

荡胸生层云： C语言开发修行实录

席国庆 等编著

1. 循序渐进的学习主线

用引人入胜的情节，渗透作者数年学习和开发经历，由浅入深地讲解完整的编程技术知识体系，并达到学以致用、学以致用的学习效果。

2. 场景式教学

生动的日记体裁，勾勒了一个程序员的成长历程。让读者在学习过程中置身其中，切换到编程思维，营造一种愉快的编程学习的氛围。

3. 精选案例，丰富的技巧提示

本书完全采用理论加实践的讲解模式，在每个知识点讲解之后，都用一个很有代表性的实例来演练知识点的用法，同时配以大量的技巧提示。

4. 用问题串联知识点，启发式讲解方式

针对学习进度，逐步提出问题，引发读者思考，环环相扣、前后关联，最终互动式地解决问题。书中提出了大量测试和问题，通过这些测试和问题将各个知识点串联起来。

5. 通俗的语言，大白的讲解模式

让读者仿佛是在看一个故事，而不是在学习深奥的编程语言知识。各处妙语，生动串起关键技术点。



编程.望岳

荡胸生层云: C语言开发修行实录

席国庆 等编著

電子工業出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

C 语言是当今使用最为频繁的编程语言之一，一直在开发领域占据重要的地位。本书循序渐进、由浅入深地详细讲解了 C 语言的核心技术，并通过具体实例的实现过程演练了各个知识点的具体使用流程。全书分为 4 篇共 18 章。其中第 1~第 8 章是基础篇，讲解了 C 语言开发的基本知识，包括 C 语言地位、算法、流程语句、数组和字符串、函数、指针；第 9~第 12 章是核心技术篇，包括结构体、共用体和枚举、链表、位运算、预编译；第 13~第 15 章是提高篇，包括文件操作、错误调试、高级编程技术；第 16~第 18 章是综合实战篇，分别通过 3 个综合实例的实现过程，讲解了 C 语言在综合项目中的使用流程。全书采用故事性、趣味性相结合的对话讲解方式，并穿插了学习技巧和职场生存法则，引领读者全面掌握 C 语言。

本书不但适用于 C 语言的初学者，也适用于有一定 C 语言基础的读者，甚至也可以作为有一定造诣的程序员们的参考书。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目 (CIP) 数据

荡胸生层云：C 语言开发修行实录 / 席国庆等编著. —北京：电子工业出版社，2011.4
(编程·望岳)
ISBN 978-7-121-13050-2

I. ①荡… II. ①席… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2011) 第 038808 号

策划编辑：田小康

责任编辑：徐津平

文字编辑：王 静

印 刷：北京天宇星印刷厂

装 订：三河市皇庄路通装订厂

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1092 1/16 印张：28.5 字数：700 千字

印 次：2011 年 4 月第 1 次印刷

印 数：4000 册 定价：59.80 元 (含 DVD 光盘 1 张)

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：(010) 88258888。

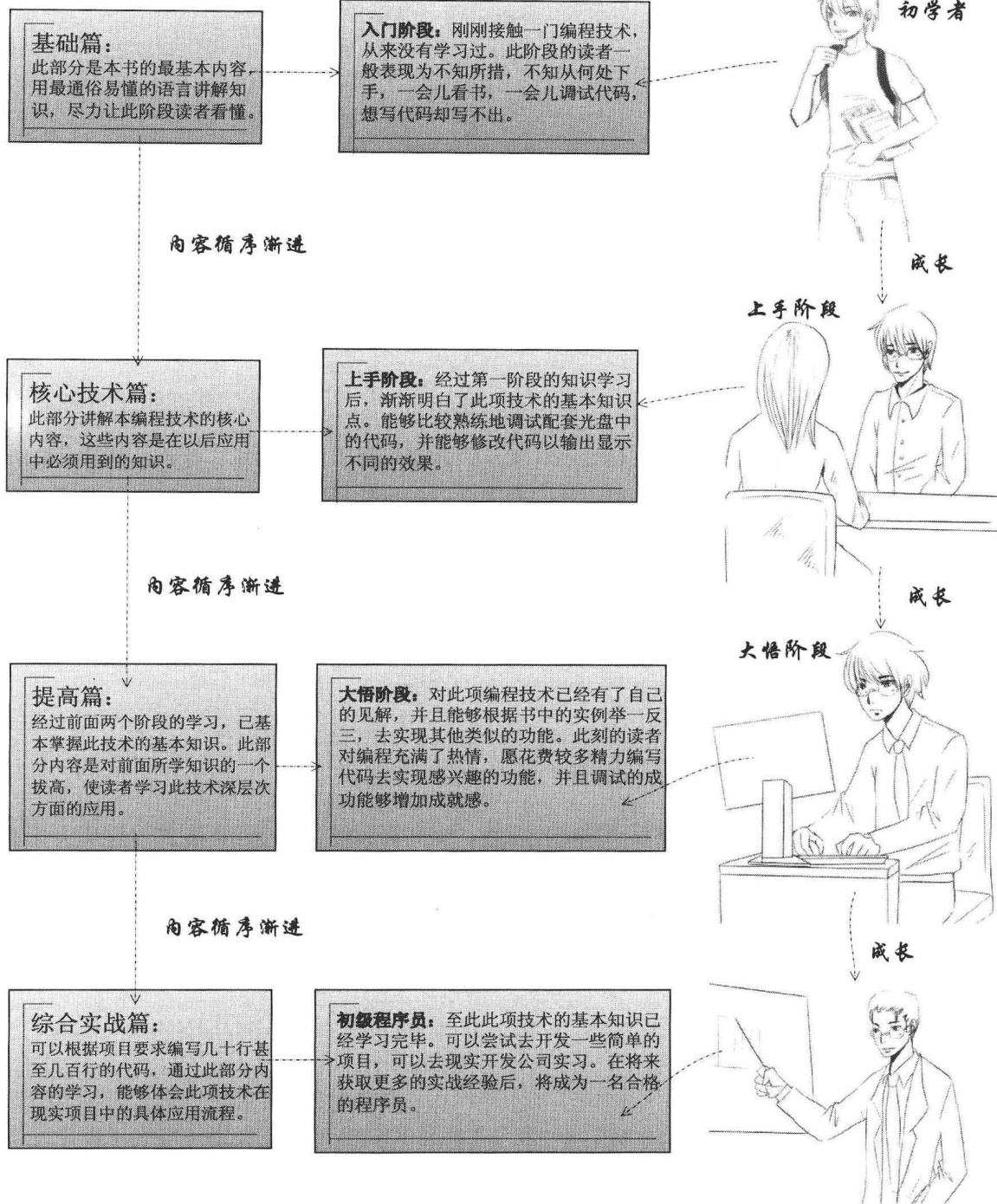
前言

C 语言是目前国内外使用最为广泛的程序设计语言之一。它具有功能丰富、表达能力强、使用方便灵活、执行效率高、可移植性好等优点,几乎可用于所有领域。C 语言既具有高级语言的特点,也具有汇编语言的功能,还具有很强的系统处理能力,可以直接对硬件和外部接口进行控制。C 语言被广泛应用于系统软件和应用软件的开发。

使用 C 语言进行程序设计和软件开发,可以熟悉并理解计算机内部的工作原理,对于深入学习计算机技术大有裨益。C 语言是计算机科学与技术专业的基础课程,是以后学习数据结构与算法的基础,也为以后选择 Visual C++或 Java 软件开发奠定了基础。因此,只有熟练地掌握了 C 语言,以后才能更加深入地掌握计算机技术。

本书分为4篇，分别是基础篇、核心技术篇、提高篇和综合实战篇。各篇内容从头开始循序渐进，和读者的成长同步。一般来说，读者学习程序的过程分为4个阶段：入门阶段、上手阶段、大悟阶段和初级程序员。为了表述得更加直观，下面通过一幅图来展示学习程序的成长历程。

本书的最大特色是用一问一答的模式穿插知识点，用故事性和趣味性的语言来讲解枯燥的编程知识。本书讲述了一个编程初学者“我”在师兄 KNOWALL 的帮助下顺利掌握开发技术的成长历程。作者用日记的格式全程记录了学习历程，日记和生活密切相关，也和知识点相关，使读者阅读后有一种豁然开朗的感觉。



