高等院校“十二五”核心课程辅导丛书

C语言
答疑解惑与典型题解

C YU YAN
DAYIJIEHUO YU DIANXINGTIJIE

吴　婷　吴晓维　宋琳琳　编著

北京邮电大学出版社
www.buptpress.com

高等院校“十二五”核心课程辅导丛书

C语言答疑解惑与典型题解

吴　婷　吴晓维　宋琳琳　编著

北京邮电大学出版社
·北京·

内 容 简 介

本书深入浅出、系统地全面地介绍了C语言的核心内容。全书共分16章，内容包括C语言基础、算法和结构化程序设计、数据类型、运算符与表达式、顺序结构、选择结构、循环结构、数组、字符串、函数、指针、结构体和共同体、预处理命令、位运算和文件操作等。

本书以“答疑解惑＋典型题解”为主线组织编写，每一章都列举了大量的题目(其中也包括各大高校的考研真题)，并对其进行了详细分析评注，以便于帮助读者掌握本章的重点及迅速回忆本章的内容。本书结构清晰、易教易学、实例丰富、学以致用、注重能力，对易混淆和历年考题中较为关注的内容进行了重点提示和讲解。

本书既可以作为学习C语言的辅导书，可也以作为复习考研、计算机等级考试的参考书，更可以作为各类培训班的培训教程。此外，本书也非常适于教师的C语言教学以及各种编程自学人员阅读参考。

图书在版编目(CIP)数据

C语言答疑解惑与典型题解/吴婷，吴晓维，宋琳琳编著. --北京：北京邮电大学出版社，2010.5
ISBN 978-7-5635-2289-7

Ⅰ.①C… Ⅱ.①吴…②吴…③宋… Ⅲ.①C语言—程序设计 Ⅳ.①TP312

中国版本图书馆CIP数据核字(2010)第065637号

书　　名：C语言答疑解惑与典型题解
作　　者：吴　婷　吴晓维　宋琳琳
责任编辑：满志文
出版发行：北京邮电大学出版社
社　　址：北京市海淀区西土城路10号(邮编：100876)
发 行 部：电话：010-62282185　传真：010-62283578
E-mail：publish@bupt.edu.cn
经　　销：各地新华书店
印　　刷：北京源海印刷有限责任公司
开　　本：787 mm×1 092 mm　1/16
印　　张：18
字　　数：541千字
版　　次：2010年5月第1版　2010年5月第1次印刷

ISBN 978-7-5635-2289-7　　定　价：35.00元

前　言

本书是为读者学习C语言而编写的教学辅导书，可帮助读者复习课程的基本内容，检验各种算法的掌握程度，培养和提高用C语言解决实际问题的能力，力争使读者在学完本书之后，在编程和解决实际问题方面都达到一个新的高度。

1. 本书阅读指南

本书对C语言知识点的常见问题进行了讲解，同时分析了近几年的考研题目，并给出了详实的参考答案，读者可以充分地了解各个学校考研题目的难度，查缺补漏，有针对性地提高自己的水平。本书共分16章。

第1章主要讲解C语言的基本概念和入门。

第2章主要讲解C语言的算法和结构化程序设计。

第3章主要讲解C语言的数据类型。

第4章主要讲解C语言的运算符和表达式。

第5章主要讲解C语言顺序结构方面的知识。

第6章主要讲解C语言选择结构方面的知识。

第7章主要讲解C语言循环结构方面的知识。

第8章主要讲解C语言数组方面的知识。

第9章主要讲解C语言字符串方面的知识。

第10章主要讲解C语言的函数，函数是C语言的核心所在，本章对函数的定义用法等进行了详细讲解。

第11章主要讲解C语言指针方面的知识。

第12章主要讲解C语言的结构体和共同体。

第13章主要讲解C语言的预处理命令。

第14章主要讲解C语言位运算方面的知识。

第15章主要讲解C语言的文件操作。

第16章提供了一套测试题和一套考研真题，为读者提供一个自我分析解决问题的过程。

2. 本书的特色与优点

(1) 结构清晰，知识完整。本书系统性强，依据高校教学大纲组织内容，同时覆盖最新版本的所有知识点，并将实际经验融入基本理论之中。

(2) 内容详实，解答完整。本书涵盖近几年各大高校的大量题目，示例众多，步骤明确，讲解细致，读者不但可以利用题海战术完善自己的弱项，更可以有针对性地了解某些重点院校的近年考研题目及解题思路。

