

Dokumentacja do  
WSI - ćwiczenie 3  
Dwuosobowe gry deterministyczne

*Wykonano przez Marfenko Mykhailo*

## Struktura programu

Program jest podzielony na 3 pliki:

minimax.py – Ten plik zawiera główną funkcję minimax alpha-beta oraz funkcje best\_move\_min i best\_move\_max dla min / max AI odpowiednio która zwraca jeden z najlepszych kroków, oraz funkcjonalność sprawdzania wyniku gry: win, tie, loss poprzez funkcję check\_win().

tic\_tac\_toe.py – Główny script, który pozwala na granie w grę kółko i krzyżyk na planszy NxN. Pętla gry jest zaimplementowana w funkcji play\_game() która przyjmuje jako parametry informację o tym, jaki tryb gry (PvP, AI\_vs\_Player, AI\_vs\_AI), kto pierwszy zaczyna grę, kto czym gra (X, O). Te parametry są {nadane} w konsoli przez użytkownika.

experiments.py – tutaj znajdują się funkcje dla analizy zależności parametru depth algorytmu minimax. Za pomocy biblioteki matplotlib, skrypt pokazuje wyniki gry dla dwóch sztucznych inteligencji z różnym parametrem depth przez dużo rund.

Rozpatrzmy działanie programu z pliku tic\_tac\_toe.py:

Przed rozpoczęciem gry użytkownik ma możliwość ustawienia parametrów gry poprzez konsolę, takich jak wielkość pola. Złożoność sztucznej inteligencji i kto zaczyna pierwszy:

1) Najpierw użytkownik musi wybrać parametr N. To znaczy rozmiar pola, który jest obliczany jako  $N \times N$ .

2) Kolejnym krokiem jest wybór typu gry: Może być to (Player vs Player; AI vs Player oraz AI vs AI).

3) Kolejnym krokiem jest wybranie tego kto pierwszy zaczyna grę jeżeli jest wybrana opcja Player vs Player. Jeżeli była wybrana opcja AI vs Player albo AI vs AI, to kolejnym krokiem jest wybranie poziomu trudności dla AI (depth).

4) Zatem jest możliwość wybrania tego, kto gra X / O oraz tego, kto pierwszy zaczyna grę.

(W przypadku błędnego wprowadzenia danych o wyborze będzie wyświetlony odpowiedni komunikat i można będzie poprawić swój wybór.)

Wyglądać to może w taki sposób:

```

Enter the dimension of the grid
3
Pick game mode (1 - PvP / 2 - PvAI / 3-AIvAI)
2
Pick difficulty for AI (depth)
5
Who starts first? (1-Player / 2-AI)
bla bla bla
Try again
1
First player plays as X or O? ('X' / 'O')
0

```

Dla takich danych gra może wyglądać w następujący sposób:

```

- - -
- - -
- - -

Pick spot [row col]
1 1

- - -
- O -
- - -

best_moves for min player (X) : [(0, 0), (0, 2), (2, 0), (2, 2)]

- - -
- O -
- - X

Pick spot [row col]
0 2

- - O
- O -
- - X

best_moves for min player (X) : [(2, 0)]

- - O
- O -
X - X

