

全国计算机等级考试

2010版

考纲·考点·考题透解与模拟 二级公共基础知识

江家宝 编著

- ↓ 详解大纲要求，分析历年真题，统计考试频率，抓住考核重点
- ↓ 原创15套超级模拟仿真卷，考点分布、难度、题型与考试相当，热身、自测两相宜

知己知彼
百战百胜

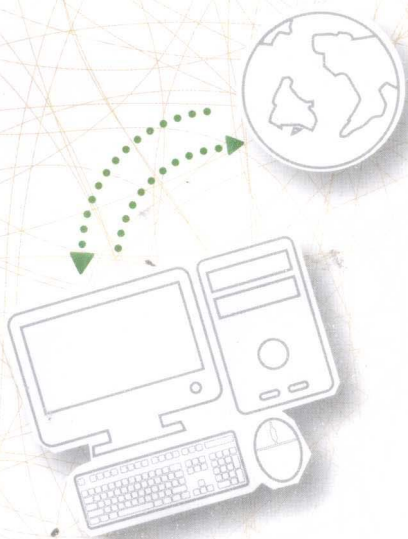
一点一练
高效实用

模拟预测
实战提高

轻松过级！



清华大学出版社



全国计算机等级考试

2010版

考纲·考点·考题透解与模拟
一级公共基础知识

江家宝 编著



清华大学出版社
北 京

内 容 简 介

本书以教育部考试中心最新颁布的《全国计算机等级考试二级公共基础知识考试大纲(2010版)》为依据,以对考生进行综合指导、全面提高应试能力为原则,深入研究近3年连续6次考试真题并结合考前辅导班教师的实际教学经验编写而成。

本书章节安排与指定教材同步。每章开始设置“考纲透解”板块,全面解读大纲要求,进行考频统计和命题方向研究;每个考点则细化为考点透解、考题透解两部分;章后为过关练习栏目。本书附有15套笔试模拟卷,并提供详细分析解答,方便读者自测。

本书抓住考纲、考点、考题3个重点,目的是让考生在较短时间内能快速提高应试能力,顺利过关。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售
版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

全国计算机等级考试考纲·考点·考题透解与模拟:2010版. 二级公共基础知识/江家宝编著.
—北京:清华大学出版社,2010.1

ISBN 978-7-302-21806-7

I. ①全… II. ①江… III. ①电子计算机—水平考试—自学参考资料 IV. ①TP3

中国版本图书馆CIP数据核字(2009)第242420号

责任编辑:夏非彼 卢 亮

责任校对:闫秀华

责任印制:孟凡玉

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市昌平环球印刷厂

装 订 者:北京国马印刷厂

经 销:全国新华书店

开 本:190×260

印 张:10.75

字 数:265千字

版 次:2010年1月第1版

印 次:2010年1月第1次印刷

印 数:1~5000

定 价:19.00元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:036091-01

前言

为了适应科学技术的发展及新形势的需要,经过专家充分论证,教育部考试中心对全国计算机等级考试的考试科目设置、考核内容和考试形式进行了调整。经过调整后的新大纲于 2008 年上半年开始实施。

为配合全国计算机等级考试的最新科目设置和考核内容的调整,现根据新大纲的要求,结合典型试题,按教育部考试中心指定教材的篇章结构,由从事全国计算机等级考试试题研究人员及在等级考试第一线从事命题研究、教学、辅导和培训的老师精心编写了《全国计算机等级考试考纲·考点·考题透解与模拟(2010 版)——二级公共基础知识》,目的是为参加全国计算机等级考试的广大考生顺利通过考试助力护航!

1. 本书特色

- ◆ 知己知彼,百战百胜:在每章开始设置一个“考纲透解”版块,指引考生明确考纲要求,真正做到知己知彼,百战百胜。
- ◆ 一点一练,高效实用:本书的章名、节名与指定教材同步,每节细分为 4 个版块。
 - 考点透解:将指定的考试内容进行浓缩,精讲考试要点、重点与难点。
 - 考题透解:精选典型试题进行深入剖析,题型丰富,讲解透彻。
 - 过关练习:针对本节知识点设计考试预测题,方便考生一点一练,巩固提高。
 - 过关练习答案:给出过关题的参考答案,便于考生检查学习效果。
- ◆ 全真模拟,实战提高:根据新大纲、新考点、新题型,自创 15 套笔试模拟预测卷,并进行分析解答,供考前热身训练。

2. 相关图书推荐

- (1) 全国计算机等级考试考纲·考点·考题透解与模拟(2010 版)
——一级 MS Office/一级 B
- (2) 全国计算机等级考试考纲·考点·考题透解与模拟(2010 版)
——二级 C 语言

(3) 全国计算机等级考试考纲·考点·考题透解与模拟(2010版)
——三级网络技术

(4) 全国计算机等级考试考纲·考点·考题透解与模拟(2010版)
——四级网络工程师

(5) 全国计算机等级考试考纲·考点·考题透解与模拟(2010版)
——四级数据库工程师

(6) 全国计算机等级考试立体化应试教程 —— 二级C语言

(7) 全国计算机等级考试立体化应试教程 —— 三级网络技术

(8) 全国计算机等级考试立体化应试教程 —— 四级网络工程师

3. 图书作者

本书由江家宝编写,此外,参与本书编写、审校、资料收集与整理工作的还有:郭龙源、李燕萍、钱博、戴仕明、许勇、高粱、耿翠红、陈玉旺、王国全、钱阳勇、王珊珊、陈海燕等。

借此机会,我们对本书所引用试题的出题老师和相关单位表示真诚的感谢。感谢出版社各位编辑对本书出版所付出的努力与大力支持。

4. 读者服务

由于作者水平有限,书中难免存在疏漏和错误之处,恳请专家和广大读者批评指正。在学习过程中,遇到疑难问题,可以通过以下方式与我们联系:booksaga@126.com, QuestionFeedBack@yahoo.com,也可以登录图格新知网站 <http://www.booksaga.com> 留言,我们将在第一时间给予答复!

编者
2009年12月

目 录

第1章 基本数据结构与算法	1
考纲透解	1
大纲要求	1
考频统计	1
命题方向	2
考点 1.1 算法***	2
考点透解	2
典型题解	5
考题透解	6
考点 1.2 数据结构的基本概念***	6
考点透解	6
典型题解	9
考题透解	10
考点 1.3 线性表及其顺序存储结构	11
考点透解	11
典型题解	13
考点 1.4 栈和队列*****	14
考点透解	14
典型题解	18
考题透解	19
考点 1.5 线性链表*	21
考点透解	21
典型题解	25
考点 1.6 树与二叉树*****	26
考点透解	26
典型题解	31
考题透解	33
考点 1.7 查找技术**	35
考点透解	35
典型题解	36
考题透解	36
考点 1.8 排序技术***	37
考点透解	37
典型题解	42
考题透解	43

一级MS Office / 一级B
二级C
二级公共基础
二级C语言
四级网络工程师
二级网络技术
四级数据库工程师
三级网络技术
四级数据库
三级网络技术
四级数据库工程师

过关练习	43
过关练习答案	45
第2章 程序设计基础	46
考纲透解	46
大纲要求	46
考频统计	46
命题方向	46
考点 2.1 程序设计方法和风格**	46
考点透解	46
典型题解	48
考题透解	48
考点 2.2 结构化程序设计***	49
考点透解	49
典型题解	50
考题透解	50
考点 2.3 面向对象的程序设计***	51
考点透解	51
典型题解	55
考题透解	55
过关练习	56
过关练习答案	57
第3章 软件工程基础	58
考纲透解	58
大纲要求	58
考频统计	58
命题方向	58
考点 3.1 软件工程基本概念***	59
考点透解	59
典型题解	64
考题透解	64
考点 3.2 结构化分析方法***	65
考点透解	65
典型题解	70
考题透解	71
考点 3.3 结构化设计方法*****	73
考点透解	73
典型题解	80
考题透解	81
考点 3.4 软件测试*****	83
考点透解	83
典型题解	87

考题透解.....	88
考点 3.5 程序的调试**.....	90
考点透解.....	90
典型题解.....	92
考题透解.....	92
过关练习.....	93
过关练习答案.....	95
第 4 章 数据库设计基础	96
考纲透解.....	96
大纲要求.....	96
考频统计.....	96
命题方向.....	96
考点 4.1 数据库系统的基本概念*****.....	97
考点透解.....	97
典型题解.....	101
考题透解.....	102
考点 4.2 数据模型*****.....	103
考点透解.....	103
典型题解.....	108
考题透解.....	109
考点 4.3 关系代数***.....	111
考点透解.....	111
典型题解.....	120
考题透解.....	121
考点 4.4 数据库设计与管理**.....	123
考点透解.....	123
典型题解.....	128
考题透解.....	129
过关练习.....	130
过关练习答案.....	132
第 5 章 模拟试卷及答案解析	133
5.1 模拟试卷一.....	133
5.2 模拟试卷二.....	134
5.3 模拟试卷三.....	135
5.4 模拟试卷四.....	136
5.5 模拟试卷五.....	137
5.6 模拟试卷六.....	138
5.7 模拟试卷七.....	139
5.8 模拟试卷八.....	140
5.9 模拟试卷九.....	141

5.10	模拟试卷十	142
5.11	模拟试卷十一	143
5.12	模拟试卷十二	144
5.13	模拟试卷十三	145
5.14	模拟试卷十四	146
5.15	模拟试卷十五	147
5.16	模拟试卷答案与解析	148
5.16.1	模拟试卷一答案解析	148
5.16.2	模拟试卷二答案解析	149
5.16.3	模拟试卷三答案解析	150
5.16.4	模拟试卷四答案解析	151
5.16.5	模拟试卷五答案解析	152
5.16.6	模拟试卷六答案解析	153
5.16.7	模拟试卷七答案解析	154
5.16.8	模拟试卷八答案解析	155
5.16.9	模拟试卷九答案解析	156
5.16.10	模拟试卷十答案解析	157
5.16.11	模拟试卷十一答案解析	158
5.16.12	模拟试卷十二答案解析	160
5.16.13	模拟试卷十三答案解析	161
5.16.14	模拟试卷十四答案解析	162
5.16.15	模拟试卷十五答案解析	163

第1章

基本数据结构与算法

考纲透解

大纲要求

- 一、算法的基本概念，算法复杂度的概念和意义（时间复杂度与空间复杂度）。
- 二、数据结构的定义，数据的逻辑结构与存储结构，数据结构的图形表示，线性结构与非线性结构的概念。
- 三、线性表的定义，线性表的顺序存储结构及其插入与删除运算。
- 四、栈和队列的定义，栈和队列的顺序存储结构及其基本运算。
- 五、线性单链表、双向链表与循环链表的结构及其基本运算。
- 六、树的基本概念，二叉树的定义及其存储结构，二叉树的前序、中序和后序遍历。
- 七、顺序查找与二分法查找算法，基本排序算法（交换类排序、选择类排序、插入类排序）。

