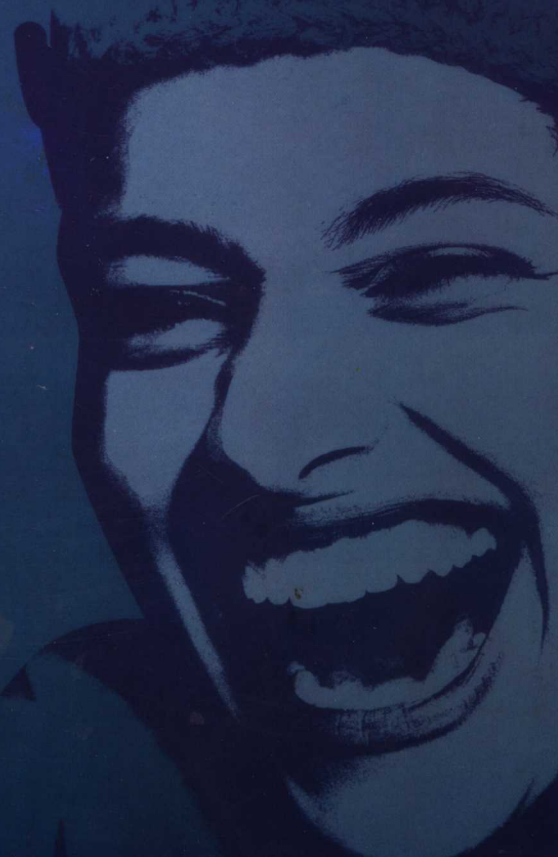


计算机专业研究生入学考试全真题解 (1)

数据结构 考研辅导教程

- ◎ 50多所重点院校真题
- ◎ 全真模拟题库
- ◎ 名师讲解
- ◎ 紧扣大纲
- ◎ 内容全面
- ◎ 通过机试

北京希望电子出版社 总策划
郝文化 主 编



电子科技大学出版社



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

计算机专业研究生入学考试全真题解 (1)

数据结构 考研辅导教程

◎ 50多所重点院校真题

◎ 全真模拟题库

◎ 名师讲解

◎ 紧扣大纲

◎ 内容全面

◎ 通过机试

北京希望电子出版社 总策划
郝文化 主 编



电子科技大学出版社



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内容简介

本书由长期从事数据结构课程教学工作的资深教师编写,书中汇集了近年来研究生入学考试常见考试知识点解析,提供了丰富的典型题例与实战练习,十分难得的是本书包含了国内 50 多所重点大学的数据结构研究生入学试题,此外还提供了 6 套全真模拟试题及参考答案。本书是帮助读者在最短时间内成功考研的“利器”,是考生顺利通过考试强有力的保障。本书从实际考试的角度出发,对大量重点、难点问题进行了详细解析,书中所有题例、名校试题及练习都配有解答,特别指出的是书中的算法题都在 Visual C++ 6.0 中通过,能快速提高考生的应试能力及软件开发的能力。

本书重点突出、系统全面,注重解题思路与上机实战训练,十分适合报考计算机专业研究生的考生进行考前冲刺复习,使广大考生更能从书中得到正规而简洁的学习方法与经验,同时也是高校计算机及相关专业学生难得的学习参考书。本书配套光盘为书中部分算法程序源代码。

需要本书或技术支持的读者,请与北京清河 6 号信箱(邮编:100085)发行部联系。电话:010-82702660, 62978181(总机)转 103 或 238, 传真:010-82702698, E-mail: tbd@bhp.com.cn。

图书在版编目(CIP)数据

数据结构考研辅导教程/郝文化主编. —成都:电子科技大学出版社, 2005.9

ISBN 7-81094-851-2

I. 数... II. 郝... III. 数据结构—研究生—入学考试—自学参考资料 IV. TP311.12

中国版本图书馆 CIP 数据核字(2005)第 064571 号

数据结构考研辅导教程

郝文化 主编

-
- 出 版: 电子科技大学出版社(成都建设北路二段四号 610054)
北京希望电子出版社(北京海淀区上地信息产业基地 3 街 9 号
金隅嘉华大厦 C 座 611 100085)
网址: www.bhp.com.cn E-mail: lwm@bhp.com.cn zmh@bhp.com.cn
电话: 010-82702660, 82702658, 62978181 转 103 或 238, 传真: 010-82702698
- 责任编辑: 杜倩 王新文
- 发 行: 新华书店经销
- 印 刷: 北京媛明印刷厂
- 开 本: 787mm×1092mm 1/16 印张 24 字数 558 千字
- 版 次: 2005 年 9 月第 1 版
- 印 次: 2005 年 9 月第 1 次印刷
- 书 号: ISBN 7-81094-851-2/TP.452
- 印 数: 0 001-4 000 册
- 定 价: 34.00 元(配 1 张光盘)
-

