|  |  |  |  |
| --- | --- | --- | --- |
| **Data: 20.06.2022** | **Dzień: poniedziałek** | | |
| **Projekt 3** | **Godzina: 15:15** | | |
| **Temat ćwiczenia:**  Projektowanie Algorytmów i Metody Sztucznej Inteligencji | | | |
| **Prowadzący :**  **Mgr Inż. Marta Emirsajłow** | | | |
| **L.p.** | **Imię i nazwisko** | **Nr indeksu** | **Ocena:** |
| 1. | Maciej Płonka | 259347 |  |

Spis treści

[Wstęp 1](#_Toc106582022)

[MinMax w warcabach 2](#_Toc106582023)

[Opis 2](#_Toc106582024)

[Wnioski i podsumowanie 2](#_Toc106582025)

[Literatura 3](#_Toc106582026)

# Wstęp

Projekt numer trzy dotyczył zastosowanie Sztucznej Inteligencji w ramach prostej gry stworzonej w C++. Zostały przeze mnie stworzone warcaby o wymiarze 8 na 8. Do dyspozycji gracz ma 12 pionków, którymi może się poruszać. Wraz z dojściem pionka do „bazy” przeciwnika zamienia się on w „damkę”, którą wyróżnia to, że może bić pionki przeciwnika „do tyłu” – pionek w postaci literki r lub q zmienia się na R lub Q. Bicie jest zawsze obowiązkowe.

# MinMax w warcabach

## Opis

Sztuczna inteligencja za pomocą algorytmu MinMax „zagłębia się w drzewo z rozpisanymi możliwymi ruchami” do przodu. Dzięki temu wylicza sobie na tej podstawie, których ruch będzie najbardziej opłacalny dla komputera. Do każdego ruchu została przydzielona pewna waga, która określa pozycję pionka lub jego danych ruch (gdy dany pionek zbija pionka gracza, gdy pionek komputera jest zbijany przez pionek gracza, gdy znajduje się on przy ścianie, czy jest to zwykły pionek, czy też jest to tzw. „Damka”. Przy okazji, program wykorzystuje algorytm alfa-beta, który ma za zadanie szybsze minimalizowanie strat i wybór najlepszego ruchu. Algorytm utrzymuje dwie wartości alfa i beta, które reprezentują minimalny wynik gracza MAX i maksymalny wynik gracza MIN. Początkowo alfa wynosi -♾, a beta + ♾. W miarę postępowania rekursji przedział (alfa; beta) staje się mniejszy i kiedy beta staje się mniejsze niż alfa, oznacza to, że obecna pozycja nie może być wynikiem najlepszej gry przez obu graczy i wskutek tego nie ma potrzeby przeszukiwania głębiej. Dodatkowo za pomocą zmieniania głębokości przeszukiwania drzewa, jesteśmy w stanie ułatwić/utrudnić grę, przeciwko komputerowi.

# Wnioski i podsumowanie

Obraz zawierający tekst

Opis wygenerowany automatycznie

Gra działa poprawnie i szybko, komputerowi nie zajmuje dużo czasu przemyślenie ruchu i wykonanie. Jeśli chodzi o wygrywanie/przegrywanie gier z komputerem, to nie byłem jeszcze w stanie jej wygrać. Komputer jest w stanie obliczyć i sprawdzić 2,3,4 czy 5 ruchów i kombinacji do przodu, lecz ja maksymalnie 2 przez co po pewnym czasie zdobywa on przewagę i już jej nie oddaje. W przypadku, gdy gracz i komputer będzie posiadał damki i będą się one znajdowały po przeciwnych krańcach pól startowych, komputer będzie wykonywał na przemiennie ruch oddalający i przybliżający się do krańca pola, co czyni grę praktycznie niemożliwą do wykonania, chyba, że posiada się mnóstwo czasu.

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst

Opis wygenerowany automatycznie

Obraz zawierający tekst, sprzęt elektroniczny

Opis wygenerowany automatycznie

Algorytm często zaczyna tak samo grę, lecz wraz z moim ruchami gra się rozwija i staje się on mniej przewidywalny. Pomimo dużej złożoności algorytmicznej program działa sprawnie.

# Literatura

<https://pl.wikipedia.org/wiki/Algorytm_alfa-beta>

<https://pl.frwiki.wiki/wiki/Algorithme_minimax>

<https://pl.wikipedia.org/wiki/Algorytm_min-max>  
<https://kcir.pwr.edu.pl/~witold/aiarr/2009_projekty/warcaby/>

<https://repo.pw.edu.pl/info/article/WUT116729/Evolutionary-based+heuristic+generators+for+checkers+and+give-away+checkers#.Yq-4-3ZBxPY>

<https://www.geeksforgeeks.org/minimax-algorithm-in-game-theory-set-1-introduction/>

https://kcir.pwr.edu.pl/~witold/aiarr/2009\_projekty/warcaby/