

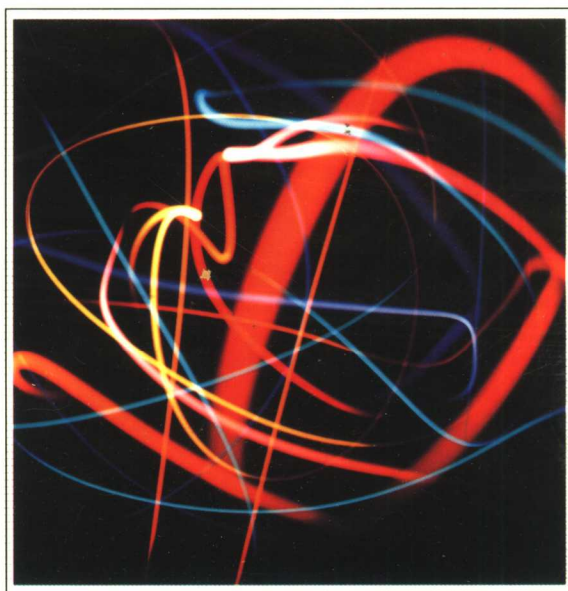
O Y A N X I N G A N X I A N
考 研 新 干 线

丛书主编 王凤兰

数 据 结 构

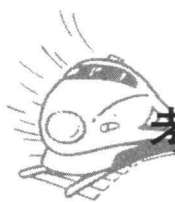
常见题型解析及模拟题

刘遵仁 编著



國防工業出版社

National Defense Industry Press



考研新干线

数据结构

常见题型解析及模拟题

丛书主编 王凤兰

编著 刘遵仁

国防工业出版社

· 北京 ·

图书在版编目(CIP)数据

数据结构常见题型解析及模拟题/刘遵仁编著. —北京:国防工业出版社, 2006. 3
(考研新干线/王凤兰主编)
ISBN 7-118-04312-5

I. 数... II. 刘... III. 数据结构—研究生—入学考试—解题 IV. TP311.12—44

中国版本图书馆 CIP 数据核字(2006)第 009842 号

※

国防工业出版社 出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100044)

天利华印刷装订有限公司印刷

新华书店经售

*

开本 787×1092 1/16 印张 18 字数 410 千字

2006 年 3 月第 1 版第 1 次印刷 印数 1—4000 册 定价 28.00 元

(本书如有印装错误, 我社负责调换)

国防书店: (010)68428422

发行邮购: (010)68414474

发行传真: (010)68411535

发行业务: (010)68472764

《考研新干线》丛书编委会

丛书主编 王凤兰

编委会成员

秦安琳	许 之	程 鹏	蒋持平
樊昌信	申利民	刘长林	韩向春
丁天昌	苏 媛	许曰滨	刘遵仁
曲继方	王淑娟	王宇野	张华弟
董五洲	张德斌	聂国权	徐亚清
戴 民	王铁军	赵晓冬	杨 茜
李继勇			

前 言

“数据结构”在计算机科学中是一门综合性的专业基础课,也是大多数高校招收计算机专业及相关专业硕士研究生的必考科目之一。

本书是编者在对全国多所著名院校研究生入学考试试题分析的基础上,结合目前考研的发展动向编写而成的。全书分为11章,第1章为绪论,第2章为线性表,第3章为栈和队列,第4章为串,第5章为数组和广义表,第6章为树和二叉树,第7章为图,第8章为查找,第9章为内排序,第10章为外排序,第11章为文件。

每章由三部分构成,即考试要点复习、典型例题解析、训练题及参考答案。考试要点复习部分高度概括了本章考试内容及要点;典型例题解析部分通过对典型例题剖析解答,融每章的重点、难点和常用方法于典型例题之中;训练题及参考答案收集了大量的相关试题,并给出了相应的参考答案。训练题按照知识层次组织为选择题、填空题、简答题、算法设计与分析题。为便于读者学习使用,方便查找,每一题型中习题基本上按照知识点的顺序从易到难进行了归纳组织。

