

程序员开发宝典系列

C语言 开发宝典



The Bible for C Developer

从小工到专家
程序员修炼宝典

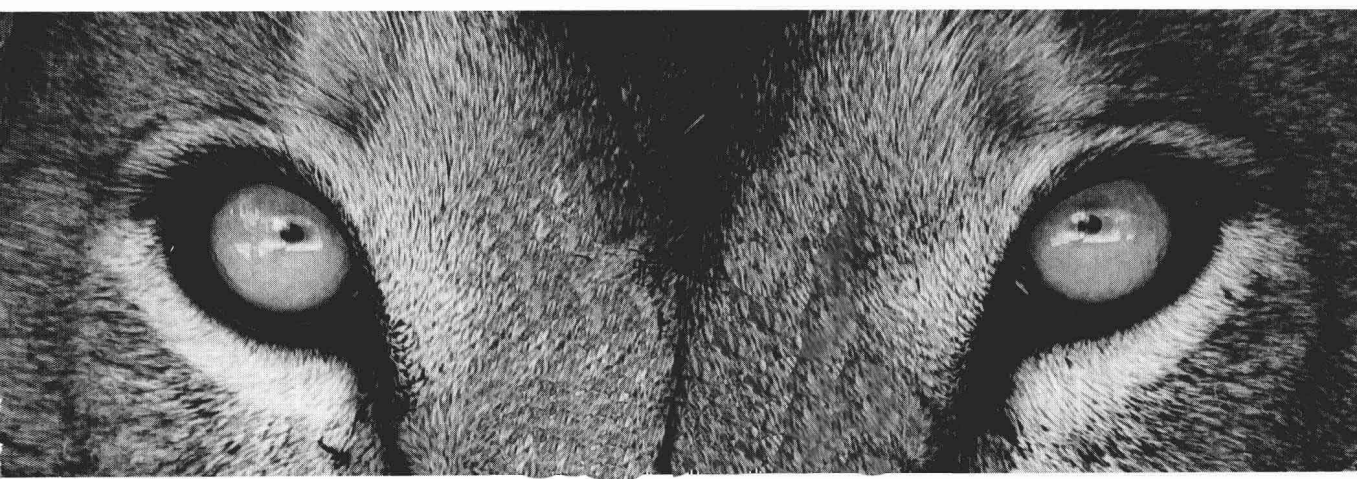
8.5小时全程多媒体视频讲解
零基础入门训练 典型模块训练
项目实战训练 能力测试系统
本书PPT课件 本书源程序



曹飞飞 高春艳 等编著

程序员开发宝典系列

C语言 开发宝典



The Bible for C Developer

曹飞飞 高春艳 等编著



机械工业出版社
China Machine Press

本书全面细致地讲解了使用C语言进行编程和实际项目开发的各种技术,是学习C语言编程的必备图书。全书以Visual C++ 6.0开发环境和Turbo C 2.0开发环境为基础,讲解了C语言编程中的各种技术。全书分两大部分共22章,其中,第一部分为开发技术,主要内容包括如何学好C语言、初识C语言、C语言的战前准备、C语言与算法、C语言基础、表达式与运算符、输入/输出函数、选择结构程序设计、循环结构程序设计、数组、函数、指针、结构体和共用体、位运算、存储管理、预处理、文件操作技术、图形图像等内容;第二部分为项目实战,主要包括五子棋游戏开发、学生管理系统、商品信息管理系统、图书管理系统等4个大型项目。

本书所配DVD光盘含有全程语音视频讲解教学录像、PPT课件以及实例的全部源代码,这些源代码都经过精心调试,在Windows XP、Windows 2000、Windows 2003和Windows 7下均全部通过测试。

本书内容详尽,实例丰富,非常适合作为零基础学习人员、有志于从事软件开发的初学者、高校计算机相关专业学生的学习用书,也适合作为相关培训机构的师生和软件开发人员的参考资料。

封底无防伪标均为盗版

版权所有,侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目(CIP)数据

C语言开发宝典 / 曹飞飞, 高春艳, 等编著. —北京: 机械工业出版社, 2012.5
(程序员开发宝典系列)

ISBN 978-7-111-37786-3

I. C… II. ①曹… ②高… III. C语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2012)第050417号

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: 秦 健

北京京师印务有限公司印刷

2012年6月第1版第1次印刷

185mm×260 mm • 28.25印张

标准书号: ISBN 978-7-111-37786-3

ISBN 978-7-89433-386-5 (光盘)

定价: 59.00元(附光盘)

