|  |  |
| --- | --- |
| **NAME:** | Atul Sharma |
| **UID:** | 2021700060 |
| **SUBJECT** | DAA |
| **EXPERIMENT NO :** | 3 |
| **AIM:** | **Experiment on finding the longest common subsequence.** |
| **PROGRAM:** | #include <stdio.h>  #include <string.h>  #define MAX\_LEN 1000  *// function to find the maximum of two integers*  int max(int a, int b) {      return (a > b) ? a : b;  }  *// function to find the LCS of two strings*  int LCS(char str1*[]*, char str2*[]*, int m, int n, char lcs\_string*[]*) {      int lcs[m+1][n+1];      int i, j;  *// building the lcs array*      for (i = 0; i <= m; i++) {          for (j = 0; j <= n; j++) {              if (i == 0 || j == 0)                  lcs[i][j] = 0;              else if (str1[i-1] == str2[j-1]) {                  lcs[i][j] = lcs[i-1][j-1] + 1;                  lcs\_string[lcs[i][j]-1] = str1[i-1]; *// adding the character to LCS string*              }              else                  lcs[i][j] = max(lcs[i-1][j], lcs[i][j-1]);          }      }  *// adding null terminator to the LCS string*      lcs\_string[lcs[m][n]] = '\0';  *// returning the LCS length*      return lcs[m][n];  }  int main() {      char str1[MAX\_LEN], str2[MAX\_LEN], lcs\_string[MAX\_LEN];      int m, n, lcs\_length;      printf("Enter the first string: ");      fgets(str1, MAX\_LEN, stdin);      m = strlen(str1) - 1; *// -1 to remove the newline character*      printf("Enter the second string: ");      fgets(str2, MAX\_LEN, stdin);      n = strlen(str2) - 1;      lcs\_length = LCS(str1, str2, m, n, lcs\_string);      printf("The length of the LCS is: %d\n", lcs\_length);      printf("The LCS string is: %s\n", lcs\_string);      return 0;  } |
| **RESULT ( SNAPSHOT)** | |
| **CONCLUSION:** | In this experiment , I understand the concept of longest common subsequence by using dynamic programming approach. |