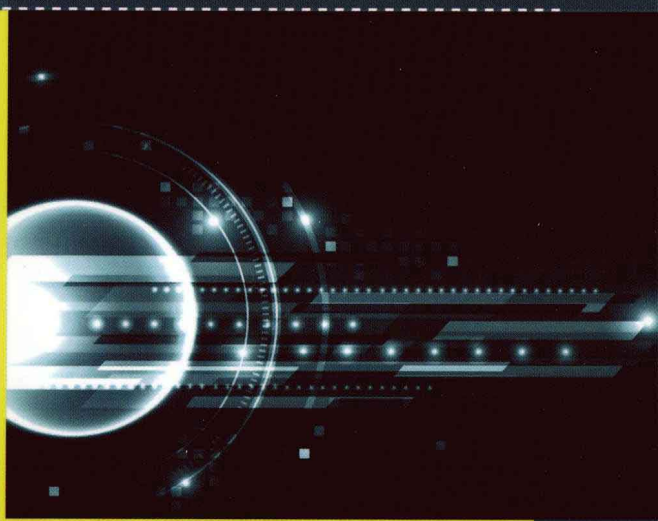




21世纪高等学校计算机科学与技术规划教材



C语言 程序设计实践教程

主 编 刘燕君 刘振安

C YUYAN CHENGXU
SHEJI SHIJIAN JIAOCHENG



北京邮电大学出版社
www.buptpress.com



21 世纪高等学校计算机专业系列教材

C 语言程序设计实践教程

主 编 刘燕君 刘振安

北京邮电大学出版社

· 北京 ·

内 容 简 介

本书是《C 语言程序设计教程》一书的配套教材,但又自成体系,可以单独使用。本书重点介绍如何使用目前流行的 Visual C++ 6.0 集成开发环境编辑、编译、调试和运行 C 语言程序的具体方法,帮助读者尽快熟练地掌握集成开发环境,使用集成环境加强编程实践并帮助学习和加深理论知识。

本书章节与主教材相对应,每章均首先介绍该章的重点和难点,然后对习题进行解答,最后给出教材中实验题的解题思路和参考程序。为了保证程序的结构化设计质量,尤其是大程序的设计方法,本书单列一章讨论程序的设计问题。

本书重在培养学生的应用技能,特别适合作为高等院校教材及工程技术人员的参考书。

图书在版编目(CIP)数据

C 语言程序设计实践教程/刘燕君,刘振安主编.--北京:北京邮电大学出版社,2012.1

ISBN 978-7-5635-2746-5

I. ①C… II. ①刘… ②刘… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 190313 号

书 名	C 语言程序设计实践教程
主 编	刘燕君 刘振安
责任编辑	向 蕾
出版发行	北京邮电大学出版社
社 址	北京市海淀区西土城路 10 号(100876)
电话传真	010-82333010 62282185(发行部) 010-82333009 62283578(传真)
网 址	www.buptpress3.com
电子信箱	ctrd@buptpress.com
经 销	各地新华书店
印 刷	北京联兴华印刷厂
开 本	787 mm×1 092 mm 1/16
印 张	16
字 数	397 千字
版 次	2012 年 1 月第 1 版 2012 年 1 月第 1 次印刷

