

WRIGHT STATE UNIVERSITY
Department of Computer Science and Engineering
CS7200: Algorithm Design and Analysis

Fall 2024

Assignment 3 (Due: November 22, 2024) (8 pts)

Prasad

Interleaving two strings: String s_3 is said to be an interleaving of strings s_1 and s_2 , if s_3 contains alternating substrings of s_1 and s_2 , and the order of all substrings in s_1 and s_2 are preserved in s_3 . Further, s_1 and s_2 are split into substrings such that the absolute difference in the number of substrings must be at most one ($|n - m| \leq 1$, where n and m are the substring counts of s_1 and s_2). For example, if $s_1 = \text{"AB"}$ and $s_2 = \text{"C"}$, any one of the following can be s_3 obtained by interleaving s_1 and s_2 : $s_3 = \text{"ABC"}$ or $s_3 = \text{"ACB"}$ or $s_3 = \text{"CAB"}$. That is, it is possible to express s_3 as concatenation of alternating substrings coming from s_1 or s_2 , with order preserved in s_3 . For example, if $s_1 = \text{"abbc"} = \text{"ab"} + \text{"bc"}$ and $s_2 = \text{"cde"} = \text{"c"} + \text{"de"}$, then s_3 can be $\text{"ab"} + \text{"c"} + \text{"bc"} + \text{"de"} = \text{"abcbcdde"}$, or "cabdebc" , or "abcdebc" , or "abbccde" , etc.

Implement a program to determine (i) whether a given string s_3 is an interleaving of two other strings, s_1 and s_2 . Additionally, (ii) compute the number of distinct ways this interleaving can be achieved, and, if non-zero, (iii) return the specific sequence of substrings from s_1 and s_2 that can form s_3 through "interleaving, that is, alternation and concatenation".

Detailed Requirements

Given three strings s_1 , s_2 , and s_3 , your task is to:

1. **Check if Interleavable:** Determine if s_3 can be formed by interleaving s_1 and s_2 .
2. **Count Distinct Interleavings:** Calculate the total number of ways s_1 and s_2 can be interleaved to form s_3 .
3. **Return Detailed Interleaving:** If s_3 can be obtained by interleaving s_1 and s_2 , list the sequence of substrings from s_1 and s_2 that can be used to alternate and concatenate to achieve interleaving.

Interleaving Rules (Repeated Differently)

- **Order Preservation:** Characters from s_1 and s_2 must appear in their respective orders within s_3 .
- **Balanced Parts:** When split into substrings, s_1 and s_2 must have similar number of substrings, that is, differ by at most one i.e., $|n - m| \leq 1$ for s_1 with n substrings and s_2 with m substrings.
- **Alternating Structure:** Informally, interleaving is formed by alternating substrings of s_1 ($x_1 + x_2 + \dots + x_p$) and s_2 ($y_1 + y_2 + \dots + y_q$), for example, $s_3 = x_1 + y_1 + x_2 + y_2 + \dots + x_p$, or $s_3 = y_1 + x_1 + y_2 + x_2 + \dots + y_q$, or ...

Input Format:

The input format in a text file is as follows. The first line contains string s1, and the second line contains string s2. And the third line contains a string s3.

Sample Input

aabcc

dbbca

aadbcbcac

Output Format:

The program should produce an output file where:

1. The first line states whether or not an interleaving exists and then provide the total number of interleavings.
2. If an interleaving exists, it should list the sequence of substrings from s1 and s2 that can form s3 through interleaving.

Sample Output

Interleaving exists: True, Count of interleavings: 3

s1 substrings: aa, bc, c

s2 substrings: dbbc, a

Additional Requirements:**python assignment3.py Input.txt**

Your Python program **assignment3.py** should be executable using the above command line command taking an input file argument **Input.txt**. The input file must be in the same directory as the program file, and the output file must be written in the same directory too. Furthermore, if the argument is **Input.txt**, then the output file must be named **Output.txt**. Similarly, for **Input0.txt**, **Input1.txt**, ... the output files must be named **Output0.txt**, **Output1.txt**, ... respectively. Please follow these conventions strictly as any deviation will be penalized.

TURNIN: Upload, to **Assignment 3 DropBox on Pilot**, one zip archive per team (with at most three members), containing at least the following text files and a subfolder with your test files as specified below (with team member names and email addresses included in the first two files):

1. **ReadMe.txt**
2. Python 3.X source code file **assignment3.py** (containing algorithm pseudocode abstracting your implementation, recurrence relation used and an informal argument for its correctness, and overall computational complexity, all succinctly included as documentation/comments)

3. Sample input and corresponding generated output (in the format shown above)
4. A separate subdirectory/subfolder containing other test inputs and outputs you have used to extensively test your code.

Include only one submission per team – not one submission per person (as this may get flagged as a case of plagiarism). Please do not send redundant submissions by email. No late submissions are allowed. We will grade your final submission only so make a submission a day before the due date to play it safe.