11. Write a pay-raise program that requests a person's first name, last name, and current annual salary, and then displays their salary for next year. People earning less than $40,000 will receive a 5% raise, and those earning $40,000 or more will receive a raise of $2,000 plus 2% of the amount over $40,000. Use Sub procedures for input and output, and a Function procedure to calculate the new salary. See Fig. 5.12.

FIGURE 5.12   Possible output for Exercise 11.

FIGURE 5.13   Possible output for Exercise 12.

12. Write a program to calculate the balance and minimum payment for a credit card statement. See Fig. 5.13. The program should use the event procedure shown in Fig. 5.14. The finance charge is 1.5% of the old balance. If the new balance is $20 or less, the minimum payment should be the entire new balance. Otherwise, the minimum payment should be $20 plus 10% of the amount of the new balance above $20.

```
Private Sub btnCalculate_Click(...) Handles btnCalculate.Click
  Dim oldBalance, charges, credits, newBalance, minPayment As Double
  InputData(oldBalance, charges, credits)
  CalculateNewValues(oldBalance, charges, credits, newBalance, minPayment)
  DisplayData(newBalance, minPayment)
End Sub
```

FIGURE 5.14   Event procedure for Exercise 12.

13. Write a program to calculate the monthly values associated with a mortgage. See Fig. 5.15. The program should use the event procedure shown in Fig. 5.16. The interest paid each

FIGURE 5.15   Sample output for Exercise 13.

```
Private Sub btnCalculate_Click(...) Handles btnCalculate.Click
    Dim annualRateOfInterest, monthlyPayment, begBalance As Double
    Dim intForMonth, redOfPrincipal, endBalance As Double
    InputData(annualRateOfInterest, monthlyPayment, begBalance)
    Calculate(annualRateOfInterest, monthlyPayment, begBalance,
            intForMonth, redOfPrincipal, endBalance)
    DisplayData(intForMonth, redOfPrincipal, endBalance)
End Sub
```

FIGURE 5.16   Event procedure for Exercise 13.

month is the monthly rate of interest applied to the balance at the beginning of the month. Each month the reduction of principal equals the monthly payment minus the interest paid. At any time, the balance of the mortgage is the amount still owed; that is, the amount required to pay off the mortgage. The end of month balance is calculated as [beginning of month balance] − [reduction of principal].

14. Write a program to determine a person's weekly pay, where they receive time-and-a-half for overtime work beyond forty hours. See Fig. 5.17. The program should use the event procedure shown in Fig. 5.18.

FIGURE 5.17   Sample output for Exercise 14.

```
Private Sub btnCalculate_Click(...) Handles btnCalculate.Click
    Dim hours, payPerHour, overtimeHours, pay As Double
    InputData(hours, payPerHour)
    CalculateValues(hours, payPerHour, overtimeHours, pay)
    DisplayData(overtimeHours, pay)
End Sub
```

FIGURE 5.18   Event procedure for Exercise 14.

Solutions to Practice Problems 5.3

1. `age before age`

2. Since *state* is a string variable, its default value is the keyword *Nothing*. The assignment of *Nothing* to an argument makes Visual Basic nervous. Therefore, to keep Visual Basic happy, we will assign the empty string to String variables that are passed to procedures. That is, we will change the first line inside the event procedure to `Dim state As String = ""`.

   Since the default value of the numeric variable *pop* is 0, Visual Basic has no issue with passing that value to a Sub procedure.

▦ Comments

1. In the function FICA_Tax, care has been taken to avoid computing social security benefits tax on income in excess of $106,800 per year. The logic of the program makes sure an employee whose income for the year crosses the $106,800 threshold during a given week is taxed only on the difference between $106,800 and their previous year-to-date earnings.

2. The two functions TaxMarried and TaxSingle use Select Case blocks to incorporate the tax brackets given in Tables 5.5 and 5.6 for the amount of federal income tax withheld. The upper limit of each Case clause is the same as the lower limit of the next Case clause. This ensures that fractional values for *adjPay*, such as 138.50 in the TaxSingle function, will be properly treated as part of the higher salary range.

CHAPTER 5   SUMMARY

1. A *general procedure* is a portion of a program that is accessed by event procedures or other general procedures. The two types of general procedures are *Function procedures* and *Sub procedures*.

2. *Function procedures* are defined in blocks beginning with Function headers and ending with End Function statements. A function is executed by a reference in an expression and returns a value.

3. *Sub procedures* are defined in blocks beginning with Sub headers and ending with End Sub statements. A Sub procedure is accessed (called) by a statement consisting of the name of the procedure.

4. In any procedure, the *arguments* appearing in the calling statement must match the *parameters* of the Sub or Function statement in number, type, and order. They need not have the same names.

5. The *lifetime* of a variable or constant is the period during which it remains in memory. (The value of the variable might change over its lifetime, but it always holds some value.)

6. The *scope* of a variable or constant is the portion of the program that can refer to it. A variable or constant declared inside a Function, Sub, or event procedure has *local* scope and is visible only inside the procedure.

7. *Structured programming* uses modular design to refine large problems into smaller subproblems. Programs are coded using the three logical structures of sequences, decisions, and loops.