ISBN 978-7-5635-2746-5

定价: 28.00 元

如有质量问题请与发行部联系

版权所有 侵权必究

前 言

本书的编写以培养学生 C 语言编程能力为主线,意在通过实践环节加强学生实际动手能力。本教材的章节与主教材相对应,主要特点如下。

(1)每章均首先介绍该章的重点和难点,有的还结合典型例题进行解析。

(2)对主教材每章的习题进行解答,有的习题给出多种解法以便给读者更多的启发。

(3)详细给出主教材中实验题的解题思路和参考编程实现方法,并给出扩充的实践练习题以便读者自己练习。

(4)用趣味程序作为例题,既提高了学习兴趣,又使学生掌握了解决实际问题的能力。

(5)本书在第 7 章结合实验题,介绍头文件的编制、多个 C 语言文件及工程文件的编制等方法,以培养学生的实际应用能力。

(6)为了保证程序的结构化设计质量,尤其是大程序的设计方法,本书在第 10 章讨论了结构化程序设计方法并给出设计实例。本章的内容对学生做毕业论文有参考价值。

(7)提供全书的源程序以便学习。

由此可见,本书虽然密切配合主教材,能为讲授和学习主教材提供方便,但又可以单独使用,为同类教材配套,从而扩大它的适用范围。

本书由刘燕君、刘振安主持编写并担任主编,刘大路等参加了编写。中国科学技术大学软件学院院长陈国良院士,原安徽大学副校长、计算机系程慧霞教授及南京大学计算机系陈本林教授在百忙之中审阅了书稿并提出许多宝贵意见,许多使用过原教材的兄弟院校老师也提出很多宝贵意见,特此感谢。本书在写作过程中还参考了大量的资料,有的在参考文献中注明了,还有些没有注明,在此对这些资料的作者表示感谢。

由于写作时间仓促,本书不妥之处在所难免,敬请各位读者不吝赐教,给予指正为盼。作者邮箱:zaliu@ustc.edu.cn。

编 者

目 录

第 1 章 C 程序设计基础	1
1.1 重点	1
1.2 难点	2
1.3 习题参考答案	3
1.4 实验——练习使用集成环境编写程序	6
1.5 实践练习	10
第 2 章 基本数据类型和表达式	11
2.1 重点	11
2.1.1 标识符和关键字	11
2.1.2 基本数据类型	13
2.1.3 运算符	15
2.2 难点	16
2.2.1 变量的属性	16
2.2.2 输入输出语句	18
2.2.3 逗号运算符与逗号表达式	19
2.3 习题参考答案	21
2.4 实验 2——编辑、编译、调试和运行一个实际程序	26
2.5 实践练习	29
第 3 章 C 语言的控制结构	30
3.1 重点	30
3.1.1 控制语句	30
3.1.2 鸡兔同笼	31
3.1.3 求解百鸡问题	33
3.1.4 简单的菜单程序	35
3.2 难点	36
3.3 习题参考答案	38
3.4 实验 3——通过调试改正程序中的错误	55
3.4.1 实验题 1——求最大值	55
3.4.2 实验题 2——体育赛程编排	57
3.5 实践练习	60

第 4 章 函数与变量类型	61
4.1 重点	61
4.1.1 变量的作用域	61
4.1.2 函数的返回值	65
4.1.3 一个源文件和一个头文件的典型结构	68
4.2 难点	70
4.3 习题参考答案	72
4.4 实验 4——编辑含有多个文件的函数调用程序	83
4.4.1 实验题目和要求	83
4.4.2 实验参考步骤	83
4.5 实践练习	86
第 5 章 指针基础	87
5.1 重点	87
5.1.1 指针运算	87
5.1.2 动态内存分配	89
5.1.3 指针作为函数的参数	94
5.2 难点	95
5.2.1 返回指针的函数	95
5.2.2 指针初始化	96
5.2.3 void 指针	98
5.3 习题参考答案	99
5.4 实验 5——使用指针	105
5.4.1 实验题 1——熟悉使用指针和动态内存分配	105
5.4.2 实验题 2——出圈游戏	107
5.5 实践练习	110
第 6 章 一维数组和指针	111
6.1 重点	111
6.1.1 数组名字就是数组的首地址	111
6.1.2 数组名与指针的相互关系	112
6.1.3 数组作为函数参数	113
6.2 难点	115
6.2.1 再谈变量的值	116
6.2.2 二分查找	116
6.2.3 冒泡排序	118
6.3 典型例题和习题	120
6.4 实验 6——使用数组和指针	129
6.4.1 实验题 1——一维数组的出圈游戏	129

