



普通高等教育“十三五”规划教材



语言案例教程

张春芳 © 主编



 科学出版社

普通高等教育“十三五”规划教材

C 语言案例教程

张春芳 主编

姜雪 朱姬凤
张宇 刘莹昕 副主编

科学出版社

北 京

内 容 简 介

本书主要包括 C 语言程序结构、运行环境、各种数据类型、运算符及使用方法、顺序结构、选择结构、循环结构、数组、函数、编译预处理、指针、结构体和共用体、文件等。

本书适合高等院校计算机及相关理工科专业的学生使用,也可以作为其他专业计算机语言类公共课教材,以及程序设计初学者、参加计算机二级考试者和计算机工程技术人员的参考用书。

图书在版编目(CIP)数据

C 语言案例教程/张春芳主编. —北京: 科学出版社, 2017
(普通高等教育“十三五”规划教材)

ISBN 978-7-03-051259-8

I. ①C… II. ①张… III. ①C 语言-程序设计-高等学校-教材
IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2016) 第 325122 号

责任编辑: 宋 丽 王 惠 / 责任校对: 刘玉靖
责任印制: 吕春珉 / 封面设计: 东方人华平面设计部

科学出版社 出版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

三河市骏杰印刷有限公司印刷

科学出版社发行 各地新华书店经销

*

2017 年 3 月第 一 版 开本: 787×1092 1/16

2017 年 3 月第一次印刷 印张: 22 1/4

字数: 513 000

定价: 49.00 元

(如有印装质量问题, 我社负责调换 <骏杰>)

销售部电话 010-62136230 编辑部电话 010-62135397-2052

版权所有, 侵权必究

举报电话: 010-64030229; 010-64034315; 13501151303

前 言



C 语言是一种较为通用的结构化程序设计语言,它兼有高级语言的特点和低级语言的功能,既可以编写系统软件,如 UNIX、FoxBASE 等,也可以用来编写应用软件,如信息系统软件等。

目前,国内外的许多高等院校开设了 C 语言程序设计课程,并将其作为计算机基础教育的入门课程,要求计算机专业和大多数非计算机专业学生必须掌握。同时,C 语言程序设计也是全国计算机等级考试中比较重要的科目之一。

C 语言课程教学的任务:一方面,使学生掌握 C 语言的语法规则,学会用 C 语言设计基本算法、编写程序解决实际问题;另一方面,使学生掌握程序设计的基本规律,能够活学活用、举一反三,为学习其他高级程序设计语言打下良好的基础。为此,编者结合多年的教学实践经验和体会编写了本书。

本书具有以下特点:

1) 以案例为主线,采用“提出问题—分析问题—解决问题”的方式阐述程序设计的思路与方法,可以有效地调动学生学习 C 语言的兴趣。

2) 从实际案例出发,介绍 C 语言的基本概念、语法规则及规则的应用方法,既能使学生掌握相应的知识点,又能培养学生应用知识及编写程序的能力。

3) 本书的体系结构简单、明了,基本内容的介绍力求由浅入深、概念清楚、结构严谨、通俗易懂。本书重点、难点内容的分析深入透彻,易于理解和掌握。

4) 良好的源程序书写风格、算法优化和程序设计风格贯穿全书。

本书共分 10 章,循序渐进地介绍了 C 语言的基本概念、数据的输入和输出、数据类型及运算、选择结构程序设计、循环结构程序设计、数组、函数、编译预处理、指针、结构体与共用体、文件操作等内容。另外,各章附有适量的习题,便于使用者自学。

本书由张春芳担任主编,姜雪、朱姬凤、张宇、刘莹昕担任副主编。

由于编者水平有限,书中难免存在不妥之处,恳请广大读者批评指正。

编 者

2016 年 12 月

目 录

第 1 章 简单 C 程序设计示例	1
1.1 一个简单的 C 语言程序	1
1.2 C 语言程序开发环境	4
1.3 运行 C 语言程序的步骤	11
第 2 章 数据的输入/输出与运算	20
2.1 整型数据的输出	21
2.2 实型数据的输出	24
2.3 实型数据的精度	26
2.4 字符型数据的输出	27
2.5 printf() 函数的附加格式说明符	30
2.6 转义字符的使用	32
2.7 用 scanf() 函数输入数据	34
2.8 scanf() 函数的附加格式说明符	37
2.9 getchar() 函数和 putchar() 函数	39
2.10 算术运算	41
2.11 自增、自减运算	43
2.12 赋值及复合赋值运算	45
2.13 关系运算	47
2.14 逻辑运算	49
2.15 类型转换	52
2.16 求字节数运算	54
2.17 条件运算	55
2.18 逗号运算符	57
2.19 位运算	59
第 3 章 选择结构程序设计	62
3.1 单分支选择结构	62
3.2 双分支选择结构	64

