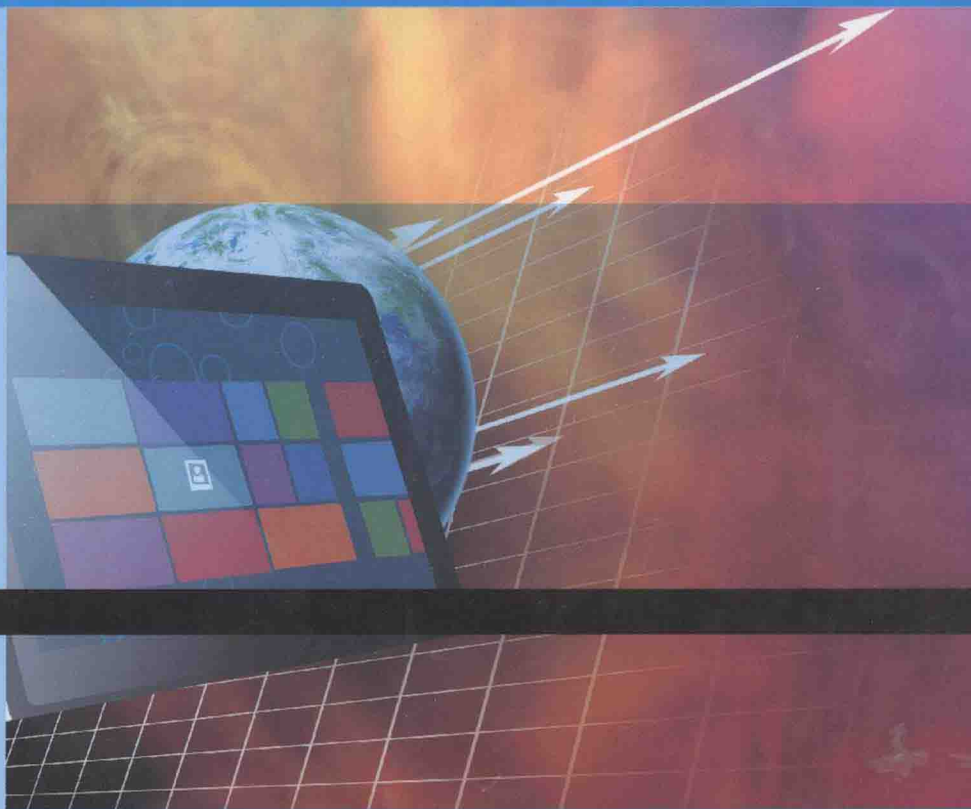


● 21世纪高等院校计算机应用规划教材

C++语言程序设计上机实验 及学习指导

主 编 王珊珊 臧 浏 张志航



21世纪高等院校计算机应用规划教材

C++语言程序设计上机实验 及学习指导

主 编 王珊珊 臧 浏



内容简介

本书是作者编写的另一本理论教材《C++ 程序设计教程第 2 版》(机械工业出版社)的配套教材。即可用作大学各专业知识学习 C++ 语言的初学者的实践教材,又可以用作全国或江苏计算机等级考试二级 C++ 语言的学习和辅导教材。

本书内容丰富,主要包含四部分内容:第一部分为上机实验,包括与理论教材中各章配套的上机实验共十四个。第二部分为与理论教材对应的共十五章的精选知识点、例题及解析、练习题。第三部分为课程设计。第四部分为笔试样卷及答案。教材最后给出的附录 A 内容为要求掌握的基本算法,附录 B 为第二部分练习题参考答案。

本书适用于大学本科学理工类各专业知识学习 C++ 程序设计语言,同时也适用于自学 C++ 语言的读者。

图书在版编目(CIP)数据

C++ 语言程序设计上机实验及学习指导/王珊珊,臧洌,张志航主编. —南京:南京大学出版社,2016.1
21 世纪高等院校计算机应用规划教材
ISBN 978-7-305-15967-1

I. ①C… II. ①王… ②臧… ③张… III. ①C 语言 - 程序设计 - 高等学校 - 教学参考资料 IV. ①TP312

中国版本图书馆 CIP 数据核字(2015)第 240596 号

出版发行 南京大学出版社
社 址 南京市汉口路 22 号 邮 编 210093
出 版 人 金鑫荣

丛 书 名 21 世纪高等院校计算机应用规划教材
书 名 C++ 语言程序设计上机实验及学习指导
主 编 王珊珊 臧 洌 张志航
责任编辑 单 宁 吴宜锴 编辑热线 025-83596923

照 排 南京理工大学资产经营有限公司
印 刷 南京京新印刷厂
开 本 787×1092 1/16 印张 15.25 字数 371 千
版 次 2016 年 1 月第 1 版 2016 年 1 月第 1 次印刷
ISBN 978-7-305-15967-1
定 价 39.00 元

