给两个数 p, y;   
求 p到y的区间中 不是2到p之间的倍数的最大数

明显可以知道如果存在这个数   
最大的数一定大于等于比y小的那个素数 因为素数与素数之间最大不超过1e3（在1e9的范围内）   
那么最大的所用时间大概就是1e3∗sqrt(1e9)

#include <bits/stdc++.h>

using namespace std;

typedef long long ll;

ll p,y;

bool is\_prime(int x)

{

for(int i=2;i<=sqrt(x)&&i<=p;i++)

if(x%i==0)

return 0;

return 1;

}

int main()

{

//freopen("input.txt","r",stdin);

cin>>p>>y;

for(int i=y;i>p;i--)

if(is\_prime(i))

{

cout<<i<<endl;

return 0;

}

cout<<-1<<endl;

return 0;

}