```

```
Pick spot [row col]
0 0

O - O
- O -
X - X

best_moves for min player (X) : [(2, 1)]

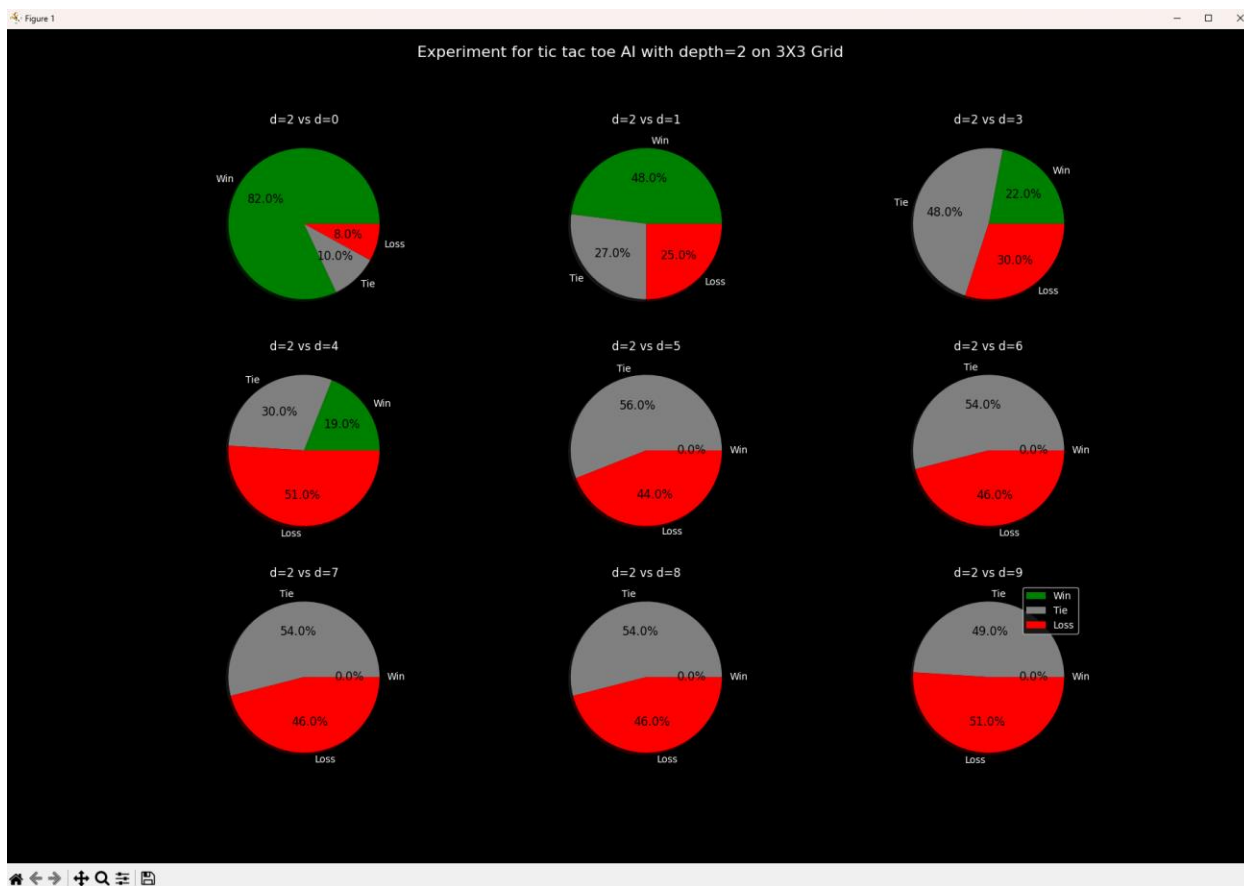
O - O
- O -
X X X

Game over! Player X has won!
```

