

2016年版

全国计算机等级考试 教程

National Computer Rank Examination

二级C语言程序设计

K

扫书中二维码 看考点微视频
掌握重点难点 等考轻松过关



全国计算机等级考试命题研究中心
未来教育教学与研究中心



电子科技大学出版社

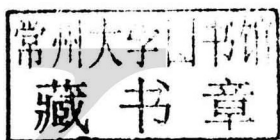
2016年版

全国计算机等级考试

教程

National Computer Rank Examination

二级C语言程序设计



全国计算机等级考试命题研究中心
未来教育教学与研究中心



电子科技大学出版社

图书在版编目(CIP)数据

二级 C 语言程序设计 / 全国计算机等级考试教材编写组
未来教育教学与研究中心主编. -- 成都: 电子科技大学
出版社, 2015. 8

全国计算机等级考试教程

ISBN 978 - 7 - 5647 - 3184 - 7

I. ①二… II. ①全… III. ①C 语言 - 程序设计 - 水
平考试 - 自学参考资料 IV. ①TP312 - 44

中国版本图书馆 CIP 数据核字(2015)第 188462 号

内 容 提 要

本书依据教育部考试中心最新发布的《全国计算机等级考试大纲》以及作者多年对等级考试的研究编写而成,旨在帮助考生(尤其是非计算机专业初学者)学习相关内容,顺利通过考试。

全书共 13 章,主要包括:程序设计和 C 语言,数据类型、运算符和表达式,顺序结构、选择结构和循环结构的程序设计,数组,函数,变量的作用域和存储类别,指针,编译预处理,结构体、共用体和用户定义类型,位运算及文件。

本书配套光盘中提供多媒体课堂,以动画的方式讲解重点和难点,为考生营造轻松的学习环境。此外,还提供了供考生熟悉无纸化考试环境的模拟系统。

本书可作为全国计算机等级考试培训教材和自学用书,也可以学习 C 语言的参考书。

全国计算机等级考试教程——二级 C 语言程序设计

出 版:电子科技大学出版社(成都市一环路东一段 159 号电子信息产业大厦 邮编:610051)

策划编辑:谢晓辉

责任编辑:谢晓辉

主 页:www.uestcp.com.cn

电子邮箱:uestcp@uestcp.com.cn

发 行:新华书店经销

印 刷:北京佳顺印刷有限公司

成品尺寸:185mm×260mm 印 张:16 字 数:364 千字

版 次:2016 年 1 月第一版第一次印刷

书 号:ISBN 978 - 7 - 5647 - 3184 - 7

定 价:32.00 元

■ 版 权 所 有 侵 权 必 究 ■

◆ 本社发行部电话:028 - 83202463;本社邮购电话:028 - 83208003

◆ 本书如有缺页、破损、装订错误,请寄回印厂调换。

编委会

EDITORIAL BOARD

主 编：詹可军

副 主 编：段中存

编委名单：(排名不分先后)

龚 敏	尚金妮	张 涛	段中存
武露文	朱爱彬	张明涛	蔡广玉
韩海洋	刘 兵	张晓玲	钱 凯
王 翔	费 菲	刘志强	范二朋
何海平	韩雪冰	曹秀义	詹可军
朱 红	胡结华	章 妹	

前言

全国计算机等级考试由教育部考试中心主办，是国内影响最大，参加考试人数最多的计算机水平考试。它的根本目的在于以考促学，这决定了它的报考门槛较低，考生不受年龄、职业、学历等背景的限制，任何人都可根据自己学习和使用计算机的实际情况，选考不同级别的考试。本书面向选考二级 C 语言的考生。

一、为什么编写本书

计算机等级考试的准备时间短，一般从报名到参加考试只有近 4 个月的时间，留给考生的复习时间有限，并且大多数考生是非计算机专业的学生或社会人员，基础比较薄弱，学习起来比较吃力。

通过对考试的研究和对数百名考生的调查分析，我们逐渐摸索出一些减少考生（尤其是初学者）学习困难的方法，以帮助考生提高学习效率和学习效果。因此我们编写了本书，将我们多年研究出的教学和学习方法贯穿全书，帮助考生巩固所学知识，顺利通过考试。

二、本书特色

1. 全新升级的教程

根据教育部考试中心《关于全国计算机等级考试体系调整的通知（教试中心函〔2013〕29 号）》文件对计算机等级考试的调整规定，对考试系统的硬件环境和软件环境均进行升级，并发布全新大纲。我们在深入研究 2013 年新大纲、新操作系统及新考试方法的基础上，组织计算机专家编写了本系列图书。书中采用了最新无纸化题库资源，适用于 Windows 7 的系统环境，考生可以通过本书全面掌握最新大纲要求的考试内容。

2. 一学就会的教程

本书的知识体系都经过巧妙设计，力求将复杂问题简单化，将理论难点通俗化，让读者一看就懂，一学就会。

○针对初学者和考生的学习特点和认知规律，精选内容，分散难点，降低台阶。

○例题丰富，深入浅出地讲解和分析复杂的概念和理论，力求做到概念清晰、通俗易懂。

○采用大量插图，并通过生活化的实例，将复杂的理论讲解得生动、易懂。

○精心为考生设计学习方案，设置各种栏目引导和帮助考生学习。

3. 衔接考试的教程

我们深入分析和研究历年考试真题，结合考试的命题规律选择内容，安排章节，坚持多考多讲、少考少讲、不考不讲的原则。在讲解各章节的内容之前，都详细介绍了考试的重点和难点，从而帮助考生安排学习计划，做到有的放矢。

4. 书盘结合的教程

本书所配的光盘主要提供两部分内容：多媒体课堂和无纸化考试模拟系统。使用本书的光盘，就等于把辅导老师请回了家。

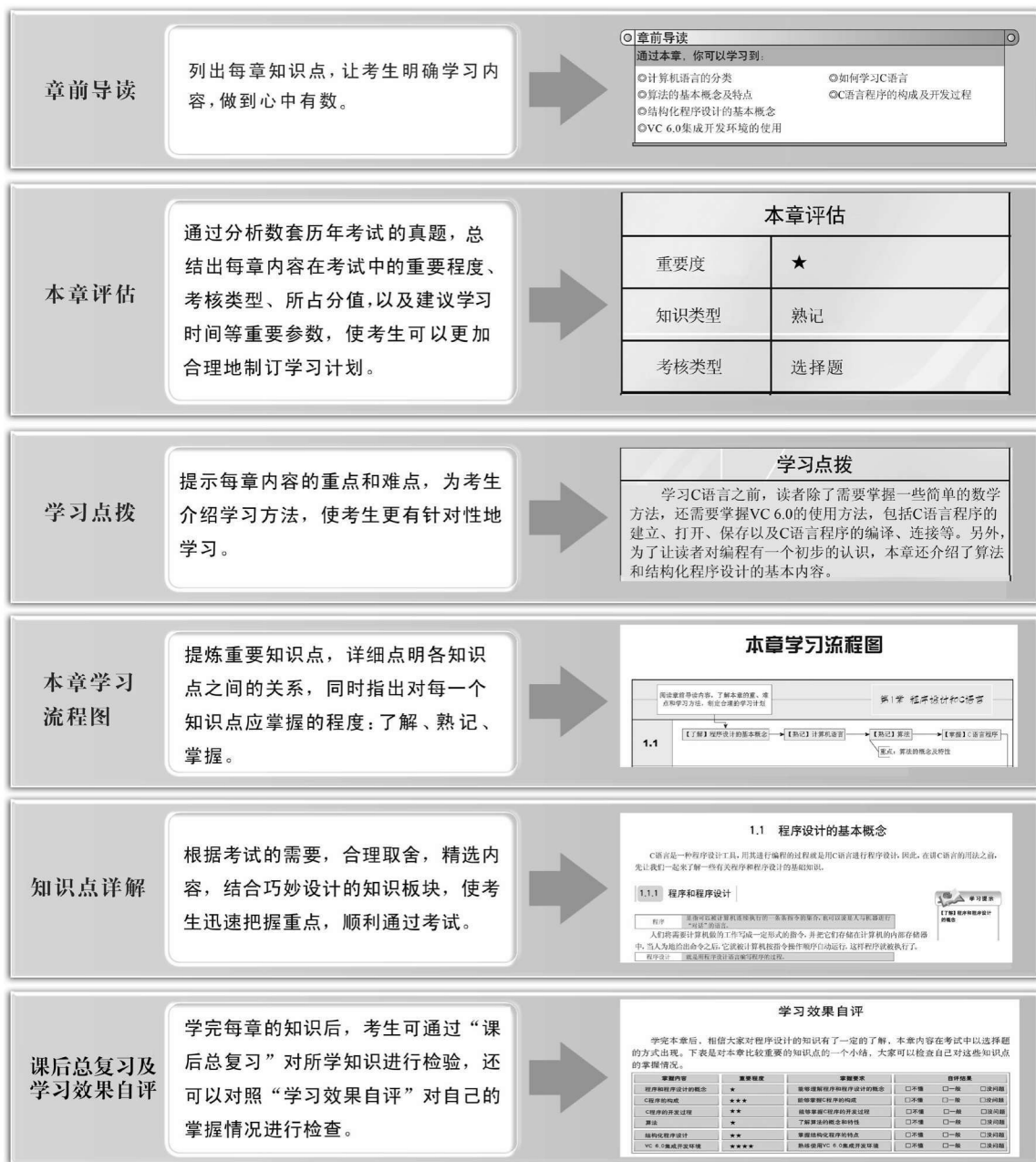
多媒体课堂用动画演绎复杂的理论知识，用视频讲解各种操作方法，使学习变得轻松而高效。

在无纸化考试模拟系统中提供大量的练习题，可真实模拟考试环境，帮助考生提前感受考试的全过程。

三、如何学习本书

1. 如何学习每一章

书中每章都安排了章前导读、本章评估、学习点拨、本章学习流程图、知识点详解、课后总复习、学习效果自评等固定板块。下面就详细介绍如何合理地利用这些资源。



2. 如何使用本书栏目

书中设计了3个小栏目，分别为“学习提示”、“请注意”和“请思考”。

(1) 学习提示

学习提示是从对应模块提炼的重点内容，读者可以通过它明确本部分内容的学习重点和掌握程度。

(2) 请注意

该栏目主要是提示读者在学习过程中容易忽视的问题，以引起大家的重视。

(3) 请思考

介绍完一部分内容后，以这种形式给出一些问题让读者思考，使读者能做到举一反三。



希望本书在备考过程中能够助您一臂之力，让您顺利通过考试，成为一名合格的计算机应用人才。

由于时间仓促，书中难免存在疏漏之处，恳请广大读者批评指正。

编 者

目 录

第 1 章 程序设计和 C 语言	1	2.4.2 实型常量	23
1.1 程序设计的基本概念	3	2.5 字符型数据	24
1.1.1 程序和程序设计	3	2.5.1 字符变量	25
1.1.2 计算机语言	3	2.5.2 字符常量	25
1.1.3 算法的概念	4	2.5.3 字符型数据的运算	26
1.1.4 结构化程序设计	5	2.5.4 字符串常量	27
1.2 Visual C++ 6.0 集成开发环境		2.6 不同数据类型之间的转换	28
简介	6	2.7 算术运算符和算术表达式	30
1.2.1 VC 6.0 的启动	6	2.7.1 算术运算符	30
1.2.2 VC 6.0 的退出	7	2.7.2 算术表达式	31
1.2.3 VC 6.0 集成开发环境		2.8 赋值运算符和赋值表达式	32
介绍	7	2.8.1 赋值运算符和赋值表达式	32
1.2.4 修改已有的源程序	7	2.8.2 复合的赋值表达式	33
1.2.5 编译、连接信息的处理	8	2.9 自增和自减运算符	34
1.3 C 语言简介	9	2.10 逗号运算符和逗号表达式	36
1.3.1 如何学习 C 语言	9	课后总复习	37
1.3.2 C 程序的构成	10	第 3 章 顺序结构程序设计	40
1.3.3 C 程序的书写格式	11	3.1 C 语句分类	42
1.3.4 C 程序的开发过程	11	3.2 赋值语句	43
1.4 C 程序的上机步骤	12	3.3 数据的输入与输出	44
课后总复习	13	3.4 数据的输入与输出——scanf 函数	44
第 2 章 数据类型、运算符和表达式	15	3.4.1 scanf 函数的调用格式	45
2.1 常量、变量和标识符	17	3.4.2 scanf 函数的格式字符	45
2.1.1 标识符	17	3.4.3 scanf 函数的使用说明	46
2.1.2 关键字	17	3.5 数据的输入与输出——printf 函数	48
2.1.3 常量	17	3.5.1 printf 函数的调用格式	48
2.1.4 符号常量	18	3.5.2 printf 函数的格式字符	49
2.1.5 变量	18	3.5.3 printf 函数的使用说明	52
2.2 数据类型	20	3.6 数据的输入与输出——getchar 函数和	
2.3 整型数据	20	putchar 函数	53
2.3.1 整型变量	20	3.6.1 字符输入函数 getchar	53
2.3.2 整型常量	22	3.6.2 字符输出函数 putchar	53
2.4 实型数据	22	3.7 顺序结构程序举例	54
2.4.1 实型变量	22	课后总复习	55

