

本章简介 本章介绍了程序设计的基本概念，C 语言的发展和特点，C 语言的结构和组成，算法的概念和流程图，最后介绍了 C 语言上机操作步骤和三种编程环境，要学会调试简单的程序，为后面的编程内容打下良好基础。

**知识点**

1. 计算机语言和程序设计概念
2. C 语言的历史、发展和特点
3. C 语言的基本结构和组成
4. C 语言算法的概念和描述工具
5. C 语言上机操作步骤和三种编程环境介绍

1.1 程序设计的基本概念

计算机系统分成两大部分：硬件系统和软件系统。其中硬件系统一般由 CPU、存储器、输入设备和输出设备构成；而软件系统则由各种程序组成，没有软件的计算机被称为裸机，什么也干不了。而软件开发（或程序开发）是使用各种计算机程序设计语言来编程实现的。

1.1.1 程序的概念

▶ 1. 程序

计算机程序是指存储在计算机中可以被计算机识别并运行的一系列指令。为了完成某种任务而编写的一系列指令的过程就是程序设计。程序设计过程中还需要不断地修改和完善，这个过程称为调试和测试。

用计算机语言描述的算法称为计算机程序，简称**程序**。人们编写程序之前，为了直观或符合人的思维方式，常常先用其他方式描述算法，然后翻译成计算机程序。

▶ 2. 程序设计语言

人和计算机通信需要通过计算机语言。计算机语言是面向计算机的人造语言，是进行程序设计的工具，因此也称为程序设计语言。

程序设计语言可以分为**机器语言**、**汇编语言**、**高级语言**。计算机硬件能直接执行的是机器语言程序，机器语言都是二进制代码。汇编语言也称符号语言，用汇编语言编写的程序称汇编语言程序。而高级语言因为代码与人类语言相近，容易理解，成为现在主流编程语言。汇编语言和高级语言都必须编译成机器语言才能执行。

高级语言程序设计经过多年的发展，分成两类：面向过程的程序设计和面向对象的程序设计。面向过程语言如 Pascal 和 C 等命令式语言；面向对象的程序设计语言如 C++、Java、Visual C++、Visual Basic、Visual C#.net 等。



视频讲解

1.1.2 程序设计的一般过程

程序设计过程一般包括以下五部分。

- (1) **问题分析与描述**。分析具体任务，确定数据描述和功能描述，为编写代码提供依据。
- (2) **确定算法**。根据数据描述和功能描述确定解决问题的方法，即完成任务的具体步骤。
- (3) **编写程序代码**。根据确定的数据结构和算法，使用选定的计算机语言编写程序代码，简称“编程”。
- (4) **编译运行与调试**。将编写完的程序输入到计算机中，对程序进行运行并验证，发现错误则需要修改，反复执行，直到程序符合任务要求。
- (5) **整理文档资料**。根据数据结构和程序整理编写相关的文档资料。

1.1.3 程序设计的方法

程序设计方法主要包括**面向过程**的程序设计和**面向对象**的程序设计方法。

面向过程程序设计是指把代码的编写看成对数据加工的过程。采用的方法主要是自顶向下，逐步求精。程序按功能划分为若干模块，这些模块形成调用的层次树状结构；模块间的关联尽可能的简单，模块的功能相对独立；每个模块均由顺序、选择和循环三种基本结构组成。Pascal 语言、C 语言属于面向过程语言的代表。

面向对象程序设计是在吸取结构化程序设计的一切优点的基础上发展起来的一种新的程序设计方法。它的本质是把数据和处理数据的过程当成一个整体——对象。其不同于面向过程的主要特点在于“代码重用”问题的解决方案。用“对象”描述事物，用“属性”和“方法”描述对象的特征和行为，用“类”抽象化“对象”，所以更容易理解和应用。Java、Visual C#.net 属于面向对象语言的代表。

1.2 C 语言简介



视频讲解

1.2.1 C 语言的发展

C 语言是贝尔实验室于 20 世纪 70 年代初期研制出来的，并随着 UNIX 操作系统的日益广泛使用，迅速得到推广。后来，C 语言又被多次改进，并出现了多种版本。20 世纪 80 年代初（1983 年），美国国家标准化协会（ANSI, American National Standard Institute）根据 C 语言问世以来的各种版本对 C 语言进行了发展和扩充，制订了 ANSI C 标准（1989 年再次做了修订）。本书以 ANSI C 标准来介绍。

目前，在微机上广泛使用的 C 语言编译系统有 Microsoft C（简称 MSC）、Turbo C（简称 TC）、Borland C（简称 BC）等。虽然它们的基本部分都是相同的，但还是有一些差异，所以请读者注意自己所使用的 C 编译系统的特点和规定（可参阅相应的手册）。

本书以 Turbo C 2.0 和 Visual C++ 6.0 为学习平台，分别简称为 TC 和 VC。这两个平台在 C/C++ 学习与实验系统中能直接调用并运行，方便了在图形化操作系统下调试 C 语言的过程。因为部分程序在这两个平台运行结果不同，对于不同之处在后面章节中都进行了说明。

1.2.2 C 语言的特点

- (1) **简洁紧凑、灵活方便**。C 语言一共只有 32 个关键字、9 种控制语句，程序书写自由，

主要用小写字母表示。它把高级语言的基本结构和语句与低级语言的实用性结合起来。

(2) **运算符丰富**。C 的运算符包含的范围很广泛, 共有 34 种运算符。C 语言把括号、赋值、强制类型转换等都作为运算符处理, 从而使 C 的运算类型极其丰富, 表达式类型多样化。灵活使用各种运算符可以实现在其他高级语言中难以实现的运算。

(3) **数据结构丰富**。C 的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等。能用来实现各种复杂的数据类型的运算。并引入了指针概念, 使程序效率更高。另外 C 语言具有强大的图形功能, 支持多种显示器和驱动器。且计算功能、逻辑判断功能强大。

(4) **C 是结构式语言**。结构式语言的显著特点是代码及数据的分隔化, 即程序的各个部分除了必要的信息交流外彼此独立。C 语言是以函数形式提供给用户的, 这些函数可方便的调用, 并具有多种循环、条件语句控制程序流向, 从而使程序完全结构化。

(5) **C 语法限制不太严格, 程序设计自由度大**。一般的高级语言语法检查比较严, 能够检查出几乎所有的语法错误。而 C 语言允许程序编写者有较大的自由度。

(6) **C 语言允许直接访问物理地址, 可以直接对硬件进行操作**。因此既具有高级语言的功能, 又具有低级语言的许多功能, 能够像汇编语言一样对位、字节和地址进行操作, 而这三者是计算机最基本的工作单元, 可以用来写系统软件。

(7) **C 语言程序生成代码质量高, 程序执行效率高**。一般只比汇编程序生成的目标代码效率低 10%~20%。

(8) **C 语言适用范围大, 可移植性好**。C 语言有一个突出的优点就是适合于多种操作系统, 如 DOS、UNIX, 也适用于多种机型。

1.2.3 C 语言的书写规则

从书写清晰, 便于阅读、理解和维护的角度出发, 在书写程序时应遵循以下规则。

(1) 一个说明或一个语句占一行。

(2) 用花括号 {} 括起来的部分, 通常表示程序的某一层结构。{} 一般与该结构语句的第一个字母对齐, 并单独占一行。

(3) 低一层次的语句或说明可以比高一层次的语句或说明缩进若干字符后书写 (一般缩进 2 个英文字符), 以便看起来更加清晰, 增加程序的可读性。

在编程时应力求遵循这些规则, 以养成良好的编程风格。

1.2.4 C 程序的几种类型文件

因为 C 语言是高级语言, 编写完的源程序 (.c) 并不能够直接在机器上运行, 必须经过编译生成二进制的目标程序 (.obj), 然后将目标程序连接系统库函数文件, 最终生成可执行程序 (.exe)。

(1) **源程序**。用高级语言或汇编语言编写的程序称为源程序。源程序不能直接在计算机上执行, 需要用编译程序将源程序翻译为二进制形式的代码。**C 语言源程序的扩展名为“.c”**。

(2) **目标程序**。源程序经过编译程序翻译所得到的二进制代码称为目标程序, **目标程序的扩展名为“.obj”**。目标代码尽管已经是机器指令, 但是还不能运行, 因为目标程序还没有解决函数调用问题, 需要将各个目标程序与库函数连接, 才能形成完整的可执行程序。

(3) **可执行程序**。目标程序与库函数连接, 形成完整的可在操作系统下独立执行的程

程序称为可执行程序。可执行程序的扩展名为“.exe”（在 DOS/Windows 环境下）。

如表 1-1 所示是源程序、目标程序和可执行程序三者之间的对照关系表。

可执行程序的运行结果是否正确需要经过验证，如果结果不正确则需要进行调试。调试程序往往比编写程序更困难、更费时间。如图 1-1 所示为 C 程序编辑、编译、连接和运行的全过程。

表 1-1 源程序、目标程序和可执行程序之间的关系

	源程序	目标程序	可执行程序
内容	程序设计语言	机器语言	机器语言
是否可执行	不可以	不可以	可以
文件名后缀	.c	.obj	.exe

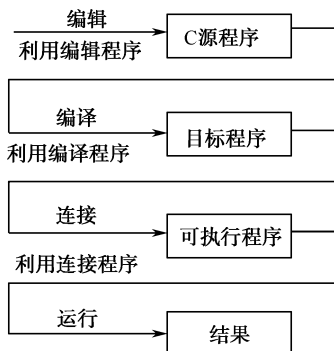


图 1-1 C 程序编辑、编译、连接和运行全过程

1.3 C 语言程序案例



视频讲解

1.3.1 第一个 C 语言程序

下面学习第一个 C 语言程序，通过这个程序可以初步了解 C 语言的简单语法特点和程序结构。

【案例 1-1】在计算机屏幕上输出“Welcome to C Program!”。



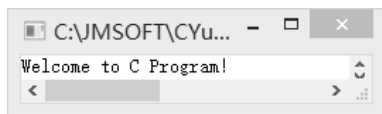
编写程序代码

```

1  /* 案例 1-1 源程序 */           /*注释信息，运行时忽略*/
2  #include "stdio.h"             /*将库文件 stdio.h 包含到该文件中*/
3  void main()                   /*主函数名*/
4  {                             /*main 函数体开始*/
5      printf("Welcome to C Program!\n"); /*在屏幕上输出字符串，\n 是换行符*/
6  }                             /*main 函数体结束*/
  
```



程序运行结果



程序分析

程序中 main 是主函数名，C 语言规定必须用 main 作为主函数名，函数名后的一对圆括号不能省略，圆括号内容可以是空的。一个 C 语言源程序可以有多个函数，但有且只能有一个 main 函数，即主函数。一个 C 程序不管主函数在什么位置，都是从主函数开始执行的。函数体需用花括号括起来，左括号表示函数体的开始，右括号表示函数体的结束。左右括号之间可以有定义部分和执行语句部分；每条语句都必须用分号“;”结束，语句的数量不限。

main 前面的 void 表示主函数的类型是无返回值类型（或称空类型）。该类型不需要用 return 返回一个值。（注意：本书中所有程序都是在 C 语言编程软件“C 与 C++ 程序设计学习与实验系

