ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук

Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Доцент департамента больших  данных и информационного поиска  ФКН НИУ ВШЭ,  канд. физ.-мат. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.Л. Чернышев  « \_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г. | УТВЕРЖДАЮ  Академический руководитель  образовательной программы  «Программная инженерия», профессор,  канд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  « \_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г. |

|  |  |
| --- | --- |
| Инв. № подл. |  |
| Подп. и дата |  |
| Взам. Инв. № |  |
| Инв. № дубл. |  |
| Подп. и дата |  |

**Программа для нахождения**

**явного вида аналитических функций,**

**связанных с метрическими графами**

**Программа и методика испытаний**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.10.03-01 51 01-1-ЛУ**

Исполнитель

студент группы БПИ181

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/А.И. Уварова/

« \_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

УТВЕРЖДЕН

RU.17701729.10.03-01 51 01-1 ЛУ

**Программа для нахождения**

|  |  |
| --- | --- |
| Инв. № подл. |  |
| Подп. и дата |  |
| Взам. Инв. № |  |
| Инв. № дубл. |  |
| Подп. и дата |  |

**явного вида аналитических функций,**

**связанных с метрическими графами**

**Программа и методика испытаний**

**RU.17701729.10.03-01 51 01-1**

**Листов 20**

**Оглавление**

[**1.** **Объект испытаний** 3](#_Toc39703945)

[**1.1** **Наименование программы** 3](#_Toc39703946)

[**2.** **Цель испытаний** 3](#_Toc39703947)

[**3.** **Требования к программе** 4](#_Toc39703948)

[**4.** **Требования к программной документации** 6](#_Toc39703949)

[**4.1. Состав программной документации** 6](#_Toc39703950)

[**5.** **Средства и порядок испытаний** 7](#_Toc39703951)

[**5.1. Технические средства** 7](#_Toc39703952)

[**5.2. Программные средства** 7](#_Toc39703953)

[**5.3. Порядок проведения испытаний** 7](#_Toc39703954)

[**6.** **Методы испытаний** 8](#_Toc39703955)

[**6.1** **Испытание выполнения требований к программной документации** 8](#_Toc39703956)

[**6.2. Испытание выполнения требований к функциональным характеристикам** 8](#_Toc39703957)

[**6.2. Проверка работы алгоритма** 14](#_Toc39703958)

[**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ** 20](#_Toc39703959)

# **Объект испытаний**

## **Наименование программы**

Наименование программы – «Программа для нахождения явного вида аналитических функций, связанных с метрическими графами».

Наименование на английском языке – «A program for finding the explicit form of analytical functions associated with metric graphs».

* 1. **Область применения**

Программа предназначена для построения моделей неориентированных метрических графов, а также позволяет вычислять по модели некоторый набор аналитических функций. Возможные области применения: исследования в области дискретной математики или теории графов, использование в обучающих целях преподавателями или студентами, изучающими графы.

# **Цель испытаний**

Испытания проводились с целью проверки корректности выполнения функций программы, перечисленных в разделе «Требования к программе».

# **Требования к программе**

Программа должна соответствовать следующим требованиям к надежности и составу выполняемых функций, указанным в документе «Программа для нахождения явного вида аналитических функций, связанных с метрическими графами». Техническое задание.

**Требования к функциональным характеристикам**:

1. Должна быть возможность редактировать граф путем добавления/удаления узлов/рёбер.
2. Должна быть возможность создать неориентированный граф.
3. Должны быть реализованы редактируемые LaTeX-веса для рёбер.
4. Должна быть возможность превратить LaTeX-выражения в подписях в вычисленные значения и обратно.
5. Ребра должны быть реализованы в виде изменяемых дуг.
6. Должна быть возможность задавать кратные рёбра.
7. Должна быть возможность по заданному пользователем графу вывести вид следующих аналитических функций:

А) Функция магнитуд (Magnitude function)

Б) Дзета-функция Ихары на вершинах с весами и без (Vertex Ihara zeta function)

В) Дзета-функция Ихары на ребрах (Edge Ihara zeta function)

Г) Дзета-функция Ихары на путях (Path Ihara zeta function)

Д) Функция путей (Custom path function)

1. Для графа необходимо вывести:

А) исходную формулу и пояснения по ней, если необходимо

Б) промежуточные вычисления, если необходимо (например, матрица для графа)

1. В) по возможности сокращенный итоговый ответ (функция с аргументом)
2. Вывод вычисленных функций необходимо осуществлять в текстовый файл (.txt, .tex) в формате LaTeX
3. Программа должна уметь вычислять функции с числовыми весами для полных графов с пятью вершинами (и для более простых графов).
4. Программа должна уметь вычислять функции для графов, где есть LaTeX-веса (корни n-ной степени)
5. У пользователя должна быть возможность сохранить граф в одном из предложенных форматов.
6. Возможность загрузить ранее сохраненную с помощью программы модель графа.
7. Должны быть реализованы всплывающие подсказки в случае некорректных действий пользователя.

**Требования к интерфейсу:**

Пользовательский интерфейс должен быть понятным и логичным, обеспечивать высокую скорость работы пользователя, соответствовать его задачам, а также способствовать быстрому обучению работе с программой.

Интерфейс должен быть реализован полностью на английском языке.

Рабочее окно программы должно иметь как минимум две области: область отрисовки(справа) и панель инструментов (слева). Панель инструментов должна соответствовать всем требованиям к составу выполняемых функций.

Должна быть реализована справка по работе с программой.

**Требования к входным данным**:

Пользователь создает модель графа с помощью мыши/тачпада. Также может быть загружена уже ранее созданная в программе модель.

**Требования к выходным данным**:

Пользователь может сохранить созданную модель графа в одном из предложенных расширений. При вычислении функции пользователь может сохранить результат в файл формата .tex/.txt.

# **Требования к программной документации**

## **4.1. Состав программной документации**

* «Программа для нахождения явного вида аналитических функций, связанных с метрическими графами». Техническое задание (ГОСТ 19.201-78)
* «Программа для нахождения явного вида аналитических функций, связанных с метрическими графами». Программа и методика испытаний (ГОСТ 19.301-78)
* «Программа для нахождения явного вида аналитических функций, связанных с метрическими графами». Пояснительная записка (ГОСТ 19.404-79)
* «Программа для нахождения явного вида аналитических функций, связанных с метрическими графами». Текст программы (ГОСТ 19.401-78)
* «Программа для нахождения явного вида аналитических функций, связанных с метрическими графами». Руководство оператора (ГОСТ 19.505-79)

**4.2. Специальные требования к программной документации**

* Пояснительная записка должна быть загружена через LMS в систему «Антиплагиат».
* Вся документация должна быть оформлена в соответствии с ГОСТ 19.106-78 и ГОСТ к данному виду документа (см. п. 5.1.)
* Документация и программа cдается в электронном виде в формате .pdf или .docx. в архиве формата .zip или .rar.

