



普通高等教育“十一五”国家级规划教材



C语言程序设计

顾治华 陈天煌 忽朝俭 编著

为教师提供教学课件



机械工业出版社
China Machine Press



普通高等教育“十一五”国家级规划教材

C语言程序设计

顾治华 陈天煌 忽朝俭 编著



机械工业出版社
China Machine Press

本书系统介绍C语言程序设计，首先介绍算法及算法设计、数据描述与算法操作内容，在此基础上，介绍C语言程序设计中的选择结构程序设计、循环结构程序设计、数组与指针、函数与模块化程序设计、结构体和共用体等内容。最后，本书还对C++语言进行了简单的介绍。本书注重基础概念及综合应用训练，包含大量例题与习题，可帮助读者快速入门并掌握编程技巧。

本书可作为理工科专业本、专科的教材，并可作为程序设计初学者的入门读物。

版权所有，侵权必究。

本书法律顾问 北京市展达律师事务所

图书在版编目（CIP）数据

C语言程序设计 / 顾治华等编著. —北京：机械工业出版社，2007.2
（普通高等教育“十一五”国家级规划教材）
ISBN 978-7-111-20761-0

I. C… II. 顾… III. C语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆CIP数据核字（2007）第007984号

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）

责任编辑：武恩玉

北京牛山世兴印刷厂印刷 · 新华书店北京发行所发行

2007年2月第1版第1次印刷

184mm×260mm · 17印张

定价：25.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换
本社购书热线（010）68326294

前 言

C语言是计算机学科重要的核心课程，也是其他理工科专业计算机基础训练的必修课。由于C语言兼备高级程序语言和低级程序语言的诸多优点，在计算机工程实践中也得到了广泛的应用，因此近年来，学习和掌握C语言的需求越来越多，特别是对于进一步提高C语言程序设计能力的需求也越来越多。

在多年的教学过程中我们发现，相当多的学生学完了“C语言程序设计”课程以后，仅能了解和掌握一些语句的语法知识和语义，但不会应用语言来编写程序，把编程视为十分艰难而又高不可攀的工作。

本书在介绍C语言的同时，注重讲解如何应用C语言来编写程序，试图帮助读者克服畏难情绪，在轻松、愉快的气氛中探索程序设计的奥妙。学习编程技巧是一个不断实践，反复练习的过程。本书编写的目标就是希望帮助读者缩短这个过程，迅速提高C语言程序设计能力和水平。本书中的算法思维训练和编程思想同样适用于其他高级程序设计语言。本书具有如下特点：

- 1) 面向初学者，使略有计算机基础的人都能容易地学会C语言编程。书中实例充满趣味性和实用性，语言叙述通俗易懂，难点分散，概念清晰，层次分明。
- 2) 为了帮助读者学会程序设计，本书还专门介绍了一些程序设计的常用方法，如穷举法、迭代法、递推法等。使读者在学习和掌握一门语言的同时养成良好的程序设计习惯。
- 3) 注重各部分知识的综合应用训练，以提高程序设计能力为目标。
- 4) 习题中选用了部分等级考试试题，对于应试也具有实用性。
- 5) 为了帮助读者充分利用Internet上十分丰富的学习资源，本书在附录中提供了百余个中英文的关键词，由这些关键词可搜索到更多更新的C语言程序设计的文献资料。

本书在写作的过程中，与多名讲授该课程的教师进行过讨论，从他们那里汲取了许多宝贵的教学经验。同时，本书的编写工作还得到了院系领导及机械工业出版社华章分社的大力支持。在此，一并表示衷心的感谢。

由于计算机科学技术发展迅速，加之编者水平有限，书中错误在所难免，恳请广大读者批评指正，并多多提出宝贵意见。

作者于武汉理工大学

2006年10月



机械工业出版社 华章公司

Huazhang Graphics & Information Co., Ltd

教师服务登记表

尊敬的老师:

您好!感谢您购买我们出版的_____教材。

机械工业出版社华章公司本着为服务高等教育的出版原则,为进一步加强与高校教师的联系与沟通,更好地为高校教师服务,特制此表,请您填妥后发回给我们,我们将定期向您寄送华章公司最新的图书出版信息,为您的教材、论著或译著的出版提供可能的帮助。欢迎您对我们的教材和服务提出宝贵的意见,感谢您的大力支持与帮助!

