20.10 Program: 5 TortoiseAndHare

Objectives

* Use visual program to observe abstract classes and concrete subclasses.
* "Ape" the code to create a third racer type.
* Add an action to the abstract class, then implement in the concrete subclasses.

Watch it

[Tortoise and Hare demo for PC/MAC](http://video.sdsu.edu/nas/capture/2018/pkraft/RaceTortoiseAndHare/RaceTortoiseAndHare_-_20180605_233350_6.html)  
[Tortoise and Hare demo for mobile devices](http://video.sdsu.edu/nas/capture/2018/pkraft/RaceTortoiseAndHare/RaceTortoiseAndHare_-_20180605_233350_38.mp4)

Instructions Part 1

Get the four files provided in the **Solution** section of this lab. Download them into an IDE project. Do not alter classes Racer, Tortoise, or Hare at this point.

**Race.java modifications**  
Fill in the three places in Race.java (the driver program) where code is required. Find these sections by comments starting with **"\*\*\*\*\* Student write"**

1. Add switch statement to prepareToRace().
2. Add for loop to paintComponent() if clause (enhanced for-loop is best).
3. Add for loop ro paintComponent() else clause (enhanced for-loop is best)

Save it, compile it, and run Race to watch the races! Run it with a variety of racers.

Instructions Part 2

1. Make a new Racer--anything you want.
   * The easiest way to do this is to make a new class and copy into it the code from either Tortoise.java or Hare.java.
   * Then modify the appearance (draw) and position (move) according to what fits your new racer.
   * Add your racer to the switch statement in Race.java so you can race it!
2. Add a new abstract method morph() to the abstract class Racer.java.
   * Implement some type of morphing to be enacted if the racer wins (or some other time if you prefer).
   * Some suggestions from students: dance, jump up and down, put on a hat, turn a different color, grow, shrink.

Additional Requirements

* Your new class draw() must be more than a simple shape; so don't make it like my example of a one-color ball.
* Your new class must have unique move(), that is, it cannot be the same as either the Tortoise or Hare.
* Your new class winner action must be more than just a single change of color.

Turning in the program

Online:

* Print out your new Racer subclass and each **morph()** method you added to Tortoise and to Hare to a pdf file. I need a **pdf** version of your .java file, not your .java file.
* Include enough screen shots of a running program to "prove" it works.
* Email all files to **mkatwala@sdsu.edu** with the subject: **CS 108 Program 5**. If it is a different subject, I will not be able to collect it with my auto-script and will create grading discrepancies.

Submission for this program is "On your honor" statements in the following program which you will upload to this lab. Only print statements that are **true**.

public class Certify {

public static void main(String[] args) {

// Print the following line if you got part 1 working and ran some races. -- 11 points

System.out.println("I certify on my honor that I ran the Race code in part one.");

// Print the following line if you added and raced a new racer type in part 2. -- 7 points

System.out.println("My new racer class is ");

// Print the following line if you added the morph method in part 2. -- 7 points

System.out.println("The action added is morph.");

}

}

Share your racer and morphing!

Upload to Piazza a screen shot of your racer and/or the morphing!