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'єктно-орієнтоване програмування СУ»

Тема:«Розробка графічного інтерфейсу для розрахункових завдань

і побудови графіків»

Виконала студентка гр. 320

Костенко Валерія

(підпис, дата) (П.І.Б.)

Перевірив

(підпис, дата)

2023

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.

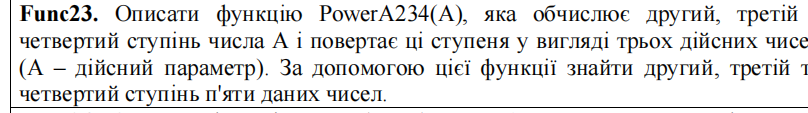
ПОСТАНОВКА ЗАДАЧІ

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:

A. установка початкових значень параметрів для побудови графіка (змінні

Tkinter)

B. 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 – '#';

C. зчитування з файлу масивів даних;

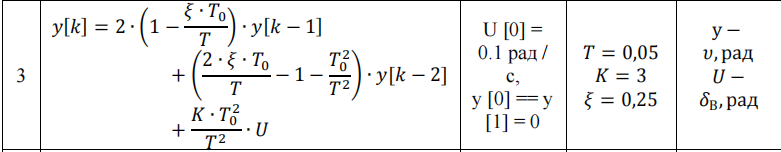
D. підрахунок і відображення мінімального / максимального значення

аргументу / функції у зчитаних масивах;

E. відображення масивів даних за допомогою пакета matplotlib у вигляді

графіка функції в декартовій системі координат з назвою функції,

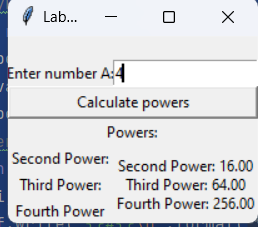
позначенням осей, оцифруванням і сіткою;

