

NOIP 模拟赛 day1

—by wcy1122

第一试

竞赛时间：2019 年 * 月 * 日上午 8:00-11:30

| | | | |
|---------|-----------|--------------|----------|
| 题目名称 | 小 w 与数字游戏 | 小 w 与密码 | 小 w 与最长路 |
| 目录 | game | password | path |
| 可执行文件名 | game | password | path |
| 输入文件名 | game.in | password.in | path.in |
| 输出文件名 | game.out | password.out | path.out |
| 每个测试点时限 | 1 秒 | 1 秒 | 3 秒 |
| 测试点数目 | 10 | 10 | 10 |
| 每个测试点分值 | 10 | 10 | 10 |
| 题目类型 | 传统 | 传统 | 传统 |
| 内存上限 | 512M | 512M | 512M |

提交源程序须加后缀

| | | | |
|--------------|----------|--------------|----------|
| 对于 Pascal 语言 | game.pas | password.pas | path.pas |
| 对于 C 语言 | game.c | password.c | path.c |
| 对于 C++ 语言 | game.cpp | password.cpp | path.cpp |

小 w 与数字游戏

空间限制：512M 时间限制：1s

【问题描述】

小 w 喜欢玩数字游戏。

数字游戏是这样的一个游戏，给你一个包含 n 个非负整数的数列，第 i 个数字为 s_i 。你可以进行若干次操作直到只剩一个数字。对于每次操作，你可以选择两个序号不同的数字 s_i 和 s_j ，将这两个数字从数列中删除，并加入以下三个数字中的任意一个。

1) $s_i + s_j$ 2) $|s_i - s_j|$ 3) $s_i * s_j$

显然在进行 $n-1$ 次操作后只会留下一个数字，小 w 希望这个数字尽量小。他希望得到这个最小的剩余数字，并将这个问题交给了你。

【输入文件】

输入的第一行为数据组数 T 。

对于每组数据，第一行输入一个整数 n ，表示数字个数。第二行输入 n 个非负整数构成的数列 s_i 。

【输出文件】

对于每组数据，输出一个整数，表示答案。

【输入样例】

```
2
3
1 5 12
4
2 6 10 28
```

【输出样例】

```
6
2
```

【数据规模和约定】

对于 30% 的数据，保证 $n \leq 7$

对于 50% 的数据，保证 $n \leq 10$

对于 100% 的数据，保证 $T \leq 5$ ， $n \leq 3 * 10^5$ ， $0 \leq s_i \leq 30$

小 w 与密码

空间限制：512M 时间限制：1s

【问题描述】

小 w 获取了一段密文，这段密文包含两个由小写英文字母构成的字符串 s , t 。除了这段密文，小 w 还同时得到关键情报，明文是由这两个字符串的非空前缀拼接而成的。并且， s 串的前缀必须要拼在 t 串前缀的前面。

比如 $s = \text{'aba'}$, $t = \text{'aa'}$ 。那么从 s 串中取 ab , t 串中取 a , 'aba' 就是一组可能的密文。而 'aab' 就不是一组可能的密文，因为无论怎么取都无法拼接成这个字符串。

小 w 现在已经得到了密文，即两个字符串 s 和 t ，他想要知道按照以上规则，由这两个字符串能够拼接成多少种互不相同的明文。

【输入文件】

第一行一个字符串 s ，第二行一个字符串 t 。数据保证两个字符串只由小写英文字母构成。

【输出文件】

一个整数，表示明文的方案数。

【输入样例】

```
aba
aa
```

【输出样例】

```
5
```

【数据规模和约定】

对于 10% 的数据， $|s|, |t| \leq 10$

对于 20% 的数据， $|s|, |t| \leq 100$

对于 50% 的数据， $|s|, |t| \leq 1000$

对于 100% 的数据， $|s|, |t| \leq 10^5$

【样例解释】

S 串的前缀 = { 'a' , 'ab' , 'aba' }

T 串的前缀 = { 'a' , 'aa' }

互不相同的明文 = { 'aa' , 'aaa' , 'aba' , 'abaa' , 'abaaa' }

小 w 与最长路

空间限制：512M 时间限制：3s

【问题描述】

小 w 有一棵树，这棵树有 n 个节点和 $n-1$ 条边。树上的每条边有边权 w_i ，任意两个不同节点间有且仅有一条路径相连。小 w 想要知道这棵树中所有这样的路径中长度最长的一条。每条路径的长度定义为这条路径上所有边的边权和。

小 w 觉得这个问题太简单了，他想要加大难度。对于独立的每条边，小 w 首先会先将其从原来的树上删掉，使这棵树变成两个连通块。然后，小 w 会将这条边加在这张图的某两个不同节点上，使这两个连通块重新联通。对于每条边，小 w 想要知道，在如此操作后树上的最长路径长度的最小长度。

Hint：操作是临时性的。

小 w 是个超级大懒虫，它懒得造数据，所以他提供的数据只有四个数字 n, num, B, D 。你需要根据以下代码生成完整的输入数据。

```
int n,B,D;
unsigned long long num;

unsigned long long get(){
    num^=(num<<13);
    num^=(num>>17);
    num^=(num<<5);
    return num;
}

void gen(){
    scanf("%d%llu%d%d",&n,&num,&B,&D);
    for(int i=2;i<=n;i++){
        a=get()%min(i-1,B)+i-min(i-1,B);
        b=get()%D;
        add_edge(i,a,b)
    }
}
```

即对于每个 $[2 \sim n]$ 的节点，通过以上方式随机生成一个相邻节点 a 和边权 b 。

鉴于输出较多，你只需要输出 $n-1$ 条边的答案和边序号的乘积对 P 的余数的异或和即

可，即 $(1 * ans_1 \% P) ^ (2 * ans_2 \% P) ^ \dots ^ ((n-1) * ans_{n-1} \% P)$ ，其中 $P=998244353$ 。

【输入文件】

输入数据只有一行，包含四个数字， n ， num ， B ， D 。

【输出文件】

你只需要输出一个数字，取值如题面最后一行描述。

【输入样例】

5 3 5 3

【输出样例】

21

【数据规模和约定】

对于 20%的数据，保证 $n \leq 50$

对于 70%的数据，保证 $n \leq 2 \times 10^5$

对于 100%的数据，保证 $n \leq 2 \times 10^6$

【样例解释】

利用以上数据生成器会产生一棵有 5 个节点的带权树，4 条边依次为：

1 2 1

1 3 2

1 4 2

3 5 2

删除第 1 条边，无论怎么连，树上最长路都是 6。

删除第 2 条边，无论怎么连，树上最长路都是 6。

删除第 3 条边，并在 3 和 4 间连一条边权为 2 的边，树上的最长路变为 2-1-3-5，长度为 5。

删除第 4 条边，并在 1 和 5 间连一条边权为 2 的边，树上的最长路变为 3-1-4，长度为 4。

$(1 \times 6)^{(2 \times 6)}(3 \times 5)^{(4 \times 4)} = 21$