
SISTEMA DE GESTIÓN DE CAMPOS AGRÍCOLAS PARA LA OPTIMIZACIÓN DE SENSORES Y ESTACIONES USANDO LISTAS ENLAZADAS

Carné 202300392 – Derek Alessandro Cordova Corado

Resumen

El presente proyecto desarrolla un sistema de gestión de campos agrícolas utilizando estructuras de datos personalizadas en Python. El sistema carga información desde archivos XML y construye matrices de frecuencias para sensores de suelo y cultivo. A partir de estas matrices, identifica patrones y realiza la reducción de estaciones mediante agrupamiento, optimizando la representación de los datos. Además, genera archivos XML de salida y produce gráficas en formato PNG con Graphviz, permitiendo visualizar las relaciones entre estaciones y sensores. La relevancia del proyecto radica en la aplicación práctica de listas enlazadas implementadas manualmente, que refuerzan la comprensión de algoritmos y estructuras de datos. De esta manera, se demuestra cómo el diseño de estructuras desde cero puede aplicarse a problemas reales de ingeniería, favoreciendo la organización y análisis de información agrícola, al mismo tiempo que se fortalece el aprendizaje en programación estructurada.

Palabras clave

Listas enlazadas, sensores, estaciones, XML, Graphviz

Abstract

El presente proyecto desarrolla un sistema de gestión de campos agrícolas utilizando estructuras de datos personalizadas en Python. El sistema carga información desde archivos XML y construye matrices de frecuencias para sensores de suelo y cultivo. A partir de estas matrices, identifica patrones y realiza la reducción de estaciones mediante agrupamiento, optimizando la representación de los datos. Además, genera archivos XML de salida y produce gráficas en formato PNG con Graphviz, permitiendo visualizar las relaciones entre estaciones y sensores. La relevancia del proyecto radica en la aplicación práctica de listas enlazadas implementadas manualmente, que refuerzan la comprensión de algoritmos y estructuras de datos. De esta manera, se demuestra cómo el diseño de estructuras desde cero puede aplicarse a problemas reales de ingeniería, favoreciendo la organización y análisis de información agrícola, al mismo tiempo que se fortalece el aprendizaje en programación estructurada.

Keywords

Linked lists, sensors, stations, XML, Graphviz

Introducción

El análisis y gestión de datos en la agricultura moderna requieren de herramientas informáticas capaces de optimizar la información recolectada en estaciones y sensores. El uso de estructuras de datos personalizadas resulta clave, ya que permite comprender cómo organizar, recorrer y procesar información de manera eficiente. En este contexto, el presente proyecto desarrolla un sistema de gestión de campos agrícolas implementado en Python, utilizando listas enlazadas construidas manualmente en lugar de librerías nativas. El sistema carga información desde archivos XML, construye matrices de frecuencias para sensores de suelo y cultivo, identifica patrones, genera matrices reducidas y permite visualizar resultados a través de gráficas con Graphviz. El propósito de este ensayo es describir el diseño y funcionamiento de cada componente, mostrando la relevancia de implementar estructuras propias como base para la resolución de problemas reales en ingeniería.

Desarrollo del tema

a. Importancia de las estructuras de datos personalizadas

El proyecto se fundamenta en la creación de listas enlazadas propias, evitando el uso de listas nativas de Python. Cada lista enlazada permite insertar, recorrer y manipular nodos que representan entidades como campos, estaciones, sensores o frecuencias. Esta decisión fortalece el aprendizaje del estudiante al comprender de forma explícita el manejo de memoria y el encadenamiento de nodos.

b. Lectura, procesamiento y escritura de XML

El sistema carga un archivo XML que contiene la definición de campos agrícolas, estaciones base y sensores de cultivo y suelo. Una vez cargada la información, esta se almacena en listas enlazadas para su posterior procesamiento. También se implementa un módulo de escritura que genera archivos XML de salida con la información reducida, lo que garantiza la persistencia y reutilización de los datos procesados.

c. Manejo de matrices de frecuencias, patrones y reducidas

A partir de la información cargada, se construyen matrices de frecuencias ($F[n,s]$ para suelo y $F[n,t]$ para cultivo). Posteriormente, estas se transforman en **matrices de patrones (Fp)**, que representan la presencia o ausencia de valores. Finalmente, se obtienen las **matrices reducidas (Fr)**, en las que estaciones con comportamientos similares se agrupan, disminuyendo la complejidad de análisis sin perder información relevante.

d. Generación de gráficas con Graphviz

La visualización de resultados se implementa mediante la librería Graphviz, que genera imágenes en formato PNG representando las relaciones entre estaciones y sensores. Estas gráficas permiten comprender de forma intuitiva las matrices de frecuencias, patrones y reducidas. El aporte visual facilita la identificación de estaciones con comportamientos similares y fortalece la interpretación de los datos.

