

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Díaz E.	1-13	Dm-C2	09/07/25; 1:19 PM
Título:	Árboles		
Palabra clave	Tema: Propiedades de los árboles		
-hoja, descendiente	Notas:		
<ul style="list-style-type: none"> • Conexos y sin ciclos ni lados paralelos. • Hay al menos una hoja si hay más de un vértice. • Tiene jerarquía: raíz, hijos, anteriores, descendientes. • Los nodos se dividen en internos (No hoja) y hoja. • Altura es el nivel más profundo del árbol. 			
Preguntas	<p>¿Cómo se mide eficiencia los altos de un árbol?</p>		

Resumen: Las propiedades de los árboles garantizan una estructura ordenada. todo árbol es conexo, no tiene ciclo y posee al menos una hoja. La raíz es en el nivel 0, y cada nodo puede tener hijos y un único padre. hay nodos internos, hojas, descendientes y anteriores. La altura es el numero máximo de niveles del árbol y es crucial para muchos operaciones computacionales.

BY CARLOS RICARDO VILLANUE

NOMBRE	Sebastián J. Lloz, C.	PÁGINAS	2 - 13	PONENTE/CLASE	PN - C2	FECHA - HORA	09/7/25; 01:34 PM
--------	-----------------------	---------	--------	---------------	---------	--------------	-------------------

Título: Árboles

Palabra clave	Tema:
balanceados	Tipos de Árboles
Notas:	<ul style="list-style-type: none"> Por número de hijos: binario, trinario, cuaternario, etc. Árbol binario: máximo 2 hijos por nodo Árbol binario Completo: cada nodo tiene 2 hijos o ninguno Por altura: balanceados (diferencia de niveles máximos) o desbalanceados

Preguntas

¿Qué implicaciones tiene el balanceo en la eficiencia?

Resumen: Los árboles se clasifican según el número de hijos por nodo (binarios, trinarios, etc.) y según su altura (balanceados o no). Un árbol binario tiene un máximo de 2 hijos por nodo, mientras que un binario Completo posee nodos con 2 hijos o ninguno. El balanceo esencial para mantener la eficiencia en operaciones como búsquedas, ya que reduce la diferencia de profundidad entre los hijos, evitando estructuras ineficientes.

BY CARLOS RICHARDO VINUEZA

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastian J. dig E.	3-13	lM-C2	9/6/25; 1:58PM

Título:

Arboles

Palabra clave

- Bosque
- Árbol con peso

Tema:

Bosques y Árboles con peso

Notas:

- Un bosque es un conjunto de árboles, desordenados
- Los árboles pueden verse como bosques
- Se puede pasar de un árbol a un bosque eliminando raíz

Preguntas

¿El Código Huffman
Siempre el más
eficiente?

- Los árboles con peso, sirven para codificar información

Resumen: Un Bosque es una colección de árboles desordenados, muestra que un árbol puede verse como un bosque ordenado. Se pueden conseguir bosques eliminando raíz de un árbol. Además los árboles con peso son útiles en compresión de datos; el Cod. Huffman, por ejemplo, coloca los caracteres más frecuentes más cerca de la raíz, reduciendo así el uso de memoria al asignarles cadenas de bits más cortas.

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Díaz E.	4 - 13	PN - CR	9/4/25; 8:15 PM

Título: Árboles

Palabra clave

Búsqueda en profundidad

Tema: Árboles Generadores y Búsqueda

Notas:

- Un árbol generador es un subgrafo de un grafo conexo que conecta todos los nodos sin ciclos.
- Se obtiene usando búsquedas en profundidad a lo ancho

Preguntas

¿Qué tipo de estructura son los Algoritmos de Prim y Kruskal en grafos densos

- Búsqueda a lo ancho: Explora todos los vecinos primero.
- Búsqueda en profundidad: Explora un camino hasta el final antes de retroceder.
- Árbol generado mínimo: Mantiene la conectividad con el menor costo posible (Usando Prim o Kruskal)

Resumen: Un árbol generador permite conectar todos los nodos de un grafo sin forma ciclos. Puede obtenerse mediante búsqueda en profundidad o a lo ancho, según la estrategia de exploración. Estos métodos son claves para construir estructuras eficientes alrededor de búsquedas minimizando costos. Se aplican algoritmos como el de Prim o Kruskal para formar árboles generados mínimos, útiles en redes eléctricas, rutas, etc...

BY CARLOS RICHARDO VINUEZA

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA		
Sebastian J. Diaz E.	5-13	PM-C2	9/07/25; 2:44 PM		
Título:		Arboles			
Palabra clave	Tema: Arbol binario de busqueda				
-Arbol binario	Notas:				
<ul style="list-style-type: none"> • Permiten insertar y reagrupar datos en forma ordenada • Cada nodo tiene hijo izquierdo (menor o igual) y uno derecho (mayor) • Se raya unico el proceso de comparación 					
Preguntas	<p>¿Qué sucede si todos los datos van en orden ascendente?</p> <p>→ Problema: Pueden volverse desbalanceados si se insertan datos en orden.</p>				

Resumen: El arbol binario de busqueda (ABB) permite organizar datos jerarquicamente donde el nodo izquierdo es menor (O menor) que el padre y el derecho mayor. Esto obtiene la búsqueda de datos haciendo mas rápida la lista desordenada. Sin embargo, si los nodos están bien distribuidos, el arbol puede volverse ineficiente. Se usa como clave en bases de datos y sistemas de información.

BY CARLOS RICARDO VINUEZA

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Pérez E.	6-13	PM - C2	9/07/25; 2:58 PM
Título: Árboles			
Palabra clave	Tema: Apllicaciones de los Árboles		
-Bases de datos	Notas:		
<ul style="list-style-type: none"> • Se usan en bases de datos, compresión, ilosificación y estructuras de datos. • Árboles binarios o AVL: eficaces cuando los datos están en memoria principal. • Árboles B o B++: ideales para manejar grandes volúmenes en disco. • Elegir el tipo de árbol depende del volumen y naturaleza de la información. • Útiles también para eliminar duplicados y ordenar registros. 			
Preguntas	¿Qué árbol es mejor para bases de datos grandes?		

Resumen: Los árboles se utilizan para gestionar información en bases de datos, estructuras dinámicas y algoritmos de compresión. La elección del tipo de árbol (binario, AVL, B) depende del volumen de datos y del acceso a memoria. En sistemas con mucha información, los árboles B son más eficientes porque agrupan varios registros por nodo y reducen la altura del árbol. Se usan mejor la selección y organización.

BY CARLOS RICHARDO VILLANUE

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Big, E.	7-13	PM-C2	9/07/25; 38 (1PM)

Título: Árboles

Palabra clave	Tema:
+ Balanceados	Árboles AVL
- Eficiencia logarítmica	Notas:
Preguntas	<ul style="list-style-type: none"> • Son árboles binarios balanceados automáticamente. • Garantiza eficiencia: la diferencia de altura entre subárboles es 1. • Se reestructuran tras cada inserción o eliminación mediante rotaciones. • Muy usados cuando las operaciones de inserción y consulta son frecuentes. • Permiten búsquedas en tiempo logarítmico incluso con gran cantidad de datos.

Resumen: Los árboles AVL solucionan el problema del desbalanceo en ABB, reorganizando automáticamente su estructura tras inserción o eliminación de datos. Mantienen la eficiencia al asegurar que la altura de los subárboles sea equilibrada. Cualquier requerimiento de trabajo al crear o modificar, permiten búsquedas y acceso muy rápidos.

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Míz E.	8-13	PM - C2	09/07/25; 4:12 P.M.

Titulo:

Árboles

Palabra clave	Tema:
Almacenamiento secundario	Árboles B
Preguntas	<ul style="list-style-type: none"> • Diseñados para manejar grandes contenidos de datos, especialmente en disco. • Permiten múltiples registros por nodo y están balanceados. • Su altura es menor que en otros árboles, lo que reduce el número de accesos. • Cada nodo suele coincidir con un bloque del disco para optimizar velocidades. • Árboles B++ mejoran aún más la eficiencia en consultas.

Resumen: Los árboles B están optimizados para sistemas que almacenan grandes volúmenes de datos en disco. A diferencia de AVL o ABB, sus nodos contienen varios registros y están diseñados para minimizar acceso a disco. Esto estructuras reduce la altura del árbol, mejora la velocidad de búsqueda y escritura y aprovecha mejor los bloques de almacenamiento.

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Díaz E.	9-13	Pn-C2	9/07/25; 5:13pm

Titulo: Introducción a los lenguajes formales.

Palabra clave Lenguaje	Tema: Gramáticas y lenguajes formales.
	Notas: <ul style="list-style-type: none"> • Una gramática es un conjunto de reglas que definen cómo se generan las palabras de un lenguaje. • Los gramáticos se clasifican en 4 tipos según Chomsky: 0, 1, 2 y 3.
Preguntas ¿Qué significa que la gramática sea libre de contexto?	<ul style="list-style-type: none"> • Cada gramática tiene mayor o menor capacidad de representar estructuras complejas. • Los gramáticos tipo 3 son los más simples y se usan con autómatas finitos. • Los gramáticos de tipo 2 son comunes en compiladores y traductores sintácticos.

