

全国重点名校系列

新版

# 全国硕士研究生招生考试 考研专业课精品资料

【电子书】2024年中原工学院

813专业基础(含数据结构、操作系统)考研精品  
资料【第1册,共2册】

策划:辅导资料编写组

真题汇编 直击考点  
考研笔记 突破难点  
核心题库 强化训练  
模拟试题 查漏补缺

高分学长学姐推荐



【初试】2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研精品资料

说明：本套资料由高分研究生潜心整理编写，高清 PDF 电子版支持打印，考研首选资料。

一、重点名校考研真题汇编

1. 附赠重点名校真题汇编

①重点名校：数据结构 2016-2022 年考研真题汇编（暂无答案）

②重点名校：计算机操作系统 2010-2022 年考研真题汇编（暂无答案）

说明：本科目没有收集到历年考研真题，赠送重点名校考研真题汇编，因不同院校真题相似性极高，甚至部分考题完全相同，建议考生备考过程中认真研究其他院校的考研真题。

二、2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研资料

2. 《数据结构》考研相关资料

(1) 《数据结构》[笔记+课件+提纲]

①中原工学院 813 专业基础(含数据结构、操作系统)之《数据结构》考研复习笔记。

说明：本书重点复习笔记，条理清晰，重难点突出，提高复习效率，基础强化阶段必备资料。

②中原工学院 813 专业基础(含数据结构、操作系统)之《数据结构》本科生课件。

说明：参考书配套授课 PPT 课件，条理清晰，内容详尽，非本校课件，版权归属制作教师，本项免费赠送。

③中原工学院 813 专业基础(含数据结构、操作系统)之《数据结构》复习提纲。

说明：该科目复习重难点提纲，提炼出重难点，有的放矢，提高复习针对性。

(2) 《数据结构》考研核心题库（含答案）

①中原工学院 813 专业基础(含数据结构、操作系统)考研核心题库之《数据结构》算法设计题精编。

②中原工学院 813 专业基础(含数据结构、操作系统)考研核心题库之《数据结构》应用题精编。

说明：本题库涵盖了该考研科目常考题型及重点题型，根据历年考研大纲要求，结合考研真题进行的分类汇编并给出了详细答案，针对性强，是考研复习首选资料。

(3) 《数据结构》考研模拟题[仿真+强化+冲刺]

①2024 年中原工学院 813 专业基础(含数据结构、操作系统)之数据结构考研专业课五套仿真模拟题。

说明：严格按照本科目最新专业课真题题型和难度出题，共五套全仿真模拟试题含答案解析。

②2024 年中原工学院 813 专业基础(含数据结构、操作系统)之数据结构考研强化五套模拟题及详细答案解析。

说明：专业课强化检测使用。共五套强化模拟题，均含有详细答案解析，考研强化复习必备。

③2024 年中原工学院 813 专业基础(含数据结构、操作系统)之数据结构考研冲刺五套模拟题及详细答案解析。

说明：专业课冲刺检测使用。共五套冲刺预测试题，均有详细答案解析，最后冲刺必备资料。

3. 《计算机操作系统》考研相关资料

(1) 《计算机操作系统》[笔记+课件+提纲]

①中原工学院 813 专业基础(含数据结构、操作系统)之《计算机操作系统》考研复习笔记。

说明：本书重点复习笔记，条理清晰，重难点突出，提高复习效率，基础强化阶段必备资料。

②中原工学院 813 专业基础(含数据结构、操作系统)之《计算机操作系统》本科生课件。

说明:参考书配套授课 PPT 课件,条理清晰,内容详尽,非本校课件,版权归属制作教师,本项免费赠送。

③中原工学院 813 专业基础(含数据结构、操作系统)之《计算机操作系统》复习提纲。

说明:该科目复习重难点提纲,提炼出重难点,有的放矢,提高复习针对性。

### (2) 《计算机操作系统》考研核心题库(含答案)

①中原工学院 813 专业基础(含数据结构、操作系统)考研核心题库之《计算机操作系统》综合题精编。

说明:本题库涵盖了该考研科目常考题型及重点题型,根据历年考研大纲要求,结合考研真题进行的分类汇编并给出了详细答案,针对性强,是考研复习首选资料。

### (3) 《计算机操作系统》考研模拟题[仿真+强化+冲刺]

①2024 年中原工学院 813 专业基础(含数据结构、操作系统)之计算机操作系统考研专业课五套仿真模拟题。

说明:严格按照本科目最新专业课真题题型和难度出题,共五套全仿真模拟试题含答案解析。

②2024 年中原工学院 813 专业基础(含数据结构、操作系统)之计算机操作系统考研强化五套模拟题及详细答案解析。

说明:专业课强化检测使用。共五套强化模拟题,均含有详细答案解析,考研强化复习必备。

③2024 年中原工学院 813 专业基础(含数据结构、操作系统)之计算机操作系统考研冲刺五套模拟题及详细答案解析。

说明:专业课冲刺检测使用。共五套冲刺预测试题,均有详细答案解析,最后冲刺必备资料。

## 三、电子版资料全国统一零售价

4. 本套考研资料包含以上一、二部分(高清 PDF 电子版,不含教材),全国统一零售价:[¥]

特别说明:

①本套资料由本机构编写组按照考试大纲、真题、指定参考书等公开信息整理收集编写,仅供考研复习参考,与目标学校及研究生院官方无关,如有侵权、请联系我们将立即处理。

②资料中若有真题及课件为免费赠送,仅供参考,版权归属学校及制作老师,在此对版权所有者表示感谢,如有异议及不妥,请联系我们,我们将无条件立即处理!

## 四、2024 年研究生入学考试指定/推荐参考书目(资料不包括教材)

5. 中原工学院 813 专业基础(含数据结构、操作系统)考研初试参考书

《数据结构》严蔚敏著,清华大学出版社

《计算机操作系统》(第 4 版)汤子瀛等著,西安电子科技大学出版社

## 五、本套考研资料适用学院和专业

计算机学院:计算机科学与技术/电子信息

前沿信息技术研究院:计算机科学与技术/网络空间安全/电子信息

### 版权声明

编写组依法对本书享有专有著作权,同时我们尊重知识产权,对本电子书部分内容参考和引用的市面

上已出版或发行图书及来自互联网等资料的文字、图片、表格数据等资料，均要求注明作者和来源。但由于各种原因，如资料引用时未能联系上作者或者无法确认内容来源等，因而有部分未注明作者或来源，在此对原作者或权利人表示感谢。若使用过程中对本书有任何异议请直接联系我们，我们会在第一时间与您沟通处理。

因编撰此电子书属于首次，加之作者水平和时间所限，书中错漏之处在所难免，恳切希望广大考生读者批评指正。

## 目录

封面.....	1
目录.....	5
2024 年中原工学院 813 专业基础(含数据结构、操作系统)备考信息.....	10
中原工学院 813 专业基础(含数据结构、操作系统)考研初试参考书目.....	10
中原工学院 813 专业基础(含数据结构、操作系统)考研招生适用院系.....	10
2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研核心笔记.....	11
《数据结构》考研核心笔记.....	11
第 1 章 绪论.....	11
考研提纲及考试要求.....	11
考研核心笔记.....	11
第 2 章 线性表.....	22
考研提纲及考试要求.....	22
考研核心笔记.....	22
第 3 章 栈和队列.....	35
考研提纲及考试要求.....	35
考研核心笔记.....	35
第 4 章 串.....	39
考研提纲及考试要求.....	39
考研核心笔记.....	39
第 5 章 数组和广义表.....	45
考研提纲及考试要求.....	45
考研核心笔记.....	45
第 6 章 树和二叉树.....	50
考研提纲及考试要求.....	50
考研核心笔记.....	50
第 7 章 图.....	71
考研提纲及考试要求.....	71
考研核心笔记.....	71
第 8 章 动态存储管理.....	87
考研提纲及考试要求.....	87
考研核心笔记.....	87
第 9 章 查找.....	94
考研提纲及考试要求.....	94
考研核心笔记.....	94
第 10 章 内部排序.....	107
考研提纲及考试要求.....	107

考研核心笔记.....	107
第 11 章 外部排序.....	116
考研提纲及考试要求.....	116
考研核心笔记.....	116
第 12 章 文件.....	121
考研提纲及考试要求.....	121
考研核心笔记.....	121
<b>2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研辅导课件 .....</b>	<b>125</b>
《数据结构》考研辅导课件 .....	125
<b>2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研复习提纲 .....</b>	<b>236</b>
《数据结构》考研复习提纲 .....	236
<b>2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研核心题库 .....</b>	<b>239</b>
《数据结构》考研核心题库之算法设计题精编 .....	239
《数据结构》考研核心题库之应用题精编 .....	269
<b>2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研题库[仿真+强化+冲刺] .....</b>	<b>291</b>
中原工学院 813 专业基础(含数据结构、操作系统)之数据结构考研仿真五套模拟题.....	291
2024 年数据结构五套仿真模拟题及详细答案解析 (一) .....	291
2024 年数据结构五套仿真模拟题及详细答案解析 (二) .....	301
2024 年数据结构五套仿真模拟题及详细答案解析 (三) .....	310
2024 年数据结构五套仿真模拟题及详细答案解析 (四) .....	318
2024 年数据结构五套仿真模拟题及详细答案解析 (五) .....	327
中原工学院 813 专业基础(含数据结构、操作系统)之数据结构考研强化五套模拟题.....	334
2024 年数据结构五套强化模拟题及详细答案解析 (一) .....	334
2024 年数据结构五套强化模拟题及详细答案解析 (二) .....	344
2024 年数据结构五套强化模拟题及详细答案解析 (三) .....	354
2024 年数据结构五套强化模拟题及详细答案解析 (四) .....	364
2024 年数据结构五套强化模拟题及详细答案解析 (五) .....	372
中原工学院 813 专业基础(含数据结构、操作系统)之数据结构考研冲刺五套模拟题.....	381
2024 年数据结构五套冲刺模拟题及详细答案解析 (一) .....	381
2024 年数据结构五套冲刺模拟题及详细答案解析 (二) .....	391
2024 年数据结构五套冲刺模拟题及详细答案解析 (三) .....	400
2024 年数据结构五套冲刺模拟题及详细答案解析 (四) .....	408
2024 年数据结构五套冲刺模拟题及详细答案解析 (五) .....	416
<b>附赠重点名校：数据结构 2016-2022 年考研真题汇编 (暂无答案) .....</b>	<b>423</b>
第一篇、2022 年数据结构考研真题汇编 .....	423
2022 年沈阳工业大学数据结构考研专业课真题.....	423

