

A LJF算距离

题目描述

对于一个点 x 和点集合 $Y = \{y_1, y_2, \dots, y_n\}$ ，定义点到集合的距离 d 为

$$d(x, Y) = \min_{y_i \in Y} |x - y_i|$$

现给出点 x 和集合 Y ，请计算 $d(x, Y)$ 。

输入

输入共三行：

第一行一个正整数 n ，表示集合 Y 中元素个数；

第二行 n 个空格分隔的整数 y_1, y_2, \dots, y_n ，表示集合 Y 中的元素；

第三行一个整数 x ，表示点 x 。

输出

输出一行，一个整数表示结果 $d(x, Y)$ 。

输入样例

```
10
1 33 4 65 6 8 13 21 0 -10
26
```

输出样例

```
5
```

数据范围

- $1 \leq n \leq 1,000$
- $-1,000 \leq y_i \leq 1,000$
- $-2,000 \leq x \leq 2,000$

B 简单的查找

题目介绍

正如题目所说，这道题只是想考察线性查找。

JJJ 有 n 个数字，和 m 次询问。每一次请求包含两个参数 l, r ，询问第 l 个数到第 r 个数之间（包含第 l 和第 r 个数），第3大的数是多少。

输入格式

第一行，两个正整数 n, m ，表示数字的个数和询问的次数。

第二行，共 n 个整数，表示 n 个数字的值。

接下来 m 行，每行2个参数 l, r ，含义如题目所示。

输出格式

共 m 行，表示每次询问得到的结果。

输入样例1

```
7 3
2 8 6 3 1 7 1
1 7
3 6
4 7
```

输出样例1

```
6
3
1
```

数据范围

对于100%的数据： $1 \leq n, m \leq 5 \times 10^3$ ， $3 \leq r - l + 1 \leq n$ ， n 个数字均在 int 范围内。

后记

这题还有复杂度更优的做法，对数据结构感兴趣的同学可以尝试一下。

Author: JJJ .

C 宋老师的名次预测3.0

题目描述

还记得 `宋老师的名次预测2.0` 吗？plus升级版3.0来了！

选手人次为50人，且实际结果仍然是这50人依次获得了第1到第50名，这次不仅需要你判断每组数据猜对了多少人，还需要你将所有猜对人数从大到小输出。

输入格式

输入共 $n + 1$ 行；

第一行一个正整数 n ，表示接下来有 n 组预测；

接下来 n 行每行为50个以空格分隔的整数，依次是对50位选手预测的名次；

输出格式

输出共 n 行，每行一个非负整数，为 n 组预测猜对人数从大到小排序后的结果。

输入样例

```
5
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
50 49 48 47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22
21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
17 2 44 43 40 16 26 22 13 7 11 12 21 48 32 41 30 14 31 42 29 4 23 24 45 38 39 28 10 19
6 15 20 46 33 8 37 1 18 36 50 3 5 9 25 34 47 27 49 35
32 2 31 4 22 11 19 8 24 48 6 12 13 41 15 16 17 39 44 20 21 49 23 7 25 26 27 28 14 30 42
45 33 34 35 36 37 38 18 40 29 3 43 10 1 46 47 9 5 50
1 2 3 18 32 36 7 27 38 10 11 12 13 15 47 16 17 4 19 21 20 22 23 24 25 9 8 28 29 30 43 5
33 34 35 6 37 26 39 40 41 42 31 44 45 46 14 48 49 50
```

输出样例

```
50
32
27
9
0
```

样例解释

对于输入的五组预测，猜对的人数分别是50、0、9、27、32，所以按从大到小排序后输出如上

数据范围

$1 \leq n \leq 1000$

HINT

在 `名次预测2.0` 的基础上，将每组猜对人数顺序存储在一个数组中，然后对数组元素进行一次冒泡排序，注意是从大到小排序哦～

当然你也可以使用计数排序（课后自行了解）

AUTHOR:Songyou

D 实矩阵乘法简单版

题目描述

计算2个矩阵的乘积。

输入

第一行为4个整数，分别为 a_1, b_1, a_2, b_2 ， $(1 \leq a_1, b_1, a_2, b_2 \leq 100)$ ，其中 a_i 表示第 i 个矩阵的行数， b_i 表示第 i 个矩阵的列数，保证有 $b_1 = a_2$ 。

