

NOIP 模拟赛（CWOI2022）

第一试

（请选手务必仔细阅读本页内容）

一、题目概况

中文题目名称	报数	随机	单调栈	后缀数组
子目录名	number	random	stack	suffix
可执行文件名	number.cpp	random.cpp	stack.cpp	suffix.cpp
输入文件名	number.in	random.in	stack.in	suffix.in
输出文件名	number.out	random.out	stack.out	suffix.out
每个测试点时限	1 秒	1 秒	1 秒	6 秒
内存上限	512M	512M	512M	512M
测试点数目	10	20	25	20
每个测试点分值	10	5	4	5
附加样例文件	有	有	有	有
结果比较方式	全文比较（过滤行末空格及文末回车）			
题目类型	传统	传统	传统	传统

二、提交源程序程序名

对于 C++ 语言	number.cpp	random.cpp	stack.cpp	suffix.cpp
-----------	------------	------------	-----------	------------

注意事项：

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/C++ 中函数 main() 的返回类型必须是 int，程序正常结束时的返回值必须是 0。
3. 特别提醒：评测在 NOI Linux 下进行。

1 报数

(number.cpp)

1.1 问题描述

一天，小 L 和小 W 在玩一个报数游戏。两人约定只能报出质数，即因数只有 1 和它自身的数。特别的，1 不被视作质数。然而，在只有两个人的情况下，找到质数还是很容易的，因此他们玩了很久也没分出胜负。此时，小 L 灵光一闪，决定把这个游戏加强：报出的数不仅要是质数，还要满足它的各位数字之和也是一个质数！

例如，31 不是一个满足条件的数，因为虽然它本身是一个质数，但它的各位数字之和为 4，不是一个质数；34 不是一个满足条件的数，因为虽然它的个位数字之和为 7，是一个质数，但它本身并不是一个质数；41 是一个满足条件的数，因为不仅它本身是一个质数，它的各位数字之和 5 也是一个质数。

现在小 L 想要集中精力报出 $[L, R]$ 之间的满足条件的数，因此他会询问你范围内满足条件的数的个数是多少。小 L 和小 W 一共会玩 T 次游戏，而你也要回答 T 个问题。

为了减少输入输出量，小 L 对输入输出的方式进行了一些处理。

1.2 输入

第一行一个整数 T ，表示小 L 询问的数量。

第二行五个整数 L, R, a, b, c ，表示第一个询问的范围为 $[L, R]$ ，对于第 $i (i \geq 2)$ 个询问，该次询问的范围 L, R 与上次询问的范围 L_0, R_0 满足如下关系：

设 $x = (L_0 \oplus b + a) \bmod c + 1, y = (R_0 \oplus b + a) \bmod c + 1$ ，则 $L = \min(x, y), R = \max(x, y)$ 。其中 \oplus 表示二进制下按位异或运算。

1.3 输出

一行一个整数，表示每组询问的答案的异或和。

1.4 输入输出样例 1

1.4.1 样例输入

```
2
2 9 1 0 20
```

1.4.2 样例输出

```
7
```

1.4.3 样例解释

对于第一次询问，满足条件的数有 2, 3, 5, 7。

对于第二次询问，询问的范围是 [4, 11]，满足条件的数有 5, 7, 11。

两次询问答案的异或和为 $4 \oplus 3 = 7$ 。

1.5 输入输出样例 2

见下发文件中的 number2.in 和 number2.out。

该样例满足测试点 8 ~ 10 的数据范围和限制。

1.6 数据范围

对于所有数据，满足 $1 \leq T, a, b, c \leq 10^7, 1 \leq L \leq R \leq 10^7$ 。

测试点编号	T	L, R, c
1 ~ 2	≤ 10	$\leq 10^4$
3 ~ 4	$\leq 10^5$	$\leq 10^4$
5 ~ 7	$\leq 10^5$	$\leq 10^7$
8 ~ 10	$\leq 10^7$	$\leq 10^7$

2 随机

(random.cpp)

2.1 问题描述

一天，小 L 正在做一道数据结构题。这道题的操作非常简单，只分为**区间加**、**区间赋值**和**单点查询**三种。当然，小 L 一下子就秒了这题，但他突然有了一个奇妙的想法，想要为这道题生成数据。

首先，序列长度 n 、询问次数 q 、单点查询的次数 k 和其他两种操作的值域 V 是给定的，即其他两种操作共有 $q - k$ 次。初始时，可以认为序列中所有的数均为 0。