统”中调试运行的，所以主函数的类型为 void 类型，如果在 TC 程序中可不加类型名 void，直接写 main 即可。)

“#include "stdio.h"”是一条预处理命令，用“#”开头，后面不能加“;”号，stdio.h 是 C 语言系统提供的头文件，文件内部包括输入/输出函数的信息。如果在 TC 环境下，此命令可以不写，但在 VC 环境中必须加上该文件包含命令。

1.3.2 定义符号常量的 C 语言程序

【案例 1-2】从键盘输入一个圆的半径 r，求该圆的面积和周长。



编写程序代码

```
1  /* 案例 1-2 输入半径 r 的值，求圆的周长和面积*/
2  #include "stdio.h"
3  #define PI 3.1415          /*定义一个符号常量 PI，在程序中代替 3.1415*/
4  void main()
5  {
6      double r,s,l;         /*定义三个主函数内部变量 r、s、l*/
7      printf("请输入圆的半径: "); /*在屏幕上输出提示信息*/
8      scanf("%lf",&r);     /*从键盘输入一个实数，%lf 是格式控制符，用来输入实数*/
9      l=2*PI*r;            /*求圆的周长*/
10     s=PI*r*r;            /*求圆的面积*/
11     printf("该圆的周长为%5.2lf，该圆的面积为%5.2lf。",l,s); /*输出圆的周长和面积值*/
12 }
```



程序运行结果



程序分析

有时 C 程序中需要使用在程序运行过程中不需要改变的值，如圆周率，此时可以将这些不变的值设为符号常量。

程序中的第 3 行就是一个符号常量定义，该定义是一个编译预处理命令，在命令后面不能加“;”，具体内容在后面章节有介绍。

一般 C 程序的主函数结构分为四部分：定义变量，输入各变量值（或直接赋值），进行运算，输出结果。

程序中第 6 行为定义变量，第 7 行为输出提示信息（在输入变量语句前一般要先输出提示信息，以方便用户），第 8 行为输入变量值，第 9~10 行为运算求得圆的周长和面积结果，第 11 行为输出结果。

其中，scanf 和 printf 为 C 语言提供的标准输入/输出函数，&r 中的“&”含义是“取地址”，程序中 scanf 函数的功能是从键盘上输入一个实数，输入到变量 r 所标志的内存单元中。如果不想从键盘中读入值，可将程序中第 7、8 行删去，改为如下语句：

```
r=2.4;          /*直接为变量 r 赋初值为 2.4 */
```

也可以为 r 赋值，但 r 的值在程序运行过程中不可以改变。

从键盘输入值和直接赋值的区别在于每次运行程序时从键盘中输入的值可以不同，程序运行结果可以不同；而直接赋值的程序多次运行时变量赋值都为同一值，程序运行结果相同。

1.3.3 有多个函数的 C 语言程序

如果 C 程序中不止一个函数，则必须有一个函数为主函数，执行时主函数会调用子函数，我们通过下面程序来了解有多个函数的 C 程序的结构。

【案例 1-3】从键盘输入两个整数，求这两个整数的和。

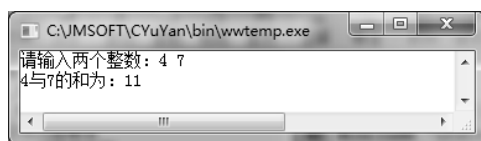


编写程序代码

```
1  /* 案例 1-3 求两个整数的和 */
2  #include "stdio.h"
3  int sum(int a,int b)                /*子函数 sum，其中 a、b 为参数，用来接收数据*/
4  {
5      return a+b;                    /*计算并返回 a 加 b 的和*/
6  }
7  void main()
8  {
9      int x,y,s;                      /*定义主函数内部变量 x、y、s*/
10     printf("请输入两个整数：");    /*在屏幕上输出提示信息*/
11     scanf("%d%d",&x,&y);           /*键盘输入 x 和 y 的值，%d 控制输入格式为十进制整数*/
12     s=sum(x,y);                      /*调用函数 sum，并将结果赋给变量 s*/
13     printf("%d 与 %d 的和为： %d",x,y,s); /*输出两个数的和，三个%d 位置各输出 x、y、s 的值*/
14 }
```



程序运行结果



程序分析

本程序由主函数 main 和子函数 sum 组成，程序从主函数开始执行。在主函数中输入两个整数 x、y，然后通过语句“s=sum(x,y);”调用函数 sum，x 将值赋给子函数中的变量 a，y 将值赋给子函数中的 b，在子函数中计算 a+b 的值，计算结果由 return 语句返回给主函数，并将结果赋给变量 s。然后主函数中第 13 行语句使用 printf 输出这两个数的和。这两个函数在位置上是独立的，可以把主函数放在前面，也可以把主函数放在后面，不影响程序运行结果。函数内容在后面章节中有详细介绍，此处只做简单了解。

1.4 算法

1.4.1 算法概述

计算机解决问题所依据的步骤称为计算机算法，简称算法。编写程序让计算机解决实



视频讲解

际问题，是算法的实现。计算机语言只是一种工具，简单地掌握各种语言的语法规则是不够的，还必须学会针对各类问题提出解题方法和算法。正确的算法应具备以下四个基本特征。

(1) **确定性**。算法中每个操作步骤都应当是明确的，而不应是含糊的、模棱两可的。在计算机算法中不允许有歧义性，所谓“歧义性”是指可以被理解为两种或多种可能的含义。因为计算机至今还没有主动思维的能力，如果给定的条件不确定，计算机就无法执行。

