



SEED

“Simuladores para Estudio de Estructuras de Datos”

Manual de Usuario

Simulador ArbolB+<T>

Versión: 1.0

Universidad Francisco de Paula Santander
Programa Ingeniería de Sistemas
2014



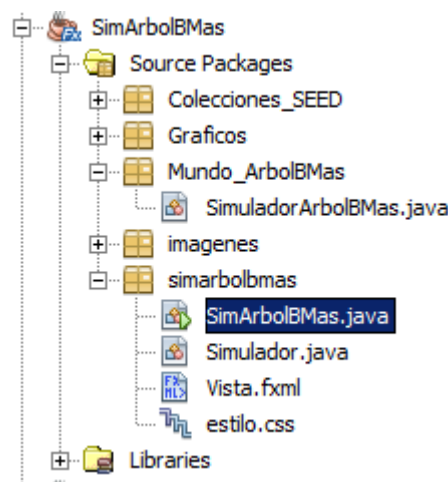
MANUAL DE USUARIO: Simulador “ArbolB+<T>”

Descripción General

El presente Manual de usuario pretende describir en detalle el conjunto de funcionalidades de la Aplicación desarrollada para la Simulación del comportamiento de la Estructura de Datos ArbolBMas<T>.

Dentro de esta aplicación encontrará el estudiante un conjunto de operaciones relacionadas con las funciones básicas implementadas para la estructura Árbol B+: Insertar, Eliminar y Buscar un dato dentro de cada una de las ramas del Árbol. Adicionalmente el estudiante podrá conocer algunas de las propiedades del Árbol como: Altura, peso y cantidad de Hojas presentes (ilustradas gráficamente).

Para la implementación de este Simulador se ha determinado la siguiente distribución de paquetes, ya conocida por el Estudiante, de forma que sea fácilmente apropiable a futuras modificaciones con el fin de hacer buen uso de esta aplicación.



“Directorio del Simulador para ArbolBMas<T>”

La implementación de este Simulador se desarrolla basada en la implementación del Simulador de Árbol B y las funciones (operaciones) desarrolladas para dicho Simulador. La diferencia de este simulador radica en el conjunto de propiedades a la hora de insertar y eliminar los datos en la estructura que lo hacen diferente a la estructura Árbol B.

A continuación se presenta la interface principal del simulador para “Árbol B+”.



"Interface principal del Simulador para ArbolBMas<T>"

Descripción de las Funcionalidades del Simulador

El simulador para ArbolBMas<T> permite al Estudiante:

1. Crear un Árbol B+:

Para Crear un nuevo Árbol B+ el estudiante deberá dar clic en el Botón "Crear", seleccionando previamente en el combo del Orden (cantidad mínima de datos en una página) que desea tener en la estructura, aunque esta estructura es dinámica, *"El Orden del Árbol B+ viene limitado a un valor mayor a cero (0) y menor a cuatro (4), por cuestiones de pintar la estructura"*.

