

全国青少年信息学奥林匹克竞赛模拟赛

CCF NOI 2023 Simulation

第二试

时间：2023 年 7 月 11 日 08:00 ~ 13:00

题目名称	生成树	素数	杨表
题目类型	传统型	传统型	传统型
目录	rgb	prime	young
可执行文件名	rgb	prime	young
输入文件名	rgb.in	prime.in	young.in
输出文件名	rgb.out	prime.out	young.out
每个测试点时限	2.0 秒	2.0 秒	2.0 秒
内存限制	1024 MiB	1024 MiB	1024 MiB
测试点数目	10	10	10
测试点是否等分	是	是	是

提交源程序文件名

对于 C++ 语言	rgb.cpp	prime.cpp	young.cpp
-----------	---------	-----------	-----------

编译选项

对于 C++ 语言	-O2 -std=c++14
-----------	----------------

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C++ 中函数 main() 的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
3. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
4. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
5. 选手提交的程序源文件必须不大于 100KB。
6. 程序可使用的栈空间内存限制与题目的内存限制一致。
7. 禁止在源代码中改变编译器参数（如使用 #pragma 命令），禁止使用系统结构相关指令（如内联汇编）和其他可能造成不公平的方法。
8. 简单的题目，既是礼物，也是毒药。

生成树 (rgb)

【题目描述】

给定一张 $n = r + g + b + 1$ 的点的简单连通无向图，每条边为红色、绿色或蓝色。判断是否存在一棵包含恰好 r 条红边、 g 条绿边和 b 条蓝边的生成树。

【输入格式】

从文件 *rgb.in* 中读入数据。

第一行四个整数 r, g, b, m ，你可以自行计算 $n = r + g + b + 1$ 。

接下来 m 行，每行两个整数 u, v 和字符 $c \in \{r, g, b\}$ ，表示边 (u, v) 的颜色为 c 。

【输出格式】

输出到文件 *rgb.out* 中。

第一行一个字符串 YES 或 NO，表示是否存在合法的生成树。

若存在合法的生成树，第二行一个长度为 m 的 01 串，第 i 个字符表示你给出的生成树是否包含第 i 条边。如果有多解，给出任意一个即可。

【样例 1 输入】

```
1 2 1 1 5
2 1 2 r
3 2 3 r
4 1 3 r
5 2 4 g
6 2 5 b
```

【样例 1 输出】

```
1 YES
2 11011
```

【样例 2 输入】

```
1 1 1 0 2
2 1 2 r
3 2 3 r
```

【样例 2 输出】

1NO

【数据范围】

保证 $1 \leq n \leq 250, n - 1 \leq m \leq \frac{n(n-1)}{2}, 1 \leq u < v \leq n$ 。

测试点编号	特殊性质
1	$n \leq 5$
2	$n \leq 30$
3 ~ 4	$n \leq 100$
5	A
6 ~ 7	B
8 ~ 10	无特殊限制

特殊性质 A: $m = \frac{n(n-1)}{2}$ ，每条边颜色随机， $r = g = b = \frac{n-1}{3}$ 。
特殊性质 B: $b = 0$ 。

素数 (prime)

【题目描述】

给定 l, r ，求满足不存在素数 p 使得 p 在十进制表示下视作字符串后是 x 的子序列的整数 $x \in [l, r]$ 的数量对 998 244 353 取模的值。

【输入格式】

从文件 `prime.in` 中读入数据。
第一行一个整数 T ，表示数据组数。
接下来 T 行，每行两个整数 l, r 。

【输出格式】

输出到文件 `prime.out` 中。
 T 行，每行一个整数，表示答案对 998 244 353 取模的值。

【样例输入】

```
1 2
2 1 4242
3 123456789 10000000000000000000
```

【样例输出】

```
1 222
2 245262832
```

【数据范围】

保证 $1 \leq T \leq 10, 1 \leq l \leq r \leq 10^{10^5}$ 。

测试点编号	$r \leq$
1	10^5
2 ~ 3	10^9
4 ~ 5	10^{12}
6 ~ 7	10^{18}
8	10^{100}
9	10^{10^4}
10	10^{10^5}

杨表 (young)

【题目描述】

定义非严格递减的无限长度的非负整数序列为**杨表**。定义给杨表中的某个数增加或减少 1 为对杨表的一次**变换**，要求变换后仍为杨表。

给定两个杨表 A, B 。有 q 次询问，每次给出 k ，求通过 k 次变换将 A 变成 B 的方案数对 998 244 353 取模的值。

【输入格式】

从文件 *young.in* 中读入数据。

第一行一个整数 n 。

第二行 n 个整数 a_1, a_2, \dots, a_n ，表示杨表 A 的前 n 个元素，其余元素均为 0。

第三行一个整数 m 。

第四行 m 个整数 b_1, b_2, \dots, b_m ，表示杨表 B 的前 m 个元素，其余元素均为 0。

第五行一个整数 q 。

接下来 q 行，每行一个整数 k 。

【输出格式】

输出到文件 *young.out* 中。

q 行，每行一个整数，表示答案对 998 244 353 取模的值。

【样例 1 输入】

```
1 3
2 3 2 1
3 3
4 3 2 1
5 1
6 2
```

【样例 1 输出】

```
1 7
```

【样例 1 解释】

变换为 $\{4, 2, 1\}/\{3, 3, 1\}/\{3, 2, 2\}/\{3, 2, 1, 1\}/\{2, 2, 1\}/\{3, 1, 1\}/\{3, 2\}$ 后再变换为 B 。

【样例 2 输入】

```
1 4
2 4 3 2 1
3 4
4 3 3 2 2
5 1
6 4
```

【样例 2 输出】

```
1 40
```

【样例 3 输入】

```
1 10
2 20 10 7 5 4 3 3 3 3 2
3 10
4 17 10 8 8 6 4 4 1 1 1
5 1
6 1000000
```

【样例 3 输出】

```
1 750051767
```

【数据范围】

保证 $0 \leq n, m, \sum a_i, \sum b_i \leq 60, 1 \leq q \leq 10^5, 1 \leq k \leq 10^7$ 。
保证 A, B 为杨表，即 $a_1 \geq a_2 \geq \cdots \geq a_n > 0, b_1 \geq b_2 \geq \cdots \geq b_m > 0$ 。

测试点编号	$\sum a_i, \sum b_i \leq$	$q \leq$	$k \leq$
1	10	1	10
2	20		20
3 ~ 4			10^6
5 ~ 6	40		
7 ~ 8	60		
9 ~ 10		10^5	10^7