```java
// Creates a linked list used in the chaining method
public class LCS {

    public LCS(){

    }

    public String calLCS(String X, String Y) {
        int XLen = X.length();
        int YLen = Y.length();
        if(XLen == 0 || YLen == 0)
        {
            return "";
        }
        else if (X.charAt(XLen-1) == Y.charAt(YLen-1))
        {
            return X.charAt(XLen-1) + calLCS(X.substring(0, XLen-1),Y.substring(0, YLen-1));
        }
        else
        {
            int subXLen = calLCS(X.substring(0, XLen-1),Y).length();
            int subYLen = calLCS(X,Y.substring(0, YLen-1)).length();

            if (subXLen>subYLen)
            {
                return calLCS(X.substring(0, XLen-1),Y);
            }
            else
            {
                return calLCS(X,Y.substring(0, YLen-1));
            }
        }
    }
    public String calLCS2(String X, String Y) {
        int XLen = X.length();
        int YLen = Y.length();
        int[][] arr = new int[XLen + 1][YLen + 1];

        for (int i = XLen - 1; i >= 0; i--)
        {
            for (int j = YLen - 1; j >= 0; j--)
            {
                if (X.charAt(i) == Y.charAt(j))
                    arr[i][j] = arr[i + 1][j + 1] + 1;
                else
                    arr[i][j] = Math.max(arr[i + 1][j], arr[i][j + 1]);
            }
        }
        int i = 0, j = 0;
        String sb = "";
        while (i < XLen && j < YLen)
        {
            if (X.charAt(i) == Y.charAt(j))
            {
                sb= sb+ X.charAt(i);
                i++;
                j++;
            }
            else if (arr[i + 1][j] >= arr[i][j + 1])
                i++;
            else
                j++;
        }
        return sb;
    }
}

```