

# 2012 年广东省青少年信息学竞赛（GDOI2012）

## 第二试

2012 年 4 月 30 日

注意事项：

1. 严格按照题目所要求的格式进行输入、输出，否则严重影响得分。
2. 题目测试数据有严格的时间限制，超时不得分。
3. 输入文件格式不用判错；输入输出文件名均已给定，不用键盘输入。
4. 源程序保存在以“考号+姓名”命名的文件夹下，按题目要求命名。评测以源程序为准。
5. 四个小时三十分完成。

试题名称	堆积木	mmm 学写字	东方非想天则	弹珠游戏	AB 棋
提交文件名	stack.pas/c/cpp	mmm.pas/c/cpp	daze.pas/c/cpp	game.pas/c/cpp	chess.pas/c/cpp
输入文件名	stack.in	mmm.in	daze.in	game.in	chess.in
输出文件名	stack.out	mmm.out	daze.out	game.out	chess.out
满分	100	100	100	100	100

## 堆积木

(stack.pas/c/cpp)

### 【问题描述】

mmm 很喜欢玩积木，现在她手上有  $a\sim z$  这 26 种积木，因为她智商很低所以只会两种堆积木方法，具体如下：

- 1、往最顶放一块积木
- 2、从积木堆中抽出一块积木，其他在这块积木上面的积木会依次掉下来但不会倒

mmm 对她第 2 招神技感到很自豪，但众人都知道她的智商是很低的，随便问一下她一些问题她就不会了，所以大家设计了如下两个问题：

- 1、从下往上数第  $k$  块积木是什么颜色？
- 2、从下往上数第  $k$  块为  $color$  颜色的积木在什么位置(从下往上数)？

最后还是那句话，mmm 的智商是很低的，她只能求助于你了。

### 【输入格式】

第一行一个整数  $n$ ，表示堆积木和询问操作的总和。一开始积木堆是空的。

下面  $n$  行，每行为如下其中一种操作。(不包托双引号)；

"p c" 将一块颜色为  $c$  的积木放到最顶端

"r k" 抽出从下往上数的第  $k$  块积木

"q k c" 询问从下往上数第  $k$  块为  $c$  颜色的积木在什么位置(从下往上数)

"w k" 询问从下往上数的第  $k$  块积木是什么颜色

输入数据保证每次询问必有结果.

**【输出格式】**

对每个 q 或 w 操作,输出一行表示结果.

**【输入样例】**

```
6
p a
p b
q 1 b
r 1
q 1 b
w 1
```

**【输出样例】**

```
2
1
b
```

**【数据范围】**

对于 60%的数据  $0 < n \leq 10000$ , r 和 q 的操作数不超过 5000

对于 100%的数据  $0 < n \leq 100000$

## mmm 学写字

(mmm. pas/c/cpp)

**【问题描述】**

传闻中 mmm 的智商是很低的,但事实是 mmm 的智商确实很低。

虽然 mmm 的智商很低,但 mmm 的字写的非常好,mmm 首创莫队体至今仍广为流传。

由于 mmm 智商很低,她只会在一张 4 行 n 列的格子纸上写字,这种规格的纸叫莫队纸。莫队体很简单,其实就在这张格子图上把某些格子涂黑。很简单对吧?所以说永远不要低估 mmm 的低智商!绝对无下限。纸上每一个的魅力值是不同的,每个格子都有一个特定的魅力值,且可能随着时间会发生改变,可正可负。老师给 mmm 布置了一个作业,需要在纸上写一个“工”字。mmm 马上拿出莫队纸,但老师还特别要求了,作业只能写在莫队纸的某个矩形区域内。假设 mmm 已经知道每个格子的魅力值,至于她是怎么知道,我当然不会告诉你魅力值写在纸背面。现在,mmm 为了证明给众人看她智商很高,所以她想要写一个魅力值总和大的工字,但是,由于.....她只能请教你了。

工字要求,上下两个横必须对齐,但不要求对称,且上下两横的厚度不限,也不必相等。中间竖线的宽度也不限。请参考以下例子以便更好地理解。

以下形状是合法“工”字的例子: 0 表示选, x 表示不选

```
00000    0000    X0000    0000X
X00XX    X0XX    XX00X    X00XX
X00XX    X0XX    XX00X    0000X
00000    X0XX    X0000    XXXXX
00000    0000    X0000
```

以下形状是不合法的“工”字的例子:

```
000X    0000    0000X    000X    000X    00000    0000
0000    X0XX    XX00X    XX00    X0XX    X0X0X    0000
X0XX    X00X    XX00X    XX00    X0XX    X0X0X    X00X
X0XX    0000    00000    000X    0000    00000    X00X
0000
(1)      (2)      (3)      (4)      (5)      (6)      (7)
```

- (1) 上面一横有“缺口”。
- (2) 中间一竖有“缺口”。
- (3) 上面的一横没有突出来, 与下面一横也没有对齐
- (4) 上下两横都写挫了, 右边不够长。
- (5) 上下两横没有对齐
- (6) 中间一竖有“气泡”?
- (7) 下面的一横居然消失了?!

