

中国高等职业教育计算机教育课程体系规划教材

C语言 程序设计教程

C YUYAN CHENGXU SHEJI JIAOCHENG

郝玉秀 主编

刘文宏 米西峰 李 旭 副主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

中国高等职业教育计算机教育课程体系规划教材

C 语言程序设计教程

主 编	郝玉秀		
副主编	刘文宏	米西峰	李 旭
参 编	左永文	陈麟珠	
主 审	栾学钢		

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书共分 8 章,介绍了 C 语言程序设计的基本知识,并配以实际应用案例和模拟练习,使所学理论运用到实际应用中。本书注重培养学生程序设计的基本技能和素养,书中的所有案例都是经过反复推敲提炼而成的,贴近生活、丰富有趣,可以调动学生的积极性,使枯燥的理论学习变得生动有趣,从而达到学好 C 语言的目的。

本书的所有程序均在 Turbo C for windows 环境下调试运行通过,随书提供电子课件和源程序,方便教师组织教学和学生自主学习使用。

本书适合作为高职高专院校 C 语言程序设计的教材,也可作为其他计算机程序设计课程的教材或自学参考教材。

图书在版编目(CIP)数据

C 语言程序设计教程 / 郝玉秀主编. —北京: 中国铁道出版社, 2011. 8
中国高等职业教育计算机教育课程体系规划教材
ISBN 978-7-113-13084-8

I. ①C… II. ①郝… III. ①C 语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 145278 号

书 名: C 语言程序设计教程

作 者: 郝玉秀 主编

策划编辑: 翟玉峰

责任编辑: 翟玉峰

编辑助理: 何 佳

封面设计: 付 巍

责任印制: 李 佳

读者热线: 400-668-0820

封面制作: 白 雪

出版发行: 中国铁道出版社(北京市宣武区右安门西街 8 号 邮政编码: 100054)

印 刷: 北京东君印刷有限公司

版 次: 2011 年 8 月第 1 版 2011 年 8 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 12 字数: 281 千

印 数: 1~3 000 册

书 号: ISBN 978-7-113-13084-8

定 价: 22.00 元

版权所有 侵权必究

凡购买铁道版图书,如有印制质量问题,请与本社教材研究开发中心批销部联系调换。

中国高等职业教育计算机教育课程体系规划教材

编审委员会

主任：高 林

副主任：鲍 洁 丁桂芝 袁 玫

樊月华 严晓舟

编 委：（按姓氏笔画排序）

于 京 王 辉 方风波 叶曲炜

史宝会 刘凤玲 孙永道 李 红

杨欣斌 周 源 郝玉秀 姜 波

秦绪好 徐 红 褚建立 翟玉峰

序

PREFACE

《国家中长期教育改革和发展规划纲要》文件指出,职业教育要面向人人、面向社会,着力培养学生的职业道德、职业技能和就业创业能力。到 2020 年,形成适应经济发展方式转变和产业结构调整要求,体现终身教育理念,中等和高等职业教育协调发展的现代职业教育体系,满足人民群众接受职业教育的需求,满足经济社会对高素质劳动者和技能型人才的需要。

高等职业教育肩负着培养生产、建设、服务和管理第一线高素质技能型专门人才的重要使命,对经济社会发展的贡献具有独特作用。近 10 多年来,我国高等职业教育规模迅速扩大,为实现高等教育大众化发挥了积极作用。同时,高等职业教育也主动适应社会需求,坚持以服务为宗旨,以就业为导向,走产、学、研结合的发展道路,切实把改革与发展的重点放到加强内涵建设和提高教育质量上来,力争为我国全面建设小康社会和构建社会主义和谐社会,建设人力资源强国做出更大贡献。自 1998 年以来,我国高职培养的毕业生已超过 1 300 万人,为经济社会各行各业生产和工作第一线培养了大批高素质技能型专门人才。目前,全国高等职业院校共有 1 200 余所,年招生规模为 310 多万人,在校生有 900 多万人;高等职业院校招生规模占到了普通高等院校招生规模的一半,已成为我国高等教育的“半壁江山”。

