

轻松幽默，浅显易懂，零门槛学习C语言，真的很简单
娓娓道来，亲切自然，老鸟带领菜鸟，快速掌握C语言编程



清华大学出版社



清华大学出版社
北 京

内 容 简 介

本书是一本与众不同的 C 语言学习读物,是一本化繁为简,把“抽象”问题“具体”化,把复杂问题简单化的书。在本书中,避免出现云山雾罩、晦涩难懂的讲解,代之以轻松活泼、由浅入深的剖析,这必将使每一位阅读本书的读者少走弯路,快速上手,从而建立学习 C 程序设计的信心。

本书 15 章,分为 5 篇,从实用出发,由遇到的问题引出解决问题的方法来系统讲述 C 语言的各个特性及程序设计的基本方法。本书内容主要包括常量、变量、程序结构、数组、字符串、指针、结构体、共同体、枚举类型、函数、局部变量和全局变量、预处理命令和文件等一些非常重要的知识。通过阅读本书,读者可以在较短的时间内理解 C 程序设计的各个重要概念和知识点,为进一步学习打好基础。

本书配带 1 张 DVD 光盘,收录了本书重点内容的教学视频和涉及的源代码,光盘中还赠送了大量超值的 C 语言进阶视频。

本书最适合没有基础的 C 语言入门新手阅读;对于有一定基础的读者,可通过本书进一步理解 C 语言的各个重要知识点和概念;对于大、中专院校的学生和培训班的学员,本书也不失为一本好教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C 语言入门很简单 / 马磊等编著. —北京:清华大学出版社, 2012.6

ISBN 978-7-302-28102-3

I. ①C… II. ①马… III. ①C 语言—程序设计—基本知识 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 030491 号

责任编辑:夏兆彦

封面设计:欧振旭

责任校对:徐俊伟

责任印制:何 芊

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:21.5 字 数:540 千字
(附光盘 1 张)

版 次:2012 年 6 月第 1 版

印 次:2012 年 6 月第 1 次印刷

印 数:1~5000

定 价:49.00 元

产品编号:045285-01

前 言

C 语言是一门历史悠久、博大精深的程序设计语言。它对计算机技术的发展起到了极其重要的促进作用，而且这种促进作用一直在持续并将继续持续下去。它从产生之时就肩负了很多重要使命，开发操作系统、开发编译器、开发驱动程序……它可深可浅，浅到你可以用几周的时间掌握它的基本概念和功能，深到几乎可以解决计算机中的大部分问题。

C 语言几乎是每一个致力于程序设计人员的必学语言。但从学习之初，它往往给读者以神秘而艰难的感觉。下面给出 C 语言入门新手的一些典型感受。

- ❑ 术语太难理解。C 语言对于没有基础的人来说比较抽象，因为一些专业术语对于初学者来说不好理解，更别说写程序了。
- ❑ 看不到界面。C 语言的编写是没有界面的，导致初学者很难理解写出来的程序是什么样，如何才能看出效果。
- ❑ 写的程序很长。对于实际开发人员，C 语言的使用概率比较小，要想实现某个效果，其程序很长，导致不容易完成。

但实际上，C 语言并非想象的那么难。它的很多优点让它一直保持着魅力而在程序语言中永葆青春。总结起来，主要体现在以下几个方面。

- ❑ C 语言是基础语言，容易理解，对初学者没有太大的限制。
- ❑ 它很灵活，一件事往往可以通过多种方式来实现。
- ❑ C 语言虽然没有界面，但是 C 程序语句看起来很直观，容易理解。
- ❑ C 语言没有那么多的库函数，没有“对象”与“类”之说，实现起来很方便。
- ❑ C 语言执行效率高，更多地执行了计算机底层的程序设计工作。
- ❑ 掌握了 C 语言，再学习其他程序设计语言往往比较容易。

本书即将展现的是一个简单的 C 语言，让 C 语言入门新手能在较短的时间内快速掌握 C 程序设计的基本思维和基础知识。本书和其他 C 语言图书的讲解方式有所不同。本书讲解时从实际出发，对 C 语言中的很多概念用生活中的例子进行类比。语言上力求幽默直白、轻松活泼，避免云山雾罩、晦涩难懂。讲解形式上图文并茂，由浅入深，抽丝剥茧。通过阅读本书，读者会少走很多弯路，会感觉到 C 语言其实没有想象的那么难。

