Problem D

Two Walls

Prof. Pang has bought a humanoid cleaning robot to clean his yard. The robot is not very sophisticated. It can either move forward or change its direction at a time, all controlled by Prof. Pang's controller.

Prof. Pang's yard is a 2D plane. The robot needs to move from its current location A to the destination B to fulfill some "cleaning" needs of Prof. Pang. However, there are two straight walls CD and EF in Prof. Pang's yard. Since the robot is clumsy, it will fall over if it touches any of the walls (even at endpoints).

Now that Prof. Pang is lazy, he wants to minimize the number of times the robot changes its direction. Can you help him?

Input

The first line contains a single integer T $(1 \leq T \leq 10^5)$ denoting the number of test cases.

For each test case, the first line contains two integers x_A, y_A, the coordinates of point A. The second line contains two integers x_B, y_B, the coordinates of point B. The third line contains four integers x_C, y_C, x_D, y_D, the coordinates of point C and D which are the endpoints of the first wall. The fourth line contains four integers x_E, y_E, x_F, y_F, the coordinates of point E and F which are the endpoints of the second wall.

It is guaranteed that neither the current location A nor the destination B of the robot are on any of the walls. A wall may degenerate to a point. It can be proved that the robot can always move from A to B without touching any of the walls. All values lie within $[-10^9, 10^9]$.

Output

For each test case, print one number d in one line, denoting the minimum number of turns.

Sample Input	Sample Output
3 0 0 1 1 2 2 3 3 4 4 5 5 0 0 1 1 2 2 3 3 2 2 3 3 0 0 10 10 10 0 0 10 1 1 2 2	0 0 1

Note

The following are illustrations for the first sample and the third sample.



