

```
#include <stdio.h>
struct st_teacher
{ int num;
  char name[10];
  char sex;
  char speciality;
  union
  { int class;
    char position[10];
  }cate;
};

void main()
{ struct st_teacher person[3];
  int i;
  for(i=0;i<3;i++)
  { printf("enter num sex spec name:\n");
    scanf("%d,%c,%c,%s",&person[i].num,&person[i].sex,&person[i].speciality,
    &person[i].name);
    printf("enter class/positon:\n");
    if(person[i].speciality=='s')
      scanf("%d",&person[i].cate.class);
    else if(person[i].speciality=='t')
      scanf("%s",person[i].cate.position);
    else printf("Input Error!");
  }
  printf("\n");
}
```



“ 十 三 五 ” 普通高等教育规划教材

C语言 程序设计实训

C YUYAN CHENGXU SHEJI SHIXUN

包 锋 李 峰 主编
赵凤芝 主审

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

“十三五”普通高等教育规划教材

C 语言程序设计实训

包 锋 李 峰 主 编
宋新起 王海英 副主编
赵凤芝 主 审

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书以突出实践应用能力为出发点,融入工学结合的 CDIO 工程理念,采用“案例驱动”模式,每部分内容从实例入手进行讲解、分析,并配有大量实用的各种习题进行实战训练,为扎实学习和巩固所学的 C 语言知识提供有力保障,也为强化 C 语言程序设计能力、参加计算机考试打下坚实基础。

全书共 10 章,主要内容包括:进入 C 语言程序世界准备和计算机交流、应用 C 的基础知识实现数据的处理与运算、应用顺序结构设计程序解决简单实际问题、应用选择结构设计程序实现分支判断、应用循环结构设计程序实现重复操作、应用数组设计程序实现批量数据处理、应用函数设计程序实现模块化设计、应用指针设计程序增加独有特色、自己定义数据类型完成复杂数据处理、应用文件管理数据。附录中列出了 C 源程序调试运行时常见错误和计算机基础知识训练题。本书与《C 语言程序设计能力教程(第四版)》(赵凤芝、包锋主编,中国铁道出版社出版)教材配套,每章包括实例解析和大量习题,读者在使用时可根据需要进行取舍。

书中习题丰富、题型全面,适合作为高等院校程序设计课程的实训教材或辅助教材,也可作为参加培训、考级、考试的人员的强化训练用书。

图书在版编目(CIP)数据

C 语言程序设计实训 / 包锋, 李峰主编. —北京: 中国铁道出版社, 2018.2

“十三五”普通高等教育规划教材

ISBN 978-7-113-24298-5

I. ①C… II. ①包… ②李… 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(2018)第 028234 号

书 名: C 语言程序设计实训
作 者: 包 锋 李 峰 主 编

策 划: 王春霞 读者热线: (010) 63550836
责任编辑: 王春霞 徐盼欣
封面设计: 刘 颖
责任校对: 张玉华
责任印制: 郭向伟

出版发行: 中国铁道出版社(100054, 北京市西城区右安门西街 8 号)
网 址: <http://www.tdpress.com/51eds/>
印 刷: 虎彩印艺股份有限公司
版 次: 2018 年 2 月第 1 版 2018 年 2 月第 1 次印刷
开 本: 787 mm × 1092 mm 1/16 印张: 9.75 字数: 234 千
书 号: ISBN 978-7-113-24298-5
定 价: 28.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社教材图书营销部联系调换。电话: (010) 63550836

打击盗版举报电话: (010) 51873659

前 言

PREFACE

C 语言不仅适合于系统软件的设计，而且适合于应用程序设计。C 语言以其强大的功能，丰富的数据类型，使用灵活，兼具面向硬件编程的低级语言特性及通用性强、可移植性好等高级语言特性，是国内外广泛流行的程序设计语言，一直长盛不衰，成为软件开发中的主流语言之一。

目前，许多开发工具（包括 Visual C++ 和 Visual C++ .NET 及 Java 等开发工具）都还遵循着标准的 C 语言基本语法。在很多嵌入式系统的软件设计中，多数都采用 C 语言来开发。可以说，C 语言是程序开发人员必须掌握的基本功，也是国内各高校广泛学习和普遍使用的一种重要的计算机语言。目前，全国计算机等级考试、全国计算机应用技术证书考试、全国计算机技能大赛等都将 C 语言列入考试范围。

本书以突出实践应用能力为出发点，融入工学结合的 CDIO 工程理念，采用流行的“案例驱动”模式，每部分内容从实例入手进行讲解、分析，并配有大量实用的各种习题进行实战训练，为扎实学习和巩固所学的 C 语言知识提供有力保障，也为强化 C 语言程序设计能力、参加计算机考试打下坚实基础。

全书共 10 章，主要内容包括：进入 C 语言程序世界准备和计算机交流、应用 C 的基础知识实现数据的处理与运算、应用顺序结构设计程序解决简单实际问题、应用选择结构设计程序实现分支判断、应用循环结构设计程序实现重复操作、应用数组设计程序实现批量数据处理、应用函数设计程序实现模块化设计、应用指针设计程序增加独有特色、自己定义数据类型完成复杂数据处理、应用文件管理数据。附录中列出了 C 源程序调试运行时常见错误和计算机基础知识训练题。本书与《C 语言程序设计能力教程（第四版）》（赵凤芝、包锋主编，中国铁道出版社出版）教材配套，每章包括实例解析和大量习题，读者在使用时可根据需要进行取舍。

本书适合作为高等院校程序设计课程的实训教材或辅助教材，也可作为参加培训、考试的人员的强化训练用书。

本书由包锋、李峰任主编，宋新起、王海英任副主编，赵凤芝主审。其中，包锋编写了第 1、7 章，宋新起编写了第 4、8 章，李峰编写了第 5、6、9、10 章及附录，王海英编写了第 2、3 章。

由于编者水平有限，书中疏漏和不足之处在所难免，敬请有关专家和广大读者不吝指正，编者的电子邮箱是 qhdcomputer@163.com。

编 者

2018 年 1 月

目 录

CONTENTS

第 1 章 进入 C 语言程序世界准备和 计算机交流	1	实例解析	32
实例解析	1	一、for、while 循环程序设计	32
一、初识 C 语言	1	二、多重循环程序设计	34
二、设计简单的 C 程序	1	三、穷举类型的程序设计	34
三、程序的调试与运行	2	四、递推类型的程序设计	36
小结	5	小结	37
实战训练	5	实战训练	37
第 2 章 应用 C 的基础知识实现数据的 处理与运算	8	第 6 章 应用数组设计程序实现批量 数据处理	50
实例解析	8	实例解析	50
一、数据类型	8	一、一维数组的应用	50
二、运算符与表达式	10	二、二维数组的应用	52
小结	12	三、字符数组的应用	53
实战训练	12	小结	55
第 3 章 应用顺序结构设计程序解决 简单实际问题	16	实战训练	56
实例解析	16	第 7 章 应用函数设计程序实现模块化 设计	66
一、数据的输入与输出	16	实例解析	66
二、顺序结构程序设计	16	一、函数的调用与参数传递	66
小结	17	二、函数的嵌套与递归	70
实战训练	18	三、变量作用域与存储类别	72
第 4 章 应用选择结构设计程序实现 分支判断	21	小结	73
实例解析	21	实战训练	73
一、if 语句多重选择结构设计	21	第 8 章 应用指针设计程序增加独有 特色	89
二、switch 流程设计	23	实例解析	89
小结	24	一、指针怎么用	89
实战训练	24	二、指针的高级应用	97
第 5 章 应用循环结构设计程序实现 重复操作	32	小结	99
		实战训练	99
		第 9 章 自己定义数据类型完成复杂 数据处理	111

