

Universidad Rafael Landívar
Facultad de Ingeniería
Campus Central
Física I

Proyecto de Aplicación 1

Ángel Daniel Ávila Arriaza
Carnet: 1180622

Programación Avanzada Segundo Ciclo 2023

Introducción

El programa es una implementación básica de un reproductor de música, representado por la clase `PlayList`. Este reproductor interactúa con CDs y permite al usuario agregar canciones desde un CD a una cola de reproducción, visualizar la cola de reproducción de diversas maneras, reproducir la siguiente canción en la cola y ordenar la cola según diferentes criterios. La aplicación utiliza las clases `CD` y `Song` para modelar discos y canciones, respectivamente. Además, incorpora la funcionalidad de pausar y reanudar la reproducción, proporcionando una experiencia simple pero funcional para gestionar y disfrutar de la música almacenada en CDs.

1) ¿Qué acciones debe poder hacer su programa? Enumérelas.

1. Cargar respaldo

- A. Leer cada uno de los archivos y procesarlo correctamente teniendo en consideración que cada línea debe de estar compuesta por los 3 elementos en el archivo de texto (Nombre de la canción, Artista y la duración de la canción). También debe tomar en cuenta que el archivo debe ser uno de texto porque si no será considerado como erróneo.
- B. Guardar información de los archivos leídos correctamente en una lista de CD's donde cada Cada CD debe guardar su propia lista de canciones con la información leída.
- C. Si el archivo tiene algún error no se puede cargar directamente al programa, pero el programa debe de mostrar un mensaje indicando el error en la consola con el nombre del archivo erróneo como, por ejemplo:
 - I. Archivo vacío
 - II. Formato de línea incorrecto
 - III. Canción duplicada
- D. Si el usuario selecciona nuevamente esta opción nuevamente, el programa tiene que limpiar la lista de CD's, la cola de reproducción y la cola de reproducción actual.

2. Reproductor de canciones

- A. El usuario utiliza esta función para poder reproducir cualquier canción en cualquiera de los CD's disponibles que el programa leyó al momento de cargar respaldo y cargando las canciones a la cola de la reproducción. El cual el reproductor de música debe de tener 5 funciones las cuales se detallarán más adelante.
- B. Agregar canción
Esta función mostrará las listas de CD disponibles para que el usuario escoja la que más le guste. Al momento de elegir un CD el programa tiene que mostrar las canciones dentro del CD seleccionado y de allí escoger la que más le guste y al seleccionar la canción, esta será agregada a la cola de reproducción.
- C. Ver Cola de reproducción
Con esta opción se le debe de consultar al usuario si desea ver la cola de reproducción actual o si desea que se imprima de forma ordenada ya sea por el nombre del artista, nombre de la canción o su duración.
- D. Reproducción Actual
Esta opción indica si hay alguna canción en la cola que se está reproduciendo, si es que no hubiera ninguna el programa debe de decir que la reproducción está en pausa. Sino fuera así tendría que mostrar su nombre, artista, duración y nombre del CD que la contiene. El programa

también tiene que almacenar en todo momento la canción que este reproduciendo hasta que sea activada la opción de reproducir siguiente.

E. Reproducir siguiente

La canción que se encuentra al inicio de la cola de reproducción se inserta nuevamente al final de la cola y también se debe mostrar la información de la nueva canción que en reproducción la cual es su nombre, Artista, Duración y nombre del CD.

F. Ordenar

En esta función como su nombre lo indica se puede ordenar la cola de forma permanente, se puede ordenar por el nombre del artista, por el orden del nombre de la canción o ya sea por la duración y si el usuario quisiera agregar una nueva canción dependiendo si lo quiere hacer por la función agregar canción o reproducir siguiente siempre debe de ir al final de la cola.

2) ¿Con qué datos va a trabajar? ¿Qué información debe pedir al usuario?, defina sus datos de entrada y el tipo de dato que utilizará para los datos principales.