# **Средства и порядок испытаний**

## **5.1. Технические средства**

Для надёжной работы программы необходим ПК, обладающий следующими минимальными характеристиками:

1) ОС Linux

2) минимум 500 МБ оперативной памяти

3) минимум 500 MB на жёстком диске

4) Клавиатура и мышь/тачпад

5) Видеокарта, поддерживающая разрешение не менее чем 900х650 точек

6) Центральный процессор c поддержкой набора инструкций SSE2

## **5.2. Программные средства**

1. ОС Linux
2. Java 11

3. Python версии 3+ и библиотека SymPy

## **5.3. Порядок проведения испытаний**

Испытания должны проводиться в следующем порядке:

1) проверка требований к программной документации;

2) проверка требований к функциональным характеристикам;

# **Методы испытаний**

Испытания представляют собой процесс установления соответствия программы и программной документации заданным требованиям.

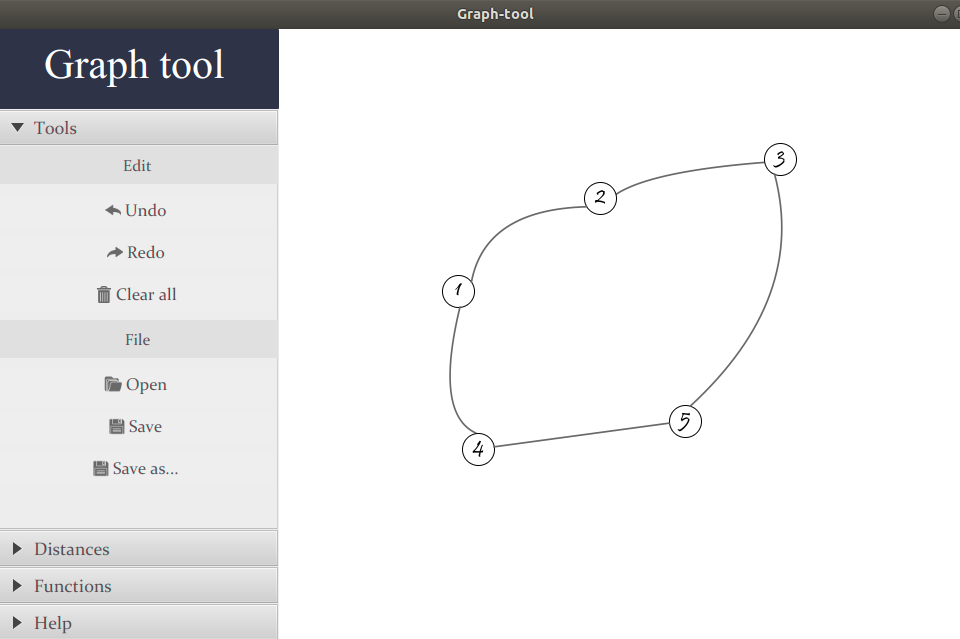
* 1. **Испытание выполнения требований к программной документации**

Состав программной документации проверяется визуально, проверяется наличие программной документации в системе LMS. Также визуально проверяется соответствие документации требованиям ГОСТ.

## **6.2. Испытание выполнения требований к функциональным характеристикам**

1. Должна быть возможность редактировать граф путем добавления/удаления узлов/рёбер.
2. Должна быть возможность создать неориентированный граф.

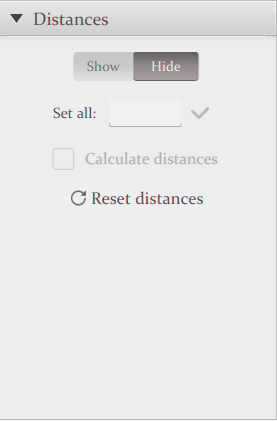
Для создания вершины графа необходимо кликнуть по области отрисовки, вершина появляется в месте клика. Чтобы переместить вершину, необходимо мышью перетащить её в нужное место. Чтобы создать ребро, необходимо создать две вершины. Затем нужно кликнуть по одной из вершин – за курсором потянется ребро. Чтобы соединить ребро со второй вершиной, достаточно кликнуть по ней. Форма дуг ребер может быть изменена с помощью перетаскивания якоря в центре ребра (рис. 1).



*Рисунок 1. Нарисованный граф*

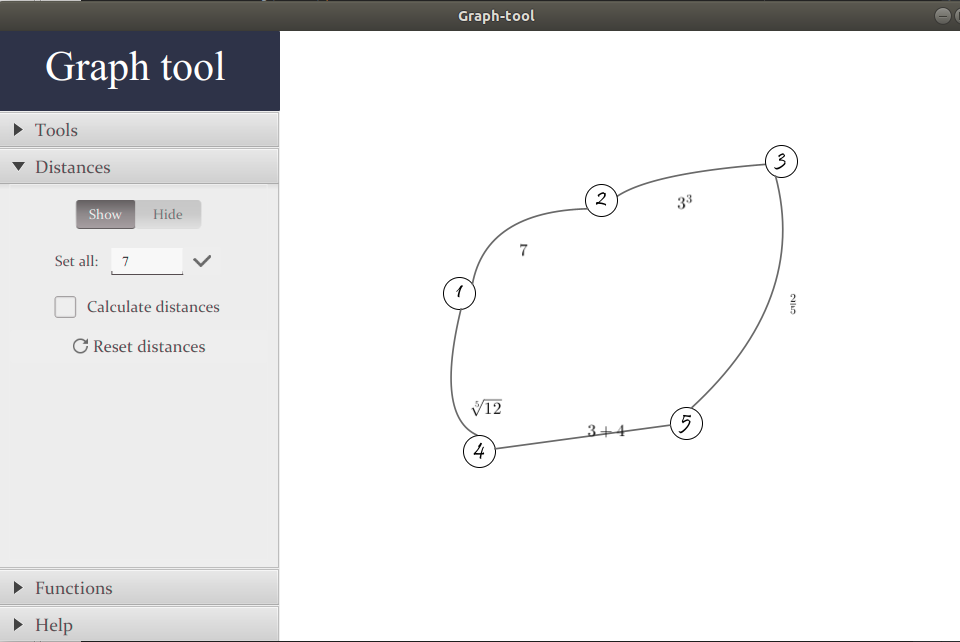
1. Должны быть реализованы редактируемые LaTeX-веса для рёбер.