6.4.2 实验题2——熟悉函数和指针	131
6.5 实践练习	132
第7章 函数和多维数组	135
7.1 重点	135
7.1.1 传递函数的参数	135
7.1.2 函数返回值	137
7.1.3 二维数组和指针	143
7.1.4 字符串数组	145
7.2 难点	147
7.2.1 常量指针	147
7.2.2 指向常量的常量指针	148
7.2.3 多维数组和指针	148
7.2.4 函数指针	148
7.3 习题参考答案	153
7.4 实验7——使用二维字符串数组设计出圈程序	166
7.5 多文件编程实验	168
7.5.1 设计与算法分析	168
7.5.2 参考程序	169
7.5.3 组成一个工程	173
7.6 实践练习	174
第8章 结构类型	177
8.1 重点	177
8.1.1 结构变量	177
8.1.2 结构作为函数参数	179
8.2 难点	182
8.2.1 枚举	182
8.2.2 联合	183
8.2.3 链表	186
8.3 习题参考答案	187
8.4 实验8——使用结构指针数组	199
8.4.1 实验题1——使用结构和动态内存分配设计出圈程序	199
8.4.2 实验题2——按给定的数据结构编程	203
8.5 实践练习	207
第9章 文件	208
9.1 重点	208
9.2 难点	209
9.3 习题参考答案	214

9.4	实验 9——在函数里使用文件	216
9.5	实践练习	218
第 10 章	C 程序结构化设计	221
10.1	重点	221
10.1.1	工程文件组织	221
10.1.2	正确设计函数	223
10.2	难点	223
10.3	实验 10——扩充完善学生成绩管理程序	224
10.3.1	实验题目和要求	224
10.3.2	改进措施	224
10.3.3	参考程序	225
10.3.4	测试	245
主要参考文献	247

第 1 章 C 程序设计基础

本章主要要求学生熟悉如何编辑、编译、调试和运行一个实际的程序,以便为以后的边学边练打下良好的基础。本章结合一个实际的例子,演示 C 语言编程的全过程。

1.1 重 点

1. 掌握编写程序的全过程

重点是要熟悉如何编辑、编译、调试和运行一个实际的程序,以便为下一步的学习打下基础。

著名计算机科学家沃思提出的“数据结构+算法=程序”思想,大大促进了程序设计的发展。由于软件产业迅速发展,产生了“软件工程”新学科,人们认识到“程序设计方法”的重要性。编程是算法的具体实现,所以程序离不开编程。过去的编译环境简单,都是命令行方式,很容易学习。目前的编译环境都朝集成环境和可视化编程方向发展,而且还实现了一定的自动化功能。由于编译环境功能强大,也使其具有一定的复杂性。典型的如 Visual C++ 6.0 (以下简称 VC),它提供了 MFC 类库,实现许多 Windows 自动编程功能。因此,如何用好这些语言工具和环境,也是提高编程效率的因素之一。可以用下面的公式来表示程序。

数据结构+算法+程序设计方法+编程工具=程序

由此可见,一个程序设计人员应该掌握以上 4 个方面的知识。人们一般容易忽视的是编程工具的使用。有些人能写出正确的程序,却无法使用编程工具产生正确的可执行文件。为了消除这种现象,本书将结合具体实验题加强训练,使学生既学会“做什么”,也掌握“如何做”的全过程。从而避免可以在纸上编程序,却不知道如何得到程序的可执行文件的现象。

2. 掌握 C 程序的基本结构模式

学习 C 语言,首先要注意掌握如下只有主函数的简单 C 程序的基本结构模式。

```
文件包含部分      //包含头文件等
int main()         //主程序
{                  //主程序开始
    声明语句部分    //声明程序中所要使用的变量
    执行语句部分    //程序的可执行语句
    return 0;      //返回值
}                  //主程序结束
```

