

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
AAD Practical 8

Institute of Computer Technology
B. Tech Computer Science and Engineering

Sub: Algorithm Analysis and Design

Practical 8

Problem : A subsequence is a sequence that can be derived from another sequence by deleting some elements without changing the order of the remaining elements. Longest common subsequence (LCS) of 2 sequences is a subsequence, with maximal length, which is common to both the sequences.

Given two sequence of characters, P = M, N, O, M and Q = M, L, N, O, M, find any one longest common subsequence.

In case multiple solutions exist, print any of them. It is guaranteed that at least one non-empty common subsequence will exist.

Code :

```
import YSL_io

def LCS(P, Q):
    s1, s2 = len(P), len(Q)
    dp = [[0] * (s2 + 1) for _ in range(s1 + 1)]

    for i in range(1, s1 + 1):
        for j in range(1, s2 + 1):
            if P[i - 1] == Q[j - 1]:
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
AAD Practical 8

```
dp[i][j] = dp[i - 1][j - 1] + 1
else:
dp[i][j] = max(dp[i - 1][j], dp[i][j - 1])

i, j = s1, s2
lcs = []

while i > 0 and j > 0:
    if P[i - 1] == Q[j - 1]:
        lcs.append(P[i - 1])
        i -= 1
        j -= 1
    elif dp[i - 1][j] > dp[i][j - 1]:
        i -= 1
    else:
        j -= 1

return ''.join(str(y) for y in lcs[::-1])
```

Demo inputs :

P = "MNOAM"

Q = "MLNOM"

P=YSL_io.inputCYN("\n\tEnter the first sequence P : ")

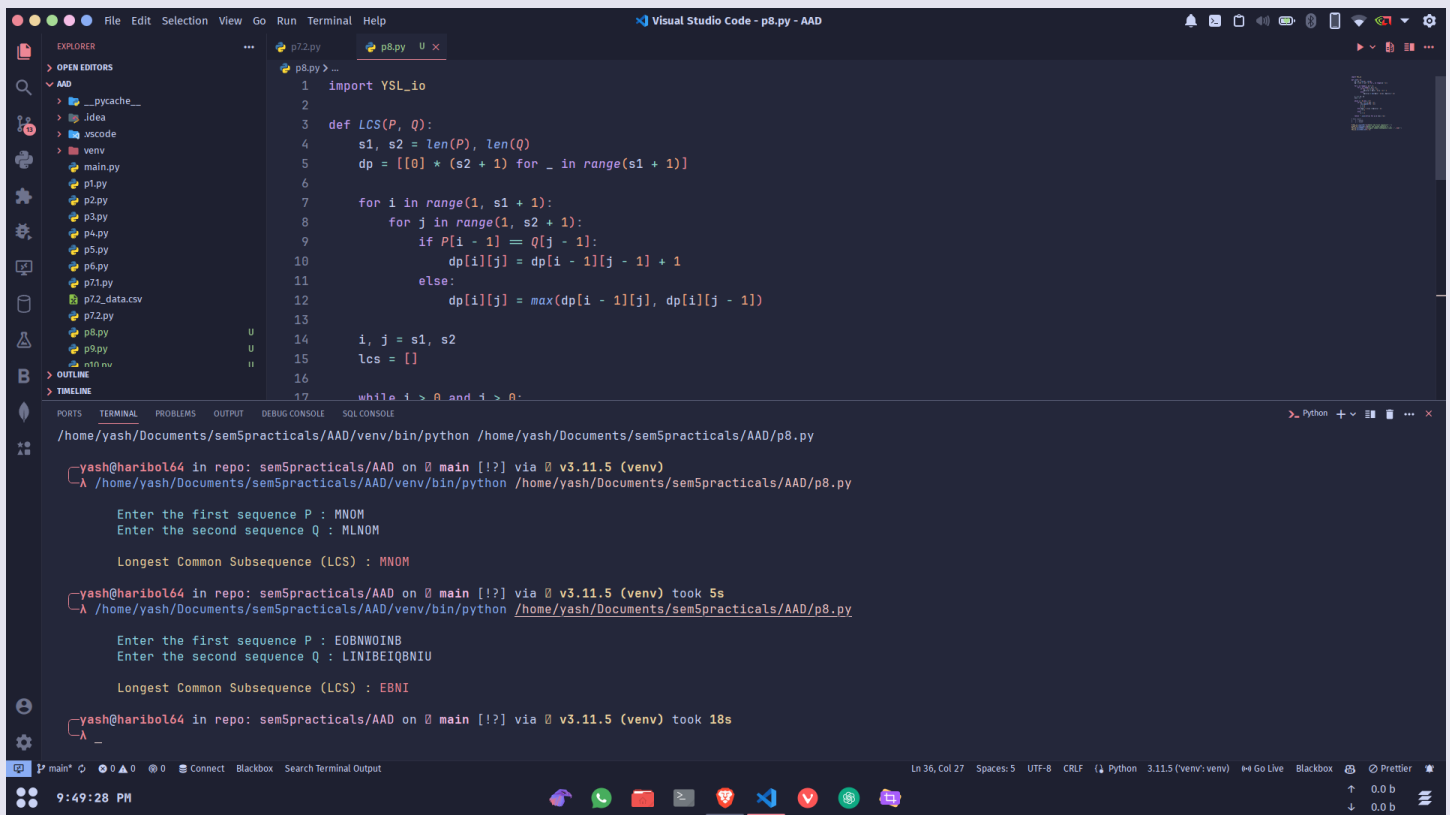
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
AAD Practical 8

