Sherlock and Squares ★

144 more points to get your gold badge!

Rank: 314037 | Points: 706/850

Problem Solving ★★★★

Your Sherlock and Squares submission got 20.00 points. Share Post

You are now 144 points away from the gold level for your problem solving badge.

Try the next challenge | Try a Random Challenge

Problem Submissions Leaderboard Editorial Topics

Watson likes to challenge Sherlock's math ability. He will provide a starting and ending value that describe a range of integers, inclusive of the endpoints. Sherlock must determine the number of square integers within that range.

Note: A square integer is an integer which is the square of an integer, e.g. $1, 4, 9, 16, 25$.

Example

$a = 24$

$b = 49$

There are three square integers in the range: $25, 36$ and 49. Return 3.

Function Description

Complete the squares function in the editor below. It should return an integer representing the number of square integers in the inclusive range from a to b.

squares has the following parameter(s):

- int a: the lower range boundary
- int b: the upper range boundary

Returns

- int: the number of square integers in the range

Input Format

The first line contains q, the number of test cases.

Each of the next q lines contains two space-separated integers, a and b, the starting and ending integers in the ranges.

Constraints

$1 \leq q \leq 100$

$1 \leq a \leq b \leq 10^9$

Sample Input

```
2
3 9
17 24
```

Sample Output

```
2
0
```

Explanation

Test Case #00: In range $[3, 9]$, 4 and 9 are the two square integers.

Test Case #01: In range $[17, 24]$, there are no square integers.

Change Theme Language C++14

```
#include <bits/stdc++.h>

using namespace std;

string ltrim(const string &);
string rtrim(const string &);
vector<string> split(const string &);

/*
 * Complete the 'squares' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts following parameters:
 *  1. INTEGER a
 *  2. INTEGER b
 */

int squares(int a, int b) {

    int lower_bound = ceil(sqrt(double(a)));
    int upper_bound = floor(sqrt(double(b)));
    return upper_bound - lower_bound + 1 ;

}

int main()
{
    ofstream fout(getenv("OUTPUT_PATH"));

    string q_temp;
    getline(cin, q_temp);
```

Line: 21 Col: 46

Upload Code as File □ Test against custom input Run Code Submit Code

You have earned 20.00 points!

You are now 144 points away from the gold level for your problem solving badge.

62% 706/850

Problem Solving ★★★★

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

✓ Test case 0

✓ Test case 1

✓ Test case 2

✓ Test case 3

✓ Test case 4

Compiler Message

```
Success
```

Hidden Test Case

Unlock this testcase for 5 hackos.

✓ Test case 5 🔒

✓ Test case 6 🔒

Unlock

Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy