

Разложение чисел на множители.Р - метод Полларда.

Меньшов Иван Сергеевич

15 декабря, 2021, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи

Цель лабораторной работы

Изучить разложение чисел на множители. Реализовать p -метод Полларда.

Выполнение лабораторной работы

Разложение чисел на множители

Задача разложения на множители — одна из первых задач, использованных для построения криптосистем с открытым ключом.

Задача разложения составного числа на множители формулируется следующим образом: для данного положительного целого числа n найти его каноническое разложение $n = p_1^{a_1} \dots p_n^{a_n}$, где p_i — попарно различные простые числа, $a_i > 1$.

На практике не обязательно находить каноническое разложение числа n . Достаточно найти его разложение на два нетривиальных сомножителя: $n = pq$, $1 < p < q < n$. Далее будем понимать задачу разложения именно в этом смысле.

Р - метод Полларда

Р - метод Полларда - алгоритм разработанный Джоном Поллардом для разложения числа n на множители. Данный алгоритм выглядит следующим образом:

Вход. Число n , начальное значение s , функция f , обладающая сжимающим свойством

Выход. Нетривиальный делитель числа n .

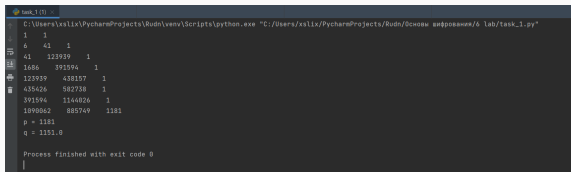
1. Положить $a = s$, $b = s$
2. Вычислить $a = f(a) \pmod{n}$, $b = f(f(b)) \pmod{n}$
3. Найти $d = \text{НОД}(a - b, n)$
4. Если $1 < d < n$, то положить $p = d$ и результат: p . При $d = n$ результат: 'Делитель не найден'. При $d = 1$ вернуться на шаг 2.

Контрольные пример - код

```
task_1.py
1 import math
2
3
4 def func(x, y):
5     return (x ** 2 + 5) % y
6
7
8 c = 1
9 a = c
10 b = c
11 n = 1359331
12 print(a, b, sep=' ')
13 while True:
14     a = func(a, n) % n
15     b = func(func(b, n), n) % n
16     d = math.gcd(a - b, n)
17     print(a, b, d, sep=' ')
18     if 1 < d < n:
19         p = d
20         print(f'p = {p}')
21         break
22     elif d == n:
23         print('Делитель не найден')
24
```

Figure 1: P - метод Полларда python

Контрольные пример - работа программы



```
C:\Users\xslix\PycharmProjects\Rudn\venv\Scripts\python.exe "C:/Users/xslix/PycharmProjects/Rudn/Основы шифрования/lab/task_1.py"
1 1
6 41 1
41 123939 1
1686 391596 1
123939 438157 1
438426 582738 1
391596 1144826 1
1890862 885749 1181
p = 1181
q = 1151.0

Process finished with exit code 0
```

Figure 2: P - метод Полларда работа алгоритма

Выводы

Мной было изучено разложение чисел на множители, а также реализован p - метод Полларда.