



普通高等教育“十二五”规划教材

程序设计基础

CHENGXU SHEJI JICHU

主编 王顺利



北京邮电大学出版社
www.buptpress.com



普通高等教育“十二五”规划教材

程序设计基础

主编 王顺利



北京邮电大学出版社
www.buptpress.com

内 容 简 介

本书主要是配合高等院校计算机程序设计基础课程教学,并参照《全国计算机等级考试二级C语言考试大纲》中“C语言程序设计基本要求”编写而成的。

本书以VC++语言集成环境为教学平台,以计算思维和程序设计思想培养为主线,结合应用型人才培养特点,主要介绍了C语言程序设计的基础知识和基本方法以及C++编程的初步知识。全书共分12章,主要内容包括C程序入门、C语言程序的三种基本结构(顺序结构、分支结构、循环结构)、数组与字符串、函数与指针、结构体与链表、其他数据类型、文件以及C++面向对象基础等。

本书“以实例为导向,以编程为驱动”,注重内容的实用性、针对性和实际可操作性,层次分明、概念清晰、通俗易懂,实例丰富,且每章均提供了大量的习题和典型的操作项目供读者巩固所学内容。

本书可作为应用型本科院校、高职高专、成人高校理工科各专业计算机程序设计课程的教材,也可作为全国计算机等级考试(二级)的培训教材和自学参考书。

图书在版编目(CIP)数据

程序设计基础 / 王顺利主编. -- 北京:北京邮电大学出版社, 2014.9

ISBN 978-5635-4100-3

I. ①程… II. ①王… 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(2014)第184711号

书 名: 程序设计基础

主 编: 王顺利

责任编辑: 刘春棠

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(邮编:100876)

发 行 部: 电话: 010-62282185 传真: 010-62283578

E-mail: publish@bupt.edu.cn

经 销: 各地新华书店

印 刷: 北京鑫丰华彩印有限公司

开 本: 787 mm×1 092 mm 1/16

印 张: 17.75

字 数: 440千字

版 次: 2014年9月第1版 2014年9月第1次印刷

ISBN 978-5635-4100-3

定 价: 36.00元

· 如有印装质量问题,请与北京邮电大学出版社发行部联系 ·

前 言

“程序设计基础”是高等院校理工科相关专业的一门重要的专业基础课程。C语言是目前国内外使用最广泛的一种计算机程序设计语言。它采用自顶向下、逐步求精的结构化程序设计技术,表达能力强,处理功能丰富,使用方便灵活,应用面广,执行程序效率高,可移植性好;同时具有高级语言和低级语言的许多优点,既适合编写系统软件,又适合编写应用软件。许多高等院校都选用C程序设计语言作为学生计算机编程的入门课程,其目的是培养学生的程序设计理念,掌握程序设计的基本方法,为后续的课程(数据结构、面向对象程序设计、网络编程、软件工程等)打下坚实的基础。由于其并不需要学生具有专门的计算机专业知识就可以学习,所以它不仅适合于计算机专业的学生,而且适合于非计算机专业的学生。

由于C程序设计语言的应用范围大,涉及的知识面广,规则较多,使用过于灵活,不易掌握,给学生的学习带来一定的困难。本书正是为很少甚至没有编程实际经验的大学生而编写的。

本书作为计算机程序设计技术基础入门教材,根据高等院校应用型人才培养的特点,以面向实际应用为目标,以计算思维和程序设计思想培养为主线,按照“以实例为导向,以编程为驱动”的原则编写教材。以VC++语言集成环境为教学平台,主要介绍了C语言程序设计的基础知识和基本方法以及C++编程的初步知识,注重基本知识和基本操作技能的学习和培养,按照知识层次递进的原则将C语言程序设计的基本内容分解为由简单到复杂的一系列学习模块,每个学习模块对应C程序设计的不同知识点,在学生完成各个学习模块的过程中,逐步引入C语言中的基本概念和语法,这样可以降低难度,有利于学生循序渐进地理解和掌握C语言程序设计方法,逐步提高编程应用能力,为学生的专业学习和今后的事业发展奠定良好的基础。

本书共12章。第1章为C程序入门,第2章、第3章、第4章介绍了C语言程序的三种基本结构:顺序结构、分支结构、循环结构,第5章讲述数组与字符串,第6章介绍函数,第7章介绍指针,第8章介绍结构体与链表,第9章讲述其他数据类型,第10章介绍文件,第11章、第12章介绍C++面向对象基础,最后是附录,涉及C语言的常用函数、关键字含义、运算优先级以及程序调试常见错误等内容。

