

CSP-J 复赛模拟

普及组

(请选手务必仔细阅读本页内容)

一. 题目概况

中文题目名称	按钮	参赛者	兴趣班	有趣的数学
英文题目与子目录名	class	Binary	cow	gift
可执行文件名	class	Binary	COW	gift
输入文件名	class.in	Binary.in	COW.in	gift.in
输出文件名	class.out	Binary.out	COW.out	gift.out
每个测试点时限	1 秒	1 秒	2 秒	1 秒
测试点数目	10	10	10	10
每个测试点分值	10	10	10	10
附加样例文件	有	有	有	有
结果比较方式	全文比较 (过滤行末 z 空格及文末回车)			
题目类型	传统	传统	传统	传统

二. 提交源程序文件名

对于 C++语言	class.cpp	Binary.cpp	cow.cpp	gift.cpp
----------	-----------	------------	---------	----------

三. 运行内存限制

内存上限	128M	128M	128M	128M
------	------	------	------	------

注意事项:

- 1、文件名 (程序名和输入输出文件名) 必须使用英文小写。
- 2、C/C++中函数 main () 的返回值类型必须是 int, 程序正常结束时的返回值必须是 0。
- 3、**特别提醒:** 评测在 NOI Linux 下进行

1 按钮 (class.cpp)

zz 在一家工厂工作。每天结束时，他必须关掉所有的灯。

工厂里的灯的索引从 1 到 n 。zz 的房间里也有 n 个按钮，索引从 1 到 n 。如果 zz 按下索引 i 的按钮，那么索引不小于 i 的灯都将关闭。

zz 不是很聪明。因此，他不是按第一个按钮，而是每天晚上随机按一些按钮。他按了 m 个不同的按钮 b_1, b_2, \dots, b_m (按给定顺序连续按下按钮)。现在他想知道对于每个灯，关闭这个灯的按钮的索引。请注意按钮 b_i 的索引实际上是 b_i ，而不是 i 。

求你了，帮帮 zz，打印这些索引。

输入的第一行包含两个以空格分隔的整数 n 和 m ($1 \leq n, m \leq 100$)，分别为工厂灯的个数和按下的按钮个数。下一行包含 m 个不同的用空格分隔的整数 b_1, b_2, \dots, b_m ($1 \leq b_i \leq n$)。

保证按下所有按钮后，所有的灯都会被关掉。

输出 n 个以空格分隔的整数，其中第 i 个数字是关闭第 i 个灯的按钮的索引。

class. in	class. out
5 4 4 3 1 2	1 1 3 4 4
5 5 5 4 3 2 1	1 2 3 4 5

说明：

在第一个示例中，在按下按钮 4 后，灯 4 和 5 被关闭，灯 1、2 和 3 仍然亮着。然后按下 3 号按钮，3 号灯也会熄灭。按下 1 号键也会关掉 1 号灯和 2 号灯，所以按下 2 号键最终没有效果。因此，按钮 4 将灯 4 和 5 关闭，按钮 3 将灯 3 关闭，按钮 1 将灯 1 和 2 关闭。

[数据范围]

n 和 m ($1 \leq n, m \leq 100$)

2 参赛者 (Binary.cpp)

比赛考试啦！在每一轮比赛开始时，参赛者被分在不同的房间里，每个房间正好包含 n 个参与者。在比赛中，参赛者被建议解决，A、B、C、D 和 E 五个问题。对于每个问题，参赛者会根据解决给定问题的时长以及是否彻底解决获得一些积分。此外，选手可以对其他选手进行黑客攻击。每一次成功的黑客攻击，选手将获得 100 分，每一次失败的黑客攻击，选手将损失 50 分。每个选手的得分是由他从所有问题(包括黑客)中获得的得分总和。

计算比赛中每个房间的最高得分。

输入格式

第一行包含一个整数 n ，它是房间中参赛者的数量 ($1 \leq n \leq 50$)。接下来的 n 行包含给定房间的参与者。

第 i 行格式为“*handlei plusi minusi ai bi ci di ei*”，分别是选手的名称、黑客成功的次数、黑客失败的次数以及他在 A、B、C、D、E 题中得到的分数。每个与会者的名称由拉丁字母、数字和下划线组成，长度为 $1 \sim 20$ 个字符。对于数量有以下限制：

$0 \leq \text{plusi}, \text{minusi} \leq 50$;

如果 A 问题没有解决，则 $150 \leq a_i \leq 500$ 或 $a_i = 0$;

如果 B 问题没有解决，则 $300 \leq b_i \leq 1000$ 或 $b_i = 0$;

如果 C 问题没有解决，则 $450 \leq c_i \leq 1500$ 或 $c_i = 0$;

如果 D 问题没有解决，则 $600 \leq d_i \leq 2000$ 或 $d_i = 0$;

