GeeksforGeeks
A computer science portal for geeks

GeeksQuiz

Login

Home Algorithms DS GATE Interview Corner Q&A C C++ Java Books Contribute Ask a Q About

Array Bit Magic C/C++ Articles GFacts Linked List MCQ Misc Output String Tree Graph

Check whether a given string is an interleaving of two other given strings

Given three strings A, B and C. Write a function that checks whether C is an interleaving of A and B.
C is said to be interleaving A and B, if it contains all characters of A and B and order of all characters in individual strings is preserved. See previous post for examples.

Solution:
Pick each character of C one by one and match it with the first character in A. If it doesn’t match then match it with first character of B. If it doesn’t even match first character of B, then return false. If the character matches with first character of A, then repeat the above process from second character of C, second character of A and first character of B. If first character of C matches with the first character of B (and doesn’t match the first character of A), then repeat the above process from the second character of C, first character of A and second character of B. If all characters of C match either with a character of A or a character of B and length of C is sum of lengths of A and B, then C is an interleaving A and B.

Thanks to venkat for suggesting following C implementation.

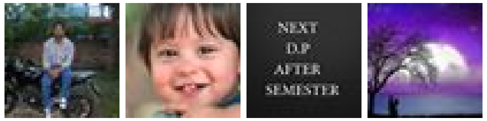
```
#include<stdio.h>

// Returns true if C is an interleaving of A and B, otherwise
// returns false
bool isInterleaved (char *A, char *B, char *C)
{
    // Iterate through all characters of C.
    while (*C != 0)
    {
        // Match first character of C with first character of A,
```

Google™ Custom Search

GeeksforGeeks

52,731 people like GeeksforGeeks.



Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion

```
        // If matches them move A to next
        if (*A == *C)
            A++;

        // Else Match first character of C with first character of B,
        // If matches them move B to next
        else if (*B == *C)
            B++;

        // If doesn't match with either A or B, then return false
        else
            return false;

        // Move C to next for next iteration
        C++;
    }

    // If A or B still have some characters, then length of C is small
    // than sum of lengths of A and B, so return false
    if (*A || *B)
        return false;

    return true;
}

// Driver program to test above functions
int main()
{
    char *A = "AB";
    char *B = "CD";
    char *C = "ACBG";
    if (isInterleaved(A, B, C) == true)
        printf("%s is interleaved of %s and %s", C, A, B);
    else
        printf("%s is not interleaved of %s and %s", C, A, B);

    return 0;
}
```

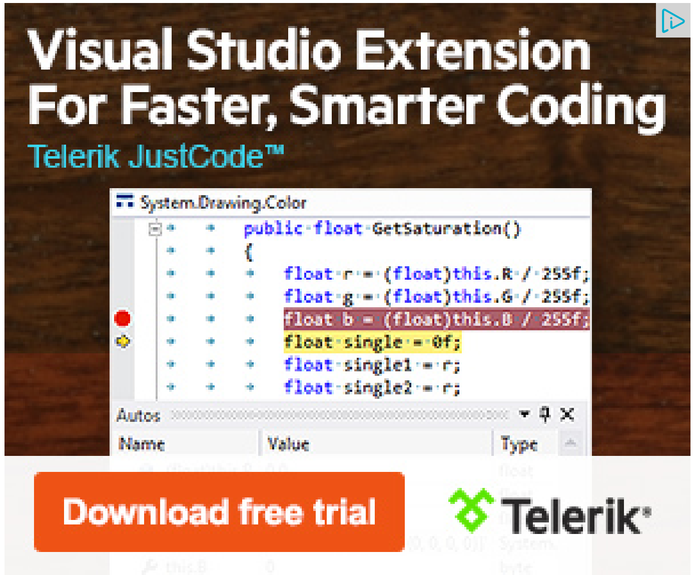
Output:

```
ACBG is not interleaved of AB and CD
```

Time Complexity: O(m+n) where m and n are the lengths of strings A and B respectively.

Note that the above approach doesn't work if A and B have some characters in common. For

Geometric Algorithms



Popular Posts

All permutations of a given string
Memory Layout of C Programs
Understanding “extern” keyword in C
Median of two sorted arrays
Tree traversal without recursion and without stack!
Structure Member Alignment, Padding and Data Packing
Intersection point of two Linked Lists
Lowest Common Ancestor in a BST.
Check if a binary tree is BST or not
Sorted Linked List to Balanced BST

example, if string A = “AAB”, string B = “AAC” and string C = “AACAAB”, then the above method will return false. We have discussed here an extended solution that handles common characters.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Related Tpoics:

- Printing Longest Common Subsequence
- Suffix Array | Set 2 (nLogn Algorithm)
- Rearrange a string so that all same characters become d distance away
- Recursively remove all adjacent duplicates
- Find the first non-repeating character from a stream of characters
- Dynamic Programming | Set 33 (Find if a string is interleaved of two other strings)
- Remove “b” and “ac” from a given string
- Dynamic Programming | Set 29 (Longest Common Substring)

5 Tweet 0 2

Writing code in comment? Please use **ideone.com** and share the link here.

695 Subscribe

Recent Comments

affiszerv Your example has two 4s on row 3, that's why it...

Backtracking | Set 7 (Sudoku) · 16 minutes ago

RVM Can someone please elaborate this Qs from above...

Flipkart Interview | Set 6 · 36 minutes ago

Vishal Gupta I talked about as an Interviewer in general,...

Software Engineering Lab, Samsung Interview | Set 2 · 36 minutes ago

@meya Working solution for question 2 of 4f2f round....

Amazon Interview | Set 53 (For SDE-1) · 1 hour ago

sandeep void rearrange(struct node *head) {...

Given a linked list, reverse alternate nodes and append at the end · 2 hours ago

Neha I think that is what it should return as, in...

Find depth of the deepest odd level leaf node · 2 hours ago