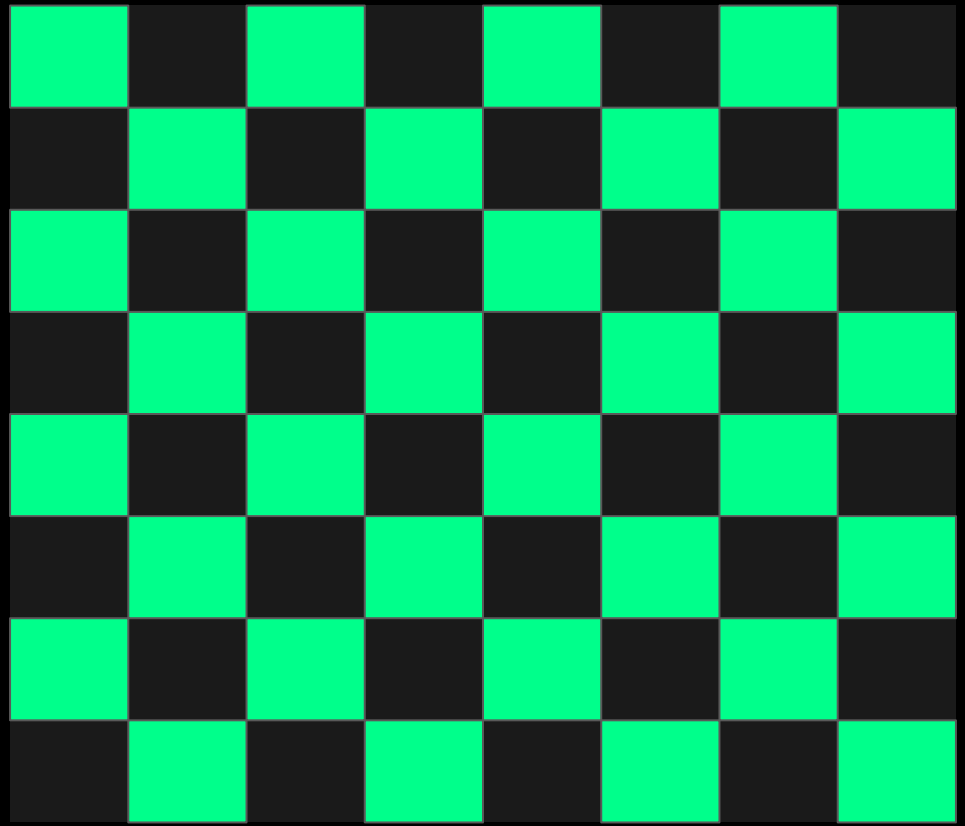

SEOUL AI GYM

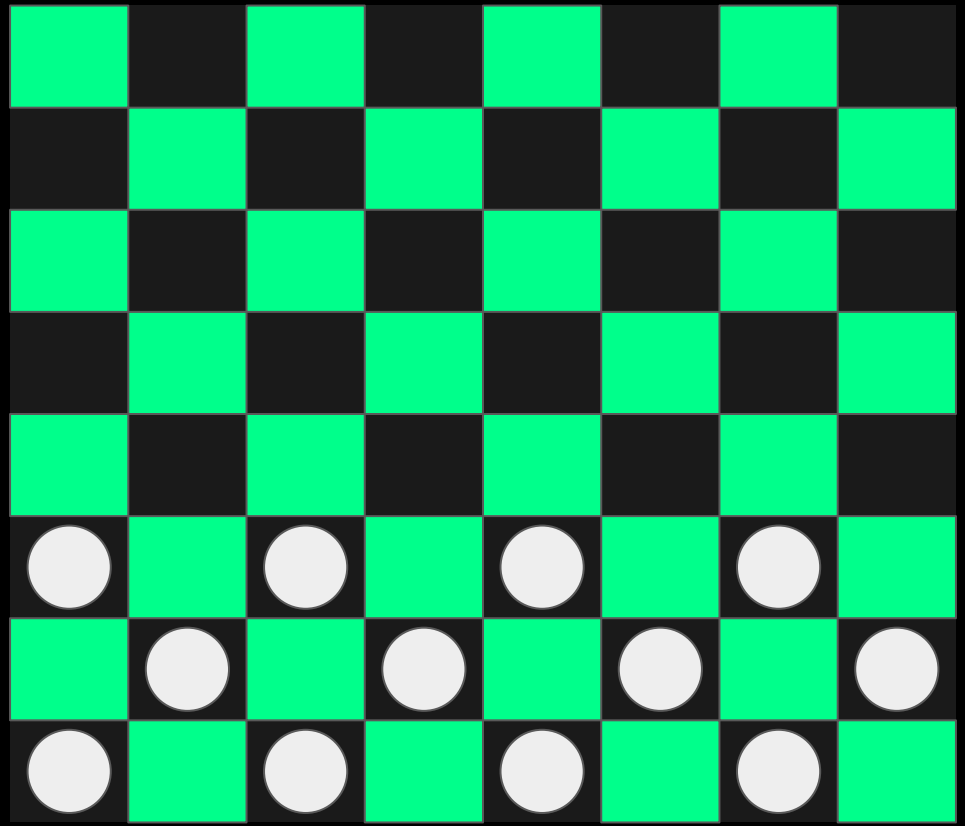
Public release on June 6, 2018

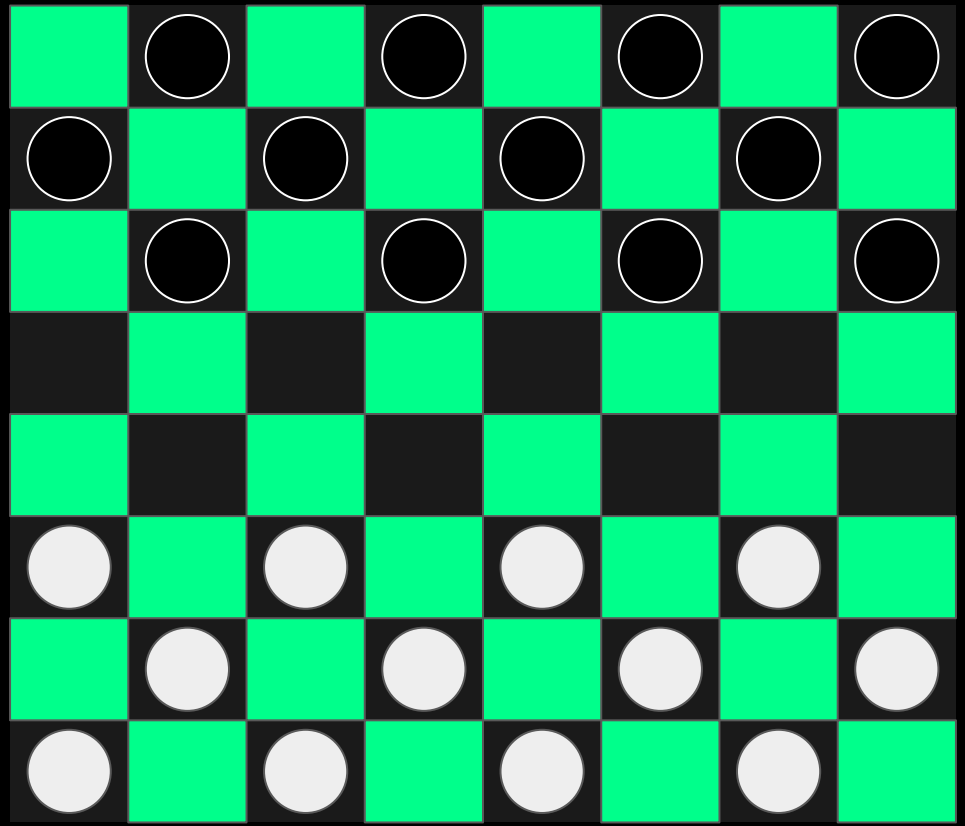
3 months later 3 environments (Checkers, Mighty, Market)

