

Testing with Pytest

- Pytest was used to test modules in my pygame project prototype
 - This is a testing framework for Python that allows writing of simple and scalable test cases
- Tested modules included:
 - Enemy class
 - Tower class
 - Map class
- Thus thoroughly testing all core functionalities in this prototype of my tower defense game.
- When running test on map class functions, the reset map function did not work as expected.
 - Assertion error shows that the expected output ('empty space') differed from the actual output ('tower') indicating that the reset_map function did not work as intended.
 - The fix here was to use python's deepcopy method (from Python's copy module) to ensure that the map itself was not a stored reference to the default map/grid but rather a copy of it. (fully independent)
- Other testing ran smoothly, with no errors; all actual outputs matched the expected outputs in tested methods.

```
test_map = <Game.map.Map object at 0x00000215A8E3CE20>

def test_reset_map(test_map):
    """Test that reset_map() restores the original grid"""
    # Modify the map by placing a tower
    for y, row in enumerate(test_map.map_grid.grid):
        for x, cell in enumerate(row):
            test_map.place_tower(x, y)
            assert test_map.check_tile((x, y)) == "tower"
            test_map.reset_map()
            assert test_map.check_tile((x, y)) == "empty space"
>
E
E
E
E
- empty space
+ tower

tests/test_map.py:77: AssertionError
FAILED tests/test_map.py::test_reset_map - AssertionError: assert 'tower' == 'empty space'
```

```
----- test session starts -----
platform win32 -- Python 3.13.2, pytest-8.3.5, pluggy-1.5.0
```

```
rootdir:
collected 21 items
```

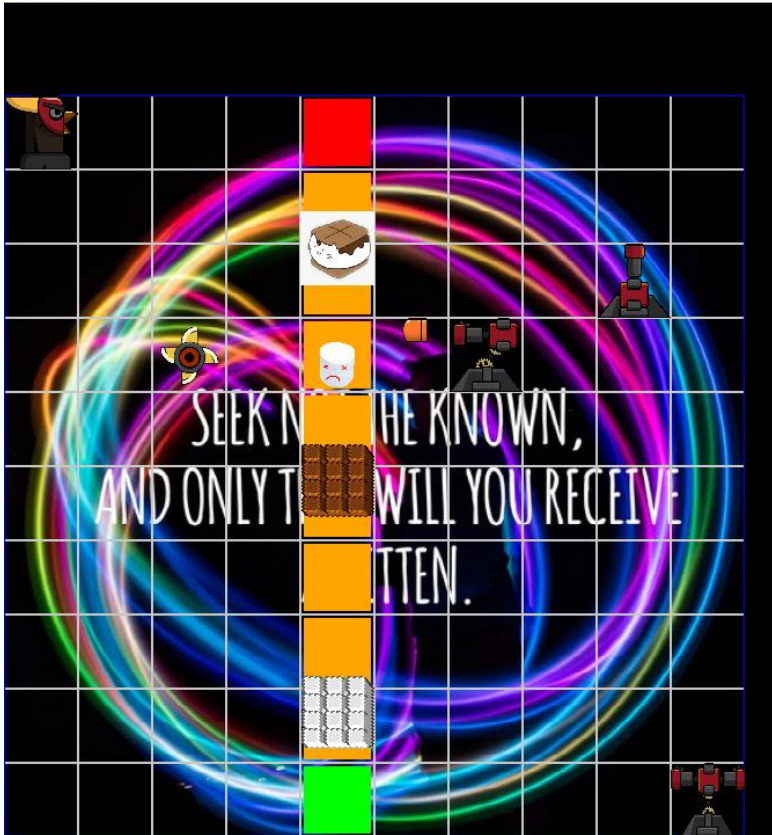
```
tests\test_enemies.py .....
tests\test_map.py .....
tests\test_towers.py .....
```

```
===== 21 passed in 0.13s =====
```

Game prototype in Action (Core Functionality)

- Prototype currently has basic towers, enemies, and a map system involving a grid module.

Marsh Tower Defense



```
[2, 0, 0, 0, 3, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 2, 0],
[0, 0, 2, 0, 1, 0, 2, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 4, 0, 0, 0, 0, 2]
```

In game grid (inside grid class inside map class) at point in prototype screenshot:

- Map backdrop, path, demonstration/placeholder towers and demonstration/placeholder enemies implemented.
 - Towers rendered on map (in grid represented by number 2)
 - Closest tower is in range to shoot – currently shoots bullets consistently horizontally left.
 - Enemies spawn in at start tile (red) and move towards end goal tile (green), and being removed once they reach such tile.
 - Bullets can successfully collide with enemies, registering a hit and lowering the enemy's health by its own damage.
- Basic path drawn based on demonstration grid
 - Orange represents basic path tile. (in grid represented by number 1)
 - Red represents start tile. (in grid represented by number 3)
 - Green represents end goal tile. (in grid represented by number 4)