



# 指针的艺术

程序设计最绚丽的星星

The Art of Pointer: Beautiful Star

蔡明志 著



赠1CD



中国水利水电出版社  
www.waterpub.com.cn

# 指针的艺术

蔡明志 著



中国水利水电出版社  
[www.waterpub.com.cn](http://www.waterpub.com.cn)

## 内 容 提 要

本书主要探讨 C 语言指针的基本概念及其与变量、数组、结构体和文件之间的应用,同时剖析 C++、C++/CLR、Java、Visual C#及 Visual Basic 中的引用 (Reference),并分别以指针和引用 (Reference) 实现链表、堆栈与队列。

全书共七篇。针对以下 14 个主题,分别在 C、C++、C++/CLR、Java、Visual C#及 Visual Basic 程序语言中比较其用法:变量的定义、算术运算符、关系运算符、逻辑运算符、输出函数、输入函数、选择结构、循环结构、数组、函数调用、指针与引用 (Reference)、指向或引用 (Reference) 到空的表示法、动态分配内存、回收内存。

本书适合 C、C++、C++/CLR、Java、Visual C#及 Visual Basic 程序设计人员使用,希望通过本书的内容,能够帮助读者透彻理解指针的概念并在实践中能够灵活运用。书中所有范例代码均可在配书光盘中找到。

本书为经台湾碁峰资讯股份有限公司独家授权发行的中文简体版。本书中文简体字版在中国大陆之专有出版权属中国水利水电出版社所有。在没有得到本书原版出版者和本书出版者书面许可时,任何单位和个人不得擅自摘抄、复制本书的一部分或全部以任何方式(包括资料和出版物)进行传播。本书原版权权属碁峰资讯股份有限公司。版权所有,侵权必究。

北京市版权局著作权合同登记号:图字 01-2009-5448 号

### 图书在版编目 (C I P) 数据

指针的艺术 / 蔡明志著. — 北京:中国水利水电出版社, 2009. 10  
ISBN 978-7-5084-6897-6

I. ①指… II. ①蔡… III. ①C语言—程序设计 IV. ①TP312

中国版本图书馆CIP数据核字(2009)第190391号

策划编辑:周春元 责任编辑:杨庆川 封面设计:李佳

书 名	指针的艺术
作 者	蔡明志 著
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 68367658 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	184mm×260mm 16 开本 31.5 印张 593 千字
版 次	2009 年 10 月第 1 版 2009 年 10 月第 1 次印刷
印 数	0001—3000 册
定 价	59.80 元 (赠 1CD)

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换  
版权所有·侵权必究

## 序言

C 语言的特色就是“指针”(Pointer)，让人又爱又恨，让熟悉它的人，将它比作是天上那一颗最美的星，而让不懂的人，头上冒星星，同样是星星，但却有不同的情境。

“指针好难，都不知道该怎么学”，常常会有人对我讲这句话，并问我有没有好方法可以“领悟”它。说实在的，真是惭愧，从 1983 年进入信息管理领域，因为工作关系，接触 C，接着在 1985 年到东吴大学计算器科学系（现在改为信息管理系）兼课，在 1987 年将上课的讲义和心得，出版了我的第一本书《C 程序语言及其应用》，算一算到现在整整有二十几年的时间了。听到指针是如此难的问题，让我心中很难过。拥护我的读者也不在少数，有些人常会跟我说，“C 先生，我是看您的书长大的”。真诚地希望你们可以从书中获取一些知识。再一次的谢谢你们，一路的相陪、支持与鼓励，让我有动力再出发。

责任在我，是我的错，我没尽力将它说明白，讲清楚，使得大家有如此彷徨，裹足不前的问题。基于此，我在二、三年前就开始规划，打算编写一本以指针为中心的书，让对指针不太懂的人，可以轻易了解指针的运作原理，让懂的人可以精益求精。现在，已梦想成真，从书名《指针的艺术》，就可以知道它是您学习与深入了解指针最佳的读本。