个人资料(请用正楷完整填写)

教师姓名		☐ 先生 ☐ 女士	出生年月		职务		职称: ☐ 教授 ☐ 副教授 ☐ 讲师 ☐ 助教 ☐ 其他
学校				学院			
联系电话	办公:			联系地址及邮编			
	宅电:						
	移动:			E-mail			
学历		毕业院校	国外进修及讲学经历				
研究领域							
主讲课程		现用教材名		作者及出版社	共同授课教师	教材满意度	
课程: ☐ 专 ☐ 本 ☐ 研 ☐ MBA 人数: 学期: ☐ 春 ☐ 秋						☐ 满意 ☐ 一般 ☐ 不满意 ☐ 希望更换	
课程: ☐ 专 ☐ 本 ☐ 研 ☐ MBA 人数: 学期: ☐ 春 ☐ 秋						☐ 满意 ☐ 一般 ☐ 不满意 ☐ 希望更换	
课程: ☐ 专 ☐ 本 ☐ 研 ☐ MBA 人数: 学期: ☐ 春 ☐ 秋						☐ 满意 ☐ 一般 ☐ 不满意 ☐ 希望更换	
备注	已出版著作			译著			
	著书	方向一					
	计划	方向二					
	是否愿意从事翻译工作 ☐ 是 ☐ 否			翻译方向			
意见和建议							

填妥后请选择以下任何一种方式将此表返回:(如方便请赐名片)

地址:北京市西城区百万庄南街1号 华章公司营销中心 邮编:100037

电话:(010) 68353079 88378995 传真:(010) 68995260

E-mail: hzedu@hzbook.com marketing@hzbook.com 图书详情可登录<http://www.hzbook.com>网站查询

目 录

前言

第1章 C语言程序设计概述	1
1.1 程序与程序设计	1
1.2 C语言简介	2
1.2.1 C语言的发展历程	2
1.2.2 C语言的优点和规则	3
1.3 简单的C语言程序	4
1.4 C语言程序的上机步骤	4
1.5 C语言的基本词法	6
1.6 C语言程序的基本结构	7
本章小结	8
习题	9
第2章 算法及算法设计简介	11
2.1 算法的概念	11
2.2 C语言基本语句类型及算法的表示方式	12
2.2.1 C语言基本语句类型	12
2.2.2 算法的表示方法	12
2.3 简单的算法实例	14
2.4 结构化程序设计方法简介	17
本章小结	18
习题	19
第3章 数据描述与基本操作	20
3.1 基本数据类型	20
3.1.1 整型	20
3.1.2 实型	23
3.1.3 字符型	24
3.2 常用的运算符和表达式	27
3.2.1 赋值运算符	27
3.2.2 算术运算符	29
3.2.3 位运算符	31
3.2.4 条件运算符和逗号运算符	34
3.2.5 长度测试运算符	35
3.2.6 数值型数据的混合运算	35
3.3 表达式及赋值语句	36
3.4 基本输入输出操作的实现	36

3.4.1 基本输入输出的概念	36
3.4.2 字符、字符串数据的输入输出	37
3.4.3 格式化输入输出函数	38
3.5 顺序结构程序设计实例	44
本章小结	46
习题	46
第4章 选择结构程序设计	52
4.1 关系运算符与关系表达式	52
4.1.1 关系运算符及其优先次序	52
4.1.2 关系表达式	53
4.2 逻辑运算符和逻辑表达式	53
4.2.1 逻辑运算符及其优先次序	53
4.2.2 逻辑表达式	54
4.3 if语句	54
4.3.1 if语句的三种形式	55
4.3.2 if语句的嵌套	57
4.4 switch语句	59
4.5 选择结构程序设计实例	61
本章小结	64
习题	64
第5章 循环结构程序设计	69
5.1 循环结构的应用场合	69
5.2 while语句	69
5.3 do~while语句	70
5.4 for语句	72
5.5 多重循环	74
5.6 几种循环语句的比较	75
5.7 转移控制语句	75
5.7.1 break语句	75
5.7.2 continue语句	76
5.7.3 goto语句	77
5.8 单循环及多重循环结构的实例	77
本章小结	79
习题	79
第6章 数组与指针	85
6.1 数组	85