在这种新的改革目标和方向的指引下,在《中国高职院校计算机教育课程体系 2007》(简称 CVC 2007)的基础上,全国高等院校计算机基础教育研究会与中国铁道出版社再度联手,邀请了有关的知名教育专家、行业企业专家和具有丰富教学经验的高职教师参加,吸取了 CVC 2007 的成功经验,并紧密结合新的教学改革理念,推出《中国高等职业教育计算机教育课程体系 2010》(简称 CVC 2010)。CVC 2010 进一步贯彻了高职课程改革的指导思想,在学习借鉴世界各国职业教育成功经验的基础上,逐步探索形成了适合中国国情的、具有中国特色的高职教育课程开发方法。

在 CVC 2010 中,提出了高等职业教育中计算机类专业和非计算机专业计算机教育改革的指导思想,阐述了以下三个高职人才培养模式和课程开发方法:

(1)“基于岗位分析和学期项目主导的专业课程模式”。该专业课程模式建立在对职业岗位分析的基础上,学期项目是配合职业岗位的工作能力要求为每一个学期设计的典型工作任务,课程学习由学期项目主导,为学期项目做理论和技术的支撑。

(2)“职业竞争力导向的工作过程—支撑平台系统课程模式和开发方法”。其主导思想是:职业竞争力导向,职业分析具有新特点,提出专业课程体系的基本结构,提出依据课程目标可以将科目课程分为三种基本类型(即相对系统的专业知识性理论课程、基本技术技能的训

练性实践课程、理论—实践一体化性质的综合课程),把获取职业资格证书融入课程设计,借鉴国外先进职业教育思想,适应国情,体现中国特色。

(3)“非计算机专业计算机教育中的课程开发原则”。非计算机专业计算机教育的教学目标已不再局限于解决简单的操作计算机的问题,而是要使学生具备在计算机及网络环境中完成职业工作的能力,全面提升学生的综合信息素质,以适应社会的需求。

根据 CVC 2010 中阐述的指导思想,我们将科目课程分为三种基本类型,并据此进行教材建设。

(1)专业知识性理论课程教材,又称知识与方法性课程教材,是支撑完成工作任务所需的重要理论基础,与之对应的是“启发式”的教学方法。

(2)训练性实践课程教材,主要是为培养学生实践能力的实训课程服务的,其目的是掌握完成工作任务所需要的基本技术技能。课程中可能涉及的技术理论、方法、规范等内容是课程的重要组成部分,但不是课程的最终目标。实训课程要贯彻“做中学”的教育理念。

(3)理论—实践一体化性质的综合课程教材,是指导学生综合应用理论知识、基本技术技能完成一个完整工作任务的教材,理论—实践一体化课程要贯彻“做中学”的教育理念,并采用行动导向的教学法。

参与本套教材编写工作的作者都是从事高职计算机教育、具有丰富教学经验的优秀教师,他们对高职教育有较深入的研究。希望通过本套教材的出版,为推广高等职业教育教学改革成果,实现优秀教材资源共享,提高教学质量,向社会输送更多高素质技能型专门人才做出更大贡献。

欢迎大家在使用教材的过程中提出宝贵意见。

高林

2011年6月

前言

FOREWORD

随着计算机技术的不断发展和日益普及,学习 C 语言程序设计,是目前国内外计算机技术基础教育的重要组成部分之一。熟练掌握 C 语言的程序开发技术,也是 21 世纪社会对信息技术应用类人才的要求之一。