实例解析.....	111	第 10 章 应用文件管理数据.....	130
一、结构体类型、变量及数组的 应用.....	111	实例解析.....	130
二、结构体指针与指针处理链表.....	113	一、对文件进行操作.....	130
三、结构体与函数参数.....	118	二、文件的应用.....	131
四、共用体、枚举类型及自定义 类型的应用.....	119	小结.....	132
小结.....	122	实战训练.....	133
实战训练.....	122	附录 A 常见编译错误信息.....	140
		附录 B 计算机基础知识训练题.....	145

第1章

进入 C 语言程序世界准备和计算机交流

C 语言简洁、紧凑，使用方便、灵活；运算符、数据结构丰富，具有现代化语言的各种数据结构；具有结构化的控制语句；语法限制不太严格，程序设计自由度大；生成目标代码质量高，程序执行效率高。本章通过对几个小程序的讲述，使读者对 C 语言程序的结构有大致的了解，熟悉 Visual C++ 6.0 的开发环境和调试步骤。

实训目标

通过本章训练，你将能够：

- 了解 C 语言程序的基本构成和简单的程序。
- 掌握 C 程序编写和运行步骤。

实例解析

一、初识 C 语言

【实例 1.1】 编写第一个 C 语言程序，输出 “I am a student.”。

解：

```
#include <stdio.h>
void main()
{
    printf("I am a student.");           /*输出 “I am a student.” */
}
```

本程序运行结果为：

I am a student.

解析：

C 程序首先要有主函数 “main()”，函数体是要输出一行文字，并没用到变量，不用说明，只是用到了输出函数 “printf()”。注意，一个语句结束后写 “;”，函数体要用 { } 括起来。在 C 程序中 “/*...*/” 表示它们之间的语句只起说明作用，并不执行。

二、设计简单的 C 程序

【实例 1.2】 从键盘输入两个数，求这两个数之积并输出。

解：

```
#include <stdio.h>
```

```

void main()
{
    int a,b,sum=0;           /*定义变量*/
    scanf("%d,%d",&a,&b);   /*输入 a 和 b 的值*/
    sum=a*b;
    printf("sum=%d",sum);   /*输出 a 与 b 的积*/
}

```

本程序运行结果为：

3,7 <回车>

sum=21

解析：

本题主要是考查 scanf()函数的使用，使用 scanf()函数时，在变量的前面不要忘记取地址符“&”。运行程序时在黑屏界面输入“3,7”（输入数据时，若在程序中两个输出格式字符间有逗号，输入时要写逗号；若输出格式字符间是空格，输入的两个值之间要有空格）。

【实例 1.3】 已知 $a=2$, $b=3$, $c=4$, $x=5$, 求表达式 ax^2+bx+c 的值。

解：

```

#include <stdio.h>
void main()
{
    int a,b,c,x=5,sum;
    a=2;
    b=3;
    c=4;
    sum=a*x*x+b*x+c;
    printf("%d*%d*%d+%d*%d+%d=%d\n",a,x,x,b,x,c,sum);
}

```

本程序运行结果为：

2*5*5+3*5+4=69

解析：

这是一道简单的计算题，在 C 语言中，两个操作数之间必须有符号，乘号“*”不能省略。对变量赋值，一种是在定义时直接赋值，它们之间用逗号分开；另一种是在执行语句中赋值，这时需要用分号结束。

三、程序的调试与运行

Visual C++ 6.0（简称 VC++或 VC）提供了可视化的集成开发环境，主要包括文本编辑器、资源编辑器、工程创建工具、Debugger 调试器等实用开发工具。Visual C++ 6.0 分为标准版、专业版和企业版 3 种，但其基本功能是相同的。

下面介绍如何在 Visual C++ 6.0 中实现 C 程序的编辑和运行。

1. Visual C++ 6.0 启动界面

在 Windows 系统任务栏中，选择“开始”→“所有程序”→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0 命令，即可启动 Visual C++ 6.0 集成开发环境，启动界面如图 1-1 所示。

