

## <<C程序设计伴侣>>

### 图书基本信息

书名：<<C程序设计伴侣>>

13位ISBN编号：9787115290953

10位ISBN编号：7115290954

出版时间：2012-10

出版时间：人民邮电出版社

作者：陈良乔

页数：315

字数：534000

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程序设计伴侣&gt;&gt;

## 前言

编辑的话：为何策划这本书我第一次看谭老师的《C程序设计》，是20年前，那时我17岁，大学一年级。

当时感觉这本书很神奇，对编程充满了好奇。

同时也感觉这本书有些难，有些地方，比如指针，虽然看过书，也有老师讲过了，但依然感觉似懂非懂。

还有些知识，书中只讲了怎么做，但没有讲为什么要这么做，比如为什么所有的C程序都要有一个main()函数呢。

就这样，带着一些无伤大雅的困惑，时光流过20年。

2011年初，我在书店无意中看到《C程序设计》已经出到了第四版。

并且在书的封面上印着1100万册的销量数字，视觉效果十分震撼。

我想这大约意味着有1100万人已经读过了这本书，其中想必会有一些和我一样在读书中遇到困惑的读者吧。

考虑到1100万的基数，我想遇到困惑的读者大概也不在少数。

考虑到这本书几乎是所有大学新生所接触的第一本编程书，其意义和重要性非同小可。

这样一想，此事未免让人放心不下。

想来想去，我决定做点什么，为这些人做些什么，为这本书做些什么，从而让这本书变得更好，让它的广大读者中有困惑的那部分能减少一些困惑。

想到古典名著多有注解、译文和批注，来帮助看不懂文言文的读者更好地理解这些经典的古典名著。

那么，我为什么不《C程序设计》这本计算机的经典名著来出一本注解和批注的书呢？

把原书中没有讲透彻没有讲明白的那些概念、知识和技术点来认认真真地讲讲清楚。

进而又想，既然已经走到了注解和批注的地步，那就不妨再往前走得更远一些吧，把原书中遗漏的、忽视的、错误的，以及那些未曾涉及的但作为初学者或称为未来的程序员绝对应该知道的基础知识和素养一并讲讲清楚。

因为，有些错误的认识会对初学者产生深远的影响。

这本书因《C程序设计》的存在而诞生，它是对《C程序设计》这本书的强有力的注解和补充。

如同擎天柱和天火合体后，能够冲天高飞壮志凌云，这本书将成为《C程序设计》的强劲的喷气引擎。

在语言风格和表现形式上，这本书比《C程序设计》更加的有趣、更加的通俗易懂。

在内容上，这本书比《C程序设计》更加的深刻、更加的透彻，具有更加广阔的视野和更加深厚的内涵。

在实用性上，这本书比《C程序设计》更加的实战，可以让你养成良好且符合业界标准的编程习惯和编程思想，为今后的继续前行打下必要而优秀的基础。

同时，由于这本书对概念、知识和疑难点讲解地非常透彻，因此，这本书也会助你轻松通过C程序设计课程的考试。

我把这本书，命名为&mdash;&mdash;《C程序设计伴侣》。

本书策划编辑：陈冰2012年2月12日作者的话更有趣、更容易、更深刻、更实战的《C程序设计伴侣》&mdash;&mdash;《C程序设计》的有力补充和最佳伴侣在中国，几乎每一个人，第一次接触程序设计，都是通过谭浩强老师的《C程序设计》开始的。

从某种意义上说，它是中国万千程序员的启蒙读本。

《C程序设计》累计发行逾1100万册，不仅创计算机图书的全国最高销量纪录，在世界上也是罕见的。

这本书也因此获得多项殊荣。

然而，这些成绩仍无法掩盖这样一个事实：很多通过这本书学习C语言的初学者，在学习的过程中，对书中讲解的内容感到晦涩生硬，难以理解和掌握；而在他们历尽辛苦终于掌握了一些知识之后，在要使用这些知识来解决一个实际问题时，却因情况的复杂而感到无从下手；即使是分析出了可能的切

