

---

## Front matter

title: "Математические основы защиты информации и информационной безопасности" subtitle: "Отчёт по лабораторной работе №4:Вычисление наибольшего общего делителя" author: "Бекбузарова Роза"

## Generic otions

lang: ru-RU toc-title: "Содержание"

## Bibliography

bibliography: bib/cite.bib csl: pandoc/csl/gost-r-7-0-5-2008-numeric.csl

## Pdf output format

toc: true # Table of contents toc-depth: 2 lof: true # List of figures lot: true # List of tables fontsize: 12pt  
linestretch: 1.5 papersize: a4 documentclass: scrreprt

## l18n polyglossia

polyglossia-lang: name: russian options: - spelling=modern - babelshorthands=true polyglossia-otherlangs:  
name: english

## l18n babel

babel-lang: russian babel-otherlangs: english

## Fonts

mainfont: PT Serif romanfont: PT Serif sansfont: PT Sans monofont: PT Mono mainfontoptions: Ligatures=TeX  
romanfontoptions: Ligatures=TeX sansfontoptions: Ligatures=TeX,Scale=MatchLowercase monofontoptions:  
Scale=MatchLowercase,Scale=0.9

## Biblatex

biblatex: true biblio-style: "gost-numeric" biblatexoptions:

- parenttracker=true
- backend=biber
- hyperref=auto
- language=auto
- autolang=other\*
- citestyle=gost-numeric

## Pandoc-crossref LaTeX customization

figureTitle: "Рис." tableTitle: "Таблица" listingTitle: "Листинг" lolTitle: "Листинги"

## Misc options

indent: true header-includes:

- \usepackage{indentfirst}
- \usepackage{float} # keep figures where there are in the text
- \floatplacement{figure}{H} # keep figures where there are in the text
- \usepackage{unicode-math}
- \setmathfont{Latin Modern Math}

---

## Цель работы

---

Основной целью работы является реализовать разными алгоритмами вычисление наибольшего общего делителя

## Выполнение лабораторной работы

---

### Алгоритм Евклида

```
[22]: # Классический алгоритм Евклида
function gcd_euclid(a::Int, b::Int)
    r0, r1 = a, b
    while r1 != 0
        r0, r1 = r1, r0 % r1
    end
    return r0
end
println("gcd_euclid(48, 18) = ", gcd_euclid(48, 18))

gcd_euclid(48, 18) = 6
```

### Бинарный алгоритм Евклида

```
[24]: function gcd_binary(a::Int, b::Int)
    if a == 0
        return b
    elseif b == 0
        return a
    end
    shift = 0
    # Убираем общие факторы 2
    while (a & 1 == 0) && (b & 1 == 0)
        a >>= 1
        b >>= 1
        shift += 1
    end
    # Делим a на 2 пока оно чётное
    while (a & 1) == 0
        a >>= 1
    end
    while b != 0
        # Делим b на 2 пока оно чётное
        while (b & 1) == 0
            b >>= 1
        end
        # Обеспечиваем a <= b
        if a > b
            a, b = b, a
        end
        b -= a
    end
    return a << shift
end
println("gcd_binary(48, 18) = ", gcd_binary(48, 18))

gcd_binary(48, 18) = 18
```

### Расширенный алгоритм Евклида

```
[26]: # 1. Расширенный алгоритм Евклида
function extended_euclid(a::Int, b::Int)
    r0, r1 = a, b
    x0, x1 = 1, 0
    y0, y1 = 0, 1

    while r1 != 0
        q = r0 ÷ r1
        r0, r1 = r1, r0 - q * r1
        x0, x1 = x1, x0 - q * x1
        y0, y1 = y1, y0 - q * y1
    end

    return r0, x0, y0 # gcd, x, y
end

a, b = 12, 244
println("Расширенный алгоритм Евклида:")
g, x, y = extended_euclid(a, b)
println("gcd($a, $b) = $g, x = $x, y = $y => $a*$x + $b*$y = $(a*x + b*y)")

Расширенный алгоритм Евклида:
gcd(12, 244) = 4, x = -20, y = 1 => 12*-20 + 244*1 = 4
```

## Расширенный бинарный алгоритм Евклида

```

        A = (A + b) >> 1
        B = (B - a) >> 1
    end
end
# Делим v на 2 пока оно четное
while (v & 1) == 0
    v >>= 1
    if (C & 1 == 0) && (D & 1 == 0)
        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
    C -= A
    D -= B
end
gcd = v * g
x = C
y = D
return gcd, x, y
end

a, b = 9, 108
println("\nРасширенный бинарный алгоритм Евклида:")
g2, x2, y2 = extended_binary_euclid(a, b)
println("gcd($a, $b) = $g2, x = $x2, y = $y2 => $a*$x2 + $b*$y2 = $(a*x2 + b*y2)")

Расширенный бинарный алгоритм Евклида:
gcd(9, 108) = 9, x = 13, y = -1 => 9*13 + 108*-1 = 9
```

## Вывод

В ходе выполнения лабораторной работы было реализовано разными алгоритмами вычисление наибольшего общего делителя.