网 址: <http://www.njupco.com>
官方微博: <http://weibo.com/njupco>
官方微信号: njupress
销售咨询热线: (025)83594756

* 版权所有,侵权必究

* 凡购买南大版图书,如有印装质量问题,请与所购
图书销售部门联系调换

前 言

C++ 语言是目前被广泛使用的程序设计语言,它语法简洁,运行高效,既可以进行面向过程的程序设计,又可以进行面向对象的程序设计。各大专院校工科类专业普遍开设了程序设计语言类课程,尤其是电类专业普遍开设了 C++ 程序设计课程。

本书是作者编写的另一本理论教材《C++ 程序设计教程第 2 版》(机械工业出版社)的配套教材,是在作者总结过去二十几年的教学和编程实践经验的基础上编写而成的。C++ 语言主要包括两个方面的内容:(1) 传统面向过程的程序设计,目的是让初学者掌握基本的程序设计知识。(2) 面向对象的程序设计,让初学者学习面向对象对象程序设计的基本概念,为今后学习以面向对象为基础的通用软件开发工具如 Visual C++、Delphi 和 C#.net 等打下坚实的基础。

本书内容丰富,即可用作大学各专业知识 C++ 语言的初学者的实践教材,又可以用作全国或江苏计算机等级考试二级 C++ 语言的学习和辅导教材。本书目前被用作南京航空航天大学本科各专业的程序设计实践教材。

本书主要包含四部分内容:第一部分为上机实验,对 C 语言的集成开发环境 Visual C++ 6.0 做了介绍,同时给出了理论教材中各章的配套上机实验共十四个。第二部分为各章知识点、例题及解析、练习题,针对各章的理论教学难的点和重点,精选共十五章的知识点和难点,对每个知识点给出例题并做出详尽的解析,在各章的最后给出练习题供同学自行练习,巩固理论教学内容。第三部分为课程设计,给出课程设计的总体要求,并提供几个课程设计选题。第四部分为笔试样卷及答案。教材最后给出的附录 A 内容为要求掌握的基本算法,附录 B 为第二部分练习题参考答案。

本书在教材编写组充分酝酿和讨论的基础上编写而成,第一部分由王珊珊执笔;第二部分的第 1~4、14 和 15 章由张志航执笔、第 5~9 章由臧浏执笔、第 10~13 章由王珊珊执笔;第三部分由王珊珊、臧浏执笔。全书由王珊珊负责统稿。王珊珊仔细阅读了本书,在基本概念以及文字叙述上做了把关。参加本书编写工作的还有尤彬彬、朱敏、张定会、张卓莹、潘梅园。

本书全部内容的建议学时为:上机实验 60 小时、课程设计 16 小时、理论教学 48 学时(内容另行安排)。本书的实验环境是 Visual C++ 6.0。本书全部例题和习题均在该环境中已通过编译和运行。

本书必定会存在疏漏、不妥和错误之处,恳请专家和广大读者指教和商榷。几位主编作者的电子邮件为:shshwang@nuaa.edu.cn(王珊珊),臧浏(zangliwen@nuaa.edu.cn),zzh20100118@qq.com(张志航)。

《C++ 语言程序设计上机实验及学习指导》教材编写组

2016 年元月

目 录

第一部分 上机实验

一 上机环境介绍	3
一、Visual C++ 6.0 集成环境和程序开发过程简介	3
二、Visual C++ 6.0 程序调试	7
二 上机实验内容	11
实验一 Visual C++ 6.0 运行环境和运行多个 C++ 程序的方法	12
实验二 数据类型、运算符和表达式	14
实验三 简单的输入输出	17
实验四 流程控制语句	21
实验五 函数	23
实验六 编译预处理	26
实验七 数组与字符串	27
实验八 结构体、共用体和枚举类型	31
实验九 指针和引用	33
实验十 类和对象	37
实验十一 静态成员、友元函数和友元类	39
实验十二 继承和派生类	40
*实验十三 多态性	42
实验十四 输入/输出流	44

第二部分 各章知识点、例题及解析、练习题及答案

第1章 C++ 概述	51
第2章 数据类型、运算符和表达式	53
第3章 简单的输入/输出	59
第4章 C++ 的流程控制	61
第5章 函数	69
第6章 编译预处理	80
第7章 数组	83
第8章 结构体、共用体和枚举类型	94

第9章 指针、引用和链表	102
第10章 类和对象	123
第11章 类和对象的其他特性	139
第12章 继承和派生	148
*第13章 多态性	164
第14章 输入/输出流	189
*第15章 模板	193

