eurToUsdRate
        println("$amount EUR = $convertedAmount USD")
    }
}
}
```

Выберите валюту для конвертации:
1: Доллары (USD)
2: Евро (EUR)
2
Введите сумму для конвертации:
100
100.0 EUR = 117.64705882352942 USD

4. Проверка на анаграмму

```
fun areAnagrams(str1: String, str2: String): Boolean {
    if (str1.length != str2.length) {
        return false
    }
    val charCount = IntArray(26)
    for (i in str1.indices) {
        charCount[str1[i] - 'a']++
        charCount[str2[i] - 'a']--
    }
    for (count in charCount) {
        if (count != 0) {
            return false
        }
    }
    return true
}

fun main() {
    val str1 = "listen"
    val str2 = "silent"
    val areAnagrams = areAnagrams(str1, str2)

    if (areAnagrams) {
        println("$str1 и $str2 являются анаграммами.")
    } else {
        println("$str1 и $str2 не являются анаграммами.")
    }
}

C:\Users\USER\.jdk\openjdk-23.0.2\bin
listen и silent являются анаграммами.

Process finished with exit code 0
```

5. Нахождение простых чисел

```
fun isPrime(num: Int): Boolean {
    if (num <= 1) return false
    for (i in 2..Math.sqrt(num.toDouble()).toInt()) {
        if (num % i == 0) return false
    }
    return true
}

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

    if (n != null && n > 0) {
        println("Простые числа до $n:")
        for (i in 2..n) {
            if (isPrime(i)) {
                println(i)
            }
        }
    } else {
        println("Ведите положительное целое число.")
    }
}
```

```
C:\Users\USER\.jdk\openjdk-23.0.2\bin\java.exe
Введите число N:
8
Простые числа до 8:
2
3
5
7
```

6. Сортировка строк

```
fun sortStrings(strings: Array<String>): Array<String> {
    return strings.sortedArray()
}

fun main() {
    val strings = arrayOf("снег", "акула", "синий", "кушать", "магичка")

    println("Исходный массив строк:")
    println(strings.joinToString(", "))

    val sortedStrings = sortStrings(strings)

    println("Отсортированный массив строк:")
    println(sortedStrings.joinToString(", "))
}
```

```
C:\Users\USER\.jdk\openjdk-23.0.2\bin\java.exe
Исходный массив строк:
снег, акула, синий, кушать, магичка
Отсортированный массив строк:
акула, кушать, магичка, синий, снег
```

7. Изменение регистра

```
fun changeCase(input: String): String {
    return input.map {
```

```

        if (it.isUpperCase()) it.lowercase()
        else it.uppercase()
    }.joinToString("")
}

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

    val changedCase = changeCase(userInput)

    println("Измененная строка:")
    println(changedCase)
}

```

```

C:\Users\USER\.jdk\
Введите строку:
ЛаЗурная Башня
Измененная строка:
лАзУрНАЯ БАШНЯ

Process finished with

```

8. Игра «Угадай число»

```

import kotlin.random.Random

fun main() {
    val randomNumber = Random.nextInt(1, 101)
    var guess: Int?
    var attempts = 0
    println("Рады видеть вас в игре 'Отгадай число!'")
    println("Я задумал число в диапазоне от 1 до 100. Ваша задача — отгадать его.")
    do {
        println("Напишите ваш вариант:")
        val input = readLine()
        guess = input?.toIntOrNull()
        if (guess == null) {
            println("Укажите допустимое числовое значение")
            continue
        }
        attempts++
        when {
            guess < randomNumber -> println("Загаданное число выше.")
            guess > randomNumber -> println("Загаданное число меньше.")
            else -> println("Ура! Вы справились $randomNumber за $attempts раз(a)!")
        }
    } while (guess != randomNumber)
}

```

```
C:\Users\USER\.jdk\openjdk-23.0.2\bin\java.exe "-javaagent:C:\Progr
Рады видеть вас в игре 'Отгадай число'!
Я задумал число в диапазоне от 1 до 100. Ваша задача – отгадать его.
Напишите ваш вариант:
20
Загаданное число выше.
Напишите ваш вариант:
45
Загаданное число меньше.
Напишите ваш вариант:
30
Загаданное число выше.
Напишите ваш вариант:
33
Загаданное число меньше.
Напишите ваш вариант:
32
Ура! Вы справились 32 за 5 раз(а)!
```

9. Генератор паролей

```
import kotlin.random.Random
fun main() {
    println("Введите желаемую длину пароля:")
    val length = readLine()?.toIntOrNull() ?: 12
    val password = generatePassword(length)
    println("Сгенерированный пароль: $password")
}
fun generatePassword(length: Int): String {
    val lowercaseLetters = ('a'..'z').toList()
    val uppercaseLetters = ('A'..'Z').toList()
    val digits = ('0'..'9').toList()
    val specialCharacters = listOf('!', '@', '#', '$', '%', '^', '&', '*',
    '(', ')', '-', '_', '=', '+')

    val allCharacters = lowercaseLetters + uppercaseLetters + digits +
specialCharacters
    return (1..length)
        .map { allCharacters.random() }
        .joinToString("")
}
C:\Users\USER\.jdk\openjdk-23.0.2
Введите желаемую длину пароля:
8
Сгенерированный пароль: Mb61GuTi
```

10. Функция, возвращающая самое длинное слово в строке

```
fun findLongestWord(text: String): String {
    require(text.isNotEmpty()) { "Строка не должна быть пустой" }
    return text.split(" ")
        .map { it.filter { it.isLetterOrDigit() } }
        .maxByOrNull { it.length } ?: ""
}
fun main() {
    println(findLongestWord("Не трать свою энергию попусту"))
}
C:\Users\
энергию
```