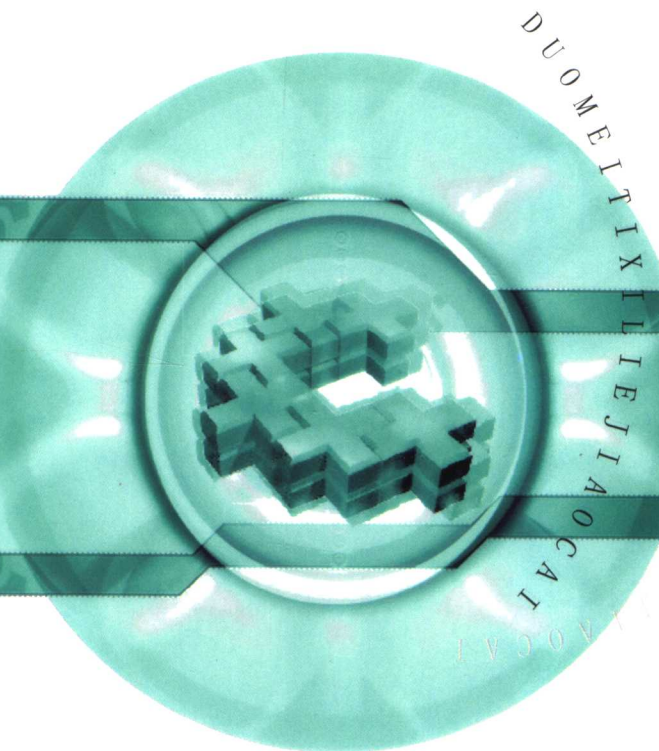


高等院校非计算机专业教材
(根据教育部对计算机基础教学的三个层次的要求编写)

C++程序设计

实习指导与习题集

李敏 王刚 崔宝深 编著
沈琴婉 主审



南开大学出版社

高等院校非计算机专业教材

(根据教育部对计算机基础教学的三个层次的要求编写)

C++程序设计

实习指导与习题集

李敏 王刚 崔宝深 编著

沈琴婉



南开大学出版社
天津

内 容 提 要

本书是与《C++程序设计教程》(李敏、王刚、崔宝深编著)配套的教材,由多年从事计算机基础教学的教师编写。其内容包括两篇:第1篇实习指导介绍了 Visual C++ 6.0 集成开发环境、程序调试方法、C++库函数及结合各章内容的上机实习指导。第2篇是习题集,习题与相应章节内容相对应。书中习题数量较多,读者可以根据个人情况选作其中部分习题。

本书不仅适合高等院校非计算机专业学生使用,同时,对正在学习 C++ 程序设计的读者也颇具参考价值。

图书在版编目(CIP)数据

C++程序设计实习指导与习题集 / 李敏, 王刚, 崔宝深编著. 天津: 南开大学出版社, 2005.9
ISBN 7-310-02354-4

I. C... II. ①李...②王...③崔... III. C 语言—程序设计—高等学校—教学参考资料 N. TP312

中国版本图书馆 CIP 数据核字(2005)第 066173 号

版权所有 侵权必究

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

*

河北省迁安万隆印刷有限责任公司印刷

全国各地新华书店经销

*

2005 年 9 月第 1 版 2005 年 9 月第 1 次印刷

787×1092 毫米 16 开本 9.25 印张 130 千字

定价: 16.00 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

对于高等院校非计算机专业的大多数同学而言,学习计算机课程的目的主要在于应用,而学习程序设计对提高计算机应用水平十分重要。程序设计是一种极富创造性的智力劳动,对培养同学们解决问题的能力、培养创新意识具有重要意义。学习程序设计要坚持理论与实践密切结合的原则,二者不能脱节或偏废。

本书是与《C++程序设计教程》(李敏、王刚、崔宝深编著)配套的教材,由多年从事计算机基础教学的教师编写。其内容包括两篇:第1篇实习指导介绍了 Visual C++ 6.0 集成开发环境、程序调试方法、C++库函数及结合各章内容的上机实习指导。第2篇是习题集,习题与相应章节内容相对应。书中习题数量较多,读者可以根据个人情况选作其中部分习题。

本书第1章、第2章、第3章、第4章中的实习1~实习6、实习20、实习21和习题1~习题4、习题12、习题13由崔宝深编写,第4章中的实习7~实习11和习题5、习题6、习题7由李敏编写,第4章中的实习12~实习19和习题8、习题9、习题10、习题11由王刚编写,全书由崔宝深教授统稿,沈琴婉教授主审。

本书编写过程中得到沈琴婉教授、姚淑琴老师和南开大学出版社张蓓同志的大力支持和帮助,责任编辑李正明同志和尹建国同志对书稿提出了许多宝贵意见,并对书中的错误和疏漏一一加以改正,这种对读者和作者高度负责的精神,令人敬佩,在此一并致以诚挚的感谢!

