

C语言

综合项目实战

C Programming
in Action

叶安胜 鄢涛 主编



科学出版社

C 语言综合项目实战

叶安胜 鄢涛 主编

科学出版社

北京

内 容 简 介

本书内容包括两部分，第一篇：磨刀不误砍柴工。主要介绍 C 语言的基本概述、发展历史以及能做什么，并对当前主流的开发平台 VC++ 和 VS.NET 2013 平台下常规项目的创建进行了介绍。第二篇：绝知此事要躬行。该篇是本书的核心，按照 C 语言本身的知识体系，包括程序流程控制、大项目组织、递归及应用、数组、结构、指针、文件、数据库及图形处理九大知识框架，通过 9 个实战项目，将 C 语言结构化程序设计思想充分应用于这九大类别实战项目中。每个实战项目包括问题的描述、算法分析、流程设计以及主要功能的实现代码等，展示了一个典型项目的完整开发过程。

本书适合作为大专院校各专业层次学生，以提升 C 语言程序设计的动手实践与编程能力。通过对本书各项目实战的练习，读者能够进一步复习、巩固和掌握 C 语言程序设计的编程应用和解决实际问题的能力。

图书在版编目 (CIP) 数据

C 语言综合项目实践/叶安胜, 鄢涛主编. —北京: 科学出版社,
2015. 3

ISBN 978-7-03-043550

I. ①C… II. ①叶… ②鄢… III. ①C 语言—程序设计
IV. ①TP312

中国版本图书馆 CIP 数据核字 (2015) 第 042709 号

责任编辑: 杨 岭 冯 铂/责任校对: 韩雨舟

责任印制: 余少力/封面设计: 墨创文化

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

成都创新包装印刷厂印刷

科学出版社发行 各地新华书店经销

*

2015 年 2 月第 一 版 开本: 16 (787×1092)

2015 年 2 月第一次印刷 印张: 19.75

字数: 600 000

定价: 48.00 元

(如有印装质量问题, 我社负责调换)

《C 语言综合项目实战》

编委会

主 编：叶安胜 鄢 涛

副主编：蒲 强 古沐松 周晓清 张修军

编 委：叶安胜 鄢 涛 蒲 强 古沐松 周晓清 刘永红 赵卫东
张修军 高朝邦 于 曦 张志强 苏长明 段林涛 刘 昶

前 言

C语言是一种非常出色的程序设计语言，用它编制出来的程序短小精练，设计思路灵活多样，广泛应用于程序开发，特别是系统驱动程序、嵌入式领域、操作系统开发等等。对于大多数初学者而言学习C语言感觉抽象、难理解，基本语法都知道，但真正面临实际业务、解决实际问题、动手编写程序时就显得无从下手。

本书以 Visual C++ 6.0 和 VS.NET 2013 作为集成开发平台，提供的所有实战项目源程序均在此环境下调试通过。C语言程序设计是一门实践性很强的基础课程，本书提供大量实战项目的基本算法思想及编程思路，建议学习者多阅读程序，模仿经典算法，然后改写程序，最后应用于解决实际问题。只有通过大量的实战项目及编程训练，逐步掌握C语言编程特点及编程思路，进而提高自己的开发应用能力。本书作者长期从事高校C、C++和Java等语言开发类课程的教学，同时从事相关软件项目开发多年，具有丰富的项目开发经验，因此本书的项目实战为广大学习者提供了重要的指引作用。

全书在内容组织上分为两部分，共两篇。

第一篇：磨刀不误砍柴工

该篇共四章内容，主要介绍C语言的基本概述，发展历史、C语言能做什么及开发过什么系统，并对当前主流的开发平台VC++和VS.NET 2013下项目的创建、调试技巧及编程规范进行了介绍。

第二篇：绝知此事要躬行

该篇是本书的核心，共包括9章内容，这些内容按照C语言本身的知识体系，包括程序流程控制、大项目组织、递归及应用、数组、结构、指针、文件、数据库及图形处理等九大知识体系，通过九大类实战项目，将C语言结构化程序设计思想充分应用于这9个实战项目当中。每个实战项目包括问题的描述、算法分析、流程设计以及主要功能的实现代码等，展示了典型项目的完整开发过程。

本书由叶安胜、鄢涛担任主编，蒲强、古沐松、周晓清、张修军担任副主编，参与者还有刘永红、赵卫东、高朝邦等。各章编写工作如下：第一篇：磨刀不误砍柴工，由叶安胜编写完成，周晓清完成其中第2章的编写，高朝邦参与第4章编写。第二篇：绝知此事要躬行，实战1由古沐松编写，实战2和实战3由周晓清和叶安胜共同编写，实战4由周晓清、张修军和鄢涛共同编写，实战5、实战6由鄢涛编写，实战7由刘永红和于曦编写，实战8由赵卫东和刘昶编写，其中蒲强参与实战5，实战6部分内容编写，实战9由蒲强编写。全书由鄢涛和叶安胜汇总统筹规划修改定稿。

在本书的编写过程中得到了成都大学信息科学与技术学院领导和老师们的关心与支持，于曦、张志强、苏长明、段林涛、刘昶以及.NET开发课程群老师等提出了许多宝贵的意见，在此对以上老师表示衷心感谢。此外，在本书的编写过程中，还参考了大量

的文献资料，在此谨向这些文献资料的作者表示感谢。

由于时间仓促和水平所限，书中的疏漏之处在所难免，恳请各位专家和读者不吝批评和指正。

前 言

编者

随着计算机科学的飞速发展，C语言作为一门重要的编程语言，在软件开发领域占据了举足轻重的地位。本书旨在为读者提供一本全面、实用的C语言编程指南，帮助读者快速掌握C语言的核心概念和编程技巧。

本书共分10章，系统地介绍了C语言的基础知识、数据类型、运算符、控制语句、函数、指针、数组、字符串、结构体和联合体、以及预处理等。本书力求做到概念清晰、重点突出、循序渐进，力求使读者在较短的时间内，能够掌握C语言的基本编程方法，并能熟练地运用C语言进行编程。

本书可作为高等院校计算机专业及相关专业的教材，也可供从事计算机工作的工程技术人员参考。

2014年10月于成都

本书在编写过程中，参考了国内外大量的文献资料，并得到了许多同行专家的指导和帮助。在此，谨向这些专家和同行表示衷心的感谢。

由于编者水平有限，书中难免存在不足之处，恳请各位专家和读者不吝批评和指正。

本书共分10章，系统地介绍了C语言的基础知识、数据类型、运算符、控制语句、函数、指针、数组、字符串、结构体和联合体、以及预处理等。本书力求做到概念清晰、重点突出、循序渐进，力求使读者在较短的时间内，能够掌握C语言的基本编程方法，并能熟练地运用C语言进行编程。

本书可作为高等院校计算机专业及相关专业的教材，也可供从事计算机工作的工程技术人员参考。

本书在编写过程中，参考了国内外大量的文献资料，并得到了许多同行专家的指导和帮助。在此，谨向这些专家和同行表示衷心的感谢。

由于编者水平有限，书中难免存在不足之处，恳请各位专家和读者不吝批评和指正。

目 录

第一篇 磨刀不误砍柴功	1
第1章 C语言也能干大事	3
1.1 C语言的发展历史	3
1.1.1 C语言早期发展	3
1.1.2 K&R C	4
1.1.3 ANSI C 和 ISO C	4
1.1.4 C99	5
1.2 为什么要学习 C 语言	5
1.2.1 C语言通用性强	5
1.2.2 C语言的排行榜	5
1.3 C语言能够做什么	7
第2章 C语言的编译环境	11
2.1 Visual C++6.0 快速入门	11
2.1.1 Visual C++6.0 集成开发环境	11
2.1.2 程序调试	17
2.2 Visual Studio 2013 快速入门	21
2.2.1 Visual Studio 2013 的安装	21
2.2.2 Visual Studio 2013 开发环境	23
2.2.3 程序调试	28
第3章 C语言的程序结构	33
3.1 第一个 C 程序: Hello World	33
3.2 C语言的程序结构	34
第4章 C语言的规范	36
4.1 软件编程规范概述	36
4.2 C代码排版	37
4.3 C代码注释	38
4.4 C代码编码规范	39
第二篇 绝知此事要躬行	43
实战1 程序流程	45
1.1 输入三个实数,判断能否构成三角形	45
1.1.1 项目功能需求	45
1.1.2 知识点分析	45

1.1.3	算法思想	45
1.1.4	系统流程图	46
1.1.5	项目实施	47
1.1.6	项目扩展	47
1.2	简单的四则计算器	47
1.2.1	项目功能需求	47
1.2.2	知识点分析	48
1.2.3	算法思想	48
1.2.4	系统流程图	49
1.2.5	项目实施	50
1.2.6	项目扩展	51
1.3	猴子吃桃问题(循环实现)	51
1.3.1	项目功能需求	51
1.3.2	知识点分析	51
1.3.3	算法思想	52
1.3.4	系统流程图	52
1.3.5	项目实施	53
1.3.6	项目扩展	53
1.4	拓展项目	54
实战2	递归及应用	58
2.1	递归思想	58
2.1.1	问题由来	58
2.1.2	递归思想	59
2.1.3	简单递归问题求解	60
2.2	猴子吃桃问题	61
2.2.1	项目功能需求	61
2.2.2	知识点分析	61
2.2.3	算法思想	62
2.2.4	系统流程图	63
2.2.5	项目实施	64
2.2.6	项目扩展	64
2.3	汉诺塔问题	65
2.3.1	项目功能需求	65
2.3.2	知识点分析	65
2.3.3	算法思想	65
2.3.4	系统流程图	67
2.3.5	项目实施	67
2.3.6	项目扩展	68

2.4	拓展项目	68
2.4.1	猴子摘桃问题	68
2.4.2	其他拓展项目	71
实战3	大型项目组织	72
3.1	程序菜单实现	72
3.1.1	菜单设计	72
3.1.2	菜单框架	74
3.2	VC++下典型项目组织	76
3.2.1	项目的功能需求	76
3.2.2	项目框架搭建	77
3.2.3	项目实现	78
3.2.4	运行结果展示	83
3.3	VS.NET下大项目组织	85
3.3.1	项目功能需求	85
3.3.2	知识点分析	86
3.3.3	算法思想	86
3.3.4	系统流程图	90
3.3.5	项目实现	91
3.3.6	项目扩展	96
实战4	数组及应用	97
4.1	约瑟夫(Joseph)问题(数组)	97
4.1.1	项目功能需求	98
4.1.2	知识点分析	98
4.1.3	算法思想	98
4.1.4	系统流程图	99
4.1.5	项目实现	100
4.1.6	项目扩展	101
4.2	分类统计输入字符串中各类字符个数	103
4.2.1	项目功能需求	103
4.2.2	知识点分析	103
4.2.3	算法思想	103
4.2.4	系统流程图	104
4.2.5	项目实现	105
4.2.6	项目扩展	106
4.3	对输入的N个单词排序	107
4.3.1	项目功能需求	107
4.3.2	知识点分析	107
4.3.3	算法思想	107

4.3.4	系统流程图	108
4.3.5	项目实施	108
4.3.6	项目扩展	110
4.4	拓展项目	111
实战5	结构体及应用	114
5.1	学生信息登记管理(结构体+数组)	114
5.1.1	项目功能需求	114
5.1.2	知识点分析	117
5.1.3	算法思想	118
5.1.4	系统流程图	119
5.1.5	项目实施	120
5.1.6	项目扩展	127
5.2	火车票系统	127
5.2.1	项目功能需求	127
5.2.2	知识点分析	129
5.2.3	系统设计思想	130
5.2.4	系统流程图	132
5.2.5	项目实施	132
5.2.6	项目扩展	147
5.3	拓展项目	148
实战6	指针及应用	150
6.1	约瑟夫(Joseph)问题(指针)	150
6.1.1	项目功能需求	150
6.1.2	知识点分析	153
6.1.3	算法思想	153
6.1.4	系统流程图	154
6.1.5	项目实施	155
6.1.6	项目扩展	162
6.2	学生信息登记管理(结构体+链表)	162
6.2.1	项目功能需求	162
6.2.2	知识点分析	165
6.2.3	算法思想	166
6.2.4	系统流程图	170
6.2.5	项目实施	171
6.2.6	项目扩展	179
6.3	贪吃蛇游戏	180
6.3.1	项目功能需求	180
6.3.2	知识点分析	181

6.3.3 游戏设计要点和主要功能实现	181
6.3.4 系统流程图	183
6.3.5 项目实施	183
6.3.6 项目扩展	192
6.4 拓展项目	192
实战7 文件操作及应用	194
7.1 简单计数器(程序运行次数统计)	194
7.1.1 项目功能需求	194
7.1.2 知识点分析	195
7.1.3 算法思想	195
7.1.4 系统流程图	195
7.1.5 项目实施	196
7.1.6 项目扩展	198
7.2 学生学籍管理系统(结构体+文件+综合)	198
7.2.1 项目功能需求	198
7.2.2 知识点分析	200
7.2.3 算法思想	200
7.2.4 系统流程图	204
7.2.5 项目实施	205
7.2.6 项目扩展	224
7.3 拓展项目	224
7.3.1 学校运动会管理系统	224
7.3.2 图书管理系统	224
7.3.3 飞机订票管理系统	225
7.3.4 工资管理系统	225
7.3.5 学生选课管理系统	226
实战8 数据库操作及应用	227
8.1 简单计数器(程序运行次数统计)	227
8.1.1 项目功能需求	227
8.1.2 知识点分析	227
8.1.3 算法思想	228
8.1.4 系统流程图	229
8.1.5 项目实施	229
8.2 学生学籍管理系统(结构体+数据库+综合)	234
8.2.1 项目功能需求	234
8.2.2 知识点分析	236
8.2.3 算法思想	236
8.2.4 系统流程图	237

8.2.5	项目实现	237
8.2.6	项目扩展	248
8.3	拓展项目	248
实战9	图形操作及应用	250
9.1	贪吃蛇游戏	250
9.1.1	项目功能需求	250
9.1.2	知识点分析	251
9.1.3	算法思想	252
9.1.4	系统流程图	254
9.1.5	项目实施	254
9.1.6	项目扩展	263
9.2	俄罗斯方块	264
9.2.1	项目功能需求	264
9.2.2	知识点分析	264
9.2.3	算法思想	265
9.2.4	系统流程图	266
9.2.5	项目实施	266
9.2.6	项目扩展	282
9.3	拓展项目	282
附录一	ASCII码表	284
附录二	C语言常用库函数	285
附录三	C语言图形处理	291
附录四	EGE图形库常用函数	294
参考文献		301

第一篇

磨刀不误砍柴功

第 1 章 C 语言也能干大事

目前流行的计算机编程语言有 C 语言、C++、Java、C#、PHP、JavaScript 等，每种语言都有自己的特点，例如：C 语言是较早开发的一种高级语言，后来很多程序设计语言都是以 C 为蓝本进行设计的；C 语言和 C++ 主要用来开发系统软件；Java 和 C# 不但可以用来开发桌面软件，还可以用来开发网站后台程序；PHP 主要用来开发网站后台程序；JavaScript 主要负责网站的前端开发工作，等等。

C 语言语法简单精炼，概念少、效率高，包含了基本的编程元素，对当今流行的语言(C++、Java 等)均有参考作用，说 C 语言是现代编程语言的开山鼻祖毫不夸张，它改变了编程世界。正是由于 C 语言的简单，对初学者来说，学习成本小、时间短，结合本教程，能够快速掌握编程技术。毫不夸张地说，C 语言是学习编程的第一门语言，你不用考虑其他的选择，也许你将来的工作或学习不会使用 C 语言，但是它能让你了解编程相关的概念，带你走进编程的大门，以后学习其他语言，自然会触类旁通，很快上手，短时间内学会一门新计算机语言决不是神话。

1.1 C 语言的发展历史

C 语言早在 19 世纪 70 年代初问世，1978 年美国电话电报公司(AT&T)贝尔实验室正式发布 C 语言，后由美国国家标准局(American National Standards Institute，简称 ANSI)制定了一套 C 语言标准，于 1983 年发表，通常称之为 ANSI C。

让我们先了解一下 C 语言的发展历史。

1.1.1 C 语言早期发展

C 语言之所以命名为 C，它的原型是 ALGOL 60 语言，也被称为 A 语言。

1963 年，剑桥大学将 ALGOL 60 语言发展成为 CPL(Combined Programming Language)语言。

1967 年，剑桥大学的 Martin Richards 对 CPL 语言进行了简化，于是产生了 BCPL 语言。

1970 年，美国贝尔实验室的 Ken Thompson 将 BCPL 进行了修改，并为它起了一个有趣的名字“B 语言”，意思是将 CPL 语言煮干，提炼出它的精华，并且他用 B 语言写了第一个 UNIX 操作系统。

1973 年，B 语言也被人“煮”了一下，美国电话电报公司(AT&T)贝尔实验室开始了 C 语言的最初研发，丹尼斯·里奇(Dennis Ritchie)(参见图 1-1-1)在 B 语言的基础上最终设计出了一种新的语言，他取了 BCPL 的第二个字母作为这种语言的名字，这就是

C 语言。据丹尼斯·里奇(Dennis Ritchie)说, C 语言最重要的研发时期是在 1972 年。



图 1-1-1 C 语言之父丹尼斯·里奇(Dennis Ritchie)

注: 丹尼斯·里奇(Dennis Ritchie), C 语言之父, UNIX 之父。1978 年与布莱恩·科尔尼干(Brian Kernighan)一起出版了名著《C 程序设计语言(The C Programming Language)》, 现在此书已翻译成多种语言, 成为 C 语言方面最权威的教材之一。2011 年 10 月 12 日(北京时间为 10 月 13 日), 丹尼斯·里奇去世, 享年 70 岁。

C 语言的诞生是和 UNIX 操作系统的开发密不可分的, 原先的 UNIX 操作系统都是用汇编语言写的, 1973 年 UNIX 操作系统的核心用 C 语言改写, 从此以后, C 语言成为编写操作系统的主要语言。

1.1.2 K&R C

1978 年, 丹尼斯·里奇(Dennis Ritchie)和布莱恩·科尔尼干(Brian Kernighan)出版了一本书, 名叫《The C Programming Language》(中文译名为《C 程序设计语言》)。这本书被 C 语言开发者们称为“K&R”, 很多年来被当作 C 语言的非正式的标准说明。人们称这个版本的 C 语言为“K&R C”。

1988 年丹尼斯·里奇(Dennis Ritchie)和布莱恩·科尔尼干(Brian Kernighan)修改此书, 出版了《The C Programming Language》第二版, 第二版涵盖了 ANSI C 语言标准。第二版从此成为大学计算机教育有关 C 语言的经典教材, 多年后也没再出现过更好的版本。

1.1.3 ANSI C 和 ISO C

20 世纪 70 到 80 年代, C 语言被广泛应用, 从大型主机到小型微机, 也衍生了 C 语言的很多不同版本。为统一 C 语言版本, 1983 年美国国家标准局成立了一个委员会, 来制定 C 语言标准。1989 年 C 语言标准得到批准, 被称为 ANSI X3.159-1989 “Programming Language C”。这个版本的 C 语言标准通常被称为 ANSI C。又由于这个版本是 1989 年发布的, 因此也被称为 C89。后来 ANSI 把这个标准提交到 ISO(国际化标准组织), 1990 年被 ISO 采纳为国际标准, 称为 ISO C。又因为这个版本是 1990 年发布的, 因此也被称为 C90。

ANSI C(C89)与 ISO C(C90)内容基本相同,主要是格式组织不一样。因为 ANSI 与 ISO 的 C 标准内容基本相同,所以对于 C 标准,可以称为 ANSI C,也可以说是 ISO C,或者 ANSI/ISO C。

注意:以后大家看到 ANSI C、ISO C、C89、C90,要知道这些标准的内容都是一样的。目前,几乎所有的开发工具都支持 ANSI/ISO C 标准。这是 C 语言用得最广泛的一个标准版本。

1.1.4 C99

在 ANSI C 标准确立之后,C 语言的规范在很长一段时期内都没有大的变动。1995 年 C 程序设计语言工作组对 C 语言进行了一些修改,成为后来 1999 年发布的 ISO/IEC 9899:1999 标准,通常被称为 C99。但是各个公司对 C99 的支持所表现出来的兴趣不同。当 GCC 和其他一些商业编译器支持 C99 的大部分特性的时候,微软和 Borland 却似乎对此不感兴趣。

说明: GCC(GNU Compiler Collection, GNU 编译器集合)是一套由 GNU 工程开发的支持多种编程语言的编译器。

1.2 为什么要学习 C 语言

1.2.1 C 语言通用性强

C 语言语法简单精炼,灵活方便,效率高,可移植性好,包含了基本的编程元素,对初学者来说,学习成本小,时间短,能够快速掌握其编程技术。

C 语言是较早的一种编程语言,说它是现代编程语言的开山鼻祖毫不夸张,它改变了编程世界,后来推出的很多语言都参照了 C 语言。

C++ 和 Objective-C 以 C 语言为基础进行扩展,加入面向对象等高级特性。

PHP、Java、Python 的底层也都由 C 语言来实现(C 语言可以开发其他高级语言)。学会了 C 语言,再学习其他语言就会容易很多,达到“一通百通”的效果。C 语言比较接近计算机底层,能够直接操作硬件,执行效率高(比 C++、Java、Python 和 Ruby 都高),而且学习 C 语言,对于理解计算机体系结构也有很大的帮助。

C 语言应用广泛,可以用来开发桌面软件、硬件驱动、操作系统、单片机等,从微波炉到手机,从汽车到智能电视,都有 C 语言的影子。

真正的编程高手,不是会很多门语言,而是用一门语言可以干所有的事情。C 语言被誉为“无所不能的语言”,你所能想到的关于编程的事情,C 语言几乎都能干。

1.2.2 C 语言使用排行榜

TIOBE 编程语言排行榜是根据互联网上有经验的程序员、课程和第三方厂商的数量,并使用搜索引擎(如 Google、Bing、Yahoo!、百度)以及 Wikipedia、Amazon、You-