NUMBER GUESSING GAME

HOMEWORK 2, DUE THURSDAY, MARCH 6, 2014

Files

- `biof309_hw2_solution.pyc` - A fully-playable solution to this assignment that has been compiled into bytecode and is not human readable.
- `biof309_hw2.py` - A partially completed program where you fill in the blanks.
- `biof309_test_hw2.py` - Autograding program that will test your work.
- `biof309_hw2_directions.pdf` - This file.

Assignment Overview

Most of the program has been written already. There are two functions for which you must fill in the code, namely `SolicitInteger` and `RunTurn`. You are not allowed to change the function signatures, i.e., you can't change the name of the function or the arguments. It is also not allowed to change any other part of the program to make your code work.

Description of the Program

The object of the game is for the user to guess an integer chosen randomly by the computer. At the start of the game the user chooses the range of values in which the random number should fall. The game can have up to ten players who will take turns guessing. The program keeps track of the time it takes for each player to guess their number and how many turns s/he needed. At the end of the game it prints out a report on how the contestants did, the winner being the one who took the least amount of time to guess their number. The program then asks the user(s) if they want to play again.

Directions

(1) Download the files. Run the solution code. Run the test code on the solution to see that it works.

(2) Rename the template file to follow the naming convention: `yourlastname_firstinitial_hw2.py`.

(3) Fill in the blanks. Make sure your program pretty much works like the solution, but it doesn't have to be exact.

(4) Run the autograder program on your file on the command line with the command `python biof309_test_hw2.py yourlastname_firstinitial_hw2.py`. Go back and fix your code so that it passes all the tests and you get a higher grade.

(5) Send me an email with your program as an attachment with the subject *Yourlastname HW2 Submission.*

Concepts used

The concepts used in this program may include, but is not limited to, information capture (i.e., `input()` etc.), conditional statements (i.e., `if ... else` etc), dummyproofing (i.e., checking to see whether user's input is valid), loops, using an infinite loop (i.e., `while( True ):`, using `break` to get out), calling a function, returning values from a function, etc...

Python Packages to be Used

`time`, `random`

Tips

- Each function has a docstring that says exactly what the function is supposed to do, just follow the directions.
- Read and understand the other parts of the program to see how your function will be used.
- Try to think about all the ways the user can enter the wrong thing, and build checks into the code to make sure it doesn't crash if the user enters garbage. At no point in playing the game should any SyntaxError or uncaught Exception be raised.
- You might find it easier to test the function you're working on independently from the program. You can do that by commenting out the call to `RunGame()` at the bottom and make the call to your function instead.
- If your program has a bug (logic error), try using print statements to print out values, or comment some lines out and see what happens, or use the Python debugger to step through your code line by line to see what it's doing in real time.