

NOI2012 浙江省队选拔赛

第二试

竞赛时间：2012 年 5 月 26 日 8:00–13:00

题目名称	灾难	网络	波浪
目录	catas	network	wavel
可执行文件名	catas	network	wavel
输入文件名	catas.in	network.in	wavel.in
输出文件名	catas.out	network.out	wavel.out
每个测试点时限	1 秒	3 秒	10 秒
测试点数目	10	10	10
每个测试点分值	20	20	20
是否有部分分	否	否	否
题目类型	传统	传统	传统
附加文件	N/A	N/A	N/A

提交源程序须加后缀

对于 Pascal 语言	catas.pas	network.pas	wavel.pas
对于 C 语言	catas.c	network.c	wavel.c
对于 C++ 语言	catas.cpp	network.cpp	wavel.cpp

注意：最终测试时，所有编译命令均不打开任何优化开关。

灾难

【问题描述】

阿米巴是小强的好朋友。

阿米巴和小强在草原上捉蚂蚱。小强突然想，如果蚂蚱被他们捉灭绝了，那么吃蚂蚱的小鸟就会饿死，而捕食小鸟的猛禽也会跟着灭绝，从而引发一系列的生态灾难。

学过生物的阿米巴告诉小强，草原是一个极其稳定的生态系统。如果蚂蚱灭绝了，小鸟照样可以吃别的虫子，所以一个物种的灭绝并不一定会引发重大的灾难。

我们现在从专业一点的角度来看这个问题。我们用一种叫做食物网的有向图来描述生物之间的关系：

一个食物网有 N 个点，代表 N 种生物，如果生物 x 可以吃生物 y ，那么从 y 向 x 连一个有向边。

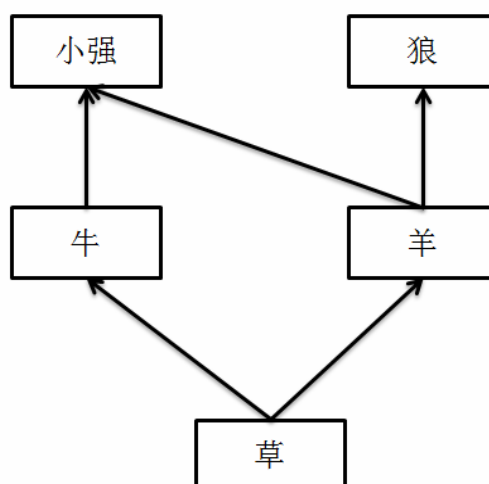
这个图没有环。

图中有一些点没有连出边，这些点代表的生物都是生产者，可以通过光合作用来生存；而有连出边的点代表的都是消费者，它们必须通过吃其他生物来生存。

如果某个消费者的所有食物都灭绝了，它会跟着灭绝。

我们定义一个生物在食物网中的“灾难值”为，如果它突然灭绝，那么会跟着一起灭绝的生物的种数。

举个例子：在一个草场上，生物之间的关系是：



如果小强和阿米巴把草原上所有的羊都给吓死了，那么狼会因为没有食物而灭绝，而小强和阿米巴可以通过吃牛、牛可以通过吃草来生存下去。所以，羊的灾难值是 1。但是，如果草突然灭绝，那么整个草原上的 5 种生物都无法幸免，所以，草的灾难值是 4。

给定一个食物网，你要求出每个生物的灾难值。

【输入格式】

输入文件 `catas.in` 的第一行是一个正整数 N ，表示生物的种数。生物从 1 标号到 N 。

接下来 N 行，每行描述了一个生物可以吃的其他生物的列表，格式为用空格隔开的若干个数字，每个数字表示一种生物的标号，最后一个数字是 0 表示列表的结束。

【输出格式】

输出文件 `catas.out` 包含 N 行，每行一个整数，表示每个生物的灾难值。

【样例输入】

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

【样例输出】

```
4
1
0
0
0
```

【样例说明】

样例输入描述了题目描述中举的例子。

【数据规模】

对 50% 的数据， $N \leq 10000$ 。

对 100% 的数据， $1 \leq N \leq 65534$ 。

输入文件的大小不超过 1M。保证输入的食物网没有环。

网络

【问题描述】

有一个无向图 G ，每个点有个权值，每条边有一个颜色。这个无向图满足以下两个条件：

1. 对于任意节点连出去的边中，相同颜色的边 不超过两条。
2. 图中 不存在同色的环，同色的环指相同颜色的边构成的环。

在这个图上，你要支持以下三种操作：

0. 修改一个节点的权值。
 1. 修改一条边的颜色。
 2. 查询由颜色 c 的边构成的图中，所有可能在节点 u 到节点 v 之间的 简单路径 上的节点的权值的最大值。

