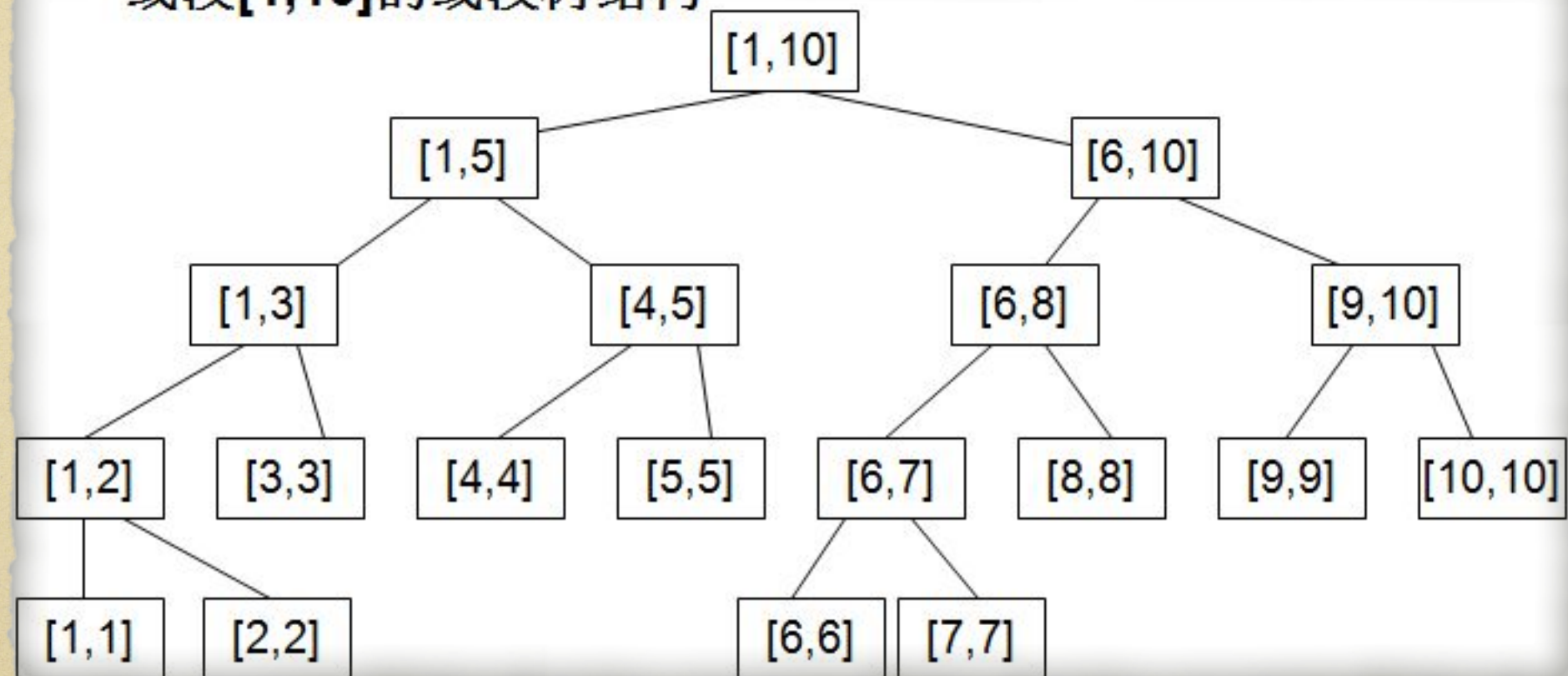


线段树

线段[1,10]的线段树结构



建树

- 1: 直接开四倍空间
- 2: 动态开点, 两倍空间
- 3: 小清新线段树, $ID(l,r) = (l + r \mid l \neq r)$, 两倍空间


```
struct Tree
```

```
{
```

```
    Tree *lc;
```

```
    Tree *rc;
```

```
    int mx, l, r;
```

```
    Tree(int l=0, int r=0): l(l), r(r) {
```

```
        mx = 0;
```

```
        if(l == r) {
```

```
            return ;
```

```
        }
```

```
        int m = l + r >> 1;
```

```
        lc = new (pool++)Tree(l, m);
```

```
        rc = new (pool++)Tree(m + 1, r);
```

```
    }
```

```
    void insert(int p, int v) {
```

```
        if(l == r) {
```

```
            mx += v;
```

```
            return ;
```

```
        }
```

```
        int m = l + r >> 1;
```

```
        if(p <= m) {
```

```
            lc->insert(p, v);
```

```
        } else {
```

```
            rc->insert(p, v);
```

```
        }
```

```
        mx = std::max(lc->mx, rc->mx);
```

```
    }
```

```
    int query() {
```

```
        if(l == r) {
```

```
            return l;
```

```
        }
```

```
        if(lc->mx == mx) {
```

```
            return lc->query();
```

```
        } else {
```

```
            return rc->query();
```

```
        }
```

```
    }
```

```
}node[1 << 18], *tree;
```


