



天勤计算机考研高分笔记系列  
天勤论坛 [www.csbjji.com](http://www.csbjji.com)

第8版

计算机考研大纲起草者

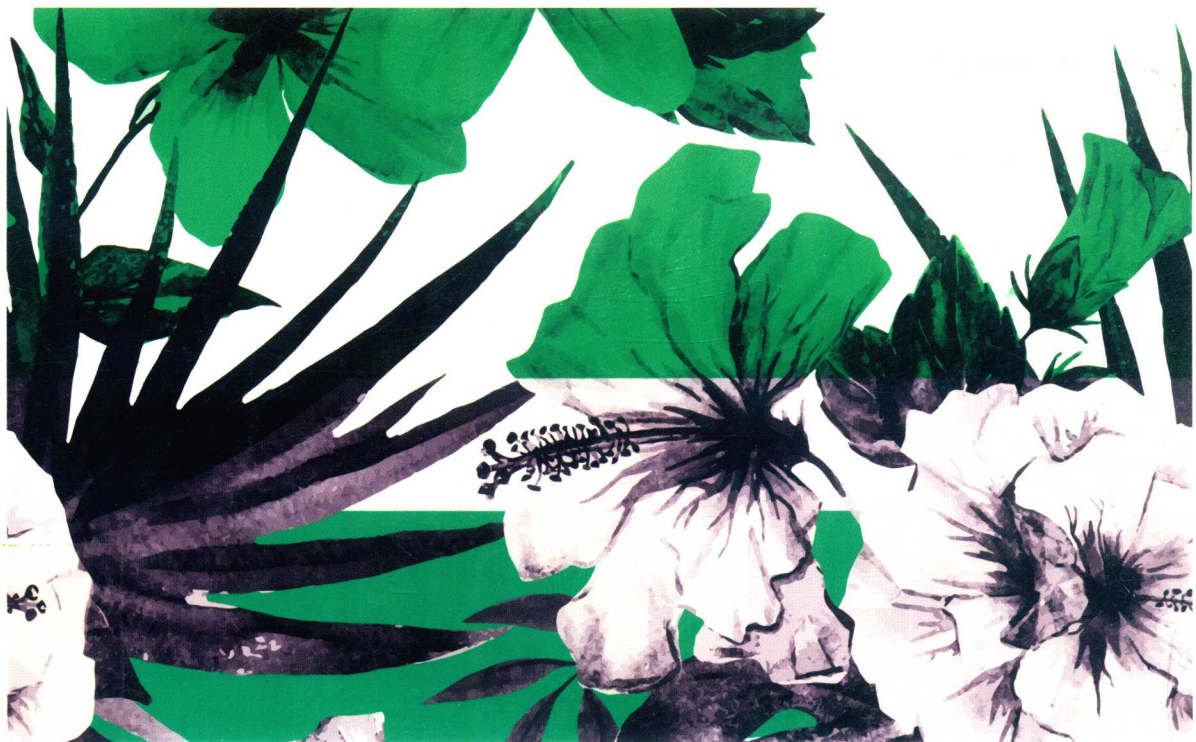
殷人昆

鼎力推荐

2020版

率辉 主编

# 数据结构 高分笔记



本书互动更新平台：



## 重要更新

- ◎修正了老版本中发现的错误
- ◎优化了部分知识点的讲解
- ◎根据考研形势变动调整了一些题目以及讲解
- ◎开设了辅导专栏作为增值服务（扫左侧二维码可找到专栏入口）
- ◎辅导专栏中即将添加对408科目的专门讲解（知识点和模拟题讲解）

公众号：shuaihui\_ds

 机械工业出版社  
CHINA MACHINE PRESS

天勤计算机考研高分笔记系列

# 2020 版数据结构高分笔记

第 8 版

率 辉 主编



机械工业出版社

本书针对近几年全国计算机学科专业综合考试大纲的“数据结构”部分进行了深入解读，以一种独创的方式对考试大纲中的知识点进行了讲解，即从考生的视角剖析知识难点；以通俗易懂的语言取代晦涩难懂的专业术语；以成功考生的亲身经历指引复习方向；以风趣幽默的笔触缓解考研压力。读者对书中的知识点讲解有任何疑问都可与作者进行在线互动，为考生解决复习中的疑难点，提高考生的复习效率。

根据计算机专业研究生入学考试形势的变化（逐渐实行非统考），书中对大量非统考知识点进行了讲解，使本书所包含的知识点除覆盖统考大纲的所有内容外，还包括了各自命题高校所要求的知识点。

本书可作为参加计算机专业研究生入学考试的复习指导用书（包括统考和非统考），也可作为全国各大高校计算机专业或非计算机专业的学生学习“数据结构”课程的辅导用书。

（编辑邮箱：jinacmp@163.com）

## 图书在版编目（CIP）数据

2020 版数据结构高分笔记 / 率辉主编. —8 版. —北京：机械工业出版社，2018.12

（天勤计算机考研高分笔记系列）

ISBN 978-7-111-61559-0

I. ①2… II. ①率… III. ①数据结构—研究生—入学考试—自学参考资料 IV. ①TP311.12

中国版本图书馆 CIP 数据核字（2018）第 278946 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

策划编辑：吉 玲 责任编辑：吉 玲 刘丽敏

责任校对：吉 玲

责任印制：孙 炜

保定市中华美凯印刷有限公司印刷

2019 年 1 月第 8 版第 1 次印刷

184mm×260mm·21 印张·651 千字

标准书号：ISBN 978-7-111-61559-0

定价：55.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

服务咨询热线：010-88361066

读者购书热线：010-68326294

010-88379203

封面无防伪标均为盗版

网络服务

机工官网：[www.cmpbook.com](http://www.cmpbook.com)

机工官博：[weibo.com/cmp1952](http://weibo.com/cmp1952)

金书网：[www.golden-book.com](http://www.golden-book.com)

教育服务网：[www.cmpedu.com](http://www.cmpedu.com)

# 序

《2020 版数据结构高分笔记》《2020 版计算机组成原理高分笔记》《2020 版操作系统高分笔记》《2020 版计算机网络高分笔记》等辅导教材问世了，这对于有志考研的同学是一大幸事。“他山之石，可以攻玉”，参考一下亲身经历过考研并取得优秀成绩的师兄们的经验，必定有益于对考研知识点的复习和掌握。

能够考上研究生，这是无数考生的追求，能够以优异的成绩考上名牌大学的全国数一数二的计算机或软件工程学科的研究生，更是许多考生的梦想。如何学习或复习相关课程，如何打好扎实的理论基础、练好过硬的实践本领，如何抓住要害，掌握主要的知识点并获得考试的经验，先行者已经给考生们带路了。“高分笔记”的作者们在认真总结了考研体会，整理了考研的备战经验，参考了多种考研专业教材后，精心编写了本套系列辅导书。

“天勤计算机考研高分笔记系列”辅导教材的特点是：

◇ 贴近考生。作者们都亲身经历了考研，他们的视角与以往的辅导教材不同，是从复习考研的学生的立场理解教材的知识点——哪些地方理解有困难，哪些地方需要整理思路，叙述处处替考生着想，有很好的引导作用。

◇ 重点突出。作者们在复习过程中做了大量习题，并经历了考研的严峻考验，对重要的知识点和考试出现频率高的题型都了如指掌。因此，在复习内容的取舍上进行了精细地考虑，使得读者可以抓住重点，有效地复习。

◇ 分析透彻。作者们在复习过程中对主要辅导教材的许多习题都进行了深入分析并亲自解答过，对重要知识点进行了总结，因此，解题思路明确，叙述条理清晰，问题求解的步骤详细，对结果的分析透彻，不但可以扩展考生的思路，还有助于考生举一反三。

计算机专业综合基础考试已经考过 11 年，今后考试的走向如何，可能是考生最关心的问题。我想，这要从考试命题的规则入手来讨论。

以清华大学为例，学校把研究生入学考试定性为选拔性考试。研究生入学考试试题主要测试考生对本学科的专业基础知识、基本理论和基本技能掌握的程度。因此，出题范围不应超出本科教学大纲和硕士生培养目标，并尽可能覆盖一级学科的知识面，一般会使本学科、本专业本科毕业的优秀考生取得及格以上的成绩。

实际上，全国计算机专业研究生入学联考的命题原则也是如此，各学科的重点知识点都是命题的重点。一般知识要考，比较难的知识（较深难度的知识）也要考。通过对 2009 年以来几年的考试进行分析可知，考试的出题范围基本符合考试大纲，都覆盖到各大知识点，但题量有所侧重。因此，考生一开始不要抱侥幸的心理去押题，应踏踏实实读好书，认认真真做好复习题，仔仔细细归纳问题解决的思路，夯实基础，增长本事，然后再考虑重点复习。这里有几条规律可供参考：

◇ 出过题的知识点还会有题，出题频率高的知识点，今后出题的可能性也大。

◇ 选择题大部分题目涉及基本概念，主要考查对各个知识点的定义和特点的理解，个别选择题会涉及相应延伸的概念。



◇ 综合应用题分为两部分：简做题和设计题。简做题的重点在设计和计算；设计题的重点在算法、实验或综合应用。

常言道：“学习不怕根基浅，只要迈步总不迟。”只要大家努力了，收获总会有的。

清华大学 殷人昆

# 辅导专栏简介

## ◆ 考研数据结构知识点讲解专栏

本书配有作者亲自制作的讲解专栏“率辉考研数据结构”，作为考生的增值服务，专栏内容为视频、音频和配有视频或者音频的图文消息。读者可通过下边的二维码找到专栏入口。



## ◆ 本书习题部分将在专栏中更新

“率辉考研数据结构”专栏在 2017 年已将高频考研知识点讲解部分更新完毕，2018 年更新了部分习题，2019 年将继续更新习题讲解。

## ◆ 其他辅导专栏（计划中）

- 408 科目知识点精讲班专栏，包含知识点讲解和模拟卷讲解；
- 408 其余科目的讲解专栏，包含知识点讲解和真题讲解。

## ◆ 本书实时更新服务

从下边的二维码可以找到本书实时更新服务入口，内容包括勘误、调整后的知识点讲解等。





# 前 言

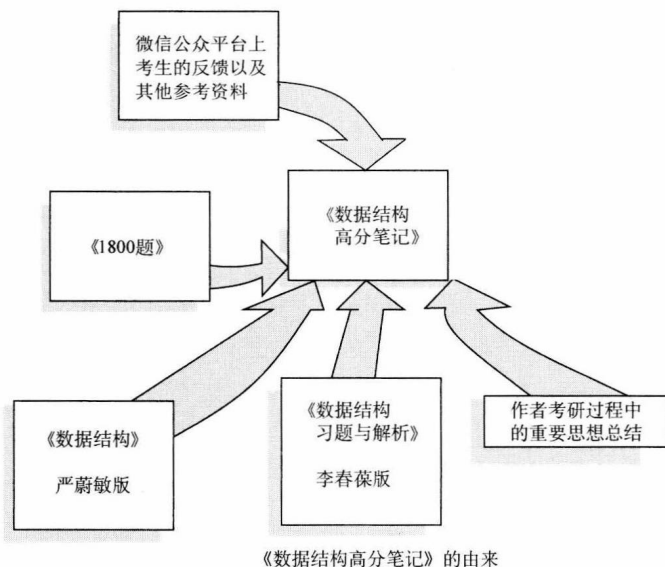
## 高分笔记系列书籍简介

高分笔记系列书籍包括《数据结构高分笔记》《组成原理高分笔记》《操作系统高分笔记》《计算机网络高分笔记》等，是一套针对计算机考研的辅导书。它们 2010 年夏天诞生于一群考生之手，其写作风格突出表现为：以学生的视角剖析知识难点；以通俗易懂的语言取代晦涩难懂的专业术语；以成功考生的亲身经历指引复习方向；以风趣幽默的笔触缓解考研压力。相信高分笔记系列书籍带给考生的将是更高效、更明确、更轻松、更愉快的复习过程。

## 数据结构高分笔记简介

众所周知，在计算机统考的四门专业课中，最难拿高分的就是数据结构。但是这门课本身的难度并不是考生最大的障碍，真正的障碍在于考生不能独自把握复习方向和考试范围。也许有学生要问，我们不是有大纲吗？照着大纲去复习不就可以了吗？表面上看是这样的，但是当你真正开始复习的时候就会发现，其实大纲只给了考生一个大致范围，有很多地方是模糊的，这些模糊的地方可能就是让你纠结的地方。比如大纲里对于栈和队列的考查中有这么一条：“栈和队列的应用。”这个知识点就讲得很模糊，因为只要涉及栈和队列的地方，都是其应用的范畴，这时考生该怎么办呢？于是把所有的希望寄托于参考书，希望参考书能帮助他们理解大纲的意图。参考书分为两种：一是课本，二是与课本配套的辅导书。对于课本，考生用得最多的就是严蔚敏老师编写的“严版”《数据结构》。因为这本书的内容非常丰富，如果能把这本书中考试大纲要求的章节理解透彻，参加考研就没有任何问题，但是这个过程是漫长的，除非本科阶段就学得非常好。计算机统考后，专业课四门加上公共课三门，一共是七门，绝大多数考生复习的时间一般也就六个月，而数据结构的复习需要占用多少时间，这点大家都很清楚。要在这么短的时间内掌握“严版”《数据结构》中考纲要求的知识点，基本上是不可能的，这就需要一本辅导书来依照大纲从课本中总结出考纲要求的知识点，才能使得考生在短时间内达到研究生考试的要求。市面上的参考书有两种：一种是四合一的辅导书，另一种是分册的。比如网上流行的《1800 题》及其第 2 版，此书中题目极多，并且有很多老式的考研题，有些算法设计题的答案是用 Pascal 语言写的。这本书中的题目一般考生全做基本上是不可能的，挑着做又会把时间浪费在选题上。不可否认，这本书确实是一本非常好的题库，但是考生直接拿来用作考研辅导书却不太合适。在这种情况下，就需要有一本优质的完全针对新大纲的辅导书出现，这就是高分笔记产生的原因。

接下来详细介绍一下这本辅导书的写作过程，请看下图：



图中所涉及的书都是大家很熟悉的。当年这些书编者都买了，花了很大心思才从中找出在考研战场上真正有用的东西。比如《1800题》，里边既有好题，又有废题，相信很多人都希望有人能从中去掉重复的题目，选出大纲要求的题目，并能把解答写得更通俗易懂些，而现在编者所做的工作就是从这1800道题中选出大纲要求的题目，并且修正部分解答，使其更容易理解。其次是“严版”《数据结构》，此书写得很严谨，语言表述非常专业，但对于基础稍差的学生来说读起来十分费力，要很长时间才能适应这本书的写作风格。如果有一本辅导书能把那些复杂程序的执行过程、拗口的专业术语、令人头大的符号翻译成容易理解的语言，就可以节省考生很多时间，因此，编者所做的事情就是根据自己复习的经验，以及对这本书的理解，把其中考试不需要的内容删掉，把需要的内容改造成一般考生容易接受的形式。对于李春葆老师的《数据结构习题与解析》，也做了类似的处理，并且在这本书中穿插讲解了一些考试大纲中没有明文规定，但是很多算法题目中大量用到的算法设计思想，来帮助大家提高解算法设计题的能力，比如搜索（打印图中两结点之间的所有路径）、分治法（二分法排序、求树的深度等）等算法思想。因此，相信本书会给读者的考研复习带来很大的帮助。

另外，本书配有微信公众号来收集读者的反馈，这也是本书不断更新完善的重要途径，即根据考生最需要的内容来作为调整讲解的依据。

#### 本书特点：

- (1) 精心挑选出适合考研的习题，并配上通俗易懂的答案，供读者自测和练习。
- (2) 总结出考研必备知识点，并且帮读者把其中过于专业、过于严谨的表述翻译成通俗易懂的语言。
- (3) 针对近年数据结构大题的出题风格（比如算法设计题目中的三段式题目：①表述算法思想；②写出算法描述；③计算算法的时间和空间复杂度），设计了独特的真题仿造部分，让读者在复习的过程中逐渐适应不同类型的题目。

参加本书编写的人员还有：章露捷，刘建萍，施伟，刘炳瑞，刘菁，孙琪，金苍宏，





蔡明婉, 吴雪霞, 孙建兴, 张继建, 胡素素, 邱纪虎, 率方杰, 李玉兰, 率秀颂, 刘忠艳, 赵建, 张兆红, 张来恩, 张险峰, 殷凤岭, 于雪友, 周桂芝, 张玉奎, 李亚静, 周莉, 李娅, 刘梅, 殷晓红, 李艳红, 王中静, 张洪英, 王艳红, 周晓红, 杨秋侠, 秦凤利, 叶萍。

编 者

# 目 录

序	
辅导专栏简介	
前言	
第 1 章 绪论	1
本章概略	1
1.1 针对考研数据结构的代码书写规范以及 C 与 C++ 语言基础	1
1.1.1 考研综合应用题中算法设计部分的代码书写规范	1
1.1.2 考研中的 C 与 C++ 语言基础	3
1.2 算法的时间复杂度与空间复杂度分析基础	12
1.2.1 考研中的算法时间复杂度分析	12
1.2.2 例题选讲	12
1.2.3 考研中的算法空间复杂度分析	14
1.3 数据结构和算法的基本概念	14
1.3.1 数据结构的基本概念	14
1.3.2 算法的基本概念	15
习题	16
习题答案	17
第 2 章 线性表	20
大纲要求	20
考点与要点分析	20
核心考点	20
基础要点	20
知识点讲解	20
2.1 线性表的基本概念与实现	20
2.2 线性表的结构体定义和基本操作	24
2.2.1 线性表的结构体定义	24
2.2.2 顺序表的操作	26
2.2.3 单链表的操作	29
2.2.4 双链表的操作	33
2.2.5 循环链表的操作	35
▲ 真题仿造	35
真题仿造答案与解析	36
习题+真题精选	37
习题答案+真题精选答案	41
第 3 章 栈和队列	55
大纲要求	55
考点与要点分析	55
核心考点	55
基础要点	55
知识点讲解	55



3.1	栈和队列的基本概念	55
3.1.1	栈的基本概念	55
3.1.2	队列的基本概念	56
3.2	栈和队列的存储结构、算法与应用	56
3.2.1	本章所涉及的结构体定义	56
3.2.2	顺序栈	57
3.2.3	链栈	59
3.2.4	栈的应用	60
3.2.5	顺序队	64
3.2.6	链队	66
3.3	抽象数据类型	69
▲	真题仿造	71
	真题仿造答案与解析	71
	习题+真题精选	74
	习题答案+真题精选答案	79
<b>第4章</b>	<b>串</b>	<b>91</b>
	知识点讲解	91
4.1	串数据类型的定义	91
4.1.1	串的定义	91
4.1.2	串的存储结构	91
4.1.3	串的基本操作	92
4.2	串的模式匹配算法	95
4.2.1	简单模式匹配算法	95
4.2.2	KMP 算法	96
4.2.3	KMP 算法的改进	99
	习题	102
	习题答案	103
<b>第5章</b>	<b>数组、矩阵与广义表</b>	<b>113</b>
	知识点讲解	113
5.1	数组	113
5.2	矩阵的压缩存储	114
5.2.1	矩阵	114
5.2.2	特殊矩阵和稀疏矩阵	115
5.3	广义表	121
	习题	122
	习题答案	123
<b>第6章</b>	<b>树与二叉树</b>	<b>132</b>
	大纲要求	132
	考点与要点分析	132
	核心考点	132
	基础要点	132
	知识点讲解	132
6.1	树的基本概念	132
6.1.1	树的定义	132



