Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Кафедра инженерной психологии и эргономики

Современные языки программирования

Отчет по лабораторной работе №4

«Использование языка программирования Swift: наследование, протоколы»

Выполнил: Богуш А. В.

Студент группы 310901

Преподаватель: Усенко Ф. В.

Минск 2024

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

**Вариант 1**: Класс рациональных дробей. класс, представляющий рациональную дробь (num – числитель, den – знаменатель). Класс содержит методы умножения и деления (дроби на дробь и дроби на целое число). Создать протокол, который служит для создания случайной дроби из заданного диапазона целых. Добавить две различные реализации протокола.

class RationalFraction {

    var num: Int

    var den: Int

    init(numerator: Int, denominator: Int) {

        num = numerator

        den = denominator

    }

    func multiply(with ratFran: RationalFraction) -> RationalFraction {

        return RationalFraction(numerator: self.num \* ratFran.num, denominator: self.den \* ratFran.den)

    }

    func devide(by ratFran: RationalFraction) -> RationalFraction {

        return RationalFraction(numerator: self.num \* ratFran.den, denominator: self.den \* ratFran.num)

    }

    func multiply(with number: Int) -> RationalFraction{

        return RationalFraction(numerator: self.num \* number, denominator: self.den)

    }

    func devide(by number: Int) -> RationalFraction {

        return RationalFraction(numerator: self.num, denominator: self.den \* number)

    }

}

protocol RandomRationalFraction {

    func generateRandomFraction(from min: Int, to max: Int) -> RationalFraction

}

class FirstRandomizer: RandomRationalFraction {

    func generateRandomFraction(from min: Int, to max: Int) -> RationalFraction {

        let num = Int.random(in: min...max)

        let den = Int.random(in: min...max)

        return RationalFraction(numerator: num, denominator: den)

    }

}

class SecondRandomizer: RandomRationalFraction {

    func generateRandomFraction(from min: Int, to max: Int) -> RationalFraction {

        var num = Int.random(in: min...max)

        var den = Int.random(in: min...max)

        repeat {

            if (num/den >= min && num/den <= max) {

                break

            }

            num = Int.random(in: min...max)

            den = Int.random(in: min...max)

        } while (true)

        return RationalFraction(numerator: num, denominator: den)

    }

}

repeat {

    print("Выберите действие:")

    print("1. Умножить дробь на дробь")

    print("2. Умножить дробь на число")

    print("3. Делить дробь на дробь")

    print("4. Делить дробь на число")

    print("5. Первый генератор случайной дроби")

    print("6. Второй генератор случайной дроби")

    print("7. Выход")

    var number : Int = 0

    if let input = readLine(), let helper = Int(input), helper >= 1, helper <= 7 {

        number = helper

    } else {

        print("Ошибка: введите корректное число.")

        continue

    }

    if number == 1 {

        var num1: Int = 0

        var num2: Int = 0

        var den1: Int = 0

        var den2: Int = 0

        repeat {

            print("Введите числитель первой дроби: ")

            if let input = readLine(), let helper = Int(input) {

                num1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите знаменатель первой дроби: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                den1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите числитель второй дроби: ")

            if let input = readLine(), let helper = Int(input) {

                num2 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите знаменатель второй дроби: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                den2 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        let fraction1 = RationalFraction(numerator: num1, denominator: den1)

        let fraction2 = RationalFraction(numerator: num2, denominator: den2)

        let newfraction = fraction1.multiply(with: fraction2)

        print("\(num1)/\(den1) \* \(num2)/\(den2) = \(newfraction.num)/\(newfraction.den)")

    }

    if number == 2 {

        var num1: Int = 0

        var den1: Int = 0

        var integer: Int = 0

        repeat {

            print("Введите числитель дроби: ")

            if let input = readLine(), let helper = Int(input) {

                num1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите знаменатель дроби: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                den1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите целое число: ")

            if let input = readLine(), let helper = Int(input) {

                integer = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        let fraction1 = RationalFraction(numerator: num1, denominator: den1)

        let newfraction = fraction1.multiply(with: integer)

        print("\(num1)/\(den1) \* \(integer) = \(newfraction.num)/\(newfraction.den)")

