

可下载教学资料

<http://www.tup.tsinghua.edu.cn>

21世纪普通高校计算机公共课程规划教材

C++程序设计 实验指导与习题

孙仁科 主编

苏成 姜薇 聂茹 陈廷杰 编著

清华大学出版社



21 世纪普通高校计算机公共课程规划教材

C++程序设计 实验指导与习题

孙仁科 主编

苏成 姜薇 聂茹 陈廷杰 编著

清华大学出版社

内 容 简 介

本书是《C++程序设计教程》的配套实验教材,包括数据类型与表达式、条件与开关语句、循环语句、数组、指针与字符串、函数、作用域与生存期、类与对象、继承与派生、运算符重载、多态性、输入输出流和模板。

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++程序设计实验指导与习题 / 孙仁科主编. --北京:清华大学出版社, 2013

21 世纪普通高校计算机公共课程规划教材

ISBN 978-7-302-33571-9

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

中国版本图书馆 CIP 数据核字 (2013) 第 203873 号

责任编辑:刘向威

封面设计:何凤霞

责任校对:时翠兰

责任印制:刘海龙

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 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:14

字 数:340 千字

版 次:2013 年 10 月第 1 版

印 次:2013 年 10 月第 1 次印刷

印 数:1~2000

定 价:25.00 元

产品编号:055517-01

前 言

本书是与《C++程序设计教程》（苏成主编，清华大学出版社出版）配套的实验与习题指导用书。

程序设计是高校重要的计算机基础课程，它以编程语言为平台，介绍程序设计的思想和方法。通过该课程的学习，学生不仅要掌握高级程序设计语言的知识，更重要的是在实践中逐步掌握程序设计的思想和方法，培养问题求解和程序语言的应用能力。

C++程序设计是一门实践性很强的课程，该课程的学习有其自身的特点，学习者必须通过大量的编程训练和练习，在实践中掌握程序设计语言的知识，培养程序设计的基本能力，并逐步理解和掌握程序设计的思想和方法。因此，C++程序设计课程的教学重点应该是培养学生的实践编程能力，教材也要适应这种要求。

本书由实验指导、习题和综合测试 3 部分组成，实验部分有 14 个实验，每个实验通过精心设计的分析程序、完善程序、编写程序和进阶提高 4 个阶段，按照“模仿—改写—编写—提高”的上机实践过程，循序渐进地熟悉编程环境，理解和掌握程序设计的思想、方法和技巧，并掌握基本的程序调试方法；习题部分则包括与教材配套的选择题、填空题；综合测试部分精选了 6 套试卷对各章节知识进行综合测试。习题和综合测试都有参考答案，以帮助读者巩固各章节的知识点。

本书由孙仁科主编并统稿，苏成、姜薇、聂茹、陈廷杰参加编写。

由于作者水平有限，对书中存在的谬误之处，敬请读者指正。

编 者

2013 年 7 月

目 录

第一部分 实验指导.....	1
实验 1 熟悉 C++开发环境和编程步骤.....	4
实验 2 数据类型与表达式.....	5
实验 3 条件与开关语句.....	7
实验 4 循环语句.....	10
实验 5 数组.....	14
实验 6 指针与字符串.....	18
实验 7 函数.....	22
实验 8 作用域、生存期、多文件结构.....	31
实验 9 类与对象.....	36
实验 10 继承与派生.....	41
实验 11 运算符重载.....	46
实验 12 多态性.....	49
实验 13 输入输出流.....	52
实验 14 模板.....	55
第二部分 各章习题.....	60
第 1 章 简单的程序设计习题.....	60
第 2 章 构造数据类型习题.....	71
第 3 章 函数习题.....	87
第 4 章 类与对象习题.....	101
第 5 章 继承与派生习题.....	121
第 6 章 多态性习题.....	133
第三部分 综合测试试卷.....	145
模拟试卷一.....	145
模拟试卷二.....	150
模拟试卷三.....	155
模拟试卷四.....	164
模拟试卷五.....	171