本书特色

1. 语言幽默直白，轻松活泼，通俗易懂

本书避免使用那些艰涩难懂的术语云山雾罩地分析问题，代之以轻松活泼、幽默直白的语言讲解书中的每一个知识点。笔者力争让 C 语言的学习变得像看故事会一样通俗易懂。

2. 实例丰富，实用性强，并注重原理的讲解

本书结合大量生活中的实例，对 C 语言中的基本概念和知识做了深入浅出的讲解，并给出了大量生动形象的图示对程序的原理进行讲解，以加深读者的理解。

3. 图示丰富，容易理解

本书针对 C 语言中一些较难理解的概念，提供了大量的图示进行介绍，让读者以更加形象、直观的方式来理解所讲解的知识，从而达到更好的学习效果。

4. 举一反三

授之以鱼不如授之以渔。本书讲解时注重由此及彼，启发读者的思维，让读者通过已经掌握的知识进一步延伸到更深、更宽、更广的领域，从而达到举一反三的作用。

5. 习题丰富

本书每章后面都提供了有针对性的典型练习题，并给出了必要分析和实现的关键代码，以便于读者巩固和提高。

本书内容及体系结构

第1篇 一切从基础开始（第1~2章）

本篇简单讲述了计算机语言的相关基础知识，帮助大家计算机语言有个感性的认识，进而讲述了 C 语言的相关背景，并重点讲述了 C 语言的开发工具和学习经验。

第2篇 简单程序的构建（第3~4章）

本篇主要讲述了 C 语言的相关基本概念，作为学习 C 语言最基本的储备。所谓万丈高楼平地起，本篇知识掌握得好坏会直接影响后面章节的学习。

第3篇 复杂数据的表示（第5~10章）

本篇讲述了 C 语言中一些比较复杂的知识点，也可以称之为高级知识。这些看着稍微复杂的知识也正是 C 语言的核心，能否使用 C 语言进行游刃有余的开发，就看对这本篇内容的掌握和理解程度了。

第4篇 复杂功能的实现（第11~13章）

本篇主要讲述了在使用 C 语言进行实际开发时需要使用的知识点，掌握了本篇内容，就可以自己开始进行实际的编程开发了。

第5篇 C语言的高级内容（第14~15章）

本篇是对前面所有知识的一个总结，主要讲述了如何使用 C 语言进行文件操作。

本书读者对象

- ☐ 没有基础的 C 语言入门新手;
- ☐ 刚入职的初、中级程序员;
- ☐ 大、中专院校的学生;
- ☐ 相关培训学校的学员;
- ☐ C 语言开发爱好者。

本书作者

本书由马磊主笔编写。其他参与编写的人员有陈世琼、陈欣、陈智敏、董加强、范礼、郭秋滢、郝红英、蒋春蕾、黎华、刘建准、刘霄、刘亚军、刘仲义、柳刚、罗永峰、马奎林、马味、欧阳昉、蒲军。

阅读本书的过程中,如果有疑问或发现本书有任何纰漏,可与笔者联系。联系邮箱:
`xd_malei@163.com`。

编著者

目 录

第 1 篇 一切从基础开始

第 1 章 概述 (🔗 教学视频: 21 分钟)	2
1.1 C 语言简介	2
1.1.1 C 语言的位置	2
1.1.2 C 语言的优缺点	3
1.1.3 C 语言适合什么开发	4
1.2 C 语言的开发环境	4
1.2.1 编辑器、编译器和链接器	4
1.2.2 集成开发环境	6
1.3 Visual Studio 使用简介	8
1.3.1 Visual Studio 版本	8
1.3.2 Visual Studio 的安装	9
1.3.3 新建项目	9
1.3.4 编写代码	13
1.3.5 编译链接	15
1.3.6 运行可执行程序	16
1.4 如何学好 C 语言	17
1.5 小结	17
1.6 习题	17
第 2 章 开始 C 语言之旅 (🔗 教学视频: 22 分钟)	19
2.1 为什么要写代码	19
2.1.1 为什么要写程序	19
2.1.2 从本书开始学编程	20
2.1.3 从一个现实的例子开始	20
2.2 编程的核心——数据	21
2.2.1 数据从哪里来	21
2.2.2 数据的表示	23
2.2.3 数据类型面面观——精度和范围	23
2.2.4 C 语言基本数据类型	23
2.2.5 数据的变与不变——变量、常量	25
2.3 使用变量和常量	26
2.3.1 变量的使用	26
2.3.2 命名的方式	27

2.3.3 关键字	28
2.3.4 常量的使用	29
2.4 小结	30
2.5 习题	30

第 2 篇 简单程序的构建

第 3 章 简单数学运算 ( 教学视频: 44 分钟)	34
3.1 什么是赋值	34
3.1.1 赋值的作用——把数据存起来	34
3.1.2 赋值运算的形式	35
3.1.3 赋值表达式	35
3.1.4 机动灵活的赋值——scanf()	35
3.1.5 看看我们的劳动成果——printf()	36
3.1.6 赋值的重要性	37
3.2 开始赋值——整型赋值	38
3.2.1 整数在计算机中的表示——二进制	39
3.2.2 更先进的表示方法——八进制和十六进制	42
3.2.3 进制之间的转换——以二进制为桥梁	44
3.2.4 给整型赋值	45
3.3 浮点型赋值	48
3.3.1 小数在计算机中的表示	48
3.3.2 给浮点型赋值	52
3.4 字符型赋值	54
3.4.1 字符在计算机中的表示——ASCII	54
3.4.2 给字符赋值	55
3.5 类型转换	58
3.5.1 什么是类型转换	58
3.5.2 类型转换的利弊	59
3.5.3 隐式类型转换和显式类型转换	59
3.5.4 赋值中的类型转换	61
3.6 基本数学运算	64
3.6.1 数学运算和数学表达式	64
3.6.2 商与余数	67
3.6.3 位运算	68
3.6.4 优先级的奥秘	73
3.6.5 数学运算中的类型转换	77
3.7 复合赋值运算	79
3.7.1 复合赋值运算	79
3.7.2 自增自减运算——特殊的复合赋值	81
3.7.3 自增自减运算的使用	82
3.8 小结	83
3.9 习题	84

第 4 章 程序结构 (教学视频: 45 分钟)	88
4.1 语句和语句块	88
4.1.1 简单语句	88
4.1.2 语句块	89
4.2 变量的作用域	90
4.2.1 局部变量的声明定义位置规则	90
4.2.2 局部变量的作用域规则	91
4.2.3 嵌套语句块的同名变量作用域规则	91
4.3 最常见的语句执行结构——顺序结构	93
4.4 判断结构	94
4.4.1 判断的基础——逻辑真假	94
4.4.2 基础的判断——关系运算	95
4.4.3 复杂的判断——逻辑运算	97
4.5 if 判断结构	101
4.5.1 基本 if 结构	101
4.5.2 if-else 结构	102
4.5.3 另类的条件判断——?运算符的使用	104
4.5.4 if-else if-else 结构	106
4.5.5 嵌套的 if 结构	109
4.6 switch 判断结构	112
4.6.1 switch 基本结构	112
4.6.2 果断结束——break 的使用	114
4.7 循环结构	116
4.7.1 while 循环结构	116
4.7.2 for 循环结构	119
4.7.3 goto 语句	123
4.7.4 循环嵌套	124
4.7.5 break 和 continue	126
4.8 真正的程序——三种结构的揉和	129
4.9 小结	131
4.10 习题	131

第 3 篇 复杂数据的表示

第 5 章 数组 (教学视频: 39 分钟)	144
5.1 数组简介	144
5.1.1 数组的用途	144
5.1.2 数组变量的定义	145
5.2 数组变量初始化和赋值	146
5.2.1 数组的初始化	146
5.2.2 数组的下标	147
5.2.3 给数组赋值	147
5.2.4 数组元素的引用	148

5.3 二维数组	149
5.3.1 数组的维	150
5.3.2 二维数组表示和含义	150
5.3.3 二维数组的初始化	151
5.3.4 二维数组的赋值	152
5.3.5 二维数组的引用	154
5.4 多维数组	155
5.5 小结	157
5.6 习题	158
第 6 章 字符数组——字符串 ( 教学视频: 31 分钟)	161
6.1 字符数组	161
6.1.1 字符数组的表示	161
6.1.2 字符数组的初始化	161
6.1.3 字符数组的赋值和引用	162
6.2 字符串	164
6.2.1 字符串的 C 语言表示	164
6.2.2 使用字符串为字符数组初始化	164
6.2.3 字符串的保存形式	165
6.3 字符串的输入/输出——scanf 和 printf 字符串	166
6.3.1 输入/输出字符串的 C 语言表示	166
6.3.2 scanf() 函数对字符串的特殊处理	168
6.4 小结	169
6.5 习题	170
第 7 章 指针 ( 教学视频: 65 分钟)	172
7.1 地址的概念	172
7.1.1 地址的含义	172
7.1.2 为什么要用地址	173
7.1.3 地址的表示与取址运算	174
7.2 指针和指针变量	175
7.2.1 指针的含义和用途	175
7.2.2 指针类型	176
7.2.3 指针变量的定义和使用	177
7.2.4 void 指针	177
7.3 指针运算	179
7.3.1 取指针元素	179
7.3.2 指针的自增自减	181
7.3.3 指针的类型转换	183
7.4 数组和指针	186
7.4.1 数组名也是指针	186
7.4.2 数组名是指针常量	187
7.4.3 使用数组名访问数组元素	187
7.4.4 三种访问数组元素的方法	189
7.4.5 数组指针和指针数组	190

7.5 多重指针和多维数组	192
7.5.1 多重指针	192
7.5.2 取多重指针元素运算	194
7.5.3 多维数组名和各维元素	195
7.5.4 使用指针访问多维数组	197
7.6 字符串和指针	199
7.6.1 字符指针	199
7.6.2 字符指针和字符串	200
7.6.3 scanf()、printf()函数和字符指针	201
7.7 小结	203
7.8 习题	203
第8章 结构体 (教学视频: 35 分钟)	205
8.1 结构体的含义	205
8.2 结构体类型的表示	206
8.2.1 结构体类型的一般格式	206
8.2.2 结构体的成员变量	207
8.2.3 复杂的结构体	208
8.3 结构体变量	209
8.3.1 结构体变量的声明定义	209
8.3.2 结构体变量初始化	210
8.3.3 取结构体成员运算	211
8.4 结构体数组	213
8.5 结构体指针	215
8.5.1 一重结构体指针	215
8.5.2 使用结构体指针取结构体数据	216
8.5.3 结构体指针例子	217
8.6 回到问题	218
8.7 小结	218
8.8 习题	219
第9章 共同体类型 (教学视频: 36 分钟)	222
9.1 共同体的含义与表示	222
9.1.1 共同体的用途	222
9.1.2 共同体的表示	222
9.1.3 复杂的共同体	224
9.2 共同体变量	225
9.2.1 共同体变量	225
9.2.2 共同体成员变量的相互覆盖	225
9.2.3 使用共同体变量	226
9.3 共同体数组	228
9.4 共同体的指针	231
9.4.1 一重共同体指针类型	231
9.4.2 共同体指针变量	231
9.4.3 完整的例子	232

9.5 小结	233
9.6 习题	233
第 10 章 枚举类型 (教学视频: 35 分钟)	235
10.1 枚举类型的含义与表示	235
10.1.1 枚举类型的含义	235
10.1.2 枚举类型的表示	236
10.2 枚举常量和枚举变量	236
10.2.1 枚举常量	237
10.2.2 枚举变量的定义	239
10.2.3 枚举变量的使用	240
10.3 枚举数组和枚举指针	241
10.3.1 枚举数组	241
10.3.2 枚举指针	241
10.3.3 用枚举指针来访问枚举数组	242
10.4 typedef 类型定义符	243
10.5 小结	245
10.6 习题	245

第 4 篇 复杂功能的实现

第 11 章 函数 (教学视频: 50 分钟)	248
11.1 函数的意义	248
11.2 函数的形式	249
11.2.1 函数的一般形式	249
11.2.2 函数的参数列表	250
11.2.3 函数的返回值类型	251
11.3 函数的声明和定义形式	251
11.3.1 函数的声明	251
11.3.2 函数的定义形式	252
11.3.3 函数的形参	253
11.3.4 return 返回值语句	254
11.4 自己动手写一个函数——加法函数	255
11.4.1 确定加法函数的样子	255
11.4.2 实现加法函数体	256
11.4.3 完整的加法函数定义	256
11.5 函数调用	257
11.5.1 函数的调用作用	257
11.5.2 函数的调用表达式	258
11.5.3 函数的实参	258
11.5.4 简单函数的调用	259
11.6 复杂参数	260
11.6.1 数组参数	260
11.6.2 指针参数	262

11.6.3 结构体、共同体和枚举参数	265
11.7 小结	268
11.8 习题	268
第 12 章 特殊的函数——main()函数 (📺 教学视频: 32 分钟)	272
12.1 main()函数的作用	272
12.2 main()函数的声明定义	273
12.2.1 main()函数的声明形式	273
12.2.2 main()函数的参数	273
12.2.3 main()函数的返回值	275
12.3 小结	276
12.4 习题	276
第 13 章 局部变量和全局变量 (📺 教学视频: 39 分钟)	277
13.1 变量的作用域和生命周期	277
13.2 函数内的局部变量	278
13.2.1 局部变量的作用域	278
13.2.2 局部变量的生命周期	278
13.2.3 局部变量的覆盖作用	280
13.3 函数外的全局变量	282
13.3.1 全局变量的作用域	282
13.3.2 全局变量的生命周期	283
13.3.3 局部变量对全局变量的覆盖作用	284
13.4 变量修饰符	286
13.4.1 使用修饰符改变变量的作用域和生命周期	286
13.4.2 C 语言中常用变量修饰符的作用	286
13.5 小结	289
13.6 习题	289

第 5 篇 C 语言的高级内容

第 14 章 预处理命令、文件包含 (📺 教学视频: 48 分钟)	292
14.1 预处理命令的作用	292
14.1.1 程序预处理	292
14.1.2 预处理命令	293
14.1.3 C 语言的几类预处理命令	294
14.2 C 语言中的宏定义	294
14.2.1 C 语言的宏定义形式	294
14.2.2 不带参的宏定义	295
14.2.3 带参的宏定义	296
14.3 预编译控制	298
14.3.1 C 语言预编译控制	298
14.3.2 三种预编译控制组合形式	299
14.3.3 一个简单的例子	300

14.4 文件包含	301
14.4.1 头文件和源文件的文件名	301
14.4.2 头文件和源文件的内容	302
14.5 include 包含头文件	303
14.5.1 自定义头文件和系统头文件	303
14.5.2 文件包含的两种形式	303
14.5.3 完整的 circle 例子	304
14.5.4 C 语言中的标准头文件	306
14.6 小结	308
14.7 习题	309
第 15 章 文件操作 (教学视频: 47 分钟)	313
15.1 文件	313
15.1.1 重新认识文件	313
15.1.2 计算机眼里的文件	314
15.1.3 开发人员能对文件干些什么	315
15.2 文件的打开与关闭	315
15.2.1 文件指针	315
15.2.2 文件打开函数	316
15.2.3 文件关闭函数	318
15.3 文件读写	318
15.3.1 读写一个字符	318
15.3.2 读写一个字符串	319
15.3.3 读写一个数据块	320
15.4 文件的其他操作	322
15.4.1 随机读写文件	322
15.4.2 出错检验	323
15.5 小结	325
15.6 习题	325
附录 A ASCII 码表	327

第 1 篇 一切从基础开始

» 第 1 章 概述

» 第 2 章 开始 C 语言之旅

第 1 章 概 述

作为本书的开始，我们不涉及 C 语言的语法和使用细节，先来看一看这门语言的相关背景，以及其在计算机程序中举足轻重的地位。然后，为给后面的学习作准备，来看看如何建立 C 语言的开发环境。最后，抛砖引玉，说说笔者自己的 C 语言学习经验。

1.1 C 语言简介

本节先来看一看 C 语言的相关背景介绍。在众多的计算机语言之中，它到底处于一个什么样的位置？相对于其他计算机语言而言，它有什么优势和弱点？另外，C 语言适合做些什么样的软件开发？带着这些问题开始本节的学习。

