

Centro de enseñanza técnica industrial

CETI

Tema: Métodos de Ordenamiento.



Alumno: Isaac Carrillo Angulo

Reg. 21110358 grupo.6E1

Prof. Mauricio Alejandro Cabrera Arellano

Fecha: 24 de octubre de 2023

Objetivos

- Identificar la estructura de algunos algoritmos de ordenamiento.
- Interpretar el algoritmo de ordenamiento en sintaxis de C#.
- Aplicar los algoritmos de ordenamiento

Material y equipo

- Guía Número 3.
- Computadora con programa Microsoft Visual C#.

Desarrollo teórico

Que es el ordenamiento

El "ordenamiento" se refiere a la acción de organizar o disponer elementos de acuerdo con ciertas reglas, criterios o patrones específicos. Puede aplicarse a una amplia variedad de contextos y áreas, desde la organización de objetos físicos hasta la clasificación de datos en un sistema informático.

¿Cuándo conviene usar un método de ordenamiento?

En programación, el método de ordenamiento que elijas dependerá del contexto y los requisitos específicos de tu aplicación. Aquí hay algunas consideraciones para determinar cuándo conviene usar un método de ordenamiento

Tipos de ordenamientos:

Los 2 tipos de ordenamientos que se pueden realizar son: los internos y los externos.

Ordenamientos internos:

Son aquellos en los que los valores a ordenar están en memoria principal, por lo que se asume que el tiempo que se requiere para acceder cualquier elemento sea el mismo (a [1], a [500], etc.).

Ordenamientos externos:

Son aquellos en los que los valores a ordenar están en memoria secundaria (disco, cinta, cilindro magnético, etc.), por lo que se asume que el tiempo que se requiere para acceder a cualquier elemento depende de la última posición accesada (posición 1, posición 500, etc.).

Algoritmos de ordenamiento.

Internos:

1. Inserción (InsertionSort).
2. Selección (SelectionSort).
3. Intercambio.
4. Ordenamiento de árbol.
5. QuickSort.
6. MergeSort.

7. RadixSort.

Externos:

1. Straight merging.
2. Natural merging.
3. Balanced multiway merging.
4. Polyphase sort.
5. Distribution of initial runs.

Algoritmos de inserción.

En este tipo de algoritmo los elementos que van a ser ordenados son considerados uno a la vez. Cada elemento es “insertado” en la posición apropiada con respecto al resto de los elementos ya ordenados. Entre estos algoritmos se encuentran el de Inserción Directa, ShellSort, Inserción Binaria y Hashing.

Algoritmos de intercambio.

En este tipo de algoritmos se toman los elementos de dos en dos, se comparan y se “intercambian” si no están en el orden adecuado. Este proceso se repite hasta que se ha analizado todo el conjunto de elementos y ya no hay intercambios. Entre estos algoritmos se encuentran el Burbuja (BubbleSort) y QuickSort.

Algoritmos de selección.

En este tipo de algoritmos se “selecciona” o se busca el elemento más pequeño (o más grande) de todo el conjunto de elementos y se coloca en su posición adecuada. Este proceso se repite para el resto de los elementos hasta que todos son analizados. Entre estos algoritmos se encuentra el de Selección Directa.

Algoritmos de enumeración.

En este tipo de algoritmos cada elemento es comparado contra los demás. En la comparación se cuenta cuántos elementos son más pequeños que el elemento que se está analizando, generando así una “enumeración”. El número generado para cada elemento indicará su posición.

Los métodos de ordenamiento se dividen en simples y complejos.

Los métodos simples:

Inserción (o por inserción directa), Selección, Burbuja y ShellSort, en donde el último es una extensión al método de inserción, siendo más rápido.

Los métodos complejos:

El QuickSort (ordenación rápida) y el HeapSort.

Conclusión

Los diferentes métodos de ordenamiento tienen diferentes eficiencias en términos de tiempo y espacio. Algunos son más rápidos para conjuntos de datos pequeños, mientras que otros son más eficientes para conjuntos de datos grandes. Es importante considerar el tamaño del conjunto de datos al elegir un método de ordenamiento.