Отчёт по лабораторной работе №7

Дискретное логарифмирование в конечном поле

Шутенко Виктория Михайловна

Содержание

# 1 Цель рабoты

Приoбрести практические навыки рабoты с разложением чисел на множители.

# 2 Хoд рабoты

## 2.1 Алгоритм, реализующий р-метод Полларда

# Laboratory Work  
# Theme: Discrete logarthmification  
# Author: Vladimir Doborschuk  
  
# --- Modules ---  
  
import numpy as np  
  
# --- Functions ---  
  
# --- mod(a, b) ---  
  
def mod(a ,b):  
 return a % b  
  
# --- find mod order ---  
  
def order(a, p):  
 x = 1  
 while mod(a\*\*x - 1, p) != 0:  
 x += 1  
   
 return x  
  
# --- Pollard's P-method for Log ---  
  
'''  
a - основание  
b - значение остатка  
p - простое число  
'''  
def po\_method(a: int, b: int, p: int):  
 print(f"\n{a}^(x) = {b} mod {p}")  
 print("-----------------------------------------------------------------")  
 print('|\tc\t|\tlog c\t|\td\t|\tlog d\t|')  
 print("-----------------------------------------------------------------")  
   
 u = np.random.randint(4)  
 v = np.random.randint(4)  
 r = order(a, p)  
   
 c = mod(np.power(a, u) \* np.power(b, v), p)  
 d = c  
   
 u\_c, u\_d = u, u  
 v\_c, v\_d = v, v  
   
 print(f'|\t{c}\t|\t{u\_c}+{v\_c}x\t|\t{d}\t|\t{u\_d}+{v\_d}x\t|')  
   
 def f(x, u\_x, v\_x):  
 if x < r:  
 return mod(a\*x, p), u\_x + 1, v\_x  
 else:  
 return mod(b\*x, p), u\_x, v\_x + 1   
  
 c, u\_c, v\_c = f(c, u\_c, v\_c)  
 tmp\_d = f(d, u\_d, v\_d)  
 d, u\_d, v\_d = f(tmp\_d[0], tmp\_d[1], tmp\_d[2])  
   
 while mod(c, p) != mod(d, p):  
 print(f'|\t{c}\t|\t{u\_c}+{v\_c}x\t|\t{d}\t|\t{u\_d}+{v\_d}x\t|')  
 c, u\_c, v\_c = f(c, u\_c, v\_c)  
 tmp\_d = f(d, u\_d, v\_d)  
 d, u\_d, v\_d = f(tmp\_d[0], tmp\_d[1], tmp\_d[2])  
   
 print(f'|\t{c}\t|\t{u\_c}+{v\_c}x\t|\t{d}\t|\t{u\_d}+{v\_d}x\t|')  
 print("-----------------------------------------------------------------")  
   
 x = 1  
 # print(v\_c - v\_d, u\_d - u\_c)  
 while mod((v\_c - v\_d)\*x, r) != mod(u\_d - u\_c, r):  
 x += 1  
   
 print(f"x = {x}")  
 print(f"\n{a}^({x}) = {b} mod {p}")  
 print("-----------------------------------------------------------------")  
 return x  
  
# --- Main ---  
  
def main():  
 po\_method(10, 64, 107)  
 po\_method(2, 1, 15)  
   
   
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

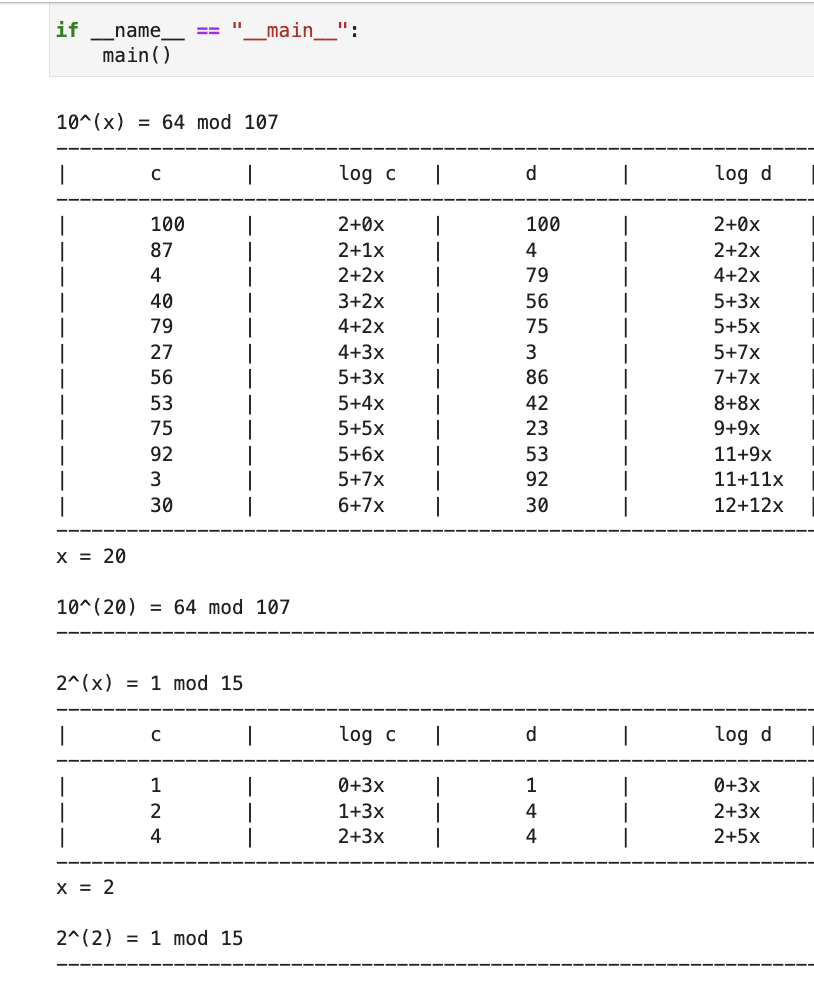


Figure 1: Тестирование