**Closest to Zero**

12290% *of* 698 *of* 237[coffman\_21](https://www.codewars.com/users/coffman_21)

C#

* [Train Again](https://www.codewars.com/kata/closest-to-zero/train/csharp)
* [Next Kata](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/closest-to-zero/solutions/csharp)

[Forks (2)](https://www.codewars.com/kata/closest-to-zero/forks/csharp)

[Discourse (15)](https://www.codewars.com/kata/closest-to-zero/discuss/csharp)

* Add to Collection
* |
* Share this kata:

Simply find the closest value to zero from the list. Notice that there are negatives in the list.

List is always not empty and contains only integers. Return None if it is not possible to define only one of such values. And of course, we are expecting 0 as closest value to zero.

Examples:

[2, 4, -1, -3] => -1

[5, 2, -2] => None

[5, 2, 2] => 2

[13, 0, -6] => 0

Fundamentals

Lists

Data Structures

Numbers

<https://www.codewars.com/kata/closest-to-zero/csharp>

public static int? Closest(int[] arr)

{

// return null if it is not possible to return 1 unique closest value

int min\_dif = int.MaxValue;

int num = -1;

for (int i = 0; i < arr.Length; i++)

{

if (Math.Abs(arr[i]) < min\_dif)

{

min\_dif = Math.Abs(arr[i]);

num = arr[i];

}

}

if (arr.Contains(-num) && num != 0)

{

return null;

}

return num;

}