Министерство образования

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Отчет по лабораторной работе №2

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

Выполнил: Крукович А.В.

Студент группы 310902

Проверил: Усенко Ф.В.

Минск 2024

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

**Задание 1.**Дано натуральное число P. Проверьте, кратно ли P трем, используя признак делимости на 3.

**Код на языке Swift:**

**Package.swift**

**// swift-tools-version:5.7**

**import PackageDescription**

**let package = Package(**

**name: "MyProject",**

**targets: [**

**.executableTarget(**

**name: "MainModule",**

**dependencies: ["ValidationModule", "DivisibilityModule"]**

**),**

**.target(**

**name: "ValidationModule",**

**dependencies: []**

**),**

**.target(**

**name: "DivisibilityModule",**

**dependencies: []**

**)**

**]**

**)**

**MainModule\main.swift**

**import Foundation**

**import ValidationModule**

**import DivisibilityModule**

**let p = 123 // Задаем значение для проверки**

**if Validation.validateInput(p) {**

**let isDivisible = DivisibilityChecker.isDivisibleByThree(p)**

**if isDivisible {**

**print("Число \(p) кратно 3.")**

**} else {**

**print("Число \(p) не кратно 3.")**

**}**

**} else {**

**print("Введенное значение должно быть натуральным числом больше 0.")**

**}**

**DivisibilityModule\DivisibilityChecker.swift**

**import Foundation**

**public struct DivisibilityChecker {**

**public static func isDivisibleByThree(\_ number: Int) -> Bool {**

**let digitSum = String(number).compactMap { $0.wholeNumberValue }.reduce(0, +)**

**return digitSum % 3 == 0**

**}**

**}**

**ValidationModule\Validation.swift**

**import Foundation**

**public struct Validation {**

**public static func validateInput(\_ input: Int) -> Bool {**

**return input > 0**

**}**

**}**

**Скриншот работающей программы:**

****

**Рисунок 1 - Результат выполнения программы**

**Задание 2. Используя функциональные типы, создайте программу согласно варианту.**

а) для вычисления натурального логарифма;

б) для вычисления десятичного логарифма.

**Код на языке Swift:**

**Package.swift**

**// swift-tools-version:5.7**

**import PackageDescription**

**let package = Package(**

**name: "MyLogarithmProject",**

**targets: [**

**.executableTarget(**

**name: "MainModule",**

**dependencies: ["ValidationModule", "LogarithmModule"]**

**),**

**.target(**

**name: "ValidationModule",**

**dependencies: []**

**),**

**.target(**

**name: "LogarithmModule",**

**dependencies: []**

**)**

**]**

**)**

**MainModule\main.swift**

**import Foundation**

**import ValidationModule**

**import LogarithmModule**

**let number = 100.0**

**if Validation.validatePositiveNumber(number) {**

**let lnResult = Logarithm.calculate(number, using: Logarithm.naturalLog)**

**print("Натуральный логарифм числа \(number): \(lnResult)")**

**let log10Result = Logarithm.calculate(number, using: Logarithm.decimalLog)**

**print("Десятичный логарифм числа \(number): \(log10Result)")**

**} else {**

**print("Ошибка: логарифм определяется только для положительных чисел.")**

**}**

**LogarithmModule\logarithm.swift**

**import Foundation**

**public struct Logarithm {**

**public typealias LogFunction = (Double) -> Double**

**public static let naturalLog: LogFunction = { number in**

**log(number) // Натуральный логарифм (основание e)**

**}**

**public static let decimalLog: LogFunction = { number in**

**log10(number) // Десятичный логарифм (основание 10)**

**}**

**public static func calculate(\_ number: Double, using logFunction: LogFunction) -> Double {**

**return logFunction(number)**

**}**

**}**

**ValidationModule\Validation.swift**

**import Foundation**

**public struct Validation {**

**public static func validatePositiveNumber(\_ input: Double) -> Bool {**

**return input > 0**

**}**

**}**

**Скриншот работающей программы:**

****

**Рисунок 2 - Результат выполнения программы**

**Вывод: В этой лабораторной работе мы разработали два приложения с помощью языке программирования Swift. При разработке были использованы функции, замыкания и перечисления. Программа была разбита на несколько модулей с учетом логики работы.**