
PROYECTO 2 – MATRICES Y TDA's

201900532 – Juan José Gerardi Hernández

Resumen

El manejo de datos es esencial para la obtener información que puede ser utilizada en diversos campos tanto como educación, salud, economía, etc. y con ello se han implementado diversas soluciones por medio de herramientas como lo son las aplicaciones que están orientadas al manejo de ellos, en estos programas se implementan diversas estructuras como pueden ser los TDA's (Tipo de Datos Abstractos) las cuales son una colección de elementos llamados nodos interconectados por punteros permitiendo el acceso a datos en posiciones como anterior, arriba y abajo. Esto se logra a través de la programación orientada a objetos (POO) la cual es una forma muy especial de programar muy cercana a como expresaríamos las cosas en la vida real que otros paradigmas de programación.

Las matrices dispersas, utilizadas como TDA (Tipos de Datos Abstractos), se representan mediante coordenadas x e y, que corresponden a columnas y filas, respectivamente. Estas matrices admiten métodos como el recorrido, que permite explorar la lista en estas dos dimensiones.

Palabras clave

- MATRIZ
- TDA
- POO
- PYTHON
- LISTA ENLAZADA

Abstract

Data management is essential for obtaining information that can be used in various fields such as education, health, economics, etc. Various solutions have been implemented through tools like applications that are oriented towards data handling. These programs implement various structures, such as TDA's (Abstract Data Types), which are collections of elements called nodes interconnected by pointers, allowing access to data in positions like previous, up, and down. This is achieved through object-oriented programming (OOP), which is very special way of programming closely resembling how we express things in real life than other programming paradigms.

Sparse matrices, used as TDA (Abstract Data Types), are represented by coordinates x and y, corresponding to columns and rows, respectively. These matrices support methods such as traversal, which allows exploring the list in these two dimensions.

Keywords

- MATRIX
- ADT
- OOP
- PYTHON
- LINKED LIST

Introducción

Al desarrollar una aplicación hay que tener en cuenta diversos factores como puede ser el acceso y conocimiento de herramientas para utilizar y planificar los procesos de desarrollo donde se de importancia a la experiencia de usuario y no sea muy complejo el uso y así como su funcionalidad donde se pueda cubrir las necesidades o funciones que se requieren. Es importante tener los conocimientos sobre los recursos que se tienen para poder implementarlos, un ejemplo podría ser en el desarrollo de este proyecto donde se utilizaron diversos recursos como TDA's y se desarrolló en un entorno gráfico por lo que se implementaron librerías que proporcionaron los medios para poder crear la aplicación.

Desarrollo del tema

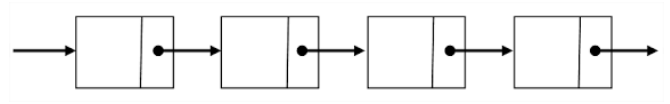
Se desarrollo una aplicación en la cual se usaron distintas herramientas las cuales facilitaron de cierto modo su desarrollo como puede ser el lenguaje de programación Python el cual al ser de un alto nivel es en cierto punto fácil de comprender tanto su lógica como su sintaxis. Se hizo uso de 3 librerías las cuales se detallarán a continuación

OS	Proporciona funciones y métodos que permiten interactuar con el sistema operativo.
TKINTER	Es utilizada para crear interfaces graficas de usuario GUI de manera sencilla. Permite crear objetos como ventanas, botones, Cuadros de dialogo, por mencionar algunos.
ElementTree	Es utilizada para manipular documentos de formato XML lo que quiere decir que puede crear, modificar y analizar arboles de elementos XML.

En parte de la lógica de desarrollo para encontrar una solución se hizo uso de TDA's en este caso se usaron específicamente 2.

• LISTA ENLAZADA SIMPLE

- ➔ Representación gráfica de la lista enlazada simple



- ➔ Código en Python de una lista enlazada simple:

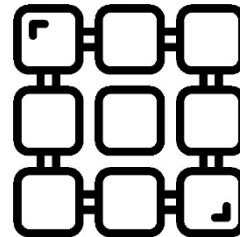
```

1 class Nodo:
2     def __init__(self,dato,indice) → None:
3         self.dato = dato
4         self.indice = indice
5         self.next = None
6
7 class listDron:
8     def __init__(self) → None:
9         self.first = None
10        self.i = 0
11        self.lenght = 0

```

• MATRIZ DISPERSA

- ➔ Representación gráfica de una matriz dispersa:



- ➔ Código en Python para crear una matriz dispersa:

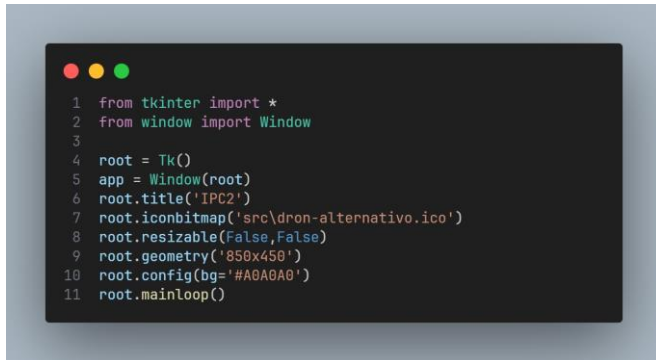
```

1 class NodoM:
2     def __init__(self,x,y,posicion) → None:
3         self.up = None
4         self.down = None
5         self.right = None
6         self.left = None
7         self.posY = y
8         self.posX = x
9         self.pos = posicion
10        self.st = False
11
12 class Matriz:
13     def __init__(self) → None:
14         self.principal = NodoM(-1,-1,'raiz')

```

Descripción de Interfaz:

La interfaz de usuario se trato que fuera lo mas simple posible y fuera fácil de usar, se desarrolló con la librería Tkinter la cual proporciona diversas herramientas para poder crear ventanas entre otros widgets los cuales son objetos que representan elementos de la interfaz.

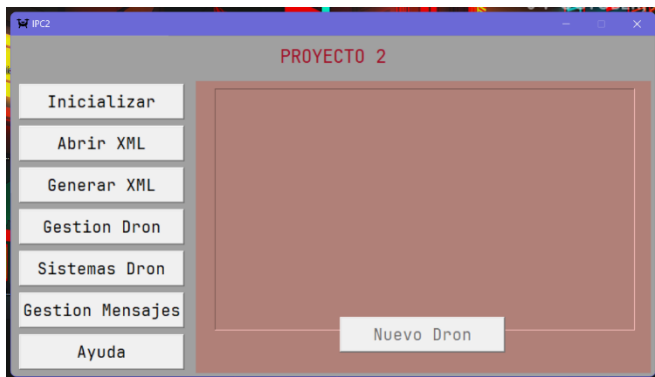


código para crear ventana principal de la interfaz.

- Interfaz principal

Esta interfaz proporciona 7 opciones las cuales permiten realizar distintos análisis del Documento de entrada que sería el XML.

Un ejemplo seria la opción de “Abrir XML” la cual leer el documento y extraer toda su información para así poder analizada por las distintas funciones



- Generar XML

Una de las funciones importantes en el proyecto es la creación del documento XML (Extensible Markup Language) el cual es un lenguaje de marcado que permite leer datos de un formato legible por humanos y estructurado.

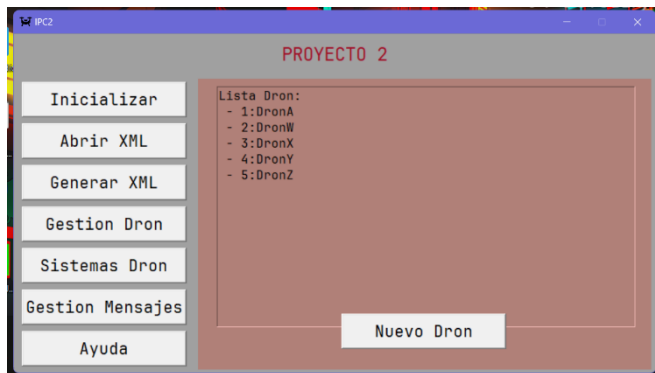
Una de las características del XML es que evita la perdida de la integridad de los datos gracias a que permite transferirlos junto con su descripción, otra característica es que permite almacenar configuraciones lo que ayuda a las aplicaciones a leer y guardar nuevas configuraciones en el documento.



Ejemplo de XML que crea la aplicación

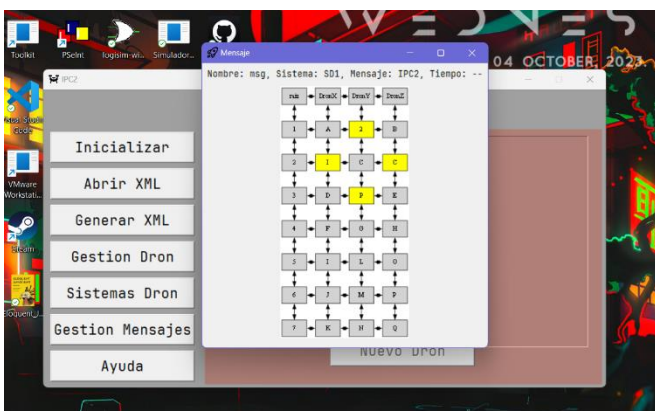
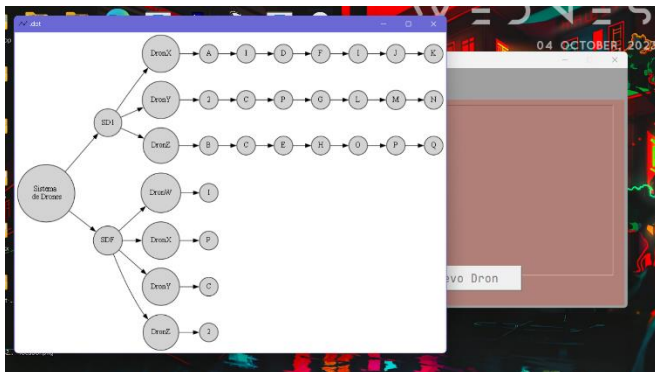
- Gestión Dron:

En esta opción se permite registrar nuevos drones y agregarlos a la lista, donde se puede visualizar de forma ordenada alfabéticamente los drones.



- Sistemas Dron:

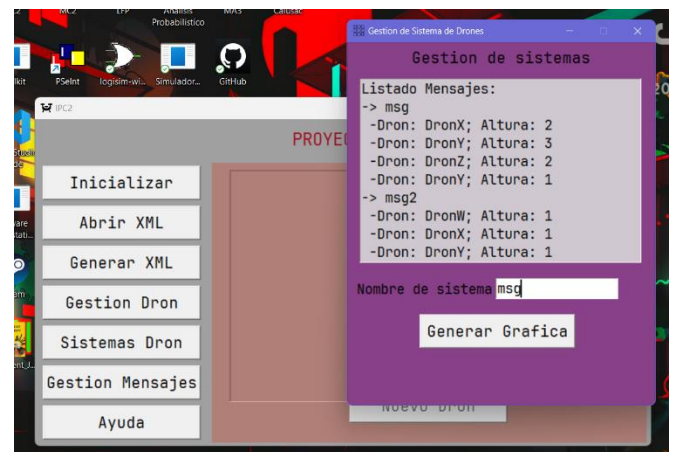
Aquí se hace uso de otra herramienta muy valiosa como lo es Graphviz la cual nos ayuda a generar arboles o grafos donde se ordenan los datos que se analizan de una forma que se puede ver su jerarquía. Aunque Graphviz es un poco complejo de entender es bastante importante su implementación en el desarrollo del proyecto ya que también se utilizó en la opción de gestión de mensajes para mostrar la matriz y las celdas donde van los caracteres.



- Gestion Mensajes:

Aquí es junto con generar XML otra de las funciones importantes en el proyecto ya que aquí se toman las instrucciones del mensaje deseado y se busca la palabra que se desea transmitir, para ellos se recorre la matriz ya sea desde un Dron (nodo) que se encuentra en Y altura o uno nuevo donde se busca el dron especificado y se mueve a la posición deseada y como se mostro anteriormente se genera una grafica con las letras marcadas y acompañado de un texto donde se especifican los datos del mensaje como puede ser:

- Tiempo
- Mensaje
- Sistema
- Nombre de mensaje



Conclusiones

→ Hoy en día las opciones de herramientas para implementar en proyectos son variadas y presentan distintos niveles de complejidad en su uso.

→ Las listas enlazadas son estructuras que presentan grandes ventajas como puede ser que son estructuras de datos que no presentan un espacio fijo en memoria.

→ Cada problema que se plantee en programación se puede solucionar de diversas maneras por lo que no hay una sola solución.

→ Los archivos XML se usan para guardar distintos tipos de datos. ¿Pero hasta donde podría llegar su importancia de implementación en sistemas actuales?

Referencias bibliográficas

- Díaz, G. *Listas Enlazadas* (Doctoral dissertation, Universidad de Los Andes, Mérida 5101 Venezuela).
- (Álvarez, s/f)
Álvarez, P. M. Á. (s/f). *Qué es la Programación Orientada a Objetos*. Dynadata.com. Recuperado el 4 de octubre de 2023, de <http://dynadata.com/ITVER/Docs/Simulacion/UNIDAD%203%20HERRAMIENTAS%20DE%20PROGRAMACION/La%20Programacion%20Orientada%20a%20Objetos.pdf>
- (s/f)
(S/f). Amazon.com. Recuperado el 5 de octubre de 2023, de <https://aws.amazon.com/es/what-is/xml/#:~:text=El%20lenguaje%20de%20marcado%20extensible,datos%20y%20aplicaciones%20de%20terceros>.

ANEXOS

