МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

“Московский государственный университет геодезии и картографии”

(МИИГАИК)

Факультет геоинформатики и информационной безопасности

Кафедра геоинформационных систем и технологий

**Лабораторная работа №6**

**“** **Разработка графического приложения с использованием циклов”**

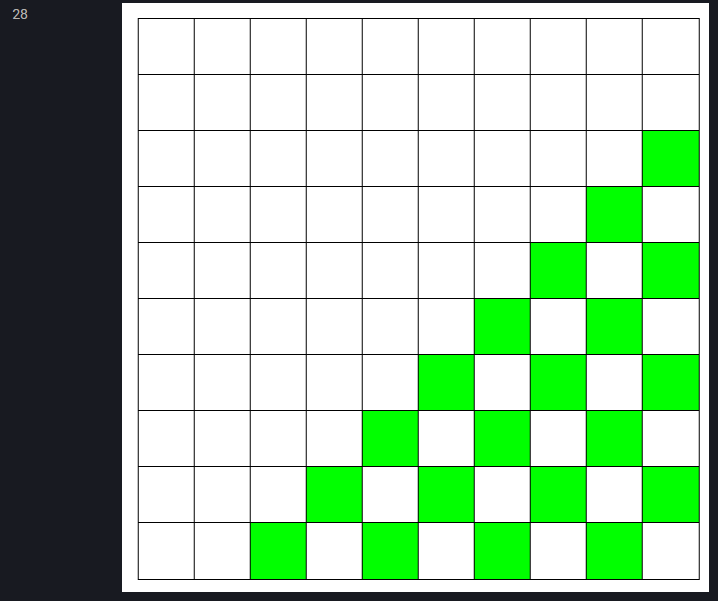
Проверил: Выполнил:

Лебедев Евгений Денисович Студент группы: 2024-ФГИИБ-ПИ-1б

Центнер Валерий Викторович

Москва 2024

Мой номер задания 28.

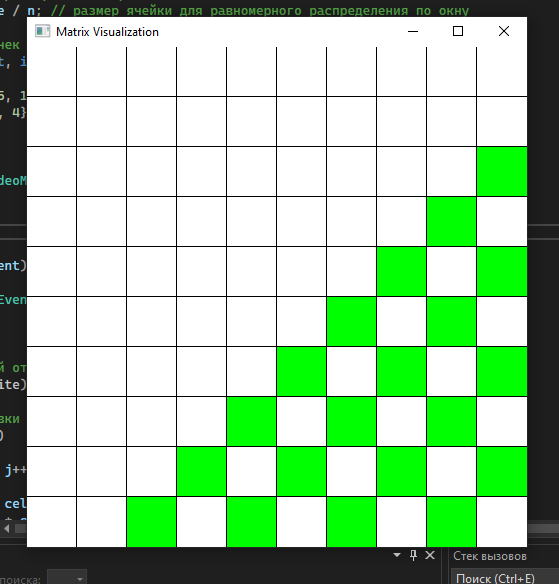


Ссылка на GitHub.

<https://github.com/guguker/inf_university>

Зайдите пожалуйста в папку “laboratory work №6”, прямая ссылка с кучей лишних символов очень много места здесь занимает))

Скриншот реализации работающего интерфейса.



Листинг 1.

Фрагмент кода приложения:

#include <SFML/Graphics.hpp>

#include <iostream>

int main() {

// Размер окна и матрицы

const int windowSize = 500;

const int n = 10; // размерность матрицы (10x10)

const int cellSize = windowSize / n; // размер ячейки для равномерного распределения по окну

// Создание окна

sf::RenderWindow window(sf::VideoMode(windowSize, windowSize), "Matrix Visualization");

while (window.isOpen()) {

sf::Event event;

while (window.pollEvent(event)) {

if (event.type == sf::Event::Closed)

window.close();

}

// Очистка окна перед новой отрисовкой

window.clear(sf::Color::White);

// Двойной цикл для отрисовки каждой ячейки в сетке n x n

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

sf::RectangleShape cell(sf::Vector2f(cellSize, cellSize));

cell.setPosition(j \* cellSize, i \* cellSize);

// Устанавливаем цвет рамки и заливки

cell.setOutlineThickness(1);

cell.setOutlineColor(sf::Color::Black);

// Логика закрашивания диагоналей

int diagonalIndex = i + j - (n - 1); // Индекс параллельной диагонали относительно побочной

if (diagonalIndex > 0 && diagonalIndex % 2 == 0) { // Только для чётных индексов ниже побочной

cell.setFillColor(sf::Color::Green); // Закрашиваем зелёным

}

else {

cell.setFillColor(sf::Color::White); // Остальные клетки остаются белыми

}

// Отрисовка ячейки

window.draw(cell);

}

}

// Отображение результата

window.display();

}

return 0;

}

Ход выполнения:

1. **Анализ задачи:**

Задача состоит в закрашивании диагоналей, параллельных побочной диагонали, которые находятся ниже её. Закрашивание производится через одну: первая диагональ под побочной остаётся незакрашенной, следующая закрашивается, и далее чередование продолжается.

1. **Определение алгоритма:**

Используется двойной цикл for для перебора строк (i) и столбцов (j) матрицы:

Внешний цикл i проходит по строкам.

Внутренний цикл j проходит по столбцам.

Для определения принадлежности клетки к параллельной диагонали ниже побочной используется формула:

diagonalIndex = i + j - (n - 1);

Где - diagonalIndex — смещение текущей диагонали относительно побочной.

Если diagonalIndex > 0, клетка находится ниже побочной диагонали.

Чередование закрашивания осуществляется по правилу:

Если diagonalIndex % 2 == 0, клетка закрашивается зелёным цветом.

1. **Методические указания по определению диагоналей**

Побочная диагональ проходит через элементы, для которых сумма индексов строки и столбца равна :

i + j = n - 1;

Для определения диагоналей ниже побочной используется формула:

diagonalIndex = i + j - (n - 1);

Где:

diagonalIndex > 0 — диагонали ниже побочной.

Чередование осуществляется с использованием условия:

diagonalIndex % 2 == 0;

1. **Вывод**

В результате выполнения лабораторной работы была создана программа, которая:

Отображает сетку размером 10x10.

Закрашивает диагонали, параллельные побочной, начиная с первой ниже побочной, через одну.

Реализует логику закрашивания с использованием вложенных циклов и условного оператора.

Программа корректно выполняет поставленную задачу и наглядно демонстрирует работу с диагоналями матрицы.