

Turbo C

程序设计实例



唐蔼明 主编

福建科学技术出版社

FUJIAN SCIENCE & TECHNOLOGY PUBLISHING HOUSE

计算程序
图形设计

Turbo C

程序设计实例

主编 唐 蔼 明

编委 谢 泉 钦

福建科学技术出版社

FUJIAN SCIENCE & TECHNOLOGY PUBLISHING HOUSE

内容简介

本书以培养学生 C 程序设计应用能力为目标,注重反映 C 程序设计的思想、方法和技巧。

本书精选了较典型、具有技巧性的程序设计实例,分八个方面系统、详细地介绍了各种算法设计与图形程序设计;着重培养学生分析问题、解决问题、调试纠错能力;旨在让学生在扩大计算机知识面的同时,提高实际编程能力,操作能力。

本书可作为本专科院校、职业技术学院、中专学校计算机应用专业及相关专业的“C 程序设计”课程的教学辅助教材,也可作为初步掌握 C 语言的读者进一步学习提高编程技巧的参考书。

图书在版编目(CIP)数据

Turbo C 程序设计实例 / 唐嵩明主编.—福州:福建科学技术出版社,2007.10
ISBN 978-7-5335-3036-5

I. T… II. 唐… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 115869 号

书 名	Turbo C 程序设计实例
主 编	唐嵩明
出版发行	福建科学技术出版社(福州市东水路 76 号,邮编 350001)
网 址	www.fjstp.com
经 销	各地新华书店
排 版	福州大学校办工厂产品经营部
印 刷	福建地质印刷厂
开 本	787 毫米×1092 毫米 1/16
印 张	17
字 数	429 千字
版 次	2007 年 10 月第 1 版
印 次	2007 年 10 月第 1 次印刷
印 数	1—3100
书 号	ISBN 978-7-5335-3036-5
定 价	25.00 元

书中如有印装质量问题,可直接向本社调换

前 言

C 语言是最常见的一种高级程序设计语言,高等院校理工科专业都开设“C 程序设计”课程,作为程序设计入门。为此,福建省高校计算机水平考试委员会将其列入福建省高等院校计算机应用水平等级考试(二级 C)考试科目,旨在让学生初步掌握程序设计的方法,为今后学习其他计算机课程打下基础,也为今后工作打下计算机应用基础。但是目前,以 Turbo C 为平台介绍各种算法的书还不多,特别是介绍图形设计程序的书更少。

笔者从 1995 年担任闽江学院计算机应用专业、电子专业的“C 程序设计”和“C 图形设计”的课程教学以来,一直注意搜集、整理、编写 Turbo C 有关的算法例题,包括图形模式下的设计程序,在课堂上进行融通讲解,系统地从数值处理、图形输出、数据处理、过程模拟、算式求值、文件(字符、指针)处理、数字组合、图形设计八个方面详细地介绍各种算法与图形设计程序,着重培养学生分析问题、解决问题、调试纠错的能力,旨在让学生在扩大计算机知识面的同时,提高实际编程能力、操作能力。有关教学实践受到学生好评,效果显著,所任教的班级参加福建省教育厅组织的省计算机等级考试(二级 C 语言),班过级率名列全省前茅;所任教学生参加福建省职业大学技能节计算机专业组竞赛,多人获得一、二、三等奖。

本书编写过程中,得到刘雄恩副教授、黄思先副教授的大力支持与精心指导。笔者任教的谢泉钦、严龙两位学生参与了全书编写工作。本书还选用了笔者任教的部分学生的优秀作品,未能一一列名。在此向他们表示衷心感谢。

本书例题均在 UC DOS 7.0、Turbo C 2.0 环境下全部调试成功通过,需要者可以拷贝。
邮箱: Tam@mju.edu.cn , tangaiming97@yahoo.com.cn。

