Problem C. Entfernung

Input file:	`entfernung.in`
Output file:	`entfernung.out`
Time limit:	2 seconds
Memory limit:	256 mebibytes

Two famous superheroes Cheburator and Crocodile-man are being teleported to Manhattan to fight evil. The teleporter is a complex device, so it's not quite certain where exactly they will end up.

More precisely, we will represent Manhattan as a coordinate plane, and the area where they can end up in is a polygon on this plane. Inside this polygon, each superhero can end up in any point.

Even more precisely, the point where each superhero ends up is picked independently and uniformly at random. By *uniformly* we mean uniformity by area: for any figure inside the polygon, the probability to end up inside that figure is proportional to the area of that figure.

The success of the mission depends on the distance between the superheroes after they land. This being Manhattan, of course we're interested in the Manhattan distance: for two points (x_1, y_1) and (x_2, y_2), the distance is $|x_1 - x_2| + |y_1 - y_2|$.

What is the expected value of the distance between the superheroes?

Input

The first line of the input file contains one integer n, $1 \le n \le 1000$. The next n lines describe the vertices of the polygon in order (either clockwise or counterclockwise). Each vertex is described by two integers — its coordinates x and y, $-1000 \le x, y \le 1000$.

The polygon does not have self-intersections or self-touchings, but might be non-convex. The distance between any two points is still measured using the above formula without regard to the form of the polygon — in other words, the shortest "Manhattan path" might well pass partly outside the polygon.

Output

Output one floating-point number — the expected distance between the superheroes. Your output will be considered correct if it's within 10^{-8} absolute or relative error from the answer.

Examples

entfernung.in	entfernung.out
4 0 0 0 1 1 1 1 0	0.6666666666666666