文章编号：2096-1472(2017)-03-24-02

“百钱买百鸡”问题的C语言算法分析

龙敏敏
（衡阳市广播电视大学，湖南 衡阳 421001）

摘　要：C语言是使用时间最久和最普及的计算机高级程序设计语言之一，属于面向过程的程序设计语言，兼有汇编语言和高级语言的双重特点，是人们学习计算机程序设计的首选语言，也是学习其他计算机课程和进行软件开发的基础。C语言程序设计中的循环语句是C语言的一个难点，可以用来解决许多具有规律性重复操作的实际问题，文章通过对“百钱买百鸡”这个问题的算法进行设计、分析和优化，以寻求解决问题的最优算法。

关键词：C语言；循环语句；百钱买百鸡
中图分类号：TP311.1　　**文献标识码**：A

An Analysis of the Algorithm for "Spending 100 Dollars on 100 Chickens" in C Programming Language

LONG Minmin
(*Hengyang Radio & TV University*,*Hengyang* 421001,*China*)

Abstract: As a process-oriented programming language,C programming language is one of the most classic and popular computer programming languages with the characteristics of the assembly language and the high-level language.It is not only the first choice for the people who begin to learn computer programming,but also the basis for other computer courses and software development.As a difficult point in C Programming Language learning,the loop statement can be used to solve many practical problems of regularly repetitive operation.Taking the case of "spending 100 dollars on 100 chickens",the paper implements design,analysis and optimization,and finally proposes the optimal algorithm.

Keywords:C programming language;loop statement;spending 100 dollars on 100 chickens

1 引言(Introduction)

计算机算法设计是计算机专业学习的核心专业内容，算法设计对于培养一个人的逻辑思维能力具有重要的作用，能进行有效的算法设计是对一个计算机学者的基本要求。求解同一个问题的算法可能有多种，进行算法设计的优劣往往在一定程度上体现出设计者的计算机应用能力，所以，我们要学会如何对一个算法进行分析并进行优化[1,2]。一个算法不一定能说它是最优的，我们所说的最优算法是指在某一种衡量标准下，优于解决该问题的所有可能的算法。一般地，解决某个问题的最优算法是指在时间复杂度的基础上的最优性，时间复杂度是指算法执行的时间长短，通过把算法的核心操作执行的次数作为算法的时间度量[3]，这里，我们使用C语言的循环语句解决“百钱买百鸡问题”，基于算法中的循环次数来判断算法的优劣[4,5]。

2 C语言循环语句(C language loop statement)

C语言是一种广泛使用的程序设计语言，它具有功能丰富、使用灵活、目标程序效率高等特点[6]。C语言是用流程控制语句来控制程序的执行流程，流程控制语句包括顺序结构、选择结构和循环结构三类，C语言中的循环语句又包括for循环语句、while循环语句和do…while循环语句三种，用它们可以解决实际应用中需要重复处理和计算的问题[7,8]。例如，计算1到100之间的整数的和，就需要重复地做加法；求数组中的最大值，需要重复地进行两个数据的比较；求素数问题，需要依次对每个整数进行判断；求百钱买百鸡问题，需要依次穷举每个可能的值通过计算来进行判断；求水仙花问题，需要对范围内的整数依次进行计算判断等等。对于这些复杂的问题，使用循环语句来解决效率很高，下面我们通过“百钱买百鸡”这个经典问题来进行分析。

3 “百钱买百鸡”问题描述(The problem description of "spending 100 dollars on 100 chickens")

中国古代数学家张丘建在他的《算经》中提出了著名的“百钱买百鸡问题”：鸡翁一，值钱五，鸡母一，值钱三，鸡雏三，值钱一，百钱买百鸡，问翁、母、雏各几何？

这是一个古典数学问题，买一只公鸡值5钱，一只母鸡值3钱，三只小鸡值1钱，问如果用100钱买100只鸡，那么公鸡、母鸡和小鸡分别多少只？我们假设公鸡、母鸡和小鸡的个数分别为a、b、c，那么买公鸡的钱数为5*a，买母鸡的钱数为3*b，买小鸡的钱数为c／3；并且a、b、c之和为100，a,b,c都是正整数且c是3的倍数，该问题用数学中的三元一次方程组表达如下：

$$\begin{cases} 5a+3b+c/3=100 \\ a+b+c=100 \end{cases}$$

4 算法设计与分析(Algorithm design and analysis)

对于上述列出的三元一次方程，最直观的方法就是采用三重循环来解决，我们对a、b、c的范围进行限定，用for循环语句进行算法设计，得出算法一如下：

```
main() {
int a,b,c;
for(a=1;a<=100;a++)
for(b=1;b<=100;b++)
for(c=1;c<=100;c++) {
if((5*a+3*b+c/3==100)&&(a+b+c==100)&&(c%3==0))
 printf("公鸡=%d,母鸡=%d,小鸡=%d\n",a,b,c);}
 }
```

程序运行结果如下：

公鸡=4,母鸡=8,小鸡=78

公鸡=8,母鸡=11,小鸡=81

公鸡=12,母鸡=4,小鸡=84

该算法直接将三元一次方程转化为C语言三重循环程序，没有进一步考虑a、b、c更小的取值范围，所以导致循环次数为100*100*100=10^6，算法的复杂度太高，所以我们可以根据每种鸡的价钱先确定a、b、c的取值范围：公鸡为5钱，所以a的取值范围为1—20，当然a的取值的不可能是20，否则100钱刚好买20只公鸡与百鸡的题意不符；b的取值范围为1—33，c的取值范围为3—99；得到算法二如下：

```
main() {
int a,b,c;
for(a=1;a<=19;a++)
for(b=1;b<=33;b++)
for(c=3;c<=99;c++) {
if((5*a+3*b+c/3==100)&&(a+b+c==100)&&(c%3==0))
 printf("公鸡=%d,母鸡=%d,小鸡=%d\n",a,b,c);}
 }
```

对于这个算法我们可以计算一下它的循环次数为19*33*97=60819次，可见算法的效率还是不高。那么我们还能不能再进行一定的改进呢？通过分析，买小鸡的钱数为c/3，那么小鸡的数量c是3的倍数，所以为了提高执行效率，我们可以把对小鸡的for循环语句中c的步长值设为3，这样可以减少一定的循环次数，也可以去掉c%3==0这个约束条件，得到算法三如下：

```
main() {
int a,b,c;
for(a=1;a<=19;a++)
for(b=1;b<=33;b++)
for(c=3;c<=99;c+=3) {
if((5*a+3*b+c/3==100)&&(a+b+c==100))
 printf("公鸡=%d,母鸡=%d,小鸡=%d\n",a,b,c);}
 }
```

此时我们再来计算一下以上算法需要执行的循环次数为19*33*33=20691次，很明显，此时算法的执行效率有了一定的提高，缩小小鸡c的取值范围使得算法的循环次数变为上个算法的三分之一。那么这一次改进后的算法就是最理想的算法吗？还有没有进一步改进的空间呢？答案是肯定的。其实只要我们再仔细观察可以发现，这个算法实际上可以不用三重循环，而只用二重循环就可以达到目的，因为二重循环比三重循环会大大降低循环次数，因而提高执行速度。由于公鸡a和母鸡b的数量确定后，其实小鸡c的数量也就等于确定了，即c=100−a−b，从而就不需要再进行加一重循环了，此时又可以去掉a+b+c=100这个约束条件，得到算法四如下：

```
main() {
int a,b,c;
for(a=1;a<=19;a++)
for(b=1;b<=33;b++)
{ c=100-a-b;
if((5*a+3*b+c/3==100)&&(c%3==0))
 printf("公鸡=%d,母鸡=%d,小鸡=%d\n",a,b,c);}
 }
```

以上算法的循环次数只有19*33=627次，相对上个算法又大幅度提高了算法的执行效率。我们尝试再进行进一步的分析：　我们可以进一步分析出公鸡、母鸡和小鸡的更小范围，根据一只公鸡5钱，一只母鸡3钱，三只小鸡1钱，得知，若a的值为15时，公鸡的总钱为75钱，剩下的25钱最多可买75只小鸡，达不到百鸡数量的要求，所以公鸡a的值必定小于15，a的大致范围是1—14；若b的值为25时，母鸡的总钱为75钱，剩下的25钱最多可买75只小鸡，刚好达到百鸡的数量，所以b的值不会超过25，b的大致取值范围为1—25；由于a、b的最大值分别为14和25，那么要满足百鸡数量的话，c的最小值至少应是61，又当c的值为90只时，小鸡的总钱才30钱，剩下10只即使都买公鸡也只50钱，总钱达不到百钱的要求，所以c的值肯定是小于90，又c是3的倍数，所以大致估算c的取值范围为63—87。既然a、b、c的大致范围都确定了，我们可以得到下列算法五：

```
main() {
int a,b,c;
for(a=1;a<=14;a++)
for(c=63;c<=87;c+=3)
{b=100-a-c;
if(5*a+3*b+c/3==100)
 printf("公鸡=%d,母鸡=%d,小鸡=%d\n",a,b,c);}
 }
```

以上算法循环语句执行的次数仅为14*9=126次，这大大提高了算法的执行速度。这个算法是先确定公鸡a和小鸡c的

（下转第17页）