



湖南省普通高等学校 2024 年“专升本”招生考试

《数据结构（C 语言版）》科目考试要求

I. 考试内容与要求

“数据结构”部分

（一）绪论

1、考核知识点

- （1）合适的数据结构在解决实际问题中的关键性；以及学习《数据结构》的意义。
- （2）数据、数据元素、数据项、数据结构等基本概念。
- （3）数据结构的四种逻辑结构和两种存储结构表示方法。
- （4）抽象数据类型的表示和实现。
- （5）算法的五个特点。
- （6）算法、算法的时间复杂度和空间复杂度、最坏的和平均的时间复杂度等概念。
- （7）算法描述和算法分析的方法，对于一般算法能分析出时间复杂度。

2、考核要求

（1）识记

- 1) 数据结构的基本概念和术语。
- 2) 合适的数据结构在解决实际问题中的关键性，以及学习《数据结构》的意义。
- 3) 数据结构的四种逻辑结构和两种存储结构表示方法。

（2）理解

算法的描述和分析：算法的时间复杂度和空间复杂度、最坏的和平均的时间复杂度。

（二）线性表

1、考核知识点

- （1）线性表的类型定义。
- （2）顺序表的含义及特点，顺序表上的插入、删除操作及其平均时间性能分析。





- (3) 链式表示和实现，单链表、双链表、循环链表链接方式上的区别。
- (4) 单链表上实现的建表、查找、插入和删除等基本算法及其时间复杂度。
- (5) 循环链表上尾指针取代头指针的作用。
- (6) 单循环链表上的算法与单链表上相应算法的异同点。
- (7) 双向链表的定义和相关算法。
- (8) 顺序表和链表的比较，以及如何选择其一作为其存储结构才能取得较优的时空性能。

2、考核要求

(1) 识记

- 1) 线性表的逻辑结构特征；
- 2) 线性表上定义的基本运算，并利用基本运算构造出较复杂的运算。

(2) 理解

- 1) 顺序表和链表的比较，各自的优缺点。
- 2) 针对线性表上所需要执行的主要操作，知道选择顺序表还是链表作为其存储结构才能取得较优的时空性能。

(3) 综合应用

- 1) 顺序表的含义及特点，顺序表上的插入、删除操作及其平均时间性能分析。
- 2) 单链表、双链表、循环链表链接方式上的区别；
- 3) 单链表上实现的建表、查找、插入和删除等基本算法及其时间复杂度。
- 4) 循环链表中尾指针取代头指针的作用，
- 5) 单循环链表上的算法与单链表上相应算法的异同点。
- 6) 双链表的定义和相关算法。

(三) 栈和队列

1、考核知识点

- (1) 栈的抽象数据类型的定义
- (2) 栈的表示和实现
- (3) 栈的简单应用
- (4) 抽象数据类型队列的定义
- (5) 队列的链式表示和实现





(6) 队列的顺序表示和实现

2、考核要求

(1) 理解

栈和队列的特点，栈和队列各自的使用情况。

(2) 综合应用

- 1) 栈的逻辑结构特点，栈与线性表的异同。
- 2) 顺序栈和链栈上实现进栈、退栈等基本算法。
- 3) 利用栈解决简单的实际问题。
- 4) 队列逻辑结构特点，队列与线性表的异同。
- 5) 顺序队列（主要是循环队列）和链队列上实现的入队、出队等基本算法。
- 6) 顺序队列的“假溢出”现象及其采用循环队列进行解决的方法。

(四) 串

1、考核知识点

- (1) 串的定义、空串、空格串、子串、主串、串相等。
- (2) 串的基本操作。
- (3) 串的顺序存储结构及在顺序存储结构下基本操作的实现。
- (4) 串的堆分配存储表示及其在堆分配存储结构下基本操作的实现。
- (5) 串的链式存储表示

2、考核要求

(1) 理解

串的有关概念及其基本运算。

(2) 简单应用

- 1) 串的三种存储表示。
- 2) 使用串解决与串相关的简单的应用问题。

(五) 数组和广义表

1、考核知识点

- (1) 数组的顺序存储结构。
- (2) 二维数组的按行存储及按列存储和计算数组元素的地址计算公式。
- (3) 矩阵的压缩存储、特殊矩阵的表示。





2、考核要求

(1) 理解

- 1) 多维数组的逻辑结构特征。
- 2) 多维数组的顺序存储结构及其地址计算方式。
- 3) 特殊矩阵和稀疏矩阵的概念。
- 4) 疏矩阵的压缩存储方式——三元组表。

(六) 树和二叉树

1、考核知识点

- (1) 树的定义和术语。
- (2) 二叉树(完全二叉树、满二叉树)的定义和性质(结论)、二叉树的存储结构——顺序表示法和链表表示法。
- (3) 二叉树的三种遍历方法及相应的递归算法。
- (4) 树的存储表示法——孩子表示法、双亲表示法、孩子兄弟表示法。
- (5) 树和森林及二叉树的转换方法。
- (6) 树的路径长度、树的带权路径长度、赫夫曼树(最优二叉树)的构造方法。
- (7) 赫夫曼编码方法。