(1) 日记记录, 以作者学习经历为主线

作者用日记格式记录了过去的学习历程，从作者学生时代讲起，一直贯穿整个学习过程。以日记的方式记录下了学习过程中的点点滴滴，每个日记故事不仅和生活紧密相关，而且还和编程的知识点相关，趣味和知识并重，情节引人入胜。

(2) 每个实例是精心挑选典型代表, 丰富的技巧提示

书中遵循了理论结合的写作模式，在知识点讲解完毕之后，都会用一个具体实例来演练知识点的用法。所有实例都具有典型和代表性。

(3) 通俗的语言，大话的讲解模式

以最通俗的语言来实现，以大话的方式将知识展现给读者。让读者仿佛是在看一个故事，而不是在学习深奥的语言知识。

(4) 用问题串联知识点

为了满足读者的成功感，书中提出了大量测试和问题，通过这些测试和问题将各个知识点串联起来，这样能让读者在问题中学会知识点的运用，一举两得。

(5) 提出问题到解决问题，问题一环扣一环

第 1 步：每章的开始，我们将提出一个能涵盖本章知识的问题。

第2步：详细讲解需要解决这些问题的基本知识。

第3步：解决这个问题。

(6) 超值 DVD 光盘 (PPT 素材和视频、全书案例源码)

- ▶ **实例源代码：**包括了书中全部实例的完整源代码。
- ▶ **开发技术视频教学课堂：**结合全书技术体系，用视频教学方式讲解了书中所涉及的所有知识点。
- ▶ **实例视频教学课堂：**用视频教学的方式全面讲解了书中每个实例的具体实现过程。
- ▶ **精美 PPT 教案：**结合全书内容体系，为教学工作者提供了精心制作的 PPT 教案。
- ▶ **超值大赠送：**额外免费赠送给读者 5 个综合案例的完整源码和实例教学视频。

本书的内容

第1~第8章是基础篇,介绍了C语言开发所必须具备的基本知识,逐一讲解了C语言的地位、算法、流程语句、数组和字符串、函数、指针等领域的知识,并采用理论结合实例的方式对各个知识点进行剖析。

第9~第12章是核心技术篇,逐一讲解了结构体、共用体和枚举、链表、位运算、预编译等领域的知识,并采用理论结合实例的方式对各个知识点进行剖析。

第13~第15章是提高篇,逐一讲解了文件操作、错误调试、图形图像编程、绘图处理、网络编程等领域的知识,并采用理论结合实例的方式对各个知识点进行剖析。

本书的读者对象

致谢

编者
2011年3月

Contents

目 录

第一篇 基础篇

第 1 章 学 C 语言还有用吗	1	2.3 不同的数据类型，品百态人生	23
1.1 引出问题	1	2.4 变量和常量，体会变和不变	23
1.2 漫漫发展历程	2	2.4.1 不变的常量	24
1.3 C 语言特点	2	2.4.2 可变的变量	25
1.4 工欲善其事，必先利其器	3	2.5 整型数据	28
1.4.1 先看 Turbo C/C++ 3.0	4	2.5.1 整型常量	28
1.4.2 再看 DEV-C++ 5	10	2.5.2 整型变量	29
1.5 我的第一次 C 语言程序开发	11	2.6 实实在在的实型数据	31
1.5.1 第一步：编辑源代码	12	2.6.1 不变的实型常量	31
1.5.2 第二步：编译	13	2.6.2 变化的实型变量	32
1.5.3 第三步：运行	13	2.6.3 实型常量的类型	33
1.6 庖丁解牛 C 程序	13	2.7 字符型数据，人生如符号	34
1.6.1 C 程序组成部分	14	2.7.1 字符常量	34
1.6.2 C 程序的书写格式	16	2.7.2 字符串常量	36
1.7 解决问题——究竟 C 语言还有用吗	17	2.7.3 字符变量	36
1.8 我的总结	18	2.8 从初始化变量谈人生初始化	38
第 2 章 看人生算法	19	2.9 整型、实型、字符型数据间的运算	39
2.1 引出问题	19	2.9.1 自动转换	39
2.2 何谓算法	20	2.9.2 强制转换	40
2.2.1 算法的概念	20	2.10 解决问题——规划你的人生算法，两弊相衡取其轻	41
2.2.2 用流程图表示算法	21	2.11 我的总结	42
2.2.3 用 C 语言表示算法	22		

