

劳动和社会保障部职业技能鉴定推荐教材

21 世纪高等职业教育
规划教材
双证系列

C 语言与数据结构

主 编 梁西陈 沈学东
副主编 吉顺如

上海交通大学出版社

内 容 提 要

本教材将计算机专业的两个骨干课程“C 语言程序设计”和“数据结构”的内容融为一体,共分 12 章,以 C 语言为主线介绍了 C 语言与数据结构的基本概念、C 语言的数据类型、C 语言基本操作和语法规则;利用 C 语言进行结构化和模块化程序设计的基本方法;线性表、树和图的数据结构及应用;排序及查找的常用算法等内容。

本书在选材上力求精练,问题描述深入浅出,并通过大量例题进行说明。每章后都配有习题和内容详实的实验,供课后练习及上机使用。例题和实验内容均给出详细的程序源代码,输入计算机后即可调试运行。

本书既可作为高、中等职业院校计算机及相关专业的教材,也可作为广大计算机工作者的参考用书。

前 言

以就业为导向的专业和课程改革是当前职业院校教育改革的热点问题之一。职业院校倡导的“学历证书”和“职业证书”并重的观念,有助于培养具有一定理论知识并熟练掌握职业基本技能、综合能力较强的技能型人才,对于增强毕业生就业能力具有极其重要的意义。以提高学生的职业实践能力和职业素养为宗旨、以就业为导向、以能力为本位、以产业(行业)需求为目标构建专业培养方案已成为专家、学者、用人单位、职业院校师生的共识。本教材正是在上述思想指导下,组织多位多年从事职业教育一线工作的专家、教师编写而成的。

本教材将计算机专业的两个骨干课程“C 语言程序设计”和“数据结构”的内容融为一体,共分 12 章,以 C 语言为主线介绍了 C 语言与数据结构的基本概念、C 语言的数据类型、C 语言基本操作和语法规则;利用 C 语言进行结构化和模块化程序设计的基本方法;线性表、树和图的数据结构及应用;排序及查找的常用算法等内容。

按照基础理论课以“必需、够用”为度、服务于实际应用的要求,教材中收集和编写了大量的程序,由浅入深地培养学生的程序分析和设计能力。在选材上力求精练,问题描述深入浅出,并通过大量例题进行说明。每章后都配有习题和内容详实的实验,供课后练习及上机使用。例题和实验内容均给出详细的程序源代码,输入计算机后即可调试运行。

本书特意在第 10 章“图”中,使用了 C++ 语言中的两个语法符号,即注释符号“//”和引用参数符号“&”,以便使读者对 C++ 语言有所了解,为以后学习 C++ 语言打下基础。读者可以直接在 Turbo C++ 的编译环境下调试运行本章所提供的所有程序或稍加修改后在 C 语言的编译环境下调试运行。

本书由梁西陈同志主编,并由梁西陈、沈学东、吉顺如同志共同负责统稿和定稿。其中,第 1、11、12 章由梁西陈同志编写,第 6、9 章由沈学东同志编写,第 4 章由吉顺如同志编写,第 3、8 章由张勇同志编写,第 2、7 章由胡秀建同志编写,第 5、10 章由滕尚峰同志编写。

本书的编写得到了劳动和社会保障部、上海交通大学出版社、宿州教育学院、上海电机学院和宿州职业技术学院等多家单位的支持,附录的收集和整理得到了罗金星同志的帮助,在此一并表示感谢。

读者在使用过程中如发现错误或不当之处,请及时与编者联系,以便再版时更正。

编者

2007 年 3 月 26 日

目 录

第 1 章 概述	1
1.1 C 语言简介	1
1.2 C 程序的基本结构	3
1.3 C 程序上机步骤	4
1.4 数据结构概述	10
本章小结	13
实验一 C 语言程序的编辑与运行	13
习题一	13
第 2 章 数据类型与基本操作	15
2.1 C 语言的数据类型	15
2.2 常量和变量	16
2.3 运算符和表达式	24
2.4 基本输入与输出	30
本章小结	35
实验二 格式化输入、输出	35
实验三 字符的输入、输出	36
习题二	37
第 3 章 C 语言程序的流程控制	39
3.1 程序设计的基础知识	39
3.2 C 语言的基本语句	42
3.3 选择结构程序设计	43
3.4 循环结构程序设计	50
本章小结	57
实验四 顺序、选择结构程序设计	58
实验五 循环结构程序设计	59
习题三	60
第 4 章 函数	63
4.1 函数的定义及调用	63
4.2 变量的存储属性	71
4.3 内部函数和外部函数	78

