БГУИР

Кафедра ЭВМ

Отчет по практическому занятию № 1

Тема: «Создание простых S-проектов на основе классов»

Выполнили:

студенты гр. №150502

Альхимович Н.Г.

Скалозуб К.А.

Проверила:

Герман Ю.О.

Минск

2023

**1 ЦЕЛЬ РАБОТЫ**

Изучить технику использования классов в Scala.

**2 КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Напишем следующий пример с классом в текстовом файле lab4.scala

class Person {

def hello(name: String): Unit = println(s"Hello, $name!")

}

object Main22 {

def main(args: Array[String]): Unit = {

val x = new Person()

x.hello("Alice")

x.hello("Bob")

}

}

Здесь объявлен класс Person с единственным методом (функцией) hello. Аргументом функции является строковая переменная name. Функция выводит это значение на экране в операторе:

println(s"Hello, $name!")

Не забываем перед телом функции ставить знак равно:

= println(s"Hello, $name!")

Хорошим стилем будет заключить тело функции в фигурные скобки, да еще поставить точку с запятой в конце оператора:

class Person {

def hello(name: String): Unit = println(s"Hello, $name!")

}

object Main22 {

def main(args: Array[String]): Unit = {

val x = new Person()

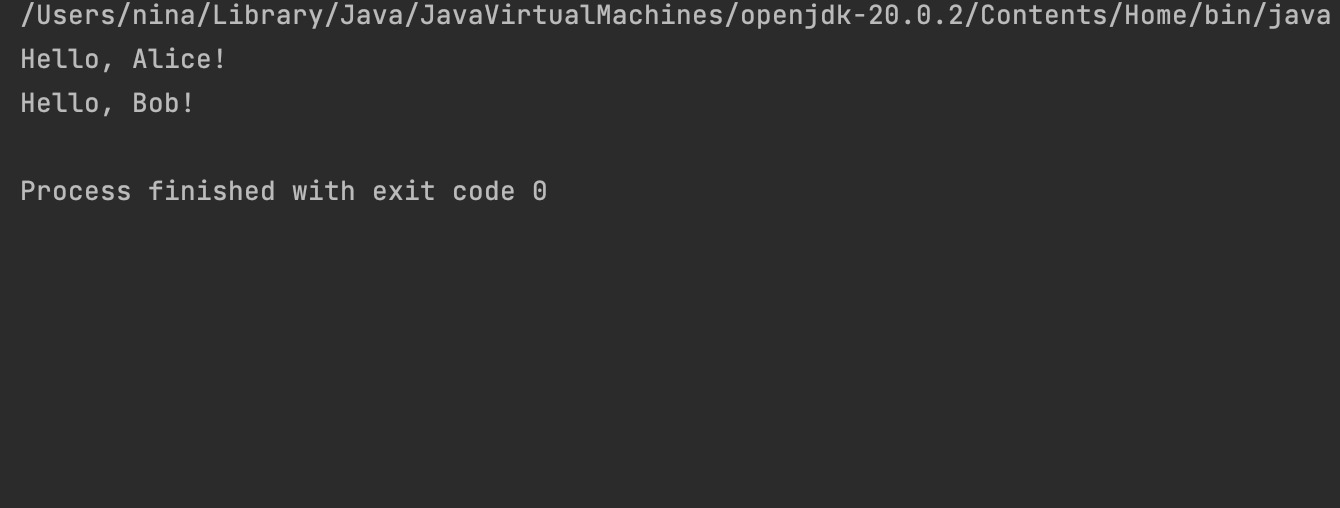
x.hello("Alice")

x.hello("Bob")

}

}

Заметим, что в операторе println(s"Hello, $name!"); терм $name используется для подстановки значения переменной name (предшествует символ доллара).



