Minimum Window Substring

Try to solve the Minimum Window Substring problem.

We'll cover the following

Statement

We are given two strings, `s` and `t`, find the minimum window substring of `t` in `s`.

The minimum window substring of `t` in `s` is defined as follows:

1. It is the shortest substring of `s` that includes all of the characters present in `t`.
2. The frequency of each character in this substring that belongs to `t` should be equal to or greater than its frequency in `t`.
3. The order of the characters does not matter here.

Constraints:

- Strings `s` and `t` consist of uppercase and lowercase English characters.
- $1 \leq$ `s.length`, `t.length` $\leq 10^3$

Examples

Sample example 1

Input

s	A	B	A	A	C	B	B	A

t	A	B	C

Output

A	C	B

1 of 3

Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check that you're solving the correct problem.

Minimum Window Substring

1 What is the output if the following strings are given as input?

`s` = “cabwefgewcwaefgcf”

`t` = “cae”

A) “cwae”

B) “cabwe”

C) “aefgc”

D) “”

Submit Answer

Question 1 of 3
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Set up a sliding, adjustable window to move across the string `s`.

Initialize two collections (of your own choice): one to store the frequency of characters in `t` and the other to track the frequency of characters in the current window.

Iterate over `s`, expanding the current window until the frequencies of characters of `t`

in the window are at least equal to their respective frequencies in `t`.

Trim the window by removing all the unnecessary characters. If the current window size is less than the length of the minimum window substring found so far, update the minimum window substring.

Continue iterating over `s` and perform the previous two steps in each iteration.

Return the minimum window substring.

Reset | Show Solution | Submit

Try it yourself

Implement your solution in the following coding playground:

```java
import java.util.*;

public class Main{
    public static String minWindow(String s, String t) {
        // your code will replace this placeholder return statement below
        return "";
    }
}
```

Powered by AI | Submit

Test Cases | Results

Case 1 | Case 2 | Case 3

Input #1

"ABCD"

Input #2

"ABC"

Minimum Window Substring

Back

Solution: Minimum Wi...

Next

Solution: Minimum Wi...

Mark as Completed