

Introduction to C++ Programming, Input/Output and Operators

2

*What's in a name? that
which we call a rose
By any other name
would smell as sweet.*

—William Shakespeare

*High thoughts must have high
language.*

—Aristophanes

*One person can make a
difference and every person
should try.*

—John F. Kennedy

Objectives

In this chapter you'll learn:

- To write simple computer programs in C++.
- To write simple input and output statements.
- To use fundamental types.
- Basic computer memory concepts.
- To use arithmetic operators.
- The precedence of arithmetic operators.
- To write simple decision-making statements.

Self-Review Exercises

2.1 Fill in the blanks in each of the following.

a) Every C++ program begins execution at the function _____.

ANS: `main`.

b) A(n) _____ begins the body of every function and a(n) _____ ends the body.

ANS: left brace (`{`), right brace (`}`)

c) Most C++ statements end with a(n) _____.

ANS: semicolon.

d) The escape sequence `\n` represents the _____ character, which causes the cursor to position to the beginning of the next line on the screen.

ANS: semicolon.

e) The _____ statement is used to make decisions.

ANS: `if`.

2.2 State whether each of the following is *true* or *false*. If *false*, explain why. Assume the statement `using std::cout; is used`.

a) Comments cause the computer to print the text after the `//` on the screen when the program is executed.

ANS: False. Comments do not cause any action to be performed when the program is executed. They're used to document programs and improve their readability.

b) The escape sequence `\n`, when output with `cout` and the stream insertion operator, causes the cursor to position to the beginning of the next line on the screen.

ANS: True.

c) All variables must be declared before they're used.

ANS: True.

d) All variables must be given a type when they're declared.

ANS: True.

e) C++ considers the variables `number` and `NUMBER` to be identical.

ANS: False. C++ is case sensitive, so these variables are different.

f) Declarations can appear almost anywhere in the body of a C++ function.

ANS: True.

g) The modulus operator (`%`) can be used only with integer operands.

ANS: True.

h) The arithmetic operators `*`, `/`, `%`, `+` and `-` all have the same level of precedence.

ANS: False. The operators `*`, `/` and `%` have the same precedence, and the operators `+` and `-` have a lower precedence.

i) A C++ program that prints three lines of output must contain three statements using `cout` and the stream insertion operator.

ANS: False. One statement with `cout` and multiple `\n` escape sequences can print several lines.

2.3 Write a single C++ statement to accomplish each of the following (assume that neither `using` declarations nor a `using` directive have been used):

a) Declare the variables `c`, `thisIsAVariable`, `q76354` and `number` to be of type `int` (in one statement).

ANS: `int c, thisIsAVariable, q76354, number;`

b) Prompt the user to enter an integer. End your prompting message with a colon (`:`) followed by a space and leave the cursor positioned after the space.

ANS: `std::cout << "Enter an integer: ";`

c) Read an integer from the user at the keyboard and store it in integer variable `age`.

ANS: `std::cin >> age;`

d) If the variable `number` is not equal to 7, print "The variable number is not equal to 7".

```
ANS: if ( number != 7 )
    std::cout << "The variable number is not equal to 7\n";
```

e) Print the message "This is a C++ program" on one line.

```
ANS: std::cout << "This is a C++ program\n";
```

f) Print the message "This is a C++ program" on two lines. End the first line with C++.

```
ANS: std::cout << "This is a C++\nprogram\n";
```

g) Print the message "This is a C++ program" with each word on a separate line.

```
ANS: std::cout << "This\nis\na\nC++\nprogram\n";
```

h) Print the message "This is a C++ program". Separate each word from the next by a tab.

```
ANS: std::cout << "This\tis\ta\tC++\tprogram\n";
```

2.4 Write a statement (or comment) to accomplish each of the following (assume that using declarations have been used for `cin`, `cout` and `endl`):

a) State that a program calculates the product of three integers.

```
ANS: // Calculate the product of three integers
```

b) Declare the variables `x`, `y`, `z` and `result` to be of type `int` (in separate statements) and initialize each to 0.

```
ANS: int x = 0;
    int y = 0;
    int z = 0;
    int result = 0;
```

c) Prompt the user to enter three integers.

```
ANS: cout << "Enter three integers: ";
```

d) Read three integers from the keyboard and store them in the variables `x`, `y` and `z`.

```
ANS: cin >> x >> y >> z;
```

e) Compute the product of the three integers contained in variables `x`, `y` and `z`, and assign the result to the variable `result`.

```
ANS: result = x * y * z;
```

f) Print "The product is " followed by the value of the variable `result`.

```
ANS: cout << "The product is " << result << endl;
```

g) Return a value from `main` indicating that the program terminated successfully.

```
ANS: return 0;
```

2.5 Using the statements you wrote in Exercise 2.4, write a complete program that calculates and displays the product of three integers. Add comments to the code where appropriate. [*Note:* You'll need to write the necessary using declarations or directive.]

ANS: (See program below.)

```
1 // Calculate the product of three integers
2 #include <iostream> // allows program to perform input and output
3 using namespace std; // program uses names from the std namespace
4
5 // function main begins program execution
6 int main()
7 {
8     int x = 0; // first integer to multiply
9     int y = 0; // second integer to multiply
10    int z = 0; // third integer to multiply
11    int result = 0; // the product of the three integers
12
```

```

13 cout << "Enter three integers: "; // prompt user for data
14 cin >> x >> y >> z; // read three integers from user
15 result = x * y * z; // multiply the three integers; store result
16 cout << "The product is " << result << endl; // print result; end line
17 } // end function main

```

2.6 Identify and correct the errors in each of the following statements (assume that the statement using `std::cout`; is used):

a) `if (c < 7);`
`cout << "c is less than 7\n";`

ANS: *Error:* Semicolon after the right parenthesis of the condition in the `if` statement.
Correction: Remove the semicolon after the right parenthesis. [*Note:* The result of this error is that the output statement executes whether or not the condition in the `if` statement is true.] The semicolon after the right parenthesis is a null (or empty) statement that does nothing. We'll learn more about the null statement in Chapter 4.

b) `if (c => 7)`
`cout << "c is equal to or greater than 7\n";`

ANS: *Error:* The relational operator `=>`.
Correction: Change `=>` to `>=`, and you may want to change "equal to or greater than" to "greater than or equal to" as well.

Exercises

NOTE: Solutions to the programming exercises are located in the ch02solutions folder.

2.7 Discuss the meaning of each of the following objects:

a) `std::cin`

ANS: This object refers to the standard input device that is normally connected to the keyboard.

b) `std::cout`

ANS: This object refers to the standard output device that is normally connected to the screen.

2.8 Fill in the blanks in each of the following:

a) _____ are used to document a program and improve its readability.

ANS: Comments

b) The object used to print information on the screen is _____.

ANS: `std::cout`

c) A C++ statement that makes a decision is _____.

ANS: `if`

d) Most calculations are normally performed by _____ statements.

ANS: assignment

e) The _____ object inputs values from the keyboard.

ANS: `std::cin`

2.9 Write a single C++ statement or line that accomplishes each of the following:

a) Print the message "Enter two numbers".

ANS: `cout << "Enter two numbers";`

b) Assign the product of variables `b` and `c` to variable `a`.

ANS: `a = b * c;`

- c) State that a program performs a payroll calculation (i.e., use text that helps to document a program).

ANS: `// Payroll calculation program`

- d) Input three integer values from the keyboard into integer variables a, b and c.

ANS: `cin >> a >> b >> c;`

2.10 State which of the following are *true* and which are *false*. If *false*, explain your answers.

- a) C++ operators are evaluated from left to right.

ANS: False. Some operators are evaluated from left to right, while other operators are evaluated right to left.

- b) The following are all valid variable names: `_under_bar_`, `m928134`, `t5`, `j7`, `her_sales`, `his_account_total`, `a`, `b`, `c`, `z`, `z2`.

ANS: True.

- c) The statement `cout << "a = 5;"` is a typical example of an assignment statement.

ANS: False. The statement is an output statement. The text `a = 5;` is output to the screen.

- d) A valid C++ arithmetic expression with no parentheses is evaluated from left to right.

ANS: False. Arithmetic operators can appear in any order in an expression, so the expression is `a = b + c * d;` actually evaluates from right to left because of the rules of operator precedence.

- e) The following are all invalid variable names: `3g`, `87`, `67h2`, `h22`, `2h`.

ANS: False. `h22` is a valid variable name. The others are invalid because they each begin with a digit.

2.11 Fill in the blanks in each of the following:

- a) What arithmetic operations are on the same level of precedence as multiplication?

_____.

ANS: division and modulus.

- b) When parentheses are nested, which set of parentheses is evaluated first in an arithmetic expression? _____.

ANS: innermost.

- c) A location in the computer's memory that may contain different values at various times throughout the execution of a program is called a(n) _____.

ANS: variable.

2.12 What, if anything, prints when each of the following C++ statements is performed? If nothing prints, then answer "nothing." Assume `x = 2` and `y = 3`.

- a) `cout << x;`

ANS: 2

- b) `cout << x + x;`

ANS: 4

- c) `cout << "x=";`

ANS: `x=`

- d) `cout << "x = " << x;`

ANS: `x = 2`

- e) `cout << x + y << " = " << y + x;`

ANS: `5 = 5`

- f) `z = x + y;`

ANS: nothing.

- g) `cin >> x >> y;`

ANS: nothing.

- h) `// cout << "x + y = " << x + y;`

ANS: nothing (because it is a comment).