3.3	多分支选择结构	66
3.4	嵌套选择结构	69
3.5	多分支 switch 语句	72
第 4 章	循环结构程序设计	77
4.1	while 语句	77
4.2	do-while 语句	79
4.3	for 语句	81
4.4	break 语句	83
4.5	continue 语句	86
4.6	多重循环	87
第 5 章	数组	90
5.1	一维数组	90
5.2	二维数组	94
5.3	字符数组	97
第 6 章	函数设计示例	103
6.1	函数概述	103
6.2	函数定义	105
6.3	函数参数与返回值	113
6.4	函数调用的一般形式	119
6.5	函数的嵌套调用	122
6.6	函数的递归调用	125
6.7	数组作为函数的参数	132
6.8	局部变量与全局变量	142
6.9	变量的存储类别	146
6.10	内部函数与外部函数	155
第 7 章	编译预处理	158
7.1	宏定义	158
7.2	文件包含	164
7.3	条件编译	167
第 8 章	指针	171
8.1	地址与指针	171
8.2	变量与指针	173

8.3 指针与数组	182
8.4 字符串与指针	199
8.5 指向函数的指针	209
8.6 返回指针值的函数	213
8.7 指针数组和指向指针的指针	216
第 9 章 结构体与共用体	227
9.1 结构体	227
9.2 结构体数组	233
9.3 指向结构体类型数据的指针	236
9.4 单向链表	242
9.5 共用体	256
9.6 枚举类型	263
9.7 用 typedef 定义类型	267
第 10 章 文件	271
10.1 文件的打开和关闭	271
10.2 文件的字符输入和输出	272
10.3 文件的字符串输入和输出	275
10.4 文件的格式化输入和输出	277
10.5 文件的数据块输入和输出	280
10.6 文件的定位	282
附录	285
附录 A 常用字符与 ASCII 码对照表	285
附录 B Visual C++ 编译出错信息	286
附录 C 全国计算机等级考试二级 C 语言程序设计考试大纲	288
附录 D 全国计算机等级考试二级 C 语言程序设计试卷	292
附录 E 练习题参考答案	315
参考文献	346

第1章 简单 C 程序设计示例

学习目标

- ◇ 掌握 C 语言程序的基本结构。
- ◇ 熟悉 C 程序的开发环境。
- ◇ 掌握运行一个 C 语言程序的步骤。
- ◇ 熟练掌握 C 语言的数据类型、常量和变量。

人们在使用计算机解决各种各样的问题时，都要预先设计好解决问题的步骤和过程，而且要用计算机可以识别的代码来表示这些步骤和过程，这些计算机可以识别的代码就称为计算机程序（Computer Program），简称程序。

编写计算机程序所使用的语言称为程序设计语言，它是程序员与计算机进行交流的工具。在程序中，一般要表达数据，包括定义用于存储数据的变量，描述数据处理的过程。

C 语言是程序设计语言中的一种高级语言，下面就以一个简单的 C 语言程序为例来了解用 C 语言编写程序的基本知识。

1.1 一个简单的 C 语言程序

1. 程序题目

编写程序，在显示器上输出文字信息，如“My first C program!”。

2. 编程分析

程序编写的目标是显示文字信息，这就需要代码执行数据输出的功能。

3. 源程序代码

```
#include<stdio.h>
void main()
{
    printf(" My first C program! \n");
}
```

4. 程序运行结果

My first C program!

5. 知识点

(1) 文件包含预处理命令 include

程序第一行以“#”开头的命令是 C 语言的预处理命令，预处理命令还有其他几种，这里的 include 称为文件包含命令，其意义是把尖括号 (<>) 或引号 (") 内指定的文件包含到本程序中，使其成为本程序的一部分。被包含的文件通常是由系统提供的，其扩展名为.h，称为头文件或首部文件。C 语言的头文件中包括了各个标准库函数的函数原型。因此，凡是在程序中调用一个库函数时，都必须包含该函数原型所在的头文件。在本例中，使用了库函数 printf()，这是一个标准输出库函数，其头文件为 stdio.h。因此，在主函数前使用 include 命令包含了 stdio.h 文件。

