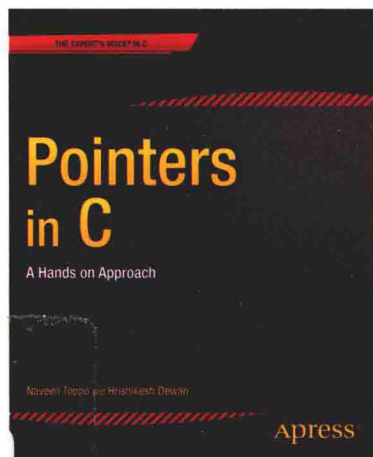


C 指针

基本概念、核心技术及
最佳实践



Pointers in C
A Hands on Approach

[印度]

纳威恩·拓奥普 (Naveen Toppo) 著
海瑞思科什·德万 (Hrishikesh Dewan)

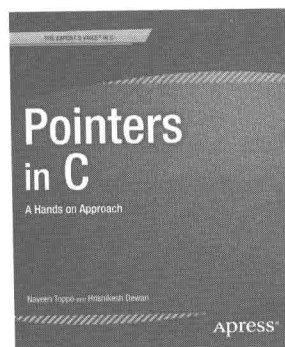
王贵财 译



机械工业出版社
China Machine Press

C 指针

基本概念、核心技术及
最佳实践



Pointers in C
A Hands on Approach

[印度] 纳威恩·拓奥普 (Naveen Toppo) 著
海瑞思科什·德万 (Hrishikesh Dewan)

王贵财 译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

C 指针: 基本概念、核心技术及最佳实践 / (印) 拓奥普 (Toppo, N.), (印) 德万 (Dewan, H.) 著; 王贵财译. —北京: 机械工业出版社, 2016.5

(C/C++ 技术丛书)

书名原文: Pointers in C: A Hands on Approach

ISBN 978-7-111-53818-9

I. C… II. ①拓… ②德… ③王… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2016) 第 104452 号

本书版权登记号: 图字: 01-2014-2246

Naveen Toppo, Hrishikesh Dewan: Pointers in C: A Hands on Approach.

Original English language edition published by Apress L.P., 2560 Ninth Street, Suite 219, Berkeley, CA 94710 USA. Copyright © 2013 by Apress L.P. Simplified Chinese-language edition copyright © 2016 by China Machine Press. All rights reserved.

This edition is licensed for distribution and sale in the People's Republic of China only, excluding Hong Kong, Taiwan and Macao and may not be distributed and sold elsewhere.

本书原版由 Apress 出版社出版。

本书简体字中文版由 Apress 出版社授权机械工业出版社独家出版。未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

此版本仅限在中华人民共和国境内 (不包括中国香港、台湾、澳门地区) 销售发行, 未经授权的本书出口将被视为违反版权法的行为。

C 指针: 基本概念、核心技术及最佳实践

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 陈佳媛

责任校对: 殷虹

印刷: 北京市荣盛彩色印刷有限公司

版次: 2016 年 6 月第 1 版第 1 次印刷

开本: 186mm×240mm 1/16

印张: 11.5

书号: ISBN 978-7-111-53818-9

定价: 49.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

C 是一门通用的编程语言，广泛用于系统软件与应用软件的开发。C 语言具有高效、灵活、功能丰富、表达力强和较高的可移植性等特点，在程序员中备受青睐，成为近几十年来使用最为广泛的编程语言。

指针是 C 语言最重要的特性，指针的运用被认为是最大的难关。关于指针的学习，我们经常听到这样的建议：“如果理解了计算机的内存和地址等概念，指针什么的就简单了。”果真如此吗？为了让读者了解指针概念，本书从介绍内存和实时内存概念开始，然后讲述了指针变量的概念和它最重要的操作（引用和解引用）、指针运算和一维数组、使用指针处理字符串、指针访问多维数组的用法、使用指针访问结构体与成员字段、函数指针概念的用法，以及文件 I/O 处理等知识。通过阅读本书，读者能快速掌握 C 指针的核心技术，本书特点如下：

- 通过分析大量实例，演示如何使用 C 指针。
- 由浅入深讲解实例，帮助读者逐步掌握 C 指针的用法。
- 图文并茂，让读者及时了解每一步的操作结果，帮助读者更好地检验学习进度。

