



C 语言 程序设计

慕课版

明日科技·出品 © 徐国华 王瑶 侯小毛 主编 © 王春红 马润 陈黎黎 王秀友 副主编

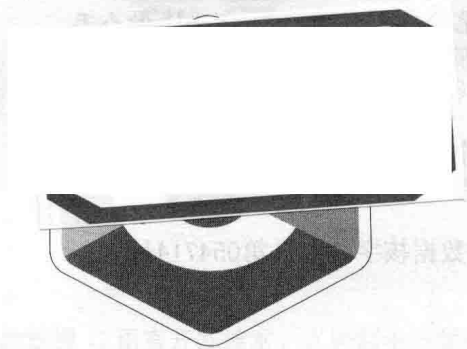
课程简介 第一本程序设计互联网教材，买书送名师，面对面授课

体系完整的 60 学时慕课教学视频同步讲解，即学即练的在线测试
书中 220 个微课视频，扫一扫二维码，随时随地学习

300 余道习题和自测题，附赠 PPT 课件、案例源代码、自测试卷、
拓展案例等资源

适用对象 学习 C 语言的第一本书





C语言 程序设计

慕课版

明日科技·出品

◎ 徐国华 王瑶 侯小毛 主编 ◎ 王春红 马润 陈黎黎 王秀友 副主编

人民邮电出版社
北京

图书在版编目 (C I P) 数据

C语言程序设计：慕课版 / 徐国华, 王瑶, 侯小毛
主编. — 北京：人民邮电出版社, 2017.6
ISBN 978-7-115-45026-5

I. ①C… II. ①徐… ②王… ③侯… III. ①C语言—
程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆CIP数据核字(2017)第054714号

内 容 提 要

本书作为C语言程序设计的教程,系统全面地介绍了有关C语言程序设计开发所涉及的各类知识。全书共分17章,内容包括C语言概述、算法、数据类型、运算符与表达式、常用的数据输入/输出函数、选择结构程序设计、循环控制、数组、函数、指针、结构体和共用体、位运算、预处理、文件、存储管理、网络套接字编程、综合实例——学生信息管理系统。全书每章内容都与实例紧密结合,有助于学生理解知识、应用知识,达到学以致用目的。

本书为慕课版教材,各章节主要内容配备了以二维码为载体的微课,并在人邮学院(www.rymoo.com)平台上提供了慕课。此外,本书还提供了课程资源包。资源包中提供了本书所有实例、上机指导、综合案例的源代码、制作精良的电子课件PPT、重点及难点教学视频、自测题库(包括选择题、填空题、操作题题库及自测试卷等内容),以及拓展综合案例和拓展实验。其中,源代码全部经过精心测试,能够在Windows XP、Windows 7系统下编译和运行。

-
- ◆ 主 编 徐国华 王 瑶 侯小毛
 - 副 主 编 王春红 马 润 陈黎黎 王秀友
 - 责任编辑 刘 博
 - 责任印制 杨林杰
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 24.25 2017年6月第1版
字数: 729千字 2017年6月河北第1次印刷

定价: 49.80 元

读者服务热线: (010) 81055256 印装质量热线: (010) 81055316

反盗版热线: (010) 81055315

广告经营许可证: 京工商广登字 20170147 号

前言

Foreword

为了让读者能够快速且牢固地掌握 C 语言开发技术，人民邮电出版社充分发挥在线教育方面的技术优势、内容优势、人才优势，潜心研究，为读者提供一种“纸质图书+在线课程”相配套，全方位学习 C 语言开发的解决方案。读者可根据个人需求，利用图书和“人邮学院”平台上的在线课程进行系统化、移动化的学习，以便快速全面地掌握 C 语言开发技术。

一、如何学习慕课版课程

本课程依托人民邮电出版社自主开发的在线教育慕课平台——人邮学院（www.rymoo.com），该平台为学习者提供优质、海量的课程，课程结构严谨，用户可以根据自身的学习程度，自主安排学习进度，并且平台具有完备的在线“学习、笔记、讨论、测验”功能。人邮学院为每一位学习者，提供完善的一站式学习服务（见图 1）。



图 1 人邮学院首页

为了使读者更好地完成慕课的学习，现将本课程的使用方法介绍如下。

1. 用户购买本书后，找到粘贴在书封底上的刮刮卡，刮开，获得激活码（见图 2）。
2. 登录人邮学院网站（www.rymoo.com），或扫描封面上的二维码，使用手机号码完成网站注册（见图 3）。



