

```

1  #include <iostream>
2  #include<conio.h>
3  #include<stdio.h>
4  #include<stdlib.h>
5  #include <string.h>
6  #include<fstream>
7  #include "lista.h"
8  #include "vino.h"
9  #include "infoMensual.h"
10 #include "funcionesCargarArchivos.h"
11 #include "rankingBodegas.h"
12 #include "RankingVarietales.h"
13 #include <iomanip>
14 using namespace std;
15
16 void lstBodegas(Lista vinos, Lista infoSeleccionados)
17 {
18     Lista lista = crearLista();
19     //Guarda los datos de los registros catalogo.txt primero en nodos(infoVinos)
20     //Y luego en struct Vino
21     int ultimo = ultimoAnio();
22     for(int j = 1; j <= infoSeleccionados->tamano; j++)
23     {
24         Nodo infoMen = nodoSeleccionado(infoSeleccionados,j);
25         infoMesual info = (infoMesual) infoMen->dato;
26         if(getsAnioSeleccion(info) == ultimo){
27             Vino vino1 = traerVino(vinos, getsVino1(info));
28             Vino vino2 =traerVino(vinos, getsVino2(info));
29             Vino vino3 =traerVino(vinos, getsVino3(info));
30             Vino vino4 =traerVino(vinos, getsVino4(info));
31             Vino vino5 =traerVino(vinos, getsVino5(info));
32             Vino vino6 =traerVino(vinos, getsVino6(info));
33
34             agregarBodega(lista,vino1);
35             agregarBodega(lista,vino2);
36             agregarBodega(lista,vino3);
37             agregarBodega(lista,vino4);
38             agregarBodega(lista,vino5);
39             agregarBodega(lista,vino6);
40         }
41     }
42
43     Lista ordenada=crearLista();
44
45     ordenarListaAuxBodega(lista,ordenada);
46
47     mostrarListaBodega(ordenada);
48
49 }
50
51 void mostrarListaBodega(Lista ordenada){
52     cout <<endl<<"Ranking de bodegas:"<<endl;
53     cout << setw( 60 ) << setfill( '-' ) << '\n' << setfill( ' ' );
54     cout << " | " << left << setw( 14 ) << "Posicion"<< " | " << right << setw( 1 ) << "
Bodega" << right << setw( 13 )<< " | "<< "Cantidad Seleccionado" << " | "<<endl;
55     cout << setw( 60 ) << setfill( '-' ) << '\n' << setfill( ' ' ) << '\n';
56     for(int b=1; b< ordenada->tamano+1 ; b++)
57     {
58         Nodo nodoRanking=nodoSeleccionado(ordenada,b);
59         Bodega bod=(Bodega)nodoRanking->dato;
60
61         int resultado=strlen(bod->nombreBodega);
62
63         cout <<"Nro en lista:"<<nodoRanking->nro<<" -Bodega " <<bod->nombreBodega<<
right<< setw(33-resultado)<<" -Cant seleccionada: " <<bod->cant<<endl;
64     }

```

```

65
66     cout<<"\n"<<endl;
67 }
68
69 Bodega buscarBodega(Lista lista, char nombre[50])
70 {
71     Nodo aux=lista->inicio;
72     int encontrado=0;
73     while(aux!=NULL && encontrado==0)
74     {
75         Bodega bodega=(Bodega) aux->dato;
76         if(strcmp(bodega->nombreBodega, nombre)==0)
77         {
78             encontrado=1;
79         }
80         else
81         {
82             aux=aux->siguiente;
83         }
84     }
85     if(encontrado==1)
86     {
87         return (Bodega) aux->dato;
88     }
89     else
90     {
91         return NULL;
92     }
93 }
94
95 void agregarBodega(Lista bodega, Vino vino){
96
97     Bodega bod = buscarBodega(bodega, getsNombreBodega(vino));
98     if(bod!=NULL)
99     {
100         bod->cant++;
101     }
102     else
103     {
104         bod = new BodegaStruct;
105         bod->cant=1;
106         strcpy(bod->nombreBodega, getsNombreBodega(vino));
107         agregarNodo(bodega, bod);
108     }
109
110
111 }
112
113 void ordenarListaAuxBodega(Lista lista, Lista ordenada){
114     Nodo actual;
115     Nodo siguiente;
116     actual=lista->inicio;
117
118     Nodo auxNodo;
119     int nro;
120
121     while (actual!=NULL )
122     {
123         nro=actual->nro;
124         siguiente=actual->siguiente;
125         Bodega auxMaximo=(Bodega) actual->dato;
126         while( siguiente !=NULL )
127         {
128             Bodega rankig13=(Bodega) siguiente->dato;
129             if(auxMaximo->cant<rankig13->cant)
130             {

```

```
131         auxMaximo=rankig13;
132         nro=siguiente->nro;
133     }
134     siguiente=siguiente->siguiente;
135
136 }
137
138 Bodega nuevo=new BodegaStruct;
139 auxNodo=nodoSeleccionado(lista,nro);
140 Bodega bodegaAnt=(Bodega)auxNodo->dato;
141 nuevo->cant=bodegaAnt->cant;
142 strcpy(nuevo->nombreBodega,bodegaAnt->nombreBodega);
143
144 agregarNodo(ordenada,nuevo);
145 eliminarNodo(lista,nro);
146
147 actual=lista->inicio;
148
149
150 }
151
152
153
154
155 }
```