夜深人静，万物安睡，世界不再喧哗，心最易平静。伴随淡淡的轻音乐文字在键盘敲打中跳跃于荧屏，一次次敲打键盘，一次次把想法寄托于文字。在这宁静与悠远的意境中翻译本书是一件美好的事情。翻译过程中常常为作者对 C 指针的深入理解和独到见解而惊讶、赞叹。对我自己而言，这也是一个学习与提高的过程，同时也为能向国内广大读者推荐这本好书而高兴。为做到专业词汇权威准确，内容忠实原书，我查阅了大量资

料。但受限于时间和精力，本书难免存在错误，恳请读者批评指正。

本书翻译过程中得到河南工业大学高层次人才基金（2012BS062）项目与河南工业大学优势专业培育工程项目（计算机科学与技术）资助。

特别感谢机械工业出版社的编辑为本书出版所付出的辛勤劳动。

感谢河南工业大学信息科学与工程学院靳小波博士对本书翻译的支持与鼓励，也特别感谢我的家人给予的包容和支持。

王贵财

2015年3月

About the Authors 作者简介



Naveen Toppo 目前担任西门子（印度）技术服务公司的 CT 直流技术部顾问。他拥有长达 7 年的工作经验，目前研究领域是硬件特性的最佳利用（性能工程）。

他还参与了研究分布式算法在空间数据库方面的应用。加入西门子之前，他是 Wipro Technologies 公司嵌入式系统技术主管，负责开发一个用于处理 L3 层的路由算法的 SoC 网络处理器。

此前他也参与了基于 Android 平台的英特尔平板电脑项目，以及 T9 词典的功能升级，语法与词汇分析等开发项目。他毕业于印度理工学院古瓦哈提校区计算机专业，并获得硕士学位。



Hrishikesh Dewan 自 2008 年 6 月一直担任西门子（印度）技术服务公司首席工程师。他毕业于位于班加罗尔的印度理工学院（IISC）计算机科学与自动化系，并获得博士学位，目前主要从事大规模分布式存储系统方面的研究。在西门子，他负责分布式系统实验室。加入 STS 之前，他创立了一个非常小的开放源码软件开发组织——Srishti，用来促进高等院校开源工具和软件的发展。他还在 Eth 公司（Dishnet 无线公司（Aircel）的一个附属的研发部门）担任两年多的项目经理。他曾出版过两本有关 WCF-SOAP 和 Visual Studio 技术的书。

前 言 *Preface*

自 1978 年 C 语言问世以来，它凭借强大的功能已成为最受程序员欢迎的编程语言。C 语言最初虽是为实现 UNIX 操作系统而设计的，可它广泛应用于非 UNIX 平台下的软件开发。C 语言一直是底层开发、设备驱动程序、嵌入式系统开发、移动设备开发等领域的首选语言。

指针是 C 语言最重要的特性，它非常有趣但不好理解。C 作为一门较低级的语言，要求程序员对许多计算机基本概念非常熟悉。C 也不是强类型语言。

指针概念的神秘性众所周知，这使得某些情况下很难理解指针的用法。本书力求使初学者、普通程序员、专家级程序员都能理解指针的概念。为让读者了解指针概念，我们引入背景信息，它无关语言本身，但也是计算机科学文献的一部分。该背景信息将帮助读者理解指针的概念。

本书的内容结构如下：

第 1 章是全书的基础。该章介绍内存和实时内存概念，帮助读者理解如何访问内存，以及数据 / 指令在内存中如何存储。该章帮助读者理解编译步骤，包括解释程序预处理、汇编和目标代码等中间结果如何生成。而且也给出编译器创建内存段的详细背景。通过详细解释内存段利弊，可以帮助读者理解各种变量的用法。另外该章还讨论了虚拟内存的概念。

第 2 章介绍指针变量的概念和它最重要的操作（引用和解引用）。该章介绍指针变量初始化、比较及内存分配、NULL 指针、悬摆指针、VOID 指针和 CONST 修饰符等概念。该章详细演示 integer、char 等基本数据类型指针的用法，也详细介绍如何用多重指针访问内存地址和存储这些位置

的值。

第3章详细解释指针运算和一维数组。该章详细介绍指针运算。演示使用指针加减运算如何访问各种连续内存位置。用一整节介绍用指针访问数组类型的用法。该章也介绍各种访问数组指定索引的方法。