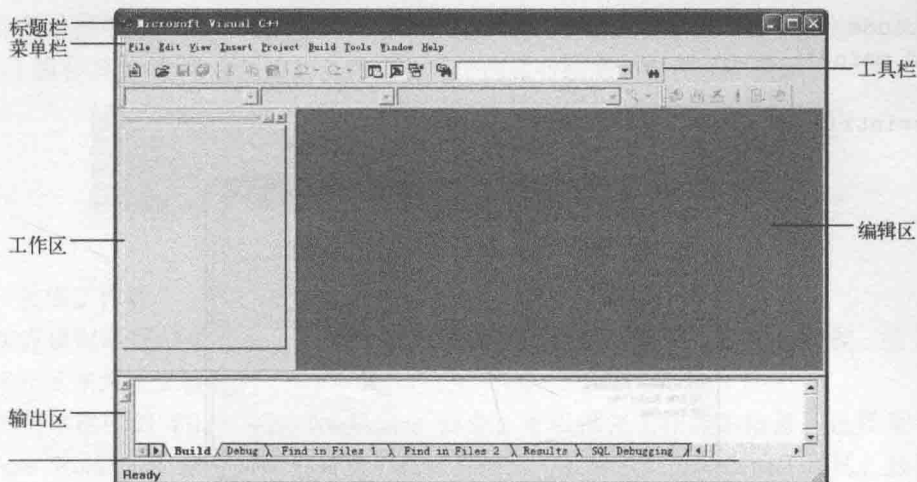


图 1-1 Visual C++ 6.0 启动界面

2. 在 Visual C++ 6.0 中编译 C 程序


(1) 创建文件

在 VC++ 中创建 C 程序文件有多种方式，现列举两种：

① 在任意位置处创建一个记事本文件，保存格式由 .txt 修改为 .c，如 exam.c。启动 VC++ 环境，选择菜单栏中的 File→Open 命令，在弹出的“打开”对话框中选择创建的 exam.c 文件，单击“打开”按钮，如图 1-2 所示，即可进入 VC++ 的代码编辑窗口。



图 1-2 打开 exam.c 文件

② 启动 VC++，选择菜单栏中的 File→New 命令，在弹出的 New 对话框中选择 Files 选项卡。在左边列出的选项中，选择 C++ Source File 或 Text File 选项，在右边 File 文本框中输入 exam.c，单击 Location 文本框右侧的  按钮修改保存的位置，如图 1-3 所示。单击 OK 按钮，即可进入 VC++ 的代码编辑窗口。

(2) 编辑代码并保存

① 编辑代码：在 VC++ 代码编辑区中输入 exam.c 的源代码，完成后如图 1-4 所示。源代码如下：

```
/** exam.c **/
```

```
#include <stdio.h>
void main()
{
    printf("欢迎使用 VC++ 编译 C 程序! \n");
}
```

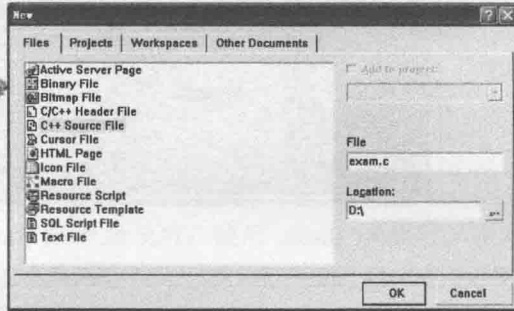



图 1-3 创建 exam.c 文件

② 保存：选择菜单栏中的 File→Save 命令（Save As...命令可修改原默认存储路径），也可单击工具栏中的“保存”按钮.

