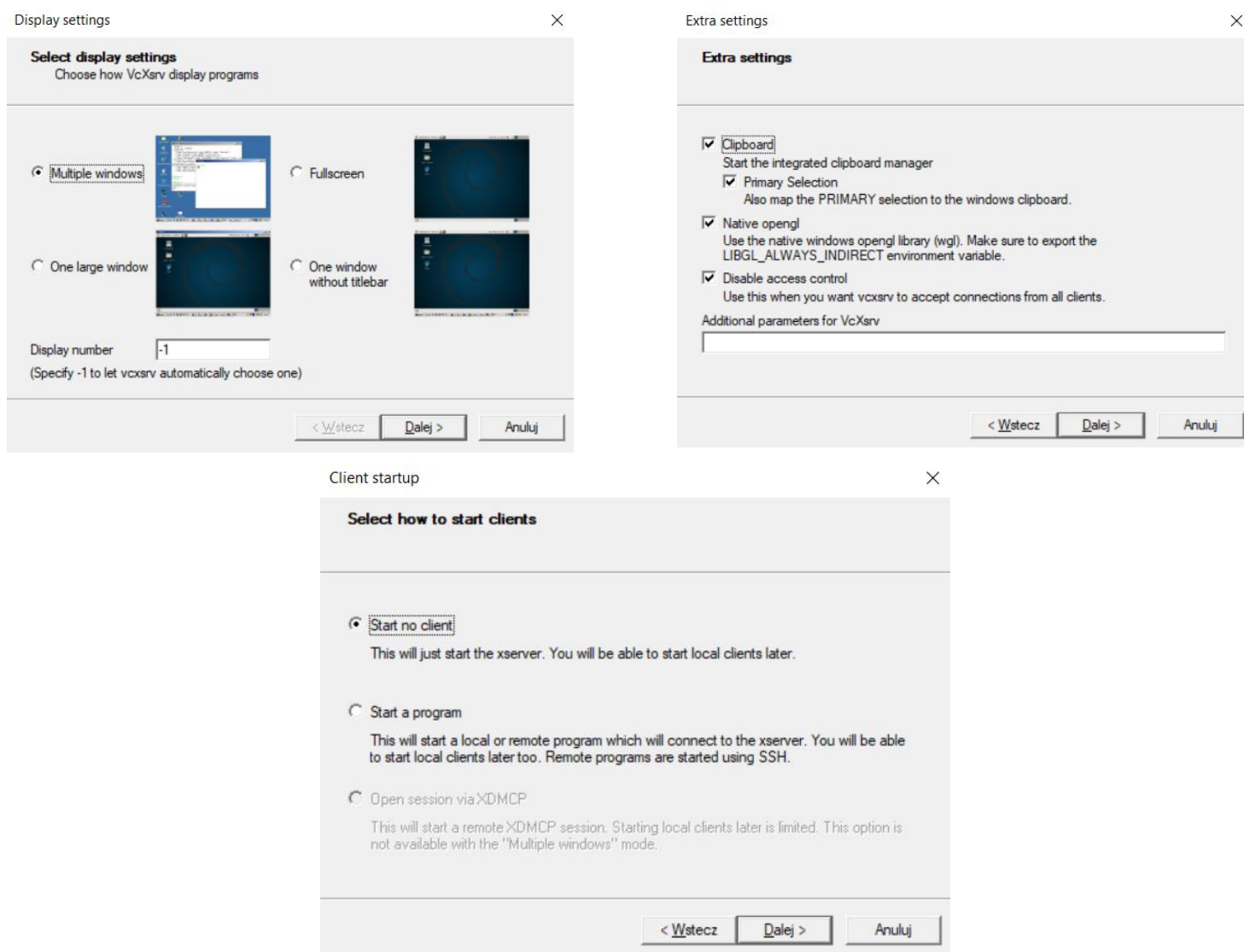


## 1. Przebieg eksploatacji programu

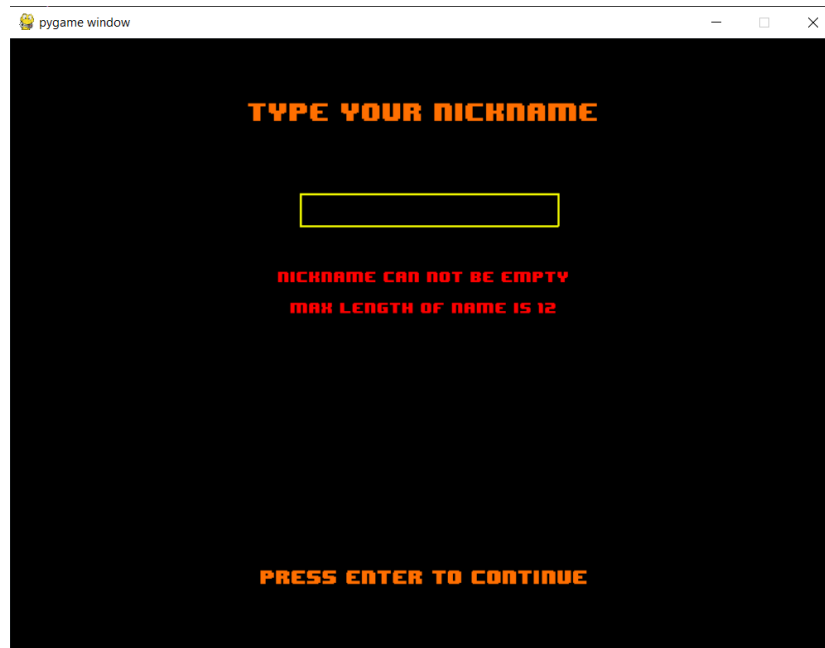
Żeby uruchomić program należy uruchomić plik *main.py*. Ponieważ program korzysta z modułu *pygame*, to żeby program działał, to potrzebuje mieć dostęp do ekranu. Żeby to zrobić można posłużyć się programem *XLaunch*. Przy uruchomieniu trzeba zaznaczyć pola, jak na rysunku 1.



