Assignment 1:

For this problem, you can work in groups of up to two and you can use any programming language you want. Just note that you'll be programming a multi-threading program, so pick your language wisely.
Let's explore the power of distributing computation. [10 points]

Task 1

Implement a single-threaded WordCount application, which takes as input a directory of text files, and generates a single output file with the frequency count of each word in all documents. Use it to analyze any of your favorite books.
Example:
Input document with text: "Be not afraid of greatness: some are born great, some achieve greatness, and some have greatness thrust upon them".
Will generate an output file with the text:
Be:1
Not:1
Afraid:1
Of:1
Greatness:3
Some:3
Are:1
Born:1
Great:1
Achieve:1
And:1
Have:1
Thrust:1
Upon:1
Them:1

Task 2

Implement a multi-threaded WordCount application.
In this application, you will create multiple threads to speed up the computation of WordCount from task 1 above. A single thread takes as input one or more text files and generates the frequency count of each word in the documents. You can synchronize the threads in any way that you see fit, but note the following challenges:

1. You will need to divide the data among the threads somehow.

2. You don’t want two or more threads to process the same document (or part of document).
3. All output needs to be placed in a single output file at the end of the program.

You can use as many data structures and as many Semaphores, as you see fit, in this task.

Task 3

Write a report on your work on Task 2. In this report you will need to include the following:

- How did you solve each challenge in task 2? Be specific, add pseudo code if you need to.
- Did you face any other challenges? How did you solve them?
- Calculate the running time of task 2, using a varying number of threads. Start with a single thread and keep increasing the number of threads till the running time doesn’t change (significantly).
- Create a plot to represent the varying running time with increased number of threads. Explain your results.

Deliverables

- The code.
- The report from Task 3.
- A document summarizing the group discussions, prior to the implementation. It should include:
    - The main ideas discussed within the group
    - The references/readings used during the discussion
    - The challenges faced while discussing an appropriate solution