

高等学校教材

C 语言程序设计

蒋清明 主编

黄晓宇 副主编

向德生 何宏 编著

人民邮电出版社

图书在版编目 (CIP) 数据

C 语言程序设计 / 蒋清明主编; 向德生等编著. —北京: 人民邮电出版社, 2005.2
高等学校教材

ISBN 7-115-13092-2

I. C... II. ①蒋...②...向 III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2005) 第 009716 号

内容提要

本书结构严谨, 循序渐进地讲述 C 语言程序设计的基本知识、基本技能和编程方法; 举例经典, 许多例题习题都是工程应用或等级考试中的基本算法的应用。

全书共分 9 章, 主要内容包括基本数据类型和运算符、控制结构、函数、数组、指针、结构与共用、文件以及文本与图形处理。每一章都附有精选的、多种类型的练习题, 有助于读者复习、巩固所学知识, 培养读者的实际编程能力。

本书配有《C 语言程序设计上机与习题选解》。该书使学生明确实验目的、实验任务和实验方法, 可大大改善实验效果, 减轻教师指导学生实验的工作量; 同时, 书中给出的题解有助于学生开阔视野, 明朗思路, 规范书写, 方便调试。

本书可以作为各类高等院校、各类高职院校计算机专业及理工科非计算机专业学生学习“计算机程序设计”课程的教材, 也可作为广大计算机爱好者学习 C 程序设计语言的参考书。

高等学校教材

C 语言程序设计

主 编 蒋清明

副 主 编 黄晓宇

编 著 向德生 向宏

责任编辑 邹文波

人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

读者热线: 010-67129259

内蒙古邮电印刷厂印刷

新华书店总店北京发行所经销

开本: 787×1092 1/16

印张: 17.25 2005 年 2 月第 1 版

字数: 413 千字 2005 年 2 月内蒙古第 1 次印刷

ISBN 7-115-13092-2/TP · 4422

定价: 29.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

随着经济全球化、社会信息化时代的到来，当代大学生不但要会使用计算机获取专业领域知识，还要会使用计算机进行编程，解决专业领域中的具体问题。C 语言是当前流行操作系统 Windows、Linux、UNIX 上的一门系统开发语言，同时又是进行各专业问题计算的有效语言，因此，C 语言已成为各高校计算机专业和非计算机专业必学的一门语言。在非计算机专业等级考试中，C 语言已替代了 Pascal 和 Fortran 语言，因此，学好 C 语言的重要性已不言而喻。

然而，在教学过程中，预期教学目标与最终效果有着明显的差距，教师感觉难教，学生感觉难学、难理解，学会了也不会编程。针对这种情况，我们在编写本书的过程中，主要采取如下措施，以求收到更好的效果。

措施 1：体系合理。本书首先讲述 C 语言的输入/输出函数、运算符和数据类型，让学生尽快入门，学会简单的编程；然后讲述结构化、模块化编程；最后讲述数组、指针、结构、共用、文件等构造类型。

措施 2：重点突出。实现平滑过渡和各个击破。本书在编排体系上做到由入门到结构化、模块化编程，最后讲述 C 语言的难点，即指针、结构、共用、文件等构造类型。

措施 3：举例经典。为了配合非计算机专业的等级考试和提高计算机专业编程能力，本书的例题基本上采用经典算法讲解，在编写教材前，我们已将等级考试一些常考的算法进行分类，然后分解到各章之中。

措施 4：问题突破。为了帮助大家提高解决问题的能力，我们还编写了一本《C 语言程序设计上机指导与题解》，该书分为实验部分、问题解答和等级考试模拟试题三个方面。在学习过程中参考该书，有助于提高解决问题的能力。

我们在内容体系上做了精心的考虑，希望这些措施在教学过程中得到体现与落实，我们更希望学生在学习语言的过程中做到：

1. 学好语法知识，任何一门计算语言都有其相应的语法知识，它们是编写计算机程序的基础；
2. 多上机，要多加加强实践锻炼，多思考，多模仿，只有这样，才能提高自己的编程能力，为将来的学习和工作打下坚实的基础。

参加本书编写工作的有蒋清明、黄晓宇、向德生、何宏等，其中蒋清明编写第 1 章、第 2 章，黄晓宇编写了第 7 章、第 8 章，向德生编写了第 5 章、第 6 章和第 9 章，何宏编写了第 3 章、第 4 章，全书由蒋清明、黄晓宇统稿。徐建波教授、肖建华教授在百忙中抽出时间对本书进行了审核，在此一并表示感谢。

