

# 全国信息学奥林匹克联赛 (NOIP2018)

## 模拟赛

租酥雨

2018 年 10 月 18 日

题目名称	基础分治练习题	大火题	小水题
题目类型	传统型	传统型	传统型
目录	cdq	easy	hard
可执行文件名	cdq	easy	hard
输入文件名	cdq.in	easy.in	hard.in
输出文件名	cdq.out	easy.out	hard.out
每个测试点时限	1.5 秒	3.0 秒	2.0 秒
内存限制	512MB	512MB	512MB
测试点数目	25	20	10
测试点是否等分	是	是	是

提交源程序文件名

对于 C++ 语言	cdq.cpp	easy.cpp	hard.cpp
对于 C 语言	cdq.c	easy.c	hard.c
对于 Pascal 语言	cdq.pas	easy.pas	hard.pas

编译选项

对于 C++ 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 C 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 Pascal 语言	说得好像你会交 Pascal 源文件似的		

注意事项

- 文件名（包括程序名和输入输出文件名）必须使用英文小写。
- 结果比较方式为忽略行末空格、文末回车后的全文比较。
- C/C++ 中函数 main() 的返回值类型必须是 int，值为 0。
- 评测时栈空间大小限制等同于内存限制。

## 1 基础分治练习题 (cdq)

### 1.1 题目描述

这是一道基础分治练习题。

给你三个数列  $\{a_i\}, \{b_i\}, \{c_i\}$ , 保证每个数列都恰好是一个排列。你需要求出满足  $a_i < a_j, b_i < b_j, c_i < c_j$  的有序对  $(i, j)$  的数目。

### 1.2 输入格式

从文件 *cdq.in* 中读入数据。

数据的第一行包含一个正整数  $n$ , 表示数列的长度。接下来一行三个非负整数  $SA, SB, SC$ 。

为了避免过量的输入对程序的运行效率产生影响,  $\{a_i\}, \{b_i\}, \{c_i\}$  由以下代码生成:

```
const int N = 2e6+5;
unsigned int SA,SB,SC;int n,a[N],b[N],c[N];
unsigned int rd(){
    SA^=SA<<16;SA^=SA>>5;SA^=SA<<1;
    unsigned int t=SA;SA=SB;SB=SC;SC^=t^SA;return SC;
}
void gen(int *P){
    for (int i=1;i<=n;++i) P[i]=i;
    for (int i=1;i<=n;++i) swap(P[i],P[1+rd()%n]);
}
int main(){
    scanf("%d%u%u%u",&n,&SA,&SB,&SC);
    gen(a);gen(b);gen(c);return 0;
}
```

你可以在程序中自由运行这段代码。在下发文件中提供了这段代码的 `cpp` 文件, 你可以选择直接在该 `cpp` 文件下编写程序, 也可以假装看不见它。

### 1.3 输出格式

输出到文件 *cdq.out* 中。

输出一行一个正整数  $ans$ , 表示满足条件的有序对  $(i, j)$  的对数。

### 1.4 样例 1 输入

```
5
233 666 667
```

### 1.5 样例 1 输出

```
4
```

### 1.6 样例 1 解释

生成的数列为  $a = \{1, 2, 3, 4, 5\}$ ,  $b = \{1, 5, 3, 2, 4\}$ ,  $c = \{3, 4, 2, 1, 5\}$ 。

显然有且仅有  $(1, 2)$ ,  $(1, 5)$ ,  $(3, 5)$ ,  $(4, 5)$  这四个数对满足条件。

### 1.7 样例 2 输入

100000

123 456 789

### 1.8 样例 2 输出

1258889897

### 1.9 子任务

对于 100% 的数据, 保证  $1 \leq n \leq 2 \times 10^6$ ,  $1 \leq SA, SB, SC \leq 10^9$ 。

请注意常数因子对程序运行效率带来的影响。

测试点编号	$n \leq$
1-3	5000
4-9	$3 \times 10^5$
10-19	$i \times 10^5$ , $i$ 为测试点编号
20-25	$2 \times 10^6$

### 1.10 提示

加油! 这是道水题! A 掉它!

