

计算机专业考研指导丛书

C 程序设计

考点精要与解题指导

李春葆 刘 斌 编著

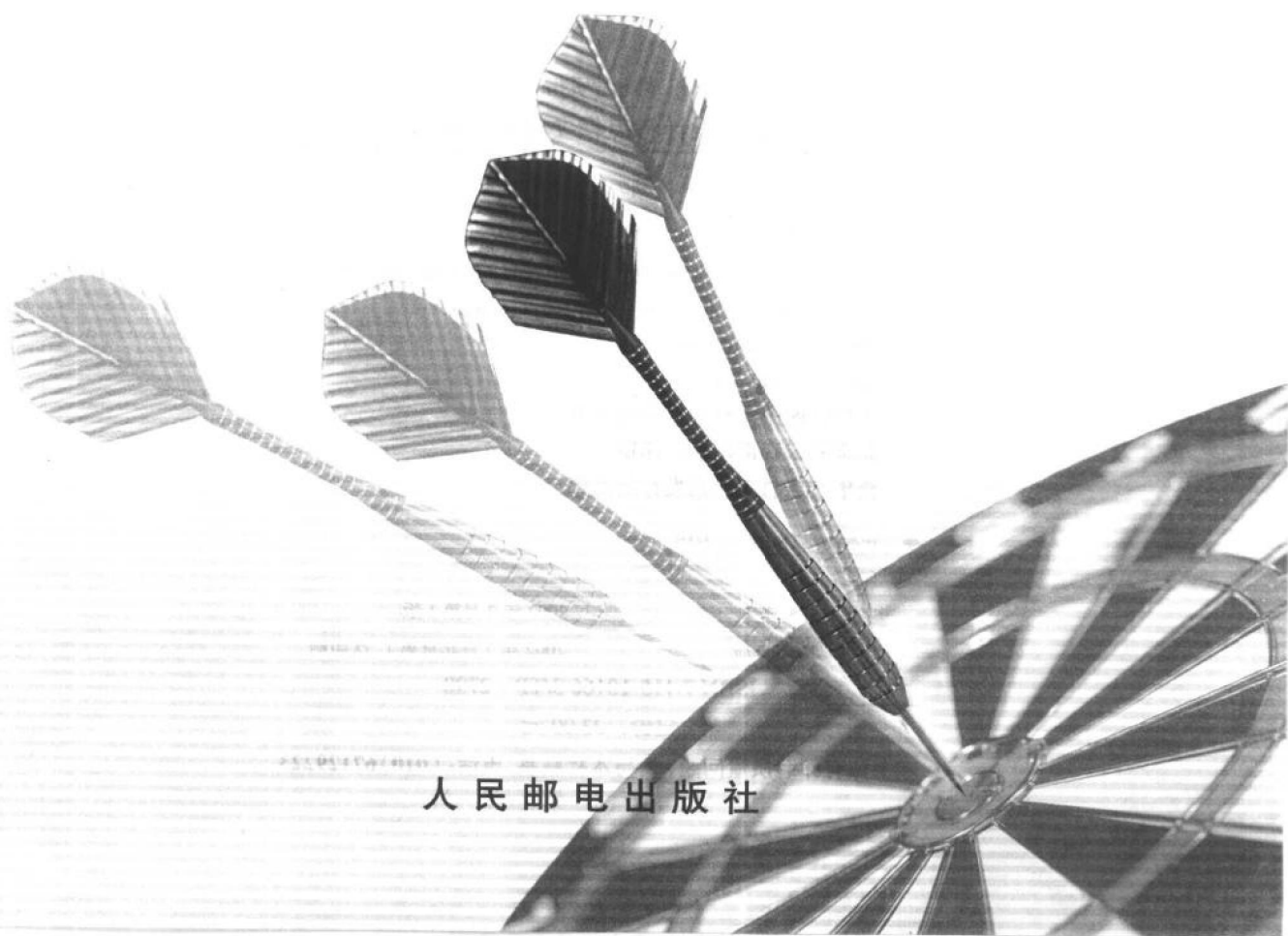
人民邮电出版社
POSTS & TELECOMMUNICATIONS PRESS

