

普通高等教育“十二五”规划教材
广东省教育厅“育苗工程（自然科学）”项目成果

C语言程序设计

——增量式项目驱动一体化教程

C Programming Language

◎ 苑俊英 谭志国 主编 ◎ 陈海山 温泉思 曹惠茹 副主编



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

广东省教育厅“育苗工程（自然科学）”项目成果
普通高等教育“十二五”规划教材

C 语言程序设计

——增量式项目驱动一体化教程

苑俊英 谭志国 主 编

陈海山 温泉思 曹惠茹 副主编

电子工业出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书按照增量式项目驱动一体化的教学模式安排教学内容,融知识点、实践案例于一体,重点讲解如何将各知识点应用于实践。每章都列出了核心知识点,并通过简单、可理解的示例,帮助读者理解和掌握核心知识的应用,增量项目驱动起着训练读者编程能力、知识点的综合应用能力。本书包括 13 章内容以及 5 个附录。

本书在内容上侧重 C 语言基本语法的学习和应用,采用增量的方式贯穿整全书,并介绍了一个 LED 数码管程序的开发过程,适合初学者对 C 语言的理解,并通过对案例的学习和模拟,将 C 语言基本技能实现其他应用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

C 语言程序设计:增量式项目驱动一体化教程 / 苑俊英,谭志国主编. —北京:电子工业出版社,2015.9
ISBN 978-7-121-26344-6

I. ① C… II. ① 苑… ② 谭… 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 (2015) 第 130310 号

策划编辑:章海涛

责任编辑:章海涛

特约编辑:何 雄

印 刷:三河市双峰印刷装订有限公司

装 订:三河市双峰印刷装订有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱邮编 100036

开 本:787×1092 1/16 印张:15.5 字数:396 千字

版 次:2015 年 9 月第 1 版

印 次:2015 年 9 月第 1 次印刷

定 价:36.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

本书是作者多年教学和实践经验的总结,按照**增量式项目驱动一体化的教学模式**安排教学内容,融**知识点、实践案例于一体**,重点讲解如何将各知识点应用于实践。

由于大一新生初次接触程序设计语言,以及C语言上课枯燥的特点,本教材教学团队抛弃传统教材单纯讲解C语言语法的形式,将知识点贯穿于案例中,以案例驱动,循序渐进、由浅入深;采取知识点与案例相对应的方式,安排教学内容;采用增量式的程序设计模式安排教学内容,将任务进行分解、简化问题;最终使读者既能掌握编程语言的思想和方法,又能学有所获。

本书为广东省教育厅“育苗工程(自然科学)”之“**计算思维与应用型人才培养结合下的计算机专业基础课程建设**”项目成果之一。

书中每章都列出了核心知识点,并通过简单、可理解的示例,帮助读者理解和掌握核心知识的应用,增量式项目驱动能够训练读者编程能力、知识的综合应用能力。

第1章主要讲解C语言的基本概念、C语言开发环境和C程序的开发步骤,要求读者初步认识C语言,并通过C语言开发环境开发简单的C程序。

第2章介绍增量项目——LED数码管,并对整个项目进行了增量划分,将每个增量与后续的章节进行知识点与增量任务的对应。

第3章介绍C语言的基本数据类型,要求读者熟练使用C语言数据类型来定义数据。

第4章介绍C语言的运算符和表达式,要求读者熟练应用各种运算符和表达式进行计算和语句表达。

第5章介绍选择结构程序设计,第6章介绍循环结构程序设计,要求读者掌握使用控制结构进行复杂程序设计的能力。

第7章介绍函数的结构、函数的定义和调用等内容,要求读者掌握用函数进行模块化程序设计的思想,并在LED数码管增量上使用函数对功能模块进行了封装。

第8章介绍数组,强调数组在解决实际问题中的重要性及使用。

第9章介绍C语言的重要内容——指针,用简单明了的方式介绍了指针的应用。

第10章介绍字符串的应用。

第11章介绍结构体、共用体和枚举类型。

第12章结合实例介绍文件的操作和使用。

第13章简单介绍C语言中的预编译命令。

本书在内容上侧重C语言基本语法的学习和应用,采用增量的方式贯穿整全书,并介绍了一个LED数码管程序的开发过程,适合初学者对C语言的理解,并通过对案例的学习和模拟,将C语言基本技能实现其他应用。

本书采用**开源软件 CodeBlocks**作为C语言开发环境,CodeBlocks可以从官方网站下载、安装和使用。本书还配有《C语言程序设计实验及习题指导》(ISBN 978-7-121-26345-3),可以作为与本书配套的实验教材。

本书可以作为计算机及相关专业程序设计课程的教学用书,不同专业在讲授时可根据学生、学时等具体情况有选择讲授不同章节的内容,还可以作为计算机等级考试的学习或参考用书。

对于计算机专业的学生，建议授课学时为 54+ 36 学时，其中 54 学时为课堂讲授，36 学时为上机实验。各章节学时分配可参考如下。

课程内容	学时（理论+实验）
第 1 章 初始 C 语言	2+1
第 2 章 C 语言知识在实践中的应用	1+1
第 3 章 基本数据类型	3+2
第 4 章 运算符与表达式	6+4
第 5 章 选择结构程序设计	6+4
第 6 章 循环结构程序设计	6+4
第 7 章 函数调用	6+4
第 8 章 数组	6+4
第 9 章 指针	6+4
第 10 章 字符串处理	3+2
第 11 章 结构体、共用体和枚举	4+2
第 12 章 读写文件	3+2
第 13 章 预编译命令	2+2

本书第 1~3 章由陈海山编写，第 4~7 章由苑俊英编写，第 8~9 章由温泉思编写，第 10~12 章由谭志国编写，第 13 章由曹惠茹编写。书中 LED 数码管案例的增量实现由陈海山、李瑞程完成，全书由苑俊英负责统稿和定稿。

本书在编写过程中得到了中山大学信息科学与技术学院杨智教授、中山大学南方学院洪维恩教授的支持与帮助，在此表示诚挚的谢意。在本书编写过程中，中山大学南方学院的李瑞程、余聪、白凯凯、李浩铭等同学参与了本书代码的测试工作。同时感谢电子工业出版社及所有编辑为本书完成所做的工作。

本书还配有教学课件、实例代码、增量项目源码和实验，有需要的读者可发邮件至 cihisa@126.com，也可以登录到 <http://www.hxedu.com.cn> 进行下载。

由于作者水平有限，编写时间仓促，在本书中难免有一些错误，恳请读者提出宝贵建议。

作 者

目 录

第 1 章 初识 C 语言	1
1.1 C 语言概述	1
1.2 C 语言开发环境	2
1.2.1 运行 C 语言程序的步骤和方法	2
1.2.2 最简单的 C 语言程序	3
1.3 算法	4
1.3.1 算法的定义	4
1.3.2 算法的表示	4
1.3.3 算法举例	7
本章小结	10
习题 1	10
第 2 章 C 语言知识在实践中的应用	11
2.1 案例介绍	11
2.2 案例分析	12
2.2.1 显示单个数字	12
2.2.2 依次显示数字	14
2.2.3 无限次或有限次循环显示数字 0~9	15
2.2.4 循环显示任意一位指定数字	15
2.2.5 保存显示过的所有数字	15
2.2.6 显示多位整数或小数	15
2.3 增量划分和进度安排	16
2.4 LED 数码管接口文件	16
本章小结	18
第 3 章 基本数据类型	19
3.1 基本技能	19
3.1.1 C 语言的数据类型	19
3.1.2 标识符	21
3.1.3 常量	21
3.2.4 变量	23
3.2.5 数据的输入、输出	26
3.2 增量式项目驱动	31
本章小结	33

