
Proyecto 1: Optimización del costo sobre cambio de pisos

202000544 – Roberto Carlos Gómez Donis

Resumen

Se realizó la creación de un programa para la creación y cambio de pisos mediante el lenguaje Python implementando TDA's (Tipo de datos abstractos) para el manejo de listas enlazadas y doblemente enlazadas, POO (programación orientada a objetos) para lograr guardar información en su respectiva clase. La lectura de archivo xml para extraer información de entrada con la ayuda de la herramienta graphviz para obtener gráficamente los nodos que se crean al momento de ingresar el archivo.

Este software es de gran ayuda para las empresas de pisos, tanto por la optimización del costo para el cambio de cada piso que se quiera hacer como tener gráficamente los pisos de como va a quedar.

Palabras clave

TDA, XML, NODO, LISTA ENLAZADA.

Abstract

The creation of a program for the creation and change of floors was carried out using the Python language, implementing TDA's (Abstract Data Type) for the management of linked and double linked lists, OOP (object-oriented programming) to achieve saving information in its respective class. The reading of the xml file to extract input information with the help of the graphviz tool to graphically obtain the nodes that are created when entering the file.

This software is of great help for flooring companies, both for optimizing the cost of changing each floor that you want to do, and graphically showing how the floors are going to look.

Keywords

TDA, XML, NODE, LINKED LIST.

Introducción

La utilización de archivos XML en la actualidad es muy común entre los softwares, ya que se facilita el manejo de información en los archivos. Al momento de analizar se utiliza TDA para guardar la información de una manera óptima para procesar y lograr graficar los nodos en pdf.

Desarrollo del tema

El proyecto se divide en 3 partes, para empezar es la lectura del archivo XML y más importante para que funcione el programa, ya que sin este no es funcional el programa. Este contendrá el nombre de los pisos que se quieren tener y sus patrones (los que se graficarán), contendrá filas, columnas, costo de cambio y delizar piso. Este archivo podrá contener n cantidad de patrones. Una vez leído el archivo se hace el uso de TDA, en este caso es una lista simple enlazada, para guardar el nombre, fila, columna, costo de delizar y costo de cambiar, sin embargo en esta lista se instancian 2 listas doblemente enlazada, para obtener el código de cada patron y los patrones por separado.

a. Nodo:

En informática y en telecomunicaciones, de forma muy general, un nodo es un punto de intersección, conexión o unión de varios elementos que confluyen en el mismo lugar.

b. Lista Simplemente enlazada:

Es una lista enlazada de nodos, donde cada nodo tiene un único campo de enlace. Una

variable de referencia contiene una referencia al primer nodo.

c. Lista doblemente enlazada:

Es una lista enlazada que a diferencia de la simple esta tiene dos enlaces, donde hace referencia al anterior y al siguiente.

d. Graphviz:

Graphviz (Graph Visualization) es un conjunto de herramientas de software para el diseño de diagramas definido en el lenguaje.

Como primera función es leer el archivo XML y cargarlo a las listas para poder manipular los datos escritos en cada archivo.

```
<pisosArtesanales>
  <piso nombre="Aejemplo01">
    <R> 2 </R>
    <C> 4 </C>
    <F> 1 </F>
    <S> 1 </S>
    <patrones>
      <patron codigo="hcod11">
        WBWBWWWB
      </patron>
      <patron codigo="acod12">
        BWBWWWWW
      </patron>
    </patrones>
  </piso>
```

Figura 1. Título o descripción breve de la figura.

Fuente: Elaboración Propia 2022

Luego de obtener la información se crean los nodos, dichos nodos crean un archivo con extensión “.txt”, aunque también se puede crear en diferentes extensiones, pero la que se utilizó para el software fue “.txt”.

```
digraph G {
  rankdir = LR
  subgraph hcodi1{
    1[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    2[label="B",color = "black",fontcolor = "white",fillcolor="black",style="filled",shape="box"];
    3[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    4[label="B",color = "black",fontcolor = "white",fillcolor="black",style="filled",shape="box"];
    5[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    6[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    7[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    8[label="B",color = "black",fontcolor = "white",fillcolor="black",style="filled",shape="box"];
  }
  subgraph acodi2{
    9[label="B",color = "black",fontcolor = "white",fillcolor="black",style="filled",shape="box"];
    10[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    11[label="B",color = "black",fontcolor = "white",fillcolor="black",style="filled",shape="box"];
    12[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    13[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    14[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    15[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
    16[label="W",color = "black",fontcolor = "black",fillcolor="white",style="filled",shape="box"];
  }
  subgraph cluster_0{
    label="Patron Inicial"
    rankdir=TB
    1->2;
    2->3;
    3->4;
    5->6;
    6->7;
    7->8;
  }
  subgraph cluster_1{
    label="Patron Final"
    rankdir=TB
  }
}
```

Figura 2. Título o descripción breve de la figura.

Fuente: Elaboración Propia 2022

Luego que se crea el archivo “.txt” se convierte en extension “.pdf” para graficar los nodos, todo con ayuda de la herramienta de graphviz, sin embargo, hay que tener en cuenta que para que se grafique bien, los nodos deben de ir enlazados de forma correcta, ya que se puede dar la ocasión que se enlacen de forma incorrecta y quede de manera esperada.

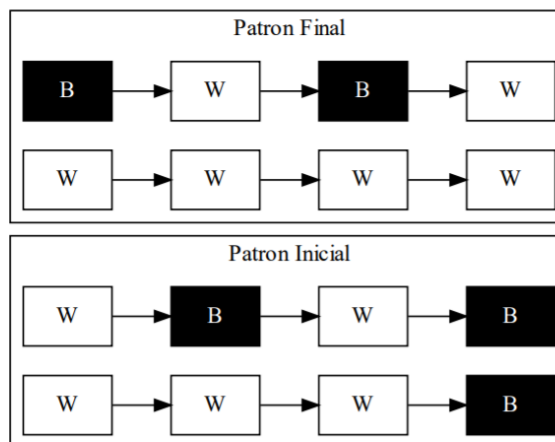


Figura 3. Título o descripción breve de la figura.

Fuente: Elaboración Propia 2022

Conclusiones

- Los Tipos de Datos Abstratos(TDA) tienen una funcionalidad vital a la hora de trabajar con memoria dinámica ya que optimiza el uso de la memoria.
- La herramienta Graphviz es eficiente a la hora de graficar grafos y TDA's
- Los Archivos xml son vitales a la hora de transferir datos.

Referencias bibliográficas

- Teh, P. S., Zhang, N., Teoh, A. B. J., & Chen, K. (2016). TDAS: a touch dynamics based multifactor authentication solution for mobile devices. International Journal of Pervasive Computing and Communications.

- Gansner, E. R. (2009). Drawing graphs with Graphviz. Technical Report, Technical Report

Diagrama de clase

