



“十二五”普通高等教育本科国家级规划教材
普通高等教育精品教材
国家精品课程主讲教材

程序设计基础

Fundamentals of Programming

(第2版)

(Second Edition)

张长海 陈娟



高等教育出版社
HIGHER EDUCATION PRESS

014003454

TP311.1-43
13-2



“十二五”普通高等教育本科国家级规划教材
普通高等教育精品教材
国家精品课程主讲教材

程序设计基础

Chengxu Sheji Jichu

(第2版)

张长海 陈娟



TP311.1-43
13-2



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING



北航 C1691256

016003424

内容提要

本书以 C 语言为载体, 引进 PAD 表示程序逻辑, 阐述基本的程序设计方法。全书共分 14 章, 主要内容包括算法、程序设计方法、函数、数据的组织、程序开发和结构化程序设计。每章都包含大量例题和习题。

本书最大的特点是“以计算思维为主导”、“以授之以渔为宗旨”、“以程序设计为主线”, 自始至终在讲“程序设计”, 而不是讲“语言”, 摒弃了很多程序设计书中流行的主要解释程序设计语言的做法。本书的另一个特点是“以案例为驱动”, 全书在案例的驱动下讲述程序设计。书中所有例题和习题全部以构造算法为基础, 全部为“讲述程序设计”, 而不是“解释语言概念”。本书的目的是教会读者怎样编写程序, 提高读者的程序设计能力和计算思维能力, 彻底改变“学了程序设计而不会编程”的现状。

本书整体结构清晰, 图文并茂, 力求体现“结构化程序设计”思想, 注重培养和训练良好的程序设计风格。

本书可作为高等学校“高级语言程序设计”、“程序设计基础”、“C 程序设计”等课程的教材和参考书, 也可以供从事计算机工作的有关人员阅读参考。

图书在版编目(CIP)数据

程序设计基础/张长海, 陈娟编著. --2 版. --北京: 高等教育出版社, 2013. 9

ISBN 978-7-04-038327-0

I. ①程… II. ①张… ②陈… III. ①

C 语言-程序设计-高等学校-教材 IV. ① TP312

中国版本图书馆 CIP 数据核字(2013)第 199018 号

策划编辑 武林晓

责任编辑 武林晓

封面设计 于涛

版式设计 杜微言

插图绘制 尹莉

责任校对 王雨

责任印制 田甜

出版发行 高等教育出版社

社址 北京市西城区德外大街 4 号

邮政编码 100120

印刷 三河市吉祥印务有限公司

开本 787mm×1092mm 1/16

印张 26.75

字数 600 千字

购书热线 010-58581118

咨询电话 400-810-0598

网 址 <http://www.hep.edu.cn>

<http://www.hep.com.cn>

网上订购 <http://www.landrace.com>

<http://www.landrace.com.cn>

版 次 2008 年 7 月第 1 版

2013 年 9 月第 2 版

印 次 2013 年 9 月第 1 次印刷

定 价 36.00 元

本书如有缺页、倒页、脱页等质量问题, 请到所购图书销售部门联系调换

版权所有 侵权必究

物料号 38327-00

第2版前言

本书第1版自2008年出版至今已经历了5个年头。5年来我国高等教育发生了很大变化,各高校的程序设计教学也发生了深刻的变化。目前“以程序设计为主”的教学思想被广泛接受,并逐步深入到讲授程序设计课程的教师心中。5年来“计算思维”思想被提出并很快得到广泛的关注和认可,程序设计课要培养学生“计算思维”能力的思想被广泛接受。“计算思维”不是狭义的计算机编程,而是运用计算的基本思想去求解问题、设计系统和理解人类的行为,如同读、写、算能力一样,它是所有人都必须具备的思维能力。本书作为“普通高等教育精品教材”,经过多年的广泛使用,在“以程序设计为主线”和“培养学生计算思维能力”方面起到一定的积极作用,并取得了一系列丰硕的成果。

2007年本书被遴选为普通高等教育“十一五”国家级规划教材。

2009年以本书为主教材的吉林大学“高级语言程序设计”课程被评为“国家级精品课”。

2010年吉林大学“高级语言程序设计”教学团队被评为“吉林省优秀教学团队”。

2011年本书被教育部评为“普通高等教育国家级精品教材”。

2011年本书被遴选为“十二五”普通高等教育本科国家级规划教材。

在这种形势下有必要对第1版进行修订,以反映近年来的教学改革成果。本版保持了第1版的特色,并且更加明确了“以程序设计为主线”和“培养学生计算思维能力”的宗旨,进一步提升了以“计算思维”主导的程序设计教学思想。具有如下鲜明特色:

(1)更加明确了培养学生“计算思维”能力的教学思想。计算机的出现和使用已经深刻地影响了人类的思维方式和思维习惯。本书的目的是教会读者怎样编写程序,怎样以“计算”的方式去思考和解决问题,怎么像计算机科学家那样思维,彻底改变“学了程序设计而不会编程序”的现状,提高读者的程序设计能力和计算思维能力。

(2)更加明确了“以程序设计为主线”的指导思想。按程序设计的思路组织全书内容,真正地讲授程序设计,而非语言,摒弃了很多程序设计书中流行的主要解释程序设计语言的做法。

(3)以“授之以渔”为理念,而非“授之以鱼”。针对程序设计的每个知识模块都采取

“提出有意义的问题→设计算法→分析算法特点→编出程序→运行程序→

介绍使用的C语言成分→配合讲述大量例题→课后习题”

模式组织教学内容,目的是教会学生怎样编写程序,而不是让学生去背几个语法规则。

(4)以“案例”为驱动,突出实践。通过大量案例讲解程序设计思想和方法。彻底改变了过去那种单纯解释语法、语义等语言成分的做法;改变了那种提出一个语言概念并解释,然后以一个小程序说明相应概念的做法。而是使程序设计始终贯穿于整个教学过程中。书中案例

多为与实际相关、实用的题目。使用案例的目的不是为了解释语言概念，而是为了讲述程序设计方法、构造算法，以训练学生的实际编程能力、计算思维能力为目标。

使用本书，一定要把握“以培养学生计算思维能力为宗旨”、“以培养学生程序设计能力为目标”这一原则。一定要“以程序设计为主线”，一定要讲程序设计，而不要回到解释语言的老套路上去。

本书第1版面向985等院校，内容偏难。应二本、三本等院校的需求，第2版与第1版相比，难度大大降低。尽管如此，本书内容中仍然有偏难的部分。书中带星号“*”的章节，其内容或者不常用，或者不影响讲授基本程序设计思想，或者难度偏大，例如各章的“程序设计实例”中带星号“*”的题目难度可能偏大。这部分内容可以根据情况选择讲授。

本书第2版的第1、2、3、4、5、6、8、10章由张长海执笔，第7、9、11、12、13、14章由陈娟执笔。最后由张长海统稿。

限于作者水平，书中难免有疏漏和不妥之处，敬请广大读者批评指正。

作者

2013年5月于长春

第1版前言

本书应用于“高级语言程序设计”、“程序设计基础”、“C语言程序设计”等课程。学好这类课程既可以为后续课程打下良好的基础，又会对学生一生的程序设计技术、技巧、风格和习惯产生深远影响。

随着计算机技术的不断发展和计算机应用的日渐普及，人们对于大学第一门程序设计课程的认识也在不断更新。

最早，在20世纪70年代及以前，计算机应用范围十分狭窄，所谓“搞计算机”的人也很少。那时的计算机专业是在研究计算机本身，相应的第一门程序设计课程称为“算法语言”。学习这门课程的目的是学习程序设计语言本身（绝大部分学校都选讲ALGOL₆₀），而对于如何使用这种语言不太关心。因为那时的教学目的是让学生研究计算机本身，是要制造计算机，是要透彻地研究语言本身的语法和语义，并实现它。

