Date: 13-10-2025

Experiment No: 11

I. Aim: Implement Longest Common Subsequence.

II. Theory:

Write an algorithm for Longest String matching.

Find length of longest common subsequence for the given stings. Also write the common subsequence.
String1: ADHERT
String2: HIDBEST

Procedure:

III. Program:

Find Length of Longest Common Subsequence from given strings:

```
#include <stdio.h>
#include <string.h>
#include <conio.h>

int max(int a, int b)
{
   return (a > b) ? a : b;
}

int lcsRecursive(char* X, char* Y, int m, int n)
{
   if (m == 0 || n == 0)
return 0;

   if (X[m - 1] == Y[n - 1])
return 1 + lcsRecursive(X, Y, m - 1, n - 1);
   else
return max(lcsRecursive(X, Y, m, n - 1),
            lcsRecursive(X, Y, m - 1, n));
}

int main()
{
   char X[20],Y[20];
   int m,n;
   clrscr();
   printf("Enter First String : ");
   gets(X);
   printf("Enter Second String : ");
   gets(Y);
   m = strlen(X);
   n = strlen(Y);
   printf("Length of LCS is %d\n",
   lcsRecursive(X, Y, m, n));
   getch();
}
```

IV. Output:

____

____

V. Complexity:

Number of characters in string 1= m
Number of characters in string 2=n
Time complexity = O (m*n)

VI. Conclusion: Successfully implemented algorithm for finding length of Longest Common Subsequence from given text.