Algorithm:

Step1: Create a class and create a method to find the longest increasing subsequence and create dynamic array to store the values.

Step2: Initialize an array `sequence` to store the subsequence elements. Initialize a variable ` last\_index ` to 0. This will store the index of the last element in the longest increasing subsequence.

Step3: Iterate over the input array from index 1 to n-1: - For each index i, iterate over the elements from index 0 to i-1

Step4: If the current element arr[i] is greater than the element arr[j], update lis[i] as the maximum between lis[i] and lis[j] + 1.

Step5: If lis[i] is updated, check if it is greater than maxLis. If so, update maxLength to lis[i] and last\_Index to i.

Step6: print the length of subsequence and longest subsequence array to using the for looping statement.