

Математические основы защиты информации и информационной безопасности. Лабораторная работа № 4 на тему “Вычисление наибольшего общего делителя”

Лубышева Ярослава Михайловна

RUDN University, Moscow, Russian Federation

- Цели и задачи
- Выполнение
- Результаты
- Список литературы

Выполнить задание к лабораторной работе № 4:

1. Ознакомиться с алгоритмами вычисления наибольшего общего делителя: алгоритм Евклида, бинарный алгоритм Евклида, расширенный алгоритм Евклида, расширенный бинарный алгоритм Евклида.
2. Реализовать все алгоритмы программно.

```
# алгоритм Евклида
# вход - целые числа  $0 < b \leq a$ 
# выход -  $d = \text{НОД}(a, b)$ 
def alg_Euclid(a, b):
    r = [a, b]
    i = 1
    while r[i-1] % r[i] != 0:
        r.append(r[i-1] % r[i])
        i += 1
    d = r[i]
    return d
```

Рис. 1: Программная реализация алгоритма Евклида

```
# бинарный алгоритм Евклида
# вход - целые числа  $0 < b \leq a$ ; выход -  $d = \text{НОД}(a, b)$ 
def bin_alg_Euclid(a, b):
    g = 1
    while a%2==0 and b%2==0:
        a /= 2
        b /= 2
        g *= 2
    u = a; v = b
    while u!=0:
        while u%2==0:
            u /= 2
        while v%2==0:
            v /= 2
        if u>=v:
            u -= v
        else:
            v -= u
    d = g*v
    return d
```

Рис. 2: Программная реализация бинарного алгоритма Евклида

```
# расширенный алгоритм Евклида
# вход - целые числа 0 < b <= a
# выход - d=НОД(a,b), целые числа x и y, что a*x+b*y=d
def advanced_alg_Euclid(a, b):
    r = [a, b]
    x = [1, 0]
    y = [0, 1]
    i = 1
    while r[i-1]%r[i]!=0:
        r.append(r[i-1]%r[i])
        q = r[i-1]//r[i]
        x.append(x[i-1]-q*x[i])
        y.append(y[i-1]-q*y[i])
        i += 1
    d = r[i]
    x = x[i]
    y = y[i]
    return d, x, y
```

Рис. 3: Программная реализация расширенного алгоритма Евклида

```
# расширенный бинарный алгоритм Евклида
# вход - целые числа  $0 < b \leq a$ ; выход -  $d = \text{НОД}(a, b)$ , целые числа  $x$  и  $y$ , что  $a \cdot x + b \cdot y = d$ 
def advanced_bin_alg_Euclid(a, b):
    g = 1
    while a%2==0 and b%2==0:
        a /= 2
        b /= 2
        g *= 2
    u = a; v = b
    big_a = 1; big_b = 0; big_c = 0; big_d = 1
    while u!=0:
        while u%2==0:
            u /= 2
        if big_a%2==0 and big_b%2==0:
            big_a /= 2
            big_b /= 2
        else:
            big_a = (big_a + b) / 2
            big_b = (big_b - a) / 2
```

Рис. 4: Программная реализация расширенного бинарного алгоритма Евклида (часть 1)

```
while v%2==0:
    v /= 2
    if big_c%2==0 and big_d%2==0:
        big_c /= 2
        big_d /= 2
    else:
        big_c = (big_c + b) / 2
        big_d = (big_d - a) / 2
if u>=v:
    u = u-v
    big_a -= big_c
    big_b -= big_d
else:
    v = v-u
    big_c -= big_a
    big_d -= big_b
d = g*v; x = big_c; y = big_d
return d, x, y
```

Рис. 5: Программная реализация расширенного бинарного алгоритма Евклида (часть 2)


```
a = 1405054161
b = 2653
print(alg_Euclid(a,b))
print(bin_alg_Euclid(a,b))
print(advanced_alg_Euclid(a,b))
print(advanced_bin_alg_Euclid(a,b))

7
7.0
(7, 59, -31246964)
(7.0, 438.0, -231968987.0)
```

Рис. 6: Результаты работы алгоритмов вычисления наибольшего общего делителя

Выполнено задание к лабораторной работе № 4.

1. Методические материалы курса