

《C语言程序设计教程》

图书基本信息

书名：《C语言程序设计教程》

13位ISBN编号：9787564117498

10位ISBN编号：7564117494

出版时间：2009-8-1

出版社：东南大学出版社

作者：张黎宁,薛联凤,何煦岚,等

页数：248

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《C语言程序设计教程》

内容概要

《C语言程序设计教程》是根据教育部高等学校计算机科学与技术教学指导委员会提出的《关于进一步加强高等学校计算机基础教学的意见暨计算机基础课程教学基本要求》的有关要求编写的。本课程是一门实践性很强的课程，《C语言程序设计教程》力求使学生掌握计算机程序设计语言的基本知识、具备程序设计能力和程序调试能力，为后续课程学习创造条件。《C语言程序设计教程》体系合理，案例丰富，可作为高等学校各类专业的程序设计语言教材，也便于初学者自学。

书籍目录

1 引言	1.1 程序及编程语言	1.2 C语言简介	1.2.1 C语言的发展	1.2.2 C语言的特点	1.3 C语言源程序的结构	1.4 准备开始编程	1.4.1 在Turbo C 2.0集成开发环境中编辑、调试程序的过程	1.4.2 在Visual C++6.0集成开发环境中编辑、调试程序的过程	1.5 如何学习本课程	习题一	2 简单的C语言程序设计	2.1 程序设计与算法	2.1.1 程序	2.1.2 算法的概念和特点	2.1.3 结构化程序设计	2.2 C语言程序的基本组成结构	2.2.1 简单的C语言程序举例	2.2.2 C语言程序的组成结构	2.3 C语言程序设计的主要过程	2.3.1 问题分析与算法描述	2.3.2 程序的编辑	2.3.3 程序的编译与运行	习题二	3 数据类型、运算符和表达式	3.1 C语言的数据类型	3.1.1 数据类型的分类	3.1.2 标识符	3.1.3 关键字	3.2 常量与变量	3.2.1 常量	3.2.2 变量	3.3 整型数据	3.3.1 整型常量	3.3.2 整型变量	3.4 实型数据	3.4.1 实型常量	3.4.2 实型变量	3.5 字符数据	3.5.1 字符常量	3.5.2 字符变量	3.5.3 字符串常量	3.6 各种类型数据之间的混合运算	3.7 C语言的运算符与表达式	3.7.1 C语言的运算符简介	3.7.2 算术运算符和算术表达式	3.7.3 赋值运算符和赋值表达式	3.7.4 逗号运算符和逗号表达式	习题三	4 基本输入输出	4.1 C语言的语句	4.2 数据输入输出的概念以及在C语言中的实现	4.3 字符数据的输入输出	4.3.1 putchar函数(字符输出函数)	4.3.2 getchar函数(键盘输入函数)	4.4 格式输入与输出	4.4.1 printf函数(格式输出函数)	4.4.2 scanf函数(格式输入函数)	习题四	5 选择结构	5.1 关系运算符和关系表达式	5.2 逻辑运算符和逻辑表达式	5.3 if语句	5.3.1 最基本的if语句	5.3.2 if—else结构	5.3.3 条件运算符和条件表达式	5.3.4 if—else if结构	5.3.5 if语句的嵌套	5.4 switch语句	习题五	6 循环结构程序设计	6.1 简介	6.2 while语句	6.3 do—while语句	6.4 for语句	6.5 goto语句	6.6 循环的嵌套	6.7 循环语句的比较	6.8 循环控制语句	6.8.1 break语句	6.8.2 continue语句	6.9 程序应用	习题六	7 数组	7.1 一维数组	7.1.1 一维数组的定义	7.1.2 一维数组的初始化	7.1.3 一维数组举例	7.2 二维数组	7.2.1 二维数组的定义	7.2.2 二维数组的初始化	7.2.3 二维数组举例	7.3 字符数组和字符串	7.3.1 字符数组的定义	7.3.2 字符数组的初始化	7.3.3 字符串常用函数	7.3.4 常用字符函数	7.3.5 字符数组举例	习题七	8 函数	8.1 函数概述	8.1.1 函数的分类	8.1.2 函数的定义	8.2 函数的参数和返回值	8.2.1 函数的参数	8.2.2 函数返回值	8.3 函数参数的传递方式	8.3.1 值传递	8.3.2 地址传递	8.4 函数调用	8.4.1 函数调用的一般形式	8.4.2 函数调用的方式	8.4.3 函数的嵌套调用	8.5 变量的作用域与变量的存储类别	8.5.1 变量的作用域	8.5.2 变量的存储类别	8.6 函数作用范围	8.6.1 内部函数	8.6.2 外部函数	习题八	9 预处理命令	9.1 概述	9.2 宏定义	9.2.1 无参宏定义	9.2.2 带参宏定义	9.3 文件包含	9.4 小结	习题九	10 指针	10.1 地址和指针	10.1.1 地址、指针和指针变量的概念	10.1.2 直接访问和间接访问	10.2 变量的指针与指针变量	10.2.1 定义一个指针变量	10.2.2 指针变量的初始化和赋值	10.2.3 指针变量的引用	10.2.4 指针的运算	10.3 数组与指针	10.3.1 数组名是指针常量	10.3.2 指向数组元素的指针变量	10.3.3 通过指针引用数组元素	10.3.4 多维数组与指针	10.4 字符串与指针	10.4.1 字符串处理方法	10.4.2 使用字符指针变量和使用字符数组处理字符串的比较	10.5 函数与指针	10.5.1 指针变量作为函数的参数	10.5.2 数组名与指向数组的指针变量作为函数参数	10.5.3 字符指针作为函数参数	10.5.4 函数指针的基本概念	10.5.5 用指向函数的指针作为函数参数	10.5.6 返回指针值的函数	10.6 指针数组和多级指针	10.6.1 指针数组的概念	10.6.2 多级指针	10.6.3 指针数组作main函数的形参	10.7 动态存储分配与void指针类型	10.7.1 void指针类型	10.7.2 动态内存分配函数	10.8 小结	习题十	11 结构体与共用体	11.1 结构体类型概述	11.1.1 结构体类型的定义	11.1.2 结构体变量的定义	11.1.3 结构体变量的引用	11.1.4 结构体变量的赋值和初始化	11.2 结构体数组	11.3 指向结构体类型数据的指针	11.3.1 结构体指针变量的定义	11.3.2 利用结构体指针变量引用结构体成员	11.3.3 用结构体变量和指向结构体的指针变量作函数参数	11.4 用结构体处理链表	11.4.1 链表概述	11.4.2 链表的建立与输出	11.4.3 链表的插入	11.4.4 链表的删除	11.4.5 链表的综合操作	11.5 共用体	11.5.1 共用体类型及变量的定义	11.5.2 共用体变量的使用	11.6 枚举类型	11.7 用typedef定义类型	习题十一	12 位运算	12.1 位运算符	12.2 位域	习题十二	13 文件	13.1 文件的概念	13.1.1 文件	13.1.2 文件指针	13.1.3 文件结束的判定	13.2 文件的打开与关闭	13.2.1 文件的打开	13.2.2 文件的使用方式	13.2.3 文件关闭函数	13.3 文件的读写	13.3.1 字符的读写	13.3.2 字符串的读写	13.3.4 格式化读/写文件函数(fread函数和fwrite函数)	13.4 文件的定位	13.5 文件检测函数	13.6 小结	综合训练(一)	综合训练(二)	综合训练(三)	综合训练(四)	附录	运算符和结合性	参考文献
