

A题题解报告：

题意：在 A 序列和 B 序列中分别找字符串 a 和 b，使得 重合度 $xLCS(a, b) - y|a| - z*|b|$ 最大。LCS(a, b)为 a 和 b 的最长公共子序列，|a| 为 a 串的长度。

令 $dp[i][j]$ 表示 a 以 A 序列的第 i 个字符为结尾、b 以 B 序列的第 j 个字符为结尾的最大重合度。

依此从头到尾枚举 i 和 j。

$dp[i][j]$ 有两种转移情况，枚举到 $A[i] == B[j]$ 或者 $A[i] != B[j]$ 。

前者情况， $dp[i][j] = \max(dp[i-1][j-1] + x - y - z, dp[i-1][j] - y, dp[i][j-1] - z, x - y - z)$ ，

max 里分别代表的是

i 和 j 均前进一步（到达现在的 i 和 j）、

只 i 前进一步（a 串长了，要减去一个 y）、

只 j 前进一步（b 串长了，要减去一个 z）、

起始位置且切换到 i、j。

因为我们最终找到的 a 串 b 串，大概率不是序列的前 k 位，而是处于中间的某个串，所以这个转移非常核心。

后者情况， $dp[i][j] = \max(dp[i-1][j] - y, dp[i][j-1] - z, -y - z)$

这里理解和前者情况类似，

最后一个转移也是，用来改变起点的。

时间复杂度 $O(nm)$

嵌套循环里 i 从头跑到尾，j 从头跑到尾巴