

精讲100个不同类型的C语言趣味编程题的求解过程
培养编程兴趣，拓宽编程思维，提高编程能力，增强求职竞争力
50小时教学视频 + 758个实例源文件 + 61个程序流程图

C语言

趣味编程100例

贾蓓 郭强 刘占敏 等编著



书中精选的100个实例都兼顾了程序的趣味性和实用性
书中的实例涵盖了C程序设计的基础知识和常用算法
多数实例来自编程大赛和企业面试题，具有很强的实用性
实例遵循先提出问题，再分析问题，最后实现的写作风格
每个实例都给出了详细的实现代码，并给出了详细的注释
超值DVD光盘中提供了大量的C语言教学视频与实例源文件



DVD-ROM

内含海量学习资料

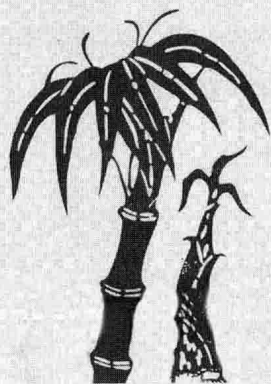
清华大学出版社



C语言

趣味编程100例

贾蓓 郭强 刘占敏 等编著



清华大学出版社

北京

内 容 简 介

本书讲解了 100 个各种类型的 C 语言编程趣味题的求解过程,旨在帮助读者培养编程兴趣,拓宽 C 语言编程思维,提高 C 语言编程能力,掌握用程序设计解决实际问题的方法与技巧。本书取材注重趣味性与实用性,内容涵盖了 C 语言编程的基础知识和常用算法,讲解时给出了实例的详细代码及注释。本书附带 1 张光盘,收录了本书配套多媒体教学视频及实例源文件,以方便读者高效、直观地学习。

本书共分 12 章。第 1 章介绍了 10 个趣味算法入门实例;第 2 章介绍了 11 个趣味数学实例;第 3 章介绍了 10 个趣味整数实例;第 4 章介绍了 8 个趣味分数实例;第 5 章介绍了 7 个趣味素数实例;第 6 章介绍了 8 个趣味逻辑推理实例;第 7 章介绍了 8 个趣味游戏实例;第 8 章介绍了 10 个趣味数组实例;第 9 章介绍了 7 个趣味函数递归实例;第 10 章介绍了 6 个定理与猜想实例;第 11 章介绍了 8 个趣味图形实例;第 12 章介绍了 7 个综合性较强的其他趣味实例。

本书适合高校、职业技术学院及社会培训学校的学生阅读,也适合 C 语言编程爱好者阅读,还可作为各级程序设计选拔赛和全国青少年信息学奥林匹克竞赛的参考书。

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 13701121933

图书在版编目(CIP)数据

C 语言趣味编程 100 例/贾蓓,郭强,刘占敏等编著.--北京:清华大学出版社,2013
ISBN 978-7-302-33808-6

I. ①C… II. ①贾… ②郭… ③刘… III. ①C 语言-程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 212254 号

责任编辑:夏兆彦

封面设计:欧振旭

责任校对:胡伟民

责任印制:李红英

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

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京鑫丰华彩印有限公司

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:21.75 字 数:543 千字

版 次:2014 年 1 月第 1 版 印 次:2014 年 1 月第 1 次印刷

附光盘 1 张

印 数:1~4000

定 价:49.80 元

产品编号:054557-01

前 言

本书以通俗易懂的语言详尽地介绍了使用 C 语言编写的 100 个实例，实例的选取兼顾了趣味性和实用性。通过这些实例的讲解，可以极大地提高读者的学习兴趣，拓宽 C 语言编程思维，提高编程能力，体会程序设计中的乐趣。

本书对每个实例都按照先分析问题再设计算法的步骤进行介绍，根据前面的分析和设计进一步确定程序框架，最后给出完整的代码及程序的执行结果。对于一些实例还特别进行了深入分析和拓展，以开阅读者的思路，加深对问题的理解。在介绍每个独立的趣味问题时，将涉及的 C 语言中的知识点也都详尽地进行了解说，使读者在解读程序的同时能够对 C 语言的常用语法做到融会贯通，牢固掌握。

本书中每个实例代码都给出了详细的注释，方便读者快速地理解代码的含义。而且为了让读者更加高效、直观地学习，作者专门为本书录制了配套的多媒体教学视频辅助读者学习。相信通过演练本书中的实例，你的 C 语言编程能力会有很大的提高，并对相关的算法也将有更进一步的理解，为进一步的实战开发奠定坚实的基础。

