

第 3 章

C 语言上机开发环境介绍

学习要点

- 掌握 Borland C++ 3.1 IDE 的使用方法;
- 掌握 Borland C++ 3.1 环境下程序调试方法;
- 掌握 Visual C++ 6.0 环境下 C 程序调试、运行方法。

3.1 Borland C++ 3.1 开发环境

Borland C++ 3.1 是 Borland 公司开发的 C/C++ 语言集成开发环境 (IDE)，它集程序编辑、编译、链接、调试与运行于一体，是许多 C 语言爱好者学习 C 语言上机编程的开发工具。本节主要介绍 Borland C++ 3.1 开发环境下的程序编辑、编译、链接及调试的有关方法，以及工程文件的使用方法和带参数的 main 函数的运行方法。

3.1.1 Borland C++ 的启动及准备工作

在启动 BC 的 IDE 环境之前，先要进行一些必要的准备工作：

- (1) 弄清楚 Borland C++ 3.1 被安装在硬盘的哪个目录(文件夹)下，假设为 C:\Borlandc。
- (2) 在存放数据的硬盘中建立自己的目录，以便将所编写的程序存放在此目录中，假设为 C:\Cppprg。

按照下面的方法之一启动 BC 的 IDE 环境：

- 如果桌面上有“Borland C++ for DOS”图标存在，使用鼠标左键进行双击，即可打开 BC 环境。
- 通过双击“我的电脑”→“本地磁盘 (C:)”→BORLANDC→BIN，再找到执行文件 BC.exe，用鼠标双击之，即可打开 BC 环境。
- 用鼠标单击桌面左下角的“开始”菜单，然后将鼠标上移到“运行 (R)”处，单击鼠标会出现一个“运行”对话框，再在弹出的对话框中键入 c:\borlandc\bin\bc.exe 字样，如图 3-1 所示，然后单击“确定”按钮即可打开 BC 环境。



图 3-1 “运行”弹出的对话框

- 进入 MS-DOS 环境（其方法：在图 3-1 中的对话框中键入 `cmd`，然后单击“确定”按钮；或单击“开始”→“程序”→“附件”→“命令提示符”），然后输入 `c:\borlandc\bin\bc.exe`，即可打开 BC 环境（↵表示回车）。进入 BC 环境以后，系统将出现如图 3-2 所示的 BC 编程环境。



图 3-2 Borland C++ 3.1 环境窗口

设定工作目录：

进入 BC 的 IDE 环境以后，需要进行工作目录的设定。其具体方法就是按 ALT+O 键激活 Options 菜单，选择 Directories，如图 3-3 所示，打开 Directories 对话框，如图 3-4 所示。

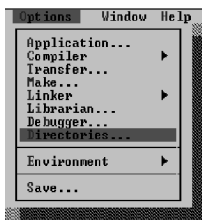


图 3-3 Options 菜单图



图 3-4 工作目录对话框

Include Directories: 用于设定 BC 中包含文件（如 `stdio.h` 文件等）所在的目录，它位于 Borlandc 下的 INCLUDE 子目录下，可根据需要进行设置。

Library Directories: 用于设定 BC 中库文件所在的目录，它位于 Borlandc 下的 LIB 子目录下，你可根据需要进行设置。

Output Directories: 用于设定源程序经编译和链接以后生成的文件（如目标文件 `.obj`、执行文件 `.exe`）所存放的目录。在图 3-4 中，设定输出文件的目录为 `C:\CPPPRG`，也就是说在 IDE 环境下编译所生成的执行文件将存放在 `C:\CPPPRG` 目录下。

Source Directories: 用于设定源程序文件目录。

以上目录设定好以后，单击 OK 按钮保存即可。一旦设定完毕，如果以后不需要更改，

就不必每次都进行设置。

3.1.2 Borland C++编辑器的使用

进入 BC 的 IDE 环境以后，如果是第一次启动 IDE，系统会打开一个新的窗口，等待用户输入源程序。这时要按 ALT+F3 键关闭该窗口，我们要重新建立一个完整的程序输入流程。

按 ALT+F 键激活 File（文件）菜单，选择 Open 命令，如图 3-5 所示或直接按 F3 键，打开 Open a File（打开文件）对话框，如图 3-6 所示。在打开文件对话框的 Name 文本框中输入 C:\cppprg\test.cpp 后，按回车键返回编辑状态，如图 3-7 所示。

