

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

Лабораторна робота № 5

з дисципліни «Об'єктивно-орієнтоване проектування
програм для мобільних систем»

Тема: «Розробка графічного інтерфейсу для
розрахункових завдань і побудови графіків»

XAI.301. 174. 322. 5 ЛР

Виконав студент гр. 322

_____ Діхтяренко М.Г.
(підпис, дата) (П.І.Б.)

Перевірив

_____ к.т.н., доц. О. В. Гавриленко
(підпис, дата) (П.І.Б.)

2024

МЕТА РОБОТИ

Застосувати теоретичні знання з основ роботи з бібліотекою tkinter на мові Python, навички використання бібліотеки matplotlib, а також об'єктно-орієнтований підхід до проектування програм, і навчитися розробляти скрипти для інженерних додатків з графічним інтерфейсом.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Описати клас, який реалізує графічний інтерфейс користувача для вирішення розрахункової задачі згідно варіанту (див. табл.1) і скрипт для роботи з об'єктом цього класу. Зазначена у задачі функція повинна бути окремим методом класу.

Завдання 2. Розробити скрипт із графічним інтерфейсом, що виконує наступні функції:

А. установка початкових значень параметрів для побудови графіка (змінні Tkinter)

В. створення текстового файлу з двома стовпцями даних: аргумент і значення функції відповідно до варіанту (див. табл.2). Роздільник в кожному рядку файлу: для парних варіантів – ';', для непарних – '#';

С. зчитування з файлу масивів даних;

Д. підрахунок і відображення мінімального / максимального значення аргументу / функції у зчитаних масивах;

Е. відображення масивів даних за допомогою пакета matplotlib у вигляді графіка функції в декартовій системі координат з назвою функції, позначенням осей, оцифруванням і сіткою;

Ф. заголовок вікна повинен містити текст: lab # - <# групи> -v <# варіанту> - <прізвище> - <ім'я>, наприклад: lab4_2-320-v01-Ivanov-Ivan

ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі 22:

Func22. Описати функцію PowerA3(A), що повертає третій ступінь числа A (A – дійсний параметр). За допомогою цієї функції знайти третій ступінь п'яти даних чисел.

Алгоритм вирішення завдання:

1. Вхідні дані:

Отримати список чисел, наприклад: $[a_1, a_2, \dots, a_n]$.

2. Обчислення третього ступеня числа:

- Прийняти одне число a.
- Обчислити його третій ступінь за формулою: $a^3 = a \times a \times a$.
- Повернути результат.

3. Обчислення для всього списку чисел:

- Для кожного числа зі списку викликати функцію, яка обчислює третій ступінь.
- Зберегти результати обчислень у новий список.

4. Виведення результату:

- Повернути список, що містить треті ступені кожного числа.

Лістинг коду вирішення задачі наведено в дод. А (стор. 6). Екран роботи програми показаний в дод.Б (стор.9).

Завдання 2. Вирішення задачі 22:

$$^{13} \left| \begin{array}{l} y[k+1] = \left(1 - \frac{T_0}{T}\right) \cdot y[k] + \frac{T_0}{T} \cdot K \cdot U \\ \end{array} \right| \begin{array}{l} U[0] = \\ 2 \text{ Вт}, \\ y[0] = 0 \end{array} \left| \begin{array}{l} T = 1,1 \\ K = 2,2 \end{array} \right| \begin{array}{l} y - T, K \\ U - \\ Q_{\text{н}}, \text{Вт} \end{array}$$

Алгоритм вирішення завдання:

1. Вхідні дані:

- U : Вхідний сигнал (постійне значення).

- τ : Постійна часу.
 - k : Коефіцієнт підсилення.
 - T_0 : Період дискретизації.
 - n_{steps} : Кількість кроків.
2. **Ініціалізація початкового значення:**
 - Початкове значення $y[0] = 0$.
 3. **Розрахунок значень $y[k]$:**
 - Для кожного k (від 0 до $n_{steps} - 1$):
 - Обчислити наступне значення $y[k+1]$
 - Додати $y[k+1]$ до списку результатів.
 4. **Збереження результатів:**
 - Отримати список усіх значень $y[k]$, включаючи початкове значення $y[0]$.
 5. **Побудова графіка:**
 - Побудувати графік залежності $y[k]$ від кроку k .
 - Позначити осі:
 - По осі x : Крок k .
 - По осі y : Значення $y[k]$.
 - Додати сітку, легенду та зберегти графік як файл
 6. **Виведення результату:**
 - Відобразити графік для візуалізації зміни $y[k]$ з часом.

