Our Solution(s)   Run Code

Solution 1

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

using System.Collections.Generic;

public class Program {
  // O(n) time | O(n) space
  public static bool BalancedBrackets(string str) {
    string openingBrackets = "([{";
    string closingBrackets = ")]}";
    Dictionary<char, char> matchingBrackets = new Dictionary<cha
    matchingBrackets.Add(')', '(');
    matchingBrackets.Add(']', '[');
    matchingBrackets.Add('}', '{');
    List<char> stack = new List<char>();
    for (int i = 0; i < str.Length; i++) {
      char letter = str[i];
      if (openingBrackets.IndexOf(letter) != -1) {
        stack.Add(letter);
      } else if (closingBrackets.IndexOf(letter) != -1) {
        if (stack.Count == 0) {
          return false;
        }
        if (stack[stack.Count - 1] == matchingBrackets[letter])
          stack.RemoveAt(stack.Count - 1);
        } else {
          return false;
        }
      }
    }
    return stack.Count == 0;
  }
}
```

Your Solutions   Run Code

Solution 1   Solution 2   Solution 3

```csharp
public class Program {
  public static bool BalancedBrackets(string str) {
    // Write your code here.
    return false;
  }
}
```

Our Tests   Custom Output   Submit Code

Run or submit code when you're ready.