第 3 章 运算符和表达式, 选择你的生活方式	43		
3.1 引出问题	43		
3.2 人生有多种选择	44		
3.2.1 运算符的种类	44		
3.2.2 运算符的优先级	46		
3.3 算术运算符拨慢时钟的运转	47		
3.3.1 最简单的单目运算符	47		
3.3.2 双目运算符	48		
3.4 赋值运算符和赋值表达式	49		
3.4.1 基本赋值运算符	50		
3.4.2 复合赋值运算符	52		
3.4.3 赋值表达式	53		
3.5 关系运算下的男女关系	54		
3.6 逻辑人生下的逻辑运算符和逻辑表达式	56		
3.7 人生需要停顿的逗号运算	58		
3.8 花落知多少, 求字节数运算	60		
3.9 解决问题——程序员的 7 种生活方式	62		
3.10 我的总结	63		
第 4 章 进进出出的 C 语句	64		
4.1 引出问题	64		
4.2 语句——C 程序的最重要组成	65		
4.2.1 C 语句简介	65		
4.2.2 看空白对语句的作用	67		
4.2.3 赋值语句给你一个值	67		
4.3 输入和输出, 是一个进进出出的过程	68		
4.3.1 putchar 函数 (字符输出函数)	69		
4.3.2 getchar 函数 (字符输入函数)	70		
4.3.3 printf 函数 (格式输出函数)	72		
4.3.4 scanf 函数 (格式输入函数)	74		
4.3.5 puts 函数 (字符串输出函数)	77		
4.3.6 gets 函数 (字符串输入函数)	78		
4.4 解决问题	79		
4.5 我的总结	80		
第 5 章 和人生一样的流程控制	82		
5.1 引出问题	82		
5.2 按部就班的顺序结构	83		
5.3 人生可以选择	84		
5.3.1 一个选择的单分支结构语句	85		
5.3.2 有多个选择的双分支结构语句	86		
5.3.3 复杂的多分支结构语句	88		
5.3.4 条件运算符和条件表达式	89		
5.4 人生是一个循环	91		
5.4.1 for 语句循环	92		
5.4.2 while 语句	94		
5.4.3 do-while 语句	96		
5.4.4 循环语句的总结	97		
5.4.5 goto 语句	98		
5.4.6 break 语句	99		
5.4.7 continue 语句	100		
5.4.8 死循环和退出程序	101		
5.5 解决问题	103		
5.6 我的总结	104		
第 6 章 数组和字符串, 心灵的港湾	105		
6.1 引出问题	105		
6.2 最简单的一维数组	106		
6.2.1 定义一维数组	106		
6.2.2 引用一维数组元素	107		
6.2.3 初始化一维数组	108		
6.3 纷乱思绪下的多维数组	110		
6.3.1 二维数组	111		
6.3.2 多维数组	114		

6.4	功名利禄下的字符数组与字符串	117	7.7	函数嵌套和递归, 组合你的武器	153
6.4.1	字符数组	117	7.7.1	嵌套调用	153
6.4.2	字符串与字符数组	118	7.7.2	递归调用	156
6.4.3	字符数组的输入输出	119	7.8	变量的作用域和生存期	157
6.5	净化心灵的字符串处理函数	121	7.8.1	变量作用域	158
6.5.1	测试字符串长度	121	7.8.2	静态存储变量和动态存储变量	160
6.5.2	字符串大小写转换	122	7.9	内外兼修的内部函数和外部函数	167
6.5.3	字符串复制	123	7.9.1	内部函数	167
6.5.4	字符串比较	124	7.9.2	何谓外部函数	168
6.5.5	字符串连接	126	7.10	库函数我来了	169
6.5.6	其他字符串函数	127	7.10.1	C 库函数介绍	169
6.5.7	将字符串转换成数值的函数	127	7.10.2	C 库函数分类	169
6.6	豁然开朗的字符处理函数	128	7.11	信心是最重要的武器	170
6.6.1	字符检测函数	129	7.12	我的总结	173
6.6.2	字符大小写转换函数	131			
6.7	解决问题	132	第 8 章	一个萝卜一个坑的指针	174
6.8	我的总结	133	8.1	引出问题	174
第 7 章	函数, 最厉害的武器	134	8.2	为需求而生的指针	175
7.1	引出问题	134	8.3	变量的指针和指向变量的指针变量	175
7.2	函数, 一个个致命的武器	135	8.3.1	声明	176
7.2.1	看函数的分类	135	8.3.2	初始化	177
7.2.2	函数的定义	137	8.3.3	引用	178
7.3	第一步, 函数的声明和函数原型	138	8.3.4	最后几个说明	179
7.4	武器的零件——参数	140	8.3.5	指针变量运算	180
7.4.1	形参和实参	140	8.3.6	指针变量作为函数参数	181
7.4.2	数组名作为函数参数	142	8.3.7	void 类型的指针	184
7.4.3	数组作为函数参数	144	8.4	双子星般的指针和数组	186
7.5	组装的结果——返回值	146	8.4.1	数组元素的指针	186
7.6	使用你的武器——调用函数	148	8.4.2	指向一维数组元素的指针变量	188
7.6.1	函数调用格式	149	8.4.3	通过指针引用数组元素	189
7.6.2	调用函数的方式	149	8.4.4	使用指针变量应该注意的问题	192
7.6.3	对被调函数的召唤	150	8.4.5	数组名作函数参数	195
7.6.4	调用函数方式的深入分析	151			

