Разложение чисел на множители

Милёхин Александр НПМмд-02-21

Цель лабораторной работы

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

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.

p-алгоритм Полларда

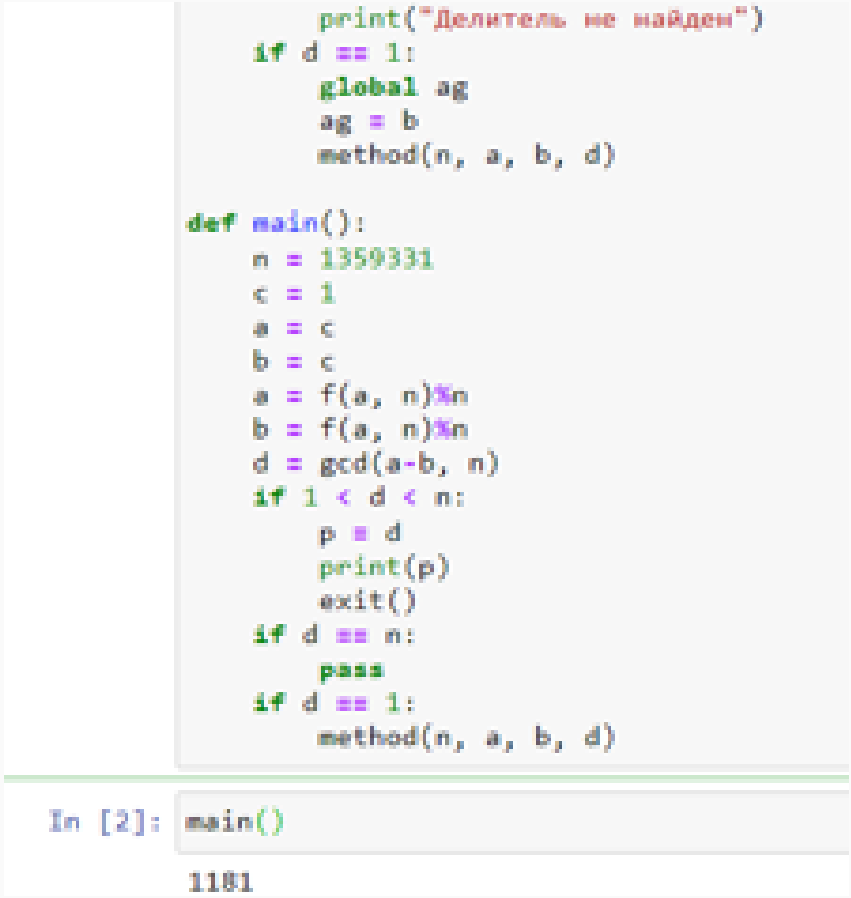
- Вход. Число n, начальное значение c, функция f, обладающая сжимающими свойствами.
- Выход. Нетривиальный делитель числа n.

1. Положить $a = c, b = c$
2. Вычислить $a = f(a)(mod n), b = f(b)(mod n)$
3. Найти $d = GCD(a - b, n)$
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.

Оценка сложности

Сложность. Заметим, что этот метод требует сделать $B–1$ операций возведения в степень $a = a^e mod n$. Есть быстрый алгоритм возведения в степень, который выполняет это за $2 * 1og_2 B$ 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 $\sqrt{n}$.

Пример работы алгоритма



```
            print("Делитель не найден")
        if d == 1:
            global ag
            ag = b
            method(n, a, b, d)

def main():
    n = 1359331
    c = 1
    a = c
    b = c
    a = f(a, n)%n
    b = f(a, n)%n
    d = gcd(a-b, n)
    if 1 < d < n:
        p = d
        print(p)
        exit()
    if d == n:
        pass
    if d == 1:
        method(n, a, b, d)
```

```
In [2]: main()
```

```
1181
```

Figure 1: Пример работы алгоритма

Таким образом, число 1181 является нетривиальным делителем числа 1359331.

Результаты выполнения лабораторной работы

Я изучил задачу разложения на множители и p-алгоритм Полларда, а также реализовал данный алгоритм программно на языке Python.

Спасибо за внимание