

Platformy programistyczne .Net i Java

Projekt

Wielowątkowa aplikacja okienkowa w Javie

Prowadzący	Wykonawcy	Nr. Indeksu	Data
mgr inż. Aneta Górniak	Patryk Marciniak Mateusz Bartkowiak	248978 249467	11.06.2021

1 Działanie aplikacji od strony użytkownika

Zaprojektowaną aplikacją jest gra Ping-Pong. Na początku gracz ma możliwość wczytania poprzedniej rozgrywki jeśli takowa istnieje albo zaczęcie gry od zera. Jest to wersja klasycznego ping-ponga w której za pomocą strzałek przesuwamy platformę w taki sposób by zapobiec dotknięcia piłki po naszej stronie ściany jednocześnie próbując trafić w ściankę przeciwnika. Naszym rywalem jest sztuczna inteligencja, który wydaje się być przeciętnym graczem, jednocześnie będąc na takim poziomie aby dać możliwość użytkownikowi zaciętą rywalizację. Kolejnym aspektem rozgrywki jest przeszkoda, znajdująca się na środku planszy, która jest w ciągłym ruchu, więc może się okazać że odbita piłka niekoniecznie może polecieć na pole przeciwnika lecz do nas zwróci. Ostatnim elementem utrudniającym rozgrywkę na pewno jest fakt, że w raz z kolejnym odbiciem piłki od przeszkody lub platformy, nabiera ona większej prędkości co znacznie utrudnia rozgrywkę użytkownikowi jak i przeciwnikowi sterowanym przez algorytm sztucznej inteligencji. Jeśli znajdzie potrzeba wyjścia lub przerwania rozgrywki, nie trzeba się martwić o stan gry, ponieważ użytkownik zostanie zapytany czy chce taką grę zapisać. Ping-Pong został oprawiony w łatwy do użytkowania interfejs a gra jest przejrzysta i płynna.

2 Zapis danej rozgrywki oraz wczytywanie gry

Elementem, którego nie mogło zabraknąć jest zapisywanie rozgrywki, które odbywa się kiedy użytkownik postanowi opuścić rozgrywkę. Zapisywanie i wczytywanie jest realizowane przez zapis i odczyt z pliku tekstowego, znajdującego się w folderze *saves*, w którym umieszczone są wyniki obu graczy. Na samym początku sprawdzamy czy istnieje już plik *save.txt* w folderze *saves*. Jeżeli tak, to podmieniamy wartości na aktualny wynik z rozgrywki. W zapisywanym pliku do pierwszej linijki zapisujemy punkty pierwszego gracza natomiast do drugiej drugiego. Natomiast jeżeli taki plik nie istnieje, najpierw tworzymy takowy i postępujemy analogicznie, jak w przypadku gdy taki już istnieje.