本书中所举的实例是为了帮助读者更好地理解学习内容而精心编写的。这些程序从教学和学习者的角度出发,对问题作了简化,尽量压缩不必要的语句,篇幅一般不长,便于读者读懂。所有例题均在 Microsoft Visual C++ 6.0 环境下运行通过。

本书力争在便于学生学习的同时,兼顾教师教学的方便。为此,每章的前面均设置了“本章导读”和“本章要点”,以便让读者了解各章整体框架和学习要求,然后引出本章主题;后面提供了丰富的“思考和练习题”,这些内容的编排有助于读者复习巩固所学的知识,逐步掌握程 C 语言程序设计的功能,从而掌握程序设计的编程思想。

本书在编写过程中参考、综合了大量的相关资料,并结合我们多年的教学 and 实践经验,力求教学内容突出实用性、针对性和实际可操作性,组织结构层次分明、编程与理论结合紧密,叙述上概念清晰、通俗易懂,易教易学。

本书由王顺利主持编写。参加编写工作的老师有常鑫(第 1 章、第 9 章、附录 1、附录 2、附录 6)、杨瑞(第 2 章、第 4 章)、赵鹏宇(第 3 章、附录 3、附录 4、附录 5)、宋金平(第 5 章、第 12 章)、马晓娟(第 6 章)、任国英(第 7 章、第 11 章)、刘宝娥(第 8 章、第 10 章)。全书由王顺利统稿、定稿。

由于时间仓促,编者水平有限,书中不足之处在所难免,恳请广大读者批评指正。

编者

2014 年 1 月

目 录

第 1 章 C 程序入门	1
1.1 创建一个简单的 C 程序	1
1.1.1 实例解析	1
1.1.2 运行 C 程序的步骤和方法	3
1.2 C 程序概述	6
1.2.1 C 程序的基本组成	6
1.2.2 C 程序的输入和输出	7
1.2.3 C 程序的特点	7
1.2.4 C 程序的书写规范	8
1.3 程序与程序设计语言	8
1.3.1 程序与指令	8
1.3.2 程序设计语言	9
1.3.3 结构化程序设计的控制流程	10
1.4 程序算法简介	11
1.4.1 算法举例	11
1.4.2 算法应具备的特征	12
思考和练习题	12
第 2 章 顺序结构程序设计	14
2.1 设计一个简单的顺序结构程序	14
2.1.1 实例解析	14
2.1.2 常量与变量	15
2.1.3 基本数据类型	16
2.1.4 字符串常量	19
2.1.5 变量的初始化	20
2.2 表达式	20
2.2.1 基本运算符	21
2.2.2 运算符的优先级和结合性	22
2.3 数据类型的转换	23

2.3.1	各类数值型数据间的混合运算	23
2.3.2	赋值转换	23
2.3.3	强制转换	24
2.4	顺序结构程序设计	25
2.4.1	C 语句分类	25
2.4.2	顺序结构	27
	思考和练习题	27
第 3 章	分支结构程序设计	30
3.1	设计一个简单的分支结构程序	30
3.1.1	实例解析	30
3.1.2	关系运算	33
3.1.3	逻辑运算	34
3.2	分支语句	36
3.2.1	if 语句	36
3.2.2	switch 语句	38
3.3	分支结构控制语句的进一步解析	41
3.3.1	分支结构的嵌套	41
3.3.2	条件运算符和条件表达式	42
3.4	典型算法	42
	思考和练习题	47
第 4 章	循环结构程序设计	50
4.1	设计一个简单的循环结构程序	50
4.1.1	实例解析	50
4.1.2	自增自减运算符	53
4.2	三种循环语句	53
4.2.1	while 语句	53
4.2.2	do-while 语句	55
4.2.3	for 语句	58
4.3	循环程序的进一步解析	60
4.3.1	三种循环语句的比较	60
4.3.2	break 语句和 continue 语句	60
4.4	循环的嵌套	62
4.5	典型算法	66
	思考和练习题	70