6.1.1 数组的概念	85	7.6.6 外部变量	138
6.1.2 数组的定义、初始化和引用	86	7.7 数组与函数参数	138
6.1.3 字符数组	91	7.7.1 数组元素作为函数实参	138
6.2 指针的概念	97	7.7.2 数组名作为函数实参	139
6.2.1 地址与指针的概念	97	7.8 指针与函数	143
6.2.2 指针的定义与引用	98	7.8.1 指向函数的指针	143
6.3 指针与数组	101	7.8.2 返回指针的函数	145
6.3.1 指向一维数组的指针	101	本章小结	146
6.3.2 指向多维数组的指针	104	习题	146
6.4 字符串的指针	107	第8章 结构体和共用体	155
6.5 指针数组和数组指针	109	8.1 结构体类型概述	155
6.5.1 指针数组	109	8.2 结构体变量的定义和引用	156
6.5.2 数组指针	111	8.2.1 结构体变量的定义	156
6.6 指向指针的指针	111	8.2.2 结构体变量的初始化	157
本章小结	112	8.2.3 结构体变量的引用	157
习题	112	8.3 结构体数组	159
第7章 函数与模块化程序设计	121	8.3.1 结构体数组的定义	159
7.1 模块化程序设计与C程序结构	121	8.3.2 结构体数组的初始化	160
7.1.1 模块化程序设计方法的指导思想	121	8.3.3 结构体数组的引用	160
7.1.2 模块分解的原则	121	8.4 结构和指针	160
7.1.3 C程序的一般结构	122	8.4.1 指向结构体变量的指针	161
7.2 函数定义与函数声明	123	8.4.2 指向结构体数组的指针	162
7.2.1 函数定义	123	8.4.3 指向结构体的指针作函数参数	162
7.2.2 函数声明	124	8.5 动态存储分配	164
7.3 参数的返回与参数传递	125	8.5.1 链表概述	164
7.3.1 函数的返回	125	8.5.2 动态存储分配的函数	164
7.3.2 形参与实参	126	8.5.3 建立链表	165
7.4 函数的调用	127	8.5.4 链表的插入操作	167
7.4.1 函数调用的一般形式	127	8.5.5 链表的删除操作	168
7.4.2 函数的传值调用	128	8.6 共用体类型	169
7.4.3 地址传送方式传递数据	129	8.6.1 共用体类型的定义	169
7.4.4 库函数的调用	130	8.6.2 共用体类型变量的特点	169
7.5 函数的嵌套与递归调用	131	8.6.3 共用体类型变量的引用方式	170
7.5.1 函数的嵌套调用	131	8.7 枚举类型	171
7.5.2 函数的递归调用	132	8.8 用typedef 定义类型	172
7.6 变量的存储类型	133	8.8.1 位域	172
7.6.1 动态存储和静态存储	133	8.8.2 typedef语句的使用	173
7.6.2 变量的作用域	133	本章小结	174
7.6.3 动态变量	136	习题	174
7.6.4 寄存器变量	136	第9章 编译预处理	178
7.6.5 局部静态变量	137	9.1 宏定义	178

9.1.1 简单宏定义	178	11.2.1 顺序查找	202
9.1.2 带参数的宏定义	179	11.2.2 折半查找	202
9.2 文件包含	182	11.3 迭代法	203
9.3 条件编译	183	11.4 递推法	205
9.4 行控制	183	11.5 穷举搜索法	206
9.5 带参数的主函数	184	11.6 递归法	210
本章小结	184	11.7 回溯法	216
习题	184	11.8 贪婪法	223
第10章 文件	187	11.9 分治法	228
10.1 C文件系统的分类	187	本章小结	229
10.2 文件的打开与关闭	188	习题	230
10.2.1 文件类型指针	188	第12章 C语言图形介绍	231
10.2.2 打开文件	188	12.1 键盘控制	231
10.2.3 关闭文件	190	12.2 鼠标控制	231
10.3 文件的读写	190	12.3 光标控制	232
10.3.1 字符输入/输出函数	190	12.4 Turbo C 的图形功能	233
10.3.2 格式输入/输出函数	191	12.5 图形方式下的文本输出函数	236
10.3.3 字符串输入/输出函数	192	12.6 常用统计图形的绘制	239
10.3.4 “记录”方式的输入和输出	193	12.7 综合应用实例	243
10.4 文件处理的其他常用函数	194	本章小结	244
10.4.1 文件的定位	194	第13章 C++介绍	245
10.4.2 出错检测	196	13.1 C++的特点	245
本章小结	196	13.1.1 C转入C++时不需改变的内容	245
习题	196	13.1.2 C转入C++时一些与类无关 新特性	246
第11章 常用程序设计方法	198	13.2 C++的核心新特性——类	248
11.1 排序及应用	198	13.2.1 类和对象	248
11.1.1 排序算法的种类	198	13.2.2 类成员的访问	249
11.1.2 冒泡排序法	199	13.2.3 构造函数和析构函数	250
11.1.3 选择排序法	200	本章小结	252
11.1.4 插入排序法	200	附录	253
11.1.5 希尔排序法	201	参考文献	266
11.2 查找	202		

第1章 C语言程序设计概述

学习计算机程序设计语言是提高人们计算机知识水平的重要步骤。C语言作为当今最为流行的程序设计语言之一，不但成为计算机专业的必修课程，而且也越来越多地成为非计算机专业的学习课程。本章首先讲述程序及程序设计的基本知识，然后介绍C语言的发展与特点，叙述C语言程序的组成与结构，阐明C语言的上机步骤和方法。

1.1 程序与程序设计

计算机通过执行程序完成其工作，如计算，控制，文字处理，图形处理，网络通信等。所谓程序，就是一组指令和数据的集合。程序就是人与机器进行“对话”的语言，该语言就是我们常说的程序设计语言。

