|  |  |  |  |
| --- | --- | --- | --- |
| **УТВЕРЖДЕНО**  **RU.17701729.04.01-01 РО 01–1-ЛУ** | |  |  |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл*** | **RU.17701729.04.01-01 РО 01-1** | | **ПРИЛОЖЕНИЕ ДЛЯ ВИЗУАЛИЗАЦИИ**  **АЛГОРИТМОВ ИЛИ СТРУКТУР ДАННЫХ**  **Руководство оператора**  **RU.17701729.04.01-01 РО 01–1**  **Листов 10** | | | | |
|  |  | | | |
|  | | | |
|  | | | | |

**2022**

|  |  |  |
| --- | --- | --- |
|  |  |  |

СОДЕРЖАНИЕ

[1. Назначение программы 3](#_Toc72226576)

[**1.1 Функциональное назначение** 3](#_Toc72226577)

[**1.2 Эксплуатационное назначение** 3](#_Toc72226579)

[2. Условия выполнения программы 3](#_Toc72226580)

[**2.1 Минимальный состав аппаратных средств** 3](#_Toc72226581)

[**2.2 Минимальный состав программных средств** 3](#_Toc72226582)

[**2.3 Требования к персоналу (пользователю)** 3](#_Toc72226583)

[3. Выполнение программы 3](#_Toc72226584)

[**3.1 Установка программы** 3](#_Toc72226585)

[**3.2 Запуск программы и работа с приложением** 3](#_Toc72226586)

[**3.4 Описание игрового процесса** 8](#_Toc72226587)

[**3.4 Управление** 8](#_Toc72226588)

[ПРИЛОЖЕНИЕ 1 9](#_Toc72226589)

[ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 10](#_Toc72226590)

# **1. Назначение программы**

## **1.1 Функциональное назначение**

Приложение позволяет строить графы и решать их с помощью алгоритмов поиска мостов и точек сочленения. Приложение также предоставляет пошаговое интерактивное решение графов с подсказками.

## **1.2 Эксплуатационное назначение**

Программа предназначена для школьников старших классов и студентов, заинтересованных в изучении алгоритмов поиска мостов и точек сочленения, а также и других алгоритмов, так как в будущем программа будет обновляться. Предусмотрено, что программа будет использоваться в тех случаях, если пользователь захочет разобраться в алгоритме поиска мостов и точек сочленения(могут быть добавлены другие алгоритмы) в графе. К дополнению пользователь сможет разобрать свой конкретный случай, путем построения своего графа, что тоже сильно поднимет уровень понимания работы алгоритма.

# **2. Условия выполнения программы**

## **2.1 Минимальный состав аппаратных средств**

1) Персональный компьютер способный поддерживать стабильную работу последней версии браузера Google Chrome.

## **2.2 Минимальный состав программных средств**

1) Последняя версия браузера Google Chrome

## **2.3 Требования к персоналу (пользователю)**

Программа предназначена для пользователей, без специальной подготовки, с целью помощи при прохождении тем с графами школьниками или студентами (в частности, алгоритма поиска мостов и точек сочленения в графе), . Для эксплуатации требуется устройство, удовлетворяющее требуемым техническим характеристикам.

# **3. Выполнение программы**

В данном разделе описан пример работы с программой.

## **3.1 Установка программы**

Приложение не требует установки

## **3.2 Запуск программы и работа с приложением**

Приложение запускается при открытии нужной страницы в браузере.

В приложении существуют всего два основных окна. Первое служит путеводной картой, которая содержит в себе общую информацию об алгоритмах, а второе непосредственно представляет из себя графопостроитель, позволяющий гибко создавать графы с возможностью их дальнейшего интерактивного решения (Рисунки 1 и 2). А теперь немного поподробнее.

Изображение выглядит как текст, снимок экрана, ноутбук, электроника

Автоматически созданное описание

Рисунок 1.

Изображение выглядит как текст, ноутбук, электроника, снимок экрана

Автоматически созданное описание

Рисунок 2.

