Contest Code: LTIME105D **Problem Code:** CHESSDIST

The Chessboard Distance for any two points (X_1, Y_1) and (X_2, Y_2) on a Cartesian plane is defined as $max(|X_1 - X_2|, |Y_1 - Y_2|)$.

You are given two points (X_1, Y_1) and (X_2, Y_2). Output their Chessboard Distance.

Note that, $|P|$ denotes the absolute value of integer P. For example, $|-4| = 4$ and $|7| = 7$.

Input Format

- First line will contain T, the number of test cases. Then the test cases follow.
- Each test case consists of a single line of input containing 4 space separated integers - X_1, Y_1, X_2, Y_2 - as defined in the problem statement.

Output Format

For each test case, output in a single line the chessboard distance between (X_1, Y_1) and (X_2, Y_2)

Constraints

- $1 \leq T \leq 1000$
- $1 \leq X_1, Y_1, X_2, Y_2 \leq 10^5$

Subtasks

Subtask #1 (100 points): original constraints

Sample Input 1

```
3
2 4 5 1
5 5 5 3
1 4 3 3
```

Sample Output 1

```
3
2
2
```

Explanation

- In the first case, the distance between $(2, 4)$ and $(5, 1)$ is $max(|2 - 5|, |4 - 1|) = max(|-3|, |3|) = 3$.
- In the second case, the distance between $(5, 5)$ and $(5, 3)$ is $max(|5 - 5|, |5 - 3|) = max(|0|, |2|) = 2$.
- In the third case, the distance between $(1, 4)$ and $(3, 3)$ is $max(|1 - 3|, |4 - 3|) = max(|-2|, |1|) = 2$.


```
1  /* package codechef; // don't place package name! */
2
3  import java.util.*;
4  import java.lang.*;
5  import java.io.*;
6
7  /* Name of the class has to be "Main" only if the class is public. */
```

```java
class Codechef
{
    public static void main (String[] args) throws java.lang.Exception
    {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        for(int i = 0; i < n; i++){
            int x1 = sc.nextInt();
            int x2 = sc.nextInt();
            int y1 = sc.nextInt();
            int y2 = sc.nextInt();
            int ans1 = Math.max(Math.abs(x1 - y1), Math.abs(x2 - y2));
            System.out.println(ans1);
        }
    }
}
```

0:0

Open File Run Submit

Custom Input

```
3
2 4 5 1
5 5 5 3
1 4 3 3
```