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) 88378991; 88361066

购书热线: (010) 68326294; 88379649; 68995259

投稿热线: (010) 88379604

读者信箱: hzjsj@hzbook.com

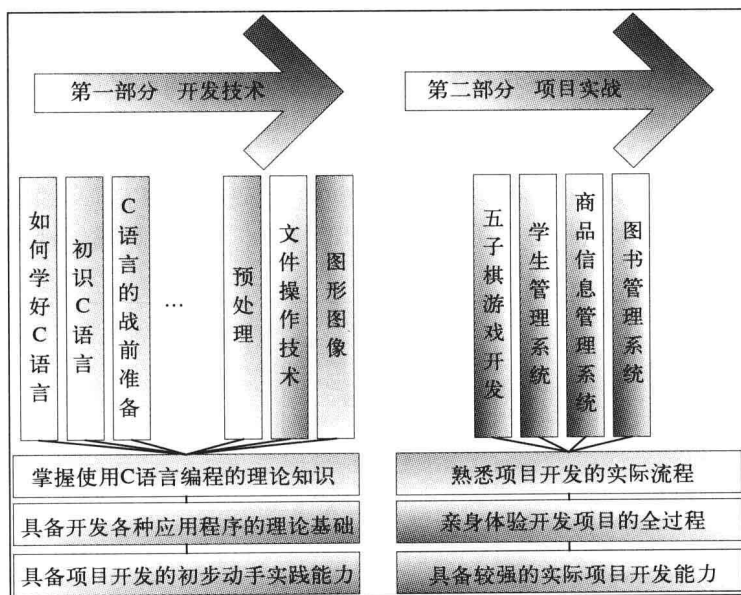
欢迎使用“程序员开发宝典系列”之《C语言开发宝典》！希望本书能成为您书架上的一部经典图书！

本书写作目的

C语言是由UNIX的研制者丹尼斯·里奇（Dennis Ritchie）和肯·汤普逊（Ken Thompson）于1970年在研制出的BCPL语言（B语言）的基础上发展和完善起来的，从其诞生以来就受到广大程序开发人员的追捧。市面上介绍C语言的书籍非常多，但真正能让读者零基础入门，并能开发出项目的书籍少之又少。为此，我们编写了本书，真正让读者能够零基础学习C语言编程，并开发出自己的项目，使本书真正成为每一位C语言编程人员的“入门第一书”。

本书主要内容

本书提供了C语言编程从零基础入门到实际项目开发所必备的所有知识，并分两大部分进行讲解，每一部分内容都是笔者经过走访各个大学、书店和培训机构，听取老师、学生和读者的建议，并结合自己多年的开发经验，精心提炼出来的。为了使读者更好地学习本书，下面给出了本书的内容结构图。



第一部分 开发技术（第0章至第17章）：主要内容包括如何学好C语言、初识C语言、C语言的战前准备、C语言与算法、C语言基础、表达式与运算符、输入/输出函数、选择结构程序设计、循环结构程序设计、数组、函数、指针、结构体和共用体、位运算、存储管理、预处理、文件操作技术、图形图像等内容。通过本篇的学习，读者能够熟练掌握使用C语言编程的理论

知识，并能具备开发各种应用程序的理论基础和初步的动手实践能力。

第二部分 项目实战（第18章至第21章）：主要内容包括五子棋游戏开发、学生管理系统、商品信息管理系统、图书管理系统4个大型项目，每个项目都运用软件工程的设计思想，让读者学习如何进行软件项目的实践开发，并按照从前期策划、设计流程到项目最终实现的整个过程进行介绍，带领读者逐步体验开发项目的全过程，使读者具备较强的实际项目开发能力。

本书特点

□ 零起点编程

初学者学习编程最大的瓶颈是基础知识薄弱，本书为了保证零基础读者能完全入门，除了讲解细致，实例丰富外，还特别提供了“入门训练营”专题模块，该模块列出了零基础读者入门必须掌握的三大部分基础专项训练，以便打好基础，建立学习的兴趣。

- 编程基础入门训练：讲解和训练开发中经常用到，但在书本中无法融入系统知识体系的知识。
- 逻辑思维训练：提供了编程者必须掌握的逻辑思维知识，建立程序开发思维，为后续编程打下坚实基础。
- 英语基本功训练：对于一些基础知识薄弱的读者，本部分提供了编写代码时经常用到的英语词汇，并进行专项讲解和训练。经过专项训练后，读者编写代码可得心应手。

□ 学练结合，适合自学

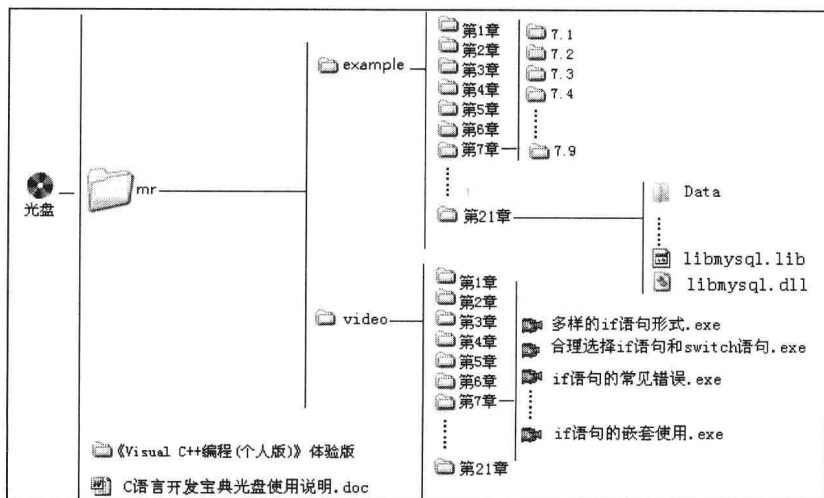
由于本书定位为“入门第一书”，因此在讲解知识时，提供了丰富的实例。为了保证学完每个实例后能学有所用，大部分实例后面都提供了“试一试”的练习题目，帮助读者进行随学随练，真正入门。

□ 全面系统，贴近实际

本书为了提升编程人员的实际开发水平，每个二级标题后面都提供了“上机练习”栏目，每个栏目提供多个实习题目，巩固所学。

□ 注释详尽，视频讲解

为了便于读者更好地学习和使用本书，书中所有的代码都提供了详尽的注释；而且本书附带的DVD光盘提供了重点知识的语音视频讲解及PPT课件，读者可以通过视频快速、直观、轻松地学习。



本书读者对象

- ☐ 零基础学习人员
- ☐ 相关培训机构的老师和学员
- ☐ 面临就业的学生
- ☐ 准备从事软件开发的求职者
- ☐ 大中专院校师生
- ☐ 编程爱好者
- ☐ 初、中级程序开发人员
- ☐ 立志编程的其他专业人士

技术支持与服务

如果您在学习或使用本书的过程中遇到问题或疑惑，可以通过以下方式与我们联系：

- ☐ 服务网站：www.mingribook.com
- ☐ 服务电话：0431-84978981/84978982
- ☐ 企业QQ：4006751066
- ☐ 学习社区：www.mrbccd.com
- ☐ 服务信箱：mingrisoft@mingrisoft.com

我们承诺将在1至5个工作日内给您提供解答。

关于作者

本书由明日科技C语言程序开发团队策划并组织编写，主要编写人员有曹飞飞、高春艳、赛奎春、吴绪铎、朱晓、赵永发、高文才、杨丽、王国辉、陈丹丹、潘凯华、刘欣、李慧、陈英、赵会东、王小科等，在编写本书的过程中，我们以科学、严谨的态度，力求精益求精，但不足和疏漏之处在所难免，敬请广大读者批评指正。

