



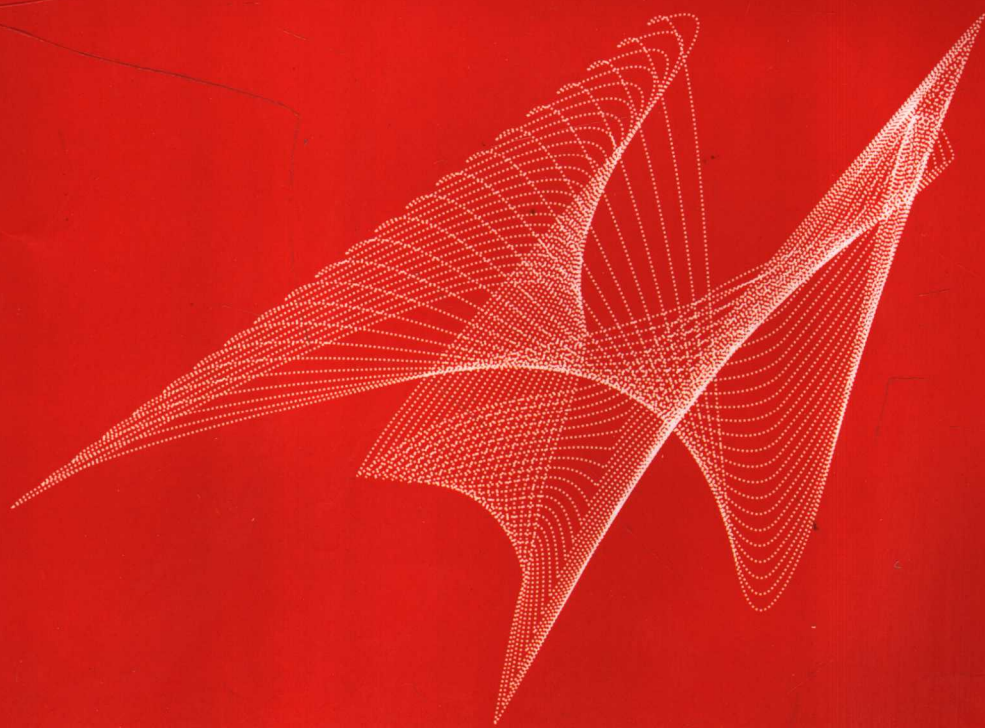
中等职业学校计算机系列规划教材

根据教育部中等职业学校新教学大纲要求编写

C语言程序设计

应用基础教程

张军安 编



西北工业大学出版社

中等职业学校计算机系列规划教材

C 语言程序设计应用基础教程

张军安 编

西北工业大学出版社

【内容提要】本书为中等职业学校计算机系列规划教材。内容包括 C 语言的入门知识、顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、函数、指针、结构体、共用体和文件。书中配有大量生动典型的实例，每章后还附有练习题，这将会使读者在学习和使用 C 语言时更加得心应手，做到学以致用。

