



C语言贯通教程

—— C、C++、Visual C++及Windows应用程序

史斌星 史佳 编著

四合一的教程

改革传统的C语言教学方式

从C编程轻松进阶到C++编程

丰富的实例造就无所畏惧的学习

成为高手也不难



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



C 语言贯通教程

——C, C ++, Visual C ++ 及 Windows 应用程序

史斌星 史 佳 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书是将传统的 C、C++、Visual C++ 和 Windows 应用程序四部分内容结合起来编成的统一教程,作为改革 C 语言教学方法的一种尝试,以提高学习这方面内容的效率。本书结构严谨、语言生动,深入浅出,既注重基本概念,又能从实际出发。全书含有 90 多个程序实例,使本来难懂的理论变得较为容易理解,形成了独特的叙述风格。本书适用于没有 C 语言基础或对 C 有一定了解,希望进一步学习 C++、Visual C++ 和 Windows 应用程序的读者。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名: C 语言贯通教程——C、C++、Visual C++ 及 Windows 应用程序

作 者: 史斌星 史 佳

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编: 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑: 章忆文

印 刷 者: 清华大学印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787 × 1092 1/16 印张: 30 字数: 727 千字

版 次: 2001 年 8 月第 1 版 2001 年 8 月第 1 次印刷

书 号: ISBN 7-900635-85-8

印 数: 0001 ~ 6000

定 价: 46.00 元

序

改进 C 语言教学的一种尝试

《C 语言贯通教程》只是本书的 C、C++ 一种简便称呼,按其内容来说,应该称为 C、C++、VC++ 和 Windows 应用程序贯通教程。将这四部分内容放在同一本书中是一种尝试,因为其中的每一部分都可以构成一本大部头的著作,现在要将它们组织成一本中等篇幅的书,确实有一定的难度。但是我们相信,只要将这些内容有机地结合起来,这个困难是可以克服的。这种一揽子做法的诸多好处激励着我们一定要把这项工作做好。

C 是目前应用最为广泛的电脑语言,但也以其难学而闻名遐尔。据笔者看来,并不是 C 天生难学,而是教学方法上的某些不足造成了这一困难。希望通过我们的努力能够摘去“C 语言难学”这顶帽子。我们将在以下几个方面进行改进:

1. 将 C 和 C++ 结合起来。C++ 是在 C 的基础上发展起来的,和 C 有着密切的联系,但目前往往将它们分别教授,先 C 后 C++,教科书也是分别编写的。分开进行教学的方法既不利于提高教学效率又不利于将问题说清楚。C++ 对 C 的继承并不亚于它的发展,因此它们间的共性也不少于它们的差异。不但 C++ 需要借助 C 的概念加以说明,反过来 C 的概念在 C++ 中的深入展开,也会加深对 C 的理解。如果将 C 和 C++ 的教学紧密结合,这种相辅相成的长处最易得到发挥。C 和 C++ 分离的教学方法也是不少人只熟悉 C 而不熟悉 C++ 的原因。

2. 将 C++ 和应用程序的编写结合起来。这两部分内容的教学也往往是分离的。C++ 的特点在于对类的发展和广泛应用。类的理论是集抽象性、多样性、灵活性和生动性的大成,这些诸多的属性在 Windows 应用程序中得到了充分的发挥,也只有应用程序中 C++ 的这些特征才能展示出来。C++ 的教学脱离了应用程序就会感到抽象、难懂。难怪有的学生说,C 好懂、C++ 难真懂。不是 C++ 真的难懂,而是缺乏使它具体化的实例,Windows 应用程序中有着数不清的 C++ 应用实例,不去充分利用这些上好的教材实在是一种损失。

3. 将 C 语言和它的编译软件结合起来。C 语言程序是必须经过编译才能运行的程序。如果只讲程序本身不讲程序的编译就无法检验所编程序的正确性。如果前面的错误概念不能得到及时纠正,后面的学习就会受到影响。通过检验程序发现和改正程序中的错误本身就是一种有效的学习手段。我们将努力克服 C 语言教学中只重语言不重编译的现象。

4. 刷新教学内容。就 PC 机来说现在已经进入了 Windows 2000 的时代,已经很少采用 DOS 操作系统了。C 语言的编译软件也有了极大的发展,即使一些在 Windows 下被广泛采用的编译软件,也有不少被淘汰了。例如曾经红极一时的 Borland C++ 已被可视化的 VC++ (Visual C++) 编译软件所代替。有的教科书却至今仍在讲解 DOS 下运行的 C 语言编译软件就显得十分落后了。本书从头到尾都以 VC++ 6.0 (可视化 C 语言编译软件 6.0 版本) 作为书中所有程序的编译软件。