单点修改

- 区间线段树
- 值域线段树
- 离散化

区间修改

- 懒惰操作，每次等到需要用到下面的节点的时候再往下走，将标记下传
- 注意要先下传标记再往下走，以确保信息是真实的

区间询问

- 本质上是将一个区间拆分成若干个子区间，相邻子区间的信息可以合并，每个子区间对应了线段树上的一个节点

多种标记如何和谐共处

- 区间加，区间乘
- 一个数可以表示为 $x * mul + add$
- 新来一个乘法标记，给两个标记都乘上
- 新来一个加法标记，给加标记单独加
- 一个点同时有两种标记，乘法标记先下传

区间最大子段和

- 每个节点维护三个值，区间最大子段和，前缀最大子段和，后缀最大子段和
- 合并信息的时候除了考虑左右儿子的最大子段和，还需要考虑左儿子的最大后缀和与右儿子的最大前缀和之和

子树操作

➤ 两种操作

➤ 1:给一颗子树的点都加一个值

➤ 2:求子树和或者某个节点的值

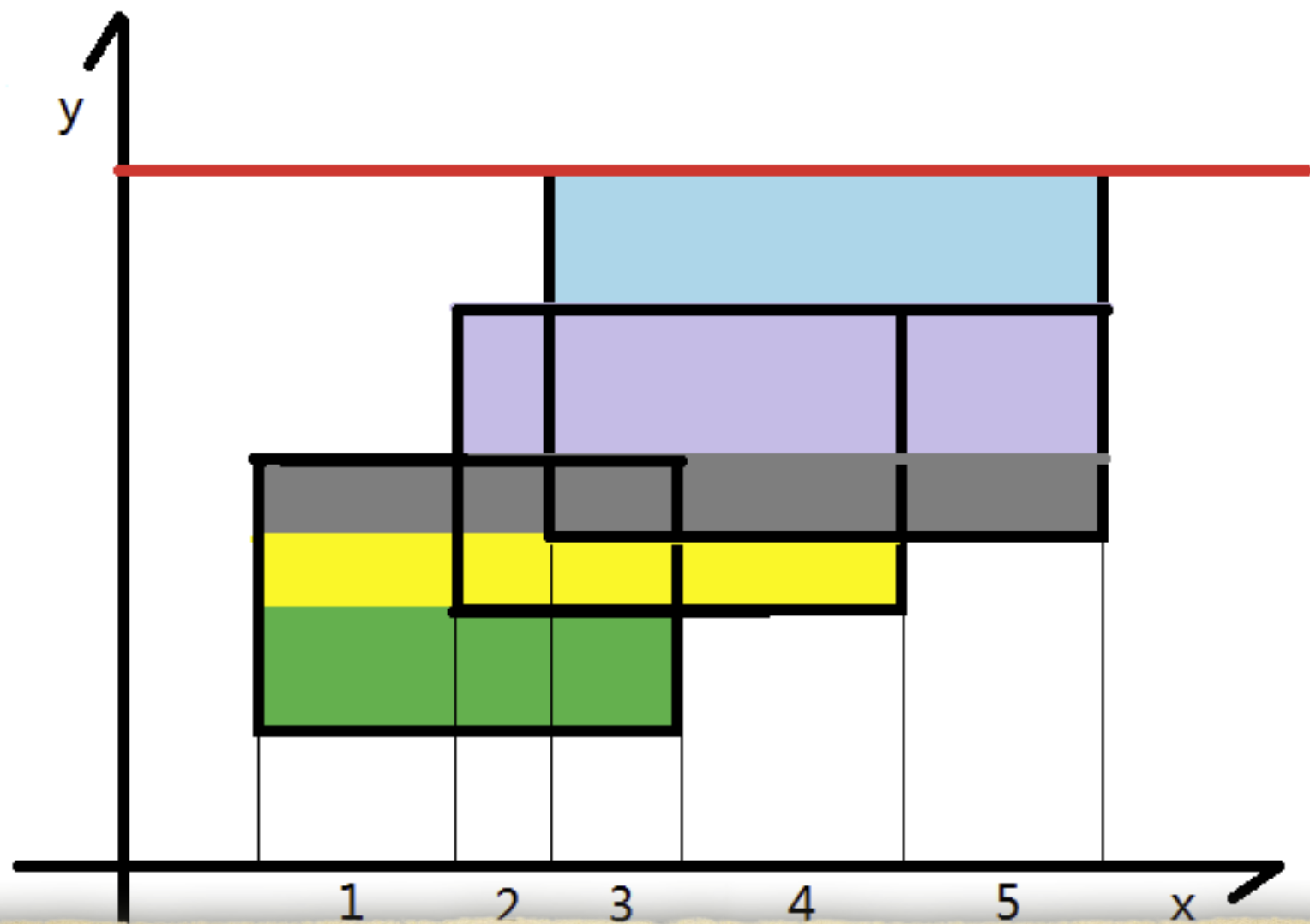
➤ 求出dfs序后对子树的操作相当于对区间的操作，转换成普通的线段树区间修改，区间询问