时间已经进入21世纪，今天计算机已经渗透到各个领域，甚至已经普及到家庭，它像电视机、冰箱一样，是家用电器之一。计算机专业的规模与30年前大不一样。据2005年统计，全国开设“计算机科学与技术”专业的院校有741所，“计算机科学与技术”专业在校生生人数超过45万；目前保守估计，全国开设“计算机科学与技术”专业的院校超过800所，“计算机科学与技术”专业在校生生人数超过50万。再加上“软件工程”、“计算机网络”、“信息技术”等计算机类的专业，全国的计算机专业数和在校生生人数将比800所和50万多得多。面对如此庞大的队伍，我们的毕业生不可能都去研究计算机本身，社会也不需要这么多人从事计算机研究。所以现在计算机专业的含义已经完全与30年前的计算机专业不同了，现在绝大部分计算机专业人员主要是做“计算机应用”，甚至是在“应用计算机”，而不是在研究计算机本身了。

在这种形式下，计算机专业怎么办？作为计算机专业的第一门程序设计课程怎么讲？摆在每个组织计算机专业教学的人的面前，更摆在教授第一门程序设计课程的教师面前。显然，目前计算机教育应该面向应用，第一门程序设计课程也不应该再去讲程序设计语言本身了，而应该讲述程序设计语言的应用，即应该“真正地讲程序设计”，把授课重点从讲授语言的符号、语句等语言成分上转移到讲授程序设计上，教学的目的是教会学生“怎么编程序”，而不是背几个语言符号、说明、语句等，应该把这门课程的名字从“xxx语言”改为“高级语言程序设计”或“程序设计基础”，把授课重点真正放在程序设计上。尤其是一些非计算机专业，更应该跳出讲授“语言”的误区，因为他们更是在“应用计算机”，是用计算机解决本专业领域的问题，更没有必要去背那些死的语言符号、语法概念，更应该学会怎么编程序。

本书秉承上述面向应用的主导思想，依托C语言讲述“程序设计”，重点在于程序设计。

在这种思想的指导下,本书对C语言本身采取“有所取、有所不取”的策略。对于那些常用的语言成分、直接与讲述程序设计方法有关的语言成分,穿插在程序设计过程中详细、准确地介绍;对于那些与程序设计方法联系不太紧密,但是还常用的部分,放在最后简单介绍;而对于那些与讲述程序设计方法关系不太大,也不常用的部分则根本不涉及。

本书主要作者从事计算机教学30余年,讲授过10多门计算机方面的课程,是吉林大学计算机科学与技术学院“高级语言程序设计”课程梯队负责人;主持和参加过许多大型项目的研发,并亲自参加这些项目的程序编码工作;曾20余次为吉林大学计算机科学系、计算机科学与技术学院、软件学院本科生主讲“程序设计”课程;对程序设计课程的教学内容和讲授方法有独到见解。本书是作者30余年教学实践的总结。

作为大学本科计算机专业基础课教材,本书具有以下特点;

(1) 最大特点是“以程序设计为主线”。按照程序设计的思路组织全书内容,真正在讲授程序设计,而不是讲语言,摒弃了目前各种程序设计书中流行的以“解释程序设计语言”为主的做法。

(2) 全书整体结构好,章节安排合理,注意由浅入深地介绍程序设计知识。比如有关函数的知识,由浅入深地分4章逐步介绍;有关指针的知识也分散到各个章节中介绍。免得集中于一章,使读者学习起来枯燥乏味,难以接受。

(3) 全书自始至终贯穿结构化程序设计思想,所有例题都具有良好的结构和程序设计风格。其目的是给读者一个示范,使读者从开始学习程序设计起就养成一个良好的程序设计习惯和风格。

(4) 本书图文并茂,引进PAD表示程序逻辑。PAD的结构比传统的流程图、NS图等都好,同时也比直接用程序表示算法更为直观,易于理解。

(5) 全书配备大量例题和习题,并且全部为程序设计题目。例题的讲解从构造算法出发,以训练读者的编程能力为目标;概念、语言成分的介绍穿插在程序设计之中。本书中大量的习题可供读者做练习和进一步提高使用。

全书共14章,大致分为4部分。

第一部分基本知识,包括第1章,介绍计算、算法和程序设计的基本概念。

第二部分程序设计,包括第2、3、4、5、9、10、12章。第2章简单程序设计,介绍顺序程序设计、数据及其类型、表达式、赋值、输入输出;第3章介绍分支程序设计;第4章介绍循环程序设计;第5章简单介绍模块化程序设计思想,引进子程序和函数的概念;第9章进一步介绍函数,讲述参数、作用域;第10章介绍递归程序设计;第12章介绍程序开发和结构化程序设计,包括结构化程序设计原则、程序风格、“自顶向下、逐步求精”的程序设计技术。

第三部分数据组织,包括第6、7、8、11、13章。第6章讲述批量数据组织——数组;第7章介绍指针;第8章讲述复杂的表单数据,引进结构体和共用体;第11章讲述外部数据组织——文件及其操作;第13章讲述动态数据组织及其在程序设计中的应用。

第四部分为第14章,介绍若干较深入的问题及C语言独有的特性。进一步介绍函数,讲

述函数作参数、函数副作用等；以及存储类别、位操作、`break` 语句、编译预处理等。

本书第1、2、3、4、5、6、8、10章由张长海执笔，第7、12、13、14章由陈娟执笔，第9、11章由刘磊执笔。最后全书由张长海统稿。

在本书的编写过程中，作者参阅并借鉴了国内外诸多同行的文章和著作，在此向他们致以谢意，并恕不一一列举、标明。

由于作者学术水平有限，错误和不足在所难免，敬请各位读者批评指正，作者十分感谢。

作者

2007年12月于长春

目 录

第1章 绪论 1

1.1 “鸡兔同笼”——计算 1

1.2 算法——程序设计精髓 2

1.3 表示算法——PAD图 4

1.4 程序 4

1.5 执行程序 6

1.6 C语言 7

1.7 Visual C++集成开发环境 9

1.7.1 启动 Visual C++ 9

1.7.2 在独立文件模式下建立
环境 10

1.7.3 在独立文件模式下录入、
编辑源程序 11

1.7.4 编译 12

1.7.5 连接 13

1.7.6 运行 13

1.7.7 调试 13

1.7.8 关闭运行环境 15

1.7.9 设定程序参数 17

1.7.10 在项目管理模式下建立
运行环境 19

1.7.11 在项目管理模式下录入、
编辑源程序 21

本章小结 23

习题一 23

第2章 顺序程序设计 24

2.1 求绿化带宽度——简单程序 24

2.2 基本符号 25

2.2.1 字符集 25

2.2.2 特定符 26

2.2.3 标识符 26

2.2.4 间隔符 27

2.2.5 注释 28

2.3 数据 29

2.3.1 字面常量 29

2.3.2 常量标识符 31

2.3.3 变量 32

2.4 语句 33

2.5 表达式 34

2.5.1 表达式语句 36

2.5.2 赋值 37

2.6 顺序控制结构 37

2.7 数据类型 38

2.7.1 浮点类型 38

2.7.2 整数类型 39

2.7.3 字符类型 40

2.7.4 混合运算 40

2.7.5 类型转换 40

2.8 输入/输出 42

2.8.1 字符输入 43

2.8.2 字符输出 43

2.8.3 格式输入 44

2.8.4 格式输出 45

本章小结 48

习题二 48

第3章 分支程序设计 50

3.1 判断成绩是否及格——双分支

程序设计	50	6.1.1 数组声明	127
3.2 成绩加上获奖信息——单分支 程序设计	52	6.1.2 下标表达式	127
3.3 逻辑判断——布尔类型	59	6.1.3 数组操作	128
3.3.1 关系运算	59	6.2 统计多科成绩——多维数组	129
3.3.2 布尔运算	60	6.3 程序设计实例	131
3.4 获奖分等级——多分支程序设计	61	6.4 线性表——分类与检索	137
3.5 表示汽车种类——枚举类型	66	6.4.1 分类	137
本章小结	68	6.4.2 检索	143