由于作者水平有限，加上时间仓促，错误之处在所难免，希望广大读者批评指正。

编 者
2005 年 1 月

目 录

第 1 章 绪论	1
1.1 C 语言的发展过程	1
1.1.1 计算机语言的发展过程	1
1.1.2 C 语言的发展过程	2
1.2 C 语言的特点	2
1.3 C 程序的结构和书写格式	3
1.3.1 C 程序的结构	3
1.3.2 C 程序的书写格式	4
1.4 Turbo C 上机操作	4
1.4.1 C 语言程序的生成过程	4
1.4.2 Turbo C 2.0 上机操作过程	4
习题	5
第 2 章 基本数据类型与运算符	7
2.1 数据类型和取值范围	7
2.1.1 数据类型和取值范围	7
2.1.2 整型常量	8
2.1.3 实型常量	9
2.1.4 字符常量	9
2.1.5 字符串类型	10
2.2 运算符	11
2.2.1 优先级与结合规则	11
2.2.2 赋值运算与连续赋值	12
2.2.3 算术运算	12
2.2.4 关系运算	14
2.2.5 逻辑运算、连续比较和逻辑优化	14
2.2.6 位运算	15
2.2.7 条件运算	16
2.2.8 复合赋值运算	17
2.2.9 逗号运算	17
2.2.10 其他运算 sizeof	18
2.2.11 类型转换与类型转换规则	18
2.3 输入/输出函数	21
2.3.1 格式化输出函数 printf	21
2.3.2 格式化输入函数 scanf	23

2.3.3 字符输入/输出函数	25
习题	26
第3章 控制结构	29
3.1 程序结构框图	29
3.1.1 自然语言描述	29
3.1.2 流程图	30
3.1.3 N-S图	31
3.1.4 结构化程序设计	31
3.1.5 复合语句	32
3.2 二分支结构	32
3.2.1 二分支结构选择语句	33
3.2.2 不平衡 if 结构	35
3.2.3 if 语句的嵌套	36
3.3 多分支结构	40
3.4 循环结构	44
3.4.1 for 循环	44
3.4.2 while 语句	47
3.4.3 do~while 语句	48
3.4.4 循环嵌套	50
3.5 break 与 continue 语句	52
3.5.1 break 语句	52
3.5.2 continue 语句	53
3.5.3 goto 语句	54
习题	55
第4章 函数	60
4.1 函数调用过程	60
4.2 函数的定义	61
4.2.1 函数定义的一般形式	61
4.2.2 函数定义中的要点说明	61
4.2.3 函数的声明	65
4.3 递归函数	66
4.3.1 递归概念	66
4.3.2 递归举例	67
4.4 存储类型、生存期和作用域	70
4.4.1 存储类型	70
4.4.2 生存期和作用域	71
4.5 编译预处理	77
4.5.1 文件包含	78
4.5.2 宏定义	79

4.5.3 条件编译	81
习题	83
第 5 章 数组	88
5.1 一维数组	88
5.1.1 一维数组的定义与初始化	88
5.1.2 一维数组的引用	90
5.1.3 字符型数组与字符串	91
5.1.4 字符串操作	93
5.2 二维数组	96
5.2.1 二维数组的定义	96
5.2.2 二维数组的引用	97
5.3 多维数组	99
5.4 函数与数组	101
5.4.1 函数与一维数组	101
5.4.2 函数与二维数组	103
5.5 问题与解答	106
5.5.1 “气泡法”排序问题	106
5.5.2 二分法查找问题	108
5.5.3 Josephus (约瑟夫) 问题	109
习题	110
第 6 章 指针	117
6.1 指针与变量	117
6.1.1 指针的基本概念	117
6.1.2 指针变量的定义与引用	118
6.1.3 指针的运算	120
6.1.4 指向指针的指针	121
6.2 指针与数组	122
6.2.1 指向数组元素的指针	122
6.2.2 指向数组的指针	130
6.2.3 指针数组	136
6.2.4 指针与字符串	140
6.3 指针与函数	143
6.3.1 指针作函数参数	143
6.3.2 返回指针值的函数	147
6.3.3 函数指针	149
6.4 问题与解答	152
6.4.1 本章重点概念的复习	152
6.4.2 “选择法”排序问题	152
6.4.3 子串定位问题	153

习题	154
第 7 章 结构与共用	161
7.1 结构类型	161
7.1.1 结构类型的定义、初始化与使用	161
7.1.2 结构类型数组	166
7.1.3 结构类型数据的指针	167
7.1.4 嵌套结构	171
7.1.5 用指针处理链表	172
7.2 共用类型	179
7.2.1 共用类型的定义	179
7.2.2 共用类型变量的引用	181
7.3 枚举类型	183
7.4 位域	185
7.4.1 位运算符与位运算	185
7.4.2 位域	186
7.5 自定义类型	188
习题	189
第 8 章 文件	199
8.1 文件概述与文件类型指针	199
8.1.1 文件概述	199
8.1.2 文件类型指针	200
8.2 文件的打开与关闭	201
8.2.1 文件的打开	201
8.2.2 文件的关闭	202
8.3 文件的读写	203
8.3.1 字符读写函数 fgetc()和 fputc()	203
8.3.2 字符串读写函数 fgets()和 fputs()	205
8.3.3 格式化读写函数 fscanf()和 fprintf()	207
8.3.4 数据块读写函数 fread()和 fwrite()	209
8.4 文件的定位	211
8.5 文件检测	213
8.6 文件的低层操作	214
习题	217
*第 9 章 文本与图形处理	224
9.1 文本的屏幕输出与键盘操作	224
9.1.1 文本的屏幕输出	224
9.1.2 键盘操作	230
9.1.3 综合实例程序	232
9.2 图形编程	235

9.2.1 图形模式初始化	235
9.2.2 独立图形运行程序的建立	239
9.2.3 屏幕颜色的设置和清屏函数	241
9.2.4 基本图形处理函数	243
9.2.5 图形模式下的文本输出	250
9.2.6 综合图形实例程序	253
附录 1 常用字符与 ASCII 值对照表	256
附录 2 C 语言保留字一览表	258
附录 3 运算符的优先级及其结合性	259
附录 4 常用 C 库函数	260
参考文献	265

第 1 章

绪 论

1.1 C 语言的发展过程

1.1.1 计算机语言的发展过程

计算机语言是人与计算机进行交互的工具，是用户进行计算机软件开发、编写计算机程序的工具。计算机语言的发展过程大致可以分为如下 3 个阶段。

1. 机器语言

计算机指令采用二进制（0、1）表示，也就是说，计算机能识别的指令代码只能是二进制形式。采用这种二进制形式表示的语言称为机器语言，或称为低级语言。如 PC 机中两个数进行加法的指令为：0000010 11111001。由于机器语言采用的是二进制序列表示指令，十分难记，这是其特点之一。另外，采用机器语言编写的计算机程序具有不可移植性，即对某一种体系结构的计算机编写的计算机程序，在另一种体系结构的计算机上不能运行。

2. 汇编语言

由于机器语言难学，难记，计算机科学家将机器语言的每一条指令采用助记符表示，即机器语言的符号法，称为汇编语言。如上面 PC 机的加法指令用符号表示为：ADD AH, BL。采用汇编语言编写的计算机程序必须经过翻译过程，变为机器语言后，计算机才能识别运行，这种翻译程序称为汇编程序，对应的过程称为汇编过程。用汇编语言编写的计算机程序仍与体系结构有关，具有不可移植性。但采用机器语言和汇编语言编写的计算机程序具有运算效率高的特点。

3. 高级语言

高级语言是一种更接近于自然的数学形式语言，如两个数的加法可写为 $z=x+y$ 。采用高级语言编写的计算机程序与机器类型无关，具有可移植性、易学易记等特点，采用高级语言编写的程序称为源程序。但高级语言编写的计算机程序，计算机不能直接识别，必须经过翻译过程将其译为机器语言后，计算机才能识别运算。其翻译过程分为两种。一种是边翻译，边运行，翻译一句，执行一句，这种过程称为解释过程，对应的语言称为解释语言。每次执行程序时，都必须经过相同的翻译过程，如早期的 BASIC 语言和 FoxBase 等。采用解释语言编写的计算机程序不能离开其解释环境。另一种是编译语言，它是将整个源程序全部翻译成机器语言指令后，计算机才能运行，这样的翻译过程称为编译过程，对应的翻译程序称为编译程序。源程序经编译后生成的机器语言程序称为目标程序，计算机不能直接运行目标程序，还必须经过连接过程，才能变为可执行文件，对应连接过程的程序称为连接程序，这样生成

的可执行文件具有永逸性，即经过一次编译、连接后，生成的可执行文件以后不需要再进行编译连接过程，可以脱离语言环境，在同类型的计算机上仍可运行，如 ForTran 语言、Pascal 语言、Lisp 语言、Ada 语言和 C 语言等。

1.1.2 C 语言的发展过程

20 世纪 70 年代初，编写计算机系统软件时使用了一种符号法的自展组合语言 BCPL，BCPL 进一步发展为一种系统软件描述语言 B 语言。20 世纪 80 年代初，美国贝尔实验室软件开发人员 Dennis M. Richard 将 B 语言发展成为 C 语言。C 语言继承了 B 语言的特点，成为编写系统软件的重要工具语言。最初 C 语言有各种不同的标准，1983 年美国标准协会制定了 C 语言标准草案，称为 83ANSI C，1989 年正式修订后成为大家公认的标准，称为 89ANSI C。该标准中规定了 C 语言的关键字为 28 个，1999 年在原 89ANSI C 基础上增加了新的面向对象特性，并增加了 4 个关键字，该标准即为现在的 99ANSI C。

不同的编译器开发商在遵照 C 语言标准的基础上，对标准 C 新增了一些特性，如增加了图形图像处理能力，或在标准 C 的基础上，增加了特定的库函数，编译器的实现方式不同，这样市面上出现了 Borland 公司的 Turbo C，Microsoft 公司的 Microsoft C 等不同的编译器，都可实现对 C 语言程序的编辑、编译、连接和运行。Microsoft C 增加面向对象特性后，发展为 Microsoft C++ 和可视化编程的 Microsoft Visual C++。

1.2 C 语言的特点

C 语言作为一种系统开发语言，与其他高级语言或中级语言相比，具有如下特点。

(1) C 语言有丰富的运算符。C 语言除提供了其他高级语言提供的算术运算、关系运算、逻辑运算、下标运算和赋值运算等运算符外，还提供了位运算、地址运算、成员运算等运算符，这些运算符有助于程序员编写出高效的系统软件。

(2) C 语言有丰富的数据类型。C 语言包括整数型、字符型、实数型、空类型等基本数据类型和数组、指针、结构、共用、枚举、位结构等构造数据类型，还允许用户自定义新的数据类型。

(3) C 语言是结构化程序设计语言。C 语言提供了结构化程序设计的 3 种基本结构，即顺序结构、选择结构和循环结构。采用 3 种基本结构反复嵌套可实现任何复杂的运算。

(4) C 语言是模块化语言。C 程序由函数组成，这些函数可以是系统提供的库函数，也可以是用户自定义的函数，程序员可以利用函数构造计算机程序。

(5) 任何一个 C 程序有且仅有一个称之为 `main` 函数的 `main` 函数。程序执行从 `main` 函数开始，其他函数通过 `main` 函数直接或间接调用才能执行，`main` 函数执行结束时，标志程序执行结束。

(6) C 语言有丰富的预处理功能。预处理有利于提高程序的可读性、可移植性、正确性和书写程序的高效性。

(7) C 语言是面向过程的语言，其函数采用面向过程的思想进行设计。

(8) C 程序具有可移植性。不同的程序员可以在不同的平台上设计实现某一大型软件中的子功能，然后在另一平台上进行组装，构成大型软件。

1.3 C 程序的结构和书写格式

1.3.1 C 程序的结构

在讲述 C 程序的基本结构与特征前，我们先看如下两个 C 程序的例子。

例 1.1 向控制台输出信息"Hello,World."。

```
#include <stdio.h> /*预处理命令：包含有标准输入输出库函数的头文件 stdio.h*/
void main() /*主函数*/
{
    printf("Hello,World.\n");
}
```

例 1.2 输入两个数 a、b，并输出其最大值。

解题思路 输入两个数 a、b，调用自定义函数 max(a,b)，求出其最大值并赋给 c，然后输出最大值 c。

```
#include <stdio.h>
int max(int a,int b)
{
    return a>b?a:b;
}
void main()
{
    int a,b,c;
    scanf("%d%d",&a,&b);
    c=max(a,b);
    printf("max=%d\n",c);
}
```

