

Table Tennis



TableTennis

DOKUMENTACJA

JĘZYK POLSKI

AUTOR: LIGAS NORBERT

AKADEMIA GÓRNICZO- HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE
WYDZIAŁ INFORMATYKI,
ELEKTRONIKI I TELEKOMUNIKACJI
KIERUNEK ELEKTRONIKA

SPIS TREŚCI

Wstęp	2
Lista oznaczeń	2
Opis	2
Wymagania Systemowe	3
Funkcjonalność	4
Analiza problemu	5
Projekt Techniczny	6
Diagram Klas	6
Diagram Stanów / Aktywności	7
Opis Realizacji	8
Opis wykonanych testów	11
Podręcznik użytkownika	12
Opis gry	12
Metodologia rozwoju i utrzymania systemu	13
Rozwój programu	13
Podziękowania	14

WSTĘP

LISTA OZNACZEŃ

Bot	Paletka sterowana przez komputer w trybie gry <i>Ty VS Komputer</i>
Rakietka	Paletka do tenisa stołowego używana w grze przez użytkownika/ów lub Bota
Plansza	Okno aplikacji w trybach gry
Stół	Widoczny w trybach gry stół tenisa stołowego
Pole	Obszar w którym może poruszać się konkretny gracz
Menu	Menu gry z możliwością wybrania trybu, bądź opuszczenia rozgrywki
Piłeczka	Piłeczka używana do gry

OPIS

Program jest grą typu „Pong”, lecz rozbudowaną na 3 różne tryby (*Ty VS Komputer, Gra dla dwojga, Ping-pongowy „Debel”*). Oprócz grafiki oraz większych możliwości manewru rakietką gra nie odbiega mechaniką od klasycznej wersji. Program został napisany w języku C++ z użyciem wieloplatformowej biblioteki multimedialnej *Simple and Fast Multimedia Library (SFML)*.

WYMAGANIA SYSTEMOWE

- Windows 7 lub nowszy
- Pliki z teksturami dołączone w folderze z grą (łącznie 5)
- Plik z czcionką dołączony w folderze z grą (Anton-Regular.ttf)
- Pliki dll dołączone w folderze z grą (msvcp140d.dll, ucrtbase.dll, ucrtbased.dll, vcruntime140d.dll) jeśli wystąpi komunikat o braku jakiegoś pliku z rozszerzeniem dll należy doinstalować odpowiedni
- Pliki dll biblioteki multimedialnej SFML (sfml-graphics-2.dll, sfml-graphics-d-2.dll, sfml-system-2.dll, sfml-system-d-2.dll, sfml-window-2.dll, sfml-window-d-2.dll)
- Rozdzielczość ekranu 1920X1080

FUNKCJONALNOŚĆ

Do funkcjonalności gry można zaliczyć:

- Przejrzyste menu z podświetlaniem wybieranej opcji (dostęp z klawiatury i myszki)
- 4 języki komunikacji z graczem
- 3 tryby gry
- Możliwość poruszania rakiety w każdym z kierunków (również przez bota)
- Możliwość włączenia pauzy w dowolnym momencie i łatwego przywrócenia gry
- Zliczanie punktów za przekroczenie granicy planszy przez piłeczkę i wyświetlanie wyniku na stole
- Brak możliwości przekroczenia rakieta swojego pola
- Odbijanie piłeczki pod odpowiednim kątem względem ustawienia rakiety
- Przyspieszanie ruchu piłeczki z każdym odbiciem
- Łatwy powrót do menu
- Zmiana strony rakiety w zależności od kierunku ruchu

ANALIZA PROBLEMU

- Poruszanie po menu
- Zmiana języków
- Dla każdego trybu
 - Ustawianie rakietek i piłeczki na planszy
 - Odpowiedni kąt odbicia piłeczki względem ustawienia rakietki
 - Sprawdzenie przekroczenia granicy planszy przez piłeczkę i naliczanie punktów
 - Załączanie pauzy po wciśnięciu odpowiedniego przycisku
 - Wyświetlanie wyników po skończonej grze z możliwością restartu gry lub powrotu do menu
 - Wyświetlanie wyniku gry na bieżąco
 - Nieprzekraczanie swojego pola rakietką
 - Zmiana strony rakietki w zależności od kierunku ruchu
- Dla trybu *Ty VS Komputer*
 - Odpowiednie wyliczanie miejsca, w którym piłeczka powinna przekroczyć granicę planszy
 - Ustawianie bota

