UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

ESTRUCTURA DE DATOS

CATEDRÁTICO: EDGAR RENÉ ORNELIS HOÍL

TUTOR ACADÉMICO: ELIAN SAÚL ESTRADA URBINA



**MANUAL TÉCNICO**

MARIANA ABIGAIL MEJÍA GARCIA

CARNÉ: 202300370

GUATEMALA, 17 DE DICIEMBRE DEL 2,024

Contenido

[**INTRODUCCION** 3](#_Toc180449551)

[**OBJETIVOS** 4](#_Toc180449552)

[**General** 4](#_Toc180449553)

[**Especifico** 4](#_Toc180449554)

[**ALCANCES DEL SISTEMA** 5](#_Toc180449555)

[**ESPECIFICACIÓN TÉCNICA** 6](#_Toc180449556)

[ REQUISITOS DEL HARDWARE 6](#_Toc180449557)

[ REQUISITOS DEL SOFTWARE 6](#_Toc180449558)

[**TABLA TOKENS** 10](#_Toc180449559)

[**METODO EL ARBOL** 11](#_Toc180449560)

[**GRAMÁTICA** 13](#_Toc180449561)

# 

# **INTRODUCCION**

Tiene como finalidad hacer saber el diseño, la organización, estructura y métodos llevados a cabo para la realización del programa renta de activos con el objetivo de ofrecer un mantenimiento adecuado y evitar errores en este.

# 

# **OBJETIVOS**

## **General**

* Lo descrito a continuación tendrá la finalidad de poder entender en qué consiste el sistema o desarrollo realizado, el lenguaje utilizado en el siguiente porgama es C++ y el IDE CLion.

## **Especifico**

* Se especificarán los métodos utilizados y las librerías consumidas a discreción del desarrollador
* Cada método tendrá una breve descripción para poder ser modificado si se desea y a su vez mejorado.

# 

# **ALCANCES DEL SISTEMA**

La compilación del sistema es realizada en consola.

Se utilizó CLion.

El lenguaje de programación es C++

# 

# **ESPECIFICACIÓN TÉCNICA**

## REQUISITOS DEL HARDWARE

* Se requieren 4GB de RAM.
* Se requieren 1MB de almacenamiento disponible.

