

高等学校计算机基础教育规划教材

Visual C++程序设计

张文波 主编



清华大学出版社

高等学校计算机基础教育规划教材

Visual C++程序设计

张文波 主编
陈晓明 周明 兰丽辉 副主编

清华大学出版社
北京

内 容 简 介

本书介绍了 Visual C++ 程序设计的基本知识,主要内容包括 C++ 的基本词法和语法规则、基本数据类型和表达式、程序控制结构、函数定义和调用、类与对象、类的继承和派生、多态和虚函数、对话框和标准控件等。

本书内容安排由浅入深,概念明确,语言简洁,重点突出,可作为高等院校非计算机专业的教材,也可作为 C++ 爱好者的参考书。

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

Visual C++ 程序设计 / 张文波主编. —北京:清华大学出版社, 2010.9

(高等学校计算机基础教育规划教材)

ISBN 978-7-302-23038-0

I. ①V… II. ①张… III. ①C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2010)第 112137 号

责任编辑:袁勤勇 张为民

责任校对:焦丽丽

责任印制:李红英

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:010-62795954, jsjic@tup.tsinghua.edu.cn

、 质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185×260 印 张:17.75

字 数:405 千字

版 次:2010 年 9 月第 1 版

印 次:2010 年 9 月第 1 次印刷

印 数:1~3000

定 价:26.00 元

产品编号:038478-01

教材编写委员会

主 任：滕国文

副主任：李 政 于晓鹏 张文波 梁海英

成 员：王洪君 董延华 王大东 陈晓明 兰丽辉

赵 瑞 韩玉彬 朱 宏 周 明 张运林

李 颖 孙 静 白文秀 丛 飏 逯 洋

刘 伟 李淑梅 张桂杰 王海燕 蓝 鹰

英昌盛 姚建盛 叶丽娜 刘 松 张 伟

李 闯 刘 哲 王继魁 李丽颖 梁 微

赵靖华 邹晓辉 赵 鹏 籍卓佳 吕 凯

王发斌

序

高等学校计算机基础教学是为非计算机专业学生提供计算机知识、能力与素质的教育,使学生掌握计算机、网络以及其他相关信息技术的基本知识,培养学生利用计算机分析问题、解决问题的能力,提高学生的计算机文化素养,为将来运用计算机知识与技术解决自己专业实际问题打下基础。

飞速发展的计算机技术和日益普及的计算机应用,对高等学校非计算机专业的计算机教学提出了越来越高的要求。计算机技术水平的高低和计算机应用能力的强弱,已经成为衡量大学毕业生素质和水平的重要尺度。为了提高计算机基础教学质量,教育部高等学校非计算机专业的计算机课程教学指导分委员会于2006年提出了“关于进一步加强高等学校计算机基础教学的意见”,被大家称为“计算机基础教学白皮书”(简称白皮书)。白皮书在“计算机基础教学内容的知识结构与课程设置”中提出了四个领域、三个层次、六门核心课程和“1+X”的课程方案。

四个领域:

- ① 计算机系统与平台。涉及计算机硬件结构、操作系统、网络等方面基础知识和应用技能。
- ② 计算机程序设计。涉及程序设计基本方法、数据结构与算法基础等。
- ③ 数据分析与信息处理。涉及应用计算机系统进行数据分析与信息处理的技术与方法,包括数据库应用、多媒体与人机交互技术,计算机接口与控制技术等。
- ④ 信息系统开发。涉及信息应用系统的设计方法、软件开发等内容。

三个层次:

- ① 概念性基础。大学生必备的计算机通识性基础知识,主要有计算机硬件技术基础、计算机软件技术基础、操作系统概述、网络与分布式计算机介绍、信息系统安全基础。
- ② 技术与方法。具有共性的知识领域中的一些技术与方法,主要有程序的控制结构、基本数据类型与数据结构、面向对象程序设计方法等。
- ③ 应用技能。涉及实际应用及开发中需要掌握的一些应用性技能,主要有程序的测试与调试技能、可视化编程环境的使用等。