4.4 C 语言的编译预处理	79
本章小结	81
实验六 函数的定义及应用	82
习题四	83
第 5 章 数组数据类型及其应用	87
5.1 一维数组	87
5.2 二维数组	90
5.3 字符数组与字符串	92
5.4 线性表	98
5.5 栈和队列	101
本章小结	105
实验七 数组及线性表	105
实验八 栈及其应用	106
实验九 队列及其应用	107
习题五	108
第 6 章 查找与排序	111
6.1 常用数据查找算法	111
6.2 常用数据排序算法	116
本章小结	125
实验十 数据的查找与排序	126
习题六	136
第 7 章 指针数据类型及其应用	139
7.1 指针的概念	139
7.2 指针变量的定义和引用	140
7.3 指针与数组	142
7.4 指针与字符串	149
7.5 指针与函数	151
7.6 链表	153
7.7 链栈和链队列	161
本章小结	164
实验十一 指针的使用	164
实验十二 链表的使用	166
习题七	166
第 8 章 结构体与共用体	169
8.1 结构体数据类型的定义和引用	169

8.2 结构体数组及初始化	175
8.3 结构体与指针	176
8.4 结构体与函数	179
8.5 共用体数据类型	183
8.6 枚举类型	185
本章小结	187
实验十三 结构体与共用体的应用	187
习题八	188
第9章 树	192
9.1 树的基本概念	192
9.2 树的存储结构	193
9.3 二叉树	194
9.4 二叉树的遍历	199
9.5 二叉查找树	201
9.6 堆排序	204
本章小结	206
实验十四 二叉树的遍历	207
习题九	213
第10章 图	216
10.1 图的基本概念	216
10.2 图的存储结构	216
10.3 图的遍历	220
10.4 拓扑排序	222
10.5 最短路径	224
本章小结	227
实验十五 图的遍历	228
习题十	229
第11章 文件	231
11.1 C文件概述	231
11.2 文件的打开与关闭	232
11.3 文件的输入与输出	233
11.4 文件的定位与随机读写	238
本章小结	239
实验十六 文件操作	239
习题十一	240

第 12 章 位运算	242
12.1 位运算符.....	242
12.2 位段.....	244
本章小结.....	246
实验十七 位运算.....	246
习题十二.....	247
附录一 部分习题参考答案.....	249
附录二 ASC II 码表.....	253

第 1 章 概述

21 世纪,以计算机技术和网络技术为代表的信息技术得到了迅猛发展,计算机在社会的各个领域得到了广泛的应用,深刻地影响着人们的生产、生活和学习。而在实际应用中,要使计算机发挥作用就离不开程序设计和程序设计语言。程序设计语言是人与计算机进行信息交流的工具。程序设计要在一定的程序设计语言环境下进行。

C 语言是当今世界广泛流行的高级计算机语言之一,它既适合系统程序的设计,也适合应用程序的设计。

数据结构是随着计算机的发展而发展起来的一门计算机学科,是设计和实现编译程序、操作系统、数据库系统以及其他系统程序和应用程序的重要基础。

1.1 C 语言简介

1.1.1 计算机程序设计语言

所谓程序就是计算机所执行的一组指令,而描述这组指令的语言就是程序设计语言。自 1946 年第一台电子计算机诞生以来,计算机程序设计语言的发展经历了三个阶段。

(1) 机器语言。

一种 CPU 所具有的全部指令的集构成该 CPU 的指令系统,它是该 CPU 可以识别的一组由 0 和 1 序列构成的指令代码。下面是某 CPU 指令系统中的两条指令:

10000000	表示两数相加
10010000	表示两数相减

每种 CPU 的指令系统就构成了该机器的机器语言,不同的机器系统有不同的机器语言。

机器语言程序就是为了让计算机完成特定的任务而按一定顺序组织到一起的机器指令序列。计算机的工作过程就是 CPU 周而复始地从内存储器中取得指令、执行指令的过程。

用机器语言编写的程序可直接被计算机理解并执行。但是这种用 0、1 代码序列组成的程序,难记、难认、难理解、不易查错,只能被少数的专业人员掌握。另外,这种程序的生产效率低,质量难以保证。