由于编者水平所限,书中难免会有错误与不妥之处,敬请读者批评指正。

编者

2005年4月

目 录

第 1 篇 实 习 指 导

第 1 章 Visual C++ 6.0 集成开发环境	3
1.1 Visual C++ 6.0 集成开发环境简介	3
1.2 开发 C++ 程序的基本过程	6
1.3 创建 C++ 程序的一般方法	8
1.4 创建 C++ 程序的简捷方法	11
1.5 增加类	12
第 2 章 程序调试	14
2.1 程序中错误的类型	14
2.2 程序调试方法简介	15
2.3 程序中常见错误分析	18
第 3 章 C++ 库函数	25
3.1 数学函数	25
3.2 I/O 函数	26
3.3 字符函数	27
3.4 字符串函数	27
3.5 其他函数	28
第 4 章 上机实习	29
实习 1 文件、文件夹及磁盘的基本操作	29
实习 2 简单程序设计	30
实习 3 选择结构	32
实习 4 循环结构 (一)	35
实习 5 循环结构 (二)	38
实习 6 数组的应用	43
实习 7 函数 (一)	46
实习 8 函数 (二)	47

实习 9	编译预处理	49
实习 10	指针的使用	50
实习 11	指针和引用	51
实习 12	类与对象 (一)	52
实习 13	类与对象 (二)	54
实习 14	继承 (一)	55
实习 15	继承 (二)	56
实习 16	虚函数与多态性	57
实习 17	运算符重载	58
实习 18	文件操作	59
实习 19	模板的使用	62

第 2 篇 习 题 集

习题 1	67
习题 2	70
习题 3	74
习题 4	87
习题 5	99
习题 6	106
习题 7	109
习题 8	119
习题 9	126
习题 10	130
习题 11	132
习题 12	136
习题 13	140

第 1 篇

实 习 指 导

第1章 Visual C++ 6.0 集成开发环境

1.1 Visual C++ 6.0 集成开发环境简介

用于 C++ 程序开发的系统有多种。我们选用 Microsoft 公司开发的 Visual C++ 6.0（以下简称 VC++），因为它是一个面向对象、能自动生成程序框架、可视化、功能强大的程序开发系统。在中文 Windows 下，它的编辑器能够处理中文信息。

VC++ 有 3 种不同的版本，即标准版、专业版和企业版。VC++ 是 Visual Studio 6.0 软件包中的一部分，它可以单独安装，也可以与 Visual Studio 同时安装。安装过程几乎是自动的。