图 2 激活码



图 3 注册人邮学院网站

3. 注册完成后，返回网站首页，单击页面右上角的“学习卡”选项（见图 4），进入“学习卡”页面

(见图 5)，输入激活码，即可获得该慕课课程的学习权限。



图 4 单击“学习卡”选项

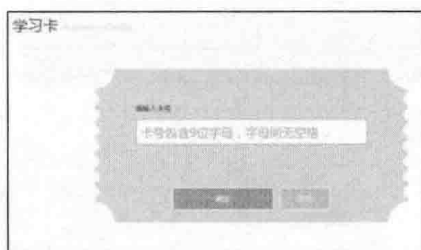


图 5 在“学习卡”页面输入激活码

4. 输入激活码后，即可获得该课程的学习权限。可随时随地使用计算机、平板电脑、手机学习本课程的任意章节，根据自身情况自主安排学习进度（见图 6）。

5. 在学习慕课课程的同时，阅读本书中相关章节的内容，巩固所学知识。本书既可与慕课课程配合使用，也可单独使用，书中主要章节均放置了二维码，用户扫描二维码即可在手机上观看相应章节的视频讲解。

6. 学完一章内容后，可通过精心设计的在线测试题，查看知识掌握程度（见图 7）。

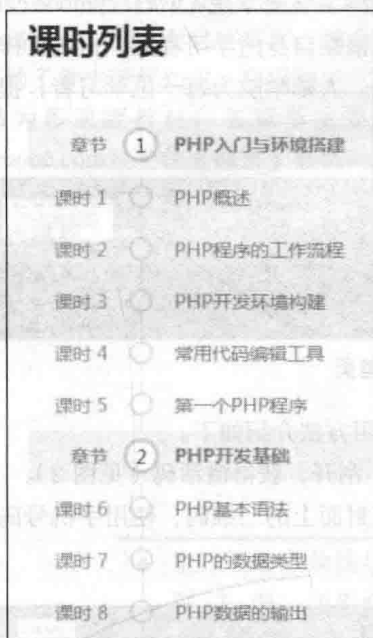


图 6 课时列表

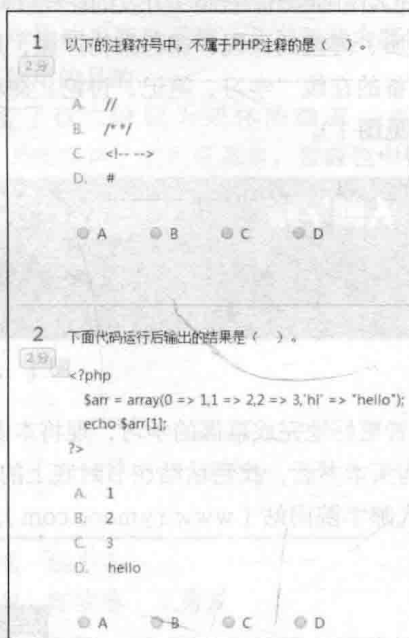


图 7 在线测试题

7. 如果对所学内容有疑问，还可到讨论区提问，除了有大牛导师答疑解惑以外，同学之间也可互相交流学习心得（见图 8）。

8. 书中配套的 PPT、源代码等教学资源，用户也可在该课程的首页找到相应的下载链接（见图 9）。

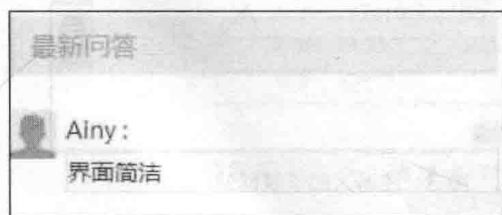


图 8 讨论区

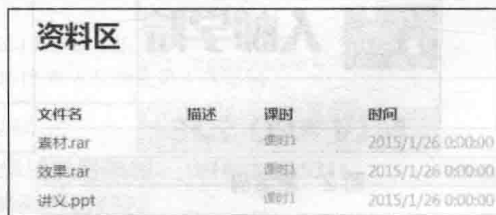


图 9 配套资源

关于人邮学院平台使用的任何疑问,可登录人邮学院咨询在线客服,或致电:010-81055236。

二、本书特点

C语言是 Combined Language (组合语言)的简称,它作为一种计算机设计语言,具有高级语言和汇编语言的特点,受到广大编程人员的喜爱。C语言的应用非常广泛,既可以用于编写系统应用程序,也可以作为编写应用程序的设计语言,还可以具体应用到有关单片机以及嵌入式系统的开发。这就是大多数学习者学习编写程序都选择C语言的原因。

在当前的教育体系下,实例教学是计算机语言教学的最有效的方法之一。本书将C语言知识和实用的实例有机结合起来,一方面,跟踪C语言的发展,适应市场需求,精心选择内容,突出重点、强调实用,使知识讲解全面、系统;另一方面,全书按照“案例贯穿”的形式组织内容,始终围绕最后的综合案例设计实例,将实例融入到知识讲解中,使知识与案例相辅相成,既有利于读者学习知识,又有利于指导读者实践。另外,本书在每一章的后面还提供了上机指导和习题,方便读者及时验证自己的学习效果(包括动手实践能力和理论知识)。

本书作为教材使用时,课堂教学建议35~40学时,上机指导教学建议10~12学时。各章主要内容和学时建议分配如下,老师可以根据实际教学情况进行调整。

章	主要内容	课堂学时	上机指导
第1章	C语言概述,包括C语言的发展史、C语言的特点、一个简单的C程序、一个完整的C程序、C语言程序的格式、开发环境	1	1
第2章	算法,包括算法的基本概念、算法的描述	3	1
第3章	数据类型,包括编程规范、关键字、标识符、数据类型、常量、变量、变量的存储类别、混合运算	2	1
第4章	运算符与表达式,包括表达式、赋值运算符与赋值表达式、算术运算符与算术表达式、关系运算符与关系表达式、逻辑运算符与逻辑表达式、位逻辑运算符与位逻辑表达式、逗号运算符与逗号表达式、复合赋值运算符	5	1
第5章	常用的数据输入/输出函数,包括语句、字符数据输入/输出、字符串输入/输出、格式输出函数、格式输入函数、顺序程序设计应用	4	1
第6章	选择结构程序设计,包括if语句、if语句的基本形式、if的嵌套形式、条件运算符、switch语句、if...else语句和switch语句的区别、选择结构程序应用	3	1
第7章	循环控制,包括循环语句、while语句、do-while语句、for语句、3种循环语句的比较、循环嵌套、转移语句	2	1
第8章	数组,包括一维数组、二维数组、字符数组、多维数组、数组的排序算法、字符串处理函数、数组应用	3	1
第9章	函数,包括函数概述、函数的定义、返回语句、函数参数、函数的调用、内部函数和外部函数、局部变量和全局变量、函数应用	2	1
第10章	指针,包括指针相关概念、数组与指针、指向指针的指针、指针变量作函数参数、返回指针值的函数、指针数组作main函数的参数	3	1
第11章	结构体和共用体,包括结构体、结构体数组、结构体指针、包含结构体的结构、链表、链表相关操作、共用体、枚举类型	3	1

续表

章	主要内容	课堂学时	上机指导
第 12 章	位运算, 包括位与字节、位运算操作符、循环移位、位段	2	1
第 13 章	预处理, 包括宏定义、#include 指令、条件编译	2	1
第 14 章	文件, 包括文件概述、文件基本操作、文件的读写、文件的定位	2	1
第 15 章	存储管理, 包括内存组织方式、动态管理、内存丢失	2	1
第 16 章	网络套接字编程, 包括内存组织方式、套接字概述、套接字函数	3	1
第 17 章	综合实例——学生信息管理系统, 包括开发背景、开发环境需求、系统功能设计、预处理模块设计、主函数设计、录入学生信息、查询学生信息、删除学生信息、修改学生信息、插入学生信息、学生成绩排名、统计学生总数、显示所有学生信息		3

编者

2017 年 1 月

目录

Contents

第1章 C语言概述	1	3.3 标识符	27
1.1 C语言的发展史	2	3.4 数据类型	28
1.1.1 程序语言简述	2	3.5 常量	29
1.1.2 C语言的历史	2	3.5.1 整型常量	29
1.2 C语言的特点	3	3.5.2 实型常量	31
1.3 一个简单的C程序	3	3.5.3 字符型常量	32
1.4 一个完整的C程序	5	🔗 实例 1: 输出字符常量	
1.5 C语言程序的格式	8	🔗 实例 2: 输出字符串常量	
1.6 开发环境	9	3.5.4 转义字符	34
1.6.1 Dev C++	9	3.5.5 符号常量	35
1.6.2 Visual C++ 6.0	13	🔗 实例: 输入半径, 得到圆的面积	
小结	14	3.6 变量	36
上机指导	15	3.6.1 整型变量	36
习题	15	3.6.2 实型变量	38
第2章 算法	16	🔗 实例: 使用各种实型变量	
2.1 算法的基本概念	17	3.6.3 字符型变量	40
2.1.1 算法的特性	17	🔗 实例: 使用字符型变量	
2.1.2 算法的优劣	17	3.7 变量的存储类别	42
2.2 算法的描述	18	3.7.1 静态存储与动态存储	42
2.2.1 自然语言	18	3.7.2 auto 变量	42
🔗 实例: 用自然语言求 n!		🔗 实例: 使用 auto 变量	
2.2.2 流程图	19	3.7.3 static 变量	43
🔗 实例: 画出趣味俄罗斯方块的流程图		🔗 实例: 使用 static 变量	
2.2.3 N-S 流程图	22	3.7.4 register 变量	43
🔗 实例: 绘制 N-S 流程图求 n!		🔗 实例: 使用 register 变量修饰整型变量	
小结	24	3.7.5 extern 变量	44
上机指导	24	🔗 实例: 使用 extern 变量	
习题	24	3.8 混合运算	45
第3章 数据类型	25	🔗 实例: int、char、float 型变量相加, 得出 double 型变量进行输出	
3.1 编程规范	26	小结	46
3.2 关键字	26	上机指导	46
		习题	47

