

WSY 02

| 题目名称 | 奇怪题 | 博弈题 | 政治题 | 地理题 |
|---------|----------|--------------|------------|------------|
| 题目类型 | 传统型 | 传统型 | 传统型 | 传统型 |
| 英文题目名称 | cube | treegame | masses | travel |
| 输入文件名 | cube.in | treegame.in | masses.in | travel.in |
| 输出文件名 | cube.out | treegame.out | masses.out | travel.out |
| 每个测试点时限 | 1s | 1s | 1.5s | 1s |
| 内存限制 | 256MB | 256MB | 256MB | 256MB |
| 提交的源文件名 | cube.cpp | treegame.cpp | masses.cpp | travel.cpp |

【C++编译选项】 `-lm -std=c++14 -w],--stack=1000000000 -o2`

【试题下载地址】 `ftp://172.16.2.202/竞赛资料/20231114.zip` 匿名访问即可

【结果上传地址】 `ftp://172.16.2.202/20231114`文件回收 用户名密码均为test

【赛后补题地址】 `http://zhb.wms.edu/d/JH2023/`

【提交文件夹格式】

--准考证号\ (平时训练用中文姓名)

```
--task1\  
|          |--task1.cpp  
--task2\  
|          |--task2.cpp  
--task3\  
|          |--task3.cpp  
--task4\  
|          |--task4.cpp
```

T1【题目名称】

奇怪题 (cube)

题目背景

新学期有很多工人加入公司。
出于人道主义考虑， PY 决定为工人们建造住所。

题目描述

PY 非常喜欢方形 千篇一律的 样子。
他给出了一张工人住所的分配表 来让你判断它方不方。
 PY 对方的定义如下：所有可供分配的住所位于一个三维空间内，且每个住所的坐标 (x, y, z) 满足 $x, y, z \in \mathbb{N}_+$ （即坐标都为正整数）。若所有被分配的住所组成了一个每条棱与坐标轴平行的 ***实心***长方体（即该长方体内的住所 ***全部被分配***），则称之为方。

输入格式

第一行给出一个数 n 代表工人个数。
接下来有 n 行 每行有三个数 x, y, z 代表 该工人住所的坐标。
保证分配的住所不重复。

输出格式

若它方 输出 YES 否则 输出 NO。

样例数据

输入样例1

```
8
1 1 1
1 1 2
1 2 1
1 2 2
2 1 1
2 1 2
2 2 1
2 2 2
```

输出样例1

```
YES
```

数据范围

$$n \leq 10^6, 1 \leq x, y, z \leq 2^{63} - 1.$$

$$2^{63} - 1 = 9223372036854775807.$$

因为某些原因，你只有答对所有测试点才能得到分数。

T2【题目名称】

博弈题 (treegame)

题目背景

一天, HH 和 PY 约好下棋, 但却不会下, 就向蒟蒻们问怎么下, 但蒟蒻们只会下五子棋, 不会下高级的棋

于是他们又希望你来解决这个问题

题目描述

这种棋是很奇怪的, 它的规则如下:

- 只有一个棋子
- 棋是在一棵**无根树**上下的
- 棋一开始可以在树上的任意一个节点处
- 黑白两方棋手轮流移动棋, 且每次只能把棋移到相邻的节点上
- 不能把棋移到棋已经移到过的地方
- 若一方无法继续移动棋, 则无法继续移动棋的那方输

输入格式

第一行给出 2 个数 n, k 分别代表树的节点数和问题编号

接下来 $n - 1$ 行, 每行 2 个数, 代表这两个节点相连

保证 $1 \leq n \leq 10^6$

输出格式

问题一: 从小到大输出***所有***先手必胜的节点的编号

问题二: 按节点编号大小从小到大输出***所有***先手必胜的节点的***每步***双方***都是最优移法***的走法数

对于两个问题, 若没有先手必胜的节点, 输出 NO!

输入输出样例

输入样例1

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

输出样例1

```
1 4
```

输入样例2

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

输出样例2

```
2 2
```

输入样例3

```
12 1
1 2
1 3
3 4
1 5
1 6
5 7
2 8
4 9
3 10
1 11
8 12
```

输出样例3

```
1 3 4 5 7 8 9 10
```

输入样例4

```
12 2
1 2
1 3
3 4
1 5
1 6
5 7
2 8
4 9
3 10
1 11
8 12
```

输出样例4

3 1 1 1 3 3 1 4

数据范围与提示

请快读快写!!!!!!!!!!!!

| Subtask | 问题 | 总分 |
|-------------------|-----|----|
| Subtask #1(1-16) | 问题一 | 50 |
| Subtask #2(17-32) | 问题二 | 50 |

对于 50% 的数据 $n \leq 4 \times 10^3$

T3 【题目名称】

政治题 (masses)

题目背景

众所周知，海吃帝国虽然是帝国，但却实行人民民主专政的方针。

题目描述

现在根据专家们的分析结果，海吃帝国的人民群众具有以下性质：

1. 第 0 天的领导人是严格随机的
2. 在对一个领导人不满意后，他们会选举出与他的政策冲突的另一个领导人
3. 与上一领导人的政策冲突的所有领导人被选到的概率相同
4. 由于他们 喜新厌旧，他们会以稳定的概率决定领导人是否更换
5. 由于他们 喜新厌旧 到了一定地步，于是他们不会选上一次更换的领导人（上一次更换的不一定前天的）