Rysunek 1. „Konfiguracja XLaunch”

## 1.1. Okno wpisywania imienia użytkownika

Po uruchomieniu *main.py* pojawi się okno, które pokazano na rysunku 2. Żeby przejść dalej należy wpisać imię, które potem będzie wykorzystane do wpisywania wyników gry do tablicy najwyższych wyników.



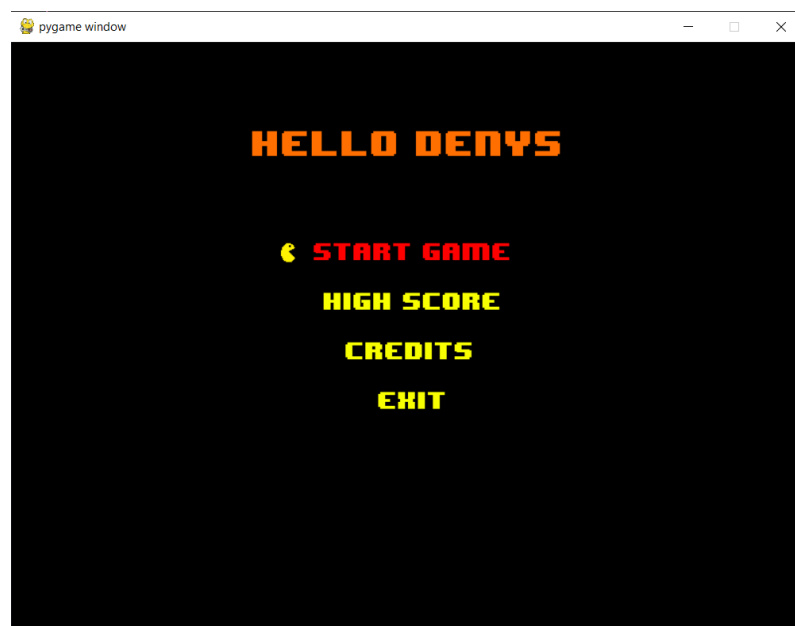
Rysunek 2. „Okno wpisywania imienia użytkownika”

Jeżeli użytkownik spróbuje wcisnąć klawisz ENTER bez wprowadzenia żadnych symboli, to nic się nie stanie. Jeżeli użytkownik spróbuje wprowadzić więcej niż 12 symboli, to następujące symbole po 12. nie zostaną zapisane. Żeby usunąć ostatni symbol należy wcisnąć klawisz BACKSPACE. Po wprowadzeniu imienia i wciśnięciu ENTER pojawi się okno głównego menu, i imię użytkownika zapisze się do programu. Imię musi składać się wyłącznie z liter z języka angielskiego. Symbole z innych języków będą wyświetlone jako []. To dlatego, że czcionka, która została wybrana przeze mnie w tym programie, wyświetla poprawnie tylko symbole z języka angielskiego. Link do tej czcionki:

<https://www.dafont.com/8bit-wonder.font>

## 1.2. Okno głównego menu

Okno głównego menu wygląda jak na rysunku 3. Tekst znajdujący się najwyżej w tym oknie ma postać: „Hello {imię użytkownika}”, gdzie imię użytkownika – tekst, który został wpisany w poprzednim oknie. Za pomocą klawisz strzałki w górę, strzałki w dół można poruszać się po tym menu. Ikonka Pac-man’a wskazuje na obecną pozycję w menu. Przy wciśnięciu klawiszy ENTER można przechodzić do następnych okien. Podobna logika działa dla innych okien w tej aplikacji.



Rysunek 3. Okno głównego menu

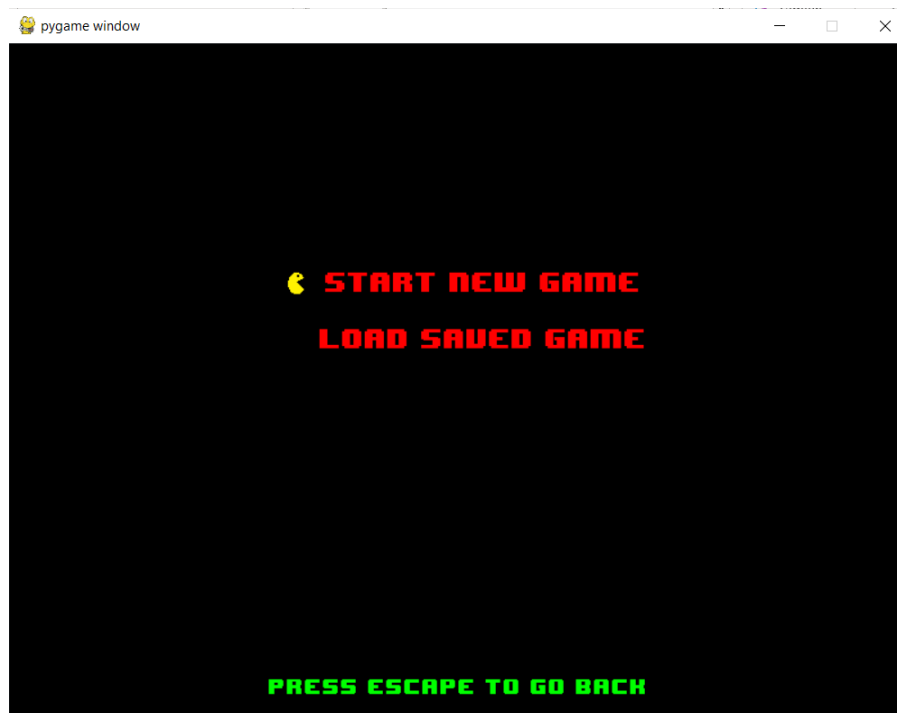
- Jeśli ikonka Pac-man’a wskazuje na „Start game” i użytkownik wciśnie ENTER, to włączy się okno wyboru gry.