2022 年广东财经大学 809 数据结构考研专业课真题 .....	425
2022 年中国人民解放军陆军工程大学 811 数据结构考研专业课真题 .....	429
2022 年南京审计大学 813 数据结构考研专业课真题 .....	433
2022 年常州大学 858 数据结构考研专业课真题 .....	435
2022 年湖南师范大学 865 数据结构考研专业课真题 .....	437
2022 年南京审计大学 911 数据结构考研专业课真题 .....	441
2022 年北京邮电大学 809 数据结构考研专业课真题 .....	443
2022 年暨南大学 830 数据结构考研专业课真题 .....	451
2022 年河北科技大学 847 数据结构考研专业课真题 .....	455
第二篇、2021 年数据结构考研真题汇编 .....	459
2021 年杭州电子科技大学数据结构考研专业课真题 .....	459
2021 年河北科技大学 847 数据结构考研专业课真题 .....	468
2021 年青岛理工大学 813 数据结构考研专业课真题 .....	471
2021 年北京化工大学 842 数据结构考研专业课真题 .....	474
2021 年常州大学 858 数据结构考研专业课真题 .....	482
2021 年广东财经大学 809 数据结构考研专业课真题 .....	484
2021 年广东财经大学 809 数据结构考研专业课真题 .....	488
2021 年湖南师范大学 865 数据结构考研专业课真题 .....	492
2021 年暨南大学 830 数据结构考研专业课真题 .....	496
2021 年南京审计大学 813 数据结构考研专业课真题 .....	500
2021 年南京审计大学 911 数据结构考研专业课真题 .....	502
2021 年宁波大学 916 数据结构与算法考研专业课真题 .....	504
2021 年沈阳工业大学 808 数据结构考研专业课真题 .....	509
2021 年四川化工大学 816 数据结构与算法考研专业课真题 .....	511
2021 年扬州大学 858 程序设计与数据结构考研专业课真题 .....	519
2021 年浙江工业大学 850 数据结构与计算机网络考研专业课真题 .....	521
第三篇、2020 年数据结构考研真题汇编 .....	525
2020 年青岛理工大学 813 数据结构考研专业课真题 .....	525
2020 年南京审计大学 813 数据结构考研专业课真题 .....	528
2020 年河北师范大学 823 计算机专业基础（数据结构）考研专业课真题 .....	530
2020 年广东工业大学 829 数据结构考研专业课真题 .....	533
2020 年暨南大学 830 数据结构考研专业课真题 .....	539
2020 年河北师范大学 833 数据结构考研专业课真题 .....	547
2020 年长沙理工大学 850 数据结构考研专业课真题 .....	555
2020 年浙江工业大学 850 数据结构与计算机网络考研专业课真题 .....	559
2020 年浙江工业大学 851 数据结构与软件工程考研专业课真题 .....	562
2020 年扬州大学 858 程序设计与数据结构考研专业课真题 .....	566
2020 年常州大学 858 数据结构考研专业课真题 .....	569
2020 年河北师范学 911 计算机专业基础（数据结构）考研专业课真题 .....	571
2020 年南京审计大学 911 数据结构考研专业课真题 .....	575

2020 年宁波大学 916 数据结构与算法考研专业课真题 .....	577
第四篇、2019 年数据结构考研真题汇编 .....	584
2019 年青岛理工大学 813 数据结构考研专业课真题 .....	584
2019 年重庆邮电大学 802 数据结构考研专业课真题 .....	587
第五篇、2018 年数据结构考研真题汇编 .....	595
2018 年贵州财经大学 808 数据结构考研专业课真题 .....	595
2018 年贵州财经大学 808 数据结构考研专业课真题 .....	597
2018 年湖南师范大学 865 数据结构考研专业课真题 .....	599
2018 年青岛理工大学 817 数据结构考研专业课真题 .....	603
2018 年山东大学 909 数据结构考研专业课真题 .....	606
2018 年沈阳工业大学 808 数据结构考研专业课真题 .....	608
2018 年天津城建大学 815 数据结构考研专业课真题 .....	610
2018 年长沙理工大学 850 数据结构考研专业课真题 .....	614
2018 年浙江理工大学 991 数据结构考研专业课真题 .....	618
2018 年重庆邮电大学 802 数据结构考研专业课真题 .....	623
第六篇、2017 年数据结构考研真题汇编 .....	629
2017 年广东工业大学 829 数据结构考研专业课真题 .....	629
2017 年桂林电子科技大学 910 数据结构考研专业课真题 .....	635
2017 年暨南大学 830 数据结构考研专业课真题 .....	639
2017 年南京师范大学 821 数据结构考研专业课真题 .....	644
2017 年青岛大学 910 数据结构考研专业课真题 .....	646
2017 年沈阳工业大学 808 数据结构考研专业课真题 .....	650
2017 年温州大学 831 数据结构考研专业课真题 .....	652
2017 年武汉纺织大学 848 数据结构考研专业课真题 .....	658
2017 年重庆邮电大学 802 数据结构 A 考研专业课真题 .....	662
第七篇、2016 年数据结构考研真题汇编 .....	668
2016 年安徽工业大学 861 数据结构考研专业课真题 .....	668
2016 年北京化工大学数据结构考研专业课真题 .....	674
2016 年广东工业大学 829 数据结构考研专业课真题 .....	683
2016 年暨南大学 830 数据结构考研专业课真题 .....	689
2016 年江苏大学 851 数据结构考研专业课真题 .....	694
2016 年解放军信息工程大学 820 数据结构考研专业课真题 .....	700
2016 年南京师范大学 829 数据结构考研专业课真题 .....	704
2016 年青岛大学 910 数据结构考研专业课真题 .....	707
2016 年沈阳工业大学 808 数据结构考研专业课真题 .....	712
2016 年武汉纺织大学 848 数据结构考研专业课真题 .....	713
2016 年湘潭大学 848 数据结构考研专业课真题 .....	716
2016 年湘潭大学 870 数据结构考研专业课真题 .....	719
2016 年燕山大学 810 数据结构考研专业课真题 .....	722
2016 年浙江理工大学 991 数据结构考研专业课真题 .....	724



## 2024 年中原工学院 813 专业基础(含数据结构、操作系统)备考信息

### 中原工学院 813 专业基础(含数据结构、操作系统)考研初试参考书目

《数据结构》严蔚敏著，清华大学出版社

《计算机操作系统》（第 4 版）汤子瀛等著，西安电子科技大学出版社

### 中原工学院 813 专业基础(含数据结构、操作系统)考研招生适用院系

计算机学院：计算机科学与技术/电子信息

前沿信息技术研究院：计算机科学与技术/网络空间安全/电子信息

2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研核心笔记

《数据结构》考研核心笔记

第 1 章 绪论

考研提纲及考试要求

- 考点：数据 (Data)
- 考点：数据对象 (DataObject)
- 考点：数据抽象与抽象数据类型
- 考点：逻辑结构
- 考点：数据结构课程地位

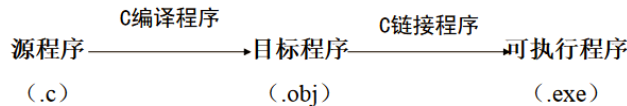
考研核心笔记

【核心笔记】数据结构的基本概念 (定义)

1. 数据 (Data)

定义

数据是描述客观事物的数值、字符以及能输入机器且能被处理的各种符号集合。  
数据包含整型、实型、布尔型、图象、字符、声音等一切可以输入到计算机中的符号集合。  
例如对 C 源程序



2. 数据元素 (DataElement)

定义

数据元素是组成数据的基本单位，是数据集合的个体，在计算机中通常作为一个整体进行考虑和处理。

例如：

数据项 ↓					
学号	姓名	性别	籍贯	出生年月	住址
101	赵虹玲	女	河北	1983.11	北京
...	...	...	...	...	...

← 数据元素

3. 数据对象 (DataObject)

定义

数据对象是性质相同的数据元素的集合，是数据的一个子集。

例如：整数集合： $N = \{0, \pm 1, \pm 2, \dots\}$  无限集

字符集合： $C = \{ 'A', 'B', \dots, 'Z' \}$  有限集

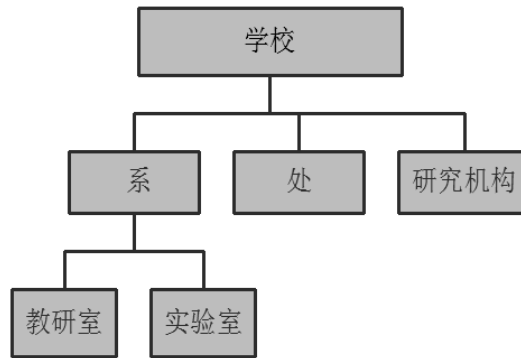
4. 数据结构 (Data Structure)

(1) 定义

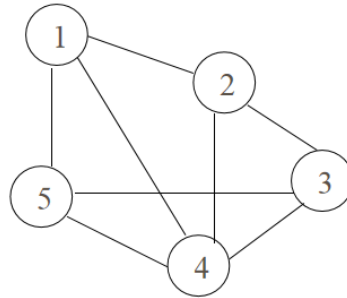
数据结构是指相互之间存在一种或多种特定关系的数据元素集合，是带有结构的数据元素的集合，它指的是数据元素之间的相互关系，即数据的组织形式。例如表结构：

学号	姓名	性别	籍贯	出生年月	住址
101	赵虹玲	女	河北	1983.11	北京
...	...	...	...	...	...

