

高等学校本科“十二五”规划教材 • 计算机类

精品课程配套指导书

《C/C++语言程序设计》 同步进阶经典100例与习题指导

主 编 李军民
副主编 林淑飞
主 审 龚尚福



西安电子科技大学出版社
<http://www.xduph.com>

高等学校本科“十二五”规划教材·计算机类

精品课程配套指导书

《C/C++语言程序设计》同步进阶 经典 100 例与习题指导

主 编 李军民

副主编 林淑飞

主 审 龚尚福

西安电子科技大学出版社

内 容 简 介

本书针对 C/C++ 语言程序设计中常见且具有特色的应用问题, 精选了 100 个案例, 从算法实现的思维入手, 以实例的形式, 结合计算机处理问题的过程, 对程序设计的方法进行了表述、讲解, 旨在使学生掌握程序设计的基本方法与技能。

本书以程序设计求解问题为主线来组织编写内容, 取材注重典型性与趣味性。每一个实例都是从问题描述、算法分析、代码实现到程序说明、思考与练习等逐层展开, 并对某些程序给出了 N-S 流程图, 在关键代码与生僻语句处做了必要注释, 力求帮助读者理解并掌握计算机程序中算法实现的真谛。

本书还为《C++ 语言程序设计》(龚尚福主编)一书的全部习题配备了解答与学习指导。

本书适合高等学校学生作为计算机程序设计应用课程案例教学使用, 也可供各级程序设计选拔赛、计算机等级考试与软件程序员水平考试练习参考。

图书在版编目(CIP)数据

《C/C++ 语言程序设计》同步进阶经典 100 例与习题指导 / 李军民主编.

—西安: 西安电子科技大学出版社, 2012.3

高等学校本科“十二五”规划教材

ISBN 978-7-5606-2748-9

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

中国版本图书馆 CIP 数据核字(2012)第 010163 号

策 划 马乐惠

责任编辑 马乐惠

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfb001@163.com

经 销 新华书店

印刷单位 陕西华沐印刷科技有限责任公司

版 次 2012 年 3 月第 1 版 2012 年 3 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 20.5

字 数 487 千字

印 数 1~4000 册

定 价 35.00 元

ISBN 978-7-5606-2748-9/TP·1329

XDUP 3040001-1

如有印装问题可调换

前 言

随着科学技术的发展和人类社会的进步，计算机语言的应用日益渗透到国防、工业、农业、企事业和人们日常生活的各个领域，其中 C/C++ 语言是应用面最广的语言之一。C/C++ 语言具有简洁、紧凑、灵活、实用、高效、可移植性好等优点，是高等院校讲授程序设计课程的首选语言。深刻领会和掌握 C 语言程序设计的编程方法，可为后续很多课程(如数据结构、算法分析与设计、操作系统等)的学习提供必要的准备，同时掌握 C 语言也是进一步学习面向对象的程序设计与基于 VC++ 可视化编程环境进行应用程序开发的基础条件。

本书编写的理念是：以问题激发兴趣，以实例驱动学习，以思考促进设计，以练习加强理解。每一个应用实例都是从问题描述到算法思路，从设计分析到代码实现，从程序说明到练习思考，层层展开，逐步引导；对某些程序给出了 N-S 流程图，并在一些关键代码与生僻语句处做了必要注释。本书注重基础知识、基本语法、基本算法的训练，强调问题分析、算法设计与程序实现的有机结合。我们的目的是：在分析问题的步骤中理解算法，在练习编程的过程中掌握设计，在应用 C/C++ 语言知识的基础上更好地学习程序设计思维，培养学生理论联系实际、触类旁通的能力。

本书分为基础篇、提高篇及习题指导三部分内容。精选的 100 个经典案例分基础篇及提高篇两个层次。其中基础篇主要就一般 C 语言程序设计常识与基本方法进行学习训练。提高篇题目主要选自古典问题、趣味游戏、常用算法、特色技巧及历年计算机等级与软件程序员考试、CAM 竞赛等，算法涉及迭代、穷举、递归、分治、贪婪、回溯及分支限界、动态规划等，在编排上尽量采用与一般高等学校教学内容相容，进度匹配，由浅入深，循序渐进，逐步进阶的方式，便于学生学习使用。习题指导部分结合龚尚福主编的《C++ 语言程序设计》，为其全书习题配备了解答与学习指导。两书相得益彰，配合使用，效果更优。

