

---

## PROYECTO NO. IPC2 D S2 2023

---

202001523 – Luis Gabriel Lopez Polanco

### Resumen

El Proyecto 2 tiene como objetivo general desarrollar una solución integral que utilice tipos de datos abstractos (TDA) y visualización de datos a través de Graphviz, todo ello bajo el enfoque de la programación orientada a objetos (POO). Los objetivos específicos incluyen la implementación de POO mediante el lenguaje Python, el uso de estructuras de programación secuenciales, cíclicas y condicionales, la visualización de TDA's mediante Graphviz y la utilización de archivos XML como entrada para la lógica y el comportamiento de la solución.

En el contexto actual de seguridad y comunicación, la necesidad de transmitir información de manera segura se ha convertido en una prioridad. En este ensayo, se abordará el desarrollo de una solución que cumple con el objetivo de crear una tecnología que permita el envío de mensajes encriptados de forma que no puedan ser interceptados y descifrados por personal o instituciones no autorizadas.

### Palabras clave

- Tkinter
- Listas Enlazadas
- XML
- Graphviz

### Abstract

*Project 1 aims to develop a comprehensive solution that utilizes Abstract Data Types (ADT) and data visualization through Graphviz, all under the paradigm of Object-Oriented Programming (OOP). The specific objectives include implementing OOP using the Python programming language, using sequential, cyclic, and conditional programming structures, visualizing ADTs through Graphviz, and using XML files as input for the solution's logic and behavior.*

*In the current security and communication context, the need to transmit information securely has become a priority. In this essay, the development of a solution that meets the objective of creating a technology that allows the sending of encrypted messages so that they cannot be intercepted and decrypted by unauthorized personnel or institutions will be addressed.*

### Keywords

- Linked Lists
- Tkinter
- XML
- Graphviz

## Introducción

El problema que se aborda en este proyecto se centra en la creación de un sistema de comunicación segura que consta de dos componentes principales: un componente emisor del mensaje y un componente receptor del mensaje.

La propuesta de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala se basa en el uso de "n" drones que pueden elevarse a diferentes alturas y emitir una luz LED de alta emisión. Cada dron, al emitir la luz a una altura específica, representará una letra del alfabeto. De esta manera, el componente receptor podrá medir las alturas de emisión de luz, determinar qué dron emitió la luz, decodificar cada letra y finalmente obtener el mensaje que se desea transmitir.

Este enfoque innovador garantiza la seguridad de la comunicación, ya que la información transmitida no será interceptada ni descifrada por personas no autorizadas, ya que solo el componente receptor tendrá los datos necesarios para la decodificación.

## Desarrollo del tema

El problema que se aborda en este proyecto se centra en la creación de un sistema de comunicación segura que consta de dos componentes principales: un componente emisor del mensaje y un componente receptor del mensaje.

Dron01,3	-	Dron01 a 3 metros (representa la H)
Dron04,4	-	Dron04 a 4 metros (representa la E)
Dron03,4	-	Dron03 a 4 metros (representa la L)
Dron02,4	-	Dron02 a 4 metros (representa la L)
Dron03,5	-	Dron03 a 5 metros (representa la O)
Dron01,8	-	Dron01 a 8 metros (representa el espacio en blanco)
Dron04,7	-	Dron07 a 7 metros (representa la W)
Dron03,5	-	Dron03 a 5 metros (representa la O)
Dron03,6	-	Dron03 a 6 metros (representa la R)
Dron03,4	-	Dron03 a 4 metros (representa la L)
Dron04,1	-	Dron04 a 1 metro (representa la D)

Tabla 1 – Ejemplo de instrucciones que recibe un sistema de 4 drones para emitir el mensaje "Hello World"

Figura 1.

Fuente: Universidad de san Carlos de Guatemala,  
Enunciado Proyecto 2, página 3.

Debido a que un sistema de drones puede tener una letra en varias alturas y/o en diferentes drones, entonces, un mismo mensaje podría ser enviado con distintas instrucciones. El sistema receptor, será capaz de detectar las alturas y los nombres de los drones, de esta forma, podrá buscar en el sistema de drones proporcionado por el Ministerio de Defensa, la letra correspondiente y reconstruir el mensaje.

## Tkinter:

Tkinter es una biblioteca estándar de Python que proporciona una interfaz de usuario gráfica (GUI, por sus siglas en inglés) para crear aplicaciones de escritorio de manera sencilla y eficiente. A continuación, se presenta un resumen de las características clave de Tkinter:

Multiplataforma: Tkinter es multiplataforma y está disponible en la mayoría de los sistemas operativos, lo que permite a los desarrolladores crear aplicaciones que se ejecutan en diferentes plataformas sin cambios significativos en el código fuente.

Soporte para Menús: Los menús y barras de herramientas se pueden crear fácilmente con Tkinter, lo que facilita la organización de las funcionalidades de la aplicación.

Integración con Otras Bibliotecas: Tkinter se puede integrar con otras bibliotecas de Python para ampliar sus capacidades, como la combinación con matplotlib para la generación de gráficos avanzados.

En resumen, Tkinter es una herramienta versátil y accesible para crear interfaces de usuario gráficas en aplicaciones de escritorio con Python. Su simplicidad y compatibilidad multiplataforma la convierten en una elección popular para desarrolladores que desean crear aplicaciones GUI de manera efectiva.

### a. Manejo de archivos Xml

Los archivos XML (Extensible Markup Language) han demostrado ser una herramienta versátil y eficaz en el mundo de la informática y la programación. Su capacidad para almacenar datos estructurados de manera legible tanto por humanos como por máquinas los convierte en una opción valiosa para el intercambio de información y el almacenamiento de configuraciones en diversas aplicaciones. En este ensayo, exploraremos en detalle el manejo de archivos XML en el lenguaje de programación Python, destacando su importancia, ventajas y ejemplos prácticos de uso.

#### Importancia de los Archivos XML

Los archivos XML desempeñan un papel crucial en el intercambio de datos entre diferentes sistemas. Su estructura jerárquica y autoexplicativa permite representar información de manera organizada y flexible. Esto es especialmente valioso en entornos donde es esencial mantener la coherencia y la integridad de los datos, como en la configuración de software, la comunicación entre aplicaciones o la serialización de objetos complejos.

#### Ventajas del Manejo de Archivos XML en Python

Python, un lenguaje de programación ampliamente utilizado, ofrece un sólido soporte para el manejo de archivos XML. Algunas de las ventajas clave de utilizar Python en combinación con archivos XML son:

1. **Facilidad de uso:** Python proporciona bibliotecas estándar, como ElementTree y minidom, que simplifican la lectura, escritura y manipulación de archivos XML. Esto permite a los programadores centrarse en la lógica de la aplicación en lugar de preocuparse por los detalles de análisis y generación XML.

2. **Interoperabilidad:** Dado que XML es un estándar ampliamente aceptado, el uso de Python para el manejo de archivos XML facilita la interoperabilidad entre diferentes sistemas y aplicaciones, ya que XML sirve como un formato de intercambio común.
3. **Flexibilidad:** Python permite una amplia gama de enfoques para el manejo de archivos XML. Los programadores pueden optar por utilizar bibliotecas estándar o herramientas de terceros, como lxml, para abordar requisitos específicos de manera más avanzada.

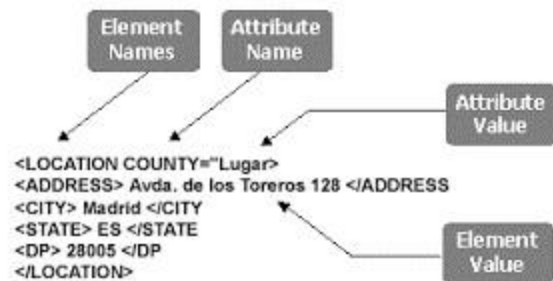


Figura 2. Estructura de un xml.

Universidad Complutense de Madrid. (s/f).

*Hipertexto: el nuevo concepto de documento en la cultura de la imagen.* Recuperado el 3 de septiembre de 2023, de <http://www.hipertexto.info/>

## b. Graphviz

Graphviz es una herramienta excepcional que ha revolucionado la forma en que representamos visualmente datos complejos y relaciones entre elementos. A través de un enfoque basado en gráficos y la generación automatizada de diagramas, Graphviz ha demostrado ser una solución poderosa para una amplia variedad de aplicaciones. Graphviz es una herramienta poderosa y accesible que ha simplificado en gran medida la visualización de datos complejos. Su capacidad para generar gráficos impactantes a partir de descripciones simples lo convierte en una herramienta esencial para profesionales de diversas disciplinas. A medida que continuamos avanzando en la era de la información, Graphviz seguirá desempeñando un papel fundamental en la representación visual de datos y relaciones, ayudando a comunicar información de manera efectiva y comprensible.

## c. Listas Enlazadas

Las listas enlazadas son una estructura de datos fundamental en informática y programación. En Python, esta estructura juega un papel crucial en la manipulación y gestión de datos de manera eficiente. En este ensayo, exploraremos en detalle el uso de listas enlazadas en Python, destacando su importancia, ventajas y ejemplos prácticos de su aplicación en el desarrollo de software.

### -Importancia de las Listas Enlazadas

Las listas enlazadas son una alternativa a los arreglos (arrays) que ofrecen una mayor flexibilidad y eficiencia en ciertos escenarios. A diferencia de los arreglos, las listas enlazadas pueden crecer o reducirse dinámicamente, lo que las hace ideales para situaciones donde el tamaño de los datos es desconocido o cambia con el tiempo. Además, las listas enlazadas permiten la inserción y eliminación de elementos de manera eficiente en cualquier posición de la lista.

## Conclusiones

El desarrollo de esta solución para la comunicación segura mediante drones representa un avance significativo en el campo de la seguridad de la información. La combinación de tecnología de drones, programación en Python y enfoques de programación orientada a objetos permitirá cumplir con los objetivos planteados de manera efectiva.

La aplicación de conceptos como TDA, manipulación de archivos XML y estructuras de programación personalizadas demuestra la versatilidad y la capacidad de adaptación de la solución a diferentes escenarios y requisitos. Además, la generación de reportes visuales con Graphviz facilitará la comprensión y la monitorización del sistema.

## Referencias bibliográficas

- *Documentation.* (s/f). Graphviz. Recuperado el 4 de septiembre de 2023, de <https://graphviz.org/documentation/>
- (S/f). Amazon.com. Recuperado el 4 de septiembre de 2023, de <https://aws.amazon.com/es/what-is/xml/>

Anexos

- **Diagrama de clases:** Diagrama que muestra el flujo de la aplicación.

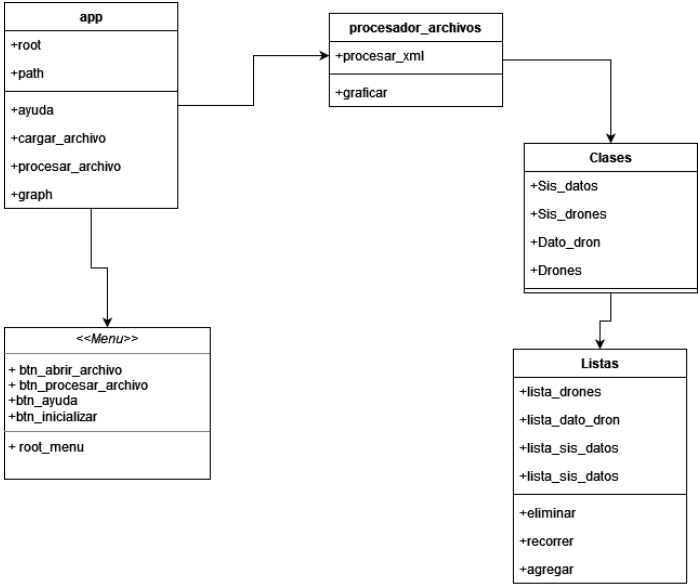


Figura 3. Diagrama de clases.

Fuente: elaboración propia,2023