Class Exercise:

Interop v1

In this exercise, you will learn how to integrate Cython, ctypes, Elmer, and Weave in a Python program. These tools are commonly used for optimizing and accelerating code execution in scientific computing and numerical simulations.

Problem Statement:

You are given a Python program that calculates the sum of squares of a given list of numbers. However, the program is quite slow and needs optimization. Your task is to modify the program using Cython, ctypes, Elmer, and Weave to improve its performance.

Instructions:

1. Install the required libraries:

* **Cython**: Use the command pip install **cython**.
* **ctypes**: No separate installation required as it is a part of the Python standard library.
* Elmer: Download and install Elmer from the official website (https://www.csc.fi/web/elmer).
* Weave: Use the command pip install weave.

1. Create a new Python file, e.g., optimization.py, and import the necessary libraries:
2. Define a Python function, **sum\_of\_squares**, that calculates the sum of squares of a given list of numbers:
3. Convert the **sum\_of\_squares** function to **Cython** by adding type annotations and using Cython syntax. Save this as sum\_of\_squares.pyx:
4. Create a C library from the **Cython** code using the **pyximport** module. Add the following code to the optimization.py file:
5. Use **ctypes** to load the C library and call the **sum\_of\_squares\_cython** function. Replace the existing **sum\_of\_squares** function with the following code:
6. Use Elmer to compile the C library. Add the following code to the optimization.py file:
7. Use Weave to optimize the original Python function. Replace the existing **sum\_of\_squares** function with the following code:
8. Test the program by creating a list of numbers and calling the **sum\_of\_squares** function. For example:

import cython

from ctypes import \*

import elmer

import weave

def sum\_of\_squares(numbers):

# Your code here

pyximport.install()

import sum\_of\_squares

def sum\_of\_squares(numbers):

# Your code here

elmer.compile("sum\_of\_squares.c", ["sum\_of\_squares.pyx"])

def sum\_of\_squares(numbers):

# Your code here

numbers = [1, 2, 3, 4, 5]

print(sum\_of\_squares(numbers))