(3) 编译、连接、运行源程序


选择菜单栏中的 Build→Compile exam.c 命令（或单击工具栏中的按钮，或按【Ctrl+F7】组合键），在弹出的对话框中单击“是（Y）”按钮，这时系统开始对当前的源程序进行编译。在编译过程中，将所发现的错误显示在“输出区”窗口中，错误信息中指出错误所在行号和错误的原因。当程序出现错误时，根据提示信息修改源程序代码，再进行编译直至编译正确，如图 1-5 所示。




图 1-4 编辑代码窗口



图 1-5 输出窗口中的编译信息

当输出窗口中的信息提示为 exam.obj - 0 error(s), 0 warning(s) 时，表示编译正确。

选择菜单栏中的 Build→Build exam.exe 命令（或单击工具栏中的按钮，或按【F7】键），连接正确时，生成可执行文件 exam.exe，如图 1-6 所示。该文件保存在 exam.c 同一文件夹下的 Debug 文件夹中。

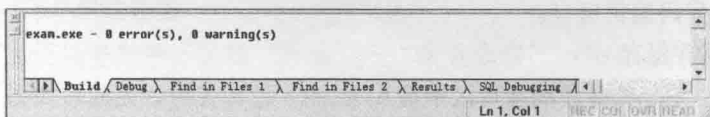



图 1-6 输出窗口中的连接信息

选择菜单栏中的 Build→Execute Program exam.exe 命令（或单击工具栏中的  按钮，或按【Ctrl+F5】组合键），即可看到控制台程序窗口中的运行结果，如图 1-7 所示。

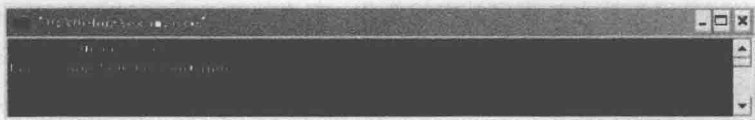


图 1-7 程序 exam.c 的运行结果

（4）关闭工作区

每次完成对程序的操作后，必须安全地保存好已经建立的应用程序与数据，应正确地使用关闭工作区来终止工程。

选择菜单栏中的 File→Save Workspace 命令，可以保存工作区的信息；选择菜单栏中的 File→Close Workspace 命令，可以终止工程、保存工作区信息、关闭当前工作区；选择菜单栏中的 File→Exit 命令，可以退出 VC++ 环境。

小 结

本章主要介绍了 C 语言的特点、程序结构、常用算法及程序运行过程。通过编写运行几个小程序，熟悉 C 语言的开发环境。

实战训练

一、选择题

1. 对于一个正常运行的 C 程序，以下叙述中正确的是（ ）。
 - A. 程序的执行总是从 main() 函数开始，在 main() 函数中结束
 - B. 程序的执行总是从程序的第一个函数开始，在 main() 函数中结束
 - C. 程序的执行总是从 main() 函数开始，在程序的最后一个函数中结束
 - D. 程序的执行总是从程序中的第一个函数开始，在程序的最后一个函数中结束
2. 以下叙述不正确的是（ ）。
 - A. 一个 C 源程序可由一个或多个函数组成
 - B. 一个 C 源程序必须包含一个 main() 函数
 - C. C 程序的基本组成单位是函数
 - D. 在 C 程序中注释说明只能位于一条语句的后面
3. 以下叙述中正确的是（ ）。
 - A. C 程序中的注释只能出现在程序的开始位置和语句的后面
 - B. C 程序书写格式严格，要求一行内只能写一条语句
 - C. C 程序书写格式自由，一条语句可以写在多行上
 - D. 用 C 语言编写的程序只能放在一个程序文件中
4. 下列说法正确的是（ ）。
 - A. main() 函数必须放在 C 程序的开头