2、考核要求

(1) 理解

- 1) 树的逻辑结构特征。
- 2) 树的不同表示方法。
- 3) 树的常用术语及含义。
- 4) 树和森林与二叉树之间的转换方法。
- 5) 树的各种存储结构及其特点。
- 6) 树的遍历方法。

(2) 简单应用

- 1) 二叉树的定义及树与二叉树的差别。
- 2) 二叉树的性质，了解相应的证明方法。
- 3) 二叉树的两种存储结构、特点及适用范围。
- 4) 最优二叉树和前缀编码的概念及特点。





- 5) 赫夫曼算法的思想。
- 6) 根据给定的叶结点及其权值构造出相应的最优二叉树。
- 7) 根据最优二叉树构造对应的赫夫曼编码。

(3) 综合应用

- 1) 二叉树的三种遍历算法，理解其执行过程。
- 2) 根据不同的遍历方法，应能得出其相应的结点访问次序。

(七) 图

1、考核知识点

- (1) 图的逻辑结构特征。
- (2) 图的常用术语及含义。
- (3) 图的邻接矩阵表示法存储结构。
- (4) 图的邻接表表示法。
- (5) 图的深度优先遍历。
- (6) 图的广度优先遍历。
- (7) 生成树和最小生成树。
- (8) 构造最小生成树的 PRIM 算法思想。
- (9) 构造最小生成树的 Kruskal 算法思想。
- (10) 拓扑排序。
- (11) 关键路径。
- (12) 关于最短路径的算法——Dijkstra 算法思想。

2、考核要求

(1) 理解

- 1) 图的逻辑结构及特征。
- 2) 图的常用术语及含义。
- 3) 生成树和最小生成树的概念。
- 4) 对给定的图遍历，画出深度优先和广度优先生成树或森林。
- 5) Prim 和 Kruskal 算法的基本思想。
- 6) 要求对给定的连通图，根据 Prim 和 Kruskal 算法构造最小生成树。。
- 7) 求单源点的最短路径问题的 Dijkstra 算法的基本思想。





- 8) 拓扑排序的基本思想和步骤。
- 9) 对给定的有向图，若拓扑序列存在，则要求写出一个或多个拓扑序列。

(2) 简单应用

- 1) 图的邻接矩阵表示法和邻接表表示法。
- 2) 根据应用问题的特点选择合适的存储结构。
- 3) 连通图及非连通图的深度优先搜索和广度优先搜索两种遍历算法。
- 4) 确定两种遍历的顶点访问序列。
- 5) 图的两种遍历和树的遍历之间的关系。
- 6) 两种遍历算法分别使用的数据结构（栈和队列）。

(八) 查找

1、考核知识点

- (1) 查找的定义关键字、查找、平均查找长度。
- (2) 静态查找表的查找算法（顺序查找、折半查找、分块查找(索引顺序表的查找)）其效率(最坏和平均长度)。
- (3) 二叉排序树的查找算法及其效率。
- (4) 平衡二叉树的定义。
- (5) 哈希法的特点。
- (6) 哈希函数和散列地址。
- (7) 处理冲突的方法：开放定址法和链地址法。开放定址法又分为线性探测再散列、二次探测再散列和伪随机探测再散列。

2、考核要求

(1) 识记

- 1) 查找在数据处理中的重要性。
- 2) 查找成功、不成功的含义。

(2) 简单应用

- 1) 顺序查找、折半查找、分块查找的基本思想、算法实现和查找效率分析。
- 2) 二叉排序树和二叉平衡树的定义、特点
- 3) 建立一棵二叉排序树的过程就是对输入序列的排序过程，输入序列对所建立的二叉排序树形态的影响





