**Estructura de datos**(Árboles AVL)

Insertar un nodo, mostrarlo en pre-orden, orden y pos-orden (ordenamiento de árboles) y borrado con interfaz gráfica.

Después de todo eso que imprima factor de equilibrio, ósea ceros y unos y la profundidad, el programa guarde en txt el pre-orden

***Requerimientos***

1. Generar árbol AVL
2. Insertar nodo
3. Imprimir en pre-orden, post-orden y in-orden
4. Imprimir factor de equilibrio
5. Imprimir profundidad
6. Guardar en un archivo txt el pre-orden

***NOTA IMPORTANTE:***

***Se debe de eliminar toda carpeta o archivo de git, ya que estos archivos guardan información del autor del código. (Son carpetas ocultas o archivos con el nombre como “.git”)***

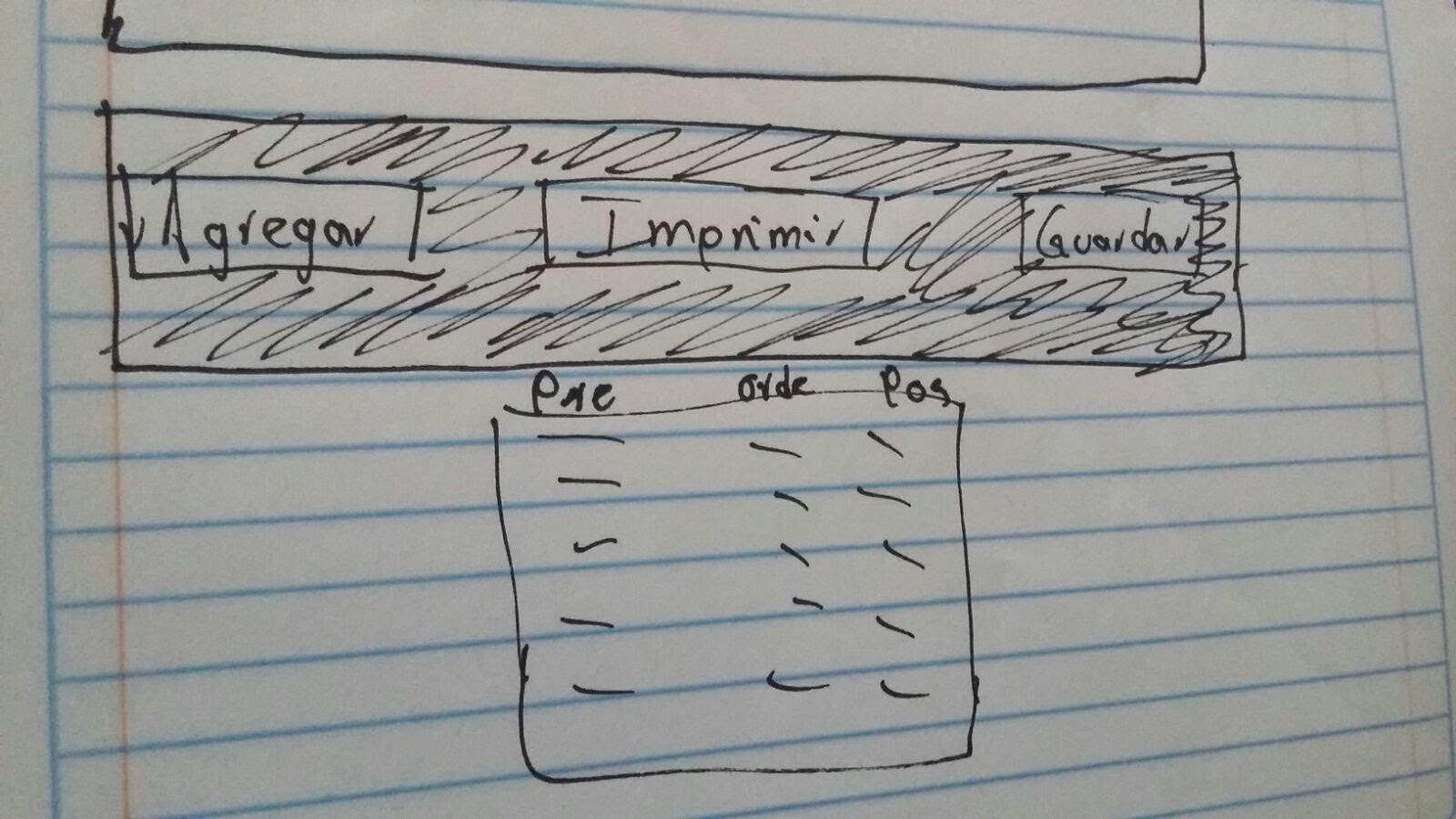
***Para ver la descripción del código en la pagina de github cambiar en la rama (Branch) a Descripcion y en el caso que solo se quiera tener el puro código en la rama (Branch) master.***

