
PROYECTO 1 IPC2 “TDA - GRAPHVIZ”

202201318 – Estephanie Alejandra Ruiz Perez

Resumen

Las listas enlazadas son una estructura de datos fundamental que se puede utilizar en la Programación Orientada a Objetos (POO) para crear aplicaciones y programas informáticos. En POO, los objetos pueden interactuar entre sí y una lista enlazada puede ser un objeto que almacena y gestiona datos. Graphviz es un software de visualización de gráficos que puede utilizarse para representar visualmente la estructura de una lista enlazada o cualquier otra estructura de datos. XML (eXtensible Markup Language) es un meta-lenguaje que permite definir lenguajes de marcado adecuados para usos específicos, como almacenar y transferir datos entre aplicaciones, incluyendo los datos almacenados en una lista enlazada. En resumen, las listas enlazadas, POO, Graphviz y XML son herramientas y conceptos que pueden utilizarse juntos para crear aplicaciones informáticas eficientes y efectivas, permitiendo una gestión eficiente de los datos y su representación visual, así como su almacenamiento y transferencia entre aplicaciones.

Palabras clave

- Listas enlazadas
- XML
- POO
- Graphviz

Abstract

Linked lists are a fundamental data structure that can be used in Object Oriented Programming (POO) to create applications and computer programs. In POO, objects can interact with each other and a linked list can be an object that stores and manages data. Graphviz is a graph visualization software that can be used to visually represent the structure of a linked list or any other data structure. XML (eXtensible Markup Language) is a meta-language that allows you to define markup languages suitable for specific uses, such as storing and transferring data between applications, including data stored in a linked list. In summary, linked lists, OOP, Graphviz and XML are tools and concepts that can be used together to create efficient and effective software applications, enabling efficient management of data and its visual representation, as well as its storage and transfer between applications.

Keywords

- *Linked Lists*
- *XML*
- *POO*
- *Graphviz*

Introducción

Comprendiendo la importancia de diversas herramientas técnicas, estamos trabajando en una interesante tarea propuesta por el Centro de Investigación de la Facultad de Ingeniería, la compresión de señales de audio. Este centro se centra en dos parámetros clave de las ondas sonoras: frecuencia y amplitud, que describen la señal de audio en función del tiempo. La frecuencia se mide en hercios (Hz) y representa el número de ciclos por segundo. La amplitud, por otro lado, se expresa en decibeles (dB) y refleja la fuerza del sonido. Pensar NP-hard hace que el problema sea aún más complicado, lo que cuesta mucho tiempo y recursos. Para resolver este problema, el Centro adopta un método de agrupación que transforma datos de tiempo, amplitud y frecuencia en una matriz de patrones de frecuencia y agrupa tuplas con patrones similares. Esto facilita la identificación de frecuencias en señales de audio.

Desarrollo del tema

Al comenzar nuestra travesía en el mundo de la programación, nos enfrentamos a un territorio desconocido lleno de desafíos intrigantes. Sin embargo, a medida que exploramos este nuevo mundo, descubrimos que siempre estamos adquiriendo nuevas habilidades y conocimientos que enriquecen nuestra experiencia.

En este contexto, desarrollemos el análisis de la aplicación. Primordialmente, debemos mantener en mente dos elementos esenciales. En primer lugar, nuestra aplicación debe ser amigable para el usuario, lo que implica contar con un menú interactivo y de fácil uso. Cuanto más sencillo sea para el usuario, mayor será la aceptación del proyecto.

En la implementación de herramientas como archivos XML, no solo debemos estar preparados para leer y escribir en ellos, sino que también debemos comprender sus características fundamentales. Los archivos XML son como un iceberg, donde conocer la estructura superficial es solo el comienzo. En realidad, un archivo XML es una jerarquía sintáctica de objetos, y al profundizar en él, encontraremos diversas ramas que a menudo se denominan "nodos".

Para comprender mejor la estructura de la aplicación, aquí presentamos un posible prototipo del menú principal, y más adelante, exploraremos cada una de sus funciones en detalle. Este enfoque nos permitirá navegar con éxito en este emocionante viaje de desarrollo de aplicaciones.

```
*****
Proyecto 1 - Introducción a la Programación 2
*****
1. Cargar archivo
2. Procesar archivo
3. Escribir archivo salida
4. Mostrar datos del estudiante
5. Generar gráfica
6. Inicializar sistema
7. Salir
```

Figura 1. Modelo para menú principal

Fuente: Universidad de san Carlos de Guatemala, Enunciado Proyecto 01, página 07 .

