|  |
| --- |
| **PROYECTO NO.2 PRIMER SEMESTRE 2021** |
| **201700747 – Dennis Alexander Gamboa Stokes** |

**Resumen**

La Programación Orientada (POO) a objetos permite que el código sea reutilizable, organizado y fácil de mantener. Sigue el principio de desarrollo de software utilizado por muchos programadores DRY (Don’t Repeat Yourself), para evitar duplicar el código y crear de esta manera programas eficientes mientras Graphviz es una herramienta utilizada por muchas personas que permite la creación de graficas usando un lenguaje especifico.

Se investigo el uso de la Programación orientada a Objetos para la elaboración de una matriz ortogonal y también diferentes métodos para la facilitación del programa como también se investigó el uso del graphviz para poder graficar la matrices que entraron al sistema por medio de una estructura XML que también se investigo acerca de y por último se investigó TDA para ser utilizado en la programación.

**Palabras clave**

Matriz ortogonal , Graphviz, XML Estructura de programación

***Abstract***

Object Oriented Programming (OOP) allows code to be reusable, organized, and easy to maintain. It follows the software development principle used by many programmers DRY (Don't Repeat Yourself), to avoid duplicating the code and thus create efficient programs while Graphviz is a tool used by many people that allows the creation of graphs using a language specific.

The use of Object-Oriented Programming was investigated for the elaboration of a Orthogonal linked list and also different methods for the facilitation of the program, as well as the use of Graphviz to be able to graph the matrices that entered the system through an XML structure. that was also investigated about and finally ADD was investigated to be used in programming.

***Keywords***

Orthogonal linked list, Graphviz, XML, Programming Structures

**Introducción**

En el Presente trabajo o programa se mostrará la información principal de los métodos utilizados en la Programación orientada a objetos (POO) como también la visualización de TDA mediante el uso de Graphviz. Se hará una carga de datos provenientes de una estructura XML y se usara POO para facilitar la interpretación de datos.

Se mostrara la estructura de una matriz ortogonal como también su métodos para la facilitación de control de datos y por ultimo las graficas generada de los datos del XML se mostrara mediante el uso de Graphviz que es una herramienta muy útil para la creación de graficas.

**Desarrollo del tema**

**XML:**

Extensible Markup Language (XML) se utiliza para describir datos. El estándar XML es una forma flexible de crear formatos de información y compartir electrónicamente datos estructurados a través de la Internet pública, así como a través de redes corporativas.

*Usos:*

**Intercambio de datos entre sistemas,**uno de los objetivos fundamentales de XML es permitir la posibilidad de intercambiar datos de forma estructurada entre diferentes sistemas. Al tratarse de un formato de texto plano y ser un lenguaje estandarizado, hace que esta transferencia sea muy ágil e independiente de la plataforma utilizada.

**Base de datos,**XML permite guardar datos de forma estandarizada para luego poder ser tratados por multitud de lenguajes diferentes. Su manejo es mucho más sencillo que bases de datos como MySQL y mucho más rico que utilizar ficheros de texto planos.

**Conversor,**actualmente son muchos los formatos que ofrecen servicios de conversión a XML, como PDF, HTML, .text, .docx o XHTML.

<? xml version = "1.0" standalone = "yes"?>

<conversacion>

<saludo> ¡Hola, mundo! </saludo>

<respuesta> Cuiden al planeta </respuesta>

</conversacion>.

*Figura 1.*Estructura de XML.

Fuente: elaboración propia

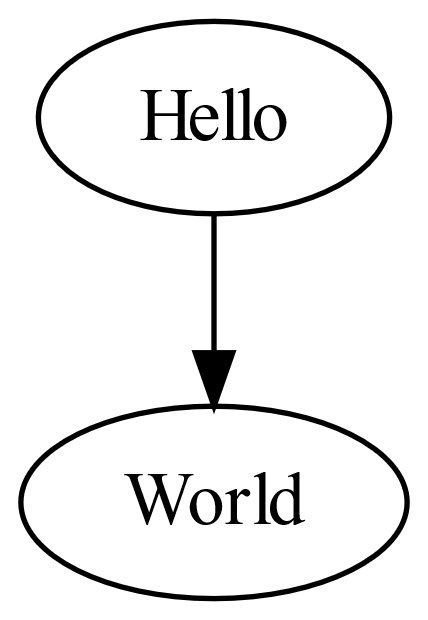
**Graphviz**:

Graphviz (abreviatura de Graph Visualization Software) es un software abierto de libre distribución para graficar, que presenta información estructural en forma de diagramas y puede aplicarse en diversas áreas como el análisis de redes, bioinformática, ingeniería de software, bases de datos, diseño de sitios web, aprendizaje por computadora y tiene interfaces gráficas para otros dominios.

Su modo de utilización se basa en el diseño de pequeños programas que toman descripciones de los diagramas de un lenguaje de texto simple y los dibuja en diversos formatos tales como archivos de imágenes, SVG, PDF ó para desplegarse en exploradores.

En la práctica, generalmente las gráficas se generan de fuentes de datos externas, (típicamente mediante el recorrido de la estructura de datos correspondiente) para crear un programa que pueda ser interpretado y ejecutado por alguno de los componentes de Graphviz y generar la imagen correspondiente.

**digraph** G {Hello**->**World}



*Figura 2.*Estructura y diagrama de graphviz.

Fuente: elaboración propia

**Matriz Ortogonal**:

Las Matrices Ortogonales son estructuras de datos que contienen dos lista doblemente enlazadas además que contiene 4 apuntadores uno que apunta hacia la derecha otro que apunta hacia la izquierda uno que apunta hacia abajo y el ultimo hacia arriba lo cual la convierte en una lista sin fin, cada nodo siempre tiene uno anterior y uno siguiente, su estructura es muy similar a las listas doblemente enlazadas por lo cual comparten características tanto en su implementación como en su manejo aunque requiere un mayor entendimiento del manejo de los punteros.

Características:

No existe ningún nodo que apunte a null.

La lista no tiene fin ya que al llegar al último nodo apunta hacia el nodo anterior

Se accede a la lista mediante el primer nodo o también llamado inicio de la lista.

Si no se tiene cuidado al manejar se pueden crear bucles infinitos.

No tiene acceso aleatorio es decir para acceder a un valor se debe recorrer toda la lista.

Operaciones:

agregar (valor): agrega el valor al final de la lista.

insertar (referencia, valor): inserta el valor después del valor de referencia en la lista.

remover (referencia): elimina el nodo con el valor que coincida con la referencia.

editar (referencia): actualiza el valor de nodo con el valor que coincida con la referencia.

esVacia (): retorna true si la lista está vacía, false en caso contrario.

buscar (valor): retorna la true si el elemento existe en la lista, false caso contrario.

eliminar(): elimina la lista

listar (): imprime en pantalla los elementos de la lista.

Diagram

Description automatically generated

*Figura 3.*Estructura Matriz Ortogonal

Fuente: Google Image

**Estructura de Programación:**

Estructura Condicional:

Permite alterar la secuencia normal de pasos en un paso específico del Algoritmo, para crear 2 alternativas de bloques de ejecución, de manera excluyente entre ambos. En otras palabras: Solo uno de los 2 bloques se ejecutará, nunca ambos bloques y permite decidir por cuál alternativa seguirá el flujo del programa dependiendo del resultado de la evaluación de una condición. Para establecer condiciones complejas se utilizan los operadores relacionales y lógicos

*Estructura cíclica*:

Se llaman problemas repetitivos o cíclicos a aquellos en cuya solución es necesario utilizar un mismo conjunto de acciones que se puedan ejecutar una cantidad específica de veces. Esta cantidad puede ser

Text

Description automatically generatedText

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

En esta imagen se puede apreciar la creación del nodo y la matriz ortogonal con algunos de sus métodos que se usó para aguardar los datos, buscar y eliminar

*Figura 5.*Estructura Matriz Ortogonal Programación

Fuente: Propia

*Figura 6. Uso de Xml minidom para leer Archivo*

Fuente: Propia

Se usa XML minidom para hacer la carga masiva con las validaciones pedidas y poder aguardar la información.

Text

Description automatically generated

Text

Description automatically generated *Figura 7. Creación de la ventana*

Fuente: Propia

Text

Description automatically generated

*Figura 8. Creación de la ventana*

*Fuente Propia*

Text

Description automatically generated

*Figura 9. Creación de la ventana*

*Fuente :Propia*

Text

Description automatically generated*Figura 10.Ingreso de datos a matriz ortogonal*

Text

Description automatically generatedText

Description automatically generated*Fuente :Propia*

*Figura 11. Lectura de datos en etiqueta Imagen*

*Fuente :Propia*

*Figura 12. Creación de Html*

Text

Description automatically generated*Fuente :Propia*

*Figura 13.Creacion de código Graphviz*

*Fuente :Propia*

*Text

Description automatically generated*

*Figura 14. Grafica Creada Con Graphviz*

*Fuente :Propia*

*Diagram

Description automatically generated*

*Figura 15. Diagrama de clases*

*Fuente :Propia*

*Diagram

Description automatically generated*

*Figura 16. Diagrama de actividades*

*Fuente :Propia*

*Diagram

Description automatically generated*

*Figura 17. Diagrama de actividades*

*Fuente :Propia*

*Diagram

Description automatically generated*

*Figura 18. Diagrama de actividades*

*Fuente :Propia*

*Figura 19. Diagrama de actividades*

*Fuente :Propia*

**Conclusiones**

* Las matrices Ortogonales nos permiten un manejo de la información mediante POO que un vector además permite el manejo de datos, ya sea para insertar, eliminar, buscar, modificar datos dentro de los nodos que contiene la lista.
* Graphviz es una herramienta que permite la visualización de estructuras TDA mediante graficas.
* Archivos XML nos permiten hacer una carga masiva de datos en un programa con una estructura muy básica, y fácil de manejar, así como importar datos de un programa
* Estructuras Secuencial permite que una instrucción siga a otra en secuencia.
* Estructuras condicionales son instrucciones que se pueden ejecutar o no en función del valor de una condición.
* Estructuras cíclicas permite ejecutar fragmentos de Código un umero limitado de veces

**Referencias bibliográficas**

Máximo 5 referencias en orden alfabético.

* Estructuras de programación secuenciales, cíclicas y condicionales Disponible en

<https://prezi.com/fmivkws5biwl/estructuras-condicionales-secuenciales-y-ciclicas/>

* Estructura de XML y usos en

<http://www.maestrosdelweb.com/xmlusos/>

* Estructura de Graphviz, Disponible en

https://graphviz.org/

* Programación orientada a objetos, Disponible en  
  <https://desarrolloweb.com/articulos/499.php#:~:text=La%20programaci%C3%B3n%20Orientada%20a%20objetos%20se%20define%20como%20un%20paradigma,los%20objetivos%20de%20las%20aplicaciones>.
* Programación orientada a objetos, Disponible en  
  <https://profile.es/blog/que-es-la-programacion-orientada-a-objetos/>
* <http://www.rdebug.com/2010/10/matriz-ortogonal-estructura-de-datos-en.html#:~:text=una%20matriz%20ortogonal%20es%20una,las%20columnas%20representan%20los%20modelos>.
* https://drive.google.com/file/d/1D3qqnpO33Qc7VH\_9yIGz0qDXHQSmXamU/view