第三部分 C++ 语言课程设计

课程设计总体要求	199
菜单设计练习	201
选题一:图书库存记录管理	203
选题二:学生成绩管理	210

第四部分 笔试样卷及答案

笔试样卷	217
笔试样卷答案	225
附录 A. 要求掌握的基本算法	228
附录 B. 第二部分习题参考答案	231

第一部分

上机实验

一 上机环境介绍

一、Visual C++ 6.0 集成环境和程序开发过程简介

在 Visual C++ 6.0 集成开发环境中集成了编辑器、编译器、连接器以及程序调试环境,覆盖了开发应用程序的整个过程,用户在这个环境中可以开发出完整的应用程序。

程序开发的一般步骤是:输入源程序文件、编译、连接、运行、调试和修改,如图 1-1 所示:

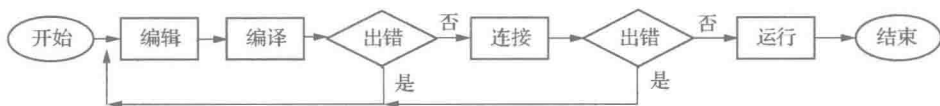


图 1-1

启动 Visual C++, 可用下述两种方法之一:

- 在桌面上选择 Visual C++ 6.0 图标 , 并双击。
- 单击任务栏“开始”按钮, 选择“所有程序 | Microsoft Visual Studio 6.0 | Microsoft Visual C++ 6.0”。

下面介绍利用 Visual C++ 的缺省项目管理方式, 开发 C++ 语言控制台程序的过程。

1. 创建新的源程序

假定用户连续创建两个新的 C++ 语言源程序 ex1.cpp, ex2.cpp, 可按如下步骤进行:

(1) 在 Windows 资源管理器中的 D 盘下以自己的学号姓名为名字创建文件夹, 如“031510899 张三”, 用于存放本次实验内容。

(2) 启动 Visual C++ 6.0 集成环境, 界面如图 1-2 所示:

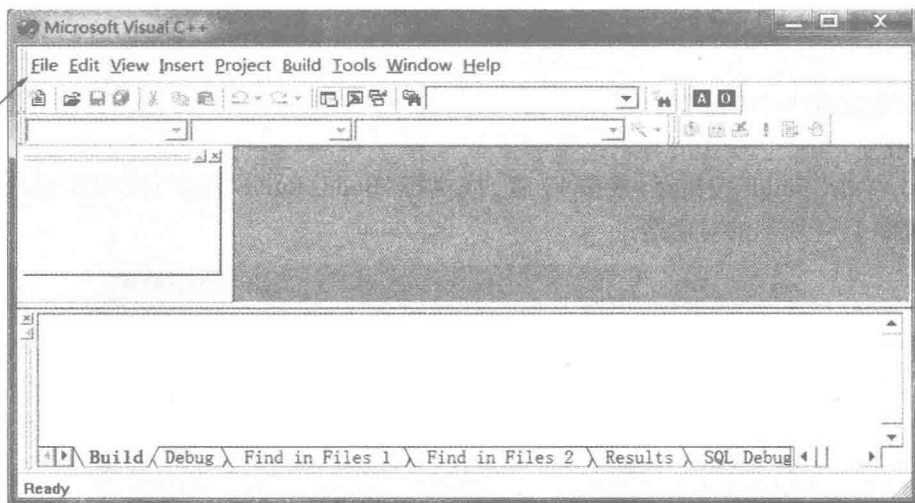


图 1-2

(3) 执行“File|New”命令,选择“Files”标签。

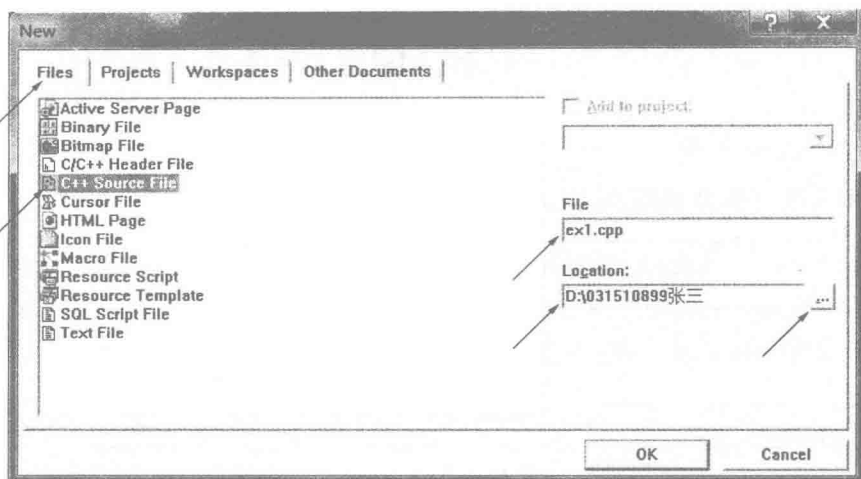


图 1-3

① 在图 1-3 左侧窗口中,选择创建文件的类型为 C++ Source File。


② 在 File 编辑框中输入 ex1.cpp。注意,扩展名 .cpp 可以不输入,因为前面已选择文件类型为“C++ Source File”,在单击“OK”按钮后,系统会自动增加该扩展名。


