NAME-ISTIAQUE MANNAFEE SHAIKAT

MATRICULATION NUMBER-303527

Exercise 0: Explain your system

PROCESSOR	INTEL CORE I7 8TH GENERATION
NUMBER OF CORES	6 cores
RAM	16GB
OS	WINDOWS 10
SOFTWARE VERSION	Anaconda 3/Python 3.7.1/spyder 3.3.2

Exercise 1A:

SIZE OF THE VECTORS	P	TIME/second
10^7	2	2.7944
	4	3.2306
	6	3.7312
	8	4.1766
	10	5.0863
	12	5.5519

For 10^12 and 10^15 the output is memory error

The python code consist of three functions they are

- Create_vector()- this function is used to create two random vectors of size N.
- Add_vector(V1,V2)-this function is used to add the chunks or split vector values and these is where the parallel programing occurs
- sort_vector(z)-this function is used to sort the vectors after merging all the values from different workers

After starting executing the code in command prompt using "mpiexec python lab_q1A.py" the program collects rank as worker and size as num_worker then num_worker is used to find out the splits we have to make in each vector for parallel addition.The chunk variable provides the range of first and last chunk.

When the worker is equal to zero new vectors are created and sent to all other workers as data on the other hand for other workers the vector addition is done to the chunks of vectors parallel way and saved it in variable vC.After that vC is returned back and worker zero communicates with other workers to receive the summed values and it is appended to create the final unsorted vector.Now it will use the sort_vector(temp) function to sort the vector according to the worker to provide the final sorted vector V3.

Exercise 1B:

SIZE OF THE VECTORS	P	TIME/second
10^7	2	0.6988
	4	0.8637
	6	0.7278
	8	0.8288
	10	0.8114
	12	0.7967

For 10^12 and 10^15 the output is memory error

The python code consist of three functions they are

- create_vector()- this function is used to create two random vectors of size N.
- sum_vector(V)-add all the values from single vector chunk

After starting executing the code in command prompt using “mpiexec python lab_q1A.py” the program collects rank as worker and size as num_worker then num_worker is used to find out the splits we have to make in a vector for parallel average summation. The chunk variable provides the range of first and last chunk.

When the worker is equal to zero new vectors are created and sent to all other workers as data on the other hand for other workers the vector values are added in parallel way and saved it in variable sum_v.After that sum_v is returned back and worker zero communicates with other workers to receive the summed values and it is appended to create the final summed values. Later on, divide by size of vector to get the average.

Exercise 2: Parallel Matrix Vector multiplication using MPI using point to point communication

SIZE OF THE VECTORS	P	TIME/second
10^2	2	0.00017
	4	0.00034
	6	0.00047
	8	0.00052
	10	0.00069
	12	0.00080

SIZE OF THE VECTORS	P	TIME/second
10^3	2	0.0032
	4	0.0041
	6	0.00401
	8	0.00403
	10	0.00442
	12	0.00464

SIZE OF THE VECTORS	P	TIME/second
10^4	2	0.221
	4	0.260
	6	0.262
	8	0.282
	10	0.266
	12	0.272

Few changes were made in this code compared to the exercise 1A code. Firstly, taking chunks horizontally from vA which is then multiplied with vB.This task is done parallel way by other workers then all the multiplied values are collected and merger together to create a matrix multiplied vector which was unsorted. Later on sorted depending on the worker.