2.注意这道题的输入只是一个数字。

3.类似于例题最邻近点对的划分方式，大家都能想出来。但是分析算法的复杂度时，关于临界区域处理的情况，严禁一点应该证明出临界区需要考虑的点的个数上限。助教用的方法是，如果左右两边的最小三角形周长用a来表示，那么我将临界区划分成周长为a的小方格，每个方格至多有两个点。

4.除了左右子树分别的情况，merge阶段要考虑跨左右子树满足的节点对。我做出来的是一个算法。我们可以为每个节点维护一个到当前根节点的距离，递归的每一层更新一次，这样我们找左右两边点对的时候只需要两边和当前根节点距离相加。分析算法复杂度时，其实是不能保证左右两个子问题每次都均匀划分的。我们可以通过分析每个节点比较的次数，节点到当前根节点距离更新的次数来分析。

5. 每层递归都需要做一个类似归并排序即可，

大家在写算法伪代码（尤其是考试）的时候，中英文都可以。可以先写主要的思路。不要写具体的高级语言代码。也不要写的过于简略，保留主要过程，突出关键细节。