最后，感谢您选择本书，希望本书能成为您学习C语言编程路上的领航者。

编 者

2012年5月

目 录

前言

第一部分 开发技术

第0章 如何学好C语言2

第1章 初识C语言4

视频讲解：14分钟

1.1 C语言介绍4

1.1.1 程序语言简述4

1.1.2 C语言的发展史5

1.2 选择C语言的理由5

1.3 一个简单的C程序6

1.4 术语7

1.5 小结8

1.6 练习8

第2章 C语言的战前准备9

视频讲解：18分钟

2.1 程序的基本编写格式9

2.2 C语言常用开发工具简介10

2.2.1 Turbo C 2.010

2.2.2 Visual C++ 6.014

2.2.3 Dev-C++18

2.2.4 Eclipse for C/C++19

2.3 术语19

2.4 小结19

2.5 练习19

第3章 C语言与算法20

视频讲解：13分钟

3.1 算法的基本概念20

3.1.1 什么是算法20

3.1.2 算法的特性20

3.1.3 算法的好坏21

3.2 算法描述21

3.2.1 自然语言22

3.2.2 流程图22

3.2.3 3种基本结构23

3.2.4 N-S流程图25

3.2.5 伪代码28

3.3 术语29

3.4 小结29

3.5 练习29

第4章 C语言基础30

视频讲解：47分钟

4.1 C语言基础编程规范30

4.1.1 合理使用注释30

4.1.2 代码的缩进31

4.1.3 常量和变量的命名规范31

4.1.4 函数的命名规范31

4.1.5 程序中对应的{}要对齐31

4.1.6 合理使用空格使代码更规范32

4.1.7 换行使代码更清晰32

4.2 语言要素33

4.2.1 关键字33

4.2.2 标识符33

4.3 数据类型35

4.4 解读常量36

4.4.1 整型常量36

4.4.2 实型常量38

4.4.3 字符型常量38

4.4.4 符号常量41

4.5 剖析变量42

4.5.1 整型变量42

4.5.2 实型变量44

4.5.3 字符型变量47

4.6 分解枚举49

4.7 使用typedef定义类型50

4.8 变量的存储类型52

4.8.1 动态存储与静态存储52

4.8.2 auto变量53

4.8.3 static变量53

4.8.4 register变量54

4.8.5 extern变量55

4.9 混合运算	56	6.3.2 字符串输入函数	78
4.10 术语	57	6.4 格式输出函数	79
4.11 小结	57	6.5 格式输入函数	82
4.12 练习	57	6.6 术语	85
第5章 表达式与运算符	58	6.7 小结	85
视频讲解：34分钟		6.8 练习	86
5.1 赋值运算符与赋值表达式	58	第7章 选择结构程序设计	87
5.1.1 变量赋初值	58	视频讲解：35分钟	
5.1.2 自动类型转换	60	7.1 了解if语句	87
5.1.3 强制类型转换	60	7.2 多样的if语句形式	87
5.2 算术运算符与算术表达式	61	7.2.1 if语句形式	87
5.2.1 算术运算符	61	7.2.2 if else语句形式	90
5.2.2 算术表达式	62	7.2.3 else if语句形式	91
5.2.3 优先级与结合性	63	7.3 if语句的嵌套使用	94
5.2.4 自增自减运算符	64	7.4 if语句的常见错误	96
5.3 关系运算符与关系表达式	66	7.4.1 使用分号的注意事项	96
5.3.1 关系运算符	66	7.4.2 条件放置注意事项	97
5.3.2 关系表达式	66	7.4.3 if和else配对的注意事项	97
5.3.3 优先级与结合性	67	7.4.4 大括号的重要性	97
5.4 逻辑运算符与逻辑表达式	68	7.5 判断分支选择结构的switch语句	98
5.4.1 逻辑运算符	69	7.5.1 switch语句的基本形式	98
5.4.2 逻辑表达式	69	7.5.2 多路开关模式的switch语句	100
5.4.3 优先级与结合性	69	7.6 合理选择if语句和switch语句	102
5.5 位逻辑运算符与位逻辑表达式	70	7.7 选择结构的应用	104
5.5.1 位逻辑运算符	70	7.8 术语	106
5.5.2 位逻辑表达式	71	7.9 小结	106
5.6 逗号运算符与逗号表达式	71	7.10 练习	107
5.7 复合赋值运算符	72	第8章 循环结构程序设计	108
5.8 判断条件运算符	73	视频讲解：46分钟	
5.9 运算符的优先级	74	8.1 循环语句的概念	108
5.10 术语	74	8.2 实现while循环	108
5.11 小结	74	8.2.1 while语句的一般形式	108
5.12 练习	74	8.2.2 while语句的死循环	110
第6章 输入/输出函数	75	8.2.3 while语句的空循环体	110
视频讲解：21分钟		8.2.4 while循环语句的常见错误	111
6.1 解读语句	75	8.3 实现for循环	112
6.2 字符数据的输入/输出	75	8.3.1 for循环语句的一般形式	113
6.2.1 字符数据输出	75	8.3.2 for循环的灵活多变	114
6.2.2 字符数据输入	76	8.3.3 for语句中逗号的使用	117
6.3 字符串的输入/输出	77	8.4 总要执行的do while语句	118
6.3.1 字符串输出函数	78	8.5 比较3种循环	120

8.6 运用循环嵌套	120
8.6.1 各种结构的循环嵌套	120
8.6.2 循环嵌套实例	122
8.7 跳转语句	123
8.7.1 goto语句	123
8.7.2 break语句	124
8.7.3 continue语句	125
8.8 3种跳转语句的区别	126
8.9 术语	127
8.10 小结	127
8.11 练习	127
第9章 数组	128
 视频讲解：30分钟	
9.1 数组概述	128
9.1.1 数组的分类	128
9.1.2 数组的维数	128
9.2 一维数组	129
9.2.1 一维数组的定义	129
9.2.2 一维数组的引用	129
9.2.3 一维数组的初始化	130
9.2.4 一维数组的注意事项	133
9.3 二维数组	134
9.3.1 二维数组的定义	134
9.3.2 二维数组的引用	135
9.3.3 二维数组的初始化	135
9.4 多维数组	138
9.5 字符数组	138
9.5.1 字符数组的定义	138
9.5.2 字符数组的引用	138
9.5.3 字符数组的初始化	139
9.5.4 字符数组的结束标志	140
9.5.5 字符数组的输入/输出	141
9.6 常见字符串处理函数	144
9.6.1 字符串复制	144
9.6.2 字符串连接	146
9.6.3 字符串比较	147
9.6.4 字符串大小写转换	148
9.6.5 获得字符串长度	150
9.7 术语	151
9.8 小结	151
9.9 练习	151

第10章 函数	152
 视频讲解：37分钟	
10.1 函数概述	152
10.2 函数的定义	154
10.2.1 函数的定义形式	155
10.2.2 定义与声明	157
10.3 函数中的返回语句	159
10.3.1 函数返回	159
10.3.2 返回值	161
10.4 函数参数	163
10.4.1 形式参数和实际参数	163
10.4.2 数组做函数参数	165
10.5 函数的调用	169
10.5.1 函数调用方式	169
10.5.2 嵌套调用	172
10.5.3 递归调用	174
10.6 内部函数和外部函数	176
10.6.1 内部函数	176
10.6.2 外部函数	177
10.7 局部变量和全局变量	179
10.7.1 局部变量	179
10.7.2 全局变量	181
10.8 灵活应用函数	183
10.8.1 数学函数	184
10.8.2 字符和字符串函数	186
10.9 术语	189
10.10 小结	189
10.11 练习	190
第11章 指针	191
 视频讲解：17分钟	
11.1 指针概述	191
11.1.1 地址与指针	191
11.1.2 变量与指针	192
11.1.3 使用指针变量	192
11.1.4 &和*运算符	195
11.1.5 指针自加自减运算	196
11.2 数组与指针	198
11.2.1 一维数组与指针	198
11.2.2 二维数组与指针	202
11.2.3 字符串与指针	205
11.2.4 字符串数组	206