Ya contando con nuestro boceto para un menú principal, podemos analizar y escoger por dónde empezar. Ya que debemos de comenzar realizando lo más “sencillo” para así poder distribuir bien nuestro tiempo y darle la atención necesaria a lo requiere un poco mas de trabajo. Es necesario ordenar nuestras ideas para poder terminar procesos, y no dejar a mitad varios procesos.

a) Cargar Archivo XML

La carga del archivo XML puede ser que parezca complicado, pero es todo lo contrario, mediante librerías podemos facilitar este proceso, por ejemplo

en dicho proyecto es necesario utilizar 2 librerías para llevar esto a cabo, que sería la 1.

FROM: Xml.dom

IMPORT: minidom

Esta nos serviría para darle lectura al XML, que sería uno de los primeros pasos para que nuestro proyecto tenga una función, sin embargo, también necesitaremos lo que es en este caso para realizar escritura XML, para este caso utilizaremos

IMPORT: Xml.etree.ElementTree

AS: ET

Una vez con el XML estipulado, cargado y leído, vendría lo importante, el corazón del proyecto, que sería las listas enlazadas, que estas por su defecto se encontrarán vacías, pero se llenarán con el nombre de las matrices, su amplitud y su tiempo. Las listas enlazadas son perfectas para este problema que se nos plantea debido a que nuestra información se encontrará conectada, lo cual es indispensable para realizar cualquier validación que deseemos, ya sea confirmar que este una matriz, mediante el nombre, que este bien estructurada, entonces mediante nodos la lista enlazada deberá corroborar eso.

b) Procesar Archivo XML y Salida

Luego de haber cargado el archivo nos vamos con el siguiente apartado del menú, que en este caso sería procesar la entrada del XML. En esta sección lo que haremos es identificar las matrices y basándonos en ellas calcular matrices binarias y matrices reducidas, recordando que se estarán imprimiendo mensajes para que el usuario pueda entender cual es cual, y que esta pasando con las matrices. Por ello es relevante que la persona haya ingresado un XML con la estructura correcta, ya que en caso contrario podría ocasionar un error en el programa, o simplemente no se tomaría en cuenta dicha matriz, provocando que no se tome en cuenta las matrices deseadas. A su vez también se hará una suma de tuplas, también se encontrarán patrones de esta. Todo esto puede suceder gracias a las listas enlazadas, que conecta todo y permite por ejemplo se encuentren patrones de la matriz 1 con la matriz 2, ya que se encuentran conectadas, por lo tanto es importante reconocer esta gran labor de las listas, que agilizan y facilitan procesos que probablemente sin ellas nos costaría mucho obtener. Una vez de haber

obtenido todos los resultados al momento de procesar el archivo, nos devolverá al menú para que así podamos obtener los resultados de la matriz reducida en formato XML, que debería ser tan sencillo como colocar la ruta en la cual se desea el archivo de salida.

c) Datos del estudiante

En este apartado pues no cumple alguna función como tal, sin embargo es un apartado importante ya que muestra los datos de la persona que realizó dicho proyecto, a su vez en que semestre se encuentra, el curso, sección, y otros datos relevantes para su identificación.

```
-----  
DATOS ESTUDIANTE  
-----  
> Estephania Alejandra Ruiz Perez  
> 202201318  
> Introducción a la Programación 2 sección C  
> Ingeniería en Ciencias y Sistemas  
> 4to Semestre
```

Figura 2. Datos sobre el estudiante

Fuente: Elaboración propia, 2023

d) Graphviz

Cuando se trabaja con estructuras de datos, es fundamental tener una representación gráfica y detallada del flujo de la aplicación. En muchas ocasiones, la consola o terminal no proporciona suficiente claridad para comprender completamente lo que está ocurriendo. Por lo tanto, es esencial incorporar herramientas gráficas en nuestro desarrollo. Una de estas herramientas es Graphviz, que nos permite generar visualizaciones detalladas de cómo funcionan nuestras operaciones mediante código y lógica. Es innegable que Graphviz es una herramienta crucial cuando se trata de crear representaciones visuales de datos y procesos. Su capacidad para generar gráficos es de gran utilidad en el desarrollo de aplicaciones y proyectos que requieren una comprensión visual de la información.

Sin embargo, es importante señalar que la implementación de Graphviz puede ser un desafío. A menudo, encontrar documentación detallada y ejemplos prácticos puede resultar una tarea laboriosa. La curva de aprendizaje puede ser difícil al principio, ya que es necesario familiarizarse con su sintaxis y configuración.

A pesar de estos obstáculos iniciales, con perseverancia y la búsqueda de recursos adecuados, es posible dominar esta valiosa herramienta. En mi experiencia, logré integrar Graphviz en mi proyecto utilizando la biblioteca de Python "os". Esto me permitió superar las barreras iniciales y aprovechar al máximo su potencial. En última instancia, la inversión de tiempo y esfuerzo en aprender a utilizar Graphviz vale la pena debido a su capacidad para mejorar significativamente la representación visual de los datos y procesos.

e) Inicializar sistema

Esta función del menú será muy simple, no tendrá mucho problema en entenderle o hacerlo, en este caso dicha función servirá para que al momento de escogerla pueda borrar la ruta anteriormente ingresada y a su vez limpiar las listas para que así se pueda volver a utilizar el sistema desde 0

Sin embargo, luego de mencionar todas las funciones que posee nuestro menú hay que retomar en las listas enlazadas, ya que sería el tema importante en este proyecto, por lo tanto es importante conocer en que ocasiones es correcto utilizarlas y cuando no.

f) Cuando utilizar listas enlazadas

Las listas enlazadas son estructuras de datos versátiles que ofrecen ventajas significativas en ciertos escenarios. Una de sus principales fortalezas radica en su capacidad para realizar inserciones y eliminaciones de elementos de manera eficiente, lo que las hace ideales en situaciones donde la manipulación frecuente de datos es esencial. Además, las listas enlazadas permiten una gestión dinámica del tamaño, lo que resulta beneficioso cuando no se conoce de antemano la cantidad exacta de elementos que contendrá la

estructura de datos. También son la elección predilecta para implementar pilas y colas, dos estructuras fundamentales en programación.

Sin embargo, hay casos en los que es mejor evitar el uso de listas enlazadas. Si necesitas un acceso aleatorio constante a los elementos (por ejemplo, acceso por índice), las listas enlazadas pueden ser ineficientes en comparación con otras estructuras como arrays o matrices. Además, tienen una sobrecarga de memoria debido a los punteros utilizados para enlazar los nodos, por lo que si la eficiencia en el uso de la memoria es una prioridad, otras estructuras de datos podrían ser más adecuadas. Además, ten en cuenta que la implementación de listas enlazadas puede ser más compleja y propensa a errores que otras estructuras más simples como arrays.

Por lo tanto, siempre el uso de algo en específico como en este caso listas enlazadas queda a discreción de la persona que desea utilizarlo, siempre evaluando si es la mejor opción, o si existe alguna que sea mas eficiente.

Conclusiones

- ◆ La programación orientada a objetos (POO) facilita la organización y reutilización del código, promoviendo el modularidad y el mantenimiento eficiente.
- ◆ XML es un formato versátil para representar datos estructurados, utilizado en intercambio de datos y configuración de aplicaciones.
- ◆ Las listas enlazadas son ideales para inserciones y eliminaciones frecuentes, pero menos eficientes en acceso aleatorio.
- ◆ Graphviz es una herramienta poderosa para visualizar datos y procesos, aunque requiere cierto aprendizaje inicial para aprovecharla plenamente.

Referencias bibliográficas

- Estructuras de datos: listas enlazadas, pilas y colas.
(n.d.). Retrieved September 1, 2023,
from <https://calcifer.org/documentos/librognome/glib-lists-queues.html>
- Mi Diario Python. (2018, July 1). Linked List: Listas enlazadas - Implementación en Python. Retrieved September 1, 2023,
from <https://pythondiario.com/2018/07/linked-list-listas-enlazadas.html>

Anexos:

-Diagrama de clases:

