

21世纪大学计算机基础规划教材

C语言程序设计

恰汗·合孜尔 单洪森 编著

- 主要讲授C语言程序设计的基本思想、方法和解决实际问题的技巧。
- 例题丰富，以实例讲解为主线，对C语言编程作了深入和具体的探讨。
- 可作为高等学校各专业C语言程序设计课程的正式教材，也可以作为计算机等级考试教材，还是计算机爱好者自学用书及各类工程技术人员的参考资料。



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

21 世纪大学计算机基础规划教材

目录

C 语言程序设计

恰汗·合孜尔 单洪森 编著

江苏工业学院图书馆
藏书章

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书讲授 C 语言程序设计的基本思想、方法和解决实际问题的技巧。例题丰富, 各章以实例讲解为主线, 对 C 语言编程做了深入和具体的探讨, 以培养学生编写程序的能力。

全书共 11 章, 分别介绍了 C 语言的基本概念、数据类型、运算符和表达式、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、编译预处理、指针、结构体和共用体、文件、位运算。

本书既可作为高等学校各专业 C 语言程序设计课程的正式教材, 也可作为计算机等级考试教材、计算机爱好者自学用书及各类工程技术人员的参考资料。

图书在版编目 (CIP) 数据

C 语言程序设计/恰汗·合孜尔, 单洪森编著. —北京: 中国铁道出版社, 2005. 7 (2006. 8 重印)
(21 世纪大学计算机基础规划教材)

ISBN 7-113-06593-7

I. C… II. ①恰… ②单… 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 090564 号

书 名: C 语言程序设计

作 者: 恰汗·合孜尔 单洪森

出版发行: 中国铁道出版社 (100054, 北京市宣武区右安门西街 8 号)

策划编辑: 严晓舟 杨东晓

责任编辑: 严 力 翟玉峰 荆 波

封面制作: 白 雪

印 刷: 三河市国英印务有限公司

开 本: 787×1092 1/16 印张: 16.75 字数: 395 千

版 本: 2005 年 8 月第 1 版 2006 年 8 月第 2 次印刷

印 数: 5 001~7 000 册

书 号: ISBN 7-113-06593-7/TP·1546

定 价: 23.00 元

版权所有 侵权必究

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

前 言

计算机应用能力是 21 世纪人才不可缺少的基本素质。程序设计是各专业计算机应用能力培养的重要技术基础，C 语言是目前国内外广泛使用的一种程序设计语言，是国内外大学讲述程序设计方法的首选语言。

全国计算机等级考试、全国计算机应用技术证书考试（NIT）和全国各地区组织的大学生计算机等级考试都已将 C 语言列入了考试的范围。许多人已经用 C 语言编写应用软件，学习 C 语言已成为广大青年学生的迫切愿望和要求。

教材是知识传授和能力培养的基础，C 语言的概念比较复杂，规则较多，使用灵活，容易出错，不少初学者感到困难。本书从程序设计的实际能力培养出发，由浅入深、深入浅出，将理论与实践有机结合，融知识传授和能力培养于一体。本书内容丰富、注重实践；突出重点、分散难点；例题丰富、结合实际；图文并茂、文字流畅。

本书的特色之一是例题丰富，并且例题涉及的面十分广泛，力求通过实际例题的讲解，逐步提高学生编写程序的能力。通过例题，介绍了程序设计常用的各种算法和各种实际问题的处理方法，并对各种方法进行了对比、分析，最后给出了最优的程序。从软件工程的角度出发，注重培养学生解决实际问题的能力。本书宗旨在于进一步巩固对基本知识的理解和掌握，提高学生的逻辑分析、抽象思维和程序设计的能力，培养学生良好的程序设计风格，进而具有编写大型程序的能力。对于准备参加计算机等级考试的各类人员，也可以通过对本书的学习，掌握各种问题的处理方法，以提高应试能力，顺利通过考试。

