PROGRAM23:GYou are given a string s. s[i] is either a lowercase English letter or '?'. For a string t having length m containing only lowercase English letters, we define the function cost(i) for an index i as the number of characters equal to t[i] that appeared before it, i.e. in the range [0, i - 1]. The value of t is the sum of cost(i) for all indices i. For example, for the string t = "aab":
cost(0) = 0
cost(1) = 1
cost(2) = 0
Hence, the value of "aab" is 0 + 1 + 0 = 1. Your task is to replace all occurrences of '?' in s with any lowercase English letter so at the value of s is minimized.

Progarm:

```
def minimize_cost(s):
    from collections import Counter

    # Step 1: Initialize a list to build the resulting string
    result = list(s)

    # Step 2: Frequency counter to keep track of character counts
    freq_counter = Counter()

    for i, char in enumerate(s):
        if char == '?':
            # Find the character with the minimum frequency count to replace '?'
            min_char = None
            min_count = float('inf')

            for candidate in 'abcdefghijklmnopqrstuvwxyz':
                if freq_counter[candidate] < min_count:
                    min_char = candidate
                    min_count = freq_counter[candidate]

            # Replace '?' with the chosen character
            result[i] = min_char

            # Update the frequency counter
            freq_counter[min_char] += 1
        else:
            # Update the frequency counter for existing characters
            freq_counter[char] += 1

    return ''.join(result)

# Example usage:
s = "a?b?c?"
print(minimize_cost(s))  # Output should be a string with minimized value
```

OUTPUT:

```
"C:\Program Files\Python312\python.exe" "C:\Work Space\DAA COADS.PYTHON\progarm 23.py"
abbccd

Process finished with exit code 0
```

TIME COMPLEXITY:
O(n)