本书中的绝大多数题目是从近几年来全国多所高校数据结构考研试题中精选出的,并给出了详解和参考答案,有一小部分题目虽然不是考研试题,但确很有代表性。不少研究生的入学考试试题来自国内外著名教材和辅导书的习题或习题的变型,由于这些题目或者思路新颖,或者涉及知识点十分重要,或者解题方法独到,代表性很强,还有可能直接或间接以某种变型的方式出现在考研试题中,本书对于此类典型的试题均安排在典型例题解析部分。

“数据结构”课程相对来说,题材比较熟悉,考查的重点比较明确,一般在入学考试中是比较容易得高分的。只要读者依据我们撰写的内容及各高校出题的特点,有针对性地复习,就一定能取得较理想的成绩。

本书中的知识要点主要参考清华大学出版社出版、严蔚敏和吴伟民编著的《数据结构(C语言版)》一书,在学习和使用的过程中,需注意部分概念在不同教材体系中描述的差异。

书中的部分程序在 Visual C++ 6.0/TC3.0 环境下调试通过,读者只需要掌握简单的 C 语言基础知识就能阅读这些程序。

本书的特点是概念清晰,文字简洁明了,所有题目都给出了详细的解答,极便于读者在短时间内掌握解题要点。

尽管本书主要针对考研者应试复习和提高,但同样也适合于作为大专院校各专业数据结构课程的复习参考书,还可供计算机软件等级考试者研习。

由于作者水平有限,书中难免存在不足之处,敬请有关专家和广大读者不吝指正。

编 者

2005 年 10 月

V

目 录

第 1 章 绪论	1
1.1 考试要点复习	1
1.1.1 数据结构的基本概念	1
1.1.2 算法和算法分析	3
1.1.3 考试要点提示	4
1.2 典型例题剖析	4
1.3 训练题及参考答案	8
1.3.1 训练题	8
1.3.2 训练题参考答案	10
第 2 章 线性表	12
2.1 考试要点复习	12
2.1.1 线性表的基本概念	12
2.1.2 线性表的顺序存储结构	13
2.1.3 线性表的链式存储结构	15
2.1.4 关于引用类型 & 的说明	16
2.1.5 重点和难点	17
2.2 典型例题解析	17
2.3 训练题及参考答案	26
2.3.1 训练题	26
2.3.2 训练题参考答案	30
第 3 章 栈和队列	45
3.1 考试要点复习	45
3.1.1 栈的基本概念	45
3.1.2 栈的顺序存储结构	46
3.1.3 顺序栈的静态存储结构	46
3.1.4 顺序栈的动态分配存储结构	47
3.1.5 栈的链式存储结构	48
3.1.6 递归算法设计	48
3.1.7 队列的基本概念	49
3.1.8 队列的顺序存储结构—循环队列	50
3.1.9 队列的链式存储结构	51
3.1.10 双端队列	51

3.1.11 重点和难点	52
3.2 典型例题解析	52
3.3 训练题及参考答案	61
3.3.1 训练题	61
3.3.2 训练题参考答案	64
第4章 串	74
4.1 考试要点复习	74
4.1.1 串的基本概念	74
4.1.2 串的顺序存储结构	75
4.1.3 串的链式存储结构	75
4.1.4 串的模式匹配	75
4.2 典型例题解析	77
4.3 训练题及参考答案	82
4.3.1 训练题	82
4.3.2 训练题参考答案	82
第5章 数组和广义表	84
5.1 考试要点复习	84
5.1.1 数组的基本概念	84
5.1.2 特殊矩阵的压缩存储	84
5.1.3 稀疏矩阵的压缩存储	86
5.1.4 广义表	87
5.1.5 重点和难点	88
5.2 典型例题解析	88
5.3 训练题及参考答案	98
5.3.1 训练题	98
5.3.2 训练题参考答案	101
第6章 树与二叉树	106
6.1 考试要点复习	106
6.1.1 树的基本概念	106
6.1.2 二叉树的基本概念	107
6.1.3 二叉树的顺序存储结构	108
6.1.4 二叉树的链式存储结构	108
6.1.5 二叉树的遍历	109
6.1.6 线索二叉树	109
6.1.7 树和森林	110
6.1.8 哈夫曼树	112
6.1.9 重点和难点	112
6.2 典型例题解析	113
6.3 训练题及参考答案	131

