Mastermind

Sonia Szeton, Timothy Romanowski, Thomas Trieu, Rachel Lewis

Description:

A program that remakes the classic board game Mastermind in Python. The computer will act as the code maker, filling four slots with any combination of six colors, the user will be the code breaker and make guesses trying to figure out what the solution is. The computer provides feedback on the correctness of each guess that the user must use along with logic to crack the code.

Vision Statement:

A remake for PC of a classic logic based board game.

Automated Test Cases:

The automated tests can be run with the AlgorithmTestCase.py. They test the program's ability to accurately judge tricky guesses and return the proper pins.

Test	Code	Guess	Expected Result	Actual Result	Status
1	B,Y,G,R	Y,R,Y,R	White, Black	White, Black	Passed
2	Y,Y,B,R	G,Y,R,G	Black, White	Black, White	Passed
3	O,G,R,B	B,G,B,O	White, Black, White	White, Black, White	Passed
4	O,P,R,P	R,P,E,G	Black, White	Black, White	Passed

```
Sonia:~/Shared/.../3308Mastermind: python AlgorithmTestCase.py
TEST 1
Code: [<Peg.blue: 5>, <Peg.yellow: 3>, <Peg.green: 4>, <Peg.red: 1>]
Guess: [<Peg.yellow: 3>, <Peg.red: 1>, <Peg.yellow: 3>, <Peg.red: 1>]
Actual Score: [<Pin.white: 2>, <Pin.black: 1>]
Expected Score: [<Pin.white: 2>, <Pin.black: 1>]
.TEST 2
Code: [<Peg.yellow: 3>, <Peg.yellow: 3>, <Peg.blue: 5>, <Peg.red: 1>]
Guess: [<Peg.green: 4>, <Peg.yellow: 3>, <Peg.red: 1>, <Peg.green: 4>]
Actual Score: [<Pin.black: 1>, <Pin.white: 2>]
Expected Score: [<Pin.black: 1>, <Pin.white: 2>]
.TEST 3
Code: [<Peg.orange: 2>, <Peg.green: 4>, <Peg.red: 1>, <Peg.blue: 5>]
Guess: [<Peg.blue: 5>, <Peg.green: 4>, <Peg.blue: 5>, <Peg.orange: 2>]
Actual Score: [<Pin.white: 2>, <Pin.black: 1>, <Pin.white: 2>]
Expected Score: [<Pin.white: 2>, <Pin.black: 1>, <Pin.white: 2>]
.TEST 4
Code: [<Peg.orange: 2>, <Peg.purple: 6>, <Peg.red: 1>, <Peg.purple: 6>]
Guess: [<Peg.red: 1>, <Peg.purple: 6>, <Peg.empty: 0>, <Peg.green: 4>]
Actual Score: [<Pin.black: 1>, <Pin.white: 2>]
Expected Score: [<Pin.black: 1>, <Pin.white: 2>]
.
----------------------------------------------------------------------
Ran 4 tests in 0.001s

OK
```

User Acceptance Test Plans:

Test Case ID: 1	Test Designed by: TJ
Test Priority: low	Test Designed date: 11/04/15
Module Name: Mastermind username	Test Executed by: Tommy
Test Title: Prompt user for username if one wasn't entered	Test Execution date: 11/08/15
Description: Test the Mastermind menu	

Step	Test Steps	Test Data	Expected Result	Actual Result	Status
1	Run gameboard.py	./gameboard.py	Menu appears	Menu appeared	
2	Press Start without entering a username		Warning appears	Warning did appear	Passed

Test Case ID: 2	Test Designed by: Rachel
Test Priority: medium	Test Designed date: 11/02/15
Module Name: Game board pegs	Test Executed by: Sonia
Test Title: Verify the pegs work	Test Execution date: 11/08/15
Description: Slots should fill only in the first row for the first guess	

Step	Test Steps	Test Data	Expected Result	Actual Result	Status
1	Enter username and press Start	username: Sonia	Game board appears	Game board appeared	
2	Click a peg then try to fill a slot not in the first		Won't allow slots not in first row to	Slot wouldn't fill	Passed

	row		fill		

Test Case ID: 3	Test Designed by: Sonia
Test Priority: high	Test Designed date: 10/28/15
Module Name: Exit Game Board	Test Executed by: Tommy
Test Title: Verify the Exit button works	Test Execution date: 11/04/15
Description: Test the Exit button	

Step	Test Steps	Test Data	Expected Result	Actual Result	Status
1	Enter username and start game	username: Tommy	Game board opens	Game board opened	
2	Click the Exit button		Game quits, program ends	Game quit, window closed	Passed

Link to Github: https://github.com/Arcohe/3308Mastermind