МИНОБРНАУКИ РОССИИ
ФГБОУ ВО «СГУ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

АЛГОРИТМЫ АЛГЕБРЫ И ТЕОРИИ ЧИСЕЛ

ЛАБОРАТОРНАЯ РАБОТА №2

студента 4 курса 431 группы
направления 10.05.01 — Компьютерная безопасность
факультета КНиИТ
Никитина Арсения Владимировича

Проверил
доцент _____ А. С. Гераськин

Саратов 2022

СОДЕРЖАНИЕ

1 Задание лабораторной работы

Решите сравнение вида $ax \equiv b \ (mod \ m)$ с помощью теоремы Эйлера.

2 Теоретическая часть

Если НОД(a, m) чисел a и m равен d и d делит b, то сравнение $ax \equiv b\ (mod\ m)$ имеет d решений. Если же d не делит b, то сравнение не имеет решений.

После нахождения d и выполнения условий, требуется разделить соотношение на это число d.

Пусть после разделения соотношения оно имеет вид:

$$a_1 x \equiv b_1\ (mod\ m_1)$$

По теореме Эйлера для чисел a_1 и m_1, удовлетворяющих условию $(a_1, m_1) = 1$, выполняется сравнение $a_1^{\varphi(m_1)} \equiv 1\ (mod\ m_1)$, где $\varphi(m)$ — функция Эйлера. Поэтому решение x_0 сравнения $ax \equiv b\ (mod\ m)$ можно найти по формуле:

$$x_0 \equiv b_1 a_1^{\varphi(m_1)-1}\ (mod\ m_1)$$

Далее все остальные реешения можно найти по формуле:

$$x \equiv x_0 + m_1 k\ (mod\ m),\ k = \overline{0, d}$$

3 Практическая часть

3.1 Пример работы алгоритма

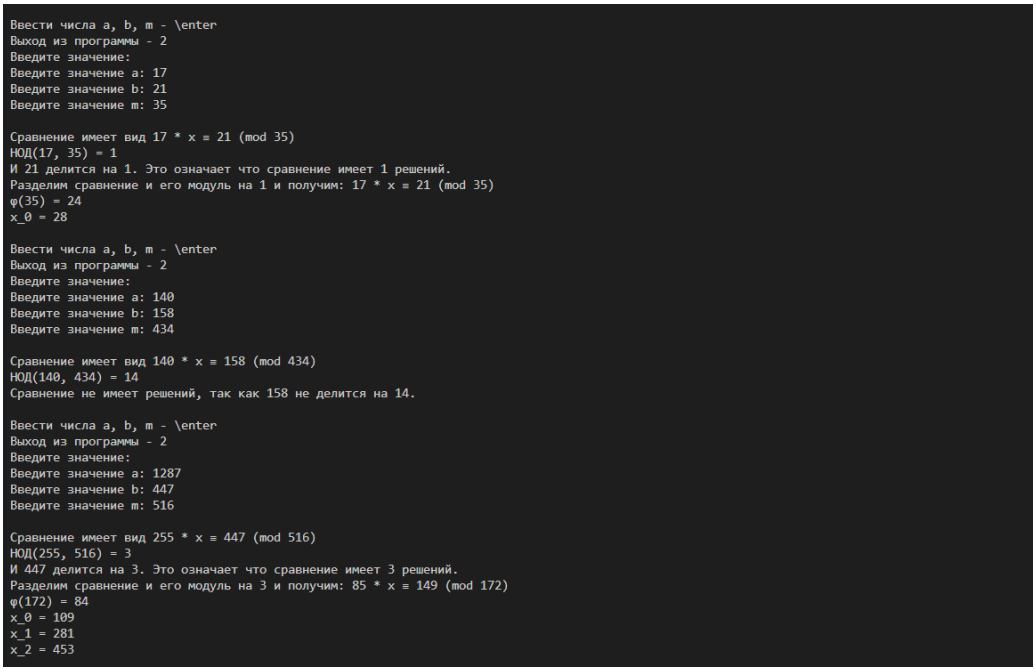

Рисунок 1

3.2 Код программы, реализующей рассмотренный алгоритм

```
1  import numpy
2
3
4  def pow(number, modula, power):
5      number_save = number
6      for _ in range(power - 2):
7          number = number * number_save % modula
8      return number
9
10
11 def gcd(a, b):
12     if b == 0:
13         return a
14     else:
15         return gcd(b, a % b)
16
17
18
19 def factor(p):
20
```

```
21      d, factors, unique_factors = 2, [], set()
22
23      while d*d <= p:
24
25          while (p % d) == 0:
26              factors.append(d)
27              unique_factors.add(d)
28              p //= d
29
30          d += 1
31
32      if p > 1:
33         factors.append(p)
34         unique_factors.add(p)
35
36      return list(unique_factors), [factors.count(i) for i in unique_factors]
37
38
39  def get_phi(p):
40
41      factors, powers = factor(p)
42
43      if len(factors) == 1:
44          return p - 1
45      else:
46          res = [factors[i] ** powers[i] - factors[i] ** (powers[i] - 1) for i
47          in range(len(powers))]
48
49          return numpy.prod(res)
50
51
52  def main():
53
54      while True:
55
56          print('\nВвести числа a, b, m - \enter')
57          print('Выход из программы - 2')
58
59          try:
60              value = int(input('Введите значение: '))
61          except ValueError:
```

```
62              value = 1
63
64          if value == 1:
65
66              a = int(input('Введите значение a: '))
67              b = int(input('Введите значение b: '))
68              m = int(input('Введите значение m: '))
69              a %= m
70              b %= m
71
72              print(f'\nСравнение имеет вид {a} * x \u2261 {b} (mod {m})')
73
74              d = gcd(a, m)
75              print(f'НОД({a}, {m}) = {d}')
76
77              if not b % d:
78                  print(f'И {b} делится на {d}. Это означает что сравнение
                    ↪  имеет {d} решений.')
79
80                  a //= d
81                  b //= d
82                  m_save = m
83                  m //= d
84                  print(f'Разделим сравнение и его модуль на {d} и получим:
                    ↪  {a} * x \u2261 {b} (mod {m})')
85
86                  phi = get_phi(m)
87                  print(f'\u03d5({m}) = {phi}')
88
89                  x_0 = b * pow(a, m, phi) % m
90                  print(f'x_0 = {x_0}')
91
92                  res = [(x_0 + m * i) % m_save for i in range(1, d)]
93                  for i in range(d - 1):
94                      print(f'x_{i+1} = {res[i]}')
95
96              else:
97                  print(f'Сравнение не имеет решений, так как {b} не делится на
                    ↪  {d}.')
98
99
```

```
100         if value == 2:
101             print('Работа программы завершена')
102             return
103 
104 
105 if __name__ == "__main__":
106     main()
```