

# Segundo Problema

## 771. Joias e Pedras

Você é dado uma string chamada de `jewels` que representa os tipos de pedras que são joias, e uma string chamada `stones` que representam as pedras que você possui.

Cada letra em `stones` é um tipo de pedra que você tem.

Crie um algoritmo que diga quantas das pedras que você tem são joias.

As letras diferem entre maiúsculas e minúsculas, então `a` e `A` são dois tipos de joias diferentes.

Restrições:

```
`jewels` e `stones` contêm no máximo 50 caracteres.  
`jewels` e `stones` consistem apenas de letras do alfabeto em inglês.  
Todos os caracteres em `jewels` são únicos.
```

Exemplo 1:

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

Exemplo 2:

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

## Algoritmo

Receber **as** joias e armazenar em uma variavel chamada ``jewels`` que será **do** tipo vetor de caracteres.  
Receber **as** pedras e armazenar na variável chamada de ``stones`` que será **do** tipo vetor de caracteres.  
Inicializar uma variavel chamada ``total`` e será **do** tipo inteiro.  
``total`` recebe o valor de `0`.  
Percorrer o vetor de caracteres ``stones``, para cada caractere encontrado no vetor de caracteres ``jewels``, incrementar a variavel ``total``.  
Retornar o valor de ``total``.

## Pseudo- Código

```
INICIO
FUNCAO includes(vetor: vetor[50] de caractere, element: caractere): logico
INICIO
    // Omitida para fins didáticos
FIM FUNCAO

VAR jewels, stones: vetor[50] de caractere
VAR total, counter: inteiro

leia(jewels)
leia(stones)

total <- 0

PARA counter DE 0 ATE 50 FAÇA
    SE includes(jewels, stones[counter]) ENTAO
        total <- total + 1
    FIM SE
FIM PARA

RETORNA total
FIM ALGORITMO
```

## Solução

```
/**
 * @param {string} jewels
 * @param {string} stones
 * @return {number}
 */
var numJewelsInStones = function(jewels, stones) {
    let total = 0;

    for(let i = 0; i < stones.length; i++) {
        const stone = stones[i];

        if (jewels.includes(stone)) {
            total += 1
        }
    }

    return total;
};
```