



普通高等教育“十一五”国家级规划教材

全国高校出版社优秀畅销书一等奖

中国高等院校计算机基础教育课程体系规划教材

# C 程序设计教程学习辅导 (第2版)

谭浩强 编著

- 《C程序设计》(已发行1250万册)的姊妹篇
- 适应“少学时”的课程需要
- 使C语言更容易学习
- 突出重点,紧扣最基本的教学要求
- 采用Visual C++作为编译环境

清华大学出版社





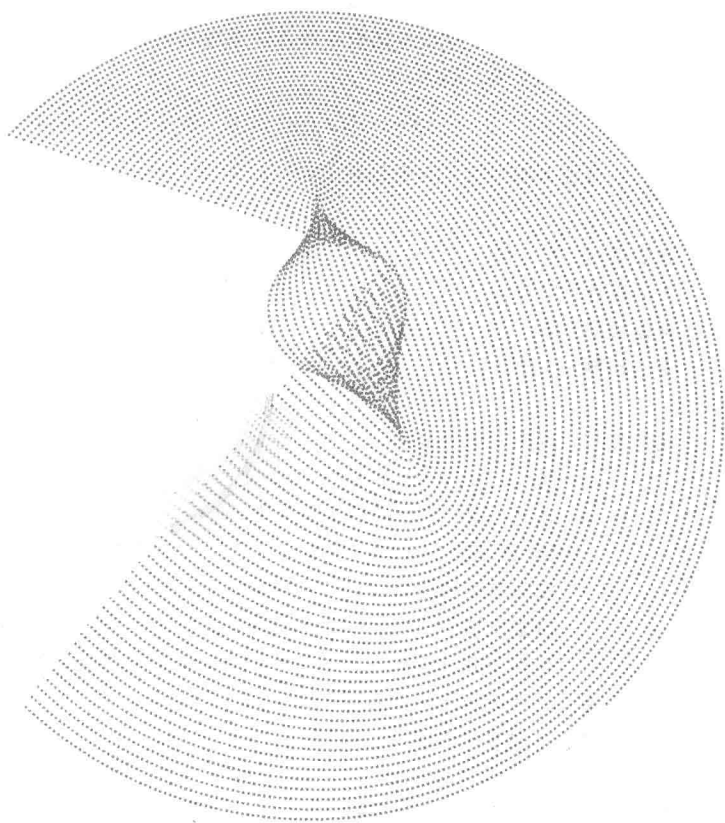
普通高等教育“十一五”国家级规划教材

全国高校出版社优秀畅销书一等奖

中国高等院校计算机基础教育课程体系规划教材

# C程序设计教程学习辅导（第2版）

谭浩强 编著



清华大学出版社

## 内 容 简 介

本书是与谭浩强所著的《C 程序设计教程(第 2 版)》(清华大学出版社出版)配合使用的参考书,全书可分四个部分。第一部分为《C 程序设计教程(第 2 版)》习题与参考解答,包括了该书各章的全部习题,对全部编程习题都给出了参考解答,包括 114 个程序;第二部分为常见错误分析和程序调试;第三部分为 C 语言上机指南,详细介绍了在 Visual C++ 6.0 集成环境下编辑、编译、调试和运行程序的方法;第四部分为上机实验,提供了学习本课程应当进行的 12 个实验。

本书内容丰富、实用性强,是学习 C 语言的一本好参考书。本书不仅可以作为《C 程序设计教程(第 2 版)》的参考书,而且可以作为任何 C 语言教材的参考书;既适于高等学校师生使用,也可供报考计算机等级考试者和其他自学者参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

### 图书在版编目(CIP)数据

C 程序设计教程学习辅导/谭浩强编著. —2 版. —北京:清华大学出版社,2013  
中国高等院校计算机基础教育课程体系规划教材  
ISBN 978-7-302-34041-6

I. ①C… II. ①谭… III. ①C 语言—程序设计—高等学校—教学参考资料 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 234294 号

责任编辑:张 民  
封面设计:傅瑞学  
责任校对:梁 毅  
责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:15.5 字 数:384 千字

版 次:2013 年 12 月第 2 版 印 次:2013 年 12 月第 1 次印刷

印 数:1~3000

定 价:29.50 元

---

产品编号:055259-01

## 中国高等院校计算机基础教育课程体系规划教材

### 编审委员会

主 任：谭浩强

委 员：(按姓氏笔画为序)

