

# **Информационная безопасность. Отчет по лабораторной работе № 6**

**Разложение чисел на множители**

Мухамеджанов Исматулло Иззатуллоевич

# Содержание

1	Цель работы	5
2	Указание к работе	6
3	Выполнение лабораторной работы	7
4	Выводы	9
5	Список литературы	10

# List of Figures

3.1	Программа (1)	. . . . .	7
3.2	Программа (2)	. . . . .	8

## List of Tables

# 1 Цель работы

Освоить на практике применение разложения числа на множители.

## **2 Указание к работе**

Метод Полларда Факторизация числа

### 3 Выполнение лабораторной работы

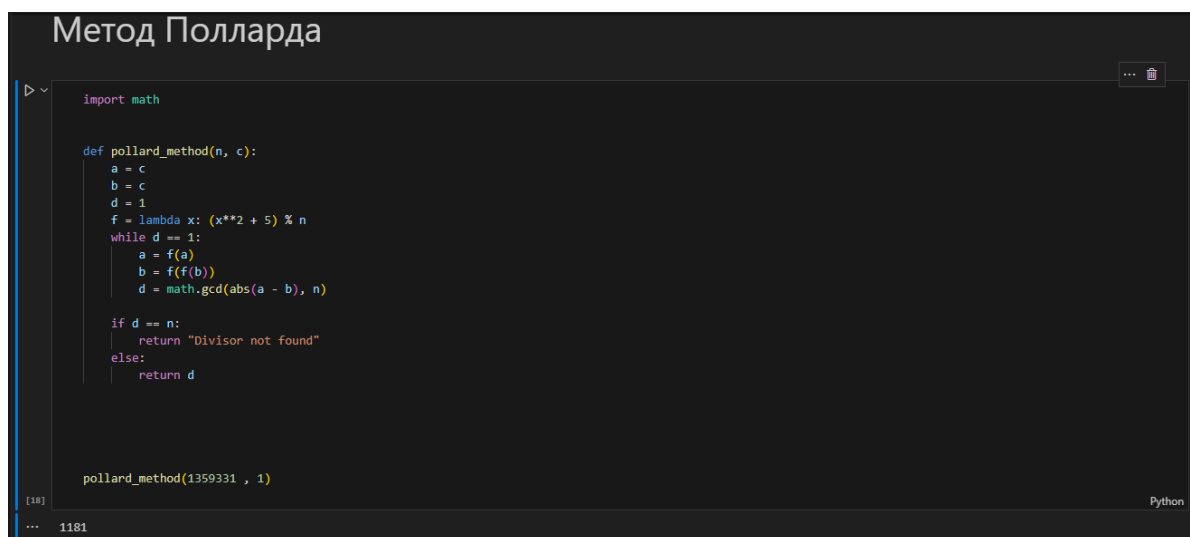
#### 1. Метод Полларда

Вход. Число  $n$ , начальное значение  $c$ , функция  $f$ , обладающая сжимающими свойствами. Выход. Нетривиальный делитель числа  $n$ .

Положить  $a \leftarrow c$ ,  $b \leftarrow c$ . Вычислить  $a \leftarrow f(a) \pmod n$ ,  $b \leftarrow f(b) \pmod n$ . Найти  $d \leftarrow \text{НОД}(a - b, n)$ . Если  $1 < d < n$ , то положить  $r \leftarrow d$  и результат:  $r$ . При  $d = n$  результат: Делитель не найден»; при  $d = 1$  вернуться на шаг 2.

#### 2. Факторизация числа

Метод квадратов. (Теорема Ферма о разложении) Для любого положительного нечетного числа  $n$  существует взаимно однозначное соответствие между множеством делителей числа  $n$ , не меньших, чем  $\sqrt{n}$ , и множеством пар  $\{s, t\}$  таких неотрицательных целых чисел, что  $n = s^2 - t^2$ .



```
import math

def pollard_method(n, c):
    a = c
    b = c
    d = 1
    f = lambda x: (x**2 + 5) % n
    while d == 1:
        a = f(a)
        b = f(f(b))
        d = math.gcd(abs(a - b), n)

    if d == n:
        return "Divisor not found"
    else:
        return d

pollard_method(1359331, 1)
```

[18] 1181 Python

Figure 3.1: Программа (1)

# Факторизация Числа

```
def factors(n):
    k = round(math.sqrt(n))
    number_list = []
    for i in range(k, n+1):
        if n % i == 0:
            number_list.append(i)

    for j in number_list:
        print(f"{n} = {j}*{n//j}", end=" , откуда ")
        res = (j + n//j) // 2
        res_2 = int(math.sqrt(res**2 - n))
        print(f"s={res}, t={res_2} и {n}={res}^2-{res_2}^2")

factors(65)
```

[35] Python

... 65 = 13\*5, откуда s=9, t=4 и 65=9<sup>2</sup>-4<sup>2</sup>  
65 = 65\*1, откуда s=33, t=32 и 65=33<sup>2</sup>-32<sup>2</sup>

Figure 3.2: Программа (2)



## **4 Выводы**

Освоены методы разложения чисел на множители

## **5 Список литературы**

1. Методические материалы курса