
PROYECTO 1

201801539 – Brenda Paola Gramajo Paniagua

Resumen

El proyecto realizado se basó en la busca de una solución al problema que se planteó, mediante la lógica y conocimientos que se han adquirido durante la clase y que han sido aplicados en el laboratorio. Este problema consistió en la elaboración de un programa que funciona como simulador de un robot para el algoritmo de ordenamiento de pisos. El programa fue desarrollado en el lenguaje de programación de Python y se utilizaron Listas Enlazadas para la creación del algoritmo y almacenamiento de datos. El problema planteado representa funciones que se pueden observar en el medio en el que vivimos, y la solución se puede utilizar como recurso de optimización en los casos similares, al tener un programa que realice tareas como estas, se mejora el entorno de trabajo y por lo tanto, la producción. La solución planteada es la muestra de conocimientos adquiridos y es preparación para los problemas que se puedan presentar en un futuro.

Palabras clave

Nodo, Algoritmo, Listas, Ordenamiento, Patrones

Abstract

The project was based on the search for a solution to the problem posed, using the logic and knowledge acquired during the class and applied in the laboratory. This problem consisted in the development of a program that works as a robot simulator for the floor ordering algorithm. The program was developed in the Python programming language and Linked Lists were used for the creation of the algorithm and data storage. The problem posed represents functions that can be observed in the environment in which we live, and the solution can be used as an optimization resource in similar cases, by having a program that performs tasks like these, the work environment is improved and therefore, the production. The proposed solution is a sample of acquired knowledge and is preparation for problems that may arise in the future.

Keywords

Node, Algorithm, Lists, Sorting, Patterns, Ordering

Introducción

El proyecto a realizar describía la solución para el problema que tenía una fábrica de pisos, que consistía en poder crear varios pisos con distintos patrones brindados por el usuario. El programa realizado es capaz de colocar los pisos de dimensiones $R \times C$ con cualquier patrón, combinando azulejos del lado blanco con azulejos del lado negro (las dimensiones dadas son leídas a partir de un archivo XML brindado por el usuario). Además, la empresa garantiza que cada piso colocado podrá cambiar el patrón, a cualquier patrón deseado. Lo solicitado es que el 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.

Desarrollo del tema

El desarrollo del proyecto consistió en dos partes importantes, la primera parte se desarrolló la lectura del archivo de entrada XML, y la extracción de los datos contenidos en el documento. Toda la segunda parte se centró en la creación de las respectivas clases para trabajar con la POO. Se creó una clase Piso que contenía los respectivos atributos, como el costo y las dimensiones del mismo. Y en este se incluía un atributo de patrones que correspondía a una lista de patrones que el usuario brindaba. Se creó también una clase "Patrones" que almacenaba los atributos

respectivos de cada patrón individual. También se implementó para la solución la creación de una clase Nodo, en la que se tenían los atributos necesarios para la creación de varios Nodos que se incluirían en la clase de Lista Enlazada. La clase de Lista Enlazada contenía las funciones necesarias para la creación de las listas, como puede ser la función insertar. Se crearon los respectivos objetos 'Piso' y 'Patrón' extrayendo los datos del archivo XML y se almacenaron en el respectivo Nodo de la Lista Enlazada. Con esto se pudieron crear funciones como lo son el ordenamiento de la lista por nombre y así mismo, el ordenamiento de los patrones en orden ascendente. Para la función de graficar, se utilizó la librería de Python llamada Graphviz. Esta se implementó como una solución práctica para graficar los pisos con cada patrón respectivo. El proyecto incluye un menú en consola agradable al usuario. Este tiene funciones como lo son:

1. Cargar Archivo:

Función que permite cargar al usuario el archivo XML solicitando la ruta del archivo.

2. Elegir Piso y Patrón inicial:

Esta función pide al usuario ingresar el nombre del piso por el cual quiere iniciar a graficar, y escoger el patrón a graficar.

3. Graficar Patrón:

Función que muestra la gráfica del patrón escogido anteriormente.

4. Elegir Nuevo Patrón:

Permite al usuario escoger un nuevo patrón a graficar.

5. Mostrar todos los Patrones:

Muestra en consola todos los pisos y patrones cargados ordenándolos en forma ascendente por su nombre y ordenando también los patrones.

6. Salir:

Termina la ejecución del programa.

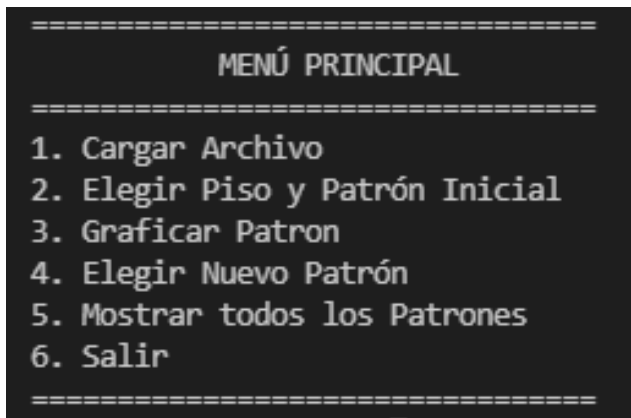


Figura 1. Menú en consola.

Fuente: elaboración propia.

Todo el programa fue desarrollado en consola y el programa se desarrollo con base de Programación Orientada a Objetos y la aplicación de las Listas Enlazadas.

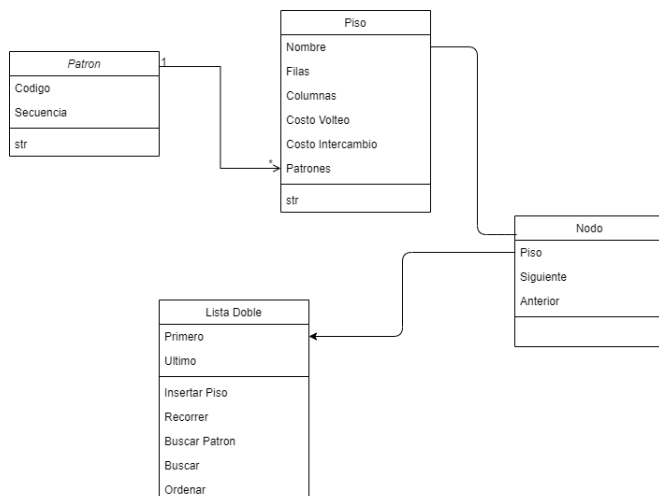


Figura 11. Diagrama de Clases.

Fuente: elaboración propia.

Conclusiones

La programación orientada a objetos se basa en el concepto de crear un modelo del problema de destino en sus programas. La programación orientada a objetos disminuye los errores y promociona la reutilización del código. Python es un lenguaje orientado a objetos.

Una lista enlazada es una colección de nodos, cada uno compuesto por una referencia y un valor. Los nodos se unen en una secuencia utilizando sus referencias.

Los archivos XML implementan una forma de almacenar datos de manera legible y jerárquica y pueden albergar diferente tipos de datos.