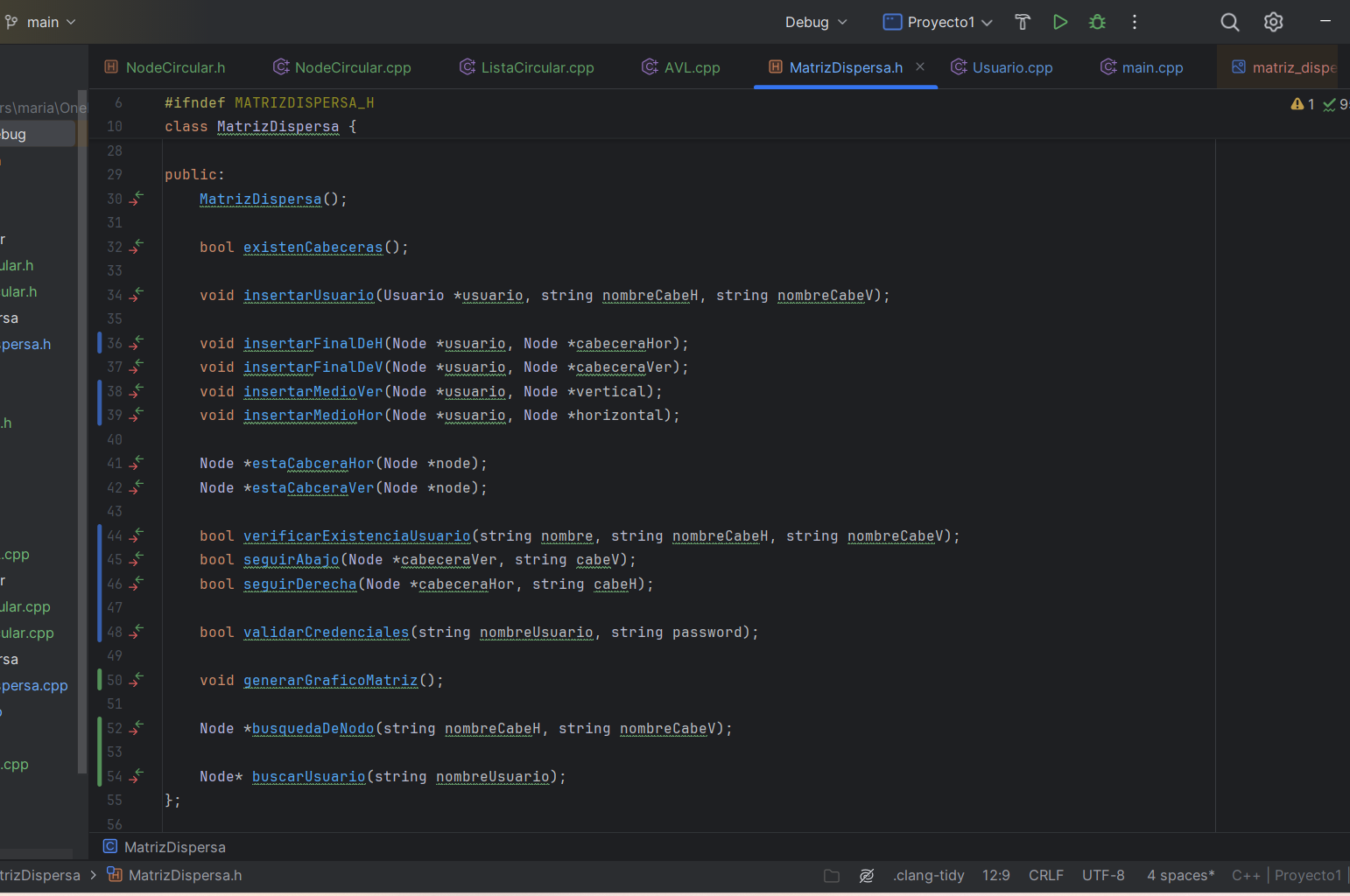
## REQUISITOS DEL SOFTWARE

* Se requiere Windows 10 o superior.
* Se requiere un IDE x64Bits

**Lógica del Programa**

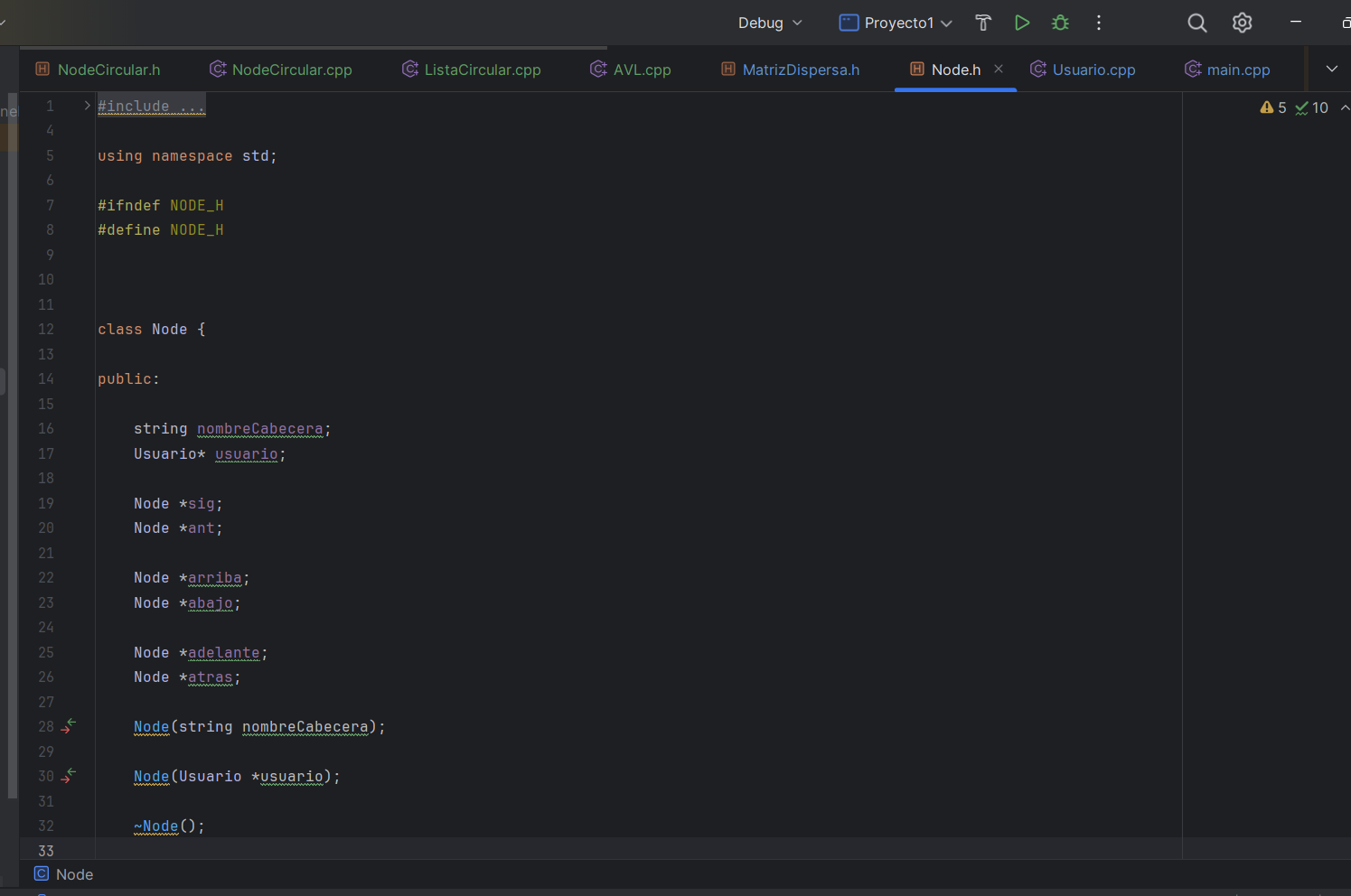
Se presentarán las principales estructuras y las funciones mas importantes.

* Matriz Dispersa:



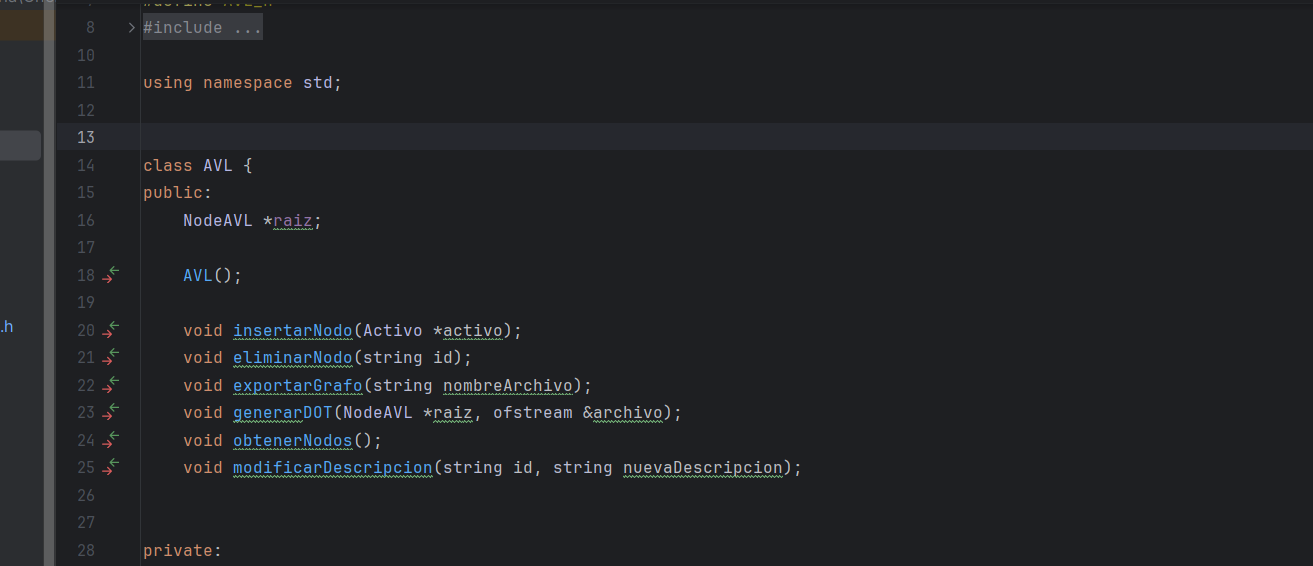
Aquí se presentan las funciones que pueden ser utilizadas fuera de matrizDispersa.cpp

* Nodo de la Matriz Dispersa:



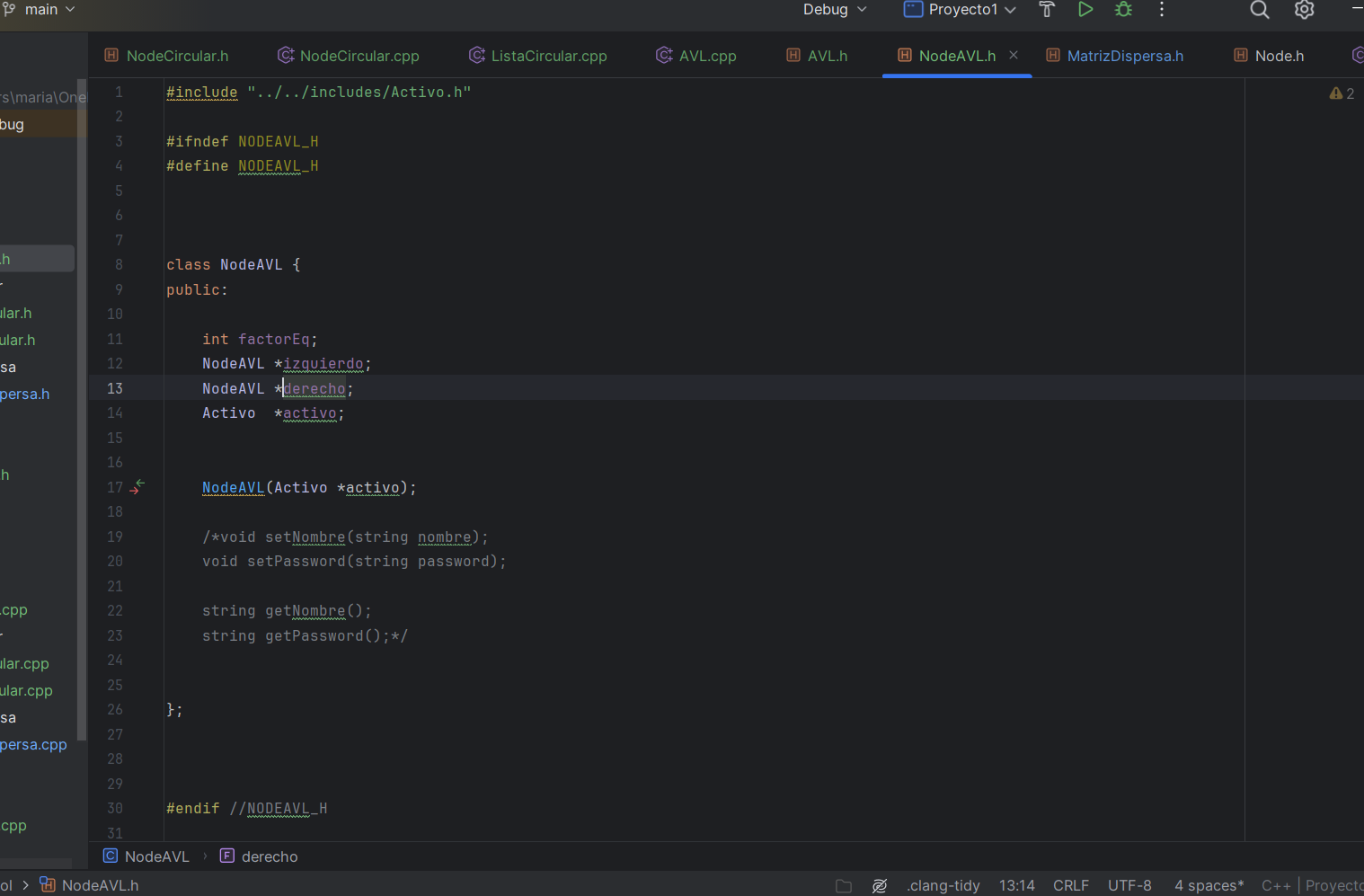
Tenemos 6 apuntadores ya que funciona como un cubo.

