Codility_

Candidate Report: trainingAGT2EY-KMH

Check out Codility training tasks

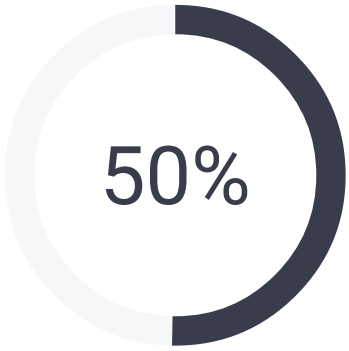
Test Name:

Summary Timeline

Tasks summary

Task	Time spent	Score
CountDiv C	23 min	50%

Total score



Tasks Details

Medium

1. CountDiv

Compute number of integers divisible by k in range [a..b].

Task Score	Correctness	Performance
50%	25%	75%

Task description

Write a function:

```
int solution(int A, int B, int K);
```

that, given three integers A, B and K, returns the number of integers within the range [A..B] that are divisible by K, i.e.:

{ i : A ≤ i ≤ B, i **mod** K = 0 }

For example, for A = 6, B = 11 and K = 2, your function should return 3, because there are three numbers divisible by 2 within the range [6..11], namely 6, 8 and 10.

Write an **efficient** algorithm for the following assumptions:

- A and B are integers within the range [0..2,000,000,000];
- K is an integer within the range [1..2,000,000,000];
- A ≤ B.

Solution

Programming language used:	C
Total time used:	23 minutes
Effective time used:	23 minutes
Notes:	*not defined yet*

Task timeline

06:38:41 07:01:16

Code: 07:01:16 UTC, c, final, score: **50**

show code in pop-up

```
1  // you can write to stdout for debugging purposes, e.g.
2  // printf("this is a debug message\n");
3
4  int solution(int A, int B, int K) {
5      // write your code in C99 (gcc 6.2.0)
6      int ret = (B-A)/K;
7      int tmp = ((B-A)%K-B%K+A%K)>=0?1:0;
8      //printf("%d,%d",(int)((B-A)/K),((B-A)%K-B%K+A%K)>=0?1
```

```
9       return ret + tmp;
10  }
```

Analysis summary

The following issues have been detected: wrong answers.

For example, for the input `[1, 1, 11]` the solution returned a wrong answer (got 1 expected 0).

Analysis

expand all — Example tests

▶ example A = 6, B = 11, K = 2	✓ OK	

collapse all — Correctness tests

▼ simple A = 11, B = 345, K = 17	✓ OK	
1. 0.001 s OK		
▼ minimal A = B in {0,1}, K = 11	✗ WRONG ANSWER	got 1 expected 0
1. 0.001 s OK		
2. 0.001 s WRONG ANSWER, got 1 expected 0		
▼ extreme_ifempty A = 10, B = 10, K in {5,7,20}	✗ WRONG ANSWER	got 1 expected 0
1. 0.001 s WRONG ANSWER, got 1 expected 0		
2. 0.001 s OK		
3. 0.001 s WRONG ANSWER, got 1 expected 0		
▼ extreme_endpoints verify handling of range endpoints, multiple runs	✗ WRONG ANSWER	got 2 expected 1
1. 0.001 s OK		
2. 0.001 s OK		
3. 0.001 s OK		
4. 0.001 s WRONG ANSWER, got 2 expected 1		
5. 0.001 s OK		
6. 0.001 s OK		

collapse all — Performance tests

▼ big_values A = 100, B=123M+, K=2	✓ OK	
1. 0.001 s OK		
▼ big_values2 A = 101, B = 123M+, K = 10K	✗ WRONG ANSWER	got 12346 expected 12345
1. 0.001 s WRONG ANSWER, got 12346 expected 12345		
▼ big_values3 A = 0, B = MAXINT, K in {1,MAXINT}	✓ OK	
1. 0.001 s OK		
2. 0.001 s OK		
▼ big_values4 A, B, K in {1,MAXINT}	✓ OK	

1. 0.001 s **OK**
2. 0.001 s **OK**
3. 0.001 s **OK**
4. 0.001 s **OK**