本书可作为中等职业学校 C 语言程序设计基础教程课程的教材，同时也可作为工程设计培训班教材及计算机爱好者的自学参考书。

图书在版编目 (CIP) 数据

C 语言程序设计应用基础教程/张军安编. —西安: 西北工业大学出版社, 2006.6
ISBN 7-5612-2076-6

I. C… II. 张… III. C 语言—程序设计—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 030159 号

出版发行: 西北工业大学出版社

通信地址: 西安市友谊西路 127 号 邮编: 710072

电 话: 029-88493844 88491757

网 址: www.nwpup.com

印 刷 者: 陕西天元印务有限公司

开 本: 787 mm × 1 092 mm 1/16

印 张: 15.25

字 数: 409 千字

版 次: 2006 年 6 月第 1 版

2006 年 6 月第 1 次印刷

定 价: 20.00 元

中等职业学校计算机系列规划教材

编审委员会

主 任： 夏清国

副主任： 王 辉 赵建国 孙玉红

李文宏 张社义

委 员： 王小娟 智永军 周苏红 李 林

杨巧云 张 昊 郭礼军 潘小明

李永胜 孟晓伟

序 言

随着国民经济发展速度的加快,操作能力强、能迅速进入职业角色的中等职业学校的学生越来越受到企业的青睐,近年来,还出现了“供不应求”的情况。这就迫切要求中职教材不断更新,始终以就业为导向来培养学生的职业能力。为了配合目前中职教育的现状及中国经济生活的发展状况,我们依据教育部职业教育与成人教育司制定的《中等职业学校计算机及应用专业教学指导方案》以及教育部等六部委最新制定的《中等职业学校计算机应用软件技术专业领域技能型紧缺人才培养方案》,调查和研究了众多中职学校计算机及相关专业的教学计划、课程设置和中职教学的实际需要,并根据中职学生的接受能力和就业要求,聘请了一线的中职骨干教师和技术专家共同组织编写了本套教材。

在本套教材的策划和编写过程中,我们听取了IT专家、中职师生、企事业单位的建议和意见,多次组织了由技术专家及一线中职骨干教师参加的大纲审定会和审稿会,总结和吸取了教师提出的意见,博采众长,使之更加适合当前中职教学的实际需求。

本套规划教材的主要特色如下:

◆ 内容实用、体例新颖

本套教材以市面上最新、最广泛的版本为蓝本,与用人单位紧密结合,在体例上采用理论知识与上机指导二合一的模式,大大节省了读者的人力与财力,达到双赢的目的。

◆ 任务驱动、案例教学

本套教材列举了大量的实例,来提高学生的学习兴趣 and 自主能力,让他们在掌握理论的基础上更多地动手进行具体操作。

◆ 老师好教、学生好学

按照中等职业学校的教育课程模块化和综合化的特点,本套教材对每本书的内容都进行了划分,独立成块。每章都在理论知识之后附有上机指导和本章总结。

◆ 简单实用、职业目的

本套教材语言简洁易懂。在理论知识方面只求“够用为度”,以未来职业方向为基础,更重视培养学生的动手能力,并穿插许多小技巧和小知识,真正让学生放下书本就能上岗。

另外,为了方便教师教学,我们免费提供了电子教学参考资料包。其中包括以下内容:

1. 教材中的程序源代码
2. 教材中涉及的实例制作的各类素材
3. 习题参考答案
4. PowerPoint 多媒体课件

中等职业学校计算机系列规划教材编审委员会

前 言

C 语言是 20 世纪 70 年代由美国贝尔实验室的 D.M.Ritchie 教授在 B 语言的基础上设计的,其广泛涉猎于事务处理、科学计算、工业控制及数据库等领域。C 语言能够得到如此迅猛的发展,不仅因为它兼具了高级语言和汇编语言的优点,既适合系统软件的开发,又适合应用程序的编写。

本书编写特点是将数据库基本原理与数据库管理系统的介绍合二为一,使读者通过本书的学习,既掌握数据库应用系统开发所需要的数据库理论知识,又能通过实际的数据库管理系统进行小型的数据库应用系统开发。

本书是为中等职业学校计算机及应用专业所编写的配套教材,根据教育部职业教育与成人教育指导方案的要求编写,目的是为满足全国中等职业学校计算机及相关专业的实际教学需求,为培养高素质的应用型人才服务。通过本书的学习,使读者能够掌握语言的基本知识和操作技能,并在实际工作中得以广泛的应用。

本书采用“任务驱动、案例教学”的形式编写,且每一章后都附有应用实例,详细介绍了 C 语言的功能与应用,具有较强的实用性和指导性。全书共分为 12 章:

第 1~3 章 介绍了 C 语言的入门知识

第 4~6 章 介绍了 3 种结构化程序设计

第 7 章 介绍了数组

第 8 章 介绍了函数和预处理

第 9 章 介绍了指针、结构体和共用体

第 10 章 介绍了文件

第 11 章 通过一个综合实例介绍了 C 语言在处理实际问题时的方法和技巧

第 12 章 介绍了几个具体上机操作实例

本书可作为中等职业学校计算机应用基础——C 语言课程的教材,同时也可作为工程设计培训班教材及计算机爱好者的自学参考书。

由于编者水平有限,不足之处在所难免。恳请广大读者将本书的使用情况及各种意见和建议及时反馈给我们,以便我们在今后的工作中不断地改进与完善。

编 者

目 录

第 1 章 C 语言概述.....	1
1.1 C 语言发展史.....	1
1.2 C 语言的特点.....	2
1.3 Turbo C 简介.....	3
1.4 C 语言程序.....	5
本章小结.....	7
习题一.....	8
第 2 章 C 语言的基本数据类型.....	9
2.1 C 语言数据类型.....	9
2.2 常量与变量.....	10
2.2.1 常量和符号常量.....	10
2.2.2 变量.....	11
2.3 整型数据.....	12
2.3.1 整型常量.....	12
2.3.2 整型变量及分类.....	13
2.4 实型数据.....	14
2.4.1 实型常量.....	14
2.4.2 实型变量及分类.....	15
2.5 字符型数据.....	16
2.5.1 字符常量.....	16
2.5.2 字符变量及分类.....	17
2.5.3 字符串常量.....	18
2.6 各类数值型数据间的混合运算.....	19
2.7 程序举例.....	21
本章小结.....	22
习题二.....	22

第 3 章 运算符与表达式	24
3.1 运算符与表达式的概述	24
3.1.1 运算符	24
3.1.2 表达式的定义	25
3.1.3 表达式的值	25
3.2 算术运算符及算术表达式	25
3.2.1 算术运算符	25
3.2.2 算术表达式的定义	26
3.2.3 算术运算程序举例	28
3.3 关系运算符及关系表达式	29
3.3.1 关系运算符	29
3.3.2 关系表达式的定义	29
3.3.3 关系运算程序举例	30
3.4 逻辑运算符及逻辑表达式	31
3.4.1 逻辑运算符	31
3.4.2 按位逻辑运算符	32
3.4.3 逻辑表达式的定义	33
3.5 其他运算符的应用	34
3.5.1 赋值运算符与赋值语句	34
3.5.2 条件运算符	36
3.5.3 逗号运算符	38
3.6 运算符的优先级与结合性	39
3.7 程序举例	39
本章小结	41
习题三	41
第 4 章 顺序结构程序设计	43
4.1 基本 C 语句	43
4.2 字符数据的输入输出	44
4.2.1 getchar 函数	45
4.2.2 putchar 函数	46
4.2.3 应用实例	46
4.3 格式输入与输出	48
4.3.1 printf 函数	48
4.3.2 scanf 函数	51
4.3.3 应用实例	53

4.4 程序举例.....	55
本章小结.....	57
习题四.....	57
 第 5 章 选择结构程序设计.....	 59
5.1 概述.....	59
5.2 if 语句.....	60
5.2.1 if 结构.....	60
5.2.2 if...else 结构.....	61
5.2.3 if...else if...else 结构.....	62
5.2.4 应用实例.....	63
5.3 if 语句的嵌套.....	64
5.4 switch 语句.....	66
5.4.1 switch 结构.....	67
5.4.2 应用实例.....	68
5.5 程序举例.....	69
本章小结.....	71
习题五.....	71
 第 6 章 循环结构程序设计.....	 73
6.1 goto 语句.....	73
6.2 while 语句.....	74
6.2.1 do...while 语句.....	74
6.2.2 while 语句.....	76
6.2.3 do...while 与 while 的区别.....	77
6.2.4 应用实例.....	78
6.3 for 语句.....	80
6.3.1 简单 for 语句.....	81
6.3.2 for 语句的嵌套.....	82
6.3.3 应用实例.....	84
6.4 循环控制 2 语句.....	86
6.4.1 break 语句.....	86
6.4.2 continue 语句.....	87
6.4.3 应用实例.....	88
6.5 几种循环结构的区别.....	89
6.6 程序举例.....	91

