Define *crossover operation* over two equal-length strings A and B as follows:

* the result of that operation is a string of the same length as the input strings
* result[i] is chosen at random between A[i]and B[i]

Given array of strings inputArray and a string result, find for how many pairs of strings from inputArray the result of the crossover operation over them may be equal to result.

Note that (A, B) and (B, A) are the same pair. Also note that the pair cannot include the same element of the array twice (however, if there are two equal elements in the array, they can form a pair).

**Example**

For inputArray = ["abc", "aaa", "aba", "bab"] and result = "bbb", the output should be  
stringsCrossover(inputArray, result) = 2.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] array.string inputArray**

A non-empty array of equal-length strings.

*Constraints:*  
2 ≤ inputArray.length ≤ 10,  
1 ≤ inputArray[i].length ≤ 10.

* **[input] string result**

A string of the same length as each of the inputArray elements.

*Constraints:*  
result.length = inputArray[i].length.

* **[output] integer**

<https://codefights.com/arcade/code-arcade/spring-of-integration/6QDtjNr3WoL5W4Kvo>

int stringsCrossover(string[] inputArray, string result)

{

int count = 0;

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

{

for (int j = (i + 1); j < inputArray.Length; j++)

{

string chars1 = inputArray[i];

string chars2 = inputArray[j];

bool isCross = true;

for (int k = 0; k < result.Length; k++)

{

if (result[k] != chars1[k] && result[k] != chars2[k])

{

isCross = false;

break;

}

}

if (isCross)

{

count += 1;

}

}

}

return count;

}