AtCoder Beginner Contest 455 题目讲解

云浅

2026-4-27

Outline

A. 455

A. 455

略

B. Spiral Galaxy

B. Spiral Galaxy

略

C. Straw Millionaire

C. Straw Millionaire

给定一个整数序列 $A = (A_1, A_2, ..., A_N)$。

在执行恰好 K 次以下操作后，求 A 所有元素之和的最小可能值。

每次操作：

- 选择一个整数 x。
- 对于所有满足 $A_i = x$ 的 i，将 A_i 的值替换为 0。

数据范围： $1 \leq K \leq N \leq 3 \times 10^5, 1 \leq A_i \leq 10^9$

C. Straw Millionaire

把 A_i 相同的分成一类，求出每一类内部的元素和，假设有 m 类，元素和分别是 $S_1, S_2, \cdots, S_m$，那么答案就是 $\sum A_i$ 减去 $S_1, \cdots, S_m$ 中前 K 大的元素之和。

可以直接排序做到 $O(N \log N)$，也可以用 kth-element 的技巧做到 $O(N)$。

D. Card Pile Query

D. Card Pile Query

有 N 张牌和 N 堆牌。

牌和堆的编号均为 $1, 2, ..., N$。初始时，第 i 堆只包含牌 i。

按顺序对每个 $i = 1, 2, ..., Q$ 执行以下操作：

- 将牌 C_i 以及堆叠在牌 C_i 之上的所有牌（保持它们原有的顺序）移动到牌 P_i 的顶部。保证在执行操作之前，牌 C_i 和牌 P_i 位于不同的堆中，且牌 P_i 位于某堆的顶部。

在所有操作完成后，求每堆牌中的牌数。

数据范围： $1 \leq N, Q \leq 3 \times 10^5$

D. Card Pile Query

考虑用链表来维护，每个 x 维护 x 前面的元素 $\mathrm{pre}(x)$。

每次修改直接把 $\mathrm{pre}(C_i)$ 改成 P_i 即可。

为了求出最后每个牌堆的牌数，可以额外对每个 x 维护一个值 $\mathrm{id}(x)$ 表示：如果 x 不在某个牌堆顶，那么 $\mathrm{id}(x) = 0$，否则 $\mathrm{id}(x)$ 是 x 所在的牌堆的编号。

最后把所有的 $(x, \mathrm{pre}(x))$ 连边，在得到的图上 DFS 一遍求出连通块就可以知道每个牌堆的牌数了。

E. Unbalanced ABC Substrings

E. Unbalanced ABC Substrings

给定一个长度为 N 的字符串 S，由字符 A、B、C 三种字符组成。

S 共有 $\frac{N(N+1)}{2}$ 个非空子串。求其中满足以下条件的子串的数量：

- 子串中 A、B、C 的出现次数两两不同。

注意：即使两个子串作为字符串相同，只要它们在 S 中的位置不同，就被视为不同的子串。

数据范围：

- $1 \leq N \leq 2 \times 10^5$

E. Unbalanced ABC Substrings

考虑容斥，用总数 $\frac{N(N+1)}{2}$ 减去 AB 出现次数相同的，再减去 BC 相同的，再减去 AC 相同的。这个时候发现 ABC 全都相同的被我们加了 1 次，但是减掉了 3 次，所以还需要再加 2 次补回来。

算出 A_i, B_i, C_i 分别表示 $S[1, i]$ 中 ABC 的出现次数，考虑怎么算 AB 出现次数相同的，那就相当于 $A_r - A_l = B_r - B_l$ 的 (l, r) 个数，其中 $0 \le l < r \le N$。化简一下得到 $A_r - B_r = A_l - B_l$，那么开一个桶就可以维护了。

BC 和 AC 都是类似的，对于 ABC 出现次数都相同的，等价于 $A_r - A_l = B_r - B_l = C_r - C_l$，即 $(A_r - B_r, A_r - C_r) = (A_l - B_l, A_l - C_l)$，同样也可以用哈希表维护。

时间复杂度可以做到 $O(N)$。

F. Merge Slimes 2

F. Merge Slimes 2

有一个长度为 N 的非负整数序列 A，初始时所有元素均为 0。按顺序处理 Q 个查询。

对于第 q 个查询，给定区间 $[l_q, r_q]$ 和一个正整数 a_q。按顺序执行以下操作：

1. 将 a_q 加到 $A_{l_q}, A_{l_q+1}, ..., A_{r_q}$ 中的每一个上。
2. 定义 $B = (B_1, B_2, ..., B_M) = \left(A_{l_q}, A_{l_q+1}, ..., A_{r_q}\right)$。然后求解以下问题：

- 有 M 个史莱姆 $1, 2, ..., M$，其中第 m 个史莱姆的重量为 B_m。
- 重复进行 $M-1$ 次操作，每次操作选择两个史莱姆并将其合并。
- 当合并重量为 x 和 y 的史莱姆时，会得到一个重量为 $x+y$ 的新史莱姆，原来的两个史莱姆消失。此次合并的代价为 $x \times y$。
- 求在所有可能的合并顺序中，操作总代价的最小值，结果对 998244353 取模。

