

National Computer Rank Examination

30天通过

全国计算机等级考试:

二级C

希赛教育等考学院 桂阳 主编



30天通过考试不难!

最新
大纲
分析

+

实用
应试
指南

+

清晰
考点
讲解

+

大量
同步
练习

+

丰富笔试和
上机模拟试卷

+

最新真题
试卷演练

→

全方位帮助考生
快速、高效、
轻松过关!



光盘特色

- ◆ 配备3种模拟方式 (固定抽题、随机抽题、重复抽题) 和大量题库, 模拟实考环境, 反复练习, 力保顺利通过考试
- ◆ 灵活的模拟操作界面, 随时自测, 实时查看成绩, 做到心中有数
- ◆ 盘内免费提供精选的教学视频, 名师讲解, 学习更轻松
- ◆ 丰富的网络配套资源, 涵盖论坛答疑、网络课堂、在线测试



全国计算机等级考试专用辅导丛书
National Computer Rank Examination

National Computer Rank Examination

30天通过

全国计算机等级考试

二级C



希赛教育等考学院 桂阳 主编

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书由希赛教育等考学院组织编写,内容紧扣教育部考试中心新推出的考试大纲,通过对历年试题进行科学分析、研究、总结和提炼而成。

本书主要内容包括应试指南、考点分析与同步训练、5套笔试全真模拟试卷及解析、10套机试全真模拟试题及解析、2010年上半年及2010年下半年的真题解析。

基于历年试题,利用统计分析的方法,科学做出结论并预测以后的出题动向,是本书的一大特色。本书可以保证既不漏掉考试必需的知识点,又不加重考生备考负担,使考生轻松、愉快地掌握知识点,并领悟考试的真谛。

本书可以作为参加全国计算机等级考试:二级C语言程序设计考试的复习用书,也可作为学习C语言程序设计的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

30天通过全国计算机等级考试. 二级C / 桂阳主编. —北京: 电子工业出版社, 2011.1

(全国计算机等级考试专用辅导丛书)

ISBN 978-7-121-12001-5

I. ①3… II. ①桂… III. ①电子计算机—水平考试—自学参考资料②C语言—程序设计—水平考试—自学参考资料 IV. ①TP3

中国版本图书馆CIP数据核字(2010)第199330号

责任编辑: 李利键

印 刷: 北京中新伟业印刷有限公司
装 订:

出版发行: 电子工业出版社

北京市海淀区万寿路173信箱 邮编100036

开 本: 860×1092 1/16 印张: 19.5 字数: 858千字

印 次: 2011年1月第1次印刷

印 数: 4000册 定价: 39.80元(含光盘1张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

前言

全国计算机等级考试（NCRE）由教育部考试中心主办，面向社会，用于考查非计算机专业人员的计算机应用知识与能力。考试客观、公正，得到了社会的广泛认可。

本书紧扣最新考试大纲，采用了表格统计法、饼状图分析法、走势图分析法，科学地研究每个知识点的命题情况，准确把握每个出题点的深浅。同时基于每个章节知识点分布统计分析的结果，科学地编写后面的同步训练试题及全真模拟试卷，完全紧扣大纲。结构科学、重点突出、针对性强。

内容超值，针对性强

本书共分 5 个部分。

第一部分为应试指南。应试指南内有考试大纲，笔试试题历年知识点分布表，笔试、机试的方法和技巧。通过学习本部分内容，考生可以对计算机等级考试的知识点分布、常用应试方法和技巧有一个整体上的认识和把握。

第二部分为知识点解析。知识点解析涵盖了新大纲中所规定的所有考试内容和考试必备的知识点。本部分充分采用多种统计分析方法研究试题并指导考生备考。每章开头都有“章节前奏”，利用表格统计法在时间上分析该章节中各个小节的历年出题情况，并利用饼状图在空间上直观表示各个小节的试题分布。每个小节开头都有“小节前奏”，利用趋势分析法精确地预测以后命题的情况。章节中的知识点解析深浅程度根据该知识点在历年试题中的统计分析结果而定。每个小节后面都有“同步训练”及其解答，帮助考生温习和巩固前面所学的理论知识，也是根据前面知识点统计分析的结果而命题。这种辅导方式保证内容全面，重点突出，为考生打造一条通向等级考试终点的捷径。

第三部分为笔试全真模拟试卷。笔试全真模拟试卷完全紧扣大纲，基于 2010 年上半年考试真题编写，无论在形式方面还是难度方面，都和真题相似，并且配有详尽的解析。

第四部分为机试指导、分类解析及模拟。上机考试是考生的一个弱项，往往令考生不知所措。对试题进行分类可以很好地指导考生应考，经过分析上机考试题库内容，按照其考试内容，将其分为 3 类：字符问题、数字问题和结构体问题，每类问题还可以分出一些子问题，对其进行逐类讲解，使考生快速适应上机考试模式和结构。本部分还配有一些习题供考生练习，考生也可以到希赛教育等考学院网站（www.csaiddk.com）下载最新的上机环境和题库，以便及时跟踪上机考试动向。

第五部分为附录。附录部分包括 2010 年上半年和 2010 年下半年考试真题及详细解析。

作者权威，阵容强大

希赛教育（www.educity.cn）专业从事人才培养、教育产品开发、教育图书出版，在职业教育方面具有极高的权威性。特别是在在线教育方面，名列前茅，希赛教育的远程教育模式得到了国家教育部门的认可和推广。

希赛教育等考学院是国内首屈一指的进行计算机等级考试在线教育的大型教育机构，在该领域取得了很好的效果。我们组织大纲制订者和阅卷组成人员编写了考试辅导教材近 20 本，内容涵盖了计算机等级考试的主要级别。组织权威专家和辅导名师录制了考试培训视频教程，对历年考试进行了跟踪研究和比较研究，编写了权威的全真模拟试题。希赛教育的计算机等级考试培训采取统一教材、统一视频、统一认证教师的形式，采取线下培训与线上辅导相结合的方式，

确保学员在通过考试的前提下能真正学到有用的知识。

本书由希赛教育等考学院组织编写，参加编写的人员来自大学教学一线和企业研发团队，具有丰富的教学和辅导经验，对等级考试有深入的研究，具有极强的应试技巧、理论知识、实践经验和责任心。