PROJEKT TECHNICZNY

DIAGRAM KLAS

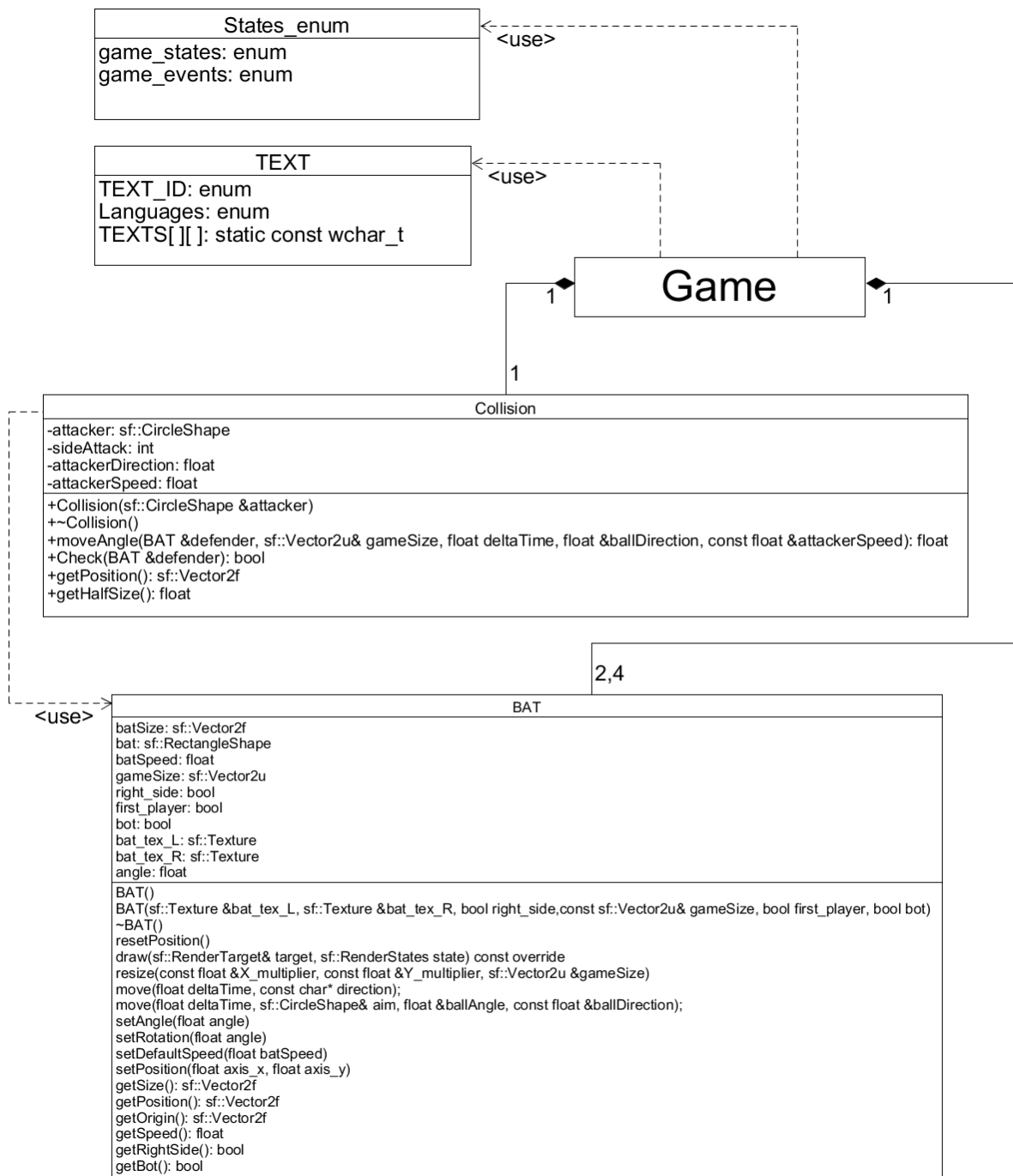
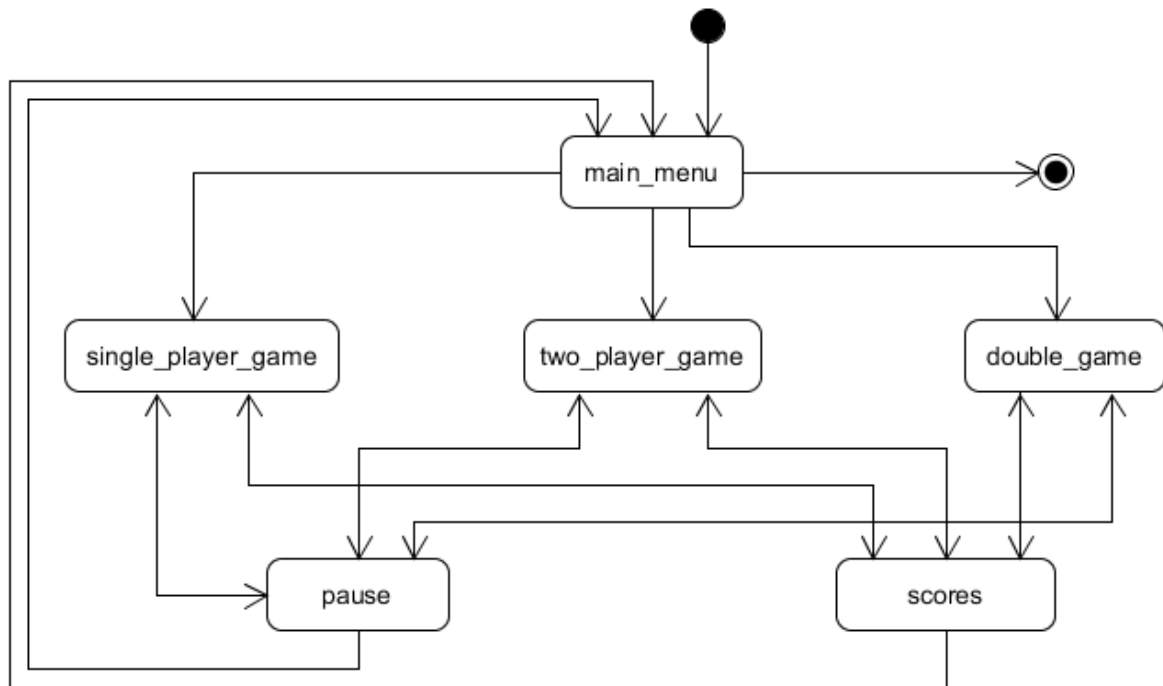