(2) 汇编语言。

由于机器语言在编程过程中难读、易错,人们想到用一种帮助记忆的符号来代替机器指令。例如上面的两条指令可表示为:

10000000	ADD	A,B
10010000	SUB	A,B

这种用助记符构成的指令系统,就称为符号语言或汇编语言。

用汇编语言编写的程序就称为汇编语言程序。汇编语言程序容易被人们所理解,而计算机却不能直接识别。要让计算机接受汇编语言程序还需要有一个将汇编指令翻译成机器指令

的过程,这个过程称为汇编,能完成这项任务的程序称为汇编程序。

汇编语言程序 $\xrightarrow{\text{汇编程序}}$ 机器语言程序

汇编语言和机器语言都是依 CPU 的不同而不同,它们都是面向机器的语言。使用这两种语言都需要对计算机的内部结构有较深刻的了解,因而用它们编写程序不仅劳动强度大而且程序的通用性差,难以维护。

(3) 高级语言。

高级语言是用接近自然语言和数学公式的形式编程的计算机语言。它脱离了机器的硬件系统,用人们更易于理解的方式编写程序,所以说高级语言是面向科学计算和实际问题的语言。

由于计算机本身只能识别 0、1 代码即机器语言,所以高级语言必须翻译成机器语言才能被计算机所接受和执行,这一过程称为编译。

高级语言程序 $\xrightarrow{\text{编译程序}}$ 机器语言程序

高级语言又可分为面向过程的语言和面向对象的语言。

高级语言(如 FORTRAN 语言和 C 语言)的出现,使人们开始摆脱进行程序设计必须先熟悉机器的桎梏,把精力集中于解题思路和方法上,使程序设计语言开始与解题方法相结合。其中一种方法是把解题过程看作是数据加工的过程。基于这种方法的程序设计语言称为面向过程的程序设计语言。C 语言就是一种面向过程的程序设计语言。

实际上,程序是对现实世界的运动状态的模拟。面向过程的程序设计认为,每个程序都要完成一些规定的功能。每个功能的实现是通过对数据进行一系列的加工的过程而实现的。因此程序设计包括组织数据即设计数据结构,以及设计对数据结构进行加工的过程即设计算法两部分。因此有以下等式:

$$\text{程序} = \text{数据结构} + \text{算法}$$

面向对象的程序设计是一种结构模拟方法,它认为客观世界是由各种各样的对象所组成的,每种对象都有各自的内部状态和运动规律,不同对象之间通过互相发送和接收消息进行联系;消息激发对象本身的运动,形成对象状态的变化。另外,客观世界中的对象都是可以分类的。也就是说,所有的对象都属于特定的“类”(class),或者说,所有的对象都是类的一个实例。因而,面向对象的程序设计的一个关键是定义“类”,并由“类”生成“对象”。因此有以下等式:

$$\text{程序} = \text{对象} + \text{消息}$$

面向对象的程序比面向过程的程序更清晰、易懂,更适宜编写更大规模的程序。面向对象的程序设计语言有 VB、JAVA、C++ 等。需要说明的是 C++ 语言是在 C 语言的基础上发展起来的,它是一种多范型的程序设计语言,不仅可以用来编写面向对象的程序,还可以用来编写面向过程的程序。

1.1.2 C 语言的发展与特点

(1) C 语言的发展。

C 语言是 1972 年由美国的 Dennis Ritchie 和 Brian Kernighan 设计发明的。它由早期的编程语言 BCPL(Basic Combind Programming Language)发展演变而来。随着微机的日益普

及,出现了许多 C 语言版本,由于没有统一的标准,使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况,美国国家标准化协会(ANSI)于 1983 年制定了一套标准,称为 ANSI C(标准 C)成为各种 C 语言版本的基础。

目前在微机中使用的 C 语言版本很多,例如 Turbo C、Borland C、MicroSoft C、Turbo C++等,它们都支持 ANSI C。本书将以 ANSI C 为基础,同时兼顾各种版本的通用性和一致性予以介绍。

(2) C 语言的特点。

C 语言发展迅速、功能强大,因而受到广泛欢迎。C 语言具有下列特点:

① C 语言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,既具有高级语言的通用性及易写易读的特点,又具有低级语言直接操作硬件的能力,这使得 C 语言不仅可用于应用软件的开发,而且可用于系统软件的开发。因此又被称为中级语言。

② C 语言是一种结构化程序设计语言。这种结构化方式可使程序层次清晰,便于使用、维护以及调试。C 语言是以函数形式提供给用户的,这些函数可方便地调用,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。

③ C 语言功能齐全。C 语言具有丰富的数据类型及各种功能强大的系统函数。程序执行效率高。

④ 可移植性好。C 语言的一个突出优点就是适合于多种操作系统,用 C 语言编写的程序可以从一种环境不加或稍加改动就可以转到另一种环境中运行。

1.2 C 程序的基本结构

每种程序设计语言都有其特定的语法要求和程序结构风格。本节通过两个简单的 C 语言程序,使大家对 C 程序的结构有一个总体的认识。

1.2.1 简单的 C 程序示例

例 1.1 一个简单的 C 程序

```
main()
{printf("Hello World !");
}
```

本程序执行后,将在输出窗口中显示如下一行信息:

Hello World !

程序中 main()表示“主函数”,这是每个 C 程序所必需的。{}中的内容为函数体。函数体中只包含一条输出语句 printf,双引号中的内容被原样输出到屏幕上。

例 1.2 从键盘上输入三个整数,然后求它们的和并输出。

```
int sum(int x,int y,int z)          /* 该函数求 x、y、z 三个整数的和 */
```

```

{int s;
 s=x+y+z;
 return(s);
}
main() /* 该程序的主函数,程序运行时将从这里开始 */
{int a,b,c;
 int m;
 scanf ("%d%d%d",&a,&b,&c); /* 输入三个整数 */
 m=sum(a,b,c); /* 调用求和函数 sum() */
 printf("sum=%d\n",m);
}

```

程序执行结果如下:

```

 2  4  6 ✓
-----
sum=12

```

本例题比例 1.1 复杂得多,程序中用到了函数的调用。被调用函数 `sum()` 的作用是求三个形式参数 `x`、`y` 和 `z` 的和。程序的执行仍从 `main()` 函数开始,其执行过程是这样的:首先等待用户从键盘输入数据,假如输入 2 4 6,程序则将这三个数分别送给变量 `a`、`b`、`c`,然后调用函数 `sum()`,并将 `a`、`b`、`c` 的值分别传送给 `x`、`y` 和 `z`,由函数 `sum()` 计算出这三个数的和,通过 `return` 语句将此值返回到主函数 `main()` 中的 `m` 中,最后输出 `m` 的值。

注: `/* …… */` 为 C 语言中的注释语句,其作用是便于读者的阅读和理解,注释语句并不影响程序的执行。

1.2.2 C 程序结构的特点

通过以上两个例子,可以总结出 C 语言程序有以下几个特点:

(1) C 程序由若干个函数组成。每个程序必须有一个主函数即 `main()` 函数。C 程序的执行总是先从 `main()` 函数开始。

(2) 每个函数均由两部分构成,即函数头部和函数体。函数头部一般包括函数返回值类型、函数名及函数参数。函数体由包含在 `{……}` 内的一系列 C 语句组成。

(3) C 程序书写格式自由,一行内可以写几个语句,一个语句也可以分写在几行上。每个语句后都要有一个“;”号。

(4) 可以用 `/* …… */` 在程序的任何位置加注释说明。注释并不影响程序的执行,其作用是便于人们阅读和理解程序。

1.3 C 程序上机步骤

程序编写好之后,还要拿到计算机上去执行,才能得到结果。

C 语言是一种高级语言,用户用 C 语言编写的程序称为源程序,其扩展名为“.C”。计算

机不能直接执行源程序,必须将源程序翻译成二进制目标程序。翻译工作由编译程序完成,翻译的过程称为编译,编译的结果称为目标程序,其扩展名为“.OBJ”。源程序翻译成目标程序后,还要进行连接,连接的目的是将目标程序和库函数或其他程序连接起来,形成可执行的机器代码,其扩展名为“.EXE”。

目前微机上使用的C语言版本很多,但只要掌握其中一种使用方法,对于其他版本,稍加认识便能掌握其使用方法。

C语言源程序在计算机上调试运行需要有C程序的编译环境。这里主要介绍Turbo C 2.0及Turbo C++ 3.0系统的使用方法。另外,Visual C++系统也可用于C程序的开发。对于初学者,建议使用Turbo C 2.0或Turbo C++ 3.0系统作为学习和开发C语言的环境。