本书特色

1. 实例丰富

本书以通俗易懂的语言，深入、细致地介绍了使用 C 语言编写的 100 个实例，在介绍实例的同时将程序开发的基本原理、基本方法和基本技术融入其中。

2. 趣味性

本书在选取例题时注意到了其趣味性，可以极大地提高读者的学习兴趣，使读者体会程序设计中的乐趣。

3. 注释详尽

本书代码注释详尽、流程图画法规范，所有的示例均通过测试可运行，对读者有很好的参考价值。

4. 讲解透彻

本书内容按照不同类型的趣味问题进行分类，力求将每一类问题都讲解透彻并总结出解决该类问题的通用的、一般的规律。

5. 注重基础

本书在注重趣味性的基础上还加强了 C 语言语法知识的学习，将解决问题时所涉及的 C 语言中重要的知识点也进行了详尽的解说。

6. 视频教学

本书中的实例都提供了对应的多媒体教学视频，读者可以先阅读本书内容，再结合多媒体教学视频进行学习，高效而直观，可以获得更佳的学习效果。

本书主要内容

全书共分 12 章。

第 1 章趣味算法入门，通过一些典型算法的介绍，带领读者走进计算机算法的世界，学会使用 C 语言来实现一个算法。

第 2 章趣味数学问题，从与生活相关的一些小例子中抽象出数学公式，再用 C 语言将这些模型化的数学问题表达出来。

第 3 章各种趣味整数，对各类整数问题进行了详细地讲解。

第 4 章趣味分数，讲述了各类与分数相关的趣味问题。

第 5 章趣味素数，介绍了判别素数的方法以及几种特殊素数的验证。

第 6 章趣味逻辑推理，提供了几个有趣的小故事，引导读者进行分析判断并使用 C 语言来实现。

第 7 章趣味游戏，使用 C 语言编写了几个小游戏，通过趣味小游戏来学习编程可以激发读者的学习兴趣。

第 8 章趣味数组，讲解了 C 语言中数组的使用方法。

第 9 章趣味函数递归，深入阐述了 C 语言中递归的概念，将递归融入各个问题的讲解中。

第 10 章定理与猜想，使用 C 语言对常用的一些定理和猜想进行了验证。

第 11 章趣味图形，演示了如何使用 C 语言画一些简单的、常用的图形。

第 12 章其他趣味问题，介绍了一些综合性较强的编程问题。

适合阅读本书的读者

本书内容全面，可读性强，适合阅读的人员有：

- C 语言编程初学者；
- C 语言编程爱好者；
- 普通高校本、专科学生；
- 职业技术学院的学生；
- 程序设计爱好者；
- 各级程序设计选拔赛学员；
- 青少年信息学奥林匹克竞赛人员；
- 有一定开发经验的读者。






本书作者





本书由贾蓓、郭强和刘占敏主笔编写。其他参与编写的人员有韩先锋、何艳芬、李荣亮、刘德环、孙姗姗、王晓燕、杨平、杨艳艳、袁玉健、张锐、张翔、陈明、邓睿、巩民顺、吉燕、水淼、宗志勇、安静、曹方、曾苗苗、陈超。

编者

目 录

第 1 章 趣味算法入门 (📹 教学视频: 90 分钟)	1
1.1 百钱百鸡问题	1
1.2 借书方案知多少	5
1.3 打鱼还是晒网	7
1.4 抓交通肇事犯	11
1.5 兔子产子问题	14
1.6 牛顿迭代法求方程根	16
1.7 最佳存款方案	19
1.8 冒泡排序	20
1.9 折半查找	24
1.10 数制转换	28
第 2 章 趣味数学问题 (📹 教学视频: 59 分钟)	36
2.1 个人所得税问题	36
2.2 存钱问题	43
2.3 分糖果	46
2.4 三色球问题	50
2.5 出售金鱼	52
2.6 求车速	54
2.7 爱因斯坦的数学题	57
2.8 猜牌术	59
2.9 舍罕王的失算	63
2.10 马克思手稿中的数学题	64
2.11 换分币	66
第 3 章 “各种”趣味整数 (📹 教学视频: 49 分钟)	69
3.1 完数	69
3.2 亲密数	72
3.3 自守数	75
3.4 回文数	78
3.5 水仙花数	81
3.6 阿姆斯特朗数	83
3.7 高次方数的尾数	86
3.8 黑洞数	87
3.9 勾股数	91
3.10 不重复的 3 位数	94

