

MANUAL TECNICO

El presente proyecto un sistema de renta de activos entre usuarios. Dentro del sistema, se manejarán rentas, usuarios, un historial de transacciones y reportes. Todo el sistema esta controlado mediante una aplicación en consola y trabajada con el lenguaje C++

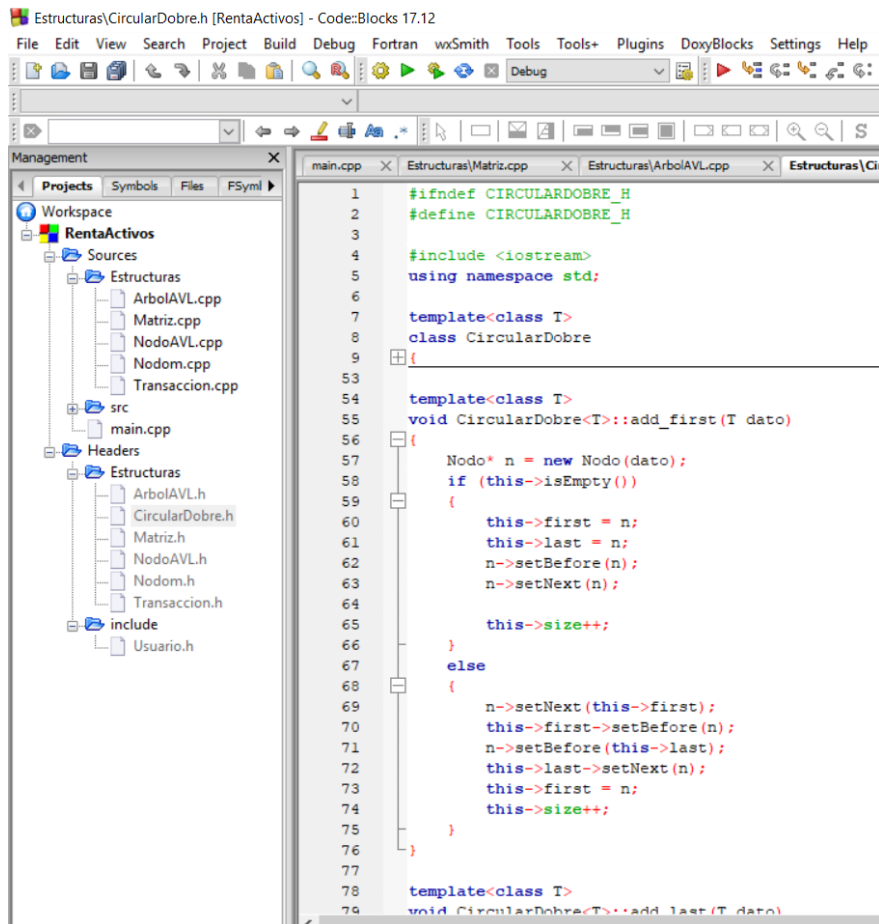
IDE utilizado: Code::Blocks

Versión: 17.12

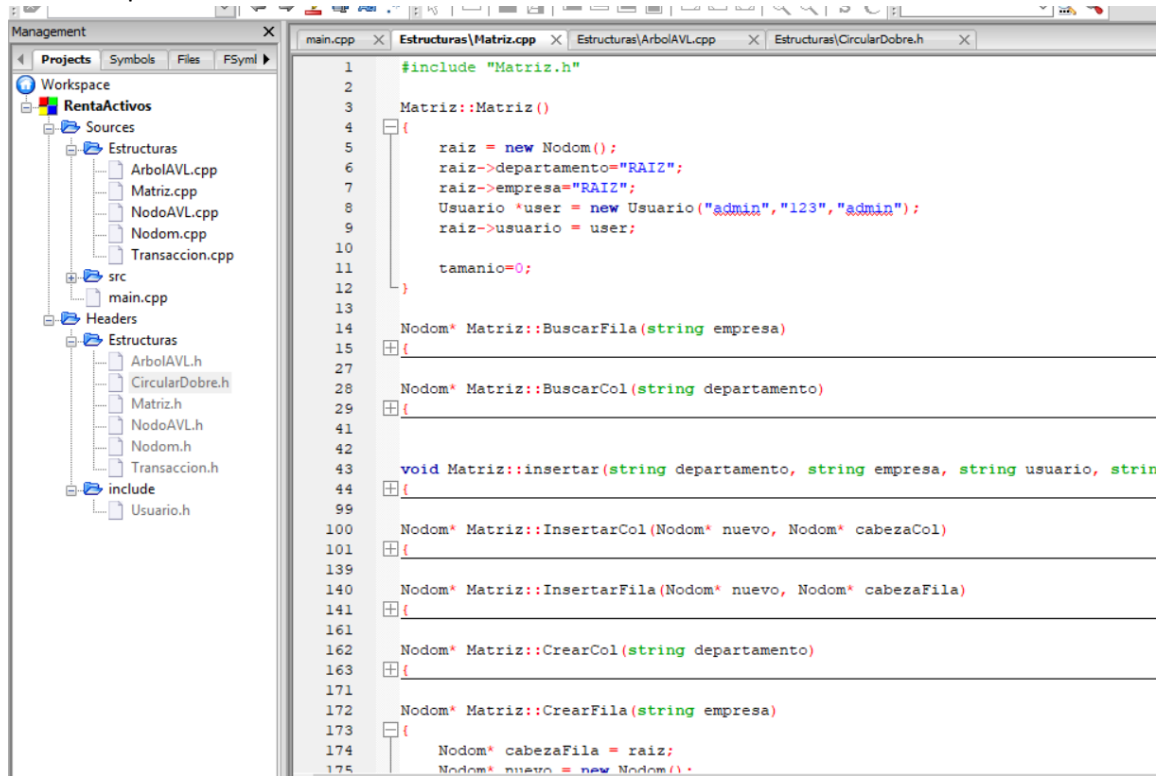
Sistema Operativo: Windows home 10

A continuación se explicaran las estructura de datos utilizadas en la aplicación.