(2) 树型结构



(3) 图结构



5.数据类型(DataType)

定义

数据类型是一组性质相同的值集合以及定义在这个值集合上的一组操作的总称。

如在高级语言中，整型类型的取值范围为：

-32768~+32767，运算符集合为加、减、乘、除、取模，即+、-、\*、/、%

(1) 高级语言中的数据类型分为两大类：

①原子类型，其值不可分解。如 C 语言中的标准类型（整型、实型、字符型、）。

②结构类型，其值是由若干成分按某种结构组成的，因此是可以分解的，并且它的成分可以是非结构的，也可以是结构的。

6.数据抽象与抽象数据类型

- (1) 数据的抽象
- (2) 抽象数据类型(AbstractDataType)
- (3) 抽象数据类型实现
- (4) ADT 的表示与实现
- (5) 面向对象的概念
- (6) 结构化的开发方法与面向对象开发方法不同点

【核心笔记】数据结构的内容

1.逻辑结构

(1) 定义

数据的逻辑结构是指数据元素之间逻辑关系描述。

(2) 形式化描述： $Data\_Structure = (D, R)$  其中 D 是数据元素的有限集，R 是 D 上关系的有限集。

(3) 四类基本的结构

集合结构、线性结构、树型结构、图状结构。

2.集合结构

(1) 定义：结构中的数据元素之间除了同属于一个集合的关系外，无任何其它关系。

例如

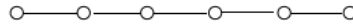
集合



### 3.线性结构

定义：结构中的数据元素之间存在着一对一的线性关系。

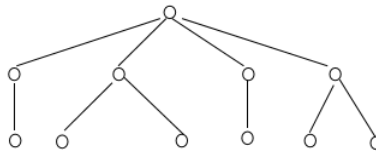
例如：线性表



### 4.树型结构

定义：结构中的数据元素之间存在着一对多的层次关系。

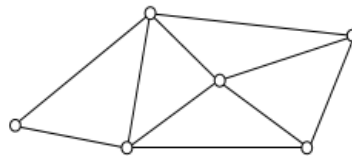
例如：树



### 5.图状结构或网状结构

定义：结构中的数据元素之间存在着多对多的任意关系。

例如



### 6.逻辑结构

综上所述，数据的逻辑结构可概括为

逻辑结构

线性结构——线性表、栈、队、字符串数组、广义表

非线性结构——树、图

### 7.存储结构

(1) 定义：存储结构（又称物理结构）是逻辑结构在计算机中存储映象，是逻辑结构在计算机中的实现，它包括数据元素的表示和关系的表示。

(2) 形式化描述：

D 要存入机器中，建立一从 D 的数据元素到存储空间 M 单元映象 S,  $D \rightarrow M$ , 即对于每一个  $d, d \in D$ , 都有唯一的  $z \in M$  使  $S(d) = z$ , 同时这个映象必须明显或隐含地体现关系 R。

(3) 逻辑结构与存储结构的关系为：

存储结构是逻辑关系的映象与元素本身映象，是数据结构的实现；逻辑结构是数据结构的抽象。

(4) 数据元素之间关系在计算机中的表示方法：

顺序映象（顺序存储结构）

非顺序映象（非顺序存储结构）

### 8.运算集合

例如工资表

编 号	姓 名	性 别	基本工资	工龄工资	应扣工资	实发工资
100001	张爱芬	女	345. 67	145. 45	30. 00	451. 12
100002	李 林	男	445. 90	185. 60	45. 00	586. 50
100003	刘晓峰	男	345. 00	130. 00	25. 00	450. 00
100004	赵 俊	女	560. 90	225. 90	65. 00	721. 80
100005	孙 涛	男	450. 60	190. 80	50. 00	591. 80
...	...	...	...	...	...	...
100121	张兴强	男	1025. 98	365. 53	100. 00	1291. 51

## 9. 数据结构的内容

综上所述，数据结构的内容可归纳为三个部分

逻辑结构、存储结构和运算集合：按某种逻辑关系组织起来的一批数据，按一定的映象方式把它存放在计算机存储器中，并在这些数据上定义了一个运算的集合，就叫做数据结构。

### 【核心笔记】算法

#### 1. 算法 (Algorithm) 定义

定义：算法是规则的有限集合，是为解决特定问题而规定的一系列操作。

#### 2. 算法的特性

- (1) 有限性：有限步骤之内正常结束，不能形成无穷循环
- (2) 确定性：算法中的每一个步骤必须有确定含义，无二义性得以实现。
- (3) 输入：有多个或 0 个输入
- (4) 输出：至少有一个或多个输出
- (5) 可行性：原则上能精确进行，操作可通过已实现基本运算执行有限次而完成

#### 3. 算法设计的要求

- (1) 算法特征
  - ① 算法的正确性
  - ② 可读性
  - ③ 健壮性
  - ④ 高效率和低存储量

### 【核心笔记】算法描述的工具

#### 1. 算法、语言、程序的关系

- (1) 算法：描述了数据对象的元素之间的关系（包括数据逻辑关系、存储关系描述）。
- (2) 描述算法的工具：算法可用自然语言、框图或高级程序设计语言进行描述。
- (3) 程序是算法在计算机中的实现。

#### 2. 设计实现算法过程步骤

- (1) 找出与求解有关的数据元素之间的关系
- (2) 确定在某一数据对象上所施加运算
- (3) 考虑数据元素的存储表示
- (4) 选择描述算法的语言
- (5) 设计实现求解的算法，并用程序语言加以描述。

#### 3. 类描述算法的语言选择

(1) 类语言：类语言是接近于高级语言而又不是严格的高级语言，具有高级语言的一般语句设施，撇掉语言中的细节，以便把注意力主要集中在算法处理步骤本身的描述上。

- (2) 对 C 语言作以下描述  
赋值语句

2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研辅导课件

《数据结构》考研辅导课件

<p style="text-align: center;"><b>第一章 绪论</b></p> <p>1.1 什么是数据结构 1.2 基本概念 1.3 抽象数据类型 1.4 算法及其分析</p> <p style="text-align: right;">1</p>	<p style="text-align: center;"><b>学习提要</b></p> <ul style="list-style-type: none"> <li>掌握本课程所涉及到的基本名词、术语和概念，特别是数据的逻辑结构和存储结构之间的关系及性质。</li> <li>了解抽象数据类型的定义、表示和实现方法。</li> <li>理解算法设计的五个要素和基本要求；掌握算法效率的度量方法，着重学习算法的时间复杂度分析。</li> </ul> <p style="text-align: right;">2</p>																				
<p style="text-align: center;"><b>教学重点</b></p> <p>(1)数据、数据元素、数据项； (2)逻辑结构和存储结构在概念上的联系与区别； (3)数据结构及其三个组成部分； (4)抽象数据类型和数据抽象； (5)评价算法优劣的标准及方法。</p> <p style="text-align: right;">3</p>	<p style="text-align: center;"><b>1.1什么是数据结构</b></p> <p>一、为什么要学习数据结构？ 1、电子计算机的主要用途： • 早期：     主要用于数值计算。 • 后来：     处理逐渐扩大到非数值计算领域（能处理多种复杂的具有一定结构关系的数据）。</p> <p style="text-align: right;">4</p>																				
<p style="text-align: center;">用计算机解决问题的过程</p> <p style="text-align: right;">5</p>	<p style="text-align: center;"><b>书目卡片</b></p> <p>登录号： 书名： 作者名： 分类号： 出版单位： 出版时间： 价格：</p> <p style="text-align: right;">6</p>																				
<p style="text-align: center;"><b>引例</b></p> <p>书目自动检索系统的数学模型</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>001</td><td>高等数学</td><td>樊映川</td><td>S01</td></tr> <tr><td>002</td><td>理论力学</td><td>罗远祥</td><td>L01</td></tr> <tr><td>003</td><td>高等数学</td><td>华罗庚</td><td>S01</td></tr> <tr><td>004</td><td>线性代数</td><td>秦汝书</td><td>S02</td></tr> <tr><td>.....</td><td>.....</td><td>.....</td><td>.....</td></tr> </table> <p style="text-align: right;">7</p>	001	高等数学	樊映川	S01	002	理论力学	罗远祥	L01	003	高等数学	华罗庚	S01	004	线性代数	秦汝书	S02	.....	.....	.....	.....	<p style="text-align: center;"><b>求解非数值计算的问题：</b></p> <p>主要考虑的是设计出合适的数据结构及相应的算法。 即：首先要考虑对相关的各种信息如何表示、组织和存储？ 因此，简单说来，数据结构是一门研究非数值计算的程序设计问题中计算机的操作对象以及它们之间的关系和操作的学科。</p> <p style="text-align: right;">8</p>
001	高等数学	樊映川	S01																		
002	理论力学	罗远祥	L01																		
003	高等数学	华罗庚	S01																		
004	线性代数	秦汝书	S02																		
.....	.....	.....	.....																		

