

CCF 全国信息学奥林匹克联赛 (NOIP2015) 复赛

提高组 day1

(请选手务必仔细阅读本页内容)

一. 题目概况

中文题目名称	神奇的幻方	信息传递	斗地主
英文题目与子目录名	magic	message	landlords
可执行文件名	magic	message	landlords
输入文件名	magic.in	message.in	landlords.in
输出文件名	magic.out	message.out	landlords.out
每个测试点时限	1 秒	1 秒	2 秒
测试点数目	10	10	20
每个测试点分值	10	10	5
附加样例文件	有	有	有
结果比较方式	全文比较 (过滤行末空格及文末回车)		
题目类型	传统	传统	传统
运行内存上限	128M	128M	1G

二. 提交源程序文件名

对于 C++语言	magic.cpp	message.cpp	landlords.cpp
对于 C 语言	magic.c	message.c	landlords.c
对于 pascal 语言	magic.pas	message.pas	landlords.pas

三. 编译命令 (不包含任何优化开关)

对于 C++语言	g++ -o magic magic.cpp -lm	g++ -o message message.cpp -lm	g++ -o landlords landlords.cpp -lm
对于 C 语言	gcc -o magic magic.c -lm	gcc -o message message.c -lm	gcc -o landlords landlords.c -lm
对于 pascal 语言	fpc magic.pas	fpc message.pas	fpc landlords.pas

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、全国统一评测时采用的机器配置为: CPU AMD Athlon(tm) II x2 240 processor, 2.8GHz, 内存 4G, 上述时限以此配置为准。
- 4、只提供 Linux 格式附加样例文件。
- 5、特别提醒: 评测在当前最新公布的 NOI Linux 下进行, 各语言的编译器版本以其为准。

1. 神奇的幻方

(magic.cpp/c/pas)

【问题描述】

幻方是一种很神奇的 $N \times N$ 矩阵：它由数字 $1, 2, 3, \dots, N \times N$ 构成，且每行、每列及两条对角线上的数字之和都相同。

当 N 为奇数时，我们可以通过以下方法构建一个幻方：

首先将 1 写在第一行的中间。

之后，按如下方式从小到大依次填写每个数 $K (K = 2, 3, \dots, N \times N)$ ：

1. 若 $(K-1)$ 在第一行但不在最后一列，则将 K 填在最后一行， $(K-1)$ 所在列的右一列；
2. 若 $(K-1)$ 在最后一列但不在第一行，则将 K 填在第一列， $(K-1)$ 所在行的上一行；
3. 若 $(K-1)$ 在第一行最后一列，则将 K 填在 $(K-1)$ 的正下方；

若 $(K-1)$ 既不在第一行，也不在最后一列，如果 $(K-1)$ 的右上方还未填数，则将 K 填在 $(K-1)$ 的右上方，否则将 K 填在 $(K-1)$ 的正下方

现给定 N ，请按上述方法构造 $N \times N$ 的幻方。

【输入格式】

输入文件名为 `magic.in`。

输入文件只有一行，包含一个整数 N ，即幻方的大小。

【输出格式】

输出文件名为 `magic.out`。

输出文件包含 N 行，每行 N 个整数，即按上述方法构造出的 $N \times N$ 的幻方。相邻两个整数之间用单个空格隔开。

【输入输出样例 1】

magic.in	magic.out
3	8 1 6 3 5 7 4 9 2

见选手目录下的 `magic/magic1.in` 和 `magic/magic1.ans`。

【输入输出样例 2】

见选手目录下的 `magic/magic2.in` 和 `magic/magic2.ans`。

【数据规模与约定】

对于 100% 的数据， $1 \leq N \leq 39$ 且 N 为奇数。

2. 信息传递

(message.cpp/c/pas)

【问题描述】

有 n 个同学 (编号为 1 到 n) 正在玩一个信息传递的游戏。在游戏里每人都有一个固定的信息传递对象, 其中, 编号为 i 的同学的信息传递对象是编号为 T_i 的同学。游戏开始时, 每人都只知道自己的生日。之后每一轮中, 所有人会同时将自己当前所知的生日信息告诉各自的信息传递对象 (注意: 可能有人可以从若干人那里获取信息, 但是每人只会把信息告诉一个人, 即自己的信息传递对象)。当有人从别人口中得知自己的生日时, 游戏结束。请问该游戏一共可以进行几轮?