*Luna Tagano*

Insertar nodos, que cuando ingrese un número que rompa el equilibrio del árbol no deje agregarlo, que imprima en pre-orden, pos-orden y orden con su respectivo punto de equilibrio a lado de cada número y la profundidad del árbol, que guarde en pre-orden, con interfaz?

bueno el guardado en txt y al imprimir solo los números

La primera sería como aparecería y si se le la Imprimir que aparezca así u.u, los número en las tes formas con sus puntos de equilibrio alado



***Requerimientos.***

1. Insertar nodos
2. Si rompe el equilibrio no permitir agregarlo
3. Imprimir en Pre-orden, Pos-orden y Orden con su respectivo punto de equilibrio y profundidad
4. Guardar en txt en pre-orden

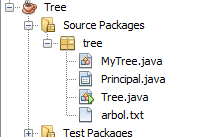
***Análisis***

1. Al iniciar el **Programa** ya debe estar estructurado un árbol equilibrado (AVL).
2. Se muestran tres **botones** *Agregar*, *Imprimir* y *Guardar*.
3. Un **espacio** inferior para ***imprimir***lo requerido.
4. Al dar clic en **Agregar** preguntar el valor del nodo a agregar.
   1. Si el nodo rompe el equilibrio mostrar mensaje que no permite agregarse.
   2. En caso contrario agregarlo.
5. Al dar clic en Imprimir mostrar mensaje para seleccionar la forma (Pre, pos u orden).
   1. En una columna mostrar el **valor del nodo**.
   2. En la segunda columna imprimir el **factor de equilibrio**.
   3. Y en la tercera columna imprimir el **valor de profundidad**.
6. Al dar clic en el botón Guardar, generar un archivo **txt** con el contenido del árbol en pre-orden

***Descripción código.***

El proyecto está constituidos de los siguientes

Archivos:



* *MyTree.java* es la clase que contiene los métodos para generar el Árbol
* *Principal.java* es la clase que contiene la interfaz gráfica del programa
* *Tree.java* Contiene el método “main” para que inicie el programa, además de que inicializa un Árbol con datos.
* *Árbol.txt* es el archivo generado con el árbol en post orden.

**El programa inicia en Tree.java, en esta clase agrega un valor al árbol e inicia la interfaz gráfica (Principal.java)**

**En la interfaz gráfica se encuentran los botones Agregar, Imprimir y guardar, además de un área con label para mostrar el árbol cuando se imprima.**