现在, 假设你是 mmm 的好朋友, 需要你写一个程序来维护以下的信息:

操作 1: mmm 想询问某一个矩形区域内合法工字的最大魅力值是多少。(所占格子的魅力值总和)

具体格式为  $1\ x1\ y1\ x2\ y2$

表示在以  $[x1, y1]$  为左上角,  $[x2, y2]$  为右下角的区域。

操作 2: 某个格子的魅力值发生改变。

具体格式为  $2\ x\ y\ value$

表示  $[x, y]$  这个格子的魅力值改为  $value$ 。

### 【输入格式】

第一行一个整数  $t$ , 表示共有  $t$  组输入数据。

每组数据第一行一个整数  $n$ , 表示纸的总宽度。

接下来 4 行每行  $n$  个整数, 表示每个格子的初始魅力值。

第 6 行一个整数  $m$ , 表示操作的总数。

接下来  $m$  行, 每行表示一个操作, 格式如题目。

$1\ x1\ y1\ x2\ y2$  或  $2\ x\ y\ value$

### 【输出格式】

对于每个 1 操作, 输出一行表示可得到的最大魅力值如果魅力值为负数直接输出 0。

### 【输入样例】

```
1
6
1 -2 4 5 6 8
1 3 -10 20 12 9
-5 1 10 -1 8 4
-5 7 -8 4 8 11
1
1 1 2 4 5
```

**【输出样例】**

```
62
```

**【数据范围】**

30%的数据， $n \leq 100, m \leq 100$

50%的数据， $n \leq 1000, m \leq 1000$ , 询问的矩形最大宽度  $w \leq 50$ 。

100%的数据， $n \leq 10000, m \leq 20000$ , 魅力值  $|value| \leq 10^9$ 。

## 东方非想天则

(daze.pas/c/cpp)

**【问题描述】**

JanD 很喜欢格斗游戏，最近他又玩起了一个格斗游戏，叫做“东方非想天则”。因为他连段很差，所以经常被网上的触手们误伤。于是他开始研究怎么才能提高他连段的伤害。

东方非想天则的每个人物招数都很多，其中灵乌路空是 janD 最喜欢的角色。于是他把灵乌路空的出招表下载了下来，仔细观察后发现每个技能都有三个属性：

Di：伤害值，使出此招后能造成的伤害

Ri：修正值，对此招以后的所有伤害进行修正

Li：极限值，保护玩家不受过大的伤害

一个连段是由许多技能组成的，所有的技能可以以任何的组合顺序形成一个连段，但每个技能在一个连段中最多出现一次。初始状态下我方的修正值 RATE 值为 100%，而极限值 Limit 值为 0%，一个技能击中对手时造成的伤害为该技能的伤害值 Di 乘以现在的 Rate 值，然后 Rate 值将会乘以该技能的修正值 Ri，Limit 值将会加上该技能的极限值 Li，当 Limit 值大于等于 100%时，对方会进入保护状态而不再受到我方的任何攻击。

现在，拿到出招表的 janD 希望知道最大的连段伤害是多少，你能够帮助他么？

**【输入格式】**

数据第 1 行为技能的数目 N。

接下来 N 行，每行共有三个数，分别是 Di, Ri, Li，其中为第 i 招的伤害值，Ri 为第 i 招的修正值\*100%, Li 为第 i 招的极限值\*100%

为了方便，Ri 与 Li 都省略了百分号

**【输出格式】**

输出共一行，输出一个连段最大的伤害总和（四舍五入到小数点后两位）。

**【输入样例 1】**

```
3
1000 99.70 20
2300 50.00 70
500 99.00 50
```

**【输出样例 1】**

```
3768.67
```

**【输入样例 2】**

```
3
3000 1.0 80
200 90.00 70
100 1.00 40
```

**【输出样例 2】**

```
3002.00
```

**【数据范围】**

$D_i$  是整数且  $1 \leq D_i \leq 5000$

$R_i$  是浮点数,其小数点后不超过两位, 且  $0 \leq R_i \leq 100$ 。

$L_i$  是偶数且  $1 \leq L_i \leq 100$

$1 \leq N \leq 10000$

**【样例说明】**

第 1 组:

首先使用第 1 个技能,造成的伤害为 1000,此时 Rate 值变为 99.7%, Limit 值变为 20%。

然后使用第 3 个技能,造成的伤害为  $500 * 99.7\% = 498.5$ ,此时 Rate 值变为 98.703%, Limit 值 90%。

然后使用第 2 个技能,造成的伤害为  $2300 * 98.703\% = 2270.169$ ,此时 Rate 值变为 49.3515%, Limit 值变为 140%。

因此该连段伤害总和为 3768.669.