6.3.1 训练题	131
6.3.2 训练题参考答案	137
第7章 图	150
7.1 考试要点复习	150
7.1.1 图的基本概念	150
7.1.2 图的存储结构	151
7.1.3 图的遍历	153
7.1.4 连通性和生成树	154
7.1.5 AOV网与拓扑排序	155
7.1.6 AOE网与关键路径	155
7.1.7 最短路径	156
7.1.8 重点和难点	157
7.2 典型例题解析	157
7.3 训练题及参考答案	174
7.3.1 训练题	174
7.3.2 训练题参考答案	182
第8章 查找	192
8.1 考试要点复习	192
8.1.1 查找的基本概念	192
8.1.2 静态查找表	193
8.1.3 静态树表的查找	194
8.1.4 动态查找表——树表查找	194
8.1.5 哈希表查找	197
8.1.6 重点和难点	198
8.2 典型例题解析	198
8.3 训练题及参考答案	216
8.3.1 训练题	216
8.3.2 训练题参考答案	220
第9章 内部排序	230
9.1 考试要点复习	230
9.1.1 排序的概念	230
9.1.2 顺序存储结构的描述	230
9.1.3 插入排序	231
9.1.4 交换排序	231
9.1.5 选择排序	232
9.1.6 归并排序	233
9.1.7 基数排序	233
9.1.8 各种排序方法性能比较	233
9.1.9 重点和难点	234

9.2 典型例题解析	234
9.3 训练题及参考答案	244
9.3.1 训练题	244
9.3.2 训练题参考答案	249
第 10 章 外部排序	258
10.1 考试要点复习	258
10.1.1 外存信息的存取	258
10.1.2 外部排序的实现	258
10.1.3 重点和难点	259
10.2 典型例题解析	259
10.3 训练题及参考答案	262
10.3.1 训练题	262
10.3.2 训练题参考答案	263
第 11 章 文件	265
11.1 考试要点复习	265
11.1.1 文件的基本概念	265
11.1.2 文件的存储组织	266
11.1.3 重点和难点	269
11.2 典型例题解析	269
11.3 训练题及参考答案	270
11.3.1 训练题	270
11.3.2 训练题参考答案	272
数据结构考试题型简介	274
参考文献	276

第 1 章 绪 论

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等的学科。

本章需要掌握的知识点、重点和难点有:

- 基本知识点:数据结构和算法的概念。
- 重点:数据结构的逻辑结构、存储结构和数据操作(运算)3个方面的概念及其相互之间的关系;算法时间复杂度分析。
- 难点:分析算法的时间复杂度。

1.1 考试要点复习

1.1.1 数据结构的基本概念

1. 数据(Data)

数据是对客观事物的符号表示,在计算机科学中是指所有能输入到计算机中,并被计算机程序处理的符号的总称。随着计算机科学的发展,数据概念的内涵在逐步扩大。例如,整数、实数、字符串、图像、声音等。

2. 数据元素(Data Element)

数据元素是数据的基本单位,通常作为一个整体进行考虑和处理。一个数据元素由若干个数据项组成,而数据项是数据不可分割的最小单位。

3. 数据对象(Data Object)

数据对象是性质相同的数据元素的集合,是一个数据的子集。

4. 数据结构(Data Structure)

数据结构是数据元素相互之间存在一种或多种特定关系的集合,其形式定义为一个二元组:

$$\text{Data_Structure} = (D, S)$$

其中,D是数据元素的有限集;S是D上关系的有限集。

数据结构一般包括以下3方面的内容:

(1) 数据的逻辑结构

数据的逻辑结构是从逻辑关系上来观察数据,它与数据的存储方式无关,是独立于计算机的。根据数据元素之间关系的不同特性,基本逻辑结构通常有4种:

- ① 集合:集合中的数据元素之间除了“同属于一个集合”的关系外,别无其他的关系。
- ② 线性结构:数据元素之间存在“一对一”的关系,有且仅有一个开始结点和终端结

点。除开始结点外,每个结点有且仅有一个前驱结点,除终端结点外,每个结点有且仅有一个后继结点。

③ 树型结构:数据元素之间存在“一对多”的关系,有一个开始结点和多个终端结点。除开始结点外,每个结点有且仅有一个前驱结点,除终端结点外,每个结点可能有多个后继结点。

