**Государственное бюджетное образовательное учреждение**

**средняя образовательная школа № 315 Пушкинского района Санкт-Петербурга**

**Пояснительная записка к проекту**

**по теме:**

**«Визуализация графов»**

**Выполнил:**

ученик Яндекс.Лицея

Стафеев Иван

**Проверил:** Бутинов

Геннадий Витальевич

г. Санкт-Петербург - Павловск

2021 г.

**Название проекта:** Визуализатор графов.

**Авторы проекта:** Стафеев Иван Алексеевич.

**Идея:** создать приложение для визуализации математических графов и нахождения их свойств на языке программирования Python.

**Цель:** создать приложение для создания, редактирования, удаления, визуализации математических графов и определения их свойств.

**Задачи:** 1) изучить теорию графов, чтобы иметь представление о способах их представления и определение их свойств; 2) изучить существующие библиотеки языка программирования Python по работе с графами; 3) определить среди библиотек наиболее простую в плане понимания предоставляемых функций и в плане вывода изображения на экран; 4) определить наиболее удобный способ работы пользователя с графами (создание, редактирование и удаление); 5) создать приложение-визуализатор графов с использованием средств библиотеки PyQt5 (как имеющей наибольший функционал среди библиотек на работу с GUI).

**Использованные методы проектной деятельности:** исследовательский, эвристический, дизайн-анализ.

**Реализация:**

**Среди использованных библиотек Python:** **PyQt5** (для создания графического пользовательского интерфейса), **matplotlib** (для создания фигур графов), **networkx** (для создания графов), **math** (для создания некоторых функций определения свойств графов), **colour** (для работы с цветом, конкретнее – для раскрашивания вершин графов в зависимости от количества их связей с другими вершинами), **os** (для получения путей к файлам), **webbrowser** (для открытия html-файлов в браузере), **sqlite3** (для работы с базами данных), **PIL** и **io** (для перевода фигур графов в поток байт и последующего его перевода в изображение), **sys**, **pyperclip** (для сохранения текста в буфер обмена пользователя), **csv** (для работы с csv-таблицами), **ctypes** (для получения информации о разрешении экрана пользователя).

**В ходе создания приложения были реализованы следующие функции:**

1. Создание баз данных для хранения графов

2. Создание, редактирование и удаление таблиц из базы данных

3. Рисование созданных пользователем графов

4. Сохранение изображения нарисованного графа

5. Определение и просмотр свойств графов

6. Сохранение свойств об одном/нескольких/всех графах в базе данных в

форматах txt или csv.

**Классы:**

* **class GraphViz(QMainWindow):** главное окно, через которое осуществляется управление базами данных и их таблицами и которое является собственно визуализатором созданных пользователем графов.
* **class Graph(QWidget):** окно представления графа в виде списка ребер (таблица из базы данных), через которое осуществляется редактирование графа
* **class GraphProperties(QWidget):** окно свойств выбранного графа, предоставляющее пользователю возможность просматривать эти свойства и копировать их буфер обмена
* **class GetGraphName(QDialog):** форма для выбора графа для открытия с возможностью поиска по имени
* **class GetRibs(QDialog):** форма для ввода ребер графа пользователем
* **class GetCsv(QDialog):** форма для получения csv-таблицы с данными ребер
* **class SaveAsForm(QDialog):** форма для выбора свойств графов для сохранения и самих графов

**Функции вне классов:**

* **screen\_size -** функция для получения разрешения экрана пользователя
* **ford\_algotithm** – функция, реализующая алгоритм Форда-Беллмана для поиска минимальных путей в графе
* **dfs\_time\_out –** функция топологической сортировки вершин графа
* **dfs\_create\_comp** – функция обхода графа в глубину с построением компонент сильной связности
* **dfs\_bridges -** функция обхода графа в глубину с поиском мостов

**Константы:**

* **WIDTH, HEIGHT –** ширина и высота экрана пользователя (определяются соответствующей функцией)
* **PROPS –** словарь свойств графа, где ключами являются названия свойств (из выпадающего списка при сохранении), а ключами – названия столбцов таблицы some\_info в БД, соответствующие свойствам
* **HEIGHT\_MAIN\_WINDOW –** высота и ширина приложения; рассчитывается в зависимости от размеров экрана

**Использования баз данных:** базы данных в проекте используются как системы, хранящие неограниченное число графов и информацию о них. Пользователь вправе создавать, редактировать и удалять таблицы в БД в неограниченном количестве, так как каждая созданная пользователем таблица является списком ребер графа. Главная таблицы любой созданной БД хранит информацию о каждом созданном графе (данные в ней обновляются в зависимости от действий пользователя), с ней пользователь не может никак иначе взаимодействовать, кроме как читать информацию из нее. Почему установлена именно такая связь таблиц в БД (одна таблица связывает все остальные, которые между собой не связаны)? Используя эту связь, удобно сохранять общую информацию, так как она хранится в одной таблице, и также удобно редактировать таблицы-графы, так как работа всего с тремя колонками данных позволяет не допустить каких-либо ошибок.

**Практическая польза:**

В современном мире графы применяются во многих областях жизни: математика, химия, биология, физика, социология, Интернет, авиация, астрономия и многие другие. В перспективе проект можно использовать во многих сферах, где имеет важное значение именно представление каких-либо систем со связями (по-другому, графов). Например, с помощью приложения можно будет создавать карты созвездий, схемы железных дорог, авиалиний, социограммы, развивать теории позиционных игр.

На данных момент развития проекта я вижу его использование в учебных целях, например, на уроках информатики при изучении графов, их способов представления, свойств, алгоритмов обхода и поиска оптимальных путей.

**Преимущества приложения:**

1. Приложение обладаетпростым интерфейсом, с которым легко разобраться даже самостоятельно, но также существует руководство.

2. Приложение имеет несколько способов добавления информации и ее сохранения.

3. Удобно представление информации о графах.

4. Наличие подсказок для работы с приложением.

**Недостатки приложения:**

1. На данный момент приложение способно только отображать граф. Хотелось бы еще добавить возможность отображать на графе его циклы, мосты, точки сочленения, отображать работу алгоритмов обхода и поиска оптимальный путей.

2. Все свойства графа пока можно исследовать в текстовом виде.

3. Работа с тремя окнами может быть неудобна для некоторых пользователей. Возможно, размещение этих окон в одном будет удобнее.

**Перспектива развития:**

В будущем хотелось бы реализовать визуализацию мостов, точек сочленения и т.п. и алгоритмов обхода графа и поиска оптимального пути прямо на графе. Также отличным дополнением будет еще некоторое количество свойств графов (конденсация, например)

Если говорить о судьбе приложения, то скорее всего я вижу его как утилиту в первую очередь для учебы, подготовки к олимпиадам по информатике, где графы часто задействованы. Если значительно расширить функции приложения, его, возможно, можно будет использовать в промышленных целях.