Description | Accepted | Editorial | Solutions | Submissions

All Submissions

Accepted 262 / 262 testcases passed

KafiaKhan submitted at Jan 08, 2025 10:48

Editorial | Solution

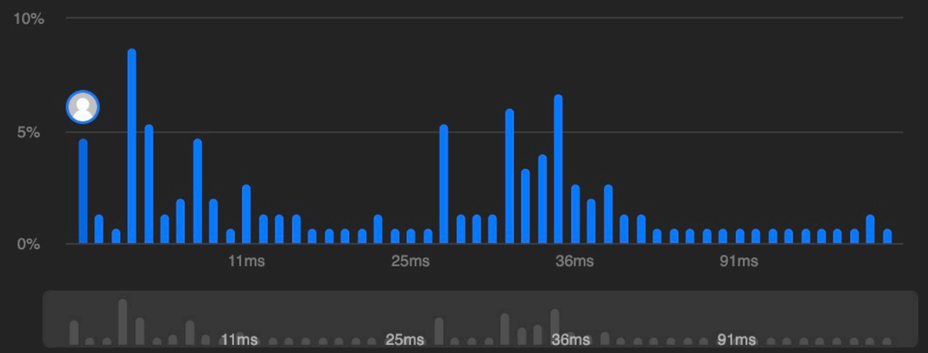
Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

10.48 MB | Beats 85.24%



Code | C++

```cpp
//Kafia Khan VU21CSEN00300362
class Solution {
public:
    bool canReplace(string& target, int pos, string& stamp) {
        // Check if we can replace the window at pos with stamp
        for(int i = 0; i < stamp.length(); i++) {
            if(target[i + pos] != '?' && target[i + pos] != stamp[i]) {
                return false;
```

View more

More challenges

- 605. Can Place Flowers
- 2901. Longest Unequal Adjacent Groups Subsequence II

Code

C++ | Auto

```cpp
//Kafia Khan VU21CSEN00300362
class Solution {
public:
    bool canReplace(string& target, int pos, string& stamp) {
        // Check if we can replace the window at pos with stamp
        for(int i = 0; i < stamp.length(); i++) {
            if(target[i + pos] != '?' && target[i + pos] != stamp[i]) {
                return false;
            }
        }
        return true;
    }

    int replace(string& target, int pos, int len, int count) {
        // Replace the window with '?' and return number of new '?' added
        int newCount = 0;
        for(int i = 0; i < len; i++) {
            if(target[pos + i] != '?') {
                target[pos + i] = '?';
                newCount++;
            }
        }
```

Saved

Ln 77, Col 3

Testcase | Test Result

Case 1 | Case 2 | +

stamp =

"abc"

target =

"ababc"

Source

Problem List | Run | Submit | Premium

Description | Accepted | Editorial | Solutions | Submissions

All Submissions

Accepted 112 / 112 testcases passed

KafiaKhan submitted at Jan 08, 2025 10:47

Editorial | Solution

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

43.66 MB | Beats 41.78%

Code | C++

```
class Solution {
public:
    int kadane(vector<int>& nums) {
        int currSum = nums[0];
        int maxSum = nums[0];
        for(int i = 1; i < nums.size(); i++) {
            currSum = max(nums[i], currSum + nums[i]);
            maxSum = max(maxSum, currSum);
```

View more

More challenges

- 1578. Minimum Time to Make Rope Colorful
- 1091. Shortest Path in Binary Matrix

Code

C++ | Auto

```
1  //Kafia Khan VU21CSEN0300362
2  class Solution {
3  public:
4      int kadane(vector<int>& nums) {
5          int currSum = nums[0];
6          int maxSum = nums[0];
7          for(int i = 1; i < nums.size(); i++) {
8              currSum = max(nums[i], currSum + nums[i]);
9              maxSum = max(maxSum, currSum);
10         }
11         return maxSum;
12     }
13
14     int maxSubarraySumCircular(vector<int>& nums) {
15         // If all numbers are negative, return maximum element
16         bool allNegative = true;
17         for(int num : nums) {
18             if(num >= 0) {
19                 allNegative = false;
20                 break;
21             }
22         }
```

Saved | Ln 8, Col 10

Testcase | Test Result

Case 1 | Case 2 | Case 3

nums =

```
[1,-2,3,-2]
```

Source