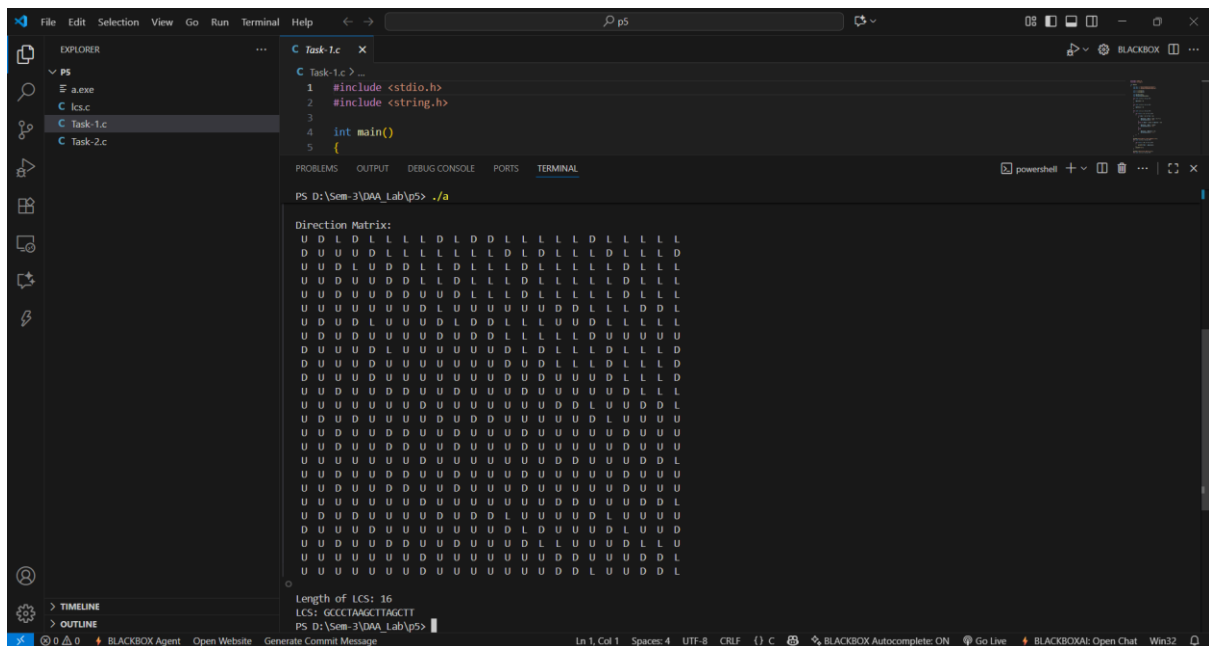
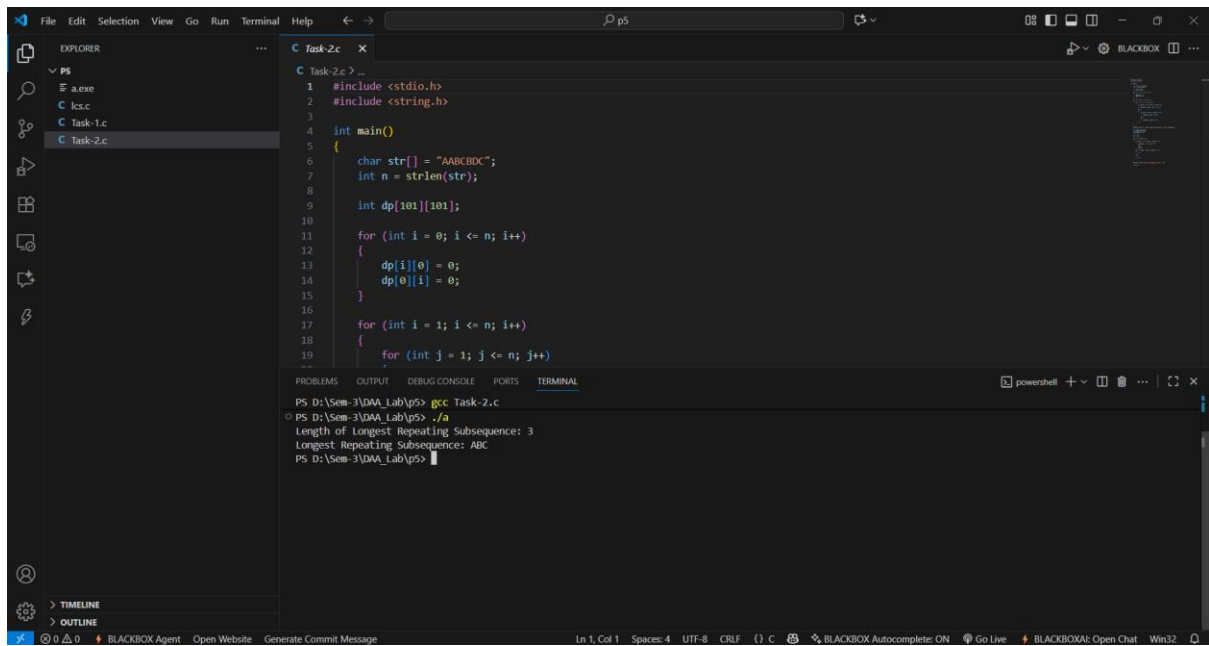