声明语句部分目前只涉及变量,不管以后章节增加何种声明,这些声明必需都放在可执行语句之前,即所有声明放在一起。只要牢牢掌握并熟练使用这种结构,就能解决很多程序设计问题。

3. 掌握输出输入语句的使用

输出输入语句是构成 C 语言程序的基本组成部分,必须熟练掌握它们。注意,scanf 语句格式的要求是变量的地址,所以在变量前面应加上“&”。

4. 正确理解表达式和语句的区别

重点要理解表达式和语句的区别。例如

```
x=x+1
```

是一个表达式,而不是语句。只有在最后面以“;”号结束,才成为一条语句。即

```
x=x+1;
```

是一条语句。一个分号就构成了一个程序语句。一个程序语句最后必须出现分号,分号是程序语句中不可缺少的一部分。

5. 项目管理

VC 是以项目的形式进行管理的。它为每个项目建立一个与项目名称同名的目录,所有的文件均建立在这个项目目录之下。虽然可以在建立过程中撤销项目,但这个目录已经建立,重新建立目录时,不能再使用相同的名称。在同一个工作目录之下,项目不能同名。要想继续使用这个名称,必须先将同名目录删除。

1.2 难 点

难点是初步认识不同编译器的区别,以便在编程中回避依赖编译器的问题。其实,只要避免使用可能产生歧义的语句,就可以编写出正确的程序。

因为学生可能是第一次接触编程语言的学习,所以首先应要求他们树立良好的书写习惯和风格,尤其是变量命名的良好习惯。学生的模仿能力很强,可以适当给出一些范例及提示,让他们去举一反三,融会贯通。

【例 1.1】 给出下面程序的输出结果。

```
#include <stdio.h>
#include <string.h>
void main()
{
    int a, b, c, d;

    a=sizeof(char);
    b=sizeof(int);
    c=sizeof(float);
    d=sizeof(double);

    printf("\n%d %d %d %d",a,b,c,d);
}
```

【分析】输出结果取决于编译器和硬件组成的方式。

由于计算机硬件发展很快,所以 C 编译器也随之更新。本书指的主要是 16 位操作系统到 32 位操作系统的变化。以 Borland C++ 3.1(本书以下简称 BC)为例,它是为 16 位操作系统配置的,所以即使在 Windows 98 下运行,其运行结果也是:1 2 4 8。

自从 32 位操作系统盛行以来,编译器也发生了变化。如 VC 的运行结果是:1 4 4 8。

这个例子提醒学生们注意,对很多 C 语言书上的结果,自己要加以分析。在第 2 章将进一步讨论这个问题。

1.3 习题参考答案

1. 单项选择题

(1)注释语句正确的格式是_____。

- A. * // B. // * C. // // D. / * * /

【解答】 D

(2)下面不是 printf 的格式字符串的是_____。

- A. s B. d C. k D. c

【解答】 C

(3)C 语言的最小程序结构所包含的函数数量是_____。

- A. 1 个 B. 2 个 C. 3 个 D. 4 个

【解答】 A

(4)C 程序的主函数是_____。

- A. printf B. scanf C. #include D. main

【解答】 D

2. 填空题

(1)空语句是_____。

【解答】 ;

(2)C 程序是由_____组成,其中只能有一个_____函数。

【解答】 函数 主

(3)用来描述语句的执行条件与顺序的语句称为_____语句。

【解答】 程序的控制

(4)语句的终结符是_____。

【解答】 ;

(5)每个引用库函数的源程序文件的开头处都必须含有_____。

【解答】 #include <stdio. h>

(6)VC 提供的编制 DOS 程序的方法是_____程序。

【解答】 控制台程序,或者填 Win32 Console Application

(7)用 scanf 输入整数变量 a 的语句是_____。

【解答】 scanf("%d",&a);

