# 最长公共子序列

## 1　题目描述

我们有两个字符串m和n，如果它们的子串a和b内容相同，则称a和b是m和n的公共子序列。子串中的字符不一定在原字符串中连续。

例如字符串“abcfbc”和“abfcab”，其中“abc”同时出现在两个字符串中，因此“abc”是它们的公共子序列。此外，“ab”、“af”等都是它们的字串。

现在给你两个任意字符串（不包含空格），请帮忙计算它们的最长公共子序列的长度。

### 1.1　输入描述:

输入包含多组数据。每组数据包含两个字符串m和n，它们仅包含字母，并且长度不超过1024。

### 1.2　输出描述:

对应每组输入，输出最长公共子序列的长度。

### 1.3　输入例子:

abcfbc abfcab

programming contest

abcd mnp

### 1.4　输出例子:

4

2

0

## 2　解题思路

一个数列S，如果分别是两个或多个已知数列的子序列，且是所有符合此条件序列中最长的，则S称为已知序列的最长公共子序列（LCS，Longest Common Subsequences）。对于一般性的LCS问题（即任意数量的序列）是属于NP-hard。但当序列的数量确定时，问题可以使用动态规划（Dynamic Programming）在多项式时间解决。

最长公共子序列的问题常用于解决字符串的相似度，是一个非常实用的算法。举个例子，myblogs这个字符串中子序列有多少个呢？很显然有27个，比如其中的my,bgs等等都是其子序列，我们可以看出**子序列不见得一定是连续的，连续的称为子串**。

理解了子序列的概念，现在可以延伸到两个字符串。如abcdegfg和acdfgbdef，它们的最长公共子序列如图2-1所示。



图2-1　abcdegfg和acdfgbdef的最长公共子序列

从图中可知最长公共子序列为acdfg和abdfg，最长公共子序列的个数不是唯一的，可能会有两个或两个以上，但是长度一定是唯一的，图2-1中最长公共子序列的长度为4。

### 2.1　解决方案

#### 2.1.1　枚举法

这种方法是最简单，时间复杂度非常高。一个长度为N的字符串，其子序列有2N个，每个子序列（假设长度为L）要在第二个长度为M的字符串中去匹配，匹配一次需要用时O(L\*M)，共有2N个字序列，这些子序列的平均长度为N/2，所以总的时间复杂度为O(N/2\*M\*2N)= O(N\*M\*2N-1)。穷举法是不可取的。

#### 2.1.2　动态规划算法

动态规划的一个计算最长公共子序列的方法如下，以两个序列X、Y为例子。设有二维数组f[i][j]表示X的i位和Y的j位之前的最长公共子序列的长度，则有：

此时，f[i][j]中最大的数便是X和Y的最长公共子序列的长度，依据该数组回溯，便可找出最长公共子序列。该算法的空间、时间复杂度均为O(n2)，经过优化后，空间复杂度可为O(n)。