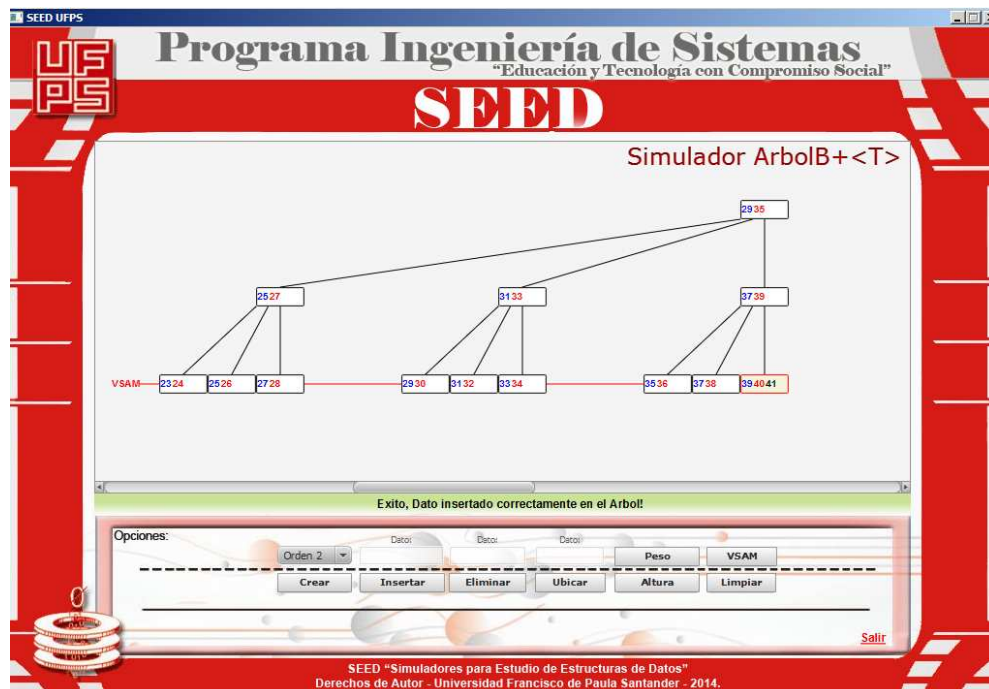


"Creación de un Árbol B+ de Orden 2".

2. Insertar Datos:

Para Insertar datos el estudiante simplemente deberá ingresar a la caja de texto el dato que desea insertar en el Árbol, este dato no puede ser menor a **-9** ni mayor a **99**, rango seleccionado por cuestiones de que no se desborde el número, además el árbol no puede superar la **altura 4** cuando el orden del Árbol es 1 ó **altura 3** para estructuras de orden 2 ó 3, con el fin de que la ilustración de la estructura no se desborde del Área de Dibujo del mediador pedagógico digital.

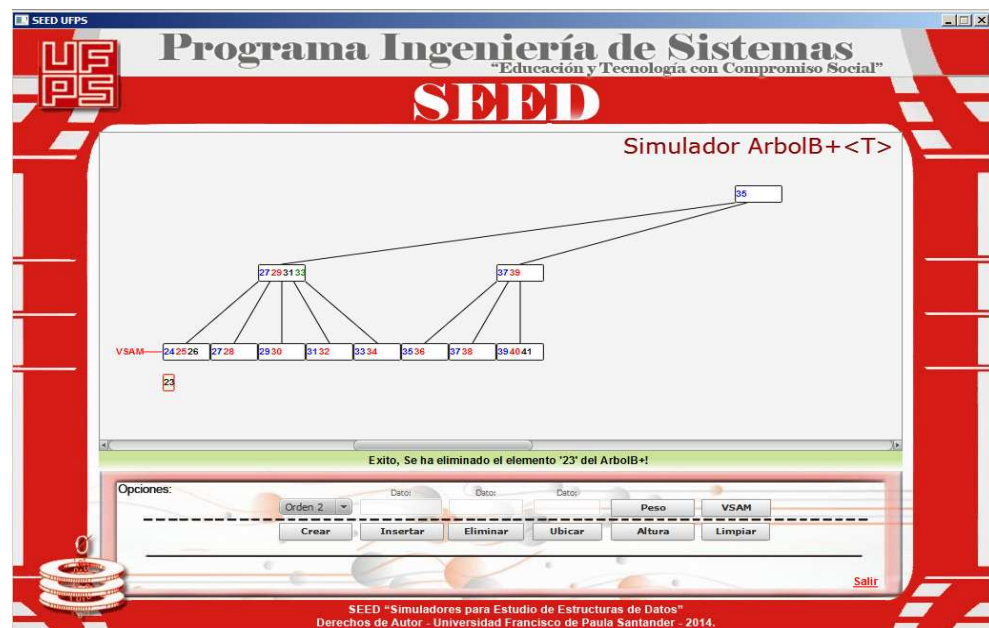
Esta estructura posee un elemento denominado VSAM que apunta a las hojas del árbol que son las que poseen la información del árbol el cual es graficado y las hojas del árbol se encuentran relacionadas esto es lo que diferencia esta estructura de los Arboles B.



“Inserción en el Árbol B+ de los datos: 23 – 24 – 25 – 26 – 27 – 28 – 29 – 30 – 31 – 32 – 33 – 34 – 35 – 36 – 37 – 38 – 39 - 40 y 41”.

