Министерство образования

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Отчет по лабораторной работе №4

«Использование языка программирования Swift: наследование, протоколы»

Выполнил: Веренич В.Г.

Студент группы 310902

Проверил: Усенко Ф.В.

Минск 2024

Цель работы: Выполнить разработку приложения с использованием языка программирования Swift: наследование, протоколы.

Задание 1:

- согласно варианта задания написать программу (Вариант = номер в списке подгруппы % 15 + 1);

- разбить функционал приложения на несколько пакетов придерживаясь логики;

- Каждый класс должен реализовывать протокол;

- Работа должна происходить не с экземплярами класса, а с экземплярами протокола;

- сделать валидацию всех вводимых значений.

Индивидуальное задание № 4: Создать класс для вычисления корней квадратного уравнения. Предусмотреть все возможные варианты. Листинг кода lib:

import Swift

import Foundation

public protocol QuadraticEquationSolver {

func solve(a: Double, b: Double, c: Double) throws -> [Double]

}

public class QuadraticSolver: QuadraticEquationSolver {

public func solve(a: Double, b: Double, c: Double) throws -> [Double] {

guard a != 0 else {

throw EquationError.notQuadratic

}

let discriminant = b \* b - 4 \* a \* c

// Обработка вариантов

if discriminant > 0 {

let root1 = (-b + sqrt(discriminant)) / (2 \* a)

let root2 = (-b - sqrt(discriminant)) / (2 \* a)

return [root1, root2]

} else if discriminant == 0 {

let root = -b / (2 \* a)

return [root]

} else {

throw EquationError.noRealRoots

}

}

}

public enum EquationError: Error {

case invalidInput(String) // Некорректный ввод

case notQuadratic // Уравнение не является квадратным

case noRealRoots // Нет вещественных корней

}

public class InputValidator {

public static func validate(input: String) throws -> Double {

guard let value = Double(input) else {

throw EquationError.invalidInput("Ошибка: Введите числовое значение вместо '\(input)'.")

}

return value

}

}

Листинг кода main:

import Foundation

import lib

func main() {

let solver: QuadraticEquationSolver = QuadraticSolver()

do {

print("Enter odds a:")

let a = try InputValidator.validate(input: readLine() ?? "")

print("Enter odds b:")

let b = try InputValidator.validate(input: readLine() ?? "")

print("Enter odds c:")

let c = try InputValidator.validate(input: readLine() ?? "")

let roots = try solver.solve(a: a, b: b, c: c)

if roots.isEmpty {

print("The equation has no real roots.")

} else {

print("Roots of the equation: \(roots.map { String(format: "%.2f", $0) }.joined(separator: ", "))")

}

} catch EquationError.invalidInput(let message) {

print(message)

} catch EquationError.notQuadratic {

print("Error: The equation is not quadratic (a = 0).")

} catch EquationError.noRealRoots {

print("The equation has no real roots.")

} catch {

print("An unknown error occurred: \(error)")

}

}

main();

Результат работы программы представлен на рисунке 1:

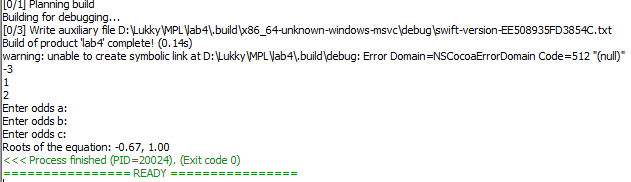


Рисунок 1