11.3 指向指针的指针	208	13.2.8 长度不同的数据进行位运算	258
11.4 函数参数中的指针变量	211	13.3 循环移位	259
11.5 返回指针的函数	219	13.4 位段的概念	262
11.6 main()函数中的指针	221	13.4.1 位段的概念与定义	262
11.7 术语	223	13.4.2 位段相关说明	263
11.8 小结	223	13.5 术语	264
11.9 练习	223	13.6 小结	264
第12章 结构体和共用体	224	13.7 练习	265
 视频讲解：28分钟		第14章 存储管理	266
12.1 理解并应用结构体	224	 视频讲解：18分钟	
12.1.1 结构体类型的概念	224	14.1 内存组织方式	266
12.1.2 结构体类型的定义	225	14.1.1 内存组织方式简介	266
12.1.3 结构体类型的引用	227	14.1.2 堆与栈	266
12.1.4 结构体类型的初始化	229	14.2 动态管理	269
12.2 掌握结构体数组	231	14.2.1 malloc()函数	269
12.2.1 定义结构体数组	231	14.2.2 calloc()函数	270
12.2.2 初始化结构体数组	233	14.2.3 realloc()函数	271
12.3 解读结构体指针	235	14.2.4 free()函数	272
12.3.1 指向结构体变量的指针	235	14.3 内存丢失	273
12.3.2 指向结构体数组的指针	237	14.4 变量的存储类别	274
12.3.3 结构体做函数参数	239	14.4.1 动态存储与静态存储	274
12.4 包含结构的结构	242	14.4.2 修饰一个局部变量为自动的auto 变量	274
12.5 理解并应用共用体	243	14.4.3 声明位置不同意义不同的static 变量	275
12.5.1 共用体的概念	243	14.4.4 寄存器存储类的register变量	276
12.5.2 共用体变量的引用	244	14.4.5 外部存储的extern变量	276
12.5.3 共用体变量的初始化	245	14.5 术语	277
12.5.4 共用体类型的数据特点	246	14.6 小结	277
12.6 术语	246	14.7 练习	277
12.7 小结	246	第15章 预处理	278
12.8 练习	247	 视频讲解：17分钟	
第13章 位运算	248	15.1 灵活使用宏定义	278
 视频讲解：20分钟		15.1.1 不带参数的宏定义	278
13.1 位与字节	248	15.1.2 带参数的宏定义	279
13.2 位运算操作符	248	15.2 #include指令	281
13.2.1 “与”运算符	249	15.3 条件编译	283
13.2.2 “或”运算符	250	15.3.1 #if命令	283
13.2.3 “取反”运算符	252	15.3.2 #ifdef及#ifndef命令	286
13.2.4 “异或”运算符	253	15.3.3 #undef命令	287
13.2.5 “左移”运算符	255	15.3.4 #line命令	287
13.2.6 “右移”运算符	257		
13.2.7 位运算复合赋值运算符	258		

15.3.5	#pragma命令	288
15.4	合理使用预处理	289
15.4.1	C语言中包括的预处理功能	289
15.4.2	使用宏时需要注意的问题	289
15.4.3	文件包含嵌套的问题	289
15.5	小结	290
15.6	练习	290
第16章 文件操作技术		291
视频讲解：36分钟		
16.1	文件概念	291
16.2	文件的基本操作	292
16.3	文件指针	293
16.4	文件的打开与关闭	293
16.4.1	文件的打开	293
16.4.2	文件的关闭	295
16.5	读/写文件的函数	295
16.5.1	写字符函数——fputc()函数	295
16.5.2	读字符函数——fgetc()函数	296
16.5.3	写字符串函数——fputs()函数	298
16.5.4	读字符串函数——fgets()函数	299
16.5.5	格式化写入函数——fprintf()函数	300
16.5.6	格式化读取函数——fscanf()函数	302
16.5.7	数据块读/写函数——fread()函数 和fwrite()函数	303
16.6	定位文件	306
16.6.1	随机读/写操作——fseek()函数	306
16.6.2	返回位置指针——rewind()函数	308
16.6.3	得到文件位置——ftell()函数	310
16.7	术语	313
16.8	小结	313
16.9	练习	313
第17章 图形图像		314
视频讲解：25分钟		
17.1	字符屏幕函数	314
17.1.1	定义文本窗口	314
17.1.2	颜色设置	315
17.1.3	文本的输入和输出	315
17.1.4	屏幕操作函数	316
17.2	图形显示函数	319
17.2.1	图形模式初始化	319
17.2.2	屏幕颜色设置	322
17.2.3	基本图形函数	324
17.2.4	封闭图形的填充	332
17.3	图形屏幕	335
17.4	图形模式下的文本输出	336
17.4.1	文本输出函数	337
17.4.2	有关文本、字体、输出方式的 设置	338
17.5	术语	340
17.6	小结	340
17.7	练习	340
第二部分 项目实战		
第18章 五子棋游戏开发		342
视频讲解：15分钟		
18.1	概述	342
18.2	需求分析	342
18.3	系统设计	342
18.3.1	设计目标	342
18.3.2	开发及运行环境	343
18.4	程序预览	343
18.5	graphics.h文件	344
18.6	设计思路	346
18.7	预处理	347
18.7.1	文件引用	347
18.7.2	宏定义	347
18.8	声明变量	347
18.9	函数声明	347
18.10	主要功能实现	348
18.10.1	主函数	348
18.10.2	开始游戏	349
18.10.3	绘制棋盘	350
18.10.4	绘制棋子	350
18.10.5	清除棋子	350
18.10.6	游戏过程	351
18.10.7	判断胜负结果	356
18.11	开发总结	358
第19章 学生管理系统		359
视频讲解：10分钟		
19.1	需求分析	359
19.2	系统设计	359

