

National Computer Rank Examination
全国计算机等级考试

二级教程

基础知识和C++语言程序设计

李季 高磊 编著



南开大学出版社

全国计算机等级考试系列

二级教程

基础知识和 C++ 语言程序设计

李 季 高 磊 编著

南开大学出版社

天津

图书在版编目(CIP)数据

全国计算机等级考试二级教程. 基础知识和 C++ 语言
程序设计 / 李季等编著. —天津: 南开大学出版社,

2004. 10

(全国计算机等级考试系列)

ISBN 7-310-02150-9

I. 全... II. 李... III. ①电子计算机—水平考试
—教材②C 语言—程序设计—水平考试—教材
IV. TP3

中国版本图书馆 CIP 数据核字(2004)第 071872 号

版权所有 翻印必究

南开大学出版社出版发行

出版人: 肖占鹏

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022)23508339 23500755

营销部传真: (022)23508542 邮购部电话: (022)23502200

*

河北昌黎太阳红彩色印刷有限责任公司印刷

全国各地新华书店经销

*

2004 年 10 月第 1 版 2004 年 10 月第 1 次印刷

787×1092 毫米 16 开本 19.75 印张 502 千字

定价: 28.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022)23507125

前言

1994年,国家教育部推出了面向社会的全国计算机等级考试。10年来,为了促进计算机知识的普及和计算机应用技术的推广,满足用人单位考核工作人员应用计算机的水平,全国计算机等级考试也在不断完善和修订。全国计算机等级考试最新版考试大纲在2004年开始推行。新考纲中,除了取消过时的考试科目和调整一些科目的考试内容之外,在二级考试科目中,新增了目前较为流行的计算机语言C++、Java和Access。

为了满足考生的要求,在全国计算机等级考试中心有关专家的指导下,我们严格按照新考试大纲的要求,组织了既熟悉等级考试又精通专业技术的强大编写队伍,专门针对新增的C++考试科目编写了本书。

本教程针对准备参加全国计算机等级考试二级C++语言程序设计的考生。帮助考生快速掌握C++语言的设计方法,提高学习效率,在较短的时间内轻松通过考试。

考试基本要求

全国计算机等级考试二级(C++语言程序设计)分为上机考试和笔试两部分,主要内容涉及公共基础知识和C++语言程序设计。

公共基础知识部分对于考生的要求是:掌握算法的基本概念;掌握基本数据结构及其操作;掌握基本排序和查找算法;掌握逐步求精的结构化程序设计方法;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;掌握数据库的基本知识,了解关系数据库的设计。

C++语言程序设计部分对于考生的基本要求是:掌握C++语言的基本语法规则;熟练掌握有关类与对象的相关知识;能够阅读和分析C++程序;能够采用面向对象的编程思路和方法编写应用程序;能熟练使用Visual C++ 6.0集成开发环境编写和调试程序。

笔试时间为90分钟,满分100分(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,占30分);上机考试90分钟,满分100分。

本书的特点

专家构成的作者队伍

本书由长期从事全国计算机等级考试授课、辅导和C++语言程序设计的专家共同编写,并得到考试中心有关专家的详细指点,内容紧扣考试大纲,书中的习题模拟考试题,避免读者走弯路,提高学习效率。

知识讲解详细清晰

本书严格按照等级考试的大纲要求,由浅入深、循序渐进地介绍C++语言程序设计技

术，并针对考点、重点和难点，进行详细的讲解。

大量的例题分析和讲解

为了让读者真正掌握 C++ 语言的编程技能，本书在介绍理论知识的同时，辅以大量程序设计例题，并进行详尽的分析和说明。

课后习题测试掌握程度

在每章的最后，都有一个习题部分，在此可以测试读者对该章内容的掌握程度。在附录中，给出了各章习题的参考答案。

本书作者

本书主要由李季、高磊主笔，另外，在编写过程中，邱代燕、宫晓琳、黄志雄、董岚枫、徐增辉、刘刚、刘朋、尚卫平、赵成璧、孙宏、于樊鹏、陈河南、贺军、梁德成、梁彩隆、廖明武、倪永智、贺民、李志云、戴军、陈安南、李晓春、王春桥、王雷、郭涛、韦笑、龚亚萍、孟丽艳、石丽霞等人，在资料整理、查错、代码调试等方面做了大量的工作。

本书的所有习题代码，均经过上机调试通过。读者如果需要，或者有其他意见和建议，可以给如下地址发邮件。

xiaoxiang-007@sohu.com

目 录

第一部分 公共基础知识

第 1 章 基本数据结构与算法.....	1
1.1 基本数据结构.....	1
1.1.1 什么是数据结构.....	1
1.1.2 逻辑结构.....	2
1.1.3 存储结构（物理结构）.....	2
1.1.4 数据的运算.....	3
1.1.5 数据结构的图形表示.....	3
1.2 算法.....	3
1.2.1 什么是算法.....	3
1.2.2 算法复杂度.....	4
1.3 线性表.....	4
1.3.1 什么是线性表.....	4
1.3.2 线性表的顺序存储结构.....	4
1.3.3 线性表的插入与删除运算.....	5
1.4 栈和队列.....	5
1.4.1 什么是栈和队列.....	5
1.4.2 栈的顺序存储结构及其基本运算.....	5
1.4.3 队列的顺序存储结构及其基本运算.....	6
1.5 线性单链表、循环链表与双向链表.....	6
1.5.1 线性单链表.....	6
1.5.2 循环链表.....	7
1.5.3 双向链表.....	7
1.5.4 链表的基本操作.....	7
1.6 树.....	8
1.6.1 什么是树.....	8
1.6.2 二叉树及其存储结构.....	9
1.6.3 二叉树的遍历.....	11
1.7 查找.....	11
1.7.1 顺序查找.....	11
1.7.2 二分查找算法.....	11
1.8 排序.....	12
1.8.1 交换排序.....	12
1.8.2 选择类排序.....	12

1.8.3 插入类排序	12
1.8.4 各种排序方法	12
本章小结	13
习题一	13
第 2 章 程序设计基础	16
2.1 程序设计	16
2.1.1 程序设计的方法	16
2.1.2 程序设计的风格	16
2.2 结构化程序设计	19
2.2.1 结构化程序设计的基本特征	19
2.2.2 结构化程序设计的主要原则	19
2.3 面向对象的程序设计方法	20
2.3.1 面向对象设计	20
2.3.2 面向对象方法的特点	20
2.3.3 面相对象程序设计的相关概念	20
2.3.4 面向对象分析与模型化	21
本章小结	22
习题二	22
第 3 章 软件工程基础	25
3.1 软件工程概述	25
3.1.1 软件工程	25
3.1.2 软件生命周期	25
3.1.3 软件工具与软件开发环境	25
3.2 结构化分析与设计	25
3.2.1 结构化分析方法	26
3.2.2 数据流图	26
3.2.3 数据字典	27
3.2.4 软件需求	27
3.2.5 结构化设计	27
3.2.6 总体设计	27
3.2.7 详细设计 (过程设计)	28
3.3 软件测试与程序调试	28
3.3.1 软件测试的目的	28
3.3.2 测试用例设计	28
3.3.3 软件测试的实施	29
3.3.4 程序调试	30
3.3.5 常见调试方法	30
本章小结	30
习题三	31

第 4 章 数据库设计基础	33
4.1 数据库与数据模型.....	33
4.1.1 数据库.....	33
4.1.2 数据库管理系统.....	33
4.1.3 数据库系统.....	33
4.1.4 实体联系模型.....	33
4.1.5 E-R 图.....	33
4.1.6 关系数据模型.....	35
4.1.7 E-R 模型到关系模型的转换.....	35
4.1.8 关系代数运算.....	36
4.1.9 数据库的规范化.....	36
4.2 数据库设计.....	37
4.2.1 需求分析.....	37
4.2.2 概念设计.....	38
4.2.3 逻辑设计.....	38
4.2.4 物理设计.....	38
本章小结.....	39
习题四.....	40

第二部分 C++语言程序设计

第 5 章 C++语言概述	42
5.1 计算机及其语言的发展.....	42
5.2 程序设计方法.....	43
5.2.1 结构化程序设计.....	43
5.2.2 面向对象程序设计.....	44
5.3 C++语言的发展历史.....	45
5.4 简单例子分析.....	46
5.4.1 C++的程序框架结构.....	46
5.4.2 C++的函数.....	47
5.4.3 C++源程序的结构特点和书写规则.....	49
5.5 C++语言程序的编写和运行.....	50
5.6 Microsoft Visual C++简介.....	51
5.6.1 Visual C++ 6.0 集成开发环境.....	51
5.6.2 编写和编译程序.....	51
本章小结.....	55
习题五.....	55
第 6 章 C++词汇、数据类型、运算符、表达式	59
6.1 C++语言的词汇.....	59

6.1.1	字符集	59
6.1.2	C++ 语言词汇	59
6.2	C++ 语言的数据类型	61
6.2.1	基本数据类型	61
6.2.2	构造数据类型	62
6.2.3	指针类型	62
6.2.4	空类型	62
6.2.5	类类型	62
6.3	常量	63
6.3.1	整型常量	63
6.3.2	实型常量	64
6.3.3	字符常量与字符串常量	65
6.3.4	枚举常量	66
6.3.5	符号型常量	67
6.3.6	逻辑型常量	67
6.4	变量	68
6.4.1	变量的命名	68
6.4.2	变量定义方式	68
6.4.3	变量初始化	68
6.4.4	整型变量	69
6.4.5	实型变量	69
6.4.6	字符变量	70
6.5	运算符和表达式	70
6.5.1	表达式	70
6.5.2	运算符的分类	71
6.5.3	运算符的优先级和结合性	71
6.5.4	数据类型转换	72
6.6	表达式类型和求值规则	74
6.6.1	赋值运算符与赋值表达式	74
6.6.2	算术运算符和算术表达式	75
6.6.3	关系运算符和关系表达式	76
6.6.4	逻辑运算符和逻辑表达式	77
6.6.5	条件运算符和条件表达式	78
6.6.6	逗号运算符和逗号表达式	79
6.6.7	自增、自减运算	80
6.6.8	位运算	80
6.6.9	求字节运算	80
6.6.10	复杂表达式求值	81
	本章小结	82

习题六.....	82
第7章 基本语句.....	87
7.1 常见语句.....	87
7.1.1 表达式语句.....	87
7.1.2 复合语句.....	87
7.1.3 空语句.....	88
7.2 输入与输出.....	88
7.2.1 输入.....	88
7.2.2 输出.....	89
7.2.3 常用的控制符.....	89
7.3 程序的基本结构.....	92
7.3.1 顺序结构.....	92
7.3.2 选择结构.....	92
7.3.3 循环结构.....	93
7.4 选择语句.....	93
7.4.1 if...else 语句.....	93
7.4.2 switch 语句.....	97
7.5 循环语句.....	99
7.5.1 for 语句.....	99
7.5.2 while 语句.....	102
7.5.3 do...while 语句.....	103
7.5.4 循环语句的嵌套.....	105
7.6 转向语句.....	106
7.6.1 break 语句.....	106
7.6.2 continue 语句.....	107
7.6.3 goto 语句.....	108
7.7 分支语句和循环语句的嵌套应用.....	109
7.7.1 程序应用 1: 素数判定.....	109
7.7.2 程序应用 2: 猴子吃桃.....	110
7.7.3 程序应用 3: 二分法解方程.....	111
7.7.4 程序应用 4: 最大公约数和最小公倍数.....	112
本章小结.....	112
习题七.....	113
第8章 数组和字符串.....	120
8.1 一维数组的定义、初始化和访问.....	120
8.1.1 一维数组的定义.....	120
8.1.2 一维数组的初始化.....	121
8.1.3 一维数组的访问.....	121
8.1.4 一维数组的赋值与引用.....	122

