CHALLENGE INFORMATION

You have already solved this challenge ! Though you can run the code with different logic !

Course	JAVA	Session	Methods	Question Information	Level 1 Challenge 53

Problem

Question description

Professor Lawrence makes a new robot. The robot is in the point with coordinates (x_1, y_1) and should go to the point (x_2, y_2). In a single step the robot can change any of its coordinates (maybe both of them) by one (decrease or increase). So the robot can move in one of the 8 directions. Find the minimal number of steps the robot should make to get the finish position.

Constraints:

$-10^9 \le x_1, y_1 \le 10^9$

$-10^9 \le x_2, y_2 \le 10^9$

Input Format:

The first line contains two integers x_1, y_1 - the start position of the robot.

The second line contains two integers x_2, y_2 - the finish position of the robot.

Output Format:

Print the only integer d — the minimal number of steps to get the finish position

Test Cases

Logical Test Cases

Test Case 1

INPUT (STDIN)

```
3 4
6 1
```

EXPECTED OUTPUT

```
3
```

Test Case 2

INPUT (STDIN)

```
1 1
-3 -5
```

EXPECTED OUTPUT

```
6
```

Mandatory Test Cases

Test Case 1

KEYWORD

```
public static int
robotmoves(int a,int b,int
c,int d)
```

Test Case 2

KEYWORD

```
int x1,x2,y1,y2;
```

Test Case 3

KEYWORD

```
x1= sc.nextInt();
```

Test Case 4

KEYWORD

```
robotmoves(x1,y1,x2,y2)
```

Complexity Test Cases

Test Case 1	Test Case 2	Test Case 3
CYCLOMATIC COMPLEXITY	TOKEN COUNT	NLOC
2	169	21

Code Editor

You have already solved this challenge ! Though you can run the code with different logic !

Code Editor | JAVA SE 1.8 | Light Theme

```java
import java.util.Scanner;
public class Class332241010280 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int x1,x2,y1,y2;
        x1= sc.nextInt();
        y1 = sc.nextInt();
        x2 = sc.nextInt();
        y2 = sc.nextInt();
        int minimalSteps = robotmoves(x1,y1,x2,y2);
        System.out.println(minimalSteps);
    }
    public static int robotmoves(int a,int b,int c,int d) {
    int dx = Math.abs(c - a);
    int dy = Math.abs(d - b);
    return Math.max(dx, dy);
}
}
```

Custom Input (stdin) | T1 | T2

Type Here

Output | MATCH T1 | MATCH T2

Empty

Complexity Analysis

Test Case Status