University of Stirling Computing Science and Mathematics

CSCU9A1

Tutorial 2 Solutions

These questions are taken from Chapters 2 and 3 of Java for Everyone (2nd edition) and the companion website for that book.

1. Answer the following questions from the review exercises in Chapters 2 of Java for Everyone.

   R2.1 :
   What is the value of `mystery` after this sequence of statements?

   ```
   int mystery = 1;
   mystery = 1 - 2 * mystery;
   mystery = mystery + 1;
   ```

   0

   R2.2:
   What is wrong with the following sequence of statements?

   ```
   int mystery = 1;
   mystery = mystery + 1;
   int mystery = 1 - 2 * mystery;
   ```

   There will be a compile-time error because `mystery` is declared twice.

   R2.3 (part a)
   Write the following mathematical expression in Java.
   $s = s_0 + v_0 t + \frac{1}{2} g t^2$

   ```
   s = s0 + v0 * t + 0.5 * g * Math.pow(t,2);
   ```

   R2.4 (part a)
   Write the following Java expression in mathematical notation

   ```
   dm = m * (Math.sqrt(1 + v / c) / Math.sqrt(1 - v / c) - 1);
   ```

   $$dm = m\left(\frac{\sqrt{1+\frac{v}{c}}}{\sqrt{1-\frac{v}{c}}} - 1\right)$$

R2.5 What are the values of the following expressions? In each line, assume that

```
double x = 2.5;
double y = -1.5;
int m = 18;
int n = 4;
```

The underlining in the solutions shows the next sub-expression to be evaluated at each step.

a. `x + n * y - (x + n) * y`

```
2.5 + 4 * -1.5 - (2.5 + 4) * -1.5
2.5 + 4 * -1.5 - 6.5 * -1.5
2.5 + -6.0 - 6.5 * -1.5
2.5 + -6.0 - -9.75
-3.5 - -9.75
6.25
```

b. `m / n + m % n`

```
18 / 4 + 18 % 4
4 + 18 % 4
4 + 2
6
```

R2.7 What are the values of the following expressions? In each line, assume that

```
String s = "Hello";
String t = "World";
```

a. `s.length() + t.length()`

```
10
```

b. `s.substring(1,2)`

```
"e"
```

c. `s.substring(s.length() / 2, s.length())`

```
"Hello".substring("Hello".length() / 2, "Hello".length())
"Hello".substring(5 / 2, "Hello".length())
"Hello".substring(2, "Hello".length())
"Hello".substring(2, 5)
"llo"
```

d. `s + t`

"HelloWorld"

e. `t + s`

"WorldHello"

2. You want to know how many feet are in 3.5 yards, and how many inches are in 3.5 yards. You write the following program for that purpose:

```
public class DistanceConverter
{
   public static void main(String[] args)
   {
      double yards = 3.5;
      double feet = yards * 3;
      double inches = feet * 12;

      System.out.println(yards + "yards are" + feet + "feet");
      System.out.println(yards + "yards are" + inches + "inches");
   }
}
```

   The problem with the program above is that using "magic numbers" makes it hard to maintain and debug. Modify the program so that it uses constants to improve legibility and make it easier to maintain.

Answer:

```
public class DistanceConverter
{
   public static void main(String[] args)
   {
      final int FEET_IN_A_YARD = 3;
      final int INCHES_IN_A_FOOT = 12;

      double yards = 3.5;
      double feet = yards * FEET_IN_A_YARD;
      double inches = feet * INCHES_IN_A_FOOT;

      System.out.println(yards + "yards are" + feet + "feet");
      System.out.println(yards + "yards are" + inches + "inches");
   }
}
```

3. Can you see any problems with the format of the output from the `DistanceConverter` program? What does the program output? What changes would you make to improve the readability of the output?

Answer:

```
3.5yards are10.5feet
3.5yards are126.0inches
```

The print statements need to be changed to make the output more readable:

```
System.out.println(yards + " yards are " + feet + " feet");
System.out.println(yards + " yards are " + inches + " inches");
```

4. Answer the following questions from the review exercises in Chapter 3 of Java for Everyone:

   R3.1
   What is the value of each variable after the `if` statement?

   a.
   ```
   int n = 1; int k = 2; int r = n;
   if (k < n) { r = k; }
   ```

   n has value 1 k has value 2 r has value 1

   b.
   ```
   int n = 1; int k = 2; int r;
   if (n < k) { r = k; }
   else { r = k + n; }
   ```

   n has value 1 k has value 2 r has value 2

   c.
   ```
   int n = 1; int k = 2; int r = k;
   if (r < k) { n = r; }
   else { k = n; }
   ```

   n has value 1 k has value 1 r has value 2

   d.
   ```
   int n = 1; int k = 2; int r = 3;
   if (r < n + k) { r = 2 * n; }
   else { k = 2 * r; }
   ```

   n has value 1 k has value 6 r has value 3