



面向对象程序设计

习题解答与上机实验

刘正林 周纯杰 刘烨 编著

- 配套教材的习题详解
- 丰富的范例程序
- 可视化编程技术的精粹
- 精心设计的大型综合上机练习



华中科技大学出版社

<http://press.hust.edu.cn>

TP312-44

27

面向对象程序设计 习题解答与上机实验

刘正林 周纯杰 刘 烨 编著

华中科技大学出版社

中国·武汉

图书在版编目(CIP)数据

面向对象程序设计习题解答与上机实验/刘正林 周纯杰 刘 烨 编著
武汉:华中科技大学出版社, 2002年4月
ISBN 7-5609-2680-0

I. 面…

Ⅰ. ①刘… ②周… ③刘…

Ⅱ. 面向对象-C语言-高等学校-教学参考资料

Ⅳ. TP312

面向对象程序设计习题解答与上机实验 刘正林 周纯杰 刘 烨 编著

责任编辑:文 林 周 筠

封面设计:潘 群

责任校对:张兴田

责任监印:张正林

出版发行:华中科技大学出版社

武昌喻家山 邮编:430074 电话:(027)87545012

录 排:华中科技大学惠友科技文印中心

印 刷:湖北省通山县印刷厂

开本:787×1092 1/16

印张:21.25

字数:500 000

版次:2002年4月第1版

印次:2002年4月第1次印刷

印数:1—4 000

ISBN 7-5609-2680-0/TP·463

定价:26.00元

(本书若有印装质量问题,请向出版社发行部调换)

内 容 简 介

本书是《面向对象程序设计》教材的配套书，不仅给出了《面向对象程序设计》教材的各章习题的详细解答，并通过剖析习题内容归纳总结编程要点，书中所给出的每个程序都有详细的注释，便于阅读和理解，并且还给出精心设计并开发成功的 8 个大型综合上机练习和可视化编程的例程，所涉及的知识不仅包括面向对象程序设计的数据封装和信息隐藏、类的继承和多态性，以及模板技术的应用等，而且还包括 Windows 编程基础知识、应用 Windows API 函数的可视化编程技术要点以及应用 MFC 进行可视化编程的思想方法等。其特点是通过实用化例程让读者全面、灵活、系统地掌握面向对象程序设计的技术思路 and 开发方法，并用“深入浅出，通俗易懂”的方式讲解令人眼花缭乱的可视化编程技术。

本书适用于大专院校理、工科各类专业本科生、研究生以及计算机应用软件开发人员，在内容安排上有深有浅，且侧重应用，适用于各个层次的读者，既适合以前从未接触过 C++ 和可视化编程的初学者，也适合具有一定编程能力的读者学习面向对象程序设计方法和可视化编程基础知识。

前 言

“面向对象程序设计”是一门实践性非常强的课程，若要真正掌握面向对象编程技术，使编程能力有实质性的提高，必须在认真听课并读懂教材内容的基础上，通过上机实验加强开发软件的基本技能训练。只有勤学苦练才能积累宝贵的编程经验，悟出面向对象编程技术的要领，牢固地掌握像 Visual C++ 这样的优秀应用程序开发工具，快速冲向计算机应用领域的前沿。

在学习这门课程时，很多学生都提出这样的困惑：课堂和教材的内容基本上都能接受和理解，但真要自己实际动手编程却又感到脑袋一片空白而无从下手；其二，即便学完了本课程的所有内容，对涉及有 Windows API 和 MFC 内容的应用程序，还是感到眼花缭乱而无从下手。究竟应该如何进一步学习可视化编程技术，攀登上面向对象程序设计方法的这一技术高峰，成为名副其实的 OOP 开发者呢？

本书是《面向对象程序设计》教材的配套书，既为初学者培养编程能力、加强开发软件的基本技能训练提供必要的辅导材料，也为那些想大幅度提高编程能力、涉足可视化编程技术的人员提供便于学习和掌握的实用化编程素材。全书分为 3 大部分，前两部分给出《面向对象程序设计》教材的各章习题的详细解答。其详细的含义是不仅简单地给出该题的正确答案，而且还通过剖析习题内容，归纳总结技术要点，使学生知其然且知其所以然，并引导学生举一反三，扎实、灵活和系统地掌握面向对象编程要点。书中所给出的每一个程序都有详细的注释，便于学生阅读和透彻理解编程思路，并且都在 Visual C++ V6.0 集成开发环境中完成了调试且能成功地运行。众所周知，可视化编程是 C++ 应用程序发展的必然趋势，它的触角几乎已触及到计算机应用的各个领域，业已成为软件应用领域的一大景观，应该说 Visual C++ 是优秀的可视化编程开发工具，不涉足可视化内容就未触及到它的灵魂。为此，本书的第三部分为精心设计并开发成功的 8 个大型综合上机练习和可视化编程的例程，所涉及的知识不仅包括面向对象程序设计的数据封装和信息隐藏、类的继承和多态性以及模板技术的应用等，而且还包括 Windows 编程基础知识、应用 Windows API 函数的可视化编程技术要点以及应用 MFC 进行可视化编程的思想方法等。其特点是通过实用化例程让读者全面、灵活、系统地掌握面向对象程序设计的技术思路 and 开发方法，并用“深入浅出，通俗易懂”的方式讲解可视化编程技术，在教给学生“面包加糖果”之类知识的同时，更重要的是要让他们掌握“猎枪和钓鱼杆”这类工具的使用方法，学会了“猎枪和钓鱼杆”这类生存工具的使用方法，学生就有了独立探索新知识的本领，这才是事半功倍的学习方法。

本书适用于大专院校理、工科各类专业本科生、研究生以及计算机应用软件开发人员，在内容安排上有深有浅，且侧重应用，适用于各个层次的读者，既适合以前从未接触过 C++

和可视化编程的初学者，也适合具有一定编程能力的读者学习面向对象程序设计方法和可视化编程基础知识。

全书的习题和例程都录入配套光盘。衷心感谢读者选用本书，欢迎批评指正！

刘正林 周纯杰 刘 烨

2001 年国庆节于武汉

目 录

第一部分 C++模块化语言基础

第1章 概 论	习题 1	(1)
第2章 数据类型、运算符和表达式	习题 2	(10)
第3章 语句和流程控制	习题 3	(24)
第4章 数组与指针	习题 4	(37)
第5章 函数	习题 5	(47)
第6章 复杂的数据结构和函数调用	习题 6	(83)
第7章 结构体	习题 7	(102)
第8章 标准函数	习题 8	(133)

第二部分 面向对象程序设计

第9章 面向对象程序设计概论	习题 9	(155)
第10章 从C快速过渡到C++	习题 10	(162)
第11章 类和对象	习题 11	(175)
第12章 派生类、基类和继承性	习题 12	(203)
第13章 多态性和虚函数	习题 13	(220)
第14章 C++的流库	习题 14	(233)

第三部分 综合上机和可视化编程

习 题 15 (综合上机练习)	(242)
一、用 new 运算符创建多维的动态对象数组	(242)
二、String 类的定义和使用	(244)
三、封装良好的快速排序程序	(255)
四、安全可靠的快速排序程序	(258)
五、遍历二叉树	(261)
六、用模板堆栈实现表达式求值	(268)
习 题 16 (可视化编程)	(281)
一、在 Visual C++集成开发环境上, 用 Windows API 实现绘图功能	(282)
二、在 Visual C++集成开发环境上, 用 MFC 实现绘图功能	(311)
参考文献	(332)

第一部分 C++模块化语言基础

第1章 概论

习 题 1

一、选择填空

1. 下列各种高级语言中, (D) 是面向对象的程序设计语言。

- A. BASIC B. PASCAL C. C++ D. JAVA

解 只有 JAVA 才是完全面向对象的程序设计语言, C++是既可以做面向过程的模块化程序设计, 又支持面向对象程序设计的混合型语言。美国 Microsoft 公司把模块化、可视化和面向对象程序设计等现代软件新技术融入到 BASIC 中, 将它改造成 Visual BASIC, 还有美国 Borland 公司将面向对象思想引入到 PASCAL, 推出了面向对象开发软件包 Delphi PASCAL, 但严格来讲, 它们都不是面向对象的程序设计语言。

2. 关于 C++和 C 语言的关系描述中, (D) 是错误的。

- A. C 语言是 C++的一个子集 B. C 语言和 C++是兼容的
C. C++对 C 语言进行了一些改进 D. C 语言和 C++都是面向对象的

解 参见教材的图 1.2, C 语言蕴涵在 C++中, 是它的一个子集, 即它的模块化语言基础的内核所在, 换句话说, C++是 C 语言的超集。在符合最新 ANSI C++标准的前提下, C 语言和 C++是兼容的, 具体来说, 在 Visual C++集成开发环境下, 在以 .cpp 为扩展名的源文件中, 既可以用模块化的 C 语言 (不符合最新 ANSI C++标准的 C 语言老版本除外) 格式编程, 也可以采用面向对象风格的 C++格式编程。C++还对 C 语言的一些不够完善的地方进行了修改, 例如, 引入了“引用变量”用以取代指针变量。

3.

```
#define PI 3.141593
#include <stdio.h>
void main(void)
{   printf("PI = %f\n", PI);   }
```

该程序的输出结果为 (B)。

- A. 3.141593 = 3.141593 B. PI = 3.141593
C. 3.141593 = PI D. 程序有错误, 无输出结果

二、问答题

1. 请列举 C 语言的主要特点。

答 C 语言的主要特点如下。(1) 介于高级语言 (PASCAL) 和汇编语言之间, 兼有两者的优点。既适合于开发系统软件, 又适合于编写应用程序, 虽然 C 语言最初设计时, 是作为一种面向系统软件 (OS 和语言处理系统) 的开发语言, 即用来代替汇编语言的, 但是由于它强大的生命力, 以致足以取代汇编语言来编写各种系统软件和应用软件, 在事务处理、科学计算、工业控制和数据库技术等各个方面都得到了广泛应用。即便进入到以计算机网络为核心的信息时代, C 语言仍然是作为通用的汇编语言使用, 由于它的开放性和兼容性, 可做到与硬件平台无关。(2) 引用模块化程序结构, 便于软件工程化。现在的计算机软件系统, 即便是应用软件, 常常不可能是一个人的劳动结晶, 而是一班人、一个梯队、多少“人年”完成的。因此, 软件必须走“工程化”的道路。即采用工程的概念、原理、技术和方法来开发和维护软件。模块化程序设计仍然是当前软件工程最基本、最普遍采用的设计方法, 自顶向下划分模块, 直到最底层的每个模块都是完成单一独立的功能为止。C 和 C++ 是以函数模块为单位来思考问题的, 每个模块有特定的目的和功能, 一个 C 和 C++ 程序只不过是把这些模块装配起来以实现编程者所要求的全部任务。(3) 语言简洁, 且表达能力强, 使用灵活, 易于学习和应用。(4) C 和 C++ 的整个源程序是一些函数的集合体, 便于分块编译。软件工程所采用的方法是将一个复杂的问题分解成许多简单的子问题, 然后再解决各个子问题。它的基本思想是把软件设计成由相对独立、功能单一的若干个模块组成, 又称模块化程序设计。而 C 和 C++ 的源程序可以划分成一个以上的源文件, 在编译该程序时, 可以按一个个源文件为单位对它们分别单独地进行编译, 然后通过系统所配有的链接程序 (LINK) 将它们连接、装配在一起, 成为一个可执行文件, 分块编译把整块程序分割成若干个相对独立的功能模块, 自顶向下得到最底层一系列模块后, 即可分发给软件开发小组的各成员在多台计算机上独立地、并行地同时进行开发各个模块, 最后再汇总进行整体调试, 这种由众人同时进行集体性开发的软件工程技术方法加快了软件开发速度, 大幅度地缩短了开发周期。(5) 配备有“预处理操作命令” (Preprocessor Directives), 增强了用户所编程序的灵活性和适应性。

2. C++ 和 C 语言的关系如何? 为什么说 C++ 是 C 语言的一个超集?

答 C 语言蕴涵在 C++ 中是它的一个子集, 即它的模块化语言基础的内核所在, 换句话说, C++ 是 C 语言的超集, 它保留了 C 的所有组成部分并与其完全兼容, 而只是增加了面向对象的编程机制, 可以说它是 C 语言的面向对象扩展, 既可以做传统的结构化程序设计, 又能进行面向对象程序设计, 是面向对象程序设计的第 1 个大众化版本, 是当前学习面向对象程序设计方法的首选语言。在系统软件的开发研究上, C++ 的运行效率与 C 相比毫不逊色, 在大型应用软件开发上, 以 Windows 开发环境为操作系统平台的 C++ 标准类库和组件正在迅猛发展, C++ 即将取代 C 已是不可抗拒的事实, 它的触角几乎已触及到计算机研究和应用的各个领域。

3. 何谓“分块编译”和“一个编译单位”?

答 C 和 C++ 的源程序可以划分成一个以上的源文件, 在编译该程序时, 可以按一个个源文件为单位对它们分别单独地进行编译, 这些单独进行编译的每个源文件称为一个“编

译单位”。然后通过系统所配有的链接程序(LINK)将它们连接、装配在一起,成为一个可执行文件,即“文件名.EXE”。这样的编译过程称为“分块编译”。这就是通常所说的“多文件程序”,一个源程序由多个用C和C++编写的程序代码文件组成,这些文件称为“源文件”,仍然使用“.C”或“.CPP”的扩展名,但每一个源文件不是一个完整的源程序,必须与其他所有的源文件组合在一起。

4. 何谓“标准函数库”和“头文件”?

答 C语言与其他高级语言不同,本身不具备输入/输出操作(Input/Output操作,简称I/O操作)语句,这些功能是由执行输入/输出操作的标准函数来实现的。由C语言系统所提供的这些标准函数的集合体称为“标准函数库”。标准函数库不仅配备了各种I/O操作的标准函数,而且还备有各种常用的数学函数、字符和字符串处理函数、内存管理和进程管理函数、屏幕绘图函数、时间和日期函数以及系统有关的函数等。这些标准函数的说明以及它们所使用的符号常量、宏指令的定义等语句都包含在扩展名为“.h”的相应文件中,因而在使用某标准函数时,必须用#include语句嵌入相应的头文件。例如,在源程序中若使用了printf()、scanf()等标准函数,则必须要写上:

```
#include <stdio.h>
```

5. 请说明#define和#include两个预处理命令的主要作用?

答 (1) #define语句又称宏定义语句,其格式为:

```
#define 字符串1 字符串2
```

它是把字符串1定义成字符串2,在预处理操作时,系统把程序中出现的字符串1一律用字符串2替换。其主要作用如下。

① 将一个常量值用编程者所启用的名字(通常用英文大写字母组成的字符串)来代替,该常量称为“符号常量”。例如:

```
#define NULL 0
#define EOF -1
#define PI 3.1415926
```

② 定义宏指令(简称“宏”)和带参数的宏指令。例如:

```
#define MAX(x, y) (((x) > (y)) ? (x) : (y))
```

求x和y中较大的一个,若程序中写有:

```
int a = 6, b = -2, c;
c = MAX(a, b);
```

在编译前的预处理操作时,不仅把字符串1“MAX(x, y)”替换成字符串2“(((x) > (y)) ? (x) : (y))”,且还把实参a和b分别赋给形参x和y,则上面的赋值语句变成:

```
c = (((a) > (b)) ? (a) : (b));
```

赋值表达式的右值为变量a和b中的较大者a,其值为6赋给变量c。

(2) #include语句是指一个源文件可以把另一个源文件的全部程序代码内容都包含进来,通常是将一个已开发完成的源文件包含到编程者正在开发的源文件之中。其格式为:

```
#include <文件名> (1) 或 #include "文件名" (2)
```

特别在使用某标准函数时，必须用#include 语句嵌入相应的头文件，这时应使用格式(1)。例如，

在源程序中若使用了 strcmp()、strcpy()等标准函数，则必须要写上：

```
#include <string.h>
```

在源程序中若使用了 sin()、cos()、tan()、asin()、acos()、exp()、log()等初等数学函数，则必须要写上：

```
#include <math.h>
```

如果编程者自行编写的一个头文件如 menu.h ，则通常采用格式(2)，即：

```
#include "menu.h"
```

6. 请说明如何编译、链接和运行单文件和多文件源程序？

答 (1) 用例 1.1 源程序来说明单文件的编译、链接和运行过程。若该程序已经显示在 Visual C++集成开发环境的编辑窗口上，如图 1.1 所示，则用鼠标沿着“Build (主菜单)/Compile p11.cpp (子菜单) Ctrl+F7 (热键)”的操作路径点击，这时编译系统对当前源程序 p11.cpp 进行编译，并把编译过程中所发现的编译错误 (Compiler Error) 显示在屏幕下方的 Build 窗口上，并指出该错误所在的行号、编译错误码和错误性质的简洁提示，本系统的编译错误码采用“C××××”的格式表示，其中×代表一位十进制数码。编程者可根据这些错误信息来修改源程序，若还需进一步了解错误信息和改错的对策，可沿着“Help (主菜单)/Contents (子菜单) 或 Search (子菜单)”操作路径用鼠标点击，启动 MSDN (Microsoft Developer Network) 库，则弹出“MSDN Library Visual Studio 6.0”窗口，简称 MSDN 窗口，在该窗口的“Multiple Page”控件上点击“搜索”页面，在该页面的“输入要查找的单词(F):”框内键入编译错误码，例如 C2660 (CR)。往往一个错误会引起多行错误信息，因此一般

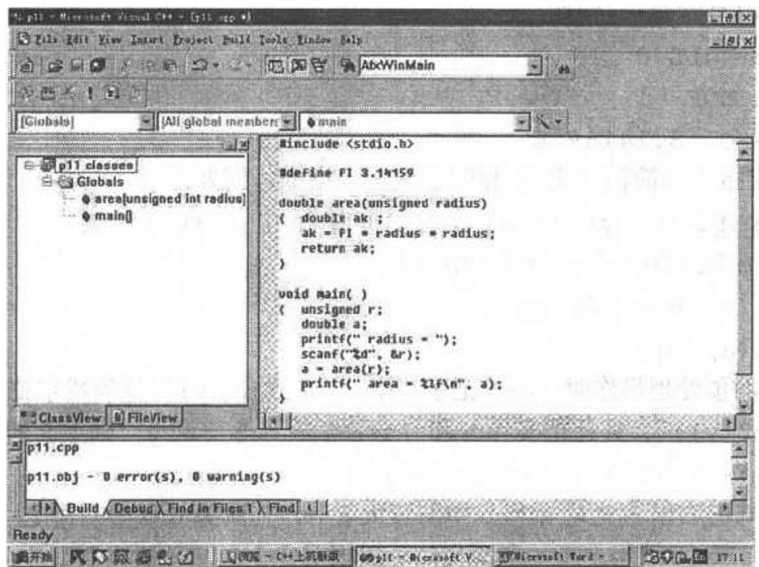


图 1.1 单个文件源程序的编译、链接和运行

是针对第 1 行错误来修改源程序再重新编译, 如果原来第 1 行错误消除了, 再针对新的第 1 行错误来修改, 再重新编译, 直到没有编译错误为止。如果没有编译错误, 则在 Build 窗口上, 将出现如下信息:

```
-----Configuration: p11 - Win32 Debug-----  
Compiling...
```

```
p11.cpp  
p11.obj - 0 error(s), 0 warning(s)
```

这说明编译成功, 并生成可重定位文件 p11.obj。接着进行链接操作, 操作方法是沿着“Build (主菜单) / Buildp11.cpp (子菜单) 或 F7 (热键)”的操作路径用鼠标点击, 链接过程中的错误处理与编译时类同, 仅连接错误码采用“LNK××××”的格式表示, 例如, LNK2660。链接成功后, 在 Build 窗口上将出现如下信息, 则说明已生成了可执行文件 p11.exe。

```
-----Configuration: p11 - Win32 Debug-----  
  
Linking...  
p11.exe - 0 error(s), 0 warning(s)
```

运行可执行文件的方法是沿“Build (主菜单) / Execute p11.exe (子菜单)”的操作路径用鼠标点击, 则执行 p11.exe, 并将弹出一个显示输出结果的窗口如图 1.2 所示。

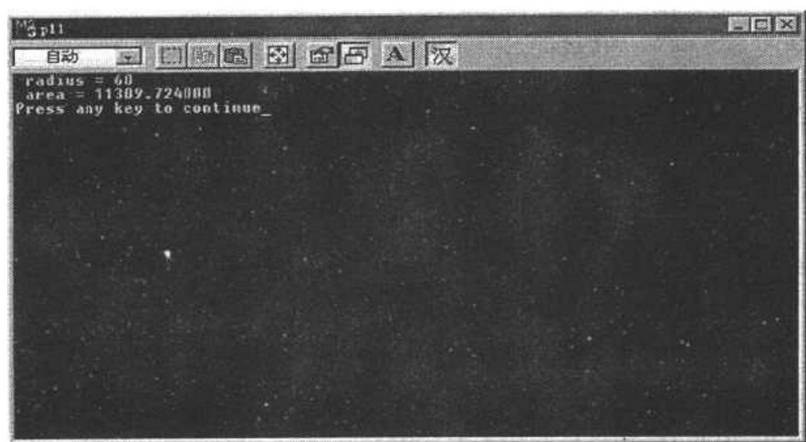


图 1.2 输出结果窗口

当执行到“printf(“ radius = ”);”语句时, 则该窗口显示出“ radius = ”提示信息, 操作者接着键入一个半径值为 60, 通过“scanf(“%d”, &r);”语句由变量 r 接收, 随后以 r 为实参调用 area() 函数, 将其返回值存放在 double 型变量 a 中, 最后执行“printf(“ area=%lf\n”, a);”语句, 在输出结果窗口上显示出计算结果值“ area = 11309.724000”。

(2) 对多文件源程序应采用建立工程项目的方法, 例如有如下 3 个源文件:

```
#include "Area.h" //Circle.cpp 计算圆面积函数 circle( ) 的定义
const double pi = 3.1415926;
double circle(double radius)
{ return pi * radius * radius; }
```

```
#include "Area.h" //Rect.cpp 计算矩形面积函数 rect( ) 的定义
double rect(double width, double length)
{ return width * length; }
```

```
#include <stdio.h> //CalculateArea.cpp
#include "Area.h"      调用不同功能函数计算面积
void main( )
{ double w , l;
  printf("Please enter a width : ");
  scanf("%lf", &w);
  printf("Please enter a length : ");
  scanf("%lf", &l);
  printf("Area of the rectangle is %lf\n", rect(w, l));

  double r;
  printf("Please enter a radius : ");
  scanf("%lf", &r);
  printf("Area of the circle is %lf\n", circle(r));
}
```

并

自行定义一个头文件 Area.h 包含各源文件内所定义的外部函数的声明语句，即：

```
// Area.h
double circle(double radius);
double rect(double width, double length);
```

然后，在凡是要用到这些外部函数的源文件开头处写上#include "Area.h"语句（源文件中划有虚线的语句），就可以调用这些函数。

建立工程项目的操作方法是：按照“File(主菜单)/New(子菜单)”的操作路径进入到 New 对话框，在 New 对话框内，再选择多页面控件中的 Project 页面，接着选择“Win32 Console Application”子菜单项，这时项目（Project）的目标平台选择框内将会出现：“Win32”。然后，在“Project name:”框内输入工程文件名为 CalculateArea，在“Location:”框内会自动出现指定路径名为 d:\test\CalculateArea，建立工程文件所需要的所有源文件应该放在该路径名下。例如，Area.cpp、Circle.cpp、Rect.cpp 和 Area.h 等文件都在 d:\test\CalculateArea 路径下，则应输入的路径名为：d:\test，在“Location :”框内将会出现“d:\test\CalculateArea”。也

可用鼠标点击“Location :”框右边的浏览键，在“Choose Directory”对话框内选取所需要的路径。再用鼠标点击 New 对话框内的 OK 键，则将弹出“Win32 Console Application - Step 1 of 1”选择框，可选择“An empty project”项后，点击 finish (完成) 键，则弹出“New Project Information”窗口，其内将出现如下信息：

- + Empty console application.

- + No files will be created or added to the project.

用鼠标点击该窗口内的 OK 键，则建立了名字为 CalculateArea 的工程文件项目。接着向工程文件项目中添加源文件，按照“Project(主菜单)/Add To Project(1 级子菜单)/New(2 级子菜单)”的操作路径进入到“New”对话框，在其内再选择多页面控件中的 Files 页面，接着选择 C++ Source File 项，并在右边 File 框内填上源文件名 Area (默认扩展名为.cpp，可以写也可以不写，.c 的扩展名必须写)，随后按回车键或用鼠标点击 OK 按钮，即将新建的源文件 Area.cpp 添加到 CalculateArea 的工程项目中，接着进入到编辑窗口，操作者用键盘敲入该文件的内容并保存，才完成了添加一个源文件的操作。然后再用同样的方法把 Circle.cpp、Rect.cpp 和 Area.h 等文件添加到 CalculateArea 的工程项目中，而在添加头文件 Area.h 时，应该选择 New 对话框 Files 页面中的 C/C++ Header File 项，则默认扩展名即为.h。

如果要添加的文件已经存在，则按照“Project(主菜单)/Add To Project(1 级子菜单)/Files(2 级子菜单)”的操作路径进入到“Insert Files into Project”对话框如图 1.3 所示，其下方的“insert into:”应出现工程项目名 CalculateArea。



图 1.3 将已存在的文件 Area.cpp 添加到 CalculateArea 中

在该对话框中用鼠标点击的方法从指定的目录（如图 1.3 中的 d:\test\CalculateArea）下选取所要添加到该工程文件项目的文件（如图 1.3 中的 Area.cpp，添加文件可以是任何盘任意目录下已存在的源文件 filename.c、filename.cpp 或目标文件 filename.obj 等，通常，为了便于管理和移植，应把它们移到工程项目的目录路径下，如 d:\test\CalculateArea），选中后双击该文件名，则将所选中的文件添加到项目 CalculateArea 中。

编译、链接和运行工程文件项目时，按照 Build(主菜单)/Rebuild All(子菜单)的操作路径用鼠标点击，则对项目中所包含的所有文件进行编译、链接。若有错误将显示

在错误信息窗口上,便于编程者修改程序时阅读。直到没有错误信息为止,这时便生成了一个可执行文件 CalculateArea.exe,其文件名就是项目名。运行工程文件的方法与单个源程序完全相同,即沿“Build(主菜单)/Execute p11.exe(子菜单)”的操作路径用鼠标点击,则执行 CalculateArea.exe,其输出结果为:

```
Please enter a width : 64.0 (CR)
Please enter a length : 32.0 (CR)
Area of the rectangle is 2048.000000
Please enter a radius : 72.0 (CR)
Area of the circle is 16286.016038
```

三、编写程序

1. 编写一个完整的可运行源程序,用人机对话方式从键盘输入 a、b、c、d 四个整数,计算表达式 $(a + b - c) * d$ 的值,并显示计算结果。

解 任何源程序都必须有一个而且只能有一个 main() 函数,它是程序的执行起点。由于需要调用标准库函数 printf() 和 scanf(),因此,应在源程序的开头处写上“#include <stdio.h>”,且 printf() 函数的 format 部分,由一对双引号包围的字符串既可以含英文字符,在中文 Windows 操作系统平台上也可以包含汉字字符,为了获得良好的人机界面,对每个变量的输入,用 printf() 函数输出汉字提示信息。所编写的源程序代码如下:

```
#include <stdio.h>
void main( )
{ int a, b, c, d;
  printf("请输入整型变量 a 的值 : ");
  scanf("%d", &a);
  printf("请输入整型变量 b 的值 : ");
  scanf("%d", &b);
  printf("请输入整型变量 c 的值 : ");
  scanf("%d", &c);
  printf("请输入整型变量 d 的值 : ");
  scanf("%d", &d);
  printf("表达式\"(a + b - c) * d\"的计算结果为%d\n",
        (a + b - c) * d);
}
```

该程序的运行结果为:

```
请输入整型变量 a 的值 : 86 (CR)
请输入整型变量 b 的值 : 24 (CR)
请输入整型变量 c 的值 : 7 (CR)
请输入整型变量 d 的值 : 6 (CR)
表达式"(a + b - c) * d"的计算结果为 618
```

2. 由初等几何学可知, 三角形的三个边为 a , b 和 c , 则其面积为:

$$\text{area} = (s * (s - a) * (s - b) * (s - c))^{1/2}$$

其中 $s = 1/2(a + b + c)$. 请定义两个带参数的宏指令, 一个用来求 s , 另一个用来求 area . 在程序中用带参数的宏名来求面积 area , 编写成一个完整的可运行源程序.

解 所编写的源程序代码如下:

```
#include <stdio.h>
#include <math.h>
#define s(x, y, z)      (x + y + z) / 2
#define area(s, x, y, z) \
    sqrt(s * (s - x) * (s - y) * (s - z))
void main( )
{ double a, b, c;
  double half, area;
  printf("请输入三角形 a 边的值 : ");
  scanf("%lf", &a);
  printf("请输入三角形 b 边的值 : ");
  scanf("%lf", &b);
  printf("请输入三角形 c 边的值 : ");
  scanf("%lf", &c);
  half = s(a, b, c);
  area = area(half, a, b, c);
  printf("三角形的三个边分别为 a = %.2lf, b = %.2lf 和 c = %.2lf, \
    \n  则其面积为: %.2lf\n", a, b, c, area);
}
```

该程序的运行结果为:

请输入三角形 a 边的值 : 32.64 (CR)

请输入三角形 b 边的值 : 46.8 (CR)

请输入三角形 c 边的值 : 52.7 (CR)

三角形的三个边分别为 a = 32.64, b = 46.80 和 c = 52.70,

则其面积为: 754.36

其中, 定义了两个带参数的宏指令 $s(x, y, z)$ 和 $\text{area}(s, x, y, z)$, 前者计算三角形周长的一半, 程序将其结果值存放在 `double` 型变量 `half` 中, 后者计算三角形的面积, 程序把它的结果值存放在 `double` 型变量 `area` 中, 由于面积计算公式含有开平方运算, 则调用标准库函数 `sqrt()`, 其函数原型在 `math.h` 头文件中, 即:

```
double sqrt( double x );
```

另外, 输入/输出 `double` 型数据时, 应在转换字符 `f` 前加修饰符 `l` 或 `L`, 最后一条 `printf()` 语句中, 还用 `"%.2lf"` 指定了输出显示的精度域宽为小数点后两位.

第2章 数据类型、运算符和表达式

习 题 2

一、选择题

下列各题 A、B、C、D 四个选项中，只有一个选项是正确的，请将正确的选项选择出来。

1. 不属于字符型常量的是(B)。

[1] A. 'A' B. "B" C. '\n' D. '\172'

解 字符串常量"B"不同于字符型常量'A'、'\n'和'\172'，它包含字符B和字符串结尾符'\0'，而字符型常量是该字符的ASCII代码值，如'A'、'\n'和'\172'的值分别为65、13和122（\172为8进制）等。

2. 属于整型常量的是(A)。

[2] A. 12 B. 12.0 C. -12.0 D. 10E10

3. 属于实型常量的是(D)。

[3] A. 'A' B. "120" C. 120 D. 1E-1

4. '\72'在内存中占(D)个字节，"\72"在内存中占(C)个字节。

[4] A. 4 B. 3 C. 2 D. 1

[5] A. 4 B. 3 C. 2 D. 1

解 '\72'是一个字符型常量，\72为8进制，是冒号': '的ASCII代码值，存放在内存中只占用1个字节。"\72"是一个字符串常量，包含冒号': '和字符串结尾符'\0'，所以在内存中占2个字节。

5. char 型常量在内存中存入的是(A)。

[6] A. ASCII 代码值 B. BCD 代码值
C. 内码值 D. 十进制代码值

6. 已知字符'A'的ASCII代码值是65，字符变量c1的值是'A'，c2的值是'D'。执行语句“printf("%d,%d", c1, c2 - 2);”后，输出的结果是(D)。

[7] A. A, B B. A, 68 C. 65, 68 D. 65, 66

解 在C或C++中，一个数值在0~255之间的正整数可以看成是一个字符常量，对应的字符就是该正整数作为ASCII代码值所对应的那个字符，如正整数65是字符'A'的