(2) **有效性**。算法中的每一个步骤都应当有效地执行，并得到确定的结果。例如当 $b=0$ 时， a/b 不能被有效执行。

(3) **有穷性**。有穷性是指一个算法的操作步骤必须是有限的、合理的，即在合理的范围之内结束算法。

(4) **输入/输出**。有 0 个或多个输入，有 1 个或多个输出（程序至少有一个输出结果）。

1.4.2 算法的图形表示

在程序设计过程中往往采用图形化设计方法进行概要设计。算法有多种表示方法，最常用的有流程图表示法和 N-S 图表示法。

1. 用流程图表示算法

流程图是用一组框图和符号来表示算法的各种操作，也称框图。用流程图表示算法直观形象，易于理解。美国国家标准化协会（ANSI）规定的一些常用流程图符号，如图 1-2 所示。

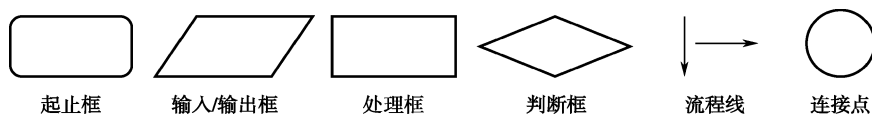


图 1-2 常用流程图符号

结构化程序设计有三种基本结构：顺序结构、选择结构和循环结构，循环结构又分为当型循环和直到型循环两种。三种基本结构流程图如图 1-3 所示。

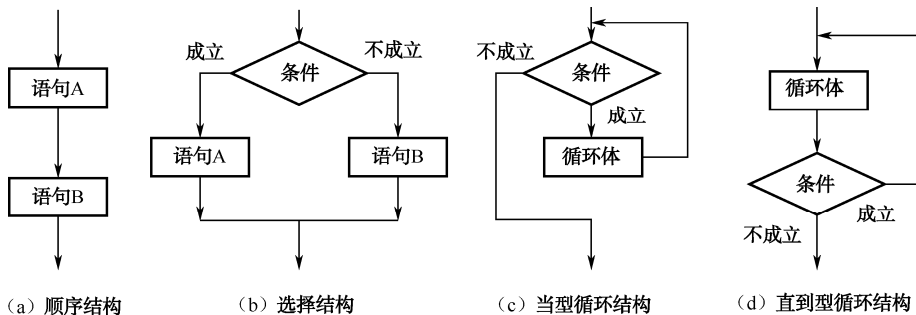


图 1-3 三种基本结构流程图

本书采用传统流程图讲解各程序。

2. 用 N-S 图表示算法

N-S 图是美国学者 I.Nassi 和 B.Shneiderman 提出的一种新的流程图形式（N 和 S 是两位学者的英文姓名的首字母）。在 N-S 图中完全去掉了流程线，全部算法写在一个矩形框内，在该框内还可以包含其他的从属于它的框，即由一些基本框组成一个大框。N-S 图用图 1-4 所示的符号表示三种基本结构。

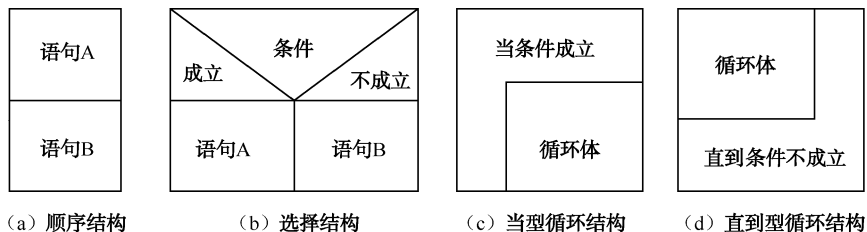


图 1-4 三种基本结构的 N-S 图符号

1.5 C 语言编程环境



视频讲解

1.5.1 Turbo C 2.0 编程环境

1. 安装

因为 Turbo C 2.0 为 DOS 应用程序，所以不用安装，直接复制到某磁盘上就可以直接使用。在磁盘上（例如 C 盘）建立一个 TC 子文件夹，将 TC 系统所有文件复制到该文件夹下即可，其中的 TC.exe 文件为 TC 的系统主程序。

2. 编程环境

在 Windows 的文件夹中双击 TC.exe 文件，进入 Turbo C 2.0 集成开发环境主界面，如图 1-5 所示。其中最上面一行为 Turbo C 2.0 主菜单，中间窗口为编辑区，再下面是信息窗口，最下面一行为参考行。这四个部分构成了 Turbo C 2.0 的主界面，编程、编译、调试及运行都在这个主界面中进行。除“Edit”项外，其他各项均有子菜单，只要按“Alt”键加上某项中的第一个字母，就可进入该项的子菜单。例如，按“Alt+F”组合键弹出“File”下拉菜单，按“↑”“↓”“←”“→”键选择菜单项。

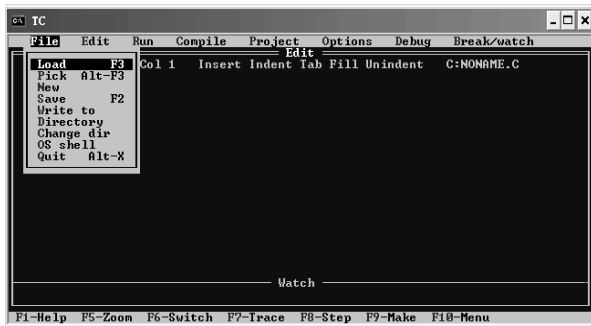


图 1-5 Turbo C 2.0 主界面

TC 环境下，功能键 F1~F10 都是热键，其功能如表 1-2 所示。

表 1-2 TC 环境下热键功能

热 键	功 能	热 键	功 能
F1	激活帮助窗口，显示与当前光标所在位置有关的操作提示信息	F2	将当前文件以指定的文件名存盘

热 键	功 能	热 键	功 能
F3	装入指定文件	F7	调试程序，执行单步操作，可进入被调用函数
F4	将程序执行到光标所在行暂停	F8	调试程序，执行单步操作，不进入被调用函数
F5	缩放当前窗口	F9	编译、连接源程序，生成可执行文件
F6	切换活动窗口	F10	激活主菜单