1.1.1 C 语言的位置

图 1.1 是计算机语言发展过程的一个简单的示意图，只展示了一些主流的计算机语言的出现时间和顺序，不是很全，但是足以显示 C 语言与其他语言的关系了。

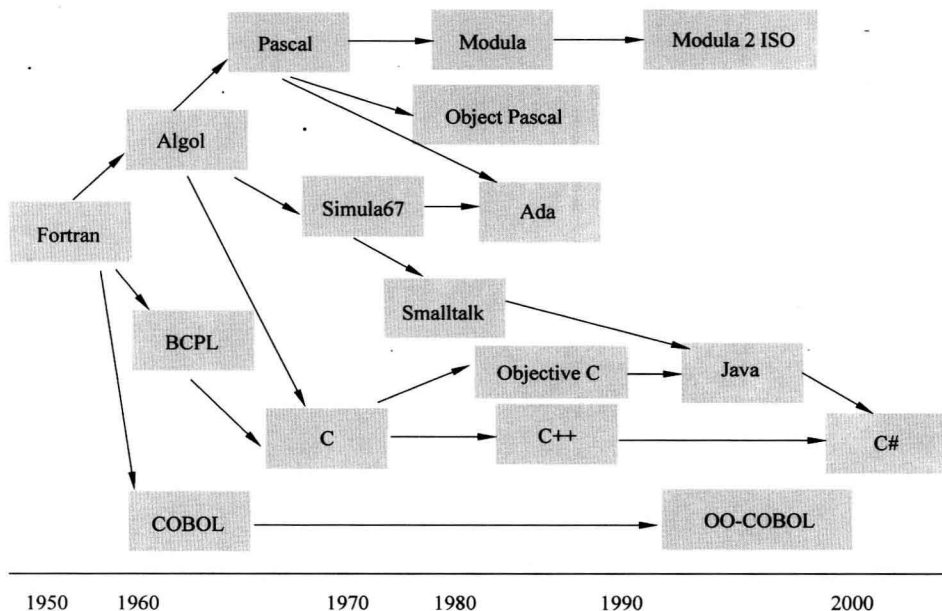


图 1.1 计算机语言发展史（简）

从图 1.1 中可以看出，所有的高级语言都起于 Fortran，之后就有不同的分支了。C 语言也是其中的一个分支，主要起源于 BCPL 语言（Basic Combined Programming Language），是对 BCPL 语言的一种简化。C 语言之所以被称为 C 语言，是因为 BCPL 语言被称为 B 语言，C 语言不能再取 BCPL 的首字母了，因此，取 BCPL 的第二个字母 C，C 语言因此得名。

C 语言诞生于 20 世纪 70 年代，之后不断完善、标准化。目前流行的 C 语言编译系统大多数是以 ANSI C（美国国家标准协会（ANSI）对 C 语言发布的标准）为基础进行开发的。但不同版本的 C 编译系统所实现的语言功能和语法规则略有差别。现在比较通用的是 1990 年，国际标准化组织 ISO（International Organization for Standards）接受的 1989 年的 ANSI C 标准，并以它作为 ISO C90 标准——ISO/IEC9899:1990。

1.1.2 C 语言的优缺点

任何一种计算机语言都有它的优势和不足，C 语言也是如此。不要以为 C 语言都那么老了，是不是没用了，也不要以为 C 语言是万能钥匙，什么问题都能解决好。

1. C 语言的优势

C 语言的优点细数起来，估计手指加脚趾都不够用，但是这些优点可以总结为一点，那就是“灵活”。C 语言的灵活主要体现在同一件事情可以通过好几种方式实现，并不会定死什么问题非得用什么方式来解决。C 语言之所以灵活，是因为它有以下这些属性，保证了它天生就是一种灵活的语言。

- ❑ 结构丰富多变：C 语言提供了三种基本的程序设计结构，通过这三种程序设计结构，就能够完成所有的计算机逻辑。而且每种结构中，又有略有差别的不同的形式，你可以选择自己喜欢的任意形式来完成需要的功能。
- ❑ 提供了多种基本运算：C 语言提供了 30 多种运算符号，分为 15 个运算优先级，不仅可以完成基本的数学运算，还可以完成类似于计算机底层操作的位运算。有的运算符号在不同情况下的含义和使用方式也是不同的，而且可以通过强制结合来改变运算符的优先级。
- ❑ 丰富的数据类型：数据类型决定了一种语言可以用来操作什么样的数据。C 语言的数据类型真可谓丰富，从基本的几种数据类型，到复杂数据类型，再到可以自定义的数据类型。从某种程度上来说，C 语言可以用来操作任何类型的数据。
- ❑ 程序设计自由，语法限制不大：C 语言代码在书写的时候，从书写格式到代码组织限制都不是很大，所以写代码的风格可谓百花齐放，什么样的都有。只要符合最基本的要求，无论你怎么写，都是对的。只不过代码风格不好，不利于阅读和理解。

对于上面提到的 C 语言的各种优点，在没有接触 C 语言之前，可能大家还体会不到。不要紧，相信大家学完这本书之后，回过头来看看这一部分内容，一定会有所感悟的。

2. C 语言的不足

一个事物，往往最强的地方也就是它最弱的地方。C 语言的弱点也正是由于它的“灵活”造成的。因为太灵活了，怎么样都行，对计算机的控制太过自由，稍不留意就会出现错误！所以，能灵活运用 C 语言，也是一种能力啊。