【输入格式】

输入文件名为 message.in。

输入共 2 行。

第 1 行包含 1 个正整数 n , 表示 n 个人。

第 2 行包含 n 个用空格隔开的正整数 T_1, T_2, \dots, T_n , 其中第 T_i 个整数表示编号为 i 的同学的信息传递对象是编号为 T_i 的同学, $T_i \leq n$ 且 $T_i \neq i$ 。

数据保证游戏一定会结束。

【输出格式】

输出文件名为 message.out。

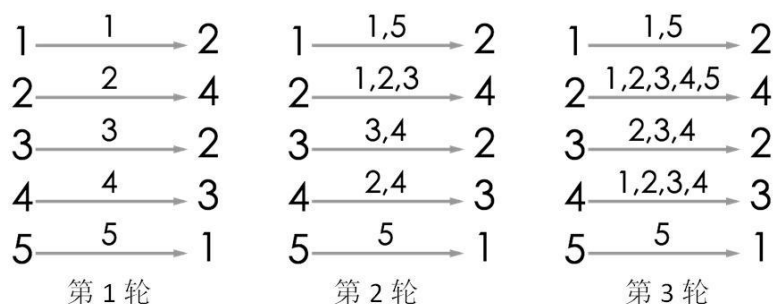
输出共 1 行, 包含 1 个整数, 表示游戏一共可以进行多少轮。

【输入输出样例 1】

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

见选手目录下的 message/message1.in 与 message/message1.ans。

【输入输出样例 1 说明】



游戏的流程如图所示。当进行完第 3 轮游戏后, 4 号玩家会听到 2 号玩家告诉他自己的生日, 所以答案为 3。当然, 第 3 轮游戏后, 2 号玩家、3 号玩家都能从自己的消息来源得知自己的生日, 同样符合游戏结束的条件。

【样例输入输出 2】

见选手目录下的 message/message2.in 与 message/message2.ans。

【数据规模与约定】

对于 30% 的数据 $n \leq 200$;

对于 60% 的数据, $n \leq 2500$;

对于 100% 的数据, $n \leq 200000$ 。

3. 全民健身(build)

时间限制: 1.000 Sec 内存限制: 128 MB

题目描述

乐乐现在掌管一个大公司，办公楼共有 n 层。为了增加员工的身体素质，他决定在每层楼都建立活动室，活动室分乒乓球和排球两种。

已知每层楼喜欢乒乓球和排球的人数。

每个人的行走楼层数是他去自己喜欢的运动室的楼层数。

请你帮乐乐算算，每层楼应该建乒乓球还是排球，使得所有人行走楼层数总和最小。

输入

第一行一个整数 n ，表示楼层数量。

接下来 n 行，每行两个整数 a 和 b ，表示喜欢乒乓球和排球的人数。

输出

输出一个整数，表示所有人行走楼层总和的最小值。

样例输入

```
2
10 5
4 3
```

样例输出

```
9
```

提示

第一层建乒乓球室，第二层建排球室。行走楼层为 $5 + 4 = 9$

对于20%的数据， n 的范围 $[2, 10]$;

对于30%的数据， n 的范围 $[2, 20]$;

对于50%的数据， n 的范围 $[2, 50]$;

对于70%的数据， n 的范围 $[2, 200]$ ，每层楼喜欢乒乓球和排球的人数范围 $[1, 1000]$;

对于80%的数据， n 的范围 $[2, 500]$;

对于100%的数据， n 的范围 $[2, 4000]$ ，每层楼喜欢乒乓球和排球的人数范围 $[1, 10^5]$;

4. 斗地主

(landlords.cpp/c/pas)