On October 13, 2018 Seoul AI Hackathon with Checkers environment

hackathon.seoulai.com





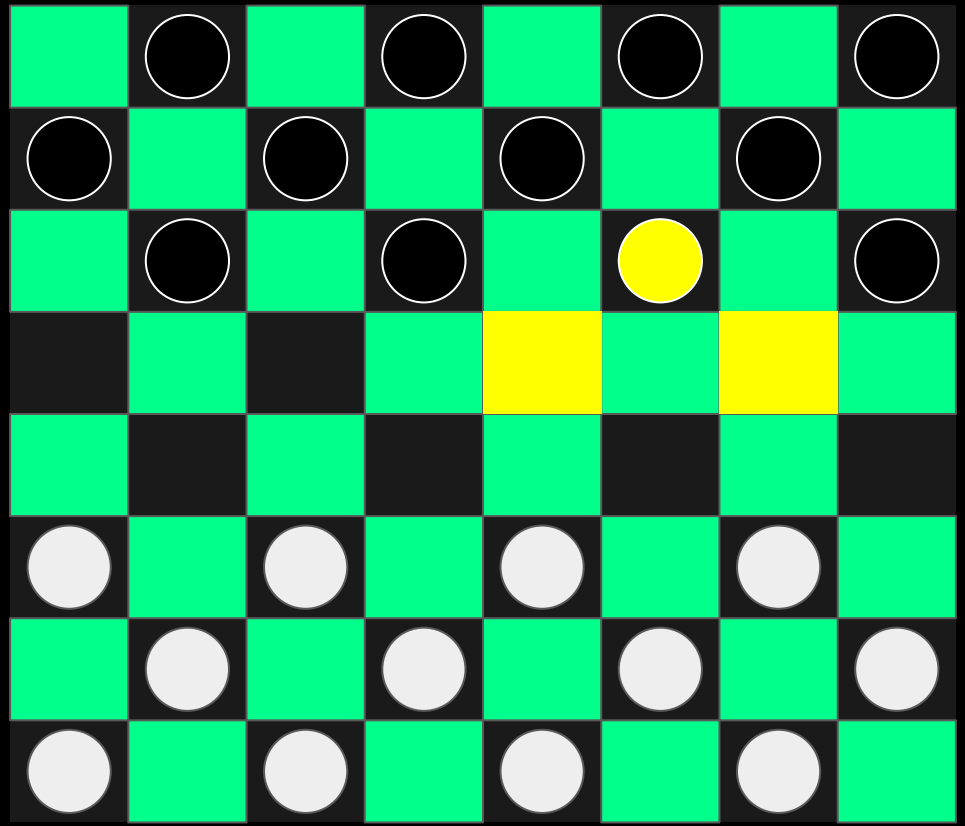


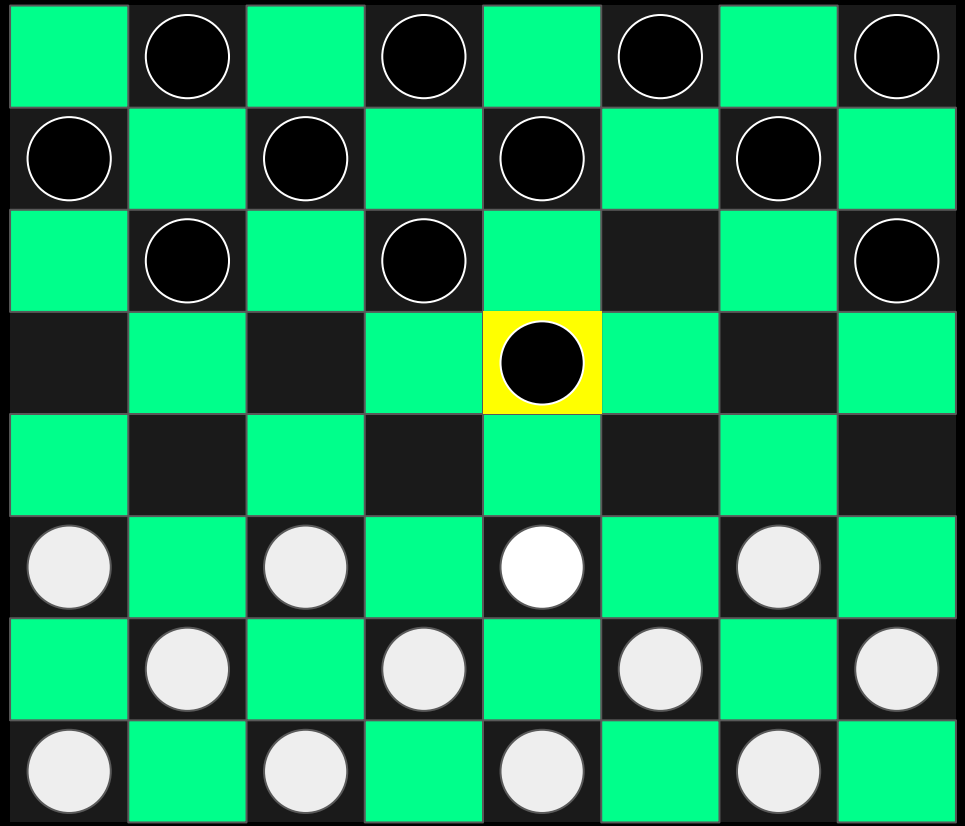
