WEEK-13 (Dynamic Programming)

1. For a given string **S,** write a program based on dynamic programming for finding the length of the longest substring of S which contains exactly K distinct vowels. Analyze the complexity of the proposed algorithm.

**Sample Input & Output**

Input: s = “artyebui”, k = 2

Output: 6 (Explanation: Longest substring with only 2 vowel is “rtyebu”)

1. A contiguous subsequence of a list S is a subsequence made up of consecutive elements of

S. For instance, if S is 5; 15; -30; 10; -5; 40; 10; then 15; -30; 10 is a contiguous subsequence

but 5; 15; 40 is not. Given a list of numbers, write a linear-time algorithm for the finding the contiguous subsequence of maximum sum.

**Sample Input & Output**

Input: 5; 15; -30; 10; -5; 40; 10;

Output: 10;-5; 40; 10, with a sum of 55.

1. There are two strings ***src*** and ***dest.*** Write a program to convert ***src*** to ***dest*** by applying

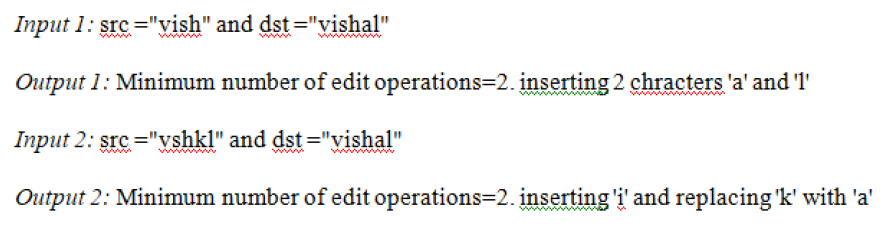
minimum edits operation on the string *src*. The edit operations are as following:

a) Insert a character

b) Delete a character

c) Replace a character

**Sample Input & Output**



1. *Given an array p[] which represents the chain of matrices such that the ith matrix Ai is of dimension p[i-1] x p[i]. Write a function MatrixChainOrder() that should return the minimum number of multiplications needed to multiply the chain.*

**Input: p[] = {40, 20, 30, 10, 30}**

**Output: 26000**

There are 4 matrices of dimensions 40x20, 20x30, 30x10 and 10x30.

Let the input 4 matrices be A, B, C and D. The minimum number of

multiplications are obtained by putting parenthesis in following way

(A(BC))D --> 20\*30\*10 + 40\*20\*10 + 40\*10\*30

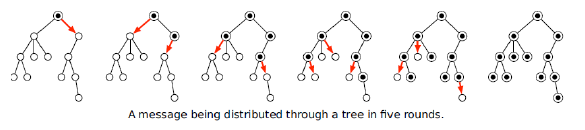


Given S1 = {B, C, D, A, A, C, D}

Then, common subsequences are {B, C}, {C, D, A, C}, {D, A, C}, {A, A, C}, {A, C}, {C, D}, ...

Among these subsequences, {C, D, A, C} is the longest common subsequence. Write program to find this longest common subsequence using dynamic programming.

1. Suppose we need to distribute a message to all the nodes in a rooted tree. Initially, only the root node knows the message. In a single round, any node that knows the message can forward it to at most one of its children. Design an algorithm to compute the minimum number of rounds required for the message to be delivered to all nodes in a given tree.



1. Suppose you are given a sequence of integers separated by + and − signs; for example: 1 + 3 − 2 − 5 + 1 − 6 + 7

You can change the value of this expression by adding parentheses in different places. For example:

1+ 3 − 2 − 5 + 1 − 6 + 7 = −1

(1 + 3 − (2 − 5)) + (1 − 6) + 7 = 9

(1 + (3 − 2)) − (5 + 1) − (6 + 7) = −17

Describe and analyse an algorithm to compute, given a list of integers separated by + and − signs, the maximum possible value the expression can take by adding parentheses. You may only use parentheses to group additions and subtractions; in particular, you are not allowed to create implicit multiplication as in 1 + 3(−2)(−5) + 1 − 6 + 7 = 33.