Lab-Report

Submitted by: - Parth Parashar

I had a very pleasant experience working with this lab-assignment.

The file I/O operations are an integral part of every programming language and the way MPI handles I/O operations is quite efficient.

I was able to compile and run all the programs for the first question. The way MPI handles the input output operations is quite commendable.

MPI is very quick as well. The use of message passing is quite efficient as compared to other thread handling techniques.

MPI is also very compact as well and the in-built functions are quite quick as well.

I had to struggle a little with the second program though. I had a hard time getting the right functionality. I was able to get the input and division of P was also performed properly along with termination message.

Array was set properly as well.

For computing the total sum, I made use of the sum-mpi.c program.

For doubling the value of each value and writing it to the output file was convenient as well.

This was a very good introduction to the functionality of MPI.

Program-1: -

1. For program -1, I was able to run it properly using the provided code.

Text

Description automatically generated with medium confidence

Text

Description automatically generated

The modified program is given: -

Text

Description automatically generated

1. When the given program is compiled and run, there are 4 output files which are created. This number 4 came from the value of n provided with the mpirun statement.

A screenshot of a computer

Description automatically generated with medium confidence

The output files are highlighted in the screenshot provided above.

Text

Description automatically generated with medium confidence

As can be seen from the screenshot and the code, each file contains 4 integers.

After making the necessary changes, we get the following output: -

A screenshot of a computer

Description automatically generated with medium confidence

The modified code is given below: -

Text

Description automatically generated

1. When we compile and run the program, we get the following output: -

A picture containing text

Description automatically generated

In the output, we can see that there are four integers according to their ranks along with the total number.

Now, the offset calculation when changed as given below will give the output as provided in the next two screens.

Text

Description automatically generated

Text

Description automatically generated

Now, when offset is changed again, then we get the following results (screenshots for changed value and the result)

Text

Description automatically generated

Text, timeline

Description automatically generated

Program-2: - Program-2 is attached in this zip file.