



全国高等教育自学考试

计算机信息管理专业和计算机网络专业自学指导丛书

# 数据结构

## 自学考试指导

全国电子信息应用教育中心 组编

丁宝康 主编

陈金海 陈 坚 副主编

1. 12

7



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



全国高等教育自学考试计算机信息管理专业和计算机网络专业自学指导丛书

# 数据结构自学考试指导

全国电子信息应用教育中心 组编

丁宝康 主编

陈金海 陈 坚 副主编

清华大学出版社

512546

(京)新登字 158 号

## 内 容 简 介

本书是全国高等教育自学考试指定教材《数据结构》的辅导材料。

全书分两个部分。第一部分是教材中各章内容辅导,包括考核知识点的分析,教材中习题的分析与解答,练习题及解答,上机实习题;第二部分为三套自测题及解答。

本书供考生复习迎考使用,也可供从事《数据结构》课程教学的教师和从事软件工作的有关人员参阅。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

## 图书在版编目(CIP)数据

数据结构自学考试指导/丁宝康主编. - 北京:清华大学出版社,2001.5

(全国高等教育自学考试计算机信息管理专业和计算机网络专业自学指导丛书)

ISBN 7-302-04260-8

I. 数… II. 丁… III. 数据结构-高等教育…自学考试-自学参考资料 IV. TP311.12

中国版本图书馆 CIP 数据核字(2001)第 07937 号

出版者:清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印刷者:清华大学印刷厂

发行者:新华书店总店北京发行所

开 本:787×1092 1/16 印张:15.75 字数:379 千字

版 次:2001 年 5 月第 1 版 2001 年 5 月第 1 次印刷

书 号:ISBN 7-302-04260-8/TP·2502

印 数:00001~20000

定 价:23.00 元

215246

# 出版前言

信息化和网络化是知识经济时代的重要特征。面对知识经济的挑战,社会急需大批计算机信息管理和计算机网络专业人才。为了适应国民经济和社会发展的迫切需要,高等教育自学考试计算机信息管理专业和计算机网络专业的开考应运而生。

计算机信息管理专业(包括专科和独立本科段)是由信息产业部委托高等教育自学考试指导委员会开设的,计算机网络专业(独立本科段)是由高等教育自学考试指导委员会与信息产业部合作开考的,国家承认其学历和学位。信息产业部指定全国电子信息应用教育中心负责全国计算机信息管理专业和计算机网络专业自学考试助学工作的统一管理,各省(市)电子信息应用主管部门也指定本省(市)的电子信息应用教育中心负责当地的助学工作。至今,全国30个省(市)教育中心在各大中城市建立了近600个教学站,招收了10多万名学员。各地的主考大学大多是名牌大学,如清华大学、复旦大学等。

为了加强计算机信息管理和计算机网络两个专业的助学指导工作,全国电子信息应用教育中心组织有关专家和有丰富教学经验的教授,建立了自学指导丛书编委会,将陆续编写出版上述两个专业各门课程的自学指导书。

本套丛书力求知识完整独立、通俗易懂、便于自学,其中还包括了大量的练习题及其参考答案,是一套很实用的自学参考丛书。我们相信对于学员以及授课教师会有较大的帮助。

由于组织编写时间仓促,书中的不足在所难免,恳请读者指正。

有关本套丛书的信息,读者可到下列网址查询。

[www.ceiaec.org](http://www.ceiaec.org)

全国电子信息应用教育中心  
自学指导丛书编委会  
2000年6月

# 全国电子信息应用教育中心自学指导丛书

## 编 委 会

主 任 姚志清

副 主 任 侯炳辉 甘仞初 罗晓沛 陈 禹

委 员 (按姓氏笔画为序):

王长梗 王守茂 王志昌 甘仞初 田孝文 龙和平

沈林兴 罗晓沛 陈 禹 杨 成 杨冬青 杨觉英

姚志清 侯炳辉 张公忠 张国鸣 张宗根 袁保宗

徐甲同 徐立华 徐玉彬 盛定宇 彭 澎 韩培尧

雷震甲 魏晴宇

秘 书 长 沈林兴

副秘书长 彭 澎

秘书处联系地址 北京 2515 信箱教育中心(邮编: 100043)

# 前 言

本书作为全国高等教育自学考试指定教材《数据结构》的辅导材料,供“计算机及应用”和“计算机信息管理”、“计算机通信工程”等专业(独立本科段)的考生在学习“数据结构”课程时使用。