本书由希赛教育等考学院和桂阳主编，张友生审核了所有稿件。参与编写的人员还有张友生、胡钊源、施游、王勇、朱小平、李雄、刘毅、周玲、符春和唐小娟。

在线测试，心中有数

上学吧在线测试平台（www.shangxueba.com）为考生准备了在线测试，其中有数十套全真模拟试题和考前密卷，考生可选择任何一套进行测试。测试完毕，系统自动判卷，立即给出分数。

对于考生做错的地方，系统会自动记忆，待考生第二次参加测试时，可选择“试题复习”。这样，系统就会自动把考生原来做错的试题显示出来，供考生重新测试，以加强记忆。

如此，读者可利用上学吧在线测试平台的在线测试系统检查自己的实际水平，加强考前训练，做到心中有数，考试不慌。

诸多帮助，诚挚致谢

在本书出版之际，要特别感谢教育部考试中心计算机等级考试办公室的命题专家们，编者在本书中引用了部分考试原题，使本书能够尽量方便读者的阅读。在本书的编写过程中，参考了许多相关的文献和书籍，编者在此对这些参考文献的作者表示感谢。

感谢电子工业出版社田小康老师，他在本书的策划、选题的申报、写作大纲的确定，以及编辑、出版等方面，付出了辛勤的劳动和智慧，给予了我们很多的支持和帮助。

感谢参加希赛教育计算机等级考试辅导和培训的学员，正是他们的想法汇成了本书的原动力，他们的意见使本书更加贴近读者。

由于编者水平有限，且本书涉及的内容很广，书中难免存在错漏和不妥之处，编者诚恳地期望各位专家和读者不吝指正，对此，我们将十分感激。

互动讨论，专家答疑

希赛教育等考学院（www.csaidk.com）是中国领先的计算机等级考试在线教育网站，该网站论坛是国内人气很旺的计算机等级考试社区，在这里，读者可以和数百万考生进行在线交流。希赛教育等考学院拥有强大的师资队伍，为读者提供全程的答疑服务，在线回答读者的提问。

有关本书的意见反馈和咨询，读者可在希赛教育等考学院论坛“等级考试教材”板块中的“希赛教育等考学院”栏目上与作者进行交流。

希赛教育等考学院

目 录

第1章 应试指南1	
1.1 二级C语言考试大纲(最新版).....1	
1.1.1 公共基础知识.....1	
1.1.2 C语言程序设计.....2	
1.2 笔试方法和技巧.....3	
1.3 机试方法和技巧.....5	
第2章 数据结构与算法7	
2.1 算法的基本概念.....7	
2.1.1 考点1: 算法的定义.....8	
2.1.2 考点2: 算法复杂度.....8	
2.2 数据结构的定义.....9	
2.2.1 考点1: 什么是数据结构.....9	
2.2.2 考点2: 数据结构的图形表示.....10	
2.3 线性表.....11	
2.3.1 考点1: 线性表.....11	
2.3.2 考点2: 线性表的顺序存储结构.....11	
2.3.3 考点3: 线性表的插入和删除操作.....11	
2.4 栈.....12	
2.4.1 考点1: 什么是栈.....12	
2.4.2 考点2: 栈的顺序存储结构.....12	
2.4.3 考点3: 栈的插入和删除运算.....13	
2.5 队列.....13	
2.5.1 考点1: 什么是队列.....13	
2.5.2 考点2: 队列的顺序存储结构.....13	
2.5.3 考点3: 队列的插入和删除运算.....14	
2.6 线性单链表、双向链表与循环链表.....14	
2.6.1 考点1: 线性单链表.....14	
2.6.2 考点2: 循环链表.....15	
2.6.3 考点3: 双向链表.....15	
2.6.4 考点4: 链表的基本操作.....15	
2.7 树.....16	
2.7.1 考点1: 树的定义.....16	
2.7.2 考点2: 二叉树的定义.....17	
2.7.3 考点3: 二叉树的顺序存储结构.....17	
2.7.4 考点4: 二叉树的链式存储结构.....18	
2.7.5 考点5: 二叉树的遍历.....18	
2.8 查找算法.....19	
2.8.1 考点1: 顺序查找.....19	
2.8.2 考点2: 二分查找.....20	
2.9 排序算法.....20	
2.9.1 考点1: 排序概述.....20	
2.9.2 考点2: 插入类排序.....21	
2.9.3 考点3: 交换类排序.....21	
2.9.4 考点4: 选择类排序.....21	
2.10 小结.....22	
2.11 同步训练.....23	
2.11.1 选择题.....23	
2.11.2 填空题.....24	
2.12 同步训练答案.....24	
2.12.1 选择题.....24	
2.12.2 填空题.....24	
第3章 程序设计基础25	
3.1 程序设计.....25	
3.1.1 考点1: 程序设计的方法.....26	