计算机专业考研指导丛书

C 程序设计

考点精要与解题指导

李春葆 刘 斌 编著



人民邮电出版社

图书在版编目 (CIP) 数据

C 程序设计考点精要与解题指导/李春葆, 刘斌编著; —北京: 人民邮电出版社, 2002.7
(计算机专业考研指导丛书)

ISBN 7-115-10166-3

I.C... II.①李...②刘... III.C 语言—程序设计—研究生—入学考试—自学参考资料
IV.TP3—44

中国版本图书馆 CIP 数据核字 (2002) 第 046959 号

计算机专业考研指导丛书 C 程序设计考点精要与解题指导

- ◆ 编 著 李春葆 刘 斌
责任编辑 邓革浩
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67180876
北京汉魂图文设计有限公司制作
北京密云春雷印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 16
字数: 385 千字 2002 年 7 月第 1 版
印数: 1-6 000 册 2002 年 7 月北京第 1 次印刷

ISBN 7-115-10166-3/TP · 2798

定价: 23.00 元

本书如有印装质量问题, 请与本社联系 电话: (010)67129223

内容提要

目前 C 语言是各大专院校计算机专业的核心课程,也是很多高校招收计算机专业研究生考试的科目之一。

本书是针对考研者编写的,书中高度概括和总结了 C 语言的基本考点,收集了大量的研究生入学考试试题并给出了分析和解答。全书共分为 9 章,每章由三部分构成,即考点精要、例题解析、自测题及参考答案。考点精要部分高度概括了本章考试内容及注意要点;例题解析部分详尽地解答了精选的考研试题,各题都包含有相关知识、例题分析和例题答案;自测题及参考答案收集了大量的相关试题并给出了相应的参考答案。

本书的特点是概念清晰,文字简洁明了,解题思路完整,极便于考研者短时间内掌握解题要点,提高考试成绩。

本书适合于考研者应试复习,同样也适合于作为大专院校各专业 C 语言程序设计课程的复习参考书,还可供计算机软件水平考试者研习。

计算机专业是当今最热门也是发展最迅速的学科之一，很多学生为了进一步提高专业水平和应用能力，纷纷报考计算机专业研究生。据统计，近几年报考计算机软件与理论、计算机应用和计算机与通信专业硕士研究生的考生远远超过报考其他专业的考生，其中有相当一部分考生原来所学专业并非计算机专业，还有很多考生是工作多年的在职人员。为了方便报考者复习计算机专业课程，我们特地组织一批计算机专业教学第一线的教授和副教授（其中大多数编写者多年参加硕士研究生入学试题命题工作）编写了本丛书。本丛书包含如下课程：

- 《C 程序设计考点精要与解题指导》
- 《离散数学考点精要与解题指导》
- 《数据结构考点精要与解题指导》
- 《操作系统考点精要与解题指导》
- 《编译原理考点精要与解题指导》
- 《计算机组成原理考点精要与解题指导》

本丛书具有以下特点：

◎ 讲述全面而详实

本丛书涵盖各门专业课程的内容，不是针对个别学校的命题特点，而是充分地讲授课程中的重点、难点和考点，并通过例题进行扩充与深化，使读者得以全面温习，不留“死角”。

◎ 阐述简洁而明了

不同于本、专科教材，本丛书的目的是使考生花较少的时间温习各门课程的内容，因此，不过多地解释简单的术语，只对基本知识点进行高度概括和总结，使读者将主要精力花在解题过程中。

◎ 重点突出解题思路

本丛书重点介绍解题的方式和方法，不仅授人以“鱼”，更在于授人以“渔”，选择的例题和习题大多是计算机专业研究生入学考试试题（题目前标有“★”号），并配上详解，具有很强的实战性。

◎ 强调内容的综合与提高

一般的教科书多是按照内容的先后顺序按部就班地介绍，这种方式有助于初学者学习，但不便于复习和综合，因为考研题一般都具有很强的综合性，往往一个题涉及好几章的概念，所以本丛书打破了一般教科书的教学模式，将相关的概念有机地融为一体，从而提高考生的解题能力。

