Description | Solution | Discuss (661) | Submissions

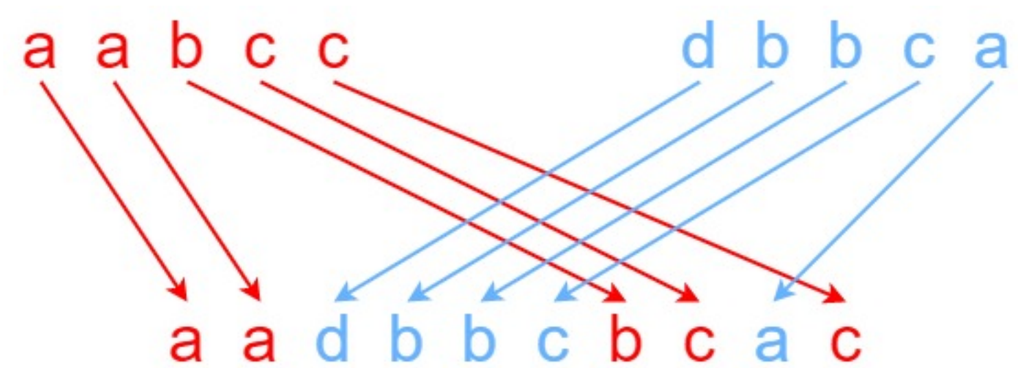
Java | Autocomplete

97. Interleaving String

Hard 1563 91 Add to List Share

Given `s1`, `s2`, and `s3`, find whether `s3` is formed by the interleaving of `s1` and `s2`.

Example 1:




```
Input: s1 = "aabcc", s2 = "dbbca", s3 = "aadbbcbcac"
Output: true
```

Example 2:

```
Input: s1 = "aabcc", s2 = "dbbca", s3 = "aadbbbaccc"
Output: false
```

Example 3:

```
Input: s1 = "", s2 = "", s3 = ""
Output: true
```

```java
class Solution {
    public boolean isInterleave(String s1, String s2, String s3) {
      // no need to initialize this 2D array because its declare its automatically
        // initialized by zero. So the bellow code was for initialization.
        boolean memo[][] = new boolean[s1.length() + 1][s2.length() + 1];
        // if the length doesn't match
        if (s3.length() != s1.length() + s2.length()) return false;

        // Initializing the matrix memo
        memo[0][0] = true;
        for (int i = 1; i <= s1.length(); i++) {
            if (s3.charAt(i - 1) == s1.charAt(i - 1))
                memo[i][0] = true;
            else
                break;
        }
        for (int j = 1; j <= s2.length(); j++) {
            if (s3.charAt(j - 1) == s2.charAt(j - 1))
                memo[0][j] = true;
            else
                break;
        }

        // Iteratively checking when a character matches and filling up the table
        for (int i = 1; i <= s1.length(); i++) {
            for (int j = 1; j <= s2.length(); j++) {
                if (s3.charAt(i + j - 1) == s1.charAt(i - 1)) {
                    if (memo[i - 1][j] == true)
                        memo[i][j] = true;
                }
                if (s3.charAt(i + j - 1) == s2.charAt(j - 1)) {
                    if (memo[i][j - 1] == true)
                        memo[i][j] = true;
                }
            }
```

Your previous code was restored from your local storage. Reset to default