Сперва на обоих рисунках можно заметить стрелочные кнопки на правых и левых частях окон, а также круглые кнопки-индикаторы в нижней центральной части окон (более темная показывает текущую локацию пользователя). Они служат для удобного и быстрого переключения между окнами графопостроителя (Рисунок 3)



Рисунок 3.

Второе окно содержит в себе, основную логику приложения. Начнем с блока ‘Node’ в блоке ‘Инструменты для построения графа’. Данный раздел содержит в себе 2 кнопки и текстовое поле. Первая кнопка ‘CreateDot’, при нажатии на которую сгенерируется дополнительная вершина с названием “Node\_” + первое незанятое натуральное число.

Кнопка ‘DeleteDot’ удаляет вершину с номером содержащийся в текстовом поле (Рисунок 4).

В блоке ‘Edge’ содержатся 2 объекта типа ‘Select’ и две кнопки ‘CreateEdge/DeleteEdge’.

Предполагается, что пользователь выберет, между какими вершинами стоит добавить ребро, а между каким удалить . (Рисунок 4).

Блок ‘graphSolver’ позволяет получить мгновенное решение к построенному графу. Также оно открывает доступ к двум другим блокам ‘BridgeSolveMap’ и ‘APSolveMap’ . (Рисунок 4).

В блоке GraphGeneration пользователь может задать количество ребер и вершин для генерации произвольного графа (стоит искусственный предел в 30 вершин). Кроме того, есть возможность создать графы, предусмотренные разработчиком . (Рисунок 4).

