Homework 10

Introductory Programming **Allen B. Downey**
Fall 2004

The reading for this assignment is Chapter 12 of *How to think...*.

10.1 Back to AmoebaWorld!

1. `wget http://wb/ip/code/ip_hw05_soln.py` and make a copy named `hw10.py`. You should have a hardcopy of `ip_hw05_soln.py`; read it and make sure you understand it.

2. Run `hw10.py` and confirm that it does what you expect.

3. Modify the program so that it creates a third Amoeba named `pat`. If you type `more /usr/X11R6/lib/X11/rgb.txt` you will see a list of the color names Python recognizes. You can make `pat` any color you want. Inside the loop, add a line to move `pat` and remove the lines that invoke `nudge`. Run the program again.

4. Rather than invoke `move` three times, you can use a `for` loop to move each of the Amoebas. The object named `world` has an attribute named `animals` that is a list of the animals that have been instantiated.

   First, add a line of code (after you create the third Amoeba) and print `world.animals`. Run the program and confirm that the list contains three instances.

   Now replace the three `move` calls with a loop that traverses the list of animals and calls `move` for each one. Run the program and confirm that the Amoebas still move.

5. The next step is to detect and handle collisions between Amoebas:

   Create a new function called `collide` that takes two Amoebas as parameters. For now, it should just print the two Amoebas.

   Now write a function called `check_collisions` that takes an Amoeba and a list of Amoebas. For each Amoeba in the list, it should compute the distance to the given Amoeba. If the distance is less than 1, it should invoke `collide`.

   Finally add code in `main` to detect collisions by invoking `check_collisions`.

6. Since you took out the `nudge` statements, you might find that it takes a while for Amoebas to find each other. But now that there are more than two Amoebas, it's not as clear who should be chasing who.

   Add an attribute named `crush` to each Amoeba. For each Amoeba `crush` should be another Amoeba. If you are feeling mischievous, you can create a love triangle. Now modify `move` so that after invoking `teleport`, it invokes `nudge` to move the Amoeba closer to `crush`.

7. Inside `mate`, you can print the Amoeba objects, but you might notice that Amoebas don't really know their names! Add an attribute named `name` to each Amoeba. Modify `collide` so it prints the names of the Amoebas that collided.

8. Modify `make_amoeba` so that each Amoeba has an attribute named `spouse` that is initially `None`.

   Modify `mate` so that if two Amoebas collide, and neither one has a spouse, they get married (using your code from Evaluation 7). You might want to change the color of the nucleus (the `color2` attribute) to indicate which Amoebas are married.

9. Challenge: Modify `mate` so that if two Amoebas collide, and they are already married, they create a new smaller Amoeba. The new Amoeba should have an attribute that identifies it as a juvenile, so that if it collides with another Amoeba, it doesn't get married.

10. Challenge: Make Amoebas wait at least 20 moves after they get married and before they spawn another Amoeba.