Roll No: 39



Task-2:



```
File Edit Selection View Go Run Terminal Help
C Task-2.c x
1 #include <stdio.h>
2 #include <string.h>
3
4 int main()
5 {
6     char str[] = "AABCBCD";
7     int n = strlen(str);
8
9     int dp[101][101];
10
11     for (int i = 0; i <= n; i++)
12     {
13         dp[i][0] = 0;
14         dp[0][i] = 0;
15     }
16
17     for (int i = 1; i <= n; i++)
18     {
19         for (int j = 1; j <= n; j++)
20         {
21             if (str[i-1] == str[j-1] && i != j)
22                 dp[i][j] = dp[i-1][j-1] + 1;
23             else
24                 dp[i][j] = max(dp[i-1][j], dp[i][j-1]);
25         }
26     }
27
28     printf("Length of Longest Repeating Subsequence: %d\n", dp[n][n]);
29     printf("Longest Repeating Subsequence: ");
30     for (int i = 1; i <= n; i++)
31     {
32         if (dp[i][dp[n][n]] > dp[i-1][dp[n][n]])
33             printf("%c", str[i-1]);
34     }
35     printf("\n");
36 }
```

PROBLEMS OUTPUT DEBUG CONSOLE PORTS TERMINAL

```
PS D:\Sem-3\DAALab\ps> gcc Task-2.c
PS D:\Sem-3\DAALab\ps> ./a
Length of Longest Repeating Subsequence: 3
Longest Repeating Subsequence: ABC
PS D:\Sem-3\DAALab\ps>
```

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF (1) C BLACKBOX Autocomplete: ON Go Live BLACKBOXAI: Open Chat Win32

Task-3:

Longest Common Subsequence: X +

← → ↺ ↻ ↵ ↶ ↷ ↸ ↹ ↺ ↻ ↵ ↶ ↷ ↸ ↹

leetcode.com/problems/longest-common-subsequence/submissions/1790478045/

🔍 ⭐ 📄 📁 🌐 ⋮

🏠 📋 Problem List < > 🔍

📄 Description 📄 Editorial 📄 Solutions 📄 Submissions

1143. Longest Common Subsequence

Solved

Medium Topics Companies Hint

Given two strings `text1` and `text2`, return the length of their longest **common subsequence**. If there is no common subsequence, return 0.

A **subsequence** of a string is a new string generated from the original string with some characters (can be none) deleted without changing the relative order of the remaining characters.

- For example, `"ace"` is a subsequence of `"abcde"`.

A **common subsequence** of two strings is a subsequence that is common to both strings.

Example 1:

Input: `text1 = "abcde", text2 = "ace"`
Output: 3
Explanation: The longest common subsequence is `"ace"` and its length is 3.

Example 2:

Input: `text1 = "abc", text2 = "abc"`
Output: 3
Explanation: The longest common subsequence is `"abc"` and its length is 3.

Example 3:

Code

Submit Ctrl Enter

C++ Auto

```
1 class Solution
2 {
3 public:
4     int longestCommonSubsequence(string text1, string text2)
5     {
6         int n = text1.size();
7         int m = text2.size();
8         vector<vector<int>> dp(n + 1, vector<int>(m + 1, 0));
9         for (int i = 1; i <= n; ++i)
10         {
11             for (int j = 1; j <= m; ++j)
12             {
13                 if (text1[i - 1] == text2[j - 1])
14                 {
15                     dp[i][j] = dp[i - 1][j - 1] + 1;
16                 }
17                 else
18                 {
19                     dp[i][j] = max(dp[i - 1][j], dp[i][j - 1]);
20                 }
21             }
22         }
23         return dp[n][m];
24     }
25 };
26
27
28
29
```

Saved Ln 29, Col 1

Testcase Test Result

Leet X Accepted X

All Submissions

Accepted 47 / 47 testcases passed submitted at Oct 04, 2025 00

Solution

Runtime 24 ms | Beats: 71.52% Analyze Complexity

Memory 27.35 MB | Beats: 69.33%

15% 10% 5% 0% 0ms 22ms 42ms

Code C++

class Solution {

Longest Common Subsequence: X +

← → ↺ ↻ ↵ ↶ ↷ ↸ ↹ ↺ ↻ ↵ ↶ ↷ ↸ ↹

leetcode.com/problems/longest-common-subsequence/

🔍 ⭐ 📄 📁 🌐 ⋮

🏠 📋 Problem List < > 🔍

📄 Description 📄 Editorial 📄 Solutions 📄 Submissions

1143. Longest Common Subsequence

Solved

Medium Topics Companies Hint

Given two strings `text1` and `text2`, return the length of their longest **common subsequence**. If there is no common subsequence, return 0.

A **subsequence** of a string is a new string generated from the original string with some characters (can be none) deleted without changing the relative order of the remaining characters.

- For example, `"ace"` is a subsequence of `"abcde"`.

A **common subsequence** of two strings is a subsequence that is common to both strings.

Example 1:

Input: `text1 = "abcde", text2 = "ace"`
Output: 3
Explanation: The longest common subsequence is `"ace"` and its length is 3.

Example 2:

Input: `text1 = "abc", text2 = "abc"`
Output: 3
Explanation: The longest common subsequence is `"abc"` and its length is 3.

Example 3:

Code

Submit Ctrl Enter

C++ Auto

```
1 class Solution
2 {
3 public:
4     int longestCommonSubsequence(string text1, string text2)
5     {
6         int n = text1.size();
7         int m = text2.size();
8         vector<vector<int>> dp(n + 1, vector<int>(m + 1, 0));
9     }
10 };
11
12
13
14
```

Saved Ln 29, Col 1

Testcase Test Result

Leet X Accepted X

All Submissions

Accepted 47 / 47 testcases passed submitted at Oct 04, 2025 00

Solution

Runtime 24 ms | Beats: 71.52% Analyze Complexity

Memory 27.35 MB | Beats: 69.33%

15% 10% 5% 0% 0ms 22ms 42ms

Code C++

class Solution {

Case 1 Case 2 Case 3 +

text1 = "abcde"

text2 = "ace"