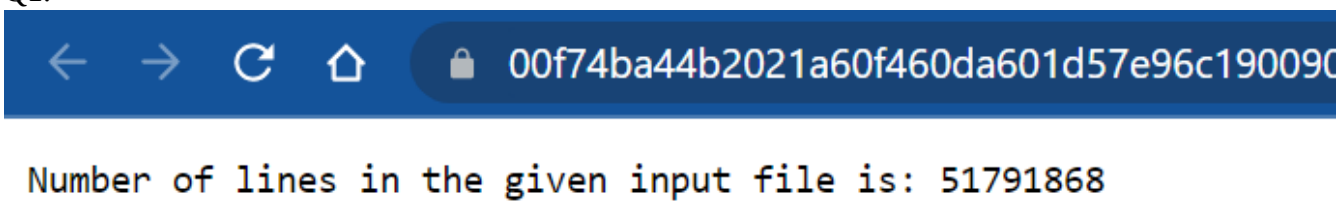
CS4830 Big Data Lab Assignment 1

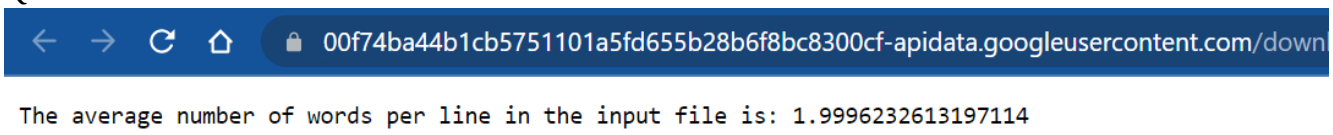
Shashank H S BE18B006

Q1.



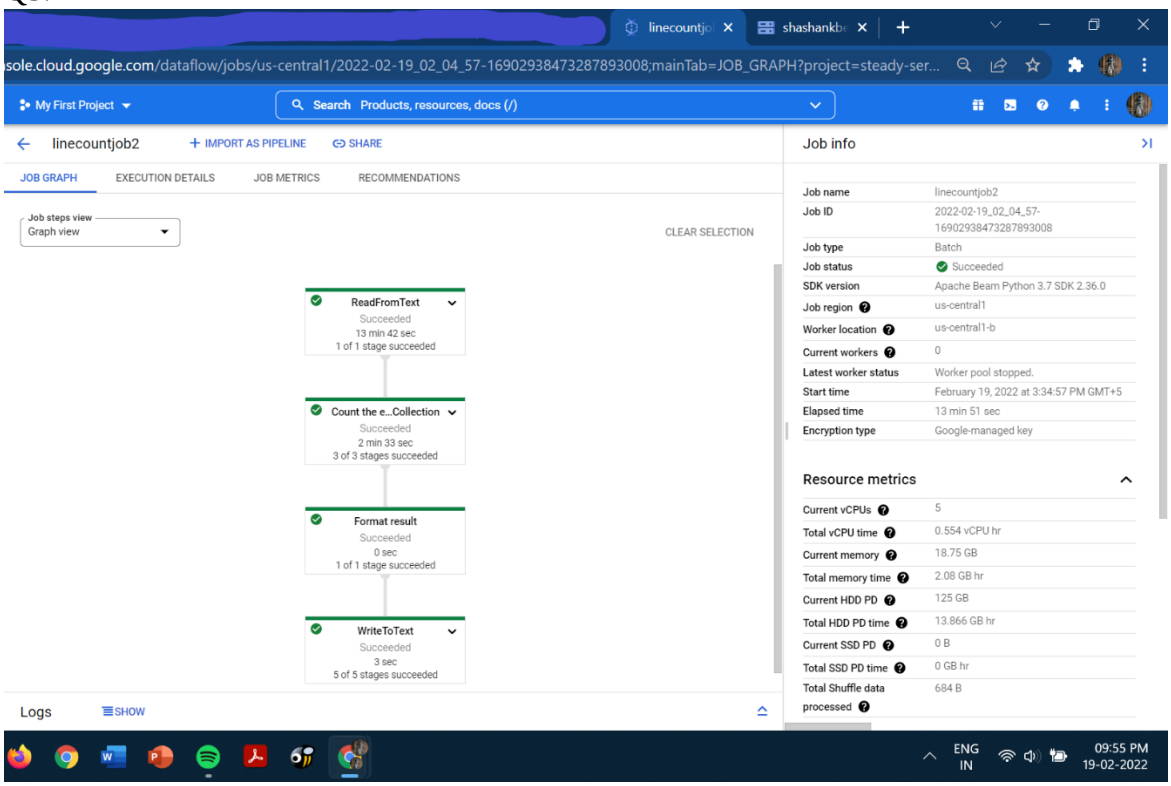
The number of lines in the given input file is 51791868. The uncropped version of the screenshot is in the zipped folder.

Q2.