500,000,000,000,000,000,000

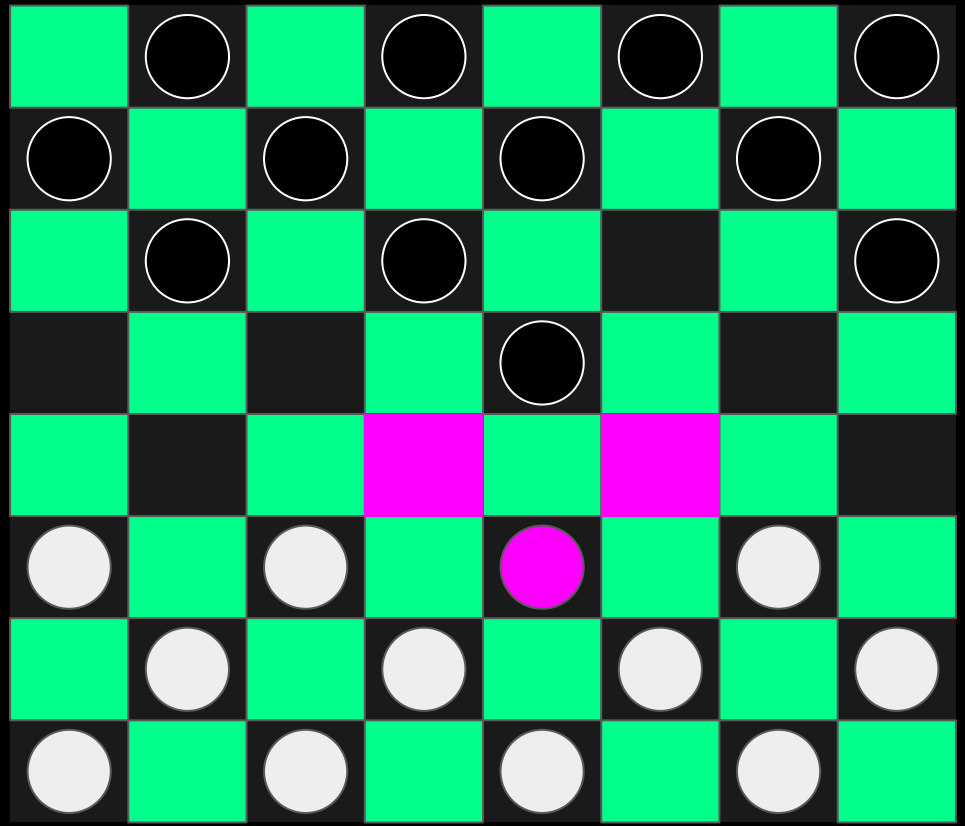
(500 quintillion) combinations

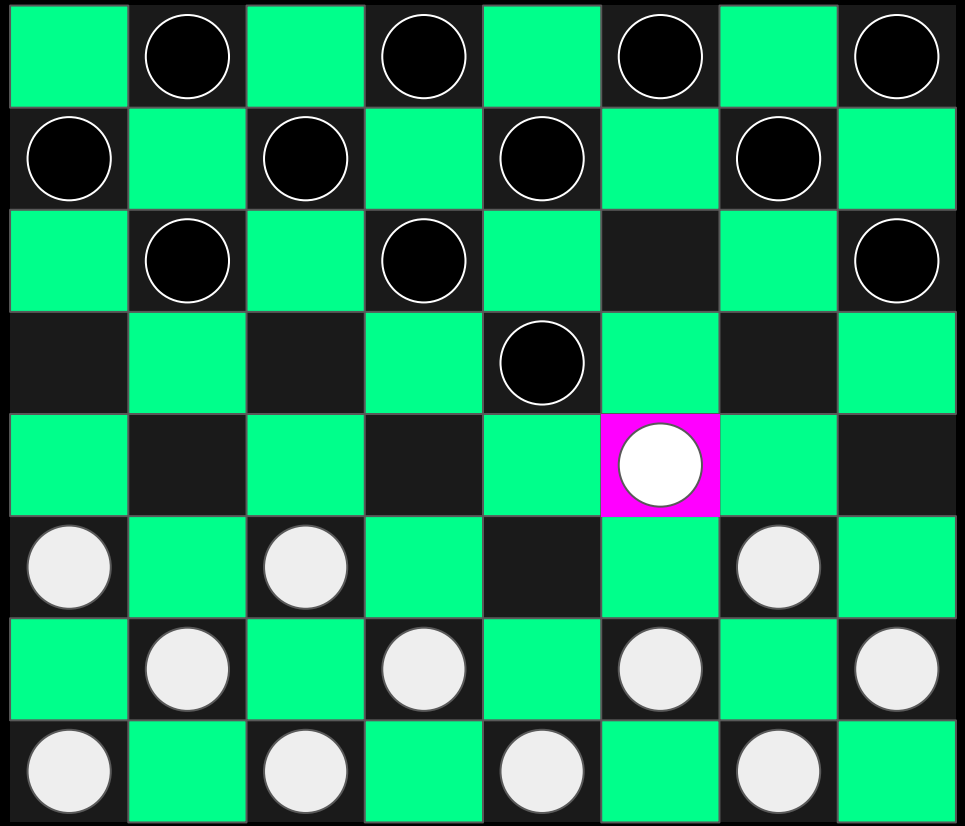
https://www.wired.com/2007/07/the-game-of-che/

