

Skręcamy w lewo, czy w prawo?

Mając dane dwa wektory na płaszczyźnie euklidesowej: $\mathbf{a}=[a_x, a_y]$ i $\mathbf{b}=[b_x, b_y]$ możemy określić ich wzajemne położenie badając znak wyrażenia: $a_x * b_y - a_y * b_x$.

W ten sposób dowiemy się, czy:

1. \mathbf{a} i \mathbf{b} mają ten sam kierunek
2. \mathbf{b} skręca w prawo względem \mathbf{a}
3. \mathbf{b} skręca w lewo względem \mathbf{a}

1. Punkty w okręgu

Dane są: okrąg o środku $o=(x_o, y_o)$ i promieniu r oraz n punktów $p=(x_i, y_i)$. Dla każdego punktu p_i sprawdź, jego położenie względem okręgu o .

Wejście

W pierwszej linii 3 liczby całkowite z przedziału $[-10000, 10000]$ będące współrzędnymi środka okręgu i jego promieniem. Następnie n - liczba punktów i w kolejnych n liniach po dwie liczby będące współrzędnymi kolejnego punktu.

Wyjście

Dla każdego punktu w osobnej linii jedna litera: I, jeśli punkt leży w obszarze wewnętrznym okręgu O, jeśli punkt leży w obszarze zewnętrznym okręgu E, jeśli punkt leży na okręgu

Przykład

Wejście:

```
5 5 2
4
0 0
6 5
5 7
5 5
```

Output:

```
O
I
E
I
```

2. Punkty w trójkącie

Dane są wierzchołki trójkąta $p_1=(x_1, y_1)$, $p_2=(x_2, y_2)$, $p_3=(x_3, y_3)$ i punkt $p=(x, y)$. Sprawdź, czy p leży wewnątrz trójkąta p_1, p_2, p_3 .

Input

W każdej linii 8 liczb całkowitych z przedziału $[1..1000]$ będących współrzędnymi kolejnych wierzchołków oraz punktu p :

```
x1 y1 x2 y2 x3 y3 x y
.....
0 0 0 0 0 0 0 0
```

Ostatnia linia, w której znajduje się 8 zer sygnalizuje koniec danych wejściowych.

Output

Dla każdego przypadku w osobnej linii jedna litera: I, jeśli punkt leży wewnątrz trójkąta O, jeśli punkt leży na zewnątrz trójkąta E, jeśli punkt leży na brzegu trójkąta

Example

Input:

```
630 421 326 242 561 432 478 332
378 212 380 550 840 735 379 381
591 916 765 191 487 490 678 554
975 75 324 166 343 28 650 120
0 0 0 0 0 0 0 0
```

Output:

```
I
E
O
I
```

3. Sterowanie robotem

Robot porusza się na płaszczyźnie wzdłuż łamanej złożonej z odcinków. Obecnie przemieszcza się z punktu A_0 do punktu A_1 . Następnie odwiedzi punkty A_2, A_3, \dots, A_n . Oblicz, w którą stronę robot powinien skręcać oraz wartość cosinusa kąta skrętu dla każdego z punktów A_1, A_2, \dots, A_{n-1} .

Wejście

W pierwszej linii jedna liczba $2 \leq n \leq 1000$, a w kolejnych $n+1$ liniach po dwie liczby całkowite: x_i, y_i z przedziału $[-1000..1000]$ będące współrzędnymi kolejnych punktów marszruty robota.

Można założyć, że kolejne punkty są różne.

Wyjście

W kolejnych $n-1$ liniach najpierw litera L dla skrętu w lewo (R dla skrętu w prawo), odstęp, a następnie cosinus kąta skrętu z dokładnością do 6 miejsc dziesiętnych po kropce. Jeśli w danym punkcie robot powinien poruszać się prosto należy wydrukować tylko literę F, a jeśli do tyłu, to literę B.

Przykład

Wejście:

```
5
0 0
1 0
2 0
-2 0
-2 -2
0 4
```

Wyjście:

```
F
B
L 0.000000
L -0.948683
```

4. Przecięcie odcinków

Dane są cztery punkty: $p_1=(x_1, y_1)$, $p_2=(x_2, y_2)$, $p_3=(x_3, y_3)$, i $p_4=(x_4, y_4)$. Sprawdź, czy odcinek p_1p_2 przecina się z p_3p_4 .

Input

W każdej linii 8 liczb całkowitych z przedziału $[1..1000]$ będących współrzędnymi kolejnych czwórek punktów:

```
x1 y1 x2 y2 x3 y3 x4 y4
```

```
.....
0 0 0 0 0 0 0 0
```

Ostatnia linia, w której znajduje się 8 zer sygnalizuje koniec danych wejściowych.

Output

Dla każdego przypadku w osobnej linii jedna litera: T, jeśli odcinki mają choć jeden punkt wspólny, N, jeśli odcinki się nie przecinają (są rozłączne)

Example

Input:

```
630 421 326 242 561 432 478 332
378 212 380 550 840 735 379 381
591 916 765 191 487 490 678 554
975 75 324 166 343 28 650 120
0 0 0 0 0 0 0 0
```

Output:

```
N
T
T
N
```

5.Sprawdzenie wypukłości

Dane są współrzędne czterech punktów będących wierzchołkami czworokąta. Sprawdź, czy jest to prawidłowy czworokąt wypukły.

Input

W każdej linii 8 liczb całkowitych z przedziału [1..1000] będących współrzędnymi wierzchołków czworokąta:

```
x1 y1 x2 y2 x3 y3 x4 y4
.....
0 0 0 0 0 0 0 0
```

Ostatnia linia, w której znajduje się 8 zer sygnalizuje koniec danych wejściowych.

Output

Dla każdego przypadku w osobnej linii jedna litera:

- T, jeśli punkty tworzą wypukły czworokąt
- N, w przeciwnym wypadku

Example

Input:

```
100 100 100 200 200 200 200 100
630 421 326 242 561 432 478 332
378 212 380 550 840 735 379 381
591 916 765 191 487 490 678 554
975 75 324 166 343 28 650 120
0 0 0 0 0 0 0 0
```

Output:

```
T
N
N
T
N
```