本书内容精彩无比，除了对 C 语言的指针有深入的探讨外，同时也将与指针有异曲同工之妙的引用 (reference) 详加讨论，所以本书不仅涉及 C 而已，还包括 C++、C++/CLR、Java、Visual C# 及 Visual Basic，有关引用的主题。其中要和大家分享心得的是，原本对 Visual Basic 没什么好感，但此次接触它，发现也不错，若您的观念和我先前一样，不妨试试看，相信您会喜欢的。

本书是以指针为核心，以引用 (reference) 为辅。在 C 程序语言篇中，从内存的基本概念开始展开，其中依次涉及，指针与变量、指针与数组、指针与函数、指针与字符串、指针与结构体、链表、指针与文件等，希望有效引导读者进入指针的世界。为了验证指针与引用的相似功能，在之后的程序语言篇着重于如何以引用实现指针可以完成的事项，举例以引用调用完成两数对调，传递数组到某一函数及链表、堆栈等有关数据结构问题的实现。

本书的第七篇是比较篇，将 C、C++、C++/CLR、Java、Visual C# 及 Visual Basic 六种程序语言在程序语言基本架构上的主题做一比较，希望此篇可以让您一窥程序语言之美。

写本书的心情不知为什么，总是觉得很愉快，因为我已做到了可以向读者交待的事，那就是您可以从本书中轻轻松松地了解什么是指针，并告诉我，指针是天上那一颗最闪亮的星星，祝福您。



(mjtsai168@gmail.com)



# 目 录

## 第一篇 C 程序语言篇

第 1 章 内存的基本概念	3
1-1 变量	4
1-2 一维数组	5
1-3 二维数组	9
1-4 结构体	11
1-5 问题演练	13
第 2 章 指针与变量	15
2-1 一颗星星：表示一把钥匙	16
2-2 两颗星星：表示两把钥匙	19
2-3 三颗星星：表示三把钥匙	21
2-4 问题演练	23
第 3 章 指针与数组	25
3-1 指针与一维数组	26
3-2 指针与二维数组	32
3-3 数组指针	35
3-4 为什么 <code>parr</code> 等同于 <code>*parr</code>	36
3-5 指向数组的指针	38
3-6 多重指针	44
3-7 命令行自变量	51
3-8 调试题	54
3-9 问题演练	55
3-10 程序实战	62
第 4 章 指针与函数	63
4-1 函数初探	64
4-2 两数对调	64
4-3 再论传址调用	66
4-3-1 寻找数组中的最大值	66
4-3-2 一维数组的元素求和	67

4-3-3 二维数组的元素求和	69
4-4 指向函数的指针	70
4-5 返回指针的函数	72
4-6 调试题	74
4-7 问题演练	77
4-8 程序实战	77
<b>第 5 章 指针与字符串</b>	<b>79</b>
5-1 字符串指针	80
5-2 常用的库函数	83
5-2-1 计算字符串的长度	83
5-2-2 字符串的复制	84
5-2-3 字符串的连接	87
5-2-4 字符串的比较	88
5-3 调试题	93
5-4 程序实战	94
<b>第 6 章 指针与结构体</b>	<b>95</b>
6-1 指向结构体变量的指针	96
6-2 结构体指针与自增运算符	100
6-3 结构体与函数	102
6-4 自引用	105
6-5 调试题	111
6-6 问题演练	119
<b>第 7 章 链表——以 C 程序实现</b>	<b>121</b>
7-1 单向链表	122
7-2 双向链表	126
7-3 链表的应用——堆栈的实现	130
7-4 链表的应用——队列的实现	134
7-5 程序实战	138
<b>第 8 章 指针与文件</b>	<b>139</b>
8-1 fprintf 与 fscanf 函数	140
8-2 fwrite 与 fread 函数	143
8-3 当前文件的指针指向何处	145
8-4 fgetpos 与 fsetpos 函数	147
8-5 rewind	149
8-6 程序实战	150

## 第二篇 C++程序语言篇

第 9 章 简易 Visual C++ 2008 使用手册 .....	153
第 10 章 C++的 reference .....	157
10-1 变量与 reference .....	158
10-2 两数对调 .....	159
10-2-1 传值调用 .....	159
10-2-2 引用调用 (call by reference) .....	160
10-3 结构体与 reference .....	163
10-4 类与 reference .....	166
10-5 问题演练 .....	171
10-6 程序实战 .....	172
第 11 章 C++的 this 指针 .....	173
11-1 this 指针 .....	174
11-2 股票买卖范例程序 .....	177
11-3 程序实战 .....	180
第 12 章 C++的 new 与 delete .....	181
12-1 new、delete 与变量 .....	182
12-2 new、delete 与数组 .....	184
12-3 new、delete 与结构体 .....	186
12-4 new、delete 与类 .....	187
12-5 程序实战 .....	190
第 13 章 C++的复制构造函数 .....	191
13-1 复制构造函数 .....	192
13-2 重载赋值运算符 .....	196
13-3 程序实战 .....	200
第 14 章 虚函数、指针与 reference .....	201
14-1 虚函数 .....	202
14-2 程序实战 .....	205
第 15 章 链表——以 C++程序实现 .....	207
15-1 单向链表 .....	208
15-2 双向链表 .....	213
15-3 堆栈 .....	218
15-4 程序实战 .....	220



**第三篇 C++/CLR 程序语言篇**

第 16 章 简易 Visual C++/CLR 使用手册 .....	223
第 17 章 浅谈 C++/CLR 程序设计 .....	227
17-1 可跟踪的 reference .....	228
17-2 可跟踪的 handle .....	228
17-3 数组 .....	229
17-3-1 一维数组 .....	229
17-3-2 二维数组 .....	231
17-3-3 不规则数组 .....	232
17-4 Interior 指针 .....	233
17-5 函数 .....	233
17-5-1 两数对调 .....	233
17-5-2 找出数组的最大值 .....	236
17-5-3 计算数组的总和 .....	237
17-5-4 计算二维数组的总和 .....	238
17-5-5 计算不规则数组的总和 .....	239
17-6 程序实战 .....	240
第 18 章 链表——以 Visual C++/CLR 实现 .....	241
18-1 以单向链表处理数据 .....	242
18-2 以双向链表处理数据 .....	245
18-3 使用链表实现堆栈 .....	248
18-4 程序实战 .....	250

**第四篇 Java 程序语言篇**

第 19 章 简易 JCreator 使用手册 .....	253
19-1 安装 JCreator 编译程序 .....	254
19-2 编写、编译与运行程序 .....	256
19-3 程序的调试 (debug) .....	257
19-4 创建项目文件 .....	258
第 20 章 变量、字符串与 reference .....	259
20-1 变量与引用 (reference) .....	260
20-2 字符串与引用 (reference) .....	261
20-3 常用的字符串库函数 .....	264
20-4 程序实战 .....	267

第 21 章 数组与 reference .....	269
21-1 数组初探 .....	270
21-2 数组的复制 .....	275
21-3 如何传送数组 .....	278
21-4 二维数组 .....	283
21-5 程序实战 .....	286
第 22 章 类与 reference .....	287
22-1 类与引用 (reference) .....	288
22-2 复制构造函数 .....	291
22-3 this 指针 .....	297
22-4 程序实战 .....	298
第 23 章 链表——以 Java 程序实现 .....	299
23-1 单向链表 .....	300
23-2 双向链表 .....	304
23-3 堆栈的操作 .....	310
23-4 程序实战 .....	314
<b>第五篇 Visual C# 2008 程序语言篇</b>	
第 24 章 简易 Visual C# 使用手册 .....	317
第 25 章 变量与 reference .....	321
25-1 传值调用 .....	322
25-2 引用调用 (call by reference) .....	323
25-3 特殊的参数 out .....	324
25-4 程序实战 .....	326
第 26 章 数组与 reference .....	327
26-1 找出一维数组的最大值 .....	328
26-2 一维数组的求和 .....	329
26-3 二维数组的求和 .....	330
26-4 不规则数组的求和 .....	331
26-5 程序实战 .....	332
第 27 章 链表——以 Visual C# 实现 .....	333
27-1 数据的转换 .....	334
27-2 单向链表 .....	334
27-3 双向链表 .....	338
27-4 链表的应用——堆栈的实现 .....	342

27-5 程序实战	346
<b>第六篇 Visual Basic 2008 程序语言篇</b>	
第 28 章 简易 Visual Basic 使用手册	349
第 29 章 函数与 reference	353
29-1 传值调用 (call by value)	354
29-2 引用调用 (call by reference)	354
29-3 程序实战	355
第 30 章 数组与 reference	357
30-1 找出数组的最大值	358
30-2 数组的求和	360
30-2-1 一维数组的求和	360
30-2-2 二维数组的求和	361
30-3 不规则数组	362
30-4 程序实战	364
第 31 章 链表——以 Visual Basic 实现	365
31-1 链表的基本概念	366
31-2 单向链表	368
31-3 双向链表	372
31-4 链表的应用——堆栈的实现	376
31-5 程序实战	380
<b>第七篇 比较篇</b>	
附录 A 解 答	393

# PART

## 1

## C 程序语言篇

C 最早是 AT&T (American Telephone and Telegraph, 美国电话与电报公司) 内部使用的语言, 但为了公布 UNIX 操作系统, 不得不公开此程序语言。主导 C 语言的两个关键人物, 分别是 Dennis Ritchie 与 Brian Kernighan。Dennis Ritchie 生于 1941 年, 哈佛大学数学博士, 可说是 C 程序语言之父。Brian Kernighan 目前服务于普林斯顿大学计算机系 (Department of Computer Science, Princeton University)。ANSI C 已于 1989 公布。

C 程序语言 (1971 年) 是受到 CPL (1962 年)、BCPL (在 1969 年由 Martin Richards 编写)、B (在 1970 年由 Ken Thompson 编写), 这些程序语言的直接或间接的影响和演化所产生的。

图 1 是这两位大师所编写的 C 程序语言课本《The C Programming Language》。我们要给予这两位大师热烈的掌声。想要了解更多的读者, 可访问他们所属的网站:

Dennis Ritchie: <http://plan9.bell-labs.com/cm/cs/who/dmr/>

Brian Kernighan: <http://www.cs.princeton.edu/~bkw/>

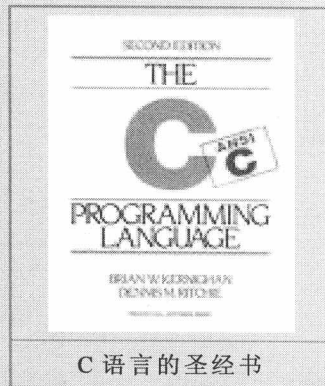


图 1

C 语言最大的特色是——拥有非常强大的指针（pointer）。这是很多学习者又爱又恨的主题。很多人学到这个主题时，头上就开始冒汗，因而对程序语言不感兴趣。但有些人却似它为天上闪亮的星星，从而爱上程序设计。我不晓得您是哪一种情况，希望您是后者。

若您是前者，没关系，让这本书为您揭开指针的神秘面纱。本篇规划了与指针相关的主题，例如：指针与变量、指针与数组、指针与函数、指针与字符串、指针与结构、指针与文件等，由浅入深，并配合图形加以解说，希望您能了解指针的精髓，从而将指针看似天上最灿烂的星星，这也是笔者写这本书的初衷。

同时也要告诉读者，C 除了有强大的指针之外，它也是 UNIX 与 Linux 操作系统的核心语言，也就是说，这两个操作系统是以 C 编写而成的。所以有人说，C 语言是系统程序语言（system programming language）。不仅如此，它也是编写有关网络、通信及嵌入式程序的最佳程序语言！

总而言之，好好学习 C 程序语言，对您的未来一定会有加分的效果。

# 内存的基本概念

- 1-1 变量
- 1-2 一维数组
- 1-3 二维数组
- 1-4 结构体
- 1-5 问题演练

当程序定义一个变量时，编译程序依据其数据类型分配适当大小的内存给它，以便存放数据。这有如地址一般，邮递员根据收件人的地址，将信件投入正确的信箱。

## 1-1 变量

程序以变量名称代表某一事物，好比每一个人都有姓名一般。若有人喊“蔡明志”，那我一定会说“有”，因为这个名字就代表我。下面就从变量谈起。请参阅范例 address-3。

