

CHUYÊN ĐỀ HÌNH HỌC

A. PHẦN NỘI DUNG

I. LÝ THUYẾT

Thuật giải cho các bài toán hình học thường không khó, nó yêu cầu học sinh phải làm việc cẩn trọng, tỉ mỉ và tập trung vì nếu không rất có thể dẫn đến những sai lầm. Mặc dù kiến thức hình học các em được học trong bộ môn Toán. Nhưng trước tiên ta đi nhắc lại một số khái niệm cơ bản:

1. Biểu diễn hình học trên máy tính.

Trong chuyên đề này, tôi thống nhất cách biểu diễn những khái niệm cơ bản như điểm, đường thẳng, đa giác như sau:

- Điểm

```
struct Point
```

```
{
```

```
    int x;
```

```
    int y;
```

```
};
```

- Đường thẳng:

```
struct LineSegment
```

```
{
```

```
    P1, P2: Point;
```

```
};
```

- Véc tơ

```
Struct Vector
```

```
{
```

```
    P1, P2: Point;
```

```
};
```

- Đa giác:

```
Point Polygon[n];
```

Để thuận lợi thì khi biểu diễn đa giác ta nên thêm hai đỉnh ở đầu và cuối: đỉnh 0 bằng đỉnh n và đỉnh n + 1 bằng đỉnh 1.

2. Điểm, đoạn thẳng - đường thẳng, diện tích đa giác.

2.1. Quan hệ giữa các điểm - hàm CCW.

a) Khoảng cách giữa hai điểm:

```
Double Dist(Point P1, Point P2){
```

```
    return sqrt(abs(P1.x - P2.x)*(P1.x-P2.x) + (P1.y-P2.y)*(P1.y-P2.y)); //bỏ abs
```

```
}
```

```
int CCW(Point p, Point q, Point r)
```

```

{
    int val = (q.y - p.y) * (r.x - q.x) - (q.x - p.x) * (r.y - q.y);
    if (val == 0) return 0;
    return (val > 0)? 1: 2;
}

```

b) Hệ số góc của đường thẳng đi qua hai điểm:

Cho hai điểm $A(x_A, y_A), B(x_B, y_B)$. Nếu AB không song song với trục Oy thì hệ số góc của đường thẳng AB là:

$$\frac{y_A - y_B}{x_A - x_B}$$

c) Kiểm tra ba điểm có thẳng hàng hay không?

Giả sử chúng ta được cho trước ba điểm $A(x_A, y_A), B(x_B, y_B)$ và $C(x_C, y_C)$

- Viết phương trình đường thẳng AB, AC
- Giả sử $AB: a_1x + b_1y + c_1 = 0$;
 $AC: a_2x + b_2y + c_2 = 0$ là các phương trình đã tối giản các hệ số và $a_1, a_2 > 0$.
- Khi đó, A, B và C thẳng hàng khi và chỉ khi $c_1 = c_2$.

d) Kiểm tra hai đoạn thẳng giao nhau hay không

```

bool onSegment(Point p, Point q, Point r)
{
    if (q.x <= max(p.x, r.x) && q.x >= min(p.x, r.x) &&
        q.y <= max(p.y, r.y) && q.y >= min(p.y, r.y))
        return true;
    return false;
}

```

```

bool doIntersect(LineSegment LS1, LineSegment LS2)
{

```

```

    Point p1, Point q1, Point p2, Point q2;
    p1 = LS1.p1; q1 = LS1.p2;
    p3 = LS2.p1; q2 = LS2.p2;
    int o1 = CCW (p1, q1, p2);
        int o2 = CCW (p1, q1, q2);
        int o3 = CCW (p2, q2, p1);
        int o4 = CCW (p2, q2, q1);
    if (o1 != o2 && o3 != o4)
        return true;

```

```

        if (o1 == 0 && onSegment(p1, p2, q1)) return true;
        if (o2 == 0 && onSegment(p1, q2, q1)) return true;
        if (o3 == 0 && onSegment(p2, p1, q2)) return true;
        if (o4 == 0 && onSegment(p2, q1, q2)) return true;
        return false;
    }

```

e) Diện tích của đa giác :

```

int Area (Point p[n]){
    int s;
    p[n] = p[0];
    for(int i = 0 ; i <= n ; i++)
        s=s+(p[i].x*p[i+1].y-p[i].y*p[i+1].x) ;
    return abs(s/2) ;
}

```

2.2. Điểm trong đa giác:

a) Bài toán : Cho một đa giác không tự cắt, hãy kiểm tra xem một điểm có nằm trong đa giác hay không?

