

# MANUAL TECNICO

## Indice

- Headers
  - Biblioteca.h
  - Mainwindow.h
- Sources
  - Biblioteca.cpp
  - Mainwindow.cpp
- Extras
  - Practica1\_201403877.pro

## Headers

- Biblioteca.h
  - Class nodo2: clase que representa una cancion y es un elemento de:
    - Lista simple (Lista de canciones de biblioteca y lista de top's)
    - Lista doble(Lista de reproduccion)
    - Pila(Lista de reproduccion)
    - Cola(Lista de reproduccion)

Para cada estructura solo se utilizan los atributos que se necesitan por lo que la clase cuenta con constructores de sobrecarga para inicializarlos.

- Class ListaCancion: Guarda un conjunto de canciones y provee metodos para administrarlos, representando una lista simple. Los nodos que utiliza son de la clase "nodo2".
- Class nodo1: representa un album y es un elemento de:
  - Lista doble(lista de albums de biblioteca)

Solo posee un constructor y contiene una lista simple de canciones de la clase "ListaCancion" la cual reserva memoria al momento de crear el nodo.

- Class ListaAlbum: Guarda un conjunto de albums en una estructura de tipo "doble circular" en la cual se accede al primer elemento a traves de "primero doble". Provee metodos para administrar la estructura y usa nodos de la clase "nodo1".
- Class nodo: representa un artista y es un elemento de:
  - Lista doble circular(Lista de artistas en la biblioteca)

Contiene una lista de albums de la clase "ListaAlbum" y se puede acceder a ella a traves del atributo "albums".

\* a traves de este nodo se ingresar a la lista de albums y lista de canciones.

- Class ListaArtistas: Contiene una lista de artistas utilizando la estructura "lista doble circular" accede a ella a traves del atributo "primeroDcircular". Provee metodos para manipular la estructura y utiliza nodos de la clase "nodo".
  - Metodo "split\_string\_archivo": Metodo que recibe como parametro la cadena completa de informacion de tipo string y una lista de artistas para añadir en ella los datos de tipo "ListaArtistas".  
Recorre el string, genera sub-string que representan una cancion y los envia uno por uno a la funcion "ingresar\_a\_biblioteca".
  - Funcion "ingresar\_a\_biblioteca": Recibe como parametros una cadena de tipo string que representa una cancion, y una lista de artistas en la cual se añadira los datos.  
Se recorre la cadena para obtener sub-cadenas que representan los atributos de la cancion. Se busca en la lista de artistas el artista, el album y la cancion, si existe uno de los atributos solo se agregan los demas.
  - Class ListaTops: utiliza nodos de la clase "nodo2" y representa una "lista simple ordenada". Contiene metodos para agregar y graficar. Puede contener top de artistas, de albums o de canciones.
  - Class Doble circular, Pila y Cola: Se utilizan para forman una lista de reproduccion y proveen metodos para administrar sus elementos, utiliza nodos del tipo "nodo2". Los nodos no hacen referencia a la cancion en la biblioteca solo poseen una copia de su informacion.
- MainWindow.h
    - Void on\_pushButton\_clicked() (boton: Cargar musica): Carga el archivo que contiene la informacion de canciones, lo abre, lo lee y concatena todo en un string para enviarlo a la funcion "split\_string\_archivo" junto con la lista de artistas.
    - Void on\_pushButton\_2\_clicked() (boton: vizualizar biblioteca): Llama al metodo de graficar de lista de artistas la cual Grafica el contenido completo de la biblioteca.
    - Void on\_pushButton\_5\_clicked() (boton: top artistas), Void on\_pushButton\_4\_clicked() (boton: top albums), Void on\_pushButton\_5\_clicked() (boton: top canciones): Crear un objeto del tipo "ListaTops" se recorre la biblioteca por medio de la ListaArtistas, se agregan los elementos correspondientes y se grafica la estructura.
    - Void on\_pushButton\_6\_clicked() (boton: establecer): obtiene el valor seleccionado de un conjunto de radio button que representan cada una de las posibles listas de reproduccion (lista doble circular, pila, cola) y reserva memoria para ella.
    - Void on\_pushButton\_7\_clicked() (boton: agregar): Obtien el string de la caja de texto, separa sus valores de sub cadenas, se busca la cancion en la biblioteca y si existe se agrega a la estructura establecida y se llama al metodo graficar de ListaTop.
    - Void on\_pushButton\_9\_clicked() (boton: reproducir): reproduce la cancion que del nodo "actual" (variable global).