第二部分 项目实战

第18章	五子棋游戏开发	342
 视频讲解: 15分钟		
18.1	概述	342
18.2	需求分析	342
18.3	系统设计	342
18.3.1	设计目标	342
18.3.2	开发及运行环境	343
18.4	程序预览	343
18.5	graphics.h文件	344
18.6	设计思路	346
18.7	预处理	347
18.7.1	文件引用	347
18.7.2	宏定义	347
18.8	声明变量	347
18.9	函数声明	347
18.10	主要功能实现	348
18.10.1	主函数	348
18.10.2	开始游戏	349
18.10.3	绘制棋盘	350
18.10.4	绘制棋子	350
18.10.5	清除棋子	350
18.10.6	游戏过程	351
18.10.7	判断胜负结果	356
18.11	开发总结	358
第19章	学生管理系统	359
 视频讲解: 10分钟		
19.1	需求分析	359
19.2	系统设计	359

19.3 预处理	360	21.1.1 需求分析	388
19.3.1 文件引用	360	21.1.2 开发工具选择	388
19.3.2 宏定义	360	21.2 系统设计	388
19.4 函数声明	360	21.2.1 系统目标	388
19.5 功能设计	361	21.2.2 系统功能结构	389
19.5.1 功能选择界面	361	21.2.3 系统预览	389
19.5.2 输入学生信息界面	362	21.2.4 开发及运行环境	391
19.5.3 查询学生信息界面	365	21.3 数据库设计	391
19.5.4 删除学生信息界面	366	21.3.1 安装MySQL数据库	392
19.5.5 修改学生信息界面	367	21.3.2 启动MySQL数据库	398
19.5.6 插入学生信息界面	369	21.3.3 创建数据库	398
19.5.7 学生成绩排名	370	21.3.4 数据表结构	399
19.5.8 统计学生人数	372	21.4 C语言开发数据库程序的流程	400
19.6 开发总结	373	21.4.1 引入头文件	400
第20章 商品信息管理系统	374	21.4.2 引入库函数	401
 视频讲解：17分钟		21.5 C语言操作MySQL数据库	402
20.1 需求分析	374	21.5.1 MySQL常用数据库操作函数	402
20.2 预处理	374	21.5.2 连接MySQL数据库	404
20.2.1 文件引用	375	21.5.3 查询图书表记录	405
20.2.2 宏定义	375	21.5.4 插入图书表记录	407
20.3 声明结构体	375	21.5.5 修改图书表记录	407
20.4 函数声明	375	21.5.6 删除图书表记录	408
20.5 功能选择界面	376	21.6 文件引用	409
20.6 输入商品信息	377	21.7 变量和函数定义	410
20.7 查询商品信息	379	21.8 主要功能模块设计	410
20.8 删除商品信息	380	21.8.1 显示主菜单信息	410
20.9 修改商品信息	382	21.8.2 显示所有图书信息	412
20.10 插入商品信息	383	21.8.3 添加图书信息	415
20.11 商品总额排名	385	21.8.4 修改图书信息	420
20.12 统计商品数	386	21.8.5 删除图书信息	425
20.13 开发总结	387	21.8.6 查询图书信息	431
第21章 图书管理系统	388	21.9 开发总结	433
 视频讲解：17分钟		附录A ASCII码表	434
21.1 概述	388	附录B 运算符的优先级	437

第一部分 开发技术

章目

- ☐ 第0章 如何学好C语言
- ☐ 第1章 初识C语言
- ☐ 第2章 C语言的战前准备
- ☐ 第3章 C语言与算法
- ☐ 第4章 C语言基础
- ☐ 第5章 表达式与运算符
- ☐ 第6章 输入/输出函数
- ☐ 第7章 选择结构程序设计
- ☐ 第8章 循环结构程序设计
- ☐ 第9章 数组
- ☐ 第10章 函数
- ☐ 第11章 指针
- ☐ 第12章 结构体和共用体
- ☐ 第13章 位运算
- ☐ 第14章 存储管理
- ☐ 第15章 预处理
- ☐ 第16章 文件操作技术
- ☐ 第17章 图形图像