## &lt;&lt;C程序设计伴侣&gt;&gt;

入点，又因为经验的缺乏，在具体实施时，感到困难重重。

在整本书的学习过程中，充满了对C语言的各种概念、知识、规则的较生硬的讲解，而对于打基础阶段十分重要的正确的编程素养、编程思想和编程技巧，却几乎没有提及。

C语言虽然是高级语言，但却是相对更接近底层的高级语言。

对于第一次接触编程的初学者而言，C语言，就像一座充满魔法的强大而庄严的森林，没有广阔视野，没有高空的俯视，没有称手的地图。

直接的贸然闯入，是十分危险的。

而谭老师的《C程序设计》仿佛在只给了初学者背包、水壶和印在门票上的小地图的情况下，就举着小旗把读者带进了这座强大的森林。

学习C语言程序设计，就像我们学习写文章一样。

仅仅掌握了具体的词法句法，是无法写出优美的文章的，除了需要阅读大量的名著外（这个毫无疑问需要时间的累积），更重要的，是你首先要在思想上真正明了文字的力量和掌握构造文章整体架构的技巧。

遗憾的是，谭老师的《C程序设计》教给了读者C语言的语法，却没能教给读者更好地去设计一个程序。

为了给这部经典著作弥补这一缺憾，陈冰老师策划了这本不可思议的《伴侣》。

而我很荣幸得到了具体来创作这本书的机会。

针对C语言中的重点和难点，本书进行了更加生动有趣、更加深刻，但同时又更容易理解的阐述。

读者在学习之后，不仅会对C语言中的各种知识有更加生动形象的理解，更重要的是，能够知道程序是怎样炼成的，在以往隔靴搔痒的表面下真正发生了什么。

并最终用C语言编写程序加以解决，顺利地完成了从“问题”到“程序”的这样一个编程解决问题的过程。

本书不深究C语言的语法规则，在这一方面，谭浩强老师的《C程序设计》已经讲得很详尽了。

我们不会浪费时间重复发明轮子。

一句话，所有《C程序设计》中已经讲明白的地方，本书都不重复讲解。

本书的笔墨放在了彻底讲清楚难以理解的概念、重点和难点上，放在了帮助初学者建立良好的编程思想和符合业界标准的编程规范上，放在了从实际应用出发，指导初学者在实际开发和应用中遇到问题时如何思考和解决上。

读者在学习这本《伴侣》后，不仅会因为拥有了一个从未有过的清晰的编程头脑而可以轻松通过C程序设计课程的考试，更会因为建立起可圈可点的专业的编程思想而拥有更强的解决实际问题的程序设计和开发能力。

谭老师的《C程序设计》比较生硬晦涩，这本书让它更加有趣；《C程序设计》所讲解的知识难以理解，这本书让它更加容易；《C程序设计》只注重知识的讲解，这本书为它补充上基础而宝贵的编程素养、编程思想、编程经验，让你从深处开始强大。

喝咖啡，常常会加上咖啡伴侣或是牛奶，它会遮盖咖啡的苦涩，让咖啡更加香醇。

如果把谭浩强老师的《C程序设计》比作一杯略带苦涩的咖啡，这本书就是它最佳的伴侣，让你的C程序设计之路更香醇。

本书作者：陈良乔2012年2月2日如何使用这本书翻开书，你会发现这本《C程序设计伴侣》（以下简称《伴侣》）的目录与谭浩强老师的《C程序设计》（第四版）的目录有些类似，但二者又有明显的区别。