Теперь мы хотим ввести с клавиатуры два целых числа и найти их наибольший целый (общий) делитель.

object Numbers {

def gcd(a: Int, b: Int): Int = {

if (b == 0) a else gcd(b, a % b)

}

}

object Main22 {

def main(args: Array[String]): Unit = {

print("Enter the first number: ")

val a = scala.io.StdIn.readInt()

print("Enter the second number: ")

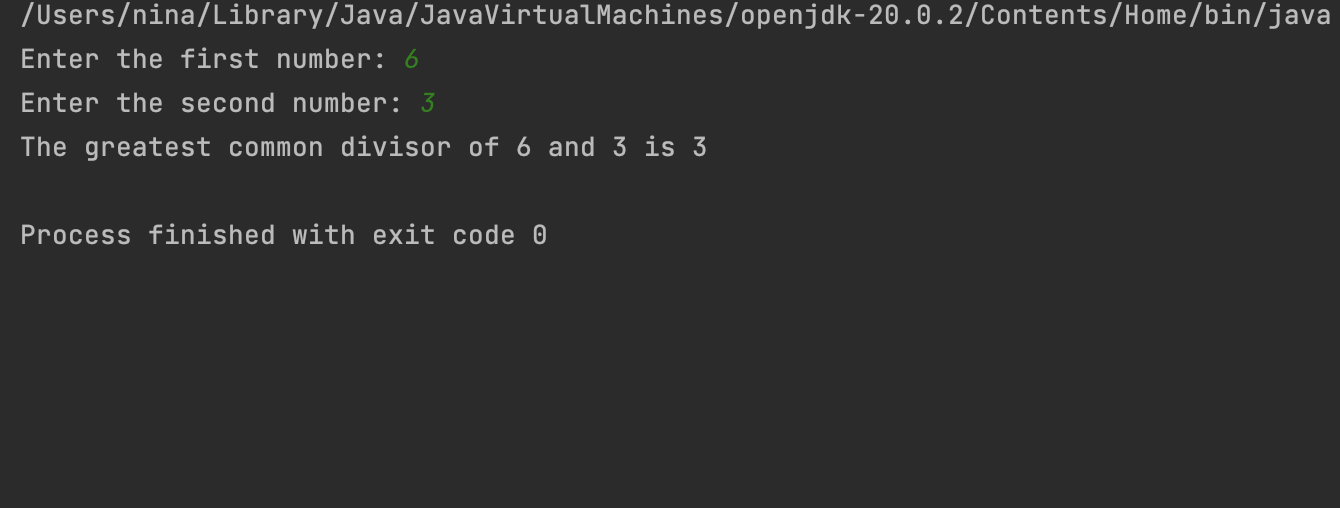
val b = scala.io.StdIn.readInt()

val gcdValue = Numbers.gcd(a, b)

println(s"The greatest common divisor of $a and $b is $gcdValue")

}

}



Читаем целые числа с клавиатуры таким образом

val a = scala.io.StdIn.readInt()

Сделаем модификации. Вводим числа как строки и конвертируем в числа:

object Numbers {

def gcd(a: Int, b: Int): Int = {

if (b == 0) a else gcd(b, a % b)

}

}

object Main22 {

def main(args: Array[String]): Unit = {

print("Enter the first number: ")

val a = scala.io.StdIn.readLine().toInt

print("Enter the second number: ")

