Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

направление подготовки: 09.03.01 - «Информатика и вычислительная техника»

**О Т Ч Е Т**

**по творческой работе**

**по дисциплине**

**«Основы алгоритмизации и программирования» семестр 2**

Выполнил студент гр. ИВТ-21-1б

Ипатов Дмитрий Сергеевич

Проверил:

(оценка) (подпись)

г. Пермь-2022

**Калькулятор:**

**Постановка задачи:**

1. Реализовать калькулятор системы с тремя неизвестными переменными;
2. Оформить калькулятор (Фон, подобрать шрифты и цветовую палитру);
3. Проверить правильность выполнения действий.

**Описание решения:**

Создание калькулятора выполнялось с помощью инструмента Windows Forms.

При работе над калькулятором пришлось пройти обучение по инструменту WinForms, трудности возникли при вводе знака “-“ в поля для ввода коэффициентов (Минус должен был быть только один и только в начале), данная проблема была преодолена. Также много времени ушло на вывод решения, так как каждое отдельное число и знак в нем – это отдельный label.

В своем калькуляторе я горжусь тем, что пользователь может посмотреть не только результат, но и как этот результат был получен, то есть в калькуляторе есть подробное решение СЛАУ. Также хочу отметить оформление калькулятора, в нем нет ничего лишнего, цвета подобраны хорошо и глазу приятно на это смотреть.

