算法基础 2023 春 Lab 2

任课教师：陈雪&邵帅 due: April 14, 23:59

（1）只准讨论思路，严禁抄袭；抄袭者和被抄袭者都不计分；

（2）只能阅读 bb 上的材料和教材算法导论。严禁网上搜寻任何材料，答案或者帮助

（3）不能直接调用库函数完成实验要求设计算法的主要部分。不按照规定算法完成的，该小题判为零分

（4）请提交实验报告，在报告中描述算法实现的框架。

问题 1：平面最近点对（30 分）问题描述：实现 $lecture3$ 中描述的求平面最近点对的算法；

问题 2：cuckoo hash（30 分）问题描述：我们知道一般的哈希函数在哈希表大小没有远大于初始值长度的时候容易出现冲突，而 $cuckoohash$ 是一种可以减少冲突可能性的哈希。它由两个哈希函数构成，每次会计算初始值的两个哈希值，如果有其中一个表中位置是空的，那就把初始值放入其中空的位置；如果两个哈希值都已经存了别的值，那么就把其中一个存在其中的值“挤出去”，把要插入的值放入空出来的位置，被挤出去的旧值重新插入。如果被挤出去的旧值的两个哈希值也都被占了，那么他也会再挤出去一个旧的值，自己放入被挤出去的位置，被挤出去的值重新插入……我们会设定一个阈值，当插入一个值，把别的旧值挤出去这样的情况重复次数多于这个阈值，我们认为产生冲突，无法再插入新的值，停止算法。实现 $cuckoohash$，完成 $Lookup(key)$，$Insert(key)$，$Delete(key)$ 等操作:

1. $Lookup(key)$: 判断键值是否存在表中；
2. $Insert(key)$: 插入一个新的值；
3. $Delete(key)$: 从表中删去键值；

问题 2：红黑树（40 分）维护一个数据结构，每个元素由 ID 和 score 组成。ID 为 $[0, 10^9]$ 中的整数，成绩为 $[0, 10^9]$ 中的整数。支持一下操作（具体描述见 OJ）

1. $Insert(ID, Key)$：插入一个新元素。
2. $Delete(ID)$：删除一个元素。
3. $Lookup(ID)$：返回该元素的 score。

4. $Select(k)$：返回 score 第 k 大的元素 ID。

5. $Minimum(k)$：返回在 $[k, +\infty)$ 区间内，score 最小的 ID（如果有多个最小 score，返回 ID 最小的）。

6. $Count(L, R)$：统计 score 在 $[L, R]$ 区间内，有多少个元素。