◎ 答疑解惑

本丛书选择的例题和习题大部分具有较高的难度，书中不仅给出了答案，而且详细介绍了解题思路和解题过程，有助于考生纠正以往的概念误区。

本丛书希望在考研指导方面作一些探索和尝试，起到抛砖引玉的作用，书中的不妥之处敬请广大的读者和同行指教。

李春葆

2002.6

前 言

C 语言是一种基础性程序设计语言，它不仅具有表达能力强、代码质量高和可移植性好等特点，并且极好地支持结构化、模块化等软件工程开发方法，更兼备高级语言和低级语言的许多优点。目前，C 语言已成为各大专院校计算机专业的核心课程，也是很多高校招收计算机专业研究生考试的科目之一。

本书共分为 9 章，第 1 章介绍概述；第 2 章介绍数据类型及其运算；第 3 章介绍选择语句和循环语句；第 4 章介绍数组，第 5 章介绍指针；第 6 章介绍函数；第 7 章介绍结构体和共用体；第 8 章介绍预编译处理；第 9 章介绍文件。

每章由三部分构成，即考点精要、例题解析、自测题及参考答案。考点精要部分高度概括了本章考试内容及注意要点；例题解析部分详尽地解答了精选的考研试题，各题都包含有相关知识、例题分析和例题答案（少数比较简单的例题没有给出例题分析）。自测题及参考答案收集了大量的相关试题并给出了相应的参考答案。

书中涵盖了大量的研究生入学考试试题（前面加有“★”号）和部分计算机软件水平考试题（前面加有“☆”号）。

本书的特点是概念清晰，文字描述简洁明了，所有题目都给出了详细的解答，极便于读者短时间内掌握解题要点。

尽管本书主要针对考研者应试复习和提高，但同样也适合于作为大专院校各专业 C 语言程序设计的复习参考书，还可供计算机软件水平考试者研习。

由于作者水平有限，书中难免存在不足之处，敬请有关专家和广大读者不吝指正。

编者
2002.6

目 录

第1章 概 述	1
1.1 考点精要.....	1
1.1.1 C语言的特点.....	1
1.1.2 C语言标识符.....	1
1.1.3 C语言的程序组成.....	2
1.1.4 C语言程序的开发过程.....	3
1.2 例题解析.....	3
1.3 自测题及参考答案.....	4
第2章 数据类型及其运算	7
2.1 考点精要.....	7
2.1.1 基本数据类型.....	7
2.1.2 变量和常量.....	8
2.1.3 运算符.....	10
2.1.4 类型转换.....	14
2.1.5 数据输入与输出.....	15
2.2 例题解析.....	17
2.3 自测题及参考答案.....	21
第3章 选择语句和循环语句	31
3.1 考点精要.....	31
3.1.1 选择语句.....	31
3.1.2 循环语句.....	34
3.1.3 穷举法.....	36
3.2 例题解析.....	38
3.3 自测题及参考答案.....	49
第4章 数 组	61
4.1 考点精要.....	61
4.1.1 数组说明和引用.....	61
4.1.2 字符数组和字符串数组.....	63
4.1.3 数组的排序.....	64
4.1.4 数组的查找.....	67
4.2 例题解析.....	68
4.3 自测题及参考答案.....	79
第5章 指 针	98
5.1 考点精要.....	98
5.1.1 指针的概念及其运算.....	98



5.1.2	指针和数组	99
5.1.3	字符指针和字符串	100
5.1.4	指针数组	101
5.1.5	多级指针	101
5.1.6	数组指针	102
5.2	例题解析	102
5.3	自测题及参考答案	111
第 6 章	函 数	123
6.1	考点精要	123
6.1.1	函数的定义和使用	123
6.1.2	函数的存储类型	125
6.1.3	变量的作用域	125
6.1.4	变量的存储类型	125
6.1.5	函数的数据传递	127
6.1.6	数组在函数间的传递	128
6.1.7	指针型函数	128
6.1.8	指向函数的指针	129
6.1.9	递归函数及其设计方法	129
6.1.10	命令行参数	130
6.2	例题解析	131
6.3	自测题及参考答案	144
第 7 章	结构体与共用体	169
7.1	考点精要	169
7.1.1	结构体说明和变量定义	169
7.1.2	结构体数组	171
7.1.3	结构体指针	172
7.1.4	函数之间结构体变量的数据传递	172
7.1.5	结构体的应用: 链表和二叉树	172
7.1.6	共用体	173
7.1.7	枚举类型	174
7.1.8	用户定义类型	174
7.2	例题解析	175
7.3	自测题及参考答案	190
第 8 章	预编译处理	200
8.1	考点精要	200
8.1.1	宏	200
8.1.2	条件编译	200
8.1.3	文件包含	201
8.2	例题解析	202

