

线性表

1. 代码 单链表插入 删除 清除 逆置

栈

1. 应用 后缀表达式 中缀转后缀 手算中缀转后缀

队列

1. 循环队列 判断队列满 $\text{front} == (\text{rear} + 1) \% \text{MaxSize}$ 与空 $\text{front} == \text{rear}$
2. 代码 顺序循环队列 DeQueue EnQueue
3. 代码 链接循环队列 DeQueue EnQueue

树

1. 非空二叉树中 叶节点 n_0 与度为 2 的节点 n_2 的关系 (用来求总结点数或一个儿子的结点个数)
2. 代码 应用 二叉树的前序 中序 后序 层次遍历 递归实现
3. 代码 求高度 删除
4. 应用 表达式树
5. 应用 哈夫曼树 ①建树的过程 ②哈夫曼用表存储要会写
6. 了解 森林部分 表示法是怎么表示的

优先级队列

1. 了解 二叉堆的概念 (完全二叉树)
2. 应用 代码 二叉堆存储 (顺序 层次遍历) 入队 (插在最后 向上过滤) 出队 (根节点出队 最后一个节点塞在根节点上 向下过滤 替换左边的) 给一堆数据要学会建堆或判断是不是堆

集合与静态查找

1. 内部查找与外部查找区别
2. 顺序查找 二分查找 插值查找 分块查找 (块与块有序 (索引查找))

动态查找表

1. 了解 二叉查找树概念 左子树小于根 右子树大于根 递归
2. 代码 查找结点 递归
3. 应用 代码 (代码看下吧) 插入结点 (小就插左子树 大插右子树 递归) 删除结点 (一个儿子 儿子取代位置 两个儿子 左子树最大或右子树最小作为替身 儿子代替替身的位置)
4. 了解 平衡二叉树 AVL 树的概念 在查找树的基础上 依旧是左小右大
平衡度: 左子树高-右子树高 最少结点 $F(n)$ 斐波拉契数列 最多结点 $(2^n - 1)$ 空-1 一个节点 0 性能 $\log N$
5. 代码 查找
6. 应用 插入结点 (LL RR LR RL 重要!! 建议自己默写过程!! 插入可以拿来建树 考核重点!)
删除结点 (向上回溯 不平衡用 LL RR 旋转法调整高度)

7. **应用** 闭散列表（除留余数法+线性探测（查找值与避免碰撞）） 概念要看！开散列表概念要看

排序

1. 稳定与不稳定 内排序外排序区别
2. 看 9 10 整理的 PDF 表格 主要看稳定 与时间复杂度
3. 时间复杂度与初始次序无关 堆与归并 都是 $N\log N$
4. 不稳定排序：希尔 直接选择 堆 快速
5. 要记住每个排序的具体表现
 - ① 插入 直插 N^2 **稳定**
希尔 **不稳定**
 - ② 选择 直选 N^2 **不稳定**
堆排序 $N\log N$ **不稳定**
 - ③ 交换 冒泡 N^2 **稳定**
代码 快速 $N\log N$ （快速在已经基本有序时最慢 N^2 ）**不稳定**
 - ④ 归并排序 $N\log N$ 合并有序表 俩放在一起 找小的往里放 **稳定**
 - ⑤ 基数排序 口袋排序法 **稳定**

外部查找与排序

1. **了解** B 树概念 m 阶 B 树 根 2 到 m 儿子 其余 $m/2$ 到 m 个儿子 关键字是儿子个数-1 支持随机查找 所有关键字在树上
2. **应用** B 树插入（分裂） B 树删除（合并） B 树总的数据量计算
3. **了解** B+ 树 叶子在同一层 有 $L/2$ 到 L 的数据 支持随机查找与顺序访问 L 与 M 的选择
4. **应用** B+ 树插入与删除（分裂与合并）
5. **应用** 外排序（归并排序） 置换选择的过程（比输出大的加入优先级队列） 两路归并的过程

图

1. **了解** 邻接矩阵（有向无向 N^2 ） 完全有向 $n(n-1)$ 完全无向 $n(n-1)/2$ 最少边 $n-1$ 总的度是边的两倍
2. **代码** 邻接表的创建 $O(n+e)$
3. **代码 应用** DFS 前序遍历 对所有顶点与边访问 BFS 层次遍历每个顶点入队一次 都是 $O(n+e)$
4. **应用** ①欧拉回路 ②有向图连通性（DFS）正着遍历反向标号 将图逆向从大到小再来一次 最终得出连通分量
 - ③拓扑排序 入度为 0 就输出 很简单
 - ④关键路径 最早发生时间 **取最大的** 最迟发生时间 用终点的最早发生时间反推 **取最小** 活动 $\langle A, B \rangle$ 最早时间是 A 的最早时间 最迟时间 V 的最迟- AB 的边值 俩一样就是关键路径