Estructura de Datos y Algoritmos

Ayudantía 4: LinkedList

Profesor: Yerko Ortiz

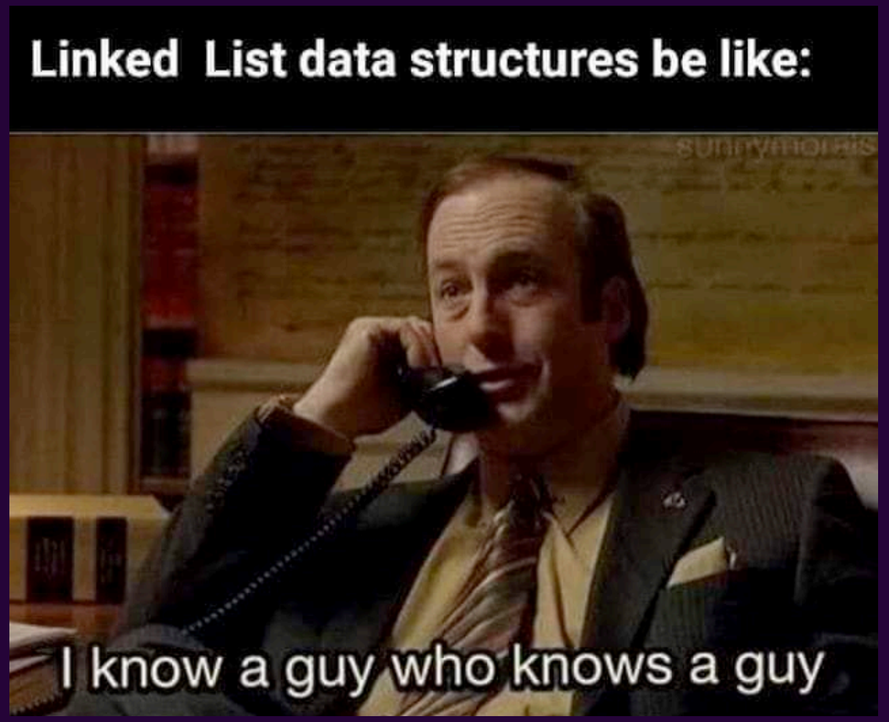
Ayudante: Diego Banda

Contacto

diego.banda@mail.udp.cl

darklouds

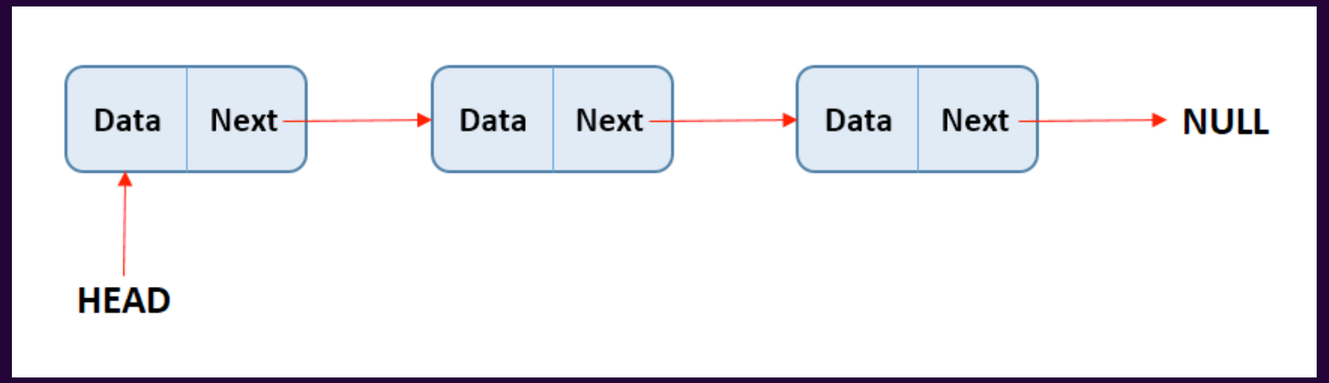
github.com/DiegoBan/EDDA2025-1



¿Qué es una LinkedList?

Lista enlazada de nodos con atributos de datos, y un next.

```java
public class linkedlist{
    Node head;
    public class Node{
        int value;
        Node next;
        Node(int value, Node next){
            this.value = value;
            this.next = next;
        }
        Node(int value){
            this.value = value;
            next = NULL;
        }
    }
    linkedlist(){
        head = NULL;
    }
}
```



¿Diferencias con arreglo?

- Métodos hechos para tener una lista dinámica
- Habilidad para realizar inserciones (con espacio dinámico) en O(1)
- Flexibilidad al conocer tu propia estructura de datos
- La base para cositas verdaderamente interesantes... jeje

Posibles metodos en una linkedlist

¿Big O de cada uno?

- **insertFirst:** Para insertar al inicio.
- **insertLast:** Para insertar al final.
- **insertAt(i):** Insertar en la posición i.
- **at(i):** Buscar valor en posición i.
- **removeFirst():** Elimina el primero.
- **removeLast():** Elimina el último.
- **removeAt(i):** Elimina el elemento en la posición i.
- **removeValue(N):** Elimina el nodo de valor N.

Y una larga lista...

Ejercicio 1: Insertar en una lista ordenada

Tienes listas enlazadas definidas como:

```
class LinkedList {
    class Node{
        int value;
        Node next;
    }
    Node Head;
}
```

Usted está loco por el orden y desea que sus listas enlazadas estén ordenadas siempre de manera ascendente, por lo tanto escribe una función de inserción que en el momento de ingresar los números los ingresa en el espacio correspondiente para que se mantenga el orden.

input: 5, 2, 7, 1, 7, 10, 500, 2

Ej: output: Head = 1 → 2 → 2 → 5 → 7 → 7 → 10 → 500

Ejercicio 2: Mezclar dos listas enlazadas ordenadas

Tienes listas enlazadas definidas como:

Diseñe un algoritmo que reciba de entrada el head de dos listas enlazadas ordenadas ascendentemente y retorne el head de una nueva lista con ambas fusionadas (debe mantener el orden).

```
1 class LinkedList {
2     class Node{
3         int value;
4         Node next;
5     }
6     Node Head;
7 }
```

input

Head 1: 1 → 4 → 8 → 8 → 14

Head 2: -1 → 3 → 5 → 12

Ej: output New Head: -1 → 1 → 3 → 4 → 5 → 8 → 8 → 12 → 14

Ejercicio 3: Notación de posfijo

Diseñe un algoritmo que reciba una expresión aritmetica de posfijo (en string) y retorne el resultado

Notación de posfijo: `2 3 + 3 3 + *` → `(2+3)(3+3)`

Ej:

Input: “2 3 + 3 3 + *”
Output: 30