Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Дисциплина: современные языки программирования

**Лабораторная работа №2**

**«Использование языка программирования Swift: функции, замыкания, перечисления»**

Вариант 15

Выполнил: Шабанович А.Б.

студент гр. 310102

Проверил: Усенко Ф.В.

Минск 2024

Цель: Выполнить разработку приложения с использованием языка программирования Swift: функции, замыкания, перечисления.

Задание 1: Дано натуральное число P. Проверьте, кратно ли P одиннадцати, используя признак делимости на 11 (знакопеременная сумма его цифр делится на 11).

Задание 2:

а) целые числа возводит в степень n;

б) из десятичных чисел извлекает корень степени n.

Листинг кода:

Задание 1:

import Foundation

func validateInput(\_ input: String) -> Int? {

    if let number = Int(input), number > 0 {

        return number

    }

    print("Ошибка: Введите натуральное число.")

    return nil

}

func isDivisibleBy11(\_ number: Int) -> Bool {

    let digits = String(number).compactMap { Int(String($0)) }

    let alternatingSum = digits.enumerated().reduce(0) { sum, pair in

        let (index, digit) = pair

        return sum + (index % 2 == 0 ? digit : -digit)

    }

    return alternatingSum % 11 == 0

}

print("Введите натуральное число P:")

if let input = readLine(), let number = validateInput(input) {

    if isDivisibleBy11(number) {

        print("Число \(number) кратно 11.")

    } else {

        print("Число \(number) не кратно 11.")

    }

} else {

    print("Некорректный ввод.")

}

Задание 2:

import Foundation

func validateIntegerInput(\_ input: String) -> Int? {

    if let number = Int(input), number >= 0 {

        return number

    }

    print("Ошибка: Введите корректное целое число (неотрицательное).")

    return nil

}

func validateDoubleInput(\_ input: String) -> Double? {

    if let number = Double(input), number >= 0 {

        return number

    }

    print("Ошибка: Введите корректное десятичное число (неотрицательное).")

    return nil

}

func calculatePower(base: Int, exponent: Int) -> Int {

    guard exponent >= 0 else {

        fatalError("Ошибка: Степень должна быть неотрицательной.")

    }

    return Int(pow(Double(base), Double(exponent)))

}

func calculateRoot(base: Double, degree: Int) -> Double? {

    guard degree > 0 else {

        print("Ошибка: Степень корня должна быть положительной.")

        return nil

    }

    return pow(base, 1.0 / Double(degree))

}

print("Введите значение n:")

if let nInput = readLine(), let n = validateIntegerInput(nInput) {

    print("Введите число для обработки (целое или десятичное):")

    if let input = readLine() {

        if let intNumber = validateIntegerInput(input) {

            // Целое, возводим в степень

            let result = calculatePower(base: intNumber, exponent: n)

            print("\(intNumber) в степени \(n) = \(result)")

        }

          else if let doubleNumber = validateDoubleInput(input) {

            // Десятичное, извлекаем корень

            if let result = calculateRoot(base: doubleNumber, degree: n) {

                print("Корень степени \(n) из \(doubleNumber) = \(result)")

            }

        } else {

            print("Ошибка: Введенное значение не является допустимым числом.")

        }

    } else {

        print("Ошибка: Ввод числа не распознан.")

    }

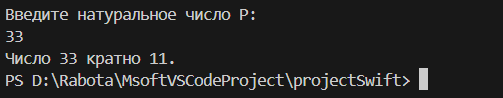
} else {

    print("Ошибка: Некорректный ввод значения n.")

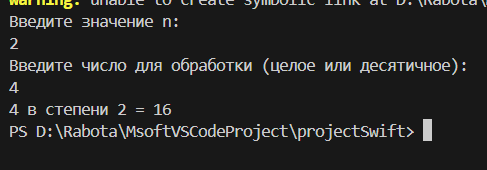
}

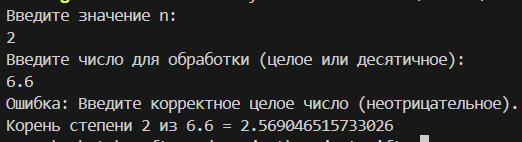
Результат выполнения кода:

Задание 1:



Задание 2:





Вывод: разработаны программы на языке программирования Swift. В результате разработки программ на языке Swift была успешно реализована функциональность, позволяющая проверять натуральное число на кратность 11, а также возводить число в заданную степень или извлекать корень степени из числа, в зависимости от вида числа. Была проделана работа с функциями, замыканиями и перечислениями.