3. Eliminar Datos:

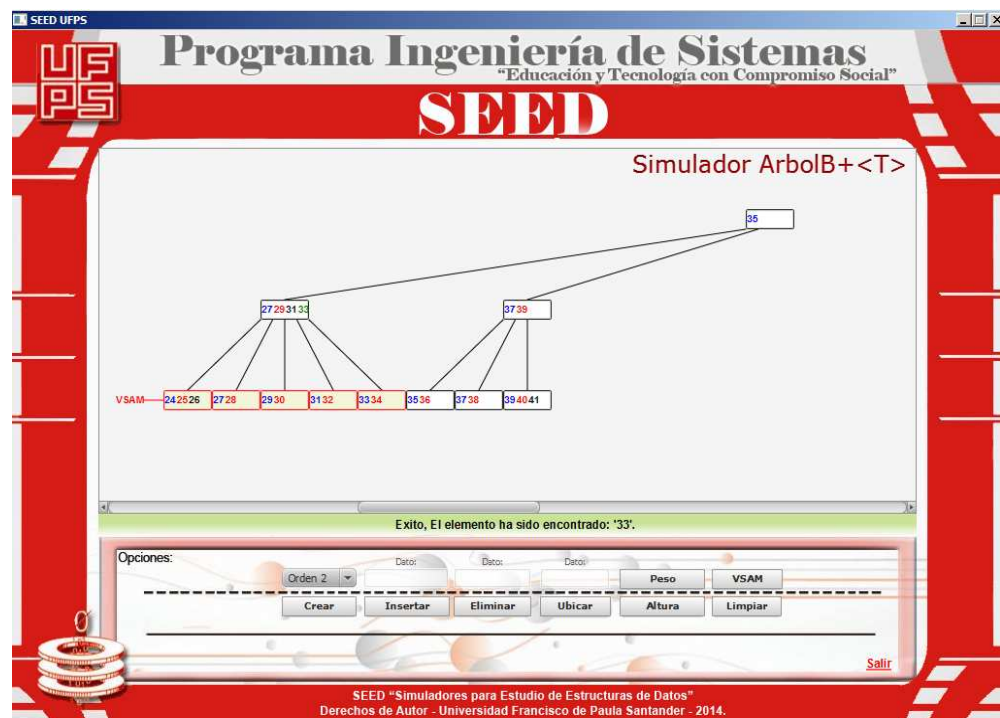
Para eliminar datos el estudiante deberá ingresar simplemente el dato que desea eliminar del Árbol B+ (previamente insertado) en la caja de texto. Una vez eliminado el dato del Árbol B+ este no será pintado e inmediatamente el Árbol ajustará sus páginas y será pintado correctamente en pantalla del Simulador.



“Eliminación del dato: 23 del Árbol B+. 23 es bajado del Árbol y desaparece”.

4. Búsqueda de un dato (Ubicar):

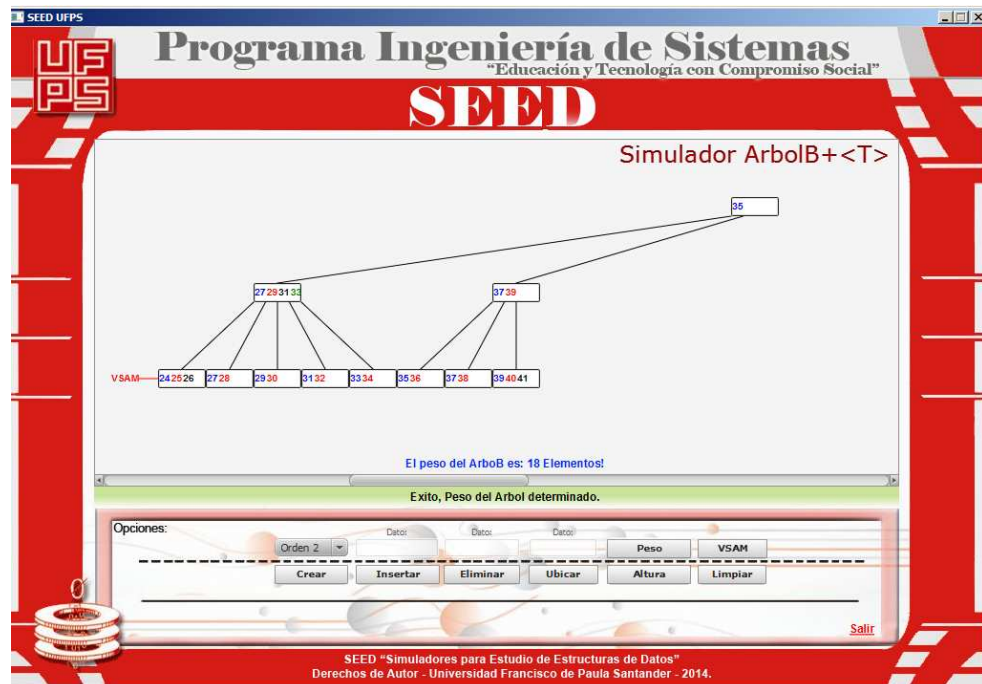
Para ubicar un dato dentro del Árbol B+ el estudiante deberá ingresar en la caja de texto el dato que desea ubicar. La búsqueda del dato se realiza de acuerdo a las propiedades del Árbol y el costo algorítmico que cada uno genera, para esta estructura se indica el camino de las páginas hojas de la búsqueda del dato teniendo como referencia el VSAM.