**Скриншоты калькулятора:**

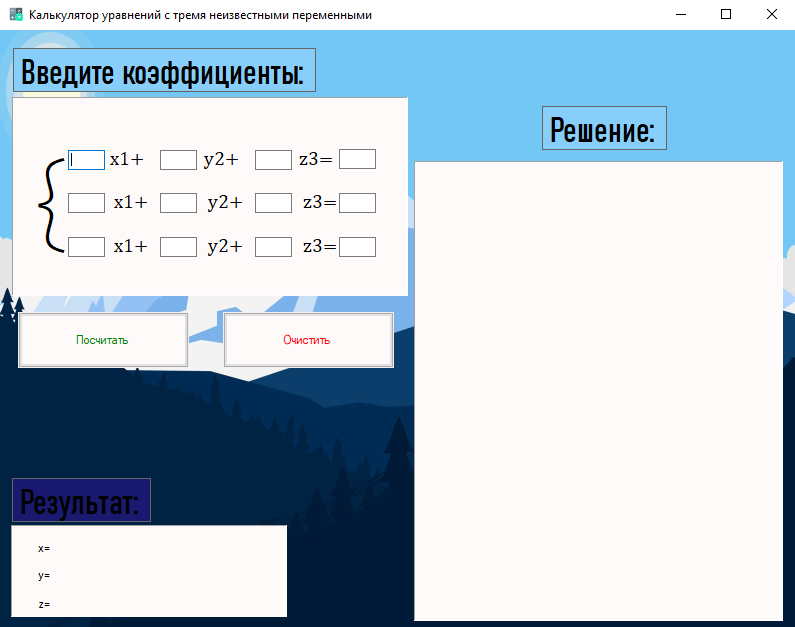
****

Рис 1. Интерфейс калькулятора при запуске

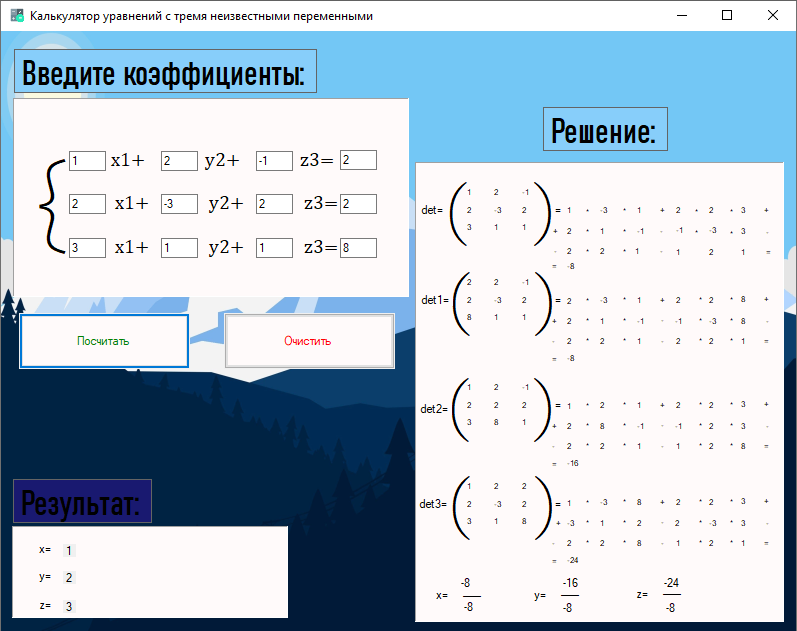