## **Eksperymenty**

### **Eksperyment 1 na planszę 3X3 dla AI z depth=2:**

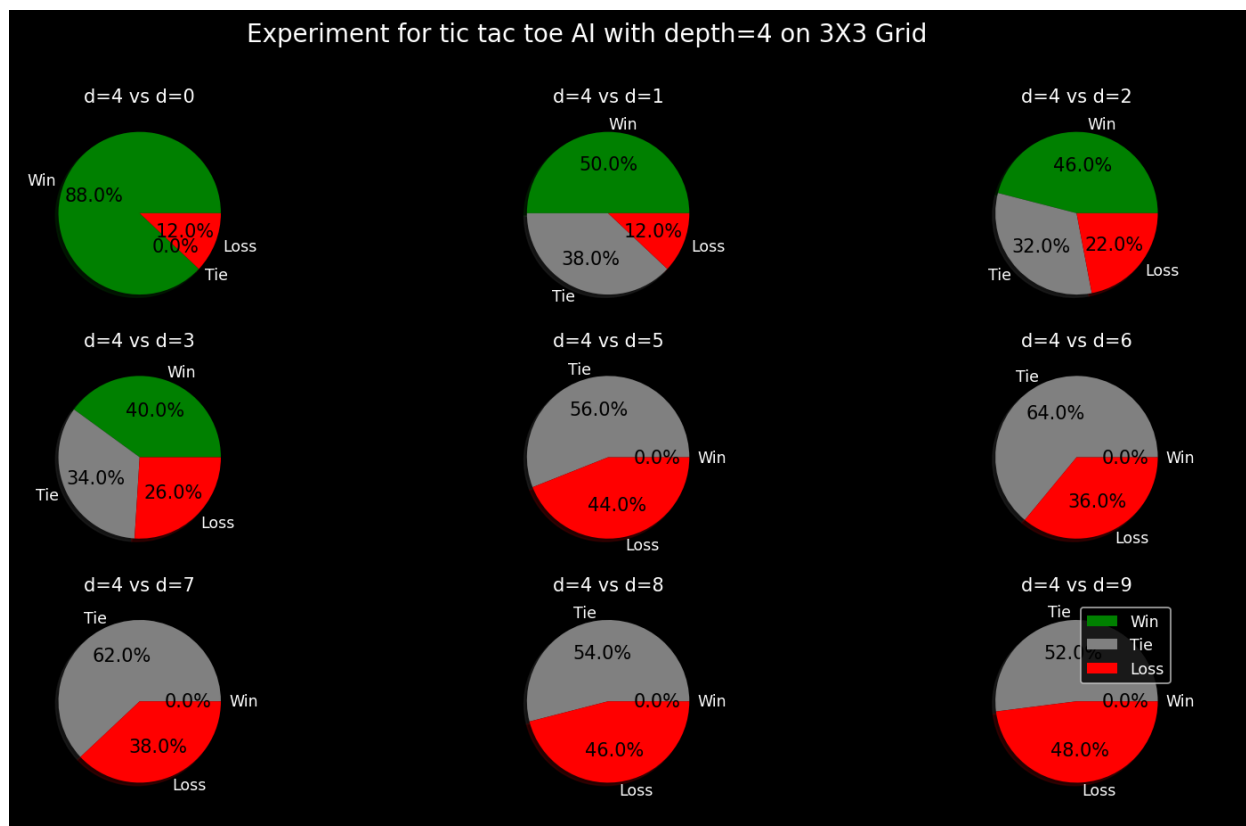
Opis: Tutaj są wyniki 100 rund w grę kółko i krzyżyk pomiędzy AI z depth = 2 a AI z depth = (0; 9) bez 2 na planszy 3X3.



Analiza: Zauważymy, że dla słabszych AI z depth 0 i 1 nasz AI z depth 2 w większości przypadkach wygrywa, a nie zawsze, bo depth = 2 nie wystarczająco, żeby wygrywać zawsze nawet z łatwymi AI. Tak jest przez to że AI z depth = 2 nie zawsze gra optymalnie. Z tego powodu on w 50% przypadkach przegrywa mocniejszym AI. Można zauważyć także to, że wyniki mocniejszych AI nie bardzo się różnią. Np. AI z depth=6 tyle samo razy wygrał jak AI z depth=8. Jest tak dlatego, że depth=6 już jest wystarczająco, żeby grać optymalnie w 90% przypadkach, więc nie ma dużej różnicy pomiędzy depth=6 i depth=9. Przynajmniej w takiej grze na plansze 3X3.