六门核心课程:

- ① 大学计算机基础
- ② 计算机程序设计
- ③ 计算机硬件技术基础

④ 数据库技术与应用

⑤ 多媒体技术与应用

⑥ 网络技术与应用

“1+X”的课程方案：

就是指 1 门“大学计算机基础”必修课，再加上几门程序设计或应用性课程（必修或选修）。

根据“白皮书”的指导精神，在已出版的前两套系列教材的基础上，我们组织具有多年计算机共同课教学经验的一线教师，又一次编写了适用于“1+X”课程方案的《大学计算机》系列教材。

本套教材共分四册：

① 计算机应用基础

② Visual Basic 程序设计

③ Visual FoxPro 程序设计

④ Visual C++ 程序设计

《大学计算机》教材编写委员会

2010 年 4 月

前言

教育部高等学校非计算机专业计算机课程教学指导分委员会提出了在新形势下进一步加强高校计算机基础教学的意见,给出计算机基础教学的总体构架,即计算机基础教学内容的知识结构与课程设置。提出了“1+X”的课程方案,即1门“大学计算机基础”(必修)加上几门重点课程(必修或选修)。

Visual C++ 是一门功能强大的计算机程序设计语言,既适合结构化程序设计,又适合面向对象程序设计。通过对本书的学习,读者能够正确理解面向对象编程方法,基本掌握 Visual C++ 中的语法、词法、程序基本结构,并能够利用 Visual C++ 编写简单程序。本书内容共分两大部分,第一部分主要介绍 C++ 的基本知识,内容包括 C++ 的基本词法和语法规则、基本数据类型和表达式、程序控制结构、数组的定义和使用、函数的定义和调用格式、指针等;第二部分主要介绍面向对象程序设计思想和 Windows 可视化编程,主要内容包括类与对象的概念和定义格式、对象的赋值和引用、友元、类的继承和派生、多态和虚函数、对话框和标准控件等。书中所有例题都在 Visual C++ 6.0 的编译系统下运行通过,每章后都附有习题。

本书第1章由陈晓明编写,第2章由周明编写,第3章由兰丽辉编写,第4章由籍卓佳编写,第5章由丛飏编写,第6章由逯洋编写,第7章由英昌盛编写,第8章由刘哲编写,第9章由李丽颖编写,第10章由张文波编写,全书由张文波统稿。

本书在编写过程中参考了有关的 Visual C++ 教材、文献和网站内容,并引用了一些资料,在此对这些作者表示衷心的感谢。

由于本书涉及的内容多、范围广,尽管作者已经尽了最大的努力,自感疏漏难免,敬请读者不吝赐教。

作 者

2010 年 4 月

目录

第 1 章 C++ 概论	1
1.1 C++ 特点	1
1.1.1 程序和程序设计语言	1
1.1.2 C++ 的特点	2
1.2 C++ 程序的实现	4
1.3 C++ 程序结构的特点	6
1.3.1 一个简单的 C++ 程序	6
1.3.2 C++ 程序结构及书写格式	7
1.4 Visual C++ 6.0 主窗口	9
1.5 C++ 上机过程	10
习题 1	13
第 2 章 数据类型、运算符和表达式	15
2.1 基本数据类型	15
2.1.1 整型	15
2.1.2 字符型	16
2.1.3 浮点型	16
2.1.4 布尔型	17
2.1.5 空型	17
2.2 常量和变量	17
2.2.1 常量	17
2.2.2 变量	19
2.3 输入输出	24
2.4 运算符和表达式	28
2.4.1 算术运算符和算术表达式	29
2.4.2 关系运算符与关系表达式	30
2.4.3 逻辑运算符和逻辑表达式	31

2.4.4	位运算符和位运算表达式	32
2.4.5	赋值运算符和赋值表达式	34
2.4.6	逗号运算符和逗号表达式	35
2.4.7	sizeof 运算符	36
2.5	数据类型的转换	37
2.5.1	自动数据类型转换	37
2.5.2	强制数据类型转换	37
2.6	构造数据类型	38
2.6.1	结构体	38
2.6.2	共用体	44
2.6.3	枚举	47
2.6.4	自定义数据类型	49
习题 2		50