    }

    if number == 3 {

        var num1: Int = 0

        var num2: Int = 0

        var den1: Int = 0

        var den2: Int = 0

        repeat {

            print("Введите числитель первой дроби: ")

            if let input = readLine(), let helper = Int(input) {

                num1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите знаменатель первой дроби: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                den1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите числитель второй дроби: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                num2 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите знаменатель второй дроби: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                den2 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        let fraction1 = RationalFraction(numerator: num1, denominator: den1)

        let fraction2 = RationalFraction(numerator: num2, denominator: den2)

        let newfraction = fraction1.devide(by: fraction2)

        print("\(num1)/\(den1) / \(num2)/\(den2) = \(newfraction.num)/\(newfraction.den)")

    }

    if (number == 4) {

        var num1: Int = 0

        var den1: Int = 0

        var integer: Int = 0

        repeat {

            print("Введите числитель дроби: ")

            if let input = readLine(), let helper = Int(input) {

                num1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите знаменатель дроби: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                den1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите целое число: ")

            if let input = readLine(), let helper = Int(input), helper != 0 {

                integer = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        let fraction1 = RationalFraction(numerator: num1, denominator: den1)

        let newfraction = fraction1.devide(by: integer)

        print("\(num1)/\(den1) / \(integer) = \(newfraction.num)/\(newfraction.den)")

    }

    if (number == 5) {

        var num1: Int = 0

        var num2: Int = 0

        repeat {

            print("Введите первое число: ")

            if let input = readLine(), let helper = Int(input) {

                num1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите второе число: ")

            if let input = readLine(), let helper = Int(input) {

                num2 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        if (num1 > num2) {

            let generator = FirstRandomizer()

            let fraction = generator.generateRandomFraction(from: num2, to: num1)

            print("Сгенерированная дробь: \(fraction.num)/\(fraction.den)")

        } else if (num1 < num2) {

            let generator = FirstRandomizer()

            let fraction = generator.generateRandomFraction(from: num1, to: num2)

            print("Сгенерированная дробь: \(fraction.num)/\(fraction.den)")

        } else {

            print("Ошибка: неправильно указан промежуток.")

        }

    }

    if number == 6 {

        var num1: Int = 0

        var num2: Int = 0

        repeat {

            print("Введите первое число: ")

            if let input = readLine(), let helper = Int(input) {

                num1 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        repeat {

            print("Введите второе число: ")

            if let input = readLine(), let helper = Int(input) {

                num2 = helper

                break

            } else {

                print("Ошибка: введите целое число.")

            }

        } while (true)

        if (num1 > num2) {

            let generator = SecondRandomizer()

            let fraction = generator.generateRandomFraction(from: num2, to: num1)

            print("Сгенерированная дробь: \(fraction.num)/\(fraction.den)")

        } else if (num1 < num2) {

            let generator = SecondRandomizer()

            let fraction = generator.generateRandomFraction(from: num1, to: num2)

            print("Сгенерированная дробь: \(fraction.num)/\(fraction.den)")

        } else {

            print("Ошибка: неправильно указан промежуток.")

        }

    }

    if number == 7 {

        break

    }

} while (true)

Результат работы программы представлен на рисунках 1, 2, 3.

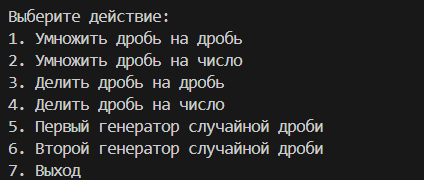


Рисунок 1 – Результат работы программы.

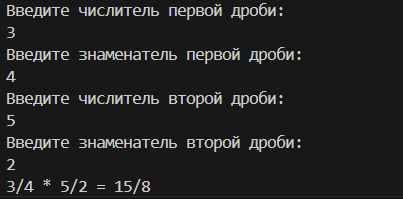


Рисунок 2 – Результат работы программы.

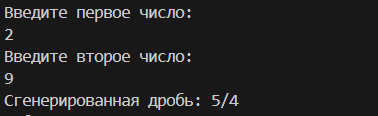


Рисунок 3 – Результат работы программы.

**Выводы**: ознакомился с наследованием и протоколами на Swift. На языке есть возможность множественного наследования, возможность расширения классов (extension). Протокол определяет образец методов, свойств или другие требования, которые соответствуют определенному конкретному заданию или какой-то функциональности. Существует множество подходов при работе с протоколами. Например, использование делегирования, опциональных требований протокола, расширение протокола, наследование протоколов.