

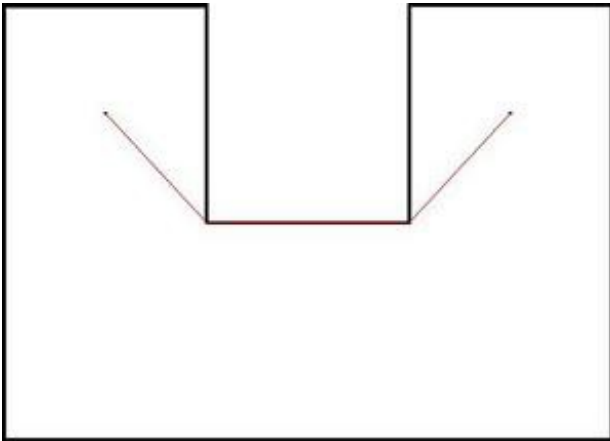
LL and ErBao

When LL and ErBao were young, they liked jumping rubber-rope (Tiao Pi Jin) very much. They jumped every day happily. But one day HH came and brought away the peaceful days. HH sometimes threw stones to them, and sometimes pushed them down suddenly. Seeing ErBao crying sadly, LL got angry finally, and decided to give HH some color see see.

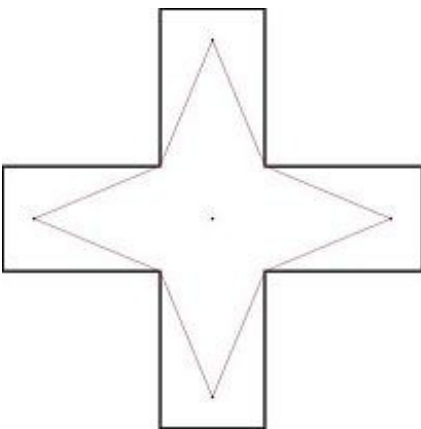
There were n trees on the ground (regarded as points), and LL planed to use his rubber-rope to wrap all the trees and form a big circle, then fooled HH into it and threw stones to him. Before finding out how to fool HH into the circle, LL wants to know how big the rubber-rope circle would be, say, the perimeter.

But LL found it difficult than expected, because their playing territory and the trees were in a bigger fence (a simple polygon with m vertices). So, the rubber-rope mustn't be out of the fence either. It's your turn to help LL find the perimeter of the circle.

sample 1



sample 2



Input

The input contains multiple cases terminated with EOF.

For each case, first line contains two integers: m, n.

($3 \leq m \leq 500$, $0 \leq n \leq 500$)

Next m lines each contain two integers: X_i, Y_i ----coordinate of the fence's i-th vertex, in Counterclockwise order.

Next n lines each contain two integers: X_i, Y_i ----coordinate of the i-th tree.

It's guaranteed that all trees were strictly in the fence, and the fence doesn't intersect or touch itself.

The absolute value of the coordinates doesn't exceed 10000.

Output

For each case output the perimeter of the rubber-band with three digits after the dot.

Example

Input:

```
8 2
0 0
30 0
30 20
20 20
20 10
10 10
10 20
0 20
5 15
25 15
```

```
12 5
5 5
5 20
-5 20
-5 5
-20 5
-20 -5
-5 -5
-5 -20
5 -20
5 -5
20 -5
20 5
0 0
0 17
0 -17
17 0
-17 0
```

Output:

```
Case #1: 48.284
Case #2: 104.000
```