**PRACTICAL NO. 5**

*Name : Anjali Lanjewar*

*Section: A4-B2*

*Roll no.-25*

**Aim:** Implement a dynamic algorithm for Longest Common Subsequence (LCS) to find the length and LCS for DNA sequences.

**Problem Statement:**

(i) DNA sequences can be viewed as strings of A, C, G, and T characters, which

represent nucleotides. Finding the similarities between two DNA sequences are an important computation performed in bioinformatics.

[Note that a subsequence might not include consecutive elements of the original sequence.]

**TASK 1:** Find the similarity between the given X and Y sequence.

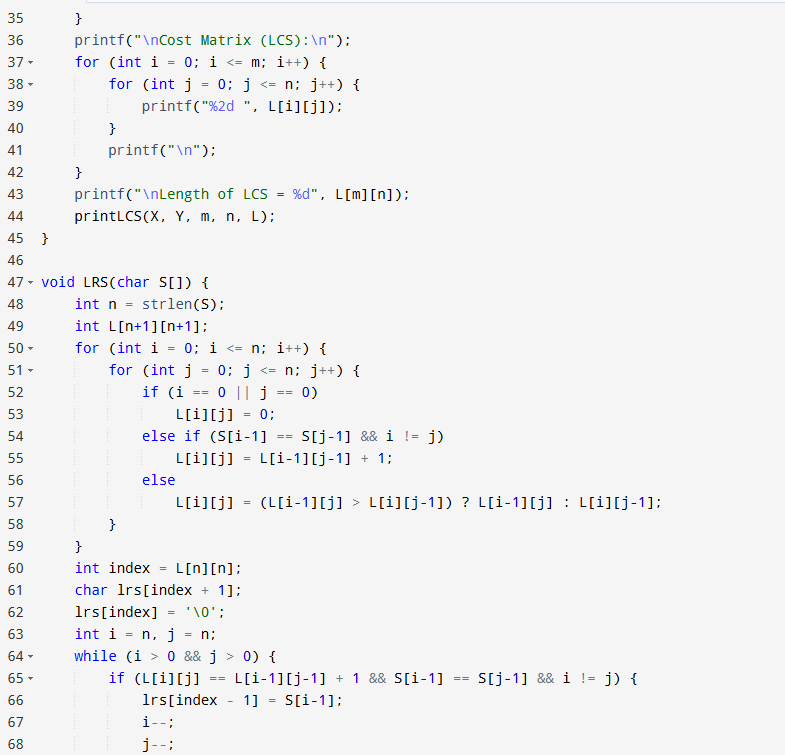
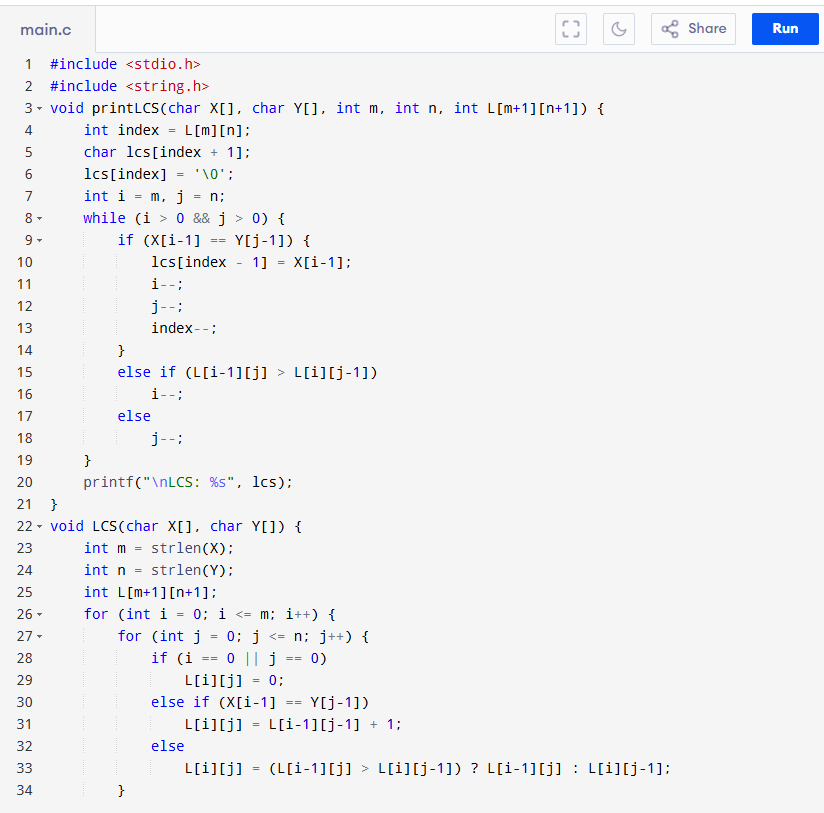
X=AGCCCTAAGGGCTACCTAGCTT