习题 3	34
第 4 章 运算符与表达式	37
4.1 基本技能	37
4.1.1 算术运算符	38
4.1.2 关系运算符	40
4.1.3 逻辑运算符	41
4.1.4 条件运算符	42
4.1.5 逗号运算符	43
4.1.6 位运算符	43
4.1.7 赋值运算符	45
4.1.7 不同数据类型间的转换	46
4.1.8 C 程序的结构	47
4.1.9 顺序结构的 C 语言程序	48
4.2 增量式项目驱动	49
本章小结	51
习题 4	51
第 5 章 选择结构程序设计	54
5.1 基本技能	54
5.1.1 单分支 if 语句	54
5.1.2 双分支 if-else 语句	55
5.1.3 if-else-if 结构	57
5.1.4 if 语句的嵌套	59
5.1.5 开关语句	62
5.2 增量式项目驱动	64
本章小结	72
习题 5	72
第 6 章 循环结构程序设计	77
6.1 基本技能	77
6.1.1 while 循环语句	77
6.1.2 do-while 循环语句	78
6.1.3 for 循环语句	81
6.1.4 循环的嵌套	84
6.1.5 break 语句	85
6.1.6 continue 语句	86
6.2 增量式项目驱动	87
本章小结	98

习题 6	98
第 7 章 函数调用	102
7.1 基本技能	102
7.1.1 函数的分类和定义	102
7.1.2 函数的参数和函数的值	103
7.1.3 函数的调用	106
7.1.4 函数的嵌套和递归调用	107
7.1.5 变量的作用域	109
7.1.6 变量的存储方式和生存期	112
7.2 增量式项目驱动	114
本章小结	125
习题 7	125
第 8 章 数组	129
8.1 基本技能	129
8.1.1 函数的分类和定义	129
8.1.2 二维数组	134
8.1.3 数组作为函数参数	137
8.2 增量式项目驱动	139
本章小结	145
习题 8	145
第 9 章 指针	150
9.1 基本技能	150
9.1.1 指针概述	150
9.1.2 指针变量	151
9.1.3 指针与数组	154
9.1.4 指针与函数	159
9.1.5 指针的内存处理	162
9.2 增量式项目驱动	162
本章小结	168
习题 9	168
第 10 章 字符串处理	173
10.1 字符数组、字符串与指针	173
10.1.1 字符数组、字符串与指针概述	173
10.1.2 字符数组的输入和输出	176
10.2 字符串处理函数	177

本章小结	183
习题 10	183
第 11 章 结构体、共用体和枚举	188
11.1 基本技能	188
11.1.1 结构体类型	188
11.1.2 结构体数组	191
11.1.3 结构体指针和函数	193
11.1.4 共用体类型	196
11.1.5 枚举类型	197
11.2 增量项目驱动	199
本章小结	202
习题 11	203
第 12 章 读写文件	207
12.1 基本技能	207
12.1.1 文件	207
12.1.2 读取文本文件	208
12.1.3 写文本文件	210
12.1.4 读写二进制文件	212
12.1.5 随机读写文件	216
12.2 增量项目驱动	218
本章小结	220
习题 12	220
第 13 章 预编译命令	225
13.1 预编译的概念和作用	225
13.2 文件包含	226
13.3 宏定义	226
13.4 条件编译	230
本章小结	231
习题 13	231
附录 A ASCII 码	233
附录 B C 语言中的关键字	235
附录 C 运算符、优先级和结合性	237
附录 D C 语言中的关键字	238
附录 E C 语言标准头文件	240

第1章

初识 C 语言

- ◎ 熟练掌握 C 语言开发环境 CodeBlocks 的使用
- ◎ 掌握运行 C 程序的基本步骤
- ◎ 了解 C 语言的历史及特点
- ◎ 熟悉算法的表示形式

1.1 C 语言概述

C 语言是国际上广泛流行的计算机高级语言，C 语言的祖先是 BCPL (Basic Combined Programming Language) 语言。1970 年美国 AT&T 贝尔实验室的 Ken Thompson 以 BCPL 语言为基础，设计出了简单且很接近硬件的 B 语言（取 BCPL 的第一个字母），但 B 语言过于简单，功能有限。1972—1973 年间，美国贝尔实验室的 D.M. Ritchie 在 B 语言的基础上设计出了 C 语言。C 语言既保持了 BCPL 和 B 语言的优点（接近硬件），又克服了它们的缺点（过于简单、无数据类型等）。最初的 C 语言只是为描述和实现 UNIX 操作系统提供一种工作语言而设计的。1973 年，Ken Thompson 和 D.M. Ritchie 合作把 UNIX 的 90% 以上用 C 语言改写，即 UNIX 第 5 版。1978 年，Brian W. Kernighan 和 Dennis M. Ritchie 合著了影响深远的名著 *The C Programming Language*，它是第一个 C 语言标准。1978 年以后，C 语言先后移植到大、中、小和微型计算机上。

C 语言是一种应用范围广泛，既可以用来编写系统应用程序，也可以用来编写不依赖计算机硬件的应用程序。C 语言从问世后发展迅速，是目前最受欢迎的编程语言之一。

C 语言具有以下特点：

- ① C 语言简洁、使用方便。C 语言源程序短，编辑程序时工作量小。
- ② C 语言可以对硬件编程，可以像汇编语言一样对位、字节和地址进行操作，在单片机和嵌入式系统中应用广泛。
- ③ C 语言是以函数形式提供给用户的，这些函数可方便调用，并具有多种循环、条件控制结构，使程序完全结构化。
- ④ C 语言具有各种数据类型和运算符，并引入指针概念，程序执行效率高。

- ⑤ 用 C 语言编写的程序可移植性好，适合多种操作系统、多种机型。
- ⑥ 生成目标代码质量高，程序执行效率高。

1.2 C 语言开发环境

C 语言开发环境有很多种，读者可根据需要选择 C 语言的开发环境，如 VC++ 6.0、DEV C++ 等。本书采用 CodeBlocks 作为 C 语言的开发工具，CodeBlocks 是一个开源的全功能跨平台 C/C++ 集成开发环境，支持 Windows 和 GNU/Linux。

1.2.1 运行 C 语言程序的步骤和方法

要编辑一个 C 源程序，并通过 C 语言编程环境 CodeBlocks 进行编译、运行，一般要经过以下步骤，具体过程见图 1-1。

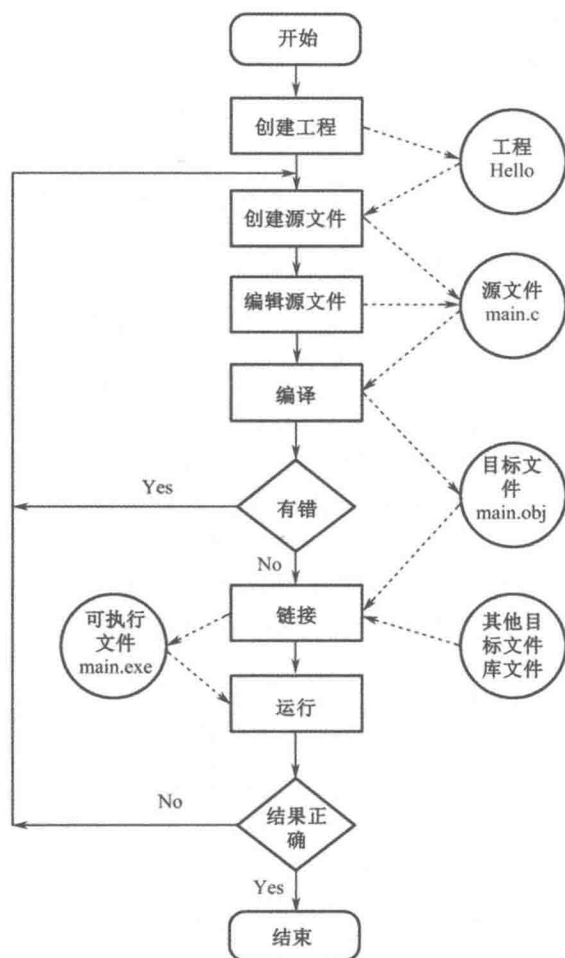


图 1-1 运行 C 程序的步骤和方法

- (1) 创建工程。通过创建一个工程为一个 C 程序提供工作环境。
- (2) 向工程添加源文件。一个工程可包含一个或多个源文件，可包含 0 个或多个头文件。

(3) 编辑源文件。根据 C 语言的语法规则, 使用文本编辑器编写的扩展名为 .c 的文件, 如 first.c、hello.c 等。

(4) 编译。计算机不能直接识别源文件, 必须把源文件转化为计算机能够识别的机器指令。C 程序编译器将检查源文件中是否有语法错误, 如果有语法错误, 将提示有关错误, 如果没有语法错误, 编译器会将源文件转化为一个二进制文件, 该二进制文件被称为源文件的目标文件。目标文件的名字与源文件的名字相同, 但扩展名为 .obj。

