**最长公共子序列**

已知两个字符串A、B，就它们的最长公共子序列。

一、什么是最长公共子序列

**子串**：字符串的一个**连续的部分**。

**子序列**：字符串中，可以由不连续的部分拼成，但先后次序不能变。1 2 3 4 5 6

的子序列：2 3, 1 4 5 ，3 5 6 都是子序列，

  举例如下，串 “abcdea”和“aebcda”的最长子序列是：“abcda”。

二、求最长公共子序列(LCS)

1、序列str1和序列str2长度分别为m和n；

L[m][n]表示：str1取前m个字符，str2取前n个时，最长公共子序列的长度。

明显： 如果m=0或n=0，L[m][n]=0

假设：str1=abcdea，str2=aebcda

那么：L[1][1]=?? str1=a str2=a

如果str1[1]=str2[1],L[1][1]=1。 **这种思考方法不对**

**应该这样思考：**

L[1][1]=max{ L[0][0] +1 如果 str1[1]=str2[1]

L[0][0] 如果 str1[1]!=str2[1]

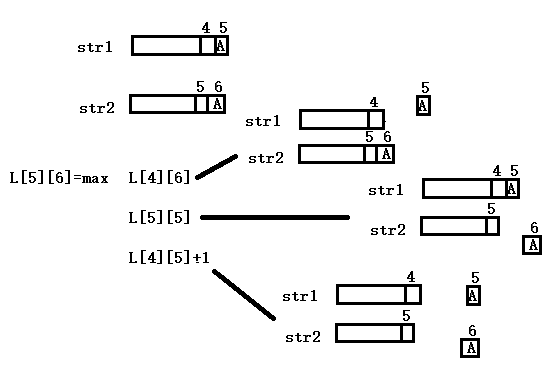
L[0][1]

L[1][0]

}

也就是L[1][1]是由三个选项中的最大值

如：如计算L[5][6]时，我们要考虑以前的子问题L[4][6]、L[5][5]、L[4][5],并且如果**以前的子问题最优**，那么，这个**结果也最优**



也就是说，str1:????A、str2:&&&&&A的最后一个字母A，可能是str1:????A与&&&&&获得更长的公共子序列，也可能……。

所以： L[i][j]=max L[i-1][j]

L[i][j-1]

L[i-1][j-1]+1 如果str1[i]=str2[j]

L[i-1][j-1] 如果str1[i]!=str2[j]

如：str1=ABCBDAB str2=BDCABA

L[0][0]=0

L[1][0]=0

L[0][1]=0

L[1][1]=0

L[2][1]= L[1][0]+1

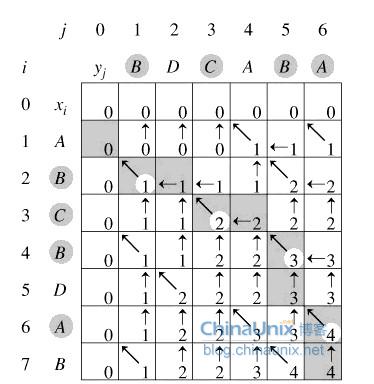


图1 效果演示图

   根据上图，我们可以得到其中公共子串：B C B A 和 B D A B。

   总感觉，上面这个过程说的不是很清楚，但是不知道怎么才能更加清楚的表述？？纠结啊。