第4章 选择结构程序设计	59	6.2.3 一维数组的初始化	96
4.1 关系运算符和关系表达式	61	6.2.4 一维数组应用举例	97
4.1.1 关系运算符	61	6.3 二维数组	99
4.1.2 关系表达式	61	6.3.1 二维数组的定义	100
4.2 逻辑运算符和逻辑表达式	62	6.3.2 二维数组的引用	101
4.2.1 逻辑运算符	62	6.3.3 二维数组的初始化	101
4.2.2 逻辑表达式	64	6.3.4 二维数组应用举例	103
4.3 条件运算符和条件表达式	65	6.4 字符数组	104
4.4 if 语句	66	6.4.1 字符数组的定义	104
4.4.1 if 语句的基本形式	66	6.4.2 字符数组的引用	105
4.4.2 if 语句的嵌套	68	6.4.3 字符数组的初始化	105
4.5 switch 语句	70	6.4.4 字符数组的输入与输出	107
课后总复习	72	6.4.5 字符串处理函数	108
第5章 循环结构程序设计	76	6.4.6 字符数组应用举例	111
5.1 用 for 语句构成的循环结构	78	课后总复习	112
5.1.1 for 循环语句的一般形式	78	第7章 函数	115
5.1.2 for 循环语句的执行过程	78	7.1 库函数	117
5.2 用 while 语句构成的循环结构	80	7.2 函数定义	118
5.2.1 while 循环语句的一般形式	80	7.3 函数的参数和返回值	119
5.2.2 while 循环语句的执行过程	80	7.3.1 函数参数	119
5.3 do...while 语句构成的循环结构	81	7.3.2 函数返回值	120
5.3.1 do...while 循环语句的一般形式	81	7.4 函数的调用	122
5.3.2 do...while 循环语句的执行过程	81	7.4.1 函数调用的一般形式	122
5.3.3 3 种循环结构小结	82	7.4.2 函数的嵌套调用	123
5.4 循环结构的嵌套	83	7.4.3 函数的递归调用	124
5.5 break 语句和 continue 语句在循环体中的作用	84	7.5 函数的声明	125
5.5.1 break 语句	84	7.5.1 函数声明的形式	125
5.5.2 continue 语句	85	7.5.2 函数声明的位置	126
5.5.3 break 语句和 continue 语句的区别	86	7.6 函数参数传递	127
5.6 goto 语句	87	课后总复习	128
课后总复习	88	第8章 变量的作用域和存储类别	132
第6章 数组	91	8.1 变量的作用域	134
6.1 有关数组的基本概念	93	8.1.1 局部变量	134
6.2 一维数组	94	8.1.2 全局变量	136
6.2.1 一维数组的定义	94	8.2 变量的存储类别	137
6.2.2 一维数组的引用	95	8.2.1 auto 变量	137
		8.2.2 static 变量	138
		8.2.3 register 变量	139
		8.2.4 extern 变量	140
		8.3 函数的存储分类	142

8.3.1 内部函数	142	10.1.2 带参数的宏定义	185
8.3.2 外部函数	142	10.2 文件包含	186
课后总复习	143	课后总复习	187
第9章 指针	145	第11章 结构体、共用体和用户定义类型	189
9.1 地址和指针的概念	147	11.1 用 typedef 定义类型	191
9.2 指针变量	149	11.2 结构体	192
9.2.1 指针变量的定义	149	11.2.1 结构体类型的说明	192
9.2.2 指针运算符	150	11.2.2 结构体变量的定义	193
9.2.3 指针变量的初始化	151	11.2.3 结构体变量的初始化	196
9.2.4 指针的运算	151	11.2.4 结构体变量的引用	197
9.3 指针与一维数组	154	11.2.5 函数之间结构体变量的数据传递	198
9.3.1 指向数组元素的指针变量	154	11.3 动态存储分配和链表	199
9.3.2 数组元素的引用	155	11.3.1 动态存储分配	199
9.4 指针与二维数组	157	11.3.2 动态链表的概念	201
9.4.1 二维数组及其元素地址	157	11.3.3 利用结构体变量构成链表	202
9.4.2 指向数组元素的指针变量	160	11.4 共用体	206
9.5 指针与字符串	161	11.4.1 共用体类型的说明和变量定义	206
9.5.1 使用字符指针实现字符串的存储	161	11.4.2 共用体变量的引用	208
9.5.2 字符指针与字符数组的区别	162	课后总复习	209
9.6 指针的指针	164	第12章 位运算	213
9.6.1 指针数组	164	12.1 位运算符	215
9.6.2 指向指针的指针	166	12.2 位运算符详解	215
9.7 指针与函数	168	12.2.1 按位与运算	215
9.7.1 指针作为函数参数	168	12.2.2 按位或运算	217
9.7.2 一维数组名作函数实参	170	12.2.3 按位异或运算	217
9.7.3 二维数组名作函数实参	171	12.2.4 按位非运算	219
9.7.4 字符指针作函数实参	172	12.2.5 左移运算	219
9.7.5 返回指针值的函数	173	12.2.6 右移运算	220
9.7.6 函数指针	173	课后总复习	221
9.8 main 函数中的参数	175	第13章 文件	223
课后总复习	176	13.1 文件概述	225
第10章 编译预处理	181	13.2 文件类型指针	226
10.1 宏替换	183	13.3 文件的打开与关闭	227
10.1.1 不带参数的宏定义	183	13.3.1 文件的打开 (fopen 函数)	227

13.3.2 文件的关闭 (fclose 函数)	228
13.4 文件的读/写	229
13.4.1 字符读/写函数 fputc 和 fgetc	229
13.4.2 字符串读/写函数 fputs 和 fgets	230
13.4.3 数据块读/写函数 fread 和 fwrite	232
13.4.4 格式化读/写函数 fprintf 和 fscanf	232
13.4.5 判断文件结束函数 feof	233
13.5 文件的定位	234
13.5.1 rewind 函数	234
13.5.2 fseek 函数	234
13.5.3 ftell 函数	235
课后总复习	235
附录	238
附录 A 无纸化上机指导	238
附录 B 全国计算机等级考试二级 C 语言 考试大纲 (2013 年版)	242
附录 C 参考答案	244

第1章

程 序 设 计 和 C 语 言



视频课堂

第1课

VC 6.0环境简介

- VC 6.0 的启动和退出
- VC 6.0 集成开发环境简介

章前导读

通过本章，你可以学习到：

- ◎计算机语言的分类
- ◎算法的基本概念及特点
- ◎结构化程序设计的基本概念
- ◎VC 6.0 集成开发环境的使用
- ◎如何学习C语言
- ◎C语言程序的构成及开发过程