(5) 链接。目标文件是供链接器使用的文件, 也就是说目标文件中含有待确定的链接信息, 链接器必须把这些信息替换成真正的链接代码、形成完整的可执行的代码, 即链接器负责产生一个可执行文件。可执行文件的名字与源文件的相同, 但扩展名为.exe。

(6) 运行。将生成的可执行文件交给操作系统去执行。

1.2.2 最简单的 C 语言程序

学习并掌握 C 程序, 首先要熟练使用 C 程序的开发环境, 能用 CodeBlocks 编写简单的 C 源程序, 并对源程序进行编译、链接和运行。

【例 1-1】 编写一个简单的程序, 要求程序输出文字 “Hello, C 程序设计——增量式项目驱动一体化教程!”。

(1) 代码实现 (chp1_1.c)

```
#include <stdio.h>
#include <stdlib.h>

int main(){
    printf("Hello, C 程序设计——增量式项目驱动一体化教程!");
    getchar();
    return 0;
}
```

(2) 运行结果



Hello, C程序设计——增量式项目驱动一体化教程!

(3) 总结

一个 C 语言程序的结构具有以下特点。

① 一个程序由一个或多个源程序文件组成。一个源文件又包含三部分:

❖ 预处理命令。例如:

```
#include <stdio.h>
```

❖ 全局声明。例如, 全局变量声明

```
int num;
```

❖ 函数定义。例如, 定义一个 max() 函数, 用来计算两个数的最大值。

② 函数是 C 程序的主要组成部分。一个 C 语言程序是由一个或多个函数组成的, 其中必须且只能有一个 main() 函数。

③ 一个函数包含两部分:

❖ 函数首部: 包括函数类型、函数名、函数参数名、参数类型。

❖ 函数体: 函数首部下面 { } 中的部分。函数体又包含两部分: 声明部分和执行部分。

- ④ 程序总是从 main()函数开始执行。
- ⑤ 程序对计算机的操作是由函数中的 C 语句完成的。
- ⑥ 每条语句结尾由分号结束。

1.3 算 法

通过以上内容的学习，可以发现一个程序主要包括两方面的信息：

- ① 对数据的描述。程序中需要使用什么样的数据来描述具体问题，数据的类型、数据的组织形式分别如何表示，这就是数据结构（Data Structure）。
- ② 对操作的描述。程序中对数据进行什么样的处理，即要求计算机进行操作的步骤，这就是算法。

1.3.1 算法的定义

广义的算法指“为解决一个问题而采取的方法和步骤”，也就是程序。计算机算法就是为了解决一个问题，计算机所需要执行的方法和步骤，也就是计算机程序。

在软件行业，程序的概念还要广一些，既包括算法，也包括算法操作的对象，即数据。数据是指所有能输入到计算机并被计算机程序处理的符号的介质的总称，各种字母、数字符号的组合、语音、图形、图像等统称为数据。数据经过加工后就成为信息，可由计算机进行处理。

算法与数据的关系是，算法操作的对象是数据。在同一应用环境下，不同的数据之间存在一定的联系，这些数据的组织形式就是数据结构。简单来说，程序 = 算法 + 数据结构。

对同一个问题，可能有不同的解题方法和步骤，不同的方法之间有优劣之分，有的方法较简单，有的较复杂，一般采用方法简单、运算步骤较少的方法。

计算机算法分为两大类：数值计算算法和非数值计算算法。数值计算问题一般可通过数学运算进行解决，如求方程的根、求微分方程等；非数值计算问题一般不能直接解决，要通过建立数学模型，设计合适的算法来解决，见图 1-2 的具体步骤。