(8)C 程序可以有_____个主函数。

【解答】 1

(9) C 程序必须从_____处开始执行,_____处结束运行。

【解答】 { }

(10) 从键盘输入一个字符给字符变量 a 的 scanf 语句格式为_____。

【解答】 scanf("%c",&a);

(11) 从键盘输入数字“235”给整型变量 a 的 scanf 语句格式为_____。

【解答】 scanf("%d",&a);

(12) 输出“235”的 printf 语句格式为_____。

【解答】 printf("235");

3. 程序设计题

(1) 编写一个计算 $25.30+43$ 的程序。

【解答】 直接使用 printf 语句输出计算结果即可。

```
#include<stdio.h>           //包含头文件
int main()
{
    printf("25.30+43=%f\n",25.30+43); //输出结果
    return 0;
}
```

(2) 编写一个计算 $y=2x+8$ 的程序,为程序加上注释并运行。

【解答】 要求询问输入数据,所以需要使用 scanf 语句接受输入。

```
#include<stdio.h>           //包含头文件
int main()
{
    float x,y;
    printf("请输入 x:");      //请求输入
    scanf("%f",&x);          //输入 x
    y=2 * x+5;
    printf("y=2x+5=%f\n",y);  //输出结果
    return 0;
}
```

运行示例如下:

```
请输入 x:25.5
y=2x+5=56.000000
```

注意:凡键盘输入的信息均用下划线标出。输入以回车键结束,但书中不给出回车符号。其实,也可以直接使用 printf 语句输出计算结果。

```
#include<stdio.h>           //包含头文件
int main()
{
    float x;
    printf("请输入 x:");      //请求输入
```

```

scanf("%f",&x);           //输入 x
printf("y=2x+5=%f\n",2 * x+5); //输出结果
return 0;
}

```

(3)编写一个输出“25+75=100”的程序。

【解答】 直接使用 printf 语句即可。

```

#include<stdio.h>           //包含头文件
int main()
{
    printf("25+75=100\n"); //输出结果
    return 0;
}

```

(4)编写一个对话程序,例如:

```

李明:HOW ARE YOU?
王莹:FINE, THANK YOU, AND YOU ?

```

【解答】 因为还没有学过字符数组,显然只是要求输出这段对话。直接使用 printf 语句即可。

```

#include<stdio.h>
int main()
{
    printf("李明:HOW ARE YOU? \n");
    printf("王莹:FINE, THANK YOU, AND YOU ? \n");
    return 0;
}

```

(5)编写不使用变量计算 $55 * 11$ 的程序。

【解答】 直接使用 printf 语句即可。

```

#include<stdio.h>           //包含头文件
int main()
{
    printf("55 * 11=%d\n",55 * 11); //输出结果
    return 0;
}

```

(6)编写要求输入变量 a 和 b 的值,输出它们的和及乘积的程序。

【解答】 要求询问输入数据,所以需要使用 scanf 语句接受输入。

```

#include<stdio.h>           //包含头文件
int main()
{
    float a,b;
    printf("请输入 a:"); //请求输入 a
}

```

```

scanf("%f",&a);
printf("请输入 b:");           //请求输入 b
scanf("%f",&b);
printf("%f+%f=%f\n",a,b,a+b); //输出其和
printf("%f*%f=%f\n",a,b,a*b); //输出乘积
return 0;
}

```

运行示例:

```

请输入 a:25
请输入 b:25
25.000000+25.000000=50.000000
25.000000*25.000000=625.000000

```

(7)编写从键盘输入正方形的边长,输出正方形面积和周长的程序。

【解答】

```

#include <stdio.h>
int main()
{
    int a, S, L           //声明 3 个整型变量
    printf("Please input side :\n");
    scanf("%d",&a);
    S=a*a;               //计算面积
    L=4*a                //计算周长
    printf("面积=%d, 周长=%d", S,L);
    return 0;
}