程序设计语言分为低级语言和高级语言两大类。低级语言直接面向机器，如机器语言和汇编语言；高级语言独立于机器，用高级语言编写的程序在不同的机器上必须使用不同的翻译程序。C语言是一种高级语言，它只有被翻译成计算机能识别的语言（即机器语言）才能在计算机上运行。

计算机只能理解机器语言，不能理解汇编语言和高级语言，所以必须把用汇编语言或者高级语言编写的程序“翻译”成机器语言计算机才能执行其程序。把汇编语言翻译成机器语言的过程称为“汇编”。把高级语言“翻译”成机器语言有两种方式：一种是“解释”，一种是“编译”。而C语言属于编译型语言。编译的过程就是由编译程序把源程序编译、链接成可执行文件，然后由机器直接运行可执行文件，具体关系如图1-1所示。

C语言源程序的编译过程：首先把C语言源程序编辑输入到计算机中，随后调用编译程序对源程序进行编译，之后产生目标文件，再调用链接程序进行链接，之后产生可执行文件，最后由机器直接运行可执行文件。

程序员编写的源程序必须遵循编译程序规定的语法。语法是编写程序的规则。各种类型的语法规则产生了C、PASCAL、BASIC、FORTRAN等语言。C语言语法标准主要由美国国家标准协会（ANSI）制定。

程序设计所需的开发环境（称为软件条件）如图1-2所示。

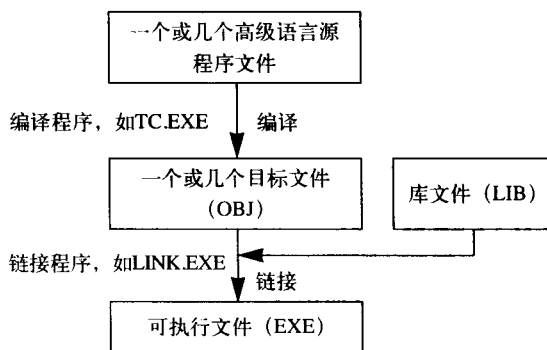


图1-1 C语言源程序的编译过程

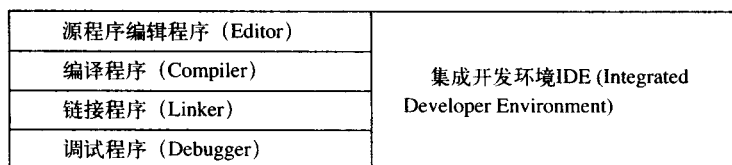


图1-2 程序开发环境

程序开发环境中，各项程序的功能如下：

- 源程序编辑程序：编程者输入源程序的编辑工具。
- 编译程序：把源程序翻译成目标程序。
- 连接程序：把目标程序及有关的库函数连接成机器可执行的程序文件。
- 调试程序：帮助编程者定位错误和改正错误的软件工具。

程序开发环境的核心是编译程序，它把程序员编写的类自然语言源程序翻译为机器指令，同时，从应用的角度，它提供程序设计思想。主要的程序设计思想有结构化程序设计思想和面向对象程序设计思想。几种典型的程序开发环境及其特点如图1-3所示。

Microsoft	Borland(Inspires)	特点
MSC	Turboc	结构化程序设计 开发DOS程序
Visual C++	C++ Builder	面向对象程序设计 可视化程序设计环境 专业化Windows 9X/NT程序
Visual Basic	Delphi (类PASCAL语法)	面向对象程序设计 可视化程序设计环境 快速设计Windows 9X/NT程序

图1-3 程序开发环境及其特点

程序设计的过程如下：

- 1) 问题的提出，要达到的要求，即明确要解决的问题。
- 2) 确定数据结构和所采用的算法，即求出解决问题的方法和思路。
- 3) 编制程序，即用程序设计语言表达出解决问题的步骤。
- 4) 调试程序，即找出程序中的错误并改正，使得程序能达到预期的目的。
- 5) 整理并写出文档，即记载解决问题的过程及程序的输入与输出的结果等。

1.2 C语言简介

1.2.1 C语言的发展历程

C语言是一种流行的高级程序设计语言。使用C语言既可以编写应用软件，也可以编写系统软件。

虽然，以前都是使用汇编语言编写操作系统，但是，汇编语言依赖特定的计算机硬件，可移植性很差，而且汇编语言编写的程序又难以读懂，所以，人们就想用一种高级语言来编写系统软件，但在高级语言中却很难实现汇编程序对硬件的操作能力，C语言应运而生。

C语言是在1972年由美国的Dennis Ritchie等人设计发明，并首次在UNIX操作系统的DEC PDP-11计算机上使用。它由早期的编程语言（Basic Combind Programming Language, BCPL）发展演变而来。在1970年，AT&T 贝尔实验室的 Ken Thompson根据BCPL语言设计出较先进的B的语言，在此基础上，C语言问世。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，美国国家标准协会（ANSI）为C语言制定了一套ANSI标准，成为现行的C语言、语法标准。

