

2020 年全国青少年信息学奥林匹克 浙江省队选拔赛第一试

竞赛时间：6 月 20 日 9:00 – 14:00

题目名称	字符串	传统艺能	序列
目录	string	segment	seq
可执行文件名	string	segment	seq
输入文件名	string.in	segment.in	seq.in
输出文件名	string.out	segment.out	seq.out
每个测试点时限	3s	4s	1s
内存限制	512MB	512MB	512MB
测试点数目	10	10	10
每个测试点分值	10	10	10
是否有部分分	否	否	否
题目类型	传统型	传统型	传统型
是否有附加文件	是	是	是

提交源程序必须加后缀

对于 C++ 语言	string.cpp	segment.cpp	seq.cpp
对于 C 语言	string.c	segment.c	seq.c
对于 Pascal 语言	string.pas	segment.pas	seq.pas

编译开关

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	-O2	-O2	-O2

1 字符串

1.1 题目描述

Bob 喜欢字符串。

Bob 觉得，重复两遍的字符串是优美的，例如，`aa,sese,abcabc,baabaa,abababab` 是优美的串，而 `ab,aadead,sesese,abba` 不是。更具体地说，如果一个字符串 S 能够被写成两个相同的字符串前后拼接的形式，即存在字符串 P 使得 $S = PP$ ，那么 S 就是优美的。

Bob 有一个长度为 n 的字符串 $T = T_1T_2 \cdots T_n$ 。现在他想知道，给定 T 的一个子串 $Q = T[l \cdots r]$ ，这个子串 Q 内一共包含多少种本质不同的优美的串作为子串。如果两个串相同，但是出现的位置不同，那么这两个串不是本质不同。

Bob 一共有 q 组不同的询问，你需要快速计算出答案。

1.2 输入格式

第一行输入两个整数 n, q 。第二行输入一个只包含小写字母 `a` 和 `b` 的字符串 T 。

接下来 q 行，每行输入两个整数 l, r ，表示一组询问。

1.3 输出格式

输出 q 行，每行一个整数表示答案。

1.4 样例输入 1

```
11 5
aabaabaaaab
1 11
1 6
7 10
5 5
3 8
```

1.5 样例输出 1

```
5
2
2
0
2
```

1.6 样例解释 1

$|T|$ 有 aa,aaaa,abaaba,aabaab,baabaa 这些本质不同的优美的串。

1.7 样例输入输出 2

见下发文件

1.8 数据范围与约定

对于前 10% 的数据, $n \leq 100$ 。

对于前 20% 的数据, $n \leq 500$ 。

对于前 40% 的数据, $n \leq 5000$ 。

对于另外 20% 的数据, 保证 T 中所有的优美的串的个数不超过 10^6 , 这里位置不同的串被视为不同的。

对于另外 20% 的数据, $q = 1$ 。

对于 100% 的数据, $1 \leq n, q \leq 200000$, $1 \leq l \leq r \leq n$, T 只包含小写字母 a 和 b。

2 传统艺能

2.1 题目描述

Bob 喜欢线段树。

众所周知，ZJOI 的第二题有很多线段树。

Bob 有一棵根为 $[1, n]$ 的**广义线段树**。Bob 需要在这个线段树上执行 k 次区间**懒标记**操作，每次操作会等概率地从 $[1, n]$ 的所有 $\frac{n(n+1)}{2}$ 个子区间中随机选择一个。对于所有在该次操作中被访问到的非叶子节点，Bob 会将这个点上的标记下推；而对于所有叶子节点（即没有继续递归的节点），Bob 会给这个点打上标记。

Bob 想知道， k 次操作之后，有标记的节点的期望数量是多少。

2.2 具体定义

线段树：线段树是一棵每个节点上都记录了一个线段的二叉树。根节点记录的线段是 $[1, n]$ 。对于每个节点，若它记录的线段是 $[l, r]$ 且 $l \neq r$ ，取 $m = \lfloor \frac{l+r}{2} \rfloor$ ，则它的左右儿子节点记录的线段分别是 $[l, m]$ 和 $[m+1, r]$ ；若 $l = r$ ，则它是叶子节点。

广义线段树：在广义的线段树中， m 不要求恰好等于区间的中点，但是 m 还是必须满足 $l \leq m < r$ 的。不难发现在广义的线段树中，树的深度可以达到 $O(n)$ 级别。

线段树的核心是**懒标记**，下面是一个带懒标记的广义线段树的伪代码，其中 `tag` 数组为懒标记：

```
1: function PUSHDOWN(Node)
2:   if tag[Node] = 1 then
3:     tag[Lson(Node)] ← 1
4:     tag[Rson(Node)] ← 1
5:     tag[Node] ← 0
6:   end if
7: end function
8:
9: function MODIFY(Node, l, r, ql, qr)
10:  if  $[l, r] \cap [ql, qr] = \emptyset$  then
11:    return
12:  end if
13:  if  $[l, r] \subseteq [ql, qr]$  then
14:    tag[Node] ← 1
15:    return
16:  end if
17:  m ← GETM(Node)
18:  PUSHDOWN(Node)
19:  MODIFY(Lson(Node), l, m, ql, qr)
20:  MODIFY(Rson(Node), m+1, r, ql, qr)
21: end function
```