8.3 自测题及参考答案	204
第9章 文 件	208
9.1 考点精要	208
9.1.1 文件概述	208
9.1.2 文件的输入输出	209
9.1.3 文件的定位操作	212
9.2 例题解析	213
9.3 自测题及参考答案	228

第1章 概述

基本知识点：C 语言程序组成及相关概念。

重点：C 程序结构和 C 程序的执行过程。

难点：C 语言程序的编译过程。

1.1 考点精要

1.1.1 C 语言的特点

C 语言的特点可大致归纳如下：

(1) C 语言短小精悍，基本组成部分精炼、简洁。C 语言一共只有 32 个标准的关键字，45 个标准的运算符以及 9 种控制语句，不但语言的组成精炼、简洁，而且使用方便、灵活。

(2) C 语言运算符丰富，表达能力强。C 语言具有“高级语言”和“低级语言”的双重特点，其运算符包含的内容广泛，所生成的表达式简练、灵活，有利于提高编译效率和目标代码的质量。

(3) C 语言数据结构丰富，结构化好。C 语言提供了编写结构化程序所需要的各种数据结构和控制结构，这些丰富的数据结构和控制结构以及以函数调用为主的程序设计风格，保证了利用 C 语言所编写的程序能够具有良好的结构化。

(4) C 语言提供了某些接近汇编语言的功能，有利于编写系统软件。C 语言提供的一些运算和操作，能够实现汇编语言的一些功能，如它可以直接访问物理地址，并能进行二进制位运算等，这为编写系统软件提供了方便条件。

(5) C 语言程序可移植性好。在 C 语言所提供的语句中，没有直接依赖于硬件的语句，与硬件有关的操作，如数据的输入、输出等都是通过调用系统提供的库函数来实现的，而这些库函数本身并不是 C 语言的组成部分，因此，用 C 语言编写的程序能够很容易地从一种计算机环境移植到另一种计算机环境中。

1.1.2 C 语言标识符

在 C 语言中，标识符是一个名字，C 语言允许用作标识符的字符有：

(1) 26 个英文字母，包括大小写（共 52 个）；

(2) 数字 0, 1, …, 9；

(3) 下划线（_）。

标识符的构成规则为：

- (1) 必须由字母 (a~z, A~Z) 或下划线开头;
- (2) 后面可以跟随任意的字母、数字或下划线。

C 语言的标识符可以分为以下 3 类。

(1) 关键字: C 语言规定了一批标识符, 它们在程序中都代表着固定的含义, 不能另作它用。例如, 用来说明变量类型的标识符 `int`、`float` 等, 它们不能再用作变量名或函数名。

(2) 预定义标识符: 这些标识符在 C 语言中也都有特定的含义, 如 C 语言提供的库函数的名字 (如 `printf`) 和预编译处理命令 (如 `define`) 等。因此为了避免误解, 建议用户不要把这些预定义标识符另作它用。

(3) 用户标识符: 由用户根据需要定义的标识符称为用户标识符, 一般用来给变量、函数、数组或文件等命名。



注意

有关标识符的注意事项如下:

(1) 在 C 语言中, 大小写字母有不同的含义, 例如: `num`, `Num`, `NUM` 为 3 个不同的标识符。

(2) 在构造标识符时, 应注意做到“见名知意”, 即选有含意的英文单词 (或汉语拼音) 作标识符, 以增加程序的可读性, 如表示年可以用 `year`, 表示长度可用 `length`, 表示和可以用 `sum` 等。