1.2.2 C语言的优点和规则

1. C语言的优点

人们之所以要采用C语言编写各种各样的程序，是因为C语言有许多独特的优点：

1) C语言简洁，灵活。

C语言的关键字简练，源程序短，输入的工作量比较少。采用C语言编程可以使程序员专注算法设计，而不必过多的考虑格式限制。

2) C语言具有丰富的运算符，运算符的灵活运用会使源程序精练，生成的代码质量高，运行速度快。

3) C语言数据类型丰富，能实现各种复杂的运算，尤其是指针类型数据，使程序更加灵活、多样。

4) C语言语法限制不是很严格。程序格式书写自由，不像FORTRAN那样有严格的格式。例如，C语言对数组下标越界不作检查，由程序员保证程序的正确性。同时，对变量类型的使用比较灵活，例如，整型与字符型及逻辑型数据可以互相通用。

5) C语言可以直接访问物理地址和计算机硬件，能进行位操作，也可以实现汇编语言的很多功能。因此，C语言具有高级语言和低级语言的双重功能，可以用来编写系统软件。

6) 用C语言编写的程序可移植性好，一般不作修改或者作少量的修改就能在不同的计算机和不同的操作系统上运行。

读者在编写C语言程序，特别是在与其他语言对比时，会对以上的优点有更加深入的体会。

2. C语言的规则

1) C语言是结构化程序设计语言。

2) C语言程序采用模块化设计风格，程序由许多函数组成。采用C语言编写的程序必须有一个main()的主函数，而且只能有一个主函数，“{”和“}”分别表示函数的起点和终点，相当于PASCAL的BEGIN...END。函数之间可以相互调用、递归调用。但一般函数不能调用主函数。

3) C语言程序可以调用其他文件的函数，这样，一个C语言程序可以由许多文件组成，更便于合作开发。

4) 组成C语言的语句可以放在一行，也可以放在多行；C语言程序的一行也可由多个语句组成。而且，各语句之间都要以“；”作为结束标志。为了便于阅读，编写C语言的程序应遵循一定的规则，例如，嵌套循环时应该有缩行。

5) 为便于维护和理解C语言程序，在C语言的关键语句后应该有注释，注释部分必须用“/*”和“*/”括起来，并且“/”和“*”之间不能有空格。编译程序在编译时忽略“/*”和“*/”之间的内容。

6) C语言的程序一般要有头文件，头文件在程序的开始用“#include”做出说明。头文件可以是程序中所用变量的说明，也可以是引用的库函数说明。

7) C语言区分大小写，因此，在使用C语言时应特别注意。

8) C语言的程序总是从主函数开始执行，并且终止于主函数。

总之，C语言灵活性大，功能强，可以编写出各种类型的程序。程序员使用C语言可以自由地编程。但是，从学习语言的角度来说，学习C语言比学习其他高级程序设计语言要难一些。这些困难大部分来自C语言的灵活性，只要掌握C语言的基本语法规则，多上机练习，C语言还是可以学好的。

1.3 简单的C语言程序

下面以“Hello World!”作为本书的开始程序:

【例1-1】 最简单的C语言程序。

```
main()
{
    printf("Hello World!\n");
}
```

程序执行后在屏幕上输出的结果如下:

Hello World!

其中, main表示主函数。每个C语言程序都必须有一个main函数。函数体由花括号{ }括起来。本例中主函数内只有一个输出语句, printf是C语言中的输出函数(详见第3章)。双引号内的字符串按原样输出, "\n"是换行符, 即在输出“Hello World!”后回车换行, 语句最后以分号结束。

【例1-2】 两个数求和。

```
#include <stdio.h>
main()
{
    int a,b,sum; /*定义变量 */
    a=3; /*给变量a赋值 */
    b=6; /*给变量b赋值 */
    sum=a+b; /*求a、b的和并赋给变量sum */
    printf("sum=%d\n",sum); /*输出运算的结果 */
}
```

程序的输出结果为:

sum=9

例1-2中, 程序先定义三个变量(变量是存放数据的存储单元) a、b和sum, 然后分别给变量a和变量b赋值, 变量sum是变量a、变量b的和, 程序输出的结果正是a与b的和。“%d”表示输出十进制整数。

书写C语言程序时应遵循的规则

从书写清晰、便于阅读、理解和维护的角度出发, 书写C语言程序时最好遵循以下规则:

- 1) 一个说明或一个语句单独占一行。
- 2) 用{}括起来的部分, 通常表示程序的某一层结构。{}一般与该结构语句的第一个字母对齐, 并单独占一行。
- 3) 每一层次的语句(说明)可在上一层次语句(说明)的基础上, 再缩进若干空格后书写, 以便使程序看起来更加清晰, 增加其可读性。