****

Рис 2. Калькулятор после выполнения решения

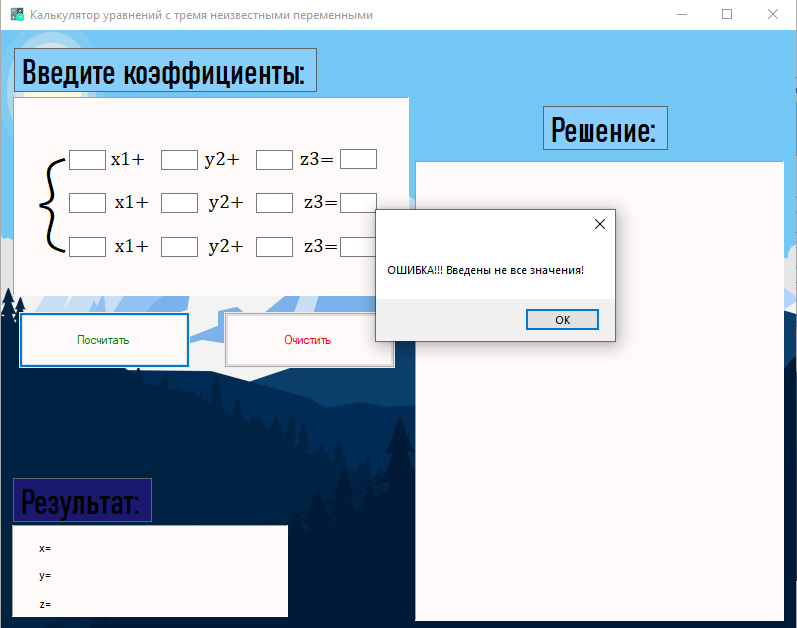
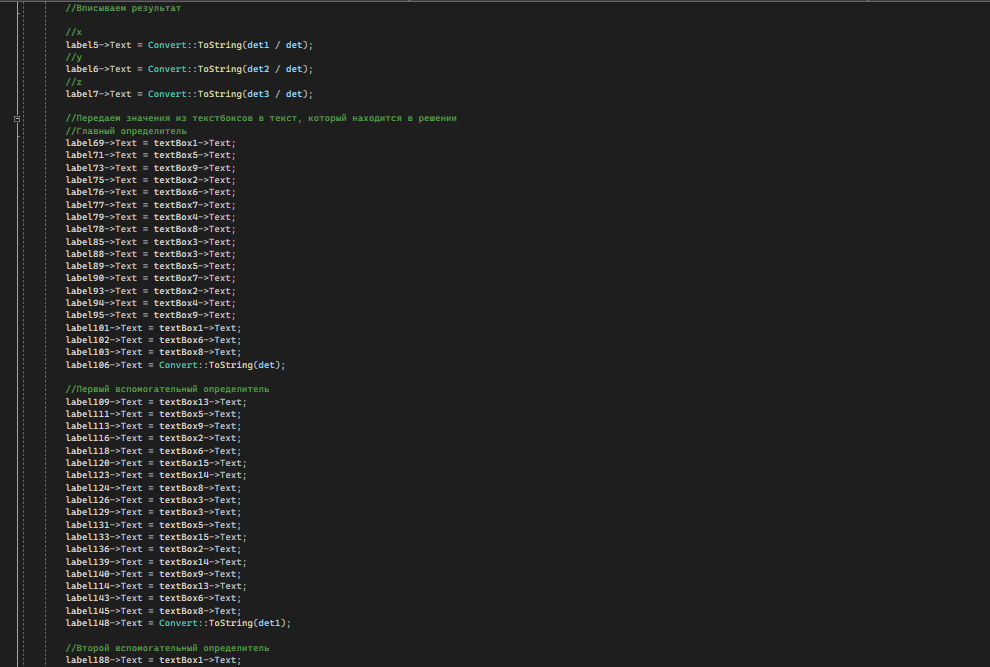
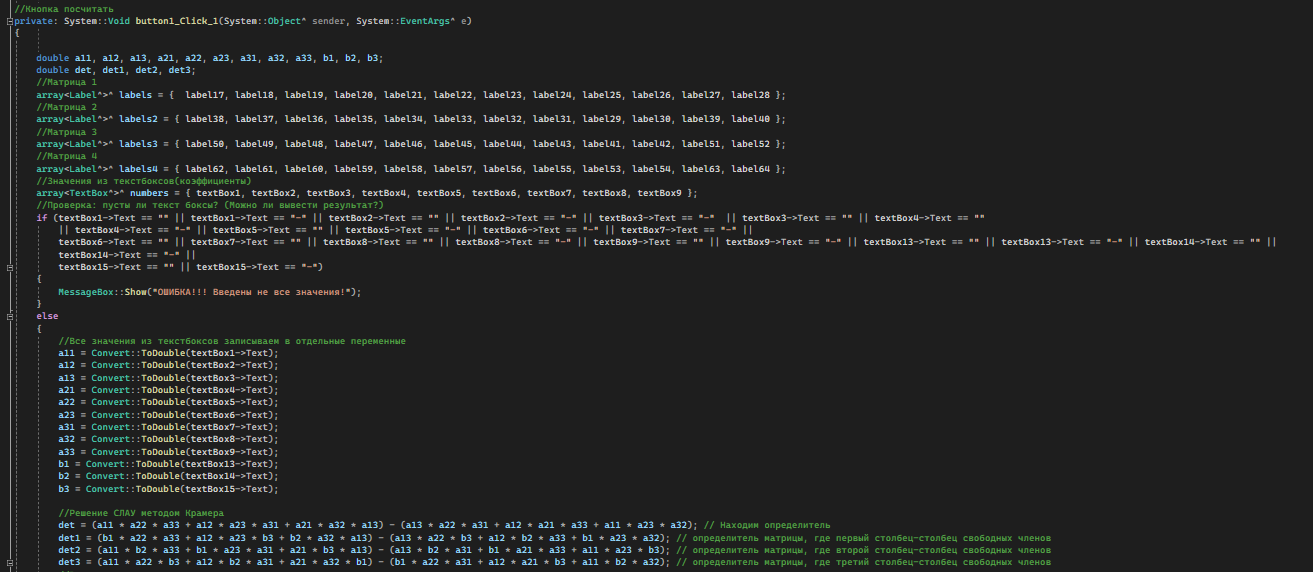
****