还有几个常用的快捷组合键如下。

- ❖ Ctrl+F9——运行程序；
- ❖ Alt+F5——用户界面，查看运行结果；
- ❖ Alt+F9（等价于 Compile）——编译；
- ❖ Alt+X——退出 TC；
- ❖ Ctrl+Y——删除光标所在的一行；
- ❖ Alt+F3——选择一个最近打开的文件。

TC 环境中要进行一个重要的配置，选择菜单“Options”下的“Directories”子菜单，其中“Include directories”和“Library directories”所对应的文件夹必须是实际存在的文件夹，最好是 TC 系统所在的文件夹。设置完成后需要立即选择“Options”下的子菜单“Save options”子菜单进行保存操作。配置信息将保存在 TCONFIG.TC 文件中。

提示：TC 安装路径不同影响程序运行

如果开始安装时没有将 TC 系统安装在 C 盘根目录下，例如安装在 D 盘根目录下，这时就需要将“Directories”中的所有路径都改为“D:\TC\”才能正常运行调试程序。

▶ 3. 如何在 TC 中调试程序

首先了解与程序调试相关的菜单项和快捷键。

(1) 设置断点（“Ctrl+F8”组合键或菜单“Break/watch”→“Add watch”）。断点就是要求程序暂停的一行，把光标移到这行，按“Ctrl+F8”组合键，出现红色横条的行就是断点所在行，一个程序中可以设置多个断点。当再次按“Ctrl+F8”组合键时，该断点被取消。

按“Ctrl+F9”组合键运行程序时，在断点处暂停，以便观察。如果在循环中设置断点，循环一次暂停一次。

(2) 单步运行（F7 快捷键或菜单“Run”→“Trace into”）。按一次 F7 快捷键，程序执行一步，然后暂停。一般先运行到设置断点处，再从断点处开始单步运行。

提示：C 语言程序的调试

当有函数的调用时，F7 要跟踪到函数的内部；F8 不跟踪到函数的内部，只把函数当一句话。

(3) 即时计算表达式的值（“Ctrl+F4”组合键或菜单“Debug”→“Evaluate”）。在程序暂停运行时，可以在对话框中输入感兴趣的表达式，看得到的值与预期的是否一致。

(4) 全程监视表达式的值（“Ctrl+F7”组合键或菜单“Break/watch”→“Add watch”）。先按 F5 快捷键打开“监视”（Watch）窗口，再按“Ctrl+F7”组合键，输入要一直监视的表达式，可以在程序单步运行的过程中对每一步的结果进行监视。

如果要清除监视的表达式，选择“Break/watch”→“Clear all breakpoints”命令即可。

1.5.2 Visual C++ 6.0 编程环境

Visual C++ 6.0 是美国微软公司开发的 C++ 集成环境，它集源程序的编写、编译、连接、调试、运行，以及应用程序的文件管理于一体。

这里以 Visual C++ 6.0 为例，讲述 C 程序的上机运行步骤。假设【案例 1-1】源文件命名为 x1_1.c，工程命名为 x1_1（在 VC 中必须建立一个工程），该源文件保存路径为 E:\C\x1_1\。

1. 启动 Visual C++ 6.0 的集成开发环境

选择“开始”→“程序”→“Microsoft Visual Studio 6.0”→“Microsoft Visual C++ 6.0”命令，运行 Visual C++ 6.0，进入 Visual C++ 6.0 集成开发环境窗口。

2. 创建一个空的控制台应用程序

用 AppWizard 创建一个控制台应用程序。

(1) 选择菜单中的“文件”(File)→“新建”(New)命令，打开“新建”对话框。

(2) 单击“工程”(Project)标签，在“工程”选项卡中选择“Win32 Console Application”（Win32 控制台应用程序）项。在“工程”(Project Name)文本框中输入工程名字，如 x1_1。单击“C 位置”文本框后的...按钮，将工程文件定位到“E:\C\x1_1”，如图 1-6 所示。



图 1-6 “新建”对话框

(3) 单击“确定”按钮，在弹出的“Win32 Console Application-Step1 of 1”对话框中选择“An empty project”选项，如图 1-7 所示。

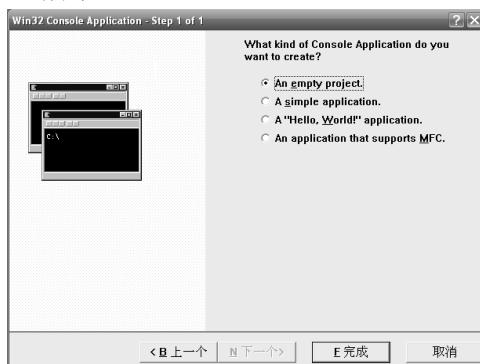


图 1-7 “Win32 Console Application-Step1 of 1”对话框

(4) 单击“完成”按钮，系统将显示向导创建的信息。最后在“新建工程信息”对话框中单击“确定”按钮，完成工程创建过程。

此时为工程 x1_1 创建了 E:\C\x1_1 文件夹，并在其中生成了 **x1_1.dsp**、**x1_1.dsw**、**Debug 文件夹**。Debug 文件夹用于存放编译、连接过程中产生的文件。

