- **Explanations of why chose the Array (w.r.t time and space complexity).**

    - **Arrays V/S Vector:**

    - Data structure used :- Arrays

        - Explanation ➖

          Time complexity ➖
          Arrays offer constant-time access to elements (O(1)), making them efficient for accessing file data.
          Iterating over arrays has a linear time complexity relative to the number of elements (O(n)), where n is the number of files.

          Space complexity ➖
          The array has fixed size providing space complexity for storing data which gives advantage when dealing with the maximum number of files (MAX_FILES).

    - **Vectors V/S Array for file deletion:**

    - **Data structure used : None (Direct manipulation of array elements)**

        - **Explanation ➖**

          Time Complexity ➖
          Direct deletion avoids the dynamic resizing and shifting, resulting in better performance in comparison to vectors.

          Space complexity ➖
          As the array size is fixed (MAX_FILES), there's no additional space overhead compared to vectors.

    - **Summary :**
        - **Arrays were chosen over vectors due to their time complexity and known maximum size which resulting in better performance and space uses.**
        - **Overall the chosen data structure strike a balance between time complexity and space complexity while efficiently handling file processing and deletion operation.**