

Компьютерная программа для демонстрации решения задачи локализации точки в планарном прямолинейном графе методом полос.

**Автор:** Снетков Михаил, ИМ14-06Б

СФУ, 2017

Руководство пользователя

Локализация в ППЛГ методом полос

Руководство пользователя

# Системные требования

* Windows 7, 8, 10, или Linux, или macOS;
* 200 МБ свободного места на диске;
* Не менее 256 МБ ОЗУ.

# основные возможности

* Визуальный вывод результата работы программы;
* Хранение списка рёбер графа в файле с расширением *«.edgeslist»*;
* Работа с графом через интерфейс программы:
  + Добавление ребра;
  + Удаление ребра;
  + Загрузка списка рёбер из файла;
  + Сохранение списка рёбер в файл;
  + Общая информация.
* Локализация точки в ППЛГ методом полос;
* Настройки вывода координатной плоскости:
  + Отображение нахождения граней на плоскости;
  + Отображение линий, ограничивающих полосы метода.
* Обработка ошибок входных данных, в том числе проверки:
  + Планарность графа;
  + Однокомпонентная связность графа;
  + Цикличность графа;
  + Дублирование рёбер графа.

# основное окно программы

Программа работает в однооконном режиме (рис. 1). Системные диалоги используются только при работе с файлами: сохранение и открытие списка рёбер графа.

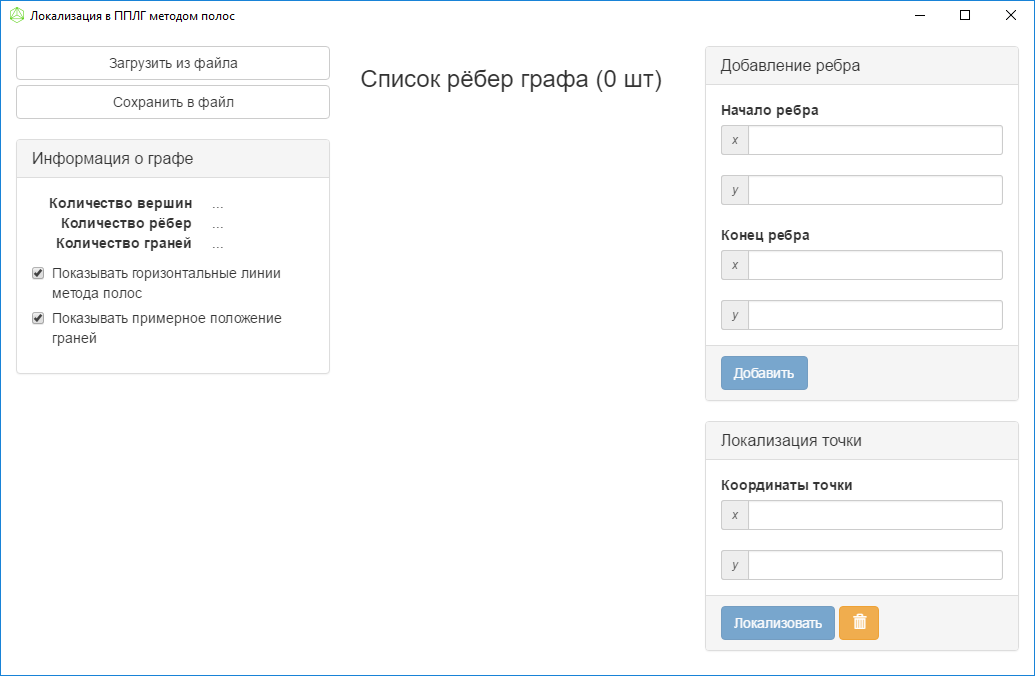


Рисунок 1. Основное окно программы

При запуске программы список рёбер графа пуст, координатная плоскость не отображается. Есть 2 способа начать работать с программой:

* Ввести рёбра графа вручную, используя раздел «Добавление ребра».
* Загрузить список рёбер из файла.

ВНИМАНИЕ! Список рёбер из файла замещает собой список рёбер программы. Однако при ручном вводе ребра список будет дополнен этим ребром.

# 

Рисунок 2. назначение основных областей программы

|  |  |  |  |
| --- | --- | --- | --- |
| **Область** | **Название** | **Описание** | **Примечание** |
| 1 | Работа с файлами | Загрузка и сохранение списка рёбер | Файл с расширением “.edgeslist” |
| 2 | Информация о графе | Численные характеристики графа, настройки отображения вывода на координатную плоскость вспомогательных элементов |  |
| 3 | Список рёбер графа | Таблица с рёбрами графа | Просмотр и удление рёбер |
| 4 | Добавление ребра | Ввод ребра | Происходят проверки данных и корректности получающегося графа |
| 5 | Локализация точки | Ввод исследуемой точки | Происходят проверки данных. Для локализации необходим корректный граф |

ВНИМАНИЕ! При работе с непустым списком рёбер над перечисленными областями будет отображена координатная плоскость с графом и сопутствующими построениями.

# Пример работы программы

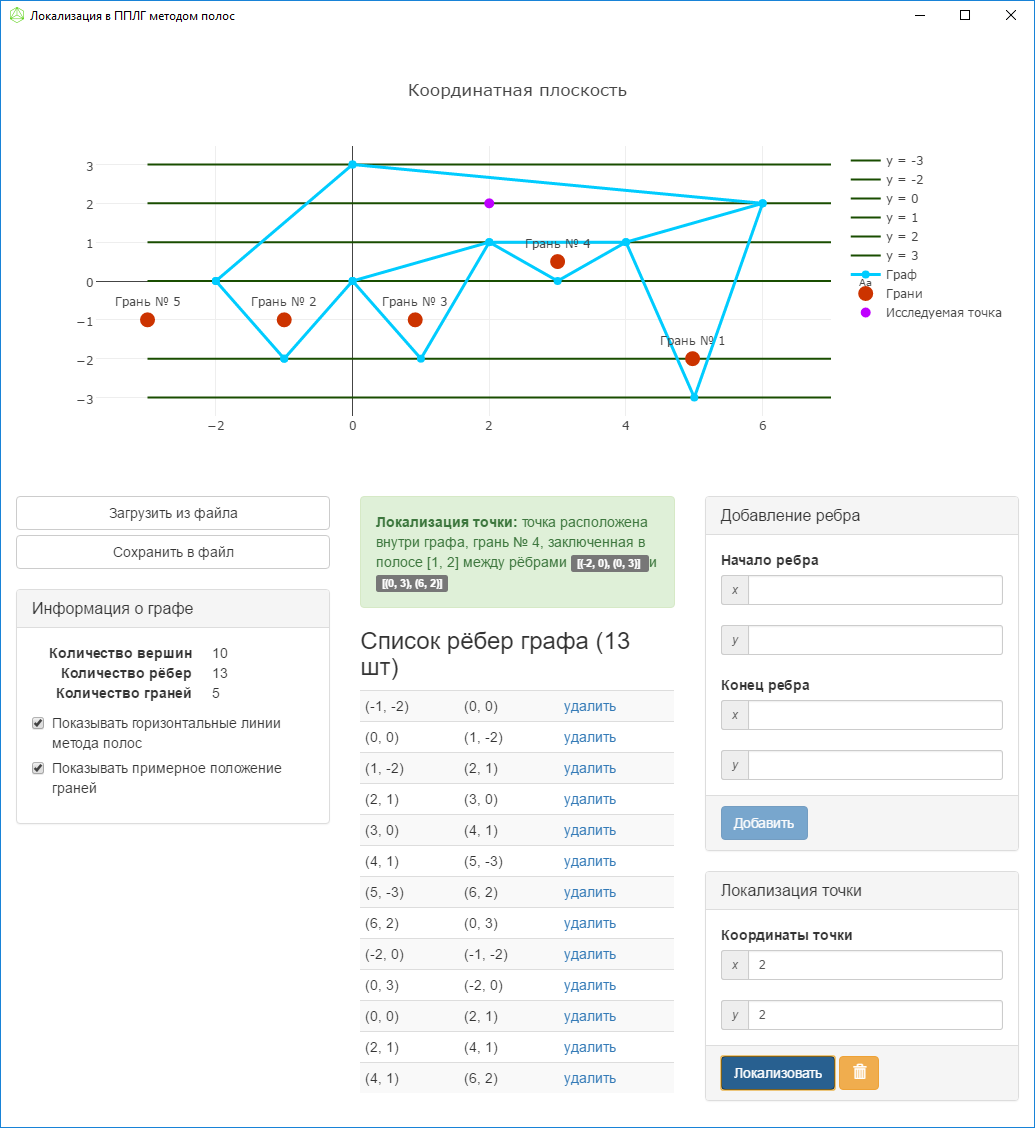


Рисунок 3. ПРИМЕР РАБОТЫ ПРОГРАММЫ

При вводе графа в области № 2 (рис. 2) появились численные характеристики графа (количество вершин, рёбер, граней). В области № 3 таблица со всеми рёбрами графа. Нажатие на «удалить» удалить соответствующую строку таблицы. В области № 5 указана точка (2, 2), для которой вверху области № 2 выведен ответ: расположение, грань, полоса и боковые стороны трапеции. Допустимые варианты ответа: точка внутри графа, на ребре графа, в вершине графа, снаружи графа (на внешней грани).