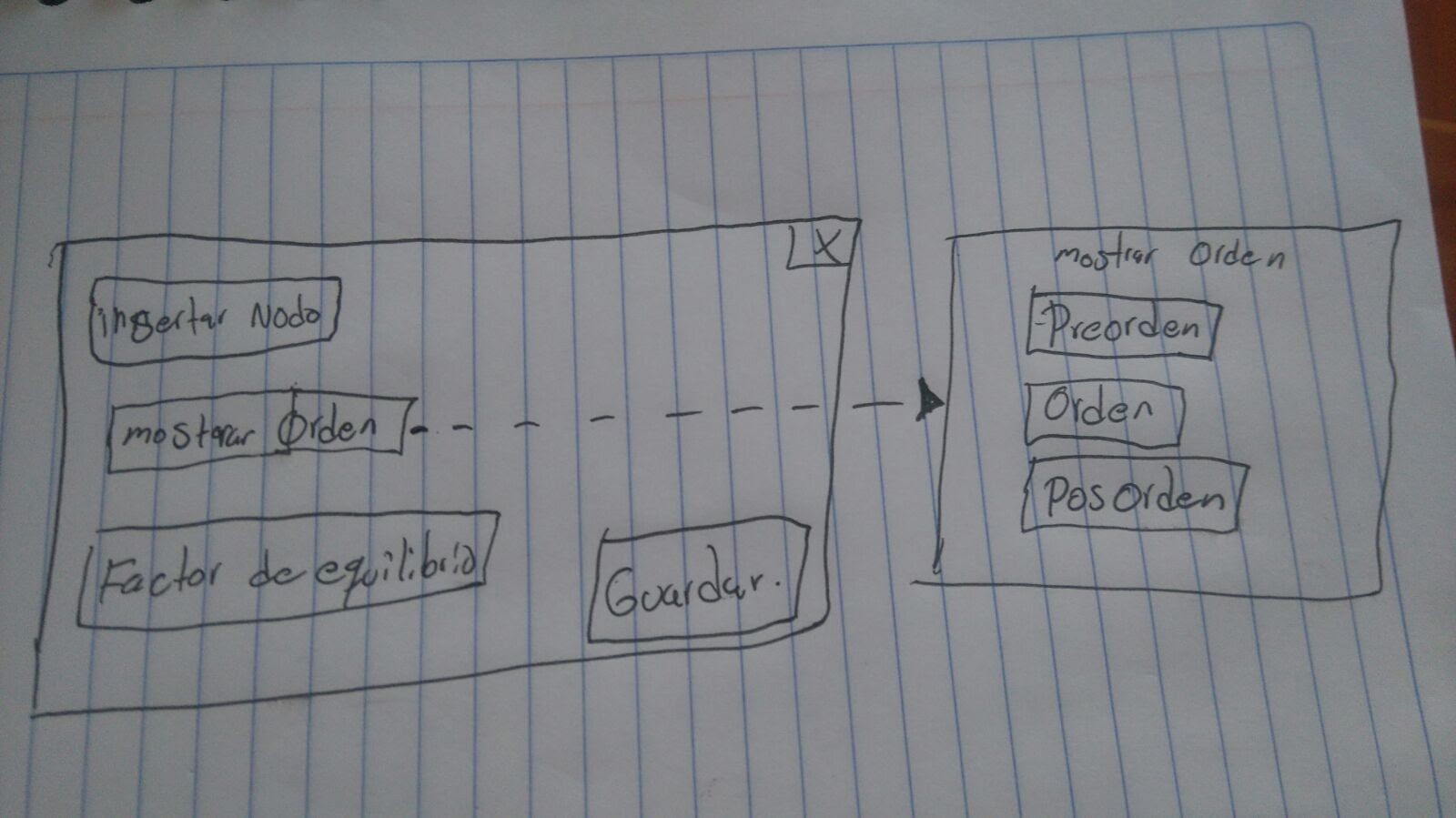
**Al presionar Agregar invoca un método de inserción de MyTree.**

**Al presionar Iprimir invoca un método para recorrer el árbol de MyTree y lo muestra en el label.**

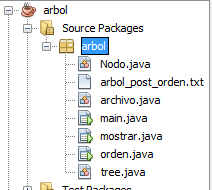
**Al presionar guardar invoca un método para crear un archivo desde MyTree.**

**Para mayor descripción leer comentarios en el código.**

Beto



El proyecto cuenta con los siguientes archivos.



* ***Nodo.java*** clase que contiene la unidad minima del árbol.
* ***Arbol\_post\_orden.txt*** Archivo generado al guardar el árbol.
* ***Archivo.java*** Clase que generar el archivo txt.
* ***Main.java*** Clase principal que genera la interfaz gráfica del programa.
* ***Mostrar.java*** Ventana que muestra el contenido del árbol
* **Orden.java** Ventana que muestra las opciones para desplegar el árbol : Pre, In o Post
* ***Tree.java*** Clase con toda la implementación del árbol.

El programa inicia en Main.java Muestra los botones Insertar…, Mostrar…, Factor…, y Guardar.

Declara una variable de tipo tree, y tree a su vez depende de una variable de tipo nodo.