考频统计

表 1-1 统计了最近 6 次与本章考点相关的试卷分析。

表 1-1 历年考题知识点分布统计表

命题范围	年、月	考核要点	命题范围	年、月	考核要点
算法	2009.9	算法的空间复杂度	树与二叉树	2009.9	二叉树结点的计算
	2008.4	算法的特性		2009.3	二叉树结点的计算
	2007.4	有关算法的概念		2009.3	二叉树的应用
数据结构的基本概念	2009.9	非线性数据结构		2008.9	二叉树的遍历
	2008.9	顺序、链式存储结构		2008.4	满二叉树的性质
	2007.9	数据结构		2007.9	二叉树的性质
	2007.9	数据的存储结构		2007.9	二叉树的中序遍历
栈和队列	2009.9	栈的基本概念		2007.4	二叉树的前序遍历
	2009.9	循环队列的概念		2007.4	二叉树的性质
	2009.3	栈和队列的概念		2007.4	树与二叉树
	2009.3	栈的基本运算	查找、排序技术	2009.3	排序算法的时间复杂度
	2008.9	栈的基本运算		2008.9	二分法查找
	2008.9	循环队列的概念		2008.4	排序算法的性能比较
	2008.4	栈的定义		2007.9	冒泡排序
	2008.4	循环队列的运算			
	2007.9	循环队列的概念			
	2007.4	队列的概念			

命题方向

1. 本章在最近几次考试中题量都在 5 题左右，其中 2007 年 9 月选择题 4 题、填空题 2 题，其后几次都是五道题。题量占公共基础知识相关试题总量的 1/3 左右。
2. 算法基本概念、算法复杂度、数据的逻辑结构和存储结构、线性结构与非线性结构、栈及其基本运算、队列、线性链表、二叉树的基本概念及其特性、顺序查找、二分法查找、交换类排序法是考核的重点。其中算法、栈和队列、二叉树是重中之重。
3. 线性表的基本概念、线性表顺序存储的特点及运算、线性链表基本运算、循环链表及其基本运算、树的基本概念和特征、插入类排序法、选择类排序法近几次考题中没出现。

考点1.1 算法***

考点透解

一、算法的基本概念

算法（Algorithm）是指为解决某个特定问题而采取的确定且有限的步骤的一种描述，它是指令的有限序列，使得给定类型的问题通过有限的指令序列、在有限的时间内被求解。其中每一条指令表示表示一个或多个操作。

1. 算法的基本特性

（1）可行性

是指算法中指定的操作都可以通过基本运算执行有限的次数后实现。在设计一个算法时，必须要考虑它的可行性，否则是不会得到满意结果的。

（2）确定性

是指算法中每一步骤都必须是有明确定义的，不允许有模棱两可的解释，也不允许有多义性。这一性质也反映了算法与数学公式的明显差别。在解决实际问题时，可能会出现这样的情况：针对某种特殊问题，数学公式是正确的，但按此数学公式设计的计算过程可能会使计算机无所适从。这是因为根据数学公式设计的计算过程只考虑了正常使用情况，而当出现异常情况时，此计算过程就不能适应了。

（3）有穷性

是指算法必须能在有限时间内做完，即算法必须能在执行有限的步骤之后终止。数学中的无穷级数，在实际计算时只能取有限项，即计算无穷级数值的过程只能是有穷的。因此，一个数的无穷级数表示只是一个计算公式，而根据精度要求确定的计算过程才是有穷的算法。

算法的有穷性还应包括合理的执行时间的含义。因为，如果一个算法需要执行千万年，显然失去了实用价值。

（4）拥有足够的情报

一个算法的执行结果总是与输入的初始数据有关，它可以有多个输入也可以不要输入，而必须有一个或多个输出，不同的输入将会有不同的输出结果。通常情况下，当情报足够时，算法才是有效的，当情报不够时，算法可能无效。



综上所述,所谓算法,是一组严谨地定义运算顺序的规则,并且每一个规则都是有效的,且是明确的,此顺序将在有限的次数下终止。

2. 算法的基本要素

一个算法通常由两种基本要素构成:一是对数据对象的运算和操作,二是算法的控制结构。

(1) 算法中对数据对象的运算和操作

每个算法实际上是按照解题要求从环境能进行的所有操作中选择合适的操作所组成的一组指令序列。因此,计算机算法就是计算机能处理的操作所组成的指令序列。

通常,计算机可以执行的基本操作是以指令形式描述的。一个计算机系统能执行的所有指令的集合,称为该计算机系统的指令系统。计算机程序就是按解题要求从计算机指令系统中选择合适的指令所组成的指令序列。在一般的计算机系统中,基本的运算和操作包括四类:

- 算术运算:主要包括加、减、乘、除等运算;
- 逻辑运算:主要包括“与”、“或”、“非”等运算;
- 关系运算:主要包括“大于”、“小于”、“等于”和“不等于”等运算;
- 数据传输:主要包括赋值、输入和输出等操作。

(2) 算法的控制结构

算法中各操作之间的执行顺序称为算法的控制结构。算法的功能不仅取决于所选用的操作,而且还与各操作之间的执行顺序有关。

算法的控制结构给出了算法的基本框架,它不仅决定了算法中各操作之间的执行顺序,更直接反映了算法的设计是否符合结构化设计原则。算法通常可以用顺序、选择和循环三种基本控制结构组合而成,而算法的描述工具通常有传统流程图、N-S 结构化流程图、伪码描述语言等。

3. 算法设计的基本方法

(1) 列举法

它是计算机算法中的一个基础算法,此方法是根据提出的问题,列举出所有可能出现的情况,并用问题中给定的条件检验哪些是需要的,哪些是不需要的。因此,列举法常用于解决“是否存在”或“有多少种可能”等类型的问题。

列举法的特点是算法比较简单。但当列举的可能情况较多时,执行列举算法的工作量将会很大。因此,在用列举法设计算法时,使方案优化,尽量减少运算工作量,是应该重点注意的。

列举算法虽然是一种比较笨拙而原始的方法,其运算量比较大,但在有些实际问题中(如查找、搜索、寻找捷径等问题),局部使用列举法却是很有效的,因此,列举算法是计算机算法中的一个基础算法。

(2) 归纳法

归纳法的基本思想是,通过列举少量的简单而又特殊的情况,经过分析,总结归纳出一般性的结论。从本质上讲,归纳就是通过观察一些简单而特殊的情况,最后总结出一般性的结论。

显然,归纳法要比列举法更能反映问题的本质,并且可以解决列举量为无限的问题。但是,从一个实际问题中总结归纳出一般的关系,并不是一件容易的事情,尤其是要归纳出一个数学模型更为困难。



（3）递推

从给定的或已知的初始条件出发，逐步推出所要求的各中间结果和最后结果。其中初始条件或是问题本身已经给定，或是通过对问题的分析与化简而确定。递推关系通常是归纳的结果，递推法其本质也属于归纳法。此方法还经常在数值计算中使用。

（4）递归

在解决复杂问题或问题的规模比较大时，将问题逐层分解，最后归结为一些简单的问题，逐个解决这些简单的问题，然后再沿原来的逆过程逐步进行综合，这就是递归基本思想。由此可以看出，递归的基础也是归纳。在工程实际中，有许多问题就是用递归来定义的，数学中的许多函数也是用递归来定义的。递归在可计算性理论和算法设计中占有很重要的地位。

（5）减半递推技术

“减半”，是指将问题的规模减半，而问题的性质不变；“递推”，是指重复减半的过程。这是工程上常用的分治法的一种，用来解决复杂程度与规模有密切联系的实际问题。

（6）回溯法

在有些实际问题中有时很难归纳出简单的公式或直观的求解步骤，也不能无限制地列举，一种有效的解决此类问题的方法是尝试。通过分析问题，找出一个解决的线索，沿此线索逐步试探，若试探成功，就得到问题的解，否则，就逐步回退，换线索再逐步试探。回溯法在处理复杂数据结构方面有着广泛的应用。

4. 算法设计的要求

- （1）正确性：算法的执行结果应当满足预先规定的功能和性能要求。
- （2）健壮性：当输入不合法数据时，应能作适当处理，不致引起严重后果。
- （3）可读性：一个算法应当思路清晰、层次分明、简单明了、易读易懂。
- （4）高效性：有效使用存储空间和有较高的时间效率。