Лістинг коду вирішення задачі наведено в дод. А (стор. 6). Екран роботи програми показаний в дод.Б (стор. 9).

ВИСНОВКИ

В результаті виконання Лабораторної роботи на практиці було застосовано теоретичні знання з основ роботи з бібліотекою tkinter на мові Python, закріплені навички використання бібліотеки matplotlib, а також навички розробляти скрипти для інженерних додатків з графічним інтерфейсом.

ДОДАТОК А

Лістинг коду програми до задачі 1

```
def PowerA3(a):
    """
    Обчислює третій ступінь числа А.
    :param a: число
    :return: A^3
    """
    return a ** 3

def calculate_powers(numbers):
    """
    Обчислює третій ступінь для списку чисел.
    :param numbers: список чисел
    :return: список третіх ступенів
    """
    return [PowerA3(num) for num in numbers]
```

Лістинг коду програми до задачі 2

```
import matplotlib.pyplot as plt

def calculate_y(U, T, K, T0, n_steps):
    """
    Розрахунок значень y[k] за заданим рівнянням.
    :param U: Вхідний сигнал
    :param T: Постійна часу
    :param K: Коефіцієнт підсилення
    :param T0: Період дискретизації
    :param n_steps: Кількість кроків
    :return: Список значень y[k]
    """
    y = [0] # Початкове значення y[0]
    for k in range(n_steps):
        next_y = (1 - T0 / T) * y[k] + (T0 / T) * K * U
        y.append(next_y)
    return y

def plot_results(U, T, K, T0, n_steps):
    """
    Побудова графіка залежності y[k].
    """
    y_values = calculate_y(U, T, K, T0, n_steps)
```

```

plt.figure(figsize=(10, 5))
    plt.plot(range(len(y_values)), y_values, marker='o',
label='y[k]')
    plt.title("Залежність y[k] від часу")
    plt.xlabel("Крок (k)")
    plt.ylabel("y[k]")
    plt.grid()
    plt.legend()
    plt.savefig("image.png")
    plt.show()

```

Лістинг коду програми main.py

```

import tkinter as tk
from task1 import calculate_powers
from task2 import plot_results

def task1_menu():
    """
    Задача 1: Обчислення третього ступеня чисел.
    """
    print("Задача 1: Обчислення третього ступеня чисел.")
    try:
        numbers = [float(x) for x in input("Введіть числа через пробіл: ").split()]
        results = calculate_powers(numbers)
        print("Третій ступінь чисел:", results)
    except ValueError:
        print("Помилка: введіть лише числові значення.")

def task2_menu():
    """
    Задача 2: Побудова графіка y[k].
    """
    print("Задача 2: Побудова графіка.")
    try:
        U = float(input("Введіть U: "))
        T = float(input("Введіть T: "))
        K = float(input("Введіть K: "))
        T0 = float(input("Введіть T0: "))
        n_steps = int(input("Введіть кількість кроків: "))
        plot_results(U, T, K, T0, n_steps)
        print("Графік збережено у файлі image.png")
    except ValueError:

```

```
print("Помилка: введіть коректні дані.")

def main_menu():
    """
    Головне меню програми.
    """
    root = tk.Tk() # Ініціалізація графічного вікна
    root.title("lab5_2-322-v13-Dikhtiarenko-Misha")

    while True:
        print("\nВиберіть задачу:")
        print("1. Обчислення третього ступеня чисел")
        print("2. Побудова графіка y[k]")
        print("3. Вийти")

        choice = input("Ваш вибір: ")

        if choice == "1":
            task1_menu()
        elif choice == "2":
            task2_menu()
        elif choice == "3":
            print("Вихід з програми...")
            break
        else:
            print("Невірний вибір. Спробуйте ще раз.")

    root.mainloop()

if __name__ == "__main__":
    main_menu()
```


ДОДАТОК Б

Скрін-шот вікна виконання програми

```
>_ Console x +
Editor
Run
Виберіть задачу:
1. Обчислення третього ступеня чисел
2. Побудова графіка  $y[k]$ 
3. Вийти
Ваш вибір: 1
Задача 1: Обчислення третього ступеня чисел.
Введіть числа через пробіл: 2 3 4
Третій ступінь чисел: [8.0, 27.0, 64.0]

Виберіть задачу:
1. Обчислення третього ступеня чисел
2. Побудова графіка  $y[k]$ 
3. Вийти
Ваш вибір: 2
Задача 2: Побудова графіка.
Введіть U: 4
Введіть T: 6
Введіть K: 8
Введіть  $T_0$ : 3
Введіть кількість кроків: 9
□
```