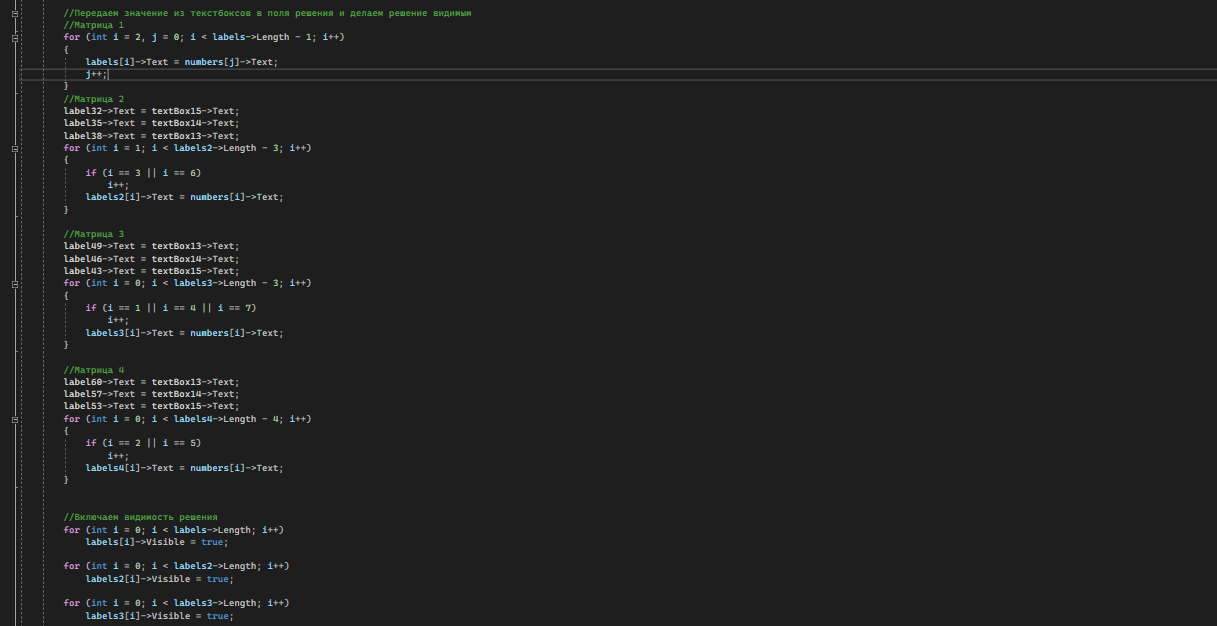
Рис 3. Калькулятор после нажатия кнопки “Посчитать” без ввода коэффициентов

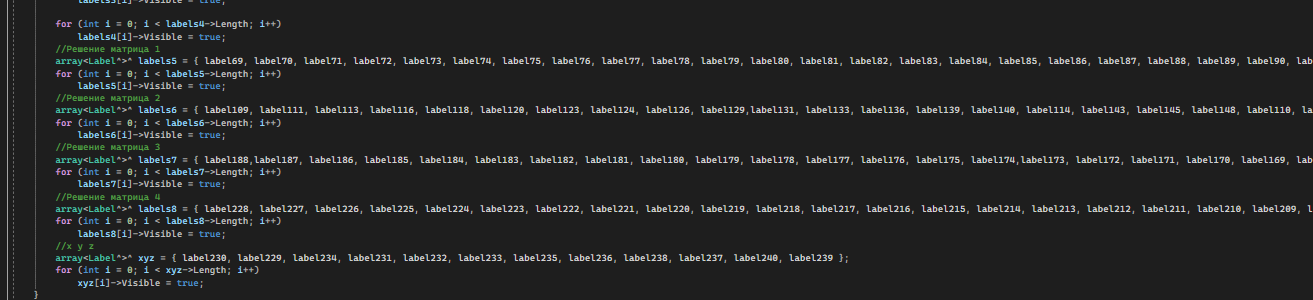
**Описание кода:**

1. Кнопка посчитать.