(2) 主函数 main()

main 是主函数的函数名，表示这是一个主函数。每个 C 源程序都必须有且只能有一个主函数。

(3) 标准输出库函数 printf()

库函数是由 C 系统提供的一些基本的、常用功能模块的集合，也称为系统函数。

printf() 函数是一个由系统定义的标准函数，可在程序中直接调用，它的功能是把要输出的内容按照指定的格式显示。

(4) 注释语句

1) 程序文件中写在 /*.....*/ 内，或 // 后面的是附加解释语句，即 C 语言中的注释语句。

2) Turbo C 系统下注释语句只能用 /*.....*/ 标明，Visual C 系统下两种方式皆可。

3) 注释语句不是程序的可执行语句，只起到说明作用，是为了让用户理解程序而书写的，可写在程序的任意位置。

4) 在上机调试程序时，不必输入注释语句。

5) 本示例添加注释语句后的源程序如下：

```
#include<stdio.h> /* 文件包含命令,使用库函数时必须写上*/
void main()        //main为主函数名
{
    //函数名的下一行必须写"{",表示该函数具体语句(函数体)的开始
    printf("My first C program!\n");//printf函数实现信息输出
}                  //函数的最后必须写"}",与开始的 "{" 相对应
```

(5) C 程序的书写规则

1) 程序是由语句组成的，每条语句的最后必有一个分号 (;) 表示语句的结束。

2) 区分大小写字母，程序主要用小写字母书写，其中系统关键词（共 32 个，如 void）、系统函数名（如 printf）等必须用小写字母书写。

3) 所有符号必须是半角（英文）符号。

4) 一行可以写若干条语句，但为了查错方便，一般每行只写一条语句。

5) 由于C语言对语法限制不太严格,为了保证程序的准确性和可读性,各行语句在书写时按结构采用阶梯缩进对齐,可以使用【Tab】键实现缩进,如以下格式所示。

```

* * * * * * ( )
{
    * * * * * * * * * * ;
    * * * * * * * * * * ;

    {
        * * * * * * ;
        ...
    }
    * * * * * * * * * * ;
}

```

(6) C程序的基本符号

1) 英文字母: 小写字母 a~z 共 26 个, 大写字母 A~Z 共 26 个。

2) 数字: 0~9 共 10 个。

3) 标点和特殊字符: 如 “,” “;” “'” “{” “}” “(” “)” “'” 等。

4) 运算符: 如 “=” “+” “*” “>” “<” “&&” 等。

5) 空白符: 空格符、制表符按【Tab】键产生、换行符等统称为空白符。空白符只在字符常量和字符串常量中起作用。在其他地方出现时, 只起间隔作用, 编译程序将忽略它们。因此在程序中使用空白符与否, 对程序的编译不发生影响, 但在程序中适当的地方使用空白符将提高程序的清晰性和可读性。

(7) C程序的主要结构特点

1) C 程序是由一个或多个函数组成的, 函数的内容称为函数体, 由一对大括号 “{ }” 括起来。

2) 主函数 main() 是由系统提供的, 一个程序有且仅有一个主函数 main()。

3) C 程序总是从主函数 main() 开始执行, main() 函数可以放在程序的任何位置。

4) 在 C 程序中可以使用注释语句, 其不影响程序的编译和运行, 程序编译时会忽略这些内容。



练习题 1.1

一、单项选择题

1. 一个C语言程序的基本构成是 ()。

A. 若干过程

B. 函数

C. 若干子程序

D. 一个主程序和若干子程序

2. 一个C语言语句必须使用的结束符号是 ()。

A. ,

B. .

C. ;

D. ;

3. C 程序中 `main()` 函数的位置是 ()。
- A. 必须出现在固定位置 B. 必须出现在所有函数之前
C. 必须出现在所有函数之后 D. 可以在任何地方出现
4. 在 C 程序中, 注释部分以符号 `/*` 开始, 结束的符号是 ()。
- A. `//` B. `*/` C. `}` D. `)`
5. C 语言程序的注释可以出现在程序中的位置是 ()。
- A. 在任何允许插入空格符的地方插入
B. 开始处
C. 结束处
D. `main()` 函数中

二、程序改错题

指出下面程序中的错误, 并改正。

```
/* 程序的功能是用*输出字母 c 的图案 */
#include "stdio.h"
void main
PRINTF("Hello C-world!\n")      PRINTF(" ****\n")
PRINTF(" * \n")      PRINTF(" * \n")
PRINTF(" ****\n")
```