事实上，准确地说，这本书的目录是谭浩强老师的《C程序设计》目录的超集。

也就是说，谭浩强老师的《C程序设计》的目录中的每一个目录都在《伴侣》中有一个与之对应的目录，这两个目录的标题序号是完全一致的，但标题名并不一样或并不完全一样。

比如：《C程序设计》的1.4节是“最简单的C语言程序”，与之对应的《伴侣》中的目录就是1.4节的“第一次的亲密接触——Hello, World”。

这些目录在《伴侣》中会以双标题的形式出现，上层是《伴侣》中的标题，下层是《C程序设计》中的对应标题。

## &lt;&lt;C程序设计伴侣&gt;&gt;

这样的目录安排，是为了读者查阅和学习的方便。

读者可以在学习完《C程序设计》的某一章节后，带着疑问再来看《伴侣》中对应的章节。

通过《伴侣》中更加生动有趣和更加透彻的解释，你可以更好地理解这一章节的内容，有一种豁然开朗的感觉。

除了有与《C程序设计》中一一对应的目录外，《伴侣》中还穿插有一些特别的目录，这些目录的标题序号中含有字母（A、B、C、D等）。

这些含有字母的目录表示是《伴侣》额外增加的，其中的内容是《伴侣》中独有的全新内容。

比如：《伴侣》中的1.4.A节“最简单的C语言程序背后的故事——它的汇编代码是如何被执行的”就是《伴侣》中独有的全新内容。

这些内容或者是学好C语言所必需的知识和经验，或者是你应理解和掌握的编程素养和思想，而由于这些内容在《C程序设计》中完全没有涉及，因此，作者以独立章节的形式给以补充。

当你的学习进展到相应章节的时候，可以独立地阅读这些章节，从而丰富你的知识、开阔你的眼界、增加你的经验。

一句话，双标题章节帮助消化，新增章节补充营养。

此外，当在《伴侣》中提到例x.x时指的是《C程序设计》中的例x.x。

因为《C程序设计》是以Visual C++ 6.0作为开发工具的，所以《伴侣》在介绍开发工具的使用时，仍以Visual C++ 6.0为例，使得大家不至于太过陌生。

而在编译代码的时候，为了让大家对C语言的最新发展有所了解，我们采用了以下两款主流编译器的最新版本作为我们的主要编译器。

本书大部分代码都可以使用MinGW（Minimalist GNU for Windows）并添加-std=gnu99编译选项进行编译。

或者是使用Visual Studio 2012的“VS 2012开发人员命令提示”（位于“开始”→“所有程序”→“Microsoft Visual Studio 2012”→“Visual Studio Tools”）下，添加/TP编译选项进行编译。

例如，你可以使用如下的命令来编译第一章的Hello World程序：使用MinGW编译：`gcc`

`&ndash;std=gnu99 HelloWorld.c -o HelloWorld.exe`使用Visual C++ 2012编译：`cl /TP HelloWorld.c`其中的少部分代码，因为用到了一些特殊的编译器扩展，只能在“VS 2012开发人员命令提示”下进行编译。

例如第七章的简易下载程序：`cl /TP get.c`关于开发工具和源代码的下载，以及开发环境的配置，欢迎访问本书作者的技术支持网站：<http://chenlq.net/>本书策划编辑：陈冰 本书作者：陈良乔

## <<C程序设计伴侣>>

### 内容概要

本书是谭浩强老师经典的《C程序设计》（以下简称原书）一书的最佳伴侣，是对原书有益且神奇的补充。本书对原书中没有讲透彻的没有讲明白的那些概念、知识和技术点进行了深刻却易于理解的讲解，并把原书中遗漏的、忽视的，甚至是多少有点问题的，以及那些未曾涉及的但作为初学者或称为未来的程序员绝对应该知道的基础知识、编程素养、编程思想，以及业内行规一并讲解清楚了。

较之原书，这本伴侣更加的有趣、更加的通俗易懂。在内容上，较之原书，这本伴侣更加的深刻、更加的透彻，具有更加广阔的视野和更加深厚的内涵。在实用性上，这本伴侣较之原书更加的实战，可以让你养成良好且符合业界标准的编程习惯和编程思想，为今后的继续前行打下必要而优秀的基础。

