

### PROBLEMA 3

Este problema pide ordenar externamente un archivo de texto de 100 MB. Se utilizó un algoritmo de mezcla o mezcla binaria, se emplean las siguientes funciones para abordar este proceso:

**generar\_numeros\_aleatorios:** Genera un archivo con números aleatorios de longitud variable.

**contar\_lineas\_archivo:** Calcula la cantidad de datos en el archivo original, esencial para determinar la cantidad de ciclos de ordenamiento necesarios.

**dividir\_archivo:** Divide un archivo grande en dos más pequeños, escribiendo alternadamente bloques de tamaño `cant_datos` en cada uno.

**unir\_archivo:** Fusiona dos archivos pequeños en uno más grande, considerando que estos archivos pequeños contienen fragmentos ordenados de longitud `cant_datos`.

**ordenamiento\_inicial:** Realiza el primer ordenamiento, leyendo bloques de `cant_datos`, cargándolos en memoria, ordenándolos y escribiéndolos alternadamente en los archivos de salida. En este caso, se fija la cantidad `cant_datos` en 10.

**ordenar:** Combina las funciones anteriores para ordenar el archivo dado como parámetro, ejecutando los pasos necesarios para lograr el ordenamiento externo del archivo completo.

Para probar este código se realizó un test en el cual se crea y ordena un archivo cuya longitud puede modificarse, se probó con archivos de 10MB y 100MB, también editando la cantidad de cifras de los valores para acelerar el proceso. El test corrobora que el archivo ordenado tenga el mismo tamaño en bytes que el archivo original y que los datos del archivo ordenado estén en orden creciente.