二、算法复杂度

算法的复杂度主要包括时间复杂度和空间复杂度。

1. 时间复杂度

所谓算法的时间复杂度（Time Complexity）是指执行算法所需要的计算工作量。

为了能够比较客观地反映出一个算法的效率，在度量一个算法的工作量时，不仅与所使用的计算机、程序设计语言以及程序编制者无关，而且还应该与算法实现过程中的许多细节无关。为此可以用算法在执行过程中所需基本运算的执行次数来度量算法的工作量。基本运算反映了算法运算的主要特征，因此，用基本运算次数来度量算法工作量是客观的也是实际可行的，有利于比较同一问题的几种算法的优劣。例如，在考虑两个矩阵相乘时，可以将两个实数之间的乘法运算作为基本运算，而对于所用的加法（或减法）运算忽略不计。又如，当需要在一个表中进行查找时，可以将两个元素之间的比较作为基本运算。

算法所执行的基本运算次数还与问题的规模有关。例如，两个 20 阶矩阵相乘与两个 10 阶矩阵相乘，所需要的基本运算次数显然是不同的，前者需要更多的运算次数。因此，在分析算法的工作量时，还必须对问题的规模进行度量。

综上所述，算法的工作量用算法所执行的基本运算次数来度量，而算法所执行的基本运算次数



是问题规模的函数，即：

$$\text{算法的工作量} = f(n)$$

通常记作：

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

它表示随着问题规模 n 的增大，算法执行时间的增长率和 $f(n)$ 的增长率相同，称为算法的时间复杂度。例如，一个程序的实际执行时间为 $T(n) = 2.7n^3 + 3.8n^2 + 5.3$ ，则 $T(n) = O(n^3)$ 。通常用 $O(1)$ 表示常数计算时间，常见的时间复杂度有：

$$O(1) < O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < O(n^3) < O(2^n)$$

2. 空间复杂度

一个算法的空间复杂度就是执行这个算法所需要的内存空间。

一个算法所占用的存储空间包括算法程序所占的空间、输入的初始数据所占的存储空间以及算法执行过程中所需要的额外空间。其中额外空间包括算法程序执行过程中的工作单元以及某种数据结构所需要的附加存储空间（例如，在链式结构中，除了要存储数据本身外，还需要存储链接信息）。如果额外空间量相对于问题规模来说是常数，则称该算法是原地（in place）工作的。

在许多实际的问题中，为了减少算法的内存空间，一般采取压缩存储技术，以便尽量减少不必要的额外空间。

典型题解

【例题 1】下列叙述中正确的是_____。

- A) 一个算法的空间复杂度大，则其时间复杂度也必定大
- B) 一个算法的空间复杂度大，则其时间复杂度必定小
- C) 一个算法的时间复杂度大，则其空间复杂度必定小
- D) 上述三种说法都不对

解 析：算法在运行过程中所需要的辅助存储空间的大小称为算法空间复杂度；算法的时间复杂度是执行该算法所需要的计算工作量，即算法执行过程中所需要的基本运算次数。为了能够比较客观地反映出算法的效率，在度量一个算法的工作量时，与所使用的计算机、程序设计语言以及程序编制者无关，而且还与算法实现过程中的许多细节无关。但可以用算法在执行过程所需基本运算的执行次数来度量算法的工作量。

答 案：D

【例题 2】算法的时间复杂度是指_____。

- A) 执行算法程序所需要的时间
- B) 算法程序的长度
- C) 算法执行过程中所需要的基本运算次数
- D) 算法程序中的指令条数

解 析：所谓算法的时间复杂度，是指执行算法所需要的计算工作量。算法的工作量用算法所执行的基本运算次数（问题规模的函数）来度量，即

$$\text{算法的工作量} = f(n)$$

其中 n 是问题的规模。



答案：C

【例题3】一个算法通常由两种基本要素组成：一是对数据对象的运算和操作，二是算法的_____。

分析：一个算法通常由两种基本要素组成：一是对数据对象的运算和操作；二是算法的控制结构。

答案：控制结构

考题透解

【试题1】算法的空间复杂度是指_____。(2009.09)