6.1.2	树的基本术语	132
6.1.3	树的存储结构	133
6.2	二叉树	134
6.2.1	二叉树的定义	134
6.2.2	二叉树的主要性质	135
6.2.3	二叉树的存储结构	137
6.2.4	二叉树的遍历算法	137
6.2.5	二叉树遍历算法的改进	146
6.3	树和森林与二叉树的互相转换	155
6.3.1	树转换为二叉树	155
6.3.2	二叉树转换为树	156
6.3.3	森林转换为二叉树	156
6.3.4	二叉树转换为森林	157
6.3.5	树和森林的遍历	157
6.4	树与二叉树的应用	158
6.4.1	二叉排序树与平衡二叉树	158
6.4.2	赫夫曼树和赫夫曼编码	159
▲	真题仿造	162
	真题仿造答案与解析	162
	习题+真题精选	163
	习题答案+真题精选答案	168
第7章	图	183
	大纲要求	183
	考点与要点分析	183
	核心考点	183
	基础要点	183
	知识点讲解	183
7.1	图的基本概念	183
7.2	图的存储结构	184
7.2.1	邻接矩阵	185
7.2.2	邻接表	186
7.2.3	邻接多重表	187
7.3	图的遍历算法操作	188
7.3.1	深度优先搜索遍历	188
7.3.2	广度优先搜索遍历	189
7.3.3	例题选讲	190
7.4	最小(代价)生成树	193
7.4.1	普里姆算法和克鲁斯卡尔算法	193
7.4.2	例题选讲	197
7.5	最短路径	198
7.5.1	迪杰斯特拉算法	198
7.5.2	弗洛伊德算法	204
7.6	拓扑排序	207
7.6.1	AOV网	207



7.6.2	拓扑排序核心算法	207
7.6.3	例题选讲	209
7.7	关键路径	209
7.7.1	AOE 网	209
7.7.2	关键路径核心算法	210
▲	真题仿造	213
	真题仿造答案与解析	213
	习题+真题精选	215
	习题答案+真题精选答案	221
<b>第 8 章</b>	<b>排序</b>	<b>234</b>
	大纲要求	234
	考点与要点分析	234
	核心考点	234
	基础要点	234
	知识点讲解	235
8.1	排序的基本概念	235
8.1.1	排序	235
8.1.2	稳定性	235
8.1.3	排序算法的分类	235
8.2	插入类排序	236
8.2.1	直接插入排序	236
8.2.2	折半插入排序	237
8.2.3	希尔排序	238
8.3	交换类排序	240
8.3.1	起泡排序	240
8.3.2	快速排序	241
8.4	选择类排序	243
8.4.1	简单选择排序	243
8.4.2	堆排序	244
8.5	二路归并排序	247
8.6	基数排序	248
8.7	外部排序	252
8.7.1	概念与流程	252
8.7.2	置换-选择排序	253
8.7.3	最佳归并树	254
8.7.4	败者树	255
8.7.5	时间与空间复杂度相关问题	257
8.8	排序知识点小结	258
▲	真题仿造	259
	真题仿造答案与解析	259
	习题+真题精选	260
	习题答案+真题精选答案	265
<b>第 9 章</b>	<b>查找</b>	<b>275</b>
	大纲要求	275



考点与要点分析 .....	275
核心考点 .....	275
基础要点 .....	275
知识点讲解 .....	275
9.1 查找的基本概念、顺序查找法、折半查找法 .....	275
9.1.1 查找的基本概念 .....	275
9.1.2 顺序查找法 .....	276
9.1.3 折半查找法 .....	277
9.1.4 分块查找 .....	279
9.2 二叉排序树与平衡二叉树 .....	280
9.2.1 二叉排序树 .....	280
9.2.2 平衡二叉树 .....	283
9.3 B-树的基本概念及其基本操作、B+树的基本概念 .....	286
9.3.1 B-树(B树)的基本概念 .....	286
9.3.2 B-树的基本操作 .....	288
9.3.3 B+树的基本概念 .....	292
9.4 散列表 .....	293
9.4.1 散列表的概念 .....	293
9.4.2 散列表的建立方法以及冲突解决方法 .....	293
9.4.3 散列表的性能分析 .....	297
▲真题仿造 .....	298
真题仿造答案与解析 .....	298
习题+真题精选 .....	299
习题答案+真题精选答案 .....	304
<b>第 10 章 考研中某些算法的分治法解释</b> .....	<b>318</b>
参考文献 .....	322