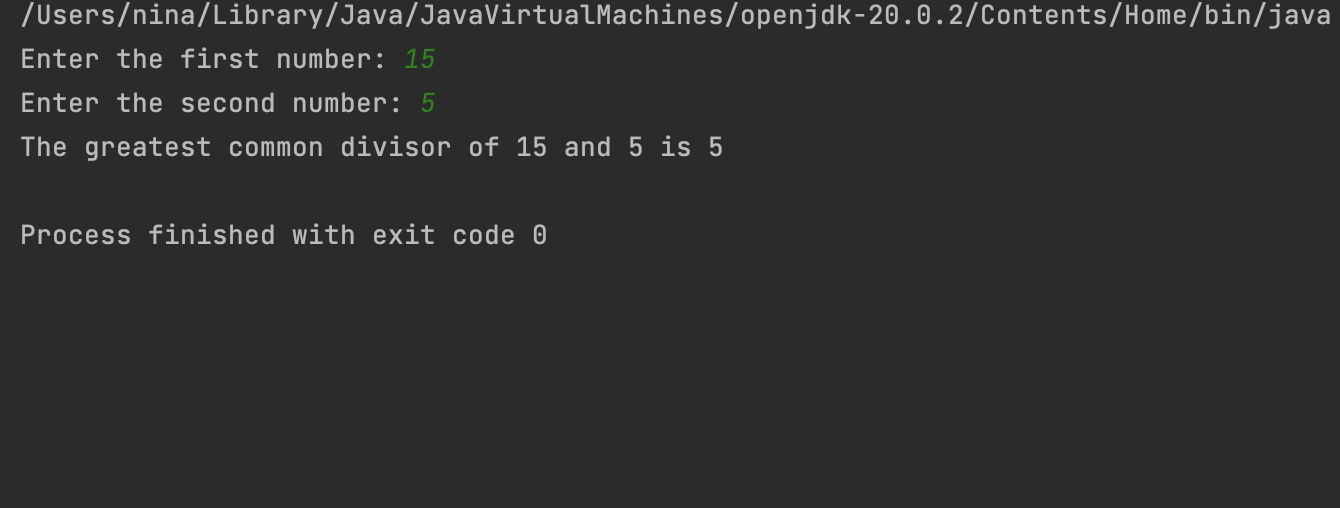
val b = scala.io.StdIn.readLine().toInt

val gcdValue = Numbers.gcd(a, b)

println(s"The greatest common divisor of $a and $b is $gcdValue")

}

}



Определить сумму цифр в записи целого числа.

object Main22 {

def main(args: Array[String]): Unit = {

print("Enter the number: ")

val a = scala.io.StdIn.readInt

var sum = 0

for (digit <- a.toString) {

sum += digit.asDigit

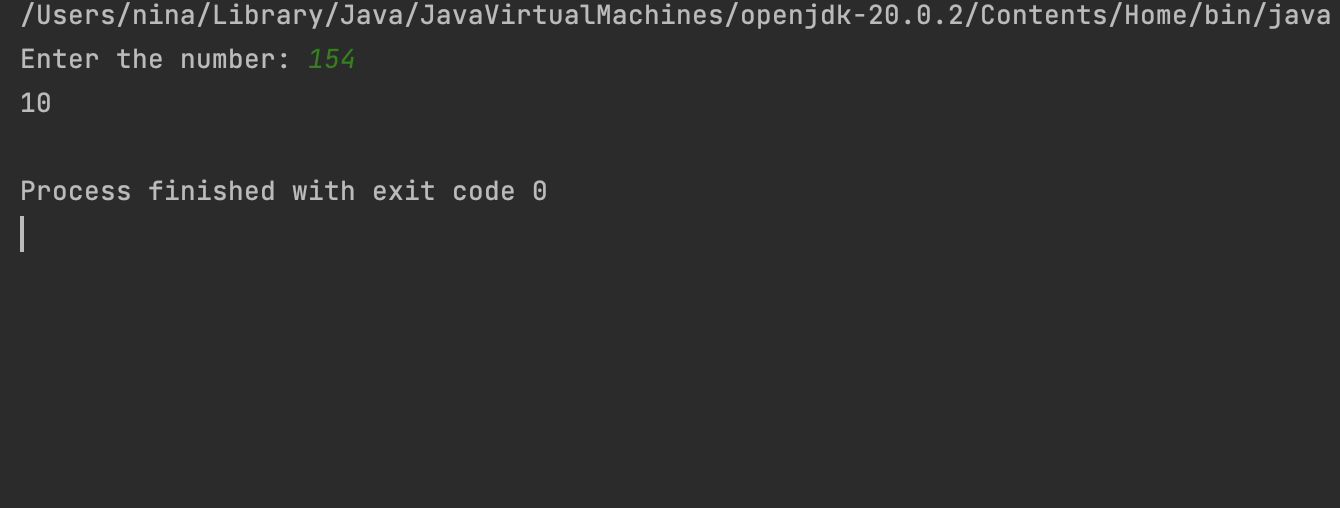
}

// Print the sum

println(sum)

