

# SYMULATOR WALK POKEMON

## 1.Cel i opis projektu :

Celem projektu było stworzenie tekstowej gry symulującej pojedynki na pokemony między dwoma graczami, niczym w grach Pokemon dostępnych na konsole Nintendo.

Gra polega na symulacji pojedynku na pokemony między dwoma graczami albo graczem i komputerem.

Na samym początku tworzy się drużynę liczącą od 1 do 6 pokemonów a następnie odbywa się pojedynek w systemie turowym, który trwa do momentu, aż któryś z graczy nie straci wszystkich pokemonów (ich życie spadnie do 0).

## 2.Podział na klasy :

- **Klasa Pokemon** : jest to klasa która reprezentuje pojedyncze pokemony w drużynie gracza. Zawiera w sobie wszystkie najważniejsze dane o pokemonie takie jak nazwa, statystyki (atak, obrona, życie, szybkość), jego typy oraz mnożniki odporności na ataki różnych typów.
- **Klasa Gracza (Player)** : jest to klasa która reprezentuje gracza w grze. Przechowuje informacje o nazwie gracza oraz o jego drużynie, czyli pokemonach które wybrał do walki
- **Klasa Komputer (Computer)**: jest to klasa dziedzicząca po klasie **Gracza** i również przechowująca informacje o drużynie, tym razem komputera. Rozbudowana jest jednak o metody które umożliwiają komputerowi wybieranie pokemonów do drużyny, wybieranie pokemonów do walki oraz metody odpowiedzialne za wykonywanie przez komputer logicznych ruchów
- **Klasa Gry (Game)**: jest to klasa która przechowuje w sobie wydarzenia w grze. Jest odpowiedzialna za ustalanie który pokemon zaatakuje szybciej, sprawdzanie wyników poszczególnych rund oraz przeprowadzanie całej walki, aż nie zostanie wyłowiony zwycięzca
- **Klasa Ruchów (Moves)**: jest to dziedzicząca po klasie **Enum** i odpowiedzialna jest za przypisywanie odpowiednich ruchów do wyborów gracza (tego co wpisze na klawiaturze)
- **Klasa Wyników (Outputs)**: jest to również klasa dziedzicząca po klasie **Enum** i odpowiedzialna jest za przypisanie odpowiedniego rezultatu rundy
- **Plik interface** : nie jest to klasa sama w sobie, lecz bardzo ważny plik odpowiedzialny za funkcje wymagające interakcji ze strony gracza. Znajdują się w nim wszystkie funkcje odpowiedzialne za wybranie przez gracza ruchu, pokemona oraz funkcje które odpowiedzialne są za wyświetlanie informacji o przebiegu rundy i całej walki w terminalu

### **3. Instrukcja użytkownika :**

Aby rozpocząć grę trzeba pobrać wszystkie pliki znajdujące się w repozytorium (łącznie z plikiem pokemon.csv, który zawiera informacje o każdym pokemonie). Następnie odpalić terminal i przejść w nim do folderu z tymi plikami.

Komenda **python main.py** powinna uruchomić program

Instrukcja gry:

- 1) Na samym początku trzeba wskazać tryb gry, który chcesz włączyć poprzez wpisanie na klawiaturze odpowiedniej cyfry (1 – gracz kontra gracz, 2 – gracz kontra komputer)
- 2) Następnie trzeba wybrać ilość pokemonów która znajdzie się w każdej drużynie. Należy wpisać na klawiaturze liczbę między 1 a 6
- 3) Kolejnym krokiem jest stworzenie graczy biorących udział w walce. Należy podać imię gracza, a następnie przechodzi się do wyboru pokemonów. Na początku gra prosi o wybranie jednej z liter alfabetu, a następnie wskazując listę dostępnych pokemonów na daną literę. W zależności od trybu gry sekwencję powtarza się raz albo dwa razy
- 4) Następnie gra prosi każdego z graczy o wybranie swojego startowego pokemona
- 5) Po wyborze startowych pokemonów przechodzimy do właściwej rozgrywki.  
Każdy z graczy wybiera jeden z dostępnych 4 ruchów:
  1. ATTACK – atak ruchem typu normalnego
  2. SPECIAL ATTACK – gra prosi gracza o wybranie jednego z typów pokemona i wykonuje atak wybranym przez nas typem. Jeśli pokemon nie posiada drugiego typu automatycznie wykonywany jest atak typem pierwszym
  3. DEFEND – powoduje wzrost defensywy pokemona o 10%
  4. SWITCH – pozwala graczowi na wybór pokemona który zastąpi jego obecnego o ile posiada jeszcze inne zdolne do walki pokemony w drużynie
- 6) Następnie odbywa się runda i jej rezultat wyświetlany jest na ekranie. Cała sekwencja począwszy od wyboru ruchu pokemona jest powtarzana do momentu aż jeden z graczy nie straci wszystkich pokemonów

### **4. Część refleksyjna :**

Pisanie projektu zajmowało strasznie dużo czasu i napotykałem po drodze wiele problemów. Jednak ostatecznie udało się stworzyć tekstowy symulator walk pokemon, który mimo że nie jest idealny to pozwala chociaż w lekkim stopniu poczuć się jak w oryginalnych grach pokemon. Cały zamysł jaki miałem na stworzenie tej gry udało się zaimplementować. Jedyne co uległo zmianie to wyświetlanie listy dostępnych pokemonów. Oryginalnie miały wyświetlać się wszystkie, ale w ostatecznej wersji wyświetlają się tylko pokemony na wprowadzoną przez gracza literę.

### Czego nie udało się osiągnąć ?

- Pisanie zajęło mi więcej czasu niż sądziłem i nie zdążyłem się wyrobić ze stworzeniem GUI. Wolałem skupić się najpierw aby gra działała jak najlepiej w terminalu i po prostu zabrakło czasu na wykonanie graficznego interfejsu
- Stworzony przeze mnie komputer potrafi wykonywać logiczne ruchy, ale nie są one na najwyższym poziomie. Do ulepszenia zostało wybieranie pokemona na którego ma się zmienić, ponieważ wybiera on losowo, więc może się zdarzyć że pokemon którego wybierze również będzie podatny na ataki przeciwnika. Podobnie jak z GUI zabrakło czasu aby usprawnić logikę komputera. Stworzenie go było dużym wyzwaniem i wielokrotnie jego logika musiała ulec zmianie
- Skrócenia metody `game.play()`. Została ona podzielona na dwie funkcje, z których jedna odpowiada za tryb gracz kontra gracz, a druga za tryb gracz kontra komputer. Mimo że zawierają wiele wspólnych części kodu to różnica pojawia się w zmienianiu pokemonów. Nie byłem w stanie znaleźć rozwiązania które połączyłoby różnice między komputerem a graczem, co umożliwiłoby skrócenie kodu.

Uważam że mój projekt jest dobry ponieważ pozwala na zabawę nie tylko z drugą osobą, ale także dzięki komputerowi umożliwia możliwość gry samemu. Starałem się żeby odczucia gracza były jak najlepsze podczas gry dlatego zwracałem uwagę na najmniejsze detale i starałem się odwzorować grę jak najlepiej potrafiłem. Dobrym przykładem jest wyświetlanie się tekstu, który tak jak w oryginalnych grach wyświetla się literka po literce.