

# 赛拟模 PSC 心良的 86723zyx

比赛时间：2019 年 10 月 05 日 14:00~17:30

## 【试题一览】

题目名称	一道图论神题	数组异或	侦探游戏	天上掉馅饼
提交程序名	god	xorarray	detective	bonus
输入文件	god.in	xorarray.in	detective.in	bonus.in
输出文件	god.out	xorarray.out	detective.out	bonus.out
每个测试点时限	1s	2s	1s	1s
内存限制	128MB	256MB	256MB	128MB
测试点数目	10	10	10	10
每个测试点分值	100	10	10	10
是否有 SPJ	否	否	否	否
题目类型	传统型	传统型	传统型	传统型

## 【注意事项】

1. 由于出题人水平有限，今天的题目难度过于低，出题人没有区分题目难度的能力，不保证题目难度呈升序排列。
2. 今天的题目过于水，请大家不要讨论，不要假，尽快 AK。
3. AK 后请不要 D 出题人，没 AK 更不要 D 出题人。
4. 最终测试时，开启 O2 优化。
5. 考试全程请不要使用搜索引擎。
6. 请独立完成考试。

## 一道图论神题 (god,1s,128MB)

### 【题目描述】

LYK 有一张无向图  $G=\{V,E\}$ , 这张无向图有  $n$  个点  $m$  条边组成。并且这是一张带权图, 只有点权。

LYK 想把这个图删干净, 它的方法是这样的。每次选择一个点, 将它删掉, 但删这个点是需要代价的。假设与这个点相连的还没被删掉的点是  $u_1, u_2, \dots, u_k$ 。LYK 将会增加  $a[u_1], a[u_2], \dots, a[u_k]$  的疲劳值。

它想将所有点都删掉, 并且删完后自己的疲劳值之和最小。你能帮帮它吗?

### 【输入格式】

第一行两个数  $n, m$  表示一张  $n$  个点  $m$  条边的图。

第二行  $n$  个数  $a_i$  表示点权。

接下来  $m$  行每行三个数  $u, v$ , 表示有一条连接  $u, v$  的边。数据保证任意两个点之间最多一条边相连, 并且不存在自环。

### 【输出格式】

你需要输出这个最小疲劳值是多少。

### 【样例输入 1】

```
4 3
10 20 30 40
1 4
1 2
2 3
```

### 【样例输出 1】

```
40
```

### 【样例解释 1】

一个合理的方法是先删 4 号点, 此时有 10 点疲劳值。接下来删 3 号点, 获得 20 点疲劳值, 再删 2 号点, 获得 10 点疲劳值, 最后删 1 号点, 没有疲劳值。总计 40 点疲劳值。

### 【样例输入输出 2】

见下发文件中的/sample/god.in 和 god.out。

### 【数据规模与约定】

对于 30% 的数据  $n \leq 10$ 。

对于 60% 的数据  $n, m \leq 1000$ 。

对于 100% 的数据  $1 \leq n, m, a_i \leq 100000$ 。

## 数组异或 (xorarray,2s,256MB)

### 【题目描述】

xor——异或，和 and 与 or 一样，是一种重要的逻辑运算，他的运算规律是  $0 \text{ xor } 0=0$ ,  $1 \text{ xor } 1=0$ ,  $1 \text{ xor } 0=1$ ,  $0 \text{ xor } 1=1$ 。

两个整数之间的异或是将两个整数转化成二进制，对他们的每一位分别进行 xor 操作，例：  $6(110) \text{ xor } 13(1101) = 11(1011)$

现在我们要介绍一种新的操作——数组异或，将两个相同大小（假设都为  $n$ ）的数组 A、B 异或成一个新的数组 C，则新数组必满足：

$$C[k] = \sum_{i=1}^k \sum_{j=1}^k A[i] \text{ xor } B[j]$$

现在给你数组大小  $n$ ，和两个数组 A、B

求他们的异或数组 C

由于最终答案可能过大，你需要对 C 的每个元素对  $10^9+7$  取模

### 【输入格式】

一共 3 行。

第一行一个正整数  $N$ 。

接下来两行每行  $N$  个正整数，表示数组 A、B。

### 【输出格式】

一共 1 行， $N$  个正整数，表示数组 C。

### 【样例输入 1】

7

20670 1316 25227 8316 21095 28379 25235

19745 6535 14486 5460 15690 1796 12403

### 【样例输出 1】

7583 52096 161325 276944 453024 675974 958287

### 【样例输入输出 2】

见下发文件中的/sample/xorarray.in 和 xorarray.out。

### 【数据规模与约定】

对于 50%的数据， $N \leq 100$ 。

对于全部的数据， $N \leq 10^5$ 。

## 侦探游戏 (detective,1s,256MB)

### 【题目描述】

小 W 最近沉迷一个侦探游戏，在这个游戏中会不断出现命案，而小 W 作为主角，需要不断地收集各种关键证据，只有当所有的关键证据都被找到，你才能驳倒所有人错误的判断，找出真正的凶手。