}

}



Сформировать целое число из первых трех цифр шестизначного числа.

object Main22 {

def main(args: Array[String]): Unit = {

print("Enter the number: ")

val number = scala.io.StdIn.readInt

try {

val numberString = number.toString

val firstThreeDigits = if (numberString.length >= 3 && number >= 0) {

numberString.substring(0, 3).toInt

} else {

throw new Exception("Number does not have at least three digits or is negative")

}

println(firstThreeDigits)

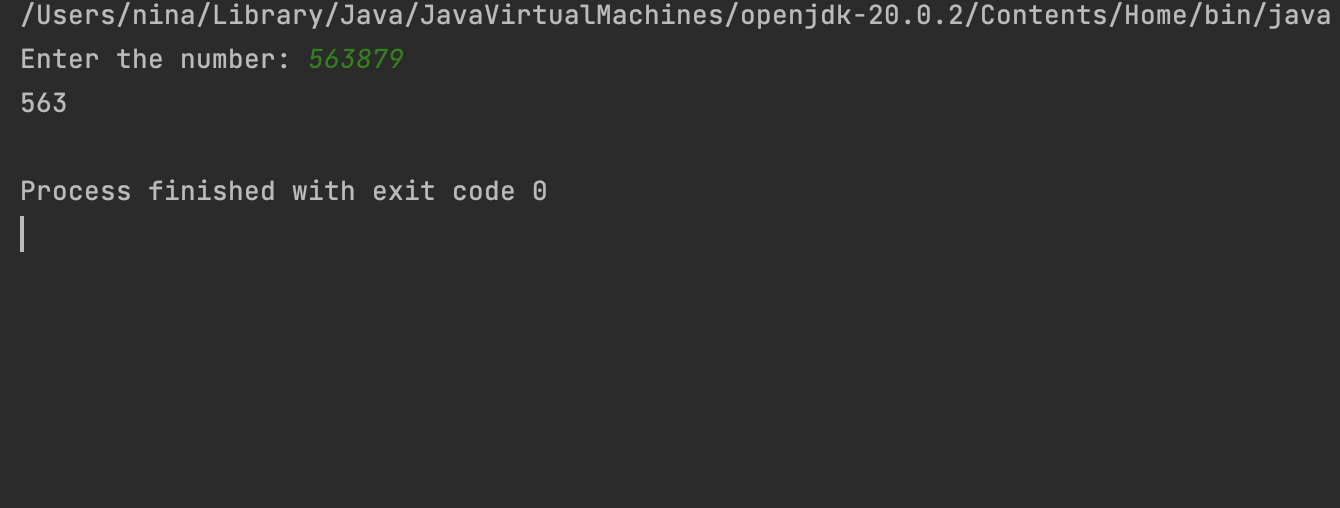
} catch {

case e: Exception => println(e.getMessage)

}

}

}



Записать число в обратном порядке цифр.

object Main5 {

def main(args: Array[String]): Unit = {

print("Enter the number: ")

