Rabbit hunt [POJ - 2606](https://vjudge.net/problem/14543/origin)

Linearity [POJ - 2780](https://vjudge.net/problem/17526/origin)

一样的题，用一份代码都能过

题意简述：输入n，输入n个整数对，即n个坐标点，问最多共线点数是多少。

就是枚举两个点（ij）然后看其他点（k）是否在他们确定的线上

(j,y-i.y)/(j.x-i.x)=(k.y-j.y)/(k.x-j.x)

#include<iostream>

#include<stdio.h>

#include<map>

#include<cstring>

#include<queue>

#include<algorithm>

#include<iomanip>

using namespace std;

struct node

{

int x,y;

};

node a[1010];

int main()

{

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

int n;

while(~scanf("%d",&n) && n)

{

for(int i=0;i<n;i++)

scanf("%d%d",&a[i].x,&a[i].y);

int ans(0);

for(int i=0;i<n;i++)

for(int j=i+1;j<n;j++)

{

int cnt(0);

for(int k=j+1;k<n;k++)

if((a[j].x-a[i].x)\*(a[k].y-a[i].y)==(a[j].y-a[i].y)\*(a[k].x-a[i].x))

//if((a[j].x-a[i].x)\*(a[k].y-a[j].y)==(a[j].y-a[i].y)\*(a[k].x-a[j].x))也可以

cnt++;

ans=max(ans,cnt);

}

printf("%d\n",ans+2);//注意加上起点和终点这两个点，也就是以他们为基准来判断其他点是否在这条线上的这两个点

}

return 0;

}