1.1.3 C 语言的程序组成

1. C 程序的构成

一个 C 程序的构成如下:

- (1) 一个 C 源程序由函数构成, 其中至少包括一个主函数 (`main` 函数);
- (2) C 程序总是由 `main` 函数开始执行;
- (3) 分号 “;” 是 C 语句的一部分, 它是语句之间的分隔符;
- (4) C 程序中可以包含常量、变量、运算和标识符;
- (5) C 程序书写格式自由, 一行内可写多条语句, 且语句中的空格和回车符均可忽略不计;
- (6) 注释用来向用户提示或解释程序的意义, C 程序的注释部分应括在 “/*” 与 “*/” 之间, 在 / 和 * 之间不允许留有空格。注释部分允许出现在程序的任何位置, 程序编译时, 忽略所有的注释符, 不对它们作任何处理。

2. C 语言的程序结构

C 语言是一种结构化程序设计语言。它直接提供了 3 种基本结构的语句; 提供了定义函数的功能; 提供了丰富的数据类型。结构化程序由下述 3 种基本结构组成。

(1) 顺序结构: 一组按书写顺序执行的语句。这种结构的控制流顺次从一个处理过程转向下一个处理过程, 比如从一个语句 `a` 转向紧接着的下一个语句 `b`, 从整体上看, `a` 和 `b` 两个语句操作步骤之间就是一个顺序执行关系。

(2) 选择结构: 当执行到 `if` 语句、`switch` 语句时都可构成选择结构。当执行到这些语句时, 先计算条件, 然后根据条件表达式值的真假, 选择相应的处理执行。

(3) 循环结构: 当执行到 `while`、`for`、`do` 等语句时都可构成循环结构。当执行到这些语

句时, 根据条件使一组语句重复执行多次或一次也不执行。

循环结构常用的有两种形式:

- **while 型循环结构:** 当条件为真时, 反复执行循环体, 直到条件为假, 即先判断重复执行的结构条件, 后执行循环体。
- **do-while 型循环结构:** 反复执行循环体, 直至条件为假时, 结束重复操作, 即先执行循环体, 后判断循环执行的结束条件。

任何 Goto 结构都可以等价地转换成上述 3 种结构。

1.1.4 C 语言程序的开发过程

开发一个 C 语言程序的基本过程如下。

(1) **编辑:** 选择适当的编辑程序, 将 C 语言源程序通过键盘输入到计算机中, 并以文件的形式存入到磁盘中, 如在 Turbo C 系统下, 经过编辑后得到的源程序文件都是以“.C”为其文件扩展名; 在 Visual C++ 系统下, 经过编辑后得到的源程序文件都是以“.cpp”为其文件扩展名。

(2) **编译:** 通过编辑程序将源程序输入到计算机后, 需要经过 C 语言编译器将其生成目标程序。在对源程序的编译过程中, 可能会发现程序中的一些语法错误, 这时就需要重新利用编辑程序来修改源程序, 然后再重新编译。经过编译后得到的目标文件都是以“.obj”为其文件扩展名。

(3) **连接:** 经过编译后生成的目标文件是不能直接执行的, 它需要经过连接之后才能生成可执行的代码。连接后所得到的可执行文件都是以“.exe”为其文件扩展名。

(4) **执行:** 经过编译、连接之后, 源程序文件就可以生成可执行的文件, 这时就可以执行了。在 DOS 系统下, 只要键入可执行的文件名, 并按回车键后, 就可执行文件了; 在 Windows 下, 通过双击可执行的文件名, 就可执行文件了。

1.2 例题解析

例 1.1 简述 C 程序的组成和执行过程。

【相关知识】 C 程序的组成、C 程序的执行过程。

【例题答案】 一个 C 程序是由若干个函数组成的, 其中必须有一个且只有一个主函数 main()。一个 C 程序的执行是从本程序的 main 函数开始, 到 main 函数结束。在执行过程中, 若遇到调用函数, 则转向该函数去执行, 然后当函数返回时再从它原来的地方继续执行。

例 1.2 分析以下叙述中哪些是不正确的:

- (1) 一个 C 程序可由一个或多个函数组成。
- (2) 一个 C 程序必须包含一个 main 函数。
- (3) C 程序的基本组成单位是函数。
- (4) 在 C 程序中, 注释说明只能位于一条语句的后面。

【相关知识】 C 程序的组成。

【例题答案】 (1)、(2) 和 (3) 是正确的, 因为一个 C 程序由若干个函数组成, 其中必须有且只有一个主函数 main(), 所以, C 程序的基本单位是函数, C 程序是函数驱动的。

(4) 是不正确的, 在 C 语言中, 注释部分允许出现在程序的任何位置。

例 1.3 ★程序的 3 种基本控制结构是什么?