前 言

《数据结构》课程是计算机专业的核心基础课，是一门研究在非数值计算程序设计问题中的操作对象及它们之间的关系与操作以及相关算法的学科，不仅涉及到硬件，而且与计算机软件的理论研究有着非常密切的关系。学生在初学《数据结构》时一般都会感到比较抽象和难于掌握，所以在考研时需要认真复习。

为了使广大考生能顺利地通过研究生入学考试，我们结合多年从事《数据结构》教学工作的实践经验，通过对全国数十所重点大学上百套《数据结构》研究生入学试题的典型试题的透彻分析，研究了试题的解答方法、考试技巧以及考生的成功体会，在归纳、总结、取其精华、找出规律的基础上精心编写了本书。

本书内容包含数据结构基础知识、线性表、栈、队列、串、数据组与广义表、树、图、查找、排序、文件与动态存储管理，并且有典型题例精讲，名校试题解析，最后精心组织的6套全真模拟试题并附参考答案。同时，本书附有一张配套光盘。在光盘中包括有本书算法的运算程序，对于本光盘的使用请参见光盘中《readme》和《算法程序使用说明》文件。

本书列出了每章的重点、常见考试知识点分析、典型题例与名校试题解析及实战练习，其中的名校试题是50多所重点大学历年考研试题中精选而出，包含了各重点大学考研试题各经典题型；书中所有算法都在 Visual C++ 6.0 中通过（在本书配套光盘中有本书每个算法的运算程序），使每个算法题都是读者的一个上机实践机会；快速强化训练，使考生在最短的时间内能掌握必备的考试知识，学会做各类试题。

本书通俗易懂、深浅得当、内容全面，适合于准备参加研究生入学考试的人员作为复习参考书，也可作为各类高校计算机及相关专业学生的学习辅导资料。

本书由郝文化主编，由游洪跃、踞生根、陈良银、于中华、陈黎、陈伟鹏等老师担任主要的编写工作。同时，参与本书编排的还有：邹素琼、王安贵、陈郭宜、程小英、谭小丽、卢丽娟、刘育志、吴淬砺、赵明星、贺洪俊、李小平、史利、张燕秋、周林英、黄茂英、李力、李小琼、李修华、田茂敏、苏萍、巫文斌、邹勤、栗德容、童芳、李中全、蒋敏、刘华菊、袁媛、李建康、袁涛、杨春华、贾小蓉、张永龙等，在此表示衷心感谢。

为充分展现本书编写特点，帮助考生深刻理解本书编写意图与内涵，进一步提高对本书学习的使用效率，我们建立了本书使用指导联络方式，这是考生与编者之间交流沟通的桥梁。欢迎读者将图书使用过程中的问题与各种探讨、建议反馈给我们，本书编者竭诚给您满意的答复。我们的联系方式是：bojia@bojia.net。

