

Лабораторная работа №4

Вычисление наибольшего общего делителя

АВТОР

Кюнкриков Д.С. -

ПРИНАДЛЕЖНОСТЬ

Российский университет дружбы народов, Москва,
Россия -

Информация

Докладчик

- Кюнкриков Даниил Саналович
- студент уч. группы НПИМд-01-24
- Российский университет дружбы народов
- 1132249574@pfur.ru
- https://github.com/DanzanK/2025-2026_math-sec/tree/main

Вводная часть

Актуальность

- Создание кода алгоритмов вычисления наибольшего общего делителя (алгоритмы: Евклида; бинарный Евклида; расширенный Евклида; расширенный бинарный Евклида) на python, чтобы понять принципы работы алгоритмов.

Объект и предмет исследования

- Алгоритмы вычисления наибольшего общего делителя
- Алгоритм Евклида
- Бинарный алгоритм Евклида
- Расширенный алгоритм Евклида
- Расширенный бинарный алгоритм Евклида
- Веб-сервис GitHub
- Язык разметки Markdown

Цели и задачи

- Реализовать алгоритмы вычисления наибольшего общего делителя на языке программирования python

Процесс выполнения работы

Реализация алгоритма Евклида

```
#1 Алг. евклида
def euclidean_gcd(a,b):
    r_prev, r_curr = a,b

    while r_curr !=0:
        r_next = r_prev % r_curr
        r_prev = r_curr
        r_curr = r_next

    d = r_prev
    return d

vala = 12345
valb = [24690, 54321, 12541]
for val in valb:
    print(f"GCD({vala}, {val}) = {euclidean_gcd(vala, val)}")
```

Реализация бинарного алгоритма Евклида

#2 Бинарный алгоритм Евклида

```
def binareuc_gcd(a,b):  
    g = 1  
  
    while a%2 == 0 and b%2 ==0:  
        a //=2  
        b //= 2  
        g *= 2  
  
    u, v = a, b  
  
    while u != 0:  
        while u% 2 ==0:  
            u //= 2  
        while v% 2 ==0:  
            v //= 2  
        if u >= v:  
            u = u-v  
        else:  
            v = v-u  
  
    d = g * v  
  
    return d  
  
vala = 12345  
valb = [24690, 54321, 12541]  
for val in valb:  
    print(f"GCD({vala}, {val}) = {binareuc_gcd(vala, val)}")
```

Реализация расширенного алгоритма Евклида

```

#3 Расширенный алгоритм Евклида
def extendedeuc_gcd(a,b):
    r_prev, r_curr = a,b
    x_prev, x_curr = 1, 0
    y_prev, y_curr = 0, 1

    while r_curr !=0:
        q = r_prev // r_curr
        r_next = r_prev % r_curr
        x_next = x_prev - q * x_curr
        y_next = y_prev - q * y_curr

        r_prev, r_curr = r_curr, r_next
        x_prev, x_curr = x_curr, x_next
        y_prev, y_curr = y_curr, y_next

    d, x, y = r_prev, x_prev, y_prev
    return d, x, y_curr

d1, x1, y1 = extendedeuc_gcd(105,91)
d2, x2, y2 = extendedeuc_gcd(154, d1)
print(f"GDC(105,91) = {d1, x1, y1}")
print(f"final GDC(154,7) = {d2, x2, y2}")

```

Реализация расширенного бинарного алгоритма Евклида

```

def extendbineuc_gcd(a,b):
    g = 1
    while a% 2 ==0 and b%2 == 0:
        a //=2
        b //= 2
        g *= 2

    u, v = a, b
    A, B = 1, 0
    C, D = 0, 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
            print("v: " + str(v))

        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
            print("u: " + str(u))

        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

dbin,xbin,ybin = extendbineuc_gcd(105, 91)
print(f"GDC(105,91) = {dbin,xbin,ybin}")

```

Результаты

- Выполнены все необходимы действия для реализации задач лабораторной работы

```
GDC(105,91) = (7, -6, 7)
PS C:\Users\Danzan\Desktop\MOZiiIB\лр> & C:/ProgramData/anaconda3/python.exe c:/Users/D
anzan/Desktop/MOZiiIB/лр/ЛР4/LR4.py
GCD(12345, 24690) = 12345
GCD(12345, 54321) = 3
GCD(12345, 12541) = 1
GCD(12345, 24690) = 12345
GCD(12345, 54321) = 3
GCD(12345, 12541) = 1
GDC(105,91) = (7, -6, -15)
final GDC(154,7) = (7, 0, -22)
v: 91
u: 7
u: 7
u: 7
GDC(105,91) = (7, -6, 7)
PS C:\Users\Danzan\Desktop\MOZiiIB\лр>
```

Вывод

Реализованы алгоритмы вычисления наибольшего общего делителя, включая следующие: - Алгоритм Евклида - Бинарный алгоритм Евклида - Расширенный алгоритм Евклида - Расширенный бинарный алгоритм Евклида