(3) 学以致用，注重能力。一些例题后面有与其相联系的知识点详解，使读者在解答问题的同时，对基础理论得到更深刻的理解。

(4) 重点突出，实用性强。突出核心知识，对重点、难点、易混淆知识点进行剖析与解释，通过对试题的分析提高读者解决实际问题的能力。

3. 本书读者定位

本书既可以作为学习C语言的辅导书，也可以作为复习考研、计算机等级考试的参考书，更可以作为各类培训班的培训教程。此外，本书也非常适于教师的C语言教学以及各种编程自学人员参考阅读。

本书由吴婷（上海电机学院）、吴晓维和宋琳琳编著。全书框架结构由何光明和吴婷拟定。另外还要感谢李海、陈玉旺、张凌云、赵传申、陈海燕、陈智、王珊珊、吴涛涛、赵梨花等同志的关心和帮助。

由于编者水平和经验有限，加之编写时间仓促，本书难免会有不妥之处，敬请广大读者批评指正。如遇到疑难问题可通过以下方式与我们联系：bjbaba@263.net。

编　者

目 录

第1章

C语言基础

1.1 答疑解惑

1.1.1 C语言是如何构成的?

函数是C程序的基本单位。一个C源程序至少要包含一个main函数,也可以包含若干个其他函数。C程序中有三种类型的函数:被main函数调用的函数可以是系统提供的库函数,也可以是用户自己定义的函数。

(1) 主函数,即main( )。

(2) 系统库函数,如:printf( )、getchar( )等。

(3) 用户自定义函数,如:getName( )。

1.1.2 C程序是如何执行的?

在C程序中,main函数是程序执行的唯一入口和出口,它可以出现在程序中的任意位置。main函数可以调用其他普通函数,包括系统提供的库函数和用户自己定义函数。

1.1.3 C程序是如何编译连接的?

计算机无法直接执行汇编语言和高级语言,而只能执行机器语言。因此,C源程序必须被翻译为二进制机器指令后才能被计算机执行,过程如下:

(1) C源程序(扩展名为.(C)经过"编译程序"编译后,生成一个二进制文件(扩展名为.obj),此过程称为"编译"。

(2) 连接程序把该二进制文件与C的各种库函数连接起来,生成一个可执行文件(扩展名为.exe),此种类型的文件才可以直接执行。在操作系统环境下,只需输入文件名称(不必输入扩展名.exe)就可直接运行该文件。

1.1.4 C程序中如何使用注释?

注释可以增强程序的可读性,在初学时就要养成良好的编程风格。C语言的注释符

为“/*… */”,“/”和“ * ”之间不允许留有空格。注释在编译时将被忽略。注释通常用于:

(1) 版本、版权声明;

(2) 函数接口说明;

(3) 重要的代码行或段落提示。

虽然注释有助于理解代码,但注意不可过多地使用注释。通常有以下规则:

(1) 注释的花样要少,尽量避免在注释中使用缩写,特别是不常用缩写。

(2) 如果代码本来就是清楚的,则不必加注释。否则多此一举,令人厌烦。

(3) 边写代码边注释,修改代码同时修改相应的注释,以保证注释与代码的一致性。不再有用的注释要删除。

典型题解 C语言基础

1.2 典型题解

题型 1 C 程序构成

【例 1-1★】(西安电子科技大学)以下叙述不正确的是_____。

(A) 一个 C 源程序可由一个或多个函数组成

(B) 一个 C 源程序必须包含一个 main 函数

(C) C 程序的基本组成单位是函数

(D) 在 C 程序中,注释说明只能位于一条语句的后面

分析:C 程序由函数构成,其中必须有且只能有一个主函数〔main( )函数〕,另外还可包含一个或多个子函数;C 程序的注释部分应包含在“/*”与“ */”之间,“/”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,注释在编译时被忽略。