由以上两例我们可以看出，C 函数的基本结构为：

```
[返回值类型] 函数名([形参说明表])
{
    变量定义部分;
    语句执行部分;
}
```

其中 C 函数中要用到的变量必须先定义，然后才能使用，因此变量定义在执行语句前。

而 C 程序结构如下：

```
[预处理语句]
[外部变量定义]
[用户自定义函数]
主函数定义
```

其中 [] 中的内容为可省略部分。

1.3.2 C 程序的书写格式

在编辑 C 语言源程序时，我们应注意如下几点。

(1) C 程序采用块注释方法，块注释书写方法为：

```
/* 注释部分 */
```

注释部分只用于提高程序的可读性，不影响程序的编译和运行。但在书写格式上要注意：“/”与“*”之间或“*”与“/”之间不能有空格。

(2) C 语言一般采用小写字母作为标识符。而 BASIC 语言中，一般采用大写字母作为标识符。

(3) C 语言是区分大小的。如：MAX、max 和 Max 表示的是 3 个不同的标识符。

(4) C 程序书写格式灵活，一个语句可连续写在多行上，一行也可以写多个语句。如例 1.2 中的 max 函数可以写成如下形式：

```
int max(int a,int b){ return a>b?a:b;}
```

(5) 为了使书写的程序结构清晰、层次分明，建议采用“缩进对齐”的格式编辑 C 语言源程序，即同一结构层次的语句应左对齐，而结构下的语句相对于结构本身而言向右缩进。

C 程序书写格式灵活，这对程序员书写程序没有什么约束，如标识符可以采用小写字母，也可以采用大写字母表示，程序可以采用缩进对齐的格式书写，也可以不采用缩进对齐的格式书写，但我们建议初学者养成良好的程序书写规范，以便于交流和调试。

1.4 Turbo C 上机操作

1.4.1 C 语言程序的生成过程

C 语言是一种编译语言，采用 C 语言生成程序的过程如下。

(1) 利用编辑器生成文本文件，该文本文件又称为 C 源程序，其扩展名为.c。编辑器可以是 C 系统提供的，也可以是其他文本编辑器，如 DOS 下的 Edit、Edlin 或 Windows 下的记事本或写字板等。

(2) 采用 C 编译器将源程序编译为二进制的机器目标文件，生成的目标文件扩展名为.obj。

(3) 采用 C 连接程序将目标文件与库文件连接，生成可执行文件，可执行文件扩展名为.exe。上述三步过程可用图 1.1 来描述。

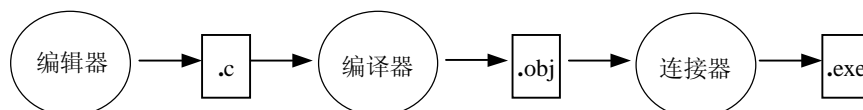


图 1.1 C 语言程序的生成过程

1.4.2 Turbo C 2.0 上机操作过程

我们以 Turbo C 2.0 为例，简单说明 C 程序上机操作过程，Turbo C 2.0 是一个集成开发

环境 (Integrated Development Environment, IDE), 它集 C 编辑器、C 编译器、C 连接器和 C 调试器于一体, 可方便 C 程序员编辑、编译、连接和调试程序。其主要操作步骤如下。

(1) 运行 tc.exe 文件。可在 DOS 命令方式下键入 tc 命令, 或在 Windows 窗口方式下激活 tc, 出现如图 1.2 所示的集成开发环境。该集成开发环境划分为 4 个部分: 菜单系统、编辑区、信息区/观察窗口和热键区。

