## 考试题型

1. **选择题20题\*1**
2. **判断题10题\*1**
3. **应用题48分，6题**
4. **算法题22分，2题**

## 期中后占80

## 有部分英文题目

## 复习素材

* 1. **PPT、PPT上的课堂练习题**
  2. **所有做过的CG、雨课堂的作业、练习题**
  3. **本复习大纲**
  4. **书，包括书后习题**

## 第1章 概述

**1、四类数据结构，集合、线性表、树、图。线性结构、非线性结构，ADT、逻辑结构与存储结构（物理结构）。**

**2、算法的时间效率和空间效率。**

**3、计算某一条语句执行次数或某段程序段的时间、空间复杂度。**

**4、时间复杂度计算，与后面的一些算法结合，比如排序算法、查找算法的时间、空间效率及比较。**

## 第2章 线性表

**1、线性表的概念和基本操作，存储结构**

**2、顺序线性表下元素的存储，类定义和算法，基本算法的时间复杂度，插入删除元素移动次数计算。**

**3、单链表下元素的存储，类定义和算法，基本算法的时间复杂度，插入和删除时需定位操作位置的前驱结点。**

**4、单循环链表、双向链表下的算法。**

**5、存储结构的对比，优缺点，具体情况下如何选择存储结构？**

**6、顺序表、链表下的算法设计，一般要求直接访问表元素，不调用其他方法。**

## 第3章 栈和队列

**1、栈的ADT（抽象数据类型）定义。**

**2、栈的操作特点（LIFO，后进先出）、overflow, underflow, push, pop后栈的状态、入栈出栈过程及对应序列**

**3、栈的类定义及顺序实现（包括各个方法的具体实现）**

**4、链栈类定义、具体实现（包括各个方法的具体实现）**

**5、栈的应用：数据逆置算法、括号匹配的判断、后缀表达式求值算法**

**6、根据中缀写出后缀表达式, 前缀、中缀和后缀的含义**

**中缀表示 (A+B)\*D-E/(F+A\*D)+C，转换为前缀、后缀**

**前缀表示 + - \* + A B D / E + F \* A D C**

**后缀表示 A B + D \* E F A D \* + / - C +**

**7、掌握队列的抽象数据类型定义。**

**8、队列的特点、性质（FIFO，先进先出表,队尾入队、队头出队后不改变原序列），队列的应用（缓冲、排队）**

**9、队列的类定义及顺序实现，顺序队列产生的假溢出问题！如何解决？**

**10、利用循环队列产生的问题？有哪些解决方案？**

**10、 循环队列实现算法。（包括各个方法的具体实现）**

**11、链队列定义、具体实现（包括各个方法的具体实现）**

**12、递归概念，递归函数由终止条件及递归部分组成；栈与函数调用和递归，用栈存放每一层函数的“调用记录”（参数、局部变量、返回地址等）**

**13、递归算法实现**

**14、栈和队列在后续内容中的应用算法**

## 第4章 串和数组

**1、字符串相等判别方法，KMP算法的特点（i不回溯）**

**2、数组的操作： ，随机存取？**

**3、各种二维数组的存储（普通二维数组，特殊的矩阵），二维表格映射到一维存储时的下标函数。k=i\*n+j**

## 第5章 树与二叉树

**1、二叉树定义、相关概念、术语**

**满二叉树：高度h，2h-1个结点的二叉树**

**完全二叉树：满二叉树基础上从右下方开始连续地删除若干个结点。高度h，结点个数的范围：2h-1 ---2h-1个**

**2、二叉树性质**

**（1）i层上最多有2i-1个结点**

**（2）二叉树高度为h，至多2h-1个结点，至少为h个结点(设空二叉树高度为0，仅含一个结点的二叉树高度为1)**

***（3）n0 = n2+1***

***b=n-1***

***b=n1+2n2+3n3***

***多叉树中的类似性质***

**（4）具有 *n* 个结点的完全二叉树的高度为 *⎣ log2n⎦ +1***

**（5）若对含 *n* 个结点的完全二叉树从上到下且从左至右进行 *1* 至 *n* 的编号，则对完全二叉树中任意一个编号为 *i* 的结点：**

**(1) 若 *i=1*，则该结点是二叉树的根，无双亲，**

**否则，编号为 *⎣i/2⎦* 的结点为其双亲结点；  
(2) 若 *2i>n*，则该结点无左孩子，  
 否则，编号为 2i 的结点为其左孩子结点；  
(3) 若 *2i+1>n*，则该结点无右孩子结点，  
 否则，编号为2i+1 的结点为其右孩子结点。**