* AVL:



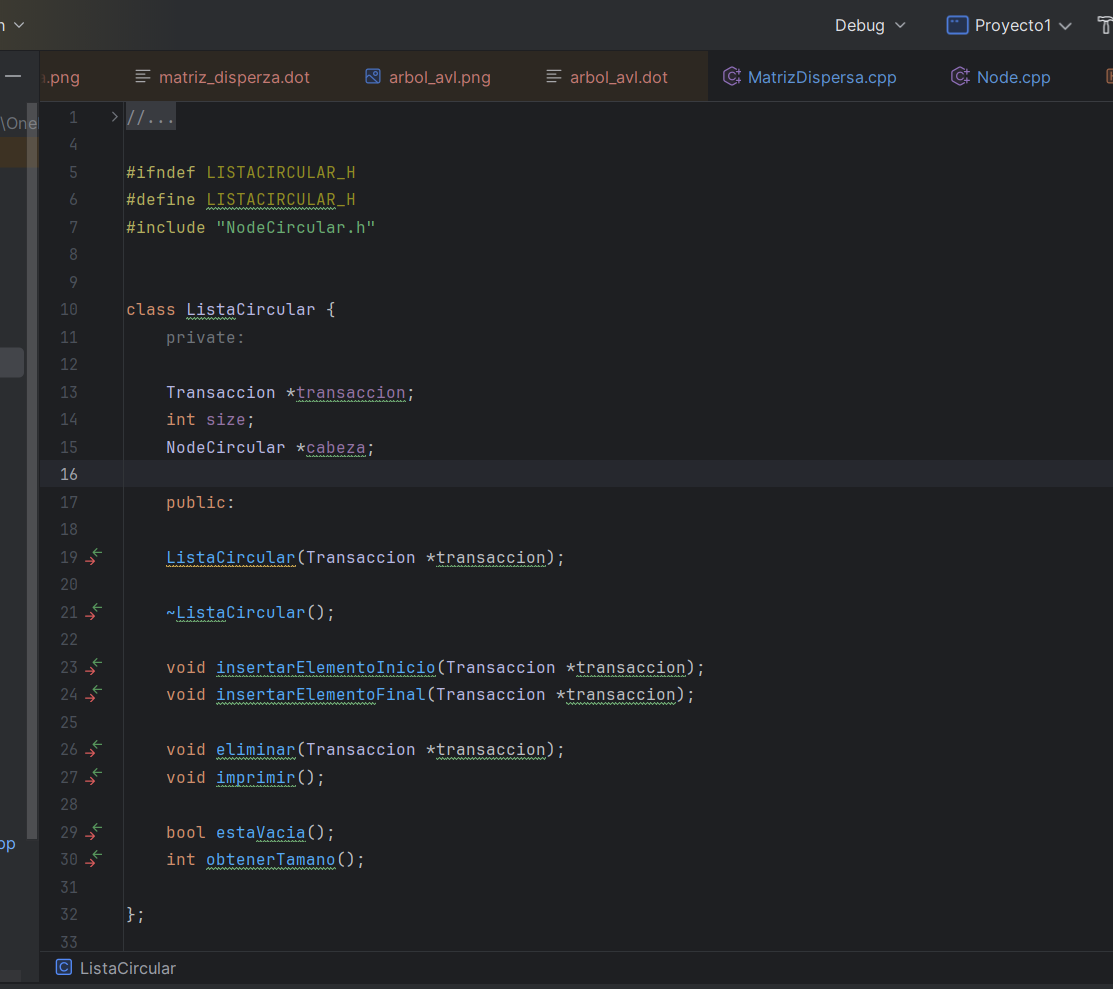
Se presentan las funciones que pueden ser utilizadas fuera de AVL.cpp

* NodeAVL:



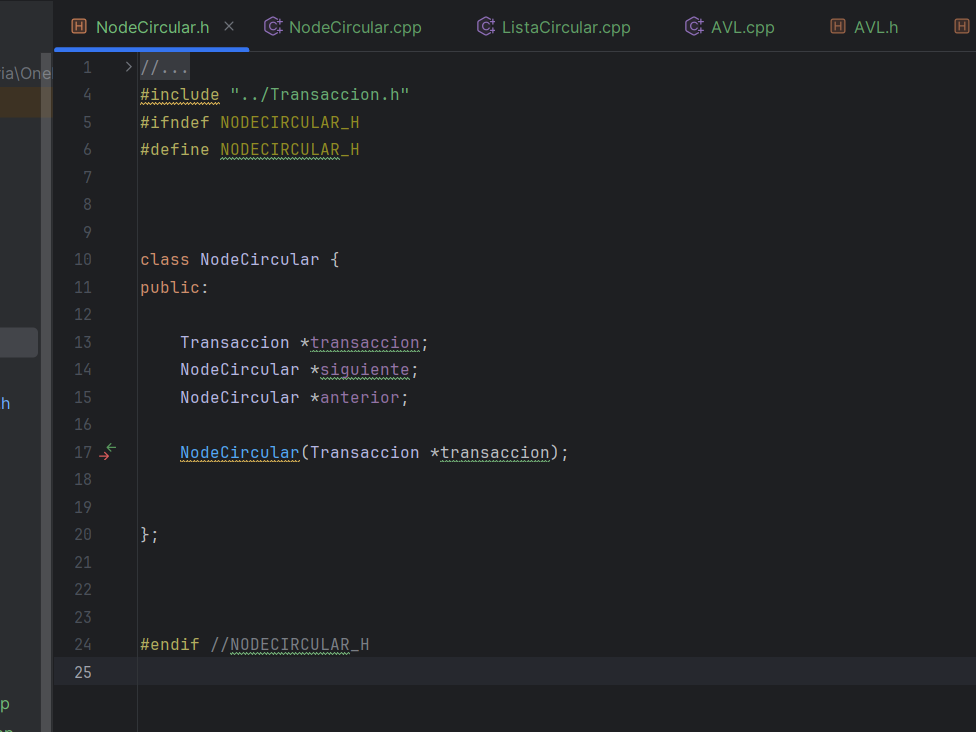
Se tiene dos apuntadores ya que cada nodo solo puede tener dos nodo hijos.