8.5	指针和多维数组	197	8.7	指针数组和多级指针	205
8.5.1	多维数组的地址	197	8.7.1	指针数组	206
8.5.2	指向多维数组的指针变量	199	8.7.2	多级指针的定义和应用	207
8.6	指针和字符串	201	8.7.3	指向指针的指针	209
8.6.1	指针访问字符串	202	8.7.4	main 函数的参数	209
8.6.2	字符串指针作函数参数	203	8.8	指针函数和函数指针	210
8.6.3	使用字符串指针变量与 字符数组的区别	204	8.9	解决问题	214
			8.10	我的总结	214

第二篇 核心技术篇

第 9 章	结构体、共用体和枚举	216	9.7	枚举	239
9.1	引出问题	216	9.7.1	定义枚举类型	239
9.2	先谈结构体	216	9.7.2	定义枚举变量	240
9.2.1	定义结构体类型	217	9.7.3	枚举变量的引用	241
9.2.2	结构体类型变量的定义	217	9.8	用 typedef 定义类型	241
9.2.3	结构体变量的引用	220	9.9	解决问题	243
9.2.4	结构体变量的初始化	221	9.10	我的总结	245
9.3	结构体数组	223	第 10 章	收放自如的链表	246
9.3.1	结构体数组定义	223	10.1	引出问题	246
9.3.2	结构体数组的初始化	225	10.2	动态内存分配	246
9.3.3	结构体数组的引用	226	10.3	链表	249
9.4	结构体指针	227	10.3.1	侃侃链表	249
9.4.1	结构体指针变量的定义	228	10.3.2	单向链表	251
9.4.2	结构体指针变量的初始化	228	10.3.3	创建链表	253
9.4.3	结构体指针变量的引用	229	10.3.4	删除整个链表	253
9.4.4	指向结构体变量的指针	229	10.3.5	在链表中插入结点	253
9.4.5	指向结构体数组的指针	231	10.3.6	在链表中删除结点	253
9.5	结构体和函数	231	10.3.7	双向链表	254
9.5.1	结构体变量和结构体指针 作为函数参数	232	10.3.8	循环链表	257
9.5.2	返回结构体类型值的函数	233	10.4	解决问题	257
9.6	朋友，共用体（联合）	234	10.5	我的总结	258
9.6.1	共用体和共用体变量 的定义	235	第 11 章	更精细的位运算	259
9.6.2	共用体变量的引用与 初始化	236	11.1	引出问题	259
9.6.3	结构体和共用体的区别	238	11.2	看 6 种位运算符和位运算	260
			11.2.1	按位与运算	260
			11.2.2	按位或运算	261

11.2.3	按位异或运算	262
11.2.4	取反运算	263
11.2.5	左移运算	264
11.2.6	右移运算	264
11.3	人生范围般的位域	265
11.4	解决问题	267
11.5	我的总结	268