Tư tưởng cho bài toán này nói qua thì rất đơn giản và dễ hiểu: Từ điểm cần kiểm tra ta kẻ một tia bất kỳ, nếu tia đó giao với đa giác một số chẵn lần thì có nghĩa là nó nằm ngoài đa giác, một số lẻ lần thì nó nằm trong đa giác.

b) Cài đặt:

```

bool isInside(Point polygon[], int n, Point p)
{
    if (n < 3) return false;
    Point extreme = {INF, p.y};
    int count = 0, i = 0;
    do
    {
        int next = (i+1)%n;
        if (doIntersect(polygon[i], polygon[next], p, extreme))
        {
            if (orientation(polygon[i], p, polygon[next]) == 0)
                return onSegment(polygon[i], p, polygon[next]);
            count++;
        }
    }
}

```

```

        i = next;
    } while (i != 0);
    return count&1;
}

```

2.3. Bao lồi:

a) bài toán: Cho một tập các điểm trên mặt phẳng. Hãy xác định bao lồi của tập điểm.

b) Hướng dẫn thuật toán : Tham khảo chi tiết các thuật toán bao lồi tại <http://vnoi.info/wiki/translate/wcipeg/Convex-Hull>

II. BÀI TẬP LUYỆN TẬP

2.1. BÀI TẬP ỨNG DỤNG HỆ SỐ GÓC CỦA ĐƯỜNG THẲNG

Trong phần này, Tôi sẽ trình bày một số bài toán ứng dụng hệ số góc để giải bài toán hình học rất hiệu quả.

Bài 1. QBPOINT – Bộ ba điểm thẳng hàng

Trong các cuộc thi tin học, sự xuất hiện của những bài toán hình học làm đội tuyển CBQ khá lúng túng. Do đó thầy Thạch quyết định cho đội tuyển luyện tập các bài toán hình học. Bắt đầu từ điểm, thầy đưa ra bài toán.

Cho n điểm phân biệt trong mặt phẳng Oxy, hãy đếm số bộ 3 điểm thẳng hàng.

Dữ liệu đầu vào:

- Dòng thứ nhất ghi N là số điểm trên mặt phẳng.
- N dòng tiếp theo, mỗi dòng ghi tọa độ của một điểm.

Kết quả:

- Ghi ra một số duy nhất là số bộ ba điểm thẳng hàng.

Ví dụ:

QBPOINT . INP	QBPOINT . OUT
6 0 0 0 1 0 2 1 1 2 0 2 2	3

Giới hạn: $1 \leq N \leq 2000$, Tọa độ các điểm có trị tuyệt đối không quá 10000.

Nguồn: <https://vn.spoj.com/problems/QBPOINT>

Hướng dẫn thuật toán:

- Cố định một điểm (giả sử điểm A) rồi tính hệ số góc của đường thẳng đi qua điểm A và tất cả các điểm còn lại.
- Sắp xếp lại mảng hệ số góc
- Đếm số lượng số hệ số góc bằng nhau, nếu có m_i số hệ số góc k_i bằng nhau thì tăng kết quả lên $k_i * \frac{k_i - 1}{2}$.

- Giả code:

```
for(int i = 1; i ≤ N - 2; i++){
    for(j = i + 1 → N)
        Tính mạng hệ số góc slops[];
        Sắp xếp lại mảng slops[];
        Đếm trên mảng slops số lượng số hệ số góc giống nhau
        Cnt +=  $k_i * \frac{k_i - 1}{2}$ .
```

Cảm nhận: Bài toán kiểm tra ba điểm thẳng hàng là một bài toán cơ bản trong toán học và có nhiều ứng dụng trong tin học. Sử dụng hệ số góc để kiểm tra ba điểm có thẳng hàng hay không phải là một phương pháp phổ biến trong toán học, tuy nhiên trong các bài toán Tin học thì

Bài 2. BALLGMVN – VOI 2014 – Trò chơi với những viên bi

Trong một hội thi Ballgame, ban tổ chức chuẩn bị một bàn lớn. Trên mặt bàn có n bi xanh đánh số từ 1 đến n và n bi đỏ đánh số từ $n + 1$ đến $2n$. Mỗi trận đấu, các vận động viên sẽ chơi luân phiên nhau. Đến lượt chơi của mình, Hùng cần tìm 3 bi mà vị trí của chúng là thẳng hàng hanu và sao cho trong số đó có hai bi đỏ và 1 bi xanh (khi đó ăn được một bi đỏ), hoặc là có hai bi xanh và 1 bi đỏ (khi đó được ăn 1 bi xanh).