解答:正确答案为 D。

【例 1-2★】(江苏大学)以下叙述正确的是_____。

(A) main( )必须是程序的第一行

(B) C 程序每一行只能写一条语句

(C) C 程序可以由一个或多个函数组成

(D) 在编译时可以发现注释中的拼写错误

分析:C 程序中,main( )函数可以出现在任意位置,并且始终是函数执行的入口;C 语言书写格式自由,一行内可以写几个语句,一个语句也可以写在几行上;C 程序中的注释在编译时被忽略,即不作任何处理,因此编译器无法发现注释中的拼写错误。

解答:正确答案为 C。

【例 1-3★】(南京林业大学)一个 C 语言程序总是从_____开始执行。

(A) 主过程　　(B) 主函数　　(C) 子程序　　(D) 主程序

分析:C 程序中,main( )函数是程序执行的唯一入口,如果遇到调用其他函数,则转去执行被调用的函数,该函数执行完毕后,重新返回 main( )函数,只有在 main( )的所有可执行语句执行完毕时,整个 C 程序才执行完成。

解答:正确答案为 B。

【例 1-4】以下叙述中正确的是_____。

(A) C 语言程序将从源程序中第一个函数开始执行

(B) 可以在程序中由用户指定任意一个函数作为主函数,程序将从此开始执行

(C) C语言规定必须用 main 作为主函数名，程序从此开始执行，在此结束

(D) main 可作为用户标识符，用以命名任意一个函数作为主函数

分析：C程序的主函数必须是 main 函数，并且无论 main 函数在程序中的位置如何，它都是C程序的主函数，是程序执行的入口和出口。

解答：正确答案为 C。

【例 1-5】对于一个正常运行的 C 程序，以下叙述中正确的是_____。

(A) 程序的执行总是从 main 函数开始，在 main 函数结束

(B) 程序的执行总是从程序的第一个函数开始，在 main 函数结束

(C) 程序的执行总是从 main 函数开始，在程序的最后一个函数中结束

(D) 程序的执行总是从程序的第一个函数开始，在程序的最后一个函数中结束

分析：C程序的执行是从 main 函数开始，在 main 函数结束。当 main 函数中调用其他函数时，转去执行其他函数，但执行完毕后必须返回 main 函数。

解答：正确答案为 A。

【例 1-6】下列叙述中正确的是_____。

(A) 每个 C 程序文件中都必须要有一个 main( )函数

(B) 在 C 程序中 main( )函数的位置是固定的

(C) C 程序中所有函数之间都可以相互调用，与函数所在位置无关

(D) 在 C 程序的函数中不能定义另一个函数

分析：每个C程序都必须至少包含一个 main 函数，它可以出现在程序中的任何位置。函数是C程序的基本模块，函数内不可再定义其他函数。函数间可相互调用，函数的调用与函数所在位置无关。

解答：正确答案为 D。

【例 1-7】下列叙述中错误的是_____。

(A) 一个 C 语言程序只能实现一种算法

(B) C 程序可以由多个程序文件组成

(C) C 程序可以由一个或多个函数组成

(D) 一个 C 函数可以单独作为一个 C 程序文件存在

分析：一个C程序包含一个 main 函数和多个其他函数。每个函数都可单独实现一个算法，也可单独作为一个C程序文件存在。如果被调用的函数在其他文件中时，调用函数就要必须首先包含该文件。因而，一个C程序可以包含一个或多个文件。

解答：正确答案为 A。

题型 2　C 程序编译连接

【例 1-8】C 语言程序名的扩展名是_____。

(A) .exe　　(B) .c　　(C) .obj　　(D) .cp

分析：C源程序，扩展名为.c；源程序被编译器编译后生成目标程序，扩展名为.obj；目标程序与C的各种库函数连接起来后，生成可执行程序，扩展名为.exe。

解答：正确答案为 B。

【例 1-9】下列叙述中错误的是_____。

(A) 计算机不能直接执行用 C 语言编写的源程序

(B) C 程序经 C 编译后，生成扩展名为.obj 的文件是一个二进制文件

(C) 扩展名为.obj 的文件，经连接程序生成扩展名为.exe 的文件是一个二进制文件

(D) 扩展名为.obj 和.exe 的二进制文件都可以直接运行

分析：计算机无法直接执行汇编语言和高级语言，而只能执行机器语言。因此，C源程序无法直接被计算机执行，C源程序.c被编译后生成目标程序.obj，目标程序被连接后生成可执行文件.exe。其中