此外，我们还为本书读者提供周到、完善的配套服务，为考生顺利通过考试提供贴心、有益的帮助。该服务请登录网址：<http://www.bojia.net>。

编 者

目 录

第1章 数据结构基础知识.....1	3.1.3 串.....56
1.1 常见考试知识点分析.....1	3.2 典型题例与名校试题解析.....58
1.1.1 基本内容及相关知识.....1	3.2.1 单项选择题.....58
1.1.2 算法和算法分析.....2	3.2.2 综合题.....64
1.2 典型题例与名校试题解析.....5	3.3 实战练习.....87
1.2.1 单项选择题.....5	3.3.1 单项选择题.....87
1.2.2 综合题.....6	3.3.2 综合题.....88
1.3 实战练习.....10	3.4 实战练习参考答案.....89
1.3.1 单项选择题.....10	3.4.1 单项选择题答案.....89
1.3.2 综合题.....10	3.4.2 综合题答案.....89
1.4 实战练习参考答案.....12	第4章 数据组与广义表.....102
1.4.1 单项选择题答案.....12	4.1 常见考试知识点分析.....102
1.4.2 综合题答案.....12	4.1.1 数组及其操作.....102
第2章 线性表.....15	4.1.2 数组的顺序存储结构.....102
2.1 常见考试知识点分析.....15	4.1.3 特殊矩阵.....103
2.1.1 线性表的逻辑结构.....15	4.1.4 稀疏矩阵及其三元组表示.....103
2.1.2 线性表的顺序存储结构.....16	4.1.5 稀疏矩阵的行逻辑链接表示.....104
2.1.3 线性表的链式存储结构.....17	4.1.6 稀疏矩阵的十字链表示.....104
2.1.4 循环链表.....17	4.1.7 广义表概念.....105
2.1.5 双向链表.....18	4.1.8 广义表的存储结构.....105
2.1.6 一元多项式的链式存储结构.....18	4.2 典型题例与名校试题解析.....106
2.2 典型题例与名校试题解析.....19	4.2.1 单项选择题.....106
2.2.1 单项选择题.....19	4.2.2 综合题.....109
2.2.2 综合题.....22	4.3 实战练习.....126
2.3 实战练习.....37	4.3.1 单项选择题.....126
2.3.1 单项选择题.....37	4.3.2 综合题.....128
2.3.2 综合题.....38	4.4 实战练习参考答案.....129
2.4 实战练习参考答案.....40	4.4.1 单项选择题答案.....129
2.4.1 单项选择题答案.....40	4.4.2 综合题答案.....129
2.4.2 综合题答案.....40	第5章 树.....139
第3章 栈、队列与串.....51	5.1 常见考试知识点分析.....139
3.1 常见考试知识点分析.....51	5.1.1 树的基本概念及基本操作.....139
3.1.1 栈.....51	5.1.2 二叉树.....140
3.1.2 队列.....54	5.1.3 遍历二叉树与线索二叉树.....142

5.1.4	树的存储结构、森林与二叉树的转换及树的遍历	144
5.1.5	树的计数	146
5.1.6	哈夫曼树的构造	147
5.2	典型题例与名校试题解析	147
5.2.1	单项选择题	147
5.2.2	综合题	153
5.3	实战练习	183
5.3.1	单项选择题	183
5.3.2	综合题	185
5.4	实战练习参考答案	187
5.4.1	单项选择题答案	187
5.4.2	综合题答案	188
第6章	图	208
6.1	常见考试知识点分析	208
6.1.1	基本概念及术语	208
6.1.2	图的基本操作	209
6.1.3	图的存储结构	210
6.1.4	图的遍历	213
6.1.5	最小生成树	213
6.1.6	有向无环图及其应用	214
6.1.7	欧拉问题	215
6.2	典型题例与名校试题解析	215
6.2.1	单项选择题	215
6.2.2	综合题	218
6.3	实战练习	236
6.3.1	单项选择题	236
6.3.2	综合题	237
6.4	实战练习参考答案	240
6.4.1	单项选择题答案	240
6.4.2	综合题答案	240
第7章	查找	251
7.1	常见考试知识点分析	251
7.1.1	基本概念及术语	251
7.1.2	顺序查找	252
7.1.3	折半查找	252
7.1.4	分块查找	253

7.1.5	二叉排序树	254
7.1.6	平衡二叉树	254
7.1.7	B树及B+树	255
7.1.8	键树	256
7.1.9	Hash表	257
7.2	典型题例与名校试题解析	260
7.2.1	单项选择题	260
7.2.2	综合题	262
7.3	实战练习	277
7.3.1	单项选择题	277
7.3.2	综合题	278
7.4	实战练习参考答案	279
7.4.1	单项选择题答案	279
7.4.2	综合题答案	279
第8章	排序	289
8.1	常见考试知识点分析	289
8.1.1	内部排序	289
8.1.2	外部排序	293
8.2	典型题例与名校试题解析	295
8.2.1	单项选择题	295
8.2.2	综合题	300
8.3	实战练习	315
8.3.1	单项选择题	315
8.3.2	综合题	317
8.4	实战练习参考答案	319
8.4.1	单项选择题答案	319
8.4.2	综合题答案	320
第9章	文件与动态存储管理	328
9.1	常见考试知识点分析	328
9.1.1	文件	328
9.1.2	动态存储管理	330
9.2	典型题例与名校试题解析	332
9.2.1	单项选择题	332
9.2.2	综合题	333
9.3	实战练习	337
9.3.1	单项选择题	337
9.3.2	综合题	338

