Eli Griffiths
Wenchi Chao
Checked by groups 10 and 11

If Statement Worksheet

Syntax

For each of the following, correct the syntax errors.

1.
```
if x > 25.0
   y = x
else
   y = z;
```

```
if (x > 25.0)
    y = x;
else
    y = z;
```

2.
```
if (x >> 0);
   cout << "positive " << x << endl;
else;
   cout << "negative " << x << endl;
```

```
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;
   x = z;
z = temp;
cout << x << "  " << y << "  " << z << endl;
```

10 2 0

6.
```
if (z > y)
{
   temp = x;
   x = z;
   z = temp;
}
cout << x << "  " << y << "  " << z << endl;
```

10 2 5

7.
```
if (x >= 6)
cout << x + y << endl;
cout << x + y << endl;
```

7

8.
```
if (x + y > z)
   x = y + z;
else
   x = y - z;
cout << x << "  " << y << "  " << z << endl;
```

-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" << "\n";
else
    cout << "not a right angle" << "\n";
```

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)
    approx = (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 (credits >= 122)
    cout << "You're ready to graduate" << "\n";
else
    cout << "You're not ready to graduate" << "\n";
```

12. Consider a quadratic expression, say

$$x^2 - x - 2$$

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.

```
x * x – x – 2 > 0
(x < -1) || (x > 2)
```

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” << “\n”;
else
    cout << “Low” << “\n”;
```

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” << “\n”;
else
    cout << “OK” << “\n”;
```

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.

a. `(count == 0) && (limit < 20)` **true**
b. `count == 0 && limit < 20` **true**
c. `(limit > 20) || (count < 5)` **true**
d. `!(count == 12)` **true**
e. `(count == 1) && (x < y)` *error: x and y are not defined*
f. `(count < 10) || (x < y)` *error: x and y are not defined*
g. `!( ((count < 10) || (x < y)) && (count >= 0) )` *error: x and y are not defined*
h. `((limit/count) > 7) || (limit < 20)` *runtime error: division by 0 is undefined*
i. `(limit < 20) || ((limit/count) > 7)` **true**
j. `((limit/count) > 7) && (limit < 0)` *runtime error: division by 0 is undefined*
k. `(limit < 0) && ((limit/count) > 7)` **false**
l. `(5 && 7) + (!6)` **true**