第 4 章 趣味分数 ( 教学视频: 47 分钟)	96
4.1 最大公约数	96
4.2 最小公倍数	100
4.3 歌星大奖赛	104
4.4 将真分数分解为埃及分数	108
4.5 列出真分数序列	112
4.6 多项式之和	116
4.7 分数比较	118
4.8 计算分数精确值	121
第 5 章 趣味素数 ( 教学视频: 36 分钟)	125
5.1 素数	125
5.2 哥德巴赫猜想	129
5.3 要发就发	133
5.4 可逆素数	136
5.5 回文素数	140
5.6 孪生素数	142
5.7 梅森素数	144
第 6 章 趣味逻辑推理 ( 教学视频: 49 分钟)	147
6.1 谁家孩子跑的最慢	147
6.2 新郎和新娘	150
6.3 谁在说谎	152
6.4 谁是窃贼	155
6.5 旅客国籍	158
6.6 委派任务	164
6.7 谜语博士的难题	166
6.8 黑与白	172
第 7 章 趣味游戏 ( 教学视频: 40 分钟)	175
7.1 人机猜数	175
7.2 搬山游戏	179
7.3 抢 30	182
7.4 黑白子交换	187
7.5 自动发牌	191
7.6 常胜将军	194
7.7 24 点	195
7.8 掷骰子	201
第 8 章 趣味数组 ( 教学视频: 53 分钟)	204
8.1 狼追兔子	204
8.2 选美比赛	208
8.3 邮票组合	211
8.4 平分 7 筐鱼	212
8.5 农夫过河	216
8.6 矩阵转置	221
8.7 魔方阵	225

8.8	马踏棋盘.....	230
8.9	删除“*”号.....	236
8.10	指定位置插入字符.....	242
第 9 章	趣味函数递归 ( 教学视频: 46 分钟)	244
9.1	递归解决年龄问题.....	244
9.2	递归解决分鱼问题.....	248
9.3	汉诺塔问题.....	251
9.4	猴子吃桃.....	255
9.5	杨辉三角形.....	257
9.6	卡布列克常数.....	260
9.7	逆序输出数字.....	263
第 10 章	定理与猜想 ( 教学视频: 32 分钟)	269
10.1	四方定理.....	269
10.2	角谷猜想.....	272
10.3	π 的近似值.....	274
10.4	尼科彻斯定理.....	277
10.5	奇数平方的有趣性质.....	281
10.6	回文数的形成.....	282
第 11 章	趣味图形 ( 教学视频: 18 分钟)	287
11.1	绘制余弦曲线.....	287
11.2	绘制空心圆.....	289
11.3	绘制空心菱形.....	291
11.4	画直线.....	295
11.5	画圆和圆弧.....	301
11.6	画彩色图形.....	303
11.7	填充彩色图形.....	305
11.8	图形模式下显示字符.....	309
第 12 章	其他趣味问题 ( 教学视频: 35 分钟)	312
12.1	双色球.....	312
12.2	填表格.....	316
12.3	求出符合要求的素数.....	318
12.4	约瑟夫环.....	324
12.5	数据加密问题.....	326
12.6	三色旗.....	329
12.7	统计学生成绩.....	333

第 1 章 趣味算法入门

算法是解决特定问题的方法，是程序设计的基础，是程序设计的灵魂。作为一个算法，应具备 5 个特性，即有穷性、确定性、可行性、输入和输出。计算机算法可分为两大类，分别是数值计算算法和非数值计算算法，数值计算的目的是求解数值，例如求方程的根；非数值计算算法主要用于处理事务领域的问题，如排序、查找等。本章旨在通过一些典型算法的介绍，引领读者走入计算机算法的世界，了解算法设计，学会用 C 语言来实现一个算法。本章主要内容如下：

- 百钱百鸡问题；
- 借书方案知多少；
- 打鱼还是晒网；
- 抓交通肇事犯；
- 兔子产子问题；
- 牛顿迭代法求方程根；
- 最佳存款方案；
- 冒泡排序；
- 折半查找；
- 数制转换。

1.1 百钱百鸡问题

1. 问题描述

中国古代数学家张丘建在他的《算经》中提出了一个著名的“百钱百鸡问题”：一只公鸡值五钱，一只母鸡值三钱，三只小鸡值一钱，现在要用百钱买百鸡，请问公鸡、母鸡、小鸡各多少只？