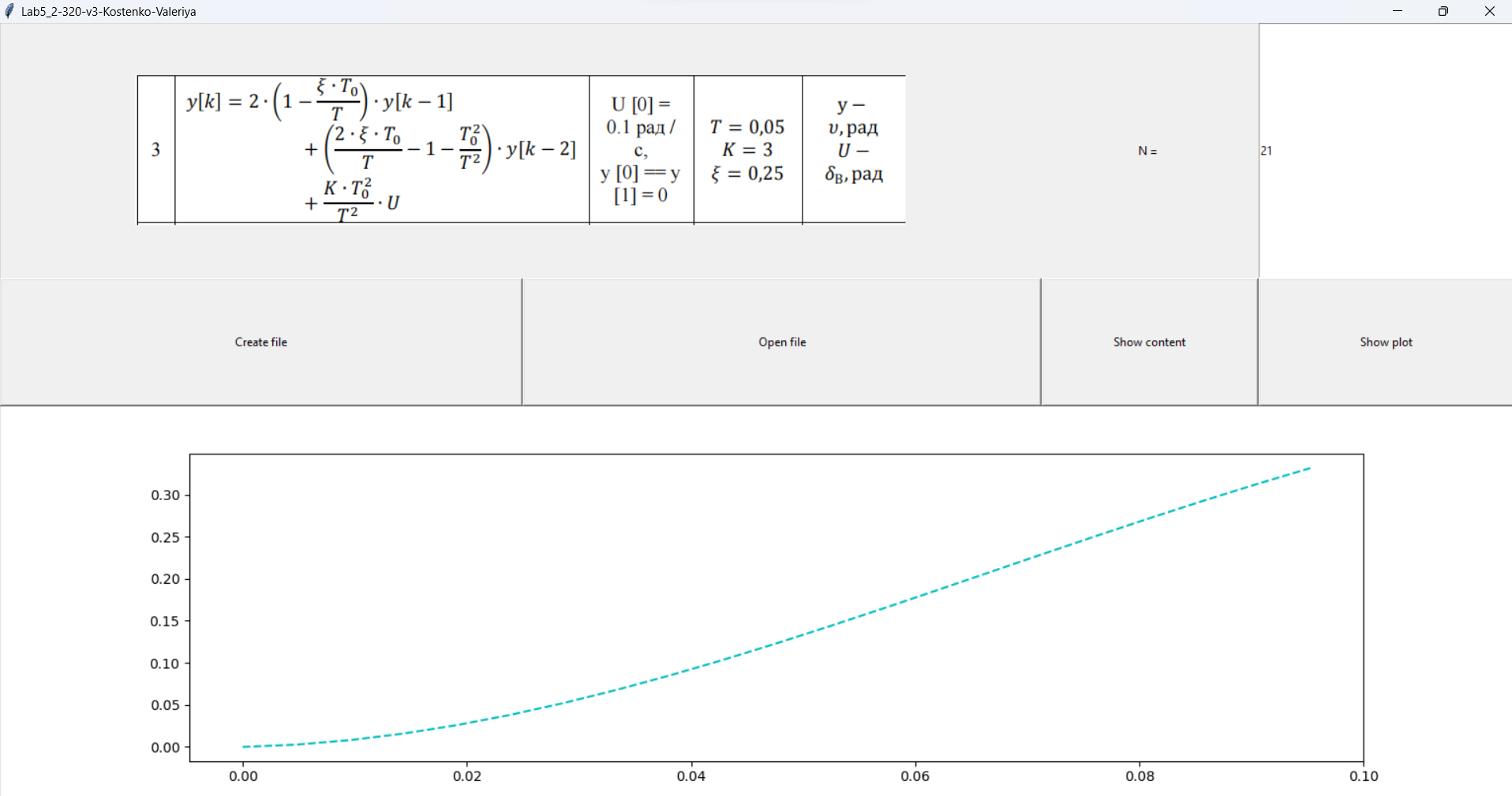


Лістинг коду  
# Підключення створенних вікон  
import tkinter  
from task1 import CalculatorWithPowerA234  
from task2 import Task2Window  
  
# словник для швидкого доступу до відповідної функції виконання  
task\_window\_dict = {  
 "1": (CalculatorWithPowerA234, "Lab5\_1-320-v23-Kostenko-Valeriya", "200x150"),  
 "2": (Task2Window, "Lab5\_2-320-v3-Kostenko-Valeriya", "600x300")  
}  
  
  
# Основна функція  
def main():  
 choice = input("Please, choose the task 1-2 (0-EXIT): ")  
 while choice != "0":  
 # якщо даний ключ є у словнику  
 if choice in task\_window\_dict.keys():  
 # Створення відповідного вікна  
 application = tkinter.Tk()  
 window\_class, window\_name, window\_size = task\_window\_dict.get(choice)  
 window = window\_class(application)  
 application.geometry(window\_size)  
 application.title(window\_name)  
 application.mainloop()  
 else:  
 print("Wrong task number!")  
 choice = input("Please, choose the task again (0-EXIT): ")  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

import tkinter  
from tkinter import messagebox  
  
class CalculatorWithPowerA234(tkinter.Frame):  
 """Graphical user interface and logic for calculating powers of a number"""  
  
 def \_\_init\_\_(self, parent):  
 super().\_\_init\_\_(parent)  
 self.pack(fill=tkinter.BOTH, expand=1)  
 self.grid\_rowconfigure(0, weight=1)  
 self.grid\_rowconfigure(1, weight=1)  
 self.grid\_columnconfigure(0, weight=1)  
 self.grid\_columnconfigure(1, weight=1)  
 self.grid\_columnconfigure(2, weight=1)  
  
