Szachy

by Miłosz Rzyczniak

## O projekcie

Jest to prosta implementacja gry w szachy w dla dwóch graczy w C#.

## Technologie

* C#: Główny język używany do implementacji logiki gry.
* .NET 7.0 Framework: Zapewnia biblioteki i narzędzia do budowy i uruchamiania aplikacji.

## Uruchomienie

Uruchom plik wykonawczy:

ChessProject/bin/Debug/net7.0/chess.exe

## Funkcjonalność

* Zaimplementowano podstawowe zasady szachowe:
  + Ruszanie figur
  + Zbijanie
  + Roszada (Castling)
  + Szachowanie (Check)
  + Matowanie (Checkmate)
  + Promocja
* Każdy ruch podlega walidacji poprawności
* Gra wykrywa, kiedy król gracza jest szachowany i uniemożliwia ruchy, które zostawiłyby króla w szachu.
* Gra śledzi i wyświetla bieżącą turę, gracza oraz stan planszy w konsoli.
* Po zakończeniu gry ogłaszany jest zwycięzca i wyświetlana jest historia ruchów.

## Struktura projektu

Projekt korzysta z typowej struktury biblioteki klas C#, zorganizowanej w przestrzenie nazw i klasy:

### Przestrzenie nazw

* chess:
  + Zawiera główne klasy związane z grą w szachy.
* chess.Components:
  + Zawiera klasy reprezentujące różne komponenty gry w szachy, takie jak figury, plansza, gracze i tury.
* chess.Helpers:
  + Zawiera klasy pomocnicze i enumeracje używane w całym projekcie.
* chess.Helpers.Enums:
  + Zawiera enumeracje używane do reprezentowania kolorów i typów figur szachowych.

### Klasy i enumeracje

* Game:
  + Reprezentuje główną logikę gry, w tym inicjalizację, przepływ gry i wykonywanie ruchów.
* Coordinates:
  + Reprezentuje współrzędne na planszy do gry.
* GameBoard:
  + Reprezentuje planszę do gry i zapewnia metody dodawania, usuwania i dostępu do figur na planszy.
* Move:
  + Reprezentuje ruch wykonany przez gracza, w tym pozycje początkowe i końcowe, ruchomą figurę, każdą zbieraną figurę i promocję.
* Player:
  + Reprezentuje gracza w grze, w tym jego kolor i wynik.
* Turn:
  + Reprezentuje pojedynczą turę w grze, w tym gracza wykonującego ruch i sam ruch.
* Piece:
  + Reprezentuje figurę szachową. Jest to klasa abstrakcyjna z klasami potomnymi dla każdego typu figury (Król, Królowa, Wieża, Goniec, Skoczek, Pionek).
* ColorType:
  + Enumeracja reprezentująca kolor figur (Czarny lub Biały).
* PieceType:
  + Enumeracja reprezentująca typy figur szachowych (Król, Królowa, Wieża, Goniec, Skoczek, Pionek).
* ConsoleHelper:
  + Zapewnia metody pomocnicze do wyświetlania komunikatów na konsoli w różnych kolorach.
* ErrorMessages:
  + Zawiera statyczne komunikaty o błędach używane w całej grze.
* InfoMessages:
  + Zawiera statyczne komunikaty informacyjne używane w całej grze.
* WarningMessages:
  + Zawiera statyczne komunikaty ostrzegawcze używane w całej grze.
* PieceSymbols:
  + Zawiera słownik mapujący typy figur na odpowiadające im symbole Unicode.

## Biblioteki zewnętrzne

Projekt nie opiera się na żadnych zewnętrznych bibliotekach ani zależnościach.