- Jeśli ikonka Pac-man'a wskazuje na pozycję „High score” i użytkownik wciśnie ENTER, to włączy się okno Z najlepszymi wynikami gry.
- Jeśli ikonka Pac-man'a wskazuje na „Credits” i użytkownik wciśnie ENTER, to włączy się okno z tekstem, zostawionym przeze mnie.
- Jeśli ikonka Pac-man'a wskazuje na pozycję „Exit” i użytkownik wciśnie ENTER, to wyłączy się gra. Żeby wyłączyć grę można wcisnąć na „X” w prawym górnym rogu programu.

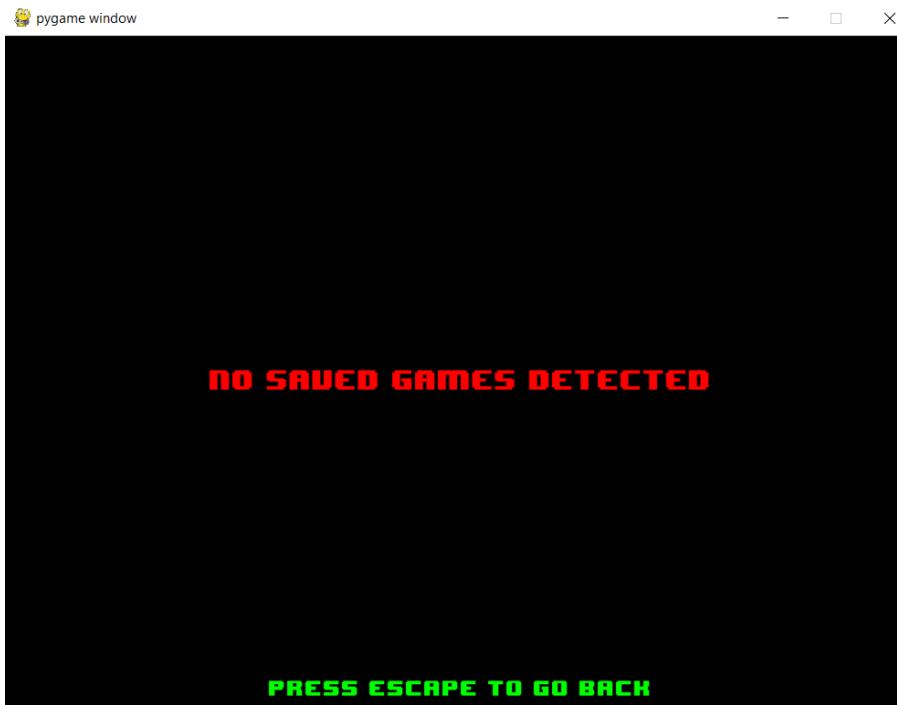
### **1.3. Okno wyboru gry**

Okno wyboru gry pokazano na rysunku 4.

- Jeśli wciśnięto klawisz ESC, to odbędzie się przejście do głównego menu.
- Jeśli ikonka Pac-man'a wskazuje na „Load saved game” i wciśnięto ENTER, to jeśli plik „saved\_game.json” jest pusty, to wyświetli się okno wyświetlające informację o tym, że nie ma zapisanej gry. Takie okno pokazano na rysunku 5. Jeśli ten plik nie jest pusty, to wczyta się zapisana gra.
- Jeśli ikonka Pac-man'a wskazuje na „Start new game” i wciśnięto ENTER, to rozpocznie się gra. Plik z informacją o grze jest „lvl1.json”.