之后有 $a_1 + a_2$ 行，首先的 a_1 行每行有 b_1 个非负整数，表示第一个矩阵，之后的 a_2 行每行有 b_2 个非负整数，表示第二个矩阵。

矩阵中的每一个元素不超过1000。

输出

程序的输出应该是一个 $a_1 * b_2$ 的矩阵。

输出 a_1 行，每行有 b_2 个非负整数，两个整数之间用空格隔开。

保证输出结果在 `int` 范围内。

输入样例

```
3 2 2 3
1 1
1 1
1 1
1 1 1
1 1 1
```

输出样例

```
2 2 2
2 2 2
2 2 2
```

E 上 α 分位点

题目描述

呱呱泡蛙通过永不消逝的牵绊，向小智发送波导信息。

“在大二秋季学期的《概率与统计》的课程中，会学习到分位点（Quantile，也称分位数）的概念。”

“标准正态分布的上 α 分位点 z_α ”是指，在 z_α 到正无穷的区间上，分布函数差为 α 。

$$\int_{z_\alpha}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt = \alpha$$

根据 Hint 中列出的内容（课外知识点，C语言无需掌握），立即得到方程：

$$\frac{1}{2} - \frac{1}{2} \operatorname{erf}\left(\frac{z_\alpha}{\sqrt{2}}\right) - \alpha = 0$$

呱呱泡蛙：“你能不能利用C语言内置函数 `erf`，在已知自变量 α 的情况下，计算上 α 分位点？其实只需要写C程序查找上面这个方程的根就可以搞定呢[吃瓜表情]”

小智听罢，两眼一黑，战斗不能。快来帮帮他罢。

输入

多行数据，每行一个浮点数 α 。输入的 α 全部落在 0 到 1 之间，保证上述方程有解 z_α 。

输出

对于每行数据，输出一行对应的上 α 分位点 z_α ，保留 8 位小数。

输入样例

```
0.1
0.01
0.001
0.0001
```

输出样例

```
1.28155157
2.32634787
3.09023231
3.71901649
```

Hint

自变量浮点数 α 的定义域是 0 到 1，而上 α 分位点 z_α 的值域是全体实数。但是由于C语言的计算精度有限，事实上可以保证所有 z_α 都在 -10 到 10 之间。

- 知识点1：在`<math.h>`中，函数 `erf` 是：

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

这个积分是积不出来的，所以它不是初等函数。

- 知识点2：标准正态分布的分布函数是：

$$\int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt = \frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{x}{\sqrt{2}}\right)$$

该分布函数在实数范围连续，在负无穷为 0，在正无穷为 1。



Author：登场动画的呱呱泡蛙

F Java今天吃什么

题目描述

Java有本菜谱，Java每天吃什么全都靠这本菜谱。Java每天会烧一页菜谱上的菜，并且今天看的菜谱一定是昨天看的菜谱的下一页。如果昨天已经看到了最后一页，那么今天就看菜谱的第一页来烧菜。

由于Java的家里经常有朋友来做客，所以Java习惯在家里囤很多菜。假设今天是这个月的 x 号，并且他今天看的是菜谱的第 n 页烧的菜，请你告诉他下个月的 y 号他将会烧什么菜，方便他提前买好菜。（一个月默认有 31 天）

输入

第一行是由空格分隔的三个整数，分别代表 x ， n 和 y ($1 \leq x, y \leq 31$)；

接下来是多行数据输入，代表Java的菜谱：其中第 i 行代表菜谱的第 i 页，是若干由 `,` 分隔的菜名，菜名是一个仅包含大小写英文字母的字符串。

保证输入的总行数不超过 100，且每一行的字符串长度不超过 100，保证 n 不大于菜谱的总页数。

输出

输出若干行字符串，为Java下个月的 y 号应该烧的菜的菜名。菜名的顺序应该和菜谱中给出的相同。

输入样例

```
31 1 1
rou,dan,cong,ji
lizhi,pinghu,duanyaojiu,yibeikou
```

输出样例

```
lizhi
pinghu
duanyaojiu
yibeikou
```

HINT

用 `fgets` 向二维数组中读入不定行数的字符串，用 `ctrl+z` 结束输入：


```

#include <stdio.h>
#define ROW 10
#define LEN 10

int main() {
    char s[ROW][LEN];
    int i = 0;
    while (fgets(s[i], LEN, stdin) != NULL) {
        printf("%s", s[i]); // fgets会将行末换行符读入
        i++;
    }
    return 0;
}