1. Datos de entrada

String (La ruta de la carpeta con los respaldos cuando el programa lo solicite)

Seleccionar cualquiera de los CD's que estén disponibles

2. Datos de las funciones

a) Agregar Canciones:

Utilizará el valor void (No devuelve ningún valor), Clase PlayList y la cola donde se guardarán las canciones tomándolas desde la clase CD.

b) Ver cola de reproducción:

Utilizará el valor void (No devuelve ningún valor), Utiliza cola con la función Agregar Canciones.

c) Reproducción Actual:

Utilizará el valor void (No devuelve ningún valor), Utiliza la cola formada en cola de reproducción para verificar que la cola tenga alguna canción, además debe mostrar el nombre usando el tipo de dato string, el nombre del artista usando el tipo string, La duración de la canción convertido en double en segundos y Nombre del CD con el tipo de dato string.

d) Reproducir siguiente:

Utilizará el valor void (No devuelve ningún valor), Utilizara la cola de reproducción para comprobar si está vacío o si tiene las canciones agregadas y tomara la canción que se encuentra al inicio para luego mandarla al final de la cola y además debe mostrar el nombre usando el tipo de dato string, el nombre del artista usando el tipo string, La duración de la canción convertido en double en segundos y Nombre del CD con el tipo de dato string.

e) Ordenar:

Utilizará el valor void (No devuelve ningún valor), int se utilizará para almacenar la opción para ordenar ingresada por el usuario.

3) ¿Qué estructuras de datos trabajará para almacenar la información?

1) Cargar respaldos

- I. Cola

2) Reproductor de Canciones

A. Agregar Canción

- I. Se dividirán en dos funciones la primera es una función para comprobar si la cola de reproducción esta llena o no. La segunda función se utilizará para mostrar la lista de canciones disponibles para luego agregarlo a una cola.

B. Ver cola de reproducción

- I. En esta función se utilizará con una cola y la estructura switch para escoger el orden de la cola de reproducción actual si quiere que se quede así, si quiere ordenarlo por el artista, por el nombre de la canción o si debería ordenarse por su duración.

C. Reproducción actual

- I. En esta función se utilizará la estructura if para ver si la cola de reproducción esta siendo pausada, sino fuera así tendría que mostrar la información de la canción.

D. Reproducir siguiente

- I. Esta función también usara una función if para ver si la cola de reproducción esta llena, si esta llena mostrara la información acerca de la canción y sino indicara si la cola de reproducción esta vacía.

E. Ordenar

- I. El utilizara dos estructuras, la primera que es un switch dentro del Main preguntando si quieren ordenar la canción por su artista, por la canción o la duración de la canción por medio de ascendente o descendente. La segunda estructura será dentro de las clases para poder hacer su debido orden con un if.

4) ¿Qué condiciones o restricciones debe tomar en cuenta? ¿Qué cálculos debe hacer?

1. Archivo vacío

Leer si el archivo de texto si tiene algo escrito y si no indicara que esta vacío.