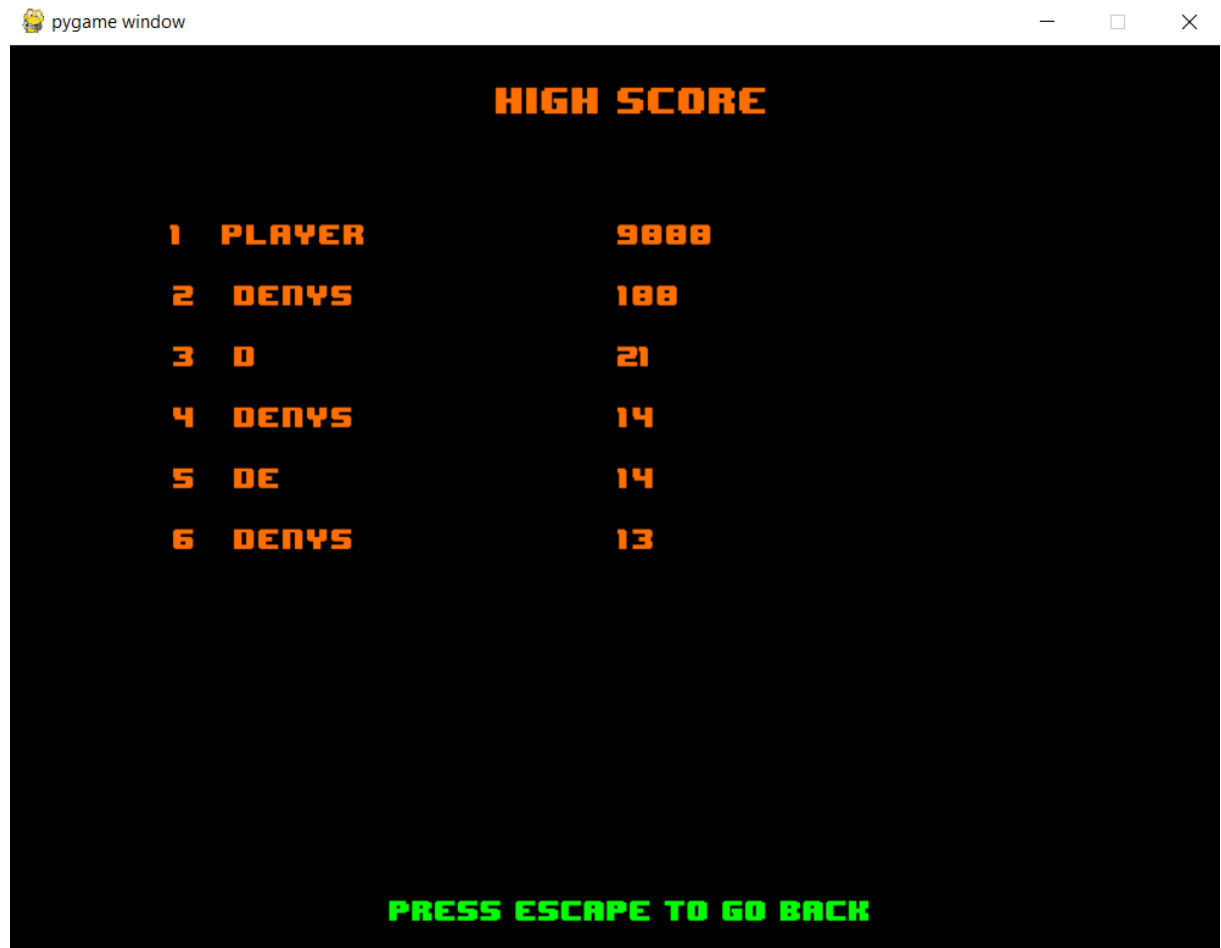
Rysunek 4. „Okno wyboru gry”



Rysunek 5. „Okno pokazujące informację, że nie ma zapisanej gry”

#### 1.4. Okno z najlepszymi wynikami gry

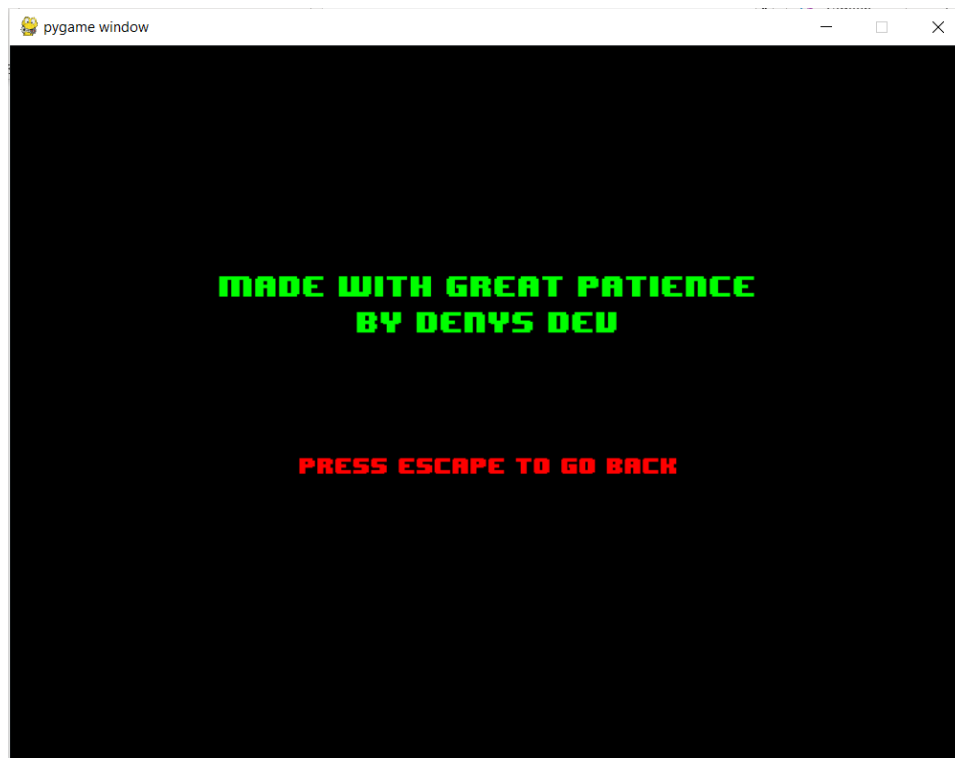
Okno z najlepszymi wynikami gry pokazano na rysunku 6. Te wyniki zapisane w pliku „score.csv”. Maksymalna liczba tych wyników wynosi 10. Program kontroluje, żeby liczba tych wyników nie przekraczała tę liczbę.



Rysunek 6. Okno z najlepszymi wynikami gry

#### 1.5. Okno „Credits”

Okno „Credits” ma postać jak na rysunku 7. Żeby przejść do poprzedniego okna należy wcisnąć klawisz ESC.

