

## Assignment -4(Discussion)

The given problem is to find out the maximum length common subsequence of 2 input arrays.

The solution code provided with this file is a bottom up approach of dynamic programming. A 2D table was used to store the numeric values with row and column as elements of the 2 supplied arrays.

Then row value and column value were used to match both the values, if matched then add 1 to the value of left upper diagonal and store in the row value pair. If not matched, then the maximum of upper and left cell value was taken.

Then the bottom most right most value was retrieved and on the basis of highest value and continued by reducing row and column values. If 2 maximum values were found, any path can be taken which would give different subsequence. In the solution upper cell was given preference as given by the text book.

The dynamic programming solved the problem in  $O(m*n)$  time complexity where  $m, n$  being the row and column values whereas with the normal approach (in the worst case when all the values mismatch) the problem could be solved with  $O(2^n)$  time complexity.