

P0

—rqy

2018.4.13

**(请选手务必仔细阅读本页内容)**

题目名称	数学	生物	物理
源文件名	physics	math	biology
输入文件名	physics.in	math.in	biology.in
输出文件名	physics.out	math.out	biology.out
时间限制	1 秒	1 秒	1 秒
内存限制	512M	512M	512M
测试点数目	10	10	10
测试点分值	10	10	10
比较方式	全文比较（过滤行末空格及文末回车）		Special Judge

**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()` 的返回类型必须是 `int`，程序正常结束时的返回值必须是 `0`。
3. 特别提醒：评测在 Linux 下进行。如有 Linux 特定问题请及时申诉。（文件名错误等申诉不予受理）。

# 1 数学

(physics.cpp/c/pas)

——人生如棋，落子无悔。

## 1.1 问题描述

小 A 正在考数学。然而由于小 A 那么神，她早已 AK 了这次考试。

看着一擦一擦的不等式，她突然想下棋（小 A 的想法当然是与常人不同的）。

小 A 构造了一棵树，准备在树上下棋。小 A 把数学题当做棋子一个一个扔到了树上，恰好一个结点扔一个。

这时，她突然发现一个问题：两个相邻的数学题会发生作用，产生一个新的数学题。但是小 A 不想让数学题越来越多，于是她决定摘下来一些题。

但是她又很喜欢数学，想要让留下来的题目尽量多。

这问题对小 A 当然不算什么，但是她想考考你。

具体地说，你需要找出一些结点，它们互不相邻，又尽量多。

小 A 发现这对你不算什么，于是高兴地提出了自己的另一个问题：

小 A 更喜欢题号小的题目（题号和结点编号对应），于是她想在留下来的题目尽量多的情况下，尽量留下第 1 道题；在这个基础上尽量留下第 2 道题等等。

具体地说，你需要找到字典序最小的一组解。

## 1.2 输入

第一行两个数  $n, type$ ，表示题目的个数，以及问题种类（见输出格式）

接下来  $n - 1$  行，每行两个数  $u, v$ ，表示树上的一条连接  $u$  和  $v$  的边。

## 1.3 输出

第一行一个数  $ans$ ，表示最多可以剩下多少道题目。

如果  $type = 1$ ，第二行输出  $ans$  个数，表示所有可行解中字典序最小的一组（显然这组解是从小到大输出的）。

## 1.4 输入输出样例 1

### 1.4.1 输入样例

```
5 0
1 2
1 3
2 5
3 4
```

### 1.4.2 输出样例

3

## 1.5 输入输出样例 2

### 1.5.1 输入样例

4 1

1 2

2 3

3 4

### 1.5.2 输出样例

2

1 3

## 1.6 样例解释

样例 1 中，选择 1,4,5 留下来最优。

样例 2 中，选择 1,3 或 2,4 都可以使留下的题目数最大，此时选择字典序更小的 1,3 。

## 1.7 约定和数据范围

数据编号	$n$	$type$
1	$\leq 5$	0
2	$\leq 20$	0
3,4,5,6	$\leq 100000$	0
7,8,9,10	$\leq 1000$	1

## 2 生物

(math.cpp/c/pas)

——为什么会变成这样呢... 第一次选择了生物，第一次应用了概率论。

两件原本快乐的事情结合在一起，本该得到的，是双重的快乐...

可是为什么，为什么会变成这样呢...

### 2.1 问题描述

小 B 看着面前的生物试卷，不禁皱起了眉头。他没有小 A 强大的计算能力，面对复杂的哥德尔遗传定律相关计算只能选择弃疗。

这让他想起了初中学生物时快乐的时光。那时候，学生物不需要大量的计算，只需要数一数就够了...

他仿佛回到了初中，面前的试卷上画着一个大大的食物网。他仿佛能感觉到，这庞大的食物网上各个物种的相互依赖关系。草如果灭绝，羊就会死掉，紧接着所有动物都会死掉。但是也有些生物没有那么重要，例如即使羊灭绝了，狼也可以通过吃兔子维持生存。

他突然想知道，每个物种有多么重要。具体地说，他想知道，对于每个物种，如果这个物种由于某种神秘原因灭绝，一共会有多少物种消失。（除了这个由于神秘原因灭绝的物种之外，一个物种只会在它所捕食的所有物种都灭绝之后才会灭绝。本题中不考虑由于不被捕食泛滥成灾导致生物圈被破坏的情况等。）

