

《数据结构》考试大纲

考核目标

1. 理解数据结构的基本概念，掌握数据结构的基础理论；
2. 熟悉并掌握线性表、栈、队列、串、数组、广义表等的逻辑结构、存储结构以及对数据的基本运算；
3. 熟悉并掌握抽象数据类型的表示、实现、运用；
4. 理解算法的基本概念、特点以及性能分析；
5. 掌握查找和排序的基本概念、思想和算法实现；

考核内容

一、绪论

1. 考核知识点

数据结构；抽象数据类型；算法；算法的时间复杂度；算法的空间复杂度。

2. 考核要求

- (1)理解数据结构的基本概念和术语；
- (2)掌握抽象数据类型的表示与实现；
- (3)掌握算法的基本概念和算法的性能分析方法。

重点掌握抽象数据类型的表示；算法的时间复杂性能分析的方法。

二、线性表

1. 考核知识点

线性表；顺序表；链表；顺序存储结构；链式存储结构。

2. 考核要求

- (1)理解线性表的定义和逻辑结构特性；
- (2)掌握线性表的顺序存储方法和基本操作算法实现；
- (3)掌握线性表的链式存储方法和基本操作算法实现；

重点掌握线性表的顺序存储结构、链式存储结构和顺序表和多种链表的算法实现。

三、栈和队列

1. 考核知识点

栈；递归；链队列；循环队列。

2. 考核要求

- (1)熟练掌握栈的类型定义、表示和基本操作的实现；
- (2)运用栈的特性设计算法；
- (3)掌握递归算法的设计思路和设计方法；

(4)熟练掌握队列的类型定义、表示和基本操作的实现
重点掌握栈和队列的特性、基本算法的实现以及应用。

四、串

1. 考核知识点

串的定义、基本运算算法，串的模式匹配定义和算法。

2. 考核要求

(1)掌握串类型的定义及其表示方法；

(2)掌握串基本算法的实现方法；

(3)了解串的应用算法。

重点掌握串的定义、串的基本算法的实现。

五、数组和广义表

1. 考核知识点

数组；稀疏矩阵；广义表的定义和基本运算

2. 考核要求

(1)了解数组的定义和数组的顺序表示方法；

(2)数组元素顺序存储的地址计算；

(3)掌握特殊矩阵和稀疏矩阵的压缩存储方法；

(4)掌握广义表的定义和基本运算；

重点掌握数组元素的地址计算方法；

六、树和二叉树

1. 考核知识点

二叉树的存储结构及其遍历的方法；二叉树的线索化；哈夫曼树的构造方法及其编码的生成。

2. 考核要求

(1)理解树和二叉树的定义、术语和基本逻辑结构特性；

(2)理解二叉树的基本性质；

(3)理解二叉树存储结构；

(4)理解二叉树的遍历算法思想，掌握递归和非递归遍历算法实现；

(5)掌握线索二叉树的基本概念和相应算法；

(6)了解树和森林的存储方法及与二叉树的之间的转换方法；

重点掌握二叉树的特性；二叉树的遍历；二叉树的线索化；哈夫曼树的构造算法。

七、图

1. 考核知识点

图的逻辑结构；邻接表；深度优先遍历；广度优先遍历；最小生成树、拓扑排序、关键路径、最短路径。

2. 考核要求

(1)理解并掌握图的基本概念、术语和基本逻辑结构特征；

(2)理解并掌握图的存储结构；

(3)掌握图的深度优先和广度优先遍历算法；

(4)了解最小生成树、拓扑排序、关键路径、最短路径的应用。

重点掌握图的逻辑结构；图的存储方法；图的深度优先、广度优先遍历算法。

八、查找

1. 考核知识点

顺序查找；折半查找；分块查找；二叉排序树；平衡二叉树；哈希表。

2. 考核要求

- (1)理解静态查找表、动态查找表和哈希查找的基本概念；
- (2)掌握静态查找表的各种查找方法如：顺序查找、折半查找、分块查找；
- (3)掌握动态查找表的各种查找方法如二叉排序树与平衡二叉树，B 树等；
- (4)掌握哈希表的概念和查找方法和哈希函数的构造方法、解决冲突的基本方法；重点掌握折半查找、二叉排序树和哈希表的查找算法的实现。

九、排序

1. 考核知识点

直接插入排序；希尔排序；冒泡排序；快速排序；堆排序；归并排序；基数排序。

2. 考核要求

- (1)理解排序的基本概念；
- (2)掌握基于插入思想的排序算法如：直接插入排序、希尔排序；
- (3)掌握基于交换思想的排序算法如：冒泡排序、快速排序；
- (4)掌握基于选择思想的排序算法如：简单选择排序、堆排序；
- (5)掌握其它排序算法如：归并排序、基数排序；重点掌握插入排序、快速排序、堆排序、合并排序、基数排序等算法的设计思想。

考试方法和考试时间

1. **考试方法**：闭卷、笔试

2. **记分方式**：百分制，满分为 100 分

3. **考试时间**：120 分钟

4. 命题的指导思想和原则

命题的总的指导思想是：全面考查学生对本课程的基本原理、基本概念和主要知识点学习、理解和掌握的情况。命题的原则是：题目数量多、份量小，范围广，最基本的知识一般要占 60%左右，中等难度的题目要占 20%左右，较难的题目要占 20%左右。

5. 题目类型

- (1) 选择题
- (2) 填空题
- (3) 简答题
- (4) 判断题
- (5) 程序阅读或填空题
- (6) 设计题