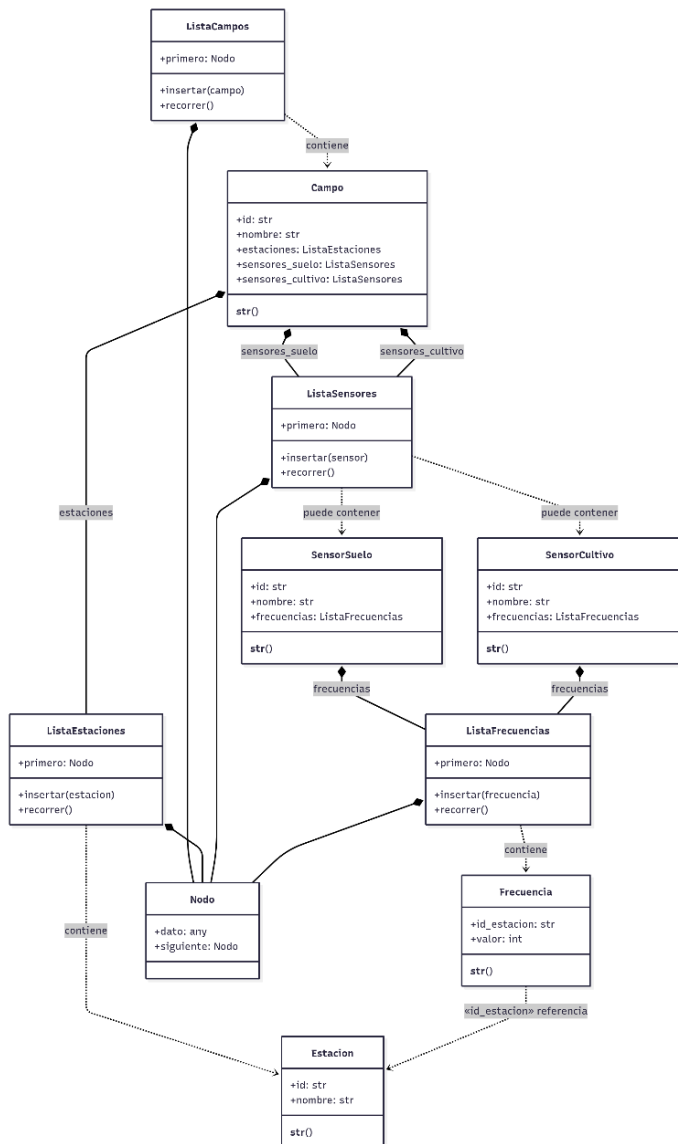


Figura 1. Diagrama de clases

Fuente: elaboración propia, o citar al autor, año y página.

Tabla I. Matriz de frecuencias F[n,s] – Estaciones (n) vs Sensores de suelo (s)

Estación / Sensor	S01	S02	S03	S04	S05
E01	3	0	5	2	1
E02	0	4	1	0	2
E03	6	2	0	3	0
E04	1	1	2	4	5

Fuente: elaboración propia.

Descripción: la tabla presenta la matriz $F[n,s]$ con el número de lecturas registradas por sensor de suelo en cada estación. Los ceros indican ausencia de lecturas para la combinación estación-sensor.

Conclusiones

El desarrollo de este proyecto permitió evidenciar la importancia de una correcta estructuración y gestión de datos para el funcionamiento eficiente de sistemas informáticos, especialmente aquellos orientados al manejo de sensores, estaciones y frecuencias. A lo largo del proceso, se destacó la relevancia de aplicar principios sólidos de diseño de software, tales como la modularidad, la reutilización de componentes y la claridad en la definición de clases y relaciones.

Uno de los principales aportes consiste en la integración de diferentes módulos que facilitan la administración y consulta de información, optimizando tanto el rendimiento como la escalabilidad del sistema. El análisis realizado también puso en manifiesto la necesidad de mantener buenas prácticas de documentación y organización del código, aspectos fundamentales para el mantenimiento y evolución futura del proyecto.

Asimismo, se reconoce el papel fundamental de las bases de datos como eje central para el almacenamiento y recuperación de datos, reafirmando la importancia de elegir estructuras adecuadas y tecnologías pertinentes para cada caso de uso.

Como reflexión, surge la pregunta sobre cómo podrían evolucionar estos sistemas ante la incorporación de inteligencia artificial o big data, y qué retos éticos y técnicos deberían abordarse en escenarios de recopilación masiva de datos.

Finalmente, se recomienda a los futuros desarrolladores profundizar en metodologías ágiles y en el estudio de nuevas tecnologías de bases de datos, con el fin de enriquecer la calidad y la robustez de proyectos similares.

Referencias bibliográficas

1. Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of Database Systems* (7th ed.). Pearson.
2. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.
3. Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson.
4. Ullman, J. D. (1988). *Principles of Database and Knowledge-base Systems*. Computer Science Press.