



# SEED

“Simuladores para Estudio de Estructuras de Datos”

## Manual de Usuario

*Simulador ArbolSplay<T>*

*Versión: 1.0*

Universidad Francisco de Paula Santander  
Programa Ingeniería de Sistemas

2014



## MANUAL DE USUARIO: Simulador “ArbolSplay<T>”

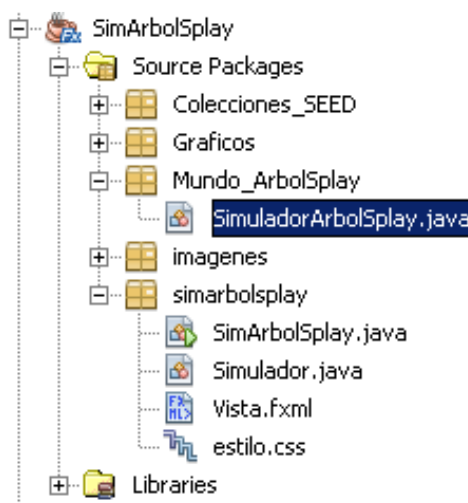
### Descripción General

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 ArbolSplay<T>.

Dentro de esta aplicación encontrará el estudiante un conjunto de operaciones relacionadas con las funciones básicas implementadas para la estructura Árbol Splay: 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).

Por último se implementa un conjunto de animaciones que pretenden ilustrar las diversas formas en que pueden ser recorridos los Arboles Binarios como el Splay, de manera que sea agradable para el estudiante la forma de comprender cada recorrido: preOrden, inOrden, postOrden y por Niveles.

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 ArbolSplay<T>”*

La implementación de este Simulador se desarrollada basada en la implementación del Simulador de Árbol Binario 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 el

Splay que lo hacen diferente a los demás Árboles que heredan la estructura del Árbol Binario General.

A continuación se presenta la interface principal del simulador para “ÁrbolSplay”.



*"Interface principal del Simulador para ArbolSplay<T>"*

## **Descripción de las Funcionalidades del Simulador**

El simulador para ArbolSplay<T> permite al Estudiante:

### **1. Insertar Datos:**

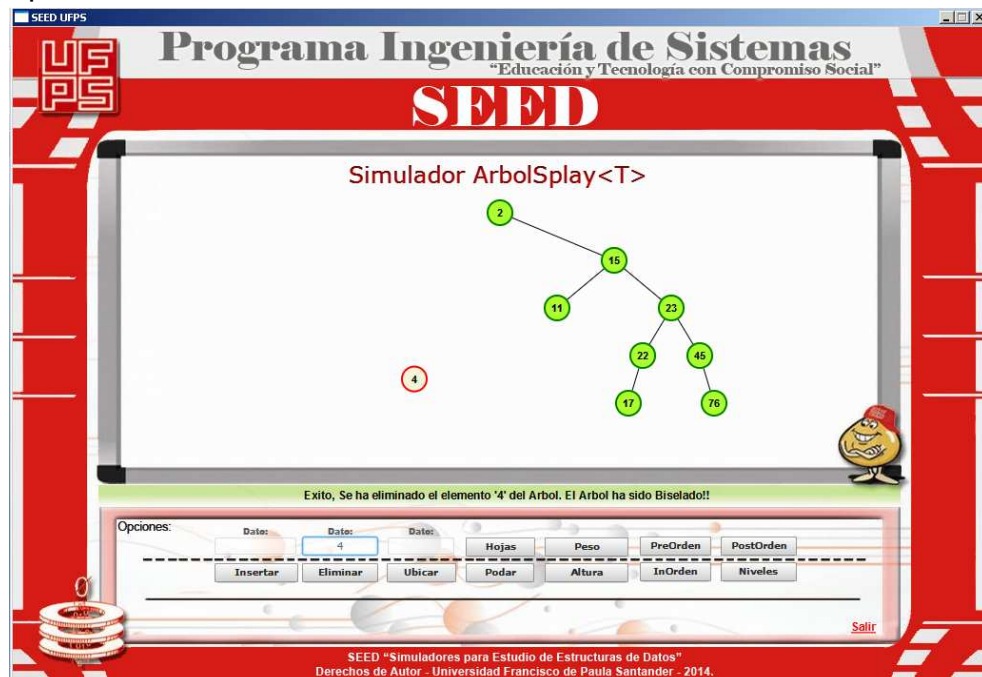
Para Insertar datos el estudiante simplemente deberá ingresar a la caja de texto el dato que desea insertar en el Árbol. A continuación el ArbolSplay, debido a sus propiedades, llevará el dato insertado a la raíz, por lo que realizara las rotaciones correspondientes para ello.



"Inserción en el ÁrbolSplay de los datos: 15 – 4 – 22 – 2 – 11 – 17 – 45 – 76 y 23"

## 2. Eliminar Datos:

Para eliminar datos el estudiante deberá ingresar simplemente el dato que desea eliminar del ÁrbolSplay (previamente insertado) e inmediatamente el Árbol será bicelado y pintado correctamente en pantalla del Simulador.



"Eliminación del dato: 4 del ÁrbolSplay. 4 es bajado del Árbol y desaparece"

### 3. Búsqueda de un dato (Ubicar):

Para ubicar un dato dentro del ÁrbolSplay 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 el ÁrbolSplay, el Árbol es biselado por lo que el dato se ubica en la raíz a la hora de ser buscado.



*"Búsqueda del dato '17' dentro del ÁrbolSplay"*

### 4. Obtener las Hojas del Árbol

Para obtener los elementos Hoja del Árbol Splayel estudiante deberá dar **clic** en el botón **Hojas**, y estas serán pintadas en el Árbol para que el estudiante las pueda identificar.





