



## 百钱买百鸡

### 【问题描述】

鸡翁一，值钱五，鸡母一，值钱三，鸡雏三，值钱一，百钱买百鸡，问鸡翁、鸡母、鸡雏各几何？

### 【运行结果】

```
0 25 75
4 18 78
8 11 81
12 4 84
```

### 【分析】

设鸡翁  $x$ ，鸡母  $y$ ，鸡雏  $z$ ，则可以列出方程组：

$$\begin{cases} x + y + z = 100 \\ 5x + 3y + \frac{z}{3} = 100 \end{cases}$$

同时满足上述两个方程的  $x$ 、 $y$ 、 $z$  值就是所求。列举  $x$ 、 $y$ 、 $z$  的所有可能解，然后判断这些可能解是否能使方程组成立。能使方程组成立的，就是真正的解。

再进一步分析， $x$  的取值范围是  $0 \sim \frac{100}{5}$ ， $y$  的取值范围是  $0 \sim \frac{100}{3}$ ， $z$  的取值范围是  $0 \sim 3 \times 100$ 。

### 【方法一】三重循环

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     //列举鸡翁数的所有可能
6.     for (int x=0; x<=100/5; x++)
7.     {
8.         //列举鸡母数的所有可能
9.         for (int y=0; y<=100/3; y++)
10.        {
11.            //列举鸡雏数的所有可能
12.            for (int z=0; z<=3*100; z++)
13.            {
```



```
14.          //满足两个方程组
15.          if (5*x+3*y+z/3==100 && x+y+z==100 && z%3==0)
16.          {
17.              //输出 x、y、z 值
18.              cout << x << " " << y << " " << z << endl;
19.          }
20.      }
21.  }
22.  }
23.  return 0;
24. }
```

这里用了一个三层循环的程序解决问题。对于 x 的每个取值，for 的 y 循环体都要遍历 y 的所有取值；对于 y 的每个取值，for 的 z 循环体都要遍历 z 的所有取值；对于 z 的每个取值，都要执行一次 if 语句。

不难算出，在程序的执行过程中，作为最内层循环体的 if 语句执行的次数是：

$$(1 + \frac{100}{5}) \times (1 + \frac{100}{3}) \times (1 + 3 \times 100) = 214914$$

能不能减少循环次数呢？

## 【方法二】二重循环

因为鸡翁、鸡母、鸡雏共 100 只，所以一旦确定鸡翁 x 和鸡母 y 的数量，鸡雏就只能购买  $100 - x - y$  只。

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     //列举鸡翁数的所有可能
6.     for (int x=0; x<=100/5; x++)
7.     {
8.         //列举鸡母数的所有可能
9.         for (int y=0; y<=100/3; y++)
10.        {
11.            //根据 x, y 计算鸡雏的数量
12.            int z = 100-x-y;
13.            //判断总钱数是否符合条件
14.            if (5*x+3*y+z/3==100 && z%3==0)
15.            {
16.                cout << x << " " << y << " " << z << endl;
```



```
17.         }  
18.     }  
19. }  
20.     return 0;  
21. }
```

类似的不定方程问题, 都可以用循环来求解。为提高效率, 可以在程序中进行适当优化, 减少循环体的执行次数。



## 百钱买百鸡

### 【问题描述】

鸡翁一，值钱五，鸡母一，值钱三，鸡雏三，值钱一，百钱买百鸡，问鸡翁、鸡母、鸡雏各几何？

### 【运行结果】

```
0 25 75
4 18 78
8 11 81
12 4 84
```

### 【分析】

设鸡翁  $x$ ，鸡母  $y$ ，鸡雏  $z$ ，则可以列出方程组：

$$\begin{cases} x + y + z = 100 \\ 5x + 3y + \frac{z}{3} = 100 \end{cases}$$

同时满足上述两个方程的  $x$ 、 $y$ 、 $z$  值就是所求。列举  $x$ 、 $y$ 、 $z$  的所有可能解，然后判断这些可能解是否能使方程组成立。能使方程组成立的，就是真正的解。

再进一步分析， $x$  的取值范围是  $0 \sim \frac{100}{5}$ ， $y$  的取值范围是  $0 \sim \frac{100}{3}$ ， $z$  的取值范围是  $0 \sim 3 \times 100$ 。

### 【方法一】三重循环

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     //列举鸡翁数的所有可能
6.     for (int x=0; x<=100/5; x++)
7.     {
8.         //列举鸡母数的所有可能
9.         for (int y=0; y<=100/3; y++)
10.        {
11.            //列举鸡雏数的所有可能
12.            for (int z=0; z<=3*100; z++)
13.            {
```



```
14.          //满足两个方程组
15.          if (5*x+3*y+z/3==100 && x+y+z==100 && z%3==0)
16.          {
17.              //输出 x、y、z 值
18.              cout << x << " " << y << " " << z << endl;
19.          }
20.      }
21.  }
22.  }
23.  return 0;
24. }
```

这里用了一个三层循环的程序解决问题。对于 x 的每个取值，for 的 y 循环体都要遍历 y 的所有取值；对于 y 的每个取值，for 的 z 循环体都要遍历 z 的所有取值；对于 z 的每个取值，都要执行一次 if 语句。

不难算出，在程序的执行过程中，作为最内层循环体的 if 语句执行的次数是：

$$(1 + \frac{100}{5}) \times (1 + \frac{100}{3}) \times (1 + 3 \times 100) = 214914$$

能不能减少循环次数呢？

### 【方法二】二重循环

因为鸡翁、鸡母、鸡雏共 100 只，所以一旦确定鸡翁 x 和鸡母 y 的数量，鸡雏就只能购买  $100 - x - y$  只。

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     //列举鸡翁数的所有可能
6.     for (int x=0; x<=100/5; x++)
7.     {
8.         //列举鸡母数的所有可能
9.         for (int y=0; y<=100/3; y++)
10.        {
11.            //根据 x, y 计算鸡雏的数量
12.            int z = 100-x-y;
13.            //判断总钱数是否符合条件
14.            if (5*x+3*y+z/3==100 && z%3==0)
15.            {
16.                cout << x << " " << y << " " << z << endl;
```



```
17.         }  
18.     }  
19. }  
20.     return 0;  
21. }
```

类似的不定方程问题, 都可以用循环来求解。为提高效率, 可以在程序中进行适当优化, 减少循环体的执行次数。