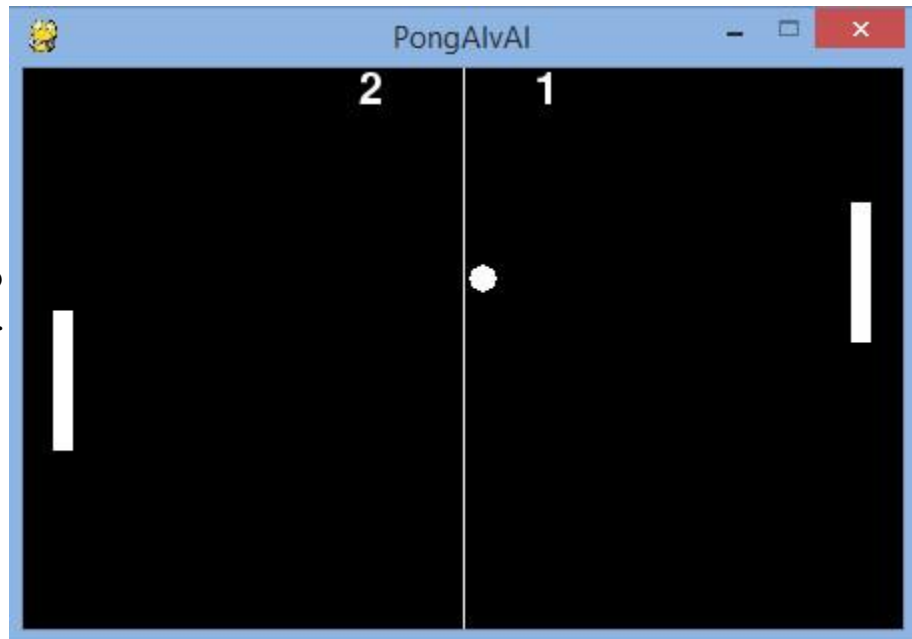


# The CSC180 AI v. AI Pong Tournament

We will have a [Pong](#) tournament for AI engines as part of CSC180 this year. The entry submission deadline will be Nov. 25. Teams of up to two people will be allowed.

**Prizes:** 4 points for first place, 3 points for second place, 2 points for third place, 0.5 points for beating The Chaser by at least 150 points when playing to 1000 points.



The game physics are included in [PongAIvAI.py](#). A sample AI engine, which you have to beat to qualify for the class tournament, is at [chaser\\_ai.py](#).

## Installing PyGame

To run PongAIvAI, you need to install [PyGame](#). (For Windows, I recommend downloading the latest version of PyGame from [here](#). See the [PyGame download page](#) for other platforms.)

## Running PongAIvAI

You can run `PongAIvAI.py` in Python if you download `chaser_ai.py` into the same folder as `PongAIvAI.py`, after you install PyGame. See lines 389-390 in `PongAIvAI.py` for how to set the AI/human input functions for PongAIvAI:

```
import chaser_ai

# To have The Chaser play against your AI engine,
# store your code in student_ai.py, import student_ai,
```

```
# and set paddles[1].move_getter to student_ai.pong_ai
paddles[0].move_getter = chaser_ai.pong_ai
paddles[1].move_getter = directions_from_input # chaser_ai.pong_ai
```

Uncomment line 87 and comment out line 83 in `PongAIvAI.py` once your function doesn't produce errors (line 82 suppresses all errors and simply doesn't move the paddle if your function produces an error). Line 82 only accepts input for 0.3ms.

```
# The program crashes if move_getter crashes. The runtime of
# move_getter is not limited
direction = self.move_getter(self.frect.copy(), enemy_frect.copy(),
ball_frect.copy(), tuple(table_size))

# The program continues if move_getter crashes. The runtime of
# move_getter is limited
# direction = timeout(self.move_getter, (self.frect.copy(),
enemy_frect.copy(), ball_frect.copy(), tuple(table_size)), {}, self.timeout)
```

## Tournament Format

A round-robin tournament will be held among all the teams that qualify for the tournament. In each match, each opponent will play both on the left and on the right. Game parameters will not change by default (except for the score needed to win), but may change at our discretion for tiebreaking purposes. There is a 0.3 ms limit on how long it takes for your function to return a move. For timing purposes, the tournament will be run on ECF or a faster machine. The parameters will stay the same as in `PongAIvAI` by default, but may be altered for tie-break purposes.

## Submission instructions

Please submit a file named `pong_ai.py`, which must contain the function `pong_ai`, which will be called the same way `chaser_ai.pong_ai()` is called.

Good Luck!