8.1.5 一维数组的应用	123
8.2 多维数组的定义、初始化和访问	125
8.2.1 二维数组的定义	125
8.2.2 二维数组的初始化	126
8.2.3 二维数组的访问	126
8.2.4 二维数组的赋值与引用	127
8.2.5 多维数组的定义与初始化	128
8.2.6 二维数组的应用	128
8.3 数组类型的定义和访问	129
8.4 字符串与字符数组	130
8.4.1 字符数组的定义	130
8.4.2 字符数组的初始化	130
8.4.3 字符数组的引用	131
8.4.4 字符串	131
8.5 常用字符串函数	132
8.6 数组应用	135
8.6.1 选择排序算法	135
8.6.2 插入排序算法	136
8.6.3 逆序输出	138
8.6.4 简单数据库	138
本章小结	139
习题八	140
第9章 函数	147
9.1 基本概念	147
9.1.1 库函数	147
9.1.2 程序结构	148
9.2 函数声明、定义和返回值	148
9.2.1 函数的声明	149
9.2.2 函数的定义	150
9.2.3 函数的返回值	151
9.3 函数的调用	152
9.3.1 实参和形参	152
9.3.2 函数调用的格式	153
9.4 变量的作用域	154
9.4.1 局部变量	154
9.4.2 全局变量	155
9.5 变量的存储类型	157
9.5.1 静态存储和动态存储	157
9.5.2 自动变量	158

9.5.3 外部变量	160
9.5.4 静态变量	160
9.5.5 寄存器变量	162
9.6 函数的递归调用	163
9.7 数组与函数	166
9.7.1 数组元素作为函数的参数	166
9.7.2 数组名作为函数的参数	167
9.8 重载函数	168
9.9 内联函数	169
本章小结	170
习题九	170
第 10 章 指针、引用及结构体	179
10.1 指针的概念	179
10.1.1 指针的基本概念	179
10.1.2 指针变量的类型说明	180
10.2 指针的基本操作	180
10.2.1 指针变量的赋值	180
10.2.2 指针变量的运算	181
10.2.3 const 指针	182
10.2.4 void 指针	183
10.3 指针与数组	183
10.3.1 数组指针变量	184
10.3.2 数组名和数组指针变量作函数参数	185
10.3.3 多维数组地址的表示方法	185
10.3.4 多维数组的指针变量	187
10.4 字符指针与字符串	187
10.4.1 字符串指针变量的说明和使用	187
10.4.2 字符串指针与函数参数	189
10.4.3 使用字符串指针变量与字符数组的区别	189
10.5 指针与函数	190
10.5.1 函数指针变量	190
10.5.2 指针型函数	191
10.5.3 指针与函数参数	192
10.6 指针数组	193
10.7 引用	196
10.7.1 引用的基本概念	196
10.7.2 引用与函数	196
10.7.3 常引用	197
10.8 结构体	198

10.8.1 结构体的概念	198
10.8.2 结构体类型变量的定义方法	199
10.8.3 结构体变量的使用形式	200
10.8.4 结构体变量的初始化	201
10.9 结构体数组与指针	202
10.9.1 结构体数组的定义	202
10.9.2 结构体数组的初始化	203
10.9.3 结构体数组的使用	203
10.9.4 结构体指针	204
10.10 综合应用举例	205
10.10.1 结构体应用	205
10.10.2 指针应用	205
10.10.3 引用应用	206
本章小结	206
习题十	207
第 11 章 类与对象	214
11.1 类和对象	214
11.1.1 类的定义	214
11.1.2 类的数据成员	215
11.1.3 类的成员函数	216
11.1.4 类和结构	217
11.2 对象	217
11.2.1 对象的定义格式	217
11.2.2 对象成员的表示	218
11.3 构造函数和析构函数	219
11.3.1 构造函数	219
11.3.2 析构函数	221
11.4 静态数据成员与静态成员函数	222
11.4.1 静态数据成员	222
11.4.2 静态成员函数	223
11.4.3 静态成员的应用	223
11.5 常成员函数与常数据成员	225
11.5.1 常成员函数	225
11.5.2 常数据成员	225
11.6 this 指针	226
11.7 友元函数与友元类	227
11.7.1 友元函数	227
11.7.2 友元类	229
11.8 对象数组和成员对象	230

