Día 2 – Arreglos y Listas Enlazadas Teoría: Diferencias entre arreglos (arrays) y listas enlazadas. Tipos: simple, doble y circular. Práctica: Implementar una lista enlazada simple desde cero. Ejercicio clave: Insertar, eliminar y buscar elementos en una lista enlazada.

1. **Arreglos (`arrays`) vs. Listas Enlazadas (`linked lists`)**

Característica	Array	Lista Enlazada
Tamaño	Fijo (al momento de crear)	Dinámico
Acceso a elementos	Acceso directo por índice (O(1))	Recorrido secuencial (O(n))
Inserción/eliminación	Costosa si no es al final (O(n))	Eficiente (O(1) si es al inicio)
Uso de memoria	Contiguo	Disperso, cada nodo apunta al siguiente

2. Tipos de Listas Enlazadas

- **Simplemente enlazada**: Cada nodo apunta al siguiente.
- **Doblemente enlazada**: Cada nodo apunta al anterior y al siguiente.
- **Circular**:
    - Simple: El último nodo apunta al primero.
    - Doble: Como la doble, pero el último conecta al primero y viceversa.

Día 3 – Pilas y Colas Teoría: LIFO vs FIFO. Casos de uso comunes. Práctica: Implementar pila con push y pop. Implementar cola con enqueue y dequeue. Ejercicio clave: Verificar balanceo de paréntesis con una pila.

LIFO vs FIFO

Estructura	Significado	Operaciones clave	Ejemplo real
Pila	LIFO (Last In, First Out)	`push` (apilar), `pop` (desapilar)	Pila de platos, deshacer en un editor
Cola	FIFO (First In, First Out)	`enqueue` (encolar), `dequeue` (desencolar)	Fila del supermercado, impresora

Día 4 – Árboles y Grafos (básico) Teoría: ¿Qué es un árbol? Nodo, raíz, hojas. Árbol binario vs. árbol binario de búsqueda. Qué es un grafo, tipos de representación.

Práctica: Implementar un árbol binario básico e insertar nodos. Ejercicio clave: Recorridos en profundidad: inorden, preorden, postorden.

🌳 Teoría – Árboles

¿Qué es un árbol?

- Es una estructura jerárquica.
- Tiene un **nodo raíz** (root), y **nodos hijos** conectados a través de **ramas**.
- Los nodos sin hijos se llaman **hojas** (leaves).

Árbol binario

- Cada nodo puede tener **como máximo 2 hijos**: izquierdo (`left`) y derecho (`right`).

Árbol binario de búsqueda (Binary Search Tree - BST)

- **Propiedad clave**:
  - Los nodos a la **izquierda** tienen valores **menores**.
  - Los nodos a la **derecha** tienen valores **mayores**.
- Permite búsquedas eficientes (O(log n) en promedio).

🌐 Teoría – Grafos

- Conjunto de **nodos (vértices)** conectados por **aristas (edges)**.
- Pueden ser:
  - **Dirigidos** (con dirección).
  - **No dirigidos** (sin dirección).
  - **Con peso** o **sin peso**.

Representaciones comunes:

1. **Matriz de adyacencia**: tabla NxN.
2. **Lista de adyacencia**: array/lista de listas.