**MasterMind**

Wykonał: Kamil Król

Dokumentacja klas:

Klasa Board

Klasa służy głównie do rysowania interfejsu i obsługi kursora ( kolor, współrzędne)

|  |  |
| --- | --- |
| Metody publiczne | |
|  | **Board** (char &wybor) |
| Konstruktor z parametrem, przekazywany jest wybór z menu | |
| void | **drawUI** () |
| Rysuje ramkę | |
| void | **drawMenu** (char ch) |
| Rysuje menu główne, jako parametr znak ch, aby ustawić w odpowiednim miejscu wskaźnik  aktualnego wyboru | |
| char | **drawHowToPlay** () |
| Rysuje okno „Jak grać?” | |
| void | **gotoxy** (int x, int y) |
| Ustawia kursor w podanych współrzędnych | |
| void | **changeColor** (int color) |
| Zmienia kolor tekstu konsoli | |
| void | **drawTitle** () |

Rysuje napis „MasterMind”

|  |  |
| --- | --- |
| Atrybuty publiczne | |
| const unsigned char | **matchColorPlace** = 178 |
|  | |
| const unsigned char | **matchColor** = 176 |
|  | |
| const unsigned char | **sphere** = 219 |
|  | |
| const unsigned char | **indicator** = 248 |
|  | |
| const unsigned char | **menuIndicator** = 62 |
|  | |
| const unsigned char | **select** = 206 |
|  | |
| const unsigned char | **eraseSelect** = 205 |
|  | |
| const unsigned char | **space** = 32 |
| Przypisanie znaków do zmiennych  Atrybuty prywatne | |
| char | **choice** |

Zmienna przechowująca wybór menu głównego

Klasa Menu

Klasa służy do obsługi zdarzeń po wciśnięciu klawiszy nawigacyjnych w menu głównym. Dziedziczy z klasy Board.

|  |  |
| --- | --- |
| Metody publiczne | |
|  | **Menu** (char &wybor) |
| Konstruktor z parametrem, przekazywany jest wybór z menu | |
| void | **upChange** () |
| Odpowiada za zdarzenie po wciśnięciu strzałki w górę, czyli zmianę aktywnego wyboru i przesunięcie wskaźnika | |
| void | **downChange** () |
| Odpowiada za zdarzenie po wciśnięciu strzałki w dół, czyli zmianę aktywnego wyboru i przesunięcie wskaźnika | |
| char | **enterHit** () |

Odpowiada za zdarzenie po wciśnięciu klawisza Enter, czyli wyświetlenie odpowiedniego ekranu/uruchomienie funkcji w zależności od aktualnego wyboru

|  |  |
| --- | --- |
| Atrybuty prywatne | |
| char | **choice** |

Zmienna przechowująca wybór menu głównego

Klasa Game

Klasa służy do obsługi funkcji rozgrywki

|  |  |  |  |
| --- | --- | --- | --- |
| Metody publiczne | | | |
|  | **Game** (char &wybor) | | |
| Konstruktor z parametrem, przekazywany jest wybór z menu | | | |
| void | | | **setMaxRoundCount** (int rounds) |
| Ustawia wartość maksymalnej liczby rund | | | |
| void | | | **setColors** (int colors) |
| Ustawia liczbę dostępnych kolorów | | | |
| void | | | **setCodeLength** (int length) |
| Ustawia długość szukanego kodu | | | |
| void | | | **setRepeating** (bool rep) |
| Ustawia zmienną sygnalizującą możliwość powtarzania się wartości w kodzie | | | |
| void | | | **generate** () |
| Generuje kod do odgadnięcia | | | |
| char | | | **play** () |
| Główna funkcja; rysuje ekran gry, obsługuje nawigację i sprawdza stan gry, zwraca wybór dokonany w funkcji ifWin()/ifLost() | | | |
| void | | | **upChange** () |
| Odpowiada za zdarzenie po wciśnięciu strzałki w górę, czyli zmianę koloru aktywnej kuli | | | |
| void | | | **downChange** () |
| Odpowiada za zdarzenie po wciśnięciu strzałki w dół, czyli zmianę koloru aktywnej kuli | | | |
| void | | | **leftChange** () |
| Odpowiada za zdarzenie po wciśnięciu strzałki w lewo, czyli zmianę aktywnej kuli | | | |
| void | | | **rightChange** () |
| Odpowiada za zdarzenie po wciśnięciu strzałki w prawo, czyli zmianę aktywnej kuli | | | |
| void | | | **enterHit** () |
| Odpowiada za zdarzenie po wciśnięciu klawisza Enter, czyli sekwencję funkcji po zatwierdzeniu kodu ( zapis, porównanie, wypisanie podpowiedzi, etc.) | | | |
| void | | | **clearIndicator** () |
| Wymazuje poprzedni indykator aktywnej kuli | | | |
| void | | | **saveCode** () |
| Zapisuje wybrany kod do tablicy przechowującej aktualny kod gracza | | | |
| void | | | **compare** () |
| Porównuje wygenerowany kod z kodem gracza, ustawia liczbę i rodzaj podpowiedzi | | | |
| char | | | **ifWin** () |
| Rysuje ekran po wygranej, czeka na akcje gracza i zwraca wybór do menu | | | |
| char | | | **ifLost** () |
| Rysuje ekran po przegranej, czeka na akcje gracza i zwraca wybór do menu | | | |
| void | | | **wipeData** () |
| Przywraca wartości zmiennych do domyślnych przed nową grą | | | |
| void | | | **drawColors** () |
| Rysuje podgląd dostępnych kolorów | | | |
| [**Game**](file:///H:\MasterMindKrol\html\class_game.html) & | | | **operator=** (const [**Game**](file:///H:\MasterMindKrol\html\class_game.html) &wzor) |

Operator przypisania dla klasy

|  |  |
| --- | --- |
| int | **getMaxRoundCount** () |
| Zwraca wartość maksymalnej liczby rund | |
| int | **getColors** () |
| Zwraca liczbę dostępnych kolorów | |
| int | **getCodeLength** () |
| Zwraca długość szukanego kodu | |
| bool | **getRepeating** () |
| Zwraca zmienną sygnalizującą możliwość powtarzania się wartości w kodzie | |

|  |  |
| --- | --- |
| Atrybuty prywatne | |
| char | **choice** |

Zmienna zapamiętująca wybór gracza

|  |  |
| --- | --- |
|  | |
| int | **codeLength** |
| Zmienna przechowująca długość szukanego kodu: domyślnie 4 | |
| Int\* | **generatedCode** |
| Wskaźnik do tablicy przechowującej szukany kod | |
| Int\* | **playerCode** |
| Wskaźnik do tablicy przechowującej aktualny kod wpisany przez gracza | |
| int | **first** |
| Kod pierwszej kuli | |
| int | **Second** |

Kod drugiej kuli

|  |  |
| --- | --- |
|  | |
| int | **third** |
| Kod trzeciej kuli | |
| int | **fourth** |
| Kod czwartej kuli | |
| int | **fifth** |
| Kod piątej kuli | |
| int | **sixth** |
| Kod szóstej kuli | |
| bool | **isWin** |

Zmienna przechowująca informacje o wygranej/przegranej

|  |  |
| --- | --- |
| int | **roundCount** |
| Licznik aktualnej rundy | |
| int | **mover** |

Wartość przesunięcia indykatora aktywnej kuli

|  |  |
| --- | --- |
|  | |
| int | **wholeMatchesCount** |
| Licznik ile jest kul w dobrym kolorze na dobrym miejscu | |
| int | **colorMatchesCount** |
| Licznik trafionych kolorów | |
| int | **colorCount** |
| Zmienna przechowująca liczbę możliwych kolorów | |
| int | **maxRoundCount** |
| Zmienna przechowująca maksymalną liczbę rund w grze | |
| bool | **repeating** |
| Zmienna przechowująca informację o możliwości powtarzania się kolorów w kodzie | |

Klasa Options

Klasa służy do obsługi menu wyboru opcji

|  |  |
| --- | --- |
| Metody publiczne | |
|  | **Options** (char &wybor, [**Game**](file:///H:\MasterMindKrol\html\class_game.html) &gra) |
| Konstruktor z parametrami, przekazywany jest wybór z menu I aktualna gra | |
| [**Game**](file:///H:\MasterMindKrol\html\class_game.html) | **changes** () |
| Funkcja główna, odpowiada za obsługę nawigacji po menu i wyzwalanie akcji | |
| void | **setState** (int defaultState) |
| Ustawia stan menu opcji | |
| void | **upChange** () |
| Odpowiada za akcję po wciśnięciu strzałki w górę, czyli zmianę aktywnego parametru i przesunięcie wskaźnika aktywnego parametru | |
| void | **downChange** () |
| Odpowiada za akcję po wciśnięciu strzałki w dół, czyli zmianę aktywnego parametru i przesunięcie wskaźnika aktywnego parametru | |
| void | **leftChange** () |
| Odpowiada za akcję po wciśnięciu strzałki w lewo, czyli zmianę wartości aktywnej opcji i przesunięcie suwaka | |
| void | **rightChange** () |
| Odpowiada za akcję po wciśnięciu strzałki w prawo, czyli zmianę wartości aktywnej opcji i przesunięcie suwaka | |
| char | **enterHit** () |
| Odpowiada za akcję po wciśnięciu klawisza Enter, czyli zatwierdzenie wyborów | |
| void | **save** () |
| Zapisuje ustawione parametry z obiekcie klasy Game | |
| void | **drawOptions** () |

Rysuje menu opcji

main.cpp

Główna część programu, odpowiada za inicjalizację obiektów i zmiennych, obsługę menu głównego.

|  |  |
| --- | --- |
| int | **main()** |
| Funkcja główna, inicjalizuje zmienne i obsługuje nawigację oraz akcje menu głównego | |