本书由工学博士恰汗·合孜尔教授组织编著，第 1~7、9~11 章由单洪森编写；第 8 章由田晓东、陈大春编写。感谢广大朋友对完成本书所给予的帮助。

为了便于教学和学习，与本书配套的《C 语言程序设计上机指导与习题集》将同时出版。

本书既可作为高等院校各专业 C 语言程序设计课程的正式教材，也可作为计算机等级考试教材、计算机爱好者自学用书及各类工程技术人员的参考资料。

由于编者水平有限，书中难免有错误和不妥之处，恳请同行和广大读者批评指正。

编 者
2005 年 7 月

目 录

第 1 章 C 语言概述	1
1-1 C 语言产生的历史背景	1
1-2 C 语言的特点	2
1-3 C 程序的基本组成	2
1-4 本章小结	4
第 2 章 C 语言的基础知识	6
2-1 数据类型	6
2-2 常量	6
2-2-1 整型常量	6
2-2-2 实型常量	7
2-2-3 字符常量	8
2-2-4 字符串常量	9
2-2-5 符号常量	9
2-3 变量	10
2-3-1 变量的概念	10
2-3-2 变量的定义与初始化	11
2-4 C 语言的运算符	14
2-4-1 C 运算符概述	14
2-4-2 C 运算符的优先级与结合性	15
2-5 算术运算符与算术表达式	16
2-5-1 算术运算符	16
2-5-2 自增与自减运算符	16
2-5-3 算术表达式	17
2-5-4 表达式中数据间的混合运算与类型转换	18
2-6 赋值运算符和赋值表达式	20
2-6-1 赋值运算符	20
2-6-2 复合赋值运算符	20
2-7 逗号运算符和逗号表达式	21
2-8 本章小结	21
第 3 章 顺序结构程序设计	23
3-1 C 语句概述	23
3-2 赋值语句	24
3-3 数据的输入/输出	25

3-3-1 字符数据的输入/输出.....	25
3-3-2 格式的输入/输出.....	26
3-4 程序举例.....	32
3-5 本章小结.....	33
习题.....	34
第 4 章 选择结构程序设计	36
4-1 关系运算符和关系表达式	36
4-1-1 关系运算符	36
4-1-2 优先次序	36
4-1-3 关系表达式	36
4-2 逻辑运算符和逻辑表达式	37
4-2-1 逻辑运算符	37
4-2-2 逻辑表达式	37
4-3 if 语句	38
4-3-1 if 语句的三种基本格式	38
4-3-2 if 语句的嵌套	40
4-4 条件运算符? :	41
4-5 switch 语句.....	42
4-6 程序举例.....	44
4-7 本章小结.....	47
习题.....	48
第 5 章 循环结构程序设计	49
5-1 概述.....	49
5-2 goto 语句及用 goto 构成的循环	49
5-2-1 无条件转移语句 goto.....	49
5-2-2 goto 和 if 构成的循环	49
5-3 while 循环	50
5-4 do...while 循环	51
5-5 for 循环	52
5-6 break 语句和 continue 语句.....	54
5-6-1 break 语句.....	54
5-6-2 continue 语句.....	54
5-7 多重循环	56
5-8 程序举例	59
5-9 本章小结	65
5-9-1 编写循环程序可能遇到的几个问题	65
5-9-2 循环结构程序设计	68

习题.....	69
第 6 章 数组.....	70
6-1 一维数组.....	70
6-1-1 一维数组的定义.....	70
6-1-2 一维数组的初始化.....	70
6-1-3 一维数组元素的引用.....	71
6-1-4 一维数组应用举例.....	71
6-2 二维数组.....	79
6-2-1 二维数组的定义.....	79
6-2-2 二维数组的存储结构.....	80
6-2-3 二维数组的初始化.....	80
6-2-4 二维数组元素的引用.....	81
6-2-5 二维数组应用举例.....	82
6-3 字符数组.....	86
6-3-1 字符数组的定义.....	86
6-3-2 字符数组的初始化.....	86
6-3-3 字符数组的引用.....	87
6-3-4 字符串和字符串的结束标志.....	88
6-3-5 字符数组的输入/输出.....	88
6-3-6 字符串处理函数.....	89
6-3-7 字符数组应用举例.....	92
6-4 本章小结.....	95
习题.....	96
第 7 章 函数和预处理命令.....	97
7-1 概述.....	97
7-2 函数的定义.....	97
7-3 函数的调用.....	99
7-3-1 函数调用的格式.....	99
7-3-2 函数调用的规则.....	99
7-3-3 函数调用的方式.....	99
7-3-4 函数的传值调用.....	100
7-4 函数的返回值.....	100
7-4-1 返回语句 <code>return</code>	100
7-4-2 <code>void</code> 类型函数.....	101
7-5 对被调函数的声明.....	102
7-6 函数的嵌套调用.....	104
7-7 函数的递归调用.....	106

7-7-1 递归模型	107
7-7-2 递归的执行过程	107
7-8 数组作为函数的参数	109
7-8-1 数组元素作函数的参数	109
7-8-2 数组名作函数的参数	110
7-8-3 多维数组名作函数的参数	111
7-9 局部变量和全局变量	112
7-9-1 局部变量	112
7-9-2 全局变量	113
7-9-3 外部变量的声明	115
7-10 变量的存储类别	116
7-10-1 内部变量的存储类型	117
7-10-2 外部变量的存储类型	119
7-10-3 存储类型小结	120
7-11 内部函数和外部函数	121
7-11-1 内部函数	121
7-11-2 外部函数	121
7-12 编译预处理	122
7-12-1 宏定义	122
7-12-2 文件包含	126
7-12-3 条件编译	128
7-13 程序举例	130
7-14 本章小结	135
习题	136
第 8 章 指针	138
8-1 地址和指针的概念	138
8-2 指向变量的指针变量	139
8-2-1 指向变量的指针变量的定义	139
8-2-2 指针运算符	139
8-2-3 指针变量的引用	140
8-2-4 指针变量的初始化	141
8-2-5 指针变量的运算	141
8-2-6 指针变量作函数的参数	144
8-3 数组的指针和指向数组的指针变量	146
8-3-1 指向数组元素的指针变量的定义	146
8-3-2 通过指针引用数组元素	147
8-3-3 数组名作为函数的参数	149
8-4 指向含有 n 个元素的一维数组的指针变量	152

8-4-1 指向含有 n 个元素的一维数组的指针变量	152
8-4-2 通过指向一维数组的指针变量引用多维数组的元素	154
8-5 字符串的指针和指向字符串的指针变量	156
8-5-1 字符串的指针	156
8-5-2 指向字符串的指针变量的定义	157
8-5-3 指向字符串的指针变量的初始化	157
8-6 函数的指针和指向函数的指针变量	161
8-6-1 指向函数的指针变量的定义	161
8-6-2 用指向函数的指针变量调用函数	161
8-6-3 函数的指针作函数的参数	162
8-7 返回指针的函数	164
8-8 指针数组	167
8-8-1 指针数组的定义	167
8-8-2 指针数组的初始化	168
8-8-3 指针数组作指针型函数的参数	170
8-9 指向指针的指针	171
8-9-1 指向指针的指针变量的定义	172
8-9-2 main()函数的参数	174
8-10 本章小结	176
习题	180
第 9 章 结构体和共用体	181
9-1 结构体	181
9-1-1 结构类型定义	181
9-1-2 结构体类型变量的说明	181
9-2 结构体变量的引用	183
9-2-1 结构体变量的赋值	183
9-2-2 结构体变量的初始化	184
9-3 结构体数组	184
9-4 结构体指针变量	186
9-4-1 结构体指针变量的说明	186
9-4-2 结构体指针变量的使用	186
9-5 动态存储分配	189
9-6 链表结构	190
9-6-1 链表概述	190
9-6-2 单向链表的基本操作	192
9-7 共用体	200
9-7-1 共用体的定义	201
9-7-2 共用体变量的说明	201

9-7-3 共用体变量的赋值和使用	202
9-8 枚举类型	203
9-8-1 枚举类型的定义和枚举变量的说明	203
9-8-2 枚举类型变量的赋值和使用	204
9-8-3 枚举变量的输入/输出	205
9-9 程序举例	206
9-10 typedef 定义新的类型标识符	207
9-11 本章小结	209
习题	211
第 10 章 文件	212
10-1 文件的基本概念	212
10-1-1 文件的分类	212
10-1-2 文件操作过程	213
10-1-3 文件缓冲区	213
10-2 文件类型指针	214
10-3 文件的打开与关闭	214
10-3-1 文件的打开函数 fopen	214
10-3-2 文件的关闭函数 fclose	216
10-4 文件的读写	216
10-4-1 字符读写函数	216
10-4-2 字符串读写函数	219
10-4-3 数据块读写函数	220
10-4-4 格式化读写函数	222
10-5 文件的随机读写	223
10-5-1 rewind 函数	223
10-5-2 fseek 函数	223
10-6 文件检测函数	224
10-6-1 文件结束检测函数 feof	224
10-6-2 读写文件出错检测函数 ferror	224
10-6-3 clearerr 函数	224
10-7 C 库文件	225
10-8 本章小结	225
习题	225
第 11 章 位运算	226
11-1 位运算符和位运算	226
11-1-1 位运算符	226
11-1-2 位运算	226