第0章 如何学好C语言

博学之，审问之，慎思之，明辨之，笃行之。

——《礼记》

俗语说：“水滴石穿，绳锯木断。”无论做什么事情，都必须坚定信心，保持高昂的斗志，持之以恒是做事的根本。学习C语言同样是如此，首先必须坚定信心，让自己充满激情，然后，脚踏实地，认真真地去学习，去迎接每一个挑战，只有经过暴风雨的洗礼，才能看到绚丽的彩虹。图0.1所示为程序员的成长之路。

C语言是应用非常广泛的一种高级语言，其历史相当久远。刚开始学习C语言，可能其灵活性、要记忆的东西很多，但正是这种灵活性带来了可读性好、语法简单、效率高等优点。当然，要学好C语言，不仅要透彻理解书中的概念，还要辅之以大量上机编程。要想提高应用水平，还要多看应用方面的书。学习C语言不是一朝一夕的事情，但也不需要花费十年八年的时间才能精通。下面就来简单介绍一下其学习方法。

首先，要有一个正确的心态，没有正确态度的人是学不好任何东西的。切不可三心二意，“三天打鱼两天晒网”。

其次，要明确学习该语言的目的，是想真正掌握这一门语言，还是单纯为了应付考试，两者有很大的区别，这将决定学习这门语言要学到怎样的一个深度。

最后，即是学的问题。应先从整体上来理解C语言，认识C语言的优势，以及C语言的特点。C语言是面向过程的一门语言，特别强调各个程序的关系，如程序、函数之间的调用等。

一门语言的基础部分只要能静下心来认真看，还是很容易理解并记住的。首先基本语法一定要牢记。有一种现象很普遍，就是看过一遍后，当时觉得懂了，但却没有加深、巩固，在看到后面知识涉及前面的语法时，才发现以前看过的知识记得不是很清晰，甚至忘得一干二净。不用着急，把书翻到前面重新查阅一下相关知识即可。这时不用看得太仔细，只需要粗略地看个大概，就可以将以前学过的内容重新捡回来。

动手实际操作环节是十分重要的，要学好一门语言就要多上机练习，发现错误之后及时改正会使自己进步更快。在这一阶段，可以编写一些简单的程序，以此来熟悉C语言的编程环境、数据类型。这样理解起来会更快些，也比较容易记忆。上机实践时，切记千万不能照着书一味地打代码，应该先看懂这个程序，觉得完全懂时，再将书合上，凭着对程序的理解，重新编写程序。如果程序编写正确，说明该内容真的完全掌握了；如果出现错误，说明在某些知识方面还有所欠缺，需要进一步改进。

接下来，就是学习一些过程语言的基础模式，如顺序、循环、选择等。这些内容的灵活性较高，千万不能死记硬背，要在理解的基础上去记忆，对同一个问题尝试用多种方法去实现。

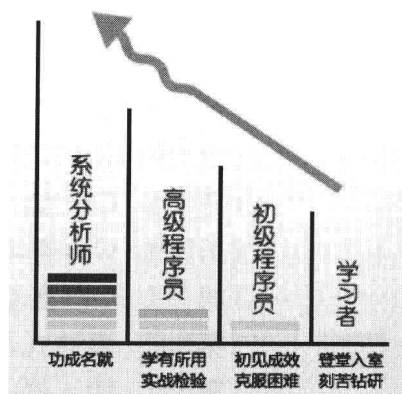


图0.1 程序员的成长之路

在前面的基础上，再开始学习函数、指针、文件等知识。这些知识相比之前的内容会觉得比较抽象，难以理解。这时也不要有什么畏惧心理，要使复杂的问题变得容易掌握就需要对其基本概念理解透彻，在扎实掌握基本知识的基础上逐步加深难度，再复杂的问题也会慢慢简单化。要多借鉴别人写好的例程，体会别人编写程序的思想，这样能让自己少走很多弯路，节省大量时间。在编程过程中不懂的地方要及时解决，上网查资料、请教老师、咨询技术人员等均可。

编程应作为一种习惯，经常写程序有助于个人能力的提升。多看一些别人写的知识点总结或编程技巧，对自己会有很大的帮助。随着学习的深入，会发现学习C语言其实是很有趣的，这样学习就不再是一件难事，而是充满了快乐。

最后，多上机编程和调试，多动脑，就会快速掌握C语言。

第1章 初识C语言

好的或者成功的IT发明不是标新立异，而是把过去一个没做完的事情做完。

——茅道临

本章主要知识点

- ☐ C语言的发展史
- ☐ C语言的特点
- ☐ C语言的程序结构

1.1 C语言介绍

 教学录像：光盘\mr\video\第1章\C语言介绍.exe

计算机语言有机器语言、汇编语言和高级语言3种，C语言属于高级语言。C语言可以说是高级语言中的元老，很多语言都是在C语言的分支中发展出来的，如VC、PHP、Java Web等，他们的身上或多或少的都带有C语言的身影。

