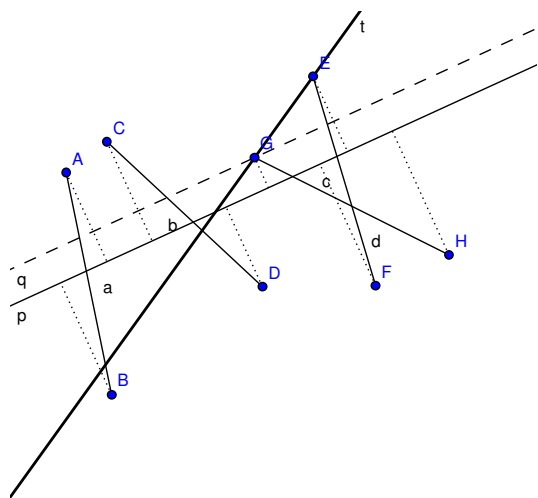


АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА ОТСЕЧКИ



Ще покажем, че ако съществува права, инцидентна с няколко отсечки, съществува и права, минаваща през поне два края на тези отсечки, инцидентна със същите отсечки. Нека правата p е инцидентна с няколко отсечки. Ако p минава през поне два края на отсечки – няма какво да доказваме. Ако p не минава през нито един край, можем, например, да я „преместим“ успоредно до най-близък край (край на отсечка, от който разстоянието до p е минимално). Новата права q , разбира се, също е инцидентна с всички отсечки. Ако минава през поне още един край – q е търсената. В противен случай започваме да въртим правата q около точката – край на отсечка, през който минава, докато за пръв път мине и през друг край. Новата права t е инцидентна с всички отсечки и минава през поне два края на отсечки. Ако p минава вече през един край, просто „пропускаме“ транслацията и извършваме само описаната ротация. Така задачата допуска изчерпващ алгоритъм по всички двойки зададени краища.

Функцията **ccw** определя ориентацията на три точки в равнината. С помощта на булевата функция **check** проверяваме дали права, минаваща през точки a и b , пресича отсечка с краища точките c и d (във вътрешни или в крайни точки). Разглеждат се три възможности – едната е правата да минава през точка c ($ccw(a, b, c)=0$); другата – да минава през точка d ($ccw(a, b, d)=0$); третата възможност е двете точки да лежат в различни полуравнини спрямо правата ($ccw(a, b, c)$ и $ccw(a, b, d)$ са с различни знаци). С функцията **numsec** броим колко от дадените отсечки пресича права, минаваща през точките a и b . В главната програма намираме максималния брой на тези отсечки.

```
#include<iostream>
using namespace std;
int n;
struct p{int x; int y;} c[200], d[200];

int ccw(p p0, p p1, p p2)
{
int dx1=p1.x-p0.x;
int dy1=p1.y-p0.y;
int dx2=p2.x-p0.x;
int dy2=p2.y-p0.y;
return dx1*dy2-dy1*dx2;
}

bool check(p a, p b, p c, p d)
```

```

{
    int e1 = ccw(a,b,c);
    int e2 = ccw(a,b,d);
    if(e1==0) return true;
    if(e2==0) return true;
    if(((e1<0)&&(e2>0))||((e1>0)&&(e2<0))) return true;
    return false;
}

int numsec(p a, p b)
{
    int num=0;
    if((a.x==b.x)&&(a.y==b.y)) return num;

    for(int k=1;k<=n;k++)
        if(check(a,b,c[k],d[k])) num++;
    return num;
}

int main()
{
    cin >> n;
    for(int i=1;i<=n;i++)
        cin >> c[i].x >> c[i].y >> d[i].x >> d[i].y;

    int r=0;
    for(int i=1;i<=n;i++)
        for(int j=i+1;j<=n;j++)
        {
            int num;
            num=numsec(c[i],c[j]);
            if(r<num) r=num;
            num=numsec(c[i],d[j]);
            if(r<num) r=num;
            num=numsec(d[i],c[j]);
            if(r<num) r=num;
            num=numsec(d[i],d[j]);
            if(r<num) r=num;
        }

    if(r==0)
    { r=1;
      if(n>1) r=2;
    }

    cout << r << endl;
}

```

Автор: Зорница Дженкова