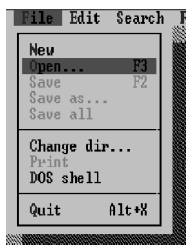


图 3-5 File 菜单图



图 3-6 打开文件对话框

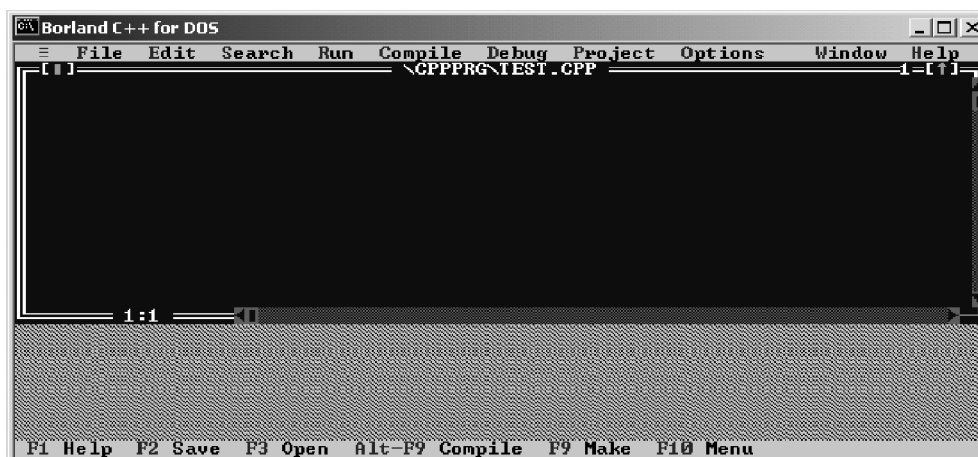


图 3-7 空的 test.cpp

在图 3-7 中，TEST.CPP 是窗口的标题（Borland C++用打开的文件名作为窗口的标题），具有蓝色背景的区域就是窗口的编辑区域，可以看到位于窗口左上角的光标正在闪烁。与窗口标题 TEST.CPP 位于同一行的有一个数字 1，它表示窗口的编号，因为 Borland C++ IDE 可同时打开多个窗口。窗口的左下角的 1:1 表示光标的位置（第几行、第几列）。这时可以在窗口中输入任何文本，也可以移动光标。具体键盘命令如表 3-1 所示。可以随便输入一些文本，再逐个试一下表 3-1 列出的命令。

表 3-1 键盘命令及其功能对照表

键盘命令		功 能
移动 光标	左方向键	向左移动一个字符
	右方向键	向右移动一个字符
	Ctrl+左方向键	向左移动一个单词
	Ctrl+右方向键	向右移动一个单词
	上方向键	向上移动一行
	下方向键	向下移动一行
	Ctrl+W	光标位置不变，窗口文本向下移动一行（窗口向上滚动一行）
	Ctrl+Z	光标位置不变，窗口文本向上移动一行（窗口向下滚动一行）
	PgUp	向上翻页
	PgDn	向下翻页
	Home	将光标移动到光标所在行的行首
	End	将光标移动到光标所在行的行尾
	Ctrl+Q E	将光标移动到窗口的顶部（注意：不是文件的顶部）
编辑 文本	Del	删除所在处的字符，如果光标位于行尾，下一行文本会续接到光标所在行上
	Backspace	删除光标左边的字符，如果光标位于行首，光标所在行会续接到上一行
	Ctrl+Y	删除光标所在行的全部文本
	Ctrl+Q Y	删除从光标处到行尾的全部字符
	Ctrl+T	删除光标所在处的一个单词，但光标左边的字符不删除
	Ctrl+N	在光标处插入新行
选择 文本块	Alt+Backspace	恢复前一次的操作
	Ctrl+K B	将光标的位置设置为被选择文本的首位置，如果被选择文本的首位置位于尾位置的前面，首位置和尾位置之间的文本将被高亮度显示
	Ctrl+K K	将光标的位置设置为被选择文本的尾位置，如果被选择文本的首位置位于尾位置的前面，首位置和尾位置之间的文本将被高亮度显示
	Ctrl+K H	隐藏或高亮度显示被选择的文本
	Ctrl+K C	将高亮度显示的被选择文本复制到光标处
	Ctrl+K V	将高亮度显示的被选择文本粘贴到光标处
	Shift+左方向键	将被选择文本块向左扩展或压缩一个字符
	Shift+右方向键	将被选择文本块向右扩展或压缩一个字符
	Shift+End	将被选择文本块扩展或压缩到行尾
	Shift+Home	将被选择文本块扩展或压缩到行首
	Shift+下方向键	将被选择文本块向下扩展或压缩一行
	Shift+上方向键	将被选择文本块向上扩展或压缩一行
	Shift+PaDn	将被选择文本块向下扩展或压缩一页
	Shift+PaUp	将被选择文本块向上扩展或压缩一页
	Shift+Ctrl+左方向键	将被选择文本块向左扩展或压缩一个单词
	Shift+Ctrl+右方向键	将被选择文本块向右扩展或压缩一个单词
	Shift+Ctrl+End	将被选择文本块扩展或压缩至文件尾
	Shift+Ctrl+Home	将被选择文本块扩展或压缩至文件首
	Ctrl+Q B	将光标移动到被选择文本的首位置
	Ctrl+Q K	将光标移动到被选择文本的尾位置