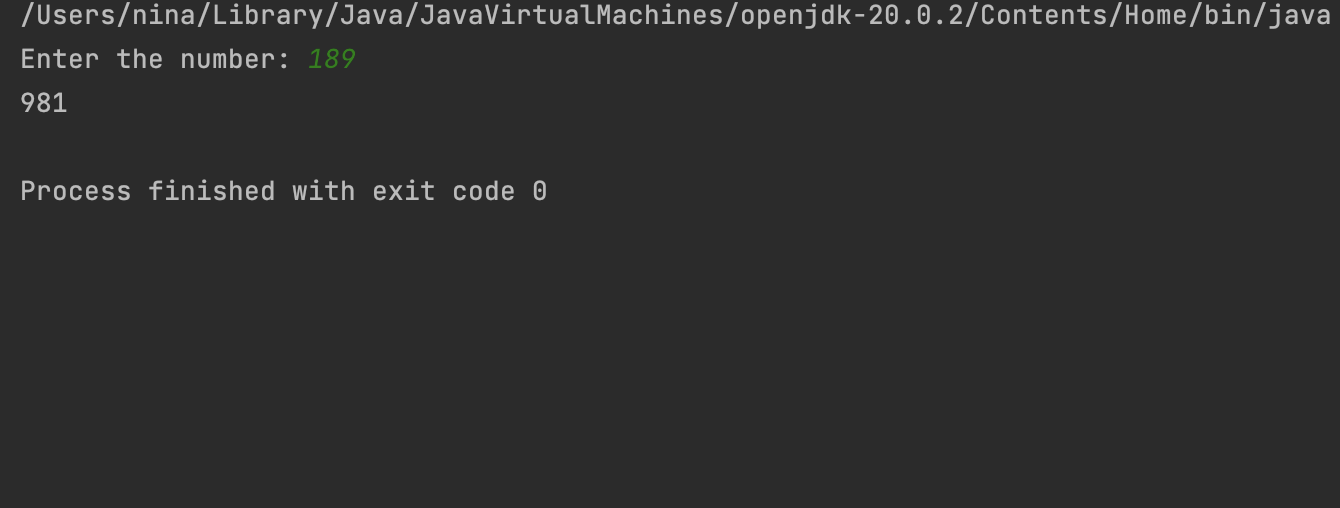
val number = scala.io.StdIn.readInt

val reversedNumber = number.toString.reverse.toInt

println(reversedNumber)

}

}

****

**3 ХОД РАБОТЫ**

**3.1 Вариант 1**

Построить генератор случайных чисел по следующей схеме. Вводите 10-значное число с клавиатуры. Формируете два новых целых числа A и B: первое A состоит из первых пяти цифр, второе B – из последних цифр введенного числа, начиная с шестой. Перемножаете числа A и B друг на друга. Первые три цифры результата С, будучи поделенными на 1000, дают первое случайное число REZ. Чтобы сформировать следующее случайное число, прибавляете к Z число С и повторяете процесс. Формируете два новых целых числа A и B: первое A состоит из первых пяти цифр, второе B – из последних цифр числа Z, начиная с шестой. Перемножаете числа A и B друг на друга. Первые три цифры результата С, будучи поделенными на 1000, дают второе случайное число. Чтобы сформировать следующее случайное число, прибавляете к Z число С и повторяете процесс. Сформируйте 5 случайных чисел.

Если z отрицателен, то это значит, что надо заменить первую цифру единицы на ноль и убрать знак “минус”, кроме того, нужно поддерживать длину строки не меньше 10 символов. Сказанное, демонстрируется следующим примером:

object Main234 {

def main(args: Array[String]): Unit = {

print("Enter the number: ")

var numberString =""

var a =0

var b=0

var c=0

var rez: Float =0.0

var z=0

numberString = scala.io.StdIn.readLine()

if ( (numberString.length() <= 10)) {

println("Incorrect number");

sys.exit(0) }

for (i <- 1 to 10) {

a = numberString.substring(0, 5).toInt

//println(numberString.length());

b= numberString.substring(5, 10).toInt

c= a\*b

rez=c.toString().substring(0,3).toFloat / 1000

z+=c

println(a);

println(b);

println(c);

println(rez);

println("z="+ z)

numberString= numberString.substring(1,numberString.length()-1)