【相关知识】 C 语言的程序结构。

【例题答案】 程序的 3 种基本控制结构是顺序、选择(条件)和循环(重复)。顺序结构是指语句从上往下顺序执行的结构; 选择结构是指根据指定的条件确定执行多个语句中的一个语句的结构; 循环结构是指根据指定的条件确定是否重复执行一个语句的结构。实际上用顺序结构和循环结构完全可以实现选择结构, 因此, 理论上最基本的控制结构只有两种。

结构化程序设计中只能使用这 3 种基本控制结构。

例 1.4 ★下面关于程序设计风格的叙述, 哪些是正确的:

- (1) 编写程序时, 应使用括号以改善表达式的清晰度。
- (2) 在程序设计中, 不要进行浮点数相符的比较。
- (3) 不要使用数据类型来对数据值进行防范。
- (4) 使用有意义的标识符。
- (5) 结构化程序设计语言中没有 Goto 语句。
- (6) 一般而言, 语言的级别越高, 用它编出的程序越短。
- (7) C 语言是一种弱类型语言。

【相关知识】 C 语言的程序结构等。

【例题答案】 (1) 是正确的。

(2) 是正确的, 因为在精度有限的情况下, 两个浮点数相等比较永远不能成立;

(3) 是错误的, 因为要考虑数据的溢出问题, 如用 int 型变量存放 1234567 就会上溢出, 因此需要使用数据类型来对数据值进行防范。

(4) 是正确的。

(5) 是错误的, 如 C 语言是一种结构化程序设计语言, 其中也包含 Goto 语句, 但从结构化程序设计看, 应尽可能不使用 Goto 语句。

(6) 就一般而言, 这句话是正确的, 例如, 同样的问题, 用 C 语言编写程序通常比用汇编语言编写的程序要短。在计算机科学中, 一般将面向特定领域的程序设计语言(或开发工具)称为第四代语言(4GL), 如数据库管理系统 FoxPro 就是 4GL, 采用 C 语句也可以开发 MIS 系统, 但使用 FoxPro 开发同样的 MIS 系统, 速度就快得多, 而且程序也短得多。这就是为什么软件开发人员总是采用较高的软件开发工具进行软件设计与开发的原因。

(7) 是错误的。C 语言是一种强类型语言。所谓强类型语言要求程序设计者在使用数据之前对数据的类型进行说明。

1.3 自测题及参考答案

1. 说明 C 语言中标识符的构成。

答: C 语言中标识符只能由字母、数字和下划线 3 种字符组成, 且第一个字符必须为字母或下划线。

2. 以下哪个是 C 程序的基本结构单位?

(1) 文件;

- (2) 语句;
- (3) 函数;
- (4) 表达式。

答：文件是 C 程序的基本编译单元；表达式是运算符和运算数等构成的一个序列，其目的是用来说明一个计算过程；语句是 C 程序的基本组成单位。只有函数才是 C 程序的基本结构单位，所以答案为 (3)。

2. ★高级程序设计语言的语句分为哪两种。

答：高级程序设计语言的语句分为两种，一种用于描述计算机执行的操作运算（如表达式语句），即操作运算语句；另一种是控制上述操作运算的执行顺序（如循环控制语句），即流程控制语句。后一类语句也称为过程化语句。例如，以下 C 程序段：

```
s = 0;                /*①*/
for (i = 1; i < 10; i + +) /*②*/
    s = s + i;        /*③*/
printf("s = %d\n", s); /*④*/
```

