

|   |                                       |   |  |  |
|---|---------------------------------------|---|--|--|
|  |                                       | <p><i>Akademia Górniczo-Hutnicza<br/>w Krakowie</i></p> |  | <p><b>MĘDRAŁA<br/>RADOSŁAW</b></p> <p><b>MIKOS<br/>WERONIKA<br/>284414</b></p> |
| <p>Wydział:<br/><b>WEAiB</b></p>  | <p>Rok studiów:<br/><b>II</b></p>     | <p>Rok akademicki:<br/><b>2019/2020</b></p>             | <p>Kierunek:<br/><b>Informatyka i elektronika medyczna</b></p> | <p>Grupa:</p>  |
| <p>Temat:</p> <p><b>Introduction to Artificial Intelligence</b></p>               |                                       |   |  |  |
| <p>Data wykonania:</p>  | <p>Data oddania:<br/>23.04.2020r.</p> | <p>Data zaliczenia:</p>                                 | <p>OCENA:</p>  |  |

## 1. Cel ćwiczenia

Celem ćwiczeń zapoznanie się z biblioteką EasyAI oraz wykonanie probabilistycznego wariantu jeden z gier.

## 2. EasyAI

**EasyAI** to framework sztucznej inteligencji wykorzystywany do tworzenia programów gier dla dwóch graczy, takich jak Tic Tac Toe, Connect 4, Hexapawn itp. Został on napisany w języku programistycznym Python i ułatwia zdefiniowanie mechanizmów gry oraz umożliwia rozgrywki z komputerem.

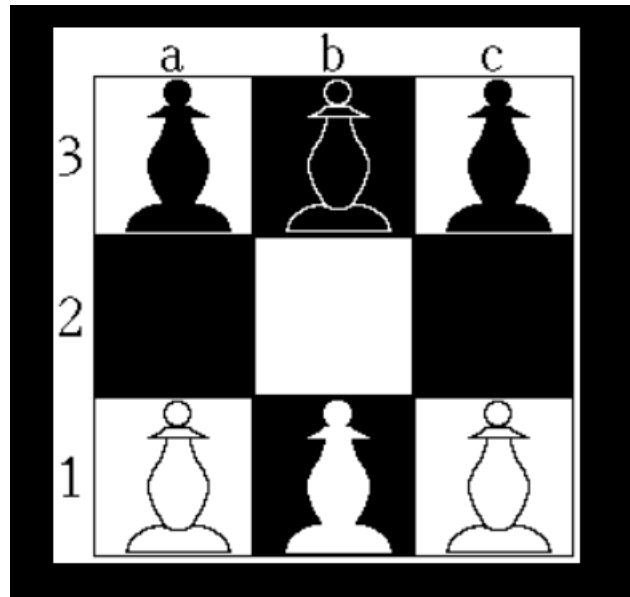
## 3. Hexapawn

**Hexapawn** to gra deterministyczna dla dwóch graczy wymyślona przez Martina Gardnera w 1962r. Została stworzona w celu zilustrowania zasady działania maszyny samouczącej.

Na szachownicy  $n \times m$  każdy z graczy rozpoczyna grę z liczbą  $m$  pionków umieszczonych w najbliższym rzędzie. Podobnie jak w szachach pionki poruszają się po linii prostej i chwytają po przekątnej. Nie posiadają uprawnień do podwójnego kroku. Gracz z białymi pionkami rozpoczyna grę.

Aby wygrać, gracz musi:

- przesunąć jeden ze swoich pionków na koniec planszy
- uniemożliwić ruch drugiego z graczy



#### 4. Octaspawn

Octaspawn jest to wariant gry hexapawn zawartej w bibliotece EasyAI. Na planszy 4x4, każdy z graczy posiada 4 pionki. W chwili wykonania ruchu przez gracza istnieje 10 % szansy na przywrócenie jednego pionka na pozycję startową (każdego schwytanego z równą szansą).

#### 5. Wyniki

Grę przetestowano przy dwóch wariantach parametru *depth* klasy Negamax. Otrzymane wyniki przedstawiono w poniższych tabelach.

*depth=4*

| Numer rozgrywki | Średni czas ruchu w grze [s] | Numer wygrywającego gracza |
|-----------------|------------------------------|----------------------------|
| 1               | 0.03617036666666666          | 1                          |
| 2               | 0.013455727272727269         | 1                          |
| 3               | 0.014744223076923059         | 1                          |
| 4               | 0.0193052                    | 1                          |
| 5               | 0.018170199999999987         | 1                          |
| 6               | 0.012005454545454571         | 1                          |
| 7               | 0.012992490909090907         | 1                          |
| 8               | 0.011230009090909088         | 1                          |
| 9               | 0.017961145454545493         | 1                          |

Gracz 1 wygrał 9 rozgrywek.

Gracz 2 wygrał 0 rozgrywek.

$depth=10$

| Numer rozgrywki | Średni czas ruchu w grze [s] | Numer wygrywającego gracza |
|-----------------|------------------------------|----------------------------|
| 1               | 0.5988382444444444           | 1                          |
| 2               | 0.5455231624999998           | 2                          |
| 3               | 0.7039738900000001           | 2                          |
| 4               | 0.3782751461538456           | 1                          |
| 5               | 0.5907688285714302           | 1                          |
| 6               | 0.5089365538461533           | 1                          |
| 7               | 0.9609723636363647           | 1                          |
| 8               | 0.4744319272727279           | 1                          |
| 9               | 0.46843184166666657          | 2                          |
| 10              | 0.6653716142857148           | 1                          |

Gracz 1 wygrał 7 rozgrywek.

Gracz 2 wygrał 3 rozgrywki.

## 6. Wnioski

Biblioteka EasyAI pozwala na zapoznanie się ze wstępem do sztucznej inteligencji i umożliwia początkującym programistom próbować swoich sił w tworzeniu gier AI. Przejrzyste oraz czytelne skrypty wraz z przydatnymi komentarzami są dużymi zaletami biblioteki.

Problem, z jakim napotkano się podczas tworzenia projektu, było zaimplementowanie części algorytmu związanej z losowością powrotu pionka na plansze. Przypisanie nazwy wszystkim pionkom w celach informacyjnych, który z pionków został złapany stanowiło największą niedogodność w całym zadaniu. Porównano algorytmy z Negamax z alpha-beta pruning dla różnych wartości depth, niestety nie udało się to dla Negamax bez alpha-beta pruning.