****



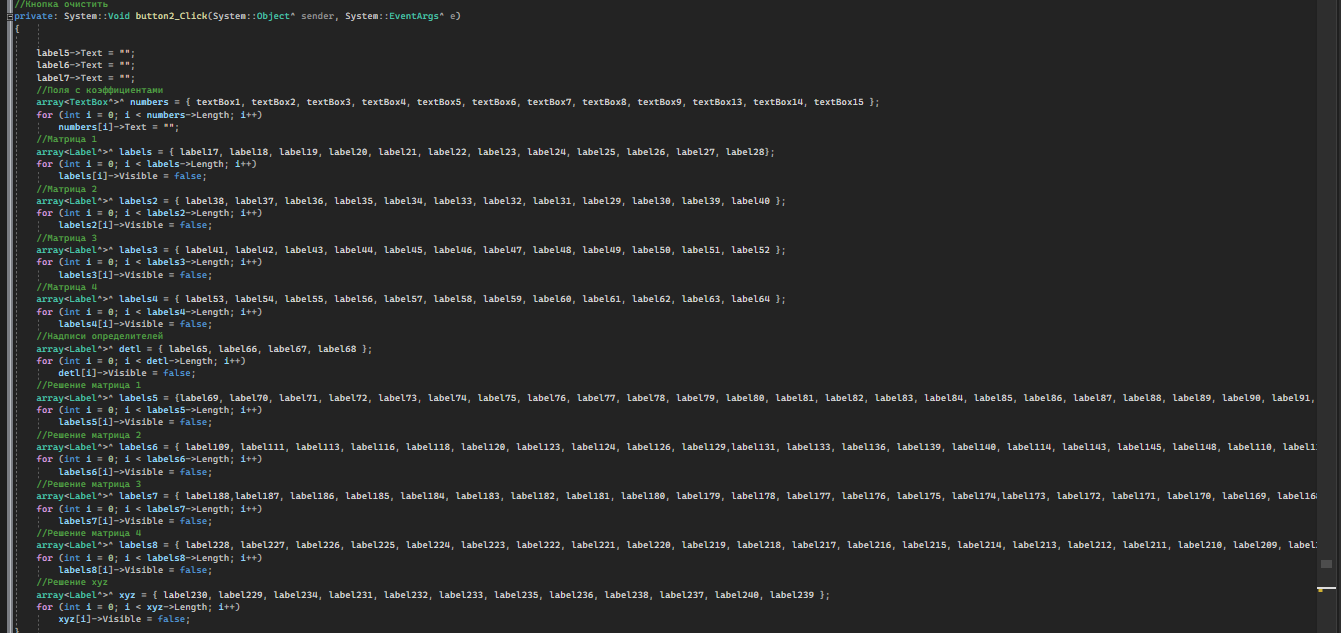
****

****

При нажатии на кнопку посчитать у нас инициализируются переменные типа double a11, a12, a13, a21, …, b1, b2, b3. В них позже запишутся значения из текст боксов, в которые вписываются коэффициенты уравнения. Также инициализируем переменные типа double под определители матриц. Создаем массивы с лейблами, которые потом прогоним через цикл for, и сделаем текст лейблов видимым. Прописываем проверку на пустоту текст боксов или если там стоит один знак “-“, если там действительно пусто, то выводим на экран окно, в котором будет прописана ошибка “ОШИБКА!!! Введены не все значения”.

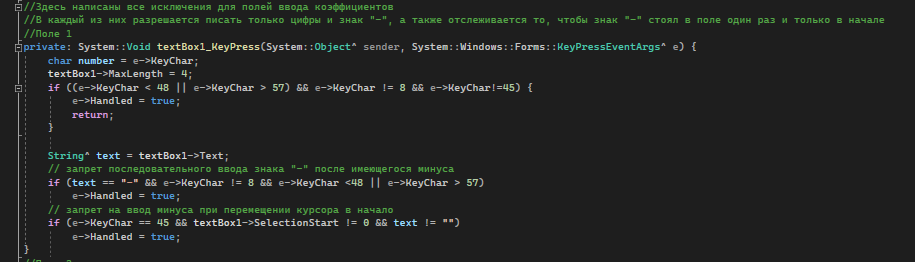
Если же текст боксы не пусты, начинается выполнение решения: в переменные a11, …, b3 вписываются значения из текст боксов, система решается методом Крамера, поэтому ищем главный и вспомогательные определители. Переменная x находится путем деления первого вспомогательного определителя на главный, переменные y, z также только с соответствующими вторым и третьим вспомогательными определителями. Далее в лейблы решения вписываются значения из текст боксов. Следующий этап — это отображение этих лейблов, то есть переключение их видимости на true.