# 第1章 绪论

作者的话:

虽然本章涉及的知识点在各种版本的计算机专业考研大纲中没有明确要求,但它是学好“数据结构”这门课以及应对各种数据结构考试的基本功。本章对于数据结构科目在考研中涉及的不同参考书中的繁杂表述和规定做了一定的说明和简化,并根据各种版本的考研大纲所要求的知识点,总结出的一套易于接受的学习方法,因此拿到本书的考生,请务必认真阅读这一章。

## 本章概略

### ▲ 针对考研中“数据结构”内容的代码书写规范以及 C 与 C++语言基础

对于考研中的数据结构,需要 C 与 C++语言作为基础,但是又不需要太多,因此此处讲解有针对性。现在你面临的是研究生考试,要在答题纸上写代码,代码的评判者是阅卷老师,而不是 TC、VC 等编译器。你如果之前只熟悉在这些编译器下写代码,那么你要看看这一部分,这里教你如何快速地写出让阅卷老师满意的代码。

### ▲ 算法的时间复杂度分析基础

虽然计算机统考模式在逐渐取消,但它留下了一种考题模式,即综合题三段式模式(算法分析、代码实现、复杂度分析),这种模式的考题已经在一些非统试卷中出现。对于算法的复杂度分析已经几乎是每年的必考内容。相对于算法的空间复杂度,时间复杂度的分析更具有统一性,因此这里抽象出时间复杂度分析的一般套路,以方便考生理解和学习。对于空间复杂度分析,则放在以后各章中具体问题具体对待。

### ▲ 数据结构和算法的基本概念

这一部分介绍一些贯穿于整本书的基本概念。

## 1.1 针对考研数据结构的代码书写规范以及 C 与 C++语言基础

### 1.1.1 考研综合应用题中算法设计部分的代码书写规范

要在答题纸上快速地写出能让阅卷老师满意的代码是有技巧的,这与写出能在编译器上编译通过的代码有所不同。为了说明这一点,首先看一个例子:

设将  $n$  ( $n > 1$ ) 个整数存放于一维数组  $R$  中,设计一个算法,将  $R$  中的序列循环左移  $P$  ( $0 < P < n$ ) 个位置,即将  $R$  中的数据由  $\{X_0, X_1, \dots, X_{n-1}\}$  变换为  $\{X_p, X_{p+1}, \dots, X_{n-1}, X_0, X_1, \dots, X_{p-1}\}$ 。要求:写出本题的算法描述。

分析:

本题不难,要实现  $R$  中序列循环左移  $P$  个位置,只需先将  $R$  中前  $P$  个元素逆置,再将剩下的元素逆置,最后将  $R$  中所有的元素再整体做一次逆置操作即可。本题算法描述如下:

```
#include<iostream> //1
#define N 50 //2
using namespace std; //3
void Reverse(int R[],int l,int r) //4
```



```
{ //5
    int i,j; //6
    int temp; //7
    for(i=l,j=r;i<j;++i,--j) //8
    { //9
        temp=R[i]; //10
        R[i]=R[j]; //11
        R[j]=temp; //12
    } //13
} //14
void RCR(int R[],int n,int p) //15
{ //16
    if(p<=0||p>=n) //17
        cout<<"ERROR"<<endl; //18
    else //19
    { //20
        Reverse(R,0,p-1); //21
        Reverse(R,p,n-1); //22
        Reverse(R,0,n-1); //23
    } //24
} //25
int main() //26
{ //27
    int L,i; //28
    int R[N],n; //29
    cin>>L; //30
    cin>>n; //31
    for(i=0;i<=n-1;++i) //32
        cin>>R[i]; //33
    RCR(R,n,L); //34
    for(i=0;i<=n-1;++i) //35
        cout<<R[i]<<" "; //36
    cout<<endl; //37
    return 0; //38
} //39
```