Для отображения подписей необходимо включить их отображение, переключив toggle-button во вкладке Distances в положение show (рис.2)



*Рисунок 2. Меню Distances*

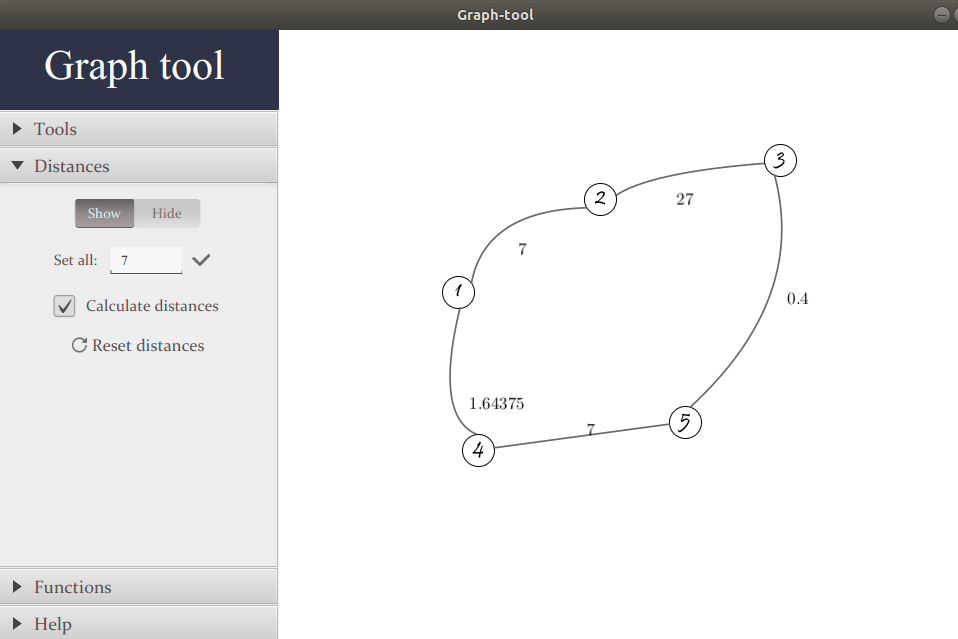
Для редактирования подписи достаточно кликнуть по ней и ввести нужное выражение в появившееся окно ввода (рис. 3)



*Рисунок 3. Граф с LaTeX-подписями на ребрах*

1. Должна быть возможность превратить LaTeX-выражения в подписях в вычисленные значения (например, «2+3» и «5») и обратно.

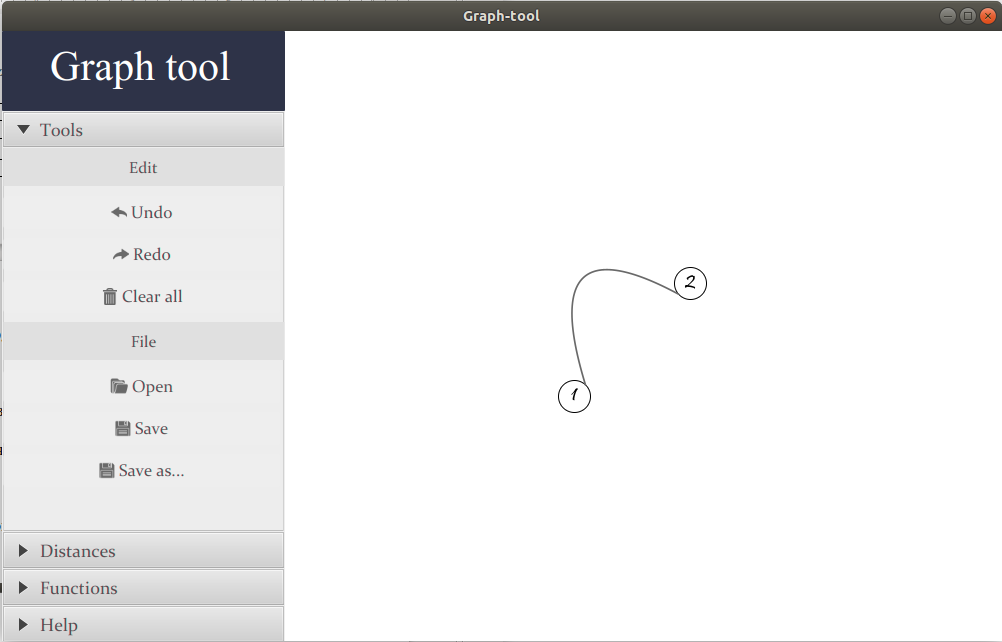
Для вычисления математических выражений достаточно пометить выбранным checkbox ‘Calculate distances’ на панели Distances. Результат для рис.3 представлен на рис. 4.



*Рисунок 4. Математические выражения в длинах рёбер после вычисления*

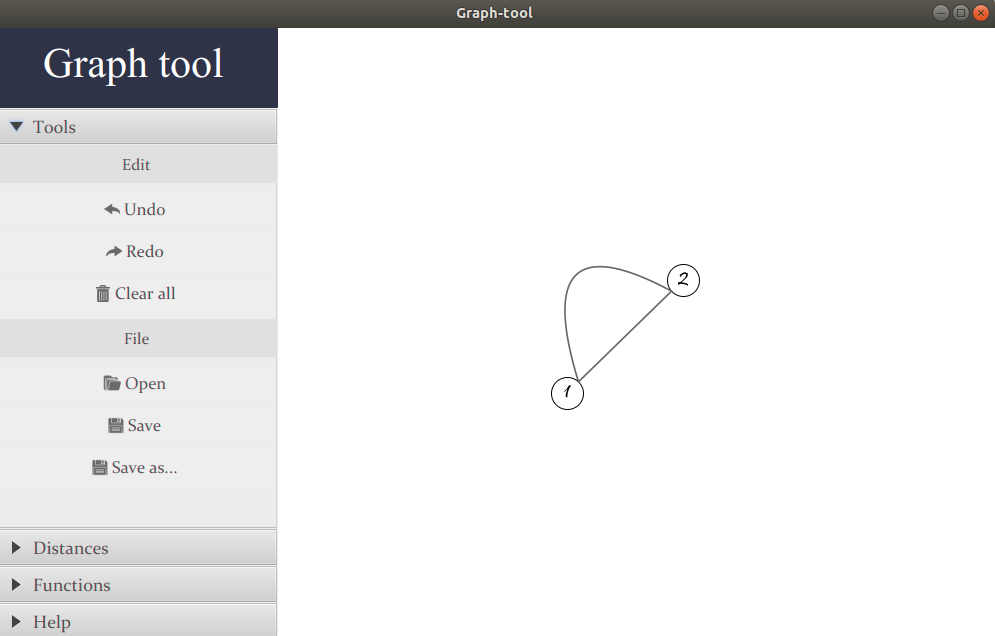
1. Ребра должны быть реализованы в виде изменяемых дуг.