1.2 C 语言程序开发环境

Visual C++ 是微软公司的重要产品——Visual Studio 工具集的重要组成部分。Visual C++ 提供了一个集源程序编辑、代码编译和调试于一体的可视化开发环境, 即集成开发环境 (Integrated Development Environment, IDE)。Visual C++ 包含文本编辑器、资源编辑器、工程编译工具、源代码浏览器、集成调试工具, 以及一套联机文档。开发环境是程序员同 Visual C++ 交互的界面, 通过该环境, 程序员可以完成创建、调试、修改应用程序等的各种操作。程序人员对集成开发环境的熟悉程度直接影响程序设计的效率。集成开发环境采用标准的多窗口 Windows 用户界面, 并增加了一些新特性, 使得开发环境更易于使用, 用户能够很容易地学会并掌握 Visual C++ 的使用方法。

Visual C++ 6.0 是微软公司推出的 Visual C++ 开发工具的 6.0 版本, 它不仅支持 Internet 特性, 还增加了 ActiveX 数据对象 (ActiveX Data Objects, ADO) 数据绑定、活动模板库 (Active Template Library, ATL) 复合控件、编辑等, 开发者可以充分利用 Visual C++ 6.0 的这些特性, 构建自己的 Windows 应用程序。

Visual C++ 是一个可视化的开发工具, 在介绍 Visual C++ 6.0 的各个组成部分之前, 首先了解一下可视化编程的概念。可视化技术是当前发展迅速并引人注目的技术之一,

它的特点是把原来抽象的数字、表格、功能逻辑等用直观的图形、图像形式表现出来。可视化编程，即可视化程序设计，其以“所见即所得”的编程思想为原则，力图实现编程工作的可视化，即随时可以看到结果，程序与结果的调整同步。可视化编程是与传统的编程方式相比而言的，这里的“可视”，指的是无须编程，仅通过直观的操作方式即可完成界面的设计工作，是目前最好的 Windows 应用程序开发方式。这种可视化的编程方法不仅易学、易用，而且大大提高了工作效率。在 Visual C++ 6.0 中提供了大量的用于可视化编程的资源编辑工具。

1.2.1 Visual C++ 6.0 集成开发环境的启动

在计算机上安装 Visual C++ 6.0 应用程序后，可以选择以下两种方式启动。

1) 打开“开始”菜单，选择“程序”→“Microsoft Visual Studio 6.0”→“Microsoft Visual C++ 6.0”命令，启动 Visual C++ 6.0。

2) 打开“开始”菜单，选择“运行”命令，输入“msdev”后单击“确定”按钮，即可启动 Visual C++ 6.0，启动后的界面如图 1-1 所示。

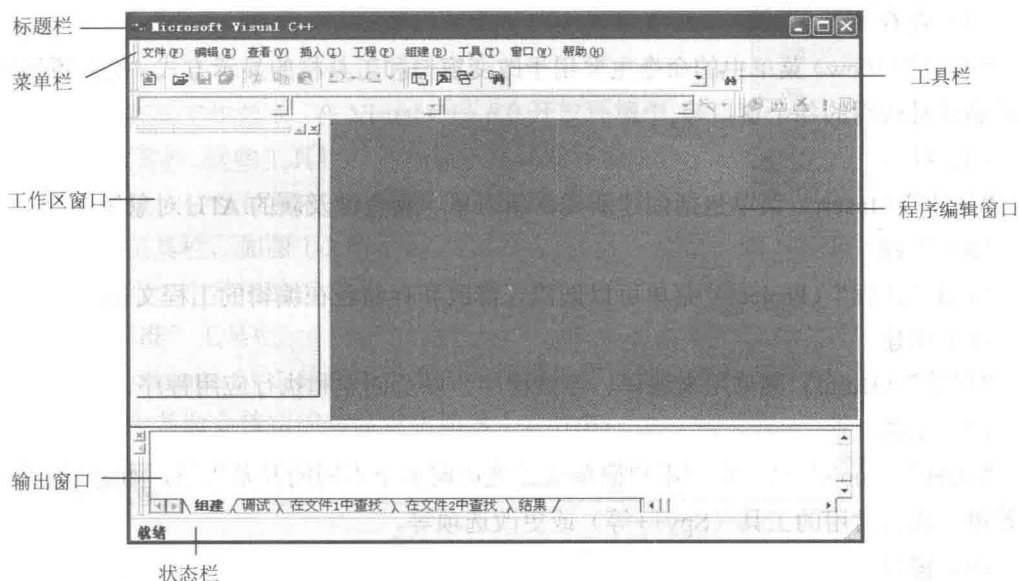


图 1-1 Visual C++ 6.0 的启动界面

Visual C++ 6.0 是一个 Windows 应用程序，启动后显示一个标准的窗口，由标题栏、菜单栏、工具栏、工作区窗口、程序编辑窗口、输出窗口和状态栏组成。