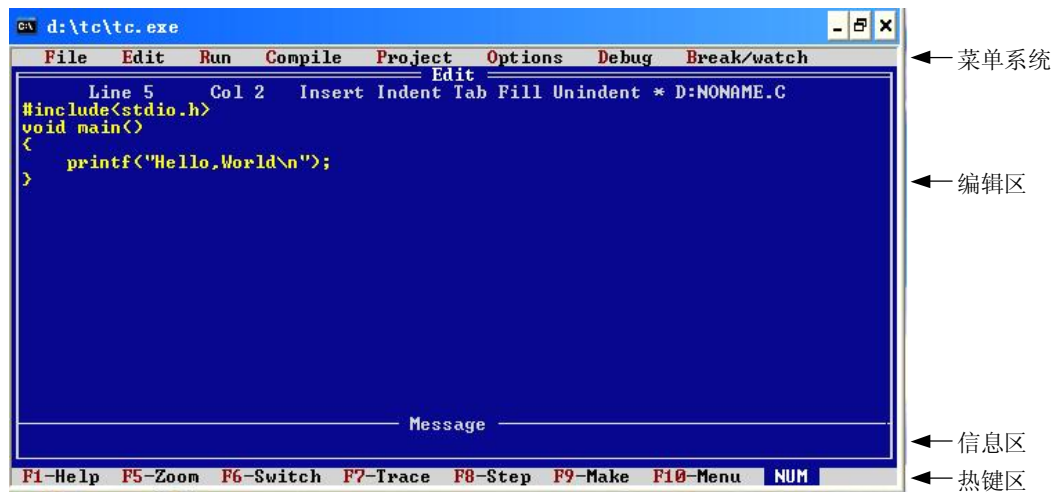


图 1.2 Turbo C2.0 的集成开发环境

其中, 菜单系统给出了 Turbo C 2.0 的主要功能, 如文件、编辑、编译、运行和调试等功能。编辑区用于输入编辑 C 源程序, 信息区/观察窗口显示编译连接信息或调试程序时变量运行值, 热键区显示菜单系统中常用命令键对应的快速操作按钮。

(2) 编辑新的源程序或打开已有的源程序。按 File/New、Alt+F/Load 或 F3 键, 弹出输入文件名对话框, 在输入文件名对话框中输入源程序名后回车; 然后在编辑区中键入源程序代码; 源程序键入结束后必须按 Alt+F/Save 或 F2 键存盘。

(3) 编译过程。按 Alt+C/Compile to OBJ, 将生成二进制的.OBJ 目标文件。

(4) 连接过程。按 Alt+C/Link EXE file 或 F9 键, 将生成可执行的.EXE 文件。

(5) 运行过程。按 Alt+R/Run 或 Ctrl+F9 键, 将运行程序。

(6) 查看运行结果。运行程序后按 Alt+F5 或 Alt+R/User screen, 将看到程序运行结果。

上面的编译、连接和运行过程可按 Ctrl+F9 组合键, 将顺次执行编译、连接和运行 3 个过程。

习 题

一、选择题

1. 一个完整的可运行的 C 源程序中 ()。

- | | |
|----------------|----------------|
| A) 可以有一个或多个主函数 | B) 必须有且仅有一个主函数 |
| C) 可以没有主函数 | D) 必须有主函数和其他函数 |

2. 构成 C 语言源程序的基本单位是 ()。
A) 子程序 B) 过程 C) 文本 D) 函数
3. 某 C 程序由一个主函数 main() 和一个自定义函数 max() 组成, 则该程序 ()。
A) 总是从 max() 函数开始执行 B) 写在前面的函数先开始执行
C) 写在后面的函数先开始执行 D) 总是从 main() 函数开始执行
4. C 语言规定, 一个 C 源程序的主函数名必须为 ()。
A) program B) include C) main D) function
5. 下列说法正确的是 ()。
A) 在书写 C 语言源程序时, 每个语句以逗号结束
B) 注释时, “/” 和 “*” 号间可以有空格
C) 无论注释内容的多少, 在对程序编译时都被忽略
D) C 程序每行只能写一个语句
6. C 语言源程序文件的后缀是 (), 经过编译后, 生成文件的后缀是 (), 经过连接后, 生成文件的后缀是 ()。
A) .obj B) .exe C) .c D) .doc
7. 编辑程序的功能是 ()。
A) 建立并修改程序 B) 将 C 源程序编译成目标程序
C) 调试程序 D) 命令计算机执行指定的操作
8. C 编译程序是 ()。
A) C 程序的机器语言版本 B) 一组机器语言指令
C) 将 C 源程序编译成目标程序的程序 D) 建立并修改程序

二、简答题

1. 如何使用注释语句? 使用注释有何好处?
2. C 程序对书写格式有何要求? 规定书写格式有何好处?
3. 简述 C 程序上机调试的一般步骤?
4. 简述 C 程序从.c 源文件到.exe 可执行文件的生成过程?