在编程时应力求遵循这些规则, 以养成良好的编程风格。

1.4 C语言程序的上机步骤

C语言是编译型语言, 源程序必须经过编译才能在计算机上执行。C语言程序的上机步骤大致如下:

- 1) 上机输入与编辑源程序。
- 2) 对源程序进行编译。
- 3) 与库函数连接。

4) 运行可执行的目标程序。

下面介绍用Turbo C运行C语言程序的具体步骤。

1) 在Turbo C所在的目录下直接键入TC, 就可以进入C语言程序的集成开发环境, 如图1-4所示。

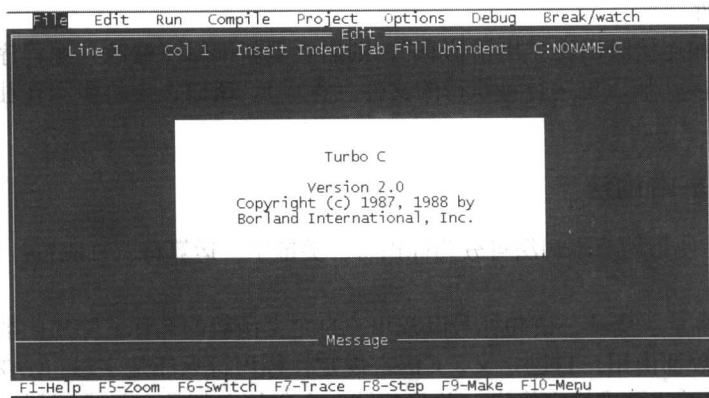


图1-4 C语言程序的集成开发环境

在Turbo C集成开发环境中, 有一行主菜单, 其中包括“File、Edit、Run、Compile、Project、Option、Debug和Break/watch”8个菜单项。

编程者可以通过以上菜单项来选择使用Turbo C集成环境所提供的各项主要功能。以上8个菜单项分别代表文件操作、编辑、运行、编译、项目、选项、调试、中断/观察等功能。按“F10”键就可选中某主菜单项, 随后可用“←”键和“→”键移动光标来选择所需要的菜单项。一旦选定了主菜单项, 再按回车键就可打开下级菜单。

2) 按Alt+F组合键可以下拉File菜单, 如图1-5所示。File菜单用来对文件进行操作, 包括装载文件, 建立新文件, 存储文件等。

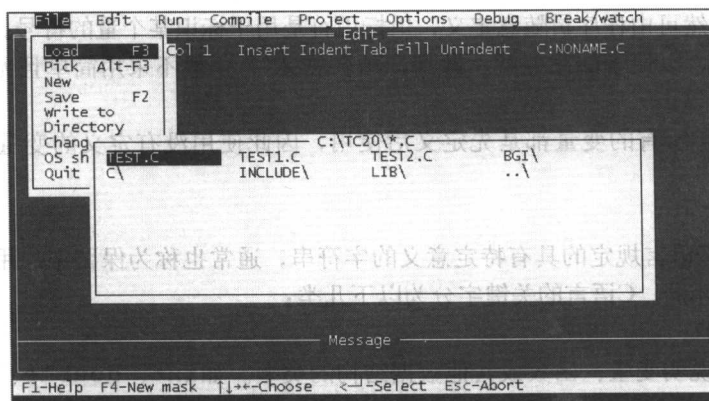


图1-5 File菜单

3) 在图1-5中, 利用光标选择“New”命令, 建立一个新文件, 然后集成开发环境就打开编辑窗口, 读者可以在此输入C语言源程序。

4) 读者编辑完源程序后, 要对源程序进行编译, 生成可执行文件方可执行。按Alt+C组合键, 选择“Compile to OBJ”, 生成目标文件, 然后选择“Link EXE file”链接目标文件。或者直接选择“Make EXE file”生成可执行文件。如果源程序没有语法和语义错误,

就可以生成后缀为“.exe”的可执行文件。如果源程序存在错误，那么集成开发环境会指出错误的行，读者可以打开编辑窗口修改源程序。

5) 生成可执行文件后，按Alt+R组合键，选择“Run”，或者直接按Ctrl+F9组合键，可以执行此文件。也可以退出集成环境，在可执行文件所在的目录下，直接输入可执行文件名称，也可以执行该文件。

如果在编译和连接时出现“出错信息”，则需要重新编辑（修改）源程序。修改后仍然需要进行编译和连接，最后再运行可执行的文件（程序）。所以，上机步骤往往是一个循环往复的过程。

1.5 C语言的基本词法

在C语言中，使用的基本词法可分为标识符、关键字、运算符、分隔符、常量、注释符等。

1. 标识符

标识符是指常量、变量、语句标号以及用户自定义函数的名称。除库函数的函数名由系统定义外，其余函数都由用户自己定义。C语言规定，标识符只能是字母(A~Z, a~z)、数字(0~9)、下划线(_)组成的字符串，并且其第一个字符必须是字母或下划线。例如，以下标识符是合法的：a, x, x3, BOOK_1, sum5, _x7。以下标识符是非法的：3s (以数字开头)、s*T (出现非法字符*)、-3x (以减号开头)、bowy-1 (出现非法字符-)。