习题三	68	6.5 带学号的成绩排序——数组 初值	145
第4章 循环程序设计	71	6.6 表示姓名——字符串	148
4.1 计算平均成绩——循环程序	71	6.7 类型定义	156
4.1.1 后判断条件的循环	72	6.8 线性表——栈和队列	157
4.1.2 先判断条件的循环	77	6.8.1 队列	158
4.1.3 for 语句	82	6.8.2 栈	161
4.2 计算全班每人平均成绩——多 重循环	85	*6.9 程序设计实例(二)	162
4.3 程序设计实例	91	本章小结	174
本章小结	101	习题六	174
习题四	101	第7章 指针	178
第5章 模块化程序设计——函数	105	7.1 指针与变量	178
5.1 求给定三角形重心——模块化 程序设计	105	7.1.1 指针类型和指针变量	179
5.2 函数	109	7.1.2 指针所指变量	180
5.2.1 函数定义	111	7.1.3 空指针、无效指针	182
5.2.2 函数调用	112	7.2 指针操作	184
5.2.3 先调用后定义——函数 原型	114	7.3 指向指针的指针	184
5.3 程序设计实例	116	7.4 指针与数组	185
本章小结	122	7.4.1 用指针标识数组	186
习题五	122	7.4.2 指针运算	187
第6章 批量数据组织——数组	125	7.4.3 应注意的问题	189
6.1 成绩统计——数组类型	125	7.4.4 多维数组与指针	192
		7.4.5 指针数组	194
		7.5 指针与字符串	197
		本章小结	203
		习题七	204

第 8 章 表单数据组织——结构体	206	10.2 程序设计实例	244
8.1 成绩单管理	206	* 10.3 计算算术表达式的值——间 接递归	252
8.2 保存成绩单——结构体	208	* 10.4 递归程序执行过程	257
8.2.1 结构体类型	208	本章小结	270
8.2.2 结构体类型名	211	习题十	270
8.2.3 结构体变量	213	第 11 章 外部数据组织——文件	273
8.2.4 指向结构体变量的指针	213	11.1 重新考虑户籍管理问题—— 文件	273
8.2.5 访问结构体变量的成分	214	11.2 文件概述	274
8.2.6 对结构体变量的操作	215	11.3 打开、关闭文件	277
8.3 程序设计实例	216	11.4 I/O 操作	279
本章小结	221	11.4.1 字符读写	279
习题八	221	11.4.2 字符串读写	281
第 9 章 再论函数	223	11.4.3 格式化读写	282
9.1 参数	223	11.4.4 数据块读写	282
9.1.1 C 参数传递规则	223	11.4.5 文件定位	285
9.1.2 传递直线方程系数—— 指针作参数	225	11.5 与操作系统文件联系——程序 参数	288
9.1.3 对任意数组排序——数 组作参数	228	11.6 程序设计实例	292
9.1.4 输出日期——结构体作 参数	231	本章小结	295
9.2 函数值	232	习题十一	295
9.2.1 打印月份名——返回指针 值的函数	232	第 12 章 程序开发	297
9.2.2 读入一张身份证信息—— 返回结构体值的函数	234	12.1 求玉米单产——自顶向下、逐 步求精	297
9.3 作用域	235	12.2 结构化程序设计原则	301
9.3.1 作用域	235	12.3 程序风格	302
9.3.2 局部量和全局量	236	12.3.1 行文格式	302
本章小结	239	12.3.2 标识符	303
习题九	239	12.3.3 注释	304
第 10 章 递归程序设计	242	12.3.4 关于程序的“说明和 定义”	304
10.1 计算 $n!$ ——递归程序设计	242	* 12.4 程序设计实例	304
		* 12.5 八皇后——穷举与试探	312

本章小结	320	14.2.4 位运算	375
习题十二	320	14.2.5 左值	377
第 13 章 动态数据组织	323	14.3 语句	378
13.1 保存实数序列——动态数据 结构	323	14.3.1 break	378
13.2 动态变量	324	14.3.2 continue	379
13.3 链表	326	14.3.3 for 的延伸	380
13.3.1 单向链表	327	14.3.4 else 两义性问题	380
13.3.2 栈	333	14.3.5 goto 和标号	381
13.3.3 队列	334	14.4 数据组织	382
13.4 程序设计实例	336	14.4.1 行指针	382
* 13.5 树	344	14.4.2 职工登记卡——共用体	387
13.5.1 树	344	14.4.3 位段	390
13.5.2 树的操作	345	14.5 存储类别	391
* 13.6 图	355	14.5.1 数据在内存中的存储	391
本章小结	358	14.5.2 生存期	392
习题十三	358	14.5.3 局部变量的存储类别	393
* 第 14 章 若干深入问题	361	14.5.4 全局变量的存储类别	396
14.1 函数	361	14.5.5 函数的存储类别	398
14.1.1 不定方向的数组排序—— 函数指针	361	14.6 编译预处理	399
14.1.2 计算定积分——函数作 参数	363	14.6.1 宏	399
14.1.3 指针形参做指针参数 实参	367	14.6.2 文件包含	402
14.1.4 实在参数计算顺序	369	14.6.3 条件编译	402
14.1.5 函数副作用	371	本章小结	405
14.2 运算	373	习题十四	406
14.2.1 赋值运算	373	附录一 ACSII 字符集	408
14.2.2 顺序表达式	374	附录二 标准函数库头文件表	411
14.2.3 条件表达式	374	附录三 常用函数库中所含常用函数	412
		参考文献	414

第1章 绪论

现代计算机从出现至今不过六、七十年时间，但其发展速度是任何一种新技术都不可比拟的，目前计算机已经渗透到各个领域甚至家庭。本书将以 C 语言为载体向大家揭示如何编制计算机程序，即如何使用计算机解决科技、生产、事务处理等方面的问题，介绍程序设计的基本方法、技术和技巧。

1.1 “鸡兔同笼”——计算

【例 1.1】我国古代数学著作《孙子算经》提出“鸡兔同笼”问题如下：“今有鸡兔同笼，上有三十五头，下有九十四足，问鸡兔各几何？”

解：怎样解决这个问题？大家都会解。条理化地分析如下：

第一步：解决该问题，应该首先把问题数学化。根据题目条件，设有 x 只鸡， y 只兔，可以列出二元一次方程组：

$$x + y = 35 \quad (1)$$

$$2x + 4y = 94 \quad (2)$$

这是一个典型的

$$ax + by = e$$

$$cx + dy = f$$

型的二元一次方程组。

第二步，解该方程组。解二元一次方程组的方法很多，这里选择消元法。

第三步，使用消元法解上述二元一次方程组，按如下步骤进行。

(1) 首先消去未知数 y 。

① 把方程 (1) 乘以 4，计算 1×4 、 35×4 ，得到方程 (3)：

$$4x + 4y = 140 \quad (3)$$

② 把方程 (2) 减去方程 (3)：计算 $2 - 4$ 、 $4 - 4$ 、 $94 - 140$ ，得到方程 (4)：

$$-2x = -46 \quad (4)$$

这是一个典型的

$$ax = b$$

型的一元一次方程。

(2) 解一元一次方程 (4)：计算 $(-46) \div (-2) \rightarrow x$ ，得到未知数 x 的值为 23。

(3) 把 x 值代入方程 (1) 得一元一次方程 (5)：

$$23 + y = 35 \quad (5)$$

(4) 解一元一次方程 (5): 计算 $35 - 23 \rightarrow y$, 得到未知数 y 的值为 12。

从而计算出有 23 只鸡, 12 只兔。

现在已经把解决该问题的各个步骤完成。分析上述步骤, 计算过程如下:

1. $1 \times 4 \rightarrow a$ // 方程(1)乘以 4, x 的系数保存在 a 中
2. $35 \times 4 \rightarrow e$ // 常数项保存在 e 中
3. $2 - a \rightarrow a$ // 方程(2)减去方程(3), 得一元一次方程(4)
// x 的系数保存在 a 中
4. $94 - e \rightarrow b$ // 常数项保存在 b 中
5. $b/a \rightarrow x$ // 解 $ax = b$ 型一元一次方程 $-2x = -46$, 结果送入 x 中
6. $35 - x \rightarrow y$ // 解一元一次方程 $23 + y = 35$, 结果送入 y 中
7. 显示 x, y

这就是“程序”。程序就是一个计算过程, 由若干计算步骤组成。选择一种程序设计语言, 把上述计算过程用该程序设计语言表示出来就是计算机程序。上述求解问题的过程就是“程序设计”。

这是一个简单的实际的计算问题, 也是一个简单的程序设计过程。一般的, 一个实际的问题要使用计算机来解决, 大致需要经过如下步骤:

- (1) 建立数学模型——把实际问题转化为数学问题;
- (2) 找出计算方法——为数学问题的求解找出方法;
- (3) 进行算法分析——为实现计算给出具体算法;
- (4) 选择一种程序设计语言, 编写出计算机程序——写程序;
- (5) 调试程序——保证程序的正确性;
- (6) 在计算机上运行和测试程序——组装测试、确认测试;
- (7) 交付使用并维护。

在例 1.1 的简单问题求解过程中经过的步骤如下:

第一步是建立数学模型, 列出了二元一次方程组。

第二步是找出计算方法, 选择“消元法”解二元一次方程组。

第三步是算法分析, 得到解二元一次方程组的具体计算步骤。

写程序、调试程序、运行测试程序、维护程序几步后面会逐步介绍。

1.2 算法——程序设计精髓

1.1 节中最后给出的计算步骤 1~7 是解决例 1.1 问题的算法。程序设计的任务就是找出算法 (算法分析)、编出计算机程序并调试、运行和测试程序。给出算法, 本课程要解决一部分, 将来还要学习“算法分析”课, 但由于现实世界是五花八门、十分复杂的, 要真正解决

算法问题还要进行长期的学习、积累、实践和悟性。

算法由某些基本成分组成，这些基本成分是一些基本的操作及控制结构。构成算法的基本成分包括：

1. 表达式以及变量赋值
2. 读（输入）
3. 写（输出）。

基本的控制结构包括：

1. 顺序控制结构
2. 分支控制结构
3. 循环控制结构
4. 函数调用
5. 函数返回。

算法是一个计算过程，指明应该进行的具体操作，描述解决问题的方法和途径，它是程序设计的基础和精髓。一个有效的算法应该具有如下特点：

1. 有穷性

一般情况下，一个算法应该在有限的时间内终止，不应该是无限的。进一步，有穷性（Finiteness）往往指在合理的时间范围内，如后面讲述的“Hanoi 塔”问题的算法，若 1 秒钟计算一次，大约需要 5 849 亿年。虽然是有穷的，但是显然是无意义的。

2. 确定性

算法中的每一个步骤都应该是确定（Definiteness）的，含义是唯一的，不应该是模糊的、模棱两可的。例如，求解一元二次方程，如果直接将算法写成

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

显然是不确定的，不能构成确定的算法步骤。例如，当 a 等于 0 时怎么办？当 $b^2 - 4ac$ 小于 0 时怎么操作？都没有确定的描述。

3. 有效性

算法中的每一个步骤都应该是有效（Effectiveness）的，都能够被有效执行并得到确定的结果，不应该存在无效的操作。例如，在一个算法中存在操作“ $x \rightarrow x$ ”，显然无意义，也是无效的；又如，操作“ $a \div 0$ ”也是不能有效执行的。

4. 若干输入

算法应有 0 个或多个输入（Input），以给定运算的初始条件。若算法有 0 个输入，则应该在算法内部设定初始条件（如例 1.1）。

5. 若干输出

算法应有一个或多个输出（Output），以给出对输入数据进行处理后的结果，如例 1.1，最终输出鸡、兔的只数。没有输出的算法没有任何意义。

1.3 表示算法——PAD图

描述算法有多种方法。例如，流程图、NS图、程序……本书采用所谓的问题分析图(Problem Analysis Diagram, PAD)来描述算法。问题分析图简称PAD图，它有许多优点。

PAD使用两维的树状结构描述程序的逻辑，因此它比直接用程序表示算法更清晰直观(可以说程序的表现形式是一维的)；PAD使用了结构化的、概括的、抽象的记号系统，所以它比用流程图表示算法更清晰、简练、紧凑、层次分明(流程图不是结构化的)；PAD是开放的，所以它比NS图更清晰、分明，也更便于修改(NS图是封闭式的)。

为简单明了起见，也为了适应C的一些特点，本书把标准PAD记号系统做了一定的修正。目前先给出PAD记号系统的基本格式，再给出例1.1算法的PAD描述。更复杂结构的PAD表示将在下文涉及的地方逐步给出。

(1) PAD把基本操作用方框括起来，表示成图1.1的形式。

(2) PAD把顺序执行的操作序列用一条竖线顺序连接起来表示成图1.2的形式。竖线从上向下表示程序执行的顺序，竖线连接起来的是一个操作成分。

【例1.2】例1.1问题算法(计算步骤1~7)的PAD描述(图1.3)。

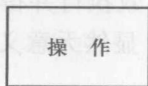


图 1.1 基本操作序列

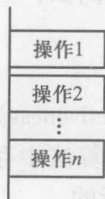


图 1.2 顺序控制结构

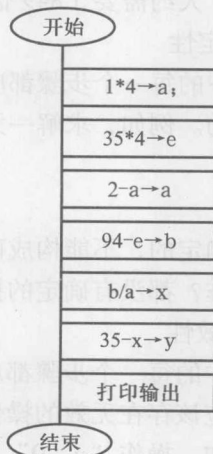


图 1.3 例 1.1 算法的 PAD 描述

1.4 程序

程序(Program)是一个指令序列，也就是用指令排成的一个工作顺序、工作步骤。人们

平常也使用程序这个名词,如运动会程序、文艺演出节目程序等。计算机程序是用计算机指令为计算机排定的工作顺序、工作步骤。为计算机编排程序的过程称为程序设计。

描述程序必须使用一种语言。程序设计语言是指用于编写、描述计算机程序的语言。C语言是使用最广泛的程序设计语言,本书以C语言为载体讲述程序设计。

对于例1.1的计算过程,按图1.3的PAD写出C程序,如例1.3所示。执行该程序将输出:

23 12

【例1.3】例1.1的C程序。

```
#include <stdio.h>           // 插入标准输入输出函数库头文件
int a,b,e,x,y;              // 声明5个变量,分别表示计算用的中间结果和最后结果
main() {                    // 主函数
    a = 1 * 4;               // 方程(1)乘以4,得方程(3)
    e = 35 * 4;
    a = 2 - a;               // 方程(2)减方程(3)
    b = 94 - e;
    x = b/a;                 // 解 ax = b 型一元一次方程 -2x = -46,求 x
    y = 35 - x;              // 解一元一次方程 23 + y = 35,求 y
    printf("%d %d\n",x,y);  // 打印输出
}
```

这是一个完整的C程序,该程序全部在一个文件中。在该程序中,从第3行的“main()”到最后是主函数,称为一个顶层声明。在该行之前,程序的第2行为变量声明,也是一个顶层声明。程序的第1行括入一个文件。在本程序中,主函数完成了全部的计算功能。

在实际应用中,若程序较大,一个程序一般由多个函数组成。往往把一个程序分成若干片段,分别存放在不同的文件中,每个文件称为一个“编译单元”。图1.4是由两个编译单元组成的一个完整的C程序,每个编译单元保存在一个源程序文件中。执行该程序将输出如下字符串:

Hello!

文件hello.c的内容:

```
#include<stdio.h>
int m;
hello(){
    printf("Hello!\n");
}
```

文件startup.c的内容:

```
extern hello();
int main(){
    hello();
    return 0;
}
```

图 1.4 一个完整的 C 程序