主窗口的左侧是工作区窗口，右侧是程序编辑窗口，下方是输出窗口。工作区窗口显示所设定的工作区的信息，程序编辑窗口用来输入和编辑源程序，输出窗口用来显示程序出错信息和结果有无错误（Errors）或警告（Warnings）。

1.2.2 Visual C++ 6.0 集成开发环境的使用

1. 菜单栏

在 Visual C++ 6.0 集成开发环境中，菜单栏位于标题栏下方，由 9 个菜单项组成。单击菜单项弹出下拉式菜单，可使用这些菜单项实现集成环境的各种功能。菜单项及其功能介绍如下。

(1) 文件

“文件”(File) 菜单包括对文件、项目、工作区及文档进行文件操作的相关命令或子菜单。

(2) 编辑

“编辑”(Compile) 菜单不仅包括常用的剪切、复制、粘贴命令，还包括为调试程序设置断点的命令，可实现设置、删除、查看断点的操作。此外，还有为方便程序员输入源代码的成员列表(List Members)、类型信息(Type Info)等命令。

(3) 查看

“查看”(View) 菜单中的命令主要用于改变窗口和工具栏的显示方式、检查源代码、激活调试时所用的各个窗口等。

(4) 插入

“插入”(Insert) 菜单包括创建新类、新表单、新资源及新的 ATL 对象等命令。

(5) 工程

使用“工程”(Project) 菜单可以创建、修改和存储正在编辑的工程文件。

(6) 组建

“组建”(Build) 菜单用来编译、连接程序，以及创建和执行应用程序。

(7) 工具

“工具”(Tools) 菜单允许用户简单快速地访问多个不同的开发工具，如定制工具栏与菜单、激活常用的工具(Spy++等)或更改选项等。

(8) 窗口

“窗口”(Window) 菜单中的命令用来进行有关窗口的操作，常用的命令有新建、分割、组合、关闭和全部关闭等。

(9) 帮助

“帮助”(Help) 菜单中的命令为用户提供了大量的帮助信息。

集成开发环境还为一些常用的命令分配了默认的快捷键，记住这些常用的快捷键，用户就可以更高效地使用集成开发环境。其中常用的快捷键如表 1-1 所示。

表 1-1 常用的快捷键

快捷键	功能	快捷键	功能
Ctrl+O	打开文件	F4	下一条错误
Ctrl+S	保存文件	Shift+F4	上一条错误
Ctrl+C	编辑复制	Ctrl+F6	下一个窗口
Ctrl+X	编辑剪切	Alt+Enter	对象属性
Ctrl+V	编辑粘贴	Ctrl+F7	编译文件
Ctrl+F	编辑查找	F7	建立可执行文件
Ctrl+Z	编辑取消	Ctrl+F5	运行

2. 工具栏

工具栏位于菜单栏的下方，由若干个命令按钮组成，单击某个按钮就可实现某种操作功能。工具栏也是 IDE 的重要组成部分，主要列出了在代码开发过程中经常用到的一些功能，具有直观和快捷的特点，熟练使用这些按钮将大大提高工作效率。在 Visual C++ 6.0 开发环境中包括 10 个工具栏，这些工具栏并不都显示在开发环境中，可以在工具栏上任意位置右击，在弹出的快捷菜单中选择要显示的工具栏，如图 1-2 所示。



图 1-2 工具栏快捷菜单

下面介绍 3 个常用的工具栏。

(1) “标准”工具栏

“标准”(Standard)工具栏用于维护和编辑工作区的文本和文件，如图 1-3 所示。“标准”工具栏中各命令按钮的功能说明如表 1-2 所示。



图 1-3 “标准”工具栏

表 1-2 “标准”工具栏中各命令按钮的功能说明

命令按钮	功能说明
	创建一个新的文件、项目和工作区
	打开一个已存在的文件、项目和工作区
	保存当前打开的文件
	保存所有打开的文件

续表

命令按钮	功能说明
	将所选择的内容剪切掉，移到剪贴板中
	将所选内容复制到剪贴板中
	在当前位置粘贴剪贴板中最新的内容
	撤销上一次的操作
	恢复被撤销的操作
	激活工作区窗口，用来管理工程中的文件和资源
	激活输出窗口，用来显示编译、调试和查找的信息
	管理当前打开的窗口
	在所有窗口中查找指定字符串
	在当前窗口中查找指定字符串
	调用 MSDN



