
Creación de patrones en azulejos utilizando un robot especializado, para Pisos Artesanales S.A

201903791 – Kevin Gerardo Ruíz Yupe

Resumen

En este ensayo se explica el desarrollo del analizador de patrones y cómo se implementan los Tipos de datos Abstractos para la optimización del precio en el cambio de patrones.

Se utilizó un algoritmo propio adaptado al problema de costo mínimo. Para una mejor experiencia para el usuario se utilizó GraphViz, esta herramienta permitió graficar cada patrón inicial y final según el elegido por el usuario.

Palabras clave

Matriz Dispersa

Algoritmo

Lenguaje de Marcado Extensible

Herramienta GraphViz

Abstract

This essay explains the development of the pattern analyzer and how Abstract Data Types are implemented for price optimization on pattern change.

A proprietary algorithm adapted to the minimum cost problem was obtained. For a better user experience Gra

phViz was reduced, this tool was able to graph each initial and final pattern according to the one chosen by the user.

Keywords

Sparse Matrix

Algorithm

XML (Extensible Markup Language)

GraphViz

Introducción

La empresa “Pisos Artesanales, S.A.” ha construido un azulejo especial con el que puede

crear pisos con distintos patrones. Cada piso consiste en una matriz de R filas y C

columnas de azulejos cuadrados. Cada azulejo es reversible, un lado es blanco y el otro es

negro, para poder crear patrones diversos. Además, la empresa garantiza que, para los

pisos ya instalados, podrá cambiar el patrón original por un nuevo patrón que el cliente

desea sin necesidad de comprar nuevos azulejos.

La empresa ha comprado un robot especializado capaz de colocar los pisos de dimensiones

$R \times C$ con cualquier patrón, combinando azulejos del lado blanco con azulejos del lado

negro.

Desarrollo del tema

Debido a que la empresa “Pisos Artesanales, S.A.” posee una cuota fija pactada por contrato con los clientes que han colocado los pisos que ofrecen, se le ha solicitado realizar un programa que garantice que, al modificar un patrón en un piso existente, el costo de hacer esta modificación sea el mínimo posible para optimizar el uso del robot especializado adquirido para este fin.

A continuación, se muestran conceptos que se abordaron para la solución de este problema:

a. Tipos de datos abstractos:

Se define como un modelo que define valores y las operaciones que se pueden realizar sobre ellos. Y se denomina abstracto ya que la intención es que quien

lo utiliza, no necesita conocer los detalles de la representación interna o bien el cómo están implementadas las operaciones.

Es por esto una práctica que nos provee un grado de abstracción que permite desacoplar al código que usa un TDA de aquel código que lo implementa.

b. Listas Simples Enlazadas:

Una lista enlazada o encadenada es un conjunto de elementos en los que cada elemento contiene la posición

—o dirección— del siguiente elemento de la lista.

Cada elemento de la lista enlazada debe tener al menos dos campos:

un campo que tiene el valor del elemento y un campo (enlace, link) que contiene la posición del siguiente elemento,

es decir, su conexión, enlace o encadenamiento. Los elementos de una lista son enlazados por medio de los campos enlaces.

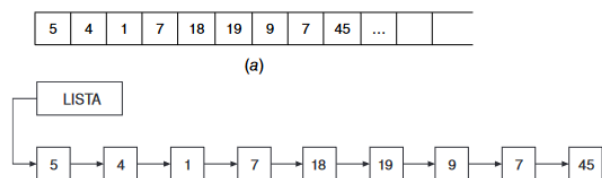


Figura 1. Ejemplo de Lista Simple Enlazada

Fuente: Capítulo 12 Estructuras Dinámicas, Joyanes.

Todas las figuras deben ir enumeradas al pie de la imagen, como se muestra en el ejemplo.

En el caso de inclusión de tablas, éstas deben pegarse en el formato de origen, conservando el modelo

c. Listas Doblemente Enlazadas:

En las listas lineales estudiadas anteriormente el recorrido de ellas sólo podía hacerse en un único sentido: de izquierda a derecha (principio a final). En numerosas ocasiones se necesita recorrer las listas en ambas direcciones.

Las listas que pueden recorrerse en ambas direcciones se denominan listas doblemente enlazadas. En estas listas cada nodo consta de la información y una referencia al próximo nodo y uno al anterior.

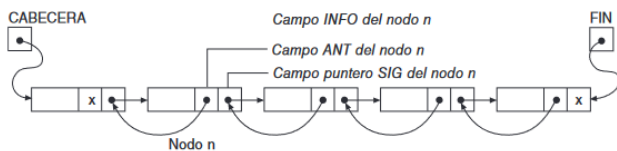


Figura 2. Ejemplo de Lista Doble Enlazada

Fuente: Capítulo 12 Estructuras Dinámicas, Joyanes.

Referencias bibliográficas

«Que es la abstracción de datos y modelos de datos». 21 de agosto de 2010. Archivado desde el original el 19 de marzo de 2011. Consultado el 21 de enero de 2020.

GraphViz, <https://graphviz.org/documentation/>

Conclusiones

Las librerías de Python para la lectura de XML facilitan la obtención de información en una aplicación, en especial cuando hay mas múltiples etiquetas. El uso de listas simples y doblemente enlazadas son fundamentales para guardar la información. Dado a que son estructuras dinámicas, es posible que en un futuro se modifique el código para ingresar pisos de forma manual y no en carga masiva.

Es claro que el algoritmo implementado tenga sus defectos, sin embargo, es simple al momento de estudiarlo y comprenderlo.

Apéndices

- Diagrama de Clases

