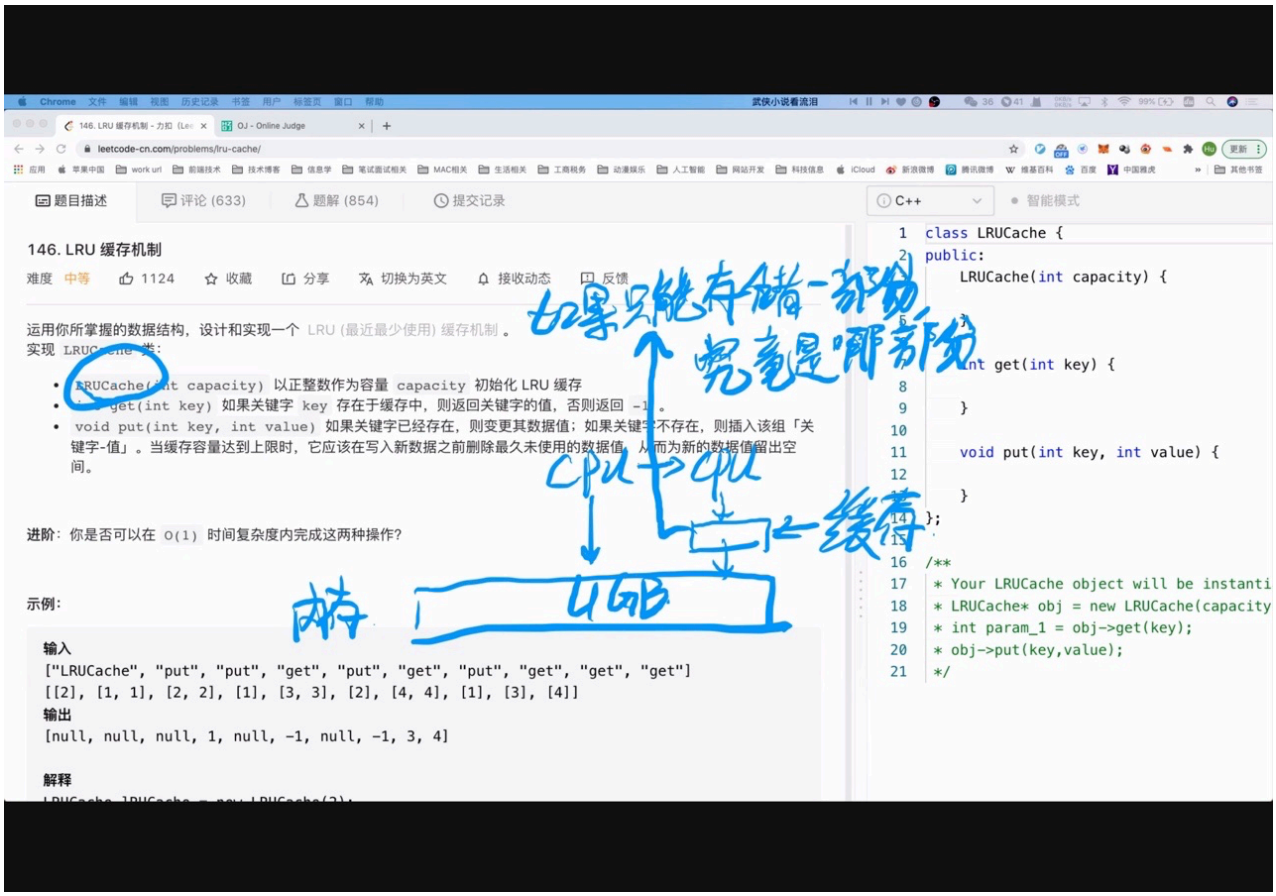
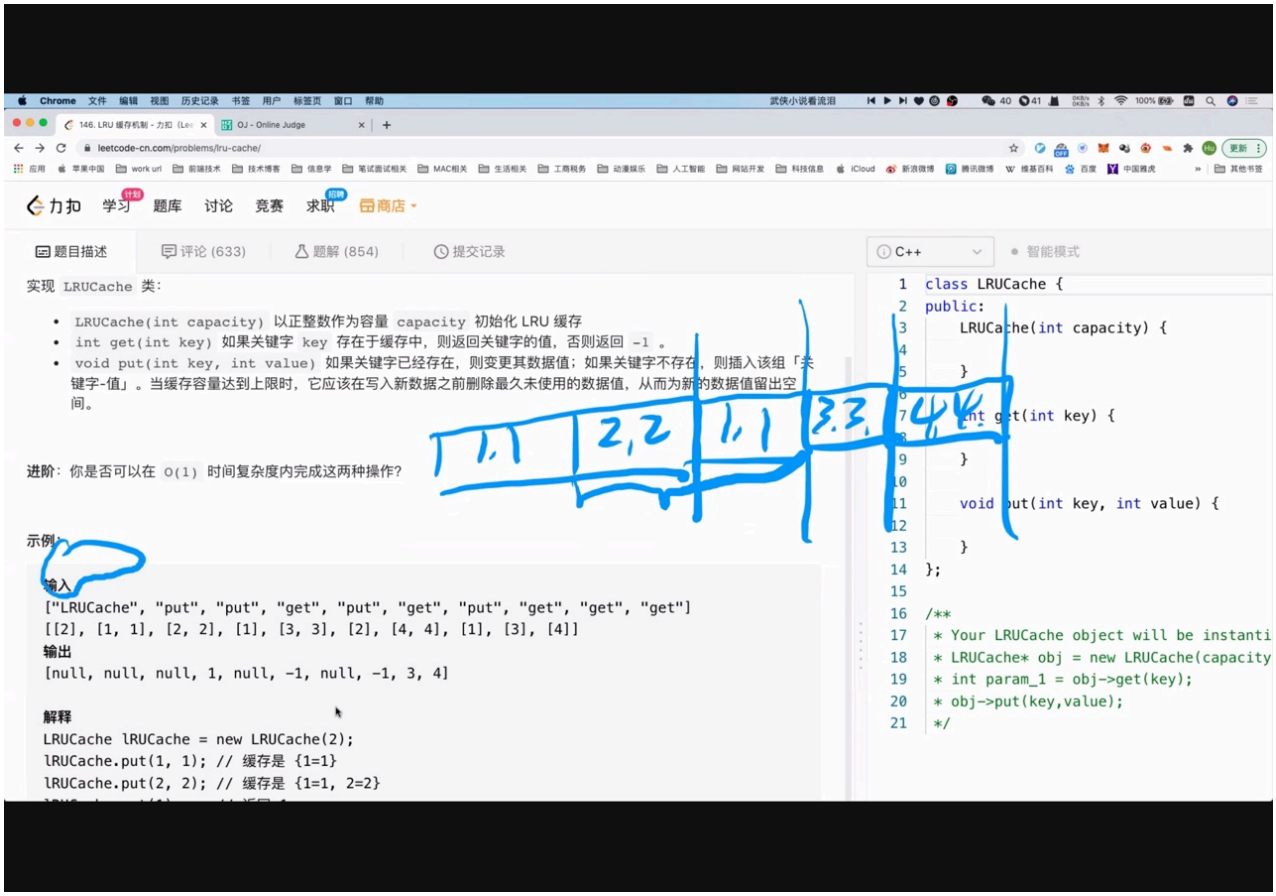
习题答疑课（三）

一、【中等】leetcode-146-LRU缓存机制

1. 哈希表 + 链表解决 $O(1)$ 读取，以及 $O(1)$ 修改缓存数据节点位置的操作
2. **弯路1：**一开始想到了哈希表，但忽略了利用链表实现 $O(1)$ 修改
3. **弯路2：**由于头尾指针会发生变化，忽略了虚拟节点的处理技巧





二、【困难】Leetcode-460-LFU缓存

1. 编码较复杂，基于十字链表实现 LFU 缓存机制
2. 将出现次数相同的节点，存储在同一个 LRUCache 中
3. 将所有非空的 LRUCache ，按照代表的次数，链接成一个大链表
4. 删除节点的操作，简化成了：删除第一个 LRUCache 中的头结点
5. **操作1：**删除一个节点
6. **操作2：**新增一个节点
7. **操作3：**将一个现有节点，移动到下一个 LRUCache 中