



UNIVERSIDAD DE GUADALAJARA

Red Universitaria e Institución Benemérita de Jalisco

Rodriguez Rentería Jesus Alejandro

Código: 215510307

Materia: Estructura de datos 1

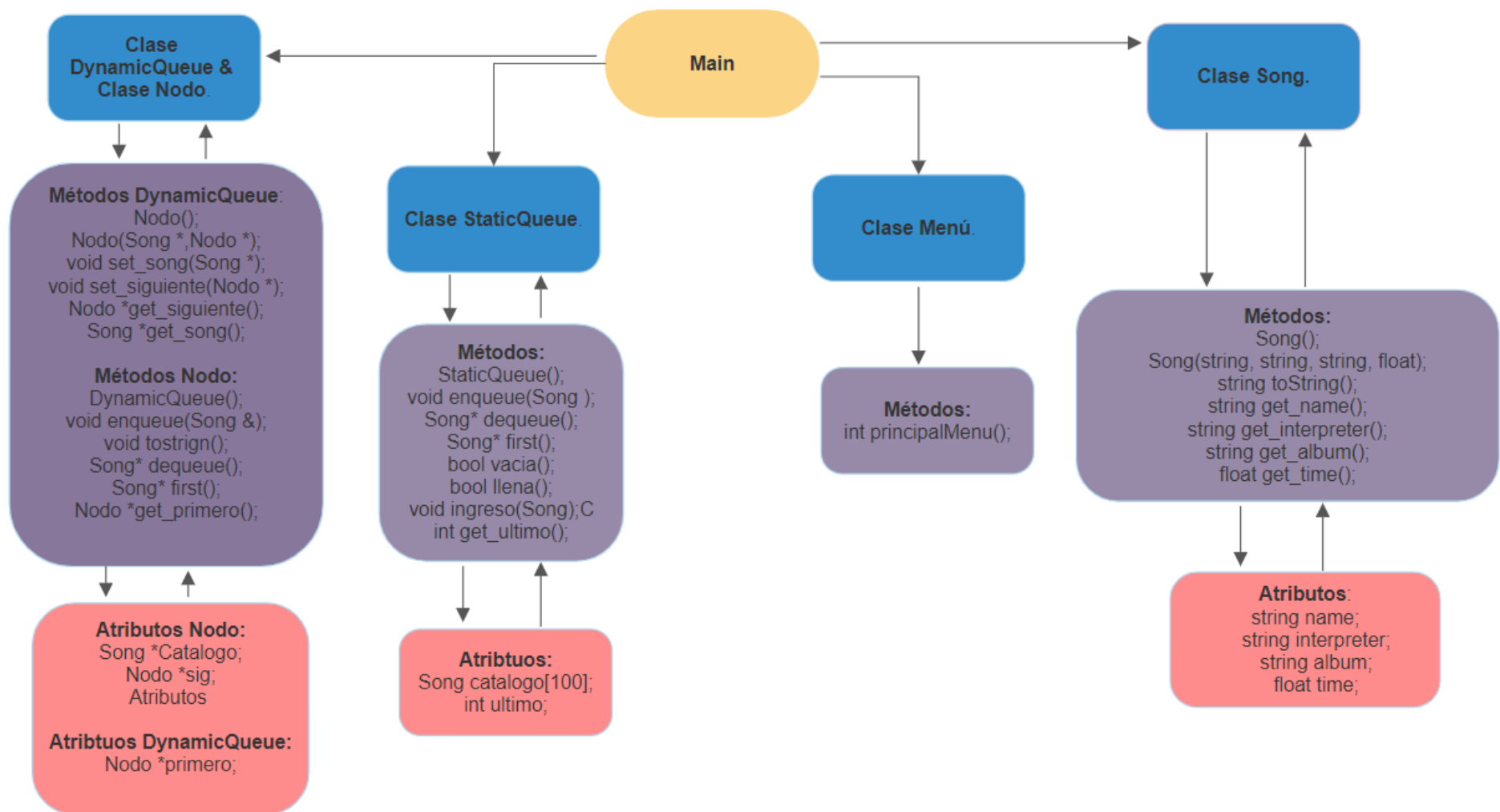
Profesor: Gómez Anaya David Alejandro

Sección: D08

Sede: Centro Universitario de Ciencias Exactas e
Ingenierías (CUCEI)

Fecha: 31 de mayo del 2020

1. Diagrama de clase propuesto.



2. Objetivo.

Comprender el funcionamiento de pila y las funciones propias de estas orientadas a programación orientada a objetos.

