

Разложение чисел на множители

Гаглов Олел Мелорович

23 ноября, 2024, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи

Цель лабораторной работы

Изучение задачи разложения на множители, изучение p -алгоритма Поллрада.

Выполнение лабораторной работы

Задача разложения на простые множители

Разложение на множители — предмет непрерывного исследования в прошлом; и такие же исследования, вероятно, продолжатся в будущем. Разложение на множители играет очень важную роль в безопасности некоторых криптосистем с открытым ключом.

р-алгоритм Поллрада

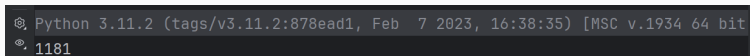
- Вход. Число n , начальное значение c , функция f , обладающая сжимающими свойствами.
 - Выход. Нетривиальный делитель числа n .
1. Положить $a = c, b = c$
 2. Вычислить $a = f(a)(\text{mod } n), b = f(b)(\text{mod } n)$
 3. Найти $d = \text{GCD}(a - b, n)$
 4. Если $1 < d < n$, то положить $p = d$ и результат: p . При $d = n$ результат: ДЕЛИТЕЛЬ НЕ НАЙДЕН. При $d = 1$ вернуться на шаг 2.

Сложность. Заметим, что этот метод требует сделать $B-1$ операций возведения в степень $a = a^e \bmod n$. Есть быстрый алгоритм возведения в степень, который выполняет это за $2 * \log_2 B$ операций. Метод также использует вычисления НОД, который требует n^3 операций. Мы можем сказать, что сложность — так или иначе больше, чем $O(B)$ или $O(2^n)$, где n_b — число битов в B . Другая проблема — этот алгоритм может заканчиваться сигналом об ошибке. Вероятность успеха очень мала, если B имеет значение, не очень близкое к величине \sqrt{n} .

Код алгоритма

```
1 usage
2 def f(x,n):
3     return (x*x+5)%n
4
5 2 usages
6 def Ext_Euclide(a,b):
7     if a==0:
8         return b,0,1
9     else:
10         r,x,y=Ext_Euclide(b%a,a)
11         return r,y-(b//a)*x,x
12
13 1 usage
14 def Polard(n:int,c:int,fn):
15     a=c
16     b=c
17     while True :
18         a=fn(a,n)
19         b=fn(fn(b,n),n)
20         d=Ext_Euclide(a-b,n)[0]
21         if 1<d<n:
22             return d
23         if d==n:
24             return -1
25         if d==1:
26             continue
27
28 if __name__=="__main__":
29     n=1359331
30     c=1
31     print(Polard(n,c,f))
```


Пример работы алгоритма



```
Python 3.11.2 (tags/v3.11.2:878ead1, Feb 7 2023, 16:38:35) [MSC v.1934 64 bit  
1181
```

Рис. 2: Работа алгоритма

Выводы

Изучили задачу разложения на множители и р-алгоритм Поллрада.