### 问题：学习数据结构有什么用？

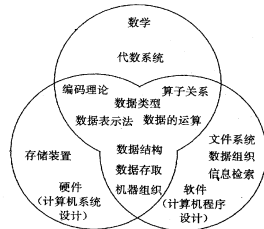
- 答：计算机内的数值运算依靠数学方程，而非数值运算（如表、树、图等）则要依靠数据结构。
- 同样的数据对象，用不同的数据结构来表示，运算效率可能有明显的差异。
  - 程序设计的实质是对实际问题选择一个好的数据结构，加之设计一个好的算法。而好的算法在很大程度上取决于描述实际问题的数据结构。
- 《算法+数据结构=程序》
- ①提高复杂程序设计的能力
  - ②培养算法设计能力
  - ③为后继课程（如操作系统、编译原理等）打基础。

### 数据结构课程的形成和发展

- 形成阶段
  - 60年代初期，“数据结构”有关的内容散见于操作系统、编译原理和表处理语言等课程。
  - 1968年，美唐·欧·克努特教授开创了数据结构的最初体系，《计算机程序设计技巧》第一卷《基本算法》，“数据结构”被列入美国一些大学计算机科学系的教学计划。
- 发展阶段：
  - 数据结构的概念不断扩充，包括了网络、集合代数论、关系等“离散数学结构”的内容。
  - 70年代后期，80年代初，我国高校陆续开设该课程。

### 《数据结构课程》所处的地位：

介于数学、计算机硬件和计算机软件三者之间的一门核心课程。



### 1.2 基本概念和术语

#### 1.2.1 基本概念

##### 1. 数据 (data)：

数据是信息的载体，是描述客观事物的数、字符、以及所有能输入到计算机中并被计算机程序识别和处理的符号的集合，是计算机程序加工的“原料”。

分类： $\begin{cases} \text{数值性数据} \\ \text{非数值性数据} \end{cases}$

### 2、数据元素 (data element)

- 数据的基本单位。在计算机程序中常作为一个整体进行考虑和处理。有时一个数据元素可以由若干数据项(Data Item)组成。
- 数据项是数据不可分割的最小标识单位。
- 数据元素又称为元素、结点、记录。

学号	姓名	成绩
001	LiLy	89
002	Yang	98
003	Zhao	78

### 3、数据对象 (data object)

- 数据对象是具有相同性质的数据元素的集合。
- 整数数据对象

$$N = \{0, \pm 1, \pm 2, \dots\}$$

- 字母字符数据对象

$$C = \{ 'A', 'B', \dots, 'Z' \}$$

- 学生成绩数据对象

$$C_j = \{ ( '101', 'jane', 80 ), ( '102', 'jack', 90 ), ( '103', 'jerry', 75 ) \}$$

学号	姓名	成绩
001	LiLy	89
002	Yang	98
003	Zhao	78

### 4、数据结构(data structure)


- 数据结构是相互之间存在一种或多种特定关系的数据元素的集合。
- 在任何问题中，数据元素都不是孤立存在的，而是在它们之间存在着某种关系，这种数据元素相互之间的关系称为结构(Structure)。
- 数据结构是一堆数据元素和这些数据元素之间的关系的总和。

按数据元素之间关系的不同特性，通常有4类基本结构

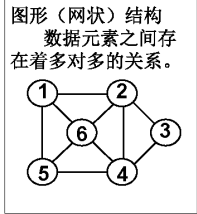
- (1) 集合 结构中的数据元素除了“同属于一个集合”外，别无其它关系。
- (2) 线性结构 结构中的数据元素之间存在一对一的关系。
- (3) 树型结构 结构中的数据元素之间存在一对多的关系。
- (4) 图状结构或网状结构 结构中的数据元素之间存在多对多的关系。



**集合**  
数据元素之间无特殊关系



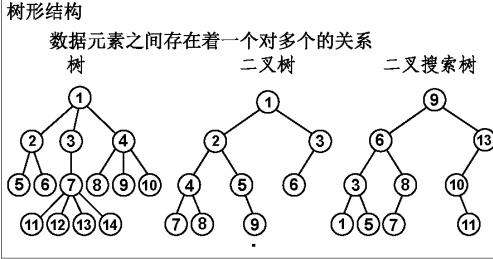
**图形（网状）结构**  
数据元素之间存在着多对多的关系。



17

**树形结构**  
数据元素之间存在着一个对多个的关系

树                      二叉树                      二叉搜索树



**线性结构**  
数据元素之间存在着一个对一个的关系

20

### 数据结构的定义

用一个二元组表示，记为：  
 $\text{Data\_Structure} = (D, S)$

其中，D 是数据元素的有限集（即一个数据对象），S 是该对象中所有数据成员之间的关系关系的有限集合。

在计算机科学中，对复数的定义：复数是一种数据结构  
 $\text{Complex} = (C, R)$

其中：C 是包含两个实数的集合  $\{C_1, C_2\}$ ，  
 $R = \{P\}$ , P 是定义在 C 上的一种关系  $\{<C_1, C_2>\}$ 。

20

**例、用数据结构如何描述2行3列矩阵：**

- 它是一个含6个数据元素  $\{a_1, a_2, a_3, a_4, a_5, a_6\}$  的集合，且集合上存在“行关系”和“列关系”两个次序关系，其中  
 行关系为  $\{<a_1, a_2>, <a_2, a_3>, <a_4, a_5>, <a_5, a_6>\}$   
 列关系为  $\{<a_1, a_4>, <a_2, a_5>, <a_3, a_6>\}$ 。
- $<x, y>$  意为 x 和 y 之间存在 "x 领先于 y" 的次序关系。

$$\begin{pmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \end{pmatrix}$$

**思考：**  
如何描述一个一行六列的矩阵？

20

**例1-5** 假设我们需要编制一个事务管理的程序，管理学校科学研究课题小组的各项事务，则首先要为程序的操作对象——课题小组设计一个数据结构。假设每个小组由一位教师、一至三名研究生及一至六名本科生组成，小组成员之间的关系是：教师指导研究生，而由每位研究生指导一至两名本科生。

则可以如下定义数据结构：  
 $\text{Group} = (P, R)$

其中：P =  $\{T, G_1, \dots, G_n, S_{11}, \dots, S_{nm}\} \quad 1 \leq n \leq 3, 1 \leq m \leq 2,$   
 $R = \{R_1, R_2\}$   
 $R_1 = \{<T, G_i> \mid 1 \leq i \leq n, 1 \leq n \leq 3\}$   
 $R_2 = \{<G_i, S_{ij}> \mid 1 \leq i \leq n, 1 \leq j \leq m, 1 \leq n \leq 3, 1 \leq m \leq 2\}$

22

### 5、数据的逻辑结构

- 数据的逻辑结构从逻辑关系上描述数据，可以看作是从具体问题抽象出来的数据模型，与数据的存储无关，也与数据元素本身的形式、内容、相对位置无关；
- 数据的逻辑结构分类：
  - 线性结构  
线性表、栈、队列、串
  - 非线性结构  
树、图（或网络）

22

### 6、数据的存储结构

- 数据结构在计算机中的表示（或称映像）称为数据的存储结构，又称为物理结构。它包括数据元素的表示和关系的表示。
  - 数据元素的表示：位、字长、元素、结点、数据域
  - 关系的表示两种基本的存储方法：
    - ①顺序映像（顺序存储结构）
    - ②非顺序映像（链式存储结构）

23

- 顺序存储结构：  
借助数据元素在存储器中的相对位置来表示数据元素之间的逻辑关系。  
所有元素存放在一片连续的存储单元中，逻辑上相邻的元素存放在计算机内存内仍然相邻。
- 链式存储结构：  
在每一个数据元素中增加一个存放地址的指针，借助该指针来表示数据元素之间的逻辑关系。  
所有元素存放在可以不连续的存储单元中，但元素之间的关系可以通过地址（指针）确定，逻辑上相邻的元素存放在计算机内存后不一定是相邻的。

24



### 存储结构的描述

- 存储结构的描述方法随编程环境的不同而不同，通常可用高级编程语言中提供的数据类型描述之。
- 例如，用一维数组类型描述顺序存储结构，用指针描述链式存储结构。
- 例如，定义“日期”为：  

```
typedef struct {
    int y; //年号 Year
    int m; //月号 Month
    int d; //日号 Day
} DateType; //日期类型
```

 同样，此时对数据元素也要借用高级编程语言中的数据类型描述之。

25

### 7、数据类型 (data type)

- 数据类型是一个值的集合和定义在这个值集上的一组操作的总称。
- 数据类型可分两类：原子类型和结构类型。
- 例如：在C语言中  
 原子类型：整型、实型、字符型等  
 结构类型：数组、结构体、联合等

26

### 9、抽象数据类型 (Abstract Data Type简称ADT)

- 由用户定义，用以表示应用问题的数据模型
- 指一个数学模型以及定义在此数学模型上的一组操作。
- 例如：计算机拥有的整数类型。
- 它与数据类型实质上是一个概念，但其特征是使用与实现分离，实行封装和信息隐蔽（独立于计算机）。

27

### 抽象数据类型的描述方法

- 抽象数据类型可用 (D, S, P) 三元组表示，其中，D是数据对象，S是D上的关系集，P是对D的基本操作集。  
 ADT 抽象数据类型名 {  
 数据对象：〈数据对象的定义〉  
 数据关系：〈数据关系的定义〉  
 基本操作：〈基本操作的定义〉  
 } ADT 抽象数据类型名
- 基本操作的定义格式为：  
 基本操作名 (参数表)  
 初始条件：〈初始条件描述〉  
 操作结果：〈操作结果描述〉
- 数据对象和数据关系的定义用伪码（不真正执行的符号）描述。