- Void on\_pushButton\_8\_clicked() (boton: anterior): solo esta habilitada si la estructura seleccionada es “doble circular”. Asigna a nodo “actual” el anterior del nodo, y reproduce la cancion.
- Void on\_pushButton\_11\_clicked() (boton: siguiente): Para la pila y la cola elimina el dato de la estructura y asigna la cola o la cabeza al nodo “actual”, grafica la estructura y reproduce la cancion. Para la lista doble circular solo asigna al nodo “actual” el elemento siguiente.
- Void on\_pushButton\_12\_clicked() (boton: eliminar): obtiene el string de la caja de texto, separa la cadena en sub-string para obtener los atributos. Busca el artista, el album y la cancion, si existen realiza la eliminacion de la cancion, si el album se queda sin canciones se elimina el album y si se queda sin albums el artista se elimina.
- Void on\_pushButton\_10\_clicked() (boton: detener): detiene la reproduccion de la cancion actual.
- Void on\_verticalSlider\_valueChanged: establece el volumen de reproduccion.

## Sources

- Biblioteca.cpp
  - ListaArtistas
    - Nodo\* buscar: Busca un elemento en la lista doble circular (lista de artistas), si encuentra el dato retorna el nodo sino retorna 0 (null).
    - Nodo\* agregar: Si el artista no existe agrega un nuevo nodo a la lista de artistas y retorno el nodo creado, sino, retorna 0 (null).
    - Void graficar: Recorre la biblioteca, escribe en un archivo de texto la informacion (nombre y enlaces). Por cada nodo de artista llama a la funcion “graficarAux” y al finalizar genera la imagen.
    - Void graficarAux: Recibe como parametro un nodo que representa un artista, por cada album llama a la funcion graficar de la clase ListaCancion. Escribe en un archivo la informacion(nombre y enlaces) del nodo en el que se encuentra.
    - Void eliminar: recibe como parametro el elemento que se debe eliminar. Verifica las referencias a otros nodos y elimina el nodo.
  - ListaAlbum
    - Nodo\* buscar: Busca un elemento en la lista doble (lista de albums), si encuentra el dato retorna el nodo sino retorna 0 (null).
    - Nodo\* agregar: Si el album no existe agrega un nuevo nodo a la lista de albums y retorno el nodo creado, sino, retorna 0 (null).
    - Void graficar: Recorre la lista de albums, escribe en un archivo de texto la informacion (nombre y enlaces). Por cada nodo de artista llama a la funcion “graficarAux”.

- Float calcularValoracion: Recibe como parametro el primer elemento de una lista doble, la recorre y calcula el promedio de la valoracion de todos los elementos y retorno el valor total.
  - Void eliminar: recibe como parametro el elemento que se debe eliminar. Verifica las referencias a otros nodos y elimina el nodo.
- ListaCancion:
  - Nodo\* buscar: Busca un elemento en la lista simple (lista de canciones), si encuentra el dato retorna el nodo sino retorna 0 (null).
  - Nodo\* agregar: Si la cancion no existe agrega un nuevo nodo a la lista de canciones y retorno el nodo creado, sino, retorna 0 (null).
  - Void graficar: Recorre la lista de canciones, escribe en un archivo de texto la informacion (nombre y enlaces).
  - Float calcularValoracion: Recibe como parametro el primer elemento de una lista simple, la recorre y calcula el promedio de la valoracion de todos los elementos y retorno el valor total.
  - Void eliminar: Recibe como parametro el nombre de la cancion, busca el elemento y si se encuentra, verifica los enlaces y elimina el nodo de la lista.
- ListaTops
  - Void agregar: recibe como parametro el nombre del elemento, su valoracion y lo añade a la lista simple en forma ordenada.
  - Void graficar: Recorre la estructura, escribe en un archivo de texto la informacion (nombre y enlaces) y grafica la estructura.
- Mainwindow.cpp
  - Atributos
    - nuevo: Representa la biblioteca. Es una Lista doble circular(ListaArtistas), se reserva memoria al ejecutar la aplicacion.
    - tops: Es una lista simple de la clase "ListaTops" y se reserva memoria cuando el usuario hace la peticion de generar algun reporte de top.
    - actual: representa la posicion de la cancion actual.

## Extras

- practica1\_201403877.pro
  - "QT += core gui multimedia": Necesario para la reproduccion de audio.
  - "QMAKE\_CXXFLAGS += -std=gnu++11": para que funcionen las conexiones de la barra de progreso.