11.8.1 对象数组	230
11.8.2 成员对象	231
11.8.3 堆对象与对象数组	232
本章小结	234
习题十一	234
第 12 章 类的继承与派生	240
12.1 派生类	240
12.1.1 继承的基本概念	240
12.1.2 基类与派生类的关系	241
12.1.3 派生类的定义	242
12.2 继承方式与访问权限	242
12.2.1 公有继承	242
12.2.2 私有继承	244
12.2.3 保护继承	245
12.2.4 访问控制	246
12.3 派生类的构造函数与析构函数	246
12.3.1 单继承的构造函数	246
12.3.2 多继承的构造函数	248
12.3.3 析构函数	249
12.3.4 二义性和支配原则	250
12.4 基类指针与派生类指针	252
12.4.1 赋值兼容规则	252
12.4.2 基类指针	252
12.5 虚基类	253
12.5.1 虚基类的引入	253
12.5.2 虚基类的构造函数	254
本章小结	257
习题十二	257
第 13 章 多态性和类模板	263
13.1 多态性概念	263
13.2 成员函数重载	263
13.2.1 成员函数的重载	263
13.2.2 有关重载的注意事项	264
13.3 运算符重载	265
13.3.1 重载的基本概念	265
13.3.2 重载的定义形式	266
13.4 虚函数	267
13.4.1 基本概念	267
13.4.2 虚函数的调用	269

13.4.3 有关虚函数的注意事项	272
13.5 纯虚函数与抽象类	272
13.5.1 纯虚函数	272
13.5.2 抽象类	273
13.5.3 多态应用	274
13.6 模板概述	275
13.7 函数模板	276
13.7.1 函数模板定义	276
13.7.2 函数模板的使用	276
13.8 类模板	277
13.8.1 类模板说明	277
13.8.2 类模板使用	278
本章小结	279
习题十三	279
第 14 章 输入输出流	286
14.1 C++流概述	286
14.1.1 I/O 流的概念	286
14.1.2 I/O 类库	286
14.1.3 四个类对象	287
14.2 格式化 I/O	288
14.2.1 ios 类中的枚举常量	288
14.2.2 ios 类中的成员函数	289
14.2.3 格式控制操作符	290
14.3 检测流操作的错误	291
14.4 文件流	292
14.4.1 文件的概念	292
14.4.2 文件的打开与关闭	292
14.4.3 文件的读写	293
14.4.4 文件读写实例	294
14.5 字符串流类	295
本章小结	295
习题十四	296
附录 习题参考答案	299

第一部分 公共基础知识

第 1 章 基本数据结构与算法

如果问一个木匠的学徒工：“你工作时用什么工具呢？”他可能回答：“一把锤子和一把锯子”。但如果去问老木匠这个问题，他会说：“我需要一些精确的工具。”同样，由于计算机所解决的问题都是从生活中抽象出来的问题，其复杂性与多样性不言而喻，所以需要精确有效的工具去解决现实生活中的复杂问题。算法、数据结构、程序设计语言都是这样的工具。瑞士的计算机学者 Niklaus Wirth 早在 1976 年就曾这样定义程序：

算法 + 数据结构 = 程序设计

很快，这就成了在计算机工作者之间流传的一句名言，从中不难看出算法与数据结构的重要性。本章将介绍有关数据结构与算法的基本知识。

1.1 基本数据结构

1.1.1 什么是数据结构

简单地说，数据结构就是数据模型。一般说来，用计算机解决一个具体问题时，大致需要经历下列几个步骤：

- (1) 首先要从具体问题抽象出一个适当的数学模型。
- (2) 然后，设计一个解此数学模型的算法。
- (3) 最后编出程序，进行测试、调整，直至得到最终解答。

