Parallel and Distributed Computing

ReadMe

Group Members:

Syed Ahmad Mustafa	21I-0886
Saman Ali Ahmed	21I-2499
Syed Ata-ul Muhaimen	21I-0888

Source Code Execution Instructions:

- Download the source code and store it into a folder.
- Install the necessary libraries required for the execution of OPENMP and MPI based codes.
- Navigate to the destination folder using the cd command.
  - For example: cd Desktop // this will navigate you to the desktop folder.
- Create an executable for the source code using the mpicc -o command.
  - For example: mpicc -o pdcProject pdcProject.c
- An executable with the name "pdcProject will be generated".
- Finally execute the program using mpiexec -n command and enter the no. of concurrent processes you wish to create.
- The source code's implementation is based on the execution of **5 concurrent processes**.
  - For example: mpiexec -n 5 ./pdcProject

Source Code Functions Reference:

- Int dataSetSize(char* filename)
  - Returns the total number of lines the file has.
- Char*** readFile(char* filename, int dataSize)
  - Returns the data stored in the file in the form of a 3D character array.
- Int** distanceMatrix(char*** dataNode, int dataSize)
  - Initializes and returns the adjacency matrix created using data read from the file.

- Int shortestPath(int source, int destination, int **adj, int V)
    - Returns the shortest path between a source and destination pair.
- Void findKthPath(int** disMatrix, int numNodes, int source, int destination, int k)
    - Uses the shortestPath() function to find k shortest paths between a source and destination.

Sample Output:

The output contains each of the 5 source and destination pairs followed by the **k** shortest paths between each of the pair and the execution time for each process.