本书的编写依据是:

(1) 全国高等教育自学考试指导委员会制定的《数据结构自学考试大纲》(1999年5月)的命题原则和命题范围。

(2) 全国高等教育自学考试指导委员会组编的《数据结构》教材(黄刘生主编,经济科学出版社出版,2000年3月第一版)。

我们依据大纲,对教材中的知识点、重点、难点、习题进行了详细的分析。

全书分为两个部分:第一部分是教材中各章内容辅导,包括考核知识的分析,教材中习题的分析与解答,练习题及解答,上机实习题;第二部分为三套自测题及解答。各章练习及自测题是按照“数据结构”自学考试大纲的要求及教材内容给出的。

“数据结构”的主要内容是算法。掌握算法的诀窍是“三会”,即对每个操作能从三个角度去认识:会在示意图上表示,会写语句,会解释。自学者对每一知识点能掌握1~2个算法,就不难学好这门课了。

教材中已有很多习题,我们又增加了许多练习题,自学者可以根据实际情况遵循由浅入深,逐步加强的原则,挑选部分习题,加以阅读、分析、理解和掌握。

本书第1、2章由复旦大学张守志编写,第3、10章由上海第一电子信息应用中心陈坚编写,第4、5章由复旦大学陈金海、蔡军编写,第6、7章由上海理工大学邓桂英编写,第8、9章由上海大学曹旻编写。全书由丁宝康任主编,陈金海、陈坚任副主编,负责全书的组织、修改、定稿工作。

复旦大学招兆铿教授、夏玲明老师,上海市经济管理干部学院谈育安老师、陈静老师为本书的出版做了大量工作,在此表示感谢。本书的出版得到了复旦大学计算机系教育培训中心、上海(国际)数据库中心和上海市第一电子信息应用教育中心的支持。

限于水平,书中欠妥之处,敬请广大读者和专家指正。

编 者

2000年9月于上海

# 目 录

|                       |    |
|-----------------------|----|
| 第一部分 各章内容辅导 .....     | 1  |
| 第 1 章 概论 .....        | 1  |
| 1.1 考核知识点的分析 .....    | 1  |
| 1.2 教材中习题的分析与解答 ..... | 5  |
| 1.3 练习题 .....         | 7  |
| 1.4 练习题解答 .....       | 8  |
| 第 2 章 线性表 .....       | 11 |
| 2.1 考核知识点的分析 .....    | 11 |
| 2.2 教材中习题的分析与解答 ..... | 17 |
| 2.3 练习题 .....         | 24 |
| 2.4 练习题解答 .....       | 26 |
| 2.5 上机实习题及程序 .....    | 29 |
| 第 3 章 栈和队列 .....      | 31 |
| 3.1 考核知识点的分析 .....    | 31 |
| 3.2 教材中习题的分析与解答 ..... | 35 |
| 3.3 练习题 .....         | 42 |
| 3.4 练习题解答 .....       | 43 |
| 3.5 上机实习题及程序 .....    | 46 |
| 第 4 章 串 .....         | 49 |
| 4.1 考核知识点的分析 .....    | 49 |
| 4.2 教材中习题的分析与解答 ..... | 50 |
| 4.3 练习题 .....         | 56 |
| 4.4 练习题解答 .....       | 58 |
| 第 5 章 多维数组和广义表 .....  | 64 |
| 5.1 考核知识点的分析 .....    | 64 |
| 5.2 教材中习题的分析与解答 ..... | 65 |
| 5.3 练习题 .....         | 69 |
| 5.4 练习题解答 .....       | 73 |
| 第 6 章 树 .....         | 79 |
| 6.1 考核知识点的分析 .....    | 79 |
| 6.2 教材中习题的分析与解答 ..... | 90 |