- B. main()函数必须放在 C 程序的最后
 C. main()函数可以放在 C 程序的中间部分,但在执行 C 程序时是从程序开头执行的
 D. main()函数可以放在 C 程序的中间部分,但在执行 C 程序时是从 main()函数开始的
5. () 是 C 程序的基本构成单位。
 A. 函数 B. 函数和过程 C. 超文本过程 D. 子程序
6. 以下说法正确的是 ()。
 A. C 语言程序总是从第一个定义的函数开始执行
 B. 在 C 语言程序中,要调用的函数必须在 main()函数中定义
 C. C 语言程序总是从 main()函数开始执行
 D. C 语言程序中的 main()函数必须放在程序的开始部分
7. 在 C 语言中,每个语句和数据定义用 () 结束。
 A. 句号 B. 逗号 C. 分号 D. 括号
8. C 语言规定:在一个源程序中,main()函数的位置 ()。
 A. 必须在最开始 B. 必须在系统调用的库函数的后面
 C. 可以在任意位置 D. 必须在源文件的最后
9. 一个 C 语言程序是由 ()。
 A. 一个主程序和若干子程序组成
 B. 函数组成,并且每个 C 程序必须且只能有一个主函数
 C. 若干过程组成
 D. 若干子程序组成
10. 下列说法正确的是 ()。
 A. 在执行 C 程序时不是从 main()函数开始的
 B. C 程序书写格式严格限制,一行内必须写一条语句
 C. C 程序书写格式自由,一条语句可以分写在多行上
 D. C 程序书写格式严格限制,一行内必须写一条语句,并要有行号
11. 一个 C 程序可以包含任意多个不同名的函数,但有且仅有一个 (),一个 C 程序总是从 () 开始执行。
 A. 过程 B. 主函数 C. 函数 D. include
12. C 语言规定,必须用 () 作为主函数名。
 A. function B. include C. main D. stdio
13. 以下叙述中正确的是 ()。
 A. 用 C 程序实现的算法必须要有输入和输出操作
 B. 用 C 程序实现的算法可以没有输出但必须要有输入
 C. 用 C 程序实现的算法可以没有输入但必须要有输出
 D. 用 C 程序实现的算法可以既没有输入也没有输出

二、判断题

1. 一个 C 程序由一个或多个函数组成。 ()
2. 一个 C 程序必须包含一个 main()函数。 ()

3. C程序的每一行上可以写多条语句。 ()
4. 在C语言程序中 main()函数必须放在程序的开始位置。 ()
5. C语言程序的开始执行点是程序中的 main()函数。 ()
6. C程序的书写格式虽然自由,但为了清晰,一般在一行内写一条语句。 ()
7. 分号是语句的必要组成部分,所以函数首部的后面和编译预处理的后面都得加分号。 ()
8. 注释在程序执行时不产生任何操作,因此在程序中不提倡注释。 ()
9. C程序的书写格式自由,一行内可以写多条语句,一条语句也可以写在多行上。 ()
10. C程序中以#include 和#define 开始的程序行均不是C语句。 ()
11. 一个C程序可以由若干源程序文件(分别进行编译的文件模块)组成。 ()
12. 在C语言中运算符的优先级高低的排列顺序是:逗号运算符、算术运算符、赋值运算符。 ()
13. C程序书写格式自由,一条语句可以写在多行上。 ()
14. C语言程序由主函数和0个或多个函数组成。 ()
15. C语言程序由主程序和子程序组成。 ()
16. C语言程序由子程序组成。 ()
17. C语言程序由过程组成。 ()
18. C程序书写格式严格,要求一行内只能写一条语句。 ()
19. 在对一个C程序进行编译的过程中,可发现注释中的拼写错误。 ()
20. C语言源程序文件通过了编译、连接之后,生成一个扩展名为.exe的文件。 ()

三、填空题

1. 一个C程序的执行是从_____函数开始。
2. 一个C程序的执行是随着_____函数的结束而结束。
3. 一个C程序必须包含一个_____函数。
4. C程序的基本组成单位是_____。
5. C程序最先执行的函数名为_____。
6. 关键字是由C语言规定的具有特定意义的字符串,通常也称_____。
7. C语言空语句的形式是_____。
8. C语言中,注释部分以_____开始。