"Búsqueda del dato '33' dentro del Árbol B+."

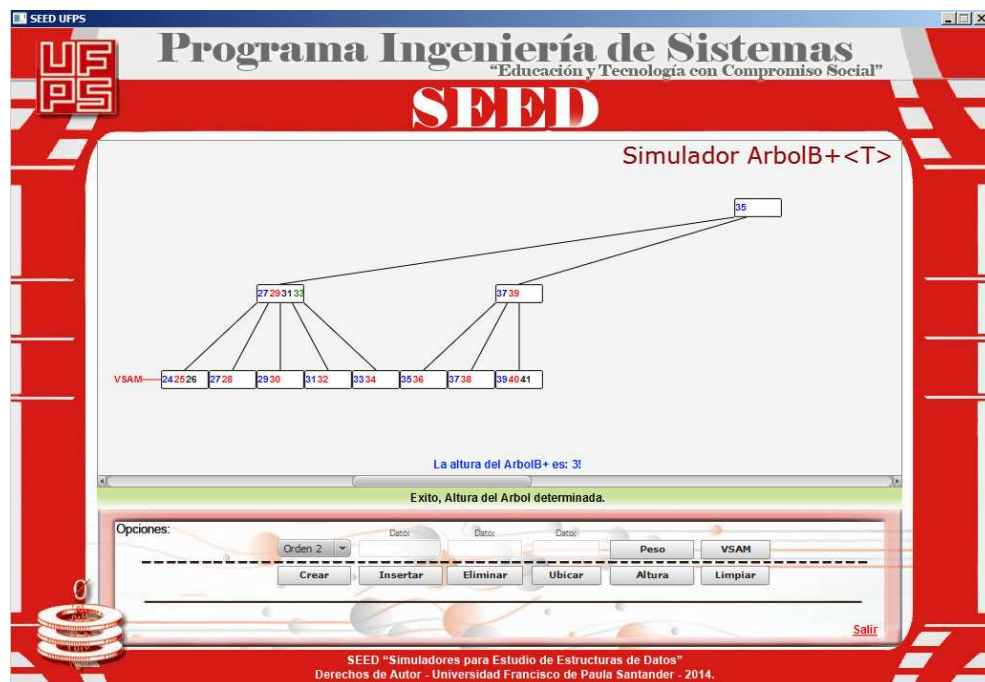
5. Determinar el Peso y Altura del Árbol B+

Para determinar el Peso del Árbol el estudiante deberá dar clic en el botón **Peso**, inmediatamente la aplicación indicará la cantidad de datos presentes en el Árbol B+ en un momento determinado.



