

Design and Analysis of Algorithms — Lab

R S Milton, C Aravindan, T T Mirnalinee
Department of CSE, SSN College of Engineering

Session 6: Dynamic Programming
Longest Common Subsequence
6 February 2020

1 Longest Common Sequence

Let $A = [A_1, \dots, A_n]$ be a sequence of elements. A subsequence of A is another sequence obtained from A by deleting zero or more elements, without changing the order of the remaining elements.

Let $A[1..m]$ and $B[1..n]$ be two arbitrary arrays. A common subsequence of A and B is both a subsequence of A and a subsequence of B . A recursive definition for the length of LCS between $A[1..i]$ and $B[1..j]$, $L(i, j)$, which gives the length of the longest common subsequence of $A[1..i]$ and $B[1..j]$.

$$L[i, j] = \begin{cases} L[i - 1, j - 1] + 1 & \text{if } A[i] = B[j] \\ \max\{L[i - 1, j], L[i, j - 1]\} & \text{if } A[i] \neq B[j] \end{cases}$$

1. Implement an algorithm using dynamic programming to find the length of LCS.
2. Implement an algorithm to backtrace the LCS itself.