总而言之,笔者希望这本书**更加现代、更加实用和更加好懂**。

全书共有 10 章,由 5 个部分组成。第 1 部分由第 1 章构成,介绍 VC++ 6.0 编辑软件;第 2 部分由第 2 章构成,介绍 C;第 3 部分由第 3 章构成,介绍 C++;第 4 部分由第 4 章构成,介绍 Win32 Application 程序;第 5 部分由第 5 章至第 10 章构成,介绍 MFC 为基础的 Windows 应用程序。为了帮助学习,本书附有一张光盘,其中包含书中列举的全部程序,并给出了编译得到的可执行程序,这些可执行程序可以在 Windows 或 DOS 方式下直接运行。光盘的使用方法另见说明。

本书得到了余加莉、沈宏、方新、庄希、高红、刘英等老师的热情帮助,在此表示衷心的感谢。由于学识浅薄,书中难免出现差错,恳望读者及专家学者批评指正。

史斌星 史佳

2001 年 4 月

怎样使用本书附带的光盘

一、光盘的章号和书中章号完全对应。

二、光盘中的项目名称和书中采用的一致。

三、由于光盘中的文件是“只读”属性,VC++ 6.0 的 Class Wizard 不能对“只读”文件进行操作管理,所以在使用VC++ 6.0 之前,应先将它们全部复制到硬盘中,并将所有文件改为“存档”属性。修改大批文件属性的方法如下:

1. 打开 Windows 资源管理器。
2. 选取某个项目。
3. 选取该项目中的所有文件。
4. 右击被选取的文件,弹出快捷菜单。
5. 单击快捷菜单中的“属性”命令,弹出“属性”对话框。
6. 在“属性”对话框中将“只读”改为“存档”,便可一次性地改变该项目中所有文件的属性。

四、如果想观看或编辑源程序,按以下步骤操作:

1. 进入VC++ 6.0。
2. 选取 File 菜单。
3. 选取 Open Workspace 命令,弹出 Open Workspace 对话框。
4. 在 Open Workspace 对话框中选取所要的章和项目。
5. 双击扩展名为 dsw 的文件。

五、也可以不打开VC++ 6.0,直接运行项目中的可执行文件,方法如下:

1. 第1、2、3章中的程序在MS-DOS环境中运行,方法如下:
 - (1) 单击Windows桌面的“开始”按钮,打开“程序”菜单。
 - (2) 单击“程序”菜单中的“MS-DOS方式”命令。
 - (3) 键入要运行选取的文件地址和文件名,按回车键,便开始运行。
 - (4) 对于执行程序 prog2_24a.exe 和 prog2_25a.exe,除了文件名外还需要键入一个或两个命令行参数。
 - (5) 由于执行程序 prog2_25.exe、prog2_25a.exe、prog2_26.exe 和 prog2_27.exe 时需要往某个文件中写入内容,所以必须将项目复制到硬盘中,并将该文件的“只读”属性修改为“存档”属性,才能运行,它们不能直接在光盘中运行。
2. 第4至9章中的程序在Windows环境中运行,方法如下:
 - (1) 单击Windows“开始”按钮,弹出“开始”菜单。
 - (2) 单击“开始”菜单中的“运行”命令,弹出“运行”对话框。

- (3) 通过“运行”对话框中的“浏览”按钮选取要运行的可执行文件。
- (4) 单击“运行”对话框的“确定”按钮便可运行。
- 3. 运行第 10 章的 tongxunlu.exe 文件有一个附加要求：
 - (1) 按照 10.2 节所述的方法配置数据源。
 - (2) 按照第 4 至 9 章中程序的运行步骤操作。

目 录

第1章 VC++ 6.0 介绍	1
1.1 C 语言介绍	1
1.1.1 计算机语言	1
1.1.2 对计算机语言的基本要求	1
1.1.3 一个最简单的 C 语言程序	2
1.1.4 编译软件	2
1.1.5 C 语言的三个阶段	3
1.1.6 可视化开发软件	4
1.1.7 VC++ 6.0	4
1.2 进入VC++6.0	5
1.2.1 编译步骤	5
1.2.2 打开VC++6.0	6
1.3 VC++ 6.0 的界面	7
1.4 创建一个 C 程序	8
1.4.1 创建控制台程序	8
1.4.2 通过工具栏创建新项目	9
1.4.3 通过工具栏创建新文件	10
1.4.4 工具栏按钮和菜单栏命令的比较	11
1.5 怎样编译和运行程序	11
1.5.1 程序的编译	11
1.5.2 输出窗中的消息	12
1.5.3 程序的运行	13
1.5.4 控制台程序的几种运行方法	14
1.5.5 Workspace 中的内容	15
1.5.6 什么是 Project	15
1.6 怎样修改老程序	16
1.6.1 怎样打开一个老程序	16
1.6.2 程序的修改	16
1.6.3 退出和存盘	17
1.6.4 Debug 和 Release 版本	18
1.7 Workspace	19
1.7.1 展开和收缩符号	19
1.7.2 隐藏 Workspace 窗口	20