启动 VC++ 后，屏幕上显示 VC++ 集成开发环境，如图 1-1 所示。

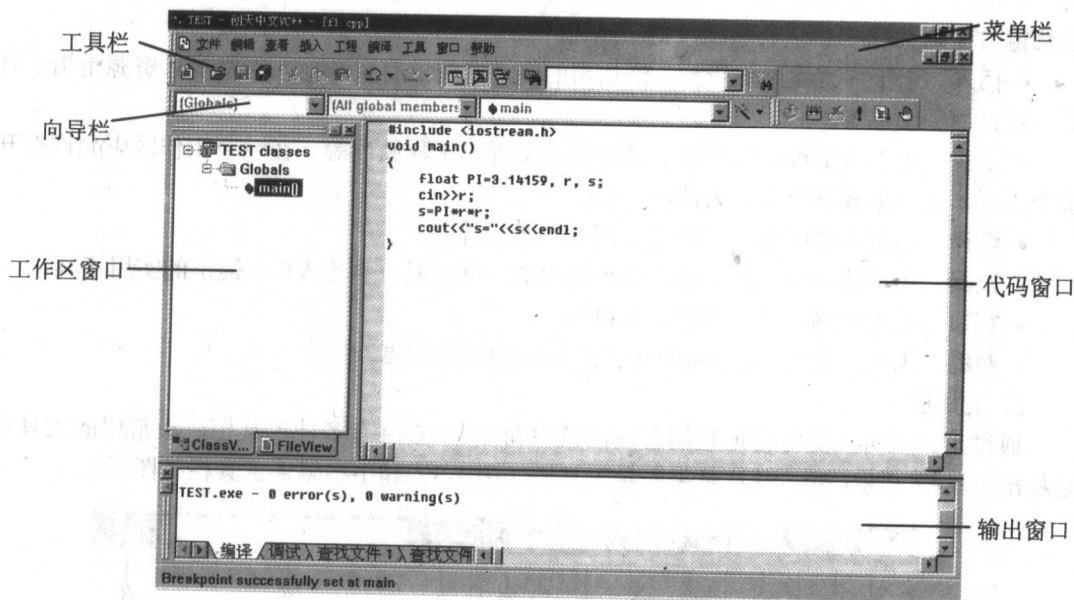


图 1-1 Visual C++ 6.0 集成开发环境

VC++ 集成开发环境主要用于 Windows 程序的开发。它工具多、功能强。由于本课程的目标是 C++，因此，下面只介绍与 C++ 程序设计有关的内容，主要是在该集成环境中编辑、编译、连接和运行 C++ 程序的操作方法。

首先介绍两个概念：工程（Project）和工作区（Workspace）。通常，一个 Windows 应用程序是由多个文件组成的，为了便于组织和管理这些文件，VC++ 引入了工程（或称项目）的概念。所谓工程，是指构成某个应用程序的一系列文件的集合，包括代码文件、资源文件和管理文件。工程的概念使得编程工作更有条理，更具模块化。工作区用于管理工程，一个工作区可以管理多个工程，甚至是不同类型的工程，工程之间彼此独立，但共用一个工作区

的环境设置。工作区由专门的管理文件管理，并且在集成环境中专门设置了工作区窗口，用以组织和显示当前工作区中的内容。

开发 C++ 程序时，首先要创建工程。新建一个工程时，同时也就新建了一个工作区。当然，也可以将已有的工程加入到当前工作区，使一个工作区中有多个工程。VC++ 提供了多种向导来帮助创建不同类型的工程，其中用于 C++ 程序开发的是控制台应用程序向导 Win32 Console Application，在它的引导下用户可以快速创建 C++ 工程。

VC++ 集成开发环境由菜单栏、工具栏、向导栏、工作区窗口、代码窗口和输出窗口等组成。

1. 菜单栏

菜单栏中主要选项有：

- 文件 用于创建、打开、保存工程及其他文件。常用的 C++ 应用程序有：Win32 Console Application（控制台应用程序）、Win32 Application（Windows 应用程序）等。

- 编辑 用于文件的编辑，如输入源程序代码、复制、粘贴、删除等操作，有些与 Word 编辑命令相同。

- 查看 用于激活调试时所用的各种窗口，如工作区窗口、输出窗口、监视窗口（Watch）、变量窗口（Variables）、内存窗口（Memory）等。

- 插入 用于向工程添加新类、添加新的资源或将已有资源插入工程等。资源是指工具栏、对话框、图标等。

- 工程 用于向工程添加文件，设置活动工程、工程属性等。在一个工作区中允许打开多个工程，可以选择某个工程为活动工程。

- 编译 用于程序的编译、构建、调试、运行等。

- 工具 用于选择或定制集成开发环境中的一些工具，调用 VC++ 提供的实用工具。

- 窗口 用于排列、显示或隐藏窗口等。

- 帮助 为用户使用 VC++ 提供了大量详细的帮助信息。

2. 工具栏

通过工具栏可以更方便地使用常用的菜单命令。VC++ 有多种工具栏，最常用的工具栏是标准工具栏（图 1-2），其次是类向导工具栏（图 1-3）和小型编连工具栏（图 1-4）。