Yêu cầu: Cho biết tọa độ trên mặt phẳng tọa độ Đề-các của vị trí và màu của các bi hiện tại trên bàn, bạn hãy giúp Hùng chọn 3 bi để chơi.

Dữ liệu đầu vào:

- Dòng đầu ghi số nguyên dương n .
- Dòng thứ i trong số n dòng tiếp theo ghi hai số nguyên là hoành độ và tung độ trên mặt phẳng tọa độ Đề-các của vị trí đặt bi xanh với chỉ số i .
- Dòng thứ i trong số n dòng cuối cùng ghi hai số nguyên là hoành độ và tung độ trên mặt phẳng tọa độ Đề-các của vị trí đặt bi đỏ với chỉ số $n + i$.

Hoành độ và tung độ không vượt quá 10^6 , vị trí các bi là đôi một phân biệt.

Kết quả:

- Ghi ra 3 chỉ số của các viên bi mà Hùng cần chọn, nếu không thể chọn được 3 bi nào, ghi ra -1. Nếu có nhiều đáp án, ghi ra một đáp án bất kỳ.

Ví dụ:

BALLGMVN . INP	BALLGMVN . OUT
3	1 2 4
1 1	
2 2	
4 9	
3 3	
6 20	
8 100	

Giới hạn

- 30% số test có $n \leq 2$;
- 30% số test khác có $n \leq 100$.
- 40% số test còn lại có $n \leq 1000$.

Nguồn: <https://vn.spoj.com/problems/BALLGMVN/>

Hướng dẫn thuật toán:

- Chia bài toán thành hai trường hợp, trường hợp 1 xét những đường thẳng đi qua hai bi xanh một bi đỏ, trường hợp hai xét đường thẳng đi qua hai bi đỏ một bi xanh.
- Trường hợp: Xét đường thẳng đi qua hai xanh một bi đỏ:
- Cố định bi đỏ (giả sử là A)
- Xác định mảng hệ số góc của đường thẳng đi qua bi đỏ A và mọi bi xanh.
- Sắp xếp lại mảng hệ số góc, từ đó kiểm tra trên mạng hệ số góc có giá trị nào xuất hiện ít nhất hai lần thì bài toán tồn tại và ghi ra chỉ số các điểm, nếu không thì kiểm tra trường hợp hai.

Độ phức tạp của thuật toán là $O(n^2 \log n)$.

Bài 3: SWAGE

Phú ông có khu vườn trồng cây ăn trái, cây thứ i được trồng tại vị trí có tọa độ (x_i, y_i) . Phú ông thuê Bờm quản lý và chăm sóc vườn cây đôi lại Bờm được thu hoạch một số cây trong vườn cho riêng mình.

Phú ông biết Bờm không được học nhiều nên ra điều kiện làm khó là Bờm chỉ được thu hoạch những cây nằm trên cùng một đường thẳng. Bờm không biết làm thế nào để có thể thu hoạch được nhiều cây nhất mà vẫn thỏa điều kiện của Phú ông đưa ra.

Yêu cầu: Bạn hãy giúp Bờm xác định số cây nhiều nhất mà Bờm được thu hoạch.

Input: SWAGE.INP

- Dòng thứ nhất ghi số N là số cây trong vườn.
- Dòng thứ i trong N dòng tiếp theo ghi tọa độ của cây thứ i .

Output: SWAGE.OUT

- Một số duy nhất là số cây nhiều nhất mà Bờm được thu hoạch.

Giới hạn: $1 \leq N \leq 2000$. Tọa độ các điểm có trị tuyệt đối không quá 15000.

Ví dụ:

SWAGE . INP	SWAGE . OUT
6	3
0 0	
0 1	
0 2	
1 1	
2 0	
2 2	

Ràng buộc:

Subtask 1 (50% điểm): Giả thiết $N \leq 100$

Subtask 2 (50% điểm): Giả thiết $N \leq 2000$

Hướng dẫn thuật toán:

for ($i = 1 \rightarrow N - 1$) cố định một điểm

```

{
    for( $j = i + 1 \rightarrow N$ )
        Tính mảng hệ số góc slopes của đường thẳng đi qua điểm i và j.
    Sắp xếp lại mảng slopes
    Tìm giá trị xuất hiện nhiều nhất trên mảng slopes
    Cập nhật lại max
}

```

Độ phức tạp của thuật toán $O(n^2 \log n)$

Chi tiết code và test tham khảo tại:

Bài 4: LINES - Game of Lines (Trò chơi của những đường thẳng)

John đã thách thức Bessie một trò chơi như sau: John có một con cái bảng với các điểm được đánh dấu tại N vị trí phân biệt ($2 \leq N \leq 200$). Điểm thứ i có tọa độ nguyên x_i, y_i ($-1000 \leq x_i, y_i \leq 1000$).

Bessie có thể ghi một điểm bằng cách chọn hai điểm và vẽ một đường thẳng giữa hai điểm đó; tuy nhiên, Bessie không được phép vẽ một đường thẳng nếu cô ấy đã từng vẽ một đường thẳng song song với nó trước đó.

Yêu cầu: Hãy lập trình tính giúp Bessie số điểm lớn nhất mà cô ấy có thể rành được.

Dữ liệu:

- Sẽ có nhiều test. Với mỗi test, dòng đầu tiên ghi số N , và mỗi dòng tiếp theo của N dòng ghi một cặp số nguyên x_i, y_i . File kết thúc với trường hợp $N = 0$.

Kết quả:

- Mỗi dòng ghi kết quả của mỗi trường hợp.

Ví dụ:

Input:	Output:
4	4
-1 1	
-2 0	
0 0	
1 1	
0	

Bessie có thể vẽ các đường thẳng với hệ số góc: -1 , 0 , $\frac{1}{3}$, and 1 .

Nguồn: <https://www.spoj.com/problems/LINES/>

Hướng dẫn thuật toán:

- Duyệt đến mọi đường thẳng được tạo bởi hai điểm bất kì
- Tính hệ số góc của đường thẳng đi qua hai điểm đó
- Dùng map để lưu hệ số góc.
- Kết quả của bài toán bằng kích thước của map.

Tham khảo code chi tiết tại: <http://spojsolutionsall.blogspot.com/>

Bài 5. Line (Đường thẳng)

Trong tọa độ đề các, cho N đoạn thẳng được xác định bởi hai điểm (x_1, y_1) và (x_2, y_2) ($|x_1, y_1, x_2, y_2| < 10^8$), N đoạn thẳng này tạo thành N đường thẳng. Đếm số lượng tất cả các cặp gồm 2 đường thẳng được tạo thành chúng vuông góc và cặp đoạn thẳng đó có độ dài bằng nhau.

Dữ liệu vào:

- Dòng đầu là số nguyên N
- N dòng tiếp theo, mỗi dòng gồm 4 số nguyên x_1, y_1, x_2, y_2 lần lượt là tọa độ của 2 đầu mút

Kết quả:

- Một dòng duy nhất là số cặp tìm được thỏa mãn yêu cầu.

Line.inp	Line.out
6 1 1 0 2 2 1 2 0 1 1 1 2 1 1 2 1 2 2 3 2 -1 -1 0 -1	6

Ràng buộc:

- 50% test : $N < 10^3$
- 100% test: $N < 10^5$

Hướng dẫn thuật toán:

- Với mỗi đường thẳng được tạo bởi đoạn thẳng AB ta tính độ dài đoạn thẳng AB và véc tơ \overrightarrow{AB} (dùng pair<int, pair<int, int>> để lưu dữ liệu) lưu vào mảng vt.
- Sắp xếp lại mảng vt.
- Với mỗi phần tử trong mảng vt có dạng (d, (a, b))
- Dùng tìm kiếm nhị phân để đếm số lượng phần tử của mảng vt có dạng (d, (-b, a)) => tăng kết quả của bài toán.

Tham khảo chi tiết code và test của bài toán tại:

Bài 6. Góc nhìn (viewangle)(nguồn: codeforces.com)

Gần đây ở Flatland đã giới thiệu một kiểu kiểm tra mắt mới cho các tài xế tham gia thi lấy giấy phép lái xe. Việc kiểm tra này như sau: Trong mặt phẳng có các hình nộm đứng trên đó. Bạn cần phải nói giá trị của góc nhìn nhỏ nhất với đỉnh tại gốc tọa độ và tất cả các hình nộm đứng bên trong hoặc trên cạnh của góc này.

