

## **ALGORITMOS FUNDAMENTALES Y ARRAY**

**1-** Explique brevemente en qué consisten los siguientes Algoritmos Fundamentales.

Algoritmos de Ordenación: **Sirven para dar un orden determinado a los elementos de una lista.**

- Ordenamiento por Inserción: **Consiste en un ordenamiento que comienza con un solo elemento ordenado y luego se van agregando elementos de a uno a la vez, cada vez que ingresa un elemento se compara con los anteriores ya ordenados y los desplaza hacia la derecha hasta encontrar su lugar adecuado, repite este proceso hasta que ya no quedan elementos.**
- Algoritmo de la burbuja: **este algoritmo consiste en comparar cada elemento con el siguiente (por parejas), y si no se encuentra en el orden correcto se intercambian, el valor más pequeño se va intercambiando hasta llegar al principio de la lista.**
- Ordenamiento por selección: **este algoritmo consiste en buscar cada elemento de a uno a la vez, primero recorre la lista buscando el menor valor luego, cuando lo encuentra lo ubica en el primer lugar, luego hace lo mismo, pero comenzando desde el segundo elemento y colocándolo en el segundo lugar y así hasta que termine de ordenar toda la lista.**
- Algoritmo Quick-Sort: **Este algoritmo lo primero que hace es elegir un elemento, luego divide la lista en dos sublistas, una con los valores más pequeños y otra con los valores más grandes, después repite este proceso de forma recursiva con cada sublista, mientras estas tengan más de un elemento, hasta que quedan todos los elementos ordenados.**

Algoritmos de Búsqueda:

- Búsqueda Secuencial: **Se utiliza en listas no ordenadas. La búsqueda secuencial consiste en buscar el elemento dado, comprobando desde el primer elemento almacenado hasta llegar al que esta buscando o bien hasta que termine la lista sin haberlo encontrado.**
- Búsqueda Binaria: **Se utiliza en listas ordenadas. Consiste en no buscar en los segmentos donde sabemos que el valor no está, divide la lista a la mitad y elimina la mitad en la que sabemos que el elemento no está, así hasta que encuentra el elemento.**

Algoritmos de Recorrido:

- Recorrido de Profundidad: **En el recorrido en profundidad, todos los descendientes de un hijo se procesan antes del siguiente hijo. Para saber cómo regresarnos, vamos guardando los nodos visitados en una estructura de pila.**
- Recorrido de Anchura: **En el recorrido en anchura, cada nivel se procesa totalmente antes de que comience el siguiente nivel**

- 2-** Codifique mediante código JAVA los pseudocódigos de los algoritmos fundamentales presentados en el apunte "Algoritmos Fundamentales" correspondiente.
- Ordenamiento por inserción
  - Algoritmo de la burbuja
  - Ordenamiento por selección
  - Búsqueda secuencial
- 3-** Modifique los algoritmos de ordenamiento anteriores (inserción, burbuja, selección) para que además de la lista a ordenar revisan como parámetro un "String orden", cuyo valor será "ASC" o "DESC", que permita definir si la lista debe ordenarse de forma ASCENDENTE o DESCENDENTE.
- 4-** Codifique un algoritmo que permita cargar un array de una dimensión de 20 elementos enteros. Finalizada la carga de los 20 elementos el programa debe solicitar como se desea ordenar el array ASCENDENTE O DESCENDENTE y que método de ordenamiento aplicar (inserción, burbuja, selección), según las opciones elegidas aplique el algoritmos fundamental de ordenamiento que corresponda. Muestre por pantalla el array original desordenado y su resultante ordenado según las opciones elegidas.
- 5-** Codifique un algoritmo que solicite un 2 valores enteros mayores o iguales a 2, que se corresponden a cantidad de filas y columnas. Cree un arreglo de tamaño[X,Y]de tipo entero, el tamaño se corresponde a los 2 valores enteros ingresados inicialmente. Solicite al usuario que ingrese números enteros para cargar cada una de las celdas de la matriz. Finalizada la carga calcule el promedio entero de los elementos asignados y cargados en la matriz. Para terminar determine si el promedio calculado se encuentra asignado en alguna de las celdas, si es así indique la posición o posiciones correspondientes, caso contrario muestre el mensaje "No se encontró ninguna coincidencia".

**Ejemplo:** Ingreso 3 y 4 como valores enteros

Creo la matriz 3 filas por 4 columnas y cargo los valores:

3	15	6	9
6	13	4	4
14	8	5	<b>7</b>

El promedio entero es: **7**

El valor 7 se encuentra en la posición con índice **(2,3)** o Fila 3, Columna 4