

Wydział Elektroniki  
Projektowanie algorytmów i metody sztucznej inteligencji - Projekt 3

Temat projektu: **IMPLEMENTACJA GRY W KÓŁKO I KRZYŻYK Z WYKORZYSTANIEM  
ALGORYTMU MIN-MAX Z ALFA-BETA CIĘCIAMI**

Projekt wykonał: Michał Nowak(235213)

data: 19.05.2018  
data oddania projektu: 5.06.2018

Prowadzący: dr inż. Łukasz Jeleń

## 1. Krótkie wprowadzenie:

Celem projektu było zaimplementowanie popularnego algorytmu min-max z alfa-beta cięciami, w grze „kółko i krzyżyk”.

## 2. Opis tworzonej gry wraz z wytłumaczeniem stosowanych technik SI:

a) Opis gry:

Tic Tac Toe - „kółko i krzyżyk”. Gra polega na naprzemiennym umieszczaniu symboli (kółka lub krzyżyka) na planszy o wymiarach 3x3, 4x4 lub 5x5. Gracz, który jako pierwszy umieści odpowiednią ilość swoich symboli 3, 4 lub 5 w jednej linii wygrywa. Każda wygrana to jeden punkt dla gracza. Po zdobyciu trzech punktów gra się kończy.

Przed rozpoczęciem rozgrywki gracz posiada możliwość zdefiniowania rozmiaru planszy, ilości znaków w linii, jakim symbolem chce grać oraz może zmienić poziom trudności.

Poziomy trudności w grze: easy (głębokość przeszukiwania 0), medium (głębokość przeszukiwania 2), hard (głębokość przeszukiwania 7).

```
*****
****  TIC TAC TOE v1.0  ****
*****
Board size: 3x3
Number of characters in a row: 3
You play as a: X
Computer play as a: O
Difficulty level: hard
Searching depth: 7
*****

1.Start game
2.Change the size of the board
3.Change the number of characters in a row
4.Change your character(O/X)
5.Change the level of difficulty
6.Game instruction
9.Exit
> 
```

Screen.1 Główne menu gry

```
*****
****  TIC TAC TOE v1.0  ****
*****

Player: 0 points
Computer: 0 points

  |  |
 0 | 0 | X
---|---|---
  |  |
 X | X | 0
---|---|---
  |  |
 0 | X | 9
  |  |

Your move> 
```

Screen.2 Przykładowa rozgrywka 3x3

b) Opis rekurencyjnego algorytmu min-max:

Złożoność czasowa algorytmu:  $O(b^m)$

Złożoność obliczeniowa:  $O(bm)$

Złożoność czasowa z cięciami:  $O(b^{m/2})$

Algorytm został zaimplementowany za pomocą czterech opisanych poniżej funkcji:

Funkcja oceniająca stan gry:

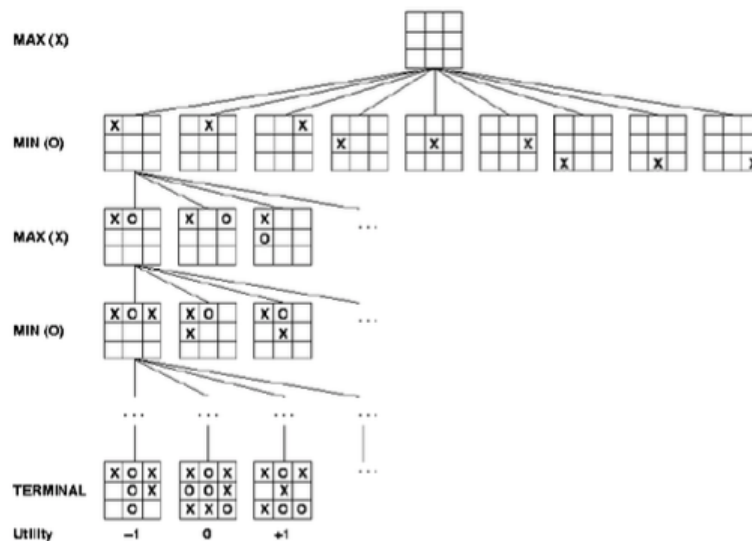
- jeśli w danym węźle (momencie gry) wygrywa gracz funkcja zwraca wartość +10
- jeśli w danym węźle (momencie gry) wygrywa przeciwnik funkcja zwraca wartość -10
- jeśli w danym węźle (momencie gry) jest remis funkcja zwraca wartość 0

Funkcja alfa-beta:

-1 krok: Jeśli dotarliśmy do ostatniego liścia lub ocena gry w danym momencie równa się 0 lub 10 lub -10 przerwij rekurencję i zwróć ocenę gry.