如果 E 问题没有解决，则 $750 \leq e_i \leq 2500$ 或 $e_i = 0$ 。

所有的数字都是整数。所有的参与者都有不同的处理方式。保证房间里只有一个最大值(即没有两个得分最多的参与者)。

输出格式

输出为单行：

输出得分最高的参与者名字。

Binary.in	Binaryt.out
5 Petr 3 1 490 920 1000 1200 0 tourist 2 0 490 950 1100 1400 0 Egor 7 0 480 900 950 0 1000 c00lH4x0R 0 10 150 0 0 0 0 some_participant 2 1 450 720 900 0 0	tourist

说明:

样例数据每个人的得分如下:

```
Petr — 3860
tourist — 4140
Egor — 4030
c001H4x0R — -350
some_participant — 2220
```

最高分是 tourist.

3 兴趣班 (cow.cpp)

陈焯不但喜欢折纸，而且也喜欢踢足球和编程，因为没有合适的折纸班，所以他决定参加足球和编程兴趣班。

足球兴趣班共有 n 个时间段选择，第 i 个时间段排在 $(L_i \sim R_i)$ 。编程兴趣班也有 m 个时间段选择，第 i 个时间段排在 $(A_i \sim B_i)$ 。他必须要选择一个足球班和一个编程班的时间段，但他希望选的这两个班中间的休息时间越长越好。

例如，他选了这两个时间段 $(L_1 \sim R_1)$ 和 $(A_1 \sim B_1)$ ，假设 $(L_1 \sim R_1)$ 这节课在前面， $(A_1 \sim B_1)$ 这节课在后面，那么，陈焯在中间休息的时间是 $A_1 - R_1$ 。特别说明，当两节课上课时间有冲突，陈焯在中间休息时间为 0。

陈焯想算一算他所选的两节兴趣课之间，能休息的时间最长是多少？请你帮助陈焯找一找，算一算。

输入第一行输入一个整数 n ，表示足球兴趣班可选择的时间段。下列 n 行，每行都输入两个整数 L_i 和 R_i ，分别表示陈焯参加第 i 个足球班的起止时间。

下面一行输入的一个整数 m ，表示编程兴趣班可选择的时间段。

下列 m 行，每行都输入两个整数 A_i 和 B_i ，分别表示陈焯参加第 i 个编程班的起止时间。对于 100% 的数据，保证 $1 \leq n \leq 200000, 1 \leq m \leq 200000, 1 \leq L_i \leq R_i \leq 1000000000, 1 \leq A_i \leq B_i \leq 1000000000$

输出一个整数，表示两个时间段之间的最长休息时间（如果所有时间段都有冲突，则输出 0）。

cow.in	cow.out
3 1 5 2 6 2 3 2 2 4 6 8	3
3 1 5 2 6 3 7 2 2 4 1 4	0

提示

样例 1，陈焯可以在这段时间 $(2, 3)$ 参加围棋班，并在另一段时间 $(6, 8)$ 参加编程班。不难算出，在这种情况下，他中间休息的时间是最长的 $6 - 3 = 3$ 。

样例 2，他选择任何一段时间，两个兴趣班上课的时间都有冲突，所以答案是 0。

4 有趣的数学

(gift.cpp)

自从泽泽参加了围棋班和编程班以后，不仅思维非常敏捷，而且有些难懂的数学问题他都能轻松解答，他爱上了数学。老师为了锻炼他，让他的思维能有突飞猛进的进步，给了他这样一题数学题：给你一个正整数 N ，请你找出 N 是由哪些若干个(两个或多个)连续的正整数的和得到。

输入格式

第一行输入一个正整数 N 。

输出格式

输出至少有一个答案，每个答案一行，每行输出这个答案中的第一个加数和最后一个加数。如果有多个答案，这些答案必须按第一个加数从小到大进行排列。（注：两个加数之间只有一个空格）

gift.in	gift.out
6	1 3
27	2 7 8 10 13 14

【样例解释】

样例 1 说明：

$6=1+2+3$,

第一个加数是 1，最后一个加数是 3， 输出 1 3

样例 2 说明：

$27=2+3+4+5+6+7$,

第一个加数是 2，最后一个加数是 7， 输出 2 7

$27=8+9+10$,

第一个加数是 8，最后一个加数是 10， 输出 8 10

$27=13+14$,

第一个加数是 13，最后一个加数是 14，输出 13 14

因为 27 有三个答案，所以我们按第一个加数从小到大的顺序进行输出。

【数据范围约定】

对于 60%的数据保证 $3 \leq N \leq 1000000$

对于 80%的数据保证 $3 \leq N \leq 100000000$

对于 100%的数据保证 $3 \leq N \leq 10000000000$