Given a list of numbers, your task is to find all its elements that are not smaller than 10. The result you should return can be found from these values as follows:

* the tens of the ith (0-based) found value should be added to the result if i is even;
* the tens of the ith (0-based) found value should be subtracted from the result if i is odd.

**Example**

For numbers = [9, 10, 6, 3, 24, 0, 7], the output should be  
TenPower(numbers) = -1.

There are two numbers than are not smaller than 10: 10and 24. The result can thus be obtained as 1 - 2 = -1.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] array.integer numbers**

*Constraints:*  
0 ≤ numbers.length ≤ 1500,  
0 ≤ numbers[i] < 231.

* **[output] integer**

<https://codefights.com/challenge/xm9eDrzYWvz7EL4Pp?utm_source=featuredChallenge&utm_medium=email&utm_campaign=email_notification>

static int TenPower(int[] numbers)

{

int sum = 0;

int index = 0;

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

{

if (numbers[i] >= 10)

{

if (index % 2 == 0)

{

sum += numbers[i] / 10 % 10;

}

else

{

sum -= numbers[i] / 10 % 10;

}

index++;

}

}

return sum;

}