第 2 章

应用 C 的基础知识实现数据的处理与运算

C 语言的数据结构是以数据类型形式出现的，数据类型是指数据的内部表示形式，它是进行 C 语言程序设计的基础。每种数据类型又分常量和变量，可以进行数据的运算。C 语言的运算符极其丰富，有 30 多种。本章将讨论 C 语言的数据类型及运算符，主要是基本数据类型及运算符与表达式等，通过对上述问题的训练，使读者能够掌握 C 语言的基础知识，为今后的程序设计打下基础。

实训目标

通过本章训练，你将能够：

- ☑ 掌握数据的基本类型。
- ☑ 掌握对变量与常量的赋值。
- ☑ 掌握常用的运算符，正确书写表达式。

实例解析

一、数据类型

1. 常量与变量

【实例 2.1】 已知一个圆的半径是 5 cm，求此圆的周长及面积。

解：

```
#include <stdio.h>
#define PI 3.14          /*定义常量 PI，值为 3.14*/
void main()
{
    float r,a,b;
    r=5;
    a=2*PI*r;           /*计算圆周长*/
    b=PI*r*r;          /*计算圆面积*/
    printf("girth=%.2f,area=%.2f\n",a,b);
}
```

本程序运行结果为：

```
girth=31.40,area=78.50
```

解析：

在 C 语言中并没有字母 π ，只能用别的名称代替，此例中定义了一个字符常量 PI。

相关知识

在程序运行过程中，其值不能被改变的量称为常量。常量的值在其作用域内不能改变，也不能再被赋值。习惯上，符号常量名用大写，变量名用小写，以示区别。C定义常量时，在程序的上方，“#define 常量名 常量定值”。

【实例 2.2】 求 1~10 的累加和。

解：

```
#include <stdio.h>
void main()                /*主函数*/
{
    int i,sum=0;           /*定义变量 i,sum*/
    for(i=1;i<=10;i++)    /*i 从 1 到 10, 共循环 10 次*/
        sum=sum+i;       /*进行累加*/
    printf("sum=%d",sum); /*输出累加和*/
}
```

本程序运行结果为：

```
sum=55
```

解析：

本题中用到了一个 for 循环语句，其作用是让 i 从 1 每次循环增加 1。关于循环语句的使用将在第 4 章学习。

相关知识

其值可以改变的量称为变量。一个变量应该有一个名字，在内存中占据一定的存储单元。注意区分变量名和变量值这两个不同的概念。标识符用来标识变量名、符号常量名、函数名、数组名。文件名只能由字母、数字和下划线三种字符组成，且第一个字符必须为字母或下划线。

2. 整型、实型与字符型数据

【实例 2.3】 若 a=1, b=2, x=1.2, y=2.1, n=128765, c1='a', 想得到以下的输出格式和结果，请写出程序（包括定义变量类型和设计输出）。

```
a= 1  b= 2
x=1.200000,y=2.100000
y-x=0.900000, x+y=3.30
n= 128765
c1='a' or 97
```

解：

```
#include <stdio.h>
void main()
{
    int a,b;
    long int n;
    float x,y;
    char c1;
    a=1;b=2;
    x=1.2;y=2.1;
```

```

n=128765;
c1='a';
printf("\n");
printf("a=%2d b=%2d \n",a,b);
printf("x=%8.6f,y=%8.6f\n",x,y);
printf("y-x=%5.2f, x+y=%5.2f \n",y-x,x+y);
printf("n=%9ld\n",n);
printf("c1='%c' or %d\n",c1,c1);
}

```

相关知识

C 语言的数据结构是以数据类型形式出现的。C 的数据类型包括基本类型、构造类型、指针类型和空类型。数据基本类型包括整型 (int)、字符型 (char)、实型 (float、double float) 和枚举类型。int 对应的输出格式字符为 %d，若变量值较大时，可使用长整型 (long int)。%md 输出格式中，m 为指定的输出字段的宽度。若数据的输出位数小于 m，则左端补空格；若大于 m，则按实际位数输出。float 定义为实数 (包括单、双精) 以小数输出，%f 默认输出为 6 位小数，%m.nf 表示共输出 m 列，其中 n 位小数。每个字符以 ASCII 码存储，用 %c 对应该字符，用 %d 对应其 ASCII 码。%e 输出格式表示以科学记数法表示，主要适用于 float 类型和 double 类型。

