

## Sprawozdanie z projektu 3 Warcaby

Kurs: Projektowanie algorytmów i metody sztucznej inteligencji  
Prowadzący: Dr inż. Łukasz Jeleń  
Wykonał: Taras Radchenko (nr indeksu 248021)  
Termin zajęć: czwartek 9:15

Wrocław  
1.06.2020

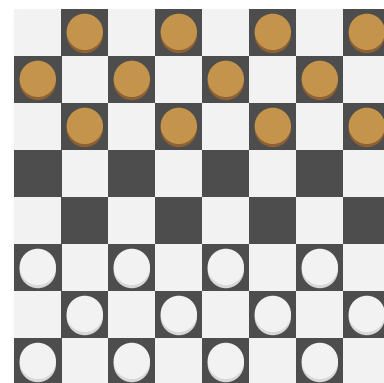
# Wprowadzenie

Sztuczna inteligencja jest ważną dziedziną informatyki. Automatyzacja przemysłu lub inteligentne budynki wymagają algorytmów podejmowania nietrywialnych decyzji. Są to systemy, pracujące bez przerwy, a integracja człowieka często jest uciążliwa lub wcale niemożliwa. Dlatego właśnie ta branża szybko się rozwija. Gry komputerowe są jednym z zakresów stosowania technik SI.

W tym projekcie implementowałem algorytm MinMax dla przeciwnika w warcabach.

## Zasady gry

Gra jest rozgrywana na planszy o rozmiarze 8x8 pól. Pole w lewym dolnym rogu (A1) zawsze jest ciemne, pozostałe są pokolorowane na przemian. Każdy gracz rozpoczyna grę z dwunastoma pionami ustawionymi na ciemniejszych polach planszy w sposób pokazany na ilustracji. Jako pierwszy ruch wykonuje grający pionami białymi, po czym gracze wykonują na zmianę kolejne ruchy. Celem gry jest zabicie albo zablokowanie wszystkich pionów przeciwnika, pozbawiając przeciwnika możliwości wykonania ruchu. Jeśli żaden z graczy nie jest w stanie tego osiągnąć (każdy z graczy wykona po 15 ruchów bez zmniejszania liczby pionów pozostających na planszy), następuje remis.



*Układ do rozpoczęcia gry*

Piony mogą się poruszać o jedno pole do przodu po przekątnej na wolne pola. Bicie pionem następuje przez przeskoczenie sąsiedniego pionu przeciwnika na wolne pole znajdujące się tuż za nim po przekątnej. Zbite piony są usuwane z planszy po zakończeniu ruchu. Piony mogą bić zarówno do przodu, jak i do tyłu. W jednym ruchu wolno wykonać więcej niż jedno bicie tym samym pionem, przeskakując przez kolejne piony lub damki przeciwnika. Bicia są obowiązkowe. Kiedy istnieje kilka możliwych bić, gracz musi wykonać wszystkie. Podczas bicia nie można przeskakiwać więcej niż jeden raz przez ten sam pion.

Pion, który kończy ruch na ostatnim rzędzie planszy, staje się damką. Damki mogą poruszać się o dowolną liczbę pól do przodu lub do tyłu po przekątnej, zatrzymując się na wolnych polach. Bicie damką jest możliwe z dowolnej odległości po linii przekątnej i następuje przez przeskoczenie pionu (lub damki) przeciwnika, za którym musi znajdować się co najmniej jedno wolne pole – damka przeskakuje na dowolne z tych pól i może kontynuować bicie na tej samej lub prostopadłej linii.

## Algorytm SI

Podstawą algorytmu jest funkcja heurystyczna — funkcja, która ocenia szansę gracza na zwycięstwo przy danym położeniu pionów na planszy. W mojej implementacji jest stosowana funkcja, która zwraca 0 jeżeli żaden z graczy nie ma przewagi, liczbę ujemną jeżeli przewagę ma pierwszy gracz (który gra pionami białymi), i dodatnią jeżeli przewagę ma drugi gracz. Im większa jest liczba, tym większe szanse ma gracz na zwycięstwo. Tak więc żeby wygrać, pierwszy gracz ma minimalizować wartość funkcji, drugi zaś — maksymalizować.

Funkcja ma odzwierciedlać cel gracza. W warcabach celem jest zabicie lub zablokowanie pionów przeciwnika. Oczywiście posiadanie damki jest pewną przewagą, ale to raczej nie może wpływać na wartość funkcji heurystycznej, ponieważ to zmienia cel: gracz będzie dążył do jak największej liczby damek, ale nie będzie dążył do zwycięstwa.

W pierwszej implementacji algorytmu funkcja heurystyczna zwracała różnicę w liczbie pionów na planszy. Ta prosta funkcja działała dobrze, ale postanowiłem ją rozszerzyć o wykrywanie pionów zablokowanych. Tak się pojawił blok, oznaczony w kodzie „dostrajanie”. Za każdego pionu przeciwnika, który może wykonać bicie, odejmuję 1 od wartości funkcji. Natomiast za każdego zablokowanego dodaję 1. Po tych zmianach gracz-komputer stał bardziej agresywny i często wymusza wykonać bicie jeżeli on będzie mógł zbić tyle samo lub więcej. Jeżeli znaleziono kilka możliwych ruchów o tej samej ocenie, losowo jest wybierany jeden z nich.

## Struktura kodu

Implementacja sztucznej inteligencji znajduje się w katalogu `inc/ai`. Abstrakcyjne klasy gry takie jak plansza (Board), pion (Checker), gracz (Player) itd. znajdują się w katalogu `inc/game`. Implementacja graficzna znajduje się w `inc/graphics`. Do rysowania jest wykorzystana biblioteka raylib ([raylib.com](https://raylib.com)).

## Wnioski

Wykonując projekt, zapoznałem się z algorytmem MinMax i implementowałem go w grze planszowej. Ten algorytm jest stosunkowo prosty do zrozumienia. „Mózgiem” sztucznej inteligencji jest funkcja heurystyczna, której dobór jest sam w sobie ciekawym zadaniem. Poza tym, spełniłem marzenie i wykonałem grę trójwymiarową.

## Literatura

1. Warcaby, Wikipedia  
[pl.wikipedia.org/wiki/Warcaby](https://pl.wikipedia.org/wiki/Warcaby), data dostępu 01.06.2020
2. Draughts (Warcaby), Wikipedia  
[en.wikipedia.org/wiki/Draughts](https://en.wikipedia.org/wiki/Draughts), data dostępu 01.06.2020
3. Minimax (Algorytm MinMax), Wikipedia  
[en.wikipedia.org/wiki/Minimax](https://en.wikipedia.org/wiki/Minimax), data dostępu 01.06.2020