1) Lista Circular Doblemente Enlazada implementa con Templates



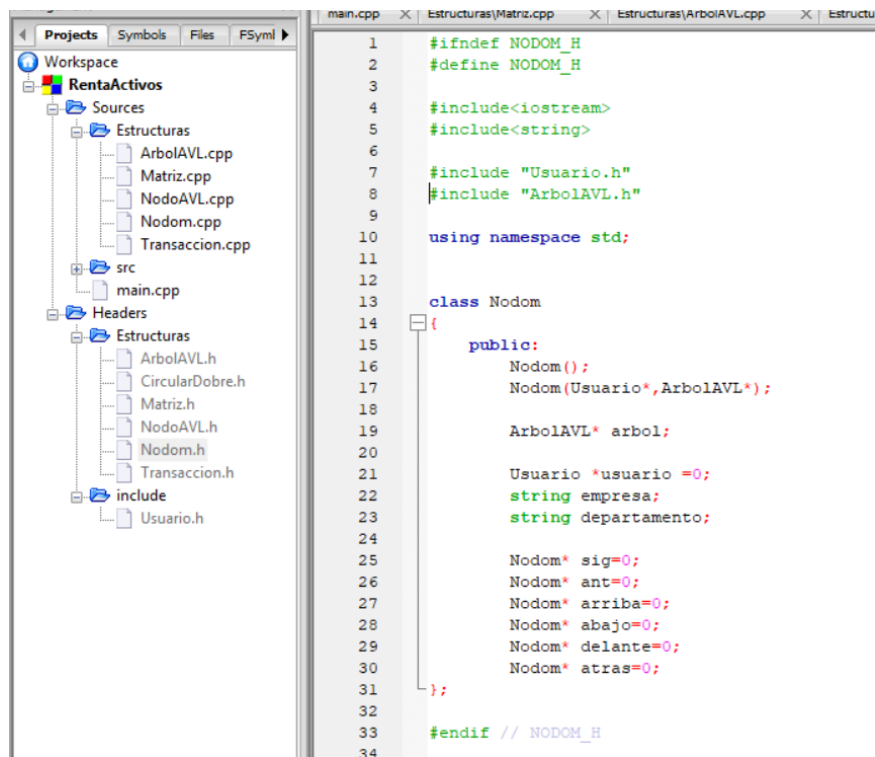
2) Matriz dispersa



The screenshot shows a C++ IDE with a project named 'RentaActivos'. The left sidebar displays the project structure with source files and headers. The main editor window shows the implementation of the 'Matriz' class in 'Estructuras\Matriz.cpp'. The code includes a constructor that initializes a root node, a user, and a size variable. It also defines several methods for searching and inserting elements into the matrix.

```
1 #include "Matriz.h"
2
3 Matriz::Matriz()
4 {
5     raiz = new Nodom();
6     raiz->departamento="RAIZ";
7     raiz->empresa="RAIZ";
8     Usuario *user = new Usuario("admin", "123", "admin");
9     raiz->usuario = user;
10
11     tamaño=0;
12 }
13
14 Nodom* Matriz::BuscarFila(string empresa)
15 {
16
17 }
18
19 Nodom* Matriz::BuscarCol(string departamento)
20 {
21
22 }
23
24 void Matriz::insertar(string departamento, string empresa, string usuario, string empresa)
25 {
26
27 }
28
29 Nodom* Matriz::InsertarCol(Nodom* nuevo, Nodom* cabezaCol)
30 {
31
32 }
33
34 Nodom* Matriz::InsertarFila(Nodom* nuevo, Nodom* cabezaFila)