【输入格式】

输入文件 `network.in` 的第一行包含四个正整数 N, M, C, K ，其中 N 为节点个数， M 为边数， C 为边的颜色数， K 为操作数。

接下来 N 行，每行一个正整数 v_i ，为节点 i 的权值。

之后 M 行，每行三个正整数 u, v, w ，为一条连接节点 u 和节点 v 的边，颜色为 w 。满足 $1 \leq u, v \leq N, 0 \leq w < C$ ，保证 $u \neq v$ ，且 任意两个节点之间最多存在一条边(无论颜色)。

最后 K 行，每行表示一个操作。每行的第一个整数 k 表示操作类型。

0. $k = 0$ 为修改节点权值操作，之后两个正整数 x 和 y ，表示将节点 x 的权值 v_x 修改为 y 。
1. $k = 1$ 为修改边的颜色操作，之后三个正整数 u, v 和 w ，表示将连接节点 u 和节点 v 的边的颜色修改为颜色 w 。满足 $0 \leq w < C$ 。
2. $k = 2$ 为查询操作，之后三个正整数 c, u 和 v ，表示查询所有可能在节点 u 到节点 v 之间的由颜色 c 构成的简单路径上的节点的权值的最大值。如果不存在 u 和 v 之间不存在由颜色 c 构成的路径，那么输出“-1”。

【输出格式】

输出文件 `network.out` 包含若干行，每行输出一个对应的信息。

1. 对于修改节点权值操作，不需要输出信息。
2. 对于修改边的颜色操作，按以下几类输出：
 - a) 若不存在连接节点 u 和节点 v 的边，输出“No such edge.”。
 - b) 若修改后不满足条件 1，不修改边的颜色，并输出“Error 1.”。
 - c) 若修改后不满足条件 2，不修改边的颜色，并输出“Error 2.”。
 - d) 其他情况，成功修改边的颜色，并输出“Success.”。
 输出满足条件的第一条信息即可，即若同时满足 b 和 c ，则只需要输出“Error 1.”。
3. 对于查询操作，直接输出一个整数。

【样例输入】

```

4 5 2 7
1
2
3
4
1 2 0
1 3 1
2 3 0
2 4 1
3 4 0
2 0 1 4
1 1 2 1
1 4 3 1
2 0 1 4
1 2 3 1
0 2 5
2 1 1 4

```

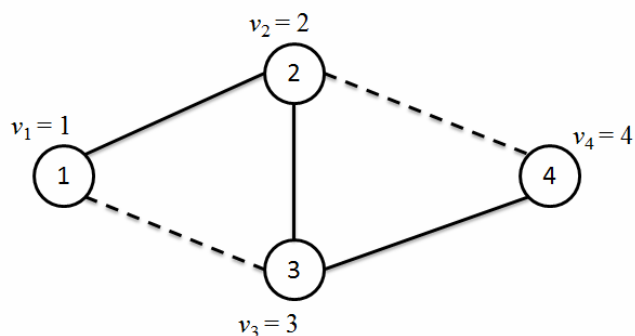
【样例输出】

```

4
Success.
Error 2.
-1
Error 1.
5

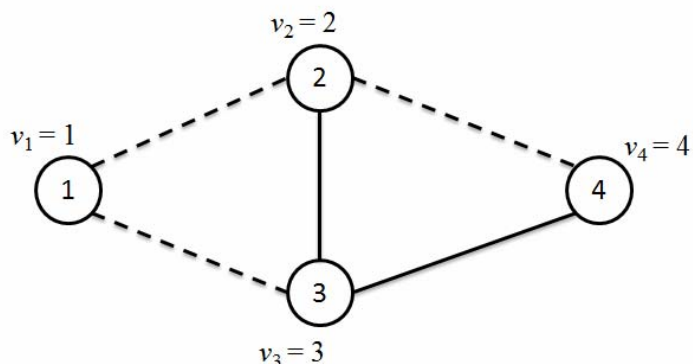
```

【样例说明】



颜色 0 为实线的边，颜色 1 为虚线的边，
 由颜色 0 构成的从节点 1 到节点 4 的路径有 1 - 2 - 4，故 $\max\{v_1, v_2, v_4\} = \max\{1, 2, 4\} = 4$ 。

将连接节点 1 和节点 2 的边修改为颜色 1，修改成功，输出“Success.”

