CS450 Assignment 2

Katya Griffiths-Julien, Jerry Lau

**Task A**

The formula was relatively simple as the mapper was used to calculate the log of the xi values of the randomly generated numbers. Then the reducer would take the sum and divide it by the total number of items.

There is no output from this task as we simply wrote the formula using the MRJob.

**Task B**

The code was changed to allow us to see the execution of the progress on the installation process using Google colab. When the -q was put in place, it seems like nothing was happening so we removed those to ensure that Hadoop, Java and other libraries would be installed properly.

There is a lot of output from this task which is included in the .ipynb file submission which as indicated above, we removed all of the -q so that we could see the installation and download progress.

**Task C**

We used the same file since it would allow us to avoid reinstalling Hadoop on another notebook. The mapper and reducer were combined into 1 code block to allow us to execute both parts to computer the execution time.

The MapReduce program does not have an output thus no output is provided as the formula is to simply calculate a division over a sum to the logarithm of random numbers.

The generated graph shows that the execution time is linear or O(n) time complexity. When we look at the code which is executed, the call to the summation is exactly the same size as the input values thus it is in line with the O(n) execution time. This is shown in the graph of the size of the arrays in function to the execution time.

Graph 1.

A graph with a line

Description automatically generated

**Conclusion**

The use of Hadoop for the distributed computation does not really show any difference than plain execution of the code. This is because the input values are not in the target range of Hadoop typical usage, 100+ files of GB/TB. In fact, having the computation distributed for such as small sample would hinder the performance. This is due to the overhead computation cost of distribution and hashing the input which is then required to rejoined later. As such, there are minimal improvements.

Since this project only touches on minimal input data size, our maximum was only 5000 items, most of the disadvantages on the use of Hadoop were touched upon. Most of the advantages that Hadoop such as scalability, flexibility was never put into play on such a small sample size. The computational overhead of using a distributed file structure could also hinder such a small sample owing to the fact that pre and post processing into many nodes then doing a small calculation might take additional time that simply having 1 node do a bunch of computation. As such, the speed of which Hadoop can leverage is not suitable for a few kb of input data.