由于时间仓促和我们水平有限,书中难免有错误之处,敬请批评指正。

唐蔼明

于闽江学院二号教学楼闽侯苑

2007 年 7 月

目 录

计算程序篇

第一章 数值处理	1
1.1 19 头牛	1
1.2 分钱	1
1.3 儿子做题	2
1.4 乐队人数	2
1.5 靶子趣谈	2
1.6 里程碑	3
1.7 位等差	4
1.8 岁数	5
1.9 打碎的鸡蛋	6
1.10 分糖	6
1.11 奖牌	7
1.12 同等遗产	7
1.13 菜票问题	8
1.14 出售金鱼	9
1.15 取苹果	9
1.16 狐狸追兔	10
1.17 报数	10
1.18 娶公主	11
1.19 递增牛群	11
1.20 徒子徒孙	12
1.21 特殊数	13
第二章 图形输出	14
2.1 左旋方阵	14
2.2 右旋方阵	15
2.3 螺阵	16
2.4 蛇阵	18
2.5 对角矩阵	19
2.6 魔方阵	20
2.7 倒三角	21
2.8 函数曲线	23
2.9 菱形_1	24
2.10 菱形_2	25
2.11 杨辉三角	26

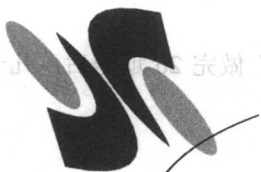
2.12	字母矩形	27
2.13	字母菱形	28
第三章	数据处理	29
3.1	勾股数	29
3.2	求素数	29
3.3	亲密数	30
3.4	三个非零数	30
3.5	平方和	31
3.6	数目平方	31
3.7	四方定理	32
3.8	求最值	33
3.9	数字反序	33
3.10	数值问题-1	33
3.11	数值问题-2	34
3.12	十进制转为二进制	34
3.13	十六进制转化成十进制	35
3.14	十进制转为二~十六任意进制	35
3.15	尾零个数	36
3.16	求7的34次方	36
3.17	印度国王	37
3.18	找最值	38
3.19	连续合数	38
3.20	特殊数列	39
3.21	同和偶数	39
3.22	环状素数	40
3.23	回文素数对	41
3.24	求双百	42
3.25	素数算式	42
3.26	外围之和	43
3.27	编号	44
第四章	过程模拟	45
4.1	称小球	45
4.2	字符滑落	46
4.3	口算小测	47
4.4	算术题分级测验	48
4.5	分牌	49
4.6	八皇后	49
4.7	摆棋子	51
4.8	桥牌	52
4.9	扑克牌	55
4.10	过河	57

4.11	走迷宫	58
4.12	抽奖	61
4.13	选举	61
第五章 算式求值		62
5.1	整数相减	62
5.2	大数相减	63
5.3	实数之和	65
5.4	大数相加	66
5.5	括号配对	67
5.6	大、中、小括号配对	68
5.7	数字重组	68
5.8	计算_1	70
5.9	算术优先级	70
5.10	计算_2	71
5.11	计算器	73
第六章 文件、字符、指针处理		76
6.1	自定义 type	76
6.2	成绩排序_1	76
6.3	成绩排序_2	78
6.4	成绩排序_3	79
6.5	自定义 comp	80
6.6	自定义 copy	81
6.7	统计字母	82
6.8	统计行数与单词数	83
6.9	语法检查	83
6.10	长度个数	84
6.11	统计单词	85
6.12	细胞数目	86
6.13	最长路径	89
6.14	删除字符串	89
第七章 数字组合		91
7.1	特殊矩阵	91
7.2	对角线和	92
7.3	矩阵鞍点	93
7.4	特殊三角形	94
7.5	古诗算式	94
7.6	填数_1	96
7.7	圣诞快乐	97
7.8	特殊序列	98
7.9	填数_2	99

图形程序篇

第八章 基本图形命令及其应用	101
8.1 小球反弹	101
8.2 清屏技术	102
8.3 生长的树	105
8.4 逐渐变浓的树叶	107
8.5 变形椭圆	108
8.6 墙纸	109
8.7 五角星	112
8.8 八卦旋转	115
8.9 随机英文盲打	118
第九章 图视窗口及其应用	123
9.1 甲虫飞行	123
9.2 直升机飞行	124
9.3 小鸡抓虫	125
9.4 卷轴	129
9.5 名片	130
第十章 图像存储及其应用	134
10.1 气球升降	134
10.2 有立体感的墙纸	135
10.3 眼睛闪烁	136
10.4 蝴蝶飞行	138
10.5 投弹	141
10.6 烟花	143
10.7 滴水	145
10.8 卫星飞行	146
10.9 坦克行军	149
10.10 汉诺塔	152
10.11 打降落伞	156
10.12 车与灯	157
10.13 火箭喷火	160
10.14 米老鼠滑雪	160
10.15 飞机炮艇	164
10.16 八卦动感	167
10.17 飞机射击游戏	168
第十一章 调色板	172
11.1 变色荟萃	172
11.2 闪电	173
11.3 风车小球	177

