**Projekt: Tetris 3D**

**Opis gry:**

Gra będzie opierała się na mechanice znanej z popularnej gry Tetris, ale z tym wyjątkiem że będzie prezentowana w przestrzeni. Dostępna będzie możliwość dostosowania wielkości planszy jak i szybkość początkowej opuszczanych obiektów aczkolwiek razem ze wzrostem ilości punktów szybkość będzie stopniowo samoczynnie wzrastać. Obracanie planszy ma być możliwe dzięki myszy a obracanie i kierowanie spadającym obiektem klawiszami które będzie możliwość dopasowania w opcjach.

Podobna praca została wykonana przez Filip Rodek - Tetris3D (2017, OpenGL) aczkolwiek moja praca ma posiadać większą funkcjonalność zbliżoną prędzej do pracy prezentowanej w poniższym filmie

<https://www.youtube.com/watch?v=ZJj-9HqRpDc> (oczywiście bez obsługi gogli do obsługi Virtual Reality i z ubożoną grafiką)

**Planowane do użycia Biblioteki:**

<SFML/Window.hpp>

<SFML/Graphics.hpp>

<SFML/OpenGL.hpp>

<Gl/glu.h>

<iostream>

<cstdlib>

<ctime>

<vector>

<math.h>

<fstream>

<random>

**Wstępny zarys struktury programu:**

Najważniejszym obiektem będzie „sześcian” z kilku tych obiektów będzie utworzony obiekt „kształt”, kiedy „kształt” osiągnie określone miejsce jego wartości zostaną przekazane do obiektu „umieszczone”.

Klasa obiektu „sześcian” będzie zawierała metodę zajmującą się sprawdzaniem kolizji z obiektem „umieszczone” i będzie przechowywała informacje o położeniu w przestrzeni.

Klasa obiektu „kształt” będzie zawierała metodę która będzie zmieniała położenie obiektów „sześcian” w zależności od decyzji gracza., metodę przemieszczają w zależności od czasu obiekt w kierunku obiektu „umieszczone”, metodę zajmującą się ustala jednolitego koloru dla obiektów „sześcian”, metodę sprawdzającą czy któraś z metoda obiektów zwróciła kolizje i w tym przypadku zatrzymanie całego kształtu i przekazanie informacji do obiektu „umieszczone”.

Klasa obiektu „umieszczone” będzie zawierała metodę która będzie sprawdzała czy dana linia jest pełna a jeśli tak to ja usunąć i przenieść górną część i naliczyć punkty, metodę która sprawdzi czy obiekt uzyskał maksymalny pułap a jeśli tak to zakończyć grę.

Oczywiście zapewne będzie więcej klas zajmującymi się np. zapisem jak i odczytem ustawień czy rysowaniem linii pomocniczych ale uważam że są one mniej istotne i nie potrzebują głębszego opisu.

**Link do repozytorium w serwisie GitHub**

<https://github.com/Worki56/Gra_na_zaliczenie_PSiO_Wiktor_Rosinski>