**Sequence Pattern Matching –**

**( both the strings matching with each other)**

**1)Suppose string s1= “AXY” and string s2= “ABCDXEF”. Always the smaller substring matches with the larger string.**

**2)The maximum range a smaller substring can match is the length of the smallest string**

**3)So, the length can be x=Math.min( length of s1 , length of s2)**

**4) After finding LCS of s1 and s2 , if the LCS length ==x -The it is true or else it is false.**

**CODE :**

**public class PatternMatching**

**{**

**public static void main(String args[])**

**{**

**String s1="AXY";**

**String s2="BXYACDE";**

**int x=Math.min(s1.length(),s2.length());**

**int LCSlength=LCS(s1,s2);**

**if(LCSlength==x)**

**{**

**System.out.println("True");**

**}**

**else**

**{**

**System.out.println("False");**

**}**

**}**

**public static int LCS(String s1,String s2)**

**{**

**int n1=s1.length();**

**int n2=s2.length();**

**int[][] dp=new int[n1+1][n2+1];**

**for(int i=0 ;i <n1+1 ;i++)**

**{**

**for(int j=0; j<n2+1 ;j++)**

**{**

**if(i==0 || j==0 )**

**{**

**dp[i][j]=0;**

**}**

**else if(s1.charAt(i-1)==s2.charAt(j-1))**

**{**

**dp[i][j]=1+dp[i-1][j-1];**

**}**

**else**

**{**

**dp[i][j]=Math.max(dp[i][j-1],dp[i-1][j]);**

**}**

**}**

**}**

**return dp[n1][n2];**

**}**

**}**