

ЛАБОРАТОРНАЯ РАБОТА №4

Вычисление наибольшего общего делителя

Кузнецов Юрий Владимирович

Введение

В данном отчёте будет представлена реализация вычисления наибольшего общего делителя

Основное содержание

Шифры простой замены:

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

Кодовая реализация

Алгоритм Евклида

```
function euclid(a::Int, b::Int)::Int
    while a != 0 && b != 0
        if a >= b
            a %= b
        else
            b %= a
        end
    end
    return a != 0 ? a : b
end

println(euclid(a, b))
```

Бинарный алгоритм Евклида

```
function euclid_bin(a::Int, b::Int)::Int
    g = 1
    while a % 2 == 0 && b % 2 == 0
```

```

    a ÷= 2
    b ÷= 2
    g *= 2
end
u, v = a, b
while u != 0
    if u % 2 == 0
        u ÷= 2
    elseif v % 2 == 0
        v ÷= 2
    elseif u >= v
        u -= v
    else
        v -= u
    end
end
return g * v
end

println(euclid_bin(a, b))

```

Расширенный алгоритм Евклида

```

function euclid_ext(a::Int, b::Int)::Tuple{Int,Int,Int}
    if a == 0
        return b, 0, 1
    else
        div, x, y = euclid_ext(b % a, a)
    end
end

```

```

        return div, y - (b ÷ a) * x, x
    end
end

println(euclid_ext(a, b))

```

Шифрование с помощью решеток

```

function euclid_bin_ext(a::Int, b::Int)::Tuple{Int,Int,Int}
    g = 1
    while a % 2 == 0 && b % 2 == 0
        a ÷= 2
        b ÷= 2
        g *= 2
    end
    u, v = a, b
    A, B, C, D = 1, 0, 0, 1
    while u != 0
        if u % 2 == 0
            u ÷= 2
            if A % 2 == 0 && B % 2 == 0
                A ÷= 2
                B ÷= 2
            else
                A = (A + b) ÷ 2
                B = (B - a) ÷ 2
            end
        elseif v % 2 == 0

```

```

    v ÷= 2
    if C % 2 == 0 && D % 2 == 0
        C ÷= 2
        D ÷= 2
    else
        C = (C + b) ÷ 2
        D = (D - a) ÷ 2
    end
elseif u >= v
    u -= v
    A -= C
    B -= D
else
    v -= u
    C -= A
    D -= B
end
end
return g * v, C, D
end

println(euclid_bin_ext(a, b))

```

Расширенный бинарный алгоритм Евклида

```

char_sequence = 1
used_positions = String[]
for char in sanitized_text

```

```

placed = false
for i in 1:(2*matrix_dim), j in 1:(2*matrix_dim)
    if encryption_matrix[i, j] == string(char_sequence) && !placed
        position_key = string(i, ",", j)
        if !(position_key in used_positions)
            encryption_matrix[i, j] = string(char)
            push!(used_positions, position_key)
            placed = true
        end
    end
end
char_sequence += 1
if char_sequence > matrix_dim^2
    char_sequence = 1
    empty!(used_positions)
end
end

```


Заключение

В данной лабораторной работе были реализованы вычисления наибольшего общего делителя