- A) 算法在执行过程中所需要的计算机存储空间
- B) 算法所处理的数据量
- C) 算法程序中的语句或指令条数
- D) 算法在执行过程中所需要的临时工作单元数

解析：一般来说，一个算法的空间复杂度是指执行这个算法所需的内存空间。一个算法所占用的存储空间包括算法程序所占的空间，输入的初始数据所占的存储空间，以及算法执行过程中所需要的额外空间。

答案：A

【试题2】算法的有穷性是指_____。(2008.04)

- A) 算法程序的运行时间是有限的
- B) 算法程序所处理的数据量是有限的
- C) 算法程序的长度是有限的
- D) 算法只能被有限的用户使用

解析：算法的基本特征包括可行性、确定性、有穷性、拥有足够的情报，其中算法的有穷性是指算法必须能在有限的时间内做完执行有限个步骤之后终止，即算法程序的运行时间是有限的。

答案：A

【试题3】下列叙述中正确的是_____。(2007.04)

- A) 算法的效率只与问题的规模有关，而与数据的存储结构无关
- B) 算法的时间复杂度是指执行算法所需要的计算工作量
- C) 数据的逻辑结构与存储结构是一一对应的
- D) 算法的时间复杂度与空间复杂度一定相关

解析：算法的复杂度主要包括时间复杂度和空间复杂度。通常用时间复杂度和空间复杂度来衡量算法效率，算法的时间复杂度就是执行该算法所需要的计算工作量；算法所执行的基本运算次数与问题的规模有关。而一个算法的空间复杂度，就是执行该算法所需要的内存空间；一般来说，一种数据的逻辑结构根据需要可以表示成多种存储结构。

答案：B

■ ■ ■ 考点1.2 数据结构的基本概念*** ■ ■ ■

考点透解

一、什么是数据结构

1. 数据结构的定义

数据结构（Data Structure）是指反映数据元素之间关系的数据元素集合的表示。数据结构作为



计算机的一门学科，主要研究和讨论以下三个方面的问题：

- (1) 数据集中各数据元素之间所固有的逻辑关系，即数据的逻辑结构；
- (2) 各数据元素在计算机中的存储关系，即数据的存储结构；
- (3) 对各种数据结构进行的运算。

研究以上问题的主要目的是为了提高数据处理的效率。提高数据处理的效率，主要包括两个方面：

- (1) 提高数据处理的速度；
- (2) 尽量节省数据处理过程中所占用的计算机存储空间。

要理解数据结构，还需要理解以下基本概念：

数据(Data)是信息的载体，它能够被计算机识别、存储和加工处理。计算机科学中，所谓数据就是计算机加工处理的对象，它可以是数值数据，也可以是非数值数据。

数据元素(Data Element)是数据的基本单位。在不同的条件下，数据元素又可称为元素、结点、顶点、记录等。

数据对象(Data Object)是具有相同性质的数据元素的集合。在某个具体问题中，数据元素都具有相同的性质(元素值不一定相等)，属于同一数据对象，数据元素是数据元素类的一个实例。

2. 数据的逻辑结构

在任何问题中，数据元素之间都不会是孤立的，在它们之间都存在着这样或那样的关系，这种数据元素之间的关系称为结构。一个数据结构应包含两个方面的信息：

- (1) 表示数据元素的信息；
- (2) 表示各数据元素之间的前后件关系。

其中数据元素之间的前后件关系是指它们之间的逻辑关系，而上述的数据结构实际上是数据的逻辑结构，它是对数据元素之间的逻辑关系的描述。所谓数据的逻辑结构，是指反映数据元素之间逻辑关系的数据结构。根据数据元素之间关系的不同特性，通常有4类基本的逻辑结构，如图1-1所示。

