

Wizja programu Warcaby

Wersja 3

2017.03.13

Autorzy:

- Mateusz Harciarek
- Marcin Oszczyk
- Paweł Szczygieł

Krótki opis programu:

Program ma za zadanie umożliwić grę w [warcaby angielskie](#) między dwoma graczami (na jednym komputerze) lub przeciw sztucznej inteligencji.

Rozgrywka:

Po włączeniu programu, pojawi się ekran rozgrywki z automatycznie załadowanym stanem ostatniej gry (jeśli nie została zakończona) lub ekran wyboru gry w przeciwnym przypadku. Jeżeli rozszerzymy program o tworzenie profili, po włączeniu pojawi się ekran umożliwiający wybór profilu, tworzenie nowego lub grę jako gość.

W ekranie rozgrywki znajdować się będą:

- Plansza, przedstawiająca aktualny stan gry
- Informacja o tym, czy dany gracz jest sterowany przez człowieka, czy przez komputer
- Pozostały graczom czas (jeśli wybrana opcja gry z zegarem)
- Informacja, do kogo należy aktualny ruch.

Informacje te postaramy się w miarę możliwości przekazać bez tekstu (np. aktualnego gracza przez podświetlenie jego kamieni). W każdym momencie będzie możliwe otworenie menu.

W ekranie menu znajdować się będą następujące opcje:

- Kontynuuj (powrót do ekranu rozgrywki)
- Nowa gra (przechodzi do ekranu wyboru)
- Zapisz grę (umożliwia wybranie nazwy dla zapisanej rozgrywki)
- Wczytaj grę (przechodzi do wyboru zapisanych gier, a następnie ekranu rozgrywki tej gry)
- Rozgrywka (cofnij, ponów, odpowiedz, zamień strony)
- Wygląd (zmiana wyglądu planszy i kamieni oraz możliwość przekręcenia planszy)
- Profil (zmiana nazwy oraz portretu, statystyki oraz możliwość wylogowania)
- Wyjdź (wyłącza program – aktualna rozgrywka zostaje automatycznie zapisana)

Być może w trakcie projektowania zdecydujemy się część z opcji (np. przycisk cofnięcia i ponowienia ruchu) umieścić także w ekranie rozgrywki.

Ekran wyboru zawierać będzie następujące elementy:

- Wybór typu warcabów (na pewno angielskie – być może klasyczne oraz polskie)
- Wybór przeciwnika (człowiek lub komputer)
- Wybór koloru kamieni gracza (w przypadku gry przeciw komputerowi)
- Wybór poziomu trudności (w przypadku gry przeciw komputerowi)
- Zegar (ustawienie czasu lub gry bez zegara)

Program będzie obsługiwany za pomocą myszki. Wszystkie opcje będą dostępne przez kliknięcie na odpowiedni przycisk. Ruchy kamieniami będą wykonywane poprzez przesunięcie kamienia (wciśnięcie LPM po najechaniu kursorem na kamień, przesunięcie kursora na wybrane przez nas pole i puszczenie LPM). Nie wykluczamy także utworzenia skrótów klawiszowych dla części akcji.

Minimum:

Minimalna część projektu, jaką chcemy wykonać obejmuje stworzenie programu wyświetlającego stan gry i umożliwiającego przeprowadzenie rozgrywki z komputerem. Plan ten możemy rozbić odpowiednio na kilka części:

- Stworzenie minimalnego interfejsu użytkownika (na początek może to być interfejs konsolowy)
- Napisanie zasad gry aby zarówno komputer jak i gracz mogli wykonywać jedynie poprawne ruchy, oraz aby gra kończyła się w odpowiedni sposób w przypadku wygranej/remisu
- Napisanie podstawowych wersji algorytmów sztucznej inteligencji umożliwiających grę z komputerem
- Stworzenie graficznego interfejsu użytkownika
- Stworzenie ekranu startowego gry (ekran menu) gdzie będą dostępne niezbędne opcje