"Determinar el peso del Árbol B+: 18 Elementos"

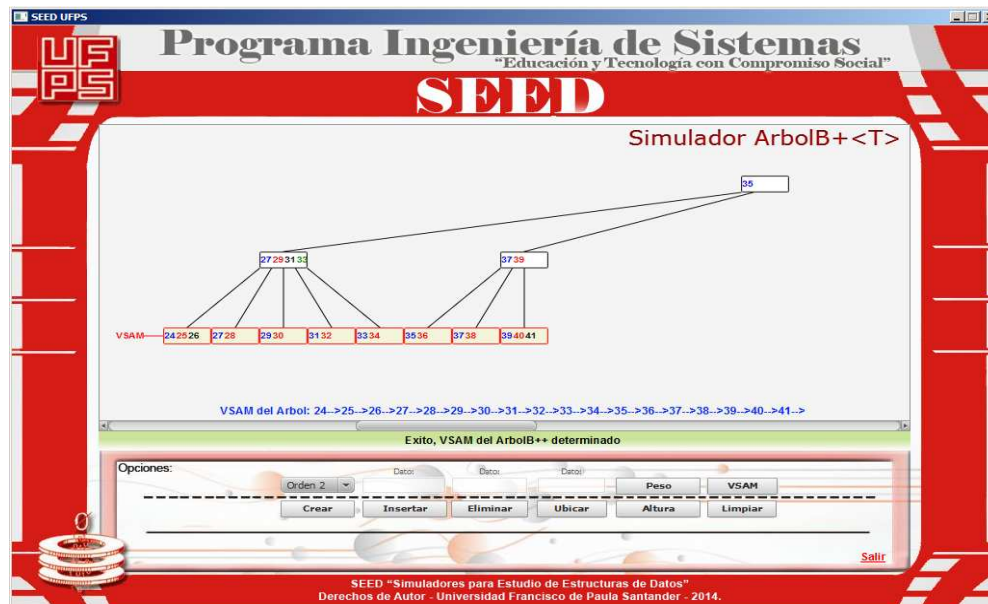
Ahora para determinar la Altura del Árbol el estudiante deberá dar clic en el botón **Altura**, inmediatamente la aplicación indicará la altura del Árbol B+ en un momento determinado.



"Determinar la altura del Árbol B+: Altura 3"

6. Listar VSAM

Para listar el VSAM del Árbol B+ el estudiante deberá dar **clic** en el botón **VSAM**, este será listado y la relación con las hojas de árbol serán pintadas en el Árbol para que el estudiante las pueda identificar el rol del Vsam en la estructura.



"Determinación de los elementos hoja del Árbol B+"

7. Adicionar nuevas funcionalidades:

Adicionalmente a las funciones incorporadas para el Simulador de Árbol B+, existe la posibilidad de que el estudiante pueda **"adicionar nuevas funcionalidades"** a la aplicación, de acuerdo a las actividades asignadas por los docentes o el interés propio de generar nuevos algoritmos en cada estructura y poder simularlos gracias a la herramienta gráfica del Simulador.

A continuación se presentan los pasos que deberá seguir el estudiante para crear una nueva funcionalidad dentro del Simulador de ArbolB+:

- 7.1. El estudiante debe generar el nuevo Algoritmo dentro de la Estructura de Datos **ArbolBMas**, presente en el paquete **SEED_Colecciones** y que desea adicionar a la funcionalidad del Simulador. (Para el ejemplo, se creará un algoritmo que permita limpiar el ArbolB+, es decir, eliminando sus datos dejando la raíz nula):

```
public void limpiarBMas() {  
    super.setRaiz(null);  
    this.vsam = super.getRaiz();  
}
```


7.2. A continuación el Estudiante debe generar un Método en la clase **SimuladorArbolBMas** del paquete **Mundo_ArbolBMas**, que realice el llamado al Método con el nuevo algoritmo creado en la Estructura de Datos **ArbolBMas**. Para el llamado debe utilizar el objeto creado en el Mundo **miArbolBMas**.

```
public void limpiarArbol() {
    this.miArbolBMas.limpiarBMas();
}
```

7.3. Por último, se deberá crear el componente grafico (para el ejemplo **Button**) que permita realizar el llamado al Método creado en **SimuladorArbolBMas**. Existe dos posibilidades para ello: Utilizar la herramienta “**JavaFX SceneBuilder**” para insertarlo, o agregar el código del Button en el Archivo **Vista.fxml**.