▶ 3. 建立 C 源程序文件

在建立完工程之后，工程中没有任何文件，还需要在工程中添加新的源程序文件，步骤如下。

(1) 选择菜单中的“文件”→“新建”命令，打开“新建”对话框，如图 1-8 所示。

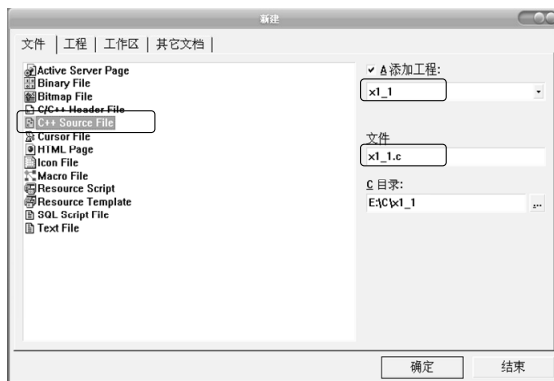


图 1-8 添加新的 C++源程序文件

(2) 在“新建”对话框的“文件”选项卡中选择“C++ Source File”（C++源程序），并在右侧“文件”文本框中填入文件名“x1_1.c”，单击“确定”按钮，完成新建的 C 源程序文件。此时会弹出一个空白的名为“x1_1.c”的代码编辑窗口，允许用户输入程序代码。

▶ 4. 编辑 C 源程序文件

Visual C++ 6.0 集成开发环境包含四个主要区域：菜单和工具栏、工程工作区窗口、代码编辑窗口和输出窗口。

(1) 在代码编辑窗口中输入源程序代码，如图 1-9 所示。

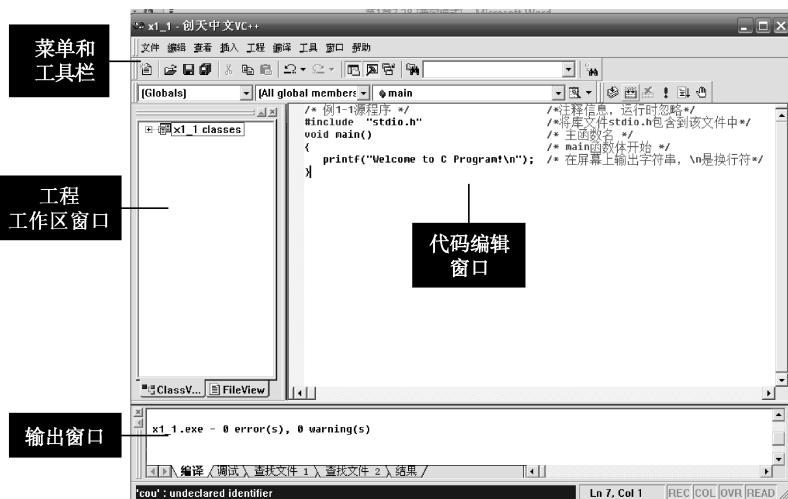











图 1-9 在文件编辑窗口中输入源程序代码

(2) 单击主菜单中的“文件”→“保存”命令保存文件，或是单击工具栏上的保存按钮 。







在代码编辑窗口中所有代码的颜色都会发生改变,这是 Visual C++ 6.0 的文本编辑器所具有的语法颜色功能,绿色表示注释,蓝色表示关键词等。

►5. 建立并运行可执行程序文件

(1) 单击编译工具条      上的生成工具按钮 , 可编译并运行程序。首先,单击工具栏中的“Compile”按钮  (或按“Ctrl+F7”组合键)进行编译,同时在编译窗口中显示编译的有关信息,再单击“Build”按钮  (或直接按快捷键“F7”)建立该应用程序,当出现:

```
x1_1.exe - 0 error(s), 0 warning(s)
```

表示程序 x1_1.exe 可执行文件已经正确无误地生成了。

(2) 单击编译工具条      上的“BuildExecute”按钮  (或直接按“Ctrl+F5”组合键)即可运行刚才生成的 x1_1.exe 文件,并显示运行结果。

►6. 修正语法错误

当程序出现错误时,输出窗口的“编译”页面会显示错误和警告的程序行数。移动“编译”页面的滚动条,使窗口显示出第一条错误信息,双击该错误信息,光标会自动定位在发生错误的代码行上,该行前面标有蓝色箭头,可快速进行错误代码定位。

►7. 退出 Visual C++ 6.0

第一种方法是单击主窗口右上角的“关闭”按钮,第二种方法是选择菜单“文件”→“退出”命令,两种方法都可以退出 Visual C++ 6.0。

►8. 打开已存在工程

选择菜单“文件”→“打开”命令,在弹出的对话框中选定 E:\C\x1_1\x1_1.dsw,单击“打开”按钮,则可打开工作空间,对已建立的工程文件进行修改。

1.5.3 “C/C++程序设计学习与实验系统”编程环境

计算机技术发展得非常快,现在计算机的操作系统大多是 Windows 10,而 Visual C++ 6.0 不能在 Windows 10 中运行,下面就介绍一款非常好用的 C 语言编程软件“C/C++程序设计学习与实验系统”。

这个软件支持最新的操作系统 Windows 10,支持 TC2、TC3、VC6、GCC 四种编译器,支持单步调试,支持重新集成在 Windows 10 系统下正常运行的 Visual C++ 6.0 简化版,支持 Visual C++ 6.0 和 Turbo C 2.0 中英文编译错误信息同步显示功能、Turbo C++ 3.0 常见编译错误信息、C 语言专业词汇的中英文对照。

►1. 安装“C/C++程序设计学习与实验系统”