2. 问题分析

用百钱如果只买公鸡，最多可以买 20 只，但题目要求买一百只，由此可知，所买公鸡的数量肯定在 0~20 之间。同理，母鸡的数量在 0~33 之间。在此把公鸡、母鸡和小鸡的数量分别设为 `cock`、`hen`、`chicken`，则 $cock+hen+chicken=100$ ，因此百钱买百鸡问题就

转化成解不定方程组
$$\begin{cases} cock + hen + chicken = 100 \\ 5 \times cock + 3 \times hen + \frac{chicken}{3} = 100 \end{cases}$$
 的问题了。

3. 算法设计

对于不定方程组，我们可以利用穷举循环的方法来解决，也就是通过对未知数可变范围的穷举，验证方程在什么情况下成立，从而得到相应的解。因公鸡的取值范围是 0~20，可用语句 `for(cock=0;cock<=20;cock++);` 实现。钱的数量是固定的，要买的鸡的数量也是固定的，所以母鸡数量是受到公鸡数量限制的，同理，小鸡数量受到公鸡和母鸡数量的限制，因此我们可以利用三层循环的嵌套来解决：第一层循环控制公鸡的数量，第二层控制母鸡的数量，最内层控制小鸡的数量。

4. 知识点补充

结构化的程序设计包括 3 种基本结构，即顺序结构、选择结构（分支结构）和循环结构（重复结构），利用这 3 种基本结构可以解决很多复杂问题。

顺序结构：一种简单的程序设计，按照程序中语句的顺序依次执行，每条语句都能被执行且只执行一次。

选择结构：包括简单选择和多分支选择结构，可根据条件，判断应该选择哪一条分支来执行相应的语句序列。简单选择结构采用简单或一般的 `if` 语句即可解决，对于复杂的选择结构有两种解决方法：嵌套的 `if` 语句或 `switch` 语句。

循环结构：可根据给定条件，判断是否需要重复执行某一相同程序段。

循环结构包括 3 种：`while(表达式){循环体}`；`do{循环体}while(表达式)`；`for(表达式 1; 表达式 2; 表达式 3){循环体}`。

`while(表达式){循环体}` 结构（又称为当型结构）：首先判断括号中的表达式是否为真，若为真则执行循环体，否则执行循环结构下面的语句，如图 1.1 所示为 `while` 循环示意图。

`do{循环体}while(表达式)` 结构（又称为直到型结构）：不论表达式的值是否为真先执行一次循环体，然后再进行表达式的判断，若表达式为真，则再次进入循环体，直到表达式的值为假，则跳出循环执行下面的语句，如图 1.2 所示为 `do-while` 循环示意图。

`for(表达式 1;表达式 2;表达式 3){循环体}` 结构：首先计算表达式 1(循环变量初值)的值，接下来判断此值是否符合表达式 2（循环的判定条件）的要求，若为真，则执行循环体，接着执行表达式 3（使循环变量值改变的表达式），再次判断表达式 2 的值是否为真，若为真，则继续执行循环体，直到表达式 2 的值为假，跳出循环，执行下面的语句，如图 1.3 所示为 `for` 循环示意图。

在循环语句中，循环体可以由一个或多个语句构成的，当其中某个语句是循环语句时，即一个循环体中完整地包含了另外一个循环，就形成了循环嵌套的结构，我们称之为多重循环，并且把这个循环语句称为外层循环语句，而把循环体中的循环语句称为内层循环语句。在理解多重循环语句时，只要把内层循环语句看作是外层循环语句的循环体的一部分就可以了。在程序执行时，外层循环语句与内层循环语句的关系，有点像钟表的时针与分针的关系，分针走了 60 格，时针才走 1 格。对于多重循环来说，只有内层循环语句执行到判定条件为假时，才返回到其上层循环语句继续执行。