28

- 基本操作有两种参数：  
 赋值参数只为操作提供输入值；  
 引用参数以&打头，除了可以提供输入值外，还将返回操作结果。
- “初始条件”描述了操作执行之前数据结构和参数应满足的条件，若不满足，则操作失败，并返回相应出错信息。若初始条件为空，则省略之。
- “操作结果”说明了操作正常完成之后，数据结构的变化状况和应返回的结果。

29

### 举例——抽象数据类型复数的定义：

```
ADT Complex {
    数据对象：D = {e1, e2 | e1, e2 ∈ RealSet }
    数据关系：R1 = {<e1, e2 | e1是复数的实数部分, e2是复数的虚数部分 }
    基本操作：
    InitComplex( &Z, v1, v2 )
        操作结果：构造复数Z, 其实部和虚部分别被赋以参数v1和v2的值。
    DestroyComplex( &Z )
        操作结果：复数Z被销毁。
}
```

30

```
GetReal( Z, &realPart )
    初始条件：复数已存在。
    操作结果：用realPart返回复数Z的实部值。
GetImag( Z, &ImagPart )
    初始条件：复数已存在。
    操作结果：用ImagPart返回复数Z的虚部值。
Add( z1, z2, &sum )
    初始条件：z1, z2是复数。
    操作结果：用sum返回两个复数z1, z2的和值。
} ADT Complex
```

假设：z1和z2是上述定义的复数，则Add(z1, z2, z3)操作的结果将得到z3=z1+z2

31

### 1.3抽象数据类型的表示与实现

- 抽象数据类型可通过固有数据类型来表示和实现，即利用处理器中已存在的数据类型来说明新的结构，用已经实现的操作来组合新的操作。
- 一、类C语言简要说明
- ```
(1) 预定义常量和类型：
//函数结果状态代码
#define TRUE 1
#define FALSE 0
#define OK 1
#define ERROR 0
#define INFEASIBLE -1
#define OVERFLOW -2
//Status是函数的类型，其值是函数结果状态代码
typedef int Status;
```

32

2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研复习提纲

《数据结构》考研复习提纲

《数据结构》复习提纲

第一章 绪论

本章主要内容:

- 1.1 什么是数据结构
- 1.2 基本概念和术语
- 1.3 抽象数据类型的表示与实现
- 1.4 算法和算法分析

复习要求: 理解数据结构的相关概念, 对本章内容及知识体系结构有初步的认识, 并能够对算法的时间效率进行分析。

重点及难点: 本章复习重点是对数据结构知识体系的理解, 难点是算法分析。

第二章 线性表

本章主要内容:

- 2.1 线性表的类型定义
- 2.2 线性表的顺序表示和实现
- 2.3 线性表的链式表示和实现
- 2.4 一元多项式的表示及相加

本章复习要求: 理解线性表的逻辑结构和两种存储结构, 并能够应用线性表解决实际问题。

本章重点及难点: 本章复习重点是对线性表的理解, 难点是线性链表的各种操作。

第三章 栈和队列

本章主要内容:

- 3.1 栈
- 3.2 栈的应用举例
- 3.3 栈与递归的实现(\*)
- 3.4 队列
- 3.5 离散事件模拟(\*)

本章复习要求: 理解操作受限的线性结构栈和队列的特点, 从栈和队列的应用体会栈和队列在解决实际问题中的重要性。

本章重点及难点: 本章复习重点是对栈和队列的理解, 难点是栈与递归的关系。

第四章 串

本章主要内容:

- 4.1 串类型的定义
- 4.2 串的实现和表示
- 4.3 串的模式匹配算法(\*)
- 4.4 串操作应用举例

本章复习要求: 理解串结构的特点, 能够综合应用串的常见操作方法解决问题。

本章重点及难点: 本章复习重点是对串的理解的串操作的应用, 难点是模式匹配算法。

第五章 数组和广义表

本章主要内容:

- 5.1 数组的定义
- 5.2 数组的顺序表示和实现
- 5.3 矩阵的压缩存储
- 5.4 广义表的定义
- 5.5 广义表的存储结构
- 5.6 M 元多项式的表示(\*)
- 5.7 广义表的递归算法(\*)

本章复习要求: 从数组和广义表的表示和实现进一步加深对线性结构的理解, 掌握矩阵的压缩存储和广义表的存储。

本章重点及难点: 本章复习重点是对矩阵的压缩存储, 难点是广义表的存储和操作。

第六章 树和二叉树

本章主要内容:

- 6.1 树的定义和基本术语

- 6.2 二叉树
- 6.3 遍历二叉树和线索二叉树
- 6.4 树和森林
- 6.5 树与等价问题(\*)
- 6.6 赫夫曼树及其应用
- 6.7 回溯法与树的遍历
- 6.8 树的计数

本章复习要求：理解非线性结构树和二叉树的特点。

本章重点及难点：本章复习重点是是二叉树的遍历和赫夫曼树，难点是线索二叉树的操作以及回溯法与树的遍历。

### 第七章 图

本章主要内容：

- 7.1 图的定义和术语
- 7.2 图的存储结构
- 7.3 图的遍历
- 7.4 图的连通性问题
- 7.5 有向无环图及其应用
- 7.6 最短路径

本章复习要求：理解图的结构特点以及图和树间的关系,掌握图的遍历和常见算法。

本章重点及难点：本章复习重点是是图的遍历和图的操作，难点是图的生成树算法和最短路径算法。

### 第八章 动态存储管理

本章主要内容：

- 8.1 概术
- 8.2 可以得用空间表及分配方法
- 8.3 边界标识法
- 8.4 伙伴系统

本章复习要求：体会数据结构在内存管理中的作用,掌握内存管理中常用结构和方法。

本章重点及难点：本章复习重点是是可利用空间表，难点是伙伴系统

### 第九章 查找

本章主要内容：

- 9.1 静态查找表
- 9.2 动态查找表
- 9.3 哈希表

本章复习要求：掌握各种查找方法和它们的优缺点,能够针对实际问题选择合适的查找方法。

本章重点及难点：本章复习重点是是二分查找，树表查找和哈希表，难点是 B-树查找。

### 第十章 内部排序

本章主要内容：

- 10.1 概述
- 10.2 插入排序
- 10.3 快速排序
- 10.4 选择排序
- 10.5 归并排序
- 10.6 基数排序
- 10.7 各种内部排序方法的比较讨论

本章复习要求：掌握各种排序方法和它们的优缺点,能够针对实际问题选择合适的排序方法。

本章重点及难点：本章复习重点是是各种排序方法的实现，难点是针对实际问题如何选择合适的排序方法。

### 第十一章 外部排序

本章主要内容：

- 11.1 外存信息的存取
- 11.2 外部排序的方法
- 11.3 多路平衡归并的实现
- 11.4 置换-选择排序
- 11.5 最佳归并树

本章复习要求：体会外部排序和内部排序的不同之处,掌握各种外部排序方法。

本章重点及难点：本章复习重点是是各种外部排序方法，难点是败者树的实现。

## 第十二章 文件

本章主要内容：

- 12.1 有关文件的基本概念
- 12.2 顺序文件
- 12.3 索引文件
- 12.4 ISAM文件和VSAM文件
- 12.5 直接存取文件(散列文件)
- 12.6 多关键字文件

本章复习要求：理解文件的概念和各种文件的结构特点。

本章重点及难点：本章复习重点是是各种文件的结构特点，难点是 ISAM 文件和 VSAM 文件。

2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研核心题库

《数据结构》考研核心题库之算法设计题精编

1. 已知顺序表中有  $n$  个记录，表中记录不依关键字有序排序，编写一算法，为该顺序表建立一个有序的索引表(依关键字递增排列)，索引表中的每一项应含有记录的关键字和该记录在顺序表中的序号。要求算法的时间复杂度在最好的情况下能达到  $O(n)$ 。

**【答案】**

```

typedef struct index{                               //索引类型
    Keytype key;                                   //关键字
    int no;                                       //在主表中的序号
}indextype;

typedef struct{                                     //表结点类型
    ElemType data;                               //表中元素
    KeyType key;                                 //表中元素关键字
}RectType;

void createindex (RectType r[], indextype &idx[ ], int n){ //为主表 r 建立一个索引表 idx
    int i,j;
    for(i=1; i<=n; i++){
        j = i-1;
        while(j>0&&idx[j].key){                 //把大于 r[i].key 的所有 idx[j].key 后移
            idx[j+1] = idx[j];
            j--;
        }
        idx[j+1].key = r[i].key;                 //把 r[i].key 插入到 idx 中，并记录下其序号 i
        idx[j+1].no = i;
    }
}
    
```

2. 已知中序线索二叉树  $T$  右子树不空。设计算法，将  $S$  所指的结点作为  $T$  的右子树中的一个叶子结点插入进去，并使之成为  $T$  的右子树的(中序序列)第一个结点(同时要修改相应的线索关系)。

**【答案】**若使新插入的叶子结点  $S$  成  $T$  右子树中序序列的第一个结点，则应在  $T$  的右子树中最左面的结点(设为  $p$ )处插入，使  $S$  成为结点  $p$  的左子女，则  $S$  的前驱是  $T$ ，后继是  $p$ 。

```

p=T->rchild;                                       //p 指向 T 的右子女
while(p->ltag==0) p=p->lchild;                   //p 最终指向 T 的右子树中最左面的结点
S->ltag=1; S->rtag=1;                             //S 是叶子，其左右标记均为 1
S->lchild=T; S->rchild=p;                         //S 的前驱是根结点 T，后继是结点 p
p->lchild=S; p->ltag=0;                           //将 p 的左子女指向 S，并修改左标志为 0
    
```

3. 有 15 个人围成一圈，顺序从 1 到 15 编号。从第一个人开始报数，凡报到  $n$  的人退出圈子。用 C 语言写出程序，输入  $n$  ( $n \geq 1$ ) 的值，输出最后留在圈子中的人的编号。