第 4 章 运算符与表达式	48		
4.1 表达式	49		
🔗 实例：求解二元一次不定方程 $ax+by=c$ 的解			
4.2 赋值运算符与赋值表达式	50		
4.2.1 变量赋初值	51		
🔗 实例：模拟钟点工的计费情况，使用赋值语句和表达式得出钟点工工作 8 个小时后所得的薪水			
4.2.2 自动类型转换	52		
4.2.3 强制类型转换	52		
🔗 实例：通过不同类型变量之间的赋值，将赋值操作后的结果进行输出			
4.3 算术运算符与算术表达式	53		
4.3.1 算术运算符	53		
🔗 实例：求 100~1000 之间满足各位数字之和是 5 的所有数			
4.3.2 算术表达式	54		
🔗 实例：使用算术表达式计算摄氏温度			
4.3.3 优先级与结合性	55		
🔗 实例：通过不同运算符的优先级和结合性，使用 printf 函数显示最终的计算结果			
4.3.4 自增/自减运算符	57		
🔗 实例：比较自增、自减运算符前缀与后缀的不同			
4.4 关系运算符与关系表达式	58		
4.4.1 关系运算符	58		
4.4.2 关系表达式	59		
4.4.3 优先级与结合性	59		
🔗 实例：使用 if 语句判断两个学科的分数大小			
4.5 逻辑运算符与逻辑表达式	61		
4.5.1 逻辑运算符	61		
4.5.2 逻辑表达式	61		
🔗 实例：判断闰年			
4.5.3 优先级与结合性	61		
🔗 实例：使用逻辑运算符构造表达式，通过输出函数显示表达式的结果			
4.6 位逻辑运算符与位逻辑表达式	62		
4.6.1 位逻辑运算符	62		
4.6.2 位逻辑表达式	63		
4.7 逗号运算符与逗号表达式	63		
🔗 实例：通过逗号运算符将其他运算符结合在一起形成表达式，再将表达式的最终结果赋值给变量			
4.8 复合赋值运算符	64		
🔗 实例：使用复合赋值运算符简化赋值运算			
小结	65		
上机指导	67		
习题	67		
第 5 章 常用的数据输入/输出函数	68		
5.1 语句	69		
5.2 字符数据输入/输出	69		
5.2.1 字符数据输出	69		
🔗 实例：使用 putchar 函数，输出字符串“Hello”			
5.2.2 字符数据输入	70		
🔗 实例 1：使用 getchar 函数获取在键盘上输入的字符，再利用 putchar 函数进行输出			
🔗 实例 2：使用 getchar 函数取消获取回车符			
5.3 字符串输入/输出	72		
5.3.1 字符串输出函数	72		
🔗 实例：使用 puts 函数对字符串常量和字符串变量都进行操作			
5.3.2 字符串输入函数	73		
🔗 实例：使用字符串输入函数 gets 获取输入信息			
5.4 格式输出函数	74		
🔗 实例：使用 printf 函数输出字符花			
5.5 格式输入函数	78		
🔗 实例：利用 scanf 函数得到用户输入的两个整型数据			
5.6 顺序程序设计应用	81		
🔗 实例：将大写字母转换成小写字母			
