.obj和.exe都是二进制文件，但只有.exe才能被计算机直接运行。

解答：正确答案为D。

【例1-10】以下叙述中错误的是_____。

(A) C语言源程序经编译后生成扩展名为.obj的目标程序

(B) C程序经过编译、连接步骤之后才能形成一个真正可执行的二进制机器指令文件

(C) 用C语言编写的程序称为源程序，它以ASCII码形式存放在一个文本文件中

(D) C语言中的每条可执行语句和非执行语句最终都将被转换成二进制的机器指令

分析：C语言编写的源程序以ASCII码形式存放在文本文件.C中，编译后生成目标程序.obj，连接后生成可执行的二进制机器指令文件.exe。C语言中的非执行语句，如注释，在编译时被忽略，而不会翻译为二进制代码。

解答：正确答案为D。

【例1-11】以下叙述中正确的是_____。

(A) C语言的源程序不必通过编译就可以直接运行

(B) C语言中的每条可执行语句最终都将被转换成二进制的机器指令

(C) C源程序经编译形成的二进制代码可以直接运行

(D) C语言中的函数不可以单独进行编译

分析：C语言编写的源程序无法被计算机直接执行，必须经过编译连接后才能执行，其中编译生成的目标文件也无法被直接执行。源程序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，而像注释这样的非执行语句将会在编译时就被忽略。

解答：正确答案为B。

题型3　注释

【例1-12】以下4个程序中，完全正确的是_____。

(A)
```
#include <stdio.h>
main( );
{/*programming */
printf("programming!\n");}
```

(B)
```
#include <stdio.h>
main( )
{/*/ programming /*/
printf("programming!\n");}
```

(C)
```
#include <stdio.h>
main( )
{/*/*programming */*/
printf("programming!\n");}
```

(D)
```
include <stdio.h>
main( )
{/*programming */
printf("programming!\n");}
```

分析：main函数后不应该有分号，选项A错误。按照/* */匹配规则，选项C中的/*/*programming */*/将只有/*/*programming */被编译器理解为注释，而多出的*/将被理解为可执行语句。预编译指令include <stdio.h>前应该有#号，选项D错误。

解答：正确答案为B。

第2章

算法和结构化程序设计

2.1 答疑解惑

2.1.1 什么是程序?

著名计算机科学家沃思(Nikikaus Wirth)这样定义程序

程序=数据结构+算法

其中数据结构是对数据的描述,定义程序数据的类型和数据组织形式;算法是对操作步骤的描述。

实际上,一个程序除了以上两个主要要素之外,还应当采取一定的程序设计方法,并通过某种计算机语言来实现。因此可以这样定义程序

程序=数据结构+算法+程序设计方法+程序设计语言和环境

2.1.2 什么是算法?

算法是为解决某个特定问题而采取的确定且有限的步骤。算法具有以下5个特征:

(1) 用穷性:对于任何合法的输入,一个算法总是在执行有限步之后结束,并且每一步也都是在有限的时间内执行完。

(2) 确定性:算法的每一条指令必须有确切的含义,不能有二义性;对于相同的输入必须得到相同的输出。

(3) 可行性:算法中描述的操作都可以通过有限次地执行已经实现的基本算法来实现。

(4) 输入:一个算法应该有零个或多个输入。

(5) 输出:一个算法应该有一个或多个输出。

2.1.3 如何评价、选择算法?

算法的性能是评价算法好坏的标准,它包括以下5个方面:

(1) 正确性:算法的执行结果应当满足预先规定的功能和性能要求。

(2) 简明性:一个算法应当思路清晰、层次分明、简单明了、易读易懂。

(3) 健壮性:当输入不合法数据时,应能做适当处理,不至于引起严重后果。

(4) 效率:算法执行时计算机资源的消耗,包括存储和运行时间的开销,通常是通过空间复杂度和时间复杂度来进行度量。

(5) 最优性:解决同一个问题可能有多种算法,应进行比较,选择最佳算法。

2.1.4 如何设计算法?

常用的算法设计策略有以下几种:

(1) 递归:最常用的算法设计思想。

(2) 分治法:采用的是分而制之的算法思想。

(3) 模拟法:用计算机模拟实际场景,常用于与概率有关的问题。

(4) 贪心算法:采用的是贪心策略。

(5) 状态空间搜索法:被称为"万能算法"的算法设计策略。

(6) 随机算法:利用随机选择自适应地决定优先搜索的方向。

(7) 动态规划:常用的最优化问题解决方法。

2.1.5 什么是结构化程序设计?