|                            |            |
|----------------------------|------------|
| 6.3 练习题 .....              | 101        |
| 6.4 练习题解答 .....            | 107        |
| 6.5 上机实习题及程序 .....         | 112        |
| <b>第7章 图 .....</b>         | <b>115</b> |
| 7.1 考核知识点的分析 .....         | 115        |
| 7.2 教材中习题的分析与解答 .....      | 129        |
| 7.3 练习题 .....              | 139        |
| 7.4 练习题解答 .....            | 143        |
| 7.5 上机实习题及程序 .....         | 146        |
| <b>第8章 排序 .....</b>        | <b>149</b> |
| 8.1 考核知识点的分析 .....         | 149        |
| 8.2 教材中习题的分析与解答 .....      | 153        |
| 8.3 练习题 .....              | 164        |
| 8.4 练习题解答 .....            | 167        |
| 8.5 上机实习题及程序 .....         | 173        |
| <b>第9章 查找 .....</b>        | <b>177</b> |
| 9.1 考核知识点的分析 .....         | 177        |
| 9.2 教材中习题的分析与解答 .....      | 182        |
| 9.3 练习题 .....              | 198        |
| 9.4 练习题解答 .....            | 201        |
| 9.5 上机实习题及程序 .....         | 210        |
| <b>第10章 文件 .....</b>       | <b>213</b> |
| 10.1 考核知识点的分析 .....        | 213        |
| 10.2 教材中习题的分析与解答 .....     | 216        |
| 10.3 练习题 .....             | 219        |
| 10.4 练习题解答 .....           | 221        |
| <b>第二部分 自测题及参考答案 .....</b> | <b>223</b> |
| 第1套 自测题 .....              | 223        |
| 第1套 自测题参考答案 .....          | 227        |
| 第2套 自测题 .....              | 230        |
| 第2套 自测题参考答案 .....          | 233        |
| 第3套 自测题 .....              | 236        |
| 第3套 自测题参考答案 .....          | 240        |



# 第一部分 各章内容辅导

## 第1章 概 论

### 1.1 考核知识点的分析

本章的目的是介绍数据结构中常用的基本概念和术语以及学习数据结构的意义,要求了解本章介绍的各种基本概念和术语,掌握算法描述和分析的方法。本章重点是了解数据结构的逻辑结构、存储结构及数据的运算三方面的概念及相互关系,难点是算法复杂度的分析方法。

#### 1.1.1 本章重点

##### 1. 数据

(1) 数据:是信息的载体,能够被计算机识别、存储和加工处理。它是计算机程序加工的“原料”。计算机可以处理的整数、实数、字符串、声音、图像等都属于数据的范畴。

(2) 数据元素:是数据的基本单位,也称为元素、结点、顶点、记录。一个数据元素可以由若干个数据项(也称为字段、域、属性)组成。数据项是具有独立含义的最小标识单位。

##### 2. 数据结构

数据结构:指的是数据及数据之间的相互关系,即数据的组织形式。它一般包括以下三个方面的内容。

(1) 数据的逻辑结构:指数据元素之间的逻辑关系。即从逻辑关系上描述数据,它与数据的存储无关,是独立于计算机的。

(2) 数据的存储结构:指数据元素及其关系在计算机存储器内的表示(也称为映象)。即它是数据的逻辑结构用计算机语言的实现,依赖于计算机语言。

(3) 数据的运算:指对数据施加的操作。数据的运算是定义在数据的逻辑结构上的,而实现是要在存储结构上进行。

##### 3. 数据类型

(1) 数据类型:是一个值的集合以及在这些值上定义的一组操作的总称。按“值”是否

可分解,可将数据类型划分为两类:原子类型,其值不可分解,如 C 语言的整型、字符型等标准类型及指针等简单的导出类型;结构类型,其值可分解成为若干个成分(也称为分量),如 C 语言的数组、结构等类型。通常数据类型可以看作是程序设计语言中已实现的数据结构。

(2) 抽象数据类型(ADT):是指抽象数据的组织和与之相关的操作。可看作是数据的逻辑结构及其在逻辑结构上定义的操作。它可看作是描述问题的模型,它独立于具体实现。其优点是将数据和操作封装在一起,使得用户程序只能通过 ADT 里定义的某些操作来访问其中的数据,从而实现了信息隐藏。

#### 4. 数据的逻辑结构

在不会产生混淆的前提下,常常将数据的逻辑结构简称为数据结构。

(1) 直接前趋和直接后继:指数据元素之间的逻辑关系中有两个结点 a 和 b,若 a 和 b 相邻,且 a 在 b 的前面,则称结点 a 为结点 b 的直接前趋,结点 b 为结点 a 的直接后继。