Над перечисленными областями появляется координатная плоскость с построениями, необходимыми для визуальной наглядности:

* Синими рёбрами выделен граф;
* Тёмно-зелёными горизонтальными отрезками выделены границы полос;
* Тёмно-красные точки визуально нумеруют соответствие граней ППЛГ;
* Фиолетовым цветом нанесена точка, задачу локализации которой решает программа.

# Обработка ошибок

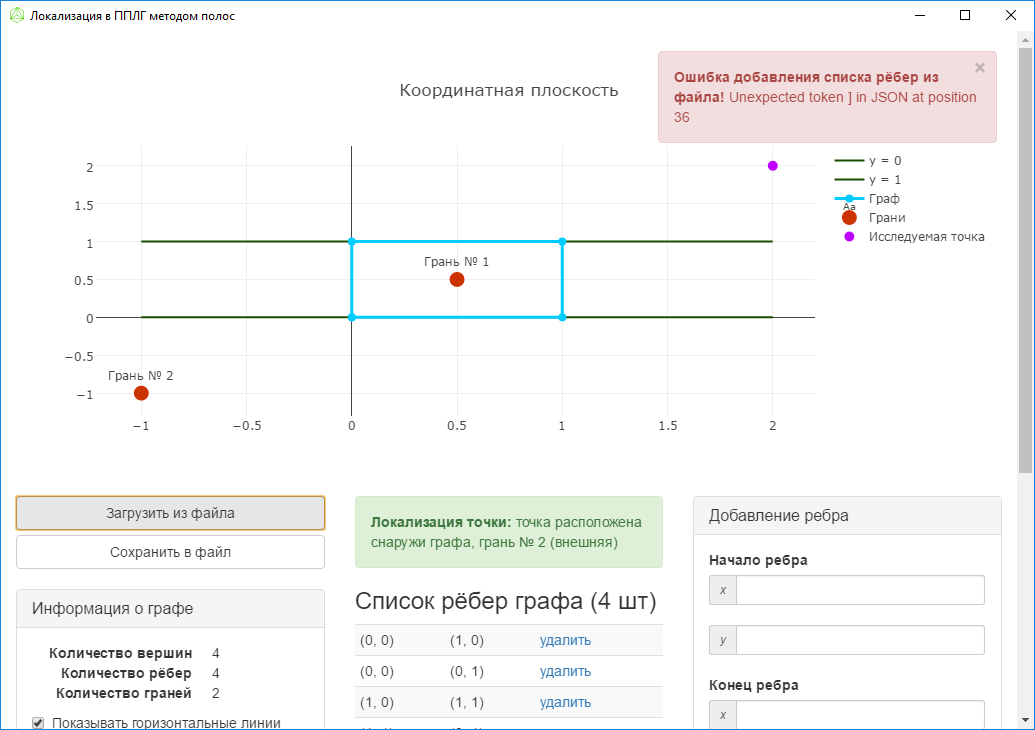


Рисунок 4. ПРИМЕР ОШИБКИ ПРИ ЗАГРУЗКЕ ФАЙЛА

В программе реализована обработка ошибок во время исполнения, связанных с обработкой входных данных (список рёбер, добавление ребра, точка локализации). Уведомления об ошибках имеют вид всплывающих сообщений на красно-розовом фоне в правом верхнем углу программы (рис. 4). Сообщений может быть несколько. Для закрытия сообщения можно использовать «крестик» в правом верхнем углу сообщения.

# ФОРМАТ ФАЙЛА с расширением «.edgeslist»

Формат файла представлен широко распространённым текстовым форматом обмена данных, основанном на JavaScript – **JSON**. Каждое ребро [(a, b), (c, d)] представлено массивом из 4 численных значений: a, b, c, d. В JSON подобный массив записывается как [a, b, c, d]. В файле хранится массив подобных массивов, разделенных запятыми.

|  |  |
| --- | --- |
| Исходный список рёбер (пример) | В формате JSON |
| [(1, 2), (3, 4)] | [ [ 1, 2, 3, 4 ] ] |
| [(1, 2), (3, 4)], [(3, 4), (5, 6)] | [ [ 1, 2, 3, 4 ] , [ 3, 4, 5, 6 ] ] |

ВНИМАНИЕ! При некорректных данных в файле список рёбер не будет загружен в программу.

ВНИМАНИЕ! Формат файл является текстовым, поэтому его можно редактировать обычными текстовыми процессорами. Не нарушая общей структуры записи лексем, можно использовать любое количество символов табуляции, пробелом и переносов строки.