5 Selection-Making Decisions

REVIEW QUESTIONS

1. Logical data are data that can be interpreted as true or false.

a. True

3. Each case-labeled statement may identify one or more statements.

a. True

5. To ensure that a character is uppercase, the *toupper* conversion function is used.

a. True

7. Which of the following is not a relational operator in C++?

b. =

9. Which of the following is not a syntactical rule for the *if…else* statement?

e. The selection expression cannot have a side effect.

11. There are two different ways to implement a multiway selection in C++. They are

b. *else-if* and *switch*

13. Which of the following statements about the *else-if* is false?

c. The *else-if* requires integral values in its expression.

EXERCISES

15. If x = 3, y = 0, and z = –4, the expressions evaluate as:

a. true

b. true

c. true

d. true

e. true

17. Given x = –2, y = 5, and t = –4, the values are:

a. false

b. true

c. true

d. true

e. false

19. Given x = 4, y = 0, and z = 2, the variables contain:

x = 4 y = 1 z = 2

21. Given x = 4, y = 0, and z = 2, the variables contain:

x = 4 y = 2 z = 2

23. Given x = 0, y = 0, and z = 1, the variables contain:

x = 0 y = 0 z = 1

25. Given x = 0, y = 0, and z = 1, the variables contain:

x = 0 y = 0 z = 1

27. Given x = 0, y = 0, and z = 1, the variables contain:

a. x = –1 y = 0 z = 0

29. Given x = 0, y = 0, and z = 1, the variables contain:

x = 1 y = 3 z = 1

31. Given x = 1, y = 3, and z = 0, the variables contain:

x = 4 y = 3 z = 0

33. The value of the *toupper* expressions is:

a. C (upper case 'C')

b. C (upper case 'C')

c. ?

d. 7

35. A minimum of three test values are required as shown in the truth table.

num	Test 1	Test2
5	T	T
15	T	F
–5	F	N/A

PROBLEMS

37.

```
// Test for best score method #2
score >= 90 ? best = 1 : best = 0;
```

39.

```
// Print num if flag true
if (flag)
   cout << num;
```

41.

```
// Test for zero divisor
if (divisor)
   quotient = static_cast<double>(dividend) / divisor;
else
   quotient = divisor;
cout << "dividend: "  << dividend
     << " divisor : " << divisor
     << " quotient: " << quotient << endl;
```

43.

```
// If character is E, add and print
if (aChar == 'E')
   {
    c++;
    cout << "Value is E\n";
   } // if
```

45.

```
// if num1 is 10, square it; if 9, read and process
if (num1 == 10)
   num1 *= num1;
else if (num1 == 9)
   {
    cout << "Enter new number: ";
    cin  >> num1;
    if (num1 == 2 || num1 == 3)
       {
        num1 *= 99;
        cout << "Number * 99 is: " << num1 << endl;
       } // if
   } // else
```

47.

```
// read and process x and y
cout << "Enter x and y: ";
cin  >> a >> y;

if (x > 0)
   {
    if (y > 0)
       {
        z = x;
        y = x + 1;
       } // y > 0
    else
       z = y;
    z++;
   } // x > 0
else
   {
    y = x - 1;
    z = 2 * x;
   } // x <= 0
cout << "x: " << x << " y: " << y << " z: " << z;
```

49.

```
/* ==================== day_of_week ================
   This function display the day of week corresponding
   to numbers between 0 and 6.
      Pre  given an integer representing a day
      Post day of week printed
*/
void day_of_week (int day)
{
   switch (day)
      {
       case 0 : cout << "Sunday";
```

```
                    break;
            case 1 : cout << "Monday";
                    break;
            case 2 : cout << "Tuesday";
                    break;
            case 3 : cout << "Wednesday";
                    break;
            case 4 : cout << "Thursday";
                    break;
            case 5 : cout << "Friday";
                    break;
            case 6 : cout << "Saturday";
                    break;
            default: cout << "\n\aNumber is not valid!\n";
                    break;
          }
      return;
   }  // day_of_week
```

51.

```
   /* ================parkingCharge==================
      Calculate the amount due for each of the valid
      vehicles.
          Pre  type of vehicle and the hours
          Post parking charge returned,
               -1 returned if invalid code
   */
   #define CAR     2.00
   #define BUS     3.00
   #define TRUCK   4.00

   double parkingCharge (char   vehicle, double hours)
   {
      double total;
      switch (vehicle)
         {
          case 'c' :  total  =  CAR * hours;
                      break;
          case 'b' :  total  =  BUS * hours;
                      break;
          case 't' :  total  =  TRUCK * hours;
                      break;
          default  :  cout << "\nVehicle not valid!\n";
                      total  =  -1;
                      break;
         } // switch
      return total;
   }  // parkingCharge
```