Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Факультет електроніки та комп’ютерних технологій

Кафедра радіофізики та комп’ютерних технологій

КУРСОВА РОБОТА

на тему: «Розробка інтерпретатора та графічного інтерфейсу для побудови графіків функцій»

Виконав ст. 2 курсу

гр. ФЕІ-25с

\_\_\_\_\_\_\_\_\_\_Тєльний Б.

Науковий керівник

\_\_\_\_\_\_\_\_\_\_доц. Хвищун І. О.

Львів 2024

**ЗМІСТ**

ВСТУП

РОЗДІЛ 1 ТЕХНІЧНІ ЗАСОБИ

* 1. Інтерпретована об’єктно-орієнтована мова програмування Python

Python — це високорівнева мова програмування загального призначення, створена Гвідо ван Россумом у 1991 році. Вона вирізняється простим і зрозумілим синтаксисом, який забезпечує високу читабельність коду, та широким спектром застосувань, включаючи веб-розробку, аналіз даних, автоматизацію, машинне навчання тощо.

Python має низку ключових характеристик, які виділяють її серед інших мов програмування:

* код Python виконується рядок за рядком за допомогою інтерпретатора, що спрощує налагодження та дозволяє уникати компіляції.
* Python підтримує об'єктно-орієнтоване програмування (ООП), що дозволяє структурувати код у вигляді класів і об'єктів.
* програми на Python можуть виконуватися на різних операційних системах, таких як Windows, macOS, Linux.
* Python постачається з безліччю вбудованих модулів для роботи.
* синтаксис Python інтуїтивно зрозумілий, що полегшує сприйняття коду.

Мова Python використовується у багатьох галузях зокрема і для візуалізації та аналізу даних, завдяки таким бібліотекам, як NumPy та Matplotlib. [1]

Переваги Python:

* Легкий та інтуїтивно зрозумілий код.
* Велика кількість навчальних матеріалів і активна спільнота розробників.
* Універсальність і багато вбудованих інструментів.
* Динамічна типізація та автоматичне керування пам’яттю. [2]

Недоліки Python:

* Порівняно низька швидкість виконання, порівняно з компільованими мовами програмування. [2]
* Відсутність суворої типізації
  1. Бібліотеки
     1. Tkinter

Tkinter — це інтерфейс до інструментарію GUI Tk, ця бібліотека є стандартною Python, тому не потребує встановлення. Tk не є окремою бібліотекою, а складається з кількох окремих модулів, кожен з яких має свої функції та власну офіційну документацію.

Tkinter має кілька ключових особливостей, які роблять його зручним для створення простих та ефективних графічних інтерфейсів:

* Tkinter має інтуїтивно зрозумілий API.
* доступний на більшості платформ Unix, включаючи macOS, а також у системах Windows. [3]
* бібліотека надає інструменти для створення різних елементів інтерфейсу, таких як кнопки, текстові поля, списки, тощо.
* Tkinter постачається разом із Python і не потребує встановлення додаткових модулів для базової роботи.

Основним елементом є головне вікно програми, яке служить базою для розміщення всіх інших компонентів.

Tkinter надає широкий набір віджетів, серед яких:

* Button — кнопки.
* Label — текстові або графічні підписи.
* Entry — однорядкові текстові поля для вводу.

Для розташування віджетів у вікні tkinter використовує три менеджери компоновки:

1. pack() — розташування елементів у вертикальному або горизонтальному напрямку.
2. grid() — розташування елементів у вигляді таблиці.
3. place() — точне позиціонування елементів за координатами.
   * 1. Ttkbootstrap

Ttkbootstrap — це бібліотека для створення сучасних і стильних графічних інтерфейсів користувача (GUI) на Python, що є надбудовою над стандартною бібліотекою Tkinter і використовує віджети з модулю ttk (themed Tkinter). Ttkbootstrap надає велику кількість попередньо налаштованих стилів для віджетів, що дозволяє значно покращити зовнішній вигляд додатків, створених за допомогою Tkinter. Завдяки цьому можна створювати привабливий інтерфейс та вручну налаштовувати стилі.

