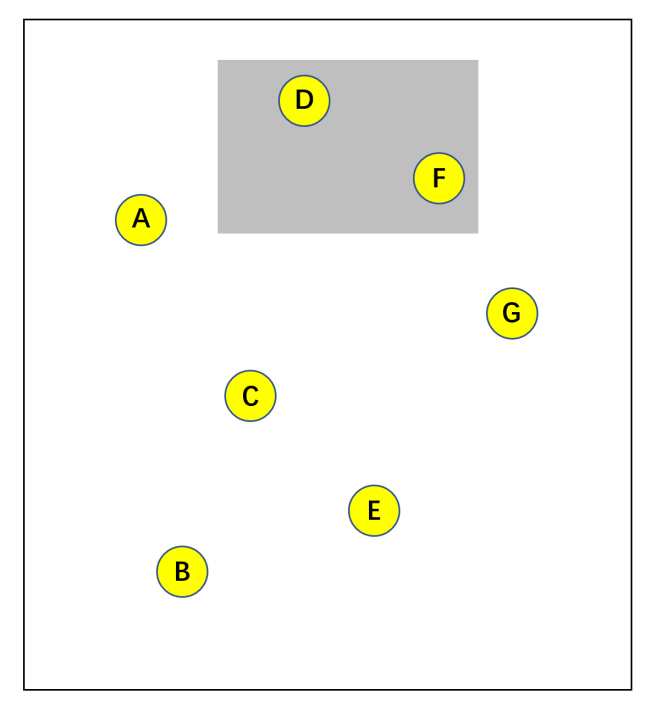
Week8 Homework

1. KD Tree。
2. 在二维空间中有一系列点。我们希望知道任意一个矩形范围内有多少个点（counting），请问我们可以采用什么简单的方法在O（logn）的时间内完成查询？它所需要的空间复杂度是多少？
3. 在一维情况下，我们有如下11个节点[5, 27, 30, 17, 10, 24, 22, 3, 14,18,20]。我们希望在O（logn+r）的时间复杂度下查找到21和27范围内的节点。请给出你的查找过程。（请画图表示）
4. 如下图所示，在二维平面上有一系列点。请画出kd-Tree在图上的平面划分以及树型图。



1. 当对上图灰色区域进行查找的时候，请画出查找过程并配用文字说明。
2. 请计算一下在二维空间上kd-tree查询的时间复杂度，请写出计算过程。并在高维上进行推广。
3. 在高维情况下kd树的效率如何？为什么？

二、 提交内容

请将答案写在后面的答题纸范围内，将作业上传 Canvas，命名使用“学号+姓名+hw8”，如“520123456789+张三+hw8.\*”。

三、 注意事项

1. 如有需要手写、画图的内容，可以拍照、截图附在答题纸题号后即可。请上传电子版。

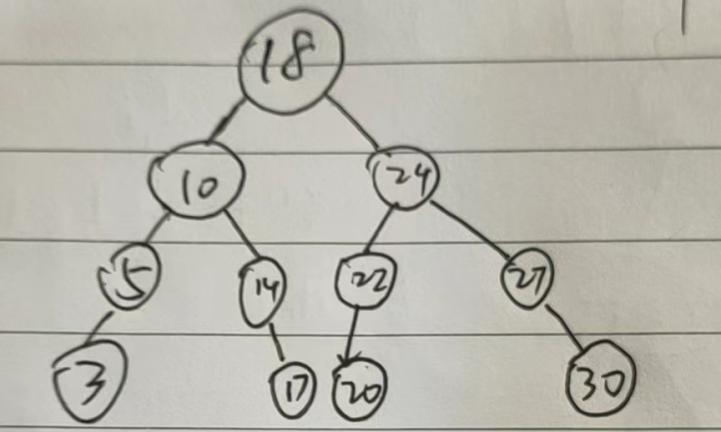
Week8 Homework答题纸

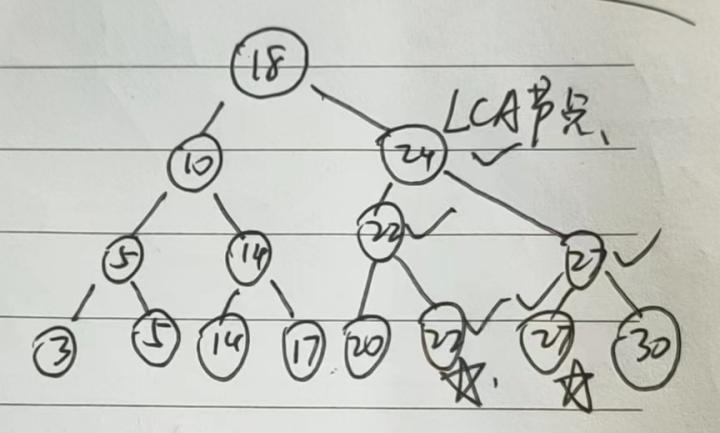
姓名： 李昱翰 学号： 520021910279

1. KD Tree

1. 按照单位长度对给定的矩形区域进行划分，之后利用容斥原理进行查询。其空间复杂度为O（n^2）.

