Programowanie obiektowe  
Gra Tetris

Autor: Patryk Tatar

Grupa

Informatyka

Przedmiot

Prowadzący:

# Krótki opis projektu

W ramach projektu zaliczeniowego została wykonana kopia gry tetris.

Tetris – komputerowa gra logiczna stworzona przez Aleksieja Pażytnowa i jego współpracowników, Dimitrija Pawłowskiego i Wadima Geriasimowa. Pojawiła się na rynku po raz pierwszy 6 czerwca 1984 roku w Związku Radzieckim.

Oryginalna wersja powstała podczas pracy zespołu Pażytnowa w Akademii Nauk ZSRR w Moskwie na komputerze Elektronika 60. Jest to jedna z najbardziej znanych gier komputerowych, posiadająca dużą liczbę różnorodnych wariacji i wariantów. Do powstania gry zainspirowała Pażytnowa układanka Pentomino.

# Opis wykorzystanej technologii – język C++

C++ został zaprojektowany przez Bjarne Stroustrupa jako rozszerzenie języka C o obiektowe mechanizmy abstrakcji danych i silną statyczną kontrolę typów. Zachowanie zgodności z językiem C na poziomie kodu źródłowego pozostaje jednym z podstawowych celów projektowych kolejnych standardów języka.

## *Właściwości języka*

* Język C++ jest językiem wieloparadygmatowym. Oznacza to, że można w nim stosować jednocześnie różne style programowania, w tym programowanie proceduralne, obiektowe, generyczne, jak również programować na poziomie asemblera.
* Język C++ zakłada statyczną kontrolę typów; posiada też elementy dynamicznej kontroli typów.
* Język C++ umożliwia bezpośrednie zarządzanie wolną pamięcią.
* Projekt języka zakłada, że żadna nowa (względem języka C) cecha języka C++ nie może mieć negatywnego wpływu na szybkość działania programu lub zapotrzebowanie na pamięć operacyjną. Dzięki temu dobrze napisany program w C++ jest z reguły co najmniej równie szybki, jak jego odpowiednik napisany w C; co więcej, dzięki możliwości zastosowania algorytmów generycznych w wielu przypadkach C++ jest wyraźnie szybszy od C (np. podczas sortowania).
* Język C++ ze względu na bardzo rozbudowaną składnię oraz surowe wymogi w zakresie wydajności kodu stanowi duże wyzwanie dla twórców kompilatorów.

# Opis wykorzystanej technologii – SFML

SFML jest darmową biblioteką, która dostarcza niskopoziomowy oraz wysokopoziomowy dostęp do karty graficznej, urządzeń wejściowych, dźwięku itp. Biblioteka SFML zyskała bardzo dużą popularność ostatnich latach i obecnie jest najczęściej polecaną oraz najchętniej wybieraną technologią przez deweloperów do wytwarzania gier 2D.

W skład SFML wchodzi pięć modułów:  
System – obsługuje czas i wątki  
Window – obsługuje okna i interakcję z użytkownikiem  
Graphics – umożliwia renderowanie grafiki  
Audio – dostarcza interfejsu do odtwarzania muzyki i dźwięków  
Network – odpowiedzialny za komunikację sieciowąNiemal wszystkie moduły mogą być używane niezależnie – wyjątek stanowi moduł Graphics (który wymaga modułu Window) oraz moduł System wymagany przez wszystkie inne.

# Wykorzystany wzorzec projektowy

Singleton – kreacyjny wzorzec projektowy, którego celem jest ograniczenie możliwości tworzenia obiektów danej klasy do jednej instancji oraz zapewnienie globalnego dostępu do stworzonego obiektu.

Dzięki zastosowaniu singletonu mamy jeden manager scen, dzięki któremu w łatwy sposób możemy przełączać okna.

