

# HÌNH HỌC

## A. LÝ THUYẾT

### 1. Véc tơ trong mặt phẳng tọa độ

Điểm trong mặt phẳng tọa độ cũng như véc tơ trong mặt phẳng luôn được biểu diễn bởi một cặp hai số thực. Do vậy chúng ta có thể khai báo kiểu dữ liệu sau để mô tả điểm cũng như véc tơ trên mặt phẳng

```
typedef pair<double,double> DD;
```

Các toán tử sau thực hiện phép cộng, trừ và phép nhân vô hướng một số thực với một véc tơ

```
DD operator +(DD u,DD v) {return DD(u.first+v.first,u.second+v.second);}
```

```
DD operator -(DD u,DD v) {return DD(u.first-v.first,u.second-v.second);}
```

```
DD operator *(double k,DD u) {return DD(k*u.first,k*u.second);}
```

### Tích chấm

Tích chấm (dot product) hay còn gọi là tích vô hướng của hai véc tơ  $\vec{u}$  và  $\vec{v}$ , ký hiệu là  $\vec{u} \cdot \vec{v}$  là một số thực được tính bằng tích độ dài của hai véc tơ  $\vec{u}$  và  $\vec{v}$  nhân với *cosin* của góc xen giữa hai véc tơ đó. Góc xen giữa hai véc tơ này là góc không định hướng, có số đo từ 0 đến  $\pi$ .

Biểu thức của tích chấm giữa hai véc tơ  $\vec{u} = (x_u, y_u)$ ,  $\vec{v} = (x_v, y_v)$  có thể diễn giải như sau:

$$\vec{u} \cdot \vec{v} = x_u \times x_v + y_u \times y_v$$

Toán tử dưới đây tính tích chấm của hai véc tơ:

```
double operator ^(DD u,DD v) {return u.first*v.first+u.second*v.second;}
```

### Tích chéo

Tích chéo (cross product) của hai véc tơ  $\vec{u}$  và  $\vec{v}$ , ký hiệu là  $\vec{u} \times \vec{v}$  là một số thực được tính bằng tích độ dài hai véc tơ  $\vec{u}$  và  $\vec{v}$  nhân với *sin* của góc xen giữa hai véc tơ đó. Góc xen giữa của hai véc tơ này là góc định hướng, có số đo từ  $-\pi$  đến  $\pi$ . Số đo mang dấu dương nếu chiều quay từ  $\vec{u}$  đến  $\vec{v}$  ngược chiều kim đồng hồ và mang dấu âm nếu chiều quay từ  $\vec{u}$  đến  $\vec{v}$  cùng chiều kim đồng hồ.

Công thức tính tích chéo của hai véc tơ  $\vec{u} = (x_u, y_u)$  và  $\vec{v} = (x_v, y_v)$  là

$$\vec{u} \times \vec{v} = x_u \times y_v - x_v \times y_u = \begin{vmatrix} x_u & y_u \\ x_v & y_v \end{vmatrix}$$

Toán tử sau thực hiện tích chéo:

```
double operator *(DD u,DD v) {return u.first*v.second-u.second*v.first;}
```

Về mặt hình học, giá trị tuyệt đối của tích chéo  $\vec{u} \times \vec{v}$  là diện tích của hình bình hành OABC trong đó O là gốc tọa độ  $\vec{OA} = \vec{u}$ ,  $\vec{OC} = \vec{v}$  và  $\vec{OB} = \vec{u} + \vec{v}$

Tích chéo còn một ứng dụng quan trọng trong việc khảo sát chiều: Giả sử ta đi từ điểm A đến điểm B theo đường thẳng và đi tiếp sang điểm C theo đường thẳng. Khi đó

- $\vec{AB} \times \vec{AC} > 0$  nếu chỗ rẽ tại B là "rẽ trái" (bẻ góc ngược chiều kim đồng hồ)
- $\vec{AB} \times \vec{AC} < 0$  nếu chỗ rẽ tại B là "rẽ phải" (bẻ góc cùng chiều kim đồng hồ)
- $\vec{AB} \times \vec{AC} = 0$  có nghĩa là ba điểm A, B, C thẳng hàng

