Swinburne University of Technology

AI for Games (2020 S1)

Doubtfire Submission

05 - Lab - PlanetWars

Submitted By:
Jahanzaib Madhwani
101974245
2020/04/11 23:46

Tutor:
Tien Pham

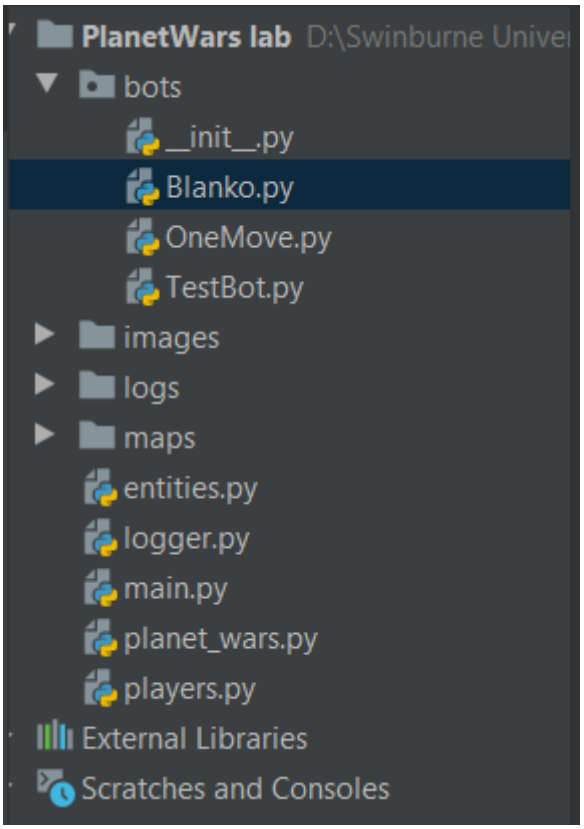
April 11, 2020

PLANET WARS

T05

JAHANZAIB MADHWANI
101974245

- Task asking to create a new bot, I created a new python file in the "**bots**" folder naming it "**TestBot**".



Then I pasted the skeleton code provided in the pdf.

- In order to code to work with the new bot, we actually have to edit the main.py file

```
if __name__ == '__main__':

    gamestate = open('./maps/map5.txt').read()
    players = ['Blanko', 'OneMove','TestBot']
    window = PlanetWarsWindow(gamestate=gamestate, players=players, max_game_length=500)
    app.run()
    window.game.logger.flush()
```

by adding my new bots name inn.

- To make sure this bot only launch one fleet at a time we added the provided code in the pdf.

```
if gameinfo.my_fleets:
    return
```

- The basic design of the bot I pasted the code from the pdf to my bot python file. The only problem was that the code apparently had errors, to fix that all I had to do was to import "choice" from the library "random".

```
# check if we should attack
if gameinfo.my_planets and gameinfo.not_my_planets:
    # select random target and destination
    dest = choice(list(gameinfo.not_my_planets.values()))
    src = choice(list(gameinfo.my_planets.values()))
    # launch new fleet if there's enough ships
    if src.num_ships > 10:
        gameinfo.planet_order(src, dest, int(src.num_ships * 0.75) )
```

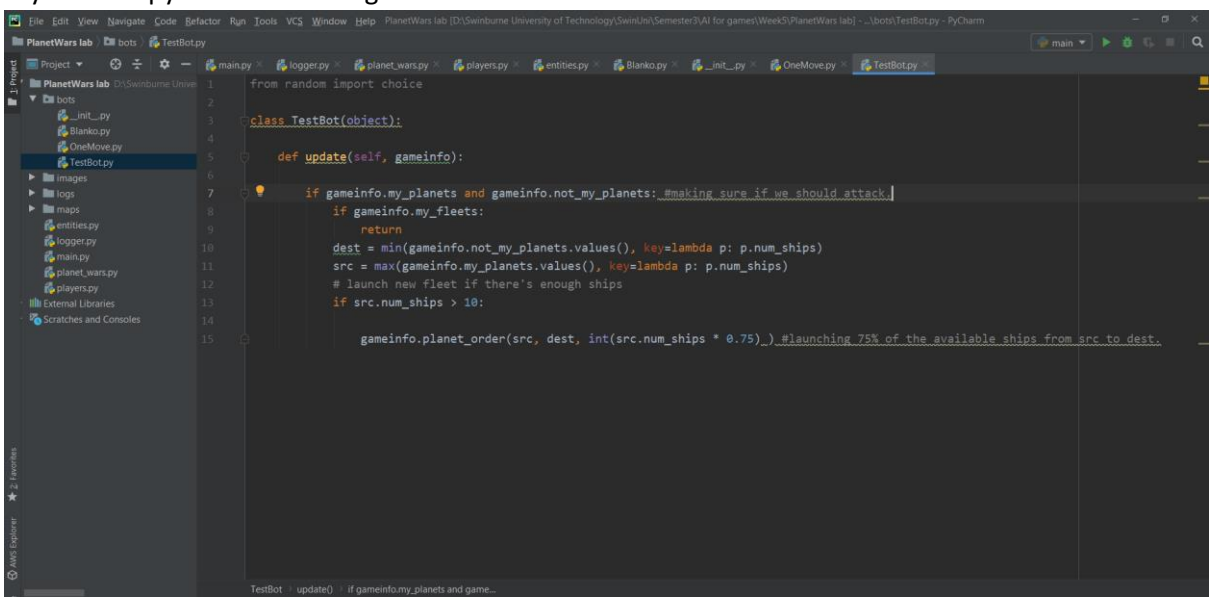
The important thing we need to remember is that we should have at least one planet to attack and launch a fleet that's why we defined the if condition in the start of the above code to avoid exception and fill the game rules.

- Previously we were using the random targets and destination, but since the nature of the game we have to sort through the available planets or fleets. To enable this we did the following:

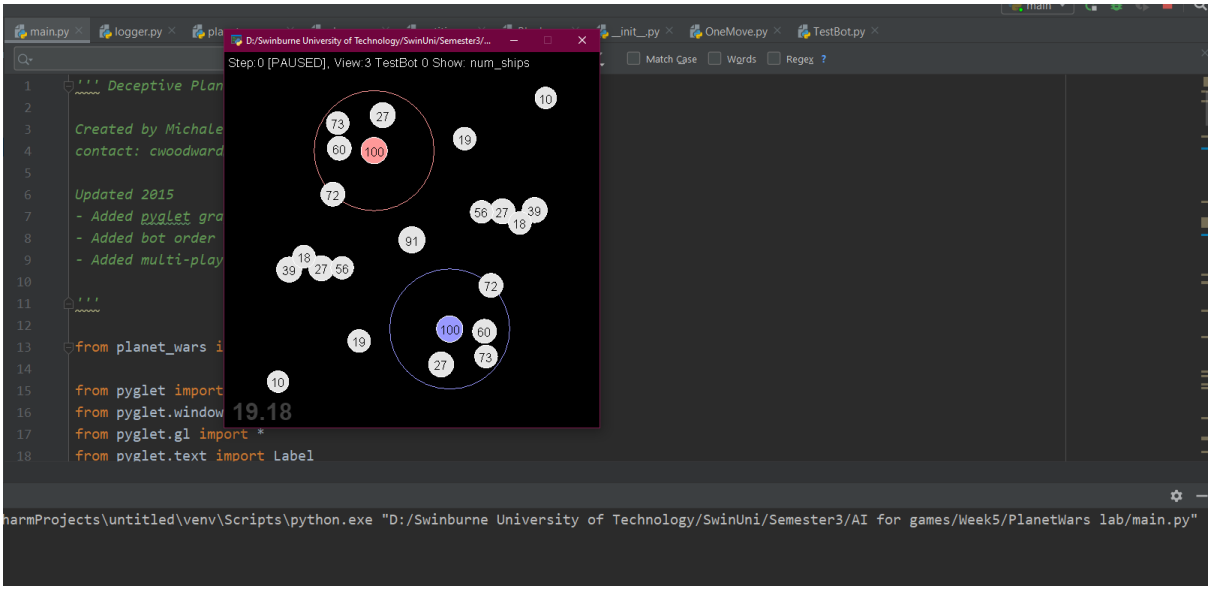
```
dest = min(gameinfo.not_my_planets.values(), key=lambda p: p.num_ships)
src = max(gameinfo.my_planets.values(), key=lambda p: p.num_ships)
```

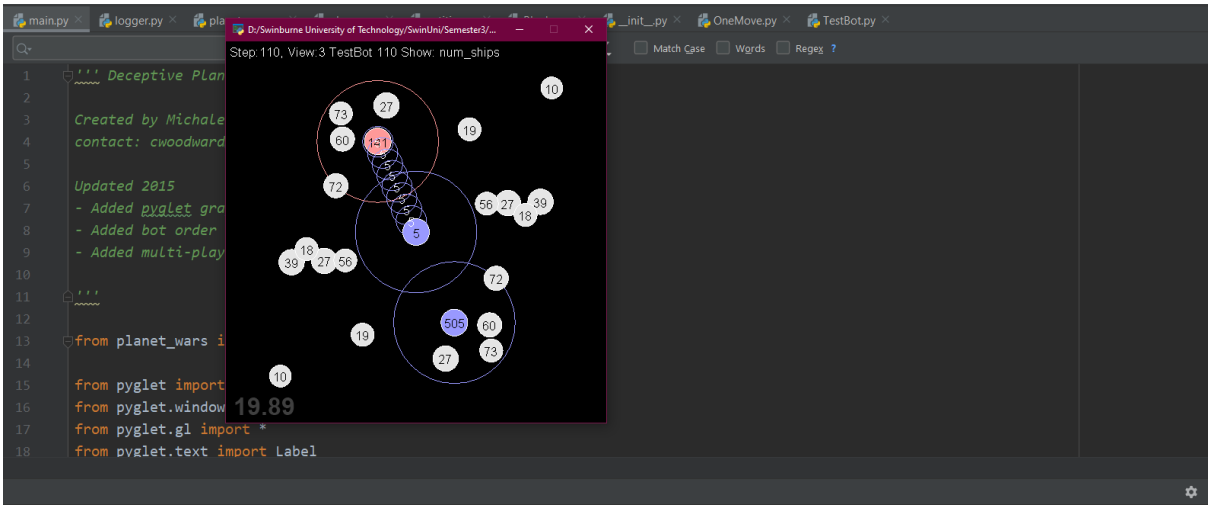
saying that we settled the destination to be minimum, and source to be maximum.
being new to the lambda function after a little bit of research I came across it being the anonymous function. Lambda functions can have any number of arguments but only one expression. The expression is evaluated and returned. Lambda functions can be used wherever function objects are required. In this scenario we used it to make our code efficient and less complex by ending the loops.

- My TestBot.py looks something like this.



- Output:


main.py
logger.py
__init__.py
OneMove.py
TestBot.py
Match Case
Words
Regex
D:/Swinburne University of Technology/SwinUni/Semester3/...
Step:0 [PAUSED], View:3 TestBot 0 Show: num_ships
19.18
''' Deceptive Plan
Created by Michale
contact: cwoodward
Updated 2015
- Added pyglet gra
- Added bot order
- Added multi-play
'''
from planet_wars i
from pyglet import
from pyglet.window
from pyglet.gl import *
from pyglet.text import Label
harmProjects\untitled\venv\Scripts\python.exe "D:/Swinburne University of Technology/SwinUni/Semester3/AI for games/Week5/PlanetWars lab/main.py"


main.py
logger.py
__init__.py
OneMove.py
TestBot.py
Match Case
Words
Regex
D:/Swinburne University of Technology/SwinUni/Semester3/...
Step:110, View:3 TestBot 110 Show: num_ships
19.89
''' Deceptive Plan
Created by Michale
contact: cwoodward
Updated 2015
- Added pyglet gra
- Added bot order
- Added multi-play
'''
from planet_wars i
from pyglet import
from pyglet.window
from pyglet.gl import *
from pyglet.text import Label