

## 精英营 day5 习题集

### 1 基础练习题

#### 1.1 与 7 无关的数 (7.cpp)

##### 【问题描述】

一个正整数，如果它能被 7 整除，或者它的某一位上的数字为 7，则称其为“与 7 相关”的数。

现在，请编程求出所有小于或等于  $n$  的“与 7 无关”的正整数个数。

##### 【输入格式】

一行一个正整数  $n$ ， $n \leq 10^6$ 。

##### 【输出格式】

一行一个整数，表示答案。

##### 【样例输入】

21

##### 【样例输出】

17

#### 1.2 数字三角形 (numTriangle.cpp)

##### 【问题描述】

输入一个正整数  $n$ ，输出  $n$  行的数字三角形。其中，第 1 行为数字 1，第 2 行为数字 23，第 3 行为数字 456，第 4 行为数字 7890，第 5 行为数字 12345，...

##### 【输入格式】

一行一个正整数  $n$ ， $1 \leq n \leq 100$ 。

##### 【输出格式】

$n$  行的数字三角形。

##### 【样例输入】

4

##### 【样例输出】

1

23

456

7890

#### 1.3 回型方阵 (hxfz.cpp)

##### 【问题描述】

输入一个正整数  $n$ ，输出  $n \times n$  的回型方阵。例如， $n=5$  时，输出：

1 1 1 1 1

1 2 2 2 1

1 2 3 2 1

1 2 2 2 1

1 1 1 1 1

##### 【输入格式】

一行一个正整数  $n$ ， $2 \leq n \leq 9$ 。

##### 【输出格式】

共  $n$  行，每行包含  $n$  个正整数，之间用一个空格隔开。

#### 1.4 津津的储蓄计划 (save.cpp)

##### 【问题描述】

津津的零花钱一直都是自己管理。每个月的月初妈妈给津津 300 元钱，津津会预算这个月的花销，并且总能做到实际花销和

预算的相同。

为了让津津学习如何储蓄，妈妈提出，津津可以随时把整百的钱存在她那里，到了年末她会加上 20% 还给津津。因此津津制定了一个储蓄计划：每个月的月初，在得到妈妈给的零花钱后，如果她预计到这个月的月末手中还会有多于 100 元或恰好 100 元，她就会把整百的钱存在妈妈那里，剩余的钱留在自己手中。

例如 11 月初津津手中还有 83 元，妈妈给了津津 300 元。津津预计 11 月的花销是 180 元，那么她就会在妈妈那里存 200 元，自己留下 183 元。到了 11 月月末，津津手中会剩下 3 元钱。

津津发现这个储蓄计划的主要风险是，存在妈妈那里的钱在年末之前不能取出。有可能在某个月的月初，津津手中的钱加上这个月妈妈给的钱，不够这个月的原定预算。如果出现这种情况，津津将不得不在这个月省吃俭用，压缩预算。

现在请你根据 2004 年 1 月到 12 月每个月津津的预算，判断会不会出现这种情况。如果不会，计算到 2004 年年末，妈妈将津津平常存的钱加上 20% 还给津津之后，津津手中会有多少钱。

【输入文件】

输入文件 `save.in` 包括 12 行数据，每行包含一个小于 350 的非负整数，分别表示 1 月到 12 月津津的预算。

【输出文件】

输出文件 `save.out` 包括一行，这一行只包含一个整数。如果储蓄计划实施过程中出现某个月钱不够用的情况，输出 `-X`，`X` 表示出现这种情况的第一个月；否则输出到 2004 年年末津津手中会有多少钱。

【样例输入 1】

```
290
230
280
200
300
170
340
50
90
80
200
60
```

【样例输出 1】

```
-7
```

【样例输入 2】

```
290
230
280
200
300
170
330
50
90
80
200
60
```

【样例输出 2】

```
1580
```

## 2 T2 真题（每题 40-60 分钟）

### 2.1 NOIP2011 统计单词数 (`stat.cpp`)

【问题描述】

一般的文本编辑器都有查找单词的功能，该功能可以快速定位特定单词在文章中的位置，有的还能统计出特定单词在文章中出现的次数。

现在，请你编程实现这一功能，具体要求是：给定一个单词，请你输出它在给定的文章中出现的次数和第一次出现的位置。注意：匹配单词时，不区分大小写，但要求完全匹配，即给定单词必须与文章中的某一独立单词在不区分大小写的情况下完全相同（参见样例1），如果给定单词仅是文章中某一单词的一部分则不算匹配（参见样例2）。

**【输入格式】**

共 2 行。第 1 行为一个字符串，其中只含字母，表示给定单词；第 2 行为一个字符串，其中只可能包含字母和空格，表示给定的文章。

**【输出格式】**

一行，如果在文章中找到给定单词则输出两个整数，两个整数之间用一个空格隔开，分别是单词在文章中出现的次数和第一次出现的位置（即在文章中第一次出现时，单词首字母在文章中的位置，位置从 0 开始）；如果单词在文章中没有出现，则直接输出一个整数 -1。

**【输入输出样例】**

输入样例 1:

```
To
to be or not to be is a question
```

输出样例 1:

```
2 0
```

输入样例 2:

```
to
Did the Ottoman Empire lose its power at that time
```

输出样例

```
-1
```

**【数据说明】**

$1 \leq \text{单词长度} \leq 10$ 。

$1 \leq \text{文章长度} \leq 1,000,000$ 。

## 2.2 NOIP2010 接水问题 (water.cpp)

**【问题描述】**

学校里有一个水房，水房里一共装有  $m$  个龙头可供同学们打开水，每个龙头每秒钟的供水量相等，均为 1。现在有  $n$  名同学准备接水，他们的初始接水顺序已经确定。将这些同学按接水顺序从 1 到  $n$  编号， $i$  号同学的接水量为  $w_i$ 。接水开始时，1 到  $m$  号同学各占一个水龙头，并同时打开水龙头接水。当其中某名同学  $j$  完成其接水量要求  $w_j$  后，下一名排队等候接水的同学  $k$  马上接替  $j$  同学的位置开始接水。这个换人的过程是瞬间完成的，且没有任何水的浪费。即  $j$  同学第  $x$  秒结束时完成接水，则  $k$  同学第  $x+1$  秒立刻开始接水。若当前接水人数  $n$  不足  $m$ ，则只有  $n$  个龙头供水，其它  $m-n$  个龙头关闭。现在给出  $n$  名同学的接水量，按照上述接水规则，问所有同学都接完水需要多少秒。

**【输入格式】**

第 1 行 2 个整数  $n$  和  $m$ ，用一个空格隔开，分别表示接水人数和龙头个数。

第 2 行  $n$  个整数  $w_1、w_2、\dots、w_n$ ，每两个整数之间用一个空格隔开， $w_i$  表示  $i$  号同学的接水量。

**【输出格式】**

输出只有一行，1 个整数，表示接水所需的总时间。

**【输入样例】**

```
5 3
4 4 1 2 1
8 4
23 71 87 32 70 93 80 76
```

**【输出样例】**

```
4
163
```

**【数据说明】**

### 【输入输出样例 1 说明】

第 1 秒, 3 人接水。第 1 秒结束时, 1、2、3 号同学每人的已接水量为 1, 3 号同学接完水, 4 号同学接替 3 号同学开始接水。

第 2 秒, 3 人接水。第 2 秒结束时, 1、2 号同学每人的已接水量为 2, 4 号同学的已接水量为 1。第 3 秒, 3 人接水。

第 3 秒结束时, 1、2 号同学每人的已接水量为 3, 4 号同学的已接水量为 2。4 号同学接完水, 5 号同学接替 4 号同学开始接水。第 4 秒, 3 人接水。第 4 秒结束时, 1、2 号同学每人的已接水量为 4, 5 号同学的已接水量为 1。1、2、5 号同学接完水, 即所有人完成接水。总接水时间为 4 秒。

$1 \leq n \leq 10000$ ,  $1 \leq m \leq 100$  且  $m \leq n$ ;

$1 \leq w_i \leq 100$ 。

## 2.3 NOIP2009 分数线划定 (sorce.cpp)

### 【问题描述】

世博会志愿者的选拔工作正在 A 市如火如荼的进行。为了选拔最合适的人才, A 市对所有报名的选手进行了笔试, 笔试分数达到面试分数线的选手方可进入面试。面试分数线根据计划录取人数的 150% 划定, 即如果计划录取  $m$  名志愿者, 则面试分数线为排名第  $m * 150\%$  (向下取整) 名的选手的分数, 而最终进入面试的选手为笔试成绩不低于面试分数线的所有选手。现在就请你编写程序划定面试分数线, 并输出所有进入面试的选手的报名号和笔试成绩。

### 【输入格式】

第一行, 两个整数  $n, m$  ( $5 \leq n \leq 5000$ ,  $3 \leq m \leq n$ ), 中间用一个空格隔开, 其中  $n$  表示报名参加笔试的选手总数,  $m$  表示计划录取的志愿者人数。输入数据保证  $m * 150\%$  向下取整后小于等于  $n$ 。第二行到第  $n+1$  行, 每行包括两个整数, 中间用一个空格隔开, 分别是选手的报名号  $k$  ( $1000 \leq k \leq 9999$ ) 和该选手的笔试成绩  $s$  ( $1 \leq s \leq 100$ )。数据保证选手的报名号各不相同。

### 【输出格式】

第一行, 有两个整数, 用一个空格隔开, 第一个整数表示面试分数线; 第二个整数为进入面试的选手的实际人数。从第二行开始, 每行包含两个整数, 中间用一个空格隔开, 分别表示进入面试的选手的报名号和笔试成绩, 按照笔试成绩从高到低输出, 如果成绩相同, 则按报名号由小到大的顺序输出。

### 【输入样例】

```
6 3
1000 90
3239 88
2390 95
7231 84
1005 95
1001 88
```

### 【输出样例】

```
88 5
1005 95
2390 95
1000 90
1001 88
3239 88
```

### 【数据说明】

$m * 150\% = 3 * 150\% = 4.5$ , 向下取整后为 4。保证 4 个人进入面试的分数线为 88, 但因为 88 有重分, 所以所有成绩大于等于 88 的选手都可以进入面试, 故最终有 5 个人进入面试。

## 2.4 NOIP2008 排座椅 (seat.cpp)

### 【问题描述】

上课的时候总有一些同学和前后左右的人交头接耳, 这是令小学班主任十分头疼的一件事情。不过, 班主任小雪发现了一些有趣的现象, 当同学们的座次确定下来之后, 只有有限的  $D$  对同学上课时 would 交头接耳。同学们在教室中坐成了  $M$  行  $N$  列, 坐在第  $i$  行第  $j$  列的同学的位置是  $(i, j)$ , 为了方便同学们进出, 在教室中设置了  $K$  条横向的通道,  $L$  条纵向的通道。于是, 聪明的小雪想到了一个办法, 或许可以减少上课时学生交头接耳的问题: 她打算重新摆放桌椅, 改变同学们桌椅间通道的位置, 因为如果

一条通道隔开了两个会交头接耳的同学,那么他们就不会交头接耳了。请你 帮忙给小雪编写一个程序,给出最好的通道划分方案。在该方案下,上课时交头接耳的学生对数最少。

**【输入格式】**

第一行,有 5 各用空格隔开的整数,分别是  $M, N, K, L, D$  ( $2 \leq N, M \leq 1000, 0 \leq K < M, 0 \leq L < N, D \leq 2000$ )。接下来  $D$  行,每行有 4 个用空格隔开的整数,第  $i$  行的 4 个整数  $X_i, Y_i, P_i, Q_i$ , 表示坐在位置  $(X_i, Y_i)$  与  $(P_i, Q_i)$  的两个同学会交头接耳 (输入保证他们前后相邻或者左右相邻)。输入数据保证最优方案的唯一性。输入数据保证最优方案的唯一性。

**【输出格式】**

共两行。

第一行包含  $K$  个整数,  $a_1 a_2 \dots a_K$ , 表示第  $a_1$  行和第  $a_1+1$  行之间、第  $a_2$  行和第  $a_2+1$  行之间、 $\dots$ 、第  $a_K$  行和第  $a_K+1$  行之间要开辟通道, 其中  $a_i < a_{i+1}$ , 每两个整数之间用空格隔开 (行尾没有空格)。

第二行包含  $L$  个整数,  $b_1 b_2 \dots b_L$ , 表示第  $b_1$  列和第  $b_1+1$  列之间、第  $b_2$  列和第  $b_2+1$  列之间、 $\dots$ 、第  $b_L$  列和第  $b_L+1$  列之间要开辟通道, 其中  $b_i < b_{i+1}$ , 每两个整数之间用空格隔开 (行尾没有空格)。

**【输入样例】**

```
4 5 1 2 3
4 2 4 3
2 3 3 3
2 5 2 4
```

**【输出样例】**

```
2
2 4
```