

# **Лабораторная работа №4. Вычисление НОД.**

**Предмет: Математические основы защиты информации и  
информационной безопасности**

**Александр Сергеевич Баклашов**

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>6</b>
3.1	Алгоритмы Евклида . . . . .	6
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
4.1	Алгоритм Евклида . . . . .	7
4.1.1	Задача . . . . .	7
4.2	Бинарный алгоритм Евклида . . . . .	8
4.2.1	Задача . . . . .	8
4.3	Расширенный алгоритм Евклида . . . . .	8
4.3.1	Задача . . . . .	8
4.4	Расширенный бинарный алгоритм Евклида . . . . .	9
4.4.1	Задача . . . . .	9
<b>5</b>	<b>Выводы</b>	<b>11</b>
<b>6</b>	<b>Библиография</b>	<b>12</b>

# List of Figures

4.1	Алгоритм Евклида . . . . .	7
4.2	Бинарный алгоритм Евклида . . . . .	8
4.3	Расширенный алгоритм Евклида . . . . .	9
4.4	Расширенный бинарный алгоритм Евклида . . . . .	10

# 1 Цель работы

Рассмотреть и реализовать алгоритмы нахождения НОД.

## 2 Задание

Реализовать следующие алгоритмы:

- Алгоритм Евклида;
- Бинарный алгоритм Евклида;
- Расширенный алгоритм Евклида;
- Расширенный бинарный алгоритм Евклида.

## 3 Теоретическое введение

### 3.1 Алгоритмы Евклида

Алгоритм Евклида — эффективный алгоритм для нахождения наибольшего общего делителя двух целых чисел (или общей меры двух отрезков). Алгоритм назван в честь греческого математика Евклида (III век до н. э.), который впервые описал его в VII и X книгах «Начал». Это один из старейших численных алгоритмов, используемых в наше время.

В самом простом случае алгоритм Евклида применяется к паре положительных целых чисел и формирует новую пару, которая состоит из меньшего числа и разницы между большим и меньшим числом. Процесс повторяется, пока числа не станут равными. Найденное число и есть наибольший общий делитель исходной пары. Евклид предложил алгоритм только для натуральных чисел и геометрических величин (длин, площадей, объёмов). Однако в XIX веке он был обобщён на другие типы математических объектов, включая целые числа Гаусса и полиномы от одной переменной. Это привело к появлению в современной общей алгебре такого понятия, как евклидово кольцо. Позже алгоритм Евклида был обобщён на другие математические структуры, такие как узлы и многомерные полиномы.

Для данного алгоритма существует множество теоретических и практических применений. В частности, он является основой для криптографического алгоритма с открытым ключом RSA, широко распространённого в электронной коммерции. Также алгоритм используется при решении линейных диофантовых уравнений, при построении непрерывных дробей, в методе Штурма. Алгоритм Евклида является основным инструментом для доказательства теорем в современной теории чисел, например таких как теорема Лагранжа о сумме четырёх квадратов и основная теорема арифметики.

## 4 Выполнение лабораторной работы

### 4.1 Алгоритм Евклида

#### 4.1.1 Задача

Реализовать алгоритм Евклида

##### 4.1.1.1 Решение

Реализуем алгоритм Евклида (рис. 4.1)

#### Алгоритм Евклида

```
In [1]: def AE ( a, b ):  
        if a == 0 or b == 0:  
            return a + b;  
        if a>b:  
            return AE( a - b, b )  
        else:  
            return AE( a, b - a )
```

```
In [2]: AE (20,10)
```

```
Out[2]: 10
```

Figure 4.1: Алгоритм Евклида

## 4.2 Бинарный алгоритм Евклида

### 4.2.1 Задача

Реализовать бинарный алгоритм Евклида

#### 4.2.1.1 Решение

Реализуем бинарный алгоритм Евклида (рис. 4.2)

Бинарный алгоритм Евклида

```
In [3]: def BAE ( a, b ):  
        g=1  
        while True:  
            if a%2==0 and b%2==0:  
                a = a/2  
                b = b/2  
                g = g*2  
            else:  
                u = a  
                v = b  
                break  
        while (u!=0):  
            while (u%2 == 0):  
                u = u/2  
            while (v%2 == 0):  
                v = v/2  
            if u >= v:  
                u = u-v  
            else:  
                v = v-u  
        d = g*v  
        return d  
  
In [4]: BAE (20,10)  
Out[4]: 10.0
```

Figure 4.2: Бинарный алгоритм Евклида

## 4.3 Расширенный алгоритм Евклида

### 4.3.1 Задача

Реализуем расширенный алгоритм Евклида (рис. 4.3)



## Расширенный алгоритм Евклида

```
In [5]: def RAE(a, b):  
        if b == 0:  
            return a, 1, 0  
  
        x1, x0, y1, y0 = 1, 0, 0, 1  
        while b > 0:  
            q = a // b  
            a, b = b, a % b  
            x1, x0 = x0, x1 - q * x0  
            y1, y0 = y0, y1 - q * y0  
  
        return a, x1, y1
```

```
In [6]: RAE(20, 10)
```

```
Out[6]: (10, 0, 1)
```

Figure 4.3: Расширенный алгоритм Евклида

## 4.4 Расширенный бинарный алгоритм Евклида

### 4.4.1 Задача

Реализуем расширенный бинарный алгоритм Евклида (рис. 4.4)

```
In [7]: def RBAE ( a, b ):
        g=1
        while True:
            if a%2==0 and b%2==0:
                a = a/2
                b = b/2
                g = g*2
            else:
                u = a
                v = b
                A = 1
                B = 0
                C = 0
                D = 1
                break
        while (u!=0):
            while (u%2 == 0):
                u = u/2
                if A%2==0 and B%2==0:
                    A=A/2
                    B=B/2
                else:
                    A=(A+b)/2
                    B=(B-a)/2
            while (v%2 == 0):
                v = v/2
                if C%2==0 and D%2==0:
                    C=C/2
                    D=D/2
                else:
                    C=(C+b)/2
                    D=(D-a)/2
            if u >= v:
                u = u-v
                A = A-C
                B = B-D
            else:
                v = v-u
                C = C-A
                D = D-B
        d = g*v
        x = C
        y = D
        return d,x,y
```

```
In [8]: RBAE (20, 10)
```

```
Out[8]: (10.0, 0, 1)
```

Figure 4.4: Расширенный бинарный алгоритм Евклида

## 5 Выводы

В ходе данной лабораторной работы я рассмотрел и реализовал следующие алгоритмы:

- Алгоритм Евклида;
- Бинарный алгоритм Евклида;
- Расширенный алгоритм Евклида;
- Расширенный бинарный алгоритм Евклида.

## 6 Библиография

1. Python documentation. [Электронный ресурс]. М. URL: Python documentation (Дата обращения: 28.09.2023).
2. Лабораторная работа №4. Вычисление НОД. - 4 с. [Электронный ресурс]. М. URL: Лабораторная работа №4. Вычисление НОД. (Дата обращения: 19.10.2023).