③ 在 Location 编辑框中选定(单击“浏览”按钮...进行选定)或输入“D:\031510899 张三”,表示将新创建的源程序文件 ex1.cpp 存入该文件夹中。

④ 单击“OK”按钮。

(4) 在 Visual C++6.0 集成环境中,在文本编辑窗口(参看本页集成环境窗口图)中输入源程序:

```
#include < iostream.h >
void main()
{
    cout << "C++ programming is powerful!\n";
}
```

(5) 输入完毕,单击“Save”按钮 (或执行“File|Save”命令),将源程序保存入文件 ex1.cpp 中。

(6) 单击“Build”按钮 (或按 F7 键,或执行“Build|Build”命令)对程序进行编译、连接,出现图 1-4 所示的对话框:

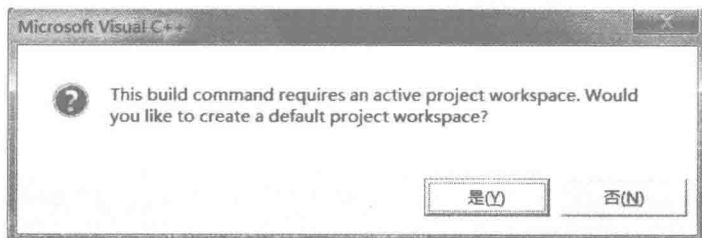



图 1-4

图 1-4 对话框询问用户是否建立一个缺省的项目工作空间,用户单击“是(Y)”按钮,系统对程序进行编译和静态连接。若程序无编译错,则窗口变为图 1-5 所示。若程序中有编译错误,则根据出错信息修改源程序后,重复第(6)步直到没有错误。注意:该步骤是将开发程序中的“编译”和“连接”工作按顺序完成。另一个按钮  表示只进行编译,不进行连接。

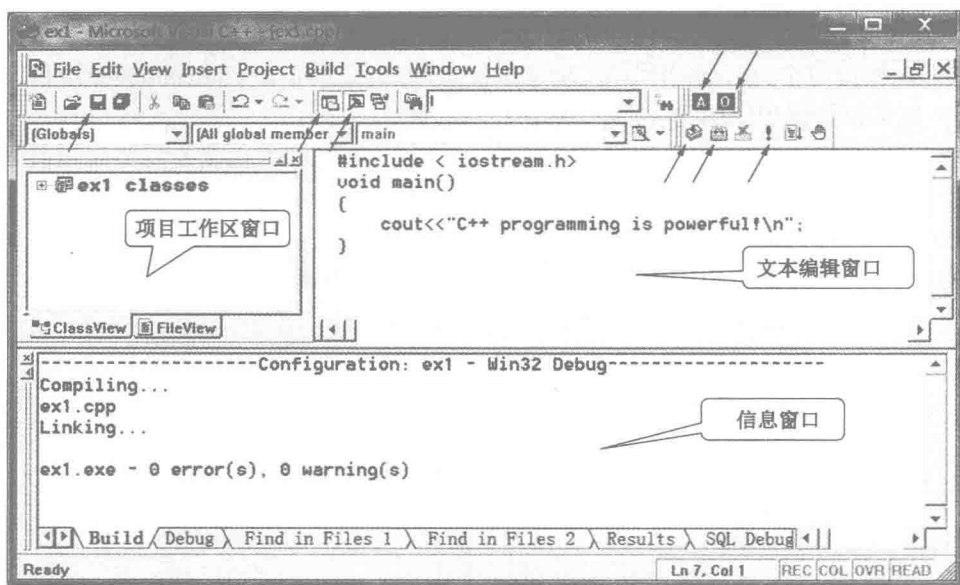





图 1-5

图 1-5 集成环境窗口有三个子窗口:

- 项目工作区窗口:在此窗口中显示工作区名称、系统资源等信息,可单击“Workspace”按钮  来显示和取消该窗口。
- 文本编辑窗口:在此窗口中,输入源程序。
- 信息窗口:在此窗口中显示出错信息或调试程序的信息。可单击“Output”按钮  来显示和取消该窗口。当信息窗口出现“ex1.exe - 0 error(s), 0 warning(s)”时,表示程序无编译错误和警告,然后可进行下一步操作:执行程序。若有错误,则在信息窗口列出若干出错信息行,双击出错信息行,则在程序文本编辑窗口中,光标自动定位于出错行。一般应从第一个出错信息行开始修改错误。