《C 语言程序设计教程》是根据《中国高等职业教育计算机教育课程体系 2010》中的“程序设计基础——C 语言程序设计”课程方案编写的。本书共分 8 章介绍了 C 语言程序设计的基本知识,每章都配以实际应用案例、模拟练习和大量的课后练习,使所学理论能运用到实际应用中。本书是在作者多年教学经验基础上,根据学生的认知规律精心组织编排的,具有高等教育层次知识系统性的特点和职业教育类型能力系统性的特点,注重培养学生程序设计的基本技能和基本素养。书中的所有案例都是经过反复推敲提炼而成的,贴近生活、丰富有趣,可以调动学生的积极性,使枯燥的理论学习变得生动趣味,从而达到学好 C 语言程序设计的目的。

本书在内容编排上注意由简到繁、由浅入深、循序渐进,理论和实际紧密结合,力求通俗易懂、简洁实用,对于 C 语言中过时的、不太常用的知识内容进行了大胆的删除,如:条件编译、共用体、联合、位操作、typedef 定义、链表等。

本书中的所有程序均在 Turbo C for windows 环境下调试运行通过。提供电子课件和源程序,方便教师组织教学和学生自主学习使用。

本书可作为高职高专院校 C 语言程序设计教材,也适合其他各类教育作为计算机程序设计课程的教材或自学参考教材。

本书由郝玉秀任主编,刘文宏、米西峰、李旭任副主编,左永文、陈麟珠参加编写,栾学钢担任主审。其中第 1 章由刘文宏编写,第 2、3 章由郝玉秀编写,第 4 章由左永文编写、第 5 章由陈麟珠编写、第 6、7 章李旭由编写,第 8 章及附录由米西峰编写。

由于作者水平有限,不妥之处在所难免,敬请广大读者批评指正,提出宝贵意见。

编 者

2011 年 6 月

目 录

CONTENTS

第 1 章 绪论	1
1.1 程序及程序设计方法	1
1.1.1 程序设计概念	1
1.1.2 C 语言简介	2
1.2 算法与程序基本结构	3
1.2.1 算法与算法描述	3
1.2.2 程序基本结构	6
1.3 开发环境与程序调试	7
习题	13
第 2 章 C 语言程序基础	14
2.1 C 语言数据类型、常量和变量	14
2.1.1 C 语言的数据类型	14
2.1.2 常量和变量	15
2.2 运算符和表达式	17
2.2.1 算术运算符及其表达式	17
2.2.2 赋值运算符及其表达式	18
2.2.3 复合运算符及其表达式	18
2.2.4 关系运算符及其表达式	19
2.2.5 逻辑运算符及其表达式	19
2.3 数据类型的转换	20
2.3.1 自动转换	20
2.3.2 强制转换	21
2.4 优先级和结合性	21
2.5 数据的输入和输出	22
2.5.1 数据的格式输入和输出	22
2.5.2 非格式输入/输出函数	25
2.6 清屏幕函数	26
2.7 光标定位函数	26
习题	28
第 3 章 结构化程序的基本结构	30
3.1 顺序结构	30
3.2 选择结构	31

3.2.1	if 选择结构	32
3.2.2	switch 语句	34
3.3	循环结构	37
3.3.1	for 循环	37
3.3.2	while 循环	40
3.3.3	do-while 循环	40
3.3.4	循环的嵌套	41
3.4	其他控制语句	43
习题		45
第 4 章	数组的应用	49
4.1	一维数组	48
4.1.1	一维数组的定义	48
4.1.2	一维数组的初始化	49
4.1.3	一维数组元素的引用	49
4.2	二维数组	50
4.2.1	二维数组的定义	50
4.2.2	二维数组元素的引用	51
4.2.3	二维数组的初始化	51
4.3	字符数组	52
4.3.1	字符数组的定义	52
4.3.2	字符数组的初始化	53
4.3.3	字符串的输入和输出	53
4.3.4	字符串处理函数	54
4.4	数组的应用	58
4.4.1	数据的检索	58
4.4.2	向数组中插入新数据	61
4.4.3	删除数组中指定的元素	62
4.4.4	数据的排序	63
习题		67
第 5 章	函数的应用	71
5.1	函数的定义和调用	70
5.1.1	函数的定义	71
5.1.2	函数的声明和调用	71
5.1.3	函数的嵌套调用	74
5.1.4	函数的递归调用	75
5.2	变量的作用域和生存期	75
5.2.1	变量的作用域	75
5.2.2	变量的生存期	77

