


# 771. Jewels and Stones

Solved 

Easy

Topics

Companies

Hint

You're given strings `jewels` representing the types of stones that are jewels, and `stones` representing the stones you have. Each character in `stones` is a type of stone you have. You want to know how many of the stones you have are also jewels.

Letters are case sensitive, so `"a"` is considered a different type of stone from `"A"`.

## Example 1:

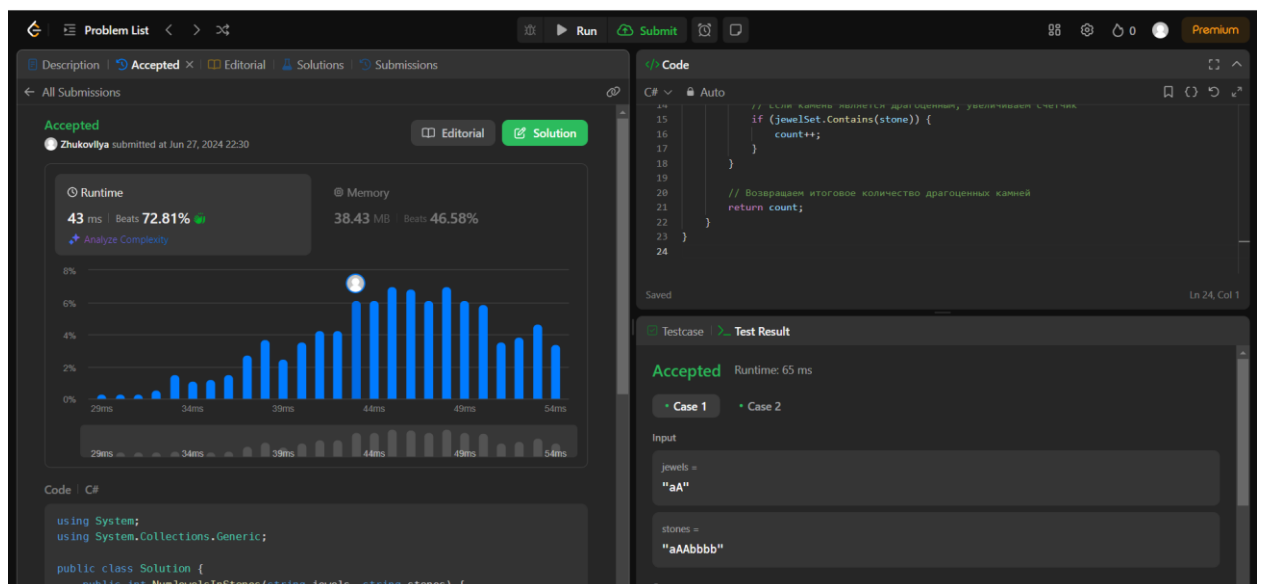
**Input:** `jewels = "aA", stones = "aAAbbbb"`  
**Output:** 3

## Example 2:

**Input:** `jewels = "z", stones = "ZZ"`  
**Output:** 0

## Constraints:

- `1 <= jewels.length, stones.length <= 50`
- `jewels` and `stones` consist of only English letters.



The screenshot shows a code editor interface with two main panes. The left pane displays submission statistics for a C# solution, including runtime (43 ms) and memory (38.43 MB). The right pane shows the C# code and test results for two cases.

**Submission Statistics:**

- Runtime: 43 ms | Beats 72.81%
- Memory: 38.43 MB | Beats 46.58%

**Code:**

```
using System;
using System.Collections.Generic;

public class Solution {
    public int NumJewelsInStones(string jewels, string stones) {
```

**Test Results:**

- Case 1: Accepted, Runtime: 65 ms
- Case 2: Accepted, Runtime: 65 ms

**Input:**

```
jewels = "aA"
stones = "aAAbbbb"
```

**Output:**

Код:

```
using System;
using System.Collections.Generic;

public class Solution
{
    public int NumJewelsInStones(string jewels, string stones)
    {
```

```
// Создаем HashSet для типов драгоценных камней
HashSet<char> jewelSet = new HashSet<char>(jewels);

// Инициализируем счетчик для подсчета количества драгоценных камней
int count = 0;

// Проходим по каждому камню в строке stones
foreach (char stone in stones)
{
    // Если камень является драгоценным, увеличиваем счетчик
    if (jewelSet.Contains(stone))
    {
        count++;
    }
}

// Возвращаем итоговое количество драгоценных камней
return count;
}
}
```