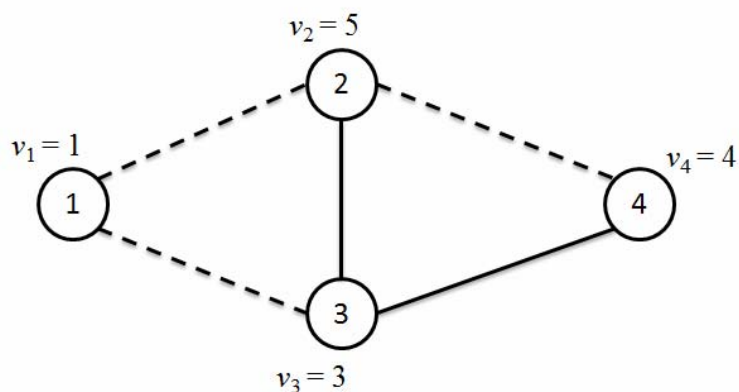


将连接节点 4 和节点 3 的边修改为颜色 1，由于修改后会使得存在由颜色 1 构成的环(1-2-4-3-1)，不满足条件 2，故不修改，并输出“Error 2”。

不存在颜色 0 构成的从节点 1 到节点 4 的边，输出“-1”。

将连接节点 2 和节点 3 的边修改为颜色 1，由于修改后节点 2 的连出去的颜色为 1 的边有 3 条，故不满足条件 1，故不修改，并输出“Error 1”。

将节点 2 的权值修改为 5。



由颜色 1 构成的从节点 1 到节点 4 的路径有 1-2-4，故 $\max\{v_1, v_2, v_4\} = \max\{1, 5, 4\} = 5$ 。

【数据规模】

对于 30% 的数据： $N \leq 1000$ ， $M \leq 10000$ ， $C \leq 10$ ， $K \leq 1000$ 。

另有 20% 的数据： $N \leq 10000$ ， $M \leq 100000$ ， $C = 1$ ， $K \leq 100000$ 。

对于 100% 的数据： $N \leq 10000$ ， $M \leq 100000$ ， $C \leq 10$ ， $K \leq 100000$ 。

波浪

【问题描述】

阿米巴和小强是好朋友。

阿米巴和小强在大海旁边看海水的波涛。小强第一次面对如此汹涌的海潮，他兴奋地叫个不停。而阿米巴则很淡定，他回想起曾经的那些日子，事业的起伏，情感的挫折……总之今天的风浪和曾经经历的那些风雨比起来，简直什么都不算。

于是，这对好朋友不可避免地产生了分歧。为了论证自己的观点，小强建立了一个模型。他海面抽象成一个 1 到 N 的排列 $P[1\dots N]$ 。定义波动强度等于 相邻两项的差的绝对值的和，即：

$$L = |P_2 - P_1| + |P_3 - P_2| + \dots + |P_N - P_{N-1}|$$

给你一个 N 和 M ，问：随机一个 $1\dots N$ 的排列，它的波动强度 不小于 M 的概率有多大？

答案请保留 小数点后 K 位输出，四舍五入。

【输入格式】

输入文件 `wavel.in` 的第一行包含三个整数 N , M 和 K ，分别表示排列的长度，波动强度，输出位数。

【输出格式】

输出文件 `wavel.out` 包含一个小数点后 K 位的实数。

【样例输入】

```
3 3 3
```

【样例输出】

```
0.667
```

【样例说明】

$N=3$ 的排列有 6 个：123, 132, 213, 231, 312, 321；他们的波动强度分别为 2, 3, 3, 3, 3, 2。所以，波动强度不小于 3 的概率是 $4/6$ ，即 0.667。

你也可以通过下面的代码来验证这个概率：

```
int a[3]={0,1,2},s=0,n=3;
for (int i=0;i<1000000;i++){
    random_shuffle(a,a+n);
    int t=0;
    for (int j=0;j<n-1;j++) t+=abs(a[j+1]-a[j]);
```

```
    if (t>=3) s++;  
  }  
  printf("%.3f\n",s/1000000.0);
```

【数据规模】

对于 30%的数据， $N \leq 10$ 。

对于另外 30%的数据， $K \leq 3$ 。

对于另外 30%的数据， $K \leq 8$ 。

对于另外 10%的数据， $N \leq 50$ 。

对于 100%的数据， $N \leq 100$ ， $K \leq 30$ ， $0 \leq M \leq 2147483647$ 。