CS440 – Creative Contest2: Multi-Agent Adversarial Pacman

Purpose: This project involves a multi-player capture-the-flag variant of Pacman, where agents control both Pacman and ghosts in coordinated team-based strategies. Your team will try to eat the food on the far side of the map, while defending the food on your home side.

Getting Started: Follow these steps to get started.

1. Download the contest2.zip files.
2. Run `capture.py` to see the two rather stupid baseline teams play against each other..

Objective: This is now an adversarial game, involving two teams competing against each other. Your team will try to eat the food on the far side of the map, while defending the food on your home side.

Your agents are in the form of ghosts on your home side and Pacman on your opponent's side. Also, you are now able to eat your opponent when you are a ghost. If Pacman is eaten by a ghost before reaching his own side of the board, he will explode into a cloud of food dots that will be deposited back onto the board.

All your code should be written in `myTeam`. To get started designing your own agent, subclass the `CaptureAgent` class which provides access to several convenience methods.

Rules: The Pacman map is now divided into two halves: blue (right) and red (left). Red agents (which all have even indices) must defend the red food while trying to eat the blue food. When on the red side, a red agent is a ghost. When crossing into enemy territory, the agent becomes a Pacman.

There are a variety of layout in the layouts directory.

As a Pacman eats food dots, those food dots are stored up inside of that Pacman and removed from the board. When a Pacman returns to his side of the board, he "deposits" the food dots he is carrying, earning one point per food pellet delivered. Red team scores are positive, while Blue team scores are negative.

If Pacman gets eaten by a ghost before reaching his own side of the board, he will explode into a cloud of food dots that will be deposited back onto the board.

If Pacman eats a power capsule, agents on the opposing team become "scared" for the next 40 moves, or until they are eaten and respawn, whichever comes sooner. Agents that are "scared" are susceptible while in the form of ghosts (i.e. while on their own team's side) to being eaten by Pacman. Specifically, if Pacman collides with a "scared" ghost, Pacman is unaffected and the ghost respawns at its starting position (no longer in the "scared" state).

Each agent can see the entire state of the game, such as food pellet locations, all pacman locations, all ghost locations, etc. See the `GameState` for more details.

In this adversarial game, a team wins when they return all but two of the opponents' dots. Games are also limited to 1200 agent moves (moves can be unequally shared depending on different speeds - faster agents get more moves). If this move limit is reached, whichever team has returned the most food wins. If the score is zero (i.e., tied) this is recorded as a tie game.

Each agent has 1 second to return each action. Each move which does not return within one second will incur a warning. After three warnings, or any single move taking more than 3 seconds, the game is forfeit. There will be an initial start-up allowance of 15 seconds (use the registerInitialState function).

Testing: You can run your various team tests by running `capture.py` and specifying who is playing for the red and blue teams with the command options like:

```
-r myTeam -b  baselineTeam
```

This would play your `myTeam` agents as red against the `baselineTeam` agents as bluewith the `defaultCapture` layout. You can try other layouts by using the command option `-l` such as `-l  alleyCapture`. You can also generate random layouts such as `-l RANDOM13` will use a map randomly generated with seed 13.

You can create multiple teams by using different python file names like `myTeam42` and trying your agents against another smart team.

Grading: You will receive the full 10pts for this lab by submitting working agents that score against the baselineTeam most of the time. Additionally, this is a contest so your agents will play in a single elimination tournament in class. The tournament winner will receive a bonus of 3 extra credit points. The second place team will receive a bonus of 2 extra credit points. The losers to the top two teams will battle it out for 1 extra credit point.

Submit your `myTeam.py` in Canvas for grading.