接下来，小 L 将这 $q - k$ 次操作随机划分为了 k 段，并在每两段之间及最后一次操作的位置**依序**插入了单点查询的操作。不难发现，当划分出的每一段长度至少为 1 时，上述过程相当于求 $x_1 + x_2 + \dots + x_k = q - k, \forall x_i \geq 1$ 这个方程的解数，因此划分的方案总共有 C_{q-k-1}^{k-1} 种。你可以认为小 L 是在这些方案中等概率选取了一种划分方案。

小 L 提前确定了每次单点查询的位置 p_1, p_2, \dots, p_k ，并以下面的方式生成另外两种操作：

- 在 $1 \sim V$ 中随机生成一个操作参数，并随机一个操作类型。
- 在 $1 \sim n$ 中随机生成两个数 x, y ，并以它们中的较小值为左端点，较大值为右端点。
- 不难发现，每次操作都有 $2n^2V$ 种可能，你可以认为小 L 是在它们之中等概率选取了一种。

数据生成完毕之后，小 L 找到你，希望你能帮他解决一个问题：按上述方式生成该题的数据，那么每次单点查询操作的结果之和的数学期望是多少？答案对 998244353 取模。

2.2 输入

第一行四个整数 n, q, k, V ，含义如题目描述所示。

第二行 k 个整数 p_1, p_2, \dots, p_k ，含义如题目描述所示。

2.3 输出

一行一个整数表示答案。

2.4 输入输出样例 1

2.4.1 样例输入

```
2 2 1 3
2
```

2.4.2 样例输出

```
499122178
```

2.4.3 样例解释

生成的操作方案只有一种可能：第 1 次操作为区间加或区间赋值，第 2 次操作为询问位置 2 的值。

随机生成的第一次操作有以下几种可能（区间加用 $(l, r + v)$ 表示，区间赋值用 $(l, r, = v)$ 表示）：

- $(1, 1, +1/2/3), (1, 2, +1/2/3), (2, 2, +1/2/3)$ 。
- $(1, 1, = 1/2/3), (1, 2, = 1/2/3), (2, 2, = 1/2/3)$ 。

共有 24 种可能的生成方式，而所有情况中询问的结果之和为 36，故数学期望为 $\frac{36}{24} = \frac{3}{2}$ ，在模意义下为 499122178。（对于 $l = 1, r = 2$ 的操作有两种生成方式）

2.5 输入输出样例 2

2.5.1 样例输入

```
114514 1314 6 1919810
233 2333 888 520 23333 12345
```

2.5.2 样例输出

```
767130984
```

2.6 输入输出样例 3

见下发文件中的 random3.in 和 random3.out。

该样例满足测试点 6 ~ 8 的数据范围和限制。

2.7 输入输出样例 4

见下发文件中的 random4.in 和 random4.out。

该样例满足测试点 15 ~ 20 的数据范围和限制。

2.8 数据范围

对于所有数据，满足 $1 \leq n, V \leq 9 \times 10^8, 1 \leq k \leq \min\{4000, \frac{q}{2}\}, 1 \leq q \leq 4000, 1 \leq p_i \leq n$ 。

测试点编号	n	q	k	V
1 ~ 2	≤ 3	≤ 3	≤ 3	≤ 3
3	1	≤ 1000	≤ 1000	$\leq 9 \times 10^8$
4 ~ 5	$\leq 9 \times 10^8$	≤ 3	≤ 3	$\leq 9 \times 10^8$
6 ~ 8	$\leq 9 \times 10^8$	≤ 300	≤ 300	$\leq 9 \times 10^8$
9 ~ 14	$\leq 9 \times 10^8$	≤ 1000	≤ 1000	$\leq 9 \times 10^8$
15 ~ 20	$\leq 9 \times 10^8$	≤ 4000	≤ 4000	$\leq 9 \times 10^8$

3 单调栈

(stack.cpp)

3.1 问题描述

最近, 小 L 对单调栈产生了浓厚的兴趣。单调栈的功能之一对于给定的序列 $a_{1\dots n}$, 求出一个序列 $b_{1\dots n}$, 满足 $b_i = \max\{j \mid j < i, a_j > a_i\}$ 。特别地, 若 $b_i = 0$, 表示不存在这样的 j 。

给定一个序列 $b_{1\dots n}$, 其中 $-1 \leq b_i < i$ 。请求出有多少个序列 $a_{1\dots n}$ 满足:

- $1 \leq a_i \leq m$ 。其中 m 是一个给定的常数。
- 对于序列 a 求出的单调栈序列 b' 满足 $b_i \neq -1 \Leftrightarrow b'_i = b_i$ 。

答案对 $10^9 + 7$ 取模。特别地, 小 L 保证至少存在一个序列 $a_{1\dots n}$ 满足上述条件。

3.2 输入

第一行两个整数 n, m 描述序列长度和值域限制。

第二行 n 个整数 b_1, \dots, b_n 描述给定的序列。数据保证 $-1 \leq b_i < i$ 。

3.3 输出

一行一个整数表示答案对 $10^9 + 7$ 取模的结果。

3.4 输入输出样例 1

3.4.1 样例输入

```
3 3
0 -1 1
```

3.4.2 样例输出

```
4
```

3.4.3 样例解释

由 $b_3 = 1$, 可知 $a_1 > a_3 \geq a_2$ 。因此, 有如下 4 种序列满足条件: $\{3, 2, 2\}, \{3, 1, 2\}, \{3, 1, 1\}, \{2, 1, 1\}$ 。

3.5 输入输出样例 2

3.5.1 样例输入

```
7 7
0 1 1 -1 -1 3 -1
```

3.5.2 样例输出

6076

3.6 输入输出样例 3

见下发文件中的 stack3.in 和 stack3.out。

该样例满足测试点 5 ~ 10 的数据范围和限制。

3.7 输入输出样例 4

见下发文件中的 stack4.in 和 stack4.out。

该样例满足测试点 19 ~ 25 的数据范围和限制。

3.8 数据范围

对于所有数据，满足 $1 \leq n \leq 300, 1 \leq m \leq 10^5, -1 \leq b_i < i$ 。

测试点编号	n	m	特殊性质 A	特殊性质 B
1 ~ 4	≤ 7	≤ 7	否	否
5 ~ 10	≤ 50	≤ 50	否	否
11 ~ 13	≤ 50	$\leq 10^5$	否	否
14 ~ 16	≤ 300	≤ 300	是	否
17 ~ 18	≤ 300	≤ 300	否	是
19 ~ 25	≤ 300	$\leq 10^5$	否	否

特殊性质 A：不存在 $b_i = -1$ 。

特殊性质 B：若 $b_i \neq -1$ ，则 $b_i = 0$ 。

4 后缀数组

(suffix.cpp)

4.1 问题描述

4.1.1 背景化题意

后缀数组 (Suffix Array) 是一种处理字符串问题的常用结构。对于给定的字符串 S , 设它长度为 i 的后缀为 T_i , 则后缀数组可以求出两个数组 sa 和 rk , 满足 $sa_{rk_i} = rk_{sa_i} = i$, 且 sa_i 表示将所有后缀 T_1, T_2, \dots, T_n 按字典序升序排序后, 第 i 大的后缀的编号 (我们定义起点在 i 的后缀编号为 i)。

现在, 小 L 手里有一个字符串, 并且他求出了该字符串的后缀数组 sa 和 rk 。同时, 小 L 还写下了一个数组 cmp , 其中 $cmp_i = [rk_i < rk_{i+1}]$ 。不幸的是, 可恶的小 W 偷走了这个字符串和两个数组 sa, rk , 现在小 L 只剩下了 cmp 数组。

可怜的小 L 找到了你, 想要问问你满足 cmp 数组的字符串有多少个。除此之外, 他还记得原来的字符串中只有 a, b, c 三种字符。因为小 L 对他的记忆抱有怀疑态度, 所以他会不时地进行一些修改。你需要在他每一次更改之后求出答案。具体地, 小 L 可能会做以下两种修改:

- Flip $l\ r$, 表示将 $cmp_l, cmp_{l+1}, \dots, cmp_r$ 全部**反转**, 即修改之后, 若 cmp_i ($l \leq i \leq r$) 原来为 0 则会变为 1; 若原来为 1 则会变为 0。

- Reverse $l\ r$, 表示将 $cmp_l, cmp_{l+1}, \dots, cmp_r$ 这个子区间**翻转**, 即修改之后, 该子区间将会变为 $cmp_r, cmp_{r-1}, \dots, cmp_l$ 。

因为满足条件的字符串可能会非常多, 所以你只需要输出答案对 $10^9 + 7$ 取模的结果即可。

4.1.2 形式化题意

给定一个长度为 $n - 1$ 的数组 cmp 。你有一个长度为 n 的字符串, 设 rk_i 表示其中起点为 i 的后缀在所有后缀中的字典序排名 (升序), 则 $cmp_i = [rk_i < rk_{i+1}]$ 。其中 $[p]$ 表示若 p 为真, 则表达式的值为 1, 否则为 0。