**【答案】**算法描述如下：

```

#include <stdio.h>
#define people 15
struct person{
    int number;
    int nextp;
    link[peoplenum+1];
}
main()
{
    
```

```

int n;
int i,count,h;
scanf('%d",&n);
IF (n<1) {
    printf('n 的值必须大于 1');
    return;
}
FOR(i=1;i<=peoplenum;i++){
    IF(i==peoplenum) link[i].nextp=1;
    ELSE link[i].nextp=i+1;
    link[i].number=1;
}
printf('\n');
count=0;
h=peoplenum;
WHILE (count<peoplenum-1){
    WHILE(i!=n){
        h=link[h].nextp;
        IF (link[h].number) i++;
    }
    link[h].number=0;
    count++;
}
printf('\n 最后留在圈里的人的序号是: ');
FOR(i=1;i<=peoplenum;i++)
    IF(link[i].number)
        printf('%6d\n',link[i].number);
}

```

4. 结点类型和存储方式如下:

```

TYPE node=RECORD
    key:integer;
    info:datatype;
    count:integer
END;
VAR R:ARRAY[1..n] OF node;

```

请给出一个排序方法,不移动结点存储位置,只在结点的 count 字段记录结点在排序中的序号(设排序码最大的结点序号为 1)。

**【答案】**算法描述如下:

```

PROCEDURE CountSort(VAR R:ARRAY[1..n] OF node);
VAR i,j:integer;
BEGIN
FOR i:=1 TO n DO
    R[i].count:=0;
FOR i:=1 TO n DO
    FOR j:=1 TO n DO
        IF R[i].key<R[j].key
            THEN R[i].count:=R[i].count+1
        ELSE R[i].count:=1
END;

```

5. 已知一个递增有序表 $R[1..4n]$ ，并且表中没有关键字相同的记录。按如下方法查找一个关键字为 $k$ 的记录：先在编号为 $4, 8, 12, \dots, 4n$ 的记录中进行顺序查找，或者查找成功，或者由此确定一个继续进行折半查找的范围。

(1) 设计满足上述过程的查找算法。

(2) 分析成功查找情况下的平均查找长度，和对整个表进行折半查找相比哪个算法较好？

(3) 为了提高效率，对本算法可以做何改进。

**【答案】** (1) 先在编号为 $4, 8, 12, \dots, 4n$ 的记录中进行顺序查找，若没有找到，找到大于 $k$ 的位置 $i$ ，然后在 $R[i-3..i-1]$ 的范围内进行折半查找。对应的算法如下。

```
int FindElem(SeqList R,int n,KeyType k)
{
    int i=4,low,high,mid;
    if (k<R[1].key || k>R[4*n].key) return -1;
    while(i<=4*n)
    {
        if (R[i].key==k)
            return i;                //查找成功返回
        else if (R[i].key<k)
            i+=4;
        else
            //找到大于 k 的位置 i
            break;
    }
    low=i-3;high=i-1;
    while (low<=high)
    {
        mid=(low+high)/2;
        if (R[mid].key==k)
            //查找成功返回
            return mid;
        if (R[mid].key>k)
            //继续在 R[low..mid-1]中查找
            high=mid-1;
        else
            //继续在 R[mid+1..high]中查找
            low=mid+1;
    }
    return -1;
}
```

(2) 在成功查找情况下，顺序查找中平均关键字比较次数为 $(n+1)/2$ ，然后在 3 个元素的范围内进行折半查找，其平均关键字比较次数为 $\log_2 4 - 1 = 1$ ，所以总的平均查找长度为 $(n+1)/2 + 1 = (n+3)/2$ 。若对整个表进行折半查找，平均查找长度为 $-\log_2(4n+1) - 1$ ，显然折半查找更好些。

(3) 对本算法可做这样的改进：由于编号为 $4, 8, 12, \dots, 4n$ 的记录是递增有序的，可以将顺序查找改为折半查找，然后在确定范围内再进行折半查找。

6. 设计一个算法，从一给定的顺序表 $L$ 中删除元素值在 $x$ 到 $y$  ( $x \leq y$ )之间的所有元素，要求以较高的效率来实现，空间复杂度为 $O(1)$ 。

**【答案】** 可以采用上例的解法，只是将 $L.data[i] == x$ 的条件改为 $L.data[i] >= x \ \&\& \ L.data[i] <= y$ 。对应算法如下。

```
void delallxy1(SqList &L,ElemType x, ElemType y)
{
    int i,k=0;
    for (i=0;j<L.length;i++)
        if (!(L.data[i]>=x && L.data[i]<=y))
        {
            L.data[k]=L.data[i];
            k++;
        }
    L.length=k;                //表长置为 k
}
```

7. 设 A 和 B 均为下三角矩阵, 每一个都有 n 行 n 列。因此在下三角区域中各有  $n(n+1)/2$  个元素。另设有一个二维数组 C, 它有 n 行 n+1 列。试设计一个方案, 将两个矩阵 A 和 B 中的下三角区域元素存放于同一个 C 中。要求将的下三角区域中的元素存放于 C 的下三角区域中, B 的下三角区域中的元素转置后存放于 C 的上三角区域中。并给出计算 A 的矩阵元素  $a_{ij}$  和 B 的矩阵元素  $b_{ij}$  在 C 中的存放位置下标的公式。

【答案】将 n 阶方阵的下三角 A 复制到 C, B 逆置后复制到 C。核心语句段如下:

```
for(i=0;i<n;i++)
    {for(j=0;j<=i;j++) C[i][j]=A[i][j];
      for(j=i+1;j<=n;j++) C[i][j]=B[j-1][i];
    }
```

A 的矩阵元素  $a_{ij}$  和 B 的矩阵元素  $b_{ij}$  在 C 中的存放位置下标的公式:

$$A[i][j]=C[i][j]; \quad \{0 \leq i < n, 0 \leq j \leq i\}$$

$$B[i][j]=C[j][i+1]; \quad \{0 \leq i < n, i \leq j \leq n\}$$

8. 完成下述用模板类表示文件记录的折半查找算法, 算法实现在线性表表示的文件 A 中查找检索码等于 key 值的记录, 若找到, 则返回其下标值, 否则返回-1。

```
template < class T >
int BinarySearch(T A[], int n, int key)
{
    int low, high, mid;
    low = 0;
    high = n - 1;
    ...
}
```

【答案】算法如下:

```
template < class T >
int BinarySearch(T A[], int n, int key)
{
    int low, high, mid;
    low = 0;
    high = n - 1;

    while (low <= high)
    {
        mid = (low + high) >> 1;

        if (key == A[mid].key)
        {
            return mid;
        }
        else if (key > A[mid].key)
        {
            low = mid + 1;
        }
        else
        {
            high = mid - 1;
        }
    }
    return -1;
}
```



2024 年中原工学院 813 专业基础(含数据结构、操作系统)考研题库[仿真+强化+冲刺]

中原工学院 813 专业基础(含数据结构、操作系统)之数据结构考研仿真五套模拟题

2024 年数据结构五套仿真模拟题及详细答案解析（一）

一、算法设计题

1. 给出一组关键字 30,19,26,48,59,11,53,10, 分别写出按下列各种排序进行排序时的变化过程:

- (1) 归并排序, 每归并一次写一个次序;
- (2) 快速排序, 每划分一次写一个次序;
- (3) 堆排序, 先建成一个堆, 然后每从堆顶取下一个元素后, 将堆调整一次。

【答案】(1) 归并排序

第一趟排序: 19,30,26,48,11,59,10,53

第二趟排序: 19,26,30,48,10,11,53,59

第三趟排序: 10,11,19,26,30,48,53,59

(2) 快速排序

第一次划分: 10,19,26,11,30,59,53,48(以 30 为枢值)

第二次划分: 10,19,26,11,30,59,53,48(以 10 为枢值的 10,19,26,11 序列)

第三次划分: 10,19,26,11,30,48,53,59(以 59 为枢值的 48,53,59 序列)

第四次划分: 10,11,19,26,30,48,53,59(以 19 为枢值的 19,26,11 序列)

第五次划分: 10,11,19,26,30,48,53,59(以 48 为枢值的 48,53 序列)

(3) 堆排序

初始堆: 59,48,53,30,19,11,26,10

第一次调整: 53,48,26,30,19,11,10,59

第二次调整: 48,30,26,10,19,11,53,59

第三次调整: 30,19,26,10,11,48,53,59

第四次调整: 26,19,11,10,30,48,53,59

第五次调整: 19,10,11,26,30,48,53,59

第六次调整: 11,10,19,26,30,48,53,59

第七次调整: 10,11,19,26,30,48,53,59

2. 写出和下列递归过程等价的非递归过程:

```
PROCEDURE test(VAR sum: integer);
```

```
  VAR a: integer;
```

```
  BEGIN
```

```
    read(a);
```

```
    IF a = 0 THEN sum := 1
```

```
    ELSE BEGIN
```

```
      test(sum);
```

```
      sum := sum * a;
```

```
    END
```

```
    write(sum);
```

```
  END;
```

【答案】PROCEDURE test;

```
VAR stack: array[1..maxsize] of integer;
```

```
  top, sum, a: integer;
```

```
  BEGIN
```

```
    top := 0;
```

```
    sum := 1;
```

```
    read(a);
```

```
    WHILE a < > 0 DO
```

```
      BEGIN
```

```

top: = top + 1;
stack[top] := a;
read(a);
END
write(sum);
WHILE top < > 0 DO
BEGIN
    sum := sum * stack[top];
    top := top - 1;
    write(sum)
END