3.1.2 考点 2: 程序设计的风格	26	4.4 软件测试	45
3.2 结构化程序设计	27	4.4.1 考点 1: 软件测试的目的	45
3.2.1 考点 1: 结构化程序设计的提出	27	4.4.2 考点 2: 测试方法	46
3.2.2 考点 2: 结构化程序设计的 基本特征	28	4.4.3 考点 3: 测试用例设计	46
3.2.3 考点 3: 结构化程序设计的 3 种 基本结构	28	4.4.4 考点 4: 软件测试策略	48
3.2.4 考点 4: 结构化程序设计方法	30	4.5 程序调试	49
3.2.5 考点 5: 结构化程序设计方法 的原则	30	4.5.1 考点 1: 静态调试	50
3.3 面向对象程序设计	31	4.5.2 考点 2: 动态调试	50
3.3.1 考点 1: 面向对象程序设计方法 的提出	31	4.5.3 考点 3: 调试策略	50
3.3.2 考点 2: 面向对象程序设计方法 的基本概念	31	4.6 软件维护	51
3.3.3 考点 3: 面向对象程序设计的基本 思想和方法	32	4.7 小结	51
3.4 小结	33	4.8 同步训练	52
3.5 同步训练	33	4.8.1 选择题	52
3.5.1 选择题	33	4.8.2 填空题	53
3.5.2 填空题	34	4.9 同步训练答案	53
3.6 同步训练答案	34	4.9.1 选择题	53
3.6.1 选择题	34	4.9.2 填空题	53
3.6.2 填空题	34	第 5 章 数据库设计基础	54
第 4 章 程序工程基础	35	5.1 数据库基础知识	54
4.1 软件工程的基本概念	35	5.1.1 考点 1: 数据库	55
4.1.1 考点 1: 软件与软件工程概述	36	5.1.2 考点 2: 数据库管理系统	55
4.1.2 考点 2: 软件生命周期	37	5.1.3 考点 3: 数据库系统	56
4.1.3 考点 3: 软件工具与软件开发环境	37	5.1.4 考点 4: 数据管理技术的发展	56
4.2 结构化分析方法	37	5.2 数据模型	57
4.2.1 考点 1: 结构化分析方法概述	37	5.2.1 考点 1: 数据模型的概述	57
4.2.2 考点 2: 数据流图	38	5.2.2 考点 2: 实体联系模型及 E-R 图	58
4.2.3 考点 3: 数据字典	40	5.2.3 考点 3: 层次、网状、关系模型	60
4.2.4 考点 4: 软件需求规格说明书	40	5.2.4 考点 4: 从 E-R 图导出关系 数据模型	61
4.3 软件设计	40	5.3 关系运算	62
4.3.1 考点 1: 概要设计	41	5.3.1 考点 1: 关系代数	62
4.3.2 考点 2: 详细设计	43	5.3.2 考点 2: 传统的集合运算	62
		5.3.3 考点 3: 专门的关系运算	63
		5.4 数据库设计	65
		5.4.1 考点 1: 数据库设计概述	65
		5.4.2 考点 2: 数据库设计方法	65
		5.4.3 考点 3: 数据库设计步骤	66

5.4.4 考点4: 数据库的规范化	66	7.1.2 考点2: 赋值语句、复合语句和 空语句	83
5.5 小结	67	7.2 数据输出	84
5.6 同步训练	67	7.2.1 考点1: 单字符输出函数 putchar	84
5.6.1 选择题	67	7.2.2 考点2: 字符串输出函数 puts	84
5.6.2 填空题	68	7.2.3 考点3: 格式输出函数 printf	85
5.7 同步训练答案	69	7.3 数据输入	86
5.7.1 选择题	69	7.3.1 考点1: 字符输入函数 getchar	86
5.7.2 填空题	69	7.3.2 考点2: 字符串输入函数 gets	87
第6章 C 程序设计的初步知识	70	7.3.3 考点3: 格式输入函数 scanf	87
6.1 C 程序简介	70	7.4 同步训练	88
6.1.1 考点1: C 源程序编译与执行	70	7.4.1 选择题	88
6.1.2 考点2: 程序设计步骤	71	7.4.2 填空题	90
6.2 C 程序构成与格式	71	7.5 同步训练答案	91
考点: C 程序构成和格式	71	7.5.1 选择题	91
6.3 数据类型	72	7.5.2 填空题	91
6.3.1 考点1: 标识符	72	第8章 选择结构程序设计	92
6.3.2 考点2: 常量	72	8.1 基本概念	92
6.3.3 考点3: 变量	72	考点: 关系运算和逻辑运算	92
6.3.4 考点4: 整型数据	73	8.2 if 语句与条件表达式	93
6.3.5 考点5: 实型数据	73	考点: if 语句及条件表达式	93
6.3.6 考点6: 字符型数据	73	8.3 switch 语句与 goto 语句	95
6.4 运算符和表达式	74	8.3.1 考点1: switch 语句使用	95
6.4.1 考点1: 算术运算符	74	8.3.2 考点2: goto 语句	96
6.4.2 考点2: 算术表达式	75	8.4 同步训练	97
6.4.3 考点3: 赋值表达式	75	8.4.1 选择题	97
6.4.4 考点4: 自加、自减运算符	76	8.4.2 填空题	98
6.4.5 考点5: 逗号运算符和逗号表达式	76	8.5 同步训练答案	99
6.5 同步训练	76	8.5.1 选择题	99
6.5.1 选择题	76	8.5.2 填空题	99
6.5.2 填空题	79	第9章 循环结构程序设计	100
6.6 同步训练答案	81	9.1 while 和 do-while 循环结构	100
6.6.1 选择题	81	9.1.1 考点1: while 语句	100
6.6.2 填空题	81	9.1.2 考点2: do-while 语句	101
第7章 顺序结构程序设计	82	9.2 for 循环结构	102
7.1 基本概念	82	考点: for 语句的基本用法	102
7.1.1 考点1: 顺序程序概述	82		

