More Queries to Array...题解

唐适之

算法1

赋值操作O(n)扫描赋值，查询操作O(nk)扫描查询。

时间复杂度O(mnk)，期望得分10分。

算法2

在算法1的基础上预处理i的0至5次方（1⩽i⩽n），使查询操作复杂度降低到O(n)。这30%的数据k较大、询问较多较长使算法一不能通过。

时间复杂度O(mn)，期望得分30分。

算法3、4

我们可以对0⩽j⩽5维护$i^j a_i$的区间和（1⩽i⩽n）。考虑一次询问l,r,k，例如k=2，询问的和为$\sum_{i=l}^{r} a_i \cdot (i-(l-1))^2 = \sum_{i=1}^{r} a_i \cdot i^2 - \sum_{i=1}^{r} a_i \cdot 2i(l-1)^2 + \sum_{i=1}^{r} a_i \cdot (l-1)^2$，便转换为分别询问0、1、2次$i^j a_i$的区间和。一般地，展开后用i整理可以将询问转换为0至k次的区间和，以此可以实现在知道$i^j a_i$的区间和的情况下O(k)查询。预处理i^j（1⩽i⩽n,0⩽j⩽5）的前缀和，即可实现插入中O(k)算出一个区间0至5次幂的区间和。以上是在假设维护了区间和的情况下，现在使用数据结构维护。

算法3：使用分块维护区间和，时间复杂度为$O(m\sqrt{n}k)$，空间复杂度O(n)，按常数不同，期望得分50至100分。

算法4：使用线段树维护区间和，时间复杂度$O(mk\log_2 n)$，空间复杂度O(n)，期望得分100分，这是最优做法。

算法5

如果没有预处理i^j（1⩽i⩽n,0⩽j⩽5）的前缀和，那么插入操作就稍微麻烦一点。如果k⩽2，1次和2次情况可以分别使用等差数列和公式和平方和公式O(1)算出一个区间的区间和。

这部分期望得分10分。

算法6

由算法5拓展，可以推出0至5次方和的公式。例如立方和公式可以按下面的方法推出：

$$(r+1)^4 - r^4 = 4r^3 + 6r^2 + 4r + 1$$

$$r^4 - (r-1)^4 = 4(r-1)^3 + 6(r-1)^2 + 4(r-1) + 1$$

$$\ldots$$

$$(l+1)^4 - l^4 = 4l^3 + 6l^2 + 4l + 1$$

各式相加得

$$S_3 = \frac{1}{4}((r+1)^4 - l^4 - 6S_2 - 4S_1 - S_0)$$

一般地，可用更低次的区间和表示出较高次的区间和（S_k表示k次和），即

$$S_k = \frac{1}{k+1}((r+1)^{k+1} - l^{k+1} - \sum_{p=0}^{k-1} C_{k+1}^{p} S_p)$$

这样求一段区间各次的区间和就要$O(k^2)$的时间，使用线段树的话插入$O(mk^2 \log_2 n)$，查询$O(mk \log_2 n)$，总时间$O(mk^2 \log_2 n)$，空间复杂度$O(n)$。这样的时间可以被接受，期望得分100分。