

Dokumentacja do gry „Easy Tetris”

Języki zorientowane obiektowo

2016/2017

Marcin Jastrzębski II rok inf. inż. stac. Uniwersytet Łódzki



Spis treści

Opis i zasady gry	3
Budowa i schemat klas	4
Funkcja „main”	4
Klasa „Game”	5
Klasa „Engine”	5
Klasa „Obiekt”	5
Sterowanie	6
Widoki	6

Opis i zasady gry

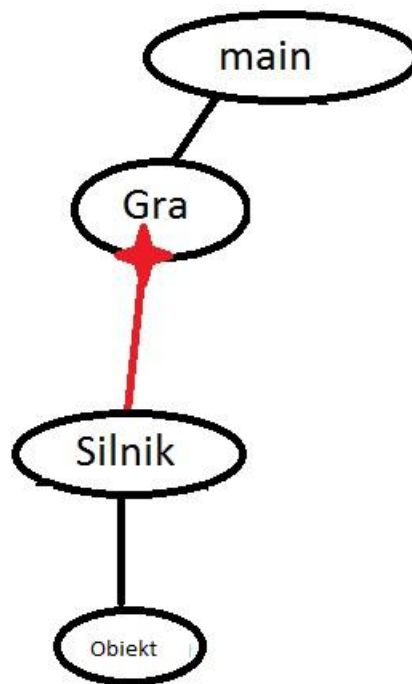
Gra „Easy Tetris” jest prostszą wersją oryginalnego tetrisa, ponieważ klocki nie spadają, a sami nimi sterujemy. Zasady są proste: Gracz układa klocki w taki sposób by tworzyły one linie poziome i generowały dodatkowe punkty dla gracza

Gra w wersji TXT. Wykonana w technologii C++. Program o architekturze 64 bitowej. Wymaga systemu Windows, klawiatury oraz monitora.

Autor programu: Marcin Jastrzębski II rok inf. Inż. stać. WFiS UŁ

Data: 2016/2017

Budowa i schemat klas



Legenda:

Pole/zmienna :

Dziedziczenie :  Po 

Klasa Gra dziedziczy po Silniku, wyposażonym w pola i metody zdolne obrabiać Klocki, Planszę i Gracza.

Funkcja „main”

```
1  #include <iostream>
2  #include <conio.h>
3  #include <time.h>
4  #include <windows.h>
5  #include "Game.h"
6  using namespace std;
7
8  int main()
9  {
10     Game m;           // tworzenie obiektu klasy Gra (Game)
11     m.Play();          // uruchomienie metody odpowiedzialnej za uruchomienie gry
12     return 0;
13 }
14
```

Klasa „Game”

```
1  #pragma once
2  #include "Engine.h"
3
4  class Game : public Engine
5  {
6  public:
7      void Play();           //metoda odpowiedzialna za start gry
8      Game();
9      ~Game();
10 }
```

Dziedziczy po Engine. Z indywidualnych metod posiada ona metodę: „void Play()” rozpoczyna ona nową grę. Obiektem tej klasy manipulujemy podczas całego programu.

Klasa „Engine”

Serce gry, wyposażony w metody:

```
void Board();           // tworzenie elementow planszy
void setScore(int _score); // ustawianie wyniku
int getScore();         // pobieranie wyniku
void setOpcja(int _opcja); // ustawianie numeru losowej figury
int getOpcja();         // pobieranie numeru losowej figury

//Ponizsze metody przyjmują argumenty:
//_x - pozycja srodkowego punktu figury na osi x
//_y - pozycja srodkowego punktu figury na osi y
//_rot - zmienna odpowiadajaca za rotacje figury
//      (0- 0 stopnii, 1 - figura obrocona o 90 stopnii,
//      2 - obrocona o 180 stopnii, 3 - o 270 stopnii)
void Init(int &_x, int &_y, int &_rot, int init);
void Init_L(int &_x, int &_y, int &_rot); //metoda wywołania figury w kształcie litery L
void Init_rL(int &_x, int &_y, int &_rot); //metoda wywołania figury w kształcie odwroconej litery L
void Init_B(int &_x, int &_y, int&_rot); //metoda wywołania figury w kształcie kwadratu
void Init_Z(int &_x, int &_y, int &_rot); //metoda wywołania figury w kształcie litery Z
void Init_rZ(int &_x, int &_y, int &_rot); //metoda wywołania figury w kształcie odwroconej litery Z
void Init_II(int &_x, int &_y, int &_rot); //metoda wywołania figury w kształcie litery I
void Init_Y(int &_x, int &_y, int &_rot); //metoda wywołania figury w kształcie litery Y
//-----

bool CheckEnd();        //metoda sprawdzająca czy przypadkiem gra sie nie skonczyla
bool Check(int _x = 0, int _y = 0, int m_gamma = 0); //sprawdzanie czy jakas figura dotyka innej

void Redraw();          // przerysowanie mapy

void CheckLineY();      // sprawdza czy nie ma pelnej linii klocek premiowanej
                        // dodatkowymi punktami

void Scene();           // rysowanie mapy
void Print();           // wypisanie wyniku zasad i sterowania
```

Klasa „Obiekt”

Klasa odpowiedzialna za stan pola obiektów

Spis metod oraz pól:

```

private:
    int m_x;           //zmienna odpowiadajaca punktow na osi x
    int m_y;           //zmienna odpowiadajaca punktow na osi y
public:
    void setX(int _x);  //metoda ustawiajaca wartosc punktu m_x
    void setY(int _y);  //metoda ustawiajaca wartosc punktu m_y
    int getX();         //metoda pobierajaca wartosc punktu m_x
    int getY();         //metoda pobierajaca wartosc punktu m_x
    Object();
    ~Object();
,

```

Sterowanie

E – Start gry
 A – ruch klocka w lewo
 W – ruch klocka w górę
 D – ruch klocka w prawo
 S – ruch klocka w dół
 SPACJA – na sam dół
 Q – rotacja klocka
 ENTER – zapisanie klocka w aktualnym miejscu na tablicy
 ESC – wyjście z gry

Widoki

Aktualnie gra wyposażona jest tylko w interfejs tekstowy.

C:\Users\Marcin\Desktop\GraZaliczeniowa\x64\Debug\Tetris.exe