注意：每个查询中对 A 的修改会影响到后续查询。

数据范围： $1 \le N \le 10^5, 1 \le Q \le 10^5, 1 \le a_i \le 10^9$

F. Merge Slimes 2

不管按什么顺序合并，代价都是唯一的：一定是 $\sum_{i<j} a_i a_j$。

证明就考虑 $(a_1 + \cdots + a_n)(b_1 + \cdots + b_m) = \sum a_i b_j$，所以任意一对 (i, j) 一定会在第一次被合并的时候恰好产生一次 $a_i a_j$ 的贡献。

那么就变成：区间加，查询区间的 $\sum_{i<j} a_i a_j$。

考虑线段树维护，那么加 x 带来的影响就是 $\sum_{i<j}(a_i + x)\big(a_j + x\big) = \sum_{i<j} a_i a_j + (n-1)x \sum a_i + \frac{n(n-1)}{2} x^2$，那么多维护一个区间和即可。

时间复杂度 $O(N + Q \log N)$

G. Balanced Subarrays

G. Balanced Subarrays

给定一个由 N 个正整数组成的序列 A。

在 A 的所有 $\frac{N(N+1)}{2}$ 个非空连续子数组中，一个序列 X 被称为**平衡序列**，当且仅当它满足以下条件：

- 在 X 中出现的每个整数，在 X 中出现的次数相同。

对于每个 $k = 1, 2, ..., K$，求以下两个值：

1. 每个整数恰好出现 B_k 次的平衡序列的数量。
2. 恰好有 B_k 个不同整数出现的平衡序列的数量。

注意：即使两个子数组作为序列相同，只要它们在 A 中取自不同的位置，就被视为不同的子数组。

数据范围：$1 \le N \le 2 \times 10^5, 1 \le K \le 10$

G. Balanced Subarrays

第一问：考虑每个数 x 的约束，相当于它要么出现 0 次，要么出现 B_k 次。

那么枚举左端点 l，对每个数 x 找到它在 l 后面的第一次出现、第 B_k 次出现、第 B_k+1 次出现的位置 a, b, c，相当于要么 $r < a$，要么 $b \le r < c$。

合法的 r 必须要满足所有 r 的约束，那么我们维护一个线段树，把 $[l, a)$ 以及 $[b, c)$ 这段区间 $+1$，然后查询最大值以及最大值个数即可。如果最大值恰好等于颜色的个数，就把最大值个数加到答案里，否则贡献是 0。

这样是单次 $O(N \log N)$ 的。

G. Balanced Subarrays

另一种做法是哈希，我们考虑先求出每个 l 对应的最大的 $r_{\max}$，使得 $[l, r_{\max}]$ 中每种颜色的出现次数不超过 B_k。那么现在合法就只需要每种颜色的出现次数都是 B_k 的倍数。

现在取一个大质数 M，对每种颜色，我们随机取 B_k 个数 $x_1, x_2, \cdots, x_{B_k}$，使得它们的和 $\bmod M$ 为 0。接下来依次循环地把 x_i 赋值给这种数的每次出现位置。

那么此时一个区间所有数的出现次数都是 B_k 的倍数，必要条件是这个区间的哈希值之和 $\bmod M$ 为 0。反过来如果区间哈希值 $\bmod M$ 为 0，那么不合法的概率只有 $\frac{1}{M}$.

于是只需要计算多少个 $r \le r_{\max}$ 满足 $[l, r]$ 的区间和 $\bmod M$ 为 0，这等价于 $S_{l-1} = S_r$，其中 S 是哈希值的前缀和。注意 $r_{\max}$ 是不断减小的，那么可以直接双指针的同时维护一个哈希表，可以做到 $O(N)$。

G. Balanced Subarrays

第二问：还是考虑枚举 l，那么使得 $[l,r]$ 颜色数恰好为 B_k 的 r 构成一段区间，这段区间可以用双指针维护出来；进一步，实际上我们还能知道 $[l,r]$ 中具体有哪些颜色：就是 l 后面出现位置最靠前的前 B_k 种颜色。

现在只需要判断有多少个 r 满足这 B_k 种颜色的出现次数相同。可以哈希，给每种颜色 c 赋一个随机的哈希值 P_c，然后我们可以求出来这 B_k 种颜色的哈希值之和 H，那么区间 $[l,r]$ 合法，就需要这个区间的哈希值之和恰好为 $\frac{r-l+1}{B_k} \times H$。

考虑求一个前缀和 $S_i = \sum_{j=1}^{i} P_{A_i}$ 表示前缀哈希值之和，那么合法当且仅当 $S_r - S_{l-1} = \frac{r-l+1}{B_k} \times H$，即 $S_r \times B_k - r \times H = S_{l-1} \times B_k - (l-1) \times H$。用一个哈希表来存哈希值即可。

这个也可以做到 $O(N)$。

G. Balanced Subarrays

- Bonus Problem: Consider how to count the number of balanced sequences in $O\left(N\sqrt{N}\right)$ time.

这个只需要注意到合法的区间要么数的种类数不超过 $\sqrt{N}$，要么所有数的出现次数不超过 $\sqrt{N}$，那么我们对 $B_k = 1, 2, \cdots, \sqrt{N}$ 都跑一遍上面两个算法就好了。对于算重的问题，我们可以在跑第二问的时候强制要求 $r - l + 1 > B_k \times \sqrt{N}$。