9.4 实战练习参考答案.....	338	10.5 模拟试题（三）.....	351
9.4.1 单项选择题答案.....	338	10.6 模拟试题（三）参考答案.....	354
9.4.2 综合题答案.....	338	10.7 模拟试题（四）.....	358
第10章 模拟试题.....	340	10.8 模拟试题（四）参考答案.....	360
10.1 模拟试题（一）.....	340	10.9 模拟试题（五）.....	363
10.2 模拟试题（一）参考答案.....	342	10.10 模拟试题（五）参考答案.....	366
10.3 模拟试题（二）.....	345	10.11 模拟试题（六）.....	370
10.4 模拟试题（二）参考答案.....	348	10.12 模拟试题（六）参考答案.....	372

第 1 章 数据结构基础知识

本章重点:

数据结构的定义、算法的特性及算法的时间复杂度。

1.1 常见考试知识点分析

1.1.1 基本内容及相关知识

1. 数据结构基础

《数据结构》是计算机专业的核心课程，数据结构是一门研究程序设计问题中的操作对象，以及它们之间的关系和操作的学科。学习此门课程时，应掌握各种主要数据结构的逻辑结构及存储结构，以及处理数据的各种算法，对于算法的时间复杂度与空间复杂度应有一定的了解。对于《数据结构》的学习，自学者应提高对程序设计的能力，建立良好的程序设计规范，初步具备评价程序的能力。

2. 数据结构相关的基本概念与术语

(1) 数据 数据在哲学意义上讲是对客观事物的符号表示，在计算机学科中表示一切能输入到计算机中并能被计算机处理的符号的总称。

(2) 数据元素 数据元素是数据的基本单位，在计算机程序中一般将数据元素作为一个整体进行处理，有时一个数据元素可由若干数据项所组成，如学生基本信息可作为一个数据元素，而学生基本信息的每一项可由学号、姓名、性别、生日…等内容所组成，数据项是不可分割的最小单位。

(3) 数据对象 数据对象是指性质相同的数据元素的集合，如偶数数据对象为 $\{0, \pm 2, \pm 4, \pm 6, \dots\}$ ，英文小写字母数据对象为 $\{'a', 'b', \dots, 'z'\}$ 。

(4) 数据结构 数据结构相互之间存在一种或多种特定关系的数据元素的集合，根据数据元素之间关系的特性，有如下 4 类基本结构：

- 集合结构：结构中数据元素之间存在“同属于一个集合”的关系，如图 1-1 所示。
- 线性结构：结构中的数据元素之间存在一个对一个的关系，如图 1-2 所示。

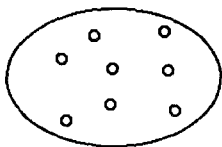


图 1-1 集合结构

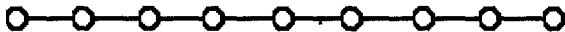


图 1-2 线性结构

- 树形结构：结构中的数据元素之间存在一个对多个的关系，如图 1-3 所示。
- 图状结构：结构中的数据元素之间存在多个对多个的关系，如图 1-4 所示。

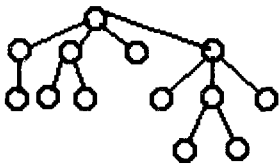


图 1-3 树形结构

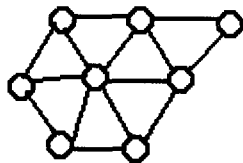


图 1-4 图状结构

(5) 数据结构形式定义 数据结构是一个如下的二元组： $\text{Data_Structure}=(D,S)$ ，其中 D 是一个数据元素的有限集合， S 为定义在 D 上的有限个关系的集合。

(6) 逻辑结构 逻辑结构是指从操作对象抽象出的数学模型，其结构定义中的关系用于描述数据元素之间的逻辑关系。

(7) 物理结构（存储结构） 数据结构在计算机中的表示称为物理结构或存储结构，根据数据元素在计算机中的表示方法，可分为顺序存储结构和链式存储结构。

(8) 数据类型 数据类型是一个值的集合与定义在此值集合上的一些操作的总称，按值的不同特性，在高级语言中的数据类型可分为原子类型和结构类型。

(9) 抽象数据类型 抽象数据类型指一个数学模型及定义在此模型上的一组操作的总称，按值的不同特性，可分为如下 3 类：

- 原子类型：此类型的变量不可分解；
- 固定聚合类型：此类型的变量的值由确定数目的成分按一定的结构所组成；
- 可变聚合类型：此类型的变量的值的成分的数目不确定。

(10) 多形数据类型 多形数据类型是指值的成分不确定的数据类型，数据结构中的各种数据类型一般都属于多形数据类型，为不增加难度，在本书的讨论中数据类型都不写成多形数据类型的形式。