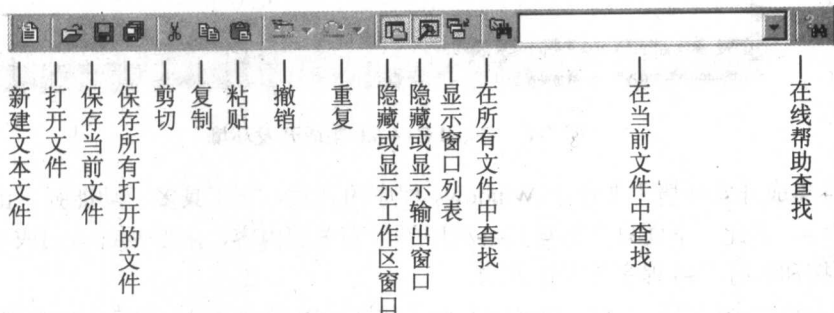


图 1-2 标准工具栏

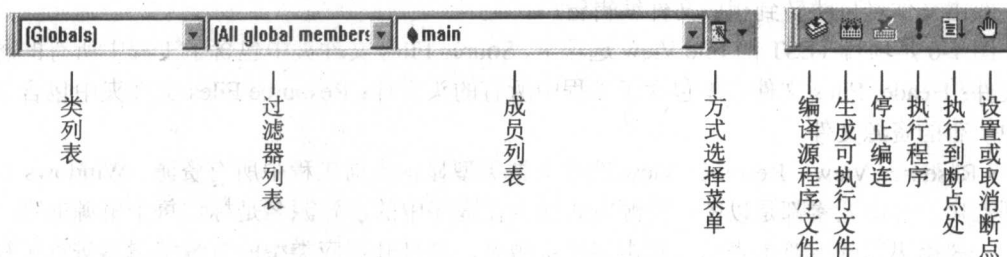


图 1-3 类向导工具栏

图 1-4 小型编连工具栏

要显示或隐藏工具栏，最便捷的方法是在任何工具栏上单击鼠标右键，在弹出的快捷菜单中如果选定一个工具栏名称，则该工具栏显示在窗口中；如果取消某工具栏的选定状态，则该工具栏被隐藏。

3. 工作区窗口

工作区窗口中有 3 个选项卡：Class View、File View 和 Resource View（如果是 Win32 Console Application 程序，则没有 Resource View 选项卡。Win32 Console Application 程序即控制台应用程序，这是在 Windows 环境下运行的 DOS 程序，它不需要为用户提供完善界面），以树状结构分别显示当前工程的类、文件和资源。

• Class View Class View 选项卡中显示当前工程中所有类、全局变量和全局函数。单击类名前的“+”号，可展开显示类的数据成员及函数成员，每个成员名前还有一个小图标，表示是数据成员或函数成员。如果该成员是私有的，则前面有一个小锁形图标；如果是保护成员，则前面有一个钥匙形的图标。双击类名或成员名，就跳转到源文件或头文件中类或成员的定义处。

图 1-5 是项目 TEST 的 Class View 选项卡，表明该项目只有一个 main() 函数。

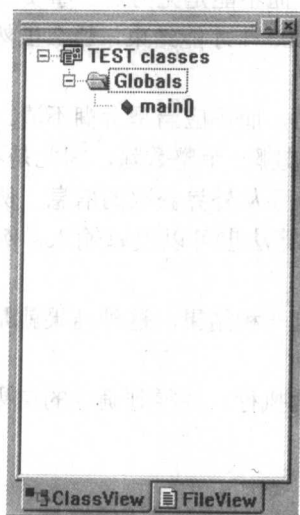


图 1-5 Class View 选项卡

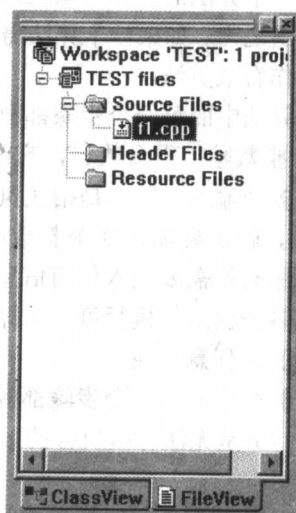


图 1-6 File View 选项卡

- **File View** File View 选项卡中按文件类型显示当前工程的源文件、头文件、资源文件等。双击文件名可跳转到相应文件编辑窗口。

图 1-6 是项目 TEST 的 File View 选项卡, Source Files 文件夹中包含了工程中所有的源程序文件; Header Files 文件夹中包含了工程中所有的头文件; Resource Files 文件夹中包含了工程中所有的资源文件。

- **Resource View** Resource View 选项卡按类型显示当前工程中所有资源。Windows 程序中图形用户界面元素都是以图形资源方式保存在程序中的, 资源名是标识每个资源的唯一名称, 一般用大写的字符串表示。双击某个资源名, 可打开相应类型的资源编辑器进行可视化资源编辑。