Cho trước tọa độ của các hình nộm trong mặt phẳng, bạn hãy viết chương trình tính giá trị nhỏ nhất của góc nhìn này.

Dữ liệu:

- Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 10^5$) – số lượng hình nộm.
- n dòng sau, mỗi dòng chứa 2 số nguyên x_i, y_i ngăn cách nhau bởi một dấu cách là tọa độ hình nộm thứ i ($-10^9 \leq x_i, y_i \leq 10^9$). Dữ liệu đảm bảo không có hình nộm nào đặt tại gốc tọa độ nhưng có thể có nhiều hơn một hình nộm ở cùng một vị trí.

Kết quả:

- Ghi ra một số thực với độ chính xác 6 chữ số thập phân sau dấu phẩy là kết quả của bài toán.

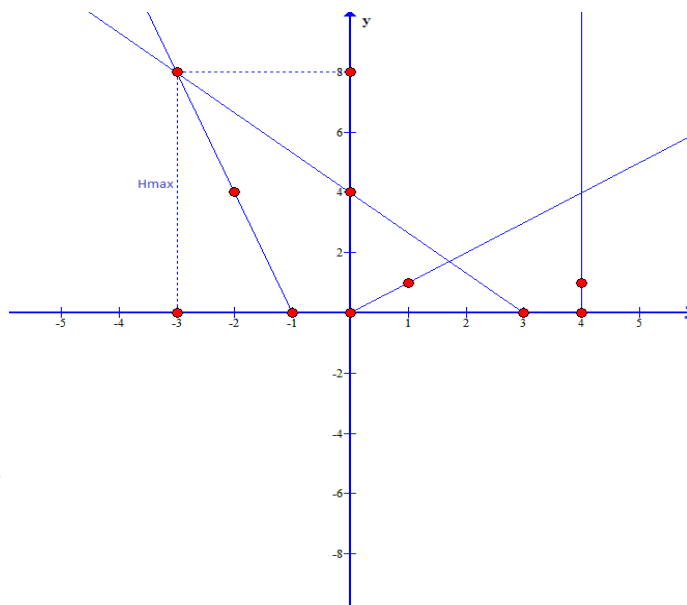
Ví dụ:

Viewangle.inp	Viewangle.out
2 2 0 0 2	90.000000
3 2 0 0 2 -2 2	135.000000
4 2 0 0 2 -2 0 0 -2	270.000000
2 2 1 1 2	36.869898

Hướng dẫn thuật toán:

- Với mỗi điểm $P[i]$ chúng ta tính được $\tan[i]$ là tan của góc tạo bởi tia $OP[i]$ và chiều dương của trục Ox .
- Sắp xếp lại dãy $\{\tan[i]\}$ theo chiều tăng dần.
- Kết quả là:

```
double res = (g[n] - g[1])/M_PI *  
180.0;  
  
for(i,2,n) res = min(res,360.0-  
(g[i] - g[i-1])/M_PI * 180.0);
```



Bài 7. Giao điểm caonhất

Trong lễ hội bắn pháo hoa năm nay. Tiết mục trình diễn ánh sáng trong lễ khai mạc của chủ nhà Đà Nẵng, có N tia laser được chiếu lên trời nhờ vào các đèn chiếu có công suất rất lớn, vì vậy các tia laser này có thể đi rất xa. Các tia laser được chiếu lên nằm trên cùng một mặt phẳng thẳng đứng nên nếu không có 2 tia laser nào song song với nhau thì 2 tia laser hoặc là cắt nhau hoặc là không cắt nhau. Các tia laser được biểu diễn bởi 3 số nguyên:

Một số là tọa độ X ở dưới đất của ngọn đèn chiếu, 2 số còn lại là tọa độ của một điểm nào đó thuộc tia laser này. Biết rằng không có 2 tia laser nào song song với nhau và cũng không có tia Laser nào trùng với mặt đất (mặt đất được coi như là đường thẳng $y = 0$). Không có đèn chiếu nào đặt cùng một vị trí trên trục tọa độ.