11.4	综合景物	181
第十二章	图形变换函数及其应用	184
12.1	矩形	184
12.2	矩形旋转	185
12.3	地球卫星	187
12.4	五边形旋转	189
12.5	任意角度旋转椭圆	190
第十三章	图像文件处理及汉字技术	195
13.1	调用两色图文件	195
13.2	调用 16 色图文件	196
13.3	调用 256 色图文件	199
13.4	UCDOS 特显汉字技术	204
13.5	字体变化	205
13.6	图形模式下不同点阵汉字的显示	208
13.7	汉字技术	214
13.8	动画名片	216
13.9	澳门回归倒计时	223
第十四章	时钟技术、鼠标的调用及发声	226
14.1	名片	226
14.2	窗户小鸟	227
14.3	时钟	230
14.4	常用鼠标调用函数	234
14.5	视窗功能	237
14.6	画笔	245
14.7	声音与音乐	256
附录	算法习题精选	260
	主要参考文献	262



计算程序篇



第一章 数值处理

1.1 19 头牛

有一个老人在临死前把 3 个儿子叫到跟前，告诉他们把 19 头牛分了，老大分 $1/2$ ，老二分 $1/4$ ，老三分 $1/5$ ，说完就死了。按当地习俗，不能宰牛。问 3 个儿子各能分多少？

(答案：10，5，4)

分析：由于 19 与 2、4、5 都不能整除，所以就不能用平常的方法来解决这个问题。但是，如果仔细一点就可以发觉到： $1/2 + 1/4 + 1/5 = 19/20$ ，而牛的数量刚好为 19。由此，就不难得出该问题的解决办法。

程序：

```
main()
{
    int i;
    for(i=1; i<=10; i++)
        if((i+i/2+2*i/5)==19)
            printf("三个儿子分别分%d头， %d头和%d头.\n", i, i/2, 2*i/5);
    getch();
}
```

1.2 分钱

一元钱分成 1 分、2 分、5 分的，问有多少种分法？

(答案：541 种)

程序：

```
main()
{
    int i, j, sum=0;
    for(i=0; i<=20; i++) /*变量 i 为 5 分钱的数量*/
        for(j=0; j<=(100-5*i)/2; j++) /*变量 j 为 2 分钱的数量，其余的就为 1 分钱*/
            sum++;
    printf("共有%d种分法.", sum);
    getch();
}
```

}

1.3 儿子做题

父亲对儿子说：“做对一道题给 8 分，做错一道题扣 5 分。”做完 26 道题后，儿子得 0 分，问儿子做对了几题？

(答案：10, 16)

程序：

```
#include "stdio.h"
main()
{int i;
  for(i=0;i<=26;i++)
    if (8*i-5*(26-i)==0) /*i 为做对的题数，26-i 为做错的题数*/
      printf("\n 做对了%d 道题，做错了%d 道题.", i, 26-i);
  getch();
}
```

1.4 乐队人数

在爱尔兰守神节那天，举行每年一度的庆祝游戏，指挥者若将乐队排成 10 人、9 人、8 人、7 人、6 人、5 人、4 人、3 人和 2 人一排时，最后的一排总是缺少一个人，那些人想这个位置大概是给数月前死去的乐队成员凯西留着的。指挥者见到总缺一恼火了，叫大家排成一列纵队前进。假定人数不超过 7000 人，那么乐队究竟有多少人？

(答案：2519 人)

程序：

```
main()
{int i, j;
  for (i=9;i<=7000;i+=10) /*用最大的步长取最外层循环变量值*/
  { for (j=9;j>=2;j--) /*模拟排队过程*/
    if ((i+1)%j!=0) break; /*不满足条件则重新取值*/
    if (j==1) /*已满足条件*/
      {printf("乐队共有%d 人。", i);
        exit(0);
      }
  }
}
```

1.5 靶子趣谈

战士们做了一个靶子，靶子分五格，中心是 39 环，四周从左起顺时针是 23、17、24、16 环。战士小李打了若干枪，每一次都击中靶子，并且正好是 100 环。问他打了几枪？每枪多少环？

(答案：6 枪，环数分别为 17、17、17、17、16、16)

