Syntax

For each of the following, correct the syntax errors.

1. if x > 25.0

{

y = x

}

else

{

y = z;

}

2. if (x >> 0)

{

cout << "positive " << x << endl;

}

else

{

cout << "negative " << x << endl;

}

Determining output of C++ code involving if statements

What is the output of each of the following statements? Assume that

x = 5, y = 2, z = 10, and temp = 0

3. if (y >= x)

y = z;

cout << x << " " << y << " " << z << endl;

5 2 10

4. if (y >= x)

{

y = z;

cout << x << " " << y << " " << z << endl;

}

no output

5. if (z < y)

temp = x; //temp = 5

x = z; //x = 10

z = temp; //z = 0

cout << x << " " << y << " " << z << endl;

output: 10 2 0

6. if (z > y)

{

temp = x; //temp = 5

x = z; //x = 10

z = temp; //z = 5

}

cout << x << " " << y << " " << z << endl;

output: 10 2 5

7. if (x >= 6)

cout << x + y << endl; //doesn't run

cout << x + y << endl;

output: 7

8. if (x + y > z) //x + y = 7

x = y + z;

else

x = y - z; //x = -8

cout << x << " " << y << " " << z << endl;

output: -8 2 10

Writing simple C++ code involving if statements

Write if statements for the following problems. Use meaningful variable names. You can assume that the variables you use are declared and have been initialized.

9. If a variable angle is equal to 90 degrees, print the message "right angle”. Otherwise print the message "not a right angle".

if (angle == 90)

{

cout << "right angle" << endl;

}

else

{

cout << "not a right angle" << endl;

}

10. If the difference between variables temp1 and temp2 is more than 2.3, set the variable approx to (temp1 - temp2) \* factor.

if (temp1 - temp2 > 2.3)

{

result = (temp1 - temp2) \* factor

}

11. A student at State U is ready to graduate if he or she has completed at least 122 credits. Write the statements to test whether a student is ready to graduate, and print a message telling whether he or she is ready.

if (student\_credits >= 122)  
{

cout << "This student has enough credits to graduate." << endl;

}

else  
{

cout << "This student does not meet the credit requirement." << endl;  
}

12. Consider a quadratic expression, say x^2 - x - 2 (x - 2)(x + 1) x = 2, -1   
Describing where this quadratic is positive (that is, greater than 0), involves describing a set of numbers that are either less than the smaller root (which is -1) or greater than the larger root (which is +2). Write two different C++ Boolean expressions that are true when this formula has positive values.

if (x > 2) || (x < -1)

13. Write an if-else statement that outputs the word “High” if the value of the variable score is greater than 100 and “Low” if the value of score is at most 100? The variables are of type int.  
if (score > 100)   
{  
 cout << "High" << endl;  
}  
else if (score <= 100)  
{  
 cout << "Low" << endl;  
}

14. Write an if-else statement that outputs the word “Warning” provided that either the value of the variable temperature is greater than or equal to 100, or the of the variable pressure is greater than or equal to 200, or both. Otherwise, the if-else statement outputs the word “OK”. The variables are of type int.  
if ((temperature >= 100) ||(pressure >= 200))  
{  
 cout << "Warning" << endl;  
}  
else  
{  
 cout << "OK" << endl;  
}

Evaluating Boolean Expressions

15. Determine the value, *true* or *false*, of each of the following Boolean expressions, assuming that the value of the variable count is 0 and the value of the variable limit is 10. Give your answer as one of the values *true* or *false*. It’s possible the outcome could be a runtime error. Hint: For some of these, don’t forget about short circuit evaluation.

* (count == 0) && (limit < 20) true
* count == 0 && limit < 20 true
* (limit > 20) || (count < 5) true
* !(count == 12) true
* (count == 1) && (x < y) false
* (count < 10) || (x < y) true
* !( ((count < 10) || (x < y)) && (count >= 0) ) true
* ((limit/count) > 7) || (limit < 20) runtime error
* (limit < 20) || ((limit/count) > 7) true
* ((limit/count) > 7) && (limit < 0) runtime error
* (limit < 0) && ((limit/count) > 7) false
* (5 && 7) + (!6) (1 + 0) true