其中，语句①、③和④是操作运算语句，语句②是流程控制语句。

3. 下面几组选项中，均不合法的标识符是哪几组？

- (1) A, P_0, do
- (2) float, la0, _A
- (3) b - a, goto, int
- (4) _123, temp, INT

答：(1) 中 A 和 P_0 是合法的标识符，do 是关键字，不是合法的标识符；

(2) 中 la0 和 _A 是合法的标识符，float 是关键字，不是合法的标识符；

(3) 中均为不合法的标识符，因为 goto 和 int 都是关键字，b - a 包含了不合法的字符“-”；

(4) 中均为合法的标识符（注意，INT 不同于 int，后者是关键字，前者不是）。

所以本题答案为 (3)。

4. C 编译程序的功能是什么？

答：C 编译程序的功能是将 C 源程序进行语法检查，若无语法错误，再翻译成目标代码，最后通过与标准库连接后形成可执行文件。

没有 C 编译程序，编写的 C 程序是无法运行的。目前常用的 C 编译程序有 Turbo C、Visual C++ 和 Borland C++ 等。

5. ★举例说明程序的 3 种结构。

答：程序的 3 种结构为顺序结构、选择结构和循环结构。

例如，以下为顺序结构：

```
int x = 10;
printf("%d\n", &x);
```

以下为选择结构：

```
if (x > 0)
    y = 1;
else if (x == 0)
```

```
y = 0;
else
    y = - 1;
以下为循环结构:
for (i = 0;i<10;i + + )
    s = s + i;
```

6. 若一个 C 程序中函数的调用关系如图 1.1 所示, 说明该程序的执行过程。

答: 将图 1.1 称为程序结构图, 因为它反映了程序中函数之间的调用关系即结构组成关系。在该程序中, 先从 main 函数开始执行; 在执行 main 函数中, 遇到了调用 f1 函数, 则转向 f1 函数去执行, f1 函数执行完毕, 又返回到 main 函数; 然后又遇到了调用 f2 函数, 则转向 f2 函数去执行, f2 函数执行完毕, 又返回到 main 函数; 继续执行 main 函数余下的语句, 当全部执行完毕, 则退出整个程序的执行过程。

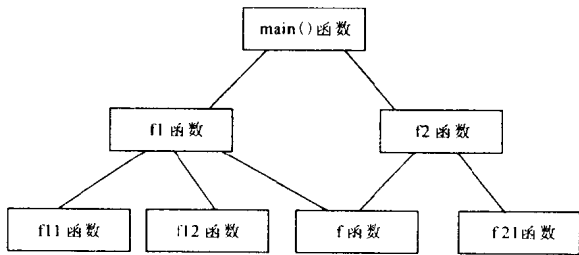


图 1.1 一个 C 程序的结构

同样, 在执行 f1 函数时, 又先后调用 f11、f12 和 f 函数执行, 最后都返回到 f1 函数。在执行 f2 函数时, 先后调用 f 和 f21 函数执行, 最后都返回到 f2 函数。整个程序的执行过程如图 1.2 所示, 图中的虚线表示程序的执行过程, 从中可以看到, 程序执行从 main 函数开始, 又是从 main 函数结束的。

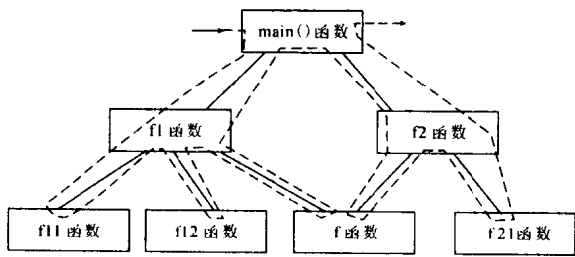


图 1.2 一个 C 程序的执行过程

第2章 数据类型及其运算

基本知识点：C 语言的数据类型，常量和变量，运算符，表达式等相关概念。

重点：各种类型的数据之间的相互转换。

难点：在程序设计中正确利用各种数据类型解决实际问题。

2.1 考点精要

2.1.1 基本数据类型

数据类型简称为类型。C 语言的数据类型有基本数据类型和非基本数据类型之分。基本数据类型是 C 语言内部预先定义的数据类型；非基本数据类型是由用户指定的，也称为用户定义数据类型。C 语言的数据类型如图 2.1 所示。