第 12 章 预编译是一种准备活动 269

12.1	引出问题	269
12.2	三种预编译指令	270
12.3	何谓宏定义	270
12.3.1	不带参数的宏定义	271
12.3.2	带参数的宏定义	273
12.3.3	字符串化运算符	274

12.3.4	并接运算符	274
12.4	文件包含	275
12.5	条件编译	275
12.5.1	#ifdef ... #else ... #endif 命令	276
12.5.2	#if defined ... #else ... #endif	277
12.5.3	#ifndef ... #else ... #endif	277
12.5.4	#if !defined ... #else ... #endif	278
12.5.5	#ifdef ... #elif ... #elif ... #else ... #endif	278
12.6	解决问题	280
12.7	我的总结	282

第三篇 提高篇

第 13 章 文件操作很重要 283

13.1	引出问题	283
13.2	说说文件	283
13.2.1	文本文件	284
13.2.2	文件分类	284
13.3	文件指针	285
13.4	文件打开与关闭	285
13.4.1	打开文件	286
13.4.2	关闭文件	288
13.5	文件读写操作	290
13.5.1	字符读写函数	290
13.5.2	字符串读写函数	292
13.5.3	格式化读写函数	294
13.5.4	读写数据块函数	295
13.5.5	其他读写函数	297
13.6	文件的随机读写	298
13.6.1	fseek 函数	299
13.6.2	rewind 函数	300
13.6.3	ftell 函数	301
13.7	文件管理函数	302

13.7.1	删除文件	303
13.7.2	重命名文件	303
13.7.3	复制文件	304
13.8	检测你的状态	306
13.8.1	feof 函数	306
13.8.2	ferror 函数	306
13.8.3	clearerr 函数	307
13.9	解决问题	308
13.10	我的总结	309

第 14 章 人孰能无错 310

14.1	引出问题	310
14.2	常见错误分析	310
14.3	今天你错了吗	316
14.4	知错就改	318
14.5	我的总结	321

第 15 章 C 语言高级编程技术 323

15.1	引出问题	323
15.2	看 C 语言的高级编程技术	323
15.3	文本的屏幕输出和键盘 输入	324

15.3.1	文本的屏幕输出	324	关画图函数	345
15.3.2	用键盘输入	330	15.4.8	图形窗口函数
15.3.3	应用实例	331	15.4.9	图形方式下的文本输出 函数
15.4	图形显示方式和鼠标输入	333	15.5	不得不说的菜单
15.4.1	图形模式的初始化	334	15.5.1	实现下拉式菜单
15.4.2	清屏和恢复显示方式的 函数	337	15.5.2	实现弹出式菜单
15.4.3	独立图形运行程序的 建立	337	15.6	网络编程是全新的境界
15.4.4	基本绘图函数	338	15.6.1	常用协议报头
15.4.5	画线的线性函数	341	15.6.2	Winsock 基础
15.4.6	颜色控制函数	342	15.7	解决问题
15.4.7	封闭图形的填色函数及有		15.8	我的总结

第四篇 综合实战篇

第 16 章	俄罗斯方块游戏	365	17.3.2	功能模块设计	392
16.1	老师的作业	365	17.4	设计数据结构	394
16.2	游戏总体设计	366	17.5	规划系统函数	395
16.2.1	规划设计	366	17.6	写代码	397
16.2.2	数据结构设计	368	17.6.1	预处理	397
16.3	构成函数介绍	370	17.6.2	初始化模块处理	398
16.4	游戏具体实现	371	17.6.3	获取参数	399
16.4.1	预处理	371	17.6.4	用户帮助模块	401
16.4.2	主函数	375	17.6.5	广播信息发送模块	402
16.4.3	初始化界面处理	377	17.6.6	广播消息接收模块	404
16.4.4	时钟中断处理	378	17.6.7	多播功能控制模块	405
16.4.5	成绩、速度和帮助处理	379	17.6.8	多播消息发送模块	407
16.4.6	满行处理	380	17.6.9	多播消息接收模块	408
16.4.7	方块显示和消除处理	382	17.6.10	主函数	409
16.4.8	游戏方块操作判断处理	385	17.7	项目调试	410
16.5	我的总结	388	17.7.1	系统调试	411
第 17 章	我的论文——UDP 传输 系统	389	17.7.2	验收	411
17.1	我的毕业论文	389	17.8	我的总结	411
17.2	实现流程	391	第 18 章	学生成绩管理系统	412
17.3	项目规划分析	391	18.1	求职路上	412
17.3.1	功能描述	392	18.2	系统规划	414
			18.2.1	项目的意义	414

