Day5 G~I题解

Setsuna

2019.7.19

H

每次交换相邻两数，使得原序列有序的最少操作次数就是原序列的逆序对数。

证明：如果不存在相邻的两个数逆序，那么显然原序列是有序的。
如果存在，那我们可以交换他们，使得逆序对-1。

可以使用归并排序或者离散化+树状数组在O(nlogn)内求出逆序对数。

答案就是min(x,y)*逆序对数

|

这题和线性基没啥关系。

把 n 个数转换成二进制插入 trie 。

那么找 a_i 异或的最大值可以在 trie 上贪心地找。

具体地来说。

设 a_i 的二进制从前往后第 k 位是 c。

我们要选择 $c \oplus 1$ 才能使异或值更大，如果 $c \oplus 1$ 儿子存在，那我们就走 $c \oplus 1$ 儿子，否则只能走 c 儿子。

单次时间复杂度$O(\log a_i)$

G

这是一道考验线段树基础和代码能力的毒瘤题。

AC此题可以进一步帮助你理解线段树的区间操作。

G

如果只有区间加，考虑如何维护区间平方和与立方和。
假设给区间 $[l,r]$ 加上 x 的操作。

区间和：

$$\sum_{i=l}^{r}(a_i+x)=\sum_{i=l}^{r}a_i+(r-l+1)*x$$

G

区间平方和:

$$\sum_{i=l}^{r}(a_i+x)^2=\sum_{i=l}^{r}(a_i^2+2*x*a_i+x^2)$$

$$=\sum_{i=l}^{r}a_i^2+2*x*\sum_{i=l}^{r}a_i+(r-l+1)*x^2$$

G

区间立方和:

$$\sum_{i=l}^{r}(a_i+x)^3 = \sum_{i=l}^{r}(a_i^3+3*x*a_i^2 + 3*x^2*a_i + x^3)$$

$$= \sum_{i=l}^{r} a_i^3 + 3*x*\sum_{i=l}^{r} a_i^2 + 3*x^2*\sum_{i=l}^{r} a_i + (r-l+1)*x^3$$

G

目前为止还不是很复杂。

如果我们加入区间乘法呢…

假设给区间 $[l,r]$ 乘上 x 的操作，好像直接分别拿 x,x^2,x^3 乘以区间的和就可以了？

看上去没问题，但是我们有 lazytag，我们不知道是加法先来的还是乘法先来的，如果胡乱的下推标记可能导致运算的顺序混乱。

考虑人工定义一个优先级，优先级高的标记应该能影响优先级低的标记，以此来体现操作先后带来的差异。

G

如果先推加法标记，那么它应该能影响到之前来的乘法标记。

不然推完加法再推乘法标记时，就会对加完的结果产生影响，我们应该试图修改乘法标记的值来抵消这种影响。

设当前的乘法标记是 mul ，加法标记是 add 。以区间和为例。

G

正常的顺序:

$$sum * mul + (r - l + 1) * add$$

先推加法，再推乘法:

$$(sum + (r - l + 1) * add) * mul$$

所以我们应该把乘法标记修正为:

$$\frac{sum * mul + (r - l + 1) * add}{sum + (r - l + 1) * add} \to mul$$

出现了除法？？原因是用乘法抵消加法带来的影响很困难，我们尝试先推乘法标记。

G

正常的顺序:

$$(sum + (r - l + 1) * add) * mul$$
$$= sum * mul + (r - l + 1) * add * mul$$

先推乘法，再推加法:

$$sum * mul + (r - l + 1) * add$$

我们只需要把加法标记重新修正:

$$add * mul \rightarrow add$$

这时候对于先来的加法，后来的乘法这种情况，先推乘法标记再推加法标记就正确了。

G

那推完区间和，这个性质对区间平方和与立方和有没有用呢?

不妨试着推一下。

正确的顺序:

$$\left(\sum_{i=l}^{r} a_i^2 + 2 * add * \sum_{i=l}^{r} a_i + (r-l+1) * add^2\right) * mul^2$$

$$= mul^2 * \sum_{i=l}^{r} a_i^2 + mul^2 * 2 * add * \sum_{i=l}^{r} a_i + mul^2 * (r-l+1) * add^2$$

G

如果我们 $mul * add \to add$

再用先推乘法的顺序计算一遍：

$$\sum_{i=l}^{r}(mul * a_i)^2 + 2 * add * mul * \sum_{i=l}^{r} mul * a_i + (r - l + 1) * add^2 * mul^2$$

$$= mul^2 * \sum_{i=l}^{r} a_i^2 + mul^2 * 2 * add * \sum_{i=l}^{r} a_i + mul^2 * (r - l + 1) * add^2$$

完 全 一 致

G

立方和用同样的方法推一遍，发现这个性质依然是成立的。

那么乘法我们也做完了。

接下来考虑区间set的操作怎么处理。

发现前面来的加和乘操作对后面的set操作不应该有任何的影响，也就是说set操作来的时候把已经有的加标记和乘标记清除就行了。

总的优先级：set>乘法>加法

G

然后看set对答案的影响，考虑操作为 $[l,r]$ set 成 x

$$\sum_{i=l}^{r} a_i \Rightarrow \sum_{i=l}^{r} x = (r-l+1)*x$$

$$\sum_{i=l}^{r} a_i^2 \Rightarrow \sum_{i=l}^{r} x^2 = (r-l+1)*x^2$$

$$\sum_{i=l}^{r} a_i^3 \Rightarrow \sum_{i=l}^{r} x^3 = (r-l+1)*x^3$$

G

于是这道题就顺利地做完了。

如果对过程中 lazytag 的先后顺序带来的影响不明白的可以拿个小数据手玩一下，加深对操作优先级的理解。