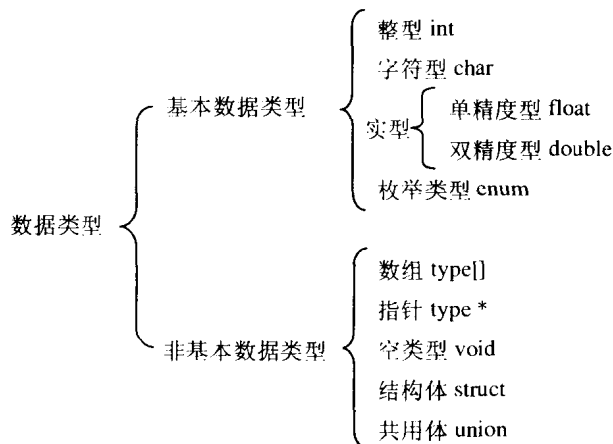


图 2.1 C 语言的数据类型

图中的 `type` 表示非空数据类型。

从中可以看到，C 语言基本数据类型只有 5 种，即整型 `int`、字符型 `char`、单精度浮点型 `float`、双精度浮点型 `double` 和枚举类型。除了这些基本数据类型外，还有一些数据类型修饰符，用来改变基本类型的意义。这些修饰符有：`long`（长型符）、`short`（短型符）、`signed`（有符号）和 `unsigned`（无符号）。

使用修饰符时有以下规定：

- `short` 只能修饰 `int`，`short int` 表示短整数类型，可以省略为 `short`。

- long 只能修饰 int 和 double。long int 为长整数类型，可省略为 long；long double 为长双精度类型。

- unsigned 和 signed 只能修饰 char 和 int。一般情况下，默认的 char 和 int 分别为 signed char 和 signed int。实型 float 和 double 总是有符号的，不能用 unsigned 修饰。

数据类型的描述确定了其内存所占空间大小，也确定了其表示范围。以 16 位计算机中表示为例，基本数据类型加上修饰符的描述如表 2.1 所示。

表 2.1 常用的基本数据类型描述

类型（等价的类型）	说 明	长度 (字节)	表示范围	备 注
char (signed char)	字符型	1	-128~127	$-2^7 \sim (2^7-1)$
unsigned char	无符号字符型	1	0~255	$0 \sim (2^8-1)$
int (signed int, short int, signed short int)	整型	2	-32768~32767	$-2^{15} \sim (2^{15}-1)$
unsigned int (unsigned short int)	无符号整型	2	0~65535	$0 \sim (2^{16}-1)$
long int (signed long int)	长整型	4	-214748364~21473647	$-2^{31} \sim (2^{31}-1)$
unsigned long int	无符号长整型	4	0~4294967295	$0 \sim (2^{32}-1)$
float	单精度型	4	$-3.4 \times 10^{38} \sim 3.4 \times 10^{38}$	7 位有效位
double	双精度型	8	$-1.7 \times 10^{308} \sim 1.7 \times 10^{308}$	15 位有效位
long double	长双精度型	10	$-3.4 \times 10^{4932} \sim 1.1 \times 10^{4932}$	19 位有效位

2.1.2 变量和常量

C 语言程序中所用到的每一个变量和常量等都是程序的基本操作对象。它们都隐式地或显式地与一种数据类型相联系。每种数据类型都表明了它的可能取值范围及能在其上所进行的运算。

1. 变量

变量是指在程序运行过程中其值可以被改变的量。变量被区分为不同的类型，不同类型的变量在内存中占用不同的存储单元，以便用来存放相应变量的值。

程序中所用到的每一个变量都应该有一个名字作为标识，而且 C 语言规定，程序中所要用到的变量应该先定义后使用。通常，对变量的定义放在函数的开头部分，但也可以放在函数的外部或复合语句的开头。

变量有类型的区分，如整型变量、实型变量、字符型变量等。C 语言在定义变量的同时说明该变量的类型，系统在编译时就能根据定义及其类型为它分配相应数量的存储单元。

