

高中组 (参考范例)

第一题：小数加法

题目内容：温教授最近在上计算机概论，要教学生使用十进制以外的不同进位制度，例如用位就能表示的二进制、方便工程师阅读的十六进制等等。不过，最近的学生没有狼性，他怎么教学生都听不懂，温教授为此非常烦恼，于是走进附近的小 7 超商想买杯咖啡。

看着架上商品的价钱之际，教授忽然灵机一动，言教不如身教，学生如果没有从做中学，怎么有办法摆脱习惯的十进制使用其他的进位制呢？于是教授决定偷偷委托亲朋好友在校内开设一间新的「真·小 7 超商」，里面所有的价格都用七进制来表示，其中有些商品比一般的小 7 超商还贵，有些则便宜很多，这样子学生为了捡便宜就会尝试来店内挖宝。

不过，超商开幕后，温教授却发现连他自己都购物得很吃力。身为教授连自己都算错的话就太糗了，只好委托你帮他开发一个能够帮他做七进制加法的程序。因为商品的价格当然都是正的，所以价钱只会是正的整数或小数。顺带一提，在小 7 超商内结账所的货币当然也是采用七进制的，因此可以不必担心如何付款的问题。

输入说明：输入共有两列，每一列包含一个七进制下的数值。输入中每一个七进制下的数值可能为七进制正整数或是大于零的七进制小数，格式如下：若是七进制正整数，则以由阿拉伯数字 0, 1, 2, 3, 4, 5, 6 所组成的连续字符串表示，且不会以 0 为开头；若是大于零的七进制小数，则以由阿拉伯数字 0, 1, 2, 3, 4, 5, 6 与恰好一个小数点所组成的连续字符串表示，且除了以「0.」开头的情况下允许以 0 为开头外，不会以 0 或小数点为开头或结尾。

输出说明：请输出一列，其中包含一个七进制下的数值，表示在七进制运算下，输入中两个数的总和，也须以七进制表示，若此总和可以用七进制正整数表示，则请将该数值表示为以阿拉伯数字 0, 1, 2, 3, 4, 5, 6 所组成的连续字符串，且不得以 0 开头；否则，请将该数值以七进制小数表示，也即表示为以阿拉伯数字 0, 1, 2, 3, 4, 5, 6 与恰好一个小数点所组成的连续字符串，且除了以「0.」开头的情况下允许以 0 为开头外，不得以 0 或小数点为开头或结尾。

范例输入一：123 45

范例输出一： 201

范例输入二： 3 4.2

范例输出二： 10.2

范例输入三： 10.6 0.1

范例输出三： 11

评分说明：正式评分所使用的测试数据共分为 7 组，每组测试资料占 2 分，满分 7+7 分。对于其中至少 3 组测试数据，输入中的数值皆不含小数点。对于所有测试数据，输入中的数值长度均不超过 7 位(小数则包含小数点)。

第二题：最短路径

题目内容：

达狗是一座长方形的幸福城市，道路四通八达，而且都与达狗的边界平行或垂直，任意两条相邻且平行的道路，或者边界与最近的平行道路的间距都恰为 1 单位距离。因为这样的特性，我们用二维平面来描述达狗的任意路口：坐标平面上轴向东侧为正、y 轴向北侧为正，并将达狗西南侧端点之坐标位置定为 $(1, 1)$ ，东北侧端点之坐标位置定为 $(2, 2)$ ，也就是说在坐标点 $(,)$ 时，向东侧走 1 单位距离，会到达 $(+1,)$ ；向北侧走 1 单位距离，则会到达 $(, +1)$ 。

达狗的交通非常便利：每当走到四周的边界，就会被传送至与该边界平行之另一侧边界之对应位置。举例来说，若走到北侧边界 $(, 2)$ 上，会被传送到 $(, 1)$ 的位置；若走到南侧边界 $(, 1)$ 上，会被传送到 $(, 2)$ 的位置。类似地，若走到东侧边界 $(2,)$ 上，会被传送到 $(1,)$ 的位置；走到西侧边界 $(1,)$ 上，会被传送 $(2,)$ 的位置。

现有两个人身在达狗的不同路口，想要约在一个不在边界上的路口会合，并且希望两人所行走的距离总和愈短最好。两人经过一番思考，惊觉这正是传说中的最短路径问题！于是已经学过最短路径算法的你，自告奋勇想帮忙他们计算这个问题。

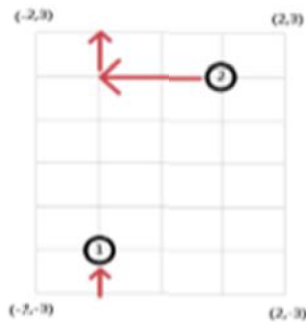
输入说明：

输入的第一列为两个整数 $1, 1$ ，表示达狗西南侧端点之坐标位置为 $(1, 1)$ 。

输入的第二列为两个整数 2, 2，表示达狗东北侧端点之坐标位置为(2, 2)。
 输入的第三列为两个整数 3, 3，表示第一个人所站的坐标位置为(3, 3)。输入
 的第四列为两个整数 4, 4，表示第二个人所站的坐标位置为(4, 4)。
 输入的两人位置不同，且 $1 < 3 < 2$ ； $1 < 4 < 2$ ； $1 < 3 < 2$ ； $1 < 4 < 2$ 。

输出说明：

请输出一列，其中包含一个正整数，表示这两个人所行走的最小距离总和。



范例输入一：

0 0

10 10

5 5

6 7

范例输出一：

3

范例说明一：第一个人由 (5,5)开始往北走 2 单 位，第二个人由(6,7)往西走 1 单 位，即可到达相同的位置(5,7)，行走距离的总和是 3 单位，此为 距离总和最短之走法。