7.3.1. Utilizando JavaFX SceneBuilder



7.3.2. Insertando directamente el elemento en Vista.fxml

```
<Button layoutX="164.0" layoutY="611.0" mnemonicParsing="false" onAction="#btLimpiar" prefWidth="72.0" text="Limpiar"
```

Es importante resaltar, para ambos casos, que se debe asignar el evento “**OnAction**” del Button, para el ejemplo “**btLimpiar**”, el cual será el nombre del **Método** dentro de la clase **Controlador** que permite realizar la nueva funcionalidad del Simulador.

A continuación el Método dentro del paquete **simarbolbmas** en la clase **Simulador** que permite realizar el llamado a la nueva funcionalidad.

```
@FXML
private void btLimpiar() {
    try{
        if(this.simulador==null || !this.simulador.existeArbol()){
            impNota("Debe crear primero una ArbolB+",1);
            return;
        }
        if(this.simulador.estaVacioArbol()){
            impNota("El ArbolB+ no posee elementos!",1);
            return;
        }

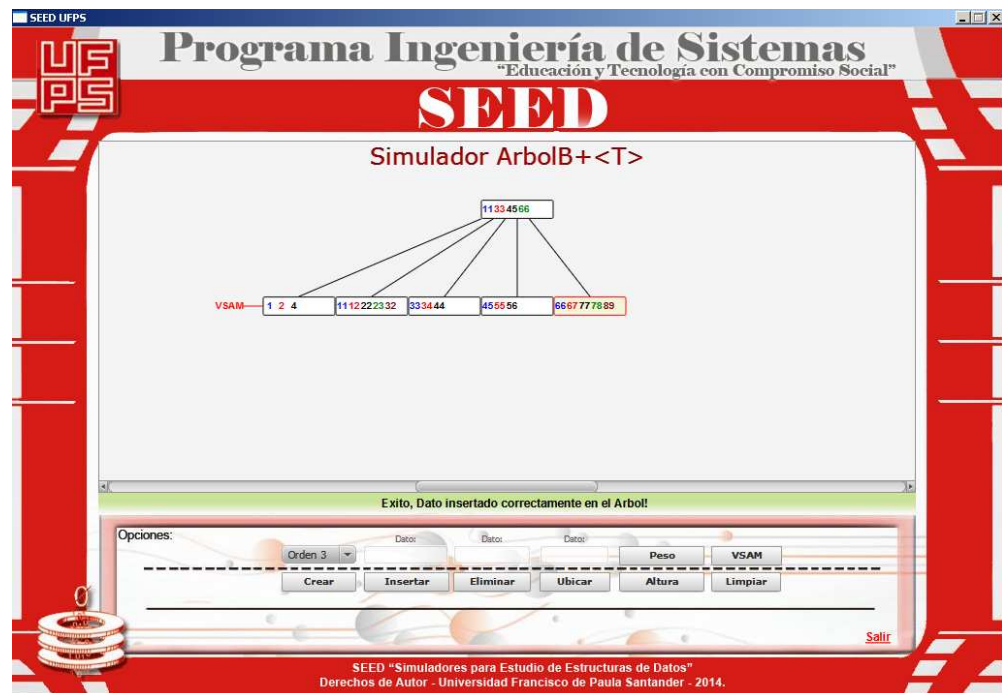
        //podar arbol
        this.simulador.limpiarArbol();
        this.pintarTDA();
        this.abg.animacion(4, 4,this.lineaTiempo);
        this.impNota("El ArbolB+ ha sido limpiado!", 0);

    }catch(Exception e){
        this.impNota("No se puede limpiar el Arbol!", 1);
    }
}
```

Además del llamado al Método creado en **SimuladorArbolBMas**, el estudiante deberá invocar el método que le permita volver pintar el Árbol, el cual siempre será “**pintarTDA()**”. Opcionalmente se recomienda enviar una mensaje con la respuesta a la operación realizada utilizando “**impNota(“Mensaje a enviar “ , tipo)”** donde tipo es cero (**0**) si en un mensaje Exitoso y uno (**1**) en caso de ser un mensaje erróneo.

A continuación se comprueba el funcionamiento del Algoritmo realizado.

Al Árbol en cuestión se le han ingresado cierta cantidad de datos:



Ahora se limpia el ArbolB+ gracias al nuevo algoritmo implementado:



"Se limpió el Árbol B+ "