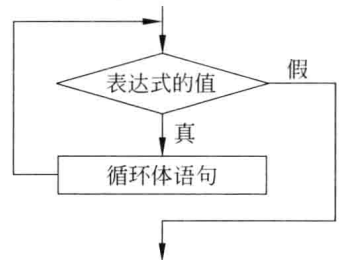


图 1.1 while 循环示意图

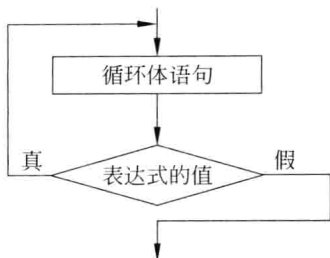


图 1.2 do-while 循环示意图

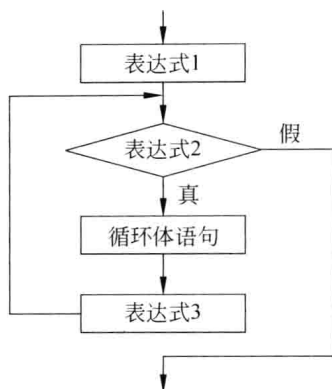


图 1.3 for 循环示意图

5. 确定程序框架

在设计循环时首先要考虑循环的三要素：循环变量的初值、循环的控制条件和使循环趋于结束的循环变量值的改变。

针对本题来说：每层循环的初值是 0（即买的 100 只鸡中，可能没有公鸡，也可能没有母鸡或小鸡）；循环的控制条件是公鸡、母鸡和小鸡用百钱最多能够买到的数量（根据上面分析可知：公鸡最多 20 只，母鸡最多 33 只，小鸡 100 只（虽然百钱可以最多买到 300 只小鸡，但题目要求只买 100 只））；使循环趋于结束的循环变量值的变化：穷举循环的特点就是把所有情况都考虑到，因此每层循环执行一次，对应循环变量的值就要加 1。

程序流程图如图 1.4 所示。

```
for (cock=0; cock<=20; cock++)          /*外层循环控制公鸡数量取值范围 0~20*/
    for (hen=0; hen<=33; hen++)          /*内层循环控制母鸡数量取值范围 0~33*/
        for (chicken=0; chicken<=100; chicken++)
            /*内层循环控制小鸡数量取值范围 0~100*/
            { /*条件控制*/
                printf("cock=%2d, hen=%2d, chicken=%2d\n", cock, hen, chicken); }
```

6. 公鸡、母鸡、小鸡数量的确定

根据这三层循环我们可以得到很多种方案，在这些方案中有些是不符合 $cock+hen+chicken=100$ 并且 $5*cock+3*hen+chicken/3=100$ 这两个条件的，因此结果输出之前我们要把合理的方案筛选出来，即如果结果满足 $cock+hen+chicken=100 \&\& 5*cock+3*hen+chicken/3=100$ 则输出。很明显控制条件即为语句 $if((5*cock+3*hen+chicken/3.0==100) \&\& (cock+hen+chicken==100))$ （注意：C 语言中两个整数相除得到的结果仍为整数，“/”两边有一个数为 float 型得到的结果即为 float 型，在以后进行编程时要注意对“/”两边数据类型处理）。

