

15-121 Fall 2022 Quiz 5

Up to 25 minutes. Show your work. No calculators, no notes, no books, no computers, no other people.

1. (6 points) **Code Tracing:** Consider the following two classes, and indicate what is printed when the main method of the B class is executed. Place your answer (and nothing else) in the box under the code.

```
public class A {
    protected int a;
    protected int b;

    public A(int a, int b) {
        this.a = a;
        this.b = b;
    }

    public int mystery() {
        return a+b;
    }

    public void printVals() {
        System.out.println("Y");
    }
}

public class B extends A {
    public B(int a, int be) {
        super(be, a);
        System.out.println("P1: " + a);
        System.out.println("P2: " + b);
        b = a - 2;
        a = be + 2;
        System.out.println("P3: " + this.a);
        System.out.println("P4: " + this.b);
    }

    public void printVals() {
        System.out.println("R");
    }

    public static void main(String[] args) {
        B bob = new B(15, 20);
        System.out.println(bob.mystery());

        A andrew = bob;
        andrew.printVals();
    }
}
```

2. Free Response: Animal Names

Consider the following code that uses the `AnimalNames` class:

```
// Constructor takes a string of comma separated names
AnimalNames a = new AnimalNames("Fido,Jack,Salk");
System.out.println(a);
a.addName("Izzy");
System.out.println(a);
a.removeName("Jack");
System.out.println(a);
```

When executed, this code provides the following output:

```
AnimalNames: Fido Jack Salk
AnimalNames: Fido Jack Salk Izzy
AnimalNames: Fido Salk Izzy
```

You will write the code for four methods of this class. Other than those four methods, you cannot modify other things about the class (such as by adding new methods or instance variables). The entire class can be seen here:

```
public class AnimalNames {
    private ArrayList<String> names;

    public AnimalNames(String nameStr) {
        // Your code here
    }

    public void removeName(String name) {
        // Your code here
    }

    public void addName(String name) {
        // Your code here
    }

    protected String getNames() {
        // Your code here
    }

    public String toString() {
        return "AnimalNames: " + getNames();
    }
}
```

This problem continues on the next page. Do not write any answers on this page.

(a) (4 points) Write the Constructor:

```
public AnimalNames(String nameStr) {
```

(b) (1 point) Write the following method:

```
public void removeName(String name) {
```

(c) (1 point) Write the following method:

```
public void addName(String name) {
```

(d) (2 points) Write the following method:

```
protected String getNames() {
```

3. (6 points) **Free Response:** *Really* Cute Dog Names

The class `CuteDogNames` is a child of the `AnimalNames` class, but it only allows names that would be *really* cute for a dog. A cute dog name is defined as one that starts with “Mister”.

Consider the following code that uses the `CuteDogNames` class:

```
CuteDogNames c = new CuteDogNames("MisterPoofy,Jack,Salk,MisterRuff,MisterFuzzyWuzzy");
System.out.println(c);
c.addName("Izzy");
c.addName("MisterSnuffles");
System.out.println(c);
```

When executed, this code provides the following output:

```
CuteDogNames: MisterPoofy MisterRuff MisterFuzzyWuzzy
CuteDogNames: MisterPoofy MisterRuff MisterFuzzyWuzzy MisterSnuffles
```

Write the `CuteDogNames` class.

Important Notes and Hints:

1. Your `CuteDogNames` class may not declare any instance variables.
2. The `names` variable in `AnimalNames` is `private`, so you can't directly access it from the child. You also aren't allowed to change the visibility of `names`, or add any new methods or instances variables to `AnimalNames`. That means solving this problem will require some creative problem solving.
3. The first line of `AnimalNames`'s constructor should be a call to the constructor of the superclass. (Java only allows the super class constructor to be called from the *first* line of the child class constructor. It can't be called later in the constructor.)
4. The `String` class has a `startsWith` method that returns a `boolean` and does what it sounds like.