Ttkbootstrap розширює можливості стандартних віджетів Tkinter та додає наступні переваги:

* стильний, сучасний вигляд віджетів, що нагадує популярні дизайнерські бібліотеки, такі як Bootstrap.
* щоб почати використовувати Ttkbootstrap, достатньо мінімальних змін у коді Tkinter. Це робить бібліотеку дуже зручною для розробників, знайомих з Tkinter. [4]
* підтримує багато стилів, кольорових схем та компонентів, що дозволяє легко налаштовувати інтерфейс.
* є попередньо налаштовані теми, що дозволяє змінювати вигляд додатку всього за кілька рядків коду.
  + 1. Matplotlib

Matplotlib — це одна з найпопулярніших бібліотек Python для візуалізації даних. Вона надає великий набір інструментів для створення різноманітних графіків і діаграм, таких як лінійні графіки, гістограми, діаграми розсіяння, графіки функцій, 3D-графіки та багато іншого. Бібліотека є потужним інструментом для аналізу даних і використовується в наукових, інженерних та статистичних розробках.

Бібліотека надає безліч параметрів для налаштування вигляду графіків. Можна змінювати шрифти для всіх текстових елементів. Задавати колір графіка, фону, осей та інших елементів. Кожну лінію можна стилізувати відповідно до потреб.

Основні компоненти Matplotlib**:**

* Figure: Це основний контейнер, що містить один або більше графіків.
* Axes: Це область в межах Figure, на якій безпосередньо будується графік. Один Figure може містити кілька осей для різних графіків.
* Plot: Це безпосередньо графік або лінія, яка відображається на осі (Axes). Наприклад, лінійний графік або точковий графік.

Переваги Matplotlib:

* можна детально налаштовувати зовнішній вигляд графіків, додаючи різноманітні елементи та змінюючи кольори, шрифти, стилі тощо.
* Matplotlib може працювати з іншими бібліотеками, такими як Ttkbootstrap для відображення графіка безпосередньо на головному вікні програми.

Недоліки Matplotlib

* може здаватися складною через безліч параметрів налаштувань.
* досить повільно працює на великих обсягах даних.

РОЗДІЛ 2 ПРОГРАМНЕ СЕРЕДОВИЩЕ

* 1. Середовище розробки Visual Studio Code

Visual Studio Code (VS Code) — це безкоштовне, потужне і легке середовище розробки, яке підтримує різні мови програмування та інструменти для розробки. Воно користується великою популярністю серед розробників завдяки своїм можливостям, гнучкості та великій кількості плагінів, що дозволяють налаштувати робоче середовище під будь-які потреби.

Було обрано саме це середовище розробки, через наступні переваги:

* VS Code є набагато легшим за багато інших, таких як PyCharm чи Eclipse. Це забезпечує швидший запуск і менше навантаження на систему.
* середовище підтримує розширення, що значно полегшує процес написання коду. Наприклад, для Python є розширення Python, яке забезпечує автодоповнення, дебаггер, інтеграцію з Jupyter Notebook, а також підтримку віртуальних середовищ.
* має вбудовану підтримку для роботи з системами контролю версій. Можна виконувати основні операції безпосередньо в редакторі.
* вбудований термінал дозволяє запускати командний рядок без виходу з редактора. Це зручний інструмент для виконання команд без необхідності відкривати окреме вікно терміналу.
* має вбудований дебаггер, який дозволяє зручно відлагоджувати код за допомогою точок зупинки, перегляду змінних, стека викликів тощо.
* VS Code може допомогти в написанні коду, підказуючи синтаксис, методи та функції для певних об'єктів, класів або бібліотек.

РОЗДІЛ 3 ПРОЄКТУВАННЯ ПРОГРАМИ

* 1. Stack

Стек — це структура даних, що працює за принципом "останній прийшов — перший вийшов" (LIFO, Last In First Out). Це означає, що елементи додаються в структуру даних тільки в один кінець (верх стека) і видаляються також з цього ж кінця.

Основні операції, які підтримуються стеком:

* Push() – додає елемент на верхівку стека
* Pop() – видаляє елемент з верхівки стека
* Empty() – повертає True, якщо стек порожній
  1. Зворотний польський запис

Зворотний польський запис — це форма математичного запису, у якій оператори записуються після операндів. Цей підхід дозволяє обчислювати математичні вирази без використання дужок і пріоритетів операцій, що робить його зручним для комп'ютерної реалізації.

