

Declaration and Initialization

1. Give the declaration for two variables called feet and inches. Both variables are of type int and both are to be initialized to zero in the declaration. Use both initialization alternatives.

```
1 // Normal initialization
2 int feet = 0;
3 int inches = 0;
4
5 // Alternative initialization
6 int feet(0);
7 int inches(0);
```

2. Give the declaration for two variables called count and distance. count is of type int and is initialized to zero. distance is of type double and is initialized to 1.5.

```
1 int count = 0;
2 double distance = 1.5;
```

Assignment

3. Give a C++ statement that will change the value of the variable sum to the sum of the values in the variables n1 and n2. The variables are all of type int.

```
1 sum = n1 + n2;
```

4. Give a C++ statement that will increase the value of the variable length by 8.3. The variable length is of type double.

```
1 length = length + 8.3;
```

5. Give a C++ statement that will change the value of the variable product to its old value multiplied by the value of the variable n. The variables are all of type int.

```
1 product = product * n;
```

6. Which of the following are valid C++ assignment statements? Assume that i, x, and percent are double variables.

- A and C are the only valid statements.

Identifiers and Reserved Words (Keywords)

7. Which of the following are valid identifiers:

- file23 and New_File

8. Give good variable names for each of the following:

- a) speed
- b) pay_rate
- c) max_score

9. Which of the following is a reserved word (keyword) in C++?

- return, void, and int

10. What is the difference between a keyword and a user-defined identifier?

- A keyword has a special and unchanging meaning in C++
- A user-defined identifier is a name made up by a programmer that is not a part of the language that has meaning or usage defined by the programmer

Input and Output

11. Give an output statement that will produce the following message on the screen:

The answer to the question of
Life, the Universe, and Everything is 42.

```
1 cout << "The answer to the question of\n Life, the Universe, and  
   Everything is 42.\n";
```

12. Give an input statement that will fill the variable the_number (of type int) with a number typed in at the keyboard. Precede the input statement with a prompt statement asking the user to enter a whole number.

```
1 cout << "Please enter a whole number:\n";  
2 cin >> the_number;
```

13. Give an output statement that produces the new-line character and a tab character.

```
1 cout << "\n\t";
```

14. What is the output of the following program lines when embedded in a correct program that declares all variables to be of type char?

```
1 a = 'b';  
2 b = 'c';  
3 c = a;  
4 cout << a << b << 'c';
```

The program lines would output:

bcc

Math in C++

15. Convert each of the following mathematical formulas to a C++ expression:

a.) $3x$

```
1 3 * x;
```

b.) $3x + y$

```
1 3 * x + y;
```

c.) $\frac{x + y}{2}$

```
1 (x + y) / 2;
```

d.) $7z^3 + 2$

```
1 7 * z * z * z + 2;
```

16. Evaluate: $30 - 5 / 2.0$

27.5

17. Evaluate: $19 + 7 \% 3 - 4$

16

Writing Simple Programs

18. Write a complete C++ program that writes the phrase “Hello world” to the screen. The program does nothing else.

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     cout << "Hello World";
7     return 0;
8 }
```

19. Write a complete C++ program that reads in two whole numbers and outputs their sum. Be sure to prompt for input, echo input, and label all output.

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     int num1, num2, sum;
7
8     // Get whole number 1
9     cout << "Enter in a whole number:\n";
10    cin >> num1;
11
12    // Get whole number 2
13    cout << "Enter in another whole number:\n";
14    cin >> num2;
15
16    // Calculate their sum
17    sum = num1 + num2;
18
19    // Display the sum
20    cout << "The sum of both whole nubers is " << sum;
21
22    return 0;
23 }
```

20. Write a program that contains statements that output the value of five or six variables that have been declared, but not initialized. Compile and run the program. What is the output? Check the program on a different environment or with another student. Do you get the same results? Explain.

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     bool boolean_value;
7     int whole_number;
```

```
8     double decimal_number;  
9     char character;  
10    string example_string;  
11  
12    cout << "Boolean: " << boolean_value << "\n";  
13    cout << "Integer: " << whole_number << "\n";  
14    cout << "Double : " << decimal_number << "\n";  
15    cout << "Char   : " << character << "\n";  
16    cout << "String : " << example_string << "\n";  
17  
18    return 0;  
19 }
```

Running the program 3 times results in:

```
1 Boolean: 0  
2 Integer: 1  
3 Double : 2.14438e-314  
4 Char   :  
5 String :
```

```
1 Boolean: 0  
2 Integer: 1  
3 Double : 2.14584e-314  
4 Char   :  
5 String :
```

```
1 Boolean: 0  
2 Integer: 1  
3 Double : 2.15885e-314  
4 Char   :  
5 String :
```

The value of the double is different between successive runs. This is because reading the value of an uninitialized variable results in undefined behavior due to it reading leftover 1's and 0's left in the memory.