续表

键盘命令		功 能
选择文本块	Ctrl+K P	打印被选择的文本
	Ctrl+K Y	删除被选择的文本
	Ctrl+Ins	将被选择的文本复制到粘贴板上
	Shift+Del	将被选择的文本剪切到粘贴板上
	Ctrl+Del	删除被选择的文本
	Ctrl+K I	将被选择的文本向右缩进一个字符
	Shift+Ins	将粘贴板上的文本复制到光标处
	Ctrl+K R	将一个文件的内容读入并插入到光标处
	Ctrl+K U	将被选择的文本向左移动一个字符
	Ctrl+K W	将被选择的文本写到一个文件中
常规命令	F3	打开一个文件
	F2	保存文件
	F5	将窗口最大化或恢复原来大小
	Ctrl+Q F	打开查找字符串的对话框
	Ctrl+L	继续上一次的字符串查找或替换
	Ctrl+Q A	打开查找替换字符串的对话框
	F1	打开求助窗口
	Shift+F1	打开求助索引窗口，这时输入要求助的函数名，就可以找到相关的帮助文档
	Ctrl+F1	打开光标所在处单词的帮助文档
	Ins	插入模式的开/关
	Alt+X	退出 Borland C 的集成开发环境

注意：

Ctrl + K C 是指按住 Ctrl 键不放，先后按一下 K 键和 C 键。

Shift + F1 是指按住 Shift 后再按一下 F1 键。

其他依次类推。

下面将正式输入一个程序，这个程序的内容如图 3-8 所示。这时窗口的左下角出现了一个*，它表明 TEST.C 文件被修改但尚未存盘。这时如果按 F2 键保存文件，此符号就会消失。

当用户打开多个窗口时，这些窗口中只有一个是活动的。这个活动的窗口显示在最前面，并且窗口的边框是白色双线，窗口的标题也是白色的。而其他窗口是不活动的，它们的窗口边框是灰色单线，窗口的标题也是灰色的。在如图 3-9 所示的编辑窗口中，新打开的窗口 TEST2.CPP 是活动的，用户可以编辑 TEST2.CPP 文件，但不能编辑 TEST.CPP 文件，因为窗口 TEST.CPP 是不活动的。用户可以激活某个窗口来编辑相应的程序。当按住 ALT 键不放，再输入窗口的编号，就可以将该编号代表的窗口激活。当然，也可以按 F6 键在各个窗口间切换。

这时，用户可以开始使用 Borland C++ IDE 了。



图 3-8 编辑 test.cpp



图 3-9 两个编辑窗口

3.1.3 程序的编译、链接、运行

如果读者已经熟悉了 Borland C++ 3.1 集成环境中的编辑器的使用。那么从现在开始，读者需要了解编程的完整流程以及程序编译、链接和运行的方法。基本操作步骤如下：

(1) 启动 BC 后，按 F3 键激活打开文件对话框，在 Name 框中输入一个你想要的文件名。这个文件名不能超过 8 个字符，并且后缀是 .cpp（假设文件名输入为 test.cpp）。按回车键开始编辑程序。

(2) 输入程序。建议事先在纸上写好程序，再输入到计算机中。程序可以是完全新编的，也可以直接打开原先已编写好的存放在磁盘上的源程序，然后根据情况进行相应的修改。

(3) 程序输入完后，仔细检查程序中是否存在语法错误或输入错误，在运行程序之前一定要按 F2 键，保存程序。

(4) 按 F9 键，系统将对源程序进行编译和链接，如果在编译和链接过程中没有任何错误，系统将生成 test.exe 可执行文件，并弹出链接成功对话框信息，如图 3-10 所示。如果要运行程序，按 Ctrl+F9 键，BC 将会执行这个程序，这个过程可能会很快，如果不注意，还会以为什么都没有发生，其实这期间进行了编译、链接和运行。要想查看程序运行的结果，可以按 ALT+F5 键，这时会看到程序运行结果，按任意键可返回到 BC 状态。

(5) 如果程序存在着语法错误，BC 会提示编译链接不成功。这时，需要根据错误提示（可能不只一条）修改程序的错误。

(6) 如果编译、链接不成功，BC 会提示程序有错误，如图 3-11 所示。这时要按回车键，BC 会显示一个错误信息窗口，显示出所有的错误信息，如图 3-12 所示。



图 3-10 编译链接成功

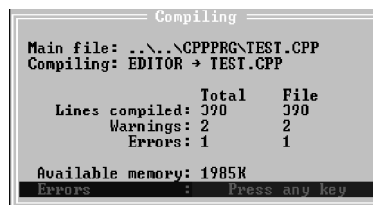


图 3-11 编译链接不成功

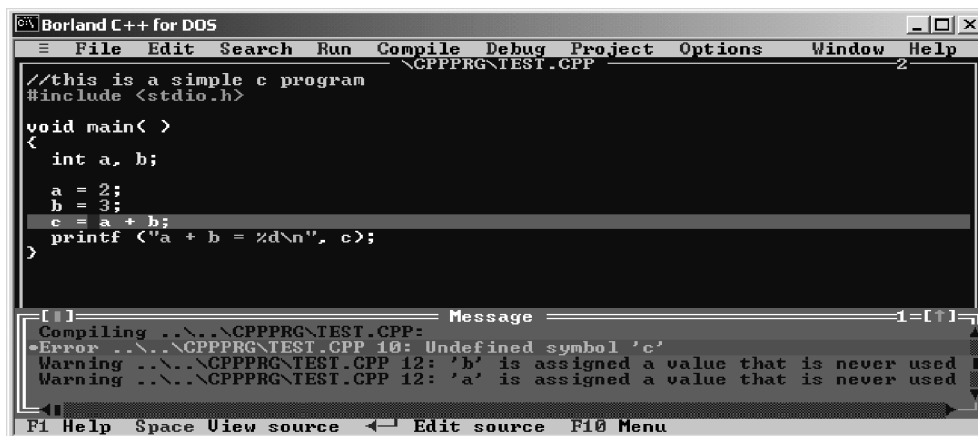


图 3-12 错误信息窗口

(7) 在错误信息窗口中，利用上下光标键移动亮度条，被高亮度显示的错误信息所对应的程序语句所在行也被高亮度显示（如图 3-12 所示）。这时，可以按回车键进入编辑状态，根据错误提示信息修改程序。

(8) 通常情况下，必须先改掉第一个错误，才能准确修改后续错误。有时仅仅是一个地方错了，就会引起多条错误信息。

(9) 修改完程序后回到步骤（4），直到程序能正常执行。这时，可以按 ALT+F5 键查看运行结果。

3.1.4 程序调试方法

任何一个程序员都无法保证所编写的程序没有错误，语法错在编译阶段就会被发现，链接错在链接时会被发现，但程序内部的逻辑错只能由程序员自己发现并定位。下面介绍的调试手段将有助于程序员快速发现程序的逻辑错。

1. 程序的单步执行

单步执行是指让程序一次只执行一行语句。单步执行程序时，程序员可以观察某些变量或表达式的值，由此来判断哪里出了问题。要想单步执行，必须使程序进入调试状态而不是运行状态。

程序只有在下面几种情况下才能进入调试状态：

- 启动 BC，打开某源程序文件，按 F7 键，进入调试状态。
- 启动 BC，打开某源程序文件，将光标移到某条语句处，按 F4 键程序执行到光标所在行的语句处便停下来，进入调试状态。
- 启动 BC，打开某源程序文件，在某条语句处设置断点，按 Ctrl+F9 键，程序开始执行，执行到断点处便停下来，进入调试状态。

进入调试状态后，程序的下一条要执行的语句将被用青色亮度条高亮度显示。这时，程序员可以按 F7 键让程序一次执行一条语句，也可以将光标移到下面的任何语句处，按 F4 键让程序直接执行到光标所在行；可以按 Ctrl+F2 键终止调试；可以按 Ctrl+F9 键直接运行程序。

下面的步骤演示了 F7 和 F4 键的作用。

- 启动 BC，输入程序。将光标移动到第 10 行，按下 F4 键，程序执行到第 10 行处停下来（如图 3-13 所示）。按 ALT+F5 键，会发现第 9 行的 printf 函数已经执行完毕。

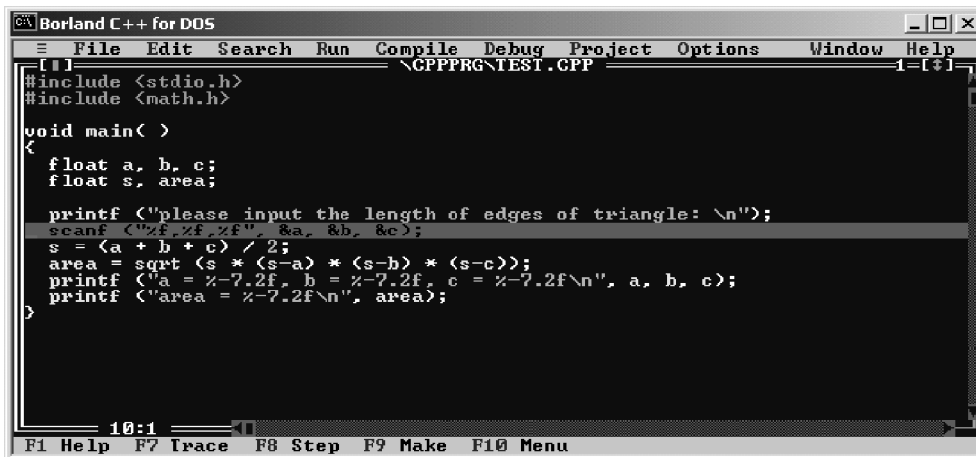


图 3-13 F4 键的作用

- 按 F7 键，程序开始执行 scanf 函数调用，输入 1, 2, 3↵，程序在第 11 行处停下来（如图 3-14 所示）。

- 接着按 F7 键，直到程序执行完毕。这一步读者可自己操作。

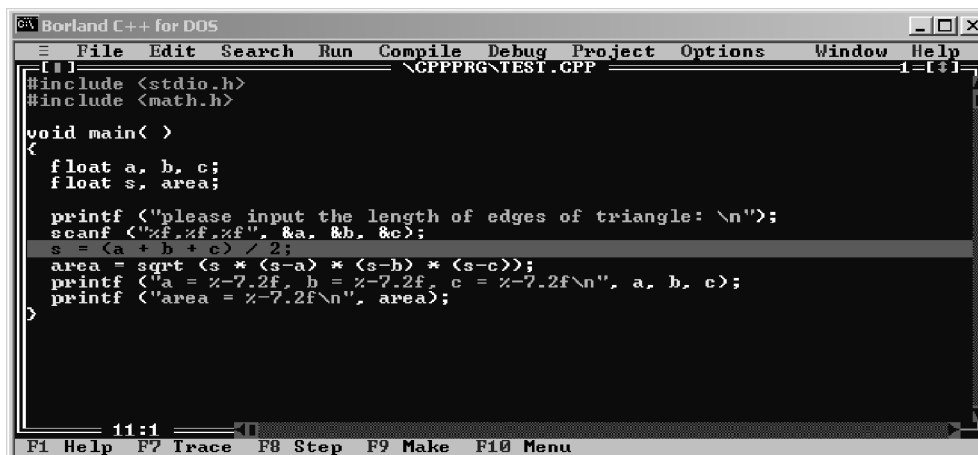


图 3-14 F7 键的作用

2. 在程序中设置断点

断点是程序中的一个标记，当程序执行到断点时，程序会暂停下来。

设置断点的命令是 Ctrl+F8 键。将光标移到某条语句处，按 Ctrl+F8 键，这行文本将被用红色亮度条高亮度显示。将光标移到断点处，按 Ctrl+F8 键可以取消这个断点。

将光标移到第 14 行，按下 Ctrl+F8 键就可以设置断点，这时，按 Ctrl+F9 键继续执行，执行到第 14 行会停下来（如图 3-15 所示）。

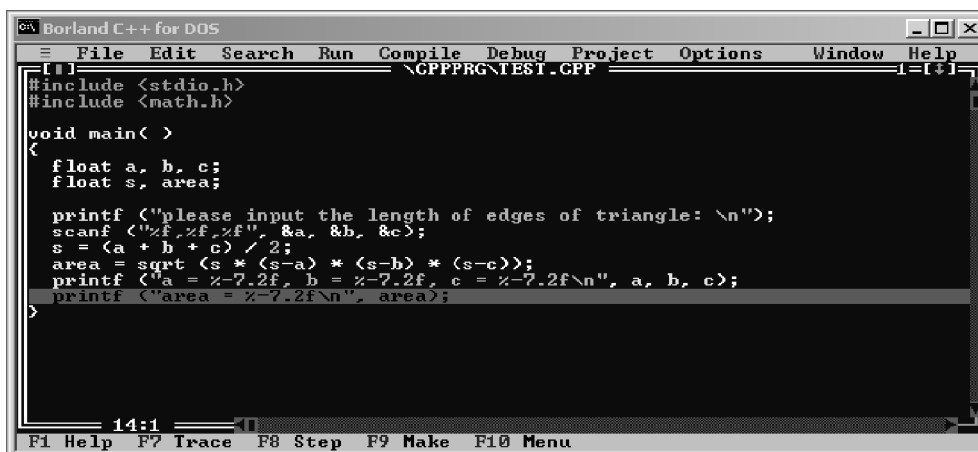


图 3-15 执行到断点

3. 在程序调试过程中观察变量和表达式的值

程序员调试程序的任务有两个：一是观察语句执行的顺序是否正确，二是观察变量或表达式的值是否正确。程序员可以从不正常的语句执行顺序和不正确的变量或表达式的值来判断程序的逻辑错出现在何处。

要想观察变量或表达式的值，首先应打开 Watch 窗口。BC 的 IDE 环境中有 Watch 窗口专门用来观察变量或表达式的值。选择 Window 菜单的 Watch 命令（按键顺序是：Alt+W，W），Watch 窗口将显示出来，如图 3-16 所示。



图 3-16 Watch 窗口

在 Watch 窗口中可以同时观察多个变量或表达式的值。当 Watch 窗口处于活动状态时，程序员可以按上下光标键，选择不同的行。如果被选择的是空行，按回车键，可以输入要观察的变量名或表达式（如图 3-17 所示）。如果被选择的行包含有变量或表达式，按回车键，可以修改变量名或表达式（如图 3-18 所示）。输入完后，按回车键即可。



图 3-17 增加观察项



图 3-18 修改观察项

如果被选择的行是非空行，按 Del 键可删除该行。

Watch 窗口中的变量和表达式的值会随着程序的执行而变化。图 3-19 演示了上述例子 test.cpp 执行到第 14 行时，Watch 窗口中的内容。

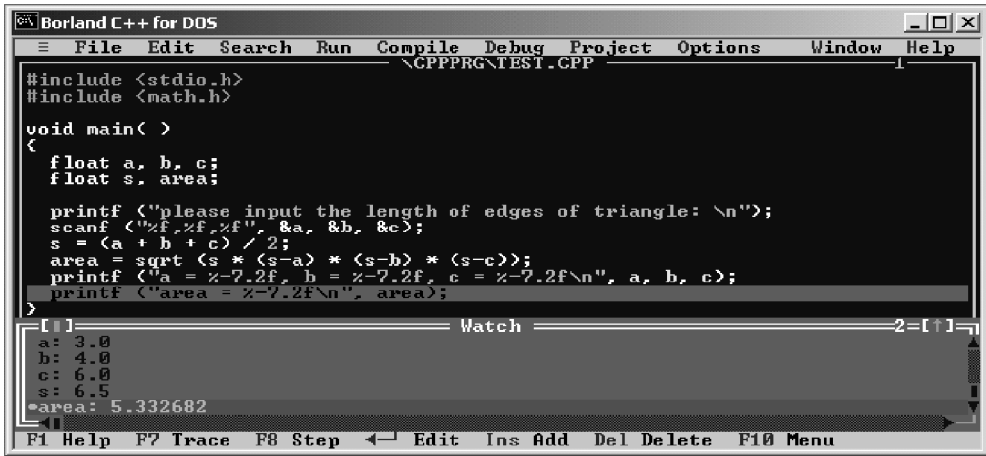


图 3-19 观察结果

4. 终止调试

当程序处于调试状态时，按 Ctrl+F2 键就会终止调试，程序也将终止运行。