7. 完整程序

根据上面的分析，编写程序如下：

```
#include<stdio.h>
main()
```

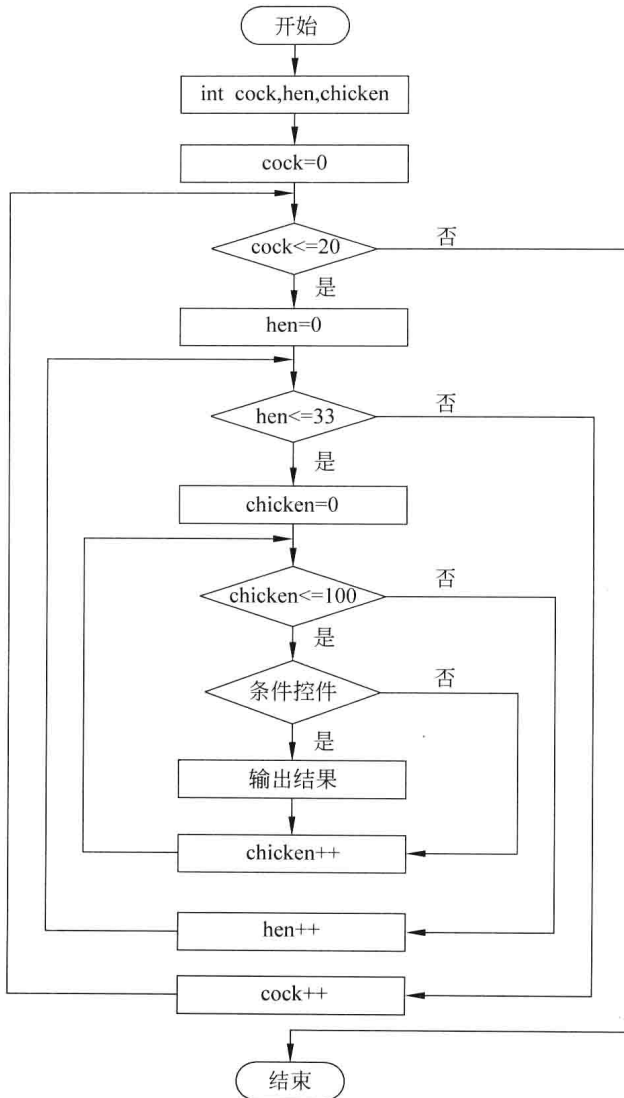


图 1.4 程序流程图

```

{
int cock, hen, chicken;
for (cock=0; cock<=20; cock++)          /*外层循环控制公鸡数量取值范围 0~20*/
    for (hen=0; hen<=33; hen++)        /*内层循环控制母鸡数量取值范围 0~30*/
        for (chicken=0; chicken<=100; chicken++) /*验证解的合理性*/
        {
            if ((5*cock+3*hen+chicken/3.0==100) && (cock+hen+chicken==100))
                printf("cock=%2d, hen=%2d, chicken=%2d\n", cock, hen, chicken);
        }
}

```

8. 运行结果

在 VC 6.0 (Visual C++ 6.0, 以下简称 VC 6.0) 下运行程序, 结果如图 1.5 所示。

9. 问题拓展

以上算法需要穷举尝试 $21 \times 34 \times 101 = 72114$ 次，算法的效率显然太低了。对于这类求解不定方程的问题，各层循环的控制变量直接与方程的未知数相关，且采用对未知数的取值范围穷举和组合的方法可得到全部的解。对于本题来说，公鸡的数量确定后，小鸡的数量就固定为 $100 - \text{cock} - \text{hen}$ ，无须进行穷举了，此时约束条件只有一个： $5 \times \text{cock} + 3 \times \text{hen} + \text{chicken} / 3 = 100$ ；这样我们利用两重循环即可实现：

```
cock= 0,hen=25,chicken=75
cock= 4,hen=18,chicken=78
cock= 8,hen=11,chicken=81
cock=12,hen= 4,chicken=84
```

图 1.5 运行结果

```
#include<stdio.h>
main()
{
    int cock,hen,chicken;
    for(cock=0;cock<=20;cock++)          /*外层循环控制公鸡数量取值范围 0~20*/
        for(hen=0;hen<=33;hen++)        /*内层循环控制母鸡数量取值范围 0~30*/
        {
            chicken=100-cock-hen;        /*在内外层循环条件控制下小鸡数量的取值限制*/
            if(5*cock+3*hen+chicken/3.0==100) /*验证解的合理性*/
                printf("cock=%2d,hen=%2d,chicken=%2d\n",cock,hen,chicken);
        }
}
```

此算法只需尝试 $21 \times 34 = 714$ 次，实现时约束条件中限定了 `chicken` 必须能被 3 整除时，只有 `chicken` 能被 3 整除时才会继续进行约束条件“ $5 \times \text{cock} + 3 \times \text{hen} + \text{chicken} / 3 = 100$ ；”的判断。这样省去了 `chicken` 不能整除 3 时需要进行的算术计算和条件判断，进一步提高了算法的效率。

1.2 借书方案知多少

1. 问题描述

小明有 5 本新书，要借给 A、B、C 这 3 位小朋友，若每人每次只能借 1 本，则可以有多少种不同的借法？

2. 问题分析

本题属于数学当中常见的排列组合问题，即求从 5 个数中取 3 个不同数的排列组合的总数。我们可以将 5 本书进行 1~5 的编号，A、B、C 3 个人每次都可以从 5 本书中任选 1 本，即每人都有 5 种选择，由于 1 本书不可能同时借给一个以上的人，因此只要这 3 个人所选书的编号不同，即为一次有效的借阅方法。

3. 算法设计

对于每个人所选书号，我们可以采用穷举循环来实现，即从每个人可选书号（1、2、3、4、5）的范围内进行穷举，从而得到可行的结果。对第 1 个人的选择，我们可以用循环将其列出：`for(a=1;a<=5;a++)`，同理对于第 2 个人、第 3 个人可以用同样的方法。由于一

本书只能借给一个人，所以第 2 个人的选择会受到第 1 个人的限制，最后一个人的选择会受到第 2 个人的限制，即后面的选择都是在前面选择的前提下进行的，所以可采用循环的嵌套来解决问题。

利用循环解决问题的时候，找到循环的三要素：循环变量的初值、循环的控制条件，以及使循环趋于结束的循环变量值的改变是进行编程的关键。读者可参照第一个例子来寻找一下本题中所对应的循环三要素。本题的输出结果有一个条件限制，即 3 个人所选书号各不相同。在输出语句前只要用一个 if 语句 `if(a!=b&&a!=c&&c!=b)` 判断即可。

4. 完整程序

根据上面的分析，编写程序如下：

```
#include<stdio.h>
main()
{
    int a,b,c,i=0; /*a,b,c 分别用来记录 3 个人所选新书编号,i 用来控制有效借阅次数*/
    printf("A,B,C 三人所选书号分别为: \n");
    for(a=1;a<=5;a++) /*用来控制 A 借阅图书编号*/
        for(b=1;b<=5;b++) /*用来控制 B 借阅图书编号*/
            for(c=1;c<=5;c++) /*用来控制 C 借阅图书编号*/
                if(a!=b&&a!=c&&c!=b) /*此条件用来控制有效借阅组合*/
                    {printf("A:%2d B:%2d c:%2d  ",a,b,c);
                    i++;
                    if(i%4==0)printf("\n");} /*每行最多输出 4 种借阅方法组合*/
    printf("共有%d 种有效借阅方法\n",i); /*输出有效的借阅方法总数*/
}
```

5. 运行结果

在 VC 6.0 下运行程序，结果如图 1.6 所示。

```
A,B,C三人所选书号分别为:
A: 1 B: 2 c: 3  A: 1 B: 2 c: 4  A: 1 B: 2 c: 5  A: 1 B: 3 c: 2
A: 1 B: 3 c: 4  A: 1 B: 3 c: 5  A: 1 B: 4 c: 2  A: 1 B: 4 c: 3
A: 1 B: 4 c: 5  A: 1 B: 5 c: 2  A: 1 B: 5 c: 3  A: 1 B: 5 c: 4
A: 2 B: 1 c: 3  A: 2 B: 1 c: 4  A: 2 B: 1 c: 5  A: 2 B: 3 c: 1
A: 2 B: 3 c: 4  A: 2 B: 3 c: 5  A: 2 B: 4 c: 1  A: 2 B: 4 c: 3
A: 2 B: 4 c: 5  A: 2 B: 5 c: 1  A: 2 B: 5 c: 3  A: 2 B: 5 c: 4
A: 3 B: 1 c: 2  A: 3 B: 1 c: 4  A: 3 B: 1 c: 5  A: 3 B: 2 c: 1
A: 3 B: 2 c: 4  A: 3 B: 2 c: 5  A: 3 B: 4 c: 1  A: 3 B: 4 c: 2
A: 3 B: 4 c: 5  A: 3 B: 5 c: 1  A: 3 B: 5 c: 2  A: 3 B: 5 c: 4
A: 4 B: 1 c: 2  A: 4 B: 1 c: 3  A: 4 B: 1 c: 5  A: 4 B: 2 c: 1
A: 4 B: 2 c: 3  A: 4 B: 2 c: 5  A: 4 B: 3 c: 1  A: 4 B: 3 c: 2
A: 4 B: 3 c: 5  A: 4 B: 5 c: 1  A: 4 B: 5 c: 2  A: 4 B: 5 c: 3
A: 5 B: 1 c: 2  A: 5 B: 1 c: 3  A: 5 B: 1 c: 4  A: 5 B: 2 c: 1
A: 5 B: 2 c: 3  A: 5 B: 2 c: 4  A: 5 B: 3 c: 1  A: 5 B: 3 c: 2
A: 5 B: 3 c: 4  A: 5 B: 4 c: 1  A: 5 B: 4 c: 2  A: 5 B: 4 c: 3
共有60种有效借阅方法
```

图 1.6 运行结果

6. 问题拓展

请大家思考，如果前 2 个人所选书号相同，那么无论第 3 个人所选书号与前两人相同与否都是无效的借阅方法。因此在执行第 3 个循环之前可先行判定前 2 人的编号是否相同，

进而提高程序效率。主要代码如下：

```

for(a=1;a<=5;a++)          /*用来控制 A 借阅图书编号*/
  for(b=1;b<=5;b++)        /*用来控制 B 借阅图书编号*/
    for(c=1;c<=5&&a!=b;c++) /*用来控制 C 借阅图书编号*/
      if(a!=c&&b!=c)       /*此条件用来控制有效借阅组合*/
        {printf("A:%2d B:%2d c:%2d  ",a,b,c);
          i++;
          if(i%4==0)printf("\n");} /*每行最多输出 4 种借阅方法组合*/
        printf("共有%d 种有效借阅方法\n",i); /*输出有效的借阅方法总数*/
    }

