

Universidad Mariano Gálvez De Guatemala

Campus De Boca Del Monte

Facultad De Ingeniería en Sistemas de Información

PROGRAMACIÓN III

Ing. Mélvín Cali

Tarea 3 Arboles B

Jonathan Joel Chán Cuellar

Carnét: 7690-22-1805

Fecha:03/04/2024

Enlace del GitHub

- <https://github.com/jonathanc4chan/tarea3de-programacion.git>

Funcionamiento de programa de árboles B.

Paso 1: ingresar el grado del árbol que va a ejecutar el programa

```
Bienvenido al programa de Árbol B
Un Árbol B es una estructura de datos de búsqueda que mantiene claves ordenadas y permite búsquedas, inserciones y eliminaciones eficientes.
Ingresa el grado del árbol B (debe ser un número entero mayor o igual a 2):
3
```


menú

```
Seleccione una opción:
1. Insertar claves
2. Eliminar clave
3. Buscar clave
4. Salir
Opción seleccionada:
```

Paso 2: seleccione la opción que el usuario prefiera en este caso la 1 que es para insertar las claves de los árboles al darle esa opción le va a pedir que ingrese la cantidad de claves a insertar que en este caso van a hacer 10.

```
Seleccione una opción:
1. Insertar claves
2. Eliminar clave
3. Buscar clave
4. Salir
Opción seleccionada: 1
Ingresa la cantidad de claves a insertar:
10
Ingresa las claves:
```

Ahora toca ingresar los números que van a hacer los nodos de los árboles o las claves para ello escogemos una serie de números los cuales nos van a servir como demostración para presentar como funciona nuestro programa.



```
Seleccione una opción:
1. Insertar claves
2. Eliminar clave
3. Buscar clave
4. Salir
Opción seleccionada: 1
Ingrese la cantidad de claves a insertar:
10
Ingrese las claves:
1
3
8
10
22
40
50
55
45
95
Clave 1 insertada correctamente.
Clave 50 insertada correctamente.
Clave 3 insertada correctamente.
Clave 22 insertada correctamente.
Clave 55 insertada correctamente.
Clave 8 insertada correctamente.
Clave 40 insertada correctamente.
Clave 10 insertada correctamente.
Clave 45 insertada correctamente.
Clave 95 insertada correctamente.
```

Ahora voy a mostrar que pasa si se ingresan números repetidos en el programa lo que el programa hará es no contarlos y pedir otro número en lugar del repetido.

```
Seleccione una opción:
1. Insertar claves
2. Eliminar clave
3. Buscar clave
4. Salir
Opción seleccionada: 1
Ingrese la cantidad de claves a insertar:
5
Ingrese las claves:
1
4
7
8
8
La clave 8 ya ha sido ingresada. Intente con otra clave.
4
La clave 4 ya ha sido ingresada. Intente con otra clave.
1
La clave 1 ya ha sido ingresada. Intente con otra clave.
7
La clave 7 ya ha sido ingresada. Intente con otra clave.
9
Clave 1 insertada correctamente.
Clave 4 insertada correctamente.
Clave 7 insertada correctamente.
Clave 8 insertada correctamente.
Clave 9 insertada correctamente.
```

Paso 3: ahora volvemos al menú y escogemos la opción 2 que es eliminar esto lo que hará es eliminar un numero de los que hemos ingresado con anterioridad.

```
Seleccione una opción:
1. Insertar claves
2. Eliminar clave
3. Buscar clave
4. Salir
Opción seleccionada: 2
Ingrese la clave a eliminar:
1
La clave 1 ha sido eliminada.
```

Ahora veremos que pasa si seleccionamos un numero que no es parte del árbol o que no se encuentra ingresado.

```
Seleccione una opción:
1. Insertar claves
2. Eliminar clave
3. Buscar clave
4. Salir
Opción seleccionada: 2
Ingrese la clave a eliminar:
22
Error: índice fuera de límites al eliminar.
La clave 22 no existe en el árbol.
```

Paso 4: ahora le daremos a la opción 3 que es la de búsqueda y para ello escogemos un numero de los que se encuentra en el árbol y lo buscamos.

```
Seleccione una opción:
1. Insertar claves
2. Eliminar clave
3. Buscar clave
4. Salir
Opción seleccionada: 3
Ingrese la clave a buscar:
8
La clave 8 está presente en el árbol.
```

Ahora que pasa si el numero que buscamos no existe el programa mostrara lo siguiente.

```
Seleccione una opción:  
1. Insertar claves  
2. Eliminar clave  
3. Buscar clave  
4. Salir  
Opción seleccionada: 3  
Ingrese la clave a buscar:  
22  
La clave 22 no está presente en el árbol.
```

