

CSE2012

DAA LAB

Name: Preyash

Registration Number: 20BPS1022

Ex 07:

Code:

Recursive Method

```
#include<iostream>
using namespace std;
int func(string a, string b, int m, int n)
{
    if (n == 0 || m == 0)
        return 0;

    if (a[m - 1] == b[n - 1])
    {
        return func(a, b, m - 1, n - 1) + 1;
    }
    return max(func(a, b, m - 1, n), func(a, b, m, n - 1));
}

int main(){
    string a, b;
    cin >> a >> b;
    int n= a.length();
    int m=b.length();

    cout<<"The longest common subsequence is: ";
    cout<< func(a,b,n,m)<<endl;
    cout<< func(a, b, n, m);
    return 0;
}
```

Output:

```
ABCD
ANDEFGH
The longest common subsequence is: 3
```

Code:

Dynamic Programming

```
#include<iostream>
using namespace std;
int func(string a, string b, int n, int m)
{
    int dp[n + 1][m + 1];

    for (int i = 0; i <= n; i++)
    {
        for (int j = 0; j <= m; j++)
        {
            if (i == 0 || j == 0)
                dp[i][j] = 0;

            else if (a[i - 1] == b[j - 1])
                dp[i][j] = 1 + dp[i - 1][j - 1];

            else
            {
                dp[i][j] = max(dp[i - 1][j], dp[i][j - 1]);
            }
        }
    }
    return dp[n][m];
}

int main(){
    string a, b;
    cin >> a >> b;
    int n= a.length();
    int m=b.length();

    cout<<"The longest common subsequence is: ";
    cout<< func(a,b,n,m)<<endl;
    cout<< func(a, b, n, m);
    return 0;
}
```

Output:

```
3 4 (2) 1 1 longest_common_subsequence  
ABCD  
AHCDEFGH  
The longest common subsequence is: 3
```