本章小结.....	93
习题六.....	93

第 7 章 数组.....96

7.1 概述.....	96
7.2 一维数组.....	96
7.2.1 一维数组的定义.....	97
7.2.2 一维数组的引用.....	97
7.2.3 一维数组的初始化.....	98
7.2.4 应用实例.....	99
7.3 二维数组.....	103
7.3.1 二维数组的定义.....	103
7.3.2 二维数组的引用.....	104
7.3.3 二维数组的初始化.....	105
7.3.4 应用实例.....	106
7.4 字符数组.....	106
7.4.1 字符数组的定义.....	106
7.4.2 字符数组的初始化.....	107
7.4.3 字符数组的引用.....	107
7.4.4 字符串和字符串结束标志.....	108
7.4.5 字符串的输入输出.....	109
7.4.6 字符串处理函数.....	111
7.4.7 应用实例.....	116
7.5 程序举例.....	117
本章小结.....	120
习题七.....	120

第 8 章 函数与预处理.....122

8.1 函数.....	122
8.1.1 函数的定义.....	122
8.1.2 函数参数及返回值.....	127
8.1.3 函数调用格式.....	132
8.1.4 变量的作用范围.....	135
8.1.5 函数的作用范围.....	137
8.1.6 多个文件的连接和运行.....	138
8.1.7 应用实例.....	140

8.2 预处理.....	142
8.2.1 宏定义	142
8.2.2 文件包含	144
8.2.3 条件编译	146
8.2.4 应用实例	147
8.3 程序举例.....	148
本章小结.....	149
习题八.....	149
 第 9 章 指针、结构体和共用体.....	151
9.1 指针	151
9.1.1 指针的概念、定义和操作	151
9.1.2 指针变量作为函数参数	153
9.1.3 指针与数组	154
9.1.4 指针的数据类型和指针运算	156
9.1.5 应用实例	157
9.2 结构体.....	159
9.2.1 结构体的概念	159
9.2.2 结构体变量初始化	160
9.2.3 结构体变量成员的访问	161
9.2.4 应用实例	165
9.3 共用体.....	167
9.3.1 共用体的概念	167
9.3.2 共用体变量的应用方式	168
9.3.3 共用体类型数据的特点	169
9.3.4 应用实例	169
9.4 程序举例.....	172
本章小结.....	174
习题九.....	174
 第 10 章 文件.....	176
10.1 概述.....	176
10.2 文件指针.....	177
10.3 文件的打开与关闭.....	177
10.3.1 fopen 函数	177
10.3.2 fclose 函数.....	178
10.3.3 应用实例	178

10.4 文件的读写	179
10.4.1 fputc 函数与 fgetc 函数	179
10.4.2 fputs 函数与 fgets 函数	181
10.4.3 fread 函数与 fwrite 函数	184
10.4.4 fprintf 函数与 fscanf 函数	186
10.4.5 应用实例	189
10.5 文件的定位	191
10.5.1 rewind 函数	191
10.5.2 fseek 函数	191
10.5.3 ftell 函数	192
10.5.4 应用实例	192
10.6 出错检测	195
10.6.1 ferror 函数	195
10.6.2 clearerr 函数	196
10.6.3 应用实例	196
10.7 程序举例	196
本章小结	199
习题十	199
 第 11 章 综合实例	 201
实例 学生信息管理程序	201
 第 12 章 上机指导	 222
12.1 结构程序设计	222
12.2 数组与函数	224
12.3 指针、结构体与文件	226
 附录	 230
附录一 常用字符与 ASCII 码对照表	230
附录二 运算符和结合性	231

第 1 章

C 语言概述

【学习目标】

C 语言是一种国际上广泛流行的计算机语言。它不仅适合系统软件的开发,也适合应用软件的开发。本章将介绍 C 语言的基础知识,通过本章的学习,使读者了解 C 语言的整体情况,并掌握简单的 C 语言上机操作。

