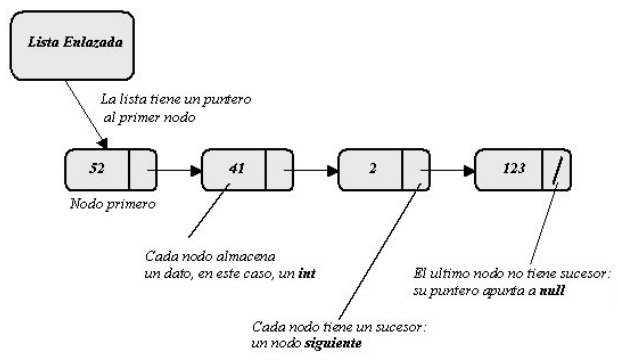
Listas enlazadas

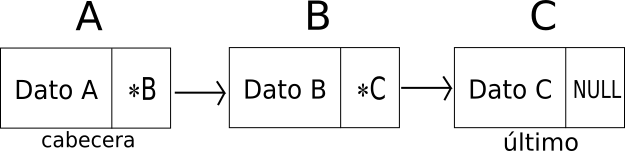
Angiee Karoline Galvis Tarazona – Semestre 1.3 – Ingeniería Multimedia- Profesor: Luis Alberto Reyes Prieto

Una lista enlazada es una colección de datos lineal de elementos llamados nodos. El orden entre ellos se establece mediante punteros; direcciones o referencias a otros nodos y estos pueden contener datos de cualquier tipo, incluyendo objetos de otras clases. En sí es una estructura d datos que desempeña el papel de almacenar y gestionar los datos de manera más dinámica que un arreglo.



Estructura

Cada nodo almacena un dato (información) y un enlace conocido como apuntador que referencia al siguiente nodo, este último definiendo el curso de la lista (→ Nuevo nodo/Null).

Estos nodos se pueden almacenar en ubicaciones aleatorias de la memoria y se conectan mediante los apuntadores, donde se pueden usar los nodos de varias formas.

Para que una estructura sea una lista enlazada, debe tener unos operadores asociados que permitan la manipulación de los datos que contiene. Los operadores básicos de una lista enlazada son:

* **Insertar:** inserta un nodo con dato x en la lista, pudiendo realizarse esta inserción al principio o final de la lista o bien en orden.
* **Eliminar:** elimina un nodo de la lista, puede ser según la posición o por el dato.
* **Buscar:** busca un elemento en la lista.
* **Localizar:** obtiene la posición del nodo en la lista.
* **Vaciar:** borra todos los elementos de la lista.

Usos

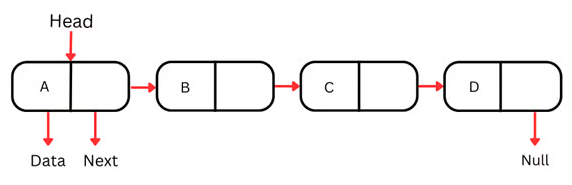
Aunque las listas enlazadas ofrecen ciertas ventajas sobre las listas y matrices normales, también tienen sus limitaciones. Por ejemplo, hay que almacenar punteros para cada elemento para referenciar el siguiente nodo, esto causa que el uso de memoria por elemento sea mayor cuando se utilizan listas enlazadas complejas o de gran tamaño. Además, esta estructura de datos no permite el acceso directo a los datos como sucede con un arreglo. Acceder a un elemento requiere un recorrido secuencial desde el principio de la lista, lo que da lugar a una complejidad de tiempo de búsqueda O(n).

La elección entre utilizar una lista enlazada o una matriz depende de las necesidades específicas de la aplicación. Las listas enlazadas son más útiles cuando:

1. Se necesita insertar y eliminar con frecuencia muchos elementos.
2. El tamaño de los datos es impredecible o puede cambiar con frecuencia.
3. El acceso directo a los elementos no es un requisito.
4. El conjunto de datos contiene elementos o estructuras grandes.

Tipos de nodos

* Listas simples



Una lista enlazada simple es el tipo más sencillo de lista enlazada, en la que cada nodo contiene algunos datos y una referencia al siguiente nodo de la secuencia. Sólo pueden recorrerse en una única dirección: de la cabeza (el primer nodo) a la cola (el último nodo). Cada nodo de una lista con un solo enlace suele constar de dos partes:

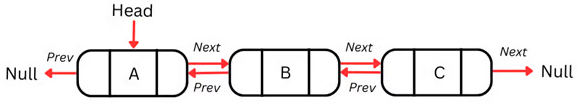
* **Data**: La información real almacenada en el nodo.
* **Siguiente Puntero**: Una referencia al siguiente nodo. El siguiente puntero del último nodo suele ser nulo.