④ 网状结构或图状结构:数据元素之间存在“多对多”的关系,每个结点可能有多个前驱结点和多个后继结点。

集合是数据元素之间关系极为松散的一种结构,因此也可以用其他结构来表示。树型结构和网状结构又称为非线性结构。

(2) 数据的存储结构

数据的存储结构或物理结构是逻辑结构在计算机存储器里的实现(也称为映像),它是依赖于计算机的。元素之间的逻辑关系在计算机中有两种不同的表示方法:顺序映像和非顺序映像。具体的存储方法可分为4种:

① 顺序存储:顺序存储是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,结点之间的逻辑关系是由物理存储单元的邻接关系来体现。通常顺序存储结构是借助于数组来描述的。

② 链式存储:在链式存储中,一个结点所占的存储单元分为两部分:一部分存放结点本身的信息,即数据项;另一部分存放该结点的后继结点所在存储单元的地址,即指针项。与顺序存储方式不同,链式存储方法不要求逻辑上相邻的结点在物理位置上必须相邻,结点间的逻辑关系是通过结点中的指针来表示的。通常链式存储结构是借助于C语言的指针类型来描述的。

③ 索引存储:索引存储方法通常是在存储结点信息的同时,还需要建立一个索引表。索引表中的每一项称为索引项,通过索引项来确定结点的存储地址。其中关键字是唯一地标识结点。

④ 散列存储:散列存储是根据结点的关键字来确定它的存储地址。

(3) 数据的运算

数据的运算是指对数据元素施加的操作。数据的运算是定义在数据的逻辑结构上的,而实现数据的运算则要在存储结构上进行。

【说明】

① 除顺序存储外,其他存储方式除存储数据本身外,还要额外地占用存储空间。因此,空间利用率降低。

② 如果所有的存储空间都分配给了数据,则这个存储结构叫紧凑结构,否则称为非紧凑结构。顺序存储的线性表是紧凑结构,而链式存储结构等是非紧凑结构。

③ 数据的逻辑结构和存储结构是密切相关的,任何一个算法的设计取决于选定的数据(逻辑)结构,而算法的实现则取决于所采用的存储结构。因此,数据结构是由数据的逻辑结构、数据的存储结构和数据的运算三部分组成,或者说按照某种逻辑关系组织一批数据,按一定的存储方式把它们存储在计算机的存储器中,并在这些数据上定义运算的集合,就叫做数据结构。

5. 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。

6. 抽象数据类型(Abstract Data Type, 简称 ADT)

抽象数据类型是指一个数学模型以及定义在该模型上的一组操作。抽象数据类型的定义仅取决于它的一组逻辑特性,而与其在计算机内部如何表示和实现无关,即不论其内部结构如何变化,只要它的数学特性不变,都不影响其外部的使用。

为了提高软件的复用率,近代程序设计方法学指出,一个软件系统的框架应建立在数据之上,而不是建立在操作之上。所定义的数据类型抽象层次越高,含有该抽象数据类型的软件模块的复用程度就越高。

1.1.2 算法和算法分析

1. 算法及其特性

算法是对特定问题求解步骤的一种描述,是指令的有序序列。

由于数据的逻辑结构和存储结构不是唯一的,往往是由用户自行选择和设计的,所以处理同一问题的算法也不是唯一的。即使对相同的逻辑结构和存储结构而言,其算法的设计思想不同,则编写出的算法也大不相同。

算法的 5 个重要特性:有穷性;确定性;输入;输出;可行性。

2. 算法的评价

评价一个算法一般从以下四个方面进行:正确性;可读性;健壮性;高效率与低存储量。

在计算机科学中,时间效率(运算量少)与空间效率(低存储量)往往是矛盾的,在实际应用中需要根据问题的要求和特点而进行选择。如:索引技术、散列技术是以空间换时间的技术;而数据压缩技术、覆盖技术则是以时间换空间的技术。