第 3 章 控制结构 52

3.1	顺序结构	52
3.2	选择结构	53
3.2.1	if 语句	54
3.2.2	switch 语句	59
3.3	循环结构	61
3.3.1	while 语句	62
3.3.2	do...while 语句	64
3.3.3	for 语句	65
3.3.4	goto 语句	69
3.3.5	break 和 continue	69
3.4	程序设计举例	72
习题 3		75

第 4 章 数组 77

4.1	一维数组	77
4.1.1	一维数组的定义	77
4.1.2	一维数组元素的引用	78
4.1.3	一维数组的初始化	78
4.1.4	一维数组的输入输出	79
4.2	二维数组	79
4.2.1	二维数组的定义	79
4.2.2	二维数组元素的引用	80
4.2.3	二维数组的初始化	81

4.2.4	二维数组的输入输出	81
4.3	字符数组和字符串	82
4.3.1	字符数组的定义	82
4.3.2	字符数组的初始化	82
4.3.3	字符数组的输入输出	84
4.3.4	常用的字符串处理函数	86
4.4	应用举例	88
习题 4		93
第 5 章	函数	97
5.1	标准函数	97
5.2	函数的定义	97
5.3	函数的调用	98
5.4	函数的原型	100
5.5	函数参数	101
5.5.1	参数的传递方式	101
5.5.2	默认参数	102
5.6	递归函数	103
5.6.1	递归函数的定义	103
5.6.2	递归调用的执行过程	104
5.7	变量的作用域和存储类	106
5.7.1	变量的作用域	106
5.7.2	变量的存储类型	108
5.8	编译预处理	113
5.8.1	宏定义	114
5.8.2	文件包含	114
5.8.3	条件编译	115
5.9	应用举例	115
习题 5		118
第 6 章	指针	121
6.1	指针的概念	121
6.1.1	地址与指针	121
6.1.2	指针定义	122
6.2	对指针变量的操作	123
6.2.1	指针的运算	123
6.2.2	new 和 delete	126
6.3	指针与数组	127

6.3.1	用指针访问一维数组.....	127
6.3.2	用指针访问二维数组.....	129
6.3.3	用指针访问字符串.....	130
6.3.4	指针数组.....	132
6.4	指针与函数.....	134
6.4.1	指针作为函数的参数.....	134
6.4.2	数组名作参数.....	138
6.4.3	指针函数.....	139
6.5	引用.....	140
6.6	应用举例.....	142
习题 6		144
第 7 章	类与对象.....	150
7.1	面向对象程序设计概念.....	150
7.2	类.....	152
7.2.1	类的声明.....	152
7.2.2	类成员的定义.....	154
7.3	对象.....	155
7.3.1	对象的定义.....	155
7.3.2	对象成员的引用.....	157
7.4	构造函数和析构函数.....	158
7.4.1	构造函数.....	159
7.4.2	析构函数.....	161
7.5	内联函数.....	162
7.6	静态成员.....	163
7.6.1	静态数据成员.....	163
7.6.2	静态成员函数.....	165
7.7	对象数组和对象指针.....	167
7.7.1	对象数组.....	167
7.7.2	对象指针.....	169
7.7.3	this 指针.....	170
7.8	友元.....	171
7.8.1	友元函数.....	172
7.8.2	友元成员函数.....	173
7.8.3	友元类.....	174
习题 7		175



