Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

Кафедра инженерной психологии и эргономики

­­­­­

Современные языки программирования

Отчет по лабораторной работе №2

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

Выполнил: Касперец Е. А.

Студент группы 310902

Преподаватель: Усенко Ф. В.

Минск 2024

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

Задание 1: Дано натуральное число P, заданное в шестнадцатеричной системе счисления. Переведите его в двоичную систему счисления. Встроенный метод перевода не использовать.

Задание 2:

а) для умножения десятичных дробей;

б) для умножения обыкновенных дробей.

Листинг кода:

Файл main.swift:

import task1Lib

import task2Lib

print("Введите целое число в шестнадцатеричном формате:")

if let hexNumber = readLine() {

    let binaryRepresentation = hexStringToBinary(hexString: hexNumber)

    print("Шестнадцатеричное число \(hexNumber) в двоичной системе: \(binaryRepresentation)")

}

else {

    print("Не удалось прочитать ввод.")

}

let decimal1: Double = 3.5

let decimal2: Double = 2.2

let result = multiplyDecimals(decimal1, decimal2)

print("\(decimal1) \* \(decimal2) = \(result)")

let fraction1 = Fraction(numerator: 1, denominator: 2)

let fraction2 = Fraction(numerator: 3, denominator: 4)

let product = multiplyFractions(fraction1, fraction2)

print("\(fraction1.numerator)/\(fraction1.denominator) \* \(fraction2.numerator)/\(fraction2.denominator) = \(product.numerator)/\(product.denominator)")

Файл task1Lib.swift:

public func hexToBinary(hexDigit: Character) -> String? {

    let hexToBinMap: [Character: String] = [

        "0": "0000", "1": "0001", "2": "0010", "3": "0011",

        "4": "0100", "5": "0101", "6": "0110", "7": "0111",

        "8": "1000", "9": "1001", "A": "1010", "B": "1011",

        "C": "1100", "D": "1101", "E": "1110", "F": "1111"

    ]

    return hexToBinMap[hexDigit]

}

public func hexStringToBinary(hexString: String) -> String {

    var binaryResult = ""

    for hexDigit in hexString {

        if let binary  = hexToBinary(hexDigit: hexDigit) {

            binaryResult += binary

        }

        else {

            return "Ошибка: недопустимый символ в шестнадцатеричной строке."

        }

    }

    return binaryResult

}

Файл task2Lib.swift:

public func multiplyDecimals(\_ a: Double, \_ b: Double) -> Double {

    return a \* b

}

public struct Fraction {

    var numerator: Int

    var denominator: Int

    init(numerator: Int, denominator: Int) {

        precondition(denominator != 0, "Знаменатель не может быть равен нулю.")

        self.numerator = numerator

        self.denominator = denominator

    }

    func reduced() -> Fraction {

        let gcdValue = gcd(abs(numerator), abs(denominator))

        let sign = (denominator < 0) ? -1 : 1

        return Fraction(numerator: (numerator / gcdValue) \* sign,

                       denominator: abs(denominator) / gcdValue)

    }

}

public func gcd(\_ a: Int, \_ b: Int) -> Int {

    var a = a

    var b = b

    while b != 0 {

        let temp = b

        b = a % b

        a = temp

    }

    return a

}

public func multiplyFractions(\_ f1: Fraction, \_ f2: Fraction) -> Fraction {

    let num = f1.numerator \* f2.numerator

    let den = f1.denominator \* f2.denominator

    return Fraction(numerator: num, denominator: den).reduced()

}

Результат работы программы представлен на рисунке 1.

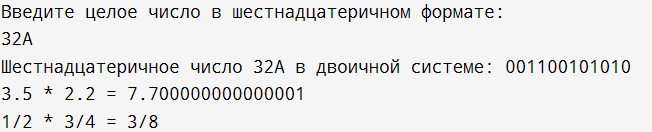


Рисунок 1 – Результат выполнения программы

Вывод: в ходе выполнения лабораторной работы номер 2 по языку программирования swift мы познакомились с основами языка и научились на практике применять основные его конструкции. Познакомились с функциями, функциональными типами, замыканиями и захватом значений.