小结	83
上机指导	83
习题	83
第6章 选择结构程序设计	85
6.1 if 语句	86
6.2 if 语句的基本形式	86
6.2.1 if 语句形式	86
🔗 实例: 使用 if 语句模拟信号灯指挥车辆行驶	
6.2.2 if...else 语句形式	89
🔗 实例 1: 使用 if...else 语句得到两个数的最大值	
🔗 实例 2: 使用 if...else 语句模拟信号灯	
6.2.3 else if 语句形式	92
🔗 实例: 使用 else if 语句编写菜单程序	
6.3 if 的嵌套形式	95
🔗 实例: 使用 if 嵌套语句选择日程安排	
6.4 条件运算符	97
🔗 实例: 使用条件运算符计算欠款金额	
6.5 switch 语句	99
6.5.1 switch 语句的基本形式	99
🔗 实例: 使用 switch 语句输出分数段	
6.5.2 多路开关模式的 switch 语句	103
🔗 实例: 使用 switch 语句设计欢迎界面的菜单选项	
6.6 if...else 语句和 switch 语句的区别	104
6.7 选择结构程序应用	106
🔗 实例: 使用 switch 语句计算运输公司的计费	
小结	108
上机指导	108
习题	109
第7章 循环控制	110
7.1 循环语句	111
7.2 while 语句	111
🔗 实例: 计算 1 累加到 100 的结果	
7.3 do-while 语句	112
🔗 实例: 使用 do-while 语句计算 1~	

100 之间的累加结果	
7.4 for 语句	114
7.4.1 for 语句使用	114
🔗 实例: 打印俄罗斯方块的游戏窗口	
7.4.2 for 循环的变体	117
🔗 实例: 省略 for 语句中的第一个表达式	
7.4.3 for 语句中的逗号应用	119
🔗 实例: 计算 1~100 之间所有偶数的累加结果	
7.5 3 种循环语句的比较	120
7.6 循环嵌套	120
7.6.1 循环嵌套的结构	120
7.6.2 循环嵌套实例	122
🔗 实例: 使用嵌套语句打印欢迎界面的边框	
7.7 转移语句	124
7.7.1 goto 语句	124
🔗 实例: 使用 goto 语句从循环内部跳出	
7.7.2 break 语句	125
🔗 实例: 使用 break 语句跳出循环	
7.7.3 continue 语句	126
🔗 实例: 使用 continue 语句结束本次的循环操作	
小结	127
上机指导	128
习题	128


第8章 数组	129
8.1 一维数组	130
8.1.1 一维数组的定义和引用	130
🔗 实例: 使用数组保存数据	
8.1.2 一维数组初始化	131
🔗 实例: 初始化一维数组	
8.1.3 一维数组应用	134
🔗 实例: 使用数组保存学生姓名	
8.2 二维数组	135
8.2.1 二维数组的定义和引用	135
8.2.2 二维数组初始化	136
🔗 实例: 使用二维数组标记游戏屏幕上的图案	


8.2.3 二维数组应用	138
 实例：打印游戏窗口，并设置左右下横框上有图案	
8.3 字符数组	141
8.3.1 字符数组的定义和引用	141
8.3.2 字符数组初始化	141
 实例：使用二维字符数组输出一个钻石形状	
8.3.3 字符数组的结束标志	143
8.3.4 字符数组的输入和输出	144
 实例：使用两种方式输出字符串	
8.3.5 字符数组应用	145
 实例：计算字符串中单词的个数	
8.4 多维数组	146
8.5 数组的排序算法	146
8.5.1 选择法排序	146
8.5.2 冒泡法排序	148
8.5.3 交换法排序	150
 实例：实现学生信息管理系统中的学生成绩排名功能	
8.5.4 插入法排序	152
8.5.5 折半法排序	154
8.5.6 排序算法的比较	156
8.6 字符串处理函数	157
8.6.1 字符串复制	157
 实例：字符串复制应用	
8.6.2 字符串连接	158
 实例：字符串连接应用	
8.6.3 字符串比较	160
 实例：字符串比较应用	
8.6.4 字符串大小写转换	161
 实例：字符串大小写转换	
8.6.5 获得字符串长度	163
 实例：获得字符串长度	
8.7 数组应用	164
8.7.1 反转输出字符串	164
8.7.2 输出系统日期和时间	165
8.7.3 字符串的加密和解密	166
小结	168
上机指导	168
习题	169