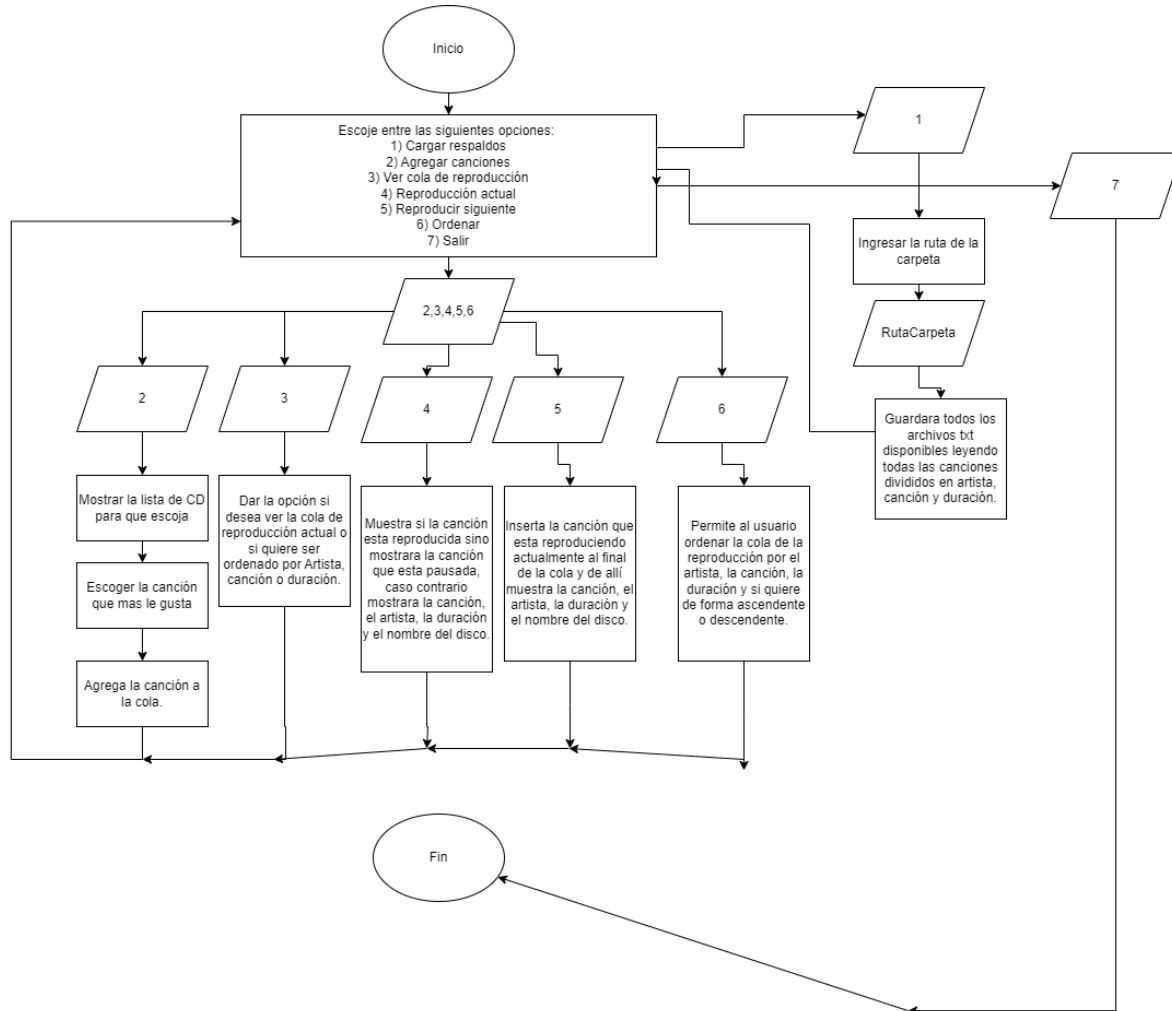
2. Formato de línea incorrecto

Verificar que estén los tres tipos de datos (Nombre de la canción, Autor y duración de la canción)

3. Archivos duplicados

Verificar si el archivo de texto ya ha sido guardado anteriormente y si es así entonces indicara que el archivo ya ha sido guardado.

5) Algoritmo para mostrar las funciones principales que debe realizar el programa.



Link del Github: <https://github.com/AngelAvila2308RS/ProgramacionAvanzada/tree/main>

Conclusión

El programa presenta una implementación básica pero funcional de un reproductor de música mediante la clase **PlayList**. Permite al usuario gestionar canciones desde CDs, agregarlas a una cola de reproducción, visualizar y ordenar la cola de diversas maneras, así como reproducir las canciones en orden. La introducción de las clases **CD** y **Song** proporciona una estructura clara y modular para modelar discos y canciones.

Sin embargo, durante la corrección, se identificaron y corrigieron algunos errores, particularmente relacionados con la ordenación de la cola de reproducción. La implementación final utiliza un enfoque que implica el uso de una cola auxiliar y un vector para ordenar los elementos antes de volver a llenar la cola original. Además, se solucionaron errores de tipos en la función **ordenarColaReproduccion**.

En resumen, el programa ahora debería funcionar correctamente, brindando una experiencia básica pero completa para gestionar y disfrutar de la música almacenada en CDs a través de una cola de reproducción.