第4章介绍如何使用指针初始化静态字符串并对其进行操作。给出了包括 `strcpy`、`substring` 等基本字符串处理函数实例。字符串操作对于解决与实现算法非常重要。

第5章介绍指针访问多维数组的用法，特别是二维数组和三维数组。

第6章详细描述怎样使用指针访问结构体与成员字段。使用结构体和指针便于实现复杂、动态的数据结构。该章也给出如何用指针处理链表和二叉树等数据结构的实例。该章用一整节内容专门介绍怎样用函数指针动态调用函数。

第7章解释函数指针概念的用法。

第8章介绍文件处理细节，并深入说明如何使用读写函数文件指针操作文件。

致谢 *Acknowledgements*

非常感谢我的父母 Clement、Xaveria，我的爱妻 Rashmi，我的兄弟 John 和 Kartik，他们给予我持续支持和无尽的耐心。感谢同事 Krishna 先生辛勤地输入代码。感谢所有朋友、同事，以及我就职的西门子公司。也感谢所有鸟友。最后，我要感谢 Saswata、Jeffrey、Anamika，以及技术编辑，感谢出版社所有帮助我将这个梦想变成现实的。

最后，我要感谢我的鸟友——Tickell's blue flycatcher 以及让我保持充沛精力来完成本书撰写的所有美丽的鸟儿。

——Naveen Toppo

非常感谢本书的版式设计人员。特别感谢 Saswata 和 Jeff 让我有机会撰写本书。

——Hrishikesh Dewan

Contents 目 录

译者序	
作者简介	
前 言	
致 谢	

第1章 内存、实时内存管理和虚拟内存	1
1.1 内存与类型	2
1.1.1 类型	2
1.1.2 内存排列	4
1.2 数据与指令	5
1.2.1 处理器如何访问主存	6
1.2.2 缓存	6
1.3 编译过程链	8
1.3.1 预处理	9
1.3.2 编译	10
1.3.3 汇编器	12
1.3.4 链接	12
1.3.5 加载器	13
1.4 内存模型	14

1.4.1	实地址内存模型	16
1.4.2	实地址模式下的地址转换	16
1.4.3	扁平内存模型	16
1.4.4	分段内存模型	17
1.5	使用分段的内存排列	18
1.5.1	分段	18
1.5.2	代码段	18
1.5.3	数据段	19
1.5.4	未初始化 /BSS 段	19
1.5.5	栈段	20
1.5.6	堆段	20
1.6	实时内存组织	21
1.6.1	函数调用的复杂性	21
1.6.2	函数调用步骤	22
1.7	内存段	24
1.8	虚拟内存组织	26
1.8.1	一窥虚拟内存系统	26
1.8.2	分页	29
1.8.3	页表	29
1.9	小结	30
第2章 指针基础		31
2.1	变量的地址	32
2.2	地址操作符	33
2.3	指针声明	34
2.4	指针赋值	35
2.5	指针变量大小	36
2.6	指针解引用	38
2.7	指针的基本用法	39

2.7.1 传值	40
2.7.2 引用传递	40
2.8 指针和常量	42
2.8.1 常量指针变量	42
2.8.2 常量指针	43
2.8.3 指针常量	44
2.9 多级指针	45
2.10 理解神秘的指针表达式	46
2.10.1 一级指针引用	47
2.10.2 一级指针解引用	47
2.10.3 二级指针引用	48
2.10.4 二级指针解引用	48
2.11 小结	50
第3章 指针运算与一维数组	51
3.1 数组内存排列	52
3.2 指针运算	55
3.2.1 指针加法	56
3.2.2 指针减法	58
3.3 数组探究	60
3.3.1 动态数组	62
3.3.2 指针数组	64
3.3.3 数组指针	66
3.4 小结	67
第4章 指针与字符串	69
4.1 内存中的字符串排列	70
4.2 动态内存分配	72
4.3 字符串操作	75

