

---

## PROYECTO 2 – IPC 2

---

201800665 – Samuel Alejandro Pajoc Raymundo

### Resumen

Se realizó un programa capaz de leer y procesar información contenida en un archivo de tipo xml en el cual se encontraba almacenada una imagen formada por asteriscos y guiones (los cuales representaban espacios en blanco), además, esta información fue analizada por distintos procesos, en los cuales, se especificaba que los datos de las matrices, y a su vez se debía validar la creación de distintos grafos que representaran el flujo de los datos dentro de la matriz de salida. Estas especificaciones se encuentran dentro del archivo llamado proyecto 2.

### Palabras clave

- Nodo
- Matriz
- Ciclo
- Archivo
- Datos

### Abstract

*It was created a program capable of Reading and processing information contained in a xml file, furthermore, it contain pictures made by asteriks and scripts (these ones represented the blanck spaces in the file) this information was analyzed for several processes, those ones, were specified that the matrix data, into the xml file, and moreover, it should confirm the creation of several graph's that would represent the data flow into the output matrix. These specifications are expressed into the document called project 2.*

### Keywords

- Node
- Matrix
- Cycle
- File
- Data

## Introducción

La importancia de analizar el comportamiento de un tipo de dato específico de distintos tamaños, como lo son las matrices, permiten que el programador pueda desarrollar un criterio muy detallado al momento de evaluar y validar cada proceso al momento de recibir el conjunto de datos contenidos en las matrices. Se tomo en cuenta que este debía ser creado con Orientación a Objetos, y se observó que cuando se trabaja con una matriz de patrones, esta presenta nuevos retos, ya que los pasos a seguir para su funcionamiento correcto, posee ciertas características que pueden llegar a confundir a un programador principiante, debido a que, si no se posee un buen control de validaciones y una visualización de los posibles errores a cometer por recorrer las filas de datos de una manera no eficiente, esto puede ocasionar fallos en el resto del programa.

Así mismo se comprendido la gran utilidad que presenta el manejo de datos dentro de archivos xml, ya que, de esta forma, cualquier base de datos comprenderá que debe hacer cuando se le presente un conjunto de datos con atributos y con especificaciones del tipo de dato al que pertenece. También se debe reconocer el valor que posee el manejo de estructuras graficas como lo fue al utilizar el lenguaje de tipo .dot, junto al entorno de graphviz, ya que por este medio se pudo generar y visualizar el flujo de los datos en el archivo de salida de la matriz reducida.

## Desarrollo del tema

Al iniciar con el programa el usuario podrá visualizar un menú llamado “menú principal”, el cual contiene seis opciones, las cuales son: cargar archivo, procesar archivo, escribir archivo de salida, mostrar datos del estudiante, generar gráfica y salir.

Dentro de la primer opción llamada “cargar archivo”, se despliega el explorador de archivos, el cual se permitirá encontrar el archivo xml a cargar, si se ingresa una ruta la cual no contenga un archivo del tipo xml, automáticamente se le indicará que no existe tal archivo y será devuelto al menú principal.

Ya que se estableció que se utilizaría el paradigma de objetos, esta primera opción fue creada con una clase, la cual recibe como parámetro la ruta del archivo a cargar, para luego devolver la estructura principal que será de utilidad en la segunda opción.

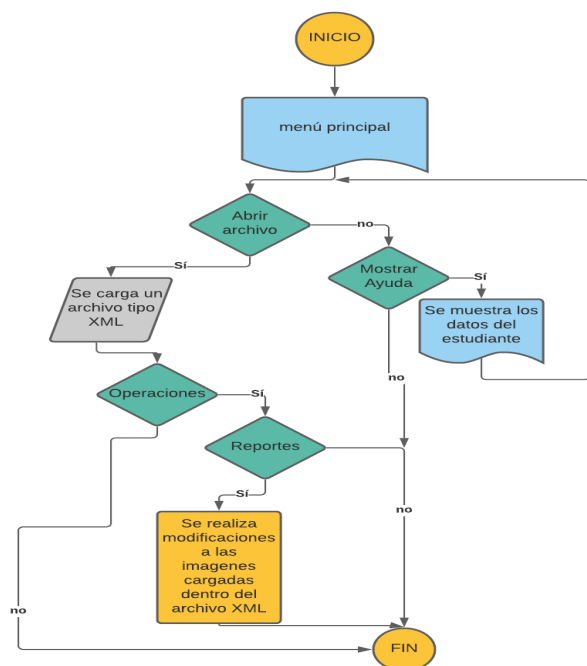
Debido a que todas las tareas serán realizadas por medio de la interfaz grafica aplicada por el estudiante, esto permite tener un entorno más dinámico con el programa y el usuario, además, esto permitirá una familiarización más rápida, ya que se comprenderá mejor el funcionamiento de esta aplicación por medio de botones, cajas de texto y ventanas emergentes.

Después de haber cargado el archivo, es posible ingresar en la segunda opción llamada “Operaciones”, la función de esta opción es leer y almacenar toda la información contenida dentro del archivo xml especificado en la opción anterior. Dentro de esta opción se realizan distintos procesos,

Los cuales se podrá observar en todo momento la edición de las imágenes previamente y después de ser procesadas por la opción que se haya elegido ejecutar.

Dentro de la tercer opción llamada “Reportes”, se encuentra Una serie de componentes html, los cuales le dan forma al reporte solicitado para este proyecto, el cual debe mostrar todas las operaciones que se realicen con las imágenes, además debe de incluir la hora y fecha que se realizó los cambios, así mismo debe mostrar la cantidad de espacios llenos y los que se encuentran en blanco.

Figura No1. Diagrama UML

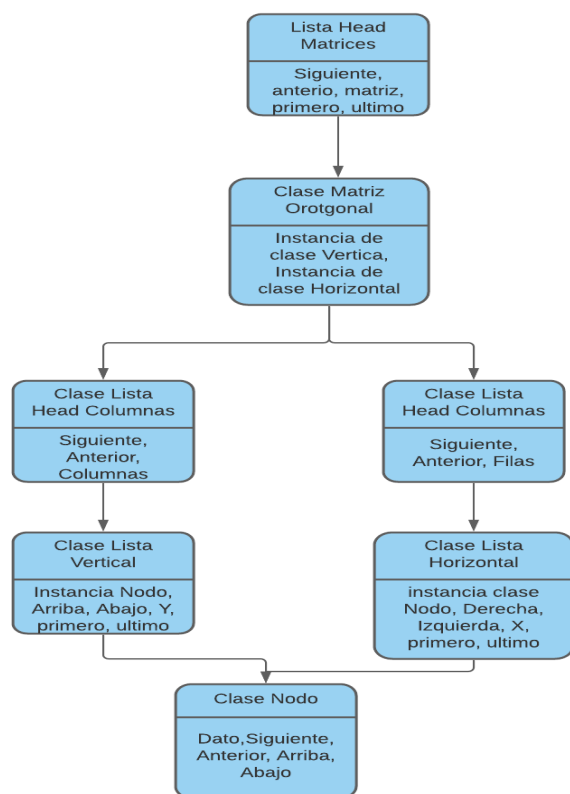


Luego procede la cuarta opción llamada “Ayuda-datos del estudiante”, en la cual como su nombre lo indica, dentro se encuentran los datos del estudiante, esta es la única opción que no necesita validación de ningún tipo, ya que únicamente posee información del desarrollador.

Por ultimo para poder salir y finalizar con el programa, se debe dar un clic sobre los botones que representan el cierre de una ventana, estos generalmente se encuentran en la parte superior derecha.

Todos los procesos de almacenamiento se realizaron utilizando listas doblemente enlazadas, lo que conlleva a diseñar una matriz ortogonal, la cual fuera capaz de almacenar a su vez múltiples matrices, todas de diferentes tamaños, a continuación se presenta un diagrama de clases, de como fue planteada su elaboración.

Figura No.2 Diagrama de clases.



## **Conclusiones**

Se demostró que es muy útil plantear soluciones con orientación a objetos, ya que de esta forma es más eficiente y fácil de crear nuevos elementos de tipo objeto.

Se comprendió y se analizó lo que representa un tipo de dato abstracto, con la implementación de listas doblemente enlazadas, ya que sin esto no hubiera sido posible tener una forma de almacenar tanta información, y que a su vez se respetara las normas de restricción.

Se comprendió el funcionamiento de un archivo de tipo xml al momento de haber creado la clase que permitió leerlo, y se desarrolló un mejor análisis en este tipo de formatos.

Se logró implementar la interfaz grafica de graphviz, utilizando como lenguaje principal Python, con declaraciones de tipo .dot, además se comprendió lo que es un nodo y una arista dentro de un grafo.

Se demostró nuevos métodos en los que se puede trabajar con listas enlazadas simples, las cuales pueden ayudar a crear matrices grandes y complejas, pero se aconseja descartar este método ya que es demasiado laborioso para la creación de una matriz simple, y se recomienda trabajar con las listas que permite implementar python.

## **Referencias bibliográficas**

Charles Severance. Date (2013). Python para informáticos. Addison-Wesley Publishing Company, Inc.