本章评估

重 要 度	★
知识类型	熟记
考核类型	选择题
所占分值	选择题:0.9分
学习时间	2课时

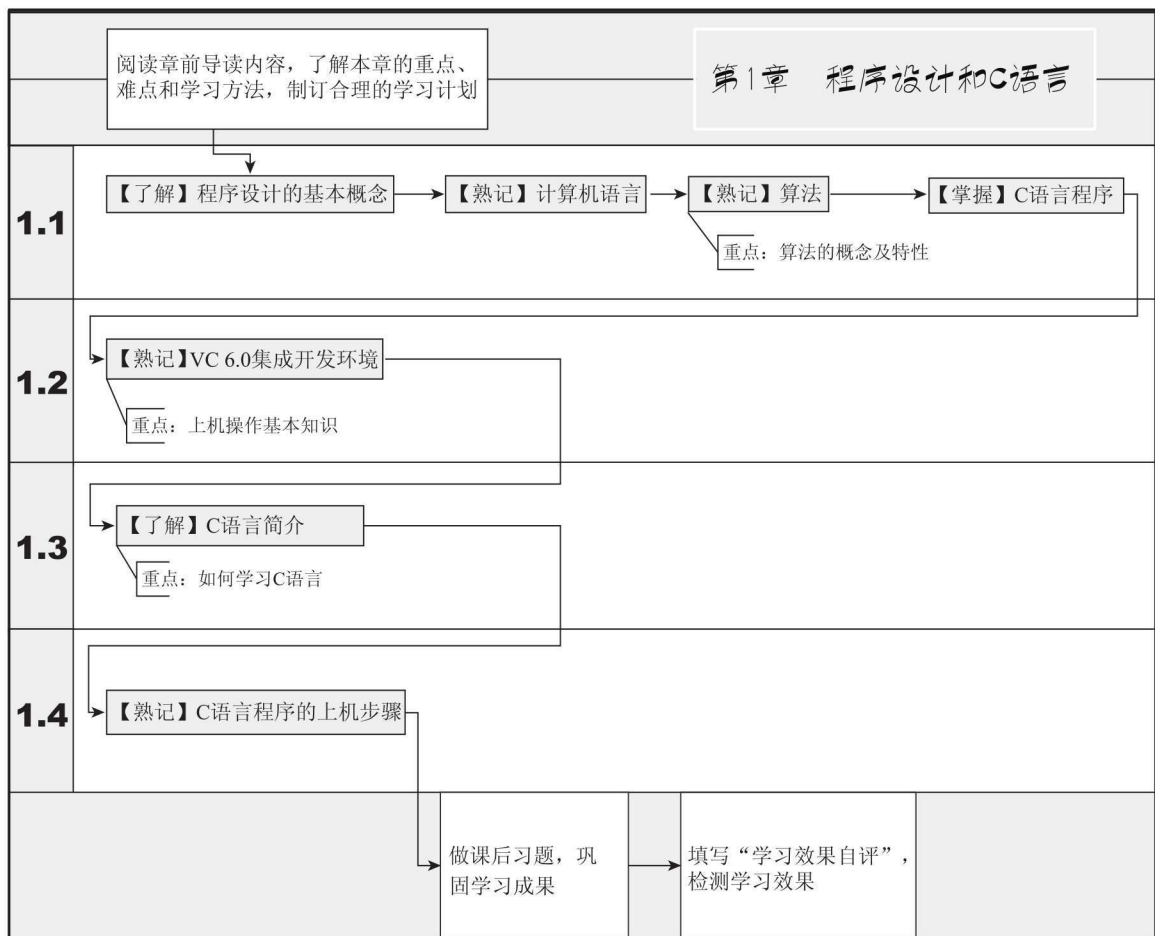
学习点拨

学习C语言之前，读者除了需要掌握一些简单的数学方法，还需要掌握VC 6.0 的使用方法，包括C语言程序的建立、打开、保存以及C语言程序的编译、连接等。另外，为了让读者对编程有一个初步的认识，本章还介绍了算法和结构化程序设计的基本内容。

本章将对上述内容进行简单的介绍，并且对VC 6.0 的使用、C程序的构成及如何进行C程序的上机操作进行详细介绍。

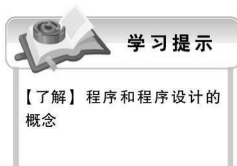
读者可以通过“本章学习流程图”把握本章的编写思路及重要知识点，理清本章的知识脉络。

本章学习流程图



1.1 程序设计的基本概念

C语言是一种程序设计工具，用C语言进行编程的过程就是程序设计的过程，因此，在讲C语言的用法之前，先了解一些有关程序和程序设计的基础知识。



1.1.1 程序和程序设计

程序	是指可以被计算机连续执行的一条指令的集合，也可以说是人与机器进行“对话”的语言。
----	--

人们将需要计算机做的工作写成一定形式的指令，并把它们存储在计算机的内部存储器中，当人为地给出命令之后，它就被计算机按指令操作顺序自动运行，这样程序就被执行了。

程序设计	就是用程序设计语言编写程序的过程。
------	-------------------

广义上说，程序设计是用计算机解决一个实际应用问题时的整个处理过程，包括提出问题、确定数据结构、确定算法、编程、调试程序及书写文档等一系列的过程。

- 提出问题：提出需要解决的问题，形成一个需求任务书。
- 确定数据结构：根据需求任务书提出的要求、指定的输入数据和输出结果，确定存放数据的数据结构。
- 确定算法：针对存放数据的数据结构确定解决问题、实现目标的步骤。
- 编写程序：根据指定的数据结构和算法，使用某种计算机语言编写程序代码，输入到计算机中并保存到磁盘上，简称编程。
- 调试程序：消除由于疏忽而引起的语法错误或逻辑错误；用各种可能的输入数据对程序进行测试，使之对各种合理的数据都能得到正确的结果，对不合理的数据都能进行适当处理。
- 书写文档：整理并写出文档资料。