范例输入二：

-2 -3

2 3

-1 -2

1 2

范例输出二：

4

范例说明二：

如图北方朝上，第一个人维持不动 第 一个人由 (1,2) 往西走 2 单位，再向北走 2 单位，途中当行走 1 单位时，会先到达 (-1,3) 之北侧边界，传送至(-1,-3)后，再行走一单位，即可到达(-1,-2)，此为与第一个人相同的位置，行走距离的总和是 4 单位，此为距离总和最短之走法。

评分说明：

正式评分所使用的测试数据共分为 10 组，每组测试资料占 1 分，满分 10 分。对于至少 4 组测试数据， $1 = 1 = 0$ ，输入中的数值皆为非负整数，且值皆小于 1000。对于所有测试数据，输入中的所有数值之绝对值皆小于 2^{31} 。

第三题：递归搜索

题目内容：

温教授正在用心的出着程序设计竞赛的考题。为了让学生能更快地想到解法，避免给予学生太大的挫折，在几经思考后，温教授决定在题目名称中放上与该题相关的解题方法，但不幸地遇到了一些小麻烦。

有些题目可能会与数种不同的解题方法有相关，而也可能有些不同的题目不巧的都与某一种解题方法所相关。温教授希望对每一题，都挑选恰好一种与该题相关的解题方法，来订为该题目的名称。此外，为了避免混淆，温教授希望不同题目的名称不可以重复。

教授已经知道每一题与哪些解题方法有相关，若想要在满足上述条件限制的情形下，对每一题按照上述规则订定其名称，是否能找出至少一种符合规则的解法呢？

为了降低决定题目名称的复杂度，每一套题目都各自独立，不需要考虑不同套题目之间的题目名称是否有重复。对每一套题目而言，只需要订定恰好三题的题目名称，且每一题恰好都与两种解题方法有相关。

教授苦思良久后，决定请已经征服递归函数与回溯法搜索的你，协助他订定每一套题目中三题的题目名称。

输入说明：

输入的第一列为正整数，表示总共有多少套题目需要决定题目名称。接着共有 列，每一列各自表示一套题目，总共有三题，也即每一列总共包含 6 个正整数，相同的数表示着相同的解题方法，反之亦然。其中每一列的第 1,2 个数所表示的解题方法与第一题有相关；第 3,4 个数所表示的解题方法与第二题有相关；第 5,6 个数所表示的解题方法与第三题有相关。输入满足 ≤ 1000 ，且所有表示解题方法的正整数皆不超过 100。

输出说明：

对于每一套题目，请依序输出一列，若对于该套题目，可以依照题意不重复的订定每一题的题目名称，请输出 Yes，否则输出 No。

范例输入一：

1
1 2 3 4 5 6

范例输出一：

Yes

范例说明一：

三题的题目名称可以分别订为 1、4、5 符合条件且不重复。

范例输入二：

2
1 2 1 2 1 2
2 1 2 1 2 1

范例输出二：

No
No

范例说明二：

均不存在不重复之题目名称订法。

范例输入三：

1
1 2 1 3 2 3

范例输出三：

Yes

范例说明三：

三题的题目名称可以分别订为 2、1、3 符合条件且不重复。

评分说明：正式评分所使用的测试数据仅有一组，满分 10 分。

第四题：二维数组

题目内容：

有一座可以视为二维的方格平面的长方形城市，由上而下共有 m 列，由左至右共有 n 行，总共划分为 $m \times n$ 个方格。这个城市因为治安很差，所以每个格子中都有机会出现野生的甩棍可以捡拾以方便斗殴。你想要捡一些甩棍，但你知道城市的其中一些方格设有警察局，如果警察局内的警察在这个方格或与其同一行或同一列的任何方格侦测到有人在捡拾甩棍，都会毫不留情地把捡拾的人预防性羁押，所以为了安全，你绝对不会去那些方格捡拾甩棍。你想要知道在上述条件下，你还可以去那些方格捡拾甩棍。

输入说明：

输入的第一列依序为正整数 m, n ，表示这座城市由上而下共有 m 列，由左至右共有 n 行，总共划分为 $m \times n$ 个方格。

接着共有 m 列，每列为一个长度恰为 n 的字符串，每个字符则可能为「.」（ASCII 编码 46）或「#」（ASCII 编码 35），若其中第 i 列的第 j 个字符为「#」，表示城市中第 i 列第 j 行的方格有装设甩棍毁灭器，否则表示没有。除此之外，该列不会有其他的字符。

输出说明：

请输出 n 列，每列为长度恰为 m 的字符串，对于第 i 列的第 j 个字符，若城市中第 i 列第 j 行的方格有机会可以捡到甩棍，则请输出「X」，否则请输出「#」。请勿输出其他的字符或空白。

范例输入一：

```
2 2
#. .
```

范例输出一：

```
##
#X
```

范例输入二：

3 2

##

..

#. .

范例输出二：

##

##

##

范例输入三：

1 4

..#. .

范例输出三：

####

评分说明：

正式评分所使用的测试数据共分为 10 组，其输入条件限制及配分列举如下，每组测试资料完全正确得该组测试资料配分，否则不给分，满分 10 分。

1. $\leq 10 \leq 10$
2. $\leq 50 \leq 50$
3. $\leq 300 \leq 300$ 至多 300 个 #
4. $\leq 300 \leq 300$
5. $\leq 300 \leq 300$
6. $\leq 3000 \leq 3000$ 至多 3000 个 #
7. $\leq 3000 \leq 3000$ 至多 30000 个 #
8. $\leq 3000 \leq 3000$ 至多 300000 个 #
9. $\leq 3000 \leq 3000$
10. $\leq 3000 \leq 3000$