**Q. Length of Closed Figures**

Two lines P1 and P2 are said to be connected if the end point of P1 and start point of P2 are same. In this problem, there is a collection of points in a two-dimensional space and the point numbers are given to represent a line. All the lines have distinct starting point and ending point. Given the start and end point numbers of ‘n’ lines and a check point number ‘p’ write a code to find out the length of the closed figure that can be formed that starts with the check point number ‘p’. For example, given ten pairs of point numbers for 10 lines as follows:

(2, 5)

(7, 11)

(13, 14)

(11, 15)

(17, 18)

(15, 17)

(18, 25)

(32, 40)

(25, 32)

(40, 7)

and a check point number as 7, a closed figure of length 8 can be formed with the points:

(7, 11) - (11, 15) - (15, 17) - (17, 18) - (18, 25) - (25, 32) - (32, 40) - (40, 7) - (7, 11)

Input Format

First line contains the number of lines, n

Next ‘n’ lines contain the number of the start and end points

Next line contains the check point, p

Output Format

Length of the closed figure that can be formed with check point, p

Print zero if a closed figure cannot be formed with check point, p

**Python Program**

n = int(input())  
list1 = []  
for counter in range(n):  
 temp = input().split()  
 list1.append([int(temp[0]), int(temp[1])])  
check\_point = int(input())  
temp\_check\_point = check\_point  
counting = 0  
for counter1 in range(n):  
 if counting != 0:  
 if check\_point == temp\_check\_point:  
 break  
 for counter2 in range(n):  
 if list1[counter2][0] == temp\_check\_point:  
 temp\_check\_point = list1[counter2][1]  
 counting += 1  
 if counting != 0:  
 if check\_point == temp\_check\_point:  
 break  
if temp\_check\_point != check\_point:  
 print(0)  
else:  
 print(counting)