4. 输出窗口

输出窗口有编译、调试等多个选项卡, 最常用的是“编译”选项卡, 在编译、连接时, 该窗口中显示有关信息, 供调试程序使用。

1.2 开发 C++程序的基本过程

开发一个 C++程序的基本过程可分为 3 个阶段:

1. 分析问题, 建立问题的模型, 确定算法

首先要对具体问题进行深入的分析、研究, 建立起问题的模型, 找出合适的算法(解决一个问题的算法往往不是唯一的)。所谓算法是指解决一个问题而采取的方法与步骤。然后将算法用某种形式(如自然语言、流程图、伪代码等)表示出来, 这是最关键的一步。我们应当养成一种好习惯: 先设计算法后编写程序, 不要拿到题目就马上动手写程序。

算法可分为两大类: 数值算法和非数值算法。前者是求数值解, 如求一个方程的根, 求一个函数的定积分等; 后者如排序、检索、事务管理等, 内容十分广泛。

算法的特性主要有:

- **有穷性** 一个算法的操作步骤应该是有穷的, 而不能是无穷的。事实上, 有穷性往往是指在合理的范围之内。如果一个算法需要执行 10 000 年才能结束, 那么虽然是有穷的, 但人们也不把它视作有效算法。

- **确定性** 算法中的每一个步骤都应当是确定的, 而不应当是含糊不清、模棱两可的。例如, “n 被一个整数除, 得余数 r”, 它没有说明 n 被哪一个整数除, 因此是不确定的。

- **有零个或多个输入** 输入是指在执行算法时需要从外界获取的信息。例如, 求 2 个整数的最大公约数, 就需要输入 2 个整数的值。一个算法也可以没有输入。例如, 在执行打印 1 行星号的程序时不需要输入任何信息。

- **有一个或多个输出** 执行每一个算法都会得到一种结果, 这种结果就是输出。试想, 没有输出的算法又有何意义呢?

- **有效性** 算法中的每一个步骤都应当能有效地执行, 并得到确定的结果。例如, 如果 x 的值为 0, 则 y/x 是无法有效执行的。

2. 编写程序

在已写出算法的基础上编写程序称为编码, 这一步相对来说比较容易。

3. 上机调试

上机调试包括创建源程序文件、编译、连接和运行, 用实际数据进行测试, 发现并修改

程序中的错误。程序调试完毕,经过一段时间的试运行没有发现新问题后,才可以交付使用。

下面举两个简单例子,用于说明如何建立模型和确定算法。

【例 1-1】在高度为 100m 的铁塔上平抛一物体,初速度为 20m/s,求其运动轨迹。以 0.1s 为时间间隔,直到物体落到地面为止。

分析:设坐标原点在塔底,物体初始位置是 $x=0$, $y=100$ 。如果用 v_0 表示初速度,则物体在时刻 t 的坐标是:

$$x = v_0 t$$

$$y = 100 - \frac{1}{2}gt^2$$

这两个公式就是问题的数学模型。该问题要求求出物体的运动轨迹,即每隔 0.1s 计算一对坐标值。采用的算法是:利用循环结构按以上公式计算每一组 x 、 y 的值,直到 $y=0$ 为止。

先用自然语言对算法进行描述:

- ① 定义变量、赋初值。
- ② 计算 $t=0.1$ 时物体的坐标 x 、 y 并输出,然后 t 增加 0.1。
- ③ 判断:如果 $y>0$,则重复步骤②,否则结束。

再用流程图描述算法,如图 1-7 所示。

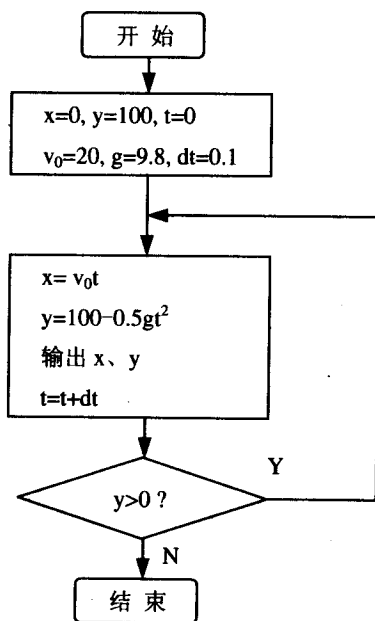


图 1-7

【例 1-2】判断一个整数 n 能否同时被 3 和 5 整除。

分析:判断一个整数 n 能否同时被 3 和 5 整除,只要让 n 分别被 3 和 5 除,如果余数都为 0,则能同时被 3 和 5 整除,否则不能同时被 3 和 5 整除。

