Here is an example to illustrate the algorithm. Suppose a credit card statement showed a previous balance of $850. Eleven days before the end of the billing cycle, a payment of $400 is made. The billing cycle for the month is 31 days, and the monthly interest rate is 1.32%. The calculation of the interest charge is as follows:

Step 1. $850 \times 31 = $26,350
Step 2. $400 \times 11 = $4,400
Step 3. $26,350 - $4,400 = $21,950
Step 4. $21,950 \div 31 = $708.06
Step 5. $708.06 \times 0.0132 = $9.34

Write a program that computes the monthly interest charge on a credit card account. Your program should acquire as input the previous balance, the payment amount, the number of days in the billing cycle, the day of the billing cycle the payment was made, and the monthly interest rate.

2.40 Suppose a, b, and c are int variables initialized respectively to 1, 2, and 3. What values are assigned to e, f, and g?

a) \text{int } e = ++a;
b) \text{int } f = b++;
c) \text{int } g = ++c + c++;

2.41 Write a code segment that defines a Scanner variable \texttt{stdin} that is associated with \texttt{System.in}. The code segment should then define two int variables \texttt{a} and \texttt{b} such that they are initialized with the next two input values from the standard input stream.

2.11 \textbf{PROGRAMMING PROJECT — YOU}

Developing software can be a rewarding experience—it is a chance to be both creative and analytical. People of all sorts report high levels of satisfaction in completing a programming task and demonstrating its capabilities. The goal of this introductory problem-solving case study and our other case studies is for you to experience such satisfaction as you learn to program and problem solve in Java.

At the end of the main body of each chapter, we present a problem or two for you to consider. The problems have been selected for their relevance to the concepts being discussed, as well as being appropriately complex and interesting to a broad range of people. We provide background and problem-solving guidance on how to go about designing and implementing solutions.

The first case study is a personal one—design and produce a display giving a short autobiography that could be used as part of an application for a club membership. The information you supply and how you present it is your choice. However, at a minimum you must supply contact information such as your name and e-mail address. To get you...
Programming project — you started, you may consider describing some of your personal favorites (e.g., author, artist, music, activities) and plans (e.g., career, travel).

**ANALYSIS AND DESIGN**

So what tasks must be accomplished to produce a Java program that creates the desired display? For one, you must determine what personal information is to be displayed. For another, you must determine how the information is to be displayed. After making these decisions, you can create a preliminary design of the information and its layout. You might want to get feedback from friends. They can make suggestions on the information to make sure it is sufficiently interesting.

Snapshots of our own personal efforts are given in Figure 2.5 and Figure 2.6.

**Figure 2.5** A MeDisplay compilation and run.

![Image of MeDisplay compilation and run]

**Implementation**

Name the program `MeDisplay.java`. To implement and run the program, you will need access to Java tools. Some versions of this text are bundled with a CDROM supplying a
Java IDE for you to use. In addition, our website www.javaprogramdesign.com contains links to some freely available tools such as the SDK.

For this case study, you will develop a class `MeDisplay` that consists of a single method `main()`. The class should be documented and laid out appropriately. For example, the code should make use of program header and method comments, whitespace and a consistent indentation scheme.

For your information, our programs produced each line of text using individual statements.

**Testing**

Just as you should seek the comments of others to evaluate your preliminary design, you should also demonstrate your working program to others to get advice on how the presentation might be improved.