

Video27 (1)

Conceptos Principales

1. a
2. b
3. c

Notas

1. Árbol de montículo (heap)

- **Definición:** Árbol **completo** utilizado principalmente para **estructuras de prioridad y ordenamientos**.
 - **Implementación típica:**
 - Generalmente en un **arreglo** en lugar de punteros.
 - La posición de los hijos se calcula directamente:
 - Subárbol izquierdo: $2*i + 1$
 - Subárbol derecho: $2*i + 2$
 - Esto permite acceso y manipulación rápida de los elementos.
-

2. Propiedades

- **Completo:** Todos los niveles están llenos excepto, posiblemente, el último, que se llena de izquierda a derecha.
 - **Relación padre-hijo:**
 - **Max-heap:** Cada nodo \geq sus hijos.
 - **Min-heap:** Cada nodo \leq sus hijos.
 - **Nota:** No mantiene orden de búsqueda horizontal; solo la relación vertical importa.
-

3. Inserción y ajuste

- **Proceso general:**
 1. Insertar el nuevo elemento en la **última posición** del árbol (arriba del arreglo).
 2. Ajustar el árbol (heapify) para mantener la propiedad de heap (max o min).
 - **Preorden o recorridos:** Se pueden realizar para procesar los elementos según la estructura del árbol, aunque no es un árbol de búsqueda.
-

4. Aplicaciones prácticas

- **Ordenamientos eficientes:** Base de algoritmos como **heapsort**.
- **Colas de prioridad:** Permite acceder rápidamente al elemento máximo o mínimo.
- **Estructuras de datos abstractas:** Optimización de memoria y rapidez en inserciones y eliminaciones.