Resumen: Los gramáticos son mecanismos que definen cómo se generan los palabras válidas de un lenguaje formal. Se dividen en 4 tipos jerárquicos según Chomsky, desde lo más general (tipo 0) hasta lo más simple (tipo 3). Cada tipo tiene diferente poder de expresión.

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Leónel F. M. E.	10-13	DN-C2	9/07/25; 05:38 p.m

Titulo: Introducción a los lenguajes formales

Palabra clave	Tema: Automatos finitos
- Transición - Autómato Finito	Notas: <ul style="list-style-type: none"> • Un automato finito es un modelo matemático que reconoce lenguajes regulares. • Estó compuesto por un conjunto de estados un estado inicial y un conjunto de estados finales y una función de transición.
Preguntas	<p>¿Cuál es la ventaja de convertir un AFN a un AFD?</p> <ul style="list-style-type: none"> • Existen 2 tipos: determinístico (AFD) y no determinístico (AFN) ambos con la misma capacidad de reconocimiento. • Se utilizan en aplicaciones como validación de texto.

Resumen: Los autómatos finitos son modelos que reconocen cadenas válidas en un lenguaje regular, funcionando mediante transiciones entre estados a partir de símbolos de entrada. Existen versiones deterministas (determinísticos) (AFD) y no determinísticas (AFN). Aunque ambas son equivalentes en potencia.

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Lebastón J. Díaz E.	11-13	PM-C2	9/7/24; 6:43

Título: Introducción a los lenguajes formales

Palabra clave	Tema: Máquinas de estado finito (MEF)
- Transición de estado	Notas: Los MEF son modelos similares a los autómatas, pero salidas ademas de estados. • Existen 2 principales tipos: máquinas de Mealy (la salida depende del estado y de la entrada) y de Moore (la salida depende solo del estado). Preguntas ¿Cuáles son las diferencias entre MEF? Cada transición entre estados puede generar una salida específica según el modelo utilizado. • Son fundamentales en ingeniería para simular sistemas reaccionales y reacciones a eventos.

Resumen:	Las máquinas de estado finito son modelos que responden a entrada con salida segun el estado actual. La diferencia de los autómatas es que estas máquinas generan salidas en cada transición (mealy) o por cada estado (moore). Se usan en sistemas informáticos, interfaz y electrónica digital.
----------	---

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Díaz E.	12-13	PN-C2	9/7/25; 6:01 P.M

Título: Introducción a los lenguajes formales

Palabra clave	Tema: Teorema de la Computación
- Computabilidad	Notas: <ul style="list-style-type: none"> La Computabilidad estudio qué problemas pueden resolverse mediante un algoritmo. introduce la máquina de turing como modelo universal de computación. los problemas se clasifican decidibles e indecidibles Es la base para comprender los límites de programación, algoritmos y inteligencia artificial.
Preguntas	<p>¿Qué problema práctico no puede resolverse algorítmicamente?</p>

Resumen: La teoría de la computabilidad analiza qué problemas pueden resolverse mediante algoritmo. Introduce la máquina de turing como modelo teórico de lo que puede hacer el computador. Clasifica los problemas en decidibles e indecidibles. También da paso a la teoría de la complejidad, que evalúa la eficiencia algorítmica.

NOMBRE	PÁGINAS	PONENTE/CLASE	FECHA - HORA
Sebastián J. Díaz E.	13-13	PM-C2	9/7/25; 6:17 P.M

Título: Introducción a los lenguajes formales

Palabra clave - Compiladores	Tema: <u>Aplicaciones de los lenguajes formales</u> Notas: <ul style="list-style-type: none"> • Los lenguajes formales se usan ampliamente en el diseño de Compiladores y analizadores sintácticos. • Permiten Validar cadenas de texto, identificar errores de sintaxis y verificar estructura de programación. • Se aplican en Inteligencia artificial, Videoguegos, protocolos de comunicación y sistemas de control. • Son clave en la definición de lenguajes de programación mediante gramáticas formales.
Preguntas ¿Cómo se usa una gramática formal en un compilador?	

Resumen: Los lenguajes formales tienen múltiples aplicaciones en informática y otras disciplinas. Se usan para construir compiladores, validar la sintaxis de programar y desarrollar el mecanismo de análisis léxico. También se aplican en IA, sistemas de control y videojuegos. Permiten estructuras correctas de la comunicación entre humanos y máquinas.