3. 算法效率的度量

一个语句的频度,是指该语句在算法中被重复执行的次数。算法中所有语句的频度之和是该算法所求解问题规模 n 的函数。一个特定算法“运行工作量”的大小,只依赖于问题的规模,或者说是问题规模的函数。

为了便于比较同一问题的不同算法,通常从算法中选取一种对于所研究的问题(或算法)来说是基本的原操作,以该基本操作重复执行的次数作为算法的时间量度。算法中基本操作重复执行的次数是问题规模 n 的某个函数 $f(n)$,算法的时间量度记作

$$T(n) = O(f(n))$$

它表示随问题规模 n 的增大,算法执行时间的增长率和 $f(n)$ 的增长率相同,称作算法的渐近时间复杂度,简称时间复杂度。

由于算法的时间复杂度考虑的只是对于问题规模 n 的增长率,在难以精确计算基本操作执行次数(或语句频度)的情况下,只需求出它关于 n 的增长率即可。

下表给出了各种有代表性的 $f(n)$ 函数的算法,在不同 n 值时的运行时间。表中凡是未注明时间单位的为 $\mu s(10^{-6}s)$ 。因算法的实际运行时间随机器而异,所以此表上的时间主要用于相互比较。

算法的运行时间与 $f(n)$ 函数的关系

$f(n)$	$\log_2 n$	n	$n \log_2 n$	n^2	n^3	n^5	2^n	$n!$
$n=20$	4.3	20	86.4	400	8 毫秒	3.2 秒	1.05 秒	771 世纪
$n=40$	5.3	40	213	1600	64 毫秒	1.7 分	12.7 天	2.59×10^{32} 世纪
$n=60$	5.9	60	354	3600	216 毫秒	13 分	366 世纪	2.64×10^{66} 世纪

从表中可以得出如下结论：

(1) 当 $f(n)$ 为对数函数、幂函数或它们的乘积时，算法的运行时间是可以接受的，我们称这些算法为有效的算法；当 $f(n)$ 为指数函数或阶乘函数时，算法的运行时间随着 n 而迅速增长是不可接受的，我们称这种算法是“坏”的算法或无效的算法。

(2) 随着 n 值的增大，各种 $f(n)$ 函数所对应的运行时间的增长速度大不相同，对数函数的增长速度最慢，线性函数稍快些，其余类推。因此，到 n 足够大时，各种不同数量级的 $f(n)$ 函数存在着下列关系：

$$O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < \dots < O(2^n) < O(n!)$$

(3) 在设计算法时一般情况下应尽可能避免使用时间复杂度是指数阶的，而使用复杂度是低阶多项式的，以便能在有效的时间内解决问题。但在计算机安全的加密算法中，却需要有足够的复杂性，以保证不能使入侵者在有效时间内攻击系统。