程序：

```
main()
{int i, j, k, l, m, n;
```

```

for (i=0;i<3;i++) /*打中 39 环的枪数*/
    for (j=0;j<=4;j++) /*打中 24 环的枪数*/
        for (k=0;k<=4;k++) /*打中 23 环的枪数*/
            for (l=0;l<=6;l++) /*打中 17 环的枪数*/
                { n=100-i*39-j*24-k*23-l*17; /*余下的就为 16 环的枪数*/
                  if (n%16==0)
                      { m=n/16;
                        printf("打中 39 环%d 个, 24 环%d 个, 23 环%d 个, 17 环%d 个, 16 环%d 个, 共打
了%d 枪.", i, j, k, l, m, i+j+k+l+m);
                        exit(0);
                      }
                }
getch();
}

```

1.6 里程碑

甲、乙两个城市有一条 999 千米长的公路。公路旁每隔一公里竖立着一个里程碑，里程碑的半边写着距甲城的距离，另半边写着距乙城的距离。有位司机注意到有的里程碑上所写的数仅用了两个不同的数字，例如 000/999 仅用了 0 和 9，118/881 仅用了 1 和 8。算一算具有这种特征的里程碑共有多少个，是什么样的？

(答案：40 个)

分析：从题意中可知每对数仅用了两个不同的数字，并且两个数字之和恒等于 9，并且每对数之和也应恒等于 999。

解法一：该解法利用三重循环分别求出每个数字的各位数。因为每个数最多只用两个不同的数字，所以每个数中至少有 2 个数字是相同的。再根据两个不同数字之和恒等于 9，不难得出下面的求解过程。

解法一程序：

```

main()
{int i, j, k, m, n=0;
  for (i=0;i<=9;i++)
    for (j=0;j<=9;j++)
      for (k=0;k<=9;k++)
        if (((i==j)&&(9-i==k))||((i==k)&&(9-i==j))||((j==k)&&(9-k==i))||((i==j)&&(j==k)))
            {m=i*100+j*10+k;
              printf("%d/%d ", m, 999-m);
              n++;
            }
  printf("\n 具有这种特征的里程碑共有%d 个.", n);
}

```

解法二：仔细分析题意，可得出如下结论：假设两个数字分别为 a 与 b，则 $b=9-a$ ，由排列组合原理可知，由 a 和 b 所能组成的三位数对如下：

aaa/bbb; aab/bba; aba/bab; abb/baa; bbb/aaa; bba/aab; bab/aba; baa/abb。

其中，每一对数之和恒等于 999(如：aab+bba=999)，并且后四位数为前面四位数中每两

个数的简单对调(如: aab/bba→bba/aab)。由此, 便可得出如下面的求解过程。

解法二程序:

```
main()
{int i, j, k, n=0, a[4];
  for (i=0;i<5;i++)
  {j=9-i;
   a[0]=i*111;
   a[1]=i*110+j;
   a[2]=i*101+j*10;
   a[3]=1*100+j*11;
   for (k=0;k<4;k++)
   { printf("%d: %d/%d, ", ++n, a[k], 999-a[k]);
     printf("%d: %d/%d, ", ++n, 999-a[k], a[k]);
   }
  }
  printf("\n 具有这种特征的里程碑共有%d 个", n);
}
```

1.7 位等差

一个 4 位数, 其千位、百位、十位数字依次组成等差数列, 百位上的数字是个位、千位数字的等比中项, 把该 4 位数的数字反序所得数与原数的和为 11110。求原 4 位数。

(答案: 2468 或 5555)

分析: 设该 4 位数为 abcd, 则由其千位、百位、十位数字依次组成等差数列可得(其中 x 为

等差系数):

$$b=a+x \quad (1)$$

$$c=a+2*x \quad (2)$$

再由“百位上的数字是个位、千位数字的等比中项”可得:

$$a*d=b*b \quad (3)$$

由(1)、(3)可得:

$$a*d=(a+x)*(a+x) \quad (4)$$

程序:

```
main()
{ int i, j;
  clrscr();
  printf("原 4 位数为:");
  for (i=0;i<=9;i++)
  for (j=(i-9)/3;j<=(9-i)/3;j++)
  if (((i+j)*(i+j))==i*(i+3*j))&&((2*i+3*j)*(1000+100+10+1)==11110))
  printf("%d ", (i*1000+(i+j)*100+(i+2*j)*10+(i+3*j)));
}
```