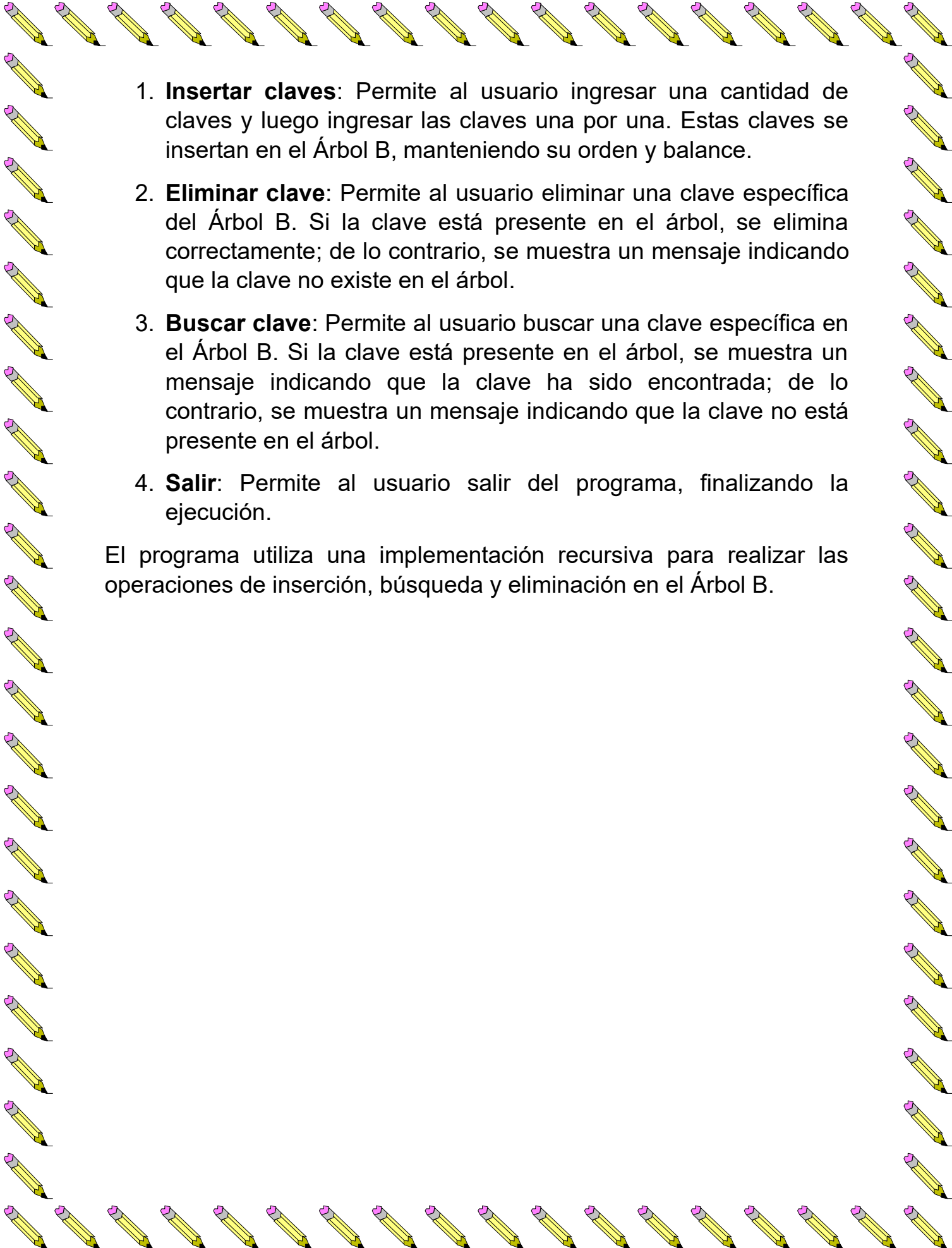
Paso 5: en este le damos a la opción 4 del menú y este nos sacara de programa finalizando su ejecución.

```
Seleccione una opción:  
1. Insertar claves  
2. Eliminar clave  
3. Buscar clave  
4. Salir  
Opción seleccionada: 4
```

Este programa implementa un Árbol B en Java, una estructura de datos utilizada para organizar y manipular un conjunto de claves de manera eficiente. El Árbol B es una estructura de datos balanceada que permite realizar operaciones de inserción, búsqueda y eliminación en tiempo logarítmico respecto al tamaño del árbol.

Al ejecutar el programa, se solicita al usuario que ingrese el grado del Árbol B, que determina el número máximo de claves que puede contener cada nodo del árbol. Una vez ingresado el grado, se crea un Árbol B vacío con el grado especificado.

El programa ofrece un menú de opciones que permite al usuario realizar diferentes operaciones en el árbol:

- 
1. **Insertar claves:** Permite al usuario ingresar una cantidad de claves y luego ingresar las claves una por una. Estas claves se insertan en el Árbol B, manteniendo su orden y balance.
 2. **Eliminar clave:** Permite al usuario eliminar una clave específica del Árbol B. Si la clave está presente en el árbol, se elimina correctamente; de lo contrario, se muestra un mensaje indicando que la clave no existe en el árbol.
 3. **Buscar clave:** Permite al usuario buscar una clave específica en el Árbol B. Si la clave está presente en el árbol, se muestra un mensaje indicando que la clave ha sido encontrada; de lo contrario, se muestra un mensaje indicando que la clave no está presente en el árbol.
 4. **Salir:** Permite al usuario salir del programa, finalizando la ejecución.

El programa utiliza una implementación recursiva para realizar las operaciones de inserción, búsqueda y eliminación en el Árbol B.