**9913 Mark Lopes**

**SE Comps A Batch C**

**LCS code:-**

#include <stdio.h>

#include <string.h>

#define MAX\_LEN 20

char S1[MAX\_LEN] = "MARK";

char S2[MAX\_LEN] = "ARCEUS";

int LCS\_table[MAX\_LEN][MAX\_LEN];

void Lcs()

{

    int m = strlen(S1);

    int n = strlen(S2);

    // Fill LCS\_table with 0's

    for (int i = 0; i <= m; i++)

        LCS\_table[i][0] = 0;

    for (int i = 0; i <= n; i++)

        LCS\_table[0][i] = 0;

    // Fill LCS\_table based on LCS algorithm

    for (int i = 1; i <= m; i++)

    {

        for (int j = 1; j <= n; j++)

        {

            if (S1[i - 1] == S2[j - 1])

                LCS\_table[i][j] = LCS\_table[i - 1][j - 1] + 1;

            else if (LCS\_table[i - 1][j] >= LCS\_table[i][j - 1])

                LCS\_table[i][j] = LCS\_table[i - 1][j];

            else

                LCS\_table[i][j] = LCS\_table[i][j - 1];

        }

    }

    // Finding the LCS string

    int length = LCS\_table[m][n];

    char lcsAlgo[length + 1];

    int i = m, j = n;

    lcsAlgo[length] = '\0';

    while (i > 0 && j > 0)

    {

        if (S1[i - 1] == S2[j - 1])

        {

            lcsAlgo[length - 1] = S1[i - 1];

            i--;

            j--;

            length--;

        }

        else if (LCS\_table[i - 1][j] >= LCS\_table[i][j - 1])

        {

            i--;

        }

        else

        {

            j--;

        }

    }

    // Printing the results

    printf("S1 : %s\nS2 : %s\n", S1, S2);

    printf("LCS: %s\n", lcsAlgo);

}

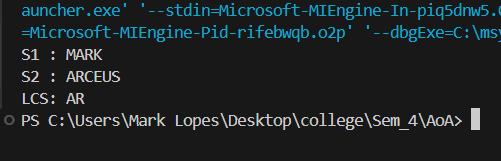
int main()

{

    Lcs();

    return 0;

}

****