Форма ребра изменяется с помощью якоря (рис.5)

*Рисунок 5. Панель инструментов*

1. Должна быть возможность задавать кратные ребра.

Пример кратного ребра (рис. 6)

*Рисунок 6. Кратное ребро*

1. Должна быть возможность по заданному пользователем графу вывести вид следующих аналитических функций:

А) Функция магнитуд (Magnitude function)

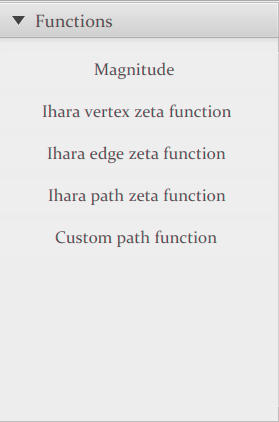
Б) Дзета-функция Ихары на вершинах с весами и без (Vertex Ihara zeta function)

В) Дзета-функция Ихары на ребрах (Edge Ihara zeta function)

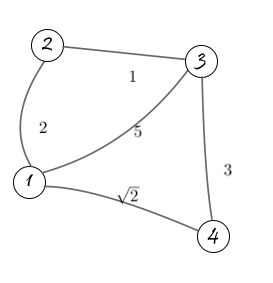
Г) Дзета-функция Ихары на путях (Path Ihara zeta function)

Д) Функция путей (Custom path function)

Нужную функцию можно запустить с помощью панели Functions (рис.7)

*Рисунок 7. Панель функций*

Граф, для которого производились все вычисления (рис. 8)

*Рисунок 8. Тестовый граф*

А) Магнитуда графа (рис.9)

Б) Функция Ихары на вершинах (рис.10), взвешенная

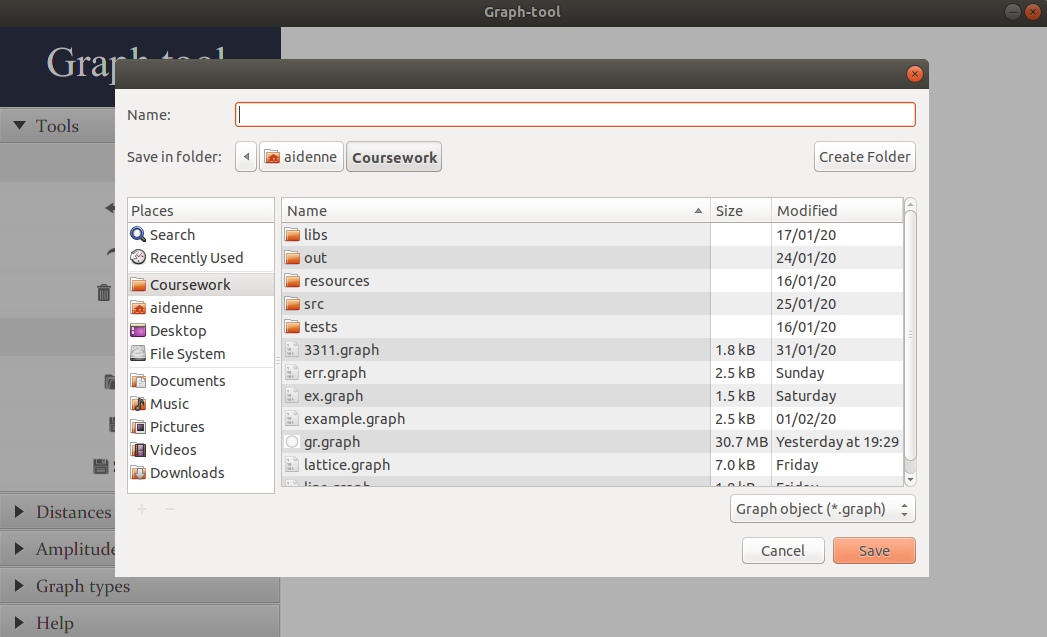
В) Функция Ихары на ребрах (рис. 11)

Г) Функция Ихары на путях (рис. 12)

Д) Функция путей (рис. 13)

1. У пользователя должна быть возможность сохранить граф в формате .graph.

Сохранить файл можно с помощью кнопок Save и Save as на панели Tools. (рис. 5,6)



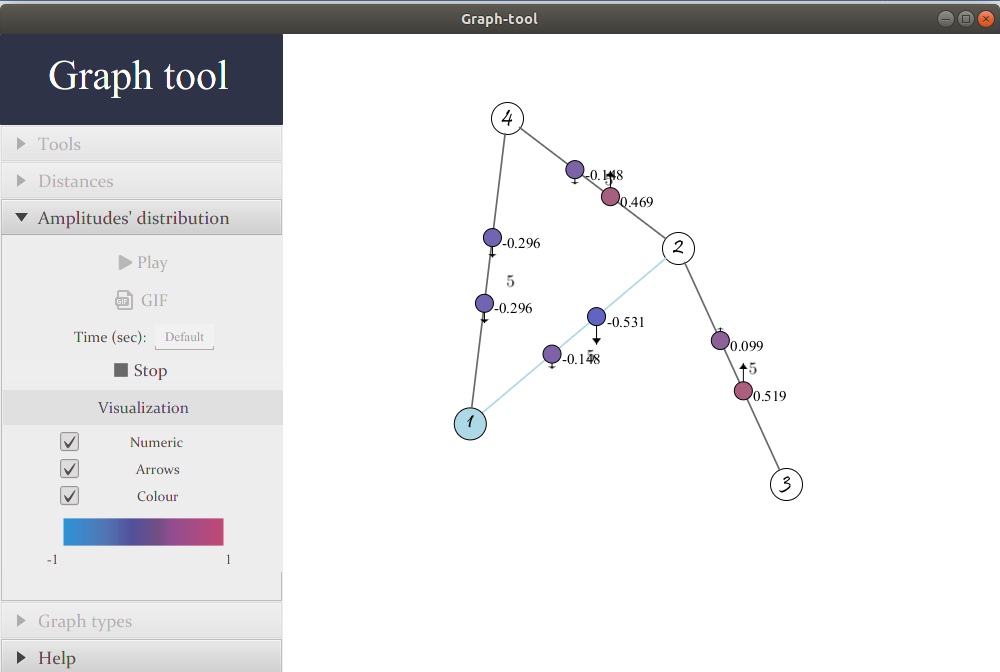
*Рисунок 6. Файловое меню*

1. Возможность загрузить ранее сохраненную с помощью программы модель графа.