王路江	冯博琴	曲建民	刘瑞挺	杨小平
吴文虎	吴功宜	李凤霞	张 森	汪 蕙
高 林	黄心渊	龚沛曾	焦 虹	詹国华

策划编辑：张 民



从 20 世纪 70 年代末、80 年代初开始，我国的高等院校开始面向各个专业的全体大学生开展计算机教育。特别是面向非计算机专业学生的计算机基础教育，牵涉的专业面广、人数众多，影响深远。高校开展计算机基础教育的状况将直接影响我国各行各业、各个领域计算机应用的发展水平。这是一项意义重大而且大有可为的工作，应该引起各方面的充分重视。

20 多年来，全国高等院校计算机基础教育研究会和全国高校从事计算机基础教育的老师始终不渝地在这片未被开垦的土地上辛勤工作，深入探索，努力开拓，积累了丰富的经验，初步形成了一套行之有效的课程体系和教学理念。20 年来高等院校计算机基础教育的发展经历了 3 个阶段：20 世纪 80 年代是初创阶段，带有扫盲的性质，多数学校只开设一门入门课程；20 世纪 90 年代是规范阶段，在全国范围内形成了按 3 个层次进行教学的课程体系，教学的广度和深度都有所发展；进入 21 世纪，开始了深化提高的第 3 阶段，需要在原有基础上再上一个新台阶。

在计算机基础教育的新阶段，要充分认识到计算机基础教育面临的挑战：

(1) 在世界范围内信息技术以空前的速度迅猛发展，新的技术和新的方法层出不穷，要求高等院校计算机基础教育必须跟上信息技术发展的潮流，大力更新教学内容，用信息技术的新成就武装当今的大学生。

(2) 我国国民经济现在处于持续快速稳定发展阶段，需要大力发展信息产业，加快经济与社会信息化的进程，这就迫切需要大批既熟悉本领域业务，又能熟练使用计算机，并能将信息技术应用于本领域的新型专门人才。因此需要大力提高高校计算机基础教育的水平，培养出数以百万计的计算机应用人才。

(3) 从 21 世纪初开始，信息技术教育在我国中小学中全面开展，计算机教育的起点从大学下移到中小学。水涨船高，这样也为提高大学的计算机教育水平创造了十分有利的条件。

迎接 21 世纪的挑战，大力提高我国高等学校计算机基础教育的水平，培养出符合信息时代要求的人才，已成为广大计算机教育工作者的神圣使命和光荣职责。全国高等院校计算机基础教育研究会和清华大学出版社于 2002 年联合成立了“中国高等院校计算机基础教育改革课题调研组”，集中了一批长期在高校计算机基础教育领域从事教学和研究的专家、教授，经过深入调查研究，广泛征求意见，反复讨论修改，提出了高

校计算机基础教育改革思路和课程方案,并于 2004 年、2006 年和 2008 年相继公布了《中国高等院校计算机基础教育课程体系》的三个版本(简称 CFC),由清华大学出版社出版。CFC 公布后,在全国高校中引起强烈的反响,国内知名专家和从事计算机基础教育工作的广大教师一致认为 CFC 提出了一个既体现先进又切合实际的思路和解决方案,该研究成果具有开创性、针对性、前瞻性和可操作性,对发展我国高等院校的计算机基础教育具有重要的指导作用。

为了实现课题组提出的要求,必须有一批与之配套的教材。教材是实现教育思想和教学要求的重要保证,是教学改革中的一项重要的基本建设。如果没有好的教材,提高教学质量只是一句空话。要写好一本教材是不容易的,不仅需要掌握有关的科学技术知识,而且要熟悉自己工作的对象、研究读者的认识规律、善于组织教材内容、具有较好的文字功底,还需要学习一点教育学和心理学的知识等。一本好的计算机基础教材应当具备以下 5 个要素:

(1) 定位准确。要十分明确本教材是为哪一部分读者写的,要有的放矢,不要不问对象,提笔就写。

(2) 内容先进。要能反映计算机科学技术的新成果、新趋势。

(3) 取舍合理。要做到“该有的有,不该有的没有”,不要包罗万象、贪多求全,不应把教材写成手册。

(4) 体系得当。要针对非计算机专业学生的特点,精心设计教材体系,不仅使教材体现科学性和先进性,还要注意循序渐进、降低台阶、分散难点,使学生易于理解。

(5) 风格鲜明。要用通俗易懂的方法和语言叙述复杂的概念。善于运用形象思维,深入浅出,引人入胜。

为了推动各高校的教学,我们愿意与全国各地、各学校的专家和老师共同奋斗,编写和出版一批具有中国特色的、符合非计算机专业学生特点的、受广大读者欢迎的优秀教材。为此,我们成立了“中国高等院校计算机基础教育课程体系规划教材”编审委员会,全面指导本套教材的编写工作。

这套教材具有以下几个特点:

(1) 全面体现 CFC 的思路和课程要求。本套教材的作者多数是课题研究组的成员或参加过课题研讨的专家,对计算机基础教育改革的方向和思路有深切的体会和清醒的认识。因而可以说,本套教材是 CFC 的具体化。

(2) 教材内容体现了信息技术发展的趋势。由于信息技术发展迅速,教材需要不断更新内容,推陈出新。本套教材力求反映信息技术领域中新的发展、新的应用。

(3) 按照非计算机专业学生的特点构建课程内容和教材体系,强调面向应用,注重培养应用能力,针对多数学生的认知规律,尽量采用通俗易懂的方法说明复杂的概念,使学生易于学习。

(4) 考虑到教学对象不同,本套教材包括了各方面所需要的教材(重点课程和一般课程;必修课和选修课;理论课和实践课),供不同学校、不同专业的学生选用。

(5) 本套教材的作者都有较高的学术造诣，有丰富的计算机基础教育的经验，在教材中体现了研究会所倡导的思路和风格，因而符合教学实践，便于采用。

本套教材统一规划、分批组织、陆续出版。希望能得到各位专家、老师和读者的指正，我们将根据计算机技术的发展和广大师生的宝贵意见随时修订，使之不断完善。

全国高等院校计算机基础教育研究会荣誉会长  
“中国高等院校计算机基础教育课程体系规划教材”编审委员会主任

谭浩强

C语言是国内外广泛使用的计算机语言。许多高校都开设了“C语言程序设计”课程。作者于1991年编写了《C程序设计》，由清华大学出版社出版，该书出版后，受到了广大读者的欢迎，大家普遍认为该书概念清晰、叙述详尽、例题丰富、深入浅出、通俗易懂，大多数高校选其作为教材。至2012年年底该书已先后修订了3次，出了4版，重印120多次，累计发行1200多万册，为国内外同类书之首。

由于全国各地区、各类学校情况不尽相同，对C语言的教学要求学时数也有所差别。因此，作者除了编写出版《C程序设计》外，还针对部分学时较少的学校的情况，于2007年编写出版《C程序设计教程》一书，适当减少内容，紧扣基本要求，突出重点。同时编写出版《C程序设计教程学习辅导》一书。2013年年初，作者对《C程序设计教程》一书进行了修订，出版了《C程序设计教程（第2版）》，同时也对《C程序设计教程学习辅导》进行了修订，出版了《C程序设计教程学习辅导（第2版）》，即本书。

本书包括4个部分。

第一部分是“《C程序设计教程（第2版）》习题与参考解答”。在这一部分中包括了《C程序设计教程（第2版）》一书的全部习题，并提供了参考解答。除对其中少数概念问答题，由于能在教材中直接找到答案，不另给出答案外，对所有编程题一律给出参考解答，包括程序清单和运行结果。对于相对简单的问题，只给出程序清单和运行结果，不作详细说明，以便给读者留下思考的空间。对于一些比较复杂的问题还对算法进行了详细的分析介绍，给出流程图，并在程序中加注释以便于读者理解。对少数难度较大的题目(如链表的插入、删除等)还作了比较详细的文字说明。对有些题目，我们给出了两种参考解答，供读者参考和比较，以启发思路。全部程序都在Visual C++ 6.0环境下调试通过。

本书提供的114个不同类型、不同难度的题目，可以作为读者学习《C程序设计教程（第2版）》的补充材料。希望教师和同学们能善于利用这部分内容。教师可以根据教学要求和学生的情况，从习题中指定学生必须完成若干题目，目的是希望通过完成这些习题巩固消化教材中的内容，同时能帮助读者举一反三，深入思考。