请注意

数据结构是指数据在计算机中的存放形式，它用来反映一个数据的内部构成，即一个数据由哪些数据成分构成，以什么方式构成，呈什么结构，如线性表、树等，本书中不做详细介绍。

1.1.2 计算机语言

计算机语言是人与计算机进行交流的工具。计算机语言分为机器语言、汇编语言和高级语言3种。

对于计算机本身来说，它并不能直接识别由高级语言编写的程序，只能接受和处理由0和1的代码构成的二进制指令或数据，这种直接面向计算机的指令称为“机器语言”。

目前，使用比较广泛的语言，如 Visual C++、Java 以及本书将要介绍的 C 语言等，它



们都被称为计算机的“高级语言”。高级语言使用接近人们习惯的自然语言作为语言的表达形式,学习和操作起来十分方便,并且用高级语言编写的程序具有良好的通用性和可移植性,不依赖于具体的计算机类型。

汇编语言是介于机器语言和高级语言之间的一种语言。

1.1.3 算法的概念

学习提示

【掌握】算法的描述方法

我们知道,“确定算法”是进行程序设计过程中一个相当重要的步骤,那么究竟什么是算法呢?

1 算法的概念

不是只有计算的问题才有算法。广义上讲,算法是为了解决一个问题而采取的方法和步骤。例如,描述跆拳道动作的图解就是跆拳道的算法;一首歌曲的乐谱也可以称为该歌曲的算法,因为它指定了歌曲演奏的每一个步骤,按照此步骤就能演奏出预定的乐曲。

计算机科学中的算法是指为解决某个特定问题而采取的确定且有限的步骤,它是为了解决“做什么”和“怎么做”的问题。著名科学家沃思(Nikiklaus Wirth)曾提出一个公式:

数据结构 + 算法 = 程序

其中,数据结构是对数据的描述,也就是在程序中数据的类型和组织形式,而算法则是对操作步骤的描述。

2 算法的描述

算法是程序设计中非常重要的概念,它的处理对象是数据。有了算法,就可以用任何一种计算机高级语言将其转换为程序。看到这里,读者可能会想:算法既然这么重要,那算法是用什么方法来描述的呢?下面就解开答案。其实,算法可以用各种描述方法进行描述,目前最常用的有3种:伪代码、流程图和N-S结构图。

伪代码是一种近似高级语言但又不受语法约束的语言描述方法,这种方法比较易于理解,但描述较冗长。

流程图是一种很好的描述算法的工具,传统的流程图由图1-1所示的几种基本图形组成。



图 1-1 流程图基本构成图形

用传统流程图表示算法的优点是形象直观、简单方便;缺点则是这种流程图对于流程线的走向没有任何限制,可以任意转向,描述算法时费时费力且不易阅读。

N-S 结构图是由美国学者 I. Nassi 和 B. Shneiderman 在 1973 年提出的。这种流程图完全去掉了流程线,算法的每一步都用一个矩形框来表示,把一个个矩形框按执行的次序连接起来就是一个算法描述。

3 算法的特性

一个算法应该具有以下几个特点：

- 有穷性；
- 确定性；
- 有零个或多个输入；
- 有一个或多个输出；
- 有效性。

1.1.4 结构化程序设计

结构化程序主要由以下 3 种基本控制结构组成，在后面的章节中将做详细的介绍。

1 顺序结构

顺序结构是最基本的算法结构，当执行由这些语句构成的程序时，将按这些语句在程序中的先后顺序逐条执行，没有分支，没有转移，没有步骤之间的相互约束，没有对某一步骤的多次使用，完全按照步骤的原有次序依次执行。顺序结构可用图 1-2 所示的流程图表示。其中图 1-2 (a) 是一般流程图，图 1-2 (b) 是 N-S 结构图。

