

# Refaktoryzacja gry Kółko i Krzyżyk

```
Player 1: Podaj swój symbol
X
Player 2: Podaj swój symbol
Y
Wybierz rodzaj gry:
1) 1 vs computer
2) 1 vs 1
1
Wpisz rozmiar planszy do gry
4
||==||==||==||==||
|| - || - || - || - ||
|| - || - || - || - ||
|| - || - || - || - ||
|| - || - || - || - ||
||==||==||==||==||
Podaj wiersz:
█
```

## Autorzy

Matuszak Rafał, Piecuch Mikołaj, Tracki Mateusz

## Opis projektu

Celem projektu było zrefaktoryzowanie oraz otworzenie kodu, który został umieszczony na [githubie](#), na przyszłe zmiany.

## Proponowane zmiany

1. Możliwość przeprowadzenia różnych trybów gry (gra 1 vs 1 lub gra 1 vs komputer)
2. Zmiana obiektu przechowującego planszę na tablicę dwuwymiarową
3. Możliwość wybrania rozmiaru planszy od 3x3 do NxN
4. Zmiana sprawdzania warunków zwycięstwa
5. Określenie gracza który wygrał grę

## Opis zmian

### Początki zmian

Istniejący kod zawierał kilka "zapachów" w kodzie z którymi na samym początku musieliśmy się uporać:

1. Attribute smell - klasa `Game` zawierała pole `attr_accessor :board`, które umożliwia jego edycję. Aby to naprawić, wystarczyło zmienić typ pola na `attr_reader`, czyli pole tylko do odczytu.
2. Nill Check - w klasie `Game` znajdowało się porównanie obiektu do `nil`, które narusza zasadę "Tell, don't ask".

Po usunięciu "zapachów" przeszliśmy do następnego etapu prac nad projektem.

## Zmiana definicji Gracza

### Change player definition

Zmiana ta polegała na zamienieniu Gracza, który dotychczas był reprezentowany przez trywialny obiekt (string przechowujący symbol), na obiekt z polem `symbol`. Umożliwiło nam to otwarcie kodu na przyszłą zmianę dotyczącą różnych trybów rozgrywki.

## Rozdzielenie logiki aplikacji

### Add gameboard class and move game mechanism to separate file

W tej zmianie została utworzona klasa `Board` w której przechowywana jest logika gry. Dotychczas plansza była reprezentowana przez string zawierający ciąg symboli graczy lub pustych pól.

## Dodanie implementacji gracza komputerowego oraz człowieka. Interaktywne przeprowadzenie gry

### Add human and computer classess. Make project playable

W tym kroku zostały utworzone konkretne implementacje graczy dziedziczące po klasie `Player`. Obie posiadają metodę `move`, różni je jednak to, że w klasie `ComputerPlayer` wybierany jest pierwszy z brzegu możliwy ruch. Natomiast w klasie `HumanPlayer` pytamy użytkownika o współrzędne na planszy, w które ma zostać wprowadzony symbol.

Ponadto zostały dodane metody wyświetlające planszę do gry w terminalu. Po zastosowaniu tych zmian, możliwe stało się przeprowadzenie rozgrywki.

## Feature Envy smell oraz dodanie losowego ruchu komputera.

Fix Feature Envy smell. Add random method for computer move Po dodaniu powyższych zmian, otrzymaliśmy Feature Envy Smell. "Zapach" ten powstaje, kiedy fragment kodu odwołuje się do innego obiektu, częściej niż do samego siebie.

Aby pozbyć się tego "zapachu", przebudowana została metoda `move` w klasie `HumanPlayer`.

Dodatkowo zmieniona została logika poruszania się gracza komputerowego. Od teraz porusza się on po losowych polach.

## Wybór trybu gry

### Add selecting game mod

Dodana została możliwość wyboru trybu gry. Można wybrać rozgrywkę z komputerem lub drugim graczem.

## Refaktoring podawania startowych parametrów

### Change passing start parameters method

Dodano metodę `start` w klasie `Game`, która odpowiada za pobranie parametrów niezbędnych do rozpoczęcia gry oraz jej rozpoczęcie.

Za pobranie parametrów odpowiadają 3 metody (`choose_symbol`, `choose_mode`, `choose_board_size`). Dodatkowo walidują one wprowadzone argumenty.

## Warunki zwycięstwa

### Add winner conditions

Zostały wprowadzone 4 warunki zwycięstwa:

- sprawdzanie wierszy
- sprawdzanie kolumn
- sprawdzanie 2 różnych przekątnychSprawdzenie polega na znalezieniu tylu takich samych symboli w dowolnym warunku, jak duży jest wymiar planszy. Gracz wygrywa, gdy któryś warunek jest spełniony.

## Wyświetlenie zwycięzcy

### Add message with winner

Ostatnią zmianą było wyświetlenie gracza który wygrał grę lub ogłoszenie remisu.