

# 2020~2021 年信息学多校联合训练

## 2021 年冬令营模拟赛

时间：2021 年 01 月 23 日 8:00 ~12:00

题目名称	王国内战 • 剑鬼 • 魔法阵	哨塔	简单序列问题
源程序文件名	magic.cpp	tower.cpp	sequence.cpp
题目类型	传统型	传统型	提交答案
目录	magic	tower	sequence
输入文件名	magic.in	tower.in	sequence.in
输出文件名	magic.out	tower.out	sequence.out
每个测试点时限	1秒	3秒	2秒
内存限制	512MB	512MB	512MB
子任务数目	捆绑测试	捆绑测试	20
测试点是否等分	否	否	是

### 提交源程序文件名

对于C++语言	magic.cpp	tower.cpp	sequence.cpp
---------	-----------	-----------	--------------

### 编译选项

对于C++语言	-std=c++11 -O2
---------	----------------

### 注意事项

- 1.程序名必须使用英文小写
- 2.结果比较方式为忽略行末空格、文末回车后的全文比较
- 3.C/C++中函数 `main()` 的返回值类型必须是 `int`，值为 0。
- 4.输入数据中，若一行包含多个整数，则相邻两个整数均以 一个空格分隔。不保证输入数据的行末没有多余的空格。

## 王国内战 · 剑鬼 · 魔法阵 (magic)

### 【题目背景】

「鬼……」

鬼。没错，是鬼。在那边的是正在挥剑的鬼。在战场上砍杀敌人、笑得开怀的鬼。

享受挥剑杀戮的乐趣，像玩乐一样量产死亡的鬼。

——亦即，剑鬼。

「来啊！再来啊！尽管过来！放马过来！我会杀了你们，然后活下去！！」  
剑鬼叫喊，每一次的银闪都夺去亚人族的性命。

面前是那样的惨状，格林慢慢转过头，俯瞰山丘下的光景。

在烟雾弥漫的战场上，只有魔法阵的淡淡光芒鲜明异常。

用上双手双脚的指头也数不清的魔法阵，在战场上注入超越人类智慧的力量，以破坏力粉碎愚蠢地被围困的王国军。

热风，地鸣，还有远方被烧红的天空中所回荡的遍野哀嚎——

「……这就是我做的选择的下场吗？」

背后是剑鬼制造的尸山血河。

眼底是被遗弃的同伴的惨叫与叫喊，听来像在诅咒生者。

不知何时格林跪地，双手掩面不停地啜泣。

「对、不起……对不起，对不起，对不起……！」

直至战斗结束，王国军吞下败战苦果并回收友军之前，格林带着哽咽的谢罪以及剑鬼的剑舞和大笑声，都没有中断过。

——撤退的小队以及托尔塔·威兹利都没有回来。

### 【题目描述】

「……原来如此。听到的时候就有想到会是这样，这果然是花了相当多的时间来设置的术式。不管是拟定战术的人还是执行的人，要是不精通魔法，可办不到～呢。这样看来，王国恐怕是岌岌可危。」

「是、是这样吗？这是那么危险的魔法……」

「这个单一魔法阵本身的危险性就不用说了，问题在于敌方阵营里头出类拔萃的魔法使者不只——一人。在整个战场铺设魔法阵，从构想来看就已经很反常了……但同样的手法，也就可以套用在其他地方——啰。」

……

经过宫廷魔导师罗兹瓦尔·J·梅札斯的分析，所有的魔法阵都按照如下规则布置：魔法阵中有  $n$  个节点和  $n$  条边，边有边权，一开始每条边是暗淡的，边权是边点亮后能够吸收的玛那量。这  $n$  条边都可以用  $(i, t_i, v_i)$  表示，即  $i$  到  $t_i$  的一条双向边，权值为  $v_i$ 。可能会出现重边。为了方便记所有边和点形成的

图为  $F$ ，亮着的边和点形成的图为  $G$ 。

现在需要破解这个魔法阵：对于一个魔法阵，王国军队需要派出一名优秀的魔法使者来对付。亚人军队也有操控这个魔法阵的使者。在启动前，王国军队需要尽快施展魔法操作魔法阵，使得最终魔法阵的破坏力最小；当然对方也会尽力阻挠，使最终的魔法阵的破坏力最大。由于操作魔法需要时间间隔，在最快速度下两人刚好轮流施展魔法，王国军队为先手。不管是哪边，施展魔法时，每次选择一条合法的边，将其点亮。某条边合法的条件当且仅当：这条边没有被点亮，且如果点亮这条边， $G$  中不会出现环。如果不存在可以选择的边，魔法阵启动，产生的破坏力和它吸收的玛那量成正比。