【实例 2.4】 编写 C 程序，从键盘输入一个英文字母 (不包括 a、z、A、Z)，在屏幕上显示出其前后相连的 3 个英文字母。

解：

```

#include <stdio.h> /*引用一个预处理文件 stdio.h*/
void main()
{
    char c; /*定义字符变量 c*/
    c=getchar(); /*获得一个字符 c*/
    if((c>'a' && c<='z') || (c>'A' && c<='Z'))
        printf("%c %c %c\n",c-1,c,c+1);
}

```

本程序运行结果为：

B <回车>

A B C

解析：

程序中获得字符 c 使用了 getchar() 函数，在程序上方，必须引用文件 stdio.h。

二、运算符与表达式

1. 算术运算符和算术表达式

【实例 2.5】 写出程序运行结果。

解：

```

#include <stdio.h>
void main()
{
    int i,j,m,n; /*定义变量 i,j,m,n*/
}

```

```

i=8;
j=10;
m=++i;           /*变量 i 先加*/
n=j++;           /*变量 j 后加*/
printf("%d,%d,%d,%d",i,j,m,n);
}

```

本程序运行结果为:

```
9,11,9,10
```

解析:

程序中变量 i 和 j 都自身加 1, 所以输出的 i 和 j 都增加了 1。但又有所不同, $m=++i$ 是变量 i 自增 1 后再把值传给变量 m ; $n=j++$ 是先把 j 值传给变量 n 后再自增 1。

相关知识

自增运算符 ($++$) 和自减运算符 ($--$) 只能用于变量, 而不能用于常量或表达式, 如 $5++$ 或 $(a+b)++$ 都是不合法的。它们的结合方向是“自右至左”。它们常用于后面章节的循环语句中, 使循环变量自动增加 1, 也用于指针变量, 使指针指向下一个地址。

【实例 2.6】强制类型转换。

解:

```

#include <stdio.h>
void main()
{
    float x;           /*定义 x 为实型*/
    int m,n;           /*定义 m、n 为整型*/
    x=2.7;             /*给 x 赋初值*/
    m=(int)x;          /*把 x 强制转换成整型赋值给 m*/
    n=(int)(x+0.5);    /*把 x 加上 0.5 的值强制取整再赋值给 n, 可实现四舍五入取整*/
    printf("x=%f,m=%d,n=%d",x,m,n);
}

```

本程序运行结果为:

```
x=2.70000,m=2,n=3
```

解析:

本题是把实型 x 强制转换成整型, 要把 x 前面的 int 用括号括起来。强制类型转换的一般形式为(类型名)(表达式), 表达式应该用括号括起来。

相关知识

类型转换有两种: 一种是在运算时不必用户指定, 系统自动进行的类型转换。另一种是强制类型转换, 当自动类型转换不能实现目的时, 可以用强制类型转换。如“ $\%$ ”运算符要求其两侧均为整型量, 若 x 为 $float$ 型, 则“ $x\%$ 3”不合法, 必须用“ $(int)x\%$ 3”。强制类型转换运算优先于 $\%$ 运算, 因此先进行 $(int)x$ 运算, 再对 3 取模。

2. 赋值运算符和赋值表达式

【实例 2.7】若 a 的初值为 12, 求赋值表达式 $a+=a--a*a$ 中最终 a 的值。

解析: 以下为此赋值表达式的求解步骤。

① 先进行“ $a--a*a$ ”运算, 相当于 $a=a-a*a=12-144=-132$, 此时 a 的值由 12 变成 -132。