Модель графа можно загрузить с помощью кнопки Open на панели Tools (рис. 5)

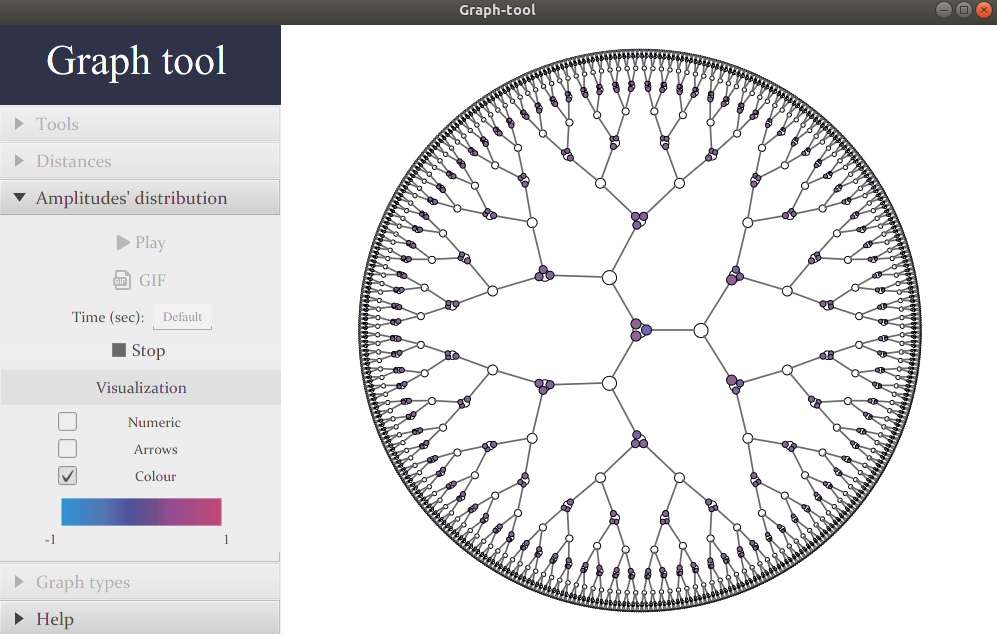
1. Возможность запустить алгоритм визуализации распределения амплитуд как для построенного пользователем конечного графа (при выполнении некоторых условий), так и на одной из встроенных в программу моделей.

Управление визуализацией происходит с помощью панели Amplitudes’ distribution. Достаточно построить корректный граф и нажать на кнопку Play (рис. 7).



*Рисунок 7. Визуализация*

Для запуска алгоритма на встроенной модели достаточно выбрать ее на панели Types и запустить визуализацию аналогичным образом (рис. 8)



*Рисунок 8. Визуализация на встроенном графе (решетка Бете)*

1. Должны быть реализованы следующие способы представления амплитуд:

А) Цветовой круг (точка меняет цвет с изменением значения амплитуды)

Б) Числовое значение над точкой

В) Стрелка над точкой, изменяющая свою длину в соответствии с изменением амплитуды

Управление способом представления амплитуд производится через соответствующие чекбоксы на панели Amplitudes’ distribution (рис.7,8)

1. Должна быть возможность сохранить работу алгоритма визуализации в формате .gif

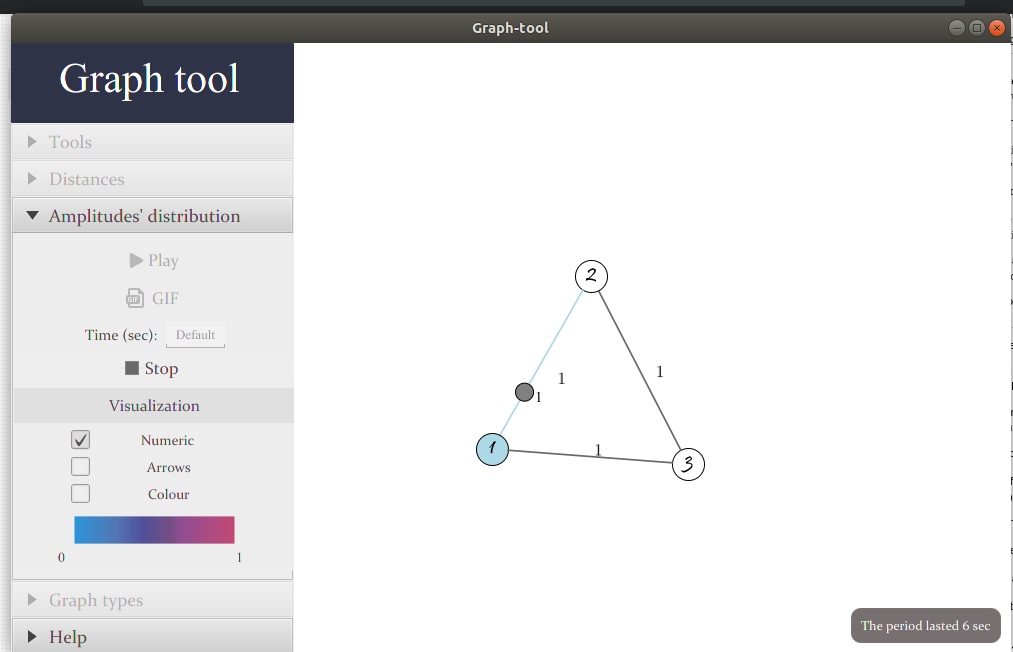
Запись алгоритма визуализации производится с помощью кнопки GIF на панели Amplitudes’ distribution (рис. 7, 8)

1. Должна присутствовать возможность самостоятельно выбрать длительность .gif анимации.

Длительность выбирается с помощью окна под кнопкой GIF (рис. 7,8)

1. Должны присутствовать всплывающие подсказки о длительности периода в визуализации, если он присутствует.