(11) 结构中所涉及的主要操作 在结构中有各种类型的操作，其中主要的操作有如下几种：

- 查找：查寻满足特定要求的数据元素的位置或值；
- 排序：重新排列数据元素之间的逻辑顺序，使其按一定的逻辑关系排序，如按关键字的值从小到大或从大到小的顺序排列；
- 插入：在特定位置上插入一个数据元素；
- 删除：删去指定位置上的数据元素；
- 更新：修改某数据元素的值。

按操作时是否改变数据结构的值，各种操作可分为如下两类：

- 加工型操作：在操作时要改变数据结构的值；
- 引用型操作：在操作时不改变数据结构的值。

1.1.2 算法和算法分析

1. 算法的重要特征及要求

(1) 算法及重要特性。算法是对特定问题的求解步骤的描述，在计算机中为指令的有限序列，并且每条指令表示为一个或多个操作。算法有如下 5 个重要特性：

- 有穷性: 任何算法必须在执行有穷步后结束, 并且每一步都在有限时间内完成;
- 确定性: 算法中的所有操作都有确切的含义, 不能产生歧义;
- 可行性: 算法中的所有操作都必须经过已实现的基本操作有限次来实现;
- 输入: 任何算法有零个或多个输入数据;
- 输出: 任何算法有一个或多个输入数据。

(2) 算法设计的要求 算法设计时一般应达到如下要求:

- 正确性: 指算法应满足特定问题的要求;
- 可读性: 可读性良好的算法有利于算法的查错及对算法的理解;
- 健壮性: 指当输入数据非法时, 算法能作适当的处理并作出反应, 而不应死机或输出异常结果;
- 效率与低存储量的平衡: 效率指算法运行速度的快慢, 当然效率越快越好, 存储量越低越好, 通常算法效率与存储量成正比, 算法应正确处理好效率与低存储量的关系。

2. 算法描述

算法描述可用自然语言描述, 但自然语言容易产生歧义, 在数据结构中不太适用, 一般用伪码语言、框图或高级语言描述, 当前一般都采用 C 语言, 但为了方便起见, 也使用某些 C++ 的语法。本书中的名校试题所涉及的算法部分已统一成本书格式进行描述, 下面对本书中用到的 C++ 部分加以说明:

(1) 输入/输出语句 本书所用到的 C 语言与 C++ 语言中的输入/输出语句如表 1-1 所示。

表 1-1 C 语言与 C++ 语言中输入/输出语句一览表

对照内容	C 语言	C++ 语言
输入	<code>scanf("%d",&i);</code>	<code>cin>>i;</code>
输出	<code>printf("%d",i);</code>	<code>cout<<i;</code>
头文件	<code>stdio.h</code>	<code>iostream.h</code>
换行符	<code>'\n'</code>	<code>endl</code>

(2) 关于要返回结果值的函数参数的处理方式 在 C 语言中如某函数参数要返回结果值, 只能使用指针实现; 在 C++ 中可简单地使用引用 (&) 即可实现, 例如对于交换两个参数值的函数。

C 语言函数:

```
void exchange(int *u,int *u)
```

```
{  
    int *tem;  
    *tem=*u;  
    *u=*v;  
    *v=*tem;  
}
```

C++语言函数:

```
void exchange(int &u,int &u)
```

```
{  
    int tem;  
    tem=u;  
    u=v;  
    v=tem;  
}
```

注意: 凡是出现了 C++语法成分的源程序的扩展名必须为.cpp。

3. 算法分析

对于一个算法的评价, 首先应考虑算法的正确性, 其次是运算量(即运行效率的高低), 有时还要考虑算法所占的存储空间的大小, 为定性分析引入时间复杂度与空间复杂度的概念, 重点考虑时间复杂度。

(1) 算法效率的度量 度量算法的运行时间一般有如下两种方法:

- 事后统计的方法;
- 事前分析估计的方法。

对于用高级语言编写的程序, 通常由控制结构(包括顺序结构、判断结构及循环结构)与原操作所构成, 则算法运行时间取决于两者的综合效果。为便于比较, 一般是从算法中选取对于所研究问题最基本运算的原操作, 用此基本操作重复执行的次数作为算法的时间度量。

(2) 时间复杂度 算法中最基本运算的原操作运算次数通常为问题规模 n 的某个函数 $f(n)$, 则算法时间度量为: $T(n)=O(f(n))$ 。

$O(f(n))$ 称为算法的渐近时间复杂度, 或简称为时间复杂度。一般常见的的时间复杂度有:

- $O(1)$: 常数阶时间复杂度, 此种时间复杂度的算法运行时间效率最高;
- $O(n)$ 、 $O(n^2)$ 、 $O(n^3)$ 、...: 多项式阶时间复杂度, 大部分算法的时间复杂度为多项式阶时间复杂度。

提示: $O(2^n)$ 指数阶时间复杂度, 它的运行效率最低, 这种复杂度的算法根本不实用。

- $O(n\log_n)$ 和 $O(\log_n)$: 对数阶时间复杂度, 此种时间复杂度除常数阶时间复杂度以外, 它的效率最高。

(3) 空间复杂度 算法中所需存储空间通常为问题规模 n 的某个函数 $f(n)$, 则算法空间度量为: $S(n)=O(f(n))$ 。

$O(f(n))$ 称为算法的空间复杂度。

1.2 典型题例与名校试题解析

1.2.1 单项选择题

【例1】数据结构在计算机中的表示称为数据的 ()。

- A. 存储结构 B. 抽象结构 C. 顺序结构 D. 逻辑结构

【分析】根据数据的存储结构的定义可知。

【答案：A】

【例2】对于下面程序段的时间复杂度为 ()。

```
for(i=1;i<=n;i++)  
    for(j=1;j<=i;j++)  
        x=x+1;
```

- A. $O(n)$ B. $O(n^2)$ C. $O(n*i)$ D. $O(n+i)$

【分析】本题程序段中基本语句为： $x=x+1$;

当 $i=1$ 时, j 从 1 到 1, $x=x+1$ 的执行次数为 1;

当 $i=2$ 时, j 从 1 到 2, $x=x+1$ 的执行次数为 2;

.....

当 $i=n$ 时, j 从 1 到 n , $x=x+1$ 的执行次数为 n 。

可知 $x:=x+1$ 的总执行次数为： $1+2+\cdots+n=\frac{n(n+1)}{2}$, 所以时间复杂度为 $O(n^2)$ 。

【答案：B】

【例3】数据结构是 ()。(东部名校经典试题)

- A. 相互之间存在一种或多种特定关系的数据元素的集合
B. 相互之间存在一种特定关系的数据元素的集合
C. 数据元素的集合
D. 前面都不正确

【分析】根据数据结构的定义, 数据结构简单地讲是指相互之间存在一种或多种关系的数据元素的集合。

【答案：A】

【例4】数据结构可形式地定义为 (D,S) , 其中 S 是 D 上 () 的有限集。(东部名校经典试题)

- A. 操作 B. 存储映像 C. 关系 D. 数据元素

【分析】数据结构形式定义中, D 指 Data (数据); S 为 Structure (结构), 实际指关系的有限集。

【答案：C】

【例5】数据结构在计算机中存储器内表示时, 物理地址和逻辑地址相同并且是连续的, 称之为 ()。(北方名校经典试题)

- A. 逻辑结构 B. 顺序存储结构 C. 链式存储结构 D. 以上都对

【分析】在计算机中存储器内表示数据时分为两种,采用高级语言的数组类型进行存储分配的称为顺序存储结构,顺序存储结构的物理地址和逻辑地址相同并且是连续的;而采用动态链表进行存储分配的称为链式存储结构,而链式存储结构的逻辑地址连续但物理地址未必连续。

【答案: B】

1.2.2 综合题

【例 1】设有如图 1-5 所示的逻辑结构图,试给出数据结构形式。

【解答】数据结构形式定义为(D,S)

其中: $D=\{a,b,c,d,e\}$

$S=\{R\}$

$R=\{(a,b),(b,c),(c,d),(c,e),(d,e)\}$

【例 2】有如下数据结构的形式定义,试画出此结构的图形表示。(东部名校经典试题)

$DS=\{D,S\}$

其中,

$D=\{1,2,3,4\}$

$S=\{R\}$

$R=\{\langle 1,2\rangle,\langle 1,3\rangle,\langle 2,3\rangle,\langle 2,4\rangle,\langle 3,4\rangle\}$

【解答】用 D 中数据元素画出顶点,用 R 中有序对画出弧,如图 1-6 所示。

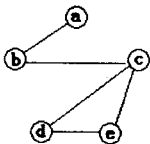


图 1-5 逻辑结构图

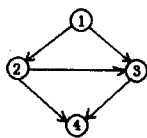


图 1-6 结构图

【例 3】(1)“数据结构”课程是计算机专业的基础课还是专业课,或者专业基础课?