9.3 循环结构嵌套及流程控制	104	11.2.1 考点 1: 指针变量作为函数参数 ..	134
9.3.1 考点 1: 循环结构嵌套	104	11.2.2 考点 2: 指向函数的指针	135
9.3.2 考点 2: break 语句的使用	104	11.3 数组基础知识	137
9.3.3 考点 3: continue 语句的使用	105	11.3.1 考点 1: 一维数组的定义与 初始化	137
9.4 同步训练	106	11.3.2 考点 2: 一维数组的引用	137
9.4.1 选择题	106	11.3.3 考点 3: 二维数组的定义与 初始化	139
9.4.2 填空题	110	11.3.4 考点 4: 二维数组的引用	140
9.5 同步训练答案	111	11.4 数组与指针	142
9.5.1 选择题	111	11.4.1 考点 1: 指针与变量的地址	142
9.5.2 填空题	111	11.4.2 考点 2: 指针与一维数组	142
第 10 章 函 数	112	11.4.3 考点 3: 指针与二维数组	145
10.1 函数基本概念	112	11.4.4 考点 4: 指针数组与指向指针 的指针	147
10.1.1 考点 1: 函数的概念	112	11.5 数组与函数参数	148
10.1.2 考点 2: 函数的定义	113	11.5.1 考点 1: 数组元素作为函数参数 ..	148
10.1.3 考点 3: 函数的调用方式	113	11.5.2 考点 2: 数组名作为函数参数	148
10.2 函数调用中的数据传递	114	11.6 字符数组与字符串	149
10.2.1 考点 1: 参数的传递	114	11.6.1 考点 1: 字符数组与字符串	149
10.2.2 考点 2: 函数的返回值	115	11.6.2 考点 2: 字符指针变量与字符串 ..	151
10.3 函数的递归调用	117	11.6.3 考点 3: 典型字符串处理函数	152
10.3.1 考点 1: 函数的嵌套调用	117	11.6.4 考点 4: 字符数组与字符串的 应用	153
10.3.2 考点 2: 函数的递归调用	118	11.7 同步训练	155
10.4 函数的存储分类和变量的存储分类	119	11.7.1 选择题	155
10.4.1 考点 1: 内部函数和外部函数	119	11.7.2 填空题	164
10.4.2 考点 2: 局部变量和全局变量	120	11.8 同步训练答案	168
10.4.3 考点 3: 变量的存储类型	120	11.8.1 选择题	168
10.5 同步训练	123	11.8.2 填空题	168
10.5.1 选择题	123	第 12 章 编译预处理与动态存储分配	169
10.5.2 填空题	129	12.1 编译预处理	169
10.6 同步训练答案	130	12.1.1 考点 1: 预处理概述	169
10.6.1 选择题	130	12.1.2 考点 2: 宏定义	170
10.6.2 填空题	130	12.1.3 考点 3: 文件包含	171
第 11 章 指针与数组	131	12.1.4 考点 4: 条件编译	172
11.1 指针基础知识	131	12.2 动态存储分配	172
11.1.1 考点 1: 地址、指针和指针变量 ..	131		
11.1.2 考点 2: 指针变量的定义与引用 ..	132		
11.2 指针在函数数据传递中的应用	134		

考点 1: 动态存储分配常用函数	172	14.4.1 选择题	204
12.3 同步训练	174	14.4.2 填空题	204
12.3.1 选择题	174	第 15 章 文件操作	205
12.3.2 填空题	175	15.1 文件指针	205
12.4 同步训练答案	176	15.1.1 考点 1: 文件基本概念	205
12.4.1 选择题	176	15.1.2 考点 2: 文件类型指针	206
12.4.2 填空题	176	15.2 文件打开与关闭	206
第 13 章 结构体与共用体	177	15.2.1 考点 1: 文件的打开	206
13.1 类型定义符	177	15.2.2 考点 2: 文件的关闭	207
考点 1: typedef	177	15.3 文件的读/写	207
13.2 结构体	178	15.3.1 考点 1: 字符的读/写	207
13.2.1 考点 1: 结构变量说明	178	15.3.2 考点 2: 字符串的读/写	207
13.2.2 考点 2: 结构变量的赋值和 初始化	180	15.3.3 考点 3: 格式化读/写	207
13.2.3 考点 3: 结构指针变量	182	15.3.4 考点 4: 成块读/写	208
13.2.4 考点 4: 结构数组	184	15.4 文件的定位	208
13.2.5 考点 5: 利用结构体变量 构成链表	185	考点 1: 文件的定位	208
13.3 联合体	190	15.5 同步训练	208
考点 1: 联合的定义和联合变量的说明	190	15.5.1 选择题	208
13.4 同步训练	192	15.5.2 填空题	210
13.4.1 选择题	192	15.6 同步训练答案	211
13.4.2 填空题	199	15.6.1 选择题	211
13.5 同步训练答案	201	15.6.2 填空题	211
13.5.1 选择题	201	第 16 章 笔试模拟试卷及解析	212
13.5.2 填空题	201	16.1 模拟试题一	212
第 14 章 位运算	202	16.2 模拟试题二	218
14.1 位运算符	202	16.3 模拟试题三	223
考点 1: 常用位运算符	202	16.4 模拟试题四	229
14.2 位运算操作	203	16.5 模拟试题五	235
14.2.1 考点 1: 移位运算	203	16.6 答案与解析	240
14.2.2 考点 2: 按位逻辑运算	203	16.6.1 模拟试题 1	240
14.3 同步训练	204	16.6.2 模拟试题 2	243
14.3.1 选择题	204	16.6.3 模拟试题 3	246
14.3.2 填空题	204	16.6.4 模拟试题 4	248
14.4 同步训练答案	204	16.6.5 模拟试题 5	251
		第 17 章 机试指导、分类解析及模拟	255
		17.1 机试纪律及步骤	255

17.1.1 上机考试纪律	255	模拟试题 10 上机模拟 10	278
17.1.2 操作步骤及考试规则	255	17.4 参考答案	279
17.2 机试分类解析	257	模拟试题 1 上机模拟 1	279
17.2.1 程序填空题	257	模拟试题 2 上机模拟 2	279
17.2.2 程序修改题	262	模拟试题 3 上机模拟 3	279
17.2.3 程序设计题	265	模拟试题 4 上机模拟 4	279
17.3 机试全真模拟	271	模拟试题 5 上机模拟 5	279
模拟试题 1 上机模拟 1	271	模拟试题 6 上机模拟 6	279
模拟试题 2 上机模拟 2	271	模拟试题 7 上机模拟 7	280
模拟试题 3 上机模拟 3	272	模拟试题 8 上机模拟 8	280
模拟试题 4 上机模拟 4	273	模拟试题 9 上机模拟 9	280
模拟试题 5 上机模拟 5	274	模拟试题 10 上机模拟 10	280
模拟试题 6 上机模拟 6	275		
模拟试题 7 上机模拟 7	276	附录 A 2010 年上半年笔试试卷及解析	281
模拟试题 8 上机模拟 8	276	附录 B 2010 年下半年笔试试卷及解析	291
模拟试题 9 上机模拟 9	277		

第1章

应试指南

全国计算机等级考试（National Computer Rank Examination，简称 NCRE）由教育部考试中心于 1994 年面向社会推出，主要考查非计算机专业人员的计算机应用知识与能力。十多年来，NCRE 为全国培养几百万各行各业的计算机应用人才。C 语言既是计算机等级二级考试中的元老和常青树，又是三级网络、信息管理技术、数据库技术的上机考核内容，有助于考生以后继续深造，并且是当今世界上几乎所有最流行语言的基础，因此备受考生的青睐，每年占报考计算机等级二级考试总人数的 55% 以上。

