|  |
| --- |
| **Simulador Ensamblaje de Productos** |
| **Carnet - 201403946** |

**Resumen**

La matriz ortogonal es una estructura de datos en donde se implementa una tabla con memoria dinámica. En este sistema los componentes forman un patrón regular, en el cual se agrupan a lo largo de un número de diagonales.

Los TDA son estructuras a los que se añade propiedades de reutilización y de compartición de códigos y la mejor solución al problema puede ser usarla. Este método al ser bien implementado tiene la ventaja en la agregación de nuevos datos a la matriz.

La estructura antes menciona se aplica en el funcionamiento del algoritmo r2e2, el cual consiste en encontrar un camino que gaste la menor cantidad de combustible. Se ingresa un mapa en forma de matriz ortogonal. El método de aplicar la estructura se emplea en una lista doble, posiciones y combustible, así se organiza los datos para que puedan ser utilizados. El método de búsqueda depende de la forma de moverse por la lista doble.

**Palabras clave**

Tipo de datos abstractos, matriz ortogonal, memoria dinámica, lista doblemente enlazada, estructura.

***Abstract***

*The orthogonal array is a data structure where a table with dynamic memory is implemented. In this system the components form a regular pattern, in which they are grouped along a number of diagonals.*

*TDAs are structures to which reusability and code sharing properties are added and the best solution to the problem may be to use it. This method when well implemented has the advantage in adding new data to the matrix.*

*The structure mentioned above is applied in the operation of the r2e2 algorithm, which consists of finding a path that spends the least amount of fuel. A map is entered in the form of an orthogonal matrix. The method of applying the structure is used in a double list, positions and fuel, thus organizing the data so that it can be used.The search method depends on the way of moving through the double list.*

***Keywords***

*Abstract data type, orthogonal matrix, dynamic memory, doubly linked list, structure.*

**Introducción**

Las matrices ortogonales en la computación son utilizadas en la optimización a gran escala, otras de las áreas de interés en donde se pueden aplicar la representación son en la teoría de grafos y de redes, métodos numéricos, entre otros.

En las operaciones que se realiza en algoritmo del robot r2e2 se utiliza las listas doblemente enlazadas para implementar una matriz dispersa y así simular el tablero. El almacenamiento de cada dato serán piezas de un mapa que se irán colocando a lo largo de la tabla.

La lista doble enlazada contiene como atributo dos punteros de forma que uno apunta al siguiente y el otro al predecesor, lo cual lo hace un TDA dinámico lineal y se aplica enlazándolos entre sí de manera que forma una matriz dispersa.

Dado que este tipo de estructuras ocurren se desarrolla distintos métodos que permiten la inserción de datos, intentando que sea lo más eficiente en la computadora y su almacenamiento en memoria.

**Desarrollo del tema**

1. **Abstracción**

La abstracción es un proceso mental mediante el cual el ser humano tiene la capacidad de extraer los rasgos esenciales de “algo” para representarlos por medio de un lenguaje gráfico o escrito.

1. **Estructura de datos**

Una estructura de datos, en general se puede definir como cualquier colección o grupo de datos organizados de tal forma que tengan asociados un conjunto de operaciones para poder manipularlos.

Las estructuras de datos se implementan a través de los lenguajes de programación y son un modelo que se caracteriza por permitir el almacenamiento y utilizar una determinada organización de datos.

1. **Estructura de datos dinámicos**

Son aquellas en las que su ocupación en memoria puede aumentar o disminuir durante el tiempo de ejecución de un programa. A su vez las estructuras de datos dinámicos se pueden clasificar en lineales y no lineales:

Estructuras lineales. Son aquellas en las que se definen secuencias como conjuntos de elementos entro los que se establece una relación de predecesor y sucesor. Las estructuras de datos basadas en este concepto se diferencian por las operaciones de acceso a los elementos y manipulación de las estructuras. Existen tres estructuras lineales especialmente importantes en las que se encuentran las listas

Estructuras no lineales. Son aquellas en las que no existe una relación de adyacencia entre sus elementos, es decir, un elemento puede estar relacionado con cero, uno o más elementos. Existen dos estructuras no lineales especialmente importantes: los árboles y los grafos

1. **Listas Dobles**

Una lista doble es aquella en que sus nodos se encuentran encadenados por dos apuntadores, es decir, cada nodo apunta al siguiente nodo de la lista, así como al nodo que le antecede en la lista. Una lista doble puede ser implementada como lineal o circular. En una lista lineal, el apuntador siguiente del último nodo y el apuntador anterior del primer nodo apunta hacia un valor nulo, mientras que en una lista circular el apuntador siguiente del último nodo apunta hacia el primer nodo de la lista y el apuntador anterior del primer nodo apunta hacia el ultimo nodo de la lista.

