```
#pragma GCC optimize(3,"Ofast","inline")
#pragma GCC optimize("Ofast")
#pragma GCC optimize("unroll-loops")
#pragma GCC target("sse,sse2,sse3,ssse3,sse4,popcnt,abm,mmx,avx,tune=native")
#include <bits/stdc++.h>
#define mp make_pair
#define ms(a,b) memset(a,b,sizeof(a))
#define maxn 1000007
typedef long long LL;
using namespace std;
inline int read()
{
    int X=0,w=1; char ch=0;
    while(ch<'0' || ch>'9') {if(ch=='-') w=-1;ch=getchar();}
    while(ch>='0' && ch<='9') X=(X<<3)+(X<<1)+ch-'0',ch=getchar();
    return X*w;
}
inline void write(int x) {
    if(x < 0)putchar('-'),x=-x;
    if (x > 9)write(x / 10);
    putchar(x % 10 + 48);
}
int main()
{
#ifndef ONLINE_JUDGE
freopen("1.in","r",stdin);
#endif
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    system("pause");
    return 0;
}
```

```
#include<iostream>
#include<iomanip>
#include<string>
#include <functional>
#include <sstream>
#include <numeric>
#include <limits>
#include <complex>
#include <bitset>
#include <memory>
#include <typeinfo>
#include <utility>
#include<cstdio>
#include<cmath>
#include<cstdlib>
#include<cstring>
#include <csignal>
#include <ctime>
#include <cctype>
```

```
#include<algorithm>
#include <vector>
#include <iterator>
#include<list>
#include<stack>
#include<queue>
#include<deque>
#include<set>
#include<unordered_map>
#define mp make_pair
#define ms(a,b) memset(a,b,sizeof(a))
#define maxn 510
typedef long long LL;
using namespace std;
inline int read()
{
    int X=0,w=1; char ch=0;
    while(ch<'0' || ch>'9') {if(ch=='-') w=-1;ch=getchar();}
    while(ch>='0' && ch<='9') X=(X<<3)+(X<<1)+ch-'0',ch=getchar();
    return X*w;
}
inline void write(int x) {
    if(x < 0)putchar('-'),x=-x;
    if (x > 9)write(x / 10);
    putchar(x % 10 + 48);
}
int main()
{


  system("pause");
  return 0;
}
```

```
int gcd(int a, int b)
{
    while (b ^= a ^= b ^= a %= b)
        ;
    return a;
}
```

```
int gcd(int a, int b)
{
    while (b != 0)
    {
        a %= b;
        b ^= a;
        a ^= b;
        b ^= a;
    }
    return a;
}
```