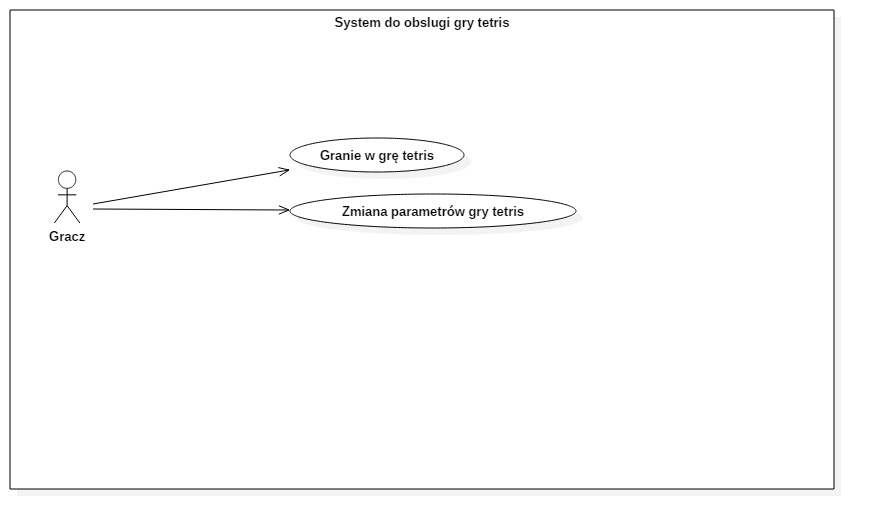
# Opis działania programu

Po uruchomieniu programu wyświetla się menu z przyciskami „play” oraz „quit game”.

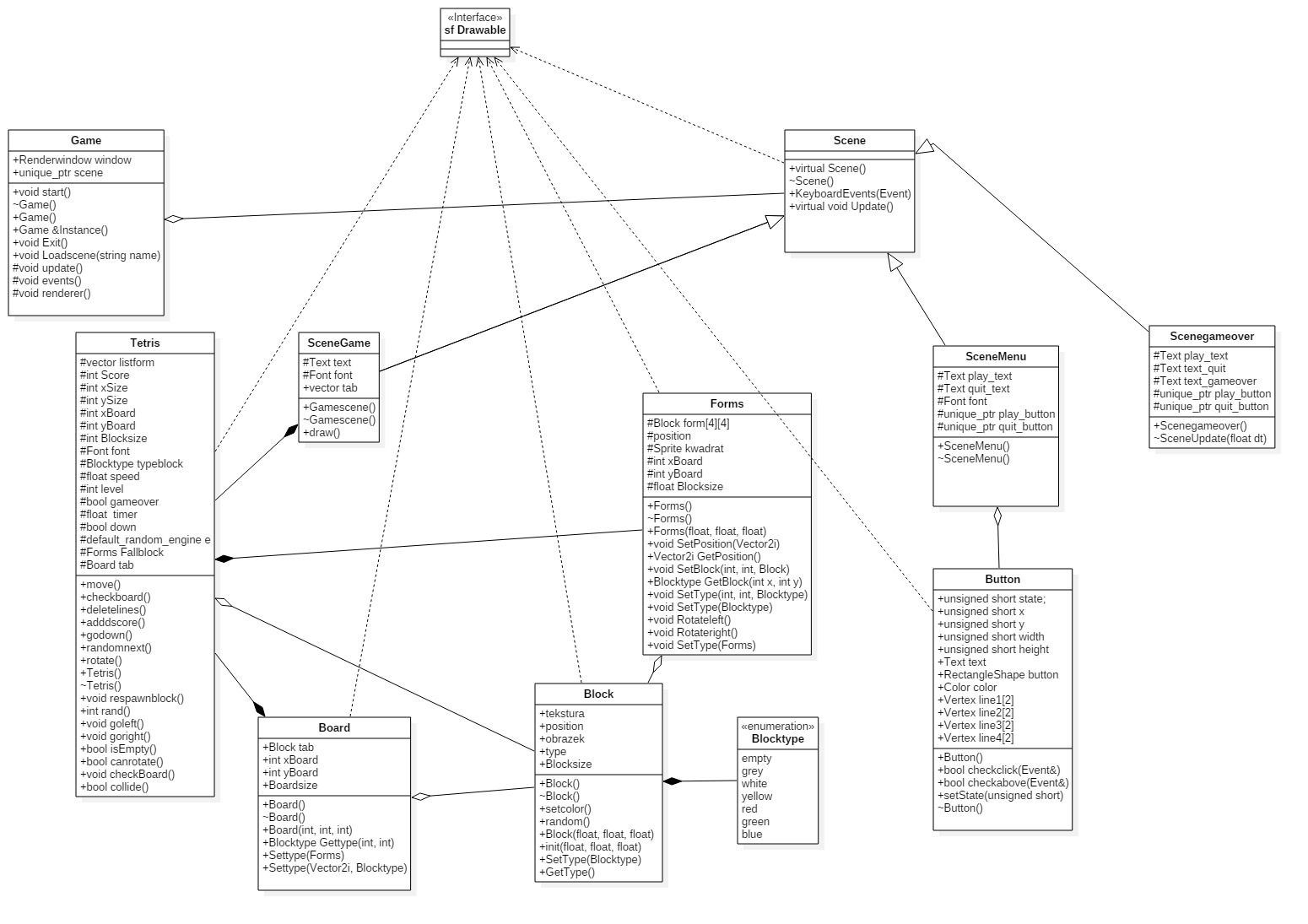
W momencie wyboru przycisku „quit game” gra kończy działanie.