```

用 `gets` 向二维数组中读入不定行数的字符串，用 `ctrl+z` 结束输入：

```

#include <stdio.h>
#define ROW 10
#define LEN 10

int main() {
    char s[ROW][LEN];
    int i = 0;
    while (gets(s[i]) != NULL) { // gets函数不安全，请谨慎使用
        printf("%s\n", s[i]); // gets会将行末换行符读入，但是会将其丢弃
        i++;
    }
    return 0;
}

```

用 `scanf` 向二维数组中读入不定行数的字符串，用 `ctrl+z` 结束输入：

```

#include <stdio.h>
#define ROW 10
#define LEN 10

int main() {
    char s[ROW][LEN];
    int i = 0;
    while (scanf("%s", s[i]) != EOF) { // scanf遇到空白符会停止，因此如果一行字符串中有空格的话
        // 会读不完一整行
        printf("%s\n", s[i]); // scanf不会将行末换行符读入
        i++;
    }
    return 0;
}

```

ps: 别问佳伟他为什么会吃荔枝、平湖、断么酒和一杯口。

G 爱偷懒的球场管理员

题目描述

AAUB幼儿园有一个负责管理羽毛球场的管理员，每天都有很多小朋友预约在某一时间段打羽毛球（时间段用区间 $[s, e]$ 表示，其中 s, e 都为整数， s 表示时间段的开始时间点， e 表示时间段的结束时间点）。

管理员希望找出有小朋友预约的所有时间段，于是他拜托擅长写代码的你帮他将这些预约中有重叠的时间段全部合并，并且返回他一个不重叠的时间段数组 t ， t 恰好覆盖所有的预约时间段（以便于在这些时间段之外自己能够摸鱼）。

输入格式

$n + 1$ 行。

第一行 n 表示一共有 n 个时间段，

接下来 n 行，每行两个整数，分别表示时间段 i 的开始时间点 s_i 和结束时间点 e_i 。

输出格式

若干行。

每行两个整数，分别表示数组 t 的第 i 个时间段的起点和终点。

数组 t 包含若干个不重叠的时间段，并且满足这些时间段的并集恰好覆盖有预约的所有时间段。

按照时间段开始时间点的升序输出结果

输入样例1

```
3
1 4
2 7
9 12
```

输出样例1

```
1 7
9 12
```

样例解释

时间段 $[1, 4]$ 和 $[2, 7]$ 有重叠部分，将其合并为 $[1, 7]$ 。

合并后的 $[1, 7]$ 和 $[9, 12]$ 没有重叠部分，所以数组 t 为 $[[1, 7], [9, 12]]$ 。

输入样例2

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

输出样例2

```
1 5
```

样例解释

时间段 $[1, 2]$, $[1, 3]$, $[1, 5]$ 有重叠部分, 将其合并为 $[1, 5]$ 。

最终的数组 t 为 $[[1, 5]]$

数据范围

$$1 \leq n \leq 1000$$

$$1 \leq s_i \leq e_i \leq 10^{11}$$

HINTS

1. 不需要考虑球场的容量限制, 同一个时间点可以容纳的预约数量没有上限。
2. 或许可以先把预约时间段排个序。
3. 以下情况是合法的: 预约的时间段为 $[x, x]$ 。

AUTHOR : zym

H 方阵数局

题目描述

有一个 $m \times n$ 的方阵，用数字填满，如图所示：

12	24	36	8
17	20	7	3
12	15	10	9
8	6	4	11

在其之上有如下移动规则：

- 若某一方格上的数为2的倍数且不为3的倍数，则可向上移动一格。
- 若某一方格上的数为3的倍数且不为2的倍数，则可向下移动一格。
- 若某一方格上的数同时为3和2的倍数，则可向右移动一格。
- 若某一方格上的数不为2的倍数且不为3的倍数，则可向左移动一格。