习题中包括了不少有价值的、适用于教学的典型问题。由于篇幅和课时的限制，在教材和讲授中不可能介绍很多例子，只能介绍一些典型的例题。教师可以从习题中选一些，作为补充例题在课堂上讲授。对于一般读者，要求他们全部完成这些题目是



困难的,但是可以提倡他们直接阅读参考解答,这对于开阔眼界,丰富知识,理解不同的程序,了解有关算法,掌握编程思路,是大有裨益的。对 C 程序设计有兴趣、有基础的读者,最好能多选一些题目独立完成编程,对提高编程水平会有很大的帮助。有人说:如果能独立完成这些题目的编程,学习 C 语言就基本过关了。

应该说明,本书给出的程序并非是唯一正确的解答,甚至不一定是最佳的一种。对同一个题目可以编出多种程序,我们给出的只是其中的一种或两种,而且程序尽可能便于初学者容易理解,不一定直接搬用于专业的程序设计中。读者在使用本书时,千万不要照抄照搬,读者完全可以编写出更好、更专业的程序。

第二部分是“常见错误分析和程序调试”。作者根据多年教学经验,总结了学生在编写程序时常出现的问题,以提醒读者少犯类似错误。此外,介绍了调试程序的知识和方法,为上机实验打下基础。

第三部分是“C 语言上机指南”。介绍了目前常用的 Visual C++ 6.0 集成环境下的上机方法,使读者上机练习有所遵循。

第四部分是“上机实验”。在这部分中提出了上机实验的要求,介绍了程序调试和测试的初步知识,并且安排了 12 个实验,便于进行实验教学。

本书不仅可以作为《C 程序设计教程(第 2 版)》的参考书,而且可以作为任何 C 语言教材的参考书;既适用于高等学校教学,也可供报考计算机等级考试者和其他自学者参考。

本书难免会有错误和不足之处,作者希望得到广大读者的指正。

谭浩强

2013 年 7 月于清华园

## 第一部分 《C 程序设计教程(第 2 版)》习题与参考解答

第 1 章	程序设计和 C 语言 .....	3
第 2 章	最简单的 C 程序设计——顺序程序设计 .....	8
第 3 章	选择结构程序设计 .....	16
第 4 章	循环结构程序设计 .....	26
第 5 章	利用数组处理批量数据 .....	43
第 6 章	利用函数进行模块化程序设计 .....	65
第 7 章	善于使用指针 .....	91
第 8 章	根据需要创建数据类型 .....	115
第 9 章	利用文件保存数据 .....	148


## 第二部分 常见错误分析和程序调试

第 10 章	常见错误分析 .....	171
第 11 章	程序的调试与测试 .....	187
11.1	程序的调试 .....	187
11.2	程序错误的类型 .....	189
11.3	程序的测试 .....	191

## 第三部分 C 语言上机指南

第 12 章	Visual C++ 的上机操作 .....	199
12.1	Visual C++ 的安装和启动 .....	199
12.2	输入和编辑源程序 .....	200

12.2.1	新建一个 C 源程序的方法 .....	200
12.2.2	打开一个已有的程序 .....	202
12.2.3	通过已有的程序建立一个新程序的方法 .....	202
12.3	编译、连接和运行 .....	203
12.3.1	程序的编译 .....	203
12.3.2	程序的调试 .....	204
12.3.3	程序的连接 .....	206
12.3.4	程序的执行 .....	207
12.4	建立和运行包含多个文件的程序的方法 .....	208
12.4.1	由用户建立项目工作区和项目文件 .....	208
12.4.2	用户只建立项目文件 .....	213
<b>第四部分 上机实验</b>		
<b>第 13 章</b>	<b>实验指导 .....</b>	<b>217</b>
13.1	上机实验的目的 .....	217
13.2	上机实验前的准备工作 .....	218
13.3	上机实验的步骤 .....	218
13.4	实验报告 .....	219
13.5	实验内容安排的原则 .....	219
<b>第 14 章</b>	<b>实验安排 .....</b>	<b>220</b>
14.1	实验 1 C 程序的运行环境和运行 C 程序的方法 .....	220
14.2	实验 2 最简单的 C 程序设计——顺序程序设计 .....	222
14.3	实验 3 选择结构程序设计 .....	224
14.4	实验 4 循环结构程序设计 .....	225
14.5	实验 5 利用数组(一) .....	226
14.6	实验 6 利用数组(二) .....	227
14.7	实验 7 函数调用(一) .....	228
14.8	实验 8 函数调用(二) .....	229
14.9	实验 9 善用指针(一) .....	230
14.10	实验 10 善用指针(二) .....	231
14.11	实验 11 使用结构体 .....	232
14.12	实验 12 文件操作 .....	233
<b>参考文献</b> .....		<b>234</b>