------	-------------	-----------	--------------	--------------	---------------	------------	-------------------------------------	---------------------------------------	-------------	-----	--------------	-------------	----------	----------------	---------------	------------------	------------------	------------------	------------------	-----------------	-------------	----------------	-----	----------------	--------------	---------------	-----------	-----------	-----------	----------	----------	----------	------------	------------	----------	------------	------------	----------	------------	------------	-------------	-------------------	-----------------	-----------------	-------------------	-------------------	-------------------	-----	----------	------------	-------------------------	---------------	-------------------------	-------------------------	-------------	------------------------	-----------------------	-----	--------	-----------------	-----------------	----------	----------------	-----------------	-------------------	--------------------	---------------	--------------	-----	------------	--------	-------------	----------------	-----------	------------	-----------	-------------	------------	---------------	------------------	----------	-----	------	----------	---------------	----------------	--------------	----------	---------------	----------------	--------------	--------------	---------------	----------------	---------------	--------------	--------------	-----	------	----------	-------------	-------------	---------------	-------------	-------------	---------------	-----------	------------	----------	-----------------	---------------	---------------	--------------------	--------------	---------------	------------	------------	------------	-----	---------	--------	---------	-------------	-------------	----------	--------	-----	-------	------------	----------------------	------------------	-----------------	-----------------	--------------------	----------------	--------------	------------	-----------------	--------------------	-------------------	----------------	-------------	----------------	--------------------------------	------------	--------------------	----------------------------	-------------------	------------------	-----------------------	-----------------	----------------	----------------	-------------	-----------------------	----------------------	-----------------	-----------------	---------	-----	------------	--------------	-----------------	-----------------	-----------------	---------------------	------------	-------------------	-------------------	-------------------------	-------------------------------	---------------	-------------	-----------------	--------------	--------------	----------------	----------	--------------------	-----------------	-----------	-------------------	------	--------	-----------	---------	------	-------	------------	-----------	-------------	----------------	---------------	--------------	----------------	---------------	------------	--------------	---------------	-------------------------------------	------------	-------------	---------	---------	---------	---------	---------	----	---------	------

