
PROYECTO 1

202103763 – Jennifer Yulissa Lourdes Taperio Manuel

Resumen

Las listas enlazadas se utilizan para almacenar y organizar datos de manera dinámica. En Python, no hay una implementación nativa de listas enlazadas como en otros lenguajes de programación (C++ o Java) pero se pueden crear fácilmente utilizando clases personalizadas. XML se utiliza para estructurar y almacenar datos, mientras que Graphviz se utiliza para representar visualmente estos datos en forma de gráficos. Juntos, permiten la representación efectiva y comprensible de los datos del proyecto en cuestión. La Programación Orientada a Objetos (POO) es un paradigma de programación basado en el concepto de "objetos", que son instancias de clases, se utiliza para organizar y estructurar el código de una manera más modular y reutilizable en Python. La programación es una herramienta poderosa para resolver problemas en diversos campos, desde la medicina hasta la agricultura. Se pueden utilizar soluciones tecnológicas para abordar desafíos, como la gestión de recursos naturales o la atención médica en áreas remotas. La programación también facilita la colaboración internacional en investigación, desarrollo y proyectos tecnológicos.

Palabras clave

Listas enlazadas, programación, soluciones, estructurar, paradigma.

Abstract

Linked lists are used to store and organize data dynamically. In Python, there is no native implementation of linked lists as in other programming languages (C++ or Java), but they can be easily created using custom classes. XML is used to structure and store data, while Graphviz is used to visually represent this data in the form of graphs. Together, they allow for the effective and understandable representation of project data. Object-Oriented Programming (OOP) is a programming paradigm based on the concept of "objects," which are instances of classes, and is used to organize and structure code in a more modular and reusable way in Python. Programming is a powerful tool for solving problems in various fields, from medicine to agriculture. Technological solutions can be used to address challenges, such as natural resource management or healthcare in remote areas. Programming also facilitates international collaboration in research, development, and technological projects.

Keywords

Linked lists, programming, solutions, structure, paradigm..

Introducción

En la era contemporánea, las herramientas tecnológicas se han convertido en componentes críticos para desentrañar los enigmas de la ciencia y la ingeniería. En este contexto, el Centro de Investigaciones de la Facultad de Ingeniería se ha embarcado en un viaje para desvelar los misterios detrás de las señales de audio. Poniendo especial énfasis en la frecuencia y amplitud, elementos cardinales en la definición de una onda sonora, han adoptado un acercamiento meticuloso y detallado. Sin embargo, la complejidad de este reto está a la par de problemas NP-Hard, demandando una considerable inversión en tiempo y recursos. La adopción de técnicas avanzadas, como la agrupación basada en patrones de frecuencia, promete revolucionar la manera en que comprendemos el sonido.

Desarrollo del tema

Al iniciar el desarrollo del programa se creó el archivo main.py, el mismo se compone de dos funciones opcionesMenu() para visualizar las opciones en el sistema, seguidamente se encuentra la función menu() que contiene las 7 opciones con los respectivos llamados a funciones, cada cual con su declaración de excepción. Posteriormente se crearon los archivos LecturaXML, este contiene funciones de lectura y obtención de datos desde la estructura preestablecida de un archivo xml, ListaSimple contiene las estructuras de datos de apuntadores que

hacen referencia a las clases, prioritariamente importa el archivo Nodo, en el cual se encuentra el nodo general de la estructura del sistema, seguidamente la creación de los archivos Señal, contiene la clase Señal cuyos parámetros son la etiqueta en el primer nivel del archivo xml, nombre, tiempoMaximo, AmplitudMaxima, Tiempo contiene un clase Tiempo con parámetros tiempo y amplitud máxima que estructuran el enlace a Amplitud y dato en el archivo Amplitud. Se ha creado un archivo Graph para la estructura y renderización de los gráficos utilizando Graphviz. Hice uso de la librería Colorama.

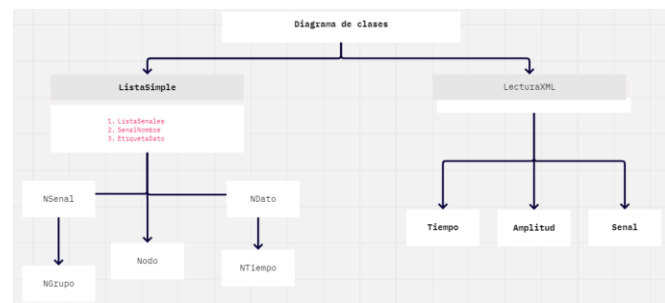


Figura 1. Diagrama de clases

Fuente: elaboración propia.

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.

Hablemos del núcleo de nuestra aplicación. Para tener éxito, 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.

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.

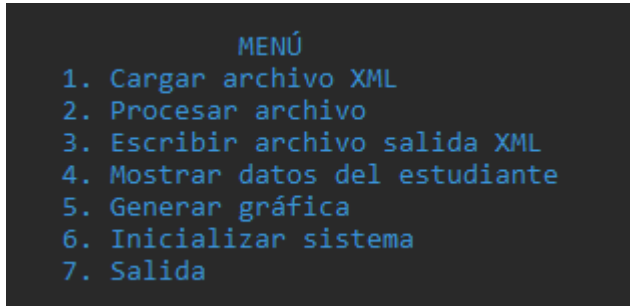


Figura 2. Modelo de menú principal

b) Procesar Archivo XML y Salida

Con el archivo XML ya cargado, el siguiente paso es procesarlo. Aquí, se identifican las matrices y se derivan matrices binarias y reducidas. Se proporcionan mensajes claros al usuario para mantenerlo informado sobre los procedimientos. La estructura correcta del XML es crucial para evitar errores. Las operaciones, desde sumar tuplas hasta encontrar patrones, se realizan eficientemente gracias a las listas enlazadas. Estas estructuras facilitan la identificación de patrones entre matrices. Después del procesamiento, el usuario puede generar resultados en formato XML.

c) Datos del estudiante

Aunque este segmento no tiene funcionalidades específicas del programa, es relevante para identificar al creador del proyecto, proporcionando detalles como el semestre académico, el curso, la sección, entre otros datos relevantes.