Gdy wybierzemy przycisk „play” zostajemy przekierowani do okna gry (planszy). Możemy wtedy zagrać w grę. Gra polega na sterowniu klockami w prawo lub lewo (sterujemy strzałkami) oraz obracania (klawisze „a” oraz „d”). Jeżeli chcemy zwiększyć prędkość klocka klikamy strzałkę w dół. W grze jest zapewniona kontrola, czy możemy obrócić klockiem. Gdy klocek jest w miejscu docelowym zostaje na planszy. W momencie, gdy ułożymy jedną linię, ona znika, a nam zwiększa się level i szybkość klocka. Gdy nie można już umieścić klocka na planszy przegrywamy grę i wyświetla nam się menu z informacją o przegranej oraz przyciskami „new game” i „quit”.

# Diagram UseCase



# Diagram klas

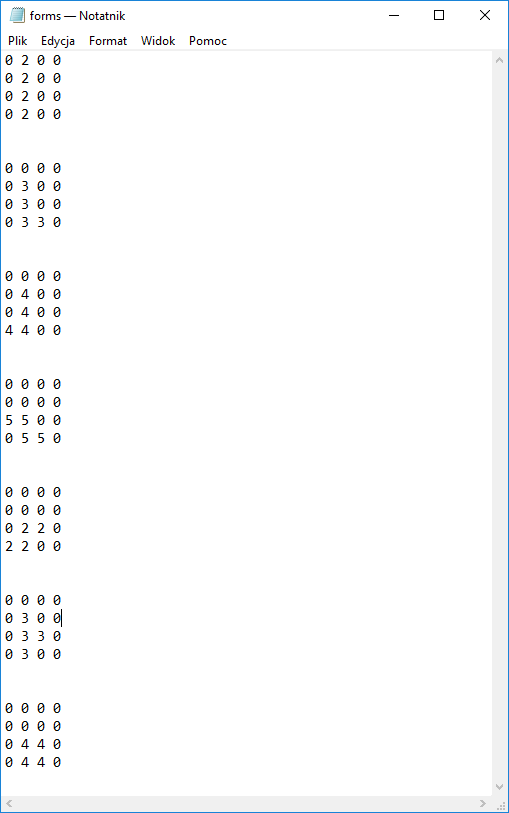


# Spis założeń

## *Gra powinna udostępniać możliwość definiowania klocków o dowolnym, spójnym kształcie i rozmiarze maksymalnym 4x4 a minimalnym 1x1*

## *Gra musi zawierać standardowe klocki z gry tetris*

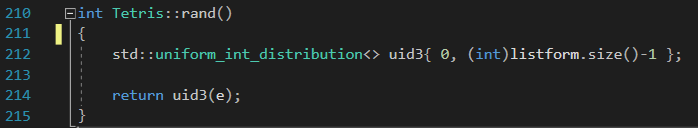
Kształty mogą zostać zdefiniowane w pliku forms.txt



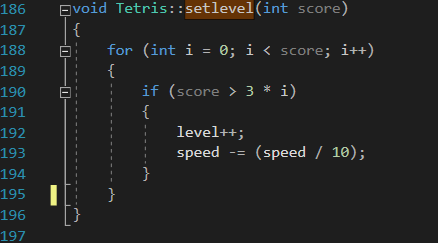
W pliku jest zdefiniowane 7 podstawowych kształtów. Numer oznacza typ bloku (kolor)