第 5 章 数组与字符串	76
5.1 设计一个简单的数组程序	76
5.1.1 实例解析	76
5.1.2 数组概述	78
5.2 一维数组	79
5.2.1 一维数组的定义和初始化	79
5.2.2 一维数组的引用	80
5.3 二维数组	85
5.3.1 二维数组的定义和初始化	85
5.3.2 二维数组的引用	87
5.4 字符串与字符数组	89
5.4.1 字符串	89
5.4.2 字符数组的定义、初始化和引用	90
5.4.3 字符串的输入输出	90
5.4.4 字符串处理标准函数	93
5.5 典型算法	98
思考和练习题	103
第 6 章 函数	108
6.1 设计一个简单的用户自定义程序	108
6.1.1 实例解析	108
6.1.2 函数概述	110
6.1.3 函数的分类	111
6.2 预处理命令	112
6.2.1 宏定义	112
6.2.2 文件包含	116
6.2.3 条件编译	117
6.3 用户自定义函数	119
6.3.1 函数定义	119
6.3.2 函数调用	120
6.3.3 函数参数	124
6.3.4 函数的嵌套及递归调用	127
6.4 变量的作用域与生命期	131
6.4.1 变量的属性	131
6.4.2 局部变量和全局变量	131

6.4.3 变量的存储类型与生命期	134
6.5 典型算法	138
思考 and 练习题	141
第7章 指针	147
7.1 设计一个指针处理学生成绩的程序	147
7.1.1 实例解析	147
7.1.2 变量的直接访问和间接访问	148
7.2 指针	149
7.2.1 指针变量	149
7.2.2 指针运算	150
7.3 指针与数组	152
7.3.1 指向数组元素的指针	153
7.3.2 数组指针	153
7.3.3 多维数组的指针	155
7.3.4 字符串指针	157
7.3.5 指针数组	159
7.4 指针与函数	161
7.4.1 指针作为函数参数	161
7.4.2 指针型函数	164
7.4.3 指向函数的指针	165
7.5 指向指针的指针	166
思考 and 练习题	168
第8章 结构体与链表	171
8.1 设计一个简单的程序	171
8.1.1 实例解析	171
8.1.2 结构体的概念	173
8.2 结构体	174
8.2.1 结构体类型的声明	174
8.2.2 结构体变量的定义	174
8.2.3 结构体变量的引用	176
8.2.4 结构体变量的初始化	177
8.2.5 结构体数组	178
8.2.6 结构体与指针	180
8.2.7 结构体与函数	183

8.3 链表	184
8.3.1 链表的概念	184
8.3.2 动态申请和释放内存存储空间	185
8.3.3 链表的操作	186
思考和练习题	192
第9章 其他数据类型	199
9.1 设计一个简单的共用体程序	199
9.1.1 实例解析	199
9.1.2 共用体的概念	201
9.2 共用体	201
9.2.1 共用体的定义和引用	201
9.2.2 带共用体的结构体	202
9.3 枚举	204
9.3.1 枚举类型的声明	204
9.3.2 枚举变量的定义	204
9.4 用 typedef 命名已有类型	205
思考和练习题	206
第10章 文件	208
10.1 设计一个存储学生成绩信息的文件	208
10.1.1 实例解析	208
10.1.2 文件概述	210
10.2 文件的基本操作	211
10.2.1 文件类型指针	211
10.2.2 文件的打开与关闭	211
10.2.3 文件的读写	213
10.2.4 其他文件处理函数	220
思考和练习题	222
第11章 C的扩展C++	225
11.1 C的扩展C++	225
11.1.1 实例解析	225
11.1.2 C++的特点	227
11.2 C++的输入/输出格式控制	227
11.3 引用和 const 修饰符	231

11.3.1	引用	231
11.3.2	const 修饰符	232
11.4	内存空间的动态分配与释放	233
11.4.1	new 运算符	233
11.4.2	delete 运算符	234
11.5	使用 string 类定义和存储字符串	235
11.6	C++ 函数的新内容	236
11.6.1	重载函数	237
11.6.2	带缺省参数的函数	239
11.6.3	内联函数	239
	思考和练习题	241
第 12 章	C++ 面向对象基础	243
12.1	设计一个类和对象的程序	243
12.1.1	实例解析	243
12.1.2	面向对象程序设计的特点	245
12.2	类和对象	246
12.3	构造函数与析构函数	248
12.3.1	构造函数	248
12.3.2	析构函数	250
	思考和练习题	250
附录 1	C 语言的输入输出函数	252
附录 2	C 语言的关键字及其含义	255
附录 3	C 语言运算符优先级和结合性	257
附录 4	常用字符与 ASCII 代码对照表	259
附录 5	C 语言常用库函数	261
附录 6	常见错误与程序调试	268
	参考文献	272

