

天津滨海迅腾科技集团有限公司

《企业级卓越人才培养解决方案》指定用书

C 语言 程序设计

王新强 主编

南开大学出版社

C 语言程序设计

王新强 主编

南开大学出版社
天 津

图书在版编目(CIP)数据

C 语言程序设计 / 王新强主编. —天津:南开大学出版社,2016.7

ISBN 978-7-310-05129-8

I. ①C… II. ①王… III. ①C 语言—程序设计 IV.
①TP312

中国版本图书馆 CIP 数据核字(2016)第 125161 号

版权所有 侵权必究

南开大学出版社出版发行

出版人:孙克强

地址:天津市南开区卫津路 94 号 邮政编码:300071

营销部电话:(022)23508339 23500755

营销部传真:(022)23508542 邮购部电话:(022)23502200

*

天津午阳印刷有限公司印刷

全国各地新华书店经销

*

2016 年 7 月第 1 版 2016 年 7 月第 1 次印刷

260×185 毫米 16 开本 21 印张 530 千字

定价:58.00 元

如遇图书印装质量问题,请与本社营销部联系调换,电话:(022)23507125

企业级卓越互联网应用型人才培养解决方案

一、企业概况

天津滨海迅腾科技集团是以 IT 产业为主导的高科技企业集团，总部设立在北方经济中心——天津，子公司和分支机构遍布全国近 20 个省市，集团旗下的迅腾国际、迅腾科技、迅腾网络、迅腾生物、迅腾日化分属于 IT 教育、软件研发、互联网服务、生物科技、快速消费品五大产业模块，形成了以科技为源动力的现代科技服务产业链。集团先后荣获“全国双爱双评先进单位”“天津市五一劳动奖状”“天津市政府授予 AAA 级和谐企业”“天津市文明单位”“高新技术企业”“骨干科技企业”等近百项殊荣。集团多年中自主研发天津市科技成果 2 项，自主研发计算机类专业教材 36 种，具备自主知识产权的开发项目包括“进销存管理系统”“中小企业信息化平台”“公检法信息化平台”“CRM 营销管理系统”“OA 办公系统”“酒店管理系统”等数十余项。2008 年起成为国家工业和信息化部人才交流中心“全国信息化工程师”项目联合认证单位。

二、项目概况

迅腾科技集团“企业级卓越互联网应用型人才培养解决方案”是针对我国高等职业教育量身定制的应用型人才培养解决方案，由迅腾科技集团历经十余年研究与实践研发的科研成果，该解决方案集三十余本互联网应用技术教材、人才培养方案、课程标准、企业项目案例、考评体系、认证体系、教学管理体系、就业管理体系等于一体。采用校企融合、产学研融合、师资融合的模式在高校内建立校企共建互联网学院、软件学院、工程师培养基地的方式，开展“卓越工程师培养计划”，开设互联网应用技术领域系列“卓越工程师班”，“将企业人才需求标准引进课堂，将企业工作流程引进课堂，将企业研发项目引进课堂，将企业考评体系引进课堂，将企业一线工程师请进课堂，将企业管理体系引进课堂，将企业岗位化训练项目引进课堂，将准职业人培养体系引进课堂”，实现互联网应用型卓越人才培养目标，旨在提升高校人才培养水平，充分发挥校企双方特长，致力于互联网行业应用型人才培养。迅腾科技集团“企业级卓越互联网应用型人才培养解决方案”已在全国近二十所高校开始实施，目前已形成企业、高校、学生三方共赢格局。未来五年将努力实现在 100 所高校实施“每年培养 5~10 万互联网应用技术型人才”发展目标，为互联网行业发展做好人才支撑。

前 言

首先感谢您选择了企业级卓越互联网应用型人才培养解决方案，选择了本教材。本教材是企业级卓越互联网应用型人才培养解决方案的载体之一，面向行业应用与产业发展需求，系统传授软件开发全过程的理论和技能，并注重 IT 管理知识的传授和案例教学。

C 语言程序设计在企业级卓越互联网应用型人才培养解决方案软件开发工程师课程体系中起了重要的作用。软件开发的各个环节是相辅相成的，编写程序是最基础的东西，只有脚踏实地地掌握好这个基础才有办法往上走，真正进入 IT 行业。编程有很多种方法，如结构化编程（面向过程编程）、基于对象编程（ADT）、面向对象编程、泛型编程等，作为初学者到底应该从哪儿入手呢？最佳的选择就是 C 语言程序设计（基于过程程序设计）。

C 语言程序设计在软件开发中是怎么做的呢？简单地说，它注重的是算法设计（过程设计），是进行以模块功能和处理过程设计为主的详细设计的基本原则。它的主要观点是采用自顶向下、逐步求精的程序设计方法。这种方法使程序层次分明、结构清晰，有效地改善了程序的可靠性，提高了程序设计的质量和效率。更多相关结构化程序设计的特点随着学习的深入您会有更多的了解。

本书对结构化程序设计的介绍是以 C 语言贯穿始终，难易程度适中，重点介绍面向过程编程中的三种基本结构，即顺序结构、选择结构、循环结构，以及数组和 C 语言中独有的数据类型——指针。在解决编程问题时总会用到一个工具——抽象，不管是哪种编程方法都利用了“抽象”这个工具。结构化程序设计中的抽象就是函数，所以函数也是本书的重点之一。全书集合了 C 语言的经典案例，能够引导学员快速地建立编程思想，掌握编程的学习方法。该解决方案强调掌握学习的方法和创造新的事务处理规则，能够触类旁通、举一反三。本书内容的设计完全符合这一原则，例如解决同一个例题，教学案例中给出了多个解决方案的示例，能够使读者融会贯通。

在信息化的潮流中，提高程序编写的技能是我们现在所涉及的 IT 行业的必修课。之所以选择 C 语言作为入门的程序开发语言是因为 C 语言正迅速成为一种最重要、也是最流行的程序设计语言。对它的使用量一直在增长也是因为初学者一接触它，就喜欢上它。当你学习 C 语言时，你也会认识到它有许多优点。

在实际解答学员相关疑问时所得到的反馈，我们已将其融入书中，希望阅读本教材的初学者在学习 C 语言程序设计时能少走弯路，得到更正确的理论知识，为后期的学习奠定坚实的基础。

天津滨海迅腾科技集团有限公司课程研发部

2016 年 5 月

目 录

理论部分

第 1 章 程序和流程图	3
1.1 本章简介	3
1.2 程序的灵魂——算法	4
1.3 结构化编程	10
1.4 C 语言背景	12
1.5 简单 C 语言程序介绍	13
1.6 基本语法成分	15
1.7 Visual Studio 2012 简介	17
1.8 小结	21
1.9 英语角	21
1.10 作业	22
1.11 思考题	22
1.12 学员回顾内容	22
第 2 章 数据类型及输入输出函数	23
2.1 本章简介	23
2.2 变量与常量	23
2.3 C 语言的数据类型	27
2.4 输入输出函数	36
2.5 小结	43
2.6 英语角	43
2.7 作业	43
2.8 学员回顾内容	44
第 3 章 运算符和表达式	45
3.1 本章简介	45
3.2 算术运算符	47
3.3 关系运算符	51
3.4 逻辑运算符	53
3.5 其他运算符	55

3.6	表达式	56
3.7	数据类型转换	57
3.8	程序举例	59
3.9	小结	61
3.10	英语角	61
3.11	作业	62
3.12	学员回顾内容	62
第 4 章	分支结构	63
4.1	本章简介	63
4.2	顺序结构	63
4.3	if 语句	65
4.4	小结	81
4.5	英语角	82
4.6	作业	82
4.7	思考题	83
4.8	学员回顾内容	83
第 5 章	循环结构	84
5.1	本章简介	84
5.2	while 循环语句	84
5.3	do...while 循环语句	87
5.4	for 循环语句	90
5.5	goto 语句以及用 goto 语句构成循环	93
5.6	几种循环的比较	94
5.7	程序举例	94
5.8	小结	97
5.9	英语角	97
5.10	作业	98
5.11	思考题	98
5.12	学员回顾内容	98
第 6 章	循环跳出和循环嵌套	99
6.1	本章简介	99
6.2	break 语句和 continue 语句	99
6.3	循环嵌套	102
6.4	程序举例	105
6.5	小结	109
6.6	英语角	109

6.7	作业	109
6.8	思考题	110
6.9	学员回顾内容	111
第 7 章	数组的简单介绍	112
7.1	本章简介	112
7.2	一维数组	112
7.3	二维数组	118
7.4	数组的基本操作	120
7.5	小结	122
7.6	英语角	122
7.7	作业	123
7.8	学员回顾内容	123
第 8 章	数组和循环嵌套	124
8.1	本章简介	124
8.2	一维数组程序举例	124
8.3	二维数组程序实例	135
8.4	小结	137
8.5	英语角	137
8.6	作业	137
8.7	思考题	138
8.8	学员回顾内容	139
第 9 章	函数	140
9.1	本章简介	140
9.2	函数的概念	140
9.3	函数的工作原理	143
9.4	函数的一般形式	146
9.5	函数调用	152
9.6	规划结构化程序	152
9.7	库函数的调用	154
9.8	小结	156
9.9	英语角	156
9.10	作业	156
9.11	思考题	157
9.12	回顾内容	157

第 10 章 字符串	158
10.1 本章简介	158
10.2 字符数组	158
10.3 字符串处理函数	161
10.4 二维字符数组的简单应用	165
10.5 字符数组举例	168
10.6 小结	170
10.7 英语角	170
10.8 作业	170
10.9 思考题	170
10.10 学员回顾内容	170
第 11 章 内存管理	171
11.1 本章简介	171
11.2 内存地址	171
11.3 指针	174
11.4 小结	189
11.5 英语角	189
11.6 作业	189
11.7 思考题	189
11.8 学员回顾内容	190
第 12 章 预处理命令	191
12.1 本章简介	191
12.2 宏定义	192
12.3 “文件包含”处理	196
12.4 条件编译	198
12.5 局部变量和全局变量	201
12.6 变量的存储类别	205
12.7 小结	209
12.8 英语角	209
12.9 作业	209
12.10 思考题	210
第 13 章 结构体与枚举类型	211
13.1 本章简介	211
13.2 定义一个结构的一般形式	211
13.3 结构类型变量的说明	212
13.4 结构变量成员的表示方法	214

13.5	结构变量的赋值	215
13.6	结构变量的初始化	215
13.7	结构数组的定义	216
13.8	结构指针变量的说明和使用	218
13.9	枚举类型	221
13.10	小结	224
13.11	英语角	224
13.12	作业	224
第 14 章	综合应用	225
14.1	程序功能要求	225
14.2	问题分析	228
14.3	程序完整代码	235
14.4	小结	241
14.5	英语角	241
14.6	作业	241
14.7	思考题	242

上机部分

第 1 章	程序和流程图	245
1.1	指导	245
1.2	作业	247
第 2 章	数据类型及输入输出函数	248
2.1	指导	248
2.2	作业	250
第 3 章	运算符和表达式	251
3.1	指导	251
3.2	作业	255
第 4 章	分支结构	256
4.1	指导	256
4.2	作业	260
第 5 章	循环结构	261
5.1	指导	261
5.2	作业	264

第 6 章 循环跳出和循环嵌套.....	265
6.1 指导.....	265
6.2 作业.....	268
第 7 章 数组的简单介绍.....	269
7.1 指导.....	269
7.2 作业.....	273
第 8 章 数组和循环嵌套.....	274
8.1 指导.....	274
8.2 作业.....	277
第 9 章 函数.....	278
9.1 指导.....	278
9.2 练习（50 分钟）	285
9.3 作业.....	286
第 10 章 字符串.....	287
10.1 指导.....	287
10.2 练习（50 分钟）	294
10.3 作业.....	294
第 11 章 内存管理.....	295
11.1 指导.....	295
11.2 练习（50 分钟）	300
11.3 作业.....	300
第 12 章 预处理命令.....	301
12.1 指导.....	301
12.2 练习（50 分钟）	307
12.3 作业.....	307
第 13 章 结构体与枚举型.....	308
13.1 指导.....	308
13.2 练习（50 分钟）	315
13.3 作业.....	316
第 14 章 综合运用.....	317
14.1 指导.....	317
14.2 练习（50 分钟）	317

理论部分

第 1 章 程序和流程图

学习目标

- ◇ 了解程序、算法和流程图的概念；
- ◇ 理解问题和处理问题的方式；
- ◇ 掌握 C 程序的基本构造；
- ◇ 掌握 C 程序的编译和运行过程；
- ◇ 掌握使用 Visual Studio 2012 创建 C 程序的步骤。

课前准备

在进入本章的学习前，你应该首先对计算机的基础知识有一定了解，知道如何启动应用程序。

1.1 本章简介

著名计算机科学家 Niklaus Wirth（沃斯）曾提出：程序是由数据结构和算法所构成的集合体。其中，数据结构（data structure）是指程序中的特定数据类型和数据组织形式，也就是需要我们加工的内容。而算法（algorithm）则是指为达到某个目的所要执行的操作步骤，是处理问题域中问题的解决方式。

计算机算法可分为两大类，数值运算算法和非数值运算算法。数值运算的目的是求数值解，例如求方程的根，求一个函数的定积分等，都属于数值运算范围。非数值运算包括的面十分广泛，最常见的是用于事务管理领域，例如图书检索、人事管理、行车调度管理等。目前，计算机在非数值运算方面的应用远远超过了在数值运算方面的应用。由于数值运算有现成的模型，可以运用数值分析方法，因此对数值运算的算法研究比较深入，算法比较成熟。对各种数值运算都有比较成熟的算法可供选用。人们常常把这些算法汇集成册（写成程序形式），或者将这些程序存放在磁盘或磁带上，供用户调用。例如有的计算机系统提供“数学程序库”使用起来十分方便。而非数值运算的种类繁多，要求各异，难以规范化，因此只对一些典型的非数值运算算法（例如排序算法）作比较深入的研究。其他的非数值运算问题，往往需要使用者参考已有的类似算法重新设计解决特定问题的专门算法。

本书不罗列所有算法，只是通过一些典型算法的介绍，帮助读者了解如何设计一个算法，推动读者举一反三。希望读者通过本章介绍的例子了解怎样提出问题，怎样思考问题，怎样

表示一个算法。

1.2 程序的灵魂——算法

在生活中我们经常会碰到一些需要解决的疑难，而这些疑难或矛盾就称为问题。为了正确的处理这些问题，首先必须理解所要解决的问题是什么，然后制定出一套相应的处理步骤来解决问题。做任何事情都有一定的步骤。例如，你想从天津去陕西西安开会，首先要去买火车票，然后按时乘车到天津站，登上火车，到西安站后乘电车到会场，参加会议。你要买家电，先要选好货物，然后开票、付款、拿发票、取货，乘车回家。要考大学，首先要填报名单、交报名费，拿到准考证，按时参加考试，得到录取通知书，到指定学校报到注册等。这些步骤都是按一定的顺序进行的，缺一不可，顺序错了也不行。从事各种工作和活动，都必须事先想好进行的步骤，然后按部就班地进行才能避免产生错乱。在实际日常生活中，由于已养成习惯，所以人们并没有意识到每件事都需要事先设计出“行动步骤”。例如吃饭、上学、打球、做作业等，事实上都是按照一定的规律进行的，只是人们不必每次都重复考虑它而已。

“算法”并不仅仅是“计算”的问题，广义地说，为解决一个问题而采取的方法和步骤，就称为“算法”。本书所涉及的只限于计算机算法，即计算机能执行的算法。例如，让计算机算 $1 \times 2 \times 3 \times 4 \times 5$ ，或将 100 个学生的成绩按高低分次序排列，这是可以做到的。

1.2.1 分析问题

我们可能都听到过这样一句话：“把大象放进冰箱里有三个步骤，第一步，把冰箱门打开。第二步，把大象放进去。第三步，把冰箱门关上。”先别忙笑，让我们换一个角度重新来看待这句话。首先，从这句话中得到我们所要解决的问题——“把大象放进冰箱”。为了完成这个任务，我们制定了相关的步骤：

第一步，打开冰箱门。

第二步，把大象放进去。

第三步，把冰箱门关上。

任务圆满完成。而编写程序也是一样，首先必须明确地知道，我们所要解决的问题是什么。例如：需要寄包裹给某个远方的朋友。从这句描述中不难发现，现在所要解决的问题是寄包裹给朋友。明确了这个任务后，我们就可以制定出相应的步骤：

第一步，将需要邮寄的东西放在一起打成一个包裹。

第二步，带着要邮寄的包裹前往附近的邮局。

第三步，在邮局中贴好足够的邮资。

第四步，邮寄包裹。

以上列出的步骤明确的定义了一系列可执行的步骤，只需要执行上面的步骤，就可以轻松解决“寄包裹”这个问题域中所要解决的问题，这些步骤都是按一定顺序进行的，遗漏或次序颠倒都可能会产生错误，这一系列的有序步骤就称为算法。

计算机解题的过程，就是模拟现实生活中对问题的解决方法来处理计算机内的问题。在

这个过程中，无论是形成解题思路还是编写程序代码，都是在实施着某些算法。

一个算法应该具有五个重要的特征：

- 有穷性：一个算法必须保证能在合理的范围内，以有限的步骤得到结果。
- 确定性：算法中的每一个步骤都必须是明确的，不能具有二义性。
- 有零个或多个输入：所谓输入是指算法在实施过程中，从外界获取必要的信息。

例如：求两数和的应用程序，就是需要从用户那获得两个输入的数据用于参加运算。而有时算法也可以不需要任何输入数据，例如：一个用于求解 $5+3$ 的加法程序，就不需要从外界获取任何信息，而直接给出运算结果 8。

- 有一个或者多个输出：算法的目的是为了求解，这里的“解”就是输出。
- 有效性：算法应该在有限步骤里得到确定的结果。

对同一个问题，可以有不同的解题方法和步骤。例如，求 $1+2+3+\cdots+100$ ，有人可能先进行 $1+2$ ，再加 3，再加 4，一直加到 100，而有的人采取这样的方法： $100+(1+99)+(2+98)+\cdots+(49+51)+50=100+49\times 100+50=5050$ 。还可以有其他的方法。当然，方法有优劣之分。有的方法只需进行很少的步骤，而有些方法则需要较多的步骤。一般我们采用运算简单，步骤少的方法。因此，为了有效地进行解题，不仅需要保证算法正确，还要考虑算法的质量，选择合适的算法。程序员要根据现实的问题来设计出相应的算法，由于处理的方法和描述的内容不同，每个设计人员所设计的过程也会有所不同。实际编写应用程序前，程序员往往采用某种语言对实际要执行的步骤进行编码描述，以便确定特定的执行步骤。

1.2.2 简单算法举例

例 1-1 求 $1+2+3+4+5$ 。

步骤 1：先求 $1+2$ ，得到结果 3。

步骤 2：将步骤 1 得到的和 3 再加 3，得到结果 6。

步骤 3：将 6 与 4 相加，得 10。

步骤 4：将 10 与 5 相加，得 15。这就是最后的结果。

这样的算法虽然是正确的，但太繁琐。如果要求 $1+2+3+\cdots+1000$ ，则要写 999 个步骤，显然是不可取的。而且每次都直接使用上一个步骤的数值结果，也不方便。应当找到一种通用的表示方法。

可以设两个变量，一个变量代表被加数，一个变量代表加数。不再另设变量存放相加的结果，而直接将每一步骤的和放在被加数变量中，设 p 为被加数， i 为加数。用循环算法来求结果。可以将算法改写如下：

S1：使 $1 \rightarrow p$ ；

S2：使 $2 \rightarrow i$ ；

S3：使 $p+i$ ，和仍放在变量中，可表示为 $p+i \rightarrow p$ ；

S4：使 i 的值加 1，即 $i+1 \rightarrow i$ ；

S5：如果 i 不大于 5，返回重新执行步骤 S3 以及其后的步骤 S4 和 S5；否则，算法结束。最后得到 p 的值就是 $1\sim 5$ 相加的和。

上面的 S1、S2 等代表步骤 1，步骤 2，……。S 是 Step（步）的缩写。这是写算法的习惯用法。

请读者仔细分析这个算法，能否得到预期的结果。显然这个算法比前面列出的算法简练。如果题目改为求 $1+3+5+7+9+11$ ，算法只需做很少的改动即可：

```
S1: 1 → p;
S2: 3 → i;
S3: p+i → p;
S4: i+2 → i;
S5: 若 i ≤ 11, 返回 S3, 否则结束。
```

可以看出，用这种方法表示的算法具有通用性、灵活性。S3 到 S5 组成一个循环，在实现算法时，要反复多次执行 S3、S4、S5 等步骤，直到某一时刻，执行 S5 步骤时经过判断，加数 i 已超过规定的数值而不返回 S3 步骤为止。此时算法结束，变量 p 的值就是所求结果。

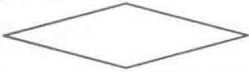
由于计算机是高速进行运算的自动机器，实现循环是轻而易举的，所有计算机高级语言中都有实现循环的语言。因此，上述算法不仅是正确的，而且是计算机能实现的较好的算法。

1.2.3 流程图符号介绍

在 1.2.2 中介绍的算法是用自然语言表示的。自然语言是人们日常使用的语言，可以是汉语、英语或者其他语言。用自然语言表示通俗易懂，但文字冗长，容易出现“歧义性”。自然语言表示的含义往往不大严格，要根据上下文才能判断其正确含义。特别是包含分支和循环的算法，用自然语言描述不很方便。因此，除了很简单的问题以外，一般不用自然语言描述算法。

最常用的描述算法方法就是流程图。流程图是用一些图框表示各种操作。用图形表示算法，直观形象，易于理解。美国国家标准协会 ANSI (American National Standard Institute) 规定了一些用于表示流程图的常用符号，如表 1-1 所示。

表 1-1 流程图常用符号表

流程图符号	含义
	起止框，用于表示流程图的开始和结束标志
	输入输出框，用于接收外部用户的输入或将要显示的内容输出到屏幕中
	判断框，用于表示程序中的某个执行逻辑步骤
	处理框，用于表示程序中的某个执行步骤
	流程线，表示应用程序的执行语句
	连接点，当流程图需要跨图存在时，可以用连接点连接位于两张图纸上的同一个流程
	注释框，用于对流程图中的补充说明