Drones

Titus Cieslewski

You have recently been hired by the Marmot Security Agency (MSA). In spite of the monumental dimensions of some of their projects, the marmots are sometimes a chaotic bunch (quite like programming contest problem setters). For instance, they have recently planned the trajectory of a drone they are going to send out to exploit vulnerabilities on target WiFi routers. However, they have forgotten what the minimal range is of the wireless device on the drone!

You now get the chance to prove whether you are a loyal rodent: Given a piecewise straight trajectory, defined point by point, and the coordinates of target routers, determine what the minimal range of the wireless device on the drone has to be such that all routers can be reached from the trajectory.

The trajectory consists of up to 100 points. Starting from the first point, the drone will fly to the next point in a straight line, and so on, until it reaches the last point, where it stops. There are are up to 10'000 routers. All coordinates are between 0 and 100. Your answer should be accurate up to 0.001. Code the function `minRange()`. There are functions to retrieve routers and trajectory, so that you can implement your own data structure, except for Scala, where the trajectory and routers are passed as sequences.

A short reminder on computational geometry:

- Scalar product, $\vec{a} \cdot \vec{b} = x_a x_b + y_a y_b = |\vec{a}||\vec{b}| \cos(\alpha)$
- Two-dimensional vector product, $\vec{a} \times \vec{b} = x_a y_b - y_a x_b = |\vec{a}||\vec{b}| \sin(\alpha)$ (area spanned by the vectors, signed)

Where α is the angle counterclockwise from $\vec{a}$ to $\vec{b}$.