Отчёт по лабораторной работе №6

Разложение чисел на множители

Бакундукизе Эжид Принц НФИмд-01-21

Содержание

# 1 Цель работы

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

# 2 Теоретические сведения

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.

Согласно Основной теореме арифметики любое положительное целое число больше единицы может быть уникально записано в следующей главной форме разложения на множители, где — простые числа и — положительные целые числа [1].

Поиск эффективных алгоритмов для разложения на множители больших составных чисел ведется давно. К сожалению, совершенный алгоритм для этого пока не найден. Хотя есть несколько алгоритмов, которые могут разложить число на множители, ни один не способен провести разложение достаточно больших чисел в разумное время. Позже мы увидим, что это хорошо для криптографии, потому что современные криптографические системы полагаются на этот факт. В этой секции мы даем несколько простых алгоритмов, которые проводят разложение составного числа. Цель состоит в том, чтобы сделать процесс разложения на множители менее трудоёмким.

В 1974 г. Джон Поллард разработал метод, который находит разложение числа на простые числа. Метод основан на условии, что не имеет сомножителя, большего, чем заранее определенное значение , называемое границей. Алгоритм Полларда показывает, что в этом случае

[2]

Сложность. Заметим, что этот метод требует сделать операций возведения в степень . Есть быстрый алгоритм возведения в степень, который выполняет это за операций. Метод также использует вычисления НОД, который требует операций. Мы можем сказать, что сложность — так или иначе больше, чем или , где — число битов в . Другая проблема – этот алгоритм может заканчиваться сигналом об ошибке. Вероятность успеха очень мала, если имеет значение, не очень близкое к величине .

## 2.1 p-алгоритм Полларда

* Вход. Число , начальное значение , функция , обладающая сжимающими свойствами.
* Выход. Нетривиальный делитель числа .

1. Положить
2. Вычислить
3. Найти
4. Если , то положить и результат: . При результат: ДЕЛИТЕЛЬ НЕ НАЙДЕН. При вернуться на шаг 2.

# 3 Выполнение работы

## 3.1 Реализация алгоритма на языке Python

from math import gcd  
  
  
def f(x, n):  
 return pow((pow(x,2)+5),1,n)  
  
def Pol(n, c):  
 a = c  
 b = c  
 d = 1  
   
 while (d==1):  
   
 a = pow(f(a, n),1, n)  
 b = pow(f(f(b, n), n),1, n)  
 d = gcd(a-b, n)  
   
 if 1< d <n:  
 p = d  
 return p  
   
 if d == n:  
 return d  
   
n = 1359331  
c = 1  
p = Pol(n, c)  
  
if p !=1:  
 print("Нетривиальный делитель числа {} - {}".format(n, p))  
 print("Разложение на множители числа {} = {}\*{}".format(n, p, int(n/p)))  
   
   
import numpy as np  
  
n = 15  
sq\_n = np.sqrt(n)  
  
for i in range(1,n+1):  
 if n%i == 0 and i >=sq\_n:  
 p = i  
 q = int(n/p)  
 print ("Разложение на множители числа {} = {}\*{}".format (n,p,q))  
 print()

## 3.2 Контрольный пример

Нахождение нетривиального делителя p-методом Полларда (Fig.1)

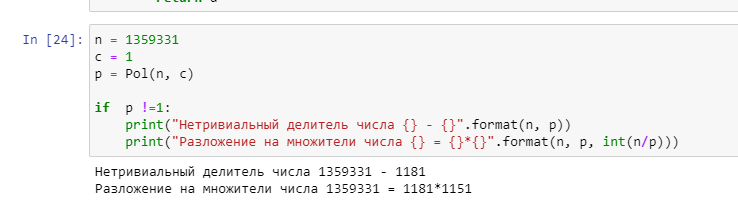


Figure 1: Нахождение нетривиального делителя p-методом Полларда

Разложение числа на простые множители (Fig.2)

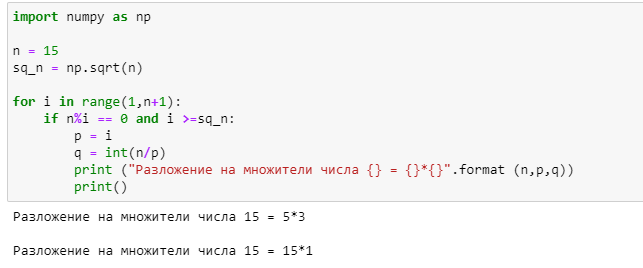


Figure 2: Разложение числа на простые множители

# 4 Выводы

В ходе выполнения данной лабораторной работы мы изучили алгоритмы разложения чисел на простые множители, изучили и реализовали p-метод Полларда для нахождения нетривиального делителя, а также написали алгоритм для разложения числа на простые множители.

# Список литературы

1. [Алгоритмы тестирования на простоту и факторизации](https://habr.com/ru/post/521876/)
2. [P-метод Полларда](https://ru.bmstu.wiki/P-метод_Полларда)