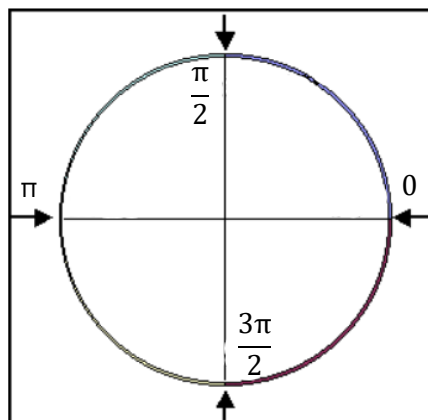
DIAGRAM STANÓW / AKTYWNOŚCI



OPIS REALIZACJI

Program został napisany w Visual Studio 2017. Efekty multimedialne zostały wykonane za pomocą bibliotek SFML 2.5.1. Głównymi problemami i ich rozwiązaniami są:

- **Poruszanie po menu:** Możliwy jest dostęp do wybierania opcji za pomocą myszki lub „strzałek” na klawiaturze. Na podstawie pozycji myszy określone jest czy znajduje się ona na którejś z opcji. Możliwość zmiany opcji przez klawiaturę istnieje tylko wtedy, gdy myszka nie znajduje się na żadnej z opcji. W przypadku, gdy myszka po włączeniu programu nie znajduje się na żadnej opcji z klawiszy, można poprzez kliknięcie strzałki w dół namierzyć pierwszą opcję lub ostatnią z opcji poprzez kliknięcie w górę. Klawiszem *Enter* lub *LPM* można przejść do konkretnego stanu (trybu gry) i rozpocząć grę lub wyjść z aplikacji.
- **Zmiana języków:** Zmiana języków komunikacji, dostępna tylko w menu z graczem, zrealizowana jest przez kliknięcie na klawiaturze odpowiedniej litery odpowiadającej pierwszej literze nazwy jednego z 4 języków w wersji angielskiej. Instrukcja znajduje się w prawym dolnym rogu okna aplikacji.
- **Dla każdego trybu**
 - **Ustawianie rakietek i piłeczki na płaszy:** Przy każdym przejściu do stanu „pause” lub „scores” pozycje paletek i piłeczki resetują się do domyślnych wartości
 - **Odpowiedni kąt odbicia piłeczki względem ustawienia rakietki:** Problem rozwiązuje klasa Collision. Na podstawie odbicia od połowy rakietki i wyżej, kąt odbicia będzie ustawiony odpowiednio w „dół” ekranu, natomiast od połowy i niżej, kąt odbicia pokieruje piłeczkę w górę okna. Kąt jest wyrażony w radianach i jego rozłożenie można przedstawić na poniższym wykresie:

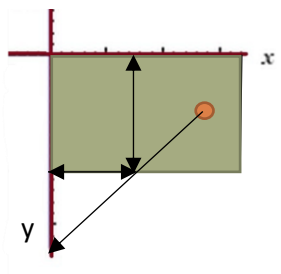


Jest on losowany pseudolosową funkcją `rand` uzależnioną od czasu i w zależności od strony okna na której się znajduje i powyższych warunków ustawiany jako zmienna `ballAngle`. Kąt w każdym z przypadków jest losowany.

- **Sprawdzenie przekroczenia granicy planszy przez piłeczkę i naliczanie punktów:** Zmienna `collision` zapisuje odpowiednio 1 w przypadku przegroczenia granic okna w jego lewej połowie, lub 2 w jego prawej, natomiast ma wartość 0 przy braku kolizji z ramą okna. Odpowiednio kolejna funkcja przechwytuje te wartości i na ich podstawie nalicza punkty. Kolizja z ramą sprawdzana jest na podstawie prymitywnych metod sprawdzania pozycji.
- **Załączanie pauzy po wciśnięciu odpowiedniego przycisku:** Przycisk *Escape* powoduje przełączenie stanu na „pause”. Resetuje to pozycje rakietek i piłeczki na ich domyślne wartości nie zerując wyniku. Przyciśnięciem klawisza *Enter* powracamy do gry.
- **Wyświetlanie wyników po skończonej grze z możliwością restartu gry lub powrotu do menu:** Po naliczeniu 21 punktów jednej z drużyn wyświetla się powiadomienie o wynikach z informacją, która strona wygrała. Klikając *Escape* przechodzimy do stanu „menu”, natomiast klikając *Enter* restartujemy grę.
- **Wyświetlanie wyniku gry na bieżąco:** Wynik na bieżąco jest aktualizowany za pomocą interpretacji zmiennej `collision` i wyświetlany na stole.
- **Nieprzekraczanie swojego pola rakiетką:** Na podstawie warunków rakiетki ustawione z prawej strony stołu nie mogą przekroczyć połowy okna poruszając się w lewą stronę okna. Odpowiednio tym samym sposobem blokowany jest ruch rakiетek z lewej strony.
- **Zmiana strony rakiетki w zależności od kierunku ruchu:** W zależności od kierunku ruchu w klasie „BAT” funkcja odpowiedzialna za poruszanie się rakiетki ładuje odpowiednią teksturę bekhenda, bądź forhenda.

➤ Dla trybu *Ty VS Komputer*

- **Odpowiednie wyliczanie miejsca, w którym piłeczka powinna przekroczyć granicę planszy:**



Pozycja piłeczki przyjmuje wartości względem okna przedstawionego jako prostokąt na rysunku powyżej. Na podstawie wzorów:

```
crossPoint.y = ballPosition.y - (sin(ballAngle)/ cos(ballAngle)) * ballPosition.x
```

Dla liczenia przecięcia z osią OX:

```
crossPoint.x = ballPosition.x - abs(cos(ballAngle)/ sin(ballAngle)) * ballPosition.y) + 200
```

Dla liczenia przecięcia z dolną krawędzią okna:

```
crossPoint.x = abs(cos(ballAngle)/ sin(ballAngle)) * (crossPoint.y + rozmiar_paletki + 50 -  
ballPosition.y) + 330
```

Otrzymywane w ten sposób koordynaty dają informacje na temat miejsca przecięcia się drogi piłeczki i krawędzi okna.

- **Ustawianie bota:** Bot na bieżąco otrzymuje koordynaty i ustawia się na drodze piłeczki, gdy ta zmierza w jego stronę (informacja ta jest pozyskiwana z cosinusa z ballAngle, który wraz z kierunkiem lotu piłeczki zmienia znak na przeciwny). W momencie gdy piłeczka nie porusza się i znajduje się na połowie bota on porusza się bezpośrednio w jej kierunku, natomiast gdy piłeczka nie leci w jego kierunku albo nie porusza się będąc na połowie przeciwnika, bot przemieszcza się w kierunku domyślnej pozycji startowej.