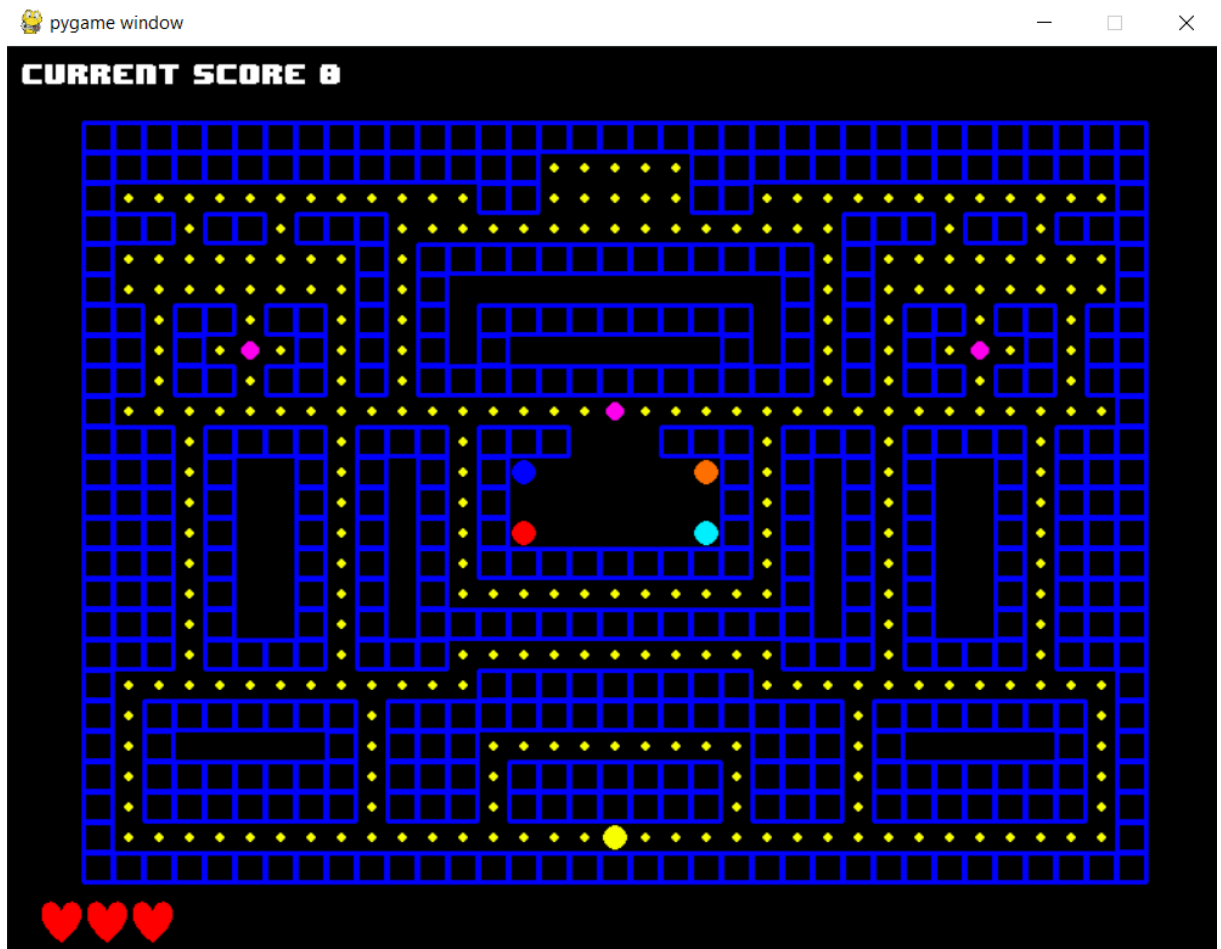


**Rysunek 7. „Okno Credits”**

## **2. Okno gry**

Okno gry wygląda jak na rysunku 8. W lewym dolnym rogu pokazano liczbę żyć gracza. W lewym górnym rogu pokazano wynik obecnej gry.

- Niebieskie kwadraty to ściany.
- Niebieski, błękitny, czerwony oraz pomarańczowy koła to wrogowie.
- Żółte większe koło – to gracz.
- Mniejsze żółte koła – to monety
- Różowe koła – to energetyki.



Rysunek 8. „Początek nowej gry”

### 2.1. Początek gry

Nowa gra ładuje się z pliku „lvl1.json”. Zapisana gra (jeżeli taka istnieje) wczytuje się z pliku „saved\_game.json”.

### 2.2. Struktura pliku wczytania gry

współrzędne w elementach „walls\_pos\_list” , „coins\_pos\_list” , „current\_grid\_pos” itp. pokazują pozycję klatki w siatce(szczególniej w punkcie 3.1. tej dokumentacji).

Ten plik zawiera takie elementy:

1. „walls\_pos\_list” – lista z współzrędnymi ścian.



2. „coins\_pos\_list” – lista z współrzędnymi monet.
3. „remaining\_time\_energizer” – zmienna, która pokazuje ile czasu działania energetyku zostało. Jeżeli równa się 0 – to znaczy, że czas działania energetyku się skończył lub że gracz jeszcze nie „napił się” energetyku.
4. „enemies” – lista z danymi o wrogach. Ona zawiera następujące elementy:
  - a. „current\_grid\_pos” – współrzędne klatki, gdzie obecnie znajduje się przeciwnik;
  - b. „start\_grid\_pos” – współrzędne klatki, gdzie po zresetowaniu przeciwnika on będzie się pojawił;
  - c. „color” – kolor w formacie RGB;
  - d. „movement\_mode” – określa w jaki sposób porusza się przeciwnik. Może być „random” lub „optimal”;
  - e. „speed” – wskazuje prędkość.
5. „player” – słownik z danymi o graczu. Zawiera elementy:
  - a. „current\_grid\_pos” – współrzędne klatki, gdzie obecnie znajduje się gracz;
  - b. „start\_grid\_pos” – współrzędne klatki, gdzie po zresetowaniu gracza będzie się pojawił;
  - c. „lives” – pozostała liczba żyć gracza.
  - d. „score” – obecny wynik gry
6. „energizers” – słownik z danymi o energetykach. Zawiera elementy:
  - a. „grid\_pos” – lista z współrzędnymi energetyków;
  - b. „color” – kolor w formacie RGB energetyków;
  - c. „seconds\_duration” – liczba sekund działania energetyków.