习题	81
第 6 章 指针的应用	83
6.1 指针的概念	84
6.1.1 指针变量的定义	84
6.1.2 指针变量的引用	84
6.1.3 指针变量的运算	85
6.2 指针与数组	87
6.2.1 指向数组元素的指针	87
6.2.2 通过指针引用数组元素	87
6.2.3 指针与二维数组	88
6.3 指针与字符串	90
6.3.1 字符指针的概念与定义	90
6.3.2 字符指针与字符数组	91
6.4 指针数组	91
6.5 指向指针的指针	92
习题	95
第 7 章 图形与动画设计	97
7.1 图形设计	97
7.1.1 图形模式的初始化	97
7.1.2 独立图形运行程序的建立	99
7.1.3 屏幕颜色的设置和清屏函数	100
7.1.4 绘图函数	101
7.1.5 封闭图形的填充	103
7.1.6 设置线型和宽度函数	104
7.1.7 图形模式下文本的输出	105
7.2 简单动画设计	107
7.2.1 用清除法实现动画	107
7.2.2 用存储再现法实现动画	108
7.2.3 用动态窗口法实现动画	109
7.2.4 用页交替法实现动画	110
习题	112
第 8 章 综合训练	113
8.1 系统设计	113
8.1.1 开发一个信息管理系统的过程	114
8.1.2 系统设计方法	114
8.2 学生数据信息结构设计	117
8.2.1 结构体类型的定义和引用	117
8.2.2 设计学生信息结构	122

8.3	系统框架设计	123
8.4	学生数据的存储与重用	131
8.5	系统维护模块的设计	138
8.6	数据查询模块的设计	141
8.7	数据统计模块的设计	143
8.8	报表输出模块的设计	145
附录		164
附录 A	常用字符与 ASCII 码对照表	164
附录 B	运算符的优先级和结合性	166
附录 C	C 语言中的关键字	168
附录 D	常用 C 语言库函数	169
附录 E	C 语言常见错误处理	172
参考文献		179

第 1 章

绪 论

学习目标

- 了解程序设计的基本概念；
- 了解 C 语言的发展；
- 掌握程序算法的设计及其表示方法；
- 了解 C 程序的结构；
- 熟悉 C 程序的运行环境。

1.1 程序及程序设计方法

1.1.1 程序设计概念

1. 程序

计算机中的程序是指为解决某一问题而向计算机发出的一连串的操作命令。操作的对象是数据，目的是对数据进行加工处理，以得到想要的结果。利用计算机解题要借助一定的程序设计方法和程序设计语言。

2. 程序设计语言

程序设计语言是程序设计人员和计算机进行信息交流的工具。它遵循一定的规则和形式，程序设计要在一定的语言环境下进行。

程序设计语言分为机器语言、汇编语言和高级语言三种。

机器语言：计算机硬件能直接识别和执行的指令系统（用二进制代码表示）。其特点是：占用资源少、运行速度快、效率高；但可读性和可移植性差、不易编写、调试和查错；主要用于编写计算机最底层的核心系统程序。

汇编语言：用助记符号（英文单词的缩写）代替机器语言的二进制代码，也称符号式的机器语言。其特点是：较机器语言的可读性、编写效率和质量有所提高，但不能直接被计算机执行，需要有编译系统将其翻译成机器语言才能执行，所以执行效率有所降低，且必须熟悉计算机的硬件，记忆的指令繁多，不便于普及。例如，Z80、8086（宏汇编）。