注意:在工具栏的右上方有两个按钮 **A** 和 **O**, **A** 代表“Project | Add to Project | Files”菜单命令, **O** 代表“File | Open File”菜单命令。这是由于软件系统兼容性而出现的问题, Visual C++ 6.0 在 Windows 7 操作系统中无法正确使用这两个命令。在安装相应软件补丁后才会出现这两个按钮。

(7) 单击“Execute Program”按钮  (或按 Ctrl + F5 键,或执行“Build | Execute ex1.exe”菜单命令)执行程序,屏幕出现如图 1-6 所示窗口:

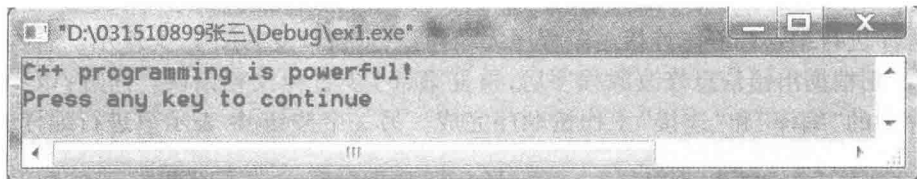


图 1-6

在图 1-6 窗口中,显示输出结果,用户按任意键(Press any key to continue)返回集成环境。用户还可以根据程序的需要在此窗口中输入数据。

(8) 当完成了 ex1. cpp 的开发,欲进行下一个程序的开发时,则首先要关闭当前程序的工作区,执行“File|Close Workspace”命令,系统提示如图 1-7 所示:

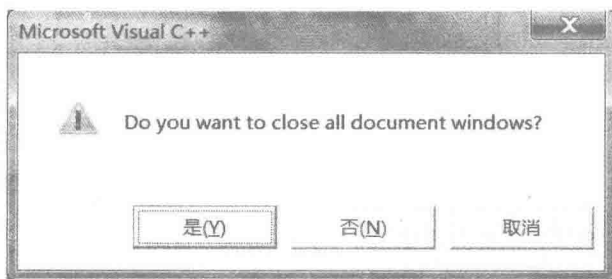


图 1-7

用户单击“是(Y)”按钮,表示在关闭工作区的同时关闭源程序的编辑窗口。若用户单击“否(N)”按钮,表示关闭工作区,但仍保留源程序的编辑窗口,以备稍后查看并修改该源程序。


(9) 新建第二个源程序 ex2. cpp,按上述(3)至(8)步骤进行。

(10) 用户可在 Visual C++ 集成环境的菜单“Window”中切换当前编辑的多个源程序窗口。

(11) 完成实验后,可将文件夹“D:\ 031510899 张三”中源程序文件 ex1. cpp、ex2. cpp 等做成压缩文档,保存到 U 盘或按教师指定的方法上交本次作业。



2. 修改旧的程序

公共机房的计算机不是个人专用的,用户硬盘每天清空,或者下一次上机换了一台计算机,此时用户欲修改前一次上机时保存在 U 盘中的旧的(即以前开发的)C++ 源程序,如 ex1. cpp、ex2. cpp,可按如下步骤进行:

(1) 在工作盘如 D 盘中建立文件夹“D:\ 031510899 张三”,将以前做过并存在 U 盘中的源程序拷入该文件夹。在资源管理器中观察,图标是  的,就是 C++ 源程序,一般自动关联到 Visual C++ 应用程序上。双击打开. cpp 源文件,即可修改运行该源程序。

或者按照下述(2)~(3)步进行:

(2) 将用户旧的源程序拷入指定文件夹,启动 Visual C++6.0 集成环境。

(3) 单击按钮  (或执行“File|Open”菜单命令,注意若操作系统是 Win7,则用按钮  替代),查找并打开指定文件夹中的 ex1. cpp 文件,如图 1-8 所示:

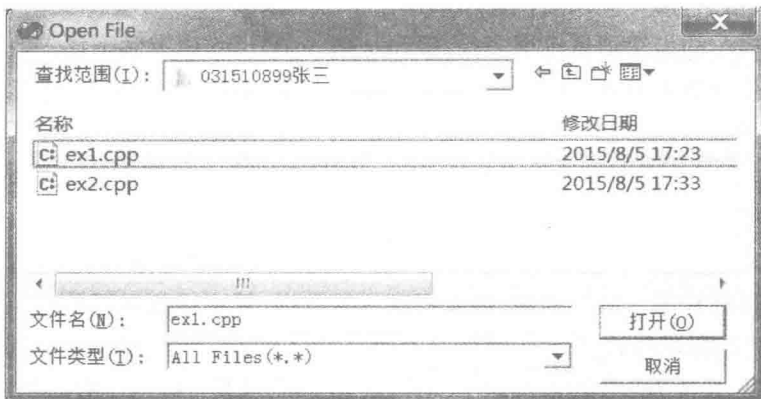


图 1-8

(4) 修改并运行完成当前的 ex1.cpp 源程序后,关闭程序 ex1.cpp 对应的工作区,再打开另一个源文件 ex2.cpp 进行修改。

二、Visual C++ 6.0 程序调试

程序的开发过程是:输入程序、编译连接运行程序、修改错误,直至将一个完整、正确的程序设计出来的过程。程序调试是指对程序的查错和排错工作。

程序的错误一般分为语法错误和逻辑错误两种。编译时出现的错误一般是语法错误,可根据编译时的“出错信息”,判断出现错误的语句。逻辑错误一般指程序算法流程设计和处理上的错误,程序运行结果不正确一般是程序逻辑错误造成的,所以需要根据算法流程来查找出错原因,这不是一件简单的事情,要求对算法的各个环节以及实现算法的各条语句有一个全面充分的认识,才能找出原因。还有一种运行时错误往往是由于程序对系统资源的使用不当造成的,一般也是程序的逻辑错误引起的。

调试程序一般应经过以下几个步骤:

(1) 首先进行人工检查,即静态检查。作为一个编程人员应当养成严谨的科学作风,每一步都要严格把关,不把问题留给后面的工序。

为了更有效地进行人工检查,所编写的源程序应注意力求做到以下几点:

① 应当采用缩进式书写格式,每行写一条语句,以增强程序的可读性。例如复合语句花括弧的匹配,若采用缩进式书写格式,就不容易出错。

② 在程序中尽可能多地加注释,以帮助理解每段程序的作用。

③ 在编写复杂的程序时,不要将全部语句都写在 main 函数中,而要多使用函数,用一个函数来实现一个单独的功能。这样既易于阅读也便于调试。

(2) 在人工(静态)检查无误后,才可以上机调试。通过上机发现错误的过程称为动态检查。在编译时系统会给出语法错误的信息(包括哪一行有错以及错误类型),可以根据提示信息具体找出程序中的出错之处并改正。应当注意的是:有时提示的出错行并不是真正的出错行,如果在提示出错行上找不到错误的话应当到可能的相关行再找。另外,有时提示的出错的类型并非绝对准确,由于出错的情况繁多而且各种错误互有关联,因此要善于分

析,找出真正的错误,而不要死抱住提示的出错信息不放。

如果系统提示的出错信息有很多条,应当从第一条开始,由上到下逐一改正。有时显示多条错误信息往往使人感到问题严重,无从下手。其实可能只有 1~2 个错误。例如,若某一变量未定义,编译时就会对所有使用该变量的语句发出错误信息。此时只要在前面增加一个变量定义,所有的错误都消除了。因此,在程序中的第一处错误修改完成之后,应立刻重新编译该程序。

(3) 在改正语法错误(包括“错误 error”和“警告 warning”)后,程序经过连接(link)就得到可执行的目标程序。运行程序,输入程序所需数据,就可得到运行结果。

应当对运行结果进行认真分析,判断是否符合题目要求。有的初学者看到针对一组数据输出了正确结果,就认为程序没有问题了,不作认真分析,这是危险的。有时程序比较复杂,难以根据一组数据判断结果是否正确,因为有可能输入另外一组数据时结果就不对了。所以应事先设计好一批全面的“实验数据”(测试数据),以验证在各种情况下程序的正确性。

事实上,当程序复杂时,很难把所有的可能的输入数据方案全部都测试一遍,选择典型数据做实验即可。

若程序的运行结果不对,大多属于逻辑错误。对这类错误往往需要仔细检查和分析程序逻辑才能发现。可以采用以下办法:

- 先检查流程图有无错误,即算法有无问题,如有错则改正之,接着修改程序。将程序与流程图(或伪代码)仔细对照,如果流程图是正确的话,那么如果程序写错了,会很容易发现的。

- 采取“分段检查”的方法。在程序不同的位置加入几个输出(cout)语句,输出有关变量的值,逐段往下检查。直到找到在某段中的变量值不正确为止。这时就把错误局限在这一段中了。不断缩小“查错区”,就可能发现错误所在。

在学习了函数和数组后,程序变得越来越复杂了,这时可使用 Visual C++ 系统提供的强有力的调试工具 Debug,跟踪程序的执行找出若干运行错误。下面用一个具体例子,说明 Visual C++ 调试程序 Debug 的使用。

下列程序的功能是:输入数组的 7 个元素值,将数组元素逆序存储后输出。

假定程序文件名为 ex3.cpp,程序如下:

```
#include <iostream.h>
void main()
{ int a[7], i, j, t;
  cout << "Please input 7 elements: ";
  for(i=0; i<7; i++)
    cin >> a[i]; //A
  for(i=0, j=6; i<j; i++, j--)
  { //B
    t = a[i];
    a[i] = a[j];
    a[j] = t;
```

```

}
for(i=0; i<7; i++)           //C
    cout <<a[i] <<" ";
cout <<endl;
cin.get();                  //D
}

