

Universidad Rafael Landívar

Facultad de Ingeniería

Programación Avanzada

Sección 06

Proyecto 1

Party Mix

Emilio Barillas Contreras

Carné 1150620

INTRODUCCIÓN

Para el proyecto de programación avanzada se tuvo como objetivo que el alumno implementara el uso de estructuras dinámicas, así como métodos de ordenamiento. Las estructuras dinámicas permiten crear arreglos de tamaño indefinido para que el usuario sea capaz de almacenar más datos. Para la realización de estas estructuras dinámicas se usaron punteros, ya que estos almacenan la dirección de memoria permitiendo así ampliar las listas o pilas indefinidamente. Se programaron algoritmos de agregar elementos, eliminar elementos, verificar si las estructuras estaban o no vacías. Todo esto implementando la programación orientada a objetos o POO.

Para la solución de este proyecto se implementaron pilas, con la característica de que el último elemento en ingresar es el primero en salir y colas, con la característica de que el primer elemento en entrar es el primer en salir.

ANÁLISIS

ENTRADAS

- Archivo CSV con datos de canción y artista para almacenar en la pila
- Nombre y artista para almacenar en la cola

SALIDAS

- Pila con canciones y artistas
- Cola con canciones y artistas
- Canción que se reproduce actualmente

PROCESOS

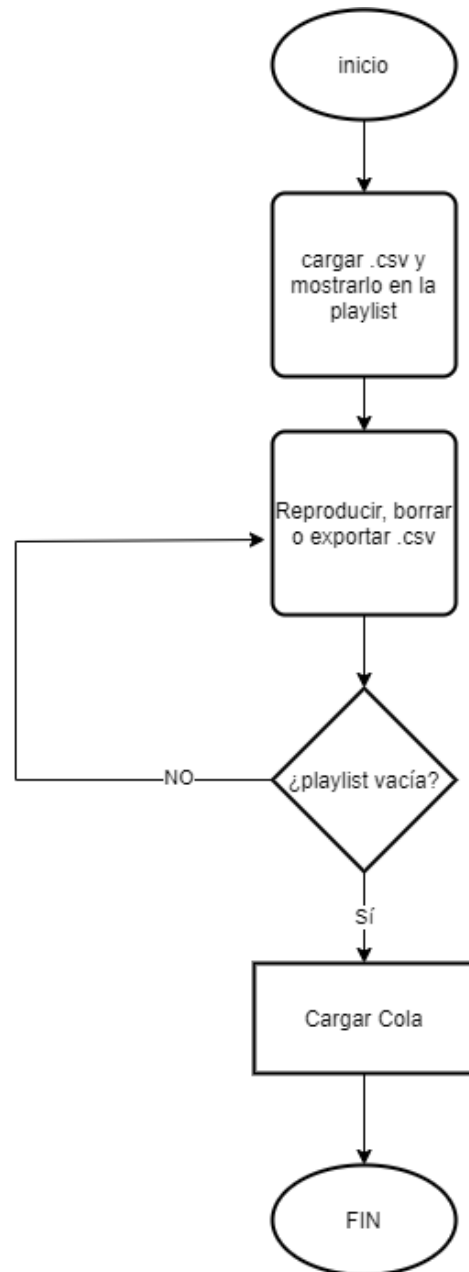
- Leer el archivo CSV
- Cargar la información a la pila
- Reproducir canciones siguiendo el orden lógico de pila
- Si la pila se vacía usar datos de la cola
- Reproducir o borrar pila
- Exportar pila a CSV
- Llenar la cola con información del usuario
- Agregar o eliminar elementos de la cola
- Sincronizar datos de la cola a la pila
- Reproducir anterior o siguiente en la cola

RESTRICCIONES

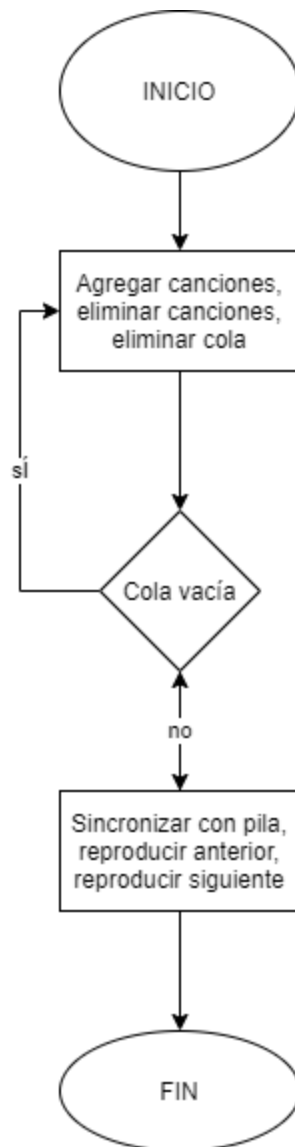
- Para canciones las cuales no se conozca el nombre o artista se clasificará con “DESCONOCIDO”
- La cola debe tener canciones para sincronizarse con la playlist
- Debe de haber canciones para poder reproducirse

DISEÑO

PILA – PLAYLIST



COLA – COLA DE REPRODUCCIÓN



CONCLUSIONES

- El uso de estructuras dinámicas facilita la creación de bases de datos para implementaciones de la vida diaria, donde no sabemos el tamaño de datos que vamos a procesar.
- Mediante el uso de pilas y colas es posible crear un sistema que permita que canciones se reproduzcan sin interrupción.
- Los métodos de ordenamiento son útiles para procesar grandes cantidades de datos.
- La programación orientada a objetos facilita la comprensión de problemas reales y su solución por medio de clases, métodos y funciones.

