**Sherlock and Squares**

https://hr-avatars.s3.amazonaws.com/bbc82ad9-3904-4f41-8ea5-813675b88ef0/150x150.png**by**[**darkshadows**](https://www.hackerrank.com/darkshadows)

* [**Problem**](https://www.hackerrank.com/challenges/sherlock-and-squares)
* [**Submissions**](https://www.hackerrank.com/challenges/sherlock-and-squares/submissions)
* [**Leaderboard**](https://www.hackerrank.com/challenges/sherlock-and-squares/leaderboard)
* [**Discussions**](https://www.hackerrank.com/challenges/sherlock-and-squares/forum)
* [**Editorial**](https://www.hackerrank.com/challenges/sherlock-and-squares/editorial)
* [**Topics**](https://www.hackerrank.com/challenges/sherlock-and-squares/topics)

Watson gives two integers ( and ) to Sherlock and asks if he can count the number of square integers between  and  (both inclusive).

**Note**: A square integer is an integer which is the square of any integer. For example, *1*, *4*, *9*, and *16* are some of the square integers as they are squares of *1*, *2*, *3*, and *4*, respectively.

**Input Format**   
The first line contains , the number of test cases.  test cases follow, each in a new line.   
Each test case contains two space-separated integers denoting  and .

**Output Format**   
For each test case, print the required answer in a new line.

**Constraints**   
 

**Sample Input**

2

3 9

17 24

**Sample output**

2

0

**Explanation**   
*Test Case #00:* In range ,  and  are the two square numbers.   
*Test Case #01:* In range , there are no square numbers.

<https://www.hackerrank.com/challenges/sherlock-and-squares?h_r=next-challenge&h_v=zen>

<http://stackoverflow.com/questions/8934919/perfect-squares-between-two-numbers>

static void Main(string[] args)

{

int t = int.Parse(Console.ReadLine());

while (t-- > 0)

{

string[] input = Console.ReadLine().Split(' ');

int A = int.Parse(input[0]);

int B = int.Parse(input[1]);

//floor(sqrt(b)) - ceil(sqrt(a)) + 1

Console.WriteLine(Math.Floor(Math.Sqrt(B)) - Math.Ceiling(Math.Sqrt(A)) + 1);

}

Console.ReadLine();

}