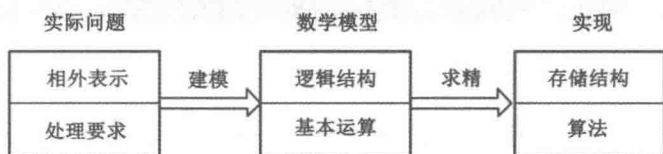


图 1-2 解决问题的步骤

1.3.2 算法的表示

常用的表示算法的方法有：自然语言、流程图、结构化（N/S）流程图和伪代码等。下面对每种方法进行简单介绍。

（1）自然语言

自然语言就是人们日常使用的语言，可以是汉语、英语或其他语言。用自然语言表示的算法通俗易懂，但是文字冗长、容易有歧义。自然语言表示的算法不太严格，要根据上下文才能判断其正确含义。用自然语言描述包含分支和循环结构的算法不太方便，因此，除了一些简单

的问题，一般不用自然语言表示算法。

【例 1-2】 计算 $1+2+3+4+5$ 。

用自然语言描述算法：

步骤 1：先计算 $1+2$ ，得到结果 3。

步骤 2：将步骤 1 得到的结果再加上 3，得到结果 6。

步骤 3：将 6 再加上 4，得 10。

步骤 4：将 10 再加上 5，得 15。

(2) 流程图

流程图是用一些框图来表示各种操作，用图形表示算法，会更直观、易于理解。一些常用的流程图符号如图 1-3 所示。



图 1-3 流程图符号

【例 1-3】 判断 x 是否为正数，画出解决该问题的流程图如图 1-4 所示。

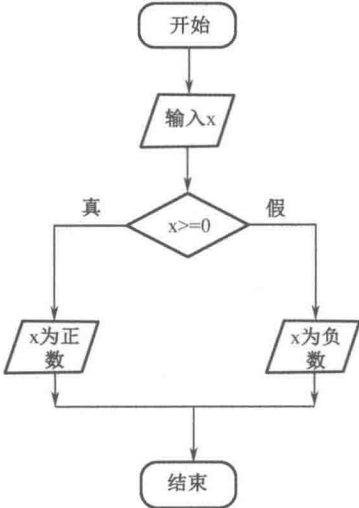


图 1-4 例 1-3 流程图

(3) N-S 结构化流程图

N-S 流程图用以下流程图符号来描述：

顺序结构用图 1-5 所示形式表示，A、B 两个框组成一个顺序结构。

选择结构用图 1-6 所示形式表示，P 表示条件，P 条件成立时执行 A 操作，否则执行 B 操作。

循环结构用图 1-7 和图 1-8 所示的两种形式表示。图 1-7 为当型循环结构，当 P1 条件成立时反复执行 A 操作，直到 P1 条件不成立结束。图 1-8 为直到型循环结构，反复执行 A 操作，直到条件 P1 成立结束。

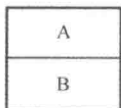


图 1-5 顺序结构

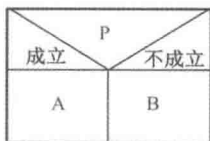


图 1-6 选择结构

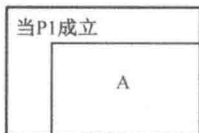


图 1-7 当型循环

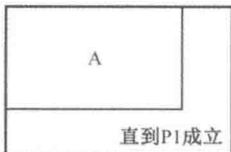


图 1-8 直到型循环

在图 1-5~图 1-8 中，A、B 可以是一个简单的操作，也可以是以上 3 种基本结构之一。

【例 1-4】 计算 5!，画出解决该问题的 N-S 流程图如图 1-9 所示。

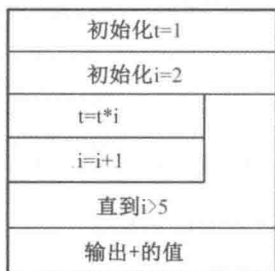


图 1-9 例 1-4 的 N-S 流程图

(4) 伪代码

伪代码是用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。一般每一行（或几行）表示一个基本操作，伪代码不使用图形符号，书写方便，格式紧凑，容易看懂，便于向计算机程序过渡。用伪代码编写的算法并无固定的、严格的语法规则，可以使用任何语言，只要意思表达清楚，便于书写和阅读即可。

【例 1-5】 计算 5!，用伪代码表示该算法如下：

```
begin
    t:=1
    i:=2
    while i ≤ 5
    {
        t:=t*i
        i:=i+1
    }
    print t
end
```

1.3.3 算法举例

【例 1-6】 计算两个整数 x 和 y 的最大值。

(1) 问题分析

如果 $x \geq y$ ，则 x 是较大者；否则， y 是较大者。

(2) 流程图（见图 1-10）

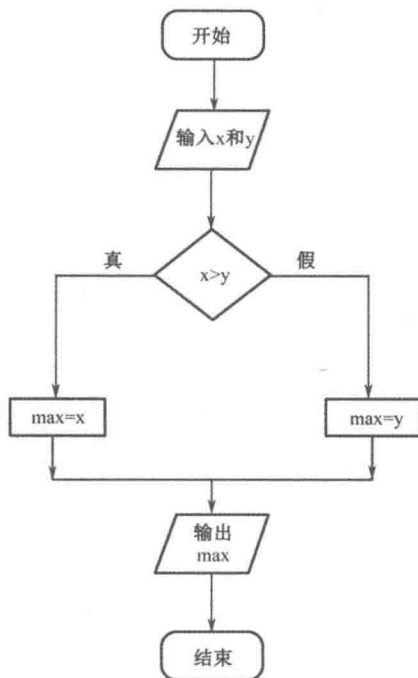


图 1-10 流程图

(3) 代码实现 (chp1_6.c)

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int x, y;
    int max;
    printf("输入 x 和 y 的值: \n");
    scanf("%d,%d",&x,&y);
    if(x>y)                                //判断 x 与 y 的大小，最大值赋给 max
    {
        max=x;
    }
    else
    {
        max=y;
    }
    printf("较大的数是%d \n",max);        //输出较大值
    return 0;
```

}

【例 1-7】 判断某一年是否为闰年，并将结果输出。

(1) 问题分析

某年 x 为闰年的条件是： x 能被 4 整除并且不能被 100 整除，或者 x 能被 100 整除并且能被 400 整除。

(2) 流程图（见图 1-11）

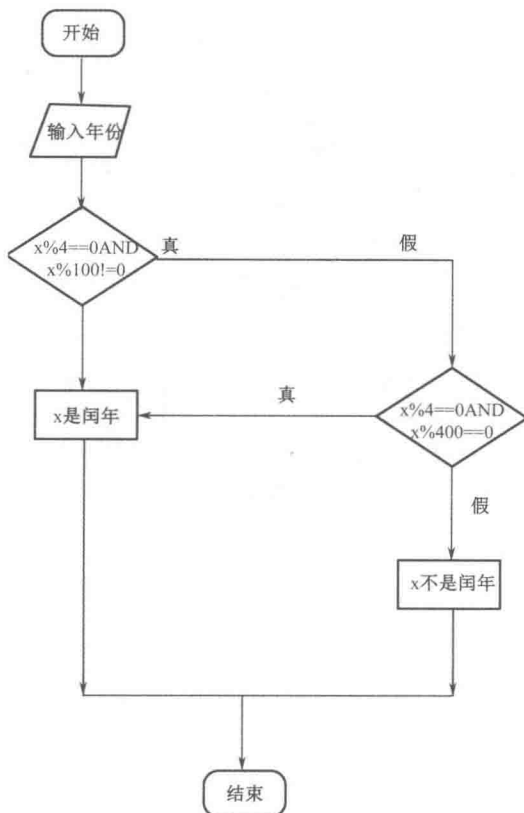


图 1-11 流程图

(3) 代码实现 (chp1_7.c)

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int year; //定义变量
    printf("输入年份:\n");
    scanf("%d", &year);
    //判断是否是闰年的条件
    if((year%4==0&&year%100!=0)||((year%100==0&&year%400==0)))
    {
        printf("%d 是闰年: \n", year);
    }
    else
```