

1. 买可乐

时限：1s

(cola.pas/c/cpp)

【问题描述】

佳佳非常喜欢喝可乐。零花钱发放之后，他决定用所有的钱去买可乐喝。

超市里出售两种不同类型的可乐，有两种包装形式：易拉罐包装和玻璃瓶包装。

现在已知一瓶易拉罐包装的可乐定价为 a 元，一瓶玻璃瓶包装的可乐定价为 b 元。

由于可乐是用玻璃瓶装的，有回收利用的价值，在佳佳喝完可乐之后，他可以将玻璃瓶子返还给超市，并获得 c 元 ($c < b$)。超市不接受易拉罐可乐瓶子的返还。

现在佳佳有 n 元的零花钱，已知 a 、 b 、 c ，请问佳佳最多能够喝多少瓶可乐？

【输入】

输入文件名为 `cola.in`。

输入共 1 行，一共四个正整数 n 、 a 、 b 、 c ，以空格分割，含义参见问题描述

【输出】

输出文件名为 `cola.out`。

输出共 1 行，输出一个数字，表示佳佳最多能喝多少瓶可乐

【输入输出样例 1】

<code>cola.in</code>	<code>cola.out</code>
10 11 9 8	2

【输入输出样例 2】

<code>cola.in</code>	<code>cola.out</code>
10 5 6 1	2

【样例解释】

对于第一个样例：

佳佳有 10 元零花钱，一瓶易拉罐装的可乐价格为 11 元，玻璃瓶装的可乐价格为 9 元，返还玻璃瓶可以获得 8 元。

佳佳先购买一个玻璃瓶装的可乐，剩余 1 元。然后将玻璃瓶出售获得 8 元。之后再购买一个玻璃瓶装的可乐，剩余 0 元。他可以再出售玻璃瓶获得 8 元，但此时佳佳已经没有办法再购买玻璃瓶装可乐或者易拉罐装可乐，因此佳佳最多可以喝到 2 瓶可乐。

【数据范围】

对于 60% 的数据 $1 \leq n, a, b, c \leq 10^4$

对于 100% 的数据 $1 \leq n, a, b, c \leq 10^{18}$

2. 抢座位游戏

时限：1s

(chair.pas/c/cpp)

【问题描述】

抢座位游戏相信大家都不陌生。在游戏中， n 位同学围绕 $n-1$ 个椅子伴随着音乐慢慢移动。当音乐结束的时候，每个人都要尽自己所能去“抢”坐一个位子。那位没有抢到位子的同学会被淘汰出局。

现在我们要去模拟这样的一个游戏场景：我们会给你一个游戏参赛者名单，按照同学的初始顺序给出。

例如：“Jack Zhang”，“Alice Wang”，“Zack”。以换行符作为分割。

同时也会给你一个长度为 n 的字符串（仅包含‘.’和‘h’），‘.’表示该位置处没有椅子，‘h’表示该位置处有椅子。

例如：hh.h

音乐结束后，假如一位同学在‘.’处，那么这位同学将会被淘汰。

现在已知有 n 位同学，以名字（字符串）的方式给出他们的初始顺序和描述座椅情况的字符串。每过 1 秒，每位同学都会向前进一个位置。例如，当前时刻有四位同学“a”，“b”，“c”，“d”，1 秒之后，他们的位置会变为“d”，“a”，“b”，“c”。在第 s 秒末音乐结束，请你求出那位被淘汰的同学，并输出他的名字。

【输入】

输入文件名为 chair.in。

输入第一行为一个正整数 n ，表示有 n 位同学

接下来 n 行，每行一个字符串，表示第 i 位同学的名字

第 $n+2$ 行有一个长度为 n 的字符串，仅包含‘h’和‘.’，描述座位情况。数据保证仅有 1 个字符为‘.’其它字符均为‘h’

第 $n+3$ 行为一个自然数 s ，表示在 s 秒之后音乐将会结束

【输出】

输出文件名为 chair.out。

输出共 1 行，输出一个字符串，表示被淘汰同学的名字

【输入输出样例 1】

chair.in	chair.out
4 Emma Sue Billy Bob Thornton Joe Cassius Clay hh.h 1	Billy Bob Thornton

【输入输出样例 2】

chair.in	chair.out
6 Jack Dempsey Joe Louis Rocky Marciano Cassius Clay George Foreman Mike Tyson h.hhhh 500	Mike Tyson

【数据范围】

对于 100% 的数据 $1 \leq n, \text{len}(\text{name}_i) \leq 50, 0 \leq s \leq 600$

3.ultmaster 的逃离

时限：1s

(escape.pas/c/cpp)

【问题描述】

中华大地上，人人都知道有一个叫做 ultmaster 的 dalao。这倒不是因为他在器乐领域、作曲领域、算法竞赛领域与机器学习领域都已经臻至化境，而是因为他实在是太萌了。

他总是喜欢到世界各处探险，可是呢遗憾的是总有那么些地方不仅难走还危机丛生。

这一次，他来到了一个 m 行 n 列的迷宫，迷宫中有的格子里有可以增加寿元的草药，有的空空如也，有的却有减少寿元的毒气。增加寿元的草药当然要吃，减少寿元的毒气却也不得不吸。在进入迷宫之前，ultmaster 拥有的寿元为 h 。要保证不能有任何一个时刻寿元值小于 0（不包括 0），不然 ultmaster 就会死在路上。