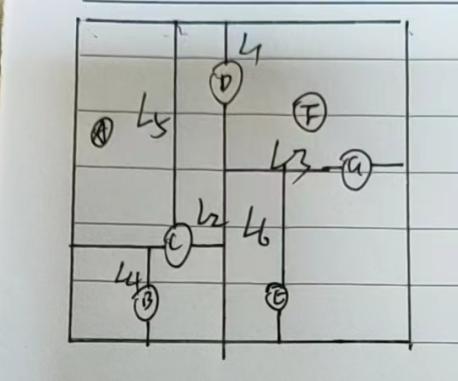
1. 首先进行建树，先要将目标数组建成一棵BST，如下：

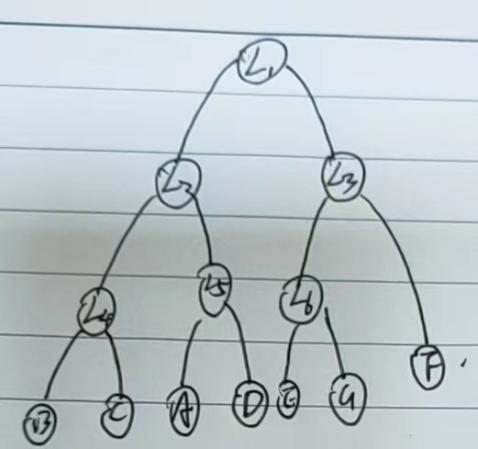




之后，首先查找21，发现没有，故将其定位在22节点所在的位置，之后查找27节点，发现可以找到，之后标记第二个目标节点(值为27)，之后找到22节点与27节点的最近公共祖先，之后重走从LCA到两个目标节点的路径，并按照查找到的顺序取输出每一个路径上的节点

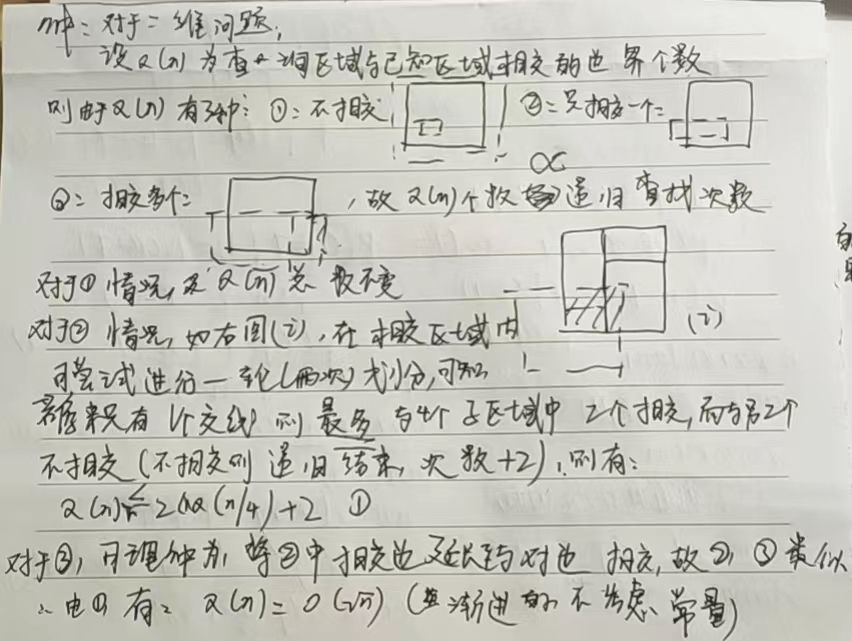
3.

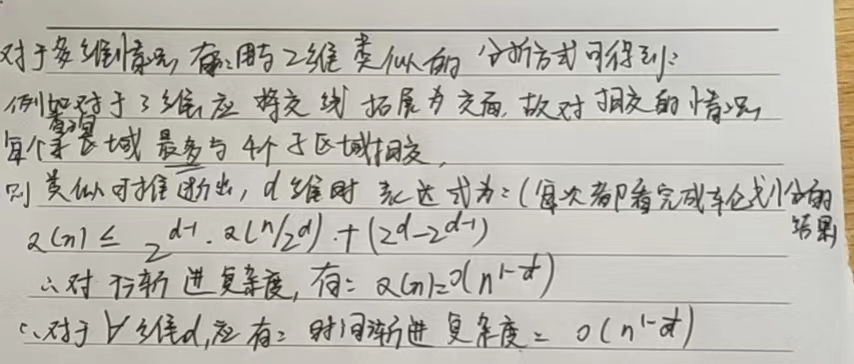




4. 根据上题中的划分方式可以知道，目标区域包括L1左侧与L1右侧两部分。故下面进行并行的查找。首先，发现L1分割线上的D点即为一个目标点，故输出D，既找到了D；之后进入L1左侧，根据查找区间可以知道其位于L1左侧，L5右侧以及L2上侧，而这一部分只有D点，故退出；之后进入右侧，发现目标区域右半部分位于L1右侧，L3上侧，而该区域只有F点，则找到新的结果F点，并输出F点，同时查找结束。

5.





6. 通过第五题的结果可以看到，随着维数的提高，KD-树的查找的时间性能也会随着提高，但是增长速度相对较慢，且无论维数如何其时间性能都优于O（n），故相对于其他的一些指数级别的算法，KD树相对而言可能更加适合于高维的查找，但是由于KD树主要作用是实现最近邻的查找，故若对于高维并且分布极其离散的数据，KD树的性能应该会明显弱于理论数值。