```

END

3. 编程求解无向图 G 的所有连通分量。

**【答案】**若图 G 是一个连通分量，执行一次 dfs 或 bfs 可确定是否有未被访问的顶点，从而确定该图是否连通。当非连通时，对于每个顶点，若已被访问过，则该顶点落在图中已被求得的连通分量上。若未被访问过，则从该顶点出发遍历图，便可求得图的另一个连通分量。

**PROCEDURE** Comp(g:Graph);

{g 有 n 个顶点}

**VAR** visited:ARRAY[1..n] **OF** Boolean;

i:integer;

**BEGIN**

**FOR** i:=1 **TO** n **DO**

visited:=false; {置初值，表示均未访问}

**FOR** i:=1 **TO** n **DO**

**IF** not visited[i] **THEN**

**BEGIN**

dfs(i); {找一个连通分量，或 bfs 也可}

outputnewvertices; {输出所有新近访问过的顶点以及所有与之关联的边}

**END**

**END;**

4. 假设一个单循环链表，其结点含有三个域 pre、data 和 link。其中 data 为数据域；pre 为指针域，它的值为空指针(NIL)；link 为指针域，它指向后继结点。请设计算法将此表改成双向循环链表。

**【答案】**设 s 为指向该循环链表中某结点的指针，引进 p 和 q 作为中间结点的指针；指针 s 在搜索过程中保持不变，用于循环结束条件的判断；每执行一次就将指针 p 赋给 q 所指结点 pre 域，并修改指针 p 和 q，保持 p 在后，q 在前。

算法描述如下：

**PROCEDURE** DoubleLinks(s:pointer);

**VAR** p,q:pointer;

**BEGIN**

p:=s;

**REPEAT**

q:=p^.link;

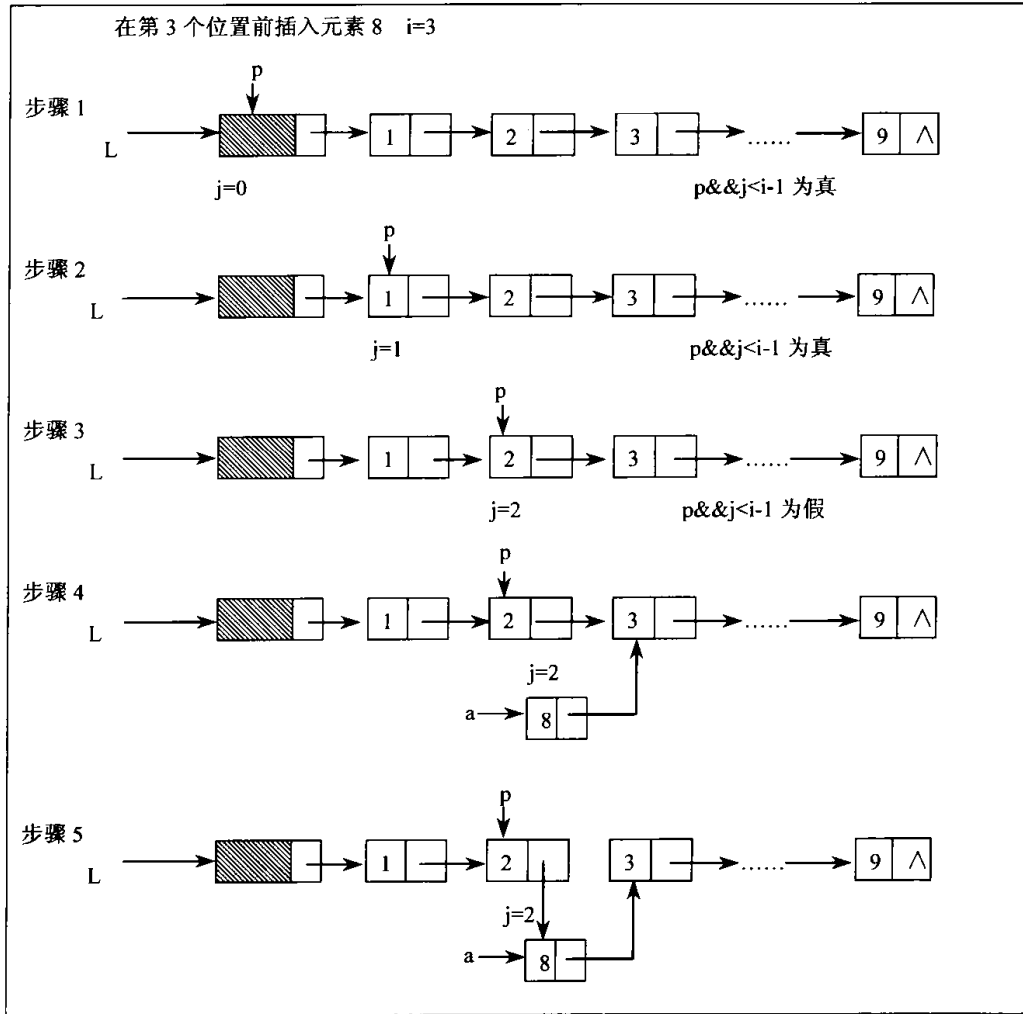
q^.pre:=p;

p:=q

**UNTIL** p:=s

**END;**

5. 根据下图的描述要求编写出该链表中插入一个元素的程序。



图：在链表中插入元素

**【答案】** `Status ListInsert_L(LinkList &L, int i, ElemType e)`

//在带头结点的单链表中第  $i$  个位置之前插入元素  $e$ 。  $i$  为 3,  $e$  为 8

```

{
    p=L; j=0;
    while(p->j < i-1)
    {
        p=p->next;
        ++j;
    }
    if(!p || j > i-1)
        return ERROR;
    a=(LinkedList)malloc(sizeof(LNode));
    a->data=e;
    a->next=p->next;
    p->next=a;
    return OK;
} //ListInsert_L
    
```

6. 请利用两个栈 S1 和 S2 来模拟一个队列。已知栈的三个运算定义如下: PUSH(ST, x): 元素 x 入 ST 栈; POP(ST, x): ST 栈顶元素出栈, 赋给变量 x; Empty(ST): 判 ST 栈是否为空。那么如何利用栈的运算来实现该队列的三个运算: enqueue: 插入一个元素入队列; dequeue: 删除一个元素出队列; queue\_empty: 判队列为空。

**【答案】**用两个栈 s1 和 s2 模拟一个队列时, s1 作输入栈, 逐个元素压栈, 以此模拟队列元素的入队。当需要出队时, 将栈 s1 退栈并逐个压入栈 s2 中, s1 中最先入栈的元素在 s2 中处于栈顶。s2 退栈, 相当于队列的出队。

(1) 插入一个元素入队列的核心语句段如下:

```
if(top1==n && !Empty(s2)) //top1 是栈 s1 的栈顶指针, 是全局变量
    {cout<<"栈满"<<endl;return(0);} //s1 满 s2 非空, 这对 s1 不能再入栈
if(top1==n && Empty(s2)) //若 s2 为空, 先将 s1 退栈, 元素再压栈到 s2
    {while(!Empty(s1)) {POP(s1,x);PUSH(s2,x);}
    PUSH(s1,x); return(1); //x 入栈, 实现了队列元素的入队
```

(2) 删除一个元素出队列的核心语句段如下:

```
if(!Empty(s2)) //栈 s2 不空, 则直接出队
    {POP(s2,x); cout<<"出队元素为"<<x<<endl; }
else //处理 s2 空栈
    if(Empty(s1)) {cout<<"队列空"<<endl;exit(0);} //判定队空
    else //先将栈 s1 倒入 s2 中, 再作出队操作
        {while(!Empty(s1)) {POP(s1,x);PUSH(s2,x);}
        POP(s2,x); //s2 退栈相当于队列出队
        cout<<"出队元素为"<<x<<endl;
        }
```

(3) 判队列为空的核心语句段如下:

```
if(Empty(s1) && Empty(s2)) return(1); //队列空
else return(0); //队列不空
```

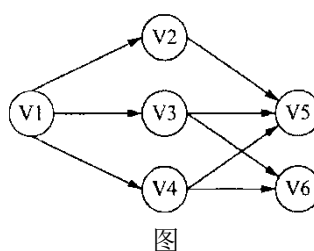
7. 已知非空线性链表第一个结点由 list 指出, 请写一算法, 交换 p 所指的结点与其下一个结点在链表中的位置(设 p 指向的不是链表最后那个结点)。

**【答案】**实现函数如下:

```
exchange(Linklist list, Node * p)
{
    q = list;
    while (q->next != p) q = q->next;
    if ! (p->next)
    {
        r = p->next;
        q->next = r;
        p->next = r->next;
        r->next = p;
    }
}
```

8. 某有向图如图所示, 要求:

- (1) 给出拓扑排序算法代码描述;
- (2) 给出图的算法执行结果。



附赠重点名校：数据结构 2016-2022 年考研真题汇编（暂无答案）

第一篇、2022 年数据结构考研真题汇编

2022 年沈阳工业大学数据结构考研专业课真题

沈阳工业大学

2022 年硕士研究生招生考试题签

（请考生将题答在答题册上，答在题签上无效）

科目名称：数据结构

第 1 页 共 2 页

一、解释下列名词(共 20 分，每小题 4 分)

1. 队列      2. 二叉树      3. 平均查找长度      4. 时间复杂度      5. 有向完全图

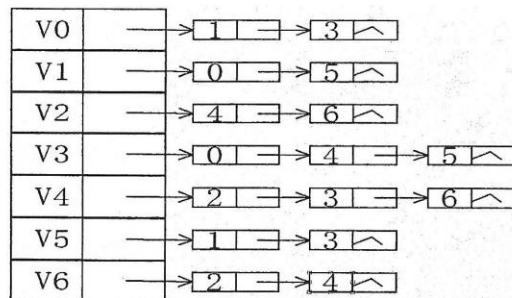
二、填空(共 30 分，每空 3 分)