迷宫的入口在左上角，出口在右下角，由于 ultmaster 方位感惊人，他竟然能保证自己只往右或者往下走。

那么问题来了，从迷宫出去之后，ultmaster 最多能剩多少寿元呢？



【输入】

输入文件名为 `escape.in`。

第一行三个整数 m, n, h

接下来的 m 行，每行 n 个整数，分别代表 ultmaster 在相应格子中将要改变的寿元（即这个整数大于 0 为上述的格子内有草药的情况，等于 0 为上述的格子内空空如也的情况，小于 0 为上述的格子内有毒气的情况），每个整数绝对值不超过 100，整数的绝对值表示对寿元产生的影响（即增加或减少的数目）。

【输出】

输出文件名为 `escape.out`。

输出一行一个整数表示答案。

如果 ultmaster 一定会死在路上，输出 -1。

【输入输出样例 1】

escape.in	escape.out
2 2 1	0
0 2	
2 -3	

【输入输出样例 2】

escape.in	escape.out
2 2 9 -1 -2 -3 -4	2

【数据范围】

对于 30% 的数据 $1 \leq n, m \leq 5$

对于 60% 的数据 $1 \leq n, m \leq 20$

对于 100% 的数据 $1 \leq n, m \leq 10^3, 1 \leq h \leq 10^5$

4. 面向对象程序设计

时限：1s

(design.pas/c/cpp)

【问题描述】

在面向对象程序设计中，常常会运用到函数的扩展与重写。当一个类继承某个类的时候，它可以调用所有父类可以调用的函数。它可以声明新的函数。当新的函数签名与父类的某个函数一致时，就会发生函数的覆盖（重写）。所以，在子类的实例调用某个函数时，它会调用最近的父类（有可能是它自己）的那个函数实现。

这里我们不考虑访问权限等情况，我们只关心某个类在调用某个函数时，这个函数是在哪个类中实现的。

具体题意请参考样例及其解释。

【输入】

输入文件名为 design.in。

输入具有如下格式

```
n
p2 p3 ... pn
t1 a11 a12 ... a1t1
t2 a21 a22 ... a2t2
...
tn an1 an2 ... antn
q
u1 r1
u2 r2
...
uq rq
```

n 表示有 n 个类，这些类从 1 到 n 编号

p_i 表示第 i 个类的父类编号。 $1 \leq p_i \leq i-1$ 。第 1 个类是所有类共同的祖先类，这个类没有父类

t_i 表示在第 i 个类中定义了多少个函数， $a_{i1}, a_{i2}, \dots, a_{it_i}$ 表示第 i 个类中的函数列表。同一个类的函数列表中不会出现两个相同的函数

q 表示询问个数

u_i, r_i 表示第 i 个询问，询问第 u_i 个类的实例在调用 r_i 函数时，调用的是哪个类中的版本

【输出】

输出文件名为 design.out。

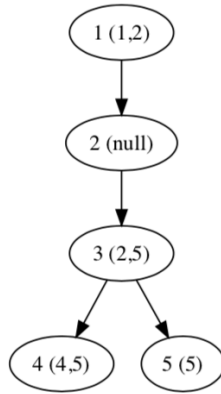
输出共 q 行，每行输出一个数，表示第 i 个询问的答案，若不存在，输出-1

【输入输出样例 1】

design.in	design.out
5	-1
1 2 3 3	3

2 2 1	4
0	-1
2 5 2	
2 4 5	
1 5	
4	
3 4	
5 2	
4 5	
1 3	

【样例解释】



一共有 5 个类，这 5 个类构成一颗树，其中 1 为其根节点。

第 2 个类的父亲为 1 号类，第 3 个类的父亲为 2 号类，第 4/5 个类的父亲为 3 号类。

1 号类中有 1, 2 号函数，3 号类中有 2, 5 号函数……

第一组询问 (3, 4) 表示询问第 3 号类里面的 4 号函数是从哪个类派生来的。从 3 号类开始向上找，发现并没有 4 号函数出现，因此输出 -1

第二组询问 (5, 2) 表示询问第 5 号类里面 2 号函数是从哪个类派生来的，从 5 号类开始往上找，发现最近的 2 号出现在 3 号类中，因此输出 3

其它询问略

【数据范围】

对于 60% 的数据：

$$2 \leq n \leq 10^3, 1 \leq a_{ij} \leq 10^3, 0 \leq t_i \leq 10^3, \sum t_i \leq 10^3, 1 \leq q \leq 10^3, 1 \leq u_i \leq n, 1 \leq r_i \leq 10^3$$

对于 100% 的数据：

$$2 \leq n \leq 10^5, 1 \leq a_{ij} \leq 10^6, 0 \leq t_i \leq 10^6, \sum t_i \leq 10^6, 1 \leq q \leq 10^5, 1 \leq u_i \leq n, 1 \leq r_i \leq 10^6$$

5. 佳佳的困惑

时限：1s

(number.pas/c/cpp)

【问题描述】

佳佳最近对个位的平方数特别着迷（也就是 0, 1, 4, 9 四个数字），因此他想要知道用 0, 1, 4, 9 四个数字组成的所有数字组成的数列中，第 n 小的数字是多少？（ n 从 0 开始）

例如：数列的前几项 0, 1, 4, 9, 10, 11, 14……

【输入】

输入文件名为 number.in。

输入共 1 行，一个自然数 n

【输出】

输出文件名为 number.out。

输出共 1 行，输出一个数字，表示数列的第 n 项

【输入输出样例 1】

number.in	number.out
0	0

【输入输出样例 2】

number.in	number.out
123	1949

【数据范围】

对于 100% 的数据 $0 \leq n \leq \text{MAX_INT} (2^{31}-1)$

注：本题共有 100 个测试点，1 个测试点 1 分