

PRACTICA # 4: ALGORITMOS DE BÚSQUEDA PARTE 1

OBJETIVO: EL ESTUDIANTE IDENTIFICARÁ EL COMPORTAMIENTO Y CARACTERÍSTICAS DE LOS PRINCIPALES ALGORITMOS DE BÚSQUEDA POR COMPARACIÓN DE LLAVES.

ACTIVIDADES PARA EL DESARROLLO DE LA PRÁCTICA**Ejercicio 1. Listas En Java**

EL manejo de listas en el lenguaje de programación Java, se realiza a través de la interfaz List<E>, dicha interfaz cuenta con varias implementaciones. Las más utilizadas son ArrayList y LinkedList, la diferencia fundamental entre ambas es que la primera funciona con arreglos dinámicos y la segunda con listas ligadas dobles.

- Mediante la ayuda de NetBeans, crea un nuevo proyecto llamado "Practica4_[ApellidoP][Nombre]". En dicho proyecto agrega una clase principal y codifica el siguiente código.

```
package [...];

import java.util.LinkedList;
import java.util.List;

public class [...] {

    public static void main(String[] args) {
        List<Integer> lista1 = new LinkedList<>() ;

        lista1.add(15);
        [Agrega 5 instrucciones más de "add" con los elements que quieras]
        lista1.add(80);

        System.out.println(" Estado punto 1 ");
        imprimirLista(lista1);
        System.out.println(" *** ");

        lista1.add(1,300);
        lista1.add(3,500);
        lista1.add(5,700);

        System.out.println(" Estado punto 2 ");
        imprimirLista(lista1);
        System.out.println(" *** ");

        lista1.set(0, 4);
        lista1.set(2, 6);
        lista1.set(7, 8);

        System.out.println(" Estado punto 3 ");
        imprimirLista(lista1);
        System.out.println(" *** ");

        List<Integer> lista2;
        lista2 = lista1.subList(3, 6);
        imprimirLista(lista2);
        System.out.println(" *** ");
        System.out.println(lista1.equals(lista2));

    }

    public static void imprimirLista(List<Integer> listaPrint){
        for(Integer var : listaPrint){
            System.out.println(var);
        }
    }
}
```

CONTINÚA EN LA PARTE POSTERIOR...

- a. Explica la diferencia entre los métodos “set” y “add”
- b. Explica el funcionamiento del método “sublist”;
- c. Investiga (y agrega las instrucciones respectivas en el programa) cuales son los métodos de java para las siguientes operaciones
 - i) borrar un elemento de la lista
 - ii) para saber si es vacía
 - iii) para buscar algún elemento.

Ejercicio 2. Búsqueda Lineal

Agrega una nueva clase al proyecto llamada BusquedaLineal, en ella implementa 3 métodos para las 3 variantes de búsqueda lineal:

- a) Un método que devuelva verdadero si la clave (llave) se encuentra en la lista y falso en caso contrario
- b) Un método que devuelva el índice donde se encuentra la clave
- c) Un método que devuelva el número de veces que aparece la clave en la lista

Comprueba el funcionamiento de tus métodos en la clase principal

Ejercicio 3. Búsqueda Binaria

Agrega una nueva clase al proyecto llamada BusquedaBinaria, en ella implementa las siguientes versiones de búsqueda binaria

- a) Un método que devuelva verdadero si la clave se encuentra en la lista y falso en caso contrario.
- b) Un método que devuelva el número de veces que la clave se encuentra en la lista

Comprueba el funcionamiento de tus métodos en la clase principal

Ejercicio 4. Búsqueda en listas de objetos

- a) Agrega una nueva clase al proyecto donde definas los atributos y métodos de un Objeto llamado “EquipoFutbol” (el diseño es libre pero al menos debe tener 4 atributos)
- b) Agrega nuevos métodos en tus clases de búsqueda (binaria y lineal), basados en los métodos anteriores, con la diferencia de que ahora en lugar de recibir como parámetro “Integer”, reciban los siguientes parámetros:

- 1) Nombre del equipo (búsqueda por nombre)
- 2) Nombre Estado (búsqueda por estado)

- c) Crea en la clase principal, una lista de equipos de Futbol y comprueba el funcionamiento de los métodos realizados para las búsquedas.

Comprueba el funcionamiento de los métodos nuevos en la clase principal

5.- Escribe las conclusiones de tu práctica