

WARCABY

1. ANALIZA OBIEKTOWA

Program który zaimplementowałem jest klasyczną grą w warcaby dla dwóch użytkowników którzy grają przeciwko sobie.

Zasady gry: Normalnym pionkiem możemy poruszać się tylko do przodu oraz bić również tylko do przodu, zbić jednorazowo możemy też tylko jednego pionka

Gdy Jesteśmy „matką” możemy poruszać się oraz bić w przód i w tył ale również tylko o jedno pole oraz bić również maksymalnie jednego pionka

Gra kończy się wtedy, gdy któryś z graczy nie może wykonać żadnego ruchu swoim pionkiem (został zablokowany) lub skończą mu się pionki, w tych przypadkach oczywiście gra się kończy wygraną gracza przeciwnego

Na samym początku rozgrywki wypisana jest legenda poleceń oraz przykład jak zagrać.

OPIS KLAS :

- Board - Klasa, która przechowuje rozmiar szachownicy, ilość pionków dla każdego gracza oraz wypełnia wszystkie pola tak zwanym „pustym pionkiem” a następnie rozstawia pionki pojedynczych graczy w odpowiednie miejsca.
Zawiera klasę Pawn
Zawiera jedną metodę jest to:
 - `_Str_()` -Metoda specjalna, która definiuje sposób w jaki obiekt ma być wypisywany, w naszym przypadku naszą planszę
- Pawn - Klasa, która przechowuje informacje na temat pojedynczego pionka (kolor pionka, współrzędne x oraz y)
- Engine – Silnik Gry
Zawiera klasę Board
Zawiera następujące metody:
 - `game_loop()` – główna pętla gry, która wypisuje składnię gry na początku oraz tutaj wykonywany jest ruch pionka w danej turze, metoda również co każde przejście pętli sprawdza czy gracz może wykonać ruch swoimi pionkami, jeżeli posiada pionki a nie może wykonać żadnego ruchu to oczywiście wygra gracz przeciwny.
 - `move()` - tutaj mamy metodę, która obsługuje ruch pionka, mamy tutaj również wywołanie metod statycznych, które walidują czy gracz nie chce wykonać ruchu niezgodnego z zasadami gry w warcaby

- `can_move_color()` - sprawdza, czy jakiś pionek danego koloru może się ruszyć. Jeśli tak to nie kończymy gry, zwracając True. W przeciwnym razie - zwracamy False
 - `can_move()` - metoda bierze pojedynczy pionek i sprawdza, czy może się ruszyć
- METODY STATYCZNE(Metody klasy):**
- `valid_pawn()` – sprawdza, czy pionek jest prawidłowy tzn. czy nie jest „pusty”
 - `valid_down()` – sprawdza, czy pionek chce się ruszyć w dół i jeśli tak to czy może (jest matką)
 - `valid_move()` - sprawdza, czy nie ruszamy się poza planszę i czy pionek, który chce się ruszyć ma teraz swój ruch
 - `encode()` - metoda zmieniająca liczbę na literę
 - `decode()` - metoda zmieniająca literę na liczbę

Klasy z wyjątkami , które dziedziczą po wbudowanej w Python klasie Exception

- TurnException
- OutOfBoardException
- WrongMoveException
- NullPawnException

WYKONAŁ : Maciej Tarkiewicz