模拟试卷六	177
-------------	-----

第四部分 各章习题及综合测试试卷参考答案	184
----------------------------	-----

第1章 简单的程序设计习题参考答案	184
第2章 构造数据类型习题参考答案	185
第3章 函数习题参考答案	187
第4章 类与对象习题参考答案	188
第5章 继承与派生习题参考答案	192
第6章 多态性习题参考答案	194
模拟试卷一参考答案	197
模拟试卷二参考答案	200
模拟试卷三参考答案	205
模拟试卷四参考答案	207
模拟试卷五参考答案	210
模拟试卷六参考答案	213

参考文献	216
------------	-----

1. C++上机实验的重要性

学习“C++程序设计”课程，一是要掌握 C++ 的语法，二是学会程序设计的方法。学习语法的目的是为编写合乎规范的程序。要真正学会程序设计实际问题，光靠“纸上谈兵”是不行的。很多同学有这样的体会，课本里的程序能看懂，但是一合上课本，自己编写就写不出来，即使写出来，也是错误百出。所以，上机实验在学习本课程中占有举足轻重的地位，一定要重视上机实践环节。从学时分配来说，上机实验的时间应不少于课堂学习的时间，为了弥补上机时间的不足，提倡增加课外上机实验。

通过上机实践，可以实现以下 3 个目标：

(1) 能够加深对课堂上讲授的 C++ 知识的理解。C++ 的很多语法规则枯燥、难记、难理解，比如，对于 `++i` 和 `i++` 的区别和用法，通过编程能深刻体会它们的含义。

(2) 可以熟悉一种编程的环境，掌握相关的开发工具的用法。C++ 语言的标准规范只有一个，但是 C++ 的编程环境却是多种多样的。除了常见的微软公司的 Visual C++ 6.0（简称 VC6）开发环境以外，还有其他编程环境，比如 Code:: Blocks、MinGW Developer Studio 等。这里之所以选择 VC6 主要是考虑它在 Windows 平台下应用广泛，而且也是全国计算机等级考试的编程环境。这些编程环境实际上都是将编辑、编译、连接、调试运行集成在一起的，使用非常方便。需要指出，由于不同的厂商对 C++ 语言的实现有差别，有些非标准的程序可能在不同的环境下运行会有不同的结果，或者在一种环境下能编译通过，而在另一种环境下编译出错。为减少这种情况的出现，不要使用 C++ 标准没有规定、而是依赖于具体的编译系统的规则，比如，`x=a*2+-a;` 语句就可能引起不同的解释。

(3) 掌握调试程序的方法和技巧，积累编程经验，提高动手编程的能力。要想成为一名合格的程序员，需要熟悉编译和连接中出现的错误信息的含义，学会用工具调试程序的方法，学会给程序准备合适的测试数据。此外，还要掌握编写多文件结构程序的方法，培养协作开发大中型程序的能力。

2. 上机实验的准备与安排

为了提高上机实验的效率，在每次实验之前，应该复习与实验相关的教学内容，将实验要求的程序事先准备好，不要等到上机时才动手写程序或者抄袭别人的程序。除了实验任务以外，对于课本中的例题、作业有疑问的地方，也可以上机运行测试，以巩固所学的知识。

本课程安排的实验总共有 14 个, 每个实验的学时数根据内容的多少不同, 一般 2~6 个学时。教师可根据总的实验学时数进行适当调整。每个实验的内容基本分成 4 部分, 由浅入深, 循序渐进。

(1) 分析程序。通过程序样例, 掌握 C++ 的基本语法, 思考问题。属于验证型实验。

(2) 完善程序。给出程序的框架和大部分代码, 对于一些重点的算法、语句进行补充完善。主要培养学生阅读和理解程序的能力。从验证型实验向设计型实验过渡。

(3) 编写程序。通过分析程序和完善程序的锻炼, 训练编写和调试基本的程序。属于设计型实验。

(4) 进阶提高。编程题目的难度和综合性进一步提高。属于设计型实验的提高。

四个部分对学生的要求逐步提高, 一般要求完成前 3 个部分的任务, 最后的进阶提高主要面向学有余力的学生。

3. 上机步骤与程序调试的方法

上机实验的步骤一般是: 编写程序、录入编辑、编译、连接、运行。这个过程往往要循环反复, 在编译、连接、运行的任何一个环节出现错误, 都要判断错误原因和错误位置, 修改程序, 重新编译、连接、运行。

下面介绍编程中常见的 3 类错误: 语法错误、逻辑错误和运行错误。

(1) 语法错误。指违背了 C++ 语言的规定, 主要是语句的结构或拼写中存在的错误。语法错误又分为两种, 第一种是严重的语法错误, 在编译时错误信息显示为 “error”, 无法生成目标代码 (.obj 文件)。出现 “error” 时, 必须修改源程序。第二种是 “轻微” 的错误, 在编译时的错误信息显示为 “warning”, 例如, 变量未初始化就使用, 双精度类型数据赋值给 float 型变量, 等等。出现 “warning” 错误时, 源程序不修改也能编译通过, 生成目标文件, 但程序员应该对这些警告信息高度重视, 因为它提示程序存在隐患, 很可能得到的运行结果不正确。

(2) 逻辑错误。由于程序员设计的算法有错或编写的程序有错, 通知给系统的指令与解题的原意不相同, 即出现了逻辑上的混乱。

与语法错误相比, 逻辑错误难于发现, 因为语法错误能够依靠编译系统来自动检查, 而逻辑错误主要从运行结果与预期不符发现错误, 然后靠程序员的知识、经验, 通过测试、调试程序, 最终确定错误的原因、位置, 然后进行修改。

(3) 运行错误。主要是由于缺少有关数据文件或者输入数据不合乎要求, 或者数组访问越界、指针使用问题、动态内存管理问题等。

经常遇到的错误是前两种错误。首先, 排除语法错误主要借助于编译器对出错信息的提示, 以及程序员的检查分析。但需要指出, 编译器提示的出错信息位置未必准确, 有可能是前面某个语句的错误导致其后的若干错误, 这需要程序调试经验的积累。其次, 出现逻辑错误往往不在 C++ 方面, 而是算法存在问题, 也就是编程的思路、逻辑出问题了, 这时需要耐心检查对问题的理解是否正确, 数学模型是否正确, 算法是否正确, 程序是否正确地描述了算法。比如, 希望判断 x 与 y 是否相等, 本来应该写成 “if($x=y$)”, 如果写成 “if($x=y$)”, 那么上述条件就跟预期完全不同了。

对于运行错误, 由于编译和连接都通过了, 单凭在系统运行时弹出的窗口错误信息,

难于确定错误的原因和位置。所以，发现运行错误主要靠编程经验。

4. 培养良好的编程风格与习惯

对于初学者来说，培养良好的编程风格与习惯非常重要。必须指出，很多初学者存在一个认识误区，认为只要编写的程序能得到正确的运行结果就万事大吉了。其实，得到正确的结果只是程序设计的基本要求。在编写简单的、面向过程的程序中，要树立模块化编程的思想；在编写复杂的、面向对象的程序中，要贯穿封装、继承和多态的思想。一个好的程序，应当在保证结果正确的前提下，满足以下要求：

(1) 程序便于阅读和理解。在编程初期就要养成良好的编程风格与习惯。比如，给标识符起有意义的名字；每行只写一句；采用缩进格式书写程序，使得程序结构分明；适当添加注释语句；少用全局变量；多用系统提供的标准函数。总之，不要为了追求代码简短，而使得程序晦涩难懂。

(2) 程序的执行效率高。运行时间短，占用的存储空间少。选择好的算法，是提高程序效率的关键。对于一些经典的问题，要把自己编写的程序与经典的程序进行比较，分析两者之间的差别，如何进行优化。比如，对于数组排序问题，在问题规模较小时，各种算法的优劣差不多，但是随着问题规模的增大，算法的性能差距就很明显，好的算法需要运行 1 分钟，差的算法可能需要几小时。

(3) 容错性强，健壮性高。往往会发生这种情况，尽管编写的程序正确，但是在运行时由于输入数据不合乎要求，而导致运行错误。比如，输入的变量值是 0，而它在程序中恰好是作为除数；再比如，求三角形面积，输入三条边长为 1、2、4。如果程序不能检测和提示这些非法数据，那么这个程序就比较“脆弱”了。

(4) 可移植性强，可扩展性强。一个调试好的程序移到另一个开发环境中（比如，从 VC6 到 GCC）能否正确运行？系统要添加功能，程序是否容易修改？

以上 4 个编程要求，对于初学者来说可能难以全部做到，但是至少应该要求自己做到第一条——编写容易理解的程序。

5. 实验报告与实验成果

写实验报告是上机实验的最后一个环节，它是对本次上机实验的分析与总结。实验报告主要包括实验目的与要求、实验内容与实验步骤、算法与流程图、程序清单、运行结果，以及编程过程中遇到的问题与解决方法、心得体会等。

对于实验报告，一定不要敷衍，不能抄袭，尤其要重视写好实验报告中调试过程和体会部分。每个人在调试程序过程中遇到的问题不同，解决的方法也不同，通过分析总结，使自己的某个模糊或者错误的概念得到澄清，某个算法得到优化，某个程序更加高效。总之，写实验报告对于积累编程经验和提高编程能力大有裨益。

实验 1 熟悉 C++ 开发环境和编程步骤

1.1 实验目的

- (1) 熟悉 Visual C++ 6.0 编程环境。
- (2) 了解 C++ 编程的步骤、C++ 程序的基本结构。

1.2 实验要求

- (1) 掌握在 Visual C++ 6.0 集成开发环境中编译、连接、调试与运行程序的方法。
- (2) 了解 C++ 源程序的特点。
- (3) 掌握一般的 C++ 程序的组成，学会使用简单的输入输出语句。

1.3 实验内容与步骤

关于 Visual C++ 6.0 编程环境的介绍可参考教材第 1 章 1.4 节，在此不多赘述。

1. 分析程序

程序 1-1 如下：

```
#include <iostream>           //A
using namespace std;          //B
int main()                     //C
{
    cout<<"我喜欢 C++ 程序设计！"<<endl;    //D
    return 0;                  //E
}
```

问题：

- ① 将程序 1-1 中的 A 行或 B 行删除（或者在该语句最前面添加//），观察编译结果。
将 A 行和 B 行替换为一行：`#include <iostream.h>`
观察编译结果。
- ② 将程序 1-1 中的西文分号改为中文的分号，编译提示什么错误信息？
将程序 1-1 中的西文双引号改为中文的双引号，编译提示什么错误信息？
- ③ 如果将 C 行的 `main` 误写成 `mian`，编译、连接能通过吗？
- ④ 如果将 E 行的 `"return 0;"` 删除，观察编译结果，有什么警告信息？

2. 完善程序

按照提示，将程序中下划线部分替换成适当的代码。

程序 1-2 如下：

```
#include<_____>
using _____
int main()
{
    _____//定义 3 个整型变量 x、y、z
```

```
____//输入 x、y、z 的值
____//输出这 3 个数的平均值
return 0;
}
```

3. 编写程序

(1) 编写一个程序，在第 1 行输出 10 个数字 0~9，在第 2 行输出 26 个大写英文字母 A~Z。

(2) 编写一个程序，针对你个人的真实情况，在屏幕上显示：

我的学号是：_____
我的姓名是：_____
我的专业是：_____
我的年龄是：_____

(3) 编写一个程序，输出如下的图形：

```
  #
 ###
#####
#####
```

(4) 编写一个程序，在屏幕上提示：“从键盘上输入两个整数（被加数 x 和加数 y），然后计算并输出 x+y 之和”。

实验 2 数据类型与表达式

2.1 实验目的

- (1) 掌握 C++ 语言数据类型。
- (2) 学会使用 C++ 的有关运算符、表达式。

2.2 实验要求

- (1) 熟悉基本数据类型变量的表示方法、范围、赋值方法。
- (2) 掌握常用运算符的优先级、结合性，特别注意 C++ 表达式与数学表达式写法的区别。
- (3) 理解自增、自减的前置、后置运算的特点。

2.3 实验内容与步骤

1. 分析程序

输入以下程序，编译、连接，分析运行结果。程序的功能如下：

- ① 测试一下当前机器上表示的各种数据类型占用的字节数是多少。
- ② 数据运算时，如果超出其表示范围，将发生溢出。
- ③ 了解常见的转义字符的作用。

④ 比较++i 与 i++的相同点与区别。

提醒：在输入程序时，为了节省时间，其中的注释部分可以省略。

程序 2-1 如下：

```
#include <iostream>
using namespace std;
int main()
{
    cout<<"sizeof(short int)="<<sizeof(short int)<<endl;
    cout<<"sizeof(int)="<<sizeof(int)<<endl;
    cout<<"sizeof(long int)="<<sizeof(long int)<<endl;
    cout<<"sizeof(char)="<<sizeof(char)<<endl;
    cout<<"sizeof(float)="<<sizeof(float)<<endl;
    cout<<"sizeof(float)="<<sizeof(double)<<endl;
    short int n=32767;
    n=n+3; //运算发生溢出
    cout<<"n="<<n<<endl; //运行结果与你预期的是否相同?
    char c1='a',c2=101;
    cout<<"c1="<<c1<<" c2="<<c2<<endl; //字符变量 c2 输出的是什么?
    cout<<'A'<<'\\t'<<'B'<<'\\n'<<'C'<<'\\x42'<<endl; //转义字符的输出
    int i,j;
    i=j=1;
    cout<<"i="<<++i<<" j="<<j++<<endl; //前置、后置运算对结果有什么影响?
    cout<<"i="<<i<<" j="<<j<<endl;
    return 0;
}
```

程序 2-1 运行结果如图 1 所示。

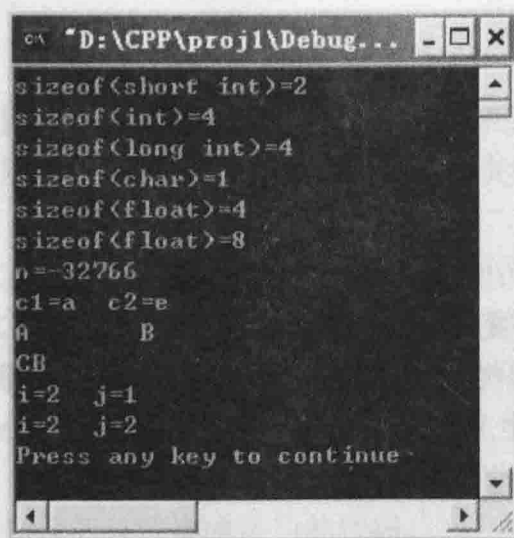


图 1 程序 2-1 运行结果

观察运行结果与预期的是否一致？如果不一致，找出预期错误的原因。

2. 完善程序

按照提示，将程序中下划线部分替换成适当的代码。

程序 2-2 如下:

```
#include<iostream>
//将数学函数库包含进来
using namespace std;
//定义圆周率常量 PI, 取值为 3.14159
int main()
{
    //定义双精度型变量 x 和 y
    //将 60°角转换成弧度赋给变量 x
    //将 x 的正弦值赋给变量 y
    //输出 y 和 y 的平方根
    //在电脑上响铃一次 (提示:输出转义字符)
    return 0;
}
```

3. 编写程序

- (1) 编写程序, 输入两个整数, 输出它们的商和余数。
- (2) 设计一个程序, 从键盘输入一个圆的半径, 求其周长和面积。
- (3) 编写一个程序, 功能是将输入的华氏温度转换成摄氏温度并输出。

温度转换的数学公式为

$$C=\frac{5}{9}(F-32)$$

问题: 在 C++中, 将上述公式是否可以写成 $C=5/9(F-32)$? 为什么?

当输入 $F=104$, 验证一下运行结果是否为 $C=40$ 。

- (4) 编写程序, 输入一个球的半径 r , 输出其表面积和体积。

提示: 球的表面积= $4\pi r^2$

球的体积= $4/3\pi r^3$

- (5) 编写程序, 从键盘输入一个小写字母, 将它转换成大写字母输出。

4. 进阶提高

- (1) 编写程序, 从键盘输入一个 3 位正整数, 将其逆序输出。例如: 输入 123, 输出 321。
- (2) 输入二整数 a 、 b , 实现二个数的交换 (不借助其他变量)。

实验 3 条件与开关语句

3.1 实验目的

- (1) 掌握条件与开关语句的用法。
- (2) 学会使用选择结构解决一般的实际问题。

3.2 实验要求

- (1) 熟练使用 if 语句、switch 语句的特点和用法。

(2) 掌握逻辑表达式的用法。

3.3 实验内容与步骤

1. 分析程序

以下程序的功能是将输入的百分制成绩转换成 5 分制成绩输出。分别使用 if 语句和 switch 语句来实现。

(1) 采用 if 语句

程序 3-1 如下:

```
#include <iostream>
using namespace std;
int main()
{
    int x;
    cout<<"请输入考试成绩: ";
    cin>>x;
    if(x>=90)
        cout<<"优秀"<<endl;
    else if(x>=80)
        cout<<"良好"<<endl;
    else if(x>=70)
        cout<<"中等"<<endl;
    else if(x>=60)
        cout<<"合格"<<endl;
    else
        cout<<"不及格"<<endl;
    return 0;
}
```

(2) 采用 switch 语句

程序 3-2 如下:

```
#include <iostream>
using namespace std;
int main()
{
    int x;
    cout<<"请输入考试成绩: ";
    cin>>x;
    switch(x/10)
    {
        case 10:
        case 9:
            cout<<"优秀"<<endl;
            break;
        case 8:
            cout<<"良好"<<endl;
            break;
    }
```

```

    case 7:
        cout<<"中等"<<endl;
        break;
    case 6:
        cout<<"合格"<<endl;
        break;
    default:
        cout<<"不及格"<<endl;
    }
    return 0;
}

```

思考:

- ① 程序 3-1 中, 能否调整各个 else if 语句的前后顺序?
- ② 程序 3-2 中, switch 语句中的 x 为什么要除以 10?
- ③ 对于程序 3-2, 如果输入成绩为 110, 运行结果怎样? 程序 3-2 应如何改进?

2. 完善程序

某商场购物打折促销。规则如下:

若每位顾客一次购物,

- ① 满 1000 元, 打九八折;
- ② 满 2000 元, 打九五折;
- ③ 满 3000 元, 打九折;
- ④ 满 4000 元, 打八八折;
- ⑤ 5000 元及以上, 打八五折。

程序 3-3 功能是输入购物款, 输出实收款。请按照题意, 将程序 3-3 中下划线部分替换成适当的代码。

程序 3-3 如下:

```

#include <iostream>
using namespace std;
int main()
{
    double y,s;    //y 是应收的购物款,s 是实际收的购物款
    cout<<"输入应收的购物款:";
    cin>>_____;
    if(y<1000)
        _____;
    else if (y<2000)
        _____;
    else if (y<3000)
        _____;
    else if (y<4000)
        _____;
    else if (y<5000)
        _____;
    else

```

```

    _____;
    cout<<"实收款: "<<_____<<endl;
    return 0;
}

```

3. 编写程序

(1) 有下列分段函数:

$$y = \begin{cases} x+1 & x < 0 \\ x^2 - 5 & 0 \leq x < 10 \\ x^3 & x \geq 10 \end{cases}$$

编写程序, 输入 x , 输出 y 的值。

(2) 从键盘上输入 3 个数, 判断能否构成三角形的 3 条边。输出判别结果。

(3) 编写程序, 输入一个整数 (在 0~6 之间, 其他数字则提示输入非法数据), 则输出它对应的星期几的英文名称, 例如, 0 对应 Sunday, 1 对应 Monday, ……

4. 进阶提高

(1) 编写一个求解一元二次方程的程序。方程: $ax^2+bx+c=0$, 根据输入的 a 、 b 、 c 的值, 输出该方程的解, 分为以下 4 种情形:

- ① 不是一元二次方程;
- ② 有 2 个不相等的实根;
- ③ 有 2 个相等的实根;
- ④ 无实根。

(2) 输入年份、月份、日期, 输出该日期是该年份的第几天。例如, 假设输入的年份、月份、日期分别为 2012, 5, 1, 则输出 122。

实验 4 循环语句

4.1 实验目的

- (1) 掌握 while、do-while、for 等 3 种语句的用法。
- (2) 掌握循环结构程序设计的方法。

4.2 实验要求

- (1) 掌握 3 种循环语句的特点、用法。
- (2) 掌握嵌套的循环程序设计。
- (3) 熟悉 break、continue 语句的区别与用法。
- (4) 掌握穷举法、迭代法、递推法等算法设计方法。

4.3 实验内容与步骤

1. 分析程序

(1) “百钱百鸡”问题。“今有鸡翁一，值钱伍；鸡母一，值钱三；鸡雏三，值钱一。百钱买鸡百只，问鸡翁、鸡母、鸡雏各几何？”

解：该问题可以用“穷举法”解决，就是利用列举出各个变量的可能取值，然后判断它们是否能构成一组解。

分析：假设用 x 、 y 和 z 分别表示鸡翁、鸡母、鸡雏，每种至少有一只。则有以下关系：

$$x + y + z = 100$$

$$5x + 3y + z/3 = 100$$

程序 4-1 如下：

```
#include <iostream>
using namespace std;
int main()
{
    int x,y,z;
    cout<<"鸡翁"<<"\t"<<"鸡母"<<"\t"<<"鸡雏"<<endl;
    for(x=1; x<=100; x++)
        for(y=1; y<=100; y++)
            for(z=1; z<=100; z++)
                if(x*5+y*3+z/3==100 && x+y+z==100 && z%3==0) //z%3==0 什么作用?
                    cout<<x<<"\t"<<y<<"\t"<<z<<endl;
    return 0;
}
```

程序 4-1 运行结果如图 2 所示。

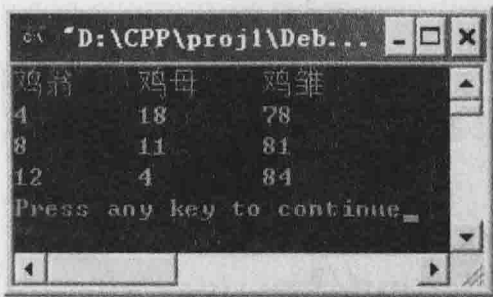


图 2 程序 4-1 运行结果

思考：

如果将程序中 if 语句改为：

```
If(x*5+y*3+z/3==100 && x+y+z==100)
```

那么运行结果会有什么变化？为什么？

对于上面的程序，可以进行优化。根据题意，可缩小 x 、 y 、 z 的范围，从而减少循环次数。首先，100 元买公鸡不超过 20 只，母鸡不超过 33 只；其次，最内层的循环可以取消，也就是说，公鸡、母鸡数量确定了，小鸡个数也就确定了。这样，循环次数从 100^3 大