为了帮助更多考生顺利通过考试，并掌握相应的操作技能，我们深入调研、根据国家教育部考试中心制定的最新二级 C 语言等级考试大纲，结合近年来考试命题规律，组织国内著名计算机专家和一线教师编写了本书。本书包括笔试和上机，考生一书在手就可以进行全面的考前演练。

二级 C 语言考试知识点繁多，复习时要采用“厚书读薄、薄书看厚”的顺序。首先要了解考试大纲，对应试知识结构有一个整体了解，然后抓住考试重点和热点（厚书读薄），有针对性地复习，把这些内容弄得烂熟（薄书看厚），通过考试则不成问题。为此，本书结合例题，全面、系统地讲述大纲中的重点和热点，帮助考生把握知识要点、建立知识体系，以便全面复习应考，最终通过二级 C 语言考试。

1.1 二级 C 语言考试大纲（最新版）

1.1.1 公共基础知识

1. 基本要求

- （1）掌握算法的基本概念。
- （2）掌握基本数据结构及其操作。
- （3）掌握基本排序和查找算法。
- （4）掌握逐步求精的结构化程序设计方法。
- （5）掌握软件工程的基本方法，具有初步应用相关技术进行软件开发的能力。
- （6）掌握数据库的基本知识，了解关系数据库的设计。

2. 考试内容

（1）基本数据结构与算法

- 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）。
- 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。
- 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。
- 栈和队列的定义；栈和队列的顺序存储结构及其基本运算。
- 线性单链表、双向链表与循环链表的结构及其基本运算。
- 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。
- 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）。

（2）程序设计基础

- 程序设计方法与风格。
- 结构化程序设计。
- 面向对象的程序设计方法、对象、方法、属性及继承与多态性。

（3）软件工程基础

- 软件工程的基本概念，软件生命周期概念，软件工具与软件开发环境。
- 结构化分析方法，数据流图，数据字典，软件需求规格说明书。

- 结构化设计方法，总体设计与详细设计。
- 软件测试的方法，白盒测试与黑盒测试，测试用例设计，软件测试的实施，单元测试、集成测试和系统测试。
- 程序的调试，静态调试与动态调试。

(4) 数据库设计基础

- 数据库的基本概念：数据库、数据库管理系统、数据库系统。
- 数据模型，实体关系模型及 E-R 图，从 E-R 图导出关系数据模型。
- 关系代数运算：集合运算及选择、投影、连接运算，数据库规范化理论。
- 数据库设计方法和步骤：需求分析、概念设计、逻辑设计和物理设计的相关策略。

3. 考试方式

- (1) 公共基础知识的考试方式为笔试，与 C 语言程序设计的笔试部分合为一张试卷。公共基础知识部分占全卷的 30 分。
- (2) 公共基础知识有 10 道选择题和 5 道填空题。

1.1.2 C 语言程序设计

1. 基本要求

- (1) 熟悉 Visual C++ 6.0 集成开发环境。
- (2) 掌握结构化程序设计的方法，具有良好的程序设计风格。
- (3) 掌握程序设计中简单的数据结构和算法并能阅读简单的程序。
- (4) 在 Visual C++ 6.0 集成环境下，能够编写简单的 C 程序，并具有基本的纠错和调试程序的能力。

2. 考试内容

(1) C 语言的结构

- 程序的构成，main 函数和其他函数。
- 头文件、数据说明、函数的开始和结束标志。
- 源程序的书写格式。
- C 语言的风格。

(2) 数据类型及其运算

- C 的数据类型（基本类型、构造类型、指针类型、空类型）及其定义方法。
- C 运算符的种类、运算优先级和结合性。
- 不同类型数据间的转换与运算。
- C 表达式类型（赋值表达式、算术表达式、关系表达式、逻辑表达式、条件表达式、逗号表达式）和求值规则。

(3) 基本语句

- 表达式语句、空语句和复合语句。
- 数据的输入与输出、输入输出函数的调用及正确设计输入/输出格式。

(4) 选择结构程序设计

- 用 if 语句实现选择结构。
- 用 switch 语句实现多分支选择结构。
- 选择结构的嵌套。

(5) 循环结构程序设计

- for 循环结构。
- while 和 do-while 循环结构。
- continue 语句和 break 语句。
- 循环的嵌套。

(6) 数组的定义和引用

- 一维数组和多维数组的定义、初始化和引用。
- 字符串与字符数组。

(7) 函数

- 库函数的正确调用。
- 函数的定义方法。
- 函数的类型和返回值。
- 形式参数与实际参数，参数值的传递。
- 函数的正确调用、嵌套调用和递归调用。
- 局部变量和全局变量。
- 变量的存储类别（自动、静态、寄存器、外部）、变量的作用域和生存期。
- 内部函数与外部函数。

(8) 编译预处理

- 宏定义和调用（不带参数的宏和带参数的宏）。

- “文件包含”处理。
- (9) 指针
 - 指针与指针变量的概念, 指针与地址运算符。
 - 变量、数组、字符串、函数、结构体的指针以及指向变量、数组、字符串、函数、结构体的指针变量。通过指针引用以上各类型数据。
 - 用指针作函数参数。
 - 返回指针值的指针函数。
 - 指针数组、指向指针的指针、main 函数的命令行参数。
- (10) 结构体与共用体
 - 用 typedef 说明一个新的类型。
 - 结构体和共用体类型数据的定义方法和引用方法。
 - 用指针和结构体构成链表, 单向链表的建立、输出、删除与插入。
- (11) 位运算
 - 位运算符的含义及使用。
 - 简单的位运算。
- (12) 文件操作

只要求缓冲文件系统(即高级磁盘 I/O 系统), 对非标准缓冲文件系统(即低级磁盘 I/O 系统)不要求。

 - 文件类型指针(file 类型指针)。
 - 文件的打开与关闭(fopen、fclose)。
 - 文件的读写(fputc、fgetc、fputs、fgets、fread、fwrite、fprintf、fscanf 函数)。
 - 文件的定位(rewind、fseek 函数)。