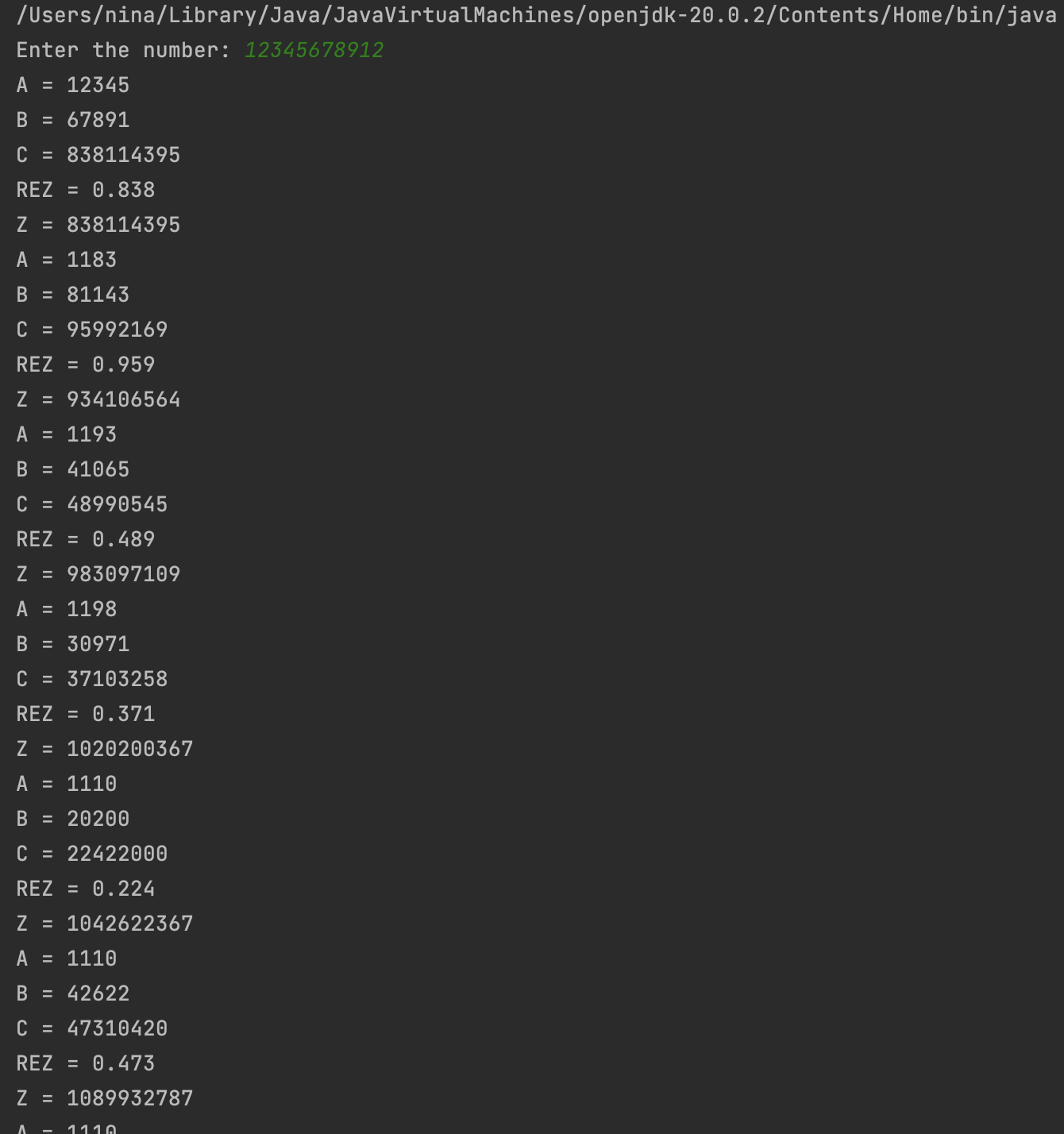
numberString="011"+z

}

}

}

Результат работы программы приведен на рисунке:

****

**3.2 Вариант 2**

Использовать предыдущий пример, но только число А получается из цифр, стоящих на четных позициях, а В – на нечетных.