### 2.3. Pauza gry i okno wstrzymania gry

Żeby wstrzymać grę należy wcisnąć klawisz ESC. Pojawi się okno wstrzymania gry, pokazane na rysunku 9.

- Jeśli ikonka Pac-man'a wskazuje na „Continue” i wciśnięto ENTER, to odbywa się kontynuacja gry.
- Jeśli ikonka Pac-man'a wskazuje na „Save game” i wciśnięto ENTER, to odbywa się zapis obecnej gry do pliku „saved\_game.json”.
- Jeśli ikonka Pac-man'a wskazuje na „Go to main menu” i wciśnięto ENTER, to odbywa się przejście do głównego menu. Odbędzie się proces zapisu wyniku do listy wyników (szczególnie w punkcie 3.5. tej dokumentacji).



Rysunek 9. „Okno wstrzymania gry”.

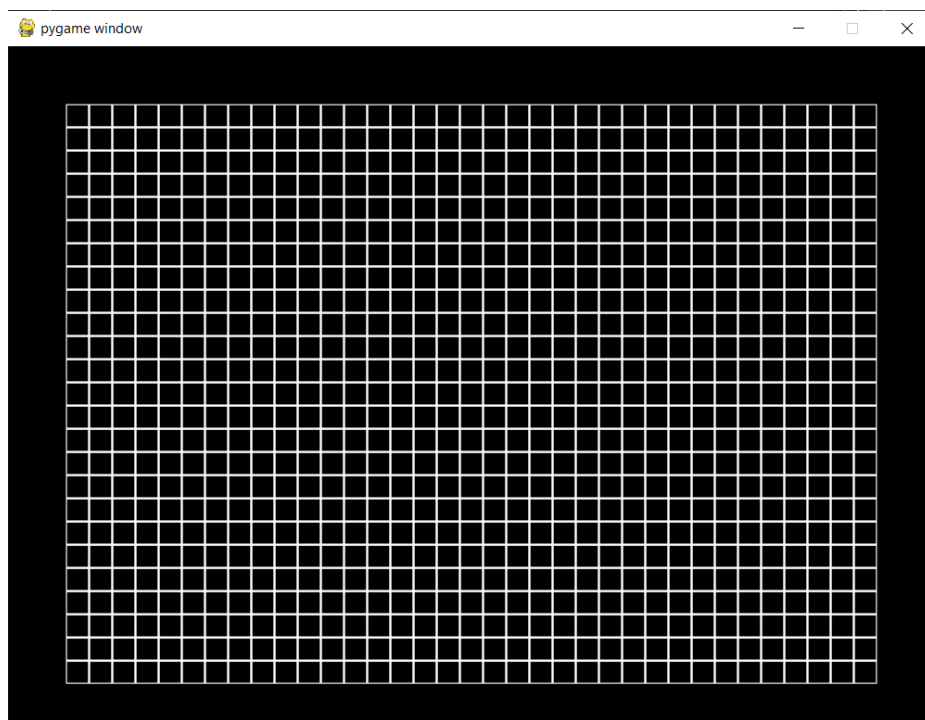
### 2.4. Zapis gry

Jak powiedziano wcześniej zapis gry odbywa się poprzez menu wstrzymania gry i pozycji „Save game” w tym oknie.

### 3. Elementy gry

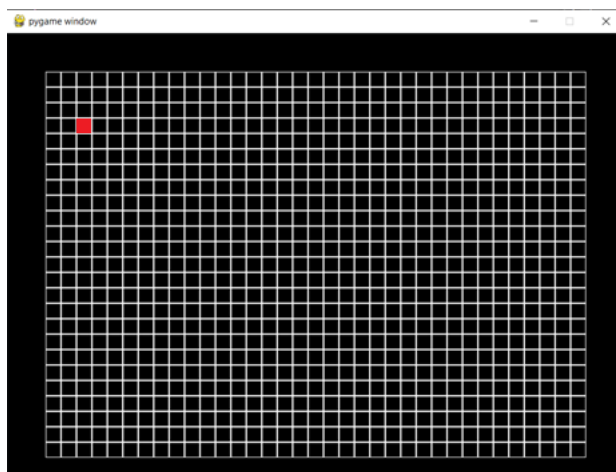
#### 3.1. Siatka

Każdy element tzn. moneta, gracz, ściana, energetyk, przeciwnik znajduje się na pełnej pozycji na siatce, którą pokazano na rysunku 10. Współrzędne klatki wskazują na lewy górny róg klatki.



Rysunek 10. „siatka”

W każdej z klatek może znajdować się jakiś element. Współrzędna  $[0, 0]$  odpowiada za klatkę, która znajduje się w lewym górnym rogu.  $[2, 3]$  – wskazuje na klatkę znajdującą się w trzeciej kolumnie( $2+1$ ) i czwartym wierszu( $3+1$ ) od lewego górnego rogu, czyli na czerwoną klatkę na rysunku 11.