1 引言 1.1 程序及编程语言 对于初涉编程领域的人来说，程序是一个高深莫测的词汇。那么，程序究竟是什么呢？通俗地讲，程序就是向计算机发出的一个个操作命令集合，来告诉计算机如何完成一个具体的任务。由于现在的计算机还不能理解人类的自然语言，所以目前还不能用自然语言来编写程序。专业地讲，程序（program）是为解决特定问题而用计算机语言编写的命令序列集合。执行程序时，由程序控制计算机来完成相应的操作而得到相应的结果。要学习编程技术，首先要掌握一种计算机编程语言。程序设计语言（Programming Language）是用于编写计算机程序的语言。计算机编程语言按照级别可以分为机器语言（Machine Language）、汇编语言（Assembly Language）和高级语言（High Level Language）三类，而机器语言和汇编语言又被统称为低级语言（Low Level Language）。机器语言是用二进制代码表示的计算机能直接识别和执行的一种机器指令集合，它是第一代计算机语言。机器语言的指令全部是由0和1组成的，可想而知，利用机器语言编写的程序对于我们来说简直就是“一串密码”，程序的可读性非常差。同时，机器语言对不同型号的计算机来说一般是不同的，所以利用机器语言编写的程序的可移植性差，重用性差，这也是早期的计算机未能迅速得到广泛应用的原因之一。为了克服机器语言难读、难编、难记和易出错的缺点，人们就用与指令代码实际含义相近的英文缩写词、字母和数字等符号来取代机器指令代码，于是就产生了汇编语言。汇编语言是机器语言符号化的结果，比机器语言易于读写、调试和修改，同时具有机器语言执行效率高、可有效访问控制硬件等优点。由于汇编语言中使用了助记符号，用汇编语言编写的程序不能被计算机直接识别和执行，必须“翻译”成能被计算机识别和处理的二进制代码程序，这个过程称为汇编。用汇编语言编写的程序称为汇编语言源程序，运行时要使用汇编程序将其翻译成目标程序，即机器语言程序。

《C语言程序设计教程》

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com