```

对原程序稍做修改之后，在长度上虽没有改进仍有三层循环，但是在程序的执行时间上有了很大的提高。对于原程序中的第三层循环来说不管 a、b 的取值是否相同，循环都要重复进行 5 次。修改后的程序在进入循环体之前首先判断 a、b 的取值，如果两者取值相同，内层循环无须重复执行 5 次便可结束。本题的数据较小，对于数据很大的问题，效率的提高更加明显。

1.3 钓鱼还是晒网

1. 问题描述

中国有句俗语叫“三天打鱼两天晒网”。某人从 1990 年 1 月 1 日起开始“三天打鱼两天晒网”，问这个人在以后的某一天中是“打鱼”还是“晒网”。

2. 问题分析

根据题意可以将解题过程分为 3 步：

- (1) 计算从 1990 年 1 月 1 日开始至指定日期共有多少天。
- (2) 由于“打鱼”和“晒网”的周期为 5 天，所以将计算出的天数用 5 去除。
- (3) 根据余数判断他是在“打鱼”还是在“晒网”。

若余数为 1, 2, 3, 则他是在“打鱼”，否则是在“晒网”。

3. 算法设计

该算法为数值计算算法，要利用循环求出指定日期距 1990 年 1 月 1 日的天数，并考虑到循环过程中的闰年情况，闰年二月为 29 天，平年二月为 28 天。判断闰年的方法可以用伪语句描述如下：

如果（能被 4 整除并且不能被 100 整除）或者（能被 400 整除）则该年是闰年；否则不是闰年。

提示：C 语言中判断能否整除可以使用求余运算符“%”。

4. 确定程序框架

程序流程图如图 1.7 所示。

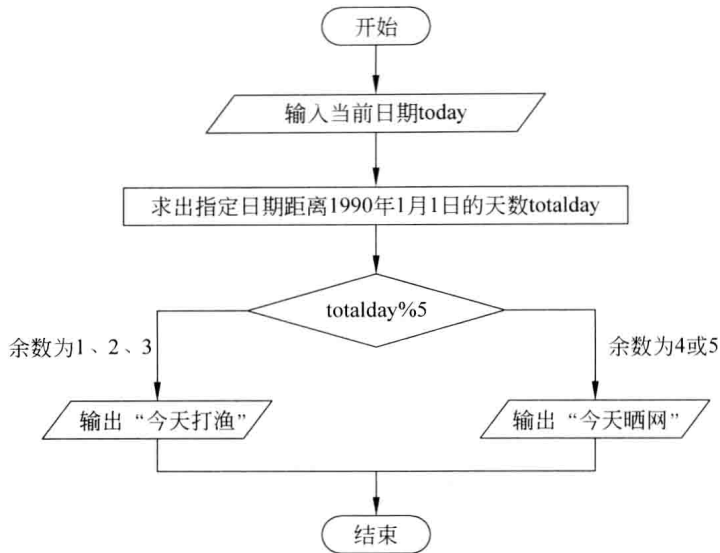


图 1.7 程序流程图

根据流程，构建程序框架如下：

```

#include<stdio.h>
/*定义日期结构体*/
typedef struct date {
int year;
int month;
int day;
}DATE;
int countDay(DATE);
int runYear(int);
void main()
{
    DATE today;                /*指定日期*/
    int totalDay;              /*指定日期距离 1990 年 1 月 1 日的天数*/
    int result;                /*totalDay 对 5 取余的结果*/
    /*输入指定日期,包括年、月、日*/
    printf("please input 指定日期 包括年,月,日 如:1999 1 31\n");
    scanf("%d%d%d",&today.year,&today.month,&today.day);
    /*求出指定日期距离 1990 年 1 月 1 日的天数*/
    totalDay=countDay(today);
    /*天数%5,判断输出打鱼还是晒网*/
    result=totalDay%5;
    if(result>0&&result<4)
        printf("今天打鱼");
    else
        printf("今天晒网");
}
  
```

5. 求出指定日期距离 1990 年 1 月 1 日的天数

这里为整个算法的核心部分。经过分析可以得到指定日期距离 1990 年 1 月 1 日的天数 $totalDay = 1990$ 年至指定年的前一年共有多少天 + 指定年中到指定日期的天数。由于每月天数不同，可以设置一个月份数组 `int perMonth[13]`，存放每月的天数。程序利用年份作为