*"Determinación de los elementos hoja del Árbol Splay"*

## 5. Podar el ÁrbolSplay

El estudiante podrá podar las hojas de un ÁrbolSplay oprimiendo **clic** en el botón **podar** de la aplicación. Los elementos Hoja del árbol serán eliminados de forma que las hojas del Árbol Splay serán ahora los padres de estos elementos.



*"Podar el Árbol Splay anterior. Las hojas han sido eliminadas"*

## 6. Determinar el Peso y Altura del Árbol Splay

Para determinar el Peso del Árbol el estudiante deberá oprimir el botón **peso**, inmediatamente la aplicación indicará la cantidad de datos presentes en el Árbol Splay en un momento determinado.



*"Determinar el peso del Árbol Splay: 6 Elementos"*

Ahora para determinar la Altura del Árbol el estudiante deberá oprimir el botón **altura**, inmediatamente la aplicación indicará la altura del Árbol Splay en un momento determinado.



*"Determinar la altura del Árbol Splay: Altura 4"*

## 7. Recorridos en el Árbol Splay

Para recorrer el Árbol Splay estudiante deberá seleccionar el recorrido que desea realizar en el Árbol e inmediatamente el simulador realizara una animación. Debe dar clic en el botón: **preOrden**, **inOrden**, **postOrden** y **por Niveles**.

Recorrido en preOrden



Recorrido en inOrden





## Recorrido en postOrden

SEED UFPS

**Programa Ingeniería de Sistemas**  
"Educación y Tecnología con Compromiso Social"

**SEED**

Simulador ArbolSplay<T>

Recorrido en PostOrden: 2, 15, 10, 17, 45, 23, 22, 20, 18.

Exito, Se esta recorriendo el Arbol en PostOrden!

Opciones:

Date:	Date:	Date:	Hojas	Peso	PreOrden	PostOrden
Insertar	Eliminar	Ubicar	Podar	Altura	InOrden	Niveles

Salir

SEED "Simuladores para Estudio de Estructuras de Datos"  
Derechos de Autor - Universidad Francisco de Paula Santander - 2014.

## Recorrido por Niveles

SEED UFPS

**Programa Ingeniería de Sistemas**  
"Educación y Tecnología con Compromiso Social"

**SEED**

Simulador ArbolSplay<T>

Recorrido por Niveles: 18, 17, 20, 10, 22, 2, 15, 23, 45.

Exito, Se esta recorriendo el Arbol por Niveles!

Opciones:

Date:	Date:	Date:	Hojas	Peso	PreOrden	PostOrden
Insertar	Eliminar	Ubicar	Podar	Altura	InOrden	Niveles

Salir

SEED "Simuladores para Estudio de Estructuras de Datos"  
Derechos de Autor - Universidad Francisco de Paula Santander - 2014.

## 8. Adicionar nuevas funcionalidades:

Adicionalmente a las funciones incorporadas para el Simulador de Árbol Splay, 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 ArbolSplay:

8.1. El estudiante debe generar el nuevo Algoritmo dentro de la Estructura de Datos **ArbolSplay**, presente en el paquete **SEED\_Colecciones** y que desea adicionar a la funcionalidad del Simulador. (Para el ejemplo, se creará un algoritmo que permita obtener el código **Lukasiewicz** del Árbol Splay:

```
/**
 * Metodo que retorna el codigo Lukasiewicz del arbol binario; Este codigo etiqueta los nodos internos con "a"
 * y los externos con una "b" y realiza el recorrido en preorden con estas convenciones. <br>
 * <b>post: </b> Se retorno el codigo Lukasiewicz del Árbol Binario.<br>
 * @return un String con el codigo Lukasiewicz del arbol binario
 */
public String Luca()
{
    return(Luca(this.raiz));
}

/**
 * Metodo que retorna el codigo Lukasiewicz del arbol binario; Este codigo etiqueta los nodos internos con "a"
 * y los externos con una "b" y realiza el recorrido en preorden con estas convenciones. <br>
 * <b>post: </b> Se retorno el codigo Lukasiewicz del arbol binario.<br>
 * @param r represente la raiz del arbol, o raiz de subarbol
 * @return un String con el codigo Lukasiewicz del arbol binario
 */
private String Luca(NodoBin<T> r) {
    if(r==null)
        return("b");
    return("a"+Luca(r.getIzq())+Luca(r.getDer()));
}
```

8.2. A continuación el Estudiante debe generar un Método en la clase **SimuladorArbolSplay** del paquete **Mundo\_ArbolSplay**, que realice el llamado al Método con el nuevo algoritmo creado en la Estructura de Datos **ArbolSplay**. Para el llamado debe utilizar el objeto creado en el Mundo **miArbolSplay**.

```
public String getLuca(){
    return " Lukasiewicz: " + this.miArbolSplay.Luca();
}
```

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

### 8.3.1. Utilizando JavaFX SceneBuilder



### 8.3.2. Insertando directamente el elemento en Vista.fxml

```
<Button layoutX="164.0" layoutY="610.0" mnemonicParsing="false" onAction="#btLuca" prefWidth="135.0" text="Dar Lukasiewicz" />
```

Es importante resaltar, para ambos casos, que se debe asignar el evento **“OnAction”** del Button, para el ejemplo **“darLuca”**, 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 **simarbolsplayen** la clase **Simulador** que permite realizar el llamado a la nueva funcionalidad.

```
@FXML
private void btLuca() {
    this.pintarTDA();
    msg.setText(this.simulador.getLuca());
    msg.setVisible(true);
    this.impNota("Codigo Lukasiewicz calculado!", 0);
}
```

Además del llamado al Método creado en **SimuladorArbolSplay**, 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:



"Código Luca del Árbol Splay".