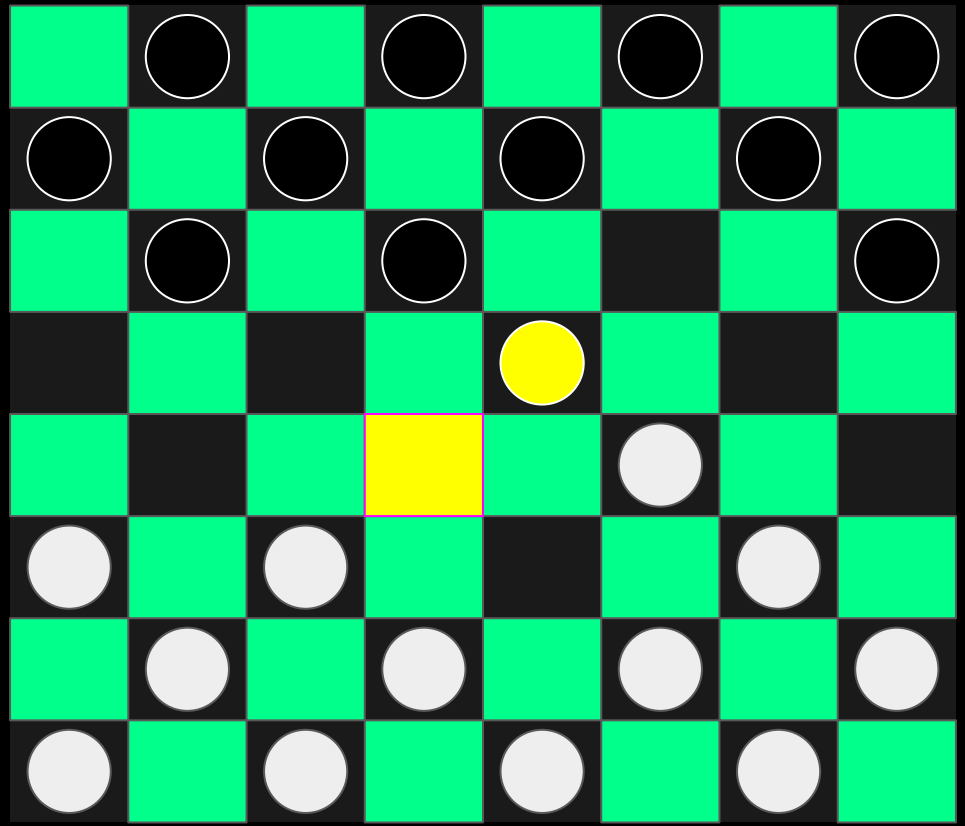
It turns out, there are a mere 500,000,000,000,000,000,000 combinations (500 quintillion) that can be made over the course of a game of checkers.