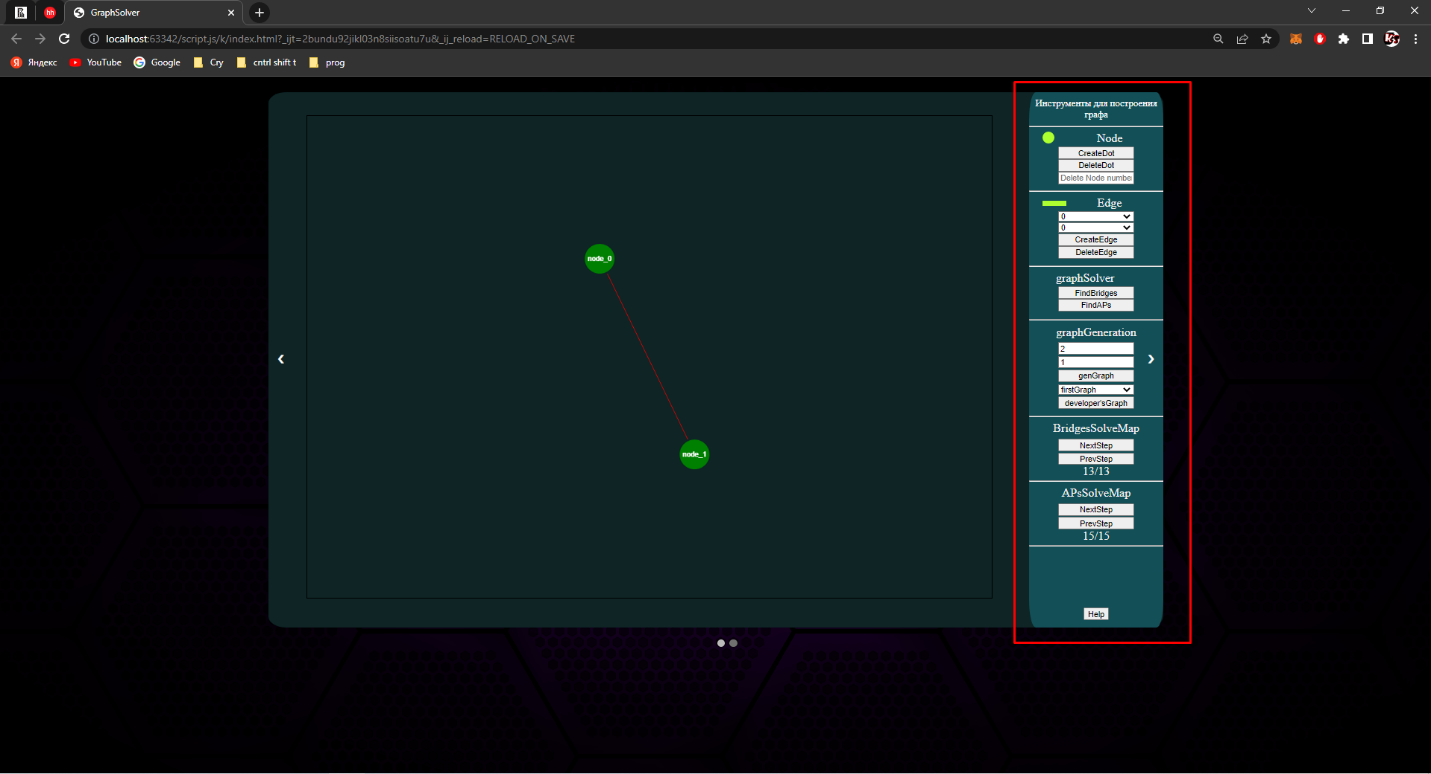


Рисунок 4.

В блоках ‘BridgeSolveMap’ и ‘APSolveMap’ у пользователя появляется возможность пройтись по всем шагам решения алгоритма (при малейшем изменении графа, данная возможность пропадает). Для удобства был предусмотрен счетчик, показывающий текущее положения в решении графа алгоритмом. При прохождении по алгоритму точек сочленения, пройденные точки окрашиваются в синий, если они не являются точками сочленения, в противном случае окрашиваются в желтый. При прохождении по алгоритму нахождения мостов в графе, пройденные мосты окрашиваются в красный, если они являются мостами, в противном случае окрашиваются в синий. Если же он остается белым, то он не был посещен и не является мостом (не был посещен из-за того, что мост ведет в посещенную вершину). Также была добавлена кнопка ‘Help’ для получения помощи с работой интерфейса (рисунки 4,5,6).

Изображение выглядит как текст, легкий

Автоматически созданное описание

Рисунок 5.