寻求数学模型的实质是分析问题，从中提取操作的对象，并找出这些操作对象之间含有的关系，然后用数学的语言加以描述。例如，求解梁架结构应力的数学模型为线性方程组；预报人口增长情况的数学模型为微分方程。然而，更多的非数值计算问题无法用数学方程加以描述。请看下面一个非数值计算问题的例子。

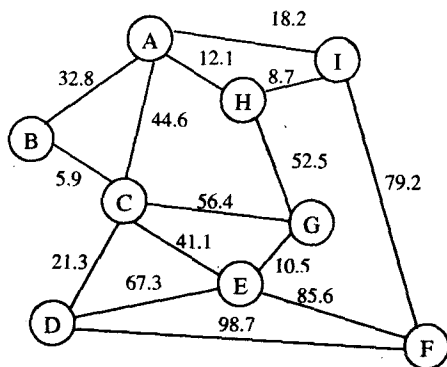
【例 1】铺设城市煤气管道的问题。

假设要在某个城市的 n 个居民区之间铺设煤气管道，则在这 n 个居民区之间只要铺设 $n-1$ 条管道即可。假设任意两个居民区之间都可以铺设管道，但由于地理环境的不同，所需经费也不同，如何规划才能使得总投资花费最少，这个问题即为“求图的最小生成树”的问题。其数学模型为如图 1-1 所示的“图”，图中“顶点”表示居民区，顶点之间的连线及其上的数值表示可以架设的管道及所需经费。求解的算法为在可能假设的 m 条管道中选

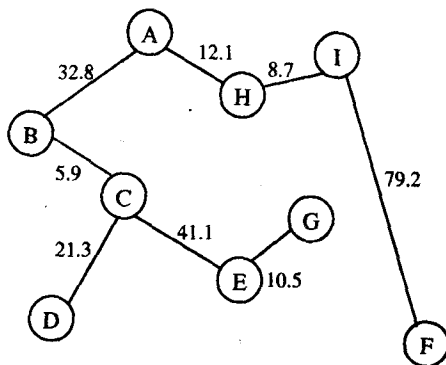
取 $n-1$ 条, 既能连通 n 个居民区, 又使总投资达到“最小”。

通常, 这类涉及道路、线路问题的数学模型是一种所谓“图”的数据结构。对于其他的非数值计算问题, 还可能用其他诸如表、树之类的数据结构来描述和解决。

在给出数据结构的定义之前, 首先要知道什么是数据。所谓数据, 就是所有能被输入到计算机中, 且能被计算机处理的符号(数值、字符等)的集合, 是计算机操作的对象总称。



(a) 居民区示意图



(b) 铺设煤气管道设计图

图 1-1 图及最小生成树示例

而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。数据结构一般包括以下三方面的内容：数据的逻辑结构、数据的物理结构、数据的运算。

1.1.2 逻辑结构

逻辑结构即数据元素之间的逻辑关系。它可以用一个数据元素的集合和定义在此集合上的若干关系来表示。

数据的逻辑结构是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。它有两类：线性结构和存储结构。

1. 线性结构

线性结构的逻辑特征是, 在数据元素的非空有限集中:

- 存在惟一的一个被称作“第一个”的数据元素;
- 存在惟一的一个被称作“最后一个”的数据元素;
- 除第一个之外, 集合中的每个数据元素均只有一个前驱;
- 除最后一个之外, 集合中的每个数据元素均只有一个后继。

例如, 线性表是一个线性结构。

2. 非线性结构

非线性结构的逻辑特征是: 一个结点可能有多个直接前驱和直接后继。例如: 树和图都是非线性结构。

1.1.3 存储结构(物理结构)

数据元素及其关系在计算机存储器内的表示——数据的存储结构是逻辑结构用计算机语言的实现, 即建立数据的机内表示。存储结构主要内容指在存储空间中使用一个存储结