第9章 函数 170

9.1 函数概述	171
 实例：在主函数中调用其他函数	
9.2 函数的定义	172
9.2.1 函数定义的形式	173
9.2.2 定义与声明	174
 实例：定义获取屏幕光标位置和设置文字颜色函数	
9.3 返回语句	177
9.3.1 从函数返回	177
 实例：从函数返回应用	
9.3.2 返回值	178
 实例：返回值类型与 return 值类型	
9.4 函数参数	180
9.4.1 形式参数与实际参数	180
 实例：形式参数与实际参数的比喻实现	
9.4.2 数组作函数参数	182
 实例：数组元素作为函数参数	
9.4.3 main 函数的参数	187
 实例：main 函数的参数使用	
9.5 函数的调用	188
9.5.1 函数的调用方式	188
 实例：调用获取屏幕光标位置和设置文字颜色函数，来设置趣味俄罗斯方块的标题图	
9.5.2 嵌套调用	193
 实例：函数的嵌套调用	
9.5.3 递归调用	195
 实例：函数的递归调用	
9.6 内部函数和外部函数	197
9.6.1 内部函数	197
 实例：内部函数的使用	
9.6.2 外部函数	198
 实例：外部函数的使用	
9.7 局部变量和全局变量	200
9.7.1 局部变量	200
 实例 1：局部变量的作用域	
 实例 2：局部变量的屏蔽作用	
9.7.2 全局变量	203
 实例：使用全局变量模拟价格调整	

9.8 函数应用 204

 实例 1: 数学库函数使用

 实例 2: 使用三角函数

 实例 3: 使用字符函数判断输入字符

小结 210

上机指导 210

习题 211


第 10 章 指针 212

10.1 指针相关概念 213


10.1.1 地址与指针 213

10.1.2 变量与指针 214

10.1.3 指针变量 214


 实例: 从键盘中输入两个数, 利用指针的方法将这两个数输出

10.1.4 指针自加自减运算 218


 实例: 整型变量地址输出

10.2 数组与指针 219


10.2.1 一维数组与指针 219

 实例: 输出数组中的元素

10.2.2 二维数组与指针 223

 实例: 利用指针对二维数组进行输入和输出


10.2.3 字符串与指针 226


 实例: 输入两个字符串 a 和 b, 将字符串 a 和 b 连接起来

10.2.4 字符串数组 228


 实例: 输出 12 个月


10.3 指向指针的指针 229


 实例 1: 使用指向指针的指针输出 12 个月

 实例 2: 利用指向指针的指针输出一维数组中是偶数的元素, 并统计偶数的个数


10.4 指针变量作函数参数 232

 实例 1: 调用自定义函数交换两变量值


 实例 2: 嵌套的函数调用

 实例 3: 任意输入 10 个数据, 先将这 10 个数据中是奇数的数据输出, 再求这 10 个数据中所有奇数之和

10.5 返回指针值的函数 240

 实例: 使用返回指针的函数查找最大值

10.6 指针数组作 main 函数的参数 242

 实例: 输出 main 函数的参数内容

小结 243

上机指导 243

习题 244


第 11 章 结构体和共用体 245

11.1 结构体 246


11.1.1 结构体类型的概念 246

11.1.2 结构体变量的定义 247

11.1.3 结构体变量的引用 248

 实例: 引用学生成绩结构体


11.1.4 结构体类型的初始化 250

 实例: 声明结构体类型表示商品, 然后定义结构体变量, 之后对变量中的成员进行赋值, 最后将结构体变量中保存的信息进行输出

11.2 结构体数组 251


11.2.1 定义结构体数组 251

11.2.2 初始化结构体数组 253


 实例: 初始化学生信息结构体数组, 并输出学生信息

11.3 结构体指针 254


11.3.1 指向结构体变量的指针 255

 实例: 通过指针使用点运算符引用结构体变量的成员


11.3.2 指向结构体数组的指针 257

 实例: 根据学生结构类型定义结构体数组, 并对其进行初始化操作

11.3.3 结构体作为函数参数 259

 实例: 声明一个简单的结构类型表示学生成绩, 编写一个函数, 使得该结构类型变量作为函数的参数

11.4 包含结构的结构 262

 实例: 定义两个结构体类型, 一个表示日期, 一个表示学生的个人信息

11.5	链表	263
11.5.1	链表概述	263
11.5.2	创建动态链表	264
11.5.3	输出链表	266
	实例: 编写一个包含学生信息的链表结构, 并且将链表中的信息进行输出	
11.6	链表相关操作	268
11.6.1	链表的插入操作	269
11.6.2	链表的删除操作	270
	实例: 对学生信息链表进行删除操作	
11.7	共用体	274
11.7.1	共用体的概念	274
11.7.2	共用体变量的引用	274
	实例: 定义共用体变量, 通过定义的显示函数, 引用共用体中的数据成员	
11.7.3	共用体变量的初始化	275
	实例: 定义共用体变量的同时进行初始化操作, 并将引用变量的值输出	
11.7.4	共用体类型的数据特点	276
11.8	枚举类型	276
	实例: 使用枚举类型	
	小结	278
	上机指导	278
	习题	279