1.8 岁数

一位学生说：“我的岁数的三次方是个4位数，四次方是个6位数。要组成我岁数的三次方和四次方，需要用遍0~9十个数字。”请问他多少岁？

(答案：18岁)

解法一：该解法先分解出岁数的三次方和四次方的每一位，然后再判断这些数字是否重复。

解法一程序：

```
main()
{
    int i, j, k, a[10];
    long k3, k4;
    for (k=10; k<40; k++)
    {
        k3=k*k*k;
        k4=k*k*k*k;
        if (k3/1000>0 && k3/1000<10 && k4/100000>0 && k4/100000<10)
        {
            a[0]=k3/1000;
            a[1]=k3/100-a[0]*10;
            a[2]=k3/10-a[0]*100-a[1]*10;
            a[3]=k3%10;
            a[4]=k4/100000;
            a[5]=k4/10000-a[4]*10;
            a[6]=k4/1000-a[4]*100-a[5]*10;
            a[7]=k4/100-a[4]*1000-a[5]*100-a[6]*10;
            a[8]=k4/10-a[4]*10000-a[5]*1000-a[6]*100-a[7]*10;
            a[9]=k4%10;
            for(i=0; i<10-1; i++)
            for(j=i+1; j<10; j++)
            {
                if (a[i]==a[j])
                {
                    break;
                }
                if (i==9 && j==10)
                {
                    printf("岁数为:%d\n", k);
                    break;
                }
            }
        }
    }
}
```

解法二：该解法与解法一比起来在判断数字是否重复方面有其独特之处，其在判断函数中用一数组 a[10] 来分别表示是否出现该数组下标所对应的数字。例如：若 a[0]=1 表示零的个数为零，若 a[0]=0，表示此前已出现过数字 0。

解法二程序：

```
int check(long a1, long a2)
{
    int i, j, a[10];
    for (i=0; i<=9; i++)
        a[i]=1;
    for (i=0; i<=3; i++, a1/=10)
        if (a[a1%10]) a[a1%10]=0; /*判断当前最后一位是否已出现过，若没有，则将其清零*/
}
```

```

        else return(0);          /*若 a[a1%10]!=0, 表示该数字已重复, 返回 0*/
    for (i=0;i<=5;i++, a2/=10)
    if (a[a2%10]) a[a2%10]=0;
        else return(0);
    return (1);
}
main()
{int i, k;
  long k3, k4;
  for (k=10; k<40; k++)
  { k3=k*k*k;
    k4=k3*k;
    if (k3/1000>0 && k3/1000<10 && k4/100000>0 && k4/100000<10 )
        if (check(k3, k4))
        { printf("岁数为:%d\n", k);
          getch();
          exit(0);
        }
    }
}
}

```

1.9 打碎的鸡蛋

某女士手里拎着一篮鸡蛋，从她身边奔跑而过一匹惊马，吓了她一跳，结果把篮里的鸡蛋打翻打碎了。她说两个一数，三个一数，四个一数，五个一数时，余数分别为 1，2，3 和 4。问篮里原有多少个鸡蛋？

（答案：59 个）

分析：解决这类问题的要点就是找到一个最大的数作为步长，以减少其循环次数，该例题的最大数为 5，故应以 5 为步长。

程序：

```

main()
{ int i;
  for (i=9;i+=5)
  if (((i%5==4)&&(i%4==3)&&(i%3==2)&&(i%2==1))
  {printf("篮里原有%d 个鸡蛋.", i);
    exit(0);
  }
}

```

1.10 分糖

新年晚会老师端着一盘糖给大家分糖，让第一个同学先拿 1 块糖，再把盘中的糖分 1/7 给他；然后让第二个同学拿 2 块糖，再把盘中的糖的 1/7 给他；第三个同学拿 3 块糖后，仍把盘中的糖的 1/7 给他。照这个办法分下去，最后一个同学拿完糖后，糖恰好分完，而且每个人分到的糖块数相同。问共有几人？每人分几块糖？

（答案： 6 人，6 块糖）

分析：由“最后一个同学拿完糖后，糖恰好分完”以及前面的条件可知，最后一个同学所拿的糖的数量刚好等于人数。再根据其他条件可知，第 i 个同学所拿的糖的数量为所剩下糖的 $1/6$ 加上 i ，并且每个同学所拿的糖的数量必为整数，即可得出如下面的求解过程。

