

## <<C程序设计>>

### 图书基本信息

书名：<<C程序设计>>

13位ISBN编号：9787040301533

10位ISBN编号：7040301539

出版时间：2010-8

出版时间：高等教育出版社

作者：赵山林 编

页数：348

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：<http://www.tushu007.com>

## &lt;&lt;C程序设计&gt;&gt;

## 前言

教育部高等学校计算机基础课程教学指导委员会在《高等学校计算机基础教学发展战略研究报告暨计算机基础课程教学基本要求》(以下简称《基本要求》)中指出,党的十七大提出了我国要从改造传统工业入手,走工业与信息技术相融合的新型工业化道路。

为此,需要培养大批新一代“专业+信息”的工程技术人才。

作为信息技术的核心,计算机基础教育的重要性被提到了空前的高度,计算机基础课程在高等学校确立了公共基础课的地位。

在实施高等学校本科教学质量与教学改革的过程中,计算机基础的教学改革朝着高水平、应用化、规范化方向推进。

在完成教育部高等学校计算机基础课程教学指导委员会课题的过程中,我们组织了十余所高等学校计算机基础教学的负责人和一线教师,对这些高等学校中几千名本科生以问卷的形式对计算机基础教育现状进行了抽样调查,内容涉及了大学生起始计算机技能基本情况、目前课程安排和课程效果评价3个方面,着重了解了当代大学生对计算机基础教育的新需求及对计算机基础课程的意见与建议等。

通过对这些调查问卷进行科学的分析,我们得到一些计算机基础教学课程体系改革的启示。

在此基础上,按照《基本要求》的精神,结合计算机技术发展和应用的实际,以“知识—技能—能力”培养为目标,对计算机基础课程体系进行了重新的设计和调整,构建了“大学计算机基础+x门计算机应用课程”和“程序设计基础+x门计算机应用课程”两种“1+x”课程体系模式,形成了新型的计算机基础课程教学方案。

在以上课题研究的基础上,我们成立了“高等学校计算机基础教育改革与实践系列教材”编审委员会,希望能编写出一套适合于此教学方案的教材并建设相应的课程教学资源。

该系列教材以“面向应用、强化基础、注重融合”为原则,从面向应用的计算机硬件基础和软件基础两个角度入手,从融合专业技术的发展、社会对现代人才知识结构的要求出发,按照两种“1+x”的课程设置方案,选择了5门比较基础且通用的计算机基础课程来组织编写。

#### 1.从实践中来,到实践中去。

所有教学内容均从应用问题出发,以引例、实例和案例作为背景,提出每章的教学内容与教学目标,使学生对学习什么知识、为什么要学这些知识有一个概括的认识,并通过解决问题使所学基础知识得到强化。

所有引例、实例和案例都具有代表性,能激发学习的积极性,达到学以致用目的。

#### 2.内容新颖,知识结构更加合理。

所有教学内容进一步体现了新版《基本要求》的精神,并在此基础上,结合多年来教学改革与实践经验及地方经济结构和行业的需要,并融合相关专业知识,适当地增加了部分内容。

同时突破了传统的知识与教学模式,对相关内容的知识结构顺序做了调整,更利于学生对计算机基础知识的理解和掌握。

## <<C程序设计>>

### 内容概要

《C程序设计》以“零基础”为起点，首先阐述了学习C语言必备的计算机和程序设计的基础知识，然后以C语言语法和句法的规则、程序设计方法和计算机思维的培养为重点，以应用为主线，由浅入深地阐述了C语言的数据类型、结构化、模块化等知识点，归纳了选择、循环结构算法的设计思路及函数设计方法，配以大量的实例详细阐述了解决实际问题的程序设计方法。