第1章

C程序入门

本章导读

计算机程序是指一组指示计算机每一步动作的指令,通常用某种程序设计语言编写。程序设计语言是能完整、准确和规则地表达人们的意图,并用以指挥或控制计算机工作的“符号系统”。C程序设计语言就是一种面向过程的高级程序设计语言。

本章要点

- 了解 C 程序上机编程的基本步骤和调试方法。
- 掌握 C 程序的基本组成和特点。
- 理解结构化程序设计的基本控制结构。
- 了解程序算法的概念。

1.1 创建一个简单的 C 程序

1.1.1 实例解析

通过编写几个简单的案例来熟悉 C 程序的结构和运行方式。

实例 1-1 设计一个简单的输出界面。

```
#include <iostream.h>
void main( ) /* main 函数定义 */
{
    cout << "*****\n ";
    cout << " * 欢迎使用学生成绩管理系统 * \n ";
    cout << "*****\n ";
}
```

这是一个简单的 C 源程序文件,其中 main 是函数名,每一个 C 源程序都必须有且只有一个 main 函数,该函数没有返回值,所以函数的类型为 void。该函数的函数体由一对大括号 { } 括起来。函数体内有三条语句,每条语句末尾有一个分号。本例 cout 的功能是将双

引号内的字符串显示在屏幕上,其中\n是用转义字符表示的换行符。C程序的标准格式化输入输出函数是由 printf()和 scanf()完成,其功能强大,但对于初学者来说比较难以掌握。由于本书采用 Visual C++ 6.0 作为调试 C 程序的开发环境,所以采用了开发环境支持的 C++ 的 cout 和 cin 流对象来实现数据的输入和输出。#include <iostream.h>是文件包含预处理命令,在该文件中包含 cout 和 cin 流对象的定义。

运行结果如图 1-1 所示。



图 1-1 实例 1-1 运行结果

实例 1-2 学生成绩管理系统的功能菜单。

```
#include <iostream.h>
#include <stdlib.h>
int menu( ) /* 功能菜单界面 */
{
    int num;
    cout <<"功能菜单界面:\n ";
    cout <<"*****\n ";
    cout <<"1. 录入学生信息模块 \n ";
    cout <<"2. 修改学生信息模块 \n ";
    cout <<"3. 浏览学生信息模块 \n ";
    cout <<"*****\n ";
    cout <<"请输入对应的数字键,进入对应的功能模块:";
    cin >> num;
    return num;
}
void fun( int n) /* 功能模块提示信息 */
{cout <<"您选择的是第"<<n<<"个功能模块。 \n "; }
int main /* main 函数定义 */
{
    int num,i; /* 这是声明部分,定义变量 num 和 i */
    num = menu(); /* 以下三行是 C 语句 */
    system("cls");
    for(i = 1;i <= 3;i++) /* for 循环 */
        if(num == i)
            {fun(i); return 0;}
    cout <<"您输入的数字出错。 \n ";
    return -1;
}
```

本程序的作用是根据系统的提示和用户的操作来显示用户所选功能模块的提示信息。该文件比较复杂,在以后的学习中会逐渐体会和掌握。

本程序由三个函数组成:menu 函数、fun 函数和 main 函数。函数之间是并列关系。其执行过程是,从 main 函数开始,首先执行 menu 函数,在屏幕上显示功能菜单,请用户输入一个数字,回车后结束 menu 函数的执行,并把函数的返回值赋给 num 变量,menu 函数的类型与返回值类型一致;main 函数继续执行 system("cls") 函数,该函数是系统标准函数,包含在标准库文件 stdlib.h 中,其功能是清屏;for 语句是循环语句,分别取变量 i 的值为 1、2、3 和 num 变量比较,若比较成功,则调用 fun 函数显示用户选择的功能模块提示信息,并结束程序的运行,如果没有比较成功,则显示用户输入信息出错的提示,由于在 main 函数中有返回值,所以函数类型为 int 型。本程序的运行结果如图 1-2 所示,在第一个运行界面输入 2 并回车后,系统清屏显示下一个界面。

```

功能菜单界面:
*****
1. 录入学生信息模块
2. 修改学生信息模块
3. 浏览学生信息模块
*****
请输入对应的数字键, 进入对应的功能模块: 2

您选择的是第2个功能模块。

