Segment Tree

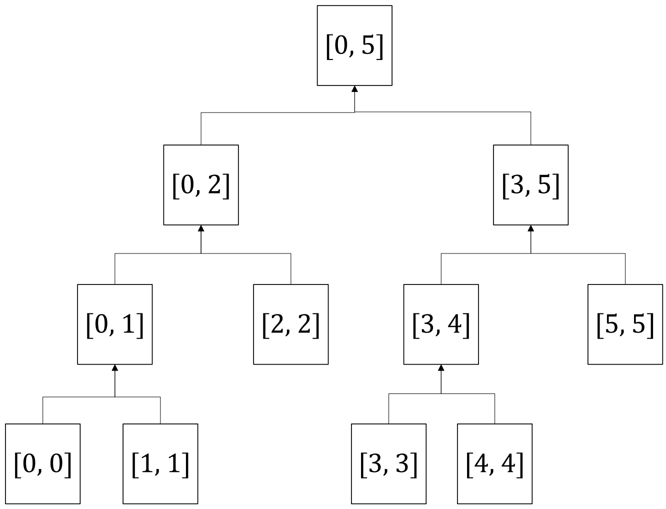
线段树

描述：

线段树是一种二叉树，它将长度为 n 数组$s[0, n-1]$划分成区间，其中每个区间对应线段树上的一个节点。线段树中的每个非叶子节点$[a, b]$表示该区域上被关心的值，例如数组 s 上该区间所有元素的和，最小元素的值，最大元素的值，第 k 大的值等。

在本节中我们计算该区域上所有元素的和，即节点$[a, b]$代表数组$s[a, b]$的和。其左子树表示$s[a, \frac{a+b}{2}]$的和，右子树表示区域$s[\frac{a+b}{2}+1, b]$的和。对于叶子节点$[i, i]$（其中$0 \le i \le n-1$），它没有孩子节点，表示的区域长度为 1。

比如线段树$s[0, 5]$如图所示：



构造操作：从根节点开始，递归的将节点$[a, b]$拆分为$[a, \frac{a+b}{2}]$和$[\frac{a+b}{2}+1, b]$，其中$0 \le a < b \le n-1$，父节点所代表的区域和等于左右孩子节点代表的区域和之和，即$sum[a, b] = sum\left[a, \frac{a+b}{2}\right] + sum[\frac{a+b}{2}+1, b]$，重复该操作直到叶子节点为止。该操作的时间复杂度为$O(N)$。

单点更新操作：修改数组 s 中任意一个值$s[i]$（其中$0 \le i \le n-1$），则包括该值的所有节点，从叶子节点一直到它的所有根节点和祖先节点，都需要修改。该操作的时间复杂度为$O(log_2 N)$。

查询操作：从根节点向下依次查询所有子节点，若节点属于被查询的区域则直接返回；若节点中只有一部分区域匹配则继续查询其左右子节点。最终将所有匹配到的区域的和加起来即为查询区域的和。该操作的时间复杂度为$O(log_2 N)$。

对于长度为 n 的数组$s[0, n-1]$，为了方便我们通过数组 t 来表示二叉树，下标为i的左孩子节点下标为$2i+1$，右孩子节点下标为$2i+2$。则$t[0]$为二叉树的根节点，代表数组 s 中$[0, n-1]$区域的和；其左孩子节点为$t[1]$，代表数组$s[0, \frac{n}{2}]$区域的和；右孩子节点为$t[2]$，代表数组$s[\frac{n}{2}+1, n-1]$区域的和；以此类推。