第12章 位运算 280

12.1	位与字节	281
12.2	位运算操作符	281
12.2.1	“与”运算符	281
	实例: 任意输入两个数分别赋给 a 和 b, 计算 a&b 的值	
12.2.2	“或”运算符	283
	实例: 任意输入两个数分别赋给 a 和 b, 计算 a b 的值	
12.2.3	“取反”运算符	284
	实例: 输入一个数赋给变量 a, 计算~a 的值	
12.2.4	“异或”运算符	285
	实例: 输入两个数分别赋给变量 a	

	和 b, 计算 a^b 的值	
12.2.5	“左移”运算符	287
	实例: 将 15 先左移两位, 将其左移后的结果输出, 在这个结果的基础上再左移 3 位, 并将结果输出	
12.2.6	“右移”运算符	288
	实例: 将 30 和 -30 分别右移 3 位, 将所得结果分别输出, 在所得结果的基础上再分别右移两位, 并将结果输出	
12.3	循环移位	289
	实例: 从键盘中输入一个八进制数, 然后输入要移位的位数, 最后将移位的结果显示在屏幕上	
12.4	位段	292
12.4.1	位段的概念与定义	292
12.4.2	位段相关说明	293
	小结	294
	上机指导	294
	习题	295

第13章 预处理 296

13.1	宏定义	297
13.1.1	不带参数的宏定义	297
	实例: 设计趣味俄罗斯方块	
13.1.2	带参数的宏定义	298
	实例: 对两个数实现乘法加法混合运算	
13.2	#include 指令	299
	实例: 文件包含应用	
13.3	条件编译	301
13.3.1	#if 命令	301
	实例: #if 应用	
13.3.2	#ifdef 及 #ifndef 命令	304
	实例: #ifdef 和 #ifndef 的具体应用	
13.3.3	#undef 命令	305
13.3.4	#line 命令	305
	实例: 输出行号	
13.3.5	#pragma 命令	306
	小结	306
	上机指导	307
	习题	307

第 14 章 文件	308
14.1 文件概述	309
14.2 文件基本操作	309
14.2.1 文件指针	309
14.2.2 文件的打开	310
14.2.3 文件的关闭	310
14.3 文件的读写	311
14.3.1 fputc 函数	311
🔗 实例：向 E:\exp01.txt 中写入 “forever...forever...”	
14.3.2 fgetc 函数	312
🔗 实例：在屏幕中显示出 exp02.txt 中 的内容	
14.3.3 fputs 函数	312
🔗 实例：向指定的磁盘文件中写入字 符串 “gone with the wind”	
14.3.4 fgets 函数	313
🔗 实例：读取任意磁盘文件中的内容	
14.3.5 fprintf 函数	314
🔗 实例：将数字 88 以字符的形式写 到磁盘文件中	
14.3.6 fscanf 函数	315
🔗 实例：将文件中的 5 个字符以整数 形式输出	
14.3.7 fread 和 fwrite 函数	316
🔗 实例：将录入的通讯录信息保存到 磁盘文件中，并显示	
14.4 文件的定位	318
14.4.1 fseek 函数	318
🔗 实例：向任意一个二进制文件中写 入一个长度大于 6 的字符 串，然后从该字符串的第 6 个字符开始输出余下字符	
14.4.2 rewind 函数	320
🔗 实例：rewind 函数的应用	
14.4.3 ftell 函数	321
🔗 实例：求字符串长度	
小结	322
上机指导	323
习题	323

第 15 章 存储管理	324
15.1 内存组织方式	325
15.1.1 内存的组织方式	325
15.1.2 堆与栈	325
🔗 实例：在堆中分配内存并释放	
15.2 动态管理	327
15.2.1 malloc 函数	327
🔗 实例：使用 malloc 函数动态分配空间	
15.2.2 calloc 函数	328
🔗 实例：使用 calloc 函数分配数组内存	
15.2.3 realloc 函数	329
🔗 实例：使用 realloc 函数重新分配内存	
15.2.4 free 函数	330
🔗 实例：使用 free 函数释放内存空间	
15.3 内存丢失	331
小结	332
上机指导	332
习题	333
第 16 章 网络套接字编程	334
16.1 内存组织方式	335
16.1.1 IP 地址	335
16.1.2 OSI 七层参考模型	335
16.1.3 地址解析	336
16.1.4 域名系统	337
16.1.5 TCP/IP	337
16.1.6 端口	339
16.1.7 套接字的引入	339
16.1.8 网络字节顺序	339
16.2 套接字概述	339
16.2.1 套接字概述	340
16.2.2 TCP 的套接字的 socket 编程	340
16.2.3 UDP 的套接字的 socket 编程	340
16.3 套接字函数	341
16.3.1 套接字函数介绍	341
16.3.2 基于 TCP 的网络聊天程序	345
🔗 实例：设计网络聊天服务器端	
小结	351
上机指导	352
习题	352