【知识要点】

- ◆ C 语言发展史
- ◆ C 语言的特点
- ◆ Turbo C 简介
- ◆ C 语言程序

1.1 C 语言发展史

自计算机诞生以来,人们编写系统软件主要是使用汇编语言,但是由汇编语言编写的程序对计算机硬件的依赖太强,程序可读性和可移植性较差。为了改进汇编语言的这些不足,就需要改用高级语言,而一般的高级语言又不具备汇编语言能够直观地实现对硬件的操控。在这种情况下,就需要一种同时具有高级语言特性和低级语言特性的语言,于是 C 语言就应运而生。C 语言是一种典型的高级语言,不仅具备一般高级语言的共性,同时也兼容了低级语言的某些特点,因而受到广大用户的青睐,成为当今最有发展前途的计算机高级语言之一。

C 语言是 20 世纪 70 年代由美国贝尔实验室的 D.M.Ritchie 教授在 B 语言的基础上设计的,此时 C 语言只在贝尔实验室内部使用。在以后的数年里,C 语言做了多次改进,但它依旧是以描述和实现 UNIX 操作系统作为贝尔实验室内部使用而存在。直到 1975 年,UNIX 第 6 版公布后,C 语言的优势才慢慢被人们普遍注意。接着出现了可移植的 C 语言,这不仅推动了 UNIX 操作系统的广泛应用,而且 C 语言也迅速得到推广。1978 年 UNIX 第 7 版的 C 语言成为 C 语言标准版本的基础。1983 年,美国国家标准协会(ANSI)根据 C 语言的各版本,对 C 语言进行发展和扩充,制定了新标准,称为 ANSI C。ANSI C 比以前的标准 C 有了很大的发展。目前流行的 C 语言编译系统是以 1990 年国际标准化组织 ISO 制定的 ISO C 为标准的。

1.2 C 语言的特点

目前 C 语言广泛涉猎于事务处理、科学计算、工业控制及数据库等领域。C 语言能够得到如此迅猛的发展,不仅因为它兼备了高级语言和汇编语言的优点,既适合系统软件的开发,又适合应用程序的编写,更主要是因为它具备以下优点:

1. 具有结构化的控制语句

在 C 语言中,有 3 种基本结构,即顺序结构、选择结构和循环结构。许多复杂的问题往往可以通过这 3 种结构的交叉使用得以解决。因此,C 语言是理想的结构化语言,符合现代编程风格的要求。

2. 高效的可移植性

C 语言程序便于移植,基本上不做修改就能用于各种型号的计算机和各种操作系统。

3. 语言简洁、明了

C 语言简洁、紧凑、使用方便、灵活。其中有 32 个关键字如表 1.1 所示;9 个控制语句如表 1.2 所示。程序书写格式自由,丰富的库函数为复杂问题的解决提供了保障。

表 1.1 C 语言中的关键字

关键字					
auto	break	case	char	const	continue
default	do	double	else	enum	extern
float	for	goto	if	int	long
register	return	short	signed	sizeof	static
struct	switch	typedef	union	unsigned	void
volatile	while				

表 1.2 C 语言中的控制语句

控制语句	控制语句
if(表达式)语句或 if(表达式)语句 1 else 语句 2	while(表达式)语句
do 语句 while(表达式)	for(表达式 1;表达式 2;表达式 3)语句
switch(表达式){case 常量表达式 1:语句 1; case 常量表达式 2:语句 2;...default:语句 n;}	break
continue	return
goto	

4. 丰富的运算符

在 C 语言中共有 34 个运算符,具体将在后面章节中逐一介绍。各种运算符的交叉使用可以实现许多高级功能。

5. 丰富的数据结构

在 C 语言中包含基本数据类型如实型、整型及字符型等,而且在此基础上设计者还可以创建很多复杂的数据结构如链表、树、堆、栈等。这些丰富的数据类型极大地增强了 C 语言的功能。