```

1.4 实验——练习使用集成环境编写程序

1. 实验题目

编写将输入的两个整数相加然后输出它们的和的程序,然后运行。

2. 要求

- (1)使用 VC 集成环境练习建立项目和文件。
- (2)练习编写、编译和运行本程序。

【解答】

1. 产生工程项目

启动并进入 VC 开发环境。利用 VC 提供的向导工具,制作 C 程序源文件,具体方法如下。

(1)选中“File”菜单的“New”命令,弹出如图 1.1 所示的“New”对话框,并自动选中“Projects”选项卡。假设目前设置工作目录是在 E 盘的“tvc\qc”下面,则在“Location”列表框中显示当前工作目录“E:\TVC\QC”。

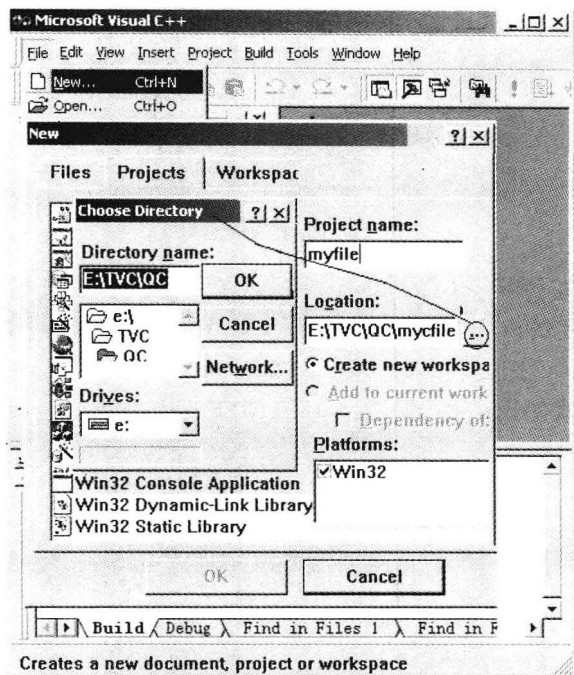


图 1.1 “New”对话框“Projects”选项卡内容示意图

(2)在图 1.1 的“Projects”选项卡栏中选择“Win32 Console Application”项,在右边的“Project name”输入框中输入项目名称“myfile”。系统自动把这个项目置于原设置工作目录 E 盘的 tvc\qc 下面,在“Location”列表框中自动加上项目文件目录,变为“E:\TVC\QC\myfile”。如果想改变工作目录,在该输入框中输入即可,也可以单击右边带 3 个点标记的按钮,在弹出的“Choose Directory”对话框中选择所要目录。

(3)选择完毕。单击“OK”按钮即可进入如图 1.2 所示的“Win32 Console Application”制作向导的第一步。

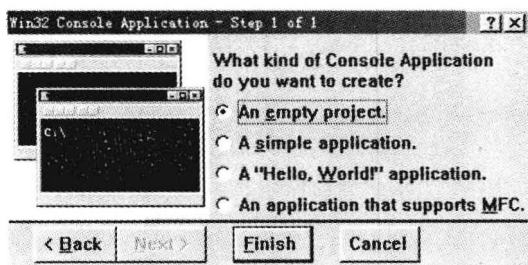


图 1.2 Win32 Console Application

(4)编辑 C 程序文件是选择“An empty project”项。由图可见,还可以使用 Back 按钮返回上一个对话框重新设置,也可以取消这次操作。单击“Finish”按钮,完成设置并弹出如图 1.3所示的信息框,可知在“E:\TVC\QC”目录下创建了一个空的项目“myfile”。这个项目已经具有目录“myfile”。