现在，给你一个这样的数字方阵，从左下角开始行动，判断是否能到达右上角。

输入

第一行两个正整数，分别为方阵的行数 m 和列数 n 。

接下来， m 行，每行 n 个用空格隔开的正整数，表示方阵中每一格的数字 a_{mn} （输入顺序为从上往下，从左往右）。

输出

第一行：若能到达右上角，则输出 `True`，否则输出 `False`。

第二行：若能到达右上角，输出一个整数，表示所走的步数，若无法到达，输出一个浮点数，表示最远到达的距离（类似直角三角形斜边长，保留小数点后3位）。

输入样例1

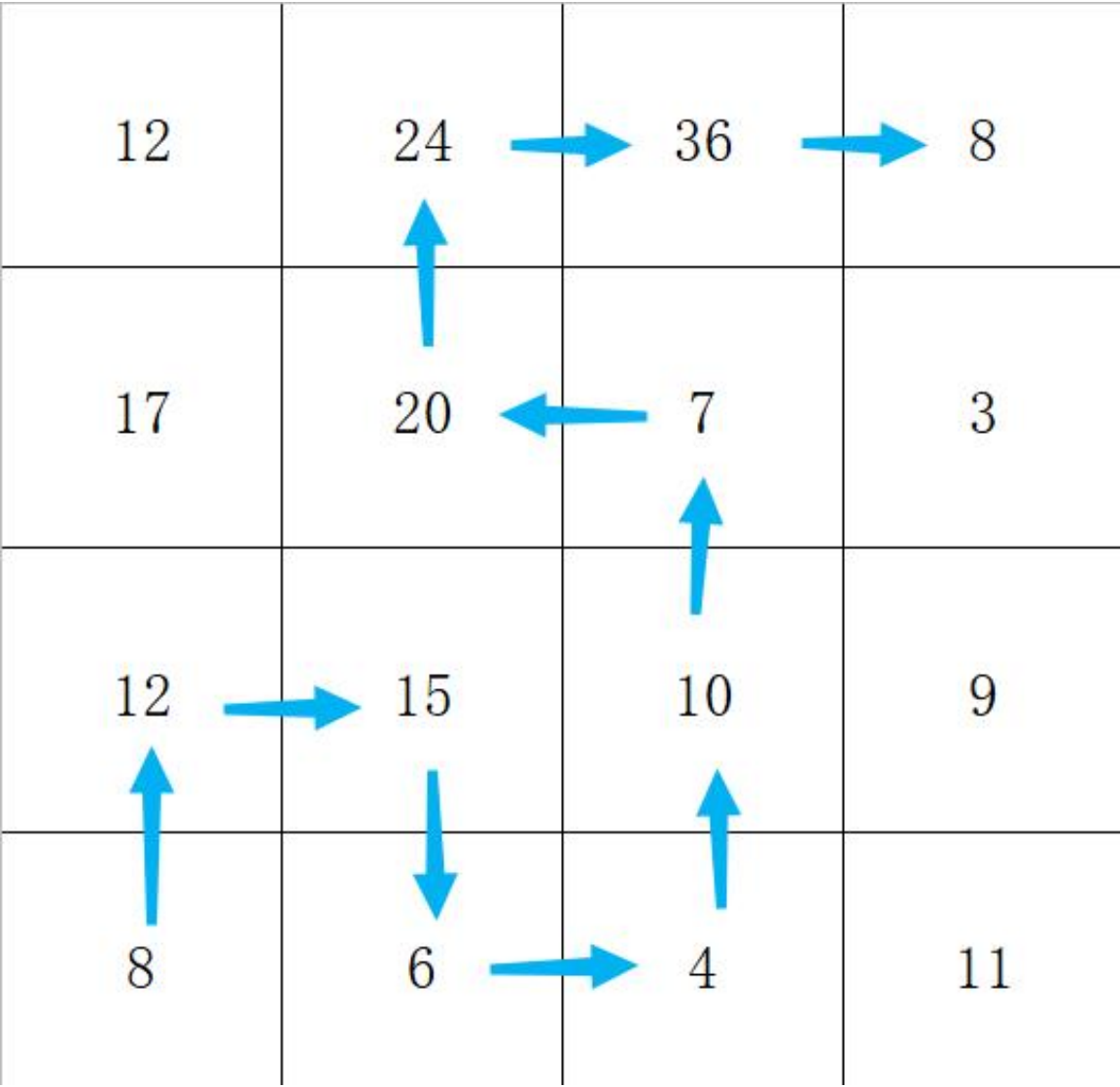
```
4 4
12 24 36 8
17 20 7 3
12 15 10 9
8 6 4 11
```

