



### **Opis projektu (główne założenia, jaki jest cel i istota aplikacji)**

Projekt jest prostą grą znaną powszechnie jako "Snake". Gra polega na kierowaniu wężem po dwuwymiarowej planszy i zbieranie jedzenia poprzez najechanie na nie powierzchnią węża. Po rozpoczęciu gry, wąż składa się z jednego segmentu, a po zdobyciu każdej kolejnej jednostki jedzenia rozrasta się o jeden segment. Gra nie ma celu zapewniającego zwycięstwo i ma swój koniec tylko w wypadku gdy gracz przegra. Rozgrywka możliwa jest w trzech trybach: *łatwy (easy)*, w którym prędkość poruszania się węża jest najwolniejsza, a przegrana możliwa jest tylko w wypadku gdy wąż uderzy on o samego siebie, *średni (medium)* gdzie prędkość węża jest nieco szybsza niż w trybie łatwym, a dodatkowym utrudnieniem są ściany dookoła planszy, których dotknięcie powoduje przegraną, oraz *trudny (hard)* gdzie prędkość węża jest największa, a zamiast ścian dookoła mapy, przeszkody ulokowane są w jej losowych punktach. Interfejs gry jest tekstowy, co było głównym założeniem projektu.

### **- Opis funkcjonalności (wymienione główne funkcjonalności oraz ich omówienie),**

- Program umożliwia granie w snake-a na trzech poziomach trudności, możliwych do wyboru po kliknięciu "play" w menu głównym, które wyświetli się użytkownikowi od razu po uruchomieniu programu.
- W trakcie rozgrywki gracz ma wgląd w to ile jedzenia uzbierał (score).
- Podczas rozgrywki możliwa jest pauza.
- Po przegranej użytkownik może wyjść do głównego menu klikając ESC lub odczekać 3 sekundy, po których rozgrywka uruchomi się od nowa.
- W programie dostępne są ustawienia, w których użytkownik ma możliwość zmiany koloru węża, przeszkód i jedzenia (do wyboru ma biały, czerwony, różowy, zielony i żółty) oraz tła (do wyboru ma czarny i fioletowy)

### **- Szczególnie interesujące zagadnienia projektowe**

Ciekawym zagadnieniem jest zapisywanie danego fragmentu rozgrywki przy pauzowaniu, W obecnym sposobie działania aplikacji każde wejście do metody odpowiadającej za rozgrywkę powoduje utworzenie nowej rozgrywki, więc "pauza" jest tak naprawdę skopiowaniem poprzedniej gry

Pauzowanie:

```
elif event.key == pygame.K_p:
    self.saved_snake = self.snake.saved()
    self.saved_food = food.saved()
    self.saved_walls = wall.positions
    self.pause()
```

Od Pauzowanie:

```
if paused:
    self.snake.restore(self.saved_snake)
    food.restore(self.saved_food)
    wall.restore(self.saved_walls)
```

#### **- Instrukcja instalacji**

Pobranie paczki z Cez'a

#### **- Instrukcja konfiguracji**

Zainstalowanie IDE obsługującego język Python i włączenie projektu po uprzednim ustawieniu konfiguracji na main.py

#### **- Instrukcja użytkownika**

Użytkownik nawiguje po aplikacji za pomocą strzałek, przycisku ENTER (do wybierania zaznaczonej opcji), ESC (wyjście, cofnij), oraz (żeby zapauzować grę) przycisku P

#### **- Wnioski**

Stworzenie aplikacji z interfejsem tekstowym nie jest skomplikowanym zadaniem, natomiast napisanie takiej aplikacji w sposób umożliwiający łatwe podmienienie warstwy prezentacji wymaga uprzedniego zaplanowania struktury projektu.

#### **- Samoocena.**

Aplikacja nie jest skomplikowana, natomiast z uwagi na możliwości wyboru poziomów, pauzowaniu, zmienianiu kolorów różnych elementów rozgrywki, oraz struktury, która umożliwia bardzo łatwą zmianę na interfejs graficzny, uważamy, że zasługujemy na ocenę 3.5/4.