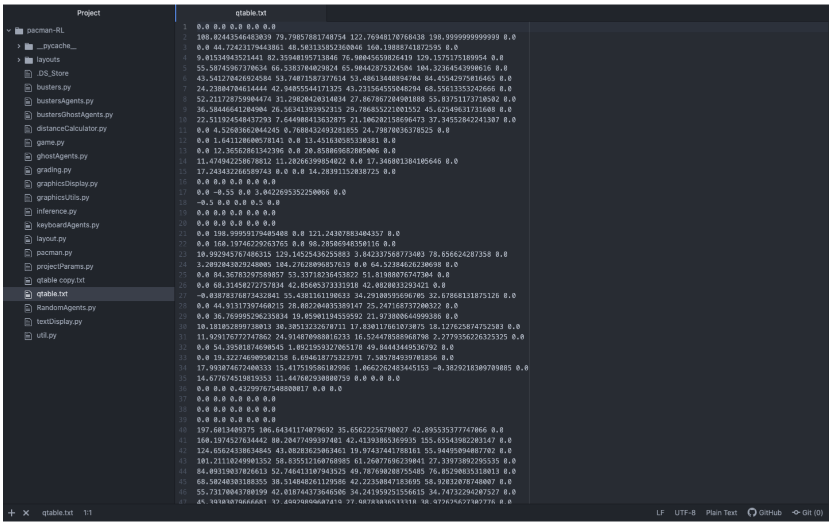
Name: Othmane Echchabi
NIA: 100477811
Name: Violette Castells
NIA: 100478225

Phase 1. Selection of the state information and reward function

First of all, we decided to take two attributes into consideration as they are the most relevant to the problem we want to solve: the distance of Pac-Man from the closest Ghost and its relative position to the latter (North East, North West, South East, South West).

That is, our Q-Table looked like this at the end of our training:



When it came to the reward function, we decided that every time Pac-Man gets closer to the closest Ghost, it gets +1, and it gets -1 if that is not the case. We also decided to make the reward for eating a Ghost the same as the bonus that adds up to the score: +200.

Phase 2. Generation of the agent

When training our agent, we realised that the more it explores, the more accurate it becomes, and thus the less the α and ε values will need to be to make it work perfectly. However, as we want it to work on several layouts with more walls, it takes way too long to reach perfection. Thus, after around 100 exploration games for each of the 5 layouts, and after many trails, we found that the best values for our model are the following:

```
self.epsilon = 0.4
self.alpha = 0.4
self.discount = 0.8
```

Phase 3. Evaluation of the agent

After finding the best optimal values for our agent, we played it in the different mazes, where it obtained the following scores:

labAA1:
Average Score: 173.0
Scores: 171, 175, 171, 171, 177
Win Rate: 5/5 (1.00)

labAA2:
Record: Win, Win, Win, Win, Win
Average Score: 365.8
Scores: 357, 373, 357, 371, 371
Win Rate: 5/5 (1.00)
Record: Win, Win, Win, Win, Win

labAA3:
Average Score: 540.6
Scores: 545, 547, 549, 511, 551
Win Rate: 5/5 (1.00)
Record: Win, Win, Win, Win, Win

labAA4:
Average Score: 546.6
Scores: 537, 551, 549, 545, 551
Win Rate: 5/5 (1.00)
Record: Win, Win, Win, Win, Win

labAA5:
Average Score: 696.0
Scores: 718, 722, 516, 734, 790
Win Rate: 5/5 (1.00)
Record: Win, Win, Win, Win, Win

Conclusion:

Q-Learning can be a great machine method method for decision making problems, however the bigger the data size (the maze in this case), the less efficient it becomes