 # Labels and Entry widgets for power calculation  
 self.lb5 = tkinter.Label(self, text="Enter number A:")  
 self.a\_power\_entr = tkinter.Entry(self)  
 self.btn2 = tkinter.Button(self, text="Calculate powers", command=self.calc\_powers)  
 self.result\_label2 = tkinter.Label(self, text="Powers:")  
 self.lb6 = tkinter.Label(self, text="Second Power:")  
 self.lb7 = tkinter.Label(self, text="Third Power:")  
 self.lb8 = tkinter.Label(self, text="Fourth Power")  
 self.result2\_str = tkinter.StringVar()  
 self.result\_label\_second\_power = tkinter.Label(self, textvariable=self.result2\_str)  
  
 self.lb5.grid(row=4, column=0, sticky=tkinter.NSEW)  
 self.a\_power\_entr.grid(row=4, column=1, sticky=tkinter.NSEW)  
 self.btn2.grid(row=5, column=0, columnspan=2, sticky=tkinter.NSEW)  
 self.result\_label2.grid(row=6, column=0, columnspan=2, sticky=tkinter.NSEW)  
 self.lb6.grid(row=7, column=0, sticky=tkinter.NSEW)  
 self.lb7.grid(row=8, column=0, sticky=tkinter.NSEW)  
 self.lb8.grid(row=9, column=0, sticky=tkinter.NSEW)  
 self.result\_label\_second\_power.grid(row=7, column=1, rowspan=3, sticky=tkinter.NSEW)  
  
 def calc\_powers(self):  
 try:  
 a\_power = float(self.a\_power\_entr.get())  
 except ValueError:  
 messagebox.showerror("Data ERROR", "Number must be a valid real number!")  
 self.a\_power\_entr.delete(0, tkinter.END)  
 else:  
 second\_power, third\_power, fourth\_power = PowerA234(a\_power)  
 result\_text = f"Second Power: {second\_power:.2f}\nThird Power: {third\_power:.2f}\nFourth Power: {fourth\_power:.2f}"  
 self.result2\_str.set(result\_text)  
  
  
def PowerA234(A):  
 """Calculate the second, third, and fourth power of a given number."""  
 square = A\*\*2  
 cube = A\*\*3  
 fourth\_power = A\*\*4  
 return square, cube, fourth\_power  
  
# Для графічного інтерфейсу  
import tkinter  
from tkinter import messagebox  
from tkinter.filedialog import askopenfile  
# Для малювання графіка  
from pylab import \*  
from matplotlib.backends.backend\_tkagg import FigureCanvasTkAgg  
from matplotlib.figure import Figure  
from PIL import Image, ImageTk  
  
  
class Task2Window(tkinter.Frame):  
 """Клас MainWindow, що наслідує Frame"""  
  
 def \_\_init\_\_(self, parent):  
 """Настройка графічного інтерфейсу"""  
 super().\_\_init\_\_(parent)  
 # Розтягнути фрейм  
 self.pack(fill=tkinter.BOTH, expand=1)  
 # Розтягнути сітку  
 self.grid\_rowconfigure(0, weight=1)  
 self.grid\_rowconfigure(1, weight=1)  
 self.grid\_rowconfigure(2, weight=1)  
 self.grid\_columnconfigure(0, weight=1)  
 self.grid\_columnconfigure(1, weight=1)  
 self.grid\_columnconfigure(2, weight=1)  
 self.grid\_columnconfigure(3, weight=1)  
 # Створення віджетів (зображення виразу та поле для введення N)  
 self.img = ImageTk.PhotoImage(file='image.png')  
 self.lb\_image = tkinter.Label(self, image=self.img)  
 self.lb1 = tkinter.Label(self, text="N = ")  
 self.N\_entr = tkinter.Entry(self)  
 # Створення віджетів (4 командні кнопки)  
 self.but1 = tkinter.Button(self, text="Create file", command=self.create\_file)  
 self.but2 = tkinter.Button(self, text="Open file", command=self.open\_file)  
 self.but3 = tkinter.Button(self, text="Show content", command=self.show\_msg)  
 self.but4 = tkinter.Button(self, text="Show plot", command=self.show\_plot)  
 # Розміщення віджетів в сітці основного вікна  
 self.lb\_image.grid(row=0, column=0, columnspan=2, sticky=tkinter.NSEW)  
 self.lb1.grid(row=0, column=2, sticky=tkinter.NSEW)  
 self.N\_entr.grid(row=0, column=3, sticky=tkinter.NSEW)  
 self.but1.grid(row=1, column=0, sticky=tkinter.NSEW)  
 self.but2.grid(row=1, column=1, sticky=tkinter.NSEW)  
 self.but3.grid(row=1, column=2, sticky=tkinter.NSEW)  
 self.but4.grid(row=1, column=3, sticky=tkinter.NSEW)  
 self.text1 = "" # вміст файлу  
  
 def create\_file(self):  
 """Розрахунок значень функції і збереження результатів у файл"""  
 try:  
 N = int(self.N\_entr.get())  
 if N < 20:  
 raise ValueError  
 except ValueError:  
 messagebox.showerror("Data ERROR", "N must be integer that >= 20!")  
 else:  
 # Параметри виразу  
 K = 3  
 T = 0.05  
 T0 = 2\*T/N  
 U = 0.1  
 eps=0.25  
 x = [0]  
 y = [0]  
 # Розрахунок N значень x, y  
 for k in range(1, N):  
 x.append(k\*T0)  
 tmp\_value = 2\*(1-((eps\*T0)/T))\* y[k-1]+(((2\*eps\*T0)/T)-1-((T0\*T0)/(T\*T)))\*y[k-2]+((K\*T0\*\*2)/T\*\*2)\*U  
 y.append(tmp\_value)  
 # збереження результатів у файл  
 with open("graph\_data.txt", 'w') as f:  
 for i, x in enumerate(x):  
 f.write("{}#{}\n".format(x, y[i]))  
 # повідомлення про успішний запис результатів у файл  
 messagebox.showinfo("File creation", "File with data was created!")  
  
 def open\_file(self):  
 """Зчитування вмісту файлу і збереження в text1"""  
 # Виклик вікна діалогу для відкриття файлу  
 fopen = askopenfile(mode='r', defaultextension=". txt",  
 filetypes=(("Text files", "\* .txt"), ("All files", "\*. \*")))  
 if fopen is None: # якщо помилка відкриття файлу  
 return  
 self.text1 = fopen.readlines() # файл -> список рядків  
 messagebox.showinfo("File opening", "File with data was opened!")  
  
 def show\_msg(self):  
 """Відобразити text1 у вікні messagebox"""  
 messagebox.showinfo("File content", self.text1)  
  
 def show\_plot(self):  
 """Рисування графіку функції"""  
 x = []  
 y = []  
 try: # розібрати список рядків text1  
 for line in self.text1: # для кожного рядка  
 words = line.split('#') # зберегти як список  
 x.append(float(words[0])) # 1 ел.списка -> число -> x  
 y.append(float(words[1])) # 2 ел.списка -> число -> y  
 except ValueError:  
 messagebox.showerror("Data ERROR", "Wrong file format!")  
 else:  
 # Область малювання графіка на полотні (Canvas)  
 fig = Figure(figsize=(3, 3)) # створення об'єкта Figure  
 a = fig.add\_subplot(111) # створення об'єкта області малювання (subplot)  
 # Настройка області побудови графіка  
 a.plot(x, y, 'c--')  
 # ...  
 # Створення об'єкта Canvas і розміщення в основному вікні  
 drawing = FigureCanvasTkAgg(fig, master=self)  
 drawing.get\_tk\_widget().grid(row=2, column=0, columnspan=4, sticky=tkinter.NSEW)  
 drawing.draw()  
 # Інформація про максимальне/мінімальне значення аргументу/функції  
 min\_x = min(x)  
 min\_y = min(y)  
 max\_x = max(x)  
 max\_y = max(y)  
 messagebox.showinfo("Basic information", "X min = {}, X max = {}\n"  
 "Y min = {}, Y max = {}".format(min\_x, max\_x, min\_y, max\_y))

Виконання





Висновок

У процессі виконання роботи я отримала теоретичні знання з основ програмування на мові Python звикористанням об'єктів і класів, навички використання бібліотеки для візуалізації масивів даних, і навчитися розробляти скрипти для роботи з об'єктами призначених для користувача класів.