

1. ТРЕБОВАНИЯ К ПРОГРАММЕ

Программа должна реализовывать алгоритм Краскала для поиска минимального остовного дерева в неориентированном графе, а также интерфейс к нему, который должен визуализировать построение МОДа по шагам алгоритма.

1.1 Исходные требования к программе:

1.1.1 Требование к вводу исходных данных:

1.1.1.1 Ввод через взаимодействие с холстом:

1.1.1.1.1 Добавление вершин кликом по холсту.

1.1.1.1.2 Добавление ребер кликом по двум вершинам.

1.1.1.1.3 Добавление и изменение веса ребра кликом по ребру.

1.1.1.2 Ввод через текстовый файл – матрицей смежности.

Вершины размещаются в форме правильного многоугольника.

1.1.1.3 Генерация матрицы смежности для входных данных. С указанием количества вершин и ребер.

1.1.2 Требования к визуализации:

1.1.2.1 Основные элементы интерфейса:

1.1.2.1.1 Холст для отображения графа. При рассмотрении потенциального ребра оно рисуется светло-серым цветом. Если ребро не добавляется из-за цикла(цикл показывается светло-розовым цветом), то само ребро перекрашивается в красный цвет.

1.1.2.1.2 Панель управления с кнопками взаимодействия.

1.1.2.1.3 Поле для отображения текущего шага алгоритма.

1.1.2.2 Функционал интерфейса:

1.1.2.2.1 При успешном добавлении ребра в МОД – раскрашивать его в зеленый цвет на холсте.

1.1.2.2.2 На каждом шаге алгоритма выводить текстовую информацию о данном шаге – какие ребра уже были

добавлены в МОД до шага; какое ребро рассматривается; добавляется ли рассматриваемое ребро в итоговый МОД или нет.

1.1.2.2.3 Функционал свободного перемещения между шагами алгоритма.

1.1.2.2.4 При отмене ввода веса ребра не появляется окно с ошибкой неверного веса ребра.

1.1.2.2.5 При последовательном перемещении вершин не совершается попытка создания ребра.

1.1.3 Требование к выходным данным:

1.1.3.1 Текстовый файл с ребрами входящими в МОД.

1.1.3.2 Графическое представление МОД в приложении.

1.1.4 Схематическое представление графического представления:

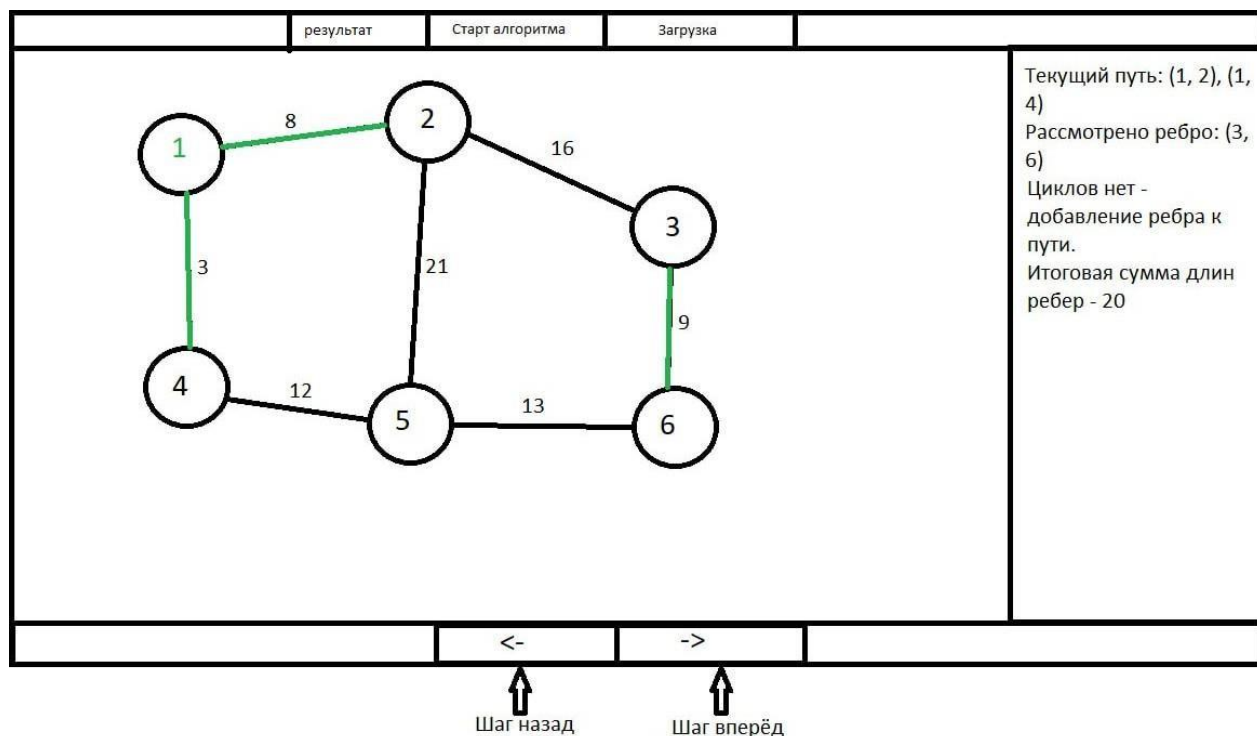


Рисунок 1 – Графический интерфейс

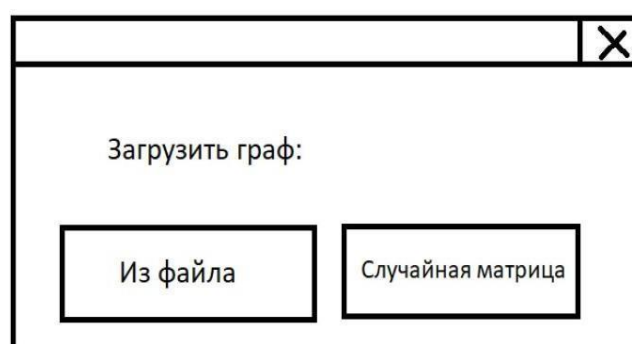


Рисунок 2 – Окно загрузки графа

1.2 Уточнение требований после сдачи прототипа

- 1.2.1 Добавить сверху приложения кнопку “Результат”.
- 1.2.2 Если нажата "Cancel" или закрыто окно для ввода веса, то не надо сообщать о некорректном вводе веса ребра.
- 1.2.3 Если пользователь сначала переместил одну вершину, то должна быть возможность затем сразу переместить другую, без диалога ввода ребра (программа должна учитывать, был ли щелчок по вершине или пользователь её двигал).

1.3 Уточнение требований после сдачи версии 1.0

- 1.3.1 Реализовать функцию запроса-подтверждения на прерывание алгоритма.
- 1.3.2 Добавить возможность удалять отдельные ребра.

1.4 Уточнение требований после сдачи версии 2.0

- 1.4.1 Исправить баг:
 - 1) переместить вершину а;
 - 2) щелчок по вершине б;
 - 3) щелчок по вершине в — ребро не создаётся;
 - 4) попытка переместить вершину а — вместо этого создаётся ребро.
- 1.4.2 Отвергнутые ребра рисовать немного более темным оттенком.
- 1.4.3 Исправить баг: “если сделать первый шаг и вернуться в начало, то появляется возможность редактировать граф: изменения не учитываются при дальнейшем выполнении алгоритма, и возможность редактирования не пропадает при последующих шагах.”

1. ПЛАН РАЗРАБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ

2.1. План разработки

Дата	Этап проекта	Реализованные возможности	Выполнено
27.06.2025	Согласование спецификации		
30.06.2025	Сдача прототипа	Разработан интерфейс а так же сам алгоритм, возможность запуска алгоритма в интерфейсе. Предоставление списка ребер входящих в МОД в текстовом окне.	
02.07.2025	Сдача версии 1	Визуализация готового МОД с отображением рёбер на холсте, включая пошаговое выполнение алгоритма с добавлением новых рёбер на полотно и текстовым описанием каждого шага, с возможностью перехода вперёд и назад	
04.07.2025	Сдача версии 2	Загрузка графа из файла, добавиться возможность сразу увидеть готовый ответ, визуализация	

		шагов алгоритма на графе пользователя.	
	Сдача версии 3	-	
07.07.2025	Сдача отчёта		
08.07.2025	Защита отчёта		

2.2. Распределение ролей в бригаде

2.2.1 Реализация графического интерфейса:

2.2.1.1 Общий дизайн графического интерфейса и его реализация –
Ягудин Д.Р.

2.2.2 Реализация алгоритма Краскала – Че М.Б.

2.2.3 Взаимодействие интерфейса с алгоритмом Краскала и тестирование
– Костромитин М.М.