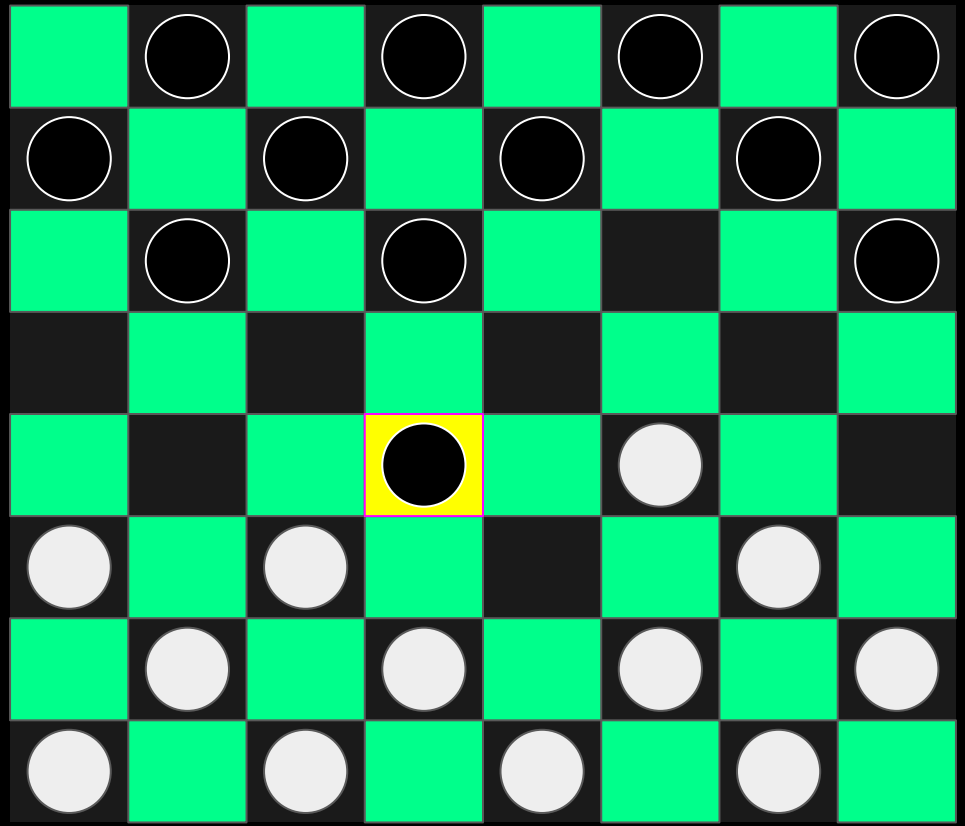


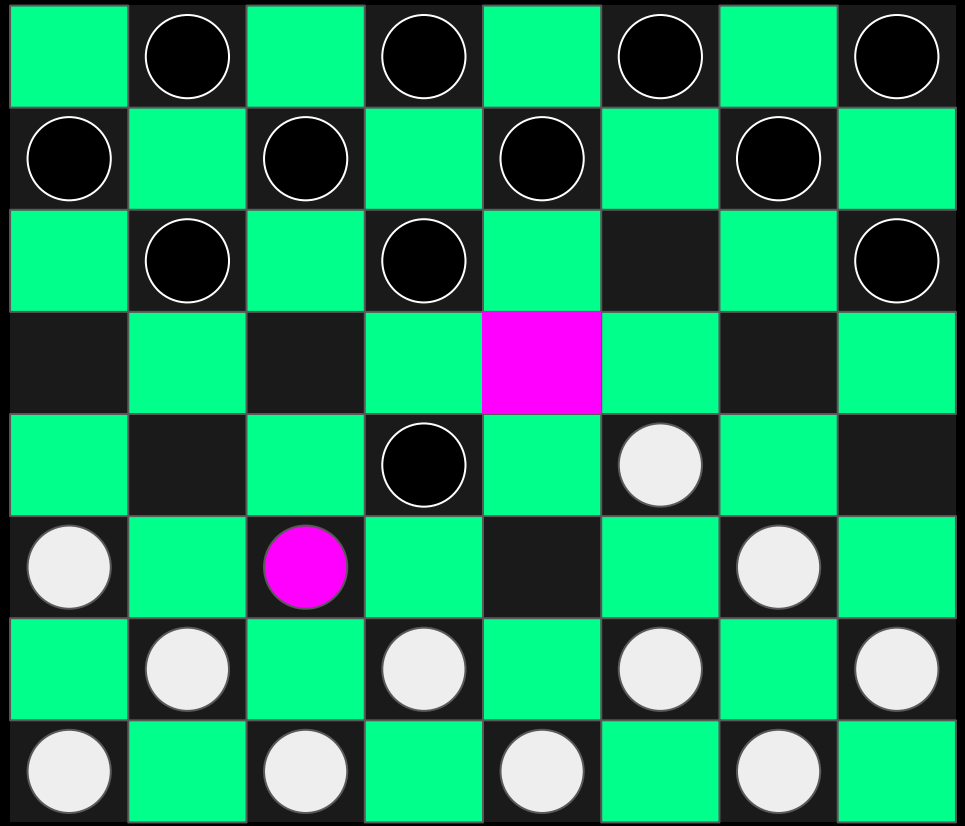


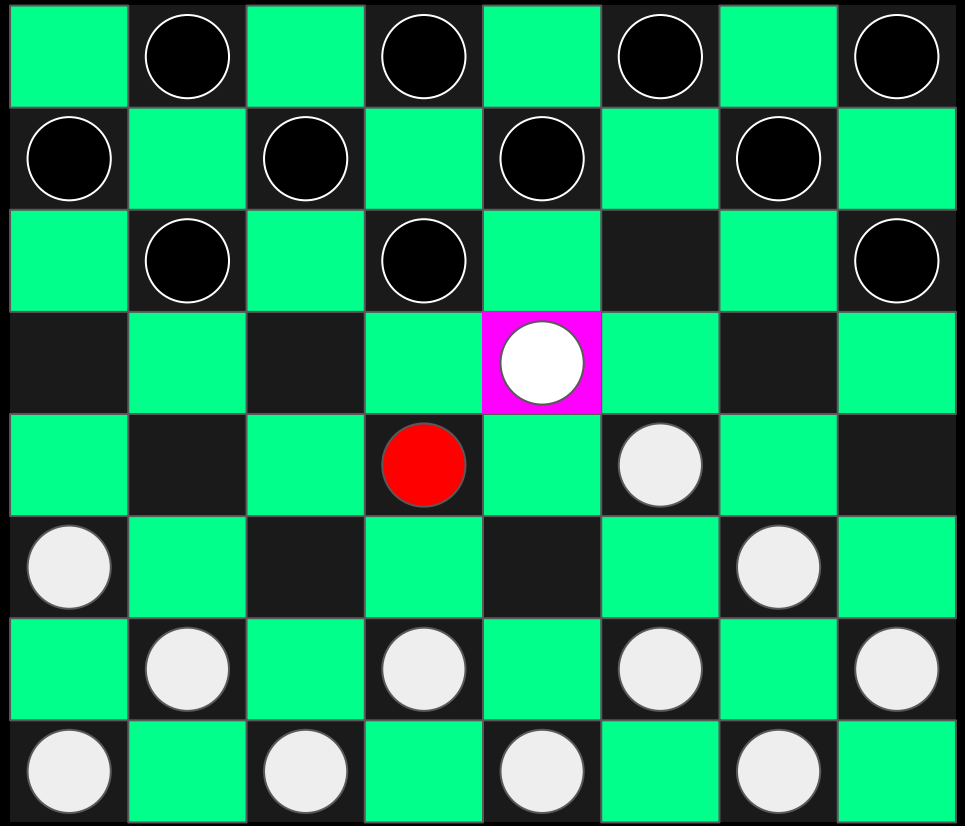


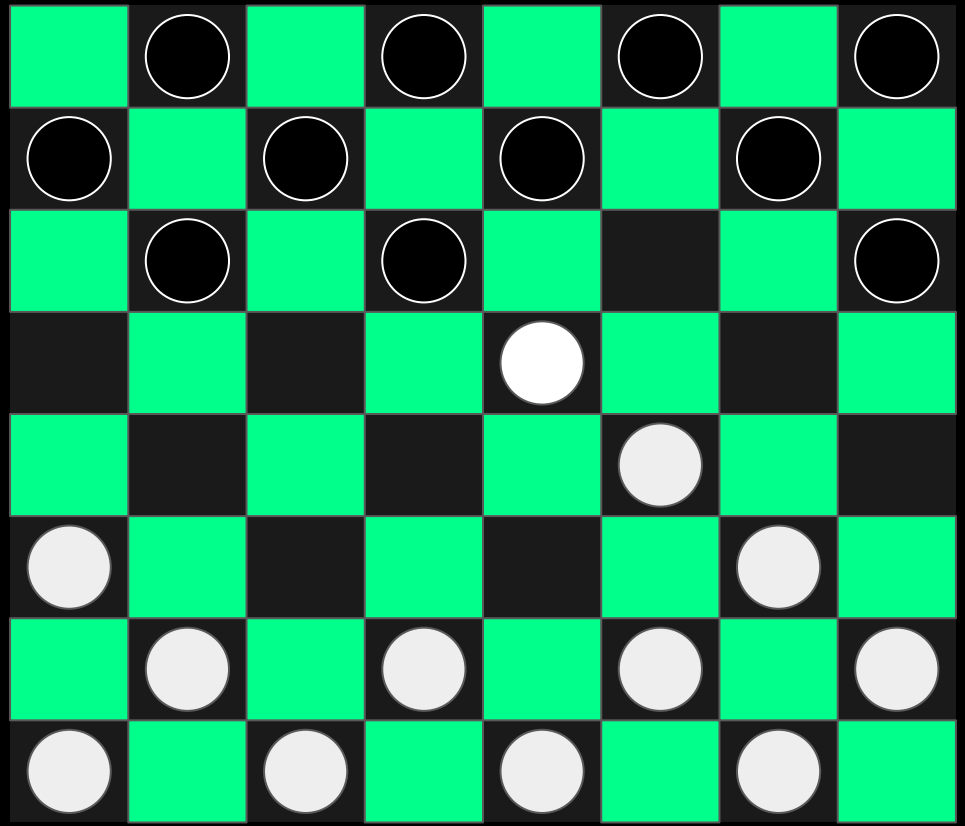




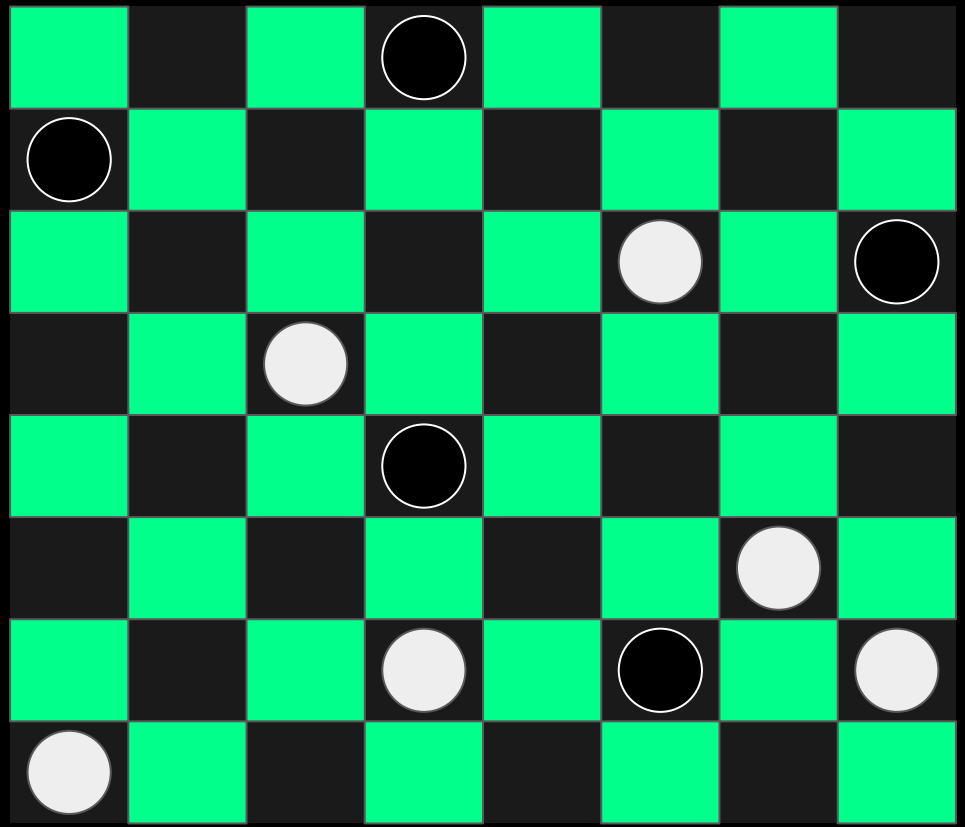




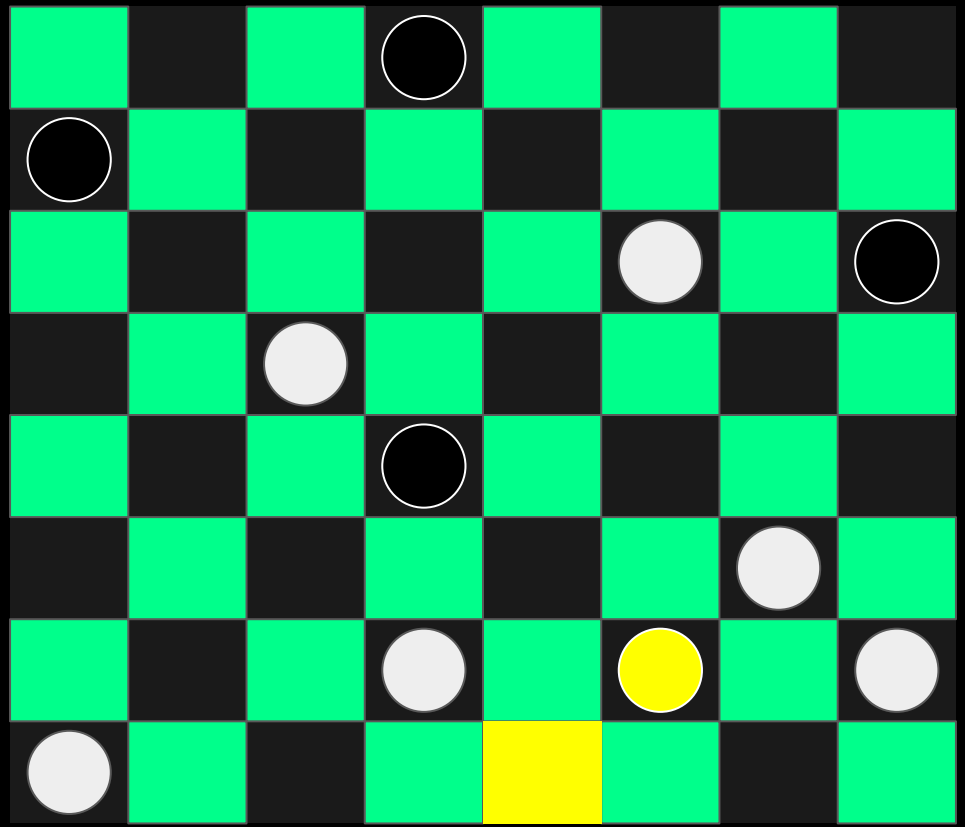




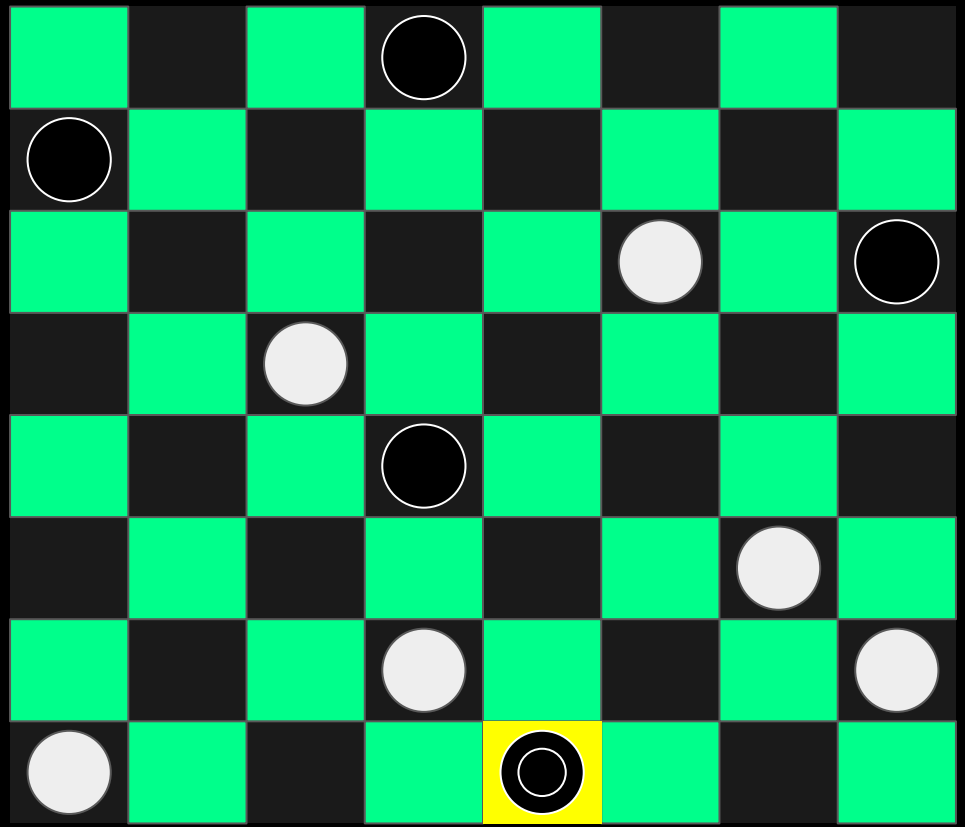
King



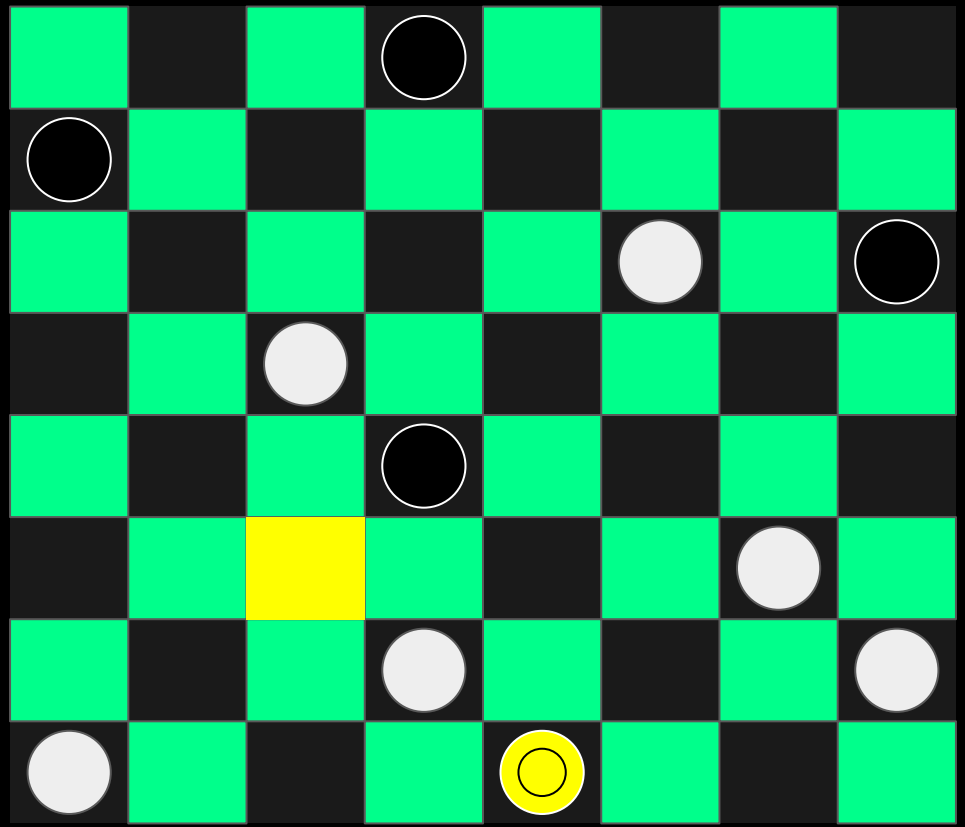
king



king

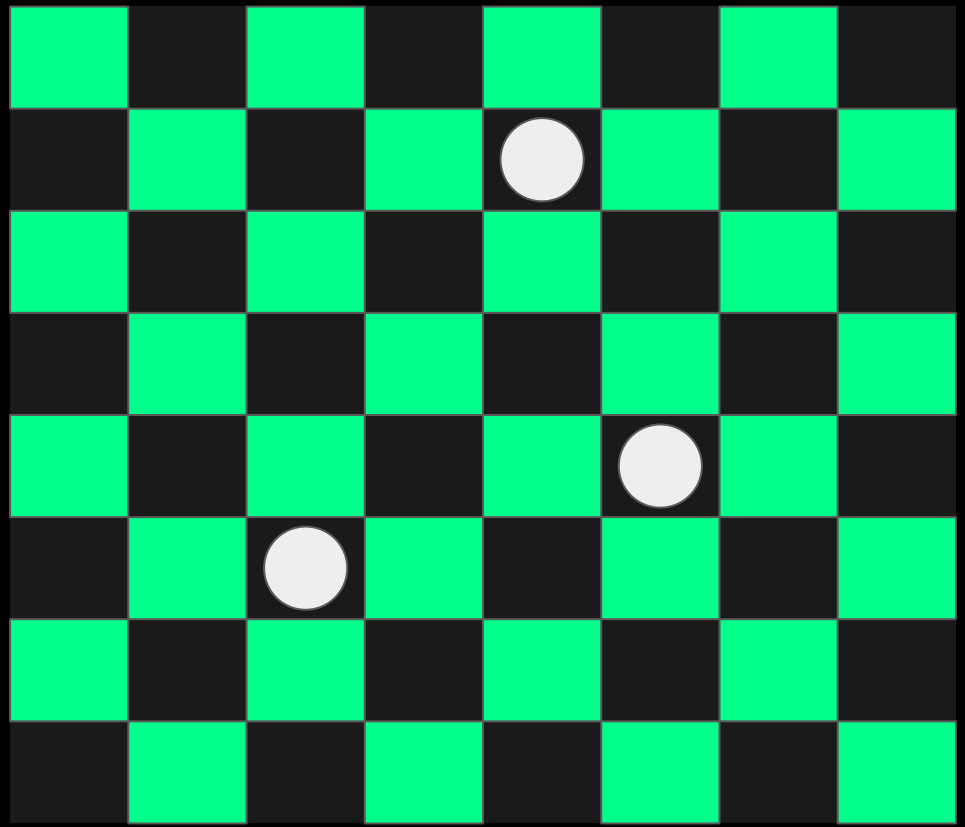


king

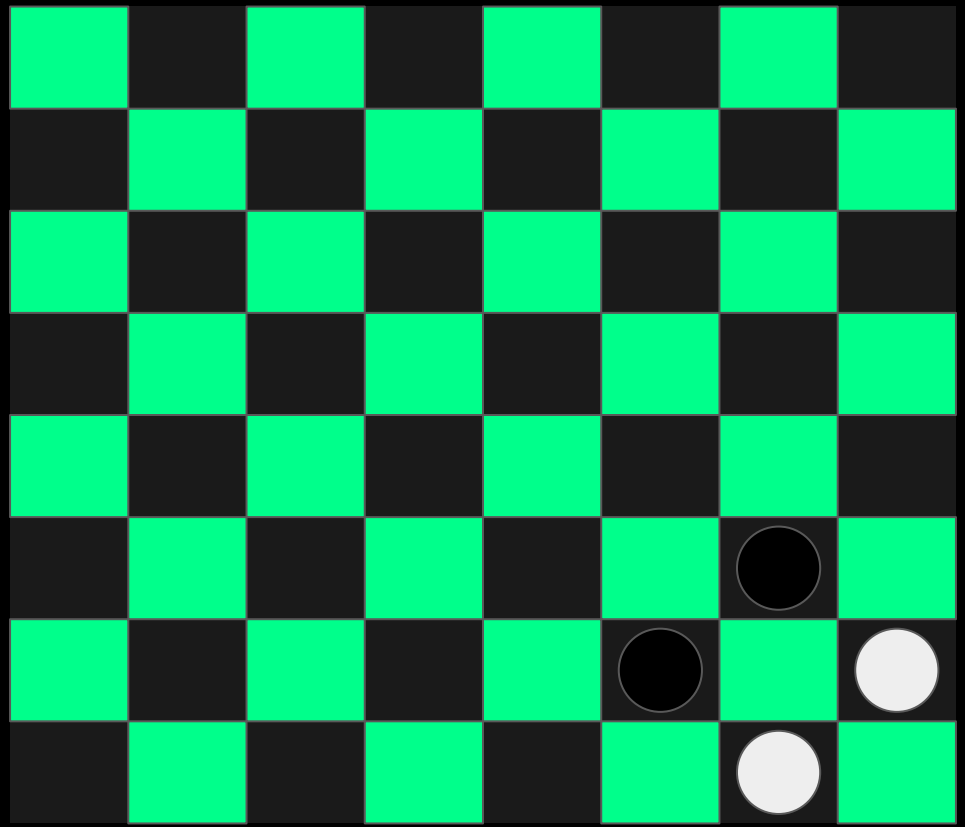


king

Game over



game over #1



game over #2

```
import seoulai_gym as gym