第 2 章

基本数据类型与运算符

89 ANSI C 定义了 5 种基本数据类型: 字符型 (char)、整型 (int)、浮点型 (float)、双精度型 (double) 和无值型 (void), 它们是构造其他数据类型的基础。本章主要研究 C 语言基本数据类型和实现在其上的操作, 这些操作包括运算符操作和对基本类型数据的输入/输出操作。

2.1 数据类型和取值范围

2.1.1 数据类型和取值范围

1. 基本数据类型

在计算机中, 所有的数据都是采用二进制形式表示的, 5 种基本数据类型规定了数据在内存中占用的二进制位数/字节数, 从而也规定了数据的取值范围。与数据类型相关的是类型修饰符, 对于整数类型有两类修饰符, 一类是符号修饰, 另一类是长度修饰。其中符号修饰符有带符号 signed 和不带符号 unsigned 之分, 默认为带符号 signed 修饰; 长度修饰有短型 short 和长型 long 之分, 这些数据长度与具体机器编译环境有关。本书以 Turbo C 2.0 作为开发环境, int 类型与 short 类型都是占 2 字节 (16 位); 而在 32 位编程环境下, int 类型与 long 类型都是占 4 字节 (32 位)。

关于数据取值范围, 我们以 16 位整型数为例进行说明, 其他类型数据取值范围可依此类推。不带符号 unsigned int、unsigned 类型最大值为 1111111111111111, 即 $65535 (2^{16}-1)$, 最小值为 0000000000000000, 即 0; 带符号 int、signed int、signed 数采用二进制补码形式表示, 其最大值为 0111111111111111, 即 $32767 (2^{15}-1)$, 其最小值为 1000000000000000, 即 $-32768 (-2^{15})$ 。数据类型和取值范围如表 2.1 所示。

表 2.1 数据类型和取值范围

类 型	别 名	长度 (字节)	范 围	精 度
char	signed char	1	$-2^7 \sim 2^7 - 1$	
unsigned char			$0 \sim 2^8 - 1$	
int	short int	2	$-2^{15} \sim 2^{15} - 1$	
	signed int			
	signed short int			
unsigned int	unsigned	2	$0 \sim 2^{16} - 1$	
	unsigned short int			

续表

类 型	别 名	长度 (字节)	范 围	精 度
long	long int	4	$-2^{31} \sim 2^{31} - 1$	
	signed long			
	signed long int			
unsigned long	unsigned long int		$0 \sim 2^{32} - 1$	
float		4	$10^{-37} \sim 10^{38}$	6
double		8	$10^{-307} \sim 10^{308}$	15

C 语言数据分为两类，一类为常量，另一类称为变量。常量是指在程序运行过程中数值不发生变化的量，如 5, 'a', "Hello, world"; 变量是指程序运行过程中，可以发生变化的量，如 a, Hello 等。

2. 变量的定义

变量的定义格式为：

[存储类型] 数据类型 变量名；

例如：

```
int a, a5, _a;
float _a5, A5;
```

其中变量类型可以是表 2.1 中的任何数据类型，变量命名必须遵守 C 标识符命名规则：

- (1) 第 1 个字符必须是字母或下划线；
- (2) 其余字符可以是字母、下划线和数字；
- (3) 标识符有长度限制，如 Turbo C 规定标识符长度不能超过 32 个字符；
- (4) 字母区分大小写。

(5) 用户自定义标识符不能与 C 语言的保留字或预定义标识符同名，并应尽量做到“见名知意”，以增加程序的可读性。

变量没有赋初值时，变量中存放的是一随机值。变量定义时可同时赋初值，称之为变量的初始化，如：

```
int a=1, a5=10;
```

定义变量包括两个方面的含义：一是给变量分配了存储空间和规定了变量的取值范围，从而可以对变量进行存储操作，如上述举例中，为变量 a、a5、_a 各分配了 2 个字节空间，为变量 _a5、A5 各分配了 4 个字节，变量有了存储空间，也就有变量地址，如 &a、&a5、&_a 分别表示变量 a、a5、_a 的首地址；二是规定了其允许的操作，如实数可进行加、减、乘、除运算，但不能进行求余运算。

2.1.2 整型常量

C 语言中整型常量按进制划分有十进制、八进制（前缀为数字 0）、十六进制（前缀为数字 0X 或 0x）3 种。数据 377 按这 3 种进制的格式可分别表示为：377、0571、0x171（或 0X171）。

常量按长度划分为两种：短整型和长整型（后缀为小写字母 l 或大写字母 L），其中默认为短整型，如 -377 是一个短整型数，而 -377l 或 -377L 表示 -377 是长整型数。