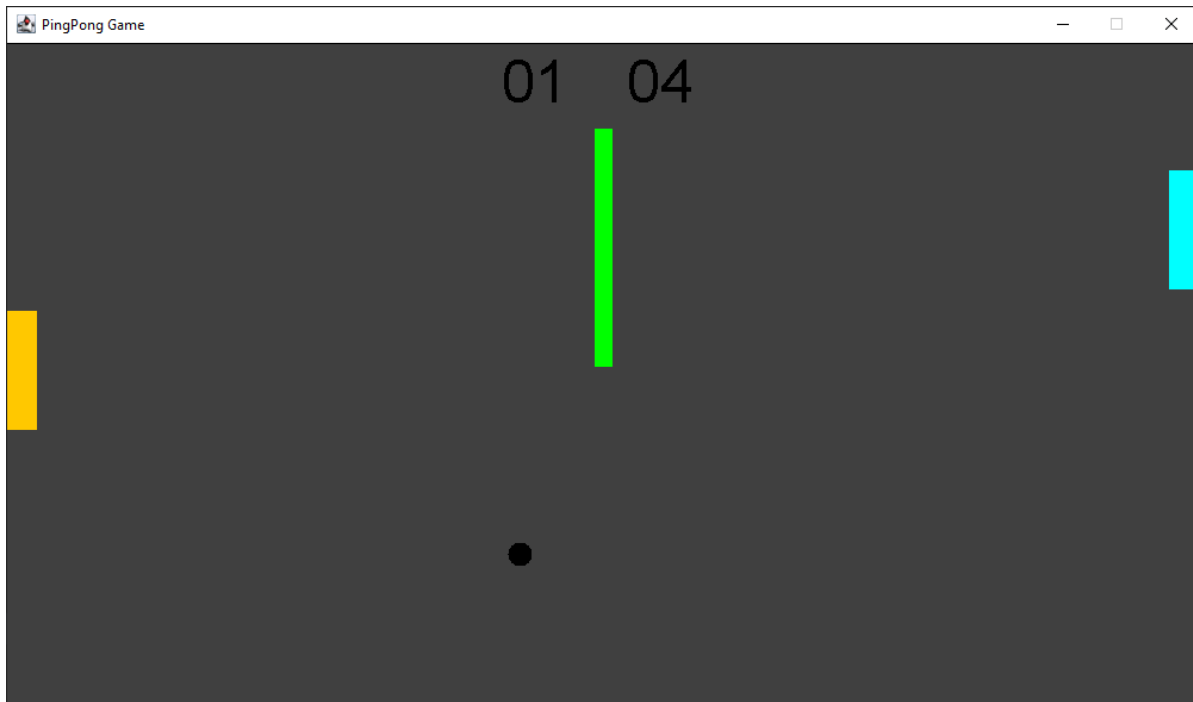
Rysunek 1: Pierwsze menu, ukazujące się po włączeniu aplikacji



Rysunek 2: Drugie menu, ukazujące się po próbie zamknięcia aplikacji

3 Główna pętla gry i wielowątkowość

Najważniejszą częścią gry jest główna pętla, która znajduje się w nadrzędnym wątku. W pętli odbywa się sprawdzanie tak zwanego "ticku" czyli weryfikacji czy nie upłynęło zbyt mało czasu między kolejnymi ruchami. Takie podejście pozwala na optymalniejszą rozgrywkę i zapewnienie, że wszelkie akcje odbywają się z określoną częstotliwością. Po tym, jak minie zdefiniowany "tick", uruchamiamy jedną z ważniejszych funkcji *moveObjectAndCheckForCollisions()*, w której tworzone są cztery wątki. Odpowiednio dla pierwszego gracza, drugiego gracza, piłki oraz przeszkody. Takie wątki po utworzeniu są uruchamiane z pomocą metody *start()* a następnie przy pomocy *join()* czekamy na wszystkie wątki aż skończą się wykonywać. Każdy wątek wykonuje ruch odpowiadającym mu elementom a następnie sprawdza, czy zaistniała kolizja. Ostatnim elementem jest ponowne narysowanie elementów na naszej planszy.



Rysunek 3: Widok okna z grą, z widocznymi: kładkami reprezentującymi graczy, piłką oraz przeszkodą

4 Algorytm sztucznej inteligencji

Drugi gracz, sterowany za pomocą algorytmu sztucznej inteligencji, został zaimplementowany w taki sposób aby jego ruchy były dość przemyślane ale jednocześnie nie pozwalały zawsze wygrywać. Wszystko zostało oparte na przemyśleniu kilku możliwości:

- jeśli piłka nie leci w jego kierunku, to przemieszcza się na środek tak aby przygotować się i mieć możliwość odbicia piłeczki z obu stron,
- jeśli piłka leci w dół planszy, to stara się także przemieścić w tę stronę, chyba że już znajduje się po tej stronie
- jeśli piłka leci w górę planszy, to analogicznie jak w poprzednim przypadku, stara się przemieścić w górę, o ile nie znajduje się już po tej stronie