1.7.3	隐藏输出窗	20
1.7.4	改变 Workspace 和输出窗的大小	20
1.7.5	FileView 和 ClassView	21
1.8	怎样检查错误	21
1.8.1	输出窗中的错误信息	21
1.8.2	制造一个错误	22
1.9	怎样使用 VC++ 6.0 帮助	23
1.9.1	怎样进入帮助	23
1.9.2	目录选项	24
1.9.3	索引	27
1.9.4	搜索	28
1.9.5	书签	28
1.9.6	F1 帮助键	29
1.10	怎样安装 VC++ 6.0	29
1.10.1	起动安装盘	29
1.10.2	签署协议	31
1.10.3	ID 数	32
1.10.4	重新启动系统	32
1.10.5	选择安装方式	33
1.10.6	选择安装内容	35
1.10.7	安装进度指示	37
1.10.8	安装帮助软件	37
1.10.9	选择 MSDN 安装类型	38
1.10.10	注册	40
1.11	本章小结	40
第 2 章	C 语言基础	42
2.1	C 语言程序的基本特点	42
2.1.1	C 语言程序的结构	42
2.1.2	C 语言程序的书写格式	43
2.2	输出输入语句	44
2.2.1	格式化的输出语句	44
2.2.2	格式化的输入语句	45
2.3	在 VC++ 6.0 中创建新项目	47
2.3.1	从源文件创建新项目	47
2.3.2	创建 Console 项目	49
2.4	数据	52
2.4.1	变量	52
2.4.2	常量	56

2.4.3	转换说明符	59
2.4.4	变量和常量的大小的测试	60
2.4.5	控制代码的测试	62
2.4.6	标识符和关键字	64
2.4.7	变量的存储属性	64
2.4.8	数组	66
2.5	运算语句.....	70
2.5.1	赋值运算	70
2.5.2	算术运算	71
2.5.3	前置和后置运算	73
2.5.4	关系运算	73
2.5.5	条件运算	73
2.5.6	逗号结合运算	75
2.5.7	逻辑运算	75
2.5.8	位操作	76
2.5.9	数据类型转换	78
2.5.10	运算顺序和结合性.....	79
2.6	控制语句.....	81
2.6.1	if 结构	81
2.6.2	swich 结构	82
2.6.3	while 结构	84
2.6.4	do...while 结构	86
2.6.5	for 结构	90
2.6.6	转移语句	92
2.7	指针.....	93
2.7.1	指针变量	94
2.7.2	指针和数组	99
2.7.3	字符指针	100
2.7.4	指针数组	101
2.7.5	多级指针	103
2.8	函数	104
2.8.1	一个最简单的函数.....	104
2.8.2	用地址量作实参数.....	107
2.8.3	利用地址量返回结果.....	108
2.8.4	参数个数不定的函数.....	109
2.8.5	利用全局变量传递数据.....	109
2.8.6	函数应用举例.....	110
2.8.7	如何将数组传递到函数中.....	115
2.8.8	如何将字符串传递到函数中.....	116

2.8.9	如何将多维数组传递到函数中	117
2.8.10	指针型函数	119
2.8.11	递归函数	121
2.8.12	函数指针	123
2.8.13	库函数	125
2.8.14	输入和输出函数的格式	127
2.9	文件	128
2.9.1	文件的打开	128
2.9.2	命令行参数	130
2.9.3	文件的复制	132
2.9.4	字符串输入和输出函数	135
2.9.5	文件的格式化输入和输出	136
2.10	结构和联合	137
2.10.1	结构	137
2.10.2	联合	146
2.10.3	枚举	148
2.10.4	类型定义	150
2.10.5	条件编译	151
2.11	动态存储分配	153
2.12	本章小结	156
第3章	C++	157
3.1	C++ 和 C 的主要差别	157
3.2	C++ 的类	170
3.2.1	什么是类	170
3.2.2	类的说明	172
3.2.3	类的定义	174
3.2.4	类的应用	175
3.2.5	类的对象	177
3.2.6	this 指针	177
3.2.7	内联成员函数	179
3.3	C++ 项目的创建	180
3.4	对象的初始化	185
3.4.1	构造函数	185
3.4.2	析构函数	188
3.4.3	默认构造函数和析构函数	191
3.4.4	建立动态对象	191
3.5	类的继承	192
3.5.1	单一继承	192