1. Кнопка очистить.



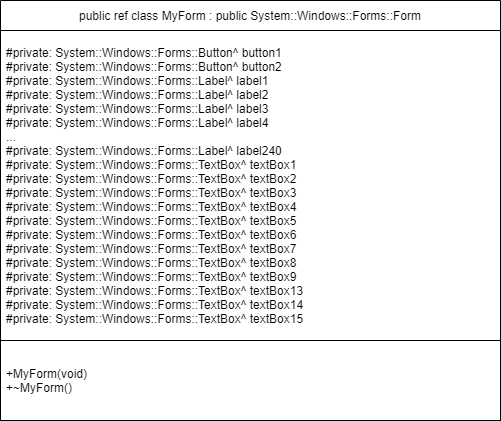
При нажатии на кнопку очистить, текст всех полей для ввода коэффициентов меняется на пустоту, то есть очищается. Видимость всех лейблов меняется на false.

1. Текст боксы (они же поля для ввода коэффициентов)



Здесь прописаны условия, которые позволяют нам вводить в текст боксы только цифры и знак “-“, при том знак “-“ может быть только один и ставиться в начале текст бокса.

**UML диаграмма:**

****

**Коммивояжер:**

**Постановка задачи:**

1. Реализовать вывод матрицы смежности и всех решений;
2. Реализовать функции для решения задачи методом ветвей и границ;
3. Реализовать нахождение наикратчайшего пути;
4. Вывести путь и расстояние, которое по нему было пройдено;
5. Реализовать отрисовку графа с помощью SFML.

**Описание решения:**

Создание программы осуществлялось на основе метода ветвей и границ, для отрисовки графа был использован SFML, в ходе работы пришлось изучить инструмент SFML для отрисовки графа. Трудностей не возникало.

В задаче Коммивояжера я горжусь понятной визуализацией графа, отображения пути и результата, которые были посчитаны и выведены верно.

**Скриншоты задачи с Коммивояжером:**

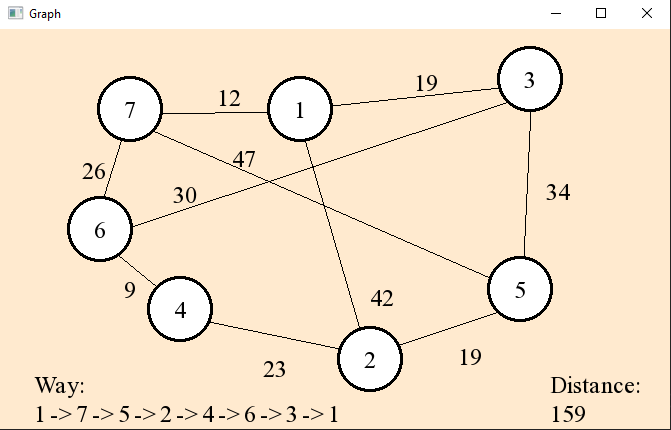


Рис 4. Визуализация графа через SFML

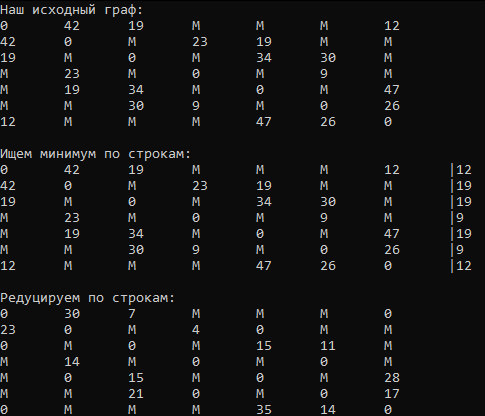


Рис 5. Вывод решения в консоль

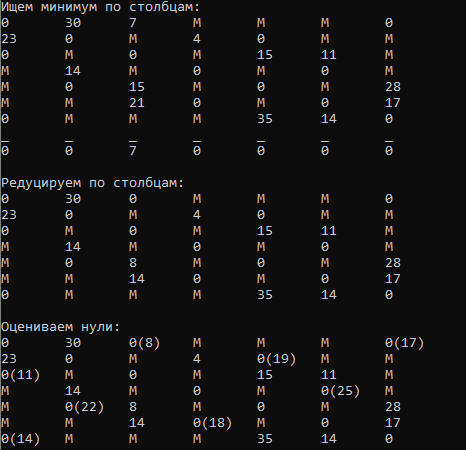


Рис 6. Вывод решения в консоль

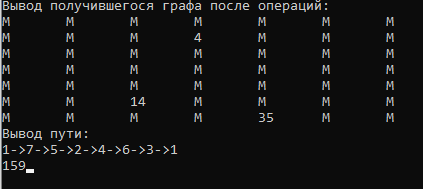
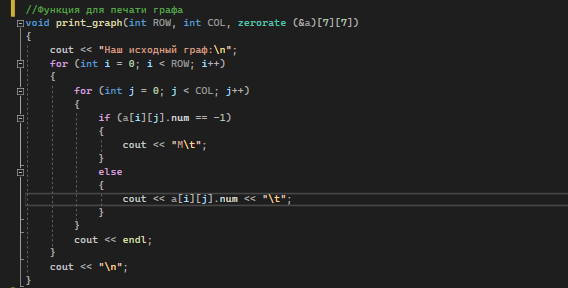