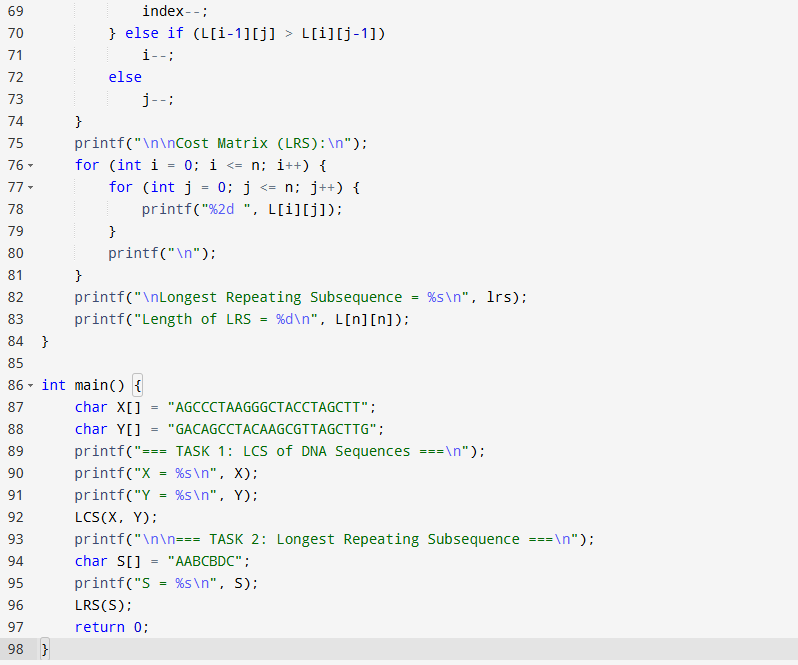
Y= GACAGCCTACAAGCGTTAGCTTG

Output: Cost matrix with all costs and direction, final cost of LCS and the LCS.

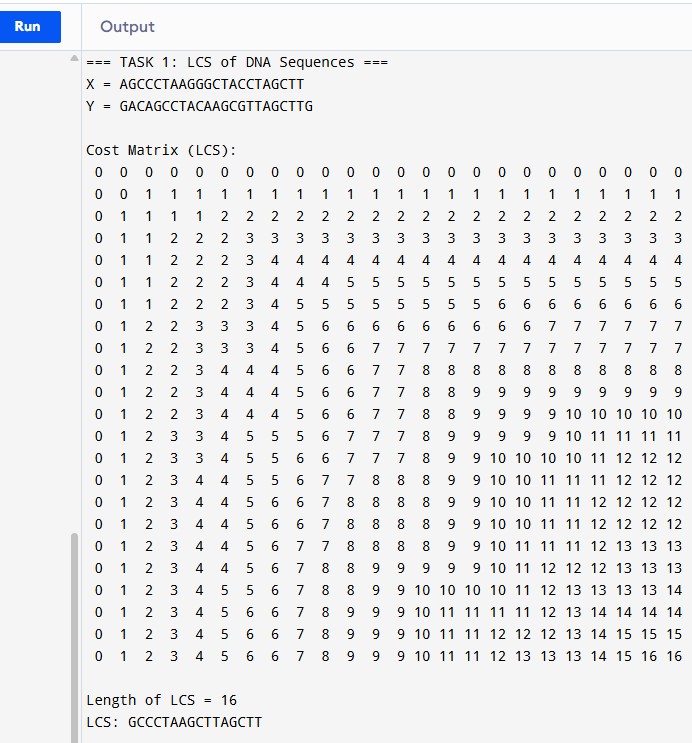
Length of LCS=16

**CODE :**





**OUTPUT :**



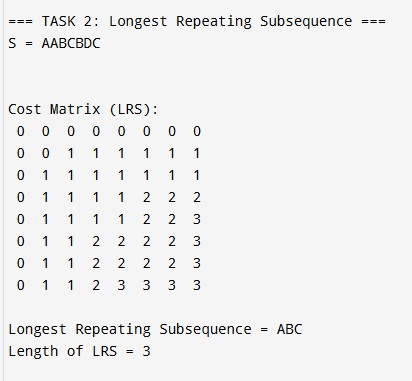
**TASK-2:** Find the longest repeating subsequence (LRS). Consider it as a variation of the longest common subsequence (LCS) problem.

Let the given string be S. You need to find the LRS within S. To use the LCS framework, you

effectively compare S with itself. So, consider string1 = S and string2 = S.

Example:

AABCBDC

LRS= ABC or ABD 

**LeetCode Assessment :**