```

输入以上程序并编译连接通过后,按 F10 键,进入调试状态。注意在调试状态下,Build 菜单变成 Debug 菜单,界面如图 1-9 所示:

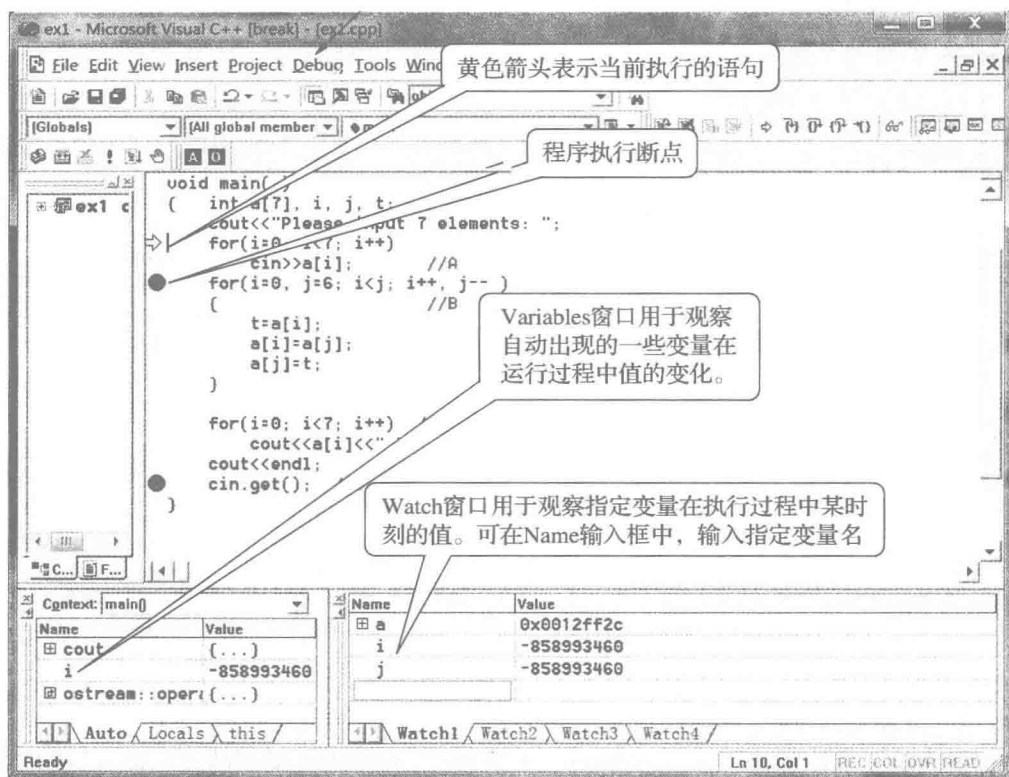


图 1-9

在调试阶段,各常用快捷键及功能见表 1-1 所示:

表 1-1 常用快捷键及功能

快捷键	工具按钮	功能	菜单命令
F10		单步跟踪执行,不进入被调用函数	Debug Step Over
F11		单步跟踪执行,进入被调用函数	Debug Step Into
Shift + F5		退出调试状态	Debug Stop Debugging
F5		执行到下一个断点处	Debug Go
F9		在光标所在行设置 / 取消断点	Edit Breakpoints

调试程序时,连续按 F10 键,单步执行各语句。按一次 F10 键,执行一条语句,当程序执行到 A 行时,无法继续执行,因为此时系统等待输入数据,用户在操作系统的任务栏中激活程序 ex3 的执行窗口,窗口中有提示信息“Please input 7 elements:”,可在提示信息后输入 7 个任意数据,如输入 1 2 3 4 5 6 7 <回车>,按回车后,再激活程序调试窗口。

如图 1-10 所示,在 Watch 窗口的第 1 个 Name 域中输入 a,第 2、第 3 个 Name 域中分别输入 i 和 j,单击数组名 a 前的“+”号,此时“+”号变成“-”号,表示展开了 a 数组元素,可观察 a 数组中的全体元素值。再单击 a 前的“-”号,则 a 数组元素折叠,又变回“+”号。

继续按 F10 键单步执行各语句,当程序执行每次到达 A 行或 B 行时,在 Watch 窗口观察数组 a 中的元素及变量 i 和 j 的变化,同时也可在 Variables 窗口查看变量值,如图 1-10 所示,是数组前 3 个元素输入完成时,程序中各变量值的情况。