1.3.1 Turbo C 2.0的集成环境

Turbo C 2.0是美国Borland公司1989年推出的产品。该系统将文本编辑、程序编译、连接以及程序运行一体化,大大方便了程序的开发。另外,Turbo C 2.0具有查错功能,并可在Tiny模式下直接生成.com文件。

Turbo C 2.0是在DOS环境下运行的C程序的集成开发环境,但在Windows系列操作系统(包括Win2000、WinXP等)下也可运行,只要双击Turbo C 2.0安装目录下的TC.EXE文件即可进入Turbo C 2.0的集成开发环境。需要说明的是Turbo C 2.0在Windows系列操作系统下不支持鼠标操作,只能使用键盘操作。

Turbo C 2.0的主窗口如图1.1所示。主窗口上半部分包括标题栏和菜单栏,中间窗口为编辑区,接下来是信息窗口,最底一行为功能键参考行。这四部分构成了Turbo C 2.0的主屏幕,以后的编程、编译、调试及运行都将在这个主屏幕中进行。

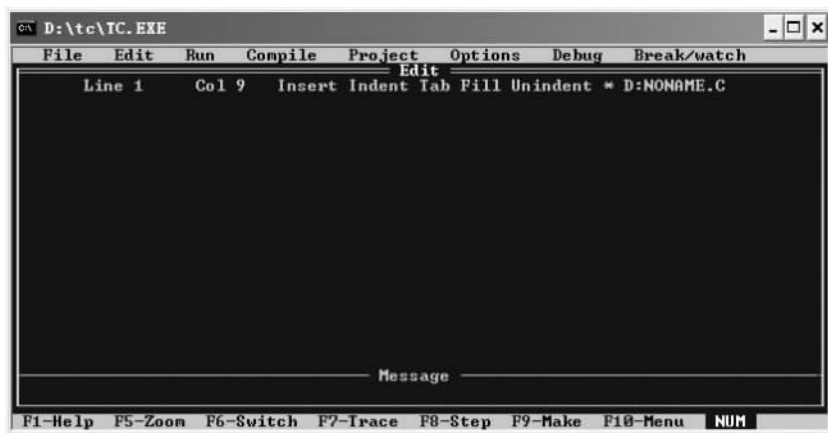


图 1.1 Turbo C 2.0的主窗口

下面对各个主菜单进行详细介绍。

(1) File(文件)。

按<Alt+F>组合键可进入File菜单,该菜单所包含的内容如表1.1所示。

表 1.1 File 菜单内容

菜单项	功能及说明
Load(加载)	装入一个文件,可用类似 DOS 的通配符(如 *.C)来进行列表选择。也可装入其他扩展名的文件,只要给出文件名(或只给路径)即可。该项的热键为 F3 键即只要在主菜单中按 F3 即可进入该项,而不需要先进入 File 菜单再选此项
Pick(选择)	将最近装入编辑窗口的 8 个文件列成一个表让用户选择,选择后将该程序装入编辑区,并将光标置在上次修改过的地方。其热键为<Alt+F3>
New(新文件)	说明文件是新的,缺省文件名为 NONAME.C,存盘时可改名
Save(存盘)	将编辑区中的文件存盘,若文件名是 NONAME.C 时,将询问是否更改文件名,其热键为 F2
Write to(存盘)	可由用户给出文件名将编辑区中的文件存盘,若该文件已存在,则询问要不要覆盖
Directory(目录)	显示目录及目录中的文件,并可由用户选择
Change dir(改变目录)	显示当前目录,用户可以改变显示的目录
Os shell(暂时退出)	暂时退出 Turbo C 2.0 到 DOS 提示符下,此时可以运行 DOS 命令,若想回到 Turbo C 2.0 中,只要在 DOS 状态下键入 EXIT 即可
Quit(退出)	退出 Turbo C 2.0,返回到 DOS 操作系统中,其热键为<Alt+X>

说明:以上各项可用光标键移动色棒进行选择,按回车键则执行。也可用每一项的第一个大写字母直接选择。若要退到主菜单或从它的下一级菜单列表框退回均可用 Esc 键, Turbo C 2.0 所有菜单均采用这种方法进行操作,以下不再说明。

