最短编辑距离

1　题目描述

UNIX 系统下有一个行编辑器 ed，它每次只对一行文本做删除一个字符、插入一个字符或替换一个字符三种操作。例如某一行的内容是“ABC”，经过把第二个字符替换成“D”、删除第一个字符、末尾插入一个字符“B”，这三步操作后，内容就变成了“DCB”。即“ABC”变成“DCB”需要经过 3 步操作，我们称它们的编辑距离为 3。

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短编辑距离。

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，输出最短编辑距离。

1.3　输入例子:

```
ABC CBCD
ABC DCB
```

1.4　输出例子:

```
2
3
```

2　解题思路

设 A 和 B 是 2 个字符串。要用最少的字符操作将字符串 A 转换为字符串 B。这里所说的字符操作包括:

(1) 删除一个字符;

(2) 插入一个字符;

(3) 将一个字符改为另一个字符。

将字符串 A 变换为字符串 B 所用的最少字符操作数称为字符串 A 到 B 的编辑距离。设 A 的长度为 m，B 的长度为 n 创建一个二维数组 d，大小为(m+1)*(n+1)，来记录 a_1-a_m 与 b_1-b_n 之间的

编辑距离，要递推时，需要考虑对其中一个字符串的删除操作、插入操作和替换操作分别花费的开销，从中找出一个最小的开销即为所求结果。

操作步骤：

(一) 情况一：当 A 的长度为 0，B 的长度为 j 时，最小编辑距离就是 j。

(二) 情况二：当 A 的长度为 i，B 的长度为 0 时，最小编辑距离就是 i。

(三) 情况三:当A的长度为i,B的长度为j时,d[i][j]=min{d[i-1][j]+1,d[i][j-1]+1, d[i-1][j-1]+(A[i]==B[j]?0:1)}

其中：

- d[i][j]表示 A 的前 i 个字符和 B 的前 j 个字符相同后的最短距离。
- dp[i][j]来自于三种状态
    - 删除，d[i-1][j]+1，$A_1,\dots,A_{i-1}$ 经过操作可以变成了 $B_1,\dots,B_j$，那么 $A_1,\dots,A_{i-1}A_i$ 变为 $B_1,\dots,B_j$ 一定要删除 A_i。
    - 插入，d[i][j-1]+1，$A_1,\dots,A_i$ 经过操作可以变成了 $B_1,\dots,B_{j-1}B_j$，那么 $A_1,\dots,A_{i-1}A_i$ 变为 $B_1,\dots,B_{j-1}B_j$ 一定要添加一个字符。
    - 替换，如果 A[i]=B[j]，可以不进行额外的操作，那么有 d[i][j]=d[i-1][j-1]，如果不 A[i]≠B[j]，那么就要进行一次替换操作，有 d[i][j]=d[i-1][j-1]+1。

注意：此处字符串中字符开始的下标从 1 开始计算

根据分析可以得到递推方程：

$$d[i][j]=\begin{cases} j & i=0 \\ i & j=0 \\ \min\{d[i-1][j]+1, d[i][j-1]+1, d[i-1][j-1]+f(i,j)\} & i>0\ and\ j>0 \end{cases}$$

$$f(i,j)=\begin{cases} 1 & A[i]\neq B[j] \\ 0 & A[i]=B[j] \end{cases} \quad i>0\ and\ j>0$$