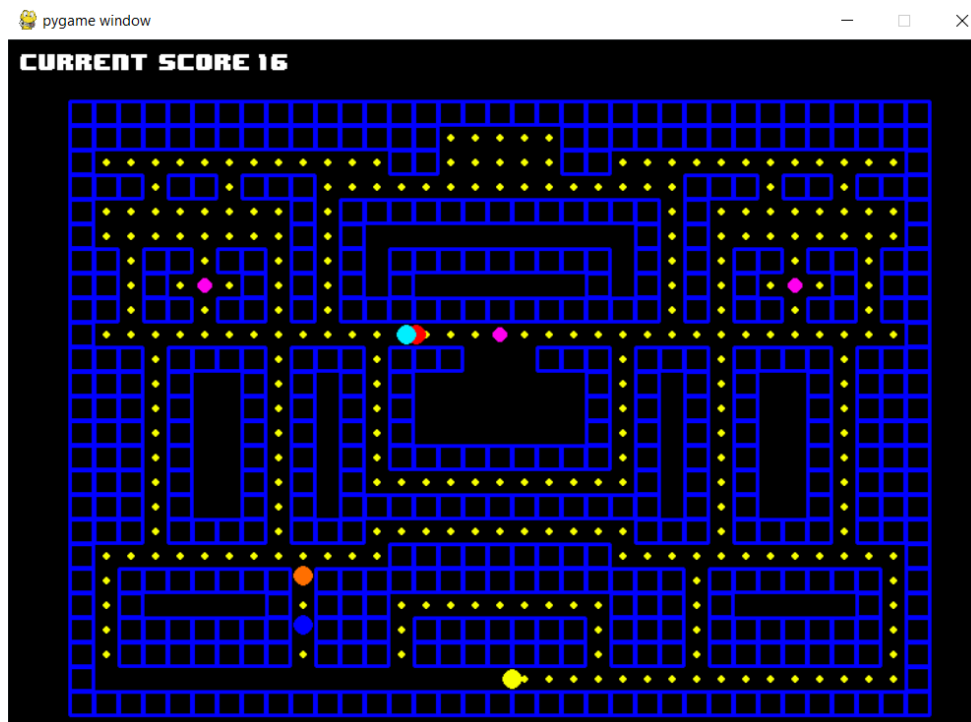
Rysunek 11. „Przykładowa klatka”

### 3.2. Gracz

Gracz jest obiektem którym steruje użytkownik.

#### 3.2.1. Sterowanie graczem

Sterowanie graczem odbywa się poprzez pojedyncze wciśnięcia klawisz strzałek w górę, w dół, w lewo oraz w prawo. Gracz będzie poruszał się w którymś z kierunków, póki nie zostanie zjedzony przez przeciwnika lub nie natrafi na ścianę lub nie zje wszystkie monety. Załóżmy, że jest sytuacja, którą pokazano na rysunku 12. I gracz porusza się w prawą stronę i wciśnięto klawisz strzałki w górę. W takim razie, informacja o tym, że wciśnięto ten klawisz zostanie zapisana do programu i kiedy gracz będzie miał możliwość zmienić kierunek poruszania się w górę, wtedy on zmieni swój kierunek, a do tego momentu będzie się poruszał w prawą stronę.



Rysunek 12. „Przykładowe rozmieszczenie elementów gry”

### **3.2.2. Zderzenie z innymi elementami**

Zderzenia z elementami odbywa się na trzy sposoby:

- a. Jeśli piksel, gdzie znajduje się gracz oraz piksel gdzie znajduje się element jest ten samy(takie zdarzenie odbywa się dla gracza z monetami, energetykami).
- b. Jeśli odległość między graczem a innym elementem jest mniejsza niż określona wartość, to odbyło się zdarzenie. Taka logika działa z przeciwnikami.
- c. Jeśli pozycja następnej w podążającym kierunku klatki jest zajęta innym elementem, to odbyło się zdarzenie. Taka logika działa dla ścian.

Jeśli gracz zderzył się z monetą, to ta moneta zostanie usunięta z gry, a wynik obecnej gry zwiększy się o jeden.

Jeśli zderzył się z energetykiem, to ten energetyk zostanie usunięty z mapy oraz zacznie swoją działalność.

Jeśli zderzył się z wrogiem, to możliwe są dwa następstwa:

- a. Jeśli energetyk wciąż działa, to gracz zje przeciwnika i ten pojawi się na swojej pierwotnej pozycji(start\_grid\_pos).
- b. Jeśli w tej chwili nie działa energetyk, to u gracza odejme się jedno życie, pozycji gracza i przeciwników zostaną zresetowane.

Jeśli zdarzył się z ścianą, to przestanie się poruszać.

## **3.3. Wróg**

### **3.3.1. Zresetowanie**

Zresetowanie znaczy zmiana obecnej pozycji, trybu poruszania się do pierwotnej.

### **3.3.2. Tryb się poruszania**

#### **3.3.2.1. Random**

Tryb poruszania się „random” znaczy, że najpierw przeciwnik analizuje w którą stronę on może pójść, czyli w której stronie nie znajduje się ściana.

Po tym procesie wybierze się przypadkowo kierunek się poruszania.

Następnie program analizuje jak długo w tym kierunku może poruszać się przeciwnik, czyli ile klatek w tym kierunku nie są zajęte ścianami. Potem wybiera się przypadkowo liczba, niech będzie to liczba  $n$ , która jest w przydziale od 1 do maksymalnej liczby kroków poruszania się w tym kierunku. Po tym całym procesie przeciwnik będzie się poruszał w tym kierunku  $n$  kroków. Potem proces się powtórzy.

#### **3.3.2.2. Optimal**

Tryb poruszania się „optimal” znaczy, że przeciwnik szuka najkrótszej drogi między sobą a graczem, uwzględniając pozycji ścian. Dla realizacji takiej funkcjonalności skorzystano z algorytmu przeszukiwania wszerz. Po każdym kroku przeciwnika dane o znajdowania gracza oraz pozycję przeciwnika odświeża się.