2 选择结构

选择结构根据不同的条件去执行不同分支中的语句。选择结构可用图 1-3 所示的流程图表示，其中图 1-3 (a) 是一般流程图，图 1-3 (b) 是 N-S 结构图。

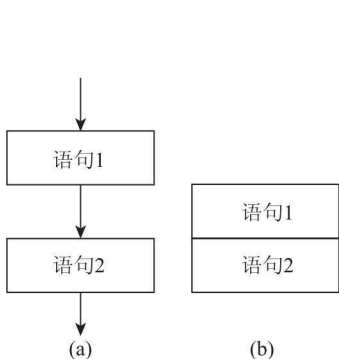


图 1-2 顺序结构

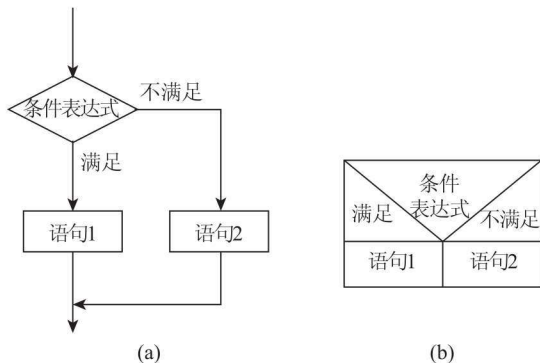


图 1-3 选择结构

3 循环结构

循环结构就是根据各自的条件，使同一组语句重复执行多次。循环结构的流程如图 1-4 和图 1-5 所示。图 1-4 是当型循环，这种循环的特点是：当指定的条件满足（成立）时，就执行循环体；否则就不执行。图 1-5 是直到型循环，该循环的特点是：执行循环体，直到指定的条件满足（成立），就不再执行循环。

由以上 3 种基本结构组成的算法结构可以解决任何复杂的问题，由它们所构成的算法称为结构化算法，由它们所构成的程序称为结构化程序。

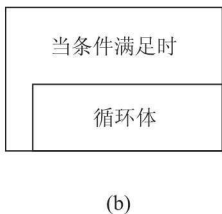
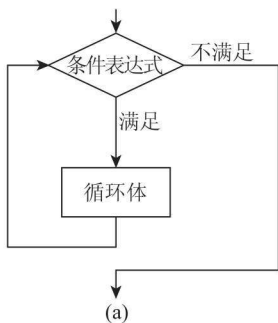


图 1-4 当型循环

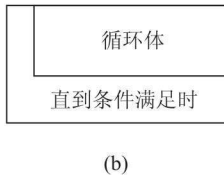
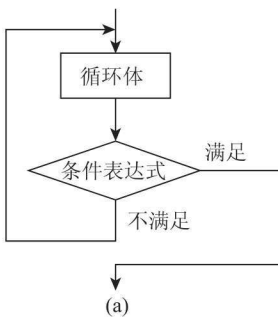


图 1-5 直到型循环

1.2 Visual C ++ 6.0 集成开发环境简介

Visual C++ 6.0（以下简称 VC 6.0）为用户提供了一个集成开发环境，它使得 C 语言程序的编辑、编译、连接、调试和运行等工作都能够在统一的操作界面下完成。下面介绍 VC 6.0 的使用方法。

1.2.1 VC 6.0 的启动

VC 6.0 的启动有以下两种方法。

(1) 双击桌面上“Microsoft Visual C++ 6.0”的快捷图标，如图 1-6 所示，即可进入 VC 6.0 的集成开发环境。

(2) 通过单击桌面左下角的“开始”按钮，弹出“开始”菜单，在“程序”子菜单中选择“Microsoft Visual Studio 6.0”选项，最后单击其中的“Microsoft Visual C++ 6.0”，如图 1-7 所示，也可以进入 VC 6.0 的集成开发环境。



图 1-6 第 1 种启动方法

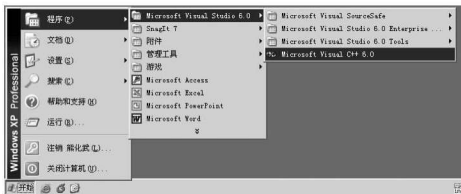


图 1-7 第 2 种启动方法