在使用标识符时还必须注意以下几点：

1) 标准C不限制标识符的长度，但标识符受各种版本的C语言编译系统限制，同时也受到具体机器的限制。例如，在某版本C中规定标识符前8位有效，当两个标识符前8位相同时，则被认为是同一个标识符。Turbo C规定标识符的长度为32。在编写程序时，应了解系统规定的标识符的长度，以免造成不必要的错误。编译系统不会发现这种错误，所以应特别小心。一般情况下，标识符的长度不要超过8个字符。

2) 在标识符中，大小写是有区别的。例如SUM和sum是两个不同的标识符。变量名应尽量使用小写字母，以增加程序的可读性。

3) 标识符虽然可由程序员随意定义，但标识符是用于标识某个量的符号。因此，命名应尽量有相应的意义，以便于阅读理解，做到“顾名思义”，一般不采用简单且无意义的符号作为变量名，如a、b等。

在C语言中，所有的变量都是先定义后使用，因此使用没有定义的变量名被认为是“非法”的。

2. 关键字

关键字是由C语言规定的具有特定意义的字符串，通常也称为保留字。用户定义的标识符不应该与关键字相同。C语言的关键字分为以下几类：

(1) 类型说明符

用于定义和说明变量、函数或其他数据结构的类型。如int、double、float、long、short、auto、signed、static、struct、unsigned、char、enum、extern、register和union等。

(2) 语句定义符

用于表示一个语句的功能。如条件语句的语句定义符if~else，循环语句的语句定义符do、while、for和goto等。

(3) 预处理命令字

用于表示一个预处理命令，使用时前面要加“#”。如include、define、ifdef、

ifndef、undef和endif等。

关键字后必须有空格、圆括号、尖括号、双引号等分隔符，否则就会与其他字符一起组成新的标识符。

3. 运算符

C语言具有相当丰富的运算符。运算符与变量、函数一起组成表达式，表示各种运算功能。运算符由一个或多个字符组成。

4. 分隔符

在C语言中，最常用的分隔符有逗号和空格两种。逗号主要用在类型说明和函数参数表中分隔各个变量。空格多用在语句各单词之间作间隔符。关键字和标识符之间必须要有一个以上的空格作间隔符，否则将会出现语法错误。例如把int a写成inta后，C编译器会把inta当成一个标识符处理，其结果必然出错。

5. 常量

C语言中使用的常量可分为数字常量、字符常量、字符串常量、符号常量、转义字符等多种。后续章节将专门介绍常量的相关内容。

6. 注释符

C语言的注释符是以“/*”开头并以“*/”结尾的字符串。在“/*”和“*/”之间内容即为注释内容。程序编译时，不对注释作任何处理。注释可出现在程序中的任何位置。注释用来向用户提示或解释程序的意义。在调试程序中对暂不使用的语句也可用注释符括起来，使编译跳过不作处理，待调试结束后再去掉注释符。

7. C语言的字符集

字符是组成语言的最基本的元素。C语言字符集由字母、数字、空格、标点和特殊字符组成。在字符常量，字符串常量和注释中还可以使用汉字或其他可表示的图形符号。

- 字母 小写字母a~z共26个，大写字母A~Z共26个。
- 数字 0~9共10个。
- 空白符 空格符、制表符、换行符等统称为空白符。空白符只在字符常量和字符串常量中起作用。在其他地方出现时只起间隔作用，编译程序会忽略它们。因此，在程序中使用空白符与否，对程序的编译不发生影响，但在程序中适当的地方使用空白符将增加程序的清晰性和可读性。

1.6 C语言程序的基本结构

C语言程序设计的基本结构可分为顺序结构、分支结构和循环结构三种。按照结构化程序设计的观点，任何功能的程序都可以通过这三种基本结构的组合来实现。

1. 顺序结构

如图1-6所示，模块A和B是顺序执行的。即只有在执行完A的操作之后，才能执行B的操作。顺序结构是最简单的一种基本结构。

我们可以把模块A和模块B合并成一个模块，但无论怎样合并，生成的新模块仍然是一个整体。模块的所有语句只能从模块的顶部进入，全部执行完之后，再从模块底部退出。这也是模块的基本性质。

2. 分支结构

当根据逻辑条件成立与否，分别选择执行不同的程序模块时，可以使用分支结构，也称选择结构。

图1-7即是一个分支结构的流程图。当满足判断条件时,执行模块A;当不满足判断条件时,执行模块B。在程序的一次执行中,模块A和模块B只可能执行其中的一个。在实际的应用中,允许其中一个模块为空,如图1-8所示。