(2)学习“数据结构”课程需要哪些课程作为它的基础(列举两门课程)?若没有这些知识,对学习“数据结构”课程可能会产生哪些影响?请举例说明(不超过 100 字)。

(3)“数据结构”课程将为哪些课程的学习奠定必要的基础?请举例说明哪些课程(列举两门课程)用到了“数据结构”课程的哪些知识(不超过 100 字)。(北方名校经典试题)

【解答】

(1)“数据结构”课程是计算机类专业的专业基础课。

(2)学习“数据结构”课程需一些课程作为它的基础,如“C 语言”与“离散数学”。如没有“C 语言”或其他语言基础(如 Java、C++),则学生难以理解描述数据结构及其算法的 C 或 Java 或 C++,更主要是使学生无法上机验证有关算法,影响对“数据结构”中算法的深刻理解;“离散数学”课程是“数据结构”课程的理论基础,其中树与图等知识为“数据结构”课程的学习提供了重要的理论基础。

(3)“数据结构”课程为“编译原理”、“数据库原理”和“操作系统原理”等课程的

学习奠定必要的基础知识。如“编译原理”的表达式求值将用到“数据结构”的栈知识，“数据库原理”的索引文件将用到“数据结构”的B+树知识；“操作系统原理”的设备链表管理将用到“数据结构”的线性链表知识。

【例4】选择某种问题的最佳数据结构的标准是什么？（西部名校经典试题）

【解答】通常的标准有两条：

- (1) 算法所需要的时间。
- (2) 算法所需要的存储空间。

【例5】下面程序段用于求两个 $n \times n$ 矩阵相乘的算法，试求其时间复杂度。

```
for(i=0;i<n;i++)
    for(j=0;j<n;j++){
        c[i][j]=0;
        for(k=0;k<n;k++)
            c[i][j]=c[i][j]+a[i][k]*b[k][j];
    }
```

【解答】本题中“乘法”运算为基本操作，其执行次数为 n^3 ，可知时间复杂度为 $O(n^3)$ 。

技巧：对于非递归算，如无循环语句，则时间复杂度为 $O(1)$ ；如只有一重循环语句，则时间复杂度为 $O(n)$ ；如最多只有二重循环语句，则时间复杂度为 $O(n^2)$ 等。

【例6】求下述最小函数渐近时间。

- (1) $T_1(n) = n \log n + n$
- (2) $T_2(n) = n^{\log n} - n$
- (3) $T_3(n) = n^3 - 100 \log n$

【解答】 $T_1(n) = O(n \log n)$, $T_2(n) = O(n^{\log n})$, $T_3(n) = O(n^3)$ ，当 n 足够大时， $T_1(n) < T_3(n) < T_2(n)$ ，所以渐近时间最小的是 $T_1(n)$ 。

【例7】试给出以下算法的时间复杂度。

```
void sort(int a[],int b[],int i,int n)
/* 对数组a进行升序排序，并将排序结果存入数组b中i为待排序序列下标
   的下界，n为待排序序列下标的上界
*/
{
    int j,k,tem;
    if(i==n)
        b[n]=a[n];           //语句1
    else
    {
        j=n;
        for(k=i;k<n;k++)      //语句2
            if(a[j]<a[k])
```

```

        j=k;
    if(j!=n)                //语句3
    {
        tem=a[j];
        a[j]=a[n];
        a[n]=tem;
    }
    b[n]=a[n];              //语句4
    sort(a,b,i,n-1);        //语句5
}
}

```

【解答】从算法中可以看出：当 $i=n$ 时，执行语句 1，时间为常量，用 $O(1)$ 表示；否则将执行 else 子句，其中语句 2 需执行 $n-i+1$ 次，时间用 $O(n-i+1)$ 表示；语句 3 执行 1 次，时间为常量，用 $O(1)$ 表示；语句 4 执行 1 次，时间为常量，用 $O(1)$ 表示；语句 5 是递归调用语句，用 $T(i,n)$ 表示 $\text{sort}(a,b,i,n)$ 的工作时间，语句 5 的工作时间就是 $T(i,n-1)$ ，这时可得如下关系：

$$T(i,n)=T(i,n-1)+(n-i+1)+2$$

用 $T(n)$ 表示 $T(0,n)$ 的工作时间，进一步可得如下的递归表达式：

$$T(n)=\begin{cases} 1 & \text{如 } n=0 \\ T(n-1)+n+3 & \text{如 } n>0 \end{cases}$$

