# Problem 3664: Two-Letter Card Game

## Problem Information

**Difficulty:** Medium
**Acceptance Rate:** 11.53%
**Paid Only:** No
**Tags:** Array, Hash Table, String, Counting, Enumeration

## Problem Description

You are given a deck of cards represented by a string array `cards`, and each card displays two lowercase letters.

You are also given a letter `x`. You play a game with the following rules:

* Start with 0 points. * On each turn, you must find two **compatible** cards from the deck that both contain the letter `x` in any position. * Remove the pair of cards and earn **1 point**. * The game ends when you can no longer find a pair of compatible cards.

Return the **maximum** number of points you can gain with optimal play.

Two cards are **compatible** if the strings differ in **exactly** 1 position.

**Example 1:**

**Input:** cards = ["aa","ab","ba","ac"], x = "a"

**Output:** 2

**Explanation:**

* On the first turn, select and remove cards `"ab"` and `"ac"`, which are compatible because they differ at only index 1. * On the second turn, select and remove cards `"aa"` and `"ba"`, which are compatible because they differ at only index 0.

Because there are no more compatible pairs, the total score is 2.

**Example 2:**

**Input:** cards = ["aa","ab","ba"], x = "a"

**Output:** 1

**Explanation:**

* On the first turn, select and remove cards `"aa"` and `"ba"`.

Because there are no more compatible pairs, the total score is 1.

**Example 3:**

**Input:** cards = ["aa","ab","ba","ac"], x = "b"

**Output:** 0

**Explanation:**

The only cards that contain the character `'b'` are `"ab"` and `"ba"`. However, they differ in both indices, so they are not compatible. Thus, the output is 0.

**Constraints:**

* `2 <= cards.length <= 105` * `cards[i].length == 2` * Each `cards[i]` is composed of only lowercase English letters between `'a'` and `'j'`. * `x` is a lowercase English letter between `'a'` and `'j'`.

## Code Snippets

**C++:**

```
class Solution {
public:
int score(vector<string>& cards, char x) {

}
```

```
};
```

**Java:**

```
class Solution {
public int score(String[] cards, char x) {

}
}
```

**Python3:**

```
class Solution:
def score(self, cards: List[str], x: str) -> int:
```