(2) 开始结点和终端结点:在数据的逻辑结构中,无直接前趋的结点称为开始结点;无直接后继的结点称为终端结点。

(3) 数据的逻辑结构有两大类。

① 线性结构:若结构是非空有限集,则有且仅有一个开始结点和一个终端结点,并且所有结点都最多只有一个直接前驱和一个直接后继。线性表就是一个典型的线性结构。

② 非线性结构:一个结点可能有多个直接前驱和直接后继。如树形结构和图结构。

#### 5. 数据存储结构的四种常用存储方法

(1) 顺序存储方法:把逻辑上相邻的结点存储在物理位置上相邻的存储单元里,这样结点间的逻辑关系由存储单元的邻接关系来体现。由此得到的存储表示称为顺序存储结构。

(2) 链接存储方法:结点间的逻辑关系是由附加的指针字段表示,此方法不要求逻辑上相邻的结点在物理位置上亦相邻。由此得到的存储表示称为链式存储结构。

(3) 索引存储方法:在存储结点信息的同时,还建立附加的索引表。索引表中的每一项称为索引项,其形式为:(关键字、地址)。若每个结点在索引表中都有一个索引项,则该索引表称为稠密索引。若一组结点在索引表中只对应一个索引项,则该索引表称为稀疏索引。

(4) 散列存储方法:其基本思想是根据结点的关键字直接计算出该结点的存储地址。

#### 6. 学习数据结构的意义

数据结构是计算机软件和计算机应用专业的核心课程,在众多的计算机系统软件和应用软件中都要用到各种数据结构。程序设计的实质是对实际问题选择一种好的数据结构,并设计一个好的算法,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。所以选择合适的数据结构是解决应用问题的关键步骤。

## 7. 算法的描述和分析

数据的运算是通过算法来描述的。

算法是任意一个量定义的计算过程,它以一个或多个值作为输入,并产生一个或多个值作为输出。即一个算法是一系列将输入转换为输出的计算步骤。

一个问题的输入实例是满足问题陈述中所给出的限制、为计算该问题的解所需要的所有输入构成。

若一个算法对于每个输入实例均能终止并给出正确的结果,则称该算法是正确的。

对算法的说明可以用不同的语言,但惟一的要求是该说明必须精确地描述计算过程。对同一问题可能有许多不同的算法;但选用的算法首先应是正确的,其次主要考虑以下三点:

- (1) 执行算法所耗费的时间,即时间效率;
- (2) 执行算法所耗费的存储空间,其中主要考虑辅助存储空间,即空间效率;
- (3) 算法应易于理解,易于编码,易于调试,等等。

实际上,上述三点要求有时相互抵触,我们只能根据具体情况有所侧重。

算法的复杂度(或效率分析)可分为时间复杂度和空间复杂度。

### (1) 时间复杂度

一个算法所耗费的时间应是该算法中每条语句执行时间之和;而每条语句的执行时间是该语句的执行次数(也称为频度)与该语句执行一次所需时间的乘积。假设每条语句执行一次所需的时间均是单位时间,那么一个算法的时间耗费就是该算法中所有语句的频度之和。

我们将算法求解问题的输入量称为问题的规模,并用一个整数表示。一个算法的时间复杂度(也称时间复杂性) $T(n)$ 则是该算法的时间耗费,它是该算法所求解问题规模  $n$  的函数。当  $n \rightarrow \infty$  时,我们把时间复杂度  $T(n)$  的数量级(阶)称为算法的渐近时间复杂度,并用大“O”记号表示。即:若有  $\lim_{n \rightarrow \infty} T(n)/f(n) = C$ ,  $C$  为不为零的常数;亦即  $T(n)$  和  $f(n)$  是同阶的,或数量级相同,可记作  $T(n) = O(f(n))$ 。当  $T(n)$  是与问题规模  $n$  无关的常数时,则算法的时间复杂度为常数级(阶),记作  $T(n) = O(1)$ 。

评价一个算法的时间性能时,主要标准是算法时间复杂度的数量级,即算法的渐近时间复杂度。往往对两者不予区分,而经常将后者简称为时间复杂度。其中  $f(n)$  一般是算法中频度最大的语句频度。

由于算法中语句的频度(执行次数)不仅与问题规模  $n$  有关,还与输入实例有关,这时,可用最坏情况下时间复杂度作为算法的时间复杂度。

平均时间复杂度是指所有可能的输入实例均以等概率出现的情况下,算法的期望运行时间。

常见的时间复杂度,按数量级递增排列,则依次是:常数阶  $O(1)$ 、对数阶  $O(\log_2 n)$ 、线性阶  $O(n)$ 、线性对数阶  $O(n \log_2 n)$ 、平方阶  $O(n^2)$ 、立方阶  $O(n^3)$ 、 $\dots$   $k$  次方阶  $O(n^k)$ 、指数阶  $O(2^n)$ 。

