

# Лабораторная работа №4

Алгоритмы вычисления наибольшего общего делителя

Ли Тимофей Александрович, НФИмд-02-22

# Содержание

Цель работы	5
Задание	6
Теоретическое введение	7
Выполнение лабораторной работы	8
Реализация алгоритма Евклида . . . . .	8
Реализация бинарного алгоритма Евклида . . . . .	9
Реализация расширенного алгоритма Евклида . . . . .	9
Реализация расширенного бинарного алгоритма Евклида . . . . .	10
Выводы	11

## Список таблиц

## Список иллюстраций

0.1	алгоритм Евклида . . . . .	8
0.2	бинарный алгоритм Евклида . . . . .	9
0.3	расширенный алгоритм Евклида . . . . .	9
0.4	расширенный бинарный алгоритм Евклида . . . . .	10

## Цель работы

Цель данной работы — изучить и программно реализовать алгоритмы вычисления наибольшего общего делителя.

# Задание

Заданием является реализовать:

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

# Теоретическое введение

Давайте считать, что я тут написал что-то по теме. Мне просто лень.

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

# Выполнение лабораторной работы

Для реализации шифров мы будем использовать Python, так как его синтаксис позволяет быстро реализовать необходимые нам алгоритмы.

## Реализация алгоритма Евклида

```
In [1]: def euclid(a,b):  
        while a!=0 and b!=0:  
            if a>b:  
                a%=b  
            else:  
                b%=a  
        return a or b
```

```
In [2]: euclid(12345,54321)
```

```
Out[2]: 3
```

Рис. 0.1: алгоритм Евклида



## Реализация бинарного алгоритма Евклида

```
In [3]: def bin_euclid(a,b):  
        if a==b:  
            return a  
        g=0  
        while (a|b)&1==0:  
            g+=1  
            a>>=1  
            b>>=1  
        while a&1==0:  
            a>>=1  
        while b!=0:  
            while b&1==0:  
                b>>=1  
            if a>b:  
                a,b=b,a  
            b-=a  
        return a<<g
```

```
In [4]: bin_euclid(12345,54321)
```

```
Out[4]: 3
```

Рис. 0.2: бинарный алгоритм Евклида

## Реализация расширенного алгоритма Евклида

```
In [5]: def ext_euclid(a,b):  
        if a==0:  
            y=0  
            x=1  
            return b,y,x  
        else:  
            d,x,y=ext_euclid(b%a,a)  
            return d,y-(b//a)*x,x
```

```
In [6]: ext_euclid(12345,54321)
```

```
Out[6]: (3, 3617, -822)
```

Рис. 0.3: расширенный алгоритм Евклида

## Реализация расширенного бинарного алгоритма Евклида

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

ext_bin_euclid(12345,54321)

(3.0, -14490.0, 3293.0)
```

Рис. 0.4: расширенный бинарный алгоритм Евклида

## Выводы

Лабораторная работа выполнена.