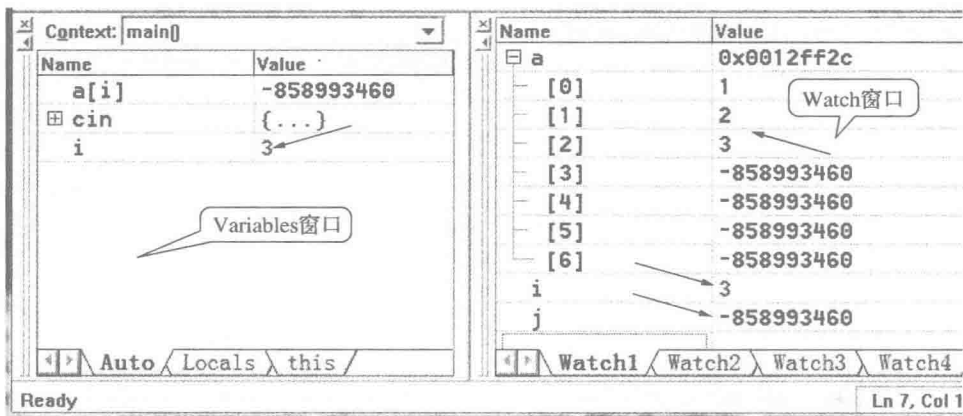


图 1-10

在程序编辑窗口中,按 F9 键可在光标所在当前行设置断点(Break Point),此时当前行的左边出现一个棕色的圆点,表示此行是断点行。断点的作用是,当某个程序段执行步数太多(如循环执行次数太多),同时可以确认该段程序无错时,对该程序段就不需要单步执行了,可在此段程序的下一语句起始处设置断点,在此段程序执行之前按 F5 键,此时程序从当前光标所在行一次性运行到下一个断点处,然后暂停,等待用户调试的下一动作,这样可以提高调试速度。

例如,用户可在 B 行所在的 for 语句前设置第 1 个断点,在 D 行设置第 2 个断点。按 F10 键启动调试,按 F5 键执行到第 1 个断点(期间需要输入数据),然后连续按 F10 键单步执行循环,在 Watch 窗口和 Variables 窗口观察各变量值的变化,直到 B 行所在的 for 循环完毕,程序执行到 C 行,然后按 F5 键执行到第 2 个断点 D 行。此时,程序已完成输出,用户可以从任务栏切换到程序的运行窗口,观察输出结果是否正确,然后再切换回到调试窗口,继续按 F10 键结束调试过程。注意上述快捷键(如 F10 键)均可以通过对应的工具按钮实现,参见表 1-1。

还有一点需要注意,按 F10 键为单步执行一条语句,若遇到函数调用语句,也是作为一条语句的(一次执行完毕)。因此若需要跟踪执行到被调函数的内部,则在函数调用语句行,需要按 F11 键进入被调函数,在被调函数内部,再按 F10 键继续单步执行。

二 上机实验内容

上机实验总的目的和要求

目的:

- (1) 熟悉 Visual C++ 6.0 程序开发集成环境,掌握C++ 语言的程序开发步骤。
- (2) 通过多次上机实习,加深对课堂教学内容的理解。
- (3) 学会上机调试程序。

要求:

(1) 每次上机前复习课堂上讲过的内容,预习实验内容。对于实验内容的每一题,以书面形式编写好程序,人工检查无误后才能上机,以提高上机效率。书面编写的程序应以缩进式书写风格书写整齐。编程时,思路要放开,应尽可能用多种算法实现。

程序缩进式的书写风格,基本要求如下(教材中和学习指导中没有按照这个格式书写是为了节省版面):

- ① 每行写一条语句;
- ② #include 和 main 之间空一行;
- ③ main 下一行的行首输入开花括号后立刻输入回车键;
- ④ 变量定义语句和可执行语句之间空一行;
- ⑤ 函数之间空一行;
- ⑥ 同一层次的缩进,开花括号和闭花括号要在同一列上;
- ⑦ 变量定义语句,如"int x, y, z;" 每个逗号之后、下一个变量之前空一格。

(2) 上机实习时应独立思考,刚开始上机时,对于一些编译错误,若自己解决不了,可请老师帮助解决。随着上机次数的增加,同学对编译出错信息等应该有意识地做一些积累,达到能独立调试程序的目的。

(3) 每次上机结束后,应将程序按照教师指定的方法上交,并整理出实验报告,内容包括:题目及源程序清单。同学可准备一个笔记本或活页夹,将每次上机实验通过的源程序记录于此本中,以备复习或与别的同学交流算法。另外也需要准备一个U盘,用于保存源程序。

(4) 本手册包含14个实验,可在U盘上建立文件夹"C++语言实验程序",并在其下建立14个子文件夹,每个子文件夹中保存相应实验的源程序及数据文件清单。文件夹结构如下:

