

2023-2024 Fall

总复习

# 期末考试题型及占比

- 一. 单项选择题, 共15小题 (30%)
- 二. 应用题, 共5小题 (50%)
- 三. 编程&算法设计题, 共2小题 (20%)

# Chapter1-3

• ADT = (D, R, O)

- 数据的逻辑结构: 线性(1-1), 非线性(1-n, n-n)
- 数据的物理结构: 顺序存储, 链式存储
- *Growth rate* :  $\log_2 n$ ,  $n$ ,  $n \log_2 n$ ,  $n^2$ ,  $n^3$ , ...,  $n^k$ , ...,  $2^n$ ,  $n!$
- 算法的时间代价函数 $f(n)$ : Best, Worst, Average Cases
- 时间复杂度及其渐进分析:  $O, \Omega, \Theta$  notation

# Chapter4 List

- List ADT: (D, R, O), 会调用 各种公共操作函数
- Alist : 表满/表空问题, 各种操作的功能, 算法及复杂度
- Llist
  - 当前指针指向的是当前结点的前一个结点, 头结点不放数据
  - 表空问题
  - 各种操作的功能, 算法及复杂度
- Free Link : 为了加快new/delete操作, 占用空间多
- Double Llist: 为了简化prev操作, 增加了空间需求
- Stack
  - LIFO
  - AStack & Lstack 各种操作 (push, pop) 的实现步骤
- Queue
  - FIFO
  - CAQueue & Lqueue 各种操作 (EnQueue, DeQueue) 的实现步骤

编程题: 能调用相关线性结构 (如List, Stack, Queue) 公共函数编写要求功能函数。

# Chapter5 BT

- 术语: Internal node, leaf, degree, height, depth, BT, FT, CBT
- BT的遍历: 深度优先(前序, 中序, 后序)遍历, 广度优先遍历
- BST的定义, 存储结构, 构造, 插入, 删除, 中序遍历
- Heap(priority queue)的定义, 存储结构, 构造, 插入, 删除
- Huffman Trees and Huffman coding: 定义, 构建, 编码

应用题: BT的深度优先遍历, 能由两种遍历结果重构出BT; heap的构造/插入/删除, BST的构建/插入/删除, Huffman树的构造/编码,

编程题: 能编写合适的(递归)函数(函数中可调用二叉结点类的公用函数)实现一些具体功能。如寻找BST中的第K大(小)结点值, 判断两棵BT树(的形状)是否一样, 统计BT树的叶子节点个数等。

# Chapter6 NBT

## ● 通用树的遍历：

- 前根，后根遍历，与对应BT的遍历的关系

## ● 通用树的表示：

- 双亲指针表示-----应用：Equivalence classes（FIND，UNION 操作）

- 左孩子/右兄弟链接表示法

## ● 通用树→二叉树的转换，森林→二叉树的转换

# Chapter7 internal sorting

- stable
- Insertion, Bubble, Selection sorting
- Shell sorting
- Quick sorting
- Merge sorting,
- Heap sorting
- Radix Sorting

● 概念：各种排序算法的思路(每趟后的中间结果，数据初始顺序对各排序算法时间的影响)，计算复杂度(best, worse and average case)，空间要求，stable or not

应用题：

给定一序列，能给出某种排序方法的整个排序过程（即每趟后的中间结果），或者其关键操作（如一次划分）的详细过程及结果；

算法编程题：

insert/selection/bubble/Quick/Heap sorting的实现

# Chapter 8 file processing and external sort

- track, sector, Cluster, Locality of Reference
- Golden Rule of File Processing, logical/physical file
- Buffer and Buffer pools
- External sorting
  - 外部排序的一般步骤：获得初始RUNs + 归并排序
  - Simple External Mergesort: 初始runs长度为1, 2路归并
  - Replacement Selection 的目的, 思路: 获得尽可能长的初始runs
  - 最优思路: RS + 多路归并

平均长度:  $2BM$

趟数  $r = \log_B N / (2BM)$ , I/O 次数:  $(r+1) * 2 * N / M$



# Chapter9 Search

- Binary search, sequential search
- Hashing:
  - hash function  $h(K)$ , Collisions
- Collision Resolution
  - Open Hashing: Separate chaining (linked list)
  - Closed Hashing: probe sequence( $d_i$ ), Double Hashing:  $d_i * h_2(K)$
- HT Insert, Search in HT, SL, ASL

应用题：给定哈希函数及冲突处理方法，

1) 能将一组数放入HT

2) 给定HT后，会计算一个(组)待查询数的SL (ASL)

# Chapter10 indexing

- 2-3 Tree的定义, search, insert

- B-tree的定义

- 所有结点都存 key/pointer pairs

给定高度, 能确定B树/B<sup>+</sup>树中可存key/pointer 的最大/最小个数

- B<sup>+</sup> tree 的定义, 特点 (与B-tree区别)

- 需要两个参数:

- m 决定内部节点的子树个数上限,
- n决定叶节点中可存放key/record 个数上限

- leaf 存 key/pointer, 内部结点仅存key

- 结点半满 (除root)

- B<sup>+</sup> tree的 search, insert, delete

应用题:

2-3树的构造, 插入;

B<sup>+</sup>树的构造, 插入, 删除

# Chapter 11 Graph

- Adjacency Matrix, Adjacency List
- Graph Traversals: DFS, BFS
- Topological Sort Algorithm of DAG — BFS based, DFS-based
- Shortest Paths problem
  - Single-Source Shortest Paths, Dijkstra's Algorithm
- Minimal Cost Spanning Trees(MST) problem:
  - Prim's algorithm
  - Kruskal's algorithm

应用题：给定图，1) 能画出邻接矩阵/邻接List； 2)若是DAG，能给出基于BFS的拓扑排序过程及结果； 3) 能用要求算法获得MST； 4)会用Dijkstra's Algorithm求出单源最短路径

编程&算法描述题： Adjacent matrix / Llist存储结构下图的基本操作；  
Dijkstra算法

# 一些注意事项

- 注意给定信息，不要想当然，比如min-heap/max-heap, B树/B+树, closed Hashing的 $d_i$
- 仔细看题，要求答的东西逐条解答，不要漏，也不要混在一起。
- 编程&算法设计描述尽量去答，若不会写原码，可写伪代码。



**Q&A**

祝大家期末考试顺利！