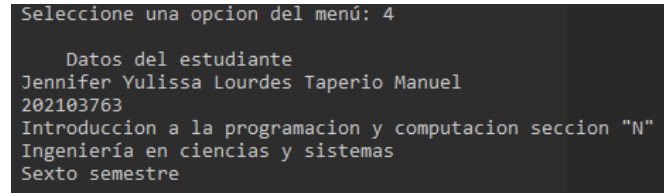


Figura 3. Modelo de menú principal

Aunque este segmento no tiene funcionalidades específicas del programa, es relevante para identificar al creador del proyecto, proporcionando detalles como el semestre académico, el curso, la sección, entre otros datos red

d) Graphviz

Al trabajar con estructuras de datos, una visualización precisa es esencial. Graphviz es una herramienta que genera representaciones visuales claras del flujo y lógica del código. Aunque su implementación puede tener un desafío inicial debido a la necesidad de familiarizarse con su sintaxis, el resultado final justifica el esfuerzo. En este proyecto, se logró su integración con la ayuda de la biblioteca "os" de Python.

e) Inicializar sistema

Este módulo permite al usuario reiniciar el sistema. Con un simple clic, borra la ruta ingresada anteriormente y limpia las listas, preparando el sistema para una nueva serie de operaciones desde cero.

f) Cuando utilizar listas enlazadas

Las listas enlazadas son estructuras dinámicas excelentes para ciertas aplicaciones, especialmente cuando se requieren inserciones y eliminaciones frecuentes. Pero en situaciones donde el acceso

directo es fundamental, pueden no ser la mejor elección. Como con cualquier herramienta, es crucial evaluar su idoneidad para la tarea en cuestión.

Conclusiones

- La POO optimiza la estructuración y reutilización del código.
- XML es un estándar fiable para la representación y transferencia de datos.
- Las listas enlazadas son útiles para manipulaciones específicas, pero no siempre son la opción óptima.
- Graphviz, a pesar de su curva de aprendizaje, es invaluable para la visualización de datos.

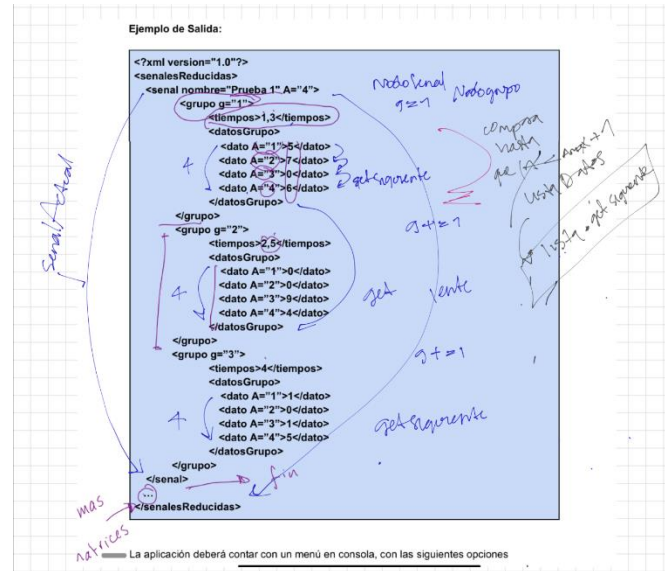


Figura 4. Recopilación de Ideas

Referencias bibliográficas

1. Listas enlazadas y estructuras de datos:
García, L. (2018). Estructuras de datos: Desde las listas enlazadas hasta árboles. Editorial Universitaria, Ciudad de México.
2. XML y su aplicación:
Pérez, J. & Ramírez, A. (2019). XML: El lenguaje de marcas extensible en la era digital. Ediciones Informáticas, Bogotá.
3. Programación orientada a objetos:
Hernández, M. & Ruiz, P. (2020). Introducción a la programación orientada a objetos con Python. Editorial Técnica, Buenos Aires.
4. Graphviz y visualización de datos:
Soto, V. (2017). Graphviz: Herramientas y técnicas para la visualización de estructuras de datos. Publicaciones Ingeniería, Santiago..

Anexos:

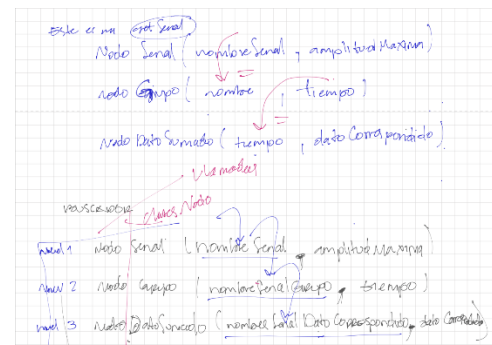


Figura 5. Recopilación de Ideas