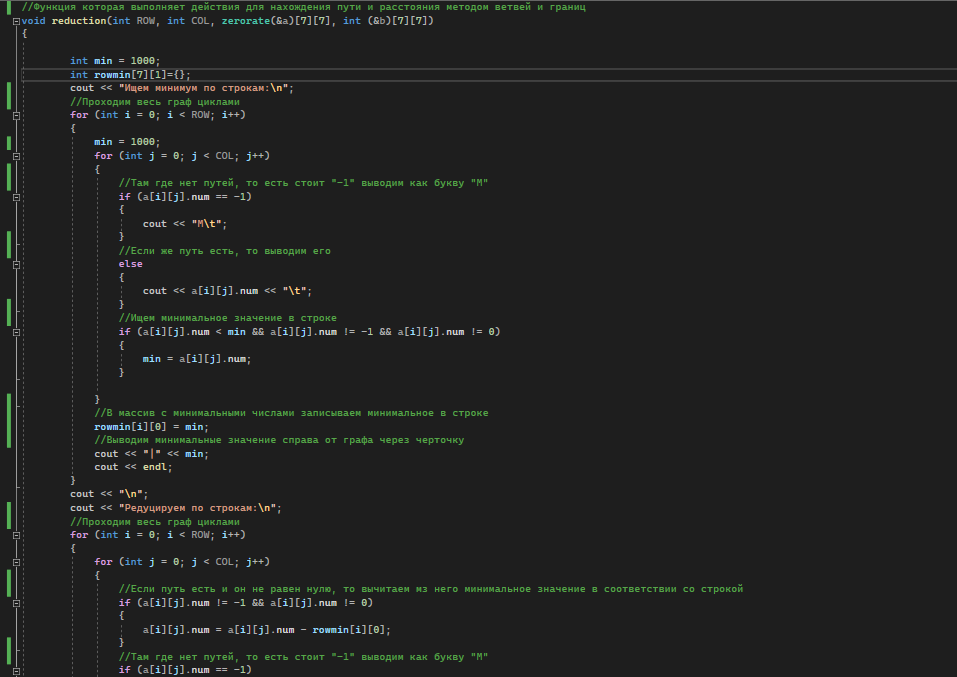
Рис 7. Вывод решения в консоль

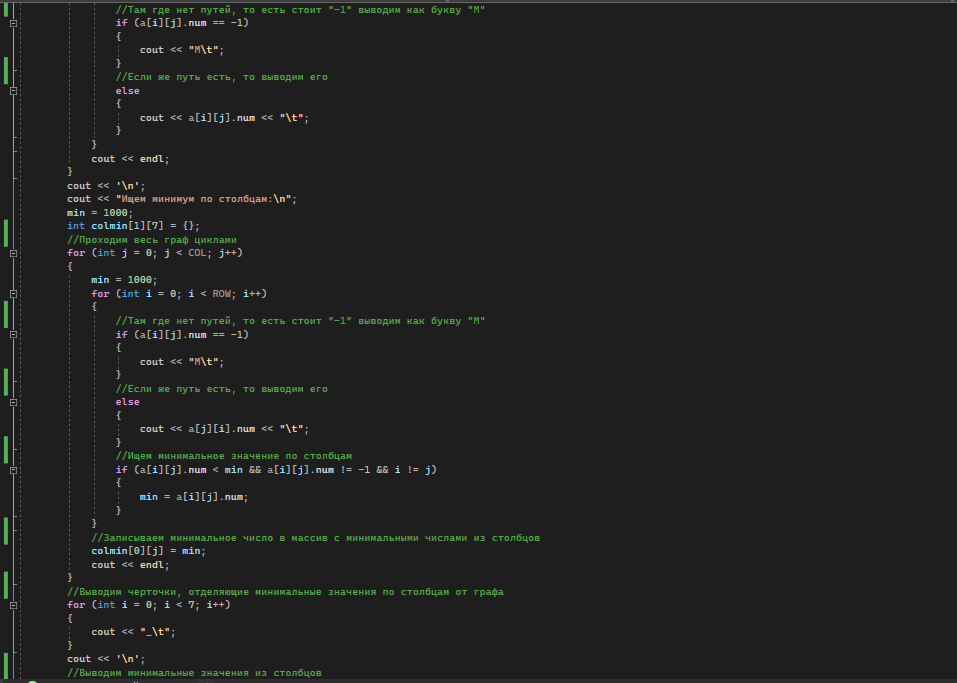
**Описание кода:**

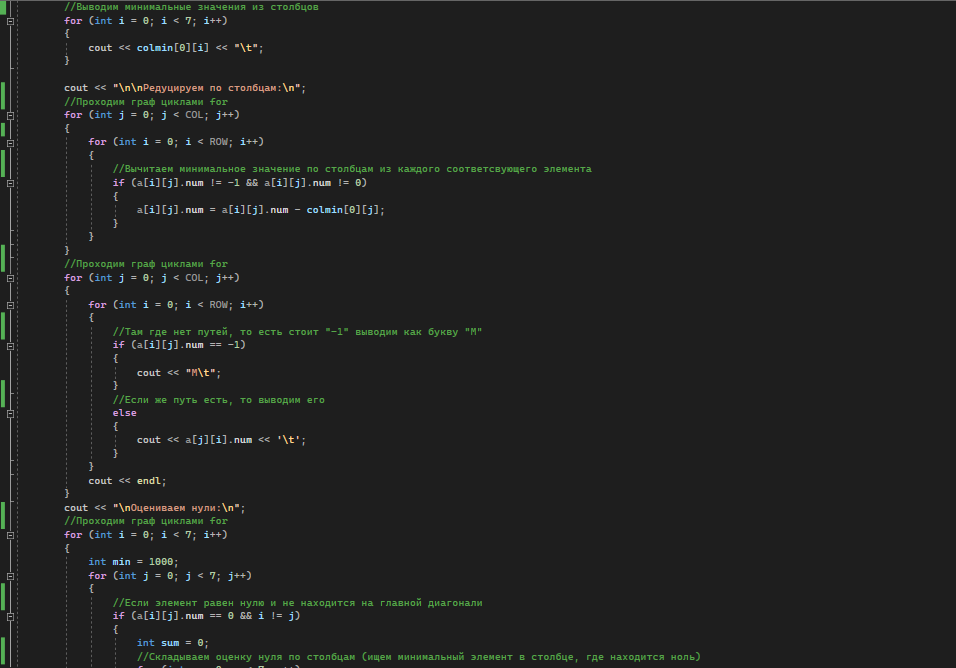
1. Вывод графа

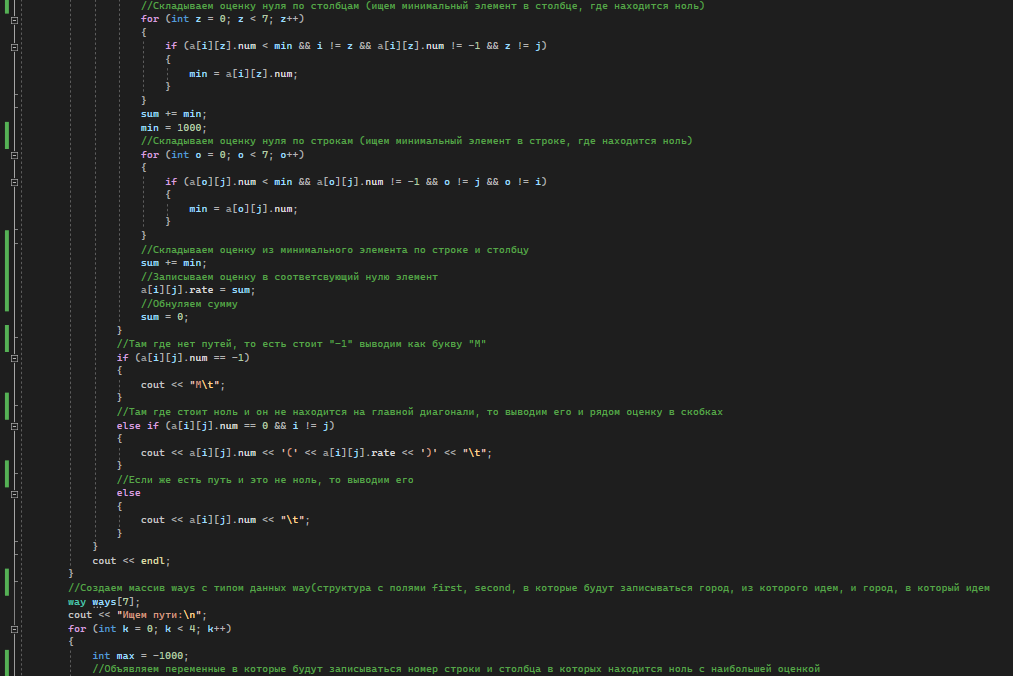
****

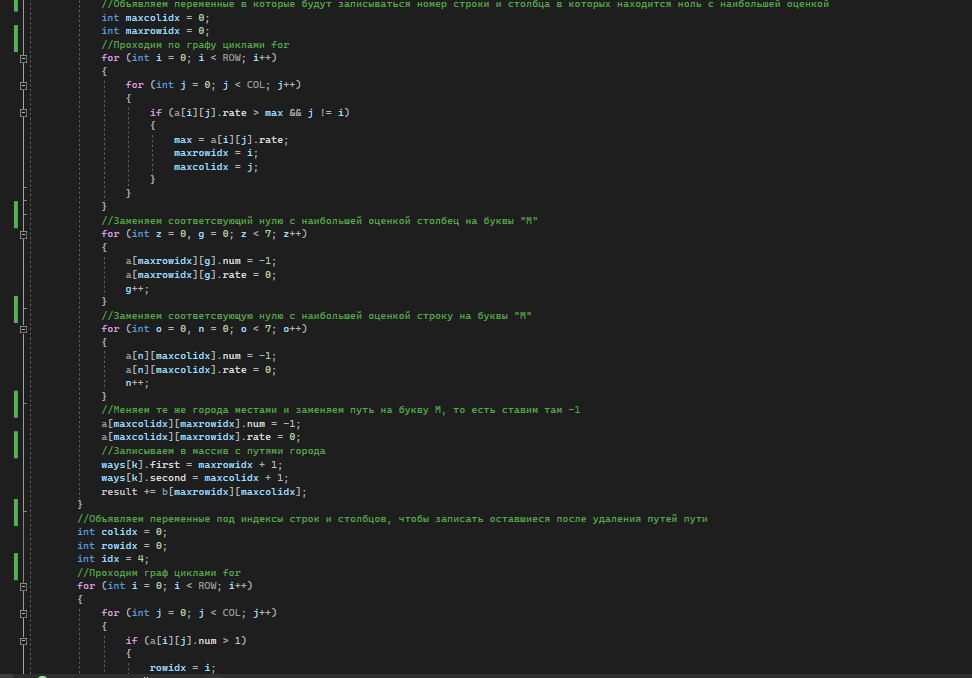