扫描线系列

- 矩形面积并
- 矩形面积交
- 矩形周长并
- 数星星

矩形面积并

- 每个矩形可以分成上下两根线段，我们可以将下面那根标记成1，上面标记成-1，并记录高度
- 从低到高考虑所有的线段，区间加，区间求和，每次计算相邻高度内的面积



矩形面积交

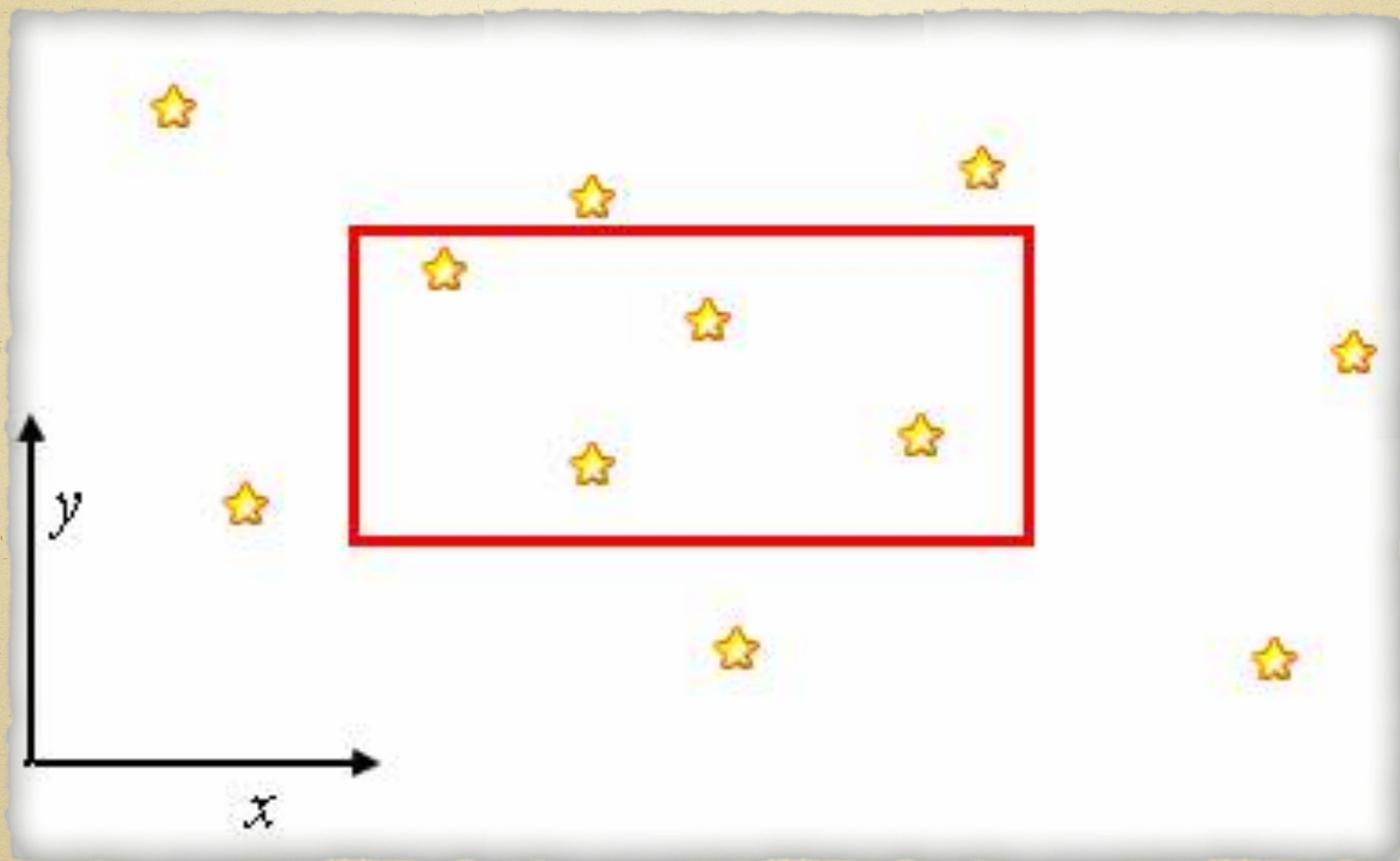
- 求覆盖至少 K 次以上的矩形的面积($K \leq 10$)
- 每个节点维护区间内覆盖 i 次的区间和
- 合并左右儿子的时候加上当前节点的覆盖次数，相当于将数组顺移覆盖次数位
- 比如左右儿子覆盖3次的长度和为 x ，当前节点有一个懒惰标记为 y ，那么覆盖 $3+y$ 次的长度和为 x

矩形周长并

➤ 利用扫描线的知识自己思考

数星星

- 平面内有很多的坐标点，每个点有一个权值，用一个平行于坐标轴的 $W \times H$ 的矩形去套，能套的最大权值和是多少



- 最优的矩形的边界上一定可以碰到某颗星星
- 那么我们将每个点看成是一条往右 W 长度的线段
- 问题就转换为高度 H 以内，线段的最大覆盖次数，左右边界的限制被我们去掉了，上下边界可以通过排序解决
- 从低到高枚举所有点，插入一个点就是插入一条线段，维护two-pointer，如果两头的点的高度差大于 H ，要将首部的一些元素删除
- 区间的最大覆盖次数就是答案