Spike Outcome Report

Number: 05

Spike Title: Tactical Analysis with PlanetWars

Personal: Peter Argent (7649991)

Goals:

Create at least two different "bot" agents for the PlanetWars simulation.

• One of your bots must utilise tactical analysis to inform its decisions. Examples:

      o Simple: include attacking "weakest", "strongest", "closest" or most productive planet.

      o Complex: include event detection ("fleet leaving planet vulnerable"), scouting or fog-of-war deception.

• Numerically compare each bots' performance and present the results of the both performances over multiple maps.

Technologies, Tools, and Resources used:

- Visual Studio 2017 with Python 3 installed
- Pyglet Documentation here: http://pyglet.readthedocs.io/en/pyglet-1.3-maintenance/
- Help from peers.
- Python 3 Documentation http://docs.python.org/
- The Lab 04 work was used as a base for the spike
  - One of the bots from this work "Simple_Stratigic.py" is used as one of the bots as it uses simple analysis to determine the planet to attack
  - The other bot used is the "Rando.py" which just attacks randomly

Tasks undertaken:

- To get the outcome of multiple games we must change the __main__ function in main.py

```python
if __name__ == '__main__':

    with open("./logs/spike11Results.txt", 'w') as file:
        file.write("Results of wins on 100 maps between Rando and Simple Strategic\n")

    x = 0
    while x < 100:
        gamestate = open('./maps/map{0}.txt'.format(x)).read()
        players = ['Simple_Strategic', 'Rando']
        window = PlanetWarsWindow(gamestate=gamestate, players=players, max_game_length=5000)
        app.run()
        app.exit()
        #window.game.logger.flush()
        x = x + 1
```

- And the update function of the PlanetWarsWindow class

```
        # Has the game ended? (Should we close?)
        if not self.game.is_alive() or self.game.tick >= self.max_tick:
            key = self.game.winner.name
            RESULTS[key] += 1
            with open("./logs/spike11Results.txt", 'a') as file:
                file.write("Winner {0}|| Tally {1}\n".format(key, RESULTS))
            self.game = None
            self.close()
```

- After this Run the program and come back to look at the results in "./logs/spike11Results.txt" file

What we found out:

```
Results of wins on 100 maps between Rando and Simple Strategic
Winner Rando|| Tally {'Simple_Strategic': 0, 'Rando': 1}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 1, 'Rando': 1}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 2, 'Rando': 1}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 3, 'Rando': 1}
Winner Rando|| Tally {'Simple_Strategic': 3, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 4, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 5, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 6, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 7, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 8, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 9, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 10, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 11, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 12, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 13, 'Rando': 2}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 14, 'Rando': 2}
Winner Rando|| Tally {'Simple_Strategic': 14, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 15, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 16, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 17, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 18, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 19, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 20, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 21, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 22, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 23, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 24, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 25, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 26, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 27, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 28, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 29, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 30, 'Rando': 3}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 31, 'Rando': 3}
Winner Rando|| Tally {'Simple_Strategic': 31, 'Rando': 4}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 32, 'Rando': 4}
Winner Rando|| Tally {'Simple_Strategic': 32, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 33, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 34, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 35, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 36, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 37, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 38, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 39, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 40, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 41, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 42, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 43, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 44, 'Rando': 5}
```

```
Winner Simple_Strategic|| Tally {'Simple_Strategic': 45, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 46, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 47, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 48, 'Rando': 5}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 49, 'Rando': 5}
Winner Rando|| Tally {'Simple_Strategic': 49, 'Rando': 6}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 50, 'Rando': 6}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 51, 'Rando': 6}
Winner Rando|| Tally {'Simple_Strategic': 51, 'Rando': 7}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 52, 'Rando': 7}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 53, 'Rando': 7}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 54, 'Rando': 7}
Winner Rando|| Tally {'Simple_Strategic': 54, 'Rando': 8}
Winner Rando|| Tally {'Simple_Strategic': 54, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 55, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 56, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 57, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 58, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 59, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 60, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 61, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 62, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 63, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 64, 'Rando': 9}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 65, 'Rando': 9}
Winner Rando|| Tally {'Simple_Strategic': 65, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 66, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 67, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 68, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 69, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 70, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 71, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 72, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 73, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 74, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 75, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 76, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 77, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 78, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 79, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 80, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 81, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 82, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 83, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 84, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 85, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 86, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 87, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 88, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 89, 'Rando': 10}
Winner Simple_Strategic|| Tally {'Simple_Strategic': 90, 'Rando': 10}
```

The above is the results from running the Planet Wars Game 1 time each on 100 different maps. It took over 6 hours to complete because of a memory issue in code (that has not been found). As we can see the Simple Strategic bot had a 90%-win rate. The win rate would have some margin of error as the other bot it was facing was a Random Bot, thus causing the results to be non-deterministic.

The tactical responses of the Simple Strategic bot were generally good enough to win on any map.

Open issues/risks [Optional]:

Issues involved in this spike include:

- A memory leak issue in the code.
- Because there was only one iteration of each map, there is a significant margin of error within the results per map, even if overall the simple strategic seems to be the winner by a large margin

Recommendations [Optional]:

- There should be more time set aside for this spike to test additional tactically aware bots as well as include more data from each of the maps rather than one game played on each