双方的魔法师都是绝顶聪明的，所以罗兹瓦尔想要知道，对于一个特定的魔法阵，最终它吸收的玛那量为多少，以判断这个魔法阵的危险程度。

### 【输入描述】

第一行一个整数  $n$ 。

第二行  $n$  个整数，表示数组  $t_i$ 。

第三行  $n$  个整数，表示数组  $v_i$ 。

数据不保证无自环

### 【输出描述】

一行一个整数表示答案。

### 【样例输入】

见下发文件。

### 【样例输出】

见下发文件。

### 【数据范围】

数据点编号	分值	$N \leq$	$V_i \leq$	特殊性质
1	5	10	10	
2	10	20	20	
3	10	100	$10^6$	$F$ 由两个环组成
4	15	$10^5$	$10^6$	$F$ 中的所有环长度为奇数
5	15	$10^5$	$10^6$	$F$ 中的所有环长度为 2
6	45	$10^5$	$10^6$	

对于所有的数据，保证  $n \leq 10^5$ ， $v_i \leq 10^6$ ， $t_i \in [1, n]$ 。

## 哨塔(tower)

### 【题目描述】

崩坏的世界，陨落的双子。跨越诸多世界的你被剥夺了力量，在提瓦特大陆上寻找着失散的哥哥（妹妹）.....

然而，由于寻找哥哥（妹妹）的路遥远而又漫长，平日你只能通过摧毁丘丘人的哨塔来消磨时光（~~才不是因为悬赏~~），阅历的不断丰富让你对丘丘人的哨塔产生了极大的兴趣。

一个丘丘人部落可以简单地看成一个  $n$  个空地  $n-1$  条路的图，并且作为一个部落显然需要任意两个空地可以通过路互相到达。

在某些空地上可能会有一个哨塔，在整个丘丘人部落中一共有  $m$  个哨塔。

通过长久地与丘丘人增进感情（~~摧毁哨塔~~），你发现如果有一个空地（或放了哨塔的位置），它距离所有  $m$  个哨塔的距离都不超过  $d$ ，那么你可以一个 AOE 迅速完成今日份的任务。

对于一个丘丘人部落，你想知道对于所有可能的哨塔的安排，最多有多少个点所有哨塔的距离不超过  $d$ 。

### 【输入描述】

第一行三个整数。

接下来一行  $n-1$  个数，第  $i$  个数  $f[i+1]$  ( $f[i+1]<i+1$ ) 表示  $i+1$ ， $f[i+1]$  之间有一条道路。

### 【输出描述】

一行一个整数，表示最多有多少个点到所有哨塔距离不超过  $d$ ，如果不存在请输出 0。

### 【样例输入 1】

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

### 【样例输出 1】

```
5
```

### 【样例解释】

选择 1, 3 作为哨塔，除了 6, 7 以外的 5 个点都满足条件。

### 【样例输入/输出 2】

见下发文件 `ex_tower2.in/ex_tower2.ans`

### 【样例输入/输出 3】

见下发文件 `ex_tower3.in/ex_tower3.ans`

**【数据范围】**

对于 100% 的数据满足  $n \leq 10^6$  ,  $1 \leq m \leq n$  ,  $0 \leq d \leq n$

子任务编号	$n \leq$	特殊性质	分值
1	20		10
2	500		15
3	5000		15
4	100000		10
5	100000	$m=n$	10
6	500000		20
7	1000000		20

## 简单序列问题 (sequence)

### 【题目描述】

一个长度为  $n$  的序列  $A$  是合法的，当且仅当它满足如下条件：

1、 $1 \leq a_1 \leq a_2 \leq \dots \leq a_n \leq n$  ；

2、对于任意  $k$  满足  $1 \leq k \leq n$ ，要求从  $A$  中取出**任意**  $k+1$  个数的和**严格大于**从  $A$  中取出**任意**  $k$  个数的和。

现在给出  $n$ ，求合法的  $A$  的个数，答案对质数  $p$  取模。

### 【输入描述】

一行两个正整数表示  $n$  和  $p$ 。

### 【输出描述】

一行一个非负整数表示合法的  $A$  的个数对  $p$  取模的结果。

### 【样例输入】

2 998244353

### 【样例输出】

3

### 【样例解释】

合法的  $A$  有  $(1, 1)$ ， $(1, 2)$ ， $(2, 2)$  3 种，每一种都满足当  $k=1$  时  $a_1 < a_1 + a_2$  且  $a_2 < a_1 + a_2$ 。

### 【数据范围】

对于 20 % 的数据， $n \leq 3333$  ；

对于 50 % 的数据， $n \leq 33333$  ；

对于 100 % 的数据， $n \leq 333333$ ， $9 \cdot 10^8 \leq p < 10^9$ 。

保证奇数编号测试点中  $n$  为奇数，偶数编号测试点中  $n$  为偶数。

### 【特别说明】

保证本题时限不少于标程最大运行用时的 2 倍，**但请注意实现常数**。