4) 哈希表、哈希函数、哈希地址（散列地址）、装填因子等有关概念

5) 哈希函数的构造方法和解决冲突的方法

（九）内部排序

1、考核知识点

（1）排序的目的、分类和排序方法的稳定性的定义。

（2）插入排序：直接插入排序的算法、折半插入排序的算法、希尔排序的思想。

（3）选择排序的思想

（4）堆排序的方法、堆的定义、初始堆的建立。

（5）起泡排序的思想。

（6）快速排序的算法、快速排序的最坏情况时间复杂度的分析。

（7）归并排序等

2、考核要求

（1）识记

1) 排序在数据处理中的重要性。

2) 排序方法稳定性的含义。

3) 排序方法的分类及算法好坏的评判标准。

（2）理解

1) 分类排序和其它几类排序方法的区别。

（3）简单应用

1) 堆、极小堆、极大堆、堆顶等有关概念和定义。

2) 堆的性质及堆与完全二叉树的关系。

3) 直接选择排序和堆排序的基本思想和算法实现。

4) 针对给定的输入序列，写出堆排序的排序过程。

（4）综合应用

1) 针对给定的输入序列，要能写出直接插入排序的排序过程。

2) 起泡排序的基本思想。

3) 快速排序的基本思想和算法实现，以及在最好、最坏和平均情况下的时间性能分析，了解算法的稳定性。

4) 枢轴元素的选择对排序的影响。

针对给定的输入序列，能写出快速排序的排序过程。





“C 语言程序设计”部分

(一) C 语言概述

1、考核知识点

C 语言的基础知识、上机编辑调试程序。

2、考核要求

(1) 了解 C 语言的特点、C 语言源程序的结构。

(2) 掌握 C 程序的上机步骤，Turbo C 集成开发环境下编辑、编译和运行 C 程序的基本操作方法。

(二) 程序的灵魂——算法

1、考核知识点

结构化程序的三种基本结构和 N-S 结构流程图。

2、考核要求

(1) 了解算法的概念、特点。

(2) 掌握结构化程序的三种基本结构和 N-S 结构流程图。

(三) 数据类型、运算符与表达式

1、考核知识点

C 语言的数据类型、运算符及表达式、输入输出函数的使用。

2、考核要求

(1) 理解标识符、常量与变量的概念；基本数据类型中的整型、单精度实型、双精度实型、字符型常量和变量的定义与使用方法。

(2) 掌握算术运算符和算术表达式、赋值运算符和赋值表达式。

(3) 掌握运算符的优先级和结合性；数据类型转换的方式；自增、自减运算。

(4) 掌握数据输入、输出函数的用法。

(四) 最简单的 C 程序设计——顺序程序设计

1、考核知识点

C 语言赋值语句，字符数据的输入与输出，格式输入与输出、顺序程序设计的基本思想。

2、考核要求

(1) 了解 C 语句格式。





(2) 理解字符数据的输入与输出，格式输入与输出。

(3) 掌握并应用顺序结构程序设计思想。

(五) 选择结构程序设计

1、考核要求

C 语言的关系运算符与关系表达式、逻辑运算符与逻辑表达式、选择结构程序设计思想。

2、考核知识点

(1) 理解关系运算符与关系表达式。

(2) 理解逻辑运算符与逻辑表达式。

(3) 掌握及应用 If 语句与 Switch 语句。

(六) 循环控制

1、考核知识点

C 语言的 While 语句、Do-While 语句和 For 语句构成循环程序设计。

2、考核要求

(1) 掌握 Break 语句与 Continue 语句的语法和用法。

(2) 掌握及应用 While 语句、Do-While 语句和 for 语句构成循环。

(七) 数组

1、考核知识点

C 语言的一维数组、二维数组及字符数组的定义和应用。

2、考核要求

(1) 掌握一维数组的定义与引用。

(2) 掌握二维数组的定义与引用。

(3) 掌握字符数组。

(4) 应用数组编写程序

(八) 函数

1、考核知识点

函数的基础知识、定义及函数的调用、应用数组作为函数参数。

2、考核要求

(1) 了解函数的定义。





(2) 理解函数的参数和函数的值、理解并应用数组作为函数参数。

(3) 掌握函数的调用、嵌套调用、递归调用。

(4) 应用函数编写程序

(九) 编译预处理

1、考核知识点

编译预处理中宏定义的概念、及文件包含处理，条件编译。

2、考核要求

(1) 了解宏定义。

(2) 了解文件包含处理，条件编译。

(十) 指针

1、考核知识点

指针与地址的概念、变量的指针和指针变量的指针变量、变量的指针和指针变量的指针变量、字符串的指针与指向字符串的指针变量、指针与函数、指针数组。

2、考核要求

(1) 理解指针与地址的概念、变量的指针和指针变量的指针变量。

(2) 掌握数组的指针与指向数组的指针变量、字符串的指针与指向字符串的指针变量。

(3) 应用指针与函数、指针数组。

(十一) 结构体与共用体

1、考核知识点

结构体和共用体类型数据的定义方法和引用方法、用指针和结构体构成链表, 单向链表的建立、输出、删除与插入。

2、考核要求

(1) 理解结构体类型的说明及结构体类型变量的定义、结构体变量的引用、共用体的基础知识

(2) 掌握结构体变量的初始化、结构体数组、指针与结构体数组

(3) 应用单向链表的建立、输出、删除与插入

(十二) 位运算

1、考核知识点





位运算符的含义及使用、简单的位运算。

2、考核要求

了解位运算符和位运算、位段。

(十三) 文件

1、考核知识点

文件类型指针、文件的打开与关闭、文件的读写。

2、考核要求

了解文件类型指针、文件的打开与关闭、文件的读写。

II. 考试形式、试卷结构及参考书

一、考试形式

考试采用闭卷、笔试形式。试卷满分 200 分，考试时间 150 分钟。

二、试卷结构

试卷包括选择题、填空题、解答题、算法设计和程序设计题。其中，选择题 60 分，填空题 30 分，解答题 60 分，算法设计和程序设计题 50 分。

三、参考书

1. 严蔚敏,吴伟民主编.数据结构(C语言版).清华大学出版社,2018年6月。
2. 《C语言程序设计教程(高职高专计算机基础教育精品教材)》,谭浩强,谭亦峰,金莹著,清华大学出版社,出版时间:2020-07

