Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра проектирования информационно-компьютерных систем

Дисциплина Программирование мобильных информационных систем

Отчет

по лабораторной работе

на тему:

**«ОСНОВЫ ЯЗЫКА *KOTLIN*»**

Выполнил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Булатевич В.Ю.

(подпись)

гр. 214301

Проверил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Усенко Ф.В.

(подпись)

Минск, 2024

**Цель работы:** Изучить основные конструкции языка Kotlin, научиться работать с типами данных, операциями ввода-вывода, а также условными операторами. Закрепить полученные знания через выполнение задач, требующих практического применения теории.

**Задание**: Напишите программу, которая вычисляет как среднее арифметическое, так и среднее геометрическое и гармоническое для ряда чисел. Реализуйте проверку на корректность ввода данных.

**1 Реализация задания**

import kotlin.math.pow  
  
fun main() {  
 var numbers: List<Double>? = null  
  
 // Цикл, который будет продолжаться до тех пор, пока не введены корректные значения  
 while (numbers == null || numbers.isEmpty()) {  
 *println*("Введите числа через пробел:")  
  
 // Чтение строки с числами  
 val input = *readLine*()  
  
 // Проверка, что строка не пустая и не null  
 if (input.*isNullOrBlank*()) {  
 *println*("Ошибка: строка пуста. Попробуйте снова.")  
 continue // Переход на следующую итерацию цикла  
 }  
  
 // Преобразование строки в список чисел, проверка на корректность ввода  
 numbers = input.*split*(" ").*mapNotNull***{ it**.*toDoubleOrNull*() **}** // Проверка, что введено хотя бы одно корректное число  
 if (numbers.isEmpty()) {  
 *println*("Ошибка: не введено корректных чисел. Попробуйте снова.")  
 }  
 }  
  
 // Вычисление среднего арифметического  
 val arithmeticMean = numbers.*sum*() / numbers.size  
  
 // Вычисление среднего геометрического  
 val geometricMean = numbers.*reduce***{** acc, num **->** acc \* num **}**.*pow*(1.0 / numbers.size)  
  
 // Вычисление среднего гармонического  
 val harmonicMean = numbers.size / numbers.*map***{** 1 / **it }**.*sum*()  
  
 // Вывод результатов  
 *println*("Среднее арифметическое: $arithmeticMean")  
 *println*("Среднее геометрическое: $geometricMean")  
 *println*("Среднее гармоническое: $harmonicMean")  
}

**2 Ответы на контрользые вопросы**

1 Что такое функция main в Kotlin, и какова её роль в программе?  
Функция main в Kotlin является точкой входа в программу. Она используется для выполнения кода, начиная с её вызова. Обычно она объявляется как fun main(), а её тело содержит инструкции, которые выполняются при запуске программы.

2 Чем отличаются переменные, объявленные с использованием val и var?  
Переменные, объявленные с val, являются неизменяемыми (аналог final в Java) и их значение нельзя изменить после присвоения. Переменные, объявленные с var, изменяемы, и их значение может быть переопределено.

3 Какие типы данных существуют в Kotlin для целых чисел? Приведите примеры.  
В Kotlin есть четыре основных типа данных для целых чисел: Byte (8 бит), Short (16 бит), Int (32 бита) и Long (64 бита).

4 Как объявить строковую переменную в Kotlin? Можно ли её изменить после объявления?

Строковую переменную можно объявить так:

val myString: String = "Hello, Kotlin!"

Если переменная объявлена с val, её изменить нельзя. Если используется var, то строку можно изменить, присвоив новое значение.

5 Что такое интерполяция строк, и как она работает в Kotlin?  
Интерполяция строк — это возможность вставлять значения переменных или выражений внутрь строки с помощью синтаксиса $.

6 Как в Kotlin можно обработать возможность появления null?  
Для обработки null в Kotlin используются безопасные типы. Переменной можно явно указать возможность быть null, добавив ? после типа:

val nullableString: String? = null

Для доступа к таким переменным используются операторы безопасного вызова (?.), проверка на null или оператор ?: для задания значения по умолчанию.

7 Как организован ввод данных с консоли в Kotlin?  
Для ввода данных с консоли используется функция readLine(), которая возвращает введённую строку.

8 Чем отличаются функции print() и println()?  
Функция print() выводит текст в консоль без перехода на новую строку. Функция println() выводит текст и автоматически добавляет перевод строки.

9 Как работают арифметические операторы в Kotlin? Приведите примеры.  
Арифметические операторы в Kotlin включают:

* + (сложение): val sum = 2 + 3
* - (вычитание): val difference = 5 - 2
* \* (умножение): val product = 3 \* 4
* / (деление): val quotient = 10 / 2
* % (остаток от деления): val remainder = 10 % 3

10 Какие операторы используются для сравнения чисел в Kotlin?  
Операторы сравнения в Kotlin включают:

* == (равно)
* != (не равно)
* < (меньше)
* > (больше)
* <= (меньше или равно)
* >= (больше или равно).

**Вывод**

В ходе выполнения лабораторной работы были изучены и закреплены основные концепции языка Kotlin. Были рассмотрены особенности работы с переменными (val и var), различия между неизменяемыми и изменяемыми объектами, а также применение различных типов данных, включая целочисленные, строки и nullable-типы.