

НОД, НОК

Наибольший общий делитель

НОД чисел a и b – наибольшее число, являющееся делителем этих двух чисел.

$$\text{НОД}(7,14)=7$$

$$\text{НОД}(15,5)=5$$

$$\text{НОД}(30,-10)=10$$

Алгоритмы нахождения НОД

Алгоритм Евклида

```
int NOD(int a,int b) //Найдите ошибку
{
    while(a!=0 && b!=0)
    {
        if(a>=b) a=a%b;
        else b=b%a;
    }
    return a+b;
}
```

Алгоритмы нахождения НОД

Бинарный алгоритм Евклида

```
long gcd_binary(long a, long b)
{
    a=abs(a); b=abs(b);
    if (a==b) return a;
    else if (a==0) return b;
    else if (b==0||a==1) return a;
    else if (b==1) return b;
    else if (a%2==0&&b%2==0) return 2*gcd_binary(a/2,b/2);
    else if (a%2==0&&b%2!=0) return gcd_binary(a/2,b);
    else if (a%2!=0&&b%2==0) return gcd_binary(b/2, a);
    else if (a<b) return gcd_binary((b-a)/2, a);
    else return gcd_binary((a-b)/2, b);
}
```

Алгоритмы нахождения НОД

Расширенный алгоритм Евклида

```
int gcd (int a, int b, int & x, int & y) {  
    if (a == 0) {  
        x = 0; y = 1;  
        return b;  
    }  
    int x1, y1;  
    int d = gcd (b%a, a, x1, y1);  
    x = y1 - (b / a) * x1;  
    y = x1;  
    return d;  
}
```

НОК

Наименьшее общее кратное чисел a и b – такое наименьшее число, которое делится на оба эти числа без остатка.

$$\text{lcm}(a, b) = \frac{|ab|}{\text{gcd}(a, b)}$$