本书直接以各实例问题名称作为目录，使读者一目了然，便于根据需要查找。

本书可作为高等学校计算机相关专业学生学习计算机程序设计的辅导教材，也可作为参加计算机等级考试与软件程序员考试的培训与学习教材，或参加各类软件程序设计竞赛与 CAM 的练习参考。

本书编写得到龚尚福教授的指导与帮助，龚教授还对本书进行了主审，作者在此深表谢意。

本书在早期整理与后期编排过程中，林淑飞、张粉霞等同学参与了许多工作，在习题学习指导部分，史晓楠编写了习题一、二、三，朱莉编写了习题四、五，刘南艳编写了习题六、七、八，孙瑜编写了习题九~习题十四，在此表示感谢。

本书是作者多年从事计算机程序设计教学的成果积累与经验汇集。由于本人水平所限，尽管竭尽全力，仍可能存在不足，敬请读者批评指正。

作 者
2011 年 12 月

目 录

第一部分 基础篇

实例 1 基本库函数的使用.....	1
实例 2 基本数据类型的使用.....	2
实例 3 求一元二次方程的根.....	3
实例 4 个人收入所得税.....	4
实例 5 判断是每年的第几天.....	6
实例 6 循环结构的简单应用(水仙花数).....	7
实例 7 循环结构的简单应用(完全数).....	9
实例 8 数字金字塔.....	10
实例 9 循环应用题(穷举算法).....	12
实例 10 循环应用题(试探算法).....	14
实例 11 一般函数应用(孪生素数).....	16
实例 12 递归调用的简单使用(连续数的倍数).....	17
实例 13 函数嵌套调用的使用(最大公约数和最小公倍数).....	19
实例 14 数组基本概念的应用.....	20
实例 15 一维数组的应用(排序问题).....	22
实例 16 二维数组的应用(矩阵相乘).....	23
实例 17 数组作为函数参数的应用.....	25
实例 18 字符串的使用(统计字符数).....	28
实例 19 指针数组的应用.....	29
实例 20 数组指针的使用.....	31
实例 21 函数指针的使用.....	32
实例 22 指针函数的应用.....	34
实例 23 指针综合应用.....	35
实例 24 利用指针传递参数值.....	38
实例 25 结构体的应用.....	40
实例 26 统计多个学生的多门课的成绩.....	41
实例 27 共用体的应用.....	43

实例 28	链表的应用	45
实例 29	移位运算	47
实例 30	与或位运算	48

第二部分 提高篇

实例 31	进制转换	51
实例 32	贪心算法应用	53
实例 33	循环检查满足条件的数	55
实例 34	求组合数	56
实例 35	猴子分桃问题	58
实例 36	精确除法计算	60
实例 37	大数阶乘	62
实例 38	银行卡问题	64
实例 39	卡布列克运算	66
实例 40	密码问题	68
实例 41	连续整数和问题	69
实例 42	因子和问题	71
实例 43	数组压缩问题	72
实例 44	两路归并排序	75
实例 45	递归全排列	77
实例 46	数组元素平移	78
实例 47	统计数组元素出现次数	80
实例 48	最大公因式问题	82
实例 49	筛选法	84
实例 50	兔子繁衍问题	85
实例 51	分数数列求和	87
实例 52	进制转换	89
实例 53	字符串处理	91
实例 54	最长公共子序列问题	93
实例 55	字符串统计问题	97
实例 56	字符串变换与复原	98
实例 57	字符串置换问题	100
实例 58	数字布置问题	101
实例 59	自然数分解问题	104

