**Homework 4**

### **CMP 326: Programming Methods II**

Lehman College, City University of New York

Fall 2019

### [**Homework 4-1 Textbook Section 9.9**](https://learn.zybooks.com/zybook/CUNYCMP326Fall2019/chapter/9/section/9)

Take the **Animal** class from the previous [**Homework 3-1**](https://drive.google.com/open?id=1z062yTrBLJ7cDP7SUp0wLWI4ZstO7y8I0erEbVu1x84) and add the following to it:

Animal

* height : double
* Animal(String, int, double, double, char) //name, birthYear, weight, height, gender
* getHeight() : double
* setHeight(double) : void //if input value is negative set height to -1
* calculateBMI() : double //using pounds and inches **BMI = weight / (height2) \* 703**
* printDetails() : void // prints Animal attributes in the following format:

// "Name: %20s | Year of Birth: %4d | Weight: %10.2f | Height: %10.2f | Gender: %c\n"

* toString() : String @Override //see Note1 for format
* equals(Object) : boolean @Override //see Note2 for equality checks

**NOTE1**:

Format the data for the **toString()** method as

"Name: %20s | Year of Birth: %4d | Weight: %10.2f | Height: %10.2f | Gender: %c\n"

**NOTE2:**

For the **equals(Object)** method, two Animal objects are considered equal if

both have the same values for **birthYear, gender, name**

and if their **weight**, **height** values are within **.5** of the other’s **weight**, **height** values

### [**Homework 4-2 Textbook Section 9.10**](https://learn.zybooks.com/zybook/CUNYCMP326Fall2019/chapter/9/section/10)

Create a class to represent a **Mammal** object that inherits from (**extends**) the **Animal** class.

Use the description provided below in UML.

Mammal

* tailLength : double
* numLegs : int
* Mammal() //default values for tailLength and numLegs should be 0
* Mammal(double, int)
* Mammal(String, int, double, double, char, double, int) //pass values to parent’s overloaded constructor

//and assign valid values to tailLength, numLegs or -1 if invalid

* setTailLength(double) : void //if the value is invalid set tailLength to -1
* getTailLength() : double
* setNumLegs(int) : void //if the value is invalid set numLegs to -1
* getNumLegs() : int
* printDetails() : void @Override //see Note1:
* toString() : String @Override //see Note1:
* equals(Object) : boolean @Override //see Note2:

**NOTE1:**

For both the **printDetails()** and **toString()** methods include the data from the Animal class, as well as Mammal attributes. Format the Mammal data as “Mammal: Tail Length: %10.2f | Number of Legs: %4d\n”

**Hint:** Use super.printDetails() and super.toString()

**NOTE2:**

For the **equals(Object)** method, two Mammal objects are considered equal if the

**parent class’s equals** method returns true,

if their **numLegs** are equal,

and their **tailLength** values are within **.1** of each other

### [**Homework 4-3 Textbook Section 9.11**](https://learn.zybooks.com/zybook/CUNYCMP326Fall2019/chapter/9/section/11)

Create a class to represent a **Dog** object that inherits from (**extends**) the **Mammal** class. Use the description provided below in UML. Create a nested class named **DogToy** within the Dog class

**Dog**

**DogToy** //an inner class (non-static)

* dogToyName: String
* dogToyRating: int

+ DogToy(String, int)

+ toString() : String

**Dog** (continued)

* toys : DogToy [ ]
* numDogToys : int
* Dog() //a dog with zero DogToys but array capacity of 10
* Dog(double, int) //pass values to parent’s overloaded constructor
* Dog(String, int, double, double, char, double, int) //pass values to parent’s overloaded constructor
* addDogToy(String,int) //uses the DogToy constructor, do not resize array
* getDogToyAsString(int) : String
* getNumDogToys() : int
* getDogToyName(int) : String
* getDogToyRating(int) : int
* getHighestDogToyRating(): int //should return the highest rating, or 0 if the array is empty
* printDetails() : void @Override //see NOTE below for formatting
* toString() : String @Override //see NOTE below for formatting

**NOTE1:**

For both the **printDetails()** and **toString()** methods include the data from the Animal class, as well as Mammal attributes.

Format the Dog data as “Dog: Number of Toys: %4d | Dog Toys:\n” followed by the

formatted DogToy String for each toy on its own line as “DogToy: Toy Name: %20s | Rating %4d\n”

**NOTE2:**

In the Dog class do not override the equals method inherited from the Mammal class

**NOTE3:**

In the Dog class do not resize the toys array. The max capacity should remain 10

**Hint1:** Use super.printDetails() and super.toString()

**Hint2:** DogToy is immutable (i.e. no setter methods, it’s values cannot be changed)

**Hint3**: DogToy belongs to Dog only, which is why it is a nested class

**Hint4:** toString() inside the DogToy should be formatted as “DogToy: Toy Name: %20s | Rating %4d\n”

**Hint5:** toString() inside the Dog should iterate through toys and append the String returned from each DogToy

### 

**This work must be completed in your textbook**  [**ZYBooks -- CMP-326: Programming Methods I**](https://learn.zybooks.com/zybook/CUNYCMP326Fall2019/)**I**

**No other forms of submission will be accepted.**