3.5.2	多重继承	196
3.5.3	友元函数	198
3.6	多态性和虚函数	201
3.6.1	什么是多态性	201
3.6.2	虚函数	202
3.7	面向对象的程序设计和本章小结	203
第4章	Win32 Application 程序	205
4.1	一个最简单的 Win32 Application 程序	205
4.2	文件 prog4_1.cpp 的说明	217
4.2.1	prog4_1.cpp 中的说明语句	218
4.2.2	LoadString() 函数	219
4.2.3	窗口类的定义和注册	220
4.2.4	CreateWindow() 函数	222
4.2.5	其他函数	224
4.2.6	事件驱动的程序	225
4.2.7	WndProc() 窗口函数	226
4.2.8	显示 About 对话框	227
4.2.9	终止程序的运行	228
4.2.10	字符的显示	228
4.3	字符显示技术	230
4.3.1	prog4_2.cpp 源文件	230
4.3.2	源程序分析	233
4.3.3	程序的运行	238
4.3.4	对消息的识别	239
4.4	图形的显示	240
4.5	图形的填充	246
4.6	本章小结	250
第5章	菜单程序	251
5.1	一个空白的菜单程序	251
5.1.1	怎样创建新项目	251
5.1.2	编译和运行程序	259
5.1.3	Windows 窗口的基本功能	260
5.2	blank 程序剖析	261
5.2.1	头文件 blank.h	261
5.2.2	源文件 blank.cpp	265
5.2.3	头文件 MainFrm.h	269
5.2.4	源文件 MainFrm.cpp	271

5.2.5	头文件 ChildView.h	275
5.2.6	源文件 ChildView.cpp	276
5.2.7	stdafx 文件	277
5.3	程序的运行顺序	278
5.3.1	程序的入口	278
5.3.2	调试方法	285
5.4	菜单编辑器	289
5.5	ClassWizard	296
5.5.1	进入 ClassWizard	296
5.5.2	改动后的程序	300
5.6	工具栏编辑器	310
5.6.1	工具栏编辑器的界面	310
5.6.2	怎样创建新按钮	311
5.6.3	怎样在按钮上作图	314
5.7	加速键编辑器	316
5.8	字符串编辑器	319
5.9	快捷菜单	321
5.10	本章小结	329
第 6 章	对话框程序	330
6.1	AppWizard 创建的对话框	330
6.2	对话框编辑器	333
6.2.1	控件工具栏	334
6.2.2	对话框工具栏	336
6.3	两个例子	339
6.3.1	征求意见表	339
6.3.2	日历查阅程序	343
6.4	对话框程序的剖析	344
6.4.1	资源文件	345
6.4.2	stdafx.cpp 源文件	345
6.4.3	blankDlg.cpp 文件	346
6.4.4	blankDlgDlg.cpp 文件	347
6.4.5	添加系统菜单项	351
6.5	组合框控件	354
6.6	对话框之间的通话	359
6.7	非模态对话框	370
6.8	本章小结	377

第 7 章 绘图程序	379
7.1 菜单控制的绘图程序	379
7.1.1 添加菜单项	379
7.1.2 添加消息处理函数	380
7.1.3 源程序代码	382
7.2 用鼠标绘图	391
7.2.1 创建项目	391
7.2.2 源程序代码	392
7.2.3 运行程序	398
7.3 本章小结	400
第 8 章 文字输入程序	401
8.1 一个简单的输入程序	401
8.1.1 创建新项目	401
8.1.2 添加代码	402
8.2 文本文件的读取	408
8.2.1 创建新项目	408
8.2.2 修改源程序	409
8.3 以 CEditView 为基类的程序	413
8.4 以 CRichEditView 为基类的程序	417
8.5 本章小结	420
第 9 章 多媒体编程	421
9.1 bmp 文件的显示	421
9.1.1 创建项目	421
9.1.2 程序分析	427
9.2 图标和位图的显示	429
9.2.1 引入或创建图像资源	430
9.2.2 添加控件	431
9.2.3 程序代码	433
9.3 声音程序	435
9.4 时钟程序	438
9.5 动画程序	441
9.6 本章小结	443
第 10 章 数据库编程	444
10.1 什么是数据库	444
10.2 配置数据源	446
10.3 创建一个支持数据库的项目	449