## <<C程序设计伴侣>>

### 作者简介

陈良乔

一个喜欢写作的程序员。

一个喜欢编程的文艺青年。

连续八年获得微软最有价值专家 (MVP) 称号。

曾任东软数字医疗项目经理；德国IB&T中国公司架构师。

挚爱技术，好分享。

作为讲师进行百人以上微软技术讲座数十次。

因撰写一系列有分量的Visual

Studio和Windows 7技术文章荣获“MSDN原创之星”称号。

因参与Visual

Studio的工作，荣获“Visual

Studio贡献水晶奖杯”。

生性乐观，敏而好学，乐于分享。

平素喜好新鲜事物，但绝非新潮技术的追捧者，至今仍执迷C/C++这些古老而新鲜的技术。

好舞文弄墨，传播心得。

著有《我的第一本C++书》。

组织并参与翻译《C++

11 FAQ中文版》。

作者网站（及本书源代码下载地址）：<http://chenlq.net>

欢迎亲爱的读者们访问交流

## &lt;&lt;C程序设计伴侣&gt;&gt;

## 书籍目录

注：浅灰色的目录标题是本书所对应的谭浩强《C程序设计》（第四版）中的相应章节的目录标题。

## 第1章 C语言背后的故事程序设计和C语言

## 1.1 计算机程序是什么什么是计算机程序

## 1.2 计算机语言又是什么什么是计算机语言

## 1.2.A 如何用计算机语言来表达一个程序

## 1.2.B 从汇编语言到高级语言——风格各异的计算机语言

## 1.3 C语言真正的特点及其演变 C语言的发展及其特点

## 1.3.A C语言与C++不得不说的那点事

## 1.3.B 我们为什么现在还要学习C语言

## 1.4 第一次的亲密接触——Hello, World 最简单的C语言程序

## 1.4.A 最简单的C语言程序背后的故事——它的汇编代码是如何被执行的

## 1.4.1 比最简单再复杂一点点最简单的C语言程序举例

## 1.4.2 程序的三要素：数据输入、数据处理与数据输出 C语言程序的结构

## 1.4.B 编码规范——如何写出简洁优美的代码

## 1.5 C程序开发的两个循环和四个步骤运行C程序的步骤与方法

## 1.5.A 工欲善其事，必先利其器——使用Eclipse编写C程序

## 1.5.B 使用GCC编译C语言程序

## 1.6 程序设计的瀑布模型程序设计的任务

## 第2章 解决问题的思路——算法算法——程序的灵魂

## 2.1 先有算法，后有程序什么是算法

## 2.1.A 算法的时间复杂度和空间复杂度

## 2.2 如何将抽象的算法实现为具体的代码简单的算法举例

## 2.3 算法的特征——简单，没啥可说的算法的特性

## 2.4 一群聪明的程序员是怎么把事情搞砸的怎样表示一个算法

## 2.5 掌握结构化的程序设计结构化程序设计方法

## 2.5.A 结构化程序设计的基本思想、步骤与流程

## 2.5.B “自顶向下，逐步求精”：把“问题”变成“程序”的魔法

## 第3章 顺序结构的程序设计最简单的C程序设计——顺序程序设计

## 3.1 顺序结构程序举例——简单，没啥可说的顺序程序设计举例

## 3.1.A 如何理解程序的执行顺序

## 3.2 如何描述数据，如何对数据进行运算数据的表现形式及其运算

## 3.2.1 常量、变量、常变量及符号常量和变量

## 3.2.2 数据类型及举例 数据类型

## 3.2.3 整型数据及如何选择合适的类型整型数据

## 3.2.4 字符型数据及举例字符型数据

## 3.2.5 浮点型数据及比较大小时的特别注意浮点型数据

## 3.2.6 如何强制指定常量的数据类型怎样确定常量的类型

## 3.2.7 运算符和表达式及使用经验和建议运算符和表达式

## 3.3 基本程序单位——语句 C语句

## 3.3.1 C语言的语句分类 C语句的作用和分类

## 3.3.2 赋值语句及其使用注意最基本的语句——赋值语句

## 3.4 数据从哪里来，又到哪里去？

## 数据的输入输出

## 3.4.1 输入输出举例——简单，没啥可说的输入输出举例

## &lt;&lt;C程序设计伴侣&gt;&gt;

- 3.4.2 输入输出的概念——简单，没啥可说的有关数据输入输出的概念
