TSN3151 Assignment

Group formation due Sunday, 4 June at midnight
Assignment due: Thursday, 22 June at midnight.

Form groups of 3 or 4 people to do this assignment. Use the MMLS group feature to form your groups by Sunday, 4 June. If you can't figure out how to do that, ask in the WhatsApp group before 5pm Friday 2 June.

Write an MPI program that would let the user specify any number of text files, then count the word frequency of every word in all the files. The counting should be parallelised using MPI.

The code should not only count the number of words but also prompt the user to specify the minimum and maximum word lengths to be included in the count. For instance, if the user sets a minimum word length of 4, words like "he", "and", "the", "is", and "I" should be excluded from the count since they have fewer than 4 letters. The output should display the words sorted in descending order based on their frequency.

You can use these files as test files, but do test with other files as well:

- The complete works of William Shakespeare text file: https://www.gutenberg.org/cache/epub/100/pg100.txt
- The Complete Poetic and Dramatic Works of Robert Browning https://www.gutenberg.org/cache/epub/50954/pg50954.txt
- The Complete Poetical Works of Percy Bysshe Shelley https://www.gutenberg.org/cache/epub/4800/pg4800.txt

Download the files to your computer before running your program – your program does not need to get them from the Internet. I may be testing with different files than these.

What you must submit:

- Complete code that is documented properly with comments showing who wrote what code. At the top of each C or C++ file must be the name of the group. If there is more than one file, please put instructions for compilation in case there is some specific order your files must be compiled in. **Every member must be involved in the programming.**
- A YouTube video presentation of 10 to 15 minutes where you show and explain your code. Every group member must be involved and show your video feed so that I can tell who is talking. You can record with Google Meet or use OBS Studio or any other tools to show your screen or slide with the person talking. **Make sure the code is easily readable** when playing the video on a standard laptop screen. The video should explain your design choices and limitations of your code – for example, what couldn't be parallelised and why.
- A report that explains the idea and algorithm of how your code works. The report should explain your design choices and limitations of your code – for example, what couldn't be parallelised and why. (Can be the same explanation as in the video, except in proper written form.)
    - The first page of the report must have:
        - Name of group
        - Name of group members
        - URL of the YouTube video

Plagiarism

I have plagiarism checker software for code. If you Google, you will find some code for word counting and MPI online. If you read their code and then write your own code from scratch, that's fine.

But if you **copy** their code instead of writing your own code, *even if you have modified or rearranged parts of it or changed the variable names etc*. **my plagiarism detector software will find you out.**

If you copy any other team's code or if you let your code be copied by any other team, **my plagiarism detector software will find you out.** All such plagiarism will get **zero** – so **guard your code like an official secret!** Do not let any other team copy your code, or you will also get **zero**.

Late submission

Late submission is allowed but will be subject to a 10% penalty per day. So, if you can't finish in time, no need to ask me for an extension. Extensions for up to 5 days are automatically given, subject to the late penalty.

So, for example, if you do a perfect assignment, you get 30 marks if you submit it on time, 27 marks if you submit it 1 day late, 24 marks if you submit it 2 days late, etc.