4.3.1 处理字符串输入	75
4.3.2 字符串遍历	75
4.3.3 字符串复制	77
4.3.4 字符串连接	77
4.4 字符串数组	78
4.5 小结	84
第5章 指针与多维数组	85
5.1 数组排列	86
5.1.1 二维数组指针	87
5.1.2 三维数组排列	94
5.2 小结	104
第6章 结构体指针	105
6.1 定义结构体	106
6.2 声明结构体变量	107
6.3 访问结构体成员	107
6.4 初始化结构体变量	108
6.4.1 方法1	108
6.4.2 方法2	108
6.5 结构体嵌套	108
6.6 结构体存储	109
6.7 结构体填充	111
6.7.1 数据对齐	111
6.7.2 字段填充	112
6.8 当结构体填充派不上用场时	115
6.9 结构体封装	118
6.10 结构体赋值与复制	120

6.11	结构体指针	122
6.11.1	访问成员变量	122
6.11.2	传递结构体指针变量	124
6.12	常见错误	125
6.13	结构体指针类型转换	126
6.14	自引用结构体	126
6.15	数据结构与算法	127
6.15.1	链表	127
6.15.2	二叉搜索树	129
6.15.3	遍历节点	131
6.16	小结	131
第 7 章	函数指针	133
7.1	定义函数指针	134
7.2	初始化函数指针	135
7.3	使用函数指针	135
7.4	函数指针调用的汇编细节	137
7.4.1	直接调用函数	138
7.4.2	利用函数指针间接调用函数	138
7.5	函数指针数组	139
7.6	从函数中返回函数指针	140
7.6.1	复杂方法	140
7.6.2	简单方法	141
7.7	Linux 内核里的函数指针用法	142
7.8	神秘的函数指针声明总结	144
7.9	小结	145
第 8 章	文件 I/O 指针	147
8.1	辅存的物理排列	148

8.1.1	盘面图	148
8.1.2	硬盘驱动器与 CPU 的接口	149
8.1.3	硬盘寻址方案	150
8.2	文件系统结构简介	150
8.2.1	逻辑文件系统	150
8.2.2	基本文件系统	152
8.3	创建一个文件系统需要什么	152
8.3.1	硬盘数据结构体	152
8.3.2	内存数据结构体	152
8.4	访问文件	153
8.5	首要任务	154
8.6	次要任务	155
8.7	从文件读取	156
8.7.1	从文件中读取单个字符	156
8.7.2	从文件中读取字符串	158
8.7.3	从文件中读取数据块	159
8.7.4	从文件中格式化读取	160
8.8	写入文件	162
8.8.1	向文件写入单个字符	162
8.8.2	向文件写入字符串	163
8.8.3	向文件写入数据块	164
8.9	在任意位置访问硬盘	165
8.10	其他函数	167
8.10.1	确定文件大小	167
8.10.2	重置文件位置的另一种方法	169
8.11	小结	170



第 1 章 *Chapter 1*

内存、实时内存管理和 虚拟内存

- 1.1 内存与类型
 - 1.2 数据与指令
 - 1.3 编译过程链
 - 1.4 内存模型
 - 1.5 使用分段的内存排列
 - 1.6 实时内存组织
 - 1.7 内存段
 - 1.8 虚拟内存组织
 - 1.9 小结
- 

我一直想知道为什么指针的概念如此让人难以把握。如果你能在脑海中将其“可视化”，将会有助于你直观理解指针的概念。我说的“可视化”是指能在心里表现其存储、生命周期、值等。可是在深入介绍指针细节前，读者需要先了解内存概念、程序实时内存管理、虚拟内存、执行模型与部分汇编语言知识。

本章通过如何建立运行组织模型常规实例和几个关于 CPU 如何访问进程中不同内存段的简单实例介绍这些前提概念。最后介绍了虚拟内存概念。

后续章节将介绍指针基础知识、用法、指针操作的高级主题，以及内存取址与取值操作算法。最后一章着重于指针的实际应用。

本章内容相互之间关联性不是很强，读者可跳读熟悉章节。

1.1 内存与类型

按定义，内存用于存储指令和数据序列。内存分为永久和临时存储两种类型。除非明确指出永久存储，否则本书中内存都理解为临时 / 非永久存储（如 RAM、cache、寄存器等）。内存是一组以二进制方式存储信息的单元，存储容量取决于底层硬件或体系结构以及位长（1、2、4、8、16、32、64 或 128 位）。

1.1.1 类型

内存类型是衡量和评价各类内存的最佳方法（见图 1-1）。