

## CSCI 201.001 Fall 2004 Quiz 2 **Solution**

26 October, 2004

### Question 1 (13 points)

Start by completing the incomplete constructor and two of the incomplete methods of the `EquilateralTriangle` method. Because capitalization is significant in Java, you need to write your answer in such a way that the case of your letters is **very** clear.

<pre>public EquilateralTriangle(double length,                              int angle) {     <b>setSide(length) ;</b>     <b>setOrientation(angle) ;</b> }</pre>
<pre>public void setSide(double length) {     <b>side = length ;</b> }</pre>
<pre>public int getOrientation() {     <b>return (orientation) ;</b> }</pre>

### Question 2 (10 points)

Which of the following are legal uses of the `EquilateralTriangle` constructor?

<pre>EquilateralTriangle T1 = new EquilateralTriangle(3.0);</pre> <p><b>Illegal: No constructor has one argument</b></p>
<pre>EquilateralTriangle T2 = new EquilateralTriangle() ;</pre> <p><b>Legal: Uses the default constructor with no arguments</b></p>
<pre>EquilateralTriangle T3 = new EquilateralTriangle(3.0,9.0);</pre> <p><b>Illegal: The second argument must a Java int</b></p>
<pre>EquilateralTriangle T4 = new EquilateralTriangle(3.0, 9) ;</pre> <p><b>Legal: Matches the double, int constructor</b></p>

### Question 3 (15 points)

Assuming that T1 and T2 are both objects of the class EquilateralTriangle, which of the following are legal Java statements?

T1.setSide(5) ;	<b>Legal: setSide accepts a double, but a “widened” int is good enough</b>
double s = EquilateralTriangle.area(4.0) ;	<b>Illegal: area method takes no arguments and is not static</b>
int a = T1.getA() ;	<b>Illegal: no getA method</b>
int alpha = T1.getOrientation(4.0) ;	<b>Illegal: getOrientation method takes no arguments</b>
T1.setOrientation(5.0) ;	<b>Legal: One setOrientation method does use a double</b>
boolean b = EquilateralTriangle.validSide(3.0) ;	<b>Legal: validSide is a static method that uses a double</b>

### Question 4 (15 points)

Assume that V and X are Java double variables and that the value of V is 5.0 and the value of X is 10.0, specify which of the following are legal expressions in Java. If the expression is legal, write its value in the column at the right.

V =< X	<b>Illegal: no =&lt; operator</b>
V < 7.0 < X	<b>Illegal: Same as (V &lt; 7.0) &lt; X</b>
(V < 13.3)    (13.3 < X)	<b>true</b>
V + 13.0 < X	<b>false: Same as (V + 13.0) &lt; X</b>
V < 7.0 != X < 15.0	<b>false: Same as (V&lt;7.0) != (X&lt;15.0)</b>
V < 13.3 && 13.3 < X	<b>false: Same as (V&lt;13.3)&amp;&amp;(13.3&lt;X)</b>

### Question 5 (27 points)

In this question assume that I, J, and K are Java `int` variables. Below is a Java conditional statement that sets J to 5, when K is positive; and sets J to 17 otherwise.

```
if (K > 0) {  
    J = 5 ;  
}  
else {  
    J = 17 ;  
}
```

Now write similar short sections of Java code for each of the following three problems.

#### Question 5A

Set J to I, when I does not equal K; and set K to 0, otherwise.

```
if (I != K) {  
    J = I ;  
}  
else {  
    K = 0 ;  
}
```

#### Question 5B

Set K to 13, when both I and J are less than 0; and set both I and K to 17, otherwise.

```
if ((I < 0) && (J < 0)) {  
    K = 13 ;  
}  
else {  
    I = 17 ;  
    K = 17 ;  
}
```

#### Question 5C

Without changing either I or J; set K to a number which is equal to neither I nor J.

*This solution given by Keith Schneider.*

```
K = I + 1 ;  
if (K == J) {  
    K = J + 1 ;  
}
```

### Question 6 (10 points)

The following piece of code is *supposed* to print a single line according to the value of a Java `int` variable `N`. However, the code doesn't work correctly. Sometimes it prints more than one line. Describe what's wrong with the code and show how it can be fixed.

```
if (N > 100)
    System.out.println("Too big") ;
else if (N >= 60)
    System.out.println("Pass") ;
else if (N <= 60) (N >= 0)
    System.out.println("Fail") ;
else if (N < 0)
    System.out.println("Too small") ;
```

### Question 7 (10 points)

What is printed by the following two rather pointless sections of Java code?

```
int N = 5 ;
while (N < 30) {
    System.out.println(N) ;
    N = 3*N + 1 ;
}
```

5  
16

```
char initial = 'E' ;
switch(initial) {
    case 'E':
        System.out.println("East") ;
    case 'W':
        System.out.println("West") ;
    default:
        System.out.println("Unknown") ;
}
```

East  
West  
Unknown