一共有  $N$  个关键证据以及  $M$  条信息，每条信息如下所示：如果你已经掌握了证据  $i$ ，那么你可以通过  $k$  个时间的搜索和推理得到证据  $j$ ，同样的，如果你掌握了证据  $j$  你也可以通过  $k$  个时间得到证据  $i$ 。

游戏开始时玩家通过初步观察现场已经得到了证据 1，于此同时，每个玩家在游戏开始阶段时都能获得一个特殊技能来加快游戏进度，增加趣味性。小 W 选了一个他以前从来没用过的技能:好运。这是一个被动技能，系统会在游戏开始时选定一对证据  $(a,b)$  ( $a \neq b$ ) 当小 W 发现其中一个证据的时候，他会很好运地立即获得另外一个证据（不计入时间）。

但是这个技能是完全随机的，小 W 完全不知道进入游戏后系统会挑选哪一对证据，他希望你能帮助他算出他花在本轮游戏上的时间的期望值，这样他心里能有点 B 数。

提供的信息保证:  $i$  不会等于  $j$ ，每个  $k$  值都互不相同， $N$  个证据都能被得到。

### 【输入格式】

一共  $M+1$  行。

第一行两个正整数  $N$  和  $M$ ，表示证据数量和信息数量。

接下来  $M$  行，每行三个数字  $i,j,k$  表示一个信息

### 【输出格式】

一共 1 行，1 个整数（期望值是实数，但这里请直接保留 2 位小数输出）。

### 【样例输入 1】

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

### 【样例输出 1】

```
2.33
```

### 【样例输入输出 2】

见下发文件中的/sample/detective.in 和 detective.out。

### 【数据规模与约定】

对于 20% 的数据， $N \leq 100$

对于 60% 的数据， $N \leq 1000$

对于全部的数据， $N \leq 20000$ ， $M \leq 10^5$ ， $1 \leq k \leq 10^6$ 。

## 天上掉馅饼 (bonus,1s,128MB)

### 【题目描述】

小 G 进入了一个神奇的世界，在这个世界，天上会掉下一些馅饼。今天，天上会随机掉下  $k$  个馅饼。

每次天上掉下馅饼，小 G 可以选择吃或者不吃（必须在下一个馅饼掉下来之前作出选择，并且现在决定不吃的话以后也不能吃）。

馅饼有  $n$  种不同的馅，根据物理定律，天上掉下这  $n$  种馅饼的概率相同且相互独立。然而，每一种馅饼  $i$  都有一个前提馅饼集合  $S_i$ 。只有当  $S_i$  中的馅饼都吃过之后，才能吃第  $i$  种馅饼。比如说，韭菜馅饼的  $S$  中有白菜猪肉馅饼和鲜虾馅饼，那么小 G 只有在吃过白菜猪肉馅饼和鲜虾馅饼之后，才能吃韭菜馅的馅饼。

同时，每个馅饼还有一个美味值  $P_i$ 。今天一天小 G 的幸福度，等于小 G 吃到的所有馅饼的美味值之和。注意： $P_i$  可能是负数。

现在考虑，采用最优策略的前提下，小 G 这一天期望的幸福度是多少？

### 【输入格式】

第一行两个正整数  $k$  和  $n$ ，表示馅饼的数量和种类。

以下  $n$  行，每行若干个数，描述一种馅饼。其中第一个数代表美味值，随后的整数表示该馅饼的前提馅饼，以 0 结尾。

### 【输出格式】

输出一个实数，保留 6 位小数，即在最优策略下期望的幸福度。

### 【样例输入 1】

```
1 2
1 0
2 0
1 2 0 2
```

### 【样例输出 1】

```
1.500000
```

### 【样例输入输出 2】

见下发文件中的/sample/bonus.in 和 bonus.out。

### 【数据范围】

对于 20% 的数据，所有的馅饼都没有“前提馅饼”。

对于 50% 的数据， $1 \leq k \leq 10$ ， $1 \leq n \leq 10$ 。

对于 100% 的数据， $1 \leq k \leq 100$ ， $1 \leq n \leq 15$ ，美味度为  $[-10^6, 10^6]$  的整数。