- 3.4.3 使用printf函数进行格式化数据输出用printf函数输出数据
- 3.4.4 使用scanf函数进行格式化数据输入用scanf函数输入数据
- 3.4.5 单字符的输入输出及敏感信息输入函数字符数据的输入输出
- 3.4.5A 字符串数据的简单输入输出
- 第4章 选择结构的程序设计选择结构程序设计
- 4.1 选择结构和条件判断——交通灯例子选择结构和条件判断
- 4.2—4.3 使用if语句实现选择结构
- 4.4 逻辑运算符及表达式逻辑运算符和逻辑表达式
- 4.4.1 逻辑运算符及优先级逻辑运算符及其优先次序
- 4.4.1A 合理的出错处理——让你的程序更受欢迎
- 4.4.2 逻辑表达式及其使用注意逻辑表达式
- 4.4.3 逻辑型变量——时控灯例子逻辑型变量
- 4.5 条件运算符及表达式条件运算符和条件表达式
- 4.6 条件结构的嵌套——拜金女择偶例子选择结构的嵌套
- 4.7 使用switch语句实现多分支选择结构——看天穿衣例子用switch语句实现多分支选择结构
- 4.8 综合运用——猜数字游戏选择结构程序综合举例
- 第5章 循环结构的程序设计循环结构程序设计
- 5.1 为什么需要循环控制——简单，没啥可说的为什么需要循环控制
- 5.1A 循环结构的四个要素
- 5.2 如果条件满足，就一直做某事的while 用while语句实现循环
- 5.3 一直做某事，直到条件不满足的do...while 用do...while语句实现循环
- 5.4 最完备最复杂的循环语句for 用for语句实现循环
- 5.5 更复杂的数据处理——嵌套的循环循环的嵌套
- 5.6 如何选择合适的循环结构几种循环的比较
- 5.7 改变循环的状态改变循环执行的状态
- 5.7.1—5.7.2 用break和continue语句提前终止循环
- 5.7.3 break和continue的不同 break语句和continue语句的区别
- 5.8 综合运用——检测磁盘剩余容量与统计字符串中的单词数循环程序举例
- 第6章 处理批量数据的利器——数组利用数组处理批量数据
- 6.1 如何定义和引用一维数组怎样定义和引用一维数组
- 6.1.1 如何定义一维数组——保存学生成绩的例子怎样定义一维数组
- 6.1.2 如何引用一维数组怎样引用一维数组元素
- 6.1.3 使用memset()函数进行一维数组的初始化一维数组的初始化
- 6.1.4 用qsort()函数对数组进行排序一维数组程序举例
- 6.2 如何定义和引用二维数组怎样定义和引用二维数组
- 6.2.1 如何定义二维数组及二维数组的内存排列方式怎样定义二维数组
- 6.2.2 如何引用二维数组怎样引用二维数组的元素
- 6.2.3 使用memset()函数进行二维数组的初始化二维数组的初始化
- 6.2.4 对二维数组中的数据进行排序二维数组程序举例
- 6.2.4.A 分析问题，定义二维数组描述批量数据
- 6.2.4.B 向二维数组输入数据
- 6.2.4.C 用qsort()函数对二维数组进行排序
- 6.2.4.D 输出二维数组结果
- 6.2.A 扩展：多维数组的使用
- 6.3 字符数组的本质 字符数组
- 6.3.1—6.3.3 如何定义字符数组