实例 60	报数问题	106
实例 61	求定积分	107
实例 62	字母合并问题.....	109
实例 63	结构体变量排序问题.....	112
实例 64	产品工件计数统计.....	113
实例 65	法雷序列	115
实例 66	杨辉三角形输出.....	118
实例 67	环形矩阵输出.....	119
实例 68	取珠子问题	120
实例 69	链表翻转	125
实例 70	部件组成	129
实例 71	数组压缩	131
实例 72	蛇型矩阵	135
实例 73	魔阵	138
实例 74	生成集合问题.....	140
实例 75	变量三角形	141
实例 76	背包问题	144
实例 77	自然数组合问题.....	147
实例 78	方格填数问题.....	149
实例 79	求定和的所有序列.....	152
实例 80	迷宫问题	155
实例 81	二进制环问题.....	160
实例 82	四色定理	162
实例 83	马遍历问题	165
实例 84	比赛日程安排.....	171
实例 85	大整数相乘	173
实例 86	公交线路问题.....	177
实例 87	树和列表问题.....	182
实例 88	三度树列表	185
实例 89	矩阵连乘问题.....	187
实例 90	两字符串的编辑距离.....	189
实例 91	短文的二叉树存储.....	192
实例 92	二叉树遍历	195

实例 93	栈的使用	196
实例 94	循环链表应用(约瑟夫问题)	198
实例 95	递归改非递归	200
实例 96	队列应用	203
实例 97	书稿校对问题	207
实例 98	数字游戏	210
实例 99	过桥问题	212
实例 100	花朵数	214

第三部分 习题指导

习题一	218
习题二	219
习题三	221
习题四	235
习题五	249
习题六	261
习题七	270
习题八	274
习题九	285
习题十	286
习题十一	294
习题十二	303
习题十三	315
习题十四	319

★ 第一部分 基础篇 ★

实例 1 基本库函数的使用

问题描述

编制 100 以内两个整数(随机产生)的加法运算练习程序。

算法分析

编程序首先需要考虑定义哪些相关变量。此题是计算 100 以内的两个整数的和,所以至少要定义三个短整型变量 a、b、c,分别表示加数、被加数、和。a 和 b 使用产生随机数的库函数为其赋初值,通过键盘输入 a 与 b 的和,即 c 的值,由计算机自动判断输入的和是否正确,若不正确给出正确结果。

本程序中用到一些库函数。其中,printf()是输出函数,可以输出一个字符串,也可输出表达式或变量的值;scanf()是输入函数,通过键盘输入一个数据给变量,这两个函数的原型在“stdio.h”文件中;伪随机数函数 int rand(),其原型在“stdlib.h”中,每调用一次函数 rand(),便产生一个 0~32 767 之间的随机数。如果要产生一个 0~99 之间的随机数,可用表达式 rand()%100(取余运算)。一般情况下,如果要得到[a,b]区间的随机数,可用表达式 a+rand()%(b-a)。

源程序

```
#include <stdlib.h>
#include <stdio.h>
void main()
{ int a,b,c,x=0;
  a=rand()%100;
  b=rand()%100;      /*定义 a,b 为 0~100 内的整数*/
  printf("\n%d+%d=",a,b);
  scanf("%d",&c);    /*输入答案*/
  if (c==a+b)       /*判断答案是否正确,并给出正确答案*/
    printf("The answer is right!\n");
  else
  {
    printf("The answer is wrong! ");
```

```
printf("%d+%d=%d\n",a,b,a+b);
}
}
```

思考与练习

1. 本程序只完成了一次计算，若想连续完成五次计算，也就是随机产生五对数，让我们练习计算，程序该如何修改？
2. 请读者复习库函数 `printf()` 和 `scanf()` 的用法。

实例 2 基本数据类型的使用

问题描述

编写一程序，当输入小写字母时，程序能自动转换为大写字母；当输入大写字母时，程序能自动转换为小写字母；当输入非字母时，不进行任何转换；当输入非字母“#”时，程序结束。

算法分析

C 语言的基本数据类型有以下四种：整型、实型、字符型和枚举型。在内存中字符型数据是以 ASCII 码存储的，它和整数的存储形式类似，所以 C 语言中字符数据和整型数据之间可以通用。一个字符数据既可以以字符形式输出，也可以以整数形式输出。以字符形式输出时，需要先将存储单元中的 ASCII 码转换成相应字符，然后输出。以整数形式输出时，直接将 ASCII 码作为整数输出。也可以对字符数据进行算术运算，此时相当于对它们的 ASCII 码进行算术运算。

因为大写字母比其相对应的小写字母的 ASCII 码小 32，所以如果字符变量 `c` 的值是大写字母(`c>='A' && c<='Z'`)，将其转化为小写字母的公式是 `c = c + 32`；如果字符变量 `c` 的值是小写字母(`c>='a' && c<='z'`)，将其转化为大写字母的公式是 `c = c - 32`。

源程序

```
#include <stdio.h>
void main()
{ char c;
  do
  {
    scanf("%c",&c);
    if (c>='a' && c<='z') /*如果是小写字母转化为大写字母*/
        printf("%c",c-32);
    else if (c>='A' && c<='Z') /*如果是大写字母转化为小写字母*/
        printf("%c",c+32);
  }while(c!='#');
}
```

思考与练习

类似还有对字母加密的问题，如将 China 译成密码，其加密规律为：用原来字母后面的第 4 个字母代替原来的字母，例如字母 A 后面第 4 个字母是 E，用 E 代替 A。因此，China 应译为 Glmre。请读者练习编写此问题程序。

实例 3 求一元二次方程的根

问题描述

用 else if 结构在复数范围内求解一元二次方程的根。

算法分析

本实例的目的是使读者掌握 C 语言中的 else if 结构。

关于一元二次方程的根，根据其 a、b、c 的取值，有以下几种情况：

(1) 若系数 a 和 b 同时为 0，则方程有无穷解或无解；

(2) 当 a=0、b≠0 时，则方程有一解，即为 -c/b；

(3) 其他情况(即 a≠0)时，要视 $b^2 - 4ac$ 而定：若 $b^2 - 4ac > 0$ ，则方程有两个实根；若 $b^2 - 4ac < 0$ ，则方程有两个共轭虚数根；若 $b^2 - 4ac = 0$ ，则方程只有一个实根(两个实根相等)。

用 else if 结构将上述分析描述出来，其 N-S 图如图 1-1 所示。

源程序

```
#include <stdio.h>
#include <math.h>
void main()
{ float a,b,c,p,x1,x2,realpart,imagepart;
  printf("请输入一元二次方程的系数 a,b,c 的值: \n");
  scanf("%f,%f,%f",&a,&b,&c);
  if (a==0 && b==0)
    printf("无解\n");
  else if (a==0 && b!=0)
    printf("方程只有一解:%.3f\n",-c/b);
  else
    /*二项式系数 a 不为 0 的情况*/
```

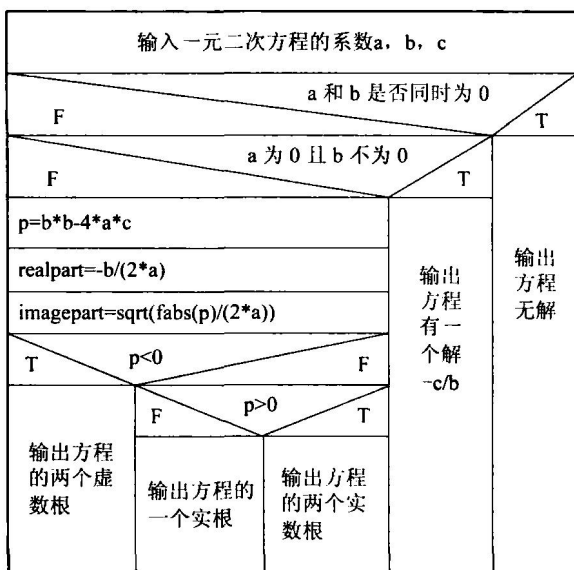


图 1-1

```

    { p=b*b-4*a*c;
      realpart=-b/(2*a);
      imagepart=sqrt(fabs(p))/(2*a);      /*一元二次方程的实虚部*/
      if (p<0)
      { printf("方程有虚数解:");
        printf("%.3f+%.3fi ",realpart,imagepart);
        printf("%.3f+%.3fi\n",realpart,imagepart);
      }
      else if (p>0)
      { x1=realpart+imagepart;
        x2=realpart-imagepart;
        printf("方程有两个实数解:");
        printf("%.3f and %.3f\n",x1,x2);
      }
      else
        printf("方程只有一个实数解:%.3f\n",realpart);
    }
  }
}