输出样例1

```
Ture
10
```

样例解释1

方阵及线路如图所示，能到达终点



输入样例2

```
4 5
7 24 9 4 10
12 26 15 16 11
4 5 3 6 54
10 20 6 8 14
```

输出样例2

```
False
4.123
```

样例解释2

方阵及线路如图所示，不能到达终点，“最远处”为54所在位置。

7	24	9	4	10
12	26	15	16	11
4	5	3	6	54
10	20	6	8	14

数据范围

$0 < m, n \leq 20$
 $0 < a_{mn} \leq 2147483647$

HINT

- 1. 你有在一个迷宫中的同一个地方来回兜兜转转，始终走不出去的经历吗？
- 2. 所有的坑点和直线完全一样，别再踩了。
- 3. 还有进阶版，别急（不是三维）。

Author: Arthas

I zhnの数字零

题目描述

众所周知，zhn是魔法少女，他使用魔法让这样一个问题摆在了你们面前：

给你 n 个数，问你从这 n 个数中取出 k 个数，这 k 个数的乘积的末尾最多有多少个0。

输入

第一行包括两个整数 n 和 k

$(1 \leq n \leq 200, 1 \leq k \leq n)$ 。

第二行有 n 个正整数 a_i

$(1 \leq a_i \leq 10^{18})$

输出

你选出的 k 个数字的乘积末尾0的个数

输入样例

```
3 2
50 4 20
```

输出样例

```
3
```

AUTHOR:魔法少女zhn

J 图图玩狼人杀2.0

题目背景

“狼人杀”分为两大阵营：狼人阵营和好人阵营，玩家发言环节，每位玩家说的话都有可能是真，也有可能是假，好人们需要明察秋毫才能赢得胜利。

上帝（发牌者）给了图图一个好人身份，同时在每轮玩家发言过后都会告诉图图，有几名玩家说的是真话。

题目描述

这天，上帝告诉图图， n 名玩家里有 m 名玩家说的是假话，并且已知图图是好人，且图图说的是真话；

与上次不同的是，由于游戏中狼人的数量通常都是不固定的，因此这次狼人数将作为一个变量输入到程序中。

输入格式

第一行四个以空格分隔的正整数 $n, m, id_{utu}, wolf$ ，分别表示玩家数，说假话的玩家数，图图所在的座位编号，以及狼人的数量；

接下来 n 行，每行一个整数 id ，第 i ($1 \leq i \leq n$) 行表示编号为 i 的玩家对编号为 $|id|$ 的玩家身份的评论，用正号表示好人，负号表示狼人；比如， -5 表示认为 5 号玩家是坏人， $+4$ 表示认为 4 号玩家是好人。

输出格式

若有解，则第一行输出 `Thank God I know!` 接下来，按递增顺序输出 $wolf$ 个狼人的编号，中间以一个空格分隔。如果存在多解，则输出多行， $wolf$ 个编号中第一个编号较小的先输出，若第一个编号相同，则第二个编号较小的先输出，若第二个编号相同，则第三个编号较小的先输出，以此类推；

若无解，输出一行 `Still can't find.`

输入样例

```
6 3 3 2
-5
-1
+3
-1
+4
+6
```

输出样例

```
Thank God I know!
4 5
5 6
```

样例解释

6 名玩家中有 2 名狼人，图图所在的座位编号为 3 号，有 3 个人说假话。

白天发言时，1 号玩家认为 5 号玩家是坏人，2 号玩家认为 1 号玩家是坏人，3 号玩家（图图）阐述自己是好人，4 号玩家同样认为 1 号玩家是坏人，5 号玩家认为 4 号玩家是好人，6 号玩家阐述自己是好人。

由于 6 人中有 3 人说的是错的，经过检验，4 号玩家和 5 号玩家是狼人，或是 5 号玩家和 6 号玩家是狼人。

数据范围

$$6 \leq n \leq 18$$
$$1 \leq m < n$$
$$1 \leq id_{tutu} \leq n$$
$$1 \leq |id| \leq n$$
$$1 \leq wolf \leq \min\{6, n - 1\}$$

Author: wqh

K 虚假的签到题

本题为附加题

题目介绍

很简单的题意，给你 n 个正整数，请找出出现次数最多的那个数，保证这个数出现次数大于 $\frac{n}{2}$

输入格式

输入共两行

第一行一个正整数 n ，表示数的个数；

第二行 n 个正整数 a_1, a_2, \dots, a_n ，代表这 n 个正整数。

输入样例1

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

输出样例1

```
2
```

输入样例2

```
9
1 1 4 5 1 4 1 9 1
```

输出样例2

```
1
```

数据范围

$3 \leq n < 10^6$

$1 \leq a_i < 10^9$

HINT

注意内存限制

AUTHOR：做签到题的cxccxc