

2020 OSS 개발자 포럼 겨울 캠프 1 일차 과제

~ 커스텀 에이전트 만들기 ~

주의사항

과제 제출 기한은 2020년 1월 10일 오후 11시 59분까지입니다.

만든 에이전트 파일을 utilForever@gmail.com 으로 제출해주시기 바랍니다.

또한 돌을 놓는 데 걸리는 시간은 최대 20초로 제한합니다. 20초를 초과하면 자동 패배 처리합니다.

과제와 관련해 궁금한 점이 있다면 카카오톡 채팅방에 문의해주시기 바랍니다.

과제 안내

오늘 수업 시간 때 배웠던 RandomAgent 를 기반으로 여러분 만의 커스텀 에이전트를 만들어 봅니다.

```
# 임의의 위치에 돌을 놓는 에이전트
class RandomBot(Agent):
    # 행동을 선택하는 메소드
    def select_move(self, game_state):
        """Choose a random valid move that preserves our own eyes."""
        candidates = []
        for r in range(1, game_state.board.num_rows + 1):
            for c in range(1, game_state.board.num_cols + 1):
                candidate = Point(row=r, col=c)
                if game_state.is_valid_move(Move.play(candidate)):
                    candidates.append(candidate)
        return Move.play(random.choice(candidates))
```

현재 RandomAgent 의 select_move 메소드를 보면 돌을 놓을 수 있는 위치 목록을 가져와 임의로 한 곳을 선택해 돌을 놓습니다. 이 부분을 팀원들과 고민해 다른 호실에서 만든 에이전트와 상대했을 때 이길 수 있도록 코드를 바꿔주세요. If 문으로 작성해도 좋고 다른 알고리즘을 사용해도 좋습니다.

참고로 만들 때 에이전트 이름을 RandomBot 이 아닌 **"C + 호실 이름 + Bot"**으로 바꿔주세요. 예를 들어 자기 호실이 302 호라면 **"C302Bot"**으로 만들면 됩니다. 또한 파일 이름도 **"C + 호실 이름.py"**로 만들어주세요. 예를 들어 자기 호실이 302 호라면 **"C302.py"**으로 만들면 됩니다.

테스트 방법

agents 폴더에 c302.py 라는 파일을 만들었고 에이전트 이름이 C302Bot 이라고 가정합니다.

이제 bot_v_bot.py 파일을 다음과 같이 변경합니다.

```
def main():
    board_size = 8
    game = connect5_board.GameState.new_game(board_size)
    bots = {
        types.Player.black: agent.C302.C302Bot(),
        types.Player.white: agent.C302.C302Bot(),
    }
    ...
```

이렇게 바꾸면 새로 만든 에이전트끼리 대결을 하게 되겠죠. 만약 새로 만든 에이전트와 랜덤 에이전트가 대결하게 만들고 싶다면 다음과 같이 바꾸면 될 겁니다.

```
def main():
    board_size = 8
    game = connect5_board.GameState.new_game(board_size)
    bots = {
        types.Player.black: agent.C302.C302Bot(),
        types.Player.white: agent.naive.RandomBot(),
    }
    ...
```

시간 측정 방법

time 모듈을 추가해 로직 수행 전에 start_time 에 현재 시각을 저장하고 로직 수행 후 현재 시각에서 start_time 을 빼면 수행 시간을 알 수 있습니다.

```
import random
from connect5.agent.base import Agent
from connect5.board import Move
from connect5.types import Point

import time
```

```

# 임의의 위치에 돌을 놓는 에이전트
class RandomBot(Agent):
    # 행동을 선택하는 메소드
    def select_move(self, game_state):
        start_time = time.time()

        """Choose a random valid move that preserves our own eyes."""
        candidates = []
        for r in range(1, game_state.board.num_rows + 1):
            for c in range(1, game_state.board.num_cols + 1):
                candidate = Point(row=r, col=c)
                if game_state.is_valid_move(Move.play(candidate)):
                    candidates.append(candidate)
            move = Move.play(random.choice(candidates))

        print('{} seconds elapsed.'.format(time.time() - start_time))
        return move

```

대결 안내

모든 대결은 토너먼트 방식, 단판으로 진행합니다. 단, 결승전은 3전 2선승제로 진행합니다.

선후공은 랜덤으로 결정합니다. 또한 대결에서 사용할 바둑판의 크기는 8 * 8 로 고정합니다.

1 등은 상품이 있으니 열심히 해주시길 바랍니다. 그럼 화이팅! :D