Yêu cầu: Các nhà tổ chức buổi trình diễn muốn bạn cho biết với các tia laser sẽ được chiếu lên trời như trong kế hoạch thì 2 tia laser nào cắt nhau tại điểm cao nhất.

Dữ liệu: Từ tệp văn bản 'H_MAX.INP' gồm:

- Dòng 1 ghi số nguyên N là số tia laser ($2 \leq n \leq 100000$).
- N dòng, mỗi dòng ghi 3 số nguyên x_i, z_i, t_i với ý nghĩa là tia laser thứ i qua 2 điểm $(x_i, 0)$ và (z_i, t_i) ($|x_i|, |z_i| < 10^6, 0 < t_i \leq 10^6$).

Kết quả: Ghi ra tệp văn bản 'H_MAX.OUT' như sau:

- Trong trường hợp không có 2 tia laser nào cắt nhau thì ghi ra duy nhất một số - 1.
- Nếu tồn tại 2 tia Laser cắt nhau thì ghi số H_{max} là độ cao lớn nhất giao điểm của 2 tia laser. (h_{max} được ghi với độ chính xác 3 chữ số sau dấu phẩy).

Ví dụ:

H_MAX.INP	H_MAX.OUT
2 -1 -24	-1
4 -1 -24 0 11 3 04 4 41	8.000

Thuật toán:

- **Thuật toán 1:**

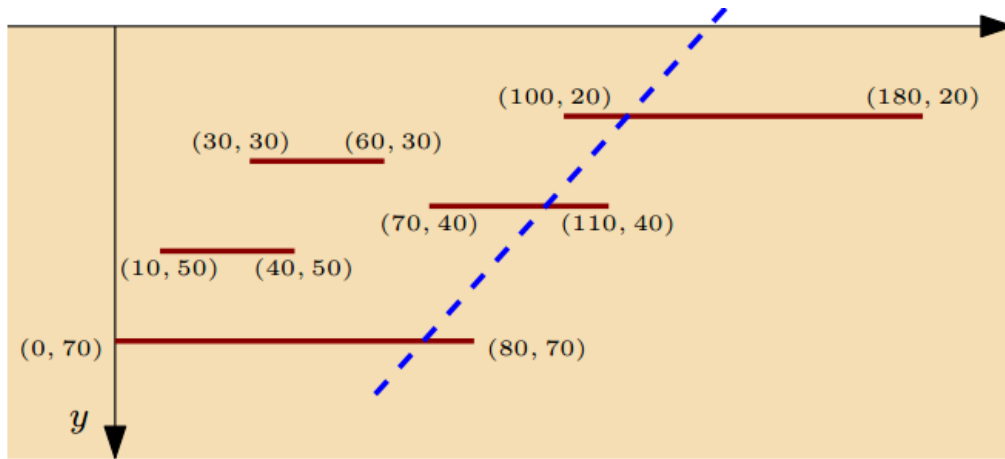
- Duyệt chọn ra 2 tia, tìm điểm giao và tính chiều cao. Trong các chiều cao đó chọn ra cặp 2 tia có chiều cao lớn nhất. Độ phức tạp thuật toán $\sim O(N^2)$. Làm cách này thí sinh có không quá 40% số điểm của bài thi.

- **Thuật toán 2:**

- Bạn nên chú ý tới giá trị N – số tia laser, nó rất lớn ($N \leq 10^5$). Ta gọi góc tạo bởi một tia laser với trục nằm ngang (chiều dương là chiều ngược chiều kim đồng hồ) là *góc của tia đó*.
- Nếu tia i và tia j giao nhau (với điều kiện góc của tia j là góc có giá trị gần với góc của tia i nhất) thì độ cao giao điểm của hai tia đó sẽ lớn hơn độ cao giao điểm của tia i với các tia khác tia j . Từ đó ta có thuật toán:
- Sắp xếp các tia theo thứ tự tăng dần (hoặc giảm dần) của góc tạo bởi các tia.
- Duyệt 1 lần theo thứ tự và xét chiều cao giao điểm của hai tia cạnh nhau theo thứ tự mảng đó.
- Lưu lại giá trị lớn nhất tìm được.
- Độ phức tạp thuật toán chủ yếu nằm ở bước sắp xếp các tia $\sim O(N \log N)$.