### **Eksperyment 2 na planszę 3X3 dla AI z depth=4:**

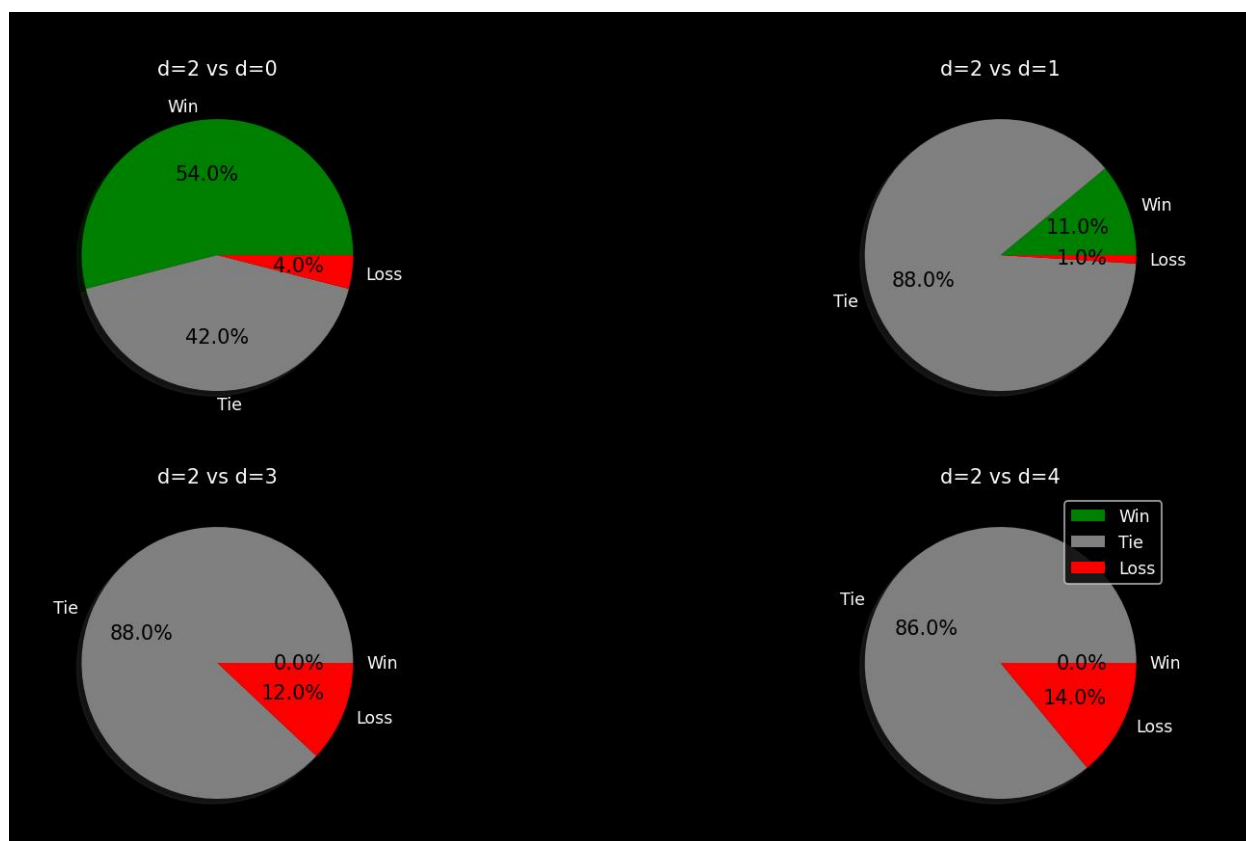
Opis: Tutaj są wyniki 50 rund w grę kółko i krzyżyk pomiędzy AI z depth = 4 a AI z depth = (0; 9) bez 4 na planszy 3X3.



Analiza: Nie trudno zauważyć, że nasz AI z depth=4 teraz jest znacznie mocniejszy. I wygrywa częściej ze słabszymi AI. Ale nadal przegrywa mocniejszym choć widać, że wynik remisowy jest częściej.

### **Eksperyment 3 na planszę 4X4 dla AI z depth=2:**

Opis: Tutaj są wyniki 100 rund w grę kółko i krzyżyk pomiędzy AI z depth = 2 a AI z depth = (0; 4) bez 2 na planszy 4X4



Analiza: Tutaj można zauważyć podobną tendencję jak i w eksperymencie na planszę 3X3. Różnica polega na tym, że AI z większym parametrem depth zajmują sporo czasu na decyzję. I dlatego w tym eksperymencie nie ma AI który gra zawsze optymalnie. Także w tym eksperymencie jest więcej remisów. To jest przez regoły gry. Na takiej planszy bardzo łatwo zapobiec przegraniu.