高级语言：用接近自然语言和数学公式的形式编写程序的计算机语言，完全脱离硬件系统，是面向科学计算和实际问题的语言。其特点是：可读性和可移植性好，易于理解和调试修改，但运行效率较汇编语言低。最早的高级语言有 FORTRAN、Pascal、BASIC、C 语言等。

3. 程序设计方法

程序设计方法分为面向过程的结构化程序设计方法和面向对象的程序设计方法两种。

面向过程的结构化程序设计方法：自顶向下、逐步求精。其程序结构是按功能划分为若干个基本模块，这些模块形成一个树状结构；各模块之间的关系尽可能简单，在功能上相对独立；每一模块内部均是由顺序、选择和循环三种基本结构组成；其模块化实现的具体方法是使用函数。

结构化程序设计的不足：

- 以算法为中心，因此功能的多变性和算法的不唯一性增加了程序的维护和测试工作量。
- 把数据和处理数据的过程分离为相互独立的实体。当数据结构变化时，所有相关的处理过程都需要进行相应的修改，程序的可重用性差。
- 由于图形用户界面的应用，使得软件使用起来越来越方便，但开发起来却越来越困难。

面向对象程序设计方法：它将数据及对数据的操作方法放在一起，作为一个相互依存、不可分离的整体——对象。对同类型对象抽象出其共性，形成类。类中的大多数数据，只能用本类的方法处理。类通过一个简单的外部接口与外界发生关系，对象与对象之间通过消息进行通信。

1.1.2 C 语言简介

1. C 语言的发展

C 语言是一种面向过程的程序设计语言。其前身是 ALGOL60，1960 年英国剑桥大学和伦敦大学将 ALGOL60 发展成 CPL，1967 年英国剑桥大学的 Martin Richards 将 CPL 改写成 BCPL；1970 年美国贝尔实验室的 Ken Thompson 将 BCPL 修改成 B 语言，并用 B 语言开发了第一个基于高级语言的 UNIX 操作系统；1972 年 Ken Thompson 与开发 UNIX 时的合作者 D.M.Ritchit 一起将 B 语言做了改进推出了 C 语言，1978 年 B.W.Kernighan 和 D.M.Ritchit 合著了著名的 *The C Programming Language* 一书，该书是 C 语言版本的基础，1983 年美国国家标准协会 ANSI (American National Standards Institute) 制定了 C 语言标准，1987 年开始实施 ANSI C。

2. C 语言版本

目前，最流行的 C 语言有以下几种：

Microsoft C 或称 MS C、Borland Turbo C 或称 Turbo C、AT&T C。

这些 C 语言版本不仅实现了 ANSI C 标准，而且在此基础上各自作了一些扩充，使之更加方便、完美。

3. C 语言的特点

① C 语言简洁、紧凑，使用方便、灵活。ANSI C 一共有 32 个关键字（关键字均为小写），9 种控制语句，且程序书写自由，主要用小写字母表示，压缩了一些不必要的成分。

② C 语言运算符丰富，一共有 34 种运算符。C 语言把括号、赋值、逗号等都作为运算符处理，可以实现其他高级语言难以实现的运算。

③ C 语言数据结构类型丰富，支持各种高级语言普遍使用的数据类型，且允许用基本的数据类型构造复杂的数据类型。

④ C 语言具有强大的图形功能，支持多种显示器和驱动器，且计算功能、逻辑判断功能比较强大。

⑤ C 语言允许直接访问物理地址,能进行位 (bit) 操作,能实现汇编语言的大部分功能,可以直接对硬件进行操作,因此有人把它称为中级语言。

⑥ C 语言生成目标代码质量高,程序执行效率高,程序可移植性好。

1.2 算法与程序基本结构

通常,一个程序应包括:

- 对数据的描述:在程序中要指定数据的类型和数据的组织形式,即数据结构 (Data Structure)。
- 对操作的描述:即操作步骤,也就是算法 (Algorithm)。