(4) 一般总是考虑在最坏情况下的时间复杂度，以保证算法的运行时间不会比它更长。

4. 算法的空间复杂度

算法的空间复杂度 $S(n)$ ，定义为该算法所耗费的存储空间，它是问题规模 n 的函数。渐近空间复杂度也常常简称为空间复杂度。

1.1.3 考试要点提示

本章考点主要有：数据（逻辑）结构、存储结构的定义及其关系、语句执行频度、算法时间复杂度、空间复杂度的计算。

1.2 典型例题剖析

【例 1—1】 试解答下列问题。

(1) 一个算法具有()等特点。

A. 可行性

B. 至少有一个输入量

C. 确定性

D. 健壮性

(2) 算法的计算量的大小称为计算的()。

A. 效率

B. 复杂度

C. 现实性

D. 难度

(3) 一个算法具有 5 个特性：_____、_____、_____、有零个或多个输入、有一个或多个输出。

(4) 一个数据结构在计算机中的_____称为存储结构。

(5) (判断题)数据的逻辑结构是指数据的各数据项之间的逻辑关系。

(6) 数据的逻辑结构是指_____。

【例题分析】

• 这类试题均属于数据结构的基本概念题。这类考题在很多学校的研究生入学考试试卷中都反复出现,请考生务必重视对有关基本概念的理解。

- 这类试题主要考核“数据结构”、“逻辑结构”和“算法”的度量和特性等的基本概念。

【解答】

(1) 备选项 D(健壮性)是一个“好”的算法应达到的目标,而不是算法的特性。备选项 B 是不正确的,因为算法允许有零个输入量。故正确答案为 A 和 C。

(2) 效率包括时间复杂度和空间复杂度, 因此, A 不正确。算法的计算量的大小用时间复杂度来度量, 因此 B 为正确答案。

(3) 有穷性、确定性、可行性。

(4) 映像。

(5) 数据的逻辑结构是指各个数据元素之间的逻辑关系,而不是指数据元素中的各数据项之间的逻辑关系,故该命题是错误的。

(6) 数据元素之间的逻辑关系。

【例 1-2】 试解答下列问题。

(1) 在计算机的存储器中表示时,物理地址和逻辑地址相同并且是连续的,称之为 ()。

A. 逻辑结构

B. 顺序存储结构

C. 链式存储结构

D. 以上都对

(2) (简答题) 数据结构与数据类型有什么区别?

(3) (简答题)什么是抽象数据类型?

(4) (简答题)什么是算法的时间复杂度?

【例题分析】

- 这类试题主要考核“逻辑结构”和算法度量等基本概念。

【解答】

(1) 顺序存储结构特点是:物理地址和逻辑地址相同并且是连续的;链式存储结构的特点是:逻辑地址连续但物理地址不一定连续。因此正确答案为 B。

(2) 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, 一般包括三个方面的内容: 数据的逻辑结构、存储结构和数据的运算。而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。

(3) 抽象数据类型是指一个数学模型以及定义在该模型上的一组操作。

(4) 为便于比较同一问题的不同算法,通常从算法中选取一种对于所研究的问题来说是基本的原操作,以该基本操作重复执行的次数作为算法的时间量度。算法中基本操作重复执行的次数是问题规模 n 的某个函数 $f(n)$,算法的时间量度记作:

$$T(n) = O(f(n))$$

【例 1-3】 试解答下列的问题。

(1) 下列函数中渐近时间复杂度最小的是()。

A. $T(n) = n \log_2 n + 5000n$

B. $T(n) = n^2 - 8000n$

C. $T(n) = n^{\log_2 2n} - 6000n$

D. $T(n) = 2n^{\log_2 2n} - 7000n$

(2) 下列函数中渐近时间复杂度最小的是()。

A. $T(n) = n \log_2 n + 1000 \log_2 n$

B. $T(n) = n^2 - 1000 \log_2 n$

C. $T(n) = n^{\log_2 2n} - 1000 \log_2 n$

D. $T(n) = 2n^{\log_2 2n} - 1000 \log_2 n$

【例题分析】

• 这类试题主要考核对算法时间复杂度好坏的判别。判断算法时间复杂度好坏的依据就是求当 $n \rightarrow \infty$ (规模趋于无限大) 时函数的极限。

【解答】

这两道题的正确答案都为 A, 并且其时间复杂度为 $O(n \log_2 n)$ 。

【例 1-4】 设 n 为正整数, 试确定下面各程序段的时间复杂度。

(1)

```
void f1(int n)
{
    int i=1, k=0;
    while( i < n ) {
        k += 10 * i;
        i++;
    }
}
```

(2)

```
void f2(int n)
{
    int i=1, j=n;
    while(j >= (i+1) * (i+1))
        i++;
}
```

(3)

```
void f3(int n)
{
    int i=91, j=100;
    while(j > 0 )
        if(i > 100){
            x -= 10;
            y--;
        } else
            i++;
}
```

```

(4)
void f4(int n)
{
    int i=1;
    while(i < n)
        i *= 3;
}

```

【例题分析】

• 这类试题考核会计算简单的算法时间复杂度。计算时间复杂度的步骤是:(1)计算算法的语句频度;(2)根据语句频度给出时间复杂度。

• 在计算算法的语句频度时,只需计算执行次数最多的那条语句即可。

【解答】

设 $f(n)$ 为算法的频度, $T(n)$ 为时间复杂度。

(1) 函数 $f1$ 包含了一个单循环语句,循环次数为 $n-1$,因此算法的频度为 $n-1$ 。

$f(n) = n-1$,所以 $T(n) = O(n)$ 。

(2) 函数 $f2$ 包含了一个单循环语句,从循环语句的条件表达式可知: $2^2 < n, 3^2 < n,$

$\dots, (i+1)^2 < n, \dots$,所以 $i < \sqrt{n}-1$ 。 $f(n) = \sqrt{n}-1$,所以 $T(n) = O(\sqrt{n})$ 。

