MapReduce using MPI

It is important to understand that Mapreduce is a paradigm, typically we’d think about doing it using Hadoop (or Spark or flink) but there’s no particular reason you can’t do it natively in C/C++ and using MPI.

So with that, let’s have a look at a simple example.

<https://mapreduce.sandia.gov/> has an MPI implementation from Sandia labs in the US,

<https://www.sandia.gov/~sjplimp/download.html>

Specifically, you download the latest version there, just click the radio button beside mapreduce-mpi and download (I think I attached it on Blackboard, it’s only 3 Mb).

Gunzip and untar that file somewhere handy that you can get into WSL/Linux

Extract that and then in the src directory Type: make mpicc (I think you need this)

Go to the examples directory (you need to go back one level from src and then down to examples)

Type make -f Makefile.mpicc (again, this is a variant of something with options, you could try make -f Makefile.linux or make -f Makefile.mac). Note: This doesn’t behave on a mac. If you’re a mac user, make friend with a windows user friend.

Now try running the program:

mpirun -np 8 wordfreq \*.cpp

That will give you a word count of all the .cpp files in the examples directory using 8 threads, you can use a smaller number of threads instead. Not really the most useful thing in the world, but you get the idea.

Try that out on some other text file (e.g. a .txt. output of this lab or some other document you have).

The files that actually power that are in the /src/ directory, specifically mapreduce.h and mapreduce.cpp, keyvalue.h, keyvalue.cpp, and the /examples/wordfreq.cpp. All told this problem is about 5000 lines long including those files but not standard libraries.

Two tiny problems for today.

First, change the ncompare function so that it sorts the list in the reverse order

Second: Change the file read to read in words, then remove the vowels from the string, then add it to the key value pair. (This should ignore punctuation etc.).

Show your changed code and results, get marks. Nothing to it. (This is more to show you were some good sample code is, and an implementation of mapreduce from scratch. We’re past the point of asking you to do anything new).