Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Дисциплина: Современные языки программирования

# ЛАБОРАТОРНАЯ РАБОТА № 2

Вариант № 4

Выполнила:

Власенко Д.Р. гр. 310901

Проверил:

Усенко Ф.В.

Минск 2024

**Задание 1:** Среди всех четырехзначных номеров машин, определите количество номеров, содержащих три или более одинаковых цифры.

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

а) для вычитания целых чисел;

б) для вычитания комплексных чисел

func check(num: String) -> Bool {

guard num.count == 4 else { return false }

let digits = num.map { Int(String($0))! } // преобразуем строку в массив цифр

var counts: [Int: Int] = [:] // создаем пустой словарь

for digit in digits {

counts[digit, default: 0] += 1 //проверяем кол-во вхождений каждой цифры

}

return counts.contains { $0.value >= 3 }// вернет true если 3 и более один цифр

}

func subtraction(int1: Int, int2: Int) -> Int {

return int1 - int2

}

func subtraction(int1: Int, int2: Int, int3: Int, int4: Int) -> String {

let sub1 = int1 - int3

let sub2 = int2 - int4

return "\(sub1)+\(sub2)\*i"

}

func main() {

var time = true

while time {

print("======================")

print("Выберите пункт меню:")

print("1. Определить число с 3 и более одинаковыми числами")

print("2. Вычитание целых чисел")

print("3. Вычитание комплексных чисел")

print("4. Выйти из программы")

print("======================")

guard let point = readLine() else { continue } // считываем пункт меню

switch point {

case "1":

var count = 0 // кол-во чисел с 3 и более один. цифрами

var numbers = [String]() //массив для хранения всех чисел

while true {

print("Введите четырехзначное число (или 0 для завершения):")

guard let input = readLine() else { continue } // считываем четырхезначное число

if input == "0" {

break

}

if input.count == 4, let \_ = Int(input) {

numbers.append(input)

} else {

print("Некорректный ввод")

}

}

for number in numbers {

if check(num: number) { // вызов функции для проверки числа

count+=1

}

}

print("Количество четырехзначных чисел с тремя и более одинаковыми цифрами: \(count)")

case "2":

print("Введите уменьшаемое целое число")

guard let num1String = readLine(), let num1 = Int(num1String) else {

print("Некорректный ввод")

break

}

print("Введите вычитаемое целое число")

guard let num2String = readLine(), let num2 = Int(num2String) else {

print("Некорректный ввод")

break

}

let sub = subtraction(int1: num1, int2: num2)

print("Ответ: \(sub)")

case "3":

print("Введите уменьшаемое комплексное число: а1 + b1\*i")

print("Введите а1:")

guard let a1String = readLine(), let a1 = Int(a1String) else {

print("Некорректный ввод")

break

}

var b1: Int = 0

while b1 == 0 {

print("Введите b1:")

guard let b1String = readLine(), let b1Value = Int(b1String) else {

print("Некорректный ввод")

break

}

b1 = b1Value

if b1 == 0 {

print("Если b = 0, то число уже не является комплексным")

}

}

print("Введите вычитаемое комплексное число: а2 + b2\*i")

print("Введите а2:")

guard let a2String = readLine(), let a2 = Int(a2String) else {

print("Некорректный ввод")

break

}

var b2: Int = 0

while b2 == 0 {

print("Введите b2:")

guard let b2String = readLine(), let b2Value = Int(b2String) else {

print("Некорректный ввод")

break

}

b2 = b2Value

if b2 == 0 {

print("Если b2 = 0, то число уже не является комплексным")

}

}

let sub2 = subtraction(int1: a1, int2: b1, int3: a2, int4: b2)

print("Ответ: \(sub2)")

case "4":

time = false

default:

print("Некорректный ввод")

}

}

}

main()

Результат работы программы представлен на рисунках 1-3.

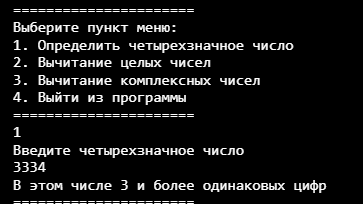


Рисунок 1 – Результат работы программы

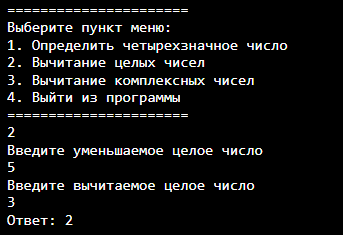


Рисунок 2 – Результат работы программы

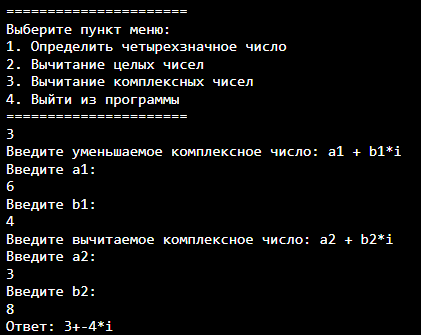


Рисунок 3 – Результат работы программы

Ответы на контрольные вопросы:

1. Что такое функция?

Это самодостаточный блок кода, который выполняет определенную задачу; она может принимать параметры и возвращать значение.

1. Что такое вариативные параметры?

Вариативные параметры в Swift позволяют функции принимать произвольное количество аргументов одного типа, используя символ ..., что делает обработку массивов значений более простой и удобной.

1. Что такое сквозные параметры?

Сквозные параметры (или escaping параметры) в Swift — это параметры замыканий, которые могут сохраняться и использоваться после завершения выполнения функции, что позволяет им продолжать работать асинхронно.

**Вывод:**С использованием языка программирования Swift создали программу, реализующую анализ цифр в номере машины, а также для вычитания целых и комплексных чисел. Также: сформировали умение разрабатывать программы на языке Swift.