Problem 2606: Find the Substring With Maximum Cost

Problem Information

Difficulty: Medium
Acceptance Rate: 57.51%
Paid Only: No
Tags: Array, Hash Table, String, Dynamic Programming

Problem Description

You are given a string `s`, a string `chars` of **distinct** characters and an integer array `vals` of the same length as `chars`.

The **cost of the substring** is the sum of the values of each character in the substring. The cost of an empty string is considered `0`.

The **value of the character** is defined in the following way:

* If the character is not in the string `chars`, then its value is its corresponding position **(1-indexed)** in the alphabet. * For example, the value of `'a'` is `1`, the value of `'b'` is `2`, and so on. The value of `'z'` is `26`. * Otherwise, assuming `i` is the index where the character occurs in the string `chars`, then its value is `vals[i]`.

Return _the maximum cost among all substrings of the string_ `s`.

Example 1:

Input: s = "adaa", chars = "d", vals = [-1000] **Output:** 2 **Explanation:** The value of the characters "a" and "d" is 1 and -1000 respectively. The substring with the maximum cost is "aa" and its cost is 1 + 1 = 2. It can be proven that 2 is the maximum cost.

Example 2:

Input: s = "abc", chars = "abc", vals = [-1,-1,-1] **Output:** 0 **Explanation:** The value of the characters "a", "b" and "c" is -1, -1, and -1 respectively. The substring with the maximum

cost is the empty substring "" and its cost is 0. It can be proven that 0 is the maximum cost.

Constraints:

* `1 <= s.length <= 105` * `s` consist of lowercase English letters. * `1 <= chars.length <= 26` * `chars` consist of **distinct** lowercase English letters. * `vals.length == chars.length` * `-1000 <= vals[i] <= 1000`

Code Snippets

C++:

```
class Solution {
public:
int maximumCostSubstring(string s, string chars, vector<int>& vals) {

}
};
```

Java:

```
class Solution {
public int maximumCostSubstring(String s, String chars, int[] vals) {

}
}
```

Python3:

```
class Solution:
def maximumCostSubstring(self, s: str, chars: str, vals: List[int]) -> int:
```