### (2) 空间复杂度

一个算法的空间复杂度  $S(n)$  定义为该算法所耗费的存储空间,它也是问题规模  $n$  的函

数。渐进空间复杂度也常常简称为空间复杂度。

## 1.1.2 本章难点

### 1. 数据结构的概念的理解

数据结构包括数据的逻辑结构、数据的存储结构和数据的运算三个方面的内容。这在以后讲解具体的数据结构时,都要涉及这三方面的内容,大家应在这儿有个总体的认识。

对一个数据结构的理解应从这三个方面来理解,且它们之间是相互联系的。如果我们用二元组  $B = \langle A, R \rangle$  来表示一个数据结构的逻辑结构,其中  $A$  是有限的结点的集合, $R$  是  $A$  中结点间关系的集合,则用计算机语言实现或存储  $B$  时,不仅要存储表示有关结点  $A$  的信息,还须存储表示关系  $R$  的信息。数据的运算是通过算法来描述,其定义是在逻辑结构  $B$  上的,而计算机实现是在存储结构上进行的。

数据结构中常用的运算有:检索、插入、删除、更新、排序等。

由于一个数据结构中三方面的内容是一个整体,所以,若其中有一方面是不同的,都冠以不同的数据结构名称来标识它们。例如,顺序表、链表和散列表的差异在于存储结构的不同;栈和队的差异在于运算的集合和运算的性质不同。

### 2. 算法的时间复杂度的分析

算法的时间复杂度  $T(n)$  是该算法的时间耗费。一个算法的时间耗费就是该算法中所有语句的执行次数(频度)之和。

算法的渐近时间复杂度(简称为时间复杂度)是当  $n \rightarrow \infty$  时(即当  $n$  相当大时), $T(n)$  的数量级(阶),用大“ $O$ ”记号表示。由于  $\lim_{n \rightarrow \infty} T(n)/f(n) = C$ ,  $C$  是不为零的常数,所以  $T(n) = O(f(n))$ 。其实, $f(n)$  就是  $T(n)$  中最高阶的那一项,是算法中频度最大的语句频度。由于我们经常用渐近时间复杂度作为时间复杂度,所以要学会求  $f(n)$ 。一般情况下,对循环语句只需考虑循环体中语句的执行次数,而忽略该语句中循环头的部分。有时,循环体中语句的频度不仅与问题规模  $n$  有关,还与输入实例等其他因素有关,这时可用最坏情况下的时间复杂度作为算法的时间复杂度;有时,也考虑算法的平均时间复杂度。这些都将在以下讲解。

## 1.1.3 考核重点

### 1. 基本概念和术语

数据、数据元素、数据项、数据结构、逻辑结构、存储结构、线性结构、非线性结构、算法、语句的频度、算法的时间复杂度、算法的渐近时间复杂度、平均时间复杂度、最坏时间复杂度、空间复杂度、直接前趋和后继、开始结点和终端结点。

### 2. 理解和分析

数据结构包含的三个方面的含义及其相互关系;按逻辑结构可把数据结构分成线性结构和非线性结构两大类;数据结构在软件系统中的作用;对一般算法时间复杂度的分析。

## 1.2 教材中习题的分析与解答

[题 1.1]、[题 1.2]、[题 1.3]略。

[题 1.4]设三个函数  $f, g, h$  分别为

$$f(n) = 100n^3 + n^2 + 1000 \quad g(n) = 25n^3 + 5000n^2 \quad h(n) = n^{1.5} + 5000n \lg n$$

请判断下列关系式是否成立:

$$(1) f(n) = O(g(n)) \quad (2) g(n) = O(f(n)) \quad (3) h(n) = O(n^{1.5})$$

$$(4) h(n) = O(n \lg n)$$

解: (1) 由于  $\lim_{n \rightarrow \infty} f(n)/g(n) = \lim_{n \rightarrow \infty} (100n^3 + n^2 + 1000)/(25n^3 + 5000n^2) = 4 (\neq 0)$