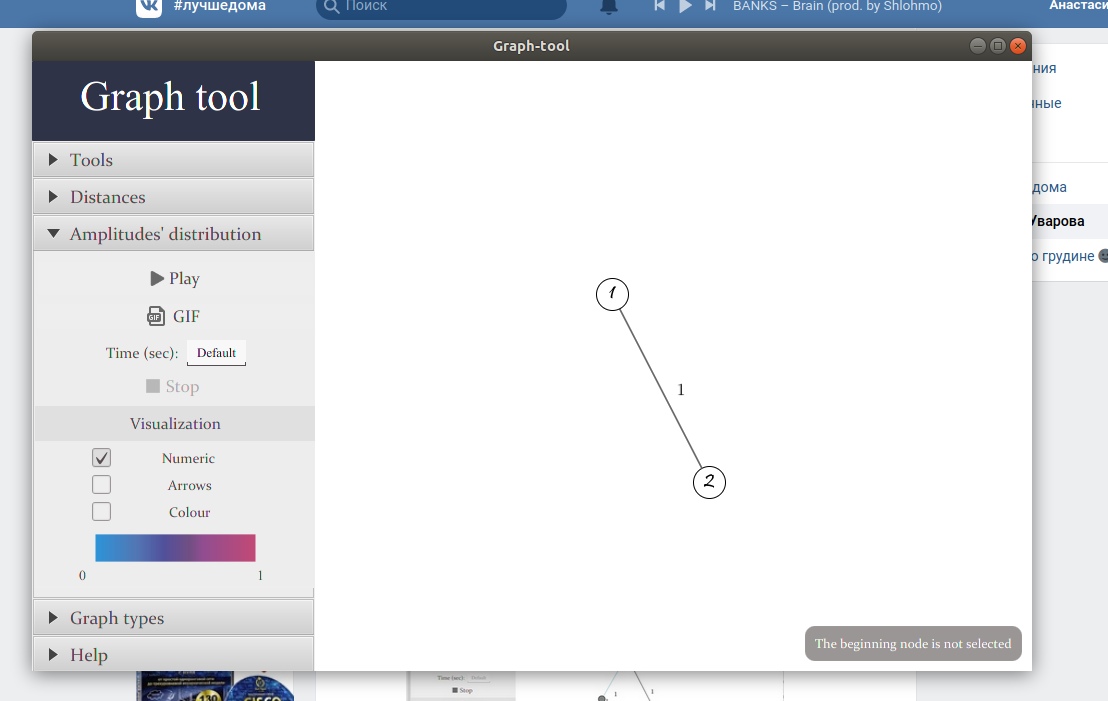
Всплывающие подсказки реализованы в виде всплывающего окна в правом нижнем углу области отрисовки (рис. 9).



*Рисунок 9. Уведомление о длительности периода*

1. Должны быть реализованы всплывающие подсказки в случае некорректных действий пользователя.

Пример на рис.10.

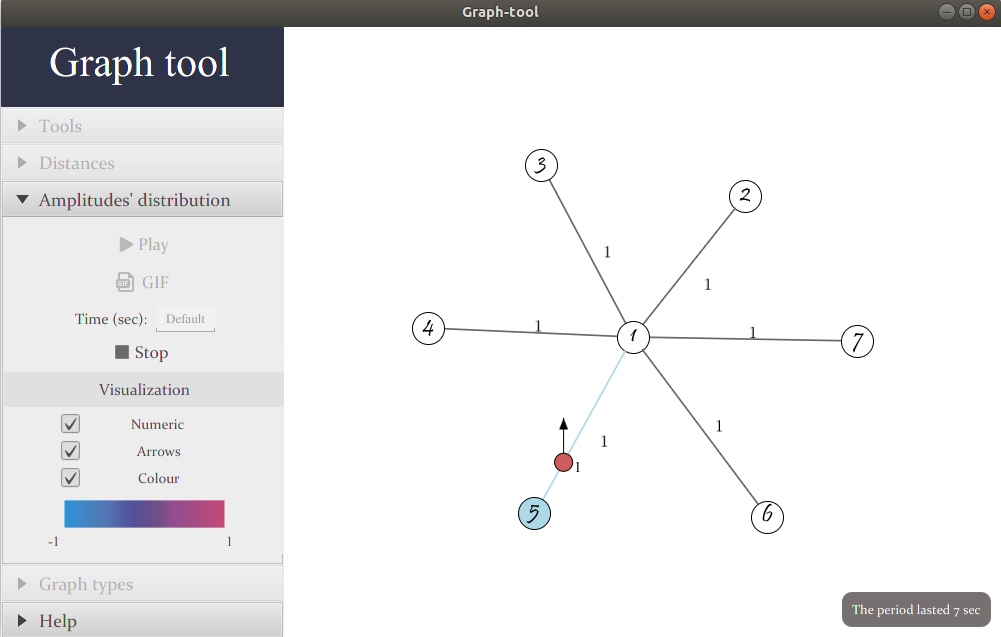


*Рисунок 10. Не выбрана стартовая вершина*

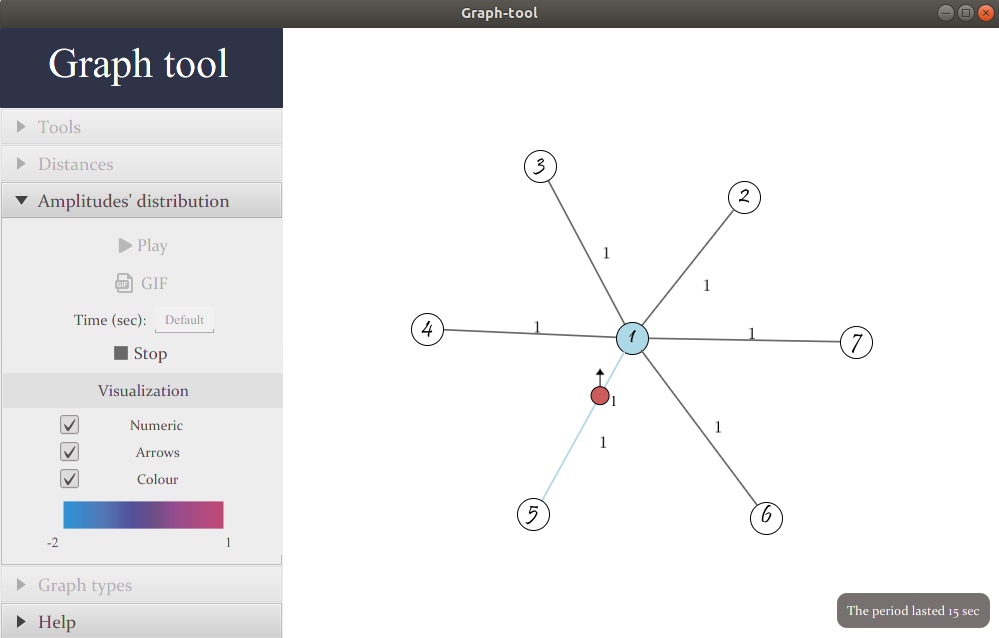
## **6.2. Проверка работы алгоритма**

Проверка алгоритма проводилась для графов следующих типов: звездный, полный, граф Петерсена, граф-колесо. Цель испытания: проверить, является ли распределение периодичным для данных типов графов.

1. Звездный граф – связный граф, в котором все ребра исходят из одной вершины. Запуск алгоритма на данном графе показал, что распределение амплитуд на графе периодично для случая равных ребер (рис. 11, 12)

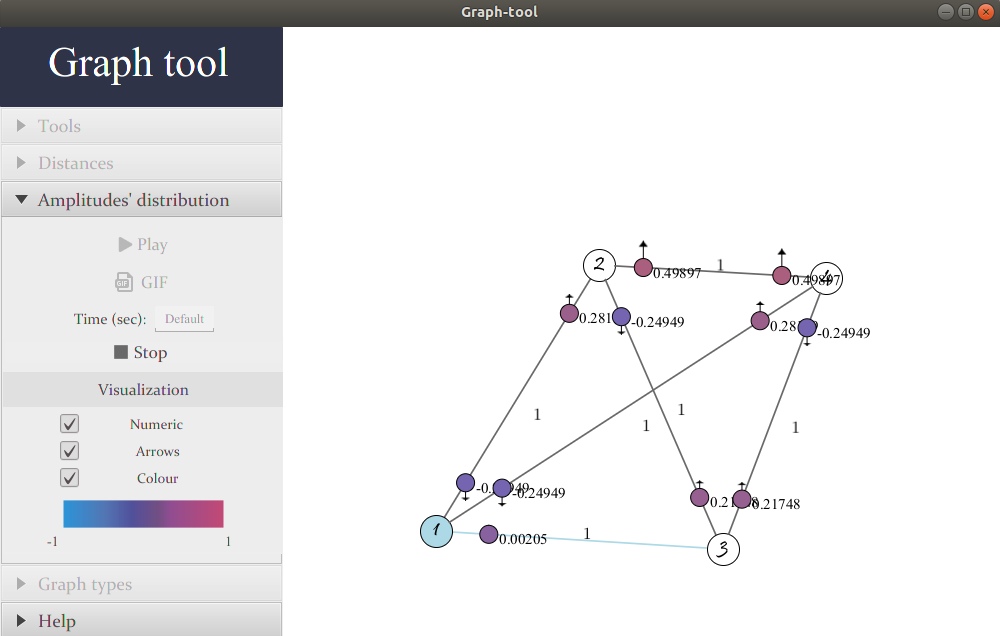


*Рисунок 11. Период 7 секунд, если стартовая вершина не центральная*

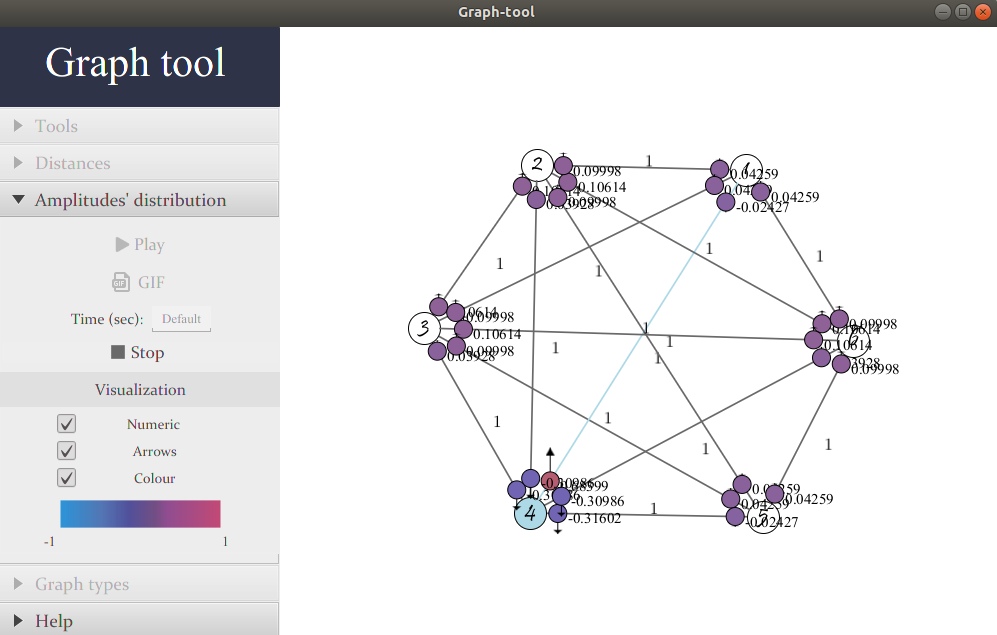


*Рисунок 11. Период 15 секунд, если стартовая вершина центральная*

1. Полный граф – граф, в котором каждая пара различных вершин смежна. Для случая двух и трех вершин визуализация периодична (рис. 9 для 3х вершин). Испытания для большего числа не дали периодичного результата (рис. 12, 13)