用自然语言描述算法如下:

- ① 定义变量,输入 n 。
- ② n 被 3 除,如果余数为 0,则执行第③步,否则执行第⑤步。

- ③ n 被 5 除, 如果余数为 0, 则执行第④步, 否则执行第⑤步。
- ④ 输出 “ n 能同时被 3 和 5 整除”, 执行第⑥步。
- ⑤ 输出 “ n 不能同时被 3 和 5 整除”。
- ⑥ 结束。

用流程图描述算法, 如图 1-8 所示。

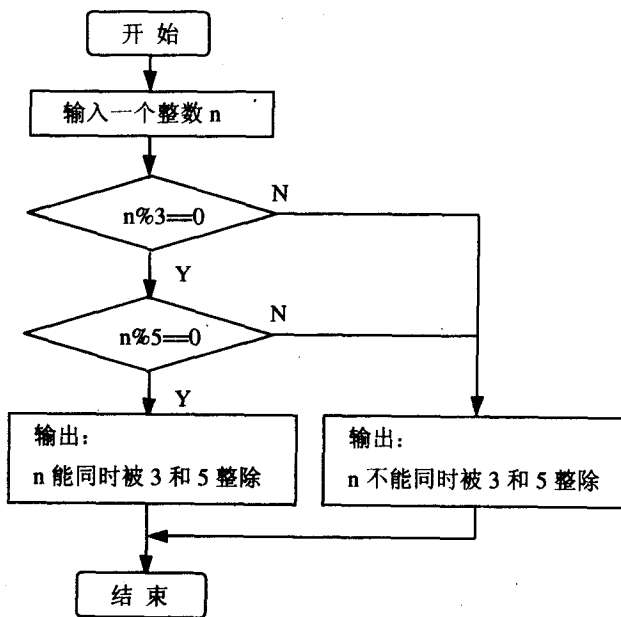


图 1-8

1.3 创建 C++ 程序的一般方法

创建一个 C++ 程序可分为 3 步: 创建一个空工程; 创建 C++ 源程序文件; 对源程序文件进行编译、连接和运行。

1. 创建一个空工程

首先创建一个空工程文件。打开“文件”菜单, 选择“新建”, 弹出“新建”对话框, 如图 1-9 所示。

在该对话框的“工程”选项卡中, 选定“Win32 Console Application”, 即 32 位控制台应用程序。输入工程名, 这时系统自动在“位置”编辑框中创建一个与工程同名的文件夹, 用于存放该工程中的所有文件。注意, 要确保单选按钮“R 创建新工作区”处于选定状态, 单击“确定”按钮, 弹出向导对话框, 如图 1-10 所示。

在该对话框中, 选定“An empty project”, 单击“完成”按钮, 显示新建工程的有关信息, 再单击“确定”按钮, 创建空工程的工作结束。此时, 在 D:\TEST 文件夹中生成了文件 TEST.dsp、TEST.dsw 和文件夹 Debug。文件夹 Debug 用于存放编译、连接过程中产生的文件。