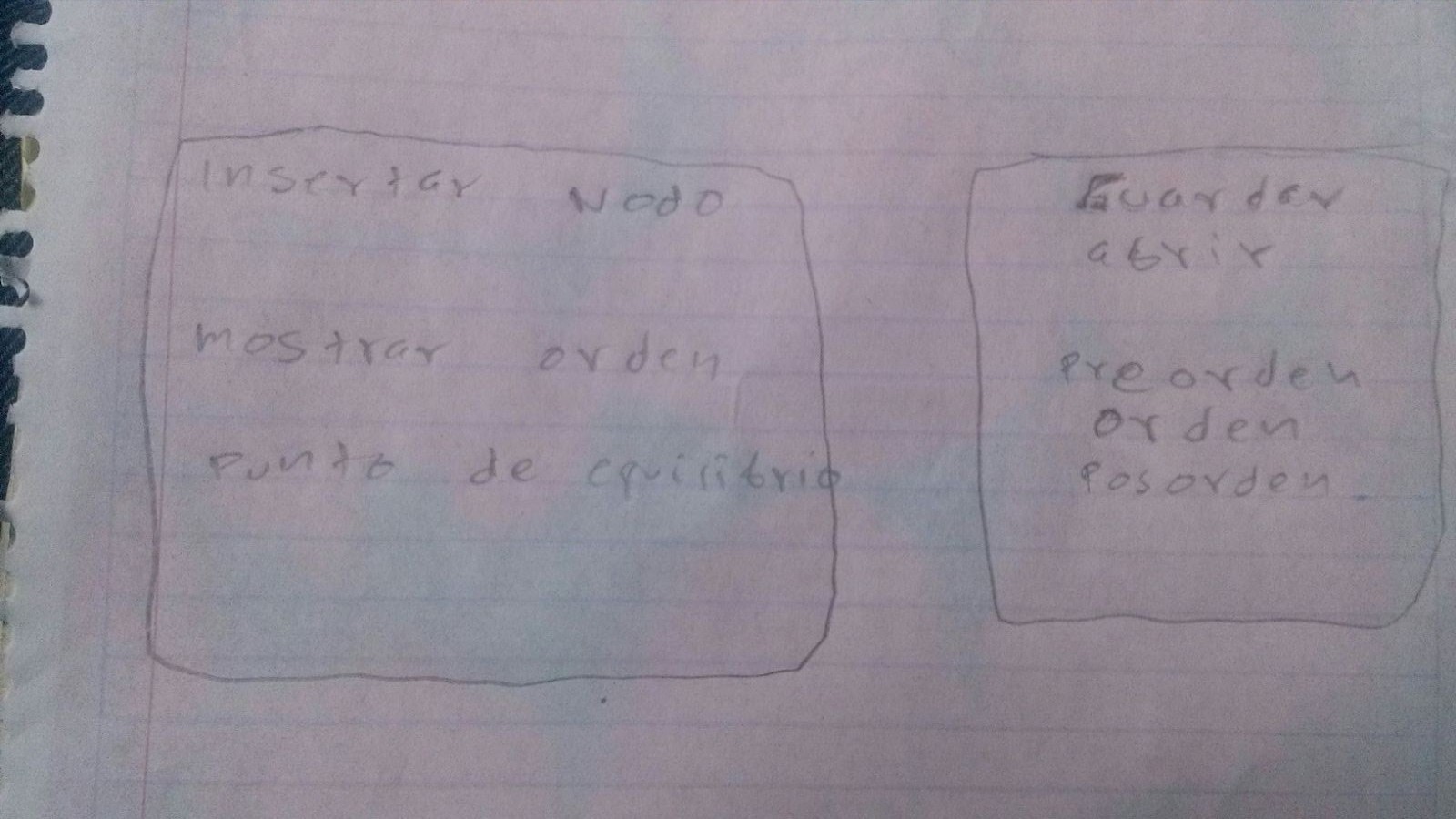
Para el botón insertar invoca un método en la clase tree, para el botón mostrar visualiza la interfaz gráfica de Orden.java, para el botón factor de equilibrio invoca un método en tree para obtenerlo y por ultimo el botón guardar invoca una instancia de la clase archivo

En cada botón de la ventana Orden.java invoca un método que recorre el árbol desde la clase tree. Al obtener la cadena con el contenido del árbol lo pasa a la interfaz Mostrar.java para que se despliegue dentro de un label.

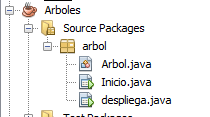
Para mayor información consultar comentarios en código.

Barradas José

Insertar nodo que imprima en preorden y posorden y que defina el punto de equilibrio la profundidad del arbol que guarde en archivo txt y lo pueda abrir para poderlo editar



El proyecto consta de los siguientes archivos:



* ***Arbol.java*** Es la clase que contiene todos los métodos para manipular un Árbol.
* ***Inicio.java*** Es la interfaz grafica del proyecto
* ***Despliega.java*** Es la GUI en la que se muestra el contenido del árbol.

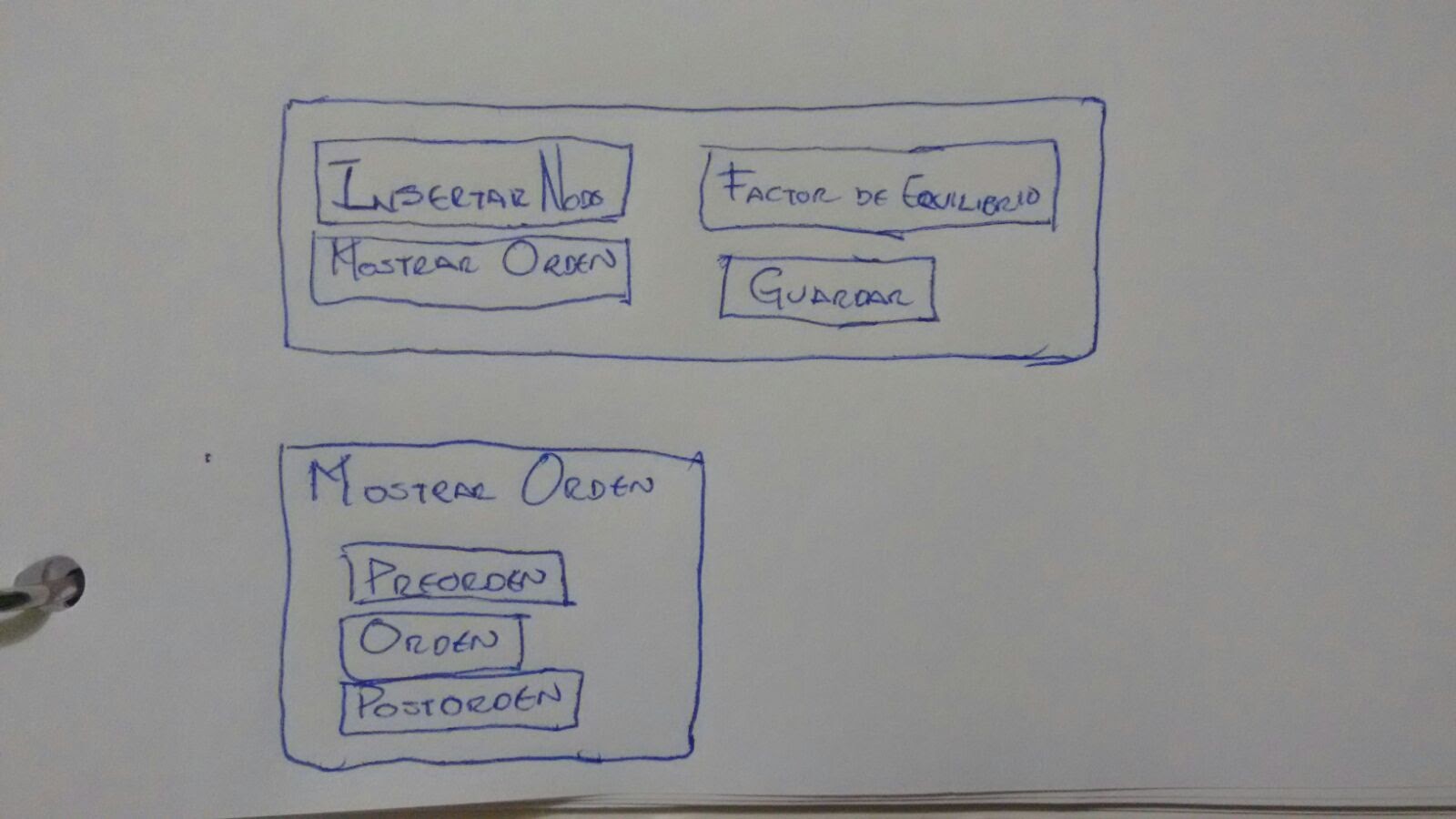
El programa comienza en Inicia.java generando una instancia de la clase Arbol.java y por cada botón invoca un método localizado en la misma clase Árbol.java

En este caso los botones para insertar, mostrar y punto de equilibrio invocan métodos que se encuentran dentro de la clase árbol.

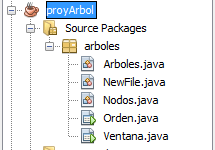
Cuenta con un menú con la opción de Guardar, mismo que invoca un método que crea el archivo en la clase Arbol.java. (El archivo generado esta ubicado en la raíz del proyecto dentro de la carpeta Arboles)

Para mayor información consultar comentarios en el código.

Antonio HedzC



El proyecta consta de los siguientes archivos:



Arboles.java Implementa todos los métodos para trabajar con un árbol.

NewFile.java Genera el archivo txt con el contenido del árbol.

Nodos.java Es la clase con la unidad minima del árbol.

Orden.java es la GUI con las opciones de recorrido del árbol.

Ventana.java Es la clase principal del programa que muestra la interfaz grafica.

El proyecto inicia desde la clase Ventana, mostrando las opciones de insertar, factor de equilibrio, mostrar orden y guardar. Para el botón insertar y factor de equilibrio invoca un método de la clase arboles, para el botón mostrar hace visible la GUI de la clase Orden.java para seleccionar el recorrido del árbol y por ultimo el botón guardar instancia a la clase NewFile para generar el txt con el contenido del árbol.

Para mayor información consultar comentarios en el código del proyecto.