但你可以发现他们的决策显然有问题，如果有领导人的政策只与一人冲突，那么如果领导人从那个人更换到这个领导人，那么就会出现 专权（无法更换领导人）的状况，但海吃帝国的人民群众 遵纪守法，在 t 天内不会暴动，于是这个独裁者不会下台。

现给出领导人会给人民带来的幸福值和领导人的政策冲突关系。

作为当代谢顿的你要求出在 t 天的人民的期望幸福值。

简化题意

给出一张无向连通图和所有点的点权，第零天的棋子的位置是严格随机的，每天有一定概率跳到与所在点通过一条边相连的点处（所有可以跳到的点跳到的概率相同），但不能跳到上一次跳的所在地（当然中间一段时间不跳也不影响上一次跳的所在地），求第 t 天的棋子位置的期望点权。

可以发现有时棋子是不能继续往其他节点跳的。

输入格式

第一行给出 3 个正整数和一个分数 n, m, t, p 分别表示领导人数，政策冲突数，天数，和每天（这可真 喜新厌旧）更换领导人的概率。

第二行给出 n 个正整数，表示领导人会给人民带来的幸福值。

接下来 m 行，每行 2 个正整数 a, b 表示这两个领导人的政策是 ***互相冲突*** 的。

输出格式

输出在 t 天的人民的期望幸福值模 998244353 的大小。

样例数据

输入样例1

```
2 1 3 1/2
1 1
1 2
```

输出样例1

```
1
```

输入样例2

```
3 2 2 5/7
4 6 3
1 2
2 3
```

输出样例2

```
940522745 //即为 1249/294
```

输入样例3

```
5 7 3 1/3
5 3 4 2 2
3 1
4 3
4 1
2 4
5 3
5 1
5 2
```

输出样例3

```
344455925
```


输入样例4

```
5 9 34 3/23
1 5 25 12 3
3 1
5 1
2 3
2 1
2 5
4 5
4 2
4 3
4 1
```

输出样例4

```
359497807
```

样例解释

样例一：

这个不用解释吧。

样例二：

| <i>Subtask</i> | 测试点 | 数据范围 | 性质 | 分值 |
|----------------|---------|--|-------------------|-------|
| 1 | 1 – 4 | $2 \leq n \leq 10, 2 \leq t \leq 6$ | 无 | 20pts |
| 2 | 5 – 6 | $2 \leq n \leq 10^5, 2 \leq t \leq 10^6$ | 给出的图是一个环 | 10pts |
| 3 | 7 – 8 | $2 \leq n \leq 10^5, 2 \leq t \leq 10^6$ | 所有领导人会给人民带来的幸福值相等 | 10pts |
| 4 | 9 – 14 | $2 \leq n \leq 10^5, 2 \leq t \leq 10^3$ | 无 | 30pts |
| 5 | 15 – 20 | $2 \leq n \leq 10^5, 2 \leq t \leq 10^6$ | 无 | 30pts |

T4 【题目名称】

地理题 (travel)

题目背景

你找到了一张藏宝图。

题目描述

因为这张藏宝图设计得十分合理，所以从 1 号点（即起点）出发可以到达其他所有地点，两个地点之间至多有一条单向边，从一个地点出发到另一个地点简单路径至多只有一条，且没有自环。

你准备按这张藏宝图寻宝，你一开始有 $\text{¥}k$ 的钱，在通过一条边时会花费 $\text{¥}v_i$ 。

现在给出 T 个终点，你要求出从起点到终点剩下钱数的情况数（中途***不能经过***这次给定的终点）。

因为寻宝需要耗费大量精力，所以你只想在终点进行寻宝。

输入格式

第一行给出 4 个数 n, m, k, T ，表示点数，边数，开始的钱数，询问数。

接下来 m 行，每行三个数 a_i, b_i, v_i ，表示每条单向边。

接下来 T 行，每行一个数，表示询问的终点。

输出格式

对于每个询问的终点，输出一个数，表示他从 1 号点到这个终点剩下的钱数的情况数。

样例数据

输入样例1

```
7 8 33 3
4 3 4
3 5 1
5 4 1
6 3 1
3 2 1
2 6 5
7 2 2
1 7 3
3
5
4
```

输出样例1

```
1
4
3
```

输入样例2

```
10 12 80 3
4 7 3
7 3 4
3 4 2
8 7 2
7 2 4
2 9 4
9 8 2
6 9 4
9 5 2
5 6 1
1 5 4
8 10 1
2
3
10
```

输出样例2

```
40
32
45
```

样例解释

样例一：

到 3 号点可以剩下 ¥22。

到 4 号点可以剩下 ¥0, 7, 14, 21。

到 5 号点可以剩下 ¥6, 13, 20。

数据范围

本题采用捆绑测试。

保证图合法，每条简单回路的边数小于 100，数据随机度高（实际上是搞不出像样的数据）。

对于 20% 的数据， $n, m, k \leq 100$ ，数据极度随机。

对于另外 30% 的数据， $m \leq n + 15, T \leq 10$ 。

对于所有数据 $1 \leq n \leq 1000, 1 \leq k \leq 10^{18}, 1 \leq v_i \leq 5, 1 \leq T \leq n$ 。