Project 1: Game Playing Using easyAI

CAP4630 - Dr. Marques

Mayte Ramirez-Calderon

Johnny Figueroa

Ethan Curtis

Roles:

- Design of the solution (“Architect”): Johnny Figueroa
- Coding of the Solution (“Developer”): Ethan Curtis
- Documentation of the Solution (“Reporter”): Mayte Ramirez-Calderon

Overview:

For Project 1, our group has been tasked with implementing three games using an easyAI framework.

Part 1:

For the first part of this project, our group was required to set up our environment in order to successfully complete the rest of our project. To do this we used the second method

provided in the guidelines which was to use Conda. We downloaded Conda for our corresponding operating system. Once we verified that Conda was installed, we created a conda environment and installed python version 3.8.

```
(base) mayteramirez@Maytes-Macbook ~ % conda create -n py38 python=3.8
Collecting package metadata (current_repodata.json): done
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.12.0
  latest version: 4.13.0

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: /Users/mayteramirez/opt/anaconda3/envs/py38

  added / updated specs:
    - python=3.8


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    ca-certificates-2022.4.26  |       hecd8cb5_0         124 KB
    certifi-2022.5.18.1        |   py38hecd8cb5_0         148 KB
    openssl-1.1.1o             |       hca72f7f_0         2.2 MB
    pip-21.2.4                 |   py38hecd8cb5_0         1.8 MB
    python-3.8.13              |       hdfd78df_0        10.8 MB
    setuptools-61.2.0          |   py38hecd8cb5_0        1012 KB
    sqlite-3.38.3              |       h707629a_0         1.2 MB
    tk-8.6.12                  |       h5d9f67b_0         3.1 MB
    xz-5.2.5                   |       hca72f7f_1         244 KB
    ------------------------------------------------------------
                                           Total:        20.5 MB

The following NEW packages will be INSTALLED:

  ca-certificates    pkgs/main/osx-64::ca-certificates-2022.4.26-hecd8cb5_0
  certifi            pkgs/main/osx-64::certifi-2022.5.18.1-py38hecd8cb5_0
  libcxx             pkgs/main/osx-64::libcxx-12.0.0-h2f01273_0
  libffi             pkgs/main/osx-64::libffi-3.3-hb1e8313_2
  ncurses            pkgs/main/osx-64::ncurses-6.3-hca72f7f_2
  openssl            pkgs/main/osx-64::openssl-1.1.1o-hca72f7f_0
  pip                pkgs/main/osx-64::pip-21.2.4-py38hecd8cb5_0
  python             pkgs/main/osx-64::python-3.8.13-hdfd78df_0
  readline           pkgs/main/osx-64::readline-8.1.2-hca72f7f_1
  setuptools         pkgs/main/osx-64::setuptools-61.2.0-py38hecd8cb5_0
  sqlite             pkgs/main/osx-64::sqlite-3.38.3-h707629a_0
  tk                 pkgs/main/osx-64::tk-8.6.12-h5d9f67b_0
```

The next step was to activate the conda environment.

```
# To activate this environment, use
#
#     $ conda activate py38
#
# To deactivate an active environment, use
#
#     $ conda deactivate

(base) mayteramirez@Maytes-Macbook ~ % conda activate py38
```

Finally, we checked to make sure our conda environment is activated.

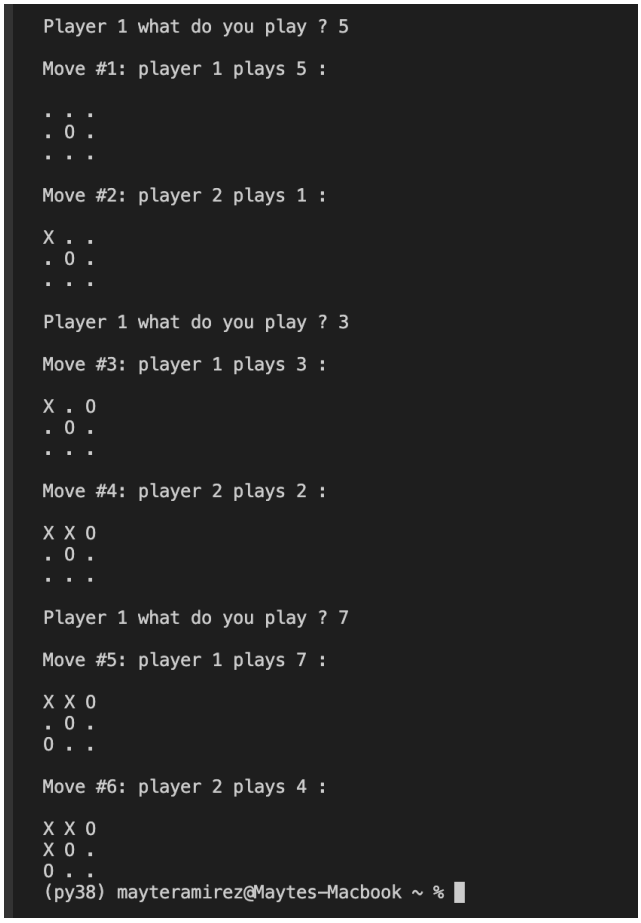
```
(py38) mayteramirez@Maytes-Macbook ~ % conda info --envs
# conda environments:
#
base                     /Users/mayteramirez/opt/anaconda3
py38                  *  /Users/mayteramirez/opt/anaconda3/envs/py38
```