6. 自由化的程序设计空间

一般高级语言语法检查比较严,能检查出几乎所有的语法错误,而 C 语言语法限制不太严格,程序设计自由空间较大,对变量的类型使用比较灵活。

7. 与汇编语言的直接接触

在 C 语言中允许设计者直接访问物理地址,并能直接对硬件进行各种操作,这样就能实现汇编语言的大部分功能。

C 语言既是成功的系统描述语言,又是程序设计语言,它的这种双重性越来越多地受到设计者的青睐。目前国内外研究和使用的 C 语言的人日益增多,不仅计算机专业工作者对其钟爱有加,而且广大的计算机爱好者对其也刮目相看,同时优秀的 C 语言版本及配套的工具软件不断出现,更为 C 语言的学习提供了广阔的平台。

1.3 Turbo C 简介

众所周知计算机所能识别和执行的只是由 0 和 1 组成的二进制代码,而源程序大多是程序员使用高级语言编写的高级语言指令,为了使计算机能够识别并执行这些高级语言指令,必须先把这些源代码翻译成二进制形式的目标程序,然后将这些目标程序与系统的函数库或其他目标程序连接起来,形成可执行的目标程序。一个 C 语言程序从编写成功到实现既定功能,需要经历以下几个基本环节。

1. 编辑

为实现某个既定功能,使用 C 语言编写程序代码,形成源文件,一般以 .c 为后缀。

2. 编译

在 C 程序的编译过程中,可以查出程序中的语法错误,并给出响应提示,编译器将源程序转化成机器二进制代码后,就生成了目标程序,一般以 .obj 为后缀。

3. 连接

C 语言是一种模块化设计程序,一个大型的 C 程序不可能只有一个设计者独立完成,在实现最终功能前需要将各个模块连接成一个整体,生成可执行文件,一般以 .exe 为后缀。

4. 运行

运行生成的可执行文件即可获得既定目标。

在 C 语言程序的整个生存周期中,最重要的编译环节是在 Turbo C 下实现的。Turbo C 是 Borland 公司开发的用于 PC 机的一种 C 语言程序开发软件。它具有良好的用户界面和丰富的库函数,具有运行速度快、效率高和功能强等优点,使用非常方便。下面将介绍使用 Turbo C 开发 C 程序的一般方法和步骤。

(1) 启动 Turbo C 后,打开 Turbo C 集成开发环境,如图 1.3.1 所示。其中, Turbo C 集成开发环境包括以下 4 个部分。

- 1) 主菜单:主菜单有 File, Edit, Run, Compile, Project, Options, Debug 和 Break/Watch。
- 2) 编辑窗:在编辑状态下可以根据需要输入或修改源程序。
- 3) 信息窗口:程序运行结果的提示。
- 4) 功能键提示:常用功能快捷键。

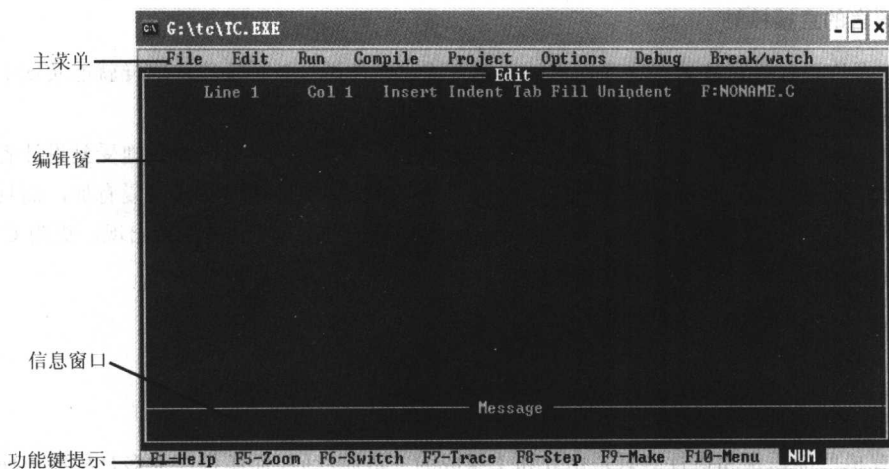


图 1.3.1 Turbo C 集成开发环境

(2) 调用 Turbo C 程序。按“F10”键，激活主菜单，然后按“F”键，选择“File”→“Load”命令，并按回车键，表示调用一个已经存在的源文件，如图 1.3.2 所示；选择“New”命令，并按回车键，表示要建立一个新的 C 程序源。