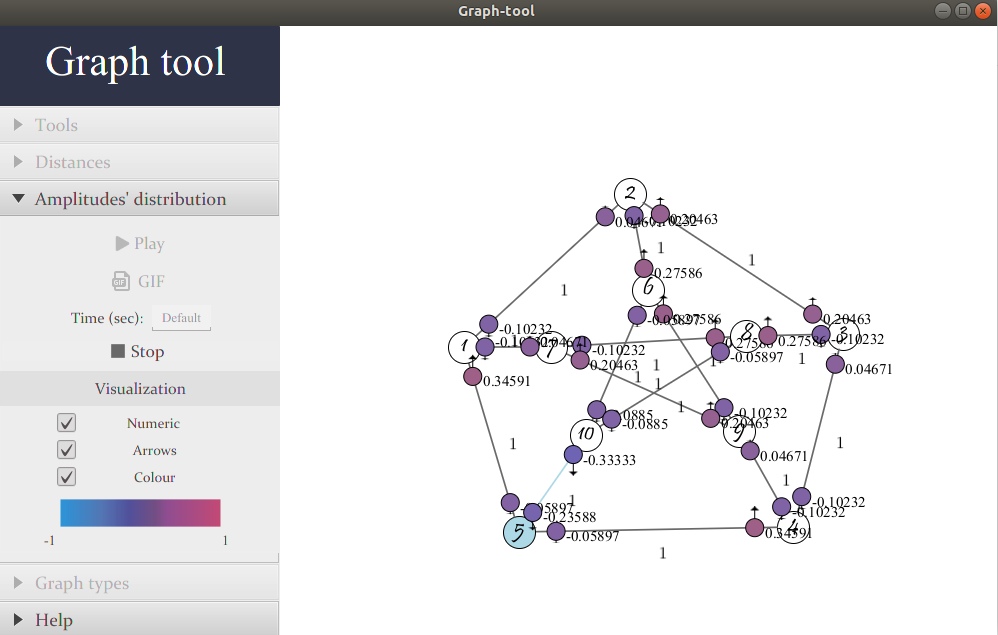


*Рисунок 12. Полный граф, 4 вершины*



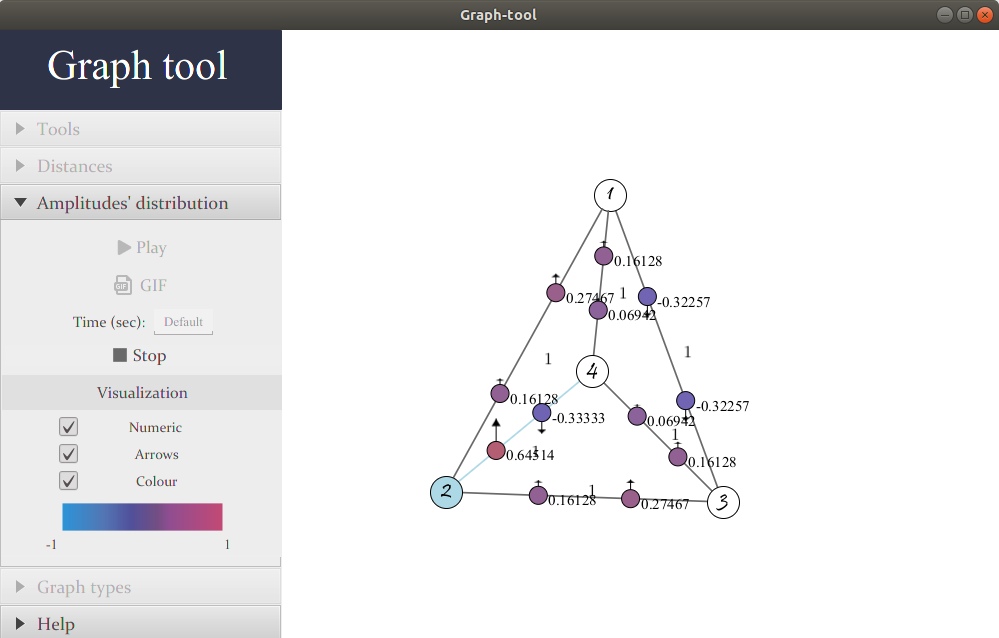
*Рисунок 13. Полный граф, 6 вершин*

1. Граф Петерсена – неориентированный граф с 10 вершинами и 15 ребрами. Испытания алгоритма на данном графе показали, что распределение амплитуд не периодично (рис. 14)

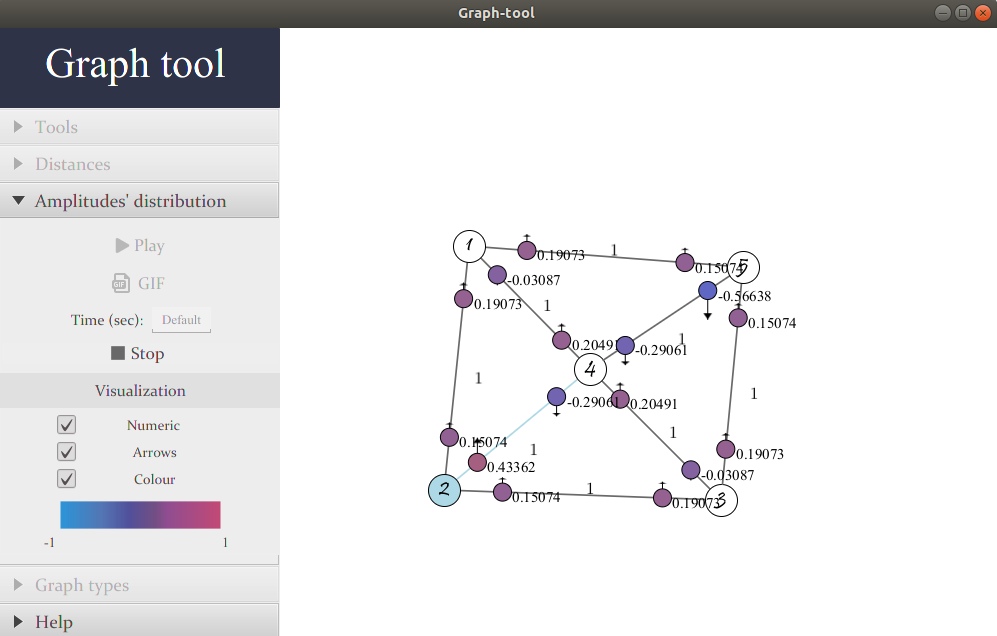


*Рисунок 14. Граф Петерсена*

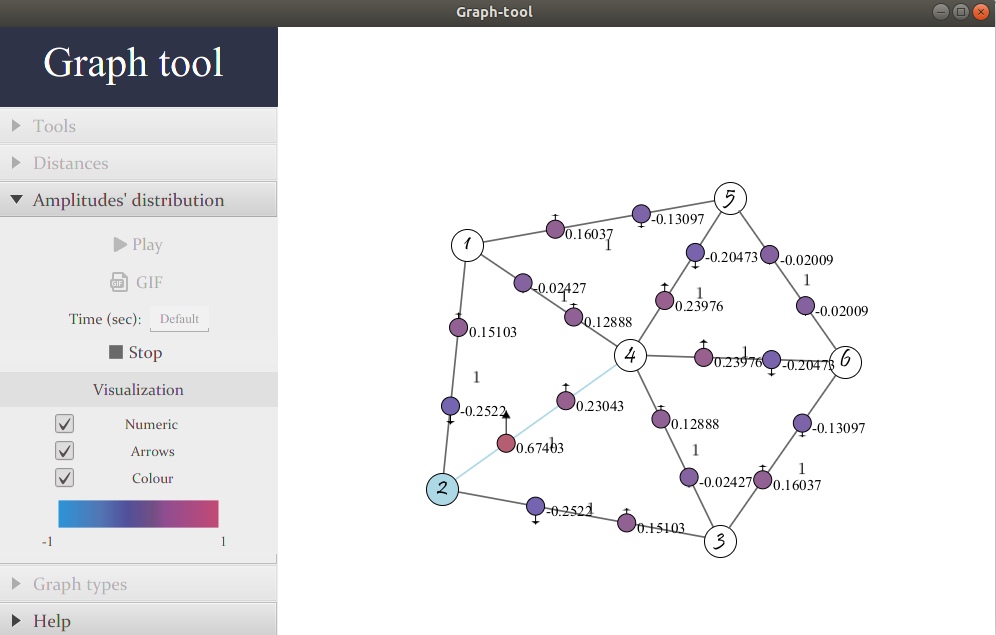
1. Граф-колесо – граф с числом вершин не менее 4-х, где центральная вершина соединена со всеми остальными, а другие, помимо этого, соединены еще с 2мя ближайшими соседями. Алгоритм визуализации амплитуд не периодичен на данном типе графа с числом вершин 4, 5, 6 (рис. 15,16,17)



*Рисунок 15. Граф-колесо, n=4*



*Рисунок 16. Граф-колесо, n=5*



*Рисунок 17. Граф-колесо, n=6*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | Подп. | Дата |
| Изм. | Измененных | Замененных | Новых | Аннулированых |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

# **ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**