Hàm ccw dưới đây là hàm thường dùng để kiểm tra xem  $A \rightarrow B \rightarrow C$  là rẽ trái, rẽ phải hay thẳng hàng. Hàm trả về 1 nếu rẽ trái, -1 nếu rẽ phải và 0 nếu thẳng hàng:

```
int ccw(DD A, DD B, DD C) {
    double xu=B.first-A.first, yu=B.second-A.second;
    double xv=C.first-A.first, yv=C.second-A.second;
```

```
double T=xu * yv - xv * y u;
if (T>0) return 1;
else if (T<0) return -1; else return 0;
}
```

## 2. Đường thẳng

Đường thẳng trong mặt phẳng tọa độ được xác định bởi hai điểm A, B trên mặt phẳng.

Đặt  $\vec{u} = \overrightarrow{AB} = (x_B - x_A, y_B - y_A)$ .  $\vec{u}$  được gọi là véc tơ chỉ phương của đường thẳng. Khi đó đường thẳng AB có thể coi là tập hợp các điểm  $M(x,y)$  có tọa độ thỏa mãn:

$$\begin{cases} x = x_A + (x_B - x_A) \cdot t \\ y = y_A + (y_B - y_A) \cdot t \end{cases} \quad t \in \mathbb{R} \quad (*)$$

(\*) được gọi là phương trình tham số của đường thẳng. Chú ý quan trọng:

- Nếu  $t < 0$  thì M nằm ngoài đoạn thẳng AB về phía A
- Nếu  $0 \leq t \leq 1$  thì M nằm trên đoạn thẳng AB.  $t=0$  thì  $M=A$  còn  $t=1$  thì  $M=B$
- Nếu  $t > 1$  thì M nằm ngoài đoạn thẳng AB về phía B

Ngoài các điểm nằm trên đường thẳng AB, các điểm còn lại được chia thành hai phần

- Các điểm C có  $\text{ccw}(A, B, C) > 0$  tạo thành nửa dương của mặt phẳng (nửa nằm phía tay trái của đường thẳng AB khi đi từ A đến B)
- Các điểm C có  $\text{ccw}(A, B, C) < 0$  tạo thành nửa âm của mặt phẳng (nửa nằm phía tay phải của đường thẳng AB khi đi từ A đến B)

## 3. Góc

Chú ý rằng giá trị  $\pi$  được tính trong C++ là **acos(-1)**.

Trong nhiều bài toán. Chúng ta cần phải xác định góc của một véc tơ  $\vec{u} = (x, y)$  so với chiều dương của trục hoành. Hàm **atan2(y,x)** trong C++ làm điều này. Tuy nhiên kết quả trả về là giá trị góc nằm trong đoạn  $[-\pi, \pi]$ . Do đó muốn giá trị góc trả về trong đoạn  $[0, 2\pi]$  chúng ta có hàm sau:

```
double goc(DD v) {
    double t=atan2(v.second,v.first);
    if (t<0) t=t+2*acos(-1);
    return t;
}
```

Phần lớn các trường hợp, góc thường chỉ dùng để so sánh ( $>, =, <$ ). Nếu sử dụng trực tiếp các hàm có sẵn của C++ thường sẽ gặp sai số. Một trong những cách khắc phục là xây dựng hàm tính góc riêng (chỉ dùng để so sánh). Hàm dưới đây làm điều này (nó trả về giá trị của góc nằm trong  $0..360$ ):

```
double theta(DD v) {
    if (v.first==0 && v.second==0) return 400.0;
    double t=v.second/(abs(v.first)+abs(v.second));
    if (v.first<0) t=2-t; else
        if (v.second<0) t+=4;
    return 90.0*t;
}
```

## 4. Đa giác

Đa giác là đường gấp khúc khép kín. Trong lập trình, một đa giác được mô tả bởi một dãy các đỉnh liên tiếp nhau  $A_1 A_2 \dots A_n$ .

Diện tích đại số của một đa giác có thể được tính bằng công thức sau (qui ước  $A_{n+1} \equiv A_1$ ):

$$S = \frac{1}{2} [\overrightarrow{OA_1} \times \overrightarrow{OA_2} + \overrightarrow{OA_2} \times \overrightarrow{OA_3} + \dots + \overrightarrow{OA_n} \times \overrightarrow{OA_{n+1}}]$$

$$= \frac{1}{2} [(x_1 \cdot y_2 - x_2 \cdot y_1) + (x_2 \cdot y_3 - x_3 \cdot y_2) + \dots + (x_n \cdot y_{n+1} - x_{n+1} \cdot y_n)]$$

Dãy đỉnh của đa giác có thể được lưu cùng hoặc ngược chiều kim đồng hồ (*cùng chiều kim đồng hồ có nghĩa là khi đi dọc theo biên của đa giác phần nằm phía tay phải là phần mặt phẳng bên trong đa giác*) dựa vào dấu của S như sau:

- Nếu  $S > 0$  thì các điểm liệt kê ngược chiều kim đồng hồ
- Nếu  $S < 0$  thì các điểm liệt kê cùng chiều kim đồng hồ

## B. BÀI TẬP

### 1. Khoảng cách [BEAN]

Trước cửa nhà Mr Bill có một cái hồ rất rộng. Giữa hồ có một hòn đảo nhỏ. Một lần Mr Bill nảy ra ý định bắc một cái cầu từ cửa nhà mình đến đảo giữa hồ để kinh doanh du lịch. Một vấn đề khá hóc búa đối với Mr Bill là làm thế nào xác định được khoảng cách từ nhà mình đến đảo giữa hồ?.

Có thể mô tả đảo giữa hồ như là một đa giác lồi còn nhà của Mr Bill như là một điểm nằm ngoài đa giác đó trên mặt phẳng tọa độ. Bạn hãy lập trình giúp Mr Bill tính khoảng cách nhỏ nhất từ nhà mình đến đảo.

#### Input

- Dòng đầu tiên ghi N là số đỉnh của đa giác ( $N \leq 1000$ )
- Dòng thứ hai ghi tọa độ của điểm được xem như là nhà của Mr Bill
- Tiếp theo là N dòng, mỗi dòng liệt kê tọa độ của một đỉnh của đa giác. Các đỉnh của đa giác được liệt kê ngược theo chiều kim đồng hồ.

**Output:** Một số thực duy nhất là khoảng cách từ nhà của Mr Bill đến đảo giữa hồ (giữ lại 4 chữ số phần thập phân)

#### Example:

Input	Output
3 0 0 2 0 0 2 2 2	1.4142

### 2. Bắc cầu [BRIDGE]

Con sông Flat chảy qua xứ Flatland từ nam lên bắc. Với hệ tọa độ Ox chạy từ tây sang đông, Oy – chạy từ nam lên bắc. Hai bờ của dòng sông là hai đường gấp khúc vô hạn.

Mỗi bờ là một đường gấp khúc không tự cắt và hai bờ không có điểm chung. Tả ngạn con sông là tia song song với trục Oy, chạy từ nam tới điểm  $(x_1, y_1)$ , sau đó là các đoạn thẳng  $(x_1, y_1) - (x_2, y_2)$ ,  $(x_2, y_2) - (x_3, y_3)$ , ...,  $(x_{m-1}, y_{m-1}) - (x_m, y_m)$  và tiếp theo là tia vô hạn lên phía bắc, song song với Oy, bắt đầu từ điểm  $(x_m, y_m)$ . Hữu ngạn con sông là

tia song song với OY, kết thúc ở điểm  $(u_1, v_1)$ , sau đó là các đoạn gấp khúc  $(u_1, v_1) - (u_2, v_2)$ ,  $(u_2,$



$v_2) - (u_3, v_3), \dots, (u_{n-1}, v_{n-1}) - (u_n, v_n)$  và tia vô hạn lên phía bắc song song với Oy bắt đầu từ  $(u_n, v_n)$ .

Do suy thoái kinh tế, chính phủ quyết định chỉ bắc một cây cầu qua sông ở vị trí đảm bảo độ dài cây cầu là nhỏ nhất.

*Yêu cầu:* Xác định độ dài của cây cầu cần bắc với độ chính xác  $10^{-5}$ .