第 2 组:

首先使用第 1 个技能,造成的伤害为 3000,此时 Rate 值变为 1.0%, Limit 值变为 80%。

然后使用第 3 个技能,造成的伤害为  $200 * 1.0\% = 2$ ,此时 Rate 值变为 0.9%, Limit 值 150%。

此时已经不能再对方进行任何攻击了。

因此该连段伤害总和为 3002.00.

# 弹珠游戏

(game.pas/c/cpp)

## 【问题描述】

mmm 很幼稚，她喜欢玩弹珠游戏，而且百玩不厌。弹珠游戏的棋盘是一个  $N \times N$  得棋盘，每个格子都是正方形，其中的  $M$  个不同的格子有洞，游戏中玩家需要用最少的步数把  $M$  个弹珠（编号  $1 \sim M$ ）移动到各自的洞里。当然为了使游戏更有趣，棋盘上某些格子之间有一些障碍。

游戏刚开始，所有弹珠摆在不同位置上，每一步 mmm 可以选择把整个棋盘的某个边抬起来（mmm 力气很大），随着重力的作用，弹珠会往相反的方向滚动，直到碰到一面墙（障碍或棋盘边界）或者其他的弹珠，或者掉到洞里，弹珠才会停止滚动。

弹珠滚动的时候需要满足以下条件：

- 1，弹珠不能跳过障碍，不能跳过其他弹珠，不能跳过洞。
- 2，弹珠不能滚到棋盘外，假设棋盘的周围是一圈障碍。
- 3，任意时刻每一个格子只能有一个弹珠。
- 4，当某个弹珠掉到了相应的洞里，那么它就填满了这个格子，之后的移动中其他弹珠可以从它之上滚过去。

当所有的弹珠都掉到了与之对应的格子，那么游戏结束。

给你一个棋盘，以及各个弹珠各个洞的位置，求所有弹珠都掉到相应的洞里的最小步数。无解输出-1。

因为 mmm 很 2，所以每组数据中有 2 个游戏，你需要两个游戏都做对，才有这组数据的分数。

## 【输入格式】

第一行三个整数  $N, M, W$ ，表示棋盘的规模、弹珠的个数和障碍的个数。

接下来  $M$  行分别给出  $M$  个弹珠的位置  $(x, y)$ 。

再接下来的  $M$  行分别给出每个弹珠对应的洞的位置  $(x, y)$ 。

最后  $W$  行，每行四个整数坐标  $(x1, y1)$  和  $(x2, y2)$  表示这两个格子之间有障碍。数据保证两个格子是相邻的。

接下来是第二组数据的描述，跟以上一样。

## 【输出格式】

两个整数，即答案。每个一行。

## 【输入样例】

```
4 3 1
0 1
1 0
1 2
2 3
```

2 1  
3 2  
1 1 1 2  
3 2 2  
0 0  
0 1  
0 2  
2 0  
2 0 1 0  
2 0 2 1

**【输出样例】**

5  
-1

**【数据范围】**

20%的数据， $M=1$

100%的数据， $N \leq 4, 0 \leq N \leq 8$

## AB 棋

(chess.pas/c/cpp)

**【问题描述】**

AB 棋是两个人玩的博弈游戏。它的棋盘是一排  $n$  个格子，从左到右标号  $0..n-1$ 。初始时某些格子上面有一只棋子。

两个人轮流走，他们都有两种走法：

A.把一只棋子往左走一步。即把  $i$  上的棋子走到  $i-1$  上（必须满足  $i \geq 1$ ， $i$  上有棋子且  $i-1$  上没棋子）

B.把一只棋子跳过左面的棋子。即把  $i$  上的棋子跳过  $i-1$  到  $i-2$  上（必须满足  $i \geq 2$ ， $i$  和  $i-1$  上有棋子且  $i-2$  上没有棋子）

谁不能走谁就输了。

给定棋盘的初始状态，你要判断出，在两个人都用最佳策略的情况下，先手胜还是后手胜。

**【输入格式】**

输入文件包含多个棋盘状态，第一行只有一个整数代表棋盘状态的数量  $T$ 。

下面  $T$  行，每行一个 01 串，代表一个棋盘状态。0 代表空格，1 代表有棋子。

01 串的长度即为  $n$ 。其方向与棋盘一致，即 01 串的左边为棋盘的左边。

**【输出格式】**

输出文件应包含  $T$  行。

每行输出对应棋盘状态的胜方。先手胜输出 **first**，后手胜输出 **second**。

**【输入样例】**

```
8
0001
11111
01001
0101001
01001001001
0111
0001111
0111001111
```

**【输出样例】**

```
first
second
second
first
second
second
first
first
```

**【数据范围】**

有 30% 的数据， $T \leq 20$ ， $n \leq 20$

另外有 30% 的数据， $T \leq 20$ ， $n \leq 10000$ ，棋盘初始时没有紧挨着的两只棋子，即 ‘11’ 不是该棋盘对应 01 串的子串。

剩下的 40% 的数据， $T \leq 20$ ， $n \leq 10000$ 。