Norberto Garcia Gaspar

Ángel Alberto Hamilton Lopez

Óscar David Martín Cabrera

Divide y vencerás

Algoritmo para contar inversiones (Counting inversions)

1. **Introducción**

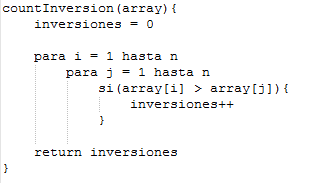
El problema del conteo de inversiones, normalmente efectuado sobre arrays, nos indica lo desordenado que esta dicho array, o lo que es lo mismo el número de movimientos que se tiene que llevar acabo para ordenar de menor a mayor (o de mayor a menor) los elementos de un array de tamaño n.

Siguiendo con el método de programación Divide y Vencerás, vamos dividiendo este array en arrays más pequeños para ordenarlos, una vez ordenados, procedemos a combinarlos y ver si hay que ordenarlo o no. El número total de inversiones por tanto serán desde los arrays más pequeños, sumados con los de las distintas combinaciones hasta obtener el array ordenado de tamaño n.

En este estudio vamos a ver una variante del MergeSort en el cual aparte de ordenar el array resolveremos el problema mostrado anteriormente, también veremos la forma de hacerlo mediante fuerza bruta para poder apreciar las diferencias y mejoría del rendimiento entre las dos técnicas.

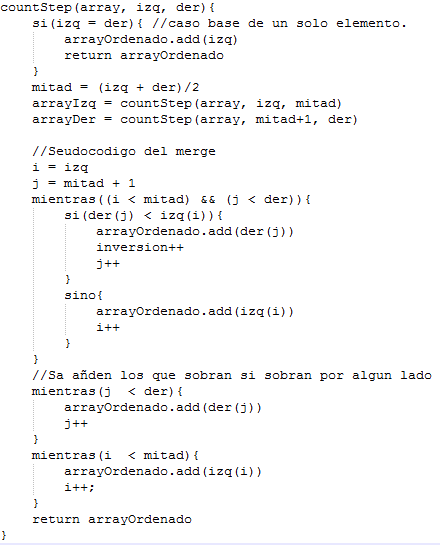
1. **Pseudocódigos**

Vamos a ver los pseudocódigos de los dos métodos para contar las inversiones. Empezamos con el de fuerza bruta en el que veremos solo obtendremos el número de inversiones, no ordenaremos también el array que estamos analizando.



Como observamos en el pseudocódigo anterior recorremos el array de izquierda a derecha buscando aquellos elementos que estén en una posición menor que la que le corresponde y vamos aumentando el contador de inversiones hasta que recorremos todo el array por cada posición del elemento.

Ahora vamos a ver como seria aplicando la técnica de Divide y Vencerás junto al algoritmo MergeSort de ordenación, con la modificación de ir contando las inversiones que hacemos en el array para resolver nuestro problema.



Vamos a destacar varias cosas para entender el análisis que viene a continuación. La primera de ella es nuestro caso base, que será cuando tengamos un array de un solo elemento. Tendremos que calcular la mitad del array, esta será la división que llevamos a cabo por la estrategia de divide y vencerás. Volvemos a llamar a la función para ir dividiendo una y otra vez nuestro array hasta obtener el caso base. Con el caso base lo que tendremos que hacer ahora es juntar dichos array de forma ordenada, comparando sus elementos e insertando el elemento correspondiente a nuestro array. Finalmente nos aseguramos de que todos los elementos están dentro de nuestro array, si no serán o mayores o menores así que los insertamos por la derecha o izquierda según corresponda. Destacar que lo que se devuelve con este algoritmo es el array ordenado, por lo que nuestra variable inversión es un atributo de la clase y como tal tenemos que llamar a su método “get” para conocer su valor.

1. **Análisis de complejidad y tiempo de ejecución**
2. **Conclusión**
3. **Referencias bibliográficas**