*Input:*

- Dòng đầu tiên chứa số nguyên  $m$ ,
- Dòng thứ  $i$  trong  $m$  dòng tiếp theo chứa 2 số nguyên  $x_i, y_i$ ,
- Dòng tiếp theo chứa số nguyên  $n$ ,
- Dòng thứ  $j$  trong  $n$  dòng tiếp theo chứa 2 số nguyên  $u_j, v_j$ .
- $2 \leq m, n \leq 100$ , các tọa độ có giá trị tuyệt đối không vượt quá  $10^4$ .

*Output:* Độ dài cây cầu cần bắc với độ chính xác  $10^{-4}$

*Example:*

input	output
4 6 1 3 1 3 0 0 3 3 9 3 2 3 6 5	1.4142

### 3. Đoạn thẳng cắt nhau [INTERSECT]

Trên mặt phẳng tọa độ cho hai đoạn thẳng. Hỏi rằng hai đoạn thẳng có cắt nhau hay không?

*Input:*

- Dòng đầu ghi số nguyên  $T$  - số bộ dữ liệu ( $T \leq 10$ )
- $T$  dòng tiếp theo, mỗi dòng mô tả một bộ dữ liệu gồm 8 số nguyên cách nhau bởi dấu cách  $xa, ya, xb, yb, xc, yc, xd, yd$  thể hiện hai đoạn thẳng nối điểm  $A(xa, ya)$  với  $B(xb, yb)$  và nối điểm  $C(xc, yc)$  với điểm  $D(xd, yd)$

*Output:* Gồm  $T$  dòng, ghi YES/NO tùy theo bộ test tương ứng cho kết quả là cắt nhau/không cắt nhau.

*Example:*

input	output
2 0 0 2 0 1 1 1 -1 0 0 2 0 1 1 1 4	YES NO

### 4. Bao lồi [CONVEX]

Cho  $n$  điểm có tọa độ nguyên trên mặt phẳng ( $3 \leq n \leq 2 \cdot 10^5$ ). Tọa độ của mỗi điểm có trị tuyệt đối không vượt quá  $10^9$ . Có thể có những điểm trùng nhau, tuy vậy bao giờ cũng tìm được ít nhất ba điểm không thẳng hàng.

*Yêu cầu:* Xác định đa giác lồi có diện tích nhỏ nhất chứa tất cả các đỉnh đã cho (tính cả trên biên) và không có ba điểm nào thẳng hàng. Chỉ ra các đỉnh của đa giác theo trình tự duyệt ngược chiều kim đồng hồ, tính chu vi và diện tích của đa giác.

*Input:*

- Dòng đầu tiên chứa số nguyên dương  $n$
- $n$  dòng tiếp theo, mỗi dòng ghi hai số nguyên - tọa độ của một điểm.

**Output:**

- Dòng đầu tiên ghi số nguyên dương - số đỉnh của đa giác lồi
- Dòng thứ hai chứa số thực - chu vi của đa giác lồi với 2 chữ số phần thập phân
- Dòng thứ ba chứa số thực - diện tích của đa giác lồi với một chữ số phần thập phân

**Example:**

Input	Output
9	4
0 0	8.00
1 1	4.0
2 2	
1 0	
0 1	
2 0	
0 2	
2 1	
1 2	

**5. Bộ ba điểm thẳng hàng [TRIPOINT]**

Trên mặt phẳng tọa độ cho N điểm. Hỏi rằng có bao nhiêu bộ ba điểm thẳng hàng (cùng nằm trên một đường thẳng)

**Input:**

- Dòng đầu tiên ghi số nguyên dương  $N \leq 2000$
- N dòng tiếp theo, mỗi dòng ghi tọa độ của một điểm. Tọa độ các điểm có trị tuyệt đối không vượt quá 10000

**Output:** Một số nguyên duy nhất là số lượng bộ ba điểm thẳng hàng

**Example:**

input	output
6	3
0 0	
0 1	
0 2	
1 1	
1 2	
2 2	

**6. Tam giác vuông [SQTRI]**

Cho N điểm trên mặt phẳng tọa độ.

Viết chương trình xác định xem có bao nhiêu tam giác vuông với 3 đỉnh là 3 điểm trong số N điểm nói trên.

**Input:**

- Dòng đầu tiên ghi số nguyên dương N ( $N \leq 1500$ )
- N dòng sau, mỗi dòng ghi tọa độ một điểm. Tọa độ là số nguyên nằm trong khoảng  $[-10^9, 10^9]$ .

**Output:** Số nguyên duy nhất là số lượng tam giác vuông tìm được

**Example:**

input	output
5	7
-1 1	
-1 0	
0 0	
1 0	
1 1	

## 7. Thiện xạ [ARCHER]

Dân chúng vùng Nottinghamshire đề nghị Robin Hood biểu diễn bắn cung. Buổi biểu diễn được thực hiện trong không gian với hệ trục tọa độ Descartes vuông góc Oxyz, trong đó mặt phẳng Oxy (bao gồm các điểm có tọa độ  $z = 0$ ) là mặt đất. Trục Oz hướng lên trên. Người ta thả  $n$  quả bóng bay đánh số từ 1 đến  $n$ , quả bóng thứ  $i$  có tọa độ  $(x_i, y_i, z_i)$  trong đó  $z_i > 0$  (có thể có nhiều quả bóng cùng một tọa độ). Robin Hood cần phải chọn một vị trí A trên mặt đất và bắn một mũi tên đi thẳng theo một tia gốc A, mũi tên sẽ bắn xuyên qua tất cả các quả bóng nằm trên tia đó. Yêu cầu: Hãy giúp Robin Hood xác định nhiều nhất số quả bóng có thể bắn trúng.

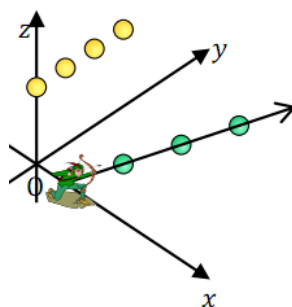
**Input:**

- Dòng 1 chứa số nguyên dương  $n \leq 2000$
- $n$  dòng tiếp theo, dòng thứ  $i$  chứa ba số nguyên  $x_i, y_i, z_i$  có trị tuyệt đối không vượt quá 2000,  $z_i > 0$

**Output:** Một số nguyên duy nhất là số lượng nhiều nhất các quả bóng có thể bắn trúng

**Example:**

input	output
7	3
2 1 1	
3 2 2	
4 3 3	
0 0 4	
0 1 4	
0 2 4	
0 3 4	



**Giải thích:** Đứng ngắm tại điểm  $A = (1,0,0)$  và ngắm quả bóng ở tọa độ  $(4,3,3)$ , khi đó bắn trúng ba quả bóng ở các tọa độ  $(2,1,1)$ ,  $(3,2,2)$ ,  $(4,3,3)$

## 8. Đa giác [POLY]

Trên mặt phẳng tọa độ, xét đa giác lồi  $n$  đỉnh, các đỉnh đều có tọa độ nguyên và có giá trị tuyệt đối không vượt quá  $10^5$ . Các đỉnh của đa giác được liệt kê theo chiều kim đồng hồ.

**Yêu cầu:** Cho đoạn thẳng xác định bởi hai điểm có tọa độ là  $(x_1, y_1)$  và  $(x_2, y_2)$  trong đó  $x_1, y_1, x_2, y_2$  là các số nguyên và có giá trị tuyệt đối không vượt quá  $10^5$ . Hãy xác định độ dài  $L$  là phần của đoạn thẳng nằm trong đa giác hay trên cạnh của đa giác và đưa ra số nguyên là phần nguyên của tích  $(L * 100)$ .

**Input:**

- Dòng đầu tiên chứa số nguyên  $n$  ( $3 \leq n \leq 100$ )
- Dòng thứ  $i$  trong  $n$  dòng sau chứa 2 số nguyên xác định tọa độ đỉnh  $i$  của đa giác,
- Dòng cuối cùng chứa 4 số nguyên  $x_1, y_1, x_2, y_2$ .

Hai số liên tiếp trên một dòng cách nhau một dấu cách.

**Output:** Ghi một số nguyên là phần nguyên của tích  $(L * 100)$ .

**Example:**

input	output
0 1	100
1 0	
0 -1	
-1 0	
-2 0 0 0	