Код программы:

object Main22 {

def substringFromSymbolsAtEvenPositions(str: String): String = {

val result = new StringBuilder

for (i <- str.indices if i % 2 == 1) {

result.append(str(i))

}

result.toString()

}

def substringFromSymbolsAtOddPositions(str: String): String = {

val result = new StringBuilder

for (i <- str.indices if i % 2 == 0) {

result.append(str(i))

}

result.toString()

}

def main(args: Array[String]): Unit = {

var numberString = ""

var a = 0

var b = 0

var c = 0

var rez: Float = 0

var z = 0

print("Enter the number: ")

numberString = scala.io.StdIn.readLine()

if ((numberString.length() <= 10)) {

println("Incorrect number");

sys.exit(0)

}

for (i <- 1 to 5) {

println("\n" + i + ".")

if (numberString.length > 10) {

numberString = numberString.substring(0, 10)

println(numberString)

}

a = substringFromSymbolsAtEvenPositions(numberString).toInt

b = substringFromSymbolsAtOddPositions(numberString).toInt

println("A = " + a);

println("B = " + b);

c = a\*b

println("C = " + c);

rez = c.toString().substring(0,3).toFloat / 1000

println("REZ = " + rez);

z += c

println("Z" + i + " = " + z);

numberString = numberString.substring(1, numberString.length()-1)