Krótki opis sztucznej inteligencji:

Sztuczna inteligencja programu będzie miała za zadanie rozgrywanie partii warcabów przeciwko użytkownikowi. Musi ona umieć rozpoznać wszystkie legalne ruchy i wybrać najlepszy (lub właściwy dla ustawionego poziomu trudności). Zostaną do tego wykorzystane:

- Algorytm minimax - oceniający, który z ruchów przy danym ustawieniu jest najbardziej korzystny. Tworzone jest drzewo symulujące d ruchów, a następnie przy użyciu funkcji wartościującej stany gry (ilość kamieni, ich pozycja, ilość królów itp.) jest odnajdywany najlepszy z możliwych do osiągnięcia stanów.
- Algorytm alfa-beta - usprawniający przeszukiwanie drzewa ruchów z algorytmu minimax poprzez znalezienie niektórych ruchów, o których z góry wiadomo że nie będą przynosiły korzyści graczowi. Algorytm utrzymuje dwie wartości alfa i beta, które reprezentują minimalny wynik gracza MAX i maksymalny wynik gracza MIN (sztuczna inteligencja i jej przeciwnik). Początkowo alfa jest -nieskończonością, a beta +nieskończonością. W miarę postępowania rekursji przedział (alfa; beta) staje się mniejszy i kiedy beta staje się mniejsze niż alfa, oznacza

to, że obecna pozycja nie jest optymalna i nie ma potrzeby przeszukiwania głębiej.

- Dostępne w internecie bazy danych przedstawiające korzystny przebieg gry przy danym ustawieniu figur na planszy (endgame tablebase) . Z uwagi na ilość możliwych sytuacji, bazy zawierają jedynie sytuacje, w których graczom pozostały nie więcej niż cztery kamienie.

Zastosowania programu:

- Program będzie udostępniał rozgrywkę gracz vs gracz, więc można go użyć do przyjacielskich rozgrywek oraz turniejów
- Tryb rozgrywki gracz vs sztuczna inteligencja może służyć do rozwijania umiejętności użytkownika
- Dzięki możliwości dostosowania poziomu trudności rozgrywki program będzie ciekawą propozycją zarówno dla początkujących jak i dla doświadczonych graczy
- Program będzie również posiadał możliwość podglądu statystyk gracza, co umożliwi użytkownikowi obserwowaniu jego postępu w doskonaleniu umiejętności gry
- Dzięki możliwości wyboru innych odmian warcabów użytkownik nie znudzi się szybko programem
- Zapis gry będzie wykonywany standardowej notacji (uniwersalny zapis partii warcabów), dzięki czemu będzie możliwe odtworzenie partii niezależnie od reszty programu

Prawa:

Program będzie dostępny w modelu open-source jako repozytorium git. Ze względu na fakt, że program jest wykonywany w ramach projektu na uczelni wyższej, prawa do niego ma również dana uczelnia.

Wydajność:

Aby zwiększyć wydajność programu (w tym przypadku – czas oczekiwania na ruch komputera) można już w trakcie tury “gracza człowieka” szukać najbardziej prawdopodobnych ruchów gracza, a następnie wybierać do nich swoje kolejne. Sprawdzi się to jedynie w przypadku wyboru wysokiego poziomu trudności, gdy program potrzebuje dużo czasu na obliczenia.

Dokumentacja:

- Dokumentacja techniczna zostanie zawarta w kodzie zgodnie ze standardami i wygenerowana automatycznie za pomocą generatora Doxygen
- Dokumentacja użytkownika zostanie umieszczona jako oddzielny plik tekstowy dołączony do programu.

Proces tworzenia:

Projekt zostanie wykonany przez trzyosobowy zespół, którego członkowie będą komunikować się bezpośrednio oraz poprzez komunikatory internetowe. Postępy w wykonaniu programu będą umieszczane przez członków zespołu w repozytorium git. Tam też będą umieszczane kolejne wersje działającego programu.