HW6

521021910522 郭俊甫

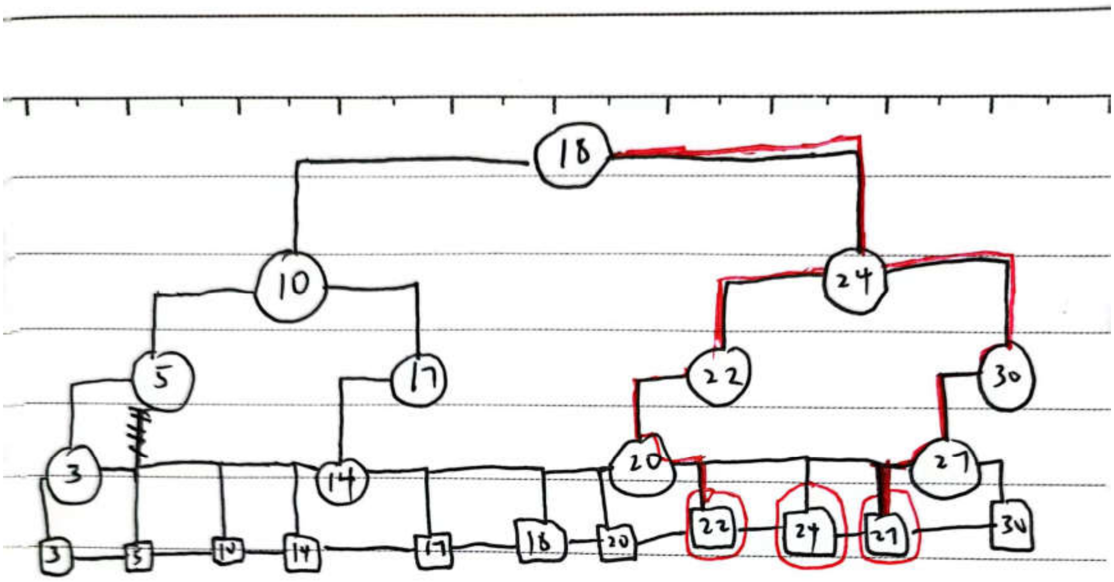
Exercise 1:

我们可以将二维平面上的点构建成 KD-Tree，然后从根节点开始查找，查找方式如下:

若该点的划分维度在矩形对应维度范围内，则将其左子树和右子树都纳入查找范围；若不在，则将该节点划分维度靠近矩形区域那侧的子树纳入查找范围，依次递归。则时间复杂度为 O(logn),空间复杂度为 O(n)，n 为节点总个数。

Exercise2:

首先以中位节点为根节点递归建立如下的二叉查找树:



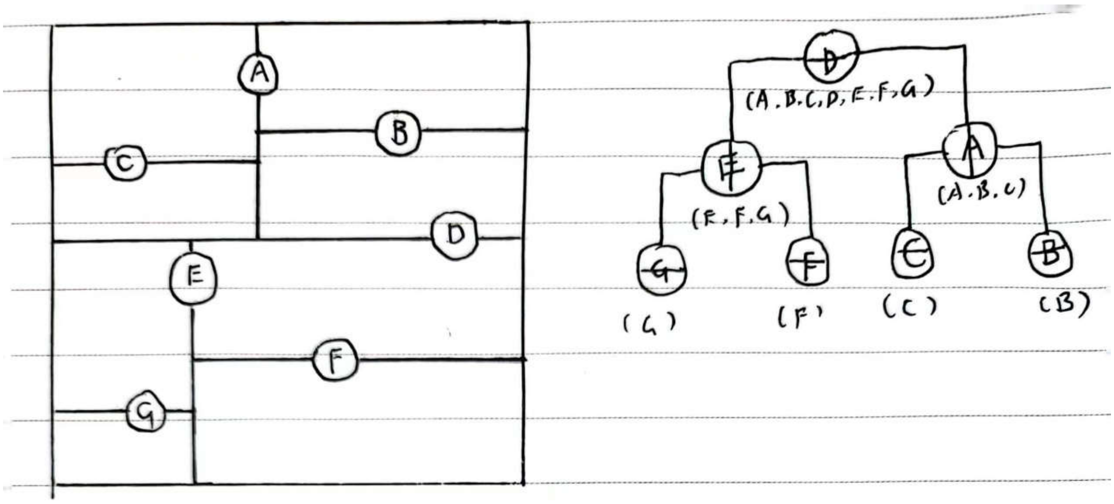
然后按如下方式进行查找:

若当前节点在查找范围内，则向左右子树均进行查找；若当前节点不在所查找范围内，则向靠近查找范围一侧的子树进行查找。由于此方法为二叉查找，且都要对查找到的节点进行输出，因此时间复杂度为 O(logn+r)

Exercise3:

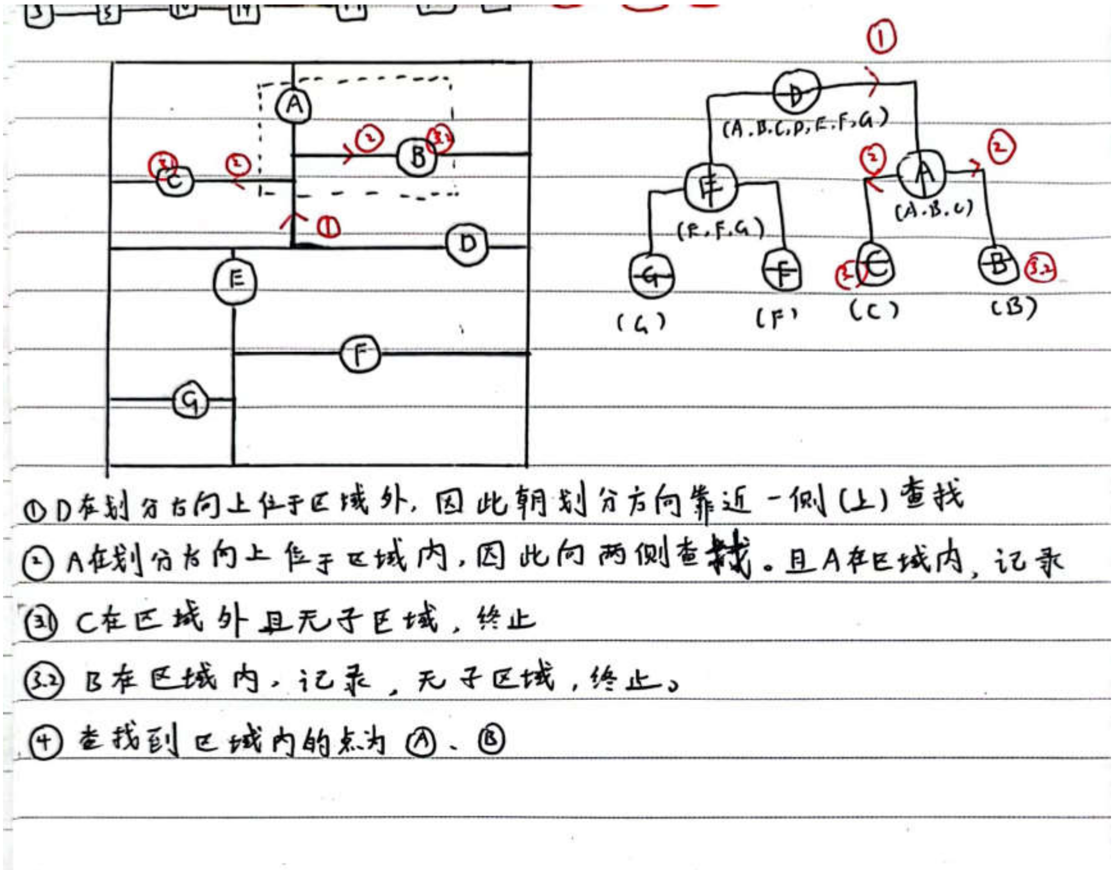
纵向的跨度更大，因此应以纵向的中位节点为根进行划分，即以 D 为根进行划分，才能使划分出的区域相对更加均匀。

划分与建树如下:



Exercise 4:

查找过程如下



Exercise5:

若多点共垂直或是共水平，可以给这些点一些随机的微小扰动对他们进行构造。如(x1，y1)，(x2，y2)，x1=x2，则令 x1 = x1 + ε1，x2=x2 + ε2，对变化后的 x1、x2 进行排序构造，但在查找取值时只取原来的 x1、x2，也就是说，变化后的 x1、x2 只是一个临时值，并不会被记录下来。