1.2 笔试方法和技巧

二级 C 语言笔试时间 90 分钟, 40 道单项选择题。其中, 1~10 题与 21~40 题每题 2 分, 11~20 题每题 1 分; 15 道填空题, 每题 2 分; 共计 100 分, 60 分合格。近年来考试通过率一般在 45% 左右, 具有一定的难度, 需要一定的方法和技巧来应对。

学习方法因人而异, 不同的人适合不同的方法。常用的通用方法如下:

(1) 及时复习, 这是最基本的方法。每隔一段时间以后, 回过头来看一下自己以前学习的内容。这种复习花费时间不多, 而且随时间的延续呈对数递减关系, 但是作用非常大, 一方面可以巩固自己以前的知识, 另一方面可以加深前后知识的连贯, 形成全面的知识体系结构。

(2) 归纳整理, 注重实践。对初学者而言, 应对二级 C 语言考试, 知识点记忆是一个难关, 除了要记忆计算机基础知识、基本概念外, 还需要适当记忆和 C 语言相关的知识点, 如 C 语言数据类型、运算符等。如此庞大的知识点体系, 采用死记硬背的方法显然不行, 我们可以在理解的基础上归纳整理, 适当记忆。此外, 因为二级 C 语言考试中题目大多都能上机实践, 所以实践非常重要。考生应在对基本知识理解的同时注意多上机实践, 通过实践, 将所学知识融会贯通。

(3) 适度模拟训练。每隔一段时间, 对自己进行一次全真模拟测试, 通过测试发现不足, 对症下药进行解决。由于模拟测试只是手段, 而不是目的, 所以不宜频繁进行这种测试, 笔试核心还是多看教程、多总结和思考。

(4) 建立错题集。把自己平时模拟测试易错的试题记录起来, 每隔一段时间, 对照错题中涉及的知识点, 专门复习, 效果极好, 可以大幅度提高学习成绩。

二级 C 语言笔试主要测试考生对基本概念、基本原理的掌握和利用 C 语言初步开发程序的能力。经分析, 大多数考题与大纲要求基本一致, 难度不大, 但内容十分广泛, 所以, 考生备考时要全面复习, 加强对基本知识点的理解的训练。同时也要研究历年试题, 把握考试重点和热点, 并掌握解题技巧。

笔试题知识点分布简单统计如表 1-1 所示。

表 1-1 笔试知识点历年分布情况(单位: 分数)

知识内容	2006.9	2007.4	2007.9	2008. 4	2008.9	2009.3	2009.9	2010.3	2010.9	总计
数据结构与算法	10	10	10	10	12	10	10	10	10	92
程序设计基础	2	2	4	4	2	2	4	0	4	24
软件工程基础	8	10	10	6	8	8	6	10	6	72
数据库设计基础	10	8	6	10	8	10	10	10	10	82
C 程序设计的初步知识	12	13	10	14	11	12	13	8	11	104
顺序结构程序设计	2	4	8	4	3	2	3	5	4	35

续表

知识内容	2006.9	2007.4	2007.9	2008. 4	2008.9	2009.3	2009.9	2010.3	2010.9	总计
选择结构程序设计	5	7	6	2	3	5	5	4	4	41
循环结构程序设计	6	2	13	8	10	8	6	7	9	69
函数	11	10	6	10	11	12	10	14	10	94
指针与数组	20	20	15	18	19	16	19	22	24	173
编译预处理与动态存储分配	3	3	3	3	5	4	4	2	2	29
结构体与共用体	6	6	6	5	4	5	6	4	2	44
位运算	1	3	1	2	2	2	2	2	2	17
文件操作	4	2	2	4	2	4	2	2	2	24

由上述统计分析可知：公共基础知识（数据结构与算法、程序设计基础、软件工程基础、数据库设计基础）分值稳定，每年均为 30 分。C 程序设计的初步知识是 C 语言编程基础，从总体趋势来看分值有所下降，一般占 10 分左右，这表明考试整体难度有一定上升。指针与数组历来都是考试重点，其分值有上升的趋势，一般占 20 分左右。函数和循环结构程序设计也是考试热点，其分值一般占 10 分左右，不容忽视。其他章节分值一般占 2~5 分左右，其中，选择结构程序设计、编译预处理与动态存储分配、文件操作分值稳定，顺序结构程序设计、结构体与共用体分值有起有落，位运算很少出题，但近几次考试分值稳定，一般是 2 分。

综上所述，考试内容有侧重点，复习迎考时应该抓住重点，这样可以事半功倍，切忌“眉毛胡子一起抓”，事倍功半。知道分数的比例还要了解各个章节的考试难度，经过分析历年试卷，考生应该按照如下策略应对教程：公共基础知识由于是考查广度而不是深度，所以遵循“广撒网”的策略，达到了解程度即可；C 程序设计的初步知识、函数和循环结构程序设计要达到理解程度，要看懂主要的知识点；指针与数组要达到掌握程度，力争看懂所有的知识点；其他章节达到了解程度即可。

下面讲一下选择题和填空题的做题方法和技巧。

1. 选择题

这种考题要求考生从 4 个备选答案中选择一个正确答案，应对这类试题，常用方法是直推法和筛选法。

① 直推法：先不分析所给的 4 个答案之间的区别和联系，根据内容直接推出正确答案，然后从 4 个答案中选出相符的一个答案。如：

有以下程序：

```
#include<stdio.h>
main ()
{
    int y=9;
    for ( ; y>0;y--)
        if (y%3==0)
            printf ("%d",--y);
}
```

程序的运行结果是（ ）。[2007 年 9 月 选择第 22 题]

- A. 741 B. 963 C. 852 D. 875421

本题考查对自减运算符的掌握。本题中的 for 循环执行 9 次，当 y 取值分别为 9、6、3 时，if 后的条件语句满足，又因为--y 是先自减 1，然后再返回，所以输出的值分别为 8、5、2，答案选择 C。

② 筛选法：将所给的 4 个答案进行逐一分析、对比、去伪存真、步步筛选并逐一排除，最后确立一个正确答案。如：

可在 C 程序中用做用户标识符的一组标识符是（ ）。[2007 年 9 月 选择第 12 题]