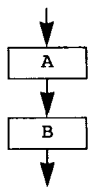


图1-6 顺序结构

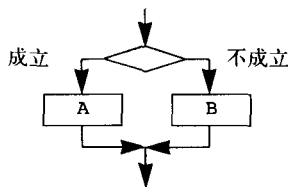


图1-7 分支结构（一）

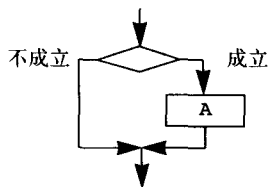


图1-8 分支结构（二）

3. 循环结构

循环结构是程序中的另一种重要结构,它和顺序结构、分支结构共同作为各种复杂程序的基本构造部件。循环结构的特点是在给定条件成立时,反复执行某个程序段。通常称给定条件为循环条件,称反复执行的程序段为循环体。循环结构一般分成两种情形,一种是当型循环,一种是直到型循环。在程序执行时,当型循环首先判断条件是否成立,条件成立时,执行模块A中的程序,然后再判断条件,如条件成立,再执行模块A的程序,这样循环往复,如图1-9所示。一旦条件不成立时,程序就会退出循环模块,执行后继的程序。

直到型循环首先执行一次模块A中的程序,然后判断条件是否成立,如果条件不成立,则继续执行模块A中的程序,然后再判断,如此往复,如图1-10所示。直到所给的条件成立时,才退出循环程序。

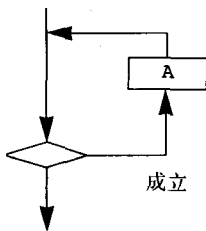


图1-9 当型循环

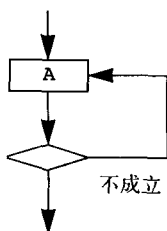


图1-10 直到型循环

在有些问题中,循环模块都是要执行的,至少应该被执行一次,这时当型循环和直到型循环没有什么区别,无论使用当型循环还是直到型循环都能很好地解决问题。但是,在其他一些问题中,循环模块有可能一次也不会被执行,这时就应该使用当型循环。一般说来,不用刻意地去设计某个问题是当型循环还是直到型循环,具体问题应具体分析。

在循环的使用中,应避免死循环,即无限地执行循环模块,永远不会退出。这样的程序在实际应用中是没有任何意义的。在设计循环条件时,应特别注意,要保证循环在有限的时间内退出。

其实,基本结构不是只限于上述三种。但是,已经证明,由以上三种基本结构组成的程序,可以完成任何复杂的问题。三种基本结构所构成的程序属于结构化的程序,它不存在无规律的向前或向后的转向,在本结构内可以用转向语句进行跳转。

本章小结

在计算机的发展历史上,依次经历了机器语言、汇编语言和高级语言的阶段。C语言属于

一种高级语言，同时，它还具有某些低级语言的功能，所以有人也把C语言称作“中间语言”。C语言是随着UNIX操作系统的发展而流行起来的，而C语言能成为一种流行很广的语言，也有它自身的优点。本章简单介绍C语言的一些简单语法和C语言程序的基本结构，目的是让读者对C语言有一个大致的了解。

习题

一、选择题

1. C语言程序是由()组成的。
A) 子程序 B) 过程 C) 函数 D) 主程序和子程序
2. C语言中的标识符只能由字母、数字和下划线三种字符组成，且第一个字符()。
A) 必须为字母 B) 必须为下划线
C) 必须为字母或下划线 D) 可以是字母、数字和下划线中任一字符
3. 以下叙述正确的是()。
A) 在C程序中，main函数必须位于程序的最前面
B) C程序的每行中只能写一条语句
C) C语言本身没有输入输出语句
D) 在对一个C程序进行编译的过程中，可发现注释中的拼写错误
4. 下面是合法C语言标识符的是()。
A) 3AT_L B) _M+N C) -M2 D) AC_7
5. 下面不是合法C语言标识符的是()。
A) 3S_L B) BOOK_N C) _A47 D) A4
6. 下面是合法C语言关键字的是()。
A) For B) Do C) *FILE D) goto
7. 下面不是合法C语言关键字的是()。
A) INT B) double C) short D) struct
8. C语言源程序的文件扩展名为()。
A) .obj B) .out C) .C D) .exe
9. 下面叙述中正确的是()。
A) C语言的源程序不必通过编译就可以直接运行
B) C语言中的函数不可以进行单独编译
C) C源程序经编译形成的二进制代码可以直接运行
D) C语言中的每条可执行语句最终都将被转换成二进制的机器指令
10. 下列叙述中，正确的是()。
A) C程序中的所有字母都必须小写
B) C程序中的关键字必须小写，其他标识符不区分大小写
C) C程序中的所有字母都不区分大小写
D) C语言中的所有关键字必须小写

二、填空题

1. C源程序的基本单位是_____。