HPC Lab Week 1

Implement the following using Multithreading in Python

1. **Matrix Multiplication:** Create two matrices (each of size n*n) consisting of random integer numbers. Perform matrix multiplication a) using serial operations with nested for loops, b) using multithreading functionality to independently perform dot product operations in matrix multiplication, c) using built-in numpy matrix multiplication. Analyze different time taken for each of these implementations by changing a) size of the matrix size n, b) random integers into random floats, c) number of threads used (small vs large)
2. **Linear Search:** Create a random integer array A of size n (preferably large). Write a program to perform linear search for a given input x (i.e. finding the indices of the search element x in the array A) a) using serial linear search iteration each value, b) using approach that divides given array and perform simultaneous search using threading, c) using built-in functions. Analyze different time taken for each of these implementations by a) changing size n of the array, b) changing integer random values into float numbers, c) number of threads used (small vs large)
3. **Array Sum:** Create a random integer array A of size n (preferably large). Write a program to perform sum of array a) using for loop which adds elements one by one, b) using an approach that divides given array and perform simultaneous addition using threading and further adding the thread results into final result, c) using built-in functions. Analyze different time taken for each of these implementations by a) changing size n of the array, b) changing integer random values into float numbers, c) number of threads used (small vs large).
4. **Prime Numbers:** Write a program to print all prime numbers from 1 to given number N, a) using serial executing of for loop and finding prime number, b) using an approach that divides search space (from 1 to N) using threading and aggregating thread results into final result. Analyze different time taken for each of these implementations by a) changing N b) number of threads used (small vs large).