(2) Edit(编辑)。

按<Alt+E>组合键可进入编辑菜单,若再回车,则光标出现在编辑窗口,此时用户可以进行文本编辑。编辑方法基本与 Windows 操作系统中的“记事本”相同,可用 F1 键获得有关编辑方法的帮助信息。

与编辑有关的功能键如下:

F1 获得 Turbo C 2.0 编辑命令的帮助信息;

F5 扩大编辑窗口到整个屏幕;

F6 在编辑窗口与信息窗口之间进行切换;

F10 从编辑窗口转到主菜单。

(3) Run(运行)。

按<Alt+R>组合键可进入 Run 菜单,该菜单所包含的内容如表 1.2 所示。

表 1.2 Run 菜单内容

菜单项	功能及说明
Run(运行程序)	运行由 Project/Project name 项指定的文件名或当前编辑区的文件。如果对上 次编译后的源代码未做过修改,则直接运行到下一个断点(没有断点则运行到结束)。否则先进行编译、连接后才运行,其热键为<Ctrl+F9>
Program reset(程序重启)	中止当前的调试,释放分给程序的空间,其热键为<Ctrl+F2>
Go to cursor(运行到光标处)	调试程序时使用,选择该项可使程序运行到光标所在行。光标所在行必须为一条可执行语句,否则提示错误。其热键为 F4

(续表)

菜单项	功能及说明
Trace into(跟踪进入)	在执行一条调用其他用户定义的子函数时,若用 Trace into 项,则执行长条将跟踪到该子函数内部去执行,其热键为 F7
Step over(单步执行)	执行当前函数的下一条语句,即使用户函数调用,执行长条也不会跟踪进函数内部,其热键为 F8
User screen(用户屏幕)	显示程序运行时在屏幕上显示的结果。其热键为<Alt+F5>

(4) Compile(编译)。

按<Alt+C>组合键可进入 Compile 菜单,该菜单所包含的内容如表 1.3 所示。

表 1.3 Compile 菜单内容

菜单项	功能及说明
Compile to OBJ (编译生成目标码)	将一个 C 源文件编译生成 .OBJ 目标文件,同时显示生成的文件名。其热键为 <Alt+F9>
Make EXE file (生成执行文件)	此命令生成一个 .EXE 的文件,并显示生成的 .EXE 文件名。其中 .EXE 文件名是下面几项之一: a. 由 Project/Project name 说明的项目文件名。 b. 若没有项目文件名,则由 Primary C file 说明的源文件。 c. 若以上两项都没有文件名,则为当前窗口的文件名
Link EXE file (连接生成执行文件)	把当前 .OBJ 文件及库文件连接在一起生成 .EXE 文件
Build all(建立所有文件)	重新编译项目里的所有文件,并进行装配生成 .EXE 文件
Primary C file(主 C 文件)	当在该项中指定了主文件后,在以后的编译中,如没有项目文件名则编译此项中规定的主 C 文件,如果编译中有错误,则将此文件调入编辑窗口,不管目前窗口中是不是主 C 文件
Get info(获得信息)	获得有关当前路径、源文件名、源文件字节大小、编译中的错误数目、可用空间等信息

(5) Project(项目)。

按<Alt+P>组合键可进入 Project 菜单,该菜单所包含的内容如表 1.4 所示。

表 1.4 Project 菜单内容

菜单项	功能及说明
Project name(项目名)	项目名具有 .PRJ 的扩展名,其中包括将要编译、连接的文件名
Break make on(中止编译)	由用户选择是否在有 Warning(警告)、Errors(错误)、Fatal Errors(致命错误)时或 Link(连接)之前退出 Make 编译
Auto dependencies(自动依赖)	当开关置为 on,编译时将检查源文件与对应的 .OBJ 文件日期和时间,否则不进行检查
Clear project(清除项目文件)	清除 Project/Project name 中的项目文件名
Remove messages(删除信息)	把错误信息从信息窗口中清除掉

