Producto matriz – vector

Para este producto se implementó la función productmv. En esta función se definen las funciones map y reduce para realizar el procesamiento necesario, además de implementar la lógica correspondiente al manejo de la restricción de memoria. A continuación, se hace una breve explicación de lo que hace cada una de las funciones (map y reduce) y la lógica para el manejo de memoria:

- Map: Esta función recibe cada una de las entradas de la matriz y haciendo uso de una función auxiliar (f) obtiene el valor o resultado de multiplicar esa entrada de la matriz por la entrada correspondiente del vector por el cual se está multiplicando. Seguidamente se mapea los elementos de la fila de la entrada recibida con el valor o resultado obtenido mediante la función auxiliar.

- Reduce: Esta función recibe un par clave – valor donde la clave es el índice de los elementos asociados a la fila con la cual se mapeo en la función map y el valor es una lista de todos los valores o resultados obtenidos en la función map que se encontraban asociados a la fila representada por la clave. Esta función realiza una sumatoria sobre el valor y luego el resultado de la sumatoria lo mapea con la clave (estos últimos pares clave – valor son almacenados posteriormente en un archivo por las restricciones de memoria).

- Manejo de memoria:

    - Para realizar la multiplicación entre la matriz y el vector se optó por cargar una entrada del vector a la vez siempre, así como también reservando un espacio para la variable resultante a escribir en un archivo resultado, partiendo de esto se permite disponer de la memoria restante para cargar la mayor cantidad de elementos de la matriz.
    - Los pares clave – valor generados en la función reduce fueron almacenados en el archivo mapreduce.csv, que se va

escribiendo en un ciclo iterativo para aprovechar el espacio disponible en memoria.

Producto matriz – matriz

Para este producto se implementó la función productmm. En esta función se toma la segunda matriz pasada por parámetro y se separa esta matriz en sus distintas columnas donde a partir de estas se obtienen vectores que luego son almacenados en archivos temporales para su procesamiento. Seguidamente se hace aprovechamiento de la función productmv de forma iterativa, pasando como parámetros la primera matriz y cada uno de los vectores construidos previamente.

En cuanto al manejo de memoria en esta función, se optó por ir cargando una a una las entradas correspondientes de la segunda matriz para ir armando los vectores necesarios.