- A. and B. Date C. Hi D. case
_2007 y-m-d Dr.Tom Bigl

本题考查对 C 语言标识符命名规则的掌握。本题中，选项 B 中的“-”、选项 C 中的“.”均不是构成标识符的基本字符，所以错误。选项 D 中标识符的字符组成上没有错误，但 case 是关键字，违反了用户标识符不能与关键字相同的原則，所以也错误。由上述对比、分析可知，本题答案应该选择 A。

当然，有了方法还需要具备一定的实力，这样才可以在选择题中立于不败之地。选择题考查的知识点广而细，考生需要有一定的概念分辨能力，这就要求考生在平时的复习中多做分析、比较和总结。考试中选择题数量比较多，分值分散，考生应该有全局观念，保证大部分题目的得分，遇到不会的题目可以暂时先略过去，等到题目做得多了，知识轮廓打开了，前面不会的题目涉及的知识就会突然想起，这时再返回前面做题，往往有事半功倍的效果。

2. 填空题

填空题主要考查考生对基础知识的准确理解，这类试题一般不会太深，以基础内容为主。碰见这种题，考生应该深

刻理解题意,明确题目要求,运用相关知识做出准确回答。对历年考试试卷分析发现,考生在这种题上得分差别较大,这类题目每题2分,共计15题30分,高分考生可以得到25~30分,而有的考生得分不到10分。所以这种题往往能拉开档次,其得分直接决定考试结果,要想通过二级C语言笔试,一定要保证填空题得到18分左右。其实,填空题得到18分并不是难事,因为除了少量综合题之外,大部分试题还是考核基础知识,而且考试内容也离不开教程,所以平时多看看书,重点理解和记忆考试热点内容,只要看得细、记得牢,就不需要为此担心。考生常犯的错误有以下几个方面:

- (1) 基础知识掌握不全面,概念理解不准确。
- (2) 答案表述不准确,会做的题得分不多或不得分。
- (3) 答题卡填写位置错误,答案张冠李戴,白白丢分,十分可惜。
- (4) 知道答案是什么,但是文字写错,没有得分。

考生把自己的情况和上述错误对比,发现自己的不足,及时改进,争取考试分数有一定的提高。

做历年的真题很重要,模拟题再好,也达不到真题的深度和广度,不要只做模拟题,而忽视了历年真题。做历年真题有利于考生把握知识点和出题方式,从宏观上了解到要考哪些内容和怎样考查这些内容。同时也可以巩固知识,更重要的是,二级C语言试题重现率比较高,很多题目在以前的考试中都能看到类似影子,其实这也是必然的,因为考来考去就那么多个知识点。所以只要熟悉并理解往年试题,应对二级C语言绰绰有余。

当然,还要防止思维定式,有时考试中心把以前试题的考查方式、考查内容变化一下重新出现。这些改编的试题往往存在陷阱,试题表面看起来相同,其实答案已经不同了,一不留心就会出错,每年都有考生在熟悉题上出错的事例。遇到这类试题不要认为以前见过就沾沾自喜、放松警惕、匆匆而过,一定要仔细读题,切实理解题意,切忌浮躁。

书上的课后习题也十分重要,对二级C语言而言,课后题的难度和考试试题的难度相当,如果教程上的课后题可以独立做出,那么考试也应该没有什么问题。

二级C语言考试中,C语言实践题比较多,这是由二级C语言考试的重点和特点所决定的。常考类型有计算程序结果、程序填空、程序段分析等。这些内容难度一般,考生应该利用纸和笔细致地分析和模拟程序实际执行,得出最终结果,而不是仅仅依靠头脑中的想象。每年都有很多学生眼高手低,思路正确但是得不出正确结果,令人可惜。此外,做好这些题还需要考生具有一定的实践能力,所以考生在理解基本知识的同时,应经常上机实践,将所学知识融会贯通。

2004年考试大纲调整之后,二级考试中基础知识的考核内容由以前的一级难度调整为三级难度,增加了考试的难度。由数据结构与算法、程序设计基础、软件工程基础、数据库设计基础四部分组成的公共基础知识成为二级所有考试中不可缺少的一部分,在考卷中固定为10道选择题和5道填空题,占有30分,比重不小。其中,数据结构与算法是需要理解的内容,比较灵活,切忌死记硬背,其余为软件开发基础,需要理解记忆。这些知识较难理解,但对软件开发具有很强的指导作用,考生如能掌握,会更好地开发程序。

考试前要预先熟悉环境,最好考试前一天就到考场看一下,熟悉地点和路线。每次考试都有考生不能及时找到自己的考点,浪费了时间,从而影响考试。同时考试时要预先到场,并且检查考试所需物品是否齐全。

进入考场,考生要保持良好的精神状态和心态。适度紧张是必然的,也有助于集中注意力,但是过分紧张和焦虑对整个考试非常不利,直接影响考生的临场发挥。考生考场中要学会调整自己的心态,如深呼吸、转移注意力等。二级考试时间90分钟,考试形式只有选择题和填空题,时间绝对够用;而且考试是对平时学习的评测和总结,成败与否在于平时,而不是考试中的侥幸,放开心态,尽量不要受其他因素的影响。

1.3 机试方法和技巧

二级C语言机试时间定为90分钟,考题分三部分:程序填空、程序修改和程序设计,分别占30分、30分和40分,所用操作系统为Windows XP,满分100分,60分合格。其中,程序填空和程序修改按照考生答对的数目分步给分,关键在于及时保存;而程序设计题按照最终正确结果的数目按比例给分,关键在于运行。从难度上比较,机试比笔试容易,一个熟练的程序员做完上机的题目只需不到10分钟,而做完笔试的题目至少需要1个小时。然而从历年的考试情况来看,上机通过率要低于笔试通过率,每年都有不少人千辛万苦过了笔试,却在机试上翻船,这说明大部分考生的实践能力还是不行。所以,考生需要掌握C语言基础知识,理解和熟悉“南开100题”中各种题型的解法,经常上机练习,调试程序,经过这样一番训练,上机通过应该不是问题。