所以可得：

$$\begin{aligned}
 T(n) &= T(n-1) + (n+3) \\
 &= T(n-2) + (n+2) + (n+3) \\
 &= T(n-3) + (n+1) + (n+2) + (n+3) \\
 &\dots \\
 &= T(0) + 4 + 5 + \dots + (n+1) + (n+2) + (n+3) \\
 &= 1 + 4 + 5 + \dots + (n+1) + (n+2) + (n+3) \\
 &= \frac{n(n+7)}{2} + 1 \\
 &= O(n^2)
 \end{aligned}$$

【例 8】试确定如下关于求 $n!$ 算法中带标号语句的执行频度以及算法的时间复杂度。

```

long fac(int n)
{
    if(n>1)                //语句1
        return n*fac(n-1); //语句2
    else
        return 1;          //语句3
}

```

【解答】本题算法为递归算法，递归条件为 $n > 1$ ，如表 1-2 所示列出 $n=1, 2, 3, \dots$ 时各语句的执行频度。

表 1-2 算法语句执行频度表

参数值	不考虑递归调用时各语句执行频度			递归调用情况
	语句1	语句2	语句3	...
1	1	0	1	无
2	1	1	0	fac(1)
3	1	1	0	fac(2)
...	...	...	...	...
n	1	1	0	fac(n-1)

对于 fac(2) 的各语句执行频度还应包括递归调用 fac(1) 时各语句的执行频度，fac(3) 各语句执行频度还应包括递归调用 fac(2) 时各语句的执行频度，……，可得各语句的执行频度如下：

语句 1 执行频度： $1+1+\dots+1=n$

语句 2 执行频度： $0+1+1+\dots+1=n-1$

语句 3 执行频度： $1+0+0+\dots+0=1$

算法的时间复杂度可基于频度最大的语句，可知应为 $O(n)$ 。

注意：对于求包含调用语句（包括递归调用）的算法的语句频度的问题，需考虑调用发生时被调用算法中各语句的执行情况。

【例 9】试编写一算法，实现将输入的 3 个整数 x, y, z 按从小到大的顺序排列起来。

【解答】依次从键盘输入 x, y 和 z 三个整数，然后通过比较交换后得到最大数 z ，再将 x, y 进行比较交换，使其从小到大排序，下面是参考程序：

```
void sort(){
    int x,y,z,tem;
    cin>>x>>y>>z;
    if(x>y){
        tem=x;
        x=y;
        y=tem;
    }
    if(z<y){
        tem=y;
        y=z;
        z=tem;
    }
    if(x>y){
        tem=x;
```

```

x=y;
y=tem;
}
cout<<x<<","<<y<<","<<z<<endl;
}

```

1.3 实战练习

1.3.1 单项选择题

- (1) 如一个结构中的数据元素之间存在一个对多个的关系,则此结构为()。
- A. 集合结构 B. 线性结构 C. 树形结构 D. 图状结构
- (2) 在数据类型中,值不可分解的类型为()。
- A. 原子类型 B. 结构类型 C. 固定聚合类型 D. 可变聚合类型
- (3) 在抽象数据类型中,值由确定数目的成分按某种结构组成的类型为()。
- A. 原子类型 B. 结构类型 C. 固定聚合类型 D. 可变聚合类型
- (4) 下面程序段的时间复杂度为()。

```

for(i=n;i>=1;i--)
    for(j=1;j<=i;j++)
        x=x+1;

```

- A. $O(n)$ B. $O(n^2)$ C. $O(n*i)$ D. $O(n+i)$
- (5) 数据类型为()。
- A. 数据项的集合 B. 值的集合及定义在其上的一组操作的总称
- C. 数据元素的集合 D. 关键字的集合
- (6) 网状结构的特征是()。
- A. 结构中数据元素之间只存在“同属于一个集合”的关系
- B. 结构中数据元素之间存在一个对应一个的关系
- C. 结构中数据元素之间存在一个对应多个的关系
- D. 结构中数据元素之间存在多个对应多个的关系
- (7) 设计一个“好”的算法应达到的目标为()。
- A. 正确性、可读性、健壮性及效率与低存储量需求
- B. 正确性、可读性、健壮性及有穷性
- C. 正确性、可读性、健壮性及可行性
- D. 正确性、可读性、健壮性及确定性

1.3.2 综合题

- (1) 设数据逻辑结构如下:

$DS=(D,S)$

$D=\{1,2,3,4,5,6,7,8\}$