NOI2013河北省队选拔赛

第二试

|  |  |  |  |
| --- | --- | --- | --- |
| 题目名称 | ALO | SAO | 钙铁锌硒维生素 |
| 目录 | alo | sao | ferrous |
| 可执行文件名 | alo | sao | ferrous |
| 输入文件名 | alo.in | sao.in | ferrous.in |
| 输出文件名 | alo.out | sao.out | ferrous.out |
| 每个测试点时限 | 3 秒 | 3 秒 | 2 秒 |
| 内存限制 | 256 MB | 256 MB | 256 MB |
| 测试点数目 | 10 | 20 | 10 |
| 每个测试点分值 | 10 | 5 | 10 |
| 是否有部分分 | 无 | 无 | 无 |
| 题目类型 | 传统型 | 传统型 | 传统型 |

提交源程序须加后缀

|  |  |  |  |
| --- | --- | --- | --- |
| 对于C++ 语言 | alo.cpp | sao.cpp | ferrous.cpp |
| 对于C 语言 | alo.c | sao.c | ferrous.c |
| 对于Pascal语言 | alo.pas | sao.pas | ferrous.pas |

注意：最终测试时，所有编译命令均不打开任何优化开关。

ALO

【问题描述】

Welcome to ALO ( Arithmetic and Logistic Online)。这是一个VR MMORPG，如名字所见，到处充满了数学的谜题。

现在你拥有*n*颗宝石，每颗宝石有一个能量密度，记为*ai*，这些宝石的能量密度两两不同。现在你可以选取连续的一些宝石（必须多于一个）进行融合，设为*ai*, *ai*+1, …, *aj*，则融合而成的宝石的能量密度为这些宝石中能量密度的次大值与其他任意一颗宝石的能量密度按位异或的值，即，设该段宝石能量密度次大值为*k*，则生成的宝石的能量密度为**max**{*k* **xor** *ap* | *ap* ≠ *k* , *i* ≤ *p* ≤ *j*}。

现在你需要知道你怎么选取需要融合的宝石，才能使生成的宝石能量密度最大。

【输入格式】

从文件***alo.in***中读入数据。

第一行，一个整数*n*，表示宝石个数。

第二行，*n*个整数，分别表示*a*1至*an*，表示每颗宝石的能量密度，保证对于*i* ≠ *j*有*ai* ≠ *aj*。

【输出格式】

输出到文件***alo.out***中。

输出一行一个整数，表示最大能生成的宝石能量密度。

【样例输入】

5

9 2 1 4 7

【样例输出】

14

【样例解释】

选择区间[1,5]，最大值为7 **xor** 9。

【数据规模与约定】

对于20%的数据有*n* ≤ 100。

对于50%的数据有*n* ≤ 2000。

对于100%的数据有1 ≤ *n* ≤ 50000, 0 ≤ *ai* ≤ 109。

SAO

【问题描述】

Welcome to SAO ( Strange and Abnormal Online)。这是一个VR MMORPG，含有*n*个关卡。但是，挑战不同关卡的顺序是一个很大的问题。

有*n* – 1个对于挑战关卡的限制，诸如第*i*个关卡必须在第*j*个关卡前挑战，或者完成了第*k*个关卡才能挑战第l个关卡。并且，如果不考虑限制的方向性，那么在这*n* – 1个限制的情况下，任何两个关卡都存在某种程度的关联性。即，我们不能把所有关卡分成两个非空且不相交的子集，使得这两个子集之间没有任何限制。

【输入格式】

从文件***sao.in***中读入数据。

第一行，一个整数*T*，表示数据组数。

对于每组数据，第一行一个整数*n*，表示关卡数。接下来*n* – 1行，每行为“*i* **sign** *j*”，其中0 ≤ *i*, *j* ≤ *n* – 1且*i* ≠ *j*，**sign**为“<”或者“>”，表示第*i*个关卡必须在第*j*个关卡前/后完成。

【输出格式】

输出到文件***sao.out***中。

对于每个数据，输出一行一个整数，为攻克关卡的顺序方案个数，**mod** 1,000,000,007输出。

【样例输入】

2

5

0 < 2

1 < 2

2 < 3

2 < 4

4

0 < 1

0 < 2

0 < 3

【样例输出】

4

6

【数据规模与约定】

对于20%的数据有*n* ≤ 10。

对于40%的数据有*n* ≤ 100。

对于另外20%的数据有，保证数据中**sign**只会是<，并且*i* < *j*。

对于100%的数据有*T* ≤ 5，1 ≤ *n* ≤ 1000。

钙铁锌硒维生素

【问题描述】

银河队选手名单出来了！小林，作为特聘的营养师，将负责银河队选手参加宇宙比赛的饮食。

众所周知，前往宇宙的某个星球，通常要花费好长好长的时间，人体情况在这之间会发生变化，因此，需要根据每天的情况搭配伙食，来保证营养。

小林把人体需要的营养分成了*n*种，这些营养包括但不限于铁，钙。他准备了2套厨师机器人，一套厨师机器人有*n*个，每个厨师机器人只会做一道菜，这道菜一斤能提供第*i*种营养*xi*微克。想要吃这道菜的时候，只要输入一个数，就能吃到对应数量的这道菜了。为防止摄入过量对身体造成的伤害，每个机器人还有防过量摄入药，只要输入一个数，就能生成一定剂量的药，吃了这些药，就能减少相当于食用对应数目的这道菜提供的营养。

小林之所以准备2套厨师机器人，正是因为旅途漫漫，难以预计，也许某一个厨师机器人在途中坏掉，要是影响了银河队选手的身体，就不好了。因此，第2套厨师机器人被用来做第1套的备用。小林需要为每一个第1套厨师机器人选一个第2套厨师机器人作备份，使得当这个机器人坏掉时，用备份顶替，整套厨师机器人仍然能搭配出任何营养需求，而且，每个第2套厨师机器人只能当一个第1套厨师机器人的备份。

【输入格式】

从文件***ferrous.in***中读入数据。

输入文件的第一行包含一个正整数*n*。接下来*n*行，每行*n*个整数，表示第1套厨师机器人做的菜每一斤提供的每种营养。再接下来*n*行，每行*n*个整数，表示第2套厨师机器人做的菜每一斤提供的每种营养。

【输出格式】

输出到文件***ferrous.out***中。

输出文件的第一行是一个字符串，如果无法完成任务，输出“NIE”，否则输出“TAK”，并跟着*n*行，第*i*行表示第*i*个第1套机器人的备份是哪一个第2套机器人。为了避免麻烦，如果有多种可能的答案，请给出字典序最小的那一组。

【样例输入1】

3

1 0 0

0 1 0

0 0 1

2 3 0

0 7 8

0 0 9

【样例输出1】

TAK

1

2

3

【样例输入2】

4

2 0 1 3

0 1 0 4

1 9 9 4

0 4 1 7

1 10 9 8

1 13 10 11

1 2 1 5

2 1 1 7

【样例输出2】

TAK

3

4

1

2

【数据规模与约定】

对于10%的数据，*n* = 2；

对于20%的数据，*n* ≤ 10；

对于40%的数据，*n* ≤ 30；

对于60%的数据，*n* ≤ 50；

对于80%的数据，*n* ≤ 100；

对于100%的数据，1 ≤ *n* ≤ 300，所有出现的整数均非负，且不超过10,000。