Problem: Sherlock & Squares

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 .

**Constraints**

**Output Format**

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

**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.

Solution

int main()

{

int cases;

long range1, range2;

cin>>cases;

for(int i=0; i<cases; i++)

{

cin>>range1 >>range2 ;

cout<<floor(sqrt(range2))-ceil(sqrt(range1))+1<<endl;

}

return 0;

}

* Anshul Aggarwal