## &lt;&lt;C程序设计伴侣&gt;&gt;

- 6.3.4 字符串和字符数组的区别字符串和字符串结束标志
- 6.3.5 字符数组和字符串的输入输出字符数组的输入输出
- 6.3.6 使用字符串处理函数完成字符串处理任务使用字符串处理函数
  - 6.3.6.A 用gets()和puts()输入输出字符串
  - 6.3.6.B 用strcat()连接两个字符串
  - 6.3.6.C 用strchr()和strstr()查找字符串
  - 6.3.6.D 用strcmp()比较字符串
- 6.3.7 综合运用——自己写一个strsub()函数截取字符串字符数组应用举例
- 6.A 如何运用数组处理批量数据——一个计算平均血糖的例子
  - 6.A.A 什么时候应该使用数组
  - 6.A.B 如何将数据输入到数组
  - 6.A.C 如何处理数组中的数据
  - 6.A.D 数组使用的三部曲：定义数组、输入数据、处理数据
- 第7章 模块化程序设计的根基——函数用函数实现模块化程序设计
  - 7.1 为什么要用函数——便于代码的管理和复用为什么要用函数
  - 7.2 如何声明函数 怎样定义函数
    - 7.2.1 为什么要声明函数为什么要定义函数
    - 7.2.2 声明函数的方法定义函数的方法
      - 7.2.2A 声明函数的三要素：返回值、函数名、参数列表
  - 7.3 如何调用函数 调用函数
    - 7.3.1 调用的形式及调用语句函数调用的形式
    - 7.3.2 函数调用时的数据传递——两个方向的传递函数调用时的数据传递
      - 7.3.2A 如何向函数传递体积很大的数据
    - 7.3.3 调用的过程——简单，没啥可说的函数调用的过程
    - 7.3.4 关于函数的返回值函数的返回值
      - 7.3.4A 如何从函数返回多个结果数据
  - 7.4 编译器对函数声明的要求对被调用函数的声明和函数原型
    - 7.4A 创建一个我们自己的头文件
  - 7.A 站在巨人的肩膀上——如何使用已有的函数库
  - 7.B 我也能成为巨人——如何创建一个函数库
  - 7.5 嵌套的函数调用函数的嵌套调用
  - 7.6 函数的递归调用——统计字符串出现的次数函数的递归调用
  - 7.C 让函数飞——在C语言中嵌入汇编代码
  - 7.7 将数组作为函数参数传递数组作为函数参数
  - 7.8 变量的生存期局部变量和全局变量
    - 7.8.1 局部变量的定义及作用域局部变量
      - 7.8.1A 不要从函数返回指向局部变量的指针
    - 7.8.2 全局变量及跨文件的全局变量全局变量
  - 7.9 变量的存储方式及其区别变量的存储方式和生存期
  - 7.10 变量的声明和定义的区别——是否建立存储空间关于变量的声明和定义
  - 7.11 控制函数的访问范围内部函数和外部函数
  - 7.D 函数使用的四项基本原则
- 第8章 指针？  
没啥好怕的！  
善于利用指针
  - 8.1 什么是指针 指针是什么
  - 8.2 什么是指针变量 指针变量

