

Лабораторная работа №4

Вычисление наибольшего общего делителя

Кубасов В.Ю.

Вводная часть

- Алгоритм Евклида применяется в широком спектре задач, а также является одним из немногих оптимальных алгоритмов нахождения НОД.
- Алгоритм Евклида используется в множестве других математических алгоритм (например, нахождение НОК).

Реализовать предложенные вариации алгоритма Евклида: - Алгоритм Евклида - Бинарный алгоритм Евклида - Расширенный алгоритм Евклида - Расширенный бинарный алгоритм Евклида

Выполнение работы



Рис. 1: Евклид

Алгоритм Евклида

```
println("Введите a");  
a = parse(Int, chomp(readline()));  
println("Введите b");  
b = parse(Int, chomp(readline()));  
  
ri_1 = a; ri = b; i = 1;  
  
while (true)  
    riplus1 = ri_1 % ri;  
    if (riplus1 == 0)  
        break;  
    end;  
    ri_1 = ri;  
    ri = riplus1;
```

Бинарный алгоритм Евклида

```
g = 1;
```

```
while ((a % 2 == 0) && (b % 2 == 0))
```

```
    a /= 2;
```

```
    b /= 2;
```

```
    g *= 2;
```

```
end;
```

```
u = a;
```

```
v = b;
```


Бинарный алгоритм Евклида

```
while (u % 2 == 0)
    u /= 2;
end;
```

```
while (v % 2 == 0)
    v /= 2;
end;
```

```
while (u != 0)
    if (u >= v)
        u = u - v;
    else
        v = v - u;
    end;
```

Расширенный алгоритм Евклида

```
ri_1 = a; ri = b; i = 1;
```

```
xi_1 = 1; xi = 0;
```

```
yi_1 = 0; yi = 1;
```

```
while (true)
```

```
    riplus1 = ri_1 % ri;
```

```
    q = (ri_1 - riplus1) / ri;
```

```
    xiplus1 = xi_1 - q * xi;
```

```
    yiplus1 = yi_1 - q * yi;
```

```
    if (riplus1 == 0)
```

```
        break;
```

```
    end;
```

```
    ri_1 = ri;
```

```
    ri = riplus1;
```

Расширенный бинарный алгоритм Евклида

```
g = 1;
```

```
while ((a % 2 == 0) && (b % 2 == 0))
```

```
    a /= 2;
```

```
    b /= 2;
```

```
    g *= 2;
```

```
end;
```

```
u = a; v = b;
```

```
As = 1; Bs = 0; C = 0; D = 1;
```

```
while (u % 2 == 0)
```

```
    u /= 2;
```

Расширенный бинарный алгоритм Евклида

```
while (v % 2 == 0)
    v /= 2;
    if ((C % 2 == 0) && (D % 2 == 0))
        C /= 2;
        D /= 2;
    else
        C += b; C /= 2;
        D -= a; D /= 2;
    end;
end;

if (u >= v)
    u -= v;
    As -= C;
```

Выводы

Реализовали предложенные вариации алгоритма Евклида: - Алгоритм Евклида - Бинарный алгоритм Евклида - Расширенный алгоритм Евклида - Расширенный бинарный алгоритм Евклида