МУНИЦИПАЛЬНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ «СРЕДНЯЯ ШКОЛА № 56 КИРОВСКОГО РАЙОНА ВОЛГОГРАДА»

Проект на тему: Программа для создания графиков функций.

Проект выполнил: Бондаренко Александр Алексеевич,

обучающийся 10 класса МОУ СШ № 56

Руководитель проекта: Яровая Е.В.,

учитель информатики МОУ СШ № 56

Волгоград, 2024.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ……………………………………………………………………………...3

1. Глава 1. Создание линейной функции с помощью библиотеки matplotlib……..…4

2. Глава 2.

Тема моего индивидуального проекта – «Программа для создания графиков функций.» тесно связана с моей будущей профессией, а так же с предметом информатика.

Использование программ для создания графиков функций остается актуальным и востребованным в современном мире в различных областях. Этот инструмент оказывается полезным для научных исследований, образования и визуализации математических концепций. Вот несколько точек, подчеркивающих актуальность темы:

1. **Визуализация математических моделей:** Программы для создания графиков функций обеспечивают возможность визуализации математических моделей, что является ключевым элементом в исследованиях и обучении. Это помогает лучше понять формулы, функции и их взаимосвязи.
2. **Обучение математике:** В образовательных целях такие программы играют важную роль. Студенты могут использовать графики функций для изучения и применения математических концепций, что способствует лучшему усвоению материала.
3. **Анализ данных и статистика:** Графики функций являются эффективным средством для визуального анализа данных и выявления закономерностей, например, в научных исследованиях, где требуется визуализация результатов.
4. **Оптимизация процессов:** Программы для создания графиков функций могут быть использованы для оптимизации различных процессов, включая инженерные расчеты и проектирование.
5. **Разработка и тестирование алгоритмов:** Специалисты по разработке и тестированию алгоритмов могут использовать графики функций для визуального анализа результатов работы алгоритмов, что облегчает процесс отладки и оптимизации.

Задачи:

**1. Создание интерфейса пользователя:**

**Разработать графический интерфейс с использованием библиотеки customtkinter.**

**Добавить элементы управления, такие как текстовые поля (CTkEntry), выпадающий список (CTkComboBox), кнопку (CTkButton) и метку (CTkLabel).**

**2. Интеграция с библиотеками:**

**Импортировать необходимые библиотеки, такие как customtkinter, PIL, и functions.**

**Подключить функциональность графического отображения с использованием библиотеки matplotlib.**

**3. Обработка выбора пользователя:**

**Реализовать функцию combobox\_callback, которая реагирует на выбор пользователя в выпадающем списке.**

**Добавить логику для скрытия/показа соответствующих элементов интерфейса в зависимости от выбранного варианта.**

**4. Темы оформления:**

**Реализовать функцию select\_theme, которая изменяет тему оформления в зависимости от выбранной пользователем темы (светлая/темная).**

**5. Отображение графика:**

**Реализовать функцию submit, которая обрабатывает введенные пользователем данные и вызывает соответствующую функцию из модуля functions.**

**Обновить отображение графика после нажатия кнопки "Submit".**

**6. Функциональность построения графика:**

**В модуле functions реализовать функции для построения графиков функций.Обеспечить сохранение графика в файл «.png».**

**7. Обработка ошибок и исключений:**

**Добавить проверки на корректность введенных данных (например, числовые значения, наличие необходимых файлов) и предусмотреть обработку возможных ошибок.**

**8. Документация:**

**Прокомментировать код, чтобы облегчить его понимание и сопровождение.**

**Добавить описание к функциям и ключевым частям кода.**

**9.Оптимизация кода:**

**Провести оптимизацию кода для повышения эффективности и улучшения читаемости.**

**Помните, что эти задачи являются общими указаниями, и вы можете дополнить их в соответствии с требованиями вашего проекта и вашими целями.**

**Глава 1: Технологический стек**

**Глава 1.1: Язык программирования**

**• Мы создали наш проект, используя язык программирования Python. Этот язык простой в использовании и позволяет делать много разных вещей. Глава 1.2: Библиотека customtkinter**

**• Мы использовали библиотеку customtkinter для того, чтобы сделать наше приложение красивым и удобным для использования. Она помогла нам сделать стандартные кнопки и окна более интересными.**

**Глава 1.3:**

**Библиотека matplotlib • Чтобы рисовать красивые графики, мы использовали библиотеку matplotlib. Она дала нам инструменты для создания разных видов графиков. Глава 1.4: Библиотека PIL (Python Imaging Library) • Для сохранения наших графиков в картинку мы использовали библиотеку PIL. Она помогла нам сохранять результаты нашей работы.**

Глава 2:

Реализация основных функций

Глава 2.1:

Выбор функции • Мы добавили функцию, которая позволяет вам выбрать, какой вид математической функции вы хотите увидеть. Просто выберите из списка.

Глава 2.2:

Ввод параметров • Если у вас есть какие-то числа или значения, которые вы хотите использовать в графике, вы можете ввести их. Например, если у вас есть уравнение для прямой линии, вы можете ввести коэффициенты здесь. Глава 2.3: Темная/Светлая тема • Вы можете выбрать, какого цвета будет окно с графиком. Если вам нравятся светлые цвета, выберите светлую тему. Если темные, выберите темную. Глава 2.4: Отображение графика • Когда вы выбрали функцию и ввели параметры, просто нажмите кнопку "Submit". Мы построим график для вас и покажем его на экране. Глава 2.5: Сохранение графика • Мы добавили кнопку, которая позволяет сохранить ваш график в файл. Это может быть полезно, если вы хотите поделиться им с кем-то или использовать позже.

Глава 3: Зависимости и требования Глава 3.1: Python версии 3.x • Просто убедитесь, что у вас установлена Python версии 3.x. Это несложно, и многие программы используют его. Глава 3.2: Библиотеки • Также вам нужно установить несколько библиотек: customtkinter, PIL, matplotlib. Не беспокойтесь, это просто шаги, которые помогут вашей программе работать.