注意，在处理叶子节点时，一旦他获得了一个标记，那么这个标记会一直存在。

你也可以这么理解题意：有一棵广义线段树，每个节点有一个 m 值。一开始 tag 数组均为 0，Bob 会执行 k 次操作，每次操作等概率随机选择区间 $[l, r]$ 并执行 $\text{MODIFY}(\text{root}, 1, n, l, r)$ 。最后所有 Node 中满足 $\text{tag}[\text{Node}] = 1$ 的期望数量就是要求的值。

2.3 输入格式

第一行输入两个整数 n, k 。

接下来输入一行包含 $n - 1$ 个整数 a_i ：按照中序遍历的顺序，给出广义线段树上所有非叶子节点的划分位置 m 。你也可以理解为从只有 $[1, n]$ 根节点开始，每次读入一个整数后，就将当前包含这个整数的节点做一次拆分，最后获得一棵有 $2n - 1$ 个节点的广义线段树。

保证给定的 $n - 1$ 个整数是一个排列，不难发现通过这些信息就能唯一确定一棵 $[1, n]$ 上的广义线段树。

2.4 输出格式

输出一行一个整数，代表期望数量对 $p = 998244353$ 取模后的结果。即，如果期望数量的最简分数表示为 $\frac{a}{b}$ ，你需要输出一个整数 c 满足 $c \times b \equiv a \pmod{p}$

2.5 样例输入 1

```
3 1
1 2
```

2.6 样例输出 1

```
166374060
```

2.7 样例解释 1

输入的线段树为 $[1, 3], [1, 1], [2, 3], [2, 2], [3, 3]$ 。

若操作为 $[1, 1]/[2, 2]/[3, 3]/[2, 3]/[1, 3]$ ，标记个数为 1。若操作为 $[1, 2]$ ，标记个数为 2。故答案为 $\frac{7}{6}$

2.8 样例输入 2

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

2.9 样例输出 2

320443836

2.10 样例输入输出 3

见下发文件。

2.11 数据范围与约定

测试点	n	k	其他约定
1	≤ 10	≤ 4	无
2	≤ 10	≤ 100	
3	≤ 5	无	
4	无	$=1$	
5	$= 32$	无	输入的线段树为完全二叉树
6	$= 64$		
7	$= 4096$		每个 m 均在 $[l, r - 1]$ 内均匀随机
8	≤ 5000		
9	≤ 100000		无
10	无		

对于 100% 的数据, $1 \leq n \leq 200000, 1 \leq k \leq 10^9$ 。

3 序列

3.1 题目描述

Bob 喜欢序列。

有一个长度为 n 的非负整数序列 a_1, a_2, \dots, a_n 。每一步你可以从以下三种操作中选择一种执行：

- 选择一个区间 $[l, r]$ ，将下标在这个区间里的所有数都减 1。
- 选择一个区间 $[l, r]$ ，将下标在这个区间里且下标为奇数的所有数都减 1。
- 选择一个区间 $[l, r]$ ，将下标在这个区间里且下标为偶数的所有数都减 1。

求最少需要多少步才能将序列中的所有数都变成 0。

3.2 输入格式

第一行输入一个整数 T ，表示数据组数。

对于每组数据，第一行输入一个整数 n ，接下来一行输入 n 个非负整数 a_1, a_2, \dots, a_n 。

3.3 输出格式

输出 T 行，对于每组测试数据，输出一行一个整数，表示答案。

3.4 样例输入 1

```
3
5
2 1 2 1 2
8
1000000000 1000000000 0 1000000000 1000000000 0 1000000000 1000000000
13
1 1 4 5 1 4 1 9 1 9 8 1 0
```

3.5 样例输出 1

```
2
30000000000
19
```

3.6 样例解释 1

第一组数据: $21212 \xrightarrow{1} 11111 \xrightarrow{1} 00000$

第三组数据: $1145141919810 \xrightarrow{1} 0034030808700 \xrightarrow{8} 0031000000700 \xrightarrow{10} 0000000000000$

3.7 样例输入输出 2

见下发文件

3.8 数据范围与约定

对于前 10% 的数据, $n \leq 5, a_i \leq 10$ 。

对于前 30% 的数据, $n \leq 50, a_i \leq 50$ 。

对于前 50% 的数据, $n \leq 200, a_i \leq 200$ 。

对于前 60% 的数据, $n \leq 200, a_i \leq 10^9$ 。

对于前 70% 的数据, $n \leq 1000, a_i \leq 10^9$ 。

对于前 90% 的数据, $n \leq 10000, a_i \leq 10^9$ 。

对于 100% 的数据, $1 \leq n \leq 100000, 0 \leq a_i \leq 10^9, 1 \leq T \leq 10$ 。