OPIS WYKONANYCH TESTÓW

Kod	Data	Autor	Opis	Stan
0.001	01.02.2019	Zuzanna Węglarz	Bot pojawia się spóźniony na początku gry. Poprawny respawn nie gwarantuje jego pozycji.	Naprawiony
0.002	01.02.2019	Tomasz Wachała, Marcin Węglarz, Zuzanna Węglarz	Poruszanie bota tylko w górę lub dół.	Naprawiony
0.003	01.02.2019	Hubert Mikołajczyk, Tomasz Wachała, Marcin Węglarz, Krzysztof Pokora	Bot drga na tyle mocno, że kilkakrotnie odbija piłeczkę powodując nawet samobójcze punkty.	Naprawiony
0.004	02.02.2019	Tomasz Wachała	Serwowanie do tyłu.	Naprawiony
0.005	03.02.2019	Ligas Norbert	Zmiana rozmiaru okna powoduje „armagedon” obliczeniowy i nie przewidywalne zjawiska w grze.	Zabezpieczony
0.006	03.02.2019	Krzysztof Pokora	Odbijanie piłeczki głównie w dół, praktycznie nigdy w górę okna.	Naprawiony
0.007	04.02.2019	Hubert Mikołajczyk	Paletki respawnują się poza oknem, albo na jego krawędzi (pkt. 0,0).	Naprawiony
0.008	05.02.2019	Ligas Norbert	Bot nie broni dolnej krawędzi jego połowy.	Naprawiony
0.009	06.02.2019	Ligas Norbert	Bot ucieka przed piłeczką przy dolnej krawędzi.	Naprawiony

PODRĘCZNIK UŻYTKOWANIA

OPIS GRY

Gra odbywa się na prostokątnej planszy 1800x1016 px. Piłeczka domyślnie ustawiana jest na prawej stronie planszy. Uderzenie rakiетką piłeczki powoduje wykrycie kolizji i nadanie piłeczce kierunku i prędkości. Gracze poruszają paletkami za pomocą klawiszy strzałek w przypadku pierwszego z graczy z prawej połowy stołu, klawiszy W, A, S, D w przypadku pierwszego z graczy z lewej strony stołu w trybach gry „gra dla dwojga” lub „pingpongowy „debel””. Kolejni gracze z ostatniego z trybów kontrolują paletkę odpowiednio z lewej połowy stołu klawiszami Y, G, H, J, a z prawej Numpad8, Numpad4, Numpad5, Numpad6. Klawisze w kolejności odpowiadają ruchowi w górę, w lewo, w dół lub w prawo. Gracz ma o tyle wpływ na kąt odbicia piłeczki, że uderzając w nią górną połową rakiетki (wyżej znajdującą się połową względem okna) piłeczka odbije się w dół, a w przeciwnym przypadku odpowiednio w górę. Punkty naliczane są odpowiednio w zależności, na której połowie piłeczka przeleci poza krawędź ekranu. Po naliczeniu za to punktu odpowiedniej drużynie piłeczka ustawia się na serw z odpowiedniej wedle zasad strony stołu. Należy pamiętać, że jest to gra i piłeczka nie musi przelecieć w ogóle nad narysowanym stołem. Mecz trwa do momentu, aż któryś z graczy lub drużyna osiągnie wynik 21 punktów.

METODOLOGIA ROZWOJU I UTRZYMANIA SYSTEMU

ROZWÓJ PROGRAMU

- Usprawnienie „sztucznej inteligencji” bota
- Usprawnienie zmiany tekstur rakietek w zależności od kierunku
- Dołożenie pozostałych 3 języków
- Zwiększenie możliwości poruszania się bota z dwóch kierunków na każdy
- Zwiększenie prędkości poruszania się rakietek
- Zwiększenie prędkości poruszania się rakietki bota
- Usprawnienie obliczania kąta odbicia piłeczki

PODZIĘKOWANIA

Szczególne podziękowania dla :

- Zuzanny Węglarz- tłumaczenie na język włoski i francuski
- Zuzanny Węglarz, Huberta Mikołajczyka, Tomasza Wąchały,
Marcina Węglarza, Krzysztofa Pokory- testowanie gry
- Marcina Węglarza- wspólne rozważania trygonometryczne