《C程序设计》可以作为高等学校计算机及相关专业学生学习C语言程序设计的教材，也可作为参加有关考试和自学的参考书。

## &lt;&lt;C程序设计&gt;&gt;

## 书籍目录

第1章 计算机基础知识1.1 计算机硬件系统的组成及工作原理1.1.1 计算机硬件系统的组成1.1.2 计算机的工作原理1.2 数制及数据的存储1.2.1 数制1.2.2 计算机中数据的表示1.2.3 数制间的转换1.2.4 数据在内存中的存储1.3 计算机软件系统的组成1.3.1 软件系统1.3.2 机器语言和高级语言第2章 C语言程序设计基础2.1 利用计算机解决实际问题的具体步骤2.2 算法初步2.2.1 算法的概念和特性2.2.2 算法的表示2.3 结构化程序设计2.3.1 结构化程序设计概念2.3.2 结构化程序设计原则2.4 C语言简介2.4.1 C语言的产生与发展2.4.2 C语言的特点2.5 C语言的基本符号2.5.1 C语言的字符集2.5.2 C语言的词汇2.6 C语言程序的基本结构第3章 数据类型及表达式3.1 数据结构3.2 C程序中数据的表示方法3.2.1 常量3.2.2 变量3.3 运算符及其表达式3.3.1 C运算符概述3.3.2 算术运算符及算术表达式3.3.3 赋值运算符和赋值表达式3.3.4 自增自减运算符及含自增自减运算的表达式3.3.5 位运算3.3.6 逗号运算符和逗号表达式3.3.7 表达式小结3.4 c语言中的类型转换3.4.1 类型的自动转换3.4.2 强制类型转换习题第4章 顺序结构程序设计4.1 C语句概述4.1.1 语句的概念4.1.2 C语句的分类4.2 输出和输入函数4.2.1 putchar ( ) 和getchar ( ) 函数4.2.2 printf ( ) 和scanf ( ) 函数4.3 顺序结构程序设计示例习题第5章 选择结构程序设计5.1 条件的表示5.1.1 逻辑量的概念5.1.2 关系表达式5.1.3 逻辑表达式5.1.4 条件表达式5.2 if语句5.2.1 if语句5.2.2 if语句的嵌套5.2.3 选择结构程序设计示例5.3 Switch语句5.3.1 Switch语句5.3.2 Switch语句在分支程序设计中的应用习题第6章 循环结构程序设计6.1 循环的概念6.2 实现循环的语句6.2.1 While语句6.2.2 do.while语句6.2.3 for语句6.3 break和continue语句6.4 循环的嵌套6.5 循环结构程序设计习题第7章 函数7.1 函数概述7.1.1 模块化程序设计思想7.1.2 函数概述7.1.3 函数的分类7.2 函数的定义7.2.1 函数的定义形式7.2.2 函数的返回值7.3 函数的声明及函数调用7.3.1 函数声明7.3.2 函数的调用及参数传递7.3.3 函数的调用形式及规范7.3.4 函数的嵌套及递归调用7.4 变量的作用域与生命期7.4.1 变量的属性7.4.2 局部变量和全局变量7.4.3 变量的存储类型与生命期7.5 函数示例习题第8章 数组8.1 数组的基本概念8.2 一维数组8.2.1 一维数组的定义8.2.2 一维数组元素的初始化8.2.3 一维数组元素的引用8.2.4 一维数组程序设计示例8.3 二维数组8.3.1 二维数组的定义8.3.2 二维数组元素的引用8.3.3 多维数组8.4 字符数组及字符串8.4.1 字符数组的定义、引用及初始化8.4.2 字符串处理函数8.4.3 字符数组示例习题第9章 指针9.1 指针的基本概念9.1.1 变量的直接访问和间接访问9.1.2 指针与指针变量9.2 指针变量9.2.1 指针变量的类型和指针变量指向变量的关系9.2.2 指针变量的引用9.2.3 指针的运算9.2.4 指针变量作为函数参数9.3 数组与指针9.3.1 指向数组元素的指针变量：9.3.2 一维数组作函数参数和指向变量的指针变量作函数参数的一致性9.3.3 多维数组的指针9.3.4 字符串和指针9.4 指向函数的指针和返回指针值的函数9.4.1 指向函数的指针9.4.2 指向函数的指针变量9.4.3 指向函数的指针变量作函数参数9.4.4 返回指针值的函数9.5 指针数组与多级指针9.5.1 指针数组9.5.2 多级指针9.5.3 命令行参数习题第10章 结构体与联合体10.1 结构体的概念10.1.1 结构体类型的定义10.1.2 结构体变量的定义、引用、初始化10.1.3 结构体的嵌套10.2 结构体数组10.2.1 结构体数组的定义10.2.2 结构体数组的引用10.2.3 结构体数组的初始化10.3 结构体指针10.3.1 指向结构体变量的指针10.3.2 指向结构体数组的指针10.3.3 结构体指针作函数参数10.4 位字段10.4.1 位字段结构体的定义10.4.2 位字段结构体变量成员的引用10.5 链表10.5.1 链表的概念10.5.2 内存的动态存储、分配与释放函数10.5.3 链表的操作10.6 联合体数据类型10.6.1 联合体的定义10.6.2 联合体变量的引用10.7 自定义类型习题第11章 文件11.1 文件概述11.1.1 文件的概念11.1.2 缓冲文件系统11.2 文件的使用11.2.1 文件类型指针11.2.2 文件的打开11.2.3 文件的关闭11.2.4 文件的读写11.2.5 文件的定位11.2.6 文件的检测11.3 文件程序设计示例习题第12章 编译预处理12.1 宏定义12.1.1 不带参数的宏定义12.1.2 带参数的宏定义12.2 文件包含12.3 条件编译习题附录1 ASCII字符表附录2 C语言的运算符及优先级和结合性附录3 VisualC++6.0集成开发环境附录4 常用标准库函数