```

程序说明

在程序中用到求实数绝对值和平方根的库函数 `fabs()`和 `sqrt()`，这些函数原型在头文件“`math.h`”中。

思考与练习

- 对于求解一元二次方程，也可以这样考虑：首先考虑系数 a 是否等于 0，如果 $a = 0$ ，则再根据 b 是否为 0 来确定解的情况；如果 $a \neq 0$ ，也需要考虑以下两种情况：
 - $b = 0$ ，此时方程变为 $ax^2 + c = 0$ ，再根据 a 和 c 是否异号来确定解。
 - $b \neq 0$ ，此时需根据 $b^2 - 4ac$ 来确定方程的解。
- 请读者根据上述思路编写求解一元二次方程根的程序。

实例 4 个人收入所得税

问题描述

假设公民交纳个人所得税的税率如下(a 代表个人收入， r 代表税率)：

$$\begin{aligned}
 r &= 0 & (a < 2000 \text{ 元}) \\
 r &= 5\% & (2000 \text{ 元} \leq a < 3000 \text{ 元}) \\
 r &= 10\% & (3000 \text{ 元} \leq a < 4000 \text{ 元}) \\
 r &= 15\% & (4000 \text{ 元} \leq a < 5000 \text{ 元}) \\
 r &= 20\% & (5000 \text{ 元} \leq a)
 \end{aligned}$$

编写程序，要求从键盘上输入个人收入，根据以上税率计算出相应的税金并输出。

算法分析

本实例的目的是让读者学习掌握 switch 多分支语句的用法。题中将收入划分为五个不同的范围，对不同范围内的收入，税收利率不同，所以首先要判断所输入的收入属于五种情况中的哪一种，才能确定其对应的税率。我们可以这样细分：小于 2000 元的收入属于第 0 种情况，2000 元~3000 元的收入属于第 1 种情况，3000 元~4000 元的收入属于第 2 种情况，4000 元~5000 元的收入属于第 3 种情况，5000 元及以上的属于第 4 种情况。Switch 多分支选择语句非常适合解决此类问题。

源程序

```
#include <stdio.h>
main()
{   float a,b;
    int i;
    printf("输入某人收入:");
    scanf("%f",&a);
    if (a>=5000)
        i=5;                /*等于或大于 5000 以后，i 值赋为 5*/
    else
        i=a/1000;          /*小于 5000 的数，i 值赋为此数千位上的数字*/
    switch(i)              /*不同收入的所得税计算*/
    {   case 0:
        case 1: b=0; break;
        case 2: b=0.05*(a-2000); break;
        case 3: b=50+0.1*(a-3000); break;
        case 4: b=150+0.15*(a-4000); break;
        case 5: b=300+0.2*(a-5000); break;
    }
    printf("此人应付税金为:%.3f\n",b);
}
```

程序说明

用变量 i 表示收入属于第几种情况。根据上面的分析，如果 $a < 5000$ ，i 就是 a 千位上的数字，此时 $i = (\text{int})(a/1000)$ ，因为 i 是整型变量所以可写成 $i = a/1000$ ；如果 $a \geq 5000$ ，就属于第 5 种情况了，所以此时 $i = 5$ 。

注意：可以将一个实数赋给一个整型的变量，系统会自动截取此实数的整数部分赋给这个整型变量。如 `int x=3.5;` 执行后 x 的值是 3。

思考与练习

1. 程序 switch 语句中每个 case 分支后的 break 是否能够省略，若省去 break 程序会怎么样？

2. 练习编写以下问题的程序。

运输公司对用户计算运费，路程(s)越远，每公里运费越低，标准如下：

$s < 250$ km	没有折扣
$250 \text{ km} \leq s < 500$ km	2%折扣
$500 \text{ km} \leq s < 1000$ km	5%折扣
$1000 \text{ km} \leq s < 2000$ km	8%折扣
$2000 \text{ km} \leq s < 3000$ km	10%折扣
$3000 \text{ km} \leq s$	15%折扣

设每公里每吨货物的基本运费为 p ，货物重为 w ，距离为 s ，折扣为 d ，则总运费 f 的计算公式为 $f = p * w * s * (1 - d)$ 。

提示：折扣的变化是有规律的，折扣的“变化点”都是 250 的倍数。而且 switch 语句有个重要的特性：多个 case 可以共用一组执行语句，如

```
case 'A':  
case 'B':  
case 'C': .....; break;
```