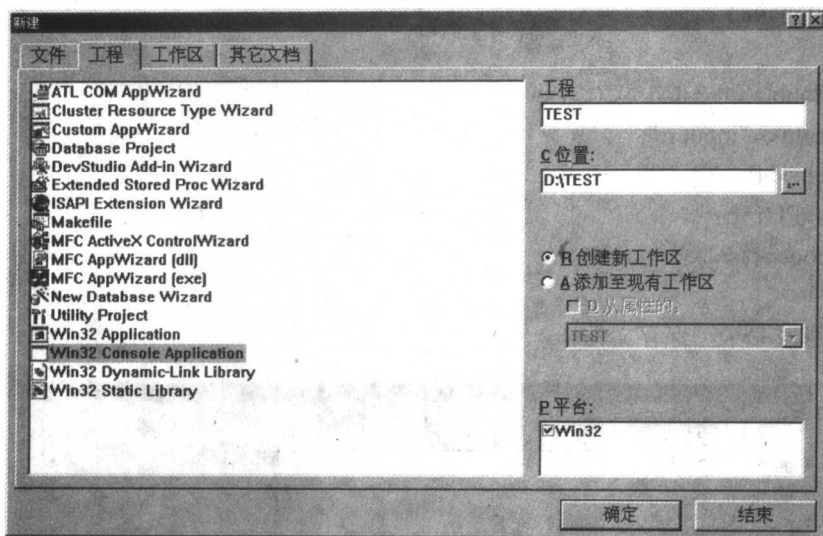


图 1-9 “新建”对话框

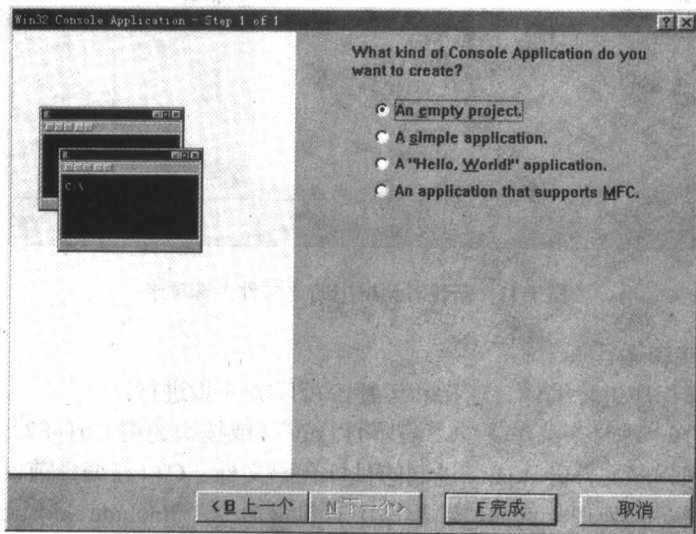


图 1-10 向导对话框

2. 创建 C++ 源程序文件

打开“文件”菜单，选择“新建”，弹出“新建”对话框。在该对话框的“文件”选项卡中，选定“C++ Source File”，并输入源程序文件名 f1，如图 1-11 所示，单击“确定”按钮。注意，如果程序中已有多个文件，则新建文件不能与已有的文件重名。这些文件中，有一个文件可以与工程同名。

然后在代码窗口中编辑源程序。编辑是指将源程序由键盘输入到计算机内并进行修改，最后以文件形式保存到磁盘中。例如，输入下面计算圆面积的源程序：

```

#include <iostream.h>
void main()
{
    double PI=3.1415926, r, s;
    cout<<"input r:";
    cin>>r;
    s=PI*r*r;
    cout<<"s="<<s<<endl;
}

```

输入、修改完毕，保存文件。

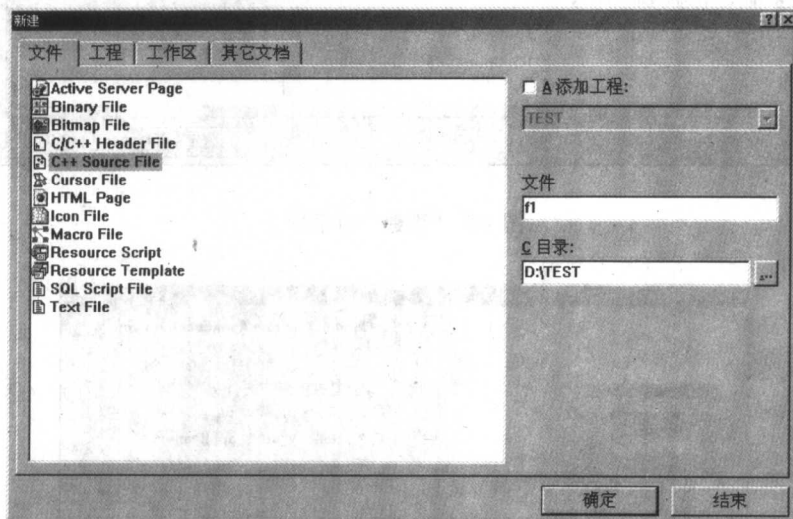


图 1-11 新建对话框中的“文件”选项卡

3. 编译、连接和运行

对已创建的源程序进行编译、连接和运行，可以分 3 步进行：

① 编译 选择“编译”菜单中的“编译 f1.cpp”（或按快捷键 Ctrl+F7），开始进行编译。编译的作用是将源程序文件翻译成二进制的目标代码文件。C++在编译前，首先使用编译预处理器对源文件进行预处理。例如，源程序中有包含命令“`#include <iostream.h>`”，预处理时把头文件 `iostream.h` 中定义的有关输入和输出的信息包含到源程序中来。编译结果显示在输出窗口中，如果没有错误，则生成 `f1.obj` 文件。