双击压缩文件 tkc20125.zip (以 2012 版为例),打开 tkc20125 文件夹,双击里面的 setup.exe 安装文件,单击“下一步”按钮,系统默认安装路径为“C:\JMSOFT\CYuYan\”,单击“下一步”按钮,快捷方式为“C 与 C++程序设计学习与实验系统”,单击“下一步”按钮,附加快捷方式,默认为选中“创建桌面快捷方式”,单击“下一步”按钮,开始安装。

安装完毕单击“完成”按钮。此时在桌面上会建立一个“C 与 C++程序设计学习与实验系统”快捷方式,软件安装完成。

2. “C/C++程序设计学习与实验系统”编程环境


在桌面上双击“C与C++程序设计学习与实验系统”图标，打开实验系统界面，如图1-10所示。



图 1-10 “C/C++程序设计学习与实验系统”界面

(1) 菜单栏。菜单栏中包含了开发环境所有的命令，它为用户提供了代码操作、程序的编译、调试、窗口操作等一系列的功能。

(2) 工具栏。通过工具栏，可以迅速地使用常用的菜单命令。


(3) 代码编辑窗口。在此输入C或C++源程序。在实验系统运行后，系统为用户自动生成一个noname0.c的源程序，里面已经自动写好了几行C语言代码。代码第一行为注释，第二行为文件包含命令，第三行为主函数头，左右花括号表示主函数体的开始和结束。一般简单程序只要在主函数花括号内部编写即可。

(4) 输出窗口。用于显示程序调试、连接错误和提示信息。

(5) 系统资源窗口和信息窗口。此开发环境中包含大量C语言资源，如“简单的入门程序实例”中就包含了20多个C和C++的入门小程序。而“C/C++语法参考”中包含了C和C++相关的语法知识点，方便用户在调试程序过程中遇到问题时查看。“函数查询”中提供了常用函数（免费）和大量C语言函数。当在资源窗口中双击相关函数名时，右侧的信息窗口中会显示相关的函数信息。

3. 调试运行C程序

打开实验系统，在代码编辑窗口中输入程序，单击工具栏中的“保存”按钮，弹出“保存”对话框，选择位置，输入源程序文件名，如x1_2.c，单击“保存”按钮，完成保存。

在工具栏上单击“运行”按钮，则弹出“运行”窗口，允许用户输入数据并运行。如果程序有错误，则在输出窗口中会显示错误信息，并自动将光标停在错误行，让用户改正。例如如图1-11中第8行中&r前面缺少逗号，单击“运行”按钮时程序出现错误，该行为反显状态，下面输出窗口中有错误提示信息。

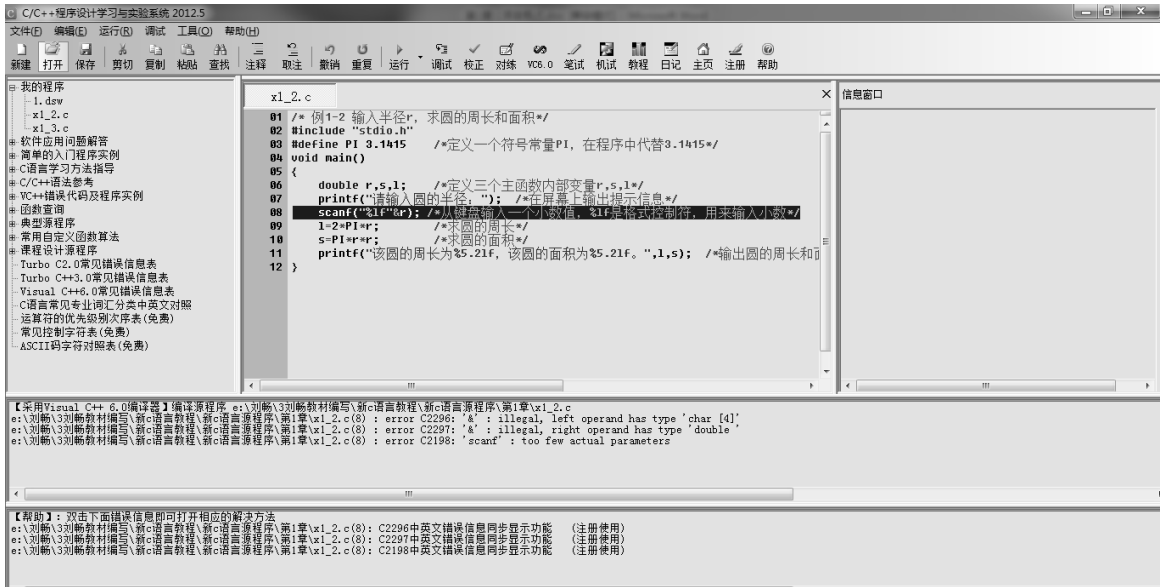


图 1-11 “C/C++程序设计学习与实验系统”调试程序出错时状态

4. 设置编译器选项

实验系统支持 TC2、TC3、VC6、GCC 四种编译器，常用的有 TC2 和 VC6 两种，如何改变编译器呢？下面介绍一下。

选择菜单“工具”→“选项”命令，弹出“选项”对话框，如图 1-12 所示。

在“编译器”中勾选“Turbo C 2.0”，即可以使用 TC 编译程序，若将该选项下面的复选框“TC 支持 WIN7 绘图”选中，则可以运行 TC 中的图形库文件应用程序；如果勾选“Visual C++6.0”，则编译器会选择 VC6。这两种编译器对绝大部分 C 语言程序运行结果是相同的，但有些地方不一样，我们在后面会进行分析讲解。