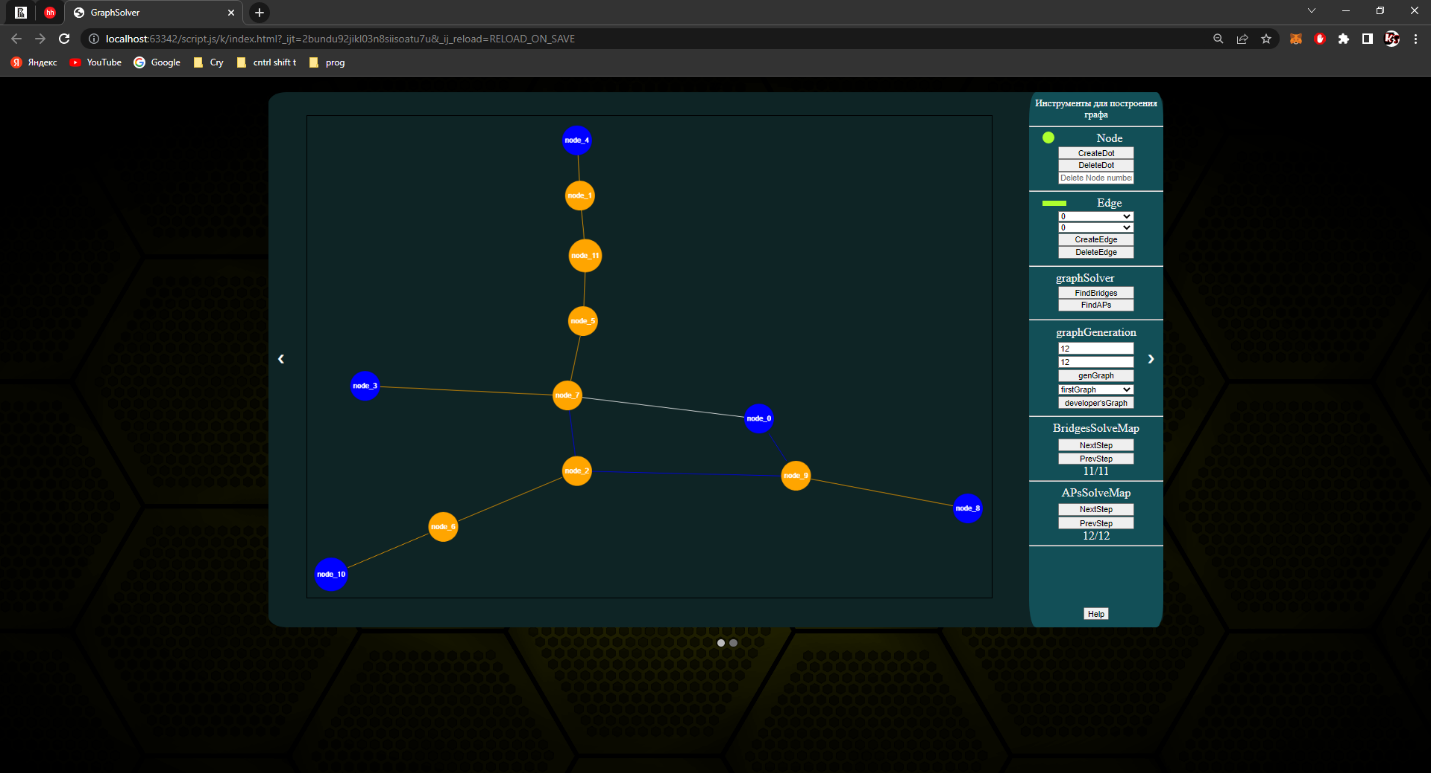


Рисунок 6.

Также пользователю периодически будут поступать подсказки пример см. на рисунке 7.

Изображение выглядит как текст, электроника, компьютер, снимок экрана

Автоматически созданное описание

## **3.3 Описание процесса использования**

В начале работы программы пользователь попадает в первое приветственное окно приложения с не ярким, темным интерфейсом (дизайн выполнен в таком формате, чтобы не казаться едким-ослепительным и навязчивым), в нем написан ознакомительный текст, с краткой общей теорией об алгоритме. Далее пользователю предлагается перейти в следующее окно приложения(статическая подсказка в виде текста с ярким цветом). Для удобства перехода в следующее окно приложения было решено добавить некоторые индикаторы, счетчик страницы (в первом окне будет написано ½) и интерактивные кружки (более темный показывает текущее положение на сайте), а также интерактивные статические стрелочки, при наведении на которые они мягко меняют цвет на черный, тем самым, предлагая пользователю, перейти к окну построения графа с последующим его интерактивным решением.

При переходе на второй экран приложения пользователю предоставляется удобный и в то же время простой интерфейс построения графа. Для построения графа предлагается использовать панель инструментов, которая находится в правой части окна программы. Чтобы пройтись по шагам алгоритма, следует решить его с помощью 2 кнопок блока ‘GraphSolver’. Блок ‘Node’ нужен для построения/удаления вершины, блок 'Edge' нужен для построения/удаления ребра, блок 'GenGraph' нужен для построения произвольного графа и для построения графов, задуманных разработчиком. В случае неправильных вводов данных для построения графа, пользователю будут даваться поясняющие подсказки, а также существует кнопка “help”, расположенная в правой нижней части приложения, с общей поясняющей информацией о работе приложения. Важным замечанием будет то, что нельзя пройтись по алгоритму, после малейших изменениях в графе. Для того чтобы, просмотреть путь алгоритма, нужно зафиксировать его решение, путем нажатия кнопок в блоке ‘GraphSolver’ (FindBridges и FindAPs). Если их нажать, то ниже появятся блоки для интерактивного просматривания ‘BridgesSolveMap’ и ‘APsSolveMap’ соответственно.

## **3.4 Управление**

Управление осуществляется путем нажатия на кнопки для перехода между окнами приложения и нажатиями кнопок на панели инструментов для построения и решения графа.

# ПРИЛОЖЕНИЕ 1

**ТЕРМИНОЛОГИЯ**

Arbor.js — это библиотека визуализации графов, созданная с помощью веб-воркеров и jQuery. Вместо того, чтобы пытаться быть всеобъемлющей структурой, arbor предоставляет эффективный, принудительно управляемый алгоритм компоновки, а также абстракции для организации графа и обработки обновления экрана.

# **ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

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