

5045 An Evil Plan

Three little prince Akor, Bonsur and Connor are playing in the garden. In this game they stand in three positions to form a triangle and they pass a magic ball to each other.

Their father, King Dimosor, is aware of the fact that the garden may not be safe for them, since the enemies can take this as an advantage. But he wants to see the kids smiling; so he permitted them to play.

Meanwhile, the enemy King Elohan wants to kill the princes. So, he asked his astrologer, who suggested, “There is a lucky Orchid flower in the garden; an archer can kill a prince if he can find a position where he can fire his arrow towards the prince through the Orchid, keeping the Orchid exactly in the middle between his bow and the prince.” So, King Elohan sent three archers to do the job.

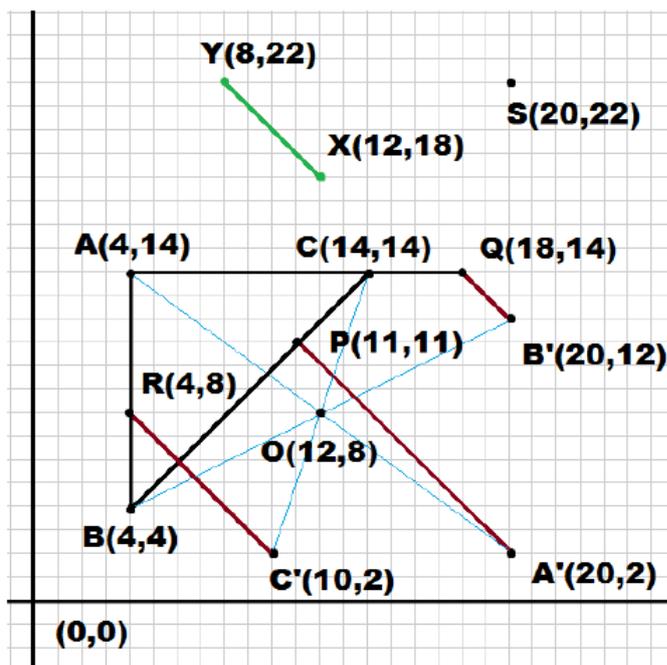
Now, King Dimosor has captured the astrologer and found the evil plan. So, he calculated the expected positions of the archers and called three guards — Pinocio, Qurota and Rubic. He planned to give three bombard canons to the guards such that they can kill the archers. But there is only one problem. The canons can be fired according to the wind direction; either in the wind direction or in opposite to the wind direction. Otherwise it would be tough for the guards to find the correct angle to shoot.

Since the guards know the wind direction, Pinocio took the position such that he forms a line with Bonsur and Connor; and he can also shoot the archer who might fire at Akor. Qurota took the position which forms a line with Akor and Connor; and he can shoot the archer who might target Bonsur. And Rubic took the position which forms a line with Akor and Bonsur and he can also shoot the archer who might fire at Connor.

After a while, the guards saw the archers and fired the canons, and the archers were killed. But the princes were quite shocked hearing the sound of the heavy firing of the canons. That’s why they want to run away from the guards as far as they can. There is a Sunflower in the garden and the princes love the flower so much that they want to run in a circular path centering the Sunflower and they will run together so that from their initial positions they all cover the same angular distance with respect to the Sunflower.

In this problem, all the positions are in a 2D plane. And for simplicity each one including the flowers is just a point in the plane. To find their farness with the guards, each prince first finds the Euclidean distance between his position and the 3 lines formed by (Pinocio, Qurota), (Pinocio, Rubic) and (Qurota, Rubic). Then from all their calculated distances, they take the minimum distance, which they call the farness. They want to maximize this farness. Since they don’t know how to find such complex thing, they seek your help.

In the picture, A , B , C , O and S denote the positions of Akor, Bonsur, Connor, the Orchid and the Sunflower respectively. A' , B' and C' denote the archer positions who aim at Akor, Bonsur and Connor respectively. P , Q and R denote the positions of Pinocio, Qurota and



Rubic respectively. X to Y denotes the wind direction.

PA' , QB' and RC' are parallel to XY (wind direction) and these lines indicate the guards aiming at the archers. Now you are given the coordinates of B , C , X , Y , P , R and S . You have to find the maximum farness as described above.

Input

The first line of input will contain an integer $T \leq 1000$ denoting the number of cases. Each case contains 7 lines, each line containing two real numbers denoting a point (x co-ordinate followed by y co-ordinate). The lines will contain the coordinates of B , C , X , Y , P , R and S respectively.

You may assume that ABC forms a valid triangle and XY is not parallel to any of AB , AC or BC . You may safely assume both co-ordinates of A , B , C , X , Y , O , P , Q and R are between -1000 to 1000 inclusive.

Output

For each case print the case number and the maximum farness rounded to 3 decimal places.

See the sample input/output for exact formatting.

Note: The test case corresponds to the picture on the right.

Sample Input

```
1
4 4
14 14
12 18
8 22
11 11
4 8
20 22
```

Sample Output

Case 1: 16.565

