

1. Генератор случайных чисел: Напишите программу, которая генерирует и выводит 10 случайных чисел от 1 до 100.

```
import kotlin.random.Random
fun main() {
    val randomNumbers = List(10) { Random.nextInt(1, 101) }
    println("Случайные числа: $randomNumbers")
}

Случайные числа: [99, 88, 54, 89, 22, 84, 22, 31, 40, 64]

Process finished with exit code 0
```

2. Строковый анализатор: Напишите программу, которая принимает строку и выводит количество гласных и согласных букв.

```
fun main() {
    println("Введите строку:")
    val input = readLine() ?: ""
    val vowels = "аеёиоуыэюяАЕЁИОУЫЭЮЯ"
    var vowelCount = 0
    var consonantCount = 0
    for (char in input) {
        if (char.isLetter()) {
            if (char in vowels) {
                vowelCount++
            } else {
                consonantCount++
            }
        }
    }
    println("Гласные: $vowelCount, Согласные: $consonantCount")
}

Введите строку:
привет
Гласные: 2, Согласные: 4

Process finished with exit code 0
```

3. Конвертер валют: Реализуйте программу, которая конвертирует одну валюту в другую (например, доллар в евро).

```

fun main() {
    println("Введите сумму в долларах:")
    val dollars = readLine()?.toDoubleOrNull() ?: return
    val exchangeRate = 0.88
    val euros = dollars * exchangeRate
    println("$dollars долларов = $euros евро")
}

```

Введите сумму в долларах:

33

33.0 долларов = 29.04 евро

Process finished with exit code 0

4. Проверка на анаграмму: Напишите функцию, которая проверяет, являются ли две строки анаграммами.

```

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

```

```
Введите первую строку:
кулон
Введите вторую строку:
клоун
Строки являются анаграммами.

Process finished with exit code 0
```

5. Нахождение простых чисел: Реализуйте программу, которая находит и выводит все простые числа до заданного числа N.

```
fun isPrime(n: Int): Boolean {
    if (n < 2) return false
    for (i in 2..Math.sqrt(n.toDouble()).toInt()) {
        if (n % i == 0) return false
    }
    return true
}
fun main() {
    println("Введите число N:")
    val n = readLine()?.toIntOrNull() ?: return
    val primes = (2..n).filter { isPrime(it) }
    println("Простые числа до $n: $primes")
}

Введите число N:
55
Простые числа до 55: [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53]

Process finished with exit code 0
```

6. Сортировка строк: Напишите функцию, которая принимает массив строк и сортирует его по алфавиту.

```
fun main() {
    println("Введите строки, разделенные запятыми:")
    val input = readLine() ?: ""
    val strings = input.split(",").map { it.trim() }.sorted()
    println("Отсортированные строки: $strings")
}
```

```
Введите строки, разделенные запятыми:
цыкуевапмроо, йцуыевапритбьюбю, зщнлорсм, кнеапм
Отсортированные строки: [зщнлорсм, йцуыевапритбьюбю, кнеапм, цыкуевапмроо]

Process finished with exit code 0
```

7. Изменение регистра: Создайте программу, которая принимает строку и меняет регистр всех букв на противоположный.

```
fun main() {
    println("Введите строку:")
    val input = readLine() ?: ""
    val toggledCase = input.map {
        if (it.isUpperCase()) it.lowercaseChar() else
it.uppercaseChar()
    }.joinToString("")
    println("Изменённая строка: $toggledCase")
}
```

```
Введите строку:
укнеаоплротбь
Изменённая строка: УКНЕАОПЛРОТБЬ

Process finished with exit code 0
```

8. Игра "Угадай число": Напишите консольную игру, в которой пользователь должен угадать случайное число от 1 до 100, а программа подсказывает, больше или меньше загаданное число.

```
import kotlin.random.Random
fun main() {
    val numberToGuess = Random.nextInt(1, 101)
    var guess: Int?
    println("Угадайте число от 1 до 100:")
    do {
        guess = readLine()?.toIntOrNull()
        when {
            guess == null -> println("Введите число!")
            guess < numberToGuess -> println("Слишком маленькое
число!")
        }
    }
```

```

        guess > numberToGuess -> println("Слишком большое
число!")
        else -> println("Поздравляю! Вы угадали число!")
    }
} while (guess != numberToGuess)
}

```

Угадайте число от 1 до 100:

55

Слишком большое число!

45

Слишком большое число!

30

Слишком маленькое число!

36

Слишком большое число!

35

Слишком большое число!

34

Слишком большое число!

33

Поздравляю! Вы угадали число!

Process finished with exit code 0

9. Генератор паролей: Реализуйте упражнение по генерации случайного пароля заданной длины, используя цифры, буквы и специальные символы.

```

import kotlin.random.Random
fun generatePassword(length: Int): String {
    val chars =
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#$$%^&*()"
    return (1..length)
        .map { chars[Random.nextInt(chars.length)] }
        .joinToString("")
}
fun main() {
    println("Введите длину пароля:")
    val length = readLine()?.toIntOrNull() ?: return
}

```

```

val password = generatePassword(length)
println("Сгенерированный пароль: $password")
}

```

```

Введите длину пароля:
7
Сгенерированный пароль: X3VmGrx

Process finished with exit code 0

```

10. По заданной строке реализуйте функцию, которая возвращает самое длинное слово в этой строке. Если есть два или более слов одинаковой длины, верните первое самое длинное слово из строки. Игнорируйте знаки препинания. Входная строка не может быть пустой или незаполненным отображением.

```

fun main() {
    println("Введите строку:")
    val input = readLnOrNull()?.takeIf { it.isNotBlank() } ?: run {
        println("Ошибка: строка не может быть пустой!")
        return
    }
    val words = input.split(Regex("[\\s\\p{Punct}]+"))
        .filter { it.isNotBlank() }
        .map { it.replace(Regex("[\\p{Punct}]"), "") }
    if (words.isEmpty()) {
        println("Ошибка: в строке нет слов!")
        return
    }
    val longestWord = words.maxByOrNull { it.length }!!
    println("Самое длинное слово: '$longestWord' (длина:
    ${longestWord.length})")
}

```

```

Введите строку:
raestdgcgvhjnm setrdycvvhbkjm srdyvtvhjknjl
Самое длинное слово: 'raestdgcgvhjnm' (длина: 13)

Process finished with exit code 0

```