```

图 1-2 实例 1-2 运行结果

1.1.2 运行 C 程序的步骤和方法

Visual C++ 6.0 开发环境是一个基于 Windows 操作系统的可视化集成开发环境(Integrated Development Environment, IDE)。在该环境下用户可以开发有关 C 和 C++ 的各种应用程序,应用程序包括建立、编辑、浏览、保存、编译、链接和调试等操作,这些操作都可以通过单击菜单选项或工具栏按钮来完成,使用方便、快捷。它还提供了项目工作区(WorkSpace)、应用程序向导(AppWizard)等实用编程工具。

本节通过建立简单的应用程序,熟悉在 Visual C++ 6.0 环境下建立 C 程序的基本过程,了解 C 程序的编写、编译、链接和运行等操作。

1. Visual C++ 6.0 的主窗口

在已安装 Visual C++ 6.0 的计算机上,我们可以直接从桌面双击 Microsoft Visual C++ 6.0 图标,进入 Visual C++ 6.0 的主窗口,或者单击“开始”→“程序”菜单,选择 Microsoft Visual Studio 6.0 中的 Microsoft Visual C++ 6.0 菜单项,进入 Visual C++ 6.0 的主窗口,如图 1-3 所示。

Visual C++ 6.0 是以项目的形式来组织文件,一个项目可以包括多个 C 程序文件。所以在编写代码之前必须先创建项目,然后再创建 C 程序源文件。

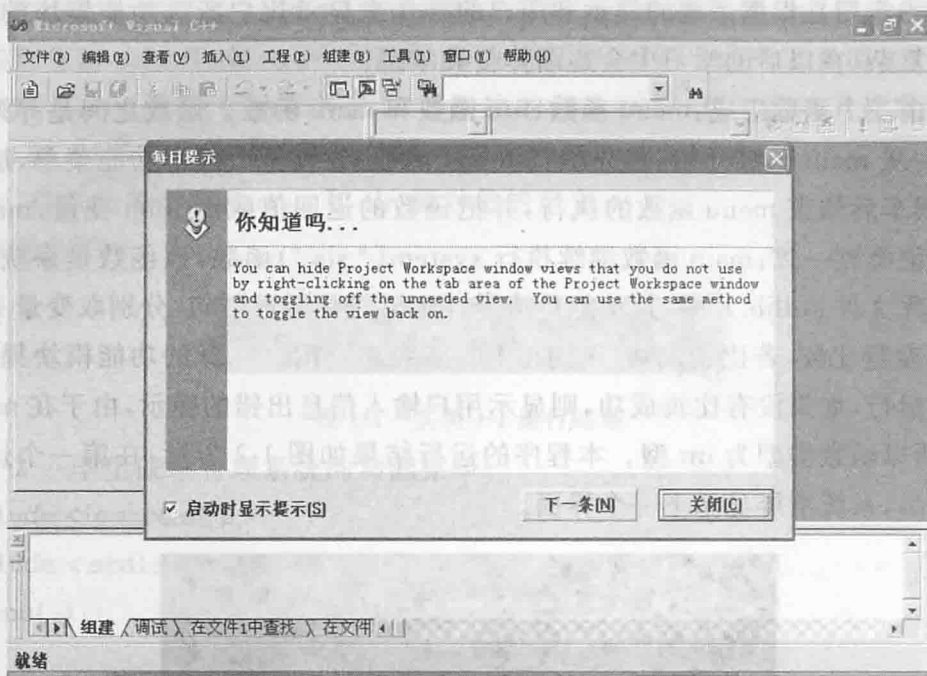


图 1-3 Visual C++ 6.0 的主窗口

2. 创建一个新的项目