### 2.2 输入

第一行两个数和一个字符  $n, m, C$ ，分别表示食物网上的物种数量和捕食关系的数量，还有数据类型（详见数据范围）。

接下来  $m$  行，每行两个数  $u, v$  表示一个捕食关系，即  $u$  吃  $v$ 。

如果输入中某物种什么都不吃，那它就是生产者，不会灭绝（除非它是第一个被灭绝的）。

### 2.3 输出

一行  $n$  个数，表示每个物种灭绝时会有多少物种灭绝。

### 2.4 输入输出样例

#### 2.4.1 输入样例

```
5 5 C
2 1
5 2
4 3
3 1
4 2
```

### 2.4.2 输出样例

5 2 1 1 1

## 2.5 样例解释

我们把 1 2 3 4 5 分别看做草、羊、牛、人、狼。

如果草灭绝，由于生产者没了，所有物种都会灭绝。

如果羊灭绝，狼没东西吃了，会灭绝；人还可以吃牛存活。

如果牛灭绝，那只有它自己会灭绝。狼本来就不吃牛，人还可以吃羊生存。

如果人或者狼灭绝，由于本来就没有其它物种吃它们，所以不会产生什么影响。

## 2.6 约定和数据范围

保证不会出现吃出一个环的形态，即 A 吃 B，B 吃 C ... X 吃 A 这种情况。

数据编号	$n, m$	数据类型
1,2	$\leq 1000$	C
3,4	$\leq 100000$	A
5,6	$\leq 100000$	B
7,8,9,10	$\leq 100000$	C

数据类型解释：

- A. 1 号物种为生产者，其它所有物种只会捕食一种物种。
- B. 除 1 号物种为生产者外，第  $i$  号物种只可能捕食第  $i-1, i-2, i-3$  中的 1 到 3 个。
- C. 无特殊性质。

## 3 物理

(biology.cpp/c/pas)

——总有一天，我会让这里座无虚席！

### 3.1 问题描述

小 C 看着面前人满为患的物理教室，突然想起了自己几年前所说的话。

那时台下只有 5 个人听自己讲课，现在却已经坐满了。

小 C 拿起粉笔，不由得又回忆起当初那段快乐的时光。

那时候，由于人少，小 C 得以拿出很多时间来研究他最喜欢的游戏——数独。

但是现在，小 C 花费了太多时间帮大家解题，已经没有业余时间了。

但他还是对一些还没解出数独耿耿于怀。你能帮帮他吗？

### 3.2 输入

一行一个数  $n$ ，表示数独的阶数（ $n = 3$  时数独是  $9 * 9$  的， $n = 4$  时数独是  $16 * 16$  的）

接下来一个  $n^2 \times n^2$  的矩阵，表示一个未完成的数独。未填的格子用 0 表示，已填的用 123456789(ABCDEFGH) 表示。

### 3.3 输出

一个填好了的数独。如果有多解，输出任意解（有 Special Judge）。

如果无解，输出全 0  $n^2 \times n^2$  矩阵（即使输入时已经填了一些数）。

### 3.4 输入输出样例 1

#### 3.4.1 输入样例

```
3
8 0 0 0 0 0 0 0 0
0 0 3 6 0 0 0 0 0
0 7 0 0 9 0 2 0 0
0 5 0 0 0 7 0 0 0
0 0 0 0 4 5 7 0 0
0 0 0 1 0 0 0 3 0
0 0 1 0 0 0 0 6 8
0 0 8 5 0 0 0 1 0
0 9 0 0 0 0 4 0 0
```

### 3.4.2 输出样例

```
8 1 2 7 5 3 6 4 9
9 4 3 6 8 2 1 7 5
6 7 5 4 9 1 2 8 3
1 5 4 2 3 7 8 9 6
3 6 9 8 4 5 7 2 1
2 8 7 1 6 9 5 3 4
5 2 1 9 7 4 3 6 8
4 3 8 5 2 6 9 1 7
7 9 6 3 1 8 4 5 2
```

## 3.5 输入输出样例 2

### 3.5.1 输入样例

```
4
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
... // 共 16 行，每行都是 16 个 0
```

### 3.5.2 输出样例

```
F E G 9 7 3 8 D 5 C 4 6 A B 1 2
D 5 8 B 9 A 4 F 7 G 2 1 E C 3 6
7 3 4 6 E C 2 1 A B 8 D G 5 F 9
A C 2 1 6 G B 5 F E 9 3 7 8 D 4
6 B D 7 F E 9 C 4 5 G 8 2 1 A 3
E G 5 F 4 8 6 3 2 D 1 A B 9 C 7
4 8 C 3 2 D 1 A B 9 F 7 6 G E 5
2 9 1 A B 5 G 7 6 3 C E 8 D 4 F
C D F G A 7 E 6 3 4 5 9 1 2 8 B
8 7 9 5 3 4 F B G 1 D 2 C E 6 A
3 4 6 E D 1 5 2 C 8 A B F 7 9 G
B 1 A 2 8 9 C G E 6 7 F 3 4 5 D
9 A B C 5 6 7 8 1 2 3 4 D F G E
5 6 7 8 1 2 3 4 D F E G 9 A B C
1 2 3 4 G F D E 9 A B C 5 6 7 8
G F E D C B A 9 8 7 6 5 4 3 2 1
```

### 3.6 输入输出样例 3

#### 3.6.1 输入样例

```
3
1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
```

#### 3.6.2 输出样例

```
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
```

### 3.7 样例解释

样例 2 答案不唯一（显然）。

### 3.8 约定和数据范围

60% 的数据  $n = 3$  。

40% 的数据  $n = 4$  。

不保证数据随机。

数据有梯度。