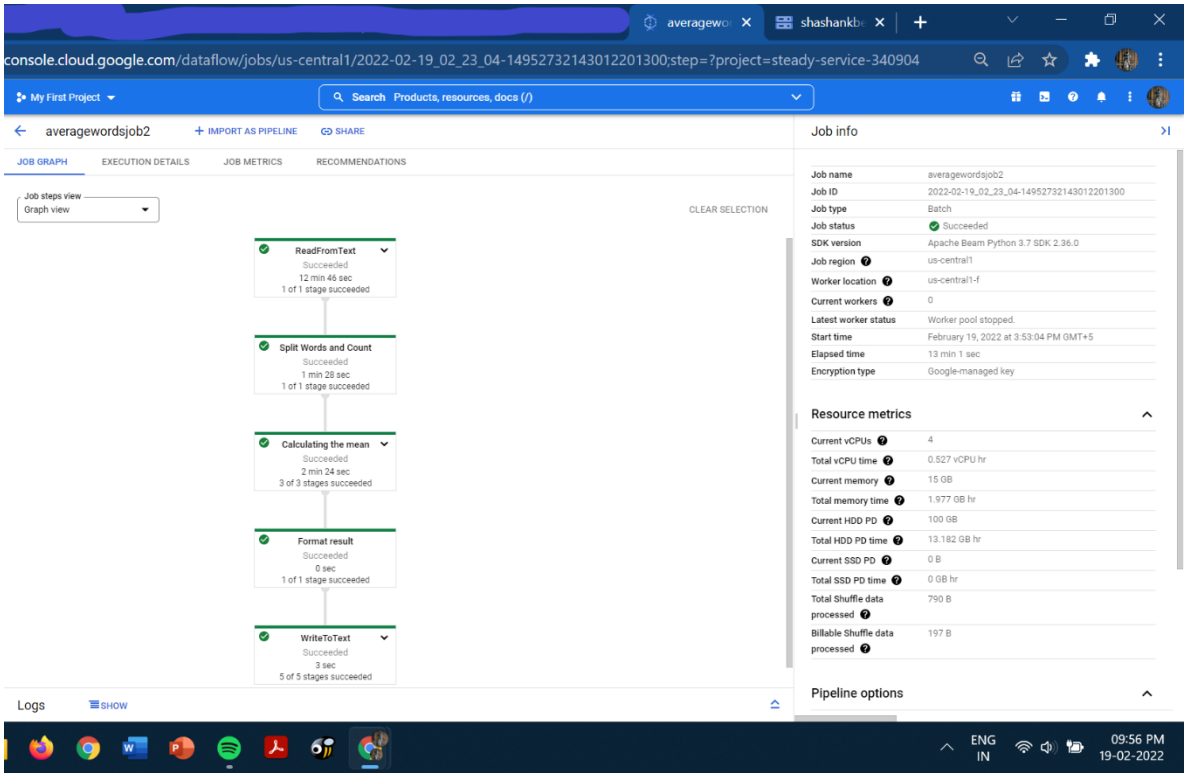


The average number of words per line in the input file is 1.9996232613197114. The uncropped version of the screenshot is in the zipped folder.

Q3.



The execution graph created by Dataflow for question 1. The uncropped version of the screenshot is in the zipped folder.



The execution graph created by Dataflow for question 2. The uncropped version of the screenshot is in the zipped folder.

Q4. The pipeline for question 1:

1. First the text file is read using the ReadFromText PTransform to get a PCollection ‘lines’.
2. The count of the elements in the PCollection ‘lines’ is calculated using beam.combiners.Count.Globally() and a new PCollection ‘countarray’ is obtained.
3. beam.Map() is use to map a simple function ‘formatresult’, which returns a string to display in the output file, and provide a PCollection named ‘output’.
4. The PTransform WriteToText is used to write the ‘output’ PCollection to the output path.

The pipeline for question 2:

1. First the text file is read using the ReadFromText PTransform and a PCollection is obtained. Next, a ParDo transform is implemented in the form of a Beam DoFn object where the class is named WordCount. The ‘process’ method in WordCount processes each element in the PCollection got from reading the file and it returns the number of words in the element (each line in the text file). The output PCollection is ‘wordcounts’.
2. Each element in the PCollection ‘wordcounts’ corresponds to the number of words in a line. The mean is calculated using beam.combiners.Mean.Globally() and a new PCollection ‘word_mean’ is obtained.
3. beam.Map() is use to map a simple function ‘formatresult’, which returns a string to display in the output file, and provide a PCollection named ‘output’.
4. The PTransform WriteToText is used to write the ‘output’ PCollection to the output path.

- While reading the documentation for Apache Beam, I felt like the important concepts were explained using complicated jargon which made it difficult to understand. I solved this problem by looking at the code examples and using that to understand the functions.
- In my initial attempt to solve question 2, I used a complicated pipeline which involved the FlatMap(), Map() and CombinePerKey() transforms. After thinking about the pipeline and looking for a better way of implementing transforms to read each line, I found the ParDo transform and then solved the question successfully.