在 Visual C++ 6.0 集成环境中选择“文件”→“新建”命令打开新建窗口,如图 1-4 所示。

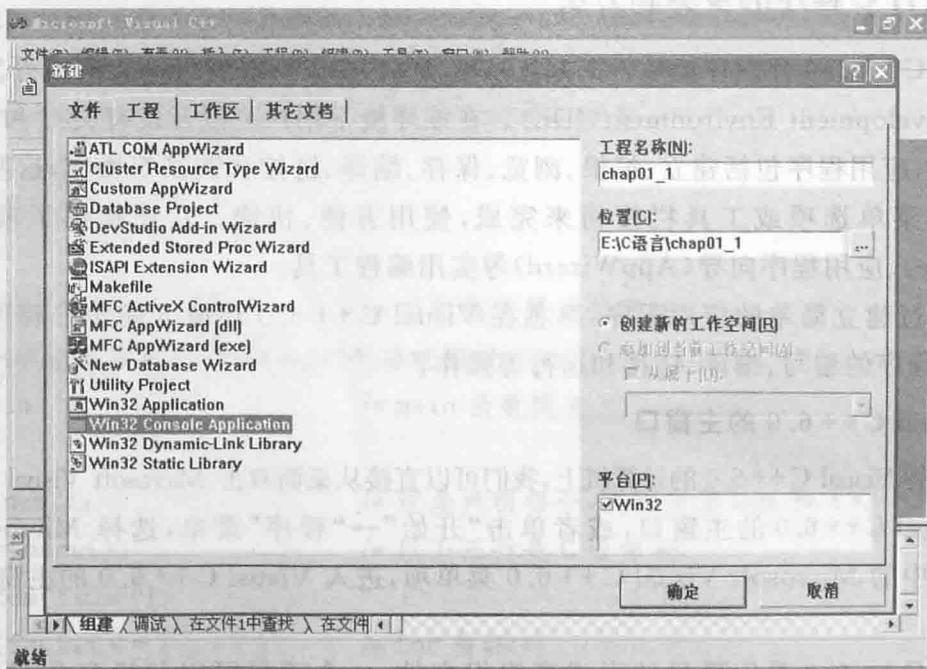


图 1-4 新建窗口

单击 Visual C++ 6.0 新建窗口的“工程”选项卡,选择 Win32 Console Application 条目,在“工程名称”文本框中输入项目名称(如 chap01_1),在位置文本框输入路径(如 E:\C

语言)。单击“确定”按钮,按照向导提示对项目的一些选项进行进一步的设置(一般采用默认设置)。这样就创建了一个项目名称为 chap01_1 的控制台程序,用户就可以在这个项目中添加或编写 C 源程序文件。

注意:在一个项目中不论有多少个 C 源程序文件,其中只能有一个 main 函数。

3. 创建一个 C 源文件

在成功创建了第一个 C++ 项目工程以后,就可以在项目工程里建立或添加 C 源程序。选择“文件”→“新建”命令打开新建窗口,先选择“文件”属性页面中的 C++ source File 条目,再在“文件名”文本框中输入文件名称 exam_1. cpp, Visual C++ 6.0 环境默认的是 C++ 源程序扩展名“. cpp”(C 源文件的扩展名为“. c”),由于在本书中的输入输出选用了 C++ 方式下的 cout 和 cin,所以本书源文件的扩展名都是用“. cpp”。选中添加到工程的复选框,工程名和位置都保持原样,最后单击“确定”按钮,进入编辑界面,如图 1-5 所示。窗口中间的左部为工作空间,双击 chap01_1 classes 可看到这个工程项目管理下的文件名和函数名等,窗口中间的右部为源文件编辑窗口,在编辑窗口输入源代码,如把前面讲解的实例 1-1 的源代码输入。

