Process Streaming data

The task aims at investigating data stream algorithms, including **Reservoir Sampling**, **Misra-Gries Summary**, and **CountMin** Sketch on real-world data sets.

In the task, you write a program to find *frequent items* on a stream. The data set is same as the first task.

The file has the format: *docID wordID count*, where *docID* is the document ID, *wordID* is the word ID in the vocabulary, and *count* is the word frequency. Ignoring the first 3 lines, we consider each line as a stream tuple (*docID, wordID, count*). In the assignment, we do not use the information of *count*. This means that you can think of *count* = 1 for each line. We want to find the most frequent words in our data set by our data stream algorithms.

The tasks are specified as follows.

1. **Execute bruteforce computation:** Compute the frequency vector of all words, descendingly sort the words by their frequencies, and save the result into file (since you might use the bruteforce result for the next tasks). You need to report:

   (a) **The average frequency of the words in stream.**

   (b) **Plot the sorted frequency of words to observe the skewed distribution.**

2. **Reservoir Summary:** Implement Reservoir Sampling to see the skewed distribution of our frequency vector. Fix the summary size S = 10, 000, you need to:

   (a) **Estimate the frequency vector from our Reservoir Summary, and plot this estimate vector to see the approximation skewness.**

   (b) **Run your Reservoir Sampling 5 times and report the average number of times the summary has been updated over these 5 runs**.

3. **Misra-Gries Summary:** Implement Misra-Gries summary to find the most frequent words whose frequency is larger than 1,000. You need to:

(a) **Explain the size of summary you choose such that you can find these frequent words.**

(b) **Run your Misra-Gries summary and report the number of decrement steps with your chosen parameter.**

4. **CountMin Sketch:** Implement CountMin sketch to estimate the frequency of words. You need to:

   (a) **Explain the size of summary you choose such that the estimate error is at most 100**.

   (b) **Run your CountMin Sketch with your chosen parameters, and report the estimate of the frequency of the words, whose frequencyis larger than 1,000 found in the bruteforce algorithm.**