

---

## Proyecto 2 IPC2 Primer semestre ciclo 2021

---

201701059 – Luis Amilcar Morales Xón

### Resumen

Un proyecto desarrollado en Python que evalúa los conocimientos y la implementación de temas en específico, tales como: Manejo de archivos XML, tipos de datos abstractos, visualizar información en un entorno gráfico hecho con la librería tkinter de python.

El archivo XML es un estándar utilizado a nivel global para almacenar, publicar e intercambiar todo tipo de información, proporciona portabilidad y facilita la gestión de la información a través de distintas plataformas. Permite que diversas aplicaciones de datos puedan funcionar de forma independiente.

Los Tipos de Datos Abstractos (TDA) permiten utilizar nuevos tipos de datos que se definirán especificando sus posibles valores y las operaciones que los manipulan. Cada operación constituye una abstracción funcional. Un tipo de dato abstracto es un tipo definido por el usuario que tiene un conjunto de valores y un conjunto de operaciones. Presentar la información de manera visual al usuario aumenta mejor la comprensión de la información que se quiere transmitir, Tkinter da la posibilidad de crear una infinidad de entornos gráficos con su vasto conjunto de widgets.

### Palabras clave

Matriz

Widget

Graficar

Operaciones

XML

### Abstract

*A project developed in Python that evaluates the knowledge and implementation of specific topics, such as: Handling XML files, abstract data types, visualizing information in a graphical environment made with the python tkinter library.*

*The XML file is a standard used globally to store, publish and exchange all kinds of information, it provides portability and facilitates the management of information through different platforms. It allows various data applications to function independently.*

*Abstract Data Types (ADT) allow the use of new data types that will be defined by specifying their possible values and the operations that manipulate them. Each operation constitutes a functional abstraction. An abstract data type is a user-defined type that has a set of values and a set of operations.*

*Presenting the information visually to the user increases better the compression of the information to be transmitted, Tkinter gives the possibility of creating an infinity of graphical environments with its vast set of widgets.*

### Keywords

*Matrix*

*Widget*

*Graph*

*Operations*

*XML*

## Introducción

Un proyecto que implementa una estructura de datos abstracta que optimiza el uso de la memoria del ordenado, una matriz ortogonal. Las matrices ortogonales se caracterizan por evitar saturar la memoria con datos innecesarios y se centra llevar el orden mediante un conjunto de punteros enlazados de manera correcta a los nodos. Las matrices cargadas desde el archivo XML se someterán a operaciones rotacionales, operaciones de conjunto y etc. Estas operaciones se podrán visualizar en el entorno gráfico creado con la ayuda de la librería de Tkinter de Python.

## Desarrollo del tema

EL proyecto se realizó en el lenguaje Python y consta con un entorno gráfico que contiene una barra de menú que permite la manipulación de los objetos creados a partir de una fuente XML las opciones que se manejan en el programa son los siguientes:

- **Cargar archivo:** En este apartado se abre una ventana de explorador de archivos para seleccionar un archivo XML para el procesamiento de las matrices y proceder a usar las funcionalidades del sistema
- **Operaciones:** Se despliega una lista de operaciones que se le pueden aplicar a una o dos matrices al mismo tiempo, seleccionando una matriz en específico y luego ingresar los datos que se necesitan para la operación o en su defecto se procederá enseguida a visualizarse en el área establecido de la interfaz gráfica.
- **Reporte HTML:** Este reporte ofrece un registro de las matrices cargadas al sistema y también una bitácora de operaciones realizadas en todo el transcurso de la ejecución del programa, esto en formato de logs.
- **Ayuda:** En este apartado podemos seleccionar la opción de *Mostrar información*

*del desarrollador*, que despliega los datos del desarrollador del sistema o bien seleccionar la opción *Documentación del programa*, que desplegará un PDF con los datos solicitados.

## Interfaces gráficas de usuario con Tk

Tk/Tcl ha sido durante mucho tiempo una parte integral de Python. Proporciona un conjunto de herramientas robusto e independiente de la plataforma para administrar ventanas. Disponible para desarrolladores a través del paquete tkinter y sus extensiones, los módulos tkinter.tix y tkinter.ttk.

El paquete tkinter es una capa delgada orientada a objetos encima de Tcl/Tk. Para usar el módulo tkinter, no tiene que escribir ningún código Tcl, pero deberá consultar la documentación de Tk, y ocasionalmente, la documentación de Tcl. El módulo tkinter es un conjunto de funciones que envuelven las implementaciones widgets Tk como clases de Python. Además, el módulo interno \_tkinter proporciona un mecanismo robusto para que los hilos Python y Tcl interactúen.

Las ventajas de los módulos tkinter son su velocidad y que generalmente se suministra de forma nativa con Python. Aunque su documentación es deficiente, existen otros recursos: referencias, tutoriales, libros y otros. El módulo tkinter también es famoso por su aspecto desactualizado, el cual se ha mejorado mucho desde Tk 8.5.

## XML: ¿Qué es y para qué sirve este lenguaje de marcado?

Detrás del diseño y el texto de los sitios web, siempre hay un lenguaje propio y uno de ellos es el XML. Este acrónimo significa Extensible Markup

Language, que es un lenguaje de marcado que define un conjunto de reglas para la codificación de documentos.

El lenguaje de marcado es un conjunto de códigos que se pueden aplicar en el análisis de datos o la lectura de textos creados por computadoras o personas. El lenguaje XML proporciona una plataforma para definir elementos para crear un formato y generar un lenguaje personalizado. un archivo XML se divide en dos partes: prolog y body. La parte de prolog consiste en metadatos administrativos, como declaración XML, instrucción de procesamiento opcional, declaración de tipo de documento y comentarios. La parte de body se compone de dos partes: estructurales y de contenido (presente en los textos simples).

### **XML simplifica el intercambio de datos**

Tanto los sistemas informáticos como las bases de datos contienen información en formatos incompatibles.

Los datos XML se almacenan en formato de texto simple, lo que nos posibilita una forma independiente de almacenar datos. Esto facilita mucho la creación de datos que pueden ser compartidos por diferentes aplicaciones.

Uno de los desafíos más difíciles para los desarrolladores es intercambiar datos entre sistemas incompatibles a través de Internet. El intercambio de datos como XML reduce en gran medida esta complejidad porque los datos pueden ser leídos por diferentes aplicaciones incompatibles.

### **XML simplifica el cambio de plataforma**

La actualización a nuevos sistemas (plataformas de hardware o software) lleva mucho tiempo. Se deben convertir grandes cantidades de datos y los datos incompatibles a menudo se pierden.

Los datos XML se almacenan en formato de texto. Esto facilita la expansión o actualización a nuevos

sistemas de información, nuevas aplicaciones o nuevos navegadores sin pérdida de datos.

### **El XML aumenta la disponibilidad de datos**

Diferentes aplicaciones pueden acceder a tus datos, no solo en páginas HTML, sino también en fuentes de datos XML.

Con el XML, tus datos pueden estar disponibles para todos los tipos de «máquinas de lectura» (computadoras de mano, máquinas de voz, feeds de noticias, etc.) y, además, facilita la accesibilidad para personas con capacidades diferentes, por ejemplo, no videntes.

### **Tipos de datos abstractos.**

Un *Tipo de dato abstracto* (en adelante *TDA*) es un conjunto de datos u objetos al cual se le asocian *operaciones*. El TDA provee de una interfaz con la cual es posible realizar las operaciones permitidas, abstrayéndose de la manera en cómo estén implementadas dichas operaciones. Esto quiere decir que un mismo TDA puede ser implementado utilizando distintas estructuras de datos y proveer la misma funcionalidad.

El paradigma de orientación a objetos permite el *encapsulamiento* de los datos y las operaciones mediante la definición de *clases* e *interfaces*, lo cual permite *ocultar* la manera en cómo ha sido implementado el TDA y solo permite el acceso a los datos a través de las operaciones provistas por la interfaz.

### **Conclusiones**

Es imprescindible entender el comportamiento de una estructura abstracta para poder manipularlo y principalmente poder representarlo dentro de un lenguaje de programación ya que esto permitirá poder modelar cualquier información que se presente en el mundo real y convertirlo en datos que la computadora sea capaz de entender, así como el programador también es capaz de hacerlo

## Referencias bibliográficas

Ignacio Onetto, A. ¿Por qué el archivo XML es importante? Nubox: Software para empresas y contadores. Recuperado de:

<https://blog.nubox.com/software/por-que-el-formato-xml-es-importante#:~:text=Es%20un%20estándar%20abierto%20C%20flexible,puedan%20funcionar%20de%20forma%20independiente.>

De Souza, I. (Jul 12, 2019). XML ¿que es y para qué sirve este lenguaje de marcado? Rockcontent: Blog.

Recuperado de: <https://rockcontent.com/es/blog/que-es-xml/>

Alvarez, M. Angel. ( Dic 11, 2019). ¿Qué es la programación orientada a objetos?. Desarrolloweb.com: Teoría de la programación orientada a objetos.

Recuperado de:

<https://desarrolloweb.com/articulos/499.php>