CHAPTER 9

Programming Languages

(Solutions to Odd-Numbered Problems)

Review Questions

1. A machine language uses only 0s and 1s for instructions and addresses. An assembly language uses symbols to represent instructions and addresses.

3. The machine language is the only language understood by the computer hardware.

5. The four steps are lexical analysis, syntax analysis, semantic analysis, and code generation.

7. In the procedural paradigm, a program is an active agent that manipulates passive objects (data). In an object-oriented paradigm, data are designed as active objects. The action to be performed on these objects are included in the object.

9. In the functional paradigm a program is designed like a mathematical function. It allows the programmer to combine predefined primitive functions to create new functions.

Multiple-Choice Questions

11. a   **13.** b   **15.** a   **17.** b   **19.** c   **21.** b

Exercises

23.

```
int count;
int index;
int level;
```

25.

```
const char name ='A';
const int count = 1;
const float height = 1.82;
```

27. The statement is executed twice (once when A = 5 and the second time when A = 7). When A becomes 9, the loop is terminated.

29. The statement is executed eight times ($i = 5, 7, 9, 11, 13, 15, 17, 19$). Note that in each iteration the value of i is incremented twice: the first time inside the header (i++), the second time in the body of the loop ($i = i + 1$).

31.

```
A = 5;
do
{
        statement;
        A = A − 2;
} while (A < 8);
```

33.

```
i = 5;
while (i < 20)
{
        statement;
        i = i + 2;
}
```

35.

```
for (int A = 5; A < 8; A = A − 2)
{
        statement;
}
```

37. This is not possible because in a *do-while* loop, the body of the loop is executed at least once.

39. The following shows one possible solution.

```
while (true)
{
        statement;
}
```

41. The following shows one possible solution.

```
for (; true; )
{
        statement;
}
```

43. *Hello* is the variable, "Hello" is the literal.

45. A and B should be passed by value, S and P by reference.

47.

    a. It should be by reference if we can allow the subprogram change the value of A in the main program. The following shows the statement:

```
cube (A);
```

    b. Alternatively, we can pass A by value and let the function **cube** return the cube of A. In this case, the original value of A remains untouched in the main program. The following shows the statement:

```
result ← cube (A);
```

49. It can be passed either by value or by reference, but it is normally passed by value to keep the value of the variable in the main untouched.