## 章节摘录

内存储器是计算机硬件系统中极其重要的部件。

根据其对数据的存储的方式不同可分为只读存储器（ROM）和随机存储器（RAM）。

只读存储器只存储计算机的基本硬件信息，其信息不可改变，这些基本信息对计算机各部件的协调工作具有极其重要的作用，控制器正是通过这些硬件的基本信息完成了对计算机硬件的控制。

随机存储器（习惯上称为内存）是计算机数据交换的中心，用以完成计算机数据的临时存放，计算机输入输出的数据都是通过内存到达各部件的。

之所以增加内存这一重要部件是由于计算机的CPU与外部设备工作速度不同造成的，CPU的工作速度快，而外部设备的工作速度相对较慢，若不设内存部件，CPU大量的时间处于等待，形成资源浪费；其次外部设备的种类繁多，通过内存大量的数据信息就可以在内存中暂时存放，有利于充分利用CPU资源和协调外部设备各部件之间的工作，从而极大地提高计算机的工作效率。

外部设备可以划分为输入设备、输出设备和外存储器。

外部设备中的输入输出（I/O）设备是人与计算机交互的重要途径，通过它们就可以输入数据、控制计算机的工作，查看计算机的工作过程和得到计算结果。

输入设备是人与计算机交互的第一通道，人对计算机的控制需通过输入设备来完成，随着计算机功能的强大，输入方式变得更加简单，输入形式日趋多样化。

例如，鼠标的应用、手写板的出现和语音系统的发展都使人与计算机的交互从频繁的键盘敲击中解脱出来。

输出设备是人与计算机交互的结果性终端设备，计算机工作所得到的文字、图像和声音等结果需通过输出设备体现出来。

通过这些终端设备用户得以监视计算机的工作过程，进而为进一步的工作做出判断。

目前输出设备向着人性化和多功能化方向发展。

外存储器是计算机用以长期存放数据的设备，外存储器主要有软磁盘（简称软盘）、硬磁盘（简称硬盘）、光盘、磁带等。

其中硬盘是计算机最重要的外存储器，几乎所有系统软件和应用软件以及用户数据都存放在这里（网络中的无硬盘瘦客户机除外），硬盘存储器已成为计算机硬件系统中最重要的数据存储介质。

光盘是利用化学材质和激光原理在盘片上记录数据的存储设备，其数据需通过光学镜头读取，所以称为光盘，光盘可以大量地永久性保存数据，已成为计算机外存储设备中重要的介质。

随着计算机技术及相关技术的提高，大容量光盘已成为永久性数据备份和存储的重要途径。

磁带是顺序存储数据的介质，磁带可以进行大规模数据的存储和备份，但由于其存取速度较慢，已趋于淘汰。

目前外存储器在稳定性、容量和存储速度上都得到了巨大的进步，同时出现了各类大容量可热拔插（计算机工作时可任意插入和拆除的外存储设备）的外存储器，更加方便了数据的输入输出。

<<C程序设计>>

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:<http://www.tushu007.com>