图 1-4 “编译微型条”工具栏

(2) “编译微型条”工具栏
“编译微型条”(Build MiniBar)工具栏用于运行程序和调试程序，如图 1-4 所示。“编译微型条”工具栏中各命令按钮的功能说明如表 1-3 所示。

表 1-3 “编译微型条”工具栏中各命令按钮的功能说明

命令按钮	功能说明
	用于编译当前在程序编辑窗口中的源文件
	用于编译、连接当前工程中的文件，生成一个可执行文件
	终止编译或连接的程序
	运行程序
	开始或继续调试程序
	编辑程序中的断点

(3) “调试”工具栏

“调试”(Debug)工具栏用于调试程序，如图 1-5 所示。“调试”工具栏中各命令按钮的功能说明如表 1-4 所示。

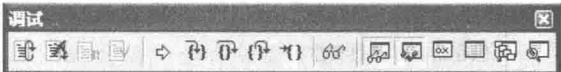


图 1-5 “调试”工具栏

表 1-4 “调试” 工具栏中各命令按钮的功能说明

命令按钮	功能说明
	放弃当前的调试，重新开始调试
	终止调试，返回编辑状态
	暂停调试
	改编代码后调试
	显示将要运行的代码行
	单步执行程序，进入函数内部
	单步执行程序，不进入函数内部
	跳出当前函数
	运行到光标
	弹出“Quick Watch”对话框
	显示/隐藏“Watch”对话框
	显示/隐藏“Variables”对话框
	显示/隐藏“Registers”对话框
	显示/隐藏“Memory”对话框
	显示/隐藏“Call Stack”对话框
	显示/隐藏“Disassembly”对话框

在图 1-2 所示的工具栏快捷菜单中选择“自定义”命令，弹出“定制”对话框，如图 1-6 所示。用户可以根据实际需求定制工具栏。



图 1-6 “定制”对话框

3. 项目和项目工作区

Visual C++应用程序的核心是工程（Project，也称项目），通常位于工程工作区（Workspace）中，Visual C++工程工作区可以容纳多个项目。

(1) 工程

在 Visual C++集成开发环境中，可以通过选择“文件”→“新建”命令，弹出“新建”对话框来创建一个新的工程。这个工程可以是一个游戏程序，或者是其他应用程序。工程名是工程中其他文件命名的基础，其扩展名为.dsp（Developer Studio Project）。

“新建”对话框的“工程”选项卡如图 1-7 所示。

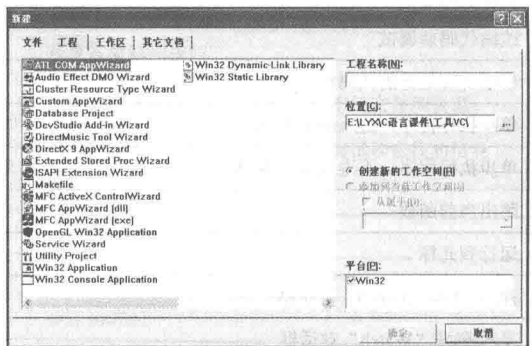


图 1-7 “新建”对话框的“工程”选项卡

“工程”选项卡中可供选择的部分工程的类型及含义如表 1-5 所示。

表 1-5 部分工程的类型及含义

类型	含义
ATL COM AppWizard	ATL 应用程序创建向导
Cluster Resource Type Wizard	簇资源类型创建向导
Custom AppWizard	自定义的应用程序创建向导
DataBase Project	数据库工程
DevStudio Add-in Wizard	插件创建向导
Extended Stored Proc Wizard	扩展存储编程创建向导
ISAPI Extension Wizard	扩展创建向导
Makefile	C/C++生成文件
MFC ActiveX ContorlWizard	MFC ActiveX 控制程序创建向导
MFC AppWizard [.exe]	MFC 可执行程序创建向导
MFC AppWizard [.dll]	MFC 动态链接库创建向导
Utility Project	单元工程
Win32 Application	Win32 应用程序
Win32 Console Application	Win32 控制台应用程序
Win32 Dynamic-Link Library	Win32 动态链接库
Win32 Static Library	Win32 静态库

需要注意的是，要建立一个扩展名是.c 的程序文件，在“工程”选项卡中要选择“Win32 Console Application”。“Win32 Console Application”的入口函数是 main()函数。