Nikiklaus Wirth 提出的公式:数据结构+算法=程序。

完整地说应为:程序=算法+数据结构+程序设计方法+语言工具和环境。

这 4 个方面是一个程序设计人员所应具备的知识。

1.2.1 算法与算法描述

1. 算法的概念

做任何事情都有一定的步骤。为解决某一个问题而采取的某种方法和步骤,称之为算法。

2. 算法的特性

算法是对特定问题求解步骤的一种描述,是指令的有限序列,其中每一条指令表示一个或多个操作。算法具有如下特点:

- 有穷性:一个算法应包含有限的操作步骤而不能是无限的;
- 确定性:算法中每一个步骤应当是确定的,而不能是含糊的、模棱两可的;
- 有效性:算法中每一个步骤应当能有效地执行,并得到确定的结果;
- 输入:有零个或多个输入;
- 输出:有一个或多个输出。

3. 算法表示法

算法有四种表示法。

(1) 自然语言描述法

特点:通俗易懂,但文字冗长,且易出现“歧义性”。

【例 1-1】对一个大于或等于 3 的正整数,判断它是不是一个素数。

算法可表示如下:

S1: 输入 n 的值;

S2: $i=2$;

S3: n 被 i 除,得余数 r ;

S4: 如果 $r=0$,表示 n 能被 i 整除,则打印 n “不是素数”,算法结束;否则执行 S5;

S5: $i+1 \rightarrow i$;

S6: 如果 $i \leq n-1$,返回 S3; 否则打印 n “是素数”; 然后算法结束。

改进:

S6: 如果 $i \leq \sqrt{n}$, 返回 S3; 否则打印 n “是素数”; 然后算法结束。

(2) 几何图形表示法

ANSI 图表示法：由一些特定意义的图形、流程线及简要的文字说明构成，它能清晰明确地表示程序的运行过程，ANSI 图又称流程图。

特点：直观形象，易于理解。

ANSI 图通常采用以下几种符号表示，如图 1-1 所示。



图 1-1 ANSI 图

起止框：表示程序流程的开始或结束。

输入/输出框：表示输入或者输出数据，框内可注明数据名、来源、用途或其他的文字说明。

处理框：表示计算或处理功能，用来执行一个或一组特定的操作，框内可注明处理名或其简化功能。

判断框：表示判断或开关，菱形内可注明判断的条件。它只有一个入口，但可以有若干个可供选择的出口，在对条件求值后，有一个且仅有一个出口被激活。求值结果可在表示出口路径的流线附近写出。

