

# 本章主要考枚举搜索

## 译题 - milk2

### 描述

三个农民每天清晨 5 点起床，然后去牛棚给 3 头牛挤奶。第一个农民在 300 时刻(从 5 点开始计时，秒为单位)给他的牛挤奶，一直到 1000 时刻。第二个农民在 700 时刻开始，在 1200 时刻结束。第三个农民在 1500 时刻开始 2100 时刻结束。期间最长的至少有一个农民在挤奶的连续时间为 900 秒(从 300 时刻到 1200 时刻)，而最长的无人挤奶的连续时间(从挤奶开始一直到挤奶结束)为 300 时刻(从 1200 时刻到 1500 时刻)。

你的任务是编一个程序，读入一个有  $N$  个农民( $1 \leq N \leq 5000$ )挤  $N$  头牛的工作时间列表，计算以下两点(均以秒为单位)：

- 最长至少有一人在挤奶的时间段。
- 最长的无人挤奶的时间段。

### 格式

**PROGRAM NAME:** milk2

**INPUT FORMAT:**

(file milk2.in)

Line 1:

一个整数  $N$ 。

Lines 2.. $N$ +1:

每行两个小于 1000000 的非负整数，表示一个农民的开始时刻与结束时刻。

**OUTPUT FORMAT:**

(file milk2.out)

一行，两个整数，即题目所要求的两个答案。

### SAMPLE INPUT

3  
300 1000  
700 1200  
1500 2100

## SAMPLE OUTPUT

900 300

## 解体报告：

有两种思想

## 离散化

按照开始时间升序排序，然后从左到右扫一遍，复杂度是  $O(n\log n+n)$  的。

所谓从左到右扫一遍，就是记录一个当前区间，

如果下一个区间是当前区间的子区间，跳过。

如果下一个区间和当前区间相交，就合并两个区间放入当前区间，

否则就将新区间作为当前区间，

然后继续往右扫。（参考程序为离散化法）

## 线段树

本题的规模是  $1e6$ ，简单的模拟是  $O(nm)$  ( $n$  是奶牛个数， $m$  是最大范围) 的，会超时。（但是本题数据远没有描述的那么恐怖，直接模拟也是很快的）

用线段树统计区间，复杂度降为  $O(n\log m+m)$ ，可以接受。

## 译题 - transform

## 描述

一块  $N \times N$  ( $1 \leq N \leq 10$ ) 正方形的黑白瓦片的图案要被转换成新的正方形图案。写一个程序来找出将原始图案按照以下列转换方法转换成新图案的最小方式：

- 1: 转 90 度: 图案按顺时针转 90 度。
- 2: 转 180 度: 图案按顺时针转 180 度。
- 3: 转 270 度: 图案按顺时针转 270 度。
- 4: 反射: 图案在水平方向翻转(以中央铅垂线为中心形成原图案的镜像)。
- 5: 组合: 图案在水平方向翻转, 然后按照 1-3 之一转换。
- 6: 不改变: 原图案不改变。
- 7: 无效转换: 无法用以上方法得到新图案。
- 如果有多种可用的转换方法, 请选择序号最小的那个。

## 格式

**PROGRAM NAME:** transform

**INPUT FORMAT:**

(file transform.in)

第一行: 单独的一个整数 N。

第二行到第 N+1 行: N 行每行 N 个字符(不是 “@” 就是 “-”); 这是转换前的正方形。

第 N+2 行到第 2\*N+1 行: N 行每行 N 个字符(不是 “@” 就是 “-”); 这是转换后的正方形。

**OUTPUT FORMAT:**

(file transform.out)

单独的一行包括 1 到 7 之间的一个数字(在上文已描述)表明需要将转换前的正方形变为转换后的正方形的转换方法。

## SAMPLE INPUT

```
3
@-@
---
@@-
@-@
@--
--@
```

## SAMPLE OUTPUT

## 解题报告:

直接枚举,除了翻转不需要交换元素.

## 译题 - namenum

### 描述

在威斯康辛州奶牛牧场经营者之中,都习惯于请会计部门用连续数字给母牛打上烙印。但是,母牛用手机时并没感到这个系统的便利,它们更喜欢用它们喜欢的名字来呼叫它们的同伴,而不是用像这个的语句“C’mon, #4734, get along.”。

请写一个程序来帮助可怜的牧牛工将一只母牛的烙印编号翻译成一个可能的名字。因为母牛们现在都有手机了,使用那标准的按键的排布来把收到从数目翻译成文字 (除了“Q”和“Z”之外)

2: A, B, C	5: J, K, L	8: T, U, V
3: D, E, F	6: M, N, O	9: W, X, Y
4: G, H, I	7: P, R, S	

可接受的名字都被放在这样一个叫作“dict.txt”的文件中,它包含一连串的少于 5,000 个可接受的牛名字。(所有的名字都是大写的)收到母牛的编号返回那些能从编号翻译出来并且在字典中的名字(并且已经按照升序排列)。

举例来说,编号 4734 能产生的下列各项名字:

```

GPDG GPDH GPGI GPEG GPEH GPEI GPGF GPFH GPGI GRDG GRDH GRDI
GREG GREH GREI GRFG GRFH GRFI GSDG GSDH GSDI GSEG GSEH GSEI
GSFG GSFH GSFI HPDG HPDH HPDI HPEG HPEH HPEI HPFG HPFH HPFI
HRDG HRDH HRDI HREG HREH HREI HRFH HRFH HRFI HSDG HSDH HSDI
HSEG HSEH HSEI HSFG HSFH HSFI IPDG IPDH IPDI IPEG IPEH IPEI
IPFG IPFH IPFI IRDG IRDH IRDI IREG IREH IREI IRFG IRFH IRFI
ISDG ISDH ISDI ISEG ISEH ISEI ISFG ISFH ISFI

```

碰巧,81 个中只有一个“GREG”是有效的(在字典中)。

写一个程序来对给出的编号打印出所有有效名字,如果没有则输出 NONE。编号可能有 12 位数字。

### 格式

PROGRAM NAME: namenum

INPUT FORMAT:

(file namenum.in)

单独的一行包含一个编号(长度可能从 1 到 12)。

OUTPUT FORMAT:

(file namenum.out)

以字典顺序输出一个有效名字的不负列表，一行一个名字。

## SAMPLE INPUT

4734

## SAMPLE OUTPUT

GREG

## 解题报告：

一个数字对应 3 个字母，如果我们先枚举出数字所代表的所有字符串，就有  $3^{12}$  种，然后再在 5000 的字典里寻找，可以用二分查找，数据规模是  $3^{12} \times \log 5000 = 6.5e6$ ，空间规模是 5000。

其实可以做的更好！上面是顺序思想,下面是逆向思维.

一个字母只对应一个数字，从字典中读入一个单词，把它转化成唯一对应的数字，看它是否与给出的数字匹配，时间规模是  $5000 \times 12 = 6e4$ ，空间规模是常数，而且编程复杂度较低

还可以先比较字符长度和数字长度，如果相等，逐位比较。

## 译题 - palsquare

### 描述

回文数是指从左向右念和从右像做念都一样的数。如 12321 就是一个典型的回文数。

给定一个进制  $B$  ( $2 \leq B \leq 20$  十进制), 输出所有的大于等于 1 小于等于 300 (十进制下) 且它的平方用  $B$  进制表示时是回文数的数。用 'A', 'B' ..... 表示 10, 11 等等。

## 格式

PROGRAM NAME: palsquare

INPUT FORMAT:

(file palsquare.in)

共一行, 一个单独的整数  $B$  ( $B$  用十进制表示)。

OUTPUT FORMAT:

(file palsquare.out)

每行两个数字, 第二个数是第一个数的平方, 且第二个数是回文数。(注意: 这两个数都应该在  $B$  那个进制下)

## SAMPLE INPUT

10

## SAMPLE OUTPUT

```
1 1
2 4
3 9
11 121
22 484
26 676
101 10201
111 12321
121 14641
202 40804
212 44944
264 69696
```

## 解题报告:

进制转换与编程技巧.

# 译题 - dualpal

Dual Palindromes 双重回文数

## 描述

如果一个数从左往右读和从右往左读都是一样，那么这个数就叫做“回文数”。例如，12321 就是一个回文数，而 77778 就不是。当然，回文数的首和尾都应是非零的，因此 0220 就不是回文数。

事实上，有一些数（如 21），在十进制时不是回文数，但在其它进制（如二进制时为 10101）时就是回文数。

编一个程序，从文件读入两个十进制数  $N$  ( $1 \leq N \leq 15$ )  $S$  ( $0 < S < 10000$ ) 然后找出前  $N$  个满足大于  $S$  且在两种或两种以上进制(二进制至十进制)上是回文数的十进制数，输出到文件上。

本问题的解决方案不需要使用大于 4 字节的整型变量。

## 格式

PROGRAM NAME: dualpal

INPUT FORMAT:

(file dualpal.in)

只有一行，用空格隔开的两个数  $N$  和  $S$ 。

OUTPUT FORMAT:

(file dualpal.out)

$N$  行，每行一个满足上述要求的数，并按从小到大的顺序输出。

## SAMPLE INPUT

3 25

## SAMPLE OUTPUT

26

27

28

**解题报告:**

同上