## 2 大火题 (easy)

### 2.1 题目描述

Cyhlmj 天天切大火题。

有一天他突然觉得切火题没什么意思，于是就翻开了这样的一道题目：

- 给你三个数  $n, G, L$ ，要你求从  $\{1, 2, \dots, n\}$  中选出一个非空子集使这个子集中所有数的最大公因数 (greatest common divisor, GCD) 恰好为  $G$ ，最小公倍数 (least common multiple, LCM) 恰好为  $L$  的方案数。

Cyhlmj 觉得这个题太牛逼了，于是他对此进行了一些加强：他会给出  $Q$  次询问，每次询问选出的子集必须包含某个正整数  $a_i$  的前提下，方案数会是多少。

因为答案可能很大，所以你需要输出答案对 998244853 取模后的结果。

### 2.2 输入格式

从文件 *easy.in* 中读入数据。

数据第一行包含四个正整数  $n, G, L, Q$ ，意义见题目描述。

接下来  $Q$  行每行一个正整数  $a_i$ ，表示询问选出的子集必须包含某个正整数  $a_i$  的前提下，方案数会是多少。

### 2.3 输出格式

输出到文件 *easy.out* 中。

输出包括  $Q + 1$  行，其中第一行一个数表示不加任何限制的情况下满足条件的方案数对 998244853 取模后的结果，接下来  $Q$  行每行一个数表示选出的子集中包含某个正整数  $a_i$  且满足条件的方案数对 998244853 取模后的结果。

### 2.4 样例 1 输入

```
6 1 6 4
1
2
3
6
```

### 2.5 样例 1 输出

```
7
5
5
5
5
```

## 2.6 样例 1 解释

满足条件的方案一共有 7 种:  $\{1, 2, 3\}$ ,  $\{1, 2, 6\}$ ,  $\{1, 2, 3, 6\}$ ,  $\{1, 3, 6\}$ ,  $\{1, 6\}$ ,  $\{2, 3\}$ ,  $\{2, 3, 6\}$ 。可以发现 1, 2, 3, 6 中的每个数都恰好在 5 种方案里出现过。

## 2.7 样例 2

见选手目录下的 *easy/easy2.in* 与 *easy/easy2.ans*。

## 2.8 子任务

对于 100% 的数据, 保证  $1 \leq n, G, L \leq 10^8$ ,  $1 \leq Q \leq 5000$ ,  $1 \leq a_i \leq n$ 。

测试点编号	特殊性质
1	$[G L] = 0$
2-3	$n, G, L \leq 20$
4-6	$n, G, L \leq 250$
7-11	$Q = 0$
12-16	$n, G, L \leq 2 \times 10^5$
17-20	

## 2.9 提示

题目并不难。

## 3 小水题 (hard)

### 3.1 题目描述

众所周知, `yyb` 是位神仙。

神仙当然就要住在仙境里。由于 `yyb` 什么都有什么都会什么都强, 所以他的仙境是一个由  $n$  个节点构成的树形结构。这个树形结构有些特殊, 主要体现在每个节点与之相邻的节点个数不会太多。

众所周知, `yyb` 很喜欢球球, 所以他的仙境里自然少不了球球。这些球球可以作为他出毒瘤题时的灵感来源, 也可以让他在闲暇时间丢着玩。

有一天, `yyb` 邀请了 `SYCstudio` 来到他的仙境。`SYCstudio` 也是球球爱好者中的一员, 他喜欢研究球球的滚动及其物理性质, 于是他设计了  $Q$  次如下的实验: 对于每次试验, 他会先拿出  $k$  个不同的球球, 依次编号为  $1$  至  $k$ , 然后给这  $k$  个球球中的每个球球选定仙境中的两个节点  $a, b$ , 并让这个球球在这两点之间来回滚动, 这两个节点被称为这个球球的折返点。出于某种原因, `SYCstudio` 希望这  $k$  个球球中任意两个球球的滚动路径的交都等于仙境中  $u, v$  两点之间的简单路径 (显然路径是唯一的)。为了研究球球之间的碰撞可能导致的对其运动状态的影响, `SYCstudio` 有时会要求任两个球球的折返点互不重复, 有时又不会做这样的要求。

心满意足地设计完实验方案后, `SYCstudio` 想知道这样的实验他一共能够做出多少不同的变种。由于球球是两两不同的, 所以两个实验不同当且仅当存在某一个编号的球球它在两个实验中的运动路径不同。由于球球做的是往返运动, 所以  $(a, b)$  与  $(b, a)$  被视为相同的运动路径。

`SYCstudio` 想把这个问题丢给 `yyb`。但是 `yyb` 表示并不想做这个问题, 于是他又丢给了来仙境中膜拜 `yyb` 的你。

于是就由你来解决 `SYCstudio` 的问题。由于答案可能很大, 所以你需要输出答案对  $998244353$  取模后的结果。

### 3.2 输入格式

从 `hard.in` 中读入数据。

数据的第一行包含两个正整数  $n, Q$ , 表示 `yyb` 居住的仙境中节点的数量, 以及 `SYCstudio` 进行实验的次数。

接下来  $n - 1$  行每行两个正整数  $u, v$ , 表示一条边连接节点  $u, v$ 。保证输入数据构成一棵树。

接下来  $Q$  行, 每行四个非负整数  $u, v, k, op$  描述一次实验的参数。其中  $u, v, k$  的意义见题面描述,  $op = 1$  表示 `SYCstudio` 要求任两个球球的折返点互不重复,  $op = 0$  表示没有这样的要求。

### 3.3 输出格式

输出到文件 `hard.out` 中。

对于每次实验操作, 输出一行表示实验的不同变种数目对  $998244353$  取模的结果。

### 3.4 样例 1 输入

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

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

### 3.5 样例 1 输出

12  
 1  
 3

### 3.6 样例 1 解释

对于第一次实验：

- 当第一个球球的运动路径为 (1, 2) 时，第二个球球的运动路径可以为 (3, 4), (3, 5);
- 当第一个球球的运动路径为 (1, 4) 时，第二个球球的运动路径可以为 (2, 3), (3, 5);
- 当第一个球球的运动路径为 (1, 5) 时，第二个球球的运动路径可以为 (2, 3), (3, 4);
- 当第一个球球的运动路径为 (2, 3) 时，第二个球球的运动路径可以为 (1, 4), (1, 5);
- 当第一个球球的运动路径为 (3, 4) 时，第二个球球的运动路径可以为 (1, 2), (1, 5);
- 当第一个球球的运动路径为 (3, 5) 时，第二个球球的运动路径可以为 (1, 2), (1, 4)。

对于第二次实验，显然两个球球的运动路径都只能是 (3, 5)。

对于第三次试验，可行的方案有  $\{(1, 4), (1, 4)\}$ ,  $\{(1, 4), (3, 4)\}$ ,  $\{(3, 4), (1, 4)\}$ 。

### 3.7 样例 2

见选手目录下的 *hard/hard2.in* 与 *hard/hard2.ans*。

### 3.8 子任务

对于 100% 的数据，保证  $1 \leq n \leq 10^5, 1 \leq u, v \leq n, u \neq v, 1 \leq k \leq \min(n, 500), op \in \{0, 1\}$ ，保证每个节点的度数不超过 500。

测试点编号	$n, Q \leq$	特殊性质
1	5	
2,3	100	
4	3000	op=1
5,6	3000	
7,8	$10^5$	op=1
9,10	$10^5$	