
PISOS ARTESANALES SA

202010770 – Luis Mariano Moreira García

Resumen

El programa en cuestión es capaz de cargar información que se extrae directamente de un archivo con la extensión mundialmente conocida como XML.

El programa logra funcionar de manera totalmente eficiente ya que funciona a través de listas creadas a partir de objetos y así lograr eliminar espacio en memoria que no se utiliza por el garbage collector.

Por medio de los XML se crean los objetos y así reproducirlos directamente a una lista de pisos para que el usuario en cuestión logre poder interactuar directamente con ellos.

El usuario será capaz de cargar listas de pisos, poder hacer que un piso tome la forma de otro piso de la misma dimensión desde la manera más económica posible.

Palabras clave

XML, lista de pisos, objetos, intercambio, volteo.

Abstract

This program is able to load the information from the file with the XML extension.

The program can work in the most efficient way this is because it works with lists created with python to clean space in the memory with help of the garbage collector.

With the XML the objects are generated and are loaded directly to the floor list to make the interaction between the program and the user.

The user can load lists of floors, the floors can take the form of another floor of the same dimension in the most economic way.

Keywords

XML, floor of lists, objects, swap, flip.

Introducción

Brindar un programa que ayuda directamente a los usuarios que compren directamente a la empresa de pisos artesanales, el programa puede cargar un archivo de tipo XML para que el usuario pueda interactuar de una manera totalmente eficiente con el programa y esto se logra ya que el programa funciona a través de una lista de objetos y así hacer de un uso correcto de los espacios en memoria.

El usuario será capaz de cargar las listas por medio de un archivo XML, una vez cargadas el usuario puede seleccionar una lista con otra lista que tiene varios patrones en su interior y así poder interactuar con el usuario.

El programa puede mostrar los patrones por medio de graphviz y también describir una serie de movimientos a seguir para poder realizar los cambios hacia un nuevo piso de la manera menos costosa posible para que para la empresa no tenga que cubrir costos de más a la hora de realizar el programa.

Desarrollo del tema

Como se ha mencionado anteriormente el programa funciona a través de listas enlazadas que interiormente tienen la forma de una matriz a pesar de que en la ejecución no se utiliza y se podría decir que es una lista doblemente enlazada.

Debido a que no se utilizaron las listas integradas directamente desde Python esto complico más el desarrollo del programa ya que se utilizaron varias clases de nodo y lista.

A continuación se enumeran las listas que se utilizaron para poder realizar el programa

- Lista de pisos
- Lista de patrones
- Lista de matriz
- Lista de textos.

Lista de pisos, muestra de una manera ordenada lo que es la lista de todos los pisos disponibles, tiene una estructura simple ya que los nodos únicamente se conectan entre si por medio de un apuntador siguiente.

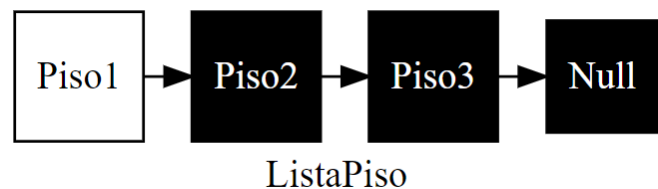


Figura 1. Estructura de la lista de pisos

Fuente: elaboración propia, 2021

Todas las listas funcionan de la misma manera con excepción de la lista matriz, la cual cuenta con la siguiente estructura:

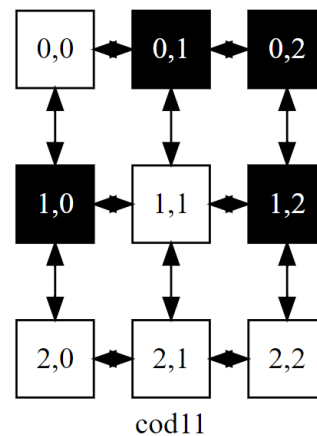


Figura 2. Estructura de la lista de matrices

Fuente: elaboración propia, 2021

El algoritmo en cuestión para realizar los cambios es totalmente eficiente cuando se trata de funciones menores o iguales a 2×2 , cuando es traspasado logra cambiar al patrón deseado, pero lo realiza con una cantidad de movimientos extras.

El algoritmo para encontrar el menor precio posible funciona de la siguiente manera:

1. Se recorre la matriz
2. Se compara si el elemento es igual al de la matriz objetivo.
3. En el caso de ser distintos ver si el valor de volteo es dos veces menor que el de intercambio (Esto ya que el intercambio involucra dos movimientos al mismo tiempo)
4. Si se satisface esta condición se comprobará que el siguiente a la derecha o abajo sea igual a la misma posición de la otra matriz, esto para realizar el cambio.
5. De no satisfacer las condiciones 3 y 4 se realizará un volteo de manera automática.

El algoritmo para realizar los volteos funciona de la siguiente manera:

1. Guardar el color a cambiar en una variable auxiliar.
2. Por medio de un set tomar el valor de color de la misma posición, pero en la otra matriz.
3. Guardar en la lista de instrucciones los cambios realizados.

El algoritmo para realizar los intercambios es el siguiente:

Existen dos mismas funciones, una para realizar intercambio en $x+1$ y otra en $y+1$.

1. Guardar el color a cambiar.

2. Que la posición actual tome el valor siguiente.
3. Que el siguiente tome el valor del anterior.
4. Registrar el movimiento en la lista de textos.

A pesar de que la lista este completamente enlazada, el programa no hace función o uso a esta opción ya que para este proyecto no fue necesario.

Conclusiones

El buen uso de las clases permite desarrollar programas de una manera más eficiente, ya que por medio de las listas enlazadas es posible desarrollar programas que sean óptimos para el uso cotidiano.

Por medio de las listas uno es capaz de poder realizar los métodos necesarios para llevar a la acción y así poder llevar un flujo más eficiente del programa.

Referencias bibliográficas

1. Calzadilla, J. C. F., Herrera, A. N., & Delfino, E. La enseñanza de los arrays estáticos, dinámicos y listas enlazadas¿ cuál usar? Análisis de códigos The teaching of static arrays, dynamics and linked lists. What to use? Code analysis.
2. Challenger-Pérez, I., Díaz-Ricardo, Y., & Becerra-García, R. A. (2014). El lenguaje de programación Python. *Ciencias Holguín*, 20(2), 1-13.
3. Fernández, A. (2013). *Python 3 al descubierto*. Alfaomega Grupo Editor.
4. Legarreta, I. V. (1995). *Diseño de un array bidimensional dinámico implementado mediante listas enlazadas y árboles AVL* (Doctoral dissertation, Universidad de Deusto).
5. Van Rossum, G. (2007, June). Python Programming language. In *USENIX annual technical conference* (Vol. 41, No. 1, pp. 1-36).