CSSE 220 – Object-Oriented Software Development
Rose-Hulman Institute of Technology

Worksheet 14

Name (Print):_____ Section:_____

1. **Recursion helper with arrays (in-class coding)**
   Complete the method below so it returns the sum of all values in `arr`. Use a helper method so you can track the current index.

```
1  public static int sumArr(int[] arr) {
2      // TODO: start the recursion at index 0
3      ________________________________;
4  }
5
6  private static int sumArrHelper(int[] arr, int index) {
7      // TODO: base case (when index is past the last element)
8      ________________________________;
9
10     // TODO: recursive case
11     ________________________________;
12 }
```

2. **CodingBat (Recursion 2) practice**
   Complete these problems on CodingBat. Check them off when done:

   ☐ `array6`
   ☐ `array11`
   ☐ `array220`

3. **Relationships: IS-A vs HAS-A vs DEPENDS-ON**
   Select from the following: **IS-A**, **HAS-A**, **DEPENDS-ON**

   (a) `class Dog extends Animal`: _____
   (b) `class Car implements Drivable`: _____
   (c) `class Car { private Engine engine; }`: _____
   (d) `class Vet { void examine(Dog d) {...}}`: _____

4. **Definition (Inheritance)**
   Inheritance is a mechanism when a _____ automatically gains the fields and methods of its _____ by using the `extends` keyword.

5. **Java class hierarchy**
   What is the name of the topmost class in Java? _____

6. **What can a subclass do? (circle/select all correct)**

(a) A subclass inherits fields and methods from its superclass, but private members are not directly accessible.

(b) A subclass inherits constructors from its superclass.

(c) A subclass must call (explicitly or implicitly) a superclass constructor.

(d) A subclass can override public and protected methods (if they are not `final` or `static`).

(e) A subclass can override private methods.

7. **Using `super`**
   Fill in the correct Java code.

   (a) Call a superclass method `eat()` from inside an overriding method:

   _____

   (b) Call a superclass constructor `Animal(String name)` from a subclass constructor:

   _____

8. **Overriding: what prints?**
   Write the output for each line:

```
1  class Animal {
2      void move() { System.out.println("Animal is moving."); }
3      void eat()  { System.out.println("Animal is eating."); }
4  }
5
6  class Dog extends Animal {
7      @Override void move() {
8          System.out.println("Dog is running.");
9      }
10     void bark() { System.out.println("Dog is barking."); }
11 }
12
13 public class Geeks {
14     public static void main(String[] args) {
15         Dog d = new Dog();
16         d.move();  // Output 1: ___________________
17         d.eat();   // Output 2: ___________________
18         d.bark();  // Output 3: ___________________
19     }
20 }
```

9. **Before You Leave**
   Write one question you still have about today's topic.