### **3.4. Energetyk**

Energetyk jest elementem gry, który pozwala graczowi przez ograniczony czas zjadać przeciwników. Po „wypiciu” energetyku zaczyna się zegar odliczający ile czasu działania energetyku zostało się. Póki ta wartość nie jest mniejsza od 0 gracz może zjadać przeciwników. Czyli jeżeli energetyk działa i gracz zetknął się z wrogiem, to pozycja wroga resetuje się do pierwotnej. Także podczas działania energetyku tryb poruszania się każdego przeciwnika zmieniono na „random”, a także kolor na biały.

### 3.5. Zapis wyników

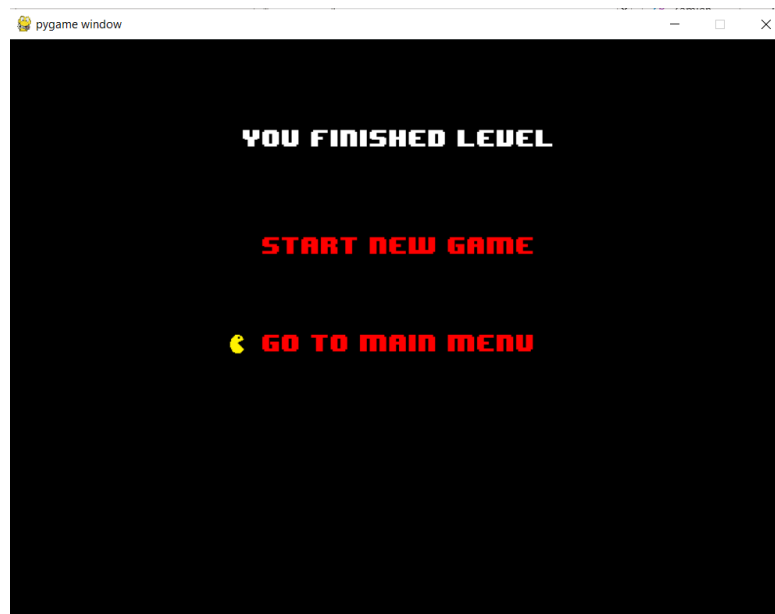
Zapis wyników gry zaczyna się za każdym razem, gdy gracz:

- a. zaczął grę i podczas gry wcisnął pauzę i przeszedł do głównego menu ;
- b. stracił wszystkie życia i gra się skończyła;
- c. skończył grę i przeszedł do głównego menu.

Wyniki gry znajdują się w pliku „score.csv”. Jeżeli wynik podczas gry jest wyższy niż najmniejszy wynik z listy wyników lub ta lista zawiera mniej niż 10 elementów, to ten wynik zapisze się do pliku „score.csv”.

### 3.6. Ukończenie gry

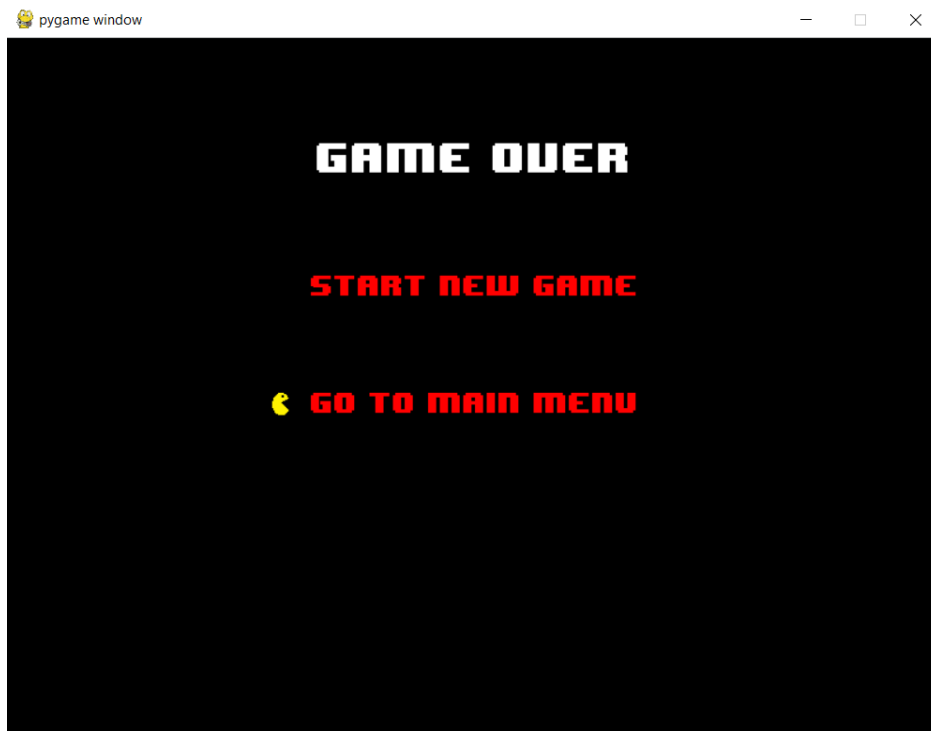
Jeśli gracz zjadł wszystkie monety na mapie, to wyświetli się okno pokazane na rysunku 13. Po skończeniu gry gracz może przejść do głównego menu lub zacząć nową grę z liczbą żyć oraz wynikiem, które została przechowane z poprzedniej rundy.



Rysunek 13. „Okno skończenia gry”

### 3.7. Przegranie gry

Jeśli u gracza zostało 0 żyć, to wyświetli się okno pokazujące taką informację, które znajduje się na rysunku 14. Z tego okna użytkownik może zacząć nową grę lub przejść do głównego menu.



Rysunek 14. „Okno przegrania gry”