所以  $f(n) = O(g(n))$  成立。

$$(2) \text{ 同样 } \lim_{n \rightarrow \infty} g(n)/f(n) = 1/4 (\neq 0)$$

所以  $g(n) = O(f(n))$  成立。

$$(3) \text{ 由于 } \lim_{n \rightarrow \infty} h(n)/n^{1.5} = \lim_{n \rightarrow \infty} (n^{1.5} + 5000n \lg n)/n^{1.5} = 1$$

(注意:  $\lg n$  的阶要比  $n^{0.5}$  的阶要小)

所以  $h(n) = O(n^{1.5})$  成立。

$$(4) \text{ 由于 } \lim_{n \rightarrow \infty} h(n)/n \lg n = \lim_{n \rightarrow \infty} (n^{1.5} + 5000n \lg n)/n \lg n = \infty$$

所以  $h(n) = O(n \lg n)$  不成立。

[题 1.5] 设有两个算法在同一机器上运行,其执行时间分别为  $100n^2$  和  $2^n$ ,要使前者快于后者,  $n$  至少要多大?

解: 要使前者快于后者,即要求满足  $100n^2 < 2^n$  的  $n$ 。由于  $100n^2$  和  $2^n$  这两个函数都是单调增加函数,随着  $n$  的增大,后者的递增幅度比前者快。在  $n \geq 1$  的整数情况下,可测出当  $n \geq 15$  时,  $100n^2 < 2^n$  (可编个小程序去测算)。

所以:要使前者快于后者,  $n$  至少要大于等于 15。

[题 1.6] 设  $n$  为正整数,利用大“O”记号,将下列程序段的执行时间表示为  $n$  的函数。

$$(1) i = 1; k = 0;$$

while ( $i < n$ ) {

$k = k + 10 * i; i++;$

}

$$(2) i = 0; k = 0;$$

do {

$k = k + 10 * i; i++;$

} while ( $i < n$ );

$$(3) i = 1; j = 0;$$

while ( $i + j \leq n$ ) {

if ( $i > j$ )  $j++$

else  $i++;$

}

$$(4) x = n; // n > 1$$

while ( $x \geq (y + 1) * (y + 1)$ )

$y++;$

$$(5) x = 91; y = 100;$$

while ( $y > 0$ )

if ( $x > 100$ ) {  $x = x - 10; y--;$  }

else x++;

解: 考虑程序段的执行时间  $T(n)$ , 用渐近时间复杂度来度量, 即计算  $f(n)$ , 它为程序段频度最大的语句频度。

(1)  $f(n)$  应为该程序段中循环体内, 两条语句的频度之和, 即  $f(n) = (n-1) + (n-1) = 2n-2$ 。

所以  $T(n) = O(2n-2) = O(2n) = O(n)$ 。

(2)  $f(n)$  应为该程序段中循环体内, 两条语句的频度之和, 即  $f(n) = n + n = 2n$ 。

所以  $T(n) = O(2n) = O(n)$ 。

(3)  $f(n)$  应为该程序段中循环体内, if 语句的频度; 注意到每执行一次 if 语句,  $(i+j)$  的值只增加 1, 所以  $f(n) = n$ 。

因而有  $T(n) = O(n)$ 。

(4) 由于这个程序段中没指明  $y$  的初值, 所以无法算出  $y++$  的执行次数。现假定  $y$  的初值为 0, 则  $y++$  的频度  $f(n) = \lfloor \sqrt{n} \rfloor$  ( $\lfloor X \rfloor$  表示不大于  $X$  的最大整数)。

所以  $T(n) = O(\lfloor \sqrt{n} \rfloor) = O(\sqrt{n}) = O(n^{0.5})$ 。

(5)  $f(n)$  应为程序段中 if 语句的频度, 注意, 这个循环语句不会死循环, 经过有限次循环后, 循环结束; 所以  $T(n)$  是常数阶即  $T(n) = O(1)$ 。具体分析为:

初始时  $X = 91, Y = 100$  经 10 次循环后  $X = 101, Y = 100$ , 再经一次循环后  $X = 91, Y = 99$ , 又经 10 次循环后  $X = 101, Y = 99$ , 再经一次循环后  $X = 91, Y = 98, \dots$ 。不管怎样,  $Y$  的值在减少; 所以 While 语句经有限次循环后会结束。其实执行其中的 if 语句的次数  $f(n) = (10 + 1) \times 100 = 1100$ 。