10.4	显示窗口的设计.....	452
10.4.1	对话框中的控件.....	452
10.4.2	添加成员变量.....	453
10.4.3	程序的运行.....	454
10.5	对程序的分析.....	455
10.6	本章小结.....	464

第1章 VC++ 6.0 介绍

1.1 C 语言介绍

1.1.1 计算机语言

计算机就是通常所说的电脑,更准确的是称为数字计算机,表明它是与数字打交道的。什么是计算机语言呢?计算机语言是操作人员与电脑进行沟通的工具。计算机虽然“聪明”,但是归根到底还要听人的指挥,人就是通过计算机语言对它进行指挥的。人与人的沟通也要通过语言,不过与人之间的交谈不同,人与计算机的“交谈”只通过数字。例如,用“10”表示“加”法,用“01”表示“减”法,用“11”表示“乘”法等。但这种数字语言对于人来说却不大方便,不但枯燥无味,还要求操作人员记忆大量的数据,实在是一件倒胃口的事,无疑将影响电脑的推广和应用。对于人来说,文字或字符比数字更容易理解和记忆,例如以“+”表示“加”法,以“-”表示“减”法,以“print”表示“打印输出”等,这就比10、01等数字容易理解和好记多了。但是对于电脑来说却不能理解+、-、print等字符和文字意义,怎样才能两全其美,使人和电脑都满意呢?人们终于找到了一个解决办法,那就是人们用文字或字符对电脑发布指令,然后通过编译器将这些指令翻译为数字。例如将“+”翻译为“10”,将“-”翻译为“01”,将print也翻译为某种规定的数码,这样一来电脑就可以懂了,人们的编码工作也就方便了。也就是人说人话,电脑说电脑话,编译器任翻译官。这种用普通英文字符写成的电脑文件称为计算机程序,编程规则就是计算机语言。在交付电脑执行前,将所编程序翻译为电脑能懂的数字文件,称为对程序进行编译。经编译产生的数字文件称为可执行文件。计算机语言由一系列基本语句构成,每一条语句都是对电脑的一条操作指令。人们创建了各种不同的计算机语言,C语言是其中的一种,也是目前应用最为广泛、最为受人关注的一种。

1.1.2 对计算机语言的基本要求

对于一种好的计算机语言来说,至少应满足以下要求:

- (1) 表达力强、功能完善,能最大限度地利用电脑的功能。
- (2) 语句严谨、结构简洁、可读性好、便于记忆。
- (3) 所编程序的移植性好。

C语言可以满足以上所有要求。关于一些更深入的属性例如继承性、派生性等将在以后说明。

1.1.3 一个最简单的 C 语言程序

正式对 C 语言的介绍要在下一章中进行,本章的任务是介绍 VC++ 6.0 程序编译软件,这是当前流行的 C 语言编译器,在以后各章中要频繁地使用。但为了说明这个软件,先要在这里介绍一个用 C 语言编写程序的例子。通过这个例子,可以更好地对 VC++ 6.0 进行介绍。下面给出的 Prog1_1.c 是一个最简单的 C 语言程序,它的惟一使命是在电脑屏幕上显示英文文字“Hello!”。

Prog1_1.c

```
#include "stdio.h"
void main()
{
    printf("Hello!\n");
}
```

如果希望输出中文“你好!”,只需将程序改成:

Prog1_2.c

```
#include "stdio.h"
void main()
{
    printf("你好!\n");
}
```

即可。由于要输出中文,需要在中文 Windows 或其他中文系统中运行。推而广之,只要将双引号中的文字改成其他文字,无论英文还是中文,运行该程序后,便会显示相应的文字。即使是这么一个简单的程序,也是有一定的实用价值。

以上程序虽然简单,也有许多需要交待的地方,但由于本章的主要任务是介绍 VC++ 6.0,最好不要节外生枝,因此全面的解说程序将在下一章中进行,现在仅就其中的一个语句予以说明:

```
printf("你好!\n");
```

这是一个完整 C 语句,它有几层含义:

- (1) printf 是一个 C 命令,意思是要将双引号中的文字“你好!”在荧光屏上显示出来。
- (2) 符号\n 也是一个 C 命令,表示要进行换行,即显示“你好!”字样后光标移到下一行。如果没有这个符号,则光标不换行,停留在“你好!”的右侧。
- (3) 分号“;”是语句结束符号,表示该语句的结束。
- (4) 该语句中双引号和圆括号都是必要的,没有它们在编译时将出现语法错误(syntax error)信息,不能通过编译。所以 C 语句有一定的句法,不能随意更改。

1.1.4 编译软件

程序 prog1_1.c 和 Prog1_2.c 虽已写好,但还不能立刻运行。这些程序是由英文字母及