Como estas estructuras de datos sólo pueden recorrerse en una única dirección, para acceder a un elemento concreto por valor o índice hay que empezar por la cabecera y recorrer secuencialmente los nodos hasta encontrar el nodo deseado. Esta operación tiene una complejidad temporal de O(n), por lo que es menos eficaz para listas grandes.

Insertar y eliminar un nodo al principio de una lista de un solo enlace es muy eficaz, con una complejidad básica. Sin embargo, la inserción y la eliminación en el medio o al final requieren recorrer la lista hasta ese punto, lo que conlleva mayor tiempo y espacio en nuestra estructura.

El diseño de las listas simples las convierte en una estructura de datos útil cuando se realizan operaciones que tienen lugar al principio de la lista.

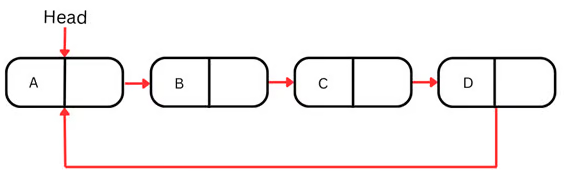
* Listas doblemente enlazadas



Una desventaja de las listas con un solo enlace es que sólo podemos recorrerlas en una dirección y no podemos iterar hacia el nodo anterior si es necesario. Esta restricción limita nuestra capacidad para realizar operaciones que requieran una navegación bidireccional.

Las listas doblemente enlazadas resuelven este problema incorporando un puntero adicional dentro de cada nodo, lo que garantiza que la lista pueda recorrerse en ambas direcciones. Cada nodo de una lista doblemente enlazada contiene tres elementos: los datos, un puntero al nodo siguiente y un puntero al nodo anterior.

* Listas enlazadas circulares



Las listas enlazadas circulares son una forma especializada de lista enlazada en la que el último nodo apunta de nuevo al primer nodo, creando una estructura circular. Esto significa que, a diferencia de las listas simple y doblemente enlazadas que hemos visto hasta ahora, la lista enlazada circular no termina, sino que hace un bucle.

La naturaleza cíclica de las listas enlazadas circulares las hace ideales para situaciones en las que hay que hacer un bucle continuo, como los juegos de mesa que hacen un bucle desde el último jugador hasta el primero, o en algoritmos informáticos como la programación por turnos.

Ejemplo práctico en PHP

En este ejemplo, el código crea una lista con diferentes animes y los muestra como HTML, los elementos se insertan en orden inverso: el último elemento agregado será el primero en la lista. Para hacer esto hay que tener en cuenta 2 componentes claves del código:

* Clase Nodo:
  + Almacena un valor (dato) y un puntero al siguiente nodo (siguiente).
  + Constructor: Inicializa el dato y establece siguiente como null.
* Clase listasAnime:
  + Contiene un nodo inicial (comienzo).
  + **Métodos:**
    - insertar($dato): Agrega nuevos elementos al inicio de la lista.
    - imprimirHTML(): Recorre la lista y genera una lista HTML no ordenada (<ul>) con los datos.

<?php

class Nodo{

public $dato;

public $siguiente;

public function \_\_construct($dato) {

$this->dato = $dato;

$this->siguiente = null;

}

}

class listasAnime{

public $comienzo;

public function \_\_construct(){

$this->comienzo = null;

}

public function insertar($dato){

$nuevoNodo = new Nodo($dato);

$nuevoNodo->siguiente = $this->comienzo;

$this->comienzo = $nuevoNodo;

}

public function imprimirHTML(){

$actual = $this->comienzo;

echo "<ul>";

while($actual != null){

echo "<li>" . $actual->dato . "</li>";

$actual = $actual->siguiente;

}

echo "</ul>";

}

}

$lista = new listasAnime();

$lista->insertar("Bungo Stray Dogs");

$lista->insertar("Vocaloid");

$lista->insertar("Hunter X Hunter");

$lista->insertar("Jibaku Shounen Hanako Kun");

$lista->insertar("Hetalia");

$lista->imprimirHTML();

?>