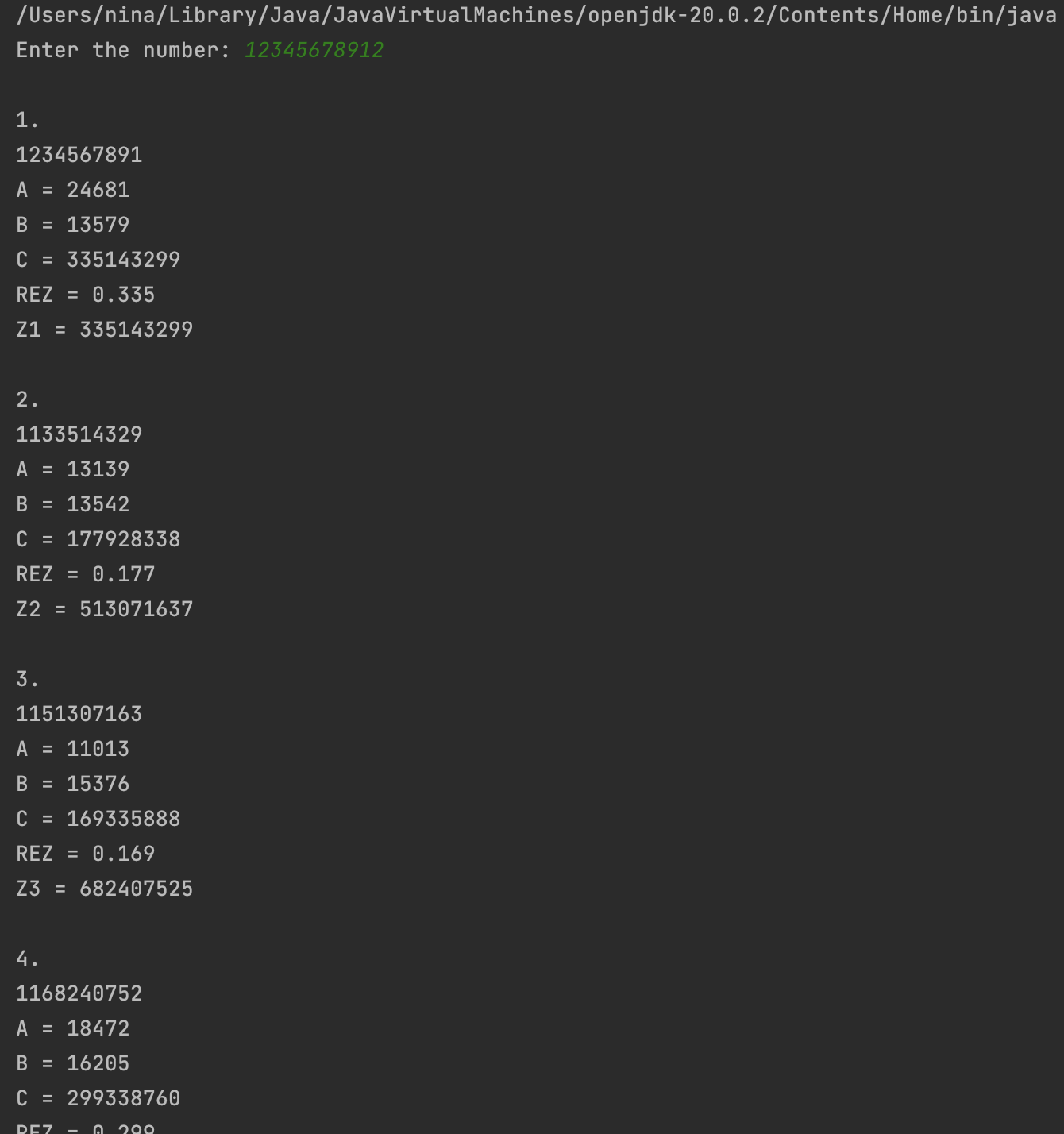
numberString = "11" + z

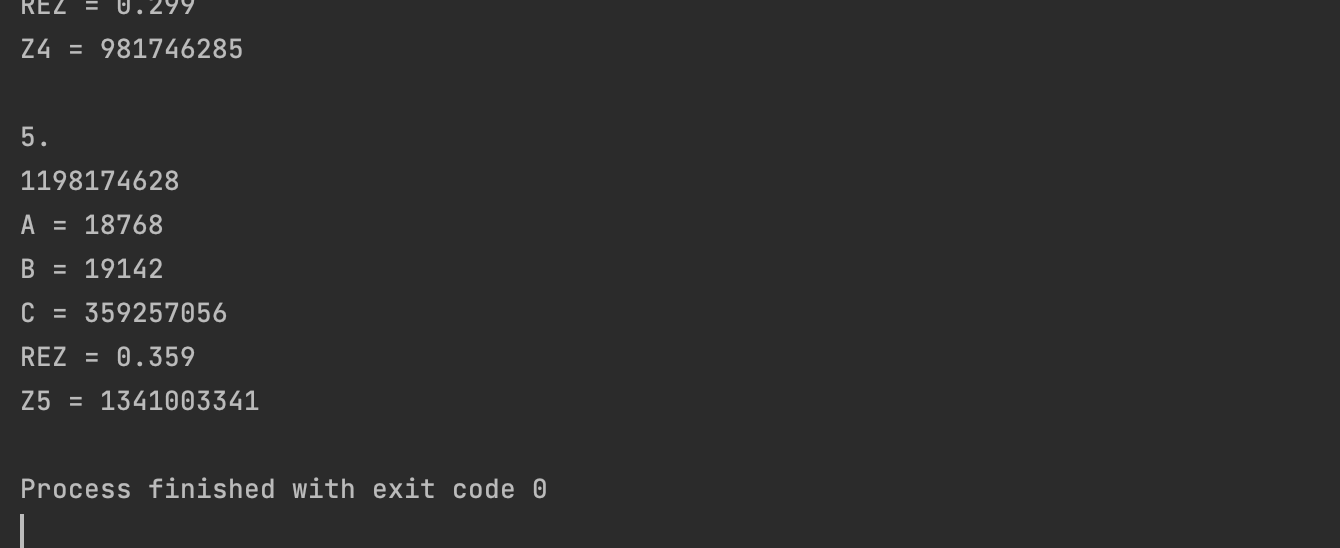
}

}

}

Результат работы программы приведен на рисунке:





**4 ВЫВОДЫ**

В процессе выполнения лабораторной работы была изучена техника использования классов в Scala и на основе полученных знаний написан простой S-проект, осуществляющий генерацию случайных чисел.