Projektowanie Algorytmów i Metody Sztucznej Inteligencji		
Temat	Termin zajęć	
Projekt 3 - kółko i krzyżyk	Poniedziałek 18:55	
Imię, nazwisko, numer albumu	Ocena	
Paweł Wójcik, 259341		
Wykładowca		
mgr Marta Emirsajłow		
Kod grupy	Data złożenia sprawozdania	
Y03-51c	21.06.2022r.	

Sprawozdanie nr. 3

Spis treści

1	Wprowadzenie	2
2	Teoretyczny opis algorytmu	2
3	Działanie programu	2
4	Wnioski	3
5	Bibliografia	3

1 Wprowadzenie

Należało stworzyć wybraną grę opartą o algorytm MinMax. Wybrałem grę kółko i krzyżyk. W grze bierze udział użytkownik przeciwko szutcznej inteligencji.

2 Teoretyczny opis algorytmu

Algorytm MinMax pozwala na analizowanie przebiegu gry i na tej podstawie podejmowanie decyzji, które będą dla nas najlepsze- doprowadzą nas do wygranej(max) lecz w tym samym czasie zakładamy, że przeciwnik również podejmuje optymalne z punktu widzenia algorytmu, które prowadzą przeciwnika do wygranej(min). Algorytm w podstawowej wersji działał dobrze i sprawnie dla planszy o wielkości 3x3, lecz dla większych placów gry, długość podejmowania decyzji była nie do przyjęcia. W związku z tym algorytm został rozbudowany o ograniczenie głębokości rekurencji, przy granicy dobranej eksperymentalnie i wynoszacej:

$$depth = \frac{30}{3*size - 2*winrow}$$

Gdzie size to rozmiar planszy, a winrow to ilość znaków w rzędzie jaka jest wymgana do wygranej.

3 Działanie programu

Gra rozpoczyna się po kompilacji programu, na początku jesteśmy proszeni o podanie wielkości planszy, ilości znaków w rzędzie potrzebnych do wygranej oraz gracza, który rozpoczyna grę, gdzie 'O' to zawsze gracz obsługujący program, a 'X' to zawsze gracz AI.

```
Kolko i krzyzyk - Tic Tac Toe

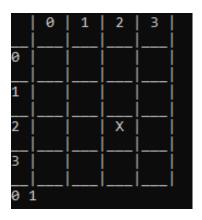
Podaj wielkosc planszy: 4

Podaj ilosc znakow potrzebna do wygranej: 4

Wybierz kto ma zaczac [O- gracz/X- AI]: X_
```

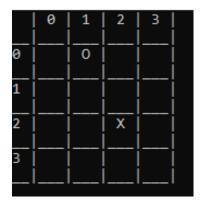
Rysunek 1: Podanie parametrów rozgrywki

Naciskamy enter i pokazuje się pole rozgrywki.



Rysunek 2: Interfejs gry

Ruch wykonujemy podając współrzędne komórki, gdzie chcemy umieścić swój znak, podając pierw wiersz, a następnie kolumnę.



Rysunek 3: Rozgrywka

Ruchy gracza i AI wykonywane są naprzemiennie aż do zakończenia gry, które następuje w momencie osiągnięcia wymaganej ilości znaków w jednym rzędzie lub gdy na planszy nie pozostaje żadne wolne miejsce - remis.



Rysunek 4: Rozgrywka

4 Wnioski

Gra działa poprawnie, pozwala na wybór wielkości pola gry oraz ilości wymaganych znaków w rzędzie do wygranej. W prawidłowy sposób interpretuje sytuacje na planszy i jest w stanie trafnie ocenić czy gra powinna się zakończyć, a jeżeli tak to z jakim wynikiem. Możliwy jest wybór gracza rozpoczynającego rozgrywkę.

W przypadku zaimplementowanego algorytmu MinMax działa on szybko i sprawnie dla planszy o wielkości 3x3. Niestety ze względu na zasadę działania, czyli przeszukiwania wszystkich możliwości ruchu, zwłaszcza dla początkowych ruchów w planszach większych niż 3x3 czas działania jest w mojej ocenie długi. Złożoność algorytmu MinMax wynosi w notacji wielkiego O:

$$O(b^m)$$

Gdzie b to ilość rozgałęzień, czyli wolnych miejsc na planszy, a m to głębokość czyli ilość ruchów analizowanych do przodu.

W celu usprawnienia czasu podejmowania decyzji zastosowana została głębokość rekurencji, która znacznie przyspieszyła działanie algorytmu, ale w niektórych przypadkach powoduje to pogorszenie podejmowanych decyzji tzn. algorytm w najlepszym przypadku zdaje się dążyć do remisu.

5 Bibliografia

- https://eduinf.waw.pl/inf/utils/001_2008/0415.php
- https://www.math.uni.lodz.pl/~marta/2012_2013_z/zz/sztuczna3.pdf

- https://pl.wikipedia.org/wiki/Algorytm_min-max
- https://monitorfx.pl/algorytm-min-max/
- https://youtu.be/l-hh51ncgDI