若在图 1-12 中单击“我的程序文件夹”文本框后面的“设置”按钮，则会弹出对话框，允许用户自定义每次打开的默认文件夹位置，如图 1-13 所示。



图 1-12 “选项”对话框设置编译器界面

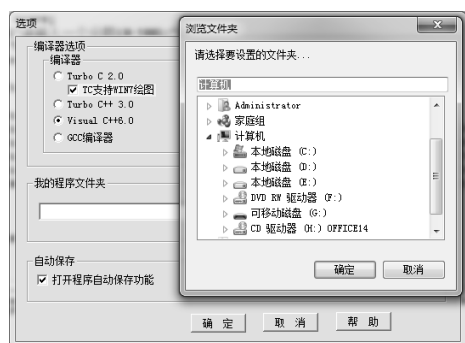


图 1-13 设置默认文件夹路径

提示：C 语言程序的调试

TC 中只能显示英文信息，不支持显示中文信息，如果输出信息有中文会显示为乱码；而 VC 中可以显示中文。所以本教材中有中文显示的程序可以用 VC 编译，默认选项也为 VC。而“TC 支持 WIN7 绘图”选项是为了在 WIN7 中使用 TC 的图形库函数，如果运行带图形函数的程序，则该项需要被选中，如果不带图形库函数程序则可不选中该项。

本章小结

本章主要介绍了 C 语言的基础知识，主要包括程序设计的概念、算法的特点、程序设计算法的图形表示法（包括流程图表示法和 N-S 图表示法两种）、C 语言的特点、Turbo C 2.0、Visual C++6.0、C/C++ 程序设计学习与实验系统三个集成开发环境、上机步骤和常用热键等，读者应熟练掌握各种集成开发环境。从学习角度来说，C/C++ 程序设计学习与实验系统使用起来更加简单、方便。

学生自我完善练习

【上机 1-1】将 Turbo C 2.0 文件夹放在 C 盘根目录下，看一看运行案例 1-1 是什么结果？如果把 Turbo C 2.0 文件夹存放在 D 盘根目录下，又会出现什么结果？

解：当 Turbo C 2.0 不是安装在 C 盘根目录下时，会出现错误，这是因为 C 语言默认的安装路径是在 C 盘根目录下（“C:\tc”），如果不是在这个路径下则找不到系统相关的文件，运行出错。可以将 Turbo C 2.0 系统的目录改为用户安装的目录，如 TC 安装在 D 盘时，可将 TC 的系统默认路径改为“D:\tc”。方法是：按“Alt+O”组合键（或选择“Options”菜单）后选择“Directories”选项，在弹出的窗口中将所有路径中的“C:\tc\”都改为当前 Turbo C 2.0 的安装路径（如“D:\tc\”），再次运行程序就不会出错了。

【上机 1-2】如果运行结果如下图所示，则该程序如何编写？



解：通过案例 1-1 可知该程序是在屏幕上输出三行字符串，应该使用输出函数 `printf`，又因为显示为三行字符串，所以输出字符串后应加换行符 `\n`，源程序如下。



编写程序代码

```
1 #include "stdio.h"
2 void main()
3 {
4     printf("*****\n");
5     printf("* 欢迎来到 C 世界! *\n");
6     printf("*****\n");
7 }
```

习题一

一、选择题

1. 以下_____不是 C 语言的特点。
A) 生成目标代码质量高
B) 语法限制严格
C) 数据类型丰富
D) 控制流程结构化
2. 以下_____不是算法的特征之一。

- A) 有穷性 B) 确定性 C) 有效性 D) 复杂性

3. 以下_____不是二进制代码文件。

- A) 标准库文件 B) 目标文件 C) 源程序文件 D) 可执行文件

4. C 语言程序的基本单位是_____。

- A) 程序行 B) 语句 C) 函数 D) 字符

5. C 语言可执行程序的开始执行点是_____。

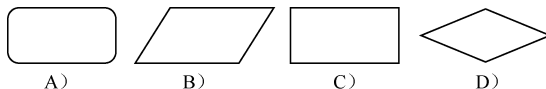
- A) 程序中第一条可执行语句 B) 程序中第一个函数
C) 程序中的 main 函数 D) 包含文件中的第一个函数

6. 一个完整的 C 源程序是_____。

- A) 要由一个主函数或一个以上的非主函数构成
B) 由一个且仅由一个主函数和零个以上（含零）的非主函数构成
C) 要由一个主函数和一个以上的非主函数构成
D) 由一个且只有一个主函数或多个非主函数构成

7. 以下_____是流程图的判断框，以下_____是流程图的起止框，以下_____是流程图的处理框，

以下_____是流程图的输入/输出框。



8. 以下叙述中错误的是_____。

- A) C 语言的可执行程序是由一系列机器指令构成的
B) 用 C 语言编写的源程序不能直接在计算机上运行
C) 通过编译得到的二进制目标程序需要连接才可以运行
D) 在没有安装 C 语言集成开发环境的机器上不能运行 C 源程序生成的.exe 文件

9. 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

二、填空题

1. 函数体以符号_____开始，以符号_____结束。
2. 一个完整的 C 程序至少要有_____函数。
3. 程序设计语言可以分为_____、_____、_____。
4. C 程序是以_____为基本单位的，整个程序由_____组成。
5. C 源程序文件的扩展名是_____，C 目标文件的扩展名是_____，C 可执行文件的扩展名是_____。
6. 程序连接过程是将目标程序、_____或其他目标程序连接装配成可执行文件。
7. 在一个 C 源程序中，注释部分两侧的分界符分别为_____和_____。
8. 一个函数是由两部分组成的，它们分别是_____和_____。
9. 在 C 语言中，一个函数的函数体一般包括_____和_____。



习题一参考答案