CS445 - Spring 2015 : Programming Assignment #1

Due on Tuesday, February 3, 2015

Virgil Bistriceanu

Tony Forlini

Contents

1 Introduction

This assignment is aimed at understand the object oriented paradigm and all the underlying mechanisms related to the object oriented programming. We will study the use of abstract classes, concrete classes, interfaces and polymorphism.

Our program is implemented using Java language and is able to run on a fresh version of ubuntu following the attached Readme file.

2 Creating a class hierarchy

First we need to create the hierarchy of Things and Creatures. Things is the concrete class representing any animal with two attributes: a name and the last thing eaten by the Thing. It also defines the toString() method which prints out the description of the object.

Then we construct the subclass Creature which implements the behavior of an animal with the methods eat(), move() and whatDidYouEat(). However Creature is an abstract class, because some of its methods are too specific and needs to be implemented in further sub-classes.

This class represents a generalization which cannot be instantiated. Consequently the previous methods aren't implemented within Creature, but will be defined in the more precise sub-classes.

3 Introducing Interfaces for implementing Additional Behavior

Then we implemented the sub classes responsible for defining more precisely the behavior of the creatures: ant, fly and bat. These classes implement the abstract class Creature. We also create the interface Flyer which defines the ability to Fly of some creature.

This behavior is implemented within an interface because it is susceptible to be useful in case we need to add another new flying creature to our program. This new creature would only have to implement its own version of the Flyer interface with polymorphism.

At this point we also created the TestCreature class which is responsible for sanity check of the creature behaviors by creating an array of Things and an array of Creatures and displaying the different instances of those classes.

4 Completing the Thing-Creature Framework

Finally in this part, we finish the stubbed methods of the sub classes by implementing the following three functions: Creature.eat() Creature.move() and Creature.whatDidYouEat().

We add those behaviors in the TestCreature class in order to check the the different classes eat and move as defined within the javadoc.

Finally we implemented the unit testing class through the assignment in order to check the validity of the outputs of the main functions of the program.

The README.md file explains how to run the program under a fresh ubuntu distribution. A precompiled .jar file is also provided.

My github repository is : https://github.com/tforlini/CS445