CHAPTER 5   PROGRAMMING PROJECTS

1. Write a program to determine a student's GPA. See Fig. 5.25. The user should enter the grade (A, B, C, D, or F) and the number of credit hours for a course, and then click on the *Record This Course* button. The user should then repeat this process for all his or her courses. After all the courses have been recorded, the user should click on the *Calculate GPA* button. A Function procedure should be used to calculate the quality points for a course. **Hint:** This program is similar to Example 5 in Section 5.1.

2. A fast-food vendor sells pizza slices ($1.75), fries ($2.00), and soft drinks ($1.25). Write a program to compute a customer's bill. The program should request the quantity of each item ordered in a Sub procedure, calculate the total cost with a Function procedure, and use a Sub procedure to display an itemized bill. A sample output is shown in Fig. 5.26.
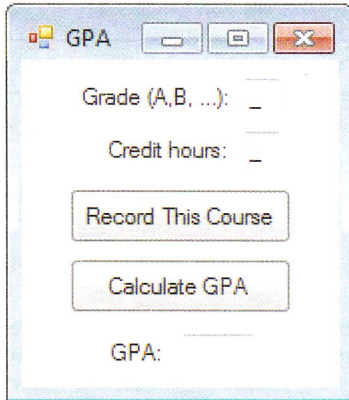
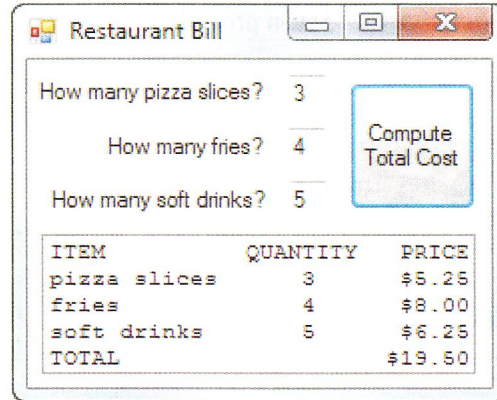**FIGURE 5.25** Form design for Programming Project 1.

FIGURE 5.26 Sample output for Programming Project 2.

3. A furniture manufacturer makes two types of furniture—chairs and sofas. The cost per chair is $350, the cost per sofa is $925, and the sales tax rate is 5%. Write a program to create an invoice form for an order. See Fig. 5.27. After the data on the left side of Fig. 5.27 are entered, the user can display an invoice in a list box by pressing the *Process Order* button. The user can click on the *Clear Order Form* button to clear all text boxes and the list box, and can click on the *Quit* button to exit the program. The invoice number consists of the capitalized first two letters of the customer's last name, followed by the last four digits of the zip code. The customer name is input with the last name first, followed by a comma, a space, and the first name. However, the name is displayed in the invoice in the proper order. The generation of the invoice number and the reordering of the first and last names should be carried out by Function procedures.
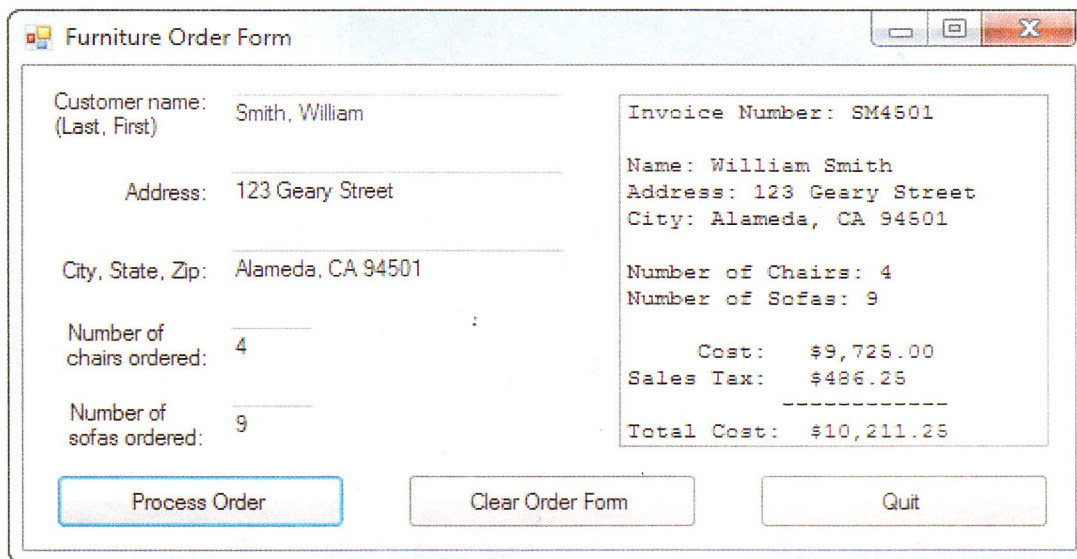
VideoNote
Hardware store (Homework)

FIGURE 5.27 Sample run for Programming Project 3.

4. Table 5.9 contains seven proverbs and their truth values. Write a program that presents these proverbs one at a time and asks the user to evaluate them as true or false. The program should then tell the user how many questions were answered correctly and display one