## &lt;&lt;C程序设计伴侣&gt;&gt;

## 8.2.1 使用指针变量的例子——指针的四大用途使用指针变量的例子

## 8.2.1A 什么是NULL指针

## 8.2.2 如何定义指针变量怎样定义指针变量

## 8.2.2A 内存的动态申请与释放

## 8.2.3 如何引用指针变量怎样引用指针变量

## 8.2.3A 指针变量存放在哪里呢？

## 8.2.4 使用指针变量作为函数参数的场合指针变量作为函数参数

## 8.3 更灵活的数组访问方式——使用指针访问数组通过指针引用数组

## 8.3.1—8.3.3A 使用指针访问数组的要点

## 8.3.4 如何向函数传递一个数组用数组名作函数参数

## 8.3.5 通过指针引用多维数组的要点——统计学生成绩的例子通过指针引用多维数组

## 8.4 利用指针引用字符串通过指针引用字符串

## 8.5 定义指向函数的指针指向函数的指针

## 8.5.1 了解函数指针什么是函数指针

## 8.5.1A 函数指针背后的故事——剖析它的汇编代码

## 8.5.2 利用函数指针变量调用函数用函数指针变量调用函数

## 8.5.3 函数指针变量的定义和使用怎样定义和使用指向函数的指针变量

## 8.5.4 用函数指针作函数参数——在函数之间传递函数，即传递操作

## 用指向函数的指针作函数参数

## 8.6 使用指针作为函数的返回值返回指针值的函数

## 8.7 指针数组及指向指针的指针指针数组和多重指针

## 8.7.1 什么是指针数组——一个电磁炉火力控制程序什么是指针数组

## 8.7.2 指向指针的指针指向指针数据的指针

## 8.7.3 向main()函数传递数据指针数组作main函数的形参

## 8.8 动态内存分配及动态调整动态内存分配与指向它的指针变量

## 8.9 指针的“醒世恒言”有关指针的小结

## 8.A 如何避免错误地使用指针

## 第9章 建立自己的数据类型用户自己建立数据类型

## 9.1 结构体变量登场定义和使用结构体变量

## 9.1.1—9.1.3 关于这三小节的特别说明

## 9.1.A 定义结构体应该注意的问题

## 9.1.B 使用结构体以及它的数据成员

## 9.2 使用结构体数组——利用qsort()函数对数组排序使用结构体数组

## 9.2.1—9.2.2 关于这两小节的特别说明

## 9.3 利用结构体指针减少传递的数据量结构体指针

## 9.3.1—9.3.3 关于这三小节的特别说明

## 9.4 笔试必考，面试必问的数据结构——链表用指针处理链表

## 9.4.1 链表是神马 什么是链表

## 9.4.2 如何建立静态链表建立简单的静态链表

## 9.4.3 如何建立动态链表建立动态链表

## 9.4.4 如何输出链表 输出链表

## 9.4.A 链表数据的处理：对链表进行排序和查找

## 9.4.B 链表数据的处理：查找结点

## 9.4.C 释放一个链表

## 9.5 共用体类型的定义和使用共用体类型

## 9.5.1—9.5.3 关于这三小节的特别说明

## &lt;&lt;C程序设计伴侣&gt;&gt;

9.6 什么是枚举类型使用枚举类型

9.7 数据类型也可以有个更好记更好用的绰号——用typedef声明新的数据类型用typedef声明新类型名

## 第10章 文件的输入与输出对文件的输入输出

10.1 在C中文件的概念是怎样的 C文件的有关基本知识

10.1.1—10.1.5 关于这五小节的特别说明

10.1A C语言所处理的文件类型

10.2 如何打开与关闭文件打开与关闭文件

10.2.1 使用fopen函数打开数据文件及注意事项用fopen函数打开数据文件

10.2.1A 用freopen()重定向标准输入输出流

10.2.2 使用fclose函数关闭数据文件及注意事项用fclose函数关闭数据文件

10.3 对数据文件进行顺序读写顺序读写数据文件

10.3.1—10.3.2 怎样向文件读写字符和字符串

10.3.3 使用函数来进行数据文件的格式化读写用格式化的方式读写文件

10.3.4 二进制方式的文件读写——读写一张图片用二进制方式向文件读写一组数据

10.3.A 文件读写的“三大纪律”

10.4 在任意位置读写数据文件随机读写数据文件

10.4.1—10.4.2 关于这两小节的特别说明

10.5—10.5A 扩展：访问文件的各种属性

10.5A.A 如何判断文件是否存在

10.5A.B 如何获得文件的日期和大小

## 第A章 程序是如何炼成的

A.A 问题的提出——统计网页中的词频

A.B 自顶向下：分析问题，划分模块

A.B.A 分析数据输入模块

A.B.B 分析数据处理模块

A.B.C 分析数据输出模块

A.C 逐步求精：用函数实现各个模块

A.C.A 实现数据输入模块

A.C.B 实现数据处理模块

A.C.C 组装实现数据预处理模块

A.C.D 实现计算词频模块

A.C.E 实现文件排序模块

A.C.F 实现数据输出模块

A.C.G 打扫战场：释放动态申请的内存

A.D 百炼成钢：如何调试我们的程序

A.E 程序就是这样炼成的

## 第B章 接下来我该.....

B.A 接下来要读的书

B.A.A 提升对C语言的理解和掌握

B.A.B 做一个有思想的程序员

## <<C程序设计伴侣>>

### 编辑推荐

它几乎是每一位程序员的启蒙读物！

它的销量超过1100万册！

它就是众所周知的《C程序设计》&mdash;&mdash;谭浩强老师的经典著作！

然而，它也有一些不足之处，这使得20年来，关于它的争议没有停止过，20年来，它，一直是孤独的。

直到，《伴侣》的出现！

《伴侣》让“它”变得更完整，我们的目的是让《C程序设计》读者的困惑和遗憾在这本《伴侣》中得到解决。

学习《C程序设计》，必配《C程序设计伴侣》！

咖啡，加牛奶，味道更香醇&hellip;&hellip;

<<C程序设计伴侣>>

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>