【问题描述】

牛牛最近迷上了一种叫斗地主的扑克游戏。斗地主是一种使用黑桃、红心、梅花、方片的 A 到 K 加上大小王的共 54 张牌来进行的扑克牌游戏。在斗地主中，牌的大小关系根据牌的数码表示如下： $3 < 4 < 5 < 6 < 7 < 8 < 9 < 10 < J < Q < K < A < 2 < \text{小王} < \text{大王}$ ，而花色并不对牌的大小产生影响。每一局游戏中，一副手牌由 n 张牌组成。游戏者每次可以根据规定的牌型进行出牌，首先打光自己的手牌一方取得游戏的胜利。

现在，牛牛只想知道，对于自己的若干组手牌，分别最少需要多少次出牌可以将它们打光。请你帮他解决这个问题。

需要注意的是，本题中游戏者每次可以出牌的牌型与一般的斗地主相似而略有不同。具体规则如下：

牌型	牌型说明	牌型举例照片
火箭	即双王（双鬼牌）。	
炸弹	四张同点牌。如四个 A。	
单张牌	单张牌，比如 3。	
对子牌	两张码数相同的牌。	
三张牌	三张码数相同的牌。	
三带一	三张码数相同的牌 + 一张单牌。例如：三张3+单4	
三带二	三张码数相同的牌 + 一对牌。例如：三张3+对4	
单顺子	五张或更多码数连续的单牌（不包括2点和双王）例如：单 7+单 8+单 9+单 10+单 J。另外，在顺牌（单顺子、双顺子、三顺子）中，牌的花色不要求相同。	
双顺子	三对或更多码数连续的对牌（不包括2点和双王）。例如：对3+对4+对5。	
三顺子	二个或更多码数连续的三张牌（不包括2点和双王）。例如：三张 3+三张 4+三张 5。	
四带二	四张码数相同的牌+任意两张单牌（或任意两对牌）例如：四张 5+单 3+单 8 或 四张 4+对 5+对 7	

【输入格式】

输入文件名为 `landlords.in`。

第一行包含用空格隔开的2个正整数 T, n , 表示手牌的组数以及每组手牌的张数。

接下来 T 组数据, 每组数据 n 行, 每行一个非负整数对 a_i, b_i , 表示一张牌, 其中 a_i 表示牌的数码, b_i 表示牌的花色, 中间用空格隔开。特别的, 我们用 1 来表示数码 A, 11 表示数码 J, 12 表示数码 Q, 13 表示数码 K; 黑桃、红心、梅花、方片分别用 1-4 来表示; 小王的表示方法为 0 1, 大王的表示方法为 0 2。

【输出格式】

输出文件名为 `landlords.out`。

共 T 行, 每行一个整数, 表示打光第 i 组手牌的最少次数。

【输入输出样例 1】

landlords.in	landlords.out
1 8 7 4 8 4 9 1 10 4 11 1 5 1 1 4 1 1	3

见选手目录下的 `landlords/landlords1.in` 与 `landlords/landlords1.ans`。

【输入输出样例 1 说明】

共有 1 组手牌, 包含 8 张牌: 方片7, 方片8, 黑桃9, 方片10, 黑桃J, 黑桃5, 方片A 以及黑桃A。可以通过打单顺子 (方片7, 方片8, 黑桃9, 方片10, 黑桃J), 单张牌 (黑桃5) 以及对子牌 (黑桃A 以及方片A) 在3次内打光。

【输入输出样例 2】

landlords.in	landlords.out
1 17 12 3 4 3 2 3 5 4 10 2 3 3 12 2 0 1 1 3 10 1 6 2	6

12 1	
11 3	
5 2	
12 4	
2 2	
7 2	

见选手目录下的 landlords/landlords2.in 与 landlords/landlords2.ans。

【样例输入输出 3】

见选手目录下的 landlords/landlords3.in 与 landlords/landlords3.ans。

【数据规模与约定】

对于不同的测试点，我们约定手牌组数与张数的规模如下：

测试点编号	T	n	测试点编号	T	n
1	100	2	11	100	14
2	100	2	12	100	15
3	100	3	13	10	16
4	100	3	14	10	17
5	100	4	15	10	18
6	100	4	16	10	19
7	100	10	17	10	20
8	100	11	18	10	21
9	100	12	19	10	22
10	100	13	20	10	23

数据保证：所有的手牌都是随机生成的。