Diagrama

Descripción generada automáticamente

Lista doble lineal debe contener como mínimo tres campos: uno para almacenar la información y los otros dos para guardar la dirección de memoria del nodo antecesor y sucesor.

Texto

Descripción generada automáticamente

1. **Listas ortogonales**

Es aquella en la que sus nodos se encuentran encadenados por cuatro apuntadores, es decir, cada nodo se encuentra doblemente enlazado en forma horizontal, y cada nodo se encuentra doblemente enlazado en forma vertical. Esta lista puede ser implementada como lineal o circular. Este tipo de listas se puede utilizar para representar matrices.

Diagrama

Descripción generada automáticamente

El nodo de esta lista debe contener como mínimo cinco campos: uno para almacenar la información y cuatro para guardar la dirección de memoria hacia el siguiente, anterior, arriba y abajo.

Diagrama

Descripción generada automáticamente

1. **Operaciones Básicas**

En una lista se pueden efectuar operaciones por medio de algoritmos que se deben desarrollar de acuerdo con el tipo de lista. Algunas de las operaciones básicas son

* **Recorrido.** Esta operación consiste en visitar todos los nodos que forman parte de una lista. Para recorrer todos los nodos de la lista es necesario posicionarse en el primer nodo de la lista y después avanzar hacia el apuntador siguiente hasta encontrar el final.
* **Inserción.** Esta operación consiste en agregar un nuevo nodo a una lista. La ubicación del nuevo nodo puede ser al inicio, al final o en cualquier posición dentro de la lista
* **Borrado.** Esta operación consiste en eliminar un nodo de la lista y redireccionar los apuntadores al nodo antecesor y al sucesor. El borrado también se aplica al primer nodo de la lista, como al último y una posición intermedia
* **Búsqueda.** Esta operación consiste en recorrer todos los nodos de la lista desde el primer nodo para ir comparando el valor de cada nodo con el valor que se está buscando hasta encontrar el nodo con el valor indicado o encontrar el fin de la lista.

1. **Ventajas y Desventajas**

Las listas son estructuras de datos que son dinámicas, esto significa que adquieren espacio y liberan espacio a medida que se necesita. Pueden definirse estructuras más complejas a partir de las listas.

Sin embargo, hay una advertencia. Como regla general siempre hay que tener cuidado al manejar direcciones de espacios de memoria, porque es posible que se acceda a una localidad de memoria de la cual no se desea cambiar su contenido.

1. **Implementación en Python**

Como parte del documento se presenta una implementación de una lista doble simulando un tablero el cual es una de las diferentes soluciones que puede haber para la estructura.

La implementación se realizó en lenguaje Python en su versión 3.9.5 corriendo en su sistema operativo Windows 10 (64 bits)

Clase Nodo

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Método insertar Final

Texto

Descripción generada automáticamente

**Conclusiones**

Al implementar tipos de datos abstractos bajo el concepto de programación los nodos son clases y dependiendo del tipo de lista se aplica una o varias clases de sus operaciones.

Al utilizar estructuras secuenciales en Python se observa cómo funciona los apuntadores, ya que son secuenciales dependemos de estos.

Se utiliza dos tipos de nodos para implementar la lista doble que simulara el mapa, el primer nodo guarda los datos de un terreno a este nodo se guarda una lista donde cada nodo esta ordenada por filas.

**Referencias bibliográficas**

Monica Carreño, Andres Sandoval, (2012). *Listas.* VABCS – Mexico.

Joyanes Aguilar, Luis (2007). Estructuras de datos, McGraw-Hill, España.

Joyanes, L., Zahonero, (2008). Estructuras de datos en Jave, McGraw Hill.

Apéndice

Diagrama de Clases

Diagrama

Descripción generada automáticamente

Salida Estado de Tablero

Patrón de fondo

Descripción generada automáticamente