-2 krok: Jeśli funkcja została wywołana dla gracza maksymalizującego ocenę gry, wywołaj rekurencyjnie funkcję alfa-beta dla każdego z wolnych pól (tak aby przejść przez drzewo gry, patrz Rys.1) oraz zapisz zwróconą rekurencyjnie (jeśli jest większa od tej w momencie wywołania) ocenę gry do zmiennej alfa. Jeśli  $\alpha \geq \beta$  przerwij rekurencję i zwróć alfa.

-3 krok: Jeśli funkcja została wywołana dla gracza minimalizującego ocenę gry, wywołaj rekurencyjnie funkcję alfa-beta dla każdego z wolnych pól (tak aby przejść przez drzewo gry, patrz Rys.1) oraz zapisz zwróconą rekurencyjnie (jeśli jest mniejsza od tej w momencie wywołania) ocenę gry do zmiennej beta. Jeśli  $\alpha \geq \beta$  przerwij rekurencję i zwróć beta.



Rys.1 Drzewo gry kółko i krzyżyk, dla planszy 3x3

Funkcja min-max: Zwróć to co zwróci wywołana funkcja alfa-beta z podanymi następującymi argumentami: gracz minimalizujący,  $\alpha = 10$ ,  $\beta = +10$

Funkcja znajdujący optymalny ruch za pomocą funkcji min-max:

-1 krok: Dla każdego z wolnych pól zapisz to co zwróci funkcja min-max i porównaj z -10. Jeśli jest większe bądź równe zapisz dany ruch do tablicy najlepszych możliwych ruchów.

-2 krok: Wylosuj najlepszy ruch z tablicy najlepszych ruchów i zwróć go.

### 3. Podsumowanie i wnioski:

Plansza 3x3, 3 znaki w rzędzie wygrywają:

- poziom trudności hard: Wygrana nie jest możliwa
- poziom trudności medium: Wygrana jest możliwa, jednak w znacznej mierze przypadków dochodzi do remisu
- poziom trudności easy: Bardzo łatwo można wygrać

Plansza 4x4, 3 znaki w rzędzie wygrywają:

- poziom trudności hard: Wygrana zależy tylko od tego kto pierwszy zacznie rozgrywkę. Ten kto zaczyna zawsze wygrywa
- poziom trudności medium: Wygrana zależy tylko od tego kto pierwszy zacznie rozgrywkę. Ten kto zaczyna zawsze wygrywa
- poziom trudności easy: Wygrana zależy tylko od tego kto pierwszy zacznie rozgrywkę. Ten kto zaczyna zawsze wygrywa

Plansza 4x4, 4 znaki w rzędzie wygrywają:

- poziom trudności hard: Wygrana nie jest możliwa
- poziom trudności medium: Wygrana nie jest możliwa
- poziom trudności easy: Bardzo łatwo można wygrać

Plansza 5x5, 3 znaki w rzędzie wygrywają:

- poziom trudności hard: Gra staje się niegrywalna ( zbyt długi czas oczekiwania na ruch )
- poziom trudności medium: Wygrana zależy tylko od tego kto pierwszy zacznie rozgrywkę. Ten kto zaczyna zawsze wygrywa
- poziom trudności easy: Wygrana zależy tylko od tego kto pierwszy zacznie rozgrywkę. Ten kto zaczyna zawsze wygrywa

Plansza 5x5, 4 znaki w rzędzie wygrywają:

- poziom trudności hard: Gra staje się niegrywalna ( zbyt długi czas oczekiwania na ruch )
- poziom trudności medium: Wygrana jest możliwa, jednak w znacznej mierze przypadków dochodzi do remisu
- poziom trudności easy: Bardzo łatwo można wygrać

Plansza 5x5, 5 znaków w rzędzie wygrywa:

- poziom trudności hard: Gra staje się niegrywalna ( zbyt długi czas oczekiwania na ruch )
- poziom trudności medium: Wygrana jest możliwa, jednak w znacznej mierze przypadków dochodzi do remisu
- poziom trudności easy: Bardzo łatwo można wygrać

Największą zaletą algorytmu min-max jest jego łatwość implementacji. Największą wadą natomiast długi czas oczekiwania na wykonanie algorytmu przy większych drzewach gry, tj. plansza 5x5. Dla umożliwienia gry dla większych planszy, należy zmniejszyć głębokość przeszukiwania drzewa gry.

### 4. Literatura:

<http://lukasz.jelen.staff.iar.pwr.edu.pl/downloads/files/wyklad11www.pdf>

[https://pl.wikipedia.org/wiki/Algorytm\\_min-max](https://pl.wikipedia.org/wiki/Algorytm_min-max)

[https://pl.wikipedia.org/wiki/Algorytm\\_alfa-beta](https://pl.wikipedia.org/wiki/Algorytm_alfa-beta)