第 8 章 继承与多态性	180
8.1 继承	180
8.1.1 单继承	180
8.1.2 多继承	183
8.2 派生类的构造函数和析构函数	187
8.3 重载	192
8.3.1 函数重载	192
8.3.2 运算符重载	195
8.4 多态性	200
8.4.1 虚函数	200
8.4.2 纯虚函数和抽象类	204
习题 8	208
第 9 章 对话框	209
9.1 MFC 应用程序	209
9.1.1 MFC 编程	209
9.1.2 MFC 应用程序框架类型	211
9.2 创建和使用对话框	215
9.2.1 创建对话框	215
9.2.2 控件的添加和布局	217
9.2.3 创建对话框类	219
9.2.4 调用对话框	220
9.3 通用对话框和消息对话框	223
9.3.1 通用对话框	223
9.3.2 消息对话框	226
习题 9	228
第 10 章 常用控件	229
10.1 控件的使用	229
10.1.1 控件的创建	230
10.1.2 控件的消息和消息映射	231
10.1.3 控件的数据交换和数据校验	232
10.2 静态控件和编辑框	233
10.2.1 静态控件	233
10.2.2 编辑框	233
10.2.3 应用举例	235
10.3 按钮控件	237

10.3.1	按钮的创建和消息	237
10.3.2	按钮的操作	238
10.3.3	应用举例	238
10.4	列表框	240
10.4.1	列表框的创建	240
10.4.2	列表框的通知消息	241
10.4.3	列表框的操作	241
10.4.4	应用举例	243
10.5	组合框	245
10.5.1	组合框的类型	245
10.5.2	组合框的数据输入	246
10.5.3	组合框的操作	246
10.5.4	组合框的消息	248
10.5.5	应用举例	248
10.6	滚动条	250
10.6.1	滚动条的结构	250
10.6.2	滚动条的消息和基本操作	251
10.6.3	应用举例	252
10.7	微调按钮	254
10.7.1	微调按钮的创建	254
10.7.2	微调按钮的操作	255
10.7.3	应用举例	256
10.8	进展条	258
10.8.1	进度条的操作	258
10.8.2	应用举例	259
10.9	列表视图	260
10.9.1	列表视图的建立	261
10.9.2	列表视图的操作	262
10.9.3	列表视图的数据结构	263
10.9.4	应用举例	264
习题 10		269

第1章

C++ 概 论

C++ 是一种应用广泛的面向对象程序设计语言, C++ 是由 C 语言发展而来的, 它保留了 C 语言的所有优点, 既可以用于面向过程的结构化程序设计, 又可以实现面向对象程序设计。本章主要讲述 C++ 的特点、语法、C++ 程序的实现过程和 Visual C++ 集成开发环境。

1.1 C++ 特 点

在具体讲述 C++ 的特点之前, 先来了解一下什么是程序和程序设计语言。

1.1.1 程序和程序设计语言

自 1946 年世界上第一台电子计算机问世以来, 计算机科学及其应用得到了迅猛的发展, 计算机已被广泛地应用于人类生产、生活的各个领域, 成为人们日常工作、生活、娱乐的一种现代化工具, 计算机已将人类带入了一个崭新的信息技术时代。对于当代大学生而言, 掌握一门计算机程序设计语言是十分必要的。那么什么是计算机呢, 这对大多数人来说, 对计算机可能并不陌生, 但对不熟悉计算机的人就会感到它很神秘。其实, 计算机只不过是一种具有内部存储能力、在程序的控制下自动工作的电子设备。一台计算机由硬件系统和软件系统两大部分组成, 硬件是物质基础, 而软件可以说是计算机的灵魂, 没有安装软件的计算机只能是一台“裸机”, 什么也干不了, 有了软件才能成为一台真正的“电脑”。而所有软件都是用计算机语言编写的。人们将需要计算机做的工作写成一定形式的计算机能够识别的指令, 并把它存储在计算机内部的存储器中, 当人们给出指令后, 它就会按照指令操作顺序自动工作。人们把这种可以连续执行的一条条指令的集合称为程序, 而程序是用程序设计语言来编写的。程序设计语言按照语言级别可以分为低级语言和高级语言。低级语言有机器语言和汇编语言。高级语言则主要有过程式语言(如 C、Basic 以及 Pascal 等)、面向对象语言(如 C++、Java 等)、应用式语言(如 Lisp)以及基于规则的语言(如 Prolog)等。

