As a noob programmer, Morti is studying loops. Today he'd like to try and write a function that calculates the sum of numbers in range fromlower to upper inclusive using a famous ProgramRights platform. However, his function hangs for no apparent reason.

Morti is pretty sure that something is wrong with the platform: it is probably either IDE or compiler issue. Your task is to prove Morti wrong by writing a function that will run without a problem in any environment.

Given numbers lower and upper, calculate the sum of numbers in the range [lower, upper].

**Example**

* For lower = 1 and upper = 1, the output should be  
  addInRange(lower, upper) = 1.
* For lower = 0 and upper = 3, the output should be  
  addInRange(lower, upper) = 6.

0 + 1 + 2 + 3 = 6.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] integer lower**

The lower bound of the range.

*Constraints*:  
-2 · 109 ≤ lower ≤ upper.

* **[input] integer upper**

The upper bound of the range.

*Constraints*:  
lower ≤ upper ≤ 2 · 109.

* **[output] integer**

The sum of integers in range [lower, upper].

<https://codefights.com/challenge/JZs5SSyPfQg55QMYZ>

static int addInRange(int lower, int upper)

{

int sum = 0;

for (int i = lower; i <= upper; i++)

{

sum += i;

}

return sum;

}