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

АЛГОРИТМЫ АЛГЕБРЫ И ТЕОРИИ ЧИСЕЛ

ЛАБОРАТОРНАЯ РАБОТА №11

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

Реализовать факторизацию Ферма.

2 Теоретическая часть

Метод Ферма основан на теореме о представлении числа в виде разности двух квадратов:

Если $n > 1$ нечётно, то существует взаимно однозначное соответствие между разложением $n = a \cdot b$ и представлением в виде разности квадратов $n = x^2 - y^2$ с $x > y > 0$, задаваемое формулами $x = (a + b)/2$, $y = (a - b)/2$, $a = x + y$, $b = x - y$.

Для разложения на множители нечётного числа n ищется пара чисел (x, y) таких, что $x^2 - y^2 = n$, или $(x - y) \cdot (x + y) = n$. При этом числа $(x + y)$ и $(x - y)$ являются делителями n, возможно, тривиальными (то есть одно из них равно 1, а другое — n.

В нетривиальном случае равенство $x^2 - y^2 = n$ равносильно $x^2 - n = y^2$, то есть тому, что $x^2 - n$ является квадратом.

Поиск квадрата такого вида начинается с $x = \lceil \sqrt{n} \rceil$ — наименьшего числа, при котором разность $x^2 - n$ неотрицательна.

Для каждого значения $k \in \mathbb{N}$, начиная с $k = 1$, вычисляют $(\lceil \sqrt{n} \rceil + k)^2 - n$ и проверяют, не является ли это число точным квадратом. Если не является, то k увеличивают на единицу и переходят на следующую итерацию.

Если $(\lceil \sqrt{n} \rceil + k)^2 - n$ является точным квадратом, то есть $x^2 - n = (\lceil \sqrt{n} \rceil + k)^2 - n = y^2$, то получено разложение:

$n = x^2 - y^2 = (x + y)(x - y) = a \cdot b$,

в котором $x = \lceil \sqrt{n} \rceil + k$.

Если оно является тривиальным и единственным, то n — простое.

На практике значение выражения на $(k + 1)$-м шаге вычисляется с учётом значения на k-м шаге:

$(s + 1)^2 - n = s^2 + 2s + 1 - n$, где $s = \lceil \sqrt{n} \rceil + k$.

3 Практическая часть

3.1 Пример работы алгоритма

```
Введите число n: 3211197185
3211197185 = (45874263 - 45874228)(45874263 + 45874228)
Выполнить факторизацию Ферма - \enter
Выход из программы - 2
Введите значение:

Введите число n: 321197185
321197185 = (18017 - 1848)(18017 + 1848)
Выполнить факторизацию Ферма - \enter
Выход из программы - 2
Введите значение:

Введите число n: 16169
16169 = (12 - 11)(12 + 11)(10 - 9)(10 + 9)(19 - 18)(19 + 18)
Выполнить факторизацию Ферма - \enter
Выход из программы - 2
Введите значение:

Введите число n: 19865
19865 = (15 - 14)(15 + 14)(3 - 2)(3 + 2)(69 - 68)(69 + 68)
Выполнить факторизацию Ферма - \enter
Выход из программы - 2
Введите значение: 18017

Выполнить факторизацию Ферма - \enter
Выход из программы - 2
Введите значение:

Введите число n: 18017
18017 = (210 - 209)(210 + 209)(22 - 21)(22 + 21)
Выполнить факторизацию Ферма - \enter
Выход из программы - 2
Введите значение: 2
Работа программы завершена
```

Рисунок 1

3.2 Код программы, реализующей рассмотренный алгоритм

```
1  import math
2
3  def factorization(n, res):
4
5      sq = math.ceil(math.sqrt(n))
6      k = 0
7
8      while True:
9
10         subres1 = (sq + k) ** 2 - n
11         subres = math.sqrt(subres1)
12
13         if not math.ceil(subres) == math.floor(subres):
14             k += 1
```

```
15          else:
16              subres = math.ceil(subres)
17              break
18
19      if len(res) > 0:
20          for i, a in enumerate(res):
21              if sq + k - subres == a[2] or sq + k + subres == a[2]:
22                  if a[2] != 1:
23                      res.remove(a)
24                  res.insert(i, [sq + k, subres, sq + k - subres])
25                  res.insert(i + 1, [sq + k, subres, sq + k + subres])
26                  break
27      else:
28
29          res.append([sq + k, subres, sq + k - subres])
30          res.append([sq + k, subres, sq + k + subres])
31
32      if sq + k - subres > 1:
33
34              factorization(sq + k - subres, res)
35
36              factorization(sq + k + subres, res)
37
38      return res
39
40
41
42  def main():
43
44      while True:
45
46          print('\nВыполнить факторизацию Ферма - \enter')
47          print('Выход из программы - 2')
48
49          try:
50              value = int(input('Введите значение: '))
51
52          except ValueError:
53              value = 1
54
55          if value == 1:
```

```
56
57              n = int(input("\nВведите число n: "))
58              twos = 0
59              n_save = n
60
61              while not n % 2:
62                  n //= 2
63                  twos += 1
64
65              res= factorization(n, [])
66
67              if twos != 0:
68                  print(f'{n_save} = {2}^{twos} *', end='')
69              else:
70                  print(f'{n_save} = ', end='')
71
72              for (a, b, _) in res[:-1:2]:
73                  print(f'({a} - {b})({a} + {b})', end='')
74
75          elif value == 2:
76              print('Работа программы завершена')
77              return
78
79
80  if __name__ == "__main__":
81      main()
```