以上程序段是一段完整的可以在编译器下编译运行的程序，程序比较长，对于考试答卷，完全没有必要这么写。

第1句和第3句，在大学学习期间所写的程序中几乎都要用到，研究生考试这种选拔考试不会用这种东西来区分学生的优劣，因此答题过程中没必要写，可去掉。

第2句定义了一个常量，如果题目中要用一个常量，则在用到的地方加上一句注释，说明某某常量之前已经定义即可。没必要再在前面补上一句`#define` `××××`，因为试卷是答题纸，不是文本编辑器，插入语句不是那么方便。为了节省考试时间且使试卷整洁，第2句也可去掉。

第26~39句是主函数部分，之前定义的函数（第4~25句）在这里调用。在答题中，只需要写出自己的函数说明（第4~25句），写清楚函数的接口（何为接口，下边会细致讲解）即可，阅卷老师就知道



你已经做好了可以解决这个题目的工具（函数），并且说明了工具的使用方法（函数接口），因此第 26~39 句也可以去掉。

经过以上删减，就变成以下程序段了，显然简洁了很多。

```
void Reverse(int R[],int l,int r)           //1
{                                           //2
    int i,j;                               //3
    int temp;                              //4
    for(i=l,j=r;i<j;++i,--j)              //5
    {                                       //6
        temp=R[i];                        //7
        R[i]=R[j];                        //8
        R[j]=temp;                        //9
    }                                       //10
}                                           //11
void RCR(int R[],int n,int p)              //12
{                                           //13
    if(p<=0||p>=n)                        //14
        cout<<"ERROR"<<endl;            //15
    else                                    //16
    {                                       //17
        Reverse(R,0,p-1);                 //18
        Reverse(R,p,n-1);                 //19
        Reverse(R,0,n-1);                 //20
    }                                       //21
}                                           //22
```

这里来说一下函数的接口。假如上述函数是一台机器，可以用原材料来加工成成品，那么接口就可以理解成原材料的入口，或成品的出口。例如，上述程序段中的第 12 句：`RCR(int R[],int n,int p)`就包含一个接口，它是原材料的一个入口。括号里所描述的就是原材料的类型以及名称，是将来函数被调用的时候所要放进去的东西，是在告诉别人，需要 3 个原材料：第一个是一个 `int` 型的数组，第二个是一个 `int` 型的变量，第三个也是一个 `int` 型的变量。第 15 句：`cout<<"ERROR"<<endl;`也是一个接口，它会在输出设备上打印出英文单词 `ERROR`，用来提示用户这里出错了，这算是成品的出口，这里的成品就是一个提示。同时，第 1 句中传入 `int` 型数组的地方也可以理解为一个产品的出口，因为从这里传入的数组的内容，将在函数执行完后被加工成我们想要的内容。通过以上说明，可以把接口理解为用户和函数打交道的地方，通过接口，用户输入了自己的数据，得到了自己想要的结果。

至此，我们可以知道考研综合应用中算法设计题中的代码部分重点需要写哪些内容了，即只需写出一个或多个可以解决问题的有着清楚接口描述的函数即可。

## 1.1.2 考研中的 C 与 C++ 语言基础

本节的标题是 C 语言及 C++ 语言，而不是单一的一种语言，是因为本书有些程序的书写包含了这两种语法。对于考试答题来说，C++ 不因为它是 C 语言的升级版就能取代 C。C 和 C++ 是各有所长的，我们从两者中挑选出来对考研答卷有利的部分，组合起来应用。下面具体介绍针对考研数据结构的 C 和 C++ 语言基础。

### 1. 数据类型

对于基本的数据类型，如整型 `int`、`long`、...（考研中涉及处理整数的题目，如果没有特别要求用 `int`