18.2.2	功能描述	415	18.4.6	格式化输入数据	426
18.3	系统总体设计	416	18.4.7	增加学生记录	426
18.3.1	功能模块设计	416	18.4.8	查询学生记录	428
18.3.2	数据结构设计	418	18.4.9	删除学生记录	429
18.3.3	构成函数介绍	419	18.4.10	修改学生记录	431
18.4	具体实现	420	18.4.11	插入学生记录	432
18.4.1	预处理	421	18.4.12	统计学生记录	434
18.4.2	主函数 main	421	18.4.13	排序处理	435
18.4.3	系统主菜单函数	423	18.4.14	存储学生信息	436
18.4.4	表格显示信息	424	18.5	项目调试	437
18.4.5	信息查找定位	425	18.6	我的总结	439

第一篇 基础篇

第 1 章 学 C 语言还有用吗

用“学 C 语言还有用吗”这个名字作为章名，我觉得是一句废话，但是现实中有很多初学者都有这个疑问。C 语言是当前所有开发语言中使用最为广泛的，从它诞生之日起就深受人们的喜爱。因为 C 语言的普及，使得后来的开发语言都或多或少地遵循了它的模式。所以，几乎所有的程序员都将 C 语言称为自己的技术起步语言。通过对 C 语言的学习和了解，使我们能够深入地理解操作系统的运作方式，以及编程思想核心理念。

1.1 引出问题

9月9日，7:00，晴

秋高气爽，天空依然很蓝！今天是升入大学后的第八天，此时我的心里还是充满了各种新奇：学校的图书馆很大，师兄师姐们的毕业产品很了不起……此刻的我带着新奇的心情领来了自己的教材。

我：“我领回教材了，竟然发现有 C 语言的教材！记得 C 语言很早就有了，我听说学习 Java、C++ 和 C# 才是王道，现在学 C 语言还有用吗？”

KNOWALL：“我只能很负责地对你说，C 语言是当今使用最为频繁的编程语言之一，并且一直位列前三甲！”

我：“C 语言真有这么厉害吗？”

KNOWALL：“你说的 C++、Java 和 C#，都是在 C 语言的基础上发展起来的，可以说 C 语言是它们的鼻祖，它们之间的关系如图 1-1 所示。”

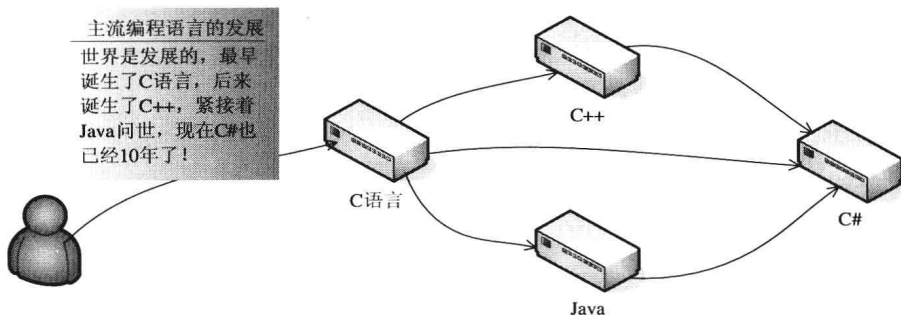


图 1-1 主编程语言的发展

1.2 漫漫发展历程

9月9日，7:10，晴

KNOWNALL 说 C 语言是编程界的少林寺，其他语言都出自 C 语言。我决定好好研究一下 C 语言的发展历程！

我：“C 语言应该很早就有了，并且至今经久不衰，所以说现在学 C 语言很有必要！”

KNOWNALL：“对，C 语言诞生于 1970 年，后来经过多次优化和改进。C 语言具有绘图能力强，可移植性，并具备很强的数据处理能力，因此适于编写系统软件，三维、二维图形和动画，它是数值计算的高级语言。所以，C 语言当之无愧地成为当今最为流行的语言之一。图 1-2 展示的是 C 语言漫漫发展历程中的几件大事！”