**3、求解二叉树的前序、中序、后序、层次遍历，根据2个序列构造二叉树。**

**4、二叉树的顺序实现**

**5、二叉树的链式实现及递归算法、层次遍历等算法的实现**

**5、树的带权路径长度WPL计算、哈夫曼树及其哈夫曼编码**

**6、树，森林的概念**

**森林、树与二叉树之间的相互转换，结点之间关系的转换**

**森林、树与二叉树遍历序列之间的关系**

**7、二叉树（包括二叉查找树）的算法、遍历的应用算法**

## 第6章 优先队列

## 第7-8章 图

**1、概念，图，有向图，无向图，连通图（connected），完全图，连通图的最少最多边数，强连通图的最少最多边数，带权图（网）network，顶点的入度，顶点的出度，顶点度的最大值，所有顶点的度数之和是边数的2倍。**

**2、图的两种存储方案：邻接矩阵、邻接表表示，空间复杂度。无向图/有向图的邻接矩阵/邻接表的特点，根据给定的图（网）画出邻接矩阵或邻接表或反之。在这两种结构下求解图中顶点的出度、入度、边数等的方法。**

**3、图的深度优先遍历dfs（树的先序遍历推广）和广度优先遍历bfs（层次遍历推广，用队列辅助完成算法）遍历序列、算法及灵活运用。给定存储结构下的dfs/bfs遍历序列，两种存储结构下的算法时间复杂度。深度优先搜索生成树和广度优先搜索生成树。遍历的应用，判断图中是否存在回路？ 有向图，拓扑排序，深度遍历**

**无向图，深度遍历，广度遍历**

**4、生成树T（含有无向连通图G中全部的n个顶点，n-1条边，连通，无环的子图），**

**判断：**

**n个顶点，少于n-1条边的图一定不连通。**

**n个顶点，多于n-1条边的图一定连通。**

**最小生成树的概念，是否唯一？求解最小生成树的实际意义，prim算法和kruscal算法具体求解过程、时间复杂度。分别适用场合。**

**5、最短路径？Dijkstra算法求解具体过程方法，floyd算法思想，时间复杂度，权值必须为正。**

**6、AOV网概念，有向无环图DAG图概念，拓扑排序的实际意义，bfs/dfs拓扑序列手工求解过程。有向图是否有环的判别？**

**7、AOE网概念、关键路径、关键活动求解方法**

## 第11章 查找

1. **各查找(顺序查找、二分查找、二叉排序树、散列查找)算法的递归算法和非递归算法实现、性能分析**
2. **画二分查找判定树（2种二分查找），判定树的形状特点，利用比较树计算平均比较次数ASL，成功和失败**
3. **二叉查找树的定义、判别、特点：中序序列为递增序列**
4. **二叉查找树下的查找算法实现、效率，不同形状的二叉查找树下的查找效率不同，越矮的二叉树越有利于查找。**
5. **二叉查找树下结点插入的方法**
6. **二叉查找树的建立，分析平均比较次数ASL，成功和失败**
7. **二叉查找树下结点删除的方法**
8. **AVL树的概念、判定方法，平衡因子，最坏情况高度为h的AVL树的结点数。**
9. **以关键字比较为基础的查找算法的最好性能 O(log2n)**
10. **哈希函数（计算简单，均匀散列），解决冲突的几种方法：开放定址法（线性探测、二次探测等），链表法。如何构造哈希表，（成功、失败）平均查找长度的计算。哈希查找理想情况下的时间效率（O（1）），实际情况的效率与哪些因素有关（~~哈希函数~~、冲突解决策略、装载因子），装载因子的概念。哈希查找需不需要关键字比较？？**
11. **查找算法效率分析和比较**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **查找方法** | **存储结构要求** | **时间效率** | **优缺点** | **查找原理** |
| **顺序查找** | **顺序结构或链式结构的线性表** | **O(n)** | **当n较大时，查找比较耗时** | **从线性表的头部对每个记录依次进行比较，直至找到一个与目标关键字相同的记录，或直到表尾都无法找到，则为失败的查找。** |
| **二分查找** | **顺序存储结构、元素已有序** | **O(logn)** | **需要事先保持记录有序** | **按照逐渐缩小被查区间的方法进行查找。** |
| **二叉查找树查找** | **二叉链表结构，记录按照二叉查找树的要求进行排序** | **在理想状态，O(logn)** | **二叉查找树的形态影响其查找性能** | **在二叉查找树下进行查找，如与根结点值相同，则查找成功；若二叉树为空，则查找失败；否则按照值的大小到左子树或右子树上进行查找。** |
| **AVL** | **平衡的二叉查找树，二叉链表结构** | **O(logn)** |  |  |
| **哈希查找** | **哈希表** | **理想状态为O（1），实际与装载因子、解决冲突等因素有关。** | **需要解决哈希函数选择、冲突解决、装载因子的选择等问题** | **在哈希表下，根据哈希函数的计算和冲突的解决方法，进行关键字对应记录的存储和查找。** |

## 第10章 内排序

**1、各种排序算法（直接插入、希尔排序、起泡排序、快速排序、选择排序、堆排序、归并排序/基数排序）的算法原理，基于不同存储结构的执行过程、最好、最坏、平均情况下的时间性能，空间性能，稳定性，根据实际情况如何选取合适的排序算法。**

**2、以关键字比较为基础的排序算法的最好性能O(nlogn)**

**3、堆的概念、性质、如何判别、堆的应用、堆的建立（递增、递减排序分别对应大根堆、小根堆），堆中插入元素，删除元素，堆排序的过程（给出前几个最大值或最小值）和性能。**

**4、快速排序的一趟划分，可以单、双方向划分**

**5、利用排序的思想对存储于顺序表（数组）或链表中的数据进行处理的算法。（查找、划分、按要求重排等）**