11-2 位段结构	231
11-2-1 位域的定义和位域变量的说明	231
11-2-2 位域的使用	232
11-3 本章小结	233
附录 A C 语言中的关键字	234
附录 B 部分字符的 ASCII 码对照表	235
附录 C 运算符的优先级与结合性	236
附录 D Turbo C 库函数	237
参考文献	253

第 1 章 C 语言概述

C 语言程序设计是高等院校计算机专业及相关专业重要的专业基础课，其目的是培养学生的程序设计理念、掌握程序设计的基本方法，为后继课程（数据结构、面向对象的程序设计和操作系统等）打下坚实的基础。

1-1 C 语言产生的历史背景

随着计算机应用的迅速发展，各种功能强大使用方便的高级语言相继出现，高级语言使用方便，可移植性好。但高级语言一般难以实现低级语言能够直接操作计算机硬件的功能（如对内存地址的操作、位操作等）。在这样的情况下，人们希望有一种语言既有高级语言使用方便的优点，又有低级语言能够直接操作计算机硬件的优点，因此，C 语言就应运而生。

C 语言是 1972 年由美国的 Dennis Ritchie 设计发明的，并首次在 UNIX 操作系统的 DEC PDP-11 计算机上使用。它由早期的编程语言 BCPL (Basic Combined Programming Language) 发展演变而来。在 1970 年，AT&T 贝尔实验室的 Ken Thompson 根据 BCPL 语言设计出较先进的并取名为 B 的语言，最后导致了 C 语言的问世。随着微型计算机的日益普及，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，美国国家标准研究所 (ANSI) 为 C 语言制定了一套 ANSI 标准，成为现行的 C 语言标准。

C 语言发展十分迅速，并成为最受欢迎的语言之一，主要因为它具有强大的功能。许多著名的系统软件，如 UNIX 操作系统就是由 C 语言编写的。用 C 语言加上一些汇编语言子程序，就更能显示 C 语言的优势了，像 PC-DOS、WORDSTAR 等就是用这种方法编写的。

Turbo C 是美国 Borland 公司的产品，Borland 公司是一家专门从事软件开发、研制的大公司。该公司相继推出了一套 Turbo 系列软件，如 Turbo Basic、Turbo Pascal、Turbo Prolog 等，这些软件很受用户欢迎。该公司在 1987 年首次推出 Turbo C 1.0 产品，其中使用了焕然一新的集成开发环境，即使用了一系列下拉菜单，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。1988 年，Borland 公司又推出 Turbo C 1.5 版本，增加了图形库和文本窗口函数库等，而 Turbo C 2.0 则是该公司 1989 年出品的。Turbo C 2.0 在原来集成开发环境的基础上增加了查错功能，并可以在 Tiny 模式下直接生成 .COM（数据、代码、堆栈处在同一 64KB 内存中）文件。还可对数学协处理器（支持 8087/80287/80387 等）进行仿真。Borland 公司后来又推出了面向对象的程序软件包 Turbo C++，它继承发展 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。1991 年为了适用 Microsoft 公司的 Windows 3.0 版本，Borland 公司又将 Turbo C++ 做了更新，开发出 Turbo C 的新一代产品：Borland C++。

1-2 C 语言的特点

C 语言的特点如下:

(1) C 语言是中级语言。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。

C 语言可以像汇编语言那样对位、字节和地址进行操作。

(2) C 语言是结构式语言。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。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 语言具有各种各样的数据类型,并引入了指针的概念,可使程序效率更高。另外 C 语言也具有强大的图形功能,支持多种显示器和驱动器。而且计算功能、逻辑判断功能也比较强大,可以实现决策目的。

(4) C 语言适用范围广。C 语言还有一个突出的优点就是适用于多种操作系统,如 DOS、UNIX,也适用于多种机型。

(5) 运算符极其丰富,数据处理能力强。C 语言的运算符包含的范围十分广泛,把括号、下标、赋值等都作为运算符处理,灵活使用这些运算符可以实现其他高级语言难以实现的运算。

(6) 数据结构丰富。C 语言的数据类型具有现代高级语言所具有的各种数据类型,如整型、实型、字符型、数组类型、结构体类型、共用体类型等,能够实现各种复杂数据结构的运算。尤其是 C 语言的指针类型,更为灵活多样。

