

Практическая работа 9

Творческие задачи

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

```
import kotlin.random.Random
fun main() {
    var f = 0
    for(i in 1..10){
        print(Random.nextInt(1,100))
        f++
        if (f>=10)
            println()
        else
            print(",")
    }
}
```

```
C:\Users\Student\.jdk\openjdk-22.0
88,13,11,43,79,55,84,99,26,73

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 текст: ")
    val text = readLine() ?: ""
    val glas = "аеёиоуыэюя"
    val sogl = "бвгджзйклмнпрстфхцчшщ"
    var glasCount = 0
    var soglCount = 0
    for (bukva in text.lowercase()) {
        if (bukva in glas) {
            glasCount++
        }
        if (bukva in sogl) {
            soglCount++
        }
    }

    println("Гласных букв: $glasCount")
    println("Согласных букв: $soglCount")
}
```

```
C:\Users\Student\.jdk
Введите текст:
ДенисКокос
Гласных букв: 4
Согласных букв: 6
```

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

```
fun main() {  
    print("Введите число в долларах: ")  
    val dollars = readLine()?.toDoubleOrNull()?:0.0  
    print("Введите текущий курс: ")  
    val exchangeRate = readLine()?.toDoubleOrNull()?:0.0  
    val euros = dollars * exchangeRate  
    println("$dollars долларов = $euros евро")  
}
```

```
C:\Users\Student\.jdk\openjdk-  
Введите число в долларах: 100  
Введите текущий курс: 0.88  
100.0 долларов = 88.0 евро
```

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

```
fun main() {  
    println("Введите первую строку:")  
    val s1 = readLine() ?: return  
    println("Введите вторую строку:")  
    val s2 = readLine() ?: return  
    val ana = s1.toCharArray().sorted() == s2.toCharArray().sorted()  
    println(if (ana) "Строки являются анаграммами" else "Строки не являются  
анаграммами")  
}
```

```
C:\Users\Student\.jdk\openjdk-  
Введите первую строку:  
пила  
Введите вторую строку:  
липа  
Строки являются анаграммами
```

```
Введите первую строку:  
день  
Введите вторую строку:  
ночь  
Строки не являются анаграммами
```

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

```
fun findPrimes(n: Int): List<Int> {  
    val sieve = BooleanArray(n+1){true}  
    sieve[0] = false  
    sieve[1] = false  
    for (i in 2..n){  
        if (sieve[i]){  
            for(j in i * i..n step i) {  
                sieve[j] = false  
            }  
        }  
    }  
}
```

```

    }
    return sieve.indices.filter {sieve[it]} .toList()
}

fun main() {
    println("Введите число N: ")
    val n = readLine()?.toIntOrNull()?:0
    val primes = findPrimes(n)
    println("Простые числа до $n: ${primes.joinToString()}")
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\j
Введите число N:
15
Простые числа до 15: 2, 3, 5, 7, 11, 13

```

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

```

fun main () {
    println("Введите строку через запятую")
    val input = readln().trim()
    val strings = input.split(",").map {it.trim()} .toTypedArray()
    val sorted = strings.sortedArray()
    println("Отсортированный массив: ${sorted.contentToString()}")
}

```

```

C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent:C
Введите строку через запятую
яблоко, кокос, банан, абрикос, вишня
Отсортированный массив: [абрикос, банан, вишня, кокос, яблоко]

```

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

```

fun main() {
    println("Введите строку для инверсии регистра: ")
    val input = readln()
    val result = input.map {
        if (it.isLowerCase()) it.uppercaseChar()
        else it.lowercaseChar()
    }.joinToString("")
    println("Результат: $result")
}

```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin
Введите строку для инверсии регистра:
Register InvErsiOn
Результат: rEGISTEr iNvErSIoN
```

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.

```
fun main() {
    val secret = (1..100).random()
    var attempts = 0
    println("Угадайте число от 1 до 100:")
    while (true) {
        val guess = readln().toIntOrNull()
        attempts++
        when {
            guess == null -> println("Введите число!")
            guess < secret -> println("Больше")
            guess > secret -> println("Меньше")
            else -> {
                println("Победа! Попыток: $attempts")
                break
            }
        }
    }
}
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin
Угадайте число от 1 до 100:
55
Меньше
20
Больше
25
Больше
40
Больше
45
Меньше
43
Меньше
42
Победа! Попыток: 7
```

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

```

import kotlin.random.Random
fun main() {
    val length = getPasswordLength()
    val password = generatePassword(length)
    println("Ваш пароль: $password")
}
private fun getPasswordLength(): Int {
    while (true) {
        print("Введите длину пароля (8-80 символов): ")
        val input = readln().trim()
        when {
            input.isEmpty() -> println("Вы ничего не ввели!")
            !input.matches(Regex("\\d+")) -> println("Введите число!")
            else -> {
                val length = input.toInt()
                when {
                    length < 8 -> println("Пароль слишком короткий!")
                    length > 80 -> println("Пароль слишком длинный!")
                    else -> return length
                }
            }
        }
    }
}
fun generatePassword(length: Int): String {
    val charGroups = listOf(
        ('a'..'z').toList(),
        ('A'..'Z').toList(),
        ('0'..'9').toList(),
        listOf('!', '@', '#', '$', '%', '^', '&', '*')
    )
    val password = mutableListOf<Char>().apply {
        repeat(length) {
            add(charGroups[Random.nextInt(charGroups.size)].random())
        }
    }
    return password.shuffled().joinToString("")
}

```

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 текст для анализа:")
    val text = readln()
    val longest = text.split(Regex("[^\\p{L}-']"))
        .filter { it.isNotBlank() && it.any(Char::isLetter) }
        .maxByOrNull { it.length } ?: ""
    println("Самое длинное слово: $longest")
}

```

C:\Users\Student\.jdk\openjdk-22

Введите текст для анализа:

Введите текст для анализа:

Самое длинное слово: Введите