## *Gra powinna udostępniać możliwość zmiany ustawień: szybkość, prawdopodobieństwo wylosowania klocka.*

Losowanie klocka: (*klasa Tetris.cpp)*

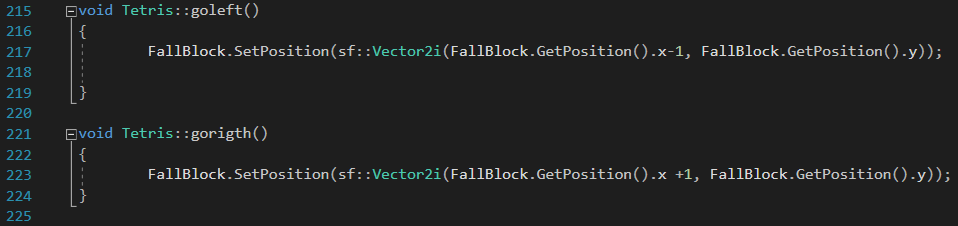
**

Zmiana szybkości następuje po przejściu na następny level (po ułożeniu linii): (*klasa Tetris.cpp)*



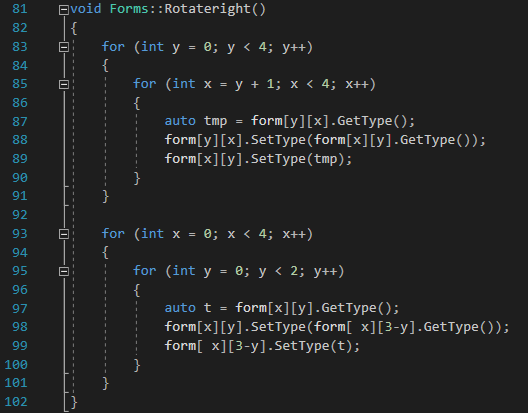
## *Gra powinna umożliwiać poruszanie klockiem prawo-lewo, przyspieszenie spadania, oraz obracanie klocka.*

Poruszanie się prawo-lewo: (*klasa Tetris.cpp)*

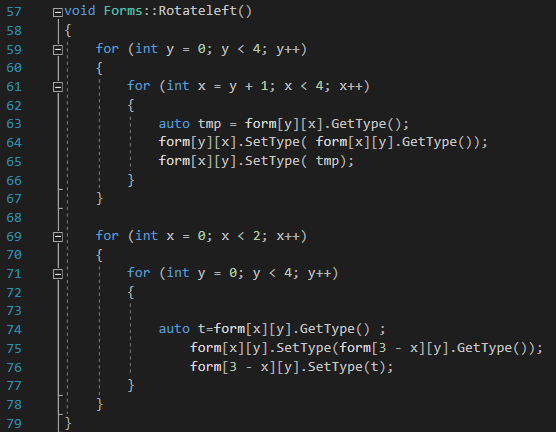


Obrót klocka jest wykonywany w punktu (2,2) w formie.

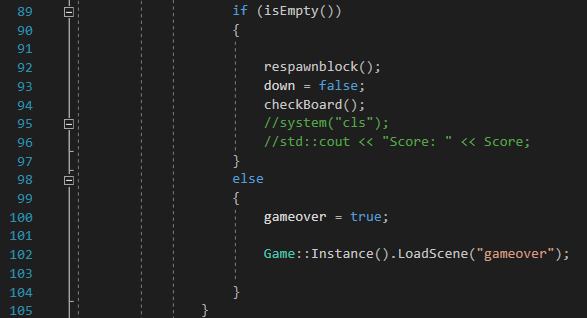
Obrót klocka w prawo: (*klasa Forms.cpp)*



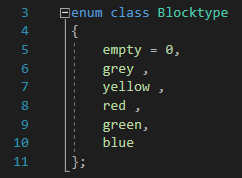
Obrót klocka w lewo: (*klasa Forms.cpp)*



## *Gra powinna umożliwiać przegranie.* (*klasa Tetris.cpp)*



## *Gra musi posiadać różne kolory klocków* (*klasa Block.h)*



# 

# Literatura:

<https://pl.wikipedia.org/wiki/C%2B%2B>

<https://pl.wikipedia.org/wiki/Singleton_(wzorzec_projektowy)>

<https://www.sfml-dev.org/documentation/2.0/>

<http://cpp0x.pl/kursy/Kurs-SFML-2-x-C++/460>

Język C++. Kompendium wiedzy, Stroustrup Bjarne

<https://www.youtube.com/watch?v=TSGOr_hhPtQ>

<https://www.youtube.com/watch?v=_9yem5dJt2E&list=PLRtjMdoYXLf7Lk_mJ0WiFNUJZOoDhIynr>