Зворотний польський запис був розроблений у середині 1950-х років австралійським філософом і спеціалістом з теорії обчислювальних машин Чарлзом Гембліном. Він базувався на польській нотації, яку ще у 1920 році запропонував польський математик Ян Лукашевич. Результати роботи Гембліна були представлені на конференції в червні 1957 року, а пізніше опубліковані в 1957 і 1962 роках. [6]

У загальному вигляді зворотний польський запис має наступну структуру:

1. Запис операцій складається з послідовності операндів та знаків операцій. При письмовому оформленні виразу операнди розділяються пробілами.
2. Обчислення виразу відбувається зліва направо.
3. Коли у виразі зустрічається знак операції, виконується відповідна дія над двома останніми операндами, які йдуть перед цим знаком, у порядку їх запису.
4. Результат операції замінює у виразі її операнди та знак, після чого обчислення продовжується за тим самим алгоритмом.
5. Підсумковим результатом є значення останньої виконаної операції.

Так як людині простіше сприймати інфіксну форму запису (оператор ставиться між операндами), потрібно вміти переводити запис у такій формі до вигляду зворотного польського запису. Алгоритм був вигаданий Едсгером Дейкстрою та вперше опублікований в 1961 р. Алгоритм базується на структурі даних stack. [7]

Алгорит Дейкстри:

1. Проходимо по кожному елементу рядка.
2. При знаходженні числа додаємо його в рядок відповіді.
3. При знаходженні оператора додаємо його в стек.
4. Додаємо до рядка відповіді всі оператори, пріоритет яких вище, за оператор, що розглядається зараз.
5. Якщо зустрічаємо ‘(‘ понижуємо пріоритет всіх наступних операторів
6. Якщо зустрічаємо ‘)’ виконуємо пункт 4.

Для обчислення значення виразу, записано в зворотній польській формі, теж зручно використовувати stack.

Алгоритм обчислення значення виразу:

1. проходимо по кожному елементу рядка.
2. якщо це число або змінна, то додаємо його в стек.
3. якщо це оператор, то витягаємо з стека два останніх елементи, здійснюємо операцію та додаємо результат в стек.
4. якщо це функція, то витягаємо з стека останній елемент, визначаємо значення функції від цього елемента та додаємо результат в стек.

В кінці стек міститиме один елемент – результат.

* 1. Опис GUI

Для зручної роботи користувача створено графічний інтерфейс за допомогою бібліотек Tkinter та ttkbootstrap. Інтерфейс програми організований таким чином, щоб забезпечити швидкий доступ до основного функціоналу.

Основні компоненти GUI:

1. Введення функції.

Вікно містить текстове поле, де користувач вводить функції

1. Налаштування меж і кроку.

Для побудови графіка користувач має вказати:

* Ліву межу
* Праву межу
* Крок табуляції

1. Натиснувши кнопку «Побудувати графік», користувач запускає алгоритм обчислення і відображення графіка введеної функції. У випадку помилки (наприклад, введення некоректних меж або функції) виводиться повідомлення про помилку.
2. Візуалізація графіка.

Графік функції відображається в центральній частині вікна. Він включає осі координат, підписи осей, назву графіка і сітку для зручності аналізу.

1. Збереження даних.

Користувач може зберегти, у файл формату .dat, табуляцію значень x та y, натиснувши кнопку «Зберегти табуляцію». У цьому файлі міститься індекс точки, координата x та відповідне їй значення y.

Інтерфейс підтримує динамічну взаємодію: введення даних у текстові поля, обробка помилок в реальному часі, збереження результатів у файл для подальшого аналізу.

РОЗДІЛ 4 РЕАЛІЗАЦІЯ ПРОГРАМИ

* 1. Стандартизація вигляду функції
  2. Обробка математичних операцій
  3. Створення графічного інтерфейсу програми

РОЗДІЛ 5 ТЕСТУВАННЯ ПРОГРАМИ

ВИСНОВОК

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. https://docs.python.org/3/
2. <https://drukarnia.com.ua/articles/movi-programuvannya-dlya-vebrozrobki-ta-suputni-tekhnologiyi-wzeAr>
3. <https://docs.python.org/uk/3/library/tkinter.html>
4. <https://ttkbootstrap.readthedocs.io/en/latest/api/>
5. <https://matplotlib.org/stable/tutorials/index.html>
6. Складанюк, Максим (2020). *Арифметичний калькулятор* (Курсова робота) (укр.). Бердичів, Україна.
7. *Stichting Mathematisch Centrum, Dr. E. W. Dijkstra*