给定 q 次修改操作, 分为如下两种:

- Flip $l\ r$, 表示将 $cmp_l, cmp_{l+1}, \dots, cmp_r$ 全部**反转**, 即修改之后, 若 cmp_i ($l \leq i \leq r$) 原来为 0 则会变为 1; 若原来为 1 则会变为 0。

- Reverse $l\ r$, 表示将 $cmp_l, cmp_{l+1}, \dots, cmp_r$ 这个子区间**翻转**, 即修改之后, 该子区间将会变为 $cmp_r, cmp_{r-1}, \dots, cmp_l$ 。

请你在第一次修改操作之前以及每次修改操作之后, 求出满足 cmp 数组, 且字符集为 $\{a, b, c\}$ 的字符串数量。答案对 $10^9 + 7$ 取模。

4.2 输入

第一行两个整数 n, q , 表示字符串长度和修改次数。

第二行 $n - 1$ 个整数 cmp_1, \dots, cmp_{n-1} , 表示 cmp 数组。

接下来 q 行，每行一个字符串 s 和两个整数 l, r ，表示一次修改操作。若 $s = \text{Flip}$ ，则表示这是一次区间反转操作；若 $s = \text{Reverse}$ ，则表示这是一次区间翻转操作。 l, r 分别是两个操作参数，含义如题目描述中所示。

4.3 输出

$q + 1$ 行，每行一个整数表示答案。

4.4 输入输出样例 1

4.4.1 样例输入

```
3 3
1 0
Flip 2 2
Flip 1 1
Reverse 1 2
```

4.4.2 样例输出

```
8
4
5
8
```

4.4.3 样例解释

未操作时，根据 cmp 数组有 $rk_1 < rk_2, rk_2 > rk_3$ ，满足条件的字符串有：aba, abb, aca, acb, acc, bca, bcb, bcc，共 8 个。

第一次修改后， cmp 数组变为 $\{1, 1\}$ ，则 $rk_1 < rk_2 < rk_3$ ，满足条件的字符串有：aab, aac, abc, bbc，共 4 个。

第二次修改后， cmp 数组变为 $\{0, 1\}$ ，则 $rk_1 > rk_2, rk_2 < rk_3$ ，满足条件的字符串有：bab, bac, cab, cac, cbc，共 5 个。

第三次修改后， cmp 数组变为 $\{1, 0\}$ ，和未操作时一致。

4.5 输入输出样例 2

见下发文件中的 suffix2.in 和 suffix2.out。

该样例满足测试点 1 ~ 3 的数据范围和限制。

4.6 输入输出样例 3

见下发文件中的 suffix3.in 和 suffix3.out。

该样例满足测试点 4 ~ 6 的数据范围和限制。

4.7 输入输出样例 4

见下发文件中的 suffix4.in 和 suffix4.out。
该样例满足测试点 7 ~ 10 的数据范围和限制。

4.8 输入输出样例 5

见下发文件中的 suffix5.in 和 suffix5.out。
该样例满足测试点 15 ~ 20 的数据范围和限制。

4.9 数据范围

对于所有数据，满足 $1 \leq n \leq 3 \times 10^5$, $1 \leq q \leq 5 \times 10^4$, $cmp_i \in \{0, 1\}$ 。对于所有操作， $s \in \{\text{Flip}, \text{Reverse}\}$, $1 \leq l \leq r < n$ 。

测试点编号	n	q	特殊性质 A	特殊性质 B	特殊性质 C
1 ~ 3	≤ 500	≤ 100	否	否	否
4 ~ 6	$\leq 3 \times 10^5$	$\leq 5 \times 10^4$	是	否	否
7 ~ 10	$\leq 3 \times 10^5$	$\leq 5 \times 10^4$	否	是	否
11 ~ 14	$\leq 3 \times 10^5$	$\leq 5 \times 10^4$	否	否	是
15 ~ 20	$\leq 3 \times 10^5$	$\leq 5 \times 10^4$	否	否	否

特殊性质 A：对于所有操作，满足 $l = r$ 。

特殊性质 B：保证不存在 Flip 操作。

特殊性质 C：保证不存在 Reverse 操作。

提示： 请注意常数因子对程序效率的影响。为了尽量避免可能的常数问题，保证下发样例与评测时所用的对应数据范围的测试点的数据生成方式强度相同。