Dokumentacja Projektu   
„Problem liszaja”

Wiktor Mazur, 173669

2EF-DI

**Cel projektu**

Celem projektu jest przeprowadzenie symulacji zarażania komórek, napisanej w języku programowania C++.

**Technologie**

Język programowania: **C++ (Standard ISO C++ 14)**

Biblioteka graficzna: **SFML v2.6.1**

Środowisko programistyczne: **Visual Studio 2022 v143**

Wersja Zestawu Windows SDK: **10.0**

**Klasa Board**

**Opis**

Klasa Board służy do wyświetlania tablicy na której symulacja oraz logika symulacji.

**Pola**

unsigned int size; - zmienna obsługująca wielkość planszy

float cellSize; - zmienna obsługująca wielkość komórki

float boardSize = 550.0f; - zmienna obsługująca wielkość planszy. Jest ona stała

float offsetX; - zmienna obsługująca wielkość miejsca po prawej jak i po lewej stronie programu

float offsetY; - zmienna obsługująca wielkość miejsca u góry jak i u dołu programu

sf::Font font; - Objekt klasy Font. Obsługuje czcionkę

vector<vector<HealthStatus>> healthStatuses; - Wektor statusu komórki

vector<vector<sf::Color>> colors; - Wektor koloru komórki

sf::Text title; - Objekt klasy tekst biblioteki SFML.

sf::Text roundText; - Objekt klasy tekst biblioteki SFML.

sf::Text deltaTimeText; - Objekt klasy tekst biblioteki SFML.

sf::Text allTimeText; - Objekt klasy tekst biblioteki SFML.

sf::Text allCellText; - Objekt klasy tekst biblioteki SFML.

sf::Text infectedCellText; - Objekt klasy tekst biblioteki SFML.

sf::Text immuneCellText; - Objekt klasy tekst biblioteki SFML.

sf::Text healthCellText; - Objekt klasy tekst biblioteki SFML.

vector<tuple<int, int, int>> toStore; - Wektor zawierający informacje o komórkach zarażonych i chorych.

**Metody**

int countCells(HealthStatus status, int boardSize); - Metoda zliczająca komórki

void updateSize(int newSize); - Metoda która aktualizuje status komórek

void drawBoard(sf::RenderWindow& window); - Metoda która rysuje plansze

void calculateboardSize(sf::RenderWindow& window); - Metoda która oblicza wielkość planszy

void handleClick(int currentround, sf::RenderWindow& window); - Metoda która pozwala na zarażenie komórek poprzez klikanie na plansze

void update(int boardSize, int currentRound, float deltaTime, float allTime, sf::RenderWindow& window, int infectionPercent, int infectedToImmune, int immuneCooldown); - Metoda która aktualizuje stan komórki

void findRowAndCol(unsigned int row, unsigned int col, int currentround, float infectionPercent); - Metoda która zaraża komórki na około komórki chorej.

void addTotoStore(int newRow, int newCol, int currentround); - Metoda która dodaje komórki zarażone do listy.

bool isInToStore(int currentround, int newRow, int newCol); - Metoda która sprawdza czy komórka jest zarażona

void drawtoStore(vector<tuple<int, int, int>>& toStore); - Metoda która wypisuje dodane komórki do listy

void spreadInfection(vector<tuple<int, int, int>>& toStore, int currentround, int infectedToImmune, int immuneCooldown); - Metoda która obsługuje rozchodzenie się infekcji

void removeHealthCells(vector<tuple<int, int, int>>& toStore, int currentround, int infectedToImmune, int immuneCooldown); - Metoda która usuwa zdrowe komórki z Listy zarażonych/odpornych

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

**Klasa Menu**

**Opis**

Klasa Menu obsługuje Menu programu w którym wywoływane są podstawowe parametry programu.

**Pola**

int menuSize=25; - zmienna obsługująca wielkość planszy

int menuMaxround = 1000; - zmienna obsługująca ilość rund

float menuHoldprocess = 0.9; - zmienna obsługująca czas wstrzymania wątku

int menuInfectionPercent = 50; - zmienna obsługująca procent szans zarażenia

int menuInfectedToImmune = 1; - zmienna obsługująca czas zarażenia komórki

int menuImmuneCooldown = 1; - zmienna obsługująca czas odporności komórki

sf::Font font; - Objekt klasy tekst biblioteki SFML.

sf::Text title; - Objekt klasy tekst biblioteki SFML.

sf::Text sizeText; - Objekt klasy tekst biblioteki SFML.

sf::Text buttonText; - Objekt klasy tekst biblioteki SFML.

sf::Text holdprocessText; - Objekt klasy tekst biblioteki SFML.

sf::Text infectionPercentText; - Objekt klasy tekst biblioteki SFML.

sf::Text infectedToImmuneText; - Objekt klasy tekst biblioteki SFML.

sf::Text immuneCooldownText; - Objekt klasy tekst biblioteki SFML.

sf::Text exit; - Objekt klasy tekst biblioteki SFML.

Board\* board; - Objekt klasy Board

Duration\* duration; - Objekt klasy Duration

**Metody**

void drawMenu(sf::RenderWindow& window); - metoda która rysuje menu

void drawButton(sf::RenderWindow& window, const std::string& text, const sf::Vector2f& position, const std::function<void()>& onClick, const sf::Color& mainColor, const sf::Color& hoverColor); - metoda która rysuje przycisk

void increase(int howMuch, const std::string& variable); - metoda która zwiększa ilość danej zmiennej

void decrease(int howMuch, const std::string& variable); - metoda która zmniejsza ilość danej zmiennej

**Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie**

**Klasa Duration**

**Opis**

Klasa ta obsłużą czas występujący w grze.

**Pola**int maxround; - zmienna obsługująca ilość rund

unsigned int holdprocess; - zmienna obsługująca czas wstrzymania wątku

int infectionPercent; - zmienna obsługująca procent szans zarażenia

unsigned int infectedToImmune; - zmienna obsługująca czas zarażenia komórki

unsigned int immuneCooldown; - zmienna obsługująca czas odporności komórki

**Metody**

Duration(int maxround, unsigned int holdprocess, int infectionPercent, unsigned int infectedToImmune, unsigned int immuneCooldown); - Konstruktor ustawiający zmienne w danej klasie

**Obraz zawierający tekst, Czcionka, linia, zrzut ekranu

Opis wygenerowany automatycznie**

**Klasa Game**

**Opis**

Klasa Game obsługuje uruchamianie projektu oraz podstawowe parametry z którymi gra jest uruchamiana.

**Pola**

Duration\* duration; - Objekt klasy Duration

Board\* board; - Objekt klasy Board

Menu\* menu; - Objekt klasy Menu

sf::Clock clock; - Objekt klasy Clock

float deltaTime; - zmienna obsługująca czas

float allTime; - zmienna obsługująca całkowity czas trwania programus

**Metody**

int run(); - metoda która uruchamia program

void delay(int time, sf::RenderWindow& window, bool& menuOpen); - metoda która wstrzymuje działanie programu

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie