**记忆化搜索**

**1568 滑雪**

Time Limit: 1000 ms

Memory Limit: 256 mb

Michael喜欢滑雪这并不奇怪， 因为滑雪的确很刺激。可是为了获得速度，滑的区域必须向下倾斜，而且当你滑到坡底，你不得不再次走上坡或者等待升降机来载你。Michael想知道在一个区域中最长的滑坡。区域由一个二维数组给出。数组的每个数字代表点的高度。下面是一个例子

1 2 3 4 5

16 17 18 19 6

15 24 25 20 7

14 23 22 21 8

13 12 11 10 9

一个人可以从某个点滑向上下左右相邻四个点之一，当且仅当高度减小。在上面的例子中，一条可滑行的滑坡为24-17-16-1。当然25-24-23-...-3-2-1更长。事实上，这是最长的一条。

**输入输出格式**

**输入描述:**

多组测试数据。输入的第一行表示区域的行数R和列数C(1 <= R,C <= 100)。下面是R行，每行有C个整数，代表高度h，0<=h<=10000。

**输出描述:**

输出最长区域的长度。

**输入输出样例**

**输入样例#:**

5 5

1 2 3 4 5

16 17 18 19 6

15 24 25 20 7

14 23 22 21 8

13 12 11 10 9

**输出样例#:**

25

**字符串相关的动态规划**

**leetcode72 编辑距离**

给你两个单词 word1和 word2，请返回将 word1 转换成 word2所使用的最少操作数。

你可以对一个单词进行如下三种操作：

插入一个字符

删除一个字符

替换一个字符

**示例 1：**

**输入：**word1 = "horse", word2 = "ros"

**输出：**3

**解释：**

horse -> rorse (将 'h' 替换为 'r')

rorse -> rose (删除 'r')

rose -> ros (删除 'e')

**示例 2：**

**输入：**word1 = "intention", word2 = "execution"

**输出：**5

**解释：**

intention -> inention (删除 't')

inention -> enention (将 'i' 替换为 'e')

enention -> exention (将 'n' 替换为 'x')

exention -> exection (将 'n' 替换为 'c')

exection -> execution (插入 'u')

**提示：**

0 <= word1.length, word2.length <= 500

word1 和 word2 由小写英文字母组成