Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

Информатики и радиоэлектроники»

Учебная дисциплина: «Программирование мобильных информационных систем»

Отчет

по выполнению лабораторной работы «Основы языка Kotlin»

Выполнил студент: гр. 210101 Савеня П.В.

Проверил: Усенко Ф.В.

Минск 2024

**Цель работы:** изучить основные конструкции языка Kotlin, научиться работать с типами данных, операциями ввода-вывода, а также условными операторами. Закрепить полученные знания через выполнение задач, требующих практического применения теории.

**Задание:** Напишите программу, которая принимает число от пользователя и разлагает его на простые множители. Программа должна учитывать случаи, когда число является простым.

**Оснащение работы:** техническое задание, технический проект, Kotlin, Intellij IDEA.

**Исходный код:**

import java.math.BigInteger

import kotlin.math.sqrt

fun sieveOfEratosthenes(limit: Int): List<Int> {

if (limit < 2) return emptyList()

val isPrime = BooleanArray(limit + 1) { true }

isPrime[0] = false

isPrime[1] = false

for (i in 2..sqrt(limit.toDouble()).toInt()) {

if (isPrime[i]) {

for (j in i \* i..limit step i) {

isPrime[j] = false

}

}

}

return isPrime.withIndex().filter { it.value }.map { it.index }

}

fun sqrtBigInteger(num: BigInteger): BigInteger {

var low = BigInteger.ZERO

var high = num

while (low <= high) {

val mid = (low + high) / BigInteger.TWO

val midSquared = mid \* mid

when {

midSquared == num -> return mid

midSquared < num -> low = mid + BigInteger.ONE

else -> high = mid - BigInteger.ONE

}

}

return high

}

fun divideStringByInt(num: BigInteger, divisor: Int): BigInteger {

return num.divide(BigInteger.valueOf(divisor.toLong()))

}

fun isDivisibleByInt(num: BigInteger, divisor: Int): Boolean {

return num.mod(BigInteger.valueOf(divisor.toLong())) == BigInteger.ZERO

}

fun primeFactorization(numStr: String) {

var num = BigInteger(numStr)

val limit = sqrtBigInteger(num)

val primes = sieveOfEratosthenes(limit.toInt())

print("Prime factors of $numStr: ")

for (prime in primes) {

while (isDivisibleByInt(num, prime)) {

print("$prime ")

num = divideStringByInt(num, prime)

}

}

if (num > BigInteger.ONE) {

print("$num ")

}

}

fun main() {

print("Enter Number to Decompose It Into Prime Factors: ")

val input = readlnOrNull()

if (input.isNullOrEmpty() || input.any { it !in '0'..'9' }) {

println("Invalid Number Entered!")

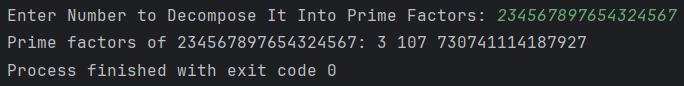
}

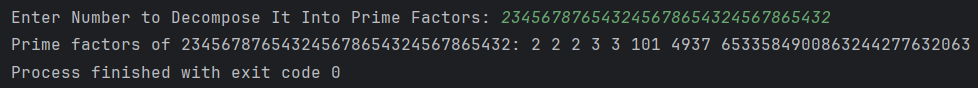
else {

primeFactorization(input)

}

}





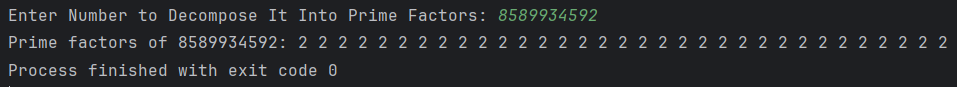


Рисунок 1 – Результаты исполнения кода программы

**Вывод:** в ходе выполнения работы были изучены основные конструкции языка Kotlin, такие как типы данных, операции ввода-вывода, условные операторы, а также работа с переменными и операциями присваивания.