Answers to Review Questions
Chapter 4

1. In an if/else if statement, the conditions are tested until one is found to be true. The conditionally executed statement(s) are executed and the program exits the if/else if statement. In a series of if statements, all of the if statements execute and test their conditions because they are not connected.

2. The trailing else provides code that is executed when none of the conditions in the if/else if statement are true.

3. A flag is a Boolean variable signaling that some condition exists in the program. When the flag is set to false it indicates the condition does not yet exist. When the flag is set to true it indicates that the condition does exist.

4. Yes. The if statement can test any value that yields a Boolean value (true or false) or a numeric value. When testing a numeric expression, a nonzero numeric value is considered true, and the value 0 is considered false.

5. It takes two expressions as operands and creates a single expression that is true only when both subexpressions are true.

6. It takes two expressions as operands and creates a single expression that is true when either of the subexpressions are true.

7. Because they test for specific relationships between items. The relationships are greater-than, less-than, equal-to, greater-than or equal-to, less-than or equal-to, and not equal-to.

8. It visually sets the conditionally-executed statements apart from the surrounding code. This is so you can easily identify the code that is conditionally-executed.

9. relational
10. true, false
11. false, true
12. false
13. true
14. braces
15. true, false
16. default
17. nested
18. &&
19. ||
20. !
21. left-to-right
22. &&
23. ||
24. block
25. strings (C-strings, specifically)
26. conditional
27. integer
28. integer constant
29. break

30. A) 1
   B) 0
   C) 0
   D) 1

31. if (y == 0)
    x = 100;

32. if (y == 10)
    x = 0;
else
    x = 1;

33. if (sales < 10000)
    commission = .10;
else if (sales <= 15000)
    commission = .15;
else
    commission = .20;

34. if (minimum)
    hours = 10;

35. if (amount1 > 10)
    if (amount2 < 100)
        cout << (amount1 > amount2 ? amount1 : amount2);

36. if (grade >= 0 && grade <= 100)
    cout << "The number is valid.";

37. if (temperature >= -50 && temperature <= 150)
    cout << "The number is valid.";

38. if (hours < 0 || hours > 80)
    cout << "The number is not valid.";

39. if (strcmp(title1, title2) < 0)
    cout << title1 << " " << title2 << endl;
else
    cout << title2 << " " << title1 << endl;

40. switch (choice)
{
    case 1:
        cout << fixed << showpoint << setprecision(2);
        break;
    case 2:
    case 3:
        cout << fixed << showpoint << setprecision(4);
        break;
    case 4:
        cout << fixed << showpoint << setprecision(6);
        break;
    default:
        cout << fixed << showpoint << setprecision(8);
}

41. C, A, B
42. false
43. false
44. true
45. true
46. false
47. true
48. false
49. true
50. false
51. true
52. false
53. false
54. T
55. F
56. T
57. T
58. The first cout statement is terminated by a semicolon too early.
The definition of score1, score2, and score3 should end with a semicolon.
The following statement:
    if (average = 100)
should read:
    if (average == 100)
perfectScore is used before it is defined.
The following if statement should not be terminated with a semicolon:
    if (perfectScore);
The conditionally executed block in this if statement should end with a closing brace.
59. The conditionally executed blocks in the if/else construct should be enclosed in braces. The following statement:
   
   ```cpp
   cout << "The quotient of " << num1 <<
   ```

   should read:
   
   ```cpp
   cout << "quotient of " << num1;
   ```

60. The trailing else statement should come at the end of the if/else construct.

61. The if statement does not properly test the strings for equality. The `strcmp` function should be used instead of the `==` operator.

62. A switch statement cannot be used to test relational expressions. An if/else if statement should be used instead.

63. It should read if (!x > 20))

64. It should use && instead of ||.

65. It should read if (count < 0 || count > 100)

66. The : and ? are transposed. The statement should read:
   
   ```cpp
   z = (a < 10) ? 0 : 7;
   ```