1. Выполнение метода ветвей и границ для поиска пути и расстояния

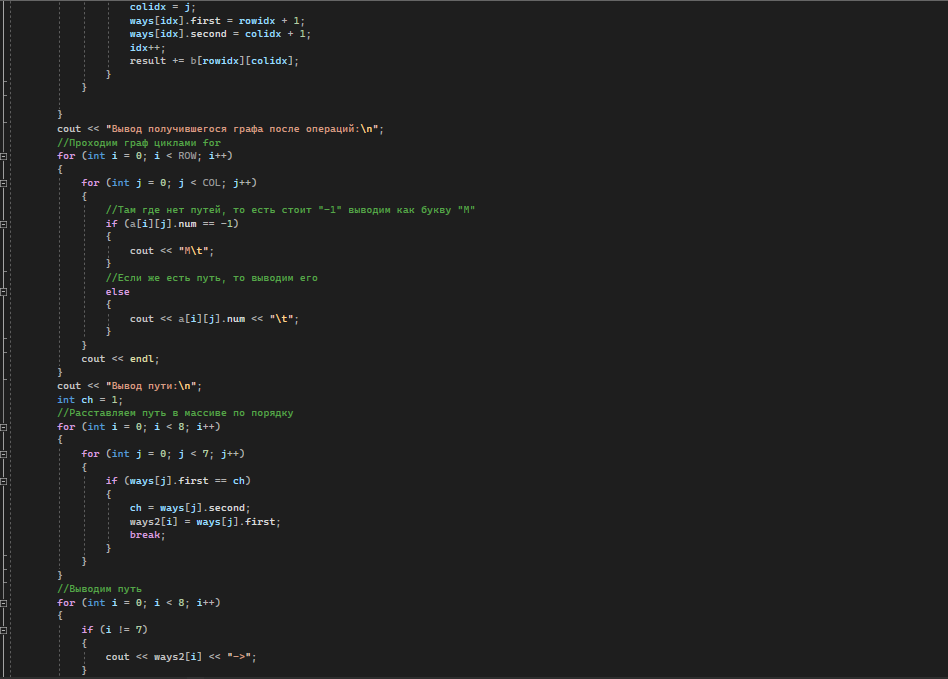


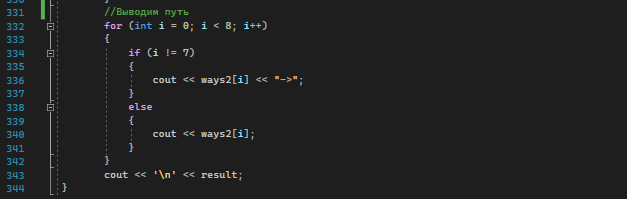




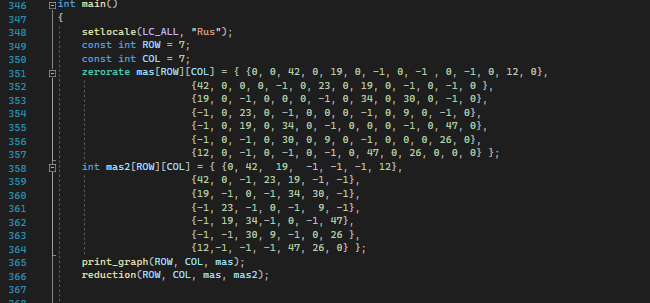








1. Функция main



1. Отрисовка графа (также в функции main)