The second requirement for Part 1 required us to install the easyAI python framework. This was successfully done using pip.

```
Collecting easyAI
  Downloading easyAI-2.0.12-py3-none-any.whl (42 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 42.2/42.2 kB 4.7 MB/s eta 0:00:00
Requirement already satisfied: numpy in ./opt/anaconda3/lib/python3.9/site-packages (from easyAI) (1.21.5)
Installing collected packages: easyAI
Successfully installed easyAI-2.0.12
```

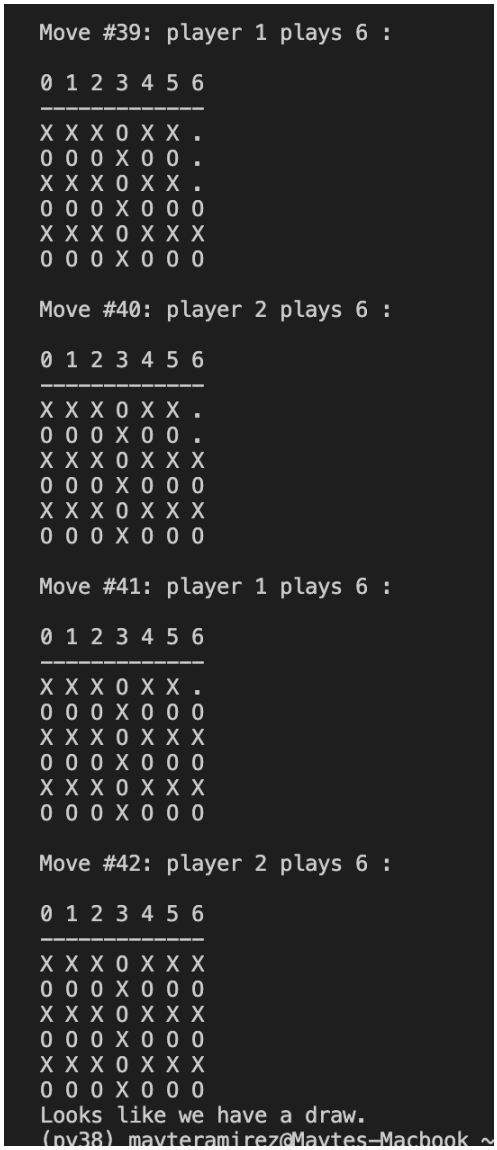
Part 2:

For the second part of this project, our group was tasked with running two examples presented in the easyAI documentation. The first example was running a tic-tac-toe game. Our group used Visual Studio as our editor to successfully run the code provided for the tic-tac-toe game. An example of a game can be seen below:



The second example presented in the easyAI documentation that our group had to run was the Connect Four game. The beginning of the game and the final results can be seen below.

```
0 1 2 3 4 5 6
-------------
. . . . . . .
. . . . . . .
. . . . . . .
. . . . . . .
. . . . . . .
. . . . . . .

Move #1: player 1 plays 0 :

0 1 2 3 4 5 6
-------------
. . . . . . .
. . . . . . .
. . . . . . .
. . . . . . .
. . . . . . .
O . . . . . .

Move #2: player 2 plays 0 :

0 1 2 3 4 5 6
-------------
. . . . . . .
. . . . . . .
. . . . . . .
. . . . . . .
X . . . . . .
O . . . . . .

Move #3: player 1 plays 0 :

0 1 2 3 4 5 6
-------------
. . . . . . .
. . . . . . .
. . . . . . .
O . . . . . .
X . . . . . .
O . . . . . .
```



It was noted that each time the fixed connect four game was executed, the players selected slots in order from 1 to 9 which provided the same results every time. A future improvement our group would have liked to work on is fixing this issue to make each move more random.

Part 3:

For the final part of this project, we had to implement a modified version of the game of checkers. We had to implement a two player game where the objective is to get as many pieces as you can from the opponent. This is done on an 8x8 board of checkers with two colors, one for each opponent.