RECOMENDACIONES

- Utilizar la programación orientada a objetos para crear funciones de pilas y colas, de ese modo se agiliza y optimiza el código.
- Usar sobrecarga para funciones con entrada de datos similares.
- Trabajar con System string ya que es el utilizado por Windows form.
- Establecer métodos y funciones para reutilizar código.

REFERENCIAS

`#include <fstream>`: para poder leer los archivos csv y definir delimitador para extraer la información.

`#include <string>`: habilita las variables `str::string` es decir estándar string que se utilizan en c++.

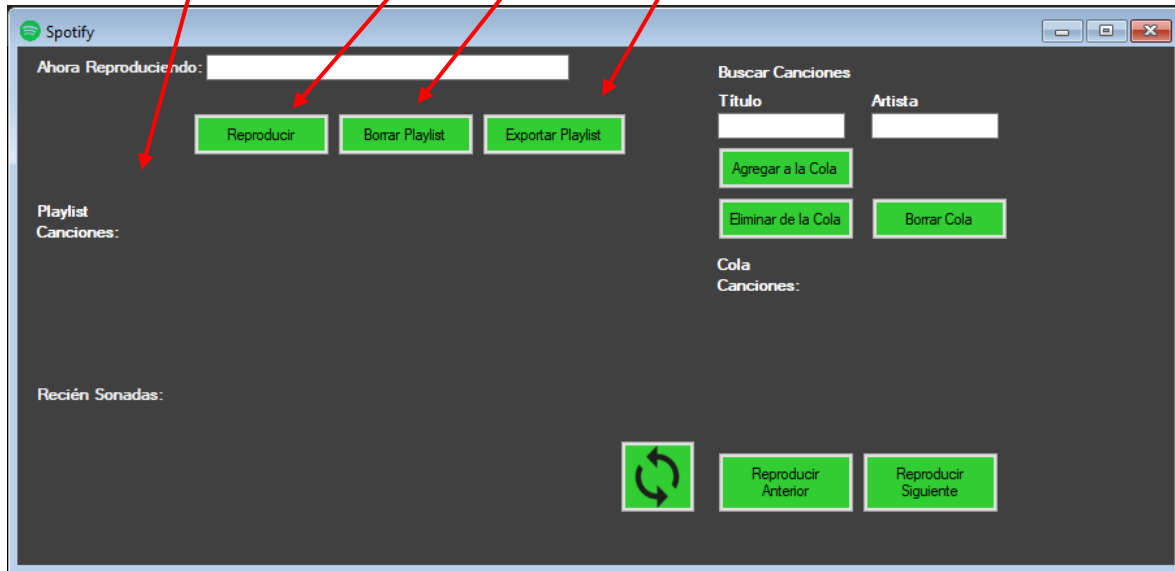
`#include <iostream>`: útil para implementaciones de entrada y salida

ANEXOS

MANUAL DE USUARIO

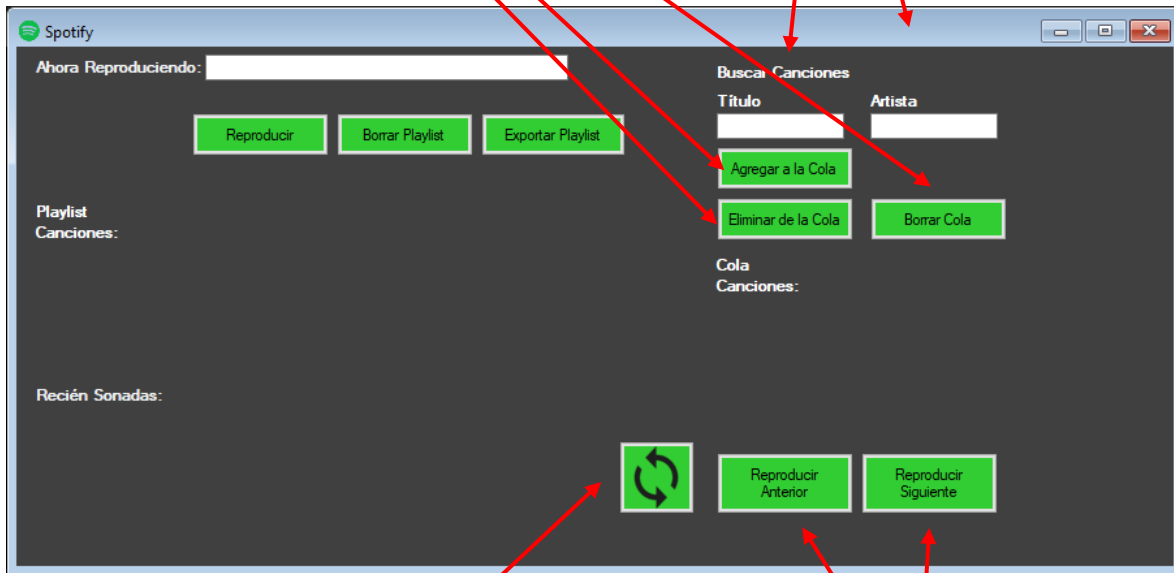
Al ejecutar el programa la playlist se carga con los datos del archivo CSV.

Con el botón reproducir elige la última canción de la pila y la saca para que se reproduzca. El botón borrar elimina la playlist y el botón exportar escribe un archivo CSV con los datos de la playlist actual



Con el botón agregar a la cola se agrega a esta estructura, o se puede eliminar el primer elemento ingresado. Con el botón de borrar cola se elimina toda la estructura.

En los cuadros de texto se pueden ingresar nuevas canciones a la cola con título y artista



El botón de sincronizar agrega la cola a la pila para tener más canciones.

El botón de reproducir anterior reproduce la canción que ha sonado y siguiente la próxima en la cola