35 {
36
37 }
38
39 Nodom* Matriz::CrearCol(string departamento)
40 {
41
42 }
43
44 Nodom* Matriz::CrearFila(string empresa)
45 {
46     Nodom* cabezaFila = raiz;
47     Nodom* nuevo = new Nodom();
```

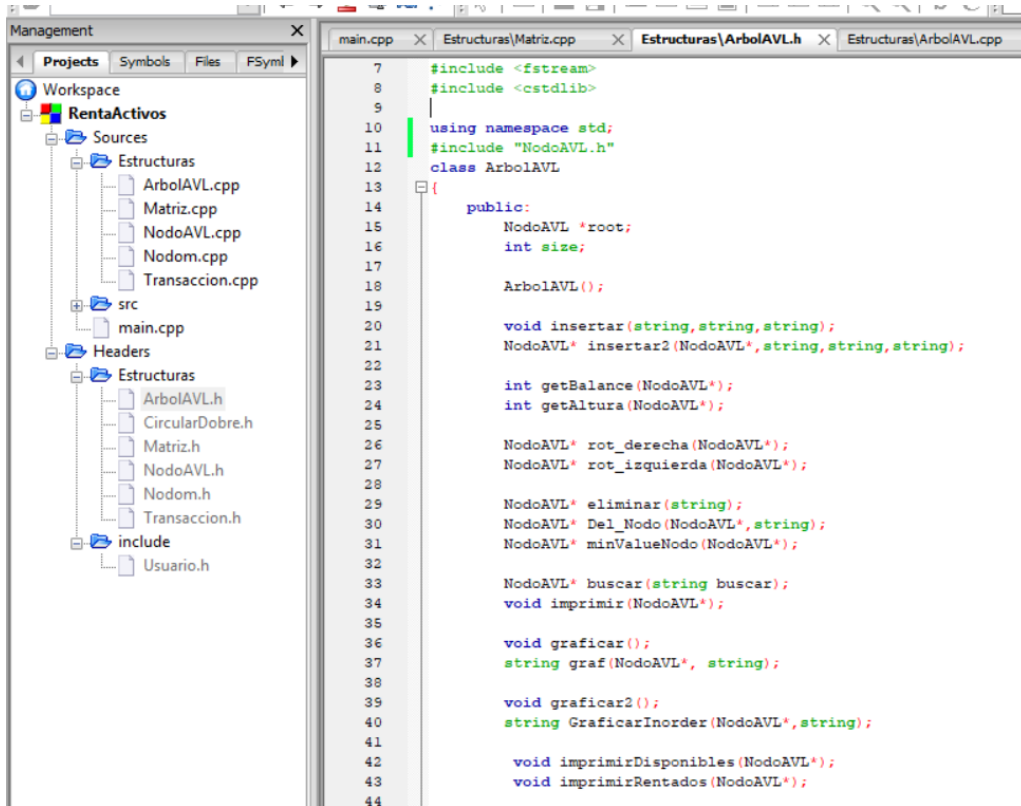
Siendo la estructura de sus nodos de esta forma



The screenshot shows the same IDE with the project structure. The main editor window shows the definition of the 'Nodom' structure in 'Estructuras\Nodom.h'. The code uses a preprocessor to define the structure and includes necessary headers. The structure contains pointers to other nodes and a pointer to a user.

```
1 #ifndef NODOM_H
2 #define NODOM_H
3
4 #include <iostream>
5 #include <string>
6
7 #include "Usuario.h"
8 #include "ArbolAVL.h"
9
10 using namespace std;
11
12 class Nodom
13 {
14 public:
15     Nodom();
16     Nodom(Usuario*, ArbolAVL*);
17
18     ArbolAVL* arbol;
19
20     Usuario *usuario =0;
21     string empresa;
22     string departamento;
23
24     Nodom* sig=0;
25     Nodom* ant=0;
26     Nodom* arriba=0;
27     Nodom* abajo=0;
28     Nodom* delante=0;
29     Nodom* atras=0;
30
31 };
32
33 #endif // NODOM_H
```

3) Arbol AVL



Con nodos AVL de estructura:

