

NOIP2008 提高组复赛试题

考试时间：3 小时

1. 笨小猴(word.pas/c/cpp)

【问题描述】

笨小猴的词汇量很小，所以每次做英语选择题的时候都很头疼。但是他找到了一种方法，经试验证明，用这种方法去选择选项的时候选对的几率非常大！

这种方法的具体描述如下：假设 maxn 是单词中出现次数最多的字母的出现次数， minn 是单词中出现次数最少的字母的出现次数，如果 $\text{maxn}-\text{minn}$ 是一个质数，那么笨小猴就认为这是个 Lucky Word，这样的单词很可能就是正确的答案。

【输入】

输入文件 word.in 只有一行，是一个单词，其中只可能出现小写字母，并且长度小于 100。

【输出】

输出文件 word.out 共两行，第一行是一个字符串，假设输入的的单词是 Lucky Word，那么输出“Lucky Word”，否则输出“No Answer”；

第二行是一个整数，如果输入单词是 Lucky Word，输出 $\text{maxn}-\text{minn}$ 的值，否则输出 0。

【输入输出样例 1】

word.in	word.out
error	Lucky Word 2

【输入输出样例 1 解释】

单词 error 中出现最多的字母 r 出现了 3 次，出现次数最少的字母出现了 1 次， $3-1=2$ ，2 是质数。

【输入输出样例 2】

word.in	word.out
Olympic	No Answer 0

【输入输出样例 2 解释】

单词 olympic 中出现最多的字母 i 出现了 2 次，出现次数最少的字母出现了 1 次， $2-1=1$ ，1 不是质数。

2. 火柴棒等式(matches.pas/c/cpp)

【问题描述】

给你 n 根火柴棍，你可以拼出多少个形如 “ $A+B=C$ ” 的等式？等式中的 A 、 B 、 C 是用火柴棍拼出的整数（若该数非零，则最高位不能是 0）。用火柴棍拼数字 0-9 的拼法如图所示：



注意：

- 1. 加号与等号各自需要两根火柴棍
- 2. 如果 $A \neq B$ ，则 $A+B=C$ 与 $B+A=C$ 视为不同的等式（ $A、B、C \geq 0$ ）
- 3. n 根火柴棍必须全部用上

【输入】

输入文件 matches.in 共一行，又一个整数 n ($n \leq 24$)。

【输出】

输出文件 matches.out 共一行，表示能拼成的不同等式的数目。

【输入输出样例 1】

matches.in	matches.out
14	2

【输入输出样例 1 解释】 2 个等式为 $0+1=1$ 和 $1+0=1$ 。

【输入输出样例 2】

matches.in	matches.out
18	9

【输入输出样例 2 解释】

9 个等式为：

$0+4=4$

$0+11=11$

$1+10=11$

$2+2=4$

$2+7=9$

$4+0=4$

$7+2=9$

$10+1=11$

$11+0=11$

3. 传纸条(message.pas/c/cpp)

【问题描述】

小渊和小轩是好朋友也是同班同学，他们在一起总有谈不完的话题。一次素质拓展活动中，班上同学安排做成一个 m 行 n 列的矩阵，而小渊和小轩被安排在矩阵对角线的两端，因此，他们就无法直接交谈了。幸运的是，他们可以通过传纸条来进行交流。纸条要经由许多同学传到对方手里，小渊坐在矩阵的左上角，坐标(1,1)，小轩坐在矩阵的右下角，坐标(m,n)。从小渊传到小轩的纸条只可以向下或者向右传递，从小轩传给小渊的纸条只可以向上或者向左传递。

在活动进行中，小渊希望给小轩传递一张纸条，同时希望小轩给他回复。班里每个同学都可以帮他们传递，但只会帮他们一次，也就是说如果此人在小渊递给小轩纸条的时候帮忙，那么在小轩递给小渊的时候就不会再帮忙。反之亦然。

还有一件事情需要注意，全班每个同学愿意帮忙的好感度有高有低（注意：小渊和小轩的好心程度没有定义，输入时用 0 表示），可以用一个 0-100 的自然数来表示，数越大表示越好心。小渊和小轩希望尽可能找好心程度高的同学来帮忙传纸条，即找到来回两条传递路径，使得这两条路径上同学的好心程度只和最大。现在，请你帮助小渊和小轩找到这样的两条路径。

【输入】

输入文件 message.in 的第一行有 2 个用空格隔开的整数 m 和 n ，表示班里有 m 行 n 列（ $1 \leq m, n \leq 50$ ）。

接下来的 m 行是一个 $m*n$ 的矩阵，矩阵中第 i 行 j 列的整数表示坐在第 i 行 j 列的学生的`好心程度`。每行的 n 个整数之间用空格隔开。

【输出】

输出文件 message.out 共一行，包含一个整数，表示来回两条路上参与传递纸条的学生的`好心程度之和`的最大值。

【输入输出样例】

message.in	message.out
3 3 0 3 9 2 8 5 5 7 0	34

【限制】

30%的数据满足： $1 \leq m, n \leq 10$

100%的数据满足： $1 \leq m, n \leq 50$

4. 双栈排序(twostack.pas/c/cpp)

【问题描述】

Tom 最近在研究一个有趣的排序问题。如图所示，通过 2 个栈 S1 和 S2，Tom 希望借助以下 4 种操作实现将输入序列升序排序。

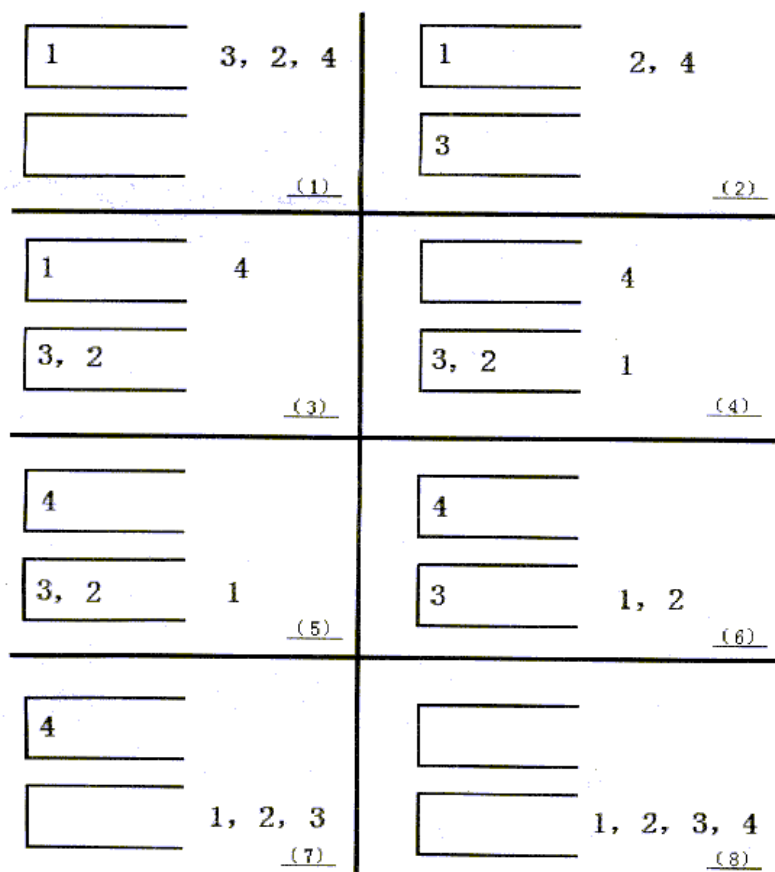
操作 a: 如果输入序列不为空, 将第一个元素压入栈 S1

操作 b:如果栈 S1 不为空, 将 S1 栈顶元素弹出至输出序列

操作 c:如果输入序列不为空, 将第一个元素压入栈 S2

操作 d:如果栈 S2 不为空, 将 S2 栈顶元素弹出至输出序列

如果一个 $1 \sim n$ 的排列 P 可以通过一系列操作使得输出序列为 $1, 2, \dots, (n-1), n$, Tom 就称 P 是一个“可双栈排序排列”。例如 $(1, 3, 2, 4)$ 就是一个“可双栈排序序列”，而 $(2, 3, 4, 1)$ 不是。下图描述了一个将 $(1, 3, 2, 4)$ 排序的操作序列: $\langle a, c, c, b, a, d, d, b \rangle$



当然，这样的操作序列有可能有几个，对于上例(1,3,2,4)，<a,c,c,b,a,d,d,b>是另外一个可行的操作序列。Tom 希望知道其中字典序最小的操作序列是什么。

【输入】

输入文件 `twostack.in` 的第一行是一个整数 n 。

第二行有 n 个用空格隔开的正整数，构成一个 $1\sim n$ 的排列。

【输出】

输出文件 `twostack.out` 共一行，如果输入的排列不是“可双栈排序排列”，输出数字 0；否则输出字典序最小的操作序列，每两个操作之间用空格隔开，行尾没有空格。

【输入输出样例 1】

twostack.in	twostack.out
4 1 3 2 4	a b a a b b a b

【输入输出样例 2】

twostack.in	twostack.out
4 2 3 4 1	0

【输入输出样例 3】

twostack.in	twostack.out
3 2 3 1	a c a b b d

【限制】

30%的数据满足： $n \leq 10$

50%的数据满足： $n \leq 50$

100%的数据满足： $n \leq 1000$