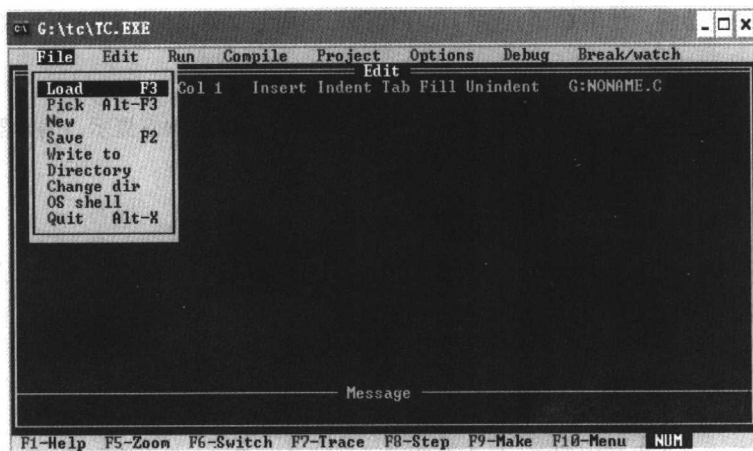


图 1.3.2 调用 Turbo C 源文件

(3) 编译源程序。按“F10”键，激活主菜单，然后按“C”键，选择“Compile”→“Compile to OBJ”命令，则进行编译，并生成目标文件，然后再选择“Link EXE file”命令，进行连接操作，即可得到可执行文件。一般情况下，将编译与连接合并成一步进行，可以通过选择“Make EXE file”命令或直接按“F9”快捷键来实现。

(4) 运行程序。按“F10”键，激活主菜单，然后按“R”键，选择“Run”→“Run”命令，或按“Ctrl+F9”组合键来实现运行操作；然后选择“User screen”命令，或按“Alt+F5”组合键显示运行结果，此时 Turbo C 集成环境消失，弹出程序运行时的输出结果。

(5) 退出 Turbo C。按“F10”键，激活主菜单，然后按“F”键，选择“File”→“Exit”命令，或按“Alt+X”组合键来实现退出操作。



1.4 C 语言程序

程序是在特定的语言环境下为解决某一具体问题而编写的一组计算机能识别和执行的指令集合。下面介绍几个简单的C程序，从中分析C程序的组成及其特性。

例 1.1 在屏幕上显示 I like C Program!

程序

```
#include<stdio.h>
main()
{
    printf("I like C Program!\n");
}
```

输出

I like C Program!

分析

main()表示主函数，大括号“{}”中的内容为函数体，这是每个C程序中必须具备的；本程序的函数体中仅包括一个输出语句printf("I like C Program!\n");，“\n”表示换行；值得注意的是每条C语句结束后都要加一个分号“;”。

例 1.2 比较两个数的大小，并输出较大的数。

程序

```
#include<stdio.h>
main()
{
    int num1,num2,                /*定义变量 num1, num2*/
    num1=12;
    num2=24;                      /*为变量 num1 和 num2 赋初值*/
    if (num1>=num2)               /*条件判断 num1 和 num2 的大小*/
        printf("%d",num1);       /*如果 num1 较大则输出*/
    else
        printf("%d",num2);       /*反之则输出 num2*/
}
```

输出

24

分析

在C语言中使用任何变量前都必须先定义，int是整型变量；在程序中“/*.....*/”包含的部分为注释，其目的是便于阅读程序，不影响整个程序的编译运行；关键字“if”是条件判断语句，其作用