(3) 因为 while 语句与 n 无关,所以 $T(n) = O(1)$ 。

(4) 函数 $f4$ 包含了一个单循环语句,从循环语句的条件表达式可知: $3^0 < n, 3^1 < n, 3^2 < n,$

$\dots, 3^{i-1} < n, \dots$,所以有 $i < \log_3 n + 1$ 。 $f(n) = \log_3 n + 1$,所以 $T(n) = O(\log_3 n)$ 。

【例 1-5】 设算法所需时间由下述方程表示,试求该算法的时间复杂度。

【说明】

n 为求解问题的规模,为简单起见,设 n 为 2 的正整数幂。

$$T(n) = \begin{cases} 1, & n=1 \\ 2T(n/2) + n, & n>1 \end{cases}$$

【例题分析】

• 这类试题考核会计算简单的算法时间复杂度。计算时间复杂度的步骤是:(1)计算算法的语句频度;(2)根据语句频度给出时间复杂度。

• 在计算算法的语句频度时,只需计算执行次数最多的那条语句即可。

【解答】

设 $n=2^k (k \geq 0)$,则有关系式

$$T(2^k) = 2T(2^{k-1}) + 2^k = 2^2 T(2^{k-2}) + 2 * 2^k$$

该关系式的一般递推关系式为

$$T(2^k) = 2^i T(2^{k-i}) + i * 2^k$$

当 $i=k$ 时,可推得

$$T(2^k) = 2^k T(2^0) + k * 2^k = (k+1)2^k$$

即

$$T(n) = n(\log_2 n + 1) = n \log_2 n = O(n \log_2 n)$$

当 n 不是 2 的正整数幂时, $T(n)$ 仍满足方程。

1.3 训练题及参考答案

1.3.1 训练题

一、选择题

1. 每种数据结构都具备 3 个基本运算:插入、删除和查找,这种说法()。
A. 正确 B. 不正确
2. 数据结构被形式定义为 (D, S) ,其中 D 是()的有限集合, S 是 D 上的()有限的集合。
A. 算法 B. 数据元素 C. 数据操作 D. 逻辑结构
E. 关系 F. 映像 G. 操作 H. 存储
3. 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(),以及它们之间的()和运算的学科。
A. 操作对象 B. 计算方法 C. 逻辑存储 D. 数据映像
E. 结构 F. 关系 G. 运算 H. 算法
4. ()不是算法的基本特征。
A. 正确性 B. 长度有限 C. 在规定的时间内完成 D. 确定性
5. 算法分析的目的是(),算法分析的两个主要方面是()。
A. 给出数据结构的合理性 B. 研究算法中的输入和输出的关系
C. 分析算法的效率以求改进 D. 分析算法的易懂性和文档性
E. 空间复杂性和时间复杂性 F. 正确性和简明性
G. 可读性和文档性 H. 数据复杂性和程序复杂性
6. 下面关于算法的说法错误的是()。
A. 算法最终必须由计算机程序实现
B. 为解决某问题的算法同为该问题编写的程序含义是相同的
C. 算法的可行性是指指令不能有二义性
D. 以上几个都是错误的
7. 线性表的顺序存储结构是一种()的存储结构,线性表的链式存储结构是一种()存储结构。
A. 随机存取 B. 顺序存取 C. 索引存取 D. 散列存取

二、填空、判断题

1. 数据对象就是一组数据元素的集合。()
2. 在数据结构中,从逻辑上可以把数据结构分成动态结构和静态结构。()
3. 数据逻辑结构包括_____三种类型,树型结构和图状结构合称为_____。
4. 评价算法的性能从利用计算机资源角度看,主要从_____方面进行分析。
5. 线性结构中元素之间存在_____关系,树型结构中元素之间存在_____关系,图状结构中元素之间存在_____关系。

三、计算、算法题

1. 按增长率由小到大的顺序排列下列各个函数: