
PROYECTO 1 – IPC2

202103206 – KEWIN MASLOVY PATZAN TZUN

Resumen

En la realización de este proyecto, se buscó poder resolver un problema utilizando para ello tipo de datos abstractos(TDA's), manipulación de datos utilizando tipos de archivos XML, conceptos de memoria dinámica utilizando listas y POO.

Para ello se realizó un programa el cual permite la realización de experimentos mediante muestras de suelos marcianos, esto para que puedan ser estudiados por científicos mediante la carga de información de dichas muestras en el programa por medio de un archivo XML.

Para dicho programa se utilizaron listas simples y programación orientada a objeto cual permitió la lectura y almacenamiento de los datos del archivo XML ingresados por el

científico para poder llevar acabo la experimentación.

Palabras clave

Lista simple, POO, manipulación de datos, TDA's.

Abstract

In the realization of this project, we sought to solve a problem using abstract data types (TDA's), data manipulation using XML file types, dynamic memory concepts using lists and OOP.

For this purpose, a program was developed which allows the realization of experiments using samples of Martian soils, so that they can be studied by scientists by loading the information of these samples in the program through an XML file.

For this program simple lists and object oriented programming were used, which allowed the reading and storage of the XML file data entered by the scientist in order to carry out the experimentation.

Keywords

Simple list, POO, data manipulation, TDA's.

Introducción

En la realización de este proyecto y mediante la solución de este mismo su busco poder implementar deferentes temas tales como la manipulación de datos a través de archivos xml, conceptos y uso de memoria dinámica a través del uso de listas enlazadas, en caso de este proyecto la utilización de listas simples enlazadas, tipos de datos abstractos y programación orientado a objeto para el buen funcionamiento y estructuración de dicho proyecto.

Además, también se llevó acabo la utilización del software Graphviz para la realización de reportes del funcionamiento correcto del programa.

Tipos de datos abstracto.

Mediante la utilización e implementación de los diferentes tipos de datos de abstractos se puede realizar diferentes acciones que pueden mejorar

La eficiencia de un programa, ya que por medio de estos podemos mejorar e implementar la reutilización de código, la abstracción y encapsulamiento todo esto con el fin de poder tener una buena flexibilidad en el programa.

Y mediante la implementación de los diferentes tipos de datos abstractos tales como, las listas, colas y pilas se pueden crear diferentes funciones que se adapten al requerimiento del programa, es por esto que debemos de aprovechar las implementaciones de estos tipos de datos.

Estos son algunas funcionalidades que nos pueden proporcionar estos tipos de datos en este caso las listas, el cual pueden ser de diferentes tipos, tales como las listas simples, las listas circulares, las listas dobles, en el caso del proyecto construido se implementó únicamente el tipo de lista simple.

Funcionalidad de la lista simple:

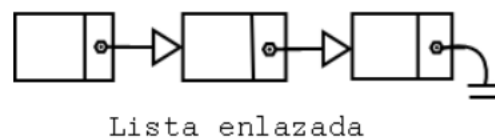
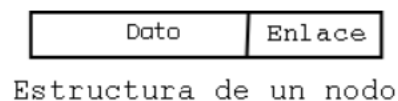


Figura 1. Elaboración propia 2023.

En el caso del proyecto realizado, y mediante la implementación de la lista simple en código mediante el lenguaje Python quedaría de la siguiente manera.

```
class Lista_simple:
    def __init__(self):
        self.primerono = None
        self.ultimono = None

    def insertar(self, dato):
        nodo = Nodo(dato)
        if self.primerono is None:
            self.primerono = nodo
            self.ultimono = self.primerono
        else:
            self.ultimono.siguiente = nodo
            self.ultimono = nodo

    def imprimir(self):
        actual = self.primerono
        while actual:
            print(actual.dato.fila)
            print(actual.dato.columna)
            print(actual.dato.codigo_organismo)
            actual = actual.siguiente

    def recorrer(self):
        actual = self.primerono
        while actual:
            yield actual.dato
            actual = actual.siguiente
```

Figura 2. Elaboración propia 2023.

Además de los tipos de datos abstractos también es importante la implementación de la programación orientada a objetos que es uno de los pilares de la programación en estos momentos, ya que su implementación facilita la elaboración de diferentes programas ya que por medio de este paradigma se puede modelar problemas complejos de manera clara y estructurada.

La implementación del paradigma POO en este proyecto mediante las diferentes clases fue de la siguiente manera.

Implementación de la una clase nodo el cual permite acceder a las diferentes funciones de la clase lista simple.

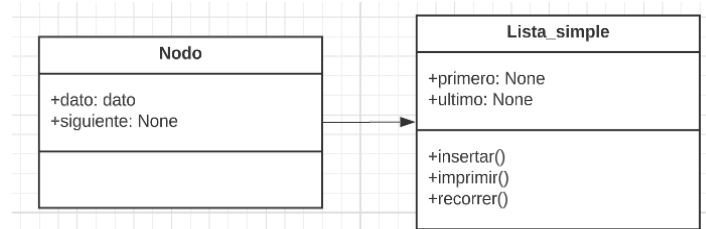


Figura 3. Elaboración propia 2023.