(7) 语法限制不严格,程序设计自由度大。用 C 语言编写程序要求对程序设计更加熟练,对程序设计人员的要求更高(程序员必须对大部分语法错误负责)。

(8) 目标代码效率高。仅比汇编程序生成的目标代码执行效率低 10%~20%。

(9) 可移植性好(与汇编语言比较)。不做修改就可以在不同型号计算机、不同操作系统上执行。

1-3 C 程序的基本组成

【例 1.1】输出英文的我爱中国: I love China!。

```
main()
{ printf("I love China!\n");
}
```

该程序的作用是在屏幕上输出一行信息“I love China!”。

本例中,主函数 main 仅包含一个语句,该语句仅由 printf 输出函数构成。语句后面有一个分号。

【例 1.2】求两个数的和。

```
main()
{ int a,b,sum; /* 定义 a、b、sum 3 个整型变量 */
  a=123;b=456;
  sum=a+b;
  printf("sum is:%d\n",sum);
}
```

程序运行结果:

```
sum is: 579
```

本程序的作用: 求两个整数 a、b 之和 sum, 并在屏幕上输出 sum。

/* */中间的内容表示“注释”。注释是程序员对程序某部分的功能和作用所做的说明, 是方便读者理解程序的含义的, 对编译和运行不起作用。

语句 `int a,b,sum;` 定义 3 个整型变量。

语句 `a = 123;` 是一个赋值 (assign) 语句, 把常数 123 赋给变量 a。语句 `b = 456;` 也是一个赋值语句。该行包括两个语句 (在 C 语言中, 一个语句可以占多行, 一行也可以有多个语句)。

语句 `sum = a + b;` 计算 a、b 之和, 并把和值赋给变量 sum, 这是一个赋值语句。

`printf()` 输出 sum 的值, 输出格式用格式字符串 “%d” 指定。%d 表示整型格式, 在输出时, 该位置用对应变量的值代替。程序的输出信息为: sum is 579。

【例 1.3】 从键盘输入两个整数, 在屏幕上输出它们的最大值。

```
main()
{   int a,b,c;
    printf("input a,b=");
    scanf("%d,%d",&a,&b);
    c=max(a,b);
    printf("max is:%d\n",c);
}

int max(int x,int y)
{   int z;
    if(x>y) z=x;
    else z=y;
    return z;
}
```

程序运行结果:

```
input a,b=6,11
max is:11
```

该程序是由 main 和 max 两个函数组成的, 主函数 main 中定义了 a、b、c 3 个整型变量。

`printf("input a,b=");` 语句的作用是: 程序执行时显示字符串 "input a,b=", 提示操作员输入数据给变量 a 和 b 赋值。

`scanf("%d,%d",&a,&b);` 语句的作用是: 从终端输入两个整数, 分别赋值给 a 和 b。

`c=max(a,b);` 语句的作用是: 用 a 和 b 作实参调用函数 max, a 和 b 的值分别传递给 max 函数中的形式参数 x 和 y, max 函数的返回值将赋给 c。

`printf("max is:%d\n",c);` 语句的作用是: 输出 c 的值。

函数 max 中定义了整型变量 z, 如果 x 的值比 y 的值大, 将 x 的值赋给 z; 否则将 y 的值赋给 z, 最后返回 z 的值。

从以上的例子中可以看到:

(1) C 程序主要由函数构成, C 程序中有 3 种类型的函数:

① `main()`: 主函数, 开发系统提供的特殊函数, 每一个 C 程序必须有且只有一个 `main()` 函数。它代表程序开始执行的起始位置。

② 开发系统提供的函数, 如 `printf()`、`scanf()` 等。Turbo C 开发系统提供了 300 多个函数。

③ 用户自定义函数, 程序员自己设计的函数, 如 `max()`。

C 程序主要由函数构成, 这个特点有利于实现程序的结构化 (结构清晰)。

(2) 一个函数由两部分构成:

① 函数的说明部分, 包括: 函数名、函数类型 (返回值类型)、函数属性 (前面的例子中未使用函数的属性)。属性包括内部 (`static`) 和外部 (`extern`)、形式参数名、形式参数类型。

② 函数体, 函数体必须放在一对大括号 `{ }` 中。函数体中包含变量定义部分和执行部分。

例如:

```
int max(int x, int y)
{
    ...
}
```

`max` 是函数名, `max` 前面的 `int` 是函数类型, 表示 `max` 函数的返回值是整型, `x`、`y` 是形式参数, `x` 和 `y` 都是整型的。