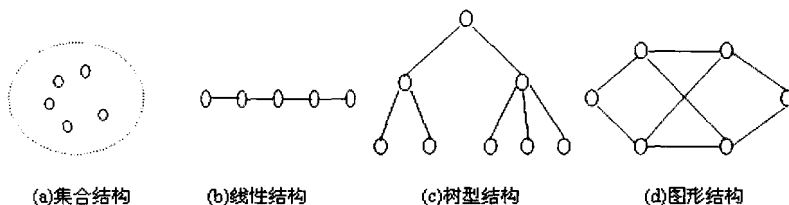


图 1-1 数据的逻辑结构

- (1) 集合结构：在集合结构中，数据元素间的关系是“属于同一个集合”，集合是元素关系极为松散的一种结构。
- (2) 线性结构：该结构的数据元素之间存在着一对一的关系。
- (3) 树型结构：数据元素之间存在着一对多的关系。
- (4) 图形结构：数据元素之间存在着多对多的关系，图形结构也称作网状结构。



3. 数据的存储结构

数据结构包括数据的逻辑结构和数据的物理结构。数据的逻辑结构可以看作是从具体问题抽象出来的数学模型，它与数据的存储无关。我们研究数据结构的目的是为了在计算机中实现对它的操作，为此还需要研究如何在计算机中表示一个数据结构。数据的逻辑结构在计算机存储空间中的存放形式称为数据的物理结构，或称数据的存储结构。

由于数据元素在计算机存储空间中的位置关系可能与逻辑关系不同，因此，为了表示存放在计算机存储空间中的各数据元素之间的逻辑关系（即前后件关系），在数据的存储结构中，不仅要存放各数据元素的信息，还需要存放各数据元素之间的前后件关系的信息。

一般来说，一种数据的逻辑结构根据需要可以表示成多种存储结构。

（1）顺序存储方法：是把逻辑上相邻的元素存储在物理位置相邻的存储单元中，由此得到的存储表示称为顺序存储结构。顺序存储结构是一种最基本的存储表示方法，通常借助于程序设计语言中的数组来实现。

（2）链式存储方法：对逻辑上相邻的元素不要求其物理位置相邻，元素间的逻辑关系通过附设的指针字段来表示，由此得到的存储表示称为链式存储结构，链式存储结构通常借助于程序设计语言中的指针类型来实现。

（3）除了通常采用的顺序存储方法和链式存储方法外，有时为了查找的方便还采用索引存储方法和散列存储方法。

二、数据结构的图形表示

1. 二元关系表示

由前面的叙述可以知道，数据的逻辑结构有两个要素：一是数据元素的集合，通常记为 D ；二是 D 上的关系，反映 D 中数据元素之间的前后件关系，通常记为 R 。一个数据结构可以表示成：

$$B=(D, R)$$

其中 B 表示数据结构，为了反映 D 中各数据元素之间的前后件关系，一般用二元组表示。例如，家庭成员数据结构可以表示成

$$B=(D, R)$$

$$D=\{\text{父亲, 儿子, 女儿}\}$$

$$R=\{(\text{父亲, 儿子}), (\text{父亲, 女儿})\}$$

2. 图形表示

在数据结构的图形表示中，对于集合 D 中的每个数据元素用标有元素值的方框表示，通常称为数据结点，并简称为结点；对于关系 R 中的每个二元组，用一条有向线段从前件结点（或称为前驱结点）指向后件结点（或称为后继结点）。例如，上例中的家庭成员数据结构也可以用图 1-2 所示。

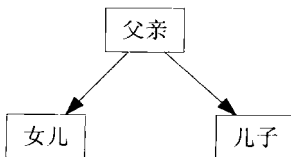


图 1-2 数据结构的图形表示