CS161: Introduction to Computer Science  
Lab Assignment 7

In today’s lab, you will practice writing multiple classes that interact with one another.

You will write an Organism class that uses the Stomach and Brain class you wrote in lab 6. If you did not finish these classes in lab, you will need to finish them before you can move on.

1 Organism

Create a new Java class named Organism. An organism should have a brain, a stomach, and a name. These represent the instance variables for the organism. Add a constructor to initialize your instance variables. You should not copy-and-paste the code from the Brain and Stomach class inside the organism. This defeats the purpose of encapsulation. Instead, you should use the Brain and Stomach classes inside the organism.

Next, add the following methods to your class. Most of these methods will be one-line methods that simply call the methods in the Stomach and Brain class.

- public void eat(int amount) – this method should cause the organism to eat the specified amount of food
- public void digest() – this method should cause the organism to digest a random amount of food
- public void think(String newThought) – this method should cause the organism to think a thought
- public void remember() – this method should cause the organism’s current thought to be stored in its memory
- public void recall() – this method should cause the organism’s memory to be loaded back into its current thought
- public void wakeUp() – this method should cause the organism to wake up
- public void sleep() – this method should cause the organism to go to sleep
- public String toString() – this method should return a string representing the state of the organism
- public boolean equals(Organism other) – this method returns true if both organisms have the same name and false otherwise

2 Reproduction

Now we’ll add the ability for our organisms to reproduce. In reality, if our organism could not reproduce, it would quickly become extinct.

1. Start by adding asexual reproduction to the Organism class. Create a method inside the Organism class called asexuallyReproduce() that returns a new organism. The name of the new organism should somehow be related to the name of the original organism. For example, if your organism is named “ozzy” then the new organism might be named “ozzy1”. The next time the method is called, the new organism has the name “ozzy2”. Then “ozzy3”, “ozzy4”, etc.
2. Continuing with this idea, how could you implement sexual reproduction? Create a method called \texttt{sexuallyReproduce()} that takes as input another organism and returns a new organism whose name is a combination of both of its parents. This method is similar to the \texttt{equals()} method in that it takes as input another organism and accesses that organisms’ instance variables.

Once your \texttt{Organism} class is written and compiles, you can start creating actual organisms. Create a new class called \texttt{OrganismController} that has a \texttt{main()} method. Inside the \texttt{main()} method, ask the user to type in a name and create an organism with that name. Have the organism perform at least 5 different actions.

Next, call both \texttt{assexuallyReproduce()} and \texttt{sexuallyReproduce()} to create children and grandchildren for your original organism. Add enough print statements so that when I execute your \texttt{main()} method, it is clear what is happening.

3 Style Guide

Be sure to go through the list below carefully and make sure your code adheres to the style guide:

- Delete any unused instance variables
- If an instance variable isn’t used in multiple methods, then demote it to a local variable inside the method where it is actually used.
- All instance variables should be initialized in the constructor
- You have a Javadoc comment at the top of the class with a brief description (written in full English sentences), you and your partner’s name, and the date.
- Each method you write has a Javadoc comment with appropriate \texttt{@param} and \texttt{@return} statements
- All variable names are lower cased (remember, only classes are capitalized in Java)
- Use inline comments (//) to explain any complicated code

4 Submitting your assignment

Please make sure to rename your folder before zipping. You should rename your folder using both of your first and last names. For example, \texttt{lab7_John_Doe_Jane_Doe}.

Submit your zipped folder via Canvas.