**复杂度证明**

我们将初值视为N次插入来讨论，不影响复杂度的证明，现在我们有了一颗纯色的线段树。

对于一次赋值操作，有O（log）的节点会被遍历到，我们考虑一个杂色节点出现的充要条件是至少包含2种颜色，也就是说有两种颜色的分界线位于其区间内。每次赋值操作最多增加左右两处分界，最多只有O(log)的节点由纯色变为杂色。

对于查询操作。用O（log）的时间确定恰好覆盖区间的O（log）个节点，当且仅当其中包含杂色节点时会有更多的复杂度，由于查询后就会对原区间进行赋值操作，所以恰好覆盖区间的所有节点都会变成纯色节点，即每个杂色节点最多贡献1的复杂度。

即修改会造成O（log）的实际复杂度和O（log）的势能，查询会有O（log + 杂色点数量）的实际复杂度 ，随后的赋值减少 O（查询时杂色点数量）的势能。势能任何时刻非负。

综上所述，算法的复杂度为O(NlogN)。