	Marwari University **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Design and Analysis of Algorithms (01CT0512)	**Aim:** Implementing Longest Common Sub-sequence using Dynamic Programming Approach	
Experiment No: 08	**Date: 13\09\2025**	**Enrollment No: 92301733054**

Longest Common Sub-sequence

Code :-

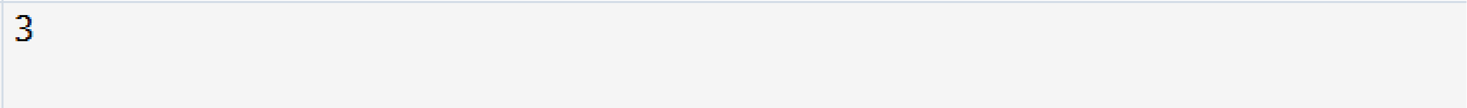
```
// LCS
#include <iostream>
#include <string>
#include <vector>
#include <algorithm>
using namespace std;
int lcs(string &str1, string &str2)
{
   int m = str1.length();
   int n = str2.length();

   vector<vector<int>> l(m + 1, vector<int>(n + 1, 0));

   for (int i = 1; i < m + 1; i++)
   {
      for (int j = 1; j < n + 1; j++)
      {
         if (str1[i - 1] == str2[j - 1])
         {
            l[i][j] = l[i - 1][j - 1] + 1;
         }
         else
         {
            l[i][j] = max(l[i][j - 1], l[i - 1][j]);
         }
      }
   }
   return l[m][n];
}

int main()
{
   string str1 = "longest";
   string str2 = "stone";
   cout << lcs(str1, str2);
   return 0;
       }}
```

Output :-



```
3
```

Conclusion:-

We learnt in this experiment that the Longest Common Subsequence (LCS) problem can be efficiently solved using dynamic programming. It helps in finding the longest sequence present in both strings, which is useful in fields like text comparison, bioinformatics, and data analysis.