1. 机器语言

机器语言是计算机硬件可以直接识别的语言,它完全用 0 和 1 组成的代码表示,也是最低层的程序设计语言。用机器语言编写的程序中,每一条机器指令都是二进制形式的指令代码。机器语言是面向机器的,不同的计算机硬件(主要是 CPU)其机器语言是不同的,因此,针对一种计算机所编写的机器语言程序不能在另一种计算机上运行。有了机器语言,人们就可以用机器语言编写程序,然后输入计算机,计算机就可以通过运行程序来体现人们的意图,计算或处理相应的问题。

2. 汇编语言

由于机器语言是面向具体机器的,所以其程序缺乏通用性,编写程序的过程繁琐复杂,易出错,错了又不易查找和修改,编出的程序可读性差。繁杂的机器代码程序很难记忆,未受过专门训练的人又不易掌握,这严重阻碍了计算机的应用和发展。于是,人们又在机器语言的基础上研制了汇编语言。汇编语言采用符号(称为指令助记符)表示指令,比机器语言的指令代码易于记忆。

用汇编语言编写的程序(又称源程序)经汇编器加工处理后,就可转换成在计算机上可以直接执行的机器语言程序。汇编语言实质上是机器语言的符号化形式,仍属面向机器的一种低级语言。

3. 高级语言

由于汇编语言也依赖于计算机的硬件体系,且助记符量大难以记忆,于是人们又发明了更加易用的所谓高级语言。这种语言其语法和结构更类似普通英文,且通用性好,不必对计算机的指令系统有深入的了解就可以编写程序。

采用高级语言编写的程序在不同型号的计算机上只需做某些微小的改动便可运行,对这些高级语言程序只要采用对应计算机上的编译程序重新编译,即把由高级语言编写的源程序转换成机器能够接受的机器语言程序,这些程序就可以在不同的机器上运行。这种具有翻译功能的程序称为编译程序,每一种高级语言都有与之对应的编译程序。

1.1.2 C++ 的特点

C++ 于 20 世纪 80 年代由美国贝尔实验室设计并实现,它是在 C 语言的基础上发展起来的,既支持传统的面向过程的程序设计,又支持面向对象的程序设计。

C 语言是在 B 语言的基础上由美国贝尔试验室的 Dennis M. Ritchie 于 1972 年设计实现的,改进了 B 语言的缺陷,其设计目标是既保持 BCPL 和 B 的精练性及接近硬件的特点,又增加这些语言的通用性。由于 C 语言简洁,功能强大,运行速度快,一直是非常重要的程序设计语言之一。较之其他高级的语言,C 语言的主要特点如下:

- (1) 功能强,应用广泛;
- (2) 语句简洁,表达能力强;

- (3) 运算符丰富;
- (4) 数据结构丰富,具有现代化语言的各种数据结构;
- (5) 具有结构化的控制语句;
- (6) 程序设计自由度大;
- (7) C 语言允许直接访问物理地址,能够进行位操作,能够实现汇编语言的大部分功能,可以直接对硬件进行操作,既有高级语言的功能,又有低级语言的功能;
- (8) 生成目标代码质量高,程序执行效率高;
- (9) 可移植性好。

除了上述优点之外,C 语言也有它的局限性:

- (1) C 语言的类型检查机制相对较弱,有些错误不能在编译阶段检查出来;
- (2) C 语言本身几乎没有支持代码重用的语言结构;
- (3) 当程序的规模达到一定的程度时,程序员就很难控制程序的复杂性。

与 C 语言不同,C++ 是一种广泛使用的面向对象的程序设计语言,C++ 包括了 C 语言的所有特征、属性和优点(如高效、灵活性),同时改进了 C 语言的一些不足,并且支持面向对象的程序设计。C++ 的特点如下:

- (1) 保持与 C 语言兼容;
- (2) 可读性更好,代码结构更合理;
- (3) 生成代码的质量高;
- (4) 可重用性、可扩充性、可维护性和可靠性有所提高;
- (5) 支持面向对象的机制。

C++ 中与面向对象有关的特征如下:

(1) 类和数据封装: C++ 支持数据封装,将数据和对该数据的操作的函数封装在一起作为一种数据类型,称为类。同时提供一种对数据访问严格控制的机制,封装体通过操作接口与外界交换信息。

(2) 结构体: 在 C 语言中可以定义结构体,但是这种结构只包含数据,而不包含函数。C++ 中的类是数据和函数的封装体,在 C++ 中,结构体可以作为一种特殊的类。

(3) 构造函数和析构函数: 构造函数是类内和类同名的成员函数,创建对象时对类的数据成员进行初始化。析构函数的功能是用来释放一个对象。

(4) 私有、保护和公有成员: C++ 类中可以定义三种不同访问控制权限的数据成员。它们分别是私有(private)、保护(protected)和公有(public)成员。私有成员是只有类本身定义的函数才能访问,而类外的其他函数不可以访问。保护成员是只有派生类可以访问,而在类外不可以访问的成员。公有成员是在类外也可以访问的成员,是该类与外界接口。

(5) 对象和消息: 对象是类的实例,对象之间通过消息来实现合作,共同完成某一任务。每个对象根据收到消息的性质来决定需要采取的行动,以响应这个消息。

(6) 友元类和友元函数: 类中的私有成员是不允许类外的任何函数访问的。但是友

元打破了类的这一限制,破坏了类的封装性,它可以访问类的私有成员。友元可以是类外定义的整个类,称为友元类,也可以是类外的函数,称为友元函数。

(7) 运算符和函数名重载:函数名重载和运算符重载都属于多态,多态是指相同的语言结构可以代表不同类型的实体,或者对不同类型实体进行操作。C++ 允许相同的运算符或标识符代表多个不同实现的函数,这就称标识符或运算符重载,用户可以根据需要定义标识符重载或运算符重载。

(8) 派生类:一个类可以根据需要生成派生类,派生类继承了基类的所有方法,同时还可以定义新的不包含在父类中的方法。派生类包含从父类继承过来的数据成员和自己特有的数据成员。

(9) 虚拟函数:C++ 可以定义虚函数,通过虚函数实现动态联编。动态联编是多态的一个重要特征。多态性的形成是由父类和它们的子类组成的一个树形结构。在这个树中的每一个子类可接收一个或多个具有相同名字的消息。当一个消息被这个树中一个类的一个对象接收时,那么这个对象动态地决定给予子类对象消息。多态中的这一特性允许使用高级抽象。

1.2 C++ 程序的实现

一般把由高级语言编写的程序称为源程序,这种程序不能在机器上直接运行,只有把它转换为由二进制代码表示的目标程序后,才能在机器上运行。与其他高级语言一样,C++ 程序的实现要经过编辑、编译、连接和运行几个步骤,其过程如图 1.1 所示。

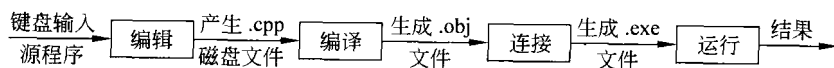


图 1.1 C++ 程序的实现过程

1. 编辑

编辑是将编写好的 C++ 源程序输入到计算机中,并生成磁盘文件的过程。C++ 程序的编辑可以利用计算机软件所提供的某种编辑器进行编辑,然后将 C++ 程序的源代码存放到磁盘文件中,磁盘文件的扩展名是 .cpp。

在实际应用中,所选用的 C++ 编译器本身都提供编辑器的功能,使用所选用的 C++ 编译器中的编辑器来编写 C++ 源程序是十分方便的。例如,Microsoft Visual C++ 6.0 版本就提供编辑功能,将 C++ 程序输入后,指定文件名便可存入磁盘文件。然后,选用编译菜单项,便可编译执行。其他的 C++ 编译器也都有编辑功能,可用它来进行源程序编辑,不必再去选用其他编辑软件。编辑器所采用的编辑方法都大致相同,采用全屏幕编辑方法。插入、覆盖、删除等简单操作都与 Word 相同或相近,也有块操作功能。例如,删除块、复制块、移动块都可以通过编辑菜单中的菜单项进行。