

Caso de Estudio: Sistema de Gestión de Estacionamiento

Descripción del Caso:

En este estudio, abordamos el problema de gestionar y ordenar grandes volúmenes de datos generados por un sistema de estacionamiento. Este sistema divide el estacionamiento en cinco zonas: Clientes, Empleados, Carga/Descarga, Discapacitados/Embarazadas y Express. Cada entrada en el sistema indica una zona, un espacio y el estado (Disponible u Ocupado) del espacio correspondiente. La necesidad de manejar y analizar datos supera las capacidades de la memoria, lo que hace indispensable utilizar un enfoque basado en ordenación externa.

Algoritmo Seleccionado:

El algoritmo implementado combina:

1. División en Bloques: Los datos se dividen en bloques manejables, cada uno ordenado individualmente en memoria.

2. Fusión de Bloques: Los bloques ordenados se combinan usando un montículo (heap) para generar una lista final completamente ordenada.

3. Exportación a Excel: Los datos ordenados se estructuran y exportan a un archivo Excel, organizado por zona, espacio y estado.

Este enfoque permite manejar grandes volúmenes de datos sin sobrecargar la memoria.

Análisis del Código:

1. Generar Datos:

- La función *generar_datos* crea una lista simulada de datos con zonas aleatorias, espacios numerados y estados (“Disponible” o “Ocupado”).

- Se utiliza *random.shuffle* para garantizar que los datos no sigan un patrón predecible.

2. Dividir y Ordenar:

- La función *dividir_y_ordenar* procesa los datos en bloques de un tamaño definido. Cada bloque se ordena utilizando *sorted*.

- Esto asegura que cada segmento de datos sea manejable en memoria.

3. Fusionar Bloques:

- *fusionar_bloques* combina los bloques ordenados usando un montículo.

- Cada iteración extrae el elemento más pequeño y lo agrega al resultado final.

Si el bloque tiene más elementos, el siguiente se agrega al montículo.

4. Exportar a Excel:

- La función *exportar_a_excel* utiliza *pandas* para estructurar los datos en un DataFrame.

- Los datos se ordenan por "Zona" y "Espacio" antes de ser exportados a un archivo Excel.

5. Ejecución del Programa:

- Se generan 1000 entradas simuladas.

- Los datos se dividen, ordenan y fusionan, mostrando los primeros 10 resultados en consola.

- Finalmente, el archivo Excel se crea con los datos completos.

Conclusiones:

- La ordenación externa permite manejar datos que exceden la memoria disponible.
- El uso de bloques y montículos optimiza la eficiencia del proceso.
- La exportación a Excel organiza los resultados de manera práctica para análisis posteriores.

Este proyecto demuestra que algoritmos simples combinados con buenas prácticas pueden resolver problemas complejos relacionados con la gestión de grandes volúmenes de datos.