第17章 综合实例——学生信息管理系统 353

17.1 开发背景	354
17.2 开发环境需求	354
17.3 系统功能设计	354
17.4 预处理模块设计	354
17.4.1 模块概述	354
17.4.2 控制输出格式	355
17.4.3 文件引用	355
17.4.4 宏定义	355
17.4.5 函数声明	355
17.5 主函数设计	356
17.5.1 功能概述	356
17.5.2 实现主函数	356
17.5.3 显示主菜单	358
17.6 录入学生信息	358
17.6.1 模块概述	358
17.6.2 实现文件的打开和关闭功能	359
17.6.3 实现录入学生信息	360
17.7 查询学生信息	362
17.7.1 模块概述	362
17.7.2 查询没有记录的文件	363

17.7.3 查找并打印学生信息	363
17.8 删除学生信息	364
17.8.1 模块概述	364
17.8.2 实现删除学生信息	364
17.9 修改学生信息	366
17.9.1 模块概述	366
17.9.2 实现修改学生信息	367
17.10 插入学生信息	368
17.10.1 模块概述	368
17.10.2 实现插入学生信息	368
17.11 学生成绩排名	370
17.11.1 模块概述	370
17.11.2 使用交换排序法实现排序功能	371
17.11.3 实现学生成绩排名	371
17.12 统计学生总数	372
17.12.1 模块概述	372
17.12.2 实现统计学生总数	372
17.13 显示所有学生信息	373
17.13.1 模块概述	373
17.13.2 读取并显示所有学生信息	374
小结	374

第1章

C语言概述

本章要点：

■ 在学习 C 语言之前，每一个刚刚学习 C 语言的人员都应该清楚地了解 C 语言的发展历程，了解为什么要选择 C 语言，以及它有哪些特性。只有了解了 C 语言的历史和特性，才会更深刻地了解这门语言，并且增加今后学习 C 语言的信心。随着计算机科学的不断发展，C 语言的学习环境也在不断变化，刚开始学习 C 语言时，大多数人会选择一些相对简单的编译器，如 Turbo C 2.0。但是，现在越来越多的人开始选择 Dev C++ 编译器或是 Visual C++ 6.0 编译器。

■ 本章致力于使读者了解 Dev C++ 的开发环境，掌握其中各个部分的使用方法，并能编写一个简单的应用程序以练习使用开发环境。

- 了解 C 语言的发展史
- 了解 C 语言的特点
- 了解 C 语言的组织结构
- 掌握如何使用 Turbo C 2.0 开发 C 程序
- 掌握如何使用 Visual C++ 6.0 开发 C 程序

1.1 C 语言的发展史

1.1.1 程序语言简述

在介绍 C 语言的发展历程之前，先对程序语言进行大概了解。

1. 机器语言

机器语言是低级语言，也称为二进制代码语言。计算机使用的是由 0 和 1 组成的二进制数组成的一串指令来表达计算机操作的语言。机器语言的特点是，计算机可以直接识别，不需要进行任何的翻译。

2. 汇编语言

汇编语言是面向机器的程序设计语言。为了减轻程序员使用机器语言编程的痛苦，用英文字母或符号串来替代机器语言的二进制码，这样就把不易理解和使用的机器语言变成了汇编语言。使用汇编语言比使用机器语言更便于阅读和理解程序。

3. 高级语言

由于汇编语言依赖于硬件体系，并且该语言中的助记符号数量比较多，所以其运用起来仍然不够方便。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，于是产生了高级语言。这种语言，其语法形式类似于英文，并且因为远离对硬件的直接操作，而易于被普通人所理解与使用。其中影响较大、使用普遍的高级语言有 Fortran、ALGOL、Basic、COBOL、LISP、Pascal、PROLOG、C、C++、Delphi、Java 等。



程序语言简述

1.1.2 C 语言的历史

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，又不失低级语言的优点。于是，C 语言产生了。

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 (简称 B 语言) 的基础上发展和完善起来的。19 世纪 70 年代初期，AT&T Bell 实验室的程序员丹尼斯·里奇第一次把 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，丹尼斯·里奇用 C 语言改写了 UNIX C 的编译程序，UNIX 操作系统的开发者肯·汤普逊又用 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 于 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 并未得到广泛使用。

尽管 C 语言是在大型商业机构和学术界的研究实验室研发的，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，并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。



C 语言的历史