

Отчёт по лабораторной работе 4

Вычисление наибольшего общего делителя

Аристова Арина Олеговна

Содержание

| | |
|---|-----------|
| 1 Цель работы | 4 |
| 2 Задание | 5 |
| 3 Выполнение лабораторной работы | 6 |
| 3.1 Решение задачи. Алгоритм Евклида | 6 |
| 3.2 Решение задачи. Бинарный алгоритм Евклида | 6 |
| 3.3 Решение задачи. Расширенный алгоритм Евклида | 7 |
| 3.4 Решение задачи. Расширенный бинарный алгоритм Евклида | 8 |
| 3.5 Проверка работы кода | 10 |
| 4 Вывод | 11 |
| Список литературы | 12 |

Список иллюстраций

| | |
|--|----|
| 3.1 Результат работы программы | 10 |
|--|----|

1 Цель работы

Изучить алгоритм Евклида для нахождения НОД (наибольшего общего делителя), бинарный алгоритм Евклида, а также расширенные их версии, которые находят также коэффициенты соотношения Безу: $\text{НОД} = x * a + y * b$, где a, b - рассматриваемые числа

2 Задание

Реализовать программный код для:

- алгоритма Евклида
- бинарного алгоритма Евклида
- расширенного алгоритма Евклида
- расширенного бинарного алгоритма Евклида

3 Выполнение лабораторной работы

3.1 Решение задачи. Алгоритм Евклида

Последовательно заменяем числа (a, b) на $(b, a \% b)$ до тех пор, пока $b \neq 0$.
Последнее ненулевое число — это НОД.

```
# Базовый алгоритм
function evclid(a:Int, b:Int)
    while b != 0
        a, b = b, a % b
    end
    return abs(a)
end
```

3.2 Решение задачи. Бинарный алгоритм Евклида

Использует только побитовые сдвиги и вычитания. Общие степени двойки выносятся отдельно.

```
function bin_evclid(a:Int, b:Int)
    if a == 0
        return abs(b)
    end
    if b == 0
```

```

        return abs(a)
    end

    shift = 0
    while iseven(a) && iseven(b)
        a >>= 1
        b >>= 1
        shift += 1
    end

    while a != b
        if iseven(a)
            a >>= 1
        elseif iseven(b)
            b >>= 1
        elseif a > b
            a = a - b
        else
            b = b - a
        end
    end
    return a << shift
end

```

3.3 Решение задачи. Расширенный алгоритм Евклида

Классический алгоритм, дополненный тем, что находит также числа x и y , такие что выполняется линейная комбинация: \$ НОД(a,b)=ax+by\$

```

function extended_evclid(a:Int, b:Int)
    old_r, r = a, b
    old_x, x = 1, 0

```

```

old_y, y = 0, 1
while r != 0
    quotient = div(old_r, r)
    old_r, r = r, old_r - quotient*r
    old_x, x = x, old_x - quotient*x
    old_y, y = y, old_y - quotient*y
end
return old_r, old_x, old_y
end

```

3.4 Решение задачи. Расширенный бинарный алгоритм Евклида

Классический бинарный алгоритм, дополненный тем, что находит также числа x и y , такие что выполняется линейная комбинация: $(a, b) = a * x + b * y$

```

function extended_bin_evclid(a:Int, b:Int)
    if a == 0
        return b, 0, 1
    end
    if b == 0
        return a, 1, 0
    end
    g = 1
    while iseven(a) && iseven(b)
        a >>= 1
        b >>= 1
        g <<= 1
    end

```

```

u, v = a, b
A, B, C, D = 1, 0, 0, 1
while u != 0
    while iseven(u)
        u >>= 1
        if iseven(A) && iseven(B)
            A >>= 1
            B >>= 1
        else
            A = (A + b) >> 1
            B = (B - a) >> 1
        end
    end
    while iseven(v)
        v >>= 1
        if iseven(C) && iseven(D)
            C >>= 1
            D >>= 1
        else
            C = (C + b) >> 1
            D = (D - a) >> 1
        end
    end
    if u >= v
        u -= v
        A -= C
        B -= D
    else
        v -= u
    end
end

```

```
C -= A  
D -= B  
end  
end  
return g * v, C, D  
end
```

```
println(evclid(12, 20))  
println(bin_evclid(12, 20))  
println(extended_evclid(12, 20))  
println(extended_bin_evclid(12, 20))
```

3.5 Проверка работы кода

Проверяю работу кода. Результаты для каждого из вариантов алгоритмов получились идентичными.

```
PS C:\Users\arist\Github\study_2025-2026_mathsec\labs\lab04> julia .\evclid.jl  
4  
4  
(4, 2, -1)  
(4, 2, -1)
```

Рис. 3.1: Результат работы программы

4 Вывод

В ходе выполнения данной лабораторной работы мною были получены знания о нахождении НОД с помощью различных вариантов алгоритма Евклиды, а также написана программа, реализующая каждый из них.

Расширенные алгоритмы позволяют вычислить коэффициенты x и y , удовлетворяющие формуле: $(a, b) = a * x + b * y$

Список литературы

- Описание лабораторной работы