env = gym.make("Checkers")
obs = env.reset()
obs = env.step(agent, from_row, from_col, to_row, to_col)
env.render()
env.close()
```

```python
class RandomAgent(Agent):
  def __init__(self,
      name: str,
      ptype: int,
  ):
    super().__init__(name, ptype)

  def act(self,
    board: List[List],
  ) -> Tuple[int, int, int, int]:
    # decide where to move based on the current state of the game

  def consume(self,
    board: List[List],
    reward: float,
    done: bool,
  ) -> None:
    # evaluate previous action
```

Board

```
board_list2numpy(List[List[Piece]]) -> np.array
```

```
array([[2., 0., 2., 0., 2., 0., 2., 0.],
       [0., 2., 0., 2., 0., 2., 0., 2.],
       [2., 0., 2., 0., 2., 0., 2., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 1., 0., 1., 0., 1., 0., 1.],
       [1., 0., 1., 0., 1., 0., 1., 0.],
       [0., 1., 0., 1., 0., 1., 0., 1.]])
```

Board

```
enc = BoardEncoding()
enc.dark = 99
enc.light = 33
board_numpy = board_list2numpy(obs, enc)

board_list2numpy(board, enc)

array([[99.,  0., 99.,  0., 99.,  0., 99.,  0.],
       [ 0., 99.,  0., 99.,  0., 99.,  0., 99.],
       [99.,  0., 99.,  0., 99.,  0., 99.,  0.],
       [ 0.,  0.,  0.,  0.,  0.,  0.,  0.,  0.],
```

Rewards

```
default
invalid_move
move_opponent_piece
remove_opponent_piece
become_king
opponent_no_pieces
opponent_no_valid_move
```

```
import seoulai_gym as gym
env = gym.make("Checkers")
rewards_map = {
  "default": 1.0,
  "invalid_move": 0.0,
}
env.update_rewards(rewards_map)
```

https://github.com/seoulai/gym/blob/checkers-dqn/checkers_dqn.py