Assignment 9: Map Reduce MPI

The purpose of this assignment is for you to learn more about

- Using Map Reduce for text processing
- Using Map Reduce to solve a data sceicen problem.

As usual all time measurements are to be performed on the cluster. The application using MapReduce MPI need to be linked against the mrmpi library. In the assignment, the Makefile accoutns for it.

Only the first problem is mandatory, the second one is extra credit. As such, the grade thresholds are : $A \geq 35$; $B \geq 30$; $C \geq 20$; $D \geq 10$

1 Word Count (40 pts)

This problem is fairly simple. Read a file and count how many times each word appear more than a given number of times. For the purpose of the assignment, count as a word any string of characters that does not contain a white character (space, line break, ...) and don't worry about punctuation.

Question: Go into the `wordfreq/` directory and write the code in `wordfreq.cpp` using MapReduce MPI. There is a sequential implementation of the problem in `wordfreq_seq.cpp`.

The code should take two parameters: the name of the file to process and a threshold on the minimum number of time a word needs to appear to be retained.

The program should:

- Print on `stdout` the list of words that appear more times than the threshold, and the number of time they appear.
- Print on `stderr` the time it took for the application to make that computation (IO included).

You can test your code with `make test`. Though you will notice that the test isn't as complete as in previous assignments.

Question: Run the code on mamba using `make bench`. And once it is over, print a time table with `make plot`. Note that in this assignment `make bench` will queue the jobs one at a time and wait for each job to complete.

2 k-Nearest Neighbor (60 pts)

The k-Nearest Neighbor algorithm is a machine learning algorithm to infer classification of a query based on known classification of a set of observation.

Mathematically, you are given a database of n points located a d-dimensional feature space, and each point is associated with a class (an integer) and queries, vectors in the same d dimensional space. The k-nearest neighbor algorithm will guess a class for each query by: identifying the k points in the database that are the closest to the query (by euclidean distance), computing which class appears the most frequently among the k nearest neighbors.

Question: Go into the `knn/` directory and write the code in `knn/knn_mrmpi.cpp` using MapReduce MPI. There is a sequential implementation of the problem in `knn/knn_seq.cpp`.

The code should take three parameters: the name of the database file, the name of the query file, and k (the number of neighbors to consider).

The database file is a list of points in a high dimensional space and classified. That is to say each line is a point in a high dimensional space. Each line is composed of comma separated values, the last value is an integer class; all others are coordinates in the high dimension space.

The query file is in the same format, without a class.

The program should:

- Print on `stdout` the queryIDs and estimated class the queryID the k-Nearest Neighboor maps it to.
- Print on `stderr` the time it took for the application to make that computation (IO included).

You can test your code with `make test`. Though you will notice that the test is not as complete as in previous assignments.

Question: Run the code on mamba using `make bench`. And once it is over, print a time table with `make plot`. Note that in this assignment `make bench` will queue the jobs one at a time and wait for each job to complete.