3. Marco teórico.

Para este programa se utiliza material de apuntadores, listas estáticas, uso de diferentes archivos (h y cpp) para la organización de la información.

4. Desarrollo.

El programa fue creado a base de un main el cual proporciona el profesor, los principales retos de este fueron el complemento de el paso de parámetros de las funciones y el desarrollo de una función a base de pseudocódigo.

El programa tiene un main con un switch de 6 opciones y cuatro funciones. Se cuenta con cinco clases (cada una tiene dos archivos cpp y h). Y por último se cuenta con un arreglo de canciones y su contador.

El menú principal se divide en las siguientes opciones:

- 1 agregar acciones al arreglo.
- 2 Encolar en cola estática.
- 3 Encolar en cola dinámica.
- 4 Mostrar canción al frente de la cola.
- 5 Desencolar canción de cola estática.
- 6 Desencolar canción de la cola dinámica.

Funciones.

Las funciones a desarrollar son cuatro nombradas de la siguiente manera addSong, enqueue, enqueue y Showfronts:

addSong : como su nombre lo dice solicita los datos necesarios para agregar una canción al arreglo que se encuentra en el menú.

Enqueue: En el cual manda como parámetros el arreglo de canciones, la lista estática y el contador de canciones. Con lo cual agrega la canción deseada del arreglo a la cola estática.

Enqueue: La segunda función con el mismo nombre, pero debido al polimorfismo podemos hacer uso ya que tenemos diferente parámetro de entrada lista dinámica arreglo de canciones y su contador. Con lo cual agrega la canción deseada del arreglo a la cola dinámica.

Showfronts: recibe la referencia de lista dinámica y estática y muestra su elemento primero de cada cola.

Clases.

Se cuenta con clases llamadas Song, Dynamicqueue, StaticQueue, Menu y Nodo.

La clase Song contiene todos los atributos para poder hacer un registro de la canción.

La clase Dynamucqueue y Nodo: eastado cos clases comparten archivo punto h y cpp la clase nodo contiene un Song apuntador y un Nodo sig. La clase DynamiQueue tiene un Nodo a primero que se utiliza como el ancla.

La clase StaticQueue. Contiene un arreglo de Song que se nombra como catalogo y un int llamado último que es nuestro contador

La clase menú: Solo realiza impresión de e datos, pero no contiene atributos.

5. Pruebas y resultados.

Las pruebas se basaron en la creación de funciones que hacían impresiones de datos para corroborar que las colas estén haciendo el trabajo según lo planeado. Estas funciones fueron removidas con el fin de que la evaluación se facilitara al docente.

Los errores más comunes en la prueba era que las funciones no realizaban los cambios necesarios o tenía problemas en tiempo de ejecución cuando las colas estaban vacías.

Después de muchas coerciones se logro hacer que el programa pueda cumplir con todos las implementaciones requeridas.

```
----- Proyecto cola -----
1. Agregar caciones al arreglos de canciones
2. Encolar en cola estatica
3. Encolar en cola dinamica
4. Mostrar cancion al frente de la cola
5. Desencolar cancion de la cola estatica
6. Desencolar canicon de la cola dinamica
Ingrese opcion
4
Pila Estatica.....
Name:      1
Interpreter: 1
Album:     1
Time:      1
No hay canciones encoladas en dinamica
```

6. Conclusiones.

Este programa hizo que pensara mucho acerca de los apuntadores ya que requería mucho de hacer referencias en los pasos de funciones y las devoluciones de algunos métodos de las clases. Mi principal conflicto fue con el arreglo de canciones del menú para pasar al arreglo dinámico o estático después de mucho tiempo invertido encontré mi solución.

Se puedo concluir que para la elaboración del proyecto se necesita un buen conocimiento del tema de apuntadores ya que lo demás hacerla de las clases lo e practicado mucho en los anteriores trabajos.

7. Apéndice(s).

Para entender completamente el funcionamiento del programa se necesita practicar acerca de los apuntadores, referencias y direcciones de memoria. Además se necesita saber el funcionamiento de las listas dinámicas con respecto de nodos yo en lo personal me ayudo creado código alternativo para hacer pruebas de direcciones de memoria o visualizar videos de gente programando.