(6) Options(选项)。

按<Alt+O>组合键可进入 Options 菜单,该菜单对初学者来说要谨慎使用。该菜单所包含的内容如表 1.5 所示。

表 1.5 Options 菜单内容

菜单项	功能及说明
Compiler(编译器)	本项选择又有许多子菜单,可以让用户选择硬件配置、存储模型、调试技术、代码优化、对话信息控制和宏定义
Linker(连接器)	本菜单设置有关连接的选择项
Environment(环境)	本菜单规定是否对某些文件自动存盘及制表键和屏幕大小的设置
Directories(路径)	规定编译、连接所需文件的路径,有下列各项: Include directories 包含文件的路径,多个子目录用“;”分开。 Library directories 库文件路径,多个子目录用“;”分开。 Output directory 输出文件(.OBJ, .EXE, .MAP 文件)的目录。 Turbo C directory Turbo C 所在的目录。 Pick file name 定义加载的 pick 文件名,如不定义则从 current pick file 中取
Arguments(命令行参数)	允许用户使用命令行参数
Save options(存储配置)	保存所有选择的编译、连接、调试和项目到配置文件中,缺省的配置文件为 TCCONFIG.TC
Retrieve options(重载)	装入一个配置文件到 TC 中,TC 将使用该文件的选择项

说明:在首次使用 Turbo c 2.0 时,要注意“Directories”选项的内容是否与文件的实际存放路径相符。如不相符,则需要对“Directories”选项的设置进行修改,即将光标移至要修改的项,按回车键,然后按屏幕提示输入正确的路径即可。在首次使用 Turbo C++ 3.0 系统时也要作同样的设置。

(7) Debug(调试)。

按<Alt+D>组合键可进入 Debug 菜单,该菜单所包含的内容如表 1.6 所示。

表 1.6 Debug 菜单内容

菜单项	功能及说明
Evaluate(计算)	可以测试甚至修改一个变量或表达式的值。注意:表达式不能包含函数调用和宏
Call Stack(调用栈)	用来跟踪当前函数的调用关系,它弹出一个包含调用栈的窗口
Find Function(查找函数)	顾名思义,就是查找函数定义。只有在运行调试时可用
Refresh Display(刷新屏幕)	如果编辑屏被你的程序改写了,可以用它恢复
Display Swapping(屏幕切换)	控制是否在程序运行时切换屏幕
Source Debugging(源码调试)	打开源代码调试

(8) Break/watch(断点/监视)。

按<Alt+B>组合键可进入 Break/watch 菜单,该菜单所包含的内容如表 1.7 所示。

表 1.7 Break/watch 菜单内容

菜单项	功能及说明
Add Watch(添加监视项)	添加监视项。可以监视一个变量或表达式的值
Delete Watch(删除监视项)	删除监视项
Edit Watch(编辑监视项)	编辑你所监视的表达式
Remove All Watches(删除所有监视项)	删除所有监视项
Toggle Breakpoint(设置或删除断点)	如果设置了断点,程序运行到这一行就会停下来
Clear All Breakpoints(清除所有断点)	清除所有断点
View Next Breakpoint(移到下一断点)	按设置顺序移到下一个断点处

1.3.2 Turbo C++ 3.0 的集成环境

为了适用 Microsoft 公司的 Windows 系列操作系统, Borland 公司又推出了面向对象的程序软件包 Turbo C++。它继承、发展了 Turbo C 2.0 的集成开发环境, 并包含了面向对象的基本思想和设计方法。上面已经对 Turbo C 2.0 的各个菜单命令进行了详细的介绍, 接下来只对 Turbo C++ 3.0 与 Turbo C 2.0 作以简单的比较, 相信读者能够举一反三, 很快熟悉它的集成环境。Turbo C++ 3.0 的主窗口如图 1.2 所示。

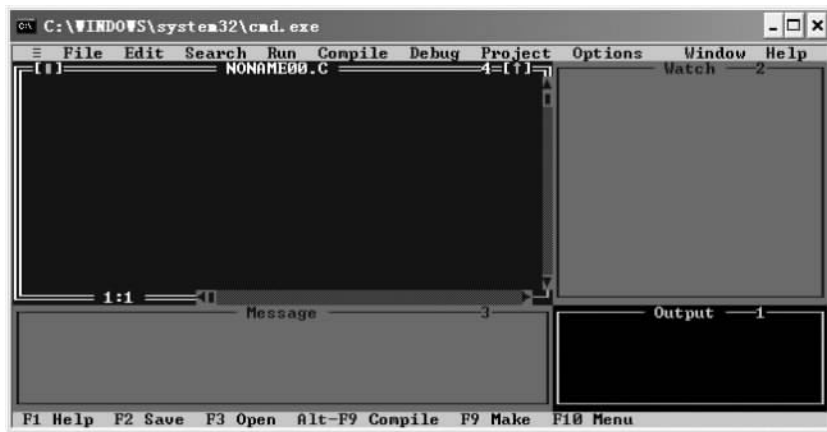


图 1.2 Turbo C++ 3.0 的主窗口

Turbo C++ 3.0 与 Turbo C 2.0 的主要区别:

(1) Turbo C++ 3.0 系统带有两种编译器, 不仅能设计和编译 C 程序文件, 而且能够设计和编译 C++ 程序文件。具体使用时只要把 Options/Compiler/C++ options 菜单项下的 Use C++ Compiler 项的值设为“CPP extension”, 系统将自动根据程序的扩展名不同而调用相应的编译器。

(2) Turbo C++ 3.0 支持多窗口操作, 窗口间可以快速切换。

(3) 完全支持鼠标选择、拖放和右键操作, 很好地照顾了习惯于图形操作环境的用户。

(4) 建立了即时帮助系统, 只需要选定关键字后按 <Ctrl+F1> 组合键即可查看详细的帮助说明, 并且每个函数都具有完整的示例解释说明, 只需要复制到新文件即可运行, 无论对

C 语言初学者还是对 C++ 开发者都是不错的实例教材。

- (5) 语句按照语法高亮多色显示,令代码编写、程序查错更加直观方便。
- (6) 程序编辑器的查找、替换等编辑功能更方便易用。
- (7) 建立和管理 Project 项目更方便容易。

1.3.3 Turbo C/C++ 中编辑及运行 C 程序的步骤

说明:这里“Turbo C/C++”的写法是 Turbo C 2.0 和 Turbo C++ 3.0 的缩写形式,以后不再说明。

C 语言源程序必须经过编译,转换为二进制的目标代码后才能在计算机上运行。从编写好一个 C 程序到完成运行一般要经过以下几个步骤。

(1) 编辑。所谓编辑,包括以下内容:将源程序逐字符输入到计算机中;修改源程序;将修改好的源程序保存。在 Turbo C/C++ 系统中可完成编辑源程序的所有操作。

(2) 编译。就是将已编辑好的源程序翻译成二进制的目标代码。在编译的过程中,编译系统将对源程序进行语法检查,如发现有错,则在屏幕上显示出错信息,此时应重新进入编辑状态,对源程序进行修改后再重新编译。这是一个重复的过程,直到通过编译为止。编译后得到的二进制代码文件在 DOS 或 Windows 操作系统下其扩展名为“.OBJ”。编译的热键为<Alt+F9>。

(3) 连接。编译后所得到的二进制目标代码还不能被计算机直接执行,因为每个模块往往是单独编译的,必须把经过编译的各个模块的目标代码与系统提供的标准模块(如 C 语言中的标准库函数)连接后才能执行。连接后得到的二进制代码文件在 DOS 或 Windows 操作系统下其扩展名为“.EXE”。连接的热键为 F9。

(4) 执行。经过编辑、编译、连接后所得到的“.EXE”文件可在计算机上执行,并输出结果。执行的热键为<Ctrl+F9>。在 Turbo C/C++ 的集成环境中,C 程序的编译、连接和执行可以合并成一步执行。按<Ctrl+F9>组合键,系统将自动地完成编译、连接和执行操作。

(5) 查看输出结果。在 Turbo C/C++ 系统中,按<Alt+F5>组合键可查看结果。按任意键返回集成环境。在 Turbo C++ 3.0 系统中,程序执行的结果会在“Output(输出)”窗口中直接显示出来。

(6) 退出。按<Alt+X>组合键可退出 Turbo C/C++ 的集成环境,回到操作系统的环境中。退出前建议按 F2 键保存源程序。

1.4 数据结构概述

数据结构是随着计算机应用的日益深入而发展起来的一门计算机学科。计算机所加工处理的对象,例如字符、表格、声音和图像等,是具有一定结构的数据。为了编写一个好的程序,必须分析待处理的数据的特性以及各数据之间的关系。这就是“数据结构”这门学科形成和发展的背景。

数据结构是设计和实现编译程序、操作系统、数据库系统以及其他系统程序和应用程序的重要基础。