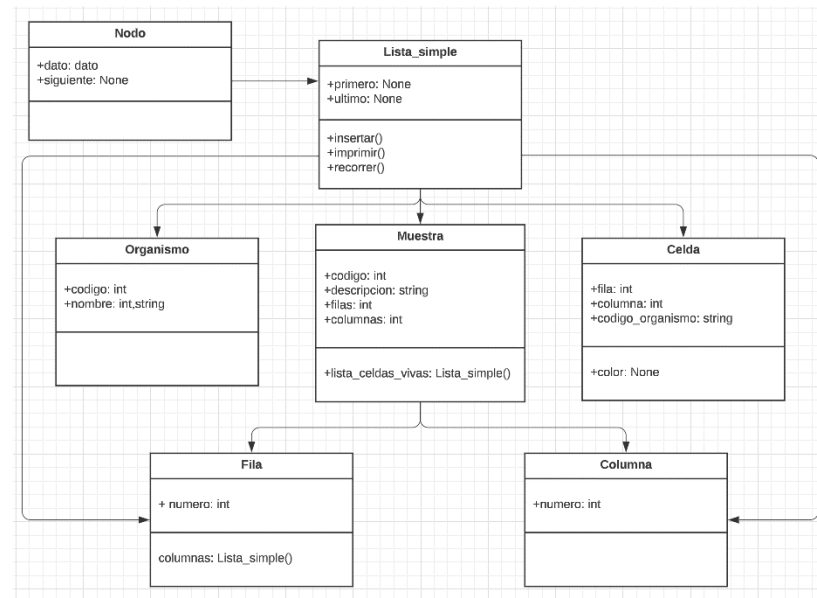


Figura 4. Elaboración propia 2023.

Conclusiones.

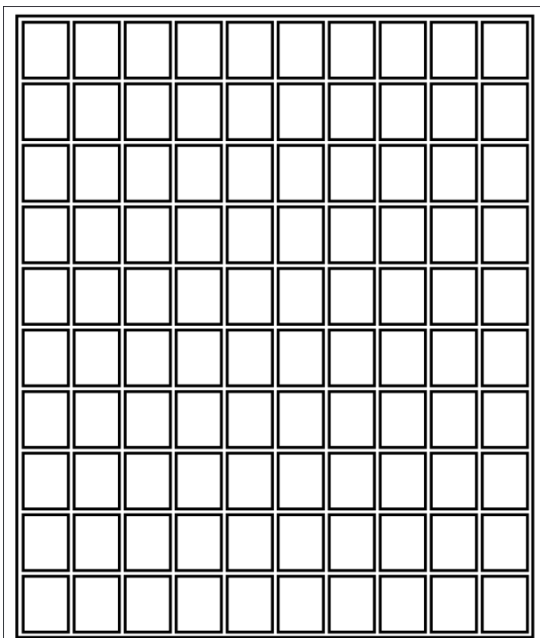
1. Mediante la implementación de los diferentes tipos de datos abstractos se puede mejorar la eficiencia de un

programa, además que también en base a estos datos se puede implementar la reutilización de código.

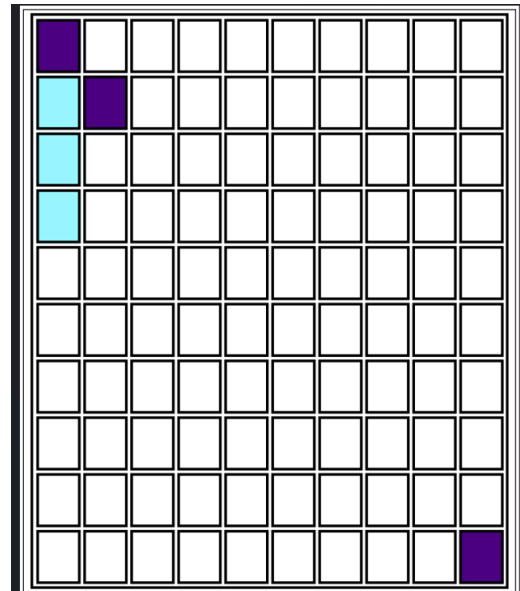
2. El paradigma de la programación orientada a objetos es una de los métodos más utilizados para para modelar problemas complejos y facilitar la creación de código.

Anexos.

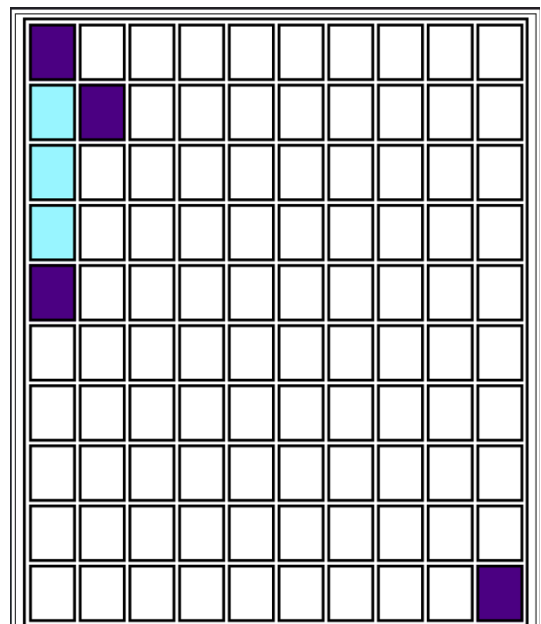
Tablero inicio.



Tablero con muestras.

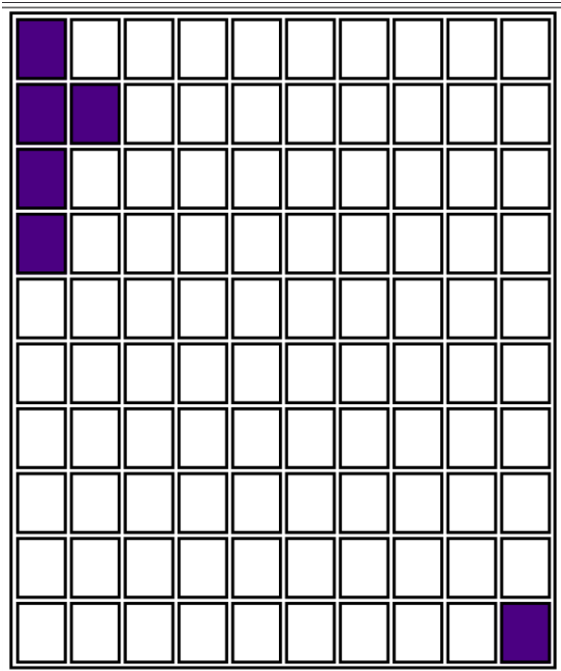


Tablero con la muestra del experimento.

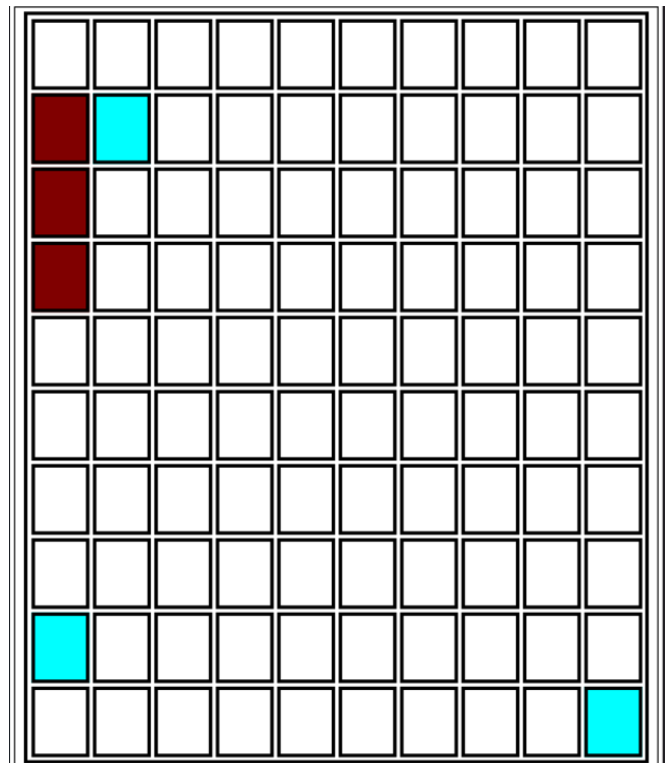
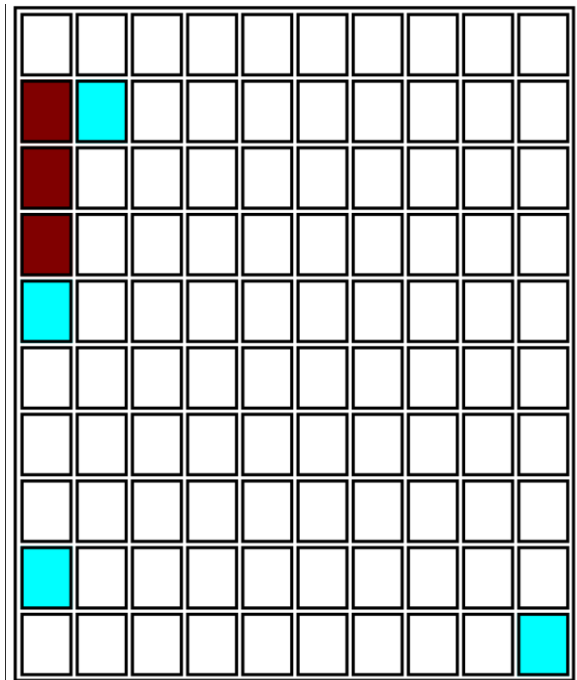


Tablero después de haber realizado el experimento.

Prospera.



No prospera.



Referencias bibliográficas.

Craig Larman; Introducción al análisis y diseño orientado a objetos. Prentice Hall.

Fundamentos de programación, Algoritmos, Estructuras de Datos y Objetos, Luis Joyanes Aguillarr, Cuarta Edición, McGraw-Hill.

Python para informáticos Versión 2.7.2 Charles Severance.