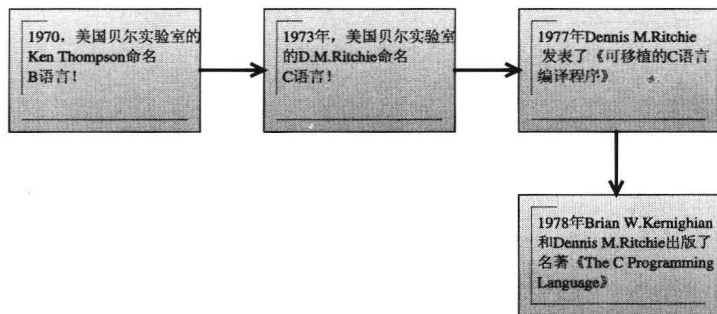


图 1-2 C 语言大事记

1970 年，美国贝尔实验室的 Ken Thompson 将 BCPL 进行了修改，并将其命名为“B 语言”，并且他用 B 语言写了第一个 UNIX 操作系统。

1973 年，贝尔实验室的 D.M.Ritchie 在 B 语言的基础上设计出了一种新的语言，并将其命名为 C 语言。

1977 年，为了使 UNIX 操作系统推广，D.M.Ritchie 发表了不依赖于具体机器系统的 C 语言编译文本《可移植的 C 语言编译程序》。

1978 年，Brian W.Kernighian 和 Dennis M.Ritchie 出版了著名的《The C Programming Language》，从而使 C 语言成为目前世界上流行最广泛的高级程序设计语言。

1988 年，随着微型计算机的日益普及，出现了许多 C 语言版本。由于没有统一的标准，使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况，美国国家标准研究所（ANSI）为 C 语言制定了一套 ANSI 标准，成为现行的 C 语言标准。

1.3 C 语言特点

9月9日，7:30，晴

我：“C 语言应该是一种汇编语言，而高级语言的特点是面向对象，那 C 语言算是高级

语言还是……？”

KNOWNALL：“C语言既有高级语言的特点，也有汇编语言的特点。它既可以编写不依赖于计算机的硬件程序，也可以编写各种计算机处理程序。总结一下，C语言的主要特点如图1-3所示。”

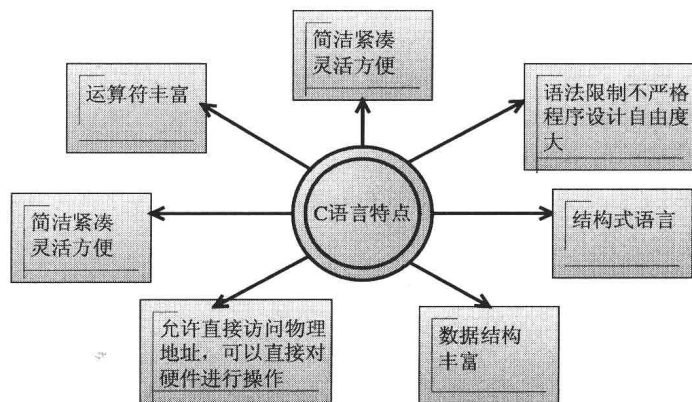


图 1-3 C语言特点

1.4 工欲善其事，必先利其器

9月9日，7:40，晴

我：“‘工欲善其事，必先利其器’是孔子名言，意思是：孔子告诉子贡，一个做手工或工艺的人，要想把工作完成，做得完善，应该先把工具准备好。”

KNOWNALL：“不错，没想到你对历史还有研究！”

我：“把这个典故放在此处有何用意呢？”

KNOWNALL：“意思是说，要做好C语言开发，有一个好的开发工具很重要！”

我：“编写C语言用哪个工具最好？”

KNOWN：“C语言的调试工具叫做编译器，它是能够编译、运行C语言程序的一种工具。”

目前最流行的C语言编译器有如下几种：

- GNU Compiler Collection 或称 GCC
- Microsoft C 或称 MS C
- Borland Turbo C 或称 Turbo C
- DEV-C++

上述C语言编译器版本不仅实现了ANSI C标准，而且在此基础上各自进行了一些扩充，使之更加方便、更加完美。在上述编译器中，其中最为常用的是Turbo C和DEV-C++。