P3 Intro

Planet Wars

The Game

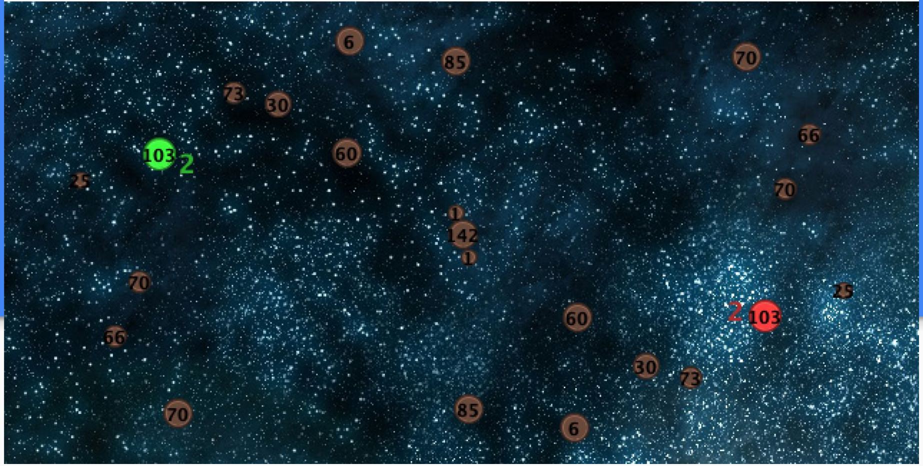
- Based off *Galcon*

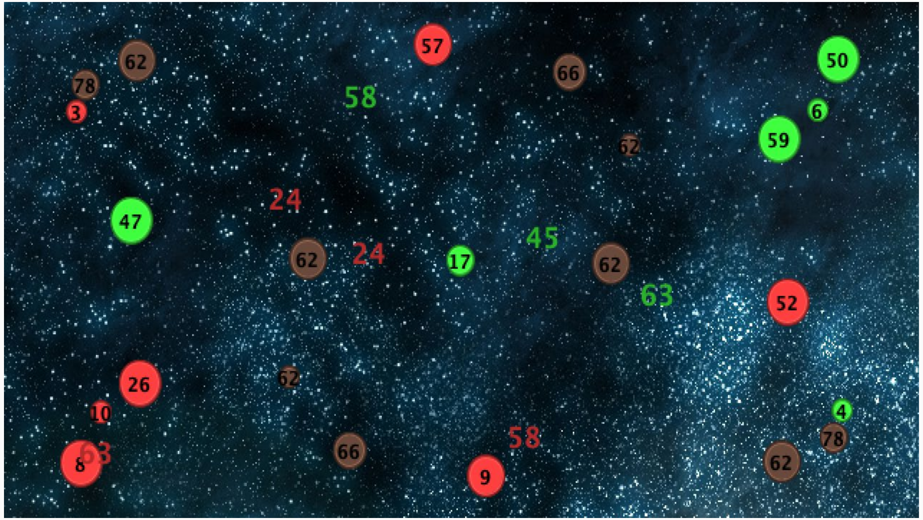
Rules

- Enemy ships cancel each other out when they battle
- Conquered planets produce ships each turn
    - The more ships there, the faster they are produced
- The numbers on the planet indicate how many ships it will take to conquer it

Planet Wars

- How to win
  - Destroy all enemy ships or conquer all the planets


6
85
70
73
30
66
103
2
60
25
70
1
142
1
70
25
2
103
60
66
30
73
70
85
6


57
62
66
50
78
58
3
6
62
59
24
47
45
62
24
17
62
63
52
26
62
10
4
78
58
63
8
66
9
62

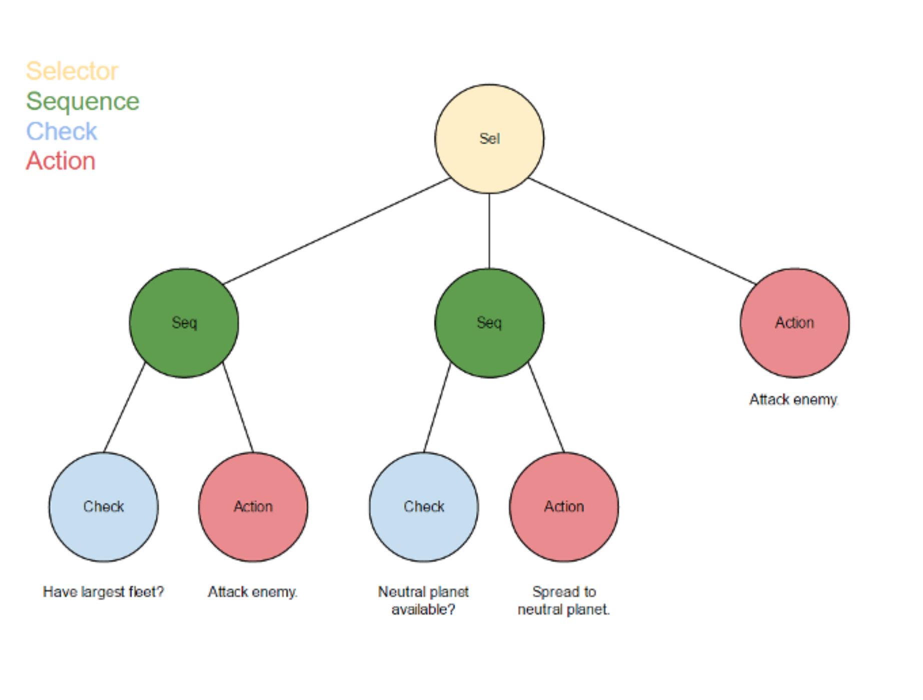
P3 Goal

- Design a reactive bot with a single behavior tree.
- Win against each unique test bot and your classmates.

Getting Started

- May have to install Java JDK:
    - https://www.oracle.com/java/technologies/javase-jdk15-downloads.html
- Before trying to build your bot, watch the test bots play out.
    - Understand strategies
    - How can you beat these bots?
- Look over bt_bot.py
    - A basic behavior tree is already provided. You can improve on this or build a new one.
    - Look at how the tree is assembled
- Look at other bot's code to see how the functions in planet_wars.py can be used

Provided Behavior Tree



Code

- Your tree will be assembled in bt_bot.py
- Check functions will be defined and implemented in checks.py
- Action functions will be defined and implemented in behaviors.py
- planet_wars.py
  - Contains classes for planets, fleets, and the game state PlanetWars.
  - Contains the functions issue_order and finish_turn
  - PlanetWars contains all the game's relevant information and methods for accessing this information

Canvas Link

- https://canvas.ucsc.edu/courses/46859/assignments/280059