Test Cases

Bowlstrike.py

*Name: Vitor Antunes Cazella*

*Student ID: 91050051*

## First Fixes

As soon I opened the code file I got several syntax errors directly detected by the IDE (VS Code), therefore I tried using the automatic formatting feature from the IDE pressing “Shift + Alt + F” that helped with simple spacing and tabbing problems.

After running for it for the first time I noticed that the program was working fine with no errors, however there was no output regarding what the testing unit should do. Then researching more about the documentation on test units with Python Unittest I understood why there was no output, it was a missing line that starts the testing cases:

if \_\_name\_\_ == '\_\_main\_\_':

unittest.main()

That ran the code with the intended testing output, but there was now a warning during the code compiling:

d:/Projects/SoftwareTesting/SoftwareTesting\_Project/Bowlstrike.py:33: DeprecationWarning: Please use assertEqual

instead.

self.assertEquals(game.score,0)

......

----------------------------------------------------------------------

Ran 6 tests in 0.001s

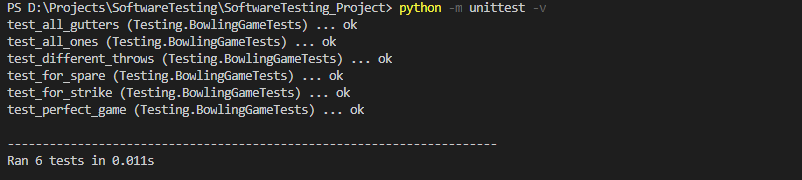
OK

Now after making the changes on methods, testing seem to be working correctly, therefore we can start with designed test cases.

## Test Cases

|  |  |  |
| --- | --- | --- |
| **Test Case #** | **Test Case Description** | **Test Function** |
| 1 | Check game with random numbers of throws but no strikes | *def* test\_different\_throws(self):  *#print("test\_different\_throws")*          game = BowlingGame()          game.throw(6)          game.throw(0)          game.throw(7)          game.throw(0)          game.throw(2)  *for* \_ *in* range(15):              game.throw(0)          game.calculate\_score()  *self*.assertEqual(game.score, 15) |
| 2 | Throws 12 strikes which is the maximum score achievable | *def* test\_perfect\_game(self):  *#print("test\_perfect\_game")*          game = BowlingGame()  *self*.throw\_many(game, 12, 10)          game.calculate\_score()  *self*.assertEqual(game.score, 300) |
| 3 | Tests when every ball strikes one pin | *def* test\_all\_ones(self):  *#print("test\_all\_ones")*          game = BowlingGame()          number\_of\_times = 20          pins = 1  *self*.throw\_many(game, number\_of\_times, pins)          game.calculate\_score()  *self*.assertEqual(game.score, 20) |
| 4 | Checks the scoring in the occasion of a strike | def test\_for\_strike(self):  #print("test\_for\_strike")  game = BowlingGame()  game.throw(10)  game.throw(4)  game.throw(2)  self.throw\_many(game, 17,0)  game.calculate\_score()  self.assertEqual(game.score, 22) |

## Test Results



## Git

This whole project is on GitHub in my personal account, so through all the process of testing and documentation files are available in this repository: [github.com/VitorCazella/SoftwareTesting\_Project](https://github.com/VitorCazella/SoftwareTesting_Project)