实例 5 判断是每年的第几天

问题描述

通过键盘输入某年某月某日，判断这一天是这一年的第几天？

算法分析

以 1999 年 5 月 20 日为例，要计算这一天是一年中的第几天，应先把前 4 个月的天数加起来，然后再加上 20 即是本年的第几天。除了 2 月份(闰年是 29 天，平年是 28 天)，其他月份的天数都是固定的，如 1 月 31 天，3 月是 31 天，4 月 30 天，……。

首先应该判断这一年是否为闰年，而判断是否为闰年的条件是：能被 4 整除但不能被 100 整除，或者能被 400 整除。然后用 switch 语句来确定每个月的前面几个月的总天数，如第 1 个月前面所有月的天数为 0，第 2 个月前面所有月的天数为 31，第 3 个月前面所有月的天数为 59(先按平年计算，若是闰年再加 1 即可)，……。

源程序

```
#include <stdio.h>  
void main()  
{ int year,month,day,leap,sum;  
  printf("input year,month,day:\n");  
  scanf("%d,%d,%d",&year,&month,&day);  
  if (year%400==0||year%4==0 && year%100!=0) /*判断是不是闰年*/  
      leap=1;  
  else
```

```

        leap=0;
switch(month)                                /*计算 month 月前的天数*/
{
    case 1: sum=0;break;
    case 2: sum=31;break;
    case 3: sum=59;break;
    case 4: sum=90;break;
    case 5: sum=120;break;
    case 6: sum=151;break;
    case 7: sum=181;break;
    case 8: sum=212;break;
    case 9: sum=243;break;
    case 10: sum=273;break;
    case 11: sum=304;break;
    case 12: sum=334;break;
    default: printf("month error:\n");break;
}
sum+=day;
if (month>2) sum+=leap;                      /*如果月份大于 2，则总天数应该加上是闰年的那一天*/
printf("it is the %dth day.\n",sum);
}

```

思考与练习

1. 语句 `if(month>2) sum+=leap;` 能否写成: `if (month>2 && leap) sum++;`
2. 如果学习了数组, 请试着用一维数组来编写本实例 5 程序, 这样程序会更简洁。(提示: 将 12 个月份的天数放到一个长度为 12 的一维数组中。)
3. 中国有句俗语叫“三天打鱼, 两天晒网”。现某人从 2000 年 1 月 1 日起开始“三天打鱼, 两天晒网”, 请编程判断此人在以后的某年某月某日是在“打鱼”还是在“晒网”。其中, 以后的某年某月某日从键盘输入。

提示: 此题需要解决两个问题, 一是计算键盘输入的年月日是从 2000 年 1 月 1 日开始后的第几天; 二是这个天数和 5 的关系, 如果它整除以 5 的余数为 1, 2, 3 则是在“打鱼”, 如果它整除以 5 的余数是 0 和 4 则是在“晒网”。

实例 6 循环结构的简单应用(水仙花数)

问题描述

打印出所有的水仙花数。所谓水仙花数是指一个三位数, 其各位数字立方和应等于该数本身。例如, 153 是一个水仙花数, 因为 $153 = 1^3 + 5^3 + 3^3$ 。

算法分析

本实例的目的是让读者掌握简单 for 循环的用法和分解整数的个位数字的方法。

要求所有的水仙花数，只要判断 100~999 之间的数是否是水仙花数即可。判断的方法是：取出此数的每一位数字，求出它们的立方和，然后判断立方和是否与此数相等。

一个三位数 n ，它的百位数等于该数整除 100 的商，即为 $n/100$ ；它的十位数等于该数整除以 100 的余数再整除以 10 的商，即 $n\%100/10$ ；它的个位数等于该数整除以 10 的余数，即 $n\%10$ 。

源程序

```
#include <stdio.h>
void main()
{   int i,j,k,n;
    printf("water flower's number is :\n");
    for(n=100;n<=999;n++)    /*利用循环验证 100~999 之间的数是否满足水仙花数的条件*/
    {   i=n%10;
        j=n%100/10;
        k=n/100;
        if (n==i*i*i+j*j*j+k*k*k)
            printf("%-5d",n);
    }
}
```

