Engine How-to

Engine link: https://github.com/mitpokerbots/engine-2021

Requirements

The engine for Pokerbots 2021 is written in Python 3. Please ensure that you have followed all the setup instructions outlined on Piazza, and check that your installation works by performing the following three commands:

1) Run `$ python3` from the command line to ensure that Python is installed. Verify that an interpretive shell pops up. You can exit this shell by typing `>>> quit()`

2) Run `$ python3 --version` from the command line to verify that your Python version is at least 3.7. If not, you can update Python by following the instructions for your OS here.

3) Run `$ pip3 freeze` from the command line to ensure that all dependencies are in place. The bot requires the following Python3 modules:

- `cython`
- `eval7`

`$ pip3 install Package` if these are not listed.

[Note: to run the engine on Windows you must use the Windows Subsystem for Linux (WSL). Details for installing and setting up WSL are posted on our Piazza.]

Development

The engine supports pokerbots written in Python 3, Java, or C++. Therefore, one of these platforms is required to develop your bot. For beginners, we strongly suggest Python, as it will be the easiest to debug. However, we will provide support for all platforms to the best of our ability.

Usage

We recommend running the engine from a command line interpreter. To do this, navigate to the directory of the engine and run the command `$ python3 engine.py`

The engine will run one game according to the parameters in `config.py` and will save the results in several files outlined below:

`A.txt`	Text log of all output (print/error) from the bot of the player named `A`.

`B.txt`	Text log of all output (print/error) from the bot of the player named `B`.
`gamelog.txt`	Text log for the game named `gamelog`. This will be the most useful file for determining what happened in the game.

Parameters

The parameters to the game engine are controlled via the configuration file `config.py` in the same directory as the engine file `engine.py`. An outline of the parameters and their meanings is provided below:

`GAME_LOG_FILENAME`	Name of the game log output file. Default `gamelog`.
`SMALL_BLIND`	Amount of the small blind. Default `1`. Does not change.
`BIG_BLIND`	Amount of the big blind. Default `2`. Does not change.
`STARTING_STACK`	Stack allocated per round. Default `200`. Does not change.
`NUM_BOARDS`	Number of Blotto boards in the game. Default `3`. Does not change.
`NUM_ROUNDS`	Number of rounds for the game. The scrimmage server uses `500`, but feel free to use any number of rounds for offline games.
`BUILD_TIMEOUT`	How long the engine will wait for your `commands.json` build command to complete, in seconds. The scrimmage server uses `10`. (You should not have to worry about this parameter.)
`CONNECT_TIMEOUT`	How long the engine will wait for you to take an action before it assumes you have disconnected, in seconds. The scrimmage server uses `10`. (You should not have to worry about this parameter.)
`QUIT_TIMEOUT`	How long the engine will wait for your bot to shutdown, in seconds. The scrimmage server uses `10`. (You should not have to worry about this parameter.)
`STARTING_GAME_CLOCK`	A bot's total time to make all its actions in the game, in seconds. The scrimmage server uses `30`. A bot that runs out of time will check/fold for all its remaining actions.
`ENFORCE_GAME_CLOCK`	Whether or not the engine will use the game clock. The scrimmage server uses `True`.

The directories of the two bots for the game may also be changed in `config.py`.