Bài 8. KHOANGĐAU- ACM/ICPC 2016-PROBLEM G:

Công ty khai thác dầu SÁNG TẠO đang cố gắng tìm cách đổi mới phương pháp khai thác dầu để được lợi nhuận tối đa. Quá trình khảo sát địa chất cho thấy các mỏ dầu nằm thành từng lớp với độ dài khác nhau trên bề mặt trái đất. Công ty muốn tạo một đường khoáng (đường đứt quãng trên hình vẽ) là đường thẳng sao cho xuất phát từ bề mặt trái đất đi qua được nhiều mỏ dầu nhất.



Hình 1. Hình mô phỏng từ ví dụ 1

Hiệu quả của đường khoang là tổng số dầu thu được từ những mỏ dầu mà đường khoang đi qua. Đường khoang vẫn có thể thu được dầu khi đi qua điểm đầu hoặc điểm cuối của mỏ dầu. Sản lượng của mỏ dầu chính bằng độ dài của mỏ dầu.

Hãy viết chương trình giúp công ty SÁNG TẠO xác định lượng dầu tối đa mà công ty thu được với 1 đường khoang.

Dữ liệu:

- Dữ liệu vào từ file KHOANGDAU.INP gồm
- Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 2000$), là số lượng mỏ dầu.
- N dòng tiếp theo mỗi dòng mô tả cho một mỏ dầu, gồm 3 số nguyên x_0, x_1, y xác định tọa độ điểm đầu và cuối của mỏ dầu là (x_0, y) và (x_1, y) .

$$(0 \leq |x_0|, |x_1| \leq 106, 1 \leq y \leq 106).$$

Kết quả:

- Ghi ra file KHOANGDAU.OUT một số nguyên duy nhất là sản lượng dầu tối đa mà đường khoang thu được.

VÍ DỤ:

KHOANGDAU.INP	KHOANGDAU.OUT
5 100 180 20 30 60 30 70 110 40 10 40 50 0 80 70	200
3 50 60 10 -42 -42 20	25

Hướng dẫn thuật toán (<https://icpc.baylor.edu/worldfinals/problems/icpc2016.pdf>):

Đây là bài cơ bản về tập hợp trong hình học.

Nếu tất cả các mỏ dầu đều nằm trên 1 đường ngang thì đường cắt chỉ lấy được dầu từ 1 mỏ dầu và chắc chắn phương án tối ưu là lấy từ mỏ dầu có trữ lượng lớn nhất. Ngược lại, thì tối thiểu đường cắt chúng ta có thể đi qua tối thiểu là 2 mỏ dầu bằng cách di chuyển đường cắt (sang trái, sang phải, xoay).

Duyệt toàn bộ với độ phức tạp $O(n^3)$. Trường hợp mỗi cặp điểm không nằm trên 1 hàng thì đường khoan sẽ đi qua hai điểm, kiểm tra tất cả các mỏ dầu mà đường khoan đi qua. Tuy nhiên cách này khá chậm.

Để cải tiến tốc độ ta sử dụng một đường quét xoay vòng, chọn điểm P bất kì mà đường cắt sẽ đi qua (chúng ta sẽ thực hiện lặp lại với tất cả các điểm P được chọn). các điểm không nằm trên cùng một mỏ dầu chứa P và đường thẳng đi qua P sẽ tạo thành 1 góc, sắp xếp theo độ dốc của các góc tạo bởi điểm P và các điểm khác với đường cắt. Nếu gặp điểm đầu của mỏ dầu, ta thêm giá trị trữ lượng dầu vào kết quả hiện tại. Nếu ta gặp điểm thứ hai, ta trừ đi trữ lượng dầu trong kết quả hiện tại. Giải thuật này chạy trong $O(n^2 \log n)$.

Khi cài đặt, cần cẩn thận trong việc sắp xếp dữ liệu theo độ dốc, xử lý đường quét, xây dựng đúng thủ tục thêm mỏ dầu và bớt mỏ dầu ra khỏi kết quả hiện tại. Đặc biệt cần lưu trữ độ dốc là những cặp số nguyên, nó khác với số thực.

Tham khảo code chi tiết để hiểu thuật toán:

Nguồn: <https://www.spoj.com/problems/FN16OIL/>

Kết luận:

Hệ số góc của đường thẳng kiểm tra ba điểm thẳng hàng là một phương pháp rất hiệu quả. Giúp chúng ta giải được nhiều bài toán hình học trong tin học một cách hiệu quả.