[题 1.7] 略。

[题 1.8] 按增长率由小至大地顺序排列下列各函数。

$$2^{100}, (3/2)^n, (2/3)^n, n^n, \sqrt{n}, n!, 2^n, \lg n, n^{\lg n}, n^{3/2}$$

解: 其排列顺序为:

$$(2/3)^n, 2^{100}, \lg n, \sqrt{n}, n^{3/2}, n^{\lg n}, (3/2)^n, 2^n, n!, n^n$$

[题 1.9] 有时为了比较两个同数量级算法的优劣, 需突出主项的常数因子, 而将低次项用大“O”记号表示。例如, 设  $T_1(n) = 1.39n \lg n + 100n + 256 = 1.39n \lg n + O(n)$ ,  $T_2(n) = 2.0n \lg n - 2n = 2.0n \lg n + O(n)$ , 这两个式子表示, 当  $n$  足够大时  $T_1(n)$  优于  $T_2(n)$ , 因为前者的常数因子小于后者。请用此方法表示下列函数, 并指出当  $n$  足够大时, 哪一个较优, 哪一个较劣?

$$(1) T_1(n) = 5n^2 - 3n + 60 \lg n \quad (2) T_2(n) = 3n^2 + 100n + 3 \lg n$$

$$(3) T_3(n) = 8n^2 + 3 \lg n \quad (4) T_4(n) = 1.5n^2 + 6000n \lg n$$

解: 由于  $T_1(n) = 5n^2 - 3n + 60 \lg n = 5n^2 + O(n)$

$$T_2(n) = 3n^2 + 100n + 3 \lg n = 3n^2 + O(n)$$

$$T_3(n) = 8n^2 + 3 \lg n = 8n^2 + O(\lg n)$$

$$T_4(n) = 1.5n^2 + 6000n \lg n = 1.5n^2 + O(n \lg n)$$

所以, 当  $n$  足够大时, 从优到劣的顺序为:  $T_4(n)$ 、 $T_2(n)$ 、 $T_1(n)$ 、 $T_3(n)$ 。

## 1.3 练习题

### 一、名词解释

数据 数据元素 数据结构 数据类型 数据的逻辑结构和存储结构 线性结构 非线性结构 算法 算法的时间复杂度 算法的渐近时间复杂度 最坏和平均时间复杂度

### 二、单项选择题

1. 数据结构通常是研究数据的( )及它们之间的相互联系。  
A. 存储和逻辑结构      B. 存储和抽象  
C. 理想与抽象          D. 理想与逻辑
2. 数据在计算机存储器内表示时,物理地址与逻辑地址相同并且是连续的,称之为( )。  
A. 存储结构              B. 逻辑结构  
C. 顺序存储结构          D. 链式存储结构
3. 非线性结构是数据元素之间存在一种( )。  
A. 一对多关系              B. 多对多关系  
C. 多对一关系              D. 一对一关系
4. 非线性结构中,每个结点( )。  
A. 无直接前趋  
B. 只有一个直接前驱和后继  
C. 只有一个直接前趋和个数不受限制的直接后继  
D. 有个数不受限制的直接前趋和后继
5. 除了考虑存储数据结构本身所占用的空间外,实现算法所用辅助空间的多少称为算法的( )。  
A. 时间效率                  B. 空间效率  
C. 硬件效率                  D. 软件效率
6. 链接存储的存储结构所占存储空间( )。  
A. 分两部分,一部分存放结点值,另一部分存放表示结点间关系的指针  
B. 只有一部分,存放结点值  
C. 只有一部分,存放表示结点间关系的指针  
D. 分两部分,一部分存放结点值,另一部分存放结点所占单元数
7. 设语句 $x++$ 的时间是单位时间,则语句:  
    for( $i = 1; i \leq n; i++$ )  
         $x++$ ;  
的时间复杂度为( )。  
A.  $O(1)$       B.  $O(n)$       C.  $O(n^2)$       D.  $O(n^3)$