```
Q=YSL_io.inputCYN("\tEnter the second sequence Q : ")

YSL_io.printORNG('\n\tLongest Common Subsequence (LCS) : ', end='')

YSL_io.printRED(LCS(P, Q))
```

Screenshot:



The screenshot displays the Visual Studio Code editor with a Python file named `p8.py` open. The script implements a dynamic programming solution for the Longest Common Subsequence (LCS) problem. The code defines a function `LCS(P, Q)` that takes two strings `P` and `Q` as input and returns the LCS. The function uses a 2D array `dp` to store the results of subproblems. The main function `main` prompts the user to enter two sequences `P` and `Q`, and then prints the LCS using the `LCS` function.

```
1 import YSL_io
2
3 def LCS(P, Q):
4     s1, s2 = len(P), len(Q)
5     dp = [[0] * (s2 + 1) for _ in range(s1 + 1)]
6
7     for i in range(1, s1 + 1):
8         for j in range(1, s2 + 1):
9             if P[i - 1] == Q[j - 1]:
10                 dp[i][j] = dp[i - 1][j - 1] + 1
11             else:
12                 dp[i][j] = max(dp[i - 1][j], dp[i][j - 1])
13
14     i, j = s1, s2
15     lcs = []
16
17     while i > 0 and j > 0:
```

The terminal output shows the execution of the script. It prompts the user to enter the first sequence `P` and the second sequence `Q`. The first input is `MNOM` and the second is `MLNOM`. The output shows the LCS as `MNOM`. The second input is `EOBNWOINB` and the second is `LINIBEQBNIU`. The output shows the LCS as `EBNI`.

```
/home/yash/Documents/semSpracticals/AAD/venv/bin/python /home/yash/Documents/semSpracticals/AAD/p8.py
yash@haribol64 in repo: semSpracticals/AAD on 0 main [!?] via 0 v3.11.5 (venv)
/home/yash/Documents/semSpracticals/AAD/venv/bin/python /home/yash/Documents/semSpracticals/AAD/p8.py
Enter the first sequence P : MNOM
Enter the second sequence Q : MLNOM
Longest Common Subsequence (LCS) : MNOM
yash@haribol64 in repo: semSpracticals/AAD on 0 main [!?] via 0 v3.11.5 (venv) took 5s
/home/yash/Documents/semSpracticals/AAD/venv/bin/python /home/yash/Documents/semSpracticals/AAD/p8.py
Enter the first sequence P : EOBNWOINB
Enter the second sequence Q : LINIBEQBNIU
Longest Common Subsequence (LCS) : EBNI
yash@haribol64 in repo: semSpracticals/AAD on 0 main [!?] via 0 v3.11.5 (venv) took 18s
```