

написать ф-ию по поиску
НОД(наибольший общий делитель) чисел a,b

91 49
91/49 = 1 (42)
49/42 = 1 (7)
42/7 = 6 (0)

a,b

b=1000000000

b проверок = 1000000000 штук

log₂(b)=30 штук

a,b=

$nod * nok = a * b \mid / nod$

nok = $a * b / nod$ - наименьший общий знаменатель

12=2*2*3
20=2*2*5
НОК=2*2*3*5
НОД=2*2

```
#include <iostream>
#include <cstdio>
#include <cstdlib>
#include <ctime>
#define SIZE 12
using namespace std;
void fillmass2(long long int a,long long int b)
{
    clock_t start_time=clock();
    long long int i=1,nod;
    while (i<=a && i<=b)
    {
        if (b%i==0 && a%i==0)
        {
            nod=i;
        }
        i++;
    }
    printf("%lld\n",nod);
    clock_t finish_time=clock();
    printf("time=%d\n",finish_time-start_time);
}
void evklid(long long int a,long long int b)
{
    long long int m,m2,o;
    if (a>b)
    {
        m=a;
        m2=b;
    }
    else
    {
        m=b;
        m2=a;
    }
    while (m2!=0)
    {
        o=m%m2;
        m=m2;
        m2=o;
    }
    printf("m=%lld\n",m);
}
void evklid_2(long long int a,long long int b)
{
    while (a!=0 && b!=0)
    {
        if (a>b)
        {
            a=a-b;
        }
        else
        {
            b=b-a;
        }
    }
    printf("%lld\n",a+b);
}
int main()
{
    long long int x=123999099999,y=321999999999;
    afillmass2(x,y);
    evklid(x,y);
    evklid_2(x,y);
}
```

Первый алгоритм



Как-то раз древнегреческого
геометра Евклида спросили: - Что
бы ты предпочел - два целых
яблока или четыре половинки? -
Конечно, четыре половинки. - А
почему? Это ведь одно и то же. -
Отнюдь. Выбирая два целых
яблока, как я узнаю, червивые они
или нет?

Алгоритм Евклида – это
алгоритм нахождения НОД двух
целых положительных чисел

основная теорема арифметики

Любое натуральное число **ЕДИНСТВЕННЫМ** образом
РАСКЛАДЫВАЕТСЯ в произведение простых чисел

1234762462134623467=1213*123*213=35751*32423