**第一部分**

---

**《C 程序设计教程(第 2 版)》  
习题与参考解答**



# 第1章

## 程序设计和 C 语言

1.1 上机运行本章 3 个例题,熟悉所用系统的上机方法与步骤。

解:略。

1.2 请参照本章例题,编写一个 C 程序,输出以下信息:

```
*****  
          Very Good!  
*****
```

解:编写程序如下:

```
#include <stdio.h>  
int main ()  
{ printf (" *****\n\n");  
  printf ("          Very Good!\n\n");  
  printf (" *****\n");  
  return 0;  
}
```

运行结果:

```
*****  
          Very Good!  
*****
```

1.3 编写一个 C 程序,输入 a,b,c 三个值,输出其中最大者。

解:编写程序如下:

```
#include <stdio.h>  
int main()  
{ int a,b,c,max;  
  printf("please input a,b,c:\n");  
  scanf ("%d,%d,%d",&a,&b,&c);  
  max = a;  
  if (max < b)  
    max = b;  
  if (max < c)
```



```

    max = c;
    printf("The largest number is %d\n",max);
    return 0;
}

```

运行结果:

```

please input a,b,c:
18, -43, 34 ↵
The largest number is 34

```

1.4 先后输入 50 个学生的学号和成绩,要求将其中成绩在 80 分以上的学生的序号和成绩立即输出。请用传统流程图表示其算法。

解: 见图 1-1。

1.5 求  $1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{99} + \frac{1}{100}$ 。请用传统流程图和结构化流程图表示其算法。

解: 传统流程图见图 1-2。结构化流程图见图 1-3。

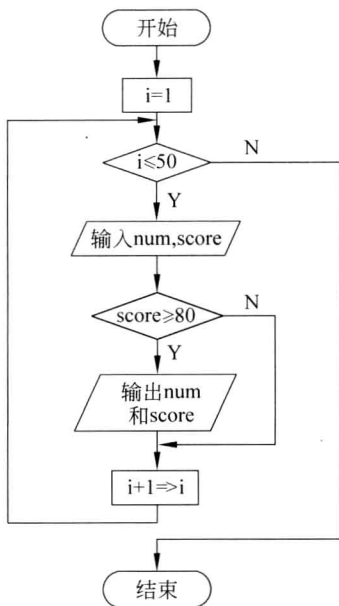


图 1-1

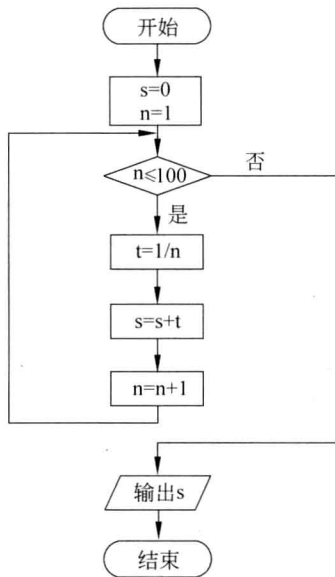


图 1-2

1.6 输入一个年份 year, 判定它是否是闰年, 并输出它是否是闰年的信息。请用结构化流程图表示其算法。

解: 闰年的条件是符合下面二者之一:

- ① 能被 4 整除, 但不能被 100 整除, 如 2016。
- ② 能被 4 整除, 又能被 400 整除, 如 2000 (注意, 能被 100 整除, 不能被 400 整除的年份不是闰年, 如