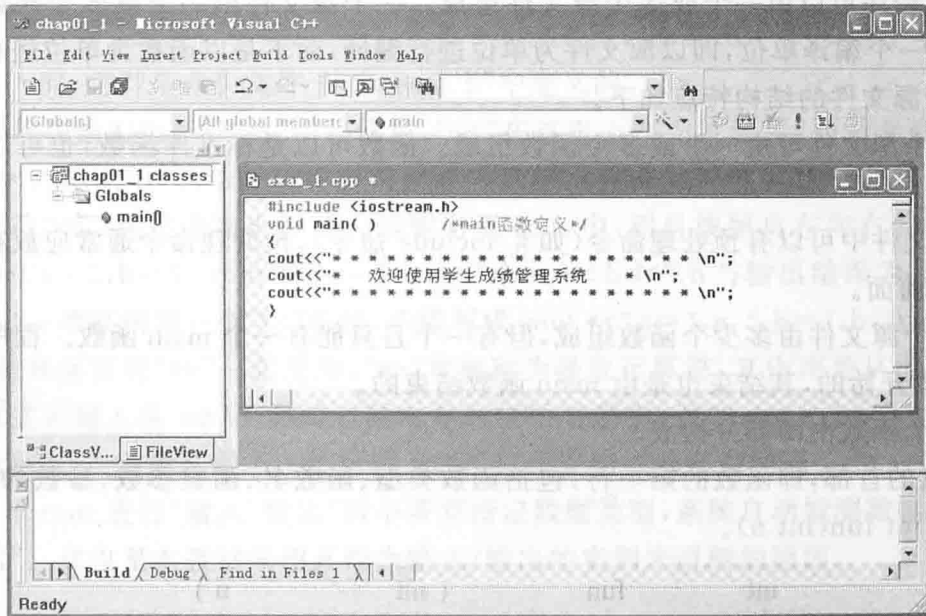


图 1-5 编辑窗口

4. 编译和调试程序

编写完成后,选择主菜单“组建”→“编译[exam_1. cpp]”命令或者按“Ctrl+F7”组合键编译源文件 exam_1. cpp,其 Output 窗口(如图 1-6 所示)中如果错误为 0,就可以组建工程,如有错误则需要反复修改和编译,直到错误为 0。

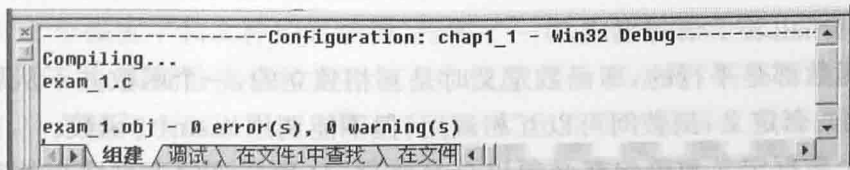


图 1-6 信息窗口

选择“组建”→“组建[exam_1.exe]”命令或者按“F7”键生成可执行文件 exam1_1.exe。
 在组建成功后,就可以选择“组建”→“执行[exam_1.exe]”命令或者按“Ctrl+F5”组合键,运行可执行文件 exam_1.exe,运行结果如图 1-7 所示。

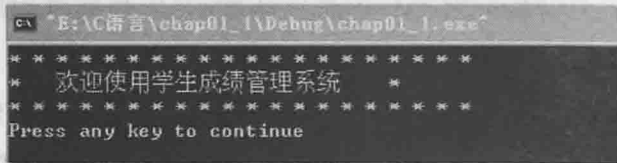


图 1-7 运行结果窗口

1.2 C 程序概述

1.2.1 C 程序的基本组成

一个 C 程序可以由一个或多个源文件组成。一个源文件由一个或多个函数组成。一个源文件是一个编译单位,即以源文件为单位进行编译,而不是以函数为单位进行编译。

一个 C 源文件的结构特点如下。

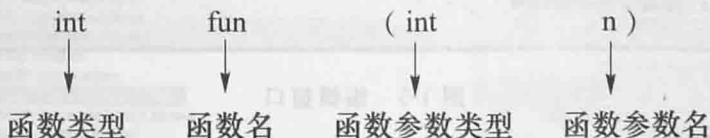
(1) 每个源文件可由一个或多个函数组成。函数可以是标准库函数,也可以是用户自定义函数。

(2) 源文件中可以有预处理命令(如 # include 命令),预处理命令通常应放在源文件或源程序的最前面。

(3) 一个源文件由多少个函数组成,但有一个且只能有一个 main 函数。程序的执行是从 main 函数开始的,其结束也是由 main 函数结束的。

(4) 一个函数由两部分构成。

① 函数的首部,即函数的第一行,包括函数类型、函数名、函数参数、参数的类型,如实例 1-2 中的 int fun(int n)。



② 函数体,即函数首部下面的 { } 内的内容。

函数体一般包括以下两个部分。

a. 声明部分:包括变量的定义和函数的声明,变量必须先定义,后使用。若无变量,则此部分可以省略。

b. 执行部分:由若干语句组成。

(5) 所有函数都是平行的,即函数定义时是互相独立的,一个函数并不从属于另一个函数,即函数不能嵌套定义,函数间可以互相调用,但不能调用 main() 函数。

(6) 每一个数据定义和语句都必须以分号结尾,分号是语句不可缺少的部分。预处理命令、函数头和花括号“}”之后不能加分号。