



Investigación

Ordenamiento y Búsqueda en Java

En el tema de dato estructurado Arreglos cada estudiante debe realizar una investigación sobre los diferentes algoritmos de ordenación y búsqueda en el lenguaje de programación Java.

Para esto hay que leer el documento anexo que les ayudara a conocer características básicas del ordenamiento y búsqueda e indagar libros, sitios web y otras publicaciones sobre este tema, con el objetivo de ampliar los conocimientos sobre cada uno de los algoritmos puntualizados en la lectura.

Una vez que investiguen sobre cada algoritmo deben de conocer el funcionamiento, ventajas y desventajas del mismo con el objetivo de resolver los problemas planteados a continuación.

Guía Práctica del tema de Investigación Métodos de Ordenamiento y Búsqueda

Esta guía práctica del tema de investigación ordenamiento y búsqueda es importante realizarla ya que permitirá comprender y afianzar los conocimientos sobre estos algoritmos que son de mucha utilidad en el uso de arreglos. Deben de realizar los programas solicitados con los algoritmos correspondientes para evidenciar su trabajo de investigación.

1. Se trata de resolver el siguiente problema escolar. Dadas las notas de los alumnos de un colegio en el primer curso de bachillerato, en las diferentes asignaturas (5, por comodidad), se trata de calcular la media de cada alumno, la media de cada asignatura, la media total de la clase y ordenar los alumnos por orden decreciente de notas medias individuales.

Nota: Utilizar como algoritmo de ordenación el método Shell.

2. Un método de ordenación muy simple, pero no muy eficiente, de elementos $x_1, x_2, x_3, \dots, x_n$ en orden ascendente es el siguiente:

Paso 1: Localizar el elemento más pequeño de la lista x_1 a x_n ; intercambiarlo con x_1 .

Paso 2: Localizar el elemento más pequeño de la lista x_2 a x_n , intercambiarlo con x_2 .

Paso 3: Localizar el elemento más pequeño de la lista x_3 a x_n , intercambiarlo con x_n .

En el último paso, los dos últimos elementos se comparan e intercambian, si es necesario, y la ordenación se termina. Escribir un programa para ordenar una lista de elementos, siguiendo este método.

3. Se leen dos listas de números enteros, A y B de 100 y 60 elementos, respectivamente.

Se desea resolver mediante procedimientos las siguientes tareas:

- a) Ordenar aplicando el método de quicksort cada una de las listas A y B.
- b) Crear una lista C por intercalación de las listas A y B.
- c) Visualizar la lista C ordenada.

4. Partiendo del siguiente array encuentre el número de pasadas que realiza el algoritmo de ordenación de Burbuja para su ordenación.

47	3	21	32	56	92
----	---	----	----	----	----

5. Un vector contiene los elementos mostrados a continuación. Los primeros dos elementos se han ordenado utilizando un algoritmo de inserción. ¿Cuál será el valor de los elementos del vector después de tres pasadas más del algoritmo?

3	13	8	25	45	23	98	58
---	----	---	----	----	----	----	----

6. Partiendo del siguiente array encuentre las particiones e intercambios que realiza el algoritmo de ordenación quicksort para su ordenación.

8	43	17	6	40	16	18	97	11	7
---	----	----	---	----	----	----	----	----	---

7. Realice un programa que permita ordenar de menor a mayor o de mayor a menor un array con los sueldos de los N empleados de una empresa por cualquiera de los algoritmos de ordenación estudiados.

```
CA Console program output
1.- Leer los datos del array
2.- Ordenar de menor a mayor por el Metodo de la Burbuja
3.- Ordenar de mayor a menor por el Metodo de la Burbuja
4.- Ordenar de menor a mayor por el Metodo de Insercion
5.- Ordenar de mayor a menor por el Metodo de Insercion
6.- Ordenar de menor a mayor por el Metodo de QuickSort
7.- Ordenar de mayor a menor por el Metodo de QuickSort
8.- Salir
Introduce una opcion:
```

8. Escribir una función de búsqueda binaria aplicado a un array ordenado en modo descendente

9. Supongamos que se tiene una secuencia de n números que deben ser clasificados:

1. Utilizar el método de Shell para determinar cuántas comparaciones y cuántos intercambios se requieren para clasificar la secuencia si:

- Ya está clasificada.
- Está en orden inverso.

2. Repetir el paso 1 para el método de *quicksort*.

10. Un array contiene los elementos indicados más abajo. Utilizando el algoritmo de búsqueda binaria, trazar las etapas necesarias para encontrar el número 88.

8	13	17	26	44	56	88	97
---	----	----	----	----	----	----	----

Igual búsqueda pero aplicada al caso del número 20.

11. Se tiene la siguiente lista:

47 3 21 32 56 92

Después de dos pasadas de un algoritmo de ordenación, el array se ha quedado dispuesto así:

3 21 47 32 56 92

¿Qué algoritmo de ordenación se está utilizando (selección, burbuja o inserción)? Justifique la respuesta.

12. Se tiene el vector A con 100 elementos almacenados. Diseñe un programa que escriba "SI" si el vector esta ordenado ascendentemente o "NO" si el vector no está ordenado.

13. Diseñe un programa que lea un número cualquiera y lo busque en el vector X, el cual tiene almacenados 80 elementos. Escribir la posición donde se encuentra almacenado el número en el vector o el mensaje "NO" si no lo encuentra. Búsqueda secuencial.

14. Un ordenamiento burbuja puede modificarse para que “burbujee” en ambas direcciones. La primera pasada mueve la lista hacia “arriba”, y la segunda pasada la mueve hacia “abajo”. Este patrón alternante continúa hasta que no son necesarias más pasadas. Implemente esta variación y describa en qué circunstancias podría ser apropiada.
15. Realice un programa que haga un ordenamiento en una empresa de acuerdo al salario y código de los empleados. Usa dos métodos diferentes entre sí.

“El hombre necesita dificultades para poder disfrutar del triunfo.”
- Abdul Kalam -