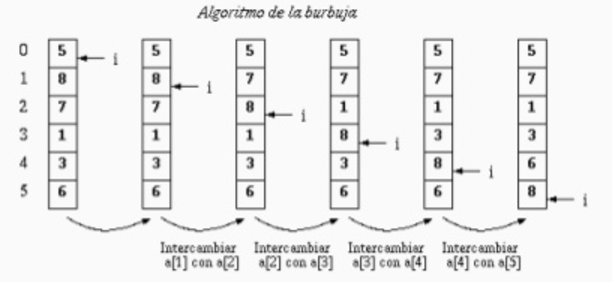
**ALGORITMOS DE ORDENAMIENTO**

* **BURBUJA**

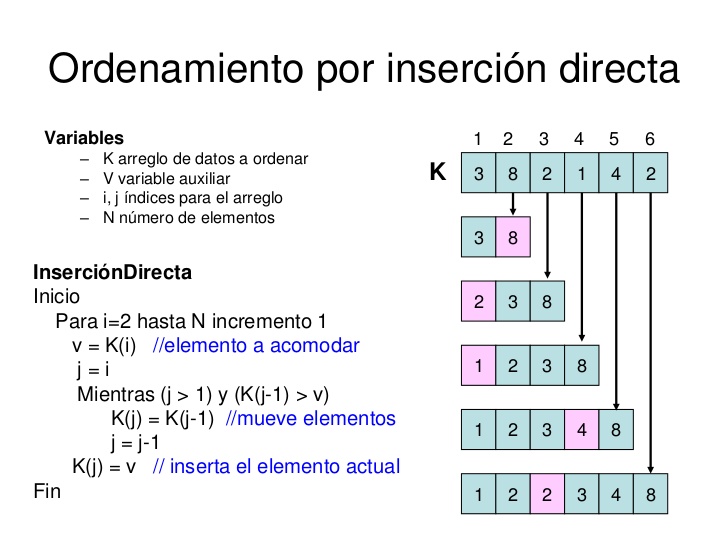
La **Ordenación de burbuja** es un sencillo [algoritmo de ordenamiento](https://es.wikipedia.org/wiki/Algoritmo_de_ordenamiento). Funciona revisando cada elemento de la lista que va a ser ordenada con el siguiente, intercambiándolos de posición si están en el orden equivocado. Es necesario revisar varias veces toda la lista hasta que no se necesiten más intercambios, lo cual significa que la lista está ordenada. Este [algoritmo](https://es.wikipedia.org/wiki/Algoritmo) obtiene su nombre de la forma con la que suben por la lista los elementos durante los intercambios, como si fueran pequeñas "burbujas". También es conocido como el **método del intercambio directo**. Dado que solo usa comparaciones para operar elementos, se lo considera un algoritmo de comparación, siendo uno de los más sencillo de implementar.



* **INSERCION**

El **ordenamiento por inserción** es una manera muy natural de ordenar para un ser humano, y puede usarse fácilmente para ordenar un mazo de cartas numeradas en forma arbitraria. Requiere **O(n²)** operaciones para ordenar una lista de **n** elementos.

Inicialmente se tiene un solo elemento, que obviamente es un conjunto ordenado. Después, cuando hay ***k*** elementos ordenados de menor a mayor, se toma el elemento ***k+1*** y se compara con todos los elementos ya ordenados, deteniéndose cuando se encuentra un elemento menor (todos los elementos mayores han sido desplazados una posición a la derecha) o cuando ya no se encuentran elementos (todos los elementos fueron desplazados y este es el más pequeño). En este punto se *inserta* el elemento ***k+1*** debiendo desplazarse los demás elementos.



* **QUICKSORT**

El **ordenamiento rápido** es un algoritmo creado por el científico británico en computación C.A.R Hoare basado en la técnica de divide y vencerás que permite, en promedio, ordenar *n* elementos en un tiempo proporcional a *n* log *n*.

El algoritmo trabaja de la siguiente forma:

* Elegir un elemento de la lista de elementos a ordenar, al que llamaremos **pivote**.
* Resituar los demás elementos de la lista a cada lado del pivote, de manera que a un lado queden todos los menores que él, y al otro los mayores. Los elementos iguales al pivote pueden ser colocados tanto a su derecha como a su izquierda, dependiendo de la implementación deseada. En este momento, el pivote ocupa exactamente el lugar que le corresponderá en la lista ordenada.
* La lista queda separada en dos sublistas, una formada por los elementos a la izquierda del pivote, y otra por los elementos a su derecha.
* Repetir este proceso de forma recursiva para cada sublista mientras éstas contengan más de un elemento. Una vez terminado este proceso todos los elementos estarán ordenados.