(3) 一个程序总是从主函数开始执行的, 无论主函数写在源程序的什么位置。

(4) C 程序书写格式自由, 一个语句可以占多行, 一行也可以有多个语句。

(5) 语句数据定义后必须要有分号, 分号是 C 语句的必备成分。例如: `c = a + b;`。

(6) C 语言没有输入输出语句, 在 C 语言中输入输出操作是由函数实现的。

(7) 在 C 程序中, 可用 `/* */` 对程序进行注释。

1-4 本章小结

1. C 语言程序的构成、main 函数和其他函数

- 一个 C 语言源程序至少包含一个函数 (`main` 函数), 也可以包含一个 `main` 函数和若干个其他函数。
- 函数是 C 程序的基本单位, 被调用的函数既可以是系统的库函数, 也可以是用户根据需要自己编写的函数。可以说, C 语言是函数式的语言, 程序的全部工作都是由函数来完成的。
- 一个函数是由若干条语句组成的, 而函数的功能就是由组成它的若干条语句来实现的。
- 一个 C 语言源程序中的各个函数的位置是不固定的, C 语言程序总是从 `main()` 开始执行的。

2. 源程序的书写格式

- C 语言共有 32 个关键字, 每个关键字在 C 程序中都代表某一个固定含义, 所有关键

字都要用英文小写字母表示，且这些关键字都不允许作为用户标识符使用。

- C 语言用户标识符其实就是一个名字，它只能由英文字母（大写或小写）、数字（0~9）和下划线 3 种字符组成，且第一个字符必须是字母或下划线。注意，大写字母和小写字母被认为是两个不同的字符，习惯上变量名用小写字母表示。另外，用户标识符不允许使用现有的 C 语言关键字，因为这些关键字是 C 语言用来表示特定含义的。C 语言标识符的长度无统一规定，随系统而不同，许多系统取 8 个字符。假设程序中出现变量名长度大于 8 个字符，则只有前面 8 个字符有效，后面的字符不被识别，因此，在写程序时应对所用系统在标识符长度的规定方面有所了解。
- 在 C 语言中，大写字母和小写字母的含义是不同的，在用户自定义的标识符中是要区别大小写的。

3. C 语言的运行

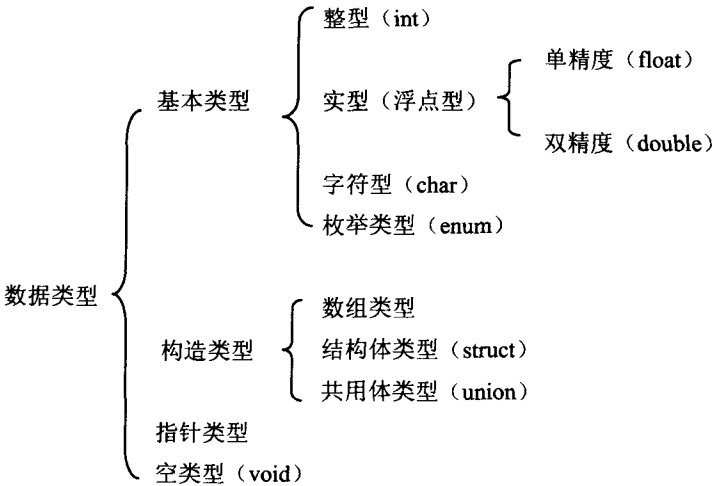
- C 语言是以编译方式执行的，用 C 语言编写的程序称为源程序，必须经过编译、连接生成可执行文件（.exe）才能执行。
- 掌握 Turbo C 的环境、基本使用方法。

第 2 章 C 语言的基础知识

2-1 数据类型

C 语言有丰富的数据类型，一种高级语言的数据类型越丰富，它处理问题的范围就越广，因而它的功能也就越强。

C 语言的数据类型如下：



C 语言有 4 种数据类型。基本类型是最简单的数据类型；构造类型的数据是由基本类型的数据按照一定的规律组成的更复杂的数据类型；指针类型是 C 语言的一个重要特色，是 C 语言的精华，有着广泛的用途；空类型（void）主要用于函数的返回。程序中用到的所有数据都必须指定其数据类型。

本章主要介绍基本类型的数据。

2-2 常量

常量是在程序运行中其值不变化的量。

2-2-1 整型常量

在 C 语言中整型常量可以用十进制、八进制、十六进制表示：

1. 十进制整数。如 241、-1 035、0。
2. 八进制整数。以 0 开头的整数是八进制整数。如 0123、0375。
3. 十六进制整数。以 0X 开头的整数是十六进制整数。如 0X123、-0X375、0XABC。

说明：

(1) 八进制整常数必须以 0 开头，即以 0 作为八进制数的前缀。数码取值为 0~7。八