程序：

```
main()
{int t=0, m, n, i, j;
  for (m=6;!t;m+=6)
    {t=1;n=m;                                /*n 为最后一个同学所拿到的糖的数量(即人数),
                                              t=1 假设当前 n 满足条件*/

      for (i=m-1;(i>=1)&&t;i--)
        {j=n/6+i;                            /*第 i 个同学分到的糖的数量*/
          n=n/6*7+i;                          /*第 i 个同学分到糖之前所剩的糖的数量*/
          if (j!=m) t=0;                       /*判断每个人分到的糖块数是否相同。不满足就退出模拟过程对人数重新取值*/
        }
      }
  printf("%6d%6d\n", m-6, m-6);
}
```

1.11 奖牌

运动会连续开了 N 天，一共发了 M 枚奖牌。第一天发了一枚再加上剩下的 $1/7$ ，即第一天发了 $[1+(M-1)/7]$ 枚；第二天发了两枚再加上剩下的 $1/7$ ，以后每天按此规律发奖牌，最后一天，即第 N 天，刚好发完剩下的 N 枚奖牌。问运动会开了几天？一共发了几枚奖牌？

(答案：6 天，36 枚)

程序：

```
main()
{int t=0, m, n, i, j;
  clrscr();
  for (i=2;!t;i++)
    { m=i; t=1;                               /*变量 i 为天数，m 为剩余奖牌数，t=1 代表该天满足条件*/
      for (j=i-1;j>=1;j--)                   /*从最后一天开始倒着往前模拟每天的发放奖牌过程*/
        { if (m%6)                           /*若有一天不满足条件就退出模拟过程对天数重新取值*/
          { t=0;
            break;
          }
          m=m/6*7+j;
        }
      }
  printf("一共开了%d天，共发了%d枚奖牌。\\n", i-1, m);
}
```

1.12 同等遗产

父亲临终时，让按下列方式分配他的遗产（克朗）：大儿子分得 100 克朗和剩下的 $1/10$ ，二儿子分得 200 克朗和剩下的 $1/10$ ，三儿子分得 300 克朗和剩下的 $1/10$ 。依此类推，最后发

现这种分法好极了，因为所有儿子分得的钱数恰好相等。问他共有几个儿子？每个儿子分得多少克朗？

（答案：10 个儿子，每人 900 克朗）

注：该题的算法可参照“分糖”。

程序：

```
main()
{ int i, j, k, m, n;
  for (n=600;;n=n+10)
  { k=100+(n-100)/10;
    m=n-k;
    for (i=2;m>0;i++)
      if ((m%10!=0)||((k!=i*100+(m-i*100)/10)) break;
    else m=(m-i*100)-(m-i*100)/10;
    if (m==0)
      { printf("他共有%d 个儿子，每个儿子分得%d 克朗.", i, k);
        getch();
        exit(0);
      }
  }
}
```

1.13 菜票问题

两衣袋中装满了一角与二角五分的菜票，但还不到 20 元，左右两个衣袋中的菜票金额相等。左口袋中每种菜票的数目相同，而右口袋中每种菜票的金额相等，每个口袋中菜票各多少？

（答案： $x=20$ ， $m=35$ ， $n=14$ ，左口袋中 $10 \times 20 + 25 \times 20 = 700$ ，右口袋中 $10 \times 35 + 25 \times 14 = 700$ ）

分析：设左口袋中每种菜票的数量都为 x ；右口袋中每种菜票的钱数都为 y ，一角菜票的张数为 m ，二角五分菜票的张数为 n 。则可得出下面的方程：

$$10x + 25x = 2y \quad (1)$$

$$10m + 25n = 2y \quad (2)$$

$$10m = 25n \quad (3)$$

$$\text{由(3)式可得:} \quad n = m \cdot 2/5 \quad (4)$$

$$\text{由(1)(2)(3)式可得:} \quad x = m \cdot 20/35 \quad (5)$$

根据(4)、(5)两式可得下面的求解过程。

程序：

```
main()
{ int i;
  for (i=5;i<=50;i+=5)
    if (i*10*2%35==0)
      printf("%4d%4d%4d", i*20/35, i, i/5*2);
  getch();
}
```