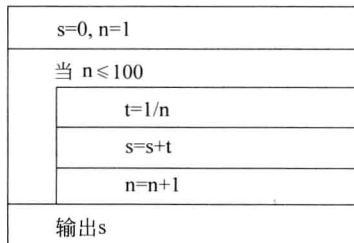


图 1-3

2100)。

画出 N-S 流程图,见图 1-4。

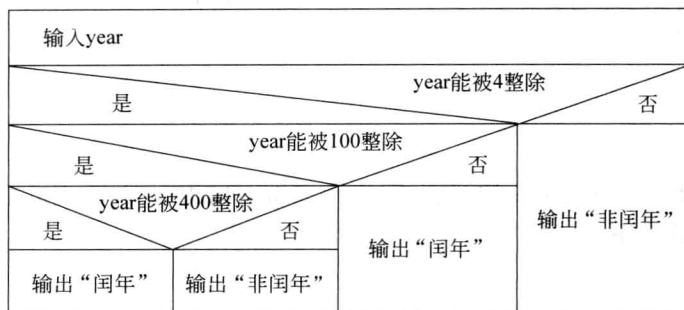


图 1-4

1.7 给出一个大于或等于 3 的正整数,判断它是不是一个素数。请用伪代码表示其算法。

**解:** 所谓素数( prime number)是指除了 1 和它本身之外,不能被其他任何整数整除的整数。例如 17 是一个素数,因为它不能被 2~16 的任何整数整除。而 21 不是素数,因为它能被 3 和 7 整除。要判定一个整数  $m$  是否是素数,只要把  $m$  被  $2 \sim (m-1)$  的各整数整除,如果都除不尽, $m$  就是素数。用伪代码表示的算法如下:

```

begin                                     (算法开始)
input m                                   (输入 m)
i = 2                                     (除数从 2 开始)
while (i <= m - 1)                       (一直进行到被 m - 1 除)
    {if (m%i is equal to 0) flag = 1      (如果 m 被 i 整除,使标志 flag 的值为 1)
      i = i + 1                           (i 加 1,准备下一次循环)
    }
if flag is equal to 1 ,print m is not a prime number
                                         (如果 flag 的值为 1,输出 m 不是素数)
else m is a prime number                 (否则 m 是素数)
end                                       (算法结束)

```

**说明:** 用伪代码写算法时,上面右侧括号内的说明是不需要的。由于有的读者对用伪代码表示算法不太习惯,所以在此加上必要的说明。从上面可以看到,用伪代码写算法,书写灵活,格式自由,修改方便,中英文均可,它是写给人看的(不是让计算机执行的),只要自己和别人能看懂就行。专业人员一般喜欢用伪代码,尽量写得接近计算机语言的形式,以便容易转换为源程序。

1.8 请尝试根据习题 1.4 的算法,用 C 语言编写出程序,并上机运行。

**解:** 编写程序如下:

```

#include <stdio.h>
int main()
{int i,num,score;
  i=1;

```

```

while (i <= 50)
{
scanf ("%d,%d", &num, &score);
if (score >= 80) printf ("%d,%d\n", num, score);
i = i + 1;
}
return 0;
}

```

1.9 请尝试根据习题 1.5 的算法,用 C 语言编写出程序,并上机运行。

解: 编写程序如下:

```

#include <stdio.h>
int main ()
{
int n;
float sum, term;
sum = 0;
n = 1;
while (n <= 100)
{
term = 1.0/n; //term 代表多项式中的某一项的值
sum = sum + term //把各项累加到 sum 中
n = n + 1; //使 n 的值加 1, 准备求下一项
}
printf ("%f\n", sum); //输出总和
return 0;
}

```

运行结果:

5.187378

说明: 第 8 行“term = 1.0/n;”, 分子是 1.0, 表示是实数, 如果写成“term = 1/n;”, 在 C 语言中规定两个整数相除, 结果是整数, 因此当 n > 1 时, 1/n 的值总是等于 0, 最后结果显然不正确。读者可以上机试验一下。关于这个问题, 在学习了第 2 章后会进一步理解的。

1.10 请尝试根据习题 1.6 的算法,用 C 语言编写出程序,并上机运行。

解: 编写程序如下:

```

#include <stdio.h>
int main ()
{
int year;
scanf ("%d", &year);
if (year % 4 == 0) //若 year 能被 4 整除
{
if (year % 100 == 0) //还能被 100 整除
if (year % 400 == 0) //还能被 400 整除
printf ("%d is a leap year.\n", year); //是闰年
else printf ("%d is not a leap year.\n", year); //不能被 400 整除的不是闰年
else printf ("%d is a leap year.\n", year); //能被 4 整除不能被 100 整除的是闰年
}
}

```