Our Solution(s) Run Code

Solution 1

```
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

using System;
using System.Collections.Generic;

public class Program {
  // O(b + s) time | O(b + s) space - where b is the length of the big
  // input string and s is the length of the small input string
  public static string SmallestSubstringContaining(string bigstring, s
    Dictionary<char, int> targetCharCounts = getCharCounts(smallstring
    List<int> substringBounds = getSubstringBounds(bigstring, targetCh
    return getstringFromBounds(bigstring, substringBounds);
  }

  public static Dictionary<char, int> getCharCounts(string str) {
    Dictionary<char, int> charCounts = new Dictionary<char, int>();
    for (int i = 0; i < str.Length; i++ ) {
      increaseCharCount(str[i], charCounts);
    }
    return charCounts;
  }

  public static List<int> getSubstringBounds(string str, Dictionary<ch
    int> targetCharCounts) {
    List<int> substringBounds = new List<int>(){
      0, Int32.MaxValue
    };
    Dictionary<char, int> substringCharCounts = new Dictionary<char, i
    int numUniqueChars = targetCharCounts.Count;
    int numUniqueCharsDone = 0;
    int leftIdx = 0;
    int rightIdx = 0;
    // Move the rightIdx to the right in the string until you've count
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```

public class Program {
  public static string SmallestSubstringContaining(string bigstring, s
    // Write your code here.
    return null;
  }
}

```

Our Tests

Custom Output Submit Code

Run or submit code when you're ready.