(5)如果满意,单击“OK”按钮确认,给出如图 1.4 所示的名为“myfile”的空项目示意图。如果不满意,单击“Cancel”按钮撤销以上的创建工作。

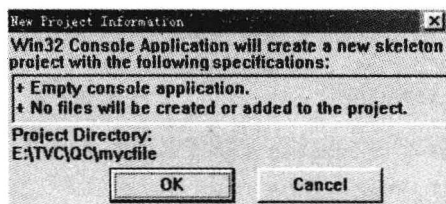


图 1.3 新项目信息框

(6)选中图 1.4 中的“FileView”，进入空项目。单击它后将展开树形结构。选中“mycfile files”结点，展开向导建立的内容。这时“Source Files”列表框里是空的，没有 C 程序的源文件名称。选中“Source Files”标记，再从“File”菜单中选“New”命令，又弹出如图 1.1 所示的“New”对话框。不过这次给出的是如图 1.5 所示的“Files”选项卡的内容。

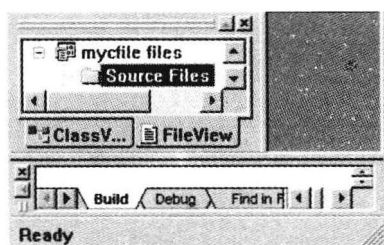


图 1.4 空项目示意图

(7)选中图 1.5 中的“C++ Source File”选项，在右方的“File”输入框中输入 C 程序文件名。这里输入的名字为“mycfile.c”。这里一定要加后缀“.c”，系统默认的后缀是“.cpp”，即 C++文件。单击“OK”按钮，返回集成环境。

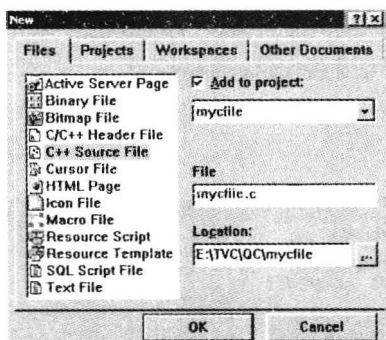


图 1.5 “New”对话框“File”选项卡内容示意图

2. 编写 C 的源程序

图 1.6 是在编辑框里编辑源程序 mycfile.c 的示意图。

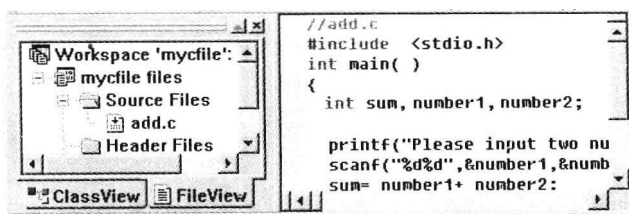


图 1.6 编辑 C 语言源程序的示意图

3. 编译并运行 C 程序

“Build”菜单主要包含有关编译、建立、执行和调试应用程序的命令选项。“Build”工具栏给出与“Build”菜单相对应的常用按钮。“Build”菜单如图 1.7 所示,部分菜单含义见表 1.1,“Build”工具栏如图 1.8 所示,描述见表 1.2。



图 1.7 “Build”菜单

表 1.1 部分“Build”菜单项描述

菜单项	描述
Compile	编译源代码窗口中的活动源文件
Build	查看工程中的所有文件,并对最近修改过的文件进行编译和链接
Rebuild All	对工程中的所有文件全部进行重新编译和链接
Clean	删除项目的中间文件和输出文件
Start Debug	选择该选项将弹出级连菜单,主要包含有关程序调试的选项
Execute	运行应用程序

工具栏对应某些菜单选项或命令,可以直接用鼠标单击工具栏按钮来完成相应的命令。菜单项的含义与一般的 Windows 软件的含义相同,不再赘述。这里仅介绍目前急需的编译和运行有关的命令。



图 1.8 “Build”工具栏

表 1.2 “Build”工具栏按钮描述

工具按钮图标	工具按钮名称	功能
	Compile	编译文件
	Build	建立工程
	Stop Build	停止建立工程
	Execute Program	执行应用程序
	Go	启动或继续程序的执行
	Insert/Remove Breakpoint	插入或删除断点