### 三、填空题

1. 数据结构包括数据的\_\_\_\_\_、数据的\_\_\_\_\_和数据的\_\_\_\_\_这三个方面的内容。
2. 数据结构按逻辑结构可分为两大类,它们分别是\_\_\_\_\_和\_\_\_\_\_。
3. 数据的存储结构可用四种基本的存储方法表示,它们分别是\_\_\_\_\_,\_\_\_\_\_,\_\_\_\_\_,\_\_\_\_\_和\_\_\_\_\_。
4. 数据的运算最常用的有五种,它们分别是\_\_\_\_\_,\_\_\_\_\_,\_\_\_\_\_,\_\_\_\_\_和\_\_\_\_\_。
5. 线性结构反映结点间的逻辑关系是\_\_\_\_\_的,非线性结构反映结点间的逻辑关系是\_\_\_\_\_的。
6. 一个算法的效率可分为\_\_\_\_\_效率和\_\_\_\_\_效率。

### 四、简答题

1. 简述线性结构与非线性结构的不同点。
2. 简述数据的存储结构中四种常用的存储方法。
3. 简述稠密索引与稀疏索引的不同点。
4. 简述算法复杂度(效率分析)的评价方法。

### 五、阅读理解题

1. 设  $n$  为正整数,利用大“O”记号,将下列程序段的执行时间表示为  $n$  的函数。

```
(1) x=0;
    for(i=1;i<n;i++)
        for(j=i+1;j<=n;j++)
            x++;
```

```
(2) x=0;
    for(i=1;i<n,i++)
        for(j=1;j<=n-i;j++)
            x++;
```

## 1.4 练习题解答

### 一、名词解释答案

**数据:** 是信息的载体,它能够被计算机识别、存储和加工处理。

**数据元素:** 是数据的基本单位,也称为结点、顶点、记录。它可以由若干个数据项组成,数据项是具有独立含义的最小标识单位。

**数据结构:** 指的是数据及数据之间的相互关系,即数据的组织形式,它包括数据的逻辑结构、数据的存储结构和数据的运算三个方面的内容。



数据的逻辑结构:指的是数据元素之间的逻辑关系,是从逻辑上描述数据,与数据的存储无关,独立于计算机。

数据的存储结构:指的是数据元素及其关系在计算机存储器内的表示,是数据的逻辑结构用计算机语言的实现,是依赖于计算机语言的。

线性结构:其逻辑特征为,若结构是非空集,则有且仅有一个开始结点和一个终端结点,并且其余每个结点只有一个直接前趋和一个直接后继。

非线性结构:其逻辑特征为,一个结点可能有多个直接前趋和直接后继。

算法:是任意一个良定义的计算过程,它以一个或多个值作为输入,并产生一个或多个值作为输出;即一个算法是一系列将输入转换为输出的计算步骤。

算法的时间复杂度  $T(n)$ :是该算法的时间耗费,它是该算法所求解问题规模  $n$  的函数。当  $n$  趋向无穷大时,我们把时间复杂度  $T(n)$  的数量级(阶)称为算法的渐近时间复杂度。

最坏和平均时间复杂度:由于算法中语句的频度不仅与问题规模  $n$  有关,还与输入实例等因素有关;这时可用最坏情况下时间复杂度作为算法的时间复杂度。而平均时间复杂度是指所有的输入实例均以等概率出现的情况下,算法的期望运行时间。

## 二、单项选择题答案

1. A      2. C      3. B      4. D      5. B      6. A      7. B

## 三、填空题答案

1. 逻辑结构,存储结构,运算
2. 线性结构,非线性结构
3. 顺序存储方法、链接存储方法、索引存储方法、散列存储方法
4. 检索、插入、删除、更新、排序
5. 一对一,多对多
6. 时间,空间

## 四、简答题答案

1. 答:线性结构的逻辑特征是除开始结点和终端结点外,其余每个结点只有一个直接前趋和一个直接后继,即结点间存在一对一的关系;而非线性结构的逻辑特征是一个结点可以有多个直接前趋和直接后继,即结点间存在多对多的关系。

2. 答:顺序存储方法:把逻辑上相邻的结点存储在物理位置相邻的存储单元里,这样结点间的逻辑关系由存储单元的邻接关系来体现。

链接存储方法:除存储结点外,还要由附加的指针字段来表示结点间的逻辑关系。

索引存储方法:在存储结点的同时,还建立附加的索引表。

散列存储方法:根据结点的关键字直接计算出该结点的存储地址。

3. 答:稠密索引:每个结点在索引表中都有一个索引项。

稀疏索引:一组结点在索引表中只对应一个索引项。

4. 答:算法复杂度可分为时间复杂度和空间复杂度。时间复杂度是该算法所耗费的时间,可用该算法中所有语句的频度之和来计算。由于评价一个算法的时间性能时,主要标准