Problem-1:

Build an approach for storing versions and files. A simple approach is to store each file as a separate version. However that is inefficient, Instead of storing each version as a separate file, look at storing the deltas. Come up with an approach to store the base version and deltas in the file. Persist them all in one file. Come up with an efficient data structure to store the file, and deltas, and persist in that. Write methods to generate any version immediately.

Approach: Linked List

1. In normal linked list we store the data,Instead of storing the data we can store the version and data in hashmap.

Problem-2:

Build a word count application, where the constraints are that you have 10 MB RAM and 1 GB text file. You should be able to efficiently parse the text file and output the words and counts in a sorted way. Write a program to read a large file, and emit the sorted words along with the count. Try to implement fuzzy search as well (fix the spelling issues) Algorithm should have Log N complexity.

Approach: HashMaps

1. Read the file and obtain words
2. Each word can be inserted in to HashMap and maintain the key,value pair and increasing the frequencies of the word.
3. Sorted the HashMap according to the frequencies.

Problem-3:

Come up with an approach for product configuration, where multiple products can be stored. Build an in-memory database or in-memory storage. We should be able to have product categories along with product descriptions and details. We should be able to store a wide range of types of products similar to Amazon, we should be able to implement efficient search of the products and flexible configuration of the products. In addition to in-memory storage, build an efficient textual search on any of the parameters (similar to search in Amazon).

Approach: HashMap in java

1. Created a class that contains the details of the product
2. Map the product name with all the details.

 Approach: Mapping and Structures in Solidity

1.Created a structure and mapped the product name with the structure and we can add multiple number of products

