

A subsequence is anything obtained from a sequence by deleting a subset of elements; the elements of the subsequence need not be contiguous in the original sequence. For example, the strings I, PRO, DAMMM, NROAIG, and DYNAMICPROGRAMMING are all subsequences of the string DYNAMICPROGRAMMING.

1. Suppose we are given two arrays  $A[1..m]$  and  $B[1..m]$ . A *common subsequence* of  $A$  and  $B$  is any subsequence of  $A$  that is also a subsequence of  $B$ . For example, AMI is a common subsequence of DYNAMIC and PROGRAMMING. Describe and analyze an efficient algorithm to compute the length of the *longest* common subsequence of  $A$  and  $B$ .
2. Describe and analyze an efficient algorithm to compute the length of the longest common subsequence of *three* given arrays  $A[1..l]$ ,  $B[1..m]$ , and  $C[1..n]$ .
3. A *shuffle* of two strings  $X$  and  $Y$  is formed by interspersing the characters into a new string, such that the characters of  $X$  and  $Y$  remain in the same order. For example, given the strings 'dynamic' and 'programming', the string 'prodgyrnamammiincg' is indeed a shuffle of the two:

p r o d g y r n a m a m m i i n c g

Given three strings  $A[1..m]$ ,  $B[1..n]$  and  $C[1..m+n]$ , describe an algorithm to determine whether  $C$  is a shuffle of  $A$  and  $B$ .

In all cases, first describe a recursive algorithm and only then transform it into an iterative algorithm.