结构化程序设计的基本思想是"分而治之",即将一个复杂问题的求解过程划分为多个阶段,每个阶段所处理的问题都控制在容易理解和处理的范围内。

在结构化程序设计中,任何程序都可以用 3 种基本结构表示,由它们经过反复组合、嵌套构成的程序称为结构化程序。这 3 种基本结构如下:

(1) 顺序结构:语句按照其在程序中的先后顺序逐条执行,没有分支,没有转移。一般赋值语句、输入/输出语句等可构成顺序结构。

(2) 选择结构:语句执行时,根据不同的条件执行不同分支中的语句。if、switch 等语句可构成选择结构。

(3) 循环结构:根据不同的条件,使同一组语句执行多次或一次也不执行。while、do-while、for 等语句可构成循环结构。

通常采取以下方法进行结构化设计:

(1) 自顶向下。

(2) 逐步细化。

(3) 模块化设计。

(4) 结构化编码。

采有自顶向下、逐步细化的方法可以使程序的结构清晰、便于阅读与修改。就像写文章一样:应先设想好整个文章分成哪几部分,然后再进一步考虑每一部分分成哪几节,每一节又分成哪几小段,每一段包含什么内容。

C 语言是一种结构化程序设计语言。它提供了实现 3 种基本结构的语句;提供"函数"以实现模块划分。这些都将在后面的章节中逐一展开讨论。

2.1.6 如何用图形描述算法?

以"输出 10 以内能被 3 整除的自然数"为例,介绍算法的两种图形化描述的方法。

(1) 用流程图描述算法

流程图是用一些图框表示各种算法。ANSI(美国国家标准化协会)规定了一些常用的流程图符号,已被广泛采用。用流程图描述的算法如图 2.1 所示。

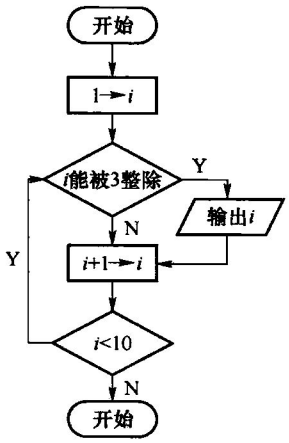

图 2.1 流程图描述的算法

(2) 用 N-S 图描述算法

如前所述,3 种基本结构的顺序组合可以表示任何复杂的算法结构。因此,只要规定 3 种基本结构的描述方法,就可以通过它们的组合来描述任何算法。

1973 年美国学者 I. Nassi 和 B. Shneiderman 提出了 N-S 结构化流程图(N 和 S 是两位学者英文名的第一个字母)。该流程图去掉了带箭头的流程线,全部算法写在一个矩形框内,框中还可包含其他从属于它的框。3 种基本结构的 N-S 描述如图 2.2 所示。

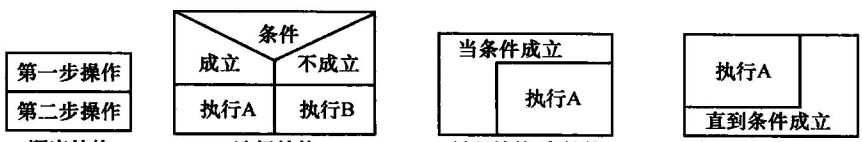

图 2.2 3 种基本结构的 N-S 描述法

用 N-S 图描述的算法如图 2.3 所示。

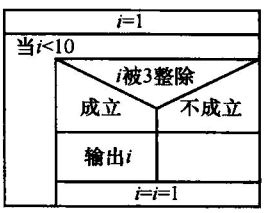

图 2.3 N-S 图描述的算法

典型题解 算法和结构化程序设计

2.2 典型题解

题型 1 算法的特征

【例 2-1】以下叙述中错误的是_____。

(A) 算法正确的程序最终一定会结束

(B) 算法正确的程序可以有零个输出

(C) 算法正确的程序可以有零个输入

(D) 算法正确的程序对于相同的输入一定有相同的结果