[POJ3090]Visible Lattice Points

Algorithm	欧拉函数 线性筛 质数
Created	@Jul 11, 2020 4:07 PM
Difficulty	普及+/提高
Related to 近期更新 (Property)	[POJ3090]Visible Lattice Points
URL	http://poj.org/problem?id=3090

备注：最近学会了$\KaTeX$语法，开始认真写题解了，学习笔记也开始启动

题目链接：

3090 -- Visible Lattice Points

欢迎参加IJCAI 2020麻将智能体竞赛，大奖等你拿！ Welcome to IJCAI 2020 Mahjong AI competition with amazing prizes! | 北京大学《ACM/ICPC大学生程序设计竞赛训练》暑期课面向全

http://poj.org/problem?id=3090

题目大意：

有一个$n*n$的矩形（有$n*n$个点）左下角坐落在坐标系的原点上，求在原点望去能看到几个点？

当$n(n \geq 2)$点共线时，第一个点只能看到第二个点（第二个点把后面的都掩盖住了）

题解：

先思考一下，什么情况下两个点与原点三点共线？

没思考请不要往下看（没思考出来没关系）。

五重分隔线……

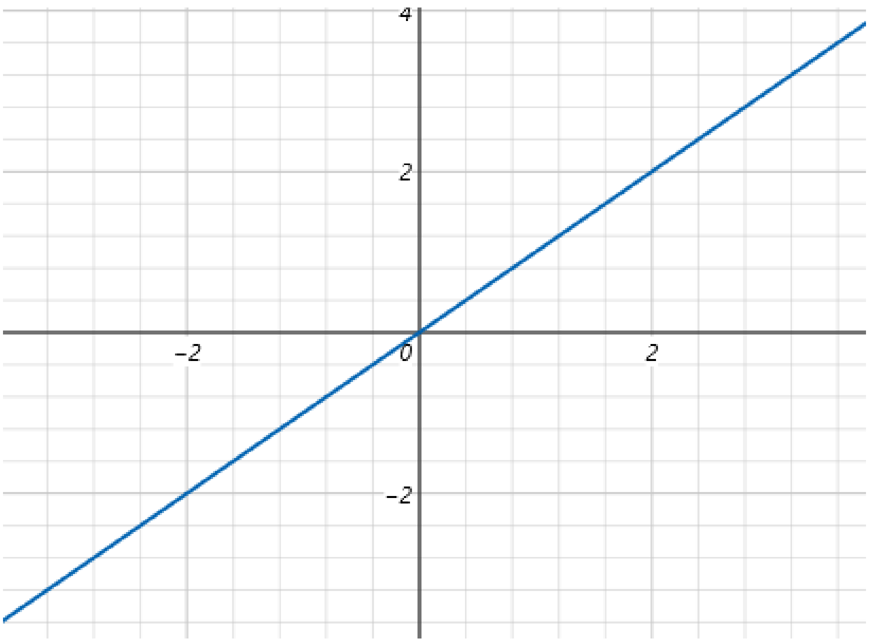
再来五个吧。

~~额，我不是来教五线谱的……~~

正经一下（敲黑板）

设点$a(x_1,y_1)$，$b(x_2,y_2)$，则当a,b在同一个正比例函数上时，与原点共线。

正比例函数大概长这样：



其实，过原点的直线都是正比例函数（这是废话😒）

既然是正比例函数，那么$\frac{y_1}{x_1}=\frac{y_2}{x_2}$。也可以说，只要满足$\frac{y_1}{x_1}=\frac{y_2}{x_2}$，$a$与$b$就在同一直线上，在原点就不可能同时看到$a$和$b$。

那么，我们就需要使$x1,y1$互质，也就是说，要使$gcd(x1,y1)=1$(gcd是最大公约数)。

介绍一下欧拉函数

欧拉函数的定义：

$$\varphi(N)=\sum_{i=2}^{N}[i,N\text{互质}]$$

翻译一下，就是$(1,N]$中与N互质的整数的个数。

介绍几个欧拉函数的性质：

①若i,j互质，则

$$\varphi(ij)=\varphi(i)\varphi(j)$$

这也是积性函数的典型性质兼判定定理，这在学习笔记中会讲到（首页上方图片下面有学习笔记的链接）

②若i是质数，则

$$\varphi(i)=i-1$$

有质数的定义得，2到i中只有$i|i$，因此该性质显然成立。

③若i是质数，$p|n$且$p^2|n$，则

$$\varphi(n)=p\varphi(\frac{n}{p})$$

这里就不证明了（如果心情好的话学习笔记里也许会证明😇)

回到题目

转换一下题意，若结果为ans，则

$$ans=\sum_{i=2}^{n}\sum_{j=2}^{n}[i,j\text{互质}]+\text{特例}$$

根据欧拉函数的定义再转换一下

$$ans=\sum_{i=2}^{n}2\varphi(i)+\text{特例}$$

因为原式中i,j可以互换，因此是欧拉函数的两倍。

特例就是$(0,1),(1,0),(1,1)$。那么，

$$ans=\sum_{i=2}^{n}2\varphi(i)+3$$

因此，用线性筛（学习笔记里一定讲）预处理2到n的所有数的欧拉函数值，即可计算。

那么，

附上AC代码：

```
//
// Created by admin on 2020/7/11.
//
#include <iostream>
using namespace std;
int prime[1500],phi[1500],c,mn=1001,n,tot;
bool vis[1500];
int main()
{
    cin>>c;
    for(int i=2;i<=mn;i++)  //线性筛取欧拉函数
    {
        if(!vis[i])
        {
            phi[i]=i-1;
            tot++;
            prime[tot]=i;
        }
        for(int j=1;j<=tot&&i*prime[j]<=mn;j++)
        {
            vis[i*prime[j]]=1;
            if(i%prime[j]==0)
            {
                phi[i*prime[j]]=phi[i]*prime[j];  //性质三
                break;
            }
            phi[i*prime[j]]=phi[i]*(prime[j]-1);  //积性函数性质
        }
    }
    for(int jj=1;jj<=c;jj++)
    {
```

```
        int ans=0;
        cin>>n;
        for(int i=2;i<=n;i++)
            ans+=phi[i];
        ans*=2;ans+=3;
        cout<<jj<<" "<<n<<" "<<ans<<endl;
    }
    return 0;
}
```

其实，2010年的NOI也有类似的题目，在这里：

[NOI2010]能量采集
栋栋有一块长方形的地，他在地上种了一种能量植物，这种植物可以采集太阳光的能量。在这些植物采集能量后，栋栋再使用一个能量汇集机器把这些植物采集到的能量汇集到一起。 栋栋的植物种得非常整齐，一共有 n 列，每列有 m 棵，植物的横竖间距都一样，因此对于每一棵植物，栋栋可以用一个坐标 (x, y)
https://www.luogu.com.cn/problem/P1447

只是哪里的$n \leq 10^5$，要用到莫比乌斯反演，后面学习笔记会讲。