- 数据的存储结构包括：顺序存储、\_\_\_\_\_、索引存储和散列存储。树形结构中的数据元素之间存在\_\_\_\_\_的关系。
- 在栈中存取数据应遵循的原则是\_\_\_\_\_。解决顺序队列“假溢出”的方法是采用\_\_\_\_\_。
- 二维数组  $A[0..3][0..4]$  的首地址是 1000，每个数据元素长度是 6，则  $A[2][3]$  的地址是\_\_\_\_\_。广义表  $((c, d), c, d)$  的表尾是\_\_\_\_\_。
- 具有  $n$  个结点的完全二叉树的深度是\_\_\_\_\_。如果一棵二叉树度为 2 的结点个数为 5，度为 1 的结点个数为 3，那么叶子结点的个数为\_\_\_\_\_。
- 有向图  $G$  是由  $(V, VR)$  组成，其中顶点的集合  $V = \{A, B, C, D, E\}$ ，弧的集合  $VR = \{\langle A, B \rangle, \langle A, C \rangle, \langle C, B \rangle, \langle C, D \rangle, \langle B, D \rangle, \langle E, A \rangle, \langle E, B \rangle, \langle E, C \rangle\}$ ，则该图的拓扑排序序列为\_\_\_\_\_，顶点  $D$  的入度为\_\_\_\_\_。

三、解答下列问题(共 50 分，每题 10 分)

- 设一棵二叉树结点的先序遍历序列是: ABFCGDEH，中序遍历序列是: FBGCADHE，要求：
  - 画出这棵二叉树，并写出这棵二叉树的后序遍历序列的结果。
  - 画出该二叉树对应的森林。
  - 画出该二叉树的后序线索二叉树。
- 设一个关键字序列为 {25, 32, 48, 16, 21, 37, 9, 12}，按要求完成以下操作：
  - 画出其对应的二叉排序树，并求在等概率情况下查找成功的平均查找长度 (ASL)。
  - 如果哈希表的长度为 12，哈希函数为  $H(\text{key}) = \text{key} \text{ MOD } 9$ ，用线性探测再散列方法创建哈希表，并求在等概率情况下查找成功的平均查找长度 (ASL)。
- 设一组权值序列为 (8, 11, 5, 17, 7, 29)，按要求完成以下操作：
  - 根据权值构建哈夫曼树。
  - 写出每个权值对应的哈夫曼编码。
  - 求带权路径的长度 (WPL)。

4. 已知一个无向图的存储结构如下图所示：



按要求完成以下操作：

- (1) 画出对应的无向图。
  - (2) 写出无向图对应的邻接矩阵。
  - (3) 写出从顶点 V0 出发深度优先和广度优先遍历的序列。
5. 已知一组关键字的序列为 {23, 48, 17, 9, 27, 3, 20, 69, 55}，请写出采用归并排序法对该序列作升序排序时每一趟的结果。

#### 四、编程题，程序设计语言不限(除 C、C++ 外请标明是哪种语言) (共 50 分，每题 10 分)

1. 从键盘输入 10 个整数存储在一个顺序表中，再输入一个整数赋值给变量 x，然后将这个顺序表划分成两部分，其中左半部分的每个整数均小于 x，右半部分的每个整数均大于等于 x。
2. 设有两个集合 A 和 B 分别存储在两个单链表中，设计求 A 和 B 的交集 C 的程序，集合 C 也存储在一个单链表中。(集合元素类型自行定义)
3. 从键盘输入二叉树数据，采用二叉链表建立二叉树，并写出对该二叉树前序遍历的递归程序。(注意：程序前面要用文字说明键盘输入二叉树数据的格式)
4. 从键盘输入一个有向图，建立一个邻接矩阵。再输入一个顶点编号 num，输出编号为 num 的顶点的度(入度和出度之和)。
5. 从键盘输入 10 个整数到一个一维整型数组，然后采用简单选择排序法进行排序并输出。

2022 年广东财经大学 809 数据结构考研专业课真题

广东财经大学硕士研究生入学考试试卷

考试年度：2022 年      考试科目代码及名称：809-数据结构（自命题）  
适用专业：085400 电子信息

[友情提醒：请在考点提供的专用答题纸上答题，答在本卷或草稿纸上无效!]

一、单项选择题（10 题，每题 2 分，共 20 分）

- 算法的时间复杂度取决于（ ）。  
A. 问题的规模      B. 待处理数据的初态      C. 计算机的配置      D. A 和 B
- 线性表的顺序存储结构中，数据元素的逻辑位置和物理位置的关系是（ ）。  
A. 不一致的      B. 一致的      C. 大致相同      D. 个别元素相同
- 在一个有  $n$  个元素的顺序表中，插入一个元素平均要移动的元素个数为（ ）。  
A.  $(n-1)/2$       B.  $n/2$       C.  $(n+1)/2$       D.  $n$
- 若顺序栈  $S$  存储在数组  $stack[MAXSIZE]$  中，栈顶位置  $top$  初值为  $-1$ ，则元素  $e$  进栈的操作是（ ）。  
A.  $S.stack[S.top++] = e;$       B.  $S.stack[++S.top] = e;$   
C.  $S.stack[S.top--] = e;$       D.  $S.stack[--S.top] = e;$
- 链队列  $Q$  的结点结构为： $(data, link)$ ，指针  $front$  指向队首元素， $rear$  指向队尾元素，则出队元素到变量  $x$  中的操作（ ）。  
A.  $x=Q.front->data; Q.front=Q.front->link;$   
B.  $Q.front=Q.front->link; x=Q.front->link;$   
C.  $x=Q.rear->data; Q.rear=Q.rear->link;$   
D.  $x=Q.rear->data; Q.rear=Q.front;$
- 一个递归算法必须包括（ ）。  
A. 递归部分      B. 终止条件和递归部分      C. 迭代部分      D. 终止条件和迭代部分
- 一棵非空二叉树的先序遍历序列和中序遍历序列相同，则该二叉树一定满足（ ）。  
A. 所有的结点均无左孩子      B. 所有的结点均无右孩子  
C. 只有一个叶子结点      D. 不存在这样的二叉树
- 按照满二叉树的编号顺序对深度为  $k$  的完全二叉树编号，则编号最小的叶结点的编号是（ ）。  
A.  $2^{k-1}-1$       B.  $2^{k-1}$       C.  $2^{k-2}+1$       D.  $2^k-1$
- 一棵完全二叉树的第 7 层有 24 个叶子结点，则整个二叉树的结点数至多为（ ）个  
A. 87      B. 206      C. 207      D. 231
- $G$  是一个非连通无向图，共有 36 条边，则该图至少有（ ）个顶点  
A. 7      B. 8      C. 9      D. 10

二、名词解释（10 题，每题 3 分，共 30 分）

- 队列
- 算法的空间复杂度
- 深度优先搜索
- 最小生成树
- 有向无环图
- 关键路径

- 7、归并排序
- 8、平衡树
- 9、邻接矩阵
- 10、基数排序

### 三、简答题（6题，每题10分，共60分）

1、二叉树和多叉树的转换。

- (1) 请将如下图-1 的多叉树转变为二叉树；
- (2) 将图-2 的二叉树转变为多叉树或森林。

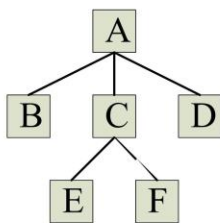


图-1

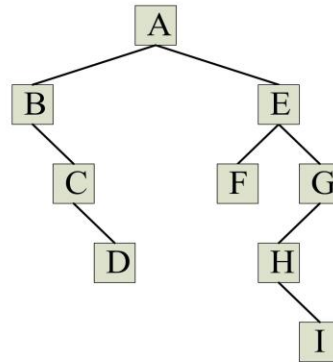


图-2

2、对序列 {81, 94, 11, 96, 12, 35, 17, 95, 28, 58, 41, 75, 15} 中的关键字按升序排序，采用希尔排序算法，增量序列分别为 5, 3, 1。

- (1) 请给出第一趟排序的结果；
- (2) 请给出第二趟排序的结果。

3、将关键字序列 (7、8、30、11、18、9、14) 散列存储到散列表中，散列表的存储空间是一个大小为 10，下标从 0 开始的一维数组，散列函数为： $H(key)=(key \times 3) \text{MOD } 7$ ，处理冲突采用线性探测法。

- (1) 请画出所构造的散列表。
- (2) 计算等概率情况下，查找成功的平均查找长度。

4、对于图-3 的 AVL 树（平衡树），依次插入关键字 35, 28。

- (1) 请画出插入关键字 35 后 AVL 树的形状。
- (2) 请画出随后再插入关键字 28 后 AVL 树的形状。

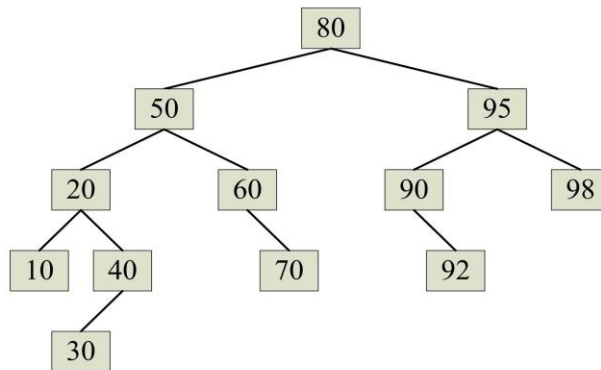


图-3

5、给定图 G 的邻接表如图-4 所示。



以上为本书摘选部分页面仅供预览，如需购买全文请联系卖家。

全国统一零售价： **¥ 184.00元**

卖家联系方式：

微信扫码加卖家好友：