首先要掌握C语言基础知识,为上机打下良好的理论基础。C语言知识庞大而复杂,但是二级C语言上机考试中所占的内容不多。掌握以下几点,足够应付二级C语言上机考试。

(1) 教材的选择:谭浩强编写的《C程序设计》是学习C语言的经典之作,初学者应该以此为入门,进入C语言殿堂,有一定编程基础者可以把此书作为上机调试的参考书。

(2) 掌握C语言语法:语法是编程的基础,在很多情况下都是因为一些细节性问题,如括号写错、分号写错等,导致程序调试不出,个人由于思维惯性检查多次也看不出问题。对于有过其他语言编程基础的考生而言,还要熟悉C语言和其他编程语言的区别,否则在考试中极易由于编程语言交叉而出错。

(3) 深刻理解C语言的运算符和运算顺序:C语言运算灵活,功能丰富,运算种类远多于其他语言,在表达式方面较其他程序语言更为简洁,如自加、自减、逗号运算和三目运算使表达式更为简单。但这种简洁性和灵活性又给初学者带来一定的难度,不易掌握,主要原因就是没有深刻理解C语言运算符和运算顺序。

(4) 正确定义数据类型:数据类型是编程的最小单元,如果数据类型定义出错,程序的正确性就难以保证。

(5) 掌握 C 语言 3 种控制结构：顺序、选择和循环，它们不仅是 C 语言，而且是所有程序设计语言从微观角度来看的 3 种控制结构。熟练掌握这 3 种控制结构，加上正确定义的数据类型，就可以编出规范的程序，满足上机考试的要求。

(6) 了解一些常用操作和简单算法的实现，如排序、数制转换、字符串运算、数组运算、指针操作等。由于二级 C 语言上机对上述内容要求深度不高，加上 60% 分值的考题都以程序阅读为主，实际编写的代码也不多，所以上述内容虽然多，但是难度不大。了解上述内容，可以加快程序阅读和编制速度。

其次要多做题并上机调试，先做一些基础编程题，巩固 C 语言基础知识，熟悉 C 语言编程环境，待掌握差不多时，就可以进入实战阶段。因为在编程中学懂不等于学会，所以要想把知识真正转化为能力、应对二级 C 语言上机考试，必须多上机、多调试。编程时一定要先画出 N-S 图，有了明确的思路以后再编写程序，不管多么简单的程序都要保证调试成功，眼高手低也是上机考试失败的原因之一。我们在希赛网上为考生提供了上机考试系统，完全模拟考试上机环境，考生可以通过该系统熟悉上机环境，提高应试能力。

刚开始使用该系统时，要采用“广撒网”原则，不管对错，把系统中的习题都练习一下，了解自己的实际优劣情况。对于自己的强项，可以略去不做，每隔一段时间找些典型习题做一下巩固即可；对于自己的弱项，首先要对照结果仔细分析理解，然后专门归类总结，重点突击上机调试，经过多次“理解—编程—调试”的过程，这些内容已经融入考生的脑中，应对考试已经不成问题。

编程过程中，考生要切记以下几个方面，以免出现不必要的错误。

(1) C 语言中首地址是 0，而不是 1。

(2) 字符串常量用双引号标识，而字符常量用单引号标识。

(3) 语句或预定义的函数名后一定要带分号。

(4) 函数和变量使用前一定要定义。

(5) else 嵌套语句中的最近匹配原则。

(6) 检查括号是否配对正确。

(7) 小心拼写错误，例如，将 else 写成 esle、continue 拼写成 contiune 等。

(8) break 和 continue 从循环体中退出时的含义不同，break 是退出而不再执行，continue 是退出本次循环转而执行下一次循环。

(9) “=” 和 “==” 含义不同，前者表示赋值，后者是逻辑比较中的等于符号。

(10) p++ 与 ++p 含义不一样，前者是变量先使用再增加，后者是变量先增加再使用。

参加上机考试时，除了知道上机考试的一般纪律及步骤外，为了防止意外情况导致上机考试失败，还要注意下面几点。

(1) 一定要带好准考证和身份证。上机考试时一定要输入准考证号是否正确，输入后要看一下系统显示的考生姓名和考生身份证号是不是自己的，否则将成为替别人考试。

(2) 登录以后会出现类似“开始考试并计时”的按钮，此时并没有真正进入考试计时阶段，要单击该按钮后才开始考试。在这段时间考生可以调节一下自己的情绪，以便适应考试。但调节时间不宜过长，否则监考老师会不愿意。

(3) 进入考试环境后，不要先急着做题，应该先分析题型，等到心中有数后再打开 VC 环境做题。做题时要保持一颗“平常心”，遇到简单题不能骄傲，碰见难题也不能灰心。做题时要注意题目中给出的源程序文件名，以免弄错程序做错题。

(4) 考试过程中一定不要退出“考生目录”，一定要在“C:*****\准考证号”的目录下做题，否则没有分数。

(5) 题目做完后，一定要根据试题说明仔细检查几遍，发现没有错误后再保存、运行、看结果。注意一定要先保存，再运行程序，否则容易发生意外而丢失程序，发现错误修改源程序时也应该按照这个顺序。程序的结果以最后一次运行结果为准，前面的结果会被后面的覆盖。

(6) 编写程序时要养成按照“缩进”格式编写的习惯，考试虽然没做这方面的要求，但这是一种良好的编程习惯，有利于搞清程序结构、检查代码，如果考生以后从事计算机编程方面的职业，掌握这点十分必要。

(7) 交卷时一定要遵循先关闭 VC 环境，再单击【交卷】按钮交卷的顺序。关闭 VC 环境时要按“Alt+F”快捷键，不要用鼠标单击窗口右上角的【关闭】按钮，否则有可能出错。

虽然上机和笔试试题形式有一定差异，但是考查内容都是 C 语言程序设计的基础知识，只不过上机侧重实际应用，而笔试侧重书面考核。因此，考生在备战上机考试时，一定要掌握计算机语言的基本知识，提高自己阅读、理解和完善程序的能力。计算机语言其实都是相通的，把一种语言熟练掌握后，学习其他语言就可以减轻不少负担，对考生以后从事计算机编程职业也有一定的帮助作用。上机考试要多动手，只有经过无数次的分析和调试，才能熟悉考试环境，深刻把握计算机语言的特点，从根本上填平理论和实际之间的鸿沟，防止由于“眼高手低”而导致的失误，从而顺利通过二级 C 语言上机考试。