

第 1 周复习

至此，读者完成了学习使用 C++ 进行编程的第一周课程，已经能够熟练地输入程序以及使用编辑器和编译器。有了一定的使用 C++ 经验后，来看一个更复杂的程序。下面的程序结合使用了前 7 章介绍的很多主题。

在程序清单 R1.1 的后面对其进行了分析，该程序涉及的每个主题都在前 1 周的课程中介绍过。在第 2 周和第 3 周课程的后面，也有类似的复习。

程序清单 R1.1 第 1 周课程复习

```
CH02
1: /* Listing: WR01.cpp
2: * Description: Week in Review listing for week 1
3: *-----*/

CH01
4: #include <iostream>

CH02
5: using namespace std;
6:

CH03
7: enum CHOICE {
8:     DrawRect = 1,
9:     GetArea,
10:    GetPerim,
11:    ChangeDimensions,
12:    Quit = 0;
13:

CH02
14: // Rectangle class declaration

CH06
15: class Rectangle
16: {

CH06
17: public:
18:     // constructors
19:     Rectangle(int width, int height);
20:     ~Rectangle();
21:
```

```

22:     // accessors
23:     int GetHeight() const { return itsHeight; }
24:     int GetWidth() const { return itsWidth; }
25:     int GetArea() const { return itsHeight * itsWidth; }
26:     int GetPerim() const { return 2*itsHeight + 2*itsWidth; }
27:     void SetSize(int newWidth, int newHeight);
28:
29:     // Misc. methods
30:

```

CH06

31: private:

CH03

```

32:     int itsWidth;
33:     int itsHeight;
34: };
35:
36: // Class method implementations
37: void Rectangle::SetSize(int newWidth, int newHeight)
38: {
39:     itsWidth = newWidth;
40:     itsHeight = newHeight;
41: }
42:
43: Rectangle::Rectangle(int width, int height)
44: {
45:     itsWidth = width;
46:     itsHeight = height;
47: }
48:

```

CH06

```

49: Rectangle::~Rectangle() {}
50:

```

CH02

```

51: int DoMenu();
52: void DoDrawRect(Rectangle);
53: void DoGetArea(Rectangle);
54: void DoGetPerim(Rectangle);
55:

```

CH02

```

56: /*=====*/
57: int main()
58: {
59:     // initialize a rectangle to 30,5

```

CH06

```

60:     Rectangle theRect(30,5);
61:

```

```
62:   int choice = DrawRect;

CH03
63:   int fQuit = false;
64:

CH04
65:   while (!fQuit)
66:   {

CH05
67:       choice = DoMenu();

CH04
68:       if (choice < DrawRect || choice > Quit)
69:       {
70:           cout << "\nInvalid Choice, try again. ";
71:           cout << endl << endl;

CH04
72:           continue;
73:       }

CH07
74:       switch (choice)
75:       {

CH07
76:           case DrawRect:
77:               DoDrawRect(theRect);
78:               break;
79:           case GetArea:

CH05
80:               DoGetArea(theRect);
81:               break;
82:           case GetPerim:
83:               DoGetPerim(theRect);
84:               break;
85:           case ChangeDimensions:

CH03
86:               int newLength, newWidth;
87:               cout << "\nNew width: ";
88:               cin >> newWidth;
89:               cout << "\nNew height: ";
90:               cin >> newLength;

CH06
91:               theRect.SetSize(newWidth, newLength);
92:               DoDrawRect(theRect);
93:               break;
```

```

91:         case Quit:

CH03
91:             fQuit = true;
96:             cout << "\nExiting... " << endl << endl;
97:             break;

CH04
98:         default:
99:             cout << "Error in choice!" << endl;
100:             fQuit = true;
101:             break;
102:         } // end switch
103:     } // end while

CH05
104:     return 0;
105: } // end main
106:

CH07
107: int DoMenu()
108: {

CH08
109:     int choice;

CH02
110:     cout << endl << endl; // create two new lines
111:     cout << " *** Menu *** " << endl;
112:     cout << "(1) Draw Rectangle" << endl;
113:     cout << "(2) Area" << endl;
114:     cout << "(3) Perimeter" << endl;
115:     cout << "(4) Resize" << endl;
116:     cout << "(5) Quit" << endl;
117:

CH03
118:     cin >> choice;
119:     return choice;
120: }

121:
122: void DoDrawRect(Rectangle theRect)
123: {

CH06
124:     int height = theRect.GetHeight();
125:     int width = theRect.GetWidth();
126:

CH07
127:     for (int i = 0; i < height; i++)
128:     {

```

```

129:     for (int j = 0; j < width; j++)
130:         cout << "x";
131:     cout << endl;
132: }
133: }
134:
135:
CH05
136: void DoGetArea(Rectangle theRect)
137: {
CH03
138:     cout << "Area: " << theRect.GetArea() << endl;
139: }
140:
CH05
141: void DoGetPerim(Rectangle theRect)
142: {
143:     cout << "Perimeter: " << theRect.GetPerim() << endl;
144: }
145: // ===== End of Listing =====

```

输出:

```

*** Menu ***
(1) Draw Rectangle
(2) Area
(3) Perimeter
(4) Resize
(5) Quit
1
*****
*****
*****
*****
*** Menu ***
(1) Draw Rectangle
(2) Area
(3) Perimeter
(4) Resize
(5) Quit
2
Area: 150

```

```

*** Menu ***
(1) Draw Rectangle
(2) Area
(3) Perimeter
(4) Resize
(5) Quit

```

```

3
Perimeter: 70

    *** Menu ***
(1) Draw Rectangle
(2) Area
(3) Perimeter
(4) Resize
(5) Quit
4

New Width: 10
New height: 8
*****
*****
*****
*****
*****
*****
*****
*****

    *** Menu ***
(1) Draw Rectangle
(2) Area
(3) Perimeter
(4) Resize
(5) Quit
2
Area: 80

    *** Menu ***
(1) Draw Rectangle
(2) Area
(3) Perimeter
(4) Resize
(5) Quit
3
Perimeter: 36

    *** Menu ***
(1) Draw Rectangle
(2) Area
(3) Perimeter
(4) Resize
(5) Quit
5
Exiting...

```

这个程序使用了本周介绍过的大多数技巧。通过这一周的学习，读者应该不仅能输入、编译、链接和运行该程序，而且能够理解该程序的功能和工作原理。如果读者对该程序清单中的任何一行代码感到迷惑，应回过头去复习前一周的内容。在程序清单中，指出了代码行涉及的主题在哪章介绍过。

该程序提供一个文本菜单,等待用户做出选择。菜单用于处理一个矩形,其中包括打印矩形数据以及计算矩形的面积和周长的选项;用户还可以修改矩形的默认值。菜单没有执行功能齐备的程序应做的各种错误检查,但确实执行了某些检查。

在第 7~12 行,该程序清单声明了一个贯穿整个程序都将使用的新类型。

第 15~34 行声明了 `Rectangle` 类,其中包含用于获取和设置矩形的宽度和高度的公有存取器方法以及计算面积和周长的方法。第 37~47 行是没有被声明为内联的函数的定义。由于第 43~47 行创建了一个构造函数,因此在第 49 行创建了一个析构函数。

第 51~54 行是非类成员函数的原型,程序从第 57 行开始。正如前面指出的,该程序的基本功能是,创建一个矩形,然后打印一个包含 5 个选项的菜单:绘制矩形、计算面积、计算周长、调整矩形的大小以及退出。

第 63 行设置了一个标记,只要该标记为 `false`,菜单循环就将不断执行。仅当用户从菜单中选择 `Quit` 时,该标记才被设置为 `true`。

除 `ChangeDimensions` 外,其他每个选项都调用一个函数。这使得第 74~102 行的 `switch` 语句更为清晰。`ChangeDimensions` 不能调用函数,因为它必须修改矩形的尺寸。如果将矩形按值传递给函数(如 `DoChangeDimensions()`),将修改 `DoChangeDimensions()` 函数中的局部矩形拷贝的尺寸,而不是 `main()` 函数中矩形的尺寸。第 8 章和第 10 章将介绍如何突破这种限制,但现在在函数 `main()` 中进行修改。

请注意这里是如何使用枚举让 `switch` 语句更为清晰且更容易理解的。如果 `switch` 语句依赖于用户选择的数值(1~5),将必须经常根据菜单描述确定用户的选择是哪个选项。

第 68 行检查用户的选择,确保它在指定的范围内。如果不在指定的范围内,就打印一条错误消息,并再次显示菜单。注意, `switch` 语句中包含一个不可能出现的默认条件,旨在帮助调试。如果程序正常运行,将永远不会执行这条语句。

祝贺你完成了第 1 周的学习!现在你能够创建和理解复杂的 C++ 程序了。当然,还有很多的事情要做,下一周将首先介绍 C++ 中最困难的概念:指针。请不要放弃,你将深入了解面向对象编程的概念和用法、虚函数以及这种强大语言的许多高级功能。

先休息一下,欣赏一下自己的成果,然后翻到下一页,开始第 2 周的学习。