流向线：表示控制流的流线，流线的标准流向是从左到右和从上到下。一般情况下，流线应从符号的左边或顶端进入，并从右边或底端离开，其进出点均应对准符号的中心。

连接点：用于将画在不同位置的流程线连接起来，用圆圈表示，圆圈中标注连接点的序号。用连接点，可以避免流程线的交叉或过长，使流程图清晰。

【例 1-2】求某班某同学 3 门课程数学（xs）、英语（yy）、计算机（jsj）的总分（zf）。用 ANSI 图表示如图 1-2 所示。

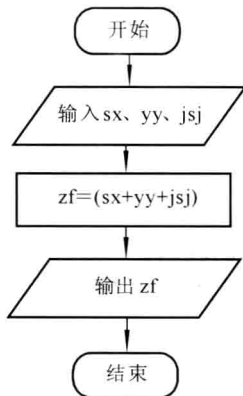


图 1-2 例 1-2 的 ANSI 图

N-S 图表示法：N-S 图是无线的流程图，它把整个程序写在一个大框图内，这个大框图由若干个小的基本框图构成，N-S 图又称盒图。

特点：取消了流程线，即不允许流程任意转移，只能从上到下顺序进行。

N-S 图分为三种基本结构：顺序结构、选择结构和循环结构，如图 1-3 所示。

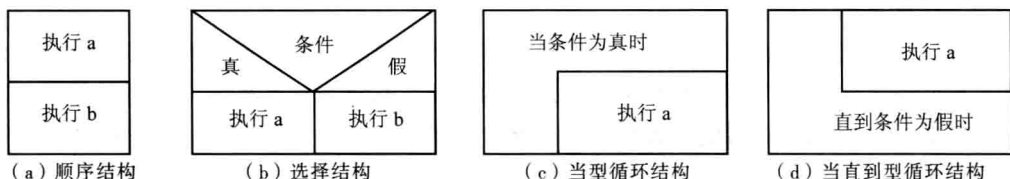


图 1-3 N-S 图三种结构

三种基本结构的共同特点：

- 只有一个入口；
- 只有一个出口；
- 结构内的每一部分都有机会被执行到；
- 结构内不存在“死循环”。

对于一般简单的问题用顺序结构或选择结构就能完成，但对复杂问题往往需要用这三种基本结构的相互组合来完成。

对于一个程序设计人员，必须熟悉设计算法，并根据算法写出程序。

【例 1-3】输入 50 个学生的成绩，统计出不及格人数，用 N-S 图表示该算法，如图 1-4 所示。

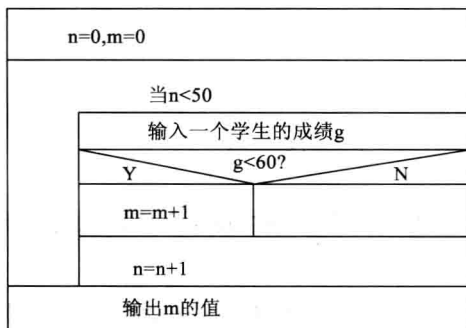


图 1-4 例 1-3 的 N-S 图

(3) 伪代码表示法

用一种介于自然语言和计算机语言之间的文字和符号来描述算法。

【例 1-4】输入 50 个学生的成绩，统计出不及格人数，用伪代码表示该算法。

```
n=0
m=0
while n less than 50
input g
if g less than 60 then m=m+1
n=n+1
while end
output m
```

伪代码没有统一语法，形式比较灵活，只要能看懂就可以，甚至也可以用文字描述。伪代码与计算机语言比较接近，因此可以很容易地转换成计算机程序。

(4) 计算机语言表示法

用某种计算机语言表示算法，即计算机能够执行的算法。让计算机完成某项任务，也就是用计算机实现算法。计算机无法识别流程图和伪代码，只有用计算机语言编写的程序经过编译系统转换成目标代码（机器语言程序）才能被计算机执行。因此，在用流程图或伪代码描述出一个算法后，还要将它转换成计算机语言程序。用计算机语言表示算法必须遵循所用语言的语法规则。

【例 1-5】用 C 语言表示输入三个整数，求出最大数和最小数的算法。

```
main()
{
    int a,b,c,max,min;
    printf("input three numbers: ");
    scanf("%d%d%d",&a,&b,&c);
    if(a>b)
    {max=a; min=b;}
    else
    {max=b; min=a;}
    if(max<c)
    max=c;
    else
    if(min>c)
    min=c;
    printf("max=%d\nmin=%d",max,min);
}
```

编写出 C 程序，只是描述了算法，并不是实现了算法，只有运行程序才能实现算法。

4. 算法设计的目标

正确性：在输入合法的数据时，能在有限的运行时间内，得出正确的结果；

可读性：方便阅读和交流；

健壮性：当有非法的数据时，能正确做出反应和处理，不会出现莫名其妙的结果；

高效和低存储性：执行时间短，存储需求低。

1.2.2 程序基本结构

1. C 程序的基本结构

【例 1-6】求圆的周长和面积的 C 程序。

```
#include "stdio.h"
main()
{
    float r,cl,cs; /*定义三个实型变量*/
    printf("请输入圆的半径:");
    scanf("%f",&r);
    cl=2*3.14159*r; /*计算周长*/
    cs=3.14159*r*r; /*计算面积*/
}
```