Assignment 4: Deep RL

This folder contains the problem statement and the basic code for the assignment. You must attempt this assignment only in python. You are provided with minimum boilerplate code to begin with, and you need to fill up the missing code such that all conditions mentioned in the question are met.

Problem statement

You need to train an AI to control that can play the game given in this assignment. The game is composed of

- A goal state : A rectangular location in the game screen.
- Multiple enemies : Rectangular blocks that move with a constant speed and bounce around the game screen.
- Player entity : A rectangular block that the player/AI must control in the 4 cardinal directions.

The player must reach the goal state (that is, the player rectangle and the goal state rectangle must intersect) while avoiding the enemies.

Every time the player reaches the goal state, or gets attacked, the goal state is reset, and the game continues from that point.

To run the game do the following steps

- Install the package `pygame`. You can do this via `pip install pygame`
- Assign a valid integer to the constant `GAME_SEED` in the file `game_constants.py`. You can see the section below on assigning the value appropriately.
- Run the file `game.py`. It will prompt if you wish to control the game with your own controls (Keyboard) or your custom AI Controller. Enter the input and launch the game.

Note: Before you run the game, or complete the implementation, you must assign the `GAME_SEED` in the file `game_constants.py`. Ensure that this value of seed is unique among all the submissions. You can do so by selecting it from your roll numbers.

Since your roll numbers are composed of letters and numbers, convert each letter their corresponding number in the alphabet, concatenate with the rest of numbers, and use this number as a seed.

For example, if a students' roll number is B21CSE001, B becomes 2, C becomes 3, S becomes 19, E becomes 5, and thus the seed becomes 2213195001

You must mention your selected seed in your report.

Part I

In your report, describe this game in terms of agent, environment, actions and observations. Describe an appropriate reward function for this game, and briefly explain why it is valid for this game. (Hint: You can try playing the game, or look at its source code in `game_state.py`).

Part II

Select a suitable Deep RL algorithm and implement it in the game. You must add this implementation in the file `game_controller.py`, within the class functions of `AIController`. Notice the comments that explain which portion of the file you are allowed to change, and complete the implementation accordingly.

On running the game from the file `game.py`, enter the prompt to use your AI Model, and the code will automatically call the `Train()` function of the controller and run a custom evaluation on it before launching the game. You are required to mention this evaluated result in your report.

Note: The game uses the library `pygame` to render the game. Use `pip install pygame` to ensure before running the game.

Part III

On basis of your selected algorithm, briefly explain if after training your model, would it still score well in this game (without further retraining) on changing the following game constants (defined in the file `game_constants.py`)?

- If `GAME_SEED` is selected randomly, rather than keeping it a constant number.
- If the dimensions of the game, i.e `GAME_WIDTH` and `GAME_HEIGHT` are changed.
- If the variable `GOAL_SIZE` is changed.
- If the variable `ENEMY_COUNT` is changed.
- If the variable `GAME_FRICTION` is changed.
- If the variable `FPS` is changed.

Answer all these points in brief in the report.

General Guidelines for the assignment

- This assignment is to be done in python. Dependencies for deep learning and linear algebra are allowed. However, the RL algorithm must be implemented based on these dependencies within your code; that is, you are not allowed to use standard RL libraries, and the implementation of your selected RL algorithm must be present your the code.

- The sections **Part I, Part III** are to be done in the in the report. The section **Part II** is to be done in the file `game_controller.py`, and the output (Evaluation of the trained AI controller) obtained from running `game.py` is to be included in the report as well.

- Before attempting the assignment, ensure that the `GAME_SEED` value is assigned correctly.

- Complete the given source code file, and upload it along with the report. (**Do not include any additional folders or compress in a the zip file**). Upload the following on google classroom.

  ```
  ├── game_controller.py
  └── Report.pdf
  ```

- In continuation with the above point, the file `game_controller.py` should only change the lines as mentioned in the comments. No other line must be changed.

- For evaluation of your assignment, your code will be tested with additional hidden test statements. They will be use the same seed value mentioned in your project report. So ensure that the your code works with the said seed value in your machine.

- All students should select the seed value uniquely as mentioned in the problem statement. For students in group, any one student's roll number is to be selected as mentioned, and must be included in the report.

- For students in group, only one submission per group is required. However, there must be private comments for all members of the group that their partner has submitted.

- All students are expected to maintain their integrity and do the assignment on their own. Any cases of cheating/copying will be awarded 0 marks.