C语言在过去非常流行，现在在编程语言的排行榜中稳居前三名，发展前景是非常可观的。原因在于，C语言具有使用方便、功能强大、应用广泛、移植性高等特点。C语言不仅仅可以开发程序，而且非常适合编写系统软件。

学习C语言，首先要对程序语言的发展有一定的了解，下面来了解3种程序设计语言所具有的特点以及C语言的发展历史。

1.1.1 程序语言简述

程序设计语言的发展一共经历了以下三个阶段。

☐ 机器语言

机器语言是低级语言，又称二进制代码语言，就是写出一串串由“0”和“1”组成的指令序列交由计算机执行。机器语言的特点是：计算机可以直接识别，不需要进行任何翻译。但是机器语言程序长，难记，难理解，且不易于查错。不同型号的计算机其机器语言是不相通的，按一种计算机的机器指令编制的程序，不能在另一种计算机上执行。

☐ 汇编语言

汇编语言是面向机器的程序设计语言。为了减轻使用机器语言编程的痛苦，用英文字母或者符号串来替代机器语言的二进制码，这样就产生了汇编语言。使用汇编语言比使用机器语言更方便阅读和容易理解程序。

☐ 高级语言

由于汇编语言依赖于硬件体系，并且汇编语言中的助记符号数量比较多，为了使程序语言能更贴近人类的自然语言，同时又不依赖于计算机硬件，便产生了高级语言。这种语言的语法形式类似于英文，且因为远离对硬件的直接操作，更易于普通人的理解与使用。其中影响较大、使用比较普遍的有：**FORTRAN**、**ALGOL**、**BASIC**、**COBOL**、**LISP**、**Pascal**、**PROLOG**、**C**、

C++、VC、VB、Delphi、Java等高级语言。

1.1.2 C语言的发展史

从程序语言的发展过程可以看到，以前的操作系统等系统软件主要是用汇编语言编写的。但是由于汇编语言依赖于计算机硬件，程序的可读性和可移植性都不是很好，为了提高程序的可读性和可移植性，人们开始寻找一种语言，这种语言应该既具有高级语言的特性，又不失低级语言的好处。于是，在这种需求下产生了C语言。

C语言是在由UNIX的研制者丹尼斯·里奇（Dennis Ritchie）和肯·汤普逊（Ken Thompson）于1970年研制出的BCPL语言（B语言）的基础上发展和完善起来的。20世纪70年代初，AT&T Bell实验室的程序员Dennis Ritchie第一次把B语言改为C语言。

最初，C语言运行于AT&T的多用户、多任务的UNIX操作系统上。后来，Ritchie用C语言改写了UNIX C的编译程序，UNIX操作系统的开发者Ken Thompson又用C语言成功地改写了UNIX，从此开创了编程史上的新篇章。UNIX成为第一个不是用汇编语言编写的主流操作系统。

1983年，美国国家标准委员会（ANSI）对C语言进行了标准化，于1983年颁布了第一个C语言草案（83ANSI C），后来于1987年又颁布了另一个C语言标准草案（87ANSI C），最新的C语言标准C99在1999年颁布，并在2000年3月被ANSI采用。但是由于未得到主流编译器厂家的支持，C99并未广泛使用。

尽管C语言发展于大型商业机构和学术界的研究实验室，但是当开发者们为第一台个人计算机提供C编译系统之后，C语言就得以广泛传播，为大多数程序员所接受。对MS-DOS操作系统来说，系统软件和实用程序都是用C语言编写的。Windows操作系统大部分也是用C语言编写的。

C语言是一种面向过程的语言，同时具有高级语言和汇编语言的优点。C语言可以广泛应用于不同的操作系统，如UNIX、MS-DOS、Microsoft Windows及Linux等。

在C语言的基础上发展起来的有支持多种程序设计风格的C++语言，如网络上广泛使用的Java、JavaScript、微软的C#语言等。也就是说，学好C语言，就会在短时间内轻松掌握其他语言。

说明：目前最流行的C语言有Microsoft C或称MS C、Borland Turbo C或称Turbo C和AT&T C几种。

1.2 选择C语言的理由

 教学录像：光盘\mr\video\第1章\选择C语言的理由.exe

C语言是一种通用的程序设计语言。C语言主要用来进行系统程序设计，其特点有以下几点。

☐ 应用面广

C语言是一种面向过程的结构化程序设计语言。既可编写系统软件（如DOS、UNIX），又可编写应用软件。

☐ 高效性

谈到高效性，不得不说C语言是“鱼与熊掌”兼得。从C语言的发展历史也可以看到，C语言继承了低级语言的优点，产生了高效的代码，并具有友好的可读性和编写性。一般情况下，