② 连接 经过编译生成的目标文件还是零散的，还没有得到系统分配的绝对地址，计算机还不能直接执行。连接的作用是将各目标文件以及需要由系统提供某些文件组装成一个具有绝对地址的可执行文件。连接的方法是：选择“编译”菜单中的“构建 TEST.exe”（或按功能键 F7）。连接结果显示在输出窗口中，如果没有错误，则生成 `f1.exe` 文件。

③ 运行 选择“编译”菜单中的“执行 TEST.exe”（或按快捷键 Ctrl+F5，或直接单击工具栏上的运行按钮），打开运行窗口，如图 1-12 所示。在运行窗口中可以输入数据并显示计算结果。例如，输入 10，则显示 `s=314.159`。按任意键即可返回 VC++集成开发环境。

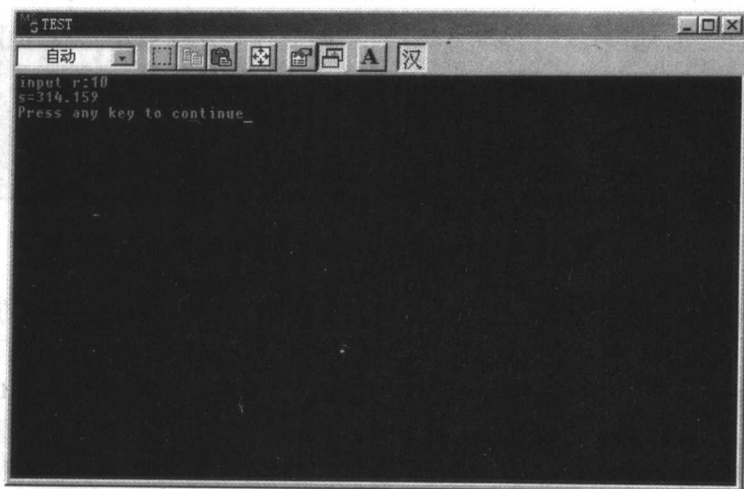


图 1-12 显示运行结果

请注意：在编译或连接时，如果发现错误，则显示出错信息。按功能键 F4，光标将自动指向错误源代码所在的行，根据提示进行修改后，再重新进行编译和连接，直到源代码没有错误为止。

此外，选择“编译”菜单中的“重建全部”选项，即可自动完成源程序的编译和连接。

至此，一个 C++ 程序的创建过程结束。在整个过程中，VC++ 为该工程生成了多种文件。它们用于工程管理。例如，工程文件 TEST.dsp 保存着工程的有关信息；工作区文件 TEST.dsw 保存工作区中工程的数量、工程名称等信息；Debug 文件夹中存放了编译、连接过程中生成的中间文件以及最终生成的可执行文件。

1.4 创建 C++ 程序的简捷方法

在学习 C++ 的过程中，我们每次上机实习都要创建、调试多个程序，如果每个程序都要先创建工程后创建程序文件，实在有些麻烦。下面介绍直接创建程序文件的简捷方法，而工程留给系统去创建。具体操作步骤如下：

① 在 VC++ 集成开发环境下，打开“文件”菜单，单击“新建”，弹出新建对话框。在该对话框的“文件”选项卡中，选定“C++ Source File”，输入源程序文件名（如 ex1.cpp）并在“目录”编辑框中输入文件夹名（如 D:\P1，该文件夹事先已创建），单击“确定”按钮。

② 在代码窗口中输入、编辑源程序并存盘。编译时弹出一个如图 1-13 所示的对话框，提示：Build 命令需要一个活动的项目工作区，你准备创建一个默认的项目工作区吗？如果单击“是”按钮，那么系统将自动创建一个默认的项目工作区，工程名与源程序同名。如果编译成功，则可继续连接和运行程序了。

注意，如果还要创建第 2 个、第 3 个等等源程序文件，则不必重复步骤①，可以在当前源程序的基础上进行修改，并以原文件名保存。如果以新文件名保存，则连接时就会因为一个程序中有两个主函数而出错。作为练习而不是开发 C++ 应用程序，这样处理是比较快的。