

---

## MANEJO DE E INTERCAMBIO DE PISOS PARA LA EMPRESA PISOS ARTESANALES S.A

---

202001471 – Luis René Yaquian Recinos

### Resumen

El programa consiste en optimizar el proceso de mover azulejos dependiendo del costo de movimiento de estos, dicho programa tendrá una vigencia a largo plazo, pero se podrán hacer modificaciones para que cada vez sea más optimo y se podrán implementar y tendrá un contexto tanto nacional como internacional ya que en cualquier empresa se podrá implementar sin importar su zona geográfica.

En nivel técnico aún se puede implementar nuevas funcionalidades ya que se podrían mover los azulejos de manera diagonal y hacerlo en un menor tiempo, todos estos cambios se pueden realizar dependiendo de las necesidades para poder adaptarlo de la mejor manera, tendrá un impacto en la empresa ya que solucionará el problema de la distribución de los pisos y se distribuirá el trabajo de una manera más optima que al final esa es la meta de cada trabajo.

### Palabras clave

Optimización, Movimiento, Azulejos, Mejora

### Abstract

*The program consists of optimizing the process of moving tiles depending on the cost of moving them, the program will have a long-term validity, but modifications can be made so that it is more and more optimal and can be implemented and will have a national and international context. international since it can be implemented in any company regardless of its geographical area.*

*On a technical level, new functionalities can still be implemented and the tiles could be moved diagonally and done in less time, all these changes can be made depending on the needs to be able to adapt it in the best way, the program will have an impact on the company since it will solve the problem of the distribution of the floors and the work will be distributed in a more optimal way that in the end that is the goal of each work.*

### Keywords

*Optimization, Movement, Tiles, Improvement*

## Introducción

Se plantea el problema que se tienen distintos patrones con R filas y C columnas, También se tiene un costo para volteo y un costo para desplazar el azulejo. Tienen colores los cuales son blanco y negro.

Debido a esta problemática se crea el programa para solucionar el problema seleccionando lo deseado en el menú y se deberá de mostrar los pasos que se realizan para la ejecución del programa y finalmente se mostrará de manera gráfica el patrón final.

## Desarrollo del tema

Para la realización de la solución de este problema fue la implementación de nodos los cuales se definen como “un punto de intersección, conexión o unión de varios elementos que confluyen en el mismo lugar”. Y también se utilizaron listas doblemente enlazadas las que tienen una definición: “La lista doblemente enlazada es una estructura de datos que consisten en conjunto de nodos enlazados secuencialmente”.

Se implementó una lista doblemente enlazada para poder navegar entre los nodos y así poder acceder a los datos deseados. Se implementó el .siguiente y el .anterior para poder recorrer los datos, al inicio y al final se debe de tener un None ya que los datos del nodo al inicio no existe un dato anterior y al finalizar de recorrer el nodo no existirá nada ya que es una lista doblemente enlazada.

Para la apertura del archivo .xml se debe de importar xml.etree.ElementTree, para poder acceder al sistema se importó os y lo más importante para poder crear grafos se debe de importar Graphviz.

Se crearon distintas clases: Por ejemplo, se creó la clase piso que era la que iba a almacenar toda la información que iba a ingresar por medio del archivo .xml.

Se creó un tmp para poder utilizarlo cuando se desea obtener la información de este, al principio el tmp nos ayudará a extraer los datos y así poder ir intercambiando en las clases, agregando los datos en nodos y poder crear sub-nodos para poder acceder a los códigos de colores.

Se crearon métodos para poder extraer los datos y poder implementarlos como una matriz y así lograr recorrerla para poder extraer la posición y así poder cambiarla como es deseada.

Se debe de tener en cuenta que cada movimiento tiene un costo y se debe de buscar optimizar el patrón para que se realicen los menos movimientos posibles y siempre buscando el menor costo para que sea rentable utilizar el programa.

Para poder generar el cambio de patrones se debe de analizar los patrones al inicio de cada archivo y luego debemos de acceder al patrón para poder intercambiarlos como desea el usuario. Se creó un bubbleSort para los códigos y al recorrerla podemos ordenarla de manera alfabética.

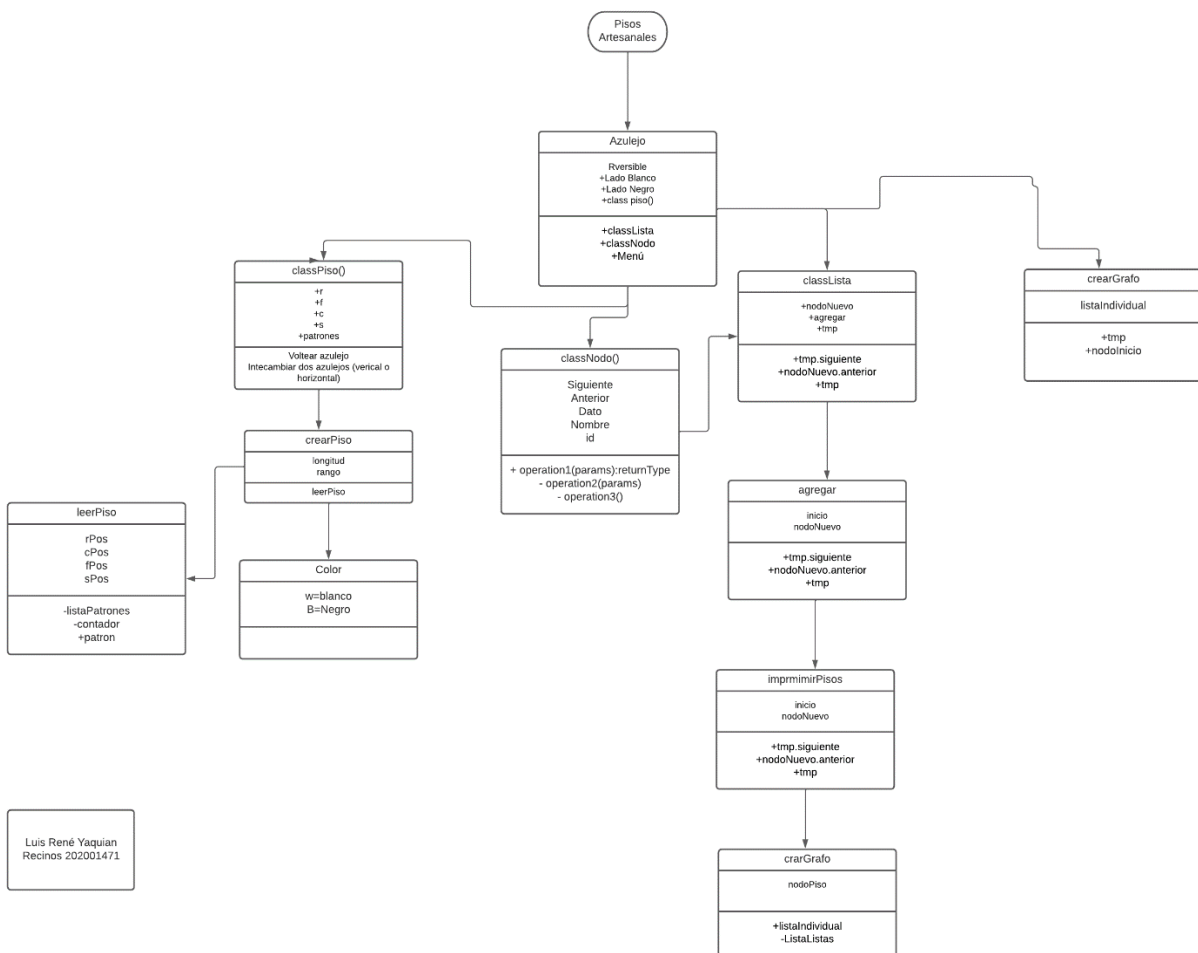


Figura 1. Diagrama de clases del programa.

Fuente: elaboración propia

Autores: Exposito Lopez Daniel, Abraham García  
Soto, Martin Gomez Antonio Jose| Sitio:  
Universidad de Granada| Título: LISTAS  
DOBLEMENTE ENLAZADAS| Fecha:| URL:  
<https://ccia.ugr.es/~jfv/ed1/tedi/cdrom/docs/ldoble.html>

## Conclusiones

Se deben de almacenar los datos en nodos para poder acceder a los datos, en el análisis de este caso a mi parecer lo más optimo fue la utilización de listas doblemente enlazadas para poder acceder y poder desplazarse en los datos para poder accederlo cuando se necesite.

Se creó una clase Piso para poder obtener los datos y se creó una lista que almacena nodos que contienen otras listas de caracteres. Para poder intercambiar los patrones se implementó los datos como una matriz con posiciones para poder desplazar o intercambiar los datos, se debe de tomar en cuenta en número de filas y columnas que se obtengan del archivo .xml.

Se pudo crear una lista ortogonal para poder extraer los datos de manera más fácil y para los movimientos, pero se necesita de un mayor conocimiento acerca de listas para poder implementar ese método.

## Referencias bibliográficas

Autor: Ramiro de la Vega| Sitio: PHAROS | Título:  
Lista doblemente enlazada con ejemplos de  
Python | Fecha: Dic. 2020 | URL:  
<https://pharos.sh/lista-doblemente-enlazada-con-ejemplos-de-python/>

Autor: Victoria Bemibre | Sitio: Definición ABC |  
Título: Nodo | Fecha: Ene. 2009 | URL:  
<https://www.definicionabc.com/tecnologia/nodo.php>