### 范例 address-3

```
/* address-3.c */
#include <stdio.h>
#include <conio.h>
int main()
{
    int x=100, y=200, z=300;
    printf("x 变量的地址是 %p\n", &x);
    printf("y 变量的地址是 %p\n", &y);
    printf("z 变量的地址是 %p\n\n", &z);

    printf("x 变量的内容是 %d\n", x);
    printf("y 变量的内容是 %d\n", y);
    printf("z 变量的内容是 %d\n", z);
    getch();
    return 0;
}
```

### 输出结果

```
x 变量的地址是 0022FF74
y 变量的地址是 0022FF70
z 变量的地址是 0022FF6C

x 变量的内容是 100
y 变量的内容是 200
z 变量的内容是 300
```

程序中有关 x、y、z 变量的地址，如图 1-1 所示。

0022FF68	0022FF6C	0022FF70	0022FF74	0022FF78
	300	200	100	

图 1-1

若想获知变量所在内存的地址，只要在变量前加上地址运算符（&）即可。我们一定要清楚每一表达式（expression）所表达的意思，是变量所在的内存地址（address），还是变量值（value）。

若要利用 scanf 函数从键盘输入数据给一变量需注意哪些。我们以范例 address-5 来说明。

### 范例 address-5

```
/* address-5.c */
#include <stdio.h>
#include <stdlib.h>

int main()
{
    double x, y, average;
    printf("请输入浮点数 x 与 y: ");
    scanf("%lf %lf", &x, &y);
    average = (x+y)/2;
    printf("x 与 y 的平均数为: %.2f\n", average);
    system("PAUSE");
    return 0;
}
```

### 输出结果

```
请输入浮点数 x 与 y: 12.34 45.67
x 与 y 的平均数为: 29.01
```

利用 scanf 函数，从键盘输入数据给变量。要特别注意的是，scanf 函数的变量要给予变量的地址，因此，会用到地址运算符（&）。这好比邮递员一定要根据收件人的地址，才能将信放到收件人的信箱。程序中变量 x 与 y 的数据类型是双精度（double），所以其对应的格式均是 %lf。

## 1-2 一维数组

我们把焦点转到一维数组，范例 addressArray-5 示范如何调用 scanf 函数，从键盘输入数据给数组。

### 范例 addressArray-5

```
/* addressArray-5.c */
#include <stdio.h>
#include <conio.h>
int main()
{
```



```

int x[5];
int i;

for(i=0; i<5; i++)
{
    printf("请输入 x[%d]的整数值: ", i);
    scanf("%d", &x[i]);
}
for(i=0; i<5; i++)
    printf("x[%d] = %d\n", i, x[i]);
printf("\n");

for(i=0; i<5; i++)
{
    printf("请输入 x[%d]的整数值: ", i);
    scanf("%d", x+i);
}
for(i=0; i<5; i++)
    printf("x[%d] = %d\n", i, x[i]);
printf("\n");
getch();
return 0;
}

```

输出结果如图 1-2 所示。

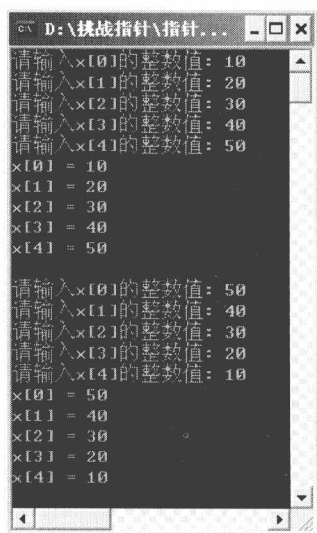


图 1-2

此数组有 5 个元素，分别为  $x[0]$ 、 $x[1]$ 、 $x[2]$ 、 $x[3]$ 、 $x[4]$ 。每一元素均为整型变量，所以只要在这些变量前加上  $\&$ ，即表示该变量在内存的地址。程序利用一个循环输入数据给  $x$  数组，如下所示：