说明：注意算术运算符“/”的多功能性。如果它的两个操作数都是整数，那它的含义就是整除(如 10/3 的结果是 3，而不是 3.333)。

思考与练习

1. 对于语句 `printf("%-5d",n)`，格式控制字符串中 5 前面的“-”说明什么？
2. 通过本例的学习，读者已初步掌握了提取一个整数每一位数字的方法，但本例只给出了提取 3 位整数的每一位数字，如果是 4 或 5 位数呢？亦或是一个未知位数的数呢？
3. 请读者练习下面两题。

(1) 从键盘上输入一个五位整数，对此整数中的五个数值(即万位、千位、百位、十位、个位上的数字)逆序重组成一个新的五位整数并输出。如 12478 可重组成为 87421。

(2) 编写一程序，从键盘任意输入一个数，判断它是否是回文数。(所谓回文数就是正着读和倒着读是一样的，如 13431 是个回文数。)

提示：对于第(1)题，提取五位整数的每一位，可顺着“提出三位整数的每一位”的思路走下去。假设这个五位数为 n ，它的万位数字等于它整除以 10000 的商，即为 $n/10000$ ；它的千位数字等于它整除 10000 的余数再整除以 1000 的商，即为 $n\%10000/1000$ ；同理，它的百位数字等于它整除以 1000 的余数再整除以 100 的商，即为 $n\%/1000/100$ ，(或它的百位数字等于它整除以 100 的商再整除以 10 的余数，即为 $n/100\%10$)；它的十位数字等于它整除以 100 的余数再整除以 10 的商，即为 $n\%100/10$ ；它的个位数字等于它整除以 10 的余数，即为 $n\%10$ 。当然，提取方法不是唯一的，读者可以再考虑其他方法。

对于第(2)题,判断一个数是否是回文数的方法是判断这个数的反序数是否和它本身相等。这就需要提取此数的每位数字,然后将它们逆序组成一个新整数。和本实例所不同的是,被提取的整数的位数是不确定的,这样我们只能从个位开始连续取出整数的“最后个位”,直到它的最高位。这里“最后个位”是指当个位被取出时,整数就变为它除以10的商了,整数的十位变为了个位,百位变为了十位,……,整数的位数少了一位。如132468,先取出8,则整数变为13246,再取出6,则整数变为1324,……,直到取出1为止。这样每次取得都是整数的最后一位,都可用式子 $n\%10$,每次提取后 n 变为 $n/10$ 。

下面是求任意整数 n 的反序数 m 的程序段:

```
m=0;
do
{ m=m*10+n%10;
  n=n/10;
}while (n!=0);
```

实例7 循环结构的简单应用(完全数)

问题描述

编写一程序,求出1000以内的所有完全数。(一个数如果恰好等于它除自身外的所有因子之和,则称该数为完全数。如 $6=1+2+3$,则6是完全数。)

算法分析

要求出1000以内的所有完全数,就需要将1~1000之间的数一一判断是否是完全数,这样“判断数是否是完全数”这一操作要执行1000次,所以使用循环结构,循环体就是“判断一个数是否是完全数”。判断一个数 a 是否是完全数的方法是:求 a 的除自身外的所有因子和 s ,先赋1给 s ,然后一一判断2到 \sqrt{a} 之间的数是否能整除 a (若能整除,此数与商均为 a 的因子),若能就将此数和它整除 a 的商加到 s 上,最后看 s 和 a 是否相等。其N-S图如图1-2所示。

源程序

```
#include <stdio.h>
#include <math.h>
void main()
{ int t,i,s,a=2;
  printf("1000 以内有以下几个完全数: \n");
  while(a<1000)
  { s=1;
```

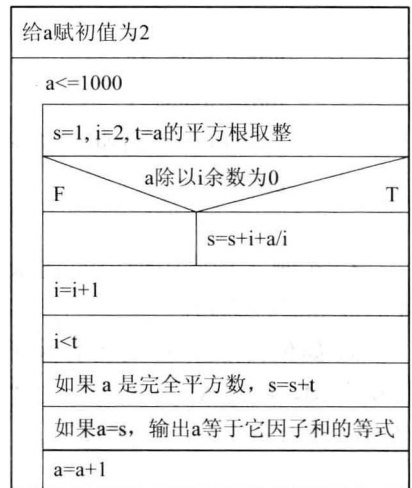


图 1-2