* ListaCircular:



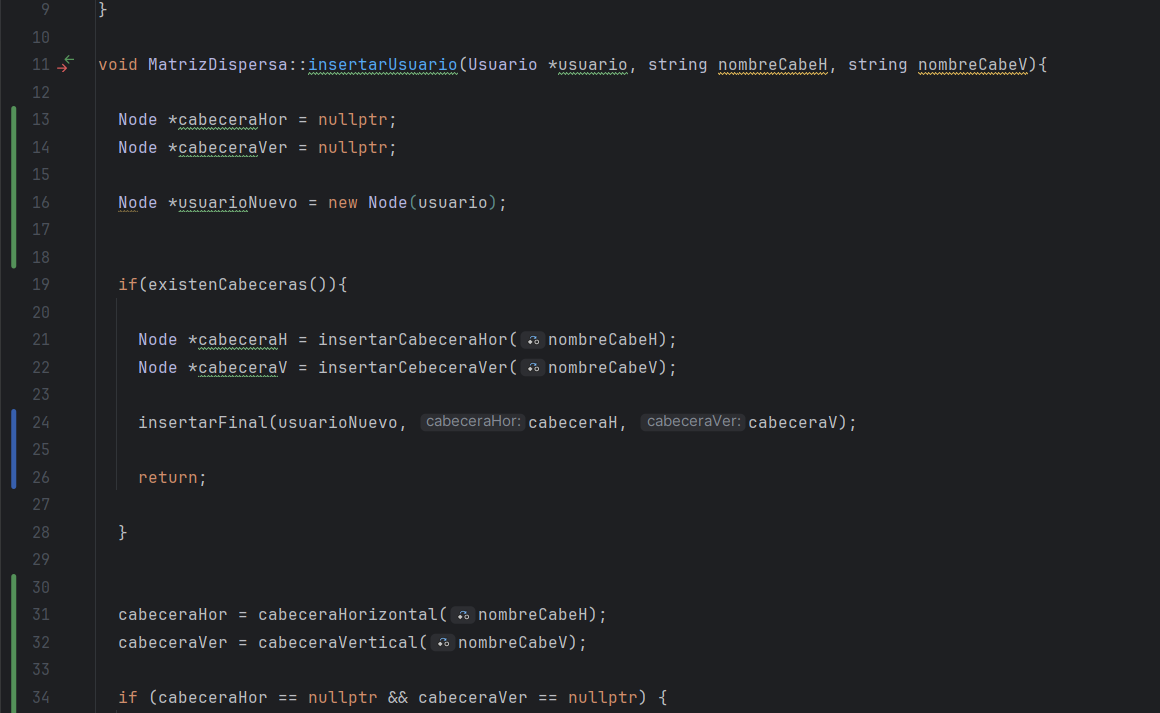
Funciones publicas de Lista Circular.

* NodeCircular:



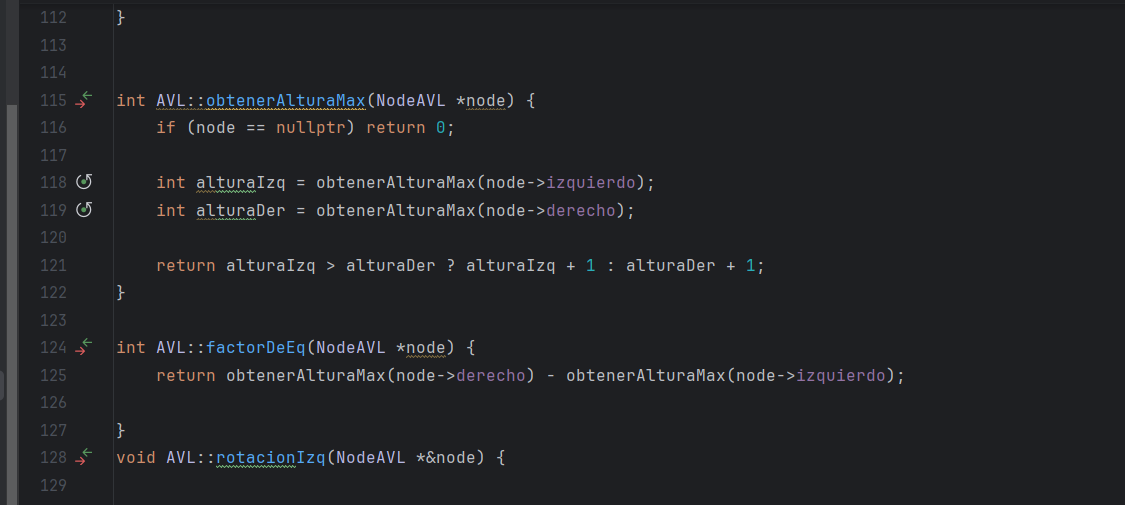
Se necesitan un anterior y un siguiente, para que sea doblemente enlazado.

* InsertarUsuario:



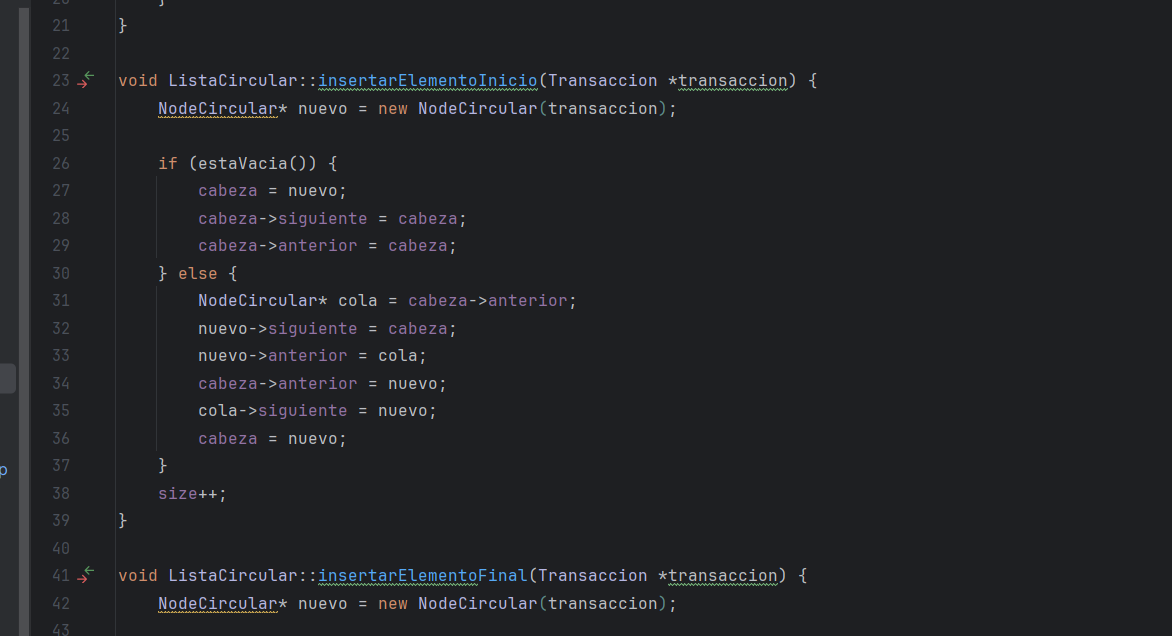
Parte fundamental para que se puede insertar el usuario a la matriz dispersa y además, se creen las cabeceras de este de manera correcta.

* FactorDeEqui:



Conocer el factor de equilibrio de los nodos es fundamental para saber si es un árbol binario de búsqueda equilibrado.

* Insertar Lista Circular:



La lista circular debe tener una cabeza y una cola para que esta no se enclicle,