“Introduction to C++ Programming” Online Course

Pretest/Posttest for Class 4

1. Given the following variable definitions, what is the result of each of the following test conditions? Mark each result with either t (for true or 1) or f (for false or 0).
   
   ```
   int a = 20, b = 10, c = 42;
   ```
   
   a. if (a > 0) t
   b. if (a > b) t
   c. if (c == 0) t
   d. if (c == a) t
   e. if (c + a > b) t
   f. if (a / 2 == b) t
   g. if (a > c / b) t
   h. if (a > 0 && b < 10) t
   i. if (a > 0 && b >= 10) t
   j. if (b > 0 || c < 10) t
   k. if (b > 0 || c >= 10) t
   l. if (b) t
   m. if (!a) t

2. Using the definitions below, what is the output of the following code?
   ```
   int a = 20, b = 10, c = 42;
   if (c <= 42)
       cout << "Hi There\n";
   else
       cout << "So long\n";
   ```

3. Using the definitions below, what is the output of the following code?
   ```
   int a = 20, b = 10, c = 42;
   ```
   ```
   int t = a > b ? c : c + 5;
   cout << t;
   ```

4. Correct all the errors in the following coding. The object is to display the fuel efficiency of a car based on the miles per gallon it gets — its mpg.
   ```
   if (mpg > 25.0) {
       cout << Gas Guzzler\n";
   else
       cout << "Fuel Efficient\n";
   ```
In the next three problems, repair the If-Then-Else statements. However, maintain the spirit of each type of If-Then-Else style. Do not just find one way to fix it and copy that same “fix” to all three problems. Rather, fix each one maintaining that problem’s coding style.

5. Correct all the errors in the following coding. The object is to display “equilateral triangle” if all three sides of a triangle are equal.
   
   ```
   if (s1 == s2 == s3);
   { 
   cout << "equilateral triangle\n"; 
   } 
   else; 
   cout >> "not an equilateral triangle\n"; 
   ```

6. Correct all the errors in the following coding. The object is to display “equilateral triangle” if all three sides of a triangle are equal.
   
   ```
   if (s1 == s2) 
   if (s2 == s3) 
   cout << "equilateral triangle\n";
   cout >> "not an equilateral triangle\n";
   ```

7. Correct all the errors in the following coding. The object is to display “equilateral triangle” if all three sides of a triangle are equal.
   
   ```
   if (s1 == s2) { 
   if (s2 == s3) { 
   cout << "equilateral triangle\n";
   } 
   else { 
   cout >> "not an equilateral triangle\n";
   } 
   } 
   ```

8. Correct this grossly inefficient set of decisions so that no unnecessary decisions are made.
   
   ```
   if (department == 1)
   cout << "Automotive\n";
   if (department == 2)
   cout << "Appliances\n";
   if (department == 3)
   cout << "Accounting\n";
   if (department == 4)
   cout << "Toys\n";
   ```
if (department == 5)
    cout << "Housewares\n";
if (department == 6)
    cout << "Ladies Apparel\n";
if (department == 7)
    cout << "Children’s Cloths\n";

9. Correct this non-optimum solution. Consider all of the numerical possibilities that the user could enter for variable number. Rewrite this coding so that the program does not crash as a result of the numerical value entered by the user. You may display appropriate error messages to the user. Ignore the possibility of the user entering in nonnumerical information by accident.

```cpp
double number;
double root;
derobece reciprocal;
cout << "Enter a Number: ";
cin >> number;
root = sqrt (number);
reciprocal = 1 / number;
cout << number << " square root is " << root
    << " reciprocal is " << reciprocal << endl;
```

10. Correct this inherently unsound calculation so that no runtime crashes can occur.

```cpp
double x;
double y;
cout << "Enter a Number: ";
cin >> x;
y = x * x + 1. / (x * x * x + 1.) - 1.;
if (!y || y == x) {
    cout << "x’s value results in an invalid state.\n"
    return 1;
}
```